

UNIVERSIDAD DE ZARAGOZA



**ANÁLISIS ESTADÍSTICO DE MODELOS
HEDÓNICOS STAR CON EFECTOS DE
VECINDAD. UNA APLICACIÓN AL MERCADO
INMOBILIARIO DE ZARAGOZA**

MEMORIA PRESENTADA POR M^a ASUNCIÓN BEAMONTE SAN AGUSTÍN
PARA ACCEDER AL GRADO DE DOCTOR EN CIENCIAS

JULIO 2008

A todos los que me han dado su cariño y
su aliento a lo largo de este viaje

Agradecimientos

Parece de rigor que el primer agradecimiento que aparezca en esta Memoria sea para los directores de esta Tesis. En este caso tengo la completa certeza de que su dedicación en este trabajo ha sido más de la obligada y así quiero hacerlo saber. Mis maestros científicos y compañeros en otras lides, han sido una inestimable e imprescindible ayuda, sin la cual este pequeño proyecto no hubiera llegado a ningún puerto. Además, me ha dado la oportunidad de constatar la máxima biológica, respaldada por la Estadística, de que la variabilidad, frente a la uniformidad, es esencial para la evolución. La diversidad de este pequeño colectivo investigador formada por los dos directores Manuel Salvador, Pilar Gargallo, y esta doctoranda, ha dado lugar a un pequeño trabajo en el campo de la Estadística Aplicada y a un importante enriquecimiento personal más allá de lo profesional. Gracias Manolo, gracias Pilar.

No he de olvidar mencionar, por una parte, a todas las instituciones que han cedido de forma altruista la información para la aplicación desarrollada en la Memoria y por otra, a Luis Fabra quién puso a Manolo en la dirección de los estudios sobre mercados inmobiliarios y a Antonio Aznar por su consejo científico, que en ocasiones puso luz a cuestiones que permitieron avanzar en la realización de este trabajo.

Y por último, a mi familia que convertidos en perfectos alquimistas han sido capaces de elaborar una mixtura a base de tolerancia, cariño y no exenta de cierta perplejidad, que me ha permitido una dedicación a este trabajo, en algunos momentos, sin ninguna duda, excesiva.

Índice

1. Introducción	1
1.1 El modelo hedónico	1
1.1.1. Orígenes del modelo hedónico.....	5
1.1.2. Los modelos hedónicos en el ámbito inmobiliario	6
1.1.3. Elaboración de índices de precios	8
1.1.4. Desarrollos más recientes	9
1.2 La Estadística Espacial	10
1.2.1. La Estadística Espacial en el ámbito inmobiliario.....	14
1.3 La Estadística Espacial Bayesiana	17
1.4 El modelo STAR con efectos de vecindad	21
1.5 Objetivo y descripción de los capítulos	24
2. El modelo espacio temporal autorregresivo con efectos de vecindad	29
2.1 Introducción	29
2.2 El modelo STAR	30
2.3 Estimación del modelo	36
2.4 Validación del modelo	39
2.4.1. Bondad de ajuste	39
2.4.2. Validación extramuestral	42
2.5 Simplificación del modelo.....	44
2.6 Aplicación al mercado de la vivienda.....	47

2.6.1.	Descripción de la base de datos	47
2.6.2.	Análisis exploratorio de la base de datos	50
2.6.2.1.	Características estructurales.....	50
2.6.2.2.	Características dinámicas.....	52
2.6.2.3.	Características de localización.....	54
2.6.2.4.	Dependencias espaciales.....	54
2.6.2.5.	Análisis de la multicolinealidad.....	55
2.6.3.	Variables incluidas en el modelo	56
2.7	Resultados.....	57
2.7.1.	Estimación del modelo.....	58
2.7.2.	Validación del modelo	61
2.7.3.	Simplificación del modelo	72
2.8	Conclusiones.....	75
3.	Inferencia bayesiana en modelos STAR con efectos de vecindad	79
3.1	Introducción	79
3.2	Planteamiento del problema.....	80
3.2.1.	Distribución a priori.....	81
3.2.2.	Distribución a posteriori	83
3.3	Estimación del modelo.....	84
3.3.1.	Distribuciones completamente condicionadas.....	85
3.3.2.	Algoritmo de estimación.....	90
3.4	Validación del modelo	93
3.4.1.	Bondad de ajuste.....	94
3.4.2.	Validación extramuestral	98
3.4.2.1.	Muestreo de importancia secuencial.....	100
3.4.2.2.	Elaboración de predicciones y bondad de ajuste	101

Índice

3.5	Simplificación del modelo	102
3.6	Resultados	108
3.6.1.	Estimación del modelo	108
3.6.2.	Validación del modelo	119
3.6.3.	Simplificación del modelo	125
3.7	Conclusiones	127
4.	Tratamiento continuo de las ponderaciones de los efectos de vecindad	129
4.1	Introducción	129
4.2	Tratamiento continuo de las ponderaciones	130
4.2.1.	Distribución a priori	131
4.2.2.	Distribución a posteriori	131
4.3	Estimación y validación del modelo	132
4.3.1.	Distribuciones completamente condicionadas	132
4.3.2.	Algoritmo de estimación	134
4.4	Resultados	136
4.4.1.	Estimación del modelo	136
4.4.2.	Validación del modelo	147
4.4.3.	Simplificación del modelo	153
4.5	Conclusiones	154
5.	Inferencia bayesiana robusta y predicciones retrospectivas	157
5.1	Introducción	157
5.2	Tratamiento heterocedástico del modelo STAR con efectos de vecindad	158
5.2.1.	Distribución a priori	159

5.2.2. Distribución a posteriori	160
5.3 Estimación y validación del modelo	161
5.3.1. Distribuciones completamente condicionadas	161
5.3.2. Algoritmo de estimación	167
5.4 Resultados	174
5.4.1. Estimación del modelo	174
5.4.2. Validación del modelo	185
5.4.3. Simplificación del modelo	190
5.5 Análisis retrospectivo	195
5.5.1. Predicciones retrospectivas	195
5.5.1.1. Algoritmo	198
5.5.2. Elaboración de índices retrospectivos de la evolución de los precios	199
5.5.2.1. Metodología	199
5.5.2.2. Aplicación	200
5.6 Conclusiones	202
6. Selección de variables en modelos STAR con efectos de vecindad mediante algoritmos genéticos (AG)	205
6.1 Introducción	205
6.2 Planteamiento del problema	208
6.2.1. Selección de variables	208
6.2.2. El modelo	211
6.2.3. Criterios de selección	212
6.3 El problema multiobjetivo	215
6.4 Algoritmo genético	217
6.5 Estimación y validación del modelo	221

Índice

6.5.1. Estimación del modelo.....	222
6.5.2. Validación del modelo	222
6.5.2.1. Validación <i>rolling</i>	225
6.6 Resultados.....	225
6.6.1. Estimación.....	226
6.6.2. Validación	231
6.6.2.1. Validación intramuestral.....	231
6.6.2.2. Validación extramuestral	234
6.6.3. Comparación.....	240
6.7 Conclusiones.....	242
Conclusiones generales y líneas de futura investigación	243
Apéndice A: Computación Bayesiana	245
Apéndice B: Programas informáticos	263
Bibliografía	375

Capítulo 1

Introducción

1.1 El modelo hedónico

La vivienda es un bien peculiar que puede ser estudiado desde diferentes perspectivas. Sociológicamente, la vivienda es el lugar de residencia de las unidades familiares básicas dentro de nuestro sistema. Desde un punto de vista legal, la Constitución española le confiere la categoría de derecho en su artículo 47 (*“Todos los españoles tienen derecho a disfrutar de una vivienda digna y adecuada. Los poderes públicos promoverán las condiciones necesarias y establecerán las normas pertinentes para hacer efectivo este derecho, regulando la utilización del suelo de acuerdo con el interés general para impedir la especulación”*) y a través del cristal económico se la analiza como un bien de consumo o una inversión.

En nuestro país de acuerdo con la Encuesta continua de presupuestos familiares de 2001, en el 85,74% de los hogares españoles el régimen de tenencia de la vivienda es la propiedad y un 14,06% de los hogares disponen, también de una segunda vivienda propia.

La vivienda es, por tanto, considerada un artículo de primera necesidad en España, acarreando importantes consecuencias en diversos niveles de la economía, tales como: el elevado nivel de endeudamiento de las familias que ocasiona su compra, el nivel de riesgo de las entidades de crédito por la cantidad de hipotecas concedidas en los últimos años que sumado al aumento de los tipos de interés, hace crecer el índice de morosidad, sin olvidar el volumen de activos financieros que genera (Fabra, 2005). Todo ello nos puede dar una idea del impacto que la evolución del mercado de la construcción tiene sobre la economía en nuestro país y que hace que la evolución del mercado inmobiliario español constituya un objeto de estudio de extraordinario interés tanto desde un punto de vista práctico como académico.

Además de su carácter heterogéneo, la vivienda presenta algunos aspectos singulares que la diferencian de otros tipos de bienes: es duradero, es uno de los bienes con vida más larga, tiene una localización fija y un alto valor económico.

En las últimas décadas se ha hecho un notable esfuerzo por intentar modelar de forma apropiada el precio de la vivienda, recogiendo todas estas particularidades. La herramienta que desde un punto de vista estadístico-económico se ha utilizado de forma habitual, ha sido el modelo hedónico de precios.

La Teoría de Precios Hedónicos dice que la mayoría de los bienes de consumo están constituidos por un cierto número de atributos, y que su precio total puede ser separado en componentes de precios para cada una de las características, quedando establecido su valor en el mercado en función de los rasgos particulares del mismo.

Un modelo hedónico de precios será, por tanto, capaz de descomponer el precio total de un artículo, obteniendo estimaciones del valor de la

El modelo hedónico

aportación que cada una de las características hace al precio del bien. Habitualmente, los modelos hedónicos se estiman utilizando técnicas del Análisis de Regresión.

Etimológicamente, el término hedónico deriva del griego “*ηδονικος* (hedónicos)” y según definiciones de la Real Academia Española significa: *Pertenciente o relativo al hedonismo o al hedonista. Que procura placer. Relacionado con el placer.*

En el contexto económico se refiere a la satisfacción que el comprador obtiene a través del consumo de bienes y servicios, denominándose utilidades o hedonicismos a las mejoras en sus características. Por ello, a veces, en lugar de observar el aumento de precio de un bien, es común estudiar el incremento del precio que resulta de la mejora de alguna de las características. Estos avances en la calidad de un producto, suponen un costo del que se responsabiliza el productor o vendedor del mismo y que el comprador debe estar dispuesto a aceptar como un aumento en el precio, haciendo en su caso, una valoración positiva de la evolución en las características del artículo, entendiendo que el consumidor obtiene su satisfacción no directamente del bien que adquiere sino de sus atributos.

Por tanto, en términos económicos, estos precios resultan ser una consecuencia de la oferta y la demanda, quedando a su vez influenciados por la coyuntura económica, es decir, en un marco temporal y circunstancial. El modelo de precios hedónicos reflejará el consenso al que llegan productores, vendedores, clientes y de forma general el mercado, al coincidir en la valoración marginal de la característica cuyo precio se quiere estimar.

Otra aplicación ampliamente utilizada, es el ajuste que el modelo hedónico hace de los índices de precios. Con este método se pueden adaptar los índices de precios al consumo para compensar las mejoras en las

características, al controlar el efecto de los cambios en la calidad de los productos, que lógicamente suponen un aumento en su precio.

Pero los modelos hedónicos, lejos de concitar la aceptación general, han sido criticados por presentar aspectos no completamente resueltos y que son controvertidos, tales como la forma funcional del modelo, las variables a incluir en el mismo y el no requerimiento en estos modelos de la segmentación del mercado.

Con respecto a la forma funcional y las variables incluidas, ambas pueden a menudo parecer ad hoc, pero es justo decir que, tanto en un caso como en otro, esto es debido a la dificultad de modelar las aplicaciones prácticas a través de este tipo de modelos.

La forma funcional se suele presentar a través de estructuras básicas tales como la lineal, la semi-logarítmica y la logarítmica-logarítmica. Según Malpezzi (2002) la forma semi-logarítmica tiene como principales ventajas permitir la variación en valor monetario de cada característica, el que sus coeficientes pueden ser fácilmente interpretados como el porcentaje de cambio en el precio debido al cambio en una unidad en la característica y además ayuda a minimizar los problemas de heterocedasticidad.

Respecto a las posibles variables independientes a incluir en el modelo para explicar el precio de un bien, éstas pueden ser muy numerosas y su incorporación suele estar fuertemente condicionada por la facilidad o dificultad en el acceso a la información.

Otros inconvenientes que pueden aparecer en estos modelos en relación con las variables son, por una parte, la alta correlación entre algunas de ellas, que puede crear problemas de estimación incluso si todas las variables no están incorporadas en el modelo, y por otra, la incorrecta

El modelo hedónico

especificación de las mismas, que puede desembocar en una difícil interpretación de los coeficientes individuales.

1.1.1 Orígenes del modelo hedónico

El origen de las aplicaciones de los modelos hedónicos resulta bastante controvertido. Por una parte Colwell y Dilmore (1999) señalan a Haas (1922) como el pionero en su utilización en un estudio dentro del sector agrícola, sin embargo, Griliches (1971) y Goodman (1998) sostienen que la primera aplicación significativa fue llevada a cabo por Court (1939) en la industria del automóvil. Al parecer la figura de Wilcox como Jefe del “Bureau of Labor Statistics” pudiera haber sido el enlace entre los trabajos de ambos investigadores.

Haas (1922) analizó los precios de parcelas agrícolas por medio del valor de compraventa de fincas en el estado norteamericano de Minnesota. Para ello llevó a cabo una regresión múltiple con cuatro variables independientes y como dependiente el precio del acre de tierra, contando con una muestra de 160 fincas. Con poco tiempo de diferencia, otros trabajos de corte similar y también en el terreno de la agricultura se realizaron en otros estados de los Estados Unidos: Wallace (1926) en Iowa, Ezequiel (1926) en Pennsylvania y George (1942) en Illinois.

En nuestro país y también en el campo agrícola existe un interesante precedente en un trabajo de Verdejo (1848) que relaciona el valor de una finca con sus características, agrupadas en dos componentes distintas: “Valoración del terreno en sí mismo” incluyendo en este grupo la estructura y composición de la tierra y “Valoración del terreno con arreglo a su disposición” considerando en éste la situación, pendientes, exposición al sol, proximidad o

distancia a una población, y otros beneficios de la hacienda como el acceso al agua (Calatrava y Cañero, 2000).

Sobre lo que no parece haber duda alguna es que el economista Andrew Court (1939) fue el primero en acuñar el término hedónico¹ para estos modelos (Goodman, 1998) en un trabajo para la “Automobile Manufacturers Association”, en el cual se muestra la justificación en los cambios en los precios de los automóviles por la mejora en sus características, comparando los precios de vehículos en diferentes momentos de las décadas de los años 1920 y 1930 en los Estados Unidos. El trabajo de investigación deja claro que el efecto de los cambios en factores de calidad puede ser determinado estadísticamente y obtener así una serie de precios capaz de representar el valor hedónico de un coche para el consumidor en la suposición de que las características del automóvil en el mismo periodo se mantuvieran constantes.

Estudios posteriores desarrollaron los fundamentos para estimar el valor de utilidad generado por las características de un bien (Lancaster, 1966) y en el caso de precios hedónicos no lineales Rosen (1974) estableció el soporte teórico del modelo de precios hedónicos.

1.1.2 Los modelos hedónicos en el ámbito inmobiliario

Utilizando los fundamentos proporcionados por estos últimos estudios, la aplicación directa de la teoría de precios hedónicos al mercado inmobiliario permite asegurar que el precio de un inmueble vendrá descrito por sus características y reflejando la aportación de cada una de ellas en el valor total.

¹Cita literal del trabajo de Court.

Webster's New Internacional says “Utilitarianism, seeking the good in the greatest happiness of the community as a whole, is the chief hedonistic doctrine” Thus, Hedonic price comparisons are those which recognize the potencial contribution of any commodity, a motor car in this instance, to the welfare and happiness of its purchasers and the community. Alexander Sachs suggested this use of HEDONIC

El modelo hedónico

Como en el caso de un bien de cualquier otro tipo, el estudio del precio de la vivienda requiere el conocimiento de las variables que pueden incidir en el mismo.

De forma general, las variables influyentes en el precio de una vivienda pueden agruparse en las siguientes categorías (Sirmans y otros, 2005):

- **Estructurales:** superficie, edad, nº de baños, nº de habitaciones.
- **Internas:** baños completos, aseos, chimeneas, aire acondicionado, tipo de suelo y sótano.
- **Externas:** plazas de garaje, piscina.
- **Medioambientales:** buenas vistas, vistas al mar o a un lago, contaminación, ruido.
- **Vecindario y situación:** localización, servicios públicos, parques, escuelas, comercios.
- **Ocupación y factores de venta:** condiciones de habitabilidad, estado de conservación, vacante en el momento de la compra, tiempo en el mercado, tendencia temporal.
- **Financiación:** impuestos sobre la propiedad, financiación favorable.

Además de las aquí recogidas, la lista puede incluir otras muchas variables independientes que pueden ser accesibles o no al investigador sin dejar por ello de ser relevantes a la hora de calcular el precio de un inmueble.

Así mismo, no se puede olvidar que la valoración de ciertas características de las viviendas, puede ser muy diferente según el lugar geográfico en el que se encuentre radicada, así como según sean los aspectos culturales y sociales mayoritarios en la sociedad donde el bien ha de ser disfrutado.

En el desarrollo de las técnicas para describir esta dependencia, se han utilizado modelos de regresión que toman como variable dependiente el precio de la vivienda transaccionada, habitualmente en forma de logaritmo con el fin de aumentar el grado de normalidad y homocedasticidad de los términos de error, y como variables independientes aquellas características de la misma que se consideran más relevantes para describir la variación de precios observada (Ridker y Henning, 1967; Li y Brown, 1980; Dubin y Sung, 1990).

Basu y Thiboudeau (1998) estudian el precio de una vivienda como función de sus características estructurales (tamaño, antigüedad, tipo de vivienda, calefacción, ascensor, garaje, trastero, aire acondicionado, número de cuartos de baño, exterior, planta, orientación, piscina comunitaria, etc.) pero además añaden información de ciertas características del vecindario que supuestamente, pueden aportar un valor al inmueble (distancia al centro, a colegios, guarderías, estaciones de metro y autobús, supermercados, niveles de contaminación de la zona, nivel socio-económico de la población del área en la que se encuentra la casa, etc.) para capturar los efectos que la localización tiene sobre el precio de la vivienda.

1.1.3. Elaboración de índices de precios

Una aplicación específica de los modelos hedónicos, mencionada anteriormente, es la construcción de números índices que, en el caso de las viviendas, describen la evolución de los precios, eliminando la influencia de la

calidad de las casas en dicha evolución (Bailey y otros, 1963; Case y Quigley, 1991; Hill y otros, 1997; David y otros, 2002).

Pero en este campo inmobiliario existe una limitación importante en la utilización del modelo hedónico para corregir por calidad de la vivienda, y es que el investigador no observa algunas de las variables que se podrían considerar determinantes o muy relevantes en el precio de un inmueble y que o bien no están recogidas o en algunos casos son difíciles de medir como, por ejemplo, los medios de transporte, el tráfico, la proximidad a los servicios o la calidad de la construcción.

La dificultad de estudiar de forma empírica los efectos de algunas variables que subyacen como omitidas en los datos, conduce a un sesgo en la medida y complica la interpretación de los coeficientes de regresión asociados con las variables mencionadas.

Para evitar este problema de sesgo por variables omitidas, algunos autores (Bailey y otros, 1963; Engle y otros, 1985; Archer y otros, 1996; Goetzman y Spiegel, 1997) utilizan los llamados modelos de medidas repetidas. En estos modelos, los datos corresponden a viviendas vendidas más de una vez y cuyas características no han cambiado en el periodo de tiempo considerado, siendo la variable dependiente la diferencia del logaritmo del precio de dos ventas consecutivas. Sin embargo, este tipo de análisis tiene el inconveniente de que las viviendas que se venden más de una vez suelen ser un bajo porcentaje, por lo que los resultados obtenidos son sesgados y poco eficientes.

1.1.4. Desarrollos más recientes

Aún cuando la localización de una vivienda es fija, la contribución de la misma a su valor puede variar como respuesta a una variedad de cambios

microeconómicos en la zona (construcción de carreteras, instalación de nuevas tiendas, colegios, líneas de autobuses o metros, zonas verdes, etc.). Para capturar estos aspectos dinámicos de la localización algunos autores proponen la utilización de modelos con efectos dinámicos (Pace y otros, 1998b, 2000; Sun y otros, 2005; Gelfand y otros, 1998, 2004), mientras que otros utilizan modelos de regresión local (Can y Megbolugbe, 1997; Case 2000, 2001, 2002; Clapp y otros, 2002; Clapp, 2004), que estiman los parámetros del modelo utilizando, para cada transacción, los datos correspondientes a aquellas más semejantes en el espacio y el tiempo.

Es destacable que la mayoría de las técnicas que en la última década han acudido en ayuda del análisis del mercado inmobiliario, están enmarcadas dentro del ámbito de la Estadística Espacial, la cual permite la incorporación de la localización de una vivienda como un factor imprescindible en la modelización de su precio. El apartado siguiente presenta algunos conceptos generales de Estadística Espacial que pueden servir de fuente de referencia en la lectura de esta Memoria.

1.2 La Estadística Espacial

Gran parte de la información que se maneja desde el punto de vista económico y estadístico se sitúa dentro de un marco espacial. El presupuesto de partida es que la localización de las observaciones marca explícitamente tanto el sentido real del proceso en estudio, como el valor de las variables y su evolución.

Los procedimientos de análisis de datos espaciales son muy variados y de diferente nivel de complejidad, pero todos ellos, desde los más sencillos hasta los más complicados, intentan explorar y demostrar la presencia de

dependencia entre observaciones en el espacio, para posteriormente modelarla y poder explicar el fenómeno de forma más precisa.

Tradicionalmente en Economía se han analizado variables con una evolución temporal, es decir, a través de una curva, algo que en el caso del estudio de una variable en el espacio pasa a hacerse a través de una superficie. El sentido de la flecha del tiempo, tanto a nivel modelizador como predictivo, no tiene duda sobre lo que se llama el pasado, cuya evolución marca el presente y permite la predicción para el futuro. Pero en el caso de un proceso que tiene lugar en el espacio estas delimitaciones quedan bastante desdibujadas. ¿Qué es el equivalente del pasado en una superficie de dos dimensiones? ¿Y el futuro? Observaciones influyentes a una dada y que eran instantes de tiempo en el pasado serán en sentido espacial los vecinos en sentido geográfico. Y las observaciones futuras, sobre las que se aplican técnicas de predicción, son lugares en el espacio en el que la observación de la variable no ha tenido lugar. A éstos se les pretende asignar un valor de acuerdo con lo que acontece en el proceso en general y en su vecindario en particular, ya que las unidades que se encuentran próximas en el espacio son a menudo más parecidas entre sí que las que se encuentran distanciadas.

Históricamente, los primeros modelos espaciales aparecieron en 1907 de la mano de Student quien llevó a cabo un trabajo sobre la distribución de las partículas de levadura en un líquido, estudiando su concentración por unidad de área. Aunque muchos otros investigadores, cuyos trabajos se desarrollaban en el campo de la experimentación, fueron conscientes de la presencia de correlación espacial como Fisher y Yates (1938) entre otros, no fue hasta 1954 de la mano de Whittle cuando apareció algún tipo de modelo poniendo de manifiesto este hecho (ver, para más información, Cressie, 1993).

El reconocimiento de la existencia de dependencias espaciales en los datos y la imposibilidad de replicarlos o de llevar a cabo estudios aleatorizados, obliga a la utilización de otros tipos de modelos estadísticos y diferentes aproximaciones para encontrar soluciones a los problemas específicos que se plantean. La necesidad de contar con modelos que puedan incorporar al mismo tiempo, parte espacial y parte que no lo es, así como datos cualitativos y cuantitativos y todo ello desde una óptica de observaciones referenciadas geográficamente, da lugar a una gran complejidad estadística y a la vez a la generación de enormes bases de datos. De forma tradicional, estas dificultades para aplicar la Estadística Espacial han limitado su atractivo y su ámbito de aplicación. Pero los enormes avances en almacenamiento de datos, computación, algoritmia y software implementado específicamente para el trabajo en Estadística Espacial, así como el avance en sistemas de información geográfica, han hecho que resulten factibles aplicaciones a una gran variedad de situaciones y la resolución de problemas planteados desde áreas de conocimiento tan dispares como la Epidemiología, el Medio Ambiente, la Ingeniería o la Economía, entre otras.

La utilización de técnicas de estimación de Estadística Espacial puede proporcionar beneficios claros, como pueden ser la mejora de la predicción o la realización de inferencias más precisas acerca de los parámetros del modelo que la obtenida, por ejemplo, por medio del método de los mínimos cuadrados el cual tiende a sesgar a la baja los errores estándar en presencia de autocorrelaciones positivas (Pace y otros, 1998a). Además, la utilización de variables espacialmente retardadas o términos de error con dependencias espaciales de carácter local, suele ser una forma efectiva de capturar el efecto de variables omitidas correladas con la localización.

El componente básico de un análisis estadístico espacial son unas localizaciones y un conjunto de valores observados en dichas localizaciones. Atendiendo al tipo de información disponible se distinguen 3 tipos de situaciones, a saber: 1) observaciones correspondientes a un fenómeno continuo en el espacio; 2) datos observados en una red fija de localizaciones y 3) observaciones acerca de la ocurrencia puntual de un conjunto de sucesos en una zona espacial dada.

A partir de estos tipos de datos espaciales surgen diferentes técnicas para abordar su análisis y modelización dando lugar a los llamados modelos referenciados geográficamente, los modelos de áreas y los procesos puntuales, respectivamente.

Los modelos referenciados geográficamente forman parte de lo que se ha denominado *Geoestadística* y se aplican al análisis de la realización, no siempre completa, de un proceso estocástico en una región del espacio continuo. Este tipo de modelos suele utilizar el variograma como herramienta básica para identificar y estimar el modelo. La técnica de predicción más habitual es el krigging, que debe su nombre al ingeniero de minas D. G. Krige (1951) cuyos trabajos inspiraron el desarrollo de las técnicas geoestadísticas.

Los modelos de áreas analizan observaciones que se obtienen en localizaciones determinadas y la predicción no tiene sentido más que en algunos puntos fijos en los que puede aparecer una realización del suceso en estudio.

Tanto en los modelos referenciados geográficamente a nivel puntual como a nivel de área tiene especial importancia la incorporación de las relaciones existentes entre vecindades. Para ello establecen relaciones de contigüidad o de proximidad entre pares de observaciones directamente, en el caso de modelos puntuales, y entre regiones en el caso de modelos de áreas.

Estas relaciones se describen a través de matrices \mathbf{W} y deben ser definidas atendiendo al tipo de problema y al nivel de información al que se trabaje.

En el caso de grandes bases de datos el tratamiento computacional de algunas de estas matrices puede resultar de una enorme complejidad y a menudo se presenta como prácticamente intratable. Una referencia notable es Pace y Barry (1997) que utilizan técnicas basadas en el tratamiento de matrices dispersas.

Finalmente, en el caso de los procesos puntuales, se dispone de información acerca de las localizaciones en las que ha ocurrido un determinado suceso. El objetivo es analizar los patrones que determinan dichas ocurrencias para lo cuál se utilizan procesos espaciales tipo Poisson y generalizaciones de los mismos.

1.2.1. La Estadística Espacial en el ámbito inmobiliario

El mercado inmobiliario es una de las áreas económicas en las que más se han aplicado y se aplican las técnicas estadísticas de análisis espacio-temporal debido especialmente a la conjunción de factores tales como: la importancia de la localización, el contexto dinámico en el que tiene lugar el proceso, el enorme volumen de datos a manejar y el gran interés que ofrece el estudio del mismo. Además, es importante resaltar que, a diferencia de lo que ocurre en otras áreas, este campo no es de reciente creación y, por tanto, existe una experiencia acumulada desde otras ópticas como la Macroeconomía, el mundo de la tasación o el quehacer cotidiano de los agentes de la propiedad inmobiliaria. Todos estos colectivos y campos de estudio han aportado importantes conocimientos al estudio del mercado de la vivienda, tanto desde un punto de vista científico como desde la heurística, que en ningún modo deben ser desdeñados. En particular, muchas de sus técnicas de valoración,

que sin lugar a dudas, han sido herramienta útil para el desenvolvimiento práctico de su trabajo dentro de este mercado, se han incorporado en la modelización espacio-temporal del precio de vivienda.

Sin embargo, el complejo análisis del mercado inmobiliario adolece, como ya se ha mencionado con anterioridad, de falta de información acerca de todas las características relevantes para el problema en general y, especialmente, sobre algunas de carácter espacio-temporal. Por dicha razón, se suele observar la presencia de correlaciones espacio-temporales en los residuos de los modelos.

Entre los trabajos existentes en la literatura cabe destacar los trabajos de Dubin (1988, 1992, 1998), Chica-Olmo (1995) o Basu y Thibodeau (1998) entre otros, que utilizan modelos hedónicos que recogen las dependencias espaciales no explicadas por las variables hedónicas mediante la utilización de modelos SAR (Simultaneous Autorregressive Models) que modelan directamente la matriz de varianzas y covarianzas de los residuos. Por su parte, existen trabajos (Pace y Gilley, 1998) basados en la utilización de modelos CAR (Conditional Autorregressive Models) o, más generalmente, a través de efectos aleatorios (Gelfand y otros, 1998) que modelan la inversa de la matriz de varianzas y covarianzas. Un buen estudio comparativo del uso de ambos tipos de modelos en el análisis de precios de vivienda puede encontrarse en Militino y otros (2004).

De entre los modelos espacio-temporales con aplicaciones al mercado inmobiliario desarrollados recientemente, y que consideran de forma conjunta los efectos espaciales y temporales, resulta destacable la corriente de modelos que se construyen incorporando matrices que utilizan información de ventas de viviendas de similares características y situadas en la cercanía para capturar efectos espaciales y temporales de carácter local. Así, Can y Megbolugbe

(1997) usan viviendas de características similares a una dada para construir variables promedio que incorporan en un modelo hedónico y demuestran que son significativas a la hora de predecir el precio de una vivienda. Pace y otros (1998b, 2000), por su parte, presentan un modelo autorregresivo con efectos de vecindad que modelan mediante filtros lineales locales aumentando, de esta forma, la precisión de las estimaciones de los parámetros del modelo.

Actualmente, la creciente capacidad de los ordenadores y los sistemas de información, particularmente los sistemas de información geográfica (GIS), han hecho posible la construcción de grandes bases de datos que contienen información espacio-temporal de la vivienda a un nivel de microdato. En este contexto han aparecido nuevas técnicas estadísticas de carácter semiparamétrico, que permiten la construcción de modelos que describen las dependencias espacio-temporales en una forma más flexible y realista, haciendo posible la obtención de inferencias más exactas acerca de los parámetros del modelo así como de mejores predicciones. Entre otros caben destacar los trabajos de LeSage (2004) que hace uso de modelos de regresión geográficamente ponderada (GWR); Clapp (2004) que utiliza técnicas de modelos de regresión local; Bao y Wan (2004) que emplean splines para construir la función hedónica; Case (2000, 2001, 2002) que propone técnicas de análisis cluster para determinar distritos en los que el comportamiento de los precios es homogéneo o Gelfand y otros (2004) que presentan modelos bayesianos con efectos aleatorios entre muchos otros. Case y otros (2004) realizan un estudio comparativo de algunas de dichas técnicas poniendo de manifiesto los pros y contras de cada una de ellas.

En este sentido, las técnicas estadísticas bayesianas constituyen un conjunto de herramientas de enorme potencial de modelización en este contexto, tal y como lo demuestra el reciente libro de Banerjee y otros (2004),

en el que se describen multitud de aplicaciones a diversos problemas de Estadística Espacial en el ámbito geoestadístico a nivel puntual y de áreas.

Dado que la mayor parte de esta Memoria se enmarca en el ámbito bayesiano, en la siguiente sección se describen las aportaciones más recientes de este enfoque en el análisis del mercado inmobiliario.

1.3 La Estadística Espacial Bayesiana

La Estadística Bayesiana es una parte de la Estadística Inferencial que, apoyada en una base axiomática clara (ver, por ejemplo, Jeffreys, 1961; De Finetti, 1973; Berger, 1985; Bernardo y Smith, 1994; O'Hagan y Forster, 2004 o Parent y Bernier, 2007, entre muchos otros) se caracteriza por el uso del Cálculo de Probabilidades como herramienta para medir todas las fuentes de incertidumbre asociadas a los procesos de estimación, predicción, comparación y selección de modelos que conlleva el análisis de un problema concreto.

Para la Estadística Bayesiana tanto los datos como los parámetros del modelo son variables aleatorias, de forma que un modelo bayesiano especifica una distribución de probabilidad conjunta que busca describir la incertidumbre asociada, por un lado, al proceso generador de los datos y, por el otro, al proceso de estimación de los parámetros (ver Apéndice A para más detalles). Una vez obtenidos los datos, las inferencias acerca de los parámetros del modelo (estimación puntual y por intervalos, contraste de hipótesis, elaboración de predicciones, etc.) se realizan a partir de su distribución a posteriori que se determina mediante la aplicación del Teorema de Bayes (y de ahí el nombre de Estadística Bayesiana). Así mismo, la metodología proporciona herramientas para elaborar predicciones y llevar a cabo los

procesos de comparación y selección de modelos, incorporando la incertidumbre (en términos probabilísticos) asociada a su estimación y selección (ver para más detalles el Apéndice A y los capítulos 3 a 6 de esta Memoria).

Todo ello conforma un conjunto de conceptos y herramientas metodológicas que hace a la Estadística Bayesiana muy atractiva para la modelización de gran cantidad de problemas y, en particular, los relacionados con el Mercado Inmobiliario.

Algunas de las razones que justifican dicho atractivo son las siguientes:

a) Las inferencias bayesianas se hacen condicionalmente a los datos observados incorporando de forma natural la información que los mismos proporcionan acerca del modelo, algo que, en nuestra experiencia, demandan la mayor parte de los investigadores aplicados (ver, en este contexto el apéndice B de Bernardo y Smith (1994)).

b) Las inferencias son exactas y no dependen de resultados asintóticos de dudosa aplicación en el ámbito de la Estadística Espacial, máxime si el estudio se realiza en un contexto temporal, donde las condiciones exigidas para su aplicación suelen ser estrictas y difíciles de verificar en la práctica (ver Stein, 1999).

c) Permite incorporar restricciones sobre los parámetros del modelo así como información externa a la contenida en los datos mediante la distribución a priori.

d) Realiza un tratamiento natural de los datos ausentes así como de las variables latentes que subyacen en las dependencias espacio-temporales utilizadas en la literatura, introduciéndolas como parámetros adicionales del modelo. Ello proporciona un mayor grado de flexibilidad a la hora de

modelarlas tal y como demuestra, convincentemente, el trabajo de Gelfand y otros (2004).

e) Permite especificar criterios no estándar de comparación y selección de modelos que se adecúen a los objetivos del estudio. Para ello basta con plantear el problema como uno de decisión en el que la función de utilidad o pérdida venga definida de acuerdo al objetivo del estudio. Aplicando herramientas estándar de Teoría de la Decisión puede llevarse a cabo, de forma sencilla, el proceso de selección del mejor modelo .

Estas y otras razones hacen muy atractivos los modelos y métodos bayesianos como herramientas de gran potencia para la resolución de gran cantidad de problemas espacio-temporales, tal y como lo demuestra el reciente libro de Banerjee y otros (2004).

Sin embargo, los métodos bayesianos vivieron tiempos de impopularidad marcados por el obstáculo que presentaba su aplicación de forma extensa. Sus virtudes, conocidas desde el punto de vista teórico, chocaban frontalmente con la dificultad asociada al proceso de cálculo, que era inabordable en la práctica, no existiendo a su vez tecnología computacional capaz de resolverla. Además, la subjetividad presente en la elección de distribuciones a priori, hace que muchos miren con suspicacia esta metodología (ver apéndice A para más detalles). Conviene hacer notar, sin embargo, que si el número de observaciones es grande (hecho que habitualmente ocurre en los problemas inmobiliarios cuando se trabaja a nivel de transacciones), la elección de la distribución a priori deja de ser un problema, puesto que su influencia queda amortiguada por la mayor cantidad de información presente en los datos.

Respecto al problema de cálculo, la aparición de los métodos MCMC (*Markov Chain Monte Carlo*) en la década de 1990 resultó ser una revolución

en el cálculo bayesiano. Métodos como el *Gibbs sampling* (Gelfand y Smith, 1990) o el algoritmo de Hastings-Metropolis existente desde los años 50 (Metropolis y otros, 1953; Hastings 1970), y rescatado para la Estadística en los años 90, acompañados de un desarrollo imparable en cuanto a la capacidad de computación y almacenamiento de los ordenadores, han hecho posible la aplicación de los métodos bayesianos a una gran cantidad de problemas.

El análisis bayesiano permite, así mismo, el tratamiento de modelos complejos, que difícilmente pueden ser atacados desde ópticas tradicionales. Un ejemplo son los llamados *modelos jerárquicos bayesianos* muy utilizados en aquellas situaciones en las que el modelo estadístico lleva implícitos varios niveles, admite efectos aleatorios o presenta complicadas estructuras de dependencia (Gelman y otros, 2004). Dichos modelos hacen posible la incorporación de estructuras flexibles mediante el encadenamiento condicional de modelos más simples. Así mismo, los modelos bayesianos presentan herramientas para resolver algunos otros problemas de interés, como es el de la alineación incorrecta. Esta situación, habitual en problemas espaciales, aparece cuando se dispone de información con distinto nivel de agregación (ver, por ejemplo, el capítulo 6 de Banerjee y otros, 2004).

En la literatura existen relativamente pocos trabajos en el campo de la vivienda, que se hayan tratado desde una perspectiva bayesiana. Los más destacados son los realizados por el profesor Gelfand y sus diversos coautores que utilizan modelos hedónicos jerárquicos bayesianos para analizar este tipo de problemas. Así, Gelfand y otros (1998) proponen un modelo espacio-temporal bayesiano que utiliza efectos aleatorios para capturar las dependencias espacio-temporales existentes en un modelo hedónico de precios. En un trabajo posterior, Gelfand y otros (2001) lo extienden permitiendo diferentes procesos para el término del error. En Gelfand y otros

(2003) se propone un modelo hedónico con coeficientes que varían espacialmente. La introducción de variables temporales dummy, utilizadas como un proxy para medir la influencia de factores dinámicos de tipo económico, político, sociológico, etc. en la evolución de dichos precios se puede ver en Gelfand y otros (2004) que, como se ha dicho anteriormente, proporciona un marco muy flexible para describir diversas formas de interaccionar las dependencias espaciales y temporales presentes en un estudio dinámico de los precios de vivienda. Majumdar y otros (2006) utilizan modelos hedónicos con coeficientes que varían espacialmente para calcular gradientes que muestran las zonas de cambio de los precios del suelo en una zona urbana. Recientemente, y desde una óptica bayesiana Banerjee y Gelfand (2006) estudian los cambios rápidos de gradiente en superficies (*wombling*) en las que se modeliza un proceso espacial con puntos georeferenciados.

En un contexto diferente es interesante mencionar los trabajos de LeSage (1999, 2004) que utiliza herramientas bayesianas para el tratamiento de modelos SAR y GWR y, más recientemente, Sun y otros (2005) que realizan un tratamiento bayesiano robusto del modelo con efectos de vecindad propuesto por Pace y otros (1998b, 2000).

1.4 El modelo STAR con efectos de vecindad

En modelos espacio-temporales los beneficios de recoger la dependencia de los errores sobre el tiempo y sobre el espacio de forma independiente son sobradamente conocidos pero poco realistas puesto que lo habitual es que existan interacciones entre ellos. Sin embargo, el modelado conjunto de dichas dependencias no ha sido aplicado de forma extensa a datos de tipo económico debido, fundamentalmente, a dificultades técnicas. La

forma óptima de especificar interacciones espaciales, temporales y espacio-temporales es una cuestión difícil de resolver por el problema que plantea la conciliación de ambas escalas. No obstante, si la especificación conjunta de los errores en espacio y tiempo funcionara adecuadamente, se podría obtener un marco interpretable sobre la dependencia del error.

Un intento con notable interés en este terreno lo proporciona el modelo presentado por Pace y otros (1998b, 2000). En dichos trabajos se propone una familia semiparamétrica de modelos lineales autorregresivos espacio-temporales (STAR) que permite capturar, de forma no paramétrica, una gran cantidad de efectos de carácter local. El método utilizado para la estimación es el de mínimos cuadrados ordinarios y, por tanto, no resulta muy costoso desde un punto de vista computacional. Estos autores aplican su metodología al análisis de precios de vivienda, obteniendo mejores resultados que otras técnicas más complejas con mayor número de parámetros.

Basándose en la máxima presente en Estadística Espacial de que, dentro de un conjunto de datos en el espacio y/o en el tiempo las observaciones que se encuentran próximas suelen ser similares, Pace y otros (1998b, 2000), proponen la utilización de filtros lineales exponencialmente ponderados de carácter local, tanto en el espacio como en el tiempo que permiten, correctamente utilizados, capturar gran cantidad de dependencias espacio-temporales, eliminando o, al menos, debilitando, su influencia en la estimación de los coeficientes hedónicos del modelo.

Esta idea dentro del mercado inmobiliario, tiene un claro referente en la práctica habitual entre los tasadores, que para llevar a cabo la valoración de un inmueble, utilizan la información procedente de transacciones que han tenido lugar inmediatamente antes en el tiempo, dentro de la misma área y con

características similares a la vivienda que se pretende tasar, lo que conduce directamente a la noción de viviendas comparables.

El tratamiento propuesto por Pace y otros (1998b, 2000) aunque muy flexible y sencillo desde un punto de vista computacional, al estar basado en el criterio de los mínimos cuadrados como método de estimación, adolece en nuestra opinión de los siguientes defectos:

1) La interpretación de los resultados obtenidos no suele ser clara al estar mezclados los efectos de las variables hedónicas y los efectos espacio-temporales de forma no lineal.

2) El proceso de inferencia acerca de los parámetros del modelo está basado en criterios frecuentistas asintóticos que suponen conocidos los valores de los parámetros de vecindad. Esta hipótesis es poco realista y aunque se puede hacer un estudio de sensibilidad con respecto a dichos parámetros, es más adecuado incorporar la incertidumbre asociada a la estimación de los parámetros en todo el proceso de inferencia.

3) El proceso de simplificación y/o comparación y/o selección de modelos está basado en el estudio de los estadísticos t y contrastes de modelos anidados. Dado que pueden aparecer modelos no anidados, estos procedimientos (válidos asintóticamente) son claramente insuficientes.

4) Los autores no realizan estudios de validación extramuestral de los modelos estimados que evalúen la capacidad predictiva de los mismos

Responder a éstas y otras cuestiones constituye el objetivo de la Memoria que aquí se presenta.

1.5 Objetivo y descripción de los capítulos

El objetivo principal de la Memoria es desarrollar una metodología bayesiana de estimación, contraste, comparación, validación, predicción y selección de modelos STAR con efectos de vecindad similares a los propuestos en Pace y otros (1998b, 2000).

La nota más relevante de la metodología propuesta es que incorpora la incertidumbre asociada a la estimación de los parámetros de vecindad a todos los procesos anteriores, utilizando procedimientos exactos de inferencia sobre todos los parámetros del modelo sin tener que recurrir a resultados asintóticos.

Este marco proporciona las condiciones para llevar a cabo la aplicación de dichos modelos al análisis de datos del mercado inmobiliario de forma más flexible y realista. En particular y a título ilustrativo la metodología elaborada se aplica al análisis de la evolución espacio-temporal del mercado inmobiliario en un área de Zaragoza.

La Memoria se ha estructurado en 6 capítulos y dos apéndices cuyo contenido es el siguiente:

En el presente capítulo y con el fin de hacer la Memoria autocontenida y facilitar así su lectura, se comentan los antecedentes de los modelos hedónicos, la Estadística Espacial y la Estadística Espacial Bayesiana, especialmente en el mercado inmobiliario, así como el modelo espacio-temporal autorregresivo STAR presentado por Pace y otros (1998b, 2000), como precedente directo del trabajo que aquí se expone.

En el capítulo 2 se presenta de forma pormenorizada el modelo STAR de Pace y otros (1998b, 2000) con las modificaciones que se proponen para simplificarlo. El modelo se estima desde un punto de vista frecuentista a través del método de la máxima verosimilitud y se utilizan procedimientos de

validación intra y extramuestral que permiten evaluar el grado de ajuste, así como la capacidad predictiva del modelo. Además, se describe un procedimiento de simplificación del modelo para poner de manifiesto cuáles de las variables hedónicas y efectos espacio-temporales de vecindad son los más significativos para explicar la evolución observada de la serie de precios. Finalmente, se presenta la base de datos con observaciones del mercado inmobiliario de Zaragoza que será utilizada para mostrar las aplicaciones en cada uno de los capítulos de esta Memoria y se muestran los resultados de la metodología propuesta en este capítulo con sus interpretaciones.

El capítulo 3 plantea un análisis bayesiano para estimar el modelo propuesto con el fin de incorporar la incertidumbre en el proceso de estimación. La aproximación se basa en realizar un tratamiento discreto de los parámetros de vecindad que resulta computacionalmente atractivo a la hora de estimar los parámetros del modelo. Se proponen, además, dentro del marco bayesiano, procesos de validación intra y extramuestral así como una simplificación del modelo. En particular, se plantea un método de muestreo secuencial de importancia que facilita la implementación de procesos bayesianos de validación tipo *rolling* en modelos STAR. El capítulo finaliza con la aplicación a la base de datos y la interpretación de los resultados.

Aún cuando la aproximación bayesiana propuesta en el capítulo 3 es simple computacionalmente puede, en algunas ocasiones ser poco flexible, puesto que requiere que las redes de los posibles valores de los parámetros de vecindad sean pequeñas. Con el fin de extender el procedimiento a situaciones más generales y realizar un análisis más realista, en el capítulo 4 se generaliza el tratamiento presentado en el capítulo anterior mediante el uso de redes más amplias para la elección del número de vecinos espaciales y temporales, así como de distribuciones a priori continuas para sus ponderaciones.

El capítulo 5 realiza una extensión del modelo que permite realizar inferencias robustas acerca de los parámetros del mismo. Para ello se utilizan distribuciones condicionalmente heterocedásticas en los términos de error del modelo que dotan de más peso a las colas de la distribución incondicional de los mismos, debilitando la posible influencia de observaciones extremas en el proceso de estimación. La metodología se aplica a la base de datos y se interpretan los resultados. El capítulo finaliza con la presentación de un método de predicciones retrospectivas que permite estimar el precio de una vivienda de características dadas en un momento dado del pasado, cuestión importante en ámbitos periciales y judiciales. Este método se aplica a la elaboración retrospectiva de números índice para precio de vivienda mediante modelos hedónicos.

En el capítulo 6 se aborda el problema de la determinación de las variables hedónicas y los efectos espacio-temporales más relevantes de un modelo STAR con efectos de vecindad, aplicando técnicas multicriterio de selección de variables que utilizan diversos criterios para evaluar la bondad de ajuste y el comportamiento predictivo del modelo. El problema planteado resulta analíticamente intratable y, por tanto, se recurre al empleo de algoritmos genéticos para su resolución. Así mismo, se propone el uso de mixturas de los modelos más eficientes para explicar la evolución de los precios de vivienda, así como, para determinar las variables hedónicas y efectos espacio-temporales de vecindad más significativos. Al igual que en capítulos anteriores, la metodología se ilustra con una aplicación a la base de datos utilizada en la Memoria.

Finalmente, se remarcan los resultados más destacados de la Memoria y se mencionan someramente algunas de las líneas de investigación en las que actualmente estamos trabajando.

Objetivo y descripción de los capítulos

Se incluyen, en último lugar, dos apéndices, uno sobre Computación Bayesiana y otro conteniendo los programas informáticos que implementan los algoritmos desarrollados en la Memoria.

Capítulo 2

El modelo espacio temporal autorregresivo con efectos de vecindad

2.1 Introducción

En este capítulo se presenta el modelo espacio temporal autorregresivo (STAR) con efectos de vecindad de Pace y otros (1998b, 2000), así como una reparametrización del mismo, que supone una reducción notable del número de parámetros del modelo, y permite una interpretabilidad más clara de los coeficientes. Además, se introducen modificaciones en las matrices de proximidad espacial S y temporal T , con el fin de posibilitar la elección sistemática de los parámetros de vecindad.

La metodología descrita se aplica, con fines ilustrativos, al análisis del mercado inmobiliario dentro de la ciudad de Zaragoza.

El contenido del capítulo se estructura de la siguiente manera: en la sección 2.2 se explicita el modelo STAR con efectos de vecindad y se presentan las modificaciones que se llevan a cabo en esta Memoria, por medio de una reparametrización del mismo, exponiéndose las novedades en la

construcción de las matrices de vecindad espacio-temporales. Las secciones 2.3, 2.4 y 2.5 describen los procesos de estimación, validación (intra y extramuestral) y simplificación del modelo, respectivamente. Posteriormente, en la sección 2.6 se introduce el contexto en el que se aplicará el modelo, describiéndose la base de datos utilizada en la Memoria; los resultados obtenidos de dicha aplicación se describen en la sección 2.7. Finalmente, la sección 2.8 concluye exponiendo, de forma breve, los resultados más relevantes del capítulo.

2.2 El modelo STAR

Considerar un conjunto de n transacciones de vivienda ordenadas de forma no decreciente por tiempo de forma que la primera transacción es la más antigua, la segunda la que tuvo lugar a continuación en el tiempo y la última, la más reciente.

Sean $\{p_i; i=1, \dots, n\}$ los precios de dichas transacciones.

Sean Z_1, \dots, Z_q un conjunto de q características hedónicas de cada transacción que son explicativas de su precio y que no presentan retardos espaciales, temporales y/o espacio-temporales.

Sean X_1, \dots, X_k un conjunto de k variables hedónicas explicativas de dichos precios que presentan retardos espaciales, temporales y/o espacio-temporales.

El modelo STAR parte de la siguiente ecuación:

$$(\mathbf{I}-\mathbf{W})\mathbf{Y} = \mathbf{Z}\boldsymbol{\alpha} + (\mathbf{I}-\mathbf{W})\mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon} \quad (2.1)$$

donde

El modelo STAR

$\mathbf{Y} = (y_i; i=1, \dots, n)$ es el vector $(n \times 1)$ donde $y_i = \log p_i$ ¹.

$\mathbf{Z} = (\mathbf{z}_1, \dots, \mathbf{z}_q)'$ es la matriz $(n \times q)$ de datos correspondiente a las variables Z_1, \dots, Z_q con $\mathbf{z}_i = (z_{i1}, \dots, z_{iq})'$ las características correspondientes a la i -ésima observación.

$\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_q)'$ es el vector $(q \times 1)$ de coeficientes hedónicos de las variables Z_1, \dots, Z_q que cuantifican su tipo y grado de influencia en el precio de la transacción.

$\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_n)'$ es la matriz $(n \times k)$ de datos correspondiente a las variables independientes X_1, \dots, X_k con $\mathbf{x}_i = (x_{i1}, \dots, x_{ik})'$.

$\boldsymbol{\beta} = (\beta_1, \dots, \beta_k)$ es el vector $(k \times 1)$ de coeficientes hedónicos de las variables X_1, \dots, X_k que cuantifican su tipo y grado de influencia en el precio de la transacción.

\mathbf{W} es una matriz $(n \times n)$ de pesos espacio-temporales, estocástica por filas, donde w_{ij} determina la influencia ejercida por la transacción j -ésima sobre la transacción i -ésima.

$\boldsymbol{\varepsilon} \sim N_n(\mathbf{0}, \sigma^2 \mathbf{I}_n)$ es el vector de errores del modelo donde $N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denota la distribución normal multivariante p -dimensional cuyo vector de medias es $\boldsymbol{\mu}$ y su matriz de varianzas y covarianzas es $\boldsymbol{\Sigma}$.

Puesto que solamente las transacciones ocurridas con anterioridad a una dada pueden ejercer algún tipo de influencia sobre su precio, la matriz \mathbf{W} será triangular inferior.

Pace y otros (1998b, 2000) introducen técnicas para acelerar el proceso de estimación, basadas en una descomposición de los efectos de vecindad,

¹ Se toma la transformación logarítmica con el fin de aumentar el grado de normalidad de los datos debido a la forma asimétrica a la derecha que habitualmente presenta la distribución de los precios.

tanto espaciales como temporales, que maximizan el uso de matrices dispersas (*sparse*). La descomposición consiste en dividir la tradicional matriz de pesos \mathbf{W} en otras dos \mathbf{S} y \mathbf{T} de forma que:

$$\mathbf{W} = \phi_T \mathbf{T} + \phi_S \mathbf{S} + \phi_{ST} \mathbf{ST} + \phi_{TS} \mathbf{TS} \quad (2.2)$$

donde la matriz \mathbf{T} especifica las relaciones temporales, \mathbf{S} las espaciales y $\{\phi_T, \phi_S, \phi_{ST}, \phi_{TS}\} \in (-1,1)$ son los coeficientes autorregresivos del modelo que cuantifican el tipo y grado de influencia de los efectos espacio-temporales no recogidos por la parte hedónica del modelo.

La matriz $\mathbf{T} = (t_{ij})$ cumple un papel similar en el tiempo al que tiene el operador retardo en series temporales pero con una periodicidad irregular. Dicha matriz especifica las relaciones temporales existentes entre las transacciones, de forma que t_{ij} especifica la influencia temporal ejercida por la transacción j -ésima sobre la i -ésima. De forma análoga se comporta \mathbf{S} en el espacio, especificando las relaciones espaciales existentes entre las transacciones, de forma que s_{ij} especifica la influencia temporal ejercida por la transacción j -ésima sobre la i -ésima.

Sustituyendo la expresión (2.2) en (2.1) y renombrando los coeficientes, el modelo STAR propuesto en el trabajo de Pace y otros (1998b, 2000) es el siguiente:

$$\begin{aligned} \mathbf{Y} = & \mathbf{Z}\boldsymbol{\alpha} + \mathbf{X}\boldsymbol{\beta}_1 + \mathbf{TX}\boldsymbol{\beta}_2 + \mathbf{SX}\boldsymbol{\beta}_3 + \mathbf{STX}\boldsymbol{\beta}_4 + \mathbf{TSX}\boldsymbol{\beta}_5 + \\ & + \phi_T \mathbf{TY} + \phi_S \mathbf{SY} + \phi_{ST} \mathbf{STY} + \phi_{TS} \mathbf{TSY} + \boldsymbol{\varepsilon} \end{aligned} \quad (2.3)$$

Este modelo es complejo e involucra un gran número de parámetros dificultando el proceso de estimación e interpretación. Con el fin de facilitar ambos aspectos, Pace y otros (1998b, 2000) deciden, tras un proceso de comparación de modelos anidados, seleccionar uno más simplificado. El nuevo modelo aparece como resultado de la supresión de aquellos

El modelo STAR

parámetros que no resultan significativos y no proporcionan una clara interpretación de su efecto.

Así mismo, proponen posibles reparametrizaciones sobre los coeficientes ϕ que conducen a modelos más parsimoniosos e interpretables, tanto en términos económicos como espaciales y temporales.

En particular, Pace y otros (2000) proponen la reparametrización:

$$\beta_2 = -\phi_T \beta_1 \quad \beta_3 = -\phi_S \beta_1 \quad \beta_4 = -\phi_{ST} \beta_1 \quad \text{y} \quad \beta_5 = -\phi_{TS} \beta_1 \quad (2.4)$$

y la ecuación (2.3) del modelo queda como:

$$\mathbf{Y} = \mathbf{Z}\boldsymbol{\alpha} + \mathbf{X}\boldsymbol{\beta} + \phi_T \mathbf{T}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) + \phi_S \mathbf{S}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) + \phi_{ST} \mathbf{ST}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) + \phi_{TS} \mathbf{TS}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) + \boldsymbol{\varepsilon} \quad (2.5)$$

En esta expresión se pueden identificar los diferentes roles que asumen sus términos de la siguiente forma: $\mathbf{Z}\boldsymbol{\alpha}$ y $\mathbf{X}\boldsymbol{\beta}$, corresponden a la parte hedónica del modelo. $\phi_T \mathbf{T}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})$, $\phi_S \mathbf{S}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})$, $\phi_{ST} \mathbf{ST}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})$ y $\phi_{TS} \mathbf{TS}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})$ recogen la influencia de las tendencias temporales, espaciales, espacio-temporales y temporo-espaciales del mercado, respectivamente, medidas a través de los errores y que no ha sido capturada por la parte hedónica del modelo.

Para lo que sigue en el desarrollo de esta Memoria, el modelo sobre el que se presentarán las diferentes metodologías, tratamientos y aplicaciones será el modelo que aparece en la expresión (2.5).

Dado que a una observación sólo le afectan las ocurridas anteriormente en el tiempo y con el fin de que las influencias espacio-temporales se ejerciten desde el pasado hasta el presente, las matrices \mathbf{S} y \mathbf{T} son triangulares inferiores (es decir, verifican que $j \geq i \leftrightarrow s_{ij} = 0$ y $t_{ij} = 0$). Dichas matrices son, además, estocásticas por filas, verificando que $\mathbf{S}\mathbf{1}_n = \mathbf{1}_n$ y $\mathbf{T}\mathbf{1}_n = \mathbf{1}_n$ ($\mathbf{1}_n$ denota

el vector de n unos) y $s_{ij} \geq 0$, $t_{ij} \geq 0$; $i, j=1, \dots, n$ lo cual las dota de un carácter de filtros lineales (Davidson y MacKinnon, 1993) que facilita la interpretación de los resultados obtenidos. Las matrices \mathbf{TS} y \mathbf{ST} son, a su vez, triangulares inferiores y permiten la modelización de efectos espacio-temporales compuestos.

La ponderación espacial se incorpora de forma explícita a través de la utilización de matrices \mathbf{S} de la forma:

$$\mathbf{S} = \frac{\sum_{\ell=1}^{m_S} \lambda^\ell \mathbf{S}_\ell}{\sum_{\ell=1}^{m_S} \lambda^\ell} \text{ con } 0 < \lambda \leq 1 \text{ y } m_S \in \mathbf{N} \quad (2.6)$$

donde $\mathbf{S}_\ell = (s_{ij}^\ell)$ verifica que $s_{ij}^\ell = 1$ si la j -ésima transacción de \mathbf{Y} es la ℓ -ésima más cercana espacialmente a la i -ésima transacción, siendo $s_{ij}^\ell = 0$ en otro caso. Por lo tanto, los elementos de \mathbf{S} sólo toman valores no nulos para las m_S transacciones previas a una dada, que están más cerca de ella espacialmente, construyendo de esta forma, filtros espaciales que asignan pesos proporcionales a $\{\lambda^\ell; \ell=1, \dots, m_S\}$ para capturar las tendencias espaciales de carácter local presentes en los datos.

Los efectos de dependencia temporal quedan modelados mediante matrices \mathbf{T} de la forma:

$$\mathbf{T} = \frac{\sum_{\ell=1}^{m_T} \gamma^\ell \mathbf{T}_\ell}{\sum_{\ell=1}^{m_T} \gamma^\ell} \text{ con } 0 < \gamma \leq 1 \text{ y } m_T \in \mathbf{N} \quad (2.7)$$

El modelo STAR

donde $\mathbf{T}_\ell = (t_{ij}^\ell)$ verifica que $t_{ij}^\ell = 1$ si la j -ésima transacción de \mathbf{Y} es la ℓ -ésima anterior en el tiempo a la i -ésima transacción, siendo $t_{ij}^\ell = 0$ en otro caso. Por lo tanto, los elementos de \mathbf{T} sólo toman valores no nulos para las m_T transacciones previas a una dada, construyendo de esta forma, filtros temporales que asignan pesos proporcionales a $\{\gamma^\ell; \ell=1, \dots, m_T\}$ para capturar las tendencias temporales de carácter local presentes en los datos.

En la literatura existen otras propuestas diferentes para modelar estas dependencias espaciales y temporales. En los trabajos de Pace y otros (1998b, 2000), la definición de la matriz \mathbf{S} es la que se presenta en (2.6) mientras que la matriz \mathbf{T} se define del siguiente modo:

$$t_{ij} = \frac{1}{m_T} \leftrightarrow (i-1) \geq j \geq (i-m_T) \quad (m_T \in \mathbf{N}) \quad (2.8)$$

Esta forma de la matriz \mathbf{T} da lugar a filtros lineales temporales de carácter local que asignan el mismo peso a las m_T transacciones previas a una dada (notar que es un caso particular de 2.7 con $\gamma = 1$).

Por su parte, Sun y otros (2005) utilizan como matriz \mathbf{T} la misma que Pace y otros (1998b, 2000) pero definen \mathbf{S} de la siguiente forma:

$$s_{ij} = \frac{\left(1 - \left(d_{ij} / D_{i,q+1}\right)^3\right)^3}{w_j} \quad j < i \quad \text{con } q \in \mathbf{N} \quad (2.9)$$

donde d_{ij} es la distancia euclídea entre las localizaciones de las viviendas i y j -ésima, calculada a partir de sus coordenadas UTM, $D_{i,q+1}$ es la $(q+1)$ -ésima distancia más pequeña entre la transacción i -ésima y las transacciones previas, indicando “ q ” el orden de retardos espaciales que deben ser considerados en la matriz espacial, para eliminar suficientemente la autocorrelación espacial, no presentando los autores, un procedimiento sistemático en la forma de elección

de su valor. El denominador w_j asigna el valor 1 si j es la transacción más cercana a i , 2 si es la segunda más cercana, 3 si es la tercera y 4 en otro caso.

Para la construcción de las matrices S y T , Pace y otros (1998b, 2000) fijan el valor de m_s sin justificación alguna y para m_T y λ la elección la hacen tras un leve análisis de sensibilidad. Además, como se ha comentado en (2.8) asignan la misma ponderación a todas las transacciones influyentes en el tiempo, lo cual dota de menor flexibilidad y realismo al modelo estimado

En nuestra opinión, la identificación de los valores máximos del número de transacciones que pueden presentar algún tipo de influencia espacial y/o temporal sobre una dada, debe ser llevada a cabo a través del estudio de las correlaciones espacio-temporales. En este capítulo, nuestra propuesta para la determinación de los valores de los parámetros m_T , γ , m_s y λ , es estimarlos utilizando un proceso de búsqueda que toma como criterio de selección el de la máxima-verosimilitud.

2.3 Estimación del modelo

El modelo (2.5) puede reescribirse del siguiente modo:

$$(\mathbf{I} - \phi_T \mathbf{T} - \phi_S \mathbf{S} - \phi_{ST} \mathbf{ST} - \phi_{TS} \mathbf{TS}) (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) - \mathbf{Z}\boldsymbol{\alpha} = \boldsymbol{\varepsilon} \quad (2.10)$$

que es un caso particular de (2.1) en el que la forma de la matriz W es la que viene dada en (2.2)

La estimación de los parámetros del modelo se realiza por el método de máxima-verosimilitud. Al igual que en los trabajos de Pace y otros (1998b, 2000), la función de verosimilitud toma como dadas de antemano las n_0 primeras transacciones, puesto que no disponen de un número apropiado de

vecinos previos ni en el espacio ni en el tiempo, debilitando así su influencia en el proceso de estimación.

Suponiendo, por el momento, que m_T , γ , m_S y λ son conocidos, sean $\boldsymbol{\theta} = (\boldsymbol{\alpha}', \boldsymbol{\beta}', \boldsymbol{\phi}')$ y τ los parámetros del modelo, donde $\tau = \sigma^{-2}$ es la precisión de la distribución de los errores del mismo.

La función de verosimilitud condicionada a las n_0 primeras observaciones viene dada por la siguiente expresión:

$$L(\boldsymbol{\theta}, \tau) \propto \tau^{\frac{n-n_0}{2}} |\mathbf{I} - \mathbf{W}| \exp \left\{ -\frac{\tau}{2} ((\mathbf{I} - \mathbf{W})(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) - \mathbf{Z}\boldsymbol{\alpha})' ((\mathbf{I} - \mathbf{W})(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) - \mathbf{Z}\boldsymbol{\alpha}) \right\} \quad (2.11)$$

puesto que el modelo (2.5) se puede expresar como:

$$(\mathbf{I} - \mathbf{W})(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) - \mathbf{Z}\boldsymbol{\alpha} = \boldsymbol{\varepsilon} \quad \boldsymbol{\varepsilon} \sim N_n(0, \sigma^2 \mathbf{I}_n) \quad (2.12)$$

A partir de (2.11) el logaritmo de la verosimilitud viene dado por:

$$\ell(\boldsymbol{\theta}, \tau) = \frac{n-n_0}{2} \log \tau + \log |\mathbf{I} - \mathbf{W}| - \frac{\tau}{2} \text{SSE}(\boldsymbol{\theta}) + \text{Cte} \quad (2.13)$$

siendo:

$$\text{SSE}(\boldsymbol{\theta}) = ((\mathbf{I} - \mathbf{W})(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) - \mathbf{Z}\boldsymbol{\alpha})' ((\mathbf{I} - \mathbf{W})(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) - \mathbf{Z}\boldsymbol{\alpha}) \quad (2.14)$$

la suma de los cuadrados de los residuos del modelo.

El cálculo del estimador máximo verosímil de los parámetros del

modelo, se obtiene resolviendo el sistema
$$\begin{cases} \frac{\partial \ell}{\partial \boldsymbol{\theta}} = 0 \\ \frac{\partial \ell}{\partial \tau} = 0 \end{cases} .$$

En particular, se tiene que:

$$\frac{\partial \ell}{\partial \tau} = \frac{n - n_0}{2} \frac{1}{\tau} - \frac{1}{2} \text{SSE}(\boldsymbol{\theta}) = 0 \Rightarrow \hat{\tau}(\boldsymbol{\theta}) = \frac{n - n_0}{\text{SSE}(\boldsymbol{\theta})} \quad (2.15)$$

Sustituyendo este estimador $\hat{\tau} = \hat{\tau}(\boldsymbol{\theta})$ en la expresión (2.13), se obtiene la llamada función de verosimilitud *profile* dada por:

$$\ell(\boldsymbol{\theta}, \hat{\tau}) = \log |\mathbf{I} - \mathbf{W}| - \frac{n - n_0}{2} \log \text{SSE}(\boldsymbol{\theta}) + \text{Cte} \quad (2.16)$$

El estimador máximo verosímil de $\boldsymbol{\theta}$ se obtiene maximizando la expresión (2.16). Como el determinante $|\mathbf{I} - \mathbf{W}|$ es igual a uno, al ser la matriz \mathbf{W} triangular inferior, se tiene que $\log |\mathbf{I} - \mathbf{W}| = 0$, y por tanto, maximizar (2.16) equivale a minimizar $\text{SSE}(\boldsymbol{\theta})$. De aquí, se sigue que el estimador máximo verosímil de $\boldsymbol{\theta}$ coincide con su estimador mínimo cuadrático, consiguiéndose una simplificación en el proceso de estimación.

Bajo condiciones estándar de regularidad (ver, por ejemplo, Gallant y White 1988, capítulos 5 y 6) el estimador máximo verosímil $\hat{\boldsymbol{\theta}}$ de $\boldsymbol{\theta}$ verificará que:

$$\hat{\boldsymbol{\theta}} \xrightarrow{d} N(\boldsymbol{\theta}, \boldsymbol{\Sigma}(\hat{\boldsymbol{\theta}})) \quad (2.17)$$

cuando $n \rightarrow \infty$, donde \xrightarrow{d} denota convergencia en distribución y siendo:

$$\boldsymbol{\Sigma}(\hat{\boldsymbol{\theta}}) = \left[\sum_{t=n_0+1}^n \frac{\partial \ell_t}{\partial \boldsymbol{\theta}} \frac{\partial \ell_t}{\partial \boldsymbol{\theta}'} \right]^{-1} \quad (2.18)$$

$$\text{donde } \ell_t = \frac{1}{2} (\log \hat{\tau} - \hat{\tau} \hat{\varepsilon}_t^2) \text{ siendo: } \hat{\varepsilon}_t = y_t - \mathbf{x}_t \hat{\boldsymbol{\beta}} - u_t \hat{\mathbf{W}}(\mathbf{Y} - \mathbf{X} \hat{\boldsymbol{\beta}}) - z_t \hat{\boldsymbol{\alpha}} \quad (2.19)$$

donde $\mathbf{u}_t = (u_{tj}; j = 1, \dots, n)'$ es el $n \times 1$ vector unitario i -ésimo con $u_{tj} = 1$ si $t = j$ e igual a 0 en otro caso.

Todo el proceso de estimación seguido hasta ahora y, en particular, el resultado (2.17), supone que los valores de m_T , γ , m_S y λ son fijos y conocidos de antemano. Sin embargo, habitualmente, serán desconocidos y tienen que ser estimados. Una posibilidad para resolver este problema es tomar como valores de los mismos, su estimación máximo verosímil, pero esta forma de actuar no tiene en cuenta la incertidumbre asociada a dicha estimación. El análisis bayesiano que se presenta en los capítulos 3, 4 y 5 y el proceso de selección de variables descrito en el capítulo 6, muestran dos formas alternativas de tratar dicha incertidumbre.

2.4 Validación del modelo

Una vez estimado el modelo, se procede a su validación intra y extramuestral. En la validación intramuestral se analiza su bondad de ajuste a los datos observados, tanto desde un punto de vista gráfico como numérico. En la validación extramuestral se analiza la capacidad predictiva del modelo, evitando el riesgo de sobreajuste (*overfitting*) que provoca la doble utilización de los datos implícita en el proceso de validación intramuestral.

2.4.1 Bondad de ajuste

El análisis de la bondad de ajuste del modelo se realiza mediante el estudio de sus residuos, analizando la existencia de patrones sistemáticos, no capturados por el modelo, tanto en media como en varianza y en autocorrelación.

Los residuos a analizar son los estandarizados que vienen dados por

$\hat{\mathbf{e}} = \hat{\tau}^{1/2} \hat{\boldsymbol{\varepsilon}}$ donde:

$$\hat{\boldsymbol{\varepsilon}} = \mathbf{Y} - \left(\mathbf{Z}\hat{\boldsymbol{\alpha}} + \mathbf{X}\hat{\boldsymbol{\beta}} + \hat{\phi}_T \mathbf{T}(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}) + \hat{\phi}_S \mathbf{S}(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}) + \hat{\phi}_{ST} \mathbf{ST}(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}) + \hat{\phi}_{TS} \mathbf{TS}(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}) \right) \quad (2.20)$$

son los residuos no tipificados del modelo y las matrices \mathbf{T} y \mathbf{S} se calculan utilizando los valores estimados de m_T , γ , m_S y λ .

El estudio del comportamiento de los errores tipificados se lleva a cabo, tanto desde un punto de vista gráfico como numérico. En el análisis gráfico se estudia su comportamiento estático mediante estimadores kernel de su función de densidad y gráficos de cuantiles (*Q-Q plots*) y su comportamiento dinámico utilizando correlogramas espacio-temporales de los residuos. En estos últimos, se analiza la existencia de dependencias espacio-temporales significativas, representando en un diagrama cartesiano las autocorrelaciones espacio-temporales dadas por:

$$r_k = \frac{\sum_{i=n_0+k+1}^n (\hat{\boldsymbol{\varepsilon}}_i - \bar{\boldsymbol{\varepsilon}}_k)(\hat{\boldsymbol{\varepsilon}}_{s(i,k)} - \bar{\boldsymbol{\varepsilon}}_{s(k)})}{\sqrt{\sum_{i=n_0+k+1}^n (\hat{\boldsymbol{\varepsilon}}_i - \bar{\boldsymbol{\varepsilon}}_k)^2} \sqrt{\sum_{i=n_0+k+1}^n (\hat{\boldsymbol{\varepsilon}}_{s(i,k)} - \bar{\boldsymbol{\varepsilon}}_{s(k)})^2}} \quad (2.21)$$

con:

$$\hat{\boldsymbol{\varepsilon}}_i = y_i - \mathbf{z}'_i \hat{\boldsymbol{\alpha}} - \mathbf{x}'_i \hat{\boldsymbol{\beta}} - \mathbf{u}'_i \left[\hat{\phi}_T \mathbf{T}(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}) + \hat{\phi}_S \mathbf{S}(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}) + \hat{\phi}_{ST} \mathbf{ST}(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}) + \hat{\phi}_{TS} \mathbf{TS}(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}) \right] \quad (2.22)$$

y siendo:

$$\bar{\boldsymbol{\varepsilon}}_k = \frac{\sum_{i=n_0+k+1}^n \hat{\boldsymbol{\varepsilon}}_i}{n - n_0 - k} \quad (2.23)$$

$$\bar{\hat{\epsilon}}_{s(k)} = \frac{\sum_{i=n_0+k+1}^n \hat{\epsilon}_{s(i,k)}}{n - n_0 - k} \quad (2.24)$$

y donde $s(i,k) \in \{1, \dots, i-1\}$ es tal que la transacción $s(i,k)$ -ésima es el k -ésimo vecino espacial más cercano a la transacción i -ésima. Esta correlación verifica que $r_k \approx N(0, n^{-1})$ si $n \rightarrow \infty$ y los residuos son ruido blanco.

Dicho estudio se complementa con un análisis numérico de la distribución de los errores a través de sus percentiles, media, mediana de los valores absolutos, coeficientes de asimetría y curtosis. Además, se calculan el coeficiente de determinación múltiple y los cubrimientos empíricos de los intervalos predictivos a un paso, dados por las siguientes expresiones:

- *El coeficiente de determinación múltiple*

$$R^2 = 1 - \frac{\sum_{i=n_0+1}^n \hat{\epsilon}_i^2}{\sum_{i=n_0+1}^n (y_i - \bar{y})^2} \quad (2.25)$$

Este criterio toma valores entre 0 y 1, y evalúa la variación de los datos observados explicada por el modelo, de forma que, cuanto mayor es su valor mejor es el grado de ajuste del modelo a los datos.

- *Cubrimientos empíricos al 95 y 99% de los intervalos predictivos a un paso*

$$\text{Cub}(0.95) = 100 \frac{\sum_{i=n_0+1}^n I(|\hat{\epsilon}_i| \leq 1.96)}{n - n_0} \quad (2.26)$$

$$\text{donde } I(|\hat{\epsilon}_i| \leq 1.96) = \begin{cases} 1 & \text{si } |\hat{\epsilon}_i| \leq 1.96 \\ 0 & \text{si } |\hat{\epsilon}_i| > 1.96 \end{cases}$$

$$\text{Cub}(0.99) = 100 \frac{\sum_{i=n_0+1}^n I(|\hat{\epsilon}_i| \leq 2.58)}{n-n_0} \quad (2.27)$$

$$\text{donde } I(|\hat{\epsilon}_i| \leq 2.58) = \begin{cases} 1 & \text{si } |\hat{\epsilon}_i| \leq 2.58 \\ 0 & \text{si } |\hat{\epsilon}_i| > 2.58 \end{cases} \text{ y } \hat{\epsilon}_i = \hat{\tau}^{1/2} \hat{\epsilon}_i$$

Con el fin de valorar un buen comportamiento del modelo según estos criterios, se realiza un contraste de hipótesis para analizar si el cubrimiento es significativamente distinto o no del valor teórico.

El estadístico de dicho contraste viene dado por la expresión:

$$z_p = \sqrt{n-n_0} \frac{\text{Cub}(p) - p}{\sqrt{p(1-p)}} \quad (2.28)$$

donde $p \in \{0.95, 0.99\}$, siendo su pvalor = $2P[Z > |z_p|]$ con $Z \sim N(0,1)$.

2.4.2 Validación extramuestral

En la validación intramuestral, la muestra de validación coincide con los datos observados, con lo cual se realiza un doble uso de los mismos, lo cual podría conducir a problemas de sobreajuste. Por dicha razón, llevamos a cabo también, una validación extramuestral, que nos permite evaluar la significación práctica de los resultados desde un punto de vista predictivo.

A tal fin, se han llevado a cabo dos tipos de validación extramuestral: una validación ordinaria y otra tipo *rolling*.

Validación del modelo

En ambos tipos de validación extramuestral, los datos observados se dividen en dos submuestras: una de estimación $\mathbf{Y}_{\text{est}} = (y_1, \dots, y_{n_{\text{est}}})'$ y otra de validación $\mathbf{Y}_{\text{val}} = (y_{n_{\text{est}}+1}, \dots, y_n)'$ donde n_{est} es el tamaño de la muestra de estimación que verifica $n_0 < n_{\text{est}} < n$.

En la validación extramuestral ordinaria, para llevar a cabo la estimación de los parámetros del modelo se utiliza la información proporcionada por la muestra de estimación $(\mathbf{Y}_{\text{est}}, \mathbf{Z}_{\text{est}}, \mathbf{X}_{\text{est}})$ donde $\mathbf{Z}_{\text{est}} = (\mathbf{z}_1, \dots, \mathbf{z}_{n_{\text{est}}})'$ y $\mathbf{X}_{\text{est}} = (\mathbf{x}_1, \dots, \mathbf{x}_{n_{\text{est}}})'$.

Por su parte, en la validación extramuestral tipo *rolling*, la predicción en cada paso del periodo de validación, utiliza toda la información disponible hasta el instante previo, realizando una reestimación completa del modelo. Esto significa que, para cada paso, el periodo de estimación se expande secuencialmente al añadirse la información proporcionada por las transacciones más recientes acaecidas en dicho periodo.

En ambos casos, y con el fin de valorar el comportamiento predictivo extramuestral del modelo, se realiza un estudio de los errores de predicción similar al estudio de residuos llevado a cabo desde el punto de vista intramuestral. Es necesario hacer notar que, así como en el caso intramuestral los residuos se han calculado utilizando las estimaciones de los parámetros con la muestra completa, en el caso de la validación extramuestral ordinaria, los errores de predicción se calculan con los parámetros estimados utilizando la información proporcionada por la muestra de estimación $(\mathbf{Y}_{\text{est}}, \mathbf{Z}_{\text{est}}, \mathbf{X}_{\text{est}})$. Por su parte, en la validación extramuestral tipo *rolling*, los errores que se analizan provienen de las estimaciones obtenidas secuencialmente. Es decir, el error de predicción $\hat{\varepsilon}_i$ se calcula utilizando la expresión (2.22) tomando como valores

de los parámetros θ , τ , m_T , γ , m_S y λ sus valores estimados utilizando la información disponible hasta el periodo previo.

2.5 Simplificación del modelo

Una vez llevado a cabo el estudio de bondad del ajuste y validación del modelo STAR, se estudia una posible simplificación del mismo. Este proceso da lugar a otros modelos, cuyo comportamiento intra y extramuestral se compara con el del modelo STAR.

La comparación entre los modelos se efectúa por medio del test de razón de verosimilitudes (cuyo estadístico de contraste lo denotaremos $-2LLF$), del coeficiente de determinación múltiple ajustado y de los criterios que se listan a continuación.

En lo que sigue, denotaremos de forma genérica con \mathbf{M} el modelo que se considera en cada situación.

- *El coeficiente de determinación múltiple ajustado:*

$$\bar{R}^2 = 1 - (1 - R^2) \frac{n - n_0 - 1}{n - n_0 - \dim(\theta)} \quad (2.29)$$

siendo $\dim(\theta_M)$ el número de parámetros de \mathbf{M} . Este criterio es una corrección del coeficiente R^2 dado en (2.25) que, a diferencia de éste, penaliza por la complejidad del modelo.

- *La raíz del error cuadrático medio*

$$\text{RMSE}(\mathbf{M}) = \sqrt{\frac{1}{n-h} \sum_{i=h+1}^n \hat{\varepsilon}_i^2} \quad (2.30)$$

Simplificación del modelo

- *La desviación absoluta media extramuestral*

$$\text{MAD}(\mathbf{M}) = \frac{1}{n-h} \sum_{i=h+1}^n |\hat{\varepsilon}_i| \quad (2.31)$$

siendo $\{\hat{\varepsilon}_i; i=h+1, \dots, n\}$ los residuos del modelo \mathbf{M} . En los dos últimos criterios, cuanto menor es su valor, mejor es el comportamiento del modelo. Los dos criterios se utilizan para hacer comparaciones tanto intra como extramuestrales. En el caso intramuestral $h = n_0$, mientras que $h = n_{\text{est}}$ en el extramuestral.

- *El criterio BIC*

Este criterio viene dado por la expresión:

$$\text{BIC}(\mathbf{M}) = -2 \ell(\hat{\boldsymbol{\theta}}_{\mathbf{M}}) + \dim(\boldsymbol{\theta}_{\mathbf{M}}) \log(n-n_0) \quad (2.32)$$

siendo $\ell(\cdot)$ el logaritmo de la función de verosimilitud del modelo que viene dado en la expresión (2.13) y $\hat{\boldsymbol{\theta}}_{\mathbf{M}}$ el estimador máximo verosímil del vector $\boldsymbol{\theta}_{\mathbf{M}}$ de parámetros del modelo \mathbf{M} .

El criterio BIC forma parte de la familia de los criterios de información y fue propuesto por Schwarz (1978). Es un criterio que combina el ajuste a los datos, utilizando la función de verosimilitud, y el grado de parsimoniosidad del modelo cuantificado por la expresión $\dim(\boldsymbol{\theta}_{\mathbf{M}}) \log(n-n_0)$. Bajo ciertas

condiciones de regularidad, $\exp\left[-\frac{\text{BIC}(\mathbf{M})}{2}\right]$ es una aproximación asintótica

de la función de densidad predictiva a priori bayesiana (ver, por ejemplo, Kass y Raftery 1995). Se elige como modelo preferido aquel con un valor menor del criterio BIC.

- *Logaritmo de la densidad predictiva extramuestral*

Este criterio evalúa el logaritmo de la densidad predictiva f de las observaciones acaecidas en el periodo de validación. Si la validación extramuestral es ordinaria viene dado por:

$$\begin{aligned} \text{LPRED}(\mathbf{M}) &= \log f(y_{n_{\text{est}}+1}, \dots, y_n | \mathbf{Z}, \mathbf{X}, \hat{\boldsymbol{\theta}}_{\mathbf{M}, n_{\text{est}}}) = \\ &= \sum_{i=n_{\text{est}}+1}^n \log f(y_i | \mathbf{Z}_i, \mathbf{X}_i, \hat{\boldsymbol{\theta}}_{\mathbf{M}, n_{\text{est}}}) = \sum_{i=n_{\text{est}}+1}^n \log \left(\frac{\hat{\tau}_{n_{\text{est}}}^{\frac{1}{2}}}{\sqrt{2\pi}} \exp \left[-\frac{\hat{\tau}_{n_{\text{est}}}}{2} \hat{\varepsilon}_i^2 \right] \right) \end{aligned} \quad (2.33)$$

donde $\mathbf{X}_i = (\mathbf{x}_1, \dots, \mathbf{x}_i)'$, $\mathbf{Z}_i = (\mathbf{z}_1, \dots, \mathbf{z}_i)'$, $\hat{\boldsymbol{\theta}}_{\mathbf{M}, n_{\text{est}}}$ es el estimador máximo-verosímil del vector de parámetros del modelo, $\boldsymbol{\theta}_{\mathbf{M}}$, utilizando la información disponible en el periodo de estimación y los errores $\{\hat{\varepsilon}_i; i=n_{\text{est}}+1, \dots, n\}$ se han calculado tomando $\boldsymbol{\theta}_{\mathbf{M}} = \hat{\boldsymbol{\theta}}_{\mathbf{M}, n_{\text{est}}}$.

Si la validación extramuestral es tipo *rolling*, el criterio viene dado por:

$$\text{LPRED}(\mathbf{M}) = \sum_{i=n_{\text{est}}+1}^n \log f(y_i | \mathbf{Z}_i, \mathbf{X}_i, \hat{\boldsymbol{\theta}}_{\mathbf{M}, i-1}) = \sum_{i=n_{\text{est}}+1}^n \log \left(\frac{\hat{\tau}_{i-1}^{\frac{1}{2}}}{\sqrt{2\pi}} \exp \left[-\frac{\hat{\tau}_{i-1}}{2} \hat{\varepsilon}_i^2 \right] \right) \quad (2.34)$$

donde $\hat{\boldsymbol{\theta}}_{\mathbf{M}, i-1}$ es el estimador máximo-verosímil del vector de parámetros del modelo, $\boldsymbol{\theta}_{\mathbf{M}}$, utilizando la información disponible hasta el periodo previo a la transacción i -ésima y el error $\hat{\varepsilon}_i$ se ha calculado tomando $\boldsymbol{\theta}_{\mathbf{M}} = \hat{\boldsymbol{\theta}}_{\mathbf{M}, i-1}$.

- *Pérdida predictiva cuadrática*

Este criterio está basado en el criterio propuesto por Gelfand y Ghosh (1998) y busca evaluar el comportamiento extramuestral del modelo en

réplicas de los datos observados de la muestra utilizando como función de pérdida el error cuadrático. Viene dado por:

$$\text{SPL}(\mathbf{M}) = \sum_{i=n_0+1}^n (y_i - \hat{y}_i)^2 + \sum_{i=n_0+1}^n \hat{\text{Var}}(\hat{y}_i) \text{ con } \hat{y}_i = E[y_i | \mathbf{Z}_i, \mathbf{X}_i, \hat{\boldsymbol{\theta}}_M, \mathbf{M}] = g(\mathbf{Z}_i, \mathbf{X}_i, \hat{\boldsymbol{\theta}}_M).$$

$$\hat{\text{Var}}(\hat{y}_i) = \left(\frac{\partial g(\mathbf{Z}_i, \mathbf{X}_i, \boldsymbol{\theta}_M)}{\partial \boldsymbol{\theta}} \right)' \Big|_{\boldsymbol{\theta}_M = \hat{\boldsymbol{\theta}}_M} \boldsymbol{\Sigma}(\hat{\boldsymbol{\theta}}_M) \left(\frac{\partial g(\mathbf{Z}_i, \mathbf{X}_i, \boldsymbol{\theta}_M)}{\partial \boldsymbol{\theta}} \right) \Big|_{\boldsymbol{\theta}_M = \hat{\boldsymbol{\theta}}_M} \quad (2.35)$$

siendo $\boldsymbol{\Sigma}(\hat{\boldsymbol{\theta}}_M)$ la matriz de varianzas y covarianzas de $\hat{\boldsymbol{\theta}}_M$. El primer término de la expresión evalúa el ajuste del modelo a los datos y el segundo penaliza por el grado de complejidad del mismo.

2.6 Aplicación al mercado de la vivienda

La metodología descrita en las secciones anteriores, así como las que se presentan en los siguientes capítulos de esta Memoria, se aplican al análisis de la evolución del mercado inmobiliario de la vivienda en un área específica de la ciudad de Zaragoza, utilizando para ello la base de datos que se presenta a continuación en el apartado 2.6.1.

2.6.1 Descripción de la base de datos

Las observaciones en esta base de datos corresponden a una muestra de 788 transacciones de vivienda llevadas a cabo entre Noviembre de 2002 y Diciembre de 2004 en la zona que compete al Registro de la Propiedad número 5 de Zaragoza (ver Figuras 2.1 y 2.2) y vienen dadas con la fecha específica en la cual fueron registradas. En cuanto a la zona en la que se encuentran las viviendas consideradas, se puede decir que la parte noroeste de

la misma es el centro económico (no histórico) de la ciudad con la presencia de los inmuebles de valor más elevado y que conforme nos desplazamos hacia el sur y hacia el este en el mapa de la ciudad, nos aproximamos a zonas periféricas, incluso en algún caso de extrarradio de la ciudad, con viviendas de muy baja calidad.

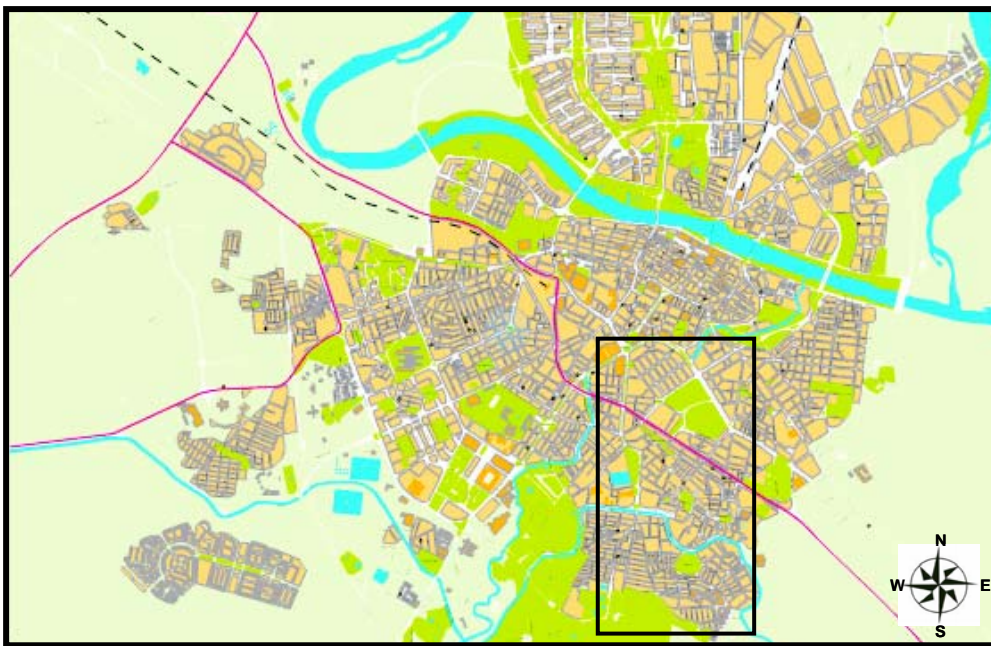


Figura 2.1: Mapa de la ciudad de Zaragoza. El área en estudio se encuentra dentro del rectángulo negro

La información utilizada procede de las siguientes entidades administrativas oficiales: Oficina Regional del Catastro, Instituto Aragonés de Estadística, Gerencia de Urbanismo del Ayuntamiento de Zaragoza y Colegio de Registradores de la Propiedad, Bienes Muebles y Mercantiles de España. La Oficina Regional del Catastro proporcionó la antigüedad del edificio y la superficie de los inmuebles. El Instituto Aragonés de Estadística cedió la información procedente del censo de viviendas del año 2001 de las características propias de cada inmueble (número de habitaciones, tipo de

calefacción y refrigeración), así como del edificio en el que se encontraba (ascensor, portero y garaje). La sección de Urbanismo del Ayuntamiento de la ciudad de Zaragoza facilitó las coordenadas UTM para la localización geográfica de los inmuebles. Finalmente, la información del Colegio de Registradores de la Propiedad, Bienes Muebles y Mercantiles de España recogía los movimientos en la titularidad de los inmuebles, así como el precio escriturado, el tipo de adquisición y si la vivienda era de protección oficial (VPO).



Figura 2.2: El área en estudio con las localizaciones específicas de las transacciones marcadas en rojo (Escala 1:25,000)

Los datos fueron depurados eliminando todos los registros que no presentaban información de las variables objetivas consideradas en el trabajo,

todos aquellos inmuebles situados bajo rasante y los dedicados a actividades no residenciales; así como las transacciones cuyos precios eran anormalmente altos o bajos de acuerdo con los límites para España publicados en la *Estadística Registral Inmobiliaria. Anuario de 2004*.

2.6.2 Análisis exploratorio de la base de datos

En las Tablas 2.1 y 2.2 se muestran las características de las viviendas analizadas. Se observa, en particular, que para un 28% de los casos (221 transacciones) no se dispone de información acerca de algunas de las características estructurales de las viviendas. Con el fin de analizar la existencia de posibles sesgos se incluye en todos los modelos estudiados una variable indicadora de dichos casos como variable independiente del modelo.

2.6.2.1 Características Estructurales

En la Tabla 2.1 se muestran las características de las variables estructurales cuantitativas de las viviendas analizadas.

Se observa la presencia de una elevada leptocurtosis, acompañada en el caso del precio y de la superficie útil, de un alto grado de asimetría positiva. Estos efectos se deben a la presencia de transacciones con valores muy altos en dichas variables. Con el fin de disminuir su influencia y aumentar el grado de normalidad de estas variables, se transforman logarítmicamente. Esta transformación tiene la ventaja adicional de que en un modelo de regresión hedónico los coeficientes se interpretan como elasticidades del precio de la vivienda con respecto a las variables independientes.

Tabla 2.1: Características estructurales cuantitativas de las viviendas analizadas

	Precio (€)	Superficie útil (m ²)	Antigüedad (Años)	Nº habitaciones*
Nº casos	788	788	788	567
Mínimo	3125.26	27.00	0.83	2
Máximo	601012.10	490.00	91.71	9
Media	121336.82	77.19	34.25	4.61
Desviación Típica	73540.28	32.57	13.02	1.02
Asimetría	2.24	4.88	-0.12	0.40
Curtosis	8.41	44.13	1.64	1.68

* Excluyendo cocina y baños

Tabla 2.2: Características estructurales cualitativas de las viviendas analizadas (en %)

VPO	Libre	Protegida			NS/NC	Total
	57.9	42.1				
Adquisición	Compraventa	Otros				100
	75.3	24.7				
Ascensor	Si	No				100
	56.3	43.7				
Portero	19.8	80.2				100
Garaje	16.8	83.2				100
Refrigeración	15.7	56.3			28.0	100
Calefacción	Colectiva	Individual	Sin calefacción		28.0	100
	23.2	26.8	22.0			
Número de habitaciones*	< 4	4	5	> 5	28.0	100
	7.1	25.5	29.7	9.6		

* Excluyendo cocina y baños

La variable número de habitaciones presenta una frecuencia muy baja para algunos de los valores, principalmente los extremos, por lo que para su introducción en los modelos se recodifica. Con el fin de que cada una de las clases contenga al menos un 5% de los datos observados, en lo que sigue dicha variable queda recodificada en 4 categorías, a saber, menos de 4 habitaciones, 4 habitaciones, 5 habitaciones y más de 5 habitaciones.

2.6.2.2 Características Dinámicas

En la Tabla 2.3 y en la Figura 2.3 se analiza la evolución trimestral del precio de la vivienda en el área considerada. Se observa, que los valores tomados por los precios medios son superiores a los de los precios medianos (en torno a un 15% más altos). Ello se debe al elevado grado de asimetría positiva en su distribución. Sin embargo, la evolución de ambos precios es muy similar (ver Figura 2.3), apreciándose la presencia de una tendencia de carácter creciente con incremento medio trimestral estimado en aproximadamente, un 3 % (alrededor de 12.55% anual).

Este crecimiento está en consonancia con la evolución estimada de los precios para Zaragoza capital, proporcionada por el Colegio de Registradores de la Propiedad. En nuestra opinión, viene dado por factores externos no recogidos por las variables hedónicas, y que puede ser expresado en términos de una tendencia lineal de carácter global, que complementaría los efectos temporales de índole más local que son capturados por los coeficientes autorregresivos espacio temporales del modelo. Por ello, se decide incluir en el modelo una variable que incorpore esta tendencia temporal lineal.

Tabla 2.3. Evolución Trimestral del precio de la vivienda

TRIMESTRE	Precio Medio (Cientos de miles de euros)	Precio Mediano (Cientos de miles de euros)
2002/T4	1.0815	0.9580
2003/T1	1.0900	0.9297
2003/T2	1.2022	1.0071
2003/T3	1.3009	1.0910
2003/T4	1.2011	1.0172
2004/T1	1.1221	0.9872
2004/T2	1.3229	1.1130
2004/T3	1.2654	1.2179
2004/T4	1.3923	1.2057
Tasa Acumulada media	3.21 %	2.92 %

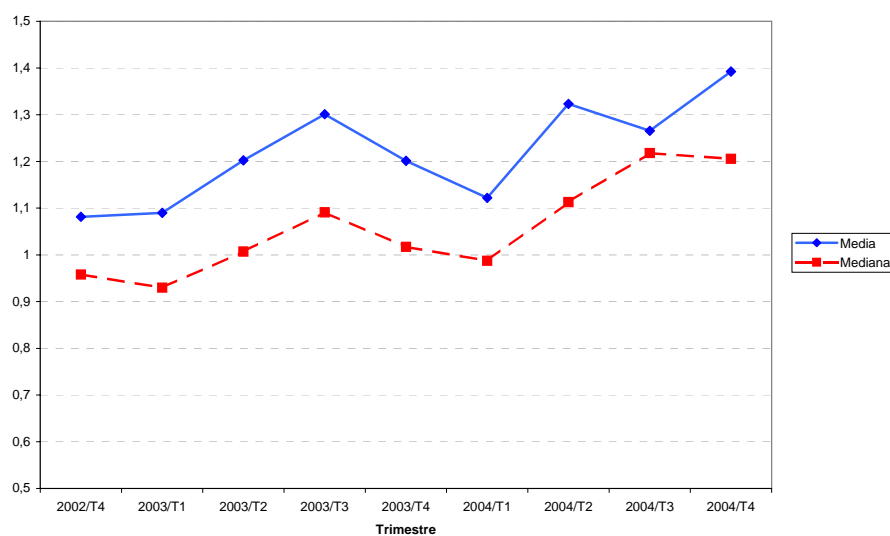


Figura 2.3: Evolución Trimestral de los precios medio y mediano (cientos de miles de €)

2.6.2.3 Características de localización

Una de las características que más afecta a los precios de los inmuebles es su ubicación. Actualmente, la misma puede ser determinada con gran precisión por medio de sistemas de información geográfica, lo cual ha propiciado que, de forma habitual, hayan sido incorporadas en los trabajos sobre precio de vivienda en los últimos años, las coordenadas de la situación de los inmuebles como la variable más apropiada para su localización (ver por ejemplo, Pace y otros (2000) o Sun y otros (2005)). En nuestro caso, todos los inmuebles de la base de datos disponen para localizar su emplazamiento de las coordenadas UTM (Sistema de coordenadas universal trasversal de Mercator Universal Transverse Mercator) que se utilizan como variables explicativas.

2.6.2.4 Dependencias Espaciales

En la Figura 2.4 se muestran las autocorrelaciones espaciales empíricas de los logaritmos de los precios de las transacciones analizadas (calculadas utilizando la expresión (2.21) en la que se han sustituido los residuos por los logaritmos de los precios) junto con las bandas de confianza del 95.5% $\left(\pm \frac{2}{\sqrt{n}}\right)$ suponiendo que la evolución de dichos logaritmos es ruido blanco. Se observa la existencia de autocorrelaciones positivas significativas, sobre todo en los primeros retardos, que reflejan la existencia de dependencias espaciales en la determinación del precio de cada transacción. Justificar la razón de dicha existencia es uno de los objetivos del modelo STAR.

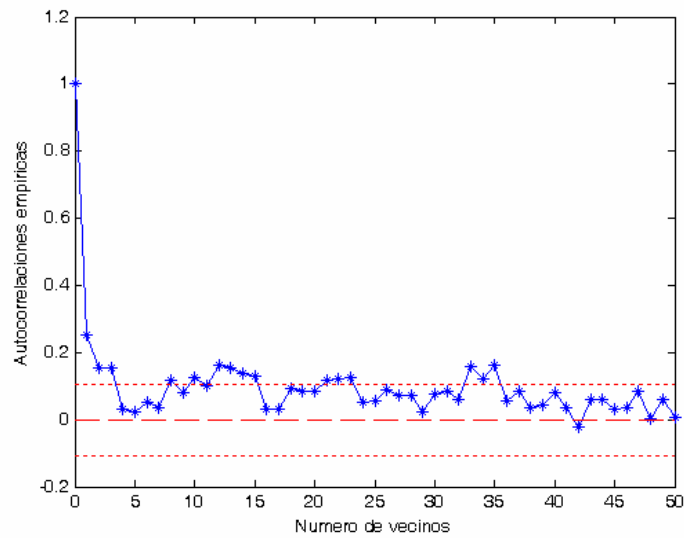


Figura 2.4: Autocorrelaciones espaciales del logaritmo del precio

2.6.2.5 Análisis de la multicolinealidad

Con el fin de prevenir problemas de multicolinealidad en las variables cuantitativas independientes que pudieran oscurecer la interpretación de los resultados obtenidos, se ha llevado a cabo un análisis de las correlaciones de dichas variables.

Como resultado de dicho estudio se observa únicamente una correlación significativamente negativa (-0.7564) entre las coordenadas UTM X y UTM Y de las viviendas analizadas, hecho que se debe fundamentalmente, a la situación geográfica de las viviendas de la muestra (ver Figura 2.5), siendo el resto de las correlaciones no significativas. Por esta razón se ha llevado a cabo un análisis de componentes principales de las coordenadas UTM.

Dichas componentes vienen dadas por las expresiones $PC1 = 0.3792*Z_x - 0.9177*Z_y$ y $PC2 = 0.9177*Z_x + 0.3792*Z_y$ donde Z_x y Z_y denotan las coordenadas UTM X e Y centradas.

La componente PC1 señala, aproximadamente, una dirección Noroeste-Sudeste y la componente PC2 una dirección Nordeste-Suroeste ortogonal a la anterior, en las localizaciones de las viviendas analizadas (ver Figura 2.5).

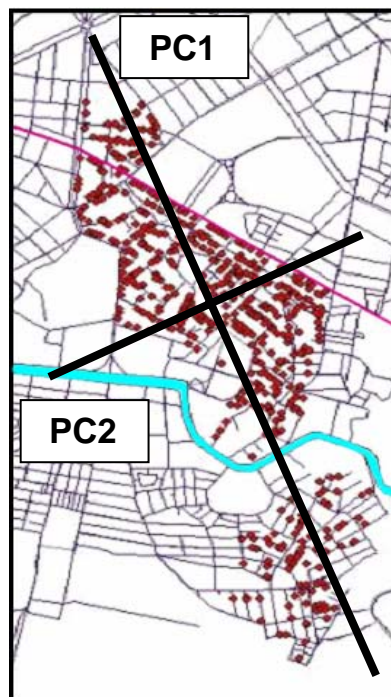


Figura 2.5: Situación de las viviendas transaccionadas y componentes principales de dichas coordenadas (marcadas en negro)

2.6.3 Variables incluidas en el modelo

Como consecuencia del análisis realizado, las variables que entran a formar parte del modelo son el logaritmo del precio del inmueble como

variable dependiente y como independientes el logaritmo de la superficie, el logaritmo de la antigüedad, VPO, tipo de adquisición, ascensor, garaje, portero, tipo de calefacción, refrigeración y número de habitaciones y un indicador de datos ausentes (*missing*). Además, y con el objetivo de capturar las tendencias espaciales y temporales de forma global, se incluyen como variables independientes las componentes PC1 y PC2, y un indicador de tendencia lineal temporal, respectivamente.

2.7 Resultados

Tal y como se ha comentado en la sección 2.4, la estimación de los parámetros del modelo se realiza condicionando a las n_0 primeras transacciones de la base de datos, y que tuvieron lugar en los meses de noviembre y diciembre de 2002 ($n_0 = 57$).

Para la construcción de las matrices de vecindad \mathbf{T} y \mathbf{S} es necesario determinar el número de transacciones potencialmente influyentes en el precio de una dada, tanto en espacio como en tiempo, así como las ponderaciones que se asignan a las mismas. Para ello, se proponen redes discretas $m_{Tred} = m_{Sred} = \{0, 5, \dots, 30, 35\}$ ($m_T = 35$ corresponde a un mes de transacciones aproximadamente) ya que en nuestro caso, el diagrama de autocorrelaciones espaciales de la Figura 2.4, muestra la existencia de dependencias significativas hasta un orden de 34 retardos. Para γ y λ , se proponen las siguientes redes $\gamma_{red} = \lambda_{red} = \{0.05, 0.10, \dots, 0.95, 1\}$.

Todos los resultados expuestos en el trabajo se han obtenido utilizando el programa MATLAB 6.5. Al no existir linealidad en los parámetros para el proceso de estimación se ha hecho uso de la subrutina *nlinfit* implementada en

MATLAB que resuelve el ajuste de mínimos cuadrados no lineales mediante el método de Gauss-Newton.

2.7.1 Estimación del modelo

Los valores encontrados para los parámetros de vecindad son $m_T = 10$, $m_S = 20$, $\gamma = 0.9$ y $\lambda = 1$. Las estimaciones del resto de los parámetros del modelo junto con sus errores estándar, se muestran en la Tabla 2.4 y han sido estimados con la matriz (2.18) calculada para los valores m_T , m_S , γ y λ mencionados anteriormente.

Se observa que las variables que ejercen una influencia significativa al nivel del 5% en el modelo STAR son: la componente PC1, el logaritmo de la superficie, el logaritmo de la antigüedad, el tipo de adquisición, así como la existencia de portero, garaje y ascensor en el edificio. Considerando un nivel de significación del 10% aparece también como influyente, la existencia de calefacción colectiva en la vivienda.

Los signos de los coeficientes hedónicos β de las variables anteriores son los esperados. Así, la componente PC1 presenta un efecto positivo significativo, reflejando el hecho de que, al seguir la dirección Noroeste de la componente se accede a la zona más céntrica de la ciudad donde se sitúan los inmuebles de una mayor calidad y precio más alto.

Las variables superficie de los inmuebles y antigüedad del edificio ejercen una influencia positiva para la primera variable, indicando que a mayor superficie mayor precio y negativa para la segunda, reflejando la penalización que tiene en el precio la edad del edificio. Se estima en el modelo STAR, en particular, que un incremento del 1% en la superficie útil de la vivienda o en la edad del edificio tiene un impacto positivo de alrededor de un

Resultados

0.70% y negativo del 0.23% en el incremento del precio de la vivienda, respectivamente.

El hecho de que el tipo de adquisición no sea una compraventa (herencias, donaciones, etc.) ejerce una influencia negativa sobre el precio debido a la necesidad de imputar un valor a la vivienda con el único fin del pago de impuestos. Más concretamente, el precio escriturado de la vivienda tiende a ser alrededor de un 45% menor.

La presencia de portero y de ascensor en el edificio tienen un efecto claramente positivo sobre el precio del inmueble, cifrándose en torno a un 14% la aportación de la primera y alrededor de un 11% la de la segunda.

La existencia de garaje en el edificio tiene una contribución negativa en la valoración de la vivienda estimándose en un 11.5% la disminución del precio de la misma por este hecho. Este resultado, que se puede antojar paradójico a primera vista, responde al tipo de vivienda que se encuentra en la zona en estudio. La zona más céntrica de la base de datos corresponde a viviendas de alta calidad que debido a su antigüedad no disponen de garaje en la casa, por no ser habitual su construcción en el periodo en que fueron edificadas, mientras que viviendas de menor valor y en zonas más alejadas con una menor antigüedad disponen de garaje, teniendo a pesar de ello una valoración inferior en el mercado.

Todas las variables mencionadas anteriormente han resultado significativas al nivel del 5%. Al aumentar el nivel hasta el 10% aparece como significativa la presencia de calefacción colectiva en el edificio, teniendo una valoración positiva, que se traduce en un aumento del precio en un 10.65%.

Tabla 2.4: Estimaciones de los parámetros (en negrita los significativos al 5% y marcados con ⁺ los significativos al 10%)

VARIABLES	STAR			STARs			hedónico		
	Coef	Desv	pvalor	Coef	Desv	pvalor	Coef	Desv	pvalor
Constante	14.1034	5.2271	0.0070	11.2628	3.1114	0.0003	8.7745	0.2342	0.0000
Tiempo	0.0467	0.0154	0.0025	0.0399	0.0119	0.0008	0.0369	0.0088	0.0000
PC1	0.0002	0.0000	0.0000	0.0002	0.0000	0.0000	0.0002	0.0000	0.0000
PC2	-0.0002	0.0001	0.1545	-0.0002	0.0001	0.0395	-0.0003	0.0001	0.0093
Log (superficie)	0.7031	0.0566	0.0000	0.7148	0.0521	0.0000	0.7196	0.0524	0.0000
Log (antigüedad)	-0.2301	0.0204	0.0000	-0.2302	0.0192	0.0000	-0.2273	0.0205	0.0000
VPO	-0.0119	0.0384	0.7558				-0.0125	0.0387	0.7465
Tipo de adquisición	0.4538	0.0402	0.0000	0.4564	0.0380	0.0000	0.4512	0.0401	0.0000
Refrigeración	-0.0042	0.0498	0.9323				-0.0009	0.0494	0.9849
Calefacción colectiva	0.1065 ⁺	0.0570	0.0617	0.0843 ⁺	0.0488	0.0843	0.0966 ⁺	0.0562	0.0857
Sin calefacción	0.0321	0.0524	0.5399				0.0131	0.0531	0.8053
Nº habitación es <4	0.0704	0.0782	0.3682				0.0927	0.0761	0.2231
Nº habitaciones =4	-0.0147	0.0508	0.7730				-0.0124	0.0508	0.8069
Nº habitaciones >5	0.0707	0.0692	0.3068				0.0787	0.0691	0.2550
Portero	0.1396	0.0575	0.0152	0.1380	0.0566	0.0148	0.1315	0.0542	0.0153
Garaje	-0.1142	0.0539	0.0341	-0.1134	0.0512	0.0269	-0.1261	0.0534	0.0182
Ascensor	0.1087	0.0402	0.0068	0.1035	0.0393	0.0084	0.1245	0.0400	0.0019
Indicador missing	0.0579	0.0550	0.2923				0.0516	0.0548	0.3462
σ	0.1863			0.1877			0.1895		
ϕ_T	0.1778 ⁺	0.1003	0.0763	0.1854 ⁺	0.0959	0.0533			
ϕ_S	0.3097 ⁺	0.1689	0.0667	0.2814 ⁺	0.1588	0.0763			
ϕ_{ST}	-0.3767	0.5632	0.5036						
ϕ_{TS}	-0.7123	0.3313	0.0316	-0.7460	0.3115	0.0166			

Finalmente, es importante destacar la no significación del indicador de datos ausentes, no apreciándose por tanto, la existencia de sesgos significativos imputables a este hecho.

La Tabla 4 muestra además, las estimaciones de los parámetros que recogen las dependencias espacio-temporales no explicadas por la parte hedónica del modelo. Se observa, que el coeficiente ϕ_{TS} resulta significativo al 5% y los coeficientes ϕ_T y ϕ_S son significativos al 10%. Estos dos últimos recogen con signo positivo la existencia de tendencias locales crecientes de carácter temporal y espacial, respectivamente. El coeficiente ϕ_{TS} recoge con signo negativo las interacciones espacio-temporales. Conviene hacer notar, sin embargo, que las estimaciones de estos coeficientes están basadas en resultados asintóticos, cuya validez solamente está probada para valores fijos de m_T , m_S , γ y λ .

2.7.2 Validación del modelo

En esta sección se analiza la validación del modelo STAR tanto desde un punto de vista intra como extramuestral. La predicción, como se ha comentado en la sección 2.5, se lleva a cabo a través de dos procedimientos: uno de validación extramuestral ordinaria y otro tipo *rolling*. Para la primera de ellas se toma como periodo de estimación el comprendido entre Noviembre de 2002 y Diciembre de 2003 ($n_{est} = 457$) y como periodo de validación el año 2004 completo ($n - n_{est} = 331$).

Para la validación extramuestral de tipo *rolling*, se inicia el proceso de estimación con el mismo número de transacciones que en el método ordinario, pero en cada paso se añade la última predicción al conjunto de elementos para la siguiente estimación.

Así, en la Tabla 2.5 y en las Figuras 2.6 a 2.21 se muestran los resultados tanto del estudio de la bondad de ajuste intramuestral, como del comportamiento predictivo extramuestral del modelo.

En Figuras 2.6, 2.12 y 2.18 se muestra la evolución dinámica de los errores tipificados para cada uno de los 3 procedimientos de validación anteriormente mencionados. Así mismo, en las Figuras 2.7, 2.13 y 2.19 se analizan las autocorrelaciones espacio-temporales de dichos errores calculadas mediante la expresión (2.21). Se aprecia que en ninguno de los casos existen tendencias sistemáticas en la evolución dinámica de dichos errores ni dependencias espacio-temporales significativas presentando, por tanto, un comportamiento adecuado tanto intramuestral como extramuestralmente.

El estudio de la distribución de los errores tipificados en las tres situaciones se lleva a cabo de forma pormenorizada en la Tabla 2.5, así como en las Figuras 2.8 y 2.9 (errores intramuestrales), 2.14 y 2.15 (errores extramuestrales ordinarios) y 2.20 y 2.21 (errores extramuestrales *rolling*) en las que se muestran estimadores tipo kernel de la función de densidad (Figuras 2.8, 2.14 y 2.20) y Q-Q plots (Figuras 2.9, 2.15 y 2.21). En todos los casos se observa la existencia de una asimetría negativa no despreciable debido a la presencia de unos pocos valores atípicos, así como una elevada leptocurtosis que pone de manifiesto la existencia de falta de normalidad de los errores. Este último hecho será tratado con más detalle en el capítulo 5 de la Memoria.

Finalmente, y con el fin de ilustrar gráficamente el comportamiento predictivo extramuestral del modelo, en las Figuras 2.10 y 2.16 se muestran las predicciones del logaritmo del precio correspondientes a la validación ordinaria y *rolling*, respectivamente, y en las Figuras 2.11 y 2.17 se muestran las predicciones correspondientes al precio.

En todos los casos se pone de manifiesto el correcto comportamiento predictivo del modelo que captura de forma adecuada la asimetría positiva presente en la distribución de los precios de las transacciones (ver Figuras 2.11 y 2.17).

Tabla 2.5: Distribución de los errores tipificados intramuestrales y extramuestrales del modelo STAR

ERRORES TIPIFICADOS DEL MODELO STAR			
	Intramuestrales	Extramuestrales (ord)	Extramuestrales (<i>rolling</i>)
Mínimo	-7.8373	-6.6780	-6.6183
1%	-3.0161	-3.6529	-3.7904
5%	-1.7430	-1.6681	-1.7198
10%	-1.0863	-0.9383	-1.012
25%	-0.4219	-0.3351	-0.4299
50%	0.1196	0.2939	0.2475
75%	0.5833	0.7281	0.6294
90%	0.9861	1.0866	0.9976
95%	1.3928	1.3376	1.2557
99%	2.0774	2.0163	1.8808
Máximo	2.4881	2.3090	2.3642
Media	0.0000	0.1217	0.0346
Mediana	0.5301	0.5856	0.5516
Asimetría	-1.7209	-2.0037	-1.9595
Curtosis	11.9675	12.0445	11.4036
R²	0.4562	0.4545	0.4348
CUB95	95.21	96.98 ⁺	97.89 ⁺
CUB99	98.50	98.79	98.79

⁺ Existen diferencias significativas con el resultado teórico

Si analizamos los cubrimientos al 99% (ver Tabla 2.5) podemos observar que en las tres situaciones lo hacen de forma correcta; sin embargo, en los cubrimientos al 95%, sólo en el caso intramuestral el cubrimiento no difiere significativamente del valor teórico. En el caso extramuestral se encuentran diferencias significativas, con pvalores de 0.0140 y 0.0003 para el caso de la validación ordinaria y para la de tipo *rolling*, respectivamente.

Es notable la concordancia que presenta el valor de R^2 tanto en el caso intramuestral como en los extramuestrales, cifrándose en un 45% la variación explicada por el modelo.

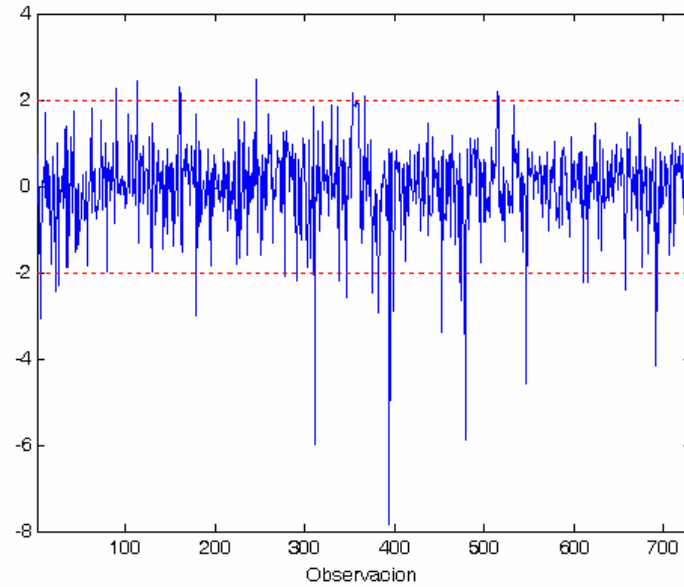


Figura 2.6: Errores tipificados intramuestrales del modelo STAR

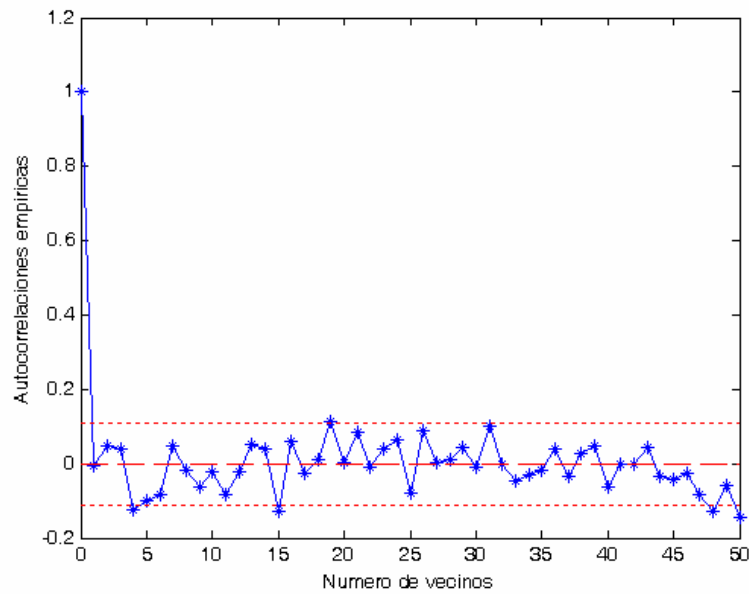


Figura 2.7: Autocorrelaciones espaciales residuales de modelo STAR (en líneas rayadas en rojo las bandas de confianza al 95%)

Resultados

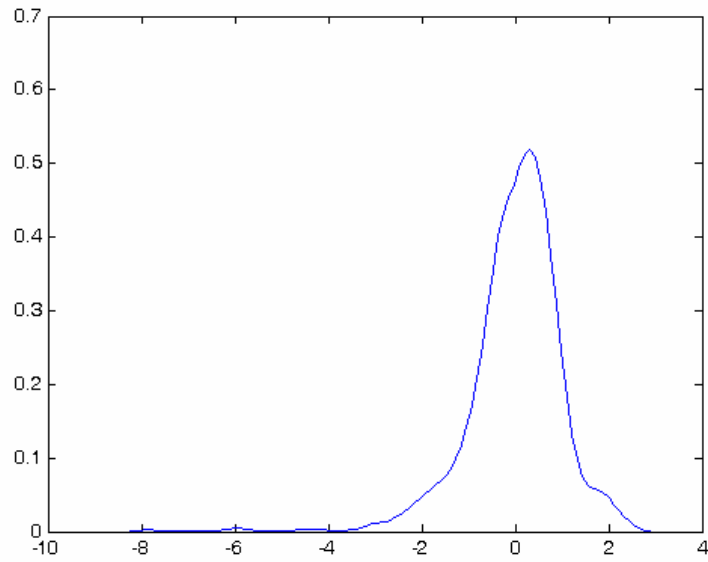


Figura 2.8: Densidad de los errores intramuestrales del modelo STAR

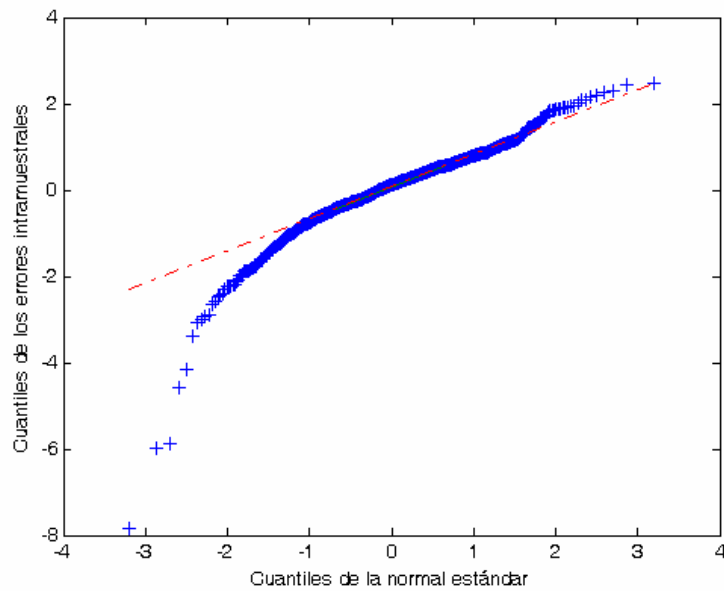


Figura 2.9: Q-Q plot de los errores intramuestrales del modelo STAR

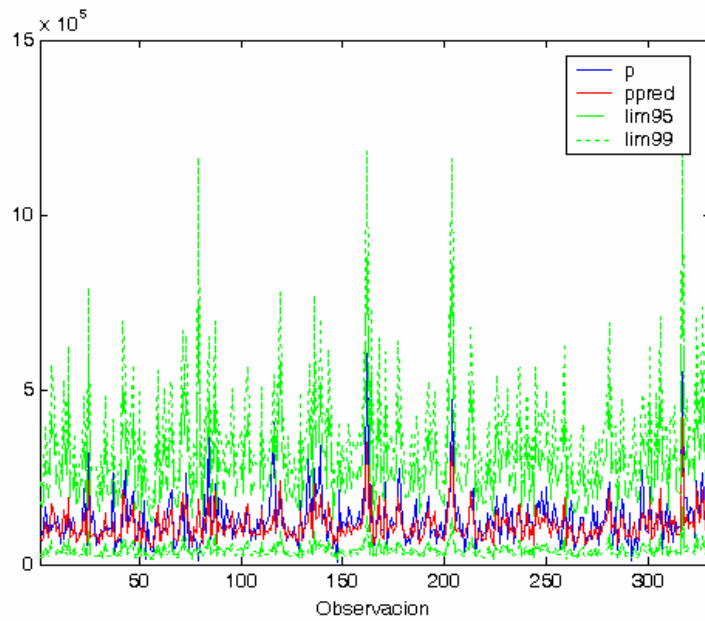


Figura 2.10: Predicciones extramuestrales del precio para el modelo STAR

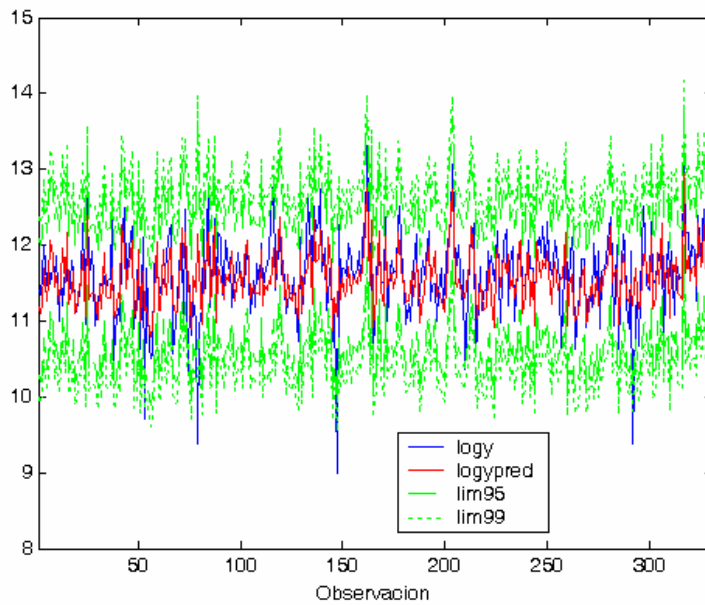


Figura 2.11: Predicciones extramuestrales del logaritmo del precio para el modelo STAR

Resultados

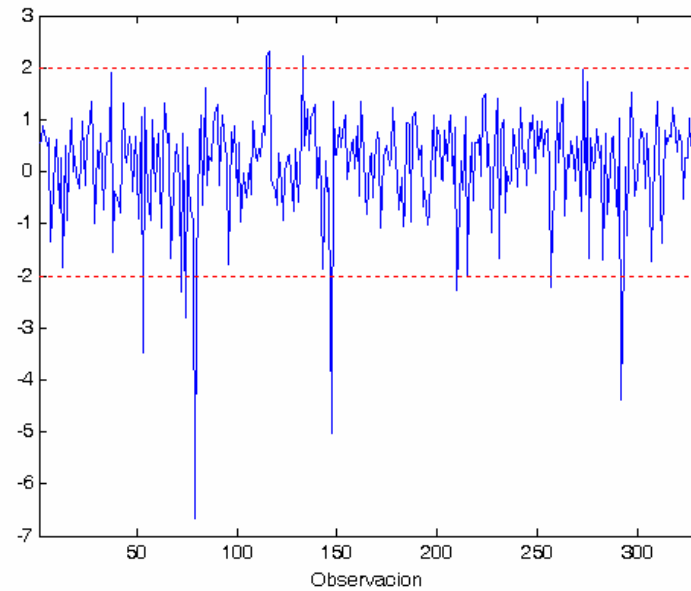


Figura 2.12: Errores tipificados extramuestrales del modelo STAR

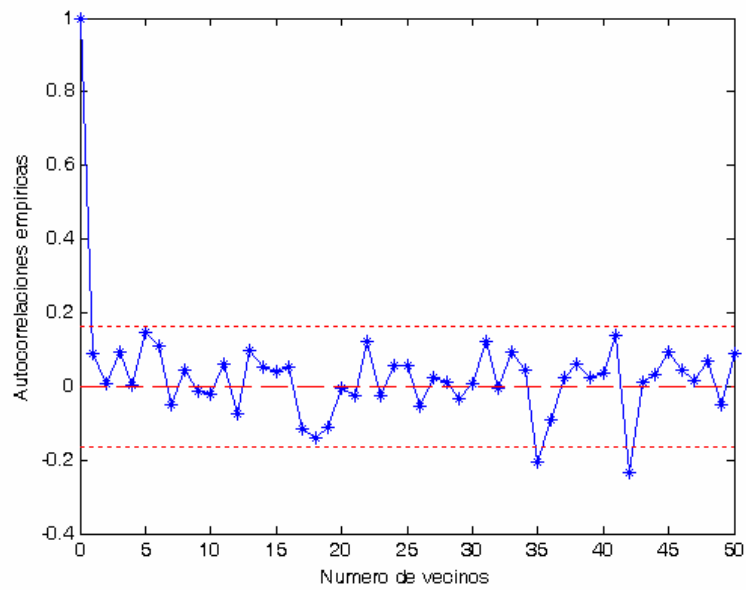


Figura 2.13: Autocorrelación de los errores extramuestrales del modelo STAR

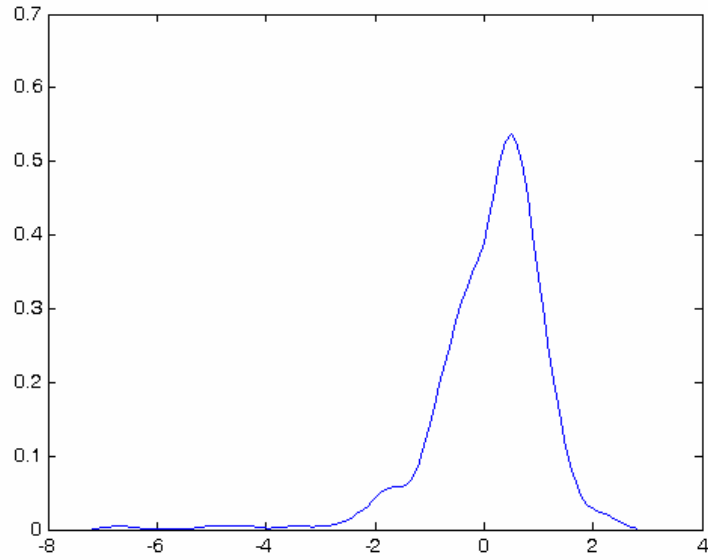


Figura 2.14: Densidad de los errores extramuestrales del modelo STAR

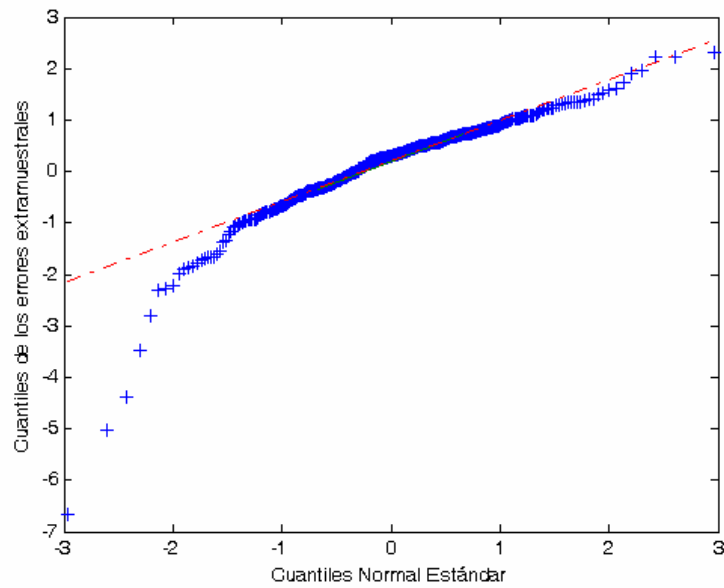


Figura 2.15: Q-Q plot de los errores extramuestrales del modelo STAR

Resultados

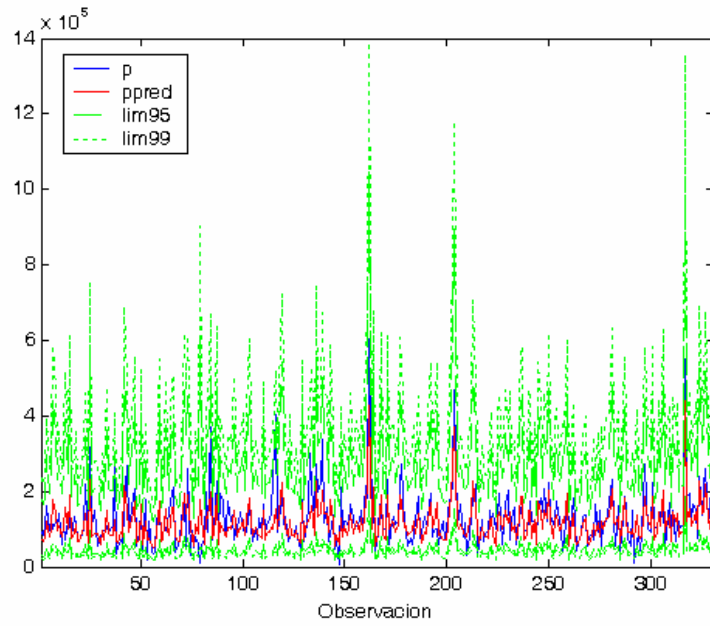


Figura 2.16: Predicciones extramuestrales (*rolling*) del precio para el modelo STAR

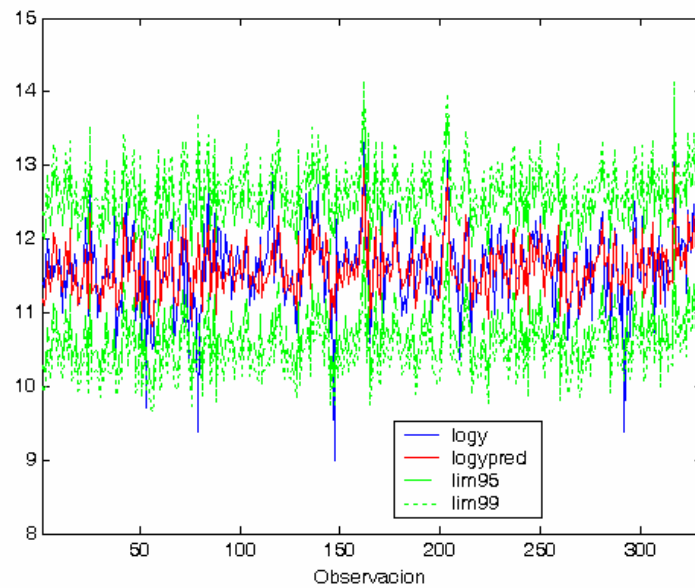


Figura 2.17: Predicciones extramuestrales (*rolling*) del logaritmo del precio para el modelo STAR

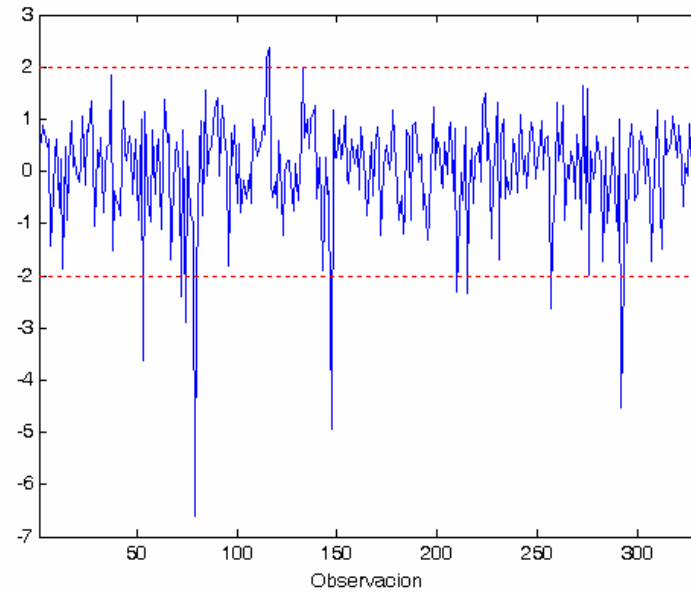


Figura 2.18: Errores tipificados extramuestrales (*rolling*) del modelo STAR

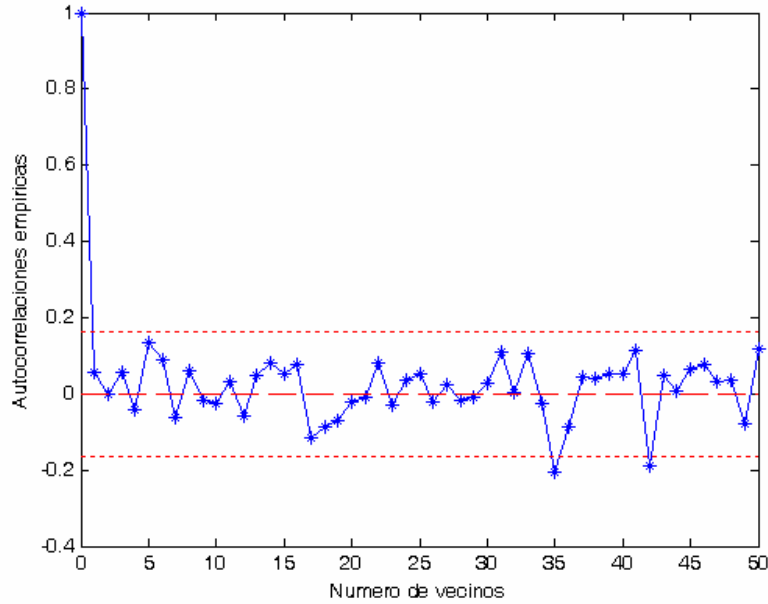


Figura 2.19: Autocorrelaciones de los errores extramuestrales (*rolling*) del modelo STAR

Resultados

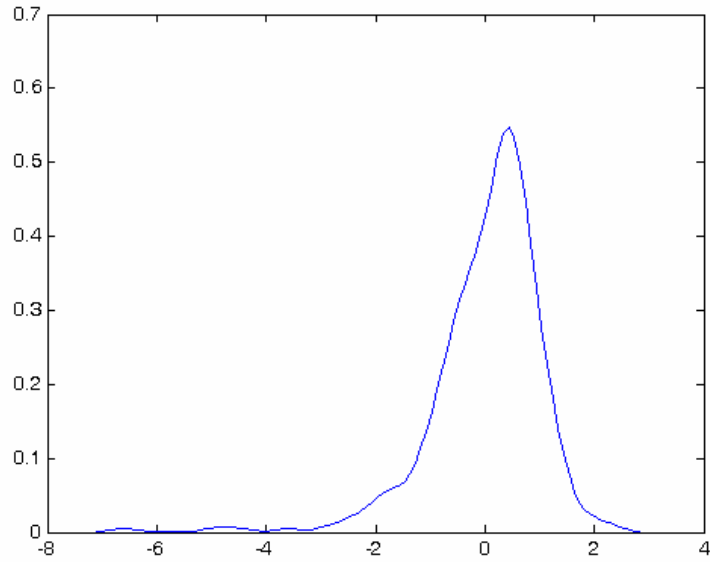


Figura 2.20: Densidad de los errores extramuestrales (*rolling*) del modelo STAR

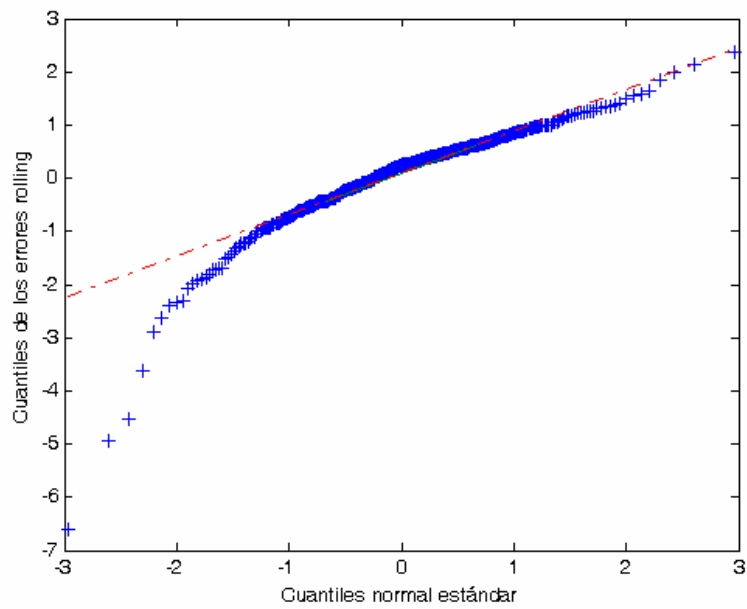


Figura 2.21: Q-Q de los errores extramuestrales (*rolling*) del modelo STAR

A la vista de los resultados expuestos, podemos concluir que, tanto desde un punto de vista intramuestral como extramuestral y predictivo, el modelo presenta un correcto grado de bondad de ajuste a los datos observados.

2.7.3 Simplificación del modelo

Con el fin de obtener una expresión más parsimoniosa de la ecuación del modelo en esta sección se realiza un proceso de simplificación del mismo aplicando un procedimiento de selección de variables tipo *backwards stepwise*, que utiliza como criterio de salida el contraste de razón de verosimilitudes (con un nivel de significación del 10%), eliminando en cada paso la variable que, tras reestimar los coeficientes, presenta el pvalor con menor significación, asociado al estadístico t. Las variables seleccionadas coinciden con las variables significativas del modelo STAR, a excepción de la variable PC2 (ver Tabla 2.4). A este modelo STAR simplificado lo denotaremos de aquí en adelante como STARs.

Así mismo, a efectos comparativos y con el fin de evaluar la significación estadística y práctica de los efectos espacio-temporales encontrados, se estima un modelo hedónico tradicional, que resulta directamente de la eliminación de la parte autorregresiva del modelo STAR, haciendo cero los coeficientes ϕ_T , ϕ_S , ϕ_{ST} y ϕ_{TS} .

Las estimaciones de los parámetros de los tres modelos comparados, a saber: el modelo STAR, el modelo STARs y el modelo hedónico se presentan en la Tabla 2.4. En general, presentan una gran concordancia en cuanto a los signos y las variables que aparecen como significativas, no hallándose ni siquiera grandes diferencias en cuanto al valor de los parámetros.

En las Tablas 2.6-2.8 se muestran los resultados obtenidos al realizar una comparación del comportamiento de los 3 modelos tanto desde un punto de vista intramuestral (Tabla 2.6) como extramuestral ordinario (Tabla 2.7) y *rolling* (Tabla 2.8) utilizando los criterios descritos en la sección 2.6.

Respecto al comportamiento intramuestral (Tabla 2.6) se observa, en primer lugar, que el modelo hedónico presenta diferencias estadísticamente significativas con respecto al modelo STAR a un nivel del 5%, puestas de manifiesto por el test de razón de verosimilitudes. No se aprecian, sin embargo, diferencias significativas entre los modelos STAR y STARs.

Si analizamos el valor de los criterios de comparación se observa que en ninguno de ellos el modelo hedónico tradicional supera a los modelos espacio-temporales, siendo el modelo STAR el que presenta mejores resultados en **RMSE** y **SPL**. El modelo STAR simplificado resulta, a su vez, ganador en los criterios \bar{R}^2 , **MAD** y **BIC**.

Tabla 2.6: Criterios de comparación intramuestrales para los modelos STAR, STARs y hedónico

CRITERIO	MODELO		
	STAR	STARs	hedónico
\bar{R}^2	0.4401	0.4424	0.4337
RMSE	0.4317	0.4332	0.4353
MAD	0.3034	0.3024	0.3054
-2LLF	846.28	851.49	858.58 ⁺
BIC	1017.8	970.22	1003.7
SPL	136.98	137.68	139.17

⁺Significativo al 5% con un pvalor = 0.0153

Respecto al comportamiento extramuestral, en la Tabla 2.7 podemos ver que el modelo STARs presenta un mejor comportamiento con respecto al **RMSE** y **MAD**; y el modelo STAR es superior en términos de **LPRED**. De nuevo el modelo hedónico presenta peores resultados en todos los criterios calculados. Conviene hacer notar, sin embargo, que las diferencias existentes entre ellos no son muy relevantes.

Tabla 2.7: Criterios de comparación extramuestrales para los modelos STAR, STARs y hedónico

CRITERIO	MODELO		
	STAR	STARs	hedónico
RMSE	0.4147	0.4088	0.4157
MAD	0.3030	0.2924	0.3057
LPRED	-163.71	-165.92	-166.67

En cuanto a los resultados obtenidos tras la validación extramuestral tipo *rolling* (ver Tabla 2.8), son muy similares a los observados tras la validación extramuestral ordinaria. De nuevo el modelo STARs obtiene mejores puntuaciones en **RMSE** y **MAD** y el modelo STAR en **LPRED** sin que se aprecie la existencia de diferencias muy relevantes a favor de ninguno de los tres modelos.

Podemos concluir, por tanto, que aunque existen diferencias estadísticamente significativas a favor de los modelos espacio-temporales en cuanto al grado de ajuste a los datos, éstas no son muy relevantes desde un punto de vista práctico. Además, no se observan diferencias significativas en el comportamiento de los modelos STAR y STARs confirmando que el proceso de simplificación de modelo llevado a cabo es adecuado.

Tabla 2.8: Criterios de comparación extramuestrales (*rolling*) para los modelos STAR, STARs y hedónico

CRITERIO	MODELO		
	STAR	STARs	hedónico
RMSE	0.4074	0.4030	0.4070
MAD	0.2916	0.2832	0.2935
LPRED	-162.19	-163.75	-165.02

2.8 Conclusiones

En este capítulo se ha presentado de forma pormenorizada el modelo espacio temporal autorregresivo STAR con efectos de vecindad, inicialmente introducido por Pace y otros (1998b, 2000), que contiene una parte correspondiente a un modelo hedónico clásico, e incorpora a través de matrices de vecindad independientes para espacio y tiempo, la influencia de las transacciones próximas. Este modelo involucra una gran cantidad de parámetros y nuestra propuesta en esta Memoria incluye una reparametrización del mismo con el fin de disminuir el número de parámetros y facilitar el proceso de estimación y su interpretabilidad. Además, se presentan innovaciones en las matrices de proximidad para la elección de los vecinos espaciales y temporales, así como en las ponderaciones asignadas a cada una de las transacciones que cuantifican su influencia sobre una transacción dada.

El tratamiento utilizado para la estimación de los parámetros ha sido la máxima verosimilitud que, en este caso en concreto, coincide con la estimación mínimo cuadrática. Pero este proceso, a pesar de su rapidez computacional, ha supuesto que los parámetros m_T , m_S , γ y λ ,

correspondientes al número de vecinos temporales, espaciales y sus ponderaciones, se consideraran fijos y conocidos de antemano para el cálculo de los errores estándar de los coeficientes hedónicos y autorregresivos del modelo. Habitualmente no lo son, y por tanto deberán ser estimados. Este tratamiento no incorpora la incertidumbre asociada a su estimación, lo cual puede aumentar la significación de los mismos. La inferencia se basa además en resultados asintóticos cuya validez no ha sido probada, a nuestro nivel de conocimiento, en este contexto.

Así mismo, se ha presentado y analizado la base de datos que sirve para ilustrar las diferentes metodologías que se proponen en esta Memoria, y comprende un conjunto de transacciones de viviendas llevadas a cabo en un área concreta de Zaragoza, durante el periodo que va desde Noviembre de 2002 hasta Diciembre de 2004.

Una vez aplicado el procedimiento introducido en la primera parte del capítulo a la base de datos, los resultados del proceso de estimación muestran que para los coeficientes correspondientes a la parte hedónica del modelo, los signos son los esperados de antemano, resultando significativas al 5% las variables indicador de tiempo, proximidad al centro de la ciudad, antigüedad, superficie, tipo de adquisición, ascensor, portero y garaje, cuantificándose la influencia de las características de los inmuebles en su precio. La estimación de los parámetros autorregresivos ha señalado como significativa, únicamente, una interacción espacio-temporal de difícil interpretación. Al aumentar el nivel de significación al 10% surgen como significativas las variables calefacción colectiva y los coeficientes autorregresivos de espacio y tiempo sin interacción.

El análisis de la bondad de ajuste del modelo ha dejado patente la falta de normalidad de los errores debida a una fuerte asimetría negativa y

leptocurtosis. No obstante, la validación del modelo, llevada a cabo extramuestral (ordinaria y *rolling*) e intramuestralmente, permite calificar de apropiada la capacidad predictiva del modelo.

El proceso de simplificación y comparación del modelo con dos versiones simplificadas del mismo, el modelo hedónico tradicional y un modelo autorregresivo más sencillo, ha dado su aprobación a los modelos espacio temporales en todos los casos y dentro de estos últimos ha premiado la parsimonia del modelo más simple en la mayor parte de los criterios utilizados.

Tras el estudio llevado a cabo en este capítulo, hay que hacer notar que los inconvenientes surgidos han venido dados por el tipo de aproximación utilizada. En capítulos posteriores, se tratará la estimación del modelo desde una óptica bayesiana, con el fin de resolver los problemas que aparecen al hacer uso de resultados asintóticos y al no incorporar la incertidumbre al proceso de estimación de los parámetros, en general, y de los referentes a los vecinos espaciales y temporales con sus ponderaciones, en particular.

Capítulo 3

Inferencia bayesiana en modelos STAR con efectos de vecindad

3.1 Introducción

En el capítulo anterior se ha presentado con detalle el modelo STAR aplicado a la modelización del precio de la vivienda en un área de la ciudad de Zaragoza, y el procedimiento utilizado para su estimación ha sido el método de la máxima verosimilitud. Este método tiene el inconveniente de que la inferencia realizada se basa en resultados asintóticos y supone, además, conocidos los valores de algunos de los parámetros del modelo (más concretamente los relacionados con los efectos de vecindad) no teniendo en cuenta la incertidumbre asociada a su estimación.

En este capítulo hacemos uso de una aproximación bayesiana que permite eludir la aplicación de resultados asintóticos, haciendo posible así mismo, la incorporación de la incertidumbre al proceso de estimación de todos los parámetros del modelo.

El capítulo está estructurado como sigue: en la sección 3.2 se presenta el tratamiento bayesiano a través de la distribución a priori y la distribución a

posteriori de los parámetros del modelo. En la sección 3.3 se muestra el proceso de estimación del modelo basado en métodos MCMC (*Markov Chain Monte Carlo*). Los apartados 3.4 y 3.5 describen los procesos de validación y simplificación del modelo tanto desde un punto de vista intra como extramuestral, presentando, en particular, un algoritmo secuencial de estimación y validación tipo *rolling* utilizando muestreo de importancia. En la sección 3.6 se aplica la metodología desarrollada a lo largo del capítulo al análisis de los datos presentados en el capítulo 2, sección 2.6.1 y finalmente, la sección 3.7 concluye exponiendo los resultados más relevantes. Los fundamentos matemáticos de algunos de los procesos de computación bayesiana descritos en este capítulo, se encuentran descritos en el apéndice A.

3.2 Planteamiento del problema

El planteamiento bayesiano se basa en considerar los parámetros del modelo como variables aleatorias cuyas distribuciones de probabilidad cuantifican el grado de incertidumbre asociado al proceso de estimación de los mismos. Dicho enfoque permite realizar un tratamiento unificado de la incertidumbre utilizando las herramientas del Cálculo de Probabilidades. En particular, posibilita llevar a cabo inferencias acerca de los parámetros del modelo, condicionadas a la información proporcionada por los datos observados. A nuestro modo de ver, esta es una forma más natural de efectuar el proceso de estimación que la propuesta por la aproximación clásica, cuyos niveles de confianza son de carácter premuestral, pudiendo provocar efectos indeseados, tal y como indican por ejemplo, Bernardo y Smith (1994).

La Estadística Bayesiana utiliza el teorema de Bayes para calcular la distribución a posteriori de los parámetros del modelo, a partir de la cual se

obtienen sus estimaciones. Esta aproximación permite, además, incorporar de forma natural restricciones e información a priori acerca de dichos parámetros, así como utilizar procesos de cálculo basados en simulaciones (ver apéndice A) que eluden la utilización de resultados asintóticos que pueden ser de dudosa validez o difícil aplicación en procesos espacio-temporales.

En esta sección aplicamos la maquinaria bayesiana al análisis del modelo STAR con efectos de vecindad descrito en la sección 2.3 y cuya ecuación viene dada por la expresión

$$\mathbf{Y} = \mathbf{Z}\boldsymbol{\alpha} + \mathbf{X}\boldsymbol{\beta} + \phi_T \mathbf{T}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) + \phi_S \mathbf{S}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) + \phi_{ST} \mathbf{ST}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) + \phi_{TS} \mathbf{TS}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) + \boldsymbol{\varepsilon} \quad (3.1)$$

con $\boldsymbol{\varepsilon} \sim N_n(\mathbf{0}, \sigma^2 \mathbf{I}_n)$.

Para ello comenzamos especificando la distribución a priori sobre sus parámetros.

3.2.1 Distribución a priori

La distribución a priori viene dada por:

$$\boldsymbol{\alpha} \sim N_q(\mathbf{0}, \boldsymbol{\Sigma}_\alpha) \quad (3.2)$$

$$\boldsymbol{\beta} \sim N_k(\mathbf{0}, \boldsymbol{\Sigma}_\beta) \quad (3.3)$$

$$\phi_i \sim U(-1, 1) \quad i \in \{T, S, ST, TS\} \quad (3.4)$$

$$\tau = \frac{1}{\sigma^2} \sim \text{Gamma}\left(\frac{d_0}{2}, \frac{d_0 s_0}{2}\right) \quad (3.5)$$

$$\mathbf{m}_T \sim U_D(\mathbf{m}_{Tred}) \quad (3.6)$$

$$\boldsymbol{\gamma} \sim U_D(\boldsymbol{\gamma}_{red}) \quad (3.7)$$

$$\mathbf{m}_S \sim U_D(\mathbf{m}_{Sred}) \quad (3.8)$$

$$\boldsymbol{\lambda} \sim U_D(\boldsymbol{\lambda}_{red}) \quad (3.9)$$

siendo (3.2)-(3.9) distribuciones independientes donde Σ_α ($q \times q$) y Σ_β ($k \times k$) son matrices definidas positivas, $U(a,b)$ denota la distribución uniforme en el intervalo (a, b) con $a < b \in \mathbf{R}$, $d_0, s_0 > 0$, $U_D(\mathbf{v})$ la distribución uniforme discreta en el conjunto $\mathbf{v} = \{v_1, \dots, v_q\}$ y

$$\mathbf{m}_{Tred} = \{m_{Tred,1}, \dots, m_{Tred,t}\} \text{ con } 0 \leq m_{Tred,1} < \dots < m_{Tred,t} < \infty \text{ y } m_{Tred,i} \in \mathbf{N} \cup \{0\}$$

$$\boldsymbol{\gamma}_{red} = \{\gamma_{red,1}, \gamma_{red,2}, \dots, \gamma_{red,g}\} \text{ con } 0 < \gamma_{red,1} < \gamma_{red,2} < \dots < \gamma_{red,g} \leq 1 \text{ y } \gamma_{red,i} \in \mathbf{R}$$

$$\mathbf{m}_{Sred} = \{m_{Sred,1}, \dots, m_{Sred,s}\} \text{ con } 0 \leq m_{Sred,1} < \dots < m_{Sred,s} < \infty \text{ y } m_{Sred,i} \in \mathbf{N} \cup \{0\}$$

$$\boldsymbol{\lambda}_{red} = \{\lambda_{red,1}, \lambda_{red,2}, \dots, \lambda_{red,\ell}\} \text{ con } 0 < \lambda_{red,1} < \lambda_{red,2} < \dots < \lambda_{red,\ell} \leq 1 \text{ y } \lambda_{red,i} \in \mathbf{R}$$

son redes discretas de valores factibles de los parámetros \mathbf{m}_T , $\boldsymbol{\gamma}$, \mathbf{m}_S , y $\boldsymbol{\lambda}$.

Las distribuciones (3.2)-(3.5) son estándar en la literatura bayesiana y son no informativas si $\Sigma_\alpha = s_\alpha^2 \mathbf{I}_q$ y $\Sigma_\beta = s_\beta^2 \mathbf{I}_k$ con $s_\alpha^2, s_\beta^2 \rightarrow \infty$ y $d_0 \rightarrow 0$. Las distribuciones (3.6)-(3.9) son discretas y han sido elegidas para facilitar la implementación del *Gibbs sampling* descrito en el apartado 3.3.2 y son no informativas si las redes \mathbf{m}_{Tred} , $\boldsymbol{\gamma}_{red}$, \mathbf{m}_{Sred} y $\boldsymbol{\lambda}_{red}$ son suficientemente amplias. Además, permiten realizar, un tratamiento más sistemático y realista en la estimación de estos parámetros que el descrito en el capítulo 2, donde se estiman sus valores por máxima-verosimilitud y se condiciona sobre ellos, no teniendo en cuenta la incertidumbre asociada al proceso. Queremos hacer notar, en referencia a este extremo, que a nuestro nivel de conocimiento, no se conoce ni si los resultados asintóticos utilizados en el proceso inferencial siguen siendo válidos cuando los parámetros \mathbf{m}_T , $\boldsymbol{\gamma}$, \mathbf{m}_S y $\boldsymbol{\lambda}$ son desconocidos, ni bajo qué condiciones se aplican con seguridad. Estos problemas no ocurren en Inferencia Bayesiana dado que, como veremos a continuación, dichas inferencias se basan en la distribución a posteriori de los parámetros del modelo.

3.2.2 Distribución a posteriori

Una vez especificadas la ecuación del modelo y la distribución a priori sobre los parámetros, las inferencias sobre estos se realizan a partir de su distribución a posteriori, que se calcula mediante el teorema de Bayes.

Con el fin de exponer más claramente los resultados obtenidos, de aquí en adelante utilizaremos la siguiente notación: dado un vector $\mathbf{b} = (b_1, \dots, b_n)'$ denotaremos por $\mathbf{b}_0 = (b_1, \dots, b_{n_0})'$ y por $\mathbf{b}_1 = (b_{n_0+1}, \dots, b_n)'$. Además, dada una matriz $n \times k$ $\mathbf{A} = (a_{ij})$ denotaremos:

$$\mathbf{A}_0 = \begin{pmatrix} a_{1,1} & \dots & a_{1,k} \\ \dots & \dots & \dots \\ a_{n_0,1} & \dots & a_{n_0,k} \end{pmatrix} \text{ y } \mathbf{A}_1 = \begin{pmatrix} a_{n_0+1,1} & \dots & a_{n_0+1,k} \\ \dots & \dots & \dots \\ a_{n,1} & \dots & a_{n,k} \end{pmatrix}$$

donde n_0 denota el número de observaciones iniciales sobre las que condicionamos para llevar a cabo la construcción de la función de verosimilitud del modelo.

Así mismo, dadas dos variables aleatorias X e Y , denotaremos las densidades de las distribuciones marginales y condicionadas de X e $Y|X$ utilizando $[X]$ y $[Y|X]$, respectivamente.

Sea $\boldsymbol{\theta} = (\boldsymbol{\alpha}', \boldsymbol{\beta}', \boldsymbol{\phi}', \tau, m_T, \gamma, m_S, \lambda)'$ el vector de parámetros del modelo con $\boldsymbol{\phi} = (\phi_T, \phi_S, \phi_{ST}, \phi_{TS})'$.

Utilizando la misma notación que en la sección 2.3 y teniendo en cuenta que la función de verosimilitud del modelo (3.1) viene dada por la expresión (2.12), aplicando el teorema de Bayes se sigue que la distribución a posteriori de los parámetros viene dada por:

$$\begin{aligned}
 & [\boldsymbol{\theta} \mid \mathbf{Y}, \mathbf{Z}, \mathbf{X}] \propto \\
 & \propto [\mathbf{Y}_1 \mid \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\phi}, \tau, m_T, \gamma, m_S, \lambda, \mathbf{Y}_0, \mathbf{Z}, \mathbf{X}] [\boldsymbol{\alpha}] [\boldsymbol{\beta}] [\boldsymbol{\phi}] [\tau] [m_T] [\gamma] [m_S] [\lambda] \propto \\
 & \propto \tau^{\frac{(n-n_0)}{2}} \exp\left[-\frac{\tau}{2} \boldsymbol{\varepsilon}'_1(\boldsymbol{\theta}) \boldsymbol{\varepsilon}_1(\boldsymbol{\theta})\right] \\
 & \times \exp\left[-\frac{1}{2} \boldsymbol{\alpha}' \boldsymbol{\Sigma}_\alpha^{-1} \boldsymbol{\alpha}\right] \exp\left[-\frac{1}{2} \boldsymbol{\beta}' \boldsymbol{\Sigma}_\beta^{-1} \boldsymbol{\beta}\right] \prod_{i \in \{T, S, ST, TS\}} I_{(-1,1)}(\phi_i) \times \\
 & \times \tau^{\frac{d_0-1}{2}} \exp\left[-\frac{d_0 s_0}{2} \tau\right] I_{(0,\infty)}(\tau) I_{m_{\text{red}}}(m_T) I_{\gamma_{\text{red}}}(\gamma) I_{m_{S\text{red}}}(m_S) I_{\lambda_{\text{red}}}(\lambda) \quad (3.10)
 \end{aligned}$$

donde I_A denota la función indicador de A , es decir: $I_A(x) = \begin{cases} 1 & \text{si } x \in A \\ 0 & \text{si } x \notin A \end{cases}$ y

$$\begin{aligned}
 & \boldsymbol{\varepsilon}(\boldsymbol{\theta}) = \boldsymbol{\varepsilon}(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\phi}, \tau, m_T, \gamma, m_S, \lambda) = \\
 & = \mathbf{Y} - \mathbf{Z}\boldsymbol{\alpha} - \mathbf{X}\boldsymbol{\beta} - \phi_T \mathbf{T}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) - \phi_S \mathbf{S}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) - \phi_{ST} \mathbf{ST}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) - \phi_{TS} \mathbf{TS}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) \quad (3.11)
 \end{aligned}$$

3.3 Estimación del modelo

La estimación de los parámetros se realiza a partir de las distribuciones marginales de la distribución a posteriori (3.10). Dado que esta distribución es analíticamente intratable, recurrimos a los llamados métodos de Monte Carlo basados en la simulación de cadenas de Markov (MCMC) para calcularlas (ver detalles sobre la fundamentación matemática de dichos métodos en el apéndice A).

Estos métodos se basan en la construcción de cadenas de Markov, cuya distribución estacionaria es la distribución que se quiere muestrear (en nuestro caso la distribución (3.10)). Bajo ciertas condiciones de regularidad

detalladas en el apéndice A, dicha cadena converge a su distribución estacionaria. Simulándola un número suficientemente elevado de veces, se obtiene una muestra de la distribución deseada.

En el siguiente apartado se describen los detalles matemáticos de dicha construcción y se proporciona un algoritmo para obtener una muestra de la distribución (3.10). Como resultado de dicho algoritmo se obtiene una muestra

$$\{\boldsymbol{\theta}^{(j)}; j=1, \dots, n_{\text{muestra}}\} = \{(\boldsymbol{\alpha}^{(j)}, \boldsymbol{\beta}^{(j)}, \boldsymbol{\phi}^{(j)}, \boldsymbol{\tau}^{(j)}, m_T^{(j)}, \gamma^{(j)}, m_S^{(j)}, \lambda^{(j)}); j=1, \dots, n_{\text{muestra}}\} \quad (3.12)$$

a partir de la cual se pueden realizar inferencias acerca de los parámetros del modelo (3.1). En particular, se pueden calcular las medias y medianas de las componentes de (3.12), que son estimaciones puntuales de la media y la mediana a posteriori, o intervalos de credibilidad bayesianos con un nivel de credibilidad dado utilizando los cuantiles adecuados (ver apéndice A para más detalles).

3.3.1 Distribuciones completamente condicionadas

El algoritmo utilizado en nuestro caso está basado en el *Gibbs sampling* (apéndice A), para lo cual es necesario especificar las distribuciones completamente condicionadas de (3.10), es decir, las distribuciones de cada una de sus componentes condicionada al resto.

Dichas distribuciones vienen dadas como resultado de los teoremas que aparecen en este apartado. Previamente, se enuncia un lema con el resultado ya clásico en la literatura del análisis bayesiano del modelo lineal general, que fue presentado por Lindley y Smith (1972) y que es de aplicación en las demostraciones de algunos de los teoremas.

Lema 3.1

Sea \mathbf{X} una matriz $n \times p$ de variables independientes, sea \mathbf{Y} un vector $n \times 1$ de datos y $\boldsymbol{\theta}$ el vector de coeficientes $p \times 1$, cuya distribución conjunta viene dada por:

$$\begin{aligned} \mathbf{Y} | \boldsymbol{\theta} &\sim N_n(\mathbf{X}\boldsymbol{\theta}, \boldsymbol{\Sigma}) \\ \boldsymbol{\theta} &\sim N_p(\mathbf{m}_0, \mathbf{C}_0) \end{aligned}$$

La distribución marginal de \mathbf{Y} y la distribución a posteriori de $\boldsymbol{\theta} | \mathbf{Y}$ son respectivamente:

$$\begin{aligned} \mathbf{Y} &\sim N_n(\mathbf{X}\mathbf{m}_0, \boldsymbol{\Sigma} + \mathbf{X}\mathbf{C}_0\mathbf{X}') \\ \boldsymbol{\theta} | \mathbf{Y} &\sim N_p(\mathbf{m}_1, \mathbf{C}_1) \end{aligned}$$

donde $\left\{ \begin{array}{l} \mathbf{C}_1^{-1} = \mathbf{X}' \boldsymbol{\Sigma}^{-1} \mathbf{X} + \mathbf{C}_0^{-1} \\ \mathbf{m}_1 = \mathbf{C}_1 \left[\mathbf{X}' \boldsymbol{\Sigma}^{-1} \mathbf{Y} + \mathbf{C}_0^{-1} \mathbf{m}_0 \right] \end{array} \right.$

Teorema 3.1

La distribución completamente condicionada del parámetro $\boldsymbol{\alpha}$ viene dada por:

$$\boldsymbol{\alpha} | \boldsymbol{\beta}, \boldsymbol{\phi}, \tau, m_T, \gamma, m_S, \lambda, \mathbf{Y}, \mathbf{Z}, \mathbf{X} \sim N_q(\mathbf{m}_\alpha, \mathbf{S}_\alpha) \quad (3.13)$$

$$\text{con } \left\{ \begin{array}{l} \mathbf{m}_\alpha = \mathbf{S}_\alpha \left(\tau \mathbf{Z}_1' \mathbf{W}_1^\alpha \right) \\ \mathbf{S}_\alpha^{-1} = \tau \mathbf{Z}_1' \mathbf{Z}_1 + \boldsymbol{\Sigma}_\alpha^{-1} \end{array} \right.$$

donde:

$$\mathbf{W}^\alpha = \mathbf{Y} - (\mathbf{X}\boldsymbol{\beta} + \phi_T \mathbf{T}(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}) + \phi_S \mathbf{S}(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}) + \phi_{ST} \mathbf{ST}(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}) + \phi_{TS} \mathbf{TS}(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}))$$

Demostración:

Condicionando a los datos $(\mathbf{Y}, \mathbf{Z}, \mathbf{X})$ y al resto de parámetros, se sigue de las ecuaciones (3.1) y (3.2) que:

$$\begin{aligned} \mathbf{W}_1^\alpha \mid \boldsymbol{\alpha} &\sim N_{n-n_0}(\mathbf{Z}_1 \boldsymbol{\alpha}, \tau^{-1} \mathbf{I}_{n-n_0}) \\ \boldsymbol{\alpha} &\sim N_q(\mathbf{0}, \boldsymbol{\Sigma}_\alpha) \end{aligned}$$

el resultado se sigue aplicando el lema 3.1.

Teorema 3.2

La distribución completamente condicionada del parámetro $\boldsymbol{\beta}$ viene dada por:

$$\boldsymbol{\beta} \mid \boldsymbol{\alpha}, \phi, \tau, m_T, \gamma, m_S, \lambda, \mathbf{Y}, \mathbf{Z}, \mathbf{X} \sim N_k(\mathbf{m}_\beta, \mathbf{S}_\beta) \quad (3.14)$$

$$\text{con } \begin{cases} \mathbf{m}_\beta = \mathbf{S}_\beta^{-1} \left(\tau (\mathbf{W}_1^\beta)' \mathbf{F}_1^\beta \right) \\ \mathbf{S}_\beta^{-1} = \tau (\mathbf{W}_1^\beta)' \mathbf{W}_1^\beta + \boldsymbol{\Sigma}_\beta^{-1} \end{cases}$$

donde:

$$\begin{aligned} \mathbf{W}^\beta &= \mathbf{X} - \phi_T \mathbf{TX} - \phi_S \mathbf{SX} - \phi_{ST} \mathbf{STX} - \phi_{TS} \mathbf{TSX} \\ \mathbf{F}^\beta &= \mathbf{Y} - \mathbf{Z}\boldsymbol{\alpha} - \phi_T \mathbf{TY} - \phi_S \mathbf{SY} - \phi_{ST} \mathbf{STY} - \phi_{TS} \mathbf{TSY} \end{aligned}$$

Demostración:

Condicionando a los datos $(\mathbf{Y}, \mathbf{Z}, \mathbf{X})$ y al resto de parámetros, se sigue de las ecuaciones (3.1) y (3.3) que:

$$\begin{aligned} \mathbf{F}_1^\beta \mid \boldsymbol{\beta} &\sim N_{n-n_0}(\mathbf{W}_1^\beta \boldsymbol{\beta}, \tau^{-1} \mathbf{I}_{n-n_0}) \\ \boldsymbol{\beta} &\sim N_k(\mathbf{0}, \boldsymbol{\Sigma}_\beta) \end{aligned}$$

el resultado se obtiene aplicando el lema 3.1.

Teorema 3.3

La distribución completamente condicionada del parámetro ϕ_i con $i \in \{T, S, ST, TS\}$ viene dada por:

$$\phi_i \mid \alpha, \beta, \phi_{-i}, \tau, m_T, \gamma, m_S, \lambda, \mathbf{Y}, \mathbf{Z}, \mathbf{X} \sim N(m_{\phi_i}, s_{\phi_i}^2) \text{ truncada en } (-1, 1) \quad (3.15)$$

$$\text{siendo } \begin{cases} m_{\phi_i} = \frac{(\mathbf{W}_1^\phi)'_i (\mathbf{F}_1^\phi)}{(\mathbf{W}_1^\phi)'_i (\mathbf{W}_1^\phi)_i} \\ s_{\phi_i}^2 = \frac{1}{\tau (\mathbf{W}_1^\phi)'_i (\mathbf{W}_1^\phi)_i} \end{cases}$$

donde:

$$\mathbf{W}^\phi = (\mathbf{T}(\mathbf{Y}-\mathbf{X}\beta), \mathbf{S}(\mathbf{Y}-\mathbf{X}\beta), \mathbf{ST}(\mathbf{Y}-\mathbf{X}\beta), \mathbf{TS}(\mathbf{Y}-\mathbf{X}\beta))$$

$$(\mathbf{F}^{\phi_i}) = \mathbf{Y} - \mathbf{Z}\alpha - \mathbf{X}\beta - (\mathbf{W}^\phi)_{-i} \phi_{-i}$$

donde $(\mathbf{W}^\phi)_i$ denota la i -ésima columna de \mathbf{W}^ϕ , $(\mathbf{W}^\phi)_{-i}$ es la submatriz de \mathbf{W}^ϕ formada al eliminar la i -ésima columna de ésta y ϕ_{-i} es el subvector de ϕ obtenido al eliminarle su i -ésima componente.

Demostración:

Condicionando a los datos $(\mathbf{Y}, \mathbf{Z}, \mathbf{X})$ y al resto de parámetros, a partir de (3.10) podemos escribir que:

Estimación del modelo

$$\begin{aligned}
& [\phi_i | \alpha, \beta, \phi, \tau, m_T, \gamma, m_S, \lambda, \mathbf{Y}, \mathbf{Z}, \mathbf{X}] \propto \\
& \propto \exp \left\{ -\frac{\tau}{2} \left(\mathbf{F}_1^{\phi} - (\mathbf{W}_1^{\phi})_i \phi_i \right)' \left(\mathbf{F}_1^{\phi} - (\mathbf{W}_1^{\phi})_i \phi_i \right) \right\} I_{(-1,1)}(\phi_i) \propto \\
& \propto \exp \left\{ -\frac{\tau}{2} (\phi_i - m_{\phi_i})' (s_{\phi_i}^2)^{-1} (\phi_i - m_{\phi_i}) \right\} I_{(-1,1)}(\phi_i)
\end{aligned}$$

de donde se obtiene el resultado (3.15)

Teorema 3.4

La distribución completamente condicionada del parámetro τ viene dada por:

$$\tau | \alpha, \beta, \phi, m_T, \gamma, m_S, \lambda, \mathbf{Y}, \mathbf{Z}, \mathbf{X} \sim \text{Gamma} \left(\frac{(n-n_0)+d_0}{2}, \frac{(d_0 s_0 + \boldsymbol{\varepsilon}_1' \boldsymbol{\varepsilon}_1)}{2} \right) \quad (3.16)$$

Demostración:

Partiendo de (3.10) se sigue que la densidad de esta distribución es proporcional a:

$$\tau^{\frac{(n-n_0)+d_0-1}{2}} \exp \left[-\frac{\tau}{2} (d_0 s_0 + \boldsymbol{\varepsilon}_1'(\boldsymbol{\theta}) \boldsymbol{\varepsilon}_1(\boldsymbol{\theta})) \right] I_{(0,\infty)}(\tau)$$

de donde se deduce el resultado.

Teorema 3.5

La distribución completamente condicionada de los parámetros $(m_T, \gamma, m_S, \lambda) | \alpha, \beta, \phi, \tau, \mathbf{Y}, \mathbf{Z}, \mathbf{X}$ es una distribución discreta en $\mathbf{m}_{Tred} \times \boldsymbol{\gamma}_{red} \times \mathbf{m}_{Sred} \times \boldsymbol{\lambda}_{red}$ y la función de probabilidad viene dada por:

$$P(m_T = m_{Tred,a}, \gamma = \gamma_{red,b}, m_S = m_{Sred,c}, \lambda = \lambda_{red,d}) \propto$$

$$\exp \left[-\frac{\tau}{2} \left(\boldsymbol{\varepsilon}_1' \left(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\phi}, \tau, \mathbf{m}_{\text{Tred},a}, \gamma_{\text{red},b}, \mathbf{m}_{\text{Sred},c}, \lambda_{\text{red},d} \right) \boldsymbol{\varepsilon}_1 \left(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\phi}, \tau, \mathbf{m}_{\text{Tred},a}, \gamma_{\text{red},b}, \mathbf{m}_{\text{Sred},c}, \lambda_{\text{red},d} \right) \right) \right]$$

con $a = 1, \dots, t$; $b = 1, \dots, g$; $c = 1, \dots, s$ y $d = 1, \dots, \ell$;

y $\boldsymbol{\varepsilon}_1(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\phi}, \tau, \mathbf{m}_T, \gamma, \mathbf{m}_S, \boldsymbol{\lambda})$ viene dado por la expresión (3.11).

Demostración:

Inmediata de (3.10).

3.3.2 Algoritmo de estimación

Una vez que las distribuciones completamente condicionadas han sido obtenidas, el siguiente algoritmo describe el proceso para obtener la muestra (3.12).

Paso 0 Determinar n_{iter} , el número de iteraciones del algoritmo. Extraer una muestra inicial $(\boldsymbol{\alpha}^{(0)}, \boldsymbol{\beta}^{(0)}, \boldsymbol{\phi}^{(0)}, \tau^{(0)}, \mathbf{m}_S^{(0)}, \boldsymbol{\lambda}^{(0)}, \mathbf{m}_T^{(0)}, \gamma^{(0)})$ de la distribución a priori (3.2)-(3.9).

Paso 1 Repetir para $i=1, \dots, n_{\text{iter}}$

1) Extraer $\boldsymbol{\alpha}^{(i)}$ de

$$\boldsymbol{\alpha} | \boldsymbol{\beta}^{(i-1)}, \boldsymbol{\phi}^{(i-1)}, \tau^{(i-1)}, \mathbf{m}_T^{(i-1)}, \gamma^{(i-1)}, \mathbf{m}_S^{(i-1)}, \boldsymbol{\lambda}^{(i-1)}, \mathbf{Y}, \mathbf{Z}, \mathbf{X} \sim N_q \left(\mathbf{m}_\alpha^{(i)}, \mathbf{S}_\alpha^{(i)} \right)$$

donde:

$$\begin{cases} \mathbf{S}_\alpha^{(i)} = \left(\tau^{(i-1)} \mathbf{Z}'_1 \mathbf{Z}_1 + \boldsymbol{\Sigma}_\alpha^{-1} \right)^{-1} \\ \mathbf{m}_\alpha^{(i)} = \left(\mathbf{S}_\alpha^{(i)} \right) \left(\tau^{(i-1)} \mathbf{Z}'_1 \left(\mathbf{W}_1^a \right)^{(i-1)} \right) \end{cases}$$

con:

Estimación del modelo

$$\begin{aligned} (\mathbf{W}^\alpha)^{(i-1)} &= \mathbf{Y} - \mathbf{X}\boldsymbol{\beta}^{(i-1)} - \phi_{\mathbf{T}}^{(i-1)}\mathbf{T}^{(i-1)}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}^{(i-1)}) - \phi_{\mathbf{S}}^{(i-1)}\mathbf{S}^{(i-1)}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}^{(i-1)}) - \\ &\quad - \phi_{\mathbf{ST}}^{(i-1)}\mathbf{S}^{(i-1)}\mathbf{T}^{(i-1)}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}^{(i-1)}) - \phi_{\mathbf{TS}}^{(i-1)}\mathbf{T}^{(i-1)}\mathbf{S}^{(i-1)}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}^{(i-1)}) \end{aligned}$$

2) Extraer $\boldsymbol{\beta}^{(i)}$ de

$$\boldsymbol{\beta} \mid \boldsymbol{\alpha}^{(i)}, \boldsymbol{\phi}^{(i-1)}, \boldsymbol{\tau}^{(i-1)}, \mathbf{m}_{\mathbf{T}}^{(i-1)}, \gamma^{(i-1)}, \mathbf{m}_{\mathbf{S}}^{(i-1)}, \lambda^{(i-1)}, \mathbf{Y}, \mathbf{Z}, \mathbf{X} \sim N_k(\mathbf{m}_{\boldsymbol{\beta}}^{(i)}, \mathbf{S}_{\boldsymbol{\beta}}^{(i)})$$

donde:

$$\begin{cases} \mathbf{S}_{\boldsymbol{\beta}}^{(i)} = \left(\boldsymbol{\tau}^{(i-1)} \left((\mathbf{W}_1^\beta)^{(i-1)} \right)' (\mathbf{W}_1^\beta)^{(i-1)} + \boldsymbol{\Sigma}_{\boldsymbol{\beta}}^{-1} \right)^{-1} \\ \mathbf{m}_{\boldsymbol{\beta}}^{(i)} = (\mathbf{S}_{\boldsymbol{\beta}}^{(i)}) \left(\boldsymbol{\tau}^{(i-1)} \left((\mathbf{W}_1^\beta)^{(i-1)} \right)' (\mathbf{F}_1^\beta)^{(i-1)} \right) \end{cases}$$

con:

$$\begin{aligned} (\mathbf{W}^\beta)^{(i-1)} &= \mathbf{X} - \phi_{\mathbf{T}}^{(i-1)}\mathbf{T}^{(i-1)}\mathbf{X} - \phi_{\mathbf{S}}^{(i-1)}\mathbf{S}^{(i-1)}\mathbf{X} - \phi_{\mathbf{ST}}^{(i-1)}\mathbf{S}^{(i-1)}\mathbf{T}^{(i-1)}\mathbf{X} - \phi_{\mathbf{TS}}^{(i-1)}\mathbf{T}^{(i-1)}\mathbf{S}^{(i-1)}\mathbf{X} \\ (\mathbf{F}^\beta)^{(i-1)} &= \mathbf{Y} - \mathbf{Z}\boldsymbol{\alpha}^{(i)} - \phi_{\mathbf{T}}^{(i-1)}\mathbf{T}^{(i-1)}\mathbf{Y} - \phi_{\mathbf{S}}^{(i-1)}\mathbf{S}^{(i-1)}\mathbf{Y} - \phi_{\mathbf{ST}}^{(i-1)}\mathbf{S}^{(i-1)}\mathbf{T}^{(i-1)}\mathbf{Y} - \phi_{\mathbf{TS}}^{(i-1)}\mathbf{T}^{(i-1)}\mathbf{S}^{(i-1)}\mathbf{Y} \end{aligned}$$

3) Denotando $(\boldsymbol{\phi}(j-))^{(i)} = (\phi_k^{(i)})_{k < j}$ y $(\boldsymbol{\phi}(j+))^{(i)} = (\phi_h^{(i)})_{h > j}$ extraer $\phi_j^{(i)}$ de

$$\begin{aligned} \phi_j \mid \boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, (\boldsymbol{\phi}(j-))^{(i)}, (\boldsymbol{\phi}(j+))^{(i-1)}, \boldsymbol{\tau}^{(i-1)}, \mathbf{m}_{\mathbf{T}}^{(i-1)}, \gamma^{(i-1)}, \mathbf{m}_{\mathbf{S}}^{(i-1)}, \lambda^{(i-1)}, \mathbf{Y}, \mathbf{Z}, \mathbf{X} \\ \sim N(\mathbf{m}_{\phi_j}^{(i)}, \mathbf{S}_{\phi_j}^{(i)}) \text{ truncada en } (-1, 1) \text{ para } j \in \{\mathbf{T}, \mathbf{S}, \mathbf{ST}, \mathbf{TS}\} \text{ donde:} \end{aligned}$$

$$\begin{cases} \mathbf{S}_{\phi_j}^{(i)} = \left(\boldsymbol{\tau}^{(i-1)} \left((\mathbf{W}_1^\phi)_j^{(i)} \right)' (\mathbf{W}_1^\phi)_j^{(i)} \right)^{-1} \\ \mathbf{m}_{\phi_j}^{(i)} = \left((\mathbf{W}_1^\phi)_j^{(i)} \right)' (\mathbf{F}_1^{\phi_j})^{(i)} \left(\left((\mathbf{W}_1^\phi)_j^{(i)} \right)' (\mathbf{W}_1^\phi)_j^{(i)} \right)^{-1} \end{cases}$$

siendo:

$$\begin{aligned} (\mathbf{W}^\phi)^{(i)} &= (\mathbf{T}^{(i-1)}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}^{(i)}), \mathbf{S}^{(i-1)}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}^{(i)}), \mathbf{S}^{(i-1)}\mathbf{T}^{(i-1)}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}^{(i)}), \mathbf{T}^{(i-1)}\mathbf{S}^{(i-1)}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}^{(i)})) \\ (\mathbf{F}^{\phi_j})^{(i)} &= \mathbf{Y} - \mathbf{Z}\boldsymbol{\alpha}^{(i)} - \mathbf{X}\boldsymbol{\beta}^{(i)} - (\mathbf{W}^\phi)^{(i)}_{-j} \boldsymbol{\phi}_{-j}^{(i)} \end{aligned}$$

con $(\mathbf{W}^\phi)^{(i)}_{-j}$ la matriz \mathbf{W}^ϕ sin su j -ésima columna y

$$\boldsymbol{\phi}_j^{(i)} = \left(\boldsymbol{\phi}(j^-)^{(i)}, \boldsymbol{\phi}(j^+)^{(i-1)} \right)'$$

4) Extraer $\tau^{(i)}$ de:

$$\tau \mid \boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\phi}^{(i)}, \mathbf{m}_T^{(i-1)}, \gamma^{(i-1)}, \mathbf{m}_S^{(i-1)}, \boldsymbol{\lambda}^{(i-1)}, \mathbf{Y}, \mathbf{Z}, \mathbf{X} \sim \left(\frac{\mathbf{d}_\tau}{2}, \frac{\mathbf{d}_\tau \mathbf{s}_\tau^{(i)}}{2} \right)$$

$$\text{donde: } \begin{cases} \mathbf{d}_\tau = \mathbf{d}_0 + \mathbf{n} - \mathbf{n}_0 \\ \mathbf{d}_\tau \mathbf{s}_\tau^{(i)} = \mathbf{d}_0 \mathbf{s}_0 + \boldsymbol{\varepsilon}_1' \boldsymbol{\varepsilon}_1 \end{cases}$$

5) Extraer $(\mathbf{m}_T^{(i)}, \gamma^{(i)}, \mathbf{m}_S^{(i)}, \boldsymbol{\lambda}^{(i)})$ de:

$(\mathbf{m}_T, \gamma, \mathbf{m}_S, \boldsymbol{\lambda}) \mid \boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\phi}^{(i)}, \tau^{(i)}, \mathbf{Y}, \mathbf{Z}, \mathbf{X}$ que es una distribución discreta cuyo soporte es $\mathbf{m}_{T_{\text{red}}} \times \gamma_{\text{red}} \times \mathbf{m}_{S_{\text{red}}} \times \boldsymbol{\lambda}_{\text{red}}$ y distribución de probabilidad dada por:

$$\begin{aligned} P(\mathbf{m}_T = \mathbf{m}_{T_{\text{red},a}}, \gamma = \gamma_{\text{red},b}, \mathbf{m}_S = \mathbf{m}_{S_{\text{red},c}}, \boldsymbol{\lambda} = \boldsymbol{\lambda}_{\text{red},d}) &\propto \\ \exp \left[-\frac{\tau^{(i)}}{2} \left(\boldsymbol{\varepsilon}_1' \left(\boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\phi}^{(i)}, \tau^{(i)}, \mathbf{m}_{T_{\text{red},a}}, \gamma_{\text{red},b}, \mathbf{m}_{S_{\text{red},c}}, \boldsymbol{\lambda}_{\text{red},d} \right) \boldsymbol{\varepsilon}_1 \left(\boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\phi}^{(i)}, \tau^{(i)}, \mathbf{m}_{T_{\text{red},a}}, \gamma_{\text{red},b}, \mathbf{m}_{S_{\text{red},c}}, \boldsymbol{\lambda}_{\text{red},d} \right) \right) \right] & \\ \mathbf{a} = 1, \dots, t \quad \mathbf{b} = 1, \dots, g \quad \mathbf{c} = 1, \dots, s \quad \mathbf{d} = 1, \dots, \ell & \end{aligned}$$

Una vez que las n_{iter} iteraciones han sido completadas, se descartan las primeras it_0 necesarias para alcanzar la convergencia a la distribución estacionaria de la cadena. Este número se determina utilizando algunos de los métodos propuestos en la literatura (ver apéndice A para más detalles). En nuestro caso it_0 se determinó mediante inspección visual de la evolución de las cadenas de dichos parámetros y tras realizar un estudio de sensibilidad respecto al punto de comienzo de las mismas.

Una vez determinado it_0 , se extrae una muestra aproximada de la distribución a posteriori que viene dada por:

$$\left\{ \left(\boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\phi}^{(i)}, \boldsymbol{\tau}^{(i)}, \mathbf{m}_S^{(i)}, \boldsymbol{\lambda}^{(i)}, \mathbf{m}_T^{(i)}, \boldsymbol{\gamma}^{(i)} \right) \quad i = it_0, it_0 + s, \dots, n_{\text{iter}} = it_0 + (n_{\text{muestra}} - 1)s \right\}$$

donde el salto s se determina de forma que las componentes de la muestra sean aproximadamente incorreladas. Dicho paso se determina analizando la función de autocorrelación de cada componente eligiendo s tal que la autocorrelación de orden s sea despreciable. En lo que sigue, y con el fin de aliviar la notación, suponemos sin pérdida de generalidad $it_0 = s = 1$.

3.4 Validación del modelo

Una vez que ya se ha estimado el modelo STAR, se lleva a cabo su validación, tanto de forma intramuestral a través del análisis de su bondad de ajuste a los datos, como extramuestralmente, comprobando la habilidad predictiva del modelo.

Para ello, y tal y como propone Zellner (1975), se analiza, tanto de forma gráfica como numérica, la distribución a posteriori de sus residuos, así como su comportamiento predictivo extramuestral mediante un algoritmo de validación tipo *rolling*, que utiliza un muestreo de importancia secuencial inspirado en Ridgeway y Madigan (2002) y Balakrishnan y Madigan (2006)

que actualiza secuencialmente la información que subyace en el proceso. Las dos subsecciones siguientes describen con más detalle los procedimientos mencionados.

3.4.1 Bondad de ajuste

El análisis de la bondad de ajuste del modelo se realiza mediante el estudio a posteriori de sus residuos, analizando la existencia de patrones sistemáticos no capturados por el modelo, tanto en media como en varianza y en autocorrelación.

Los residuos a analizar son los residuos estandarizados $\mathbf{e}(\boldsymbol{\theta}) = \tau^{1/2} \boldsymbol{\varepsilon}(\boldsymbol{\theta})$ donde $\boldsymbol{\varepsilon}(\boldsymbol{\theta})$ viene dado por la expresión (3.11).

Dicho estudio utiliza las distribuciones a posteriori de los mismos, calculadas a partir de (3.10). Para ello se toma la muestra (3.12) y se calculan $\left\{ \mathbf{e}(\boldsymbol{\theta}^{(j)}) = (\tau^{(j)})^{1/2} \boldsymbol{\varepsilon}(\boldsymbol{\theta}^{(j)}) ; j=1, \dots, n_{\text{muestra}} \right\}$ que constituyen una muestra de la distribución a posteriori de $\mathbf{e}(\boldsymbol{\theta})$

En el análisis gráfico de los residuos tipificados, además de estudiar su comportamiento estático calculando estimaciones puntuales o intervalos de credibilidad bayesianos mediante la mediana o los correspondientes cuantiles a posteriori, se analiza su comportamiento dinámico utilizando correlogramas espacio-temporales, con el fin de comprobar si existen dependencias espacio-temporales significativas. Para ello se representan en un diagrama cartesiano dichas autocorrelaciones dadas por:

$$r_k(\boldsymbol{\theta}) = \frac{\sum_{i=n_0+k+1}^n (\varepsilon_i(\boldsymbol{\theta}) - \bar{\varepsilon}_k(\boldsymbol{\theta}))(\varepsilon_{s(i,k)}(\boldsymbol{\theta}) - \bar{\varepsilon}_{s(k)}(\boldsymbol{\theta}))}{\sqrt{\sum_{i=n_0+k+1}^n (\varepsilon_i(\boldsymbol{\theta}) - \bar{\varepsilon}_k(\boldsymbol{\theta}))^2} \sqrt{\sum_{i=n_0+k+1}^n (\varepsilon_{s(i,k)}(\boldsymbol{\theta}) - \bar{\varepsilon}_{s(k)}(\boldsymbol{\theta}))^2}} \quad (3.17)$$

$$\bar{\varepsilon}_k(\boldsymbol{\theta}) = \frac{\sum_{i=n_0+k+1}^n \varepsilon_i(\boldsymbol{\theta})}{n - n_0 - k} \quad \bar{\varepsilon}_{s(k)}(\boldsymbol{\theta}) = \frac{\sum_{i=n_0+k+1}^n \varepsilon_{s(i,k)}(\boldsymbol{\theta})}{n - n_0 - k} \quad (3.18)$$

y donde $s(i,k) \in \{1, \dots, i-1\}$ es tal que la transacción $s(i,k)$ -ésima es el k -ésimo vecino espacial más cercano a la transacción i -ésima. Esta correlación verifica que $r_k \approx N(0, n^{-1})$ si $n \rightarrow \infty$ y los residuos son ruido blanco.

A partir de (3.12) se obtiene la correspondiente muestra $\{r_k(\boldsymbol{\theta}^{(j)}) ; j=1, \dots, n_{\text{muestra}}\}$ de la distribución a posteriori de $r_k(\boldsymbol{\theta})$, a partir de la cual se pueden calcular la mediana y los cuantiles a posteriori adecuados para la construcción de las bandas de credibilidad bayesianas correspondientes a un nivel de credibilidad dado. Estas bandas ayudan a determinar si existen autocorrelaciones espaciales significativas, analizado si el 0 está contenido dentro de sus límites. Observar que la elaboración de dichas bandas está basada en la incertidumbre a posteriori asociada al proceso de estimación de todos los parámetros del modelo (incluyendo los parámetros de vecindad) y no requiere utilizar ningún resultado de tipo asintótico lo cual confiere un mayor grado de fiabilidad y exactitud al estudio realizado.

Para completar el estudio de la bondad del ajuste, se calculan dos criterios numéricos que buscan cuantificar el grado de ajuste del modelo a los datos, a saber, el coeficiente de determinación múltiple y los cubrimientos empíricos de los intervalos predictivos a un paso, los cuales vienen dados por las siguientes expresiones.

- El coeficiente de determinación múltiple:

$$R^2 = 1 - \frac{\sum_{i=h+1}^n (y_i - E[y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \mathbf{g}])^2}{\sum_{i=h+1}^n (y_i - \bar{y})^2} \quad (3.19)$$

donde:

$$E[y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \mathbf{g}] = \int E[y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \boldsymbol{\theta}] g(\boldsymbol{\theta}) d\boldsymbol{\theta}$$

siendo $\mathbf{Z}_i = (\mathbf{z}_1, \dots, \mathbf{z}_i)$, $\mathbf{X}_i = (\mathbf{x}_1, \dots, \mathbf{x}_i)$ e $\mathbf{Y}_i = (y_1, \dots, y_i)$

$$E[y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \boldsymbol{\theta}] = \mathbf{z}_i' \boldsymbol{\alpha} + \mathbf{x}_i' \boldsymbol{\beta} + \mathbf{u}_i' [\phi_T \mathbf{T}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) + \phi_S \mathbf{S}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) + \phi_{ST} \mathbf{ST}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) + \phi_{TS} \mathbf{TS}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})] \quad (3.20)$$

En la validación intramuestral tomamos $h = n_0$, mientras que en la extramuestral $h = n_{est}$ donde $n_{est}+1$ indica el inicio del periodo de validación y $g(\boldsymbol{\theta})$ es la función de densidad de la distribución a posteriori sobre $\boldsymbol{\theta}$ utilizada en el proceso de validación y que viene dada por:

$$g(\boldsymbol{\theta}) = \begin{cases} [\boldsymbol{\theta} | \mathbf{Y}, \mathbf{Z}, \mathbf{X}] & \text{en el caso intramuestral} \\ [\boldsymbol{\theta} | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i] & \text{en el caso } rolling \end{cases}$$

El criterio R^2 es similar al coeficiente de determinación múltiple utilizado en la estadística clásica y estima el porcentaje (en tanto por uno) de la varianza de los datos del periodo considerado, explicada por el modelo. Habitualmente tomará valores entre 0 y 1 de forma que cuanto mayor es su valor mejor es el grado de ajuste del modelo a los datos. A diferencia del coeficiente clásico, este coeficiente R^2 utiliza las esperanzas condicionales $\{E[y_i | \mathbf{Y}_i, \mathbf{Z}_i, \mathbf{X}_i, \mathbf{g}]; i = h+1, \dots, n\}$ para elaborar las predicciones del modelo que minimizan la función de pérdida cuadrática cuando se toma como distribución

a posteriori de los parámetros la distribución g que recoge la incertidumbre asociada al proceso de estimación del modelo.

- *Cubrimientos empíricos de los intervalos predictivos bayesianos al 95 y 99% de credibilidad:*

A partir de los cuantiles 0.5, 2.5, 97.5 y 99.5 de las componentes de la muestra $\{\varepsilon_i(\boldsymbol{\theta}^{(j)}); i = n_0 + 1, \dots, n; j = 1, \dots, n_{\text{muestra}}\}$ se construyen las bandas de credibilidad bayesianas con un nivel del 95% y del 99% para $\{\varepsilon_i(\boldsymbol{\theta}); i = n_0 + 1, \dots, n\}$ que vendrán dadas, respectivamente, por

$$\begin{aligned} & \{(\hat{\varepsilon}_i(\boldsymbol{\theta}; 0.025), \hat{\varepsilon}_i(\boldsymbol{\theta}; 0.975)); i = n_0 + 1, \dots, n\} \text{ y} \\ & \{(\hat{\varepsilon}_i(\boldsymbol{\theta}; 0.005), \hat{\varepsilon}_i(\boldsymbol{\theta}; 0.995)); i = n_0 + 1, \dots, n\} \end{aligned} \quad (3.21)$$

donde $\hat{\varepsilon}_i(\boldsymbol{\theta}; q)$ denota el cuantil q de la muestra $\{\varepsilon_i(\boldsymbol{\theta}^{(j)}); j = 1, \dots, n_{\text{muestra}}\}$ con $0 < q < 1$.

Con el fin de evaluar la bondad de ajuste del modelo se calculan los cubrimientos empíricos de las bandas anteriores los cuales vienen dados por:

$$\begin{aligned} \text{Cub}(0.95) &= \frac{100}{n - n_0} \sum_{i=n_0+1}^n I_{(\hat{\varepsilon}_i(\boldsymbol{\theta}; 0.025), \hat{\varepsilon}_i(\boldsymbol{\theta}; 0.975))}(\mathbf{0}) \\ \text{Cub}(0.99) &= \frac{100}{n - n_0} \sum_{i=n_0+1}^n I_{(\hat{\varepsilon}_i(\boldsymbol{\theta}; 0.005), \hat{\varepsilon}_i(\boldsymbol{\theta}; 0.995))}(\mathbf{0}) \end{aligned} \quad (3.22)$$

Si la bondad de ajuste del modelo es buena $\text{Cub}(0.95) \approx 0.95$ y $\text{Cub}(0.99) \approx 0.99$.

3.4.2 Validación extramuestral

Al igual que en el capítulo anterior, para llevar a cabo la validación extramuestral, los datos observados se dividen en dos submuestras: una de estimación $\mathbf{Y}_{\text{est}} = (y_1, \dots, y_{n_{\text{est}}})'$ y otra de validación $\mathbf{Y}_{\text{val}} = (y_{n_{\text{est}}+1}, \dots, y_n)'$ donde n_{est} es el tamaño de la muestra de estimación que verifica $n_0 < n_{\text{est}} < n$. Utilizando la información proporcionada por la muestra de estimación $(\mathbf{Y}_{\text{est}}, \mathbf{Z}_{\text{est}}, \mathbf{X}_{\text{est}})$ donde $\mathbf{Z}_{\text{est}} = (\mathbf{z}_1, \dots, \mathbf{z}_{n_{\text{est}}})$ y $\mathbf{X}_{\text{est}} = (\mathbf{x}_1, \dots, \mathbf{x}_{n_{\text{est}}})$, se realiza la estimación de los parámetros del modelo calculando la distribución a posteriori $\boldsymbol{\theta} | \mathbf{Y}_{\text{est}}, \mathbf{Z}_{\text{est}}, \mathbf{X}_{\text{est}}$.

El estudio de validación extramuestral que se va a realizar en este caso para el modelo STAR va a ser de tipo *rolling*. La predicción en cada paso del periodo de validación utiliza toda la información disponible hasta el instante previo, realizando una reestimación completa del modelo. Esto significa que, para cada paso, el periodo de estimación se expande secuencialmente al añadirse la información proporcionada por las transacciones más recientes acaecidas en dicho periodo.

La densidad que debe ser calculada en cada etapa es $[y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i]$. Para ello se utiliza un muestreo secuencial de importancia basado en ideas de Ridgeway y Madigan (2002) y Balakrishnan y Madigan (2006). Dicho muestreo secuencial esta basado en una actualización de los pesos de importancia fundamentada en la expresión:

$$\begin{aligned} [y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i] &= \int [y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \boldsymbol{\theta}] [\boldsymbol{\theta} | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i] d\boldsymbol{\theta} = \\ &= \int [y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \boldsymbol{\theta}] \frac{[\boldsymbol{\theta} | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i]}{g(\boldsymbol{\theta})} g(\boldsymbol{\theta}) d\boldsymbol{\theta} = \end{aligned}$$

$$\begin{aligned}
&= \int [y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \boldsymbol{\theta}] \omega(\boldsymbol{\theta} | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i; \mathbf{g}) g(\boldsymbol{\theta}) d\boldsymbol{\theta} \approx \\
&\approx \frac{\sum_{j=1}^{n_{\text{muestra}}} [y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \boldsymbol{\theta}^{(j)}] \omega(\boldsymbol{\theta}^{(j)} | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i; \mathbf{g})}{\sum_{j=1}^{n_{\text{muestra}}} \omega(\boldsymbol{\theta}^{(j)} | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i; \mathbf{g})} \quad (3.23)
\end{aligned}$$

donde g denota la densidad de importancia, $\{\boldsymbol{\theta}^{(j)}; j=1, \dots, n_{\text{muestra}}\}$ es una muestra de g y los pesos de importancia son $\omega(\boldsymbol{\theta} | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i; \mathbf{g}) = \frac{\pi(\boldsymbol{\theta} | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i)}{g(\boldsymbol{\theta})}$.

El algoritmo se basa en tomar inicialmente $g(\boldsymbol{\theta}) = [\boldsymbol{\theta} | \mathbf{Y}_{\text{est}}, \mathbf{Z}_{\text{est}}, \mathbf{X}_{\text{est}}]$, $\omega(\boldsymbol{\theta} | \mathbf{Y}_{\text{est}}, \mathbf{Z}_{\text{est}+1}, \mathbf{X}_{\text{est}+1}) = 1$ y actualizar secuencialmente los pesos de importancia mediante la fórmula:

$$\omega(\boldsymbol{\theta} | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i; \mathbf{g}) = [y_{i-1} | \mathbf{Y}_{i-2}, \mathbf{Z}_{i-1}, \mathbf{X}_{i-1}, \boldsymbol{\theta}] \omega(\boldsymbol{\theta} | \mathbf{Y}_{i-2}, \mathbf{Z}_{i-1}, \mathbf{X}_{i-1}; \mathbf{g}) \quad (3.24)$$

Dado que g puede deteriorarse a lo largo del tiempo, conviene monitorizar secuencialmente la calidad de la aproximación. Para ello utilizamos el tamaño muestral efectivo ESS propuesto por Kong y otros. (1994) que viene dado por la expresión:

$$\text{ESS}_i(\mathbf{g}) = \frac{\left(\sum_{j=1}^{n_{\text{muestra}}} \omega(\boldsymbol{\theta}^{(j)} | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i; \mathbf{g}) \right)^2}{\sum_{j=1}^{n_{\text{muestra}}} \omega(\boldsymbol{\theta}^{(j)} | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i; \mathbf{g})^2} \quad (3.25)$$

de forma que si $\text{ESS}_i(\mathbf{g}) \leq p n_{\text{muestra}}$ con $0 < p < 1$ se concluye que la calidad de la densidad de importancia ha empeorado significativamente y debe ser, por tanto, reemplazada. En este caso se toma $g(\boldsymbol{\theta}) = [\boldsymbol{\theta} | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i]$ calculada

utilizando el *Gibbs sampling* de la sección 3.3.2. En nuestro caso hemos tomado $p = 0.5$ que es el valor utilizado en Balakrishnan y Madigan (2006).

A continuación se detalla el algoritmo:

3.4.2.1 Muestreo de importancia secuencial para la validación extramuestral tipo *rolling*

Paso 0: Inicialización

Aplicar el algoritmo descrito en la sección 3.3.2 para obtener una muestra $\{\boldsymbol{\theta}^{(j)}; j=1, \dots, n_{\text{muestra}}\}$ de la distribución $\boldsymbol{\theta} \mid \mathbf{Y}_{\text{est}}, \mathbf{Z}_{\text{est}}, \mathbf{X}_{\text{est}}$.

Tomar $\mathbf{g}(\boldsymbol{\theta}) = [\boldsymbol{\theta} \mid \mathbf{Y}_{\text{est}}, \mathbf{Z}_{\text{est}}, \mathbf{X}_{\text{est}}]$ y poner $i = n_{\text{est}} + 1$.

Inicializar los pesos de importancia:

$$\left\{ \omega(\boldsymbol{\theta}^{(j)} \mid \mathbf{Y}_{i-1}, \mathbf{X}_i, \mathbf{Z}_i; \mathbf{g}) = 1; j = 1, \dots, n_{\text{muestra}} \right\}$$

Fijar $p \in (0, 1)$ y poner $k = 0$ y $\text{ESS}_i(\mathbf{g}) = n_{\text{muestra}}$.

Paso 1: (Condición de finalización)

Poner $k = k + 1$. Si $i + k > n$ PARAR. En otro caso ir al paso 2.

Paso 2: (Actualización de los pesos de importancia)

$$\text{Para } j = 1, \dots, n_{\text{muestra}}. \omega(\boldsymbol{\theta}^{(j)} \mid \mathbf{Y}_{i+k-1}, \mathbf{Z}_{i+k}, \mathbf{X}_{i+k}; \mathbf{g}) =$$

$$= \left[y_{i-1} \mid \mathbf{Y}_{i+k-2}, \mathbf{Z}_{i+k-1}, \mathbf{X}_{i+k-1}, \boldsymbol{\theta}^{(j)} \right] \omega(\boldsymbol{\theta}^{(j)} \mid \mathbf{Y}_{i+k-2}, \mathbf{Z}_{i+k-1}, \mathbf{X}_{i+k-1}; \mathbf{g})$$

Calcular $\text{ESS}_{i+k}(\mathbf{g})$ utilizando la expresión (3.25).

Si $\text{ESS}_{i+k}(\mathbf{g}) > p n_{\text{muestra}}$ ir al paso 1. En otro caso ir al paso 3

Paso 3. (Reemplazamiento de la densidad de importancia)

Poner $i = i+k$. Tomar $g(\boldsymbol{\theta}) = [\boldsymbol{\theta} | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i]$ y ejecutar el algoritmo 3.3.2 para obtener una muestra $\{\boldsymbol{\theta}^{(j)}; j=1, \dots, n_{\text{muestra}}\}$ de g .

Poner $k = 0$ y $\{\omega(\boldsymbol{\theta}^{(j)} | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i; g) = 1; j=1, \dots, n_{\text{muestra}}\}$. Ir al paso 1.

3.4.2.2 Elaboración de predicciones y bondad de ajuste extramuestral

De esta forma, y para cada valor de $i \in \{n_{\text{est}}+1, \dots, n\}$ se obtiene una muestra ponderada

$$\left\{ \boldsymbol{\theta}^{(j)}; \frac{\omega(\boldsymbol{\theta}^{(j)} | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i; g)}{\sum_{k=1}^{n_{\text{muestra}}} \omega(\boldsymbol{\theta}^{(k)} | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i; g)}; j=1, \dots, n_{\text{muestra}} \right\} \quad (3.26)$$

de la distribución a posteriori $[\boldsymbol{\theta} | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i]$ que podremos utilizar para realizar predicciones a varios horizontes vista mediante el muestreo de composición.

En nuestro caso estamos interesados en realizar predicciones del valor de y_i que obtenemos repitiendo para $k=1, \dots, n_{\text{muestra}}$ los siguientes pasos:

Extraer de la distribución discreta (3.26) una muestra:

$$\boldsymbol{\theta}^{*(k)} = (\boldsymbol{\alpha}^{*(k)}, \boldsymbol{\beta}^{*(k)}, \boldsymbol{\phi}^{*(k)}, \boldsymbol{\tau}^{*(k)}, m_{\Gamma}^{*(k)}, \gamma^{*(k)}, m_S^{*(k)}, \lambda^{*(k)})$$

y posteriormente, $y_i^{(k)}$ de $[y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \boldsymbol{\theta}^{*(k)}]$, que, en nuestro caso proviene

de una distribución $N(\mu_i^{(k)}, (\tau^{*(k)})^{-1})$ donde $\mu_i^{(k)} = E[y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \boldsymbol{\theta}^{*(k)}]$ viene dada por la expresión (3.20).

Fruto de este algoritmo se obtiene una muestra $\{y_i^{(k)}; k=1, \dots, n_{\text{muestra}}\}$ de la distribución predictiva $[y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i]$, necesaria para elaborar predicciones puntuales de y_i utilizando la mediana a posteriori o para construir intervalos predictivos bayesianos de un nivel de credibilidad dado a partir de los correspondientes cuantiles de la muestra. Análogamente, se pueden construir predicciones e intervalos predictivos bayesianos para el nivel de precios, sin más que considerar la muestra $\{p_i^{(k)} = \exp(y_i^{(k)}); k=1, \dots, n_{\text{muestra}}\}$ de la distribución predictiva $[p_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i]$ con $p_i = \exp(y_i)$. Además, utilizando un estimador tipo Blackwell-Rao (ver apéndice A) se puede estimar la esperanza predictiva $E[y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i]$ mediante la expresión:

$$\frac{1}{n_{\text{muestra}}} \sum_{k=1}^{n_{\text{muestra}}} E(y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \boldsymbol{\theta}^{*(k)}) \quad (3.27)$$

Así mismo, y con el fin de evaluar el comportamiento extramuestral del modelo, se estudia la existencia de patrones sistemáticos de los errores de predicción extramuestrales tipificados. Para ello se utilizan las muestras $\left\{e_i^{(j)} = \sqrt{\tau^{*(j)}} (y_i - y_i^{(j)}); j=1, \dots, n_{\text{muestra}}; i=n_{\text{est}}+1, \dots, n\right\}$ y se realizan estudios de cubrimiento empírico y de autocorrelación espacio-temporal similares a los descritos en el proceso de validación intramuestral.

3.5 Simplificación del modelo

Tras el estudio de bondad del ajuste y de validación extramuestral del modelo STAR, se estudia una posible simplificación del mismo. Para ello utilizamos criterios de comparación de modelos que evalúan el grado de

bondad de ajuste a los datos de cada uno de ellos, así como el comportamiento predictivo extramuestral de los mismos y que describimos a continuación.

En lo que sigue, denotaremos de forma genérica con M el modelo que se considera en cada situación, siendo $[Y | Z, X, \theta_M]$ su función de densidad y θ_M su vector de parámetros.

Los criterios utilizados son los siguientes:

- *El criterio DIC (Deviance Information Criterium)*

Este criterio es una generalización del criterio AIC y fue propuesto por primera vez por Spiegelhalter y otros (2002) para subsanar su falta de justificación asintótica en modelos jerárquicos, y aparece rutinariamente calculado en programas estadísticos tan populares como WINBUGS. Está basado en el estadístico desviación (*deviance*) dado por:

$$D_M(\theta_M) = -2 \log[Y | Z, X, \theta_M]$$

y viene dado por la expresión:

$$DIC(M) = \bar{D} + p_D \quad (3.28)$$

donde $\bar{D} = E[D_M(\theta_M) | Y, Z, X]$ mide el grado de ajuste del modelo M a los datos y $p_D = \bar{D} - D_M(\bar{\theta}_M)$, al cual Spiegelhalter y otros (2002) denominan “*número efectivo de parámetros*” del modelo, y que es un término que penaliza por el grado de complejidad del modelo, siendo $\bar{\theta}_M$ un estimador puntual de θ_M . De esta forma, cuanto menor es el valor del criterio, mejor es su ajuste a los datos observados.

El término p_D depende de la parametrización $\boldsymbol{\theta}_M$ y es igual a $E[D_M(\boldsymbol{\theta}_M) - D_M(\bar{\boldsymbol{\theta}}_M) | \mathbf{Y}, \mathbf{Z}, \mathbf{X}]$, buscando de este modo cuantificar en términos de la desviación, el error cometido al estimar la distribución $[Y | Z, X, \boldsymbol{\theta}_M]$ por medio de la distribución $[Y | Z, X, \bar{\boldsymbol{\theta}}_M]$. No está claro sin embargo, la forma de elegir la parametrización $\boldsymbol{\theta}_M$ ni el estimador $\bar{\boldsymbol{\theta}}_M$. Spiegelhalter y otros (2002) toman $\bar{\boldsymbol{\theta}}_M = E[\boldsymbol{\theta}_M | \mathbf{Y}, \mathbf{Z}, \mathbf{X}]$, demostrando que cuando $\log [Y | Z, X, \boldsymbol{\theta}_M]$ es log-cóncava en $\boldsymbol{\theta}_M$, entonces p_D es positiva. Además, tal y como Celeux y otros (2006) apuntan, p_D puede ser negativa si \mathbf{Y} o $\boldsymbol{\theta}_M$ contienen componentes discretas, como por ejemplo en la estimación de mixturas (ver Marín y otros, 2005). Celeux y otros (2006) aconsejan para evitar este aspecto indeseable del criterio DIC, tomar $\bar{\boldsymbol{\theta}}_M$ igual a la moda a posteriori o, para evitar los problemas generados por la dependencia de la parametrización, tomar como criterio

$$DIC^*(M) = \bar{D} + p_D^* \quad (3.29)$$

con $p_D^* = \bar{D} + 2\log[Y | \mathbf{Y}, \mathbf{Z}, \mathbf{X}, M]$, donde:

$$[Y | \mathbf{Y}, \mathbf{Z}, \mathbf{X}, M] = \int [Y | \mathbf{X}, \mathbf{Z}, \boldsymbol{\theta}_M][\boldsymbol{\theta}_M | \mathbf{Y}, \mathbf{X}, \mathbf{Z}] d\boldsymbol{\theta}_M \quad (3.30)$$

es la distribución predictiva a posteriori utilizada por Aitkin (1991) como herramienta para calcular el llamado *factor Bayes a posteriori*.

Celeux y otros (2006) argumentan que:

$$p_D^* = E[D_M(\boldsymbol{\theta}_M) + 2\log[Y | \mathbf{Y}, \mathbf{Z}, \mathbf{X}, M] | \mathbf{Y}, \mathbf{Z}, \mathbf{X}]$$

Simplificación del modelo

es invariante a la reparametrización considerada y mide, en términos de desviación, el error cometido al estimar la distribución $[\mathbf{Y} | \mathbf{Z}, \mathbf{X}, \boldsymbol{\theta}_M]$ por medio de la distribución predictiva a posteriori $[\mathbf{Y} | \mathbf{Y}, \mathbf{Z}, \mathbf{X}, \mathbf{M}]$, de gran relevancia a la hora de elaborar predicciones extramuestrales bayesianas.

En ambos casos el valor de los criterios DIC y DIC* se calcula por simulación, partiendo de la distribución a posteriori de $\boldsymbol{\theta}$. Así $E[\boldsymbol{\theta} | \mathbf{Y}, \mathbf{Z}, \mathbf{X}]$ se calcula a partir de la media muestral de las componentes de la muestra (3.12) mientras que $E[D(\boldsymbol{\theta}) | \mathbf{Y}, \mathbf{Z}, \mathbf{X}]$ viene dada aproximadamente por:

$$\frac{-2}{n_{\text{muestra}}} \sum_{j=1}^{n_{\text{muestra}}} \sum_{i=n_0+1}^n \log[y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \boldsymbol{\theta}^{(j)}]$$

donde

$$\log[y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \boldsymbol{\theta}^{(j)}] = \frac{1}{2} \left(\log \left(\frac{\tau^{(j)}}{2\pi} \right) - \tau^{(j)} \varepsilon_i^2(\boldsymbol{\theta}^{(j)}) \right) \quad (3.31)$$

con $\varepsilon_i(\boldsymbol{\theta}^{(j)})$ la i -ésima componente del vector $\boldsymbol{\varepsilon}(\boldsymbol{\theta})$ dado en la expresión (3.11). Así mismo, la densidad predictiva a posteriori $[\mathbf{Y} | \mathbf{Y}, \mathbf{Z}, \mathbf{X}, \mathbf{M}]$ se calcula mediante la expresión:

$$\frac{1}{n_{\text{muestra}}} \sum_{j=1}^{n_{\text{muestra}}} \prod_{i=n_0+1}^n [y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \boldsymbol{\theta}^{(j)}] \quad (3.32)$$

- Logaritmo de la densidad predictiva extramuestral

Este criterio evalúa el logaritmo de la densidad predictiva de las observaciones acaecidas en el periodo de validación, condicionado a la

información proporcionada por las observaciones del periodo de estimación y viene dado por la expresión:

$$\text{LPRED}(M) = \log[\mathbf{Y}_{\text{val}} | \mathbf{Y}_{\text{est}}, \mathbf{Z}, \mathbf{X}, M] \quad (3.33)$$

Específicamente en nuestra aplicación, se calcula a partir de la información proporcionada por la muestra (3.12). Para ello se utiliza la siguiente fórmula que estima la densidad predictiva $[\mathbf{Y}_{\text{val}} | \mathbf{Y}_{\text{est}}, \mathbf{Z}, \mathbf{X}, M]$

$$\frac{1}{n_{\text{muestra}}} \sum_{j=1}^{n_{\text{muestra}}} [\mathbf{Y}_{\text{val}} | \mathbf{Y}_{\text{est}}, \mathbf{Z}, \mathbf{X}, \boldsymbol{\theta}^{(j)}] = \frac{1}{n_{\text{muestra}}} \sum_{j=1}^{n_{\text{muestra}}} \prod_{i=n_{\text{est}}+1}^n [y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \boldsymbol{\theta}^{(j)}]$$

con $[y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \boldsymbol{\theta}]$ la densidad de una normal de media $E[y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \boldsymbol{\theta}]$ cuya expresión viene dada en (3.20) y varianza τ^{-1} .

Este criterio se utiliza para calcular el *factor Bayes parcial* (PBF) dado por la expresión:

$$\text{PBF}(M, M') = \frac{\exp(\text{LPRED}(M))}{\exp(\text{LPRED}(M'))} \quad (3.34)$$

cuya dependencia de la distribución a priori sobre los parámetros del modelo es menor que en el factor Bayes tradicional.

Finalmente, utilizamos los siguientes 3 criterios de comparación basados en los resultados del proceso de validación extramuestral tipo *rolling* descrito en la sección anterior.

- *La raíz del error cuadrático medio*

$$\text{RMSE}(M) = \sqrt{\frac{1}{n-n_{\text{est}}} \sum_{i=n_{\text{est}}+1}^n (y_i - E(y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, M))^2} \quad (3.35)$$

donde $E(y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i)$ se calcula mediante la expresión (3.27).

Simplificación del modelo

- *La desviación absoluta media extramuestral*

$$\text{MAD}(M) = \frac{1}{n-n_{\text{est}}} \sum_{i=n_{\text{est}}+1}^n |y_i - \text{Med}(y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, M)| \quad (3.36)$$

donde $\text{Med}(y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i)$ denota la mediana de la densidad predictiva $[y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i]$, que se calcula con el procedimiento descrito en la sección anterior.

Tanto para el criterio RMSE, como para el criterio MAD, cuanto menor es su valor, mejor es el comportamiento predictivo extramuestral del modelo M.

- *Logaritmo de la densidad predictiva extramuestral a 1 paso*

Los dos criterios anteriores tienen el inconveniente de que evalúan la capacidad predictiva del modelo en términos puntuales sin tener en cuenta la incertidumbre asociada a dicha predicción. Con el fin de recoger este aspecto en el proceso de comparación, utilizamos el logaritmo de la densidad predictiva extramuestral a un paso dado por la expresión:

$$\text{LPRED}_{\text{rol}}(M) = \sum_{i=n_{\text{est}}+1}^n \log[y_i | \mathbf{Y}_{i-1}, \mathbf{X}_i, \mathbf{Z}_i, M] \quad (3.37)$$

donde $[y_i | \mathbf{Y}_{i-1}, \mathbf{X}_i, \mathbf{Z}_i, M]$ se calcula por medio de la expresión (3.23).

Mayor valor en este criterio indica mejor capacidad predictiva.

3.6 Resultados

Una vez descrita la metodología estadística utilizada para estimar y evaluar la bondad de ajuste y capacidad predictiva del modelo STAR descrito en (3.1), en esta sección se exponen los resultados obtenidos al aplicarla al análisis de los datos presentados en el capítulo 2, sección 2.6.1.

Tal y como se ha comentado con anterioridad, la variable dependiente del modelo es el logaritmo del precio de la vivienda transaccionada y las independientes son las enumeradas en 2.6.3.

3.6.1 Estimación del modelo

Con el fin de que la distribución a priori sea poco informativa tomamos $\Sigma_{\alpha} = 10000\mathbf{I}_4$, $\Sigma_{\beta} = 10000\mathbf{I}_{18}$, $d_0 = 0.01$ y $s_0 = 1$. Los resultados expuestos son poco sensibles a estos valores de los hiperparámetros, dado que el ratio tamaño muestral/parámetro ($788/23 \approx 34$) es elevado.

El análisis del diagrama de autocorrelaciones espaciales de la Figura 2.4 del capítulo 2 muestra la existencia de dependencias significativas hasta un orden de 34 retardos. Por dicha razón se han tomado para las redes del número de vecinos espaciales y temporales $m_{Tred} = m_{Sred} = \{0, 5, \dots, 30, 35\}$; ($m_T = 35$ corresponde, aproximadamente, a un mes de transacciones).

Así mismo, se propone $\gamma_{red} = \lambda_{red} = \{0.2, 0.4, 0.6, 0.8, 1\}$. Este elenco de valores permite la elección de un número suficiente de posibilidades para los parámetros γ y λ , sin elevar excesivamente el coste computacional.

Al igual que se hizo en el capítulo 2, se toma $n_0 = 57$ observaciones para disminuir el efecto de las transacciones iniciales sobre las estimaciones de los parámetros y que corresponden al periodo comprendido entre Noviembre de 2002 a Diciembre de 2002.

Resultados

El algoritmo descrito en el apartado 3.3.2 se ejecuta durante 10000 iteraciones. Las Figuras 3.1 a 3.7 muestran la evolución de las cadenas correspondientes a cada uno de los parámetros del modelo así como los correlogramas correspondientes.

De la inspección de dichas figuras se concluye que la convergencia se alcanza tras aproximadamente 1000 iteraciones. Descartando dichas iteraciones, un análisis de las autocorrelaciones revela que, para la mayor parte de los parámetros, la autocorrelación de las series resultantes se hace despreciable a partir de 10 retardos. Únicamente la cadena correspondiente al coeficiente β_{isup} (coeficiente hedónico de la superficie, ver Figura 3.2) muestra un decrecimiento más lento de la dependencia. Sin embargo, un estudio de sensibilidad de las inferencias sobre dicho parámetro respecto a la longitud del salto entre iteraciones no ha revelado diferencias significativas con los resultados expuestos. Por dicha razón, hemos decidido tomar una muestra cada 10 pasos para todos los parámetros. El error estándar de simulación ha sido inferior a 0.005 en todos los casos.

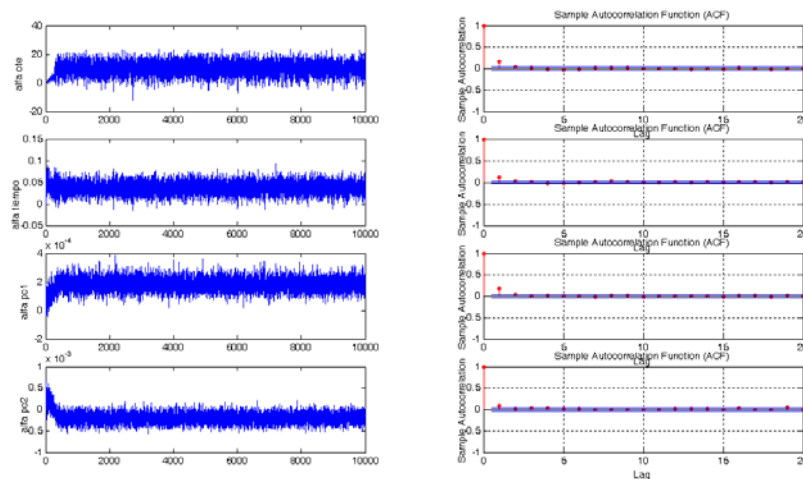


Figura 3.1: Iteraciones del algoritmo de Gibbs y funciones de autocorrelación para las cadenas correspondientes a los coeficientes α de la constante, el ind. de tendencia temporal, PC1 y PC2, respectivamente

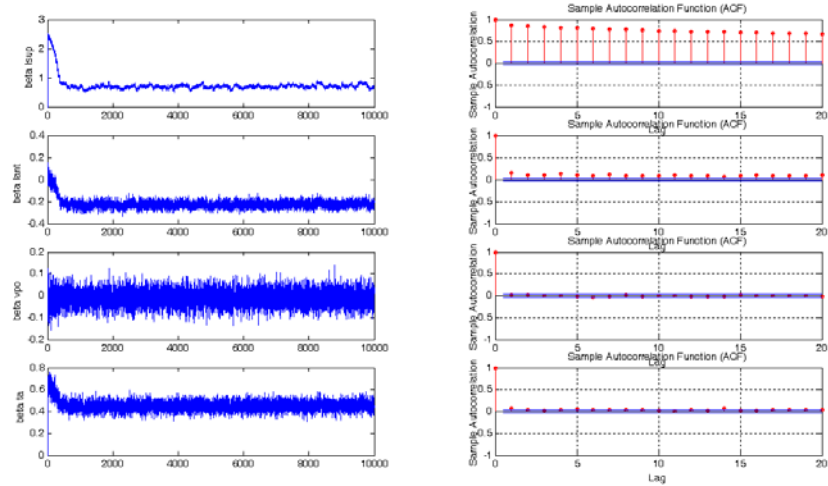


Figura 3.2: Iteraciones del algoritmo de Gibbs y funciones de autocorrelación para las cadenas correspondientes a los coeficientes β de log superficie, log antigüedad, VPO y tipo de adquisición, respectivamente

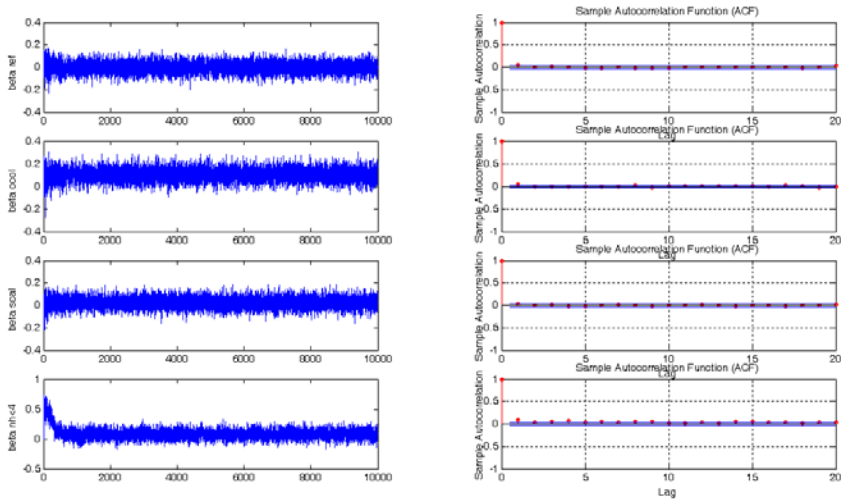


Figura 3.3: Iteraciones del algoritmo de Gibbs y funciones de autocorrelación para las cadenas correspondientes a los coeficientes β de refrigeración, calefacción colectiva, sin calefacción y n° habitaciones <4

Resultados

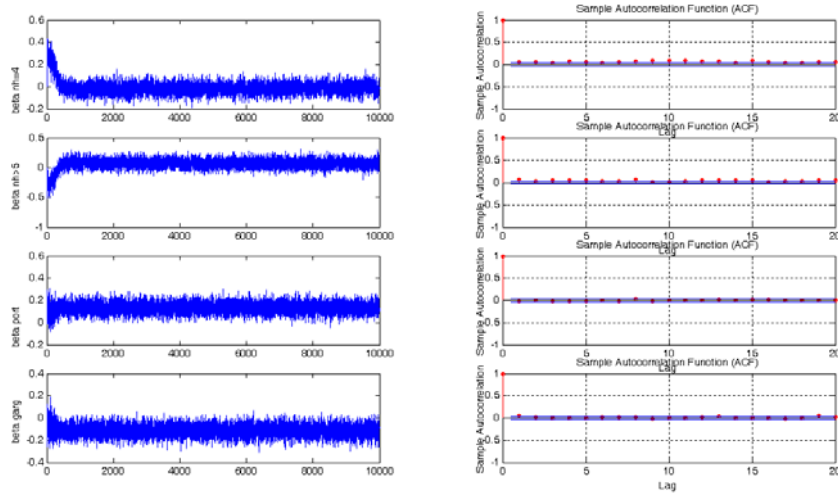


Figura 3.4: Iteraciones del algoritmo de Gibbs y funciones de autocorrelación para las cadenas correspondientes a los coeficientes β_n^4 de habitaciones = 4, β_n^5 de habitaciones > 5, portero y garaje, respectivamente

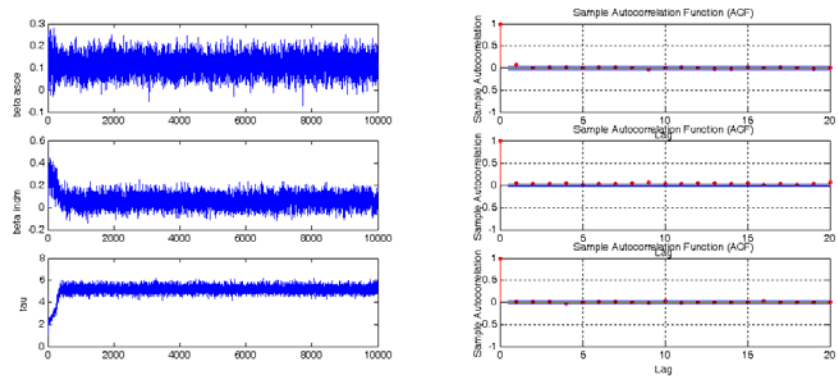


Figura 3.5: Iteraciones del algoritmo de Gibbs y funciones de autocorrelación para las cadenas correspondientes a los coeficientes β_{ascor} de ascensor e indicador de datos ausentes, y la última del parámetro τ

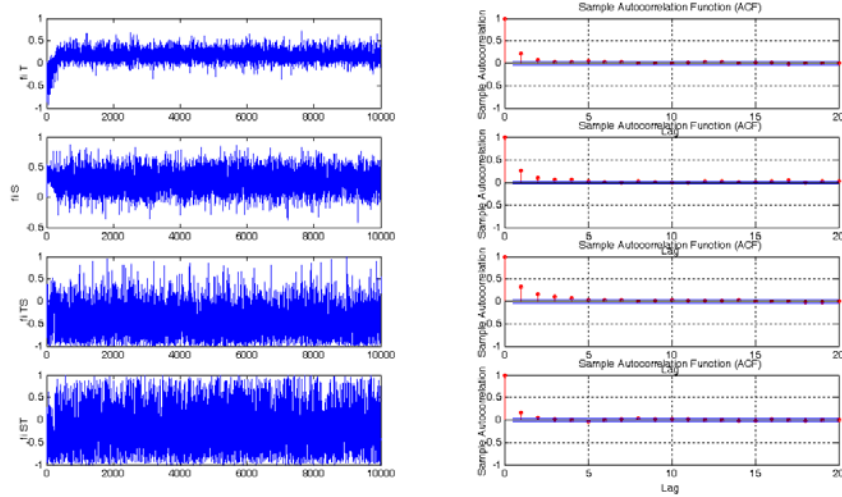


Figura 3.6: Iteraciones del algoritmo de Gibbs y funciones de autocorrelación para las cadenas correspondientes a los coeficientes ϕ_T , ϕ_S , ϕ_{TS} y ϕ_{ST}

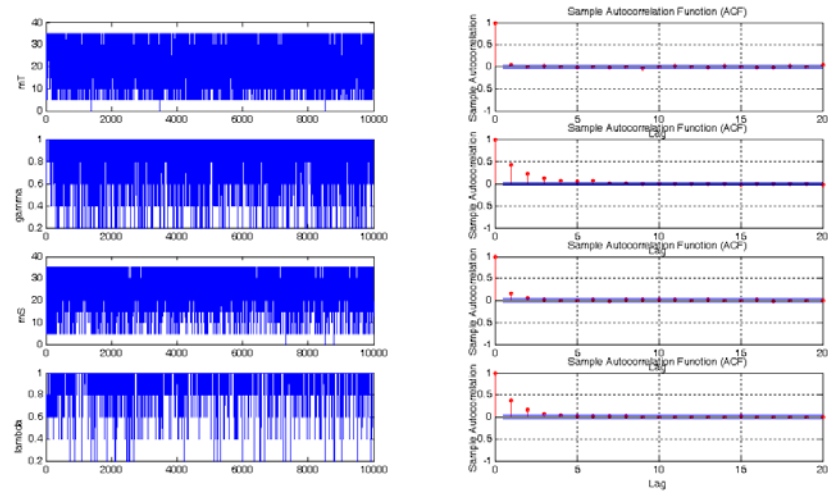


Figura 3.7: Iteraciones del algoritmo de Gibbs y funciones de autocorrelación para las cadenas correspondiente a los coeficientes m_T , γ , m_S y λ

En la Tabla 3.1 se muestran las estimaciones puntuales de los parámetros del modelo STAR, así como los límites de los intervalos de credibilidad bayesianos al 95% contruidos a partir de los cuantiles 2.5 y 97.5 de sus distribuciones a posteriori. Así mismo, en las Figuras 3.8 a 3.13 se muestran las densidades a posteriori de dichos parámetros calculadas a partir de la muestra obtenida por el algoritmo 3.3.2 y aplicando un estimador tipo kernel no paramétrico de la función de densidad utilizando el programa *ksdensity* de MATLAB 6.5.

Se observa que las estimaciones de los parámetros hedónicos del modelo (coeficientes α y β), así como su grado de significación, coinciden en términos generales con las estimaciones obtenidas en el capítulo anterior (ver Tabla 2.4). Este hecho es debido a que las distribuciones a posteriori de estos parámetros son aproximadamente normales. Las mayores discordancias se dan en los coeficientes ϕ_i y, más concretamente, en los coeficientes ϕ_{ST} y ϕ_{TS} que no resultan significativos en la estimación bayesiana. Esta diferencia puede ser ocasionada por la falta de normalidad de las densidades a posteriori de estos parámetros, así como por un mayor grado de incertidumbre (ver Figura 3.13) provocado por la presencia de no linealidades en la ecuación de la media condicionada del modelo (3.1).

Finalmente, en la Tabla 3.1 se muestran las estimaciones correspondientes a los parámetros de vecindad m_T , m_S , γ y λ . Así mismo, en la Figura 3.14 se muestra la distribución a posteriori de dichos parámetros. Las estimaciones coinciden, en términos generales, con las obtenidas en el capítulo anterior. No obstante, se aprecia la existencia de una gran incertidumbre asociada a la estimación del parámetro m_T no puesta de manifiesto con el tratamiento frecuentista (ver Figura 3.14). Este hecho no ocurre en el resto de los parámetros en los que se observa una gran concentración de masa de

probabilidad, en el caso de m_s en $\{20,25\}$, para γ en $\{0.8,1\}$ y para λ en el valor máximo de la red $\{1\}$ con una probabilidad superior al 80%.

La incertidumbre en la elección del número de vecinos temporales m_T , se acompaña de altos valores en sus ponderaciones γ que asignan influencias muy parecidas a todos los vecinos considerados.

En cuanto a los elevados valores del parámetro m_s , pueden considerarse menos informativos, ya que el área de interés no es muy grande y con más de 20 transacciones que afectan espacialmente a una dada se puede entender que la zona de influencia para una transacción se expande suficientemente lejos. Las ponderaciones adjudicadas a estos vecinos espaciales, al ser $\lambda=1$, indican una asignación de influencia igual para todas ellas.

Estos valores respaldan la utilización que en la práctica hacen los tasadores inmobiliarios, de la información procedente de transacciones previas en el tiempo y próximas en el espacio (las denominadas transacciones comparables) para imputar un valor a una vivienda.

Todo esto, puede ser el reflejo de un periodo con un vertiginoso ritmo de aumento en el precio de las viviendas que, independientemente de su situación espacial, afectaron por igual a toda el área de estudio, dando lugar a un número más bien alto de transacciones consideradas influyentes, valorando de forma similar la información que todas y cada una de ellas pudieran proporcionar en la estimación del precio de una transacción dada.

Tabla 3.1: Estimaciones de los parámetros (en negrita los significativos para un nivel de credibilidad del 95% y los marcados con ⁺ al 90%)

VARIABLES	STAR			STARs			hedónico		
	C 2.5	Mediana	C 97.5	C 2.5	Mediana	C 97.5	C 2.5	Mediana	C 97.5
Constante	0.9901	10.0069	18.0446	2.2805	6.3694	8.7444	8.2443	8.7743	9.3095
Tiempo	0.0096	0.0367	0.0616	0.0088	0.0278	0.0478	0.0231	0.0369	0.0507
PC1	0.0001	0.0002	0.0003	0.0001	0.0002	0.0003	0.0002	0.0002	0.0003
PC2	-0.0004	-0.0002 ⁺	0.0000	-0.0005	-0.0002 ⁺	0.0000	-0.0005	-0.0003	-0.0001
Log (superficie)	0.6065	0.7070	0.8489	0.6105	0.7200	0.8172	0.6004	0.7168	0.8357
Log (antigüedad)	-0.2818	-0.2274	-0.1756	-0.2806	-0.2278	-0.1722	-0.2764	-0.2269	-0.1753
VPO	-0.0766	-0.0086	0.0581				-0.0825	-0.0125	0.0500
Tipo de adquisición	0.3736	0.4501	0.5290	0.3690	0.4459	0.5235	0.3729	0.4543	0.5311
Refrigeración	-0.0927	-0.0016	0.0881				-0.0966	0.0048	0.0985
Calefacción colectiva	0.0000	0.0975 ⁺	0.2059	-0.0029	0.0770 ⁺	0.1669	-0.0081	0.0962 ⁺	0.2107
Sin calefacción	-0.0759	0.0255	0.1108				-0.0848	0.0129	0.1075
Nº habitación es <4	-0.0547	0.0797	0.2065				-0.0425	0.0891	0.2133
Nº habitaciones =4	-0.1013	-0.0151	0.0733				-0.1125	-0.0135	0.0738
Nº habitaciones >5	-0.0535	0.0635	0.1901				-0.0345	0.0800	0.2033
Portero	0.0382	0.1309	0.2187	0.0415	0.1309	0.2102	0.0396	0.1313	0.2178
Garaje	-0.2143	-0.1169	-0.0208	-0.2120	-0.1131	-0.0189	-0.2272	-0.1237	-0.0302
Ascensor	0.0294	0.1165	0.2084	0.0404	0.1168	0.1931	0.0378	0.1261	0.1921
Indicador missing	-0.0530	0.0579	0.1585				-0.0464	0.0490	0.1580
ϕ_T	-0.0574	0.1604 ⁺	0.4351	-0.0133	0.1224 ⁺	0.3822			
ϕ_S	-0.0481	0.2457 ⁺	0.6029	-0.0460	0.1263 ⁺	0.5707			
ϕ_{ST}	-0.9471	-0.4142	0.3677						
ϕ_{TS}	-0.8915	-0.2319	0.8470						
τ	4.6615	5.1796	5.7939	4.6363	5.1642	5.7506	4.6043	5.1571	5.7319
m_T	5	20	35	5	20	35			
γ	0.4	0.8	1	0.2	0.8	1			
m_S	5	25	35	5	20	35			
λ	0.6	1	1	0.2	1	1			

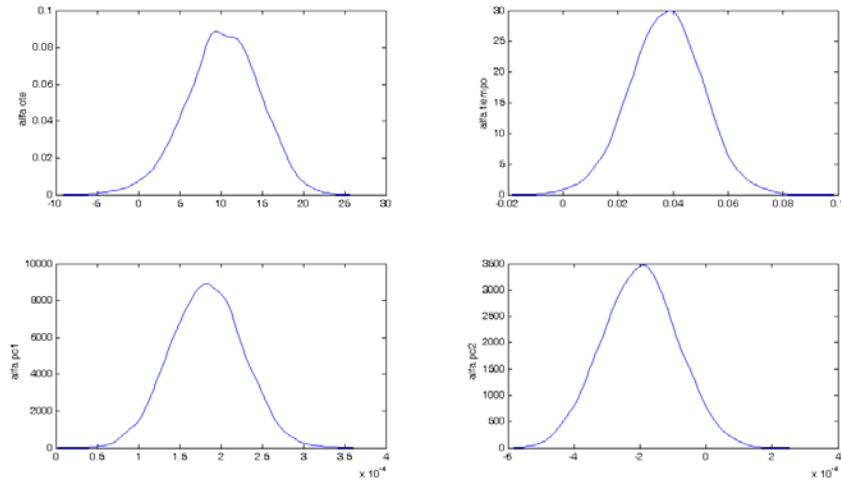


Figura 3.8: Densidades a posteriori de los coeficientes α . De izquierda a derecha y de arriba abajo las correspondientes a la constante, el indicador de tiempo, la componente PC1 y la componente PC2, respectivamente

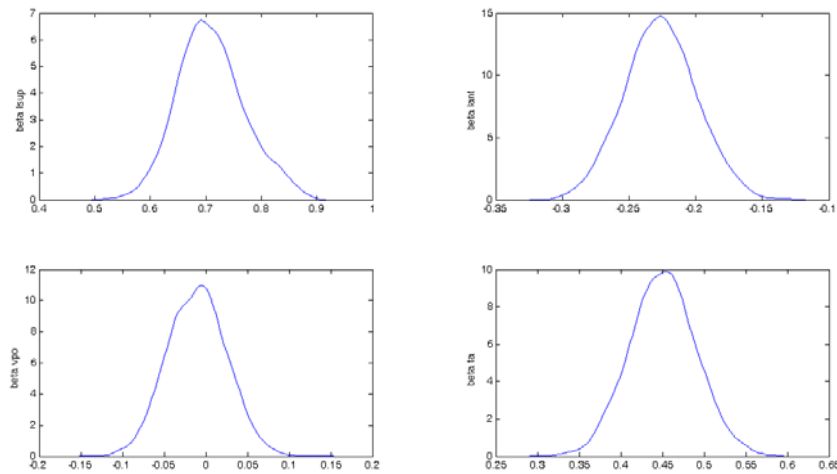


Figura 3.9: De izquierda a derecha y de arriba abajo densidades a posteriori de los coeficientes β correspondientes al logaritmo de superficie, logaritmo de antigüedad, VPO y tipo de adquisición, respectivamente

Resultados

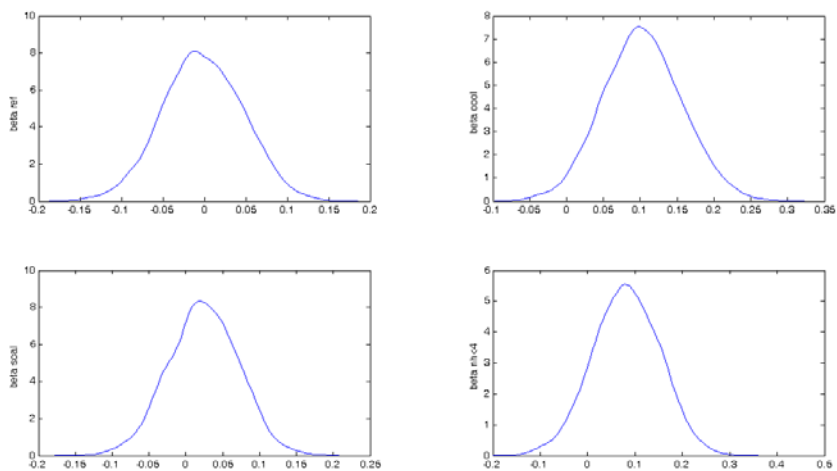


Figura 3.10: De izquierda a derecha y de arriba abajo densidades a posteriori de los coeficientes β correspondientes a las variables refrigeración, calefacción colectiva, sin calefacción y n° de habitaciones < 4, respectivamente

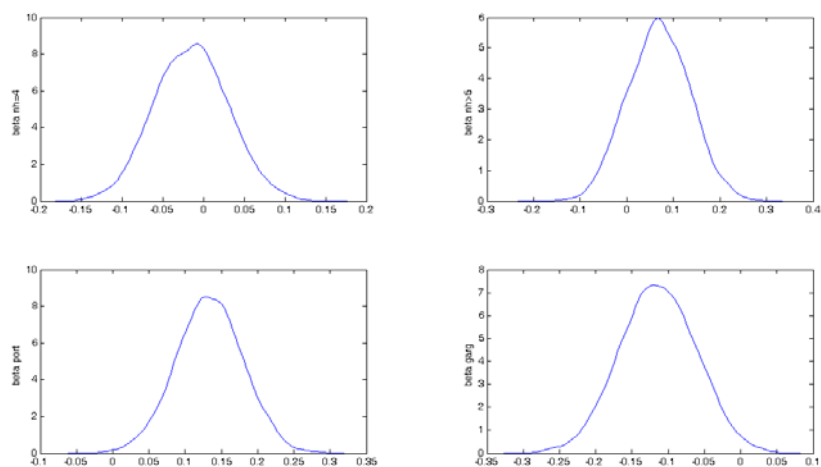


Figura 3.11: De izquierda a derecha y de arriba abajo densidades a posteriori de los coeficientes β correspondientes a las variables n° de habitaciones = 4, n° de habitaciones > 5, portero y garaje, respectivamente

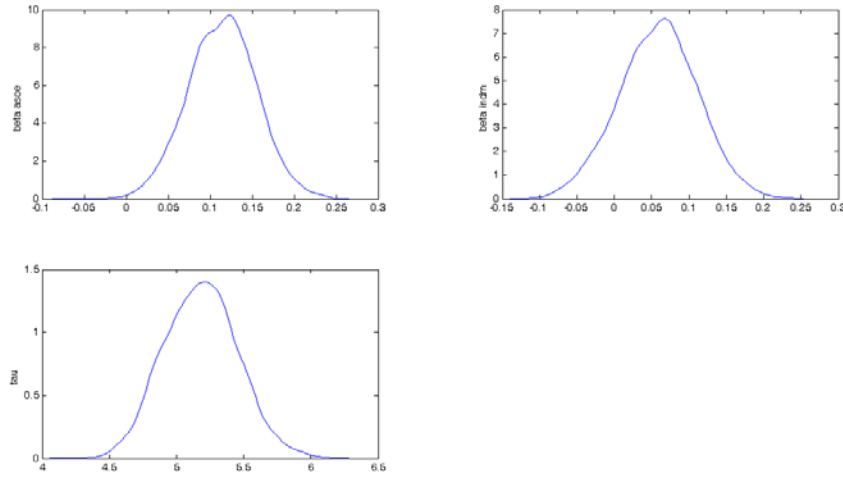


Figura 3.12: De izquierda a derecha y de arriba abajo densidades a posteriori de los coeficientes β correspondientes a las variables ascensor, indicador de datos ausentes y del parámetro τ , respectivamente

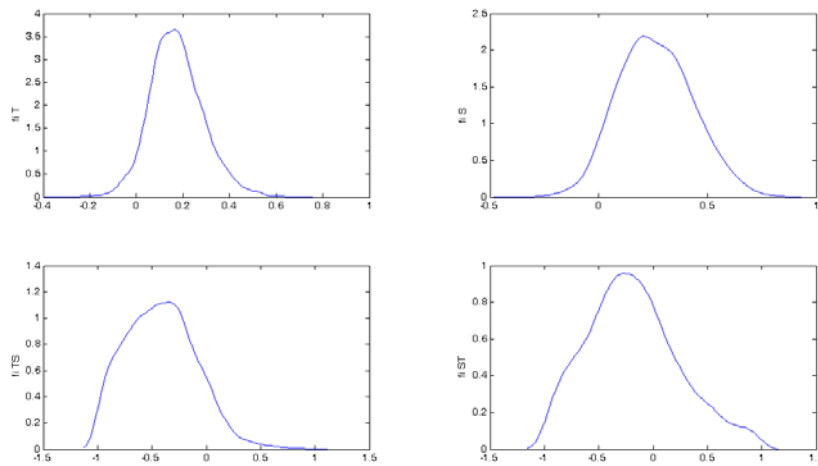


Figura 3.13: De izquierda a derecha y de arriba abajo densidades de los coeficientes ϕ_T , ϕ_S , ϕ_{ST} y ϕ_{TS} , respectivamente

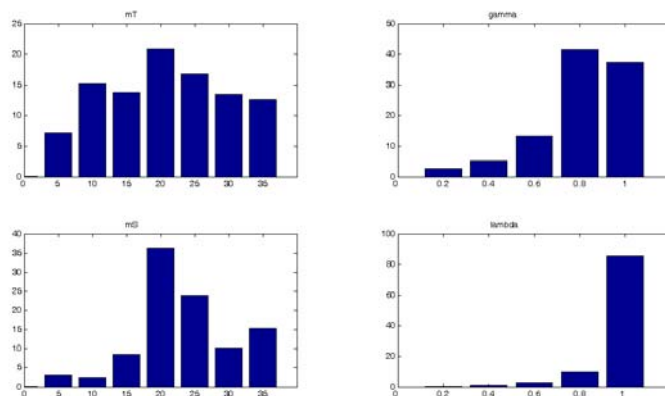


Figura 3.14: De izquierda a derecha y de arriba abajo gráficos de probabilidad de m_T , γ , m_S y λ , respectivamente

3.6.2 Validación del modelo

El estudio de validación del modelo STAR se lleva a cabo intramuestral y extramuestralmente. En el caso de la validación extramuestral el periodo de estimación es el comprendido entre Noviembre de 2002 y Diciembre de 2003 ($n_{est} = 457$), tomándose como periodo de validación el año 2004 completo ($n - n_{est} = 331$).

Así, en las Tabla 3.2 y 3.3 y en las Figuras 3.15 a 3.22 se muestran los resultados tanto del estudio de la bondad de ajuste intramuestral, como del comportamiento predictivo extramuestral del modelo.

Las Figuras 3.15 y 3.19, en las que aparecen los errores intramuestrales y extramuestrales tipificados con sus correspondientes bandas de credibilidad tanto al 95% como al 99%, indican que en ninguno de los casos, existen desviaciones sistemáticas de los errores, presentando un comportamiento adecuado en ambas situaciones.

Así mismo, en las Figuras 3.17 y 3.18 y en la Tabla 3.2 se realiza un estudio de los residuos tipificados, calculados utilizando como estimador del parámetro θ , su media a posteriori. Este análisis, al igual que ocurría en el capítulo 2, pone de manifiesto la existencia de una notable asimetría negativa, debida fundamentalmente a la presencia de varios atípicos, así como una importante leptocurtosis.

En las Figuras 3.16 y 3.20 se muestran las autocorrelaciones residuales intra y extramuestrales con las respectivas bandas de credibilidad al 95%, se puede apreciar que no existen autocorrelaciones significativas en ninguna de las dos situaciones.

Los gráficos de las predicciones tanto para el precio como para el logaritmo del precio, que se presentan en las Figuras 3.21 y 3.22 respectivamente, señalan un adecuado comportamiento predictivo del modelo, que recoge correctamente la asimetría que aparece en los precios de las viviendas consideradas en la base de datos.

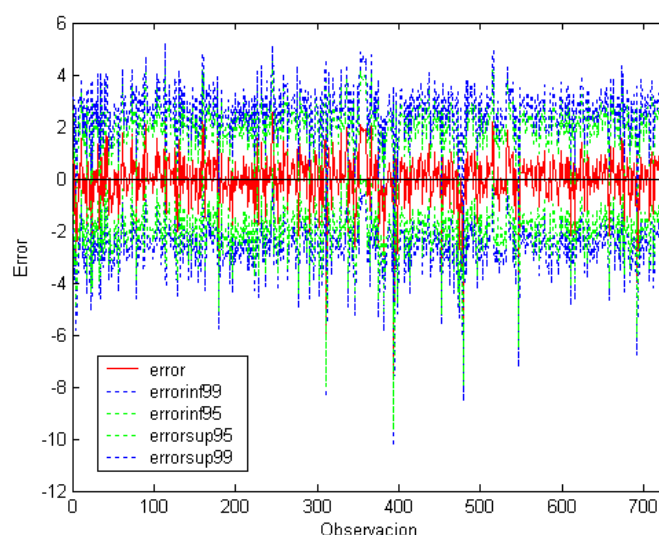


Figura 3.15: Errores intramuestrales del modelo STAR (con bandas de credibilidad al 95% y 99% en verde y azul, respectivamente)

Resultados

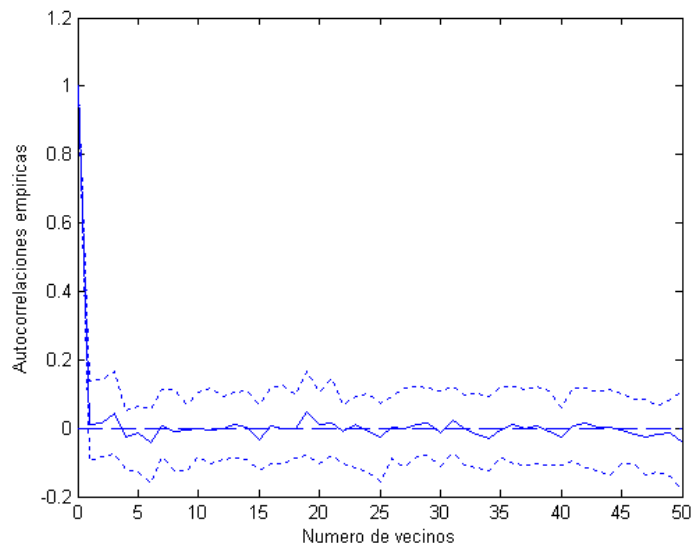


Figura 3.16: Autocorrelaciones espaciales residuales intramuestrales del modelo STAR (en líneas punteadas las bandas de credibilidad al 95%)

Tabla 3.2: Distribución de los errores tipificados intramuestrales del modelo STAR (como estimador de los parámetros su media a posteriori)

ERRORES TIPIFICADOS Modelo STAR	
Mínimo	-7.4623
1%	-3.0440
5%	-1.6753
10%	-1.1641
25%	-0.4898
50%	0.1220
75%	0.6197
90%	1.0924
95%	1.4499
99%	2.1091
Máximo	2.6865
Media	0.0114
Mediana e 	0.5663
Asimetría	-1.4351
Curtosis	9.9662

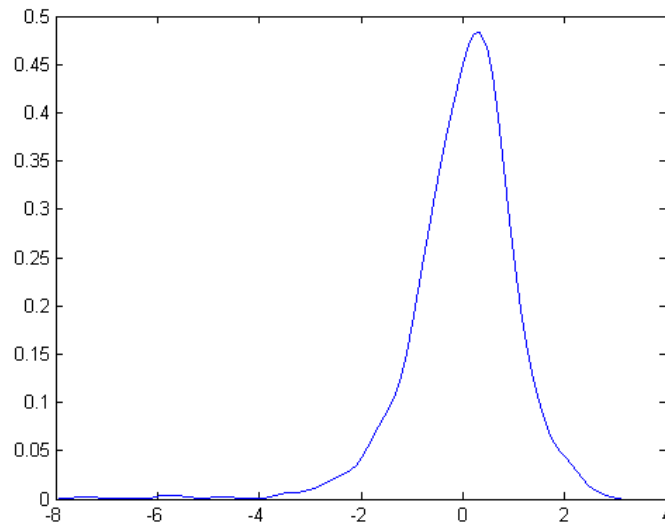


Figura 3.17: Estimación de la densidad de los errores tipificados intramuestrales calculados tomando como estimador de los parámetros su media a posteriori

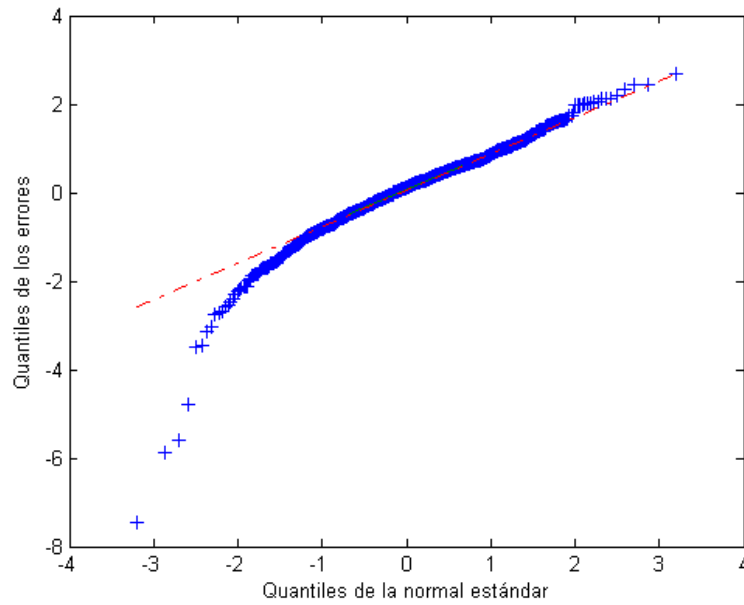


Figura 3.18: Q-Q plot de los errores tipificados intramuestrales calculados tomando como estimador de los parámetros su media a posteriori

Resultados

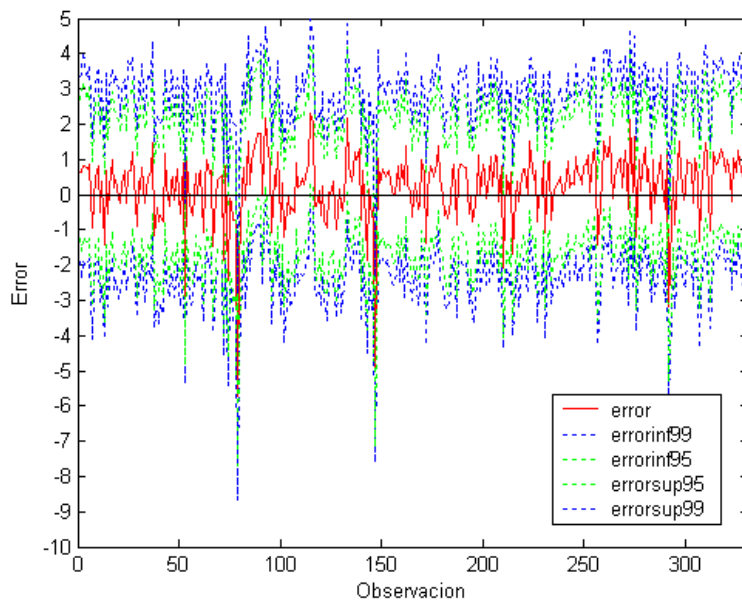


Figura 3.19: Errores tipificados extramuestrales (*rolling*) del modelo STAR

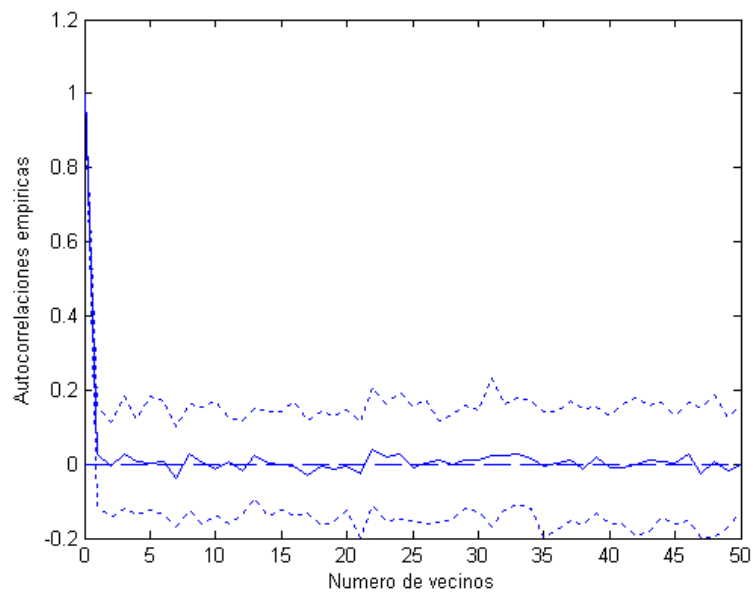


Figura 3.20: Autocorrelaciones espaciales residuales extramuestrales (*rolling*) del modelo STAR (en líneas rayadas las bandas de credibilidad al 95%)

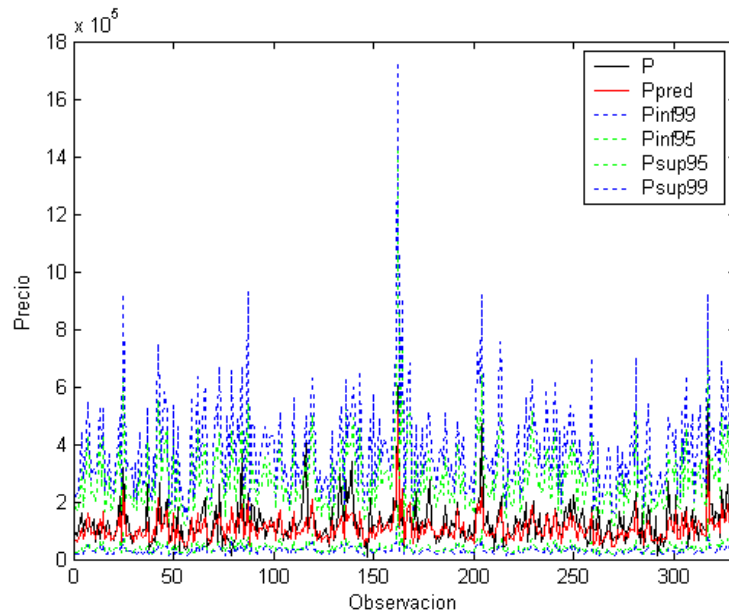


Figura 3.21: Predicciones extramuestrales (*rolling*) del precio para el modelo STAR

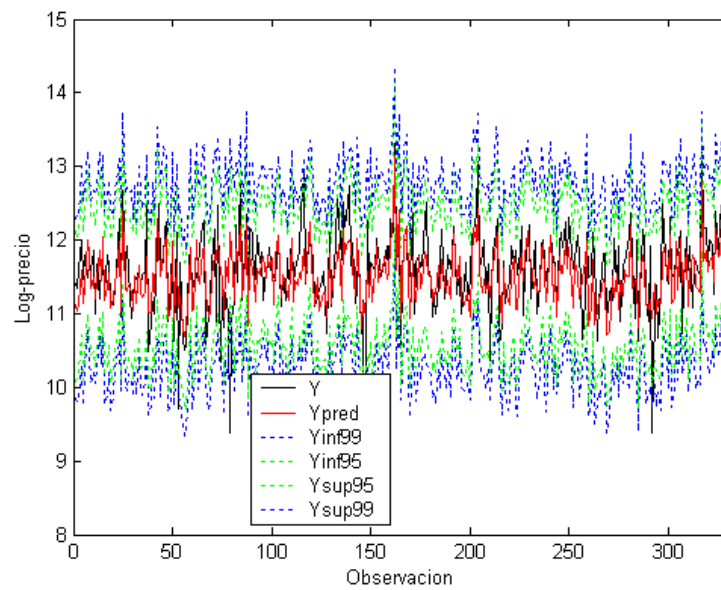


Figura 3.22: Predicciones extramuestrales (*rolling*) del logaritmo del precio para el modelo STAR

Finalmente, y en cuanto a los criterios utilizados para medir la bondad del ajuste, en la Tabla 3.3 se observa que los cubrimientos al 99%, tanto intramuestral como extramuestralmente, son correctos, pero el cubrimiento al 95% en el caso extramuestral difiere significativamente del valor teórico con un pvalor de 0.037. Podemos concluir, por tanto, que la bondad de ajuste del modelo es correcta a excepción de la falta de normalidad de los residuos debido a la presencia de un grupo de atípicos negativos y a la presencia de leptocurtosis. Posteriormente, se propone una forma de tratar este problema con el fin de obtener estimaciones más robustas de los parámetros del modelo.

Tabla 3.3: Validación del modelo STAR

CRITERIOS	Intramuestrales	Extramuestrales (<i>rolling</i>)
R^2	0.4552	0.4305
CUB95	95.76	96.68*
CUB99	98.50	98.79

*Existen diferencias significativas con el resultado teórico al 5%

3.6.3 Simplificación del modelo

En esta sección se lleva a cabo un proceso de comparación del modelo STAR con una simplificación del mismo, que denotaremos de aquí en adelante STARs. Este modelo reducido contiene todas las variables que han resultado significativas al 10% en el proceso de estimación del modelo STAR (ver Tabla 3.1) y coincide con el considerado como modelo simplificado en el capítulo 2. Se incorpora, así mismo, al proceso de comparación el modelo hedónico tradicional, que al carecer de parámetros que capturen efectos de tipo espacial y/o temporal, resulta de interés para poder evaluar la significación práctica de los términos que en ambos modelos STAR y STARs tratan de capturar dichos efectos.

Los resultados de la comparación se muestran en la Tabla 3.4. Se observa que, con la única excepción del criterio DIC, el modelo reducido STARs es el que presenta el mejor comportamiento respecto a los 3 criterios confirmando la superioridad del mismo ya encontrada en el capítulo 2. Sin embargo, el número efectivo de parámetros de los modelos STAR es negativo. Como se mostrará en el capítulo 4, ello es debido al tratamiento discreto dado a los parámetros de ponderación λ y γ . Esto no ocurre con el criterio DIC* propuesto por Celeux y otros (2006) que selecciona el modelo STARs como el mejor modelo de los 3 comparados debido a la mejor calidad de su distribución predictiva a posteriori como estimación de las densidades muestrales de Y .

Tabla 3.4: Criterios de comparación para los modelos STAR, STARs y hedónico.

CRITERIO	MODELO		
	STAR	STARs	hedónico
DIC	848.45	862.88	895.94
\bar{D}	873.04	873.23	877.35
p_D	-24.59	-10.35	18.59
DIC*	880.55	878.15	882.96
P_D^*	7.51	4.91	5.61
LPRED	-181.41	-174.39	-179.07
PBF(Modelo STARs)	0.0009	1118.79	0.0093
RMSE	0.4507	0.4179	0.4194
MAD	0.3452	0.2998	0.3123
LPRED _{rol}	-206.73	-185.85	-190.53

Podemos concluir, por tanto, que los resultados obtenidos confirman los ya mostrados en el capítulo 2 cuando se tienen en cuenta todas las fuentes de incertidumbre asociadas a la estimación de los parámetros del modelo.

3.7 Conclusiones

En este capítulo se ha llevado a cabo un análisis bayesiano del modelo espacio temporal autorregresivo STAR con efectos de vecindad. Dicho análisis, además de evitar la utilización de resultados asintóticos que solo son aplicables suponiendo conocidos los valores de los parámetros de vecindad, permite incorporar en la comparación de modelos, así como en la elaboración de predicciones extramuestrales, la incertidumbre asociada al proceso de estimación de todos los parámetros y especialmente, de los coeficientes autorregresivos y del número de vecinos espaciales y temporales con sus correspondientes ponderaciones.

Los resultados obtenidos en la aplicación han sido los esperados para los coeficientes hedónicos del modelo con una gran concordancia, tanto en valores como en grado de significación, con los extraídos por medio de la estimación frecuentista, corroborando los resultados expuestos en el capítulo anterior. En el caso de la estimación de algunos de los parámetros de vecindad, el análisis bayesiano ha sacado a la luz la existencia de una gran incertidumbre y algunos de los parámetros autorregresivos ϕ_i han dejado de ser significativos. Este hecho puede ser debido a la falta de normalidad en sus distribuciones a posteriori, que ha quedado claramente de manifiesto y que sin embargo, en la estimación llevada a cabo en el capítulo 2 se daba por sentado.

El tratamiento de los parámetros de vecindad mediante redes discretas de valores, aunque eficiente computacionalmente con redes pequeñas, no lo es si son amplias. Además, el procedimiento seguido hasta este momento, obvia

que los parámetros de ponderación λ y γ son continuos, hecho que puede ser la causa de la negatividad del número efectivo de parámetros en el criterio DIC. En el capítulo siguiente se propone una modificación del algoritmo que considera λ y γ como variables continuas y permite tratar eficientemente redes más amplias de valores de m_T y m_S .

Capítulo 4

Tratamiento continuo de las ponderaciones de los efectos de vecindad

4.1 Introducción

En el capítulo anterior se ha presentado una aproximación bayesiana para estimar el modelo espacio temporal autorregresivo STAR con efectos de vecindad. El tratamiento seguido es eficiente computacionalmente si las redes de posibles valores de los parámetros de vecindad son pequeñas. Sin embargo, esta eficiencia se pierde cuando las redes aumentan de tamaño. Además, el tratamiento discreto dado a los parámetros de ponderación es poco realista debido a su carácter esencialmente continuo, lo cual puede ser el motivo de la negatividad del número efectivo de parámetros en el criterio DIC.

En este capítulo se propone una extensión mediante distribuciones continuas en las ponderaciones de los parámetros de vecindad. Además, se incorpora una modificación en el algoritmo de estimación que permite un procesamiento computacionalmente menos costoso de redes más amplias del número de vecinos espaciales y temporales. De este modo, se amplía la libertad en el proceso de estimación para seleccionar aquellos valores que

mejor representen a los datos y/o ayuden a progresar al modelo en su habilidad predictiva. Tras la realización de la estimación y validación del modelo, se presentan los resultados de la aplicación a la base de datos, para poder comprobar los efectos de estos cambios, analizando su aportación tanto en la estimación, como en la interpretación de los parámetros.

Dado que la metodología que se va a seguir es similar a la presentada en el capítulo anterior, se continua con la misma estructura de trabajo, y únicamente se reseñan las modificaciones que aparecen sobre lo ya mostrado.

El capítulo está organizado del siguiente modo: en la sección 4.2 se presenta el nuevo tratamiento del modelo. En el apartado 4.3 se muestran las innovaciones en los procesos de estimación y validación. La sección 4.4 detalla los resultados de la aplicación a la base de datos y en el apartado 4.5 se exponen las conclusiones del capítulo.

4.2 Tratamiento continuo de las ponderaciones

El modelo STAR con efectos de vecindad que utilizamos en esta sección es el descrito anteriormente y viene dado por la expresión

$$\mathbf{Y} = \mathbf{Z}\boldsymbol{\alpha} + \mathbf{X}\boldsymbol{\beta} + \phi_T\mathbf{T}(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}) + \phi_S\mathbf{S}(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}) + \phi_{ST}\mathbf{ST}(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}) + \phi_{TS}\mathbf{TS}(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}) + \boldsymbol{\varepsilon} \quad (4.1)$$

con $\boldsymbol{\varepsilon} \sim N_n(\mathbf{0}, \sigma^2 \mathbf{I}_n)$.

Como resulta de rigor para llevar a cabo el tratamiento bayesiano del modelo, especificamos a continuación la distribución a priori sobre sus parámetros.

4.2.1 Distribución a priori

La distribución a priori viene dada por:

$$\boldsymbol{\alpha} \sim N_q(\mathbf{0}, \boldsymbol{\Sigma}_\alpha) \quad (4.2)$$

$$\boldsymbol{\beta} \sim N_k(\mathbf{0}, \boldsymbol{\Sigma}_\beta) \quad (4.3)$$

$$\phi_i \sim U(-1, 1) \quad i \in \{T, S, ST, TS\} \quad (4.4)$$

$$\tau = \frac{1}{\sigma^2} \sim \text{Gamma}\left(\frac{d_0}{2}, \frac{d_0 s_0}{2}\right) \quad (4.5)$$

$$m_T \sim U_D(\mathbf{m}_{Tred}) \quad (4.6)$$

$$\gamma \sim U(\gamma_{\min}, \gamma_{\max}) \quad (4.7)$$

$$m_S \sim U_D(\mathbf{m}_{Sred}) \quad (4.8)$$

$$\lambda \sim U(\lambda_{\min}, \lambda_{\max}) \quad (4.9)$$

siendo (4.2)-(4.6) y (4.8) las distribuciones presentadas en el capítulo 3. Las distribuciones correspondientes a γ y λ , que hasta ahora se consideraban discretas en una red de valores, son en este capítulo, distribuciones continuas uniformes, con:

$$0 \leq \gamma_{\min} \leq \gamma_{\max} \leq 1 \quad 0 \leq \lambda_{\min} \leq \lambda_{\max} \leq 1$$

4.2.2 Distribución a posteriori

Sea $\boldsymbol{\theta} = (\boldsymbol{\alpha}', \boldsymbol{\beta}', \boldsymbol{\phi}', \tau, m_T, \gamma, m_S, \lambda)'$ el vector de parámetros del modelo con $\boldsymbol{\phi} = (\phi_T, \phi_S, \phi_{ST}, \phi_{TS})'$ tal y como se ha considerado anteriormente. En este caso la distribución a posteriori de los parámetros viene dada por:

$$[\boldsymbol{\theta} | \mathbf{Y}, \mathbf{Z}, \mathbf{X}] \propto$$

$$\propto [Y_1 | \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\phi}, \tau, m_T, \gamma, m_S, \lambda, Y_0, \mathbf{Z}, \mathbf{X}] [\boldsymbol{\alpha}] [\boldsymbol{\beta}] [\boldsymbol{\phi}] [\tau] [m_T] [\gamma] [m_S] [\lambda] \propto$$

$$\begin{aligned}
 & \propto \tau^{\frac{(n-n_0)}{2}} \exp\left[-\frac{\tau}{2} \boldsymbol{\varepsilon}_1'(\boldsymbol{\theta}) \boldsymbol{\varepsilon}_1(\boldsymbol{\theta})\right] \mathbf{x} \\
 & \times \exp\left[-\frac{1}{2} \boldsymbol{\alpha}' \boldsymbol{\Sigma}_\alpha^{-1} \boldsymbol{\alpha}\right] \exp\left[-\frac{1}{2} \boldsymbol{\beta}' \boldsymbol{\Sigma}_\beta^{-1} \boldsymbol{\beta}\right] \prod_{i \in \{T, S, ST, TS\}} I_{(-1,1)}(\phi_i) \mathbf{x} \\
 & \times \tau^{\frac{d_0-1}{2}} \exp\left[-\frac{d_0 s_0}{2} \tau\right] I_{(0,\infty)}(\tau) I_{m_{T_{\text{red}}}}(m_T) I_{(\gamma_{\min}, \gamma_{\max})}(\gamma) I_{m_{S_{\text{red}}}}(m_S) I_{(\lambda_{\min}, \lambda_{\max})}(\lambda) \quad (4.10)
 \end{aligned}$$

4.3 Estimación y validación del modelo

La estimación de los parámetros se realiza a partir de las distribuciones marginales de la distribución a posteriori (4.10), y siendo en este caso, de nuevo, analíticamente intratable, se recurre a los métodos MCMC para calcularlas.

Para obtener una muestra de la distribución a posteriori, análogamente, se utiliza un algoritmo basado en el *Gibbs sampling*. Las distribuciones completamente condicionadas de (4.10) se especifican a continuación.

4.3.1 Distribuciones completamente condicionadas

Las distribuciones completamente condicionadas correspondientes a los parámetros $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, $\boldsymbol{\phi}$ y τ son las dadas en los teoremas 3.1-3.4, respectivamente. Sin embargo, a diferencia del algoritmo de estimación empleado en el capítulo anterior, en este caso el proceso de muestreo de los parámetros de vecindad m_T y m_S y de sus ponderaciones γ y λ se realiza por separado, debido a la diferente naturaleza de cada bloque y para facilitar la implementación del *Gibbs sampling*. A continuación se detallan las distribuciones completamente condicionadas de ambos bloques.

Teorema 4.1

La distribución completamente condicionada de:

$$(m_T, m_S) \mid \alpha, \beta, \phi, \tau, \gamma, \lambda, \mathbf{Y}, \mathbf{Z}, \mathbf{X}$$

es una distribución discreta con soporte $\mathbf{m}_{Tred} \times \mathbf{m}_{Sred}$ y función de probabilidad dada por:

$$P(m_T = m_{Tred,a}, m_S = m_{Sred,c} \mid \alpha, \beta, \phi, \tau, \gamma, \lambda, \mathbf{Y}, \mathbf{Z}, \mathbf{X}) \propto \exp \left[-\frac{\tau}{2} \varepsilon_1'(\alpha, \beta, \phi, m_{Tred,a}, \gamma, m_{Sred,c}, \lambda) \varepsilon_1(\alpha, \beta, \phi, m_{Tred,a}, \gamma, m_{Sred,c}, \lambda) \right]$$

$$a = 1, \dots, t \quad c = 1, \dots, s$$

Demostración:

Inmediata de (4.10).

La distribución completamente condicionada de los parámetros (γ, λ) ; $(\gamma, \lambda) \mid \alpha, \beta, \phi, \tau, m_T, m_S, \mathbf{Y}, \mathbf{Z}, \mathbf{X}$ es una distribución continua que tiene por soporte $(\gamma_{min}, \gamma_{max}) \times (\lambda_{min}, \lambda_{max})$ y función de densidad proporcional a:

$$\exp \left[-\frac{\tau}{2} \varepsilon_1'(\alpha, \beta, \phi, m_{Tred,a}, \gamma, m_{Sred,c}, \lambda) \varepsilon_1(\alpha, \beta, \phi, m_{Tred,a}, \gamma, m_{Sred,c}, \lambda) \right]$$

$$I_{(\gamma_{min}, \gamma_{max}) \times (\lambda_{min}, \lambda_{max})}(\gamma, \lambda)$$

Esta distribución no es estándar y utilizamos un paso de Hastings-Metropolis para obtener una muestra de la misma, tomando como densidad de transición la distribución uniforme en $(\gamma_{min}, \gamma_{max}) \times (\lambda_{min}, \lambda_{max})$. Este procedimiento se detalla en el algoritmo de estimación.

4.3.2 Algoritmo de estimación

Una vez que las distribuciones completamente condicionadas han sido obtenidas, el algoritmo que se emplea, es similar al descrito en 3.3.2. A continuación se explicitan únicamente los cambios que aparecen, debidos al nuevo tratamiento de los parámetros m_T , m_S , γ y λ .

En el paso 1, el subpunto 5 queda desglosado en dos, 5.1 y 5.2, ya que los parámetros de vecindad m_T y m_S aunque siguen tomando valores en una red discreta, esta red se va a ampliar notablemente y con el fin de agilizar el proceso de computación, especialmente lento en este caso, se lleva a cabo un paso de Hasting-Metropolis.

5.1) Extraer $(m_T^{(i)}, m_S^{(i)})$ de

$(m_T, m_S) \mid \boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\phi}^{(i)}, \tau^{(i)}, \gamma^{(i-1)}, \lambda^{(i-1)}, \mathbf{Y}, \mathbf{Z}, \mathbf{X}$ que es una distribución discreta cuyo soporte es $m_{T_{red}} \times m_{S_{red}}$ y su distribución de probabilidad viene dada por:

$$P(m_T = m_{T_{red},a}, m_S = m_{S_{red},c}) \propto \exp \left\{ -\frac{\tau^{(i)}}{2} \boldsymbol{\varepsilon}_1 \left(\boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\phi}^{(i)}, m_{T_{net},a}, \gamma^{(i-1)}, m_{S_{net},q}, \lambda^{(i-1)} \right) \boldsymbol{\varepsilon}_1 \left(\boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\phi}^{(i)}, m_{T_{net},p}, \gamma^{(i-1)}, m_{S_{net},e}, \lambda^{(i-1)} \right) \right\}$$

$$a = 1, \dots, t \quad c = 1, \dots, s$$

Se aplica un paso de Hastings-Metropolis utilizando como distribución de salto la distribución a priori $U_D(\mathbf{m}_{T_{red}} \times \mathbf{m}_{S_{red}})$. En este caso se extrae (m_T^*, m_S^*) de dicha distribución y $u \sim U(0,1)$.

$$\text{Se toma } (m_T^{(i)}, m_S^{(i)}) = \begin{cases} (m_T^*, m_S^*) & \text{si } u \leq \min\{1, \alpha\} \\ (m_T^{(i-1)}, m_S^{(i-1)}) & \text{si } u > \min\{1, \alpha\} \end{cases}$$

$$\text{donde } \alpha = \frac{P(m_T = m_T^*, m_S = m_S^*)}{P(m_T = m_T^{(i-1)}, m_S = m_S^{(i-1)})}$$

5.2) Extraer $(\gamma^{(i)}, \lambda^{(i)})$ de :

$(\gamma, \lambda) | \alpha^{(i)}, \beta^{(i)}, \phi^{(i)}, \tau^{(i)}, m_T^{(i)}, m_S^{(i)}, \mathbf{Y}, \mathbf{Z}, \mathbf{X}$ mediante un paso de Hastings-Metropolis utilizando como distribución de transición la distribución uniforme en $(\gamma_{\min}, \gamma_{\max}) \times (\lambda_{\min}, \lambda_{\max})$. Para ello se simulan (λ^*, γ^*) de la distribución uniforme anterior y $u \sim U(0, 1)$.

$$\text{Se toma } (\gamma^{(i)}, \lambda^{(i)}) = \begin{cases} (\gamma^*, \lambda^*) & \text{si } u \leq \min\{1, \alpha\} \\ (\gamma^{(i-1)}, \lambda^{(i-1)}) & \text{si } u > \min\{1, \alpha\} \end{cases}$$

$$\text{donde } \alpha = \frac{g(\gamma^*, \lambda^*)}{g(\gamma^{(i-1)}, \lambda^{(i-1)})} \text{ y siendo } g(\gamma, \lambda) =$$

$$= \exp \left\{ -\frac{\tau^{(i)}}{2} \varepsilon_1(\alpha^{(i)}, \beta^{(i)}, \phi^{(i)}, m_T^{(i)}, \gamma, m_S^{(i)}, \lambda) \varepsilon_1(\alpha^{(i)}, \beta^{(i)}, \phi^{(i)}, m_T^{(i)}, \gamma, m_S^{(i)}, \lambda) \right\}$$

Tras la aplicación del algoritmo se obtiene una muestra de la distribución (4.10):

$$\{\theta^{(j)}; j=1, \dots, n_{\text{muestra}}\} = \{(\alpha^{(j)}, \beta^{(j)}, \phi^{(j)}, \tau^{(j)}, m_T^{(j)}, \gamma^{(j)}, m_S^{(j)}, \lambda^{(j)}); j=1, \dots, n_{\text{muestra}}\} \quad (4.11)$$

a partir de la cual se realizan inferencias sobre los parámetros del modelo (4.1).

Una vez concluido el proceso de estimación, se procede a la validación del modelo tanto desde el punto de vista intramuestral, valorando la bondad de ajuste a los datos, como de forma extramuestral para examinar el comportamiento predictivo del modelo. Con tal fin, se lleva a cabo un estudio de los residuos de forma gráfica y numérica de modo que puedan detectarse posibles comportamientos sistemáticos de los errores estandarizados, utilizando sus distribuciones a posteriori. Para el análisis del funcionamiento del modelo a nivel predictivo, se realiza de nuevo un procedimiento ordinario de validación y otro de tipo *rolling* y se utilizan criterios que, al mismo tiempo, nos permiten comparar con modelos simplificados. Para todo lo mencionado, se utilizan los mismos procedimientos que en el capítulo 3, descritos con detalle en el apartado 3.4

4.4 Resultados

La metodología expuesta en las secciones anteriores se aplica a la base de datos presentada en 2.6.1, reseñándose únicamente, las modificaciones que aparecen sobre lo ya mostrado.

4.4.1 Estimación del modelo

En el proceso de estimación del modelo, la red de valores para el número de vecinos espaciales y temporales ha sido ampliada y pasa a ser $m_{Tred} = m_{Sred} = \{0, 1, 2, 3, \dots, 48, 49, 50\}$.

El algoritmo explicado en el apartado anterior se ejecuta durante 10000 iteraciones. Las Figuras 4.1 a 4.7 muestran las cadenas de cada uno de los parámetros del modelo y sus correspondientes correlogramas.

Resultados

En casi todas ellas, tal y como se puede observar, se alcanza la convergencia tras aproximadamente 1000 iteraciones y la autocorrelación de las series resultantes es prácticamente inexistente a partir de los 10 retardos. De forma más pormenorizada, se puede advertir que en el caso de las cadenas correspondientes a los coeficientes γ y λ (Figura 4.7) el decaimiento se produce de forma más lenta, pero de modo claro va reduciéndose la dependencia. De nuevo, la cadena correspondiente al coeficiente β_{lsup} (coeficiente hedónico correspondiente al logaritmo de la superficie, ver Figura 4.2) muestra un decrecimiento más pausado. Con el fin de estudiar de forma más precisa su comportamiento, se ha realizado un análisis de sensibilidad de las inferencias sobre este parámetro en relación con la longitud de salto, que no ha mostrado diferencias con lo reseñado anteriormente.

Se toma, por tanto, una muestra cada 10 pasos para todos los parámetros estimados.

El error estándar de simulación no ha superado el 0.005 excepto en el caso de la constante (0.0442), m_T (0.1531) y m_S (0.1031).

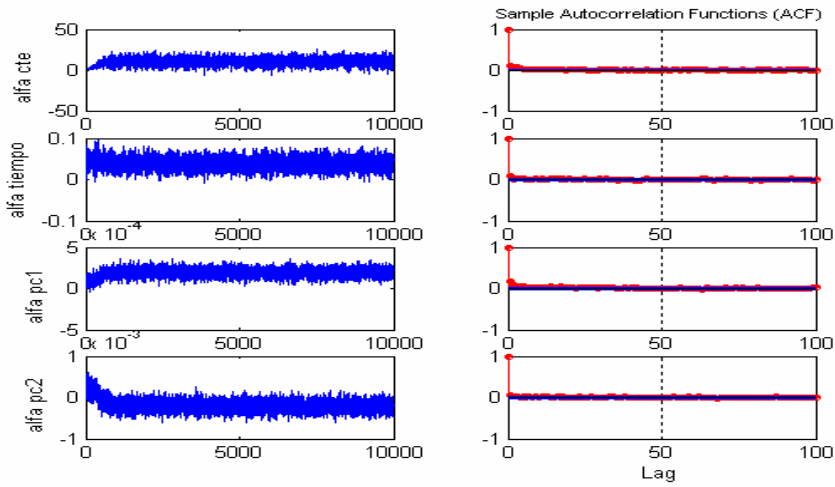


Figura 4.1: Iteraciones del algoritmo de Gibbs y funciones de autocorrelación para las cadenas correspondientes a los coeficientes α de la constante, el indicador de tendencia temporal, PC1 y PC2, respectivamente

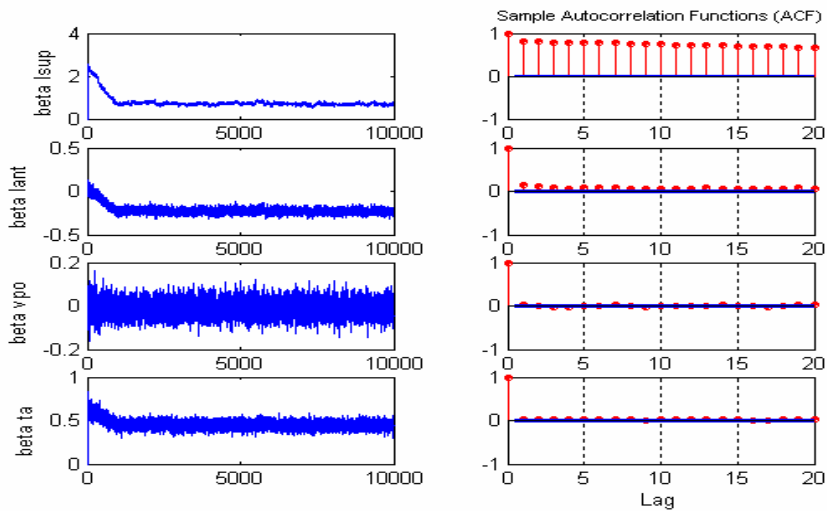


Figura 4.2: Iteraciones del algoritmo de Gibbs y funciones de autocorrelación para las cadenas correspondientes a los coeficientes β de log superficie, log antigüedad, VPO y tipo de adquisición, respectivamente

Resultados

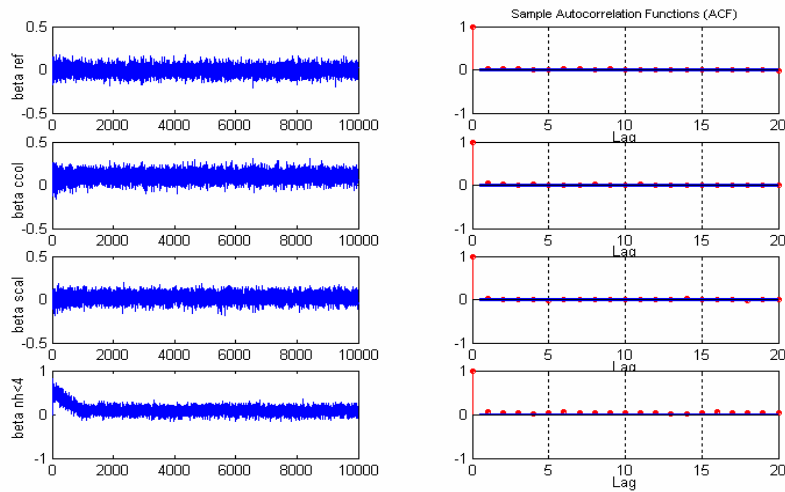


Figura 4.3: Iteraciones del algoritmo de Gibbs y funciones de autocorrelación para las cadenas correspondientes a los coeficientes β de refrigeración, calefacción colectiva, sin calefacción y n° habitaciones <4

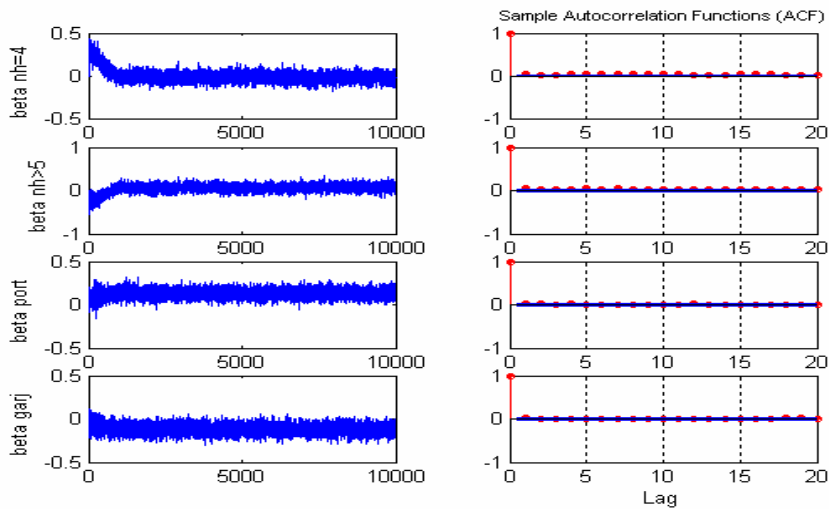


Figura 4.4: Iteraciones del algoritmo de Gibbs y funciones de autocorrelación para las cadenas correspondientes a los coeficientes β n° de habitaciones = 4, n° de habitaciones > 5, portero y garaje, respectivamente

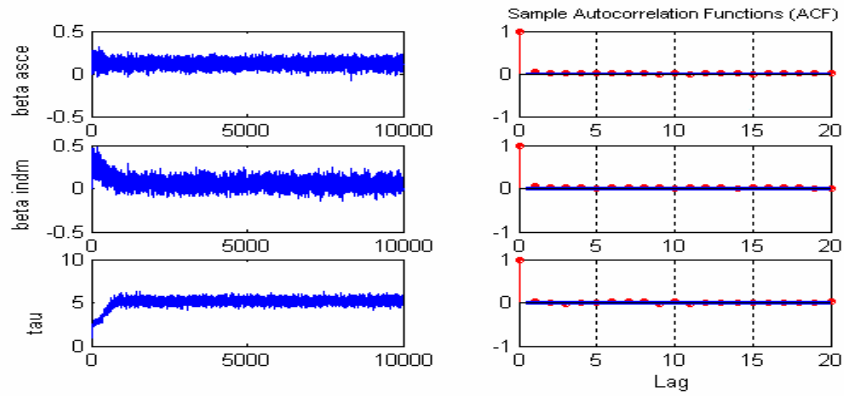


Figura 4.5: Iteraciones del algoritmo de Gibbs y funciones de autocorrelación para las cadenas correspondientes a los coeficientes β de ascensor e indicador de datos ausentes, y la última del parámetro τ

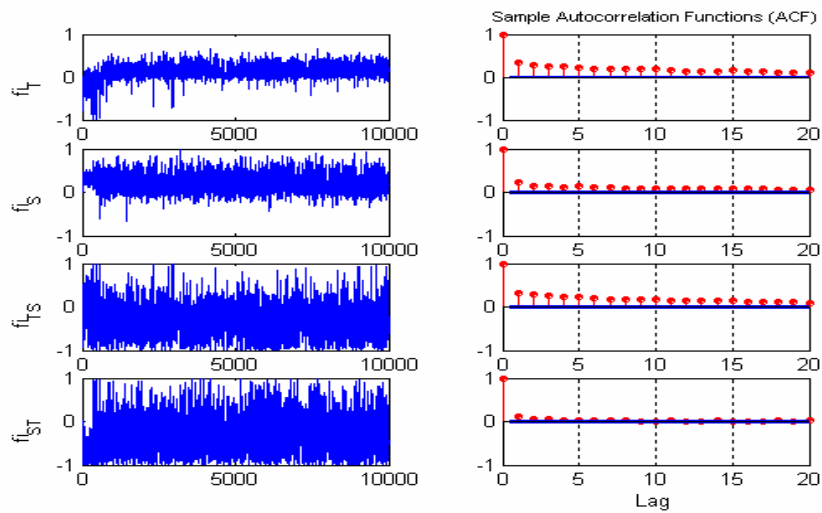


Figura 4.6: Iteraciones del algoritmo de Gibbs y funciones de autocorrelación para las cadenas correspondientes a los coeficientes ϕ_T , ϕ_S , ϕ_{TS} y ϕ_{ST}

Resultados

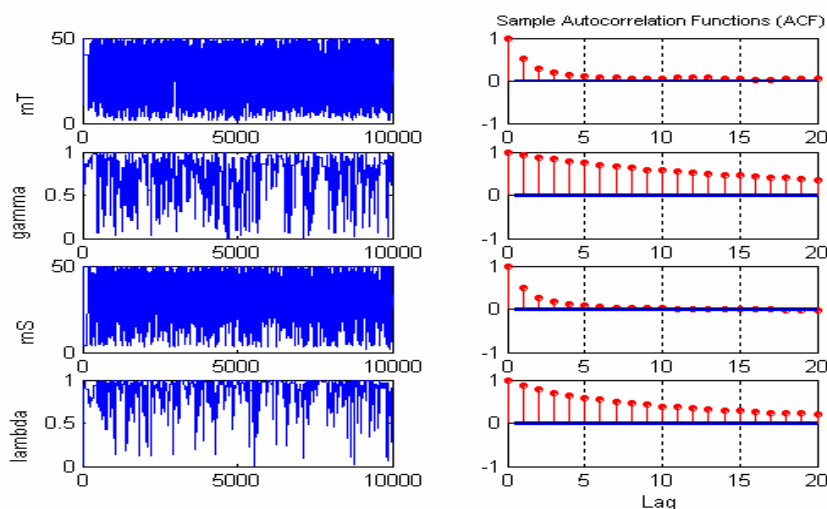


Figura 4.7: Iteraciones del algoritmo de Gibbs y funciones de autocorrelación para las cadenas correspondientes a los coeficientes m_T , γ , m_S y λ

Las estimaciones puntuales con sus respectivos límites de credibilidad bayesianos al 95% contruidos a partir de la distribución a posteriori, se muestran en la Tabla 4.1. En las Figuras 4.8 a 4.13 se presentan sus respectivas densidades a posteriori calculadas con la muestra producida tras la ejecución del algoritmo 4.3.2.

Los valores, los signos y el grado de significación de los coeficientes hedónicos, coinciden con los resultados obtenidos en los capítulos 2 y 3 (ver Tablas 2.4 y 3.1), dejando claro la trascendencia de algunas variables sobre el precio de los inmuebles que conforman la base de datos. Análogamente, el hecho de que los coeficientes ϕ_{ST} y ϕ_{TS} no resulten significativos, insiste en la dificultad de capturar los efectos de interacción entre la parte espacial y temporal, y concuerda con lo aparecido en el capítulo 3. En el caso de los coeficientes autorregresivos sin interacción ϕ_T y ϕ_S , sólo aparecen como

significativos al nivel del 90%, siendo notoria la falta de normalidad en las densidades a posteriori de estos parámetros (ver Figura 4.13).

En cuanto a las estimaciones de los parámetros de vecindad, es importante destacar que la ampliación de la red para el número de vecinos temporales y espaciales, m_T y m_S , ha contribuido a aumentar la incertidumbre en su estimación apareciendo bandas todavía más dilatadas, incluyendo dentro de éstas más valores de rango alto (ver Figura 4.14). Los valores altos ponen de manifiesto la tendencia del modelo a utilizar información procedente de un mayor número de vecinos, no resultando satisfactorio el resultado con un pequeño núcleo de vecinos influyentes. No obstante, hay que hacer notar que, aunque en el caso de m_S quedan prácticamente descartados los valores más bajos de la distribución, esto no ocurre en el caso de m_T donde la probabilidad se encuentra repartida por todas las zonas de la distribución, incluso en valores próximos a cero. Para las ponderaciones γ y λ , las distribuciones a posteriori presentan un alejamiento notable de la normalidad, estando claramente concentradas en los valores altos de su soporte (ver Figura 4.14), aunque también para λ se aprecia una concentración mayor de la distribución cerca de 1, algo que ocurre de forma más suave y por tanto no tan concentrada en el caso de γ . Estos elevados valores próximos a uno, tienden a asignar el mismo peso a la información que proporcionan todos los vecinos considerados espacialmente influyentes, aun cuando se encuentren alejados de la transacción en estudio.

Tabla 4.1: Estimaciones de los parámetros (en negrita los significativos para un nivel de credibilidad del 95% y con⁺ al 90%)

VARIABLES	STAR			STARs			hedónico		
	C 2.5	Mediana	C 97.5	C 2.5	Mediana	C 97.5	C 2.5	Mediana	C 97.5
Constante	2.4058	10.7714	19.0973	5.9615	7.7084	8.8993	8.2443	8.7743	9.3095
Tiempo	0.0135	0.0382	0.0618	0.0191	0.0317	0.0449	0.0231	0.0369	0.0507
PC1	0.0001	0.0002	0.0003	0.0002	0.0002	0.0003	0.0002	0.0002	0.0003
PC2	-0.0004	-0.0002 ⁺	0.0000	-0.0004	-0.0003	-0.0001	-0.0005	-0.0003	-0.0001
Log (superficie)	0.5994	0.7063	0.8253	0.6471	0.7229	0.8157	0.6004	0.7168	0.8357
Log (antigüedad)	-0.2790	-0.2265	-0.1758	-0.2684	-0.2276	-0.1850	-0.2764	-0.2269	-0.1753
VPO	-0.0786	-0.0123	0.0543				-0.0825	-0.0125	0.0500
Tipo de adquisición	0.3679	0.4456	0.5274	0.3826	0.4497	0.5104	0.3729	0.4543	0.5311
Refrigeración	-0.1015	-0.0076	0.0841				-0.0966	0.0048	0.0985
Calefacción colectiva	-0.0047	0.0989 ⁺	0.2158	0.0045	0.0849	0.1588	-0.0081	0.0962 ⁺	0.2107
Sin calefacción	-0.0679	0.0250	0.1131				-0.0848	0.0129	0.1075
N° habitación es <4	-0.0578	0.0869	0.2168				-0.0425	0.0891	0.2133
N° habitaciones =4	-0.1064	-0.0161	0.0751				-0.1125	-0.0135	0.0738
N° habitaciones >5	-0.0530	0.0641	0.1954				-0.0345	0.0800	0.2033
Portero	0.0390	0.1332	0.2269	0.0579	0.1327	0.1987	0.0396	0.1313	0.2178
Garaje	-0.2120	-0.1120	-0.0210	-0.2036	-0.1240	-0.0415	-0.2272	-0.1237	-0.0302
Ascensor	0.0369	0.1190	0.1953	0.0618	0.1248	0.1899	0.0378	0.1261	0.1921
Indicador missing	-0.0541	0.0531	0.1578				-0.0464	0.0490	0.1580
ϕ_T	-0.0466	0.1482 ⁺	0.4091	-0.0032	0.1211 ⁺	0.3138			
ϕ_S	-0.0969	0.2055	0.5954						
ϕ_{ST}	-0.9178	-0.3053	0.3149						
ϕ_{TS}	-0.9184	-0.2800	0.4933						
τ	4.6686	5.1865	5.7675	4.7767	5.1870	5.6387	4.6043	5.1571	5.7319
m_T	4	27	49	4	26	48			
γ	0.1230	0.8052	0.9896	0.1912	0.6801	0.9519			
m_S	9	31	49	4	26	48			
λ	0.5617	0.9445	0.9965	0.0602	0.5061	0.9498			

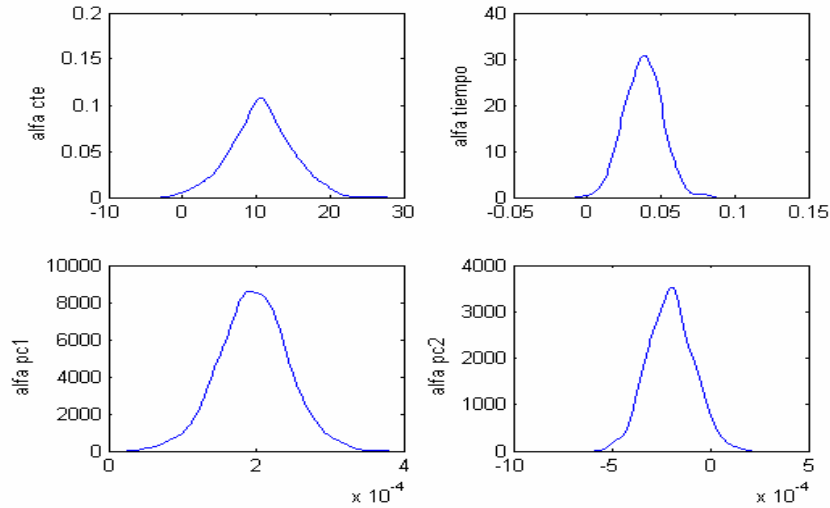


Figura 4.8: Densidades a posteriori de los coeficientes α . De izquierda a derecha y de arriba abajo las correspondientes a la constante, el indicador de tiempo, la componente PC1 y la componente PC2, respectivamente

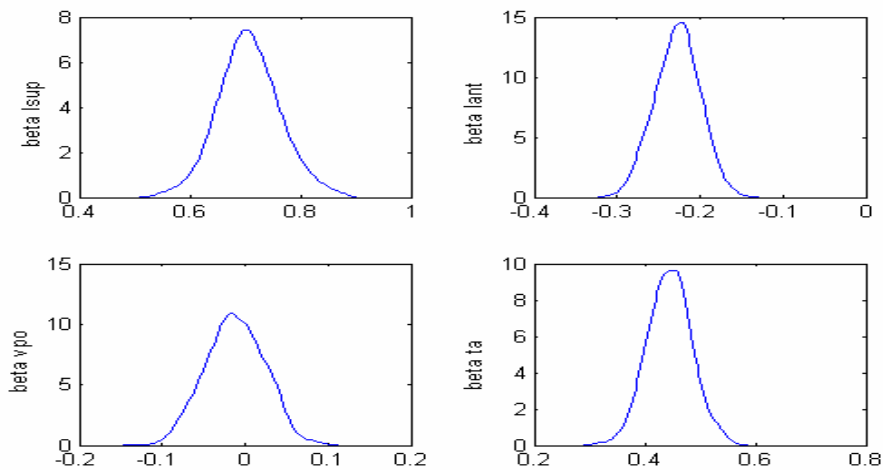


Figura 4.9: De izquierda a derecha y de arriba abajo densidades a posteriori de los coeficientes β correspondientes al logaritmo de superficie, logaritmo de antigüedad, VPO y tipo de adquisición, respectivamente

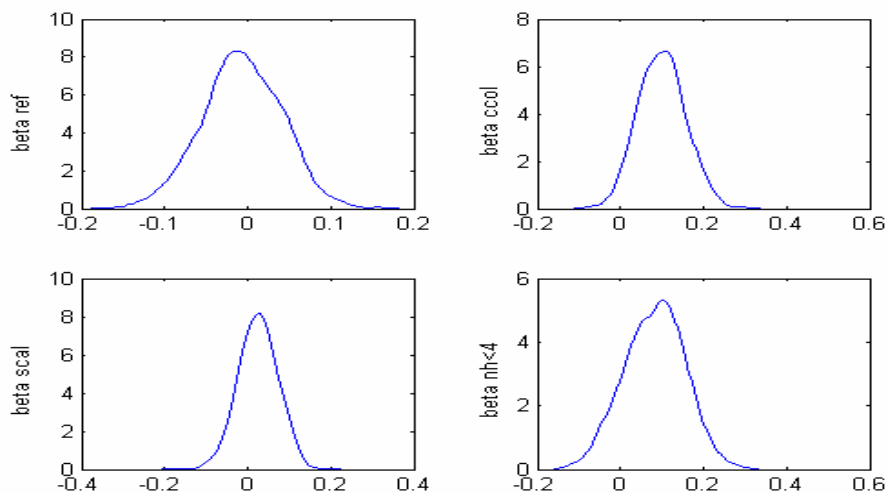


Figura 4.10: De izquierda a derecha y de arriba abajo densidades a posteriori de los coeficientes β correspondientes a las variables refrigeración, calefacción colectiva, sin calefacción y nº de habitaciones < 4, respectivamente

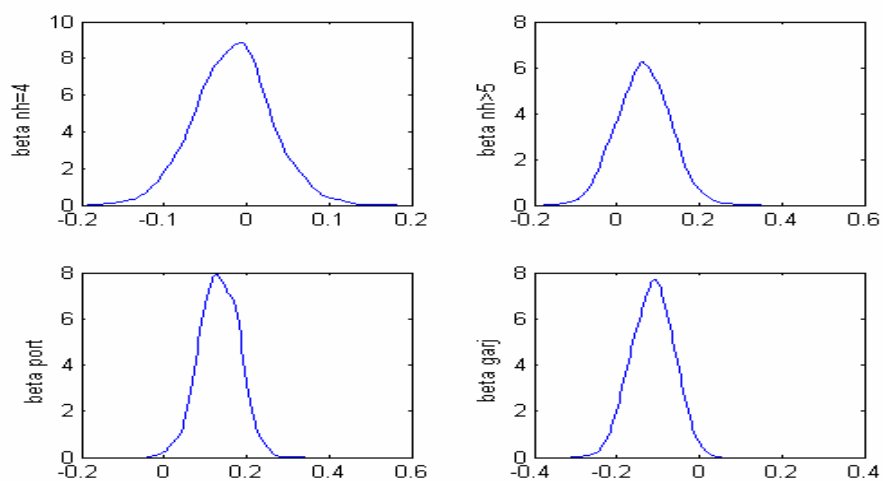


Figura 4.11: De izquierda a derecha y de arriba abajo densidades a posteriori de los coeficientes β correspondientes a las variables nº de habitaciones = 4, nº de habitaciones > 5, portero y garaje, respectivamente

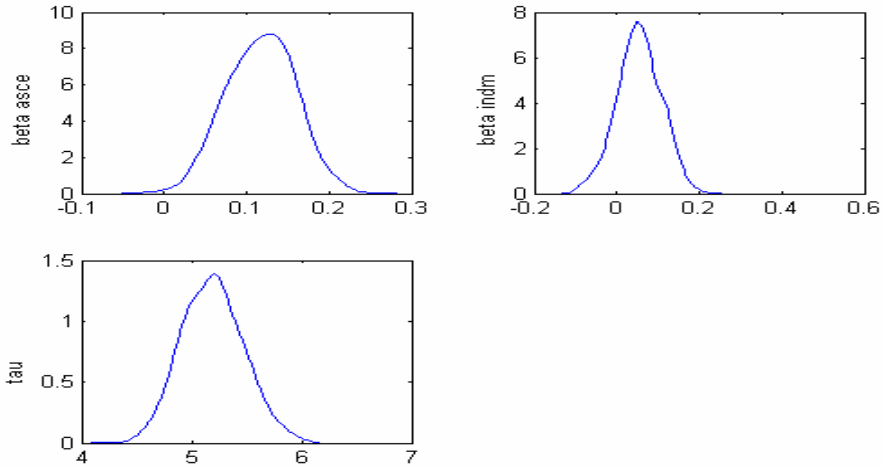


Figura 4.12: De izquierda a derecha y de arriba abajo densidades a posteriori de los coeficientes β correspondientes a las variables ascensor, indicador de datos ausentes y del parámetro τ , respectivamente

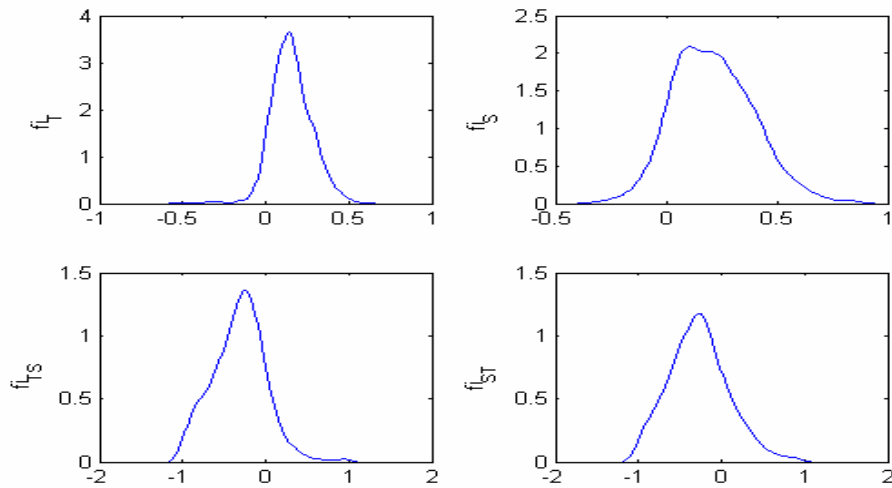


Figura 4.13: De izquierda a derecha y de arriba abajo densidades a posteriori de los coeficientes ϕ_T , ϕ_S , ϕ_{ST} y ϕ_{TS} , respectivamente

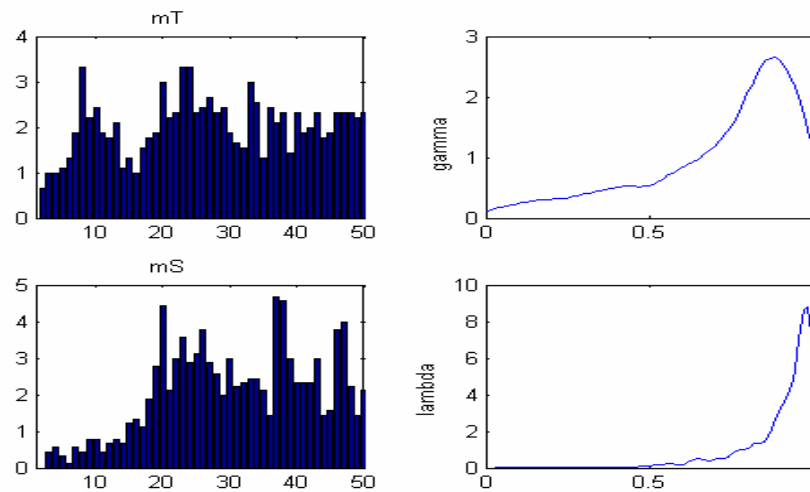


Figura 4.14: A la izquierda gráficos de probabilidad de m_T , y m_S y a la derecha densidades a posteriori de γ y λ

4.4.2 Validación del modelo

La validación del modelo se estudia desde el punto de vista intra y extramuestral y se presenta en las Tablas 4.2 y 4.3 y en las Figuras 4.15 a 4.22.

Las Figuras 4.15 a 4.19 y la Tabla 4.2 muestran de forma reiterativa, que no existen comportamientos sistemáticos en los errores intra ni extramuestrales y que no se evidencia la presencia de autocorrelaciones significativas. De nuevo se observa la existencia de asimetría a la izquierda, así como una elevada leptocurtosis causadas por la presencia de atípicos negativos.

Las Figuras 4.20 y 4.21 informan de una buena capacidad predictiva extramuestral del modelo, que tanto para el precio, como para el logaritmo del precio de los inmuebles capturan la marcada asimetría positiva de los datos.

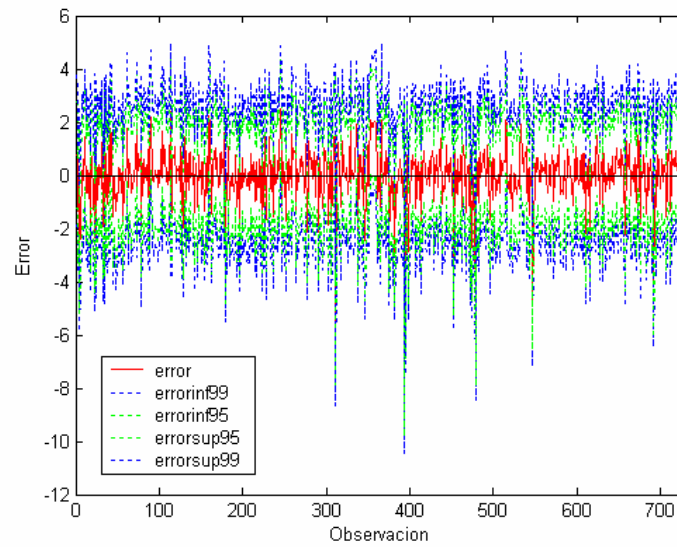


Figura 4.15: Errores intramuestrales del modelo STAR (en líneas rayadas en verde y en azul las bandas de credibilidad al 95% y 99%, respectivamente)

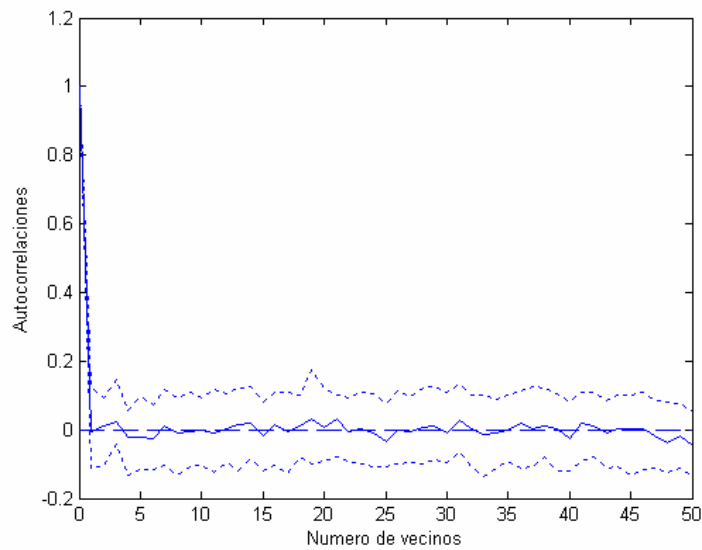


Figura 4.16: Autocorrelaciones espaciales residuales intramuestrales del modelo STAR (en líneas rayadas en rojo las bandas de credibilidad al 95%)

Tabla 4.2: Distribución de los errores tipificados intramuestrales del modelo STAR (como estimador de los parámetros su media a posteriori)

ERRORES TIPIFICADOS Modelo STAR	
Mínimo	-7.8043
1%	-3.0126
5%	-1.7040
10%	-1.0985
25%	-0.3982
50%	0.1284
75%	0.5929
90%	0.9679
95%	1.3493
99%	2.0313
Máximo	2.5130
Media	0.0156
Mediana e 	0.5115
Asimetría	-1.7733
Curtosis	12.3742

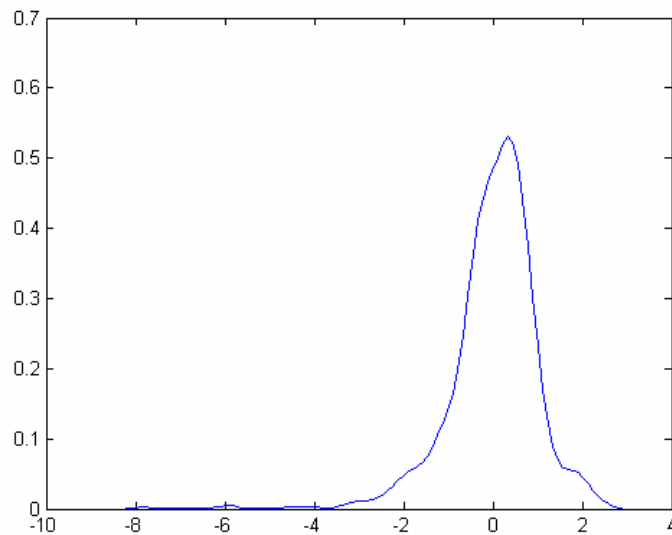


Figura 4.17: Estimación de la densidad de los errores tipificados intramuestrales calculados tomando como estimador de los parámetros su media a posteriori

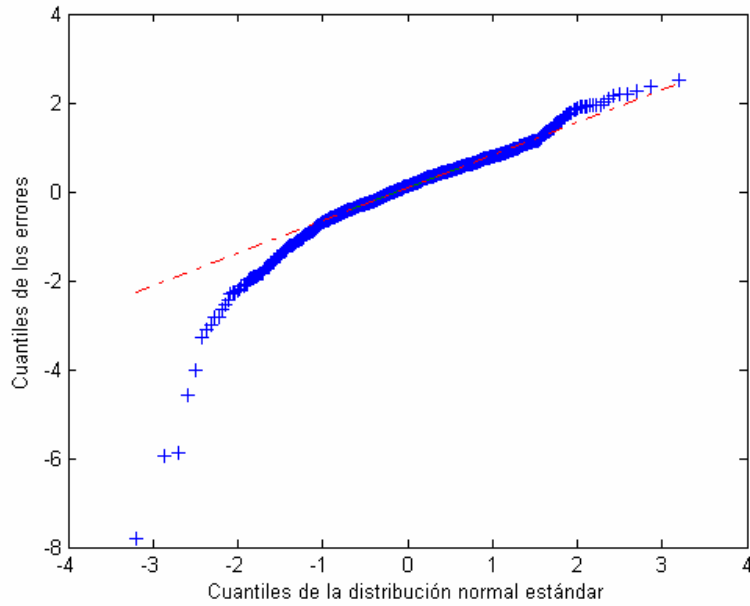


Figura 4.18: Q-Q plot de los errores tipificados intramuestrales calculados tomando como estimador de los parámetros su media a posteriori

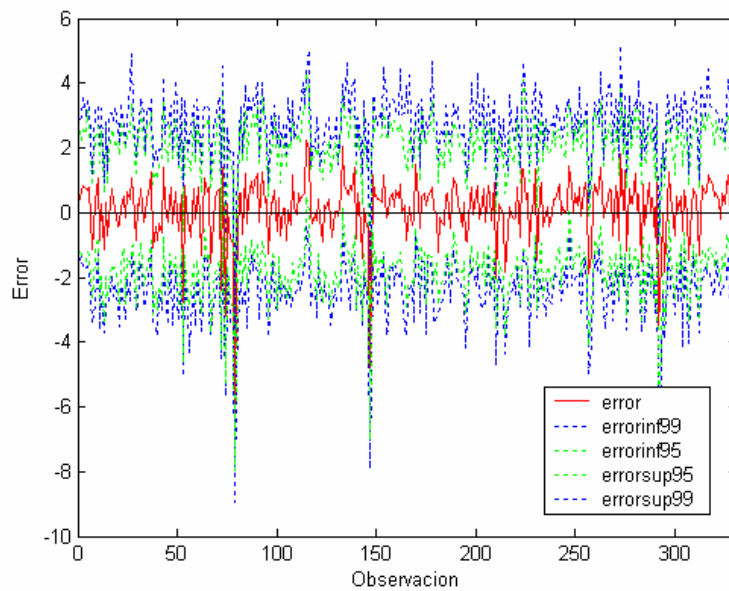


Figura 4.19: Errores tipificados extramuestrales (*rolling*) del modelo STAR

Resultados

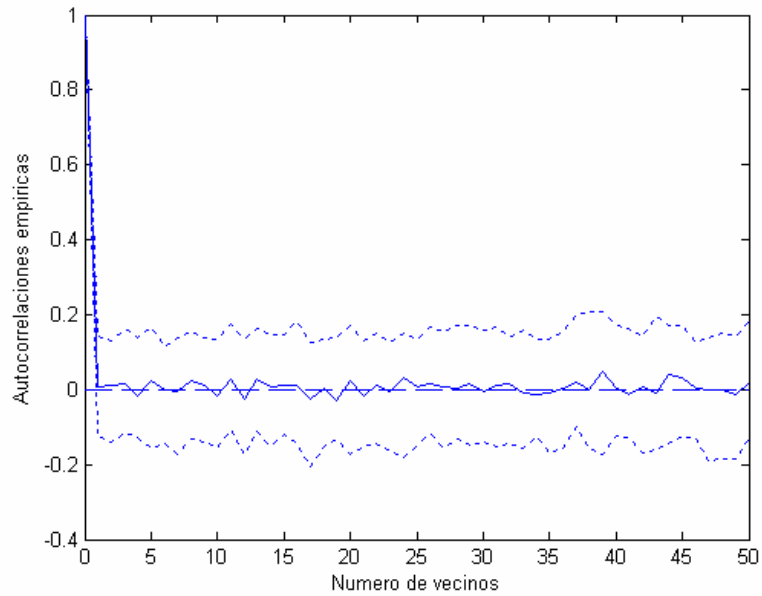


Figura 4.20: Autocorrelaciones espaciales residuales extramuestrales (rolling) del modelo STAR (en líneas rayadas las bandas de credibilidad al 95%)

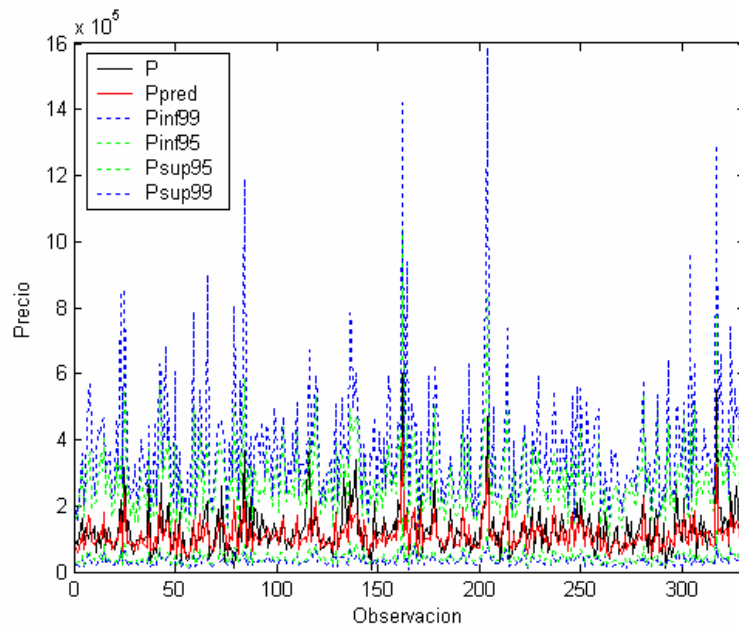


Figura 4.21: Predicciones extramuestrales (rolling) del precio para el modelo STAR

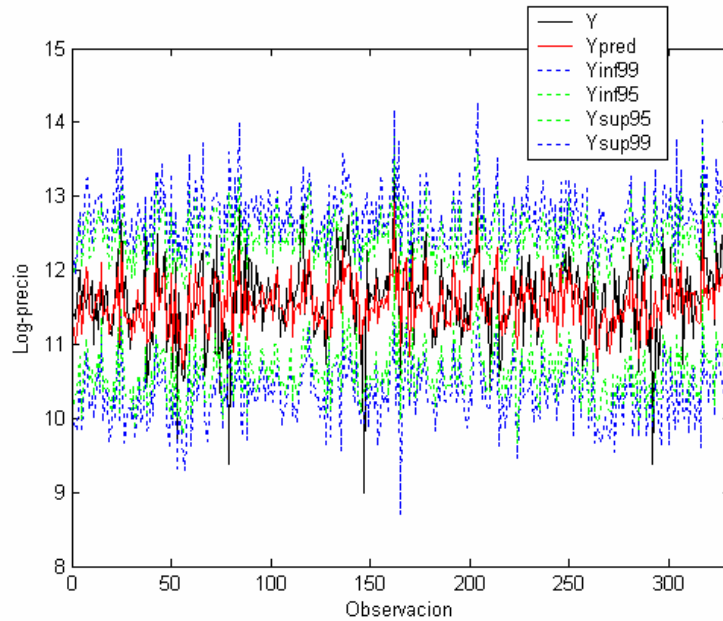


Figura 4.22: Predicciones extramuestrales (*rolling*) del logaritmo del precio para el modelo STAR

Por último, los criterios que nos informan sobre la bondad del ajuste, (ver Tabla 4.3) manifiestan que los cubrimientos al 95%, tanto intramuestral como extramuestralmente, son correctos, y solamente el cubrimiento al 99% en el caso intramuestral se desvía significativamente del valor teórico con un pvalor de 0.034. Por su parte, los valores del coeficiente de determinación no difieren significativamente de los calculados en los modelos del capítulo anterior, evaluándose en torno a un 40-45% la capacidad explicativa del modelo.

Tabla 4.3: Validación del modelo STAR

CRITERIOS	Intramuestrales	Extramuestrales (<i>rolling</i>)
R^2	0.4539	0.4148
CUB95	95.62	96.07
CUB99	98.22*	98.79

*Existen diferencias significativas con el resultado teórico al 5%

4.4.3 Simplificación del modelo

El proceso de simplificación que se realiza en esta sección es similar al que se efectuó en el capítulo anterior obteniendo un modelo reducido, que incorpora solamente aquellas variables que fueron significativas al nivel del 10% en el proceso de estimación (ver Tabla 4.1). Del mismo modo, lo denominaremos STARs y a excepción del coeficiente ϕ_s , coincide con el modelo simplificado considerado en el capítulo 3. Por las razones ya mencionadas, se incluye en el proceso de comparación el modelo hedónico tradicional.

En la Tabla 4.4 se pueden examinar los resultados obtenidos en el proceso de comparación.

Tabla 4.4: Criterios de comparación para los modelos STAR, STARs y hedónico.

CRITERIO	MODELO		
	STAR	STARs	hedónico
DIC	895.98	886.85	895.94
\bar{D}	873.86	873.44	877.35
pd	22.12	13.41	18.59
DIC*	882.13	877.52	882.96
P_D^*	8.27	4.08	5.61
LPRED	-178.52	-173.75	-179.07
PBF(Modelo STARs)	0.0085	1.0000	0.0049
RMSE	0.4220	0.4056	0.4194
MAD	0.3088	0.2987	0.3123
LPRED _{rol}	-195.82	-185.80	-190.53

Es importante hacer notar que, a diferencia de lo sucedido en el capítulo 3 (ver Tabla 3.4), el número efectivo de parámetros p_D toma valores positivos en todos los modelos comparados. Tal y como apuntan Celeux y otros (2006) para casos similares, esto puede ser debido al tratamiento discreto dado a los parámetros γ y λ . Se advierte la clara superioridad del modelo reducido STARs, que muestra mejores puntuaciones en todos los criterios utilizados lo que confirma la mayor adecuación del modelo para describir la evolución espacio-temporal de los precios de las transacciones analizadas.

4.5 Conclusiones

En este capítulo se ha trabajado sobre el modelo STAR presentado al comienzo de la Memoria desde una aproximación bayesiana de forma similar a lo llevado a cabo en el capítulo 3, añadiendo un tratamiento continuo de las ponderaciones λ y γ e incorporando modificaciones al algoritmo de estimación de los parámetros de forma que se mejora la eficiencia computacional cuando el soporte de m_T y m_S es extenso. Aunque la complejidad computacional del proceso de estimación se incrementa, el tratamiento dado en este capítulo aumenta el grado de flexibilidad del modelo mejorando su bondad de ajuste a los datos analizados.

La estimación de los parámetros del modelo STAR, ha puesto de manifiesto, una vez más, cuales son las variables hedónicas que resultan significativas sobre el precio de un inmueble, existiendo una perfecta concordancia con la información aportada por las estimaciones de los capítulos 2 y 3. Con respecto a los coeficientes autorregresivos, solamente aparece como significativo ϕ_T al 10% como ya ocurriera con anterioridad.

Conclusiones

El análisis de los residuos vuelve a mostrar un comportamiento predictivo aceptable del modelo a nivel extramuestral. No obstante, en la validación intramuestral, el modelo no es capaz de capturar la influencia de los atípicos ni la elevada leptocurtosis, poniendo en tela de juicio la bondad de ajuste del modelo a los datos.

Las consecuencias de las modificaciones introducidas en este capítulo, en cuanto a la ampliación de la red de valores para el número de vecinos espaciales y temporales, han quedado reflejadas, de manera contundente, a través de la selección de los valores altos en ambos parámetros, de forma que la estimación ha mostrado como influyentes un número alto de transacciones. De este modo ni geográfica ni temporalmente parece que el papel de vecino, quede constreñido a inmuebles del entrono más próximo a uno dado, añadiendo a través de sus ponderaciones, el interés en dar relevancia a la información proporcionada por un gran número de transacciones.

En el capítulo siguiente se proponen nuevas mejoras a través de métodos robustos de estimación de los parámetros del modelo que debiliten la posible influencia de los atípicos en el proceso de estimación, y mejoren la bondad de ajuste del modelo.

Capítulo 5

Inferencia bayesiana robusta y predicciones retrospectivas

5.1 Introducción

Tras el tratamiento bayesiano llevado a cabo en el capítulo 3 y las mejoras propuestas en el capítulo 4, consistentes en flexibilizar los parámetros de vecindad presentes en el modelo STAR, se plantean nuevas modificaciones con el fin de disminuir el impacto que puedan ejercer en el proceso de estimación aquellas observaciones que se consideren atípicos. Para ello, se utiliza un modelo STAR con efectos de vecindad y errores heterocedásticos que permite llevar a cabo una estimación robusta, a través de la utilización de distribuciones con colas más pesadas. Así mismo, se presentan los resultados de la aplicación a la base de datos, valorando la mejora que han supuesto las novedades implementadas en el capítulo.

Para finalizar, se introduce un método de predicción retrospectiva que permite estimar el precio de una transacción que tuvo lugar en un momento dado del pasado, dentro del periodo de estimación del modelo, pero cuya cuantía se desconoce. Este tipo de predicciones hechas “a pasado” resultan,

desde un punto de vista práctico, de especial utilidad en ámbitos judiciales y periciales. Utilizando dicho método se presenta una aplicación para la construcción retrospectiva de números índice de precios de vivienda con modelos hedónicos.

El capítulo está estructurado como sigue: en la sección 5.2 se presenta el modelo STAR con errores heterocedásticos y su tratamiento completo, como en el capítulo anterior. El punto 5.3 muestra como se lleva a cabo el proceso de estimación y validación del modelo. La sección 5.4 expone los resultados de la aplicación a la base de datos. En el apartado 5.5 se propone una aplicación para calcular predicciones retrospectivas y la construcción de números índice de precios de viviendas. Por último, el punto 5.6 detalla las conclusiones del capítulo.

5.2 Tratamiento heterocedástico del modelo STAR con efectos de vecindad

Con el fin de disminuir la posible influencia ejercida por observaciones atípicas en el proceso de estimación de los parámetros del modelo STAR con efectos de vecindad, se considera en este apartado un modelo con errores heterocedásticos, que viene dado por la expresión:

$$\mathbf{Y} = \mathbf{Z}\boldsymbol{\alpha} + \mathbf{X}\boldsymbol{\beta} + \phi_T\mathbf{T}(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}) + \phi_S\mathbf{S}(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}) + \phi_{ST}\mathbf{ST}(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}) + \phi_{TS}\mathbf{TS}(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}) + \boldsymbol{\varepsilon} \quad (5.1)$$

con $\boldsymbol{\varepsilon} \sim N_n(\mathbf{0}, \sigma^2\mathbf{V})$ donde $\mathbf{V} = \text{diag}(v_1, \dots, v_n)$ ($v_i > 0 \ i = 1, \dots, n \ \sigma^2 > 0$)

Para llevar a cabo el tratamiento bayesiano del mismo, como es habitual, comenzamos especificando la distribución a priori sobre sus parámetros.

5.2.1 Distribución a priori

La distribución a priori viene dada por:

$$\boldsymbol{\alpha} \sim N_q(\mathbf{0}, \boldsymbol{\Sigma}_\alpha) \quad (5.2)$$

$$\boldsymbol{\beta} \sim N_k(\mathbf{0}, \boldsymbol{\Sigma}_\beta) \quad (5.3)$$

$$\phi_i \sim U(-1, 1) \quad i \in \{T, S, ST, TS\} \quad (5.4)$$

$$\tau_i = \frac{1}{v_i} \sim \frac{\chi_r^2}{r} \quad i=1, \dots, n \quad (5.5)$$

$$r \sim \text{Gamma}(m_1, m_2) \quad (5.6)$$

$$\tau = \frac{1}{\sigma^2} \sim \text{Gamma}\left(\frac{d_0}{2}, \frac{d_0 s_0}{2}\right) \quad (5.7)$$

$$\mathbf{m}_T \sim U_D(\mathbf{m}_{Tred}) \quad (5.8)$$

$$\gamma \sim U(\gamma_{min}, \gamma_{max}) \quad (5.9)$$

$$\mathbf{m}_S \sim U_D(\mathbf{m}_{Sred}) \quad (5.10)$$

$$\lambda \sim U(\lambda_{min}, \lambda_{max}) \quad (5.11)$$

siendo (5.2)-(5.4) y (5.7)-(5.11) las distribuciones que fueron presentadas en el capítulo 4. Las distribuciones (5.5) y (5.6) siguen el tratamiento dado en Geweke (1993), LeSage (1999) y Sun y otros (2005) al problema de errores del modelo heterocedasticos, y son difusas si m_1 , m_2 y $d_0 \rightarrow 0$.

Los factores de escala $\{\tau_i ; i=1, \dots, n\}$ modelan la heterocedasticidad condicional de los términos de error del modelo. Observar que tienen un valor medio igual a 1 por lo que la distribución a priori supone que el modelo es homocedástico. Sin embargo, la varianza de dichos factores (condicional al valor de r), es igual a $\frac{2}{r}$ de forma que, cuanto menor (respectivamente mayor) es el valor de r , menor (respectivamente mayor) es la confianza inicial en la

homocedasticidad del modelo. Además, se verifica que $\varepsilon_i|r \sim t_r$ donde t_r denota la distribución t de Student con r grados de libertad, por lo que, cuanto menor es r , mayor es el grado de leptocurtosis de la distribución de los errores y más pesadas son sus colas. Por lo tanto, pequeños valores de r se asocian con una creencia a priori de que existen observaciones con errores anormalmente grandes, en valor absoluto, con respecto a la distribución normal. Este tratamiento de la distribución de los errores posibilita, como se verá más adelante, la disminución de la influencia en el proceso de estimación de los parámetros de aquéllos que son muy grandes (y, por tanto, mal explicados por el modelo), dotándolo de esta manera de una mayor robustez.

5.2.2 Distribución a posteriori

Una vez especificadas las ecuaciones del modelo y la distribución a priori sobre los parámetros, la distribución a posteriori es la que sigue:

$$\begin{aligned}
 & [\boldsymbol{\theta} | \mathbf{Y}, \mathbf{Z}, \mathbf{X}] \propto \\
 & \propto [\mathbf{Y}_1 | \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\phi}, \boldsymbol{\tau}_n, \tau, m_T, \gamma, m_S, \lambda, \mathbf{Y}_0, \mathbf{Z}, \mathbf{X}] [\boldsymbol{\alpha}] [\boldsymbol{\beta}] [\boldsymbol{\phi}] [\boldsymbol{\tau}_n | r] [\tau] [m_T] [\gamma] [m_S] [\lambda] \propto \\
 & \propto \tau^{\frac{(n-n_0)}{2}} |\mathbf{V}|^{-\frac{1}{2}} \exp\left[-\frac{\tau}{2} \boldsymbol{\varepsilon}_1' \mathbf{V}^{-1} \boldsymbol{\varepsilon}_1\right] \exp\left[-\frac{1}{2} \boldsymbol{\alpha}' \boldsymbol{\Sigma}_\alpha^{-1} \boldsymbol{\alpha}\right] \exp\left[-\frac{1}{2} \boldsymbol{\beta}' \boldsymbol{\Sigma}_\beta^{-1} \boldsymbol{\beta}\right] \times \\
 & \times \prod_{i \in \{T, S, ST, TS\}} I_{(-1,1)}(\phi_i) \prod_{i=n_0+1}^n \frac{\tau_i^{\frac{r}{2}-1} r^{r/2} \exp\left[-\frac{r}{2} \tau_i\right]}{2^{\frac{r}{2}} \Gamma\left(\frac{r}{2}\right)} I_{(0,\infty)}(\tau_i) r^{m_1-1} \exp[-m_2 r] I_{(0,\infty)}(r) \times \\
 & \times \tau^{\frac{d_0}{2}-1} \exp\left[-\frac{d_0 s_0}{2} \tau\right] I_{(0,\infty)}(\tau) I_{m_{T_{red}}}(\tau) I_{(\gamma_{min}, \gamma_{max})}(\gamma) I_{m_{S_{red}}}(\gamma) I_{(\lambda_{min}, \lambda_{max})}(\lambda)
 \end{aligned} \tag{5.12}$$

donde $\boldsymbol{\theta} = (\boldsymbol{\alpha}', \boldsymbol{\beta}', \boldsymbol{\phi}', \boldsymbol{\tau}_n, r, \tau, m_T, \gamma, m_S, \lambda)'$ es el vector de parámetros del modelo, $\boldsymbol{\phi} = (\phi_T, \phi_S, \phi_{ST}, \phi_{TS})'$, $\boldsymbol{\tau}_n = (\tau_1, \dots, \tau_n)'$ y $\mathbf{V} = \text{diag}(v_1, \dots, v_n)$.

5.3 Estimación y validación del modelo

Al igual que en el capítulo anterior, la estimación de los parámetros se realiza a partir de las distribuciones marginales de la distribución a posteriori (5.12). Dado que ésta vuelve a ser analíticamente intratable, se recurre a los métodos MCMC para calcularlas.

Para ello utilizamos de nuevo el *Gibbs sampling*, que nos permite obtener una muestra de la distribución a posteriori a partir de la cual se pueden realizar inferencias acerca de los parámetros del modelo. A continuación calculamos las distribuciones completamente condicionadas de (5.12) necesarias para implementar el algoritmo de estimación.

5.3.1 Distribuciones completamente condicionadas

Dichas distribuciones vienen dadas como resultado de los teoremas.

Teorema 5.1

La distribución completamente condicionada del parámetro α viene dada por:

$$\alpha \mid \beta, \phi, \tau_n, r, \tau, m_T, \gamma, m_S, \lambda, \mathbf{Y}, \mathbf{Z}, \mathbf{X} \sim N_q(\mathbf{m}_\alpha, \mathbf{S}_\alpha) \quad (5.13)$$

$$\text{con } \begin{cases} \mathbf{m}_\alpha = \mathbf{S}_\alpha (\tau \mathbf{Z}'_1 \mathbf{V}^{-1} \mathbf{W}_1^\alpha) \\ \mathbf{S}_\alpha^{-1} = \tau \mathbf{Z}'_1 \mathbf{V}^{-1} \mathbf{Z}_1 + \Sigma_\alpha^{-1} \end{cases}$$

donde:

$$\mathbf{W}^\alpha = \mathbf{Y} - (\mathbf{X}\beta + \phi_T \mathbf{T}(\mathbf{Y} - \mathbf{X}\beta) + \phi_S \mathbf{S}(\mathbf{Y} - \mathbf{X}\beta) + \phi_{ST} \mathbf{ST}(\mathbf{Y} - \mathbf{X}\beta) + \phi_{TS} \mathbf{TS}(\mathbf{Y} - \mathbf{X}\beta))$$

Demostración:

Condicionando a los datos $(\mathbf{Y}, \mathbf{Z}, \mathbf{X})$ y al resto de parámetros, se sigue de las ecuaciones (5.1) y (5.2) que:

$$\begin{aligned} \mathbf{W}_1^\alpha \mid \boldsymbol{\alpha} &\sim N_{n-n_0}(\mathbf{Z}_1 \boldsymbol{\alpha}, \tau^{-1} \mathbf{V}) \\ \boldsymbol{\alpha} &\sim N_q(\mathbf{0}, \boldsymbol{\Sigma}_\alpha) \end{aligned}$$

el resultado se sigue aplicando el lema 3.1 (ver capítulo 3)

Teorema 5.2

La distribución completamente condicionada del parámetro $\boldsymbol{\beta}$ viene dada por:

$$\boldsymbol{\beta} \mid \boldsymbol{\alpha}, \phi, \tau_n, r, \tau, m_T, \gamma, m_S, \lambda, \mathbf{Y}, \mathbf{Z}, \mathbf{X} \sim N_k(\mathbf{m}_\beta, \mathbf{S}_\beta) \quad (5.14)$$

$$\text{con } \begin{cases} \mathbf{m}_\beta = \mathbf{S}_\beta \left(\tau (\mathbf{W}_1^\beta)' \mathbf{V}^{-1} \mathbf{F}_1^\beta \right) \\ \mathbf{S}_\beta^{-1} = \tau (\mathbf{W}_1^\beta)' \mathbf{V}^{-1} \mathbf{W}_1^\beta + \boldsymbol{\Sigma}_\beta^{-1} \end{cases}$$

donde:

$$\mathbf{W}^\beta = \mathbf{X} - \phi_T \mathbf{TX} - \phi_S \mathbf{SX} - \phi_{ST} \mathbf{STX} - \phi_{TS} \mathbf{TSX}$$

$$\mathbf{F}^\beta = \mathbf{Y} - \mathbf{Z}\boldsymbol{\alpha} - \phi_T \mathbf{TY} - \phi_S \mathbf{SY} - \phi_{ST} \mathbf{STY} - \phi_{TS} \mathbf{TSY}$$

Demostración:

Condicionando a los datos $(\mathbf{Y}, \mathbf{Z}, \mathbf{X})$ y al resto de parámetros, se sigue de las ecuaciones (5.1) y (5.3) que:

$$\mathbf{F}_1^\beta \mid \boldsymbol{\beta} \sim N_{n-n_0}(\mathbf{W}_1^\beta \boldsymbol{\beta}, \tau^{-1} \mathbf{V})$$

$$\boldsymbol{\beta} \sim N_k(\mathbf{0}, \boldsymbol{\Sigma}_\beta)$$

el resultado se obtiene aplicando el lema 3.1.

Teorema 5.3

La distribución completamente condicionada del parámetro ϕ viene dada por:

$$\phi_i \mid \boldsymbol{\alpha}, \boldsymbol{\beta}, \phi_{-i}, \tau_n, \Gamma, \tau, m_T, \gamma, m_S, \lambda, \mathbf{Y}, \mathbf{Z}, \mathbf{X} \sim N(m_{\phi_i}, s_{\phi_i}^2) \text{ truncada en } (-1, 1) \quad (5.15)$$

$$\text{con } i \in \{ T, S, ST, TS \} \text{ y } \begin{cases} m_{\phi_i} = \frac{(\mathbf{W}_1^\phi)'_i \mathbf{V}^{-1}(\mathbf{F}_1^\phi)}{(\mathbf{W}_1^\phi)'_i \mathbf{V}^{-1}(\mathbf{W}_1^\phi)_i} \\ s_{\phi_i}^2 = \frac{1}{\tau(\mathbf{W}_1^\phi)'_i \mathbf{V}^{-1}(\mathbf{W}_1^\phi)_i} \end{cases}$$

siendo:

$$\mathbf{W}^\phi = (\mathbf{T}(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}), \mathbf{S}(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}), \mathbf{ST}(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}), \mathbf{TS}(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}))$$

$$(\mathbf{F}^{\phi_i}) = \mathbf{Y} - \mathbf{Z}\boldsymbol{\alpha} - \mathbf{X}\boldsymbol{\beta} - (\mathbf{W}^\phi)_{-i} \phi_{-i}$$

donde $(\mathbf{W}^\phi)_i$ denota la i -ésima columna de \mathbf{W}^ϕ , $(\mathbf{W}^\phi)_{-i}$ es la submatriz de \mathbf{W}^ϕ formada al eliminar la i -ésima columna de ésta y ϕ_{-i} es el subvector de ϕ obtenido al eliminarle su i -ésima componente.

Demostración:

Condicionando a los datos $(\mathbf{Y}, \mathbf{Z}, \mathbf{X})$ y al resto de parámetros, a partir de (5.12) podemos escribir que:

$$\begin{aligned}
 & [\phi_i | \alpha, \beta, \phi, \tau_n, r, \tau, m_T, \gamma, m_S, \lambda, \mathbf{Y}, \mathbf{Z}, \mathbf{X}] \propto \\
 & \propto \exp \left\{ -\frac{\tau}{2} \left(\mathbf{F}_i^\phi - (\mathbf{W}_i^\phi)' \phi_i \right)' \mathbf{V}^{-1} \left(\mathbf{F}_i^\phi - (\mathbf{W}_i^\phi)' \phi_i \right) \right\} I_{(-1,1)}(\phi_i) \propto \\
 & \propto \exp \left\{ -\frac{\tau}{2} (\phi_i - m_\phi)' (s_\phi^2)^{-1} (\phi_i - m_\phi) \right\} I_{(-1,1)}(\phi_i)
 \end{aligned}$$

de donde se obtiene el resultado (5.15)

Teorema 5.4

La distribución completamente condicionada del parámetro τ_n viene dada por:

$$\tau_i | \alpha, \beta, \phi, \tau_{-i}, r, \tau, m_T, \gamma, m_S, \lambda, \mathbf{Y}, \mathbf{Z}, \mathbf{X} \sim \text{Gamma} \left(\frac{r+1}{2}, \frac{r + \tau \varepsilon_{li}^2}{2} \right) \quad (5.16)$$

$i = 1, \dots, n$ independientes donde ε_{li} es la i -ésima componente del vector $\boldsymbol{\varepsilon}_i$ y con τ_{-i} el subvector de $\boldsymbol{\tau}_n$ que se obtiene al quitarle su i -ésima componente.

Demostración

De (5.12) se sigue que la densidad de $\tau_i | \alpha, \beta, \phi, \tau_{-i}, r, \tau, m_T, \gamma, m_S, \lambda, \mathbf{Y}, \mathbf{Z}, \mathbf{X}$ es proporcional a:

$$\prod_{i=1}^n \tau_i^{\frac{r+1}{2}-1} \exp \left[-\frac{\tau_i}{2} (r + \tau \varepsilon_{li}^2) \right] I_{(0,\infty)}(\tau_i) \quad (5.17)$$

de donde se deduce el resultado.

Teorema 5.5

La distribución completamente condicionada de τ viene dada por:

$$\tau \mid \alpha, \beta, \phi, \tau_n, r, m_T, \gamma, m_S, \lambda, \mathbf{Y}, \mathbf{Z}, \mathbf{X} \sim \text{Gamma}\left(\frac{d_0 + n - n_0}{2}, \frac{d_0 s_0 + \boldsymbol{\varepsilon}_1' \mathbf{V}^{-1} \boldsymbol{\varepsilon}_1}{2}\right) \quad (5.18)$$

Demostración:

De (5.7) se sigue que la densidad de la distribución es proporcional a:

$$\tau^{\frac{d_0 + n - n_0 - 1}{2}} \exp\left[-\frac{\tau}{2}(d_0 s_0 + \boldsymbol{\varepsilon}_1' \mathbf{V}^{-1} \boldsymbol{\varepsilon}_1)\right] I_{(0, \infty)}(\tau)$$

de donde se deduce el resultado.

Teorema 5.6

La distribución completamente condicionada de:

$$(m_T, m_S) \mid \alpha, \beta, \phi, \tau_n, r, \tau, \gamma, \lambda, \mathbf{Y}, \mathbf{Z}, \mathbf{X}$$

es una distribución discreta con soporte $\mathbf{m}_{Tred} \times \mathbf{m}_{Sred}$ y función de probabilidad dada por:

$$P(m_T = m_{Tred,a}, m_S = m_{Sred,c} \mid \alpha, \beta, \phi, \tau_n, r, \tau, \gamma, \lambda, \mathbf{Y}, \mathbf{Z}, \mathbf{X}) \propto \exp\left[-\frac{\tau}{2} \boldsymbol{\varepsilon}_1'(\alpha, \beta, \phi, m_{Tred,a}, \gamma, m_{Sred,c}, \lambda) \mathbf{V}^{-1} \boldsymbol{\varepsilon}_1(\alpha, \beta, \phi, m_{Tred,a}, \gamma, m_{Sred,c}, \lambda)\right] \quad (5.19)$$

a = 1, ..., t c = 1, ..., s

Demostración

Es inmediata a partir de (5.8) y (5.10).

Por (5.6) la distribución completamente condicionada del parámetro r $r \mid \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\phi}, \boldsymbol{\tau}_n, \tau, m_T, \gamma, m_S, \boldsymbol{\lambda}, \mathbf{Y}, \mathbf{Z}, \mathbf{X}$ es proporcional a:

$$\left(\frac{r}{2}\right)^{\binom{n}{\frac{r}{2}}} \frac{\exp\left[-\frac{1}{2}\left(\sum_{i=1}^n \tau_i\right)r\right]}{\left(r^{\frac{1}{2}}\Gamma\left(\frac{r}{2}\right)\right)^n} r^{m_1 + \frac{n}{2} - 1} \exp\left[-\frac{1}{2}\left(2m_2 - \sum_{i=1}^n \log(\tau_i)\right)r\right] I_{(0,\infty)}(r) \quad (5.20)$$

Esta distribución no es estándar. Para extraer una muestra recurrimos a un paso de Hastings-Metropolis (ver Robert y Casella, 2004) tomando como

densidad de transición una distribución Gamma $\left(m_1 + \frac{n}{2}, \frac{(m_1 + 0.5n)^2}{m_2 + \frac{n}{s_\tau^2}}\right)$,

donde $s_\tau^2 = \frac{1}{n} \sum_{i=1}^n (\tau_i - \bar{\tau})^2$ es la varianza muestral de $\{\tau_1, \dots, \tau_n\}$. Dicha distribución se obtiene tras un análisis de los términos más relevantes de la expresión (5.20). El procedimiento se detalla en el algoritmo de estimación.

La distribución completamente condicionada de los parámetros $(\gamma, \boldsymbol{\lambda})$: $(\gamma, \boldsymbol{\lambda}) \mid \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\phi}, \boldsymbol{\tau}_n, \tau, m_T, m_S, \mathbf{Y}, \mathbf{Z}, \mathbf{X}$ es una distribución continua con soporte $(\gamma_{\min}, \gamma_{\max}) \times (\boldsymbol{\lambda}_{\min}, \boldsymbol{\lambda}_{\max})$ y función de densidad proporcional a:

$$\exp\left[-\frac{\tau}{2} \boldsymbol{\epsilon}_1' \mathbf{V}^{-1} \boldsymbol{\epsilon}_1\right] I_{(\gamma_{\min}, \gamma_{\max}) \times (\boldsymbol{\lambda}_{\min}, \boldsymbol{\lambda}_{\max})}(\gamma, \boldsymbol{\lambda})$$

Esta distribución no es estándar y utilizamos un paso de Hastings-Metropolis para obtener una muestra de la misma, tomando como densidad de transición la distribución uniforme en $(\gamma_{\min}, \gamma_{\max}) \times (\boldsymbol{\lambda}_{\min}, \boldsymbol{\lambda}_{\max})$, procedimiento que se detalla en el algoritmo de estimación.

Observación 5.1

Notar que, con la única excepción de la distribución (5.20), la matriz $V^{-1} = \text{diag}(\tau_i)$ aparece “ponderando” la información proporcionada por la observaciones $(\mathbf{Y}, \mathbf{Z}, \mathbf{X})$ sobre los parámetros en todas las distribuciones completamente condicionadas del modelo. En particular, el parámetro τ_i pondera la información proporcionada por la i -ésima observación $(y_i, \mathbf{z}_i, \mathbf{x}_i)$. Ahora bien, de la distribución (5.16) se sigue que:

$$E[\tau_i \mid \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\phi}, \boldsymbol{\tau}, r, \tau, m_T, \gamma, m_S, \lambda, \mathbf{Y}, \mathbf{Z}, \mathbf{X}] = \frac{r+1}{r + \tau \varepsilon_{ii}^2}$$

por lo que, cuanto mayor es el error ε_{ii}^2 , menor tiende a ser el valor obtenido de τ_i y, por lo tanto, menor será el peso que dicha observación reciba en el proceso de extracción de la muestra del resto de los parámetros del modelo por lo que, menor será su influencia en el proceso de estimación.

Por consiguiente, el planteamiento hecho en este capítulo permite realizar inferencias más robustas acerca de los parámetros del modelo al disminuir la influencia de las observaciones que aún siendo considerados fuertes atípicos por el modelo, son explicadas por el mismo.

5.3.2 Algoritmo de estimación

En este punto se describe el algoritmo utilizado para obtener una muestra de la distribución a posteriori (5.12) necesaria para realizar inferencias acerca de los parámetros del modelo. Dicho algoritmo está basado en el *Gibbs sampling*.

Paso 0 Determinar n_{iter} , el número de iteraciones del algoritmo. Extraer una muestra inicial $\left(\boldsymbol{\alpha}^{(0)}, \boldsymbol{\beta}^{(0)}, \boldsymbol{\phi}^{(0)}, \boldsymbol{\tau}_n^{(0)}, \mathbf{r}^{(0)}, \boldsymbol{\tau}^{(0)}, \mathbf{m}_T^{(0)}, \gamma^{(0)}, \mathbf{m}_S^{(0)}, \lambda^{(0)}\right)$ de la distribución a priori del vector de parámetros $\boldsymbol{\theta}$

Paso 1 Repetir para $i=1, \dots, n_{\text{iter}}$

1) Extraer $\boldsymbol{\alpha}^{(i)}$ de

$$\boldsymbol{\alpha} \mid \boldsymbol{\beta}^{(i-1)}, \boldsymbol{\phi}^{(i-1)}, \boldsymbol{\tau}_n^{(i-1)}, \mathbf{r}^{(i-1)}, \boldsymbol{\tau}^{(i-1)}, \mathbf{m}_T^{(i-1)}, \gamma^{(i-1)}, \mathbf{m}_S^{(i-1)}, \lambda^{(i-1)}, \mathbf{Y}, \mathbf{Z}, \mathbf{X} \sim$$

$$N_q \left(\mathbf{m}_\alpha^{(i)}, \mathbf{S}_\alpha^{(i)} \right) \text{ donde:}$$

$$\begin{cases} \mathbf{S}_\alpha^{(i)} = \left(\boldsymbol{\tau}^{(i-1)} \mathbf{Z}'_1 \left(\mathbf{V}^{(i-1)} \right)^{-1} \mathbf{Z}_1 + \boldsymbol{\Sigma}_\alpha^{-1} \right)^{-1} \\ \mathbf{m}_\alpha^{(i)} = \left(\mathbf{S}_\alpha^{(i)} \right) \left(\boldsymbol{\tau}^{(i-1)} \mathbf{Z}'_1 \left(\mathbf{V}^{(i-1)} \right)^{-1} \left(\mathbf{W}_1^\alpha \right)^{(i-1)} \right) \end{cases}$$

con:

$$\begin{aligned} \left(\mathbf{W}^\alpha \right)^{(i-1)} = & \mathbf{Y} - \mathbf{X} \boldsymbol{\beta}^{(i-1)} - \phi_T^{(i-1)} \mathbf{T}^{(i-1)} \left(\mathbf{Y} - \mathbf{X} \boldsymbol{\beta}^{(i-1)} \right) - \phi_S^{(i-1)} \mathbf{S}^{(i-1)} \left(\mathbf{Y} - \mathbf{X} \boldsymbol{\beta}^{(i-1)} \right) - \\ & - \phi_{ST}^{(i-1)} \mathbf{S}^{(i-1)} \mathbf{T}^{(i-1)} \left(\mathbf{Y} - \mathbf{X} \boldsymbol{\beta}^{(i-1)} \right) - \phi_{TS}^{(i-1)} \mathbf{T}^{(i-1)} \mathbf{S}^{(i-1)} \left(\mathbf{Y} - \mathbf{X} \boldsymbol{\beta}^{(i-1)} \right) \end{aligned}$$

$$\left(\mathbf{V}^{(i-1)} \right)^{-1} = \text{diag} \left(\tau_1^{(i-1)}, \dots, \tau_n^{(i-1)} \right)$$

2) Extraer $\boldsymbol{\beta}^{(i)}$ de:

$$\boldsymbol{\beta} \mid \boldsymbol{\alpha}^{(i)}, \boldsymbol{\phi}^{(i-1)}, \boldsymbol{\tau}_n^{(i-1)}, \mathbf{r}^{(i-1)}, \boldsymbol{\tau}^{(i-1)}, \mathbf{m}_T^{(i-1)}, \gamma^{(i-1)}, \mathbf{m}_S^{(i-1)}, \lambda^{(i-1)}, \mathbf{Y}, \mathbf{Z}, \mathbf{X} \sim$$

$$N_k \left(\mathbf{m}_\beta^{(i)}, \mathbf{S}_\beta^{(i)} \right) \text{ donde:}$$

$$\begin{cases} \mathbf{S}_\beta^{(i)} = \left(\boldsymbol{\tau}^{(i-1)} \left(\left(\mathbf{W}_1^\beta \right)^{(i-1)} \right)' \left(\mathbf{V}^{(i-1)} \right)^{-1} \left(\mathbf{W}_1^\beta \right)^{(i-1)} + \boldsymbol{\Sigma}_\beta^{-1} \right)^{-1} \\ \mathbf{m}_\beta^{(i)} = \left(\mathbf{S}_\beta^{(i)} \right) \left(\boldsymbol{\tau}^{(i-1)} \left(\left(\mathbf{W}_1^\beta \right)^{(i-1)} \right)' \left(\mathbf{V}^{(i-1)} \right)^{-1} \left(\mathbf{R}_1^\beta \right)^{(i-1)} \right) \end{cases}$$

con:

$$(\mathbf{W}^\beta)^{(i-1)} = \mathbf{X} - \phi_T^{(i-1)} \mathbf{T}^{(i-1)} \mathbf{X} - \phi_S^{(i-1)} \mathbf{S}^{(i-1)} \mathbf{X} - \phi_{ST}^{(i-1)} \mathbf{S}^{(i-1)} \mathbf{T}^{(i-1)} \mathbf{X} - \phi_{TS}^{(i-1)} \mathbf{T}^{(i-1)} \mathbf{S}^{(i-1)} \mathbf{X}$$

$$(\mathbf{F}^\beta)^{(i-1)} = \mathbf{Y} - \mathbf{Z}\boldsymbol{\alpha}^{(i)} - \phi_T^{(i-1)} \mathbf{T}^{(i-1)} \mathbf{Y} - \phi_S^{(i-1)} \mathbf{S}^{(i-1)} \mathbf{Y} - \\ - \phi_{ST}^{(i-1)} \mathbf{S}^{(i-1)} \mathbf{T}^{(i-1)} \mathbf{Y} - \phi_{TS}^{(i-1)} \mathbf{T}^{(i-1)} \mathbf{S}^{(i-1)} \mathbf{Y}$$

$$(\mathbf{V}^{(i-1)})^{-1} = \text{diag}(\tau_1^{(i-1)}, \dots, \tau_n^{(i-1)})$$

3) Denotando $(\boldsymbol{\phi}(j-))^{(i)} = (\phi_k^{(i)})_{k < j}$ y $(\boldsymbol{\phi}(j+))^{(i)} = (\phi_h^{(i)})_{h > j}$ extraer $\phi_j^{(i)}$ de

$$\phi_j \left| \boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, (\boldsymbol{\phi}(j-))^{(i)}, (\boldsymbol{\phi}(j+))^{(i-1)}, \boldsymbol{\tau}_n^{(i-1)}, r^{(i-1)}, \boldsymbol{\tau}^{(i-1)}, m_T^{(i-1)}, \gamma^{(i-1)}, m_S^{(i-1)}, \lambda^{(i-1)}, \mathbf{Y}, \mathbf{Z}, \mathbf{X} \right.$$

$\sim N(\mathbf{m}_{\phi_j}^{(i)}, \mathbf{S}_{\phi_j}^{(i)})$ truncada en $(-1, 1)$ para $j \in \{T, S, ST, TS\}$ donde:

$$\left\{ \begin{array}{l} \mathbf{S}_{\phi_j}^{(i)} = \left(\boldsymbol{\tau}^{(i-1)} \left((\mathbf{W}_1^\phi \right)_j^{(i)} \right)' (\mathbf{V}^{(i-1)})^{-1} (\mathbf{W}_1^\phi \right)_j^{(i)} \right)^{-1} \\ \mathbf{m}_{\phi_j}^{(i)} = \left((\mathbf{W}_1^\phi \right)_j^{(i)} \right)' (\mathbf{V}^{(i-1)})^{-1} (\mathbf{F}_1^\phi \right)_j^{(i)} \left(\left((\mathbf{W}_1^\phi \right)_j^{(i)} \right)' (\mathbf{V}^{(i-1)})^{-1} (\mathbf{W}_1^\phi \right)_j^{(i)} \right)^{-1} \end{array} \right.$$

con:

$$(\mathbf{W}^\phi)^{(i)} = (\mathbf{T}^{(i-1)} (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}^{(i)}), \mathbf{S}^{(i-1)} (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}^{(i)}), \mathbf{T}^{(i-1)} \mathbf{S}^{(i-1)} (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}^{(i)}), \mathbf{S}^{(i-1)} \mathbf{T}^{(i-1)} (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}^{(i)}))$$

$$(\mathbf{F}^{\phi_j})^{(i)} = \mathbf{Y} - \mathbf{Z}\boldsymbol{\alpha}^{(i)} - \mathbf{X}\boldsymbol{\beta}^{(i)} - (\mathbf{W}^\phi)_{-j}^{(i)} \phi_j^{(i)}$$

siendo $(\mathbf{W}^\phi)_{-j}^{(i)}$ la matriz \mathbf{W}^ϕ sin su j -ésima columna y

$$\phi_{-j}^{(i)} = \left(\boldsymbol{\phi}(j-)^{(i)}, \boldsymbol{\phi}(j+)^{(i-1)} \right)'.$$

4) Extraer $\tau_j^{(i)}$ de:

$$\tau_j \left| \boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\phi}^{(i)}, \mathbf{r}^{(i-1)}, \boldsymbol{\tau}^{(i-1)}, m_T^{(i-1)}, \gamma^{(i-1)}, m_S^{(i-1)}, \lambda^{(i-1)}, \mathbf{Y}, \mathbf{Z}, \mathbf{X} \sim \right.$$

$$\text{Gamma} \left(\frac{r^{(i-1)} + 1}{2}, \frac{(r^{(i-1)})^2 + \tau^{(i-1)} (\boldsymbol{\varepsilon}_{1j})^2}{2} \right) \text{ para } j = 1, \dots, n \text{ donde } \boldsymbol{\varepsilon}_{1j} \text{ es la}$$

$$j\text{-ésima componente de } \boldsymbol{\varepsilon}_1 \left(\boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\phi}^{(i)}, m_T^{(i-1)}, \gamma^{(i-1)}, m_S^{(i-1)}, \lambda^{(i-1)} \right)$$

5) Extraer $r^{(i)}$ de:

$r \left| \boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\phi}^{(i)}, \boldsymbol{\tau}_n^{(i)}, \boldsymbol{\tau}^{(i-1)}, m_T^{(i-1)}, \gamma^{(i-1)}, m_S^{(i-1)}, \lambda^{(i-1)}, \mathbf{Y}, \mathbf{Z}, \mathbf{X} \right.$ Para ello recurrimos a un paso de Hastings-Metropolis extrayendo $u \sim U(0,1)$

$$r^* \text{ de } \text{Gamma} \left(m_1 + \frac{n}{2}, \frac{(m_1 + 0.5n)^2}{m_2 + \frac{n}{s_\tau^{2(i)}}} \right) \text{ donde } s_\tau^{2(i)} = \frac{1}{n} \sum_{j=1}^n (\tau_j^{(i)} - \bar{\tau}^{(i)})^2$$

es la varianza muestral de $\{\tau_1^{(i)}, \dots, \tau_n^{(i)}\}$.

Se toma entonces $r^{(i)} = r^*$ si $u \leq \min\{1, \alpha\}$ e igual a $r^{(i-1)}$ en otro caso donde:

$$\alpha = \frac{g^{(i)}(r^*) h^{(i)}(r^{(i-1)})}{g^{(i)}(r^{(i-1)}) h^{(i)}(r^*)}$$

$$\text{con } g^{(i)}(r) =$$

$$= \left(\frac{r}{2} \right)^{\binom{n}{\frac{r}{2}}} \frac{\exp \left[-\frac{1}{2} \left(\sum_{j=1}^n \tau_j^{(i)} \right) r \right]}{\left(\frac{1}{r^2} \Gamma \left(\frac{r}{2} \right) \right)^n} r^{m_1 + \frac{n}{2} - 1} \exp \left[-\frac{1}{2} \left(2m_2 - \sum_{j=1}^n \log(\tau_j^{(i)}) \right) r \right] I_{(0, \infty)}(r)$$

siendo $h^{(i)}(\mathbf{r})$ la función de densidad de la distribución

$$\text{Gamma} \left(m_1 + \frac{n}{2}, \frac{(m_1 + 0.5n)^2}{m_2 + \frac{n}{s_\tau^{2(i)}}} \right).$$

6) Extraer $\tau^{(i)}$ de:

$$\tau \mid \boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\phi}^{(i)}, \boldsymbol{\tau}_n^{(i)}, \mathbf{r}^{(i)}, \mathbf{m}_T^{(i-1)}, \gamma^{(i-1)}, \mathbf{m}_S^{(i-1)}, \lambda^{(i-1)}, \mathbf{Y}, \mathbf{Z}, \mathbf{X} \sim$$

$$\text{Gamma} \left(\frac{d_\tau}{2}, \frac{d_\tau s_\tau^{(i)}}{2} \right)$$

donde: $d_\tau = d_0 + n - n_0$ y

$$d_\tau s_\tau^{(i)} = d_0 s_0 + \\ + \boldsymbol{\varepsilon}_1' \left((\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\phi})^{(i)}, (\mathbf{m}_T, \gamma, \mathbf{m}_S, \lambda)^{(i-1)} \right) (\mathbf{V}^{(i)})^{-1} \boldsymbol{\varepsilon}_1 \left((\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\phi})^{(i)}, (\mathbf{m}_T, \gamma, \mathbf{m}_S, \lambda)^{(i-1)} \right)$$

siendo

$$(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\phi})^{(i)} = \left(\boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\phi}^{(i)} \right) \text{ y } (\mathbf{m}_T, \gamma, \mathbf{m}_S, \lambda)^{(i-1)} = \left(\mathbf{m}_T^{(i-1)}, \gamma^{(i-1)}, \mathbf{m}_S^{(i-1)}, \lambda^{(i-1)} \right)$$

$$\text{con } (\mathbf{V}^{(i)})^{-1} = \text{diag}(\tau_1^{(i)}, \dots, \tau_n^{(i)})$$

7) Extraer $(\mathbf{m}_T^{(i)}, \mathbf{m}_S^{(i)})$ de:

$(\mathbf{m}_T, \mathbf{m}_S) \mid \boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\phi}^{(i)}, \boldsymbol{\tau}_n^{(i)}, \mathbf{r}^{(i)}, \tau^{(i)}, \gamma^{(i-1)}, \lambda^{(i-1)}, \mathbf{Y}, \mathbf{Z}, \mathbf{X}$ que es una distribución discreta cuyo soporte es $m_{T_{\text{red}}} \times m_{S_{\text{red}}}$ y distribución de probabilidad dada por:

$$\begin{aligned}
 & P\left(\mathbf{m}_T = \mathbf{m}_{T_{\text{red},a}}, \mathbf{m}_S = \mathbf{m}_{S_{\text{red},c}}\right) \propto \\
 & \propto \exp\left\{-\frac{\tau^{(i)}}{2} \boldsymbol{\varepsilon}_1\left(\boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\phi}^{(i)}, \mathbf{m}_{T_{\text{net},a}}, \gamma^{(i-1)}, \mathbf{m}_{S_{\text{net},c}}, \boldsymbol{\lambda}^{(i-1)}\right) \left(\mathbf{V}^{(i)}\right)^{-1} \boldsymbol{\varepsilon}_1\left(\boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\phi}^{(i)}, \mathbf{m}_{T_{\text{net},a}}, \gamma^{(i-1)}, \mathbf{m}_{S_{\text{net},c}}, \boldsymbol{\lambda}^{(i-1)}\right)\right\} \\
 & \quad \quad \quad a = 1, \dots, t \quad \quad c = 1, \dots, s
 \end{aligned}$$

Como el tamaño de la red de valores posibles de \mathbf{m}_T y \mathbf{m}_S es muy grande, se aplica un paso de Hastings-Metropolis utilizando como distribución de salto la distribución a priori $U_D(\mathbf{m}_{T_{\text{red}}} \times \mathbf{m}_{S_{\text{red}}})$. Se extrae $(\mathbf{m}_T^*, \mathbf{m}_S^*)$ de dicha distribución y $u \sim U(0,1)$. Se toma:

$$\left(\mathbf{m}_T^{(i)}, \mathbf{m}_S^{(i)}\right) = \begin{cases} \left(\mathbf{m}_T^*, \mathbf{m}_S^*\right) & \text{si } u \leq \min\{1, \alpha\} \\ \left(\mathbf{m}_T^{(i-1)}, \mathbf{m}_S^{(i-1)}\right) & \text{si } u > \min\{1, \alpha\} \end{cases}$$

$$\text{donde: } \alpha = \frac{P\left(\mathbf{m}_T = \mathbf{m}_T^*, \mathbf{m}_S = \mathbf{m}_S^*\right)}{P\left(\mathbf{m}_T = \mathbf{m}_T^{(i-1)}, \mathbf{m}_S = \mathbf{m}_S^{(i-1)}\right)}$$

8) Extraer $(\gamma^{(i)}, \boldsymbol{\lambda}^{(i)})$ de:

$(\gamma, \boldsymbol{\lambda}) \mid \boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\phi}^{(i)}, \boldsymbol{\tau}_n^{(i)}, \mathbf{r}^{(i)}, \boldsymbol{\tau}^{(i)}, \mathbf{m}_T^{(i)}, \mathbf{m}_S^{(i)}, \mathbf{Y}, \mathbf{Z}, \mathbf{X}$ mediante un paso de Hastings-Metropolis utilizando como distribución de transición la distribución uniforme en $(\gamma_{\min}, \gamma_{\max}) \times (\boldsymbol{\lambda}_{\min}, \boldsymbol{\lambda}_{\max})$. Para ello se simulan (λ^*, γ^*) de la distribución uniforme anterior y u de $U(0,1)$ y se toma:

$$\left(\gamma^{(i)}, \boldsymbol{\lambda}^{(i)}\right) = \begin{cases} \left(\gamma^*, \boldsymbol{\lambda}^*\right) & \text{si } u \leq \min\{1, \alpha\} \\ \left(\gamma^{(i-1)}, \boldsymbol{\lambda}^{(i-1)}\right) & \text{si } u > \min\{1, \alpha\} \end{cases}$$

$$\text{donde } \alpha = \frac{g\left(\gamma^*, \boldsymbol{\lambda}^*\right)}{g\left(\gamma^{(i-1)}, \boldsymbol{\lambda}^{(i-1)}\right)} \text{ con:}$$

$$g(\gamma, \lambda) = \exp \left\{ -\frac{\tau^{(i)}}{2} \boldsymbol{\varepsilon}_i' \left(\boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\phi}^{(i)}, m_T^{(i)}, \gamma, m_S^{(i)}, \lambda \right) \left(\mathbf{V}^{(i)} \right)^{-1} \boldsymbol{\varepsilon}_i \left(\boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\phi}^{(i)}, m_T^{(i)}, \gamma, m_S^{(i)}, \lambda \right) \right\}$$

Una vez que las n_{iter} iteraciones han sido completadas, se descartan las primeras it_0 que sean necesarias para alcanzar la convergencia. Una muestra aproximada de la distribución a posteriori vendría dada por:

$$\left\{ \left(\boldsymbol{\alpha}^{(i)}, \boldsymbol{\beta}^{(i)}, \boldsymbol{\phi}^{(i)}, \boldsymbol{\tau}_n^{(i)}, r^{(i)}, \boldsymbol{\tau}^{(i)}, m_T^{(i)}, \gamma^{(i)}, m_S^{(i)}, \lambda^{(i)} \right) \mid i = it_0, it_0 + s, \dots, n_{iter} = it_0 + (n_{muestra} - 1)s \right\} \quad (5.21)$$

donde el salto s se determina para que las componentes de la muestra sean incorreladas.

Concluido el proceso de estimación, se procede a la validación del modelo tanto desde el punto de vista intramuestral como extramuestral de forma similar a lo llevado a cabo en capítulos anteriores y que se encuentran descritos con detalle en el apartado 3.4, con las modificaciones que se especifican a continuación.

Para la elaboración de predicciones y bondad de ajuste extramuestral presentado en el apartado 3.4.2.2, la muestra 3.26 de la distribución a posteriori viene dada en este caso por:

$$\boldsymbol{\theta}^{*(k)} = \left(\boldsymbol{\alpha}^{*(k)}, \boldsymbol{\beta}^{*(k)}, \boldsymbol{\phi}^{*(k)}, \boldsymbol{\tau}^{*(k)}, r^{*(k)}, \boldsymbol{\tau}^{*(k)}, m_T^{*(k)}, \gamma^{*(k)}, m_S^{*(k)}, \lambda^{*(k)} \right)$$

Así mismo, la distribución $\left[y_i \mid \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \boldsymbol{\theta}^{*(k)} \right]$ es una distribución $t_{r^{(k)}} \left(\boldsymbol{\mu}_r^{(k)}, \left(\boldsymbol{\tau}^{*(k)} \right)^{-1} \right)$, donde $t_n(m, s^2)$ denota una distribución t de Student con n grados de libertad de media m y parámetro de escala s^2 .

5.4 Resultados

La metodología expuesta en la sección anterior se aplica a la base de datos presentada en 2.6.1 reseñándose únicamente, los cambios y novedades que aparecen sobre lo ya mostrado en el capítulo anterior.

5.4.1 Estimación del modelo

El algoritmo de estimación se ejecuta durante 10000 iteraciones. Las Figuras 5.1 a 5.7 muestran las cadenas de cada uno de los parámetros del modelo y sus correspondientes correlogramas. En casi todas ellas se alcanza la convergencia tras aproximadamente 1000 iteraciones y la autocorrelación de las series resultantes se hace despreciable a partir de 10 retardos. Las cadenas correspondientes a los coeficientes τ y r (Figura 5.5), y γ y λ (Figura 5.7) muestran un decaimiento lento pero nítido de la dependencia y de nuevo la cadena correspondiente al coeficiente β_{sup} (coeficiente hedónico de la superficie, ver Figura 5.2) muestra un decrecimiento más pausado.

Al igual que en el capítulo anterior se ha llevado a cabo un análisis de sensibilidad sobre las inferencias de este parámetro, sin encontrar diferencias. Se toma también en este caso una muestra de cada 10 pasos para todos los parámetros.

Los errores estándar de simulación de los parámetros han sido menores que 0.005 para la mayor parte de ellos, con las únicas excepciones de la constante (0.0332), y los parámetros m_T (0.1625), m_S (0.1355) y r (0.0178)

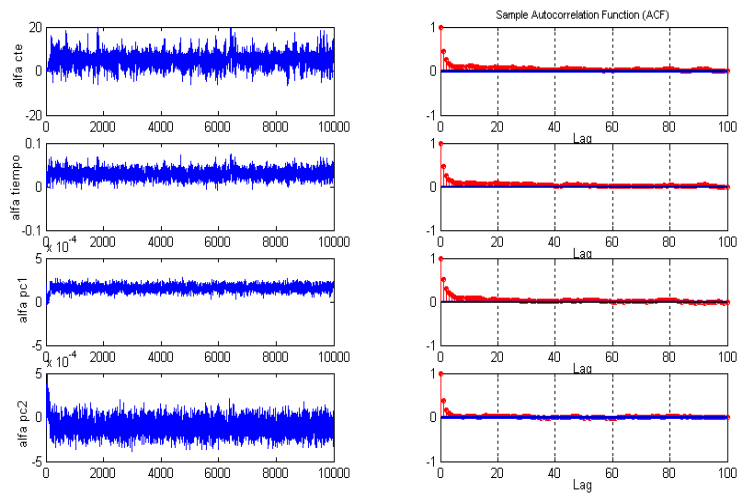


Figura 5.1: Iteraciones del algoritmo de Gibbs y funciones de autocorrelación para las cadenas correspondientes a los coeficientes α de la constante, el indicador de tendencia temporal, PC1 y PC2, respectivamente

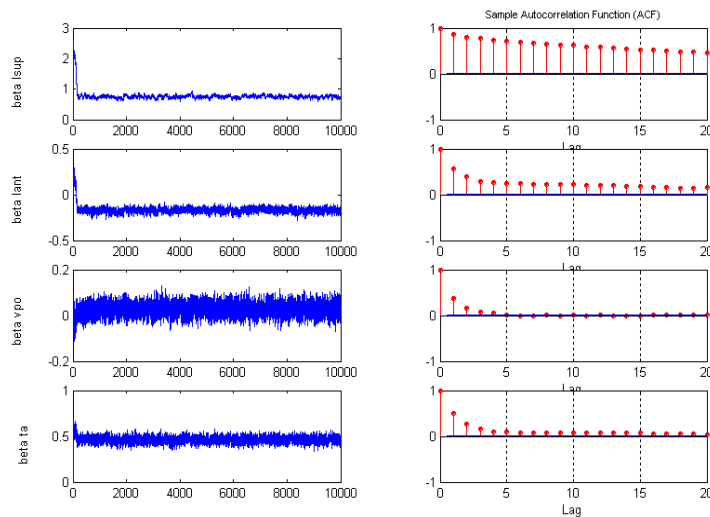


Figura 5.2: Iteraciones del algoritmo de Gibbs y funciones de autocorrelación para las cadenas correspondientes a los coeficientes β de log superficie, log antigüedad, VPO y tipo de adquisición, respectivamente

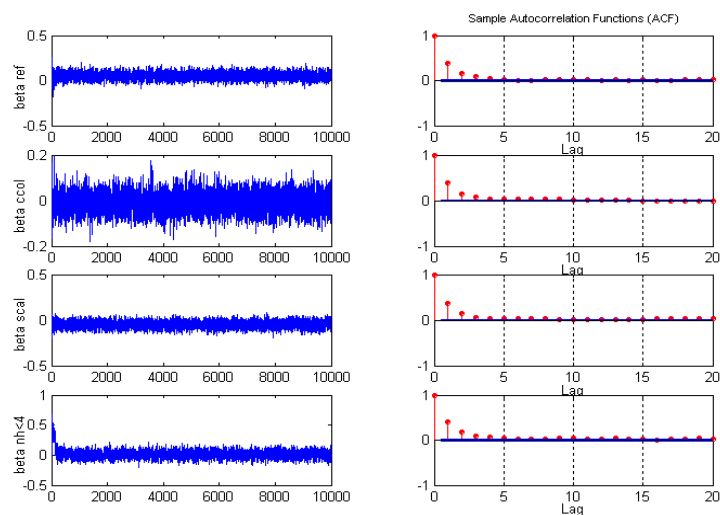


Figura 5.3: Iteraciones del algoritmo de Gibbs y funciones de autocorrelación para las cadenas correspondientes a los coeficientes β de refrigeración, calefacción colectiva, sin calefacción y n° habitaciones < 4

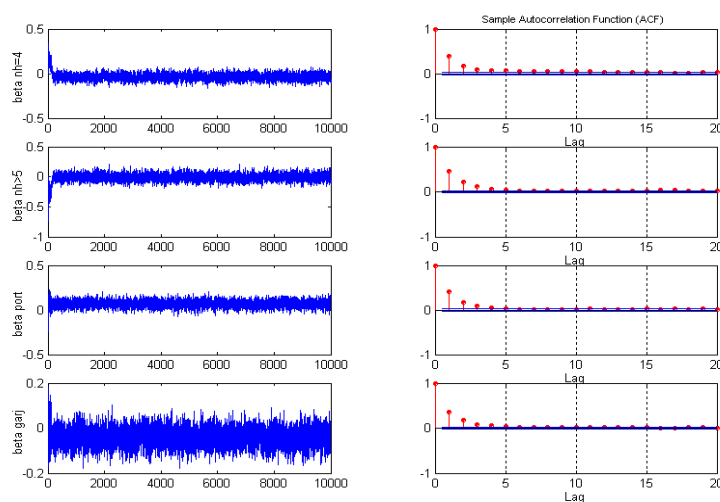


Figura 5.4: Iteraciones del algoritmo de Gibbs y funciones de autocorrelación para las cadenas correspondientes a los coeficientes β n° de habitaciones = 4, n° de habitaciones > 5 , portero y garaje, respectivamente

Resultados

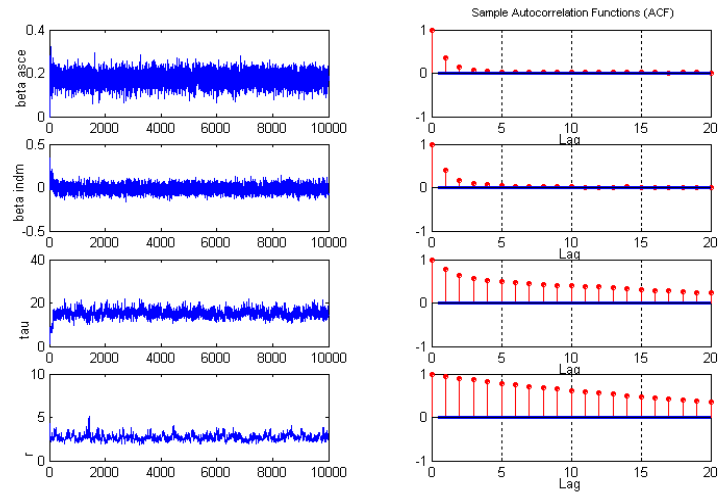


Figura 5.5: Iteraciones del algoritmo de Gibbs y funciones de autocorrelación para las cadenas correspondientes a los coeficientes β de ascensor e indicador de datos ausentes, del parámetro τ y del parámetro r

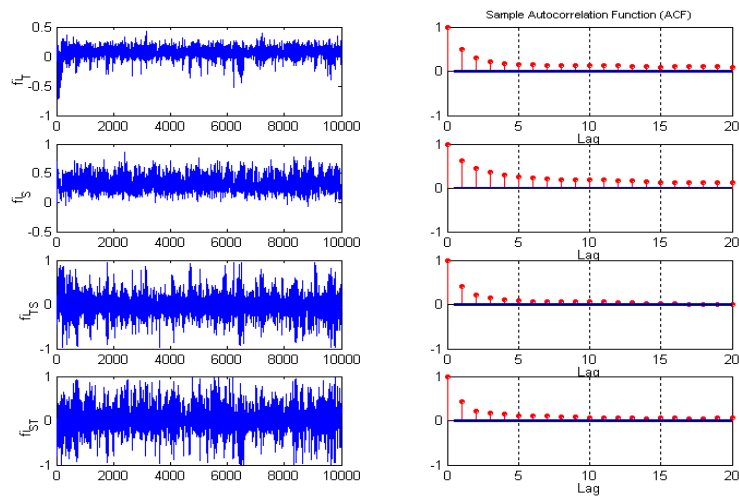


Figura 5.6: Iteraciones del algoritmo de Gibbs y funciones de autocorrelación para las cadenas correspondientes a los coeficientes ϕ_T , ϕ_S , ϕ_{TS} y ϕ_{ST}

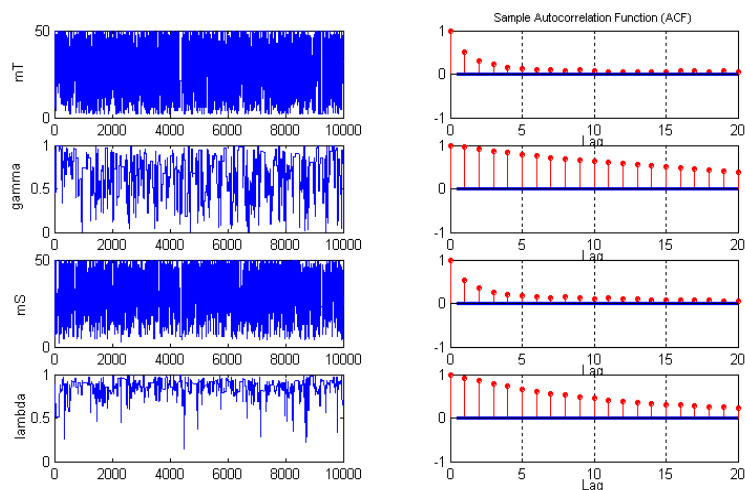


Figura 5.7: Iteraciones del algoritmo de Gibbs y funciones de autocorrelación para las cadenas correspondientes a los coeficientes m_T , γ , m_S y λ

La Tabla 5.1 expone las estimaciones puntuales de los parámetros con sus correspondientes intervalos de credibilidad bayesianos al 95%. Sus respectivas densidades a posteriori, calculadas con la muestra (5.13), se pueden observar en las Figuras 5.8 a 5.14.

Se observa, con respecto a los resultados de capítulos anteriores, coincidencia en los signos y valores similares en las estimaciones puntuales de los parámetros del modelo, constatándose un aumento de su precisión que se manifiesta en una reducción de la longitud de los intervalos de credibilidad. Esta mejora se debe, fundamentalmente, a la disminución de la influencia ejercida por las observaciones atípicas, tal y como se ha explicado en la observación 5.1.

Esta disminución ha propiciado, así mismo, que la débil significación estadística de los efectos de algunas de las variables (PC2, garaje, calefacción colectiva) haya desaparecido, con respecto a los modelos estimados en los capítulos anteriores, probablemente, al estar fundamentadas en unas pocas transacciones. En este sentido los modelos estimados en este capítulo, clarifican la influencia ejercida por las variables hedónicas discerniendo cuáles están soportadas por un mayor número de observaciones y cuáles no.

Los parámetros de vecindad m_T y m_S , así como la ponderación de los vecinos temporales γ , siguen presentando gran incertidumbre aunque puede apreciarse que el parámetro m_S es significativamente distinto de 0 (ver Figura 5.14). Por otro lado, la estimación de la ponderación sobre los vecinos espaciales λ , se ciñe con más claridad a valores altos del mismo (entre 0.68 y 0.98), dando similar importancia a la información proporcionada por todos los vecinos espaciales que el modelo ha considerado influyentes sobre el precio de una vivienda. Todo ello pone de manifiesto la existencia de dependencias espaciales de carácter local significativas no apreciándose, sin embargo, la de dependencias temporales.

Estos hechos son coherentes con las estimaciones de los parámetros autorregresivos ϕ , en los que tan solo ϕ_S , que recoge efectos espaciales, resulta significativo con signo positivo. Este resultado concuerda con la significación estadística de los parámetros de vecindad m_S y λ comentada anteriormente y encontradas por todos los modelos estimados en la Memoria. En dichos modelos, sin embargo, ninguno de los parámetros relacionados con los efectos espaciales (ϕ_S , ϕ_{ST} , ϕ_{TS}) resultó significativo debido al alto grado de incertidumbre asociado a su estimación, provocado muy posiblemente, por la existencia de observaciones atípicas. Al disminuir la influencia de éstas queda puesto de manifiesto cuáles de dichos parámetros son significativos y cuáles

no. Mención aparte merece la no significación del parámetro ϕ_T , fiel reflejo de la no significación de los parámetros de vecindad m_T y γ .

La estimación del parámetro r , entre 2.10 y 3.51, se debe fundamentalmente, a la presencia de observaciones cuyos términos de error ε_i son grandes en valor absoluto y, más concretamente, muy negativos (ver Figuras 5.15 y 5.17) correspondiendo en su mayor parte, a transacciones cuyo tipo de adquisición no fue una compraventa y con un nivel de precios anormalmente bajo. Este hecho hace que aumente el grado de leptocurtosis de la distribución del término de error y provoque un bajo valor de los grados de libertad r . Los gráficos de probabilidad de m_T y m_S , así como las densidades del resto de los parámetros (ver Figuras 5.8 a 5.14) inciden en todo lo reseñado en los comentarios de los capítulos anteriores.

Tabla 5.1: Estimaciones de los parámetros (en negrita los significativos para un nivel de credibilidad del 95% y con⁺ al 90%)

VARIABLES	STAR			STARs			hedónico		
	C 2.5	Mediana	C 97.5	C 2.5	Mediana	C 97.5	C 2.5	Mediana	C 97.5
Constante	-0.6476	5.1031 ⁺	10.6244	3.9749	5.5794	6.8945	7.9465	8.4512	8.8855
Tiempo	0.0100	0.0285	0.0499	0.0221	0.0322	0.0408	0.0304	0.0416	0.0519
PC1	0.0001	0.0002	0.0002	0.0001	0.0001	0.0002	0.0002	0.0002	0.0003
PC2	-0.0003	-0.0001	0.0001				-0.0003	-0.0002	0.0000
Log (superficie)	0.6626	0.7513	0.8367	0.7053	0.7639	0.8318	0.6713	0.7643	0.8537
Log (antigüedad)	-0.2226	-0.1699	-0.1254	-0.2091	-0.1711	-0.1386	-0.2300	-0.1750	-0.1266
VPO	-0.0246	0.0279	0.0780				-0.0260	0.0251	0.0727
Tipo de adquisición	0.3999	0.4628	0.5266	0.4246	0.4792	0.5295	0.4060	0.4772	0.5426
Refrigeración	-0.0263	0.0532	0.1262				-0.0187	0.0578	0.1322
Calefacción colectiva	-0.0906	-0.0096	0.0700				-0.0789	0.0009	0.0813
Sin calefacción	-0.1236	-0.0498	0.0213				-0.1173	-0.0480	0.0213
Nº habitación es <4	-0.0941	0.0086	0.1201				-0.0814	0.0261	0.1277
Nº habitaciones =4	-0.1033	-0.0381	0.0283				-0.0932	-0.0237	0.0412
Nº habitaciones >5	-0.1077	-0.0054	0.1022				-0.1056	0.0010	0.1017
Portero	-0.0030	0.0696 ⁺	0.1359	0.0111	0.0693	0.1274	0.0148	0.0851	0.1518
Garaje	-0.1180	-0.0396	0.0375				-0.1308	-0.0560	0.0213
Ascensor	0.1159	0.1729	0.2302	0.1335	0.1783	0.2264	0.1227	0.1788	0.2375
Indicador missing	-0.0919	-0.0125	0.0658				-0.0914	-0.0125	0.0732
ϕ_T	-0.1181	0.0722	0.2199						
ϕ_S	0.1219	0.3319	0.6124	0.1862	0.3438	0.5250			
ϕ_{ST}	-0.4603	-0.0210	0.4229						
ϕ_{TS}	-0.5465	0.0309	0.5562						
τ	12.2541	15.0717	18.6544	12.6008	15.0720	18.2134	11.3619	14.4399	18.4722
r	2.0977	2.6590	3.5138	2.1940	2.6543	3.5225			
m_T	2	29	49	2	26	48			
γ	0.1079	0.7238	0.9863	0.0517	0.4860	0.9493			
m_S	6	31	50	6	29	49			
λ	0.6750	0.8768	0.9769	0.6899	0.8589	0.9520			

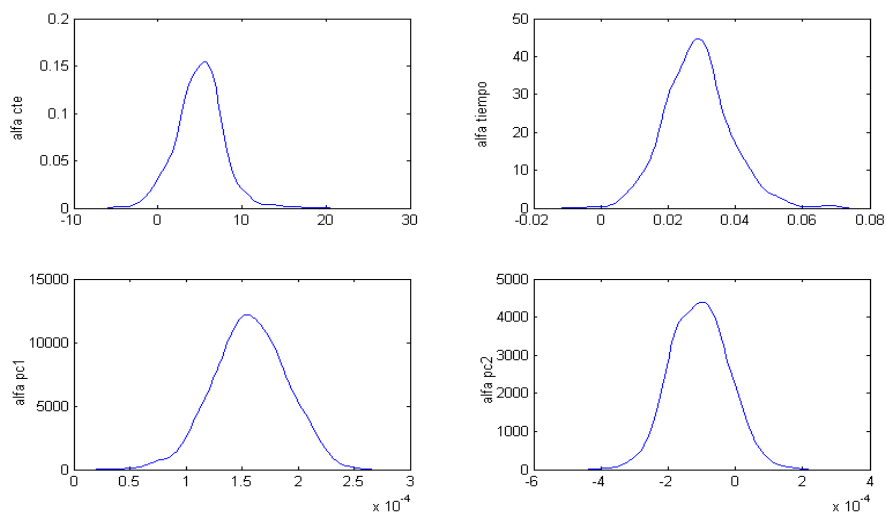


Figura 5.8: Densidades a posteriori de los coeficientes α . De izquierda a derecha y de arriba abajo las correspondientes a la constante, el indicador de tiempo, la componente PC1 y la componente PC2, respectivamente

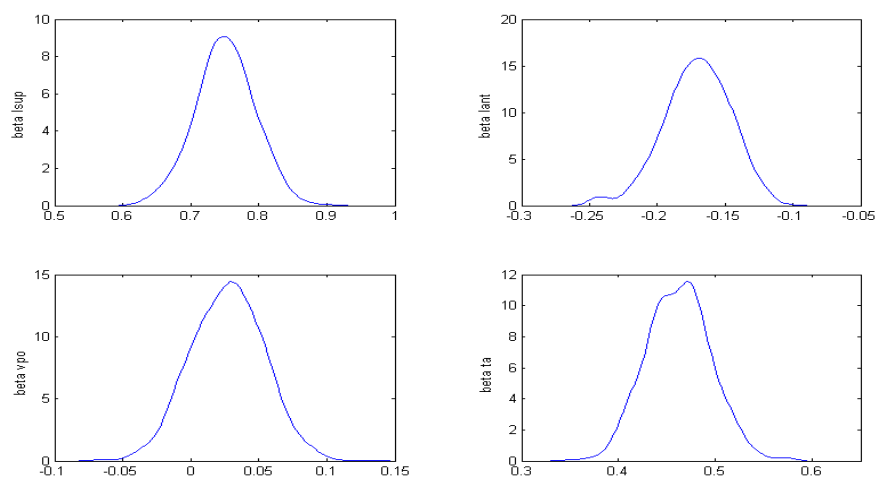


Figura 5.9: De izquierda a derecha y de arriba abajo densidades a posteriori de los coeficientes β correspondientes al logaritmo de superficie, logaritmo de antigüedad, VPO y tipo de adquisición, respectivamente

Resultados

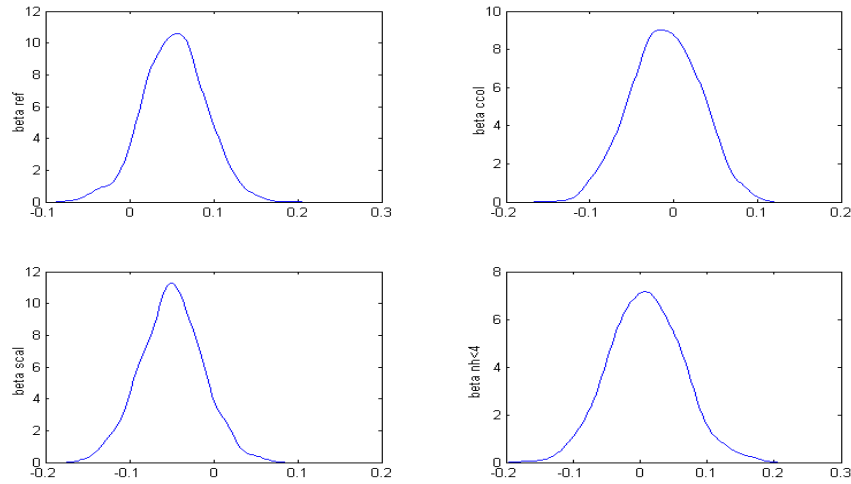


Figura 5.10: De izquierda a derecha y de arriba abajo densidades a posteriori de los coeficientes β correspondientes a las variables refrigeración, calefacción colectiva, sin calefacción y nº de habitaciones < 4, respectivamente

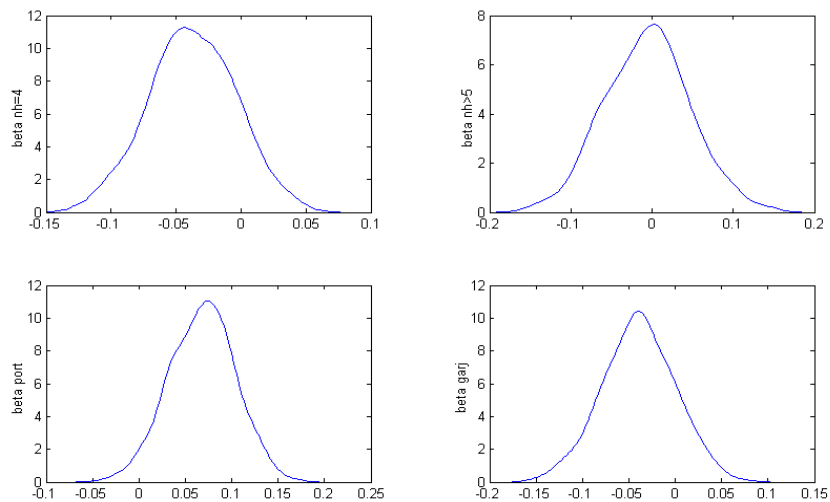


Figura 5.11: De izquierda a derecha y de arriba abajo densidades a posteriori de los coeficientes β correspondientes a las variables nº de habitaciones = 4, nº de habitaciones > 5, portero y garaje, respectivamente

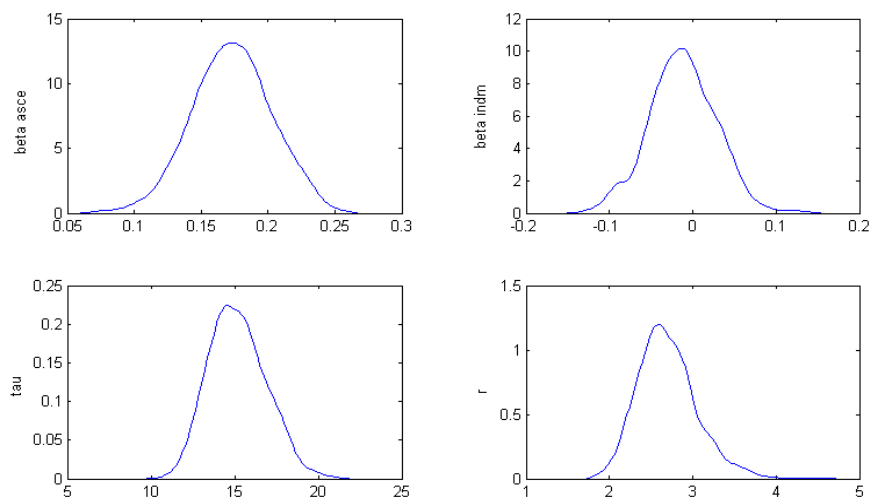


Figura 5.12: De izquierda a derecha y de arriba abajo densidades a posteriori de los coeficientes β correspondientes a las variables ascensor, indicador de datos ausentes, de los parámetro τ y r respectivamente

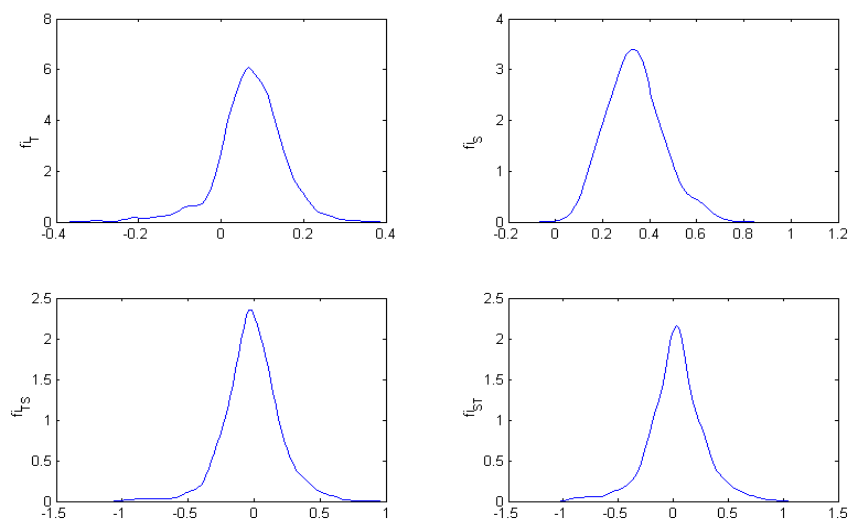


Figura 5.13: De izquierda a derecha y de arriba abajo densidades a posteriori de los coeficientes ϕ_T , ϕ_S , ϕ_{ST} y ϕ_{TS} , respectivamente

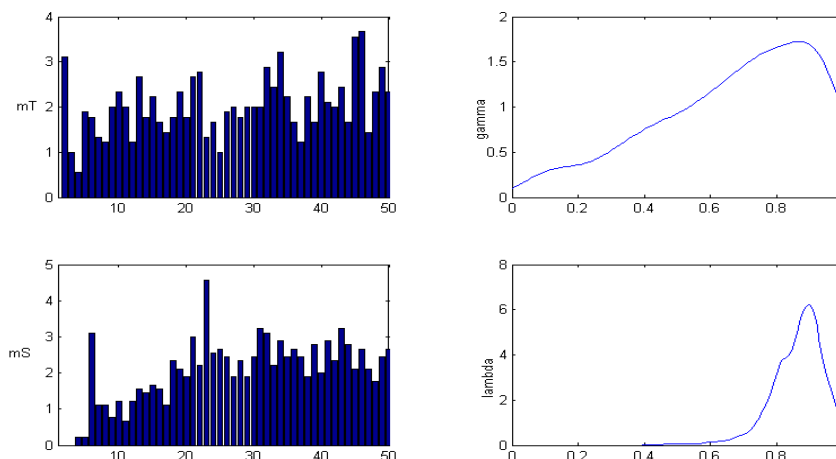


Figura 5.14: A la izquierda gráficos de probabilidad de m_T , y m_S y a la derecha densidades a posteriori de γ y λ .

5.4.2 Validación del modelo

En la Tabla 5.2 se puede observar la distribución de los residuos que presenta, como ya era esperado, unas colas más pesadas, con valores en los criterios de comparación (ver Tabla 5.3) similares a los obtenidos en el capítulo 4. En las Figuras 5.15 a 5.19 se puede confirmar de forma gráfica el adecuado comportamiento de los errores intra y extramuestrales sin patrones sistemáticos, y sin la presencia de autocorrelaciones significativas, tan solo se observa la existencia de atípicos negativos más claramente realzados que en otros modelos como consecuencia de la disminución de su influencia en el proceso de estimación de los parámetros. La apropiada conducta predictiva del modelo, que captura la asimetría positiva que presenta la distribución de los precios, se puede observar en las Figuras 5.20 y 5.21.

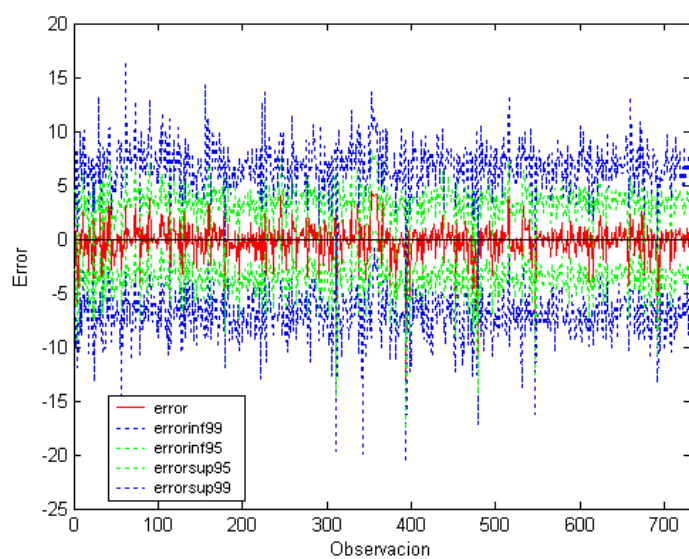


Figura 5.15: Errores intramuestrales del modelo STAR (en líneas rayadas en verde y en azul las bandas de credibilidad al 95% y 99%, respectivamente)

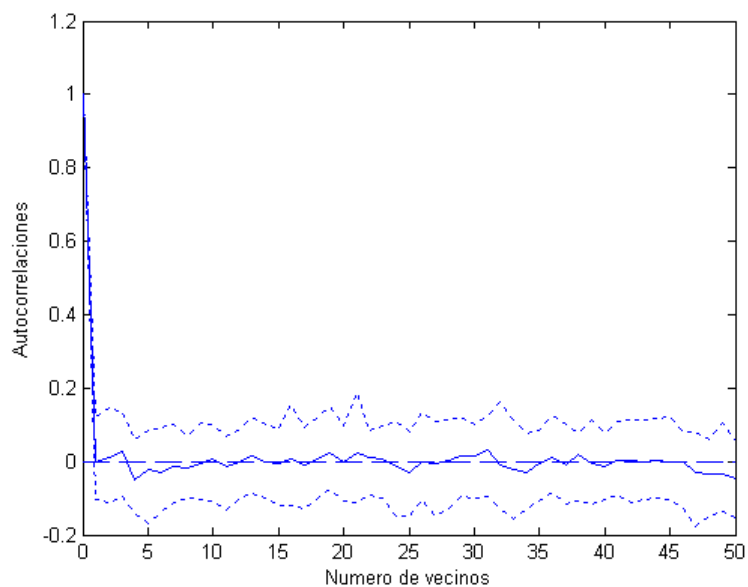


Figura 5.16: Autocorrelaciones espaciales residuales intramuestrales del modelo STAR (en líneas punteadas las bandas de credibilidad al 95%)

Tabla 5.2: Distribución de los errores tipificados intramuestrales del modelo STAR

ERRORES TIPIFICADOS Modelo STAR	
Mínimo	-13.5179
1%	-5.6923
5%	-3.1285
10%	-2.1127
25%	-0.8089
50%	0.0096
75%	0.7145
90%	1.3769
95%	2.2180
99%	4.0290
Máximo	4.5313
Media	-0.1737
Mediana e 	0.7594
Asimetría	-1.7328
Curtosis	12.8550

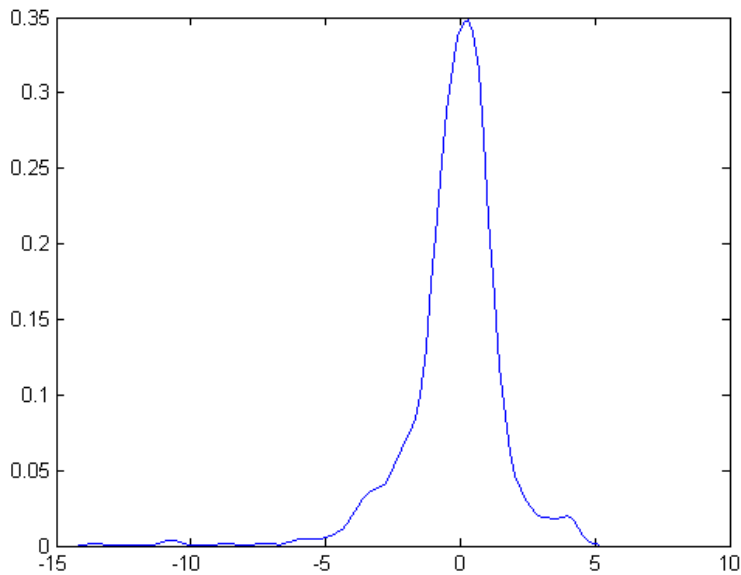


Figura 5.17: Estimación de la densidad de los errores tipificados intramuestrales calculados tomando como estimador de los parámetros su media a posteriori

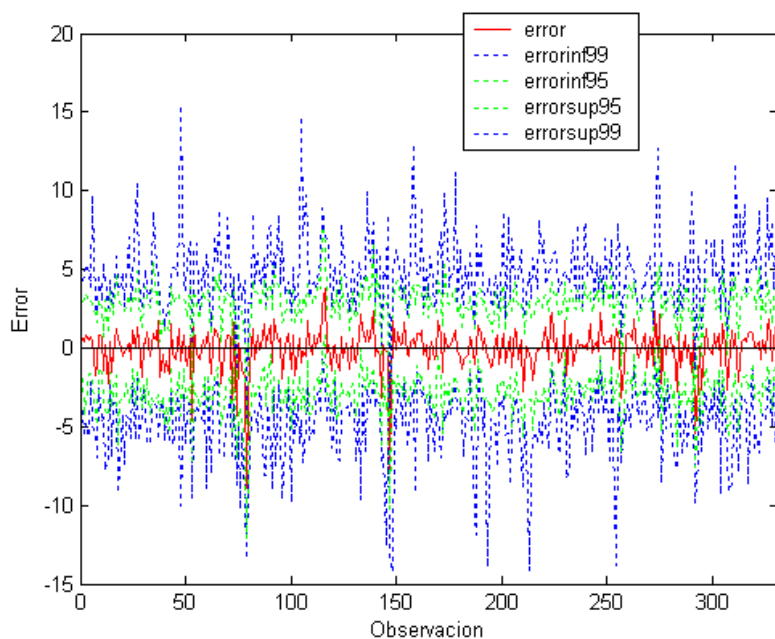


Figura 5.18: Errores tipificados extramuestrales (*rolling*) del modelo STAR

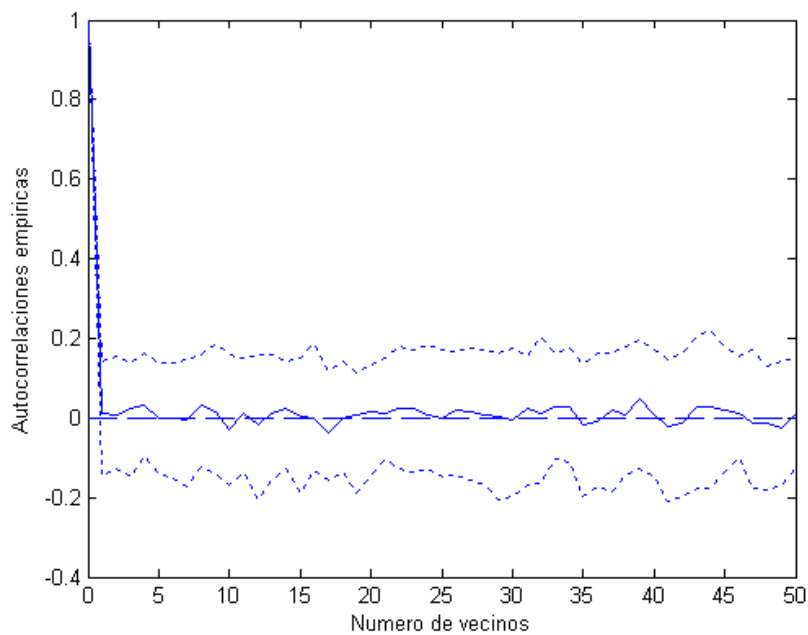


Figura 5.19: Autocorrelaciones espaciales residuales extramuestrales (*rolling*) del modelo STAR (en líneas rayadas las bandas de credibilidad al 95%)

Resultados

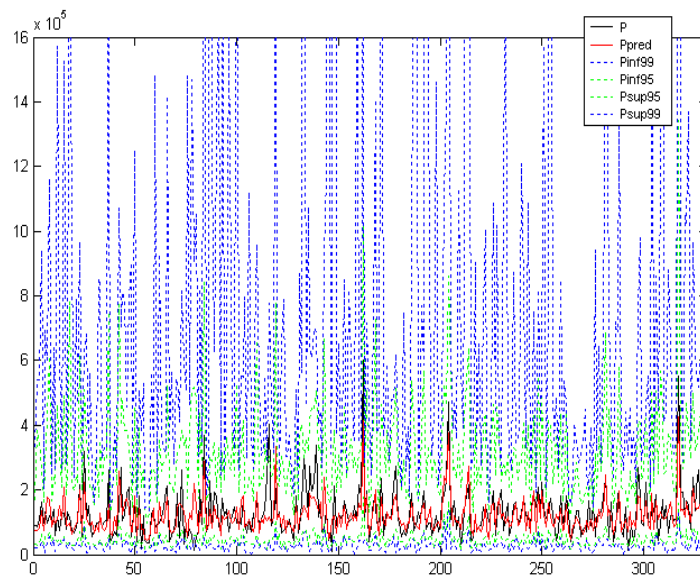


Figura 5.20: Predicciones extramuestrales (*rolling*) del precio para el modelo STAR

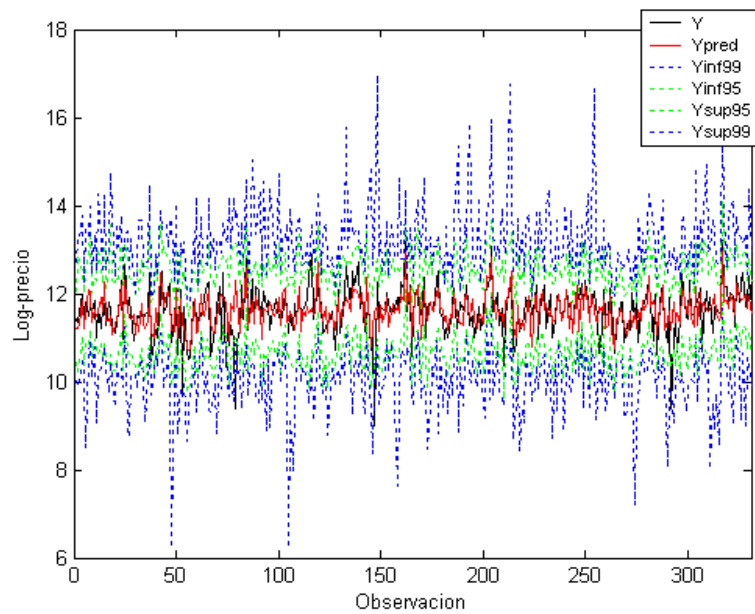


Figura 5.21: Predicciones extramuestrales (*rolling*) del logaritmo del precio para el modelo STAR

Finalmente, la Tabla 5.3 muestra un correcto comportamiento tanto a nivel intramuestral como extramuestral con unos cubrimientos al 95% y al 99%, que en ningún caso presentan diferencias con los resultados teóricos. Se estima que el modelo explica un 42.80% de la varianza, que es ligeramente menor que el porcentaje explicado por los modelos de los capítulos anteriores, debido fundamentalmente, a la influencia ejercida por los errores de las observaciones atípicas, cuya cuantía en valor absoluto tiende a ser mayor como consecuencia de la disminución de su influencia en el proceso de estimación de los parámetros.

Tabla 5.3: Validación del modelo STAR

CRITERIOS	Intramuestrales	Extramuestrales (<i>rolling</i>)
R²	0.4280	0.3756
CUB95	93.98	96.37
CUB99	99.18	98.49

5.4.3 Simplificación del modelo

El objetivo de este apartado es llevar a cabo un procedimiento de comparación del modelo STAR, con una simplificación del mismo (STARs) y con el modelo hedónico tradicional. El modelo STARs se ha obtenido seleccionando aquellas variables que han resultado significativas en la estimación del modelo completo (ver Tabla 5.1) al nivel del 10%.

Este es el mismo proceso que se realizó en los capítulos 3 y 4, donde el modelo resultante fue idéntico, sin embargo, esto no ha ocurrido en el presente capítulo. Aquí el modelo reducido contiene importantes diferencias con respecto al considerado anteriormente, en particular tres variables hedónicas menos, a saber: PC2, calefacción colectiva y garaje, y en cuanto a los coeficientes autorregresivos, únicamente permanece ϕ_s .

En la Tabla 5.4 se pueden examinar los resultados obtenidos en el proceso de comparación. Se advierte la superioridad del modelo reducido **STARs** en comportamiento intramuestral, mostrando mejores puntuaciones en los criterios **DIC** y **DIC***. En el aspecto predictivo extramuestral nos encontramos con resultados, en apariencia al menos, contradictorios. La mejor calificación del criterio **LPRED**, deja clara una mejor conducta predictiva extramuestral del modelo simplificado después de un procedimiento de validación ordinario. Sin embargo, los criterios calculados tras la validación de tipo *rolling* informan de la superior capacidad predictiva del modelo hedónico tradicional.

Tabla 5.4: Criterios de comparación para los modelos STAR, STARs y hedónicos con errores t de Student

CRITERIO	MODELO		
	STAR	STARs	hedónico
DIC	705.74	693.11	711.93
\bar{D}	680.42	681.43	691.57
pd	25.32	11.68	20.36
DIC*	689.65	685.49	698.06
P_D^*	9.23	4.06	6.49
LPRED	-130.57	-112.41	-126.86
PBF(Modelo STARs)	0.0000	1.000	0.0001
RMSE	0.4359	0.4412	0.4103
MAD	0.3013	0.3081	0.2938
LPRED_{rol}	-158.78	-165.08	-149.12

Con el fin de analizar las razones de este comportamiento dispar en el proceso de validación *rolling* analizamos la evolución dinámica del logaritmo de la densidad predictiva dado por la expresión:

$$LPRED_{rol,j}(M) = \sum_{i=n_{est}+1}^j \log([y_i | Y_{i-1}, X_i, Z_i, M]) \text{ para } j= n_{est}+1, \dots, n$$

En la Figura 5.22 se muestran los valores de este criterio para cada uno de los modelos ajustados en los capítulos 3 a 5 de la Memoria. Se observa, en primer lugar, la existencia de un grupo de 9 observaciones¹ en las que se produce una brusca disminución del criterio en todos los modelos comparados. Ello es debido a que el precio predicho por el modelo es, en general, mucho mayor que el valor declarado de la transacción, tal y como se puede ver, por ejemplo, en la Figura 5.18 en la que se aprecia que dichas observaciones corresponden a residuos tipificados significativamente negativos y responsables, en parte, de la elevada asimetría de la distribución de los errores. Un estudio exploratorio de dichas transacciones reveló que corresponden en su mayor parte a transacciones de viviendas no adquiridas mediante compraventa (55,56%), con precios escriturados muy bajos.

Se observa, además, que los modelos con errores normales reaccionan peor al carácter atípico de dichas observaciones debido, fundamentalmente, al poco peso de las colas de su distribución. Por el contrario, los modelos con errores t de Student amortiguan la influencia de dichas observaciones por el elevado valor de los términos v_i correspondientes (o, equivalentemente, por el escaso valor de los términos τ_i) tal y como puede verse en la Figura 5.23 en la que se muestran las bandas de credibilidad bayesiana del 99% para τ_i ,

¹ Corresponden a los instantes 53, 72, 74, 79, 116, 143, 147, 257 y 292 del periodo de validación

Resultados

observándose que para estas observaciones en concreto, las bandas se sitúan muy por debajo de 1.

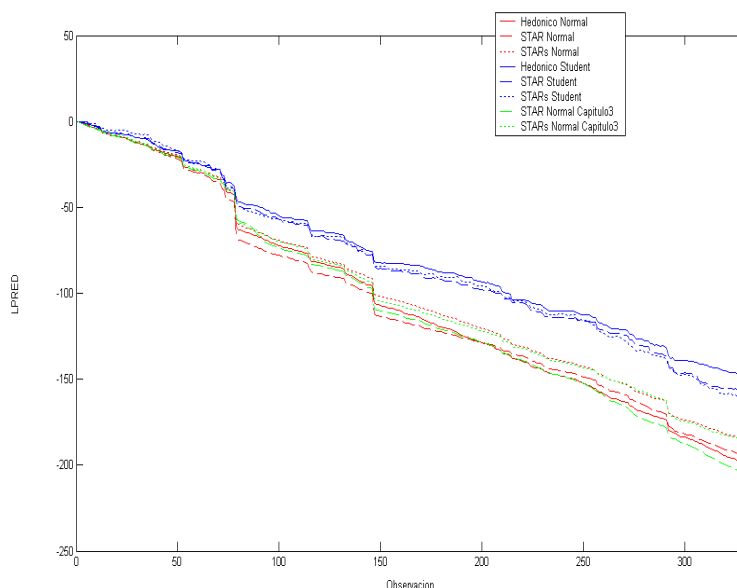


Figura 5.22: Evolución del logaritmo de la densidad predictiva del log-precio utilizando el proceso de validación *rolling* para cada uno de los modelos ajustados en los capítulos 3 a 5

Dentro de los modelos con errores t de Student es el modelo hedónico el que mejor comportamiento presenta. Este hecho es consecuencia, por una parte, de que este modelo contiene como explicativas todas las variables hedónicas proporcionándole mayor flexibilidad que al modelo STARs, más parsimonioso, y por otra, de la ausencia de los efectos espacio-temporales que, en las observaciones atípicas mencionadas, tienden a inflar el peso de las predicciones del modelo STAR por el carácter creciente de dichos efectos.

Estos resultados dejan claro que no siempre el mejor ajuste a los datos, viene acompañado de un mejor comportamiento predictivo extramuestral y que, en nuestro caso, la significación estadística de los efectos espacio-

temporales encontrados puede, en casos aislados y aberrantes, dar lugar a predicciones que contradicen las pautas seguidas por la mayor parte de las observaciones.

Podemos concluir, por tanto, que el modelo STARs es el que presenta mejor grado de ajuste a los datos observados en la mayor parte de los criterios, pero puede fallar en observaciones con precios anormalmente bajos debidos, probablemente, a razones relacionadas con la forma de adquisición y/o con la tributación de dichos inmuebles.

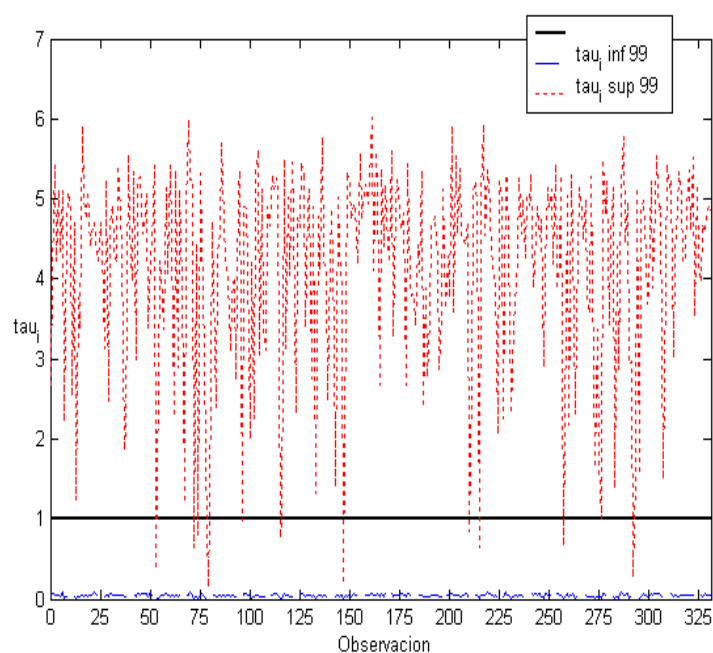


Figura 5.23: Bandas de credibilidad del 99% para los parámetros $\{\tau_i; i=n_{est}+1, \dots, n\}$ en el periodo de validación correspondientes al modelo STAR con errores t de Student (la línea negra sitúa el valor 1 que es el valor esperado de dichos parámetros si el modelo es homocedástico)

5.5 Análisis retrospectivo

La predicción retrospectiva es un tipo de predicción que se lleva a cabo si el fenómeno en estudio tuvo lugar en el pasado o hubiera tenido lugar en un instante anterior al momento en el que se hace la predicción. Habitualmente, este tipo de procedimientos ha estado circunscrito a algunas áreas científicas, como la Meteorología o la Bioestadística, pero resulta poco habitual en general y especialmente en Economía. En el caso particular de la aplicación al mercado inmobiliario, hacer una valoración del precio que hubiera tenido una vivienda de unas características dadas en un momento determinado del pasado, es una situación que se presenta con cierta frecuencia a efectos periciales, tanto en ámbitos fiscales como judiciales.

En esta sección se presenta un procedimiento para llevar a cabo dicha valoración utilizando los modelos STAR con efectos de vecindad analizados a lo largo de la Memoria.

5.5.1 Predicciones retrospectivas

Sean $\mathbf{z} = (z_1, \dots, z_q)'$ y $\mathbf{x} = (x_1, \dots, x_k)'$ las características de una vivienda cuyo precio p se quiere predecir.

La elaboración de predicciones se realiza a partir de la distribución predictiva a posteriori dada por:

$$y \mid \mathbf{Y}, \mathbf{Z} \cup \{\mathbf{z}\}, \mathbf{X} \cup \{\mathbf{x}\} \sim \int [y \mid \mathbf{Y}, \mathbf{Z} \cup \{\mathbf{z}\}, \mathbf{X} \cup \{\mathbf{x}\}, \boldsymbol{\theta}] [\boldsymbol{\theta} \mid \mathbf{Y}, \mathbf{Z}, \mathbf{X}] d\boldsymbol{\theta} \quad (5.22)$$

donde $y = \log(p)$. A partir de esta distribución se obtendrían predicciones puntuales de los precios utilizando la mediana a posteriori de y , así como intervalos de credibilidad Bayesianos calculados a partir de los cuantiles apropiados.

El cálculo de (5.22) se realiza mediante el método de Monte Carlo utilizando el muestreo de composición a partir de la muestra (5.21). Para ello utilizamos el siguiente resultado intermedio:

Proposición 5.1

Sea M el modelo (2.1) para un valor dado del vector de parámetros θ .

Sea i la posición que ocuparía en el vector de observaciones la transacción cuyo precio se quiere predecir.

Sea $Y_{i,ret} = \log(P_{i,ret})$ donde $P_{i,ret}$ es el precio de dicha transacción

Se tiene que $Y_{i,ret} | \mathbf{Y}, \mathbf{Z}, \mathbf{X}, \theta \sim N(m_i(\theta), s_i^2(\theta))$ con

$$m_i(\theta) = \frac{\tau_i \mu_i + \sum_{j \in F_i} \tau_j f_{ji}(y_j - a_{ji})}{\tau_i^2 + \sum_{j \in F_i} \tau_j f_{ji}^2}, \quad s_i^2(\theta) = \frac{\sigma^2}{\tau_i^2 + \sum_{j \in F_i} \tau_j f_{ji}^2} \quad (5.23)$$

donde $F_i = NT_i \cup NS_i \cup NTS_i \cup NST_i$ con

$$NT_i = \{i+1, \dots, i+m_T\},$$

$$NS_i = \{j > i: \ell_{ji} \in \{1, \dots, m_S\}\},$$

$$NTS_i = \bigcup_{k \in NS_i} NT_k$$

$$NST_i = \bigcup_{k \in NT_i} NS_k$$

donde ℓ_{ji} denota el ranking de la distancia d_{ji} entre la transacción j -ésima y la transacción i dentro del conjunto de distancias $\{d_{jk}: k < j\}$.

$$f_{ji} = b_{ji} + c_{ji} + d_{ji} + h_{ji} \text{ con}$$

$$b_{ji} = (\phi_T + \phi_{TX})t_{ji}(\gamma)$$

$$c_{ji} = (\phi_S + \phi_{SX})s_{ji}(\lambda)$$

$$d_{ji} = (\phi_{TS} + \phi_{TSX}) \left(\sum_{k \in B_j \wedge i \in C_k} t_{jk}(\gamma) s_{ki}(\lambda) \right)$$

$$h_{ji} = (\phi_{ST} + \phi_{STX}) \left(\sum_{k \in C_j \wedge i \in B_k} s_{jk}(\lambda) t_{ki}(\gamma) \right)$$

$$a_{ji} = \mu_j - f_{ji} Y_{i,rep}$$

$$\mu_j = \begin{cases} E[Y_{i,ret} | Y_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \boldsymbol{\theta}] & \text{si } j = i \\ E[Y_j | \{y_k; k < i, Y_{i,ret}, y_k; i < k < j\}, \mathbf{Z}_i, \mathbf{X}_i, \boldsymbol{\theta}] & \text{si } j > i \end{cases}$$

Demostración:

Sean $\mathbf{Y}_< = \{y_j; j < i\}$ e $\mathbf{Y}_> = \{y_j; j > i\}$. Sean $\mathbf{X}_< = (\mathbf{x}_j; j < i)$ y $\mathbf{X}_> = (\mathbf{x}_j; j > i)$.

De (2.1) se tiene que:

$$\begin{aligned} [Y_{i,ret} | \mathbf{Y}, \mathbf{Z}, \mathbf{X}, \boldsymbol{\theta}] &\propto [Y_{i,ret} | \mathbf{Y}_<, \mathbf{Z}_<, \mathbf{X}_<, \boldsymbol{\theta}] [Y_{i,ret} | \mathbf{Y}_>, \mathbf{Y}_<, Y_{i,ret}, \mathbf{Z}, \mathbf{X}, \boldsymbol{\theta}] \\ &\propto \exp\left[-\frac{\tau}{2} \tau_i (Y_{i,ret} - \mu_i)^2\right] \exp\left[-\frac{\tau}{2} \sum_{j \in F_i} \tau_j (y_j - \mu_j)^2\right] = \\ &= \exp\left[-\frac{\tau}{2} \left\{ \tau_i (Y_{i,ret} - \mu_i)^2 + \sum_{j \in F_i} \tau_j (y_j - a_{ji} - f_{ji} Y_{i,ret})^2 \right\}\right] \propto \\ &\propto \exp\left[-\frac{\tau}{2} \left\{ Y_{i,ret}^2 \left(\tau_i + \sum_{j \in F_i} \tau_j f_{ji}^2 \right) - 2Y_{i,ret} \left(\tau_i \mu_i + \sum_{j \in F_i} \tau_j f_{ji} (y_j - a_{ji}) \right) \right\}\right] \propto \\ &\propto \exp\left[-\frac{1}{2} S_i^2(\boldsymbol{\theta}) (Y_{i,ret} - m_i(\boldsymbol{\theta}))^2\right] \end{aligned}$$

de donde se sigue la tesis.

Utilizando este resultado, la muestra (5.21) obtenida de la distribución a posteriori $\boldsymbol{\theta} | \mathbf{Y}, \mathbf{Z}, \mathbf{X}$ mediante el algoritmo 5.3.2 y el muestreo de composición, es posible calcular la distribución $Y_{i,ret} | \mathbf{Y}, \mathbf{Z}, \mathbf{X}$.

Así, por ejemplo, una estimación de la densidad $[Y_{i,\text{ret}} | \mathbf{Y}, \mathbf{Z}, \mathbf{X}]$ se calcularía de la siguiente manera:

$$\frac{1}{n_{\text{muestra}}} \sum_{j=1}^{n_{\text{muestra}}} [Y_{i,\text{ret}} | m_i(\boldsymbol{\theta}^{(j)}), s_i^2(\boldsymbol{\theta}^{(j)})] \quad (5.24)$$

donde $Y_{i,\text{ret}} | m_i(\boldsymbol{\theta}^{(j)}), s_i^2(\boldsymbol{\theta}^{(j)}) \sim N(m_i(\boldsymbol{\theta}^{(j)}), s_i^2(\boldsymbol{\theta}^{(j)}))$ con $m_i(\boldsymbol{\theta}^{(j)})$ y $s_i^2(\boldsymbol{\theta}^{(j)})$ vienen dados por las expresiones (5.23) calculadas en $\boldsymbol{\theta} = \boldsymbol{\theta}^{(j)}$; $j=1, \dots, n_{\text{muestra}}$.

Del mismo modo, estimadores de $E[Y_{i,\text{ret}} | \mathbf{Y}, \mathbf{Z}, \mathbf{X}]$ y de $V[Y_{i,\text{ret}} | \mathbf{X}, \mathbf{Y}]$ serían:

$$\hat{E}[Y_{i,\text{ret}} | \mathbf{Y}, \mathbf{Z}, \mathbf{X}] = \frac{1}{n_{\text{muestra}}} \sum_{j=1}^{n_{\text{muestra}}} m_i(\boldsymbol{\theta}^{(j)}) \quad (5.25)$$

$$\begin{aligned} \hat{V}[Y_{i,\text{ret}} | \mathbf{Y}, \mathbf{Z}, \mathbf{X}] &= \\ &= \frac{1}{n_{\text{muestra}}} \sum_{j=1}^{n_{\text{muestra}}} s_i^2(\boldsymbol{\theta}^{(j)}) + \frac{1}{n_{\text{muestra}}} \sum_{j=1}^{n_{\text{muestra}}} (m_i(\boldsymbol{\theta}^{(j)}) - E[Y_{i,\text{ret}} | \mathbf{Y}, \mathbf{Z}, \mathbf{X}])^2 \end{aligned} \quad (5.26)$$

Finalmente el siguiente algoritmo obtendría una muestra de la distribución $Y_{i,\text{ret}} | \mathbf{Y}, \mathbf{Z}, \mathbf{X}$, aplicando el muestreo de composición:

5.5.1.1 Algoritmo

Paso 1: Aplicar el algoritmo 5.3.2 para obtener una muestra $\{\boldsymbol{\theta}^{(i)}; i=1, \dots, n_{\text{muestra}}\}$ de la distribución $\boldsymbol{\theta} | \mathbf{Y}, \mathbf{Z}, \mathbf{X}$

Paso 2: Para $j=1, \dots, n_{\text{muestra}}$ extraer $Y_{i,\text{ret}}^{(j)}$ de $N(m_i(\boldsymbol{\theta}^{(j)}), s_i^2(\boldsymbol{\theta}^{(j)}))$

La muestra finalmente obtenida sería $\{Y_{i,\text{ret}}^{(j)}; j=1, \dots, n_{\text{muestra}}\}$ que podría utilizarse para calcular estimaciones retrospectivas del nivel de precios de una transacción utilizando la media o la mediana, o para calcular intervalos

bayesianos de un nivel de credibilidad a posteriori dado, a partir de los correspondientes cuantiles.

5.5.2 Elaboración de índices retrospectivos de la evolución de los precios

En la medida en que los modelos hedónicos de regresión ajustan el precio de un bien por cambios de calidad en sus características, son unos candidatos excelentes para la construcción de índices de precios. Una aplicación directa de la metodología presentada en el apartado anterior es la elaboración retrospectiva de índices de precios.

En nuestro caso, construimos una serie de estos índices retrospectivos de precios de viviendas para el área y el periodo de tiempo considerados en la base de datos que se ha presentado en el capítulo 2.

5.5.2.1 Metodología

Sean $\{1 < t_1 < \dots < t_m < T\}$ los periodos en los que se va a calcular los valores de los índices de precios. Sean $1 \leq n_1 < n_2 < \dots < n_m \leq n$ el orden que ocuparían dichas transacciones en el vector \mathbf{Y} de log-precios observados en la muestra.

Se eligen, además, una muestra de viviendas que conforman la “cesta” de viviendas cuyas características $\{(z'_i, x'_i); i = 1, \dots, r\}$ se van a utilizar para llevar a cabo la construcción del índice de precios.

Utilizando el algoritmo descrito en 5.5.1.1 se obtiene, para cada combinación de tiempo t_k y vivienda de la cesta, (z'_i, x'_i) una muestra

$$\{p_{i,k}^{(j)}; j=1, \dots, n_{\text{muestra}}\}; i=1, \dots, r; k=1, \dots, m$$

de la distribución retrospectiva

$$p_{i,k} | \mathbf{Y}, \mathbf{Z} \cup \{\mathbf{z}_i\}, \mathbf{X} \cup \{\mathbf{x}_i\} \sim \int [p_{i,k} | \mathbf{Y}, \mathbf{Z} \cup \{\mathbf{z}_i\}, \mathbf{X} \cup \{\mathbf{x}_i\}, \boldsymbol{\theta}] [\boldsymbol{\theta} | \mathbf{Y}, \mathbf{Z}, \mathbf{X}] d\boldsymbol{\theta} \quad (5.27)$$

con $i=1, \dots, r$; $k=1, \dots, m$ y donde $p_{i,k}^{(j)} = \exp(Y_{i,k}^{(j)})$.

Si t_1 es el periodo de referencia se calcula el valor del índice de precios IP en los periodos $\{t_k; k=1, \dots, r\}$ mediante las expresiones

$$\left\{ \text{IP}_{t_1}^{t_k} = 100 \frac{V_k}{V_1}; k = 1, \dots, r \right\} \quad (5.28)$$

donde $V_k = \sum_{i=1}^r \text{Mediana}(p_{i,k} | \mathbf{Y}, \mathbf{Z} \cup \{\mathbf{z}_i\}, \mathbf{X} \cup \{\mathbf{x}_i\})$ es el valor estimado de la “cesta” de viviendas en el periodo t_k obtenido a partir de las medianas a posteriori de las distribuciones (5.27). Utilizamos la mediana en lugar de la media a posteriori de la distribución de los precios con el fin de evitar la influencia de la elevada asimetría que habitualmente tienen dichas distribuciones la cual puede distorsionar el papel representativo del índice de precios elaborado.

5.5.2.2 Aplicación

En la Figura 5.24 se muestra la evolución trimestral estimada retrospectivamente de los precios escriturados de la vivienda durante los años 2003 y 2004, en el área de estudio considerada. La estimación se realiza mediante un número índice con base 100 en el primer trimestre de 2003 y que se construye tomando como muestra representativa del conjunto de viviendas transaccionadas, las 57 primeras transacciones de la base de datos, que tuvieron lugar en Noviembre y Diciembre del año 2002.

Utilizando esta muestra se predice² retrospectivamente el precio al que se hubiera escriturado cada una de dichas viviendas si la transacción hubiera tenido lugar a mediados del trimestre correspondiente³. La predicción se basa en la distribución predictiva a posteriori dada por la expresión (5.27) y se realiza mediante la mediana a posteriori de la misma. Su cálculo se lleva a cabo aplicando el método de Monte Carlo utilizando el muestreo de composición descrito en 5.5.1.1 con 1000 simulaciones. Así mismo, y con fines comparativos, se muestra la estimación realizada por el Ministerio de la Vivienda para Zaragoza capital en dicho periodo junto con las estimaciones de los modelos considerados en el capítulo.

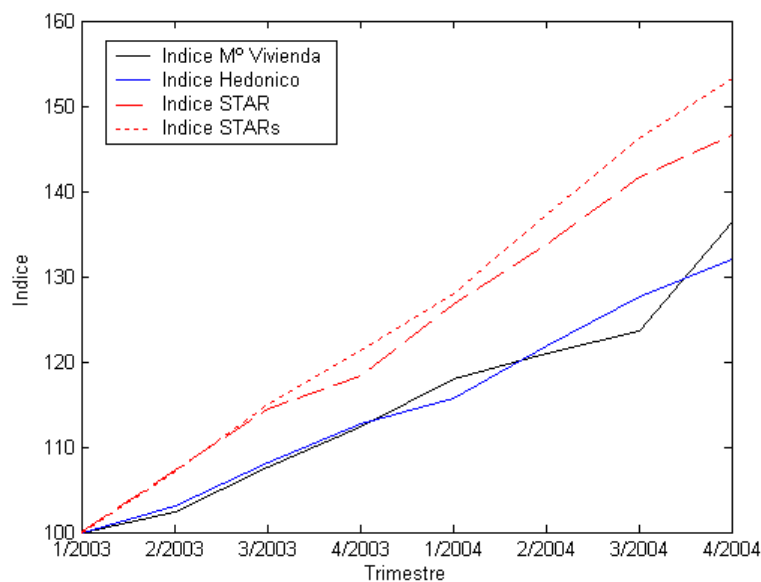


Figura 5.24: Estimación retrospectiva de la evolución del Índice de precio de la vivienda durante 2003 y 2004 (Base 100= 1^{er} trimestre de 2003)

² Este problema recibe el nombre de *kriging* en la literatura de Estadística Espacial

³ Las fechas concretas utilizadas para cada trimestre fueron 14 de Febrero, 15 de Mayo, 15 de Agosto y 15 de Noviembre de los años 2003 y 2004

Se observa, en primer lugar, que durante los dos años 2003 y 2004 las estimaciones de todos los modelos siguen una clara tendencia alcista, al igual que la elaborada por el Ministerio de la Vivienda. Ésta cifra un crecimiento en el periodo considerado de alrededor de un 4% trimestral, que coincide con la estimada por el modelo hedónico, en términos generales. Por su parte los modelos STAR (que coinciden esencialmente en las estimaciones realizadas) lo valoran por encima del 6%, trasladando al precio la influencia de tendencias crecientes de carácter local capturadas a través de efectos espaciales, que provocan un aumento, de forma gradual y constante de las distancias que les separan de la serie oficial (ver Figura 5.24).

5.6 Conclusiones

En este capítulo se ha presentado un procedimiento de estimación bayesiana robusta de los parámetros del modelo STAR con efectos de vecindad analizado a lo largo de la Memoria. Para ello se ha propuesto un modelo con errores condicionalmente heterocedásticos y con una distribución *t* de Student que, al tener colas más pesadas que la distribución normal, disminuye la posible influencia en el proceso de estimación de los parámetros, ejercida por observaciones atípicas anormalmente grandes. La metodología propuesta permite, a la vez, identificar dichas observaciones lo que puede redundar en un proceso de modelización con un mayor grado de ajuste a los datos. Además, se han extendido a este contexto los procesos de validación y simplificación de modelos realizados en los capítulos previos.

Los resultados obtenidos en la aplicación a la base de datos, reafirman que las variables hedónicas con una influencia más significativa son: superficie, antigüedad, tipo de adquisición, portero y ascensor, así como los la

cercanía al centro de la ciudad y el indicador de tiempo. Sin embargo, dejan de ser significativas las variables calefacción colectiva, garaje y la segunda componente espacial, cuyos efectos, débilmente significativos en anteriores modelos, estaban fundamentados en pocas observaciones.

Así mismo, se aprecia la existencia de dependencias espaciales de carácter local significativas no observándose, sin embargo, la de dependencias temporales.

Los efectos tienen mayor precisión que las estimaciones de los modelos en capítulos anteriores, debido a la disminución de la influencia ejercida por las observaciones atípicas.

La distribución del error es muy leptocúrtica por la presencia de atípicos negativos, correspondiendo en su mayor parte, a transacciones cuyo tipo de adquisición no fue una compraventa y con un nivel de precios anormalmente bajo.

Los residuos muestran, a nivel intramuestral, una mayor capacidad de ajuste del modelo confirmando la utilidad de las modificaciones incluidas en el capítulo. Además, a nivel extramuestral, el comportamiento predictivo del modelo resulta de una mejor calidad a tenor de los valores obtenidos en los criterios, con el hallazgo de que tras la validación de tipo *rolling* el modelo hedónico es el que presenta un mejor proceder. Para estudiar las posibles causas, se ha llevado a cabo la monitorización de la dinámica seguida por el criterio LPRED para todos los modelos considerados en los últimos tres capítulos de esta Memoria, poniéndose al descubierto la existencia de un grupo de observaciones, con precios anormalmente bajos, que producían un gran descenso del criterio en todos los modelos comparados. Este hecho fue asumido de forma menos adecuada por los modelos normales de los capítulos 3 y 4, mostrando una mejor adaptación los modelos con errores *t* de Student.

Dentro de ellos, la presencia de todas las variables junto con la ignorancia de los efectos espacio-temporales parece haber sido la clave.

Por último, se ha presentado un método bayesiano de elaboración de predicciones retrospectivas que permite la imputación de precio a viviendas cuya transacción tuvo lugar en el pasado, problema de gran interés a efectos periciales, tanto en ámbitos fiscales como judiciales.

Utilizando dicho procedimiento se ha desarrollado una aplicación para la construcción retrospectiva de índices de precios de vivienda. A efectos comparativos se muestra el índice de precios calculado por el Ministerio de la Vivienda, coincidiendo todos los modelos en un ritmo claramente alcista, pero inferior en el caso de los datos oficiales, al no tener en cuenta la influencia de los efectos de vecindad en la evolución de dichos precios.

Capítulo 6

Selección de variables en modelos STAR con efectos de vecindad mediante algoritmos genéticos (AG)

6.1 Introducción

El proceso de estimación adoptado en el capítulo 2, aunque sencillo computacionalmente, tiene el inconveniente de ignorar de forma general, la incertidumbre asociada a los procesos de selección y estimación del modelo, y en particular, a los parámetros de vecindad que se tratan como dados. En los capítulos 3 a 5 de esta Memoria se ha solventado esta cuestión utilizando metodología bayesiana. Sin embargo, este enfoque presenta la dificultad de ser demasiado complejo desde el punto de vista de su computación, no incorporando además, la incertidumbre en el proceso de selección de variables.

En este capítulo buscamos realizar un tratamiento de selección de las variables hedónicas y/o de los efectos espacio-temporales del modelo STAR analizado a lo largo de la Memoria, dando por supuesto que no todas las variables y efectos de vecindad que aparecen incluidos en la ecuación del

modelo (2.1) han de estar presentes. Este modelo es complejo con un elevado número de parámetros y la permanencia de algunos de ellos que no resultan significativos, puede provocar distorsiones en la estimación del resto, bien por la existencia de multicolinealidad entre las variables hedónicas o entre los efectos de vecindad presentes en el modelo, bien por la pérdida de eficiencia en la estimación debida a la disminución de la ratio del número de observaciones por parámetro. A estos inconvenientes se añaden, además, los problemas de interpretación de los resultados, aumentando el riesgo de detectar patrones específicos de la muestra objeto de estudio. El método que se va a proponer, incorpora además, la incertidumbre asociada al proceso de selección de modelos que, como ya se ha mencionado, se había ignorado en los capítulos precedentes.

El procedimiento para la selección de variables debe realizarse de forma cuidadosa, siguiendo unos criterios previamente elegidos, de modo que, tras la valoración que hagan del modelo, determine que variables y/o efectos espacio-temporales deberán formar parte del modelo. Al respecto hay que señalar que no existe un criterio universalmente aceptado en la literatura para seleccionar variables. La novedad del procedimiento propuesto en este capítulo es confiar esta tarea a más de uno de ellos, decisión provocada por la firme convicción, patente en todos los capítulos anteriores, de que una validación correcta del modelo, exige explorar aspectos como su bondad del ajuste a los datos y su capacidad predictiva.

La mayor parte de los criterios propuestos en la literatura son una combinación de una medida del grado de ajuste del modelo a los datos, y una medida de parsimonia que penaliza por la complejidad del mismo. Ejemplos de este tipo son los llamados *criterios de información* o de *verosimilitud penalizada* entre los que destacan los criterios AIC de Akaike (1973), el BIC

de Schwarz (1978), el criterio HQ de Hannan-Quin (1979) o los criterios PIC y PICF de Phillips (1996) y Phillips y Ploberger (1996). Aznar y otros (2001) dan una visión exhaustiva de este tipo de criterios. Otros autores adoptan un planteamiento bayesiano y utilizan el factor Bayes como criterio de selección (George y MacCulloch (1993), Geweke (1993), Madigan y Raftery (1994), Madigan y otros (1995)). Todo ello abunda en el sentido de que pueden utilizarse una gran variedad de criterios persiguiendo un mismo fin, pero midiendo distintos aspectos y valorando de forma diferente el funcionamiento de un modelo.

Por todo ello y con el fin de incorporar las distintas facetas mencionadas, proponemos una aproximación multicriterio para llevar a cabo el proceso de selección de variables y efectos de vecindad del modelo STAR. Estos criterios calificarán el comportamiento de cada modelo no sólo desde un punto de vista estadístico, analizando su bondad de ajuste a los datos observados, sino también desde un punto de vista más práctico, evaluando su comportamiento predictivo intra y extramuestral. Ello da mayor flexibilidad y realismo al proceso de selección a través de varios criterios al mismo tiempo, lo que plantea un problema que no es analíticamente tratable y que obliga a recurrir a métodos heurísticos, concretamente en nuestro caso a algoritmos genéticos.

A efectos prácticos, y tal y como se ha hecho en los capítulos anteriores, la metodología se ilustra con la aplicación al conjunto de datos presentado en el apartado 2.6.1.

El plan del trabajo es el siguiente: en la sección 6.2 se plantea el problema de selección de variables, se comentan los criterios utilizados en el proceso de selección y se plantea el problema multiobjetivo; en el apartado 6.3 se muestra el algoritmo genético utilizado en la resolución de dicho problema;

en el punto 6.4 se indica como llevar a cabo el proceso de estimación de los parámetros del modelo y se describen los procesos de validación intra y extramuestral. La sección 6.5 aplica la metodología al mercado inmobiliario de la ciudad de Zaragoza y, finalmente, en 6.6 se concluye exponiendo los resultados de mayor interés.

6.2 Planteamiento del problema

En esta sección se describe el problema de selección de variables en modelos de regresión indicando los criterios que se van a utilizar junto con el problema multiobjetivo a resolver.

6.2.1 Selección de variables

La selección de las variables independientes en modelos de regresión, a pesar de ser una cuestión que ha recibido gran atención y ha sido estudiada extensamente en la literatura, es todavía una cuestión abierta. Desde el punto de vista matemático no existe completo acuerdo sobre la definición del problema y puede depender de la aplicación concreta. De forma general, se pueden distinguir dos objetivos, por una parte, descubrir todas las variables relevantes para una situación específica determinando su importancia y la forma en que se relacionan con las demás, y por otra, encontrar un subconjunto mínimo de variables útiles para predecir.

Los problemas de selección de variables se plantean de forma generalizada en todos los ámbitos de la aplicación estadística, no siendo patrimonio exclusivo de un método. Se pueden encontrar en técnicas tan dispares como, por ejemplo, las máquinas de aprendizaje, tanto supervisado como no supervisado, métodos de clasificación, modelos de regresión o en

predicción de series de tiempo, dando una idea del interés que provoca en ámbitos tan diferentes.

La forma de atacar la cuestión de la selección de variables, es diversa. La estrategia más habitual consiste en evaluar cada variable individualmente y seleccionar aquellas que aporten una mayor información. Este sistema, aunque simple computacionalmente, no siempre es el más adecuado, ya que algunos métodos conducen a subconjuntos de características que son altamente inestables bajo mínimas perturbaciones (como la eliminación de algunos ejemplos de entrenamiento), que puede ser indeseable para interpretar los resultados (Xu y Zhang, 2001)

Otra cuestión a tener en cuenta es la elección de los criterios para cuantificar el comportamiento del modelo como ya se ha mencionado en la introducción y que, en nuestro caso, al utilizar varios de forma simultánea, nos lleva a un problema de optimización multiobjetivo.

Finalmente, otro aspecto muy relevante en la selección de variables es el relativo al procedimiento de búsqueda de las variables que deben permanecer en el modelo. Desde un punto de vista teórico la situación más deseable es aquella en la que se analizan todos los posibles subconjuntos de las variables iniciales del modelo o del conjunto de datos. En este caso, la calidad del resultado obtenido no deja lugar a dudas, pero no resulta práctico, ya que en algunos problemas, la necesidad de explorar espacios con decenas e incluso centenares de miles de variables, hace inabordable su consideración desde el punto de vista computacional. Por esta razón, resulta imprescindible encontrar sistemas, que evitando la exploración exhaustiva, puedan aportar resultados que aún no siendo el óptimo global, se encuentren suficientemente próximos (Broadhurst y otros, 1997; Gualdrón, 2006).

Existen propuestas de estrategias de búsqueda, que tratan de paliar los inconvenientes de la búsqueda exhaustiva, de forma que no sea preciso recorrer todo el espacio de modelos como son los sistemas secuenciales y los sistemas aleatorios.

Los sistemas secuenciales (conocidos en la literatura como *stepwise*) trabajan eliminando (*backward*) o incorporando (*forward*) variables al modelo de forma ordenada, utilizando procedimientos de contraste y comparación de modelos. Su principal desventaja radica en el hecho de que puede que no se alcancen óptimos globales al quedar enmarcadas las posibles soluciones alrededor de óptimos locales dando lugar a soluciones sub-óptimas. El mérito más relevante de este tipo de métodos es el gran ahorro en tiempo computacional.

Existe otro grupo de algoritmos, los Algoritmos Evolutivos, que emplean procedimientos aleatorios y son utilizados, principalmente, en problemas con espacios de búsqueda extensos y no lineales, donde otros métodos no son capaces de encontrar soluciones en un tiempo razonable. Están inspirados en la teoría darwiniana de la selección natural, según la cual los individuos más aptos tienen mayor probabilidad de transmitir sus genes a las generaciones posteriores (Goldberg, 1989). Al conjunto de individuos se le denomina población.

Los individuos, en el algoritmo, son los entes (en nuestro caso los diferentes modelos) que representan las soluciones al problema, que son transformados por operadores probabilísticas, llamados operadores genéticos, y sirven para generar nuevas poblaciones de individuos a partir de la población actual. Los principales operadores son: de recombinación, mutación y selección. Los individuos que representan las soluciones más adecuadas al

problema tienen una probabilidad más alta de supervivencia y la población va mejorando paso a paso.

Este proceso lo utilizan las tres principales ramas de los Algoritmos Evolutivos: Estrategias Evolutivas (EEs), Algoritmos Genéticos (AGs) y Programación Evolutiva (PE).

Dada la naturaleza de nuestros indicadores de selección de variables en este capítulo de la Memoria utilizamos un algoritmo genético que es la técnica que mejor se adapta a este contexto.

6.2.2 El modelo

Comenzamos planteando el problema de la selección de variables para el modelo STAR con efectos de vecindad (2.1). El procedimiento lleva implícita una selección de modelos de la forma $\mathbf{M}_{\text{ind}_Z, \text{ind}_X, \text{ind}_\phi, \theta_V}$ que vendrá dado por la expresión:

$$\begin{aligned} \mathbf{Y} = & \mathbf{Z}\mathbf{D}_{\text{ind}_Z} \boldsymbol{\alpha} + \mathbf{X}\mathbf{D}_{\text{ind}_X} \boldsymbol{\beta} + \text{ind}_T \phi_T \mathbf{T}(\mathbf{Y} - \mathbf{X}\mathbf{D}_{\text{ind}_X} \boldsymbol{\beta}) + \text{ind}_S \phi_S \mathbf{S}(\mathbf{Y} - \mathbf{X}\mathbf{D}_{\text{ind}_X} \boldsymbol{\beta}) + \\ & + \text{ind}_{\text{ST}} \phi_{\text{ST}} \mathbf{ST}(\mathbf{Y} - \mathbf{X}\mathbf{D}_{\text{ind}_X} \boldsymbol{\beta}) + \text{ind}_{\text{TS}} \phi_{\text{TS}} \mathbf{TS}(\mathbf{Y} - \mathbf{X}\mathbf{D}_{\text{ind}_X} \boldsymbol{\beta}) + \boldsymbol{\varepsilon} \end{aligned} \quad (6.1)$$

donde:

$\boldsymbol{\varepsilon} \sim N_n(\mathbf{0}, \sigma^2 \mathbf{I}_n)$, $\boldsymbol{\theta}_V = (m_T, \gamma, m_S, \lambda)'$ son los parámetros de vecindad del modelo (2.1), \mathbf{ind}_Z y \mathbf{ind}_X son vectores $q \times 1$, $k \times 1$, respectivamente, cuyas componentes toman valores iguales a 0 ó 1 e indican qué variables de \mathbf{Z} y \mathbf{X} deben ser incluidas (valor 1) o no (valor 0) en el modelo y $\mathbf{D}_{\text{ind}_Z} = \text{diag}(\mathbf{ind}_Z)$, $\mathbf{D}_{\text{ind}_X} = \text{diag}(\mathbf{ind}_X)$. Así mismo $\mathbf{ind}_\phi = (\text{ind}_T, \text{ind}_S, \text{ind}_{\text{ST}}, \text{ind}_{\text{TS}})'$ es un vector de indicadores cuyas componentes $\{\text{ind}_i; i \in \{T, S, \text{ST}, \text{TS}\}\}$ toman valores en $\{0, 1\}$

para señalar qué efectos ϕ_T , ϕ_S , ϕ_{ST} y ϕ_{TS} deben ser incluidos o no en el modelo.

6.2.3. Criterios de selección

El proceso de selección debe dar como resultado aquéllos modelos $\mathbf{M}_{\text{ind}_Z, \text{ind}_X, \text{ind}_0, \theta_v}$ que mejor se ajusten a los datos observados. Lo habitual será que se obtenga un conjunto de modelos reflejando, de esta forma, la incertidumbre asociada al proceso de estimación y selección llevado a cabo (ver, por ejemplo, George y MacCulloch (1993), Geweke (1993) o Madigan y Raftery (1994)).

Tal y como se ha comentado en la introducción, los criterios que proponemos para llevar a cabo la selección de variables intentan evaluar el comportamiento del modelo desde diferentes ópticas: bondad de ajuste, complejidad y comportamiento predictivo intra y extramuestral. De esta forma se incorporan no sólo los aspectos más estadísticos relacionados con el ajuste del modelo, sino también aspectos más prácticos relacionados con su comportamiento predictivo.

En nuestro caso, como criterio de ajuste a los datos utilizamos el BIC de Schwarz (1978) dado en la expresión (2.31). Este criterio es un típico ejemplo de combinación de ajuste y parsimonia donde, como ya se ha mencionado en el capítulo 2, el primer término evalúa el grado de ajuste del modelo a los datos observados mientras que el segundo penaliza por su falta de parsimonia de forma que, cuanto menor sea el valor del criterio BIC mejor es el grado de ajuste del modelo a los datos.

Sin embargo, utilizar el criterio BIC como único criterio de selección tiene el peligro de considerar únicamente la significación estadística del

modelo para explicar los datos observados, sin considerar otros aspectos más prácticos del mismo como puede ser su capacidad predictiva. Además, penaliza todos los parámetros del modelo por igual sin tener en cuenta cuáles de ellos están mejor estimados y cuáles no, lo cual puede redundar en una mayor incertidumbre en la elaboración de predicciones. Por dicha razón, consideramos otros criterios que evalúen dicho aspecto buscando determinar aquéllos modelos que muestren un buen ajuste a los datos y un buen comportamiento predictivo intra y extramuestral. Para ello consideramos dos criterios adicionales, presentados también en la sección 2.5 del capítulo 2: la pérdida predictiva cuadrática (SPL) y la densidad predictiva extramuestral (LPRED).

El criterio pérdida predictiva cuadrática (ver sección 2.5) viene dado por:

$$\text{SPL}(\mathbf{M}) = \sum_{i=n_0+1}^n \left(y_i - E \left[y_i \mid \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \hat{\boldsymbol{\theta}}_M, \mathbf{M} \right] \right)^2 + \sum_{i=n_0+1}^n \text{Var} \left(y_i \mid \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \hat{\boldsymbol{\theta}}_M, \mathbf{M} \right) \quad (6.2)$$

donde $\mathbf{Y}_{i-1} = (y_1, \dots, y_{i-1})'$, $\mathbf{Z}_i = (z_1, \dots, z_i)'$, $\mathbf{X}_i = (x_1, \dots, x_i)'$

$$\begin{aligned} E \left[y_i \mid \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \hat{\boldsymbol{\theta}}_M, \mathbf{M} \right] = & \\ & \mathbf{z}_i' \mathbf{D}_{\text{indZ}} \hat{\boldsymbol{\alpha}} + \mathbf{x}_i' \mathbf{D}_{\text{indX}} \hat{\boldsymbol{\beta}} + \mathbf{u}_i' \left(\text{ind}_T \hat{\phi}_T \mathbf{T} (\mathbf{Y} - \mathbf{X} \mathbf{D}_{\text{indX}} \hat{\boldsymbol{\beta}}) + \text{ind}_S \hat{\phi}_S \mathbf{S} (\mathbf{Y} - \mathbf{X} \mathbf{D}_{\text{indX}} \hat{\boldsymbol{\beta}}) + \right. \\ & \left. + \text{ind}_{\text{ST}} \hat{\phi}_{\text{ST}} \mathbf{ST} (\mathbf{Y} - \mathbf{X} \mathbf{D}_{\text{indX}} \hat{\boldsymbol{\beta}}) + \text{ind}_{\text{TS}} \hat{\phi}_{\text{TS}} \mathbf{TS} (\mathbf{Y} - \mathbf{X} \mathbf{D}_{\text{indX}} \hat{\boldsymbol{\beta}}) \right) + \boldsymbol{\varepsilon} \quad (6.3) \end{aligned}$$

$$\begin{aligned} \text{Var} \left(y_i \mid \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \hat{\boldsymbol{\theta}}_M, \mathbf{M} \right) \approx & \\ \approx \left(\frac{\partial}{\partial \boldsymbol{\theta}_M} E \left[y_i \mid \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \hat{\boldsymbol{\theta}}_M, \mathbf{M} \right] \right)' \text{Var} \left(\hat{\boldsymbol{\theta}}_M \right) \frac{\partial}{\partial \boldsymbol{\theta}_M} E \left[y_i \mid \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \hat{\boldsymbol{\theta}}_M, \mathbf{M} \right] & (6.4) \end{aligned}$$

donde $\text{Var}(\hat{\boldsymbol{\theta}}_M) \approx \left(\left(\frac{\partial \ell(\hat{\boldsymbol{\theta}}_M | \mathbf{Y}, \mathbf{X}, \mathbf{Z}, \mathbf{M})}{\partial \boldsymbol{\theta}_M} \right)' \left(\frac{\partial \ell(\hat{\boldsymbol{\theta}}_M | \mathbf{Y}, \mathbf{X}, \mathbf{Z}, \mathbf{M})}{\partial \boldsymbol{\theta}_M} \right) \right)^{-1}$ y donde se toma

igual a 0 el valor del estimador máximo-verosímil de los efectos no incluidos en el modelo, es decir, de aquellos cuyo indicador de inclusión toma el valor 0. Este criterio, como ya se mencionó en el capítulo 2, está basado en Gelfand y Ghosh (1998) y busca evaluar el comportamiento predictivo extramuestral del modelo en réplicas de los datos observados de forma que, cuanto menor es su valor, mejor es dicho comportamiento. El primer término de (6.2) evalúa el ajuste de las predicciones a los datos mientras que el segundo está relacionado de forma directa con el grado de complejidad del modelo teniendo en cuenta el hecho de que no todos los parámetros del mismo se estiman con el mismo grado de acuracidad, lo cual puede manifestarse en un menor o mayor grado de variabilidad de la distribución predictiva.

El criterio SPL, sin embargo, hace un doble uso de los datos como ya sabemos, lo cual puede llevar a problemas de sobreajuste. Por tanto, en este proceso de selección de variables utilizamos un criterio adicional que también se ha empleado en los capítulos anteriores y que no tiene este problema. Dicho criterio es la densidad predictiva extramuestral LPRED cuyo valor viene dado por:

$$\text{LPRED}(\mathbf{M}) = \sum_{i=n_{\text{est}}+1}^n \log(f(y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \mathbf{M})) \quad 1 \leq n_{\text{est}} < n \quad (6.5)$$

con

$$y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \mathbf{M} \approx N\left(E(y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \hat{\boldsymbol{\theta}}_{M, n_{\text{est}}}, \mathbf{M}), \text{Var}(y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \hat{\boldsymbol{\theta}}_{M, n_{\text{est}}}, \mathbf{M})\right)$$

donde $E(y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \hat{\boldsymbol{\theta}}_{M, n_{\text{est}}}, \mathbf{M})$ y $\text{Var}(y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \hat{\boldsymbol{\theta}}_{M, n_{\text{est}}}, \mathbf{M})$ se calculan mediante las expresiones (6.3) y (6.4) y $\hat{\boldsymbol{\theta}}_{M, n_{\text{est}}}$ denota el estimador máximo-

El problema multiobjetivo

verosímil de θ_M obtenido con los datos $\{y_1, \dots, y_{n_{est}}\}$. La expresión (6.5) es una aproximación de la densidad predictiva extramuestral de la muestra de validación $\{y_{n_{est}+1}, \dots, y_n\}$ utilizando la información proporcionada por dicha muestra $\{y_1, \dots, y_{n_{est}}\}$ de forma que, cuanto mayor es su valor, mejor es el comportamiento predictivo del modelo al hacer más verosímil la muestra observada en el periodo de validación. Este criterio evalúa el comportamiento predictivo extramuestral del modelo teniendo en cuenta no sólo el comportamiento en media de la predicción recogido por la esperanza condicionada $E(y_i | Y_{i-1}, Z_i, X_i, \hat{\theta}_{M, n_{est}}, M)$, sino también el grado de acuracidad de las predicciones elaboradas recogido por el término $Var(y_i | Y_{i-1}, Z_i, X_i, \hat{\theta}_{M, n_{est}}, M)$ que, como se ha comentado con anterioridad, está habitualmente relacionado con el grado de complejidad del modelo.

El criterio LPRED tiene menos riesgo de *sobreajuste* que el criterio SPL al evaluar el comportamiento predictivo del modelo en datos que son diferentes de los utilizados para estimarlo. Depende, sin embargo, del valor de n_{est} , es decir, de los tamaños de los periodos elegidos para estimar y validar el modelo para los cuales no existe unanimidad en la literatura. Es, por esta razón, por la que no lo adoptamos como único criterio de evaluación del comportamiento predictivo del modelo utilizando, además, el criterio SPL que no tiene este problema.

6.3 El problema multiobjetivo

Con respecto a los criterios utilizados para llevar a cabo la selección de variables, el hecho de que se utilicen de forma simultánea más de uno, como

ya se ha indicado anteriormente, conduce a un problema de optimización multiobjetivo.

Los problemas de optimización multiobjetivo se caracterizan por una familia de alternativas que deben ser consideradas equivalentes en la ausencia de información sobre la relevancia de un objetivo sobre los otros. Cuanto mayor es el número de objetivos que compiten, más complejo resulta encontrar una solución satisfactoria.

En un problema multiobjetivo general las soluciones están formadas por todas las soluciones factibles del espacio tales que las componentes de las funciones objetivo no pueden ser simultáneamente mejoradas. Estas soluciones se llaman Pareto óptimas o eficientes. Este conjunto suele constar generalmente, de muchas soluciones, alguna de las cuales se suele seleccionar como decisión final.

En nuestro problema concreto, una vez presentados los criterios utilizados en la selección, se hace necesario desarrollar un método para encontrar los modelos que tengan un buen comportamiento con respecto a los mismos. Nuestra propuesta consiste en seleccionar los que sean eficientes en un sentido Pareto, con respecto a algunos de los criterios BIC, SPL y LPRED. A partir de dicha selección y, en un espíritu similar al contexto bayesiano usado en Madigan y Raftery (1994) o Madigan y otros (1995) tomar como modelo final una mixtura que refleje el grado de compatibilidad con los datos de cada una de sus componentes.

Con tal fin planteamos el siguiente problema de optimización multiobjetivo:

$$\text{Min}_{(\text{ind}_Z, \text{ind}_X, \text{ind}_\phi, \theta_v) \in I} \left\{ \text{BIC}(\mathbf{M}_{\text{ind}_Z, \text{ind}_X, \text{ind}_\phi, \theta_v}), \text{SPL}(\mathbf{M}_{\text{ind}_Z, \text{ind}_X, \text{ind}_\phi, \theta_v}), \text{LPRED}(\mathbf{M}_{\text{ind}_Z, \text{ind}_X, \text{ind}_\phi, \theta_v}) \right\} \quad (6.6)$$

Algoritmo genético

donde:

$I = \{0,1\}^{q+k+4} \times \mathbf{m}_{T,\text{red}} \times \boldsymbol{\gamma}_{\text{red}} \times \mathbf{m}_{S,\text{red}} \times \boldsymbol{\lambda}_{\text{red}}$ siendo

$\mathbf{m}_{T,\text{red}} = \{1 \leq m_{T,1} < m_{T,2} < \dots < m_{T,p_T}\}$ con $m_{T,i} \in \mathbf{N}$; $i=1, \dots, p_T = 2^{q_T}$

$\boldsymbol{\gamma}_{\text{red}} = \{1 \leq \gamma_1 < \gamma_2 < \dots < \gamma_{p_\gamma}\}$ con $\gamma_i \in (0,1)$; $i=1, \dots, p_\gamma = 2^{q_\gamma}$

$\mathbf{m}_{S,\text{red}} = \{1 \leq m_{S,1} < m_{S,2} < \dots < m_{S,p_S}\}$ con $m_{S,i} \in \mathbf{N}$; $i=1, \dots, p_S = 2^{q_S}$

$\boldsymbol{\lambda}_{\text{red}} = \{1 \leq \lambda_1 < \lambda_2 < \dots < \lambda_{p_\lambda}\}$ con $\lambda_i \in (0,1)$; $i=1, \dots, p_\lambda = 2^{q_\lambda}$

un conjunto de redes de valores posibles de las componentes del vector de parámetros de vecindad $\boldsymbol{\theta}_v$.

El problema (6.6) no es tratable analíticamente por lo que recurrimos a métodos heurísticos y, más concretamente, a algoritmos genéticos, para resolverlo. La siguiente sección detalla el algoritmo genético utilizado.

6.4 Algoritmo genético

Los Algoritmos Evolutivos son métodos de búsqueda y optimización basados en los modelos de evolución orgánica, con la naturaleza como fuente de inspiración. Modelan el proceso de aprendizaje colectivo de una población de individuos, cada uno de los cuales representa no sólo un punto de búsqueda en el espacio de soluciones potenciales de un problema dado, sino que también puede ser un contenedor temporal del conocimiento actual sobre las leyes del entorno. La población de partida se inicializa mediante un método dependiente del algoritmo y evoluciona hacia regiones cada vez mejores del espacio de búsqueda por medio de procesos más o menos aleatorizados de recombinación, mutación y selección. La recombinación consiste en la mezcla

de la información de dos o más individuos; la mutación es un cambio aleatorio en los individuos; y la selección, consistente en la elección de los individuos que sobrevivirán y conformarán la siguiente generación. La función objetivo informa sobre la calidad de los nuevos puntos de búsqueda y el proceso de selección favorece a aquellos individuos con mejores aptitudes para reproducirse más frecuentemente que los individuos peores.

Los Algoritmos Evolutivos son particularmente adecuados para resolver problemas multiobjetivo, ya que tratan de forma simultánea en cada generación un conjunto de posibles soluciones, y no sólo un único candidato. Este hecho permite encontrar varios miembros del conjunto Pareto óptimo en una ejecución del algoritmo, en lugar de tener que realizar una serie de ejecuciones separadas como en el caso de las técnicas de programación matemáticas habituales (ver, para más detalles, Coello y otros. (2007)).

En nuestro caso la población de individuos viene dada por:

$$\mathbf{P} = \{\mathbf{c} = (c_1, \dots, c_Q)' \text{ con } c_i \in \{0, 1\}, Q = q+k+4+q_T+q_\gamma+q_S+q_\lambda\}$$

donde

$$\mathbf{ind}_Z = (c_1, \dots, c_q)$$

$$\mathbf{ind}_X = (c_{q+1}, \dots, c_{q+k})$$

$$\mathbf{ind}_\phi = (c_{q+k+1}, \dots, c_{q+k+4})$$

$(c_{q+k+5}, \dots, c_{q+k+4+q_T})$ es la codificación en binario del lugar que ocupa el valor de $m_{T,i}$ en $\mathbf{m}_{T,red}$.

$(c_{q+k+4+q_T+1}, \dots, c_{q+k+4+q_T+q_\gamma})$ es la codificación en binario del lugar que ocupa el valor de γ_i en $\boldsymbol{\gamma}_{red}$.

Algoritmo genético

$(c_{q+k+4+q_T+q_Y+1}, \dots, c_{q+k+4+q_T+q_Y+q_S})$ es la codificación en binario del lugar que ocupa el valor de $m_{S,i}$ en $\mathbf{m}_{S,red}$.

$(c_{q+k+4+q_T+q_Y+q_S+1}, \dots, c_{q+k+4+q_T+q_Y+q_S+q_\lambda})$ es la codificación en binario del lugar que ocupa el valor de λ_i en λ_{red} .

Pondremos, de ahora en adelante, $\mathbf{M}_c = \mathbf{M}_{ind_z, ind_x, ind_\phi, \theta_v}$.

Una vez definida la población, el algoritmo consta de los siguientes pasos:

Paso 0 (Inicialización)

Se fija $N =$ tamaño de la población, $p_{mut} =$ probabilidad de mutación, $p_{comb} =$ probabilidad de recombinación, $iter_{max} =$ número máximo de iteraciones y $contador_{max} =$ máximo número de repeticiones de la misma solución.

Se selecciona una población inicial $\mathbf{P}^{(0)} = \{ \mathbf{c}_1^{(0)}, \dots, \mathbf{c}_N^{(0)} \}$ con $\mathbf{c}_i^{(0)} = (c_{i,1}^{(0)}, \dots, c_{i,Q}^{(0)})$ y $c_{i,j}^{(0)} \sim \text{Bernoulli}(0.5)$; $j=1, \dots, Q$; $i=1, \dots, N$

Se calcula el valor de los criterios BIC, SPL y LPRED para los modelos $\{ \mathbf{M}_{c_i^{(0)}}; i=1, \dots, N \}$.

Colocar it (número de iteraciones) = 1 y $contador = 0$.

Paso 1 (Recombinación)

Se pone $\mathbf{P}^{(it)} = \mathbf{P}^{(it-1)}$. Se selecciona qué individuos de la población $\mathbf{P}^{(it)}$ se van a recombinar de forma que cada individuo $c_i^{(it)}$ tiene una probabilidad p_{comb} de ser elegido para este procedimiento. Una vez seleccionados los individuos, se desordenan aleatoriamente, se emparejan y se sortea para cada

una de las parejas seleccionadas, $c_i^{(it)}$ y $c_j^{(it)}$, un punto de corte r . Entonces con cada pareja se genera una nueva pareja c_i^* y c_j^* , intercambiando las colas de las soluciones (desde el punto de corte hasta el final) de forma que $c_{i,\ell}^* = c_{j,\ell}^{(it)}$ y $c_{j,\ell}^* = c_{i,\ell}^{(it)}$; $\ell = r+1, \dots, Q$. Los individuos así generados se incorporan a la población $\mathbf{P}^{(it)}$. Una vez incorporados a $\mathbf{P}^{(it)}$ se calcula el valor de los criterios BIC, SPL y LPRED para los modelos $\mathbf{M}_{c_i^*}$ y $\mathbf{M}_{c_j^*}$.

Paso 2 (Mutación)

Para cada componente $c_{i,\ell}^{(it)}$ de cada individuo de la población $\mathbf{P}^{(it)}$ se sorteja si muta con probabilidad p_{mut} . En el caso de que un individuo $c_i^{(it)}$ se decida mutar se crea un nuevo individuo c_i^* tal que $c_{i,\ell}^* = 1 - c_{i,\ell}^{(it)}$ en las componentes ℓ en las que ha habido mutación y $c_{i,\ell}^* = c_{i,\ell}^{(it)}$ en las que no ha habido. Todos los individuos contruidos de esta forma se incorporan a $\mathbf{P}^{(it)}$. Una vez incorporados a $\mathbf{P}^{(it)}$ se calcula el valor de los criterios BIC, SPL y LPRED para los modelos $\left\{ \mathbf{M}_{c_i^*}; i \in \{1, \dots, N\} \text{ tal que } c_i^{(it)} \text{ ha mutado} \right\}$.

Paso 3 (Asignación de rankings Pareto)

Se establece un ranking Pareto entre las soluciones de $\mathbf{P}^{(it)}$ utilizando todos los criterios, asignando el ranking 1 a las soluciones eficientes, es decir, aquéllas para las que no existe ninguna otra en $\mathbf{P}^{(it)}$ que las domine en todos los criterios. Posteriormente se asigna el ranking 2 a aquéllas soluciones de $\mathbf{P}^{(it)}$ que son eficientes una vez eliminadas de la población las soluciones de ranking 1, etc. El proceso se repite hasta que todas ellas tienen asignado un valor de ranking.

Paso 4 (Selección elitista)

Se seleccionan, en primer lugar, las soluciones de ranking 1. Si su número supera el tamaño de la población N se va al Paso 5. En caso contrario se incorporan las de ranking 2, 3, etc. hasta que el tamaño de la población así creada sea mayor o igual que N . Una vez conseguido este objetivo se va al Paso 5.

Paso 5 (Condición de finalización)

Si $\mathbf{P}^{(it)} = \mathbf{P}^{(it-1)}$ poner contador = contador+1. Poner $it = it + 1$. Si contador > contador_{max} o $it > iter_{max}$ finalizar. En caso contrario ir al Paso 1.

6.5 Estimación y validación del modelo

Como resultado del algoritmo se obtiene una población de individuos \mathbf{P} y un conjunto de modelos $\{\mathbf{M}_c; c \in \mathbf{P}\}$ que son eficientes con respecto a todos los criterios. Siguiendo la filosofía bayesiana propuesta por Madigan y Raftery (1994), las inferencias acerca de los parámetros del modelo o la elaboración de predicciones extramuestrales podrían hacerse utilizando la

$$\text{mixtura } \sum_{c \in \mathbf{P}} w_c \mathbf{M}_c \text{ donde } w_c = \frac{\exp\left[-\frac{\text{BIC}(\mathbf{M}_c)}{2}\right]}{\sum_{c' \in \mathbf{P}} \exp\left[-\frac{\text{BIC}(\mathbf{M}_{c'})}{2}\right]} \quad (6.7)$$

En esta sección se describen los procesos de estimación y validación intra y extramuestral de dicha mixtura.

6.5.1 Estimación del modelo

Las inferencias acerca de un parámetro θ cualquiera del modelo STAR (2.1) que sea significativo, podrían hacerse a partir de la media y varianza a posteriori que pueden ser aproximadas por (ver Raftery y otros, 1996)

$$E[\theta | \mathbf{Y}, \mathbf{Z}, \mathbf{X}] \approx \sum_{c \in T} \hat{\theta}_{M_c} w_c \quad (6.8)$$

$$\text{Var}(\theta | \mathbf{Y}, \mathbf{Z}, \mathbf{X}) \approx \sum_{c \in T} (se_{M_c}^2 + \hat{\theta}_{M_c}^2) w_c - (E[\theta | \mathbf{Y}, \mathbf{Z}, \mathbf{X}])^2 \quad (6.9)$$

donde $T = \{c \in \mathbf{P}: c_\ell = 1\}$, ℓ es la componente del vector c correspondiente a θ , $\hat{\theta}_{M_c}$ es el estimador máximo-verosímil de θ utilizando el modelo M_c y $se_{M_c}^2$ es el cuadrado del error estándar de dicha estimación.

Así mismo,

$$\sum_{c \in T} w_c \quad (6.10)$$

podría tomarse como un estimador de la probabilidad a posteriori de que el parámetro θ sea significativamente distinto de cero y por lo tanto, de que el efecto que modela deba incorporarse al modelo STAR.

6.5.2 Validación del modelo

Una vez estimado el modelo, al igual que en los capítulos anteriores se procede a su validación intra y extramuestral. Ambos tipos de validaciones están basadas en el análisis de los residuos no tipificados del modelo dados por $\{\hat{\epsilon}_i = y_i - E[y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \mathbf{M}]; i = n_0 + 1, \dots, n\}$ y/o los residuos tipificados

$$\left\{ \hat{\epsilon}_i = \frac{y_i - E[y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \mathbf{M}]}{\sqrt{\text{Var}(y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \mathbf{M})}}; i = n_0 + 1, \dots, n \right\} \text{ donde } \mathbf{M} \text{ es la mixtura de}$$

modelos obtenida tras la aplicación del algoritmo descrito en la sección 6.3 y

Resultados

$E[y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \mathbf{M}]$ y $\text{Var}(y_i | \mathbf{Y}_{i-1}, \mathbf{Z}_i, \mathbf{X}_i, \mathbf{M})$ vienen dadas por las expresiones (6.11) y (6.12) tomando $\mathbf{z} = \mathbf{z}_i$, $\mathbf{x} = \mathbf{x}_i$ y $\mathbf{D} = (\mathbf{Y}_{i-1}, \mathbf{X}_{i-1}, \mathbf{Z}_{i-1})$.

En la validación intramuestral el modelo $\mathbf{M} = \sum_{c \in \mathbf{P}} w_c \mathbf{M}_c$ donde \mathbf{P} y los pesos $\{w_c; c \in \mathbf{P}\}$ se calcularán utilizando como muestra de estimación los datos correspondientes al periodo $\{1, \dots, n\}$.

Sin embargo, este tipo de validación, puede tener problemas de sobreajuste por lo que conviene complementarla con una validación extramuestral. En ésta última, la muestra observada se dividirá en dos submuestras $\mathbf{D}_{\text{est}} = (\mathbf{Y}_{\text{est}}, \mathbf{Z}_{\text{est}}, \mathbf{X}_{\text{est}})$ y $\mathbf{D}_{\text{val}} = (\mathbf{Y}_{\text{val}}, \mathbf{Z}_{\text{val}}, \mathbf{X}_{\text{val}})$ que corresponden a los datos observados de \mathbf{Y} , \mathbf{Z} y \mathbf{X} en los periodos $\{1, \dots, n_{\text{est}}\}$ (periodo de estimación) y $\{n_{\text{est}+1}, \dots, n\}$ (periodo de validación) con $n_0 < n_{\text{est}} < n$. La muestra de estimación se utilizaría para determinar tanto \mathbf{P} como los pesos $\{w_c; c \in \mathbf{P}\}$ aplicando el algoritmo de la sección 6.3. La muestra de validación, por su parte, se utilizará para evaluar el comportamiento predictivo del modelo tanto desde un punto de vista gráfico como numérico.

La elaboración de predicciones estará basada en la distribución predictiva dada por la mixtura $\sum_{c \in \mathbf{P}} w_c [y | \mathbf{z}, \mathbf{x}, \mathbf{D}, \mathbf{M}_c]$ donde $\mathbf{z} = (z_1, \dots, z_q)'$, $\mathbf{x} = (x_1, \dots, x_k)'$ son las características hedónicas de la vivienda cuyo precio $y = \log(p)$ se quiere predecir, $\mathbf{D} = \{\mathbf{Y}, \mathbf{Z}, \mathbf{X}\}$ son los datos disponibles para realizar la predicción e $[y | \mathbf{z}, \mathbf{x}, \mathbf{D}, \mathbf{M}_c]$ denota la distribución predictiva del modelo \mathbf{M}_c que, en nuestro caso, vendrá dada por:

$$N\left(E\left[y | \mathbf{z}, \mathbf{x}, \mathbf{D}, \hat{\boldsymbol{\theta}}_{\mathbf{M}_c}, \mathbf{M}_c\right], \text{Var}\left(y | \mathbf{z}, \mathbf{x}, \mathbf{D}, \hat{\boldsymbol{\theta}}_{\mathbf{M}_c}, \mathbf{M}_c\right)\right)$$

donde $E[y | \mathbf{z}, \mathbf{x}, \mathbf{D}, \hat{\boldsymbol{\theta}}_{M_c}, \mathbf{M}_c]$, $\text{Var}(y | \mathbf{z}, \mathbf{x}, \mathbf{D}, \hat{\boldsymbol{\theta}}_{M_c}, \mathbf{M}_c)$ indican las correspondientes adaptaciones de las expresiones (6.3) y (6.4). En particular una predicción puntual de y vendría dada por la expresión:

$$E[y | \mathbf{z}, \mathbf{x}, \mathbf{D}] = \sum_{c \in \mathbf{P}} w_c E[y | \mathbf{z}, \mathbf{x}, \mathbf{D}, \hat{\boldsymbol{\theta}}_{M_c}, \mathbf{M}_c] \quad (6.11)$$

Así mismo, la varianza de la distribución predictiva vendría dada por:

$$\begin{aligned} \text{Var}(y | \mathbf{z}, \mathbf{x}, \mathbf{D}) = & \sum_{c \in \mathbf{P}} w_c \text{Var}(y | \mathbf{z}, \mathbf{x}, \mathbf{D}, \hat{\boldsymbol{\theta}}_{M_c}, \mathbf{M}_c) + \\ & + \sum_{c \in \mathbf{P}} w_c \left(E[y | \mathbf{z}, \mathbf{x}, \mathbf{D}, \hat{\boldsymbol{\theta}}_{M_c}, \mathbf{M}_c] - E[y | \mathbf{z}, \mathbf{x}, \mathbf{D}] \right)^2 \end{aligned} \quad (6.12)$$

Finalmente, la elaboración de intervalos predictivos se realizaría a partir de los cuantiles correspondientes de la mixtura $\sum_{c \in \mathbf{P}} w_c [y | \mathbf{z}, \mathbf{x}, \mathbf{D}, \mathbf{M}_c]$ obtenidos mediante simulación aplicando el siguiente muestreo de composición.

Repetir $s = 1, \dots, n_{\text{sim}}$

Paso 1. Extraer $\mathbf{c}^{(s)} \in \mathbf{P}$ de la distribución discreta $\{(\mathbf{c}, w_c); \mathbf{c} \in \mathbf{P}\}$

Paso 2. Extraer $y^{(s)}$ de la siguiente distribución:

$$N\left(E[y | \mathbf{z}, \mathbf{x}, \mathbf{D}, \hat{\boldsymbol{\theta}}_{M_{c^{(s)}}}, \mathbf{M}_{c^{(s)}}], \text{Var}(y | \mathbf{z}, \mathbf{x}, \mathbf{D}, \hat{\boldsymbol{\theta}}_{M_{c^{(s)}}}, \mathbf{M}_{c^{(s)}})\right)$$

donde $E[y | \mathbf{z}, \mathbf{x}, \mathbf{D}, \hat{\boldsymbol{\theta}}_{M_c}, \mathbf{M}_c]$, $\text{Var}(y | \mathbf{z}, \mathbf{x}, \mathbf{D}, \hat{\boldsymbol{\theta}}_{M_c}, \mathbf{M}_c)$ vienen dadas por las correspondientes adaptaciones de las expresiones (6.3) y (6.4).

6.5.2.1 Validación *rolling*

Para llevar a cabo la validación *rolling*, es necesario actualizar los pesos tras la incorporación de cada nueva observación al proceso de predicción.

Con la entrada de la información que aporta una nueva transacción, se reestiman todos los modelos que han quedado finalmente como los eficientes de la población tras la ejecución del algoritmo. A cada uno de ellos se le calcula su probabilidad por medio de la expresión (6.11) donde el criterio BIC se computa utilizando el nuevo periodo de estimación y esas probabilidades se utilizan como los pesos para construir la mixtura de modelos que deberá hacer la predicción en ese paso. De esta forma, con la actualización de los pesos asignados a cada modelo de la mixtura, que se va adaptando a la nueva información recibida, se consigue, en términos bayesianos, una mejor cuantificación de la incertidumbre asociada al proceso de selección de modelos.

6.6 Resultados

La metodología que se ha expuesto en las secciones anteriores de este capítulo se ilustra con la aplicación a la base de datos presentada en 2.6.1.

En nuestro caso, tras analizar las autocorrelaciones espaciales de la variable dependiente mostradas en la Figura 3 y no apreciar dependencias significativas más allá del retardo 35, se decide tomar $m_{Tred} = m_{Sred} = \{0, 5, \dots, 30, 35\}$ ($m_T = 35$ corresponde a un mes de transacciones aproximadamente). Para γ y λ , se proponen las siguientes redes $\gamma_{red} = \lambda_{red} = \{0.25, 0.50, 0.75, 1\}$.

Tomamos como probabilidades de mutación y recombinación:

$$p_{\text{mut}} = \frac{1}{q + k + 4 + q_T + q_\gamma + q_S + q_\lambda}, p_{\text{comb}} = 0.2, N = 100$$

que son valores habituales en los algoritmos genéticos. El número de iteraciones máximo $\text{iter}_{\text{max}} = 1000$. Como periodo de validación se toma el mismo que en los capítulos anteriores y, por lo tanto, $n_{\text{est}} = 457$ para el cálculo del criterio LPRED. Sin embargo, y de cara a llevar a cabo el proceso de validación extramuestral (ordinaria y *rolling*) de los modelos estimados, fue necesario determinar un nuevo periodo de validación. Con el fin de tener observaciones suficientes para calcular el valor del criterio LPRED por un lado, y por otro, poder llevar a cabo los procesos de validación de los resultados obtenidos, se toma la decisión salomónica de dividir el año 2004 en dos semestres, el primero para calcular el valor del criterio LPRED y el segundo para validar el modelo. De esta forma el periodo de validación extramuestral abarca las observaciones 657 a 788, conteniendo un total de 132 observaciones.

Un estudio de sensibilidad llevado a cabo no cambió sustancialmente los resultados obtenidos.

6.6.1 Estimación

La aplicación del algoritmo genético ha dado lugar a la selección de 91 modelos. En la Tabla 6.1 se presentan los valores estimados de los coeficientes de los modelos $M_{\text{ind}_z, \text{ind}_x, \text{ind}_\theta, \theta}$, con probabilidades w_i superiores a un 1%, con sus respectivas probabilidades.

Con fines comparativos, se aplicó, la metodología descrita en el capítulo 2 para estimar un modelo STAR utilizando las redes discretas anteriores para los parámetros de vecindad. Los valores óptimos encontrados

por el criterio de la máxima-verosimilitud para dichos parámetros corresponden respectivamente a $m_T = 10$, $m_S = 20$, $\gamma = \lambda = 1$. En la Tabla 6.2 se muestran también los coeficientes estimados para las variables seleccionadas mediante el proceso de simplificación del modelo descrito en la sección 2.5.

Se puede observar que las variables hedónicas significativas al 95% coinciden, así como su signo, en todos los modelos. La variable PC2, ha sido seleccionada solo por los modelos M4 a M7, sin ser significativa en uno de ellos, y el garaje únicamente se encuentra en el modelo menos probable M7. Con respecto a la cuantificación, es notable el grado de concordancia entre todos ellos, excepto en la constante, donde existen diferencias muy marcadas.

Los coeficientes autorregresivos que resultan significativos son los que recogen influencia espacial, bien sola ϕ_S , bien en interacción con el tiempo ϕ_{TS} , de forma positiva el primero y negativa el segundo, pero en ninguno de los siete modelos aparecen como significativos los coeficientes ϕ_T ni ϕ_{ST} . Uno de los modelos no recoge ninguno de estos cuatro coeficientes, quedando por tanto como un modelo hedónico.

Con respecto a los vecinos temporales, los modelos más probables rechazan su influencia, y el resto seleccionan 10 vecinos con una ponderación igual a uno, dando de este modo la misma importancia a todos ellos. Sin embargo, la información que proporcionan los vecinos espaciales es valorada por todos, situando la elección de su número en 20 o 25 en todos los casos y ponderando a todos por igual. De forma global, se puede apuntar que los dos modelos con mayor probabilidad coinciden, en cuanto a las variables que incorporan, con el modelo STARs del capítulo 5.

Como se ha descrito en el apartado 6.4, una vez obtenidos los modelos eficientes de la población, seleccionados por el algoritmo genético de acuerdo con el arbitraje de los criterios escogidos, se construye una mixtura con el fin de realizar inferencias. La Tabla 6.2 muestra las probabilidades a posteriori de que cada una de las variables y efectos del modelo STAR formen parte del modelo evaluadas utilizando la expresión (6.10), así como los coeficientes estimados y sus desviaciones típicas a posteriori estimadas mediante las expresiones (6.8) y (6.9), respectivamente.

Los resultados indican que las variables y efectos seleccionados (aquéllos con probabilidades a posteriori superiores a un 50%) coinciden esencialmente con los seleccionados por los dos modelos más probables de la Tabla 6.1, así como por el modelo STARs con errores t de Student del capítulo anterior y por el seleccionado por la metodología utilizada en el capítulo 4 (ver Tabla 6.2). Todos ellos seleccionan, además de la constante, la tendencia temporal, la componente PC1, la superficie, antigüedad, tipo de adquisición, tener portero, garaje y el efecto ϕ_S como las variables y efectos que deben ser incorporados al modelo, existiendo también unanimidad en sus signos y una gran coincidencia en la mayor parte de los valores estimados de los mismos. Todos los modelos coinciden esencialmente en la estimación del número de vecinos m_S que toma valores en torno a 10 así como en el valor estimado del coeficiente λ , esencialmente igual a 1. Estos resultados dotan de un mayor grado de robustez al análisis realizado a lo largo de la Memoria.

Tabla 6.1: Estimaciones de los parámetros (en negrita los significativos para un nivel de credibilidad del 95%) de los modelos más probables seleccionados por el algoritmo genético

VARIABLES	MODELOS MÁS PROBABLES						
	M1	M2	M3	M4	M5	M6	M7
Constante	5.2286	4.9165	11.0107	8.6959	14.2522	11.8278	14.2312
Tiempo	0.0259	0.0257	0.0411	0.0365	0.0515	0.0438	0.0517
PC1	0.0002	0.0002	0.0002	0.0002	0.0003	0.0002	0.0003
PC2				-0.0003	-0.0003	-0.0002	-0.0003
Log (superficie)	0.7335	0.7297	0.7281	0.7365	0.7357	0.7239	0.7433
Log (antigüedad)	-0.2197	-0.2189	-0.2211	-0.2160	-0.2177	-0.2200	-0.2301
VPO							
Tipo de adquisición	0.4508	0.4486	0.4530	0.4439	0.4463	0.4526	0.4486
Refrigeración							
Calefacción colectiva							
Sin calefacción							
Nº habitación es <4							
Nº habitaciones =4							
Nº habitaciones >5							
Portero	0.1341	0.1318	0.1454	0.1242	0.1321	0.1378	0.1463
Garaje							-0.1019
Ascensor	0.1097	0.1124	0.0991	0.1198	0.1111	0.1005	0.1309
Indicador missing							
ϕ_T							
ϕ_S	0.4045	0.4415	0.4270			0.3309	
ϕ_{ST}							
ϕ_{TS}			-0.6884		-0.6438	-0.6842	-0.6392
τ	0.1925	0.1927	0.1910	0.1930	0.1916	0.1902	0.1904
m_T			10		10	10	10
γ			1.0000		1.0000	1.0000	1.0000
m_S	20	25	20		20	20	20
λ	1.0000	1.0000	1.0000		1.0000	1.0000	1.0000
Probabilidad del Modelo	0.3026	0.2033	0.1983	0.1360	0.0620	0.0340	0.0221

Tabla 6.2: Probabilidades a posteriori de que las variables y efectos espacio-temporales del modelo STAR formen parte del modelo, y media y desviación típica (DE) a posteriori de los coeficientes hedónicos si fueran significativos(en negrita las estimaciones significativas)

VARIABLES	P (%)	Mixtura			STARs		
		Coef	Desv	t-stadist	Coef	Desv	t-stadist
Constante	100.00	8.0038	3.6798	2.1751	5.2286	1.2245	4.2699
Tiempo	100.00	0.0337	0.0131	2.5714	0.0259	0.0093	2.7805
PC1	100.00	0.0002	0.0001	3.6726	0.0002	0.0000	4.3337
PC2	29.58	-0.0003	0.0001	-2.5611			
Log (superficie)	100.00	0.7317	0.0485	15.0975	0.7335	0.0506	14.4952
Log (antigüedad)	100.00	-0.2196	0.0184	-11.9226	-0.2197	0.0186	-11.7930
VPO	0.00	0.0000	0.0000				
Tipo de adquisición	100.00	0.4494	0.0359	12.5106	0.4508	0.0363	12.4112
Refrigeración	0.00	0.0007	0.0470	0.0144			
Calefacción colectiva	0.79	0.0867	0.0472	1.8383			
Sin calefacción	0.00	0.0313	0.0518	0.6041			
Nº habitación es <4	0.00	0.0702	0.0708	0.9917			
Nº habitaciones =4	0.02	-0.0441	0.0432	-1.0215			
Nº habitaciones >5	0.00	0.0683	0.0626	1.0923			
Portero	100.00	0.1350	0.0534	2.5298	0.1341	0.0549	2.4404
Garaje	4.60	-0.1041	0.0500	-2.0819			
Ascensor	100.00	0.1100	0.0379	2.9026	0.1097	0.0389	2.8169
Indicador missing	0.00	0.0397	0.0428	0.9276			
ϕ_T	3.26	0.1783	0.0853	2.0904			
ϕ_S	76.13	0.4141	0.1368	3.0272	0.4044	0.1431	2.8267
ϕ_{ST}	35.50	-0.6679	0.2903	-2.3005			
ϕ_{TS}	0.02	-0.4306	0.4981	-0.8644			
τ		0.1920	0.0010	198.0182			
m_T		10.5895	3.0565	3.4646			
γ		0.9816	0.0653	15.0260			
m_S		21.1809	2.1237	9.9737			
λ		1.0000	0.0000				
σ					0.1290		

6.6.2 Validación

A continuación se lleva a cabo la validación de la mixtura tanto intramuestral como extramuestralmente. Su comportamiento intramuestral se analizará a través de sus residuos de forma gráfica y numérica y su proceder predictivo se estudiará tras la ejecución de dos procedimientos de validación, uno ordinario y otro de tipo *rolling*, como en los anteriores capítulos.

6.6.2.1 Validación intramuestral

La distribución de los errores intramuestrales de la mixtura se puede examinar en la Tabla 6.3 y en las Figuras 6.1 a 6.3. Una vez más se pone de manifiesto la asimetría negativa y la enorme leptocurtosis, debida a atípicos (ver Tabla 6.3 y Figura 6.2). En cuanto a la evolución dinámica de los errores no se aprecian autocorrelaciones significativas.

De forma numérica, los cubrimientos al 95% muestran diferencias con el resultado teórico (pvalor 0.034), pero son correctos al 99%.

Tabla 6.3: Distribución de los errores tipificados intramuestrales y extramuestrales de la mixtura

ERRORES TIPIFICADOS DE LA MIXTURA			
	Intramuestrales	Extramuestrales (ord)	Extramuestrales (<i>rolling</i>)
Mínimo	-7.6370	-3.1734	-3.9759
1%	-3.2399	-1.8926	-2.5108
5%	-1.6362	-0.8434	-1.4259
10%	-1.1126	-0.4385	-1.0184
25%	-0.4120	0.2851	-0.2396
50%	0.1200	0.6482	0.1751
75%	0.5376	0.9678	0.5463
90%	0.9002	1.2306	0.7963
95%	1.3809	1.3510	1.0006
99%	2.1019	1.8163	1.4431
Máximo	2.6910	1.9383	1.5713
Media	-0.0011	0.5364	0.0521
Mediana e 	0.4840	0.7545	0.4416
Asimetría	-1.7579	-1.6983	-1.6326
Curtosis	12.4429	8.0216	7.7986
R²	0.4371	0.2996	0.5304
CUB95	94.79	98.48 ⁺	96.97 ⁺
CUB99	98.22 ⁺	99.24	99.24

⁺ Existen diferencias significativas con el resultado teórico al 5%

Resultados

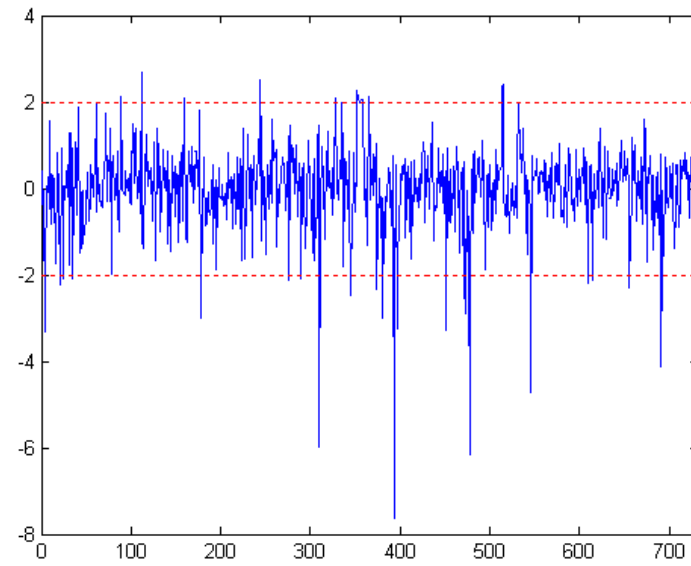


Figura 6.1: Errores tipificados intramuestrales de la mezcla

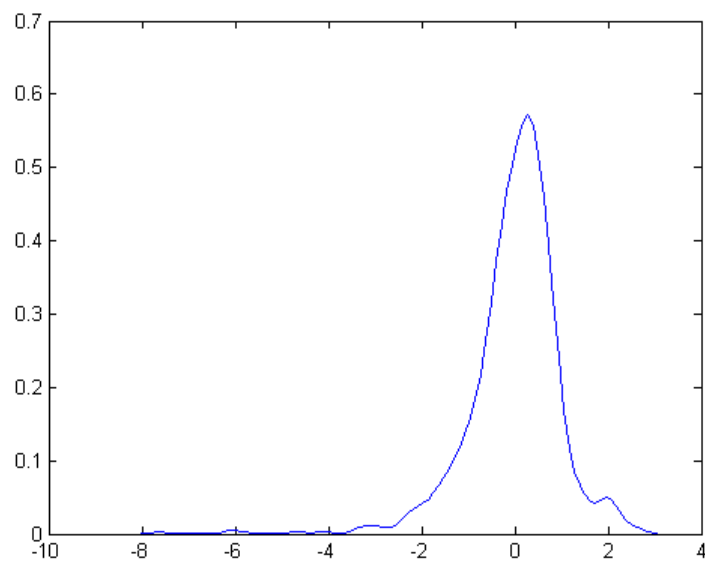


Figura 6.2: Densidad de los errores intramuestrales de la mezcla

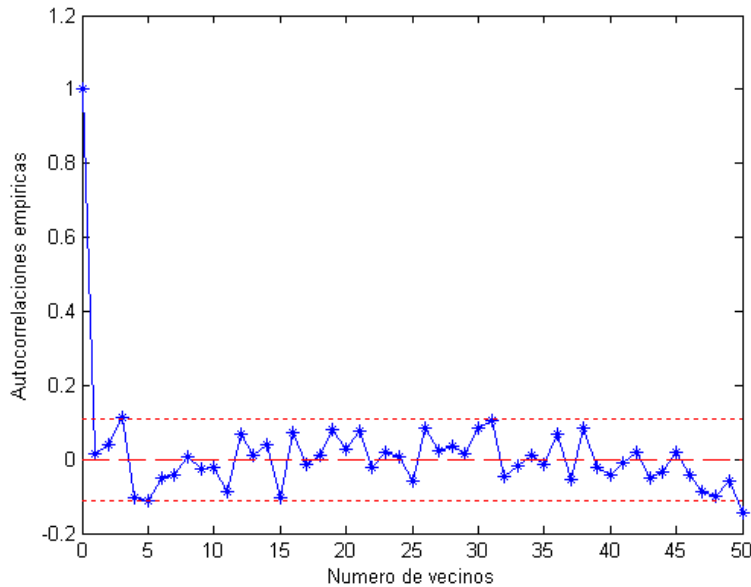


Figura 6.3: Autocorrelaciones espaciales residuales de la mixtura (en líneas rayadas en rojo las bandas de confianza al 95%)

6.6.2.2 Validación extramuestral

La distribución de los errores extramuestrales (ver Tabla 6.3) indica un buen comportamiento tanto en el procedimiento ordinario como en el de tipo *rolling*, con una importante reducción de la curtosis respecto a lo observado en las validaciones de los capítulos anteriores. Es imprescindible mencionar a este respecto, que el periodo de validación en este caso ha sido mucho más pequeño que en anteriores análisis (132 observaciones), pudiendo encontrarse aquí una explicación a este hecho. De forma gráfica los errores (ver Figuras 6.6 a 6.8 y 6.11 a 6.13) inciden en lo mencionado mostrando, en general, un correcto proceder y manifestando la ausencia de autocorrelaciones significativas. Tan sólo se observan errores significativamente negativos en unas pocas transacciones, que coinciden con algunas de las identificadas como

Resultados

fuertes atípicos en el capítulo anterior y que provocan, de nuevo, la existencia de asimetría significativamente negativa en la distribución de los errores. En cuanto a las predicciones, tanto del precio como del logaritmo del precio (ver Figuras 6.4, 6.5, 6.9 y 6.10) son adecuadas y podemos hablar de un apropiado ajuste tanto intra como extramuestral de la mixtura, con la salvedad hecha sobre el tamaño del periodo de validación.

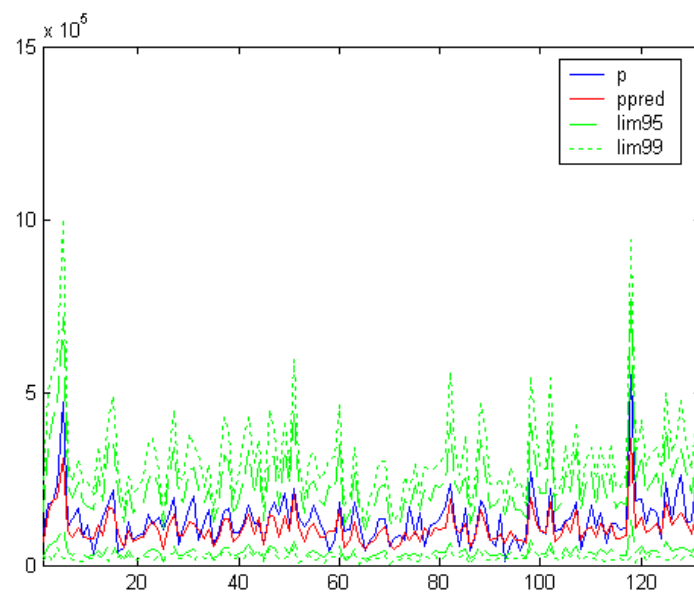


Figura 6.4: Predicciones extramuestrales del precio para la mixtura

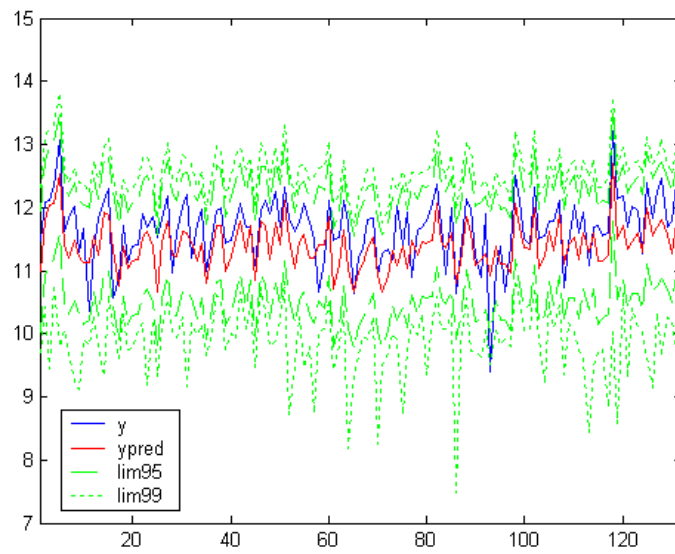


Figura 6.5: Predicciones extramuestrales del logaritmo del precio para la mixtura

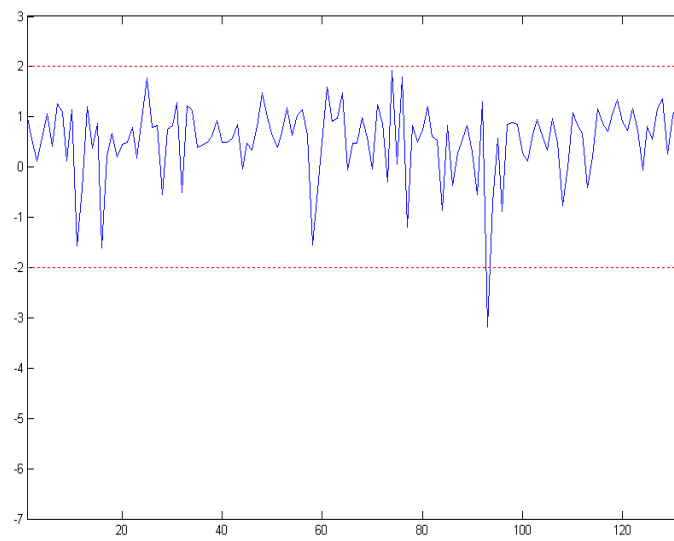


Figura 6.6: Errores tipificados extramuestrales de la mixtura

Resultados

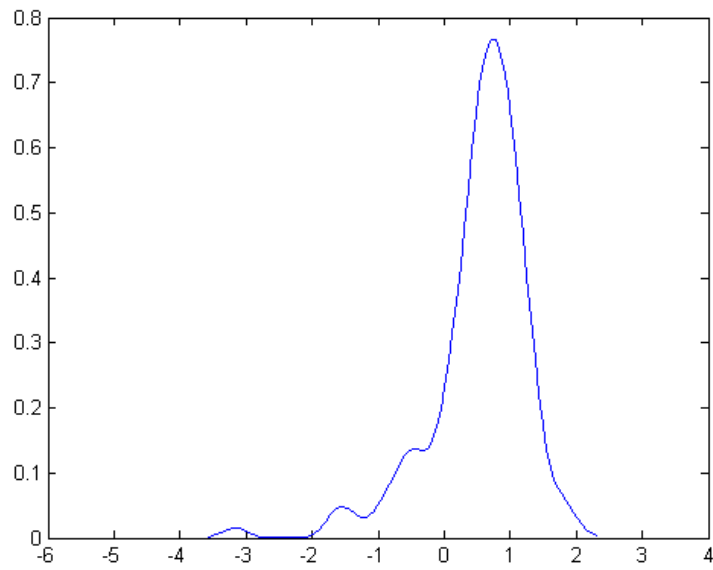


Figura 6.7: Densidad de los errores extramuestrales de la mezcla

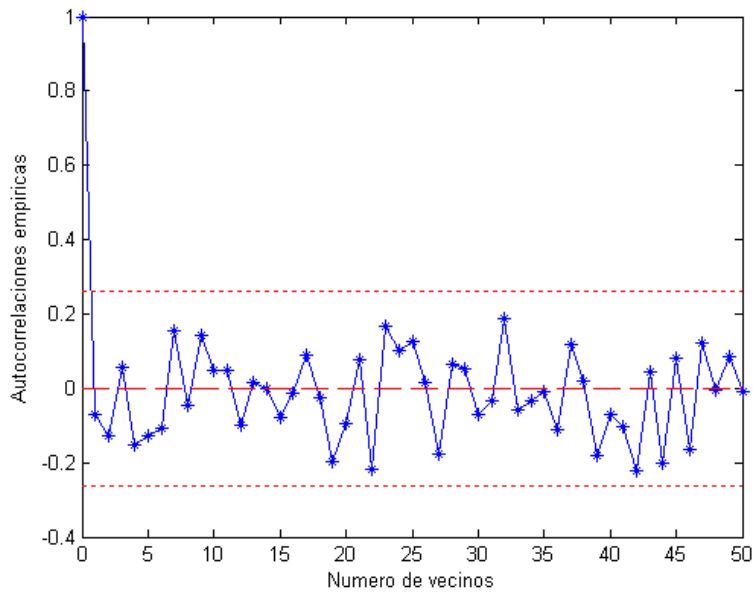


Figura 6.8: Autocorrelación de los errores extramuestrales de la mezcla

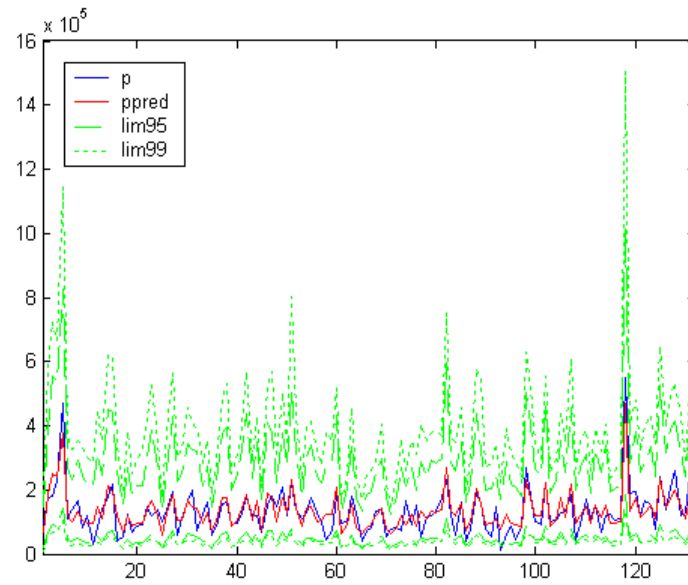


Figura 6.9: Predicciones extramuestrales (*rolling*) del precio para la **mixtura**

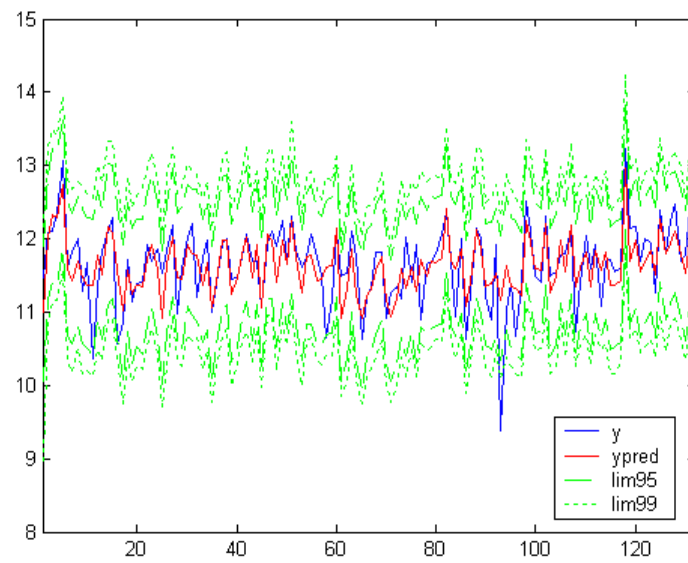


Figura 6.10: Predicciones extramuestrales (*rolling*) del logaritmo del **precio para la mixtura**

Resultados

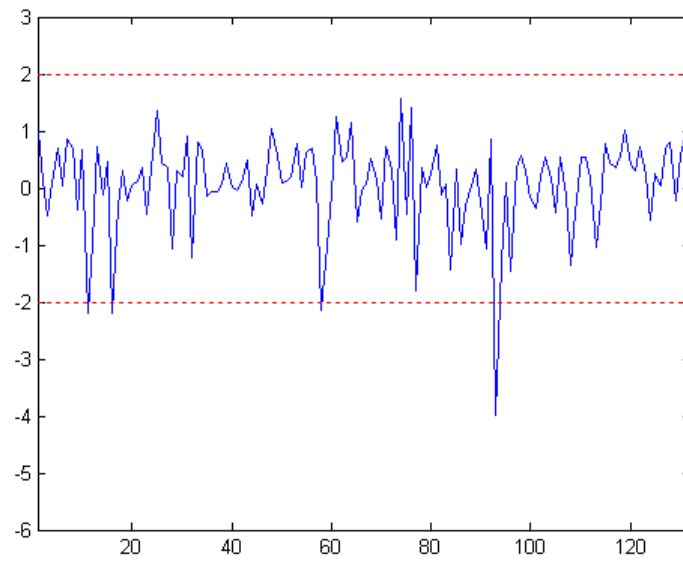


Figura 6.11: Errores tipificados extramuestrales (*rolling*) de la mezcla

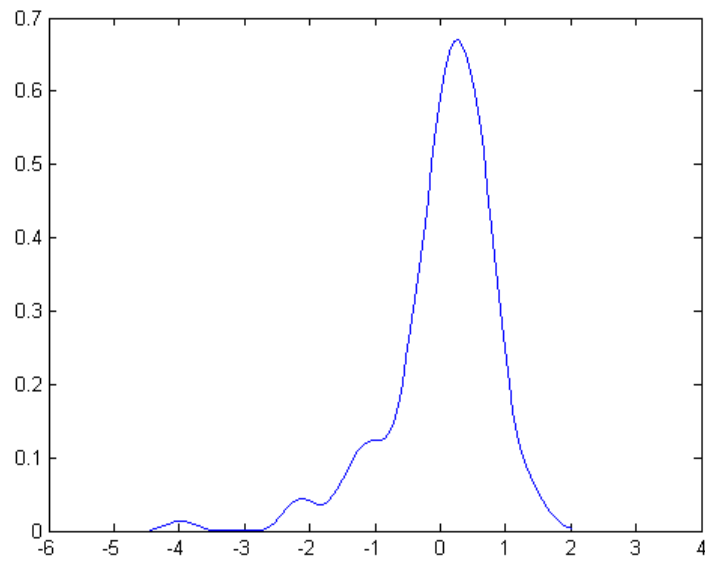


Figura 6.12: Densidad de los errores extramuestrales (*rolling*) de la mezcla

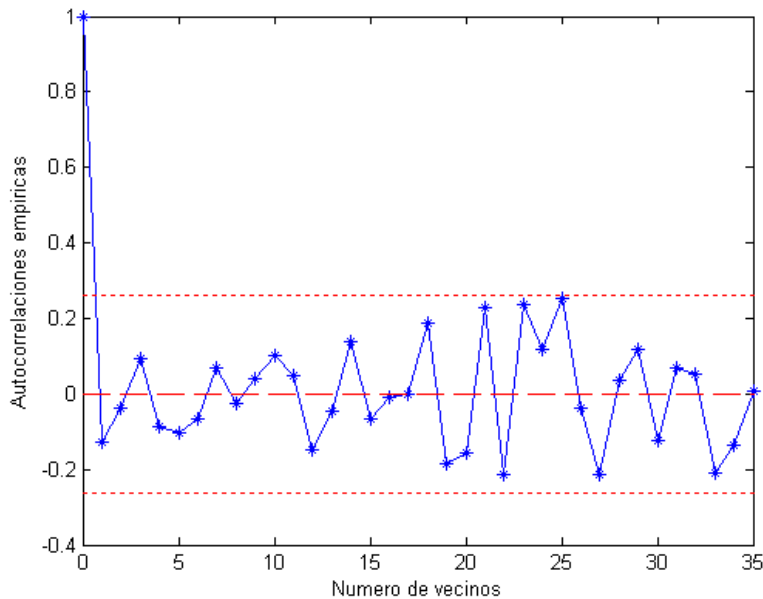


Figura 6.13: Autocorrelaciones de los errores extramuestrales (*rolling*) de la mixtura

6.6.3 Comparación

De forma análoga al proceder en todos los capítulos de esta Memoria, se va a llevar a cabo un procedimiento de comparación. En este caso se realizará entre la mixtura y el modelo STAR restringido seleccionado en el capítulo 2. De forma numérica se procede a la comparación de sus comportamientos. Los resultados se presentan en las Tablas 6.4 a 6.6.

Se observa que los valores de los criterios no difieren, en general, significativamente entre ambos modelos mostrando un mejor comportamiento el modelo STARs en la mayor parte de ellos. El resultado obtenido es lógico si tenemos en cuenta que los dos seleccionan las mismas variables y efectos espacio temporales. La validación *rolling* únicamente añade que este

comportamiento es bastante estable a lo largo del tiempo de validación utilizado sin que ocurran diferencias dignas de mención.

Tabla 6.4: Criterios de comparación intramuestrales entre el modelo, STARS y la mixtura (en negrita el modelo con mejor comportamiento)

CRITERIOS	MODELO	
	Mixtura	STARS
RMSE	0.4385	0.4388
MAD	0.3011	0.3012
-2LLF	870.24	870.17
SPL	144.02	141.07

Tabla 6.5: Criterios de comparación extramuestrales entre el modelo, STARS y la mixtura (en negrita el modelo con mejor comportamiento)

CRITERIOS	MODELO	
	Mixtura	STARS
RMSE	0.4391	0.3597
MAD	0.3788	0.2646
LPRED	-77.67	-58.56

Tabla 6.6: Criterios de comparación extramuestrales (*rolling*) entre el modelo, STARS y la mixtura (en negrita el modelo con mejor comportamiento)

CRITERIOS	MODELO	
	Mixtura	STARS
RMSE	0.3595	0.3594
MAD	0.2596	0.2621
LPRED	-58.79	-57.82

6.7 Conclusiones

En este capítulo se ha propuesto una metodología multicriterio de selección de variables y efectos espacio-temporales en el modelo STAR con efectos de vecindad analizado a lo largo de la Memoria. Los criterios utilizados en el proceso de selección (BIC, pérdida predictiva cuadrática y logaritmo de la densidad predictiva) están basados en la literatura bayesiana de selección y comparación de modelos y valoran la bondad de ajuste a los datos de los modelos comparados, así como su comportamiento predictivo extramuestral.

Así mismo, se ha propuesto un algoritmo genético para resolver el problema multiobjetivo planteado que permite, en particular, identificar las soluciones eficientes del mismo. Utilizando dichas soluciones se construye una mixtura de modelos basada en ideas bayesianas, que permite, por un lado, identificar cuáles de las variables y efectos de vecindad ejercen una influencia significativa sobre la evolución de los precios de vivienda y, por otro, realizar estimaciones de los parámetros del modelo y predicciones extramuestrales que reflejen más adecuadamente, la incertidumbre asociada al proceso de selección y estimación de modelos llevado a cabo.

Los resultados obtenidos al aplicar la metodología desarrollada a la base de datos analizada en la Memoria confirman los resultados expuestos en los capítulos anteriores dando mayor robustez y generalidad a los mismos.

Conclusiones generales y líneas de futura investigación

En esta Memoria se ha presentado una reparametrización del modelo espacio temporal autorregresivo STAR con efectos de vecindad introducido por Pace y otros (1998b, 2000) al que se le han aplicado diferentes aproximaciones, tratamientos de estimación y procedimientos de selección de variables.

Desde una óptica frecuentista, tras un proceso de estimación de máxima verosimilitud, se detectan falta de normalidad en los errores.

A continuación, se ha planteado un análisis bayesiano del modelo con el fin de evitar la utilización de resultados asintóticos de dudosa aplicabilidad en el caso de la estimación de algunos de los parámetros del modelo y para incorporar, además, la incertidumbre al proceso de estimación. Se han implementado mejoras paso a paso, para resolver las disfunciones que presentan los errores aumentando en primer lugar la flexibilidad de los parámetros de vecindad a través de redes más amplias para algunos y distribuciones continuas para otros, en una segunda etapa se ha planteado el modelo con errores heterocedásticos con el fin de minimizar el impacto de observaciones atípicas llevando a cabo un proceso de estimación robusto, que logra mejorar el ajuste de forma notable.

La Memoria ha presentado, así mismo, un método de predicciones retrospectivas y una construcción de números índice, útiles para la aplicación considerada en la memoria

Por último, se ha propuesto una metodología multicriterio resuelta por medio de un algoritmo genético, que permite incorporar la incertidumbre al

proceso de estimación y selección de variables, no siendo a la vez computacionalmente costoso.

Todas las metodologías y procedimientos han sido aplicadas a una base de datos de precios de transacciones de viviendas de un área concreta de la ciudad de Zaragoza durante los años 2003 y 2004. Los resultados obtenidos en las estimaciones de todos los Capítulos han mostrado gran concordancia en todos los modelos analizados, indicando como relevantes sobre el precio de una vivienda las variables hedónicas indicador de tiempo, proximidad al centro de la ciudad siguiendo la dirección Noroeste-Sureste, antigüedad, superficie, tipo de adquisición, ascensor y portero. Otras han aparecido solamente en algunas de las estimaciones, a saber: la dirección Nordeste-Suroeste hacia el centro y el garaje. Los coeficientes autorregresivos del modelo no muestran la misma estabilidad.

Finalmente, en cuanto a las líneas de investigación abiertas y en las cuales nos encontramos trabajando:

- 1) Conseguir una base de datos más amplia que permita a las técnicas desarrollar todo su potencial para obtener resultados de la aplicación útiles en ámbitos económicos.
- 2) Búsqueda de distribuciones más generales del error que permitan modelar de forma más correcta la asimetría y curtosis presente en los datos de precios de vivienda.
- 3) Modelización con coeficientes no constantes por medio de métodos de regresión geográficamente ponderados.

Apéndice A

Computación bayesiana

A.1 Introducción

La Inferencia Bayesiana combina las creencias a priori sobre los parámetros del modelo con la información que proporcionan los datos, utilizando el teorema de Bayes. La mayor diferencia con el enfoque frecuentista viene dada por la interpretación subjetiva de las probabilidades que la teoría bayesiana ofrece y su tratamiento de la incertidumbre.

De forma general, el objetivo de la Inferencia Bayesiana es tomar decisiones o alcanzar conclusiones acerca de un parámetro θ o futuras observaciones y_{pred} , las cuales considera desconocidas o variables aleatorias, utilizando información probabilística condicionada a los datos observados.

Un modelo bayesiano, considera, por tanto, que los datos \mathbf{Y} , así como los parámetros del modelo θ , son variables aleatorias cuya distribución conjunta viene dada por una expresión de la forma

$$p(\mathbf{Y}|\theta) = f(\mathbf{Y}|\theta)\pi(\theta)$$

donde $f(\mathbf{Y}|\boldsymbol{\theta})$ especifica el modelo muestral de los datos observados y recibe el nombre de *verosimilitud* y $\pi(\boldsymbol{\theta})$ cuantifica la incertidumbre asociada al valor de los parámetros del modelo previa a la observación de los datos y recibe el nombre de *distribución a priori*.

El proceso de aprendizaje acerca del valor de los parámetros se lleva a cabo mediante su *distribución a posteriori*, que se calcula aplicando el teorema de Bayes y viene dada por la expresión:

$$p(\boldsymbol{\theta} | \mathbf{Y}) = \frac{f(\mathbf{Y} | \boldsymbol{\theta}) \pi(\boldsymbol{\theta})}{\int_{\Theta} f(\mathbf{Y} | \boldsymbol{\theta}) \pi(\boldsymbol{\theta}) d\boldsymbol{\theta}}$$

A partir de esta distribución se pueden realizar inferencias acerca de los parámetros $\boldsymbol{\theta}$, bien sea en términos de momentos o de cuantiles de la misma que proporcionan estimaciones puntuales y cuantifican la incertidumbre asociada a su proceso de estimación. Es importante observar que, a diferencia de los procedimientos de estimación de la Inferencia Clásica que utilizan procesos de estimación y medidas de incertidumbre pre-muestrales, los procesos de estimación bayesianos se realizan siempre condicionalmente a los datos observados y tienen, por tanto, un carácter post-muestral.

Esta forma de tratar la incertidumbre ha sido conocida desde hace tiempo (De Finetti, 1937; Savage, 1954; Jeffreys, 1961; Berger, 1985; Bernardo y Smith, 1994). Pero este reconocimiento ha chocado por una parte con las críticas sobre su falta de objetividad, y por otra, con las dificultades de índole práctica a la hora de llevar a cabo los cálculos matemáticos necesarios en su tratamiento.

De aquí que, para conseguir un mayor grado de aplicabilidad a la resolución de casos prácticos, la Inferencia Bayesiana haya tenido ante si dos

importantes retos: en primer lugar, la elección de la distribución a priori y en segundo lugar, y desde una óptica más pragmática, el cálculo de la distribución a posteriori y de momentos a partir de ella.

A.2 La distribución a priori

La elección de la distribución a priori, es una de las mayores críticas que la Estadística Bayesiana ha recibido desde sus inicios debido a su carácter eminentemente subjetivo.

En la literatura bayesiana se han propuesto diversas formas de especificarla, entre las que cabe destacar las distribuciones a priori conjugadas y las llamadas distribuciones a priori no informativas.

Las distribuciones a priori y a posteriori son conjugadas si pertenecen a la misma familia. Si esto ocurre el cálculo de la distribución a posteriori resulta sencillo, ya que se conoce su forma. Sin embargo, existen relativamente pocas situaciones prácticas en las que se dé esta situación y, como por ejemplo Berger (1985) argumenta, suelen imponer restricciones importantes sobre los parámetros, principalmente en lo que hace referencia a su comportamiento en las colas, todo lo cual limita su grado de aplicabilidad.

Cuando las distribuciones a priori y a posteriori no son conjugadas, se presentan en formas paramétricas diferentes. En la práctica esto lleva a la resolución de integrales difíciles, incluso inabordables, lo cual conduce a la utilización de otros métodos de cálculo aproximado, como se verá más adelante.

Las distribuciones a priori no informativas, por su parte, reflejan una situación de falta de conocimiento previo sobre los parámetros. Si el

parámetro es unidimensional el método más utilizado es la llamada distribución a priori de Jeffreys (Jeffreys, 1946, 1961) basado en la matriz de información de Fisher. En el caso multivariante el problema es mucho más complicado siendo uno de los métodos más utilizados las llamadas distribuciones de referencia propuesto por Bernardo (1979) y analizado más recientemente por Berger y Bernardo (1989) y Berger y otros (2007). Dicho método tiene como inconveniente principal que no es invariante al orden de las componentes de la parametrización elegida. Otro inconveniente es que a menudo estas distribuciones a priori son impropias lo cual no impide que la distribución a posteriori lo sea, aunque en cada caso particular este aspecto deberá de ser comprobado (Sun y Berger, 1998; Berger y otros, 2001).

A.3 La distribución a posteriori

La escasa aplicación de los métodos bayesianos a la resolución de problemas reales hasta los años 90 vino dada, sin embargo, por cuestiones prácticas derivadas, por una parte, de la poca capacidad de los medios informáticos y por otra, de la complejidad de los problemas reales que había que modelizar, cuyo tratamiento requería cálculos inabordables desde el punto de vista analítico. Pero la revolución de los últimos años con la mejora de métodos ya existentes, los nuevos métodos MCMC y la potencia computacional, impensable hace solo algunos años, han hecho que en la actualidad los métodos bayesianos sean métodos de elección en multitud de situaciones.

La mencionada necesidad de métodos de cálculo complejo viene dada por las herramientas necesarias en las dos fases que componen la aproximación bayesiana. La primera es aquella en la que se realizan

La distribución a posteriori

inferencias basadas en las distribuciones a posteriori de los parámetros θ de un modelo, que contiene toda la información sobre la que se elaboran los juicios inductivos bayesianos a través del cálculo de momentos, cuantiles o intervalos de credibilidad. Y la segunda fase es de tipo decisional y en ella se evalúan las consecuencias de cada elección que, en ocasiones, requiere el cálculo de la distribución predictiva a posteriori dada por

$$f(\mathbf{y}_{\text{pred}} | \mathbf{Y}) = \int_{\Theta} f(\mathbf{y}_{\text{pred}} | \theta) p(\theta | \mathbf{Y}) d\theta$$

En ambos casos es necesario el cálculo de cocientes de integrales de la forma

$$\int_{\Theta} g(\theta) p(\theta | \mathbf{Y}) d\theta = \frac{\int_{\Theta} g(\theta) f(\mathbf{Y} | \theta) \pi(\theta) d\theta}{\int_{\Theta} f(\mathbf{Y} | \theta) \pi(\theta) d\theta}$$

para funciones $g: \Theta \rightarrow \mathbf{R}$ que no suelen ser tratables analíticamente (salvo para el caso de distribuciones conjugadas) y muy difíciles de calcular si θ es un vector de parámetros con muchas componentes, situación habitual en la resolución de problemas reales.

Para poder realizar este tipo de cálculos, podemos distinguir tres tipos de métodos: analítico/asintóticos, integración numérica y simulación.

Entre los analíticos se encuentra el método basado en las distribuciones a priori conjugadas adaptadas a los modelos exponenciales, ya mencionados. Los métodos asintóticos por su parte, se pueden aplicar cuando el tamaño de la muestra es suficientemente grande para que en la fórmula dada del teorema de Bayes, la contribución de la verosimilitud predomine sobre la distribución a priori, y en este caso, la distribución a posteriori será aproximadamente normal con media la moda a posteriori y como matriz de covarianzas el

inverso del hessiano del logaritmo de la posteriori, evaluado en la moda (Tierney y Kadane, 1986; Kass, Tierney y Kadane, 1989).

Otros autores como, por ejemplo, Naylor y Smith (1982, 1988), utilizan métodos de integración numérica y más concretamente, fórmulas de cuadratura para llevar a cabo el cálculo de las integrales anteriores. Dichos métodos, aunque muy precisos en problemas de dimensionalidad pequeña, pueden no trabajar muy bien en problemas grandes (Thisted, 1988).

Los métodos más utilizados son los basados en el método de Monte Carlo y se revisan, brevemente, en la sección siguiente.

A.4 Métodos de simulación

El método de Monte Carlo está basado en obtener muestras $\{\boldsymbol{\theta}^{(k)}; k=1, \dots, K\}$ de la distribución a posteriori mediante simulación estocástica. A partir de dichas muestras se obtendrán estimaciones de las características de la distribución a posteriori a partir de las correspondientes características muestrales.

Así, por ejemplo, se pueden estimar momentos a posteriori de la forma $E[f(\boldsymbol{\theta})|\mathbf{Y}]$ mediante promedios de la forma $\frac{1}{K} \sum_{k=1}^K f(\boldsymbol{\theta}^k)$, o intervalos de credibilidad para una componente θ_i del vector de parámetros, con una probabilidad dada α dado por $(\hat{\theta}_i(\alpha/2); \hat{\theta}_i(1-(\alpha/2)))$ siendo $0 < \alpha < 0.5$.

Además, se pueden calcular estimadores de la función de densidad de su distribución a posteriori mediante histogramas o estimadores kernel de la densidad, cuando se necesite información explícita sobre la distribución. Así

mismo, se puede aumentar la precisión de estos estimadores, simplemente aumentando el tamaño K de la muestra.

Dentro de estos métodos podemos distinguir dos tipos: los no iterativos y los métodos MCMC (*Monte Carlo Markov Chain*). Los métodos no iterativos pueden ser a su vez directos e indirectos.

A.4.1 Métodos no iterativos directos

El muestreo directo dentro de la familia de métodos no iterativos está basado en la integración de Monte Carlo.

Supongamos $\boldsymbol{\theta}$ con distribución $h(\boldsymbol{\theta})$ y queremos calcular la esperanza de una función suya $E[f(\boldsymbol{\theta})] = \int f(\boldsymbol{\theta})h(\boldsymbol{\theta})d\boldsymbol{\theta}$, entonces si $\theta_1, \dots, \theta_K$ son independientes e igualmente distribuidas con distribución $h(\boldsymbol{\theta})$, tenemos que $E[f(\boldsymbol{\theta})] \approx \frac{1}{K} \sum_{j=1}^K f(\boldsymbol{\theta}_j)$ y por la Ley de los grandes números convergerá a $E[f(\boldsymbol{\theta})]$ con probabilidad 1 cuando $K \rightarrow \infty$.

Este resultado se puede aplicar de forma directa a la situación en la que $h(\boldsymbol{\theta})$ sea la distribución a posteriori y $E[f(\boldsymbol{\theta})]$ la media a posteriori de $f(\boldsymbol{\theta})$. De este modo, el cálculo de esperanzas a posteriori solo requiere una muestra de tamaño K de la distribución a posteriori.

Esta aproximación será mejor cuanto mayor sea K . Además, presenta la ventaja de que K es independiente del tamaño de los datos de que se disponga. Otra virtud adicional de estos métodos sobre los asintóticos es que se puede evaluar su precisión para un K dado.

Pero desafortunadamente, no siempre es posible obtener una muestra aleatoria simple de la distribución a posteriori, siendo necesario recurrir a métodos menos precisos que se describen a continuación.

A.4.2 Métodos no iterativos indirectos

Cuando no es posible muestrear directamente de la distribución a posteriori, se utilizan otros métodos indirectos y entre los que destacan el muestreo de aceptación y rechazo y el muestreo de importancia.

A.4.2.1 Muestreo de aceptación rechazo

Se utilizan cuando no se puede generar muestras de la distribución a posterior y sin embargo es posible muestrear de otra densidad $q(\cdot)$ llamada densidad propuesta, que verifique la siguiente condición:

$$p(\boldsymbol{\theta} | \mathbf{Y}) \leq Mq(\boldsymbol{\theta}) \text{ para todo valor de } \boldsymbol{\theta} \text{ del dominio de } p$$

El método consiste en generar muestras candidatas de la densidad propuesta $q(\cdot)$ y aceptarlas como muestras de la densidad objetivo $p(\boldsymbol{\theta} | \mathbf{Y})$

con probabilidad $\frac{p(\boldsymbol{\theta} | \mathbf{Y})}{Mq(\boldsymbol{\theta})}$

El algoritmo opera como sigue:

Paso 1: Poner el contador $k=1$.

Paso 2: Extraer una muestra candidata $\tilde{\boldsymbol{\theta}}_k$ de $g(\cdot)$ y generar u de una distribución uniforme $U(0,1)$.

Paso 3: Aceptar el candidato $\boldsymbol{\theta}_k = \tilde{\boldsymbol{\theta}}_k$ si $u \leq \frac{p(\tilde{\boldsymbol{\theta}}_k | \mathbf{Y})}{Mq(\tilde{\boldsymbol{\theta}}_k)}$ e ir al Paso 4.

En otro caso volver al paso 2.

Paso 4: Si $k < K$ entonces $k = k+1$ y volver al paso 2.

De este modo, las extracciones aceptadas se distribuirán aproximadamente como la densidad objetivo $p(\boldsymbol{\theta} | \mathbf{Y})$. La eficiencia del procedimiento depende de M y es mayor cuanto más próximo está M de 1.

A.4.2.2 Muestreo de importancia

Esta aproximación fue perfilada por Hammersley y Handscomb (1964), pero fueron Kloeck y Van Dijk (1978) y Geweke (1989) quienes la impulsaron en el análisis bayesiano.

El muestreo de importancia está basado en las siguientes igualdades:

$$\begin{aligned} E[g(\boldsymbol{\theta}) | \mathbf{Y}] &= \int_{\Theta} g(\boldsymbol{\theta}) p(\boldsymbol{\theta} | \mathbf{Y}) d\boldsymbol{\theta} = \int_{\Theta} \frac{g(\boldsymbol{\theta}) p(\boldsymbol{\theta} | \mathbf{Y})}{q(\boldsymbol{\theta})} q(\boldsymbol{\theta}) d\boldsymbol{\theta} = \\ &= \int_{\Theta} g(\boldsymbol{\theta}) \omega(\boldsymbol{\theta}) q(\boldsymbol{\theta}) d\boldsymbol{\theta} \end{aligned}$$

donde $q(\boldsymbol{\theta})$ es una densidad propuesta, llamada función de importancia, que tiene que ser elegida y $\omega(\boldsymbol{\theta}) = \frac{p(\boldsymbol{\theta} | \mathbf{Y})}{q(\boldsymbol{\theta})}$ los pesos de importancia. Se tiene

que:

$$E[g(\boldsymbol{\theta}) | \mathbf{Y}] \approx \frac{\sum_{k=1}^K g(\boldsymbol{\theta}^{(k)}) \omega(\boldsymbol{\theta}^{(k)})}{\sum_{k=1}^K \omega(\boldsymbol{\theta}^{(k)})} \quad (\text{A.1})$$

donde $\{\boldsymbol{\theta}^{(k)}; k=1, \dots, K\}$ son extracciones independientes e igualmente distribuidas de la densidad propuesta $q(\boldsymbol{\theta})$.

El método funciona bien si la densidad de importancia $q(\boldsymbol{\theta})$ es una buena aproximación de $p(\boldsymbol{\theta}|Y)$ y sus colas dominan a las de la distribución a posteriori. En estos casos los pesos de importancia serán aproximadamente iguales a uno. En caso contrario la mayor parte de los pesos estarán próximos a cero y unos pocos elementos $\boldsymbol{\theta}^{(k)}$ dominarán las sumas, produciendo estimadores poco precisos pudiendo llegar a ser la varianza de los promedios (A.1) infinita (ver Geweke, 1989 para más detalles).

El algoritmo se expresa como sigue:

Paso 1: Generar K extracciones $\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, \dots, \boldsymbol{\theta}^{(K)}$ de la densidad propuesta $q(\boldsymbol{\theta})$.

Paso 2: Calcular los pesos de importancia $\omega(\boldsymbol{\theta}^{(k)}) = \frac{p(\boldsymbol{\theta}^{(k)} | Y)}{q(\boldsymbol{\theta}^{(k)})}$ para $k=1, \dots, K$.

Paso 3: Calcular las cantidades de interés de la distribución a posteriori que se deseen a través de la ecuación (A.1).

A.4.3 Métodos iterativos. Métodos MCMC

Los métodos no iterativos extraen una única muestra de tamaño N , y lo único que se requiere es que sea suficientemente grande. Pero para problemas de gran dimensionalidad, puede ser difícil o imposible a veces encontrar, por ejemplo, una densidad para el muestreo de importancia. Los métodos MCMC (iniciales de *Monte Carlo Markov Chain*) permiten hacer inferencias de las

distribuciones a posteriori en problemas de gran dimensión reduciéndolos a un proceso recursivo resolviendo una serie de problemas de menor dimensionalidad, a menudo unidimensionales. Del mismo modo que los métodos tradicionales de Monte Carlo, los métodos MCMC no producen formas cerradas de la distribución a posteriori, sino una muestra de ella.

La idea general es construir una cadena de Markov que tenga la distribución a posteriori como distribución estacionaria y que converja a ella. Simulando la cadena un número suficientemente grande de iteraciones, hasta que se haya alcanzado la convergencia, es posible extraer muestras, habitualmente dependientes, de dicha distribución mediante el método de Monte Carlo, y a partir de dichas muestras, obtener estimadores de la densidad a posteriori. De hecho los métodos MCMC permiten cálculos inferenciales por medio de aproximaciones del siguiente modo:

$$E[g(\boldsymbol{\theta}|\mathbf{Y})] \approx \frac{1}{L} \sum_{\ell=1}^L g(\boldsymbol{\theta}^{(k_0+(\ell-1)s)}) \quad \text{con } k_0 + (L-1)s = K \quad (\text{A.2})$$

siendo $\{\boldsymbol{\theta}^{(k)}; k=1, \dots, K\}$ todas las muestras generadas en el proceso, k_0 el final del periodo de convergencia (*burn-in*), en el cual la cadena de Markov ha alcanzado finalmente la condición de estacionariedad, es decir, que las muestras generadas a partir de ese instante proceden de la distribución estacionaria y s es el intervalo de muestreo cogiendo muestras cada s pasos. Dicho valor se determina de forma que la correlación entre las muestras $\{g(\boldsymbol{\theta}^{(k_0+(\ell-1)s)}); \ell=1, \dots, L\}$ sea prácticamente despreciable con el fin de disminuir significativamente el grado de dependencia existente entre los elementos de la muestra y aumentar, de este modo, la precisión del estimador (A.2).

Así pues, los pasos que se requieren para calcular estimadores a posteriori por medio de estos métodos MCMC se pueden resumir como:

- Paso 1:** Comenzar desde un punto arbitrario del parámetro $\theta^{(0)}$ desconocido.
- Paso 2:** Generar K muestras de la cadena de Markov cuyo kernel de transición es $T(\theta^{(k)}|\theta^{(k-1)})$ y que admite como distribución estacionaria la distribución a posteriori.
- Paso 3:** Descartar las k_0 primeras muestras que corresponden al periodo de entrenamiento/calentamiento burn-in, estimar s analizando el correlograma de las componentes de $\{\theta^{(k_0+k)}; k = 0, \dots, K - k_0 + 1\}$ y calcular los estimadores necesarios a través de la ecuación (A.2)

De forma básica este es la estructura que utilizan los algoritmos MCMC y se diferencian unos de otros en la forma en la que se define el paso 2.

La diferencia con los métodos tradicionales de Monte Carlo, viene del hecho de que los métodos MCMC generan muestras dependientes de la distribución a posteriori, puesto que se obtienen de una cadena de Markov, cuya distribución estacionaria es la misma que la distribución de la que se quiere muestrear, que en este caso es la distribución a posteriori.

La convergencia de la cadena de Markov a la distribución estacionaria correcta, está garantizada para la mayor parte de las distribuciones a posteriori. No obstante, esta cuestión lleva al planteamiento de dos problemas, por una parte decidir cuándo es adecuado parar la ejecución del algoritmo y proceder a realizar las inferencias, lo cual es conocido como el diagnóstico de la convergencia y por otra la calidad de las estimaciones obtenidas, puesto que

no están basadas en muestras independientes e igualmente distribuidas de la distribución a posteriori, sino en muestras correladas.

A continuación se revisarán dentro de los métodos MCMC los dos algoritmos más utilizados en la práctica: el algoritmo de Hasting-Metropolis y el algoritmo de *Gibbs sampling*.

A.4.3.1 Muestreo de Hastings-Metropolis

Históricamente es el primero de los métodos MCMC. Está basado en los trabajos de Metropolis y otros (1953) y Hastings (1970). Inicialmente fue desarrollado por físicos que trabajaban en el proyecto Manhattan (sobre teoría de partículas) y fue rescatado por los estadísticos en los años 90 (Gilks y otros, 1996)

Se basa en la elección de una distribución de transición instrumental condicional $q(\theta^{(k)}|\theta^{(k-1)})$ generalización de la distribución independiente $q(\theta^{(k)})$ del algoritmo de aceptación-rechazo y que es una densidad válida para cada posible valor de la variable que condiciona $\theta^{(k-1)}$.

El algoritmo se expresa de la siguiente manera.

Paso 1: Inicializar la cadena seleccionando un punto arbitrario de comienzo $\theta^{(0)}$ en el instante $k=0$.

Paso 2: En el paso k , extraer una muestra θ^* de $q(\cdot|\theta^{(k-1)})$

Paso 3: Aceptar la muestra $\theta^{(k)} = \theta^*$ con probabilidad

$$\alpha(\theta^{(k-1)}, \theta^*) = \min \left\{ \frac{q(\theta^{(k-1)}|\theta^*)p(\theta^*)}{q(\theta^*|\theta^{(k-1)})p(\theta^{(k-1)})}, 1 \right\}$$

en otro caso tomar $\theta^{(k)} = \theta^{(k-1)}$.

Este paso de aceptación se lleva a cabo generando una extracción aleatoria u de la distribución uniforme $U(0,1)$. Si $u < \alpha(\theta^{(k-1)}, \theta^*)$, entonces $\theta^{(k)} = \theta^*$; en otro caso $\theta^{(k)} = \theta^{(k-1)}$.

Paso 4: Si $k < K$, incrementar $k=k+1$ y volver al paso 2.

Paso 5: Eliminar las muestras del periodo de entrenamiento burn-in y calcular los estimadores necesarios.

Este algoritmo tiene como principal ventaja que impone pocas limitaciones teóricas a la elección de la función de exploración. Sin embargo, presenta un posible inconveniente de tipo práctico: la velocidad de convergencia del algoritmo depende de la elección de q .

A.4.3.2 Gibbs sampling

Este algoritmo fue introducido a través de los trabajos de Geman y Geman (1984), Gelfand y Smith (1990) y Gelfand y otros (1990). Se utiliza cuando se quiere obtener una muestra de una distribución de probabilidad complicada analíticamente.

Sea $\theta = (\theta_1, \theta_2, \dots, \theta_k)'$ el parámetro del modelo. El muestreo de Gibbs asume que se pueden generar muestras de todas las distribuciones completamente condicionadas $\{p(\theta_i | \theta_{j \neq i}, Y), i=1, \dots, k\}$ donde $\theta_{j \neq i}$ denota el vector resultante de quitar al vector de parámetros θ la componente θ_i . En el caso de que las distribuciones completamente condicionadas tengan formas conocidas, las muestras están disponibles de forma directa, y; en otro caso, de forma indirecta a través de una aproximación como, por ejemplo, el muestreo

de rechazo o el algoritmo de Hastings-Metropolis. En cualquiera de estos casos bajo condiciones no muy fuertes (ver Robert y Casella, 2004, capítulos 9 y 10 para más detalles), la colección de las distribuciones completamente condicionadas, determinan de forma única la distribución conjunta a posteriori $p(\boldsymbol{\theta}|Y)$ y, por tanto, todas las distribuciones marginales a posteriori $p(\theta_i|Y)$ $i=1,\dots,k$.

Dado un punto arbitrario de comienzo $\{\theta_2^{(0)}, \dots, \theta_k^{(0)}\}$ el algoritmo actúa de la siguiente manera:

Para t de 1 a T

Paso 1: Extraer $\theta_1^{(t)}$ de $p(\theta_1 | \theta_2^{(t-1)}, \theta_3^{(t-1)}, \dots, \theta_k^{(t-1)}, Y)$

Paso 2: Extraer $\theta_2^{(t)}$ de $p(\theta_2 | \theta_1^{(t)}, \theta_3^{(t-1)}, \dots, \theta_k^{(t-1)}, Y)$

.....

Paso k: Extraer $\theta_k^{(t)}$ de $p(\theta_k | \theta_1^{(t)}, \theta_3^{(t)}, \dots, \theta_{k-1}^{(t)}, Y)$

Bajo condiciones de regularidad que generalmente se satisfacen en la mayor parte de los modelos estadísticos (ver Robert y Casella, 2004 capítulo 10 para más detalles) la cadena así construida es irreducible, aperiódica, recurrente Harris positiva y geoméricamente ergódica por lo que la cadena convergerá a su distribución estacionaria $p(\boldsymbol{\theta} | Y)$. En nuestro caso no es difícil comprobar que la densidades completamente condicionadas verifican la condición de positividad detallada en el lema 10.11 de Robert y Casella (2004) al ser todos sus soportes convexos. Por lo tanto los algoritmos propuestos en la Memoria convergerán a la distribución a posteriori correspondiente.

Se obtiene así una k-tupla en la iteración t , $(\theta_1^{(t)}, \dots, \theta_k^{(t)})$, que converge en distribución a una muestra de la distribución a posteriori conjunta $p(\theta_1, \dots, \theta_k | Y)$ de forma que, para t_0 suficientemente grande, $\{\theta^{(t)}, t=t_0+1, \dots, T\}$ es una muestra dependiente de dicha distribución, siendo t_0 el periodo de entrenamiento o *burn-in*, con la que se estiman las cantidades a posteriori de interés.

A.4.3.2.1 Rao-Blackwellización

A la hora de calcular momentos a posteriori de la forma $E[g(\boldsymbol{\theta} | \mathbf{Y})]$ existe un método (ver Casella y Robert, 1996) que mejora los resultados de las extracciones en los métodos MCMC, y especialmente en el caso del *Gibbs sampling* y proviene del teorema de inferencia estadística clásica de Blackwell-Rao.

Suponer que el vector de parámetros $\boldsymbol{\theta}$ se puede partir en dos componentes $(\boldsymbol{\theta}_1', \boldsymbol{\theta}_2')$ de forma que $u(\boldsymbol{\theta}_1) = E[g(\boldsymbol{\theta} | \mathbf{Y}) | \boldsymbol{\theta}_1]$ se puede calcular de forma exacta. En este caso un estimador de $E[g(\boldsymbol{\theta} | \mathbf{Y})]$ más eficiente que (A.2) viene dado por:

$$\frac{1}{L} \sum_{\ell=1}^L u(\boldsymbol{\theta}_1^{(k_0 + (\ell-1)s)}) \text{ con } k_0 + (L-1)s = K$$

A.4.3 Diagnostico de convergencia en métodos MCMC

La implementación de los métodos MCMC requiere tener en cuenta las cuestiones relacionadas con la convergencia, tanto en cuanto a la duración del periodo de convergencia “*burn in*”, como en el punto en el que el algoritmo

debe finalizar. El periodo de entrenamiento representa el periodo en el que la cadena de Markov no ha alcanzado la estacionariedad y está todavía muy influenciado por el punto de partida.

El número de iteraciones que se requieren tras el periodo de entrenamiento determinará el número de muestras que deberán ser extraídas después de que la convergencia se haya alcanzado para calcular los momentos que sean necesarios de la distribución a posteriori. Si se ha alcanzado la convergencia, todas las muestras son idénticamente distribuidas, sin embargo son muestras autocorreladas. En general la convergencia suele ser rápida y de tipo geométrico pero, no obstante, conviene aplicar algún tipo de método gráfico o numérico para analizar si este es el caso (ver Robert y Casella, 2004 Capítulo 12 para una revisión reciente de diversos métodos).

Entre éstas se pueden destacar el método de Raftery y Lewis, 1996, el de Geweke, 1992 y el de Gelman y Rubin, 1992. El primero de ellos, tiene como objetivo calcular la longitud del periodo de entrenamiento y el número total de iteraciones requeridas para estimar un cuantil dado de la distribución a posteriori. Está basado en una única cadena de Markov, y requiere la introducción de un cuantil que debe ser estimado, la precisión para él y una probabilidad de que el cuantil estimado tenga la precisión dada con una probabilidad prefijada. Habitualmente, se utiliza el cuantil 0.025 con una precisión 0.005 y una probabilidad de 0.95.

El segundo está basado en considerar dos subconjunto de una cadena simple, (habitualmente el primer 10% y el último 50%) en el cual se ha obviado el periodo de entrenamiento. Se realiza un test suponiendo que las medias de ambos conjuntos son iguales. Si no se rechaza la igualdad de medias, las muestras se supone que provienen de la misma distribución estacionaria y significa que la convergencia ha sido alcanzada.

El último propone llevar a cabo varias simulaciones empezando en diferentes valores de los parámetros, elegidos suficientemente dispersos en referencia a la densidad objetivo. Una vez que estos valores han sido escogidos, el algoritmo se basa en mezclar cadenas simuladas y comparar la varianza dentro de cada cadena con respecto a la varianza total de la mixtura de cadenas por medio de un coeficiente R “*estimated potencial scale reduction*” que cuando las simulaciones convergen desciende hacia 1, indicando de esta forma que las cadenas paralelas se superponen esencialmente.

No obstante es también habitual, proceder a poner en marcha varias cadenas paralelamente en lugar de una sola. La inspección visual de estas cadenas puede ser útil para asegurar la convergencia de la muestra y además no añade mucho más tiempo computacional

Si la cadena tarda en converger ello puede ser debido a una mala parametrización del modelo que provoque problemas de quasi-identificabilidad entre algunas de las componentes de θ y, en este caso, se aconseja o bien reparametrizar θ , o bien agrupar las componentes de θ a la hora de calcular las distribuciones completamente condicionadas o, si fuera posible, eliminar algunos de los parámetros del modelo mediante marginalización.

Apéndice B

Programas informáticos

Este apéndice contiene los programas informáticos utilizados en la realización de la Memoria elaborados con el paquete MATLAB 6.5.

Programas correspondientes al capítulo 2

Programa 2.1.- Esta función calcula las correlaciones espaciales de los residuos contenidos en el vector Y para un número nmax de vecinos. Devuelve las correlaciones junto con un gráfico de las mismas.

```
function correlaciones = autocorrelaciones_espaciales(Y,IS,S,ind);
n = size(S,1);
nmax = size(S,2);
correlaciones = zeros(nmax,1);
for(i=1:nmax)
    aux = IS(:,i)*Y(S(:,i));
    Y1 = Y(aux>0);
    aux = aux(aux>0);
    aux = corrcoeff(Y1,aux);
    correlaciones(i) = aux(1,2);
end
correlaciones = [1;correlaciones];
if(ind==1)
    plot(0:nmax,correlaciones,'b*',0:nmax,correlaciones,'b-',0:nmax,
        zeros(nmax+1,1),'r--',0:nmax,(3/sqrt(n))*ones(nmax+1,1),'r:',
        0:nmax,(3/sqrt(n))*ones(nmax+1,1),'r:');
    xlabel('Numero de vecinos');
    ylabel('Autocorrelaciones empiricas');
end
```

Programa 2.2.- Esta función calcula las matrices TX, SX, TSX, STX, TY, SY, TSY y STY en el modelo STAR para un conjunto de valores de los parametros mT, gamma, mS y lambda dados.

```
function [TX,SX,TSX,STX,TY,SY,TSY,STY] =
calculo_TS_c(Y,X,IT,T,IS,S,mT,gamma,mS,lambda);
% Creacion de las variables auxiliares
n = length(Y);
p = size(X,2);
IT1 = IT>0;
IS1 = IS>0;
% Calculo de TX y TY
if(p>0)
    TX = zeros(n,p);
else
    TX = zeros(n,1);
end
TY = zeros(n,1);
if(mT>0)
    auxT = gamma.^IT;
    sumasT = IT1(:,1:mT).*auxT(:,1:mT);
    sumasT = sum(sumasT)';
    for(i=1:mT)
        TY(sumasT>0) = TY(sumasT>0) + (auxT(sumasT>0,i).*
Y(T(sumasT>0,i)).*IT1(sumasT>0,i))./sumasT(sumasT>0);
        for(j=1:p)
            TX(sumasT>0,j) = TX(sumasT>0,j) + (auxT(sumasT>0,i).*
X(T(sumasT>0,i),j)).*IT1(sumasT>0,i))./sumasT(sumasT>0);
        end
    end
end
if(p>0)
    SX = zeros(n,p);
    STX = zeros(n,p);
    TSX = zeros(n,p);
else
    SX = zeros(n,1);
    STX = zeros(n,1);
    TSX = zeros(n,1);
end
SY = zeros(n,1);
STY = zeros(n,1);
TSY = zeros(n,1);
if(mS>0)
    auxS = lambda.^IS;
    sumas = IS1(:,1:mS).*auxS(:,1:mS);
    sumas = sum(sumas)';
    % Calculo de SX y SY
    for(k=1:mS)
        SY(sumas>0) = SY(sumas>0) + (auxS(sumas>0,k).*
Y(S(sumas>0,k)).*IS1(sumas>0,k))./sumas(sumas>0);
        for(j=1:p)
            SX(sumas>0,j) = SX(sumas>0,j)+(auxS(sumas>0,k).*
X(S(sumas>0,k),j)).*IS1(sumas>0,k))./sumas(sumas>0);
        end
    end
end
if(mT>0)
```

Programas correspondientes al capítulo 2

```

% Calculo de STX y STY
for(k=1:mS)
    STY(sumas>0) = STY(sumas>0) + (auxS(sumas>0,k).*
    TY(S(sumas>0,k)).*IS1(sumas>0,k))./sumas(sumas>0);
    for(j=1:p)
        STX(sumas>0,j) = STX(sumas>0,j) + (auxS(sumas>0,k).*
        TX(S(sumas>0,k,j)).*IS1(sumas>0,k))./sumas(sumas>0);
    end
end
% Calculo de TSX y TSY
for(i=1:mT)
    TSY(sumasT>0) = TSY(sumasT>0) + (auxT(sumasT>0,i).*
    SY(T(sumasT>0,i)).*IT1(sumasT>0,i))./sumasT(sumasT>0);
    for(j=1:p)
        TSX(sumasT>0,j) = TSX(sumasT>0,j) + (auxT(sumasT>0,i).*
        SX(T(sumasT>0,i,j)).*IT1(sumasT>0,i))./sumasT(sumasT>0);
    end
end
end
end
end

```

Programa 2.3.- Esta función presenta los resultados obtenidos al estimar un modelo STAR mediante la metodología descrita en el Capítulo 2. Las variables índicesZ e índicesX contienen las componentes de las matrices Z y X utilizadas en la estimación del modelo. Las variables indicadoras índicesTS valen 1 si el modelo contiene la componente correspondiente y 0 en otro caso.

```

% Estimación del modelo completo
[alfas_mle,betas_mle,fis_mle,std_parametros,sigma_mle,BIC,res,R2,LLF, LOSS]
=f_mle_estandar(Y,Z,X,1:4,1:14,[1,1,1,1,0,0,0,0],IT,T,IS,S,mT,
gamma,mS,lambda,n0,0);
% Estimación del modelo reducido
[alfas_mle,betas_mle,fis_mle,std_parametros,sigma_mler,BICr,rsr,R2r,LLFr,
LOSSr] = f_mle_estandar(Y,Z,X,indicesZ,indicesX,indicesTS,IT,T,IS,S,mT,
gamma,mS,lambda,n0,0);
sigma_mler = sigma_mler;
zr = resr/sqrt(sigma_mler);
CUB95 = 100*mean(abs(zr)<=1.96);
CUB99 = 100*mean(abs(zr)<=2.58);
% Residuos tipificados intramuestrales
nfigura = 1;
figure(nfigura);
plot(1:length(zr),zr,1:length(zr),-2*ones(length(zr),1),'r',1:length(zr),2*ones(length(zr),1),'r');
xlim([1 length(zr)]);
xlabel('Observacion');
% Impresión de los coeficientes y sus errores estándar
titas_mle = [alfas_mle;betas_mle;fis_mle];
estimaciones_parametros = strcat(strvcat(literal_z,literal_x,literal_fis),
num2str([titas_mle,std_parametros,titas_mle./std_parametros,2*(1-
normcdf(abs(titas_mle./std_parametros))))))

```

```

% Estudio de los errores intramuestrales tipificados y no tipificados
estudio_zr_intra = [min(zr),prctile(zr,[1,5,10,25,50,75,90,95,99]),max(zr),
mean(zr),median(abs(zr)),skewness(zr),kurtosis(zr)]'
% Histograma
nfigura = nfigura+1;
figure(nfigura)
[f,xi] = ksdensity(zr);
plot(xi,f);
% qqplot
nfigura = nfigura+1;
figure(nfigura)
qqplot(zr);
% Autocorrelacion residual
nfigura = nfigura+1;
figure(nfigura)
correlaciones = autocorrelaciones_espaciales(resr,IS1,S1,1);
nfigura = nfigura+1;
figure(nfigura)
correlaciones = autocorrelaciones_espaciales(resr,IT1,T1,1);
% Calculo del pvalor
npar = length(indicesZ)+length(indicesX)+sum(indicesTSr);
criterios_comparacion = [R2r,sqrt(mean(resr.*resr)),mean(abs(resr)),LOSSr,CUB95,CUB99]'
pvalor = 1-chi2cdf(LLFr-LLF,size(Z,2)+size(X,2)+sum(indicesTS)-npar)
BIC = BICr
clear correlaciones pvalor res resr BIC R2 R2r alfas_mle betas_mle fis_mle std_parametros sigma_mle sigma_mler
LLF LLFr titas_mle;
clear estudios_res_intra estudios_zr_intra estimaciones_parametros zr CUB95 CUB99 criterios_comparacion;
pause;
% Validacion rolling
'Validacion rolling'
% Calculo de las predicciones
nfigura = nfigura+1;
[RMSEr,MADr,CUB95r,CUB99r,LPREDr,errorr] = validacion_rolling(Y,Z,
X,tiempo,indicesZ,indicesX,indicesTS,IT,T,IS,S,mT,gamma,mS,lambda,n0,nval,nfigura);
xlabel('Observacion');
% Analisis de los residuos extramuestrales
nfigura = nfigura+2;
figure(nfigura)
zr = errorr/sqrt(var(errorr));
plot(1:length(errorr),zr,1:length(errorr),-2*ones(length(errorr),1),'r',1:length(errorr),2*ones(length(errorr),1),'r');
xlim([1 length(errorr)]); xlabel('Observacion');
estudios_zr_rolling= [min(zr),prctile(zr,[1,5,10,25,50,75,90,95,99]),max(zr),
mean(zr),median(abs(zr)),skewness(zr),kurtosis(zr)]'
% Histograma
nfigura = nfigura+1;
figure(nfigura)
[f,xi] = ksdensity(zr);

```

Programas correspondientes al capítulo 2

```
plot(xi,f);
% qqplot
nfigura = nfigura+1;
figure(nfigura)
qqplot(zr);
% Autocorrelaciones residuales extramuestrales
nfigura = nfigura+1;
figure(nfigura)
correlaciones = autocorrelaciones_espaciales(errorr,IS2,S2,1);
nfigura = nfigura+1;
figure(nfigura)
correlaciones = autocorrelaciones_espaciales(errorr,IT2,T2,1);
% Criterios de prediccion
R2r = 1-(RMSEr*RMSEr)/var(Y(nval:end));
criterios_prediccion = [R2r,RMSEr,MADr,CUB95r,CUB99r,LPREDr]
pause;
% Validacion extramuestral
'Validacion extramuestral'
% Calculo de las predicciones
nfigura = nfigura+1;
[RMSEr,MADr,CUB95r,CUB99r,LPREDr,errorr] =
validacion_extramuestral(Y,Z,X,indicesZ,indicesX,indicesTS,IT,T,IS,S,mT,gamma,mS,lambda,n0,nval,nfigura);
xlabel('Observacion');
% Analisis de los residuos extramuestrales
nfigura = nfigura+2;
figure(nfigura)
zr = errorr/sqrt(var(errorr));
plot(1:length(errorr),zr,1:length(errorr),-2*ones(length(errorr),1),'r',1:length(errorr),2*ones(length(errorr),1),'r');
xlim([1 length(errorr)]); xlabel('Observacion');
estudios_zr_extra = [min(zr),prctile(zr,[1,5,10,25,50,75,90,95,99]),max(zr),
mean(zr),median(abs(zr)),skewness(zr),kurtosis(zr)]'
% Estimador kernel de la densidad
nfigura = nfigura+1;
figure(nfigura)
[f,xi] = ksdensity(zr);
plot(xi,f);
% qqplot
nfigura = nfigura+1;
figure(nfigura)
qqplot(zr);
% Autocorrelaciones residuales extramuestrales
nfigura = nfigura+1;
figure(nfigura)
correlaciones = autocorrelaciones_espaciales(errorr,IS2,S2,1);
nfigura = nfigura+1;
figure(nfigura)
correlaciones = autocorrelaciones_espaciales(errorr,IT2,T2,1);
```

```
% Criterios de prediccion
R2r = 1-(RMSEr*RMSEr)/var(Y(nval:end));
criterios_prediccion = [R2r,RMSEr,MADr,CUB95r,CUB99r,LPREDr]
clear estudios_res_rolling errorr RMSEr MADr CUB95r CUB99r LPREDr criterios_prediccion zr indicesX indicesZ
indicesTS correlaciones;
clear estudios_zr_rolling estudios_res_extra estudios_zr_extra;
```

Programa 2.4.- Esta funcion calcula el estimador MLE del modelo de Pace y otros (2000) teniendo en cuenta las variables y componentes seleccionados. Utiliza para ello la funcion fpace y la rutina nlinfit. El modelo que utiliza es:

$$Y = Z*\alpha + X*\beta + \text{fis}(1)*\text{indTX}*T*\text{error} + \text{fis}(2)*\text{indSX}*S*\text{error} + \text{fis}(3)*\text{indTSX}*T*S*\text{error} + \text{fis}(4)*\text{indSTX}*S*T*\text{error} + \text{fis}(5)*\text{indTY}*T*Y + \text{fis}(6)*\text{indSY}*S*Y + \text{fis}(8)*\text{indTSY}*T*S*Y + \text{fis}(8)*\text{indST}*S*T*Y$$

donde error = Y - X*beta.

Las variables indicesZ e indicesX contienen las componentes de las matrices Z y X utilizadas en la estimacion del modelo. Las variables indicadoras indicesTS valen 1 si se contiene la componente correspondiente y 0 en otro caso.

Devuelve el valor del estimador MLE asi como los residuos del modelo y los valores de los criterios BIC y R2.

```
function [alfas_mle,betas_mle,fis_mle, sigma_mle,res,BIC, R2] = f_mle(Y,Z,
X,indicesZ,indicesX,indicesTS,IT,T,IS,S,mT,gamma,mS,lambda,n0);
% Creacion de las variables auxiliares y de salida
pz = size(Z,2);
p = size(X,2);
alfas_mle = zeros(pz,1);
betas_mle = zeros(p,1);
fis_mle = zeros(8,1);
Z = Z(:,indicesZ);
X = X(:,indicesX);
n = size(X,1);
pz= size(Z,2);
p = size(X,2);
if(mT==0)
    indmT=0;
else
    indmT=1;
end
if(mS==0)
    indmS=0;
else
    indmS=1;
end
if(mT==0)
    indicesTS([1,3,4,5,7,8]) = 0;
```


Programas correspondientes al capítulo 2

```

end
if(mS==0)
    indicesTS([2,3,4,6,7,8]) = 0;
end
indicesfis = indicesTS.*[1:8]';
indicesfis = indicesfis(indicesfis>0);
nfis = length(indicesfis);
% Construccion de la matriz de diseño
if(p==0)
    X = zeros(n,1);
end
[TX,SX,TSX,STX,TY,SY,TSY,STY] = calculo_TS_c(Y,X,IT,T,IS,S,mT,gamma,mS,lambda);
U = [];
if(pz>0)
    U = Z;
end
if(p>0)
    U = [U,X];
end
U = [U,TX,SX,TSX,STX,TY,SY,TSY,STY];
% Calculo del valor inicial de beta necesario para utilizar la rutina
% nlinfit
X1 = U((n0+1):end,1:(p+pz));
Y1 = Y((n0+1):end);
betas0 = inv(X1'*X1)*(X1'*Y1);
betas0 = [betas0;zeros(nfis,1)];
U = [zeros(n,1),U];
U = U((n0+1):end,:);
U(1,1) = pz;
U(2,1) = p;
U(3,1) = nfis;
if(nfis>0)
    U(4:(3+nfis),1) = indicesfis;
end
% Calculo del estimador MLE utilizando la funcion @fpace y nlinfit
[betas,res,aux] = nlinfit(U,Y1,@fpace,betas0);
% Asignacion de los resultados del algoritmo a las variables de salida
if(pz>0)
    alfas_mle(indicesZ) = betas(1:pz);
end
if(p>0)
    betas_mle(indicesX) = betas((pz+1):(p+pz));
end
fis_mle(indicesfis) = betas((p+pz+1):end);
sigma_mle = mean(res.*res);
BIC = (n-n0)*(log(sigma_mle)+1+log(2*pi)) + (p+nfis+pz+2*indmT+2*
indmS)*log(n-n0+1);

```

R2 = 1-sigma_mle/var(Y1);

Programa 2.5.- Esta funcion calcula el estimador MLE del modelo de Pace y otros (2000) teniendo en cuenta las variables y componentes seleccionados. El modelo que utiliza es:

$$Y = Z*\alpha + X*\beta + \text{fis}(1)*\text{indTX}*T*\text{error} + \text{fis}(2)*\text{indSX}*S*\text{error} + \text{fis}(3)*\text{indTSX}*T*S*\text{error} + \text{fis}(4)*\text{indSTX}*S*T*\text{error} + \text{fis}(5)*\text{indTY}*T*Y + \text{fis}(6)*\text{indSY}*S*Y + \text{fis}(8)*\text{indTSY}*T*S*Y + \text{fis}(8)*\text{indST}*S*T*Y$$

donde error = Y - X*beta.

Las variables indicesZ e indicesX contienen las componentes de las matrices Z y X utilizadas en la estimacion del modelo. Las variables indicadoras indicesTS valen 1 si se contiene la componente correspondiente y 0 en otro caso.

Devuelve el valor del estimador MLE asi como el del criterio BIC y el error estandar de los coeficientes ind2pasos = 1 si se quiere calcular con mle en dos pasos y 0 en otro caso

```
function [alfas_mle,betas_mle,fis_mle,std_parametros,sigma_mle,BIC,res,R2, LLF,SPL]
=f_mle_estandar(Y,Z,X,indicesZ,indicesX,indicesTS,IT,T,IS,S,mT,
gamma,mS,lambda,n0,ind2pasos);
% Estimacion de los parametros y calculo de los residuos y los criterios BIC,
% R2 y -2*logaritmo de la funcion de verosimilitud (LLF)
if(ind2pasos==0)
    [alfas_mle,betas_mle,fis_mle, sigma_mle,res,BIC, R2] = f_mle(Y,Z,
    X,indicesZ,indicesX,indicesTS,IT,T,IS,S,mT,gamma,mS,lambda,n0);
else
    [alfas_mle,betas_mle,fis_mle, sigma_mle,res,BIC, R2] =
    f_mle_2pasos(Y,Z,X,indicesZ,indicesX,indicesTS,IT,T,IS,S,mT,
    gamma,mS,lambda,n0);
end
paz = size(Z,2);
pa = size(X,2);
std_parametros = zeros(paz+pa+8,1);
Z = Z(:,indicesZ);
X = X(:,indicesX);
n = size(X,1);
pz = size(Z,2);
p = size(X,2);
if(mT==0)
    indicesTS([1,3,4,5,7,8]) = 0;
end
if(mS==0)
    indicesTS([2,3,4,6,7,8]) = 0;
end
indicesfis = indicesTS.*[1:8];
indicesfis = indicesfis(indicesfis>0);
nfis = length(indicesfis);
LLF = (n-n0)*(log(sigma_mle)+1+log(2*pi));
```

Programas correspondientes al capítulo 2

```

% Calculo de los errores estandar
if(p==0)
    X = zeros(n,1);
end
[TX,SX,TSX,STX,TY,SY,TSY,STY] =
calculo_TS_c(Y,X,IT,T,IS,S,mT,gamma,mS,lambda);
U1 = [];
if(pz>0)
    U1 = [U1,Z((n0+1):end,:)];
end
if(p>0)
    U1 = [U1,X((n0+1):end,:)-fis_mle(1)*TX((n0+1):end,:)-
        fis_mle(2)*SX((n0+1):end,:)-fis_mle(3)*TSX((n0+1):end,:)-
        fis_mle(4)*STX((n0+1):end,:)];
    U2 = TY((n0+1):end)-TX((n0+1):end,:)*betas_mle(indicesX);
    U2 = [U2,SY((n0+1):end)-SX((n0+1):end,:)*betas_mle(indicesX)];
    U2 = [U2,TSY((n0+1):end)-TSX((n0+1):end,:)*betas_mle(indicesX)];
    U2 = [U2,STY((n0+1):end)-STX((n0+1):end,:)*betas_mle(indicesX)];
else
    U2 = zeros(size(U1,1),4);
end
U2 = [U2,TY((n0+1):end),SY((n0+1):end),TSY((n0+1):end),
STY((n0+1):end)];
U1 = [U1,U2(:,indicesfis)];
for(i=1:size(U1,2))
    U1(:,i) = res.*U1(:,i);
end
covar_betas = (sigma_mle*sigma_mle)*inv(U1'*U1);
aux = sqrt(diag(covar_betas));
if(pz>0)
    std_parametros(indicesZ) = aux(1:pz);
end
if(p>0)
    std_parametros(paz+indicesX) = aux((pz+1):(p+pz));
end
std_parametros(pa+paz+indicesfis) = aux((p+pz+1):end);
% Calculo de la perdida SPL
FIT = (n-n0)*sigma_mle;
PARSIMONIA = 0;
if(nfis+p+pz>0)
    U2 = U1;
    for(i=(n0+1):n)
        aux = U2(i-n0,:);
        PARSIMONIA = PARSIMONIA + aux*covar_betas*aux';
    end
end
SPL = FIT+PARSIMONIA;

```

Programa 2.6.- Esta funcion calcula predicciones unipuntuales e intervalos predictivos para un periodo nval, ..., T utilizando el modelo de Pace y otros (2000) teniendo en cuenta las variables y componentes seleccionados. El periodo de estimacion es 1:nest.

El modelo que utiliza es:

$$Y = Z*\alpha + X*\beta + \text{fis}(1)*\text{indTX}*T*\text{error} + \text{fis}(2)*\text{indSX}*S*\text{error} + \text{fis}(3)*\text{indTSX}*T*S*\text{error} + \text{fis}(4)*\text{indSTX}*S*T*\text{error} + \text{fis}(5)*\text{indTY}*T*Y + \text{fis}(6)*\text{indSY}*S*Y + \text{fis}(8)*\text{indTSY}*T*S*Y + \text{fis}(8)*\text{indST}*S*T*Y$$

donde error = Y - X*beta.

Las variables indicesZ e indicesX contienen las componentes de las matrices Z y X utilizadas en la estimacion del modelo. Las variables indicadoras indicesTS valen 1 si se contiene la componente correspondiente y 0 en otro caso.

Devuelve el valor de la prediccion y la anchura de los intervalos del 95% y el 99% de credibilidad (ypred) asi como la varianza de la distribucion predictiva y el parámetro sigma del modelo STAR. Utiliza para ello una distribucion t de Student

```
function [ypred,sigma_pred,sigma] =
f_mle_prediccion(Y,Z,X,indicesZ,indicesX,indicesTS,IT,T,IS,S,mT,gamma,mS,lambda,n0,nest,nval);
% Creacion de las variables auxiliares
Z = Z(:,indicesZ);
X = X(:,indicesX);
n = size(X,1);
pz = size(Z,2);
p = size(X,2);
if(mT==0)
    indicesTS([1,3,4,5,7,8]) = 0;
end
if(mS==0)
    indicesTS([2,3,4,6,7,8]) = 0;
end
indicesfis = indicesTS.*[1:8];
indicesfis = indicesfis(indicesfis>0);
nfis = length(indicesfis);
% Construccion de la matriz de diseño
[TX,SX,TSX,STX,TY,SY,TSY,STY] =
calculo_TS_c(Y,X,IT,T,IS,S,mT,gamma,mS,lambda);
U = [];
if(pz>0)
    U = Z;
end
if(p>0)
    U = [U,X];
end
U = [U,TX,SX,TSX,STX,TY,SY,TSY,STY];
% Calculo del valor inicial de beta necesario para utilizar la rutina
```

Programas correspondientes al capítulo 2

```

% nlinfit
X1 = U((n0+1):end,1:(p+pz));
Y1 = Y((n0+1):end);
betas0 = inv(X1'*X1)*(X1'*Y1);
betas0 = [betas0;zeros(nfis,1)];
U = [zeros(n,1),U];
U = U((n0+1):nest,:);
U(1,1) = pz;
U(2,1) = p;
U(3,1) = nfis;
if(nfis>0)
    U(4:(3+nfis),1) = indicesfis;
end
% Obtencion del estimador MLE mediante la rutina nlinfit
[betas,res,J] = nlinfit(U,Y((n0+1):nest),@f_pace,betas0);
npar = length(betas);
% Calculo de los errores estandar
fis = zeros(8,1);
fis(indicesfis) = betas((p+pz+1):end);
if(pz>0)
    alfas = betas(1:pz);
end
sigma = mean(res.*res);
U1 = [];
if(pz>0)
    U1 = Z((n0+1):end,:);
end
if(p>0)
    U1 = [U1,X((n0+1):end,:)-fis(1)*TX((n0+1):end,:)-fis(2)*SX((n0+1):end,:)-
    -fis(3)*TSX((n0+1):end,:)-fis(4)*STX((n0+1):end,:)];
    betas = betas((pz+1):(p+pz));
    U2 = TY((n0+1):end)-TX((n0+1):end,)*betas;
    U2 = [U2,SY((n0+1):end)-SX((n0+1):end,)*betas];
    U2 = [U2,TSY((n0+1):end)-TSX((n0+1):end,)*betas];
    U2 = [U2,STY((n0+1):end)-STX((n0+1):end,)*betas];
else
    U2 = zeros(size(U1,1),4);
end
U2 =
[U2,TY((n0+1):end),SY((n0+1):end),TSY((n0+1):end),STY((n0+1):end)];
U1 = [U1,U2(:,indicesfis)];
for(j=1:(n-nval+1))
    if(pz>0)
        res(nval-n0-1+j) = Y(nval+j-1)-Z(nval+j-1,)*alfas;
    end
    if(p>0)
        res(nval-n0-1+j) = res(nval-n0-1+j)-X(nval+j-1,)*betas;
    end
end

```

```

res(nval-n0-1+j) = res(nval-n0-1+j)-fis(1)*(TY(nval-n0-1+j)-
TX(nval-n0-1+j,:)*betas);
res(nval-n0-1+j) = res(nval-n0-1+j)-fis(2)*(SY(nval-n0-1+j)-
SX(nval-n0-1+j,:)*betas);
res(nval-n0-1+j) = res(nval-n0-1+j)-fis(3)*(TSY(nval-n0-1+j)-
TSX(nval-n0-1+j,:)*betas);
res(nval-n0-1+j) = res(nval-n0-1+j)-fis(4)*(STY(nval-n0-1+j)-
STX(nval-n0-1+j,:)*betas);
end
res(nval-n0-1+j) = res(nval-n0-1+j)-fis(5)*TY(nval-n0-1+j);
res(nval-n0-1+j) = res(nval-n0-1+j)-fis(6)*SY(nval-n0-1+j);
res(nval-n0-1+j) = res(nval-n0-1+j)-fis(7)*TSY(nval-n0-1+j);
res(nval-n0-1+j) = res(nval-n0-1+j)-fis(8)*STY(nval-n0-1+j);
end
for(i=1:size(U1,2))
    U1(:,i) = res.*U1(:,i);
end
U = U1(1:(nval-n0-1),:);
cov_betas = (sigma*sigma)*inv(U*U);
% Calculo de las predicciones puntuales y los intervalos predictivos del
% 95% y 99% de credibilidad
n = length(Y);
ypred = zeros(n-nval+1,1);
if(pz>0)
    ypred = ypred + Z(nval:end,:)*alfas;
end
if(p>0)
    ypred = ypred + X(nval:end,:)*betas;
    ypred = ypred + fis(1)*(TY(nval:end)-TX(nval:end,:)*betas);
    ypred = ypred + fis(2)*(SY(nval:end)-SX(nval:end,:)*betas);
    ypred = ypred +fis(3)*(TSY(nval:end)-TSX(nval:end,:)*betas);
    ypred = ypred +fis(4)*(STY(nval:end)-STX(nval:end,:)*betas);
end
ypred = ypred + fis(5)*TY(nval:end);
ypred = ypred + fis(6)*SY(nval:end);
ypred = ypred +fis(7)*TSY(nval:end);
ypred = ypred +fis(8)*STY(nval:end);
sigma_pred = sigma*ones(n-nval+1,1);
U = U1((nval-n0):end,:);
for(i=nval:n)
    aux = U(i-nval+1,:);
    sigma_pred(i-nval+1) = sigma +aux*cov_betas*aux';
end
ypred_inf = ypred+tinv(0.025,nval-n0-1-npar)*sqrt(sigma_pred);
ypred_sup = ypred+tinv(0.975,nval-n0-1-npar)*sqrt(sigma_pred);
ypred_inf1 = ypred+tinv(0.005,nval-n0-1-npar)*sqrt(sigma_pred);
ypred_sup1 = ypred+tinv(0.995,nval-n0-1-npar)*sqrt(sigma_pred);

```

Programas correspondientes al capítulo 2

```
ypred = [ypred_inf1,ypred_inf,ypred,ypred_sup,ypred_sup1];
```

Programa 2.7.- Esta función calcula el valor predicho del vector de log-precios Y para el modelo de Pace y otros (2000) en el que la matriz de diseño X se ha construido de forma que, si alguna variable o componente falta, la columna correspondiente está rellena de ceros. La primera columna de X debe contener el número de fis. Después contiene, en el orden que se cita a continuación, los siguientes elementos Z, X, TX, SX, TSX, STX, TY, SY, TSY, STY. Devuelve el valor de las predicciones de la serie Y necesarias para calcular el error cuadrático medio que luego utiliza la rutina nlinfit para obtener el estimador MLE.

```
function ypred = f_pace(beta,X);
% Extracción de las componentes de beta
n = size(X,1);
q = X(1,1);
p = X(2,1);
nfis = X(3,1);
if(nfis>0)
    indices_fis = X(4:(3+nfis),1);
end
X = X(:,2:end);
if(q>0)
    Z = X(:,1:q);
    alfa = beta(1:q);
else
    Z = zeros(n,1);
    alfa = 0;
end
if(p>0)
    X1 = X(:,(q+1):(p+q));
    beta1 = beta((q+1):(p+q));
else
    X1 = zeros(n,1);
    beta1 = 0;
end
% Cálculo de la predicción
ypred = Z*alfa + X1*beta1;
if(nfis>0)
    fis = beta((p+q+1):end);
    X = X(:,(p+q+1):end);
    for(i=1:nfis)
        switch indices_fis(i)
            case 1
                TX = X(:,1:p);
                TY = X(:,4*p+1);
                aux = TY-TX*beta1;
```

```

case 2
    SX = X(:,(p+1):(2*p));
    SY = X(:,4*p+2);
    aux = SY-SX*beta1;
case 3
    TSX = X(:,(2*p+1):(3*p));
    TSY = X(:,4*p+3);
    aux = TSY-TSX*beta1;
case 4
    STX = X(:,(3*p+1):(4*p));
    STY = X(:,4*p+4);
    aux = STY-STX*beta1;
case 5
    TY = X(:,4*p+1);
    aux = TY;
case 6
    SY = X(:,4*p+2);
    aux = SY;
case 7
    TSY = X(:,4*p+3);
    aux = TSY;
case 8
    STY = X(:,4*p+4);
    aux = STY;
end
ypred = ypred + aux*fis(i);
end
end

```

Programa 2.8.- Esta función determina la matriz de proximidad S necesaria para implementar el modelo de Pace y otros (1998, 2000). La devuelve en forma de una matriz n x mS tal que S(i,j) es el indice en el vector de datos Y correspondiente a la j-esima transaccion mas cercana a la transaccion i-esima. Tambien devuelve una matriz IS que señala el rango que tiene cada observación previa a una dada en el ranking formado por las distancias poniendo IS(i,j)=0 si la transaccion j-esima no debe ser utilizada en el calculo de los efectos de vecindad de la transaccion i-esima, X (n x p) contiene los datos utilizados para calcular la distancia. Se supone que las transacciones contenidas en el vector Y ya estan ordenadas por tiempo.

```

function [IS,S] = matriz_proximidad_espacial(tiempo,X,mS);
% Creacion de las variables de salida
n = size(X,1);
S = ones(n,mS);
IS = zeros(n,mS);
IS(1,1) = 1;
% Calculo de las matrices IS y S

```


Programas correspondientes al capítulo 2

```
for(i=2:n)
    % Primero se calculan las distancias de cada transaccion a las
    % transacciones que le preceden
    distancias = zeros(i-1,1);
    distancias1 = distancias;
    for(j=1:(i-1))
        aux = X(i,:)-X(j,:);
        distancias(i-j) = sum(aux.*aux);
        aux = tiempo(i,:)-tiempo(j,:);
        distancias1(i-j) = sum(aux.*aux);
    end
    % Luego se ordenan estas de forma no decreciente eliminando las
    % transacciones cuyo tiempo coincide con el de la transaccion actual
    % y se calculan los rangos por distancias para construir la matriz IS
    % Si hay empates se asigna el rango medio
    [distancias,indices] = sort(distancias);
    distancias1 = distancias1(indices);
    indices1 = i-indices(tiempo(i)~=tiempo(i-indices));
    if(length(indices1)==0)
        indices1 = (i-1):-1:1;
    end
    lugar = 1;
    i1 = 1;
    isindices = zeros(length(indices1),1);
    for(j=2:length(indices1))
        if(distancias(j)>distancias(j-1))
            indices2 = i1:(j-1);
            [aux,indices3] = sort(distancias1(indices2));
            indices2 = indices2(indices3);
            lugar = 1;
            for(k=2:length(indices2))
                if(aux(k)>aux(k-1))
                    isindices((i1+lugar-1):(i1+k-2)) = (2*i1+lugar+k-3)/2;
                    lugar = k;
                end
            end
            isindices((i1+lugar-1):(j-1)) = (i1+lugar+j-2)/2;
            indices1(i1:(j-1)) = indices1(indices2);
            i1 = j;
        end
    end
    indices2 = i1:length(indices1);
    [aux,indices3] = sort(distancias1(indices2));
    indices2 = indices2(indices3);
    lugar = 1;
    for(k=2:length(indices2))
        if(aux(k)>aux(k-1))
```

```

    isindices((i1+lugar-1):(i1+k-2)) = (2*i1+lugar+k-3)/2;
    lugar = k;
end
end
isindices((i1+lugar-1):end) = (i1+lugar+length(indices1)-1)/2;
indices1(i1:end) = indices1(indices2);
indices1 = indices1(1:min(mS,length(indices1)));
S(i,1:length(indices1)) = indices1';
IS(i,1:length(indices1)) = isindices(1:length(indices1));
end

```

Programa 2.9.- Esta función selecciona los valores de mT, gamma, mT y lambda del modelo STAR aplicando el criterio BIC (indR2=0) o R2 (indR2=1). Las variables indicesZ e indicesX contienen las componentes de las matrices Z y X utilizadas en la estimacion del modelo. Las variables indicadoras indicesTS valen 1 si se contiene la componente correspondiente y 0 en otro caso. Denotamos por mTred, gammared, mSred y lambdared la red de valores de mT, gamma, mS y lambda utilizadas en la busqueda de los valores de dichos parámetros. El valor de ind2pasos es 0 si se utiliza el programa f_mle en el calculo del estimador MLE e igual a 1 si se utiliza el programa f_mle_2pasos. Devuelve los valores seleccionados de los parametros de vecindad.

```

function [mT,gamma,mS,lambda] = seleccion_vecinos(Y,Z,X,indicesZ,
indicesX,indicesTS,IT,T,IS,S,mTred,gammared,mSred,lambdared,n0,indR2,
ind2pasos);
% Inciacion de las variables
p = size(X,2);
pz = size(Z,2);
n = size(X,1);
nSred = length(mSred);
nTred = length(mTred);
nlambda = length(lambdared);
ngamma = length(gammared);
if(indR2==1)
    critmin = 1;
else
    critmin = 0;
end
% Calculo de los valores optimos de los parametros de vecindad segun el
% criterio elegido
for(i=1:nTred)
    if(mTred(i)==0)
        ngamma1 = 1;
    else
        ngamma1 = ngamma;
    end
    for(j=1:nSred)

```

Programas correspondientes al capítulo 2

```
[i,j]
if(mSred(j)==0)
    nlambda1 = 1;
else
    nlambda1 = nlambda;
end
for(k=1:nlambda1)
    for(k1 = 1:ngamma1)
        if(ind2pasos==0)
            [alfas_mle1,betas_mle1,fis_mle1,sigma_mle1,res,BIC,R2] =
                f_mle(Y,Z,X,indicesZ,indicesX,indicesTS,IT,T,IS,S,mTred(i),
                    gammared(k1),mSred(j),lambdared(k),n0);
        else
            [alfas_mle1,betas_mle1,fis_mle1,sigma_mle1,res,BIC,R2] =
                f_mle_2pasos(Y,Z,X,indicesZ,indicesX,indicesTS,IT,T,IS,S,
                    mTred(i),gammared(k1),mSred(j),lambdared(k),n0);
        end
        if(indR2==1)
            criterio = 1-R2;
        else
            criterio = BIC;
        end
        if(criterio<critmin)
            ind_vecinos = [i,k1,j,k]
            critmin = criterio
        end
    end
end
end
end
end
% Asignacion de los valores calculados a las variables de salida
mT = mTred(ind_vecinos(1));
mS = mSred(ind_vecinos(3));
lambda = lambdared(ind_vecinos(4));
gamma = gammared(ind_vecinos(2));
```

Programa 2.10.- Esta función realiza una validación extramuestral del modelo STAR utilizando un proceso de estimación rolling a un paso. Devuelve el valor de la raíz del error cuadrático medio (RMSE), el error absoluto medio (MAD), el cubrimiento del intervalo predictivo a un paso del 95% (CUB95) y del 99% (CUB99), y el logaritmo de la densidad predictiva extramuestral a un paso (LLF). Las variables `indicesZ` e `indicesX` contienen las componentes de las matrices Z y X utilizadas en la estimación del modelo. Las variables indicadoras `indicesTS` valen 1 si se contiene la componente correspondiente y 0 en otro caso. Se toman como argumentos los mismos que los del programa `pace_general` así como el inicio del periodo de validación (`nval`).

```

function [RMSE,MAD,CUB95,CUB99,LPRED,error] =
validacion_extramuestral(Y,Z,X,indicesZ,indicesX,indicesTS,IT,T,IS,S,mT,
gamma,mS,lambda,n0,nval,nfigura);
% Inicializacion del output
RMSE = 0;
MAD = 0;
CUB95 = 0;
CUB99 = 0;
LPRED = 0;
% Desarrollo de la validacion
% Calculo de las predicciones y de los errores de prediccion
n = length(Y);
ypred_inf95 = zeros(n-nval+1,1);
ypred_sup95 = ypred_inf95;
ypred_inf99 = ypred_inf95;
ypred_sup99 = ypred_inf95;
ypred = ypred_inf95;
error = ypred_inf95;
[ypred,sigma_pred,sigma] =
f_mle_prediccion(Y,Z,X,indicesZ,indicesX,indicesTS,IT,T,IS,S,mT,gamma,
mS,lambda,n0,nval-1,nval);
error = Y(nval:end)-ypred(:,3);
% Calculo de los criterios
RMSE = sqrt(mean(error.*error));
MAD = mean(abs(error));
CUB95 = 100*mean((Y(nval:end)>=ypred(:,2)).*(Y(nval:end)<=ypred(:,4)));
CUB99 = 100*mean((Y(nval:end)>=ypred(:,1)).*(Y(nval:end)<=ypred(:,5)));
ypred_inf95 = ypred(:,2);
ypred_inf99 = ypred(:,1);
ypred_sup95 = ypred(:,4);
ypred_sup99 = ypred(:,5);
ypred = ypred(:,3);
LPRED = sum(log(normpdf(Y(nval:end),ypred,sqrt(sigma_pred)))));
% Presentacion grafica de los resultados obtenidos
plot(1:(n-nval+1),Y(nval:end),1:(n-nval+1),ypred,'r',1:(n-nval+1),
ypred_inf95,'g--',1:(n-nval+1),ypred_inf99,'g:',1:(n-nval+1),ypred_sup95,
'g--',1:(n-nval+1),ypred_sup99,'g:');
legend('y','ypred','lim95','lim99',0);
xlim([1 n-nval+1]);
nfigura = nfigura+1;
figure(nfigura);
plot(1:(n-nval+1),exp(Y(nval:end)),1:(n-nval+1),exp(ypred),'r',1:(n-nval+1),
exp(ypred_inf95),'g--',1:(n-nval+1),exp(ypred_inf99),'g:',1:(n-nval+1),
exp(ypred_sup95),'g--',1:(n-nval+1),exp(ypred_sup99),'g:');
legend('p','ppred','lim95','lim99',0);
xlim([1 n-nval+1]);

```

Programas correspondientes al capítulo 2

Programa 2.11.- Esta función realiza una validación extramuestral del modelo STAR utilizando un proceso de estimación rolling a un paso. Devuelve el valor de la raíz del error cuadrático medio (RMSE), el error absoluto medio (MAD), el cubrimiento del intervalo predictivo a un paso del 95% (CUB95) y del 99% (CUB99), y el logaritmo de la densidad predictiva extramuestral a un paso (LLF). Las variables `indicesZ` e `indicesX` contienen las componentes de las matrices Z y X utilizadas en la estimación del modelo. Las variables indicadoras `indicesTS` valen 1 si se contiene la componente correspondiente y 0 en otro caso. Se toman como argumentos los mismos que los del programa `pace_general` así como el inicio del periodo de validación (`nval`).

```
function [RMSE,MAD,CUB95,CUB99,LPRED,error] =
validacion_rolling(Y,Z,X,tiempo,indicesZ,indicesX,indicesTS,IT,T,IS,S,mT,
gamma,mS,lambda,n0,nval,nfigura);
% Inicializacion del output
RMSE = 0;
MAD = 0;
CUB95 = 0;
CUB99 = 0;
LPRED = 0;
% Desarrollo de la validacion
n = length(Y);
ypred_inf95 = zeros(n-nval+1,1);
ypred_sup95 = ypred_inf95;
ypred_inf99 = ypred_inf95;
ypred_sup99 = ypred_inf95;
ypred = ypred_inf95;
error = ypred_inf95;
nest = nval-1;
for(i=nval:n)
    % Estimacion del modelo
    % Se actualiza el periodo de estimacion cada vez que el tiempo aumenta
    if(tiempo(i)>tiempo(i-1))
        nest = i-1;
    end
    % Calculo de las predicciones y los errores
    j = 1:i;
    [ypred1,sigma_pred,sigma] = f_mle_prediccion(Y(j),Z(j,:),X(j,:),
indicesZ,indicesX,indicesTS,IT(j,:),T(j,:),IS(j,:),S(j,:),mT,gamma,
mS,lambda,n0,nest,i);
    ypred(i-nval+1) = ypred1(3);
    error(i-nval+1) = Y(i)-ypred(i-nval+1);
    % Calculo de los criterios
    RMSE = RMSE + error(i-nval+1)*error(i-nval+1);
    MAD = MAD + abs(error(i-nval+1));
```

```

ypred_inf95(i-nval+1) = ypred1(2);
ypred_sup95(i-nval+1) = ypred1(4);
CUB95 = CUB95 +
(Y(i)>=ypred_inf95(i-nval+1))*(Y(i)<=ypred_sup95(i-nval+1));
ypred_inf99(i-nval+1) = ypred1(1);
ypred_sup99(i-nval+1) = ypred1(5);
CUB99 = CUB99 +
(Y(i)>=ypred_inf99(i-nval+1))*(Y(i)<=ypred_sup99(i-nval+1));
LPRED = LPRED +
log(normpdf(Y(i),ypred(i-nval+1),sqrt(sigma_pred)));
end
CUB95= 100*CUB95/(n-nval+1);
CUB99= 100*CUB99/(n-nval+1);
RMSE = sqrt(RMSE/(n-nval+1));
MAD = MAD/(n-nval+1);
% Presentacion grafica de los resultados
figure(nfigura);
plot(1:(n-nval+1),Y(nval:end),1:(n-nval+1),ypred,'r',1:(n-nval+1),
ypred_inf95,'g--',1:(n-nval+1),ypred_inf99,'g:',1:(n-nval+1),ypred_sup95,
'g--',1:(n-nval+1),ypred_sup99,'g:');
legend('y','ypred','lim95','lim99',0);
xlim([1 n-nval+1]);
nfigura = nfigura+1;
figure(nfigura);
plot(1:(n-nval+1),exp(Y(nval:end)),1:(n-nval+1),exp(ypred),'r',1:(n-nval+1),
exp(ypred_inf95),'g--',1:(n-nval+1),exp(ypred_inf99),'g:',1:(n-nval+1),
exp(ypred_sup95),'g--',1:(n-nval+1),exp(ypred_sup99),'g:');
legend('p','ppred','lim95','lim99',0);
xlim([1 n-nval+1]);

```

Programas correspondientes al capítulo 3

Programa 3.1.- Esta función implementa el Gibbs sampling para un modelo STAR de la forma:

$$Y = Z*\alpha + X*\beta + \text{indfi}(1)*\text{fiTX}*T*(Y-X*\beta) + \text{indfi}(2)*\text{fiSX}*S*(Y-X*\beta) + \text{indfi}(3)*\text{fiSTX}*S*T*(Y-X*\beta) + \text{indfi}(4)*\text{fiTSX}*T*S*(Y-X*\beta) + \text{indfi}(5)*\text{fiT}*T*Y + \text{indfi}(6)*\text{fiS}*S*Y + \text{indfi}(7)*\text{fiST}*S*T*Y + \text{indfi}(8)*\text{fiTS}*T*S*Y + \text{epsilon}$$

con $\text{epsilon} \sim N(0, \sigma^2 I(n))$

y siendo la distribución a priori:

$$\begin{aligned}
\alpha &\sim N(0, \Sigma_{\alpha}) \\
\beta &\sim N(0, \Sigma_{\beta}) \\
\text{fi}_i &\sim U(-1, 1) \text{ para } i = \text{TX, SX, TSX, STX, TY, SY, TSY, STY} \\
\tau &\sim \text{Gamma}(d_0, s_0); \\
m_T &\sim \text{Unif}(m_{Tred}) \\
\gamma &\sim \text{Unif}(\gamma_{mared}) \\
m_S &\sim \text{Unif}(m_{Sred})
\end{aligned}$$

Programas correspondientes al capítulo 3

lambda~Unif(lambdared)

El vector indfi indica que componentes de la parte no parametrica del modelo se quieren incluir en el proceso de estimacion indmuestra0 indica si se incluye una muestra inicial almacenada en la muestra0. El programa devuelve una muestra de la distribucion a posteriori.

```
function muestra = gibbs_pace(Y,Z,X,indicesTS,IT,T,IS,S,
priori,n0,nitermax,indmuestra0,muestra0);
% Creacion de variables auxiliares
n1 = length(Y);
Y1 = Y((n0+1):end);
Z1 = Z((n0+1):end,:);
X1 = X((n0+1):end,:);
% Lectura de las dimensiones del espacio parametrico
n = length(Y1);
p = size(X,2);
q = size(Z,2);
% Lectura de la distribucion a priori
Sigma_alfa = inv(priori.Sigma_alfa);
Sigma_beta = inv(priori.Sigma_beta);
mTred = priori.mTred;
mSred = priori.mSred;
gammared = priori.gammared;
lambdared = priori.lambdared;
d0 = priori.d0;
s0 = priori.s0;
if(sum(indicesTS([1,3,4,5,7,8]))==0)
    mTred = 0;
    gammared = 0;
end
if(sum(indicesTS([2,3,4,6,7,8]))==0)
    mSred = 0;
    lambdared = 0;
end
nTred = length(mTred);
ngamma = length(gammared);
nSred = length(mSred);
nlambda = length(lambdared);
Sigma = zeros(q+8,q+8);
if(q>0)
    Sigma(9:end,9:end) = Sigma_alfa;
end
%
% Creacion de las variables auxiliares
%
TX = priori.TX(1:n1,,:);
SX = priori.SX(1:n1,,:);
```

```

STX = priori.STX(1:n1, :, :, :);
TSX = priori.TSX(1:n1, :, :, :);
TY = priori.TY(1:n1, :, :);
SY = priori.SY(1:n1, :, :);
STY = priori.STY(1:n1, :, :, :);
TSY = priori.TSY(1:n1, :, :, :);
% Inicializacion de las variables
if(p>0)
    betas = zeros(p,nitermax);
end
fis = zeros(8+q,nitermax);
taus = zeros(nitermax,1);
df_tau = n+d0;
mT = taus;
gammas = mT;
mS = mT;
lambdas = mT;
% Lectura de la muestra inicial
if(indmuestra0==1)
    mT(1) = muestra0.mT;
    mS(1) = muestra0.mS;
    lambdas(1) = muestra0.lambdas;
    gammas(1) = muestra0.gammas;
    taus(1) = muestra0.taus;
    fis(1:8,1) = muestra0.fis;
    if(q>0)
        fis(9:end,1) = muestra0.alfas;
    end
    if(p>0)
        betas(:,1) = muestra0.betas;
    end
else
    if(sum(indicesTS>0))
        indmT = round(0.5*nTred);
        indgamma = round(0.5*ngamma);
        indmS = round(0.5*nSred);
        indlambda = round(0.5*nlambda);
        mT(1) = indmT;
        gammas(1) = indgamma;
        mS(1) = indmS;
        lambdas(1) = indlambda;
    end
    taus(1) = 1/s0;
end
% Ejecucion del algoritmo
for(it=2:nitermax)
    if(sum(indicesTS)>0)

```


Programas correspondientes al capítulo 3

```

TX1 = reshape(TX((n0+1):end, :, mT(it-1), gammas(it-1)), n, p);
SX1 = reshape(SX((n0+1):end, :, mS(it-1), lambdas(it-1)), n, p);
TSX1 = reshape(TSX((n0+1):end, :, mT(it-1), gammas(it-1),
mS(it-1), lambdas(it-1)), n, p);
STX1 = reshape(STX((n0+1):end, :, mT(it-1), gammas(it-1),
mS(it-1), lambdas(it-1)), n, p);
TY1 = reshape(TY((n0+1):end, mT(it-1), gammas(it-1)), n, 1);
SY1 = reshape(SY((n0+1):end, mS(it-1), lambdas(it-1)), n, 1);
TSY1 = reshape(TSY((n0+1):end, mT(it-1), gammas(it-1),
mS(it-1), lambdas(it-1)), n, 1);
STY1 = reshape(STY((n0+1):end, mT(it-1), gammas(it-1),
mS(it-1), lambdas(it-1)), n, 1);
end
% Distribucion condicionada de beta
if(p>0)
if(sum(indicesTS)>0)
U = X1-fis(1,it-1)*TX1-fis(2,it-1)*SX1-fis(3,it-1)*TSX1-
fis(4,it-1)*STX1;
else
U = X1;
end
if(q>0)
V = Y1 - Z1*fis(9:end,it-1);
else
V = Y1;
end
if(sum(indicesTS)>0)
V = V-(fis(1,it-1)+fis(5,it-1))*TY1-(fis(2,it-1)+fis(6,it-1))*
SY1-(fis(3,it-1)+fis(7,it-1))*TSY1-(fis(4,it-1)+fis(8,it-1))*STY1;
end
S_beta = Sigma_beta + taus(it-1)*(U'*U);
S_beta = inv(S_beta);
m_beta = S_beta*taus(it-1)*(U'*V);
betas(:,it) = mvnrnd(m_beta,S_beta)';
end
% Distribuciones condicionadas de alfa y fis
if(p>0)
u1 = Y1 - X1*betas(:,it);
else
u1 = Y1;
end
U = zeros(n,q+8);
if(sum(indicesTS)>0)
if(p>0)
U(:,1) = TY1-TX1*betas(:,it);
U(:,2) = SY1-SX1*betas(:,it);
U(:,3) = TSY1-TSX1*betas(:,it);

```

```

    U(:,4) = STY1-STX1*betas(:,it);
end
U(:,5) = TY1;
U(:,6) = SY1;
U(:,7) = TSY1;
U(:,8) = STY1;
if(q>0)
    U(:,9:end) = Z1;
end
if(mTred(mT(it-1))>0)
    if(mSred(mS(it-1))>0)
        indices = 1:8;
        indices = indices(indicesTS==1);
    else
        indices = [1,5];
        indices = indices(indicesTS(indices)==1);
    end
end
else
    if(mSred(mS(it-1))>0)
        indices = [2,6];
        indices = indices(indicesTS(indices)==1);
    else
        indices = [];
    end
end
if(q>0)
    indices = [indices,9:(8+q)];
end
if(length(indices)>0)
    Si = taus(it-1)*(U(:,indices)'*U(:,indices))+Sigma(indices,indices);
    Si = inv(Si);
    mi = taus(it-1)*(U(:,indices)*u1);
    mi = Si*mi;
    ind=1;
    contador = 0;
    while(ind>=1 & contador<=10)
        contador = contador+1;
        fis(indices,it) = mvnrnd(mi,Si)';
        ind=sum(abs(fis(1:8,it))>1);
    end
    if(contador>10)
        fis(indices,it) = fis(indices,it-1);
        for(j=1:length(indices))
            i = indices(j);
            indices1 = indices(indices~=i);
            Vi = u1 - U(:,indices1)*fis(indices1,it);
            Ui = U(:,i);
        end
    end
end

```

Programas correspondientes al capítulo 3

```

    Si = taus(it-1)*(Ui*Ui)+Sigma(i,i);
    mi = (taus(it-1)*(Ui*Vi));
    Si = 1/Si;
    mi = Si*mi;
    Si = sqrt(Si);
    fi_inf = normcdf(-1,mi,Si);
    fi_sup = normcdf(1,mi,Si);
    u = unifrnd(0,1);
    f = u*fi_inf + (1-u)*fi_sup;
    if(0<f & f<1)
        fis(i,it) = norminv(f,mi,Si);
    else
        if(f==0)
            fis(i,it) = -0.9999;
        else
            fis(i,it) = 0.9999;
        end
    end
    end
    Vi = u1-U(:,1:8)*fis(1:8,it);
    if(q>0)
        Ui = U(:,9:end);
        Si = taus(it-1)*(Ui*Ui)+Sigma(9:end,9:end);
        Si = inv(Si);
        mi = (taus(it-1)*(Ui*Vi));
        mi = Si*mi;
        fis(9:end,it) = mvnrnd(mi,Si);
    end
    end
    end
    end
    end
    % Distribucion condicionada de tau
    error = u1-U*fis(:,it);
    a = d0*s0 + sum(error.*error);
    taus(it) = gamrnd(0.5*df_tau,2/a);
    % Distribucion condicionada de (mT,gamma,mS,lambda)
    if(sum(indicesTS)>0)
        prob = [];
        lugar = 0;
        U = zeros(n,8+q);
        if(q>0)
            U(:,9:end) = Z1;
        end
        for(i=1:nTred)
            if(mTred(i)>0)
                ngamma1 = length(gammared);
            else

```

```

ngamma1 = 1;
end
for(i1=1:ngamma1)
for(j=1:nSred)
if(mSred(j)>0)
nlambda1 = length(lambdared);
else
nlambda1 = 1;
end
for(j1 = 1:nlambda1)
U(:,1:8) = 0;
TX1 = reshape(TX((n0+1):end, :, i, i1), n, p);
SX1 = reshape(SX((n0+1):end, :, j, j1), n, p);
TSX1 = reshape(TSX((n0+1):end, :, i, i1, j, j1), n, p);
STX1 = reshape(STX((n0+1):end, :, i, i1, j, j1), n, p);
TY1 = reshape(TY((n0+1):end, i, i1), n, 1);
SY1 = reshape(SY((n0+1):end, j, j1), n, 1);
TSY1 = reshape(TSY((n0+1):end, i, i1, j, j1), n, 1);
STY1 = reshape(STY((n0+1):end, i, i1, j, j1), n, 1);
if(mTred(i)>0)
if(p>0)
U(:,1) = TY1-TX1*betas(:,it);
end
U(:,5) = TY1;
end
if(mSred(j)>0)
if(p>0)
U(:,2) = SY1-SX1*betas(:,it);
end
U(:,6) = SY1;
if(mTred(i)>0)
if(p>0)
U(:,3) = TSY1-TSX1*betas(:,it);
U(:,4) = STY1-STX1*betas(:,it);
end
U(:,7) = TSY1;
U(:,8) = STY1;
end
end
lugar = lugar+1;
numero{lugar} = [i,i1,j,j1];
error = u1-U*fis(:,it);
prob(lugar) = -0.5*taus(it)*sum(error.*error);
% prob1 = prob(lugar)
% pause;
end
end
end

```

Programas correspondientes al capítulo 3

```
    end
end
prob = exp(prob - max(prob));
prob = prob/sum(prob);
numero = numero(prob>0);
prob = prob(prob>0);
if(length(prob)==0)
    it
    tau = taus(it)
    if(q>0)
        alfa = fis(9:end,it)'
    end
    if(p>0)
        beta = betas(:,it)'
    end
    fi = fis(1:8,it)'
    pause;
end
for(i=2:length(prob))
    prob(i) = prob(i)+prob(i-1);
end
u = unifrnd(0,1);
indices1 = 1:length(prob);
indices1 = min(indices1(prob>=u));
numero = numero(indices1);
numero = numero{1};
mT(it) = numero(1);
gammas(it) = numero(2);
mS(it) = numero(3);
lambdas(it) = numero(4);
clear numero;
end
end
% Almacenamiento de la muestra
if(q>0)
    muestra.alfas = fis(9:end,:);
end
if(p>0)
    muestra.betas = betas;
end
muestra.fis = fis(1:8,:);
muestra.taus = taus;
muestra.mT = mT;
muestra.gammas = gammas;
muestra.mS = mS;
muestra.lambdas = lambdas;
```

Programa 3.2.- Esta función realiza el postprocesamiento de la muestra obtenida de la distribución a posteriori del modelo STAR.

```

%Lectura de la muestra y de las variables Z, X y los efectos fis que entran en
el modelo
muestra = muestra_stear;
indicesZ = 1:4;
indicesX = 1:14;
indicesfis = 1:4;
% Indicador de si el modelo es hedonico puro o no
ind_hedonico = 0;
% Indicador de si se quiere calcular los errores estandar de simulacion
ind_corr = 1;
if(ind_corr==1)
    std = length(indicesZ)+length(indicesX)+length(indicesfis)+1;
    std = zeros(std,1);
    limite = 2/sqrt(9000);
    lugar = 0;
end
alfas = muestra.alfas;
betas = muestra.betas;
fis = muestra.fis;
taus = muestra.taus;
mT = muestra.mT;
gammas = muestra.gammas;
mS = muestra.mS;
lambdas = muestra.lambdas;
niter = size(alfas,2);
mTred = priori.mTred;
gammared = priori.gammared;
mSred = priori.mSred;
lambdared = priori.lambdared;
%
% Dibujo de las alfas
%
niter0 = 1001;step = 10;
for(i=1:size(alfas,1))
    if(ind_corr==1)
        corr = autocorr(alfas(i,niter0:end),100);
        lugar = lugar+1;
        std(lugar) = var(alfas(i,niter0:end))*(1+sum(corr(abs(corr)<limite)));
        std(lugar) = sqrt(std(lugar)/9000);
    else
        nfigura = 1;
        figure(nfigura);
        subplot(size(alfas,1),2,2*(i-1)+1);
    end
end

```

Programas correspondientes al capítulo 3

```
    plot(1:niter,alfas(i,:));
    ylabel(literales_coeficientes(indicesZ(i,:)));
    subplot(size(alfas,1),2,2*i);
    autocorr(alfas(i,niter0:end)',100);
end
end
% Dibujo de betas
for(i=1:size(betas,1))
    if(ind_corr==1)
        corr = autocorr(betas(i,niter0:end)',100);
        lugar = lugar+1;
        std(lugar) = var(betas(i,niter0:end))*(1+sum(corr(abs(corr)<limite)));
        std(lugar) = sqrt(std(lugar)/9000);
    else
        if(mod(i-1,4)==0)
            nfigura = nfigura+1;
            figure(nfigura);
        end
        subplot(4,2,2*mod(i-1,4)+1);
        plot(1:niter,betas(i,:));
        ylabel(literales_coeficientes(4+indicesX(i,:)));
        subplot(4,2,2*mod(i-1,4)+2);
        autocorr(betas(i,niter0:end)');
    end
end
end
% Dibujo de taus
if(ind_corr==1)
    corr = autocorr(taus(niter0:end),100);
    lugar = lugar+1;
    std(lugar) = var(taus(niter0:end))*(1+sum(corr(abs(corr)<limite)));
    std(lugar) = sqrt(std(lugar)/9000);
else
    subplot(4,2,5);
    plot(1:niter,taus);
    ylabel(literales_coeficientes(19,:));
    subplot(4,2,6);
    autocorr(taus(niter0:end));
end
end
% Dibujo de fis, mT, mS, lambda y gamma
if(length(indicesfis)>0)
    if(ind_corr==1)
        for(i=1:length(indicesfis))
            corr = autocorr(fis(i,niter0:end)',100);
            lugar = lugar+1;
            std(lugar) = var(fis(i,niter0:end))*(1+sum(corr(abs(corr)<limite)));
            std(lugar) = sqrt(std(lugar)/9000);
        end
    end
end
```

```

else
    nfigura = nfigura+1;
    figure(nfigura);
    for(i=1:length(indicesfis))
        subplot(4,2,2*(i-1)+1);
        plot(1:niter,fis(i,:));
        ylabel(literales_coeficientes(19+indicesfis(i,:)));
        subplot(4,2,2*i);
        autocorr(fis(i,niter0:end)',100);
    end
    nfigura = nfigura+1;
    figure(nfigura);
    subplot(4,2,1);
    plot(1:niter,mTred(mT));
    ylabel(literales_coeficientes(24,:));
    subplot(4,2,2);
    autocorr(mTred(mT(niter0:end)));
    subplot(4,2,3);
    plot(1:niter,gammared(gammas));
    ylabel(literales_coeficientes(25,:));
    subplot(4,2,4);
    autocorr(gammared(gammas(niter0:end)));
    subplot(4,2,5);
    plot(1:niter,mSred(mS));
    ylabel(literales_coeficientes(26,:));
    subplot(4,2,6);
    autocorr(mSred(mS(niter0:end)));
    subplot(4,2,7);
    plot(1:niter,lambdared(lambdas));
    ylabel(literales_coeficientes(27,:));
    subplot(4,2,8);
    autocorr(lambdared(lambdas(niter0:end)));
end
end
% Dibujo de las densidades
% Densidades de alfa
nfigura = nfigura+1;
figure(nfigura);
for(i=1:size(alfas,1))
    subplot(2,2,i);
    [f,xi] = ksdensity(alfas(i,niter0:step:end)');
    plot(xi,f);
    ylabel(literales_coeficientes(indicesZ(i,:)));
end
% Dibujo de betas
for(i=1:size(betas,1))
    if(rem(i-1,4)==0)

```


Programas correspondientes al capítulo 3

```
nfigura = nfigura+1;
figure(nfigura);
end
subplot(2,2,rem(i-1,4)+1);
[f,xi] = ksdensity((betas(i,niter0:step:end)));
plot(xi,f);
ylabel(literales_coeficientes(4+indicesX(i,:)));
end
% Dibujo de taus
subplot(2,2,3);
[f,xi] = ksdensity(taus(niter0:end));
plot(xi,f);
ylabel(literales_coeficientes(19,:));
% Dibujo de fis, mT, mS, lambda y gamma
if(length(indicesfis)>0)
nfigura = nfigura+1;
figure(nfigura);
for(i=1:length(indicesfis))
subplot(2,2,i);
[f,xi] = ksdensity((fis(indicesfis(i),niter0:step:end)));
plot(xi,f);
ylabel(literales_coeficientes(19+indicesfis(i,:)));
end
nfigura = nfigura+1;
figure(nfigura);
subplot(2,2,1);
a = tabulate(mT(niter0:step:end));
bar(mTred,a(:,3));
xlim([mTred(1) mTred(end)]);
title(literales_coeficientes(24,:));
subplot(2,2,2);
a = tabulate(gammas(niter0:step:end));
bar(gammared,a(:,3));
xlim([0 1]);
title(literales_coeficientes(25,:));
subplot(2,2,3);
a = tabulate(mS(niter0:step:end));
bar(mSred,a(:,3));
xlim([mSred(1) mSred(end)]);
title(literales_coeficientes(26,:));
subplot(2,2,4);
a = tabulate(lambdas(niter0:step:end));
bar(lambdared,a(:,3));
xlim([0 1]);
title(literales_coeficientes(27,:));
end
% Calculo de estimadores puntuales
```

```

if(ind_hedonico==0)
    strcat(strvcat(literal_z(indicesZ,:),literal_x(indicesX,:),literal_fis(1:4,:),
        'tau','mT','gamma','mS','lambda'),num2str([prctile(alfas(:,niter0:step:end)',
        [2.5 50 97.5]);prctile(betas(:,niter0:step:end)',[2.5 50 97.5]);
        prctile(fis(1:4,niter0:step:end)',[2.5 50 97.5]);
        prctile(taus(niter0:step:end),[2.5 50 97.5]);
        prctile(mTred(mT(niter0:step:end)),[2.5 50 97.5]);
        prctile(gammared(gammas(niter0:step:end)),[2.5 50 97.5]);
        prctile(mSred(mS(niter0:step:end)),[2.5 50 97.5]);
        prctile(lambdared(lambdas(niter0:step:end)),[2.5 50 97.5])))
else
    strcat(strvcat(literal_z,literal_x,'tau'),num2str([prctile
        (betas(:,niter0:step:end)',[2.5 50 97.5]);prctile(taus(niter0:step:end),
        [2.5 50 97.5]);
end

```

Programa 3.3.- Esta función calcula las predicciones extramuestrales del modelo STAR analizado desde una optica bayesiana utilizando la muestra de la distribucion a posteriori calculada con el algoritmo gibbs_pace, nval indica el inicio del periodo de validación. Utiliza el muestreo de composición, niter0 contiene el inicio del periodo de toma de la muestra, step contiene el numero de pasos para reducir el valor de la autocorrelacion de la muestra a valores poco significativos. Devuelve una muestra de la distribucion predictiva, diferentes criterios de evaluacion de la misma asi como los errores de prediccion no tipificados. Realiza, ademas, un estudio grafico de las predicciones.

```

function [ypred,CUB95,CUB99,AJUSTE,PD,DIC,LPRED,R2,errores1] =
pred_bay(Y,tiempo,Z,X,Zespaciales,IT,T,IS,S,priori,nval,muestra,niter0,step);
% Lectura de la distribucion a priori
mTred = priori.mTred;
mSred = priori.mSred;
gammared = priori.gammared;
lambdared = priori.lambdared;
% Lectura de la muestra de la distribucion a posteriori
q = size(Z,2);
p = size(X,2);
if(q>0)
    alfas = muestra.alfas;
end
if(p>0)
    betas = muestra.betas;
end
fis = muestra.fis;
taus = muestra.taus;
mT = muestra.mT;
gammas = muestra.gammas;
mS = muestra.mS;

```

Programas correspondientes al capítulo 3

```
lambdas = muestra.lambdas;
if(p>0)
    betas = betas(:,niter0:step:end);
end
if(q>0)
    alfas = alfas(:,niter0:step:end);
end
fis = fis(:,niter0:step:end);
taus = taus(niter0:step:end);
mT = mT(niter0:step:end);
mT(mT>0) = mTred(mT(mT>0));
gammas = gammas(niter0:step:end);
gammas(gammas>0) = gammared(gammas(gammas>0));
mS = mS(niter0:step:end);
mS(mS>0) = mSred(mS(mS>0));
lambdas = lambdas(niter0:step:end);
lambdas(lambdas>0) = lambdared(lambdas(lambdas>0));
% Inicializacion de las variables
niter = size(betas,2);
Y1 = Y(nval:end);
n = length(Y1);
ypred = zeros(n,niter);
medias=ypred;
LLF=0;
LPRED = 0;
% Calculo de las predicciones y los criterios utilizando la informacion
% proporcionada por la muestra y aplicando el muestreo de composicion
for(it=1:niter)
    media = zeros(n,1);
    if(q>0)
        media = media + Z(nval:end,:)*alfas(:,it);
    end
    if(p>0)
        media = media + X(nval:end,:)*betas(:,it);
    end
    [TX,SX,TSX,STX,TY,SY,TSY,STY] = calculo_TS_c(Y,X,IT,T,IS,S,mT(it),gammas(it),mS(it),lambdas(it));
    if(mT(it)>0)
        if(p>0)
            U = TY(nval:end)-TX(nval:end,:)*betas(:,it);
        else
            U = zeros(n,1);
        end
        if(mS(it)>0)
            if(p>0)
                U = [U,SY(nval:end)-SX(nval:end,:)*betas(:,it)];
                U = [U,TSY(nval:end)-TSX(nval:end,:)*betas(:,it)];
                U = [U,STY(nval:end)-STX(nval:end,:)*betas(:,it)];
            end
        end
    end
end
```

```

else
    U = [U,zeros(n,3)];
end
U = [U,TY(nval:end)];
U = [U,SY(nval:end)];
U = [U,TSY(nval:end)];
U = [U,STY(nval:end)];
media = media + U*fis(:,it);
else
    U = [U,TY(nval:end)];
    media = media + U*fis([1,5],it);
end
else
    if(mS(it)>0)
        if(p>0)
            U = SY(nval:end)-SX(nval:end,)*betas(:,it);
        else
            U = zeros(n,1);
        end
        U = [U,SY(nval:end)];
        media = media + U*fis([2,6],it);
    end
end
ypred(:,it) = normrnd(media,sqrt(1/taus(it)));
medias(:,it) = media;
LLF = LLF + sum(log(normpdf(Y1,media,sqrt(1/taus(it)))));
LPRED = LPRED + prod(normpdf(Y1,media,sqrt(1/taus(it))));
end
%
% Calculo del criterio DIC y del logaritmo de la densidad predictiva
%
AJUSTE = -2*LLF/niter;
LPRED = log(LPRED/niter);
% Calculo de la media a posteriori
if(q>0)
    alfas_media = mean(alfas,2);
end
if(p>0)
    betas_media = mean(betas,2);
end
fis_media = mean(fis,2);
taus_media = mean(taus);
mT_media = round(mean(mT));
gamma_media = mean(gammas);
mS_media = round(mean(mS));
lambda_media = round(mean(mS));
% Calculo de la desviianza

```

Programas correspondientes al capítulo 3

```
media = zeros(n,1);
if(q>0)
    media = media + Z(nval:end,:)*alfas_media;
end
if(p>0)
    media = media + X(nval:end,:)*betas_media;
end
[TX,SX,TSX,STX,TY,SY,TSY,STY] =
calculo_TS_c(Y,X,IT,T,IS,S,mT_media,gamma_media,mS_media,
lambda_media);
if(mT_media>0)
    if(p>0)
        U = TY(nval:end)-TX(nval:end,:)*betas_media;
    else
        U = zeros(n,1);
    end
    if(mS_media>0)
        if(p>0)
            U = [U,SY(nval:end)-SX(nval:end,:)*betas_media];
            U = [U,TSY(nval:end)-TSX(nval:end,:)*betas_media];
            U = [U,STY(nval:end)-STX(nval:end,:)*betas_media];
        else
            U = [U,zeros(n,3)];
        end
        U = [U,TY(nval:end)];
        U = [U,SY(nval:end)];
        U = [U,TSY(nval:end)];
        U = [U,STY(nval:end)];
        media = media + U*fis_media;
    else
        U = [U,TY(nval:end)];
        media = media + U*fis_media([1,5]);
    end
end
else
    if(mS_media>0)
        if(p>0)
            U = SY(nval:end)-SX(nval:end,:)*betas_media;
        else
            U = zeros(n,1);
        end
        U = [U,SY(nval:end)];
        media = media + U*fis_media([2,6]);
    end
end
PD = AJUSTE + 2*sum(log(normpdf(Y1,media,sqrt(1/taus_media))));
DIC = PD + AJUSTE;
% Estudio grafico y numerico de errores de prediccion
```

```

errores = Y1*ones(1,niter)-ypred;
for(i=1:niter)
    errores(:,i) = sqrt(taus(i))*errores(:,i);
end
errores1 = errores;
error = prctile(errores',[0.5 2.5 50 97.5 99.5]);
CUB95 = 100*mean((error(:,2)<=0).*(error(:,4)>=0));
CUB99 = 100*mean((error(:,1)<=0).*(error(:,5)>=0));
nfigura = 1;
figure(nfigura);
plot(1:n,error(:,3),'r',1:n,error(:,1),'b',1:n,error(:,2),'g',1:n,error(:,4),'g',1:n,
error(:,5),'b',1:n,zeros(n,1),'k');
legend('error','errorinf99','errorinf95','errorsup95','errorsup99',0);
xlabel('Observacion');
ylabel('Error');
xlim([0 length(Y1)]);
% Estudios de las autocorrelaciones residuales con bandas de confianza del 95%
nfigura = nfigura+1;
figure(nfigura);
nmax = size(S,2);
correlaciones = zeros(nmax+1,100);
[IS,S] =
matriz_proximidad_espacial(tiempo(nval:end),Zespaciales(nval:end,:),nmax);
for(i=(niter-99):niter)
    correlaciones(:,i-niter+100) =
autocorrelaciones_espaciales(errores(:,i),IS,S,0);
end
correlaciones = prctile(correlaciones',[2.5 50 97.5]);
plot(0:nmax,correlaciones(:,2),0:nmax,zeros(nmax+1,1),'b--
',0:nmax,correlaciones(:,1),'b',0:nmax,correlaciones(:,3),'b');
xlabel('Numero de vecinos');
ylabel('Autocorrelaciones empiricas');
% Estudio de normalidad de los errores tomando como estimador de los parametros la media a posteriori
zr = sqrt(taus_media)*(Y1-media);
estudio_zr = [min(zr),prctile(zr,[1,5,10,25,50,75,90,95,99]),max(zr),mean(zr),
median(abs(zr)),skewness(zr),kurtosis(zr)]'
nfigura = nfigura+1;
figure(nfigura);
plot(1:length(zr),zr,1:length(zr),-2*ones(length(zr),1),'r',1:length(zr),2*ones(length(zr),1),'r');
nfigura = nfigura+1;
figure(nfigura);
[f,xi] = ksdensity(zr);
plot(xi,f);
nfigura = nfigura+1;
figure(nfigura);
qqplot(zr);
% Predicciones puntuales e intervalos de credibilidad del logaritmo de los precios
nfigura = nfigura+1;

```

Programas correspondientes al capítulo 3

```
figure(nfigura);
ypred = prctile(ypred',[0.5 2.5 50 97.5 99.5]);
plot(1:n,Y1,'k',1:n,ypred(:,3),'r',1:n,ypred(:,1),'b',1:n,ypred(:,2),'g',1:n,ypred(:,4),'g',1:n,ypred(:,5),'b');
legend('Y','Ypred','Yinf99','Yinf95','Ysup95','Ysup99',0);
xlabel('Observacion');
ylabel('Log-precio');
xlim([0 length(Y1)]);
% Predicciones puntuales e intervalos de credibilidad de los precios
nfigura = nfigura+1;
figure(nfigura);
plot(1:n,exp(Y1),'k',1:n,exp(ypred(:,3)), 'r',1:n,exp(ypred(:,1)), 'b',1:n,exp(ypred(:,2)), 'g',1:n,exp(ypred(:,4)), 'g',1:n,exp(ypred(:,5)), 'b');
legend('P','Ppred','Pinf99','Pinf95','Psup95','Psup99',0);
xlabel('Observacion');
ylabel('Precio');
xlim([0 length(Y1)]);
% Calculo R2
media = mean(medias,2);
MSE = mean((Y1-media).^2);
R2 = 1-MSE/var(Y1);
```

Programa 3.4.- Esta función calcula las predicciones extramuestrales del modelo STAR analizado desde una optica bayesiana utilizando la muestra de la distribución a posteriori calculada con el algoritmo gibbs_pace, nval indica el inicio del periodo de validación. Utiliza el muestreo de composición asignando los pesos w a los elementos de la muestra de la distribución a posteriori calculada en el periodo de estimación. Devuelve una muestra de la distribución predictiva, su media y su varianza, el logaritmo de la densidad predictiva y los errores de predicción tipificados. Realiza, además, un estudio gráfico de las predicciones.

```
function [ypred,medias,varianzas,LLF,errores] =
pred_bay_rolling(Y,Z,X,TX,SX,STX,TSX,TY,SY,STY,TSY,muestra,w1);
% Lectura de la muestra de la distribución a posteriori
q = size(Z,2);
p = size(X,2);
if(q>0)
    alfas = muestra.alfas;
end
if(p>0)
    betas = muestra.betas;
end
fis = muestra.fis;
taus = muestra.taus;
mT = muestra.mT;
mS = muestra.mS;
niter = size(fis,2);
% Extracción de la muestra de acuerdo a los pesos de importancia w1
```

```

if(var(w1)>0)
    u = unifrnd(0,1,niter,1);
    indices_simulados = ones(niter,1);
    for(i=2:niter)
        w1(i) = w1(i-1)+w1(i);
        indices_simulados = indices_simulados + uplus(u>=w1(i));
    end
    if(q>0)
        alfas = alfas(:,indices_simulados);
    end
    if(p>0)
        betas = betas(:,indices_simulados);
        TX = TX(:,indices_simulados);
        SX = SX(:,indices_simulados);
        STX = STX(:,indices_simulados);
        TSX = TSX(:,indices_simulados);
    end
    fis = fis(:,indices_simulados);
    taus = taus(indices_simulados);
    mT = mT(indices_simulados);
    mS = mS(indices_simulados);
    TY = TY(indices_simulados);
    SY = SY(indices_simulados);
    STY = STY(indices_simulados);
    TSY = TSY(indices_simulados);
end
% Inicializacion de las variables
ypred = zeros(1,niter);
errores = ypred;
medias = 0;
medias2 = medias;
LLF=0;
% Calculo de las predicciones y los criterios utilizando la informacion
% proporcionada por la muestra y aplicando el muestreo de composicion
for(it=1:niter)
    media = 0;
    if(q>0)
        media = media + Z*alfas(:,it);
    end
    if(p>0)
        media = media + X*betas(:,it);
    end
    if(mT(it)>0)
        if(p>0)
            U = TY(it)-TX(:,it)*betas(:,it);
        else
            U = 0;
        end
    end
end

```


Programas correspondientes al capítulo 3

```
end
if(mS(it)>0)
  if(p>0)
    U = [U,SY(it)-SX(:,it)*betas(:,it)];
    U = [U,TSY(it)-TSX(:,it)*betas(:,it)];
    U = [U,STY(it)-STX(:,it)*betas(:,it)];
  else
    U = [U,zeros(1,3)];
  end
  U = [U,TY(it)];
  U = [U,SY(it)];
  U = [U,TSY(it)];
  U = [U,STY(it)];
  media = media + U*fis(:,it);
else
  U = [U,TY(it)];
  media = media + U*fis([1,5],it);
end
else
  if(mS(it)>0)
    if(p>0)
      U = SY(it)-SX(:,it)*betas(:,it);
    else
      U = 0;
    end
    U = [U,SY(it)];
    media = media + U*fis([2,6],it);
  end
end
medias = medias + media;
medias2 = medias2 + media.*media;
ypred(1,it) = normrnd(media,sqrt(1/taus(it)));
errores(1,it) = sqrt(taus(it))*(Y - ypred(1,it));
LLF = LLF + prod(normpdf(Y,media,sqrt(1/taus(it))));
end
medias = medias/niter;
varianzas = medias2/niter-medias.*medias;
LLF = log(LLF/niter);
```

Programa 3.5.- Esta función devuelve los parámetros de la distribución a priori utilizada en el programa `gibbs_pace`. Así mismo construye las matrices auxiliares `TX`, `SX`, `STX`, `TSX`, `TY`, `SY`, `STY`, `TSY` utilizadas en dicho programa para las redes de los parámetros de vecindad especificadas en la distribución a priori.

```
function priori = prior(Y,Z,X,IT,T,IS,S);
```

```

n = length(Y);
p = size(X,2);
q = size(Z,2);
priori.Sigma_alfa = 10000*eye(q);
priori.Sigma_beta = 10000*eye(p);
priori.d0 = 0.01;
priori.s0 = 1;
priori.mTred = 0:5:35;
priori.mSred = 0:5:35;
priori.gammared = 0.2:0.2:1;
priori.lambdared = 0.2:0.2:1;
nTred = length(priori.mTred);
nSred = length(priori.mSred);
ngamma = length(priori.gammared);
nlambda = length(priori.lambdared);
TX = zeros(n,p,nTred,ngamma);
SX = zeros(n,p,nSred,nlambd);
STX = zeros(n,p,nTred,ngamma,nSred,nlambd);
TSX = STX;
TY = zeros(n,nTred,ngamma);
SY = zeros(n,nSred,nlambd);
STY = zeros(n,nTred,ngamma,nSred,nlambd);
TSY = STY;
for(i=1:nTred)
    for(i1=1:ngamma)
        for(j=1:nSred)
            for(j1=1:nlambd)
                [TX(:,i,i1),SX(:,j,j1),TSX(:,i,i1,j,j1),STX(:,i,i1,j,j1),
                TY(:,i,i1),SY(:,j,j1),TSY(:,i,i1,j,j1),STY(:,i,i1,j,j1)] =
                calculo_TS_c(Y,X,IT,T,IS,S,priori.mTred(i),
                priori.gammared(i1),priori.mSred(j),priori.lambdared(j1));
            end
        end
    end
end
priori.TX = TX;
priori.SX = SX;
priori.TSX = TSX;
priori.STX = STX;
priori.TY = TY;
priori.SY = SY;
priori.TSY = TSY;
priori.STY = STY;

```

Programa 3.6.- Esta función realiza una validacion extramuestral del modelo STAR tipo rolling analizado desde una optica bayesiana utilizando la muestra de la distribucion a

Programas correspondientes al capítulo 3

posteriori calculada con el algoritmo gibas_pace, nval indica el inicio del periodo de validación. Utiliza el muestreo de importancia secuencial descrito en el Capítulo 3. La muestra inicial se supone obtenida por medio del Gibbs sampling aplicado en el periodo de estimación 1, ..., nval-1, niter0 contiene el inicio del periodo de toma de la muestra, step contiene el número de pasos para reducir el valor de la autocorrelación de la muestra a valores poco significativos.

Devuelve los valores de diversos criterios de evaluación de la actividad predictiva entre los que destaca el vector LPRED que contiene los logaritmos de la densidad predictiva en cada periodo. Realiza, además, un estudio gráfico de los resultados obtenidos.

```
function[MSE,MAD,CUB95,CUB99,SESGO,VARIANZA,LOSS,
LPRED,errores]=validacion_bay_rolling(Y,tiempo,Z,X,Zespaciales,indicesTS
,IT,T,IS,S,priori,n0,nval,muestra,niter0,step);
% Lectura de la distribucion a priori
Sigma_alfa = inv(priori.Sigma_alfa);
Sigma_beta = inv(priori.Sigma_beta);
mTred = priori.mTred;
mSred = priori.mSred;
gammared = priori.gammared;
lambdared = priori.lambdared;
n = length(Y);
TX1 = priori.TX(1:n, :, :);
SX1 = priori.SX(1:n, :, :);
STX1 = priori.STX(1:n, :, :, :);
TSX1 = priori.TSX(1:n, :, :, :, :);
TY1 = priori.TY(1:n, :, :);
SY1 = priori.SY(1:n, :, :);
STY1 = priori.STY(1:n, :, :, :);
TSY1 = priori.TSY(1:n, :, :, :, :);
% Lectura de la muestra de la distribucion a posteriori
q = size(Z,2);
p = size(X,2);
if(q>0)
    alfas = muestra.alfas(:,niter0:step:end);
    muestra.alfas = alfas;
end
if(p>0)
    betas = muestra.betas(:,niter0:step:end);
    muestra.betas = betas;
end
fis = muestra.fis(:,niter0:step:end);
taus = muestra.taus(niter0:step:end);
mT = muestra.mT(niter0:step:end);
gammas = muestra.gammas(niter0:step:end);
mS = muestra.mS(niter0:step:end);
lambdas = muestra.lambdas(niter0:step:end);
```

```

muestra.fis = fis;
muestra.taus = taus;
muestra.mT = mT;
muestra.gammas = gammas;
muestra.mS = mS;
muestra.lambdas = lambdas;
% Inicializacion de las variables
niter = size(fis,2);
Y1 = Y(nval:end);
n1 = length(Y1);
ypred = zeros(n1,niter);
errores = ypred;
medias = zeros(n1,1);
LPRED = medias;
VARIANZA = 0;
w = zeros(1,niter);
w1 = ones(1,niter)/niter;
TY = zeros(1,niter);
SY = TY;
STY = TY;
TSY = TY;
if(p>0)
    TX = zeros(p,niter);
    SX = TX;
    STX = TX;
    TSX = TX;
else
    TX = zeros(1,niter);
    SX = TX;
    STX = TX;
    TSX = TX;
end
% Calculo de las predicciones y los criterios utilizando la informacion
% proporcionada por la muestra y aplicando el muestreo de composicion
% Utilizamos el muestreo de importancia para actualizar las distribuciones
% a posteriori en cada paso siempre y cuando su calidad (medida por el
% indicador ESS) sea alta
% En caso contrario se reinicia el proceso de muestreo
% Por si acaso hay empate en el tiempo guardamos en waux, las ultimas
% actualizaciones de los pesos de importancia;
waux1 = zeros(1,niter);
for(it=1:niter)
    media = 0;
    if(q>0)
        media = media + Z(nval,:)*alfas(:,it);
    end
    media = media + X(nval,:)*betas(:,it);

```

Programas correspondientes al capítulo 3

```

if(p>0)
  if(mT(it)>0)
    TX(:,it) = reshape(TX1(nval, :, mT(it), gammas(it)), p, 1);
    if(mS(it)>0)
      TSX(:,it) =
        reshape(TSX1(nval, :, mT(it), gammas(it), mS(it), lambdas(it)), p, 1);
      STX(:,it) =
        reshape(STX1(nval, :, mT(it), gammas(it), mS(it), lambdas(it)), p, 1);
    end
  end
  if(mS(it)>0)
    SX(:,it) = reshape(SX1(nval, :, mS(it), lambdas(it)), p, 1);
  end
end
if(mT(it)>0)
  TY(it) = TY1(nval, mT(it), gammas(it));
  if(mS(it)>0)
    TSY(it) = TSY1(nval, mT(it), gammas(it), mS(it), lambdas(it));
    STY(it) = STY1(nval, mT(it), gammas(it), mS(it), lambdas(it));
  end
end
if(mS(it)>0)
  SY(it) = SY1(nval, mS(it), lambdas(it));
end
if(mT(it)>0)
  if(p>0)
    U = TY(it) - TX(:,it)*betas(:,it);
  else
    U = 0;
  end
  if(mS(it)>0)
    if(p>0)
      U = [U, SY(it) - SX(:,it)*betas(:,it)];
      U = [U, TSY(it) - TSX(:,it)*betas(:,it)];
      U = [U, STY(it) - STX(:,it)*betas(:,it)];
    else
      U = [U, zeros(1,3)];
    end
    U = [U, TY(it)];
    U = [U, SY(it)];
    U = [U, TSY(it)];
    U = [U, STY(it)];
    media = media + U*fis(:,it);
  else
    U = [U, TY(it)];
    media = media + U*fis([1,5],it);
  end
end

```

```

else
    if(mS(it)>0)
        if(p>0)
            U = SY(it)-SX(:,it)*betas(:,it);
        else
            U = 0;
        end
        U = [U,SY(it)];
        media = media + U*fis([2,6],it);
    end
end
waux1(it) = log(normpdf(Y(nval),media,sqrt(1/taus(it))));
end
% Inicializacion de los valores de los criterios
if(p*q>0)
    [ypred(1,:),medias(1),varianzas,LLF1,errores(1,:)] =
    pred_bay_rolling(Y(nval),Z(nval,:),X(nval,:),TX,SX,STX,TSX,
    TY,SY,STY,TSY,muestra,w1);
else
    if(p>0)
        [ypred(1,:),medias(1),varianzas,LLF1,errores(1,:)] =
        pred_bay_rolling(Y(nval),[],X(nval,:),TX,SX,STX,TSX,
        TY,SY,STY,TSY,muestra,w1);
    else
        [ypred(1,:),medias(1),varianzas,LLF1,errores(1,:)] =
        pred_bay_rolling(Y(nval),Z(nval,:),[],TX,SX,STX,TSX,
        TY,SY,STY,TSY,muestra,w1);
    end
end
end
VARIANZA = varianzas;
LPRED(1) = LLF1;
waux = waux1;
periodo = 1:nval;
% Proceso de validacion rolling
for(i=(nval+1):n)
    % Actualizacion de los pesos de importancia
    w = w + waux1;
    if(tiempo(i)>tiempo(i-1))
        w = w - waux1;
        w = w + waux;
        waux = zeros(1,niter);
        w1 = exp(w-max(w));
        w1 = w1/sum(w1);
        % Analisis de la calidad de los pesos
        ESS = niter/(1+var(w1))
        if(ESS<0.5*niter)
            % Si la calidad es escasa, rehacer el muestreo;

```

Programas correspondientes al capítulo 3

```
if(q>0)
    muestra.alfas = muestra.alfas(:,end);
end
if(p>0)
    muestra.betas = muestra.betas(:,end);
end
muestra.fis = muestra.fis(:,end);
muestra.taus = muestra.taus(end);
muestra.mT = muestra.mT(end);
muestra.gammas = muestra.gammas(end);
muestra.mS = muestra.mS(end);
muestra.lambdas = muestra.lambdas(end);
if(p*q>0)
    muestra = gibbs_pace1(Y(periodo),Z(periodo,:),X(periodo:),
    indicesTS,IT(periodo:),T(periodo:),IS(periodo:),S(periodo:),
    priori,n0,niter,101,1,1,muestra);
else
    if(p>0)
        muestra = gibbs_pace1(Y(periodo),[],X(periodo:),
        indicesTS,IT(periodo:),T(periodo:),IS(periodo:),
        S(periodo:),priori,n0,niter,101,1,1,muestra);
    else
        muestra = gibbs_pace1(Y(periodo),Z(periodo:),[],
        indicesTS,IT(periodo:),T(periodo:),IS(periodo:),S(periodo:),
        priori,n0,niter,101,1,1,muestra);
    end
end
w1 = ones(1,niter)/niter;
if(q>0)
    alfas = muestra.alfas;
end
if(p>0)
    betas = muestra.betas;
end
fis = muestra.fis;
taus = muestra.taus;
mT = muestra.mT;
gammas = muestra.gammas;
mS = muestra.mS;
lambdas = muestra.lambdas;
periodo = 1:i;
else
    waux = waux + waux1;
end
% Calculo de matrices auxiliares para la prediccion y la actualizacion
% de los pesos
for(it=1:niter)
```

```

media = 0;
if(q>0)
    media = media + Z(i,:)*alfas(:,it);
end
if(p>0)
    media = media + X(i,:)*betas(:,it);
    if(mT(it)>0)
        TX(:,it) = reshape(TX1(i,:,mT(it),gammas(it)),p,1);
        if(mS(it)>0)
            TSX(:,it) = reshape(TSX1(i,:,mT(it),
                gammas(it),mS(it),lambdas(it)),p,1);
            STX(:,it) = reshape(STX1(i,:,mT(it),
                gammas(it),mS(it),lambdas(it)),p,1);
        end
    end
    if(mS(it)>0)
        SX(:,it) = reshape(SX1(i,:,mS(it),lambdas(it)),p,1);
    end
end
if(mT(it)>0)
    TY(it) = TY1(i,mT(it),gammas(it));
    if(mS(it)>0)
        TSY(it) = TSY1(i,mT(it),gammas(it),mS(it),lambdas(it));
        STY(it) = STY1(i,mT(it),gammas(it),mS(it),lambdas(it));
    end
end
if(mS(it)>0)
    SY(it) = SY1(i,mS(it),lambdas(it));
end
if(mT(it)>0)
    if(p>0)
        U = TY(it)-TX(:,it)*betas(:,it);
    else
        U = 0;
    end
    if(mS(it)>0)
        if(p>0)
            U = [U,SY(it)-SX(:,it)*betas(:,it)];
            U = [U,TSY(it)-TSX(:,it)*betas(:,it)];
            U = [U,STY(it)-STX(:,it)*betas(:,it)];
        else
            U = [U,zeros(1,3)];
        end
    end
    U = [U,TY(it)];
    U = [U,SY(it)];
    U = [U,TSY(it)];
    U = [U,STY(it)];
end

```


Programas correspondientes al capítulo 3

```

        media = media + U*fis(:,it);
    else
        U = [U,TY(it)];
        media = media + U*fis([1,5],it);
    end
else
    if(mS(it)>0)
        if(p>0)
            U = SY(it)-SX(:,it)*betas(:,it);
        else
            U = 0;
        end
        U = [U,SY(it)];
        media = media + U*fis([2,6],it);
    end
end
waux1(it) = log(normpdf(Y(i),media,sqrt(1/taus(it))));
end
end
% Calculo de las predicciones y del logaritmo de la densidad predictiva
if(p*q>0)
    [ypred(i-nval+1,:),medias(i-nval+1),varianzas,LLF1,errores(i-nval+1,:)]
    =pred_bay_rolling(Y(i),Z(i,:),X(i,:),TX,SX,STX,TSX,
    TY,SY,STY,TSY,muestra,w1);
else
    if(p>0)
        [ypred(i-nval+1,:),medias(i-nval+1),varianzas,LLF1,
        errores(i-nval+1,:)] =
        pred_bay_rolling(Y(i),[],X(i,:),TX,SX,STX,TSX,
        TY,SY,STY,TSY,muestra,w1);
    else
        [ypred(i-nval+1,:),medias(i-nval+1),varianzas,LLF1,
        errores(i-nval+1,:)] =
        pred_bay_rolling(Y(i),Z(i,:),[],TX,SX,STX,TSX,
        TY,SY,STY,TSY,muestra,w1);
    end
end
end
VARIANZA = VARIANZA+varianzas;
LPRED(i-nval+1) = LPRED(i-nval)+LLF1;
end
% Errores de prediccion
n = length(Y1);
error = prctile(errores',[0.5 2.5 50 97.5 99.5]);
nfigura = 1;
figure(nfigura);
plot(1:n,error(:,3),'r',1:n,error(:,1),'b',1:n,error(:,2),'g',1:n,error(:,4),'g',1:n,error(:,5),'b',1:n,zeros(n,1),'k');
legend('error','errorinf99','errorinf95','errorsup95','errorsup99',0);

```

```

xlabel('Observacion');
ylabel('Error');
xlim([0 length(Y1)]);
% Estudios de las autocorrelaciones residuales con bandas de confianza del 95%
nfigura = nfigura+1;
figure(nfigura);
nmax = size(S,2);
correlaciones = zeros(nmax+1,100);
[IS,S] =
matriz_proximidad_espacial(tiempo(nval:end),Zespaciales(nval:end,:),nmax);
for(i=(niter-99):niter)
    correlaciones(:,i-niter+100) =
    autocorrelaciones_espaciales(errores(:,i),IS,S,0);
end
correlaciones = prctile(correlaciones',[2.5 50 97.5]);
plot(0:nmax,correlaciones(:,2),0:nmax,zeros(nmax+1,1),'b--
',0:nmax,correlaciones(:,1),'b':,0:nmax,correlaciones(:,3),'b:');
xlabel('Numero de vecinos');
ylabel('Autocorrelaciones empiricas');
% Predicciones puntuales e intervalos de credibilidad del logaritmo de los precios
nfigura = nfigura+1;
figure(nfigura);
ypred = prctile(ypred',[0.5 2.5 50 97.5 99.5]);
plot(1:n,Y1,'k',1:n,ypred(:,3),'r',1:n,ypred(:,1),'b:',1:n,ypred(:,2),'g:',1:n,ypred(:,4),'g:',1:n,ypred(:,5),'b:');
legend('Y','Ypred','Yinf99','Yinf95','Ysup95','Ysup99',0);
xlabel('Observacion');
ylabel('Log-precio');
xlim([0 length(Y1)]);
% Predicciones puntuales e intervalos de credibilidad de los precios
nfigura = nfigura+1;
figure(nfigura);
plot(1:n,exp(Y1),'k',1:n,exp(ypred(:,3)), 'r',1:n,exp(ypred(:,1)), 'b:',1:n,exp(ypred(:,2)), 'g:',1:n,exp(ypred(:,4)), 'g:',1:n,ex
p(ypred(:,5)), 'b:');
legend('P','Ppred','Pinf99','Pinf95','Psup95','Psup99',0);
xlabel('Observacion');
ylabel('Precio');
xlim([0 length(Y1)]);
% Calculo de los criterios numericos
error = Y1-ypred(:,3);
MAD = mean(abs(error));
CUB95 = 100*mean((Y1<=ypred(:,4)).*(Y1>=ypred(:,2)));
CUB99 = 100*mean((Y1<=ypred(:,5)).*(Y1>=ypred(:,1)));
MSE = mean((Y1-medias).^2);
R2 = 1 - MSE/var(Y1)
SESGO = sum((Y1-medias).^2);
LOSS = SESGO+VARIANZA;

```

Programas correspondientes al capítulo 4

Programa 4.1.- Este programa implementa el Gibbs sampling para un modelo hedónico% de la forma $Y = X*\beta + \epsilon$ con $\epsilon \sim N(0, \sigma^2 I(n))$ siendo la distribución a priori $\beta \sim N(0, \Sigma_{\beta})$ $\tau \sim \text{Gamma}(d_0, s_0)$; nitermax es el número máximo de iteraciones. Devuelve la muestra de la distribución a posteriori

```
function muestra = gibbs_hedonico_norm(Y,X,Sigma_beta,d0,s0,n0,nitermax);
% Creacion de las variables auxiliares
Y = Y(n0:end);
X = X(n0:end,:);
n = length(Y);
p = size(X,2);
Sigma_beta = inv(Sigma_beta);
% Inicializacion de las variables
betas = zeros(p,nitermax);
fis = zeros(8,nitermax);
taus = zeros(nitermax,1);
mT = taus;
mS = mT;
lambdas = mT;
gammas = mT;
df_tau = n+d0;
taus(1) = s0;
% Ejecucion del algoritmo
for(it=2:nitermax)
    % Distribucion condicionada de beta
    S_beta = Sigma_beta + taus(it-1)*(X'*X);
    S_beta = inv(S_beta);
    m_beta = S_beta*taus(it-1)*(X'*Y);
    betas(:,it) = mvnrnd(m_beta,S_beta);
    % Distribucion condicionada de tau
    error = Y - X*betas(:,it);
    a = d0*s0 + sum(error.*error);
    taus(it) = gamrnd(0.5*df_tau,2/a);
end
% Almacenamiento de la muestra
muestra{1} = betas;
muestra{2} = fis;
muestra{3} = taus;
muestra{4} = mT;
muestra{5} = gammas;
muestra{6} = mS;
muestra{7} = lambdas;
```

Programa 4.2- Este programa implementa el Gibbs sampling para un modelo de Pace y otros (1998, 2000) de la forma $Y = Z*\alpha + X*\beta + \text{indfi}(1)*\text{fiTX}*T*(Y-X*\beta) + \text{indfi}(2)*\text{fiSX}*S*(Y-X*\beta) + \text{indfi}(3)*\text{fiSTX}*S*T*(Y-X*\beta) + \text{indfi}(4)*\text{fiTSX}*T*S*(Y-X*\beta) + \text{indfi}(5)*\text{fiT}*T*Y + \text{indfi}(6)*\text{fiS}*S*Y + \text{indfi}(7)*\text{fiST}*S*T*Y + \text{indfi}(8)*\text{fiTS}*T*S*Y + \text{epsilon}$ con $\text{epsilon} \sim N(0, \sigma^2 I(n))$ siendo la distribucion a priori $\alpha \sim N(0, \text{Sigma_alfa})$, $\beta \sim N(0, \text{Sigma_beta})$ independientes $\text{fi}_i \sim U(-1, 1)$ para $i = \text{TX, SX, TSX, STX, TY, SY, TSY, STY}$ $\tau \sim \text{Gamma}(d0/2, (d0*s0)/2)$; $mT \sim \text{Unif}(mTred)$ $mS \sim \text{Unif}(mSred)$ $\lambda \sim \text{Unif}(\lambda_{\min}, \lambda_{\max})$ $\gamma \sim \text{Unif}(\gamma_{\min}, \gamma_{\max})$. El vector `indicesTS` indica que componentes de la parte no parametrica del modelo se quieren incluir en el proceso de estimacion `indmuestra0 = 1` si se da un valor inicial de la muestra en `muestra0` En otro caso toma el valor inicial de la distribucion a priori `nitermax` es el numero maximo de iteraciones El programa devuelve la muestra de la distribucion a posteriori

```
function muestra = gibbs_pace_c_norm(Y,Z,X,IT,T,IS,S,indicesTS,priori,n0,nitermax,indmuestra0,muestra0);
% Creacion de variables auxiliares
Y1 = Y;
X1 = X;
Y = Y((n0+1):end);
X = X((n0+1):end,:);
Z = Z((n0+1):end,:);
n = length(Y);
q = size(Z,2);
p = size(X,2);
% Lectura de la distribucion a priori
Sigma_alfa = inv(priori.Sigma_alfa);
Sigma_beta = inv(priori.Sigma_beta);
mTred = priori.mTred;
mSred = priori.mSred;
gamma_min = priori.gamma_min;
gamma_max = priori.gamma_max;
lambda_min = priori.lambda_min;
lambda_max = priori.lambda_max;
d0 = priori.d0;
s0 = priori.s0;
% Inicializacion de las variables
Sigma = zeros(8+q,8+q);
if(q>0)
    Sigma(9:end,9:end) = Sigma_alfa;
end
if(p>0)
    betas = zeros(p,nitermax);
end
fis = zeros(8+q,nitermax);
taus = zeros(nitermax,1);
```

Programas correspondientes al capítulo 4

```

df_tau = n+d0;
mT = taus;
mS = mT;
lambdas = mT;
gammas = mT;
nmT = length(mTred);
% Lectura de la muestra inicial
if(indmuestra0==1)
    mT(1) = muestra0.mT;
    mS(1) = muestra0.mS;
    lambdas(1) = muestra0.lambdas;
    gammas(1) = muestra0.gammas;
    taus(1) = muestra0.taus;
    fis(1:8,1) = muestra0.fis;
    if(q>0)
        alfas(:,1) = muestra0.alfas;
    end
    if(p>0)
        betas(:,1) = muestra0.betas;
    end
else
    if(sum(indicesTS>0))
        mT(1) = mTred(unidrnd(nmT));
        gammas(1) = unifrnd(gamma_min,gamma_max);
        nmS = length(mSred);
        mS(1) = mSred(unidrnd(nmS));
        lambdas(1) = unifrnd(lambda_min,lambda_max);
    end
    taus(1) = 1/s0;
end
% Ejecucion del algoritmo
for(it=2:nitermax)
    [TX,SX,TSX,STX,TY,SY,TSY,STY] = calculo_TS_c(Y1,X1,IT,T,IS,S,mT(it-1),gammas(it-1),mS(it-1),lambdas(it-1));
    TX = TX((n0+1):end,:);
    SX = SX((n0+1):end,:);
    TSX = TSX((n0+1):end,:);
    STX = STX((n0+1):end,:);
    TY = TY((n0+1):end);
    SY = SY((n0+1):end);
    TSY = TSY((n0+1):end);
    STY = STY((n0+1):end);
    % Distribucion condicionada de beta
    if(p>0)
        U = X-fis(1,it-1)*TX-fis(2,it-1)*SX-fis(3,it-1)*TSX-fis(4,it-1)*STX;
        V = Y-(fis(1,it-1)+fis(5,it-1))*TY-(fis(2,it-1)+fis(6,it-1))*SY-(fis(3,it-1)+fis(7,it-1))*TSY-(fis(4,it-1)+fis(8,it-1))*STY;
        if(q>0)

```

```

    V = V -Z*fis(9:end,it-1);
end
S_beta = Sigma_beta + taus(it-1)*(U*U);
S_beta = inv(S_beta);
m_beta = S_beta*taus(it-1)*(U*V);
betas(:,it) = mvnrnd(m_beta,S_beta);
end
% Distribuciones condicionadas de los fis
if(p>0)
    u1 = Y - X*betas(:,it);
else
    u1 = Y;
end
U = zeros(n,q+8);
if(p>0)
    U(:,1) = TY-TX*betas(:,it);
    U(:,2) = SY-SX*betas(:,it);
    U(:,3) = TSY-TSX*betas(:,it);
    U(:,4) = STY-STX*betas(:,it);
end
U(:,5) = TY;
U(:,6) = SY;
U(:,7) = TSY;
U(:,8) = STY;
if(q>0)
    U(:,9:end) = Z;
end
if(mT(it-1)>0)
    if(mS(it-1)>0)
        indices = 1:8;
        indices = indices(indicesTS==1);
    else
        indices = [1,5];
        indices = indices(indicesTS(indices)==1);
    end
else
    if(mS(it-1)>0)
        indices = [2,6];
        indices = indices(indicesTS(indices)==1);
    else
        indices = [];
    end
end
if(q>0)
    indices = [indices,9:(8+q)];
end
if(length(indices)>0)

```

Programas correspondientes al capítulo 4

```

Si = taus(it-1)*(U(:,indices)*U(:,indices))+Sigma(indices,indices);
Si = inv(Si);
mi = taus(it-1)*(U(:,indices)*u1);
mi = Si*mi;
ind=1;
contador = 0;
while(ind>=1 & contador<=10)
    contador = contador+1;
    fis(indices,it) = mvnrnd(mi,Si);
    ind=sum(abs(fis(1:8,it))>1);
end
if(contador>10)
    fis(indices,it) = fis(indices,it-1);
    for(j=1:length(indices))
        i = indices(j);
        indices1 = indices(indices~=i);
        Vi = u1 - U(:,indices1)*fis(indices1,it);
        Ui = U(:,i);
        Si = taus(it-1)*(Ui*Ui)+Sigma(i,i);
        mi = (taus(it-1)*(Ui*Vi));
        Si = 1/Si;
        mi = Si*mi;
        Si = sqrt(Si);
        fi_inf = normcdf(-1,mi,Si);
        fi_sup = normcdf(1,mi,Si);
        u = unifrnd(0,1);
        f = u*fi_inf + (1-u)*fi_sup;
        if(0<f & f<1)
            fis(i,it) = norminv(f,mi,Si);
        elseif(f==0)
            fis(i,it) = -0.9999;
        else
            fis(i,it) = 0.9999;
        end
    end
end
Vi = u1-U(:,1:8)*fis(1:8,it);
Ui = U(:,9:end);
Si = taus(it-1)*(Ui*Ui)+Sigma(9:end,9:end);
Si = inv(Si);
mi = (taus(it-1)*(Ui*Vi));
mi = Si*mi;
if(q>0)
    fis(9:end,it) = mvnrnd(mi,Si);
end
end
end
% Distribucion condicionada de tau

```

```

error = u1-U*fis(:,it);
a = d0*s0 + sum(error.*error);
taus(it) = gamrnd(0.5*df_tau,2/a);
% Distribucion condicionada de (mT,mS)
U = zeros(n,8+q);
if(mT(it-1)>0)
    if(p>0)
        U(:,1) = TY-TX*betas(:,it);
    end
    U(:,5) = TY;
end
if(mS(it-1)>0)
    if(p>0)
        U(:,2) = SY-SX*betas(:,it);
    end
    U(:,6) = SY;
end
if(mT(it-1)*mS(it-1)>0)
    if(p>0)
        U(:,3) = TSY-TSX*betas(:,it);
        U(:,4) = STY-STX*betas(:,it);
    end
    U(:,7) = TSY;
    U(:,8) = STY;
end
if(q>0)
    U(:,9:end) = Z;
end
error = u1-U*fis(:,it);
prob = -0.5*taus(it)*sum(error.*error);
mT_estrella = mTred(unidrnd(nmT));
mS_estrella = mSred(unidrnd(nmS));
[TX,SX,TSX,STX,TY,SY,TSY,STY] = calculo_TS_c(Y1,X1,IT,T,IS,S,mT_estrella,gammas(it-
1),mS_estrella,lambdas(it-1));
TX = TX((n0+1):end,:);
SX = SX((n0+1):end,:);
TSX = TSX((n0+1):end,:);
STX = STX((n0+1):end,:);
TY = TY((n0+1):end);
SY = SY((n0+1):end);
TSY = TSY((n0+1):end);
STY = STY((n0+1):end);
U(:,1:8) = zeros(n,8);
if(mT_estrella>0)
    if(p>0)
        U(:,1) = TY-TX*betas(:,it);
    end
    U(:,5) = TY;

```


Programas correspondientes al capítulo 4

```

end
if(mS_estrella>0)
    if(p>0)
        U(:,2) = SY-SX*betas(:,it);
    end
    U(:,6) = SY;
end
if(mT_estrella*mS_estrella>0)
    if(p>0)
        U(:,3) = TSY-TSX*betas(:,it);
        U(:,4) = STY-STX*betas(:,it);
    end
    U(:,7) = TSY;
    U(:,8) = STY;
end
error = u1-U*fis(:,it);
prob_estrella = -0.5*taus(it)*sum(error.*error);
alfa = exp(prob_estrella-prob);
u = unifrnd(0,1);
if(u<=alfa)
    mT(it) = mT_estrella;
    mS(it) = mS_estrella;
    prob = prob_estrella;
else
    mT(it) = mT(it-1);
    mS(it) = mS(it-1);
end
% Distribucion de (lambda,gamma)
lambda_estrella = unifrnd(lambda_min,lambda_max);
gamma_estrella = unifrnd(gamma_min,gamma_max);
[TX,SX,TSX,STX,TY,SY,TSY,STY] =
calculo_TS_c(Y1,X1,IT,T,IS,S,mT(it),gamma_estrella,mS(it),lambda_estrella);
TX = TX((n0+1):end,:);
SX = SX((n0+1):end,:);
TSX = TSX((n0+1):end,:);
STX = STX((n0+1):end,:);
TY = TY((n0+1):end);
SY = SY((n0+1):end);
TSY = TSY((n0+1):end);
STY = STY((n0+1):end);
U = zeros(n,8+q);
if(mT(it)>0)
    if(p>0)
        U(:,1) = TY-TX*betas(:,it);
    end
    U(:,5) = TY;
end
if(mS(it)>0)

```

```

if(p>0)
    U(:,2) = SY-SX*betas(:,it);
end
U(:,6) = SY;
end
if(mS(it)*mT(it)>0)
    if(p>0)
        U(:,3) = TSY-TSX*betas(:,it);
        U(:,4) = STY-STX*betas(:,it);
    end
    U(:,7) = TSY;
    U(:,8) = STY;
end
if(q>0)
    U(:,9:end) = Z;
end
error = u1-U*fis(:,it);
prob_estrella = -0.5*taus(it)*sum(error.*error);
alfa = exp(prob_estrella-prob);
alfa = min(1,alfa);
u = unifrnd(0,1);
if(u<=alfa)
    lambdas(it) = lambda_estrella;
    gammas(it)= gamma_estrella;
else
    lambdas(it) = lambdas(it-1);
    gammas(it) = gammas(it-1);
end
end
% Almacenamiento de la muestra
if(p>0)
    muestra.betas = betas;
end
if(q>0)
    muestra.alfas = alfas;
end
muestra.fis = fis;
muestra.taus = taus;
muestra.mT = mT;
muestra.gammas = gammas;
muestra.mS = mS;
muestra.lambdas = lambdas;

```

Programa 4.3.- Este programa calcula las predicciones extramuestrales del modelo de Beamonte y otros (2006) analizado desde una optica bayesiana utilizando la muestra de la

Programas correspondientes al capítulo 4

distribucion a posteriori calculada con los algoritmo gibbs_pace_c_norm y gibbs_pace_c_student nval indica el inicio del periodo de validacion ind_student = 1 si la muestra se ha obtenido con gibbs_pace_c_student y 0 si se ha obtenido con gibbs_pace_norm niter0 contiene el inicio del periodo de toma de la muestra step contiene el numero de pasos para reducir el valor de la autocorrelacion de la muestra a valores poco significativos Utiliza el muestreo de composicion Devuelve el valor de diversos criterios numericos y hace un estudio grafico de las predicciones

```
function [CUB95,CUB99,AJUSTE,PD,DIC,LPRED,R2] =
pred_bay_c(Y,Z,X,tiempo,coordenadas,IT,T,IS,S,nval,muestra,niter0,step,indstudent,indextra);
% Lectura de la muestra
p = size(X,2);
q = size(Z,2);
if(p>0)
    betas = muestra.betas;
end
if(q>0)
    alfas = muestra.alfas;
end
fis = muestra.fis;
taus = muestra.taus;
mT = muestra.mT;
gammas = muestra.gammas;
mS = muestra.mS;
lambdas = muestra.lambdas;
if(p>0)
    betas = betas(:,niter0:step:end);
end
if(q>0)
    alfas = alfas(:,niter0:step:end);
end
fis = fis(:,niter0:step:end);
taus = taus(niter0:step:end);
mT = mT(niter0:step:end);
gammas = gammas(niter0:step:end);
mS = mS(niter0:step:end);
lambdas = lambdas(niter0:step:end);
niter = size(fis,2);
n = length(Y)-nval+1;
if(indstudent==1)
    r = muestra.r;
    r = r(niter0:step:end);
    if(indextra==1)
        vi = zeros(n,niter);
        for(i=1:n)
            indices = unidrnd(niter,1,niter);
```

```

    r1 = r(indices);
    vi(i,:) = gamrnd(0.5*r1,2./r1,1,niter);
end
else
    vi = muestra.vi;
    vi = vi(:,niter0:step:end);
end
else
    vi = ones(n,niter);
end
% Iniciacion de las variables
ypred = zeros(n,niter);
Y1 = Y(nval:end);
LLF=0;
LPRED = 0;
medias = zeros(n,niter);
errores = zeros(n,niter);
% Algoritmo de prediccion
for(it=1:niter)
    if(rem(it,100)==0)
        it
    end
    media = zeros(n,1);
    if(q>0)
        media = media+Z(nval:end,:)*alfas(:,it);
    end
    if(p>0)
        media = media + X(nval:end,:)*betas(:,it);
    end
    [TX,SX,TSX,STX,TY,SY,TSY,STY] = calculo_TS_c(Y,X,IT,T,IS,S,mT(it),gammas(it),mS(it),lambdas(it));
    TX = TX(nval:end,:);
    SX = SX(nval:end,:);
    TSX = TSX(nval:end,:);
    STX = STX(nval:end,:);
    TY = TY(nval:end);
    SY = SY(nval:end);
    TSY = TSY(nval:end);
    STY = STY(nval:end);
    if(mT(it)>0)
        if(p>0)
            media = media + fis(1,it)*(TY-TX*betas(:,it));
        end
        media = media +fis(5,it)*TY;
    end
    if(mS(it)>0)
        if(p>0)
            media = media + fis(2,it)*(SY-SX*betas(:,it));
        end
    end
end

```

Programas correspondientes al capítulo 4

```

end
media = media + fis(6,it)*SY;
end
if(mT(it)*mS(it)>0)
if(p>0)
media = media + fis(3,it)*(TSY-TSX*betas(:,it));
media = media + fis(4,it)*(STY-STX*betas(:,it));
end
media = media + fis(7,it)*TSY;
media = media + fis(8,it)*STY;
end
if(indstudent==0)
ypred(:,it) = normrnd(media,sqrt(1/taus(it)));
else
ypred(:,it) = media + sqrt(1/taus(it))*trnd(r(it),length(Y1),1);
end
errores(:,it) = sqrt(taus(it)).*(Y1-ypred(:,it));
medias(:,it) = media;
if(indstudent==0)
LLF = LLF + sum(log(normpdf(Y1,media,sqrt(1./(vi(:,it)*taus(it))))));
LPRED = LPRED + prod(normpdf(Y1,media,sqrt(1./(vi(:,it)*taus(it)))));
else
LLF = LLF + sum(log(sqrt(taus(it))*tpdf(sqrt(taus(it))*(Y1-media),r(it))));
LPRED = LPRED + prod(sqrt(taus(it))*tpdf(sqrt(taus(it))*(Y1-media),r(it)));
end
end
end
% Calculo del criterio DIC y del logaritmo de la densidad predictiva
AJUSTE = -2*LLF/niter;
LPRED = log(LPRED/niter);
% Calculo de la media a posteriori
if(p>0)
betas_media = mean(betas)';
end
if(q>0)
alfas_media = mean(alfas)';
end
fis_media = mean(fis)';
taus_media = mean(taus);
mT_media = round(mean(mT));
gammas_media = mean(gammas);
mS_media = round(mean(mS));
lambdas_media = mean(lambdas);
if(indstudent==1)
r_media = mean(r);
else
vi_media = ones(n,1);
end
end

```

```

[TX,SX,TSX,STX,TY,SY,TSY,STY] =
calculo_TS_c(Y,X,IT,T,IS,S,mT_media,gammas_media,mS_media,lambda_media);
TX = TX(nval:end,:);
SX = SX(nval:end,:);
TSX = TSX(nval:end,:);
STX = STX(nval:end,:);
TY = TY(nval:end);
SY = SY(nval:end);
TSY = TSY(nval:end);
STY = STY(nval:end);
media = 0;
if(q>0)
    media = media + Z(nval:end,:)*alfas_media;
end
if(p>0)
    media = media + X(nval:end,:)*betas_media;
end
if(mT_media>0)
    if(p>0)
        media = media + fis_media(1)*(TY-TX*betas_media);
    end
    media = media + fis_media(5)*TY;
end
if(mS_media>0)
    if(p>0)
        media = media + fis_media(2)*(SY-SX*betas_media);
    end
    media = media + fis_media(6)*SY;
end
if(mT_media*mS_media>0)
    if(p>0)
        media = media + fis_media(3)*(TSY-TSX*betas_media);
        media = media + fis_media(4)*(STY-STX*betas_media);
    end
    media = media + fis_media(7)*TSY;
    media = media + fis_media(8)*STY;
end
if(indstudent==0)
    PD = AJUSTE + 2*sum(log(normpdf(Y1,media,sqrt(1/taus_media))));
else
    PD = AJUSTE + 2*sum(log(sqrt(taus_media)*tpdf(sqrt(taus_media)*(Y1-media),r_media)));
end
DIC = PD + AJUSTE;
% Errores de prediccion
error = prtileerrores',[0.5 2.5 50 97.5 99.5]';
errores1 = errores;
CUB95 = 100*mean((error(:,2)<=0).*(error(:,4)>=0));
CUB99 = 100*mean((error(:,1)<=0).*(error(:,5)>=0));

```

Programas correspondientes al capítulo 4

```

nfigura = 1;
figure(nfigura);
plot(1:n,error(:,3),'r',1:n,error(:,1),'b',1:n,error(:,2),'g',1:n,error(:,4),'g',1:n,error(:,5),'b',1:n,zeros(n,1),'k');
legend('error','errorinf99','errorinf95','errorsup95','errorsup99',0);
xlabel('Observacion');
ylabel('Error');
xlim([0 length(Y1)]);
% Estudio de las autocorrelaciones residuales con bandas de confianza del 95%
nfigura = nfigura+1;
figure(nfigura);
nmax = size(S,2);
[IS1,S1] = matriz_proximidad_espacial(tiempo(nval:end),coordenadas(nval:end,:),nmax);
correlaciones = zeros(nmax+1,100);
for(i=(niter-99):niter)
    correlaciones(:,i-niter+100) = autocorrelaciones_espacialeserrores(:,i),IS1,S1,0);
end
correlaciones = prctile(correlaciones',[2.5 50 97.5]);
plot(0:nmax,correlaciones(:,2),0:nmax,zeros(nmax+1,1),'b--',0:nmax,correlaciones(:,1),'b',0:nmax,correlaciones(:,3),'b');
xlabel('Numero de vecinos')
ylabel('Autocorrelaciones')
% Estudio de normalidad de los errores tomando como estimador de los parametros la media a posteriori
zr = sqrt(taus_media)*(Y1-media);
estudio_zr =
[ min(zr),prctile(zr,[1,5,10,25,50,75,90,95,99]),max(zr),mean(zr),median(abs(zr)),skewness(zr),kurtosis(zr)]'
nfigura = nfigura+1;
figure(nfigura);
plot(1:length(zr),zr,1:length(zr),-2*ones(length(zr),1),'r',1:length(zr),2*ones(length(zr),1),'r');
xlim([0 length(Y1)]);
nfigura = nfigura+1;
figure(nfigura);
[f,xi] = ksdensity(zr);
plot(xi,f);
nfigura = nfigura+1;
figure(nfigura);
if(indstudent==1)
    qqplot(zr,trnd(r_media,length(zr),1));
else
    qqplot(zr);
end
% Predicciones puntuales del logaritmo de los precios e intervalos de credibilidad bayesianos
nfigura = nfigura+1;
figure(nfigura);
ypred1 = ypred;
ypred = prctile(ypred',[0.5 2.5 50 97.5 99.5]);
plot(1:n,Y1,'k',1:n,ypred(:,3),'r',1:n,ypred(:,1),'b',1:n,ypred(:,2),'g',1:n,ypred(:,4),'g',1:n,ypred(:,5),'b');
xlim([0 length(Y1)]);
legend('Y','Ypred','Yinf99','Yinf95','Ysup95','Ysup99',0);

```

```

% Predicciones puntuales de los precios e intervalos de credibilidad bayesianos
nfigura = nfigura+1;
figure(nfigura);
plot(1:n,exp(Y1),'k',1:n,exp(ypred(:,3)), 'r',1:n,exp(ypred(:,1)), 'b',1:n,exp(ypred(:,2)), 'g',1:n,exp(ypred(:,4)), 'g',1:n,exp(ypred(:,5)), 'b');
legend('P', 'Ppred', 'Pinf99', 'Pinf95', 'Psup95', 'Psup99', 0);
xlim([0 length(Y1)]);
% Localizacion de atipicos
if(indstudent==1)
    nfigura = nfigura+1;
    figure(nfigura);
    vi_cuantiles = prctile(vi,[2.5 50 97.5]);
    plot(1:n,ones(n,1),'k',1:n,vi_cuantiles(:,2), 'b--',1:n,vi_cuantiles(:,1), 'g',1:n,vi_cuantiles(:,3), 'r');
    xlim([0 length(Y1)]);
end
% Calculo de R2
ypred_media = mean(medias,2);
error1 = Y1-ypred_media;
R2 = 1-mean(error1.*error1)/var(Y1);

```

Programa 4.4.- Este programa calcula las predicciones extramuestrales del modelo de Beamonte y otros (2006) analizado desde una optica bayesiana utilizando la muestra de la distribucion a posteriori calculada con los algoritmo gibbs pace_c_norm (ind_student=0) y gibbs_pace_student (ind_student=1) nval indica el inicio del periodo de validacion Utiliza el muestreo de composicion y se aplica en el programa validacion_bay_c_rolling Devuelve una muestra de la distribucion predictiva, su media y su varianza, el logaritmo de la densidad predictiva y los errores de prediccion tipificados Realiza, ademas, un estudio grafico de las predicciones

```

function [ypred,medias,varianzas,LLF,errores] =
pred_bay_c_rolling(Y,Z,X,TX,SX,STX,TSX,TY,SY,STY,TSY,muestra,w1,ind_student);
% Lectura de la muestra
p = size(X,2);
q = size(Z,2);
if(p>0)
    betas = muestra.betas;
end
if(q>0)
    alfas = muestra.alfas;
end
fis = muestra.fis;
taus = muestra.taus;
mT = muestra.mT;
gammas = muestra.gammas;
mS = muestra.mS;
lambdas = muestra.lambdas;

```


Programas correspondientes al capítulo 4

```
niter = size(fis,2);
if(ind_student==1)
    r = muestra.r;
end
% Extraccion de la muestra de acuerdo a los pesos de importancia w1
if(var(w1)>0)
    u = unifrnd(0,1,niter,1);
    indices_simulados = ones(niter,1);
    for(i=2:niter)
        w1(i) = w1(i-1)+w1(i);
        indices_simulados = indices_simulados + uplus(u>=w1(i));
    end
    if(q>0)
        alfas = alfas(:,indices_simulados);
    end
    if(p>0)
        betas = betas(:,indices_simulados);
        TX = TX(:,indices_simulados);
        SX = SX(:,indices_simulados);
        STX = STX(:,indices_simulados);
        TSX = TSX(:,indices_simulados);
    end
    fis = fis(:,indices_simulados);
    taus = taus(indices_simulados);
    mT = mT(indices_simulados);
    mS = mS(indices_simulados);
    TY = TY(indices_simulados);
    SY = SY(indices_simulados);
    STY = STY(indices_simulados);
    TSY = TSY(indices_simulados);
    if(ind_student==1)
        r = r(indices_simulados);
    end
end
% Iniciacion de las variables
ypred = zeros(1,niter);
errores = ypred;
medias = 0;
medias2 = medias;
LLF=0;
% Algoritmo de prediccion
for(it=1:niter)
    media = 0;
    if(q>0)
        media = media+Z*alfas(:,it);
    end
    if(p>0)
```

```

    media = media + X*betas(:,it);
end
if(mT(it)>0)
    if(p>0)
        media = media + fis(1,it)*(TY(it)-TX(:,it))*betas(:,it);
    end
    media = media +fis(5,it)*TY(it);
end
if(mS(it)>0)
    if(p>0)
        media = media + fis(2,it)*(SY(it)-SX(:,it))*betas(:,it);
    end
    media = media +fis(6,it)*SY(it);
end
if(mT(it)*mS(it)>0)
    if(p>0)
        media = media + fis(3,it)*(TSY(it)-TSX(:,it))*betas(:,it);
        media = media + fis(4,it)*(STY(it)-STX(:,it))*betas(:,it);
    end
    media = media + fis(7,it)*TSY(it);
    media = media + fis(8,it)*STY(it);
end
medias = medias+media;
medias2 = medias2 + media*media;
if(ind_student==0)
    ypred(1,it) = normrnd(media,sqrt(1/taus(it)));
else
    ypred(1,it) = media + (1/sqrt(taus(it)))*trnd(r(it));
end
errores(1,it) = sqrt(taus(it))*(Y - ypred(1,it));
if(ind_student==0)
    LLF = LLF + log(normpdf(Y,media,sqrt(1./taus(it))));
else
    LLF = LLF + log(sqrt(taus(it))*tpdf(sqrt(taus(it))*(Y-media),r(it)));
end
end
% Calculo de las media, varianza y logaritmo de la densidad predictiva
medias = medias/niter;
varianzas = medias2/niter-medias.*medias;
LLF = LLF/niter;

```

Programa 4.5.- Este programa devuelve los parametros de la distribucion a priori utilizada en los programas gibbs_pace_c_student y gibbs_pace_c_norm

```
function priori = prior_c();
```

Programas correspondientes al capítulo 4

```
priori.Sigma_alfa = 10000*eye(3);
priori.Sigma_beta = 10000*eye(5);
priori.d0 = 0.01;
priori.s0 = 1;
priori.m1 = 1;
priori.m2 = 0.25;
priori.mTred = 0:50;
priori.mSred = 0:50;
priori.gamma_min = 0;
priori.gamma_max = 1;
priori.lambda_min = 0;
priori.lambda_max = 1;
```

Programa 4.6.- Este programa realiza una validacion extramuestral del modelo de Pace y otros (1998, 2000) tipo rolling analizado desde una optica bayesiana utilizando la muestra de la distribucion a posteriori calculada con el algoritmo gibbs pace nval indica el inicio del periodo de validacion nval indica el inicio del periodo de validacion Utiliza el muestreo de importancia secuencial descrito en el Capitulo 3 La muestra inicial se supone obtenida por medio del Gibbs sampling aplicado en el periodo de estimacion 1, ..., nval-1 ind_student = 1 si se utiliza el programa gibbs_pace_student y 0 si se utiliza el programa gibbs_pace_norm niter0 contiene el inicio del periodo de toma de la muestra step contiene el numero de pasos para reducir el valor de la autocorrelacion de la muestra a valores poco significativos Devuelve los valores de diversos criterios de evaluacion de la actividad predictiva entre los que destaca el vector LPRED que contiene los logaritmos de la densidad predictiva en cada periodo Realiza, ademas, un estudio grafico de los resultados

```
function [RMSE,MAD,CUB95,CUB99,LPRED] =
validacion_bay_c_rolling(Y,tiempo,Z,X,Zespaciales,indicesTS,IT,T,IS,S,priori,n0,nval,muestra,niter0,step,ind_stude
nt);
% Lectura de la distribucion a priori
Sigma_alfa = inv(priori.Sigma_alfa);
Sigma_beta = inv(priori.Sigma_beta);
mTred = priori.mTred;
mSred = priori.mSred;
gamma_min = priori.gamma_min;
gamma_max = priori.gamma_max;
lambda_min = priori.lambda_min;
lambda_max = priori.lambda_max;
d0 = priori.d0;
s0 = priori.s0;
if(ind_student==1)
    m1 = priori.m1;
    m2 = priori.m2;
end
%
% Lectura de la muestra de la distribucion a posteriori
%
q = size(Z,2);
p = size(X,2);
if(q>0)
    alfas = muestra.alfas(:,niter0:step:end);
    muestra.alfas = alfas;
end
```

```

if(p>0)
    betas = muestra.betas(:,niter0:step:end);
    muestra.betas = betas;
end
fis = muestra.fis(:,niter0:step:end);
taus = muestra.taus(niter0:step:end);
mT = muestra.mT(niter0:step:end);
gammas = muestra.gammas(niter0:step:end);
mS = muestra.mS(niter0:step:end);
lambdas = muestra.lambdas(niter0:step:end);
niter = size(fis,2);
if(ind_student==1)
    vi = muestra.vi(:,niter0:step:end);
    r = muestra.r(niter0:step:end);
else
    vi = ones(nval-1,niter);
end
muestra.fis = fis;
muestra.taus = taus;
muestra.mT = mT;
muestra.gammas = gammas;
muestra.mS = mS;
muestra.lambdas = lambdas;
if(ind_student==1)
    muestra.vi = vi;
    muestra.r = r;
else
    muestra.vi = vi;
end
% Inicializacion de las variables
n = length(Y);
ypred = zeros(n-nval+1,niter);
errores = ypred;
medias = zeros(n-nval+1,1);
LPRED = medias;
VARIANZA = 0;
w = zeros(1,niter);
w1 = ones(1,niter)/niter;
TY = zeros(1,niter);
SY = TY;
STY = TY;
TSY = TY;
if(p>0)
    TX = zeros(p,niter);
    SX = TX;
    STX = TX;
    TSX = TX;
else
    TX = zeros(1,niter);
    SX = TX;
    STX = TX;
    TSX = TX;
end
for(it=1:niter)
    [TX1,SX1,TSX1,STX1,TY1,SY1,TSY1,STY1] =
calculo_TS_c(Y(1:nval),X(1:nval,:),IT(1:nval,:),T(1:nval,:),IS(1:nval,:),S(1:nval,:),mT(it),gammas(it),mS(it),lambdas
(it));
    TY(it) = TY1(nval);
    SY(it) = SY1(nval);
    STY(it) = STY1(nval);
    TSY(it) = TSY1(nval);
    TX(:,it) = TX1(nval,:);
    SX(:,it) = SX1(nval,:);

```

Programas correspondientes al capítulo 4

```

STX(:,it) = STX1(nval,:);
TSX(:,it) = TSX1(nval,:);
end
% Calculo de las predicciones y los criterios utilizando la informacion
% proporcionada por la muestra y aplicando el muestreo de composicion
% Utilizamos el muestreo de importancia para actualizar las distribuciones
% a posteriori en cada paso siempre y cuando su calidad (medida por el
% indicador ESS) sea alta
% En caso contrario se reinicia el proceso de muestreo
for(i=nval:n)
    if(i>nval)
        % Actualizacion de los pesos de importancia
        i1=i-1;
        for(it=1:niter)
            media = 0;
            if(q>0)
                media = media + Z(i1,:)*alfas(:,it);
            end
            if(p>0)
                media = media + X(i1,:)*betas(:,it);
            end
            [TX1,SX1,TSX1,STX1,TY1,SY1,TSY1,STY1] =
calculo_TS_c(Y(1:i1),X(1:i1,:),IT(1:i1,:),T(1:i1,:),IS(1:i1,:),S(1:i1,:),mT(it),gammas(it),mS(it),lambdas(it));
            TY(it) = TY1(i1);
            SY(it) = SY1(i1);
            STY(it) = STY1(i1);
            TSY(it) = TSY1(i1);
            TX(:,it) = TX1(i1,:);
            SX(:,it) = SX1(i1,:);
            STX(:,it) = STX1(i1,:);
            TSX(:,it) = TSX1(i1,:);
            if(mT(it)>0)
                if(p>0)
                    U = TY(it)-TX(:,it)*betas(:,it);
                else
                    U = 0;
                end
                end
            if(mS(it)>0)
                if(p>0)
                    U = [U,SY(it)-SX(:,it)*betas(:,it);
                    U = [U,TSY(it)-TSX(:,it)*betas(:,it)];
                    U = [U,STY(it)-STX(:,it)*betas(:,it)];
                else
                    U = [U,zeros(1,3)];
                end
                end
            U = [U,TY(it);
            U = [U,SY(it);
            U = [U,TSY(it);
            U = [U,STY(it);
            media = media + U*fis(:,it);
            else
                U = [U,TY(it);
                media = media + U*fis([1,5],it);
            end
            end
            else
                if(mS(it)>0)
                    if(p>0)
                        U = SY(it)-SX(:,it)*betas(:,it);
                    else
                        U = 0;
                    end
                    end
                U = [U,SY(it);
                media = media + U*fis([2,6],it);
            end
        end
    end
end

```

```

        end
    end
    if(ind_student==1)
        w(it) = w(it) + log(tpdf(sqrt(taus(it))*(Y(i1)-media),r(it)));
    else
        w(it) = w(it) + log(normpdf(Y(i1),media,sqrt(1/taus(it))));
    end
end
w1 = exp(w-max(w));
w1 = w1/sum(w1);
periodo = 1:i1;
% Analisis de la calidad de los pesos
ESS = niter/(1+var(w1))
if(ESS<0.5*niter)
    % Si la calidad es escasa, rehacer el muestreo;
    if(q>0)
        muestra.alfas = muestra.alfas(:,end);
    end
    if(p>0)
        muestra.betas = muestra.betas(:,end);
    end
    muestra.fis = muestra.fis(:,end);
    muestra.taus = muestra.taus(end);
    muestra.mT = muestra.mT(end);
    muestra.gammas = muestra.gammas(end);
    muestra.mS = muestra.mS(end);
    muestra.lambdas = muestra.lambdas(end);
    if(ind_het==1)
        muestra.vi = muestra.vi(:,end);
        muestra.r = muestra.r(end);
    end
    if(p*q>0)
        if(ind_student==0)
            muestra =
gibbs_pace_c_norm(Y(periodo),Z(periodo,:),X(periodo:),IT(periodo:),T(periodo:),IS(periodo:),S(periodo:),indices
TS,priori,n0,niter,101,1,1,muestra);
        else
            muestra =
gibbs_pace_c_student(Y(periodo),Z(periodo,:),X(periodo:),IT(periodo:),T(periodo:),IS(periodo:),S(periodo:),indic
esTS,priori,n0,niter,101,1,1,muestra);
        end
    else
        if(p>0)
            if(ind_student==0)
                muestra =
gibbs_pace_c_norm(Y(periodo),[],X(periodo:),IT(periodo:),T(periodo:),IS(periodo:),S(periodo:),indicesTS,priori,
n0,niter,101,1,1,muestra);
            else
                muestra =
gibbs_pace_c_student(Y(periodo),[],X(periodo:),IT(periodo:),T(periodo:),IS(periodo:),S(periodo:),indicesTS,prior
i,n0,niter,101,1,1,muestra);
            end
        else
            if(ind_student==0)
                muestra =
gibbs_pace_c_norm(Y(periodo),Z(periodo:),[],IT(periodo:),T(periodo:),IS(periodo:),S(periodo:),indicesTS,priori,n
0,niter,101,1,1,muestra);
            else
                muestra =
gibbs_pace_c_student(Y(periodo),Z(periodo:),[],IT(periodo:),T(periodo:),IS(periodo:),S(periodo:),indicesTS,priori
,n0,niter,101,1,1,muestra);
            end
        end
    end
end
end
end

```

Programas correspondientes al capítulo 4

```

end
w1 = ones(1,niter)/niter;
if(q>0)
    alfas = muestra.alfas;
end
if(p>0)
    betas = muestra.betas;
end
fis = muestra.fis;
taus = muestra.taus;
mT = muestra.mT;
gammas = muestra.gammas;
mS = muestra.mS;
lambdas = muestra.lambdas;
if(ind_het==1)
    vi = muestra.vi;
    r = muestra.r;
end
end
end
% Calculo de las predicciones y del logaritmo de la densidad predictiva
if(p*q>0)
    [ypred(i-nval+1,:),medias(i-nval+1),varianzas,LLF1,errores(i-nval+1,:)] =
pred_bay_c_rolling(Y(i),Z(i,:),X(i,:),TX,SX,STX,TSX,TY,SY,STY,TSY,muestra,w1,ind_student);
else
    if(p>0)
        [ypred(i-nval+1,:),medias(i-nval+1),varianzas,LLF1,errores(i-nval+1,:)] =
pred_bay_c_rolling(Y(i),[],X(i,:),TX,SX,STX,TSX,TY,SY,STY,TSY,muestra,w1,ind_student);
    else
        [ypred(i-nval+1,:),medias(i-nval+1),varianzas,LLF1,errores(i-nval+1,:)] =
pred_bay_c_rolling(Y(i),Z(i,:),TX,SX,STX,TSX,TY,SY,STY,TSY,muestra,w1,ind_student);
    end
end
end
VARIANZA = VARIANZA+varianzas;
LPRED(i-nval+1) = LLF1;
end
% Errores de prediccion
Y1 = Y(nval:end);
n = length(Y1);
error = prctile(errores',[0.5 2.5 50 97.5 99.5]);
nfigura = 1;
figure(nfigura);
plot(1:n,error(:,3),'r',1:n,error(:,1),'b',1:n,error(:,2),'g',1:n,error(:,4),'g',1:n,error(:,5),'b',1:n,zeros(n,1),'k');
legend('error','errorinf99','errorinf95','errorsup95','errorsup99',0);
xlabel('Observacion');
ylabel('Error');
xlim([0 length(Y1)]);
% Estudios de las autocorrelaciones residuales con bandas de confianza del 95%
nfigura = nfigura+1;
figure(nfigura);
errores = Y1*ones(1,niter)-ypred;
nmax = size(S,2);
correlaciones = zeros(nmax+1,100);
[IS,S] = matriz_proximidad_espacial(tiempo(nval:end),Zespaciales(nval:end,:),nmax);
for(i=(niter-99):niter)
    correlaciones(:,i-niter+100) = autocorrelaciones_espaciales(errores(:,i),IS,S,0);
end
correlaciones = prctile(correlaciones',[2.5 50 97.5]);
plot(0:nmax,correlaciones(:,2),0:nmax,zeros(nmax+1,1),'b--',0:nmax,correlaciones(:,1),'b',0:nmax,correlaciones(:,3),'b');
xlabel('Numero de vecinos');
ylabel('Autocorrelaciones empiricas');
% Predicciones puntuales e intervalos de credibilidad del logaritmo de los precios

```

```

nfigura = nfigura+1;
figure(nfigura);
ypred = prctile(ypred',[0.5 2.5 50 97.5 99.5]);
plot(1:n,Y1,'k',1:n,ypred(:,3),'r',1:n,ypred(:,1),'b',1:n,ypred(:,2),'g',1:n,ypred(:,4),'g',1:n,ypred(:,5),'b');
legend('Y','Ypred','Yinf99','Yinf95','Ysup95','Ysup99',0);
xlabel('Observacion');
ylabel('Log-precio');
xlim([0 length(Y1)]);
% Predicciones puntuales e intervalos de credibilidad de los precios
nfigura = nfigura+1;
figure(nfigura);
plot(1:n,exp(Y1),'k',1:n,exp(ypred(:,3)),r',1:n,exp(ypred(:,1)),b',1:n,exp(ypred(:,2)),g',1:n,exp(ypred(:,4)),g',1:n,exp(ypred(:,5)),b');
legend('P','Ppred','Pinf99','Pinf95','Psup95','Psup99',0);
xlabel('Observacion');
ylabel('Precio');
xlim([0 length(Y1)]);
% Calculo de los criterios numericos
error = Y1-ypred(:,3);
MAD = mean(abs(error));
CUB95 = 100*mean((Y1<=ypred(:,4)).*(Y1>=ypred(:,2)));
CUB99 = 100*mean((Y1<=ypred(:,5)).*(Y1>=ypred(:,1)));
RMSE = sqrt(mean((Y1-medias).^2));

```

Programas correspondientes al capítulo 5

Programa 5.1.- Este programa implementa el Gibbs sampling para un modelo hedonico con errores t de Student de la forma $Y = X*\beta + \epsilon$ con $\epsilon \sim N(0, \sigma^2 V)$ $V = \text{diag}(v_1, \dots, v_n)$ siendo la distribucion a priori $\beta \sim N(0, \text{Sigma_beta})$ $r/v_i \sim \text{Chi}^2(r)$ $i=1, \dots, n$ $r \sim \text{Gamma}(m_1, m_2)$ $\tau = 1/\sigma \sim \text{Gamma}(d_0/2, (d_0*s_0)/2)$; nitermax es el numero maximo de iteraciones Devuelve la muestra de la distribucion a posteriori

```

function muestra = gibbs_hedonico_student(Y,X,priori,n0,nitermax);
Y = Y(n0:end);
X = X(n0:end,:);
n = length(Y);
p = size(X,2);
% Lectura de la distribucion a priori
d0 = priori.d0;
s0 = priori.s0;
m1 = priori.m1;
m2 = priori.m2;
Sigma_beta = inv(priori.Sigma_beta);
%
% Inicializacion de las variables
%
betas = zeros(p,nitermax);
fis = zeros(8,nitermax);
vi = zeros(n,nitermax);
taus = zeros(nitermax,1);

```


Programas correspondientes al capítulo 5

```

mT = taus;
mS = mT;
lambdas = mT;
gammas = mT;
r = mT;
taus(1) = 1/s0;
vi(:,1) = (m2/m1)*ones(n,1);
r(1) = m1/m2;
%
% Ejecucion del algoritmo
%
for(it=2:nitermax)
    if(rem(it,100) == 0)
        it
    end
    % Distribucion condicionada de beta
    V = diag(vi(:,it-1));
    S_beta = Sigma_beta + taus(it-1)*(X'*V*X);
    S_beta = inv(S_beta);
    m_beta = S_beta*taus(it-1)*(X'*V*Y);
    betas(:,it) = mvnrnd(m_beta,S_beta);
    % Distribucion condicionada de vi
    error = Y-X*betas(:,it);
    a = r(it-1) + taus(it-1)*error.*error;
    vi(:,it) = gamrnd(0.5*(r(it-1)+1)*ones(n,1),2./a);
    %
    % Distribucion condicionada de r: utilizamos un paso de
    % Hastings-Metropolis
    %
    media = (n/var(vi(:,it)) + m2)/(m1+0.5*n);
    a = (m1+0.5*n)/media;
    r_estrella = gamrnd(m1+0.5*n,1/a);
    u = unifrnd(0,1);
    alfa = log(gampdf(r(it-1),m1+0.5*n,1/a))-log(gampdf(r_estrella,m1+0.5*n,1/a));
    alfa = alfa + (n*0.5*(r_estrella-1)*log(r_estrella))-0.5*n*r_estrella*log(2)-
n*gammaLn(0.5*r_estrella)+0.5*r_estrella*(sum(log(vi(:,it)))-sum(vi(:,it))-m2)+(m1+0.5*n-1)*log(r_estrella);
    alfa = alfa - ((n*0.5*(r(it-1)-1)*log(r(it-1)))-0.5*n*r(it-1)*log(2)-n*gammaLn(0.5*r(it-1))+0.5*r(it-
1)*(sum(log(vi(:,it)))-sum(vi(:,it))-m2)+(m1+0.5*n-1)*log(r(it-1)));
    alfa = min(exp(alfa),1);
    if(u<=alfa)
        r(it) = r_estrella;
    else
        r(it) = r(it-1);
    end
    % Distribucion condicionada de tau
    V = diag(sqrt(vi(:,it)));
    error = V*(Y - X*betas(:,it));
    a = d0*s0+sum(error.*error);

```

```

    taus(it) = gamrnd(0.5*(n+d0),2/a);
end
% Almacenamiento de la muestra
muestra.betas = betas;
muestra.fis = fis;
muestra.taus = taus;
muestra.mT = mT;
muestra.gammas = gammas;
muestra.mS = mS;
muestra.lambdas = lambdas;
muestra.vi = vi;
muestra.r = r;

```

Programa 5.2.- Este programa implementa el Gibbs sampling para un modelo de Pace y otros (1998, 2000) de la forma $Y = Z*\alpha + X*\beta + \text{indfi}(1)*\text{fiTX}*T*(Y-X*\beta) + \text{indfi}(2)*\text{fiSX}*S*(Y-X*\beta) + \text{indfi}(3)*\text{fiSTX}*S*T*(Y-X*\beta) + \text{indfi}(4)*\text{fiTSX}*T*S*(Y-X*\beta) + \text{indfi}(5)*\text{fiT}*T*Y + \text{indfi}(6)*\text{fiS}*S*Y + \text{indfi}(7)*\text{fiST}*S*T*Y + \text{indfi}(8)*\text{fiTS}*T*S*Y + \text{epsilon}$ con $\text{epsilon} \sim N(0, \sigma^2 \text{inv}(V))$ $V = \text{diag}(v_1, \dots, v_n)$ siendo la distribución a priori $\alpha \sim N(0, \Sigma_\alpha)$, $\beta \sim N(0, \Sigma_\beta)$ independientes $r/v_i \sim \text{Chi}^2(r)$ $i=1, \dots, n$ $r \sim \text{Gamma}(m_1, m_2)$ $\text{fi}_i \sim U(-1, 1)$ para $i = \text{TX}, \text{SX}, \text{TSX}, \text{STX}, \text{TY}, \text{SY}, \text{TSY}, \text{STY}$ $\tau \sim \text{Gamma}(d_0/2, (d_0*s_0)/2)$; $mT \sim \text{Unif}(mTred)$ $mS \sim \text{Unif}(mSred)$ $\lambda \sim \text{Unif}(\lambda_{min}, \lambda_{max})$ $\gamma \sim \text{Unif}(\gamma_{min}, \gamma_{max})$ El vector indicesTS indica que componentes de la parte no paramétrica del modelo se quieren incluir en el proceso de estimación $\text{indmuestra} = 1$ si se da un valor inicial de la muestra en muestra En otro caso toma el valor inicial de la distribución a priori $nitermax$ es el número máximo de iteraciones El programa devuelve la muestra de la distribución a posteriori

```

function muestra = gibbs_pace_c_student(Y,Z,X,IT,T,IS,S,indicesTS,priori,n0,nitermax,indmuestra0,muestra0);
% Creacion de variables auxiliares
Y1 = Y;
X1 = X;
Y = Y((n0+1):end);
X = X((n0+1):end,:);
Z = Z((n0+1):end,:);
n = length(Y);
p = size(X,2);
q = size(Z,2);
% Lectura de la distribución a priori
Sigma_alfa = inv(priori.Sigma_alfa);
Sigma_beta = inv(priori.Sigma_beta);
mTred = priori.mTred;
mSred = priori.mSred;
gamma_min = priori.gamma_min;
gamma_max = priori.gamma_max;
lambda_min = priori.lambda_min;

```

Programas correspondientes al capítulo 5

```
lambda_max = priori.lambda_max;
d0 = priori.d0;
s0 = priori.s0;
m1 = priori.m1;
m2 = priori.m2;
%
% Inicializacion de las variables
%
Sigma = zeros(8+q,8+q);
if(q>0)
    Sigma(9:end,9:end) = Sigma_alfa;
end
if(p>0)
    betas = zeros(p,nitermax);
end
fis = zeros(8+q,nitermax);
vi = zeros(n,nitermax);
r = zeros(nitermax,1);
taus = zeros(nitermax,1);
mT = taus;
mS = mT;
lambdas = mT;
gammas = mT;
nmT = length(mTred);
if(indmuestra0==1)
    mT(1) = muestra0.mT;
    mS(1) = muestra0.mS;
    lambdas(1) = muestra0.lambdas;
    gammas(1) = muestra0.gammas;
    taus(1) = muestra0.taus;
    fis(1:8,1) = muestra0.fis;
    vi(:,1) = muestra0.vi;
    r(1) = muestra0.r;
    if(q>0)
        alfas(:,1) = muestra0.alfas;
    end
end
if(p>0)
    betas(:,1) = muestra0.betas;
end
else
    if(sum(indicesTS>0))
        mT(1) = mTred(unidrnd(nmT));
        gammas(1) = unifrnd(gamma_min,gamma_max);
        nmS = length(mSred);
        mS(1) = mSred(unidrnd(nmS));
        lambdas(1) = unifrnd(lambda_min,lambda_max);
    end
end
```

```

taus(1) = 1/s0;
vi(:,1) = (m2/m1)*ones(n,1);
r(1) = m1/m2;
end
% Ejecucion del algoritmo
for(it=2:nitermax)
    [TX,SX,TSX,STX,TY,SY,TSY,STY] = calculo_TS_c(Y1,X1,IT,T,IS,S,mT(it-1),gammas(it-1),mS(it-1),lambdas(it-1));
    TX = TX((n0+1):end,:);
    SX = SX((n0+1):end,:);
    TSX = TSX((n0+1):end,:);
    STX = STX((n0+1):end,:);
    TY = TY((n0+1):end);
    SY = SY((n0+1):end);
    TSY = TSY((n0+1):end);
    STY = STY((n0+1):end);
    % Distribucion condicionada de beta
    V = diag(vi(:,it-1));
    if(p>0)
        U = X-fis(1,it-1)*TX-fis(2,it-1)*SX-fis(3,it-1)*TSX-fis(4,it-1)*STX;
        if(q>0)
            U1 = Y-Z*fis(9:end,it-1)-(fis(1,it-1)+fis(5,it-1))*TY-(fis(2,it-1)+fis(6,it-1))*SY-(fis(3,it-1)+fis(7,it-1))*TSY-(fis(4,it-1)+fis(8,it-1))*STY;
        else
            U1 = Y-(fis(1,it-1)+fis(5,it-1))*TY-(fis(2,it-1)+fis(6,it-1))*SY-(fis(3,it-1)+fis(7,it-1))*TSY-(fis(4,it-1)+fis(8,it-1))*STY;
        end
        S_beta = Sigma_beta + taus(it-1)*(U'*V*U);
        S_beta = inv(S_beta);
        m_beta = S_beta*taus(it-1)*(U'*V*U1);
        betas(:,it) = mvnrnd(m_beta,S_beta);
    end
    % Distribuciones condicionadas de los fis
    fis(:,it) = fis(:,it-1);
    if(p>0)
        u1 = Y - X*betas(:,it);
    else
        u1 = Y;
    end
    U = zeros(n,q+8);
    if(p>0)
        U(:,1) = TY-TX*betas(:,it);
        U(:,2) = SY-SX*betas(:,it);
        U(:,3) = TSY-TSX*betas(:,it);
        U(:,4) = STY-STX*betas(:,it);
    end
    U(:,5) = TY;
    U(:,6) = SY;

```

Programas correspondientes al capítulo 5

```

U(:,7) = TSY;
U(:,8) = STY;
if(q>0)
    U(:,9:end) = Z;
end
if(mT(it-1)>0)
    if(mS(it-1)>0)
        indices = 1:8;
        indices = indices(indicesTS==1);
    else
        indices = [1,5];
        indices = indices(indicesTS(indices)==1);
        fis([2:4,6:8],it) = 0;
    end
end
else
    if(mS(it-1)>0)
        indices = [2,6];
        indices = indices(indicesTS(indices)==1);
        fis([1,3:5,7:8],it) = 0;
    else
        indices = [];
        fis(:,it) = 0;
    end
end
end
if(q>0)
    indices = [indices,9:(8+q)];
end
if(length(indices)>0)
    Si = taus(it-1)*(U(:,indices)*V*U(:,indices))+Sigma(indices,indices);
    Si = inv(Si);
    mi = taus(it-1)*(U(:,indices)*V*u1);
    mi = Si*mi;
    ind=1;
    contador = 0;
    while(ind>=1 & contador<=10)
        contador = contador+1;
        fis(indices,it) = mvnrnd(mi,Si);
        ind=sum(abs(fis(1:8,it))>1);
    end
    if(contador>10)
        fis(indices,it) = fis(indices,it-1);
        for(j=1:length(indices))
            i = indices(j);
            indices1 = indices(indices~=i);
            Vi = u1 - U(:,indices1)*fis(indices1,it);
            Ui = U(:,i);
            Si = taus(it-1)*(Ui*V*Ui)+Sigma(i,i);

```

```

mi = (taus(it-1)*(Ui*V*Vi));
Si = 1/Si;
mi = Si*mi;
si = sqrt(Si);
fi_inf = normcdf(-1,mi,si);
fi_sup = normcdf(1,mi,si);
u = unifrnd(0,1);
f = u*fi_inf + (1-u)*fi_sup;
if(0<f & f<1)
    fis(i,it) = norminv(f,mi,si);
elseif(f==0)
    fis(i,it) = -0.9999;
else
    fis(i,it) = 0.9999;
end
end
Vi = u1-U(:,1:8)*fis(1:8,it);
if(q>0)
    Ui = U(:,9:end);
    Si = taus(it-1)*(Ui*V*Ui)+Sigma(9:end,9:end);
    Si = inv(Si);
    mi = (taus(it-1)*(Ui*V*Vi));
    mi = Si*mi;
    fis(9:end,it) = mvnrnd(mi,Si);
end
end
end
% Distribucion condicionada de vi
error = u1-U*fis(:,it);
a = r(it-1) + taus(it-1)*error.*error;
vi(:,it) = gamrnd(0.5*(r(it-1)+1)*ones(n,1),2./a);
% Distribucion condicionada de r: utilizamos un paso de
% Hastings-Metropolis
media = (n/var(vi(:,it)) + (m1*m1)/m2)/(m1+0.5*n);
a = (m1+0.5*n)/media;
r_estrella = gamrnd(m1+0.5*n,1/a);
u = unifrnd(0,1);
alfa = log(gampdf(r(it-1),m1+0.5*n,1/a))-log(gampdf(r_estrella,m1+0.5*n,1/a));
alfa = alfa + (n*0.5*(r_estrella-1)*log(r_estrella))-0.5*n*r_estrella*log(2)-
n*gammaLn(0.5*r_estrella)+0.5*r_estrella*(sum(log(vi(:,it)))-sum(vi(:,it))-m2)+(m1+0.5*n-1)*log(r_estrella);
alfa = alfa - ((n*0.5*(r(it-1)-1)*log(r(it-1)))-0.5*n*r(it-1)*log(2)-n*gammaLn(0.5*r(it-1))+0.5*r(it-
1)*(sum(log(vi(:,it)))-sum(vi(:,it))-m2)+(m1+0.5*n-1)*log(r(it-1)));
alfa = min(exp(alfa),1);
if(u<=alfa)
    r(it) = r_estrella;
else
    r(it) = r(it-1);
end
end

```

Programas correspondientes al capítulo 5

```

% Distribucion condicionada de tau
V = diag(sqrt(vi(:,it)));
error = V*error;
a = d0*s0+sum(error.*error);
taus(it) = gamrnd(0.5*(n+d0),2/a);
% Distribucion condicionada de (mT,mS)
U = zeros(n,8+q);
if(mT(it-1)>0)
    if(p>0)
        U(:,1) = TY-TX*betas(:,it);
    end
    U(:,5) = TY;
end
if(mS(it-1)>0)
    if(p>0)
        U(:,2) = SY-SX*betas(:,it);
    end
    U(:,6) = SY;
end
if(mT(it-1)*mS(it-1)>0)
    if(p>0)
        U(:,3) = TSY-TSX*betas(:,it);
        U(:,4) = STY-STX*betas(:,it);
    end
    U(:,7) = TSY;
    U(:,8) = STY;
end
if(q>0)
    U(:,9:end) = Z;
end
error = u1-U*fis(:,it);
error = V*error;
prob = -0.5*taus(it)*sum(error.*error);
mT_estrella = mTred(unidrnd(nmT));
mS_estrella = mSred(unidrnd(nmS));
[TX,SX,TSX,STX,TY,SY,TSY,STY] = calculo_TS_c(Y1,X1,IT,T,IS,S,mT_estrella,gammas(it-1),mS_estrella,lambdas(it-1));
TX = TX((n0+1):end,:);
SX = SX((n0+1):end,:);
TSX = TSX((n0+1):end,:);
STX = STX((n0+1):end,:);
TY = TY((n0+1):end);
SY = SY((n0+1):end);
TSY = TSY((n0+1):end);
STY = STY((n0+1):end);
U(:,1:8) = 0;
if(mT_estrella>0)
    if(p>0)

```

```

        U(:,1) = TY-TX*betas(:,it);
    end
    U(:,5) = TY;
end
if(mS_estrella>0)
    if(p>0)
        U(:,2) = SY-SX*betas(:,it);
    end
    U(:,6) = SY;
end
if(mT_estrella*mS_estrella>0)
    if(p>0)
        U(:,3) = TSY-TSX*betas(:,it);
        U(:,4) = STY-STX*betas(:,it);
    end
    U(:,7) = TSY;
    U(:,8) = STY;
end
error = u1-U*fis(:,it);
error = V*error;
prob_estrella = -0.5*taus(it)*sum(error.*error);
alfa = exp(prob_estrella-prob);
u = unifrnd(0,1);
if(u<=alfa)
    mT(it) = mT_estrella;
    mS(it) = mS_estrella;
    prob = prob_estrella;
else
    mT(it) = mT(it-1);
    mS(it) = mS(it-1);
end
end
% Distribucion de (lambda,gamma)
lambda_estrella = unifrnd(lambda_min,lambda_max);
gamma_estrella = unifrnd(gamma_min,gamma_max);
[TX,SX,TSX,STX,TY,SY,TSY,STY] =
calculo_TS_c(Y1,X1,IT,T,IS,S,mT(it),gamma_estrella,mS(it),lambda_estrella);
TX = TX((n0+1):end,:);
SX = SX((n0+1):end,:);
TSX = TSX((n0+1):end,:);
STX = STX((n0+1):end,:);
TY = TY((n0+1):end);
SY = SY((n0+1):end);
TSY = TSY((n0+1):end);
STY = STY((n0+1):end);
U = zeros(n,8+q);
if(mT(it)>0)
    if(p>0)

```


Programas correspondientes al capítulo 5

```
    U(:,1) = TY-TX*betas(:,it);
end
U(:,5) = TY;
if(mS(it)>0)
    if(p>0)
        U(:,2) = SY-SX*betas(:,it);
        U(:,3) = TSY-TSX*betas(:,it);
        U(:,4) = STY-STX*betas(:,it);
    end
    U(:,6) = SY;
    U(:,7) = TSY;
    U(:,8) = STY;
end
else
    if(mS(it)>0)
        if(p>0)
            U(:,2) = SY-SX*betas(:,it);
        end
        U(:,6) = SY;
    end
end
if(q>0)
    U(:,9:end) = Z;
end
error = u1-U*fis(:,it);
error = V*error;
prob_estrella = -0.5*taus(it)*sum(error.*error);
alfa = exp(prob_estrella-prob);
alfa = min(1,alfa);
u = unifrnd(0,1);
if(u<=alfa)
    lambdas(it) = lambda_estrella;
    gammas(it) = gamma_estrella;
else
    lambdas(it) = lambdas(it-1);
    gammas(it) = gammas(it-1);
end
end
% Almacenamiento de la muestra
if(p>0)
    muestra.betas = betas;
end
if(q>0)
    muestra.alfas = alfas;
end
muestra.fis = fis;
muestra.taus = taus;
```

```

muestra.mT = mT;
muestra.gammas = gammas;
muestra.mS = mS;
muestra.lambdas = lambdas;
muestra.vi = vi;
muestra.r = r;

```

Programa 5.3.- Este programa calcula un conjunto de índices de precios tomando como base el primer periodo indicado en el vector de posiciones. Utiliza el proceso de kriging descrito en Beamonte y otros (2006) asociado al modelo hedónico descrito en Beamonte y otros (2006) y estimado bayesianamente con los programas `gibbs_pace_c_norm` y `gibbs_pace_c_student`. Para ello hay que especificarle las características `Xpred` de las viviendas que forman el conjunto de referencia y cuyos precios se quiere predecir así como su localización a lo largo del tiempo (posiciones) en el vector `Y` de log-precios que se supone ordenado por fecha de compra. Las posiciones describen la posición en el nuevo vector a construir. Así mismo hay que proporcionarle una muestra de la distribución a posteriori calculada mediante alguno de los programas anteriores. El programa devuelve los índices construidos a partir de las estimaciones de los precios de las viviendas especificadas construidas utilizando la media y la mediana a posteriori, las cuales se determinan a partir de la muestra de sus distribuciones a posteriori construidas utilizando los programas `kriging_bayesianoc` (`indsuavizado = 0`) y `kriging_bayesianoc_suavizado` (`indsuavizado=1`). El programa `kriging_bayesiano` calcula una muestra de la distribución retrospectiva utilizando la distribución a posteriori de los parámetros del modelo. El programa `kriging_bayesiano_suavizado` calcula una muestra de la distribución retrospectiva utilizando como valor del vector de parámetros del modelo su mediana a posteriori.

```

function [precios,indices_precios] =
indices_bayesianoc(posiciones,Zpred,Xpred,tpred,coordpred,Y,Z,X,tiempo,coordenadas,priori,muestra,niter0,step,n0,
ind_het,indsuavizado);
% Lectura de la distribución a priori
mTred = priori.mTred;
mSred = priori.mSred;
% Construcción de la variable de salida
q = size(Z,2);
p = size(X,2);
npred = size(Xpred,1);
npos = length(posiciones);
mT = muestra.mT;
niter = length(mT);
niter = (niter-niter0+1)/step;
ypred = zeros(niter,npred,npos);
precios_medios=zeros(npred,npos);
precios_medianos = precios_medios;
indices_precios_medianos = zeros(npred,npos);
indices_precios_medios = zeros(npred,npos);
% Calculo de los elementos de la muestra de precios

```

Programas correspondientes al capítulo 5

```

% Primero se calcula la muestra de log-precios y luego se aplica la
% transformacion exponencial
for(j=1:npos)
    for(i=1:npred);
        if(indsuavizado==1)
            if(p*q>0)
                ypred(:,i,j) =
kriging_bayesianoc_suavizado(posiciones(j),reshape(Zpred(i,:,j),1,q),reshape(Xpred(i,:,j),1,p),tpred(j),coordpred(i,:),
Y,Z,X,tiempo,coordenadas,mTred,mSred,muestra,niter0,step,n0,ind_het);
            else
                if(p>0)
                    ypred(:,i,j)=
kriging_bayesianoc_suavizado(posiciones(j),[],reshape(Xpred(i,:,j),1,p),tpred(j),coordpred(i,:),Y,[],X,tiempo,coorden
adas,mTred,mSred,muestra,niter0,step,n0,ind_het);
                else
                    ypred(:,i,j)=
kriging_bayesianoc_suavizado(posiciones(j),reshape(Zpred(i,:,j),1,q),[],tpred(j),coordpred(i,:),Y,Z,[],tiempo,coordena
das,mTred,mSred,muestra,niter0,step,n0,ind_het);
                end
            end
        else
            if(p*q>0)
                ypred(:,i,j) =
kriging_bayesianoc(posiciones(j),reshape(Zpred(i,:,j),1,q),reshape(Xpred(i,:,j),1,p),tpred(j),coordpred(i,:),Y,Z,X,tiem
po,coordenadas,mTred,mSred,muestra,niter0,step,n0,ind_het);
            else
                if(p>0)
                    ypred(:,i,j) =
kriging_bayesianoc(posiciones(j),[],reshape(Xpred(i,:,j),1,p),tpred(j),coordpred(i,:),Y,[],X,tiempo,coordenadas,mTred
,mSred,muestra,niter0,step,n0,ind_het);
                else
                    ypred(:,i,j) =
kriging_bayesianoc(posiciones(j),reshape(Zpred(i,:,j),1,q),[],tpred(j),coordpred(i,:),Y,Z,[],tiempo,coordenadas,mTred,
mSred,muestra,niter0,step,n0,ind_het);
                end
            end
        end
    end
end
precios = exp(ypred);
precios_medianos = reshape(median(precios),npred,npos);
indices_precios = sum(precios_medianos);
indices_precios = 100*indices_precios/indices_precios(1);

```

Programa 5.4.- Este programa implementa el proceso de kriging descrito en Beamonte y otros (2006) asociado al modelo hedonico descrito en Beamonte y otros (2006) y estimado bayesianamente con los programas gibbs_pace_norm y gibbs_pace_c_student Para ello hay que especificarle las características Xpred de la vivienda que se quiere predecir así como su localización a lo largo del tiempo (posición) en el vector Y de log-precios que se supone

ordenado por fecha de compra Las posiciones describen la posición en el nuevo vector a construir Así mismo hay que proporcionarle una muestra de la distribución a posteriori calculada mediante alguno de los programas anteriores y un indicador de si es homo (ind_het==0) o heterocedastico (ind_het==1) Zpred y Xpred contienen los vector de covariables de la vivienda cuyo precio se quiere estimar retrospectivamente tpred contiene el tiempo en el que se produjo la transacción coordpred contiene las coordenadas de la vivienda ind_student = 1 si la muestra se ha obtenido con gibbs_pace_c_student y 0 si se ha obtenido con gibbs_pace_norm niter0 contiene el inicio del periodo de toma de la muestra step contiene el número de pasos para reducir el valor de la autocorrelación de la muestra a valores poco significativos El programa devuelve una muestra de la distribución a posteriori de log-precios para la vivienda

```
function ypred =
kriging_bayesianoc(posicion,Zpred,Xpred,tpred,coordpred,Y,Z,X,tiempo,coordenadas,mTred,mSred,muestra,niter0,step,n0,ind_student);
% Insercion de las viviendas a predecir entre las viviendas de la muestra
p = size(X,2);
q = size(Z,2);
Y = [Y(1:(posicion-1));0;Y(posicion:end)];
if(p>0)
    X = [X(1:(posicion-1,:);Xpred;X(posicion:end,:)];
end
if(q>0)
    Z = [Z(1:(posicion-1,:);Zpred;Z(posicion:end,:)];
end
tiempo = [tiempo(1:(posicion-1));tpred;tiempo(posicion:end)];
coordenadas = [coordenadas(1:(posicion-1,:);coordpred;coordenadas(posicion:end,:)];
[IT,T] = matriz_proximidad_espacial(tiempo,tiempo,max(mTred));
IT = IT>0;
[IS,S] = matriz_proximidad_espacial(tiempo,coordenadas,max(mSred));
IS = IS>0;
n = length(Y);
% Extraccion de la muestra
if(p>0)
    betas = muestra.betas;
end
if(q>0)
    alfas = muestra.alfas;
end
fis = muestra.fis;
taus = muestra.taus;
mT = muestra.mT;
gammas = muestra.gammas;
mS = muestra.mS;
lambdas = muestra.lambdas;
niter = size(fis,2);
if(p>0)
```

Programas correspondientes al capítulo 5

```
    betas = betas(:,niter0:step:end);
end
if(q>0)
    alfas = alfas(:,niter0:step:end);
end
fis = fis(:,niter0:step:end);
taus = taus(niter0:step:end);
mT = mT(niter0:step:end);
gammas = gammas(niter0:step:end);
mS = mS(niter0:step:end);
lambdas = lambdas(niter0:step:end);
j = posicion;
vi = zeros(n-j+1,niter);
if(ind_student==0)
    vi = ones(n-j+1,niter);
else
    vi1 = muestra.vi;
    r=muestra.r;
    if(j<=n0)
        for(k=j:n0)
            vi(k-j+1,:) = gamrnd(0.5*r,2./r,niter,1)';
        end
        vi((n0-j+2):end,:) = vi1;
    else
        vi(1,:) = gamrnd(0.5*r,2./r,niter,1)';
        vi(2:end,:) = vi1((j-n0):end,:);
    end
end
niter = size(fis,2);
vi = vi(:,niter0:step:end);
% Iniciacion de las variables
media = 0;
media2 = 0;
varianza = media;
ypred = zeros(niter,1);
indicesn = (j+1):n;
n1 = length(indicesn);
% Algoritmo que imprementa el proceso de kriging
for(it=1:niter)
    % if(rem(it,10)==1)
    % it
    % end
    muj = 0;
    if(q>0)
        muj = muj+Z(j,:)*alfas(:,it);
    end
    if(p>0)
```

```

    muj = muj + X(j,:)*betas(:,it);
end
sj = vi(1,it);
[TX,SX,TSX,STX,TY,SY,TSY,STY] = calculo_TS_c(Y,X,IT,T,IS,S,mT(it),gammas(it),mS(it),lambdas(it));
if(mS(it)>0)
    lambdas_mS = lambdas(it).^(1:mS(it));
    if(p>0)
        muj = muj + fis(2,it)*(SY(j)-SX(j,:)*betas(:,it));
    end
    muj = muj + fis(6,it)*SY(j);
end
if(mT(it)>0)
    gammas_mT = gammas(it).^(1:mT(it));
    if(p>0)
        muj = muj + fis(1,it)*(TY(j)-TX(j,:)*betas(:,it));
    end
    muj = muj + fis(5,it)*TY(j);
end
if(mS(it)*mT(it)>0)
    if(p>0)
        muj = muj + fis(3,it)*(TSY(j)-TSX(j,:)*betas(:,it))+fis(4,it)*(STY(j)-STX(j,:)*betas(:,it));
    end
    muj = muj + fis(7,it)*TSY(j)+fis(8,it)*STY(j);
end
muj = sj*muj;
% Calculo de las transacciones conectadas con la actual asi como de sus coeficientes fij asociados
% En cada caso hay que tener en cuenta que los coeficientes asociados a
% lambda y gamma pueden cambiar en funcion del numero de vecinos reales
% de cada transaccion
% Dichos vecinos los identifican las matrices IT e IS
% Comenzamos por los vecinos asociados con T
fij = zeros(n1,1);
if(mT(it)>0)
    jinf = min(j+1,n);
    jsup = min(j+mT(it),n);
    aux1 = jinf:jsup;
    aux1 = aux1(diag(IT(aux1,aux1-j))>0);
    for(k=1:length(aux1))
        gammas_mT1 = gammas_mT(1:sum(IT(aux1(k),1:mT(it))));
        gammas_mT1 = gammas_mT(k)/sum(gammas_mT1);
        fij(aux1-j) = (fis(1,it)+fis(5,it))*gammas_mT1;
    end
    % Vecinos asociados con ST
    for(k2 = 1:mS(it))
        j1 = IS(aux1(k),k2)*S(aux1(k),k2)-j;
        if(j1>0)
            lambdas_mS1 = lambdas_mS(1:sum(IS(aux1(k),1:mS(it))));
            lambdas_mS1 = lambdas_mS(k2)/sum(lambdas_mS1);
        end
    end
end

```

Programas correspondientes al capítulo 5

```

        fij(j1) = fij(j1) + (fis(4,it)+fis(8,it))*lambdas_mS1*gammas_mT1;
    end
end
end
end
% Vecinos espaciales
for(k=1:mS(it))
    indicesS = indicesn(S(indicesn,k)==j & IS(indicesn,k)==1);
    if(length(indicesS)>0)
        for(k2=1:length(indicesS))
            lambdas_mS1 = lambdas_mS(1:sum(IS(indicesS(k2),1:mS(it))));
            lambdas_mS1 = lambdas_mS(k)/sum(lambdas_mS1);
            fij(indicesS(k2)-j) = fij(indicesS(k2)-j) + (fis(2,it)+fis(6,it))*lambdas_mS1;
            % Vecinos asociados con TS
            for(k1=1:mT(it))
                aux = indicesS(k2)+k1;
                if(aux<=n & IT(aux,k1)==1)
                    gammas_mT1 = gammas_mT(1:sum(IT(aux,1:mT(it))));
                    gammas_mT1 = gammas_mT(k1)/sum(gammas_mT1);
                    fij(aux-j) = fij(aux-j) + (fis(3,it)+fis(7,it))*lambdas_mS1*gammas_mT1;
                end
            end
        end
    end
end
end
% Una vez localizados los vecinos de la transaccion i-esima se pasa a
% calcular la media y la varianza
V = diag(vi(2:end,it));
sj = sj+fij'*V*fij;
indicesn2 = indicesn(fij~=0);
if(length(indicesn2)>0)
    muj1 = 0;
    if(p>0)
        muj1 = muj1 + X(indicesn2,:)*betas(:,it);
    end
    if(q>0)
        muj1 = muj1 +Z(indicesn2,:)*alfas(:,it);
    end
    if(mS(it)>0)
        if(p>0)
            muj1 = muj1 + fis(2,it)*(SY(indicesn2)-SX(indicesn2,:)*betas(:,it));
        end
        muj1 = muj1 + +fis(6,it)*SY(indicesn2);
    end
    if(mT(it)>0)
        if(p>0)
            muj1 = muj1 + fis(1,it)*(TY(indicesn2)-TX(indicesn2,:)*betas(:,it));
        end
    end
end

```

```

end
muj1 = muj1 + +fis(5,it)*TY(indicesn2);
end
if(mS(it)*mT(it)>0)
if(p>0)
muj1 = muj1 + fis(3,it)*(TSY(indicesn2)-TSX(indicesn2,:)*betas(:,it));
muj1 = muj1 + fis(4,it)*(STY(indicesn2)-STX(indicesn2,:)*betas(:,it));
end
muj1 = muj1 + fis(7,it)*TSY(indicesn2)+fis(8,it)*STY(indicesn2);
end
muj = muj + fij(indicesn2-j)*diag(vi(indicesn2-j,it))*(Y(indicesn2)-muj1);
end
varianza1 = 1/sj;
media1 = varianza1*muj;
varianza1 = varianza1/taus(it);
media = media+media1;
media2 = media2+media1*media1;
varianza = varianza + varianza1;
desv1 = sqrt(varianza1);
ypred(it) = normrnd(media1,desv1);
end
media = media/niter;
varianza = varianza/niter + media2/niter-media*media;
desv = sqrt(varianza);
ypred = ypred';

```

Programas correspondientes al capítulo 6

Programa 6.1.- rograma principal que implementa el algoritmo genetico descrito en el Capítulo 6 de la Memoria

```

% Iniciacion de la poblacion
mogahouse;
% Evaluacion de los criterios
evalua;
i1=1;
contador = 0;
tamano_anterior = 0;
while((contador<10)&(i1<=sistema.numeroIteraciones))
disp(['Iteracion :' num2str(i1)]);
% Pasos de recombinacion, mutacion, evaluacion, ranking, eliminacion de
% repetidos y seleccion elitista del algoritmo
recombinacion;
mutacion;
evalua;
ranking_de_poblacion;

```


Programas correspondientes al capítulo 6

```

eliminacionrepetidos;
seleccionElitista;
% Criterio de convergencia
tamano = sistema.tamagnoPoblacion
tamano1 = length(sistema.ranking(sistema.ranking==1))
if(tamano1==tamano_anterior)
    contador = contador+1;
else
    contador = 0;
    tamano_anterior = tamano1;
end
ranking = max(sistema.ranking)
i1=i1+1;
end
clear i1 tamano tamano1 ranking contador tamano_anterior;

```

Programa 6.2.- Esta funcion calcula el valor de los criterios de seleccion para un modelo de Pace y otros (2000) teniendo en cuenta las variables y componentes seleccionados El modelo que utiliza es $Y = Z*\alpha + X*\beta + \text{fis}(1)*\text{indTX}*T*\text{error} + \text{fis}(2)*\text{indSX}*S*\text{error} + \text{fis}(3)*\text{indTSX}*T*S*\text{error} + \text{fis}(4)*\text{indSTX}*S*T*\text{error} + \text{fis}(5)*\text{indTY}*T*Y + \text{fis}(6)*\text{indSY}*S*Y + \text{fis}(7)*\text{indTSY}*T*S*Y + \text{fis}(8)*\text{indST}*S*T*Y$ donde $\text{error} = Y - X*\beta$ siendo Y un vector $n \times 1$, Z una matriz $n \times q$ y X una matriz $n \times p$ Las variables indices Z e indices X contienen los indices de las variables de Z y X contenidas en el modelo La variable indices $TS = [\text{indTX}, \text{indSX}, \text{indTSX}, \text{indSTX}, \text{indTY}, \text{indSY}, \text{indTSY}, \text{indST}]$ donde dichas componentes valen 0 o 1 segun no contenga o contenga la componente espacio-temporal correspondiente Los arrays TX ($n \times p \times n_{\text{Tred}}$), SX ($n \times p \times n_{\text{Sred}} \times n_{\text{lambda}}$), TSX ($n \times p \times n_{\text{Sred}} \times n_{\text{lambda}} \times n_{\text{Tred}}$), STX ($n \times p \times n_{\text{Tred}} \times n_{\text{Sred}} \times n_{\text{lambda}}$) TY ($n \times n_{\text{Tred}}$), SY ($n \times n_{\text{Sred}} \times n_{\text{lambda}}$), TSY ($n \times n_{\text{Sred}} \times n_{\text{lambda}} \times n_{\text{Tred}}$) y STY ($n \times n_{\text{Tred}} \times n_{\text{Sred}} \times n_{\text{lambda}}$) contienen las matrices TX , SX , TSX , STX , TY , SY , TSY y STY para una combinacion dada de los parametros mT , mS y λ Estos arrays se calculan con el programa calculo_TS indmT, indmS, indlambda contienen los indices correspondientes a los lugares de los valores mT , mS y λ en los vectores $mTred$, $mSred$ y $lambdared$ que contienen los posibles valores de dichos parametros Finalmente, $n0$ contiene el numero de observaciones iniciales que se descartan para debilitar el efecto de inicializacion del proceso El programa utiliza un proceso de estimacion OLS en dos pasos Devuelve el valor de los criterio BIC, LOSS Y LPRED

```

function [BIC,SPL,LPRED,coeficientes,std] =
criterios_seleccion(Y,Z,X,TX,SX,STX,TSX,TY,SY,STY,TSY,indicesZ,indicesX,indicesTS,indmT,indgamma,indmS,
indlambda);
global mTred;
global mSred;
global n0;
global nest;
% Creacion de las variables auxiliares
n = length(Y);
pa = size(X,2);

```

```

qa = size(Z,2);
U = [];
q = length(indicesZ);
if(q>0)
    Z = Z(:,indicesZ);
    U = [U,Z];
end
p = length(indicesX);
if(p>0)
    X = X(:,indicesX);
    U = [U,X];
end
if(p==0)
    indicesTS(1:4) = 0;
end
if(mTred(indmT)==0)
    indicesTS([1,3,4,5,7,8]) = 0;
end
if(mSred(indmS)==0)
    indicesTS([2,3,4,6,7,8]) = 0;
end
indices_fis = indicesTS.*[1:8];
indices_fis = indices_fis(indices_fis>0);
nfis = length(indices_fis);
if(nfis>0)
    if(p>0)
        TX = reshape(TX(:,indicesX,indmT,indgamma),n,p);
        SX = reshape(SX(:,indicesX,indmS,indlambd),n,p);
        TSX = reshape(TSX(:,indicesX,indmT,indgamma,indmS,indlambd),n,p);
        STX = reshape(STX(:,indicesX,indmT,indgamma,indmS,indlambd),n,p);
        U = [U,TX,SX,TSX,STX];
    end
    TY = TY(:,indmT,indgamma);
    SY = SY(:,indmS,indlambd);
    TSY = TSY(:,indmT,indgamma,indmS,indlambd);
    STY = STY(:,indmT,indgamma,indmS,indlambd);
    U = [U,TY,SY,TSY,STY];
end
% Calculo del estimador MLE: utilizamos la funcion nlinfit
X1 = [];
if(q>0)
    X1 = [X1,Z];
end
if(p>0)
    X1 = [X1,X];
end
if(p+q>0)

```

Programas correspondientes al capítulo 6

```

    betas0 = inv(X1'*X1)*(X1'*Y);
else
    betas0 = 0;
end
betas0 = [betas0;zeros(nfis,1)];
U = [nfis*ones(n,1),U];
U1 = U((n0+1):end,:);
U1(1,1) = q;
U1(2,1) = p;
U1(3,1) = nfis;
U1(4:(3+nfis),1) = indices_fis;
[betas,res,J] = nlinfit(U1,Y((n0+1):end),@f_pace1,betas0);
% Calculo de los criterios BIC y FIT
sigma_mle = mean(res.*res);
BIC = (n-n0)*log(sigma_mle) + (p+q+nfis)*log(n-n0);
FIT = (n-n0)*sigma_mle;
% Calculo del criterio PARSIMONIA
if(q>0)
    alfas = betas(1:q);
end
fis = zeros(8,1);
if(nfis>0)
    fis(indices_fis) = betas((q+p+1):end);
end
if(p>0)
    betas = betas((q+1):(q+p));
end
coeficientes = zeros(pa+qa+9,1);
if(q>0)
    coeficientes(indicesZ) = alfas;
end
if(p>0)
    coeficientes(qa+indicesX) = betas;
end
if(nfis>0)
    coeficientes(pa+qa+indices_fis) = fis(indices_fis);
end
coeficientes(pa+qa+9) = sigma_mle;
U1 = [];
if(q>0)
    U1 = [U1,Z((n0+1):end,:);];
end
if(p>0)
    U2 = X((n0+1):end,:);
if(nfis>0)
    U2 = U2 -fis(1)*TX((n0+1):end,:)-fis(2)*SX((n0+1):end,:)-fis(3)*TSX((n0+1):end,:)-fis(4)*STX((n0+1):end,:);
end
end

```

```

U1 = [U1,U2];
end
if(nfis>0)
    U2 = zeros(n-n0,4);
    if(p>0)
        U2(:,1) = TY((n0+1):end)-TX((n0+1):end,:)*betas;
        U2(:,2) = SY((n0+1):end)-SX((n0+1):end,:)*betas;
        U2(:,3) = TSY((n0+1):end)-TSX((n0+1):end,:)*betas;
        U2(:,4) = STY((n0+1):end)-STX((n0+1):end,:)*betas;
    end
    U2 = [U2,TY((n0+1):end),SY((n0+1):end),TSY((n0+1):end),STY((n0+1):end)];
    U1 = [U1,U2(:,indices_fis)];
end
nval = n0+1;
for(j=1:(n-nval+1))
    res(nval-n0-1+j) = Y(nval+j-1);
    if(q>0)
        res(nval-n0-1+j) = res(nval-n0-1+j)-Z(nval+j-1,:)*alfas;
    end
    if(p>0)
        res(nval-n0-1+j) = res(nval-n0-1+j)- X(nval+j-1,:)*betas;
    end
    if(nfis>0)
        res(nval-n0-1+j) = res(nval-n0-1+j)-fis(1)*(TY(nval-n0-1+j)-TX(nval-n0-1+j,:)*betas);
        res(nval-n0-1+j) = res(nval-n0-1+j)-fis(2)*(SY(nval-n0-1+j)-SX(nval-n0-1+j,:)*betas);
        res(nval-n0-1+j) = res(nval-n0-1+j)-fis(3)*(TSY(nval-n0-1+j)-TSX(nval-n0-1+j,:)*betas);
        res(nval-n0-1+j) = res(nval-n0-1+j)-fis(4)*(STY(nval-n0-1+j)-STX(nval-n0-1+j,:)*betas);
    end
end
if(nfis>0)
    res(nval-n0-1+j) = res(nval-n0-1+j)-fis(5)*TY(nval-n0-1+j);
    res(nval-n0-1+j) = res(nval-n0-1+j)-fis(6)*SY(nval-n0-1+j);
    res(nval-n0-1+j) = res(nval-n0-1+j)-fis(7)*TSY(nval-n0-1+j);
    res(nval-n0-1+j) = res(nval-n0-1+j)-fis(8)*STY(nval-n0-1+j);
end
end
for(i=1:size(U1,2))
    U1(:,i) = res.*U1(:,i);
end
cov_betas = (sigma_mle*sigma_mle)*inv(U1'*U1);
std1 = sqrt(diag(cov_betas));
std = zeros(pa+qa+9,1);
if(q>0)
    std(indicesZ) = std1(1:q);
end
if(p>0)
    std(qa+indicesX) = std1((q+1):(q+p));
end
end

```

Programas correspondientes al capítulo 6

```

if(nfis>0)
    std(pa+qa+indices_fis) = std1((p+q+1):end);
end
std(pa+qa+9) = 2*(sigma_mle*sigma_mle)/(n-n0);
PARSIMONIA = 0;
if(nfis+p+q>0)
    U2 = U1((nval-n0):end,:);
    for(i=nval:n)
        aux = U2(i-nval+1,:);
        PARSIMONIA = PARSIMONIA + aux*cov_betas*aux';
    end
end
SPL = FIT+PARSIMONIA;
% Calculo del criterio LPRED
% Calculo del estimador MLE en el periodo de estimacion: utilizamos un procedimiento de dos pasos
X1 = [];
if(q>0)
    X1 = [X1,Z(1:nest,:)];
end
if(p>0)
    X1 = [X1,X(1:nest,:)];
end
if(p+q>0)
    betas0 = inv(X1'*X1)*(X1'*Y(1:nest));
else
    betas0 = 0;
end
betas0 = [betas0;zeros(nfis,1)];
U1 = U((n0+1):nest,:);
U1(1,1) = q;
U1(2,1) = p;
U1(3,1) = nfis;
U1(4:(3+nfis),1) = indices_fis;
[betas,res,J] = nlinfit(U1,Y((n0+1):nest),@f_pace1,betas0);
sigma_mle = mean(res.*res);
Y1 = Y((n0+1):nest);
n = length(Y1);
%
% Estimacion de la matriz de varianzas y covarianzas de los parametros del
% modelo
%
if(q>0)
    alfas = betas(1:q);
end
fis = zeros(8,1);
if(nfis>0)
    fis(indices_fis) = betas((q+p+1):end);

```

```

end
if(p>0)
    betas = betas((q+1):(q+p));
end
U1 = [];
if(q>0)
    U1 = [U1,Z((n0+1):nest,:)];
end
if(p>0)
    U2 = X((n0+1):nest,:);
    if(nfis>0)
        U2 = U2 -fis(1)*TX((n0+1):nest,:)-fis(2)*SX((n0+1):nest,:)-fis(3)*TSX((n0+1):nest,:)-
fis(4)*STX((n0+1):nest,:);
    end
    U1 = [U1,U2];
end
if(nfis>0)
    U2 =zeros(nest-n0,4);
    if(p>0)
        U2(:,1) = TY((n0+1):nest)-TX((n0+1):nest,:)*betas;
        U2(:,2) = SY((n0+1):nest)-SX((n0+1):nest,:)*betas;
        U2(:,3) = TSY((n0+1):nest)-TSX((n0+1):nest,:)*betas;
        U2(:,4) = STY((n0+1):nest)-STX((n0+1):nest,:)*betas;
    end
    U2 = [U2,TY((n0+1):nest),SY((n0+1):nest),TSY((n0+1):nest),STY((n0+1):nest)];
    U1 = [U1,U2(:,indices_fis)];
end
for(j=1:(n-nval+1))
    res(nval-n0-1+j) = Y(nval+j-1);
    if(q>0)
        res(nval-n0-1+j) = res(nval-n0-1+j)-Z(nval+j-1,:)*alfas;
    end
    if(p>0)
        res(nval-n0-1+j) = res(nval-n0-1+j)- X(nval+j-1,:)*betas;
    end
    if(nfis>0)
        res(nval-n0-1+j) = res(nval-n0-1+j)-fis(1)*(TY(nval-n0-1+j)-TX(nval-n0-1+j,:))*betas;
        res(nval-n0-1+j) = res(nval-n0-1+j)-fis(2)*(SY(nval-n0-1+j)-SX(nval-n0-1+j,:))*betas;
        res(nval-n0-1+j) = res(nval-n0-1+j)-fis(3)*(TSY(nval-n0-1+j)-TSX(nval-n0-1+j,:))*betas;
        res(nval-n0-1+j) = res(nval-n0-1+j)-fis(4)*(STY(nval-n0-1+j)-STX(nval-n0-1+j,:))*betas;
    end
end
if(nfis>0)
    res(nval-n0-1+j) = res(nval-n0-1+j)-fis(5)*TY(nval-n0-1+j);
    res(nval-n0-1+j) = res(nval-n0-1+j)-fis(6)*SY(nval-n0-1+j);
    res(nval-n0-1+j) = res(nval-n0-1+j)-fis(7)*TSY(nval-n0-1+j);
    res(nval-n0-1+j) = res(nval-n0-1+j)-fis(8)*STY(nval-n0-1+j);
end

```

Programas correspondientes al capítulo 6

```

end
for(i=1:size(U1,2))
    U1(:,i) = res.*U1(:,i);
end
U = U1;
cov_betas = (sigma_mle*sigma_mle)*inv(U'*U);
% Calculo de las predicciones extramuestrales
Y1 = Y((nest+1):end);
n = length(Y1);
ypred = zeros(n,1);
nval = nest+1;
if(q>0)
    ypred = ypred + Z(nval:end,:)*alfas;
end
if(p>0)
    ypred = ypred + X(nval:end,:)*betas;
    if(nfis>0)
        ypred = ypred + fis(1)*(TY(nval:end)-TX(nval:end,:))*betas;
        ypred = ypred + fis(2)*(SY(nval:end)-SX(nval:end,:))*betas;
        ypred = ypred + fis(3)*(TSY(nval:end)-TSX(nval:end,:))*betas;
        ypred = ypred + fis(4)*(STY(nval:end)-STX(nval:end,:))*betas;
    end
end
if(nfis>0)
    ypred = ypred + fis(5)*TY(nval:end);
    ypred = ypred + fis(6)*SY(nval:end);
    ypred = ypred + fis(7)*TSY(nval:end);
    ypred = ypred + fis(8)*STY(nval:end);
end
if(nfis+p+q>0)
    U1 = [];
    if(q>0)
        U1 = [U1,Z(nval:end,:)];
    end
    if(p>0)
        U2 = X(nval:end,:);
        if(nfis>0)
            U2 = U2 -fis(1)*TX(nval:end,:)-fis(2)*SX(nval:end,:)-fis(3)*TSX(nval:end,:)-fis(4)*STX(nval:end,:);
        end
        U1 = [U1,U2];
    end
    if(nfis>0)
        U2 = zeros(n,4);
        if(p>0)
            U2(:,1) = TY(nval:end)-TX(nval:end,:)*betas;
            U2(:,2) = SY(nval:end)-SX(nval:end,:)*betas;
            U2(:,3) = TSY(nval:end)-TSX(nval:end,:)*betas;

```

```

        U2(:,4) = STY(nval:end)-STX(nval:end,:)*betas;
    end
    U2 = [U2,TY(nval:end),SY(nval:end),TSY(nval:end),STY(nval:end)];
    U1 = [U1,U2(:,indices_fis)];
end
LPRED = 0;
for(i=nval:length(Y))
    aux = U(i-nval+1,:);
    sigma = sigma_mle*(1+aux*cov_betas*aux');
    LPRED = LPRED + log(normpdf(Y(i),ypred(i-nval+1),sqrt(sigma)));
end
else
    LPRED = sum(log(normpdf(Y1,ypred,sqrt(sigma_mle))));
end
LPRED = -LPRED;

```

Programa 6.3.- Este programa elimina los elementos repetidos de la poblacion seleccionada por el programa geneticoBEA Para ello asigna un numero a cada elemento de la poblacion calculado a partir de la representacion binaria del mismo, y se queda con los numeros no repetidos

```

global sistema;
% Proceso de asignacion de numeros
poblacion = sistema.Poblacion;
VFO = sistema.VFO;
ranking = sistema.ranking;
coeficientes = sistema.coeficientes;
std = sistema.std;
N = size(poblacion,2);
p = size(poblacion,1);
bases = 2.^[0:(p-1)];
numeros = bases*poblacion;
% Ordenamos los numeros antes de proceder al proceso de eliminacion de los
% repetidos
[numeros,indices] = sort(numeros);
poblacion = poblacion(:,indices);
VFO = VFO(:,indices);
coeficientes = coeficientes(:,indices);
std = std(:,indices);
ranking = ranking(indices);
% Eliminacion de elementos repetidos
lista_final = poblacion(:,1);
VF_final = VFO(:,1);
ranking_final = ranking(1);
coeficientes_final = coeficientes(:,1);

```


Programas correspondientes al capítulo 6

```
std_final = std(:,1);
lugar = 1;
for(i=2:N)
    if( numeros(i) > numeros(lugar) )
        lugar = i;
        lista_final = [lista_final, poblacion(:,i)];
        VF_final = [VF_final, VFO(:,i)];
        ranking_final = [ranking_final, ranking(i)];
        coeficientes_final = [coeficientes_final, coeficientes(:,i)];
        std_final = [std_final, std(:,i)];
    end
end
end
% Reasignacion del sistema
sistema.VFO = VF_final;
sistema.Poblacion = lista_final;
sistema.ranking = ranking_final;
sistema.tamagnoPoblacionExt = length(ranking_final);
sistema.tamagnoPoblacion = length(ranking_final);
sistema.coeficientes = coeficientes_final;
sistema.std = std_final;
clear poblacion p N bases numeros indices VFO ranking lista_final VF_final ranking_final lugar i;
clear std coeficientes coeficientes_final std_final;
```

Programa 6.4.- Version de evaluacion que sistematicamente evalua desde el individuo sistema.tamagnoPoblacion+1 hasta sistema.tamagnoPoblacionExt (ambos incluidos) (si tamagnoPoblacion y tamagnoPoblacionExt son distintos y desde 1 hasta sistema.tamagnoPoblacion si son iguales. Esta funcion es la encargada de la evaluacion de las F.O. posteriormente habra que llamar a la que construya y asigne el fitness

```
function evaluacion()
global sistema;
global X;
global Y;
global Z;
global SX;
global SY;
global TX;
global TY;
global STX;
global STY;
global TSX;
global TSY;
global mTred;
global mSred;
global gammared;
```

```

global lambdared;
global n0;
global nest;

if (sistema.tamagnoPoblacion==sistema.tamagnoPoblacionExt)
    desde =1;
    hasta = sistema.tamagnoPoblacion;
    sistema.VFO = zeros(sistema.Q,sistema.tamagnoPoblacion);
else
    desde = sistema.tamagnoPoblacion+1;
    hasta = sistema.tamagnoPoblacionExt;
    sistema.VFO = [sistema.VFO, zeros(sistema.Q,hasta-desde+1)];
end
% Calculo de los criterios para cada elemento nuevo de la poblacion
for i=desde:hasta
    sistema.Poblacion(23:26,i) = 0;
    mT = mTred(bin2dec(num2str(sistema.Poblacion(27:29,i)))+1);
    if(mT==0)
        sistema.Poblacion(30:31,i)=0;
        sistema.Poblacion([19,21,22],i) = 0;
    end
    mS = mSred(bin2dec(num2str(sistema.Poblacion(32:34,i)))+1);
    if(mS==0)
        sistema.Poblacion(35:36,i) = 0;
        sistema.Poblacion(20:22,i) = 0;
    end
    if(sum(sistema.Poblacion([19,21,22],i))== 0)
        sistema.Poblacion(27:31,i) = 0;
    end
    if(sum(sistema.Poblacion(20:22,i)) == 0)
        sistema.Poblacion(32:36,i)=0;
    end
    end
    [a b c coeficientes std]= ...
    criterios_seleccion(Y,Z,X,TX,SX,STX,TSX,TY,SY,STY,TSY,find(sistema.Poblacion(1:4,i)),...
    find(sistema.Poblacion(5:18,i)),...
    sistema.Poblacion(19:26,i),...
    bin2dec(num2str(sistema.Poblacion(27:29,i)))+1,...
    bin2dec(num2str(sistema.Poblacion(30:31,i)))+1,...
    bin2dec(num2str(sistema.Poblacion(32:34,i)))+1,...
    bin2dec(num2str(sistema.Poblacion(35:36,i)))+1);
    mT = mTred(bin2dec(num2str(sistema.Poblacion(27:29,i)))+1);
    mS = mSred(bin2dec(num2str(sistema.Poblacion(32:34,i)))+1);
    gamma = gammared(bin2dec(num2str(sistema.Poblacion(30:31,i)))+1)*(mT>0);
    lambda = lambdared(bin2dec(num2str(sistema.Poblacion(35:36,i)))+1)*(mS>0);
    sistema.coeficientes = [sistema.coeficientes,[coeficientes;mT;gamma;mS;lambda]];
    sistema.std = [sistema.std,[std;zeros(4,1)]];
    sistema.VFO(1,i)=a;

```

Programas correspondientes al capítulo 6

```
sistema.VFO(2,i)=b;
sistema.VFO(3,i)=c;
end
clear a b c i desde hasta coeficientes std mT gamma mS lambda;
```

Programa 6.5.- Esta función calcula predicciones unipuntuales e intervalos predictivos para un periodo t utilizando el modelo de Pace y otros (2000) teniendo en cuenta las variables y componentes seleccionados. El modelo que utiliza es $Y = X*\beta + \text{fis}(1)*\text{indTX}*T*\text{error} + \text{fis}(2)*\text{indSX}*S*\text{error} + \text{fis}(3)*\text{indTSX}*T*S*\text{error} + \text{fis}(4)*\text{indSTX}*S*T*\text{error} + \text{fis}(5)*\text{indTY}*T*Y + \text{fis}(6)*\text{indSY}*S*Y + \text{fis}(8)*\text{indTSY}*T*S*Y + \text{fis}(8)*\text{indST}*S*T*Y$ donde $\text{error} = Y - X*\beta$; Las variables indicadoras valen 1 si se contiene la componente correspondiente y 0 en otro caso. Devuelve la media y la varianza de la distribución predictiva así como una muestra de la predicción.

```
function [media,varianza] = f_mle_prediccion_genetico(nest,nval,coeficientes,t);
global X;
global Y;
global Z;
global SX;
global SY;
global TX;
global TY;
global STX;
global STY;
global TSX;
global TSY;
global mTred;
global mSred;
global gammared;
global lambdared;
global n0;
% Lectura de los parametros
q = size(Z,2);
p = size(X,2);
n = length(Y);
if(q>0)
    alfas = coeficientes(1:q);
end
if(p>0)
    betas = coeficientes((q+1):(q+p));
end
fis = coeficientes((q+p+1):(q+p+8));
sigma = coeficientes(p+q+9);
mT = coeficientes(p+q+10);
gamma = coeficientes(p+q+11);
```

```

mS = coeficientes(p+q+12);
lambda = coeficientes(p+q+13);
% Obtencion de las matrices auxiliares
indices = 1:length(mTred);
indmT = indices(mTred == mT);
if(mT>0)
    indices = 1:length(gammared);
    indgamma = indices(gammared == gamma);
else
    indgamma = 1;
end
indices = 1:length(mSred);
indmS = indices(mSred == mS);
if(mS>0)
    indices = 1:length(lambdared);
    indlambda = indices(lambdared == lambda);
else
    indlambda=1;
end
TX1 = reshape(TX(:, :, indmT, indgamma), n, p);
SX1 = reshape(SX(:, :, indmS, indlambda), n, p);
TSX1 = reshape(TSX(:, :, indmS, indlambda, indmT, indgamma), n, p);
STX1 = reshape(STX(:, :, indmT, indgamma, indmS, indlambda), n, p);
TY1 = TY(:, indmT, indgamma);
SY1 = SY(:, indmS, indlambda);
TSY1 = TSY(:, indmS, indlambda, indmT, indgamma);
STY1 = STY(:, indmT, indgamma, indmS, indlambda);
% Calculo de los errores estandar
res = Y;
if(q>0)
    res = res-Z*alfas;
end
if(p>0)
    res = res-X*betas;
    res = res - fis(1)*(TY1-TX1*betas);
    res = res-fis(2)*(SY1-SX1*betas);
    res = res-fis(3)*(TSY1-TSX1*betas);
    res = res-fis(4)*(STY1-STX1*betas);
end
res = res - fis(5)*TY1;
res = res - fis(6)*SY1;
res = res - fis(7)*TSY1;
res = res - fis(8)*STY1;
U1 = [];
if(p>0)
    U1 = [U1,X((n0+1):end,:)-fis(1)*TX1((n0+1):end,:)-fis(2)*SX1((n0+1):end,:)-fis(3)*TSX1((n0+1):end,:)-
fis(4)*STX1((n0+1):end,:)];

```

Programas correspondientes al capítulo 6

```
end
U2 = [TY1((n0+1):end),SY1((n0+1):end),TSY1((n0+1):end),STY1((n0+1):end)];
if(p>0)
    U2 = [STY1((n0+1):end)-STX1((n0+1):end,:)*betas,U2];
    U2 = [TSY1((n0+1):end)-TSX1((n0+1):end,:)*betas,U2];
    U2 = [SY1((n0+1):end)-SX1((n0+1):end,:)*betas,U2];
    U2 = [TY1((n0+1):end)-TX1((n0+1):end,:)*betas,U2];
end
indicesfis = 1:8;
indicesfis = indicesfis(fis~=0);
U1 = [U1,U2(:,indicesfis)];
U = U1(1:(nest-n0),:);
res = res((n0+1):nest);
for(i=1:size(U1,2))
    U(:,i) = res.*U(:,i);
end
cov_betas = (sigma*sigma)*inv(U'*U);
% Calculo de la media y varianza de la distribucion predictiva
media = 0;
if(q>0)
    media = media + Z(t,:)*alfas;
end
if(p>0)
    media = media + X(t,:)*betas;
    media = media + fis(1)*(TY1(t)-TX1(t,:)*betas);
    media = media + fis(2)*(SY1(t)-SX1(t,:)*betas);
    media = media +fis(3)*(TSY1(t)-TSX1(t,:)*betas);
    media = media +fis(4)*(STY1(t)-STX1(t,:)*betas);
end
media = media + fis(5)*TY1(t);
media = media + fis(6)*SY1(t);
media = media +fis(7)*TSY1(t);
media = media +fis(8)*STY1(t);
U1 = U1((nval-n0+1):end,:);
aux = U1(t-nval,:);
varianza = sigma + aux*cov_betas*aux';
```

Programa 6.6.- Este programa lista los modelos encontrados por el programa algoritmo_genetico Calcula, ademas, las estimaciones puntuales de los parametros utilizando la mixtura de modelos encontrados

```
global sistema;
% Seleccion de las soluciones eficientes
q = size(Z,2);
p = size(X,2);
indices = 1:size(sistema.VFO,2);
indices = indices(sistema.ranking(indices)==1);
```

```

nmodelos = length(indices);
% Calculo de los pesos asignados a cada modelo
probabilidades = exp(-0.5*(max(-sistema.VFO(1,indices))+sistema.VFO(1,indices)));
probabilidades = probabilidades/sum(probabilidades);
% Listado de modelos mas probables
sistema.Poblacion(1:(p+q+4),indices(probabilidades>=0.01))
probabilidades(probabilidades>=0.01)
pause;
% Listado de los coeficientes estimados para los modelos mas probables
coeficientes = sistema.coeficientes(:,indices(probabilidades>=0.01));
std = sistema.std(:,indices(probabilidades>=0.01));
aux = [];
for(i=1:length(indices(probabilidades>=0.01)))
    aux = [aux,coeficientes(:,i),coeficientes(:,i)/std(:,i)];
end
strcat(literales_coeficientes,num2str(aux))
pause;
% Listado de las probabilidades de que cada variable y efecto tenga un
% efecto significativo
aux = 100*sistema.Poblacion(1:(p+q+4),indices)*probabilidades';
strcat(literales_coeficientes(1:(p+q+4),:),'',num2str(aux));
[aux(1:(p+q+4));nmodelos]
pause;
%
% Estimacion de los parametros
q = size(Z,2);
p = size(X,2);
coeficientes = zeros(p+q+13,1);
std = zeros(p+q+13,1);
for(j=1:length(indices))
    i = indices(j);
    coeficientes = coeficientes + probabilidades(j)*sistema.coeficientes(:,i);
    std = std + probabilidades(j)*(sistema.std(:,i).*sistema.std(:,i)+sistema.coeficientes(:,i).*sistema.coeficientes(:,i));
end
for(i=1:(p+q+4))
    aux = sum(probabilidades(sistema.Poblacion(i,indices)==1));
    if(aux>0)
        coeficientes(i) = coeficientes(i)/aux;
        std(i) = std(i)/aux;
    end
end
% Calculo de los estimadores de mT y gamma
aux = sum(sistema.Poblacion(27:31,:));
aux = aux(indices);
aux = sum(probabilidades(aux>0));
if(aux>0)
    for(i=(p+q+10):(p+q+11))
        coeficientes(i) = coeficientes(i)/aux;
        std(i) = std(i)/aux;
    end
end
% Calculo de los estimadores de mS y lambda
aux = sum(sistema.Poblacion(32:36,:));
aux = aux(indices);
aux = sum(probabilidades(aux>0));
if(aux>0)
    for(i=(p+q+12):(p+q+13))
        coeficientes(i) = coeficientes(i)/aux;
        std(i) = std(i)/aux;
    end
end
end
std = sqrt(std - coeficientes.*coeficientes);

```

Programas correspondientes al capítulo 6

```
% Listado de resultados
strcat(literales_coeficientes,num2str([coeficientes,std,coeficientes./std]))
coeficientes([1:(p+q+4),(p+q+9):(p+q+13)])
pause;
```

Programa 6.7.- IMPLEMENTACION LIBRE DE UN PSEUDO-ALGORITMO GENETICO MULTIOBJETIVO

Incorporamos representacion binaria 0-1, tanto para la seleccion de las variables del modelo como para la representacion del valor que toman las variables indT/S e indLambda. Consideraremos como fitness, una primera version que sea sin mas la capa a la que pertenecen las soluciones eficientes- Implementaremos inicialmente una seleccion elitista pura, con una mutacion estandar 0->1, 1->0 y una recombinacion por punto de corte sistema: variable encargada de almacenar características generales de la aplicacion del metodo, se pasa como un elemento global a todas las rutinas

```
global sistema;
sistema.nombre='Viviendas';
sistema.tipoMutacion=1;
sistema.tipoRecombinacion=1;
sistema.tipoSeleccion=1;
sistema.Q=3; %Numero de objetivos, respectivamente BIC,LOSS,LPRED
% Cada solucion se codifica en un vector de ceros y unos de tamaño.
% Tiene una dimensión fija de 26 para codificar los elementos
% indicesZ, indicesX e indicesTS (4+14+8 elementos respectivamente)
% Para indicesZ la codificacion se interpreta de la forma siguiente
% unas componentes 1001 para indicesZ representaria respecto a la notacion
% de Manolo que indicesZ toma valores 1 y 4, o unas componentes 1100
% representaria que indicesZ toma valores 1 y 2. IndicesX funciona
% exactamente igual pero con un vector de 18 componentes. Por ejemplo, un
% vector 100001111000000000, corresponderia a que indicesX contiene los elementos
% 1, 6,7,8,9,10. La interpretacion para indicesTS es la misma que la
% vuestra 8 componentes que toman valor 0 o 1.
%Tras la parte fija viene la codificacion de las variables indmT, indgamma, indmS e
% indlambda. Para codificar dichas variables almacenamos su representacion
% binaria. Para ello necesitamos saber cual es el maximo valor que puede
% tomar cada uno de ellos y reservamos las posiciones necesarias para su
% representacion binaria (log2 de dicho valor). Por comodidad y para simplificar las cuentas vamos
% a imponer que el valor maximo que pueden tomar dichas variables es siempre
% un multiplo de 2 (en consecuencia el tamaño de la discretizacion asociada
% a dichos vectores es de dicho valor maximo)
sistema.vpIT=log2(8); % indmT
sistema.vpIG=log2(4); % indgamma
sistema.vpIS=log2(8); % indmS
sistema.vpIL=log2(4); % indlambda
sistema.N=26+sistema.vpIT+sistema.vpIG+sistema.vpIS+sistema.vpIL;
sistema.tamagnoPoblacion=N2; % A fijar por el usuario
sistema.tamagnoPoblacionExt=N2; % A fijar por el usuario
% cada columna un individuo, cada fila una variable
% inicializamos aleatoriamente con ceros y unos
sistema.Poblacion=(rand(sistema.N, sistema.tamagnoPoblacion)>0.5);
% Vector de funciones objetivo, cada columna un individuo, cada fila
% una funcion objetivo
sistema.VFO=[];
```

```

sistema.ranking=[];
sistema.probabilidadMutacion=pmut;% A fijar por el usuario
sistema.probabilidadRecombinacion=pcomb;% A fijar por el usuario
sistema.numeroIteraciones=1000;% A fijar por el usuario
sistema.coeficientes = [];% Coeficientes de cada uno de los modelos
sistema.std = [];% Errores estandar de los coeficientes
% Variables del programa de criterios_seleccion, se ponen como globales para poder
% utilizarlas directamente en la funcion evalua (desde la cual se llama a
% la funcion criterios seleccion)
global X;
global Y;
global Z;
global SX;
global SY;
global TX;
global TY;
global STX;
global STY;
global TSX;
global TSY;
global mTred;
global mSred;
global gammared;
global lambdared;
global n0;
global nest;
global Cmax;
global TamanoPoblacionmin;
global BICmin;
TamanoPoblacionmax = N2;
BICmin = Inf;

```

Programa 6.8.- Esta funcion implementa el paso de mutacion de genes de alguno de los componentes de la población

```

function mutacion()
global sistema;
P=(rand(sistema.N, sistema.tamagnoPoblacionExt)<sistema.probabilidadMutacion);
mutados=sum(sum(P>0));
if(mutados>0)
    columnas=find(any(P));
    mutadosMatriz=sistema.Poblacion;
    mutadosMatriz(P)=1-mutadosMatriz(P);
    mutadosMatriz=mutadosMatriz(:,columnas);
    sistema.Poblacion=[sistema.Poblacion ,mutadosMatriz];
    sistema.tamagnoPoblacionExt=size(sistema.Poblacion,2);
    clear P mutados columnas mutadosMatriz;
end

```

Programa 6.9.- Esta funcion determina los ranking de Pareto de las soluciones encontradas hasta el momento en la poblacion ranking = 1 si la solucion correspondiente es eficiente

Programas correspondientes al capítulo 6

```
function ranking_de_poblacion()
global sistema;
desde =1;
hasta = sistema.tamagnoPoblacionExt;
sistema.ranking = zeros(1,sistema.tamagnoPoblacionExt);
num_trozos=sistema.Q;
salir=0;
num_rankineados=0;
capa=0;sit=0;
% p_prima contiene las alternativas eficientes en una capa dada
% card_p_prima contiene el numero de alternativas eficientes en una capa
% dada
p_prima = zeros(1,sistema.tamagnoPoblacionExt);
while(salir==0)
    j=desde;
    while(j<=hasta & sistema.ranking(1,j)~=0)
        j=j+1;
    end
    i=j+1;
    while(i<=hasta & sistema.ranking(1,i)~=0)
        i=i+1;
    end
    card_p_prima=1;
    p_prima(1,1)=j;
    capa=capa+1;
    sQ=sistema.Q;
    while(i<=hasta)
        j=1;
        while(j-->1)
            %determinamos la dominacion o no
            % sit determina el resultado de la comparacion entre los
            % cromosomas i y j
            % sit = 1 i domina a j
            % sit = 2 j domina a i
            % sit = 3 i, j no comparables
            sit=3;
            if(any(sistema.VFO(1:sQ,i) -sistema.VFO(1:sQ,p_prima(1,j))>1e-6))
                if(all(sistema.VFO(1:sQ,i) -sistema.VFO(1:sQ,p_prima(1,j))>=-1e-6 ))
                    sit=2;
                else
                    sit=3;
                end
            elseif(any(sistema.VFO(1:sQ,i) - sistema.VFO(1:sQ,p_prima(1,j))<-1e-6))
                if(all(sistema.VFO(1:sQ,i) - sistema.VFO(1:sQ,p_prima(1,j))<=1e-6))
                    sit=1;
                else
                    sit=3;
                end
            end
        end
    end
end
```

```

        sit=3;
    end
end
% i domina a j
if (sit==1) % i domina a j
    card_p_prima=card_p_prima-1;
    if(card_p_prima>0)
        p_prima(1,j:card_p_prima+1)=p_prima(1,j+1:card_p_prima+2);
        j=1;%p_prima(1,card_p_prima);
    else
        card_p_prima=1;
        p_prima(1,card_p_prima)=i;
        i=i+1;
        while(i<=hasta & sistema.ranking(1,i)~=0)
            i=i+1;
        end
        j=-1;
    end % j domina a i
elseif(sit==2) % j domina a i
    j=-1;
    i=i+1;
    while(i<=hasta & sistema.ranking(1,i)~=0)
        i=i+1;
    end
else
    if(j==card_p_prima)
        card_p_prima=card_p_prima+1;
        p_prima(1,card_p_prima)=i;
        i=i+1;
        while(i<=hasta & sistema.ranking(1,i)~=0)
            i=i+1;
        end
        j=-1;
    else
        j=j+1;
    end
end
end % while(j!=-1)
end % while(i<=tam_pob+hm+hr)
num_rankineados=num_rankineados+card_p_prima;
sistema.ranking(1,p_prima(1,1:card_p_prima))=capa;
if(num_rankineados==hasta)
    salir=1;
end
end % del while(salir)

clear i j salir num_rankineados card_p_prima capa hasta desde sit p_prima sQ;
```

Programa 6.10.- Esta funcion implementa el paso de recombinacion del algoritmo genetico e incorpora los nuevos elementos a la poblacion

```
function recombinacion()
global sistema;
v=rand(1,sistema.tamagnoPoblacionExt);
indices=find(v<sistema.probabilidadRecombinacion);
v=v(indices)/sistema.probabilidadRecombinacion;
recombinados=length(indices);
recombinados=2*floor(recombinados/2);
Pcopy=sistema.Poblacion(:,indices(randperm(recombinados)));
v=1+floor(v*(sistema.N-1));
for i=1:2:recombinados
    almacenado=Pcopy(v(i):sistema.N,i);
    Pcopy(v(i):sistema.N,i)=Pcopy(v(i):sistema.N,i+1);
    Pcopy(v(i):sistema.N,i+1)=almacenado;
end
sistema.Poblacion=[sistema.Poblacion,Pcopy];
sistema.tamagnoPoblacionExt=size(sistema.Poblacion,2);
clear v indices recombinados Pcopy i almacenado;_
```

Programa 6.11.- Es una seleccion elitista basada en el valor del ranking Se queda con al menos TamanoPoblacionmin elementos

```
function seleccionElitista1
global sistema;
global TamanoPoblacionmin;
[ordenado, indices1]=sort(sistema.ranking);
indices = indices1(ordenado==1);i=2;
while((length(indices)<TamanoPoblacionmax)&(i<=max(ordenado)))
    indices = [indices,indices1(ordenado==i)];
    i=i+1;
end
sistema.tamagnoPoblacion = length(indices);
sistema.Poblacion=sistema.Poblacion(:,indices);
sistema.ranking=sistema.ranking(indices);
sistema.coeficientes=sistema.coeficientes(:,indices);
sistema.std=sistema.std(:,indices);
sistema.VFO=sistema.VFO(:,indices);
sistema.tamagnoPoblacionExt=sistema.tamagnoPoblacion;
clear ordenado indices indices1 i aux;_
```

Programa 6.12.- Este programa realiza una validacion predictiva de la mixtura de modelos surgidas de la aplicacion del algoritmo genetico implementado en el programa algoritmo_genetico Utiliza el periodo 1,...,nest como estimacion y nval+1,...,n como muestra de validacion Calcula el valor de diversos criterios: RMSE, MAD, LPRED, R2, CUB95 Y CUB99 asi como analiza los errores predictivos tipificados y muestra las bandas predictivas de prediccion generada por la mixtura Realiza, ademas, un estudio grafico de los resultados obtenidos

```

function [RMSE,MAD,LPRED,R2,CUB95,CUB99] = validacion_genetico(nest,nval,tiempo,coordenadas_utm);
global sistema;
global X;
global Y;
global Z;
global SX;
global SY;
global TX;
global TY;
global STX;
global STY;
global TSX;
global TSY;
global mTred;
global mSred;
global gammared;
global lambdared;
global n0;
% Comenzamos seleccionando los modelos con ranking 1 y los ordenamos de
% mayor a menor probabilidad
Y1 = Y((nval+1):end);
nsim = 1000;
indices = 1:size(sistema.Poblacion,2);
indices = indices(sistema.ranking==1);
[aux,indices] = sort(sistema.VFO(1,indices));
sistema.VFO = sistema.VFO(:,indices);
sistema.Poblacion = sistema.Poblacion(:,indices);
sistema.coeficientes = sistema.coeficientes(:,indices);
sistema.std = sistema.std(:,indices);
sistema.ranking = sistema.ranking(indices);
% Calculo de las probabilidades a posteriori
probabilidades = exp(-(sistema.VFO(1,:)-min(sistema.VFO(1,:)))));
probabilidades = probabilidades/sum(probabilidades);
% Calculo de las predicciones y errores de prediccion asi como los
% intervalos de prediccion
n = length(Y);
errores = zeros(n-nval,1);
zr = errores;
ypred = zeros(5,n-nval);
ppred = ypred;
LPRED = 0;
for(i=(nval+1):n)
    % Calculo de los errores
    media = zeros(length(sistema.ranking),1);
    varianza = media;
    media_pred = 0;
    varianza_pred = 0;
    precio_pred = 0;
    llf = 0;
    for(j=1:length(sistema.ranking))
        [media(j),varianza(j)] = f_mle_prediccion_genetico(nest,nval,sistema.coeficientes(:,j),i);
        media_pred = media_pred + probabilidades(j)*media(j);
        precio_pred = precio_pred + probabilidades(j)*exp(media(j)+0.5*varianza(j));
        varianza_pred = varianza_pred + probabilidades(j)*varianza(j);
        llf = llf + probabilidades(j)*normpdf(Y(i),media(j),sqrt(varianza(j)));
    end
    LPRED = LPRED + log(llf);
    varianza_pred = varianza_pred + sum(probabilidades.*(media-media_pred).*(media-media_pred));
    errores(i-nval) = Y(i)-media_pred;
    zr(i-nval) = (Y(i)-media_pred)/sqrt(varianza_pred);
    u = unifrnd(0,1,nsim,1);
    lugares = ones(nsim,1);

```

Programas correspondientes al capítulo 6

```

aux = 0;
for(j=1:length(sistema.ranking));
    aux = aux+probabilidades(j);
    lugares = lugares + uplus(u>aux);
end
aux=normrnd(media(lugares),sqrt(varianza(lugares)));
ypred(:,i-nval) = prctile(aux',[0.5 2.5 50 97.5 99.5]);
ppred(:,i-nval) = exp(ypred(:,i-nval));
ypred(3,i-nval) = media_pred;
ppred(3,i-nval) = precio_pred;
end
ypred = ypred';
ppred = ppred';
% Dibujo de las predicciones
nfigura = 1;
figure(nfigura);
plot(1:(n-nval),Y1,1:(n-nval),ypred(:,3),'r',1:(n-nval),ypred(:,2),'g--',1:(n-nval),ypred(:,1),'g',1:(n-nval),ypred(:,4),'g--',1:(n-nval),ypred(:,5),'g:');
legend('y','ypred','lim95','lim99',0);
xlim([1 n-nval]);
nfigura = nfigura+1;
figure(nfigura);
plot(1:(n-nval),exp(Y1),1:(n-nval),ppred(:,3),'r',1:(n-nval),ppred(:,2),'g--',1:(n-nval),ppred(:,1),'g',1:(n-nval),ppred(:,4),'g--',1:(n-nval),ppred(:,5),'g:');
legend('p','ppred','lim95','lim99',0);
xlim([1 n-nval]);
% Calculo de los criterios numericos
RMSE = sqrt(meanerrores.*errores);
MAD = mean(abs(errores));
R2 = 1-(RMSE*RMSE)/var(Y1);
CUB95 = 100*mean((Y1>=ypred(:,2)).*(Y1<=ypred(:,4)));
CUB99 = 100*mean((Y1>=ypred(:,1)).*(Y1<=ypred(:,5)));
% Estudio de los errores tipificados
estudio_zr =
[ min(zr),prctile(zr,[1,5,10,25,50,75,90,95,99]),max(zr),mean(zr),median(abs(zr)),skewness(zr),kurtosis(zr)]
% Dibujo de los errores
nfigura = nfigura+1;
figure(nfigura);
plot(1:(n-nval),zr,1:(n-nval),-2*ones(n-nval,1),'r',1:(n-nval),2*ones(n-nval,1),'r:');
xlim([1 n-nval]);
% Histograma
nfigura = nfigura+1;
figure(nfigura)
[f,xi] = ksdensity(zr);
plot(xi,f);
% qqplot
nfigura = nfigura+1;
figure(nfigura)
qqplot(zr);
% Autocorrelacion residual
[IS1,S1] = matriz_proximidad_espacial(tiempo((nval+1):end),coordenadas_utm((nval+1):end,:),50);
nfigura = nfigura+1;
figure(nfigura)
correlaciones = autocorrelaciones_espaciales(errores,IS1,S1,1);
[IT1,T1] = matriz_proximidad_espacial(coordenadas_utm((nval+1):end,:),tiempo((nval+1):end,:),50);
nfigura = nfigura+1;
figure(nfigura)
correlaciones = autocorrelaciones_espaciales(errores,IT1,T1,1);

```

Programa 6.13.- Este programa realiza una validacion predictiva de la mixtura de modelos surgidas de la aplicacion del algoritmo genetico implementado en el programa

algoritmo_genetico Utiliza el periodo 1,...,nest como estimacion y nval+1,...,n como muestra de validacion El metodo seguido es el rolling en el sentido de que, tras obtener un nuevo dato, los pesos de la mixtura se actualizan Calcula el valor de diversos criterios: RMSE, MAD, LPRED, R2, CUB95 Y CUB99 asi como analiza los errores predictivos tipificados y muestra las bandas predictivas de prediccion generada por la mixtura Realiza, ademas, un estudio grafico de los resultados obtenidos

```

function [RMSE,MAD,LPRED,R2,CUB95,CUB99,pesos] =
validacion_genetico_rolling(nest,nval,tiempo,coordenadas_utm);
global sistema;
global X;
global Y;
global Z;
global SX;
global SY;
global TX;
global TY;
global STX;
global STY;
global TSX;
global TSY;
global mTred;
global mSred;
global gammared;
global lambdared;
global n0;
% Comenzamos seleccionando los modelos con ranking 1 y los ordenamos de
% mayor a menor probabilidad
Y1 = Y((nval+1):end);
nsim = 1000;
q = size(Z,2);
indicesq = 1:q;
p = size(X,2);
indicesp = 1:p;
indices = 1:size(sistema.Poblacion,2);
indices = indices(sistema.ranking==1);
[aux,indices] = sort(sistema.VFO(1,indices));
sistema.VFO = sistema.VFO(:,indices);
sistema.Poblacion = sistema.Poblacion(:,indices);
sistema.coeficientes = sistema.coeficientes(:,indices);
sistema.std = sistema.std(:,indices);
sistema.ranking = sistema.ranking(indices);
indmT = zeros(length(indices),1);
indicesT = 1:length(mTred);
indicesgamma = 1:length(gammared);
indicesS = 1:length(mSred);
indiceslambda = 1:length(lambdared);

```

Programas correspondientes al capítulo 6

```

indgamma = indmT;
indmS = indmT;
indlambda = indmT;
for(i=1:length(indices))
    indmT(i) = indicesT(mTred == sistema.coeficientes(p+q+10,i));
    indgamma(i) = indicesgamma(gammared == sistema.coeficientes(p+q+11,i));
    indmS(i) = indicesS(mSred == sistema.coeficientes(p+q+12,i));
    indlambda(i) = indiceslambda(lambdared == sistema.coeficientes(p+q+13,i));
    aux = indicesq'.*sistema.Poblacion(1:q,i);
    aux = aux(aux>0);
    indicesZ{i} = aux;
    aux = indicesp'.*sistema.Poblacion((q+1):(q+p),i);
    aux = aux(aux>0);
    indicesX{i} = aux;
end
% Calculo de las probabilidades a posteriori
probabilidades = exp(-(sistema.VFO(1,:)-min(sistema.VFO(1,:))));
probabilidades = probabilidades/sum(probabilidades);
pesos = probabilidades;
% Calculo de las predicciones y errores de prediccion asi como los
% intervalos de prediccion
n = length(Y);
errores = zeros(n-nval,1);
zr = errores;
ypred = zeros(5,n-nval);
ppred = ypred;
LPRED = 0;
for(i=(nval+1):n)
    % Calculo de los errores
    media = zeros(length(sistema.ranking),1);
    varianza = media;
    media_pred = 0;
    varianza_pred = 0;
    precio_pred = 0;
    llf = 0;
    for(j=1:length(sistema.ranking))
        [media(j),varianza(j)] = f_mle_prediccion_genetico(nest,nval,sistema.coeficientes(:,j),i);
        media_pred = media_pred + probabilidades(j)*media(j);
        precio_pred = precio_pred + probabilidades(j)*exp(media(j)+0.5*varianza(j));
        varianza_pred = varianza_pred + probabilidades(j)*varianza(j);
        llf = llf + probabilidades(j)*normpdf(Y(i),media(j),sqrt(varianza(j)));
    end
    LPRED = LPRED + log(llf);
    varianza_pred = varianza_pred + sum(probabilidades.*(media-media_pred).*(media-media_pred));
    errores(i-nval) = Y(i)-media_pred;
    zr(i-nval) = (Y(i)-media_pred)/sqrt(varianza_pred);
    u = unifrnd(0,1,nsim,1);

```

```

lugares = ones(nsim,1);
aux = 0;
for(j=1:length(sistema.ranking));
    aux = aux+probabilidades(j);
    lugares = lugares + uplus(u>aux);
end
aux=normrnd(media(lugares),sqrt(varianza(lugares)));
ypred(:,i-nval) = prctile(aux',[0.5 2.5 50 97.5 99.5]);
ppred(:,i-nval) = exp(ypred(:,i-nval));
ypred(3,i-nval) = media_pred;
ppred(3,i-nval) = precio_pred;
% Actualizacion de los pesos
for(j=1:length(sistema.ranking))
    [BIC,LOSS,LPRED,coeficientes,std] =
criterios_seleccion(Y(1:i),Z(1:i,:),X(1:i,:),TX(1:i,:,:),SX(1:i,:,:),STX(1:i,:,:),TSX(1:i,:,:),TY(1:i,:),SY(1:i,:),
STY(1:i,:,:),TSY(1:i,:,:),indicesZ{j},indicesX{j},sistema.Poblacion((p+q+1):(p+q+8),j),indmT(j),indgamma(j),i
ndmS(j),indlambda(j));
    sistema.coeficientes(1:(p+q+9),j) = coeficientes;
    sistema.std(1:(p+q+9),j) = std;
    sistema.VFO(1,j) = BIC;
end
probabilidades = exp(-(sistema.VFO(1,:)-min(sistema.VFO(1,:))));
probabilidades = probabilidades/sum(probabilidades);
pesos = [probabilidades,pesos];
end
ypred = ypred';
ppred = ppred';
% Dibujo de las predicciones
nfigura = 1;
figure(nfigura);
plot(1:(n-nval),Y1,1:(n-nval),ypred(:,3),'r',1:(n-nval),ypred(:,2),'g-',1:(n-nval),ypred(:,1),'g',1:(n-nval),ypred(:,4),'g-',
1:(n-nval),ypred(:,5),'g');
legend('y','ypred','lim95','lim99',0);
xlim([1 n-nval]);
nfigura = nfigura+1;
figure(nfigura);
plot(1:(n-nval),exp(Y1),1:(n-nval),ppred(:,3),'r',1:(n-nval),ppred(:,2),'g-',1:(n-nval),ppred(:,1),'g',1:(n-nval),ppred(:,4),'g-',
1:(n-nval),ppred(:,5),'g');
legend('p','ppred','lim95','lim99',0);
xlim([1 n-nval]);
% Calculo de los criterios numericos
RMSE = sqrt(meanerrores.*errores));
MAD = mean(abs(errores));
R2 = 1-(RMSE*RMSE)/var(Y1);
CUB95 = 100*mean((Y1>=ypred(:,2)).*(Y1<=ypred(:,4)));
CUB99 = 100*mean((Y1>=ypred(:,1)).*(Y1<=ypred(:,5)));
% Estudio de los errores tipificados
estudio_zr =
[min(zr),prctile(zr,[1,5,10,25,50,75,90,95,99]),max(zr),mean(zr),median(abs(zr)),skewness(zr),kurtosis(zr)]'

```


Programas correspondientes al capítulo 6

```
% Dibujo de los errores
nfigura = nfigura+1;
figure(nfigura);
plot(1:(n-nval),zr,1:(n-nval),-2*ones(n-nval,1),'r',1:(n-nval),2*ones(n-nval,1),'r');
xlim([1 n-nval]);
% Histograma
nfigura = nfigura+1;
figure(nfigura)
[f,xi] = ksdensity(zr);
plot(xi,f);
% qqplot
nfigura = nfigura+1;
figure(nfigura)
qqplot(zr);
% Autocorrelacion residual
[IS1,S1] = matriz_proximidad_espacial(tiempo((nval+1):end),coordenadas_utm((nval+1):end,:),50);
nfigura = nfigura+1;
figure(nfigura)
correlaciones = autocorrelaciones_espacialeserrores,IS1,S1,1);
[IT1,T1] = matriz_proximidad_espacial(coordenadas_utm((nval+1):end,:),tiempo((nval+1):end),50);
nfigura = nfigura+1;
figure(nfigura)
correlaciones = autocorrelaciones_espacialeserrores,IT1,T1,1);
```


Bibliografía

- Archer, W.R., Gatzlaff, D.H. y Ling, D.C. (1996). Measuring the Importance of Location in House Price Appreciation. *Journal of Urban Economics*, **40**, 334-353.
- Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. B.N. Petrov y F. Csaki Eds. *Second International Symposium on Information Theory*. Budapest: Akademiani Kiado, 267-281.
- Aznar, A., Ayuda, M. y García Olaverri, M.C. (2001). The role of simplicity in an econometric model selection process. En M.A. Feuzenkamap, M., M. McAlcer y A. Zellner (eds.) *Simplicity, Inference, and Econometric Modelling*. Cambridge, UK: Cambridge University Press.
- Bailey, M., Muth, R. y Nourse, H. (1963). A Regression Method for Real Estate Price Index Construction. *Journal of the American Statistical Association*, **58**, 933-942.
- Balakrishnan, S. y Madigan, D. (2006). A One-Pass Sequential Monte Carlo Method for Bayesian Análisis of Massive Datasets. *Bayesian Analysis*, **1**, 345-362.
- Banerjee, S., Carlin, B. P. y Gelfand, A. E. (2004). *Hierarchical Modeling and Analysis for Spatial Data*. Chapman and Hall/CRC Press.

- Banerjee, S. y Gelfand, A.E. (2006). Bayesian Wombling: Curvilinear Gradient Assessments under Spatial Process Models. *Journal of the American Statistical Association*, **476**, 1487-1501.
- Bao, H. y Wan, A. (2004). On the Use of Spline Smoothing in Estimating Hedonic Housing Price Models: Empirical Evidence Using Hong Kong Data. *Real Estate Economics*, **32**, 487-507
- Basu, S. y Thibodeau, T.G. (1998). Análisis of Spatial Correlation in House Prices. *Journal of Real Estate Finance and Economics*, **17**, 61-85.
- Berger, J.O. (1985). *Statistical Decision Theory and Bayesian Analysis*. Berlin: Springer.
- Berger, J.O. y Bernardo, J.M. (1989). Estimating a Product of means: Bayesian Analysis with Reference Priors. *Journal of the American Statistical Association*, **84**, 200-207.
- Berger, J.O., De Oliveira, V. y Sanso, B. (2001). Objective Bayesian Analysis of spatially correlated data. *Journal of the American Statistical Association*, **96**, 1361-1374
- Berger, J.O., Bernardo, J.M. y Sun, D. (2007). The formal definition of referente prior. *Annals of Statistics* (en prensa).
- Bernardo, J.M. (1979). Reference posterior distributions for Bayesian inference. *Journal Royal Statistical Society B*, **41**, 113-147
- Bernardo, J.M. y Smith, A.F.M. (1994). *Bayesian Theory*. John Wiley and Sons Ltd.

Bibliografía

- Broadhurst, D., Goodacre, R., Jones, A., Rowland, J.J., Dell, D.B. (1997). Genetic algorithms as a method for variable selection in multiple linear regression and partial least squares regression, with applications to pyrolysis mass spectrometry. *Analytica Chimica Acta*, **348**, 71-86.
- Calatrava, J. y Cañero, R. (2000). Valoración de fincas olivareras de secano mediante métodos econométricos. *Investigaciones agrarias: Producción y protección vegetal* **15** (1-2), 2000.
- Can, A. y Megbolugbe, I. (1997). Spatial Dependence and House Price Index Construction. *Journal of Real Estate Finance and Economics*, **14**, 203-222.
- Case, B (2000). Co-Movements in International Commercial Real Estate Markets. Working paper. Federal Reserve Bank.
- Case, B (2001). Co-Movements in U. S. House Price Appreciation Patterns. Working paper. Federal Reserve Bank.
- Case, B (2002). Homogeneous Within-County Districts for Hedonic Price Modeling. Working paper. Federal Reserve Bank.
- Case, B.; Clapp, J., Dubin, R. y Rodriguez, M. (2004). Modeling Spatial and Temporal House Price Patterns: A Comparison of Four Models. *Journal of Real Estate Finance and Economics*, **29**, 167-191.
- Case, B. y Quigley, J.M. (1991). The Dynamics of Real Estate Prices. *The Review of Economics and Statistics*, **73**, 50-58.
- Casella, G. y Robert, C.P. (1996). Rao-Blackwellisation of sampling schemes. *Biometrika*, **83**, 81-94.

- Celeux, G., Forbes, F., Robert, C.P. y Titterton, D.M. (2006). Deviance Information Criteria for Missing Data Models. *Bayesian Analysis* **1**, 651-674.
- Chica-Olmo, J. (1995). Spatial Estimation of Housing Prices and Locational Rents, *Urban Studies*, **32**, 1331 - 1344.
- Clapp, J.M. (2004). A Semiparametric Method for Estimating Local House Price Indices. *Real Estate Economics*, **32**, 127-160.
- Clapp, J.M., Kim, H. y Gelfand, A. E. (2002). Spatial Prediction of House Prices Using LPR and Bayesian Smoothing. *Real Estate Economics*, **30**, 505-532.
- Coello, C.A., Lamont, G.B. y Van Veldhuizen, D.A. (2007). *Evolutionary Algorithms for Solving Multi-Objective Problems* (2ª ed). Springer, Berlin.
- Colegio de Registradores de la Propiedad, Bienes Muebles y Mercantiles de España. *Estadística Registral Inmobiliaria. Anuario de 2004*.
- Colwell, F. y Dilmore, G. (1999). Who was first? An examination of an Early Hedonic Study. *Land Economics*, **75**, No. 4, 620-626.
- Court, A.T. (1939). *The Dynamics of Automobile Demand*. New York. General Motors.
- Cressie, N. (1993) *Statistics for Spatial Data*. Wiley Interscience.
- David, A., Dubujet, F; Gourieroux, C. y Laferrère, A. (2002). Les indices de prix des logements anciens. *INSEE Méthode*, **98**, 119 p.

Bibliografía

- Davidson, R. y MacKinnon, J.G. (1993). *Estimation and Inference in Econometrics*. Oxford University Press.
- De Finetti, B. (1937/1964). La Prévission: ses Lois Logiques, ses Sources Subjectives. *Ann. Inst. H. Poincaré*, 7, 1-68. Reeditado en 1980 como "Foresight; its logical laws, its subjective sources in *Studies in Subjective Probability* (H.E. Kyburg and H.E. Smolker, eds) New York: Dover, 93-158.
- De Finetti, B. (1972). *Probability, Induction and Statistics*. Chinchester:Wiley.
- Dubin, R.A. (1988). Estimation of Regression Coefficients in the Presence of Spatial Autocorrelation Error Terms. *Review of Economics and Statistics*, 70, 466-474.
- Dubin, R.A. (1992). Spatial Autocorrelation and Neighbourhood Quality. *Regional Science and Urban Economics* 22, 433-452.
- Dubin, R.A. (1998). Predicting House Prices Using Multiple Listings Data. *Journal of Real Estate Finance and Economics* 17, 35-59.
- Dubin, R.A. y Sung, C. H. (1990). Specification of Hedonic Regressions: Non-nested Tests on Measures of Neighbourhood Quality. *Journal of Urban Economics* 27, 97-110.
- Engle, R.F., Lilien, D. M. y Watson, M. (1985). A Dymimic Model of Housing Price Determination. *Journal of Econometrics*, 28, 307-326.
- Ezequiel, M. (1926). Factor affecting farmer's earnings in southern eastern Pennsylvania. *USDA Bulletin Department n° 1400*.

- Fabra, L. (2005). *Valoración de los Bienes Inmuebles e Impacto Económico en la Empresa*. Ministerio de Economía y Hacienda.
- Fisher, R.A. y Yates, F. (1938). *Statistical Tables for Biological Agricultural and Medical Research* 3ª Ed. London: Oliver & Boyd.
- Gallant, A.R. y White, H. (1988). *A Unified Theory of Estimation & Inference for Nonlinear Dynamic Models*. Basil Blackwell. Oxford and New York.
- Gelfand, A.E., Ecker, M. D., Knigh, J.R. y Sirmans, C.F. (2004). The Dynamics of Location in Home Price. *Journal of Real Estate Finance and Economics*, **29**, 149-166.
- Gelfand, A.E. y Ghosh, S.K. (1998). Model choice: a minimum posterior predictive loss approach. *Biometrika*, **85**, 1-11.
- Gelfand, A.E., Hills, S.E., Racine-Poon, A. y Smith, A.F.M. (1990). Illustration of Bayesian Inference in Normal Models using Gibbs Sampling. *Journal of the American Statistical Association*, **85**, 972-985.
- Gelfand, A.E. y Smith, A.F.M. (1990). Sampling based approaches to calculating marginal densities. *Journal of the American Statistical Association*, **85**, 398-409.
- Gelfand, A.E., Kim, H.J., Sirmans, C.F. y Banerjee, S.K (2003). Spatial Modeling with Spatially Varying Coefficient Processes. *Journal of the American Statistical Association*, **98**, 387-396.
- Gelfand, A.E., Zhu, L. y Carlin, B.P. (2001). On the Change of Support Problem for Spatio Temporal Data. *Biostatistics*, **2**, 31-45.

Bibliografía

- Gelfand, A.E., Ghosh, S.K.; Knighth, J.R. y Sirmans, C.F. (1998). Spatio-Temporal Modeling of Residential Sales Data. *Journal of Business and Economics Statistics* **16**, 312-321.
- Gelman, A. y. Rubin, D.B. (1992). Inference from iterative simulation using multiple sequences. *Statist. Sci.* **7**, 457-511.
- Gelman, A. Carlin, J.B., Stern, H.S. y Rubin, D.O. (2004). *Bayesian Data Analysis* (2ª edición). Chapman y Hall.
- Geman, S. y Geman, D. (1984). Stochastic Relaxation, Gibbs Distribution and the Bayesian Restoration of Images, *IEEE Trans. On Pattern Analysis and Machine Intelligence, PAMI*, **6**, 721-741.
- Geweke, J. (1989). Bayesian Inference in Econometric Models using Monte Carlo Integration. *Econometrica*, **57**, 1317-1339.
- Geweke, J. (1993). Bayesian Treatment of the Independent Student Linear Model. *Journal of Applied Econometrics*, **8**, 19-40.
- Geweke, J. (1992). Evaluating the Accuracy of Sampling-Based Approaches to the Calculation of Posterior Moments. *Econometrica*, **24**, 1317-1399.
- George, J.P., (1942). Correlation Analysis of Farm Land Values *Journal of Farm Economics*.
- George, E.I. y MacCulloch, R.E. (1993). Variable Selection via Gibbs Sampling. *Journal of the American Statistical Association*, **88**, 881-889.
- Gilks, W.R., Richardson, S. y Spiegelhalter, D.J. (1996). *Markov Chain Monte Carlo in Practice*. London: Chapman & Hall.

- Goldberg, D.E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Publishing Company, Reading, Massachusetts.
- Goodman, A. C., (1998). Andrew court and the Invention of Hedonic Price Analysis. *Journal of Urban Economics*, **44**, 291-298.
- Goetzman, W.N. y Spiegel, M. (1997). A Spatial Model of Housing Return and Neighborhood Substitutability. *Journal of Urban Economics* **29**, 89-103.
- Griliches, Z. (1971). *Price Indexes and Quality Change. Studies in New Methods of Measurement*. Harvard University Press, Cambridge.
- Haas, G.C. (1922). Sale prices as a basis for farm land appraisal", *Technical Bulletin 9*, Agricultural Experiment Station, University of Minnesota.
- Hammersley, J.M. y Handscomb, D.C. (1964). *Monte Carlo methods*. Wiley New York.
- Hannan, E. J. and Quinn, B. G. (1979). The determination of the order of an autoregression. *Journal of the Royal. Statistical Society B*, **41**, 190--195.
- Hastings, W. K. (1970). Monte Carlo Sampling Methods using Markov Chains and their Applications. *Biometrika*, **57**, 97-109.
- Hill, R.C., Kinght, J.R. y Sirmans, C.F. (1997). Estimating Capital Asset Price Indexes. *The Review of Economics and Statistics*, **79**, 226-233.
- Jeffreys, H., (1946). An Invariant Form for the prior probability in estimation problems. *Proceedings of the Royal Society A*, **186**,453-461.

Bibliografía

- Jeffreys, H., (1961). *Theory of Probability* (3ª Ed.) Oxford University Press, London.
- Kass, R. E. y Raftery, A. E. (1995). Bayes Factor. *Journal of the American Statistical Association*, **90**, 773-795.
- Kass, R. E., Tierney, L. y Kadane, J.B. (1989). Approximate Methods for Assessing Influence and Sensitivity in Bayesian Analysis. *Biometrika*, **76**, 663-674.
- Kloeck, T. y Van Dijk, H.K. (1978). Bayesian Estimates of System Equation Parameters: an application of Integration by Monte Carlo. *Econometrica*, **46** :1-19.
- Kong, A., Liu, J. y Wong, W. (1994). Sequential Imputation and Bayesian Missing Data Problems. *Journal of the American Statistical Association*, **89**, 278-288.
- Kruger, D.G. (1951). A Statistical Approach to Some Basic Mine Valuation Problems on the Witwatersrand. *Journal of the Chemical Metallurgical and Mining Society of South Africa*, **52**, 119-139.
- Lancaster, K. (1966). A New Approach to Consumer Theory. *Journal of Political Economy*, **74** :132-157.
- LeSage, J. P.(2004). A Family of Geographically Weighted Regression Models. En *Advances in Spatial Econometrics: Methodology, Tools and Applications*, Springer-Verlag Anselin, L., Florax, J.G.M. y Rey, S. (eds.), pág. 241-264.

- LeSage, J. (1999). The Theory and Practice of Spatial Econometrics (Unpublished Manuscript).
- Li, M. y Brown, H.J. (1980). Micro-neighborhood Externalities and Hedonic Housing Prices. *Land Economics*, **56**, 125-141.
- Lindley, D.V. y Smith, A.F.M. (1972). Bayes Estimates for the linear model. *Journal of the Royal Statistical Society B*, **34**, 1-41.
- Madigan, D. y Raftery, A. E. (1994). Model Selection and Accounting for Model Uncertainty in Graphical Models using Occam's Window *Journal of the American Statistical Association*, **89**, 1535-1546.
- Madigan, D., Gavrin, J. y Raftery, A.E. (1995). Eliciting orior information to enhance the predictive performance of Bayesian graphical models. *Communications in Statistics: Theory and Methods*, **24**, 2271-2292.
- Majumdar, A., Munneke, H.J., Gelfand, A.E., Banerjee, S. y Sirmans, C.F.(2006). Gradients in Spatial Response Surfaces with Application to Urban Land Values. *Journal of Business & Economic Statistics*, **24**, 77-90.
- Malpezzi, S., (2002). Hedonic Pricing Models: A Selective and Applied Review. Prepared for: *Housing Economics: Essays in Honor of Duncan Maclennan*.
- Marín, J.M., Mengersen, K. y Robert, C. (2005). *Bayesian Modelling Inference on Mixtures of Distributions*. En Rao, C. y Dey, D.(eds), Handbook of Statistics, vol. **25**, 654-661. Springer-Verlag, New York.

Bibliografía

- Metropolis, N. y Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. y Teller, E. (1953). Equation of State Calculations by Fast Computing Machines. *Journal of Chemical Physics*, **21**, 1087-1092.
- Militino, A.F., Ugarte, M.D. y García-Reinaldos, L. (2004). Alternative Models for Describing Spatial Dependence among Dwelling Selling Prices. *Journal of Real Estate Finance and Economics* **29** 193-209.
- Naylon, J.C. y Smith, A.F.M. (1982). Applications of a Method for the Efficient computation of Posterior Distributions. *Applied Statistics*. **31**, 214-225.
- Naylon, J.C. y Smith, A.F.M. (1988). Economic Illustrations of Novel Numerical Integration Methodology for Bayesian Inference. *Journal of Econometrics*, **38**, 103-125.
- O'Hagan, A y Forster, J. (2004). *Kendall's Advanced Theory of Statistics*. Vol. 2 *Bayesian Inference*. Oxford University Press.
- Pace, R.K., Barry, R. (1997). Quick computation of spatial autorregressive estimators. *Sgeographical Analysis*, **29**, 232-247.
- Pace, R.K., Barry, R. y Sirmans, C.F (1998a). Spatial Statistics and Real Estate. *Journal of Real Estate Finance and Economics*, **17**;1, 5-13.
- Pace, R.K., Barry, R.; Clapp, J.M. y Rodriguez, M. (1998b). Spatiotemporal Autorregressive Models of Neighborhood Effects. *Journal of Real Estate Finance and Economics*, **17**,1, 15-33.

- Pace, R.K., Barry, R.; Gilley, O.W. y Sirmans, C.F. (2000). A method for spatial-temporal forecasting with an application to real estate prices. *International Journal of Forecasting*, **16**, 229-246.
- Pace, R.K. y Gilley, O.W. (1998). Generalising OLS and the Grid Estimator. *Real Estate Economics*, **26**, 331-347.
- Parent, E. y Bernier, J. (2007). *Le raisonnement bayésien*. Springer Verlag.
- Phillips, P. C. B. (1996). Econometric model determination. *Econometrica*, **64**, 763-812.
- Phillips, P.CB y Ploberger, W. (1996). An Asymptotic Theory of Bayesian Inference for Time Series. *Econometrica*, **64**, 381-412.
- Raftery, A.E., Madigan, D.M.y Volinsky, C.T. (1996). Accounting for Model Uncertainty in Survival Analysis improves Predictive Performance. En *Bayesian Statistics 5*, ed. J.M. Bernardo, J.O. Berger, A.P. Dawid y A.F.M. Smith. Oxford Press, 323-350.
- Raftery, A.E. and Lewis, S.M. (1996). Implementing MCMC. In *Markov Chain Monte Carlo in Practice* (W.R. Gilks, D.J. Spiegelhalter and S. Richardson, eds.), London: Chapman and Hall, pág. 115-130.
- Ridgeway, G. y Madigan, D. (2002). A Sequential Monte Carlo method for Bayesian Analysis of Massive Datasets. *Data Mining and Knowledge Discovery*, **7**, 301-319.
- Ridker, R.G. y Henning, J.A. (1967). The Determinants of Property Values with Special Reference to Air Pollution. *The Review of Economics and Statistics*, **49**, 246-257.

Bibliografía

- Robert, C.P. y Casella, G. (2004). *Monte Carlo Statistical Methods*. Springer Second Edition.
- Rosen, S. (1974). Hedonic Prices and Implicit Markets: Product Differentiation in Pure Competition. *The Journal of Political Economics*, **82**, 34-55.
- Savage, L.J. (1954/1972). *The foundations of Statistics*. New York: Wiley. Second Edition in 1972, New York: Dover.
- Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, **6**, 461-464.
- Spiehehalter, D.J., Best, N.G., Carlin, B.P. y van der Linde, A. (2002). Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society Series B*, **64**, 583-639.
- Stein, M.L. (1999). Predicting Random Fields with Increasingly Dense Observations. *Annals of Applied Probability*, **9**, 242-273.
- Sirmans, G.S., Macpherson, D., Zietz, E. (2005). The Composition of Hedonic Pricing Models. *Journal of Real Estate Literature. Review*, **13**,1.
- Student (1907). On the error of counting with a haemocytometer. *Biometrika*, **5**, 351-360.
- Sun, D. y Berger, J.O. (1998). Reference priors with partial information. *Biometrika*, **85**, 55-71.
- Sun, H.; Tu, Y. y Yu, S-M. (2005). A Spatio-Temporal Autorregressive Model for Multi-Unit Residencial Market Analysis. *The Journal of Real Estate Finance and Economics*, **31**, 155-185.

- Tierney, L. y Kadane, J.B. (1986). Accurate Approximation for Posterior Moments and Marginal Densities. *Journal of the American Statistical Association*, **81**, 82-86.
- Thisted, R. (1988). *Elements of Statistical Computing: Numerical Computation*. New York: Chapman and Hall.
- Verdejo, F. (1848). Guía práctica de agrimensores y labradores. Tratado completo de agrimensura y aforaje. Imprenta de José Repullés, Madrid, 224 pp.
- Wallace, H. (1926). Comparative farm-land values in Iowa. *Journal of land and public utility economics*, October issue.
- Whittle, P. (1954). On Stationary Processes in the plane. *Biométrica*, **41**, 434-449.
- Zellner, H. (1975). Bayesian Analysis of Regression Error Terms. *Journal of the American Statistical Association*, **70**, 138-144.