UNIVERSITAT POLITÈCNICA DE VALÈNCIA

_ TELECOM ESCUELA TÉCNICA VLC SUPERIOR DE UPV INGENIEROS DE TELECOMUNICACIÓN

DISEÑO Y DESARROLLO DE UN SISTEMA AUTOMÁTICO PARA LA PREDICCIÓN DEL NIVEL DE FUNCIONALIDAD QUE PUEDEN ALCANZAR PACIENTES CON TRAUMATISMO CRANEOENCEFÁLICO A PARTIR DE VALORACIONES FUNCIONALES Y TÉCNICAS DE MACHINE LEARNING.

Vicente Moreno Quintana

Tutor: Valeriana Naranjo Ornedo

Cotutor: Julio José Silva Rodríguez

Trabajo Fin de Máster presentado en la Escuela Técnica Superior de Ingenieros de Telecomunicación de la Universitat Politècnica de València, para la obtención del Título de Máster Universitario en Ingeniería de Telecomunicación

Curso 2019-20

Valencia, 1 de septiembre de 2020

Resumen

Los traumatismos craneoencefálicos son la causa principal de fallecimiento y pérdida de funcionalidad en el conjunto de enfermedades traumáticas. La neurorrehabilitación es uno de los principales tratamientos utilizados para mejorar las capacidades funcionales pérdidas en diversos dominios (motores, cognitivos, independencia, logopedia, etc.). En el proceso del tratamiento se evalúa al paciente a través de valoraciones funcionales utilizando escalas clínicas. Esta evolución se manifiesta de múltiples formas en cada individuo, con grupos particulares de pacientes en los cuales no se observa una mejora de los niveles funcionales, esto supone un alto coste en recursos y la necesidad en algunos pacientes de modificar el tratamiento.

El proyecto actual se basa en el desarrollo de un sistema automático basado en técnicas de *machine learning* capaces de predecir el nivel de funcionalidad final que podrán lograr los pacientes que hayan sufrido un traumatismo cerebral. Los datos utilizados para entrenar estos sistemas son las variables tomadas durante las valoraciones de los pacientes en el proceso de neurorrehabilitación, facilitados por el servicio de neurorrehabilitación de Hospitales Nisa.

La herramienta desarrollada realizará en primer lugar un correcto preprocesamiento y adecuación de los datos para poder utilizarlos en los modelos de inteligencia artificial. Posteriormente permitirá analizar e identificar las variables óptimas para usarlas en los modelos de predicción, los cuales son finalmente entrenados y evaluados. Este sistema podrá generar y almacenar los modelos predictivos y los datos generados, para realizar diferentes estudios utilizando variables estáticas, dinámicas e incorporando la evolución temporal de estas variables gracias a las técnicas de *Long-Short Term Memory*. Al realizar estos estudios se podrá evaluar la mejorará en el nivel de predicción al utilizar ciertos modelos de *machine learning* y variables clínicas. En el estudio realizado en la memoria el sistema desarrollado es capaz de obtener una correlación del 93% en la predicción de la funcionalidad final del paciente.

Palabras clave*: Machine learning, deep learning*, sistemas de predicción, traumatismo craneoencefálico, FIM y FAM.

Abstract

Traumatic brain injury are the main cause of death and loss of functionality in traumatic diseases domain. Neurorehabilitation is one of the main treatments used to improve functional capacities lost in various domains (motor, cognitive, independence, speech therapy, etc.). In the treatment process, the patient is evaluated through functional assessments using clinical scales. This evolution manifests itself in multiple ways in each individual, with particular groups of patients in whom an improvement in functional levels is not observed, this implies a high cost in resources and the need in some patients to modify the treatment.

The current project is based on the development of an automatic system based on machine learning techniques capable of predicting the level of final functionality that patients who have suffered a brain injury will be able to achieve. The data used to train these systems are the variables taken during the evaluations of patients in the neurorehabilitation process, provided by the neurorehabilitation service of Nisa Hospitals.

The developed system will first carry out a correct pre-processing and adaptation of the data to be able to use them in the artificial intelligence models. Subsequently, the system will allow to analyze and identify the optimal variables in order to use them in the prediction models, which are finally trained and evaluated. This system will be able to generate and store the predictive models and the data generated, to carry out different studies using static and dynamic variables and incorporating the temporal evolution of these variables thanks to the Long-Short Term Memory techniques. When conducting these studies, it will be possible to evaluate the improvement in the level of prediction when using certain machine learning models and clinical variables. In the study carried out in memory, the developed system is capable of obtaining a correlation of 93% in the prediction of the final functionality of the patient.

Keywords: Machine learning, deep learning, prediction systems, traumatic brain injury, FIM and FAM.

Índice

1. INTRODUCCIÓN

1.1. TCE: Traumatismo cráneo encefálico

Un traumatismo craneoencefálico (TCE) se define como la afectación del cerebro causada por una fuerza externa que puede producir una disminución o disfunción del nivel de conciencia e implica una alteración de las habilidades cognitivas, físicas y emocionales del individuo. Actualmente es la principal causa de fallecimiento y pérdida de funcionalidad del conjunto de enfermedades traumáticas, con 69 millones de casos anuales a nivel mundial. Los accidentes de tráfico suponen la causa más importante con un total del 73% de los casos, seguido de las caídas (20%) y lesiones deportivas (5%) [1].

En función del nivel de pérdida de la disfunción que sufre el individuo de los TCE suelen clasificarse en distintos grados, el primero sería leve o también conocido como "conmoción cerebral" este tipo de traumatismo es el más frecuente, no suele haber pérdida de conocimiento o su duración suele ser de unos minutos después de la contusión. La mayoría de las personas que sufren un TCE leve suelen recuperarse de forma completa en los días o semanas siguientes a la conmoción. El segundo grado que se encuentra es moderado, con una pérdida de conocimiento con un margen de tiempo entre 30 minutos y un día. Los pacientes suelen tener un periodo de amnesia post-traumática inferior a una semana en que tienen dificultades para aprender información nueva. Por último, el TCE grave donde la pérdida de conocimiento es mayor a un día y el periodo de amnesia post-traumática es mayor a una semana.

La pérdida de conciencia es uno de los primeros resultados tras un TCE y su duración y nivel son dos de los factores más relevantes en la gravedad del mismo. La mayoría de los pacientes después de recuperar gradualmente el nivel de conciencia y orientación sufren secuelas físicas (alteraciones motoras y sensoriales), cognitivas (dificultad de abstracción y resolución de problemas, trastornos de aprendizaje y memoria) y de comportamiento (control de conducta, alteraciones personales y emocionales) que varían tanto de la gravedad del TCE como de las propiedades de la personalidad e inteligencia anteriores a la conmoción.

1.2. FIM y FAM: valoraciones de funcionalidad

La escala FIM (Medida de independencia funcional) y FAM (Medida de evaluación funcional) es una escala directamente proporcional al nivel de independencia del paciente en distintas actividades básicas de un individuo, es decir permite medir la cantidad de ayuda que un individuo necesita para determinadas funciones.

Para poder realizar las valoraciones del paciente en esta escala hay una serie de principios básicos de puntuación que deben seguir los evaluadores y se resume en los siguientes párrafos.

- La puntuación se debe decidir en grupo acordando la puntuación más justa, en caso de desacuerdo entre miembros se aceptará la puntuación menor.

- La funcionalidad tiene que ser evaluada por el terapeuta con observación directa del paciente, por lo tanto, se requiere que este terapeuta este familiarizado con dicho paciente. La puntuación en el momento de admisión se debe realizar durante los 10 días iniciales de trabajo con el paciente y la puntuación al alta se realizará durante la semana posterior a esta.

- El paciente debe evaluarse en función de las actividades que realiza a lo largo del día y no basándose en lo que podría haber hecho en circunstancias diferentes, por lo tanto, el FIM depende del entorno y puede ser amigable o no para el paciente.

- El valor de la puntuación por cada ítem varía entre 1 y 7. Generalmente se puntúa como 1 a 4 si requiere la ayuda de una persona, y en función de la frecuencia de intervención, siendo dependencia completa (1 o 2) en caso de no poder realizar el 50% de la tarea y asistencia moderada (3) o asistencia mínima (4) cuando la autosuficiencia del paciente para realizar la tarea el mayor a 50% pero aún necesita de ayuda externa. En el caso de que únicamente requiera supervisión sin contacto físico, se otorga la puntuación de 5 y por último si no se necesita ayuda de una persona se puntúa 6 o 7 según la rapidez, la seguridad y la ayuda técnica que necesite el paciente.

- No se debe dejar una puntuación en blanco, si el paciente no es capaz de ser evaluado en algún ítem se debe indicar la puntuación mínima de 1.

- Una puntuación del valor de 1 en un ítem significa.:
    a. No puede realizar la actividad de ninguna manera
    b. Necesita ayuda de 2 personas para realizar la actividad
    c. El paciente se puede poner en riesgo de lesión si se evalúa la actividad
    d. La información no está accesible

- Se debe usar un árbol de decisiones para realizar la valoración y comprobar que el nivel descrito coincide con la puntuación. El árbol de decisión general de puntuación se representa en la figura 1:

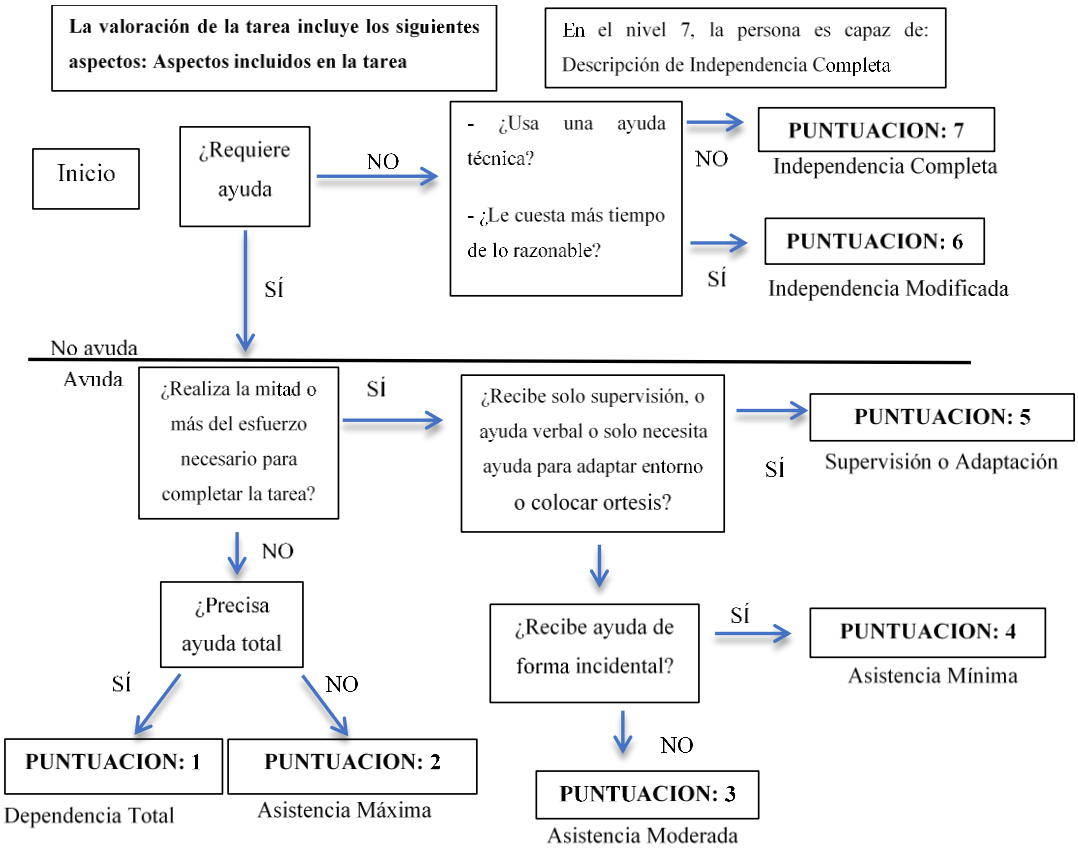


Figura 1. Árbol de decisión FIM/FAM

Los evaluadores utilizan el árbol de decisión de la figura 1 para evaluar a los pacientes cada tarea realizada. Estas tareas se pueden clasificar en cognitivas y motoras de la escala FIM/FAM, las cuales se encuentran en las siguientes tablas desglosadas por las respectivas subtareas asociadas. Cada subtarea vale un total de 7 puntos y la suma total de ellas representa el valor de la tarea principal.

ESCALA FIM/FAM MOTORA:

Actividades Básicas	Control de esfínteres	Movilidad
Alimentación	Control Vesical-Grado de asistencia	Transferencias cama/silla
Deglución	Control Vesical-Frecuencia de escapes	Transferencias WC
Higiene Personal	Control Intestinal-Grado de asistencia	Transferencias bañera/ducha
Baño	Control Intestinal-Frecuencia de escapes	Transferencias al vehículo
Vestido parte superior		Locomoción Marcha/silla
Vestido parte inferior		Locomoción escaleras
WC		Locomoción desplazamientos

Tabla 1. FIM/FAM Motora.

ESCALA FIM/FAM COGNITIVA:

Comunicación	Interacción psicosocial	Funciones cognitivas	Actividades avanzadas día a día
Compresión	Interacción social	Resolución de Problemas	Preparación de comida
Expresión	Estado emocional	Memoria	Colada
Lectura	Ajuste a limitaciones/conciencia de enfermedad	Orientación	Limpieza domestica
Escritura	Ocio y tiempo libre	Concentración	Compras
Articulación/Inteligibilidad del lenguaje		Conciencia de riesgos y seguridad	Finanzas económicas
			Trabajo/Educación

Tabla 2.FIM/FAM Cognitiva.

1.3. Neurorrehabilitación

Padecer un daño cerebral pone en riesgo la vida de la persona que lo padece y la atención urgente y posterior estabilización clínica son vitales para el paciente. Durante un tiempo posterior al traumatismo puede que el paciente requiera permanecer en el centro hospitalario de agudos hasta lograr una estabilidad clínica y cuando se consigue se procede al alta para posteriormente enfrentarse a un proceso de neurorrehabilitación.

La neurorrehabilitación se define como un proceso médico complejo con finalidad de ayudar a la recuperación de una lesión del sistema nervioso y minimizar o nivelar cualquier variación funcional resultante de la misma. La neurorrehabilitación es uno de los principales tratamientos para pacientes con TCE y se utiliza con el objetivo de mejorar los niveles de funcionalidad disminuidos, además esta debe ser [1]:

- **Temprana:** Es primordial empezar la rehabilitación lo antes posible después de la estabilización clínica.
- **Individualizada:** Debe ser personalizada para cada paciente, teniendo en cuenta sus características y necesidades.
- **Intensiva:** Se debe adecuar un nivel de trabajo a las capacidades del paciente.
- **Multidisciplinar:** Se requiere la intervención de distintos profesionales ofreciendo un trabajo coordinado para obtener una respuesta concreta en cada paciente, ya que las secuelas neurológicas que puede presentar cada paciente son variadas.

Los tratamientos que se ofrecen son diferentes según las características del paciente, generalmente se pueden ofrecer un tratamiento con ingreso hospitalario y un tratamiento ambulatorio [1].

Los pacientes que más se benefician de un ingreso hospitalario suelen ser:

- Pacientes agudos que han conseguido una estabilidad clínica, pero necesitan cuidados médicos.

- Pacientes dependientes que necesitan una adaptación del entorno y una preparación de los familiares para poder cuidar de ellos.

- Pacientes que por su condición cognitiva, física o conductual resulta arduo establecerlos en sus residencias.

Las ventajas que tienen los pacientes que son ingresados en el hospital para ser el tratamiento son:

- Uso de los recursos del hospital para inicia la rehabilitación temprana, atención médica y cuidado de enfermería
- Se ofrece un tratamiento hospitalario intensivo y continuo, que a medida que el paciente mejora sus capacidades se permite entrenar de forma adecuada las actividades de la vida diaria.
- Desde el ingreso del paciente se estudia su situación y se establecen objetivos individualizados en función de la gravedad.

Cuando se consiguen alcanzar los objetivos marcados en el tratamiento hospitalario, se trabaja con el paciente para la integración en su residencia y finalizar la rehabilitación con un tratamiento ambulatorio.

En el proceso de neurorrehabilitación se realizan una serie de actividades para mejorar las capacidades funcionales del paciente durante un periodo de tiempo y se toman medidas cada seis meses de los niveles de funcionalidad FIM y FAM comentados en el punto anterior, además de otros parámetros útiles para poder estudiar la evolución del paciente. Después de un periodo de rehabilitación y según sus características los pacientes que logran cierto nivel de funcionalidad son dados de alta de la neurorrehabilitación, pero en algunos grupos de pacientes no consta una evolución positiva del nivel de funcionalidad durante el periodo de neurorrehabilitación, esto deriva en un alto coste en cuanto a recursos y la posibilidad de variar el proceso de rehabilitación según el paciente.

1.4. Sistemas de Inteligencia Artificial

Se conoce la inteligencia artificial como la simulación de la inteligencia humana por parte de las máquinas, es una disciplina que se basa en crear sistemas que sean capaces de aprender y razonar como un ser humano, utilizando datos como experiencia para entrenar y poder llevar a cabo acciones que maximicen las posibilidades de éxito en un objetivo o tarea.

En los últimos años se ha producido un gran crecimiento en el interés de la sociedad y las empresas por la rama de la inteligencia artificial en concreto por el uso del *machine learning*, ya que se ha comprobado que es una tecnología que puede producir un gran impacto en la sociedad y los negocios a través de la explotación de los datos. Este crecimiento se debe a varias razones:

- **Aumento de la cantidad de datos:** El volumen de la información crece día a día de forma exponencial y la disponibilidad actual de esta enorme cantidad de datos ha aportado valor a las herramientas de *machine learning*. Se estima que en 2025 tendremos entre tres y cuatro veces la cantidad de datos actual en el mundo.

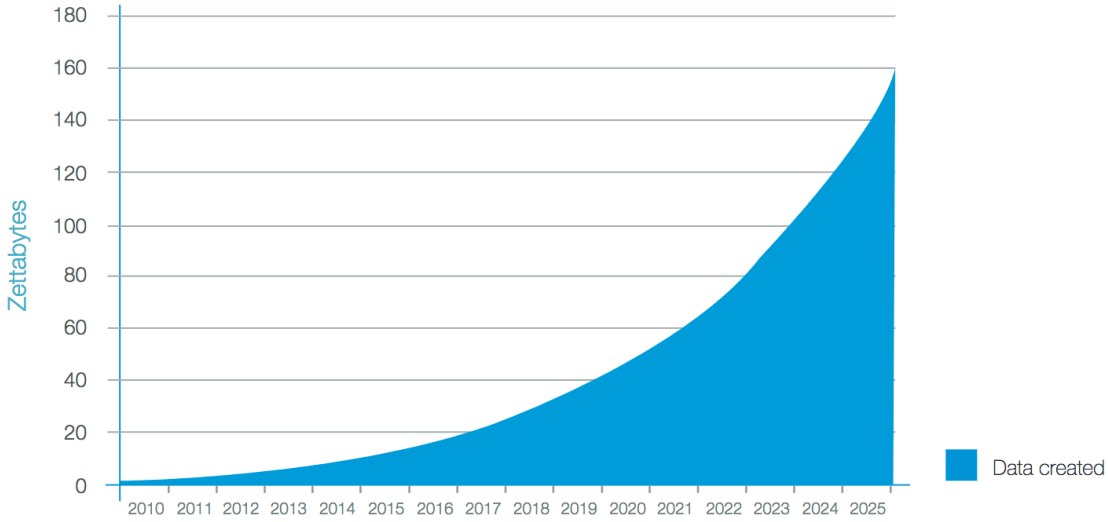


Figura 2. Aumento de datos. Obtenido de [19].

- **Aumento de la capacidad de computación:** La aparición de tecnologías capaces de procesar enormes cantidades de datos, como la incorporación de las GPUs o la computación en la nube han supuesto un gran avance en el uso de herramientas de inteligencia artificial, ya que han permitido reducir este tiempo de computación drásticamente gracias a poder paralelizar el procesamiento de los datos.

- **Aparición de nuevos algoritmos:** Gracias a la investigación en esta rama se han conseguido interesantes resultados con la aparición de nuevos algoritmos.

    - Support vector machines (Corinna Cortes and Vladimir Vapnik-1995)
    - Long short-term memory(Jürgen Schmidhuber and Sepp Hochreiter-1997)
    - Gradient-based learning(Yann LeCun -1998), AlexNet (Alex Krizhevsky-2011)
    - Generative Adversarial Networks (GAN) (equipo de investigadores dirigido por Ian Goodfellow-2014)

El *machine learning* es una rama de la inteligencia artificial que permite crear sistemas que aprenden automáticamente sin necesidad de tener unas reglas programadas, lo que quiere decir que permite identificar patrones complejos en los datos que utilizamos para el entrenar al sistema. La máquina que aprende en este caso es un algoritmo que analiza los datos y es capaz de predecir comportamientos futuros. Dentro del *machine learning* encontramos diferentes clases o grupos de algoritmos que podemos hacer uso: aprendizaje no supervisado, aprendizaje semisupervisado, aprendizaje por refuerzo y aprendizaje supervisado, esta última clase será la que se ha utilizado en este proyecto.

Aprendizaje supervisado: El sistema aprende utilizando datos etiquetados, es decir el algoritmo se alimenta de un conjunto de datos conocidos que incluye las entradas y salidas deseadas, y este debe encontrar patrones en estos datos para establecer un modelo de predicción más preciso. Mientras el algoritmo se encuentra en fase de entrenamiento va realizando predicciones y es corregido gracias a los datos etiquetados de las salidas que debería obtener. Este proceso se repite hasta que se logre un nivel de precisión establecido o finalice el número de iteraciones máximo. Indicar que este la clase de algoritmo es la que hemos utilizado en nuestro sistema de predicción para este proyecto.

Un conjunto de técnicas de aprendizaje que ha adquirido mucho valor en los últimos años ha sido el deep learning. Es un subcampo del *machine learning* que nace haciendo uso de las redes neuronales artificiales, que permiten modelar abstracciones de alto nivel usando arquitecturas computacionales más complejas para el procesado de la información, permitiendo que la máquina aprenda tareas más complicadas que no serían posibles sin el uso de las redes neuronales.

La inteligencia artificial está llegando a muchos sectores en la actualidad y aunque con una integración más lenta también se está integrando al sector sanitario. Las posibilidades que ofrece en este sector son muchas:

- Procesamiento de grandes cantidades de información clínica en poco tiempo.
- Mejorar la calidad de los diagnósticos identificando patologías.
- Innovaciones en los tratamientos ofrecidos a los pacientes.
- Predicción de la evolución del paciente.
- Herramienta de apoyo en países en vías de desarrollo o donde carecen determinados grupos de especialistas.

Con todas estas posibles herramientas la misión actual de la inteligencia artificial en ámbito sanitario es funcionar como herramienta de apoyo con el objetivo de facilitar la labor al personal

sanitario. En este proyecto versa sobre la predicción de la evolución del paciente, donde la finalidad es crear un sistema que en base a algoritmos de *machine learning* que sean capaces de realizar una predicción más exacta de la evolución del nivel de funcionalidad final que podrá tener el paciente haciendo uso de variables dinámicas. Con esta herramienta el personal sanitario podrá tomar mejores decisiones sobre que tratamiento ofrecer a los pacientes según la predicción realizada o también saber cuáles son las variables más significativas que marcan la evolución del paciente.

1.5. Motivación

En el proceso de neurorrehabilitación se debe considerar tanto los puntos de vista del paciente afectado como la perspectiva económica, ya que los tratamientos a los que se someten los pacientes que hayan sufrido un TCE suelen tener costes elevados, teniendo en cuenta no solo el tratamiento del paciente si no también costos indirectos como servicio y apoyo por parte de los cuidadores y familiares. Por lo tanto, los tratamientos ofrecidos a los pacientes deben de ser personalizados para tener más probabilidades de aumentar en nivel final funcional de paciente y disminuir los costes de estos.

Actualmente la información basada en ensayos no tiene respuestas exactas de qué tratamientos funcionan mejor para diferentes pacientes a largo plazo y qué servicios de rehabilitación son la mejor opción, teniendo en cuenta la relación calidad-precio. Además, la rehabilitación de una lesión cerebral traumática (LCT) es un proceso complejo que cuenta con varios desafíos para la investigación clínica, igualmente hay diversas características que dificultan el uso de los ensayos clínicos aleatorios (ECA) para responder a todas las preguntas sobre la rehabilitación de un TCE [7]:

- Actualmente se cuenta con un número relativamente pequeño de estudios relacionados con el TCE y existe una acentuada heterogeneidad con respecto a la gravedad del TCE, la intervención y el contexto clínico. Además, los resultados en los pacientes pueden diferir en cada etapa de la recuperación.
- También existen consideraciones éticas, ya que muchos de los pacientes con una lesión cerebral traumática de moderada a grave pueden no disponer de la capacidad cognitiva para dar un consentimiento informado para participar voluntariamente en la investigación.

- El tiempo en el cual la rehabilitación puede tener efecto en los pacientes puede ser de meses o años, por lo que suele ser más larga que cualquier otro proyecto de investigación financiado.

En la siguiente tabla se muestran las recomendaciones para la práctica clínica utilizando el sistema GRADE (*Grading of Recommendations Assessment,Development, and Evaluation*) para los diferentes enfoques de neurorrehabilitación en un TCE:

Calidad de la Evidencia	Rehabilitación	Categoría del paciente	Resultado	Potencial ahorro de costes	Recomendaciones (Sistema GRADE)
Alta	Intensiva	TBI grave	-Aumento temprano del nivel de independencia -Reducción de la duración de estancia hospitalaria	+	Muy recomendada
Moderada/Alta	Especializada	TBI muy grave/grave	-Independencia mejorada -Cuidado continuo reducido	++	Recomendada
	Programas vocacionales especializados	TBI moderado/grave	Aumento de la productividad	++	Muy recomendad
Moderada	Temprana	TBI grave	-Aumento temprano del nivel de independencia -Reducción de la duración de estancia hospitalaria	+	Recomendada
	Basada en comunidad	TBI moderado/grave	Mejora de la productividad	++	Recomendada
Baja/moderada	Programas de control del comportamiento	TBI con graves problemas de conducta	-Mejora del comportamiento social -Reducción del apoyo del cuidado continuo	+	Recomendada
	Rehabilitación tardía y continua	TBI moderado/grave con discapacidad duradera	Mantenimiento de la independencia/productividad	+/-	Condicionalmente recomendada

Tabla 3. Recomendaciones para el enfoque de neurorehabilitación en un TCE. Obtenido en [7].

Los neurólogos y otros médicos involucrados en la rehabilitación de un TCE deben tener en consideración diferentes aspectos relacionados con los patrones de la lesión, las presentaciones clínicas, los procesos típicos de recuperación en el TCE de moderado a grave y las complicaciones que pueden aparecer en el progreso de la rehabilitación, ya que existen grupos de paciente que no presentan mejorías durante la neurorrehabilitación en todos los niveles de funcionalidad o estos estos están limitados, esto supone modificar el tratamiento y un aumento de los costos del mismo.

Por lo tanto, teniendo en cuenta la heterogeneidad de las secuelas físicas, cognitivas, psicosociales y conductuales después de un TCE, la rehabilitación tiene que tener una orientación individualizada y centrada en los objetivos de cada paciente, necesidades, carencias y recursos [7].

Actualmente se ha intentado realizar predicciones de cuales son los posibles niveles de funcionalidad que puede alcanzar el paciente utilizando modelos simples basados en variables estáticas obteniendo unos resultados no muy aproximados. En este proyecto haciendo el uso de técnicas de inteligencia artificial se van a estudiar modelos más complejos con variables estáticas y dinámicas con el fin de obtener predicciones más precisas, que pueden ayudar a la toma de decisiones clínicas en los pacientes con TCE, con el objetivo de aumentar los niveles de funcionalidad de los pacientes y ahorrar recursos y costes en los tratamientos.

2. OBJETIVOS

El objetivo de este Trabajo Final de Máster es desarrollar un sistema predictivo de los niveles de funcionalidad final que podrán lograr pacientes que han sufrido un traumatismo craneoencefálico, para que el personal sanitario pueda ofrecer tratamientos más óptimos para estos pacientes y conocer las principales variables que son marcadores de la evolución final del paciente. Dentro de este proyecto se han establecido una serie de propósitos para el desarrollo del mismo:

Preprocesamiento de la información, adecuación de los datos y obtención de parámetros de interés

- Preprocesado de la información inicial para limpiarla, estructurarla y transferirla a la siguiente fase de adecuación y filtrado.
- Adecuación y filtrado de la base de datos para poder entrenar a los modelos desarrollados.
- Estudio de los parámetros más relevantes en la predicción de los niveles de funcionalidad final del paciente durante el proceso de rehabilitación.

Generación y Experimentación de modelos predictivos

- Desarrollo de modelos predictivos con *machine learning* utilizando los algoritmos Random Forest, Linear Regression, Support Vector Machine, Multi-Layer Perceptron y Redes Neuronales LSTM (Long-Short Term Memory).
- Experimentación de los modelos y el nivel de predicción utilizando variables estáticas, variables estáticas y dinámicas y teniendo en cuenta la evolución temporal de las variables estáticas y dinámicas a lo largo del tiempo.

3.MATERIALES

Los materiales utilizados para el desarrollo del proyecto han sido una base de datos facilitada por el Hospital Vithas Valencia al Mar y de su grupo de neurorrehabilitación y el software para el desarrollo del sistema de predicción.

La base de datos consta de un total de 2400 registros, 340 variables y 800 pacientes del histórico de datos clínicos recogidos en la evaluación de los pacientes. A esta base de datos se le ha realizado un primer filtrado utilizando la variable "Alta" seleccionando solamente a los pacientes que hayan sido dados de alta, quedando un total de 2026 registros y 667 pacientes. Para los datos de los pacientes se ha realizado una partición aleatoria de los datos por cada paciente, asignando individualmente las diferentes categorías "Entrenamiento", "Test" y "Validación" y los porcentajes indicados en la tabla 4:

Dato	**Número de pacientes**
Datos completos	**667**
Entrenamiento (65%)	**433**
Validación (15%)	**100**
Test (20%)	**134**

Tabla 4. Partición de datos.

De la base de datos proporcionada se han reducido el número de variables a utilizar a un total de 27, debido a que las demás variables no tenían suficientes muestras o no aportaban mucha información. Estas variables se han clasificado en estáticas y dinámicas diferenciando entre categóricas, numéricas y FIM/FAM. Las últimas variables FIM/FAM son numéricas pero se diferencian ya que se debe interpretar de manera diferentes en el sistema, debido a que se tiene que saber el valor máximo de cada una para poder evitar valores incorrectos en el procesamiento de los datos.

Dinámicas

- Categóricas: Clasificación neurológica, Tiene control, Cuidados y necesidades, GOS.
- Numéricas: Coma días, Años escolaridad, Edad ingreso.
- FIM: FIM cuidados personales, FIM movilidad, FIM Locomoción, FIM conciencia del mundo exterior, FIM comunicación, FIM cognitiva, FIM control de esfínteres, FIM *mobility*, FIM ADL, FIM *sphincter*. FIM *executive.*
- FAM: FAM total (utilizando las valoraciones anteriores a la que se intenta predecir)

Estáticas

- Categóricas: Sexo, Nivel de estudios, GCS pronóstico, Coma pronóstico, APT pronóstico
- Numéricas: IB total, DOS total, DRs

Para el desarrollo del sistema de predicción se ha utilizado el software ®Matlab R2020a y sus *toolbox Neural net fitting*, *Neural network toolbox,* y *Paralel computing toolbox*,

4. MÉTODOS

4.1. Preprocesamiento de los datos.

La parte de preprocesamiento de la información es fundamental para poder proceder a una correcta adecuación de los datos y posteriormente para entrenar al sistema, ya que obtenemos un filtrado y ordenamiento inicial de la base de datos, el vector de partición de los datos que indica con qué tipo de dato (entrenamiento, validación o test) hemos marcado a cada paciente y finalmente obtenemos un vector idpaciente que no venía reflejado en la base de datos facilitada. Mantener un registro de los idpaciente en las bases de datos es casi necesario para poder realizar un tratamiento correcto de estos datos, ya que de no tenerlo y basarse en otras variables para realizar un seguimiento de cada paciente muchos registros de la base de datos pueden contener información incorrecta y aportaríamos ruido al sistema que puede afectar al rendimiento de los modelos de *machine learning* entrenados.

Los pasos seguidos para el correcto preprocesamiento de los datos se detallan a continuación:

Lectura y partición de los datos

- Leemos la base de datos seleccionada por el usuario y filtramos los registros por pacientes dados de alta.
- Se realiza un segundo filtrado más exhaustivo de la información, ordenamos los registros de los datos y obtenemos la información de los idpaciente en la función Orden_BBDD, que es explicada más adelante.
- Ejecutamos la partición de los datos y registramos cada idpaciente con los tipos de muestra con los porcentajes indicamos en la tabla X. Indicar que estos porcentajes han sido utilizados en este proyecto por ser habituales en sistemas de *machine learning*, pero el usuario que utilice la herramienta puede indicar otros ya que son parametrizables.

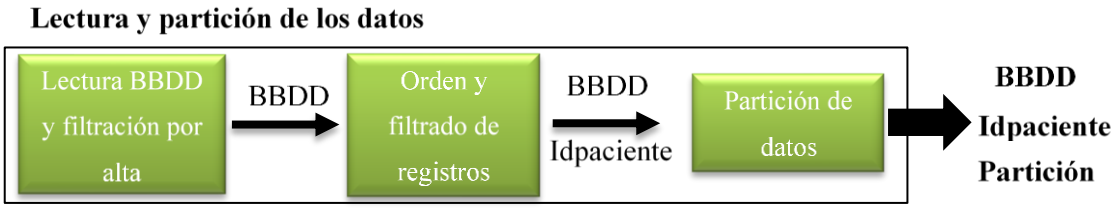


Figura 3. Lectura y partición de datos.

Ordenamiento de registros, filtrado y obtención de idpaciente

- Obtención del vector idpaciente en todos los registros de la base de datos, utilizando un conjunto de variables para identificar a los pacientes y confirmar su identificación. Este vector idpaciente contiene una numeración individual asignada a cada registro del paciente, empezando desde el primer paciente de la tabla que tendrá el idpaciente uno.

- Se realiza un primer filtrado de los pacientes que solamente tengan un registro en la base de datos. Esto significa que solo tendrán una valoración y por lo tanto no se podrá realizar un seguimiento de la evolución de este paciente.

- Se comprueban los registros de información que se repiten y se eliminan dejando el original, ya que se puede encontrar pacientes con la misma valoración repetidas veces y estos datos no serían correctos para entrenar el sistema.

- Se vuelve reorganizar la variable idpaciente, teniendo en cuenta los registros eliminados.

- Se calcula el número de valoraciones máximas de cada paciente y se comprueba que concuerda con el número de registros por idpaciente almacenando los idpacientes que no coincidan.

- Se eliminan los registros incorrectos en la BBDD a partir de la información detectada anteriormente: Se buscan los registros de los pacientes detectados, comprobando las valoraciones que se repiten y se establece una serie de criterios a partir de la fecha de valoración de cada registro para seleccionar los registros incorrectos y eliminarlos.

- Se ordena de nuevo la información de idpaciente, teniendo en cuenta los registros eliminados.

- Se ordena la base de datos a partir de la información de idpaciente y los datos de la variable valoración: Se necesita tener los registros ordenados en función de los idpaciente y el número de valoración, ya que se ha detectado en diversas ocasiones registros incorrectos y es necesario para poder realizar un acondicionamiento correcto de los datos en la fase de adecuación.

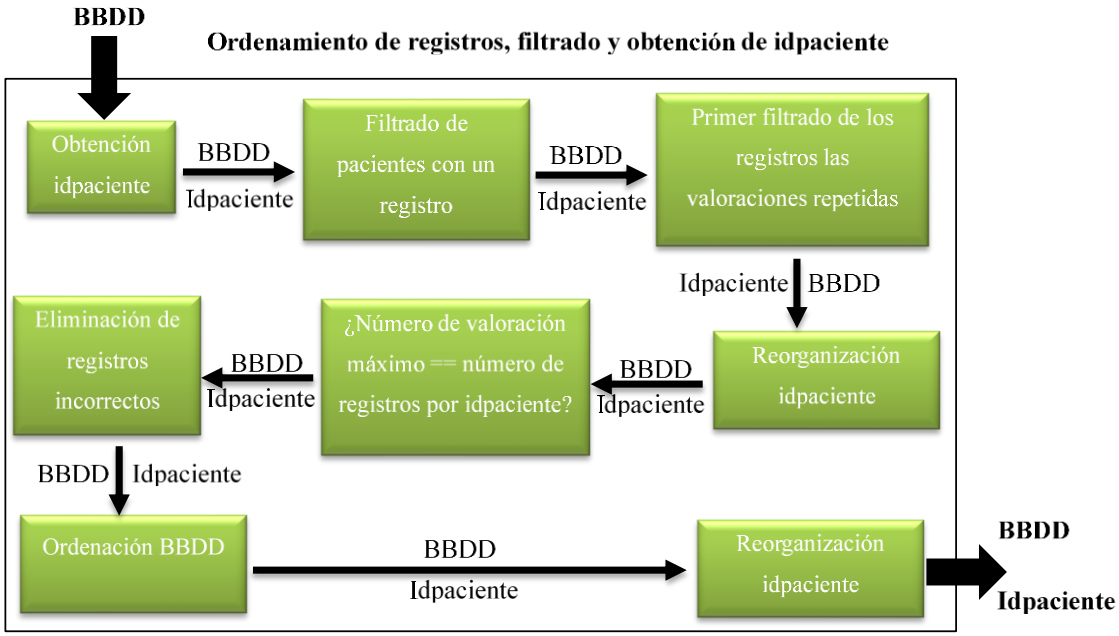


Figura 4. Ordenamiento, filtrado y obtención idpaciente.

4.2. Adecuación de los datos para entrenar a los modelos.

Después de realizar el preprocesamiento de los datos viene la fase de adecuación de la información, en la cual se realiza un filtrado seleccionando el número de valoraciones y las variables a emplear para entrenar al modelo de *machine learning*. Se creará una base de datos auxiliar con las variables de interés que servirá para ir realizando filtrados y reestructuraciones cada vez que se detectan registros incorrectos y estos se eliminen. Finalmente se creará la base de datos a utilizar en el sistema haciendo uso de la auxiliar creada anteriormente, para realizar el último filtrado de los registros con valores NaN (Estos valores se producen al no incorporar en la base de datos la información correcta, cuando se procesa esta base de datos en el sistema se generar estos errores para esa información) y los registros de las variables numéricas con valores atípicos que superen una desviación indicada por el usuario(es un parámetro de entrada al sistema) del valor medio de la variable. Esta parte es igual de relevante que la de preprocesamiento de la información, debido a que se necesita disponer de unos datos correctos para entregárselos a los modelos de *machine learning* para realizar unas predicciones más precisas. La base de datos al tener tanta información puede contener muchos registros incorrectos y el tratamiento de esta información es muy delicado, por lo que la definición de una buena arquitectura para adecuar la información es clave para evitar errores en la predicción o en el propio sistema cuando procesa los datos.

El resultado final de la adecuación de la información es una base de datos con las variables de interés, filtrado y estructura correspondiente.

Indicar que en la adecuación se contemplan dos escenarios: el uso únicamente de la primera y última valoración, o el uso de la evolución temporal, es decir utilizar más valoraciones disponibles. Esto es debido a que se pretende realizar diferentes estudios utilizando unos modelos de machine learning u otros según estos escenarios y la estructura de la información que se entrega a los modelos es distinta. Por ello existen dos arquitecturas definidas según el caso que se esté realizando, estas se explican a continuación:

Arquitectura de adecuación sin evolución temporal

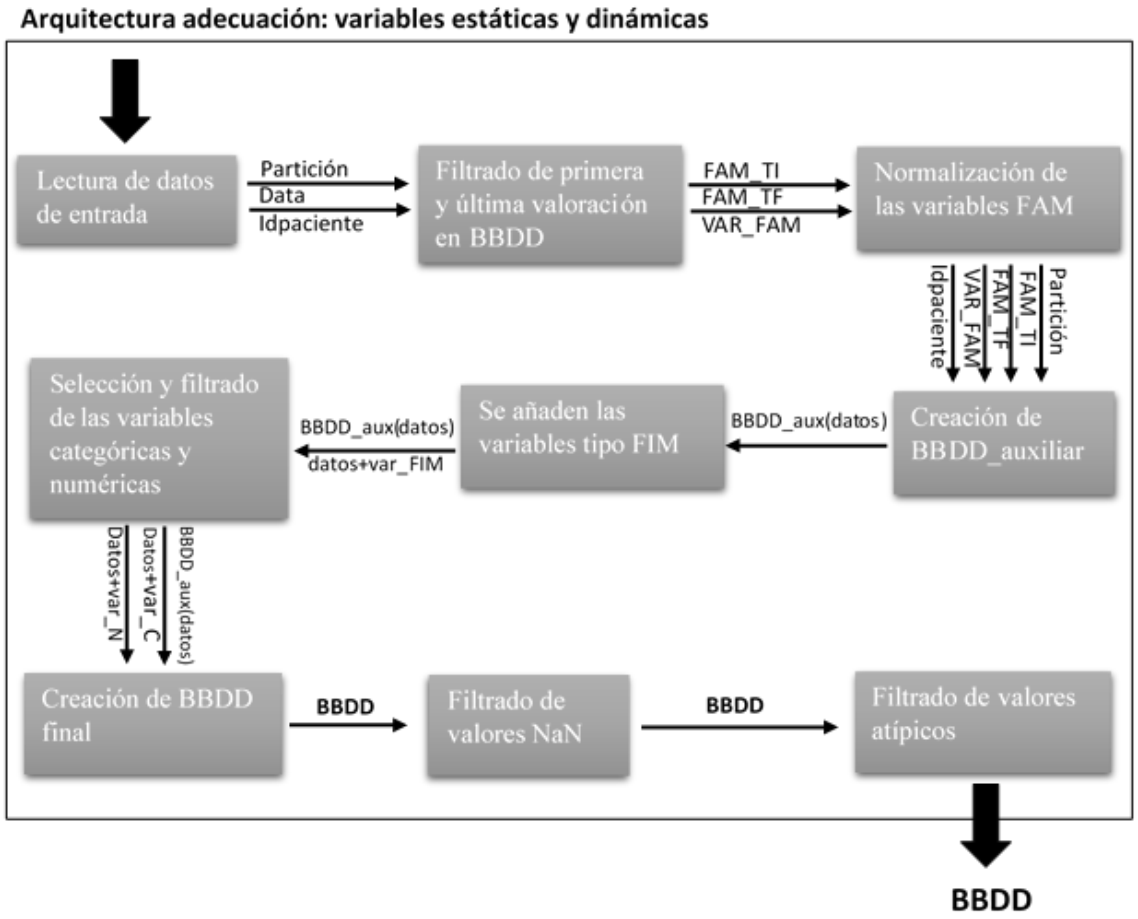


Figura 5. Arquitectura adecuación sin evolución temporal.

- Lectura de datos de entrada: Se realiza una lectura de la base de datos, la información de los identificadores de pacientes y la partición de los datos.

- Filtrado de valoraciones:

    - Se realiza una selección de los registros almacenados con la primera valoración y se estructuran las variables.

    - Se realiza la misma acción para los registros con la última valoración y se obtiene el vector de valoraciones finales.

    - Posteriormente se obtiene los valores de las variaciones de la variable FAM: var_FAM=FAM_total_inicial – FAM_total_final

- Se normalizan los valores de las variables FAM: FAM inicial, FAM final y variación del FAM. FAM inicial contendrá los valores de la primera valoración de cada paciente, FAM final los valores de la última valoración y la variación del FAM será la diferencia entre ambos. Debido a esto tendremos estructurada la base de datos final con un registro de información por paciente.

- Creación de la base de datos auxiliar con las variables obtenidas

- Obtención y filtrado de las variables adicionales tipo FIM:

    - Utilizando la base de datos se obtienen los valores de las variables tipo FIM, que se han seleccionado para entrenar al sistema.

    - Se realiza un filtrado comprobando que los registros de las nuevas variables sean correctos, confirmando que no hay valores NaN o valores superiores al máximo de cada variable tipo FIM.

- Selección y filtrado de las variables categóricas y numéricas.

    - Se obtienen las variables categóricas y numéricas y se guardan los valores de los registros incorrectos (NaN, registros vacíos, valores categóricos con pocas muestras).

    - Filtrado de las variables categóricas: Haciendo uso de los registros guardados detectados como incorrectos, se realiza un filtrado de las variables categóricas eliminando los registros o modificándolos a las categorías más comunes del paciente.

    - Se reestructura los valores de identificadores de los pacientes(idpaciente)

    - Se realiza una conversión One-Hot-Encoding de las variables categóricas: Se traducen las variables categóricas en tablas con valores numéricos, para cada variable se crea una tabla con tantas columnas como resultados posibles tenga la variable, se asigna el valor 1 en la columna correspondiente al resultado en ese registro y 0 a las demás columnas (Ver figura 6).

    - Se remodela la base de datos auxiliar con las nuevas variables introducidas

- Creación de la base de datos final: Se diseña la base de datos con las variables para entrenar al sistema filtradas y estructuradas, con un registro por paciente. Esta base de datos contendrá las siguientes variables:

    - Idpaciente: Permite distinguir los datos por cada paciente, aunque cada registro será un paciente independiente.

    - FAM final: Contendrá los valores del nivel de funcionalidad final que ha logrado cada paciente, servirá para entrenar a los modelos como variable a predecir.

    - Partición: Indica el tipo de dato que hemos asociado a cada registro (entrenamiento, validación o test).

    - FAM inicial: Esta variable no se añadirá en caso de realizar el estudio solamente con variables estáticas.

    - Variación del FAM: Esta variable no se incorporará en la base de datos si se está realizando el estudio solo con variables estáticas. Servirá para entrenar a los modelos como variable a predecir si se selecciona.

    - Variables adicionales tipo FIM: El conjunto de estas variables no se añadirá en el caso de estar realizando el estudio solo con variables estáticas o de haber indicado en los parámetros de entrada que no queremos añadir variables tipo FIM en la base de datos final para entrenar el sistema. Estas variables serían las que se han comentado en el punto 1.1.2 ***FIM y FAM: valoraciones de funcionalidad.***

    - Variables adicionales numéricas: El conjunto de variables adicionales numéricas puede contener tanto variables estáticas como dinámicas, por lo tanto, se incorporarán ambos grupos de variables según el tipo de estudio que estemos realizando. Algunos ejemplos de variables numéricas estáticas serían: Edad de ingreso, años de escolaridad, días en coma. En el caso de las variables numéricas dinámicas: DRs, DOS neurológico, DOS conducta, etc.

    - Variables adicionales categóricas: Del mismo modo que las variables adicionales numéricas la agrupación de las categóricas puede incluir variables estáticas y dinámicas, por consiguiente, se incluirán los dos grupos de variables en función del tipo de estudio que se desee realizar. Algunos ejemplos de variables categóricas estáticas serían: Sexo, Estado civil, Consumo de drogas. Otros ejemplos de variables dinámicas: Clasificación neurológica, GOS, Cuidados y necesidades, etc.

- Depuración de los valores NaN: Último filtrado de la base de datos creada para eliminar o sustituir los valores NaN encontrados.

- Filtración de los valores atípicos detectados: Se analizan las variables numéricas y se eliminan los registros de los valores que superen una desviación establecida respecto del valor medio de la variable que se analice.

Label Encoding

Food Name	Categorical #	Calories
Apple	1	95
Chicken	2	231
Broccoli	3	50

→

One Hot Encoding

Apple	Chicken	Broccoli	Calories
1	0	0	95
0	1	0	231
0	0	1	50

Figura 6. One Hot Encoding. Obtenido de [20].

Arquitectura para adecuación con evolución temporal

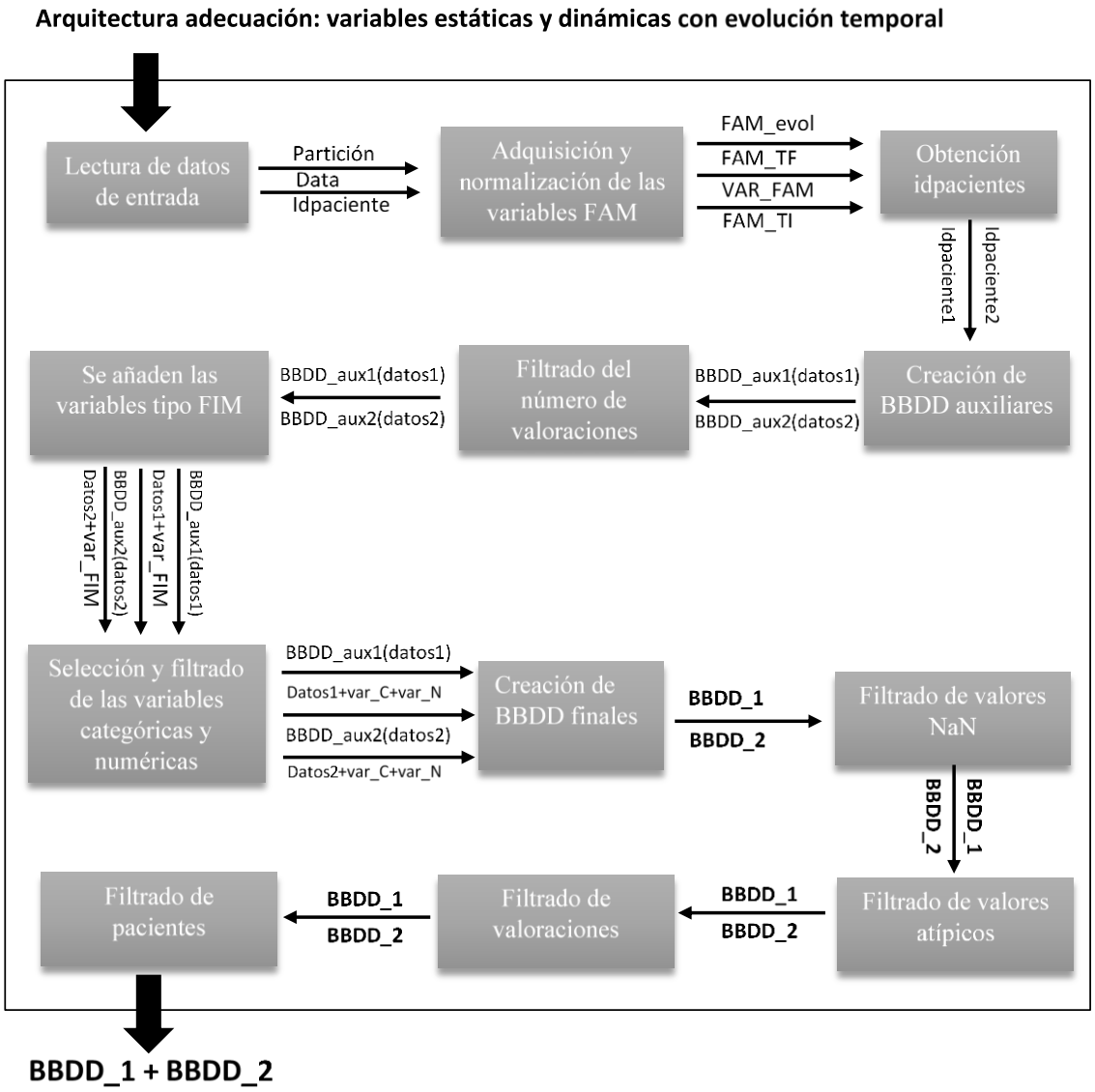


Figura 7. Arquitectura adecuación con evolución temporal.

La arquitectura que sigue esta adecuación de la información es similar a la comentada en el punto anterior, salvo que en esta se añaden los módulos de obtención de idpacientes, filtrado de valoraciones y filtrado de pacientes, además de algunas modificaciones en las partes de filtrado del número de valoraciones, adquisición y normalización de las variables FAM, creación de BBDD auxiliares y creación de BBDD finales. Todos estos componentes están explicados en los siguientes párrafos:

- Adquisición y normalización de las variables FAM: Evolución del FAM, FAM inicial, FAM final y variación del FAM. La evolución del FAM contiene todos los valores de las valoraciones de cada paciente menos el valor final, ya que este se almacena en FAM final. FAM inicial contiene solamente la primera valoración por cada paciente y la variación del FAM la diferencia entre FAM inicial y FAM final.

- Obtención de los vectores idpaciente para las bases de datos auxiliares: Como la información de los valores de la evolución del FAM generalmente no va a tener los mismos registros que las otras variables FAM inicial, FAM final, variación del FAM, se necesita tener esta información en dos bases de datos distintas, por eso se debe obtener los identificadores de los pacientes para poder trabajar correctamente con los datos en ambas tablas.

- Creación de las bases de datos auxiliares: Se añaden dos bases de datos con las variables de utilidad, utilizando los identificadores de los pacientes obtenidos en el paso anterior.

- Filtrado del número de valoraciones: Se realiza un primer filtrado del número de valoraciones máximas que se indica como parámetro de entrada. De esta forma, más adelante se podrá evaluar la precisión que tiene el sistema, en base al número de valoraciones que ha utilizado para realizar la predicción.

- Creación de las bases de datos finales: Se añaden ambas bases de información con las variables para entrenar al sistema filtradas y estructuradas. La base de datos que se utilizará para entrenar al sistema, con la evolución de los valores de los niveles de funcionalidad FAM tendrá uno o más registros por paciente, mientras que la otra base de datos que contenga las variables FAM inicial, FAM final y variación del FAM solamente tendrá un registro por paciente. Indicar que en esta última base de datos las variables adicionales que se añaden son exclusivamente para obtener gráficas estadísticas en relación con las variables a predecir y tomar decisiones en base a visualizar estos resultados. Las bases de datos contendrán las siguientes variables:

  BBDD_1

  - Idpaciente: Los valores de esta variable sirven para identificar que registros corresponden a cada paciente. En esta base de datos se tendrá uno o más registros por paciente.

  - Evolución del FAM(FAM_evol): Contiene los valores de los niveles de funcionalidad evaluados, exceptuando la última, ya que este dato estará en FAM final y es la variable que se está intentando predecir, junto con la variación del FAM.

  - Partición: Identifica el tipo de dato de cada registro (entrenamiento, validación o test).

- Valoración: Contiene el número de la valoración que se ha realizado sobre el paciente para seguir la evolución.

- Variables seleccionadas tipo FIM: Contiene las variables seleccionadas tipo FIM comentadas en el punto 1.1.2 ***FIM y FAM: valoraciones de funcionalidad*** de la memoria.

- Variables adicionales numéricas: El conjunto de estas variables podrían ser tanto estáticas como dinámicas.

- Variables adicionales categóricas: El conjunto de estas variables podrían ser tanto estáticas como dinámicas.

BBDD_2

- Idpaciente: Los valores de esta variable sirven para identificar que registros corresponden a cada paciente. En esta base de datos se solamente un registro por paciente.

- FAM inicial: Contiene los valores de los niveles de funcionalidad tomados en la primera valoración del paciente.

- FAM final: Contiene los valores finales de los niveles de funcionalidad en la última valoración del paciente. Será una variable para predecir si es seleccionada como tal.

- Variación del FAM: Incluye la variación de los niveles de funcionalidad entre la primera y última valoración del paciente. Al igual que el FAM final si se indica también puede ser una variable para predecir si es seleccionada para ello.

- Partición: Indica la distribución de datos asignada para cada registro (entrenamiento, validación o test)

- Variables adicionales tipo FIM: Contiene las variables seleccionadas tipo FIM comentadas en el punto 1.2 ***FIM y FAM: valoraciones de funcionalidad*** de la memoria.

- Variables adicionales numéricas: La agrupación de estas variables podrían ser tanto estáticas como dinámicas.

- Variables adicionales categóricas: La agrupación de estas variables podrían ser tanto estáticas como dinámicas.

- Filtrado de valoraciones: Se eliminan los registros que contengan valoraciones incorrectas.

- Filtrado de pacientes: Se eliminan los registros de pacientes que no estén en ambas bases de datos, debido a los filtrados que se han realizado en los pasos anteriores y se reinician los contadores de los identificadores de cada paciente.

4.3. Modelos de *machine learnring*.

Posteriormente a una correcta adecuación de los datos se lleva a cabo a la fase de entrenamiento de los modelos de *machine learning*. Se han realizado varios estudios utilizando variables estáticas, dinámicas y variables estáticas y dinámicas teniendo en cuenta su evolución temporal, con un número concreto de valoraciones. En estos estudios se han empleado diferentes modelos de *machine learning* para comprobar los niveles de predicción de la variable objetivo (FAM final) de cada uno de ellos, de esta forma se podrá valorar en cada caso que modelo obtiene unos mejores resultados. Los algoritmos utilizados se explican a continuación:

4.3.1 Random Forest

Random Forest es una técnica muy utilizada en el *machine learning* basado en árboles de decisión combinados con bootstrapping. Utiliza múltiples clasificadores de árboles aleatorios para obtener una clasificación genérica del conjunto de datos de entrada. Generalmente cada rama del árbol tiene el mismo peso para realizar los cálculos. En el sistema diseñado este modelo se utiliza con los datos obtenidos en la primera arquitectura con las variables estáticas y dinámicas [3].

El desarrollo de estos árboles se realiza de la siguiente manera:

- El conjunto de datos se forma mediante un muestreo utilizando la técnica bootstrapping. Al utilizar esta técnica entregamos a distintos árboles diferentes muestras de datos, esto permite que cada árbol procese distintos datos en el entrenamiento y que ningún árbol disponga de todo el conjunto de datos. De esta forma se combinan los modelos ruidosos obtenidos, reduciendo la variación y obteniendo una mejor predicción generalizada.
- El número de muestras en el conjunto de datos es igual al del conjunto de datos de entrenamiento, donde en esta nueva agrupación de datos se puede contener ejemplos duplicados del conjunto de entrenamiento. Usando la técnica de *bootstrapping*, generalmente un tercio de los datos de entrenamiento no está presente en la información entregada a cada árbol.
- Se elige un número aleatorio de atributos para cada árbol. Estos atributos forman los nodos y hojas (terminaciones de los nodos) utilizando algoritmos de construcción de árboles estándar.
- Cada árbol se crea en la mayor medida posible de que no sea necesario eliminar hojas.

La siguiente imagen sería un ejemplo general del modelo Random Forest:

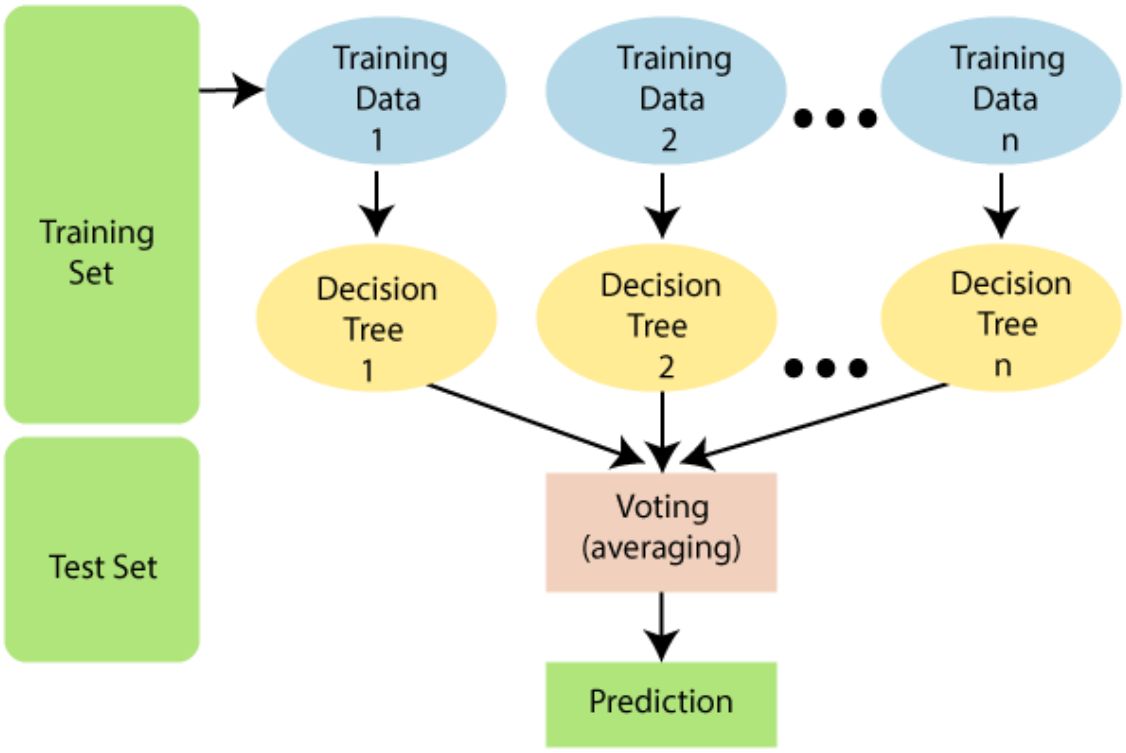


Figura 8. *Random Forest*. Obtenido de [2].

En el programa diseñado se ha tenido en cuenta el uso del parámetro número de árboles y en opciones el tipo de método utilizado para el entrenamiento del modelo:

- Número de árboles: Normalmente cuanto el número de árboles es mayor suele mejorar el nivel de predicción, pero a partir de cierto punto la precisión deja de mejorar y solo hace que el algoritmo vaya ejecutando la información más lentamente. En este caso se han hecho diversas pruebas y se ha comprobado que con valores mayores a 50 para este parámetro la precisión no mejora.

- Método: De las opciones disponibles clasificación o regresión, se ha seleccionado el método de regresión ya que es el problema que se intenta resolver es la estimación de un valor numérico, no una clase.

Ventajas

- Random Forest es capaz de realizar tareas de clasificación y regresión.
- El modelo es más simple de entrenar en comparación otros modelos más complejos, pero con un rendimiento similar.
- Es capaz de procesar y manejar grandes bases de datos eficientemente.
- Puede procesar cientos de predictores sin descartar ninguno y lograr diferenciar cuáles son los más importantes, por ello esta técnica también se utiliza para la reducción de dimensionalidad
- Es capaz de conservar su precisión, aun teniendo un gran cantidad de datos perdidos.

Inconvenientes

- La visualización gráfica de los resultados puede ser difícil de interpretar.
- Puede tener cierto sobreajuste en algunos grupos de datos con existencia de ruido.
- Actualmente se tiene poco control sobre lo que hace el modelo (en cierto sentido es como una caja negra).

Ventajas e inconvenientes información obtenida de las fuentes [4].

4.3.2 Linear Regression

La regresión lineal es un algoritmo que se utiliza en el *machine learning* y en estadística. Su misión es realizar una aproximación para modelar la relación entre una variable escalar dependiente $\boldsymbol{Y}$ y una o más variables independientes definidas como $\boldsymbol{X}$. En el sistema diseñado, este modelo se utiliza con los datos obtenidos en la primera arquitectura con las variables estáticas y dinámicas [17].

Se pueden realizar regresiones simples como en el siguiente ejemplo:

$$y = mx + b$$

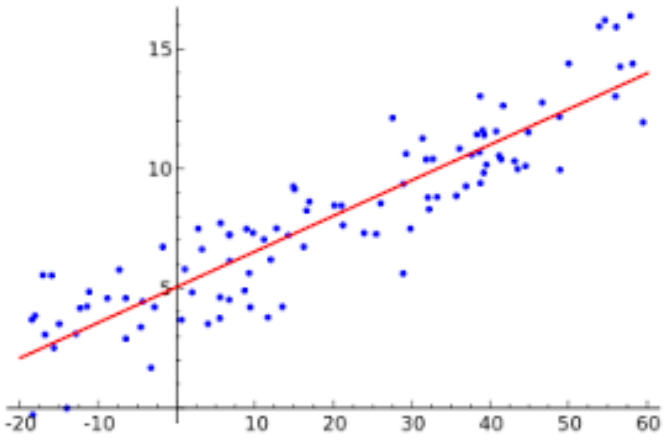


Figura 9. Linear Regression. Obtenido de [17].

O también es posible realizar regresiones lineales múltiples como en la siguiente imagen:

$$y = m_1 x_1 + m_2 x_2 + b$$

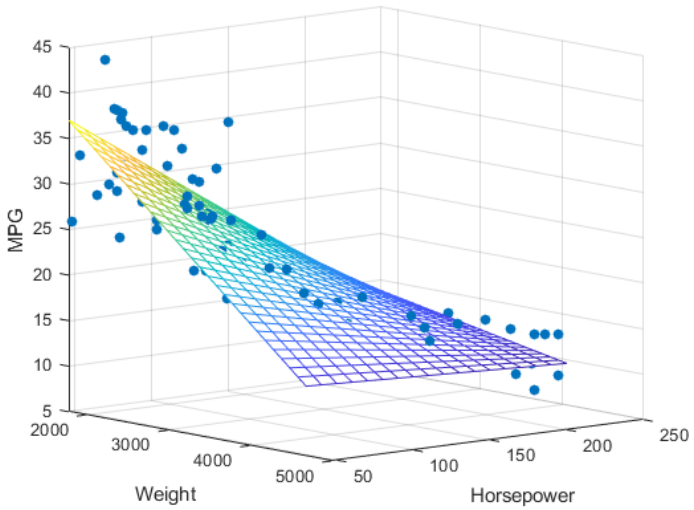


Figura 10. Multiple Linear Regression. Obtenido de [5].

4.3.3 Support Vector Machine

Las máquinas de vectores de soporte son un conjunto de algoritmos de aprendizaje supervisado desarrollador por Vladimir Vapnik, que constituyen un método para la resolución de problemas de clasificación y regresión. En este trabajo se ha utilizado la primera arquitectura definida para

adecuar los datos con variables estáticas y dinámicas, para preparar los datos y entrenar a este modelo [8].

Para problemas de clasificación este algoritmo funciona como un clasificador discriminatorio, definido por un hiperplano de separación. A partir de los datos de entrenamiento etiquetados el algoritmo crea un hiperplano óptimo que permite predecir una clasificación de los nuevos datos en dos espacios dimensionales. En el ejemplo siguiente se genera una línea que separa el plano en dos partes correspondientes a las clases.

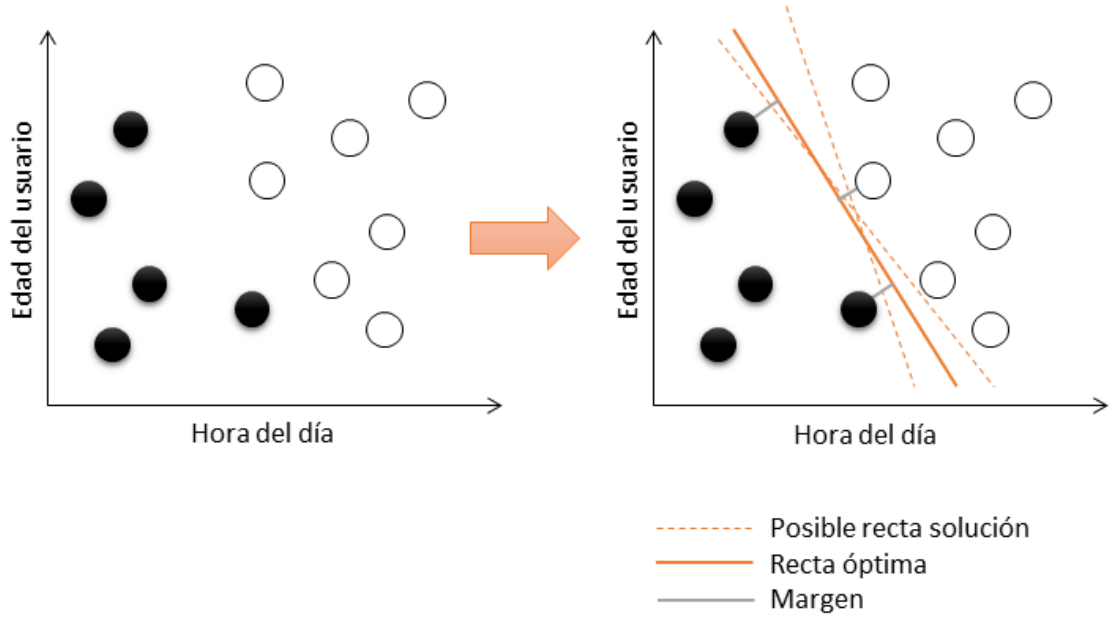


Figura 11. SVM Classification. Obtenido de [8].

En la imagen anterior se está resolviendo un problema bidimensional, pero si hubiera tres dimensiones esta recta de separación sería un plano y en el caso de más dimensiones un hiperplano con las dimensiones adecuadas.

En el caso de que el problema no sea lineal, este algoritmo utiliza las funciones Kernel (no lineales). Estas funciones toman un espacio de baja dimensión y lo transforman a un espacio dimensional más alto, convirtiendo un problema no separable en uno separable, es decir se

traslada los datos a un espacio donde el hiperplano solución es lineal y más sencillo de obtener, para posteriormente transformar la solución al espacio original [8].

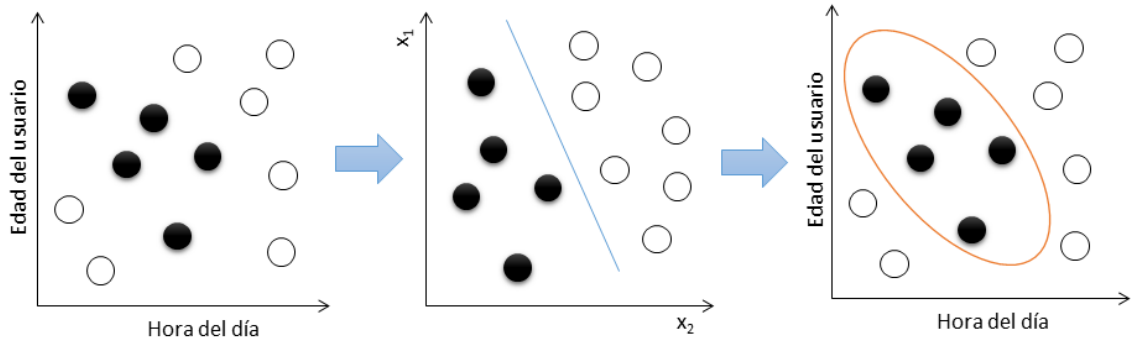


Figura 12. SVM Kernel. Obtenido de [8].

En el caso de la regresión, el algoritmo se basa en buscar la curva del hiperplano que modele mejor la tendencia de los datos de entrenamiento en base a obtener mejores predicciones. También se utiliza un margen máximo que indica los valores que están fuera y dentro del rango en la curva, los valores fuera de rango son considerados errores y se necesita calcular la distancia entre el punto de error y el margen, ya que estas distancias se utilizan en la ecuación del modelo. Este es el tipo de algoritmo que se ha utilizado en este proyecto, en concreto los datos que se han entregado al algoritmo han sido acondicionados utilizando la arquitectura de adecuación de los datos para las variables estáticas y dinámicas [8].

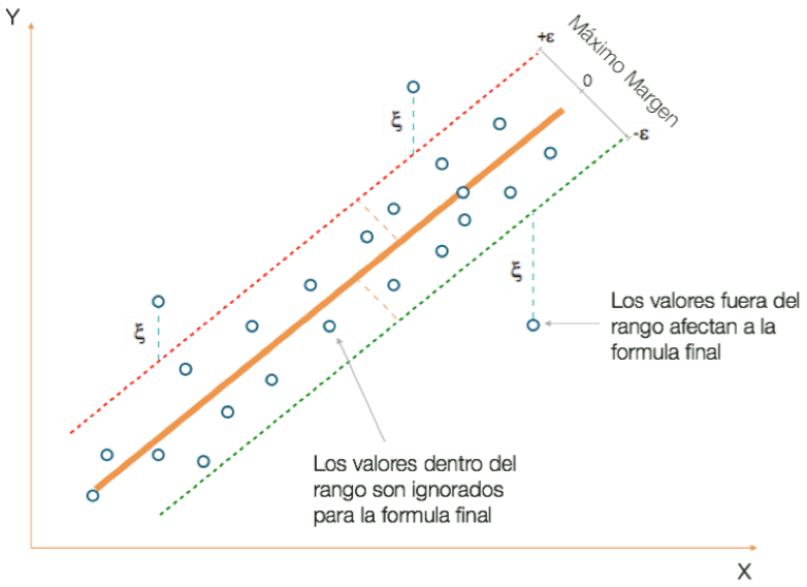


Figura 13. SVM Regression. Obtenido de [8].

De igual forma que en los problemas de clasificación, en problemas no lineales es posible utilizar las funciones Kernel para obtener la curva que mejor modele los datos.

4.3.4 Multi Layer Perceptron

Un perceptrón es un algoritmo simple de clasificación binaria, creado por Frank Rosenblatt. A diferencia de otros algoritmos de clasificación este se diseñó a partir de la unidad esencial del cerebro, la neurona. A partir de un conjunto de señales de entrada, cada una de las cuales tiene asociada un peso, genera una respuesta binaria “0” o “1” [9].

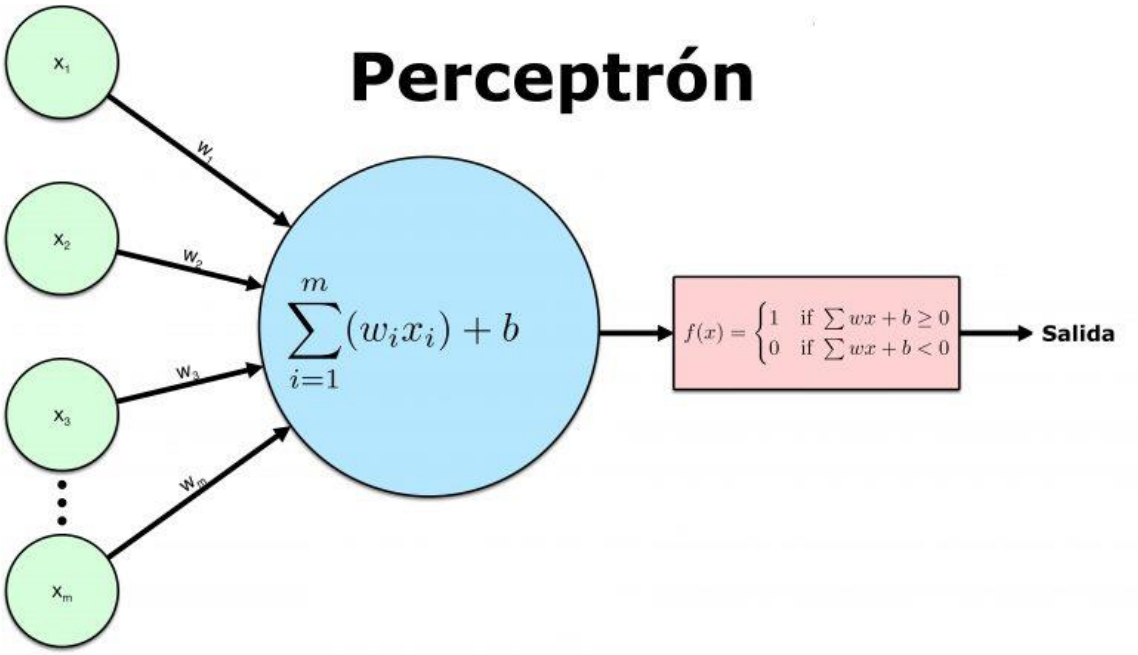


Figura 14. Perceptrón. Obtenido de [11].

Para obtener los datos de salida un perceptrón sigue los siguientes pasos:

1. Procesa los datos de entrada, las multiplica por los pesos asignados a cada señal y realiza el cálculo de la suma total de estos valores. Al utilizar estas ponderaciones en cada señal se permite estimar la importancia relativa de cada una de las salidas.
2. Agregamos al sumatorio total anterior el factor sesgo ***b,*** esto es el valor 1 multiplicado por el peso asignado a la señal. Esto paso ayuda ajustar con precisión la salida numérica del perceptrón.
3. Se recibe el valor resultado del cálculo anterior en la función de activación y esta emite una salida binaria según los valores anteriores.

Sin embargo, cuando se realiza una combinación con muchos otros perceptrones, se forma una red neuronal artificial. De esto es lo que se trata el perceptrón multicapa, la unión de perceptrones apilados en varias capas creando una red de dimensiones mayores que, con los suficientes datos de entrenamiento, es capaz de dar respuesta a problemas altamente complejos. Para entrenar el modelo del perceptrón multicapa se ha hecho con los datos adecuados a partir de la arquitectura definida para variables estáticas y dinámicas.

En la siguiente imagen se muestra el esquema de un perceptrón multicapa con 3 capas. Cada perceptrón de la capa de entrada envía información a todos los perceptrones de la segunda capa (capa oculta), de igual forma todos los perceptrones de la segunda capa envían las salidas a la capa final. Cada una de estas señales que envía un perceptrón a los demás perceptrones de la capa siguiente va asociado a un peso [9].

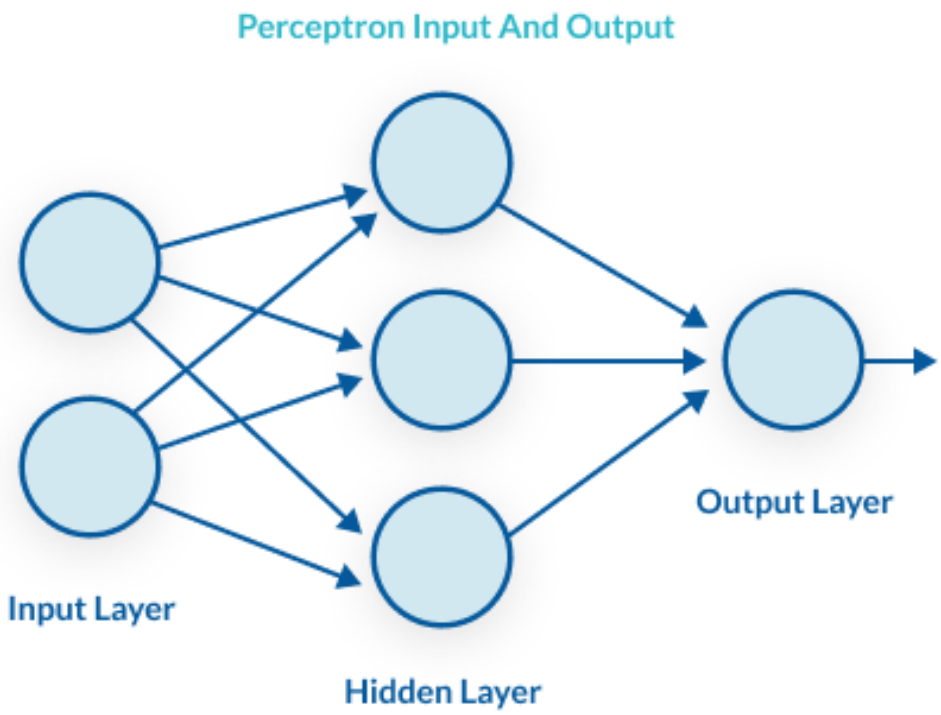


Figura 15. Multi Layer Perceptron. Obtenido de [9].

En los algoritmos que componen un sistema de *Deep Learning* se encuentran diferentes capas neuronales compuestas por pesos, principalmente divididos en tres capas:

- **Capa de entrada (Input Layer):** Compuesto por redes neuronales que recogen los datos de entrada, pudiendo ser una imagen o una tabla de datos.
- **Capa oculta (Hidden Layer):** En esta parte de la red se realiza el procesamiento de la información y los cálculos para obtener información de salida que se envía a la última capa. Cuantas más neuronas hay en la capa, más complejos son los cálculos que se realizan.
- **Capa de salida (Output Layer):** Es la última capa de la red que interpreta la información obtenida por la capa oculta y toma decisiones en base a obtener un resultado de salida.

La red puede estar conformada por múltiples capas con una gran cantidad de perceptrones, por lo que el sistema puede adquirir rápidamente una gran complejidad. En el caso de la red neuronal anterior se denomina también red neuronal poco profunda, ya que hace uso de tres capas solamente. Para redes que utilicen un mayor número de capas se las conoce como redes neuronales profundas, estas redes contienen en la capa oculta un número de capas igual o superior a dos.

Una diferencia entre un perceptrón multicapa y una red neuronal es qué en el perceptrón tradicional, la función de activación es una función escalonada que da lugar a una salida binaria. En las redes que utilizan perceptrones multicapa pueden utilizar otras funciones de activación que ofrecen otros valores reales de salida, generalmente entre 0 y 1 o entre -1 y 1. Esto posibilita realizar predicciones basadas en probabilidades o resolver problemas de clasificación de elementos [9]. Estas redes neuronales utilizan algoritmos de retropropagación, que permite actualizar los pesos de la red emitiendo señales a los perceptrones de la capa anterior. Ajustando estas ponderaciones ayuda a minimizar las pérdidas y obtener unos resultados más precisos en la predicción final de la red neuronal.

Finalmente, para conformar una red neuronal utilizando perceptrones multicapa hay que tener en cuenta los siguientes aspectos:

- **Arquitectura de la red neuronal:** Se ha de tener en cuenta el número de capas ocultas que se va a utilizar para la construcción de la red.

- **Hiperparámetros**: El ajuste de los hiperparámetros de la red también es un factor importante a tener en cuenta. En este modelo cada vez que se entrenaba la red neuronal

se realizaban diversas simulaciones para la optimización los hiperparámetros en el conjunto de datos de entrenamiento y validación, el modelo resultado era el que aportaba una mejor predicción. En este caso se ha hecho la siguiente selección de estos parámetros, en los que algunos de ellos se han escogido diversos valores para que se realicen diferentes simulaciones determinando el valor más óptimo [18]:

- *Backpropagation algorithm*: Se ha escogido el algoritmo trainlm, que realiza la actualización de los pesos y sesgos teniendo en cuenta la optimización de Levenberg-Maquadt. Es el algoritmo más rápido de retropropagación de los disponibles en Matlab, altamente recomendado como primera elección de algoritmo supervisado. No obstante, requiere más memoria que otros algoritmos.
- Número de épocas(*epochs)*: Número de iteraciones en que se utilizarían las muestras para actualizar los pesos.
- *Batch size*: Este parámetro define el tamaño de las muestras que se propagarán a través de la red para entrenar el modelo en cada iteración.
- *Goal*: Valor objetivo a lograr en la predicción.
- Número máximo de fallos en la validación.
- Valor mínimo objetivo del gradiente.
- Valor inicial de mu: Mu es el parámetro de adaptación que se utiliza en el proceso de optimización de Levenberg-Marquardt al calcular la actualización de los parámetros.
- Factor de disminución de mu.
- Factor de aumento de mu.
- Mu máximo.

4.3.5 Redes Neuronales Long-Short Term Memory

Las redes neuronales LSTM (Long-Short Term Memory) fueron introducidas por Hochreiter & Schmidhuber en el 1997 y son un tipo especial de redes neuronales recurrentes (RNN), estas últimas destacan por integrar bucles de realimentación permitiendo a través de ellos que la información se mantenga durante diversas secuencias de entrenamiento(epochs), recordando estados previos y utilizando esta información para decidir cuál será el siguiente estado. La diferencia entre las redes LSTM y las RNN es que las redes recurrentes básicas pueden modelar relaciones a corto plazo, mientras que las LSTM pueden aprender dependencias largas, ya que tienen una capacidad mayor de memoria a largo plazo. Los datos que se han utilizado para

entrenar el modelo LSTM han sido adecuados con la arquitectura de adecuación para variables estáticas y dinámicas con evolución temporal mostrado en la figura 7 [12].

La siguiente imagen muestra un ejemplo de la estructura de una RNN, donde se aprecia que el módulo central tiene una estructura simple para ejecutar una realimentación de la señal, utilizando solamente una capa *tanh* [12]-[13].

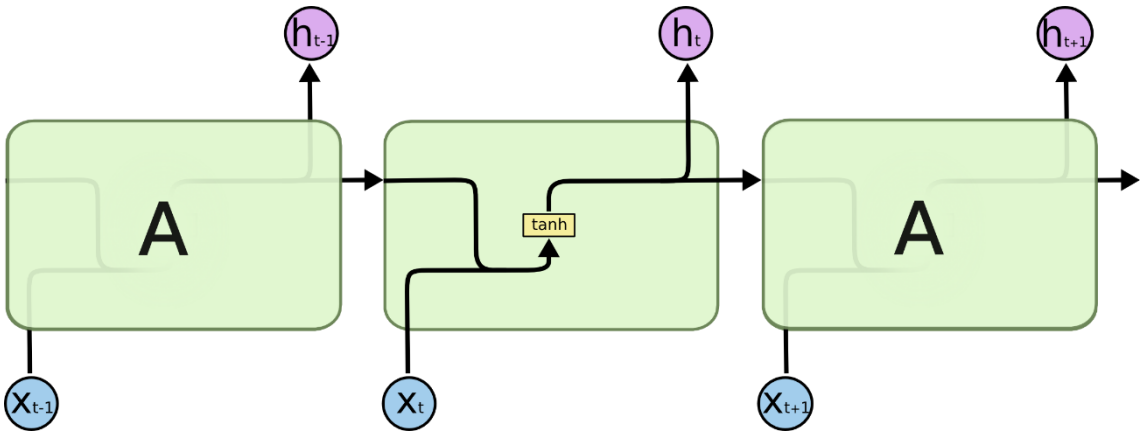


Figura 16. *Recurrent Neural Network*. Obtenido de [12].

En la siguiente figura se muestra la estructura de una red LSTM, en la cual se puede ver que la complejidad del módulo central de realimentación ha aumentado, concretamente se está utilizando 3 capas más adicionales a la RNN, estableciendo una serie de relaciones y operaciones entre ellas [12]-[13].

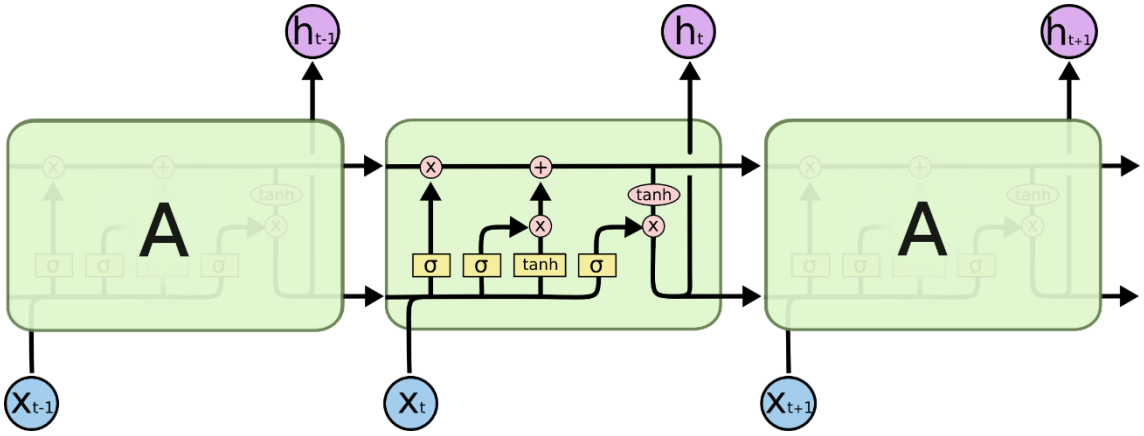


Figura 17. LSTM. Obtenido de [12].

La base de las LSTM son los estados de las celdas, estos se transmiten en la línea superior del módulo estableciendo unas dependencias con algunas de las capas inferiores [11]-[12]. Esto se puede ver en la siguiente figura:

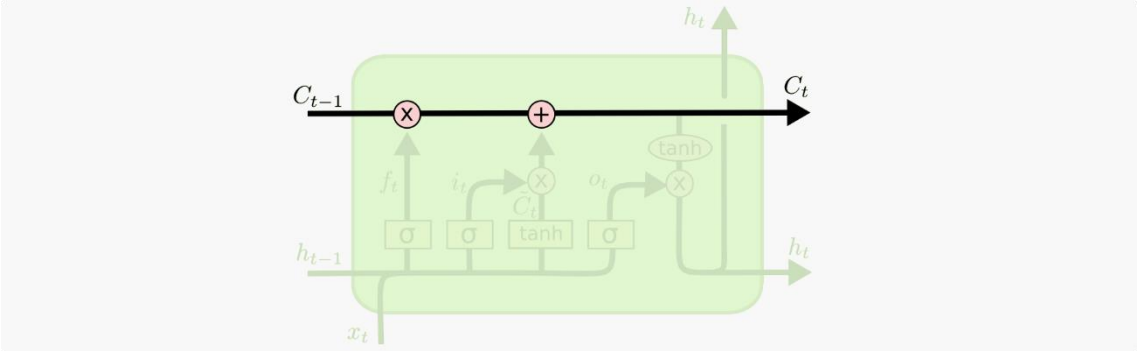


Figura 18. LSTM - Estados de las celdas. Obtenido de [12].

Algunos de los elementos que intervienen en el resultado del estado son las capas sigmoides, que actúan como puertas emitiendo un resultado de salida con valor entre 0 y 1. La redes LSTM utilizan tres puertas para el control del estado de la celda [12]-[13].

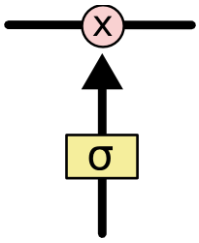
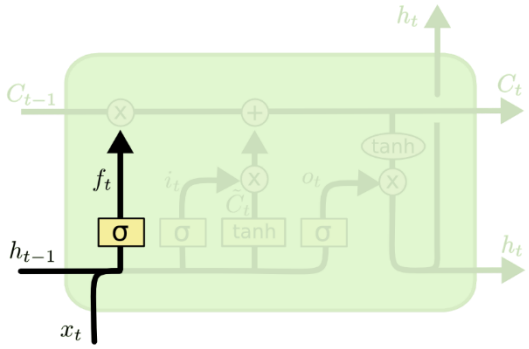


Figura 19. LSTM - Capas sigmoides. Obtenido de [12].

Para que las redes LSTM actualicen el estado de la celda y obtengan el valor de salida se establecen un conjunto de operaciones que podemos dividir en cuatro pasos:

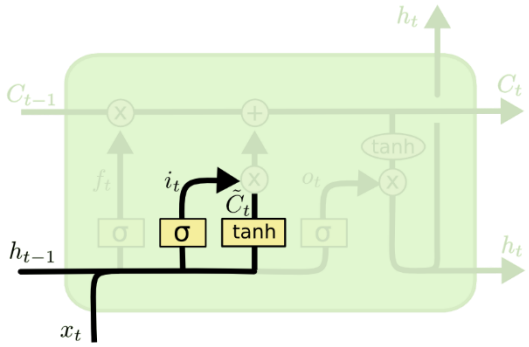
El primer paso consiste en decidir la información a eliminar del estado de la celda. Esta decisión se toma mediante una capa sigmoide llamada "Forget gate layer" (f_t). Haciendo uso de las señales X_t y h_{t-1}, se emite un resultado entre 0 y 1 [12]-[13].



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

Figura 20. LSTM - Forget gate layer. Obtenido en [12].

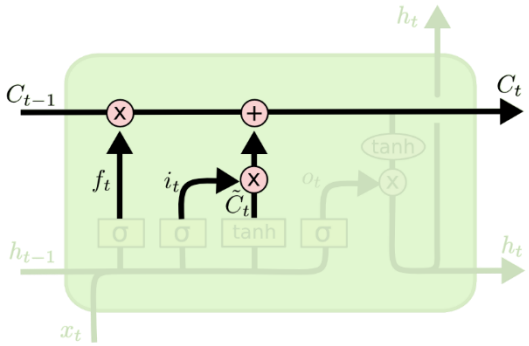
El segundo paso sería decidir la nueva información que se va a almacenar en el estado de la celda. Se compone de dos partes 1) una capa sigmoide llamada capa de puerta de entrada(i_t) que decide los valores a actualizar y 2) la capa *tanh* que genera un vector con los nuevos valores candidatos ($\hat{C}t$) [11]-[12].



$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$

Figura 21. LSTM - Nueva información estado celda. Obtenido en [12].

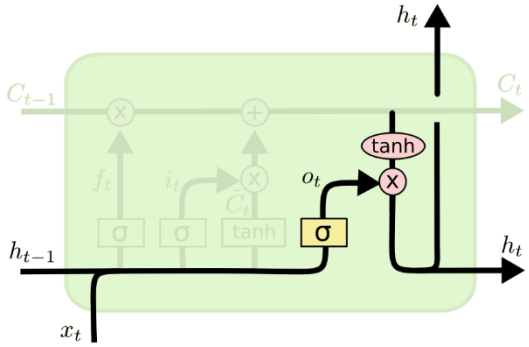
Se actualiza el estado de la celda anterior (C_{t-1}). Esta parte se realiza teniendo en cuenta los valores obtenidos en las etapas anteriores, multiplicando f_t con el estado anterior C_{t-1} y sumando el resultado de la multiplicación de i_t y $\hat{C}t$ [12]-[13].



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Figura 22. LSTM - Actualiza estado de la celda. Obtenido en [12].

Por último, se decide el resultado de la salida generada (h_t): Esta salida dependerá del nuevo estado de la celda (C_t) filtrado por una función *tanh* para que los valores generados estén entre -1 y 1. Este resultado se multiplicara por O_t (el valor de la salida anterior(h_{t-1}) filtrado por una capa sigmoide) [12]-[13] .



$$o_t = \sigma\left(W_o\ [h_{t-1}, x_t]\ +\ b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

Figura 23. LSTM - Salida generada. Obtenido en [12].

Para el entrenamiento de este modelo de *machine learning* se han realizado diferentes simulaciones para escoger los parámetros más óptimos con los que se obtienen mejores predicciones, además del número de capas ocultas de la red neuronal. Los parámetros que se han utilizado para optimizar la red neuronal han sido los siguientes [14]:

- Backpropagation algorithm: Se ha elegido el algoritmo Adam [15], un algorimto de optimización que se ha diseñado específicamente para entrenar redes neuronales profundas. Se puede usar en vez del procedimiento del descenso del gradiente estocástico clásico, para actualizar los pesos de red de forma iterativa en función de los datos de

entrenamiento. Es un algoritmo popular en el mundo del deep learning, porque puede lograr buenos resultados en poco tiempo. Combina las mejores propiedades de los algoritmos AdaGrad y RMSProp para proporcionar un algoritmo de optimización que puede manejar gradientes dispersos en problemas ruidosos, además funciona bien en la mayoría de los problemas utilizando los valores por defecto de los parámetros.

- Entorno de ejecución: Se ha programado para que por defecto se haga uso de la GPU si está disponible para utilizar en el PC, en caso contrario utiliza la CPU para procesar la información.

- Max*epochs*: Máximo número de actualizaciones en los pesos.

- MiniBtachSize: Este parámetro define el tamaño de muestras que se propagarán a través de la red para entrenar al modelo en cada iteración.

- GradientThreshold: Umbral del gradiente. Cuando el gradiente excede el valor de GradientThreshold, el gradiente se recorta de acuerdo con GradientThresholdMethod.

- InitialLearnRate: Tasa de aprendizaje inicial.

- LearnRateDropPeriod: Este parámetro Especifica cuántos intervalos temporales entre cada decrecimiento de la tasa de aprendizaje.

- LearnRateDropFactor: Este parámetro especifica el factor para reducir la tasa de aprendizaje.

- *GradientDecayFactor* : Factor de degradación de gradiente.

- *SquaredGradientDecayFactor* : Factor de decaimiento de gradiente cuadrado.

5. EXPERIMENTOS

5.1. Estrategia experimental y métricas utilizadas.

En este trabajo se ha seguido un conjunto de pasos para conseguir unos objetivos establecidos al inicio del proyecto indicados en el punto ***1.3 Objetivos*** de la memoria. Estos son ***la adecuación de los datos y obtención de los parámetros de interés*** y ***la generación y experimentación de modelos predictivos***. En los siguientes párrafos se tratará de explicar de manera generalizada el esquema global los pasos realizados en el proyecto para conseguir estos propósitos.

Adecuación de los datos y obtención de los parámetros de interés

- Preprocesamiento de la información: En esta tarea se realiza un primer filtrado de los datos, se prepara la información, se organiza en la base de datos obteniendo el vector de los identificadores de los pacientes en los registros y finamente se asigna aleatoriamente a cada paciente una etiqueta (entrenamiento, validación o test) para hacer uso de esta información más adelante cuando la utilicemos en los modelos de *machine learning*. Esta parte esta detallada en el punto ***2.1 Preprocesamiento de los datos.***

- Adecuación de los datos: En esta segunda parte se toma la información obtenida en la tarea anterior, se realiza una adaptación de esta y finalmente se obtienen unas bases de datos con la información necesaria para poder entrenar a nuestros modelos o estudiar los parámetros que más afectan al nivel de predicción final de estos. Esta parte esta detallada en el punto ***2.2 Adecuación de los datos.*** En este proceso de adecuación se siguen diferentes arquitecturas según el estudio que el usuario quiera realizar con el sistema, en el caso de realizar estudios con variables solamente estáticas o variables estáticas y dinámicas se utiliza la arquitectura de la figura 5, para poder utilizar estos datos en los modelos de *machine learning* de ***Random Forest, Linear Regression, Support Vector Machine y Multi-Layer Perceptron***. En el caso de utilizar variables estáticas y dinámicas con su evolución temporal se hace uso de la arquitectura detallada en la figura 7, los datos obtenidos se pasan a celdas LSTM para posteriormente entrenar el modelo ***Long-Short Term Memory.***

- Estudio de los parámetros más relevantes en la predicción de los niveles de funcionalidad: Gracias a la información obtenida en el paso anterior de la adecuación de los datos, se puede realizar un estudio de las variables que se hayan seleccionado previamente gracias a la obtención y almacenamiento de gráficas en carpetas. Las gráficas adquiridas muestran para las variables categóricas en diagramas de caja y las variables numéricas son representadas en gráficos de dispersión. Este estudio está explicado en el punto ***5.2 Estudio de las variables predictivas.***

Generación y Experimentación de modelos predictivos

Una vez se dispone de la información adecuada correctamente los modelos predictivos son entrenados. Según las variables empleadas para el estudio se hará uso de unos modelos u otros para realizar la simulación, con esto se quiere obtener una comparativa de las predicciones que podemos obtener con cada modelo y según el tipo de variables que se utilicen. Cuando se haga uso de variables estáticas y dinámicas para realizar el estudio, solo añadiendo en la base de datos la primera y la última valoración del paciente, se realizarán las simulaciones entrenando los algoritmos de ***Random Forest, Linear Regression, Support Vector Machine y Multi-Layer Perceptron***. En el caso de utilizar variables estáticas y dinámicas con evolución temporal(esto es no utilizar simplemente la primera y la última valoración de cada paciente, si no utilizar también las demás valoraciones disponibles) se utilizará el modelo de inteligencia artificial ***Long-Short Term Memory.*** Además, en este último estudio se realizarán simulaciones filtrando por un número máximo de valoraciones, de manera que se obtendrán resultados de las predicciones realizadas en función del número de valoraciones. Con esto se pretende observar lo eficaz que es el modelo para pronosticar el nivel de funcionalidad final del paciente utilizando el mínimo número de valoraciones posible.

El primer paso es la extracción y clasificación de la información obtenida en el apartado de adecuación de la información según el tipo de dato asignado a cada registro. Estos datos se han clasificado en 6 variables distintas como se puede ver en la tabla 5.

X: Contiene información de las variables que se han seleccionado para entrenar al modelo

Y: Contiene información de la variable a predecir con la última valoración final.

Clase	Variable
Entrenamiento	Xtrain
Entrenamiento	Ytrain
Validación	Xval
Validación	Yval
Test	Xtest
Test	Ytest

Tabla 5. División de daos pre-entrenamiento.

Después de las simulaciones el modelo obtenido será el que tenga unos parámetros de configuración más óptimos y por lo tanto obtenga el modelo de predicción con un error menor.

Entrenamiento

Después de tener clasificada esta información ya podemos realizar el entrenamiento de los modelos utilizando los datos de entrenamiento (Xtrain, Ytrain).

$$Modelo\ obtenido = Modelo\ Machine\ Learning(Xtrain, Ytrain, Configuración\ modelo)$$

Los modelos generados después de las simulaciones son validados en la fase de validación.

Validación

Posteriormente de obtener los modelos en la fase de entrenamiento, son utilizados para realizar predicciones con las muestras de validación y medir el factor de error de estos. El parámetro utilizado para medir los errores es el RMSE (Raíz del error cuadrático medio) y estos cálculos son realizados aplicando las siguientes fórmulas:

$$Ypredecida_{val} = predict(Modelo\ obtenido, X_{val})$$

$$RMSE_{val} = \sqrt{mean((Ypredecida_{val} - Y_{val})^2)}$$

Para cada modelo de *machine learning* se realizan múltiples simulaciones dependiendo del número de valores asignados a los parámetros de configuración, el modelo elegido del conjunto será el que proporcione un menor valor de RMSE en validación y por lo tanto tendrá los parámetros más óptimos.

Una vez se elige cada modelo, se calcula el nivel de correlación de las predicciones para cada uno con los valores de funcionalidad final de la variable en las muestras de validación, utilizando la siguiente expresión:

$$Correlación_{val} = corr(Ypredecida_{val}, Y_{val})$$

Finalmente se calcula una métrica de sobreajuste del modelo, como un parámetro de información adicional, utilizando la siguiente fórmula:

$$Sobreajuste = \frac{RMSE_{train} - RMSE_{val}}{RMSE_{train}}$$

De los modelos de *machine learning* anteriores elegidos que pueden ser ***Random Forest, Linear Regression, Support Vector Machine y Multi-Layer Perceptron*** si se están utilizando variables estáticas y dinámicas o bien solamente ***Long-Short Term Memory*** si se está utilizando variables estáticas y dinámicas con evolución temporal, se elige el modelo que proporciona un menor valor de RMSE en validación y se realizan las pruebas de testeo

Test

La última fase sería la del testeo del modelo elegido, obteniendo los valores del parámetro RMSE utilizando los datos reservados para test, de igual forma que se ha realizado en validación. También obtenemos el parámetro de correlación como dato adicional para tener más información.

$$Ypredecida_{test} = predict(Modelo\ obtenido, X_{test})$$

$$RMSE_{test} = \sqrt{mean((Ypredecida_{test} - Y_{test})^2)}$$

$$Correlación_{test} = corr(Ypredecida_{test}, Y_{test})$$

El resultado obtenido en test sirve para evaluar el nivel de sobreajuste del modelo, ya que podemos obtener un buen resultado en validación, pero si el modelo está demasiado ajustado a los valores de entrenamiento al realizar el testeo podemos no tener tan buenos resultados y obtener unos valores de RMSE en test mayores de lo esperado. En caso comprobar que existe un sobreajuste del modelo se testaría otro modelo para observar los resultados o se cambiaría la selección de las variables utilizadas inicialmente para realizar la predicción. También se podría considerar realizar ajustes en código como otro tipo de normalizaciones en las variables o cambiar el valor de algún parámetro de entrada que modifique la adecuación de los datos como el umbral_NaN (porcentaje del valores NaN respecto al número total de registros de la variable, para eliminar la información o asignar el valor medio a los datos detectados como NaN) o el umbral_outlier (umbral de desviación estándar, se utiliza para eliminar los datos que superen esta desviación).

5.2. Estudio de las variables predictivas.

El estudio de las variables predictivas tiene como finalidad reducir la incertidumbre a la hora de seleccionar las variables más adecuadas para realizar las predicciones en los modelos, para ello primero se ha consultado documentación en este ámbito para encontrar menciones a variables que fueran indicadores del nivel de funcionalidad final del paciente [6]-[10]. En esta documentación se ha encontrado variables de interés como edad, sexo, escolarización, gravedad del traumatismo, consumo de alcohol/tóxicos, nivel social y económico, tiempo en coma, etc. De todas estas variables también se han añadido algunas más interesantes que se han encontrado en la base de datos facilitada para hacer pruebas y sacar resultados. Para poder introducir estas variables en el modelo el único requisito que se tenía que cumplir era que tuvieran un número significativo de muestras válidas, ya que se ha encontrado en la base de datos con muchos campos incompletos en varios tipos de variables, por lo que estas no se incluirían, ya que por el uso de estas variables al tener campos un número considerable de nulos o valores incorrectos se eliminarían muchos registros de la base de datos con la que entrenaríamos a los modelos de *machine learning*.

Las variables utilizadas se encuentran clasificadas por tipo de variable e indicando el rango de valores numéricos o el tipo de valores categóricos que pueden tener y representando las gráficas para observar relaciones entre la variable estudiada y la variable a predecir. Se han utilizado para las variables numéricas gráficos de correlación y para las variables categóricas diagramas de cajas.

Los diagramas de cajas representan la distribución de la variable en cuartiles, en la siguiente imagen el cuartil 1 presenta el 25% de los datos, mientras que el cuartil 3 representa al 75% de los datos. La caja representa en que valores se sitúan la distribución de estos porcentajes indicando la media, fuera de la caja se representan los valores atípicos detectados. Como norma general se clasifica como variables más importantes para incluir en el modelo de predicción, aquellas en que los *boxplot* estén alineadas horizontalmente y presenten cierta variación. Indicar que en las gráficas donde se represente la distribución de las variables categóricas se muestran todas las clases de la variable, en aquellas clases que tengan pocas muestras serán eliminadas en el proceso de adecuación.

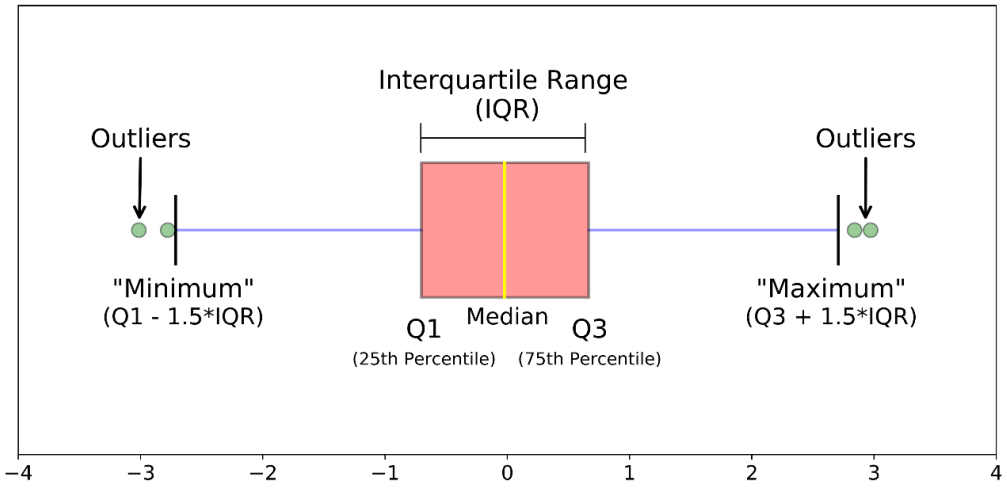


Figura 24. Boxplot. Obtenido en [16].

En la tabla 6 se muestra la clasificación de las variables numéricas según la correlación obtenida:

Clase	Variable	Correlación	Correlación Valor absoluto	Orden
FIM	FIM Cuidados Personales	0,97448	0,97448	1
FIM	FIM Movilidad	0,94588	0,94588	2
FIM	FIM Locomoción	0,93073	0,93073	3
Dinámica	IB TOTAL	0,92696	0,92696	4
FIM	FIM Comunicación	0,92515	0,92515	5
FIM	FIM Cognitiva	0,92169	0,92169	6
FIM	FIM Control esfínteres	0,90725	0,90725	7
FIM	FIM MOBILITY	0,89523	0,89523	8
FIM	FIM ADL	0,88851	0,88851	9
FIM	FIM SPHINCTER	0,86919	0,86919	10
Dinámica	DOS TOTAL	0,86625	0,86625	11
Dinámica	DRs	-0,86609	0,86609	12
FIM	FIM Conciencia del mundo exterior	0,8626	0,8626	13
FIM	FIM EXECUTIVE	0,7871	0,7871	14
Estática	Coma días	-0,39094	0,39094	16
Estática	Años escolaridad	0,089385	0,089385	17
Estática	Edad ingreso	-0,061229	0,061229	18

Tabla 6. Clasificación de variables numéricas.

A continuación, algunos ejemplos más significativos de las gráficas obtenidas de estas variables:

- **FIM Cuidados personales**: Rango de valores entre 7-49

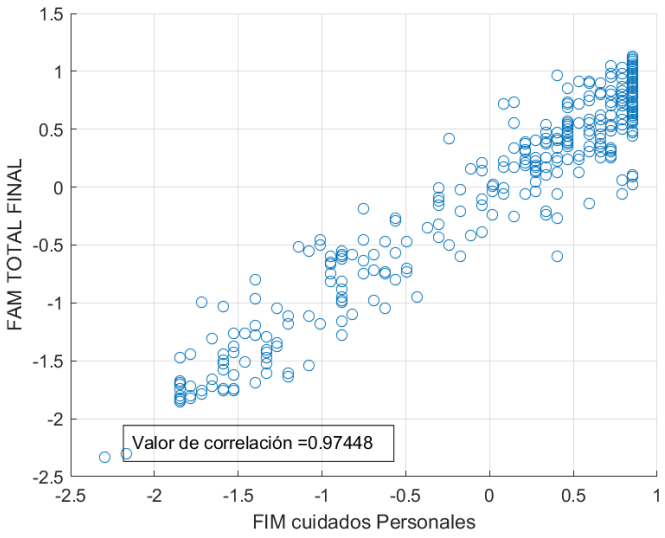


Figura 25. FIM Cuidados personales.

- **FIM Movilidad**: Rango de valores entre 4-28

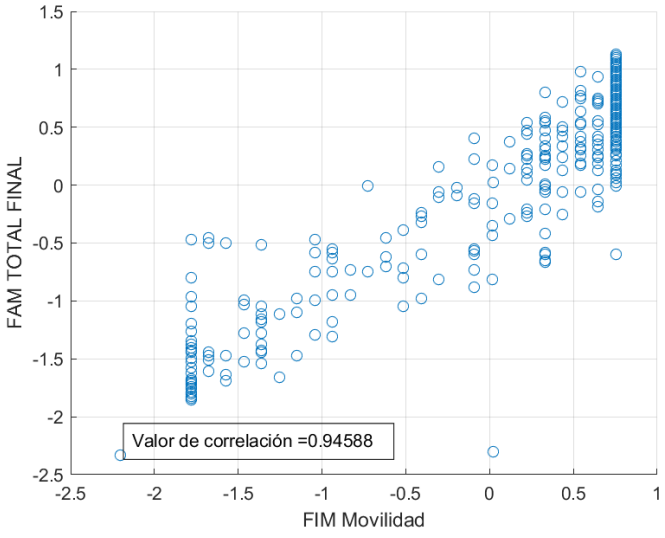


Figura 26. FIM Movilidad.

- **FIM Locomoción**: Rango de valores entre 3-21

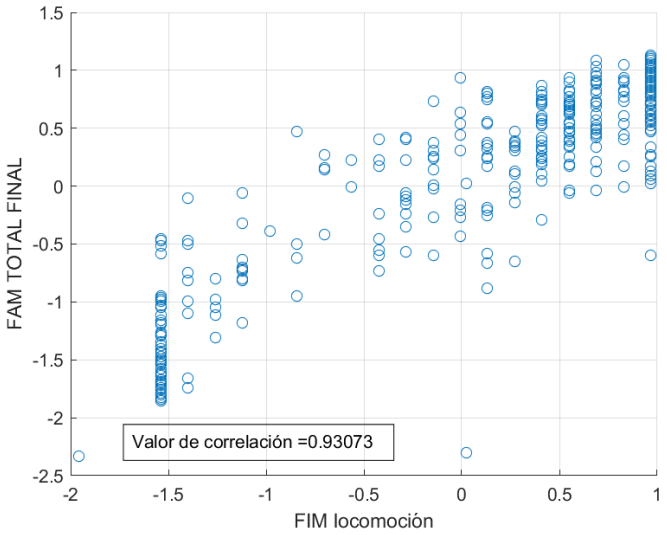


Figura 27. FIM Locomoción.

- **FIM Conciencia del mundo exterior**: Rango de valores entre 4-28

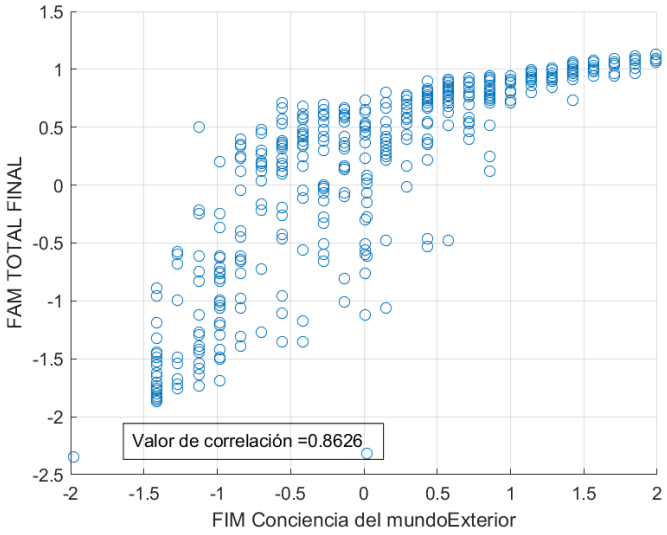


Figura 28. Conciencia del mundo exterior.

- **IB Total**: Rango de valores entre 0-100

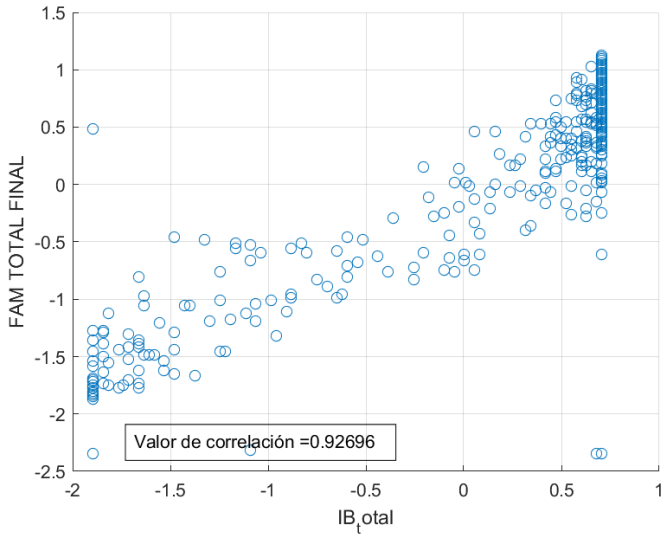


Figura 29. IB Total.

- **FIM Comunicación**: Rango de valores entre 5-35

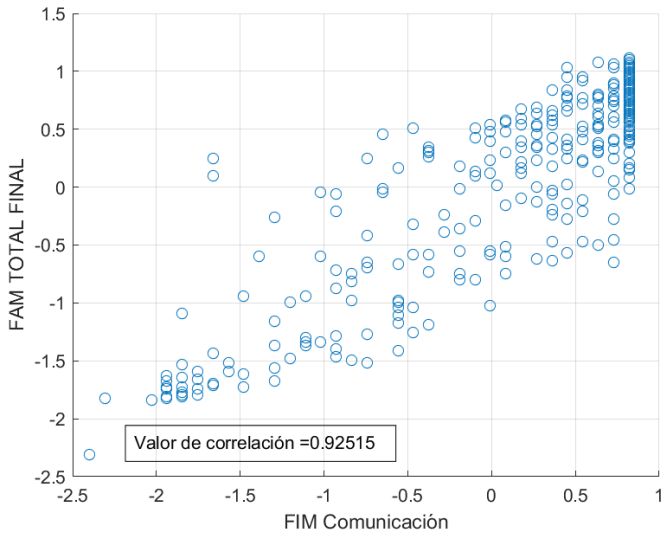


Figura 30. FIM Comunicación.

- **FIM Cognitiva:** Rango de valores entre 5-35

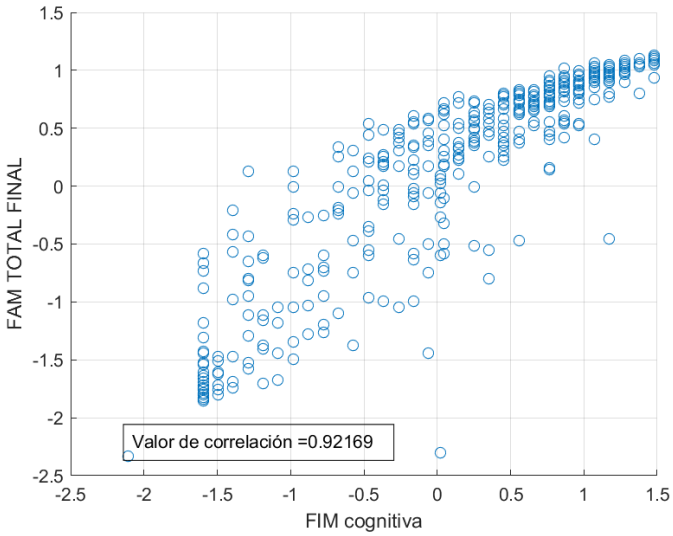


Figura 31. FIM Cognitiva.

- **FIM Control esfínteres**: Rango de valores entre 2-14

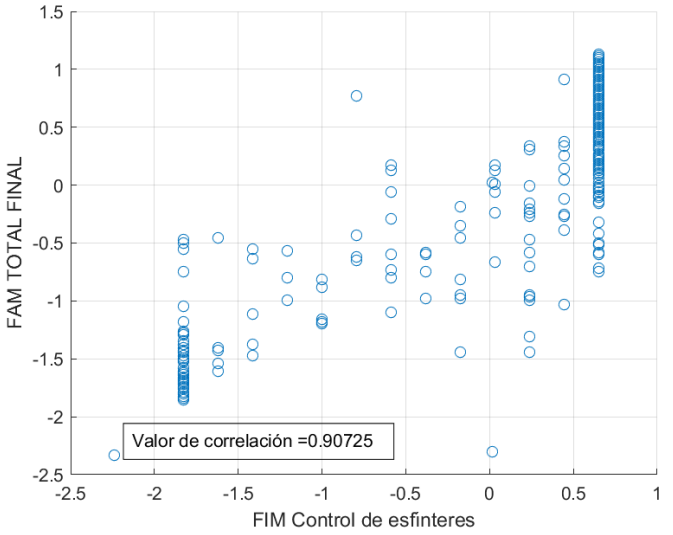


Figura 32. FIM Control de esfínteres.

En la siguiente tabla se muestra la clasificación de las variables categóricas, según la variación que se ha comprobado en las gráficas *boxplot*.

Clase	Variable	Capacidad discriminación
Dinámica	**Clasificación neurológica**	Elevada
Dinámica	**Cuidados y necesidades**	Elevada
Estática	**APT pronóstico**	Elevada
Estática	**Coma pronóstico**	Elevada
Dinámica	**GOS**	Buena
Estática	**GCS pronóstico**	Buena
Dinámica	**Tiene control**	Normal
Estática	**Nivel de estudios**	Baja
Estática	**Sexo**	Muy baja

Tabla 7. Clasificación de variables categóricas.

A continuación, se muestran los variables más importantes seleccionadas en la clasificación.

- **Clasificación neurológica**: Respuestas Mínimas, Fuera de APT, Periodo APT, Estado Vegetativo.

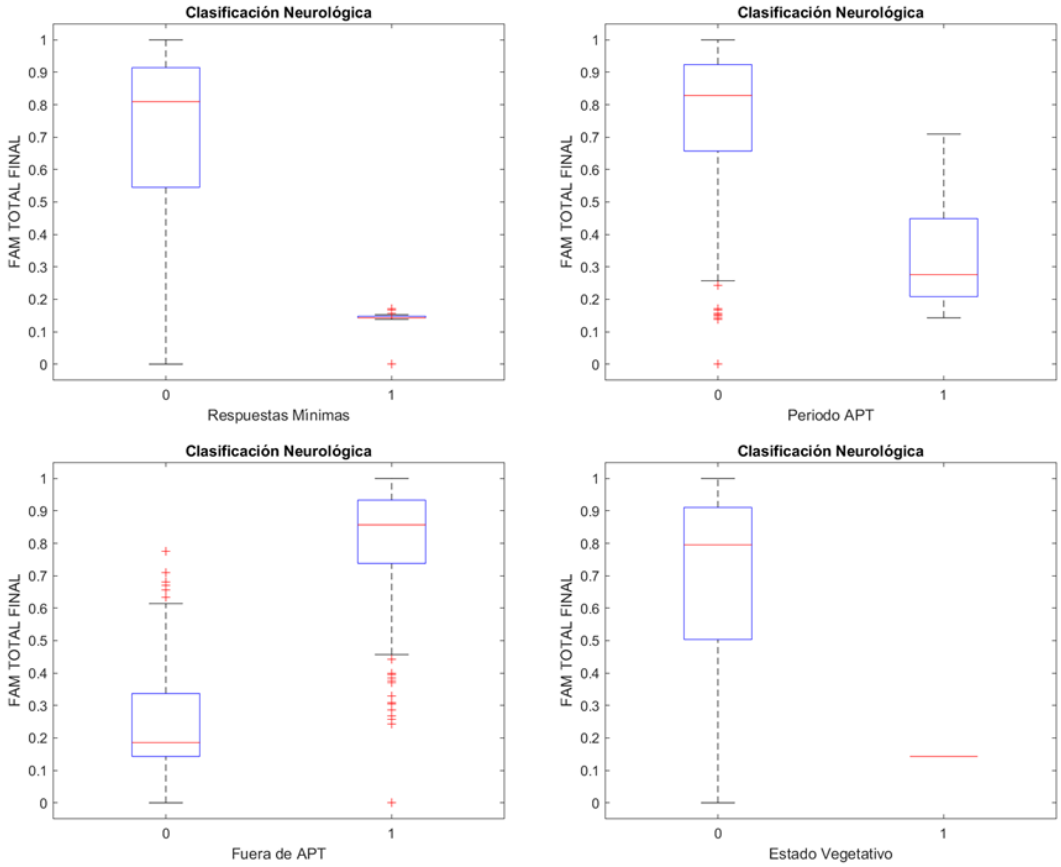


Figura 33. Clasificación neurológica.

- **Cuidados y necesidades**: Nivel 1, Nivel 2, Nivel 3, Nivel 4, Nivel 5, Nivel 6, Nivel 7, Nivel 8, Nivel 9.

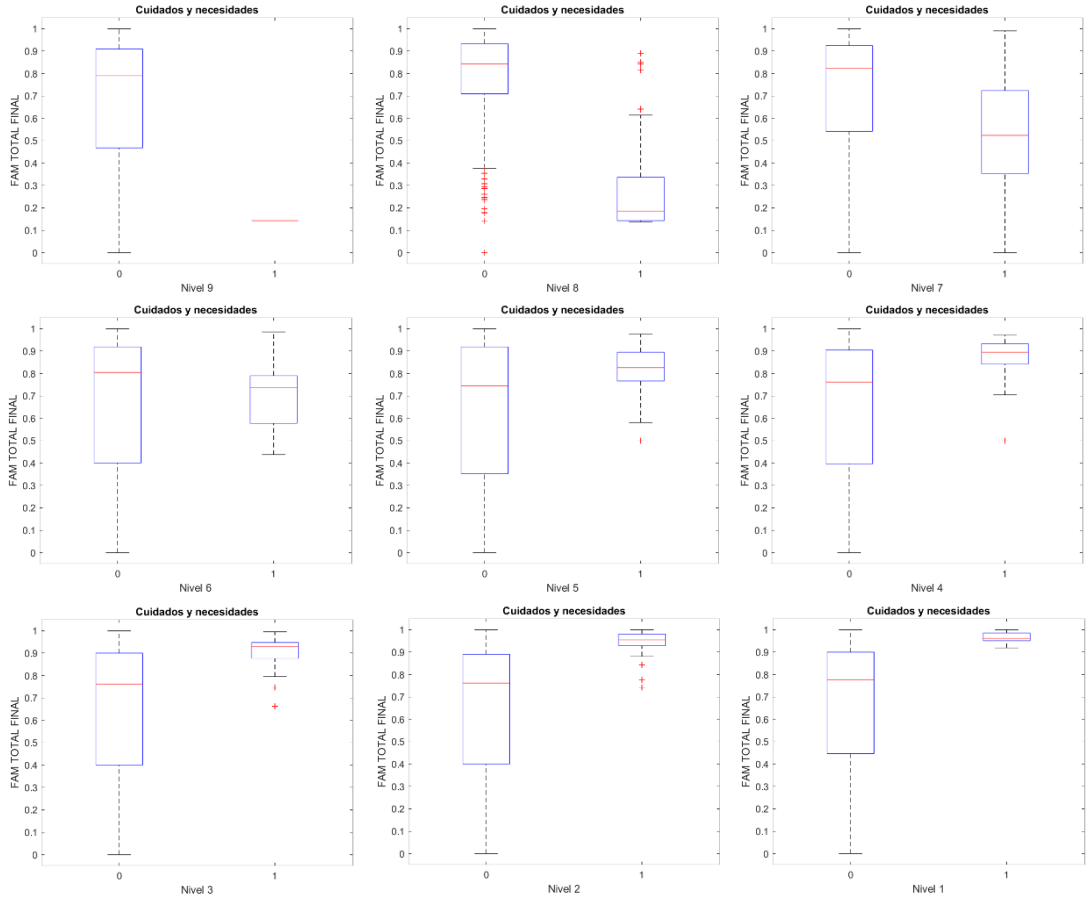


Figura 34. Cuidados y necesidades.

- **APT pronóstico**: Muy leve(<5min), Leve (5-60 min), Moderado (1-24 h), Grave (1-7 días), Muy grave (7-28 días), Extremadamente grave (>28 días)

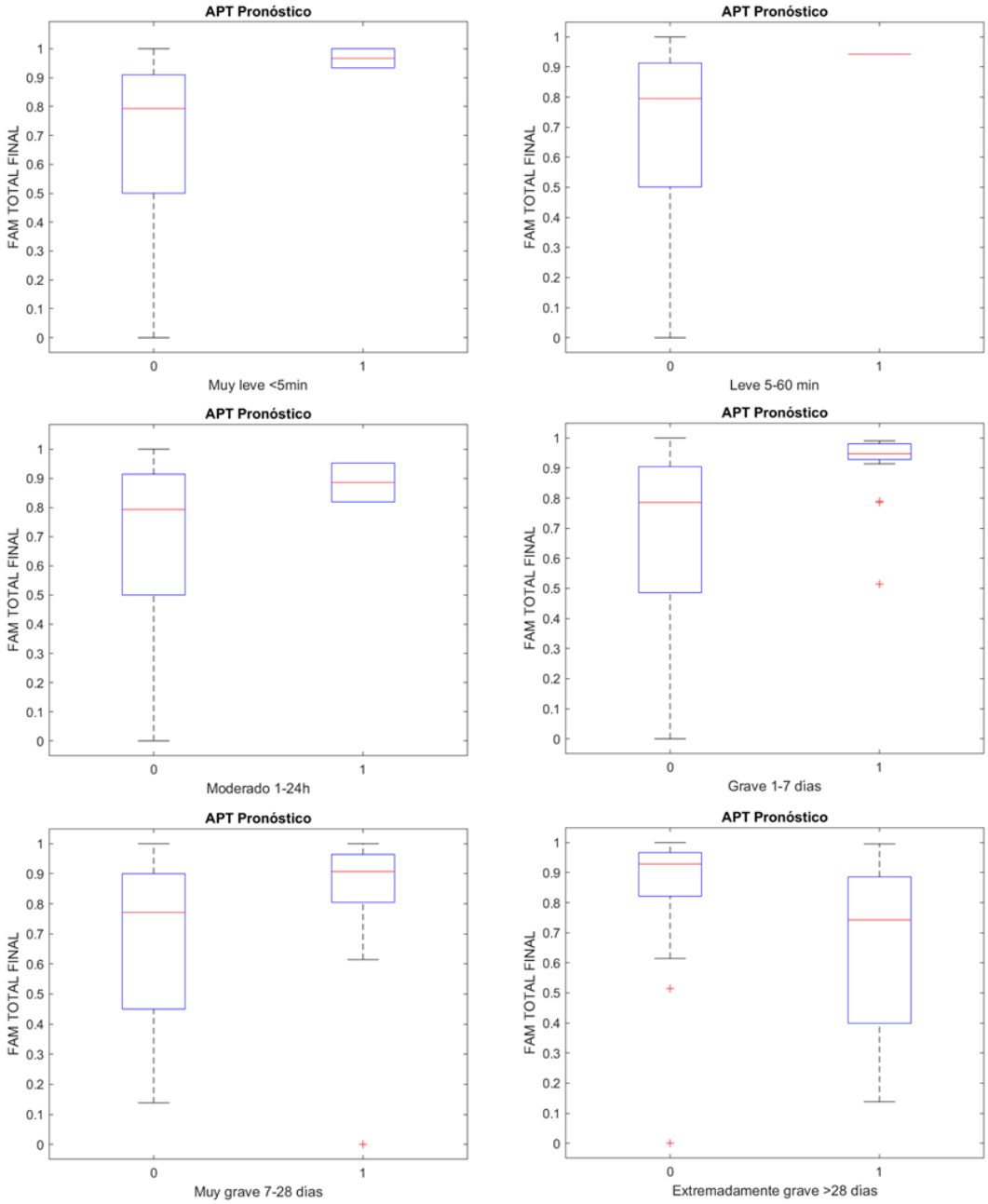


Figura 35. APT Pronóstico.

- **Coma pronóstico**: No coma, Leve(minutos), Moderado(<24h), Grave (<7 días), Muy grave (<1 mes), Extremadamente grave (>1 mes), Estado Vegetativo.

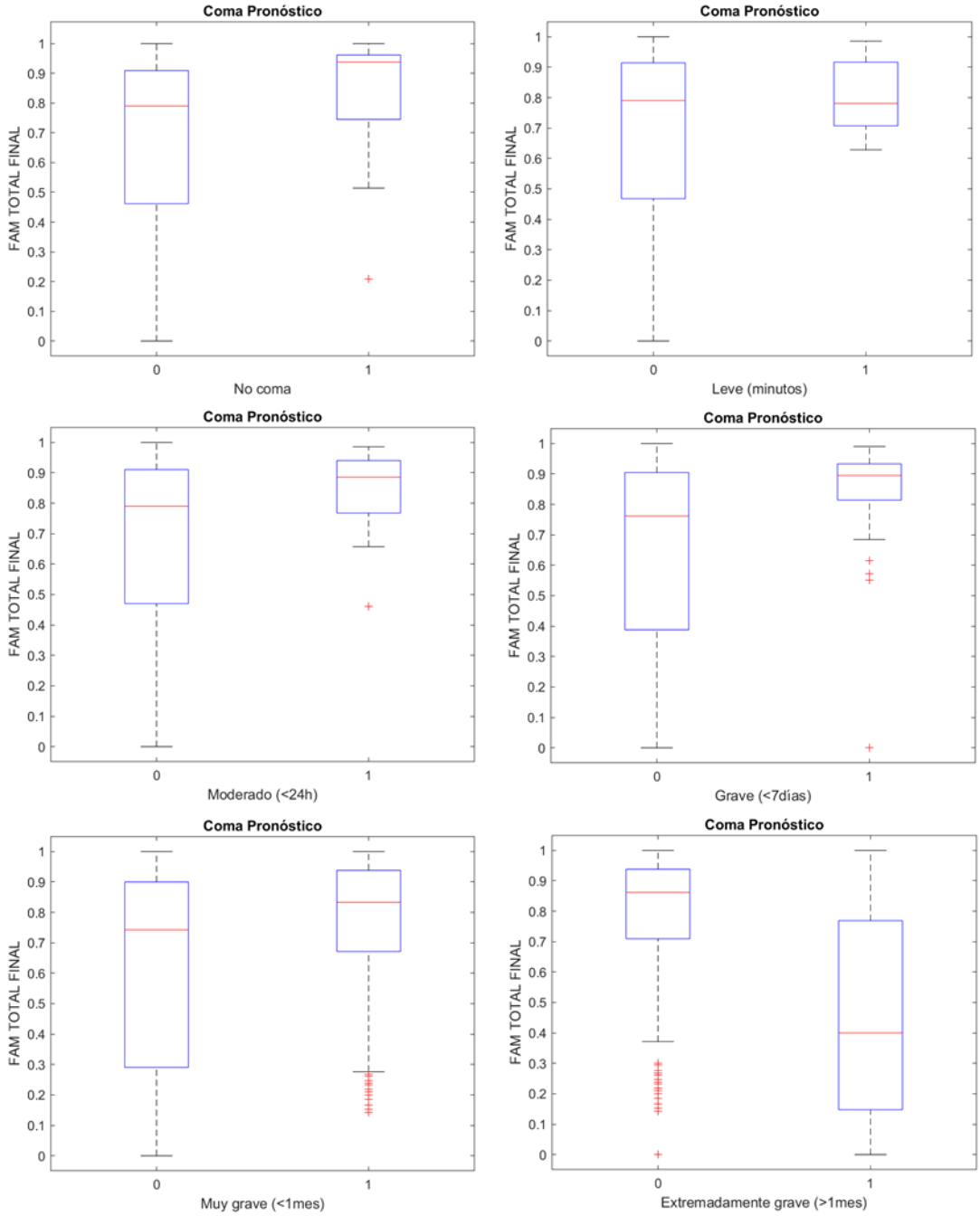


Figura 36. Coma pronóstico.

Se puede concluir del estudio realizado en este apartado que el nivel de correlación de las variables dinámicas es mucho mayor que el de las estáticas mostrado en la tabla 6, esta es una razón de por qué en anteriores estudios que no han integrado estas variables en sus modelos predictivos no han

obtenido resultados satisfactorios. En cuanto a las variables categóricas mostradas en la tabla 7 nos encontramos variables estáticas y dinámicas con buena capacidad de discriminación, potencialmente útiles para añadirlas en los modelos, no obstante, también encontramos variables estáticas no muy indicadas como el sexo o nivel de estudios, que se han utilizado en estudios anteriores para medir la funcionalidad final del paciente.

A partir de los resultados obtenidos en esta parte de la memoria, se realizarán diferentes estudios teniendo en cuenta la clasificación obtenida en las variables tanto numéricas como categóricas.

5.3. Experimentación de los modelos y nivel de predicción.

En esta sección se detallará la experimentación realizada con las variables de interés obtenidas en el punto *5.2 Estudio de las variables predictivas*. El objetivo de este apartado es demostrar qué modelos de *machine learning* funcionan mejor utilizando un conjunto de variables seleccionado previamente y basándonos en las medidas de los valores de RMSE, correlación y sobreajuste en la fase de validación. La variable a la que se está realizando la predicción es el nivel funcional final del paciente (FAM) y está variable está normalizada a la unidad, por lo tanto los valores que se obtendrán de RMSE serán respecto a la unidad también. Los mejores resultados que se obtengan en los tres estudios siguientes se mostrarán en el punto *5.4 Sistema de predicción final*, donde se evaluaran los mejores modelos en la fase de testeo.

5.3.1 Predicción mediante variables estáticas

En el primer estudio se ha utilizado los modelos de *machine learning* de *Random Forest, Linear Regression, Support Vector Machine* y *Multi-Layer Perceptron* y la arquitectura de las variables sin evolución temporal de la figura 5, haciendo simulaciones y obteniendo los resultados de los valores de RMSE y correlación. El mejor modelo será el que proporcione el mejor resultado en cuanto a RMSE, teniendo en cuenta el algoritmo de *machine learning* y el conjunto de variables utilizadas.

Para el modelo *Support Vector Machine* se ha asignado que el número total de árboles para entrenar el modelo sea de 30.

Para el *Multi-Layer Perceptron* se ha definido los siguientes valores para la arquitectura de la red y los hiperparámetros para realizar las simulaciones:

- Número de capas ocultas: [1,2]

- *Backpropagation algorithm*: trainlm
- *Epochs*: 1000
- *Goal*:0
- Número máximo de fallos en la validación: [10, 15, 20]
- Valor mínimo objetivo del gradiente: [10^{-7}, $10^{-8,}$ 10^{-9}]
- Valor inicial de mu: 0.01
- Factor de disminución de mu: 0.1
- Factor de aumento de mu: [2.5, 5, 10]
- Mu máximo: 10^{9}

En el caso del modelo *Long-Short Term Memory* se han utilizado los siguientes valores para la arquitectura de la red y los hiperparámetros:

- *Backpropagation algorithm*: Adam
- Número de capas ocultas:[10-15]
- Max*epochs*: 150.
- MiniBtachSize: 64
- GradientThreshold: 1
- InitialLearnRate: [0.01, 0.05]
- LearnRateDropPeriod: [175,125]
- LearnRateDropFactor: 0.3
- *GradientDecayFactor*: 0.99
- *SquaredGradientDecayFactor*: 0.99

Se ha elegido realizar la primera simulación (Modelo 0) del estudio solamente utilizando la variable numérica Coma días de la tabla 6, al ser la variable numérica y estática que mayor correlación tiene de las elegidas. Las simulaciones posteriores que se han realizado se ha incluido también esta variable, pues el objetivo ha sido reducir el valor de RMSE obtenido en este modelo añadiendo otras combinaciones de variables.

Modelo 0

Variables Numéricas: Coma días.

	RMSE	Correlación	Sobreajuste
***Random Forest Model* (RF)**	0,301542	0,28792	-0,323537
***Linear Regression* (LR)**	0,29578	0,301834	-0,237327
***Suppor Vector Machine* (SVM)**	0,313219	0,301834	-0,25536
***Multi-Layered Perceptron* (MLP)**	0,299	0,293218	-0,295204
MEDIA	0,30238525	0,2962015	-0.277857

Tabla 8. Estudio 1 - Modelo 0.

Se puede ver que el modelo que obtiene mejores resultados es *Linear Regression* con un RMSE de 0,29578. El valor medio de RMSE es de 0,30238525 y es el que se deberá mejorar añadiendo nuevas variables a los siguientes modelos.

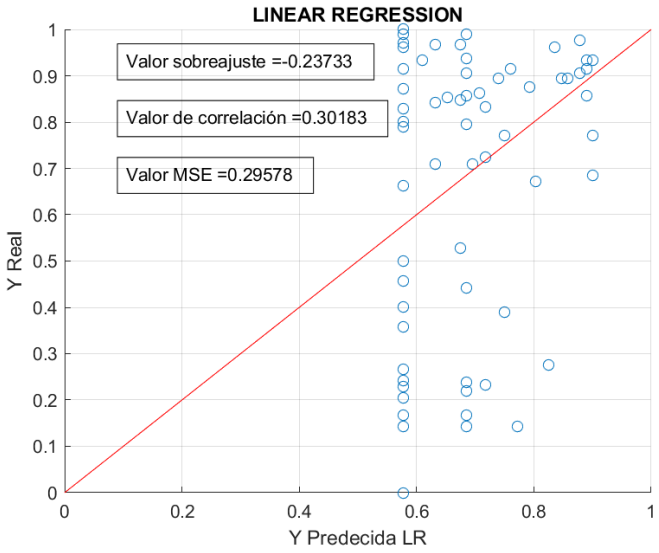


Figura 37. Estudio 1 - Modelo 0.

En la tabla siguiente se muestra los resultados obtenidos para las simulaciones realizadas en función de las variables numéricas y categóricas utilizadas. Se ha tener en cuenta que los valores de RMSE, correlación y sobreajuste han sido obtenidos como la media de los cuatro modelos de *machine learning* empleados.

Modelo	RMSE	Correlación	Sobreajuste	Variables numéricas	Variables categóricas
Modelo 1	0,30238525	0,2962015	-0.277857	Coma días	APT Pronóstico
Modelo 2	0,27091175	0,328277	-0.1746725	Coma días	CGS Pronóstico
Modelo 3	0,29666075	0,33229625	-0.269489	Coma días	Coma Pronóstico
Modelo 4	0,310541	0,28333	-0.33908625	Coma días	APT Pronóstico, Coma Pronóstico
Modelo 5	0,27986725	0,32285675	-0.2227485	Coma días, Edad	APT Pronóstico
Modelo 6	0,261139	0,4399645	-0.22667525	Coma días, Edad	APT Pronóstico, Coma Pronóstico
Modelo 7	0,26047125	0,442755	-0.24136725	Coma días, Edad	APT Pronóstico, Coma Pronóstico, GCS Pronóstico
Modelo 8	0,2571095	0,4592715	-0.232689	Coma días, Edad	APT Pronóstico, Coma Pronóstico, GCS Pronóstico, Sexo

Tabla 9. Estudio 1 - Tabla general resultados.

Se puede comprobar que modelo 8 es el que proporciona un mayor nivel de predicción, ya que presenta el menor medio de RMSE, también hay que tener en cuenta que es el modelo que utiliza el mayor número de variables estáticas. Se consigue apreciar un aumento en el nivel de correlación a medida que aumentamos el número de variables estáticas en el modelo.

En la siguiente tabla se refleja la efectividad de los modelos utilizados generalmente en todos los grupos de variables usados:

Modelo	Variables numéricas y categóricas		
	RMSE	**Correlación**	**Sobreajuste**
Random Forest	0,31032314	0,44250943	-0,33329614
Linear Regression	0,31777643	0,38831557	-0,26045957
Support Vector Machine	0,32665071	0,39721957	-0,25759514
Multi-Layer Perceptron	0,29859943	0,518483	-0,158359

Tabla 10. Estudio 1 - Resultados Modelos.

Se comprueba que Multi-Layer Perceptron es el modelo que mejor se adapta para este estudio de variables estáticas al tener el menor valor de RMSE de los cuatro modelos.

A continuación, se exponen los resultados del modelo 8 que como se ha comentado ha obtenido el mejor resultado de los modelos planteados en la tabla 9.

Modelo 8

Variables Numéricas: Coma días, Edad.

Variables categóricas: APT Pronóstico, Coma Pronóstico, GCS Pronóstico, Sexo.

Modelo	RMSE	Correlación	Sobreajuste
***Random Forest Model* (RF)**	0,252804	0,490964	-0,311216
***Linear Regression* (LR)**	0,270677	0,36714	-0,2403
***Suppor Vector Machine* (SVM)**	0,273794	0,374214	-0,229796
***Multi-Layered Perceptron* (MLP)**	0,231163	0,604768	-0,149444
MEDIA	0,2571095	0,4592715	-0,232689

Tabla 11. Estudio 1 - Modelo 8.

Como se aprecia en la tabla 11 el modelo que obtiene el mejor resultado es el *Multi-Layered Perceptron* con un RMSE de 0,231163. Este resultado se puede visualizar en la siguiente gráfica:

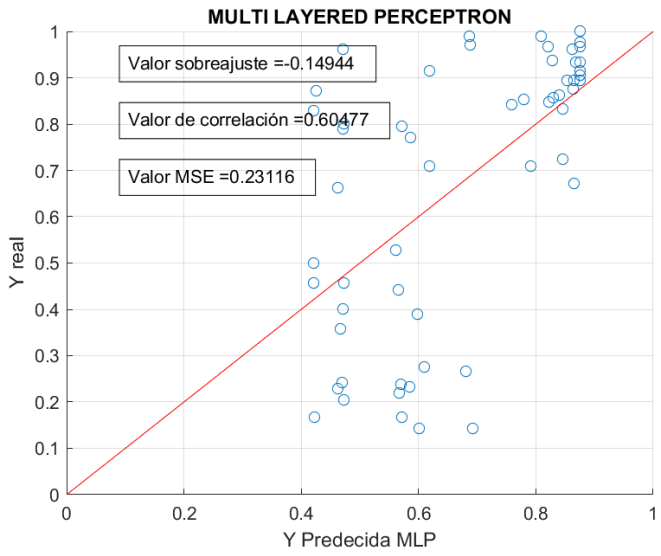


Figura 38. Estudio 1 - Modelo 8.

5.3.2 Predicción utilizando variables estáticas y dinámicas

Para este segundo estudio se ha empleado los mismos modelos y arquitectura que en el punto anterior de predicción de variables estáticas y se han calculado los valores de RMSE y correlación de cada modelo estudiado. Se tendrá en cuenta para el modelo ganador tanto el conjunto de variables seleccionado como el modelo de *machine learning* que obtenga el mejor resultado. También se analizará cual es el modelo de *machine learning* que proporciona mejores resultados a nivel general para todos los conjuntos de variables mencionados, para ello se calculará la media de los resultados medidos.

Los hiperparámetros utilizados en este estudio son los mismos que los utilizados en el estudio anterior, ya que los modelos que se utilizan son los mismos.

Hay que considerar que se parte del Modelo 0, el cual utiliza solamente la variable dinámica FAM, a partir de la primera valoración del paciente se intenta predecir cual será su nivel funcional final. Posteriormente se introducirán nuevas variables para mejorar este modelo partiendo de este nivel de predicción.

Modelo 0

Variables Numéricas: FAM.

	RMSE	Correlación	Sobreajuste
***Random Forest Model* (RF)**	0,16034	0,86327	-0,176092
***Linear Regression* (LR)**	0,190903	0,811261	-0,066741
***Suppor Vector Machine* (SVM)**	0,191099	0,811261	-0,061605
***Multi-Layered Perceptron* (MLP)**	0,151594	0,890415	0,041603
MEDIA	**0,173484**	0,84405175	-0,06570875

Tabla 12. Estudio 2 - Modelo 0.

Se observa que el mejor resultado en este caso sería utilizando *Multi-Layer Perceptron* con un RMSE de 0,151594. Se parte sobre la predicción realizada en este modelo con un valor medio de RMSE de 0,173484, el objetivo será mejorar el RMSE generalmente en todos los modelos haciendo una selección adecuada de las variables y seleccionar el mejor algoritmo de *machine learning* que permite realizar la mejor predicción.

Se ha iniciado el estudio haciendo uso solamente de variables numéricas para concretar cuales son más efectivas en el nivel final de la predicción, posteriormente se añadirán las variables categóricas para acabar de afinar la predicción de los modelos y concretar resultados. A continuación, los siguientes resultados mostrados han sido los que han obtenido un menor RMSE en la predicción utilizando combinaciones de variables numéricas de la tabla 6. Los valores de RMSE y correlación indicados fuera de las gráficas es el valor medio de los resultados de los cuatro modelos de *machine learning*, con el fin de comprobar la efectividad cada conjunto de variables

Modelo 1

Variables Numéricas FIM: FIM cuidados Personales, FIM Movilidad, FIM locomoción, FIM Comunicación, FIM Conciencia del mundo exterior, FIM cognitiva, FIM control esfínteres, FAM.

Modelo	RMSE	Correlación	Sobreajuste
Random Forest (RF)	0,15771	0,88724	0,029514
Linear Regression (LR)	0,14978	0,89247	-0,24098
Support Vector Machine (SVM)	0,14912	0,88587	0,11313
Multi-Layer Perceptron (MLP)	0,14646	0,89949	-0,14926
MEDIA	0,1507675	0,8912675	-0.061899

Tabla 13. Estudio 2 - Modelo 1.

Para este conjunto de variables el modelo ganador es *Multi-Layer Perceptron* con un RMSE de 0.14646.

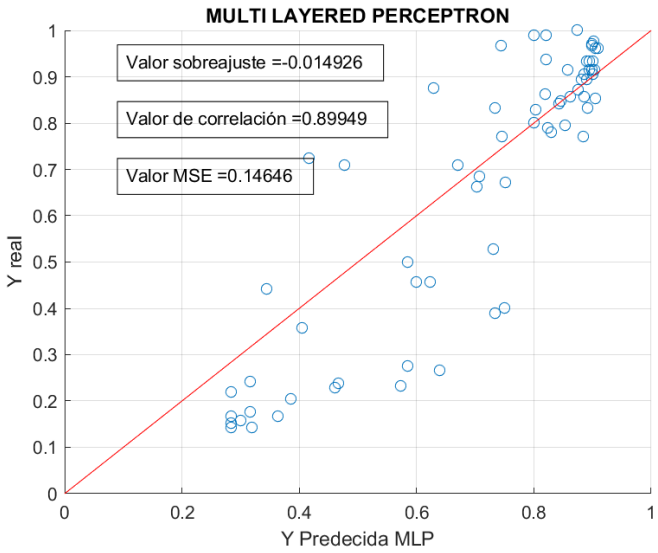


Figura 39. Estudio 2 - Modelo 1 - MLP.

Modelo 2

Variables Numéricas: FIM cuidados Personales, FIM Movilidad, FIM locomoción, FIM Comunicación, FIM Conciencia del mundo exterior, FIM cognitiva, FIM control esfínteres, IB Total.

Modelo	RMSE	Correlación	Sobreajuste
Random Forest (RF)	0,149035	0,892617	0,274162
Linear Regression (LR)	0,15872	0,885728	0,021782
Support Vector Machine (SVM)	0,151185	0,880964	0,099086
Multi-Layer Perceptron (MLP)	0,149039	0,892348	0,026959
MEDIA	0,15199475	0,88791425	0.10549725

Tabla 14. Estudio 2 - Modelo 2.

Para este conjunto de variables el modelo ganador es *Random Forest* con un RMSE de 0,149035.

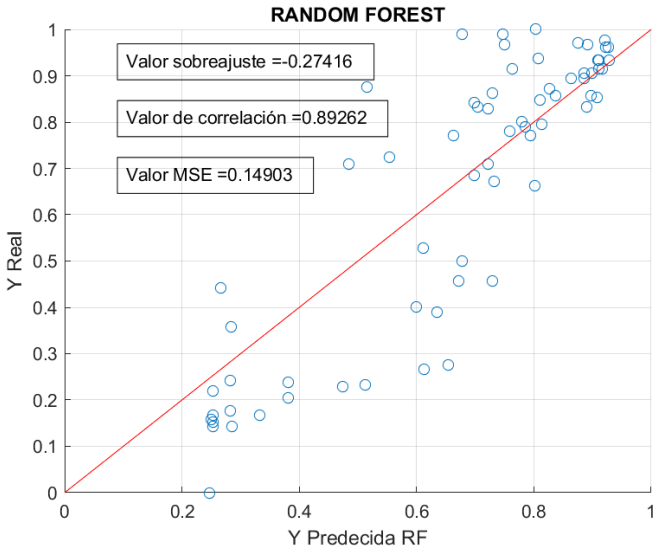


Figura 40. Estudio 2 - Modelo 2 - RF.

Modelo 3

Variables Numéricas: FIM cuidados Personales, FIM Movilidad, FIM locomoción, FIM Comunicación, FIM Conciencia del mundo exterior, FIM cognitiva, IB Total, FAM.

Modelo	RMSE	Correlación	Sobreajuste
Random Forest (RF)	0,153082	0,884269	-0,293239
Linear Regression (LR)	0,157814	0,88538	0,031537
Support Vector Machine (SVM)	0,154063	0,874524	0,077948
Multi-Layer Perceptron (MLP)	0,148852	0,894357	-0,008538
MEDIA	0,15345275	0,8846325	-0.048073

Tabla 15. Estudio 2 - Modelo 3.

Para este grupo de variables el modelo ganador es *Multi-Layer Perceptron* con un RMSE de 0,148852.

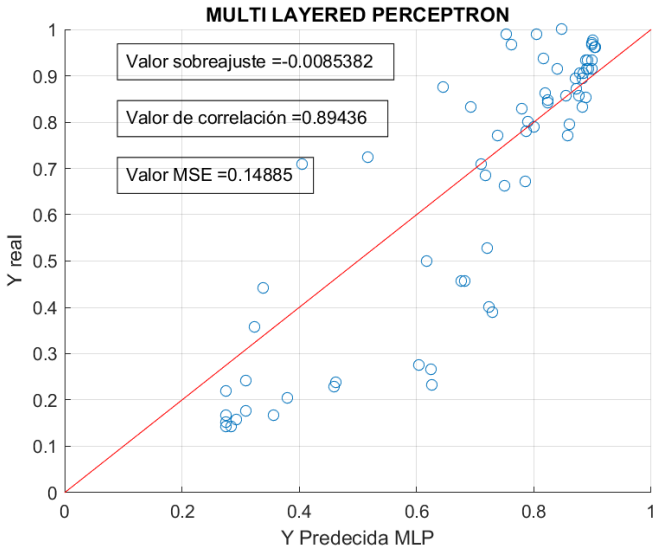


Figura 41. Estudio 2 - Modelo 3 - MLP.

Modelo 4

Variables Numéricas: FIM cuidados Personales, FIM Movilidad, FIM locomoción, FIM Comunicación, FIM cognitiva, FIM control esfínteres, IB Total, FAM.

Modelo	RMSE	Correlación	Sobreajuste
Random Forest (RF)	0,151318	0,889319	-0,283595
Linear Regression (LR)	0,157591	0,885659	0,033636
Support Vector Machine (SVM)	0,151957	0,880249	0,094639
Multi-Layer Perceptron (MLP)	0,147	0,895543	0,045212
MEDIA	0,1519665	0,8876925	-0.027527

Tabla 16. Estudio 2 - Modelo 4.

Utilizando esta agrupación de variables el modelo con menor RMSE es *Multi-Layer Perceptron* con un valor de 0,147.

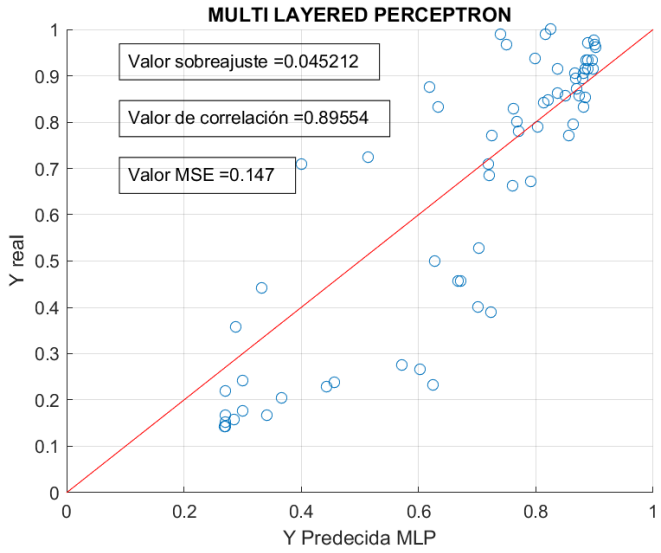


Tabla 17. Estudio 2 - Modelo 4 - MLP.

Modelo 5

Variables Numéricas: FIM cuidados Personales, FIM Movilidad, FIM locomoción, FIM Comunicación, FIM cognitiva, FIM control esfínteres.

Modelo	RMSE	Correlación	Sobreajuste
Random Forest (RF)	0,155079	0,883535	-0,307089
Linear Regression (LR)	0,156619	0,887149	0,040908
Support Vector Machine (SVM)	0,148353	0,885456	0,123946
Multi-Layer Perceptron (MLP)	0,146712	0,896832	0,049621
MEDIA	0,15169075	0,888243	-0.0231535

Tabla 18. Estudio 2 - Modelo 5.

Haciendo uso de este conjunto de variables el modelo ganador es *Multi-Layer Perceptron* con un RMSE de 0,146712.

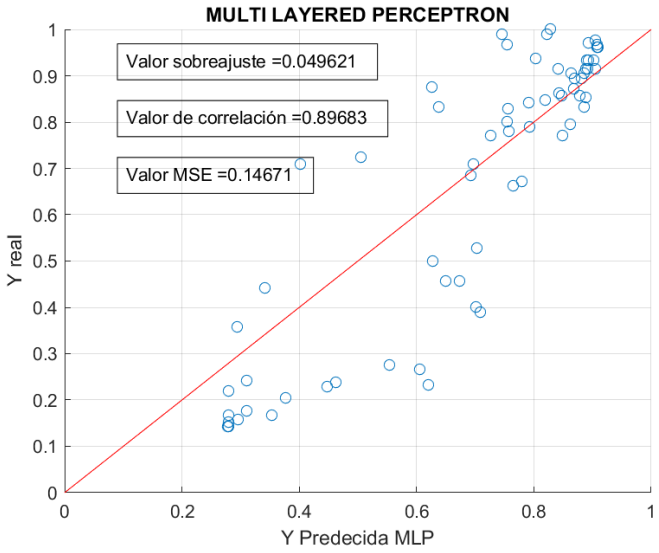


Figura 42. Estudio 2 - Modelo 5 - MLP.

Los resultados anteriores agrupados están contenidos en la siguiente tabla para una mejor evaluación general:

Modelo	RMSE	Correlación	Sobreajuste
Modelo 1	0,1507675	0,8912675	-0,061899
Modelo 2	0,15199475	0,88791425	0,10549725
Modelo 3	0,15345275	0,8846325	-0,048073
Modelo 4	0,1519665	0,8876925	-0,027527
Modelo 5	0,15169075	0,888243	-0,0231535

Tabla 19. Resumen estudio 3 con variables numéricas.

Observando los resultados se aprecia que el modelo que mejor resultados aporta utilizando solamente variables numéricas es el modelo 1. En la selección de estos modelos se puede apreciar la existencia de las variables FIM, esto es debido a que son el principal indicador de la funcionalidad final del paciente, sin tener en cuenta la variable FAM. Además, en todos se ha mejorado el valor obtenido del RMSE en el modelo 0 que era de 0,173484. También notarse que los niveles de correlación en general son bastante buenos debido puesto que la selección de variables es adecuada. Los resultados de los modelos anteriores serán tenidos en cuenta para realizar los próximos estudios añadiendo variables categóricas.

En la siguiente tabla se muestran diez estudios elaborados para realizar comparaciones entre los resultados obtenidos en cada grupo de variables utilizado. Notar que el RMSE y la correlación son los valores medios obtenidos del conjunto de modelos de *machine learning.*

- **Combinación 1 variables numéricas**: FIM cuidados Personales, FIM Movilidad, FIM locomoción, FIM Comunicación, FIM Conciencia del mundo exterior, FIM cognitiva, FIM control esfínteres, FAM.
- **Combinación 2 variables numéricas:** FIM cuidados Personales, FIM Movilidad, FIM locomoción, FIM Comunicación, FIM cognitiva, FIM control esfínteres, IB Total, FAM.

Modelo	RMSE	Correlación	Sobreajuste	Variables numéricas	Variables categóricas
Modelo 6	0,14977975	0,89186975	-0,07110175	Combinación 1	Clasificación neurológica
Modelo 7	0,1524655	0,8821075	-0,02167025	Combinación 1	Cuidados y necesidades
Modelo 8	0,15378625	0,86436125	-0,045863	Combinación 1	APT Pronóstico
Modelo 9	0,168365	0,86525975	-0,183408	Combinación 1	Coma pronóstico
Modelo 10	0,15468775	0,8813555	-0,05375025	Combinación 1	GOS
Modelo 11	0,15636425	0,88015975	-0,0666125	Combinación 1	GCS Pronóstico
Modelo 12	0,1608715	0,847482	-0,11093175	Combinación 1	APT Pronóstico, GCS Pronóstico
Modelo 13	0,15383125	0,8671175	-0,09588925	Combinación 1	APT Pronóstico, Clasificación neurológica
Modelo 14	0,15428975	0,86230025	-0,0327185	Combinación 2	APT Pronóstico, Cuidados y necesidades.
Modelo 15	0,1519385	0,880017	-0,088493	Combinación 2	Clasificación neurológica, Cuidados y necesidades.

Tabla 20. Resumen estudio 2 con variables numéricas y categóricas.

Como se ve en la tabla el único modelo que logra superar el valor RMSE de 0,1507675 obtenido en el modelo 1, ha sido el modelo 6 utilizando la variable categórica dinámica Clasificación neurológica, sin embargo, el valor de RMSE no se ha reducido apenas debido a que las variables que más afectan a la predicción final del paciente son las numéricas tipo FIM como se ha comentado.

En la siguiente tabla se refleja la efectividad de los modelos utilizados generalmente en todos los grupos de variables usados:

Modelo	Variables numéricas			Variables numéricas y categóricas		
	RMSE	**Correlación**	**Sobreajuste**	**RMSE**	**Correlación**	**Sobreajuste**
Random Forest	0,1532448	0,887396	0.2375198	0,1581137	0,1581137	0,2961252
Linear Regression	0,1561048	0,8872772	0,0737686	0,1577411	0,8739591	0,0286628
Support Vector Machine	0,1509356	0,8814126	0,1017498	0,1535422	0,8689372	0,0557046
Multi-Layer Perceptron	0,1476126	0,895714	0,055918	0,1531548	0,8795812	0,0408434

Tabla 21. Resumen modelos - Estudio 2.

Se puede observar que el modelo Multi-Layer Perceptron es el más efectivo utilizando tanto variables numéricas solamente, como variables numéricas y categóricas.

A continuación, se muestran los resultados del modelo 6 que ha obtenido el mejor resultado con el menor RMSE medio.

Modelo 6

Variables numéricas: FIM cuidados Personales, FIM Movilidad, FIM locomoción, FIM Comunicación, FIM Conciencia del mundo exterior, FIM cognitiva, FIM control esfínteres, FAM.

Variables categóricas: Clasificación neurológica.

Modelo	RMSE	Correlación	Sobreajuste
Random Forest (RF)	0,156659	0,878903	-0,358523
Linear Regression (LR)	0,149691	0,897722	0,012468
Support Vector Machine (SVM)	0,144691	0,89771	0,059932
Multi-Layer Perceptron (MLP)	0,148078	0,893144	0,001716
MEDIA	0,14977975	0,89186975	-0.07110175

Tabla 22. Estudio 2 - Modelo 6.

Para la selección de variables que han proporcionado un mejor nivel en la predicción el modelo de *machine learning* que ha obtenido el mejor resultado ha sido *Support Vector Machine,* con un RMSE de 0.144691, además de todas las simulaciones realizadas, aunque el *Multi-Layer Perceptron* es el que realiza las mejores predicciones en general, este ha sido el mejor resultado obtenido y por lo tanto el ganador en este estudio.

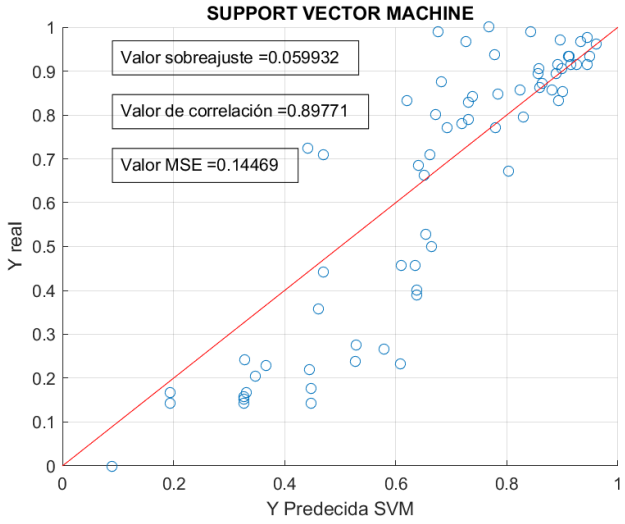


Figura 43. Estudio 2 - Modelo 6 - SVM.

5.3.3 Predicción utilizando la evolución temporal de variables estáticas y dinámicas

En esta parte del estudio se ha utilizado el modelo LSTM y la arquitectura de las variables con evolución temporal de la figura 7. En este estudio se ha medido en las simulaciones realizadas los valores de RMSE y correlación para un número de valoraciones entre 1 y 7. El modelo elegido será aquel que obtenga menores resultados de manera general en todas las valoraciones estudiadas, por lo tanto, el modelo ganador será el que tenga una media de RMSE menor.

Para el entrenamiento de la red neuronal se han hecho simulaciones teniendo en cuenta el número de capas ocultas indicadas (se ha utilizado entre 10 y 15 capas), además de los parámetros utilizados para optimizar los modelos, los cuales se recogen en la siguiente tabla:

Backpropagation algorithm	Adam
Max*epochs*	150
MiniBtachSize	64
GradientThreshold	1
InitialLearnRate	0.01, 0.05
LearnRateDropPeriod	175, 125
LearnRateDropFactor	0.3, 0.4
GradientDecayFactor	0.99
SquaredGradientDecayFactor	0.99

Tabla 23. Parámetros estudio 3 - LSTM.

Hay que tener en cuenta que en todas las simulaciones mostradas en este punto se está utilizando la evolución de la variable FAM, obviamente porque es el principal indicador para predecir la funcionalidad final del paciente. Dicho esto, se parte de un Modelo 0 en el cual solo se utiliza esta variable para predecir la funcionalidad final y posteriormente se intenta mejorar el nivel de predicción utilizando variables numéricas y categóricas obtenidas del apartado ***5.2 Estudio de variables predictivas.***

Modelo 0

Variables Numéricas: FAM

Multi-Layered Perceptron(MLP) - LSTM	RMSE	Correlación	Sobreajuste
1 valoraciones	0,148745	0,879145	0,111774
2 valoraciones	0,103874	0,945242	0,208083
3 valoraciones	0,067273	0,976748	0,37704
4 valoraciones	0,071734	0,973968	0,349264
5 valoraciones	0,067297	0,977738	0,372634
6 valoraciones	0,062777	0,98143	0,354131
7 valoraciones	0,070674	0,974141	0,111774
MEDIA	0,08462486	0,95834457	0,31417914

Tabla 24. Estudio 3 - Modelo 0.

En el modelo cero tenemos el mejor resultado para la predicción utilizando 6 valoraciones con un incremento notable en la siguiente valoración. El valor RMSE medio el modelo 0 es de 0.08462486, en las siguientes predicciones que se realicen se intentará reducir este valor integrando al modelo variables numéricas y categóricas.

De la misma manera que el estudio realizado en el punto anterior, se ha empezado simulando modelos con solo variables numéricas obtenidas de la tabla 6, para posteriormente añadir las variables categóricas que sean más relevantes y comparar estudios.

Las agrupaciones de las variables numéricas que se han utilizado en cada modelo se indican a continuación:

- **Variables numéricas modelo 1**: FIM cuidados Personales, FIM Movilidad, FIM locomoción, FIM Comunicación, FIM Conciencia del mundo exterior, FIM cognitiva, FIM control esfínteres, FAM.
- **Variables numéricas modelo 2**: Variables Numéricas: FIM cuidados Personales, FIM Movilidad, FIM locomoción, FIM Comunicación, FIM Conciencia del mundo exterior, FIM cognitiva, FIM control esfínteres, IB Total, FAM.
- **Variables numéricas modelo 3**: Variables Numéricas: FIM cuidados Personales, FIM Movilidad, FIM locomoción, FIM Comunicación, FIM Conciencia del mundo exterior, FIM cognitiva, IB Total, FAM.
- **Variables numéricas modelo 4:** Variables Numéricas: FIM cuidados Personales, FIM Movilidad, FIM locomoción, FIM Comunicación, FIM cognitiva, FIM control esfínteres, IB Total, FAM.

- **Variables numéricas modelo 5**: Variables Numéricas: FIM cuidados Personales, FIM Movilidad, FIM locomoción, FIM Comunicación, FIM cognitiva, FIM control esfínteres, FAM.

Los resultados obtenidos de los modelos anteriores se recogen en la siguiente tabla para un mejor análisis:

Modelo	RMSE	Correlación
Modelo 1	0,079460143	0,96320314
Modelo 2	0,08415943	0,96111571
Modelo 3	0,08355357	0,96125471
Modelo 4	0,080852	0,96324486
Modelo 5	0,08157471	0,96192

Tabla 25. Resumen estudio 3 con variables numéricas.

Se puede concluir que el mejor modelo utilizando solo variables numéricas es el primero al obtener un menor RMSE. Se observa en los modelos mostrados la presencia de las variables FIM, pues son los mayores indicadores de la funcionalidad final del paciente, igualmente los niveles de correlación similares en cada modelo muestran una elección acertada en las variables. Además, en todos los modelos se ha disminuido el valor medio de RMSE de 0.08462486 obtenido en el modelo cero. El modelo 1 será tenido en cuenta para realizar los siguientes estudios añadiendo variables categóricas, aunque también se probará otro tipo de combinaciones con otras variables numéricas que no estén en el mejor modelo para comprobar la variación de resultados.

En la tabla siguiente se muestran diez estudios realizados para ver los diferentes de resultados utilizando distintas combinaciones de variables. Indicar que RMSE y correlación son los valores medios obtenidos de cada modelo para todas las valoraciones.

Combinación 1 variables numéricas: FIM cuidados Personales, FIM Movilidad,FIM locomoción,FIM Comunicación, FIM Conciencia del mundoExterior, FIM cognitiva,FIM control esfínteres, FAM.

Combinación 2 variables numéricas: FIM cuidados Personales, FIM Movilidad, FIM locomoción, FIM Comunicación, FIM cognitiva, FIM control esfínteres, IB Total, FAM.

Modelo	RMSE	Correlación	Variables numéricas	Variables categóricas
Modelo 6	0,08311043	0,96007929	Combinación 1	Clasificación neurológica
Modelo 7	0,08450514	0,958426	Combinación 1	Cuidados y necesidades
Modelo 8	0,07591971	0,96043271	Combinación 1	APT Pronóstico
Modelo 9	0,07867443	0,96584	Combinación 1	Coma pronóstico
Modelo 10	0,08311271	0,95986843	Combinación 1	GOS
Modelo 11	0,07756471	0,96558629	Combinación 1	GCS Pronóstico
Modelo 12	0,08174671	0,95299543	Combinación 1	APT Pronóstico, GCS Pronóstico
Modelo 13	0,07743057	0,95917571	Combinación 1	APT Pronóstico, Clasificación neurológica
Modelo 14	0,07959043	0,95847871	Combinación 2	APT Pronóstico
Modelo 15	0,08229243	0,96154886	Combinación 2	Clasificación neurológica

Tabla 26. Resumen estudio 3 con variables numéricas y categóricas.

Como se puede apreciar los modelos 8, 9 y 11 mejoran ligeramente el resultado obtenido del modelo uno con un RMSE de 0,079460143, por lo tanto, introducir las variables APT pronóstico, Coma pronóstico o GCS pronóstico pueden ayudar a mejorar el modelo final de manera breves siendo la variable APT pronóstico la más destacada, ya que el peso mayor de la predicción recae en las variables numéricas FIM de la combinación 1 mencionada. Hay que mencionar que en la tabla el modelo 13 también presenta un mejor resultado que el modelo 1, pero este no se considera entre los modelos ganadores, ya que se ha utilizado las variables categóricas APT pronóstico y clasificación neurológica, por lo tanto este modelo se debería comparar con el modelo 8 que solo utiliza la variable categórica APT pronóstico, siendo este último el que mejor resultado

proporciona, por lo tanto el añadir la variable clasificación neurológica al modelo LSTM en este estudio parece no indiciar mejorar el nivel de predicción. Esto también se puede comprobar comparando los modelos 1 y 6, donde en el modelo 6 se añade la variable clasificación neurológica y hace empeorar la predicción en el modelo.

A continuación, se muestran los resultados del modelo 8, que ha sido el que mejor resultado ha tenido en el estudio de este apartado con un valor de RMSE más bajo.

Modelo 8

Variables Numéricas: FIM cuidados Personales, FIM Movilidad, FIM locomoción, FIM Comunicación, FIM cognitiva, FIM control esfínteres, FAM.

Variables categóricas: APT Pronóstico.

Multi-Layered Perceptron(MLP) - LSTM	RMSE	Correlación	Sobreajuste
1 valoraciones	0,14755	0,870788	0,106452
2 valoraciones	0,098335	0,945703	0,167028
3 valoraciones	0,064031	0,976676	0,28644
4 valoraciones	0,055938	0,982275	0,381265
5 valoraciones	0,052388	0,983794	0,413444
6 valoraciones	0,055592	0,982328	0,420311
7 valoraciones	0,057604	0,981465	0,360788
MEDIA	0,07591971	0,96043271	0,305104

Tabla 27. Estudio 3 - Modelo 8 - LSTM.

Se observa en la tabla 27 que el modelo adquiere su mejor predicción en la quinta valoración con un RMSE de 0,052388, con un aumento ligero de RMSE en las siguientes valoraciones.

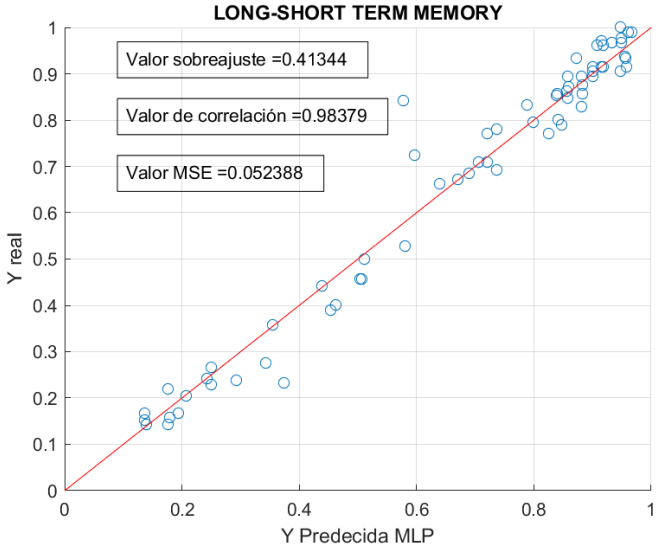


Figura 44. Estudio 3 - Modelo 8 - LSTM.

5.4. Sistema de predicción final.

En este apartado de la memoria se recogen los mejores modelos simulados en cada estudio del apartado anterior para los datos de validación y se evalúan con los datos de testeo, para obtener los resultados finales.

Predicción mediante variables estáticas

- Variables Numéricas: Coma días, Edad.
- Variables categóricas: APT Pronóstico, Coma Pronóstico, GCS Pronóstico, Sexo.
- Modelo: *Multi-Layer Perceptron.*

Los valores obtenidos en el primer estudio para el modelo elegido han sido de 0,2526 para RMSE y un valor de 0,5726 en correlación y se pueden visualizar en la figura 45.

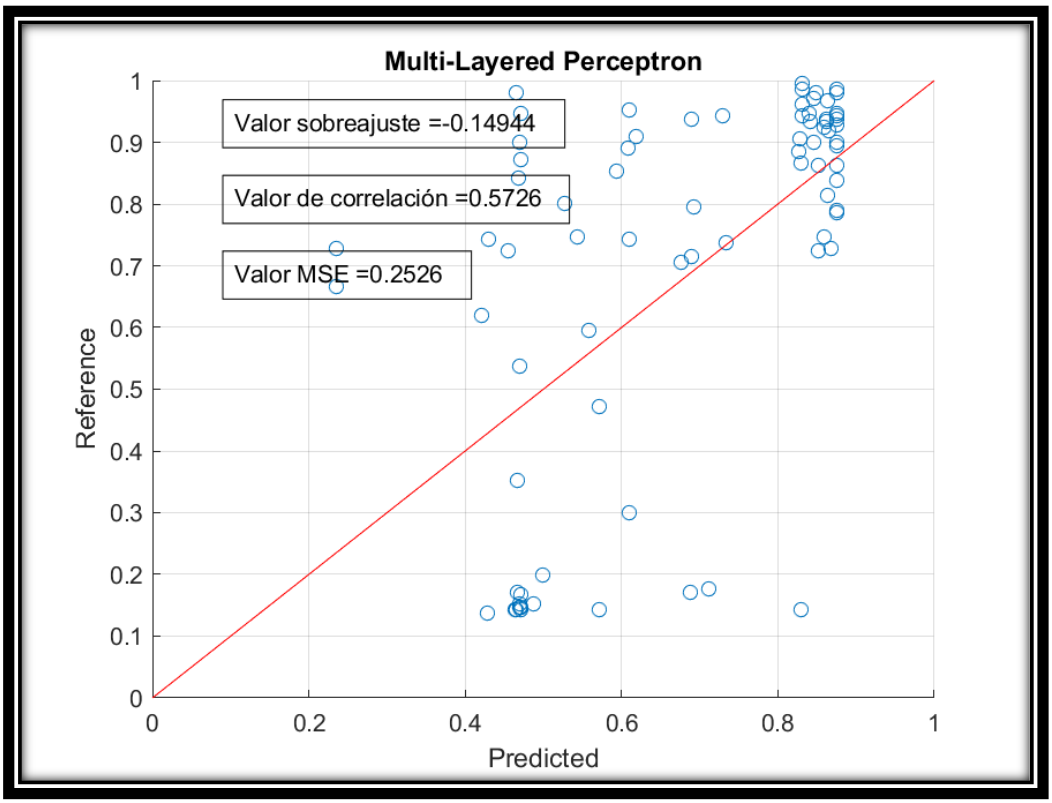


Figura 45. Estudio 1 - Testeo - MLP.

Predicción utilizando variables estáticas y dinámicas

- Variables numéricas: FIM cuidados Personales, FIM Movilidad, FIM locomoción, FIM Comunicación, FIM Conciencia del mundo exterior, FIM cognitiva, FIM control esfínteres, FAM.
- Variables categóricas: Clasificación neurológica.
- Modelo: *Support Vector Machine.*

Los valores obtenidos en el estudio 2 mostrados en la figura 46 para el modelo ganador han sido un valor de RMSE de 0,17727, con 0,833838 en correlación.

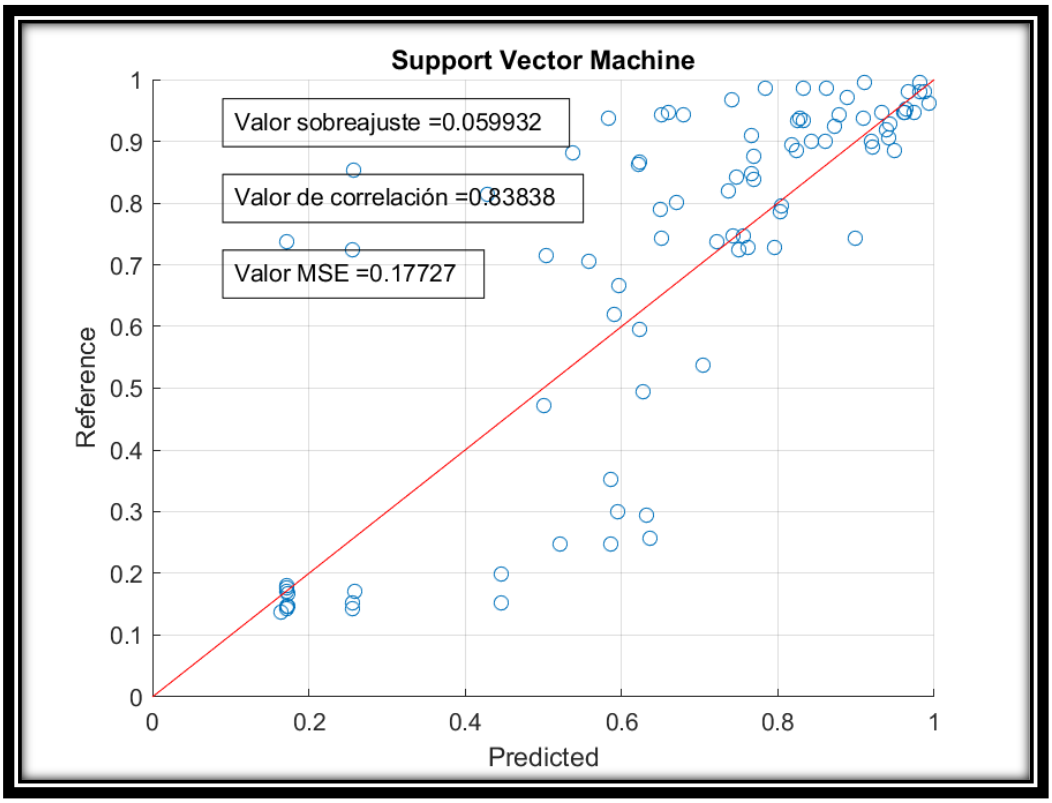


Figura 46. Estudio 2 - Testeo - SVM.

Predicción utilizando la evolución temporal de variables estáticas y dinámicas

- Variables Numéricas: FIM cuidados Personales, FIM Movilidad, FIM locomoción, FIM Comunicación, FIM cognitiva, FIM control esfínteres, FAM.
- Variables categóricas: APT Pronóstico.
- Modelo: *Long-Short Term Memory*.

Los resultados obtenidos en el tercer estudio para el modelo ganador se pueden visualizar en la siguiente tabla en función de las valoraciones utilizadas.

Multi-Layered Perceptron(MLP) - LSTM	RMSE	Correlación
1 valoraciones	0,17439	0.836442
2 valoraciones	0,112956	0.932005
3 valoraciones	0,101788	0.944297
4 valoraciones	0,097662	0.948600
5 valoraciones	0,092427	0.955654
6 valoraciones	0,099139	0.948145
7 valoraciones	0,089896	0.957684
MEDIA	0,10975114	0,93183243

Tabla 28. Estudio 3 - Testeo - LSTM.

Como se ve en la tabla valor medio del RMSE que consigue este modelo es de 0,10975114 con una correlación de 0,93183243 y la mejor predicción se realiza utilizando todas las valoraciones, con un valore de RMSE de 0,089896 y correlación de 0.957684. Este último resultado se puede comprobar en la figura 47:

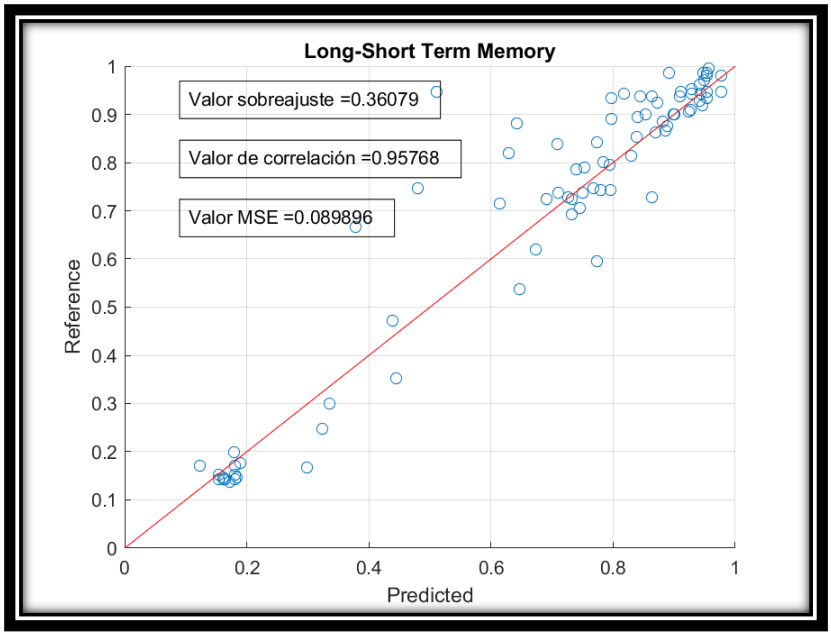


Figura 47. Estudio 3 - Testeo - LSTM**.**

5.5. Discusión e interpretación de los resultados.

A partir de los datos obtenidos en los diferentes estudios realizados en el punto *5. Experimentos*, se puede analizar los resultados para realizar una serie de conclusiones. En primer lugar, comparando los mejores modelos de cada estudio claramente podemos ver una mejora en los valores de RMSE y correlación. En el primer estudio donde solo se ha utilizado variables estáticas obtenemos un valor bastante elevado de RMSE, ya que la variable FAM que se está intentando predecir está normalizada sobre la unidad y una correlación bastante baja, por lo tanto, se puede decir que solamente con el uso de las variables estáticas se pueden generar modelos relativamente limitados que no son muy prometedores. En el estado del arte actual sobre el estudio de pacientes con traumatismos cerebrales, se ha utilizado únicamente el tipo de anterior de modelos considerando variables estáticas, obteniendo como resultados unos niveles bajos de predicción sobre la funcionalidad final que pueden alcanzar los pacientes. Al integrar en los modelos las variables dinámicas ya se puede apreciar un cambio notable en el nivel de predicción, pues los valores de RMSE se reducen aproximadamente un 30% y la correlación aumenta un 45%. Finalmente considerando la evolución temporal de estas variables se consigue reducir el RMSE en un 32% respecto el modelo anterior y la correlación aumenta un 12%.

Estudio	RMSE	Correlación
1. Variables estáticas	0,2526	0,5726
2.Variables estáticas y dinámicas	0,17727	0,833838
3.Evolución temporal de variables estáticas y dinámicas	0,10975114	0,93183243

Tabla 29. Resultados estudios.

Gracias al estudio de las variables predictivas se ha podido comprobar como las variables dinámicas aportan mucho valor a los modelos de predicción. Considerando las variables numéricas se ha visto que las dinámicas muestran una correlación mucho mayor que las estáticas estudiadas. En cuanto a las variables categóricas se ha visto tanto estáticas como dinámicas con bastante capacidad de discriminación, pero en los estudios se ha observado que generalmente suele ser más efectivo emplear las variables numéricas, sobre todo en los estudios que se utilizaban variables dinámicas, ya que había variables muy prometedoras como todas las que son tipo FIM o algunas otras como IB Total, DOS Total o DRs con valores de correlación mayores al 85%. Respecto a los modelos de *machine learning* estudiados, el modelo que permite un mayor nivel de predicción es el *Long-Short Term Memory*, puesto que puede evaluar el progreso de las variables en sus modelos para mejorar la predicción. Comparando los cuatro modelos restantes se ha visto como *Multi-Layer Perceptron* generalmente es más afectivo que los otros modelos

tanto utilizando variables estáticas y dinámicas como estáticas solamente, aunque en el segundo estudio el modelo que ha obtenido un menor valor de RMSE en las simulaciones ha sido *Support Vector Machine.*

6. CONCLUSIONES

Actualmente el personal sanitario implicado en la rehabilitación de los pacientes con TCE tiene que tomar decisiones clínicas complejas para determinar el tratamiento más adecuado para cada paciente. Muchas veces no se puede predecir la evolución del nivel funcional que podrá tener el paciente debido a que es un proceso bastante heterogéneo para cada individuo, además también existe la posibilidad de que el paciente no presente una mejoría durante la neurorrehabilitación, esto implica adaptar el tratamiento y un aumento de los costos del mismo.

La información existente basado en ensayos clínicos no tiene soluciones exactas de qué tratamientos o servicios de rehabilitación son más adecuados para cada paciente, además hay diversos factores que complican la práctica de ensayos clínicos aleatorios relacionados con TCE, ya que actualmente existen pocos estudios relacionados con este campo, los pacientes con TCE muchas veces no presentan unas capacidades cognitivas para dar el consentimiento y participar voluntariamente en nuevos estudios y el tiempo de rehabilitación de los pacientes suele ser más largo que la mayoría de proyectos de investigación. Además, todos los estudios que se han realizado para intentar predecir el nivel final funcional que puede lograr el paciente han sido basándose en variables estáticas obteniendo unos modelos de predicción bastante limitados.

El sistema que se ha creado en este proyecto permite generar y guardar modelos predictivos en función de las variables seleccionadas para los modelos utilizando técnicas de *machine learning*. En este sistema se utiliza una arquitectura de datos para el preprocesamiento y adecuación de los datos para poder entrenar correctamente a los modelos de inteligencia artificial. Gracias a esta herramienta se ha podido realizar un estudio sobre las variables disponibles en la base de datos facilitada por el hospital Vithas Valencia al Mar. Se ha analizado qué variables son las más recomendables para utilizarlas en los modelos de predicción, tras realizar el estudio se ha comprobado como en las variables numéricas estudiadas, las dinámicas eran mejores indicadores del nivel funcional final del paciente y por lo tanto más recomendables para añadir a los modelos predictivos que las variables estáticas, ya que presentaban valores de correlación mucho más elevados. Respecto a las variables categóricas, se ha podido ver que existen variables con buen nivel de discriminación tanto para las estáticas como para las dinámicas. No obstante, en los estudios realizados posteriormente se ha comprobado como las variables numéricas dinámicas eran las que aportaban más valor al modelo de predicción.

Comparando los diferentes estudios realizados, en el primero se ha utilizado variables estáticas simulando los modelos de los estudios existentes obteniendo unos niveles de predicción bastante

bajos con unos resultados para el mejor modelo de 0,2526 en RMSE y 0,5726 en correlación. Posteriormente se han integrado al siguiente estudio las variables dinámicas y se ha comprobado como el nivel de predicción mejora notablemente hasta alcanzar el valor de 0,177727 en RMSE y una correlación de 0,833838. Finalmente, mediante el modelo de *Machine Long-Short Term Memory* se han podido evaluar la evolución temporal de las variables estáticas y dinámicas de los modelos anteriores mejorando los niveles de predicción y consiguiendo un valor en RMSE de 0,10975114 y una correlación de 0,93183243.

Respecto a los diferentes modelos de *machine learning* utilizados, se ha confirmado que de los modelos utilizados para variables estáticas y dinámicas sin evolución temporal, el modelo que mejor se adapta de manera genérica es el *Multi-Layer Perceptron*, aunque en el segundo estudio el modelo con mejor resultado haya sido el *Support Vector Machine*. Finalmente, el modelo que permite un mayor nivel de predicción es *Long-Short Term Memory* ya que puede procesar los datos de las variables con evolución temporal.

7.BIOGRAFÍA

[1] Neurorrehabilitación traumatisco craneoencefálico. <https://neurorhb.com/traumatismo-craneoencefalico/> [Consulta 20 de Agosto de 2020]

[2] *Random Forest Algorithm.* <https://www.javatpoint.com/machine-learning-random-forest-algorithm.> [Consulta 20 de Agosto de 2020]

[3] Espinosa Zuñiga, J.(2020). *Application of Random Forest and XGBoost algorithms based on a credit card applications database.* <https://www.revistaingenieria.unam.mx/numeros/2020/v21n3-02.pdf> [Consulta: 01 de Septiembre de 2020].

[4] Livingston, F(2005). *Implementation of Breiman's Random Forest Machine Learning Algorithm*.<https://datajobs.com/data-science-repo/Random-Forest-[Frederick-Livingston].pdf>[Consulta: 01 de Septiembre de 2020].

[5] *Image. Multiple linear regression.* <https://es.mathworks.com/help/stats/regress.html>[Consulta: 25 de Agosto de 2020].

[6] Balasch i Bernat.M , Balasch i Aparici.S, Noé Sebastián,E, Dueñas Moscardó, L., Ferri Campos,J. y López Bueno,L.(2015*). Determining cut-off points in functional assessment scales in stroke*

[7] Oberholzer,M. y M. Müri,R(2019).*Neurorehabilitation of Traumatic Brain Injury (TBI): A Clinical Review.*

[8] Álvarez, J.(2016). *Machine Learning y Support Vector Machines* porque el tiempo es dinero. < https://www.analiticaweb.es/>[Consulta: 01 de Septiembre de 2020].

[9] Missinglink.ai. *Perceptrons and Multi-Layer Perceptron: The Artificial Neuron at the Core of Deep Learning*<https://missinglink.ai/guides/neural-network-concepts/perceptrons-and-multi-layer-perceptrons-the-artificial-neuron-at-the-core-of-deep-learning/> [Consulta: 26 de Agosto de 2020].

[10] Fundación Vithas(2020): Máster propio en Neurociencias: Cuidados Médico-Quirúrgicos y Rehabilitación del Paciente Neurológico. Tema: Pronóstico en la rehabilitación del Daño Cerebral Traumático.

[11] Jose Mariazo Alvarez. El perceptrón como neurona artificial. <http://blog.josemarianoalvarez.com/2018/06/10/el-perceptron-como-neurona-artificial/>[Consulta 26 Agosto de 2020].

[12] Colah.github.io. *Understanding LSTM Networks*<http://colah.github.io/posts/2015-08-Understanding-LSTMs/>[Consulta 26 Agosto de 2020].

[13] Pech May,F(2018). Procesamiento del lenguaje natural con *DeepLearning*. <https://rios.tecnm.mx/cdistribuido/recursos/DLScr/PLN.html>[Consulta 27 Agosto de 2020].

[14] Parámetros *Multi-Layer Perceptron.* <https://es.mathworks.com/help/deeplearning/ref/nnet.cnn.trainingoptionsadam.html>[Consulta 27 Agosto de 2020].

[15] Brownlee, J(2017). *Gentle introduction to the Adam Optimization Algorithm for Deep Learning*<https://machinelearningmastery.com/adam-optimization-algorithm-for-deep-learning/> [Consulta 28 Agosto de 2020].

[16] Galarynk,M(2018). *Understanding Boxplots*. <https://towardsdatascience.com/understanding-boxplots-5e2df7bcbd51> [Consulta 28 Agosto de 2020].

[17] Roman,V(2019).*Machine Learning* Supervisado: Fundamentos de la Regresión Lineal. <https://medium.com/datos-y-ciencia/machine-learning-supervisado-fundamentos-de-la-regresi%C3%B3n-lineal-bbcb07fe7fd> [Consulta 30 Agosto de 2020].

[18] Hiperparámetros *Long-Short Term Memory*. <https://es.mathworks.com/help/deeplearning/ref/trainlm.html?s_tid=srchtitle>[Consulta 30 Agosto de 2020].

[19] *Big Data Image*.<https://www.nonteek.com/en/dna-storage-solution-to-big-data-in-a-strand/> [Consulta 20 Agosto de 2020].

[20] *One Hot Encoding* Image.<https://medium.com/@michaeldelsole/what-is-one-hot encoding-and-how-to-do-it-f0ae272f1179>:[Consulta 25 Agosto de 2020].