



# Sistema de aprendizaje federado jerárquico para la estimación de la somnolencia mediante Interfaces Cerebro-Máquina

Máster Universitario en Tecnologías de  
Análisis de Datos Masivos: Big Data

## Trabajo Fin de Máster

Autor:

José Manuel Hidalgo Rogel

Tutor/es:

Sergio López Bernal

Dr. Gregorio Martínez Pérez



**Facultad  
Informática  
Universidad  
Murcia**

2 de Junio de 2022

# Sistema de aprendizaje federado jerárquico para la estimación de la somnolencia mediante Interfaces Cerebro-Máquina

---

Trabajo Fin de Máster

**Autor**

José Manuel Hidalgo Rogel

**Tutor/es**

Sergio López Bernal

*Departamento de Ingeniería de la Información y las Comunicaciones*

Dr. Gregorio Martínez Pérez

*Departamento de Ingeniería de la Información y las Comunicaciones*



Máster Universitario en Tecnologías de Análisis de Datos Masivos: Big Data



Murcia, 2 de Junio de 2022

*A mi familia por todas las facilidades y el apoyo que me han brindado para que pueda estudiar lo que realmente me apasiona. También a Sergio, Enrique, Mario, Alberto y Gregorio por los incontables consejos y la supervisión que me han dedicado.*

*Cualquier tecnología lo suficientemente  
avanzada es indistinguible de la magia*

(Arthur C. Clarke)

# Declaración firmada sobre originalidad del trabajo

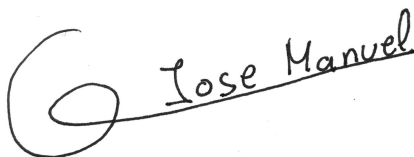
D./Dña. **José Manuel Hidalgo Rogel**, con DNI **48848317C**, estudiante de la titulación de **Máster Universitario en Tecnologías de Análisis de Datos Masivos: Big Data** de la Universidad de Murcia y autor del TFM titulado “**Sistema de aprendizaje federado jerárquico para la estimación de la somnolencia mediante Interfaces Cerebro-Máquina**”.

De acuerdo con el Reglamento por el que se regulan los Trabajos Fin de Grado y de Fin de Máster en la Universidad de Murcia (aprobado C. de Gob. 30-04-2015, modificado 22-04-2016 y 28-09-2018), así como la normativa interna para la oferta, asignación, elaboración y defensa de los Trabajos Fin de Grado y Fin de Máster de las titulaciones impartidas en la Facultad de Informática de la Universidad de Murcia (aprobada en Junta de Facultad 27-11-2015)

DECLARO:

Que el Trabajo Fin de Máster presentado para su evaluación es original y de elaboración personal. Todas las fuentes utilizadas han sido debidamente citadas. Así mismo, declara que no incumple ningún contrato de confidencialidad, ni viola ningún derecho de propiedad intelectual e industrial

Murcia, a 2 de Junio de 2022

A handwritten signature in black ink, consisting of a large, stylized 'G' followed by the words 'Jose Manuel' written in a cursive script.

Fdo.: José Manuel Hidalgo Rogel  
Autor del TFM

# Lista de Acrónimos y Abreviaturas

<b>ADAS</b>	Sistemas Avanzados de Asistencia a la Conducción.
<b>BCI</b>	Interfaces Cerebro-Máquina.
<b>BLE</b>	Bluetooth Low Energy.
<b>DL</b>	Deep Learning.
<b>EEG</b>	Electroencefalografía.
<b>EOG</b>	Electrooculografía.
<b>FL</b>	Federated Learning.
<b>GDPR</b>	Reglamento General de Protección de Datos.
<b>HIPAA</b>	Ley de Portabilidad y Responsabilidad de Seguros Médicos.
<b>IoT</b>	Internet de las cosas.
<b>KSS</b>	Escala de Somnolencia de Karolinska.
<b>LSL</b>	Lab Streaming Layer.
<b>ML</b>	Machine Learning.
<b>MLP</b>	Perceptrón Multicapa.
<b>MSLT</b>	Test de Latencias Múltiples del Sueño.
<b>MWT</b>	Test de Mantenimiento de la Vigilia.
<b>NASA-TLX</b>	NASA Task Load Index.
<b>PERCLOS</b>	PERcentage of eye CLOSure.
<b>PSD</b>	Densidad Espectral de Potencia.
<b>TFG</b>	Trabajo Fin de Grado.
<b>TFM</b>	Trabajo Fin de Máster.
<b>TL</b>	Transfer Learning.
<b>UDP</b>	User Datagram Protocol.

# Resumen

Las tecnologías basadas en Machine Learning (ML) han sufrido un rápido crecimiento a lo largo de las dos últimas décadas, lo que ha convertido a los datos en el activo más valioso de las empresas. Un correcto tratamiento de los mismos permite a las empresas hacer análisis exhaustivos y certeros que garanticen una firme toma de decisiones. Por estos motivos, la inteligencia artificial se extiende a sectores de aplicación cada vez más amplios. Pero, a la misma vez, aumenta la preocupación por la privacidad de los usuarios y la confidencialidad de los datos. Así, los gobiernos han establecido reglamentos de privacidad, como el Reglamento General de Protección de Datos (GDPR) en la UE o la ley Ley de Portabilidad y Responsabilidad de Seguros Médicos (HIPAA) en EE. UU.

Estas nuevas medidas hacen que sea muy difícil integrar datos dispersos de diferentes instituciones. La metodología tradicional de ML, consistente en recopilar y agregar los datos en una ubicación central, donde potentes ordenadores puedan entrenar y construir modelos predictivos, está siendo progresivamente reemplazada por nuevas alternativas emergentes como son los enfoques distribuidos y federados, que han ganado mucho peso en el ámbito empresarial y de la investigación. Particularmente, Federated Learning (FL) es la opción más novedosa, puesto que permite el entrenamiento de modelos predictivos de forma colaborativa y descentralizada, lo que permite disponer de una mayor privacidad y seguridad de los datos de cada cliente, pues estos nunca abandonan su dispositivo. Es por esto que un gran número de empresas están investigando sobre esta metodología tanto para securizar sus sistemas como para promover la colaboración con otras empresas.

Son varias las motivaciones que contribuyen a la realización de este Trabajo Fin de Máster (TFM). En primer lugar, la labor social mediante la mejora en la seguridad de los datos que ofrece el uso de FL, lo que beneficiaría tanto a las empresas como a los ciudadanos cuyos datos sean tratados en el proceso. Segundo, la posibilidad de estudiar más a fondo esta nueva metodología, que se encuentra en desarrollo y con grandes vistas a futuro. Finalmente, debido a la novedad de este campo, hay ciertos sectores de aplicación no explorados en el estado del arte para los que es de gran interés comprobar la viabilidad de esta solución y avanzar en este sentido en la literatura.

A partir de estas motivaciones, se realizó un estudio del estado del arte para determinar dónde se podría hacer una puesta de valor para esta metodología. En tal sentido, se identificó la falta de un sistema de colaboración mediante FL jerárquico capaz de preservar la privacidad de los datos tanto dentro de cada empresa como entre ellas. Al mismo tiempo, la aplicación de FL en la detección de la somnolencia es

bastante limitada, y lo es aún más empleando los datos relativos a las señales cerebrales de los conductores como base de las predicciones. La razón de emplear señales cerebrales para evaluar la somnolencia se justifica por la neurociencia cognitiva, que ha demostrado que los diferentes niveles de actividad cerebral están relacionados con los diferentes estados cognitivos del sujeto. Debido a esto, existe la necesidad de estudiar las señales cerebrales, donde el método más típico para la adquisición de señales es la Electroencefalografía (EEG).

Con el objetivo de abordar las limitaciones mencionadas previamente, este trabajo presenta el diseño de un sistema basado en FL jerárquico que permite la colaboración entre empresas. Este sistema presenta como característica principal el mantenimiento de la privacidad de los datos tanto dentro de cada empresa como entre ellas. Asimismo, tras el diseño, se implementó y validó el sistema mediante un escenario de aplicación basado en la predicción de la somnolencia al volante usando Interfaces Cerebro-Máquina (BCI) y FL.

La implementación del escenario consiste en tres casos de uso, en los que se implantará FL a nivel intraempresa, interempresa y de forma jerárquica, respectivamente, que servirán a su vez como hitos para la validación del sistema. En primer lugar, se obtuvieron las señales fisiológicas de varios conductores mediante BCIs, procedentes de un *dataset* público. Posteriormente, las señales se procesaron para eliminar errores y ruido, permitiendo extraer los aspectos clave (*features*) mediante las que determinará el estado del conductor. Los datos de cada conductor fueron repartidos entre las diferentes empresas según los requisitos del escenario. Posteriormente, se diseñaron los experimentos a efectuar en cada caso de uso para validar el sistema. Para que los resultados y las conclusiones fuesen lo más confiables posible, se establecieron *i*) mecanismos de reproducibilidad, *ii*) una arquitectura común del modelo predictivo y *iii*) métricas de evaluación para los modelos entrenados.

A partir de la experimentación efectuada se midió el rendimiento de los modelos federados en los diferentes casos de uso. Se estudió su comportamiento tanto para conductores conocidos, con los que se entrenó el modelo, como para nuevos, ajenos a la fase de entrenamiento del modelo. En tal sentido, se efectuaron varias pruebas que dejaban fuera de la fase de entrenamiento a más conductores de manera incremental para validar la evolución del rendimiento de dichos modelos.

Como resultado de este trabajo y de la experimentación realizada, se observa que la colaboración entre empresas supone una mejora en las predicciones para los conductores nuevos, siendo hasta un 3% y un 5% mejor el rendimiento de los modelos generales con ML y FL respectivamente. Además, se verifica la viabilidad de mantener la seguridad tanto a nivel intraempresa como interempresa, pues la implementación del sistema federado jerárquico afecta levemente al rendimiento de los modelos predictivos federados, con pérdidas del 0% al 2%.

---



# Extended Abstract

Machine Learning (ML) technologies have experienced significant growth over the past two decades. The success of these technologies has been driven by the availability of large amounts of data combined with the increased technological capabilities of current computers. In this sense, data has become the most valuable resource of any company. A proper data analysis allows them to make exhaustive and accurate diagnoses that guarantee robust decision making, forecast what may happen in the future, and measure the performance of the actions carried out. This is the reason why data-driven companies such as Apple, Microsoft, Amazon, Google, and Meta have established themselves as the world's largest companies when it comes to stock market capitalization.

As artificial intelligence broadens into more and more fields, users are increasingly concerned that their private information will be used for commercial and political purposes without their permission. In the legal sphere, governments have established privacy regulations, such as the General Data Protection Regulation (GDPR) in Europe or the Health Insurance Portability and Accountability Act (HIPAA) in the United States. In this emerging legislative landscape, the collection and exchange of data between different organizations are becoming increasingly difficult. In addition, the sensitive nature of certain data such as financial transactions or medical records, prohibits the free flow of data and forces them to exist in isolated silos maintained by data owners. These new measures make the integration of scattered data from different institutions extremely challenging. Moreover, certain limitations have been identified over the years related to physical aspects, such as the need for supercomputers for the most complex problems and no fault tolerance, resulting in single points of failure. Therefore, traditional methodology of collecting and aggregating data in a central location, where powerful computers can train and build predictive models, is being progressively replaced by new innovative alternatives.

More recently, distributed and federated ML approaches are being preferred since they support broader data analysis due to their superior scalability. Distributed Learning consist of having centralized data but distributing the model training to different nodes, while Federated Learning (FL) is about having decentralized data and training and in effect having a central model. FL is the most innovative option, this achieves greater privacy and security of each client's data, as data never leave the client's device. This is why a large number of companies are investigating this methodology both to secure their systems and to encourage collaboration with other companies. In fact, this methodology not only respects the new legislation but also protects the privacy of the

data of the different clients that participate in the process.

The steps involved in training a federated model are as follows. First, a central server distributes its model to the clients that will participate in the federated training process. Then, each client receiving the model fine-tunes the model locally using its data, obtaining a new gradient. Then, all clients send back their corresponding gradient to the central server, which receives and aggregates them, thus creating a new main model. Finally, this process is repeated, where each iteration is referred to as a *round*. The FL process achieves convergence of the main model by running numerous rounds until the loss function converges, thus completing the collaborative training process.

The aspects that motivated the development of this research include a socio-economic contribution due to the improvement in data security offered by the use of FL. Consequently benefiting both companies and citizens whose data are processed. As well as the possibility of a further and deeper analysis of this new methodology, which is under constant development and with great potential for the future. Finally, as a result of the novelty of this field, certain application areas have not yet been explored in the state of the art. Therefore, it is highly beneficial to test the feasibility of these solutions and to make some progress in the literature in this regard.

Despite the progress achieved in the literature, some open challenges still require more attention from the research point of view. Specifically, the lack of a federated collaborative system between companies, able to preserve the privacy of data both within and between companies, has been detected. In addition, there is a shortage of papers that involve brain signal classification or drowsiness detection while driving utilizing FL.

To solve the limitations detected in the state of the art, the main objective of this work is to design a federated learning system to secure data in business-to-business training of predictive models. This system must guarantee that data privacy is maintained both within and between companies, so it must be implemented following a federated hierarchical scheme. The term *hierarchical* is used in some of the literature to define a federated training scheme divided into client-edge-cloud layers. It consists of a central server and certain edge servers, each with a disjoint set of clients. Through this architecture, it is intended to take advantage of the resources of the intermediate nodes of the network (edge) to perform partial aggregations of the federated model making use of the data of its associated clients. With this system, after  $e_1$  local training epochs on each client, each edge server aggregates the local gradients of its clients. After each  $e_2$  model aggregations at the edge, the cloud server aggregates all edge models, which means that communication with the cloud occurs every  $e_1 \cdot e_2$  updates. The above-described system is adapted to the business environment, where each edge server is a company participating in the federation, the cloud refers to the central server and clients are each of the data devices involved in the process.

Therefore, the designed system must be validated, for this reason, a practical scenario that will foster the interest of several companies in collaborating is designed. Particularly, the scenario consists in drowsiness detection while driving using Brain-Computer

Interfaces (BCIs) and FL, based on the UE 2019/2144 norm, which requires the inclusion of a driver drowsiness and distraction warning system in all new homologated cars from May 2022 onwards. This scenario is used to measure the performance of the hierarchical federated system models compared to predictive models which are trained following the traditional ML methodology, as well as determining whether there is a benefit in the collaboration for the companies.

The reason behind employing brain signals to assess drowsiness is justified by cognitive neuroscience, which has shown that the different levels of brain activity are related to the different cognitive states of the subject. Due to this, there is a need to study brain signals, where the most typical method for signals acquisition is Electroencephalography (EEG). The signals are studied in different frequency bands, since different levels of brain activity need to be studied, being the lower frequency rhythms (delta, theta, and alpha) directly related to the states of relaxation and drowsiness. Apart from brain signals, other physiological signals can be used to estimate the cognitive status of subjects, such as eye movements, commonly acquired through Electrooculography (EOG). BCIs allow the monitoring of those signals passively to determine patterns of cognitive activity such as drowsiness. Furthermore, they are used to carry out medical evaluations and have the ability to interact with external devices based on brain activity and in real-time. The BCI workflow consists of the following steps. First, the brain signals are acquired through the electrodes placed on the subject's scalp, then the processing phase of the acquired data is next. In this phase, data are processed to eliminate noise and artifacts. Then, features are extracted to be used later by the predictive models in order to make assessments. Finally, based on the result of the classification, the system produces a response, for example, notifying the user or sending an order to the car.

In this sense, the solution is implemented as follows. First, after establishing the scenario, three use cases were defined to achieve the implementation of the proposed hierarchical federated system. In the first one, there is no collaboration between the companies, where data is only secured within each one of the companies. In the second, a collaboration between organizations takes place, maintaining data privacy only between companies. Finally, the third use case combines the two previous ones with the designed hierarchical architecture, achieving privacy at both organization layers, inside each company and between them.

With the scenario and three use cases defined, which serve as milestones for the implementation and validation of the system, the following procedures were carried out for its implementation. First, a publicly available dataset was selected to enable the scenario, the SEED-VIG dataset, which consists of 23 experiments among 21 different subjects (two subjects repeated the experiment). Each experiment has about two hours of EEG/EOG signals recorded while the subjects were using a driving simulator. Second, Python was selected as the general-purpose programming language together with Keras and Flower as specific frameworks for deep learning and FL, respectively.

Then, a data mining process was performed for the brain signals of the drivers in

---

the dataset. The process consisted of three phases: *i*) preprocessing of the data to eliminate noise and artifacts, improving data quality, *ii*) feature extraction that allow determining the driver's alert level through a predictive model, specifically a Multi-Layer Perceptron, and *iii*) dataset splitting among the different companies according to the requirements of the scenario. After this, reproducibility mechanisms, fairness in the comparisons by using a common architecture of the predictive model and the performance metrics to be used for experimentation (accuracy, sensitivity, specificity, and f1-score) were established. Finally, the experiments to validate the proposed hierarchical federated system were designed for each use case. For the first one, where FL is applied within each company, customized models are studied for each driver, a common model per company without preserving data security and using FL are compared. For the second use case, the performance of a centralized model with and without data privacy employing data of the three companies is compared to the federated per-company model of the first use case. Finally, for the third use case, the performance of the centralized federated model of the previous use case is studied with respect to its construction with the hierarchical federated system designed in this work.

To evaluate the system, the performance of the federated models was measured in comparison with traditional models with no data privacy. The behavior of the models was studied both for known drivers, the ones with whom the model was trained, and for unknown drivers, who are left outside the training phase of the model. Several tests were performed leaving an increasingly amount of drivers out of the training phase each time.

As a result of this work and from the experimentation carried out, it can be observed that the collaboration between companies leads to an improvement in the assessment of the cognitive state of both known and unknown drivers. In fact, the accuracy of the general models is better for unknown drivers compared to the models trained with the data of just one company, specifically, between 3% for ML and 5% for FL. However, general models have a slight performance penalty for known drivers, in this case, the accuracy loss is around 4% for ML and 6% for FL. This implies that federated models tend to generalize better than the traditional ones, which adapt better to the peculiarities of the drivers used for training. In addition, the feasibility of maintaining safety both within and between companies is verified since the implementation of the hierarchical federated system presents a minimal impact on the performance of the predictive models (1% or 2%). Thus, it is concluded that the proposed hierarchical FL mechanism works and is valid for this scenario, from these data, it is determined that collaboration between companies to determine drowsiness in new drivers, either through a federated or traditional methodology, is beneficial. Moreover, the performance difference between the traditional and federated versions is small enough to always consider the federated methodology as a good option, since it adds security benefits to the process.

---

# Índice general

<b>Lista de Acrónimos y Abreviaturas</b>	<b>vi</b>
<b>1 Introducción</b>	<b>1</b>
<b>2 Background</b>	<b>5</b>
2.1 Neurociencia cognitiva aplicada a la somnolencia . . . . .	5
2.1.1 Métodos para la cuantificación de la somnolencia . . . . .	5
2.2 La Interfaz Cerebro-Máquina . . . . .	6
2.2.1 Estimación de la somnolencia mediante una BCI . . . . .	8
2.3 Aprendizaje federado . . . . .	9
<b>3 Estado del arte</b>	<b>15</b>
3.1 Empleo de topologías jerárquicas . . . . .	15
3.2 Soluciones adaptadas al sector sanitario . . . . .	16
<b>4 Análisis de objetivos y metodología</b>	<b>18</b>
4.1 Objetivos del trabajo . . . . .	18
4.2 Metodología . . . . .	19
<b>5 Diseño y resolución del trabajo realizado</b>	<b>21</b>
5.1 Escenario y casos de uso . . . . .	21
5.1.1 UC1: Federated learning intraempresa . . . . .	21
5.1.2 UC2: Federated learning interempresa . . . . .	22
5.1.3 UC3: Federated learning jerárquico . . . . .	23
5.2 Conjunto de datos y herramientas empleadas . . . . .	24
5.2.1 Conjunto de datos . . . . .	24
5.2.2 Lenguajes de programación y frameworks empleados . . . . .	25
5.3 Minería de los datos . . . . .	25
5.3.1 Preprocesado y extracción de características . . . . .	26
5.3.2 Análisis del comportamiento de las características . . . . .	28
5.3.3 División del dataset . . . . .	28
5.4 Equidad en la experimentación . . . . .	30
5.4.1 Arquitectura común para el modelo predictivo . . . . .	30
5.5 Diseño de la experimentación . . . . .	32
5.5.1 Experimentos para el UC1 . . . . .	32
5.5.2 Experimentos para el UC2 . . . . .	32

5.5.3	Experimentos para el UC3 . . . . .	33
<b>6</b>	<b>Análisis de resultados</b>	<b>34</b>
6.1	Resultados para el UC1 . . . . .	35
6.2	Resultados para el UC2 . . . . .	36
6.3	Resultados para el UC3 . . . . .	38
6.4	Discusión de los resultados . . . . .	41
<b>7</b>	<b>Conclusiones y vías futuras</b>	<b>42</b>
	<b>Bibliografía</b>	<b>44</b>

---

# Índice de figuras

2.1	Sistema Internacional 10-20 para la colocación de los electrodos [1]. . .	7
2.2	Análisis de los elementos presentes en una señal EOG para la que se muestra tanto el canal vertical como el horizontal [2]. . . . .	9
2.3	Arquitectura de un sistema de aprendizaje federado horizontal. . . . .	12
2.4	Arquitectura de un sistema de aprendizaje federado jerárquico. . . . .	14
5.1	Escenario para el UC1 para el que se hace entrenamiento federado independientemente para cada empresa. . . . .	22
5.2	Escenario para el UC2 para el que se hace entrenamiento federado entre las empresas, pero no dentro de cada una de ellas. . . . .	23
5.3	Escenario para el UC3 para el que se hace entrenamiento federado tanto entre las empresas como dentro de cada una mediante el sistema jerárquico diseñado. . . . .	24
5.4	Procedimiento para el cálculo de las <i>features</i> de las cinco bandas del PSD. . . . .	26
5.5	Resultado para la detección de los parpadeos en el canal vertical del EOG. . . . .	27
5.6	Análisis de las <i>features</i> para determinar el escenario IID o non-IID. . . . .	29
5.7	División de los sujetos para el escenario de FL. Las etiquetas D y S hacen referencia a la cantidad de muestras con estado <i>despierto</i> y <i>somnoliento</i> respectivamente. . . . .	29
5.8	Arquitectura del MLP tras optimizar sus hiperparámetros mediante optimización bayesiana y validación cruzada de cinco pliegues. . . . .	31
6.1	Comparativa de rendimiento del UC2 entre los modelos por empresa con FL, general con ML y general con FL. . . . .	39
6.2	Comparativa del rendimiento entre los modelos general con FL del UC2 y el general con FL jerárquico del UC3. . . . .	40

# Índice de tablas

2.1	Distinción de las bandas de frecuencia cerebrales según [3]. . . . .	8
2.2	Comparativa entre frameworks de Federated Learning. . . . .	13
3.1	Estado del arte de las aplicaciones en el ámbito del aprendizaje federado.	17
6.1	Comparativa de resultados del UC1 entre los enfoques individuales, por empresa con ML y por empresa con FL. . . . .	36
6.2	Sumario del rendimiento en accuracy de los diferentes experimentos realizados a lo largo de los tres casos de uso. Los indicadores $C$ y $N$ hacen referencia al rendimiento en conductores conocidos y nuevos, respectivamente. . . . .	41



# 1 Introducción

Las tecnologías basadas en Machine Learning (ML) han sufrido un rápido crecimiento a lo largo de las dos últimas décadas. El éxito de estas tecnologías se ha visto impulsado por la disponibilidad de grandes cantidades de datos junto con el aumento de la potencia de cálculo que ofrecen los ordenadores. Debido a estos motivos, los datos se han convertido en el activo más valioso de las empresas. Un correcto tratamiento de los mismos les permite hacer análisis exhaustivos y certeros que garanticen una firme toma de decisiones, prever lo que pueda ocurrir en el futuro y medir el rendimiento de las acciones llevadas a cabo. Es por esto que las empresas que disponen y trabajan con grandes cantidades de datos como Apple, Microsoft, Amazon, Google y Meta se afianzan como las mayores compañías del mundo en cuanto a capitalización.

Mientras que la inteligencia artificial se extiende a sectores de aplicación cada vez más amplios, aumenta la preocupación por la privacidad de los usuarios y la confidencialidad de los datos. A los usuarios les preocupa cada vez más que su información privada sea utilizada con fines comerciales y políticos sin su permiso. En el ámbito legal, los gobiernos han establecido reglamentos de privacidad, como el Reglamento General de Protección de Datos (GDPR) (UE) [4] o Ley de Portabilidad y Responsabilidad de Seguros Médicos (HIPAA) (EE. UU.) [5]. A partir de estas leyes surgen cuestiones relativas a la propiedad de los datos y quién tiene derecho a emplearlos para crear tecnologías de aprendizaje automático.

En este nuevo panorama legislativo, la metodología tradicional de ML que consiste en recopilar y agregar los datos en una ubicación central, donde potentes ordenadores puedan entrenar y construir modelos predictivos es cada vez más difícil, por lo que la forma tradicional de trabajar está siendo progresivamente reemplazada por nuevas alternativas emergentes. Al mismo tiempo, la naturaleza sensible de ciertos datos tales como las transacciones financieras y los registros médicos, prohíbe la libre circulación de datos y obliga a que estos existan en silos de datos aislados mantenidos por los propietarios de los mismos. Además, a lo largo de los años, se han identificado ciertas limitaciones tanto en aspectos físicos, como la necesidad de supercomputadores para los problemas más complejos y la no tolerancia a fallos (single point of failure) [6, 7, 8], junto a los recientes problemas legislativos sobre la privacidad de los datos.

Recientemente, los enfoques distribuidos y federados de ML están siendo las opciones preferidas, ya que permiten un análisis de datos más amplio debido a su mejor escalabilidad. De estos, Federated Learning (FL) es la opción más novedosa, puesto que permite entrenar un modelo predictivo de forma colaborativa y con una mayor privacidad y seguridad de los datos de cada cliente, pues estos nunca abandonan su

dispositivo. Es por esto que un gran número de empresas están investigando sobre esta metodología tanto para securizar sus sistemas como para abrir puertas a la colaboración con otras empresas. Puesto que no solo respeta la nueva legislación, sino también protege la privacidad de los datos de los diferentes clientes que participen en el proceso.

Hay varias razones que contribuyen a la realización de este Trabajo Fin de Máster (TFM). Primero, la labor social debido a la mejora de la seguridad de los datos mediante el uso de FL, que beneficiaría tanto a las empresas como a los ciudadanos cuyos datos se procesan en el proceso. Segundo, la posibilidad de profundizar en el estudio de esta nueva metodología, que está en desarrollo y tiene grandes perspectivas de futuro. Finalmente, debido a la novedad de este campo, existen ciertas áreas de aplicación que no han sido suficientemente estudiadas, por lo que es de gran interés comprobar la viabilidad de aplicar esta técnica a esas áreas y avanzar en esta dirección en la literatura.

En esta perspectiva, se ha realizado un estudio acerca del estado del arte de FL a fin de identificar carencias y vías de trabajo. En relación a esta idea y a pesar de los avances logrados en los trabajos estudiados de la literatura, se han detectado una serie de retos que requieren de mayor investigación. Concretamente, se ha detectado la carencia de un sistema de colaboración federado entre empresas capaz de preservar la privacidad de los datos tanto dentro de cada empresa como entre ellas, junto con la escasez de estudios que validen sistemas implementados mediante topologías jerárquicas, que están muy alineadas con los intereses de colaboración empresariales. Además se tienen pocos trabajos involucrados en la clasificación de las señales cerebrales o la detección de somnolencia al volante, siendo nula la existencia de artículos que traten ambos temas a la vez, es decir, la detección de la somnolencia al volante mediante señales cerebrales y Interfaces Cerebro-Máquinas (BCIs).

Para solventar las limitaciones detectadas, este TFM tiene como objetivo principal diseñar un sistema que mediante FL permita la colaboración entre empresas para el entrenamiento de modelos predictivos. Este sistema debe garantizar que se mantiene la privacidad de los datos tanto dentro de cada empresa como entre ellas, por lo que se debe implementar siguiendo un esquema jerárquico federado. Para validar el sistema se plantea un escenario de aplicación que ponga en vista el interés en colaborar por parte de varias empresas. Este escenario será empleado para medir los beneficios de efectuar una colaboración, junto con el rendimiento del sistema federado frente al entrenamiento de modelos predictivos que siguen la metodología tradicional de ML. El escenario en cuestión es la predicción de la somnolencia al volante mediante BCI y FL, que se seleccionó en relación con la otra limitación mencionada previamente junto con la normativa UE 2019/2144 [9], que obliga a implantar sistemas de advertencia de la somnolencia y distracciones del conductor en todos los nuevos coches homologados a partir de mayo de 2022. La razón de emplear señales cerebrales para evaluar la somnolencia se justifica mediante las bases que establece la neurociencia cognitiva, que ha demostrado que los diferentes niveles de actividad cerebral están relacionados con los diferentes estados cognitivos del sujeto. Debido a esto, existe la necesidad de

---

estudiar las señales cerebrales, donde el método más típico para la adquisición de señales es la Electroencefalografía (EEG). Además de las señales cerebrales, se pueden utilizar otras señales fisiológicas para estimar el estado cognitivo de los sujetos, como los movimientos oculares, comúnmente adquiridos a través de la Electrooculografía (EOG). Las BCIs permiten monitorizar esas señales de forma pasiva para determinar patrones de actividad como la somnolencia.

En este sentido, se procede al desarrollo de la solución. En primer lugar, tras determinar el escenario, se definieron tres casos de uso para lograr la implementación del sistema jerárquico propuesto. Cada uno de ellos supera progresivamente al anterior tanto en complejidad como en el uso de FL. En el primero de ellos no hay colaboración entre las empresas, únicamente se privatizan los datos dentro de cada una de ellas. En el segundo, se da una colaboración entre empresas, manteniendo la privacidad de los datos entre las empresas. Finalmente, el tercer caso de uso une los dos anteriores, logrando la arquitectura jerárquica deseada mediante una colaboración entre empresas que mantiene la privacidad en los dos niveles, dentro de cada empresa y entre empresas.

A continuación, se seleccionó un conjunto de datos público para crear el escenario y las herramientas necesarias para implementar la solución. Se realizó una comparativa entre las opciones disponibles y se seleccionaron aquellas que mejor se adaptaban a los requisitos de la investigación. Respecto al *dataset*, se acabó por seleccionar SEED-VIG [3], el cuál consta de 23 experimentos entre 21 sujetos diferentes (dos sujetos repitieron el experimento). Cada experimento tiene una duración de dos horas de señales fisiológicas, grabadas mientras los sujetos utilizaban un simulador de conducción y recogidas mediante una BCI capaz de medir los datos de EEG y EOG. En cuanto a herramientas para la implementación, se seleccionó Python como lenguaje de programación de propósito general junto con Keras y Flower como frameworks de Deep Learning (DL) y FL, respectivamente.

Después, se efectuó un proceso de minería de datos para el dataset seleccionado. El proceso consta de tres fases: *i*) preprocesamiento de los datos para eliminar valores erróneos y mejorar su calidad, *ii*) extracción de las características de las señales fisiológicas EEG y EOG de los conductores que permitirán determinar el estado del conductor mediante un modelo predictivo, concretamente un Perceptrón Multicapa (MLP) y *iii*) división de los datos entre las diferentes empresas según los requisitos de cada caso de uso. Tras esto, se establecieron los mecanismos de reproducibilidad, equidad en las comparaciones mediante el uso de una arquitectura común del modelo predictivo y las métricas de rendimiento a emplear para la experimentación (accuracy, sensibilidad, especificidad y f1-score).

Por último, se diseñaron los experimentos a realizar en cada caso de uso para validar el sistema jerárquico propuesto. Para el primero, donde se aplica FL dentro de cada empresa, se estudian modelos personalizados para cada conductor, un modelo general por empresa con ML y por empresa con FL. Respecto al segundo caso de uso, se compara el rendimiento de un modelo general, tanto con ML como FL, a partir de los datos de conductores de las tres empresas frente al modelo por empresa federado del

---

caso de uso anterior. Finalmente, para el caso de uso tercero, se estudia el rendimiento del modelo federado general del caso de uso anterior respecto a su construcción con el sistema federado jerárquico diseñado en este trabajo.

A partir de los experimentos definidos se midió el rendimiento de los modelos federados frente a las versiones tradicionales sin privacidad de datos. Se estudió el comportamiento de los modelos tanto para conductores conocidos, con los que se entrenó el modelo, como para nuevos, ajenos a la fase de entrenamiento del modelo. Para obtener unos resultados más fiables, se realizaron varias pruebas en las que se iba incrementando el número de conductores que no participaban en la fase de entrenamiento.

En cuanto a los resultados obtenidos, el uso de modelos generales mejora el accuracy de los modelos por empresa para conductores nuevos hasta un 3% para ML y un 5% en el caso de FL. Sin embargo, para la evaluación de los conductores conocidos, los modelos generales presentan un rendimiento inferior de hasta un máximo del 4% en ML y un 6% en FL. A partir de estos datos, se determina que la colaboración entre las empresas para determinar la somnolencia en nuevos conductores, bien sea mediante una metodología federada o tradicional, es beneficiosa. Además, la pérdida de rendimiento de la versión jerárquica del caso de uso tercero respecto al segundo, que solo mantiene la privacidad entre empresas, es únicamente del 0% al 2%. Mediante estas observaciones, se puede concluir que el mecanismo de FL jerárquico propuesto funciona y es válido para este escenario, ya que logra aportar un beneficio tanto en seguridad como rendimiento a las empresas participantes. En relación a la idea anterior, es preferible el uso de FL frente a un método tradicional de ML pese a la leve pérdida de rendimiento que puede presentar en algunas situaciones.

---

## 2 Background

A lo largo de este apartado se detallan las nociones necesarias para comprender el trabajo posteriormente realizado. En primer lugar se presentan los conceptos de neurociencia cognitiva y somnolencia. Seguido a esto, se dan a conocer las Interfases Cerebro-Máquina, su funcionamiento y cómo es posible cuantificar la somnolencia mediante ellas. Finalmente, se presenta FL, que es la metodología de aprendizaje automático cuya viabilidad y rendimiento se estudian a lo largo de este trabajo.

### 2.1 Neurociencia cognitiva aplicada a la somnolencia

La neurociencia cognitiva es un área de estudio interdisciplinar, donde convergen dos disciplinas: *i)* la neurociencia, encargada de estudiar las células del sistema nervioso y su organización dentro de circuitos funcionales que procesan la información y median en el comportamiento y *ii)* la psicología cognitiva, que estudia las funciones mentales superiores como la concentración, la somnolencia, etc [10, 11].

Una de sus ramas de estudio se centra en identificar qué sucesos dan lugar a un aumento o decremento en el estado de alerta y qué aspectos clave se dan a nivel fisiológico en cada uno de los diferentes niveles del mismo. El nivel de alerta establecido como *usual* es el estado de vigilia. Se define como un estado consciente, y se caracteriza por un nivel de actividad alto en relación al intercambio de información entre el sujeto y su ambiente. Por otra parte, la somnolencia se define como la habilidad de transición de la vigilia al sueño o como la disminución del estado de alerta dando lugar a las primeras fases del sueño. Cuando se habla de somnolencia al volante, realmente se hace referencia a la somnolencia aguda, que determina la tendencia de un sujeto a quedarse dormido en un momento puntual, no persistiendo en el tiempo. Así, de ahora en adelante para este trabajo, cuando se hable de somnolencia se hará referencia a la somnolencia aguda.

#### 2.1.1 Métodos para la cuantificación de la somnolencia

La cuantificación de la somnolencia es un proceso complejo, para cuya medición existen en la literatura tres aproximaciones [12, 13, 14].

El primer método consiste en mediciones del comportamiento del sujeto, donde se miden aspectos como los bostezos, el tiempo de reacción ante un evento, el ritmo cardíaco, la conductancia de la piel o la frecuencia del parpadeo y su duración. En la

práctica, son muchos los test disponibles, siendo los más aplicados para la detección de somnolencia los siguientes: test de tiempo de reacción [15, 16, 17] y mediciones del PERcentage of eye CLOSure (PERCLOS) [18, 19, 20]. En concreto, PERCLOS es una medida psicofisiológica del sujeto que cuantifica el porcentaje de tiempo que el sujeto permanece con los ojos cerrados al menos en un 80% durante el intervalo de tiempo de la medición [21]. A partir de dicha medida, se establece un umbral a partir del cuál se considera que el sujeto presenta un estado somnoliento.

El segundo método para la cuantificar la somnolencia consiste en una autoevaluación con escalas. Esta técnica se basa en preguntar al sujeto cómo de despierto se encontraría en diversas situaciones o qué tan somnoliento se ha sentido en los últimos minutos. La autoevaluación con escalas es el procedimiento más sencillo y barato a la hora de realizar la experimentación sobre los sujetos. Sin embargo, tienen una parte negativa, ya que el sujeto tiende a sobreestimar sus capacidades, por lo que es esperable obtener una evaluación de somnolencia por debajo del valor real. A la hora de estimar la somnolencia al volante, estas son las autoevaluaciones más empleadas: Escala de Somnolencia de Karolinska (KSS) [22, 23] y NASA Task Load Index (NASA-TLX) [24, 25].

Como tercer método y con el objetivo de conseguir la objetividad a la hora de cuantificar la somnolencia, se diseñaron los test neurofisiológicos. Los más populares son el Test de Latencias Múltiples del Sueño (MSLT) y el Test de Mantenimiento de la Vigilia (MWT). Este tipo de pruebas requieren de la colocación de electrodos en el sujeto, una habitación adaptada y un técnico experto en el manejo del sistema, lo que hace prácticamente inviable su uso en la detección de la somnolencia al volante. A pesar de esto, en los últimos años han aparecido soluciones que permiten realizar este tipo de experimentos debido a la gran reducción tanto en coste como en infraestructura necesarios. La solución en cuestión es el uso de sistemas BCI. Estos sistemas son un compromiso entre los test neurofisiológicos (MWT y MSLT) de ámbito sanitario y la portabilidad y accesibilidad requeridos para los estudios realizados por equipos de investigación.

## 2.2 La Interfaz Cerebro-Máquina

Las Interfaces Cerebro-Máquina, en inglés *Brain-Computer Interfaces*, son tecnologías que han favorecido enormemente el desarrollo de la neurociencia cognitiva. Estos sistemas brindan la capacidad de analizar las diferencias de potencial eléctrico producidas por neuronas o grupos de estas a través de electrodos.

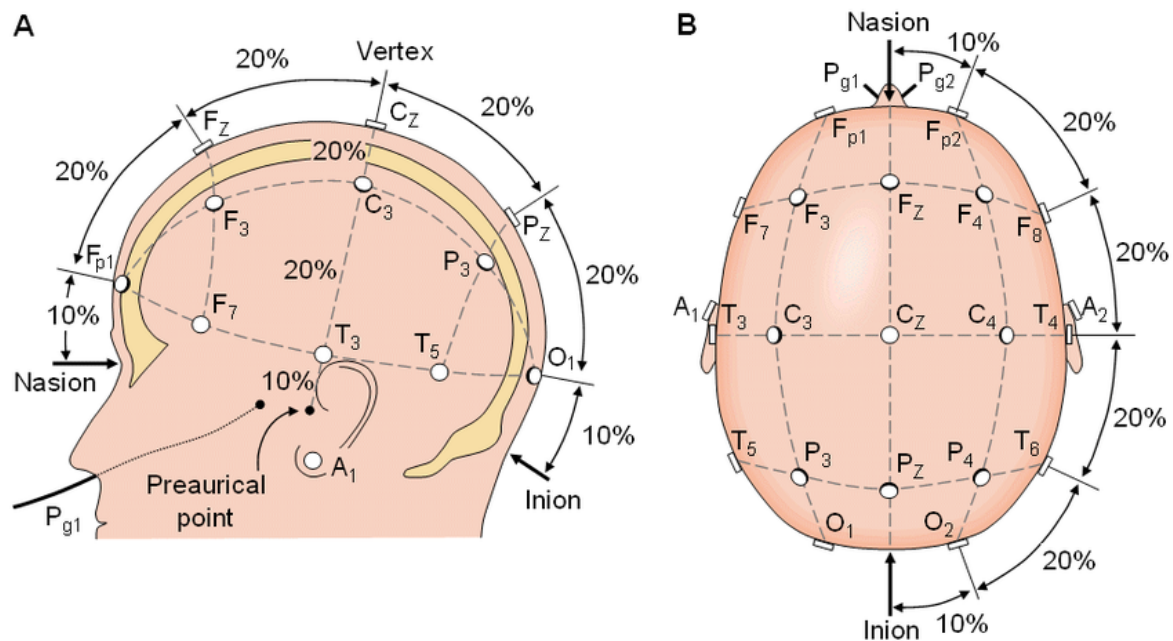
Una BCI [26, 27] es un dispositivo cuya finalidad radica en la lectura y el análisis de la actividad cerebral a través de electrodos. Entre sus aplicaciones se encuentran: *i)* monitorizar la actividad de forma pasiva y poder así analizar patrones de actividad cognitiva, motora o emocional; *ii)* evaluar o diagnosticar las capacidades cognitivas o emocionales en base a patrones cerebrales detectados permitiendo la identificación de anomalías y *iii)* habilitan la interacción con dispositivos, procesando la actividad

---

cerebral en tiempo real [28].

Existen dos tipos de BCIs en base al grado de invasividad de los electrodos dispuestos en el sujeto. Las BCIs invasivas, que sitúan los electrodos en las distintas capas del cráneo dependiendo de su propósito, requieren un proceso quirúrgico para su implantación pero a cambio ofrecen una calidad máxima de resolución y precisión de las muestras. Por otro lado, los electrodos de las BCIs no invasivas se colocan directamente en el cuero cabelludo del sujeto, lo que evita un procedimiento quirúrgico pero a costa de la calidad de las señales, que se ve perturbada debido al movimientos del sujeto o la interposición del cabello con los electrodos. Es por esto que los datos de las BCIs no invasivas deben ser procesados posteriormente para eliminar los artefactos causados por la actividad de los sujetos. En este estudio se hará uso de BCIs no invasivas.

La disposición de los electrodos en los sistemas no invasivos está regido por el Sistema Internacional 10-20 definido por Malmivuo et al. en [1] (ver Figura 2.1), el cuál define la colocación de los electrodos extracraneales.



**Figura 2.1:** Sistema Internacional 10-20 para la colocación de los electrodos [1].

Cualquier sistema BCI no invasivo centrado en adquisición de actividad neuronal consta de las mismas fases de funcionamiento, independientemente del hardware y software que emplee. En primer lugar se debe adquirir la señal cerebral a partir de los electrodos dispuesto en el sujeto. A continuación, dichos datos se envían a un dispositivo de control. Como protocolo de comunicación para el envío se suele emplear Bluetooth Low Energy (BLE) mientras que User Datagram Protocol (UDP) y Lab Streaming Layer (LSL) se usan para la transmisión de los datos entre el controlador y los otros dispositivos de la red. Una vez recibidos los datos, se procesan las señales para

aumentar su calidad, eliminado cualquier tipo de ruido, distorsiones o errores que no tienen correlación con el sujeto o la región cerebral estudiada. Tras el procesamiento, se extraen las características de la señal, en inglés *features*, que es la información relevante que será el objeto de estudio y que variará en base al experimento que se realice. Finalmente, estas *features* se suelen emplear para entrenar un modelo predictivo, que clasifica la señal en base al vector de características. Esta clasificación será interpretada por una interfaz de control y producirá alertas acerca del estado del sujeto (somnolencia para este trabajo) o en comandos significativos para una prótesis, por ejemplo.

### 2.2.1 Estimación de la somnolencia mediante una BCI

Referente a las señales biológicas que se pueden obtener mediante una BCI para el estudio de la somnolencia, se analiza el EEG y el EOG, ya que serán empleados en este trabajo.

Respecto al EEG, es una técnica capaz de detectar y analizar la actividad cerebral, basada en el registro de actividad bioeléctrica. Estas señales son captadas por los electrodos de la BCI dispuestos en el cuero cabelludo. Cada electrodo envía una señal a la BCI, que muestra la fluctuación de las ondas cerebrales, lo que permite su estudio en toda clase de experimentos.

Estas señales son analizadas en diferentes bandas de frecuencia (ver Tabla 2.1), ya que los distintos estados cognitivos del sujeto están relacionados con diferentes niveles de actividad cerebral [11, 29]. En particular, los ritmos más bajos (theta, delta y gamma) son característicos en etapas de sueño y relajación, siendo particularmente notable el ritmo alfa cuando el sujeto tiene los ojos cerrados. Al contrario, los ritmos más altos (beta y gamma), implican estados de concentración e incluso estrés del sujeto. Mediante la transformada rápida de Fourier se calcula la Densidad Espectral de Potencia (PSD), esta medida es utilizada para medir la energía en cada banda de frecuencia de las señales cerebrales, proporcionando buenos resultados a la hora de estimar la somnolencia [29, 30].

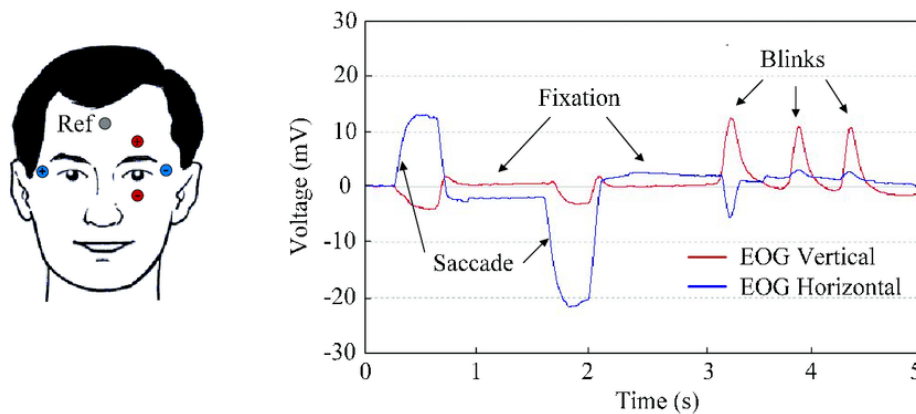
**Tabla 2.1:** Distinción de las bandas de frecuencia cerebrales según [3].

Nombre	Frecuencia (Hz)	Estado del sujeto
Ritmo Delta ( $\delta$ )	1 - 4	Sueño profundo.
Ritmo Theta ( $\theta$ )	4 - 8	Sueño ligero y sueño REM.
Ritmo Alfa ( $\alpha$ )	8 - 14	Estado de relajación y poco nivel de actividad mental.
Ritmo Beta ( $\beta$ )	14 - 31	Estado mental de concentración.
Ritmo Gamma ( $\gamma$ )	> 31	Actividades cerebrales intensas y estrés.

Como complemento al EEG, en una BCI se pueden colocar electrodos faciales en el sujeto con el objetivo de recopilar otro tipo de señales fisiológicas de manera simul-



tánea, siendo la EOG una de las opciones más frecuentes. Esta señal permite medir los movimientos de los músculos oculares mediante el registro de la diferencia de potencial existente entre la córnea y la retina. En la Figura 2.2 se muestra un ejemplo de señal EOG con la interpretación de diferentes eventos, siendo el más importante la identificación de los parpadeos donde cada pico de potencial en el canal vertical se corresponde con uno. Mediante esta técnica es posible monitorizar aspectos como la concentración, el parpadeo y los movimientos oculares sacádicos (movimientos rápidos del ojo).



**Figura 2.2:** Análisis de los elementos presentes en una señal EOG para la que se muestra tanto el canal vertical como el horizontal [2].

Con esta información, se podrá distinguir un estado de vigilia por parte de un conductor siempre y cuando en su actividad cerebral predominen los ritmos beta y gamma, esto estará acompañado de un ritmo de parpadeos constante y de corta duración junto con una presencia mayor de movimientos oculares sacádicos. Al contrario, si el conductor presenta un estado somnoliento, se reflejará en su actividad cerebral por la atenuación de las actividades gamma y beta, junto con un incremento en las alfa, theta y delta. Por último, es importante destacar que en estado de somnolencia, el tiempo que se pasa con los ojos cerrados será mayor, además de presentar una menor varianza de la señal EOG (menos movimientos sacádicos) [10, 11, 19, 29, 31].

## 2.3 Aprendizaje federado

La metodología tradicional de ML consiste en recopilar y agregar los datos en una ubicación central, donde potentes ordenadores entrenan modelos predictivos. En los últimos tiempos, los enfoques distribuidos y federados de ML están ganando relevancia, ya que permiten un análisis de datos más amplio, siendo FL la opción más novedosa, ya que permite el tratamiento de datos de diferentes clientes de forma segura y privada.

En contraste con los enfoques tradicionales, que son centralizados, el aprendizaje

federado [32, 33, 34] es un entorno de aprendizaje automático en el que el objetivo es entrenar un modelo de forma colaborativa y con los datos descentralizados. En este entorno, los datos de entrenamiento permanecen distribuidos entre un gran número de clientes. Cada participante (cliente) de la federación calcula de forma independiente una actualización del modelo actual basada en sus datos locales, y comunica esta actualización a un servidor central, donde las actualizaciones del lado del cliente se agregan para calcular un nuevo modelo global. Gracias a esto, se logra que los datos de cada cliente nunca abandonen su dispositivo.

El aprendizaje federado considera el entrenamiento de varios tipos de modelos: modelos lineales, máquinas de vectores soporte, árboles de decisión y redes neuronales [35]. Los modelos lineales se dividen principalmente en tres categorías: regresión lineal, ridge y lasso. En comparación con otros modelos, el modelo lineal es simple y fácil de implementar, y es eficaz para implementar el aprendizaje federado. FL también se ha adaptado para entrenar máquinas de vector soporte y árboles de decisión simples o múltiples, como los Gradient Boosting Decision Tree o Random Forest. Las redes neuronales (MLP, redes recurrentes o redes convolucionales) son una tendencia popular del aprendizaje automático en la actualidad. En el entorno federado, el entrenamiento de modelos DL es lo más popular y extendido.

FL se puede clasificar en una u otra categoría en función de las características de distribución de los datos [32, 33, 35, 36]. Para comprender las diferencias entre las categorías es necesario emplear la siguiente notación: sea la matriz  $D_i$  la que denote los datos que posee un cliente  $i$ . Cada fila de la matriz representa una observación, y cada columna representa una característica. Al mismo tiempo, algunos conjuntos de datos pueden contener también datos etiquetados. El espacio de características se denomina  $X$ , el espacio de etiquetas se denomina  $Y$  y el espacio de identificación de las muestras se denomina  $I$ . A continuación se presentan las tres categorías de FL en relación con la distribución de los datos del escenario:

- *Horizontal Federated Learning*: Escenarios en los que los conjuntos de datos comparten el mismo espacio de características  $(X, Y)$  pero un espacio diferente en las muestras  $(I)$ . Queda definido formalmente mediante la siguiente expresión:

$$X_i = X_j, Y_i = Y_j, I_i \neq I_j, \forall D_i, D_j, i \neq j$$

Se asemeja a la situación en la que los datos están divididos horizontalmente dentro de una vista tabular. De hecho, la palabra "horizontal" procede del término "partición horizontal", muy utilizado en el contexto de la vista tabular tradicional de una base de datos.

- *Vertical Federated Learning*: Aplicable a los casos en los que dos conjuntos de datos comparten el mismo espacio  $I$  de muestras pero difieren en el espacio de características  $(X, Y)$ .

$$X_i \neq X_j, Y_i \neq Y_j, I_i = I_j, \forall D_i, D_j, i \neq j$$

Similar al horizontal, en este caso la "vertical" proviene del término "partición vertical", del contexto de la vista tabular en una base de datos, donde las columnas de una tabla se dividen verticalmente en diferentes grupos y cada columna representa una característica de todas las muestras.

- *Federated Transfer Learning*: En la práctica, se dan situaciones en las que no hay suficientes características o muestras compartidas entre las partes participantes. Para estos casos, se puede construir un modelo de aprendizaje federado combinado con Transfer Learning (TL), que es un método que permiten transferir conocimientos adquiridos gracias a la resolución de ciertos problemas para resolver otros. Gracias a esto, se permite traspasar el conocimiento entre partes que no comparten ni las muestras ni el espacio de características, logrando mejor rendimiento. La combinación de aprendizaje federado y TL se denomina *Federated Transfer Learning* y queda definido mediante la expresión:

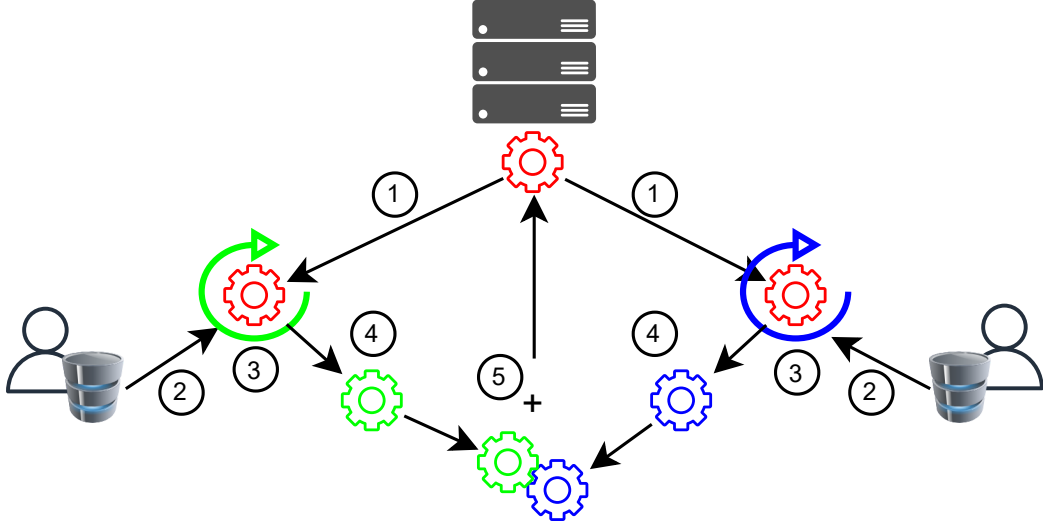
$$X_i \neq X_j, Y_i \neq Y_j, I_i \neq I_j, \forall D_i, D_j, i \neq j$$

*Horizontal Federated Learning* es la categoría de FL que originalmente fue propuesta en el 2016 por Google en [37], y posteriormente fueron emergiendo el resto de categorías. Es por esto que ha sido la más estudiada y probada en la literatura. Es también sobre la que se centrará este TFM y, por este motivo, sobre la que se centrarán las subsiguientes explicaciones.

Dado que FL horizontal será la categoría principal de este estudio, es necesario conocer cómo se entrena un modelo mediante este sistema. La Figura 2.3 muestra la arquitectura de un sistema de aprendizaje federado horizontal, que es independiente del modelo predictivo que se emplee. Los pasos de los que constan el entrenamiento son los siguientes:

- Paso 1: El servidor distribuye su modelo a los clientes que participarán en el proceso de entrenamiento federado.
- Paso 2, 3 y 4: Los participantes reciben el modelo, calculan localmente los nuevos gradientes mediante sus datos y los envían de vuelta al servidor central.
- Paso 5: El servidor recibe todos los gradientes de los modelos locales y realiza una agregación, creando así un nuevo modelo principal.
- Paso 6: Repetir el proceso. La iteración de los pasos del 1 al 5 se denomina *ronda*. FL logra la convergencia del modelo colaborativo mediante la ejecución de varias rondas hasta que la función de pérdida converge, completando así todo el proceso de entrenamiento.

Cabe destacar la existencia de frameworks dedicados a la implementación de soluciones para FL, suelen tener como características el ser de código abierto y el estar



**Figura 2.3:** Arquitectura de un sistema de aprendizaje federado horizontal.

programados en Python y sobre alguno de sus frameworks de ML/DL como Tensorflow o Pytorch. Principalmente, se han explorado a fondo tres frameworks, cuyas ventajas y desventajas se muestra en la Tabla 2.2. Aunque también cabe destacar la existencia de otras alternativas como FATE [38] que está dirigido a un nivel más industrial y es de pago o PySyft, [39] que presta gran atención en los protocolos de comunicación y la seguridad del proceso.

A partir del diagrama anterior (Figura 2.3), se puede identificar un claro escenario de comunicación *muchos a uno*, lo que puede llegar a generar sobrecargas en la red durante el entrenamiento de un modelo federado. Tal es el caso, que se han diseñado esquemas *jerárquicos* donde se intenta abordar esta problemática. Una de las propuestas de sistemas jerárquicos consiste en agrupar clientes con características o situación geográfica común y entrenar con ellos un modelo federado más personalizado. Un ejemplo de esto serían las sugerencia personalizadas de *Gboard* [43, 44]. Otra propuesta es la de definir un esquema de entrenamiento federado dividido en subniveles cliente-edge-cloud, donde el cloud actúa como servidor central, que dispone de  $E$  servidores edge, cada uno de ellos con un conjunto disjunto de  $N$  clientes [45, 46]. Mediante esta arquitectura (ver Figura 2.4), se pretende aprovechar los recursos de los nodos intermedios de la red (edge) para realizar agregaciones parciales del modelo federado haciendo uso de los datos de sus clientes asociados. Con este sistema, tras  $e_1$  épocas locales de entrenamiento en cada cliente, cada servidor edge agrega los gradientes locales de sus clientes. Después de cada  $e_2$  agregaciones de modelos en el edge, el servidor de la nube agrega todos los modelos del edge, lo que significa que la comunicación con la nube se produce cada  $e_1 \cdot e_2$  actualizaciones.

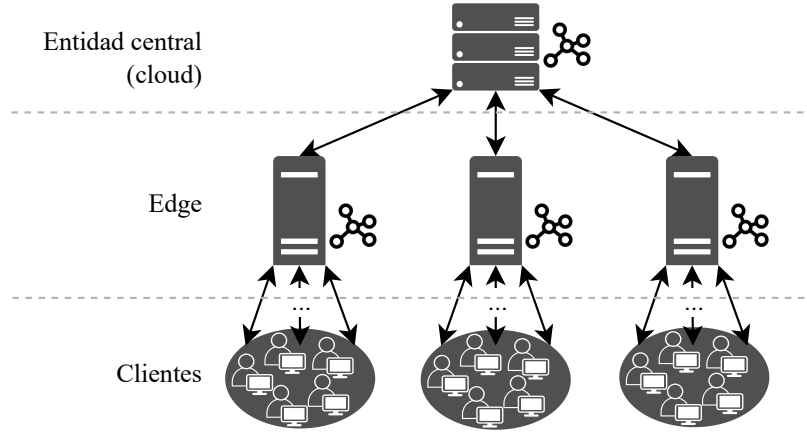
Existen diferentes algoritmos a la hora de agregar los pesos locales en un único modelo, siendo *FedAvg* el más popular y extendido, donde la importancia que se le

**Tabla 2.2:** Comparativa entre frameworks de Federated Learning.

Framework	Ventajas	Desventajas
Flower [40]	<ul style="list-style-type: none"> <li>• Curva de aprendizaje media, API de alto nivel.</li> <li>• Posibilidad de configuración avanzada.</li> <li>• Adaptabilidad con TensorFlow, SKLearn y PyTorch.</li> <li>• Buena documentación y ejemplos de uso.</li> </ul>	<ul style="list-style-type: none"> <li>• Mala escalabilidad para simulaciones en una máquina (solucionado a partir de la versión 0.17.0).</li> </ul>
IBM Federated Learning [41]	<ul style="list-style-type: none"> <li>• Computación en la nube de IBM.</li> <li>• Curva de aprendizaje baja, fácil configuración de los experimentos mediante GUI.</li> <li>• Adaptabilidad con TensorFlow, SKLearn y PyTorch.</li> <li>• Buena documentación.</li> </ul>	<ul style="list-style-type: none"> <li>• Pocas opciones de configuración.</li> <li>• Recursos limitados en el plan gratuito.</li> </ul>
TensorFlow Federated [42]	<ul style="list-style-type: none"> <li>• Muchas posibilidades de configuración.</li> <li>• Buena documentación y ejemplos de uso.</li> <li>• Buena escalabilidad.</li> </ul>	<ul style="list-style-type: none"> <li>• Difícil construcción de un dataset federado.</li> <li>• Solo permite modelos de TensorFlow.</li> <li>• Curva de aprendizaje alta, API de nivel intermedio/bajo.</li> </ul>

da a cada resultado local viene ponderado por la cantidad de datos que ese cliente ha empleado durante su entrenamiento local [37]. Este algoritmo tiene la limitación de favorecer a los dispositivos que más datos aportan, por lo que puede llegar a situaciones de no convergencia. Es por esto que se han propuesto otras alternativas como *FedProx* [47], que introduce un término de regularización que penaliza cambios grandes en el modelo, o *q-FedAvg* [48], que penaliza a los clientes que peor rendimiento tienen, para mejorar así la función de pérdida. Cuanto mayor sea  $q$ , más peso tendrán los que peor funcionan.

Una suposición típica es que los participantes son honestos mientras que el servidor es honesto pero curioso; por lo tanto, no se permite la fuga de información de ningún participante al servidor [36]. No obstante, la seguridad y la privacidad son pilares fundamentales de FL. Esto requiere modelos y análisis de seguridad que proporcionen garantías de privacidad significativas, disponiendo por tanto de técnicas adicionales



**Figura 2.4:** Arquitectura de un sistema de aprendizaje federado jerárquico.

para garantizar la seguridad de los datos. A continuación, se presentan diferentes técnicas de privacidad para FL [32, 36]. La primera, *Secure Aggregation* [49], consiste en encriptar las actualizaciones de pesos de cada cliente. Para poder desencriptarlos es necesario combinar las actualizaciones de otros clientes. De esta forma, el modelo global no aprende cambios específicos de un cliente, sino agregaciones de todos ellos, aumentando la seguridad de los datos. Otra técnica es *Differential Privacy* [50], consiste en añadir ruido cuidadosamente seleccionado a las salidas para anonimizar y privatizar los datos. Por último, *Homomorphic Encryption* [51, 52] es un esquema de cifrado que permite realizar ciertas operaciones algebraicas sobre el contenido cifrado que al desencriptarlo da como resultado una salida idéntica a la producida si las operaciones se hubieran realizado con los datos originales.

Por último, dentro de las problemáticas que surgen al hacer uso de FL, ya se ha visto: *i)* cómo se aborda el *fairness* del proceso mediante distintos algoritmos de agregación y *ii)* qué mecanismos de seguridad adicionales se pueden agregar al proceso. No obstante, hay una tercera cuestión a considerar que puede afectar significativamente al rendimiento de FL. Esta es la situación de datos non-IID, que se define como la heterogeneidad de los datos entre clientes a la hora de medir un mismo fenómeno. Esto, aplicado al caso de la detección de la somnolencia al volante, se traduce en que puede que existan diferentes conductores, ambos etiquetados con estado somnoliento, pero cuyas características extraídas difieran entre sí. Esto supone una dificultad añadida al emplear FL, ya que cada modelo local se adaptará a las particularidades de un conductor y es posible que la agregación de modelos se vea afectada negativamente si las actualizaciones locales agregadas son muy diferentes entre sí. Esto produciría situaciones de no convergencia de los modelos federados [32, 53, 54].

## 3 Estado del arte

FL es un mecanismo innovador que permite entrenar modelos predictivos colaborativos sobre datos descentralizados entre participantes sin comprometer su privacidad y seguridad. Por este motivo, tiene aplicaciones prometedoras en muchos campos como las finanzas, atención sanitaria, educación, Internet de las cosas (IoT), edge computing y blockchain, donde los datos no pueden agregarse directamente para entrenar modelos de ML. Esta sección revisa y estudia varias aplicaciones actuales donde se ha empleado FL, poniendo especial interés en *i)* trabajos que empleen topologías jerárquicas y *ii)* soluciones adaptadas a la parte sanitaria, concretamente somnolencia al volante o uso de señales cerebrales.

### 3.1 Empleo de topologías jerárquicas

Como se comentó en la Sección 2.3, existen dos versiones de topologías dinámicas, la *clusterizada*, que consiste en agrupar clientes con características o situación geográfica común y entrenar modelos personalizados para cada cluster y la *pura*, donde se define un esquema de entrenamiento federado dividido en subniveles. En la Tabla 3.1 se presentan diferentes trabajos, varios de ellos recopilados por Abdulrahman et al. en [55]. Para cada trabajo se indica cuáles son los objetivos y el interés para este estudio.

En primer lugar se presentan los trabajos relativos al FL jerárquico clusterizado, que son los más abundantes en la literatura. Destacan los trabajos de [43, 44] para la aplicación de teclado *Gboard* de Google, ya que, como se mencionó en 2.3, fue Google quién propuso esta metodología de entrenamiento colaborativo y con datos descentralizados. El *Gboard* es uno de los grandes casos de éxito que se tienen en esta disciplina, que se sigue utilizando en la actualidad desde su implantación sobre el año 2017, siendo en parte el que motivó a otros investigadores y empresas a indagar sobre FL. De los dos trabajos sobre el *Gboard*, es particularmente beneficioso el uso de FL en [44], donde se entrena un modelo de red neuronal más complejo, demostrando un mejor rendimiento que un modelo entrenado con datos centralizados. Además, es bien conocido que esta aplicación hace agrupamiento de sujetos por localización geográfica para lograr ofrecer unas recomendaciones más personalizadas. Otro trabajo que emplea una topología jerárquica clusterizada es el de [56], en este caso para agrupar pacientes con características similares y entrenar un modelo de DL de manera federada personalizado para cada clúster de sujetos, mejorando el rendimiento de las predicciones.

Finalmente, relativo a FL jerárquico *puro*, únicamente se tienen dos trabajos a nivel

teórico, estos son [45, 46]. Es aquí donde se define la esquema de entrenamiento federado dividido en subniveles cliente-edge-cloud. En concreto, en [45] se detalla por completo el algoritmo *HierFAVG*, propuesto como esquema de entrenamiento jerárquico, se estudian la convergencia de esta nueva solución y se indica que en la experimentación realizada, este sistema ofrecía unas mejores prestaciones en tiempo y energía requeridos respecto a FL. Por otra parte, [46] define de manera similar la topología por niveles de agregación, haciendo un posterior análisis sobre cuestiones de privacidad del sistema que proponen y promoviendo que se estudie más a fondo este esquema de entrenamiento debido a que suple las carencias de privacidad que podrían existir en FL.

## 3.2 Soluciones adaptadas al sector sanitario

Dentro de los trabajos acerca de somnolencia al volante o uso de señales cerebrales mostrados en la Tabla 3.1, se destaca [57], que propone un algoritmo genético que logra reducir la transferencia de datos entre los clientes y el servidor a cambio de un compromiso en el tiempo de convergencia y el rendimiento obtenido por el modelo, útil en situaciones donde se pretenda sacrificar algo de rendimiento y tiempo de ejecución a cambio de un menor gasto de ancho de banda. Mientras que [58] y [59] proponen una representación común de los datos procedentes de diferentes dispositivos y para datos de diferentes sujetos, respectivamente. A diferencia de los estudios analizados hasta el momento en la tabla, [60] es el primero en reúne varios de los requisitos para que un estudio sea de interés, ya que combina un clustering de sujetos como topología jerárquica a la vez que se realizan predicciones acerca de la somnolencia al volante. En el sistema que proponen, cada cliente dispone de una red convolucional personalizada para la detección de la fatiga. Esos datos se cargan posteriormente al edge, que se encarga de configurar los parámetros necesarios para que el servidor central agregue los parámetros de los modelos y actualice los modelos de los clientes.

A partir del análisis del estado del arte realizado, se identifica al FL como una metodología realmente útil e interdisciplinar. Que habilita no solo el entrenamiento de modelos sin comprometer la privacidad de los datos de los clientes, sino que también la colaboración entre empresas de un mismo sector FL horizontal o de distintos FL vertical o *transfer*. Sin embargo, se han detectado una serie de carencias tras el análisis de la literatura. Esta son *i)* disponer de trabajos donde se mida el rendimiento de una solución de FL jerárquico *puro*, *ii)* la escasez de trabajos dedicados a la estimación de la somnolencia al volante o que empleen señales cerebrales y *iii)* un sistema que permita mantener la privacidad de los datos simultáneamente dentro de una empresa y entre las empresas que colaboren en el proceso federado. Ya que en todos los estudios analizados se trabaja FL a un único nivel de federación. Mientras que en un escenario más realista, lo más común sería disponer de varias empresas interesadas en la generación de un modelo federado, cada una de ellas con su dataset distribuido de sus clientes. Dichas



empresas desearán mantener la privacidad de sus datos frente a las otras empresas pero a su vez también dentro de la empresa.

Por este motivo, el principal objeto de estudio de este TFM es la implementación y evaluación de un sistema de colaboración federado entre empresas. Capaz de preservar la privacidad de los datos tanto dentro de cada empresa como entre ellas. Para conseguirlo, se adaptará el funcionamiento de FL jerárquico descrito por [45, 46]. Donde, para este sistema, los servidores edge serán cada una de las empresas participantes en la federación. Además, será evaluado en un escenario de predicción de la somnolencia al volante mediante señales cerebrales, de esta forma también se conseguirá ampliar la literatura en este sentido tal y como se puede observar en la última fila de la Tabla 3.1, ya que actualmente no se tiene un trabajo con FL que lo estudie.

**Tabla 3.1:** Estado del arte de las aplicaciones en el ámbito del aprendizaje federado.

Referencia	Objetivo	FL jerárquico	Somnolencia al volante	Señales cerebrales
Yang et al. (2018) [43]	Modelado del lenguaje: Sugerencia de búsqueda en el teclado.	Clusterizado	No	No
Hard et al. (2018) [44]	Modelado del lenguaje: Predicción de la siguiente palabra.	Clusterizado	No	No
Chen et al. (2020) [56]	Diagnóstico de enfermedades mediante <i>wearables</i> .	Clusterizado	No	No
Liu et al. (2020) [45]	Diseño del esquema jerárquico a tres niveles.	Sí	No	No
Wainakh et al. (2020) [46]	Diseño del esquema jerárquico a tres niveles.	Sí	No	No
Szegedi et al. (2019) [57]	Reducir la transferencia en FL mediante un algoritmo genético federado.	No	No	Sí
Gao et al. (2019) [58]	Creación de un espacio común de features para todos los sujetos.	No	No	Sí
Ju et al. (2020) [59]	Creación de un espacio común de features para todos los sujetos.	No	No	Sí
Zhao et al. (2021) [60]	Detección de somnolencia mediante imágenes.	Clusterizada	Sí	No
Este trabajo	Detección de somnolencia mediante señales EEG y EOG.	Sí	Sí	Sí

## 4 Análisis de objetivos y metodología

La presente sección presenta los objetivos del TFM, así como la metodología seguida a lo largo del trabajo.

### 4.1 Objetivos del trabajo

La premisa con la que parte este TFM es el estudio de FL como técnica para el entrenamiento de modelos predictivos que preserven la privacidad y confidencialidad de los datos de los clientes involucrados en el proceso. En particular, se ha focalizado el estudio en la medición del rendimiento de una topología jerárquica para el escenario de la detección de la somnolencia al volante mediante BCIs, motivando la colaboración entre empresas debido a la la nueva normativa europea UE 2019/2144 [9]. Para poder llevar a cabo dicho objetivo principal, se ha desglosado este mismo en una serie de subobjetivos más específicos:

- Realizar una revisión bibliográfica acerca de la neurociencia cognitiva y las BCIs. Concretamente sobre la evaluación del estado cognitivo. Además, se investigará FL tanto de manera general como aplicado a la detección de la somnolencia al volante, identificando las limitaciones de los trabajos existentes y posibles mejoras que se deben investigar.
- Definir el escenario y los casos de uso a implementar para completar satisfactoriamente el estudio sobre FL.
- Seleccionar las características de interés, el framework de FL y el algoritmo de aprendizaje para realizar el estudio. La decisión se tomará en base a la revisión bibliográfica y los casos de uso definidos previamente.
- Adquirir el conjunto de datos con el que se realizará el estudio. Se debe decidir si bien generar un dataset propio o emplear uno de acceso público. En caso de emplear un público, se estudiarán las opciones disponibles que se adapten a los requerimientos de este TFM y se seleccionará el más adecuado.
- Limpiar y procesar los datos del *dataset* seleccionado para posteriormente analizarlos y distribuirlos conforme al escenario y los diferentes casos de uso.

- Establecer mecanismos de reproducibilidad, una arquitectura común del modelo predictivo y métricas de rendimiento que permitan realizar una comparativa justa entre los diferentes modelos predictivos de los experimentos realizados.
- Configurar e implementar los experimentos correspondientes a cada caso de uso.
- Estudiar y analizar los resultados obtenidos para sacar conclusiones acerca de la viabilidad y el rendimiento de FL.

## 4.2 Metodología

En este apartado se detalla la metodología seguida para cumplir los objetivos definidos previamente. La metodología ha sido marcada basándose en los objetivos propuestos y el orden cronológico de las tareas desarrolladas.

En primer lugar, se realizó una revisión bibliográfica acerca de la neurociencia cognitiva y, concretamente, sobre la evaluación del estado cognitivo. A continuación, se estudiaron las BCIs, las distintas fases que forman su ciclo de funcionamiento y el por qué son de interés en la evaluación del estado cognitivo. En particular, se identificaron los aspectos claves que determinan si un sujeto está en estado de vigilia o somnolencia, junto con las diferentes técnicas para su detección. Además, se investigó a fondo sobre FL y sus topologías jerárquicas. Se indagó en el proceso completo de su funcionamiento, qué mecanismos permiten preservar la privacidad de los datos y qué problemáticas surgen con esta nueva metodología. Se identificaron también escenarios donde se ha empleado y qué resultados se obtuvieron.

Tras adquirir todos estos conocimientos previos, se llevó a cabo un estudio acerca de FL en la actualidad, tanto de manera general como aplicado al escenario de la somnolencia. En este estudio se identificó la inexistencia de un sistema de colaboración federado entre empresas que permita preservar la privacidad de los datos, tanto dentro de cada empresa como entre ellas. Adicionalmente, se detectó una falta de trabajos donde se mida el rendimiento de topologías jerárquicas puras. Todo esto junto con un escaso repertorio de trabajos relacionados con la clasificación de señales cerebrales o con la detección de la somnolencia al volante, siendo nula la existencia de artículos que traten la detección de la somnolencia al volante mediante la clasificación de señales cerebrales.

Con las anteriores limitaciones identificadas, se diseñó el escenario donde se estudiará el sistema de FL anteriormente mencionado mediante un esquema jerárquico, para el que se definen unos casos de uso que irán conformando la solución deseada. Para cada uno de ellos se estableció una definición formal y sus puntos clave a estudiar. Finalmente, se definieron tres casos de uso, siendo cada uno de ellos incremental respecto al anterior, tanto en complejidad como en la utilización de FL.

Una vez definidos los casos de uso, se realizó una selección de las herramientas y el *dataset* a usar. Se compararon diferentes opciones disponibles y se seleccionó aquella

---

que mejor se adaptaba a los requerimientos del estudio. Respecto al conjunto de datos, se estudió la viabilidad de generar uno propio, pero tanto por cuestiones sanitarias como de recursos disponibles finalmente se optó por emplear uno de dominio público.

Seguidamente, se aplicaron técnicas de preprocesamiento y extracción de las características de interés de las señales fisiológicas proporcionadas en el conjunto de datos seleccionado. Los datos refinados de los conductores fueron posteriormente analizados y divididos en relación con el escenario diseñado y las especificaciones técnicas de cada caso de uso.

Con los diferentes casos de uso planteados y los datos distribuidos, se definieron las métricas que se emplearon para medir el rendimiento en cada situación. Para que la comparativa fuese lo más justa posible, se empleó un mecanismo de semillas generadoras de números pseudoaleatorios que garantiza la reproducibilidad de los experimentos junto con el uso de una configuración idéntica de la arquitectura del modelo predictivo para todos los casos de uso. Acto seguido, se detallaron los procedimientos seguidos para la configuración de los experimentos y finalmente se implementaron.

Tras la experimentación, los datos quedan capturados, almacenados y disponibles para su visualización. De este modo, para cada caso de uso se analizaron los puntos clave definidos y su correspondencia con los resultados obtenidos. Aquí, lo que principalmente se pretende determinar en cada caso de uso es si es factible emplear FL y cómo se ve afectado el rendimiento de los modelos predictivos. Por último, se realizó una comparativa entre los resultados obtenidos en este trabajo y los recopilados en el estado del arte, permitiendo identificar contribuciones del trabajo sobre la literatura existente.

---

## 5 Diseño y resolución del trabajo realizado

Esta sección detalla todo el proceso de diseño del mecanismo federado jerárquico propuesto para la colaboración entre empresas. Concretamente, se especifica el diseño del escenario y los casos de uso que darán lugar a la solución. A continuación, se presenta la selección del conjunto de datos y las herramientas para implementar la solución. Seguido a esto, se detallan las fases de preprocesado, extracción de características y división del *dataset*. Después, se establecen los mecanismos de reproducibilidad y equidad en las comparaciones mediante una arquitectura común del modelo predictivo. Por último, se diseña la experimentación a realizar para validar la solución.

### 5.1 Escenario y casos de uso

Para realizar el estudio acerca de las bondades que puede ofrecer FL y sus topologías jerárquicas, se ha diseñado un escenario compuesto por tres empresas dedicadas a la implantación de Sistemas Avanzados de Asistencia a la Conducción (ADAS). Estas empresas deciden indagar sobre las ventajas que les puede ofrecer el uso de FL y sus topologías jerárquicas respecto a la nueva normativa UE 2019/2144 [9], que obliga a incluir un sistema de advertencia de somnolencia y pérdida de atención del conductor en todos los nuevos coches homologados a partir de mayo de 2022, para el que se indica que también es necesario cumplir con la vigente GDPR.

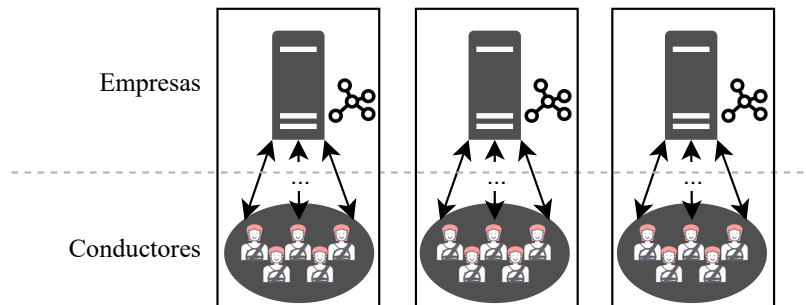
En este escenario, se definen tres casos de uso, siendo los dos primeros estudios previos e incrementales que ayudarán a validar la viabilidad del tercero, que es el más completo y se basa en los dos anteriores. Los datos empleados para la estimación de la somnolencia serán extraídos mediante un simulador de conducción realista y una BCI, concretamente se usarán datos EEG y EOG. El identificador de cada caso de uso viene definido por la expresión *UCX*, siendo *X* el número del caso de uso.

#### 5.1.1 UC1: Federated learning intraempresa

Es el más sencillo de los tres casos de uso, en el que se estudia de manera independiente en cada una de las tres empresas el entrenamiento de un modelo federado con los datos de sus conductores. Los objetivos de este caso de uso son:

- Comprobar la viabilidad de generar un modelo único para cada empresa, que prediga correctamente la somnolencia para todos sus conductores y preserve la privacidad de sus datos.
- Estudiar si la pérdida de rendimiento del modelo federado respecto a entrenar un modelo personalizado para cada cliente es razonable.
- Estudiar el rendimiento del modelo federado respecto a entrenar un único modelo para cada empresa siguiendo la metodología de ML tradicional, es decir, sin mantener la privacidad de los datos.

Para el escenario diseñado en este trabajo, la Figura 5.1 muestra para el UC1 a qué nivel se realizará el entrenamiento federado de los modelos, representado con rectángulos negros. Como se puede apreciar, cada empresa realizará un entrenamiento federado independiente del resto de empresas con los datos de sus conductores.



**Figura 5.1:** Escenario para el UC1 para el que se hace entrenamiento federado independientemente para cada empresa.

### 5.1.2 UC2: Federated learning interempresa

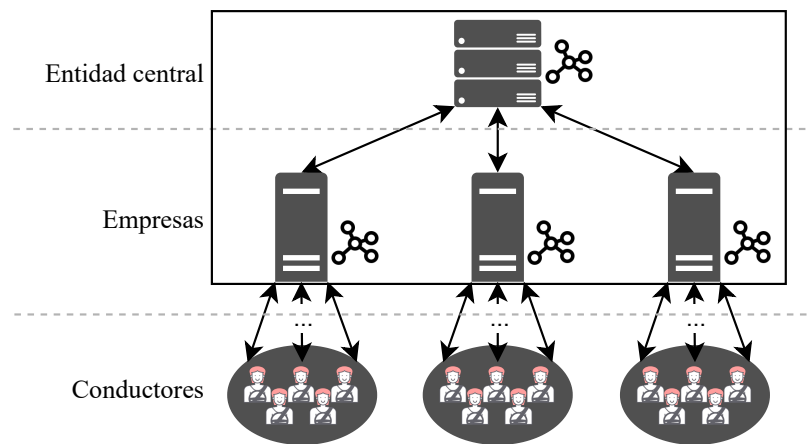
Contrario al UC1, ahora se propone una colaboración entre las tres empresas del escenario planteado, donde se quiere lograr la confidencialidad de los datos entre ellas pero no dentro de cada una de ellas. El objetivo de la colaboración es unir fuerzas y construir un único modelo federado a partir de los datos de las tres organizaciones. En este caso, se pretende determinar:

- La viabilidad de generar un único modelo predictivo que prediga correctamente la somnolencia para los conductores de las tres empresas a la vez que preserve la confidencialidad de los datos de cada organización.
- Si hay un beneficio en las predicciones en el proceso de colaboración, tanto para conductores conocidos (empleados para entrenar el modelo) como nuevos (se dejan fuera de la fase de entrenamiento y posteriormente se evalúa el modelo

con ellos). Es decir, si el rendimiento del modelo colaborativo es superior al que tendría cada empresa generándolo únicamente con sus datos.

- Cómo afecta al rendimiento el uso de FL respecto a construir el modelo centralizado siguiendo una metodología tradicional de ML.

La Figura 5.2 muestra para el UC2 dónde se tiene la federación mediante un rectángulo. Respecto al UC1, se destaca la aparición de la *entidad central*, que permite colaborar a las empresas. Gracias al esquema se puede apreciar cómo las empresas logran la confidencialidad de los datos respecto al resto de empresas pero no dentro de cada una.



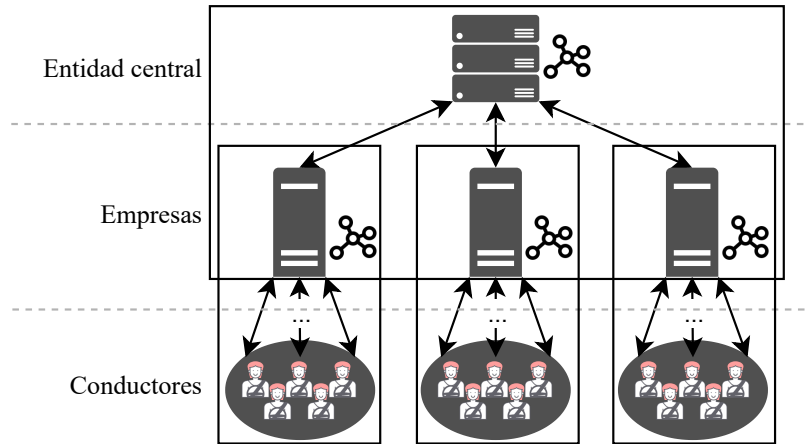
**Figura 5.2:** Escenario para el UC2 para el que se hace entrenamiento federado entre las empresas, pero no dentro de cada una de ellas.

### 5.1.3 UC3: Federated learning jerárquico

Es el más completo de los tres, cuya implementación se basa en los dos UCs anteriores. Consiste en la implantación de un sistema de colaboración entre empresas, al igual que el UC2, que permite el entrenamiento colaborativo de un modelo predictivo a la vez que se mantiene la privacidad de los datos tanto dentro de cada empresa (UC1) como entre empresas (UC2). En consecuencia, se medirá:

- Si es posible entrenar un modelo capaz de mantener la privacidad y confidencialidad de los datos tanto intraempresa como interempresa.
- Qué tal funciona el modelo para predecir la somnolencia tanto en conductores conocidos como nuevos.
- Si existe una pérdida de rendimiento respecto al UC2 al añadir también el mantenimiento de la privacidad de los datos a nivel intraempresa.

La Figura 5.3 muestra cómo se logra preservar la privacidad de los datos tanto entre empresas como dentro de cada una de ellas mediante FL jerárquico. Es importante observar que el proceso de federación se realiza a dos niveles para lograr el objetivo de seguridad.



**Figura 5.3:** Escenario para el UC3 para el que se hace entrenamiento federado tanto entre las empresas como dentro de cada una mediante el sistema jerárquico diseñado.

## 5.2 Conjunto de datos y herramientas empleadas

En esta sección se justifica el uso tanto de las herramientas tecnológicas como del conjunto de datos empleados. Como se comentó previamente en la Sección 4.2, se ha optado por buscar un *dataset* público, ya que por cuestiones sanitarias y de recursos disponibles no era viable el generar uno propio.

### 5.2.1 Conjunto de datos

El punto de partida consiste en la adquisición de un conjunto de datos basado en la detección de la somnolencia en escenarios de conducción. Se realizó una búsqueda de datasets que cumpliesen con los requerimientos exigidos para este trabajo. Los dos mejores candidatos fueron SEED-VIG [3], generado por la universidad Jiao Tong de Shanghái, y C2-CAR-SIMULATOR [61], generado para el proyecto SimuSafe del Horizonte 2020 [62]. Finalmente, se seleccionó el *dataset* SEED-VIG debido a sus condiciones realistas, su entorno multimodal (datos EEG y EOG), la adecuación al objetivo del estudio, la cantidad y calidad de los datos aportados y la familiaridad con el mismo, ya que fue objeto de estudio en mi Trabajo Fin de Grado (TFG).

El dataset consiste en 23 experimentos realizados sobre 21 sujetos diferentes (dos sujetos repitieron el experimento). Para generarlo, se desplegó un simulador de conducción



que consistía en una carretera de cuatro carriles en línea recta, durante prácticamente la totalidad del experimento. Las mediciones se realizaban generalmente después de la hora de comer. Gracias al horario y la monotonía de la prueba, se lograba inducir con mayor facilidad la somnolencia en los sujetos. Cada experimento consta de unas dos horas de señales EEG y EOG registradas mientras los sujetos utilizaban el simulador. Las señales fisiológicas se adquirieron a partir de una BCI de 17 canales según el sistema 10-20. En particular, se empleó un dispositivo *Neuroscan* [63]. El conjunto de datos se etiquetó cada ocho segundos con los valores PERCLOS de los sujetos obtenidos por un dispositivo de *eye-tracking* de *SensoMotoric Instruments* [64].

Para cada experimento, se proveen todos los datos en bruto de las dos horas de duración junto con una variedad de datos ya procesados. En particular, durante este estudio se utilizan los siguientes datos: *i*) media móvil PSD relativa a las cinco bandas de frecuencia de las señales cerebrales, *ii*) datos brutos del canal vertical de EOG y *iii*) etiquetas PERCLOS.

### 5.2.2 Lenguajes de programación y frameworks empleados

Uno de los componentes fundamentales para la realización de este proyecto es el uso de un entorno de trabajo (framework) que ponga a disposición del científico de datos la estructura base y funcionalidad necesaria para implementar los casos de uso diseñados.

En el escenario de la ciencia y analítica de datos, Python suele ser con diferencia el lenguaje de programación por excelencia. Sobre él se construyen los frameworks de ML/DL más populares, como TensorFlow, Pytorch y SKLearn. Así mismo, sobre estos frameworks nacen los dedicados a FL. Debido a ello, se emplea Python para este proyecto.

De las opciones consideradas en la Tabla 2.2, se ha seleccionado Flower, ya que, aunque no sería la mejor opción para implementar en una solución final, es el framework que mejor se adapta para realizar las diferentes pruebas de concepto en cada caso de uso. Tiene una curva de aprendizaje buena, escalabilidad suficiente para 21 sujetos, una excelente documentación e integración con los frameworks de ML/DL. Gracias a la adaptabilidad de Flower, se ha optado por usar la API de alto nivel de Keras que por debajo correrá TensorFlow.

## 5.3 Minería de los datos

Con las herramientas de trabajo concretadas, se realizará un preprocesamiento de los datos seguido de una extracción de las características de interés seleccionadas en base a su efectividad en la literatura (ver Sección 2.2.1). A continuación, estos datos serán analizados para determinar si presentan un comportamiento IID o non-IID y, por último, se lleva a cabo su distribución siguiendo el diseño del escenario.

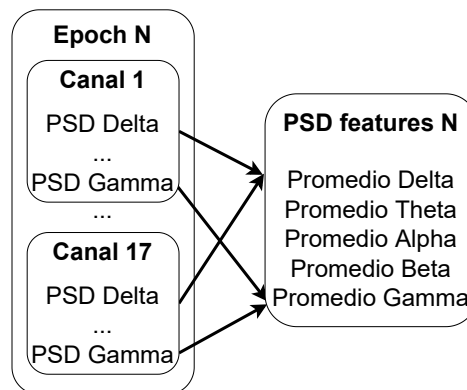
---

### 5.3.1 Preprocesado y extracción de características

Con el *dataset* seleccionado y previamente a distribuir los conductores entre las tres empresas del escenario planteado, se deben preprocesar los datos y generar los vectores de características y las etiquetas acerca del estado cognitivo de los conductores que serán empleadas para entrenar los modelos predictivos de los casos de uso.

Las librerías empleadas durante esta fase han sido: *NumPy* como librería matemática para poder trabajar con vectores y realizar operaciones genéricas de todo tipo, junto con *Pandas*, que se complementan especialmente bien, ya que permite trabajar con cualquier tipo de datos de forma sencilla y compatible con *Numpy* con estructuras llamadas *dataframes*, que posteriormente pueden exportarse como ficheros *CSV*. Adicionalmente, ha sido preciso utilizar librerías específicas para trabajar con señales fisiológicas, en este sentido, *MNE* ha sido ampliamente usada tanto para la carga de los datos del *dataset*, almacenados en formato *MATLAB* (*.mat*) como para su posterior procesamiento. Finalmente, *NeuroKit* también ha sido requerida, ya que ofrece una muy buena funcionalidad para el tratamiento específico de señales oculares.

A partir de los datos disponibles, se ha decidido extraer para cada intervalo de ocho segundos del que se dispone un valor PERCLOS los datos PSD para extraer la cantidad de energía presente en cada banda de frecuencia (ver Tabla 2.1). En este caso no será necesario calcular manualmente el PSD, ya que viene facilitado por el propio *dataset*. En particular, se proporciona la media móvil de las cinco bandas de frecuencia para cada uno de los 17 canales EEG. Para calcular las cinco *features* descritas, se promedian los valores de los 17 electrodos del dataset tal y como se describe en la Figura 5.4. Finalmente, se aplicó una detección de *outliers* para eliminar aquellas *epochs* donde los sensores fallaron y se registraron valores anómalos.

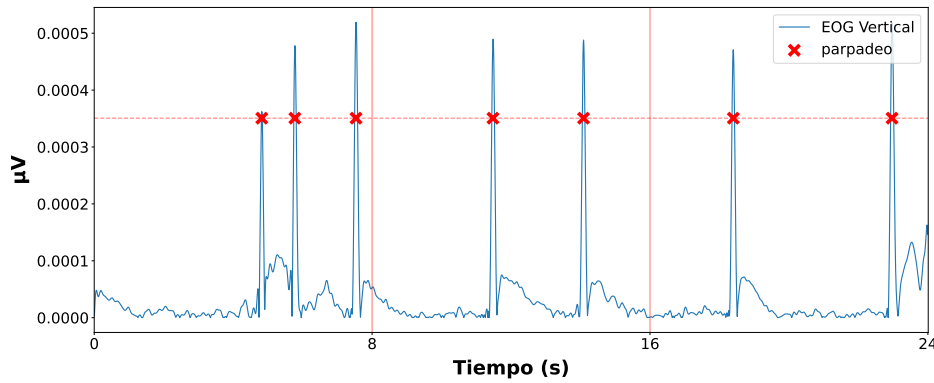


**Figura 5.4:** Procedimiento para el cálculo de las *features* de las cinco bandas del PSD.

De forma complementaria a las cinco *features* del PSD, se desea extraer dos características de la señal EOG para cada *epoch*: *i*) el número de parpadeos que realizan los sujetos y *ii*) la varianza de la señal. Con este objetivo en mente, se comienza con el procesamiento de la señal tras cargar los datos en una estructura de la librería *MNE*.

Se hace uso de la función `eog_clean` de *NeuroKit* para eliminar las frecuencias más altas de la señal y una posterior corrección de valores erróneos debido a fallos en el sensor, permitiendo así una correcta extracción de características. Para la detección de los parpadeos se transforma la señal a su valor absoluto y se emplea `peak_finder` de *MNE*, con un *thresh* del percentil 95 de la señal, es decir, cualquier pico cuyo valor sobrepase ese percentil será considerado un parpadeo. Por otra parte, el cálculo de la varianza emplea únicamente la señal suavizada. Es importante destacar que el cálculo de las *features* se realiza en el canal vertical de la señal EOG, ya que es el de mayor interés en la estimación del estado cognitivo de los conductores.

En la Figura 5.5 se muestran los parpadeos detectados para el conductor con ID 4 durante los 24 primeros segundos de su conducción. La detección es realizada para el valor absoluto del canal vertical de la señal EOG. Las líneas verticales delimitan las distintas épocas, mientras que la horizontal discontinua indica el parámetro **thresh** establecido.



**Figura 5.5:** Resultado para la detección de los parpadeos en el canal vertical del EOG.

Finalmente, los datos PERCLOS deben ser discretizados a las etiquetas *despierto* y *somnoliento*, ya que estos se proporcionan en el intervalo de cero a uno, donde el valor uno se corresponde con que el conductor permanece con los ojos cerrados o parcialmente cerrados durante la totalidad del intervalo de ocho segundos.

Para realizar la discretización se debe definir un umbral numérico (*threshold*) a partir del cual las muestras con un valor PERCLOS igual o superior serán identificadas con estado *somnoliento*. Para determinar dicho valor se emplea la Fórmula 5.1, que tiene una gran ventaja respecto a emplear el valor estático 0.25 como se propone en cierta parte de la literatura [65, 66, 67]. El método dinámico tiene en cuenta las particularidades fisiológicas de cada sujeto y, por tanto, se obtiene una división personalizada en la detección de somnolencia que mejora el etiquetado de los datos, tal como se propone en [68].

$$threshold = \min(perclos) + (\max(perclos) - \min(perclos)) * 0.25 \quad (5.1)$$

Tras esta etapa de preprocesamiento y extracción de características, se obtiene para

cada sujeto un conjunto de  $N$  observaciones, cada una respectiva a una *epoch* de ocho segundos, referentes a su estado cognitivo. Cada una de estas observaciones está conformada por siete *features*, cinco del PSD y dos del EOG, y de su etiqueta correspondiente al estado cognitivo.

Respecto al uso de los datos en el entrenamiento y evaluación de los modelos predictivos, se debe remarcar que se estandarizan las características de cada sujeto, restándoles la media y dividiéndolas por la desviación estándar. Se ha optado por la estandarización frente a la normalización *min-max*, ya que en esta última los valores anormales registrados por algún sensor pueden llegar a dominar el proceso de normalizado. Por último, el particionado de los datos entre los conjuntos de *train* y *test*, se realizará siempre con una proporción del 70% para *train* y el 30% restante para *test*.

### 5.3.2 Análisis del comportamiento de las características

Es conveniente determinar de antemano si los datos del problema a resolver se comportan de forma IID o non-IID. Para el escenario IID las características extraídas del PSD y EOG para un determinado estado cognitivo presentarían tendencias similares entre sujetos. Para el caso contrario, en el que los valores difiriesen entre sujetos, tendríamos un escenario non-IID.

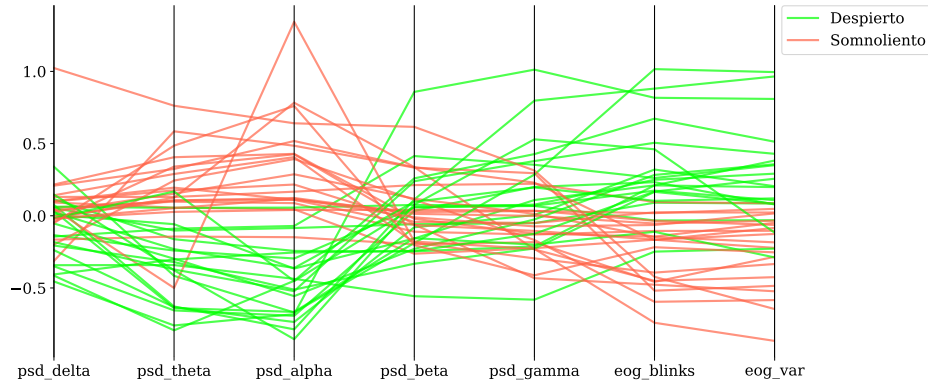
Como se indica en más detalle en la sección 2.3, un escenario non-IID supone que las diferencias particulares entre conductores pueden generar problemas en la convergencia del modelo predictivo que se entrene de manera federada, ya que la agregación de los pesos del modelo federado se verá afectada negativamente si las actualizaciones locales agregadas son muy diferentes entre sí.

Como resultado del análisis de las *features* (ya estandarizadas) se presenta un diagrama de coordenadas paralelas (Figura 5.6). En este tipo de gráfico, cada *feature* recibe su propio eje y todos los ejes se colocan en paralelo entre sí. Los valores se representan como líneas conectadas a través de cada eje. En particular, se muestra dos líneas para cada sujeto, una verde y otra roja, que corresponden al valor promedio de cada predictor para las observaciones particulares de ese sujeto marcadas como despierto y somnoliento, respectivamente. Se puede observar que tanto para el estado *despierto* como *somnoliento*, los valores promedios difieren entre los sujetos, por lo que se puede concluir que se tiene un escenario non-IID.

### 5.3.3 División del dataset

Dado que el caso de uso consta de tres empresas, se procede a la distribución de los 21 sujetos del *dataset* en tres grupos de siete, uno para cada empresa. Para hacer la división más interesante, se ha analizado la distribución de clases de los 21 conductores y se les ha clasificado en tres grupos diferentes en base a esta. El primer grupo consta de los conductores cuya proporción de muestras con estados despierto y somnoliento es balanceada, mientras que los grupos dos y tres presentan una proporción de al menos el

---

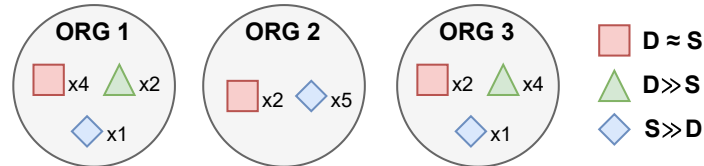


**Figura 5.6:** Análisis de las *features* para determinar el escenario IID o non-IID.

66% de las observaciones etiquetadas como *despierto* o *somnoliento*, respectivamente. Los resultados de la agrupación en estos tres grupos son los siguientes:

- Grupo 1 (despierto  $\approx$  somnoliento): sujetos 3, 4, 5, 12, 13, 14, 15 y 18.
- Grupo 2 (despierto  $\gg$  somnoliento): sujetos 2, 8, 9, 17, 20 y 21.
- Grupo 3 (somnoliento  $\gg$  despierto): sujetos 1, 6, 7, 10, 11, 16 y 19.

Ahora, se distribuyen los sujetos entre las tres empresas que conforman el caso de uso. Se hace de forma equitativa respecto a la cantidad de sujetos, pero heterogénea respecto a la distribución de sus clases, de esta forma se tiene el escenario non-IID mencionado anteriormente y con el agregado adicional de sesgos en los datos de las diferentes empresas debido a la distribución heterogénea de clientes de diferentes grupos. Como se puede apreciar en la Figura 5.7, la *Organización 1* tendrá sujetos de los tres grupos, siendo el grupo 1 el de mayor proporción, mientras que la *Organización 2* tendrá una gran proporción de sujetos del grupo 3, ninguno del grupo 2 y alguno del grupo 1. Por último, la *Organización 3* consta de una gran cantidad de sujetos del grupo 2 pero muy pocos del 1 y del 3.



**Figura 5.7:** División de los sujetos para el escenario de FL. Las etiquetas D y S hacen referencia a la cantidad de muestras con estado *despierto* y *somnoliento* respectivamente.

## 5.4 Equidad en la experimentación

Para que los resultados y las conclusiones del trabajo sean lo más confiables posibles, es necesario establecer mecanismos de reproducibilidad de los experimentos. Además, también se debe establecer una configuración común para los modelos predictivos que se entrenen.

Respecto a la reproducibilidad, se ha establecido semillas generadoras de números pseudoaleatorios para garantizar unos resultados justos y reproducibles entre los diferentes casos de uso. De esta forma, las particiones de datos de train y test estarán compuestas siempre por los mismos datos. Además, el entrenamiento de los modelos predictivos también generará siempre los mismos resultados.

### 5.4.1 Arquitectura común para el modelo predictivo

A lo largo de los casos de uso diseñados siempre se persigue el mismo objetivo: la correcta estimación del estado cognitivo de los conductores. Esta tarea se efectúa a través de la predicción de la etiqueta PERCLOS en base a las características extraídas de las señales EEG y EOG del conductor. Esta tarea de clasificar los elementos de un conjunto en dos grupos se define como una tarea de clasificación binaria, ya que se tienen dos etiquetas (grupos) a predecir: *despierto* y *somnoliento*.

FL funciona particularmente bien en combinación con DL, es decir, con redes neuronales. En base al escenario del que se dispone, donde para cada predicción a realizar se reciben las siete características extraídas en la sección 5.3.1, se identifica al MLP como el tipo de modelo predictivo a usar. Además, el framework *Flower* no aporta facilidades para entrenar otros algoritmos de ML federados que no sean redes neuronales. Ahora, para lograr una comparativa lo más justa posible, se definirá una configuración de los hiperparámetros del MLP que se mantendrá fija y que será suficientemente general para ofrecer un buen rendimiento en todos los escenarios que se plantean.

En este sentido se realiza un proceso de búsqueda de los mejores hiperparámetros del modelo para que las predicciones sean lo mejor posible en cualquier situación. Debido a que las redes neuronales son unos modelos altamente configurables, se ha optado por emplear una optimización bayesiana de hiperparámetros junto con una validación cruzada de cinco pliegues para validar los resultados obtenidos. Con esta estrategia, se consigue que la búsqueda se vaya redirigiendo en cada iteración hacia las regiones de mayor interés, reduciendo así el número de combinaciones de hiperparámetros con las que se evalúa el modelo, eligiendo únicamente los mejores candidatos [69].

Se configura el proceso de optimización de tal forma que se busque la mejor combinación de: neuronas, número de capas ocultas, uso de *batch normalization*, uso de *dropout* y su *rate*, la función de activación entre capas ocultas y el optimizador a usar junto con su *learning rate*.

La optimización bayesiana exploró 29 combinaciones de parámetros usando los datos de los 21 sujetos del dataset. Cinco de estas combinaciones obtuvieron el mejor accuracy

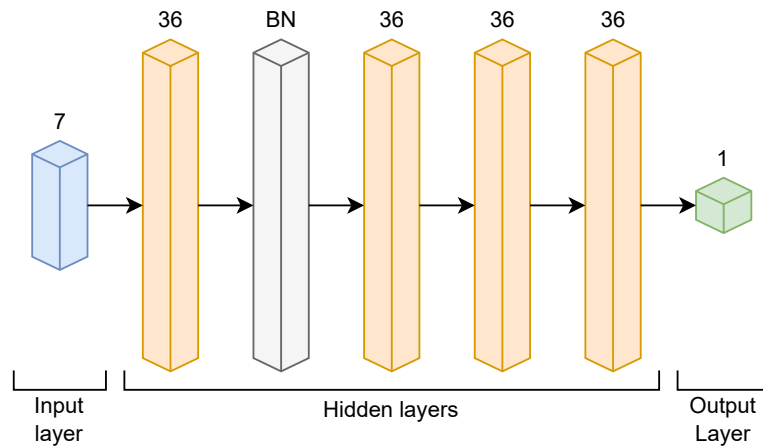
---

y, de estas combinaciones, finalmente se seleccionó la que menor número de parámetros configurables tenía. De esta forma se asegura el correcto funcionamiento tanto para los modelos más sencillos en los que se emplee una menor cantidad de datos para el entrenamiento, como para los modelo federados de los UC2 y UC3, donde se emplean los datos de 20 de los 21 sujetos.

La arquitectura final del MLP consta de 4 393 parámetros entrenables. En la Figura 5.8 se muestra la arquitectura de la red neuronal, donde los números encima de cada capa de la red indican el número de neuronas que tiene cada capa. Por otra parte, el valor *BN* sobre la segunda capa oculta indica que es una capa de *batch normalization*. La lista de hiperparámetros optimizados es la siguiente:

- Neuronas por capa oculta = 36
- Capas densas ocultas = 4
- Dropout = No
- Batch normalization = Sí
- Función de activación entre capas = ReLU
- Optimizador = Adadelta
- Learning rate = 0.180165

Los hiperparámetros con valores fijos fueron: *i*) la capa de entrada con siete neuronas, una para cada *feature* de las extraídas en la Sección 5.3.1, *ii*) la capa de salida con una neurona, es el *output* que produce el modelo y al ser una única neurona actuará como indicador binario acerca del estado del conductor (0: despierto, 1: somnoliento), *sigmoid* como función de activación para la capa de salida y *iii*) *binary crossentropy* como función de pérdida para optimizar el modelo.



**Figura 5.8:** Arquitectura del MLP tras optimizar sus hiperparámetros mediante optimización bayesiana y validación cruzada de cinco pliegues.

Las cuatro métricas alcanzan su mejor valor en 1 y el peor en 0. Concretamente para este estudio, la sensibilidad reflejará la probabilidad de que un sujeto *somnoliento* sea

etiquetado como tal, mientras que la especificidad es similar pero para el estado *despierto*. Por su parte, accuracy medirá la proporción de muestras correctamente clasificadas, mientras que f1-score da una mejor medición para los ejemplos mal clasificados.

## 5.5 Diseño de la experimentación

Se ha determinado la experimentación a realizar en base a los objetivos de cada caso de uso. Se debe recordar que siempre se empleará el MLP con hiperparámetros optimizados (ver sección 5.4.1) para generar tanto los modelos federados como el resto de alternativas.

### 5.5.1 Experimentos para el UC1

Para comprobar si es factible construir un único modelo federado por empresa a partir de los datos de sus conductores y, estudiar su rendimiento respecto a si se tuviesen modelos personalizados y si no se usase FL, se deben estudiar los siguientes modelos:

- Modelos individuales: Cada empresa entrenará un modelo personalizado para cada uno de sus conductores empleando únicamente los datos de ese conductor.
- Modelo por empresa con ML: Cada empresa emplea su dataset de conductores para entrenar un modelo general. No se preserva la privacidad de los datos de sus conductores.
- Modelo por empresa con FL: Similar al modelo por empresa pero usando FL, por lo que se preserva la privacidad de los datos de sus conductores. En el proceso federado, en cada empresa se tendrán siete modelos locales, uno para cada conductor.

Los resultados de los modelos individuales se obtendrán promediando el rendimiento para cada sujeto dentro de una misma empresa. De esta forma se podrán comparar con los otros modelos entrenados en este caso de uso.

### 5.5.2 Experimentos para el UC2

Para poder comparar el rendimiento del modelo federado entrenado a partir de los datos de las tres empresas, tanto para conductores conocidos como nuevos, es necesario estudiar las siguientes alternativas:

- Modelo por empresa con FL: Cada empresa emplea su propio dataset de conductores para entrenar su modelo federado privado. Estos resultados están ya disponibles del UC1.



- Modelo general con ML: No preserva la privacidad de los datos. Los datos de las tres empresas se unen en un único dataset con el que se entrena el modelo predictivo.
- Modelo general con FL: Las diferentes empresas colaboran en el entrenamiento de un modelo federado. Se preserva la confidencialidad de los datos entre las empresas. Se tendrán tres modelos locales en el proceso federado, uno por empresa.

### 5.5.3 Experimentos para el UC3

En este último caso de uso se pretende mantener la privacidad tanto a nivel intraempresa como interempresa, por lo que será necesario entrenar este nuevo modelo federado y compararlo con los resultados del UC2. Los modelos estudiados son:

- Modelo general con FL: Las diferentes empresas colaboran en el entrenamiento de un modelo federado. Se preserva la confidencialidad de los datos entre las empresas. Resultados disponibles del UC2.
  - Modelo general con FL jerárquico: Se emplea el sistema de colaboración jerárquico diseñado en este trabajo. Las diferentes empresas colaboran en el entrenamiento de un modelo federado. Se preserva la privacidad y confidencialidad de los datos tanto dentro de cada empresa como entre las empresas. Respecto a los modelos locales, se tendrán tantos como conductores conocidos participen en la federación.
-

## 6 Análisis de resultados

Una vez expuesto el diseño de la solución junto con las técnicas que van a ser empleadas, se procede a la implementación y a la medición de los resultados de las alternativas diseñadas para cada caso de uso y a la posterior discusión de los mismos.

En primer lugar, se seleccionaron las siguientes métricas para evaluar y comparar el rendimiento de los modelos entrenados a lo largo de los casos de uso son las siguientes (VP: Verdaderos Positivos, VN: Verdaderos Negativos, FP: Falsos Positivos, FN: Falsos Negativos):

$$\begin{aligned} \bullet \text{ Sensibilidad} &= \frac{VP}{VP+FN} & \bullet \text{ Accuracy} &= \frac{VP+VN}{VP+VN+FP+FN} \\ \bullet \text{ Especificidad} &= \frac{VN}{VN+FP} & \bullet \text{ F1 - score} &= \frac{VP}{VP+\frac{1}{2}(FP+FN)} \end{aligned}$$

Tras definir las métricas se procede con la experimentación. En concreto para los UC2 y UC3, donde se requiere estudiar el comportamiento de los modelos tanto para usuarios conocidos como nuevos, se han realizado varias pruebas dejando fuera de la fase de entrenamiento a más sujetos cada vez. Concretamente, se ha estudiado el rendimiento desde la versión en la que se entrena con los datos de 20 sujetos y hay un único conductor desconocido hasta la versión en la que se entrena con siete conductores y se tienen 14 desconocidos. A la hora de determinar qué usuarios son seleccionados como conocidos, se define como condición que debe haber al menos un usuario de cada empresa presente. Además, para garantizar unos resultados sólidos y fiables, se ha repetido la experimentación con 30 combinaciones diferentes de conductores para cada situación de  $N$  conductores desconocidos, con  $N$  en el rango 2 a 14. Para un único cliente desconocido se pueden probar las 21 combinaciones disponibles, una para cada conductor del *dataset*. De esta forma, el teorema del límite central nos garantiza una buena aproximación del promedio de los experimentos al valor real de la media si se tiene una muestra de al menos 30 observaciones [70].

Para el entrenamiento de los modelos federados, se ha establecido que no se hará uso de técnicas adicionales para garantizar la seguridad de los datos como *Secure Aggregation* o *Differential Privacy*, ya que el escenario diseñado presupone un entorno de confianza entre las empresas participantes, por lo que no es necesario establecer mecanismos adicionales de protección. Hay que tener en cuenta que el uso de estos mecanismos tiene un coste, tanto en tiempo adicional de ejecución como en un empeoramiento del rendimiento de los modelos predictivos, por lo que si el escenario es fidedigno, es preferible no utilizarlos. Por otra parte, se ha optado por *FedAvg* como algoritmo de

agregación para los pesos de los modelos locales debido al buen funcionamiento que ofrece en la experimentación. Si no hubiese funcionado bien, se habrían probado otros algoritmos de agregación de los descritos en la Sección 2.3.

## 6.1 Resultados para el UC1

La configuración tanto para los modelos individuales como los por empresa empleando ML es similar: *i)* se toman los datos correspondientes según la alternativa, *ii)* se carga el modelo con los hiperparámetros optimizados y *iii)* se ejecuta el entrenamiento por 150 épocas utilizando *early stopping* si no se produce una mejora en el accuracy del conjunto de test en 20 épocas. Por su parte, el entrenamiento de los modelos por empresa federados, que es el objeto principal de estudio para este caso de uso, ha sido configurado de la siguiente manera. Tras cargar los datos y la arquitectura común, similar a la experimentación anterior, se establece la configuración del proceso federado. Para cada empresa, la entidad centralizada, que en este caso será la propia empresa, inicializa los pesos del modelo que los clientes (conductores de la empresa) comenzarán a entrenar. Así, todos los clientes disponen de la misma configuración inicial del modelo. Después, el proceso de entrenamiento se configura con 100 rondas federadas, con una época de entrenamiento de los MLP locales por ronda.

Los resultados obtenidos tras implementar los experimentos se muestran en la Tabla 6.1 donde, para cada uno de los tres enfoques a comparar, se muestran las métricas obtenidas por cada empresa, designadas como  $EX$ , siendo  $X$  el identificador de cada empresa descrita en la Figura 5.7. Se observa que los mejores resultados se tienen en los modelos individuales con unos valores promedios de accuracy y f1-score de 0.8766 y 0.86 respectivamente. Esto es algo esperable, ya que al entrenarse únicamente con los datos de un sujeto, el modelo predictivo es capaz de adaptarse mejor a sus peculiaridades.

Para los modelos por empresa, donde se tiene un único modelo construido a partir de los datos de todos los conductores de esa empresa, la versión federada ofrece un accuracy y f1-score promedios de 0.7733 y 0.78, lo que supone una pérdida de rendimiento respecto a la versión individual de un 0.1033 y 0.08, a cambio de tener un único modelo en cada empresa frente a los siete necesarios en la alternativa individual. Si se comparan estos resultados frente a tener un único modelo por empresa pero usando ML tradicional, los resultados son: accuracy=0.8233; f1-score=0.8066. En consecuencia, la pérdida de rendimiento que se tiene en promedio a cambio de mantener la privacidad de los datos es del 0.05 en accuracy y 0.0266 para f1-score.

También se puede apreciar el sesgo introducido por el escenario non-IID heterogéneo definido en la Sección 5.3.3. La empresa 2, que tenía una mayor cantidad de conductores predominantemente somnolientos, tiene una especificidad especialmente baja comparada con el resto de empresas, es decir, sus modelos no son capaces de distinguir correctamente cuando un conductor está realmente despierto, produciendo un gran número de falsos positivos. El mismo caso, pero esta vez respectivo a la sensibi-

lidad, sucede en la empresa 3, donde predominan conductores despiertos, se tiene un mayor número de falsos negativos a la hora de estimar el estado de somnolencia. Por el contrario, esta situación no se da en la empresa 1, ya que cuenta con un abanico de conductores lo suficientemente diverso para identificar correctamente tanto los estados *despierto* como *somnoliento*.

Finalmente, también es posible realizar una comparación entre los resultados de los modelos individuales del presente TFM respecto a los obtenidos en el mejor caso de mi TFG [71]. Se debe tener en cuenta que no es una comparación 100% justa, ya que en el TFG no se extrajo la varianza de la señal EOG como *feature*, se empleó normalización *min-max* y no se eliminaron las muestras con valores atípicos. El resto de condiciones son similares en ambos trabajos: optimización de los modelos mediante búsqueda de hiperparámetros con validación cruzada, mismos sujetos y mismas métricas. En el TFG se obtuvo con Random Forest como modelo y seis features (PSD más número de parpadeos) un accuracy de 0.846 y un f1-score de 0.756 promedios. Mientras que en el presente trabajo, los valores de accuracy y f1-score son 0.8766 y 0.86, respectivamente. Esto supone una mejora obtenida en este trabajo del 0.0306 en accuracy y 0.104 en f1-score comparado con el TFG desarrollado previamente.

**Tabla 6.1:** Comparativa de resultados del UC1 entre los enfoques individuales, por empresa con ML y por empresa con FL.

Tipo de modelo	Accuracy	Sensibilidad	Especificidad	F1-score
Individuales	E1: 0.87	E1: 0.81	E1: 0.85	E1: 0.84
	E2: 0.87	E2: 0.93	E2: 0.58	E2: 0.91
	E3: 0.89	E3: 0.77	E3: 0.91	E3: 0.83
Empresa con ML	E1: 0.80	E1: 0.77	E1: 0.80	E1: 0.76
	E2: 0.84	E2: 0.94	E2: 0.46	E2: 0.90
	E3: 0.83	E3: 0.72	E3: 0.88	E3: 0.76
Empresa con FL	E1: 0.74	E1: 0.74	E1: 0.75	E1: 0.76
	E2: 0.80	E2: 0.93	E2: 0.39	E2: 0.88
	E3: 0.78	E3: 0.59	E3: 0.92	E3: 0.70

## 6.2 Resultados para el UC2

La configuración del entrenamiento del modelo general con ML es similar al realizado en la sección previa. Se juntan los datos de los sujetos conocidos en un mismo dataset, se carga el modelo con los hiperparámetros optimizados y se realiza un entrenamiento de 150 épocas con *early stopping*. Referente a la configuración de los modelos federados, la única diferencia en el proceso es el número de rondas federadas. Se ha intentado reducir todo lo posible ya que se debe tener en cuenta la gran cantidad de entrenamientos que se van a realizar. Concretamente, 30 experimentos para cada valor  $N$  de conductores

desconocidos ( $N$  de 2 a 14) más los 21 referentes a un conductor desconocido, lo que hace un total de 411 entrenamientos  $\times$  número de rondas. Se han hecho pruebas con diferentes números de rondas y se ha observado la convergencia de los modelos. Finalmente se ha determinado que con cinco rondas federadas por entrenamiento es suficiente.

La Figura 6.1 muestra dos gráficas, ambas compuestas de los mismos ejes. El eje Y, indica la media y el intervalo de confianza del 95% del accuracy obtenido por los modelos estudiados, mientras que el eje X señala el número de conductores que se han usado para entrenar el modelo (conocidos). En particular, la gráfica superior muestra los resultados de evaluar los modelos del UC2 con los conductores conocidos, es decir, para los mismos con los que fue entrenado, mientras que la inferior lo hace para los conductores desconocidos, que se dejaron a parte en la fase de entrenamiento. En ambas gráficas se compara:

- El rendimiento promedio obtenido por los modelos federados por empresa del UC1. Se indica como una línea discontinua negra. Se emplea como *baseline*, es decir, para ver cuánto mejora o empeoran las predicciones de los modelos generales respecto a los modelos por empresa.
- La evolución del rendimiento del modelo general con ML conforme varía el número de conductores desconocidos. Descrito por la línea de color verde.
- La evolución del rendimiento del modelo general con FL conforme varía el número de conductores desconocidos. Descrito por la línea de color azul.

A partir de la figura Figura 6.1a, se puede determinar que aumentar el número de sujetos con los que se entrena el modelo no es beneficioso. El accuracy se reduce lentamente conforme aumentan los conductores conocidos. Esto es algo esperable, ya que el modelo predictivo resultante tenderá a generalizar cada vez más en lugar de poder adaptarse correctamente a las peculiaridades de cada conductor. En contraposición (ver Figura 6.1b), el aumentar el número de conductores con los que se entrena el modelo (conocidos) sí que es beneficioso para estimar la somnolencia en conductores nuevos. La misma generalización que previamente afectaba al rendimiento negativamente, ahora lo hace positivamente, logrando una mejora en las predicciones gracias a la colaboración de las empresas, que aumenta conforme lo hace el número de clientes con los que se entrena el modelo.

Otro apunte interesante es que los modelo generales tienen un rendimiento superior para conductores desconocidos respecto al modelo federado por empresa, hasta un máximo del 3% para ML y un 5% para FL. En cambio, el accuracy promedio es superior en el modelo por empresa con FL para conductores conocidos, ya que tanto los modelos tradicionales como federados sufren pérdidas de rendimiento de hasta el 4% y el 6% respectivamente.

Cabe mencionar que los modelos generales propuestos en este caso de uso no tiene como objetivo ser lo más precisos posibles para un sujeto en particular, sino que están pensados para usarse cuando un nuevo conductor use el vehículo las primeras veces

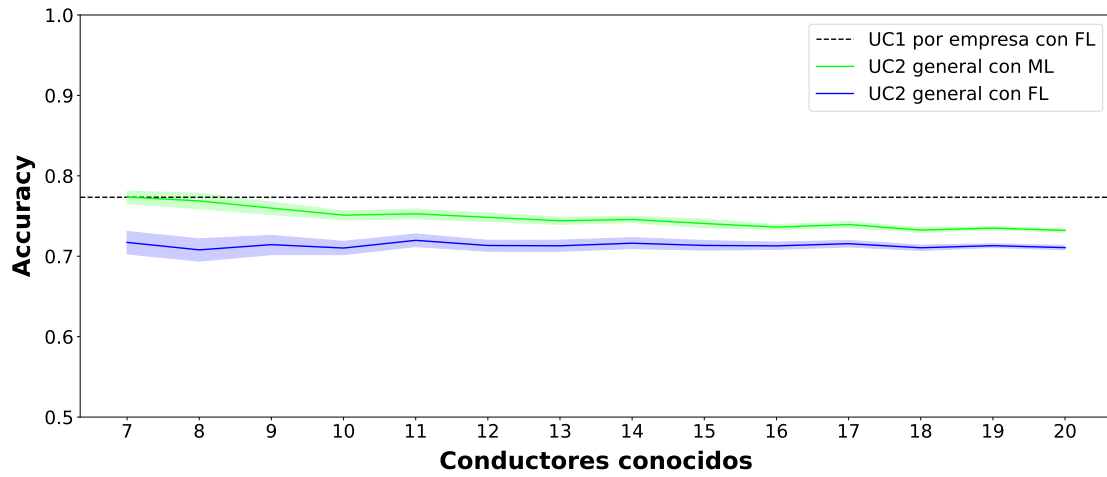
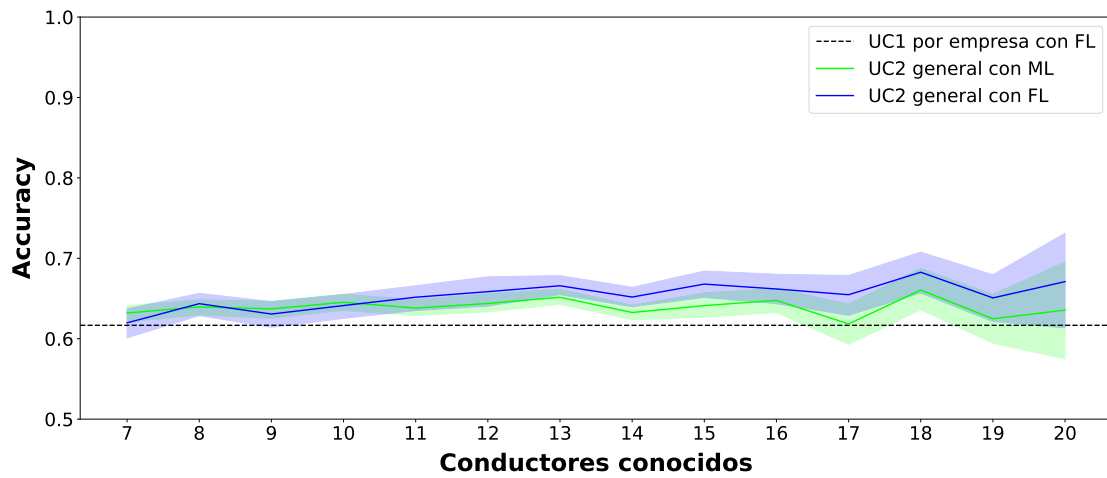
hasta que pueda disponer de un modelo personalizado. En esta situación, se observa que la mejor opción disponible para las empresas sería la de colaborar empleando FL ya que es el que mejor rendimiento ofrece en conductores nuevos.

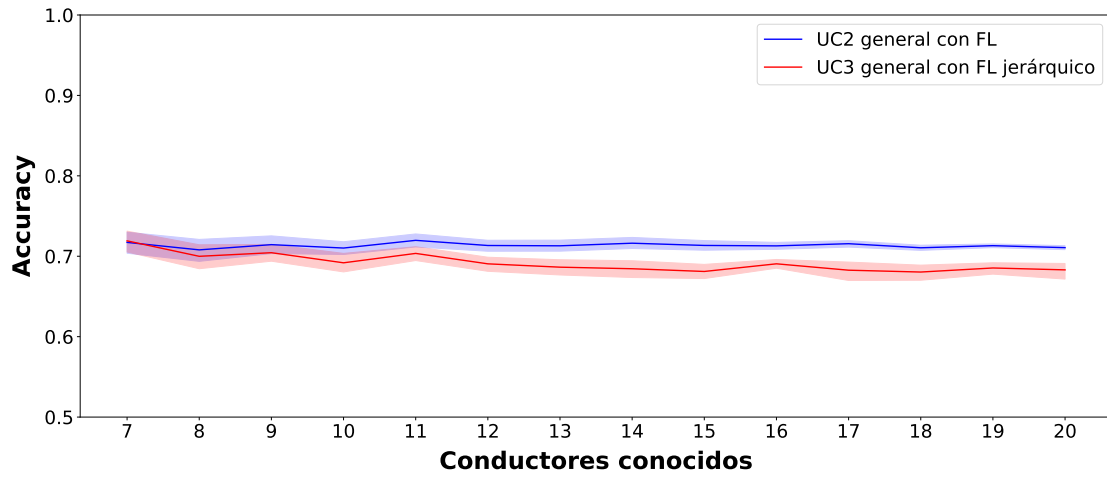
### 6.3 Resultados para el UC3

Este último caso de uso tiene como objetivo probar la propuesta de FL jerárquico diseñada en este TFM, midiendo su rendimiento para conductores conocidos y nuevos frente al modelo general con FL del UC2. Por consiguiente, solo es necesario entrenar los modelos generales federados jerárquicos, para los que se mantiene la privacidad de los datos tanto dentro de la empresa como entre ellas. Al igual que para el modelo federado del UC2, se tienen 411 experimentos a realizar, por lo que es necesario establecer el número de rondas federadas para el entrenamiento lo más bajo posible, pero garantizando la convergencia de los modelos. Para este caso, es suficiente con 15 rondas federadas para que los modelos converjan. El número  $e_1$  de épocas locales de entrenamiento en cada cliente será de  $e_1 = 1$ , lo que implica que la empresa, que actúa como servidor edge en el esquema jerárquico, realizará agregaciones de los gradientes locales de sus conductores en cada ronda del proceso federado. También se ha configurado  $e_2 = 1$  de manera que el servidor central agregue los modelos de las empresas cada ronda. Nótese que el número de rondas es superior al valor empleado por el UC2, debido a que en el UC2 se tienen tres modelos locales, uno por empresa, mientras que en el modelo federado jerárquico del UC3 se tienen  $21-N$  modelos locales, siendo  $N$  el número de conductores desconocidos. Así, la convergencia será más costosa.

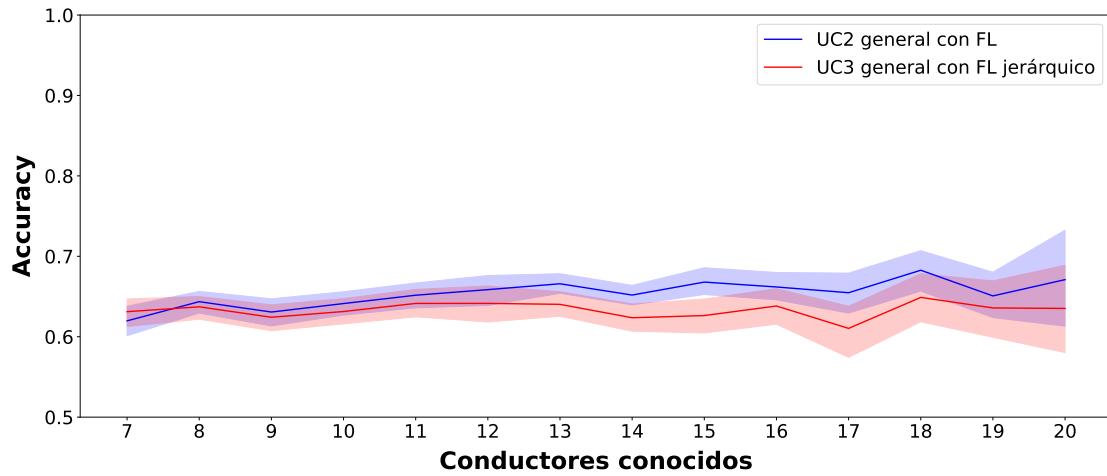
La Figura 6.2 muestra la comparativa entre ambos modelos federados. La línea azul, al igual que en la figura anterior, se corresponde con el modelo general federado del UC2, mientras que la línea roja es para el modelo general federado jerárquico. Los ejes X e Y son similares a la figura anterior (6.1) y de igual forma, la gráfica superior muestra los resultados de evaluar los modelos con los conductores conocidos, mientras que la inferior lo hace para los conductores desconocidos.

A partir de estas gráficas, se puede observar que el modelo jerárquico presenta el mismo comportamiento que el del UC2. Empeora para usuarios conocidos pero crece para usuarios nuevos, y en ambos casos se incrementa el efecto conforme disminuye el número de conductores nuevos. También se observa que el modelo jerárquico presenta una pequeña pérdida de rendimiento respecto al modelo del UC2 de aproximadamente entre un 0%-2%. Pese a esta pérdida, se justifica totalmente su uso para escenarios como el que se ha diseñado, donde la preservación de la privacidad de los datos es un requisito fundamental.

(a) Rendimiento de los modelos para usuarios **conocidos**.(b) Rendimiento de los modelos para usuarios **desconocidos**, es decir, cuyos datos no se usaron durante el entrenamiento.**Figura 6.1:** Comparativa de rendimiento del UC2 entre los modelos por empresa con FL, general con ML y general con FL.



(a) Rendimiento de los modelos para usuarios **conocidos**.



(b) Rendimiento de los modelos para usuarios **desconocidos**, es decir, cuyos datos no se usaron durante el entrenamiento.

**Figura 6.2:** Comparativa del rendimiento entre los modelos general con FL del UC2 y el general con FL jerárquico del UC3.



## 6.4 Discusión de los resultados

Esta sección está dedicada a comparar qué tan buenos son los resultados de cada caso de uso de forma que se justifique si es de interés tanto la colaboración entre empresas como el uso de FL y el sistema jerárquico diseñado.

La Tabla 6.2 de manera sintetizada lo expuesto a lo largo de este apartado. Cada fila muestra los resultados promedios de accuracy para la experimentación realizada en cada caso de uso, junto con las respectivas ganancias o pérdidas de rendimiento que suponen los modelos propuestos. En particular, el UC1 muestra que la pérdida de rendimiento a cambio de emplear un único modelo por empresa para estimar la somnolencia de todos sus conductores es en torno al 6% para ML y del 10% para FL.

Después, en el UC2, se observa que los modelos generales, que usan datos de los conductores de las tres empresas, presentan un peor rendimiento conforme aumenta el número de conductores involucrados (hasta un 4% para ML y un 6% para FL). Aunque también es muy importante destacar existe una mejora de las estimaciones de la somnolencia cuando esos modelos generales se evalúan con conductores nuevos para el modelo, en este caso, las ganancias en cuanto a rendimiento son hasta del 3% para ML y del 5% para FL, consolidando al modelo general federado como la mejor opción para conductores nuevos.

Finalmente, es importante destacar los resultados obtenidos en el UC3, el sistema jerárquico propuesto presenta unas pérdidas de rendimiento alrededor del 0%-2% tanto para conductores conocidos como nuevos respecto al modelo federado del UC2 a cambio de una mejora de la seguridad de los datos durante el proceso de entrenamiento del modelo.

Por lo tanto, a partir de esta discusión de los resultados, se concluye que tanto FL como el mecanismo jerárquico para la colaboración entre empresas diseñado ofrecen unos resultados positivos para este escenario en particular. Esto es debido a que se consigue un rendimiento prácticamente similar o superior a las versiones con ML. Por último, cabe mencionar que el buen rendimiento del modelo jerárquico requiere de más evaluación en otros escenarios.

**Tabla 6.2:** Sumario del rendimiento en accuracy de los diferentes experimentos realizados a lo largo de los tres casos de uso. Los indicadores *C* y *N* hacen referencia al rendimiento en conductores conocidos y nuevos, respectivamente.

Caso de uso	Rendimiento de los modelos en la experimentación		
UC1	<b>Personalizados</b> 87.7%	<b>Por empresa ML</b> 82.3% ( $\nabla \sim 6\%$ )	<b>Por empresa FL</b> 77.3% ( $\nabla \sim 10\%$ )
UC2	<b>Por empresa FL</b> C: 77.3% N: 61.8%	<b>General ML</b> C: 77.4%-73.2% ( $\nabla \sim 0\% - 4\%$ ) N: 63.2%-63.6% ( $\Delta \sim 2\% - 3\%$ )	<b>General FL</b> C: 71.7%-71.1% ( $\nabla \sim 6\%$ ) N: 62.0%-67.1% ( $\Delta \sim 1\% - 5\%$ )
UC3	<b>General FL</b> C: 71.7%-71.1% N: 62.0%-67.1%	<b>General FL jerárquico</b> C: <b>71.9%-69.3%</b> ( $\nabla \sim 0\% - 2\%$ ) N: <b>63.1%-65.5%</b> ( $\nabla \sim 1\% - 2\%$ )	

## 7 Conclusiones y vías futuras

El aprendizaje federado es un enfoque de ML que permite entrenar un modelo de forma colaborativa y descentralizada, logrando una mayor privacidad y seguridad de los datos de cada cliente, ya que estos nunca abandonan su dispositivo. Esta metodología fomenta que varias empresas deseen entrenar modelos predictivos de manera federada en colaboración con otras empresas, buscando mantener la privacidad de sus datos tanto dentro de su propia empresa como con el resto de las participantes en el proceso. Sin embargo, aunque parezca un escenario bastante común, no hay una solución directa a este problema que se proponga en la literatura. Es por esto que la principal contribución de este TFM consiste en diseñar este mecanismo de colaboración entre empresas de forma segura, adaptando a un ámbito empresarial la propuesta de arquitectura jerárquica descrita por [45, 46].

En esta perspectiva, se plantea un escenario de aplicación para el sistema con el objetivo de ilustrar su funcionamiento y medir su rendimiento frente a alternativas tradicionales de ML, donde no se tiene privacidad de los datos. El escenario en cuestión se seleccionó en relación con otra de las limitaciones encontradas en la literatura, que es la inexistencia de un trabajo previo que aborde la predicción de la somnolencia al volante mediante BCIs y FL. Para lograr la implementación del sistema jerárquico propuesto se definieron tres casos de uso, siendo cada uno de ellos incremental respecto al anterior, tanto en complejidad como en la utilización de FL. El primero de ellos mantiene a las empresas aisladas (no hay colaboración) y solo conserva la seguridad los datos dentro de cada empresa. En el segundo, se da una colaboración entre empresas, manteniendo la privacidad de los datos entre las empresas pero no intraempresa. Finalmente, el tercer caso de uso une los dos anteriores, logrando la arquitectura jerárquica deseada mediante una colaboración entre empresas que mantiene la privacidad en los dos niveles, dentro de cada empresa y entre empresas.

A partir de la experimentación realizada se midió el rendimiento de los modelos federados frente a las versiones tradicionales sin privacidad de datos. Se estudió el comportamiento de los modelos tanto para conductores conocidos, con los que se entrenó el modelo, como para nuevos, ajenos a la fase de entrenamiento del modelo, efectuando varias pruebas que dejaban fuera de la fase de entrenamiento a más conductores cada vez. Se obtuvo como resultado que los modelos con ML presentan una pérdida de hasta el 4%, siendo de hasta el 6% para el caso de FL. Sin embargo, cuando se evalúan los modelos en el grupo de conductores nuevos, los modelos generales presentan un rendimiento superior que los por empresa, con mejora del rendimiento del 3% para ML y del 5% con FL. Esto implica que los modelos federados tienden a generalizar me-

jor frente a los tradicionales, y que suelen adaptarse mejor a las peculiaridades de los conductores con los que se entrenan. Esto también demuestra que existe un beneficio para las empresas que participan en el proceso ya sea mediante aprendizaje federado o una alternativa tradicional, ya que obtienen un modelo predictivo con unas prestaciones y una capacidad de generalización mayor. Además, la pérdida de rendimiento de la versión jerárquica del caso de uso tercero respecto al segundo, que solo mantenía la privacidad entre empresas, es mínima (entre el 0% y el 2%). Cabe destacar que el escenario identificado como non-IID no ha supuesto problemas de convergencia en los modelos federados. De esta forma, se concluye que el mecanismo de FL jerárquico propuesto funciona y es válido para este escenario. Además, la diferencia de rendimiento entre las versiones tradicionales y federadas es lo suficientemente pequeña como para optar siempre que sea posible por la versión federada, ya que añade grandes beneficios de seguridad al proceso.

Estos resultados acerca de FL y el mecanismo jerárquico para la colaboración entre empresas diseñado han sido positivos para este escenario en particular, lo que abre las puertas a indagar y medir su rendimiento en otros escenarios de aplicación. Se plantean las siguientes vías futuras para este trabajo. En primer lugar, sería de gran interés el generar un conjunto de datos multimodal propio mediante una BCI y desplegando un simulador de conducción. De esta forma, sería posible contrastar los resultados y las conclusiones que se han obtenido del presente trabajo. Otro análisis que sería muy interesante realizar es el de incorporar los mecanismos de *Secure Aggregation*, *Differential Privacy* u *Homomorphic Encryption* al proceso de entrenamiento federado. Estas medidas agregan una capa adicional de protección de la seguridad de los datos, por lo que es interesante incorporarlas y evaluar la pérdida de rendimiento que se introduce. También, se contempla la posibilidad de emplear frameworks más potentes como *TensorFlow Federated* o la librería de IBM que permitan efectuar una experimentación más potente y optimizada en recursos que *Flower* a la vez que permiten el entrenamiento de otros algoritmos de ML de manera federada como Random Forest, K-means o Naïve Bayes.

---

# Bibliografía

- [1] Jaakko Malmivuo, Robert Plonsey, et al. *Bioelectromagnetism: principles and applications of bioelectric and biomagnetic fields*. Oxford University Press, USA, 1995. doi: 10.1093/acprof:oso/9780195058239.003.0013.
- [2] Alberto López, Francisco Ferrero, and Octavian Postolache. An affordable method for evaluation of ataxic disorders based on electrooculography. *Sensors*, 19(17), 2019. doi: 10.3390/s19173756.
- [3] Wei-Long Zheng and Bao-Liang Lu. A multimodal approach to estimating vigilance using eeg and forehead eeg, 2017. URL <http://stacks.iop.org/1741-2552/14/i=2/a=026017>.
- [4] Reglamento (ue) 2016/679 del parlamento europeo y del consejo de 27 de abril de 2016 relativo a la protección de las personas físicas en lo que respecta al tratamiento de datos personales y a la libre circulación de estos datos y por el que se deroga la directiva 95/46/ce (reglamento general de protección de datos). *Diario Oficial de la Unión Europea*, 2016.
- [5] Health insurance portability and accountability act. *Department of Health and Human Services*, 1996.
- [6] Mu Li, David G. Andersen, Jun Woo Park, Alexander J. Smola, Amr Ahmed, Vanja Josifovski, James Long, Eugene J. Shekita, and Bor-Yiing Su. Scaling distributed machine learning with the parameter server. In *Proceedings of the 11th USENIX Conference on Operating Systems Design and Implementation*, OSDI'14, page 583–598, USA, 2014. USENIX Association. ISBN 9781931971164.
- [7] Vivienne Sze, Yu-Hsin Chen, Joel Emer, Amr Suleiman, and Zhengdong Zhang. Hardware for machine learning: Challenges and opportunities. In *2017 IEEE Custom Integrated Circuits Conference (CICC)*, pages 1–8, 2017. doi: 10.1109/CICC.2017.7993626.
- [8] Lina Zhou, Shimei Pan, Jianwu Wang, and Athanasios V. Vasilakos. Machine learning on big data: Opportunities and challenges. *Neurocomputing*, 237:350–361, 2017. doi: 10.1016/j.neucom.2017.01.026.
- [9] Reglamento (ue) 2019/2144 del parlamento europeo y del consejo de 27 de noviembre de 2019 relativo a los requisitos de homologación de tipo de los vehículos

- de motor y de sus remolques, así como de los sistemas, componentes y unidades técnicas independientes destinados a esos vehículos, en lo que respecta a su seguridad general y a la protección de los ocupantes de los vehículos y de los usuarios vulnerables de la vía pública, por el que se modifica el reglamento (ue) 2018/858 del parlamento europeo y del consejo y se derogan los reglamentos (ce) 78/2009, (ce) 79/2009 y (ce) 661/2009 del parlamento europeo y del consejo y los reglamentos (ce) 631/2009, (ue) 406/2010, (ue) 672/2010, (ue) 1003/2010, (ue) 1005/2010, (ue) 1008/2010, (ue) 1009/2010, (ue) 19/2011, (ue) 109/2011, (ue) 458/2011, (ue) 65/2012, (ue) 130/2012, (ue) 347/2012, (ue) 351/2012, (ue) 1230/2012 y (ue) 2015/166 de la comisión. *Diario Oficial de la Unión Europea*, L 325, 2019.
- [10] Jamie Ward. *The student's guide to cognitive neuroscience*. Routledge, 2019.
- [11] Diego Redolar Ripoll. *Neurociencia cognitiva*. QUITO/UIDE/2015, 2015.
- [12] Edmundo Rosales Mayor and Jorge Rey De Castro Mujica. Somnolencia: Qué es, qué la causa y cómo se mide. *Acta médica peruana*, 27(2):137–143, 2010.
- [13] Benny Mwengue Gimbada and Daniel Rodenstein. Evaluación de la somnolencia. *Archivos de Bronconeumología*, 45(7):349–351, 2009. doi: 10.1016/j.arbres.2008.10.002.
- [14] David F. Dinges and John W. Powell. Microcomputer analyses of performance on a portable, simple visual RT task during sustained operations. *Behavior Research Methods, Instruments, & Computers*, 17(6):652–655, 1985. doi: 10.3758/bf03200977.
- [15] Paul Whitney and John M. Hinson. Measurement of cognition in studies of sleep deprivation. *Progress in Brain Research*, 185:37–48, 2010. doi: 10.1016/B978-0-444-53702-7.00003-8.
- [16] Yu-Ting Liu, Shang-Lin Wu, Kuang-Pen Chou, Yang-Yin Lin, Jie Lu, Guangquan Zhang, Wen-Chieh Lin, and Chin-Teng Lin. Driving fatigue prediction with pre-event electroencephalography (eeg) via a recurrent fuzzy neural network. In *2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 2488–2494, 2016. doi: 10.1109/FUZZ-IEEE.2016.7738006.
- [17] Yuqi Cui, Yifan Xu, and Dongrui Wu. Eeg-based driver drowsiness estimation using feature weighted episodic training. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 27(11):2263–2273, 2019. doi: 10.1109/TNSRE.2019.2945794.
- [18] United States. Federal Motor Carrier Safety Administration. Technology Division. Perclos: A valid psychophysiological measure of alertness as assessed by psychomotor vigilance, 1998.
-

- 
- [19] Noa Quevedo López. Estudio del parpadeo durante la conducción de vehículos (aspectos cognitivos y de flujo de información). Master's thesis, Universitat Politècnica de Catalunya, 2012.
- [20] Control de fatiga y somnolencia de conductores para buses y camiones, 2020. URL <https://www.movilflix.com/control-de-fatiga-y-somnolencia/>.
- [21] Thomas A. Dingus, H. Lenora Hardee, and Walter W. Wierwille. Development of models for on-board detection of driver impairment. *Accident Analysis & Prevention*, 19(4):271–283, 1987. doi: 10.1016/0001-4575(87)90062-5.
- [22] Tomohiko Igasaki, Kazuki Nagasawa, Nobuki Murayama, and Zhencheng Hu. Drowsiness estimation under driving environment by heart rate variability and/or breathing rate variability with logistic regression analysis. In *2015 8th International Conference on Biomedical Engineering and Informatics (BMEI)*, pages 189–193, 2015. doi: 10.1109/BMEI.2015.7401498.
- [23] Izzat Aulia Akbar and Tomohiko Igasaki. Drowsiness estimation using electroencephalogram and recurrent support vector regression. *Information*, 10(6), 2019. doi: 10.3390/info10060217.
- [24] G. Borghini, G. Vecchiato, J. Toppi, L. Astolfi, A. Maglione, R. Isabella, C. Caltagirone, W. Kong, D. Wei, Z. Zhou, L. Polidori, S. Vitiello, and F. Babiloni. Assessment of mental fatigue during car driving by using high resolution eeg activity and neurophysiologic indices. In *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 6442–6445, 2012. doi: 10.1109/EMBC.2012.6347469.
- [25] A. Maglione, G. Borghini, P. Aricò, F. Borgia, I. Graziani, A. Colosimo, W. Kong, G. Vecchiato, and F. Babiloni. Evaluation of the workload and drowsiness during car driving by using high resolution eeg activity and neurophysiologic indices. In *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 6238–6241, 2014. doi: 10.1109/EMBC.2014.6945054.
- [26] Luis Fernando Nicolas-Alonso and Jaime Gomez-Gil. Brain computer interfaces, a review. *Sensors*, 12(2), 2012. doi: 10.3390/s120201211.
- [27] Sergio López Bernal, Alberto Huertas Celdrán, Gregorio Martínez Pérez, Michael Taynnan Barros, and Sasitharan Balasubramaniam. Security in brain-computer interfaces. *ACM Computing Surveys*, 54(1):1–35, 2022. doi: 10.1145/3427376.
- [28] Pablo Ballarin Usieto and Javier Minguez. La importancia de la ciberseguridad en brain-computer interfaces, 2019. URL <https://www.bitbrain.com/es/blog/ciberseguridad-cerebro-computadora>.
-

- 
- [29] Izzat A. Akbar, Arthur M. Rumagit, Mitaku Utsunomiya, Takamasa Morie, and Tomohiko Igasaki. Three drowsiness categories assessment by electroencephalogram in driving simulator environment. In *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2017. doi: 10.1109/embc.2017.8037464.
- [30] Enrique Tomás Martínez Beltrán, Mario Quiles Pérez, Sergio López Bernal, Gregorio Martínez Pérez, and Alberto Huertas Celdrán. Safecar: A brain-computer interface and intelligent framework to detect drivers' distractions. *Expert Systems with Applications*, 203:117402, 2022. doi: 10.1016/j.eswa.2022.117402.
- [31] Byung-Chan Chang, Jung-Eun Lim, Hae-Jin Kim, and Bo-Hyeok Seo. A study of classification of the level of sleepiness for the drowsy driving prevention. In *SICE Annual Conference 2007*. IEEE, 2007. doi: 10.1109/sice.2007.4421521.
- [32] Qiang Yang, Yang Liu, Yong Cheng, Yan Kang, Tianjian Chen, and Han Yu. Federated learning, 2019.
- [33] Muhammad Habib ur Rehman. *Federated Learning Systems: Towards Next-Generation AI*, volume 965. Springer Nature, 2021. doi: 10.1007/978-3-030-70604-3.
- [34] Jakub Konečný, H. Brendan McMahan, Felix X. Yu, Peter Richtarik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. In *NIPS Workshop on Private Multi-Party Machine Learning*, 2016. URL <https://arxiv.org/abs/1610.05492>.
- [35] Chen Zhang, Yu Xie, Hang Bai, Bin Yu, Weihong Li, and Yuan Gao. A survey on federated learning. *Knowledge-Based Systems*, 216:106775, 2021. doi: 10.1016/j.knosys.2021.106775.
- [36] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Trans. Intell. Syst. Technol.*, 10(2), 2019. doi: 10.1145/3298981.
- [37] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54, pages 1273–1282, 2016.
- [38] Fate: An industrial grade federated learning framework. URL <https://fate.fedai.org/>.
- [39] Pysyft - openmined blog. URL <https://blog.openmined.org/tag/pysyft/>.
- [40] Flower: A friendly federated learning framework. URL <https://flower.dev>.
-

- 
- [41] Ibm federated learning. URL <https://ibmfl.mybluemix.net>.
- [42] Tensorflow federated. URL <https://www.tensorflow.org/federated>.
- [43] Timothy Yang, Galen Andrew, Hubert Eichner, Haicheng Sun, Wei Li, Nicholas Kong, Daniel Ramage, and Françoise Beaufays. Applied federated learning: Improving google keyboard query suggestions. *arXiv preprint arXiv:1812.02903*, 2018.
- [44] Andrew Hard, Kanishka Rao, Rajiv Mathews, Swaroop Ramaswamy, Françoise Beaufays, Sean Augenstein, Hubert Eichner, Chloé Kiddon, and Daniel Ramage. Federated learning for mobile keyboard prediction. *arXiv preprint arXiv:1811.03604*, 2018.
- [45] Lumin Liu, Jun Zhang, S.H. Song, and Khaled B. Letaief. Client-edge-cloud hierarchical federated learning. In *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, pages 1–6, 2020. doi: 10.1109/ICC40277.2020.9148862.
- [46] Aidmar Wainakh, Alejandro Sanchez Guinea, Tim Grube, and Max Mühlhäuser. Enhancing privacy via hierarchical federated learning. In *2020 IEEE European Symposium on Security and Privacy Workshops (EuroS PW)*, pages 344–347, 2020. doi: 10.1109/EuroSPW51379.2020.00053.
- [47] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems*, 2:429–450, 2020.
- [48] Tian Li, Maziar Sanjabi, Ahmad Beirami, and Virginia Smith. Fair resource allocation in federated learning. *arXiv preprint arXiv:1905.10497*, 2019.
- [49] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for privacy-preserving machine learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS '17*, page 1175–1191. Association for Computing Machinery, 2017. doi: 10.1145/3133956.3133982.
- [50] Cynthia Dwork and Kobbi Nissim. Privacy-preserving datamining on vertically partitioned databases. In *Advances in Cryptology – CRYPTO 2004*, pages 528–544, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg. doi: 10.1007/978-3-540-28628-8\_32.
- [51] Ronald L Rivest, Len Adleman, Michael L Dertouzos, et al. On data banks and privacy homomorphisms. *Foundations of secure computation*, 4(11):169–180, 1978.
-



- 
- [52] Craig Gentry. Fully homomorphic encryption using ideal lattices. In *Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing*, page 169–178. Association for Computing Machinery, 2009. ISBN 9781605585062. doi: 10.1145/1536414.1536440.
- [53] Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin, and Vikas Chandra. Federated learning with non-iid data, 2018.
- [54] Hangyu Zhu, Jinjin Xu, Shiqing Liu, and Yaochu Jin. Federated learning on non-iid data: A survey. *Neurocomputing*, 465:371–390, 2021. doi: 10.1016/j.neucom.2021.07.098.
- [55] Sawsan Abdulrahman, Hanine Tout, Hakima Ould-Slimane, Azzam Mourad, Chamseddine Talhi, and Mohsen Guizani. A survey on federated learning: The journey from centralized to distributed on-site learning and beyond. *IEEE Internet of Things Journal*, 8(7):5476–5497, 2021. doi: 10.1109/JIOT.2020.3030072.
- [56] Y. Chen, X. Qin, J. Wang, C. Yu, and W. Gao. Fedhealth: A federated transfer learning framework for wearable healthcare. *IEEE Intelligent Systems*, 35(04): 83–93, 2020. doi: 10.1109/MIS.2020.2988604.
- [57] Gábor Szegedi, Péter Kiss, and Tomás Horváth. Evolutionary federated learning on eeg-data. In *ITAT*, pages 71–78, 2019.
- [58] Dashan Gao, Ce Ju, Xiguang Wei, Yang Liu, Tianjian Chen, and Qiang Yang. Hhhfl: Hierarchical heterogeneous horizontal federated learning for electroencephalography. *arXiv preprint arXiv:1909.05784*, 2019.
- [59] Ce Ju, Dashan Gao, Ravikiran Mane, Ben Tan, Yang Liu, and Cuntai Guan. Federated transfer learning for eeg signal classification. In *2020 42nd Annual International Conference of the IEEE Engineering in Medicine Biology Society (EMBC)*, pages 3040–3045, 2020. doi: 10.1109/EMBC44109.2020.9175344.
- [60] Chen Zhao, Zhipeng Gao, Qian Wang, Kaile Xiao, Zijia Mo, and M Jamal Deen. Fedsup: A communication-efficient federated learning fatigue driving behaviors supervision framework. *arXiv preprint arXiv:2104.12086*, 2021.
- [61] Gianluca Di Flumeri. C2 - car - simulator - biometric features, 2021.
- [62] Simulator of behavioural aspects for safer transport, 2017. URL <https://cordis.europa.eu/project/id/723386>.
- [63] 64-channel quik-cap. URL <https://compumedicsneuroscan.com/product/64-channels-quik-cap-synamps-2-rt/>.
-

- 
- [64] Smi eye tracking glasses - imotions. URL <https://imotions.com/hardware/smi-eye-tracking-glasses/>.
- [65] Feng Zhou, Areen Alsaied, Mike Blommer, Reates Curry, Radhakrishnan Swaminathan, Dev Kochhar, Walter Talamonti, Louis Tijerina, and Baiying Lei. Driver fatigue transition prediction in highly automated driving using physiological features. *Expert Systems with Applications*, 147:113204, 2020. doi: 10.1016/j.eswa.2020.113204.
- [66] Burcu Kir Savas and Yasar Becerikli. Real time driver fatigue detection system based on multi-task ConNN. *IEEE Access*, 8:12491–12498, 2020. doi: 10.1109/access.2020.2963960.
- [67] Qianyang Zhuang, Zhang Kehua, Jiayi Wang, and Qianqian Chen. Driver fatigue detection method based on eye states with pupil and iris segmentation. *IEEE Access*, 8:173440–173449, 2020. doi: 10.1109/access.2020.3025818.
- [68] Wang Huan Gu, Yu Zhu, Xu Dong Chen, Lin Fei He, and Bing Bing Zheng. Hierarchical CNN-based real-time fatigue detection system by visual-based technologies using MSP model. *IET Image Processing*, 12(12):2319–2329, 2018. doi: 10.1049/iet-ipr.2018.5245.
- [69] Jia Wu, Xiu-Yun Chen, Hao Zhang, Li-Dong Xiong, Hang Lei, and Si-Hao Deng. Hyperparameter optimization for machine learning models based on bayesian optimization. *Journal of Electronic Science and Technology*, 17(1):26–40, 2019. doi: 10.11989/JEST.1674-862X.80904120.
- [70] Distribución normal - matlab & simulink - mathworks española. URL <https://es.mathworks.com/help/stats/normal-distribution.html>.
- [71] José Manuel Hidalgo Rogel. Framework para la detección de somnolencia en escenarios de conducción usando interfaces cerebro-máquina. Master’s thesis, Universidad de Murcia, 2021.
- [72] Theodora S. Brisimi, Ruidi Chen, Theofanie Mela, Alex Olshevsky, Ioannis Ch. Paschalidis, and Wei Shi. Federated learning of predictive models from federated electronic health records. *International Journal of Medical Informatics*, 112:59–67, 2018. doi: 10.1016/j.ijmedinf.2018.01.007.
- [73] Kai Yang, Tao Jiang, Yuanming Shi, and Zhi Ding. Federated learning via over-the-air computation. *IEEE Transactions on Wireless Communications*, 19(3):2022–2035, 2020. doi: 10.1109/TWC.2019.2961673.
- [74] Dianbo Liu, Timothy Miller, Raheel Sayeed, and Kenneth D. Mandl. Fadl: Federated-autonomous deep learning for distributed electronic health record. *ArXiv*, abs/1811.11400, 2018.
-

- 
- [75] Li Huang, Andrew L. Shea, Huining Qian, Aditya Masurkar, Hao Deng, and Dianbo Liu. Patient clustering improves efficiency of federated machine learning to predict mortality and hospital stay time using distributed electronic medical records. *Journal of Biomedical Informatics*, 99:103291, 2019. doi: 10.1016/j.jbi.2019.103291.
- [76] Santiago Silva, Boris A. Gutman, Eduardo Romero, Paul M. Thompson, Andre Altmann, and Marco Lorenzi. Federated learning in distributed medical databases: Meta-analysis of large-scale subcortical brain data. In *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, pages 270–274, 2019. doi: 10.1109/ISBI.2019.8759317.
- [77] Yang Zhao, Jun Zhao, Linshan Jiang, Rui Tan, and Dusit Niyato. Mobile edge computing, blockchain and reputation-based crowdsourcing iot federated learning: A secure, decentralized and privacy-preserving system. *arXiv*, 2020.
- [78] Haoye Chai, Supeng Leng, Yijin Chen, and Ke Zhang. A hierarchical blockchain-enabled federated learning algorithm for knowledge sharing in internet of vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 22(7):3975–3986, 2021. doi: 10.1109/TITS.2020.3002712.
- [79] Pedro Miguel Sánchez Sánchez, Alberto Huertas Celdrán, José Rafael Buendía Rubio, G  r  me Bovet, and Gregorio Mart  nez P  rez. Robust federated learning for execution time-based device model identification under label-flipping attack. *arXiv preprint arXiv:2111.14434*, 2021.
- [80] Valerian Rey, Pedro Miguel S  nchez S  nchez, Alberto Huertas Celdr  n, and G  r  me Bovet. Federated learning for malware detection in iot devices. *Computer Networks*, 204:108693, 2022. doi: 10.1016/j.comnet.2021.108693.
- [81] Pedro Miguel S  nchez S  nchez, Alberto Huertas Celdr  n, Timo Schenk, Adrian Lars Benjamin Iten, G  r  me Bovet, Gregorio Mart  nez P  rez, and Burkhard Stiller. Studying the robustness of anti-adversarial federated learning models detecting cyberattacks in iot spectrum sensors. *arXiv preprint arXiv:2202.00137*, 2022.
- [82] Boyi Liu, Lujia Wang, and Ming Liu. Lifelong federated reinforcement learning: A learning architecture for navigation in cloud robotic systems. *IEEE Robotics and Automation Letters*, 4(4):4555–4562, 2019. doi: 10.1109/LRA.2019.2931179.
- [83] Solmaz Niknam, Harpreet S. Dhillon, and Jeffrey H. Reed. Federated learning for wireless communications: Motivation, opportunities, and challenges. *IEEE Communications Magazine*, 58(6):46–51, 2020. doi: 10.1109/MCOM.001.1900461.
-

- 
- [84] Christopher Briggs, Zhong Fan, and Peter Andras. Federated learning with hierarchical clustering of local updates to improve training on non-iid data. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–9, 2020. doi: 10.1109/IJCNN48605.2020.9207469.
-