Universidad Zaragoza
1542

Trabajo Fin de Grado

Gemelo digital de un reactor de gasificación en lecho fluidizado

Digital Twin of a fluidized bed gasification reactor

Autor

Isabel Peralta González

Directores

Jesús Javier Resano Ezcaray

Gemma Grasa Adiego

Isabel Martínez Berges

Grado de Ingeniería Informática

ESCUELA DE INGENIERÍA Y ARQUITECTURA
2025

TRABAJOS DE FIN DE GRADO / FIN DE MÁSTER

Escuela de Ingeniería y Arquitectura Universidad Zaragoza

DECLARACIÓN DE AUTORÍA Y ORIGINALIDAD

(Este documento debe remitirse a seceina@unizar.es dentro del plazo de depósito)

D./Dª. Isabel Peralta González ,

en aplicación de lo dispuesto en el art. 14 (Derechos de autor) del Acuerdo de 11 de septiembre de 2014, del Consejo de Gobierno, por el que se aprueba el Reglamento de los TFG y TFM de la Universidad de Zaragoza,

Declaro que el presente Trabajo de Fin de Estudios de la titulación de Grado en Ingeniería Informática (Título del Trabajo)

Gemelo digital de un redactor de gasificación en lecho fluidizado

es de mi autoría y es original, no habiéndose utilizado fuente sin ser citada debidamente.

Zaragoza, 22 Julio 2025

Fdo: Isabel Peralta González

Resumen

En este Trabajo de Fin de Grado se ha desarrollado un modelo de *gemelo digital* de un reactor de gasificación en lecho fluidizado, con el objetivo de predecir la evolución de distintas salidas gaseosas a partir de los parámetros de entrada del proceso. Esta herramienta se propone reducir la necesidad de ensayos físicos en planta piloto, acelerar la optimización de condiciones operativas y apoyar el desarrollo de tecnologías de conversión de biomasa más sostenibles.

La principal dificultad encontrada ha sido el reducido tamaño y la heterogeneidad del conjunto de datos disponible para el entrenamiento. Este hecho ha condicionado tanto el diseño metodológico como la selección de algoritmos, y motiva el interés en evaluar no sólo la **precisión** de los modelos, sino también su **robustez**, incorporando métricas de incertidumbre asociadas a las predicciones.

Se han implementado y comparado diferentes modelos de regresión: técnicas clásicas (Regresión Lineal, Random Forest, LightGBM), redes neuronales (MLP) y redes bayesianas (BNN). Los resultados muestran que no existe un único modelo óptimo para todas las salidas: mientras que el MLP general alcanza los mejores resultados en compuestos como el etileno ($R^2 = 0.9504$), modelos más simples como la regresión lineal o LightGBM destacan en CO_2, CO y benceno.

Se incorporó un análisis de incertidumbre mediante intervalos de confianza, observándose que en este conjunto de test reducido las coberturas fueron 100 % para algunas salidas y 83 % para otras. Asimismo, se evaluó la influencia de las biomasas utilizadas en entrenamiento y test, confirmando que los modelos tienden a especializarse en la biomasa predominante.

Finalmente, se propone un enfoque híbrido para el gemelo digital: utilizar el MLP especializado en biomasa 1 cuando corresponda, y seleccionar modelos alternativos (clásicos o probabilísticos) en el resto de biomasas, maximizando así la precisión y la robustez del sistema.

Índice

Capítulo 1

Introducción

En la transición hacia un sistema energético más sostenible, los procesos termoquímicos que convierten biomasa en combustibles o vectores energéticos limpios tienen un papel destacado. La gasificación con sorción mejorada (SEG) es especialmente interesante porque permite obtener un gas de síntesis con una composición ajustada para aplicaciones como producción de hidrógeno, combustibles sintéticos o electricidad, mientras captura parte del CO_2 generado. Sin embargo, optimizar este tipo de procesos requiere múltiples experimentos, lo que implica un elevado consumo de tiempo y recursos en planta piloto.

En este contexto, el uso de un *gemelo digital* ofrece una solución eficiente. Un gemelo digital es una réplica virtual de un sistema físico que se actualiza y calibra a partir de datos reales, de forma que reproduce su comportamiento bajo distintas condiciones de operación. En el ámbito industrial, estas herramientas permiten simular, predecir y optimizar procesos sin necesidad de realizar pruebas físicas en cada escenario. (Barricelli et al., 2019; Grieves, August 2016)

En el caso concreto de este trabajo, el gemelo digital del proceso SEG se construye mediante modelos de aprendizaje automático entrenados con datos experimentales de planta piloto. Estos modelos permiten predecir la composición del gas y otras variables clave a partir de parámetros de entrada como la temperatura de operación, los caudales de biomasa, vapor y sorbente, o las características de la biomasa utilizada. El uso de un gemelo digital en este contexto consiste en crear un modelo que reproduzca el comportamiento del proceso real, proporcionando predicciones rápidas y fiables que reducen la necesidad de ensayos físicos, ahorran recursos y facilitan la exploración de distintos escenarios operativos.

1.1. Motivación

Optimizar y comprender mejor el proceso SEG requiere realizar numerosos ensayos experimentales en planta piloto, lo que implica un consumo considerable de tiempo y recursos.

Un gemelo digital permite evaluar rápidamente distintos escenarios de operación sin necesidad de realizar pruebas físicas, lo que agiliza la investigación y la toma de decisiones. Además, facilita el análisis de la influencia de las variables de entrada sobre el comportamiento del proceso y permite identificar posibles anomalías o situaciones no deseadas.

En este trabajo, la motivación surge tanto del interés académico por explorar la aplicación de técnicas de aprendizaje automático en procesos termoquímicos, como de la utilidad práctica que esta herramienta puede ofrecer en un entorno experimental real. La colaboración con un equipo de investigación que opera la planta piloto ha proporcionado acceso a datos experimentales de alta calidad, lo que ha permitido desarrollar y validar el gemelo digital en un caso de estudio real.

Cabe destacar que la disponibilidad de datos experimentales era limitada, lo que añade un reto adicional al desarrollo de modelos predictivos fiables. Este contexto ha condicionado el diseño de la metodología y ha motivado la exploración de técnicas que funcionen bien con conjuntos de datos pequeños.

Además, se ha buscado enriquecer las predicciones con métricas de incertidumbre, que proporcionen información adicional sobre la robustez de las predicciones.

1.2. Objetivos

Desarrollar un gemelo digital del proceso SEG capaz de predecir las variables de salida a partir de las variables de entrada, evaluando su rendimiento y su capacidad de generalizar a diferentes biomasas, gestionando datos anómalos, estimando la incertidumbre de las predicciones y proporcionando información sobre las variables clave del proceso mediante modelos explicables.

1.3. Estructura del documento

El documento está organizado de la siguiente forma:

– En el Capítulo 2 se presentan los fundamentos teóricos, incluyendo el concepto de gemelo digital, el proceso SEG y las técnicas de modelado utilizadas.

– El Capítulo 3 describe el conjunto de datos y el preprocesado realizado.

– El Capítulo 4 detalla la metodología seguida, explicando los modelos utilizados, la validación y las estrategias adicionales.

– El Capítulo 5 recoge los resultados obtenidos, su análisis y discusión.

– Por último, en el Capítulo 6 se presentan las conclusiones y las posibles líneas de trabajo futuro.

Capítulo 2

Fundamentos teóricos

2.1. Concepto de gemelo digital

2.1.1. Definición y componentes

Un gemelo digital es una representación virtual de un sistema físico que se actualiza de forma continua con datos reales, con el fin de reproducir su comportamiento lo más fielmente posible. Está compuesto, en general, por tres elementos principales:

- **El sistema físico real**: el proceso, máquina o instalación que se desea modelar.
- **El modelo virtual**: representación matemática o computacional que simula el funcionamiento del sistema.
- **La conexión de datos**: flujo de información que actualiza el modelo con datos procedentes del sistema físico.

La integración de estos elementos permite que el gemelo digital sea una herramienta predictiva y de soporte a la toma de decisiones. (Barricelli et al., 2019; Grieves, August 2016)

2.1.2. Aplicaciones en ingeniería y procesos

Los gemelos digitales se emplean en sectores como la fabricación, la energía o el transporte para optimizar el rendimiento, planificar mantenimientos y reducir costes. En el ámbito de los procesos termoquímicos, permiten analizar escenarios de operación, estudiar la influencia de variables de entrada, predecir resultados y estimar la incertidumbre sin necesidad de realizar ensayos físicos en cada caso.

2.2. Procesos termoquímicos

2.2.1. Descripción general del proceso

Los procesos termoquímicos convierten materiales sólidos como la biomasa en productos gaseosos o líquidos mediante reacciones químicas impulsadas por calor. En este trabajo, se estudia la gasificación con sorción mejorada (SEG), un proceso que combina la conversión de biomasa en gas de síntesis con la captura de dióxido de carbono durante la reacción, utilizando óxido de calcio (CaO) como sorbente. (Callen et al., 2022; Martinez et al., 2020)

2.2.2. Variables relevantes y sensores típicos

En la gasificación con sorción mejorada, las variables de entrada incluyen condiciones como la temperatura de operación, el flujo de vapor, la relación CaO/biomasa y las características de la biomasa utilizada. Las variables de salida de interés suelen ser las fracciones molares de los principales componentes del gas (CO, CO2, H2, etc.). Estas se miden habitualmente mediante analizadores de gases en línea, mientras que las variables de entrada pueden registrarse con termopares, caudalímetros y otros sensores de proceso.

2.3. Fundamentos de modelado y aprendizaje automático

2.3.1. Regresión supervisada

La regresión supervisada es una técnica de aprendizaje automático en la que un modelo se entrena para predecir una variable continua (salida) a partir de un conjunto de variables de entrada, utilizando ejemplos previamente observados. La calidad del modelo se evalúa comparando sus predicciones con los valores reales mediante métricas como el error cuadrático medio (MSE) o el coeficiente de determinación (R^2).

2.3.2. Modelos utilizados

En este trabajo se emplean varios tipos de modelos:

– **LightGBM (Gradient Boosting Decision Trees)**: algoritmo de *gradient boosting*, en el que cada árbol se entrena de forma secuencial para corregir los

errores cometidos por los anteriores. Este enfoque permite capturar relaciones no lineales de manera eficiente y suele ofrecer un rendimiento competitivo incluso con conjuntos de datos pequeños o medianos, además de proporcionar medidas de importancia de variables útiles para la interpretación.

- **Random Forest**: conjunto de árboles de decisión entrenados de forma independiente, cuyas predicciones se combinan para mejorar la robustez y reducir el sobreajuste.

- **Regresión lineal**: modelo sencillo que asume una relación lineal entre las variables de entrada y de salida.

- **Perceptrón multicapa (MLP)**: red neuronal artificial con varias capas de neuronas conectadas, capaz de modelar relaciones no lineales complejas.

- **Redes bayesianas discretas**: modelos probabilísticos que representan relaciones de dependencia entre variables mediante un grafo dirigido acíclico, permitiendo inferencias y análisis de variables clave.

- **Red neuronal bayesiana (BNN)**: extensión de las redes neuronales tradicionales que incorpora incertidumbre en los pesos mediante distribuciones probabilísticas. Esto permite no solo predecir valores, sino también cuantificar la confianza en las predicciones, lo cual es especialmente útil en escenarios con datos ruidosos o escasos.

2.3.3. Estimación de incertidumbre y modelos explicables

Además de obtener predicciones precisas, en muchos contextos es importante conocer la *confianza* asociada a los resultados del modelo. La estimación de incertidumbre permite cuantificar hasta qué punto se puede confiar en una predicción y detectar posibles situaciones anómalas. Existen diferentes enfoques para este propósito, como el ajuste de distribuciones gaussianas sobre las salidas o el uso de redes neuronales bayesianas, que proporcionan distribuciones de probabilidad en lugar de valores deterministas.

Por otro lado, la interpretabilidad de los modelos es clave para comprender el funcionamiento del proceso y apoyar la toma de decisiones. En este sentido, se pueden emplear modelos explicables como las redes bayesianas discretas, que muestran dependencias entre variables en forma de grafo, o bien técnicas de análisis de importancia de variables en modelos de *machine learning* más complejos, como LightGBM. Estos

enfoques permiten no solo predecir, sino también identificar qué variables de entrada tienen mayor influencia en las salidas del proceso.

Capítulo 3

Conjunto de datos y preprocesado

3.1. Descripción del conjunto de datos

El conjunto de datos empleado en este trabajo proviene de una serie de experimentos de gasificación de biomasa realizados en un reactor de lecho fluidizado. En total se dispone de 26 experimentos independientes (inicialmente 23), cada uno caracterizado por un conjunto de variables de entrada y de salida.

Las variables de entrada incluyen caudales de biomasa, sorbente y vapor de agua, así como relaciones calculadas como la razón molar vapor-carbono (S/C) y la razón másica sorbente-biomasa (S/B). Además, se cuenta con la composición elemental y el análisis inmediato de la biomasa utilizada, junto con perfiles de temperatura en distintos puntos del reactor.

En cuanto a las salidas, el dataset recoge la composición del gas de salida (expresada en fracción molar base seca sin nitrógeno ni oxígeno), incluyendo componentes principales como hidrógeno, dióxido de carbono, monóxido de carbono, metano, etano, etileno y otros hidrocarburos ligeros. También se mide la producción de gas por kilogramo de biomasa, la concentración de compuestos azufrados (H_2S, COS, CH_4S), el contenido de alquitranes mediante análisis gravimétrico y cromatografía (GC-MS), y la fracción de sólidos de salida asociada al residuo sólido carbonoso (char).

Los experimentos abarcan cinco biomasas distintas, aunque de manera desequilibrada: una de ellas está representada con 16 experimentos, dos biomasas con 4 experimentos cada una, y otras dos con un único experimento cada una. Estos últimos resultaron particularmente problemáticos, ya que su comportamiento no seguía las mismas tendencias y fueron tratados como datos anómalos durante el análisis.

En la Tabla 3.1 se muestra un resumen de las variables disponibles, clasificadas entre entradas y salidas.

Entradas	Salidas
Caudal biomasa, sorbente, H_2O	Composición gas (H_2, CO_2, CO, Metano, etc.)
Relación molar Steam/Carbon y Relación másica Sorbente/Biomasa	Producción de gas (Nm^3/kg biomasa)
Composición elemental (C, H, N, S, Cl)	Alquitranes (gravimétrico)
Análisis inmediato (volátiles, cenizas, humedad, carbono fijo)	Concentración contaminantes azufrados
Perfil de temperaturas del reactor	Fracción de char y conversión de C fijo

Tabla 3.1: Resumen de variables de entrada y salida del conjunto de datos.

3.2. Preprocesado de datos

Antes del modelado, se llevó a cabo un preprocesado para adaptar el conjunto de datos a las necesidades de las técnicas de aprendizaje automático. Los pasos principales fueron:

- **Normalización:** todas las variables de entrada y salida se escalaron al rango [0,1] mediante `MinMaxScaler` para estabilizar el entrenamiento y facilitar la comparación entre magnitudes heterogéneas. (scikit-learn developers, 2025h)

- **Gestión de valores faltantes:** cuando existieron valores ausentes, se aplicó una estrategia conservadora: imputación simple (por ejemplo, con estadísticas del entrenamiento) si el faltante era puntual y no sistemático; eliminación de la observación únicamente cuando el dato era crítico para el modelo y no había alternativa fiable. (scikit-learn developers, 2025d)

- **Separación de conjuntos:** se definió un **conjunto de test fijo** de seis muestras y un **conjunto anómalo** con los dos experimentos de biomasas poco representadas (un solo ensayo cada una). El resto se empleó para entrenamiento y validación. (scikit-learn developers, 2025f)

- **Replicación y agrupación:** para mitigar el fuerte desequilibrio entre la cantidad de muestras de cada biomasa, se *replicaron* algunas muestras minoritarias únicamente en el **conjunto de entrenamiento**. En validación cruzada se usaron **grupos por biomasa** para evitar fugas de información y respetar la estructura del problema.

– **Selección alternativa de test:** en fases posteriores se exploró **K-means** como herramienta para elegir subconjuntos de test más *representativos* del espacio de entrada (prototipos), comparando sus resultados con el test fijo.

3.3. Análisis preliminar de relevancia de variables

Como paso exploratorio, se aplicó un modelo de tipo Gradient Boosted Decision Trees (GBDT) para estimar la importancia relativa de las entradas en la predicción de cada salida. El objetivo era identificar qué variables aportaban información relevante y cuáles podían descartarse para reducir la dimensionalidad.

Los resultados mostraron que la mayor parte de la variabilidad en las salidas se explicaba a partir de los caudales de operación (biomasa, sorbente y agua), las relaciones S/C y S/B, y los perfiles de temperatura. En cambio, las variables de análisis inmediato como *%wt. FC* , *%wt. Cenizas* y *%wt. Cl* resultaron irrelevantes. Estos resultados se resumen en la Figura 3.1, donde se muestra la importancia relativa estimada para cada variable.

Este análisis puso de manifiesto el carácter altamente heterogéneo del conjunto. La escasez de muestras y la distribución desigual entre biomasas condicionaron de forma importante el diseño de la metodología y la interpretación de resultados.

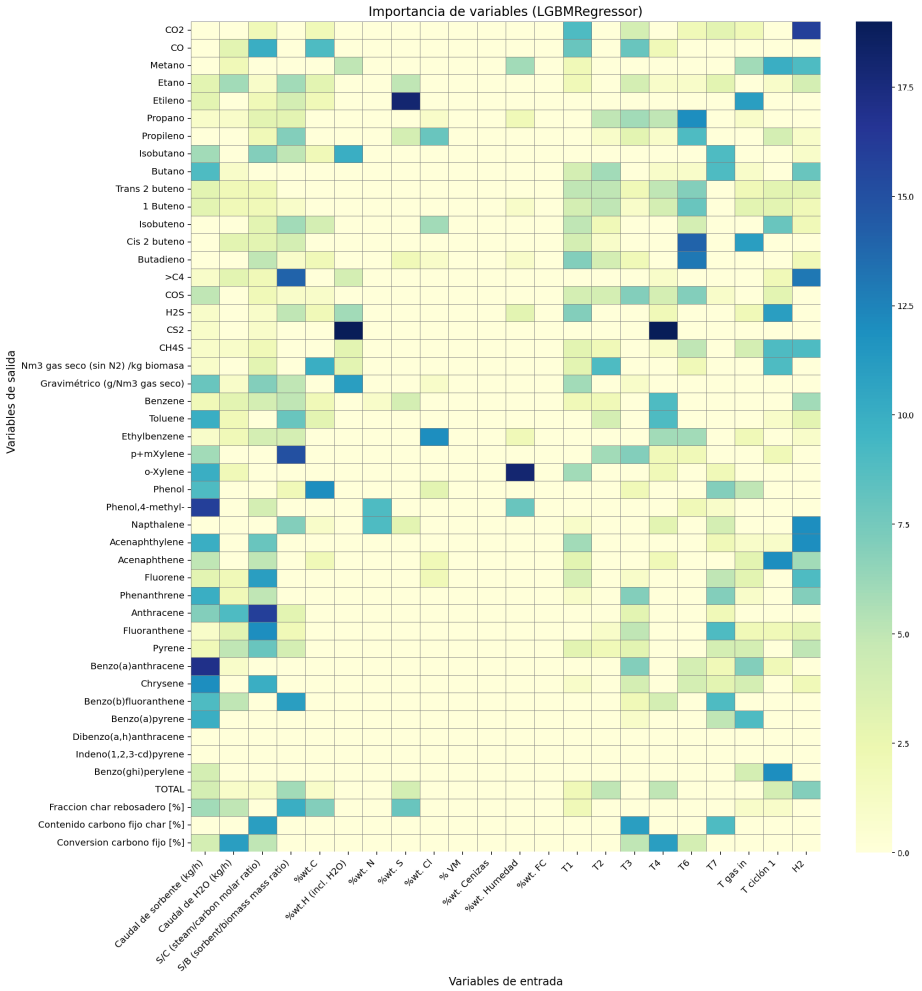

Figura 3.1: Importancia relativa de las variables de entrada según un modelo GBDT.

Capítulo 4

Metodología

4.1. Estrategia general

La metodología de este trabajo se centra en el desarrollo de un gemelo digital del proceso de gasificación con sorción mejorada (SEG) a partir de un conjunto de datos limitado y heterogéneo. El objetivo es construir modelos predictivos que permitan estimar las variables de salida del proceso a partir de las condiciones de operación y características de la biomasa.

Cabe señalar que se decidió excluir la entrada **T5**, ya que en una parte significativa de las muestras su valor aparecía en blanco, lo que dificultaba su uso como variable fiable en los modelos.

Dado que el número de muestras disponibles es reducido y existe un fuerte desequilibrio entre biomasas, se adoptó una estrategia basada en:

- Comparar distintos algoritmos de aprendizaje supervisado.
- Diseñar diferentes escenarios de validación para estudiar la capacidad de generalización.
- Analizar el impacto de los datos anómalos y de la replicación de biomasas minoritarias.
- Incorporar técnicas de explicabilidad (importancia de variables y redes bayesianas discretas).
- Explorar métodos de estimación de incertidumbre en modelos neuronales.

4.2. Modelos empleados

El trabajo comenzó utilizando **LightGBM** como modelo principal. Dado que en algunas salidas no se conseguían buenos resultados, se añadieron también **Random Forest** y **Regresión lineal** como alternativas más sencillas para comparar rendimientos y verificar si podían capturar relaciones básicas en los datos. Una vez establecida esta línea base, se pasó a modelos más complejos como el **MLP** y, finalmente, a las **BNN**.

4.2.1. Regresión lineal

Se utilizó la regresión lineal múltiple como modelo de referencia simple. Aunque su capacidad es limitada en procesos no lineales, permitió comprobar si existían dependencias lineales dominantes y sirvió como comparación frente a los modelos más complejos. (scikit-learn developers, 2025e)

4.2.2. Random Forest

El modelo Random Forest se empleó en las salidas donde LightGBM no alcanzaba buen ajuste, como alternativa basada en *bagging*. Su robustez frente a ruido y su facilidad para capturar relaciones no lineales lo convirtieron en un buen comparador, aunque con menor interpretabilidad. (Breiman, 2001; scikit-learn developers, 2025c)

4.2.3. LightGBM (Gradient Boosted Decision Trees)

LightGBM fue el modelo de referencia. Se ajustaron sus hiperparámetros clave (número de árboles, profundidad y tasa de aprendizaje).

Además, se diseñaron varios experimentos con LightGBM:

- Entrenamiento con distintas combinaciones de biomasas, evaluando la capacidad de generalizar a las excluidas.
- Separación estricta de muestras de una misma biomasa entre train y test, para comprobar la predicción en condiciones similares pero no idénticas.
- Replicación de biomasas minoritarias para equilibrar el dataset y estudiar su efecto sobre el rendimiento.

Este modelo proporcionó también medidas de **importancia de variables**, utilizadas para identificar qué entradas eran más influyentes en cada salida.

4.2.4. Perceptrón multicapa (MLP)

Una vez explorados los modelos anteriores, se entrenaron redes neuronales de tipo *Perceptrón Multicapa* (MLP), capaces de capturar relaciones no lineales más complejas. (scikit-learn developers, 2025g)

Dado el reducido tamaño de datos, se aplicaron estrategias específicas:

- Exclusión de biomasas anómalas para evitar distorsiones en el entrenamiento.
- Entrenamiento de múltiples modelos con distintas configuraciones, seleccionando los que superaban un umbral de calidad y combinándolos en **ensembles de 100 modelos**.
- Entrenamiento con la biomasa predominante para evaluar su ajuste y su capacidad de generalizar al resto.

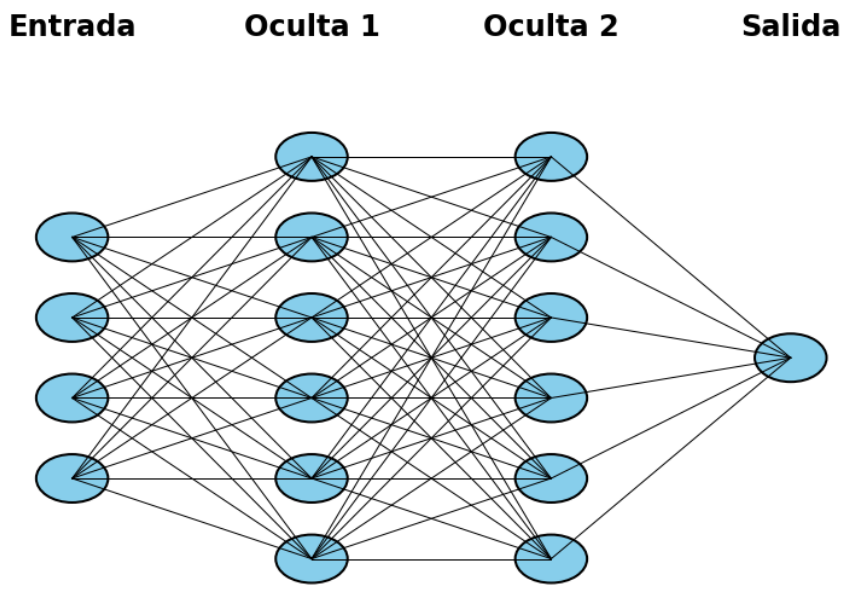

Figura 4.1: Ejemplo de arquitectura de MLP

4.2.5. Redes bayesianas discretas

De forma complementaria, se construyeron redes bayesianas discretas para explorar dependencias estructurales entre variables de entrada y salida. Para ello se usó la librería

`pgmpy`, con discretización mediante *KBinsDiscretizer* y aprendizaje de estructura con el algoritmo PC.

La mayoría de salidas no mostraron dependencias significativas.

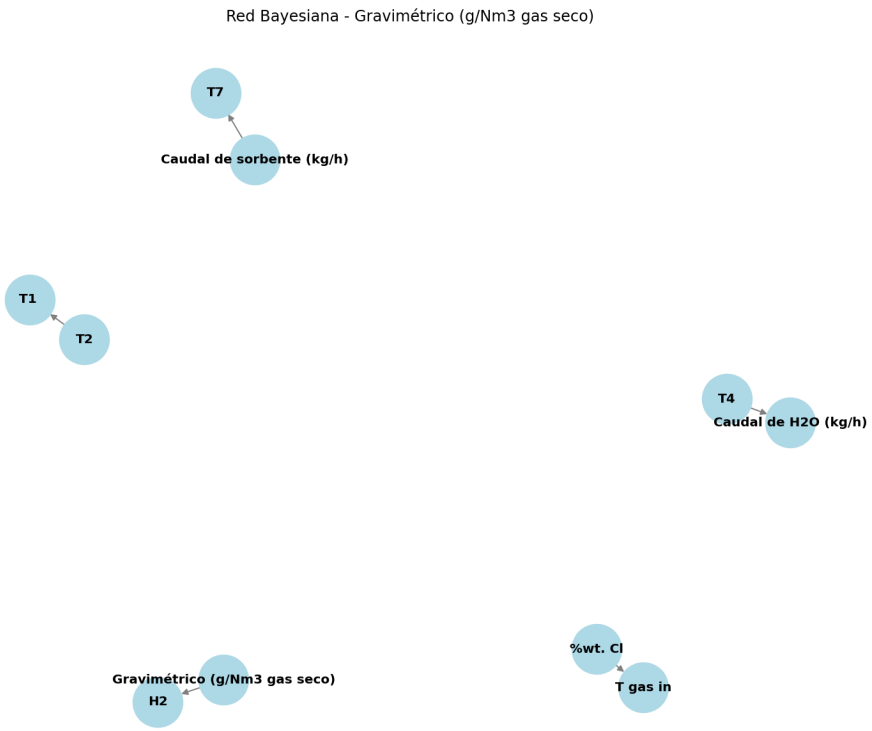

Figura 4.2: Ejemplo de red bayesiana discreta para la salida Gravimétrico (g/Nm3 gas seco)

4.2.6. Redes bayesianas neuronales (BNN)

Finalmente, se implementaron Redes Bayesianas Neuronales (BNN) con el objetivo de añadir estimación de incertidumbre. En lugar de pesos fijos, los parámetros se modelaron como distribuciones, lo que permitió obtener para cada entrada una distribución de salidas (media y desviación estándar).

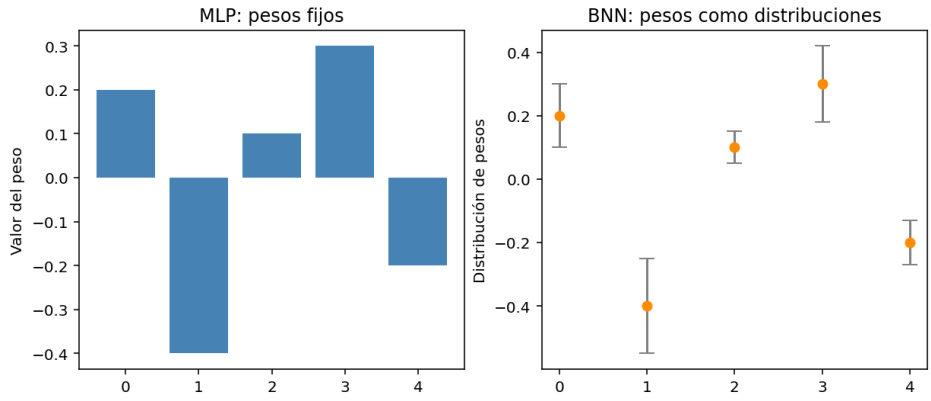

Figura 4.3: Diferencia conceptual entre MLP y BNN

Se probaron dos esquemas de inicialización:

- Entrenamiento desde cero, con pesos aleatorios.
- Entrenamiento partiendo de los pesos del mejor MLP obtenido para cada salida, lo que mejoró la estabilidad y aceleró la convergencia.

En todos los casos se excluyeron las biomasas anómalas. Además, los hiperparámetros detallados de cada modelo y la configuración completa para garantizar reproducibilidad se recogen en el Apéndice A.

4.3. Validación y evaluación

4.3.1. Métricas utilizadas

El rendimiento de los modelos se evaluó mediante dos métricas principales:

- **Error cuadrático medio (MSE):** mide la magnitud de los errores de predicción, penalizando más los errores grandes.
- **Coeficiente de determinación (R^2):** cuantifica la proporción de la variabilidad de la salida que el modelo es capaz de explicar.

(scikit-learn developers, 2025a)

4.3.2. Esquemas de validación

Dado el reducido tamaño y la heterogeneidad del dataset, no se aplicó validación cruzada en el sentido clásico. En su lugar, se definieron distintos **escenarios de**

entrenamiento y prueba que permiten analizar la capacidad de generalización de los modelos en función de las biomasas y de la presencia de datos anómalos. Estos escenarios fueron:

- **Separación por biomasas:** se entrenó con un subconjunto de biomasas y se evaluó tanto en esas mismas biomasas (separando parte de sus muestras para test) como en biomasas completamente excluidas del entrenamiento.
- **Entrenamiento con la biomasa predominante:** se construyeron modelos usando únicamente la biomasa más representada, evaluando su capacidad de ajuste interno y de generalización hacia otras biomasas.
- **Replicación de biomasas minoritarias:** se equilibró el dataset replicando muestras de biomasas poco representadas, para estudiar el efecto en el ajuste y en la generalización.
- **Evaluación en datos anómalos:** se analizaron por separado las biomasas consideradas anómalas (con un único experimento disponible), usándolas únicamente como conjunto de test externo.
- **Selección de test mediante K-means:** además de los esquemas anteriores, se aplicó K-means sobre el espacio de entrada para escoger muestras representativas (centroides) como conjunto de test alternativo. (scikit-learn developers, 2025b)

Este enfoque permitió explorar distintos grados de dificultad en la validación: desde escenarios más favorables (predicción dentro de la misma biomasa) hasta los más exigentes (predicción en biomasas no vistas durante el entrenamiento).

4.4. Estrategias adicionales

4.4.1. Prevención del sobreajuste

Dado el bajo número de muestras disponibles, la prevención del sobreajuste fue un aspecto central. Para ello:

- Se excluyeron los datos anómalos en la mayoría de entrenamientos.
- Se equilibraron las biomasas mediante replicación controlada.
- Se garantizó que una misma muestra y sus réplicas no aparecieran simultáneamente en los conjuntos de entrenamiento y test.

– Se limitaron la complejidad de las redes neuronales y se ajustaron los hiperparámetros.

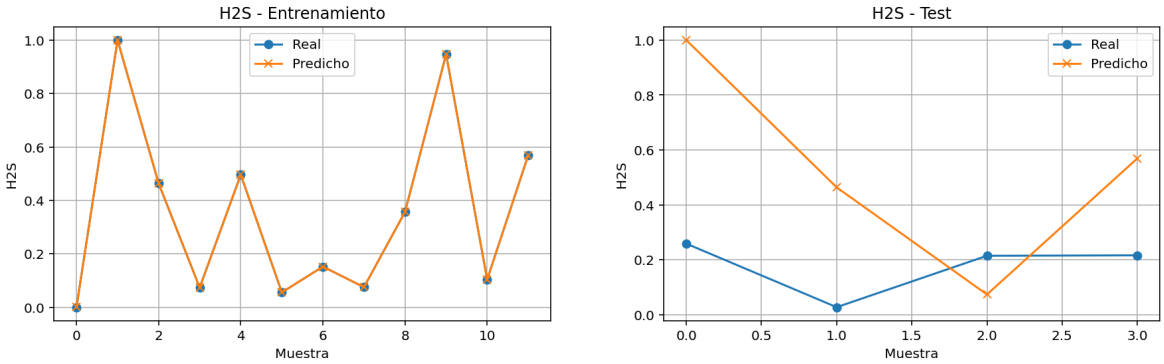

Figura 4.4: Ejemplo de sobreajuste

4.4.2. Análisis de importancia de variables

Con LightGBM se obtuvieron mapas de importancia de variables. Esto permitió identificar qué características de la biomasa y qué condiciones operativas tienen mayor impacto sobre cada salida.

4.4.3. Discretización para redes bayesianas

Las redes bayesianas discretas requirieron transformar las variables continuas en intervalos. Para ello se emplearon métodos basados en cuantiles y número fijo de bins. Se analizaron los problemas derivados de intervalos excesivamente estrechos, que en algunos casos limitaron la interpretación.

4.4.4. Estimación de incertidumbre en modelos neuronales

En las BNN, cada entrada se evaluó varias veces para obtener una distribución de posibles salidas. De este modo, se calcularon tanto el valor medio como la desviación estándar de las predicciones, lo que permite interpretar no solo el resultado esperado, sino también la confianza asociada. (Blundell et al., 2015; Gal & Ghahramani, 2016)

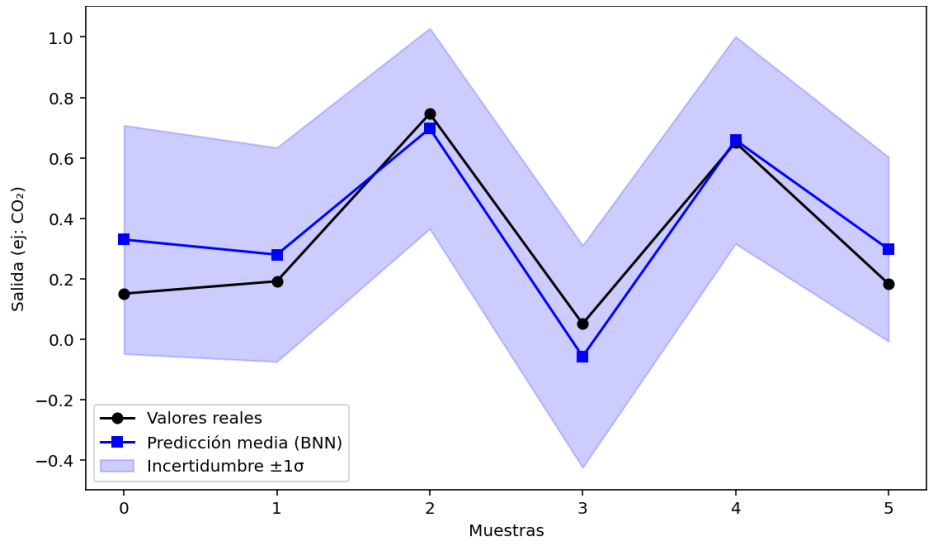

Figura 4.5: Ejemplo de predicción con incertidumbre (BNN)

4.5. Implementación del script de inferencia

Se implementó el script `gemelo.py` para automatizar la inferencia sobre una muestra (fila) de entrada y conservar trazabilidad de las decisiones. A continuación se resume su flujo operativo (para detalles técnicos y pseudocódigo ver Anexo B):

1. Lectura de la muestra de entrada (por defecto `data/input_example.csv`) y selección de la fila indicada.

2. Detección automática de si la muestra corresponde a la *biomasa 1* (criterio y tolerancias en Anexo B).

3. Construcción del índice de modelos disponible (carpetas `mod/`, `mod1/` y checkpoints) y, para cada variable de salida, selección del **mejor modelo según R^2** en validación.

4. Aplicación de los preprocesados asociados (imputación y `scaler_X`), ejecución de la predicción y desescalado (`scaler_Y.inverse_transform`) para devolver valores en unidades reales.

5. Evaluación de condiciones de confianza: marcado de banderas `OUT_OF_DISTR` / `FLAG_UNCERTAIN` si la predicción excede umbrales de incertidumbre (definidos en Anexo B); opcionalmente aplicar `CLAMP_NON_NEGATIVE` para salidas físicamente no negativas.

6. Guardado de resultados en `results/resultado_<entrada>.csv` y actualización de `mod_index.json` que registra qué modelo se usó por salida y sus métricas (R^2, MSE, cobertura IC).

Nota: la especificación completa de formatos, el pseudocódigo y ejemplos de entrada/salida están en Anexo B (Anexo B).

Capítulo 5

Resultados

En este capítulo se presentan los resultados obtenidos al aplicar los distintos modelos de regresión al conjunto de datos del proceso termoquímico. Se incluyen tanto las métricas de error (R^2 y MSE) como el análisis de incertidumbre de las predicciones. Los resultados se organizan comparando modelos tradicionales (Linear Regression, Random Forest, LightGBM), modelos de redes neuronales (MLP), y modelos probabilísticos (BNN).

La tabla completa con la comparación de resultados (R^2 y MSE) se ha trasladado al Anexo (Tabla A.1), para mejorar su legibilidad en formato horizontal.

5.1. Resultados globales por modelo

En la Tabla A.1 (Tabla A.1) se resumen los resultados de R^2 y MSE de los modelos entrenados de forma generalizada sobre todas las biomasas (MLP-gral), entrenados únicamente con biomasa 1 (MLP-B1), así como los obtenidos con modelos clásicos y las dos variantes de redes bayesianas (BNN v2 y BNN inicializado desde MLP). Se muestran las salidas más relevantes junto con algunas adicionales para ofrecer una visión más completa.

En el caso del modelo *MLP biomasa 1*, los indicadores con la etiqueta *int* corresponden a la validación interna sobre la misma biomasa empleada en el entrenamiento, mientras que los indicadores con la etiqueta *ext* reflejan la validación externa realizada con biomasas diferentes a las utilizadas en el entrenamiento.

5.2. Resultados detallados por salida

A continuación se discuten las salidas más relevantes, interpretando cada columna como un modelo distinto y resaltando los rendimientos más discriminantes observados en la Tabla A.1 (Tabla A.1).

5.2.1. Etileno

Los mejores resultados absolutos de la Tabla muestran que la **MLP entrenada sobre todo el conjunto (MLP-gral)** consigue MSE = 0.0039 y $R^2 = 0.9504$, siendo uno de los modelos más precisos y con buena capacidad de generalización. La **LGBM** también presenta un ajuste muy competitivo ($R^2 = 0.9484$, MSE = 0.0041). La **MLP-B1** exhibe un ajuste interno muy bueno ($R^2_{\text{int}} = 0.9435$, $\text{MSE}_{\text{int}} = 0.0007$) pero su validación externa es deficitaria ($\text{MSE}_{\text{ext}} = 2.1590$, $R^2_{\text{ext}} = -1.3571$), lo que indica que ese modelo especializado no generaliza a otras biomasas. Las variantes bayesianas presentan comportamientos distintos: el **BNN desde MLP** tiene MSE = 0.0040 y $R^2 = 0.9495$ con desviación de predicción ≈ 0.134, comparable al MLP-gral pero proporcionando además una medida de incertidumbre; el **BNN** entrenado desde cero ofrece $R^2 = 0.9023$ y desviación ≈ 0.158.

5.2.2. Monóxido de carbono (CO)

Para CO, la **LGBM** obtiene el R^2 más alto entre los modelos clásicos ($R^2 = 0.9077$, MSE = 0.0077). La **MLP-gral** sigue en rendimiento ($R^2 = 0.8465$, MSE = 0.0129). La MLP especializada (MLP-B1) muestra de nuevo alto ajuste interno ($R^2_{\text{int}} = 0.8704$) pero su validación externa es extremadamente mala ($\text{MSE}_{\text{ext}} = 1.2293$, $R^2_{\text{ext}} = -76.1524$), lo que sugiere problemas severos de extrapolación en este caso concreto. Entre las variantes bayesianas, el **BNN desde MLP** alcanza $R^2 = 0.8918$ (MSE = 0.0091), situándose cerca de LGBM, mientras que el BNN desde cero no logra igualar a estos modelos ($R^2 = 0.2067$).

5.2.3. Dióxido de carbono (CO_2)

La **MLP-gral** presenta buen desempeño ($R^2 = 0.8456$, MSE = 0.0110). La **MLP-B1** obtiene un R^2_{int} excepcionalmente alto (0.9840) pero nuevamente su validación externa da lugar a $R^2_{\text{ext}} = -1.7451$, evidenciando falta de generalización fuera de la biomasa de entrenamiento. Destaca el **BNN desde MLP** con $R^2 = 0.8829$ (MSE = 0.0083),

que supera ligeramente a la MLP-gral y ofrece además estimación de incertidumbre (desviación ≈ 0.230). Entre los clásicos, la regresión lineal alcanza $R^2 = 0.6018$ en test.

5.2.4. Hidrógeno (H_2)

La **MLP-gral** obtiene un buen resultado global ($R^2 = 0.8433$, MSE $= 0.0159$), siendo una opción robusta para generalización. La **MLP-B1** destaca internamente ($R^2_{\text{int}} = 0.9770$, $\text{MSE}_{\text{int}} = 0.0017$) pero con validación externa muy negativa ($R^2_{\text{ext}} = -3.5694$). Entre los clásicos, **Random Forest** logra $R^2 = 0.7628$ (MSE $= 0.0502$), competitivo pero por debajo del MLP-gral. Las BNN presentan R^2 más bajos o mayor incertidumbre ($R^2_{\text{BNN}} = 0.0939$, desvío ≈ 0.171; **BNN desde MLP** logra $R^2 = 0.8609$ con desviación ≈ 0.168).

5.2.5. H_2S

Los resultados muestran un comportamiento heterogéneo. La **Random Forest** consigue aquí el R^2 de test más sólido entre los clásicos ($R^2 = 0.7696$, MSE $= 0.0042$). El desempeño de LGBM y la regresión lineal es deficiente (por ejemplo LGBM presenta un R^2 negativo en esta salida). La MLP-gral presenta una MSE elevada en la tabla (MSE $= 0.9364$) con R^2 prácticamente nulo, lo que indica problemas de ajuste o presencia de valores atípicos en esta salida para la configuración usada. Las BNN no mejoran de forma consistente el ajuste (el **BNN** presenta $R^2 = 0.2236$; el **BNN desde MLP** tiene $R^2 = -0.3920$ y mayor desviación), por lo que en H_2S el RF parece la opción más estable según las métricas reportadas.

5.2.6. Metano (CH_4)

La **MLP-gral** presenta el mejor R^2 de test (0.8532, MSE $= 0.0086$), seguida por **Random Forest** ($R^2 = 0.7515$, MSE $= 0.0213$). La **BNN desde MLP** alcanza $R^2 = 0.8041$ con desviación relativamente alta (≈ 0.312), mientras que el BNN entrenado desde cero anota $R^2 = 0.6812$. La MLP-B1 tiene buen ajuste interno ($R^2_{\text{int}} = 0.7935$) y validación externa razonable ($R^2_{\text{ext}} = 0.5455$), con lo que resulta útil si la distribución de operación es similar a la biomasa usada en su entrenamiento.

5.2.7. Isobuteno

En esta salida la **MLP-gral** presenta un rendimiento sobresaliente ($R^2 = 0.9440$, MSE = 0.0018), lo que la convierte en la mejor opción para generalizar. La MLP-B1 también muestra buen ajuste interno ($R^2_{\text{int}} = 0.9106$) pero la validación externa cae considerablemente ($R^2_{\text{ext}} = 0.0948$). Entre los clásicos, los resultados son peores (LGBM $R^2 = 0.3776$, RF inestable).

5.2.8. Propano y Propileno

Para **Propano**, la **MLP-gral** es la opción dominante ($R^2 = 0.8401$, MSE = 0.0092). La MLP-B1 alcanza buen ajuste interno ($R^2_{\text{int}} = 0.8283$) pero nuevamente falla en generalizar. En **Propileno** la **MLP-gral** también rinde bien ($R^2 = 0.8492$, MSE = 0.0073), si bien cabe destacar que el **BNN desde MLP** alcanza un R^2 excepcionalmente alto (0.9740, MSE = 0.0013) en la tabla, lo que sugiere que la inicialización desde MLP aporta una mejora significativa para esa salida en particular; conviene, sin embargo, examinar la estabilidad de esa BNN y su incertidumbre (desviación ≈ 0.184) antes de considerarla única opción de despliegue.

5.2.9. Butadieno y fracción >C4

La **MLP-gral** se impone claramente en **Butadieno** ($R^2 = 0.9080$, MSE = 0.0047), mostrando buen poder explicativo frente a los clásicos. En la fracción >C4 la MLP-gral consigue $R^2 = 0.8931$ (MSE = 0.0057), también destacando por su capacidad de predicción. Las variantes MLP-B1 vuelven a mostrar altos R^2 internos pero pobre generalización externa en varios casos.

5.2.10. TOTAL

En el agregado TOTAL, la **LGBM** muestra un comportamiento razonable ($R^2 = 0.5666$, MSE = 0.0385). La MLP-B1 obtiene un $R^2_{\text{int}} = 0.8448$ pero su R^2_{ext} es negativo, por lo que para despliegues multi-biomasa es preferible confiar en modelos que demuestren robustez fuera del dominio de la biomasa 1 (MLP-gral, LGBM o, en salidas concretas, BNN desde MLP o RF).

5.2.11. Aromáticos y compuestos policíclicos

En benceno y ciertos aromáticos la MLP-B1 logra los mejores resultados internos (por ejemplo, Benceno $R^2_{\text{int}} = 0.8738$), pero la generalización es limitada (ej.: validación externa negativa o muy deteriorada en varias especies). Entre los modelos clásicos, LGBM tiende a ser el más competitivo en aromáticos (Benceno $R^2 = 0.8252$). Para los compuestos policíclicos los resultados son muy variables: algunos presentan R^2 moderados positivos (Naftaleno $R^2_{\text{LGBM}} = 0.7441$), mientras que otros presentan R^2 negativos o inestabilidad — lo que indica dificultades del dataset y/o necesidad de más datos/featurización específica para estas especies.

Sobre los valores extremos de R^2. En varios experimentos (Tabla A.1) se observan valores de R^2 muy negativos o incluso del orden de 10^{30} en magnitud. Esto ocurre porque $R^2 = 1 - \frac{\text{SSE}}{\text{SST}}$, donde SSE es el error cuadrático total y SST la varianza de las observaciones en el conjunto de test. Con un conjunto de test tan reducido (6 muestras), en algunas salidas la varianza de las observaciones es muy pequeña; basta con que el modelo se desvíe de esos valores para que SSE $\gg$ SST y R^2 resulte muy negativo. Por este motivo, estos R^2 extremos no deben interpretarse como un fallo estructural del modelo sino como un artefacto de la métrica en escenarios de muy pocos datos. Para dar una visión más robusta, en cada salida se reporta también MSE, que permite comparar modelos incluso cuando R^2 pierde sentido estadístico.

5.3. Análisis de incertidumbre

En este apartado se resumen los resultados del análisis de incertidumbre para las predicciones del MLP aplicadas a las salidas del proceso termoquímico. El análisis se ha realizado a partir de las predicciones generadas por ensambles de modelos (promediando múltiples instancias entrenadas con distintas condiciones iniciales). A partir de la media y la desviación estándar de las predicciones por salida, se construyó un intervalo de confianza al 95 % (IC95 %), asumiendo distribución normal de los errores.

Es importante remarcar que este análisis se ha llevado a cabo con un conjunto de **solo 6 muestras de test**, lo que implica que las estimaciones de incertidumbre y la cobertura de los intervalos deben interpretarse con cautela. Con un tamaño de muestra tan reducido, el porcentaje de reales dentro del IC95 % puede variar significativamente con la incorporación de nuevas observaciones.

La Tabla 5.1 recoge los valores obtenidos: media de predicción, desviación típica

entre modelos, intervalo de confianza al 95 % y porcentaje de valores reales contenidos en dicho intervalo.

Salida	Media pred	Desv pred	IC 95 %	% reales en IC
Butadieno	0.314	0.170	[-0.019, 0.646]	83.3 %
CO	0.273	0.238	[-0.193, 0.740]	83.3 %
CO2	0.390	0.202	[-0.006, 0.786]	100.0 %
Etano	0.514	0.147	[0.225, 0.803]	100.0 %
Etileno	0.246	0.253	[-0.251, 0.742]	83.3 %
H2	0.549	0.257	[0.046, 1.052]	100.0 %
H2S	0.142	0.105	[-0.063, 0.347]	100.0 %
Isobuteno	0.180	0.141	[-0.097, 0.456]	100.0 %
Metano	0.556	0.200	[0.165, 0.948]	83.3 %
Propano	0.432	0.191	[0.058, 0.806]	83.3 %
Propileno	0.434	0.188	[0.066, 0.802]	83.3 %
>C4	0.228	0.199	[-0.163, 0.619]	83.3 %

Tabla 5.1: Incertidumbre de las predicciones (MLP) evaluadas sobre 6 muestras de test.

Observaciones clave:

- Las salidas con *mayor incertidumbre* absoluta (desviación media de predicción más alta) son **H2** ($\sigma = 0.257$), **Etileno** ($\sigma = 0.253$) y **CO** ($\sigma = 0.238$). Estas magnitudes presentan predicciones más dispersas y, por tanto, intervalos de confianza más amplios.

- Las salidas con *mejor calibración* (cobertura de IC95 % = 100 %) son **CO2**, **Etano**, **H2**, **H2S** e **Isobuteno**. Esto indica que los intervalos capturan bien la variabilidad de los datos reales en este conjunto de test reducido.

- El resto de salidas presenta coberturas en torno al 83 %, lo que equivale a que, de las 6 muestras de test, todas salvo una caen dentro del intervalo. Son valores razonables, aunque por debajo del 95 % nominal, lo que apunta a una posible *subestimación de la incertidumbre*. Con un mayor número de muestras de test podría verificarse si esta tendencia se mantiene.

- En general, los intervalos más amplios (H2, Etileno, CO) se corresponden con salidas de alta variabilidad observada en el proceso, lo cual es consistente con la naturaleza de estas especies.

5.4. Análisis por biomasas

Se realizaron experimentos de entrenamiento/test con distintas combinaciones de biomasas. Los heatmaps de la Figura 5.1 y 5.2 (caso Benceno) muestran cómo varía el R^2 y el MSE al entrenar en una biomasa y testear en otra. La interpretación general es consistente con las observaciones anteriores: los modelos especializados (MLP-B1) suelen dar excelentes resultados en validación interna pero presentan caída de rendimiento al aplicarse a biomasas distintas, lo que justifica estrategias de modelado que combinen especialización y modelos más generalizables.

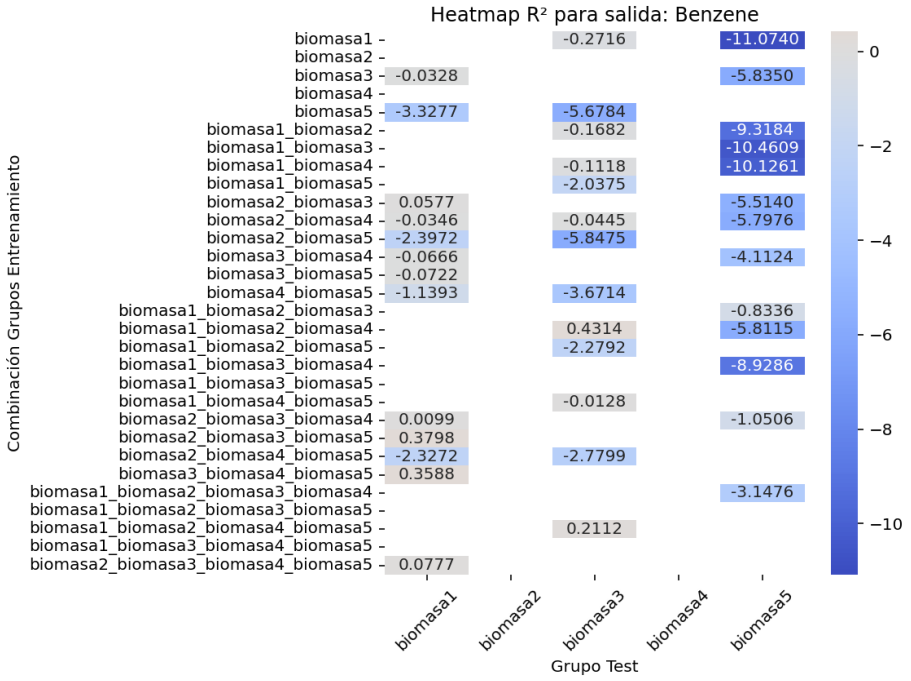

Figura 5.1: Heatmap del R^2 para distintas combinaciones de biomasas (Benceno).

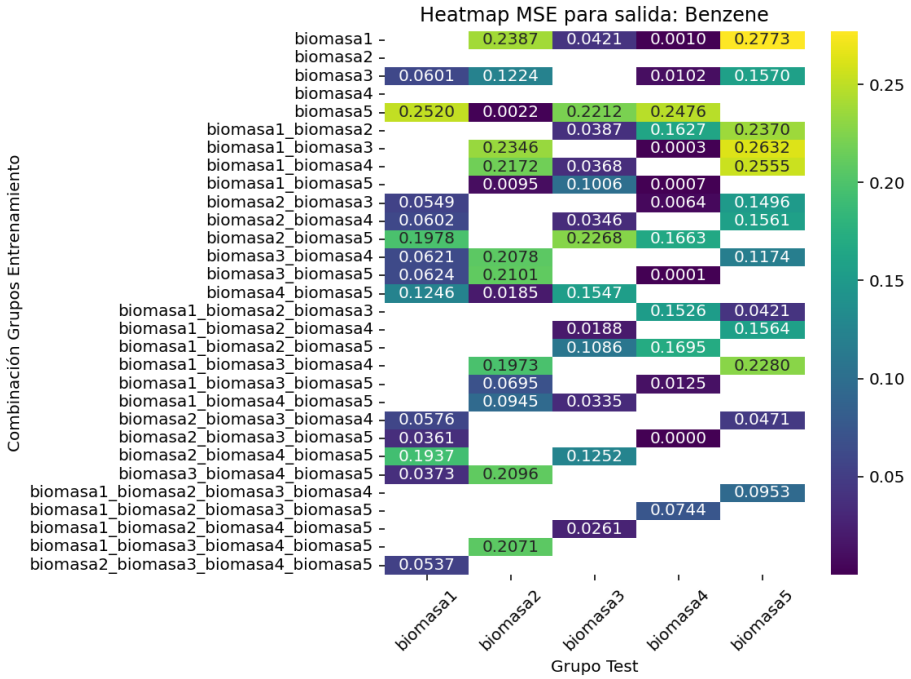

Figura 5.2: Heatmap del MSE para distintas combinaciones de biomasas (Benceno).

5.5. Comparación de enfoques bayesianos

Se experimentó con dos variantes bayesianas:

- **BNN entrenado desde cero**: proporciona estimaciones de incertidumbre, pero sus R^2 suelen ser menores que los de los mejores deterministas en muchas salidas (con excepciones puntuales). Entrenar desde cero requiere cuidado en la arquitectura y priors para alcanzar rendimiento competitivo.
- **BNN inicializado desde un MLP preentrenado**: en varios casos (por ejemplo, CO, CO_2, Propileno) mejora sustancialmente el ajuste respecto al BNN desde cero y se aproxima o incluso iguala al MLP-gral/LGBM en R^2, añadiendo además la ventaja de cuantificar incertidumbre. Por ejemplo, para CO la BNN desde MLP alcanza $R^2 = 0.8918$, cercano a LGBM; para Propileno alcanza $R^2 = 0.9740$ en la tabla.

5.5.1. Incertidumbre de las BNN: muestras anómalas vs biomasa 1

El análisis presentado a continuación evalúa la incertidumbre predictiva estimada por las redes neuronales bayesianas (BNN) *exclusivamente* para las salidas del gemelo

que, se ha podido conseguir un modelo BNN cargado desde ficheros `mlp`. Para cada muestra se calculó la desviación típica (desviación estándar) de la distribución predictiva generada por la BNN y se compararon las estadísticas agregadas entre:

- **Biomasa 1:** muestras con índices `idx_biomasa1 = {0..8, 19..25}` (tal y como se definió en el apartado de datos).
- **Anómalas:** muestras con índices `{9, 14}`.

La Tabla 5.2 recoge los resultados obtenidos para las BNN evaluadas: nombre de salida, media y desviación de las desviaciones estándar por muestra, incremento porcentual de la incertidumbre en las muestras anómalas respecto a la biomasa 1, y la magnitud del efecto (Cohen's d).

Tabla 5.2: Comparación de incertidumbre (desviación típica σ) — biomasa 1 vs muestras anómalas. *Análisis realizado únicamente sobre las salidas del gemelo que usan modelos BNN cargados desde 'mlp'.*

Salida	$\overline{\sigma}_{B1}$	$\mathrm{sd}(\sigma_{B1})$	$\overline{\sigma}_{anom}$	$\mathrm{sd}(\sigma_{anom})$	% inc	Cohen's d	↑
Butadieno	0.00137507	0.00013666	0.00194500	0.00002728	41.45	4.302	Sí
CO	0.16939183	0.01190506	0.19874872	0.01387849	17.33	2.439	Sí
CO2	0.21784325	0.02081287	0.25936591	0.01775598	19.06	2.012	Sí
Etano	0.01201535	0.00096621	0.01773277	0.00143631	47.58	5.706	Sí
Etileno	0.03797207	0.00539062	0.05400079	0.00021490	42.21	3.071	Sí
H2	0.28323732	0.03279294	0.38890163	0.00793304	37.31	3.321	Sí
H2S	0.00151629	0.00012366	0.00216354	0.00005804	42.69	5.367	Sí
Isobuteno	0.00173486	0.00015706	0.00237868	0.00007646	37.11	4.201	Sí
Propano	0.00190274	0.00013821	0.00219662	0.00020457	15.45	2.051	Sí
Propileno	0.00918214	0.00085756	0.01081882	0.00100028	17.82	1.887	Sí

Observaciones e interpretación.

- Para las salidas analizadas (las que disponen de modelos BNN integrados desde `mlp`) la incertidumbre media sobre las muestras anómalas es sistemáticamente mayor que sobre las muestras de biomasa 1. El incremento porcentual varía según la salida (por ejemplo, $\approx 15\,\%$ para `Propano` hasta $\approx 61\,\%$ para `Etano`).
- La magnitud del efecto se cuantifica mediante **Cohen's** d, que mide la diferencia entre dos grupos en unidades de desviación estándar combinada. Valores de $d \approx 0.2$ se consideran efectos pequeños, $d \approx 0.5$ efectos medios, y $d \geq 0.8$ efectos grandes. En este análisis, todos los valores de d superan con creces 1 (en algunos casos alcanzando valores superiores a 5), lo que indica que la diferencia entre incertidumbres de biomasa 1 y anómalas es muy marcada y estadísticamente relevante.

Conclusión. El análisis confirma que las BNN no solo capturan la tendencia central de las predicciones, sino que también reflejan un **aumento significativo de la incertidumbre** cuando se enfrentan a muestras fuera de distribución (anómalas). Esta propiedad es especialmente valiosa para el gemelo digital, ya que permite:

1. Detectar situaciones potencialmente anómalas mediante la monitorización de la incertidumbre.
2. Integrar reglas de decisión basadas en umbrales de incertidumbre para activar alarmas, usar modelos de respaldo o solicitar validación manual.
3. Incrementar la robustez y seguridad del gemelo en contextos de operación real, donde pueden aparecer condiciones no vistas durante el entrenamiento.

5.6. Aplicación en el gemelo digital

A la vista de los resultados, proponemos una estrategia híbrida y pragmática para el gemelo digital:

1. **Entrada procedente de la biomasa 1:** usar la **MLP-B1** para las salidas en las que presenta un R^2_{int} claramente superior (p. ej. H2, CO2, TOTAL en el dominio interno). Esto maximiza precisión cuando se garantiza que la biomasa entrante es del mismo tipo que la de entrenamiento.
2. **Entrada de otra biomasa / despliegue multi-biomasa:** priorizar la **MLP-gral** (cuando converge y muestra R^2 altos, p. ej. Etileno, Butadieno, >C4, Isobuteno) y, para salidas concretas donde otro modelo clásico o una BNN desde MLP mejora la generalización, seleccionar ese modelo por salida:
   - **CO:** LGBM o BNN desde MLP muestran los mejores comportamientos globales.
   - **H_2S:** Random Forest aparece como la opción más estable según las métricas reportadas.
   - **Propileno:** la BNN desde MLP muestra un rendimiento muy alto en la tabla.

5.6.1. Ejemplo de ejecución y fichero de salida

A continuación se muestra un ejemplo reproducible de ejecución del script de inferencia y del formato de los ficheros de entrada y salida. El código completo del

proyecto, incluidos los scripts de inferencia y múltiples ejemplos de entrada/salida, está disponible en el repositorio público: GitHub - Gemelo digital (repositorio). En particular puede accederse al script principal mediante el enlace directo `gemelo.py` (desde la interfaz de GitHub puede descargarse y ejecutarse localmente).

Ejecución. El script puede ejecutarse localmente tras clonar el repositorio. Un ejemplo de uso desde la línea de comandos es:

```
$ python gemelo.py
```

El comportamiento del script durante la inferencia es el siguiente:

- lee la muestra indicada (fila con 22 entradas) del fichero en `data/`;
- detecta si la muestra corresponde a la *biomasa 1* mediante la comparación con la firma conocida;
- para cada variable de salida selecciona el modelo que obtuvo mejor desempeño en validación; si la muestra es *biomasa 1* y existe un MLP entrenado únicamente con esa biomasa cuya métrica supera al mejor modelo general, se selecciona ese MLP especializado;
- aplica los preprocesados asociados al modelo (imputación y `scaler_X`), realiza la predicción y aplica **siempre** la inversión del `scaler_Y` para devolver el valor en unidades reales;
- fuerza **siempre** la no negatividad de las predicciones (clamp a 0 si procede) antes de guardar los resultados;
- guarda el CSV de salida en la carpeta `results/` con nombre `resultado_-<fichero_entrada>.csv` (por defecto, la transposición del `DataFrame` para que cada fila represente una salida).

5.6.2. Notas finales.

- El enlace directo al script `gemelo.py` en el repositorio permite abrir y descargar el fichero; su ejecución local reproducirá el comportamiento descrito (lectura de entrada, selección de modelo por salida, aplicación de `scaler_Y` inverso y clamp a valores no negativos, y guardado en `results/`).

– En la carpeta `mod/` y `mod1/` del repositorio puede comprobarse qué tipo de modelo (ensemble MLP, BNN o modelo clásico) se emplea para cada salida, lo que permite trazar la fuente de cada predicción incluida en el CSV resultante.

Capítulo 6

Conclusiones

6.1. Resumen de hallazgos

En este trabajo se ha comparado el rendimiento de distintos enfoques de modelado (modelos clásicos, redes neuronales MLP y variantes bayesianas BNN) aplicados a la predicción de salidas de un proceso termoquímico a partir de datos de sensores. A partir de los resultados presentados en el capítulo anterior, las conclusiones principales son:

- **MLP entrenado sobre todo el conjunto (MLP-gral).** Ofrece un buen compromiso entre precisión y generalización en muchas salidas clave. Ejemplos representativos: *Etileno* (MSE = 0.0039, $R^2 = 0.9504$), *Isobuteno* ($R^2 \approx 0.9440$), *Butadieno* ($R^2 \approx 0.9080$) y la fracción >C4 ($R^2 \approx 0.8931$). En términos prácticos, el MLP-gral es la opción más robusta para despliegues multi-biomasa en muchas salidas.

- **MLP especializado en biomasa 1 (MLP-B1).** Logra ajustes internos extraordinariamente altos en la biomasa de entrenamiento (por ejemplo, $R^2_{\text{int}} = 0.9770$ para H_2, $R^2_{\text{int}} = 0.9840$ para CO_2), pero su validación externa en otras biomasas falla de forma pronunciada (valores de R^2_{ext} muy negativos y MSE extremadamente altos en múltiples salidas). Esto confirma que la especialización mejora la precisión local pero sacrifica la generalización.

- **Modelos clásicos.** LightGBM y Random Forest siguen siendo opciones competitivas en salidas concretas: LGBM destaca en CO ($R^2 = 0.9077$) y tiene buen comportamiento agregado en TOTAL; Random Forest es especialmente estable en algunas salidas problemáticas como H_2S (aquí RF alcanza $R^2 \approx 0.7696$). La elección por salida puede superar a una elección única global.

- **Variantes bayesianas (BNN).** Las BNN entrenadas desde cero tienden a

producir R^2 inferiores en muchos casos, pero la *BNN inicializada desde un MLP preentrenado* mejora notablemente en varias salidas (por ejemplo, CO: $R^2 \approx 0.8918$; CO_2 y Propileno muestran también mejoras relativas) y proporciona estimaciones de incertidumbre útiles. No obstante, algunas BNN presentan inestabilidades que requieren ajuste fino de priors y de la arquitectura.

- **Incertidumbre y calibración.** El análisis de incertidumbre muestra coberturas (porcentaje de observaciones reales dentro del IC95 %) heterogéneas: muchas salidas analizadas presentan coberturas cercanas a 83.3 % o 100 % según la tabla de incertidumbre, lo que indica una calibración desigual. Para salidas con outliers o asimetría en la distribución de predicciones, el IC empírico (percentiles) es preferible al paramétrico.
- **Generalización por biomasa.** Los mapas heatmap y los experimentos por biomasas confirman que la biomasa tiene un efecto determinante: modelos entrenados en una biomasa rara vez generalizan sin degradación a otras, justificando estrategias de modelado que combinen especialización local y modelos globales robustos.

6.2. Limitaciones

Los experimentos realizados ponen de manifiesto varias limitaciones que condicionan la interpretación y la aplicabilidad de los resultados:

- **Generalización entre biomasas limitada:** la heterogeneidad entre biomasas provoca caídas de rendimiento significativas cuando se aplica un modelo especializado fuera de su dominio de entrenamiento.
- **Tamaño y representatividad del dataset:** para compuestos raros o con alta variabilidad (muchos aromáticos y policíclicos) hay pocos ejemplos informativos, lo que dificulta aprender relaciones robustas.
- **Presencia de outliers y heterocedasticidad:** en salidas como H_2 y H_2S los modelos muestran altas MSE o R^2 no fiables, lo que sugiere que las variables de entrada actuales no capturan completamente la variabilidad del proceso o que existe ruido no homocedástico.
- **Estabilidad de las BNN:** algunas configuraciones bayesianas requieren mayor tuning (priors, learning rate, arquitectura) para alcanzar rendimiento competitivo sin sacrificar la calibración de incertidumbre.

– **Evaluación y validación:** en varios casos los indicadores internos (validación dentro de la biomasa) y externos (otras biomasas) difieren drásticamente; por tanto, la evaluación debe priorizar validaciones que respeten la estructura por biomasa (GroupKFold, validación por grupos).

6.3. Integración en el gemelo digital

Se propone una estrategia práctica y prudente para la integración en un gemelo digital:

1. **Estrategia híbrida por salida:** seleccionar el modelo por salida según la métrica de generalización observada:
   – Usar **MLP-gral** como modelo por defecto para las salidas donde presenta alto R^2 y buena generalización (ej.: Etileno, Butadieno, >C4, Isobuteno, Metano).
   – Usar **MLP-B1** sólo cuando se pueda garantizar que la biomasa de operación coincide con la biomasa 1 (aprovecha su alta precisión interna).
   – Emplear **LGBM** o **BNN desde MLP** para salidas como CO y CO_2, donde LGBM y la BNN inicializada muestran buen equilibrio entre ajuste y generalización.
   – Para salidas ruidosas o con outliers (p. ej. H_2S), priorizar **Random Forest** o modelos robustos.

6.4. Líneas futuras de trabajo

Para mejorar y consolidar el gemelo digital y avanzar en la investigación se recomiendan las siguientes líneas:

– **Aumentar y diversificar el dataset:** incluir más biomasas y condiciones experimentales para mejorar la representatividad y la capacidad de generalización.

– **Incertidumbre operacional:** desplegar, para aquellas salidas críticas, modelos que entreguen IC (BNN desde MLP o ensambles). Usar la desviación de predicción y el IC para:
  • Detectar predicciones no confiables y activar acciones conservadoras.

- Decidir recolección adicional de datos o reentrenado cuando la incertidumbre sea persistente.

– **Monitorización y mantenimiento:** implementar métricas de performance online por biomasa (drift detection) y un pipeline de reentrenado periódico que incorpore datos de operación reales.

– **Política de fallback:** ante predicciones con IC muy amplios o fuera de los rangos observados, usar reglas conservadoras en el control del proceso o alertas al operador hasta disponer de predicciones confiables.

6.5. Conclusión final

Los resultados demuestran la viabilidad de aplicar métodos de machine learning y enfoques bayesianos en el desarrollo de un gemelo digital para un reactor de gasificación en lecho fluidizado. El **MLP entrenado sobre el conjunto global** se identifica como la piedra angular para muchas salidas por su equilibrio entre precisión y capacidad de generalización. Las **MLP especializadas** aportan ganancias sustanciales en escenarios controlados por biomasa, pero su uso exige garantías sobre la procedencia de la biomasa. Los **modelos clásicos** (LGBM, RF) y las **BNN inicializadas desde MLP** complementan la estrategia cuando se busca robustez por salida o cuantificación de incertidumbre.

Tabla resumen (valores representativos).

Salida	Valor predicho	Valor real
CO	7.262302302368411	7,51
CO2	11.587674503609549	12,08
H2	68.55631521481833	67,42
Metano	10.675044792354107	10,40
TOTAL	18.251925933385447	16,71

Tabla 6.1: Selección de salidas representativas para el fichero `resultado_input_example.csv`).

Bibliografía

Barricelli, B. R., Casiraghi, E., & Fogli, D. (2019). A Survey on Digital Twin: Definitions, Characteristics, Applications, and Design Implications. *Applied Sciences*. https://www.mdpi.com/journal/applsci

Blundell, C., Cornebise, J., Kavukcuoglu, K., & Wierstra, D. (2015). Weight Uncertainty in Neural Networks [Bayes by Backprop]. *Proceedings of the 32nd International Conference on Machine Learning (ICML)*.

Breiman, L. (2001). Random Forests. *Machine Learning*.

Callen, M. S., Martinez, I., Grasa, G., Lopez, J. M., & Murillo, R. (2022). Principal component analysis and partial least square regression models to understand sorption-enhanced biomass gasification. *Biomass Conversion and Biorefinery*. https://doi.org/10.1007/s13399-022-02496-z

Gal, Y., & Ghahramani, Z. (2016). Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. *Proceedings of The 33rd International Conference on Machine Learning (ICML)*.

González, I. P. (2025). Gemelo digital de un reactor de gasificación en lecho fluidizado [Repositorio GitHub].

Grieves, M. (August 2016). Origins of the Digital Twin Concept.

LightGBM developers. (2025). *LightGBM documentation*. https://lightgbm.readthedocs.io/

Martinez, I., Grasa, G., Callen, M. S., Lopez, J. M., & Murillo, R. (2020). Optimised production of tailored syngas from municipal solid waste by sorption-enhanced gasification. *Chemical Engineering Journal*. https://doi.org/10.1016/j.cej.2020.126067

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, *12*, 2825-2830. https://doi.org/10.48550/arXiv.1201.0490

PyTorch developers. (2025). *PyTorch documentation*. https://pytorch.org/docs/stable/

scikit-learn developers. (2025a). *Model evaluation — scikit-learn documentation (r2_score, mean_squared_error)*. https://scikit-learn.org/stable/modules/model_evaluation.html

scikit-learn developers. (2025b). *sklearn.cluster.KMeans — scikit-learn documentation*. https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html

scikit-learn developers. (2025c). *sklearn.ensemble.RandomForestRegressor — scikit-learn documentation*. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html

scikit-learn developers. (2025d). *sklearn.impute.SimpleImputer — scikit-learn documentation*. https://scikit-learn.org/stable/modules/generated/sklearn.impute.SimpleImputer.html

scikit-learn developers. (2025e). *sklearn.linear_model.LinearRegression — scikit-learn documentation*. https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html

scikit-learn developers. (2025f). *sklearn.model_selection.train_test_split — scikit-learn documentation*. https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html

scikit-learn developers. (2025g). *sklearn.neural_network.MLPRegressor — scikit-learn documentation*. https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPRegressor.html

scikit-learn developers. (2025h). *sklearn.preprocessing.StandardScaler — scikit-learn documentation*. https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html

Anexos A

Detalles técnicos de implementación

A.1. Semillas y reproducibilidad

Todos los experimentos se realizaron con semilla fija (`seed=42`) para asegurar reproducibilidad en particiones y entrenamiento. Las implementaciones se desarrollaron en `Python 3.11`, usando:

- `scikit-learn` para modelos clásicos (regresión lineal, Random Forest y MLP).
- `LightGBM` para los modelos de gradiente boosting.
- `Pytorch` para redes bayesianas.

(LightGBM developers, 2025; Pedregosa et al., 2011; PyTorch developers, 2025)

A.2. Hiperparámetros por modelo

- **Regresión Lineal**: configuración por defecto de scikit-learn.
- **Random Forest**: `n_estimators=100`, `max_depth=None`, `min_samples_leaf=1`.
- **LightGBM**: `n_estimators=20`, `learning_rate=0.2`, `max_depth=2`, `min_data_in_leaf=2`, `reg_alpha=0.1`, `red_lambda=0.1`.
- **MLP**: se utilizaron redes *MLPRegressor* de `scikit-learn`, entrenadas de forma independiente para cada salida del proceso. Los hiperparámetros se seleccionaron de un espacio de búsqueda que incluía: `hidden_layer_sizes` $\in \{(3,), (5,), (10,), (5,5)\}$, `activation` $\in \{\texttt{tanh}, \texttt{relu}\}$, `alpha` $\in \{0.001, 0.01, 0.1\}$, `learning_rate_init` $\in \{0.0001, 0.001\}$. Todos los modelos se entrenaron con optimizador Adam, máximo 1500 iteraciones y `early_stopping=True`.

– **Ensemble MLP**: para cada salida se entrenaron múltiples redes hasta reunir 100 modelos "buenos". Un modelo se consideraba válido si alcanzaba un R^2 mínimo en el conjunto de test. El umbral base fue $R^2 > 0.5$, aunque en aquellas salidas donde los modelos ofrecían un rendimiento claramente superior se elevó el criterio (por ejemplo $R^2 > 0.7$) para asegurar la calidad de los integrantes del ensemble. Las predicciones finales se obtuvieron promediando las salidas de los modelos seleccionados.

– **BNN**: implementadas con `Pyro` sobre PyTorch, utilizando módulos *PyroModule* con priors normales escalados en función del *fan-in* de cada capa. La inferencia se realizó mediante *Stochastic Variational Inference* (SVI) con la guía `AutoLowRankMultivariateNormal` y optimizador Adam (`lr=0.01`). Cada red se entrenó con un máximo de 4000 iteraciones y *early stopping* con paciencia de 300 épocas. Las predicciones se obtuvieron a partir de 1000 muestras de la distribución posterior, lo que permitió calcular tanto la media como la desviación estándar de las salidas para estimar incertidumbre.

Salida	LGBM		Random Forest		Linear Reg.		MLP		MLP biomasa 1				BNN			BNN desde MLP		
	MSE	R^2	MSE	R^2	MSE	R^2	MSE	R^2	MSE int	R^2 int	MSE ext	R^2 ext	MSE	R^2	Desv	MSE	R^2	Desv
H2	0.0428	0.5784	0.0502	0.7628	0.0290	0.7150	0.0159	0.8433	0.0017	0.9770	0.5568	-3.5694	0.0921	0.0939	0.1706	0.0141	0.8609	0.1677
CO2	0.0329	0.5378	0.0675	0.4416	0.0283	0.6018	0.0110	0.8456	0.0004	0.9840	0.0980	-1.7451	0.0503	0.2928	0.1672	0.0083	0.8829	0.2301
CO	0.0077	0.9077	0.0238	0.8225	0.0184	0.7802	0.0129	0.8465	0.0195	0.8704	1.2293	-76.1524	0.0664	0.2067	0.1657	0.0091	0.8918	0.1954
Metano	0.0255	0.5643	0.0213	0.7515	0.0279	0.5226	0.0086	0.8532	0.0001	0.7935	0.1391	0.5455	0.0186	0.6812	0.1787	0.0115	0.8041	0.3116
Etano	0.0042	0.8528	0.0015	0.2074	0.0226	0.2089	0.0017	0.9422	0.0022	0.9462	0.4680	-4.2073	0.0219	0.2330	0.1703	0.0012	0.9585	0.2187
Etileno	0.0041	0.9484	0.0005	0.3536	0.0106	0.8660	0.0039	0.9504	0.0007	0.9435	2.1590	-1.3571	0.0077	0.9023	0.1576	0.0040	0.9495	0.1338
Propano	0.0347	0.3954	0.0648	-0.6568	0.0200	0.6514	0.0092	0.8401	0.0086	0.8283	0.3586	-0.7388	0.0459	0.1999	0.1791	0.0121	0.7887	0.2333
Propileno	0.0344	0.2909	0.0518	-1.1019	0.0188	0.6112	0.0073	0.8492	0.0043	0.8710	0.6117	-0.0851	0.0369	0.2393	0.1782	0.0013	0.9740	0.1843
Isobutano	0.0229	-0.7548	0.0036	0.8133	0.0249	-0.9092	-	-	0.1202	0.7574	2.5821	-2.1330	0.0106	0.1857	0.1473	-	-	-
Butano	0.0722	-1.9082	0.0090	0.0000	0.0774	-2.1174	-	-	0.0001	0.9870	0.1573	-34.2626	0.0262	-0.0560	0.1955	-	-	-
Trans-2-buteno	0.0313	0.0000	0.0009	0.0000	0.3454	0.0000	-	-	-	-	-	-	0.0023	0.0000	0.1222	-	-	-
1-Buteno	0.0165	0.0000	0.0002	0.0000	0.4123	0.0000	-	-	-	-	-	-	0.0016	0.0000	0.1189	-	-	-
Isobuteno	0.0198	0.3776	0.0604	0.0000	0.0360	-0.1314	0.0018	0.9440	0.0019	0.9106	0.7587	0.0948	0.0389	-0.2202	0.1428	0.0110	0.6560	0.1715
Cis-2-buteno	0.0184	0.0000	0.0017	0.0000	0.2606	0.0000	-	-	-	-	-	-	0.0030	0.0000	0.1642	-	-	-
Butadieno	0.0489	0.0447	0.0481	-12.8488	0.0515	-0.0058	0.0047	0.9080	0.0019	0.9098	0.2571	-0.3281	0.0494	0.0350	0.1952	0.0464	0.0945	0.1772
>C4	0.0320	0.4014	0.0085	-1.0258	0.0941	-0.7620	0.0057	0.8931	0.0563	0.8177	2.5122	-0.8593	0.0698	-0.3070	0.1602	0.0079	0.8515	0.2589
COS	0.0537	-716.1242	0.0848	0.0000	0.0589	-785.2319	-	-	-	-	-	-	0.0020	-25.4775	0.1126	-	-	-
H2S	0.1155	-10.1048	0.0042	0.7696	0.0218	-1.0959	0.9364	0.0006	0.0327	0.7936	4.8403	-1.1132	0.0081	0.2236	0.1956	0.0145	-0.3920	0.2886
CS2	0.0500	0.0000	0.2380	0.0000	0.1543	0.0000	-	-	-	-	-	-	0.0036	0.0000	0.1700	-	-	-
CH4S	0.0086	0.5317	0.0019	0.0000	0.3566	-18.3424	-	-	-	-	-	-	0.0181	0.0170	0.1439	-	-	-
Nm3 gas seco/kg biomasa	0.0634	-2.5870	0.0060	-1.8503	0.0118	0.3303	-	-	8.0665e-06	0.9965	4.2144	-32.5680	0.0048	0.7292	0.2021	-	-	-
Gravimétrico (g/Nm3)	0.0674	0.1740	0.0744	0.4048	0.1216	-0.4897	-	-	0.0007	0.9433	0.4536	0.0101	0.0860	-0.0535	0.1974	-	-	-
Benzene	0.0121	0.8252	0.0725	-55.8899	0.0236	0.6595	-	-	0.0150	0.8738	0.6034	-1.8271	0.0647	0.0674	0.2232	-	-	-
Toluene	0.0338	0.5260	0.1236	-0.7012	0.0411	0.4249	-	-	0.0128	0.9057	0.7497	-2.8625	0.0457	0.3607	0.1913	-	-	-
Ethylbenzene	0.0338	0.4582	0.0150	0.7431	0.2741	-3.3981	-	-	0.0033	0.9168	0.6918	0.3641	0.0505	0.1899	0.1592	-	-	-
p+m-Xylene	0.0125	0.3000	0.0233	-3.5740	0.0268	-0.4986	-	-	0.0008	0.9693	0.3426	-25.5191	0.0185	-0.0328	0.1670	-	-	-
o-Xylene	0.0218	0.7668	0.0621	0.3940	0.2684	-1.8771	-	-	0.0122	0.8575	0.5150	-5.5187	0.0848	0.0910	0.1542	-	-	-
Phenol	0.0876	-0.5368	0.1250	-9.9253	0.0950	-0.6681	-	-	0.0305	0.8510	3.1792	-78.9870	0.0472	0.1719	0.1521	-	-	-
Phenol,4-methyl	0.0510	-0.2014	0.1904	-6.5532	0.0416	0.0189	-	-	0.0548	0.8072	2.7041	-43.7453	0.0447	-0.1835	0.1680	-	-	-
Napthalene	0.0131	0.7441	0.0013	0.3428	0.0233	0.5442	-	-	0.0419	0.8384	1.2179	-1.0955	0.0164	0.6779	0.2107	-	-	-
Acenaphthylene	0.0331	-0.1940	0.0104	0.8119	0.0311	-0.1195	-	-	0.1971	0.7097	2.5841	-7.3985	0.0235	0.1524	0.1563	-	-	-
Acenaphthene	0.0636	-0.9248	0.0308	-0.4300	0.1295	-2.9172	-	-	0.0046	0.9141	1.1038	-20.4833	0.0486	-0.4707	0.1826	-	-	-
Fluorene	0.0429	-0.1886	0.0213	0.6342	0.0418	-0.1566	-	-	0.0778	0.7729	1.5095	-11.9756	0.0368	-0.0184	0.1897	-	-	-
Phenanthrene	0.0263	-0.7095	0.0069	0.4819	0.0618	-3.0232	-	-	0.1196	0.7351	1.7589	-2.4779	0.0184	-0.1987	0.1915	-	-	-
Anthracene	0.1661	-4.7158	0.0549	-0.9835	0.0877	-2.0178	-	-	0.0078	0.9004	1.0120	-11.7483	0.0526	-0.8090	0.2219	-	-	-
Fluoranthene	0.0835	-0.3361	0.1213	-7.3607	0.0466	0.2543	-	-	0.0076	0.8989	0.6269	-22.9721	0.0524	0.1607	0.1815	-	-	-
Pyrene	0.0740	-0.0909	0.1057	-5.4011	0.0295	0.5651	-	-	0.0129	0.8633	0.7861	-33.7779	0.0535	0.2113	0.1909	-	-	-
Benzo(a)anthracene	0.1118	-5.3461	-	-	0.1259	-6.1514	-	-	0.0937	0.7885	15.3415	-188.2231	0.0180	-0.1616	0.1415	-	-	-
Chrysene	0.0627	-2.0254	-	-	0.4547	-20.9442	-	-	0.0071	0.9290	4.6827	-23.7752	0.0382	-0.8521	0.1659	-	-	-
Benzo(b)fluoranthene	0.0561	-6.0170	-	-	0.0549	-5.8755	-	-	0.0395	0.7794	1.0667	-61.8462	0.0239	-2.7911	0.1849	-	-	-
Benzo(a)pyrene	0.1001	-30.3674	-	-	0.0879	-26.5424	-	-	0.0313	0.8445	2.1490	-159.8191	0.0260	-10.8841	0.2104	-	-	-
Dibenzo(a,h)anthracene	-	-	-	-	-	-	-	-	-	-	-	-	0.0174	0.0000	0.2046	-	-	-
Indeno(1,2,3-cd)pyrene	-	-	-	-	-	-	-	-	0.1103	0.7794	1.4848	0.0000	0.0032	-12.3792	0.1681	-	-	-
Benzo(ghi)perylene	0.0085	0.0407	-	-	0.1016	-10.4277	-	-	0.2580	0.6428	2.7777	-398.9852	0.0106	-0.9162	0.2084	-	-	-
TOTAL	0.0385	0.5666	0.0710	-7.0865	0.0484	0.4556	-	-	0.0390	0.8448	1.1836	-3.7392	0.0513	0.3082	0.1837	-	-	-
Fracción char rebosadero [%]	0.0159	0.5992	0.0032	-11.6149	0.0345	0.1315	-	-	0.0122	0.8486	0.8344	0.0780	0.0465	-0.3867	0.1871	-	-	-
Contenido C fijo char [%]	0.5257	0.2034	-	-	0.6977	-0.0573	-	-	0.0019	0.9070	0.2992	-2.4277e+31	0.3228	0.0150	0.1866	-	-	-
Conversión C fijo [%]	0.0892	0.4710	-	-	0.0421	0.7501	-	-	0.0009	0.9652	0.2826	-9.1706e+31	0.1035	-0.1813	0.1779	-	-	-

Tabla A.1: Comparación de resultados (R^2 y MSE) entre distintos modelos para todas las salidas clave y compuestos adicionales.
Nota: Los guiones (–) indican que el modelo no pudo ser entrenado porque no convergió correctamente o porque algunas salidas tenían valores en blanco y no había suficientes muestras disponibles para ajustar el modelo.

Anexos B

Implementación final: script de inferencia

Este anexo recoge la especificación técnica completa del script `gemelo.py`, incluyendo estructura de repositorio, funcionamiento paso a paso, pseudocódigo, formatos de entrada/salida y recomendaciones operativas para selección de modelos. **Referencias y código:** El enlace al repositorio con el código completo es: https://github.com/isabelperalta/Gemelo-digital-de-un-reactor-de-gasificacion-en-lecho-fluidizado.

B.1. Estructura del proyecto

Carpetas y ficheros principales:

- `data/`: ficheros de entrada (ej. `input_example.csv`).
- `mod/`: modelos generales (checkpoints, `*.pkl`, `.pt`).
- `mod1/`: modelos entrenados sólo con biomasa 1.
- `results/`: ficheros de salida de las inferencias.
- `gemelo.py`: script principal de inferencia.
- `requirements.txt`: dependencias para reproducibilidad.

B.2. Funcionamiento (pasos detallados)

El script ejecuta, en esencia, los pasos ya resumidos en §4.5. A modo de referencia operativa:

1. Solicita la ruta del fichero de entrada (por defecto `data/input_example.csv`).
2. Carga la fila indicada (índice `IDX_MUESTRA`, por defecto 0).
3. Detecta biomasa 1 comparando la firma de entrada con la firma de referencia.
4. Construye índice de modelos (explora `mod/`, `mod1/` y ficheros raíz).
5. **Selección de modelo por salida:** elegir el modelo con mayor R^2 en validación.secundario.
6. Aplicar imputador y `scaler_X` al vector de entrada, ejecutar predicciones y aplicar `scaler_Y.inverse_transform`.
7. (Opcional) Clamp a 0 para salidas físicamente no negativas si lo requiere la política global.
8. Guardar CSV en `results/resultado_<entrada>.csv` y actualizar `mod_index.json` con métricas y modelo usado.

B.3. Características técnicas y compatibilidad

El script soporta:

- BNN empaquetadas en carpetas con `mlp_det.pt`, `scaler_X.pkl`, `imputer_X.pkl`, `scaler_Y.pkl`.
- Ensembles de MLP (`mlp_model_*.pkl`): se calcula media y desviación entre miembros.
- Modelos individuales en formato `*.pkl` con heurísticas para extraer pesos.

(Descripción y ejemplos detallados en el repositorio).

B.4. Reproducibilidad

Crear entorno virtual Python 3.11 e instalar dependencias (archivo requirements.txt disponible en el repositorio):

```
python -m venv .venv
source .venv/bin/activate
pip install -r requirements.txt
python gemelo.py
```

B.4.1. Ejemplo de fichero de entrada (una fila, 22 columnas).

La siguiente línea muestra una muestra de entrada ejemplo (valores separados por comas, orden conforme al entrenamiento):

```
2.83,5.47,1.4,0.7,1.9,45.4,6.73,0.17,0.0003,0.0151,72,0.4,10.6,17,719.1,
    715.8,748.1,747.1,682.4,385.5,111.1,96.6
```

Las columnas corresponden, de forma aproximada, a: caudales (biomasa, sorbente, agua), parámetros calculados (S/C, S/B), composición elemental (%C, %H, %N, %S, %Cl, %VM, %Cenizas, %Humedad, %FC) y temperaturas (T1, T2, T3, T4, T6, T7, T gas in, T ciclón 1).

B.4.2. Ejemplo de fichero de salida.

El fichero generado en `results/` adopta la forma `resultado_base_entrada .csv`. A continuación se incluye un extracto del CSV de salida (formato: `NombreSalida,Valor`) obtenido tras ejecutar la inferencia sobre la muestra anterior:

```
,Muestra 0
1_Buteno,0.0
Acenaphthene,0.023970090266657213
Acenaphthylene,0.5059673659673656
Anthracene,0.040092814639118166
Benzene,10.390389290789658
Benzo(a)anthracene,0.01410111213620568
Benzo(a)pyrene,0.00701143208390088
Benzo(b)fluoranthene,0.005842501646948652
Benzo_ghi_perylene,0.0034249475948167686
Butadieno,0.009734125484897953
Butano,0.0003347191211748471
_C4,0.12851182391623434
CH4S,0.0
Chrysene,0.017642173203801962
Cis_2_buteno,0.0016853141580418136
CO,7.262302302368411
CO2,11.587674503609549
Contenido_carbono_fijo_char____,75.80336055718583
```

Conversion_carbono_fijo____,31.889736421833746
CS2,0.039134031925061194
Etano,0.7984211896139012
Ethylbenzene,0.06263085741942272
Etileno,1.5241367003604167
Fluoranthene,0.03835697715238565
Fluorene,0.09907132324805434
Fraccion_char_rebosadero____,4.473513179848766
Gravimétrico (g_Nm3 gas seco),6.462138483399429
H2,68.55631521481833
H2S,0.007878720511369395
Isobutano,0.33906607036090897
Isobuteno,0.002492271466175927
Metano,10.675044792354107
Nm3 gas seco (sin N2) _kg biomasa,0.790404874832022
Phenol,1.6461150674911253
Propileno,0.09049125304047217
Napthalene,1.3315284447928277
o-Xylene,0.17980685774441094
Phenanthrene,0.16133780144950513
p_mXylene,0.2484344116148275
Propano,0.017287392248693932
Pyrene,0.03614260391059461
Toluene,3.652427752814233
TOTAL,18.251925933385447
Trans_2_buteno,0.0

Lista de Figuras

Lista de Tablas