CLASIFICACIÓN DE NIVELES DE OBESIDAD MEDIANTE MODELOS DE APRENDIZAJE AUTOMÁTICO: UN ENFOQUE PREDICTIVO BASADO EN DATOS

Simón Olivieri
Leandro Urturi
Camila Muñoz

19 de octubre del 2024

Facultad de Ciencias Exactas y Naturales y Agrimensura
UNIVERSIDAD NACIONAL DEL NORDESTE
FaCENA - UNNE

DIPLOMATURA EN CIENCIAS DE DATOS

Facultad de Ciencias Exactas y Naturales y Agrimensura
UNIVERSIDAD NACIONAL DEL NORDESTE

Clasificación de niveles de obesidad mediante modelos de aprendizaje automático: un enfoque predictivo basado en datos

Simón M. Olivieri[1], D. Leandro Urturi[2], Camila M. Muñoz[3]
1. AB-InBev Corrientes.
2. Ingeniero Civil - Profesional Independiente.
3. Facultad de Ciencias Aplicadas a la Industria - Universidad Nacional de Cuyo.

Resumen

La obesidad está vinculada de manera directa a diversas enfermedades, entre las que se encuentran dificultades respiratorias, diabetes, la reproducción, afecciones cardíacas, aumenta el riesgo de que aparezcan ciertos tipos de cáncer, colesterol alto y osteoartritis. Este problema constituye una pandemia global. Es vital reflexionar sobre la importancia de implementar medidas preventivas y de promoción para mejorar la salud y el bienestar de las personas. Para apoyar estas iniciativas, es fundamental investigar y aplicar nuevas soluciones impulsadas por las tecnologías actuales. El uso de técnicas de Aprendizaje Automático para evaluar los hábitos alimenticios y la actividad física permitiría clasificar el nivel de salud y bienestar de los individuos basándose en datos históricos. Este trabajo tiene por objetivo desarrollar y comparar modelos de clasificación del nivel de obesidad basados en el Aprendizaje Automático utilizando variables características de la enfermedad. La base de datos utilizada para este trabajo es "*Estimation of Obesity Levels Based On Eating Habits and Physical Condition*", la cual se encuentra disponible públicamente en UCI Machine Learning Repository. El conjunto de datos está conformado por 2111 registros que responden a 17 variables. Finalizada la etapa de preprocesamiento, se implementaron y evaluaron dos modelos de clasificación: Random Forest y Redes Neuronales Simples. Los resultados indicaron que el modelo basado en redes neuronales (Modelo 5) fue el más eficaz, logrando una precisión del 97.1% en la clasificación de los niveles de obesidad. Como conclusión, este trabajo puede resultar útil para la detección e intervención temprana de la enfermedad, con el propósito de tomar decisiones que minimicen su impacto en el bienestar de las personas.

Introducción

La Organización Mundial de la Salud (OMS o en sus siglas en inglés WHO) define a la obesidad como una acumulación excesiva de grasa que puede ser perjudicial para la salud (World Health Organization, 2024). La obesidad está vinculada de manera directa a diversas enfermedades, entre las que se encuentran dificultades respiratorias, diabetes, la reproducción, afecciones cardíacas, aumenta el riesgo de que aparezcan ciertos tipos de cáncer, colesterol alto y osteoartritis. La obesidad ha sido reconocida como una enfermedad crónica y recurrente, con una etiología multifactorial (Kaufer-Horwitz & Hernández, 2021). Es el resultado de una compleja interacción entre factores genéticos y ambientales, donde los cambios en la dieta (mayor consumo de alimentos ricos en azúcares y alto contenido de grasas) y el estilo de vida asociados con la urbanización (sedentarismo) han activado genes que predisponen a la obesidad (epigenética). Esto ha

[1] E-mail: simon.olivieri01@gmail.com
[2] E-mail: hannnnsen@gmail.com
[3] E-mail: camilam.m08@gmail.com

alterado los patrones de salud y enfermedad en las poblaciones, aumentando la morbilidad y mortalidad, además de generar problemas en diversas áreas de la vida de quienes la padecen.

Este problema constituye una pandemia global (Aguilera et al., 2019). De acuerdo a la OMS (2024), la prevalencia de la obesidad en todo el mundo aumentó en más del 100% entre 1990 y 2022. En 2022, aproximadamente, 2500 millones de adultos mayores a 18 años tenían sobrepeso, de los cuales más de 890 millones eran obesos. Esto representa que el 43% de los adultos (43% de los hombres y 44% de las mujeres) tenían sobrepeso, un incremento notable en comparación con 1990, cuando el porcentaje de adultos con sobrepeso era del 25%. En Argentina, más del 50% tiene exceso de peso (Ministerio de Salud, 2024).

Se utilizan diversos métodos para el diagnóstico de sobrepeso y obesidad, el más utilizado es el Índice de Masa Corporal (IMC), marcador indirecto de la grasa. Para su cálculo se tiene en cuenta el peso y la estatura de la persona: peso (kg)/estatura^2 (m^2). También existen otros métodos como el perímetro de la cintura, el grosor de los pliegues cutáneos y la bioimpedancia. Aunque el IMC es ampliamente aceptado, es importante destacar que las diferencias en la cantidad de grasa corporal y las proporciones musculares limitan su eficacia como medida universal (Mariana et al., 2017). Sin embargo, en la mayoría de los casos, es un buen indicador de un peso saludable. En el caso de personas adultas, la OMS (2024) define al sobrepeso con un IMC igual o superior a 25 y a la obesidad con un IMC igual o superior a 30.

Desde el Ministerio de Salud (2024) de la República Argentina se propone una alimentación saludable y actividad física como medidas efectivas para prevenir y controlar la obesidad. En su sitio web sugiere algunas de las siguientes recomendaciones: *realizar 4 comidas al día; aumentar el consumo de frutas, verduras y de pescado; disminuir el consumo de alimentos ultraprocesados con mucha azúcar, grasa y/o sal; y realizar al menos 150 minutos de actividad física a la semana a intensidad moderada, sumando como mínimo bloques de 10 minutos*.

Los modelos de predicción y clasificación basados en Aprendizaje Automático han demostrado recientemente ser una herramienta importante e innovadora en una gran variedad de aplicaciones biomédicas. En Malakouti et al. (2024), los autores aplicaron técnicas de Aprendizaje Automático para clasificar tumores cerebrales, obteniendo una precisión del 100% para personas enfermas y del 95,7% para personas sanas. En el trabajo de Oh et al. (2024), se evaluó la predicción de mortalidad en neonatos aplicando distintos modelos de Aprendizaje Automático basados en árboles: Random Forest, Gradient Boosting y LightGBM. En otro estudio realizado por Weatherall et al. (2024), se llevó a cabo una revisión sistemática en dieciocho estudios que informaron una variedad de modelos de Aprendizaje Automático para clasificar y predecir con éxito úlceras del pie diabético. Los autores concluyen que los modelos propuestos demuestran el potencial para mejorar el manejo clínico de estos pacientes.

El problema que representa la obesidad en la sociedad actual nos lleva a reflexionar sobre la importancia de implementar medidas preventivas y de promoción para mejorar la salud y el bienestar de las personas. Para apoyar estas iniciativas, es fundamental investigar y aplicar nuevas soluciones impulsadas por las tecnologías actuales. El uso de técnicas de Aprendizaje Automático para evaluar los hábitos alimenticios y la actividad física permitiría

clasificar el nivel de salud y bienestar de los individuos basándose en estas variables. Esto facilitaría la modificación de tendencias negativas, como la desnutrición, el sobrepeso o la obesidad, al permitir la toma de medidas correctivas a tiempo. Este trabajo tiene por objetivos contribuir en: desarrollar modelos de clasificación del nivel de obesidad basados en el Aprendizaje Automático utilizando variables características; y comparar el rendimiento de los distintos modelos propuestos para esta clasificación.

Metodología

Etapa 1: **Recopilación de datos** *(08/10/2024)*

La base de datos utilizada para este trabajo es "*Estimation of Obesity Levels Based On Eating Habits and Physical Condition*", la cual se encuentra disponible públicamente en UCI Machine Learning Repository (2019) en formato **.CSV*. El dataset cuenta con 2111 registros y 17 variables. Sus autores aclaran que el 77% de los datos se generó sintéticamente utilizando la herramienta Weka y el filtro SMOTE, mientras que el 23% (485 registros) de los datos restantes se recopilaron directamente de los usuarios a través de una plataforma web con una encuesta en línea disponible durante 30 días (Palechor & De la Hoz Manotas, 2019). Los participantes eran de Perú, México y Colombia, con edades entre 14 y 61 años; los datos comprenden una diversidad de variables que van desde la ingesta de alimentos, la condición física, el estilo de vida y los hábitos alimenticios, entre otros (Tabla 1). El etiquetado del Nivel de Obesidad se realizó calculando el IMC para cada individuo (Palechor & De la Hoz Manotas, 2019), dividiendo los datos en seis niveles: peso insuficiente, peso normal, sobrepeso nivel I, sobrepeso nivel II, obesidad nivel I, obesidad nivel II y obesidad nivel III.

Tabla 1. Variables del dataset.

Pregunta	**Posible respuesta**	**Tipo de variable**
Gender *¿Cuál es tu género?*	Mujer Hombre	Categórica, binaria.
Age *¿Cuál es tu edad?*	Valor numérico	Cuantitativa, discreta.
Height *¿Cuál es tu altura?*	Valor numérico en metros	Cuantitativa, continua.
Weight *¿Cuál es tu peso?*	Valor numérico en kilogramos	Cuantitativa, continua.
FHWO *¿Ha sufrido o padece sobrepeso un familiar?*	Si No	Categórica, binaria.
FAVC *¿Comes comida alta en calorías con frecuencia?*	Si No	Categórica, binaria.
FCVC *¿Suele comer verduras en sus comidas*	Nunca A veces Siempre	Categórica, ordinal.
NCP *¿Cuántas comidas principales tienes a diario?*	Entre 1 y 2 Tres Más de tres	Categórica, ordinal.
CAEC *¿Comes algún alimento entre comidas*?	No A veces	Categórica, ordinal.

Pregunta	**Posible respuesta**	**Tipo de variable**
	Frecuentemente Siempre	
SMOKE *¿Fumas?*	Sí No	Categórica, binaria.
CH2O *¿Cuánta agua bebes a diario?*	Menos de un litro Entre 1L y 2L Más de 2L	Categórica, ordinal.
SCC *¿Usted controla las calorías que come diariamente?*	Sí No	Categórica, binaria.
FAF *¿Con qué frecuencia realiza actividad física?*	No tengo 1 o 2 días 2 o 4 días 4 o 5 días	Categórica, ordinal.
TUE *¿Cuánto tiempo utiliza dispositivos tecnológicos como teléfono celular, videojuegos, televisión, ordenador y otros?*	0 - 2 horas 3 - 5 horas Más de 5 horas	Categórica, ordinal.
CALC *¿Con qué frecuencia toma alcohol?*	No bebo A veces Frecuentemente Siempre	Categórica, ordinal.
MTRANS *¿Qué medio de transporte utiliza habitualmente?*	Automóvil Moto Bicicleta Transporte público Caminando	Categórica, nominal.
NOBESITY *Nivel de obesidad*	Peso insuficiente Peso normal Sobrepeso Nivel I Sobrepeso Nivel II Obesidad Nivel I Obesidad Nivel II Obesidad Nivel III	Categórica, ordinal.

Etapa 2: **Preprocesamiento de datos** *(08/10/2024)*

En la Tabla 1 se detallan las diecisiete variables recopiladas por Palechor & De la Hoz Manotas (2019). Como variable dependiente se utilizó "NObesity", mientras que el restante de variables se usó como variables independientes del modelo. Al observar la Tabla 1, las variables independientes son una combinación de variables categóricas (nominales, ordinales o binarias) y numéricas (discretas o continuas), donde 13 de 16 son categóricas; sin embargo, al evaluar el dataset encontramos que 8 de 16 se encontraban convertidas en forma numérica. En consecuencia, se utilizan los objetos *LabelEncoder* y *OneHotEncoder* de la librería *Scikit-learn* para transformar las restantes variables categóricas en representaciones numéricas. Para las variables independientes (*features*), se aplica *OneHotEncoder*, que convierte cada categoría en columnas binarias, evitando la colinealidad. Por otro lado, para la variable dependiente (*target*), se utiliza *LabelEncoder*, que asigna un número entero único a cada clase, lo que facilita la interpretación y mejora el proceso de entrenamiento en modelos de machine learning, incluidos aquellos basados

en Deep Learning. Este enfoque optimiza la representación de los datos y contribuye a un rendimiento más eficiente del modelo.

No se encontraron valores faltantes en el dataset; no obstante, sí observamos que algunas variables anteriormente categóricas y ahora numéricas, que debían contener valores discretos, contenían valores continuos. Para unificar el tipo de variable decidimos convertir estos valores continuos en discretos, esto fue posible a través de la función *round().astype()* de la librería *Pandas* y Python se redondearon los valores y así poder entrenar el algoritmo correctamente.

Por último, el dataset fue separado en conjuntos de datos de entrenamiento, validación y prueba. La partición de un conjunto de datos en diferentes subconjuntos es una práctica esencial en el desarrollo de modelos de machine learning. Esta técnica permite evaluar el rendimiento del modelo de manera efectiva y evitar problemas como el sobreajuste. A continuación, se describen los tres conjuntos que se crean y su propósito:

1. **Conjunto de Entrenamiento (*train_set*)**: Este subconjunto se utiliza para ajustar y entrenar el modelo. Contiene la mayor parte de los datos, en nuestro caso el 70% (1477 registros) y es donde el modelo aprende a realizar predicciones basadas en las características del conjunto.
2. **Conjunto de Validación (*val_set*)**: Este conjunto se utiliza para ajustar los hiperparámetros del modelo y tomar decisiones sobre el diseño del modelo. Al evaluar el rendimiento del modelo en datos no vistos durante el entrenamiento, se puede evitar el sobreajuste y seleccionar el modelo que generaliza mejor, en nuestro caso 15% (317 registros).
3. **Conjunto de Prueba (*test_set*)**: Este subconjunto se utiliza para evaluar el rendimiento final del modelo. Una vez que se ha completado el entrenamiento y la validación, el conjunto de prueba proporciona una estimación imparcial de cómo se comportará el modelo en datos completamente nuevos, en nuestro caso 15% (317 registros).

Se utilizan las librerías de *sklearn* y se realiza una función llamada *particionador* que toma el dataframe y devuelve 3 subconjuntos de datos *train_set – val_set – test_set*, utilizando la función *train_test_split.*

Etapa 3: **Selección de características más importantes** *(09/10/2024)*
La selección de características basada en modelos es una técnica fundamental en el preprocesamiento de datos que tiene como objetivo identificar y retener sólo las variables más relevantes para la predicción del objetivo. Al emplear un modelo como el Random Forest, es posible calcular de manera efectiva la importancia de cada característica, dado que este tipo de modelos incorpora mecanismos que evalúan su impacto en las predicciones.
Para obtener las características más importantes del conjunto de datos, se utilizó la librería de *Scikit-learn* con los siguientes componentes:

- **RandomForestClassifier**: Este modelo permite calcular la importancia de las características.
- **SelectFromModel**: Esta clase se utiliza para realizar la selección de características basada en la importancia determinada por el modelo.

Los hiperparámetros utilizados fueron:

- **threshold='median'**: Este umbral establece que sólo se seleccionarán las características cuya importancia es mayor que la mediana de todas las importancias calculadas por el modelo, lo que permite filtrar de manera efectiva las características menos relevantes.

Etapa 4: **Modelado** *(09/10/2024-14/10/2024)*

Random Forest

Para el modelo de Random Forest, se llevaron a cabo dos enfoques de optimización de hiperparámetros: uno mediante *RandomizedSearchCV* y otro usando *Optuna* (Tabla 2). En ambos casos, se utilizó el conjunto de datos completo, ya que ambos optimizadores buscan la mejor selección de parámetros y también iteran sobre los hiperparámetros clave como:

- *n_estimators*: número de árboles en el bosque.
- *max_depth*: profundidad máxima de los árboles.
- *bootstrap:* opción para utilizar el muestreo con reemplazo.
- *max_features:* número de características a considerar al buscar la mejor división.

Tabla 2. Descripción de los optimizadores.

Optimizador	Descripción
RandomizedSearchCV	Este enfoque permite buscar de manera aleatoria a través de un espacio definido de hiperparámetros, lo que puede ser más eficiente que una búsqueda exhaustiva. Se configuró un rango para los hiperparámetros. El modelo se entrenó utilizando el conjunto de entrenamiento y se evaluó utilizando las métricas Accuracy y F1-score.
Optuna	Esta herramienta proporciona una manera eficiente de realizar la optimización de hiperparámetros mediante la búsqueda de parámetros de manera más inteligente. En este caso, se definió una función objetivo que sugiere valores para *n_estimators*, *max_depth*, *bootstrap*, y *max_features*, entrenando el modelo con el conjunto completo de datos y devolviendo los valores de Accuracy y F1-score como resultado.

Modelos de Deep Learning

Para entrenar los modelos de Deep Learning, se realizó un escalado de los datos utilizando la librería *sklearn* y la clase *MinMaxScaler()*, lo que permitió transformar todos los grupos de datos de manera uniforme.

En este trabajo, se implementó un modelo de Deep Learning basado en una red neuronal simple para abordar un problema de clasificación. Las librerías utilizadas incluyeron *torch* para la construcción y entrenamiento de la red neuronal, y *sklearn.metrics* para la evaluación del rendimiento del modelo. En las Tablas 3 y 4 se encuentra la configuración de las redes. Durante el entrenamiento, se monitoreó el rendimiento en los conjuntos de entrenamiento y validación, calculando métricas como el *F1-score* y la exactitud (*accuracy*) en cada *epoch*. Finalmente, se evaluó el modelo con el conjunto de prueba (*test_set*), calculando nuevamente el *F1-score* y la exactitud para determinar el rendimiento del modelo en datos no vistos.

Tabla 3. Arquitectura de las redes neuronales.

Modelo Red neuronal	Número de capas ocultas	cantidad de nodos de capas ocultas	Ecuación de Agregación	Ecuación de activación capas de entrada y ocultas	Ecuación de activación capa de salida	DropOut por capa
Modelo 1	2	64	Lineal	Relu	Softmax	No tiene
Modelo 2	2	64	Lineal	Relu	Softmax	0.2
Modelo 3	2	64	Lineal	Tanh	Softmax	0.2
Modelo 4	2	64	Lineal	Tanh	Softmax	0.2
Modelo 5	1	64	Lineal	Tanh	Softmax	No tiene

Tabla 4. Funciones de coste, optimizadores e hiperparámetros de las redes neuronales.

Modelo Red neuronal	Función de coste	Optimizador				
		Optimizador	learning rate	beta	Regularización L2	amsgrad
Modelo 1	CrossEntropyLoss	Adam	0.001	Valor por defecto: (0.9, 0.999)	0	False
Modelo 2	CrossEntropyLoss	Adam	0.0005	Valor por defecto: (0.9, 0.999)	0	False
Modelo 3	CrossEntropyLoss	Adam	0.001	Valor por defecto: (0.9, 0.999)	0	False
Modelo 4	CrossEntropyLoss	Adam	0.0005	Valor por defecto: (0.9, 0.999)	0	False
Modelo 5	CrossEntropyLoss	Adam	0.0005	Valor por defecto: (0.9, 0.999)	0	False

Etapa 5: **Evaluación del modelo** *(14/10/2024)*

Para evaluar los diferentes modelos a fin de determinar cuál se comporta mejor tanto en el conjunto de validación como en el conjunto de prueba se evaluaron las siguientes métricas:

Acurracy

Se define como el porcentaje de la muestra que se clasificó correctamente. Se puede medir con la siguiente ecuación (1):

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

Recall

Es la tasa de verdaderos positivos o la proporción de todos los positivos reales que se clasificaron correctamente como positivos. Se puede definir matemáticamente como (2):

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

Precision

Es la proporción de todas las clasificaciones positivas del modelo que son realmente positivas. Se calcula de la siguiente manera (3):

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

F1-Score

Se describe como la media armónica de precision y recall.

Su rango es de [0,1]. Esta métrica nos indica qué tan preciso (clasifica correctamente cuántas instancias) y cuán robusto (no pierde ningún número significativo de instancias) es nuestro clasificador. Se puede medir de la siguiente manera (4):

$$F1 - score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \quad (4)$$

Resultados

A partir del análisis exploratorio de los datos, podemos observar el tipo de variables que contiene nuestro dataset de estudio (Fig. 1), de esta forma se verifica lo mencionado en la Etapa 2 sobre las problemáticas que tuvimos con las variables.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2111 entries, 0 to 2110
Data columns (total 17 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   Gender                          2111 non-null   object
 1   Age                             2111 non-null   float64
 2   Height                          2111 non-null   float64
 3   Weight                          2111 non-null   float64
 4   family_history_with_overweight  2111 non-null   object
 5   FAVC                            2111 non-null   object
 6   FCVC                            2111 non-null   float64
 7   NCP                             2111 non-null   float64
 8   CAEC                            2111 non-null   object
 9   SMOKE                           2111 non-null   object
 10  CH2O                            2111 non-null   float64
 11  SCC                             2111 non-null   object
 12  FAF                             2111 non-null   float64
 13  TUE                             2111 non-null   float64
 14  CALC                            2111 non-null   object
 15  MTRANS                          2111 non-null   object
 16  NObeyesdad                      2111 non-null   object
dtypes: float64(8), object(9)
memory usage: 280.5+ KB
```

Figura 1. Tipo de variables.

Proseguimos con la visualización de nuestros datos, para ellos realizamos gráficos de frecuencia para las variables categóricas. En este análisis observamos que dos de las variables categóricas (MTRANS y CALC) presentaban pocos registros en sus niveles (Fig. 2). En consecuencia, decidimos realizar un agrupamiento entre niveles próximos o afines. Para la variable CALC se unieron los registros de "always" y "Frequently", y para la variable MTRANS: "Bike" con "Walking" y "Motorbike" con "Automobile" (Fig. 3).

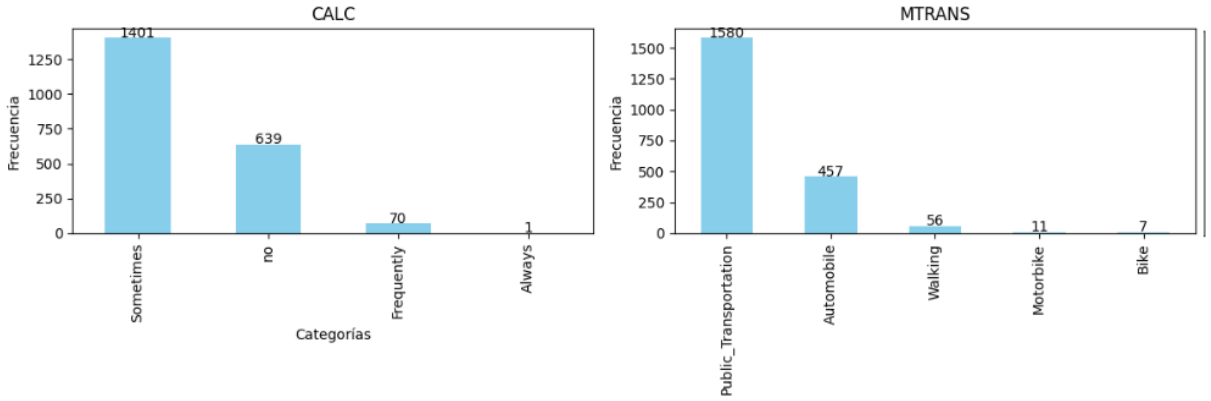

Figura 2. Variables MTRANS y CALC antes del preprocesamiento.

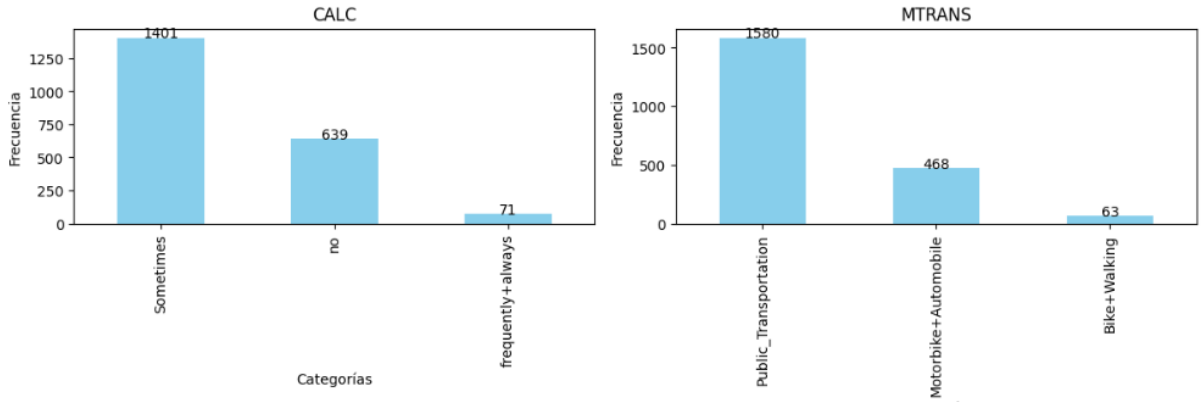

Figura 3. Variables MTRANS y CALC después del preprocesamiento.

Para las variables numéricas (Age, Weight y Height) realizamos histogramas (Fig. 4). Se observa que las variables presentan una distribución normal, aunque se contempla cierta asimetría hacia la derecha (Age y Weight).

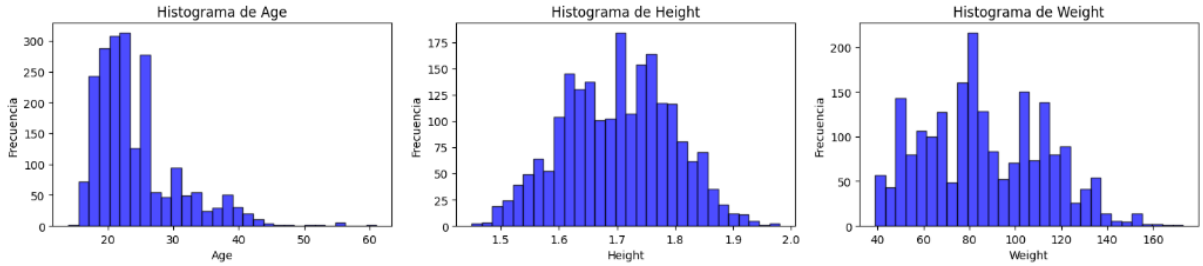

Figura 4. Histograma de las variables numéricas.

Continuamos analizando las variables dependientes con nuestra variable independiente de interés (Fig. 5). En este gráfico se muestra la relación que existe entre el historial familiar de obesidad y el nivel de obesidad. En los niveles más altos de obesidad el historial familiar se encuentra muy presente.

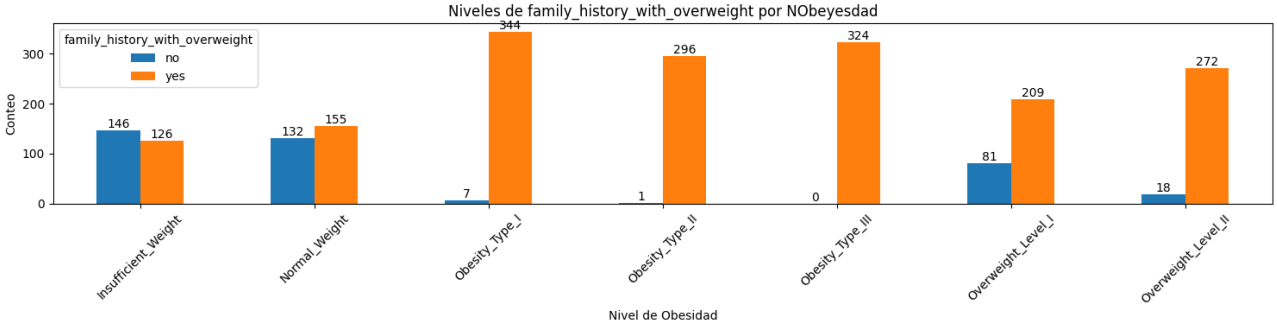

Figura 5. Relación entre la variable de historial familiar de obesidad y los niveles de obesidad.

Para las variables dependientes numéricas realizamos gráficos de dispersión. Entre las variables Height y Weight se visualiza una relación positiva (Fig. 6), con una clara estratificación de los niveles de obesidad.

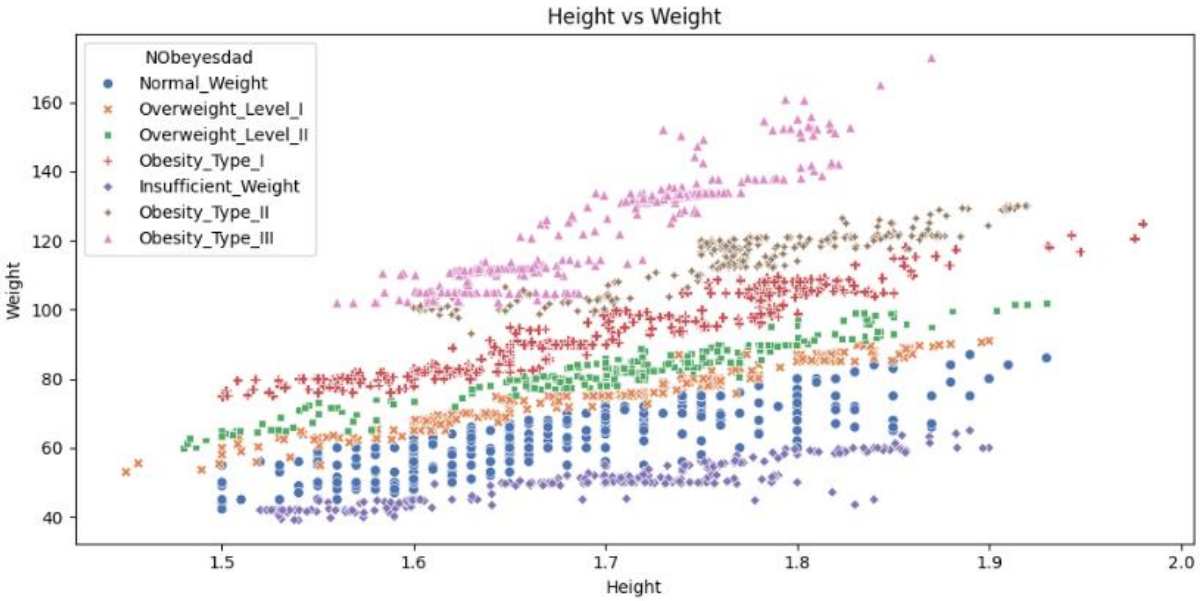

Figura 6. Relación entre las variables height y weight.

Como indicamos en la Etapa 2, transformamos las variables categóricas restantes en variables numéricas a fin de homogeneizar el dataset para llevar a cabo nuestros modelos. En la Fig. 7 se muestra la matriz de correlación de la totalidad de las variables del dataset de estudio. Destacamos que Weight es la más variable influyente con todos los niveles de la variable NObesity presentaciones correlaciones moderadas entre los niveles: *weight-obesity type II*: 0.44; *weight-obesity_type_III:* 0.56; y *weight-normal weight:* -0.37. También presenta una correlación moderada entre los niveles *gender_male* y o*besity_type_II* con un valor de 0.39. Y una correlación negativa moderada entre *gender_male* y *obesity_type_III* de -0.43.

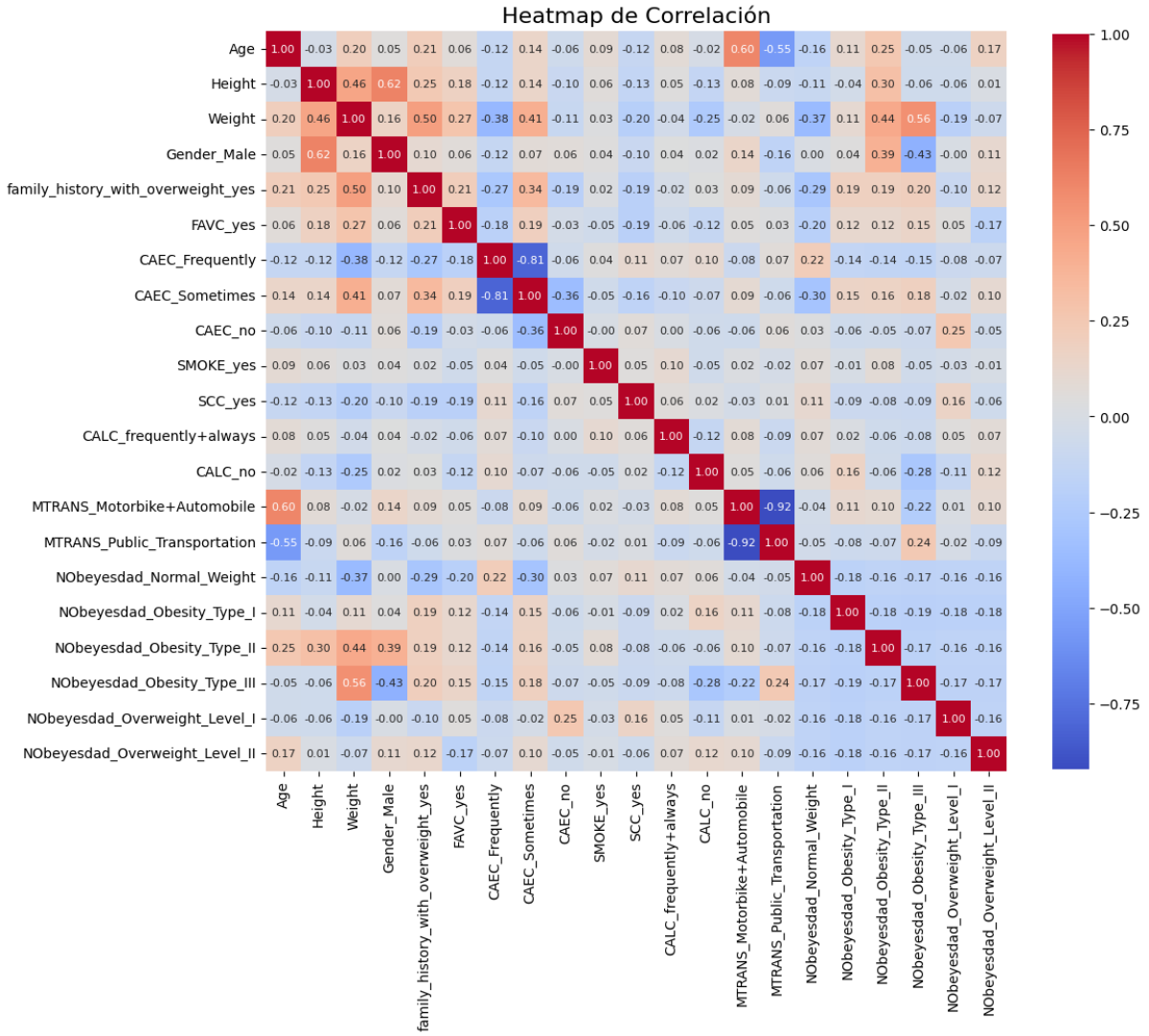

Figura 7. Matriz de correlación de las variables del dataset.

En la Tabla 5 se muestra el resultado de la selección de características obtenidas con *SelectFromModel, Optuna y RandomizedSearchCV* que se han identificado como las variables más importantes para la predicción.

Tabla 5. Selección de características mediante modelos.

Optimización de Hiperparámetros	**max_features**							
	1	2	3	4	5	6	7	8
RandomizedSearchCV	Weight	Height	Age	Gender_Male	family_history_with_overweight_yes	CALC_no	FAVC_yes	
Optuna	Weight	Height	Age	Gender_Male	family_history_with_overweight_yes	CALC_no	FAVC_yes	
Select model y RandomizedSearchCV	Weight	Height	Age	Gender_Male	family_history_with_overweight_yes	CALC_no	FAVC_yes	CAEC_Sometimes

En la etapa 4, se ejecutaron los modelos de Random Forest con los optimizadores *RandomizedSearchCV* y *Optuna* para la clasificación de las variables. El rendimiento de

estos modelos se determinó a partir de las métricas *Acurracy* y *F1-score*. El modelo que presentó mejor rendimiento fue el que utilizó a *Optuna* como optimizador (Tabla 6).

Tabla 6. Modelo de aprendizaje: RandomForestClassifier.

Optimización de Hiperparámetros	Mejores hiperparametros				Muestra de validación		Muestra de testeo	
	n_estimators	max_depth	bootstrap	max_features	Acurracy	F1-score	Acurracy	F1-score
RandomizedSearchCV	120	58	False	7	0.952	0.952	0.927	0.927
Optuna	203	84	False	7	0.952	0.95	0.946	0.946

En Fig. 8 y Fig. 9, graficamos las matrices de confusión para el conjunto de validación utilizando los hiperparámetros obtenidos por los dos optimizadores. En la diagonal de la matriz se encuentran los valores que fueron clasificados correctamente; en la parte superior de la matriz se encuentran los valores clasificados como falsos negativos; y en la parte inferior de la diagonal, los falsos positivos. También podemos observar que en *obesity_type_I* en el modelo con *Optuna* presenta 46 verdaderos positivos contra los 44 que se obtienen con el optimizador *RandomizedSearchCV*. Además, podemos ver que en *obesity_type_I* predice un falso negativo al categorizarlo con *overweight_level_I*, siendo este el único falso negativo en el modelo optimizado por *Optuna* para este nivel. Si lo comparamos con el modelo *RandomizedSearchCV*, en este último se presentan 2 falsos negativos. En cambio, la cantidad de los falsos positivos en ambos modelos coinciden.

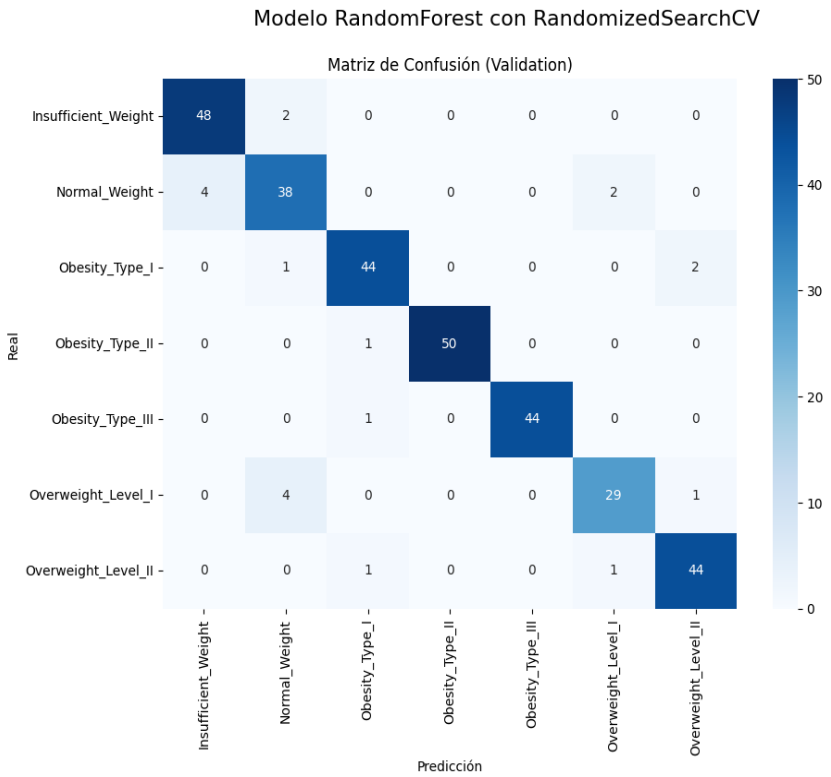

Figura 8. Matriz de Confusión del conjunto de validación en el modelo optimizado por RandomizedSearchCV.

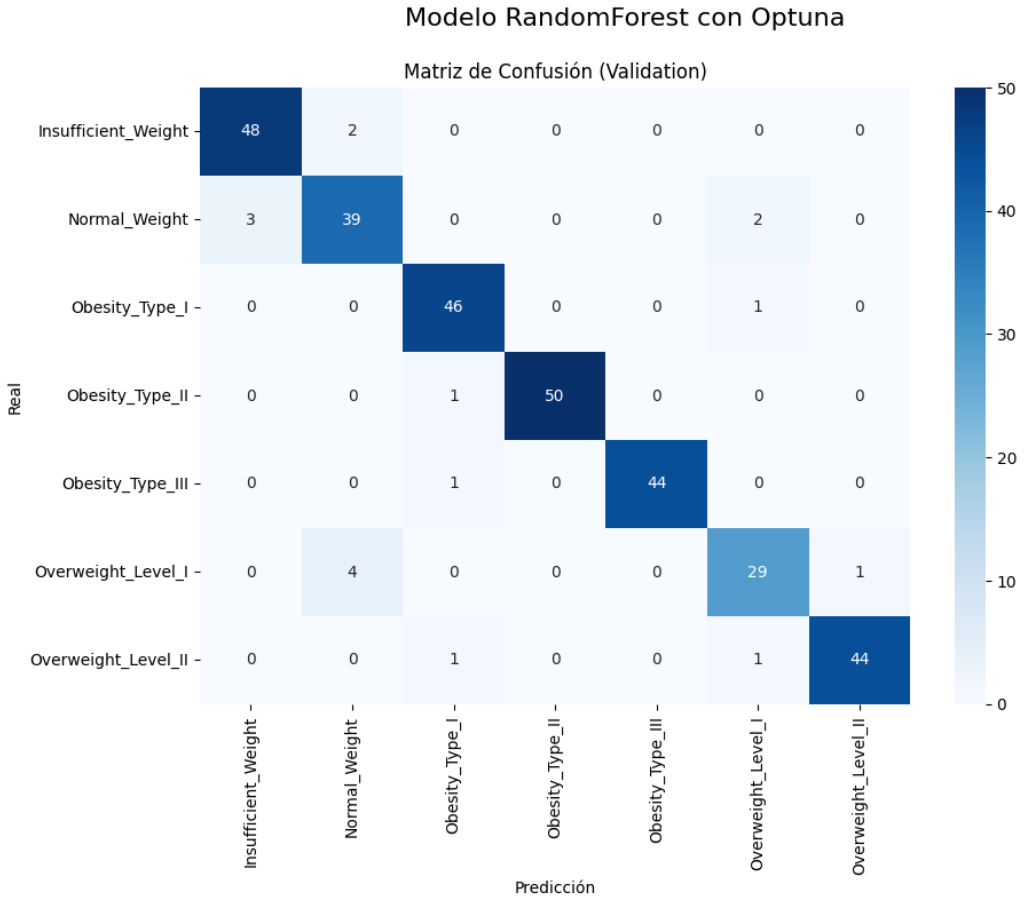

Figura 9. Matriz de Confusión del conjunto de validación en el modelo optimizado por Optuna.

Los resultados que obtuvimos en los modelos Deep Learning se presentan en la Tabla 7, de acuerdo con estos resultados el Modelo 5 se destaca como el mejor entre los modelos evaluados.

Tabla 7. Entrenamiento y resultados de los modelos de redes neuronales.

Modelo Red neuronal	**Optimizador**		**Muestra de validación**			**Muestra de testeo**	
	mini batch	**epochs**	**Acurracy**	**F1-score**	**Error**	**Acurracy**	**F1-score**
Modelo 1	32	1250	0.959	0.958	0.30	0.981	0.981
Modelo 2	32	1250	0.962	0.961	0.30	0.974	0.974
Modelo 3	32	1250	0.962	0.962	0.12	0.981	0.981
Modelo 4	64	1250	0.962	0.962	0.09	0.977	0.978
Modelo 5	32	1250	0.971	0.971	0.12	0.971	0.971

En el Modelo 5, para la muestra de validación las métricas de *Accuracy* y *F1-score* son de 0.971, los valores más altos entre todos los modelos. Aunque el error es de 0.12, ligeramente superior que el del Modelo 4, sigue siendo competitivo. Para la muestra de testeo, los valores de *Accuracy* y *F1-score* (0.971 respectivamente) no son los más altos; sin embargo, reflejan una consistencia sólida entre las métricas de validación y testeo.

Si bien los Modelos 1 y 3 presentan una mayor *Accuracy* en la muestra de testeo, el Modelo 5 muestra un rendimiento más equilibrado entre las métricas de validación y testeo. Por lo tanto, su consistencia lo posiciona como el modelo más robusto para este análisis.

Al comparar los gráficos de los modelos 1 y 5 (Fig. 10), se observa que el Modelo 5 ofrece un rendimiento superior. El gráfico de pérdida del Modelo 5 muestra una disminución constante tanto en los datos de entrenamiento como en los de validación, lo que indica un ajuste más adecuado y una menor diferencia entre ambas curvas, sugiriendo una menor propensión al sobreajuste. En contraste, el Modelo 1 presenta un aumento en la pérdida de validación a partir de las 300 *epochs*, señal de sobreajuste. Además, el *F1-score* del Modelo 5 es más estable y consistente entre las muestras de validación y prueba, manteniéndose muy cercano al de entrenamiento, lo que refleja una mejor capacidad de generalización. Por el contrario, el *F1-score* del Modelo 1 muestra mayor variabilidad, lo que compromete su capacidad de generalizar.

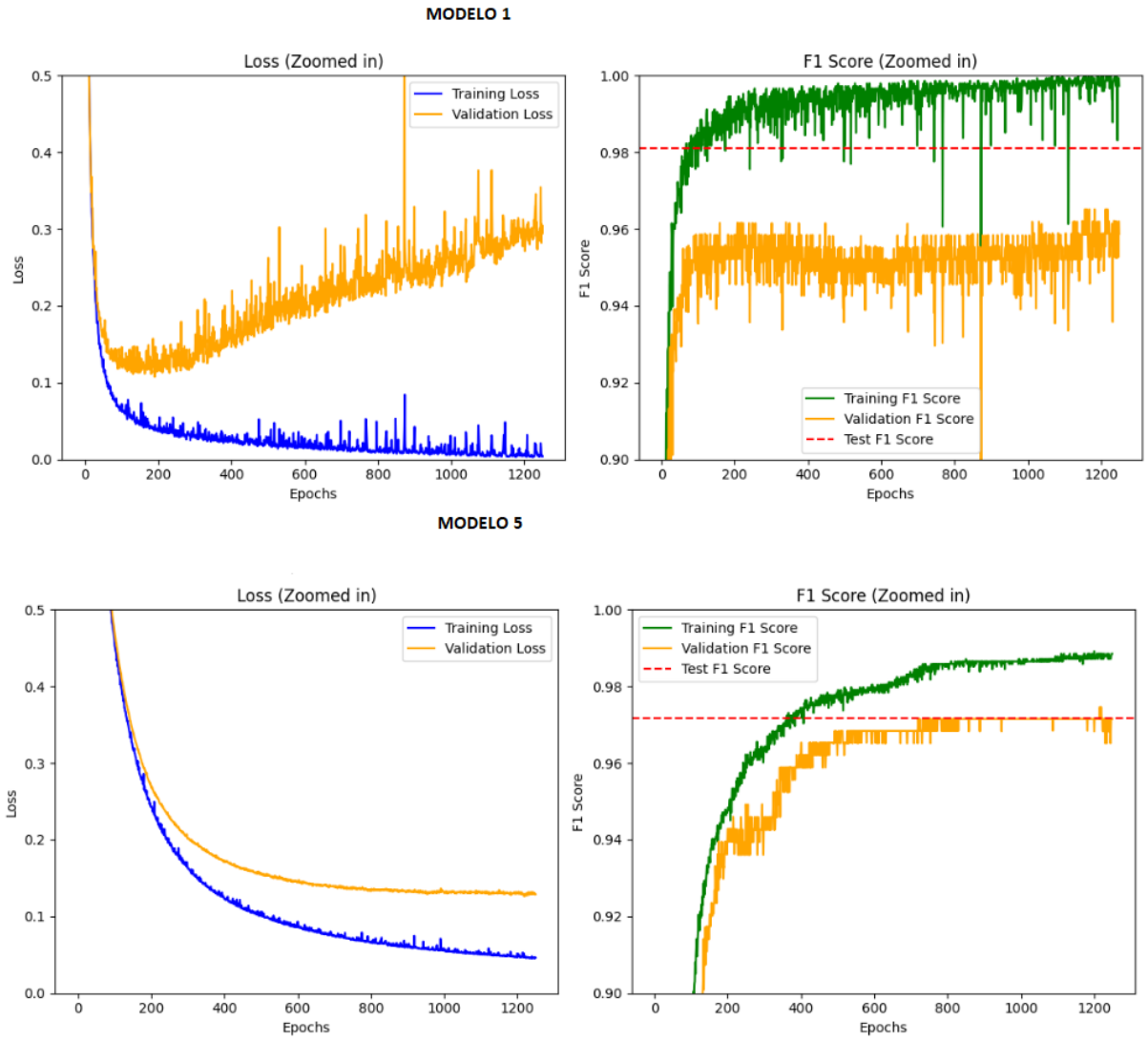

Figura 10. Comparación de gráficos de función de pérdida y F1-score de los Modelos 1 y 5.

Conclusiones

La obesidad es una enfermedad que puede tener graves consecuencias en la salud de las personas. Es importante controlar el estilo de vida y los hábitos alimenticios de la población que la padece. Así mismo, es fundamental que los gobiernos tomen medidas preventivas y en la promoción de hábitos saludables de alimentación, reeducando a la sociedad para tener una mejor calidad de vida y bienestar. Este trabajo propone 2 tipos de modelos de clasificación: Random Forest y Deep Learning (Redes Neuronales Simples), para determinar los niveles de obesidad. El modelo 5 de Redes Neuronales obtuvo el mejor rendimiento en comparación a los otros modelos. Este método permite clasificar los diferentes niveles de obesidad con una precisión del 97.1%.

Los resultados muestran que el Modelo 5 fue el más efectivo en la muestra de validación, superando en rendimiento a los otros modelos. Este modelo mostró una notable consistencia entre las métricas de validación y testeo, lo que indica una alta capacidad de generalización. Este trabajo puede resultar útil para la detección e intervención temprana de la enfermedad, con el propósito de tomar decisiones que minimicen su impacto en el bienestar de las personas o de la sociedad.

El dataset utilizado presenta algunas limitaciones. La cantidad de registros es relativamente pequeña, sería beneficioso para próximas investigaciones ampliar el número de registros. Como se dijo al comienzo de este trabajo, el IMC no es una medida efectiva para determinar el nivel de obesidad en las personas, ya que quienes se encuentren realizando musculación (como fisicoculturistas) podrían obtener un IMC alto, sin embargo, si se realizaran mediciones antropométricas como el grosor del tejido adiposo subcutáneo, podría no coincidir con el IMC. Se sugiere incluir otras variables como: perímetro de cintura, grosor de los pliegues cutáneos, cantidad de calorías que consume a diario y tipo de actividad física, para obtener una caracterización más precisa de las personas.

Bibliografía

World Health Organization: WHO. (2024). *Obesidad y sobrepeso*. https://www.who.int/es/news-room/fact-sheets/detail/obesity-and-overweight

Aguilera, C., Labbé, T., Busquets, J., Venegas, P., Neira, C., & Valenzuela, Á. (2019). Obesidad: ¿Factor de riesgo o enfermedad? *Revista Médica de Chile*, *147*(4), 470-474.

Ministerio de Salud (2024). *Sobrepeso y obesidad*. Argentina.gob.ar. https://www.argentina.gob.ar/salud/alimentacion-saludable/obesidad

Mariana, O. G., Barahona, A., & Raquel, S. L. (2017). *Índice de masa corporal y porcentaje de grasa en adultos indígenas ecuatorianos Awá*. https://ve.scielo.org/scielo.php?script=sci_arttext&pid=S0004-06222017000100006

Kaufer-Horwitz, M., & Hernández, J. F. P. (2021). La obesidad: aspectos fisiopatológicos y clínicos. *INTERdisciplina*, *10*(26), 147. https://doi.org/10.22201/ceiich.24485705e.2022.26.80973

UCI Machine Learning Repository. (2019). https://archive.ics.uci.edu/dataset/544/estimation+of+obesity+levels+based+on+eating+habits+and+physical+condition

Palechor, F. M., & De la Hoz Manotas, A. (2019). Dataset for estimation of obesity levels based on eating habits and physical condition in individuals from Colombia, Peru and Mexico. *Data In Brief*, *25*, 104344. https://doi.org/10.1016/j.dib.2019.104344

Malakouti, S. M., Menhaj, M. B., & Suratgar, A. A. (2024). Machine learning and transfer learning techniques for accurate brain tumor classification. *Clinical eHealth*, *7*, 106-119. https://doi.org/10.1016/j.ceh.2024.08.001

Oh, M., Kim, S., Kim, M., Seo, Y. M., & Yum, S. K. (2024). Machine-learning-based evaluation of the usefulness of lactate for predicting neonatal mortality in preterm infants. *Pediatrics & Neonatology*. https://doi.org/10.1016/j.pedneo.2024.09.003

Weatherall, T., Avsar, P., Nugent, L., Moore, Z., McDermott, J. H., Sreenan, S., Wilson, H., McEvoy, N. L., Derwin, R., Chadwick, P., & Patton, D. (2024). The impact of machine learning on the prediction of diabetic foot ulcers – A systematic review. *Journal Of Tissue Viability*. https://doi.org/10.1016/j.jtv.2024.07.004