



Universidad
Zaragoza

Trabajo Fin de Máster

Diseño y Desarrollo de una Aplicación para la
Detección de Episodios de Estrés

Design and Development of an Application for Stress
Episodes Detection

Autor

David José Silvera Villegas

Directores

Sandra Baldassarri

Pedro Álvarez

ESCUELA POLITÉCNICA DE TERUEL
2022-2023

AGRADECIMIENTOS

En primer lugar agradecer a mis padres y familia que me apoyaron para llegar aún más lejos de lo que pensé llegar. A mis profesores Sandra Baldassarri y Pedro Álvarez que me abrieron las puertas y me permitieron adentrarme en este maravilloso tema del que hablaré a lo largo del documento. A mi hermano en especial que se adentró en esta aventura conmigo y con quien de a pocos logramos uno a uno nuestros objetivos. A quienes me hospedaron durante mi estadía y a todas aquellas personas que he conocido a lo largo de este viaje y que de una u otra forma me han aportado en este camino. Reconocer que los títulos son personales sería equívoco sin reconocer el apoyo que día a día se recibe por parte de tantísimas personas.

RESUMEN

El reconocimiento de emociones, como campo de estudio de la computación afectiva, presenta aún desafíos por desarrollar: problemas como clasificar las emociones humanas o provocar una emoción específica, entre otros; son motivo de estudios hoy en día. De entre tantas emociones, el estrés es una emoción frecuente que afecta a personas de todo el mundo, relacionada con una variedad de problemas de salud que invita a crear sistemas que permitan monitorear de manera automática dicha emoción. Sin embargo, lograr estos sistemas requiere de varios elementos funcionando correctamente entre sí y de allí que la complejidad aumente. El presente trabajo fin de máster tiene como objetivo diseñar un sistema de reconocimiento de emociones basado en datos fisiológicos que permita identificar si una persona experimenta o no, estrés.

Este trabajo se divide en dos partes: la primera incluye la revisión de trabajos previos que permiten el estudio y diseño de una solución tecnológica. En la segunda, se realiza la programación y evaluación de múltiples modelos de machine learning basados en datos fisiológicos para lograr una clasificación de episodios de estrés que se integran posteriormente en una aplicación móvil. Estos modelos de machine learning son entrenados con el dataset WESAD y puestos a prueba en un experimento piloto realizado con 17 participantes en la ciudad de Zaragoza, usando además, una pulsera E4 de Empatica y la aplicación móvil diseñada previamente.

De esta manera, a lo largo del escrito se presentan la evaluación de diferentes algoritmos de machine learning como *“Decision Tree”*, *“Linear Regression”*, entre otros; para la predicción de estrés a partir de 4 señales fisiológicas provenientes de la pulsera E4 de Empatica. Se llevó a cabo un preprocesamiento de las señales, y posteriormente la extracción de features se realizó tanto en el dominio temporal como en el frecuencial, además de una extracción que toma características específicas de cada señal.

Durante de la evaluación, el modelo con el algoritmo *“Decision Tree”* obtuvo el mejor resultado con un 66.1 % de aciertos sobre la prueba piloto hecha. Los hallazgos muestran que los algoritmos de aprendizaje automático pueden detectar el estrés a partir de señales fisiológicas, pero su funcionamiento mejora al usar de manera conjunta más de un modelo. El sistema tiene aplicaciones potenciales en varios dominios, incluidos el cuidado de la salud o el bienestar. La investigación futura debe centrarse en probar el sistema en una población más grande y diversa y mejorar el uso del sistema en el mundo real.

Palabras Claves: Reconocimiento de emociones, computación afectiva, aprendizaje automático, reconocimiento de estrés.

ABSTRACT

The recognition of emotions, as a field of study in affective computing, still presents challenges to be addressed: issues such as classifying human emotions, recognizing or eliciting a specific emotion, among others, are subjects of current research. Among the many emotions, stress is a prevalent emotion that affects people worldwide, associated with a variety of health problems, which prompts the development of systems that enable automatic monitoring of this emotion. However, achieving these systems requires multiple elements working correctly together, thereby increasing complexity. The aim of this master's thesis is to design an emotion recognition system based on physiological data that allows the identification of whether a person is experiencing stress or not.

This work is divided into two parts: the first involves reviewing previous research works that enable the study and design of a technological solution. In the second part, the programming and evaluation of multiple machine learning models based on physiological data are carried out to achieve stress episode classification, which are then integrated into a mobile application. These machine learning models are trained using the WESAD dataset and tested in a pilot experiment conducted with 17 participants in the city of Zaragoza, using an Empatica E4 wristband and the previously designed mobile application.

Throughout the thesis, the evaluation of different machine learning algorithms, such as Decision Tree, K-nearest neighbors, Linear Regression, among others, for the prediction of stress states based on four physiological signals from an Empatica E4 wristband, is presented. Preprocessing of the signals was performed, and feature extraction was carried out in both the time and frequency domains, as well as extraction of signal-specific characteristics.

During the evaluation, the Decision Tree algorithm yielded the best result, achieving a 66.1% accuracy in the pilot study. The findings demonstrate that machine learning algorithms can detect stress based on physiological signals, but their performance improves when multiple models are used in conjunction. The system has potential applications in various domains, including healthcare and well-being. Future research should focus on testing the system on a larger and more diverse population and enhancing the usability of the system in real-world scenarios.

Keywords: Emotion recognition, affective computing, machine learning, stress recognition.

Índice

1. Introducción y objetivos	1
1.1. Introducción	1
1.2. Objetivos	2
1.3. Estructura del documento	3
2. Estado del Arte	5
2.1. Computación afectiva	5
2.2. Reconocimiento de Emociones	6
2.2.1. Modelos de Representación de Emociones	7
2.2.2. Técnicas de Reconocimiento de Emociones	8
2.3. Reconocimiento de Estrés	10
2.4. Aplicaciones del reconocimiento de estrés en Salud	14
3. Solución Tecnológica	17
3.1. Contexto	17
3.2. Diseño de la Aplicación	18
3.3. Descripción de los Módulos	20
3.3.1. Componente de <i>Communication</i>	20
3.3.2. Componente de <i>Database</i>	21
3.3.3. Componente de <i>Data Processing</i>	21
3.4. Proceso de Uso	22
4. Machine-learning para el reconocimiento de estrés	25
4.1. Descripción del Proceso	25
4.2. Selección de datos Fisiológicos	26
4.3. Preprocesamiento de los datos	28
4.4. Extracción de Features	30
4.4.1. Features en el Dominio del tiempo	30
4.4.2. Features en el Dominio de la Frecuencia	30
4.4.3. Features Específicas	32

4.5. Construcción del Modelo de Machine Learning	33
4.5.1. Resultados de Entrenamiento	34
4.6. Implementación de los modelos	36
5. Evaluación del Sistema y Resultados	37
5.1. Diseño del Experimento	37
5.2. Primer Experimento	40
5.2.1. Resultados de la prueba SAM	40
5.2.2. Comparación del Modelo de Predicción	41
5.3. Segundo Experimento	43
5.3.1. Resultados del Mejor Modelo de reconocimiento	44
5.3.2. Comparación de los Modelos Restantes	47
5.4. Tercera Prueba: Mejoras del modelo de predicción	48
6. Conclusiones y Trabajo Futuro	51
Bibliografía	55
Lista de Figuras	67
Lista de Tablas	69

Capítulo 1

Introducción y objetivos

Como humanos, la acción de sentir emociones resulta completamente natural y cercana. Así, para muchos resulta sencillo reconocer si se está alegre o triste, contento o enojado. Sin embargo, puede resultar difícil diferenciar entre emociones cercanas como feliz, alegre y contento. Si esta acción, propiamente humana, puede resultar complicada; para una máquina lo es aún más.

El siguiente apartado muestra la introducción y los objetivos del trabajo fin de máster *Diseño y Desarrollo de una Aplicación para la detección de Estrés*. Se inicia con una breve introducción que contextualiza brevemente el desarrollo del trabajo, siguiendo con presentación de los objetivos generales y específicos y finalmente se enseña la estructura del trabajo.

1.1. Introducción

En un mundo donde las interacciones humanas nos moldea, comprender y reconocer las emociones se ha convertido en un campo particular de estudio. Las emociones son la base de nuestras experiencias, modelan nuestras relaciones y dan forma a nuestras decisiones. Sin embargo, captar y comprender estas sutiles expresiones emocionales es un desafío complejo y transversal que abarca disciplinas como la psicología, la neurociencia y recientemente, la inteligencia artificial. Se forma entonces un punto en común donde se encuentra la clave para desarrollar nuevas tecnologías capaces de interpretar y responder a las emociones humanas de manera significativa y personalizada a ciertas necesidades que mejoren nuestra interacción con los sistemas computacionales.

La computación afectiva (CA) es un área que ha cobrado un alto interés en múltiples ámbitos de estudio debido a la mejora en la interacción entre un humano y un ordenador, al ser capaz, entre otras cosas, de procesar emociones [1]. Esta interpretación permite que la computación afectiva tenga aplicaciones de interés en múltiples campos

como la educación o la medicina y en un contexto social en el que los dispositivos de cómputo cobran mayor relevancia, motiva su incorporación a nuestra realidad [2].

Para reconocer una emoción por ejemplo, tradicionalmente se han utilizado cuestionarios que preguntan directamente a una persona por su estado emocional y no es sino hasta que se incorpora la tecnología en el siglo XX que estos procesos empiezan a cambiar, adquiriendo y caracterizando señales que responden a ciertas emociones provocadas en un usuario [2]. Este proceso, facilita el entendimiento de nuevas emociones y motiva el interés por conseguir sistemas mucho más eficientes y sofisticados que permitan interactuar de una forma más adecuada con las emociones de una persona [1].

Sin embargo, pese a los avances conseguidos, existen aún limitaciones que deberían ser superadas para lograr mejores resultados. Por un lado, el hecho de obtener una mayor cantidad de datos implica en algunas ocasiones equipos especializados y ambientes controlados que permitan garantizar la calidad de las señales [3] y por el otro, profundizar en las técnicas necesarias para procesar dichas señales con mayor precisión, permitiendo diferenciar entre múltiples estados emocionales [4]. Además, esto no sólo implica los desafíos computacionales sino también los desafíos relacionados al componente humano, algo que ocasiona que las emociones de una persona no sea necesariamente la misma que siente otra [5]. Una situación que evidencia lo complejo que puede ser la clasificación de emociones.

De entre tantas emociones, el estrés se caracteriza por ser una emoción común a varias personas y a varias situaciones. Ha cobrado importancia por su creciente cotidianidad y los efectos que ocasiona en el cuerpo humano [6]. Además, la forma en la que se valida que una persona está estresada se suele llevar a cabo mediante la realización de formularios, dependiendo entonces del instante en el que es provocada la emoción, del contexto propio de la persona y de la forma en la que es percibida [7].

Tales particularidades permiten que su estudio por medio de la computación afectiva, motive a la realización de un sistema que logre identificar episodios de estrés en personas mediante el uso de las tecnologías existentes y que pueda ser utilizado para mejorar la calidad de vida en algunas personas.

1.2. Objetivos

El presente trabajo tiene como objetivo general desarrollar un sistema de reconocimiento de estrés basado en el análisis de señales fisiológicas y la programación de un modelo que permita el reconocimiento a partir de datos obtenidos mediante un dispositivo portable. Para diseñar y desarrollar este sistema, se proponen los siguientes

objetivos secundarios.

- Realizar una revisión de las tecnologías y técnicas de análisis que permitan la medición y caracterización de diferentes señales fisiológicas para la detección de estrés.
- Implementar un método de captura y registro de diferentes señales fisiológicas que permitan detectar estrés mediante el uso de un dispositivo “wearable” y un dispositivo móvil.
- Evaluar diferentes técnicas de procesamiento de señales y algoritmos de “*machine learning*” que proporcione un modelo que tome un conjunto de datos que permita la caracterización y clasificación de ventanas de estrés.
- Desarrollar una aplicación que permita integrar tanto la captura de las señales fisiológicas, así como el procesamiento de señales y los modelos de “*machine learning*” necesarios en un dispositivo móvil.
- Realizar una prueba piloto de provocación de estrés para validar el funcionamiento del sistema propuesto como sistema de reconocimiento de estrés.

1.3. Estructura del documento

La memoria se estructura en 6 capítulos.

En el capítulo 2 se presenta el estado del arte relacionado con la detección y clasificación de estrés. Realizando una revisión sobre la computación afectiva, así como los trabajos realizados con la detección y reconocimiento de emociones, técnicas de “*machine learning*” utilizados y trabajos previos en este campo.

En el capítulo 3 se describe la solución tecnológica propuesta, adentrándose en el diseño de cada uno de los bloques que la componen y que permiten obtener un dispositivo capaz de detectar estados de estrés, como una fase primaria de un sistema más complejo.

El capítulo 4 detalla los algoritmos de “*machine learning*” usados para el análisis e inferencia de los estados de estrés basado en las señales fisiológicas obtenidas de un dispositivo wearable, presentando además las estrategias que se siguieron para el pre-procesamiento de las señales y la construcción del modelo de “*machine learning*”.

El capítulo 5 describe el diseño de un experimento piloto para la provocación de estrés y enseña los resultados obtenidos durante la validación de la solución sugerida.

Finalmente el capítulo 6 expone las conclusiones obtenidas como síntesis del trabajo realizado y describe qué líneas se pueden trabajar a futuro.

Capítulo 2

Estado del Arte

El siguiente capítulo presenta el estado del arte relacionado con el reconocimiento de emociones. Inicialmente se enseña qué es la computación afectiva y cómo se relaciona ésta con el reconocimiento de emociones. Luego se presentan las técnicas de reconocimiento existentes, así como los procedimientos seguidos comúnmente para una correcta clasificación. Posteriormente, se detallan los trabajos específicos para la clasificación de emociones, pasando por explicar qué modelos de representación de emociones existen y qué trabajos se han realizado en reconocimiento de estrés. Finalmente se describen trabajos que visibilicen las aplicaciones e implicaciones en el ámbito de la salud en los que se aplican dichos métodos.

2.1. Computación afectiva

La computación afectiva (CA) es un campo de estudio propuesto por Rosalind Picard [2] en el que invita al estudio de las emociones humanas aprovechando el crecimiento de las capacidades tecnológicas y de cómputo existentes. No sólo es la capacidad de una máquina para entender lo que siente una persona, sino que se extiende a otro tipo de posibilidades como expresar, provocar emociones e interactuar teniendo en cuenta el estado de ánimo de un humano. Como lo menciona Picard, es considerar que una máquina puede llegar a desarrollar algún tipo de inteligencia emocional [8] capaz de convivir con un humano. Algo que como él dice: “asusta y emociona”. Su aplicación se puede extender a diversas áreas como las leyes [9], los videojuegos [10] o la enseñanza [11] y resulta de interés estudiar su uso en la salud humana, pues ésta se ve estrechamente relacionada (positivamente o no) con nuestros estados emocionales. A pesar de su potencial, aún existen ciertos desafíos asociados con la integración y el uso de forma cotidiana. Por ejemplo, Picard [8] cuestiona la confianza en el uso de sensores para reconocer lo que está sintiendo una persona al demostrar que un fenómeno físico como sudar puede estar relacionado a una emoción particular o bien, algún estímulo

externo como el calor. En este caso, la dificultad para definir qué emoción causa tal respuesta fisiológica en particular, conlleva dos partes: por un lado se debe entender qué señales son accesibles con las tecnologías existentes y qué información se puede extraer de ellas.

Además, cuando se quiere provocar una emoción la diversidad humana dificulta la consecución de un sistema que unifique cómo nos emocionamos ya que en muchas ocasiones obedece de aspectos más abstractos como la cultura, la idiosincrasia o incluso la personalidad [1]. Esta variabilidad puede ocasionar problemas al momento de extender los mismos resultados a un grupo, aunque existen casos como los algoritmos de detección de sonrisas implementados comercialmente en dispositivos de consumo masivo, que sirven como ejemplo de la forma en la que puede interactuar una máquina con una emoción humana [12] y sirven como un buen ejemplo de cómo incorporar sistemas de computación afectiva en la vida cotidiana.

Aún así, los esfuerzos por crear nuevos y mejores sistemas, que garanticen una mejor experiencia de usuario, siguen siendo necesarios [13], forzando a crear sistemas de computación afectiva con objetivos muy específicos y cubriendo estas demandas con herramientas que se ajustan a un caso en particular, más que un caso general. En este contexto, es notable cómo la interacción entre la máquina y los humanos necesita de varios elementos dentro de los cuales existe un subcampo que se centra en el reconocimiento de emociones.

2.2. Reconocimiento de Emociones

Se ha visto cómo la computación afectiva responde a un interés de mejorar la interacción entre humanos y máquinas desde una perspectiva emocional. De este campo, el reconocimiento de emociones surge como un área que se centra en la captura y análisis de diferentes datos para reconocer una emoción humana [12]. Durante el proceso, se parte con un conjunto de datos provenientes del mundo real, pero que para ser clasificados necesitan de algún modelo que permita tomar esos datos y encontrar en ellos diferentes características hasta converger en una emoción particular. Esto abre varios pasos intermedios y a la vez, abre interrogantes asociados a qué representaciones de emociones existen o de qué manera podemos analizar los datos para extraer esas características particulares.

Picard [2] postulaba que partir de una representación discreta, que permita precisar entre un grupo finito de emociones, es importante para establecer una referencia común que se pueda usar y validar en diferentes investigaciones. Además, permite entender y ordenar de una mejor forma tales emociones para facilitar su implementación y

articulación con otras aplicaciones [12]. Por ejemplo un modelo que identifica un grupo de 4 emociones permite una clasificación más sencilla pero menos precisa en la descripción de qué siente una persona, todo contrario a un segundo modelo con 9 emociones. Éste podría ser más preciso pero es más complejo de implementar.

2.2.1. Modelos de Representación de Emociones

Aunque las expresiones emocionales se definen como reacciones o impulsos humanos frente a diversos estímulos externos y son principalmente psicológicas; normalmente están acompañadas de una respuesta fisiológica que puede ser medible y que facilita una clasificación [14] de forma más objetiva. El proceso de estructurar los datos que permita a un modelo la clasificación de emociones, requiere la asignación de un nombre o etiqueta a ciertos patrones emocionales.

Los intentos por diferenciar estos estados afectivos de una manera efectiva han partido de múltiples enfoques como análisis temporal, análisis frecuencial o combinaciones entre ellas [15]; sin embargo, llegar a una representación que sea aceptada por su efectividad en la descripción de emociones es aún objeto de estudio y define en gran medida lo simple o complejo que puede ser un sistema de computación afectiva [2].

En general, existen dos grandes modelos de representación de emociones [16]. El primero incorpora los modelos categóricos y permite clasificar emociones en grandes grupos con la ayuda de etiquetas que se relacionan directamente con una emoción específica. En este sentido, tiene la ventaja de describir con bastante facilidad las emociones que siente una persona, pues aprovecha el uso de un lenguaje natural. Sin embargo, puede suceder que ciertas emociones que no se puedan agrupar bajo una etiqueta se queden sin clasificar o bien, que una misma emoción pueda ser clasificada en diferentes grupos dependiendo del contexto socio-cultural de la persona [16].

El segundo grupo se integra por los modelos dimensionales y buscan realizar una descripción cuantitativa de las emociones en las que cada una de ellas ocupa una posición exacta en el espacio, ya sea en dos o tres dimensiones. Esto permite no sólo clasificar diferentes emociones, sino que también permite encontrar relaciones entre ellas.

De entre representaciones que existen, en 1980 [17], Russell propuso una representación cartesiana de distintos estados emocionales basado en dos dimensiones independientes llamadas *valencia* y *arousal*. La *valencia* representa el eje horizontal y está relacionada con qué tan placentera o no es la emoción, así pues a mayor *valencia* mayor placer. Por su parte, el *arousal* conforma el eje vertical y representa qué tanta intensidad genera un estímulo.

La figura 2.1, permite visualizar la representación propuesta por Russell donde las

diferentes emociones son ubicadas dentro de uno de los 4 cuadrantes. Allí la posición de cada emoción en el plano, se relaciona con un par de coordenadas en función del valor *valence* - *arousal* y por tanto, se podría considerar como una representación continua donde además, se puede establecer una región alrededor de una emoción [17]. No obstante, el modelo de Russell no es la única representación dimensional que existe.

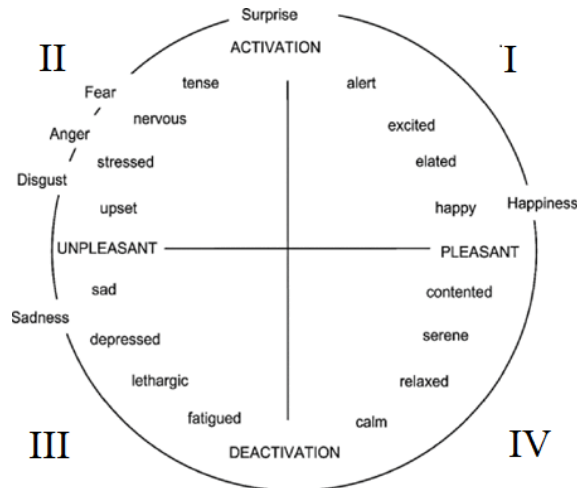


Figura 2.1: Representación de los estados emocionales basados en clasificación de *valence*(unpleasant - pleasant)-*arousal* (deactivation - activation) [17].

En cualquiera de los casos, establecer un modelo de referencia constituye la primera parte del trabajo para el reconocimiento de emociones, siendo el modelo de Russell (en dos ejes) ampliamente aceptado por su integración sencilla y eficacia [8].

2.2.2. Técnicas de Reconocimiento de Emociones

Elegir un sistema de representación de emociones que proporcione un marco de referencia común, que permita la búsqueda de técnicas para relacionar un conjunto de señales con una emoción particular, exige una serie de pasos intermedios.

Para ello, varios sistemas de clasificación de emociones aprovechan el uso de herramientas como el procesamiento digital de señales y el aprendizaje automático para lograr interpretar los datos necesarios y así reconocer correctamente alguna emoción específica [18]. El proceso general consiste en detectar diferentes señales fisiológicas provocadas por una emoción en el cuerpo humano, seguido de un proceso de filtrado y análisis que permita extraer un conjunto de datos estructurados que describan una emoción. Finalmente, se utilizan herramientas como los algoritmos de machine learning para extraer más información de este conjunto de datos y relacionar directamente un conjunto de señales con una emoción.

Trabajos previos en este campo han hecho uso de señales como el pulso cardiaco (HR) [19], [20], [21], electroencefalografía (EEG) [22], [23] o la actividad electrodérmica

(EDA) [3], [24], [25]. No obstante, no sólo se limita a señales fisiológicas, sino que puede involucrar imágenes o audios [26] e incluso de variables que describan el entorno de una persona o su interacción con ella [24], [3], [26]. Algo que Paredes et al. [27] llaman, “*sensar sin sensores*” y que permite entender que los datos analizables pueden provenir de múltiples fuentes.

Una vez definidos el tipo de datos o señales que se quieren adquirir, el siguiente paso es el procesamiento y análisis de la señal. En este aspecto, en función del tipo de señal, así mismo se eligen las técnicas necesarias para su procesamiento y por ende, influyendo en los resultados obtenidos [28]. Esto permite pensar en qué técnicas de procesamiento son necesarias para obtener una clasificación más eficiente, una pregunta compleja cuya respuesta depende de variables como la cantidad de señales, las técnicas de procesamiento o la tecnología disponible [28]. Un proceso que en el campo del reconocimiento de emociones sigue siendo objeto de estudio.

El proceso continua con la extracción de diferentes características que permitan estructurar los datos crudos provenientes de las señales. Respecto a qué métodos utilizar para extraer mayor cantidad de características o con mejor calidad no existe una metodología concreta, pero sí se reconoce que una buena elección permite que los sistemas de reconocimiento de emociones funcionen con mayor eficacia [28], causando además que aunque dos modelos trabajen con las mismas señales, los datos y análisis que ellos efectúan sean diferentes.

Evidentemente la tecnología disponible condiciona qué análisis se puede realizar. Por ejemplo, se han usado señales fisiológicas en equipos como el polígrafo donde una persona experta analiza la señal y encuentra patrones que permiten describir el estado emocional de una persona [29]. Sin embargo, debido a los avances en computación y técnicas de machine learning, la tendencia actual ha sido implementar estas técnicas para lograr el análisis de dichas características. De hecho, por mencionar algunos ejemplos, estudios como el realizado por Kim et. al. [30], Valenza et al. [31] o Ayata et al. [19], entre otros, confirman la efectividad del uso de algoritmos de machine learning para analizar y concluir qué emoción experimenta una persona bajo una serie de estímulos.

Como se observa, el proceso de reconocimiento de emociones mantiene una metodología que si bien no es igual en todos los casos, sí conserva una estructura muy similar. Con ello se busca ejemplificar la importancia de definir qué características resultan importantes para cada señal y cada caso de estudio en específico, además de cómo éstas permiten obtener como resultado sistemas de clasificación más eficaces y de mayor precisión. En el caso del reconocimiento de emociones y en particular de estrés, es necesario considerar qué señales tienen mayor relevancia y qué características de

estas señales le describen mejor, logrando así que los algoritmos de machine learning clasifiquen de mejor manera.

2.3. Reconocimiento de Estrés

Partiendo del reconocimiento de emociones, el estrés puede observarse como un caso particular de este campo. El estrés puede definirse como un proceso subjetivo en el que un individuo responde a situaciones excitantes o desafiantes [32] y conlleva una reacción de la persona que está sometida a tales estímulos. Esta definición permite plantear algunas consideraciones que posibilitan entender qué proceso de análisis que se debe llevar a cabo para lograr el reconocimiento del estrés como emoción. La primera es que existen factores que causan estrés (estresantes) [32], la segunda corresponde al factor psicológico que no siempre garantiza que la reacción sea constante, y un tercer factor que involucra la respuesta fisiológica que servirá como fuente de las señales que serán procesadas [33].

Es importante acotar que el estrés puede tener interpretaciones tanto positivas como negativas que dependen del contexto. A pesar de considerar tradicionalmente el estrés como algo malo [7], hay una respuesta de estrés que puede corresponder con un proceso biológico que busca salvaguardar al individuo, y existe una respuesta de estrés que está asociada al componente psicológico y que en muchas ocasiones causa daños perjudiciales en el cuerpo humano si se prolonga en el tiempo [7].

Trabajos como el de Tran et al. [34], Bustamente [14] o Liapis [35], demuestran una fuerte correlación entre el estrés y los valores de *valence-arousal* que le ubican en el cuadrante II (ver figura 2.1) y que coincide con la propuesta de Russell [17]. En ambas interpretaciones, la respuesta fisiológica se caracteriza por presentar un valor bajo de “*valence*” y un valor alto de “*arousal*”, que en algunas ocasiones puede presentar una valor similar a emociones como enojo, miedo o disgusto [34]. Sin embargo, esta correlación sólo permite establecer una tendencia o región que se puede definir como estrés, pero a la vez, tal región puede ser común a otro tipo de emociones como el miedo [14], el aburrimiento [14] o emociones no identificadas pero con igual valor “*valence-arousal*” [35]; demostrando así que medir y clasificar una respuesta como el estrés es un proceso complejo.

A pesar de ello, existen una serie de trabajos que buscan implementar técnicas para identificar y clasificar episodios de estrés. La idea principal radica en la dificultad de medir los niveles de estrés, ya que tradicionalmente se hace mediante el uso de pruebas como la medición de cortisol en el organismo [36], el uso de formularios u encuestas [37] o el diagnóstico por parte de un profesional [38] y por tanto, motiva a

realizar una clasificación mediante machine learning que permita evitar estos métodos tradicionales.

Los intentos para lograr reconocer episodios de estrés se pueden ilustrar desde diferentes propuestas, pues el trabajo se puede concentrar en mejorar la elección de señales que describen tales episodios, encontrar nuevas y mejores técnicas de procesamiento de señales, obtener mejores modelos de reconocimiento, entre otros. Todas estas mejoras se agrupan en el mismo marco del reconocimiento de estrés, así que cada la mejoría desde cada parte es una mejoría en general. En [39] por ejemplo, el objetivo de los investigadores fue comprobar la efectividad de un sistema para la clasificación binaria de pacientes con estado de estrés y no estrés. En este trabajo, utilizaron un grupo de señales provenientes de una pulsera y mediante el uso de algoritmos de machine learning, lograron encontrar que existen señales capaces de servir como biomarcadores de estrés. Es decir, que este conjunto de señales tiene una correlación directa con episodios de estrés.

El uso de este tipo de señales y la metodología de procesamiento se ve respaldado por otros estudios como los vistos en [40], [41], [42], [6] y [43], donde buscan la clasificación de episodios de estrés por medio de algoritmos de machine learning usando variables como el volumen de pulso sanguíneo (BVP), el ritmo cardiaco (HR), la actividad electrodérmica (EDA) o la Temperatura (TEMP). Tales señales, además de servir para la clasificación como en los casos anteriores, tienen el común denominador de poder obtenerse de un dispositivo “*wearable*”, con la misma fiabilidad que presentan otras técnicas tradicionales.

La figura 2.2 resume las señales que han sido usadas para clasificar episodios de estrés en distintas investigaciones y los resultados obtenidos. Como se puede observar, la señal EDA es la única señal común en todos los trabajos, seguida de BVP. Siendo Carreriro et al. [41], el único estudio en que no se usó pero que, en cambio a dicha señal, fue usado el pulso cardiaco (HR). Una señal que puede ser derivada de la misma señal BVP [39] y que por tanto permite sugerir una relación entre ellas.

Además, se puede observar que no existe una estimación directa en la cantidad de señales que se pueden usar, pues trabajos como el de Indikawati et al. [6] hacen uso de solo 3 señales, mientras que Carreiro et al. [41] hacen uso de 5 señales. Incluso, resaltan trabajos como el de [42] donde se usa del acelerómetro (ACC) y marcas de eventos (Tag), o el de [40] que hace uso de la señal IBI (“*Inter-beat Interval*”). Este comportamiento reafirma que el problema depende también de las técnicas de procesamiento, pero abre la cuestión respecto a cuántas señales o qué señales se deben analizar.

En un sentido general, estas señales han demostrado que tienen una respuesta rápida

en diferentes eventos estresantes [44] y han mostrado tener una fuerte correlación que les relaciona estrechamente con episodios de estrés [45]. Aún así, a pesar de los avances alcanzados, el esfuerzo por encontrar señales con mejor calidad o de correlación más fuerte sigue siendo objetivo de estudio [18].

Author	Signals Used						
	Temp	BVP	EDA	IBI	HR	ACC	Tag
Sevil et al. [40]	x	x	x	x	x	For preprocessing	
Carreiro et al. [41]	x		x		x	x	x
Can et al. [42]	x (for EDA Preprocessing)	x	x			x (for EDA Preprocessing)	
Indikawati et al.[6]	x	x	x				
Schmidt et al.[43]	x	x	x			x	

Figura 2.2: Resumen de las señales usadas en trabajos previos para la clasificación de cuadros de estrés y sus resultados

Por su parte, al revisar qué features se han empleado, se encontró que en su mayoría corresponden a características de tipo estadísticas de cada una de las señales, en general, la media, varianza, máximos, mínimos, el valor eficaz (RMS), entre otros. En el caso de Sevil et al. [40] en el que recurrieron a un número mayor de features producto de relaciones entre las señales, aunque al final del estudio fueron descartadas por su baja importancia para el algoritmo de clasificación. Por ejemplo, medir el cambio de EDA respecto a ACC permite pensar si la primera ha cambiado debido a un incremento en el movimiento de la persona (bien por hacer ejercicio o un movimiento constante) y permite pensar, que puede tener una fuerte correlación para identificar un estado de estrés [40]. Aunque en la práctica los resultados pueden demostrar que este tipo de features no tienen relevancia.

Author	Own Dataset	Features	ML Algorithms												
			K-Nearest Neighbors	SVM	Decision Tree	Naive Bayes	Ensemble Learning	Linear Discriminant	Deep learning	Multilayer Perceptron	Logistic Regression	PCA	Random Forest		
Sevil et al. [40]	N	Statical & Math Features Ratios between signals. Total: 2068 features (718 primary + 1350 secondary)	x	x	x	x	x	x	x						
Carreiro et al.	N	Statical Features & Math Features (25 Features)	x	x	x	x	x	x					x		
Can et al. [42]	N	Statical Features	x	x					x			x		x	x
Indikawati et al. [6]	Y (wesad)	Not Specified			x								x		x
Schmidt et al. [43]	N	Statical Features Non Linear Features			x								x		x

Figura 2.3: Resumen de los algoritmos de machine learning y las features analizadas en trabajos previos.

En la figura 2.3, se observan los algoritmos y features que han sido usados en clasificación binaria (estrés - no estrés) en diversos estudios. Se ve que algoritmos de machine learning como el de “*Linear Discriminant Analysis*” (LDA), “*Decision Tree*” (DT) o “*K-Nearest Neighbors*” (KNN) son las técnicas más recurrentes, aunque no son necesariamente las de mejor resultado pues en [40] obtuvieron que el algoritmo “*Support Vector Machine*” (SVM) era mejor que el algoritmo “*Random Forest*” (RF) usado en [6].

Si bien los algoritmos de machine learning y un conjunto de features estadísticas permiten obtener resultados con alta precisión, es necesario abordar el aspecto del origen de los datos para realizar el entrenamiento y las condiciones en las que se deben ejecutar tales procesos. En la figura 2.3, se observa que de los 5 trabajos presentados solo uno (Indikawati [6]) utilizó un dataset externo mientras que en [40], [41], [42] y [43] se aplicaron pruebas de estrés específicas para generar datasets propios en cada estudio. Esto genera que tales datos sólo se puedan asociar a esas condiciones específicas del experimento y que el proceso de comparar los resultados entre los diferentes modelos no se pueda realizar de forma directa aunque se refiera a la misma emoción.

En particular, el trabajo de Indikawati [6] a pesar de utilizar un dataset externo, en realidad utilizó los datos de Schmidt et al. [43]. De este modo, los dos trabajos comparten un dataset en común y permiten establecer una referencia en el uso de unos algoritmos de machine learning, unas features en común y un resultado similar: 90 % [6] respecto al 93 % [43] en la clasificación binaria de estrés. Sugiriendo además, que el dataset WESAD [43] se pueda utilizar como parte del entrenamiento del sistema de machine learning que se presenta en el capítulo 5, ya que en general no se dispone de muchos datasets públicos que se relacionen directamente con estrés [26].

La revisión de los trabajos realizados permite tener sólo un panorama de qué tantas

señales se consideran al momento de clasificar estrés, sino que se cuenta con una guía sobre qué metodología se puede seguir, qué features se pueden extraer y qué algoritmos de clasificación permitieron un buen resultado. Esto permite definir el punto inicial para la aplicación que se propone como solución, donde se toman en consideración las señales y algoritmos que resultaron relevantes en la figura 2.2, así como los resultados obtenidos a modo de referencia.

2.4. Aplicaciones del reconocimiento de estrés en Salud

En los últimos años, ha habido un creciente interés en el desarrollo y aplicación de sistemas de reconocimiento de emociones en el ámbito de la salud. Con los avances tecnológicos y los constantes progresos en la computación afectiva, se ha abierto un mundo de posibilidades para utilizar dispositivos electrónicos en la detección y seguimiento del estado emocional de las personas [20]. Por ejemplo, en el mercado de consumo masivo, encontramos una variedad de dispositivos que ofrecen seguimiento de señales fisiológicas y prometen detectar y gestionar el estrés. El Oura Ring, el Fitbit y el Apple Watch son solo algunos de los dispositivos populares que utilizan sensores para recopilar datos relacionados con la frecuencia cardíaca, la variabilidad del ritmo cardíaco y la saturación de oxígeno en sangre. Estos datos pueden ser analizados y utilizados para inferir el nivel de estrés de una persona, de hecho, pese a que tienen configuraciones y hardware similares, se diferencian por sus modelos de predicción y análisis [46]. Sin embargo, a pesar de la existencia de estos dispositivos, aún existen desafíos para lograr mejores resultados en algunas aplicaciones específicas [13].

Es importante destacar que no solo los dispositivos de consumo masivo están abordando esta temática. En entornos clínicos y de investigación, se llevan a cabo estudios con el objetivo de comprender y aplicar modelos de reconocimiento de estrés en situaciones más específicas [47]. Por ejemplo, se están investigando las respuestas emocionales de los pacientes durante las fases postoperatorias o en el manejo de afecciones cardíacas, donde a través de dispositivos wearables y aplicaciones móviles, se captan diferentes señales fisiológicas que luego son analizadas por expertos médicos, quienes proporcionan retroalimentación y recomendaciones a los pacientes [20] [21].

Hay otros enfoques que se están explorando para el reconocimiento de emociones en tiempo real [48], mientras que otros se han centrado en el uso de electroencefalogramas para desarrollar sistemas que puedan responder de manera inmediata a los cambios emocionales. Estos esfuerzos buscan lograr la implementación de modelos de reconocimiento de estrés en escenarios reales [49], donde la detección oportuna y precisa

pueda marcar la diferencia en la salud y el bienestar de las personas [47] [50].

En conclusión, el campo de los sistemas de reconocimiento de emociones en salud ha experimentado un notable crecimiento y diversificación. Tanto en el ámbito de consumo masivo como en entornos clínicos y de investigación, se están realizando avances significativos para mejorar la detección y seguimiento del estrés en las personas. A medida que la tecnología continúa avanzando, se espera que estos sistemas desempeñen un papel cada vez más relevante en la promoción de la salud emocional y el bienestar general.

Capítulo 3

Solución Tecnológica

En este capítulo se describen los pasos realizados para lograr una solución que tecnológicamente permita el reconocimiento de estados de estrés. Inicialmente se enseñan las partes que componen el diseño del sistema, introduciendo después las decisiones y elementos que dan forma a cada módulo y cómo operan entre sí. Finalmente se presenta un sistema compuesto por una pulsera E4 de Empatica cuya información se transmite a un dispositivo móvil en el que una aplicación se encarga de la gestión de las señales capturadas por la pulsera, y determinar el estado de estrés del usuario.

3.1. Contexto

Hasta este punto se han presentado diferentes trabajos que permiten comprender qué elementos han sido usados para la detección de diferentes emociones, incluyendo al estrés. En términos generales, los modelos de representación de emociones dimensionales permiten clasificar diferentes emociones de forma cuantitativa, sirviendo como referencias que se pueden relacionar de algún modo a un conjunto de datos. Por otro lado, múltiples investigaciones han confirmado el uso de señales fisiológicas para la detección de estrés mediante el uso de algoritmos de machine learning, donde la selección del algoritmo y de las técnicas de procesamiento son importantes.

Para la consecución de una solución que reconozca estrés se realizaron dos grandes procesos como se observa en la figura 3.1. La primera parte se concentró en aplicar y evaluar diferentes algoritmos de machine learning que permitan obtener un modelo capaz de reconocer estrés a partir de señales fisiológicas y posteriormente, estos modelos de reconocimiento se utilizan en una segunda etapa para integrarlos dentro de una aplicación móvil que sirva como una implementación práctica del modelo. Así, el proceso para encontrar un modelo de clasificación se describe en el capítulo 4 mientras que en este capítulo se describen los detalles del diseño de la aplicación como se muestra en la figura 3.1. En general, la propuesta consta de un sistema compuesto por dos

dispositivos: una pulsera E4 de la empresa Empatica [51] y una aplicación móvil que interactúa con la pulsera donde operan 4 módulos principales como se observa en la figura 3.1. El primer módulo corresponde al de comunicaciones (“*Communication*”), el segundo al de almacenamiento de los datos (“*Database*”), el tercero al de procesamiento de los datos (“*Data Processing*”) y el cuarto a la interfaz de usuario (“*User Interface*”). En primer lugar la E4 será la encargada de captar 4 señales fisiológicas y comunicarse con la aplicación mediante protocolo Bluetooth. Por otra parte, se tiene un teléfono móvil que servirá como dispositivo que aloje la aplicación y como interfaz para interactuar con el usuario, informándole si se encuentra o no en un estado de estrés.

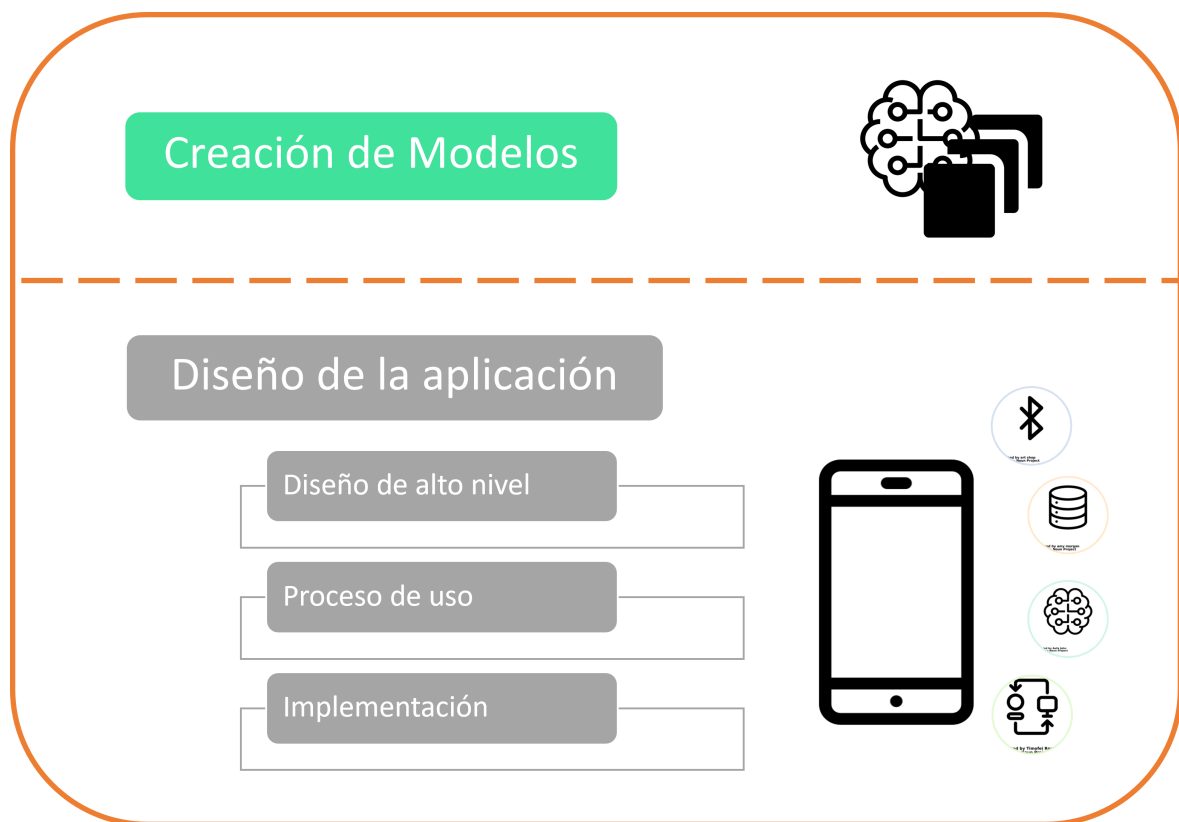


Figura 3.1: Descripción general de la solución. La primera parte busca encontrar un modelo que permita reconocer estrés, mientras que en la segunda se describe el proceso de diseño de una aplicación que permita implementar el modelo. Fuente propia

3.2. Diseño de la Aplicación

El concepto principal es tener dos dispositivos totalmente portables, que puedan ser fácilmente transportados por un usuario y que, como se aprecia en la figura 3.2, sólo dependen de una conexión Bluetooth entre sí para operar. Este esquema, permite que la aplicación funcione en ambientes aislados de Internet, permite aprovechar mejor la batería, o incluso permite que un único terminal (con la misma aplicación) pueda

alojar a varias pulseras a la vez.



Figura 3.2: Concepto de la solución propuesta.

Tal como se ve en la figura 3.3, los 4 módulos inician la operación con la transmisión por parte de la pulsera E4. Una vez encendida, el módulo de (*“Communication”*) entra en acción como el responsable de establecer la conexión Bluetooth y permitir la transmisión de las señales entre la pulsera y la aplicación. Además de recibir los datos, este módulo también envía los datos a los demás módulos. El segundo módulo es *Database* y se encarga de guardar y organizar los datos de las señales recibidas tan pronto como se registra la recepción del primer dato por parte del módulo de comunicaciones. El tercero, *Data Processing*, se encarga de todo el procesamiento de los datos provenientes de las señales así como del reconocimiento de los estados de estrés, para finalmente enviar el resultado al cuarto módulo (*“User Interface”*) quien se encarga de la interacción con el usuario presentándole los resultados.

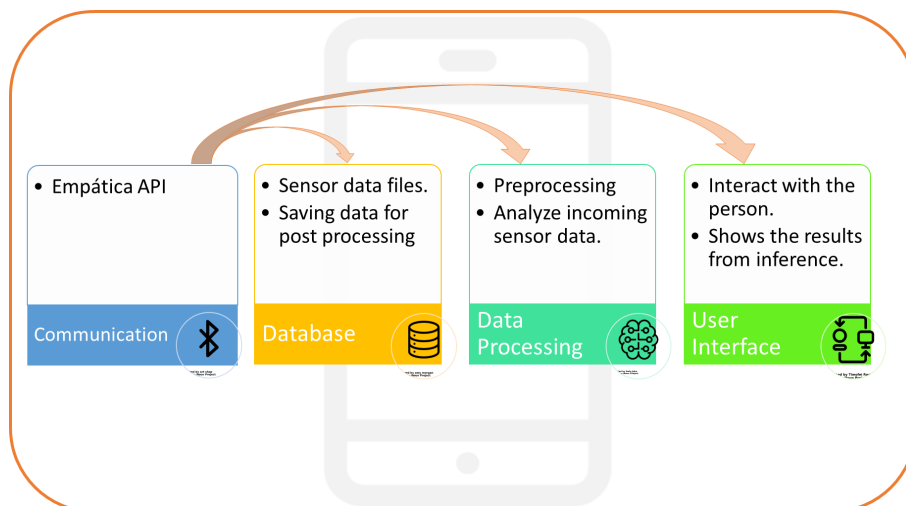


Figura 3.3: Diagrama de bloques dentro de la aplicación.

3.3. Descripción de los Módulos

De los 4 módulos vistos en la figura 3.3, uno de ellos se encarga del flujo o transmisión de los datos, uno al análisis de las señales, un tercero al almacenamiento de la información y el restante a la visualización. En el caso de este último, para la implementación de esta solución la interfaz de usuario sólo mostrará los resultados obtenidos sin profundizar en el desarrollo de una interfaz gráfica en detalle. Con esto, quedan tres componentes sobre los que se trabajó en mayor detalle y que brindan una estructura para que los 4 módulos operen conjuntamente. A continuación, se profundizará con mayor detalle en las características propias de estos tres componentes.

3.3.1. Componente de *Communication*

De fábrica la pulsera cuenta con un protocolo de comunicación Bluetooth que sólo permite la transmisión de datos de forma unidireccional hacia la aplicación, con lo cual el módulo de “*Communication*” está limitado a la recepción de información. Desde el lado de la aplicación, mediante el uso de las librerías de Empatica, se registra un evento cada vez que hay un dato nuevo proveniente de cada sensor. Estos datos se dividen por señal según el sensor de origen y son enviados al componente “*Database*” para que sea guardado localmente en un archivo de texto plano, a la frecuencia propia de cada sensor. Por cada señal (en azul oscuro) procedente de la pulsera, se crea 1 archivo para un total de 7 archivos por sesión como se observa en la figura 3.4. En adición, se observan en azul claro 3 eventos que la pulsera registra como parte de su funcionamiento. Estos eventos corresponden a las condiciones de operación de la pulsera y permiten informar al usuario de un correcto funcionamiento. En este caso, cada evento registrado no ocurre a una frecuencia establecida como sí ocurre con las señales y, en consecuencia, cada vez que la pulsera los envía el módulo de “*Communication*” lo captura junto con el instante en el que ocurrió para que sea gestionado por el módulo de “*Database*”.

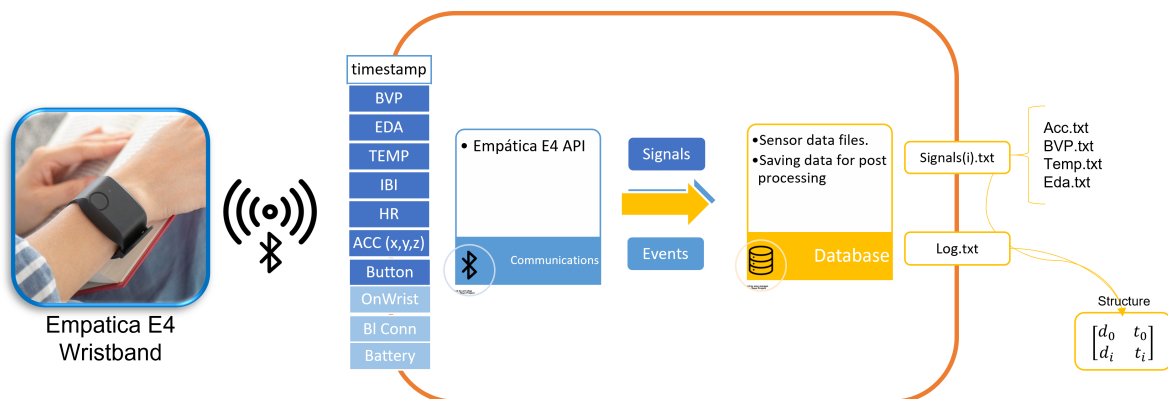


Figura 3.4: Relación entre el módulo de “*Communication*” y “*Database*”.

3.3.2. Componente de *Database*

Por su parte, en la figura 3.4, se muestra en amarillo cómo el módulo de “*Database*” recibe cada uno de los datos capturados por el módulo de “*Communication*” y los almacena localmente en un archivo plano para que sean accesibles tanto por el módulo de “*Data Processing*”, como por el componente de “*User Interface*”. Esto permite también que al estar disponibles localmente en el teléfono, se pueda acceder a los archivos aún cuando se haya perdido la conexión con la pulsera.

Dentro de la aplicación, además de las señales, se registran los eventos de operación dentro de un archivo de registros (*log.txt*) que permite realizar un seguimiento de los eventos ocurridos previamente. En este archivo, se guardan los sucesos relacionados con errores en la comunicación Bluetooth, eventos por batería baja o si el usuario no porta la pulsera. Estos datos resultan importantes, ya que en ninguna documentación oficial es mencionado si los eventos se actualizan a una frecuencia constante (como en los sensores) o no. Además, es importante aclarar que, aunque se registre alguno de estos eventos, la aplicación sólo puede notificarlo mediante la interfaz de usuario ya que la pulsera no cuenta con ningún método que permita su operación a distancia y por ende, una falla en la pulsera implica realizar el reinicio de la misma.

Tanto el archivo *log.txt*, como los archivos provenientes de los sensores, comparten un formato común que en función del tipo de señal, requerirá más o menos cantidad de columnas. Por ejemplo, el archivo proveniente del acelerómetro (*acc.txt*) tendrá una columna por cada eje, más una cuarta que registra la marca de tiempo asociada a dicho evento. En cambio, el archivo de temperatura (*temp.txt*) sólo tendrá dos columnas: una que registra el valor de la temperatura y la marca de tiempo correspondiente.

3.3.3. Componente de *Data Processing*

El tercer módulo de la aplicación corresponde al de procesamiento de datos, que a su vez contiene 3 componentes internos que conforman el núcleo de la aplicación. Estos tres componentes se encargan de una función específica en el manejo de la señal, siguiendo las estructuras y tendencias encontradas en trabajos previos para finalmente estimar la probabilidad de que una persona se encuentre estresado o no.

En la figura 3.5 se observan los tres componentes. El primero es el bloque del preprocesamiento que encapsula todos aquellos procesos encargados de filtrar y limpiar la señal y el segundo se encarga de los procesos de extracción de features según la señal que reciban como entrada. Una vez extraídas, el conjunto de features servirán como datos de entrada al tercer componente que es el de de machine learning, encargado de

analizar los datos y enviar el resultado a la interfaz de usuario.

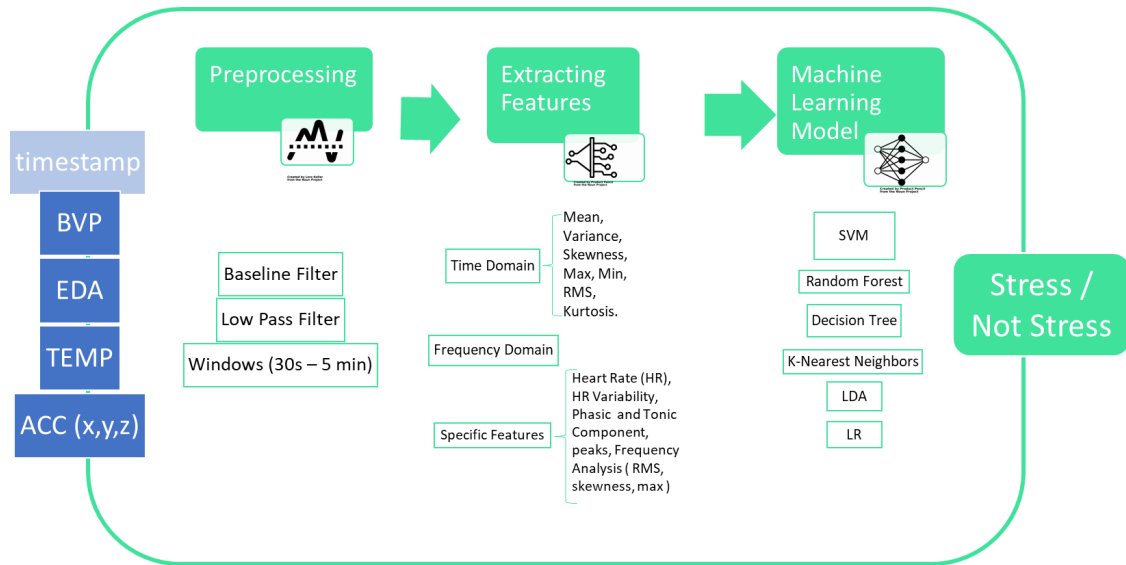


Figura 3.5: Diagrama de bloques del módulo de procesamiento de datos. Fuente propia

El módulo de *“Data Processing”* toma entonces 4 señales directamente del módulo de *“Communication”*. Transcurrido un periodo configurado por la aplicación, éste adquiere los datos hasta tener la cantidad necesaria para realizar el pre procesamiento y extracción de features requeridas por el modelo de machine learning. En un primer caso, se calculan un conjunto de features en el dominio del tiempo (media, mediana, desviación estándar, etc) para el modelo más sencillo, incluyendo de forma uniforme a todas las señales adquiridas. Sin embargo, puede ser necesario más de un modelo para lograr un desempeño óptimo.

3.4. Proceso de Uso

Los 4 bloques presentados en la figura 3.3 se alojan localmente en el teléfono. De esta manera, la pulsera actúa como un dispositivo que envía la información y que el usuario, después de encenderla y posicionarla en la muñeca, sólo deberá portarla verificando que ésta esté correctamente puesta.

Una vez se enciende la pulsera, la aplicación es ejecutada y ella automáticamente inicia la transmisión de las señales captadas por la pulsera mostrando en pantalla el resultado del modelo de clasificación, así como las señales adquiridas. En el caso de este trabajo, la aplicación sólo muestra los datos en texto plano. Por otro lado, la implementación todos los módulos, deberán ser ejecutados de forma que todo el tratamiento de la señal se pueda realizar de la forma más fluida posible. Este paso resulta importante pues dependiendo de cómo se realice la implementación pueden cambiar los tiempos

de análisis y crear un retraso entre la captura de las señales, el análisis y la interacción con el usuario.

Dentro de este escenario existen dos posibles soluciones: La primera corresponde al uso de un servidor externo que permita alojar el módulo de *“Data Processing”* y cuya implementación puede derivar en un mayor tiempo de respuesta. La segunda opción es la de implementar el módulo para que pueda ejecutarse de manera local. Ambas soluciones presentan ventajas y desventajas, como por ejemplo una mayor capacidad de cómputo, velocidad de procesamiento, entre otras; sin embargo, para esta etapa se decidió optar por la solución que permite tener el módulo *“Data Processing”* de forma local. Esta decisión condicionó el tratamiento de la señal y los modelos de machine learning, así como el hardware sobre el que el sistema se ejecuta.

Capítulo 4

Machine-learning para el reconocimiento de estrés

A lo largo de este capítulo se detalla la estructura del modelo utilizado para el reconocimiento de estrés, así como las partes que lo componen. Siguiendo la estructura presentada en el capítulo anterior, se ha establecido que el sistema de reconocimiento de estrés se basa en un modelo que aprovecha las técnicas de procesamiento de señales y los algoritmos de aprendizaje automático para determinar el estado de estrés de una persona. Por tanto, en este capítulo se abordan en detalle los pasos seguidos para obtener el modelo empleado en el sistema de reconocimiento de estrés propuesto, profundizando en las técnicas de procesamiento y los algoritmos de aprendizaje automático evaluados hasta llegar al modelo que se implementará en la solución propuesta.

4.1. Descripción del Proceso

La creación del modelo de reconocimiento que se propone como solución, involucró una serie de procesos (“*Training*” en la figura 4.1) que inician con la selección de los datos fisiológicos que servirán como referencia para el entrenamiento de los algoritmos. Una vez seleccionados, se realizó un preproceso de limpieza y filtrado, para continuar con la extracción de features. Luego, este conjunto de features y sus respectivas etiquetas sirven para el entrenamiento de diferentes algoritmos de machine learning, los cuáles se evalúan hasta encontrar aquel que obtenga los mejores resultados. Finalmente, esta combinación de procesos constituirán un modelo que sirve para la detección y reconocimiento de estrés que se examinará (“*Evaluation*” en la figura 4.1) en el siguiente capítulo .

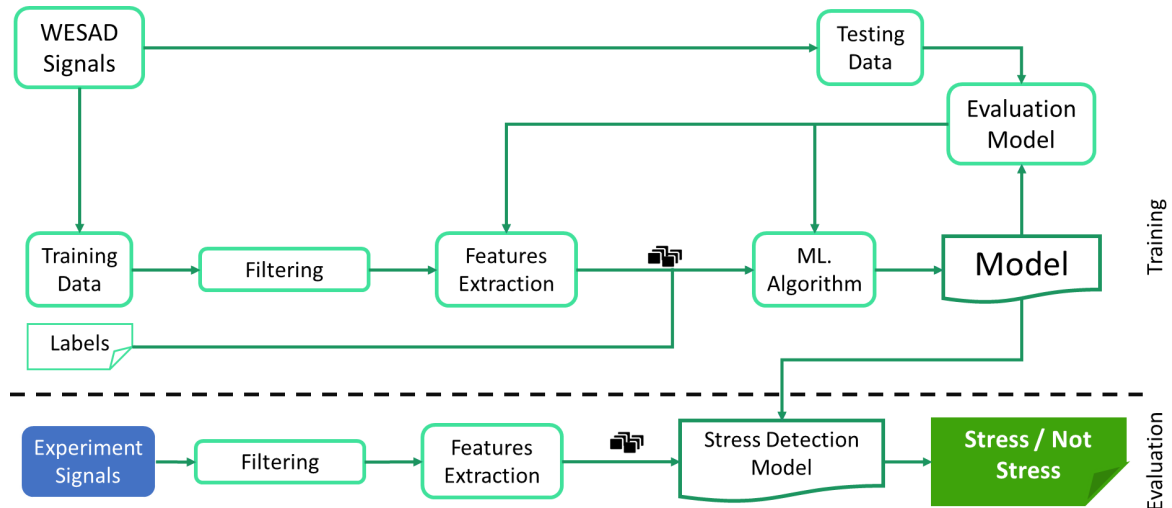


Figura 4.1: Proceso de creación del modelo de para el reconocimiento de estrés.

4.2. Selección de datos Fisiológicos

Al observar la figura 4.1 se observa que el primer paso para la creación del modelo es definir qué señales fisiológicas servirán como dataset para el entrenamiento y evaluación del modelo. Al revisar los trabajos previos se observa que varios de los algoritmos de clasificación utilizados son algoritmos supervisados, y por consiguiente, requieren que el dataset sea previamente etiquetado para ser usado durante el entrenamiento. En este caso, se necesitó entonces un conjunto de señales que estén asociadas a un episodio de estrés, así como a un episodio de no estrés.

Como fuente de estas señales, se optó por utilizar el dataset “*Wearable Stress and Affect Detection*” (WESAD) usado previamente en [43] [6]. Además de evitar la creación de un dataset propio, al usar WESAD es posible comparar los resultados obtenidos durante el entrenamiento respecto a los trabajos previos a modo de referencia. WESAD brinda un conjunto de datos provenientes de señales fisiológicas bajo condiciones de estrés “*Stress*” y en condiciones de reposo (“*Baseline*”), junto con sus respectivas etiquetas. Para la captura de estas señales se reclutaron 15 personas (etiquetados como S2 - S17) con un promedio de edad de 27 años y compuesto por 12 hombres y 3 mujeres. Los participantes utilizaron dos dispositivos donde uno de ellos es la E4 de Empatica. Esta pulsera permite grabar hasta 6 señales, entre las que se encuentran TEMP, EDA, ACC y BVP, con una frecuencia de muestreo de entre 4Hz y 64Hz. Para obtener las señales, se sometió cada participante a una prueba de estrés social de Trier (TSST) [52] en la que, se realizaron unas pruebas de discurso y de aritmética.

La figura 4.2 describe el protocolo seguido durante la prueba. En ésta, se diferencian dos versiones (A y B) compuestas de las mismas 6 etapas pero en una secuencia diferente,

consiguiendo con ello que el orden no influya en la adquisición de los datos. De estas 6 etapas, para este trabajo sólo se consideraron dos, aún cuando podrían ser consideradas como no estrés.

La primera etapa es llamada “*Baseline*” y consiste en 20 minutos en los que cada participante está en un estado neutral que sirve como referencia para el estado de no estrés. Luego como segunda etapa de interés, se tiene “*Stress*” que consiste en la provocación de esta emoción mediante la realización de un discurso personal de 5 minutos, seguido de unas pruebas de aritmética hasta completar un total de 10 minutos [43].

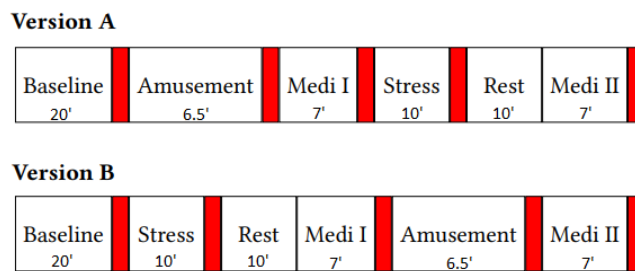


Figura 4.2: Protocolo del estudio realizado para la captura de las señales fisiológicas en condiciones de Estrés. Fuente: [43]

Como resultado de este trabajo, los autores generaron un dataset público compuesto por 7 señales: TEMP, EDA, ACC, BVP, IBI (“*Inter-Beat Interval*”), HR (Heart Rate) y EMB (“*Event Mark Button*”); provenientes de la pulsera comercial E4, etiquetadas por estados emocionales (*Stress*, *Baseline*, etc) y que se pueden usar para entrenar diferentes modelos de machine learning. No obstante, dada la baja cantidad de participantes, los resultados deberán ser analizados con moderación [43].

Finalmente, para continuar con la creación del modelo de reconocimiento, se tomaron 4 señales de la pulsera E4: pulso de volumen de sangre (BVP), actividad electrodérmica (EDA), acelerómetro (ACC) y temperatura (TEMP); asociadas a dos etiquetas: estrés (“*Stress*” en WESAD) y no estrés (“*Baseline*” en WESAD); provenientes de 15 participantes; con un tiempo de 450 minutos de muestras en total que servirán como dataset para el entrenamiento y evaluación de los diferentes algoritmos de machine learning.

De las señales descartadas, HR e IBI se pueden derivar de la señal BVP y, además, presentan un desfase respecto al resto de señales debido a los algoritmos propios de la pulsera. Asimismo, la señal ACC es descompuesta en 3 componentes (X , Y , Z) correspondientes a los 3 ejes cartesianos de movimiento. Obteniendo finalmente un vector con los datos de entrada compuesto de 6 columnas ($[ACC_x, ACC_y, ACC_z, EDA, Temp, BVP]$).

4.3. Preprocesamiento de los datos

La figura 4.3 presenta los pasos durante el preprocesamiento de las señales. Una vez definidas, el primer paso es aplicar un filtro para quitar posibles ruidos que pueden entorpecer el análisis. Para el caso de la señal EDA, se aplicó un filtro de paso bajo con frecuencia de corte en 5Hz similar al trabajo [53] para limpiar la señal de posibles ruidos y así evitar lecturas erróneas de picos que no están asociados a un cambio fisiológico. Por su parte, la señal BVP se filtró usando un filtro de paso alto con frecuencia de corte en 0.03Hz siguiendo a [54] que permite quitar ruidos de base y también, se aplicó un filtro de paso bajo con frecuencia de corte en 3Hz para disminuir posibles ruidos provenientes del movimiento de la pulsera. Las señales del acelerómetro y temperatura no contaron con ningún tipo de filtro del mismo modo que en [6]. Siguiendo la figura 4.3, una vez filtrada cada señal, el siguiente paso consta de aumentar y normalizar la cantidad de muestras por señal.

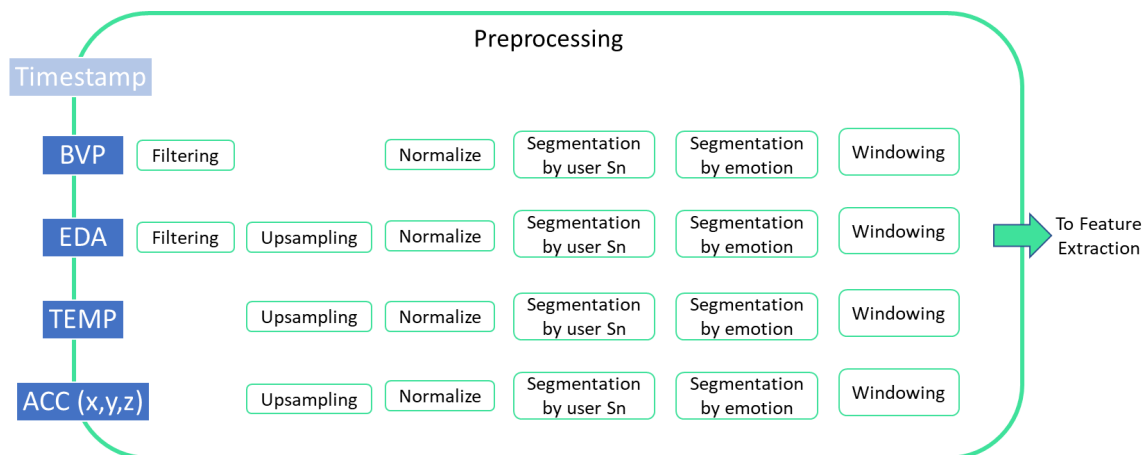


Figura 4.3: Diagrama de bloques del preprocesamiento de la señal.

Dado que las señales son adquiridas por la pulsera a una frecuencia diferente, se propuso aumentar la cantidad de muestras de aquellas señales que tenían una frecuencia menor, al igual que en [55]. A diferencia de [6], este procedimiento permite que las dinámicas de la señal BVP no se vean afectadas por la reducción de muestras y por tanto, permite realizar un análisis más preciso durante la extracción de features. Por el contrario, aumentar la cantidad de muestras de las otras señales puede causar que la señal presenten picos de forma errónea [55] y por tanto, una segunda etapa de filtrado podría ser necesaria. Así pues, tal como se ve en la figura 4.3, después del proceso de filtrado, se realizó un “*upsampling*” [56] hasta obtener un conjunto de 6 señales con una frecuencia de muestreo a 64Hz, en las que las señales EDA, TEMP y $ACC_{x,y,z}$

fueron aumentadas.

Como tercer paso (ver figura 4.3) se normalizó la amplitud de cada una de las señales siguiendo las técnicas usadas en [6] [43]. Esta operación permite que los valores de todas las señales compartan un rango en común sin distorsionar su propia dinámica, algo que resulta necesario para evitar que las magnitudes de una señal enmascaren a otras [57]. Posteriormente, este conjunto de señales se segmentó por participante y por estado emocional (1-Baseline, 2-Stress), obteniendo finalmente un total de 150 minutos en datos de estrés y 300 minutos de datos de no estrés.



Figura 4.4: Proceso de segmentación en ventanas de 30s de duración e incremento de 15s.

Una vez segmentados los datos por participante y por estado, se realizó (ver figura 4.3) la división de éstas en ventanas de tiempo (*“windowing”*) que se van desplazando a lo largo del tiempo y que crean la matriz de datos con la que se estiman las features para los modelos de machine learning.

En [43] utilizan ventanas de 60 segundos con incrementos de 0.25 segundos y ya que el dataset tiene 10 minutos de datos de estrés, para este trabajo se propuso analizar ventanas más cortas con incrementos más largos. Se decidió entonces extraer ventanas de tiempo de 30 segundos con incrementos de 15 segundos, valores que además fueron elegidos por la manera en la que el estrés puede manifestarse en una persona. Un periodo que puede estar comprendido entre unos pocos segundos hasta unos pocos minutos [45].

La figura 4.4 muestra cómo se realiza dicha segmentación. Como primer paso se toman 30 segundos de señal para crear la ventana de datos 1 (W1), luego se desplaza dicha ventana 15 segundos y este nuevo conjunto de datos componen la ventana 2 (W2). Este procedimiento se realiza sucesivamente, hasta llegar al final de la señal. Finalmente, una vez terminado este pre procesamiento, el conjunto de datos está listo para la extracción de features que se presenta en la siguiente sección.

4.4. Extracción de Features

Una vez definido el preprocesamiento de los datos, se continúa con la extracción de features (siguiendo la figura 4.1).

Normalmente los datos por sí mismos no describen en muchos casos el fenómeno que está ocurriendo, dificultando entonces, encontrar patrones que indicaran episodios de estrés. Así, la extracción de features permitió transformar esos datos en crudo a un conjunto de datos que describen con mejor detalle las características propias del estrés. Jugando un rol tan importante como el de seleccionar el algoritmo de machine learning en sí mismo y sobre el que se puede iterar en varias ocasiones como se aprecia en la figura 4.1.

Este proceso se realizó en tres pasos: un primer paso para generar features en el dominio del tiempo, un segundo paso para generar features en el dominio de la frecuencia y un tercer paso para crear features no lineales o específicas a cada una de las señales. En la tabla 4.1 se resumen qué features se extrajeron en cada paso.

4.4.1. Features en el Dominio del tiempo

Este conjunto de features fueron extraídas para la totalidad de las señales, siendo el primer conjunto que se usó como entrada de los algoritmos de machine learning (ML) y describen de forma estadística, aspectos temporales sobre la amplitud de la señal. La tabla 4.1 resume el conjunto de features que se estimaron para cada señal, con un total de 12 features aplicado a las 4 señales (EDA , BVP , $TEMP$, $ACC_{x,y,z}$) agrupando features como la amplitud media (μ), varianza (σ), amplitud mínima (min), amplitud máxima (max), entre otras.

Estas características temporales de la señal, pueden extenderse al crear más features que describan la relación entre las diferentes señales como en [40] [41], sin embargo, en trabajos como [22], las features en el dominio del tiempo se limitan a una cantidad pequeña como las mencionadas en la tabla 4.1. Con esto se quiere aclarar que si bien la cantidad de features puede extenderse tanto como se requiera, la cantidad final puede cambiar después de realizar diferentes entrenamientos de los algoritmos de ML.

4.4.2. Features en el Dominio de la Frecuencia

Para esta sección, se calculó en primer lugar la transformada de Fourier para cada una de las 4 señales y desde el análisis frecuencial, siguiendo a trabajos como el de [43] o [40], se estimaron las features de manera similar a como se hicieron en el dominio del tiempo. Es decir, se calculó en este caso (ver tabla 4.1) la frecuencia media

Type	Signal	Feature	
Time D. Features	EDA[uS]	μ σ Max Min Skewness Energy	Kurtosis # Peaks std Med Range Interquartile Range
	BVP [nW]	μ σ Max Min Skewness Energy	Kurtosis # Peaks std Med Range Interquartile Range
	ACC [1/64g]	μ σ Max Min Skewness Energy	Kurtosis # Peaks std Med Range Interquartile Range
	TEMP[C]	μ σ Max Min Skewness Energy	Kurtosis # Peaks std Med Range Interquartile Range
Frequency D. Features	EDA[uS]	f_{μ} f_{σ} f_{Max} f_{Min} $f_{Skewness}$ f_{Energy}	$f_{Kurtosis}$ $f_{\#Peaks}$ f_{std} f_{Med} f_{Range} $f_{InterquartileRange}$
	BVP [nW]	f_{μ} f_{σ} f_{Max} f_{Min} $f_{Skewness}$ f_{Energy}	$f_{Kurtosis}$ $f_{\#Peaks}$ f_{std} f_{Med} f_{Range} $f_{InterquartileRange}$
	ACC [1/64g]	f_{μ} f_{σ} f_{Max} f_{Min} $f_{Skewness}$ f_{Energy}	$f_{Kurtosis}$ $f_{\#Peaks}$ f_{std} f_{Med} f_{Range} $f_{InterquartileRange}$
	TEMP[C]	f_{μ} f_{σ} f_{Max} f_{Min} $f_{Skewness}$ f_{Energy}	$f_{Kurtosis}$ $f_{\#Peaks}$ f_{std} f_{Med} f_{Range} $f_{InterquartileRange}$
Specefic Features	EDA [uS]	peaks peaks max peaks min max min diff tonic mean tonic std tonic max	tonic min phasic mean phasic std phasic max phasic min scr duration
	BVP [nW]	Heart Rate Mean(HR) Hear Rate Variability (HRV) IBI # of Peaks Statistical of the RR intervals	Energy of Signal by frequency band (ULF, VF, LF, HF, VHF)
	ACC [1/64g]	$Amplitud_{x,y,z}/f$	
	TEMP[C]	∂T	

Tabla 4.1: Lista de features.

(f_μ), la varianza frecuencial (f_σ), la desviación estándar (f_{std}) entre las frecuencias que componen la señal y así sucesivamente; describiendo de forma estadística los comportamientos de la señal desde el espacio frecuencial.

A priori, las señales con baja dinámica frecuencial como EDA o TEMP no tendrían un gran aporte de features [40]. No obstante, esto podría ser rectificado y en cuyo caso (al igual que en el anterior), agregar o descartar estas features.

4.4.3. Features Específicas

En tercer lugar se estimaron un conjunto de features asociadas a la naturaleza y dinámica propia de cada señal. De esta manera, se pueden agregar características que describen con mayor precisión los procesos fisiológicos de la persona o algunas features que sólo se pueden estimar para una señal en específico. Para la estimación de estas features se hizo uso de dos toolbox de uso libre que realizan el análisis de diferentes señales o datos fisiológicas. Biosppy [58] y Neurokit2 [59] son dos toolbox que permiten extraer diferentes características como el intervalo entre pulsos (IBI) o pulso cardiaco (HR) derivado de una señal BVP y características como las componentes tónica y fásica de la señal EDA, que (por ejemplo) están relacionadas con la detección de estrés [6].

De esta forma, la selección de features con el objetivo de reconocer estrés, se basa en la revisión de trabajos previos y encontrar aquellas features en común. Este conjunto se cruzó con los métodos disponibles en los dos toolbox mencionadas y producto de este análisis se obtuvieron finalmente el conjunto de features específicas para cada señal.

La tabla 4.1 resume qué características fueron estimadas para cada señal haciendo uso de las dos librerías, donde se encuentran features como la media de la componente tónica de la señal EDA (“*EDA tonic mean*”), la media de la componente fásica de la señal EDA (“*EDA phasic mean*”), la cantidad de picos en la señal EDA (“*EDA peaks*”); o en el caso de la señal BVP, features como el HR (“*Heart Rate mean*”), la variabilidad de pulso cardiaco (“*HRV*”), el “*Inter-Beat Interval*” (IBI) o el aporte de energía en bandas de frecuencia: ultra baja frecuencia (ULF), baja frecuencia (LF), alta frecuencia (HF) y así sucesivamente.

Después, se comprobó que los resultados de los análisis fuesen similares entre ambas librerías para descartar algún algoritmo erróneo y a la vez confirmar el uso de éstas. Producto de este análisis, se encontró que ambas coinciden bastante en funciones como derivar el ritmo cardiaco (HR) de los datos de una señal BVP o encontrar el intervalo entre pulsos (IBI) e igualmente son capaces de estimar aquellas dinámicas que describen los datos de la señal EDA como las componente tónica y fásica de la señal. No obstante, aunque coinciden en la mayoría de eventos, se notó que la librería Neurokit2 extraía más información de cada señal como describir el inicio de los picos, finalización de los

picos, amplitud, tiempo de subida, entre otras; o en el caso de los datos de la señal BVP, un análisis por banda de frecuencias mucho mayor [59]. Con lo cual, dependiendo del desempeño de las features seleccionadas, se podrá aprovechar ese extra de información. Al final del proceso de extracción de features se obtuvieron un total de 190 características, agrupando los tres tipos de features. A este conjunto se le realizó una primera inspección visual con el fin de verificar que todas las features tuviesen información, esto es, que los resultados no tuviesen valores atípicos como el resultado de una división entre cero. También se inspeccionó visualmente aquellas features que tuviesen valor cero, ya que son features que se pueden estimar pero por razones de la señal no tienen un valor y por ende, podría empeorar el desempeño de los algoritmos durante el entrenamiento.

4.5. Construcción del Modelo de Machine Learning

Hasta este punto, se ha seguido un proceso que permite obtener un dataset que describe diferentes características relacionadas a un grupo de señales, a episodios de estrés y de no estrés. Siguiendo la figura 4.1, el siguiente paso es tomar este dataset para entrenar, evaluar y seleccionar el algoritmo de machine learning que mejor favorezca el reconocimiento de estrés. Sobre esta etapa, puede ocurrir que se deba iterar nuevamente sobre la selección de features, así como en la selección del algoritmo. Luego es necesario comparar entre ellos la precisión con la que funcionan hasta obtener un resultado que conforme finalmente el modelo. Para este trabajo, se evaluaron 6 algoritmos de clasificación: Support Vector Machine (SVM), K-nearest Neighbors (KNN), Decision Tree (DT), Linear Discriminant Analysis (LDA), Random Forest (RF) y Linear Regression (LR).

Un paso previo a realizar el entrenamiento, supone que el dataset obtenido sea dividido en dos partes: un dataset para el entrenamiento (*“Training Data”* en la figura 4.1) y otro para comprobar que el modelo funciona (*“Testing Data”* en la figura 4.1). No obstante, al analizar el dataset WESAD, se notó que la distribución de los datos entre los etiquetas de estrés y no estrés estaba desbalanceada. Debido a ello, se propusieron tres dataset: un primer dataset en el que se tomaron todos los datos disponibles en WESAD para las clases “No Estrés” (NE) y “Estrés” (ST) y cuya proporción demuestra que $NE > ST$, un segundo dataset en el que el estado de $NE < ST$ y un tercer conjunto con igual cantidad de datos para ambos estados ($NE = ST$). Una vez se dividió el dataset WESAD en tres conjuntos, éstos se dividieron en los dos subconjuntos que se ven en la figura 4.1: *“Training Data”*, con el 70% de los datos y, *“Testing Data”* con el 30%. Una vez separados los datos, se entrenó y evaluó cada algoritmo usando

el “*F1-Score*” como indicador de calidad siguiendo a [43] [6] y, de igual forma que se variaron cada uno de los algoritmos, también se varió los tres dataset, y los tipos de features para verificar el resultado de cada una de estas combinaciones.

Finalmente, dado que el dataset es pequeño, los modelos son validados mediante la técnica de “*Cross Validation*”, similar a [43] [6], evaluando el desempeño del modelo con cada uno de los participantes del dataset WESAD. No obstante, en adición y a diferencia de trabajos previos, el modelo obtenido será validado también con los datos recopilados de un experimento propio que se describe en el siguiente capítulo.

4.5.1. Resultados de Entrenamiento

De todas las posibles combinaciones que surgen de variar los tres datasets ($NE > ST$, $NE < ST$ y $NE = ST$), las 4 señales de la pulsera, los tres tipos de features y los múltiples algoritmos de machine learning, se pueden obtener múltiples modelos. No obstante, sucede que algunos de éstos ya han sido evaluados en trabajos previos, lo que permitió elegir una lista de combinaciones que se resume a continuación y que serán cada uno de los modelos a evaluar. Es importante aclarar, que de estos modelos propuestos se realiza una evaluación con cada uno de los algoritmos de machine learning planteados hasta llegar a establecer aquellos que mejor resultados obtenga.

- Modelo 1: Usar todas las señales de la pulsera (ACC, EDA, TEMP, BVP), usando solo features en el dominio del tiempo con cada uno de los dataset ($NE > ST$, $NE < ST$ y $NE = ST$).
- Modelo 2: Usar todas las señales de la pulsera (ACC, EDA, TEMP, BVP), usando solo features en el dominio del tiempo, con cada uno de los datasets $NE > ST$, $NE < ST$ y $NE = ST$ y agregando las features específicas de la señal EDA.
- Modelo 3: Usar sólo la señal EDA, calculando los tres tipos de features con cada uno de los datasets ($NE > ST$, $NE < ST$ y $NE = ST$).
- Modelo 4: Usar las señales EDA y BVP, estimando los tres tipos de features, con cada uno de los datasets ($NE > ST$, $NE < ST$ y $NE = ST$).
- Modelo 5: Usar las señales EDA y BVP, calculando todas las features con cada uno de los datasets ($NE > ST$, $NE < ST$ y $NE = ST$) y agregando la señal ACC con features estadísticas.
- Modelo 6: Usar las señales EDA y BVP, calculando todas las features con cada uno de los datasets ($NE > ST$, $NE < ST$ y $NE = ST$) y agregando la señal TEMP con features estadísticas y específicas.

Una vez establecidos los modelos de interés, se realizó el entrenamiento y evaluación de los mismos usando la librería “*Sci-Kit Learn*” [60]. Alguno de los resultados que se encontró es que el hecho de usar un dataset con mayor cantidad de datos de una etiqueta ($NE > ST$ o $NE < ST$) no permitió obtener algún modelo bueno. Es decir, predecía con mayor inclinación hacia la etiqueta que mayor cantidad de datos presentaba, provocando que durante la validación varias ventanas de tiempo estuviesen mal clasificadas. Al final, se obtuvo que usar el dataset balanceado ($NE = ST$) era la estrategia más factible. En la tabla 4.2, por ejemplo, se observa el resultado de cada modelo considerando este dataset.

Dataset: ($NE = ST$)		Modelo 1	Modelo 2	Modelo 3	Modelo 4	Modelo 5	Modelo 6
SVM	F1-score	0.12	0.00	0.08	0.33	0.22	0.16
	Accuracy	0.64	0.55	0.21	0.45	0.44	0.64
KNN	F1-score	0.05	0.62	0.17	0.27	0.72	0.53
	Accuracy	0.57	0.45	0.47	0.31	0.56	0.36
RF	F1-score	0.21	0.49	0.71	0.41	0.45	0.48
	Accuracy	0.51	0.33	0.55	0.48	0.50	0.43
LDA	F1-score	0.37	0.42	0.51	0.64	0.69	0.72
	Accuracy	0.23	0.34	0.52	0.53	0.61	0.70
DT	F1-score	0.35	0.66	0.79	0.96	0.68	0.41
	Accuracy	0.52	0.58	0.78	0.91	0.74	0.56
LR	F1-score	0.41	0.63	0.78	0.97	0.84	0.73
	Accuracy	0.57	0.46	0.75	0.94	0.70	0.79

Tabla 4.2: Evaluación de los diferentes modelos propuestos con cada uno de los algoritmos de ML para la clasificación de estrés. Abreviaciones: SVM= Support Vector Classification, KNN = k-nearest neighbour, RF = Random Forest, LDA = Linear discriminant analysis, DT = Decision Tree, LR = Linear Regression.

De cada modelo se resalta en negrita aquellos algoritmos con mejores resultados durante el entrenamiento. Así, tal como se observa, el dataset balanceado ($NE = ST$) permitió obtener los mejores resultados, haciendo uso de las señales EDA, BVP y siendo los algoritmos “*Linear Regression*” (LR) y “*Decision Tree*” (DT) aquellos con mejores indicadores, coincidiendo además con [43] o [41].

Además, de los resultados obtenidos en la tabla 4.2 se puede observar que el modelo 1 y 2 tienen baja precisión, llegando incluso a 0 al clasificar estrés. De igual forma, las features propuestas para los modelos 3, 4, y 5 se entenderían como aquellas que describen mejor un episodio de estrés, siendo notorio cómo al agregar los datos de TEMP en el modelo 6 hacen que baje el indicador “*F1-Score*”.

También, es posible notar que aún cuando el indicador *F1-Score* es mayor que el *accuracy*, éste no significa que el modelo sea necesariamente mejor. De hecho, en algunos casos se notó que a pesar de obtener un “*F1-Score*” alto, el “*accuracy*” bajaba, algo que se notaba a medida que el dataset tenía mayor cantidad de datos de estrés. Un efecto que podría asociarse al “*overfitting*”, es decir, a que el modelo se ajustaba demasiado

a los datos de entrenamiento y no logra generalizar correctamente para nuevos datos. Por lo que es necesario ajustar nuevamente la cantidad de features o el tamaño de los dataset de entrenamiento.

Por su parte, el conjunto de features que permitió obtener el mejor resultado se describe en la tabla 4.3. Se puede observar cómo la mayoría de features se concentran en los datos EDA y BVP, donde se estiman todos los tipos de features posibles. Por su parte, para la construcción de este modelo no fue necesario agregar features específicas de ACC y TEMP, siendo esta última la que aporta menor cantidad de features.

Signal	Feature	Signal	Feature
ACC	Time-domain Features (ACC_x) Frequency-Domain features (ACC_x)	EDA	Time-domain Features Frequency-Domain features Specific Features
BVP	Time-domain Features Frequency-Domain features Specific Features	TEMP	Time-domain Features

Tabla 4.3: Lista de features usadas como input de los algoritmos de ML.

4.6. Implementación de los modelos

La última etapa de la selección del modelo de machine learning va de la mano con la validación experimental del mismo. En este caso específico, el objetivo es lograr la implementación y ejecución del modelo de reconocimiento de estrés en la aplicación sin la necesidad de introducir más elementos como se definió en la sección 3.

Durante este proceso, surgió un inconveniente por la imposibilidad de ejecutar el modelo seleccionado directamente en la aplicación móvil. Entonces, fue necesario proponer una adaptación de dicho modelo que se pudiese implementar con los recursos disponibles. La solución se logró mediante el uso de las librerías de Tensorflow y su versión para dispositivos móviles, Tensorflow Lite. En este caso, todo el flujo de pre-procesamiento, extracción de features y selección del algoritmo de machine learning debió ajustarse en función de los recursos del dispositivo móvil y de las posibilidades que la librería permite.

Capítulo 5

Evaluación del Sistema y Resultados

Con varios modelos de reconocimiento definidos y un sistema que los integra con la pulsera, la siguiente fase buscó valorar el desempeño de la solución completa. Esta evaluación permite conocer qué tan robusto y qué tan efectivo es el sistema, así como encontrar qué problemas se pueden presentar. La robustez se refiere a la capacidad del modelo para mantener un rendimiento consistente y confiable en diversas situaciones, superando desafíos como variaciones en los datos de entrada, ruido o presencia de factores confusos. Por su parte, la precisión se refiere a la capacidad para identificar y clasificar correctamente los niveles de estrés, logrando una concordancia entre las predicciones del modelo y la percepción de estrés de una persona. Para el caso de este trabajo, y a diferencia de los trabajos presentados anteriormente, se puso a prueba el modelo con un conjunto de datos de un experimento real. Por ende, los resultados presentados pretenden demostrar una aproximación a un caso real, respaldando su utilidad en aplicaciones prácticas.

A continuación se presentan los resultados obtenidos del experimento hecho. En primer lugar se enseña cómo fue diseñado, seguido de cada uno de los resultados obtenidos en diferentes iteraciones del mismo.

5.1. Diseño del Experimento

Con el sistema de reconocimiento entrenado, resulta de interés poner a prueba su desempeño en un experimento real. El objetivo de éste es evaluar la validez del modelo entrenado previamente y contrastar los resultados obtenidos por el modelo durante el entrenamiento y un escenario real. Esta comparación abre la posibilidad de evaluar otros modelos y así determinar si alguno de ellos obtiene un mejor resultado. Adicionalmente, estos experimentos permite evaluar el funcionamiento del sistema completo, haciendo uso de la aplicación y la pulsera con lo que se puede corroborar un

correcto flujo y análisis de la señal.

Con un modelo de reconocimiento seleccionado, se diseñó entonces un experimento que busca inducir estrés en un grupo de personas. Para ello, se realizó un prueba en la que cada participante debe observar una serie de vídeos a la vez que se capturan las señales procedentes de la pulsera E4. Estas señales son procesadas por la aplicación, donde el modelo realizará una predicción sobre el estrés que cada uno de los vídeos puede generar en los participantes. Finalmente, al término de cada vídeo y como complemento a las señales adquiridas, cada participante responderá un test en el que se le pregunta por su percepción sobre el mismo, así como si conocían alguno de los vídeos propuestos o si habían realizado una prueba anteriormente.

Este experimento necesitó por lo tanto, la selección de un grupo de vídeos que permitan estresar y relajar a una persona para poder contrastar los resultados del modelo en dos casos y además, implica que el test aplicado permita relacionar si los vídeos seleccionados provocaron efectivamente la emoción que se quería. Es decir, que si el vídeo provoca estrés, que en efecto el participante sienta estrés.

Para la selección de los vídeos se recurrió a la consulta de bases de datos de manera que se obtengan vídeos que provoquen estrés, así como de no estrés. El primer inconveniente se encontró en el momento de hallar una base de datos que permitiese este objetivo. Sin embargo, existen un conjunto de estudios que han generado diferentes bases de datos para la provocación de otras emociones [61], [62], [63], [64], [65]. Por ejemplo, Koeltra et al. [61] crearon una base de datos para el análisis de 6 estados afectivos en humanos, donde cada vídeo está clasificado en función de su valor de valence y arousal. A pesar de que ninguno de los vídeos está catalogado como estrés directamente, la selección de ellos se realizó con base en los valores de valence-arousal que estén más cercanos a la zona de estrés descrita en el capítulo 2. Este proceso se realizó igualmente para otras publicaciones hasta completar un conjunto del que (a criterio personal) se seleccionaron 3 vídeos estresantes y 3 vídeos no estresantes.

En el caso del test, se realizó un test SAM (*Self-Assessment Manikin*) que proporciona la percepción emocional de cada participante sobre cada vídeo y permite derivar el valor de *valence-arousal* para cada uno de ellos. Con esta información se espera comparar los resultados obtenidos por el modelo y las pruebas SAM de manera que se pueda explorar algún tipo de relación.

El experimento se realizó en la ciudad de Zaragoza con la participación de 17 personas con una edad comprendida entre los 20 y los 30 años, estudiantes de diversas áreas y con buen estado de salud. Previo a la prueba se les contó el contexto de la prueba y las instrucciones para resolverla. Durante, cada participante vio los vídeos, respondió el test individualmente, aislado de cualquier distracción y con la libertad de terminar

la prueba en cualquier momento. Finalmente, a cada uno de los participantes se les preguntó la autorización para el uso de los datos que se iban a capturar.

La figura 5.1 muestra el protocolo del experimento propuesto. Como se aprecia en la figura, inicia con la conexión de la E4 en el participante. Acto seguido, se conecta la pulsera al participante y se enlaza con la aplicación para transmitir las señales de todo el experimento. A la vez, delante de la persona se ubicaba un computador, con audífonos, en el que la persona daba inicio al experimento. Éste consiste en 7 vídeos, el primero de ellos es uno de 30 segundos que permite establecer una línea de base para el resto y garantiza que la conexión entre la pulsera y el teléfono funciona correctamente. Del resto de vídeos, 3 corresponden a estrés y se presentan en orden aleatorio, finalmente, los 3 restantes corresponden a no estrés y también se presentan de forma aleatoria.

Asimismo, en la figura 5.1, se observa que después del vídeo de línea de base se presenta un vídeo no estrés, seguido de su respectivo formulario SAM. Continúa con el vídeo de estrés y nuevamente el formulario SAM para este último vídeo y así sucesivamente hasta completar los 6 elegidos. Para este experimento, cada vídeo tendrá una duración de 2 minutos en promedio con un total de 12 minutos de señales por participante y será analizado en ventanas de 30 segundos con incremento de 15 segundos para cada uno de los vídeos.

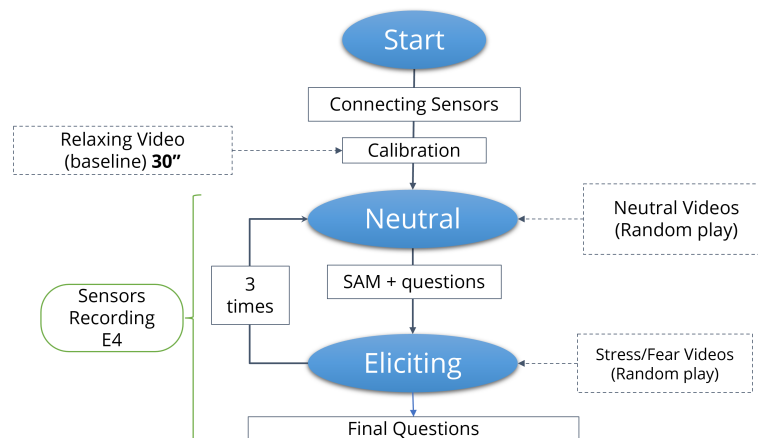


Figura 5.1: Protocolo del experimento realizado para evaluar el sistema de detección de estrés propuesto.

Como resultado del experimento se busca obtener un conjunto de 4 señales provenientes de la pulsera asociados a cada participante y a dos clases de vídeos (estresantes y no estresantes), además de las predicciones hechas por el modelo de reconocimiento implementado en la aplicación. También, se obtendrán los resultados de la prueba SAM incorporada en el experimento con el interés de su análisis posterior.

5.2. Primer Experimento

El primer experimento tuvo como objetivo verificar el funcionamiento de la solución tecnológica y comparar los resultados obtenidos por el modelo de reconocimiento respecto a los resultados obtenidos en el entrenamiento. Para ello, se ejecutó el experimento descrito anteriormente a la vez que la aplicación móvil almacenaba las señales y las predicciones obtenidos. Como modelo de reconocimiento, se utilizó el Modelo 1 en el que se utilizan como señales de entrada ACC, BVP, EDA y TEMP, se extraen las features en el dominio del tiempo y se entrena al algoritmo LR con un dataset No balanceado. Es decir, el entrenamiento con el dataset WESAD completo. De igual forma, las predicciones del modelo se almacenaron en la aplicación, indicando además el porcentaje con el que el modelo predijo en cada ventana de tiempo. Además de validar el funcionamiento del modelo y la aplicación, se almacenaron los resultados de las pruebas SAM para procesarlos junto con los resultados del modelo.

5.2.1. Resultados de la prueba SAM

A los 17 participantes se les preguntó cómo les hacía sentir cada vídeo mediante la aplicación de un test SAM y los resultados permitieron establecer si los vídeos presentados en el test eran percibidos como vídeos estresantes. De existir una concordancia entre los vídeos y la respuesta de los participantes, el modelo tendría un criterio de comparación. Es decir, si el vídeo se percibe como muy poco estresante luego entonces el modelo podría predecir que la persona no está estresada, aún cuando el vídeo esté etiquetado como estrés. La tabla 5.1 recoge los resultados por el test. La escala de evaluación va de 1 a 10, en la que 1 es bajo *valence* o bajo *arousal*. Por el contrario, 10 implica un alto nivel de *valence* o *arousal* y de esta manera, si un vídeo es percibido como estresante por un participante debería estar comprendido en el cuadrante 1 con valores $1 < val < 5$ y $5 < aro < 10$. Si los resultados de la pulsera soportan dicha percepción, se podría pensar que el modelo realiza predicciones válidas. Así, en la tabla 5.1 se muestra la media y desviación estándar de puntuación de cada vídeo, para todos los participantes. La lista completa y discriminada por usuario está disponible como anexo.

De los resultados obtenidos, los vídeos 1, 3 y 5 están etiquetados como no estresantes y para clasificarlos como tal, los valores de la prueba SAM deberían estar alrededor del 5 ($4,5 < val < 5,5$, $3,5 < aro < 4,5$,). No obstante, definir esta región es bastante complicado pues no resulta preciso. En trabajos como [43], por ejemplo, definen el área de no estrés como ($val = 6,7 \pm 0,9$, $aro = 2,5 \pm 0,9$) y el estrés como ($val = 4,5 \pm 1,6$, $aro = 6,8 \pm 1,8$); aún así se ve que mientras el área de no estrés tiene

una desviación menor a 1, en el caso del estrés aumenta hasta 1.8 con lo que el área sigue siendo imprecisa de definir.

Siguiendo con la tabla, es posible encontrar que los vídeos 1, 3 y 5 se encuentran con una puntuación muy cercana entre sí con un valor de *valence - arousal* medio de (5,3) y que se ubican dentro de las áreas establecidas en [43] como vídeos de no estrés. En el caso del estrés, los vídeos 2, 4 y 6 comparten un valor medio de *valence - arousal* aproximado a (4,5, 6,5), coincidiendo con la región definida en [43] y con el primer cuadrante visto en los trabajos previos. Por último, pese a que la desviación es mayor a 2, del conjunto de vídeos se puede decir que el vídeo 6 se percibe como el más estresante.

<i>Usuario 1-17</i>		
<i>Video</i>	<i>Valence</i>	<i>Arousal</i>
1	5,00 ± 1,2	3,06 ± 2,0
2	5,12 ± 2,4	6,88 ± 2,1
3	4,18 ± 1,8	3,29 ± 1,4
4	4,65 ± 1,9	6,06 ± 2,3
5	4,12 ± 1,4	2,94 ± 1,7
6	4,06 ± 2,4	6,71 ± 2,2

Tabla 5.1: Resultados de la prueba SAM de los 17 participantes con los 6 vídeos propuestos.

5.2.2. Comparación del Modelo de Predicción

Los datos recolectados por la pulsera se almacenaron y fueron analizados en ventanas de 30 segundos e incrementos de 15 segundos, siguiendo los parámetros en el entrenamiento del modelo. Un total de 306 ventanas de tiempo asociadas al estado de no estrés junto con 289 ventanas de estrés fueron analizadas con un primer modelo que validará el funcionamiento del sistema. La figura 5.2 muestra los resultados obtenidos con el primer modelo clasificando un vídeo de estrés y uno de no estrés.

Tal como se ve, se indican los datos de *valence* y *arousal* de cada usuario y a continuación, el modelo calcula la probabilidad de que dicha ventana sea un episodio de estrés o no estrés, según el caso. Por ejemplo, en la figura 5.2a, al observar el usuario U1 en la primera ventana, se ve que el modelo predijo con un 69% que el participante estaba estresado para el vídeo 4. En contraste, el mismo U1 en la ventana 1 (W1) de la figura 5.2b tuvo un 97% de probabilidad de no estar estresado.

En adición, en las figuras 5.2a y 5.2b se observa una línea roja que simboliza la finalización del vídeo y el inicio del formulario SAM. En caso de que la persona lo llene tan rápido que el tiempo no sea suficiente para crear una nueva ventana, el sistema no lo toma para clasificación, como por ejemplo en el usuario 12. Este mismo comportamiento se evidencia en el vídeo 2 (disponible como anexo), cuya duración es

de tan solo 42s y por tanto, sólo se puede analizar una ventana como parte del vídeo y el resto corresponden al tiempo de respuesta del formulario.

		Video 4: Stress													
		Participant	Val	Aro	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11
Model 1: All variables, statistics features, (Bl-st) Dataset	U1	2	7	0.69	0.67	0.78	0.79	0.71	0.71	0.73	0.74	0.82	1		
	U2	6	7	0.51	0.51	1	1	1	1	1	1	1	1		
	U3	7	3	0.76	0.75	0.71	0.75	0.65	0.66	0.51	0.51	0.89	0.77	0.95	
	U4	8	6	0.87	0.88	0.89	0.89	0.88	0.81	0.76	0.73	0.58	0.8		
	U5	6	9	0.99	0.98	0.51	0.58	0.98	0.97	0.67	0.96	0.72	0.74		
	U6	2	9	0.98	0.76	0.51	0.94	0.65	0.68	0.93	0.87	0.58			
	U7	5	1	0.73	0.96	0.93	0.93	0.92	0.92	0.93	0.93	0.7	0.76		
	U8	5	2	0.75	0.85	0.53	0.63	0.91	0.93	0.95	0.96	0.8			
	U9	7	6	0.9	0.66	0.65	0.97	0.95	0.95	0.93	0.93	0.5			
	U10	4	5	0.62	0.91	0.85	0.82	0.74	0.73	0.73	0.71	0.71			
	U11	4	5	0.98	0.54	0.51	0.68	0.62	0.65	0.63	0.84	0.96			
	U12	5	5	0.74	1	0.98	0.95	0.95	0.95	0.95	0.95				
	U13	5	8	0.83	0.81	0.87	0.6	0.66	0.74	0.77	0.65				
	U14	2	9	1	1	0.96	0.92	1	0.88	0.51	0.95	0.77	0.66	0.78	
	U15	5	8	0.99	0.86	0.99	0.97	0.97	0.98	0.98	0.99	0.75	0.75	0.88	
	U16	1	7	0.96	0.79	0.89	0.78	0.75	0.75	0.74	0.74	0.79			
	U17	5	6	0.5	1	1	1	0.72	0.71	1	1	0.51			

(a) Vídeo de estrés.

		Video 3: No Stress												
		Participant	Val	Aro	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10
Model 1: All variables statistics features	U1	5	4	0.97	0.87	0.78	0.83	0.76	0.73	0.65	0.99	0.84		
	U2	5	3	1	1	0.83	0.82	0.85	1	1	1	1	1	
	U3	7	2	0.83	0.51	0.51	0.77	0.78	0.5	0.8	0.64	0.64	0.51	
	U6	7	4	0.65	0.72	0.76	0.78	0.81	0.69	0.77	0.8	0.65		
	U4	5	5	0.99	0.7	0.87	0.87	0.6	0.83	0.82	0.97	0.98		
	U5	5	4	0.97	0.89	0.93	0.93	0.99	0.99	0.99	0.99			
	U7	5	2	1	0.97	0.94	0.93	0.92	0.92	0.66	0.82			
	U8	5	2	0.78	0.89	0.81	0.51	0.86	0.86	0.68	0.73			
	U9	5	5	0.87	0.93	0.99	0.55	0.51	0.96	0.94	0.56			
	U10	2	2	0.98	0.84	0.75	0.8	0.9	0.87	0.96	0.96			
	U11	2	1	0.98	1	0.99	1	1	1	1	0.51			
	U12	5	3	0.91	0.5	0.5	0.99	0.5	0.78	0.9	0.98			
	U14	2	4	0.63	0.73	0.72	0.75	0.74	0.72	0.73	0.69			
	U13	1	7	0.99	1	1	0.71	0.78	1	1	1	1	1	0.76
	U15	1	2	0.68	0.98	0.94	0.91	0.83	0.88	0.92	0.83	0.7	0.83	
	U16	4	3	0.96	0.67	0.67	0.52	0.52	0.7	0.6	0.95	0.95		
	U17	5	3	0.55	1	1	1	1	1	1	1			

(b) Vídeo de no estrés.

Figura 5.2: Resultados del modelo 1 durante el primer experimento para dos vídeos. W_n corresponde a cada una de las ventanas, U_n a cada uno de los usuarios. En naranja los casos de predicción fallidos y en verde los casos de predicción acertados..

A priori, la figura 5.2 logra clasificar adecuadamente algunos episodios de estrés con un porcentaje medio de aciertos del 65,4%, no obstante los resultados no muestran un patrón definido. Por ejemplo, a partir del usuario 9 en el vídeo de estrés (figura 5.2a), las ventanas de estrés se concentran al principio, pero la predicción no es del todo continua y se generan nuevamente ventanas de no estrés. Resaltan algunos participantes como el U6 y el U14 quienes califican al vídeo como muy estresante y cuyas reacciones logran ser identificadas correctamente por el modelo de forma casi continua. Sin embargo, ocurren casos en el que los participantes U10 y U11 tienen la misma percepción del vídeo y el modelo no clasifica las ventanas de la misma manera. Asimismo, se observa que participantes como el U16 califica al vídeo como algo estresante y su respuesta fisiológica no fue clasificada como estrés.

En el caso del vídeo de no estrés (figura 5.2b), se tienen casos como U1 y U2 en el que el modelo indica que ambos no están estresado y este comportamiento es continuo a lo largo del vídeo. En contraste, para el U5, el modelo indica que está constantemente estresado, al igual que en el vídeo de estrés (5.2a). Aún así, para el caso de los vídeo de no estrés, el modelo parece obtener un mejor resultado y ser más preciso.

Para este primer experimento se obtuvo una diferencia entre los resultados del entrenamiento y este primer experimento de aproximadamente 30 %, siendo el entrenamiento significativamente mejor con un 84.3 % respecto al desempeño (en términos de aciertos) de 52.1 % durante el experimento. Estas particularidades en algunos resultados y la diferencia en el desempeño, dan la idea que, a pesar de que el modelo está prediciendo adecuadamente algunas de las ventanas de tiempo, aún existen factores que no permiten obtener una clasificación continua y precisa y, da pie a que exista un segundo experimento en el que se pueda probar con diferentes modelos hasta encontrar alguno que mejore el desempeño obtenido.

5.3. Segundo Experimento

Al tener este comportamiento con aparente aleatoriedad de la primera prueba hecha, se decidió realizar diferentes pruebas con el resto de modelos y compararlos respecto al primer experimento. Hasta este punto, los mejores algoritmos eran LR y DT con una precisión en el entrenamiento del 84 % y 68 % (respectivamente) y aunque los resultados pueden estar acorde a otros trabajos previos, el resultado de éstos no garantiza obtener los mismos resultados durante el experimento.

Para el segundo experimento, el objetivo se enfocó en encontrar qué modelo tenía mejor desempeño (nuevamente en términos de aciertos) con las señales obtenidas del experimento 1. Dado que la aplicación almacena las señales de cada uno de los participantes, éstas pueden usarse para evaluar diferentes configuraciones, en diferentes modelos y sin la necesidad de realizar nuevamente un experimento real. Esta posibilidad permite iterar más rápidamente y una vez encontrado qué modelo tiene mejores resultados, se puede pensar en integrarlo nuevamente a la aplicación. Así, este segundo experimento se llevó a cabo de manera virtual, utilizando el lenguaje de programación Python para realizar múltiples iteraciones en un entorno computacional, prescindiendo de la necesidad de llevar a cabo un experimento con participantes reales. Durante cada iteración se variaron parámetros como el dataset con el que se entrenó el modelo, las señales, features y los algoritmos de machine learning, con el fin de encontrar un mejor modelo.

5.3.1. Resultados del Mejor Modelo de reconocimiento

De los modelos propuestos en el capítulo, el modelo 5 demostró ser el que mejor desempeño tiene en los resultados obtenidos. Para el entrenamiento de este modelo en particular, se utilizó el dataset con los datos balanceados $NE = ST$, las señales EDA, BVP y la componente en X del acelerómetro (ACC_x). Como features se estimaron las correspondientes al dominio del tiempo y la frecuencia (media, desviación estándar, etc) para las 3 señales, junto con las features específicas de la señal EDA y BVP y como algoritmo de clasificación, DT. La figura 5.3 resume los resultados obtenidos para este modelo, donde cada ventana (W_n) con resultado “1” corresponde a una predicción de no estrés y “2” corresponde a una clasificación de estrés. En cada caso, las ventanas en verde son predicciones correctas para el vídeo correspondiente.

	Video 2: Stress			Video 4: Stress								Video 6: Stress											
	Val	Aro	W1	Val	Aro	W1	W2	W3	W4	W5	W6	W7	W8	Val	Aro	W1	W2	W3	W4	W5	W6	W7	W8
U1	8	7	1	2	7	2	2	1	1	2	1	2	2	6	8	2	2	2	2	2	1	2	2
U2	9	8	1	6	7	2	2	1	1	2	1	2	1	9	9	1	1	1	1	1	2	2	2
U3	6	4	2	7	3	1	1	1	2	2	2	2	2	6	5	1	1	1	1	2	2	2	2
U6	7	7	1	8	6	1	1	2	1	2	2	2	2	8	6	1	1	1	1	2	2	1	2
U4	4	8	2	6	9	1	1	1	1	1	2	2	2	2	8	1	1	2	2	2	1	2	1
U5	3	9	1	2	9	2	2	2	1	1	1	1	2	1	2	2	2	1	1	2	2	2	1
U7	2	4	1	5	1	2	2	2	1	1	1	1	1	2	5	1	2	2	1	1	2	1	1
U8	5	1	2	5	2	2	2	2	2	1	2	1	1	5	2	1	1	2	2	1	1	1	2
U9	8	9	1	7	6	1	2	2	1	1	1	1	1	6	8	1	2	2	2	2	2	2	1
U10	4	6	1	4	5	1	1	1	2	1	1	2	2	5	5	2	1	1	1	1	2	1	2
U11	3	8	2	4	5	1	1	1	1	2	2	2	2	3	7	1	1	1	2	1	1	2	2
U12	7	6	1	5	5	2	2	1	1	1	1	2	2	5	6	1	2	1	1	1	1	1	2
U14	7	8	2	5	8	2	2	2	1	1	2	1	2	3	9	1	2	1	1	1	2	2	1
U13	7	9	2	2	9	2	2	2	2	1	1	1	1	1	9	2	2	1	1	1	2	1	1
U15	1	9	2	5	8	2	1	1	1	1	1	1	1	3	9	1	1	1	2	2	2	2	1
U16	3	7	1	1	7	1	1	2	2	1	2	2	2	1	8	2	2	1	1	1	1	1	1
U17	3	7	1	5	6	1	1	1	1	2	2	2	2	3	8	1	1	1	1	1	1	2	2

(a) Vídeos de estrés

	Video 1: No Stress						Video 3: No Stress						Video 5: No Stress											
	Val	Aro	W1	W2	W3	W4	W5	W6	Val	Aro	W1	W2	W3	W4	W5	W6	Val	Aro	W1	W2	W3	W4	W5	W6
U1	4	2	2	2	1	1	1	1	5	4	2	1	2	2	1	1	3	2	2	1	1	1	2	2
U2	8	3	1	1	1	2	2	2	5	3	2	2	2	1	1	1	5	4	2	2	2	2	2	1
U3	6	2	1	1	1	1	2	2	7	2	1	1	2	2	2	2	6	3	1	1	1	1	2	2
U6	5	5	1	1	1	1	2	2	7	4	2	1	2	2	1	2	5	1	1	1	1	1	1	2
U4	5	5	1	1	1	1	1	2	5	5	1	1	2	1	2	2	4	6	1	2	1	1	1	2
U5	7	8	2	2	2	1	1	1	5	4	2	1	1	1	2	2	4	7	2	2	2	2	2	2
U7	5	1	1	1	1	2	2	1	5	2	2	2	2	1	1	1	5	2	2	2	2	1	1	1
U8	5	1	1	1	1	1	2	2	5	2	2	1	1	1	1	1	5	2	2	2	2	2	1	1
U9	6	7	2	2	1	1	2	2	5	5	1	1	1	1	1	2	4	4	1	1	1	1	1	1
U10	3	2	1	1	1	2	2	2	2	2	1	1	2	1	1	2	3	3	1	2	1	1	1	2
U11	3	4	1	1	1	2	1	1	2	1	1	1	1	2	2	2	1	1	2	2	2	1	1	1
U12	5	3	2	2	2	1	1	1	5	3	1	2	2	2	2	1	5	4	2	1	1	2	1	1
U14	4	2	1	1	1	1	1	1	2	4	1	1	1	1	2	2	1	1	1	1	2	2	2	2
U13	5	2	1	2	2	2	2	2	1	7	1	1	1	1	1	2	5	1	1	1	1	1	2	2
U15	4	1	2	2	2	2	2	1	1	2	2	1	1	1	2	2	4	2	2	2	2	2	2	2
U16	5	2	1	1	1	2	1	2	4	3	1	1	1	1	1	1	5	3	1	2	2	2	1	1
U17	5	2	1	1	1	2	2	2	5	3	1	1	1	1	2	2	5	4	1	1	1	1	2	2

(b) Vídeos de no estrés

Figura 5.3: Resultados de clasificación de los vídeos para los datos obtenidos en el experimento 1 con el modelo 5. Al inicio de cada tabla se presentan los resultados de *valence-arousal* de cada vídeo, seguido de los resultados de clasificación. W_n corresponde a la ventana n-ésima de cada vídeo. U_n corresponde a los diferentes participantes.

En la figura 5.3a, se observa cómo el modelo 5 clasificó los tres vídeos etiquetados como estresantes. En el caso del vídeo 2, se aprecia que en menos de la mitad de los casos (41 %) el modelo logra clasificar adecuadamente. No obstante, en el caso del vídeo 4 (48,5 %) y del vídeo 6 (44,1 %) los porcentajes de aciertos aumentan respecto al vídeo 2. Más aún, se empieza a notar cierta tendencia de clasificación. Por ejemplo, en el vídeo 6 se empieza a notar cierta tendencia de clasificar correctamente hacia el final de los vídeos. Además, las ventanas que clasifica bien empiezan a mostrar una continuidad que en casos como el usuario 4 (U4), mantienen concordancia con su evaluación SAM. Por su parte, la figura 5.3b muestra los resultados obtenidos por el mismo modelo para la clasificación de los vídeos etiquetados como no estrés. En este caso, los porcentajes de aciertos son 57,8 %, 60,7 % y 52,3 % para los vídeos 1, 3 y 5 respectivamente. Resalta el vídeo 3 que en la mayoría de los usuarios, el modelo logra clasificar de manera continua las ventanas de no estrés, obteniendo resultados como el del usuario 16 con un 100 % de los aciertos.

La tabla 5.2 muestra los resultados obtenidos con el Modelo 5 durante el entrenamiento. En términos generales, al cambiar el dataset y dejar la cantidad de señales en proporciones iguales, el sistema demuestra un comportamiento que durante el entrenamiento es similar al de otras publicaciones, en este caso, un *F1-Score* de 74 % y un *accuracy* de 68 % respecto a un 64,1 % y un 75,21 % en [43]. La condición de que el *F1-Score*, sea mayor que el *accuracy* denota que el dataset aún puede no estar totalmente balanceado [60] y en cuyo caso, se aconseja tomar el *F1-Score* como referente de comparación. Por último, es importante acotar que dentro del estado del arte revisado es la primera vez que se pone a prueba un sistema de machine learning entrenado con el dataset WESAD, en un experimento diferente al propuesto en [43] y de ahí que el porcentaje de aciertos baje hasta en un 20 % durante la prueba real.

	Precision	Recall	<i>F1-Score</i>
Not Stress	0.52	0.66	0.58
Stress	0.80	0.69	0.74
Accuracy	0.68		

Tabla 5.2: Resultados del entrenamiento del modelo 5, hecho con un algoritmo DT, usando un dataset balanceado entre muestras de estrés y no estrés.

Al observar la tabla 5.2 se observa que el sistema responde peor a la detección de cuadros no estrés. Una posible causa se debe a que durante el entrenamiento, los datos relacionados con el estado no estrés son proporcionalmente iguales que los datos relacionados a estrés y aunque se balanceó la cantidad de datos en más de una ocasión, las diferentes señales asociadas a no estrés se comportan de una manera más homogénea

que los que correspondientes al estrés. Aún así, se empezó a notar que los mejores modelos de estrés, acumulaban hacia las ventanas correctas los extremos. Es decir, en el inicio y el final del vídeo como se ve en la figura 5.4.

	Video 0	Video 1: No Stress									Video 2: Stress				Video 3: No Stress							
	W1 W2	W1 W2 W3 W4 W5 W6 W7 W8 W9	W1 W2 W3 W4	W1 W2 W3 W4 W5 W6 W7 W8																		
U5	2 1	2 2 2 1 1 1 1 1 1 1	1 1 2 2	1 1 1 1 1 1 1 1																		
U9	2 1	2 2 1 1 2 1 1 1 1	1 1 2 2	1 1 1 1 1 1 1 1																		
U11	1 1	1 1 1 1 1 1 1 1 1	1 1 2 2	1 1 1 1 1 1 1 1																		
U12	1 2	1 2 2 1 1 1 1 1 1	1 2 2 2	1 1 1 2 2 1 1 1																		
U14	1 1	1 1 1 1 1 1 1 1 1	1 1 1 2	1 1 1 1 1 1 1 1																		

	Video 4: Stress									Video 5: No Stress									Video 6: Stress									
	W1 W2 W3 W4 W5 W6 W7 W8 W9	W1 W2 W3 W4 W5 W6 W7 W8 W9	W1 W2 W3 W4 W5 W6 W7 W8 W9	W1 W2 W3 W4 W5 W6 W7 W8 W9 W10																								
	1 1 2 1 1 1 1 1 1	1 1 1 1 2 2	1 1 1	2 1 1 1 1 1 1 2																								
	1 2 1 1 1 1 1 1 1	1 1 1 1 1 1	2 2 2	1 2 2 1 1 1 1 1 1																								
	1 1 1 1 1 1 1 2 2	1 1 1 1 1 2	2 2	1 1 1 1 1 2 1 2 2																								
	1 1 1 1 1 1 2 2 1	1 1 1 1 1 1	1 2 2	1 1 1 1 1 1 1 2 2 2																								
	1 1 1 1 1 1 2 2 2	1 1 1 1 2 2	2 2	1 2 1 1 1 1 1 1 1 2																								

Figura 5.4: Resultados de predicción para 4 usuarios siguiendo el orden original del experimento, usando el modelo del Modelo 5. “2” corresponde a estrés y “1” corresponde a no estrés. En verde los resultados correctamente clasificados.

En la figura 5.4, ocurren una serie de fenómenos que se rescatan del modelo. En primer lugar, es necesario recalcar que los experimentos fueron hechos en diferente orden para evitar cualquier tipo de predisposición de forma similar a [43]. Así que al observar de cerca los resultados en el orden original del experimento se nota una tendencia a predecir ventanas de estrés hacia los finales del vídeo, con excepción del vídeo 1. En el caso de este, podría deberse a una posible tensión antes de iniciar la prueba, pero en el resto de vídeos coincide con el periodo en el que se realiza el test SAM de percepción visual. Más aún, existen casos de vídeos como el 2 en el que a pesar de ser un vídeo corto, el efecto de éste se puede ver en las siguientes ventanas, junto con el cambio adecuado al Video 3. Es decir, mantiene una continuidad en el reconocimiento.

Este modelo, que hasta ahora ha sido el mejor por sí mismo, cuenta sólo con la señal ACC_x , una condición que coincide con otros trabajos previos. De hecho, aunque la aceleración no se relaciona como una variable que influye de forma directa en el estrés [43], [61] la variable ACC_x sí demostró mejorar el acierto en ventanas de estrés. Incluso da la percepción de que esta variable permite que el sistema respondiera más rápido y que en vídeos cortos como el Video 2, el número de aciertos aumentara. No obstante, este comportamiento podría deberse a los tipo de prueba de estrés que se realizaron en WESAD y que de cierta manera, coincide con este experimento.

Otra causa de los resultados obtenidos, se puede deber a los periodos que se analizaron en el experimento descrito anteriormente. Durante la prueba, se mostraron vídeos de

estrés con un promedio de 2 minutos de duración, que en comparación al experimento descrito en [43], son 5 veces más cortos y exige que la reacción de los participantes al estrés sea lo suficientemente rápida para evidenciarse en las señales. Esta hipótesis está relacionada de igual forma con el hecho de que durante la prueba realizada, se usaron vídeos de miedo cuya naturaleza es la de .^asustar hasta el final del vídeo es justo en ese pico de tensión en la que el participante se podría estresar más y mostrar la respuesta fisiológica en las ventanas posteriores.

5.3.2. Comparación de los Modelos Restantes

Durante cada iteración, se variaron diferentes parámetros y producto de estas combinaciones se encontró que el modelo 5 fue el de mejor desempeño, sin embargo, es posible rescatar algunas características de los demás modelos. La tabla 5.3 y la figura 5.5 resumen el desempeño, en porcentaje de aciertos al evaluar diferentes modelos.

	% Aciertos					
	V1	V2	V3	V4	V5	V6
Modelo 1	0.68	0.71	0.58	0.35	0.60	0.44
Modelo 2	0.55	0.65	0.58	0.46	0.49	0.41
Modelo 3	0.51	0.1	0.68	0.16	0.55	0.125
Modelo 4	0.51	0.76	0.51	0.56	0.44	0.41
Modelo 5 ($NE > ST$)	0.58	0.41	0.61	0.48	0.52	0.44
Modelo 5 ($NE = ST$)	0.80	0.059	0.902	0.25	0.77	0.15

Tabla 5.3: Evolución del desempeño de los diferentes modelos, en función de cada vídeo del experimento. V_n corresponde a los vídeos 1 al 6 del experimento 1 y los modelos 1-5 corresponden a los modelos descritos en el capítulo 4. $NE > ST$ significa que el dataset no estaba balanceado y $NE = ST$ que el dataset estaba balanceado. En negrita el mejor modelo.

Al profundizar en la tabla 5.3 se observa el porcentaje de aciertos de cada modelo respecto a un mismo vídeo. Por ejemplo, la clasificación del vídeo 1 muestra que el mejor resultado se obtuvo al usar el Modelo 5 con el dataset que está balanceado. Mejorando de un 58% a un 80% sólo con el cambio del dataset, algo similar al vídeo 5 (V5) que también mejoró en proporciones similares. No obstante, los resultados para los vídeos de estrés empeoraron en comparación al modelo 1 o 2, pasando por ejemplo de un 71% a un 6%. Una disminución que se mantiene en varios de los vídeos de estrés. A pesar de que la percepción es que el modelo empeoró, en realidad, las observaciones de cómo van prediciendo a lo largo del tiempo, sugieren que estos primeros modelos no son del todo fiables. En primer lugar, porque éstos demuestran una clasificación similar a la observada en el experimento 1, donde cada modelo iba prediciendo aparentemente sin algún orden y a pesar de obtener mayor cantidad de aciertos, éstos carecían de

alguna lógica dentro del contexto. Es decir, si se sigue al modelo 1 (por ejemplo), un participante puede cambiar repentinamente de estar estresado a no estresado en un lapso de 15 segundos y luego, volver a estar estresado, aún cuando el vídeo sea de no estrés. Estos cambios repentinos, pueden ser tomados como falsos positivos (en algunos casos) o bien pueden estar ligados a que los modelos no están bien logrados. Por eso, a pesar de que existen mejores resultados de clasificación de estrés en otros modelos, se determinó que el modelo 5 era el mejor.

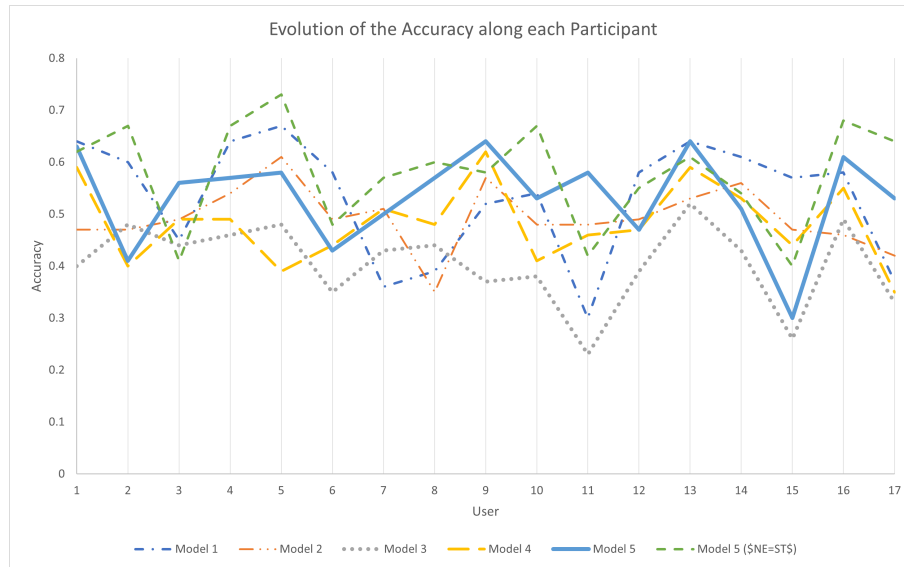


Figura 5.5: Evolución de los aciertos de cada modelo a través de los múltiples usuarios. Para este caso, el modelo 5 (en azul continuo) con los datos balanceados $NE = ST$ vuelve a ser el mejor con una media del 57,7% de aciertos.

En el caso de la figura 5.5, ésta permite ver el desempeño de los modelos a lo largo de los diferentes usuarios. En este sentido, es posible evaluar si un modelo cambia su desempeño en función de algún usuario en particular o si algún usuario en particular tiene el mismo efecto a lo largo de los modelos. Para este caso, se tomó el porcentaje de acierto total (no estrés + estrés) sobre cada participante y resalta como los usuarios U11 y U12 causan que la mayoría de modelos tengan menor cantidad de aciertos con menos del 50%. Aunque en términos generales, los modelos se comportan muy similar, mostrando que los usuarios no influyen en ello.

5.4. Tercera Prueba: Mejoras del modelo de predicción

Aunque se realizaron varias iteraciones y combinaciones, de los dos experimentos anteriores a penas si se pudo superar el 50% de los aciertos en el reconocimiento de estrés. A pesar de ello, se planteó una hipótesis con el objetivo de realizar una prueba

alternativa que mejore aún más los resultados anteriores. Si algunos modelos responden mejor a los vídeos de no estrés y otros a los de estrés, la pregunta es ¿Qué pasaría si no sólo se usa un único modelo de reconocimiento, sino que se usa más de uno?

Esta pregunta motivó a realizar una tercera ronda de experimentación en las mismas condiciones del anterior, en el que los mejores modelos de predicción fuesen usados para operar conjuntamente. Esta idea de mezclar más de un algoritmo o modelo es similar al algoritmo *AdaBoost* usado en [43] y que muestra una mejora en los resultados en dicha publicación. La idea de Adaboost es realizar una clasificación en dos etapas y dos algoritmos, donde los resultados fallidos del primero son clasificados por el segundo.

Sin embargo, a diferencia de Adaboost, en este experimento un primer modelo clasificará los vídeos de la misma forma que en los experimentos anteriores y aquellos ventanas mal clasificadas, serán procesadas por un segundo modelo. Para este caso, los 2 modelos elegidos corresponden a aquellos modelos entrenados previamente y con los que se obtuvieron mejores resultados durante el experimento 2. Esperando en consecuencia, que el reconocimiento de estrés y no estrés sea mejor para cada vídeo. Las señales con las que se realizó el experimento, corresponden a las adquiridas durante el experimento 1, obteniendo como resultado la tabla 5.4.

	% Aciertos					
	V1	V2	V3	V4	V5	V6
Modelo 1*	1	0.059	1	0.27	1	0.17
Modelo 2*	1	0.059	1	0.375	1	0.2721

Tabla 5.4: Desempeño de los modelos en función del porcentaje de aciertos usando dos algoritmos de clasificación. En el modelo 1* se usaron los modelos 5 y 4, con features de las señales EDA y BVP. En el caso del segundo modelo, se usaron los modelos 5 y 4, con features de las señales EDA, BVP y ACC_x .

Para este último caso, se puede ver cómo los vídeos 1, 3 y 5 mejoran en la cantidad de aciertos llegando a un 100% de reconocimiento. En caso contrario, los vídeos de estrés tienen un porcentaje de aciertos más bajo que en los experimentos anteriores, aunque las predicciones en los vídeos 2, 4 y 6 empiezan a mostrar una predicción más continua, concentrándose en los extremos.

Al observar la figura 5.6 se puede ver el efecto de cambiar las features en el modelo 2*, respecto al modelo 1*. Empero, lo interesante está en que las zonas de estrés correctamente clasificadas en el primer modelo, siguen reafirmando en el segundo. Es decir, al agregar más features no sólo se mejora la precisión del modelo, sino que a pesar de las variaciones entre el Modelo 1 y el Modelo 2, se observa una persistencia en la tendencia de reconocer estrés en las mismas regiones de tiempo. Algo que no es baladí, pues la interpretación de estos resultados podría sugerir que los participantes

se estresan más hacia el final, siendo ésta una reacción más natural para este tipo de vídeos. Esta última conclusión abre la oportunidad de explorar aún más, diferentes configuraciones con los modelos obtenidos previamente o modelos nuevos, con el fin de mejorar el desempeño aún más.

Video 4: Stress										
Usuario	Val	Aro	W1	W2	W3	W4	W5	W6	W7	W8
U1	2	7	1	1	1	1	1	2	2	1
U2	6	7	1	1	1	1	1	1	2	2
U3	7	3	2	2	1	1	1	2	1	1
U4	8	6	1	1	1	1	1	2	2	2
U5	6	9	1	1	1	1	1	2	2	2
U6	2	9	1	1	2	2	1	2	2	1
U7	5	1	2	2	2	1	1	1	1	1
U8	5	2	2	2	2	2	1	1	1	1
U9	7	6	2	2	1	1	1	1	1	1
U10	4	5	1	1	1	1	2	1	2	2
U11	4	5	1	1	1	1	1	1	2	2
U12	5	5	1	1	1	1	1	1	2	2
U13	5	8	1	1	1	1	1	1	1	1
U14	2	9	2	2	1	1	1	1	1	1
U15	5	8	1	1	1	1	1	1	1	1
U16	1	7	1	1	1	1	1	1	1	2
U17	5	6	1	1	1	1	1	1	1	2

Video 4: Stress										
Usuario	Val	Aro	W1	W2	W3	W4	W5	W6	W7	W8
U1	2	7	1	1	1	2	2	2	2	2
U2	6	7	1	1	1	1	1	1	2	2
U3	7	3	2	2	2	1	2	2	1	1
U4	8	6	1	1	1	1	1	2	2	2
U5	6	9	1	1	1	1	2	2	2	2
U6	2	9	1	1	2	2	2	2	2	1
U7	5	1	2	2	2	2	1	1	1	1
U8	5	2	2	2	2	2	2	1	1	1
U9	7	6	2	2	2	1	1	1	1	1
U10	4	5	1	1	1	1	2	2	2	2
U11	4	5	1	1	1	1	1	2	2	2
U12	5	5	1	1	1	1	1	1	2	2
U13	5	8	1	1	1	1	2	1	1	2
U14	2	9	2	2	1	1	1	1	1	1
U15	5	8	1	1	1	1	1	1	1	1
U16	1	7	1	1	1	1	1	1	2	2
U17	5	6	1	1	1	1	1	1	1	1

Figura 5.6: Resultados para el video 4 con el modelo que combina dos algoritmos en dos configuraciones distintas: Arriba el Modelo 1 con un algoritmo LR y DT, features con señales EDA y BVP. Abajo el Modelo 2 con un algoritmo LR, DT y features de señales EDA, BVP y ACC_x . Cada ventana (W_n) con “2” significa estrés y “1”, no estrés. En verde las ventanas clasificadas correctamente.

Capítulo 6

Conclusiones y Trabajo Futuro

En conclusión, este trabajo fin de máster se orientó al desarrollo de un sistema de reconocimiento de estrés basado en el análisis de señales fisiológicas mediante técnicas de *“machine learning”*. Para ello, se planteó una primera parte orientada al estudio de varios trabajos previos que permitieran comprender qué señales y técnicas pueden usarse para la detección de emociones y en especial, de estrés.

Después, la segunda parte se destinó al desarrollo del sistema para lo cual, esta parte se dividió en dos procesos. El primero realizó una evaluación de los algoritmos de *“machine learning”* para lograr conformar un modelo de reconocimiento de estrés y el segundo proceso se destinó a tomar este modelo e integrarlo como parte de una aplicación que interactúa con una pulsera E4 de Empatica.

Finalmente, para la validación de este modelo se realizó un experimento con la participación de 17 personas en la ciudad de Zaragoza y que permitió validar un modelo que reconocía estrés con una precisión de hasta un 66.1 % durante la realización del mismo.

Los resultados obtenidos demuestran que es posible detectar ventanas de estrés haciendo uso de 4 señales registradas por la pulsera E4 y para cuyo caso, hace falta realizar un proceso de filtrado y extracción de features adecuado. Más específicamente, con la pulsera E4 de Empatica, las señales EDA y BVP demostraron ser las señales con mayor importancia, donde el uso de features de estas dos señales en el dominio del tiempo, en el dominio de la frecuencia y features específicas, aportaron con mayor importancia a la consecución del modelo que detecta episodios de estrés.

Con el modelo elegido, se logró en el mejor de los casos, una predicción continua de estrés de hasta 180 segundos. Si bien estos valores pueden variar de persona a persona, se logró verificar que el sistema compuesto por una aplicación móvil y la pulsera E4, funcionan adecuadamente como solución al objetivo planteado.

También, se logró demostrar la implementación de los modelos de reconocimiento en plataformas móviles, siendo capaces de tener solamente dos dispositivos que

interactúan entre ellos y que le brindan al usuario información sobre su estado de estrés.

En adición, este trabajo permitió la integración y manejo de diferentes señales, con la posterior implementación de los algoritmos de *“machine learning”* y extracción de features dentro de dispositivos móviles usando plataformas como Edge-Impulse, permitiendo que estos modelos puedan ser llevados a otros dispositivos comerciales o a investigaciones en campo.

Respecto a los algoritmos de *“machine learning”* usados, es posible notar que algoritmos como Linear Regression o Decision Tree permiten la mejor clasificación de ventanas de estrés. Este resultado respalda otros trabajos previos que encontraron que estos mismos algoritmos de clasificación permiten una detección adecuada de situaciones de estrés y a diferencia de ellos, se realizó una prueba piloto en un caso de prueba real.

En cuanto a los dispositivos, es necesario mejorar la implementación de los algoritmos de extracción de features y clasificación para lograr la implementación de modelos más complejos y grandes. Asimismo, la aplicación que se realice deberá contar con una mejor interfaz y arquitectura que el piloto usado en esta prueba.

Del mismo modo, se nóto la escasez en cuanto a bases de datos se refiere. Bien para tener algún dataset que pueda usarse durante los entrenamientos de este tipo de modelos, así como bases de datos de multimedia que se puedan usar para provocar emociones y que brinden una alternativa a los test de estrés que comúnmente se realizan en otros estudios.

Finalmente, entender que el estrés es una respuesta psicológica que se evidencia en el cuerpo mediante la variación de diferentes señales y, a pesar de que campos como la computación afectiva buscan entender mejor este tipo de comportamientos, aún queda bastante trabajo por delante para lograr sistemas que respondan mejor, más asertivamente a las particularidades de una persona y que puedan interactuar de una forma emocional con ellas.

Yo, a título personal, la realización de este trabajo me permitió adentrarme en el mundo del machine learning, todo su proceso y la construcción de modelos de aprendizaje autónomo. Además, el objetivo de poner a prueba dicho modelo en un escenario real, me permitió aprender nuevas técnicas de implementación de machine learning en dispositivos de hardware y aprender nuevos lenguajes de programación que me motivaron a mejorar un poco más mis habilidades técnicas.

De igual modo, la realización del documento en sí mismo, me ha permitido enriquecer mis habilidades de expresión. Algo que en mi opinión, no es baladí, y que complementa mis habilidades como ingeniero. Ha sido un trabajo largo y quizá se escaparon

cosas, pero la consecución de este trabajo también pone a prueba la paciencia y la persistencia hacia alcanzar un nuevo objetivo.

Como líneas de trabajo futuro, se espera mejorar en la implementación de nuevos modelos, con algoritmos de “*machine learning*” diferentes, e integrando arquitecturas y configuraciones nuevas que permitan mejorar los resultados en términos de precisión. También queda para trabajos posteriores, mejorar la cantidad de datos implementados para el entrenamiento, así como las pruebas que se puedan realizar, variando características como grupos de personas, más participantes, o diferentes edades. De igual forma, sería interesante probar este tipo de tecnologías fuera de un ambiente controlado para verificar que sí hay diferencias.

Bibliografía

- [1] Jianhua Tao and Tieniu Tan. Affective computing: A review. In Jianhua Tao, Tieniu Tan, and Rosalind W. Picard, editors, *Affective Computing and Intelligent Interaction*, pages 981–995, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.
- [2] Rosalind W Picard. *Affective computing*. MIT press, 2000.
- [3] Stanislaw Saganowski, Przemyslaw Kazienko, Maciej Dziezyc, Patrycja Jakimów, Joanna Komoszynska, Weronika Michalska, Anna Dutkowiak, Adam Polak, Adam Dziadek, and Michal Ujma. Consumer wearables and affective computing for wellbeing support. In *MobiQuitous 2020-17th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, pages 482–487, 2020.
- [4] Martin Ragot, Nicolas Martin, Sonia Em, Nico Pallamin, and Jean-Marc Diverrez. Emotion recognition using physiological signals: laboratory vs. wearable sensors. In *International Conference on Applied Human Factors and Ergonomics*, pages 15–22. Springer, 2017.
- [5] James J Gross and Robert W Levenson. Emotion elicitation using films. *Cognition & emotion*, 9(1):87–108, 1995.
- [6] Fitri Indra Indikawati and Sri Winiarti. Stress detection from multimodal wearable sensor data. In *IOP Conference Series: Materials Science and Engineering*, volume 771, page 012028. IOP Publishing, 2020.
- [7] Firdaus S Dhabhar. Effects of stress on immune function: the good, the bad, and the beautiful. *Immunologic research*, 58(2):193–210, 2014.
- [8] Rosalind W. Picard. Affective computing: challenges. *International Journal of Human-Computer Studies*, 59(1):55–64, 2003. Applications of Affective Computing in Human-Computer Interaction.

- [9] Xuan Zhou and Weihui Dai. Affective computing on speeches of court defense lawyer. In *2022 International Seminar on Computer Science and Engineering Technology (SCSET)*, pages 117–120, 2022.
- [10] Eva Hudlicka. Affective computing for game design. In *Proceedings of the 4th Intl. North American Conference on Intelligent Games and Simulation*, pages 5–12. McGill University Montreal, 2008.
- [11] Jose Maria Garcia-Garcia, Víctor MR Penichet, María Dolores Lozano, Juan Enrique Garrido, and Effie Lai-Chong Law. Multimodal affective computing to enhance the user experience of educational software applications. *Mobile Information Systems*, 2018, 2018.
- [12] Domen Novak, Guillaume Chanel, Philippe Guillotel, and Alexander Koenig. Guest editorial: toward commercial applications of affective computing. *IEEE Transactions on Affective Computing*, 8(2):145–147, 2017.
- [13] Preeti Kumari, Lini Mathew, and Poonam Syal. Increasing trend of wearables and multimodal interface for human activity monitoring: A review. *Biosensors and Bioelectronics*, 90:298–307, 2017.
- [14] P. A. Bustamante, N. M. Lopez Celani, M. E. Perez, and O. L. Quintero Montoya. Recognition and regionalization of emotions in the arousal-valence plane. pages 6042–6045, 2015.
- [15] Lawrence R. Rabiner and Ronald W. Schafer. Introduction to digital speech processing. *Foundations and Trends® in Signal Processing*, 1(1–2):1–194, 2007.
- [16] Sreeja PS and G Mahalakshmi. Emotion models: a review. *International Journal of Control Theory and Applications*, 10(8):651–657, 2017.
- [17] JONATHAN POSNER, JAMES A. RUSSELL, and BRADLEY S. PETERSON. The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology. *Development and Psychopathology*, 17(3):715–734, 2005.
- [18] Elena Smets, Walter De Raedt, and Chris Van Hoof. Into the wild: the challenges of physiological stress detection in laboratory and ambulatory settings. *IEEE journal of biomedical and health informatics*, 23(2):463–473, 2018.

- [19] Değer Ayata, Yusuf Yaslan, and Mustafa E Kamasak. Emotion recognition from multimodal physiological signals for emotion aware healthcare systems. *Journal of Medical and Biological Engineering*, 40(2):149–157, 2020.
- [20] Stefan Sebastian Busnatu and Crina Julieta Sinescu. Fi-star online personalised cardiac rehabilitation solution. *International Journal of Online and Biomedical Engineering (iJOE)*, 11(1):pp. 67–70, Jan. 2015.
- [21] Shreya Tadas, Claudette Pretorius, Emma J Foster, Trish Gorely, Stephen J Leslie, David Coyle, et al. Transitions in technology-mediated cardiac rehabilitation and self-management: Qualitative study using the theoretical domains framework. *JMIR cardio*, 5(2):e30428, 2021.
- [22] Cheng-Jie Yang, Nicolas Fahier, Chang-Yuan He, Wei-Chih Li, and Wai-Chi Fang. An ai-edge platform with multimodal wearable physiological signals monitoring sensors for affective computing applications. In *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–5, 2020.
- [23] Giorgos Giannakakis, Dimitris Grigoriadis, and Manolis Tsiknakis. Detection of stress/anxiety state from eeg features during video watching. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 6034–6037, 2015.
- [24] P. Álvarez, F.J. Zarazaga-Soria, and S. Baldassarri. Mobile music recommendations for runners based on location and emotions: The dj-running system. *Pervasive and Mobile Computing*, 67:101242, 2020.
- [25] Jongyoon Choi, Beena Ahmed, and Ricardo Gutierrez-Osuna. Development and evaluation of an ambulatory stress monitor based on wearable sensors. *IEEE Transactions on Information Technology in Biomedicine*, 16(2):279–286, 2012.
- [26] Lin Shu, Jinyan Xie, Mingyue Yang, Ziyi Li, Zhenqi Li, Dan Liao, Xiangmin Xu, and Xinyi Yang. A review of emotion recognition using physiological signals. *Sensors*, 18(7):2074, 2018.
- [27] Pablo Paredes, David Sun, and John Canny. Sensor-less sensing for affective computing and stress management technology. In *2013 7th International Conference on Pervasive Computing Technologies for Healthcare and Workshops*, pages 459–463, 2013.
- [28] Charles S Lessard. Signal processing of random physiological signals. *Synthesis Lectures on Biomedical Engineering*, 1(1):1–232, 2005.

- [29] National Research Council et al. *The polygraph and lie detection*. National Academies Press, 2003.
- [30] Kyung Hwan Kim, Seok Won Bang, and Sang Ryong Kim. Emotion recognition system using short-term monitoring of physiological signals. *Medical and biological engineering and computing*, 42(3):419–427, 2004.
- [31] Gaetano Valenza, Mimma Nardelli, Antonio Lanata, Claudio Gentili, Gilles Bertschy, Rita Paradiso, and Enzo Pasquale Scilingo. Wearable monitoring for mood recognition in bipolar disorder based on history-dependent long-term heart rate variability analysis. *IEEE Journal of Biomedical and Health Informatics*, 18(5):1625–1635, 2013.
- [32] R. Sinha. How does stress increase risk of drug abuse and relapse? *Psychopharmacology*, (158), 2001.
- [33] Richard S. Lazarus. *Stress and Emotion: A New Synthesis*. Springer Publishing, 2006.
- [34] Thi-Dung Tran, Junghee Kim, Ngoc-Huynh Ho, Hyung-Jeong Yang, Sudarshan Pant, Soo-Hyung Kim, and Guee-Sang Lee. Stress analysis with dimensions of valence and arousal in the wild. *Applied Sciences*, 11(11), 2021.
- [35] Alexandros Liapis, Christos Katsanos, Dimitris G Sotiropoulos, Nikos Karousos, and Michalis Xenos. Stress in interactive applications: analysis of the valence-arousal space based on physiological signals and self-reported data. *Multimedia Tools and Applications*, 76(4):5051–5071, 2017.
- [36] MedlinePlus. Prueba de cortisol. Consulta en: 2021.
- [37] Shalom Greene, Himanshu Thapliyal, and Allison Caban-Holt. A survey of affective computing for stress detection: Evaluating technologies in stress detection for better health. *IEEE Consumer Electronics Magazine*, 5(4):44–56, 2016.
- [38] Alexandra D Crosswell and Kimberly G Lockwood. Best practices for stress measurement: How to measure psychological stress in health research. *Health psychology open*, 7(2):2055102920933072, 2020.
- [39] Rajdeep Kumar Nath and Himanshu Thapliyal. Smart wristband-based stress detection framework for older adults with cortisol as stress biomarker. *IEEE Transactions on Consumer Electronics*, 67(1):30–39, 2021.

- [40] Mert Sevil, Mudassir Rashid, Iman Hajizadeh, Mohammad Reza Askari, Nicole Hobbs, Rachel Brandt, Minsun Park, Laurie Quinn, and Ali Cinar. Discrimination of simultaneous psychological and physical stressors using wristband biosignals. *Computer Methods and Programs in Biomedicine*, 199:105898, 2021.
- [41] Stephanie Carreiro, Keerthi Kumar Chintla, Sloke Shrestha, Brittany Chapman, David Smelson, and Premananda Indic. Wearable sensor-based detection of stress and craving in patients during treatment for substance use disorder: A mixed methods pilot study. *Drug and alcohol dependence*, 209:107929, 2020.
- [42] Yekta Said Can, Heather Iles-Smith, Niaz Chalabianloo, Deniz Ekiz, Javier Fernández-Álvarez, Claudia Repetto, Giuseppe Riva, and Cem Ersoy. How to relax in stressful situations: a smart stress reduction system. In *Healthcare*, volume 8, page 100. MDPI, 2020.
- [43] Philip Schmidt, Attila Reiss, Robert Duerichen, Claus Marberger, and Kristof Van Laerhoven. Introducing wesad, a multimodal dataset for wearable stress and affect detection. In *Proceedings of the 20th ACM International Conference on Multimodal Interaction, ICMI '18*, page 400–408, New York, NY, USA, 2018. Association for Computing Machinery.
- [44] Palanisamy Karthikeyan, Murugappan Murugappan, and Sazali Yaacob. A review on stress inducement stimuli for assessing human stress using physiological signals. In *2011 IEEE 7th International Colloquium on Signal Processing and its Applications*, pages 420–425. IEEE, 2011.
- [45] Sylvia D Kreibitz. Autonomic nervous system activity in emotion: A review. *Biological psychology*, 84(3):394–421, 2010.
- [46] Stanisław Saganowski, Przemysław Kazienko, Maciej Dziezyc, Patrycja Jakimów, Joanna Komoszynska, Weronika Michalska, Anna Dutkowiak, A Polak, Adam Dziadek, and Michal Ujma. Review of consumer wearables in emotion, stress, meditation, sleep, and activity detection and analysis. *arXiv preprint arXiv:2005.00093*, 2020.
- [47] Fanny Larradet, Radoslaw Niewiadomski, Giacinto Barresi, Darwin G Caldwell, and Leonardo S Mattos. Toward emotion recognition from physiological signals in the wild: approaching the methodological issues in real-life data collection. *Frontiers in psychology*, 11:1111, 2020.

- [48] Jeril V Raj and TV Sarath. An iot based real-time stress detection system for fire-fighters. In *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, pages 354–360. IEEE, 2019.
- [49] V Vanitha and Pandian Krishnan. Real time stress detection system based on eeg signals. 2017.
- [50] Yekta Said Can, Niaz Chalabianloo, Deniz Ekiz, and Cem Ersoy. Continuous stress detection using wearable sensors in real life: Algorithmic programming contest case study. *Sensors*, 19(8):1849, 2019.
- [51] Empatica. E4 wristband. Consulta en: 2021.
- [52] Clemens Kirschbaum, Karl-Martin Pirke, and Dirk H Hellhammer. The ‘trier social stress test’—a tool for investigating psychobiological stress responses in a laboratory setting. *Neuropsychobiology*, 28(1-2):76–81, 1993.
- [53] Cornelia Setz, Bert Arnrich, Johannes Schumm, Roberto La Marca, Gerhard Tröster, and Ulrike Ehlert. Discriminating stress from cognitive load using a wearable eda device. *IEEE Transactions on Information Technology in Biomedicine*, 14(2):410–417, 2010.
- [54] Eduardo Gil, Michele Orini, Raquel Bailon, José María Vergara, Luca Mainardi, and Pablo Laguna. Photoplethysmography pulse rate variability as a surrogate measurement of heart rate variability during non-stationary conditions. *Physiological measurement*, 31(9):1271, 2010.
- [55] Sara Campanella, Ayham Altaleb, Alberto Belli, Paola Pierleoni, and Lorenzo Palma. A method for stress detection using empatica e4 bracelet and machine-learning techniques. *Sensors*, 23(7), 2023.
- [56] Scipy. scipy.signal.resample. Consulta en: 2022.
- [57] Rajiv Ranjan Singh, Sailesh Conjeti, and Rahul Banerjee. A comparative evaluation of neural network classifiers for stress level analysis of automotive drivers using physiological signals. *Biomedical Signal Processing and Control*, 8(6):740–754, 2013.
- [58] Carlos Carreiras, Ana Priscila Alves, André Lourenço, Filipe Canento, Hugo Silva, Ana Fred, et al. BioSPPy: Biosignal processing in Python, 2015–. [Online; accessed `today`].

- [59] Dominique Makowski, Tam Pham, Zen J. Lau, Jan C. Brammer, François Lespinasse, Hung Pham, Christopher Schölzel, and S. H. Annabel Chen. NeuroKit2: A python toolbox for neurophysiological signal processing. *Behavior Research Methods*, 53(4):1689–1696, feb 2021.
- [60] Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122, 2013.
- [61] Sander Koelstra, Christian Muhl, Mohammad Soleymani, Jong-Seok Lee, Ashkan Yazdani, Touradj Ebrahimi, Thierry Pun, Anton Nijholt, and Ioannis Patras. Deap: A database for emotion analysis; using physiological signals. *IEEE transactions on affective computing*, 3(1):18–31, 2011.
- [62] T Lee Gilman, Razan Shaheen, K Maria Nylocks, Danielle Halachoff, Jessica Chapman, Jessica J Flynn, Lindsey M Matt, and Karin G Coifman. A film set for the elicitation of emotion in research: A comprehensive catalog derived from four decades of investigation. *Behavior research methods*, 49(6):2061–2082, 2017.
- [63] Alexandre Schaefer, Frédéric Nils, Xavier Sanchez, and Pierre Philippot. Assessing the effectiveness of a large database of emotion-eliciting films: A new tool for emotion researchers. *Cognition and emotion*, 24(7):1153–1172, 2010.
- [64] Antonio Maffei and Alessandro Angrilli. E-movie-experimental movies for induction of emotions in neuroscience: An innovative film database with normative data and sex differences. *Plos one*, 14(10):e0223124, 2019.
- [65] Lantana Dioren Rumpa, Adhi Dharma Wibawa, Mauridhi Heri Purnomo, and Harmelia Tulak. Validating video stimulus for eliciting human emotion: A preliminary study for e-health monitoring system. In *2015 4th International Conference on Instrumentation, Communications, Information Technology, and Biomedical Engineering (ICICI-BME)*, pages 208–213. IEEE, 2015.
- [66] Chih-Hung Wu, Yueh-Min Huang, and Jan-Pan Hwang. Review of affective computing in education/learning: Trends and challenges. *British Journal of Educational Technology*, 47(6):1304–1323, 2016.

- [67] Peter Lang. Behavioral treatment and bio-behavioral assessment: Computer applications. *Technology in mental health care delivery systems*, pages 119–137, 1980.
- [68] Georgios N Yannakakis. Enhancing health care via affective computing. 2018.
- [69] Nancy L Stein and Keith Oatley. Basic emotions: Theory and measurement. *Cognition & Emotion*, 6(3-4):161–168, 1992.
- [70] Kees Leidelmeijer. *Emotions: An experimental approach*. Tilburg University Press, 1991.
- [71] Paul Ekman. An argument for basic emotions. *Cognition & emotion*, 6(3-4):169–200, 1992.
- [72] Robert Plutchik. The nature of emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice. *American scientist*, 89(4):344–350, 2001.
- [73] Albert Mehrabian. Comparison of the pad and panas as models for describing emotions and for differentiating anxiety from depression. *Journal of psychopathology and behavioral assessment*, 19(4):331–357, 1997.
- [74] Martin Gjoreski, Hristijan Gjoreski, Mitja Luštrek, and Matjaž Gams. Continuous stress detection using a wrist device: in laboratory and real life. In *proceedings of the 2016 ACM international joint conference on pervasive and ubiquitous computing: Adjunct*, pages 1185–1193, 2016.
- [75] Shiyi Wu, Xiangmin Xu, Lin Shu, and Bin Hu. Estimation of valence of emotion using two frontal eeg channels. In *2017 IEEE international conference on bioinformatics and biomedicine (BIBM)*, pages 1127–1130. IEEE, 2017.
- [76] Henry Candra, Mitchell Yuwono, Rifai Chai, Ardi Handojoseno, Irraivan Elamvazuthi, Hung T Nguyen, and Steven Su. Investigation of window size in classification of eeg-emotion signal with wavelet entropy and support vector machine. In *2015 37th Annual international conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 7250–7253. IEEE, 2015.
- [77] Andrew Steptoe and Mika Kivimäki. Stress and cardiovascular disease. *Nature Reviews Cardiology*, 9(6):360–370, 2012.

- [78] Andrej Luneski, Panagiotis D Bamidis, and Madga Hitoglou-Antoniadou. Affective computing and medical informatics: state of the art in emotion-aware medical applications. *Studies in health technology and informatics*, 136:517, 2008.
- [79] RANA el KALIOUBY, Rosalind Picard, and Simon Baron-Cohen. Affective computing and autism. *Annals of the New York Academy of Sciences*, 1093(1):228–248, 2006.
- [80] Warner M Mampuya. Cardiac rehabilitation past, present and future: an overview. *Cardiovascular diagnosis and therapy*, 2(1):38, 2012.
- [81] Hindra Kurniawan, Alexandr V Maslov, and Mykola Pechenizkiy. Stress detection from speech and galvanic skin response signals. In *Proceedings of the 26th IEEE International Symposium on Computer-Based Medical Systems*, pages 209–214. IEEE, 2013.
- [82] Empatica E4. E4 - ibi expected signal. 2022.
- [83] Empatica E4. E4 data - hr.csv explanation. 2022.
- [84] Gary E Strangman, Vladimir Ivkovic, and Quan Zhang. Wearable brain imaging with multimodal physiological monitoring. *Journal of Applied Physiology*, 124(3):564–572, 2018.
- [85] A Otero, P Félix, and S Barro. Current state of critical patient monitoring and outstanding challenges. In *Handbook of Research on Developments in E-Health and Telemedicine: Technological and Social Perspectives*, pages 981–1003. IGI Global, 2010.
- [86] Mayo Clinic. Electroencefalografía (eeg). Consulta en: 2021.
- [87] Sapienza Universidad de Roma. Respuesta galvánica de la piel (gsr). Consulta en: 2021.
- [88] Jaskanwal Deep Singh Sara, Elad Maor, Diana Orbelo, Rajiv Gulati, Lliach O. Lerman, and Amir Lerman. Noninvasive voice biomarker is associated with incident coronary artery disease events at follow-up. *Mayo Clinic Proceedings*, 97(5):835–846, 2022.
- [89] Rod K. Dishman, Yoshio Nakamura, Melissa E. Garcia, Ray W. Thompson, Andrea L. Dunn, and Steven N. Blair. Heart rate variability, trait anxiety, and perceived stress among physically fit men and women. *International Journal of Psychophysiology*, 37(2):121–133, 2000.

- [90] Empatica. E4 data - ibi expected signal. Consulta en: 2021.
- [91] Empatica. What should i know to use eda data in my experiment? Consulta en: 2021.
- [92] Empatica. E4 data - eda expected signal. 2020.
- [93] Nicholas D Giardino, Paul M Lehrer, and Robert Edelberg. Comparison of finger plethysmograph to ecg in the measurement of heart rate variability. *Psychophysiology*, 39(2):246–253, 2002.
- [94] Sanjeev Kumar, John Buckley, John Barton, Melusine Pigeon, Robert Newberry, Matthew Rodencal, Adhurim Hajzeraaj, Tim Hannon, Ken Rogers, Declan Casey, Donal O’sullivan, and Brendan O’Flynn. A wristwatch-based wireless sensor platform for iot health monitoring applications. *Sensors*, 20, 03 2020.
- [95] Ph.D Bryn Farnsworth. What is gsr (galvanic skin response) and how does it work? Consulta en: 2021.
- [96] Tobii Connect. Galvanic skin response (gsr). Consulta en: 2021.
- [97] Ross Arena, Mark Williams, Daniel E Forman, Lawrence P Cahalin, Lola Coke, Jonathan Myers, Larry Hamm, Penny Kris-Etherton, Reed Humphrey, Vera Bittner, et al. Increasing referral and participation rates to outpatient cardiac rehabilitation: the valuable role of healthcare professionals in the inpatient and home health settings: a science advisory from the american heart association. *Circulation*, 125(10):1321–1329, 2012.
- [98] Richard V Milani, Carl J Lavie, Mandeep R Mehra, and Hector O Ventura. Impact of exercise training and depression on survival in heart failure due to coronary heart disease. *The American journal of cardiology*, 107(1):64–68, 2011.
- [99] Christian Albus. Psychological and social factors in coronary heart disease. *Annals of Medicine*, 42(7):487–494, 2010. PMID: 20839918.
- [100] David S Krantz, Kerry S Whittaker, and David S Sheps. Psychosocial risk factors for coronary heart disease: Pathophysiologic mechanisms. 2011.
- [101] Chih-Hung Wu, Yi-Lin Tzeng, and Yueh Min Huang. Understanding the relationship between physiological signals and digital game-based learning outcome. *Journal of Computers in Education*, 1(1):81–97, 2014.
- [102] Fitbit. Dispositivos fit-bit. Consulta en: 2022.

- [103] Edge Impulse. What is edge machine learning (edge ml)? Consulta en: 2021.
- [104] PluxBiosignals. Wearables. Consulta en: 2022.
- [105] Empatica. E4 data - hr.csv explanation. Consulta en: 2021.
- [106] Empatica. E4 wristband for developers. Consulta en: 2021.
- [107] Empatica. E4 for developers: Android.
- [108] Edge Impulse. We put ml into real products.
- [109] Yoann Baveye, Emmanuel Dellandrea, Christel Chamaret, and Liming Chen. Liris-accede: A video database for affective content analysis. *IEEE Transactions on Affective Computing*, 6(1):43–55, 2015.
- [110] Kylee T Ack Baraly, Lydia Muyingo, Christine Beaudoin, Sanaz Karami, Melina Langevin, and Patrick SR Davidson. Database of emotional videos from ottawa (devo). *Collabra: Psychology*, 6(1), 2020.
- [111] Peter Kuppens, Francis Tuerlinckx, Michelle Yik, Peter Koval, Joachim Coosemans, Kevin J Zeng, and James A Russell. The relation between valence and arousal in subjective experience varies with personality and culture. *Journal of personality*, 85(4):530–542, 2017.

Lista de Figuras

2.1.	Representación de los estados emocionales basados en clasificación de <i>valence</i> (unpleasant - pleasant)- <i>arousal</i> (deactivation - activation) [17]. .	8
2.2.	Resumen de las señales usadas en trabajos previos para la clasificación de cuadros de estrés y sus resultados	12
2.3.	Resumen de los algoritmos de machine learning y las features analizadas en trabajos previos.	13
3.1.	Descripción general de la solución. La primera parte busca encontrar un modelo que permita reconocer estrés, mientras que en la segunda se describe el proceso de diseño de una aplicación que permita implementar el modelo. Fuente propia	18
3.2.	Concepto de la solución propuesta.	19
3.3.	Diagrama de bloques dentro de la aplicación.	19
3.4.	Relación entre el módulo de “ <i>Communication</i> ” y “ <i>Database</i> ”.	20
3.5.	Diagrama de bloques del módulo de procesamiento de datos. Fuente propia	22
4.1.	Proceso de creación del modelo de para el reconocimiento de estrés. . .	26
4.2.	Protocolo del estudio realizado para la captura de las señales fisiológicas en condiciones de Estrés. Fuente: [43]	27
4.3.	Diagrama de bloques del preprocesamiento de la señal.	28
4.4.	Proceso de segmentación en ventanas de 30s de duración e incremento de 15s.	29
5.1.	Protocolo del experimento realizado para evaluar el sistema de detección de estrés propuesto.	39
5.2.	Resultados del modelo 1 durante el primer experimento para dos vídeos. W_n corresponde a cada una de las ventanas, U_n a cada uno de los usuarios. En naranja los casos de predicción fallidos y en verde los casos de predicción acertados..	42

5.3.	Resultados de clasificación de los vídeos para los datos obtenidos en el experimento 1 con el modelo 5. Al inicio de cada tabla se presentan los resultados de <i>valence-arousal</i> de cada vídeo, seguido de los resultados de clasificación. W_n corresponde a la ventana n-ésima de cada vídeo. U_n corresponde a los diferentes participantes.	44
5.4.	Resultados de predicción para 4 usuarios siguiendo el orden original del experimento, usando el modelo del Modelo 5. “2” corresponde a estrés y “1” corresponde a no estrés. En verde los resultados correctamente clasificados.	46
5.5.	Evolución de los aciertos de cada modelo a través de los múltiples usuarios. Para este caso, el modelo 5 (en azul continuo) con los datos balanceados $NE = ST$ vuelve a ser el mejor con una media del 57,7% de aciertos.	48
5.6.	Resultados para el video 4 con el modelo que combina dos algoritmos en dos configuraciones distintas: Arriba el Modelo 1 con un algoritmo LR y DT, features con señales EDA y BVP. Abajo el Modelo 2 con un algoritmo LR, DT y features de señales EDA, BVP y ACC_x . Cada ventana (W_n) con “2” significa estrés y “1”, no estrés. En verde las ventanas clasificadas correctamente.	50

Lista de Tablas

4.1.	Lista de features.	31
4.2.	Evaluación de los diferentes modelos propuestos con cada uno de los algoritmos de ML para la clasificación de estrés. Abreviaciones: SVM= Support Vector Classification, KNN = k-nearest neighbour, RF = Random Forest, LDA = Linear discriminant analysis, DT = Decision Tree, LR = Linear Regression.	35
4.3.	Lista de features usadas como input de los algoritmos de ML.	36
5.1.	Resultados de la prueba SAM de los 17 participantes con los 6 vídeos propuestos.	41
5.2.	Resultados del entrenamiento del modelo 5, hecho con un algoritmo DT, usando un dataset balanceado entre muestras de estrés y no estrés.	45
5.3.	Evolución del desempeño de los diferentes modelos, en función de cada vídeo del experimento. V_n corresponde a los vídeos 1 al 6 del experimento 1 y los modelos 1-5 corresponden a los modelos descritos en el capítulo 4. $NE > ST$ significa que el dataset no estaba balanceado y $NE = ST$ que el dataset estaba balanceado. En negrita el mejor modelo.	47
5.4.	Desempeño de los modelos en función del porcentaje de aciertos usando dos algoritmos de clasificación. En el modelo 1* se usaron los modelos 5 y 4, con features de las señales EDA y BVP. En el caso del segundo modelo, se usaron los modelos 5 y 4, con features de las señales EDA, BVP y ACC_x	49