UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Dpto. de Estadística e Investigación Operativa Aplicadas y Calidad

Análisis multivariante del movimiento de la mirada y de la cabeza en conversaciones con avatares virtuales y desarrollo de modelos para la detección de síntomas depresivos.

Trabajo Fin de Máster

Máster Universitario en Ingeniería de Análisis de Datos, Mejora de Procesos y Toma de Decisiones

AUTOR/A: Serrano Aladrén, Víctor

Tutor/a: Marín Morales, Javier

Director/a Experimental: Altozano Fernández, Alberto

CURSO ACADÉMICO: 2023/2024

Resumen

En este TFM se explora una manera de poder identificar a personas con síntomas depresivos de forma objetiva con un procedimiento estandarizado. En concreto se analizan datos del movimiento de la mirada y la cabeza de unas personas durante conversaciones con avatares virtuales emocionales. A partir de ellos se desarrollan y comparan diversos modelos predictivos tanto de clasificación de personas según si tienen o no síntomas depresivos como también otros modelos que estimen el grado de estos síntomas según el índice PHQ.

En primer lugar, se realiza un análisis multivariante empleando la técnica de componentes principales (PCA) de las 104 características extraídas en 514 conversaciones de 92 personas distintas. Este análisis permite identificar como se relacionan las variables y agruparlas en función de estas relaciones. También se explora en el espacio de variables latentes si las conversaciones se agrupan en función de los individuos o en función del grado de sus síntomas depresivos. Además, se estudia si el estado de ánimo, el género o la ropa del avatar virtual influyen en el comportamiento de los individuos durante las conversaciones.

En segundo lugar, se desarrollan modelos tanto para clasificación en función de la existencia de síntomas como predictivos del valor de PHQ empleando técnicas de minería de datos sobre las características extraídas de las conversaciones. Los modelos de clasificación se comparan entre ellos empleando la precisión, sensibilidad y especificidad mientras que los modelos de predicción del PHQ se comparan con el error cuadrático medio.

En último lugar se aplican diversos modelos de Deep learning directamente sobre las series temporales de todas las conversaciones en lugar de usar las características que resumían las conversaciones en 514 valores. Se ajustan modelos para los dos objetivos (clasificación y predicción de PHQ) y se comparan entre ellos para determinar cuáles son más efectivos y también respecto a los obtenidos con técnicas de minería de datos.

Palabras clave: Seguimiento mirada, Análisis de datos, PCA, Minería de datos, Aprendizaje profundo, Avatares virtuales emocionales, PHQ, Autoevaluación de síntomas depresivos.

Resum

En aquest TFM s'explora una manera de poder identificar persones amb símptomes depressius de manera objectiva amb un procediment estandarditzat. En concret, s'analitzen dades del moviment de la mirada i el cap d'unes persones durant converses amb avatars virtuals emocionals. A partir d'ells es desenvolupen i comparen diversos models predictius tant de classificació de persones segons si tenen símptomes depressius o no com també altres models que estimin el grau d'aquests símptomes segons l'índex PHQ.

En primer lloc, es fa una anàlisi multivariant emprant la tècnica de components principals (PCA) de les 104 característiques extretes en 514 converses de 92 persones diferents. Aquesta anàlisi permet identificar com es relacionen les variables i agrupar-les segons aquestes relacions. També s'explora a l'espai de variables latents si les converses s'agrupen en funció dels individus o en funció del grau dels símptomes depressius. A més, s'estudia si l'estat d'ànim, el gènere o la roba de l'avatar virtual influeixen en el comportament dels individus durant les converses.

En segon lloc, es desenvolupen models tant per a classificació en funció de l'existència de símptomes com a predictius del valor de PHQ emprant tècniques de mineria de dades sobre les característiques extretes de les converses. Els models de classificació es comparen entre ells emprant la precisió, sensibilitat i especificitat, mentre que els models de predicció del PHQ es comparen amb l'error quadràtic mitjà.

En últim lloc, s'apliquen diversos models de Deep Learning directament sobre les sèries temporals de totes les converses en lloc d'usar les característiques que resumien les converses en 514 valors. S'ajusten models per als dos objectius (classificació i predicció de PHQ) i es comparen entre ells per determinar quins són més efectius i també respecte dels obtinguts amb tècniques de mineria de dades.

Paraules clau: Seguiment mirada, Anàlisi de dades, PCA, Mineria de dades, Aprenentatge profund, Avatars virtuals emocionals, PHQ, Autoavaluació de símptomes depressius

Abstract

This TFM explores a way to objectively identify people with depressive symptoms using a standardized procedure. Data on people's gaze and head movements during conversations with emotional virtual avatars are analyzed. From them, several predictive models are developed and compared, both for classifying people according to whether or not they have depressive symptoms as well as other models that estimate the degree of these symptoms according to the PHQ index.

First, a multivariate analysis using principal components analysis (PCA) is performed on the 104 features extracted from 514 conversations with 92 different individuals. This analysis allows us to identify how the variables are related and to group them on the basis of these relationships. It the latent variable space, it is also examined whether the conversations are grouped according to the individuals or according to the level of their depressive symptoms. In addition, it is investigated whether the mood, gender or clothing of the virtual avatar influence the behavior of individuals during conversations.

Second, models are developed for both classification based on the presence of symptoms and prediction of the PHQ scores using data mining techniques on the features extracted from the conversations. The classification models are compared with each other in terms of accuracy, sensitivity and specificity while the PHQ prediction models are compared with the mean square error.

Finally, several Deep Learning models are applied directly on the time series of all conversations instead of using the features that summarized the conversations in 514 values. Different models are adjusted for the two objectives (PHQ classification and prediction) and compared between them to determine which are more effective and also with respect to those obtained with data mining techniques.

Keywords: Eye Tracking, Data Analysis, PCA, Data Mining, Deep Learning, Emotional Virtual Avatars, PHQ, Self Assessment for Depression.

Agradecimientos

A mi tutor Javier Marín Morales y a Albert Altozano Fernández por los consejos para enfocar el trabajo y la ayuda a la hora de trabajar con los datos.

A la fundación valgrAI (Valencian Graduate School and Research Network of Artificial Intelligence) por ayudarme en mi formación al concederme una de sus becas para estudios de posgrado.

Al profesorado del Máster que ha ampliado los horizontes de mi conocimiento con su gran dedicación a la enseñanza.

A mi familia y amigos por el apoyo en la vuelta a la UPV y el cambio en mi rumbo laboral.

Índice

Índice de tablas

Índice de figuras

1. Introducción

La depresión es una enfermedad muy presente en nuestra sociedad y cuya presencia ha aumentado en los últimos años. Como otras enfermedades mentales su diagnosis es complicada. En la actualidad la diagnosis de la depresión la realizan profesionales al evaluar el paciente. Esto supone además de coste elevado de recursos un cierto componente de subjetividad al realizar la evaluación. Además, existe cierto estigma entorno a la salud mental que provoca que una gran parte de personas eviten acudir a especialistas de la salud mental.

En este sentido se han desarrollado avatares virtuales emocionales [1] que son capaces de establecer una conversación con personas y representar emociones. Estos avatares presentan distintos estados de ánimo y personalidades los cuales pueden expresar gracias al uso de Large Language Model. Con ellos aparece la posibilidad de desarrollar un método para el diagnosis de los síntomas asociados a la depresión automatizado. Esto sería posible mediante la recolección de datos durante las conversaciones que tengan los pacientes con los avatares.

Entre todas las cosas que se podrían medir durante las conversaciones este trabajo se centra en el movimiento de la cabeza y de los ojos. Se ha realizado un experimento con 92 individuos los cuales han tenido diversas conversaciones cada uno relacionadas con su situación y estado actual. Los avatares con los que se realizaban las conversaciones aparecían en una pantalla a tamaño real. Con unas gafas de realidad aumentada se realizaba la medición del movimiento de la cabeza y el lugar donde se fijaban los ojos cada 4 milisegundos. A partir de estas series temporales de los distintos sensores es posible sintetizar la información de cada conversación en 102 variables o características extraídas para facilitar su tratamiento y exploración.

Para cuantificar los síntomas depresivos de cada individuo se opta por realizarles antes de comenzar las conversaciones un test PHQ [2]. Con esta prueba una persona es capaz de realizar un diagnóstico de los síntomas de depresión que autopercibe que tiene y su grado de intensidad. Se obtiene una puntuación cuyo valor va desde 0 hasta 27, siendo mayor cuanto más síntomas estén presentes y mayor sea su gravedad. Habitualmente se considera que una persona con menor índice PHQ que 9 no tiene síntomas depresivos y forma parte del grupo control. Aunque también es posible establecer 3 grupos de individuos: los que tienen PHQ menor a 5 no tienen síntomas, entre 5 y 14 tienen unos síntomas moderados y a partir de 14 tienen síntomas depresivos severos.

Estudios iniciales de las características extraídas de las conversaciones de este experimento [3] muestran ciertas diferencias entre las personas con y sin síntomas depresivos en el movimiento de su mirada durante estas conversaciones. En este trabajo se continuará con el análisis tanto de las características extraídas como de las series temporales originales. Los análisis iniciales tienen un carácter univariante mientras que los análisis de este trabajo pasan a tener un enfoque multivariante. Con ello se pretende aumentar su comprensión y determinar si es posible emplearlos para realizar un diagnóstico inicial de personas que muestren síntomas depresivos.

El primer objetivo es analizar las características extraídas de las conversaciones mediante un PCA. En él se podrán detectar las relaciones y patrones entre las 102 variables formadas por las características extraídas recopiladas de las conversaciones. También se podrán detectar conversaciones anómalas, algo útil para mejorar los avatares virtuales.

El segundo objetivo es desarrollar modelos de minería de datos a partir de estas características extraídas que sean capaces de determinar el grado de los síntomas de los individuos. Por un lado, se desarrollan modelos que predigan el valor numérico de PHQ de los usuarios. Entre ellos se encuentra un modelo PLS, el cual se podrá interpretar de forma similar al PCA usado en el objetivo anterior. Por otro lado, se desarrollan modelos que sean capaces de clasificar los usuarios en grupos en función de sus síntomas. Se estudiará la división en 2 grupos o en 3 grupos establecidos previamente en función del valor del PHQ.

El tercer objetivo es crear modelos de Deep Learning que usen directamente las series temporales originales para identificar los usuarios con síntomas depresivos. De forma similar al objetivo anterior se prueban tanto modelos que predigan el valor numérico del PHQ como modelos que clasifiquen a los individuos. En este caso debido al mayor coste computacional solamente se estudiarán modelos que clasifiquen en 2 grupos: los que no y los que sí tienen síntomas depresivos.

Para llevar a cabo estos objetivos en la primera sección del trabajo se describirá la base de datos explicando tanto las series temporales de partida como las características extraídas de ellas. En la siguiente sección se explica la metodología empleada para el tratamiento y el análisis de los datos, la creación de los modelos y su validación. Posteriormente se exponen los resultados obtenidos del análisis y de los modelos. Finalmente se extraerán las conclusiones del trabajo.

2. Descripción de la base de datos

La base de datos está formada por los datos de 514 conversaciones de 92 personas a las que previamente se les ha asignado una puntuación PHQ mediante una autoevaluación que indica la cantidad de síntomas relacionados con la depresión que tienen. Hay una variable categórica que identifica la persona y una cuantitativa indica su puntuación PHQ que será la variable observada. La información de la persona a la que pertenece cada conversación solamente se usará para que las conversaciones de una persona solo se usen para entrenar los modelos o para validarlos.

Estas conversaciones se han realizado con avatares de distinto género, con distinta ropa y con 4 emociones distintas además de un estado emocional neutro. Por lo tanto, hay 3 variables categóricas para describir los avatares. Se ha buscado que los avatares sean lo más variados, intentando que todos los tipos de avatares estén igualmente representados. En cuanto a la emoción de los avatares, hay 170 conversaciones con avatares cuya emoción es neutra. Esta cantidad es aproximadamente el doble que el resto de las emociones, las cuales están muy parejas: 87 felices, 86 relajados, 86 tristes y 85 enfadados. En 261 conversaciones el avatar se viste de forma casual y en 253 de forma semiformal, por lo que en la ropa también están compensadas las conversaciones. Y por último también están equilibradas respecto al género del avatar: 258 tienen género masculino y 256 tienen género femenino.

Por conversación se ha recopilado cada 4 milisegundos la posición de la mirada, si los ojos están cerrados y el movimiento de la cabeza. A partir de esta información se han creado las siguientes variables para cada instante de tiempo:

- 1 variable binaria que indica si los ojos están cerrados durante parpadeo
- 3 variables numéricas para la velocidad angular en los 3 ejes
- 3 variables numéricas para la aceleración lineal en los 3 ejes
- 2 variables numéricas para los grados de cabeceo o pitch y alabeo o roll
- 1 variable binaria indica si en ese instante está sucediendo una fijación
- 1 variable binaria que indica si esta fijación se encuentra en el avatar
- 2 variables numéricas indican los pixeles medios en X e Y en el cuerpo del avatar donde ha sucedido la fijación en caso de que esté sucediendo
- 1 variable numérica indica la distancia al ojo de la fijación en curso
- 9 variables binarias indican a que parte del cuerpo se dirige la observación: cabeza, brazo izquierdo, brazo derecho, torso, piernas, pies, ojos, nariz y boca

La información de las series temporales será usada en los modelos de Deep Learning, pero para los modelos de minería de datos se extraerán características para cada conversación. Cada conversación se sintetiza en 102 variables cuantitativas con un valor por conversación que registran el movimiento de los ojos y cabeza durante las conversaciones. Esto incluye parpadeos, sacadas y las fijaciones a cada una de las partes del cuerpo. Gran parte de estas variables son medias, medianas y desviaciones típicas que agrupan las mediciones realizadas a lo largo de la conversación. También está la duración de la conversación y otros conteos que se almacenan directamente.

Finalmente cabe destacar que se ha intentado que haya un número similar de conversaciones de individuos con síntomas depresivos respecto a los de control. Si consideramos a los que tienen un PHQ menor a 9 como individuos sin síntomas, hay 298 usuarios control y 216 con síntomas depresivos. Agrupando los individuos en 3 grupos se observa que hay 224 conversaciones de usuarios sin síntomas, 209 con síntomas moderados y 81 con síntomas severos. En el histograma de la se ve como el PHQ asociado a las conversaciones analizadas está bastante repartido, intentando que la muestra sea lo más balanceada posible.

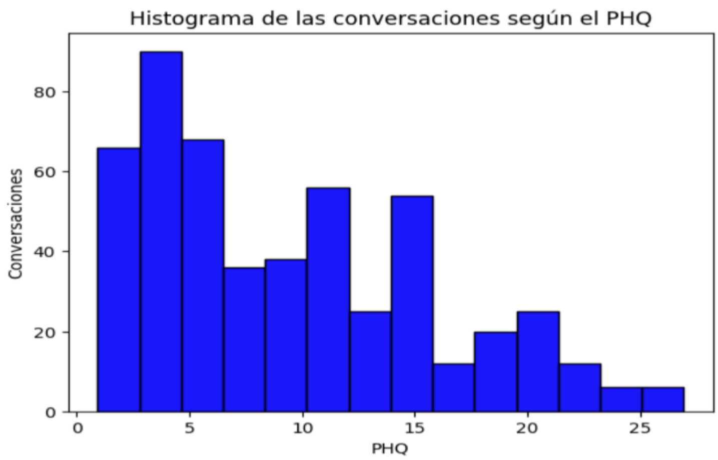

Figura 1. Histograma del PHQ asociado al individuo de cada una de las conversaciones del estudio

3. Metodología

A continuación, se detallan las herramientas utilizadas durante el trabajo para el procesamiento de los datos, su análisis y la creación de los modelos. Además se detallan los pasos necesarios para poder usar las variables formadas por las características extraídas de las series temporales.

3.1 Herramientas de software y librerías utilizadas

Para la realización de este trabajo se han empleado diversas herramientas de software. Se ha empleado Python en su versión 3.12.3 para todo salvo para el análisis de componentes principales y la interpretación del modelo PLS que se ha empleado el programa Aspen Pro MV 10.0. Esta elección se ha realizado debido a la rapidez a la hora de realizar ajustes en los parámetros del PCA y PLS y mostrarlos gráficamente. De forma adicional se ha programado parte de estos gráficos en Python para comprobar que los resultados eran consistentes en ambos software.

En Python se emplean una serie de librerías generales para el manejo de los datos las cuales son Numpy [4] y pandas [5]. Para graficar estos datos se emplean las librerías Matplotlib [6] y Seaborn [7]. En la imputación de las características extraídas de las conversaciones y en el desarrollo de los modelos de minería de datos se emplean 3 librerías.

De la librería Sklearn [8] se emplean las funciones asociadas con la imputación de los datos faltantes y con los modelos Random Forest, SVM, SVR, KNN, PLS, Gaussian Naive Bayes y Gaussian Process Regression. Los modelos Random Forest, KNN y PLS cuentan con funciones distintas según si el modelo tiene como objetivo predecir un valor numérico o clasificar. En cambio, modelos SVM y Gaussian Process Regression solo sirven para predecir un valor numérico mientras que los modelos SVR y Gaussian Naive Bayes solo pueden ser usados para clasificar.

Las otras dos librerías necesarias para los modelos que emplean las características extraídas son Xgboost [9] y Tensorflow [10]. Con la primera se implementa el modelo Gradient boosted trees y con la segunda el modelo de red neuronal Perceptrón Multicapa (MLP). Ambos modelos son válidos tanto para predicción de valores numéricos como para clasificación.

La implementación de los modelos de Deep Learning empleados sobre las series temporales completas se realiza con funciones de la librería Tsai [11]. Esta librería es muy completa e incluye una gran cantidad de modelos aptos para trabajar con series temporales. La base de Tsai es el popular framework de Deep Learning PyTorch [12]. Gran parte de los modelos de esta librería permiten la activación de dropout [13] también conocido como dilución, el cual permite evitar el sobreajuste en determinadas situaciones.

Los modelos de Deep Learning que se han probado son: MiniRocket [14], Transformer [15], LSTM [16], GRU [17], FCN [18] [19], Residual Network [20] e Inception Time [21]. Las funciones que implementa Tsai permiten adaptar estos modelos para que su salida pueda usarse para clasificar en distintas clases o para realizar la estimación por regresión de un valor numérico. Estos modelos de Deep Learning contienen un amplio abanico de arquitecturas adaptadas para su uso con series temporales multivariantes.

Al grupo de las redes neuronales recurrentes (RNN) pertenecen los modelos LSTM (Long Short-Term memory) y GRU (Gated Recurrent Unit). El grupo más numeroso es el de las redes neuronales convolucionales (CNN), al cual pertenecen los modelos FCN (Full Convolutional Network), Residual Network e Inception Time. Al margen de estos dos grupos se encuentra el modelo MiniRocket, el cual es una mejora del modelo ROCKET en el que se transforman los datos usando un gran conjunto aleatorio de kernels convolucionales. Con MiniRocket en lugar de aleatorios estos kernels convolucionales son fijos y muchos menos. Con ellos se consigue un modelo casi determinista que requiere menor potencia de cálculo con resultados similares.

En cuanto al modelo de tipo Transformer empleado en esta librería también está adaptado a las series temporales multivariantes. De la estructura base de un modelo de Transformer encoder se prescinde de la parte del decoder. Como la estructura inicial es insensible al orden de los datos de entrada, algo fundamental en las series temporales, se añaden también encoders posicionales. Estos encoders posicionales son totalmente ajustables en lugar de ser deterministas. Para poder usarlo en regresión o clasificación al modelo se le añade una capa lineal final cuyo tamaño depende de la cantidad de clases o valores a estimar.

Por último, cabe destacar que en la librería Sklearn [8] hay una función que permite agrupar las conversaciones en 5 grupos para poder realizar la validación cruzada correctamente. Es imprescindible que todas las conversaciones de un individuo estén en el mismo grupo. Con ello se evita que el modelo se entrene y se valide con conversaciones de un mismo usuario. Si esta condición no se cumpliese se incumpliría el principio de independencia de los datos de validación. En modelos complejos esto podría causar que se distinguiese entre los distintos individuos de la muestra y su puntuación PHQ en lugar de poder predecir el PHQ de un nuevo individuo del que no se haya tenido información previa.

Durante la validación cruzada de todos los modelos se seleccionan 4 de los 5 grupos para el entrenamiento del modelo. Posteriormente se validan las predicciones del modelo sobre el quinto grupo. Este proceso se repite 5 veces, realizando la validación con todos los grupos. Los grupos están formados por las mismas conversaciones en todos los modelos para poder compararlos de forma consistente.

3.2 Transformación e imputación de las características extraídas

En la base de datos hay 514 conversaciones de 92 usuarios distintos. La variable “user” identifica a cada uno de los individuos, pero no será usada para la creación de los modelos, solamente como información adicional en la exploración. De ellos hay 83 individuos que tienen 6 conversaciones con avatares en distinto estado de ánimo: 2 con avatares neutros y una con el resto de los avatares (enfadado o anger, alegre o happy, triste o sad y relajado o relax). El resto de los individuos tienen menos conversaciones y no necesariamente con avatares de todos los estados de ánimo.

La información del estado de ánimo de los avatares se encuentra en la variable categórica “emotion”. Para poder usar la variable al haber 5 emociones es necesario crear 5 variables binarias que representan si el avatar tiene cada una de las emociones empleando la técnica de One Hot Encoding. Además, la variable categórica “type” representa si el avatar va vestido de forma semiformal (valor 1) o casual (valor 0). El género del avatar es hombre si la variable “gender” es 0 y mujer si tiene el valor 1.

En cuanto a los valores faltantes en la base de datos hay 40 conversaciones con 2 valores faltantes y una conversación con 3. La mayoría están en la media y desviación típica de la distancia de la mirada a los ojos del avatar.

Tabla 1. Valores faltantes por variable en los datos

Variable	Número faltantes	Porcentaje faltantes
dist_to_eye_mean	40	7.8
dist_to_eye_std	40	7.8
blinks_mean	1	0.2
blinks_med	1	0.2
blinks_std	1	0.2

Como no representan un valor elevado porcentualmente (20% o más) en ninguna variable ni observación se podrán imputar. Se opta por esta solución y se emplea el método de imputación iterativo de la librería “scikit learn” que modela cada variable con valor faltante como función del resto de variables con un algoritmo iterativo round-robin.

Al tener la información de los síntomas depresivos del individuo que ha realizado cada conversación en valor numérico es necesario crear dos variables categóricas adicionales para usarla en los modelos que tienen como objetivo clasificar los individuos. Las dos variables contienen a que grupo pertenece el usuario según en cuantos grupos se dividen estos. Una contendrá la información en caso de que se separen los individuos en dos grupos: las personas con PHQ menor o igual a 9 se considera que no tienen síntomas depresivos (“control”) y el resto que sí los sufren (“SDS”). La otra variable se usará en caso de que se dividan los individuos en 3 grupos: con PHQ menor a 5 no sufren síntomas depresivos (“control”), PHQ mayor a 5 y menor que 14 se considera que sus síntomas depresivos son moderados (“mild”) y el resto tienen síntomas severos (“severe”).

4. Resultado

En este apartado se presentan los resultados obtenidos en el trabajo. En primer lugar, se presenta el análisis de componentes principales en el que se estudian las relaciones entre las variables y la existencia de conversaciones anómalas. En segundo lugar, se presenta la interpretación del modelo PLS que predice el valor numérico de PHQ, la cual se asemeja al análisis de componentes principales al ser también un modelo basado en variables latentes.

En tercer lugar, se muestran los resultados de todos los modelos de minería de datos planteados sobre las características extraídas de las conversaciones. Estos modelos tendrán como objetivo la predicción del valor numérico de PHQ, la clasificación en 2 y 3 clases de los individuos en función de sus síntomas depresivos. En cuarto lugar, se comenta el ajuste y los resultados de los modelos de Deep Learning aplicados directamente a las series temporales de los sensores capturadas durante las conversaciones para clasificar los individuos en dos clases. Por último, se resumen los resultados de las 7 tipologías de modelos Deep Learning empleadas anteriormente para predecir el valor numérico del PHQ.

4.1. Análisis de componentes principales de las características

Con el análisis de componentes principales se estudia la variabilidad de las conversaciones en el espacio de todas las características extraídas del movimiento de la mirada y cabeza durante las conversaciones. Del PCA se excluye la información del tipo de avatar y su estado de ánimo, así como la puntuación PHQ correspondiente al usuario que realiza cada conversación. Todas las variables incluidas son numéricas y se estandarizan con la misma importancia cada una (media cero y desviación típica de 1).

La cantidad de componentes principales empleadas debe alcanzar un equilibrio que evite que se modele el ruido de los datos y se incluya la mayor cantidad de información posible. El criterio empleado para la selección consiste en seleccionar la cantidad de componentes que produzcan el modelo con mayor Q cuadrado. Concretamente se seleccionan 13 componentes principales. En la Figura 2 se muestra el R cuadrado y el Q cuadrado obtenido en el PCA de 13 componentes principales más importantes. Con estas componentes se consigue un R cuadrado acumulado de 0.68 y un Q cuadrado acumulado de 0.62.

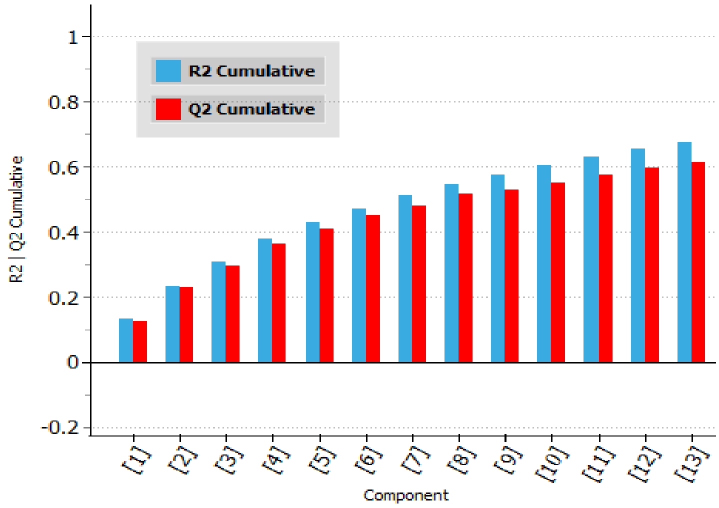

Figura 2. R2 acumulado y Q2 acumulado del PCA con todas las observaciones

Sobre este modelo de 13 componentes principales se realiza el análisis inicial. El primer paso para realizar el análisis es eliminar las conversaciones anómalas detectadas en la primera iteración del PCA. Con el SPE o error de predicción al cuadrado en el espacio de las variables se pueden detectar observaciones atípicas que están alejadas del espacio de proyección. Las variables en estas conversaciones no siguen la estructura de correlación habitual en la mayoría de las conversaciones, la cual intenta modelar el PCA. En la Figura 3 se distingue como la conversación 80 es claramente la más atípica, seguida de las conversaciones 386, 426 y 94. Todas estas conversaciones tienen valor por encima del percentil 99, lo que hace muy improbable que estén bien explicadas por el modelo.

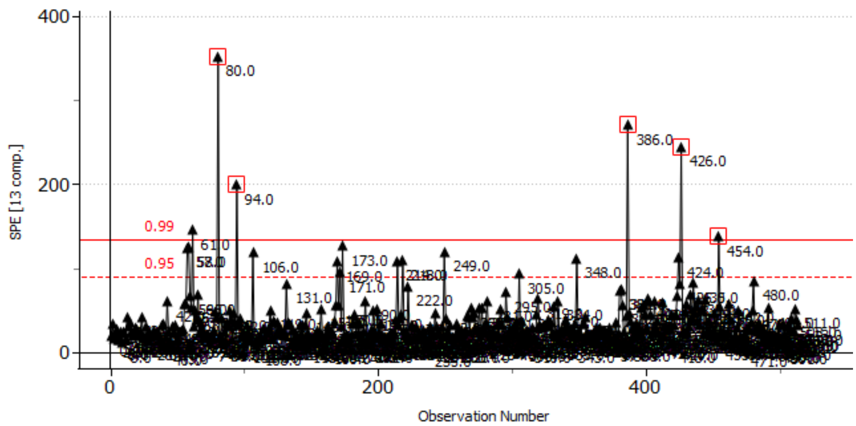

Figura 3. Error cuadrado de predicción del PCA con todas las observaciones

También es posible detectar conversaciones que cuyos valores extremos tienen mucha influencia en el ajuste del modelo. Algunos de estas conversaciones pueden tener poco error respecto al modelo al seguir la estructura de correlación habitual pero tener valores muy elevados o muy bajos en algunas variables. En la Figura 4 se observa como la conversación 426 no es muy influyente en el modelo al no tener una distancia de Mahalanobis (T2) muy elevada. Sin embargo, la 454 sí que lo es, teniendo un SPE elevado por encima del percentil 99 por lo que también se eliminará.

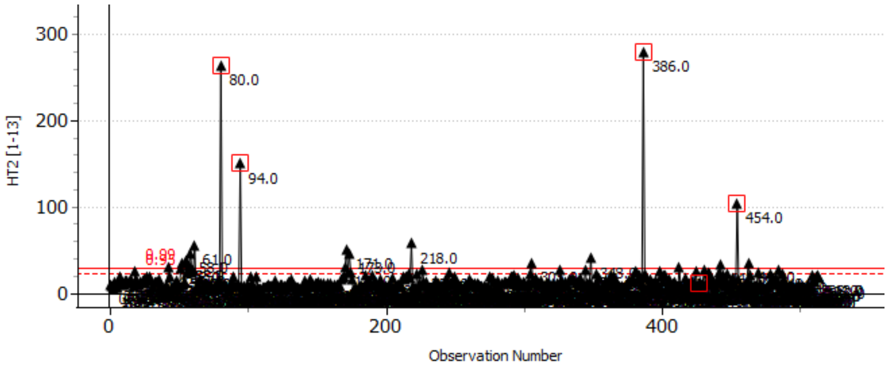

Figura 4. Distancia al cuadrado de Mahalanobis (T2) del PCA con todas las observaciones

Resulta interesante analizar el motivo por el que las observaciones son atípicas viendo en que variables tienen los valores anómalos que provocan los valores anómalos en las componentes. Analizar lo sucedido podría aportar información que mejorase los avatares virtuales o detectar sucesos extraños durante la toma de los datos que se habían pasado por alto y que dificultase la realización de los modelos. También se podría detectar si un usuario en particular tiene un comportamiento especialmente errático.

Respecto a la conversación 80 se muestran las contribuciones de las variables a los residuos anormales en la Figura 5.

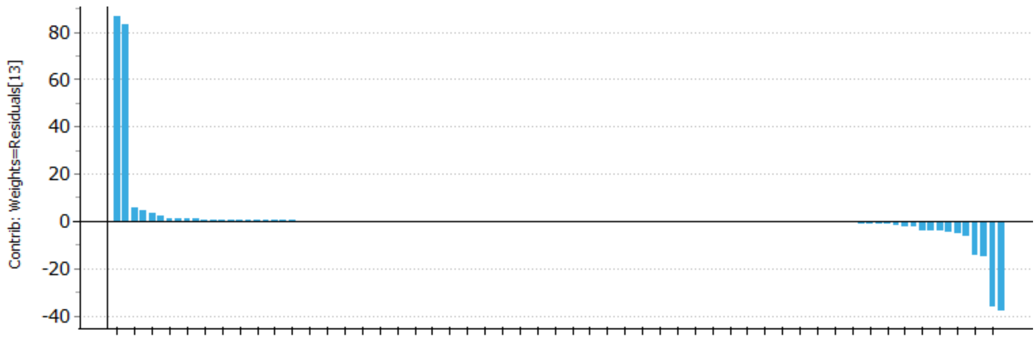

Figura 5. Contribuciones de las variables en el error de predicción para la conversación 80

En cuanto a las contribuciones positivas destacan la desviación típica y media de los parpadeos con una contribución de 86 y 83 respectivamente. Esto indica que esta conversación tiene valores anormalmente altos en estas variables. También destacan contribuciones negativas elevadas para las variables “sac_t1_std”, “sac_time_std”, “sac_t0_mean” y “sac_t0_median”.

Poniendo los valores de las dos variables relacionadas con los parpadeos de todas las conversaciones en un gráfico de dispersión (Figura 6) se aprecia como sus valores son anormalmente altos en esta observación (resaltada con el cuadrado rojo) respecto al resto.

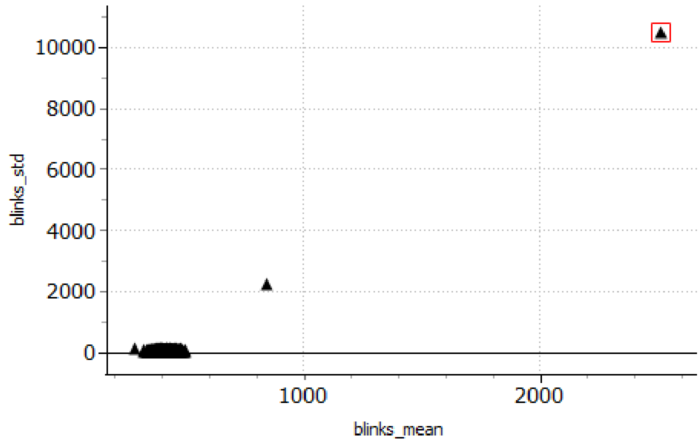

Figura 6. Variable "blinks_mean" respecto a "blinks_std" resaltando la observación 80

En la conversación 386 el valor de las contribuciones (Figura 7) de “sac_time_std” y “sac_t1_std” está cercana a 150. Otras variables con contribuciones elevadas son “sac_t1_mean”, “sac_time_mean” también están relacionadas con las sacadas.

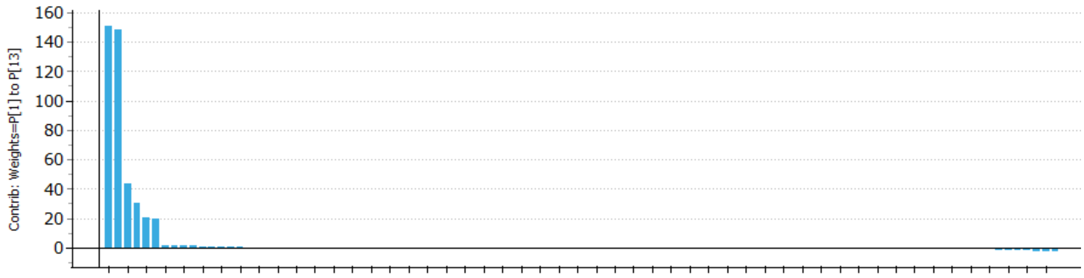

Figura 7. Contribuciones de las variables en el error de predicción para la conversación 386

Además, el tiempo de parpadeo medios y su desviación típica tienen también contribuciones cercanas a 20. De hecho, en la Figura 6 el segundo punto más anómalo es el correspondiente a la conversación 386.

En cuanto a la conversación 94 hay claramente 3 contribuciones anormalmente altas que se pueden ver en la Figura 8. Estas contribuciones son de las variables relacionadas con el brazo derecho “fixcount_arm_right”, “fixtime_arm_right” y visits_arm_right”.

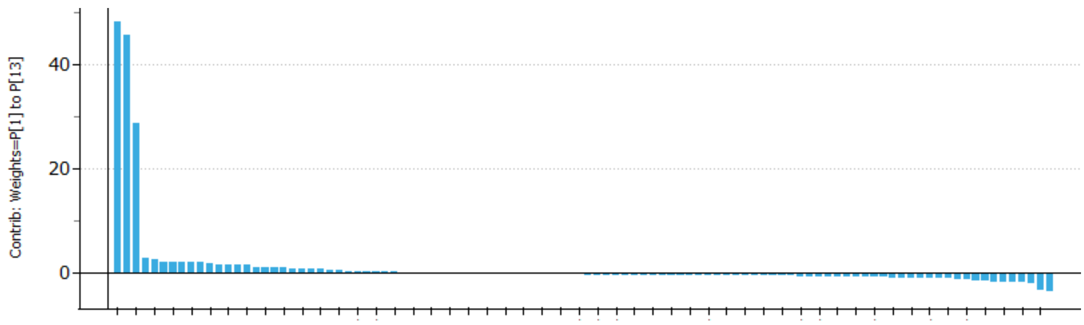

Figura 8. Contribuciones de las variables en el error de predicción para la conversación 94

En la conversación 454 las variables que tienen una contribución elevada (Figura 9) están relacionadas con los pies: “fixtime_foots”, “fixcount_foots” y “visits_foots”.

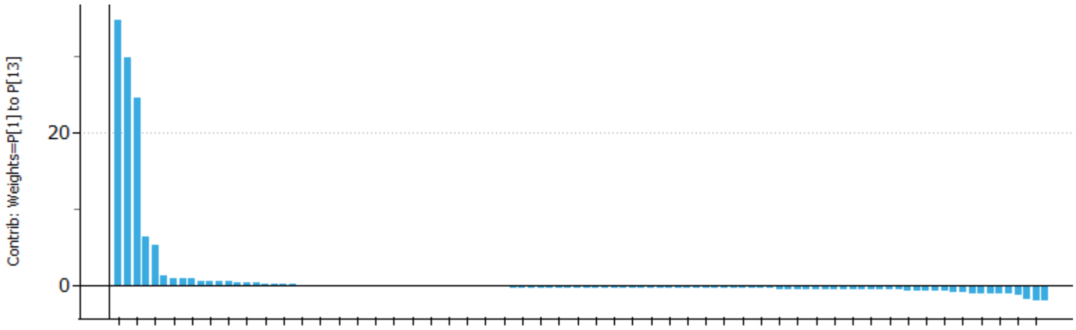

Figura 9. Contribuciones de las variables en el error de predicción para la conversación 454

De forma iterativa se van eliminando las conversaciones anormales hasta que no haya ninguna con excesivo error SPE, especialmente aquellas que tengan mucha influencia en el modelo. Finalmente, las conversaciones descartadas son las siguientes: 57, 58, 61, 80, 94, 106, 169, 214, 218, 249, 386, 424, 426, 454 y 480. Tras su eliminación en la Figura 10 y la Figura 11 ninguna conversación destaca por tener un valor elevado respecto al resto de conversaciones. Ninguna de las conversaciones supera el valor del 30% del SPE asociado al percentil 99, que en este caso sería 80. De estas conversaciones cabe destacar que hay 3 del sujeto “USER_15_CB2”. En cuanto al estado de ánimo de los avatares, 7 de las 12 conversaciones descartadas restantes son con avatar neutro.

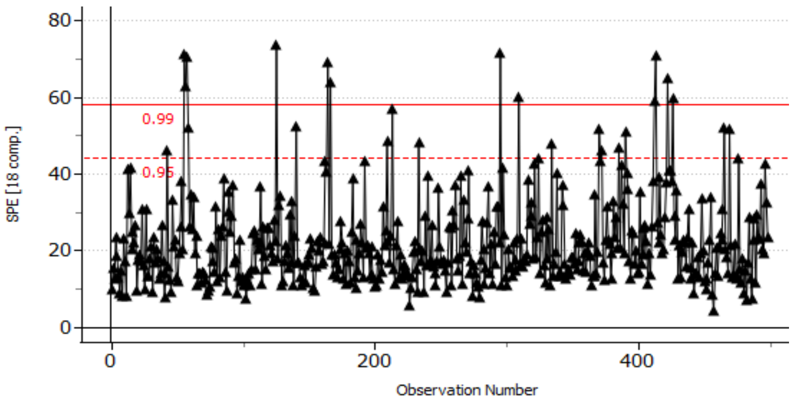

Figura 10. Error cuadrado de predicción del PCA tras eliminar las conversaciones anómalas

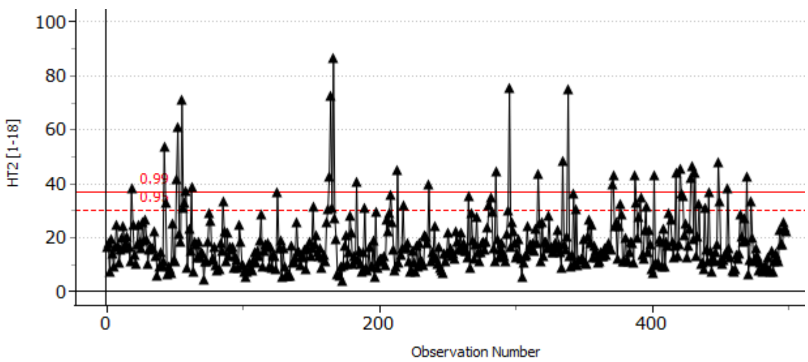

Figura 11. Distancia al cuadrado de Mahalanobis (T2) del PCA tras eliminar las anómalos

En la Figura 12 se observa como ahora siguiendo el criterio del máximo Q cuadrado acumulado el mejor modelo es el que tiene 18 componentes principales, incrementando el R cuadrado acumulado desde 0.68 hasta 0.78 y un Q cuadrado acumulado desde 0.62 hasta 0.75. Si solo se hubieran seleccionado 13 componentes como antes también habrían incrementado ligeramente, siendo el R cuadrado acumulado 0.7 y el Q cuadrado acumulado de 0.67.

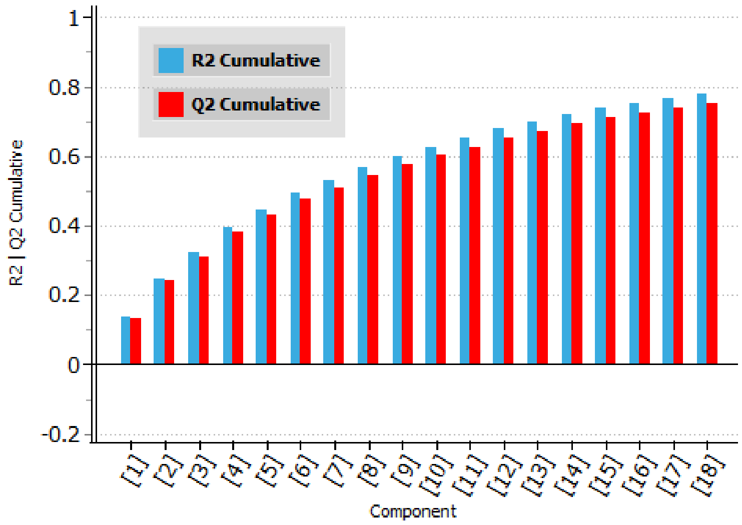

Figura 12. R2 acumulado y Q2 acumulado del PCA sin observaciones anómalas

El segundo paso en el análisis es el estudio de los scores para cada observación. Este estudio se va a centrar en las 4 componentes principales que más variabilidad explican para ver si hay algún patrón o agrupación definida en las conversaciones. No aparece un patrón claro en Figura 13 relacionado con los individuos considerados con síntomas depresivos (naranjas) y los individuos de control del estudio (negros).

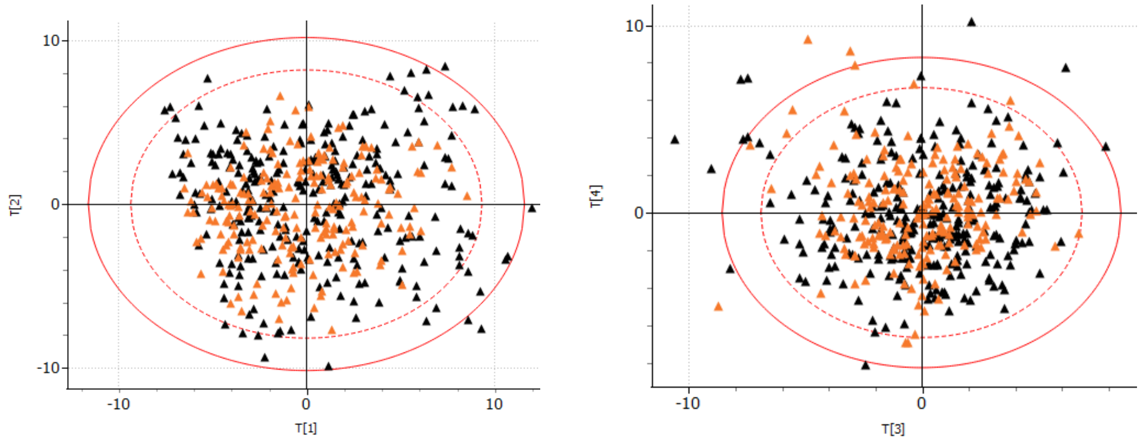

Figura 13. Scores en las 4 componentes principales del PCA coloreado si tienen o no síntomas depresivos

Tampoco se agrupan las conversaciones en función de las emociones de los avatares (Figura 14), su género o forma de vestir.

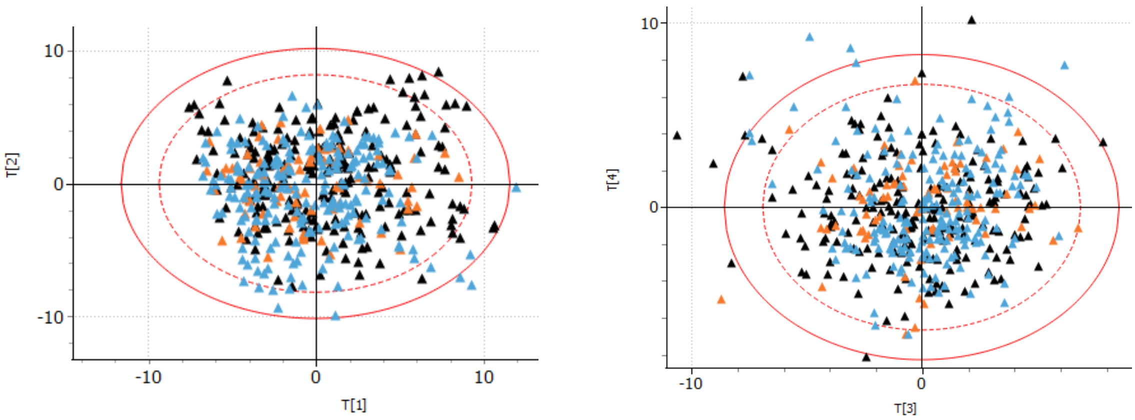

Figura 14. Scores 4 componentes principales del PCA coloreado con la emoción de los avatares

Una vez analizadas las observaciones con los scores se analizan las variables con sus loadings de las 2 componentes principales más importantes. Variables que tengan loadings similares estarán relacionadas directamente en cierta medida. En las 2 primeras componentes de la Figura 15 se observan 5 grupos de variables además de unas variables con poco peso en estas dos componentes principales situadas cerca del punto de origen de los ejes.

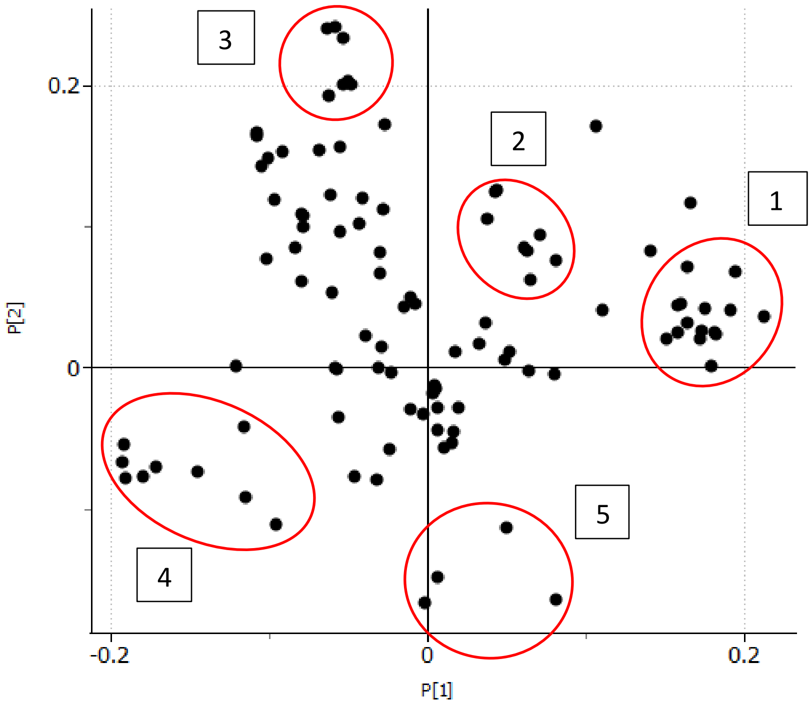

Figura 15. Loadings PCA en 2 primeras componentes principales y aumento de zona central

En el grupo número 1 se agrupan las siguientes variables con elevado loading en la primera componente y bajo en la segunda: “visits_eyes”, “fixcount_eyes”, “fixtime_eyes”, “fixtime_mean_eyes”, “fixcount_head”, “fixtime_head, “fixtime_mean_head”, “visits_mouth”, “fixcount_mouth”, “fixtime_mouth”, “fixtime_mean_mouth”, “visits_nose”, “fix_time_nose”, “fixtime_nose” y “fixtime_mean_nose”. Este grupo muestra la relación entre las visitas y fijaciones (tanto en cantidad como en tiempo) de la cabeza, ojos, nariz y boca. Algo lógico teniendo en cuenta su gran cercanía física.

En el segundo grupo están las siguientes variables con loading bastante alto tanto en la primera como en la segunda componente: “visits_foots”, “fixcount_foots”, “fixtime_foots”, “fixtime_foots”, “visits_arm_right”, “fixcount_arm_right”, “fixtime_mean_arm_right” y “visits_legs. Todas estas variables están relacionadas con las visitas y fijaciones de los pies, las piernas y el brazo derecho. Estas zonas del cuerpo son las más periféricas y alejadas de la cara, en contraposición con el grupo anterior.

El tercer grupo incluye las variables con mayor valor de loading en la segunda componente y ligero loading negativo en la primera. Junto con algunas relacionadas con los grados de las sacadas también está la desviación típica de la altura de las fijaciones: “sac_deg_ratio”, “sac_deg_mean”, “sac_deg_med”, “sac_deg0_mean”, “fix_elevation_std”, “sac_deg0_median” y “sac_deg1_count”.

El cuarto grupo lo forman todas las variables de tiempo de fijación total en las distintas partes del cuerpo registradas: cabeza, ojos, nariz, boca, torso, piernas, brazo izquierdo y brazo derecho. Todas ellas se caracterizan por un elevado loading negativo en la primera componente. Al ser tiempo total estas variables podrían estar bastante relacionadas con la duración de la conversación.

Al último grupo pertenecen 4 variables con elevados loading negativos en la segunda componente: media y mediana de la duración de las fijaciones y la mediana y la ratio del tiempo de sacada.

Una vez estudiadas las conversaciones y las características extraídas en el espacio X será interesante tener en cuenta simultáneamente su relación con la variable observada que se pretende predecir, la cual forma el espacio Y. Por tanto, a continuación, se presenta la interpretación del modelo PLS para la predicción del valor de PHQ en el cual se relacionan el espacio X e Y.

4.2. Interpretación modelo PLS predicción valor PHQ de las características

Partiendo nuevamente de todas las conversaciones se ajustará un PLS con todas las variables que prediga la variable numérica del PHQ. Esta vez sí se incluye la información del tipo de avatar y su estado de ánimo por si fuera relevante en el modelo. Estas nuevas variables al ser categóricas será necesario transformarlas en numéricas para el PLS y estandarizarlas de la misma forma que se había hecho en el PCA con el resto de las variables que ya son numéricas. Al solo haber 2 géneros y 2 tipos de ropa que distinguen los tipos de avatar, estas variables se transforman en binarias sin tener que añadir más variables. Sin embargo, para las 5 emociones es necesario crear 5 variables binarias que representan si el avatar tiene cada una de las emociones empleando la técnica de One Hot Encoding.

El R cuadrado acumulado de este modelo con 3 componentes y su Q cuadrado acumulado son bastante bajos (menores a 0,2) y se pueden ver en la Figura 16.

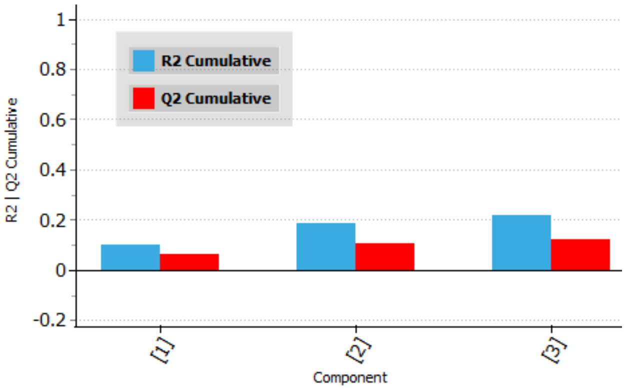

Figura 16. R2 acumulado y Q2 acumulado del PLS con todas las variables

El siguiente paso es eliminar aquellas variables que no sean significativas para la predicción del PHQ. Fijándose en los coeficientes asociados a cada variable es necesario eliminar aquellos con un valor bajo o nulo, lo cual indica su escasa importancia. En la Figura 17 aparecen los coeficientes iniciales y sus intervalos de confianza. Todas las variables cuyo intervalo de confianza incluye al 0 son susceptibles de su eliminación.

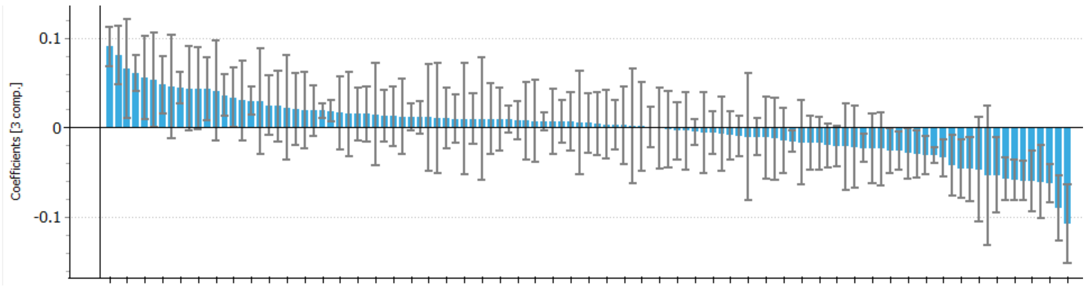

Figura 17. Coeficientes de todas las variables para el PLS sin depurar

Este proceso es similar al que se sigue para eliminar las observaciones anómalas debido a su carácter iterativo. Después de cada eliminación de variables se recalculan los coeficientes del resto para ver cuales no siguen siendo significativas. Por ejemplo, en la primera iteración se eliminan las 14 variables cuyo valor de coeficiente es más bajo y su intervalo de incertidumbre cruza el valor 0. Estas variables se pueden ver en la Figura 18.

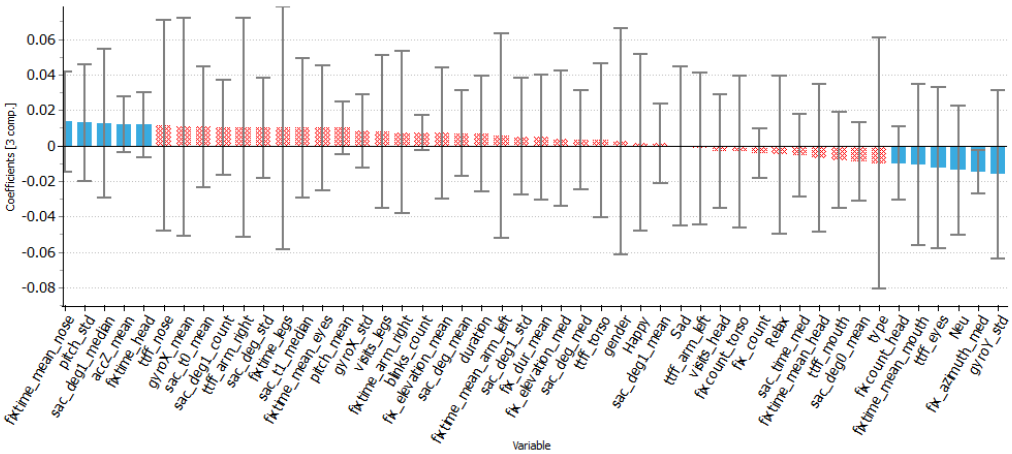

Figura 18. Variables eliminadas en la primera iteración de depuración del PLS

Finalmente, tras todas las iteraciones las 22 variables significativas son las mostradas en la Figura 19.

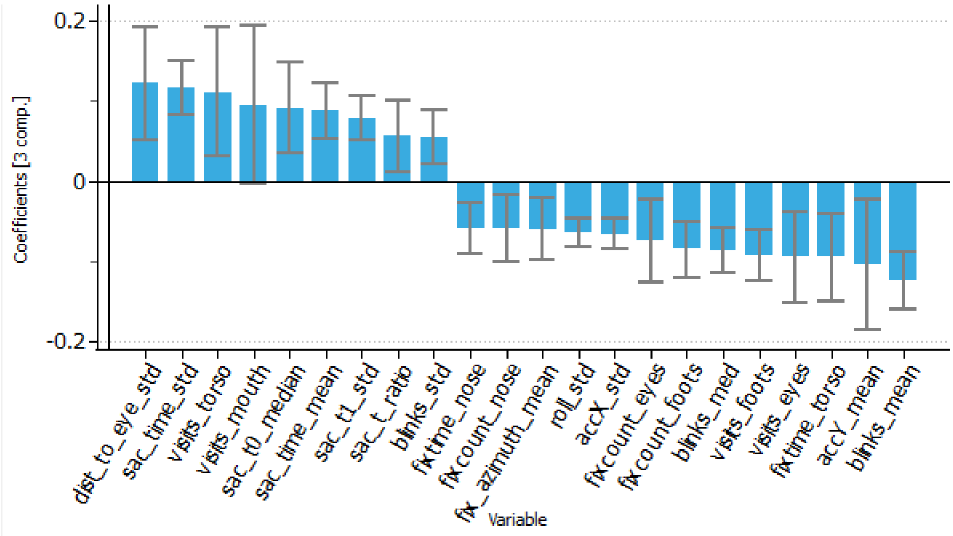

Figura 19. Coeficientes de todas las variables significativas para el PLS depurado

Otra forma alternativa de realizar esta criba de las variables significativas se podría hacer fijándonos en la importancia de las variables para la proyección (VIP). En la Figura 20

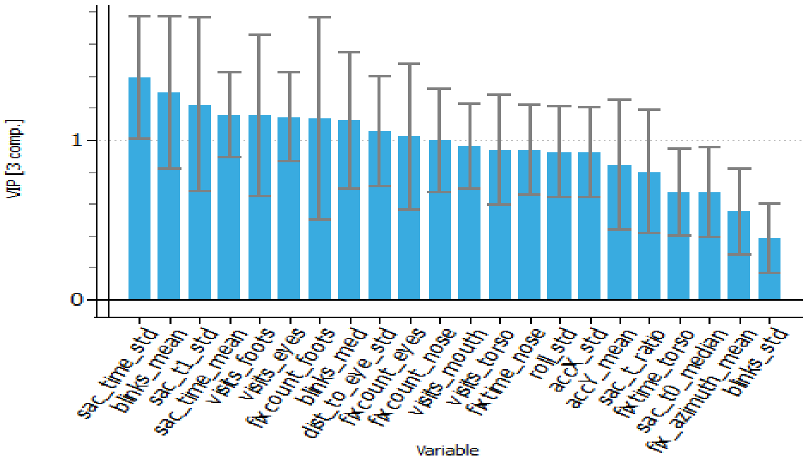

Figura 20. Importancia de las variables significativas en el PLS (VIP)

Con esta reducción de variables se aumenta el Q cuadrado acumulado reduciendo muy ligeramente el R cuadrado acumulado (Figura 16) respecto al modelo inicial (Figura 21).

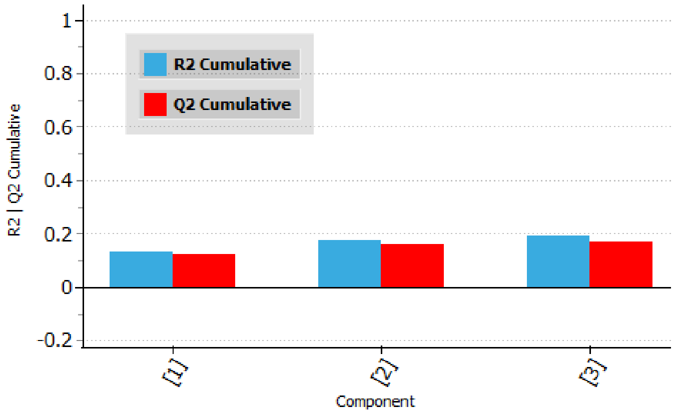

Figura 21. R2 acumulado y Q2 acumulado del PLS excluyendo las variables no significativas

En segundo lugar, se comprueba si existen relaciones no lineales que perjudiquen al modelo lineal PLS planteado. Para ello se extrae el gráfico que enfrenta la proyección t de los individuos en el subespacio X y la proyección u de los individuos en el subespacio Y para la primera componente principal. En la Figura 22 no se aprecia una recta muy definida formada por las observaciones, sino que hay una ligera curva por lo que es posible que haya alguna no linealidad que no se tenga en cuenta en el modelo PLS.

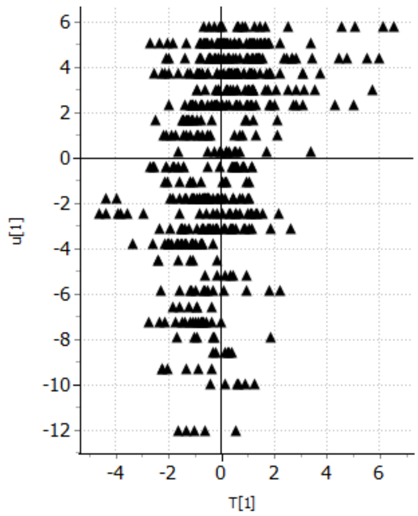

Figura 22. Proyección en el subespacio X respecto a la del subespacio Y en el PLS con v

Una vez ajustado el modelo se pasa a la interpretación de los scores de las conversaciones y los loadings de las variables. En los scores de las observaciones del PLS (Figura 23) hay mayor separación en función de su PHQ comparado con el PCA (Figura 13) en la primera componente. El PHQ se relaciona inversamente con esta componente, por lo que conversaciones con scores altos en esta componente tienden a provenir de usuarios con menor PHQ. En el resto de las componentes no se ve clara una asociación con el PHQ de forma visual.

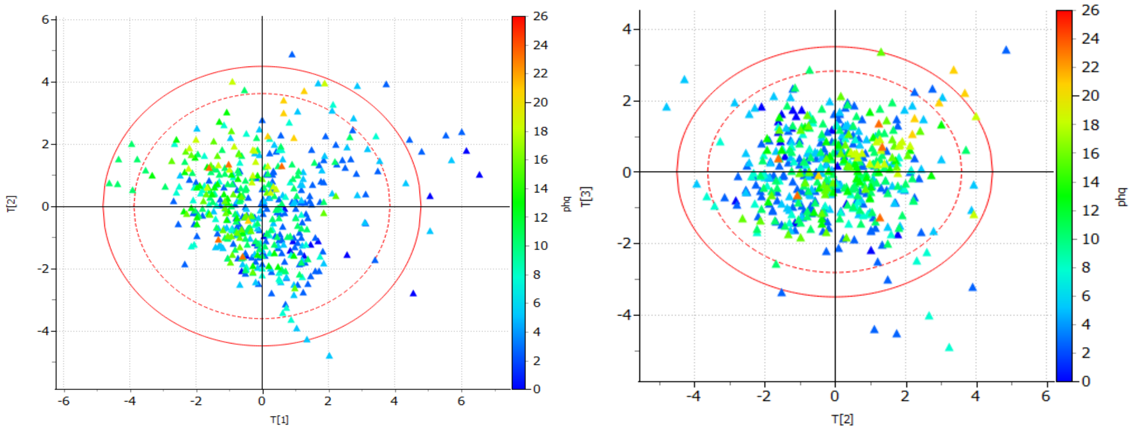

Figura 23. Scores 2 componentes PLS coloreado puntuación PHQ y etiqueta (SDS naranja)

Respecto a los loadings de las variables para las dos componentes principales más importantes, en la Figura 24 se pueden ver como las variables están relacionadas con el PHQ. Las 4 variables significativas relacionadas con las sacadas se relacionan directamente con el PHQ ya que se encuentran situadas cerca de él en la gráfica. Es decir, valores altos en la media del tiempo de sacada y su desviación típica están relacionados con un valor elevado de PHQ.

En cambio, aquellas variables alejadas en la dirección entre el PHQ y el centro de coordenadas son aquellas que están inversamente relacionadas con el PHQ. Las más importantes son la media y mediana de parpadeos. Esto quiere decir que un número elevado de parpadeos en la conversación suele conllevar un PHQ reducido. Además de estas dos variables también relacionadas inversamente las siguientes variables: media de aceleración vertical, desviación típica de aceleración en horizontal, visitas y cantidad de fijaciones de ojos y pies.

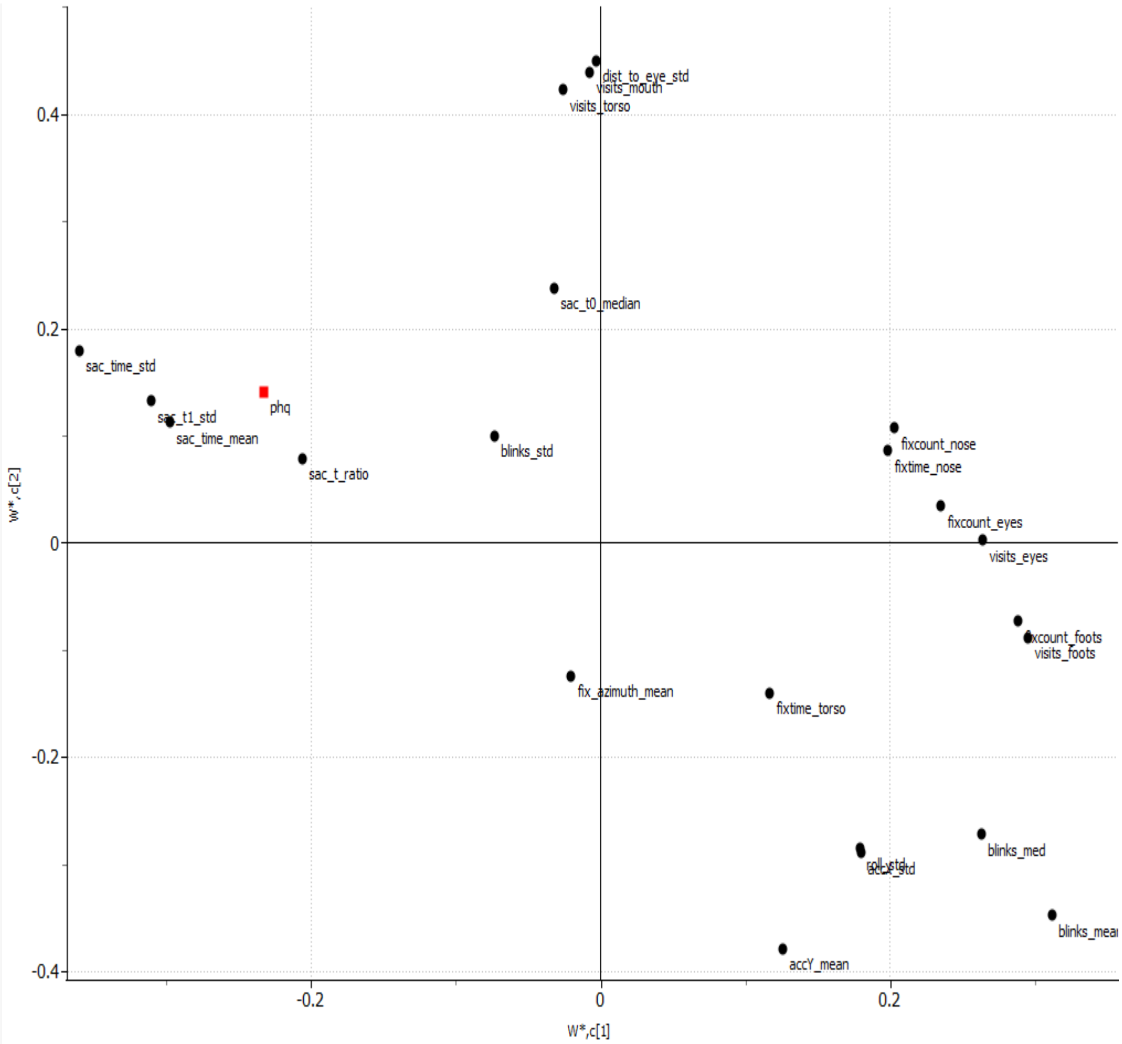

Figura 24. Loadings PLS de las variables significativas

Para finalizar es importante comprobar como de buenas son las predicciones del modelo PLS interpretado. En la Figura 25 están los resultados de la predicción para todo el conjunto de observaciones, las cuales se habían usado en el entrenamiento. El RMSE es de 5.66, un valor bastante elevado, por lo que las predicciones tienen un gran error.

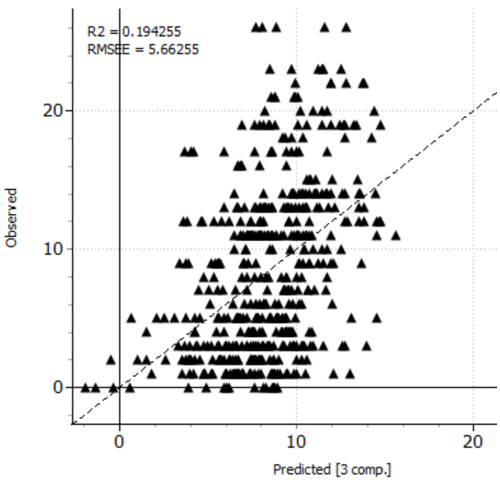

Figura 25. Predicción PLS conjunto de validación (izquierda) y el de entrenamiento (derecha)

El error en las predicciones junto con el bajo R cuadrado acumulado y Q cuadrado acumulado indican que este modelo no se ajusta del todo a los datos. Por lo tanto, la información extraída de su interpretación no es completa ni precisa, aunque si nos sirve para hacernos idea de algunas de las relaciones existentes. A continuación, se incluirá el modelo PLS en la comparación de modelos de minería de datos, pero se ajustará para una parte de las observaciones y se validará con el resto como el resto de los modelos.

4.3. Modelos de minería de datos con características extraídas

En este apartado se presentan los modelos que usan como datos las características extraídas de las conversaciones. Se implementan modelos para cada uno de los siguientes 3 objetivos:

- Predicción del valor numérico del PHQ del usuario que realiza una conversación
- Clasificar si el usuario muestra síntomas depresivos (PHQ igual o mayor a 9) o si no los muestra
- Clasificar si el usuario no tiene síntomas depresivos (PHQ menor o igual a 5), síntomas depresivos medios (PHQ mayor que 5 y menor que 14) o síntomas depresivos severos (PHQ mayor a 14)

Para el primer objetivo es un problema de regresión en el cual hay que emplear modelos que puedan predecir un valor numérico. El segundo objetivo es un problema de clasificación binario o con dos clases. En cuanto al tercer objetivo, si bien el tercer objetivo también es de clasificación, al tener 3 clases distintas será necesarios modelos de clasificación multiclase. Algunos modelos sirven para los 3 tipos de problemas planteados sin grandes modificaciones, mientras que otros solo sirven para clasificación o regresión.

En los 3 objetivos es imprescindible validar los modelos empleando la validación cruzada evitando que conversaciones de un individuo se empleen para entrenar y validar los modelos. En concreto se dividen las conversaciones en 5 grupos de tal forma que todas las conversaciones de cada usuario estén en uno de estos grupos. En esta división se intenta mantener un equilibrio entre los usuarios para que en cada grupo haya la mayor variedad posible de PHQ. Durante el ajuste del modelo se emplean sucesivamente 4 de ellos para entrenar los modelos y el restante para la validación. Este proceso se realiza 5 veces, empleando los 5 grupos para validar los modelos 1 vez.

Es fundamental no entrenar y validar el modelo con conversaciones del mismo usuario ya que el modelo, especialmente los más complejos, aprende a identificar el usuario y su PHQ en lugar de los rasgos que puedan diferenciar a las personas con PHQ elevado de las que no lo tienen. En pruebas previas al uso de la validación cruzada holdout empleando el 80% para entrenamiento y el restante para validación se consiguieron modelo de regresión Random Forest con 21.63 error cuadrático medio de predicción de PHQ. También se consiguió un modelo de clasificación con una precisión del 85% (sensibilidad del 84% y especificidad del 87%) y un área bajo la curva del 95%. Estos modelos son mucho mejor que los obtenidos al no mezclar los usuarios, por lo que queda claro que los modelos se enfocaban en identificar a los usuarios y no tenían aplicación real a los problemas planteados.

4.3.1 Predicción valor PHQ

En primer lugar, se van a ajustar y comparar modelos de regresión que predigan el valor numérico del PHQ. Para comparar los modelos se emplea como métrica el error cuadrático medio de predicción. Ninguno de los modelos tiene un error de predicción lo suficientemente bajo para poder considerarlos adecuados para su uso con los datos disponibles.

Se emplean todas las observaciones, tanto las detectadas como anómalas por el PCA y PLS como las imputadas al tener alguna variable constante. Se han hecho pruebas quitando las variables anómalas detectadas con el PCA y el error de los modelos aumenta ligeramente salvo en los modelos PLS y SVM que disminuye. Estas variaciones no son lo suficientemente grandes como para determinar que eliminando estas observaciones mejoran los modelos.

También se ha probado a eliminar las conversaciones con datos faltantes en lugar de incluirlas tras realizar la imputación, pero todos los modelos empeoran por lo que también se ha descartado.

4.3.1.1 PLS

El primer tipo de modelo ajustado es el PLS. Gracias al análisis e interpretación previa realizada con el PLS en el apartado 5.3 existe la posibilidad de aplicar el modelo solamente sobre las variables que se han considerado significativas. Se va a comprobar si efectivamente este modelo es mejor prediciendo que el realizado con todas las variables. También en este análisis previo se ha establecido que 3 componentes es el número más adecuado para el modelo. Por ello independientemente del número de variables empleado para que la comparación sea justa en todos se usará el mismo número de componentes principales.

En la Tabla 2 se puede ver el error cuadrático medio de las predicciones realizadas usando cada división de la base de datos como validación del modelo entrenado con las otras 4, así como la media de las 5 validaciones.

Tabla 2. MSE del modelo PLS para predicción del PHQ

	Val. 1	**Val. 2**	**Val. 3**	**Val. 4**	**Val. 5**	**Media**
PLS todas	46.14	43.31	105.03	42.03	39.42	55.19
PLS significativas	39.99	63.55	141.96	34.38	31.09	62.19

El error en todas ellas es muy alto teniendo en cuenta que los valores de PHQ van desde 0 hasta 26. En promedio usando todas las variables se consigue un error ligeramente menor, aunque en 3 de las 5 validaciones se obtiene menor error usando únicamente las variables significativas. En la Figura 26 para la validación con el primer conjunto se observa como ambas predicciones forman una nube de puntos que no se ajusta a la recta punteada que correspondería a la predicción perfecta.

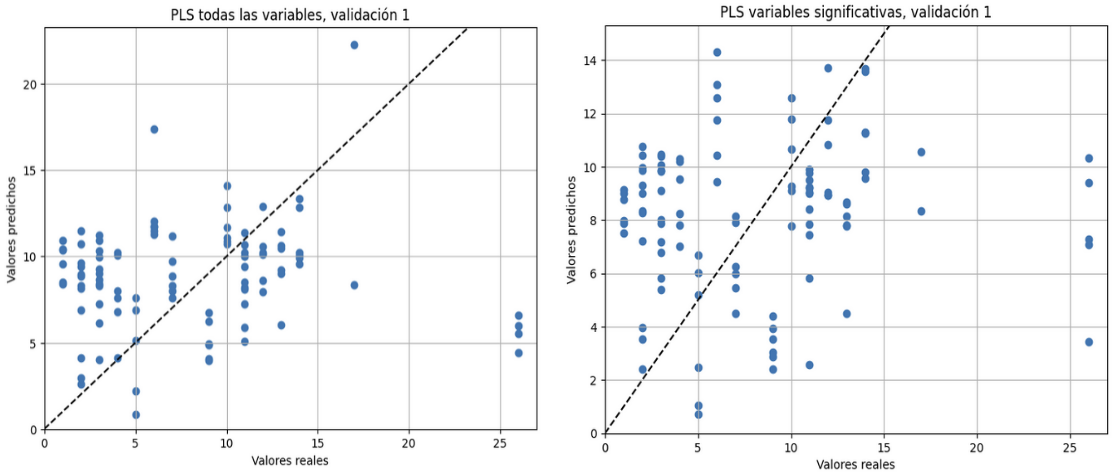

Figura 26. PHQ predicho por PLS respecto al real validando con la primera división

Se observa como las conversaciones con 26 de PHQ son las que mayor error tienen al no diferenciarse su predicción de otras conversaciones con menor PHQ. En la validación del segundo conjunto (Figura 27) también se observa una nube de puntos en ambas predicciones sin tendencia ascendente clara. Tampoco se ve que el modelo sea capaz de separar las conversaciones con mayor PHQ de las que tienen un PHQ muy bajo, aunque esta vez hay menor error con todas las variables a diferencia de con la validación anterior.

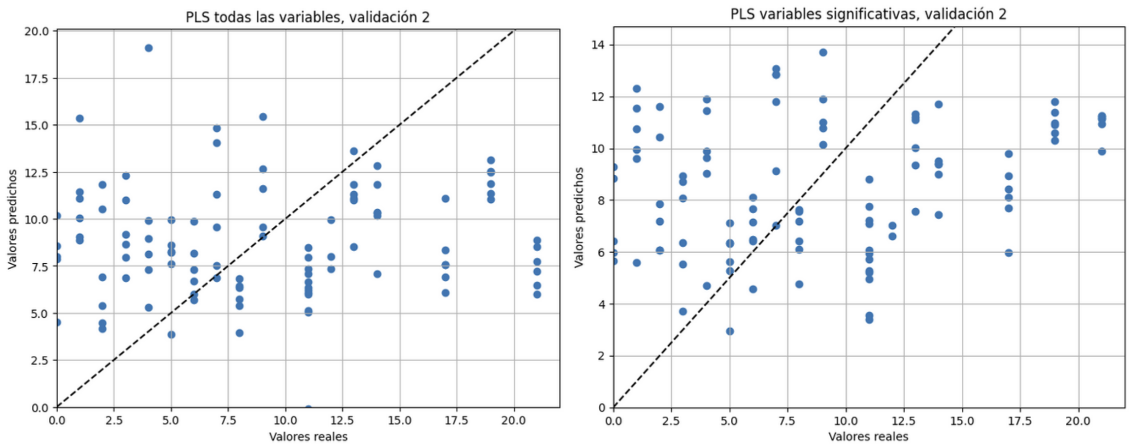

Figura 27. PHQ predicho por PLS respecto al real validando con la segunda división

En la tercera validación se aprecia de forma más destacada como la conversación con 26 de PHQ es la que mayor error tiene con mucha diferencia. De hecho, este error es el que provoca principalmente que en esta validación al usar todas las variables tenga menor media de error cuadrático. En la Figura 28 se aprecia como la predicción pasa de unos 90 a más de 120 en la predicción de esta conversación al pasar de usar todas las variables a solo las consideradas significativas.

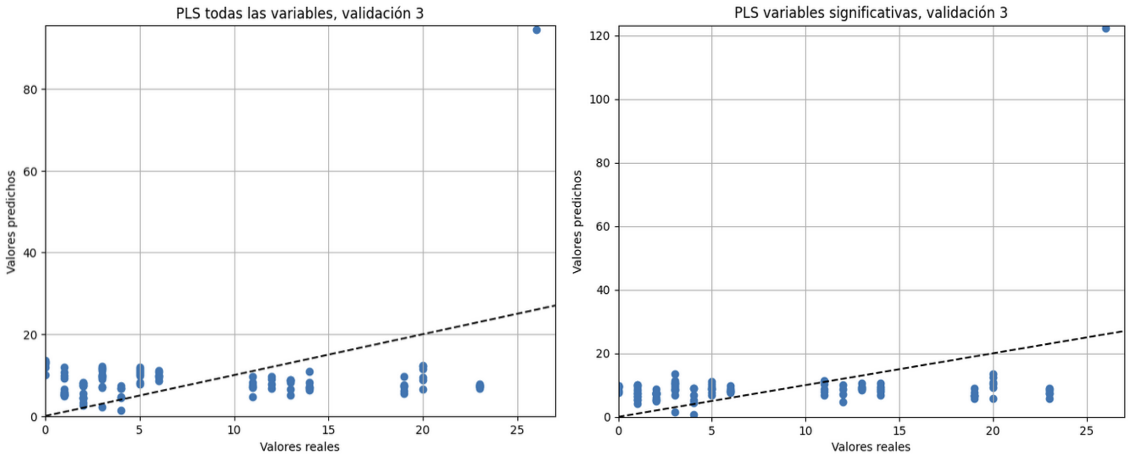

Figura 28. PHQ predicho por PLS respecto al real validando con la tercera división

De todas formas, nuevamente las predicciones son una nube de puntos sin tendencia destacable por lo que son malas en ambos modelos. En el resto de las validaciones tampoco se observa ninguna tendencia ascendente en las predicciones respecto al valor real de la Figura 29 y la Figura 30.

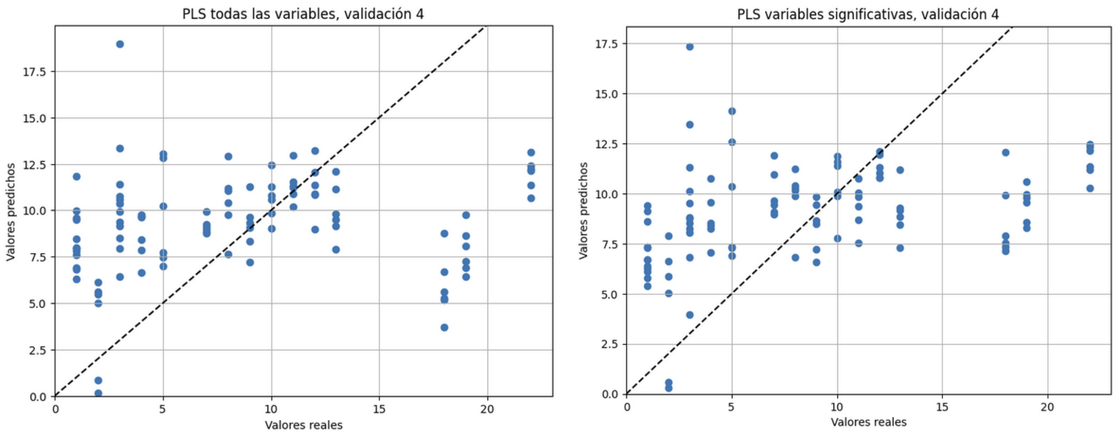

Figura 29. PHQ predicho por PLS respecto al real validando con la cuarta división

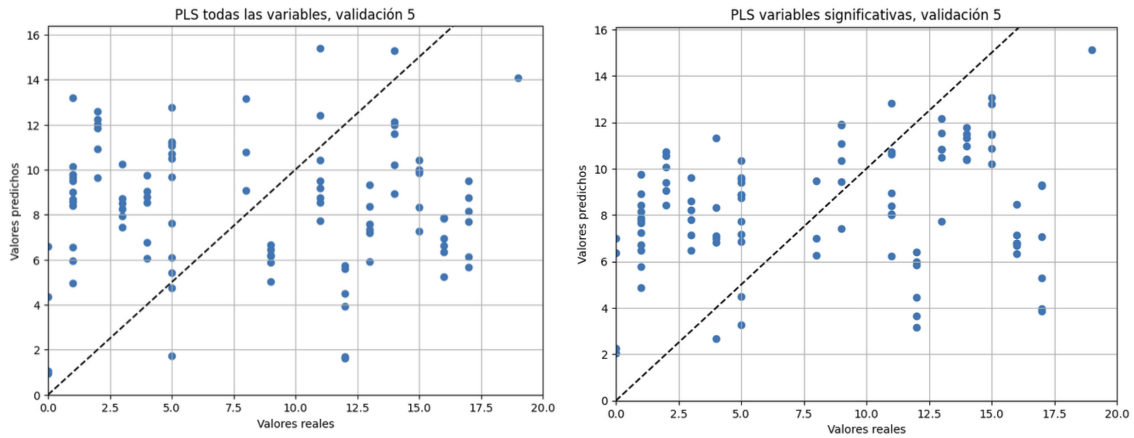

Figura 30. PHQ predicho por PLS respecto al real validando con la quinta división

Viendo que las predicciones son malas se ha probado a añadir y reducir componentes principales pero los resultados han sido semejantes. En definitiva, los modelos PLS no dan buenos resultados para predecir el valor numérico de PHQ con estas características.

4.3.1.2 Random Forest

El siguiente modelo ajustado es el de Random Forest. Debido a la complejidad de este tipo de modelos hay una gran cantidad de parámetros configurables. Se han probado varias combinaciones de 8 parámetros hasta encontrar con la que el error cuadrático medio de predicción era el inferior.

El primer parámetro ajustado es el número de árboles generados en el bosque. Se comienzan con 100 árboles y se observa que con 500 se obtienen los mejores resultados independientemente del resto de parámetros. Otro parámetro importante es usar muestras aplicando el método de remuestreo Bootstrap a la hora de construir los árboles, el cual también mejora en este caso los modelos. También se activa el uso de observaciones fuera de la muestra para estimar la puntuación generalizada, siendo en este caso el R cuadrado o coeficiente de determinación. Como criterio para medir la calidad de los splits o bifurcaciones se selecciona poisson, el cual consiste en usar la reducción de la desviación de Poisson para encontrar los splits o bifurcaciones.

Se prueban distintos valores de profundidad máxima de los árboles, pero no se aprecian mejoras claras al reducir la complejidad de los árboles y no se limita finalmente. En cuanto al mínimo número de muestras por Split y el mínimo de muestras por hojas se prueban conjuntamente varias combinaciones que aumenten las dos muestras mínimas por Split y una muestra mínima por hoja. Los aumentos en estos dos parámetros no mejoran las predicciones y se acaban dejando los valores iniciales. Por último, se comienzan las pruebas considerando todas las características buscando el mejor Split. Sin embargo, este parámetro se acaba dejando en el 20% de todas las características ya que daba mejor resultado.

En resumen, los parámetros seleccionados son: criterio de poisson, empleando Bootstrap y observaciones fuera de la muestra, 500 árboles, sin máxima profundidad de árbol, 2 mínimas muestras por Split y una por hoja, 20% de las características totales consideradas para cada split. El error de predicción del modelo con los parámetros finales en las distintas validaciones se resume en la Tabla 3.

Tabla 3. MSE del modelo Random Forest para predicción del PHQ

	Val. 1	**Val. 2**	**Val. 3**	**Val. 4**	**Val. 5**	**Media**
Random Forest	40.29	36.09	56.94	43.5	37.11	42.79

Una media del error cuadrático medio en las predicciones de 42.79 sigue siendo un error muy elevado, aunque menor que el obtenido con PLS. En la Figura 31 se ve nuevamente como las predicciones no se ajustan bien a los valores reales, resultando en una nube de puntos con apariencia aleatoria. Sin embargo, comparado con el PLS no hay ninguna observación con una predicción extrema que estropee la media. Con este modelo las predicciones tienen menos variación, situándose la mayoría en torno a 9 de PHQ. Por tanto, aunque parezca que es un modelo mejor que el PLS realmente este modelo tampoco sirve para predecir.

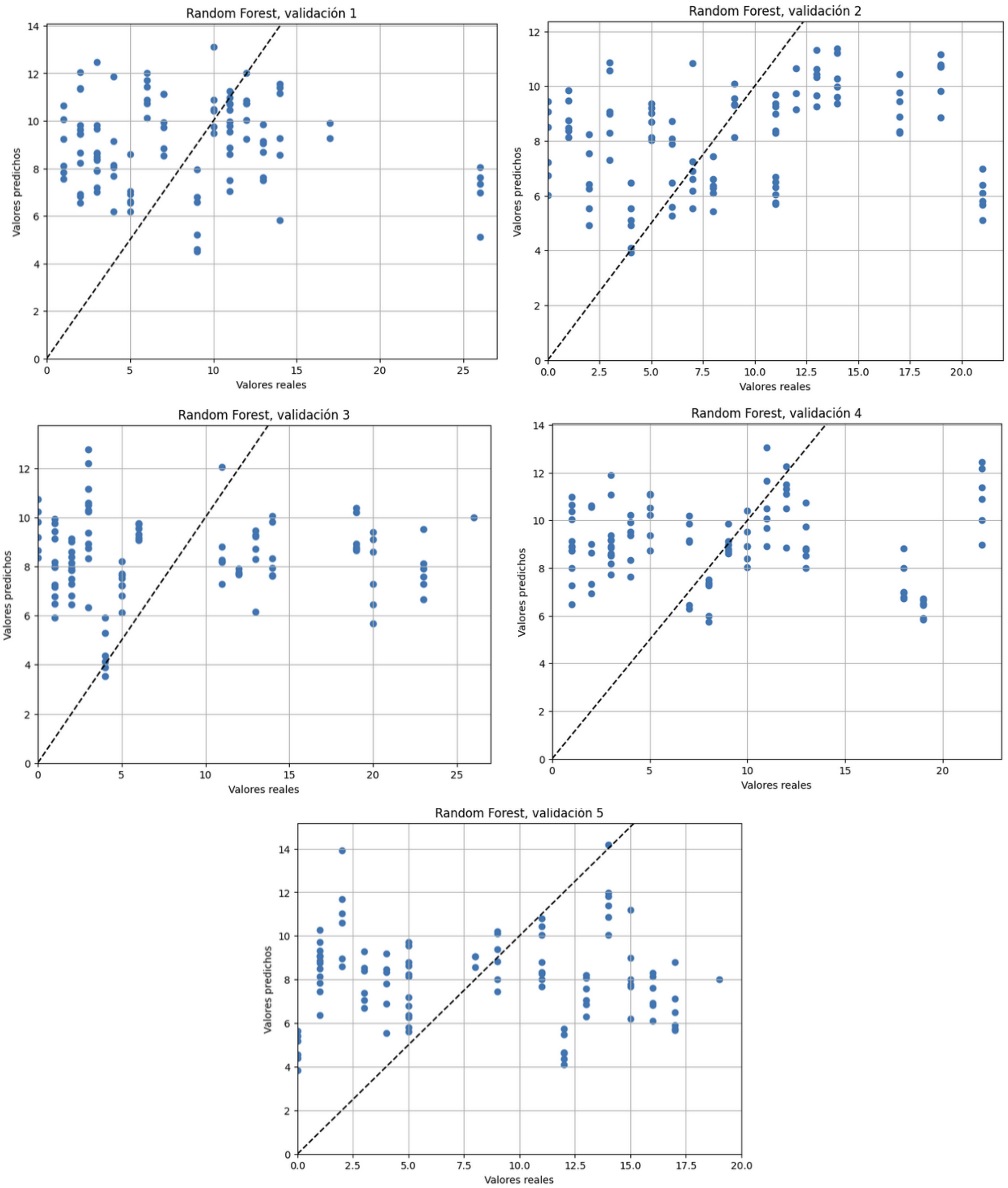

Figura 31. PHQ predicho por Random Forest respecto al real en todas las validaciones

4.3.1.3 Gradient boosted trees

A continuación, se ajusta el modelo bastado en Gradient boosted trees o árboles con potenciación del gradiente. Para conseguir el modelo con menor error cuadrático medio de predicción de este tipo se han configurado 7 parámetros. Primero se define la estructura del modelo con dos parámetros. El objetivo a minimizar del algoritmo es el error al cuadrado. Tras descartar el booster de tipo gblinear y el de tipo dart finalmente se usa el gbtree.

Una vez estructurado el modelo un parámetro fundamental es la tasa de aprendizaje. Un valor pequeño evita el sobreajuste del modelo reduciendo los pesos de las características para hacer el proceso de refuerzo más conservador. Comenzando en un valor de 0.1 se ha incrementado hasta el 0.3 definitivo. Otro parámetro para controlar como de conservador es el algoritmo es el gamma o reducción mínima de pérdidas necesaria para realizar otra partición en un nodo hoja del árbol. Se le asigna el valor 0, el cual es el menos conservador. El último parámetro relacionado con la agresividad del ajuste del algoritmo es el mínimo peso de instancia del nodo hoja. Comenzando las pruebas en 0 se incrementa su valor hasta 1 para que el algoritmo sea algo más conservador.

Respecto a la complejidad del algoritmo, es posible controlarla fijando la máxima profundidad de los árboles. Comenzando en 6 se va aumentando este valor en las pruebas hasta que el sobreajuste del modelo empeora los resultados, llegando a la conclusión de que la profundidad 10 es la más adecuada. Por último, se prueba a reducir las observaciones con las que se entrena cada árbol a un subconjunto aleatorio del conjunto de entrenamiento total para evitar el sobreajuste. Al ver que no se consiguen mejoras se acaban usando todas las observaciones.

En definitiva, los parámetros seleccionados son: error cuadrático como objetivo, gbtree como booster, tasa de aprendizaje de 0.3, gamma de 0, máxima profundidad de 10, peso mínimo de instancia del nodo hoja de 1 y como subconjutno se usa el 100% de los datos. El error de predicción del modelo con los parámetros finales en las distintas validaciones se resume en la Tabla 4.

Tabla 4. MSE del modelo Gradient boosted trees para predicción del PHQ

	Val. 1	**Val. 2**	**Val. 3**	**Val. 4**	**Val. 5**	**Media**
Gradient boosted trees	52.48	42.11	59.74	55.97	43.94	50.85

Una media del error cuadrático medio en las predicciones de 50.85 es un error muy elevado. Esta media de hecho es mayor que la media de error cuadrático conseguido con el modelo de Random Forest. Esto puede ser debido a que este algoritmo se esté sobreajustando más que el Random Forest y al validar con muestras que no están en su conjunto de entrenamiento empeore.

En la Figura 32 se ve nuevamente como las predicciones no se ajustan bien a los valores reales, resultando en una nube de puntos con apariencia aleatoria.

Como en el Random Forest no hay ninguna observación con una predicción extrema que estropee la media. Sin embargo, respecto al Random Forest se observa una mayor dispersión de las predicciones. En definitiva, el error excesivo en la predicción provoca que este modelo tampoco sea apto.

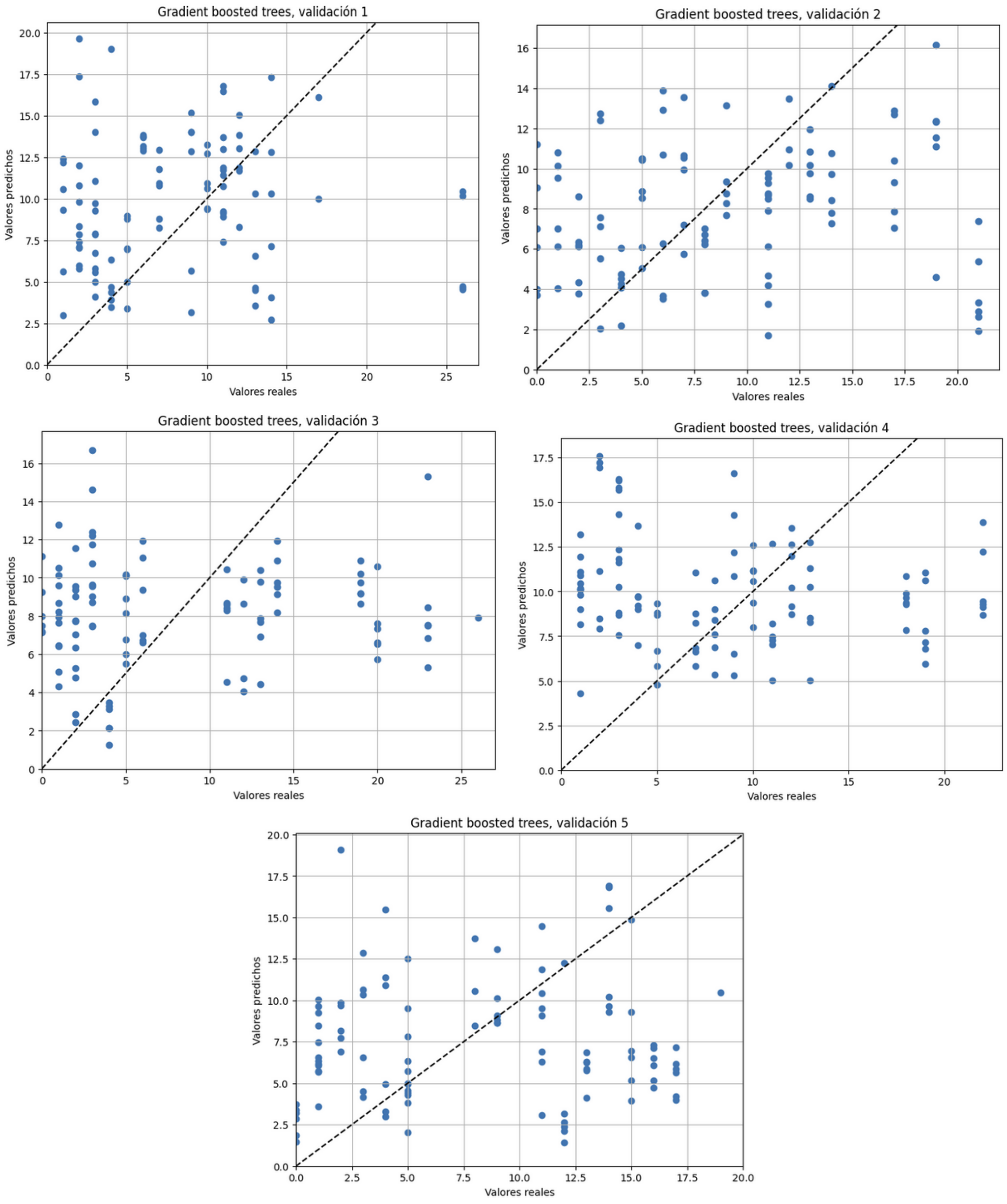

Figura 32. PHQ predicho por Gradient boosted trees respecto al real en todas las validaciones

4.3.1.4 SVR

En este apartado se comprueban los resultados obtenidos al aplicar un modelo de Regresión de Vectores de soporte o SVR (Support Vector Regression). En cuanto a los ajustes de parámetros para conseguir el menor error cuadrático medio de predicción se ha optado por un núcleo o kernel de la función de base radial o RBF. La tolerancia para detener la optimización es la que tiene por defecto la función empleada de 0.001. No se imponen iteraciones máximas debido a la velocidad de cálculo observada. El error de predicción del modelo en las distintas validaciones se resume en la Tabla 4.

Tabla 5. MSE del modelo SVR para predicción del PHQ

	Val. 1	**Val. 2**	**Val. 3**	**Val. 4**	**Val. 5**	**Media**
SVR	41.95	39.13	58.75	44.43	32.6	43.37

La media de 43.37 es un error elevado similar al obtenido con el Random Forest. Nuevamente se observa como las predicciones se distribuyen de forma aleatoria entorno al valor de PHQ de 8 en la Figura 33. Este modelo tampoco sirve para realizar las predicciones

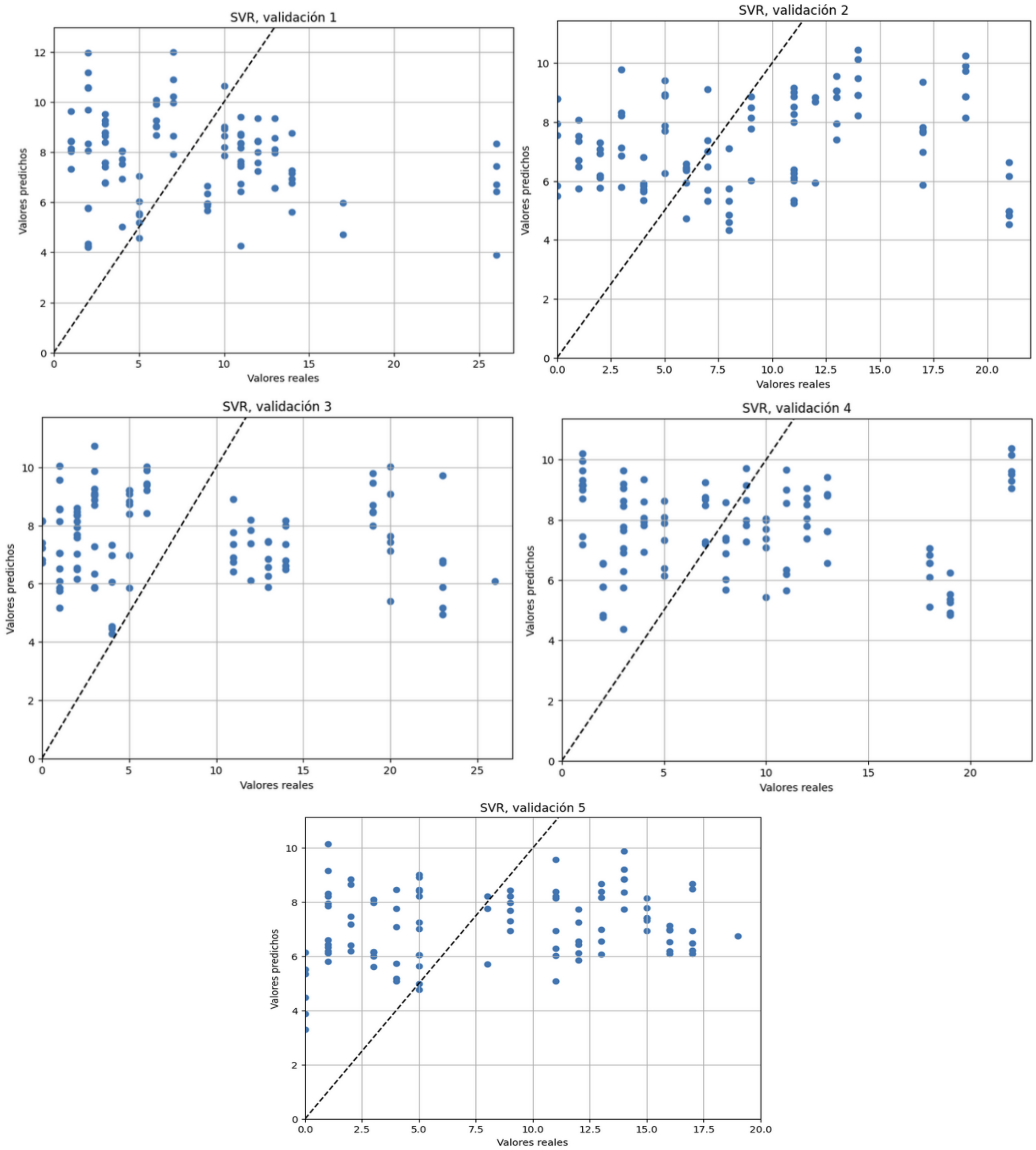

Figura 33. PHQ predicho por SVM respecto al real en todas las validaciones.

4.3.1.5 Regresión KNN

El modelo de regresión basada en K vecinos más cercanos seleccionado tras realizar varias pruebas usa 5 vecinos y su error de predicción está en la Tabla 6.

Tabla 6. MSE del modelo regresor KNN para predicción del PHQ

	Val. 1	Val. 2	Val. 3	Val. 4	Val. 5	Media
KNN	53.12	40.77	73.78	56.85	44.92	53.89

El error medio es de 53.89, siendo este demasiado alto. Nuevamente las predicciones de la Figura 34 no se ajustan a los valores reales.

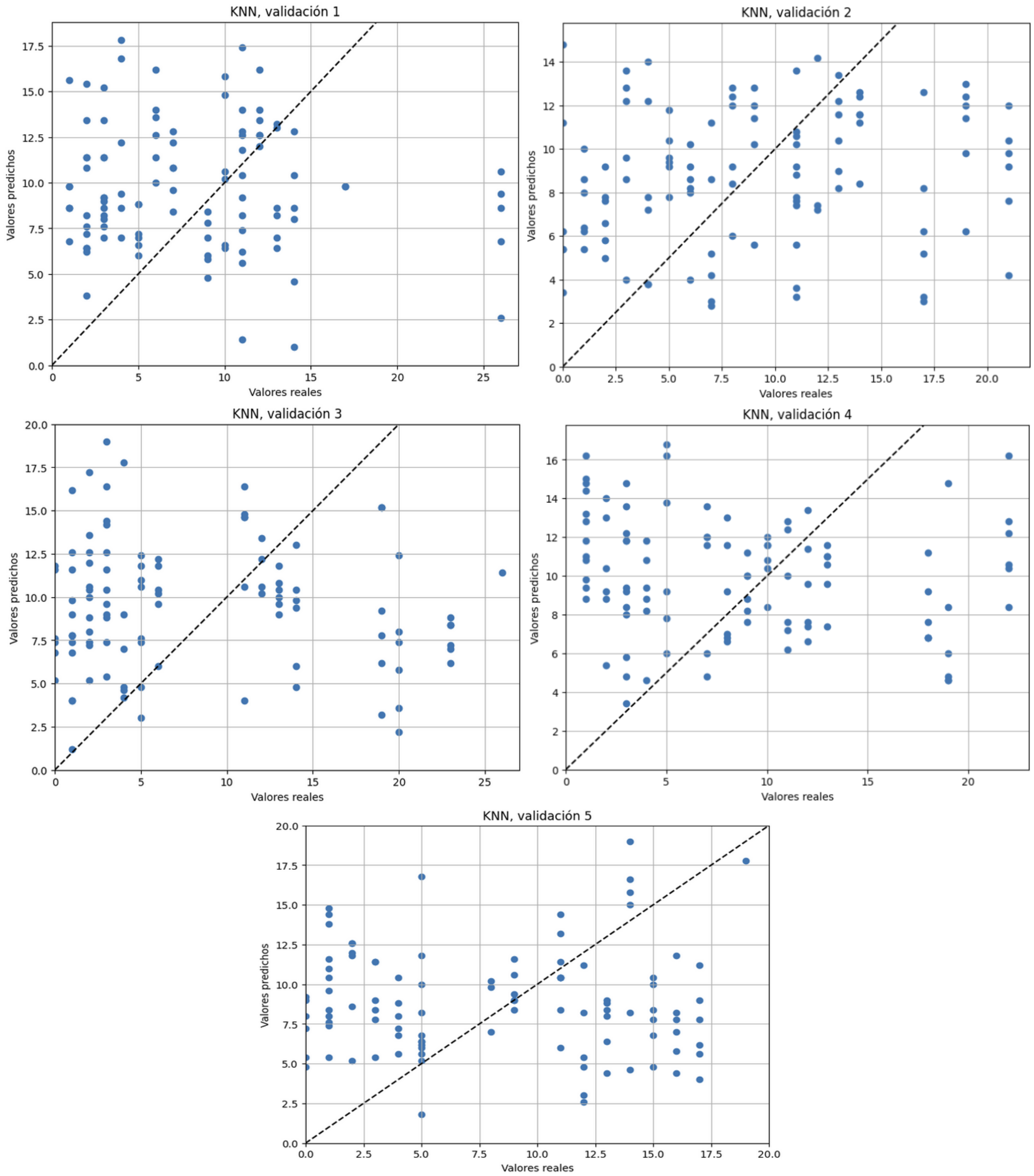

Figura 34. PHQ predicho por regresión KNN respecto al real en todas las validaciones.

4.3.1.6 Gaussian Process Regression

El modelo Gaussian Process Regression se parametriza definiendo el punto de partida del kernel y cuantas veces se reinicia la optimización de sus hiperparámetros. En concreto se ha elegido una combinación de un kernel constante y otro basado en función de base radial o RBF.

El kernel constante multiplica la salida del otro kernel por un valor constante. Se escoge como valor inicial de este kernel 1 y los límites entre los que puede variar en la optimización son 0.001 y 1000. El kernel RBF define la suavidad de la función modelada, lo cual depende de la escala de longitud que aplique. El valor inicial de la escala es de 10 y se permite que fluctúe durante la optimización entre 0.01 y 200. Además, se determina que el optimizador se reinicia hasta 9 veces para encontrar los mejores hiperparámetros. Con ello se aumentan las posibilidades de encontrar el óptimo global en lugar de solamente un óptimo local.

En la Tabla 7 se presenta el error cuadrático medio en las predicciones para este modelo en las 5 validaciones, siendo su media de 53.72.

Tabla 7. MSE del modelo Gaussian Process Regression para predicción del PHQ

	Val. 1	**Val. 2**	**Val. 3**	**Val. 4**	**Val. 5**	**Media**
Gaussian Process Regression	56.86	48.1	68.15	55.26	40.26	53.72

Nuevamente al representar las predicciones de los 5 conjuntos de validación respecto a sus valores reales de PHQ (Figura 35) se aprecia una gran dispersión en las predicciones de conversaciones con el mismo PHQ. No hay un patrón claro en las predicciones que indique que el modelo sea adecuado para los datos.

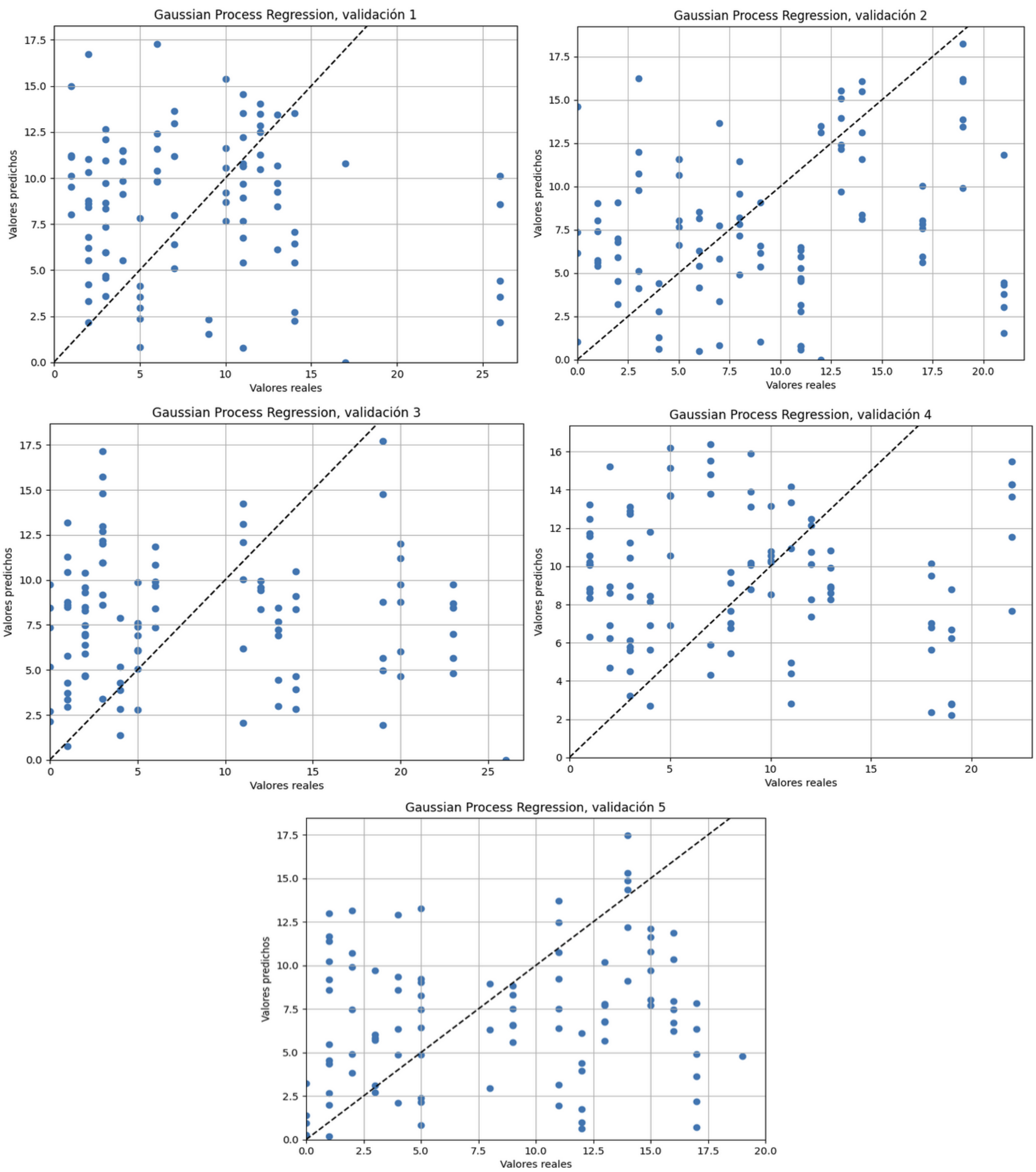

Figura 35. PHQ predicho por Gaussian Process Regression respecto al real en todas las validaciones

4.3.1.7 Perceptrón multicapa (MLP)

En un modelo de perceptrón multicapa es necesario ajustar tanto el ratio de aprendizaje como el objetivo para el optimizador. Se fija el error mínimo cuadrático como el objetivo a minimizar y el ratio de aprendizaje en 0.001.

Respecto a los parámetros específicos de estos modelos es fundamental definir el número de capas, el tipo de función de activación de cada capa y la cantidad de neuronas por capa. Se determina que todas las capas tengan una función de activación ReLU o activación linear rectificada.

Para establecer el número de capas y la cantidad de neuronas por capa se prueban entre 1 y 10 capas las cuales tienen desde 32 hasta 480 neuronas. Debido al elevado número de combinaciones posibles (más de 54 millones) se decide probar de forma aleatoria 100 de ellas. La mejor entre las probadas está compuesta por 3 capas de 352, 288 y 320 neuronas respectivamente además de la capa de salida. Otras de las mejores combinaciones han dado un error de predicción parecido, el cual es elevado y se puede ver en la Tabla 8.

Tabla 8. MSE del modelo Perceptrón multicapa para predicción del PHQ

	Val. 1	**Val. 2**	**Val. 3**	**Val. 4**	**Val. 5**	**Media**
Red Neuronal	56.91	58.01	90.34	53.15	53.77	62.43

En la Figura 36 se corrobora el elevado error cuadrático de predicción medio de 62.43 al no ajustarse las predicciones a los valores reales. Destaca el error de predicción en el tercer grupo con la conversación de 26 PHQ.

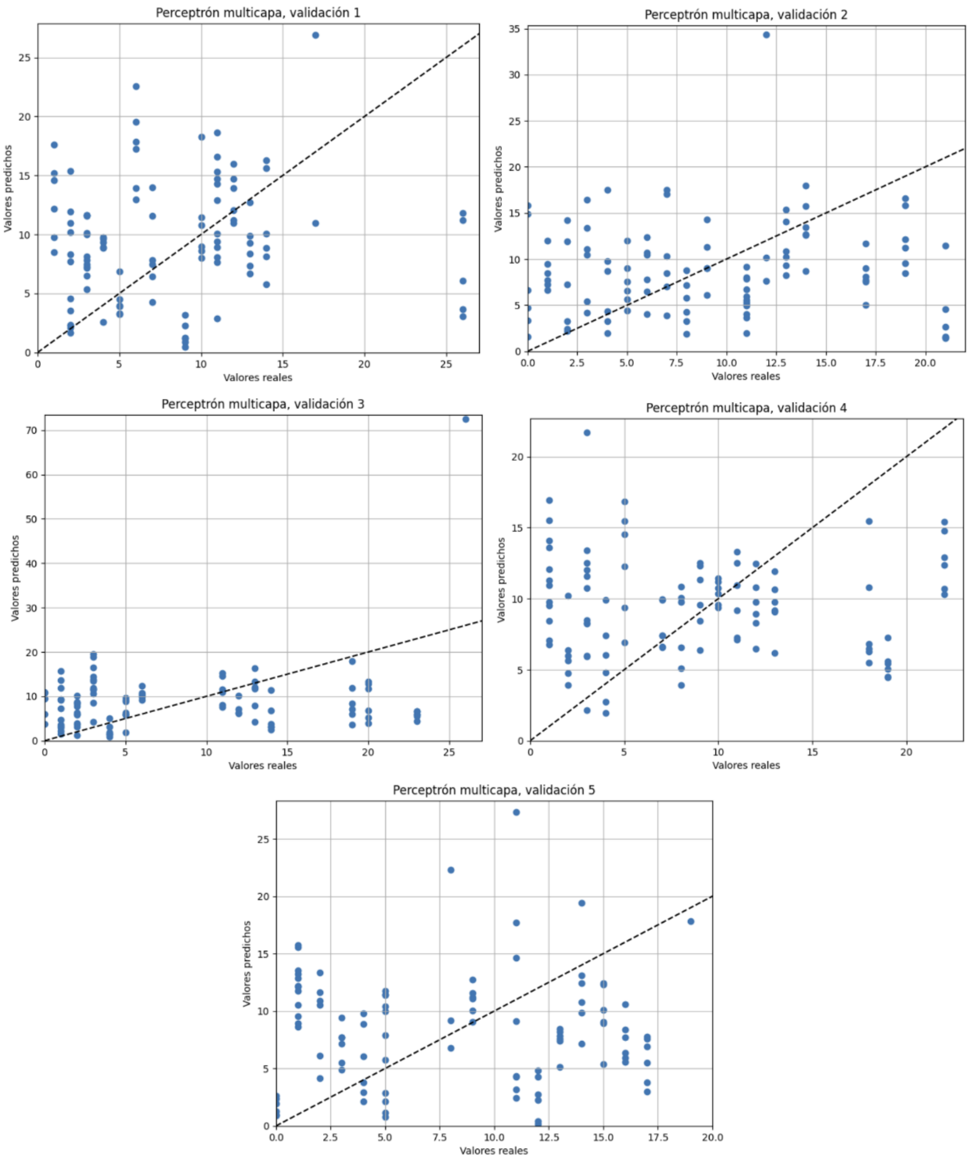

Figura 36. PHQ predicho por Perceptrón multicapa respecto al real en todas las validaciones

4.3.2 Clasificación 2 clases

En segundo lugar, se ajustan y comparan modelos de clasificación de 2 clases: con y sin síntomas depresivos. La principal métrica usada para comparar los modelos es la precisión en las predicciones. También se comprueban las métricas de sensibilidad, especificidad, el área bajo la curva ROC, F-score, coeficiente de correlación de Matthews.

Igual que con los modelos de regresión también se ha probado el efecto que tendría eliminar las conversaciones anómalas del PCA y las que tienen valores faltantes. En ambos casos las precisiones de los modelos varían levemente según el modelo, en algunos mejorando y en otros empeorando. Este cambio es en gran parte debido a la aleatoriedad a la hora de optimizar los parámetros de los modelos y no implica que los modelos sean mejores al quitar estas observaciones. Este efecto se observa también cambiando la semilla aleatoria usada por los algoritmos de optimización de los modelos al inicio.

Una vez ajustados todos los modelos planteados no hay ninguno cuya precisión media en las validaciones sea superior al 60%. Por lo tanto, ningún modelo realiza buenas precisiones. El modelo con la mejor precisión media, 60.5%, es el PLS usando las variables que se habían determinado como significativas en el análisis previo. El resto de los modelos tienen una precisión media superior al 50%. De ellos destaca el de Gradient boosted trees, el cual con 56% de precisión media tiene la menor variación de la precisión entre las distintas validaciones.

4.3.2.1 PLS-DA

El primer tipo de modelo probado son los PLS. Los modelos PLS adaptados a problemas de clasificación se denominan PLS-DA. Tras unas pruebas se ha establecido que 3 componentes principales es el número más adecuado para el modelo, igual que sucedía al aplicar PLS a la predicción del PHQ. Se emplean las mismas variables significativas que se habían detectado en el apartado 5.3 para el valor numérico de PHQ. Probando con todas las variables los modelos resultantes tenían menor precisión de media en las predicciones.

En la Tabla 9 se pueden ver las distintas métricas de las predicciones realizadas usando cada división de la base de datos como validación del modelo entrenado con las otras 4, así como la media de las 5 validaciones.

Tabla 9. Métricas de validación del modelo PLS-DA para clasificación en 2 clases

PLS-DA	Val. 1	Val. 2	Val. 3	Val. 4	Val. 5	Media
Precisión	0.59	0.52	0.69	0.62	0.6	0.6
Sensibilidad	0.63	0.37	0.56	0.42	0.41	0.48
Especificidad	0.57	0.63	0.78	0.77	0.75	0,7
AUC	0.59	0.47	0.69	0.66	0.56	0.59
F-score	0.56	0.4	0.6	0.48	0.47	0.50

Coef. Cor. Matthews	0.19	0	0.35	0.2	0.16	0.18

La precisión promedio en las 5 validaciones es del 60%. Existe gran variabilidad entre las validaciones de la precisión, yendo desde 0.69 en la tercera (Figura 37) hasta solo 0.52 en la segunda (Figura 38).

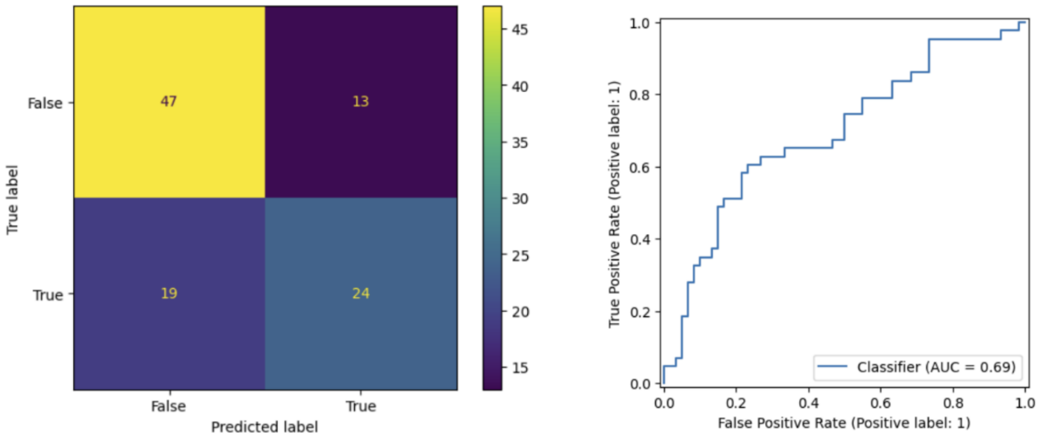

Figura 37. Matriz de confusión y curva ROC del PLS-DA de 2 clases en la tercera validación

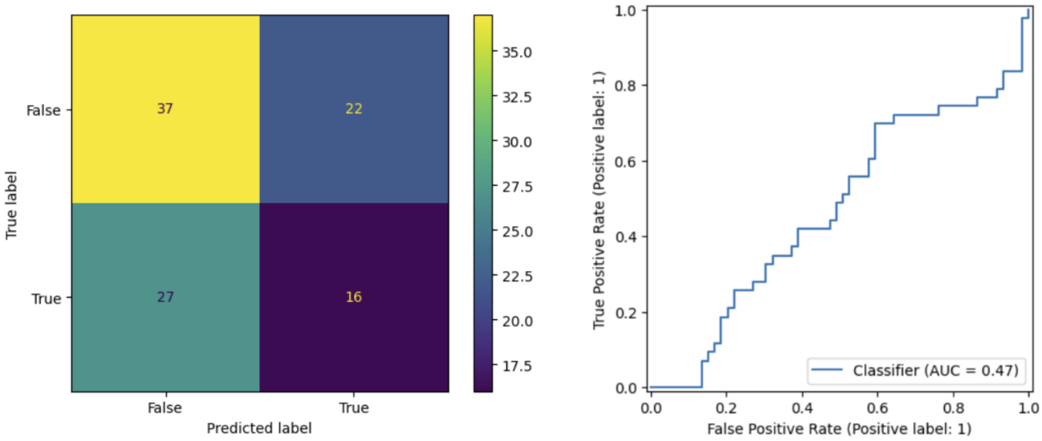

Figura 38. Matriz de confusión y curva ROC del PLS-DA de 2 clases en la segunda validación

En todas las validaciones menos la primera la especificidad es bastante superior a la sensibilidad. Es decir, los modelos son capaces de detectar mejor que una persona no tiene síntomas depresivos que detectar cuando sí los tiene. El valor del F-score es bastante bajo, siendo aún más bajo el coeficiente de correlación de Matthews al existir cierta descompensación en las observaciones. Hay más conversaciones de usuarios clasificados como personas sin síntomas depresivos que las que pertenecen a usuarios clasificados con síntomas depresivos.

4.3.2.2 Random Forest

El siguiente modelo ajustado es el de Random Forest clasificador. Se han probado varias combinaciones de 8 los mismos parámetros ajustados para el

Random Forest del PHQ. El objetivo era encontrar la combinación que tuviera la mayor precisión media en las predicciones con los distintos conjuntos de validación.

El primer parámetro ajustado es el número de árboles generados en el bosque. Se comienzan con 100 árboles y se observa que incrementando el número se obtienen peores resultados independientemente del resto de parámetros. Esto puede ser debido a la naturaleza del problema de clasificación respecto al de regresión, el cual se ve beneficiado de un modelo más sencillo en este caso. Tampoco se observan mejoras significativas reduciéndolos por lo que se fija 100 como el valor definitivo.

Otro parámetro importante es usar muestras aplicando el método de remuestreo Bootstrap a la hora de construir los árboles, el cual también mejora en este caso los modelos. También se activa el uso de observaciones fuera de la muestra para estimar la puntuación generalizada, siendo en este caso el R cuadrado o coeficiente de determinación. Como criterio para medir la calidad de los splits o bifurcaciones se selecciona gini, el cual consiste en usar la reducción de la impureza de Gini para encontrar los splits o bifurcaciones.

Se prueban distintos valores de profundidad máxima de los árboles, pero no se aprecian mejoras claras al reducir el tamaño de los árboles y no se limita finalmente. En cuanto al mínimo número de muestras por Split y el mínimo de muestras por hojas se prueban conjuntamente varias combinaciones que aumenten las dos muestras mínimas por Split y una muestra mínima por hoja. Los aumentos en estos dos parámetros no mejoran las predicciones y se acaban dejando los valores iniciales. Por último, se comienzan las pruebas considerando todas las características buscando el mejor Split. Al no apreciar mejoras reduciéndolo, este parámetro se acaba dejando en el 100% de todas las características ya que daba mejor resultado.

En resumen, los parámetros seleccionados son: criterio de gini, empleando Bootstrap y observaciones fuera de la muestra, 100 árboles, sin máxima profundidad de árbol, dos mínimas muestras por Split y una por hoja, 100% de las características totales consideradas para cada split. Las métricas de las predicciones del modelo con los parámetros finales en las distintas validaciones se presentan en la Tabla 10.

Tabla 10. Métricas de validación del modelo Random Forest para clasificación en 2 clases

Random Forest	Val. 1	Val. 2	Val. 3	Val. 4	Val. 5	Media
Precisión	0.6	0.55	0.56	0.69	0.5	0.58
Sensibilidad	0.42	0.37	0.35	0.58	0.39	0.42
Especificidad	0.73	0.68	0.72	0.77	0.58	0.70
AUC	0.54	0.49	0.56	0.75	0.49	0.57
F-score	0.47	0.41	0.4	0.61	0.4	0.46
Coef. Cor. Matthews	0.16	0.05	0.07	0.35	- 0.04	0.12

La media de la precisión es 0.58, ligeramente inferior a la obtenida con el PLS y la fluctuación de la precisión entre validaciones es similar en ambos. En el resto de las métricas la media también es inferior en el Random Forest, por lo que el modelo es ligeramente peor. En la cuarta validación se obtiene la mejor precisión (Figura 39) y en la quinta se obtiene la peor (Figura 40).

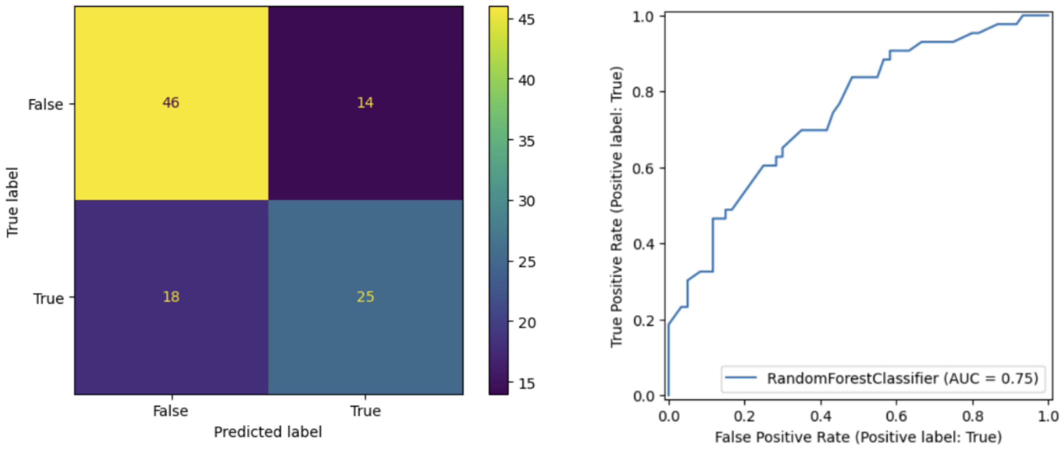

Figura 39. Matriz de confusión y curva ROC del Random Forest de 4 clases en la cuarta validación

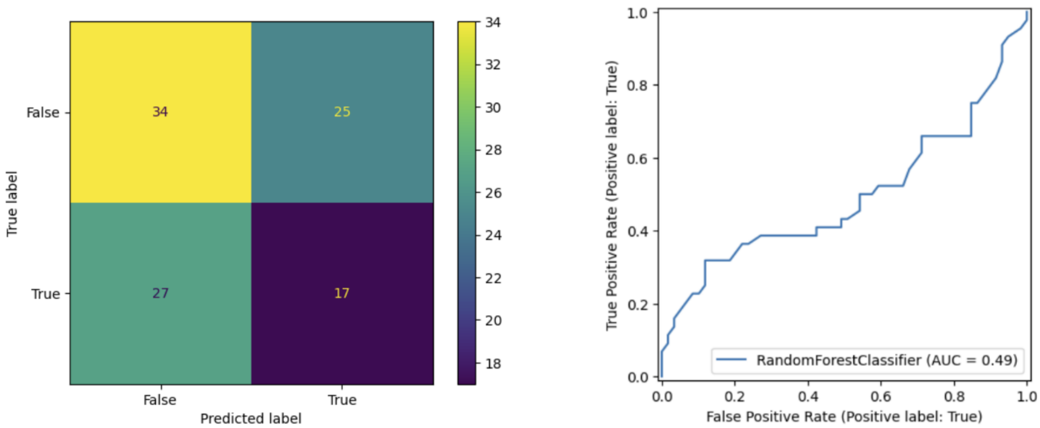

Figura 40. Matriz de confusión y curva ROC del Random Forest de 2 clases en la quinta validación

Nuevamente la especificidad media es superior a la sensibilidad media, siendo esta vez en todas las validaciones. El modelo para la validación 5 al solo tener un 50% de precisión sería equivalente a elegir aleatoriamente la clase, por lo que el modelo no es bueno en algunas validaciones. No se han encontrado parámetros que produzcan resultados más consistentes entre todos los grupos de validación, algo fundamental para que el modelo mejorase. Por tanto, el modelo PLS además de ser menos intensivo computacionalmente ofrece predicciones ligeramente mejores sin empeorar en consistencia. En definitiva, el modelo de PLS es más adecuado que el de Random Forest.

4.3.2.3 Gradient boosted trees

A continuación, se ajusta el modelo bastado en Gradient boosted trees o árboles con potenciación del gradiente. Para conseguir el modelo con mejores predicciones de este tipo se han configurado 7 parámetros. Primero se define la estructura del modelo con dos parámetros. El objetivo a minimizar del algoritmo es el logaritmo de la función de verosimilitud o “logloss”. Tras descartar el booster de tipo gblinear y el de tipo dart finalmente se usa el gbtree.

Una vez estructurado el modelo un parámetro fundamental es la tasa de aprendizaje. Comenzando en un valor de 0.1 se ha incrementado hasta el 0.5 definitivo, casi el doble del modelo del mismo tipo empleado para predecir el PHQ. Otro parámetro para controlar como de conservador es el algoritmo es el gamma o reducción mínima de pérdidas necesaria para realizar otra partición en un nodo hoja del árbol. Se le asigna el valor 0, el cual es el menos conservador. El último parámetro relacionado con la agresividad del ajuste del algoritmo es el mínimo peso de instancia del nodo hoja. Comenzando las pruebas en 0 se incrementa su valor hasta 1 para que el algoritmo sea algo más conservador.

Respecto a la complejidad del algoritmo, es posible controlarla fijando la máxima profundidad de los árboles. Comenzando en 6 se va aumentando este valor en las pruebas hasta que el sobreajuste del modelo empeora los resultados, llegando a la conclusión de que la profundidad 10 es la más adecuada. Por último, se prueba a reducir las observaciones con las que se entrena cada árbol a un subconjunto aleatorio del conjunto de entrenamiento total para evitar el sobreajuste. Al ver que no se consiguen mejoras se acaban usando todas las observaciones.

En definitiva, los parámetros seleccionados son: logloss como objetivo, gbtree como booster, tasa de aprendizaje de 0.5, gamma de 0, máxima profundidad de 10, peso mínimo de instancia del nodo hoja de 1 y como subconjunto se usa el 100% de los datos. Las métricas de las predicciones del modelo con los parámetros finales en las distintas validaciones se resumen en la Tabla 11.

Tabla 11. Métricas de validación del modelo Gradient boosted trees para clasificación en 2 clases

Grad. b. trees	Val. 1	Val. 2	Val. 3	Val. 4	Val. 5	Media
Precisión	0.55	0.54	0.57	0.6	0.56	0.57
Sensibilidad	0.47	0.37	0.37	0.6	0.43	0.45
Especificidad	0.62	0.66	0.72	0.6	0.66	0.65
AUC	0.55	0.51	0.56	0.66	0.52	0.56
F-score	0.47	0.41	0.42	0.56	0.46	0.46
Coef. Cor. Matthews	0.08	0.03	0.09	0.2	0.09	0.1

Si bien la media de precisión es ligeramente inferior al PLS y Random Forest, la variación de la precisión entre las validaciones es menor. La mejor precisión, cuyo valor es del 60%, se consigue en la cuarta validación (Figura 41), siendo la peor precisión del 54%. Mientras tanto en los otros dos tipos de modelos alguna de las validaciones llegaba al 69% y la peor era del 50%. Por tanto, es un tipo de modelo con predicciones más robustas y consistentes que el PLS y Random Forest con estos datos, aunque la precisión media sea ligeramente peor.

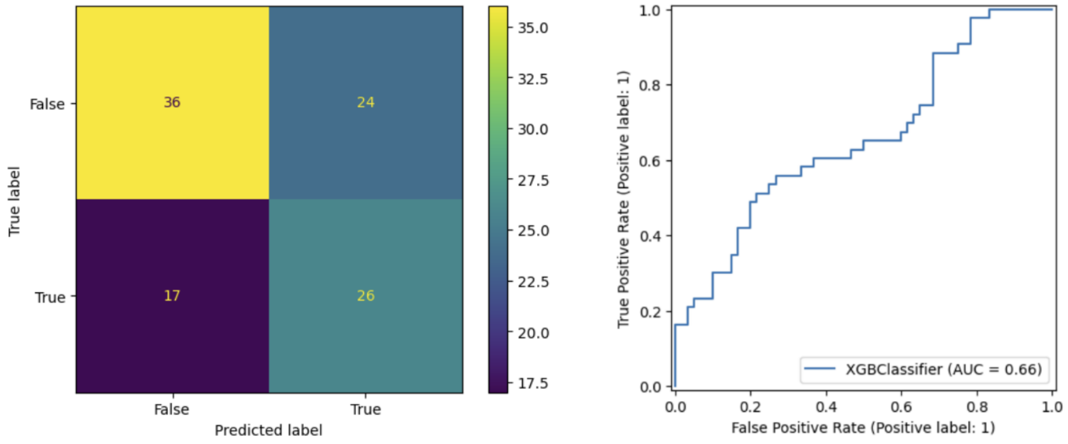

Figura 41. Matriz de confusión y curva ROC del Gradient boosted trees de 2 clases en la cuarta validación

4.3.2.4 SVM

En este apartado se comprueban los resultados obtenidos al aplicar un modelo de Máquinas de Vectores de soporte o SVM (Support Vector Machine). En cuanto a los ajustes de parámetros para conseguir las mejores clasificaciones de las conversaciones se ha optado por un núcleo o kernel de la función de base radial o RBF. La tolerancia para detener la optimización es la que tiene por defecto la función empleada de 0.001. No se imponen iteraciones máximas debido a la velocidad de cálculo observada. Las métricas de las predicciones del modelo con los parámetros finales en las distintas validaciones se resumen en la Tabla 12.

Tabla 12. Métricas de validación del modelo SVM para clasificación en 2 clases

SVM	Val. 1	Val. 2	Val. 3	Val. 4	Val. 5	Media
Precisión	0.52	0.47	0.5	0.6	0.6	0.54
Sensibilidad	0.47	0.28	0.33	0.4	0.39	0.37
Especificidad	0.57	0.61	0.63	0.75	0.76	0.66
AUC	0.58	0.43	0.52	0.62	0.6	0.55
F-score	0.45	0.31	0.35	0.45	0.45	0.4
Coef. Cor. Matthews	0.03	-0.12	-0.04	0.15	0.16	0.04

La precision media es solamente del 54%, habiendo 3 validaciones en torno al 50% (Figura 43) y otras dos con 60% de precisión (Figura 42). El resto de las métricas también son peores a los 3 modelos analizados previamente, por lo que este tipo de modelos se descarta.

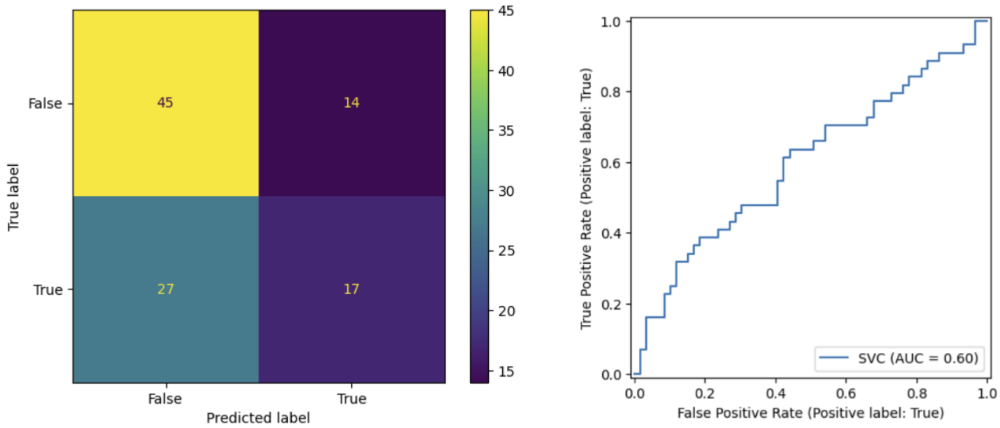

Figura 42. Matriz de confusión y curva ROC del SVM de 2 clases en la quinta validación

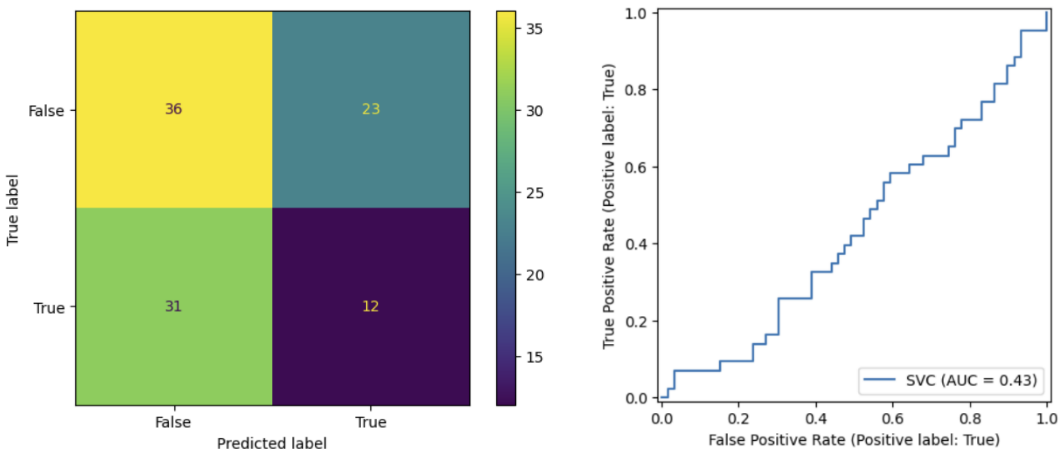

Figura 43. Matriz de confusión y curva ROC del SVM de 2 clases en la segunda validación

4.3.2.5 KNN

El modelo de clasificación basada en K vecinos más cercanos seleccionado tras realizar varias pruebas usa 5 vecinos y sus métricas de predicción están en la Tabla 13.

Tabla 13. Métricas de validación del modelo KNN para clasificación en 2 clases

KNN	Val. 1	Val. 2	Val. 3	Val. 4	Val. 5	Media
Precisión	0.5	0.41	0.57	0.58	0.53	0.52
Sensibilidad	0.37	0.3	0.4	0.63	0.61	0.46
Especificidad	0.6	0.49	0.7	0.55	0.47	0.56
AUC	0.48	0.41	0.56	0.62	0.58	0.53
F-score	0.39	0.3	0.44	0.56	0.53	0.44
Coef. Cor. Matthews	-0.03	-0.2	0.1	0.18	0.09	0.03

Los resultados del modelo son muy dispares según el conjunto de validación empleado. Si bien en el cuarto conjunto de validación (Figura 44) y en el tercero la precisión de las predicciones es cercana al 60%, en el segundo (Figura 45) es de solo un 41%. Por ello no se considera el uso de este modelo.

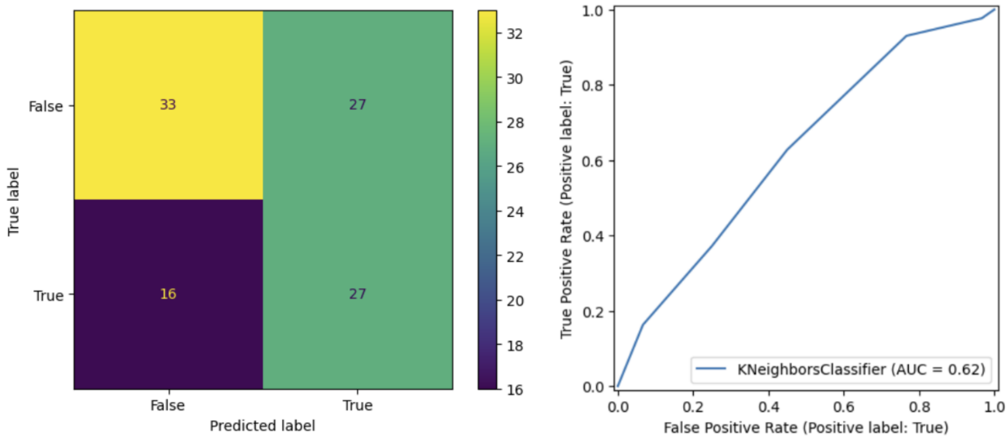

Figura 44. Matriz de confusión y curva ROC del KNN de 2 clases en la segunda validación

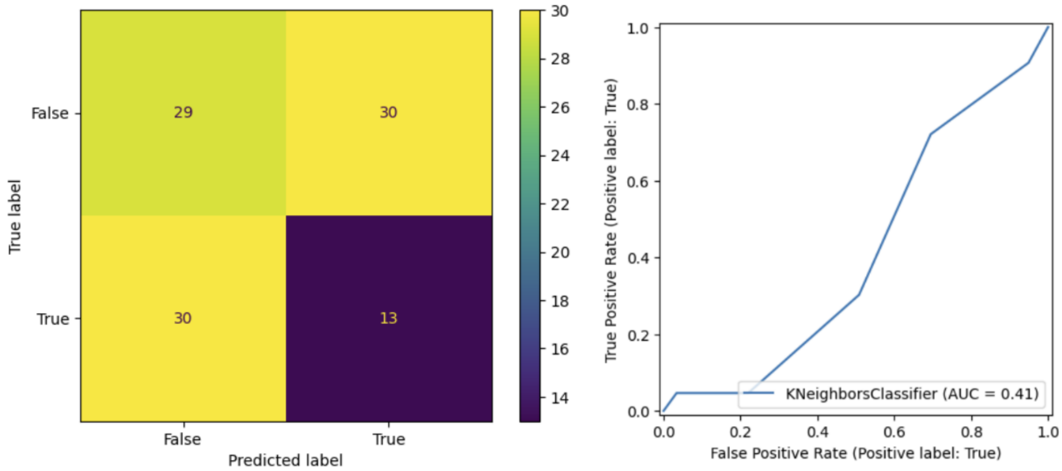

Figura 45. Matriz de confusión y curva ROC del KNN de 2 clases en la segunda validación

4.3.2.6 Gaussian Naive Bayes

El modelo Naive Bayes Gaussiano se parametriza definiendo únicamente la parte de la varianza mayor de todas las variables que se añade para calcular la estabilidad. Se deja el valor por defecto de 10-9, obteniendo los resultados de la Tabla 14.

Tabla 14. Métricas de validación del modelo Gaussian Naive Bayes para clasificación en 2 clases

Gaussian N. Bayes	**Val. 1**	**Val. 2**	**Val. 3**	**Val. 4**	**Val. 5**	**Media**
Precisión	0.45	0.47	0.52	0.58	0.58	0.52
Sensibilidad	0.33	0.23	0.56	0.51	0.18	0.36
Especificidad	0.53	0.64	0.5	0.63	0.88	0.64
AUC	0.43	0.44	0.53	0.57	0.53	0.5
F-score	0.33	0.27	0.49	0.5	0.27	0.37
Coef. Cor. Matthews	-0.14	-0.13	0.06	0.14	0.09	0

Las mejores precisiones se obtienen en la cuarta (Figura 46) y quinta validación (Figura 47). Sin embargo, con este modelo se ve claramente como no hay que fijarse únicamente en la precisión para valorar las predicciones de los modelos. Si bien en ambas validaciones tienen la misma precisión, en la quinta validación el F-score es casi la mitad. Esto es debido a que casi todas las predicciones predicen que los usuarios no tienen síntomas depresivos, como también se puede ver al tener una sensibilidad de solo 0.18 y una especificidad muy elevada del 0.88. Por lo tanto, realmente las predicciones para el quinto grupo de validación son peores a las del cuarto al ser equivalentes a si se fijase que todas las conversaciones no son de individuos con síntomas.

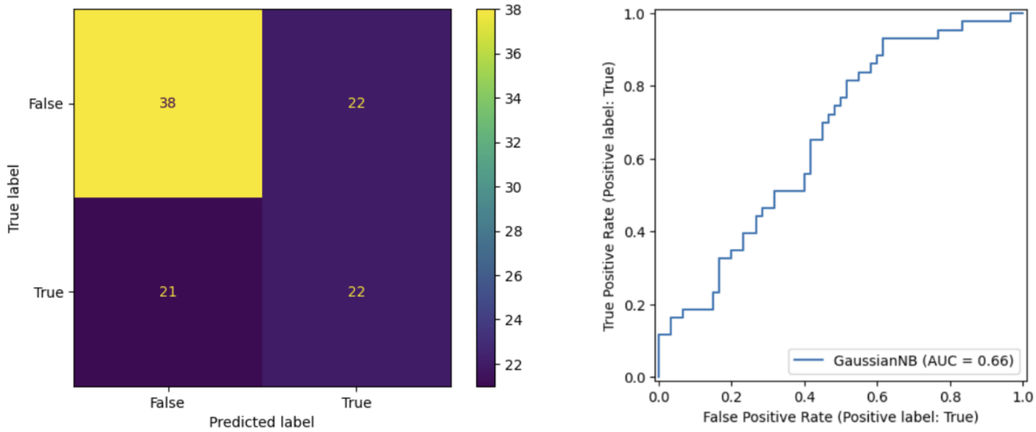

Figura 46. Matriz de confusión y curva ROC del Gaussian Naive Bayes de 2 clases en la cuarta validación

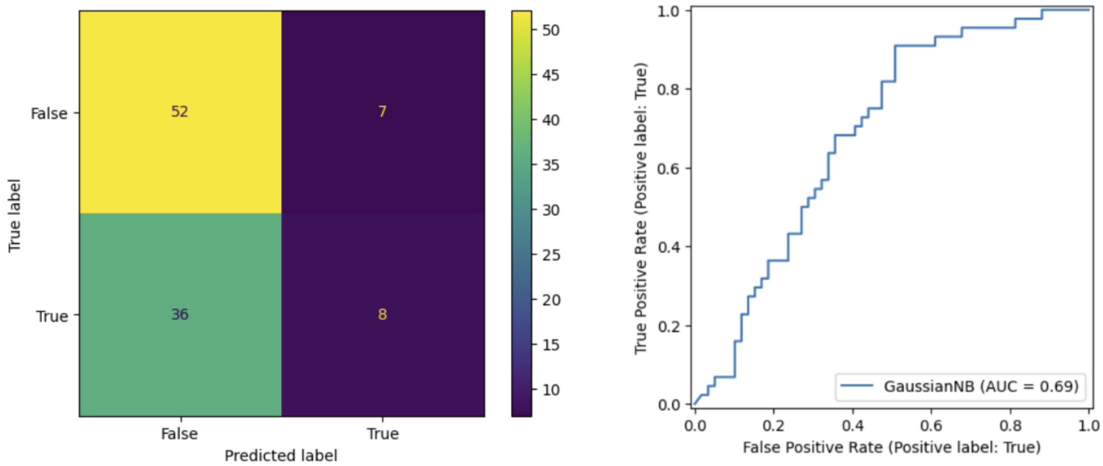

Figura 47. Matriz de confusión y curva ROC del Gaussian Naive Bayes de 2 clases en la quinta validación

El resto de las validaciones tienen una precisión mala, acertando en la primera (Figura 48) y segunda menos del 50%. La media de las predicciones es de 52% y se decide descartar este tipo de modelos.

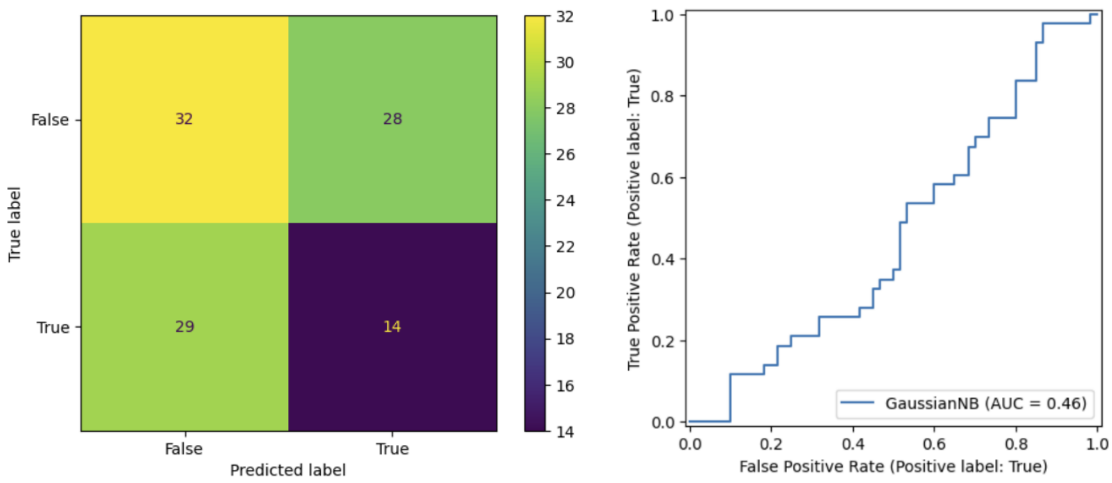

Figura 48. Matriz de confusión y curva ROC del Gaussian Naive Bayes de 2 clases en la primera validación

4.3.2.7 Perceptrón multicapa (MLP)

En un modelo de perceptrón multicapa es necesario ajustar tanto el ratio de aprendizaje como el objetivo para el optimizador. Se fija la entropía cruzada binaria como el objetivo y el ratio de aprendizaje en 0.01.

Respecto a los parámetros específicos de estos modelos es fundamental definir el número de capas, el tipo de función de activación de cada capa y la cantidad de neuronas por capa. Se determina que todas las capas tengan una función de activación ReLU o activación linear rectificada menos la capa de salida al ser un modelo de clasificación. La capa de salida tendrá una función de activación de tipo sigmoide para establecer de forma clara si la predicción es negativa (sin síntomas) o positiva.

Para establecer el número de capas y la cantidad de neuronas por capa se prueban entre 1 y 10 capas las cuales tienen desde 32 hasta 480 neuronas. Debido al elevado número de combinaciones posibles (más de 54 millones) se decide probar de forma aleatoria 100 de ellas. La mejor entre las probadas está compuesta por 3 capas de 192, 64 y 32 neuronas respectivamente además de la capa de salida. Otras de las mejores combinaciones han dado un error de predicción parecido a esta combinación, el cual es elevado y se puede ver en la Tabla 15.

Tabla 15. Métricas de validación del modelo Perceptrón multicapa para clasificación en 2 clases

MLP	Val. 1	Val. 2	Val. 3	Val. 4	Val. 5	Media
Precisión	0.58	0.52	0.51	0.63	0.53	0.55
Sensibilidad	0.63	0.28	0.4	0.6	0.36	0.45
Especificidad	0.55	0.69	0.6	0.65	0.66	0.63

AUC	0.59	0.48	0.53	0.65	0.51	0.55
F-score	0.56	0.33	0.4	0.57	0.4	0.45
Coef. Cor. Matthews	0.17	-0.02	0	0.25	0.03	0.08

La media de precisión no es especialmente buena al ser del 55%. Existe bastante diferencia entre la cuarta validación donde se consiguen las mejores predicciones con precisión del 63% (Figura 49) y la primera con el 58% respecto al resto de validaciones. Las predicciones del resto de validaciones tienen una precisión de en torno a 50%, por ejemplo, en la Figura 50 se ve la tercera validación y su 51% de precisión.

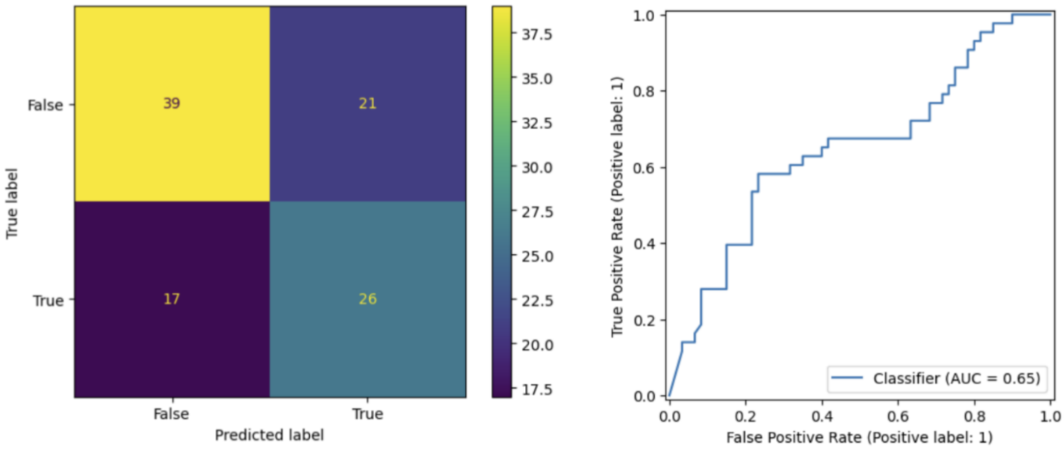

Figura 49. Matriz de confusión y curva ROC del Perceptrón multicapa de 2 clases en la cuarta validación

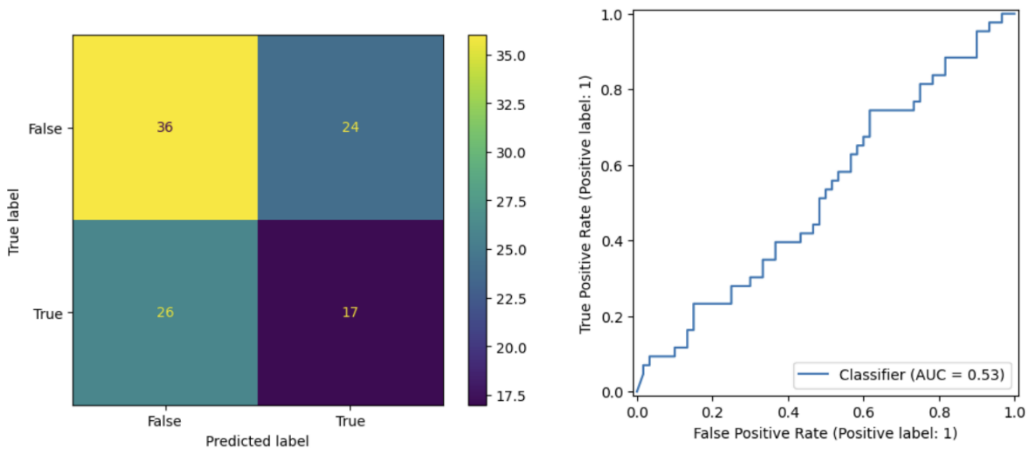

Figura 50. Matriz de confusión y curva ROC del Perceptrón multicapa de 2 clases en la tercera validación

Debido al gran esfuerzo de cálculo necesario para el ajuste de este tipo de modelos respecto al resto de los planteados y a que los resultados no son los mejores de forma clara se descarta su uso.

4.3.3 Clasificación 3 clases

En tercer lugar, se ajustan y comparan modelos de clasificación para 3 clases: con síntomas depresivos severos (clase 2), leves (clase 1) o sin síntomas depresivos (clase 0). La principal métrica usada para comparar los modelos es la precisión en las predicciones además del F-score.

Una vez ajustados todos los modelos planteados no hay ninguno cuya precisión media en las validaciones sea superior al 50%. Esta precisión es algo mejor que la obtenida en la clasificación de 2 clases ya que es un problema de clasificación más difícil en el que una predicción aleatoria daría como resultado un 33%. De todas formas, sigue habiendo muchos fallos, especialmente en los casos sin depresión o con mucha puntuación en el test PHQ. Por lo tanto, ningún modelo realiza predicciones con un nivel de precisión aceptable.

4.3.3.1 PLS-DA

El primer tipo de modelo de clasificación para 3 grupos probado son los PLS-DA. Tras unas pruebas se ha establecido que 3 componentes principales es el número más adecuado para el modelo, igual que sucedía al aplicar PLS a la predicción del PHQ o la clasificación en 2 clases. Se emplean las mismas variables significativas que se habían detectado en el apartado 5.3 para el valor numérico de PHQ. Probando con todas las variables los modelos resultantes tenían una precisión de media en las predicciones muy similar.

En la Tabla 16 se pueden ver las distintas métricas de las predicciones realizadas usando cada división de la base de datos como validación del modelo entrenado con las otras 4, así como la media de las 5 validaciones.

Tabla 16. Métricas de validación del modelo PLS-DA para clasificación en 3 clases

PLS-DA	Val. 1	Val. 2	Val. 3	Val. 4	Val. 5	Media
Precisión	0.43	0.49	0.43	0.49	0.58	0.48
F-score	0.37	0.44	0.39	0.46	0.55	0.44

La precisión promedio en las 5 validaciones es del 48%. Existe gran variabilidad entre las validaciones de la precisión, yendo desde 0.58 en la quinta (Figura 51) hasta solo 0.43 en la primera y segunda. El F-score también es también bajo, siendo su media de 0.44.

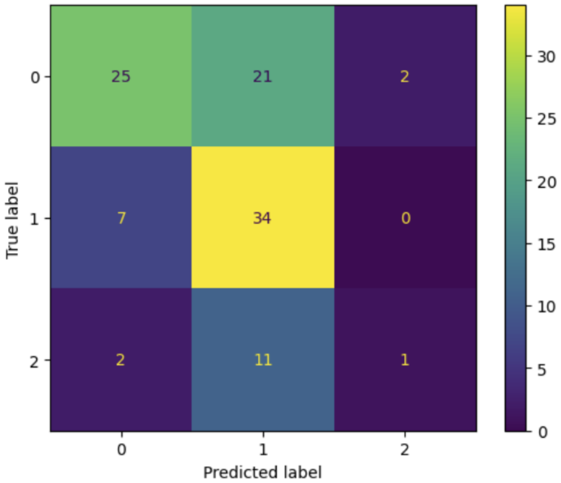

Figura 51. Matriz de confusión del PLS-DA de 3 clases en la quinta validación

4.3.3.2 Random Forest

El siguiente modelo ajustado es el de Random Forest clasificador. Para este tipo de modelos se van a probar una adicionalmente nueva estrategia propia de los problemas de clasificación multiclase denominada uno contra el resto (One vs Rest). En lugar de entrenar un modelo para identificar las conversaciones de cada clase se entrena un modelo por cada clase los cuales detectan si una conversación pertenece a una clase concreta.

Para comparar las dos estrategias se realiza la parametrización para cada una de forma independiente. Se han probado varias combinaciones de 8 los mismos parámetros de los modelos Random Forest usados para los otros objetivos. El objetivo era encontrar la combinación que tuviera la mayor precisión media en las predicciones con los distintos conjuntos de validación. Finalmente, no se han encontrado mejoras significativas de los modelos al cambiar por parámetros iniciales del algoritmo para ninguna de las dos estrategias. Por lo tanto, se fijan estos parámetros como los definitivos para las validaciones.

En resumen, los parámetros seleccionados son: criterio de gini, empleando Bootstrap sin observaciones fuera de la muestra, 100 árboles, sin máxima profundidad de árbol, dos mínimas muestras por Split y una por hoja, 100% de las características totales consideradas para cada split. Las métricas de las predicciones del modelo con los parámetros finales en las distintas validaciones se presentan en la Tabla 17.

Tabla 17. Métricas de validación del modelo Random forest para clasificación en 3 clases con la estrategia estándar

Random Forest	**Val. 1**	**Val. 2**	**Val. 3**	**Val. 4**	**Val. 5**	**Media**
Precisión	0.41	0.44	0.44	0.37	0.53	0.44
F-score	0.37	0.4	0.4	0.34	0.5	0.4

Tabla 18. Métricas de validación del modelo Random forest para clasificación en 3 clases con la estrategia uno contra el resto

Random Forest	Val. 1	Val. 2	Val. 3	Val. 4	Val. 5	Media
Precisión	0.43	0.43	0.45	0.36	0.61	0.46
F-score	0.39	0.39	0.41	0.34	0.57	0.42

La precisión media al usar la estrategia uno contra el resto es 2 centésimas mejor, pero ambas tienen valores inferiores a 0.5. Esta mejora viene producida por la mejora de precisión en la quinta validación, la cual pasa del 53% en el modelo con estrategia estándar a 61% con la estrategia específica para clasificación de más de dos clases. Por lo tanto, no se aprecia una mejora relevante respecto a usar un único modelo para identificar entre las 3 clases y no justifica el uso de esta estrategia.

Existe gran variabilidad entre las validaciones de la precisión, yendo desde 0.53 en la quinta hasta solo 0.37 en la cuarta. El F-score también es también bajo, siendo ligeramente inferior en todas las validaciones a la precisión. Por ejemplo, en la Figura 52 se aprecia como en la quinta validación la clase 3 es donde más fallos en la predicción se aprecian, mientras que la segunda clase se aciertan bastantes.

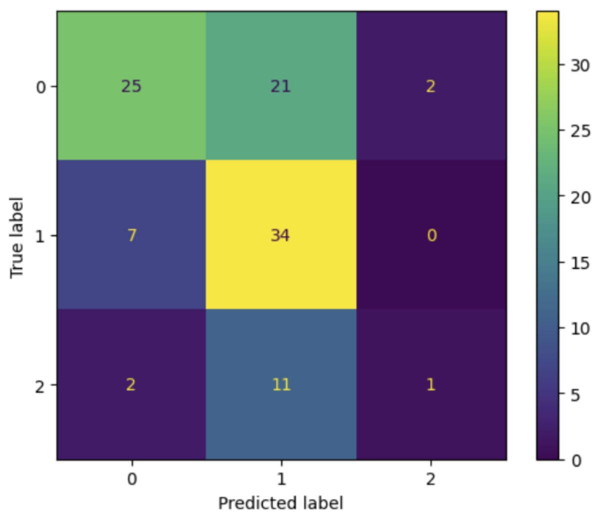

Figura 52. Matriz de confusión del Random Forest estándar de 3 clases en la quinta validación

4.3.3.3 Gradient boosted trees

A continuación, se ajusta el modelo bastado en Gradient boosted trees o árboles con potenciación del gradiente. Para conseguir el modelo con mejores predicciones de este tipo se han configurado los mismos 7 parámetros que para los objetivos anteriores. Primero se define la estructura del modelo con dos parámetros. El objetivo a minimizar del algoritmo es el logaritmo de la función de verosimilitud o “logloss”. Tras descartar el booster de tipo gblinear y el de tipo dart finalmente se usa el gbtree.

Para el resto de los parámetros se comienza con los valores por defecto de la función y se van probando nuevas combinaciones de sus valores para ver si alguna mejora significativamente la precisión de las predicciones. Entre estas combinaciones también se ha incluido la elegida para la clasificación de 2 clases. Con ninguna de ellas se ha superado

En definitiva, los parámetros seleccionados son: logloss como objetivo, gbtree como booster, tasa de aprendizaje de 1, gamma de 0.0001, sin máxima profundidad, peso mínimo de instancia del nodo hoja de 1 y como subconjunto se usa el 80% de los datos. Las métricas de las predicciones del modelo con los parámetros finales en las distintas validaciones se resumen en la Tabla 19.

Tabla 19. Métricas de validación del modelo Gradient boosted trees para clasificación en 3 clases

Grad. b. trees	Val. 1	Val. 2	Val. 3	Val. 4	Val. 5	Media
Precisión	0.4	0.45	0.42	0.41	0.5	0.44
F-score	0.38	0.41	0.4	0.41	0.48	0.42

La precisión media es del 44%, nuevamente muy baja. En la quinta validación (Figura 53) presenta su valor más alto, el cual es 50%.

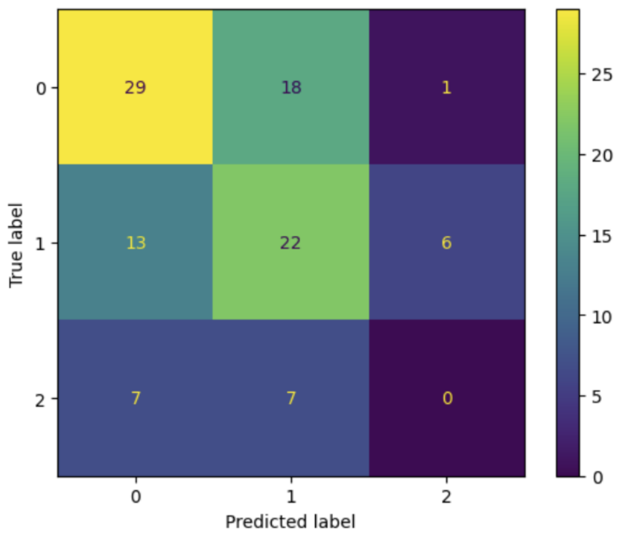

Figura 53. Matriz de confusión del Gradient boosted trees de 3 clases en la quinta validación

4.3.3.4 SVM

En este apartado se comprueban los resultados obtenidos al aplicar un modelo de Máquinas de Vectores de soporte o SVM (Support Vector Machine). En cuanto a los ajustes de parámetros para conseguir las mejores clasificaciones de las conversaciones se ha optado por un núcleo o kernel de la función de base radial o RBF. La tolerancia para detener la optimización es la que tiene por defecto la función empleada de 0.001. No se imponen iteraciones máximas debido a la velocidad de cálculo observada. Las métricas de las predicciones del modelo con los parámetros finales en las distintas validaciones se resumen en la Tabla 20.

Tabla 20. Métricas de validación del modelo SVM para clasificación en 3 clases

SVM	Val. 1	Val. 2	Val. 3	Val. 4	Val. 5	Media
Precisión	0.41	0.41	0.39	0.41	0.57	0.44
F-score	0.4	0.38	0.39	0.41	0.56	0.43

De nuevo la precisión de este tipo de modelo es muy baja, esta vez del 44%. Nuevamente la quinta validación (Figura 54) es la que tiene la mayor precisión, esta vez de un 57%.

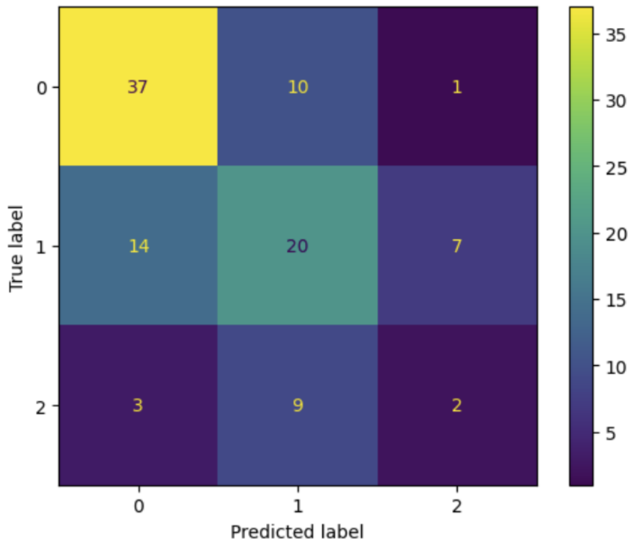

Figura 54. Matriz de confusión del SVM de 3 clases en la quinta validación

4.3.3.5 KNN

El modelo de clasificación basada en K vecinos más cercanos seleccionado tras realizar varias pruebas usa 5 vecinos. Se comienza por esta cantidad de vecinos al ser la elegida para el modelo de clasificación en 2 clases. Como ninguno de los otros valores probados mejora la precisión media se mantiene y sus métricas de predicción están en la Tabla 21.

Tabla 21. Métricas de validación del modelo KNN para clasificación en 3 clases

KNN	Val. 1	Val. 2	Val. 3	Val. 4	Val. 5	Media
Precisión	0.51	0.43	0.37	0.48	0.40	0.44
F-score	0.5	0.4	0.35	0.48	0.38	0.42

La media de precisión es del 44%, lo que no hace apto este tipo de modelo para la clasificación en 3 clases con estos datos. A diferencia de los tipos de modelo anteriormente discutidos, en lugar de en la quinta esta vez la mejor precisión se obtiene en la primera validación (Figura 55).

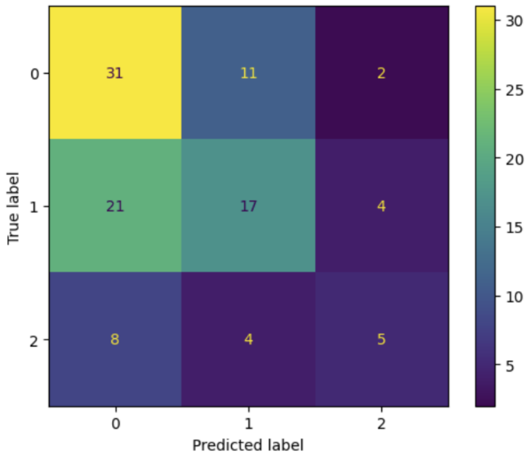

Figura 55. Matriz de confusión del KNN de 3 clases en la primera validación

4.3.3.7 Perceptrón multicapa (MLP)

En un modelo de perceptrón multicapa es necesario ajustar tanto el ratio de aprendizaje como el objetivo para el optimizador. Se fija la entropía cruzada binaria como el objetivo y el ratio de aprendizaje en 0.01.

Respecto a los parámetros específicos de estos modelos es fundamental definir el número de capas, el tipo de función de activación de cada capa y la cantidad de neuronas por capa. Se determina que todas las capas tengan una función de activación ReLU o activación linear rectificada menos la capa de salida al ser un modelo de clasificación. La capa de salida tendrá una función de activación de tipo sigmoide para establecer de forma clara si la predicción es negativa (sin síntomas) o positiva. A diferencia de cuando se clasifica entre 2 grupos, ahora con 3 grupos la última capa pasa de 1 a 3 neuronas. Cada una de estas neuronas determinará si la conversación pertenece a cada una de las clases por separado.

Para establecer el número de capas y la cantidad de neuronas por capa se prueban entre 1 y 10 capas las cuales tienen desde 32 hasta 480 neuronas. Partiendo de los parámetros de la red escogida para la clasificación de 2 clases se prueban aleatoriamente otras combinaciones. Al no encontrar ninguna que mejore significativamente los resultados se deja la combinación usada para clasificar en 2 clases. Por lo tanto, la red neuronal se compone 3 capas de 192, 64 y 32 neuronas respectivamente además de la capa de salida. Las métricas de las predicciones se encuentran en la Tabla 22.

Tabla 22. Métricas de validación del modelo MLP para clasificación en 3 clases

MLP	Val. 1	Val. 2	Val. 3	Val. 4	Val. 5	Media
Precisión	0.44	0.44	0.5	0.45	0.55	0.48
F-score	0.39	0.39	0.45	0.43	0.52	0.44

La precisión media del 48% es ligeramente superior a la obtenida con el resto de los modelos, aunque sigue siendo muy baja. En la quinta validación se obtiene 55% de precisión (Figura 56), siendo el valor más alto entre ellas.

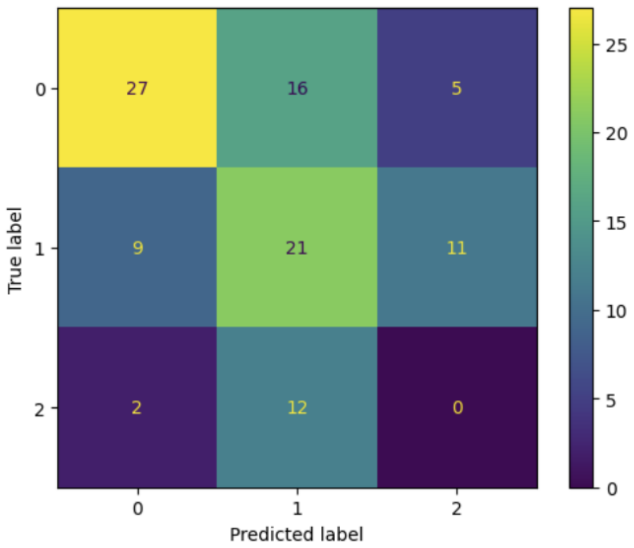

Figura 56. Matriz de confusión del MLP de 3 clases en la quinta validación

4.4. Modelos Deep Learning para series temporales de clasificación en 2 clases

En este apartado se presentan los modelos que usan como datos las series temporales extraídas durante las conversaciones para clasificación entre individuos con y sin síntomas depresivos. Se considera que el usuario muestra síntomas depresivos si su PHQ igual o mayor a 9, no mostrándolos si es inferior a esta cantidad.

Todas las variables empleadas se estandarizan para que puedan ser usadas por los modelos. Además, se añaden ceros al final de las series temporales de las conversaciones que no son la más larga para que todas las conversaciones tengan datos del mismo tamaño. Se realizan pruebas usando todas las variables o descartando alguno de los grupos de variables: parpadeos, movimiento de la cabeza o el lugar y duración de las fijaciones.

Se validan los modelos empleando la validación cruzada con los mismos 5 conjuntos usados en los modelos de las características extraídas. Con ello se evita que conversaciones de un individuo se empleen para entrenar y validar los modelos. Durante el ajuste del modelo se emplean sucesivamente 4 de ellos para entrenar los modelos y el restante para la validación. Este proceso se realiza 5 veces, empleando los 5 grupos para validar los modelos 1 vez.

Ninguno de los modelos ofrece buenas predicciones, teniendo todos precisiones medias cercanas al 50%. Sí que hay diferencias en el tiempo de computación necesario para el ajuste o en la consistencia de los resultados entre los diversos grupos de validación. A continuación, se detallan los ajustes elegidos para cada uno de los modelos y los resultados observados.

4.4.1 MiniRocket

El modelo de tipo MiniRocket es el que mejor precisión media consigue al tener un valor del 54,29%. Se emplean todas las variables, ya que al usar solo el movimiento de la cabeza o las fijaciones las predicciones empeoran. En cuanto a sus parámetros tras diversas pruebas no se aprecian mejoras al variar el número de características usadas partiendo de las 10000 iniciales o las 32 dilaciones máximas por kernel. En la primera validación (Figura 57) se consigue un 60.19% de precisión, el valor más alto de todos.

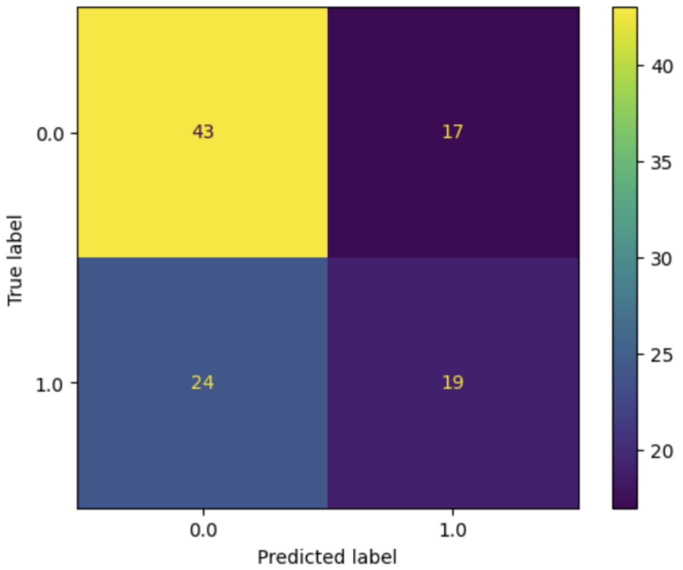

Figura 57. Matriz de confusión del modelo MiniRocket de 2 clases en la primera validación

En la segunda validación se consigue una precisión similar del 56,79%. Sin embargo, en la tercera y cuarta validaciones se predice que todas las conversaciones pertenecen a usuarios con síntomas depresivos. En la quinta validación sucede al revés, clasificando todas las conversaciones como pertenecientes a usuarios sin síntomas depresivos. Por lo tanto, este modelo es muy inconsistente entre las distintas validaciones y no es adecuado para los datos, aunque la precisión media sea elevada debido a las dos primeras validaciones.

4.4.2 Transformer

Las predicciones obtenidas con el Transformer tienen una precisión media usando todas las variables del 52,6%. En este modelo tampoco se mejora la precisión de forma significativa descartando algunas de las variables, empleando los mismos parámetros del modelo durante las diversas pruebas.

En cuanto a los parámetros seleccionados el modelo se optimiza durante 25 épocas o “epoch” y un ratio de aprendizaje de 0.00002. El ratio de aprendizaje de este tipo de modelos suele ser inferior al ratio de aprendizaje de otros modelos de Deep Learning. Para poder entrenar el modelo en el equipo empleado con 16 GB de RAM y 16 GB de VRAM se fija el tamaño de lote en 8 y la longitud de secuencia máxima en 2000.

El resto de los parámetros del modelo se han dejado con los valores por defecto en la librería empleada. La dimensión total del modelo es de 128 características. Existen 16 cabezas de atención paralelas. La dimensión de la red neuronal de propagación hacia delante es de 256 y su función de activación en las capas intermedias es “gelu”. El número de capas intermedias del sub-encoder es 3.

Para evitar el sobreajuste durante el entrenamiento se ha probado a incrementar el dropout del encoder desde el 10% inicial hasta el 30%. También se ha probado a aplicar un 10% de dropout a la capa final totalmente conectada. Ambos cambios realizados por separado y conjuntamente no mejoran las predicciones por lo que no se aplican en los parámetros finales.

Tabla 23. Métricas de validación del modelo Transformer para clasificación en 2 clases

Transformer	**Val. 1**	**Val. 2**	**Val. 3**	**Val. 4**	**Val. 5**	**Media**
Precisión	0.57	0.6	0.41	0.53	0.52	0.52
Sensibilidad	0.42	0.49	0.12	0.33	0.2	0.31
Especificidad	0.68	0.68	0.6	0.67	0.77	0.68
AUC	0.55	0.58	0.36	0.5	0.49	0.5
F-score	0.45	0.51	0.14	0.37	0.27	0.35
Coef. Cor. Matthews	0.1	0.17	-0.3	0	-0.03	-0.012

En la Tabla 23 se puede observar como el modelo predice mejor la primera y la segunda validación (Figura 58). En cambio, la tercera validación (Figura 59) la predice muy mal, siendo su precisión del 41%. Como en los modelos de clasificación usando las características extraídas, la especificidad es mayor que la sensibilidad de forma consistente entre las distintas validaciones y significativa. La diferencia es elevada, llegando a ser 0.57 superior la especificidad en la quinta validación respecto a la sensibilidad.

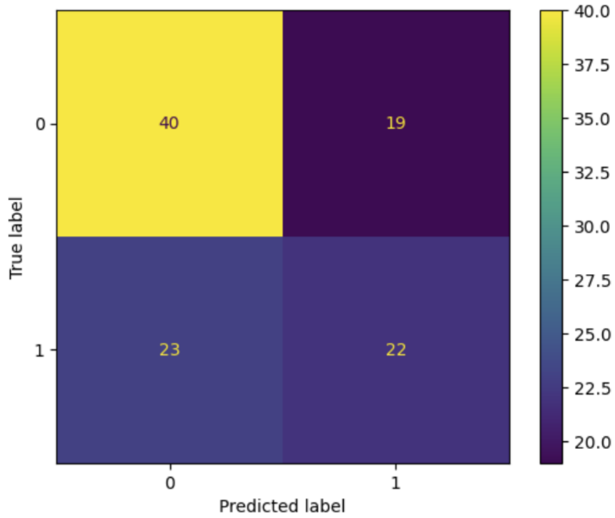

Figura 58. Matriz de confusión del Transformer de 2 clases en la segunda validación

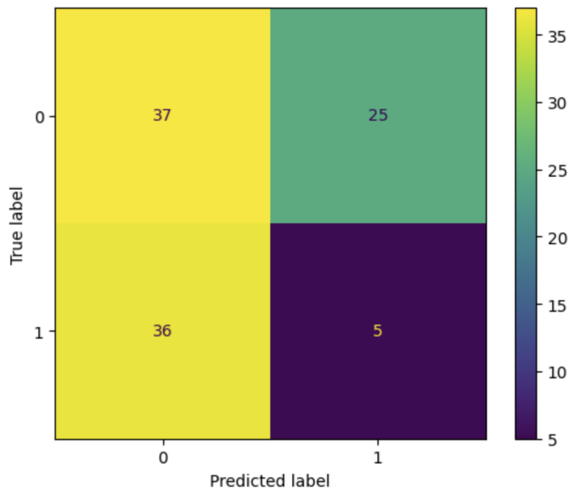

Figura 59. Matriz de confusión del Transformer de 2 clases en la tercera validación

4.4.3 LSTM

Respecto a modelo LSTM, ninguna de las pruebas realizadas con diversos parámetros de su configuración ha logrado que se predigan individuos en ambas clases. Todos los modelos predicen que todas las observaciones pertenecen a individuos que no tienen síntomas depresivos. Se han usado tanto todas las variables como descartando alguno de los 3 grupos de variables sin haber cambios en los resultados.

En cuanto a la estructura del modelo se ha probado desde una capa hasta 5. El tamaño de estas capas ha comenzado en 100 neuronas y se ha aumentado hasta 300, sin observar cambios significativos en la respuesta del modelo. En todos los casos se ha probado tanto a procesar los datos en una dirección como en ambas, lo que se denomina bidireccional. Además, también se ha probado con distintas cantidades de dropout a la capa final totalmente conectada llegando hasta el 50% sin observar cambios en las predicciones. Se ha considerado incrementar la cantidad de épocas o "epoch" durante la optimización, pero a partir de 10 el modelo no evoluciona más y los valores de las métricas de evaluación se estabilizan.

4.4.4 GRU

El modelo GRU se comporta similar al modelo LSTM, el cual también en un modelo del tipo red neuronal recurrente. En lugar de predecir siempre que todas las observaciones pertenecen a individuos que no tienen síntomas depresivos, con este modelo según el conjunto de validación se predice que casi todas las conversaciones son de alguna de las dos clases. Concretamente en los 3 primeros grupos de validación se ha predicho que todas las conversaciones pertenecen a usuarios sin síntomas depresivos. En la cuarta validación todas se han clasificado sin síntomas depresivos menos 5 conversaciones de usuarios sin síntomas depresivos que se han clasificado con síntomas depresivos. En cambio, en la quinta validación se ha predicho que todas las conversaciones pertenecen a usuarios con síntomas depresivos.

Se han usado tanto todas las variables como descartando alguno de los 3 grupos de variables sin haber cambios en los resultados. En cuanto a la estructura del modelo se ha probado desde una capa hasta 3. El tamaño de estas capas ha comenzado en 100 neuronas y se ha aumentado hasta 300, sin observar cambios significativos en la respuesta del modelo. En todos los casos se ha probado tanto a procesar los datos en una dirección como en ambas, lo que se denomina bidireccional. Además, también se ha probado con distintas cantidades de dropout a la capa final totalmente conectada llegando hasta el 50% sin observar cambios en las predicciones. Se ha considerado incrementar la cantidad de épocas o "epoch" durante la optimización, pero a partir de las primeras 3 o 4 el modelo no evoluciona más y los métricas de evaluación se estabilizan.

4.4.5 FCN

Respecto al modelo de red neuronal convolucional (FCN) la mejor precisión media que se ha obtenido es del 51%. La estructura del modelo es de 3 capas las cuales tienen 128 neuronas salvo la capa intermedia que tiene 256. Modelos con mayor número de neuronas o capas no han mejorado los resultados de forma significativa. Se ha descartado incluir dropout en las capas finalmente conectadas tras probar valores de hasta el 50% y no ver mejoras en las predicciones. Se han empleado 50 de épocas o "epoch" durante la optimización al ser la cantidad donde se estabiliza el modelo.

Se han empleado todas las variables al no apreciar mejoras significativas al descartar algunas. En la Tabla 24 se pueden observar las métricas de las predicciones para cada validación.

Tabla 24. Métricas de validación del modelo FCN para clasificación en 2 clases

FCN	Val. 1	Val. 2	Val. 3	Val. 4	Val. 5	Media
Precisión	0.54	0.54	0.45	0.52	0.5	0.51
Sensibilidad	0.58	0.53	0.27	0.02	0	0.28
Especificidad	0.52	0.54	0.56	0.87	0.89	0.68
AUC	0.55	0.54	0.42	0.45	0.45	0.48
F-score	0.52	0.5	0.28	0.04	0	0.27
Coef. Cor. Matthews	0.1	0.08	-0.17	-0.19	-0.22	-0.13

Las predicciones para la cuarta y quinta validaciones son muy malas ya que casi todas las predicciones son de usuarios sin síntomas depresivos, como se ve en la Figura 60. En las dos primeras validaciones las predicciones son algo mejores pero debido a la falta de consistencia del modelo se determina que el modelo no es adecuado.

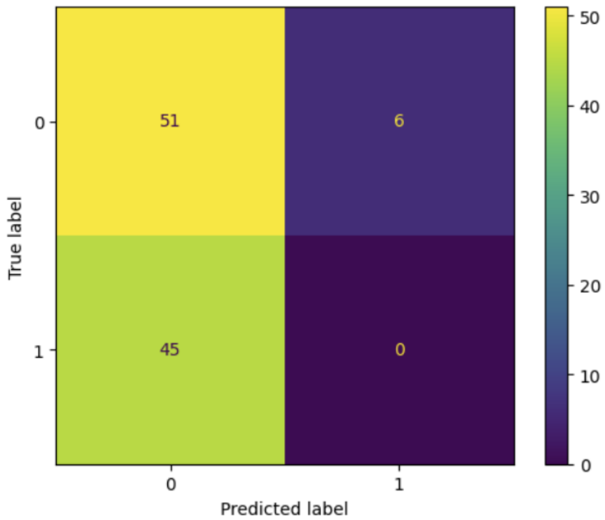

Figura 60. Matriz de confusión del FCN de 2 clases en la quinta validación

4.4.6 Residual Network

Con el modelo de Residual Network se ha conseguido una precisión media del 52%. Se escoge un modelo con 64 filtros al obtener peores resultados con menos cantidad. También se incluye dropout en las capas finalmente conectadas del 30% al apreciar una ligera mejora en las predicciones. Se han empleado 50 de épocas o “epoch” durante la optimización al ser la cantidad donde se estabiliza el modelo.

Se han empleado todas las variables al no apreciar mejoras significativas al descartar algunas. En la Tabla 25 se pueden observar las métricas de las predicciones para cada validación.

Tabla 25. Métricas de validación del modelo Residual Network para clasificación en 2 clases

Residual Network	Val. 1	Val. 2	Val. 3	Val. 4	Val. 5	Media
Precisión	0,59	0,58	0,46	0,55	0,44	0,52
Sensibilidad	0,67	0,51	0,29	0,07	1	0,51
Especificidad	0,53	0,63	0,56	0,88	0	0,52
AUC	0,6	0,57	0,43	0,48	0,5	0,52
F-score	0,58	0,51	0,3	0,12	0,61	0,42
Coef. Cor. Matthews	0,21	0,14	-0,14	-0,07	0	0,03

En la primera y segunda validaciones el modelo se comporta relativamente bien comparado con otros modelos. Sin embargo, en la cuarta predice que casi todas las conversaciones son de usuarios sin síntomas depresivos y en la quinta que todas las conversaciones pertenecen a usuarios con síntomas depresivos. Por lo tanto, las predicciones del modelo son muy inconsistentes.

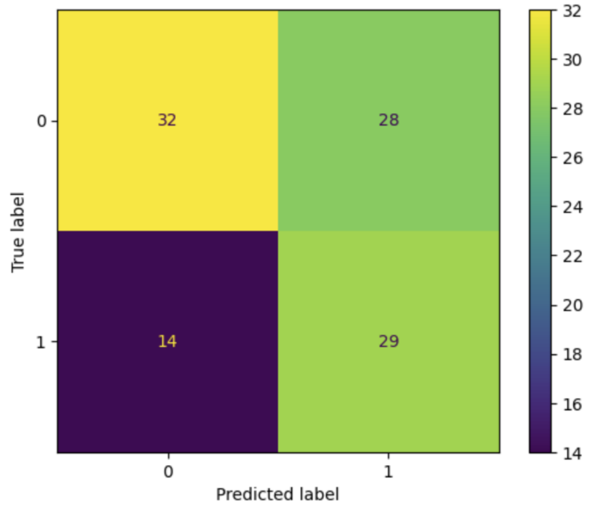

Figura 61. Matriz de confusión del Residual Network de 2 clases en la primera validación

4.4.7 Inception Time

Por último, el modelo Inception Time se comporta de forma similar al de Residual Network. Se obtiene la misma precisión media de 52% con uno modelo con menor cantidad de filtros, ya que esta vez es de 32. No se ha empleado dropout ni en las capas finalmente conectadas ni en el enconder. Se han empleado todas las variables al no apreciar mejoras significativas al descartar algunas. Las métricas de las predicciones en la validación tras ajustar el modelo con 50 épocas se encuentran en la Tabla 26.

Tabla 26. Métricas de validación del modelo Inception Time para clasificación en 2 clases

Inception Time	Val. 1	Val. 2	Val. 3	Val. 4	Val. 5	Media
Precisión	0.59	0.56	0.48	0.52	0.44	0.52
Sensibilidad	0.58	0.47	0.27	0.05	1	0.47
Especificidad	0.6	0.63	0.61	0.85	0	0.54
AUC	0.59	0.55	0.44	0.45	0.5	0.51
F-score	0.54	0.48	0.29	0.08	0.61	0.4
Coef. Cor. Matthews	0.18	0.09	-0.12	-0.16	0	0

Igual que con el modelo de Residual Network, el modelo funciona mejor en las dos primeras validaciones. Además, en la cuarta nuevamente se establece que casi todos los casos son negativos mientras que en la quinta el 100% de los casos se predice que son positivos.

4.5. Modelos Deep Learning para series temporales de estimación PHQ

Tras ver el mal rendimiento de los modelos de Deep Learning para la clasificación de los usuarios sobre los datos de las series temporales se aplican estos modelos para la predicción del valor numérico de PHQ. Se prueban los mismos 7 tipos de modelos de Deep Learning empleados para la clasificación, realizando pruebas entre distintos valores de los mismos parámetros ajustados anteriormente.

También se estudia si alguno de los grupos de variables es mejor que usar todas al mismo tiempo, pero no se observan cambios importantes en las predicciones. Se validan los modelos empleando la validación cruzada con los mismos 5 conjuntos usados en los modelos de las características extraídas y en los de Deep Learning para clasificación. Con ello se evita que conversaciones de un individuo se empleen para entrenar y validar los modelos. Además, se evita que los modelos enfocados a clasificación tengan un rendimiento distinto a los de regresión explorados en este apartado solo por el hecho de usar otros conjuntos para el entrenamiento y validación.

Se extrae como métrica de validación de los modelos el error cuadrático medio de predicción. Sin embargo, esta métrica no es muy útil ya que ningún modelo genera predicciones que se ajusten a los valores reales mínimamente. En la mayoría de los modelos las predicciones resultantes al ser graficadas forman una nube de puntos, mostrando el carácter aleatorio e impredecible de las mismas. Por lo tanto, como ya pasaba en el apartado anterior, ninguno de los modelos de Deep Learning es lo suficientemente bueno como para considerar su uso con los datos.

Aunque todos los modelos tengan mal comportamiento, existen algunas diferencias entre ellos. Los modelos LSTM y GRU independientemente de los parámetros usados predicen que todos los usuarios en todas las conversaciones tienen un PHQ de 9 con muy poca dispersión. Por ejemplo, en la Figura 62 se pueden observar las predicciones del modelo GRU en las distintas validaciones.

Esta homogeneidad en las predicciones es similar a lo que sucedía con este tipo de modelos cuando clasificaban que casi todas las conversaciones de cada grupo de validación pertenecían al mismo tipo de usuario.

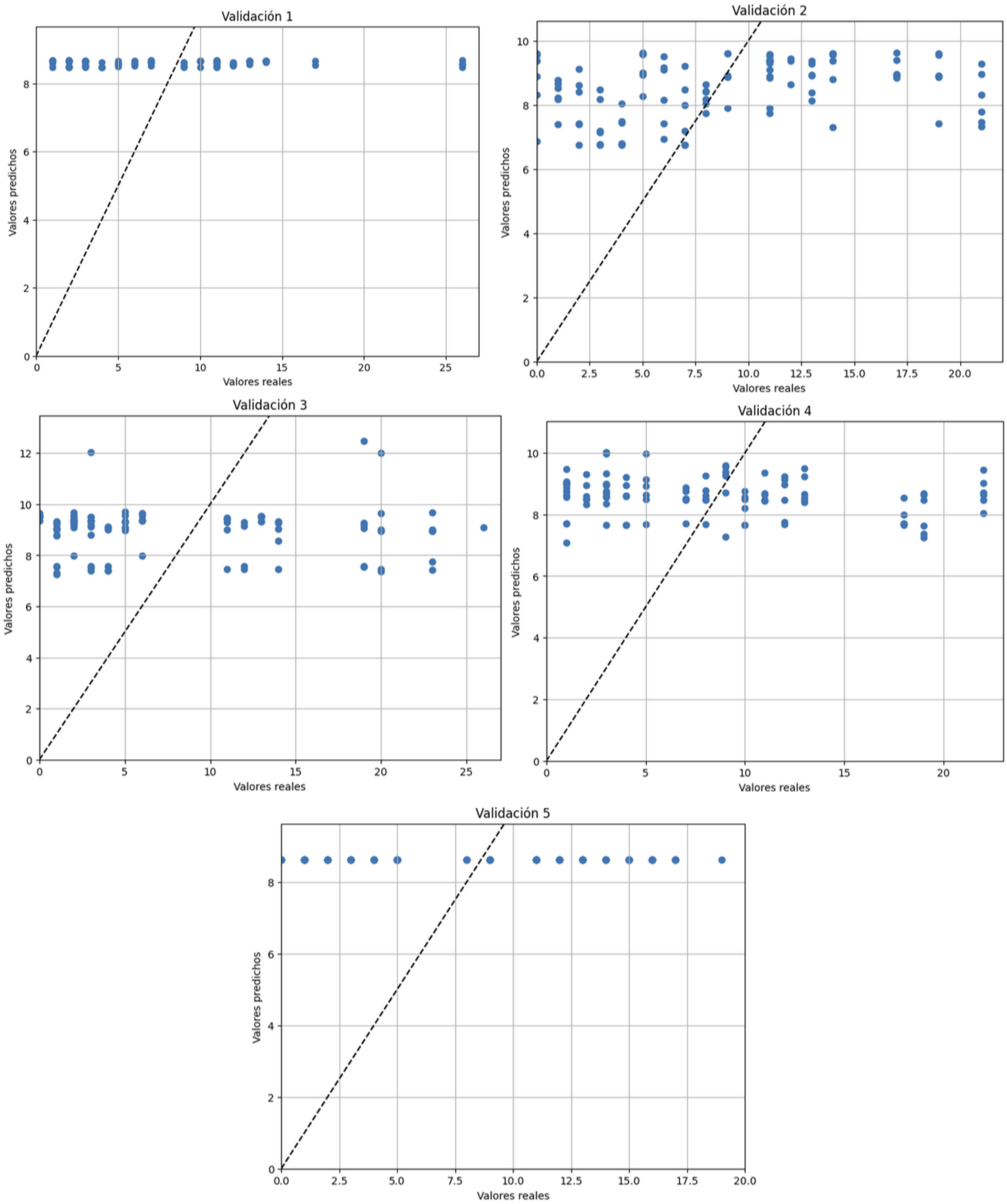

Figura 62. PHQ predicho por modelo GRU respecto al real en todas las validaciones

El resto de los modelos sí que realizan predicciones con mucha variabilidad sin un patrón claro. En las 4 primeras validaciones todos los modelos son similares y ofrecen predicciones dentro del rango razonable de PHQ, viéndose por ejemplo en la Figura 63 las predicciones del modelo Inception Time.

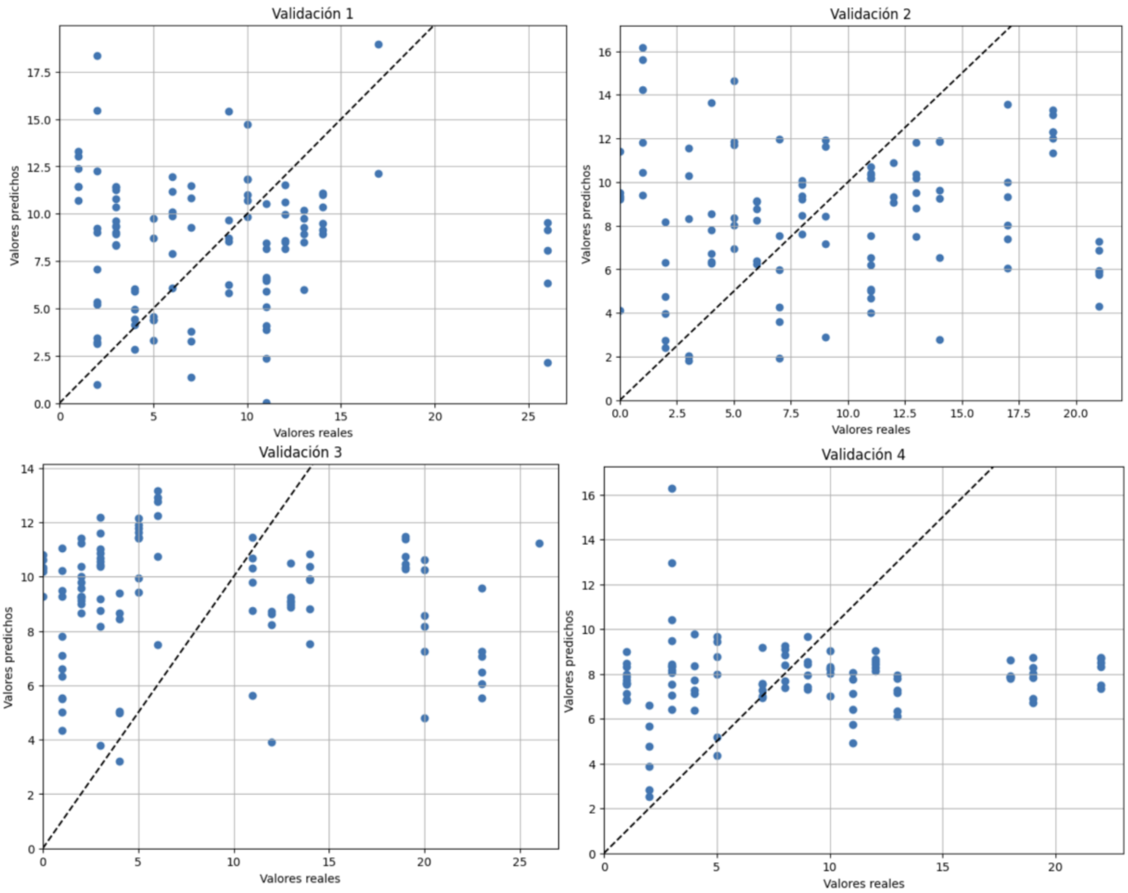

Figura 63. PHQ predicho por modelo Inception Time respecto al real en las cuatro primeras validaciones

Mientras tanto en la quinta validación todos los modelos dan valores extremos de PHQ muy alejados del rango real. En los modelos MiniRocket, FCN y Residual Network las predicciones de PHQ son superiores a 50 en muchos casos, valores que no tiene sentido lógico. En cambio, con los modelos Inception Time y Transformer la mayoría de las predicciones son 0 o valores muy similares.

5. Conclusiones

En conclusión, no se ha encontrado ningún modelo que realice predicciones del valor numérico del PHQ o que clasifique los individuos correctamente según sus síntomas depresivos. Las mejores predicciones obtenidas no han sido mucho mejor que las obtenidas si se hubieran realizado totalmente al azar, no siendo capaces los modelos de aprovechar ni las series temporales en bruto ni las características extraídas de las conversaciones.

En cuanto a las características extraídas que sintetizan cada conversación sí que se han podido extraer relaciones entre ellas gracias al análisis PCA. Con este análisis también se pueden detectar conversaciones anómalas que merecería la pena estudiar detenidamente. A través de esta información se podrían mejorar los avatares virtuales al poder identificar problemas que de otra manera no se detectarían.

Cabe destacar el buen resultado del modelo Random Forest cuando se mezclan conversaciones del mismo usuario en su entrenamiento y predicción. Esto es un indicativo de que existen diferencias significativas entre los individuos lo cual permite predecir los síntomas depresivos del usuario que realiza una conversación prediciendo a que usuario pertenece esta conversación. Estas diferencias dificultan agrupar el comportamiento de las personas con problemas depresivos ya que existen grandes diferencias entre ellas.

Pese a no haber funcionado ninguno de los modelos con los datos actuales no habría que descartar totalmente esta vía de investigación. Se podrían introducir nuevos modelos de Deep Learning híbridos con arquitecturas más complejas que los probados, como por ejemplo la combinación CNN-LSTM. Además, se podría estudiar la incorporación de datos adicionales de las conversaciones con otros sensores o información adicional de los individuos. También podría emplearse el mismo enfoque y modelos para el estudio del movimiento de la mirada en otro tipo de conversaciones con el objetivo de diagnosticar otras enfermedades.

6. Bibliografía

[1] J. Llanes-Jurado, L. Gómez-Zaragozá, M. E. Minissi, M. Alcañiz y J. Marín-Morales, «Developing conversational Virtual Humans for social emotion elicitation based on large language models,» *Expert Systems with Applications,* vol. 246, p. 123261, 2024.

[2] K. Kroenke, R. L. Spitzer and J. B. Williams, "The PHQ-9: validity of a brief depression severity measure," *J Gen Intern Med,* vol. 16, p. 606–613, September 2001.

[3] J. Marín-Morales, J. Llanes-Jurado, M. E. Minissi, L. Gómez-Zaragozá, A. Altozano y M. Alcañiz, «Gaze and Head Movement Patterns of Depressive Symptoms During Conversations with Emotional Virtual Humans,» de *2023 11th International Conference on Affective Computing and Intelligent Interaction (ACII)*, 2023.

[4] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke y T. E. Oliphant, «Array programming with NumPy,» *Nature,* vol. 585, p. 357–362, September 2020.

[5] T. pandas development team, *pandas-dev/pandas: Pandas,* Zenodo, 2024.

[6] T. M. D. Team, *Matplotlib: Visualization with Python,* Zenodo, 2024.

[7] M. L. Waskom, «seaborn: statistical data visualization,» *Journal of Open Source Software,* vol. 6, p. 3021, 2021.

[8] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot y E. Duchesnay, «Scikit-learn: Machine Learning in Python,» *Journal of Machine Learning Research,* vol. 12, p. 2825–2830, 2011.

[9] T. Chen y C. Guestrin, «XGBoost: A Scalable Tree Boosting System,» de *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA, 2016.

[10] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu y X. Zheng, *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems,* 2015.

[11] I. Oguiza, *tsai - A state-of-the-art deep learning library for time series and sequential data,* 2023.

[12] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai y S. Chintala, *PyTorch: An Imperative Style, High-Performance Deep Learning Library,* 2019.

[13] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever y R. Salakhutdinov, «Dropout: A Simple Way to Prevent Neural Networks from Overfitting,» *Journal of Machine Learning Research,* vol. 15, pp. 1929-1958, June 2014.

[14] A. Dempster, D. F. Schmidt y G. I. Webb, «MiniRocket A Very Fast (Almost) Deterministic Transform for Time Series Classification,» de *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2021.

[15] G. Zerveas, S. Jayaraman, D. Patel, A. Bhamidipaty y C. Eickhoff, «A Transformer-based Framework for Multivariate Time Series Representation Learning,» de *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, New York, NY, USA, 2021.

[16] S. Hochreiter y J. Schmidhuber, «Long Short-Term Memory,» *Neural Computation,* vol. 9, pp. 1735-1780, 1997.

[17] J. Chung, Ç. Gülçehre, K. Cho y Y. Bengio, «Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling,» *CoRR,* vol. abs/1412.3555, 2014.

[18] J. Long, E. Shelhamer y T. Darrell, «Fully Convolutional Networks for Semantic Segmentation,» *CoRR,* vol. abs/1411.4038, 2014.

[19] Z. Wang, W. Yan y T. Oates, «Time Series Classification from Scratch with Deep Neural Networks: A Strong Baseline,» *CoRR,* vol. abs/1611.06455, 2016.

[20] K. He, X. Zhang, S. Ren y J. Sun, «Deep Residual Learning for Image Recognition,» *CoRR,* vol. abs/1512.03385, 2015.

[21] H. I. Fawaz, B. Lucas, G. Forestier, C. Pelletier, D. F. Schmidt, J. Weber, G. I. Webb, L. Idoumghar, P.-A. Muller y F. Petitjean, «InceptionTime: Finding AlexNet for Time Series Classification,» *CoRR,* vol. abs/1909.04939, 2019.

7. Anexos

<u>Anexo I.</u> Relación del trabajo con los Objetivos de Desarrollo Sostenible de la agenda 2030.

Grado de relación del trabajo con los Objetivos de Desarrollo Sostenible (ODS).

Objetivos del Desarrollo Sostenible	Alto	Medio	Bajo	No proc.
1. Fin de la pobreza.				X
2. Hambre cero.				X
3. Salud y bienestar.	X			
4. Educación de calidad.				X
5. Igualdad de género.				X
6. Agua limpia y saneamiento.				X
7. Energía asequible y no contaminante.				X
8. Trabajo decente y crecimiento económico.				X
9. Industria, innovación e infraestructuras.		X		
10. Reducción de las desigualdades.	X			
11. Ciudades y comunidades sostenibles.				X
12. Producción y consumo responsables.				X
13. Acción por el clima.				X
14. Vida submarina.				X
15. Vida de ecosistemas terrestres.				X
16. Paz, justicia e instituciones sólidas.				X
17. Alianzas para lograr objetivos.				X

El presente trabajo se encuentra muy relacionado con el **ODS 3: Salud y bienestar** y el **ODS 10: Reducción de las desigualdades**. Esto es debido a que el objetivo principal del trabajo es facilitar el diagnóstico de la depresión a través de un método automático. Con un proceso de este tipo se puede diagnosticar a mayor número de personas y de forma más rápida con los mismos recursos. Un diagnóstico temprano de esta enfermedad permite un tratamiento eficaz, mejorando la salud y el bienestar tanto de las personas directamente afectadas como de su entorno. Además, una forma automática de diagnóstico que pueda realizarse idealmente desde un dispositivo móvil reduciría las desigualdades entre la gente que pueda ir a un psicólogo a realizar el diagnóstico inicial y la gente que por su situación social o económica no pueda ir al psicólogo.

Otro objetivo que está relacionado con el trabajo es el **ODS 4: Industria, innovación e infraestructuras**, concretamente en relación con la innovación. En el trabajo se emplean técnicas innovadoras para la modelización, tanto sobre características extraídas mediante minería de datos o con técnicas de Deep Learning sobre las series temporales. El uso de estas técnicas se amplía cada vez más a diversos campos, entre ellos el de la salud humana.

<u>Anexo II.</u> Fichero Python con exploración inicial características extraídas y los modelos de minería que las emplean.

Fichero empleado para la importación de las características extraídas de las conversaciones en Python, exploración inicial y modelos machine learning:

https://drive.google.com/drive/folders/1h8v2x5q7IxGrfiJ9EsbcugN3KqT7Wt6q?usp=sharing

<u>Anexo III.</u> Ficheros de AspenPro MV con el análisis PCA y PLS.

Archivos del programa AspenPro MV con el análisis PCA y PLS respectivamente sobre las características extraídas de las conversaciones:

https://drive.google.com/drive/folders/1YTAGYXVB9On105qftoi5lszYfpAwPhWG?usp=sharing

<u>Anexo IV.</u> Ficheros Python con importación de las series temporales y los modelos Deep Learning.

Ficheros de Python empleados para realizar los modelos sobre las series temporales. En un fichero se realiza la importación de los datos y su transformación. Posteriormente para entrenar cada uno de los 7 modelos se emplea dos archivo independientes, uno para cada objetivo (clasificación o regresión):

https://drive.google.com/drive/folders/1QxOjr9hTj4_koJXE-8tucCEYRZlcRqm9?usp=sharing