UNIVERSIDAD PROVINCIAL DEL SUDOESTE

FACULTAD DE LA MICRO, PEQUEÑA

Y MEDIANA EMPRESA

CARRERA DE TECNICATURA UNIVERSITARIA EN TECNOLOGÍAS

DE PROGRAMACIÓN SEDE PUNTA ALTA

ELEMENTOS DE APRENDIZAJE DE MÁQUINA YBIG DATA

TRABAJO FINAL

ANÁLISIS DE FACTORES DE ESTILO DE VIDA Y OBESIDAD: UN ENFOQUE DE APRENDIZAJE AUTOMÁTICO

Autores:

Altamiranda Cristian Daniel

Aramayo María Soledad

Carballo Daniel Alberto

Docente:

Valentín Barco

Punta Alta – Argentina

2023

CONTENIDO

1. TÍTULO

ANÁLISIS DE FACTORES DE ESTILO DE VIDA Y OBESIDAD: UN ENFOQUE DE APRENDIZAJE AUTOMÁTICO

2. INTRODUCCIÓN

2.1. Contexto del Problema

El conjunto de datos proporcionado se centra en evaluar diversos factores relacionados con la obesidad. La obesidad es un problema de salud global que afecta a un número significativo de personas en todo el mundo. Entender los factores que contribuyen a la obesidad y desarrollar modelos predictivos puede ser crucial para abordar y prevenir este problema de salud.

Los datos consisten en la estimación de los niveles de obesidad en personas de los países de México, Perú y Colombia, con edades entre 14 y 61 años y diversos hábitos alimentarios y condición física, los datos fueron recolectados mediante una plataforma web con una encuesta donde usuarios anónimos respondieron cada uno pregunta, luego se procesó la información obteniendo 17 atributos y 2111 registros.

A continuación se detalla las variables o atributos que contiene el dataset que se utilizó en el proyecto:

Los atributos relacionados con los hábitos alimentarios son:

- Consumo frecuente de alimentos ricos en calorías (FAVC)
- Frecuencia de consumo de vegetales (FCVC)
- Número de comidas principales (NCP)
- Consumo de alimentos entre comidas (CAEC)
- Consumo de agua diario (CH20)
- Consumo de alcohol (CALC)
- Fuma(SMOKE)

Los atributos relacionados con la condición física son:

- Monitoreo del consumo de calorías (SCC)
- Frecuencia de actividad física (FAF)
- Tiempo de uso de dispositivos tecnológicos (TUE)
- Transporte utilizado (MTRANS)

Variables Demográficas:

- Género
- Edad

Variables Antropométricas:

- Altura
- Peso

Historia Familiar:

- Antecedentes familiares

El dataset contiene variables numéricas y categóricas.

A continuación se detalla los nombres de las variables utilizadas en este proyecto y sus valores:

- 'Gender': 'Genero',
- 'Age': 'Edad',
- 'Height': 'Altura',
- 'Weight': 'Peso',
- 'family_history_with_overweight': 'Antecedentes_familiares',
- 'FAVC': 'Consumo_Calorico',
- 'FCVC': 'Consumo_Vegetal',
- 'NCP': 'Comidas_Principales',
- 'CAEC': 'Consumo_Entre_Comidas',
- 'SMOKE': 'Fuma',
- 'CH2O': 'Consumo_Agua',

- 'SCC': 'Monitoreo_Calorico',
- 'FAF': 'Actividad_Fisica',
- 'TUE': 'Uso_Tecnologia',
- 'CALC': 'Consumo_Alcohol',
- 'MTRANS': 'Medio_Transporte',
- 'NObeyesdad': 'Nivel_Obesidad_Original'

Valores de Nivel_Obesidad_Original: 'Insufficient_Weight', 'Normal_Weight', 'Overweight_Level_I', 'Overweight_Level_II', 'Obesity_Type_I', 'Obesity_Type_II', 'Obesity_Type_III'
Valores para Genero: 'Female', 'Male'
Valores para Edad: de 14 a 61 años
Valores para Altura: de 1.45 a 1.98 mts.
Valores para Peso: de 39.0 a 173.0 kgs.
Valores para Antecedentes_familiares: 'yes', 'no'
Valores para Consumo_Calorico: 'no', 'yes'
Valores para Consumo_Vegetal: de 1.00 a 3.00
Valores para Comidas_Principales: de 1.00 a 4.00
Valores para Consumo_Entre_Comidas: 'no', 'Sometimes', 'Frequently', 'Always'
Valores para Fuma: 'no', 'yes'
Valores para Consumo_Agua: de 1.00 a 3.00
Valores para Monitoreo_Calorico: 'no', 'yes'
Valores para Actividad_Fisica: de 0.00 a 3.00
Valores para Uso_Tecnologia: de 0.00 a 2.00
Valores para Consumo_Alcohol: 'no' 'Sometimes' 'Frequently' 'Always'
Valores para Medio_Transporte: 'Walking', 'Bike', 'Public_Transportation', 'Motorbike', 'Automobile'

2.2. Importancia y Relevancia

La obesidad está vinculada a numerosos problemas de salud, como enfermedades cardiovasculares, diabetes tipo 2 y otros trastornos metabólicos. Abordar la obesidad es esencial para mejorar la calidad de vida y reducir la carga económica asociada a las enfermedades relacionadas con la obesidad en sistemas de salud. Comprender los

factores que contribuyen a la obesidad y poder predecir su aparición puede ser valioso para implementar intervenciones preventivas y programas de salud personalizados.

El conjunto de datos proporciona información sobre diversas variables relacionadas con la salud y el estilo de vida de individuos, como género, edad, altura, peso, antecedentes familiares, consumo de alimentos calóricos, frecuencia de consumo de vegetales, hábitos alimenticios, consumo de agua, hábito de fumar, monitoreo calórico, actividad física, uso de tecnología, consumo de alcohol, medio de transporte y nivel de obesidad.

La importancia y relevancia de este problema/tema pueden entenderse desde varios puntos de vista:

- **Salud Pública:**
  La salud es un aspecto fundamental en la calidad de vida de las personas y también influye en los sistemas de salud pública. Comprender los factores que contribuyen a la obesidad y otros problemas de salud puede ayudar a desarrollar estrategias de prevención y tratamiento más efectivas.

- **Investigación Científica:**
  Los datos recopilados pueden servir como base para la investigación científica en el campo de la salud y la nutrición. Los científicos pueden utilizar estos datos para identificar patrones, correlaciones y posibles causas de diversos problemas de salud, lo que contribuye al avance del conocimiento en la materia.

- **Intervenciones Personalizadas:**
  Comprender los hábitos y comportamientos de las personas puede ser crucial para diseñar intervenciones personalizadas. Por ejemplo, el monitoreo calórico y la actividad física son elementos clave en el control del peso. Identificar patrones de comportamiento puede ayudar a desarrollar estrategias personalizadas para mejorar la salud de las personas.

- **Prevención de Enfermedades:**

El análisis de los datos puede proporcionar información valiosa sobre los factores de riesgo asociados con enfermedades crónicas como la obesidad, la diabetes y las enfermedades cardiovasculares. Con esta información, se pueden implementar medidas preventivas para reducir la incidencia de estas enfermedades.

- **Políticas de Salud:**
  Los resultados derivados de este tipo de estudios pueden influir en el desarrollo de políticas de salud pública. Por ejemplo, si se identifican patrones de consumo de alimentos poco saludables, las autoridades pueden implementar políticas para promover una alimentación más equilibrada.

En resumen, entender los factores que contribuyen a la salud y la obesidad es crucial para abordar estos problemas a nivel individual y comunitario. Además, la información recopilada puede tener implicaciones significativas en términos de políticas de salud pública y enfoques preventivos y terapéuticos personalizados.

2.3. Objetivos del Proyecto

2.3.1. Objetivo General

El objetivo general de este proyecto es avanzar en la comprensión de los factores relacionados con la obesidad y desarrollar un modelo de clasificación multiclase capaz de predecir los niveles de obesidad en individuos. La aplicación de técnicas de aprendizaje automático utilizando algoritmos de aprendizaje supervisado permitirá la creación de un sistema predictivo preciso.

Este modelo obtenido será resultado de un proceso que abarcará desde un detallado Análisis Exploratorio de Datos (EDA), donde se estudiará la distribución, las relaciones y las tendencias presentes en el conjunto de datos, hasta el posterior Preprocesamiento de Datos, que incluirá tareas de limpieza y transformación para preparar el conjunto de datos para su uso en modelos de aprendizaje automático.

Una vez preparado el conjunto de datos, se procederá al desarrollo de un Modelo Predictivo. Este modelo utilizará técnicas de aprendizaje automático para predecir niveles de obesidad basándose en los atributos proporcionados en el conjunto de datos.

Posteriormente, se realizará una Evaluación del Modelo, donde se medirá el rendimiento del modelo utilizando métricas pertinentes, y se realizarán ajustes según sea necesario para mejorar su eficacia. Finalmente, se llevará a cabo una Interpretación de Resultados que proporcionará hallazgos clave del análisis y del modelo predictivo, contribuyendo así a informar decisiones y acciones futuras en el ámbito de la prevención y gestión de la salud, así como en el diseño de estrategias más efectivas de intervención y promoción de un estilo de vida saludable.

2.3.2. Objetivo Específico

La asignación del nivel de obesidad se obtiene comúnmente a partir del Índice de Masa Corporal (IMC). Sin embargo, el IMC depende directamente de la relación entre el peso y la altura de una persona. Esta relación introduce sesgos, ya que el mismo peso puede resultar en diferentes categorías de obesidad según la altura de la persona. Por ejemplo, el nivel de obesidad de una persona que pesa 70 kg variará si mide 1.50 m o 1.70 m.

El objetivo de este proyecto es crear un modelo de clasificación capaz de determinar los niveles de obesidad basándose únicamente en el peso de una persona y en un conjunto específico de variables relacionadas con sus antecedentes familiares, hábitos de vida y actividades diarias. Se busca evitar el uso de la altura en este modelo, ya que dicha variable es determinante en el cálculo del Índice de Masa Corporal (IMC) y, por lo tanto, en la clasificación de los niveles de obesidad. La finalidad es lograr un modelo más robusto que proporcione una evaluación precisa de la obesidad, permitiendo así una comprensión más completa de los factores que contribuyen a esta condición.

3. METODOLOGÍA

3.1. Datos Utilizados

3.1.1. Fuente de Datos:

El dataset utilizado en este proyecto se obtuvo de: https://www.kaggle.com/datasets/aravindpcoder/obesity-or-cvd-risk-classifyregressorcluster

3.1.2. Descripción de la Fuente de Datos:

Los datos utilizados en este proyecto provienen de un conjunto de datos relacionados con la obesidad. Este conjunto de datos contiene información sobre variables como género, edad, altura, peso, antecedentes familiares, hábitos alimenticios, actividad física, entre otros. La relevancia radica en su potencial para identificar factores de riesgo y comprender mejor las relaciones entre diferentes variables y niveles de obesidad.

3.1.3. Relevancia de los Datos

La obesidad es un problema de salud global con múltiples factores contribuyentes. El conjunto de datos proporciona una oportunidad para realizar un análisis integral y desarrollar modelos predictivos que ayuden en la comprensión y prevención de esta condición. Al examinar las características individuales y sus correlaciones, se pueden obtener conocimientos valiosos para informar políticas de salud y programas de intervención.

3.2. Herramientas y Tecnologías

A continuación se detallan las herramientas y tecnologías empleadas en el desarrollo de este proyecto:

3.2.1. Entorno de Desarrollo:

En el desarrollo de este proyecto, se empleó Google Colab como entorno de desarrollo. Google Colab proporciona un entorno de ejecución en la nube que permite ejecutar código Python de manera interactiva y colaborativa. Esta elección facilita la colaboración en línea y el acceso a recursos computacionales poderosos sin la necesidad de configurar entornos locales.

3.2.2. Bibliotecas de Python Utilizadas:

Las bibliotecas de Python utilizadas en este código son:

- **Pandas:** Utilizada para la manipulación y análisis de datos mediante el uso de DataFrames.

- **NumPy:** Empleada para realizar operaciones numéricas eficientes en matrices y arreglos.

- **Matplotlib:** Usada para la creación de visualizaciones estáticas, como gráficos y diagramas.

- **Seaborn (import seaborn as sns):** Utilizada para crear visualizaciones estadísticas atractivas y mejorar la presentación de los gráficos.

- **ipywidgets:** Utilizada para la creación de widgets interactivos en el entorno de Jupyter, como menús desplegables y controles deslizantes.

- **IPython:** Importa la función display de IPython para visualizar widgets y gráficos en el notebook de Jupyter.

- **Scikit-Learn:** Utilizada para tareas relacionadas con machine learning, como división de datos, modelado, evaluación de modelos y métricas de rendimiento.

- **XGBoost:** Biblioteca para implementar el algoritmo XGBoost, utilizado para modelado predictivo y clasificación.

Estas bibliotecas cubren una amplia gama de funcionalidades, desde la manipulación de datos hasta la implementación y evaluación de modelos de machine learning, proporcionando un conjunto completo de herramientas para el análisis de datos y la construcción de modelos predictivos.

3.2.3. Tecnologías de Machine Learning:

A continuación, se mencionan algunas de las tecnologías específicas de machine learning que se implementaron en este proyecto:

- **Random Forest Classifier (RandomForestClassifier):** Un modelo de clasificación basado en la técnica de ensemble, que combina múltiples árboles de decisión para mejorar la precisión y generalización del modelo.

- **K-Nearest Neighbors (KNeighborsClassifier):** Un modelo de clasificación que clasifica nuevos puntos de datos según la mayoría de votos de sus vecinos más cercanos en el espacio de características.

- **Support Vector Machines (SVC):** Utilizado para implementar máquinas de soporte vectorial, que son eficaces para la clasificación y regresión.

- **Decision Tree Classifier (DecisionTreeClassifier):** Un modelo de clasificación basado en la estructura de un árbol de decisiones, que divide iterativamente los datos en función de las características más informativas.

- **Gaussian Naive Bayes (GaussianNB):** Un modelo de clasificación basado en el teorema de Bayes y la suposición de que las características siguen una distribución gaussiana.

- **Multinomial Naive Bayes (MultinomialNB):** Otro modelo de clasificación Naive Bayes diseñado para características discretas, comúnmente utilizado en problemas de clasificación de texto.

- **Logistic Regression (LogisticRegression):** Un modelo de clasificación que utiliza la función logística para predecir la probabilidad de pertenencia a una clase.

- **XGBoost (XGBClassifier):** Una implementación del algoritmo XGBoost, un potente algoritmo de boosting utilizado para tareas de clasificación y regresión.

- **Voting Classifier (VotingClassifier):** Un clasificador de votación que combina las predicciones de varios clasificadores para mejorar la precisión y la robustez.

- **Stacking Classifier (StackingClassifier):** Un clasificador de apilamiento que combina múltiples clasificadores mediante otro clasificador para mejorar la generalización.

Estas tecnologías abarcan diferentes enfoques y técnicas de machine learning, lo que nos permitió elegir el modelo más adecuado para nuestro proyecto. La combinación de estos modelos y algoritmos ofrece flexibilidad y versatilidad en el análisis y modelado de datos.

3.3. Proceso de Análisis/Desarrollo

Este proyecto de análisis y desarrollo se enfoca en comprender los factores asociados a la obesidad, empleando un conjunto de datos específico. A continuación, se describe paso a paso el proceso llevado a cabo.

3.3.1. Importación de Datos:

Se inicia cargando el conjunto de datos desde un archivo CSV alojado en GitHub. Se utiliza la biblioteca Pandas para almacenar los datos en un DataFrame.

3.3.2. Estructuración de Datos:

Se renombran las columnas del DataFrame a español y se visualizan los valores únicos de cada columna. Esto facilita la comprensión y preparación de los datos para el análisis.

3.3.3. Transformación de Datos:

Las columnas categóricas se transformaron en variables numéricas para facilitar el análisis. Se mapearon valores como género, antecedentes familiares, consumo calórico, entre otros, a valores numéricos. También se realizó la conversión de tipos de datos y redondeo de valores.

Se realizó el cálculo del IMC y a partir de estos datos se obtuvo la columna ‘Nivel_Obesidad’. El Índice de Masa Corporal (IMC) es una medida que se utiliza para evaluar el peso corporal en relación con la altura. Es una herramienta comúnmente utilizada para clasificar el grado de obesidad o delgadez en adultos. Viendo que hay diferencias entre los valores de Niveles de Obesidad provistos en la fuente de datos y los niveles Obesidad que se obtuvieron al partir del cálculo del IMC. Para este proyecto, utilizamos los valores derivados del cálculo del IMC y los reflejamos en la columna 'Nivel_Obesidad' ya que se desconoce el criterio de clasificación que se utilizó en el conjunto de datos original.

Para este proyecto se utilizarón los siguientes valores de IMC para clasificar los Niveles de Obesidad (World Health Organization):

- Bajo peso Menos de 18,5 de IMC
- Normal 18,5 a 24,9 de IMC
- Sobrepeso 25,0 a 29,9 de IMC
- Obesidad I 30,0 a 34,9 de IMC
- Obesidad II 35,0 a 39,9 de IMC
- Obesidad III Mayor a 40 de IMC

Se realizó una comparación de las columnas Nivel_Obesidad_Original y Nivel_Obesidad. Viendo que había diferencias entre los valores de Niveles de Obesidad provistos en la fuente de datos y los niveles Obesidad que se obtuvieron al partir del cálculo del IMC. Para este proyecto, se utilizó los valores derivados del cálculo del IMC y los reflejamos en la columna 'Nivel_Obesidad' ya que se desconoce el criterio de clasificación que se utilizó en el conjunto de datos original.

3.3.4. Análisis Exploratorio de Datos (EDA):

Descripción General y Estadísticas Descriptivas: Se muestra la información general del conjunto de datos y se calculan estadísticas descriptivas básicas cómo..

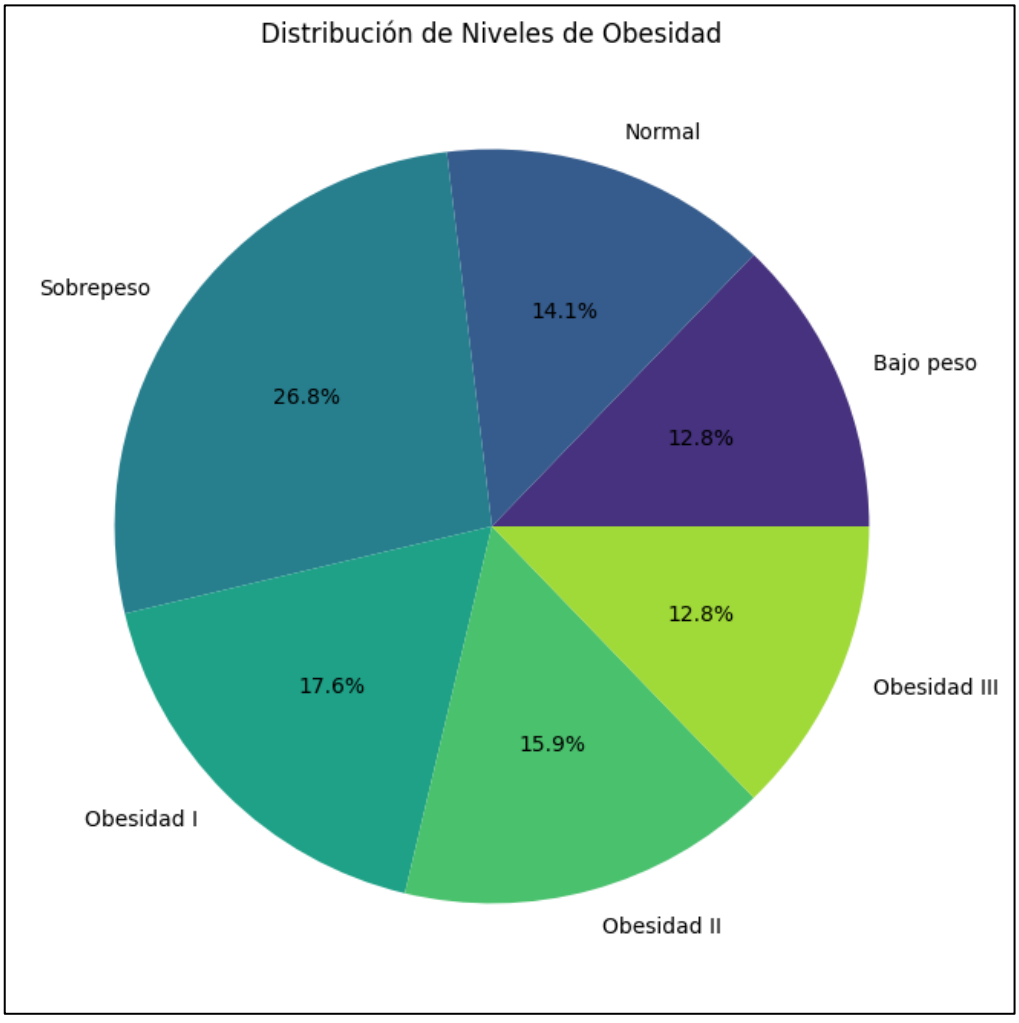
Visualización de Distribuciones: Histogramas interactivos y gráficos de barras permiten explorar la distribución de variables numéricas y categóricas.

Gráficos de Dispersión Interactivos: Se implementan gráficos de dispersión interactivos que permiten explorar relaciones entre variables seleccionadas.

A continuación se muestra algunas relaciones de como se distribuyen los datos de acuerdo a las variables que consideramos relevantes para nuestro proyecto:

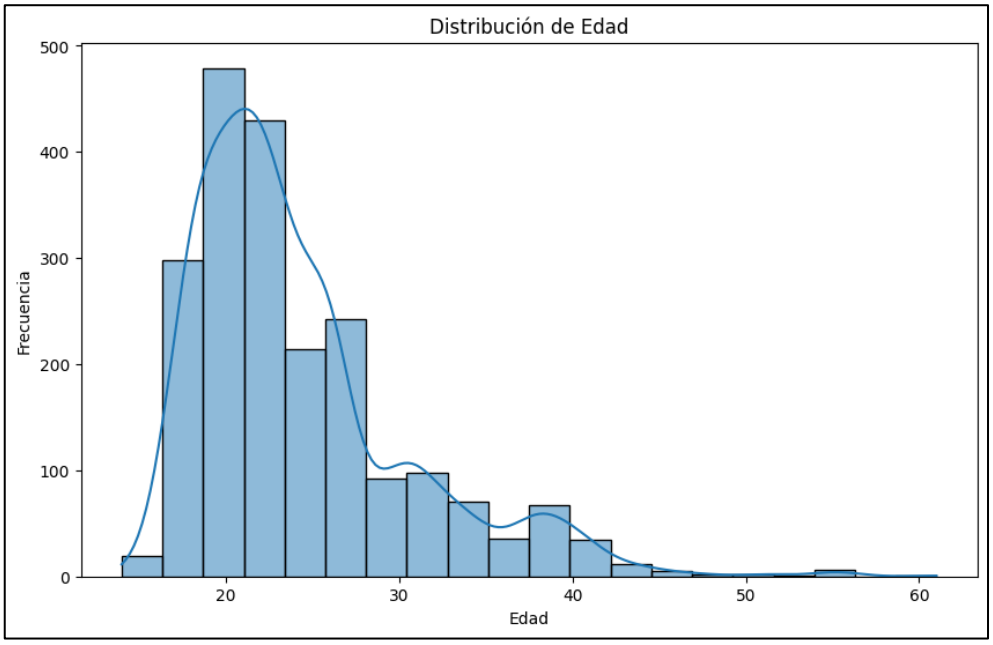
En la **Figura 3.3.4.1** se muestra la distribución de los registros del dataset de acuerdo a los Niveles de Obesidad.

Figura 3.3.4.1



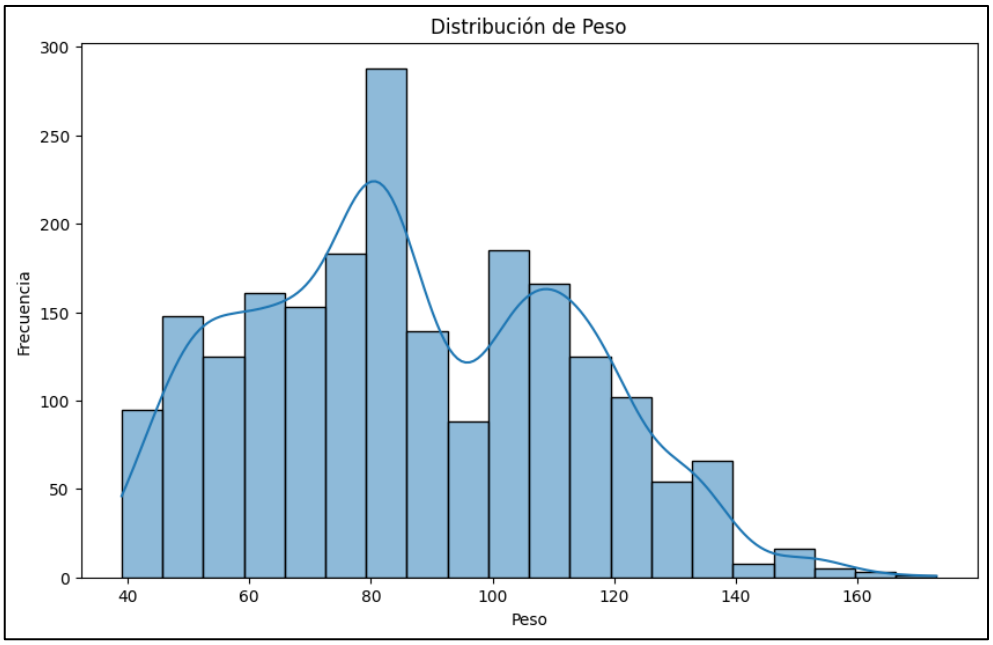
En la **Figura 3.3.4.2** se muestra la distribución de los registros del dataset de acuerdo a la Edad.

Figura 3.3.4.2



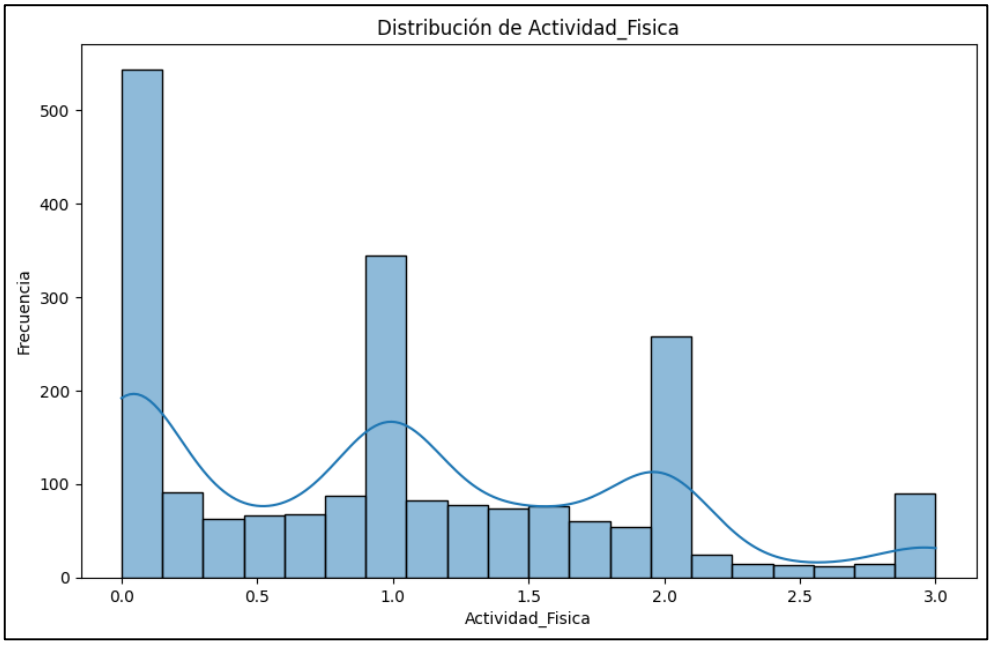
En la **Figura 3.3.4.3** se muestra la distribución de los registros del dataset de acuerdo a la Peso.

Figura 3.3.4.3



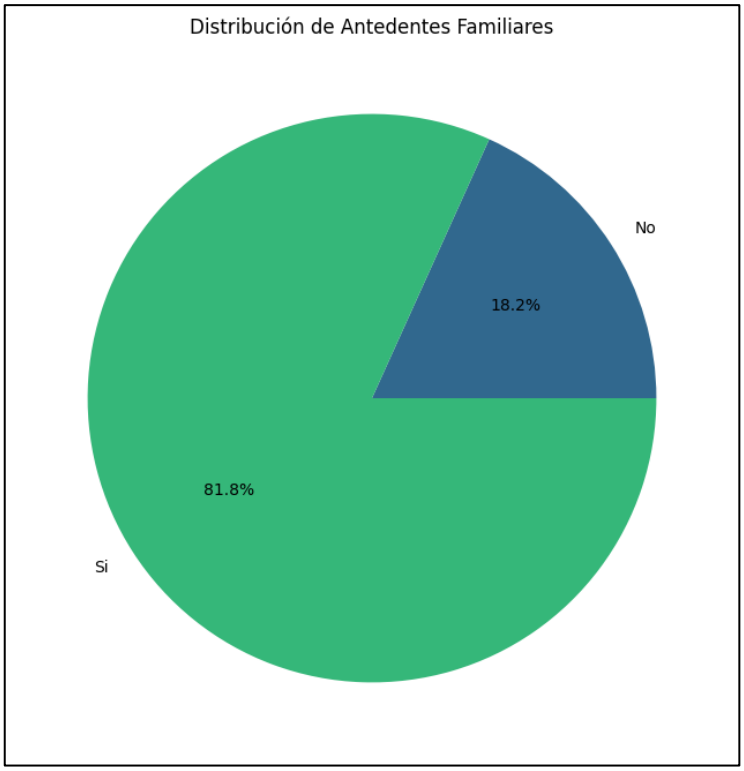
En la **Figura 3.3.4.4** se muestra la distribución de los registros del dataset de acuerdo a la Actividad física.

Figura 3.3.4.4



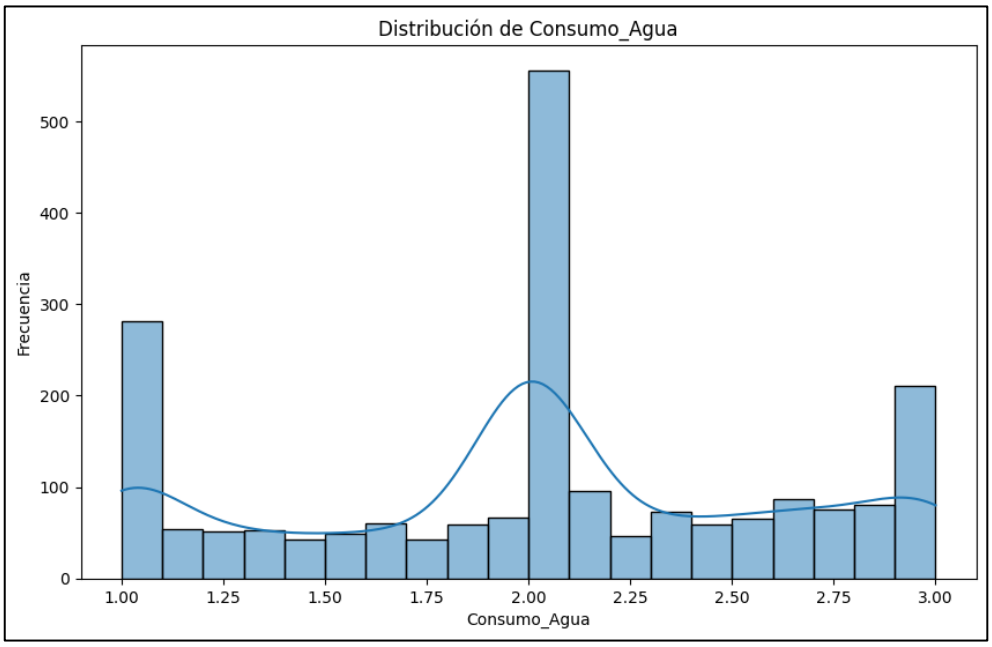
En la **Figura 3.3.4.5** se muestra la distribución de los registros del dataset de acuerdo a los Antecedentes familiares.

Figura 3.3.4.5



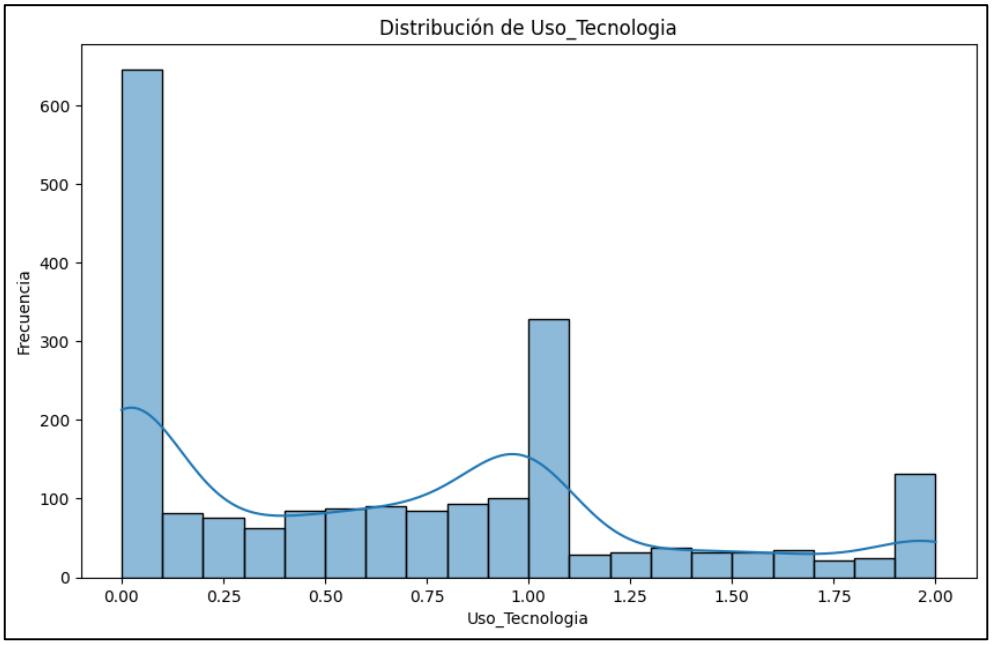
En la **Figura 3.3.4.6** se muestra la distribución de los registros del dataset de acuerdo al Consumo de Agua.

Figura 3.3.4.6



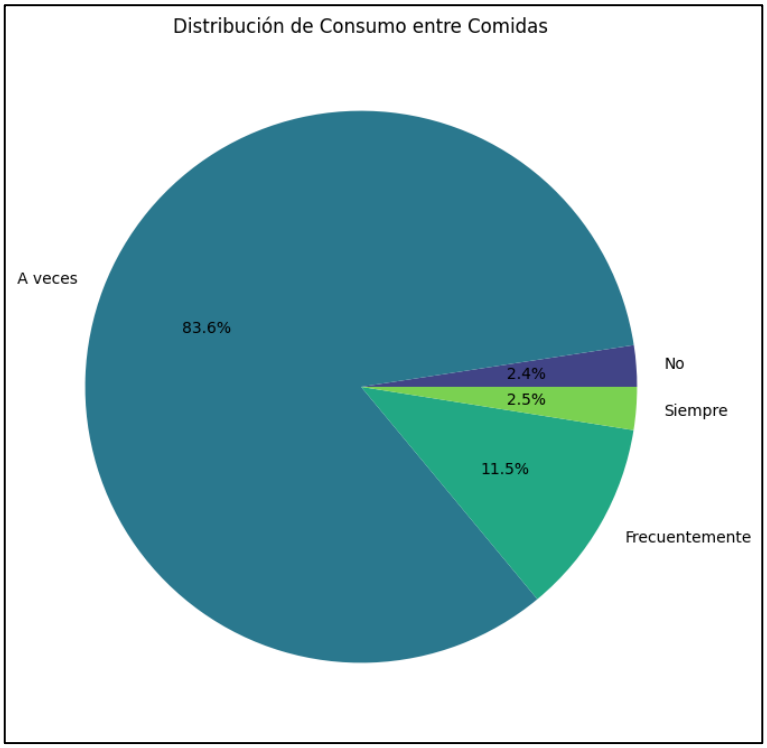
En la **Figura 3.3.4.7** se muestra la distribución de los registros del dataset de acuerdo a la Edad.

Figura 3.3.4.7



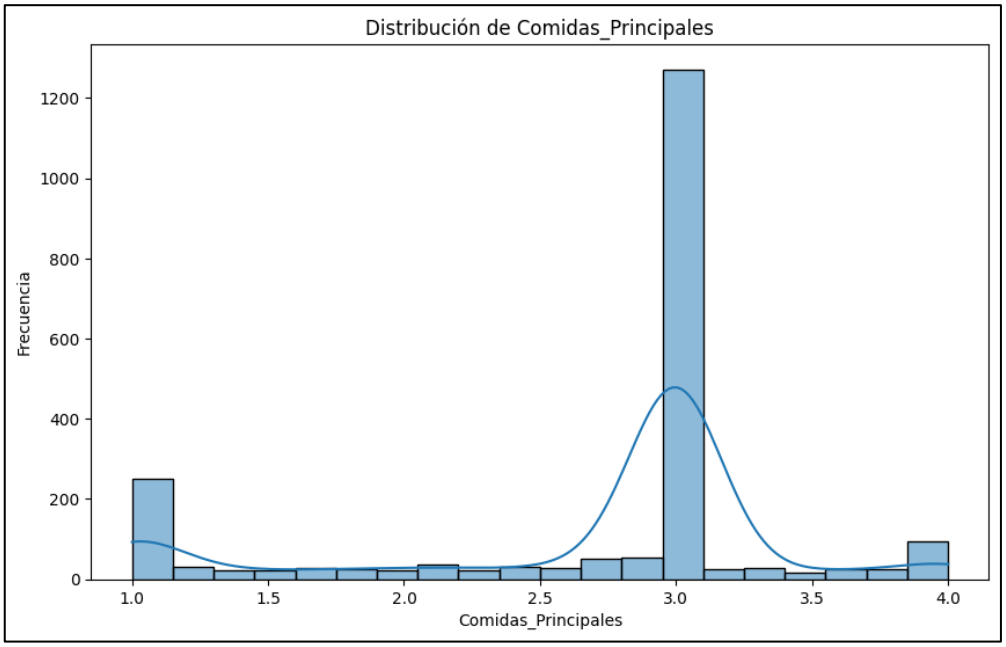
En la **Figura 3.3.4.8** se muestra la distribución de los registros del dataset de acuerdo al Consumo entre Comidas.

Figura 3.3.4.8



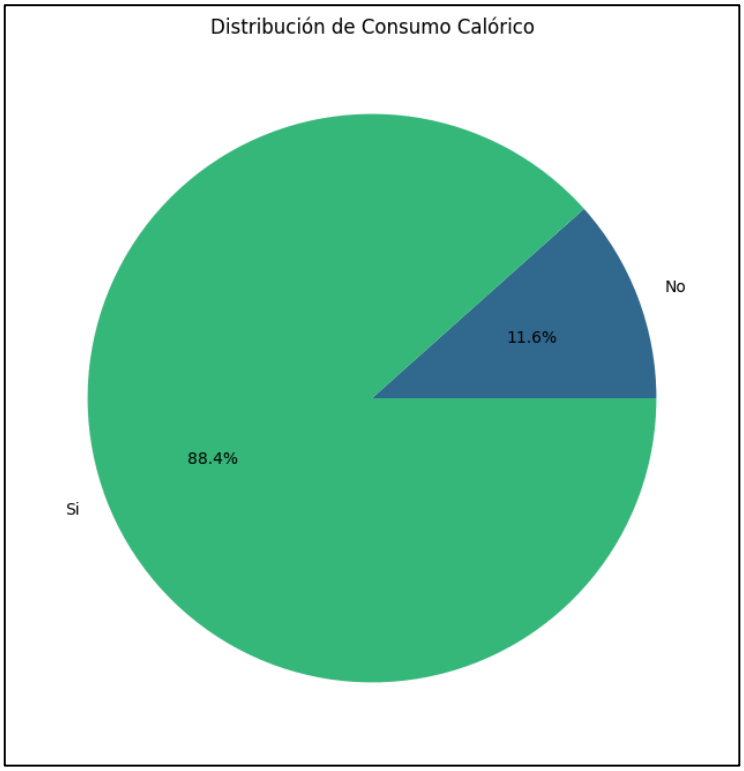
En la **Figura 3.3.4.9** se muestra la distribución de los registros del dataset de acuerdo a las Comidas Principales.

Figura 3.3.4.9



En la **Figura 3.3.4.10** se muestra la distribución de los registros del dataset de acuerdo al Consumo Calórico.

Figura 3.3.4.10



3.3.5. Aplicación de varios modelos de Machine Learning

Antes de la aplicación de los modelos de Machine Learning, se procedió a eliminar la columna Altura para cumplir con el objetivo de este proyecto: que es “reemplazar” la variable Altura (determinante en el IMC) con un conjunto específico de variables asociadas al peso, las cuales buscarán capturar de manera más precisa la relación entre la obesidad y los factores asociados a los antecedentes familiares, hábitos de vida y actividades diarias de cada individuo, evitando así la dependencia de la altura.

3.3.5.1. Identificación de características relevantes:

Para la identificación de las características más relevantes se utilizó:

MATRIZ DE CORRELACIÓN DE PEARSON:

Con el objetivo de identifica las características que son más relevantes para predecir los niveles de obesidad se utilizará como técnica de selección la Matriz de Correlación de Pearson (**Figura 3.3.5.1.),** para observar la dependencia de las características entre sí mediante el estudio de su correlación (UTEC).

Se calculará la correlación entre cada característica y la etiqueta de nivel de obesidad ('Nivel_Obesidad': 0, 1, 2, 3, 4, 5, 6). Las características con correlaciones más altas (positivas o negativas) pueden considerarse como las más relevantes para el modelo de predicción.

Correlación positiva: Un valor de correlación cercano a 1 indica una correlación positiva fuerte. Esto significa que a medida que el valor de una característica aumenta, es más probable que la etiqueta de preferencia también sea 1.

Correlación negativa: Un valor de correlación cercano a -1 indica una correlación negativa fuerte. Esto significa que a medida que el valor de una característica aumenta, es menos probable que la etiqueta de preferencia sea 1 y más probable que sea 0.

Las variables con mayor correlación con 'Nivel_Obesidad' son:

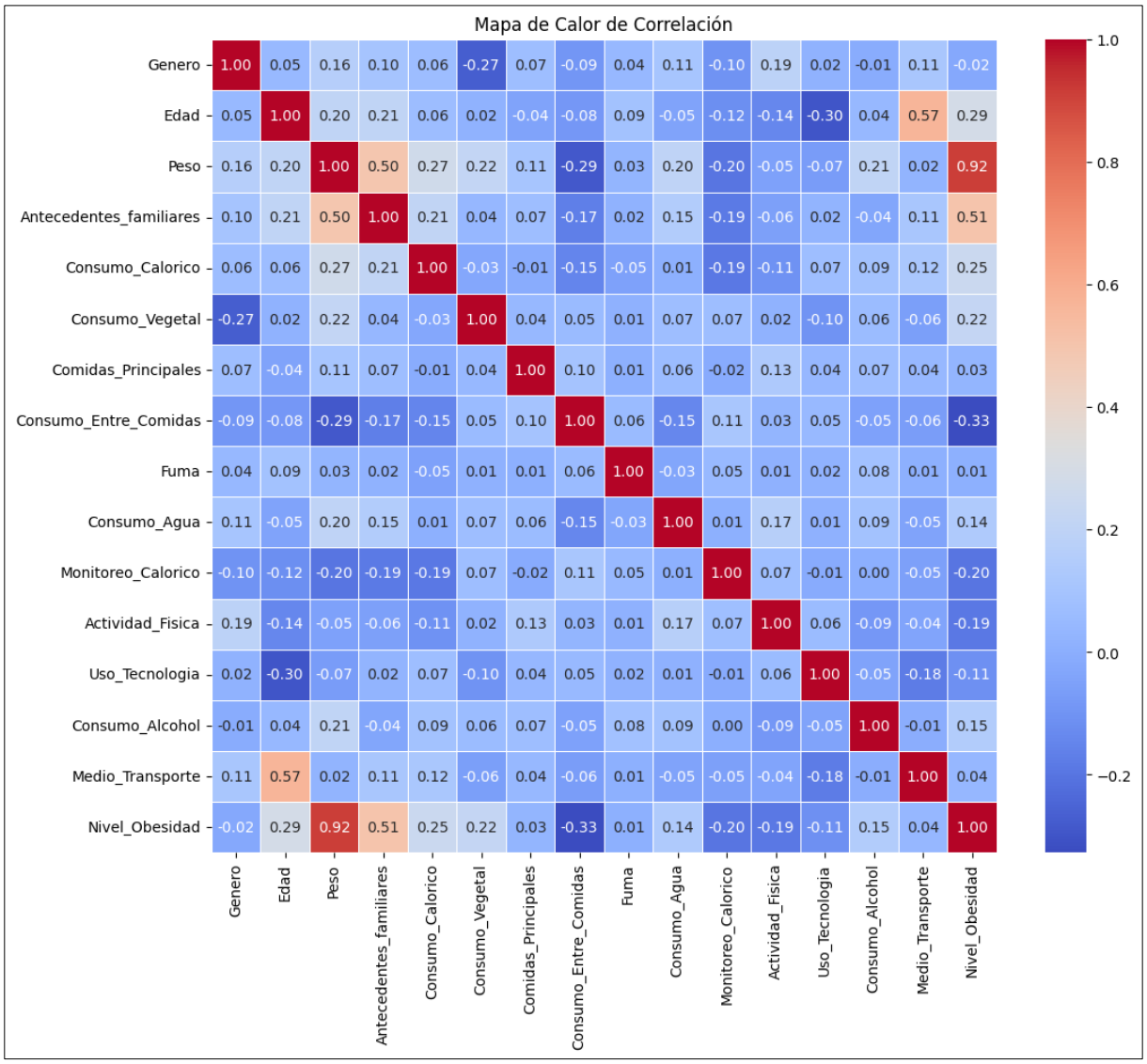
Coeficiente positivo:
-Peso: 0.92
-Antecedentes_familiares: 0.51
-Edad: 0.29
-Consumo_Calorico: 0.25

Coeficiente negativo: Consumo_Entre_Comidas: -0.33.

Las variables con mayor correlación positiva son:
-Medio_Transporte y Edad: 0.57
-Antecedentes_familiares y Peso: 0.50

Posibles características a evaluar:
Peso,
Antecedentes_familiares,
Edad,
Consumo_Calorico,
Consumo_Entre_Comidas

Figura 3.3.5.1.1.



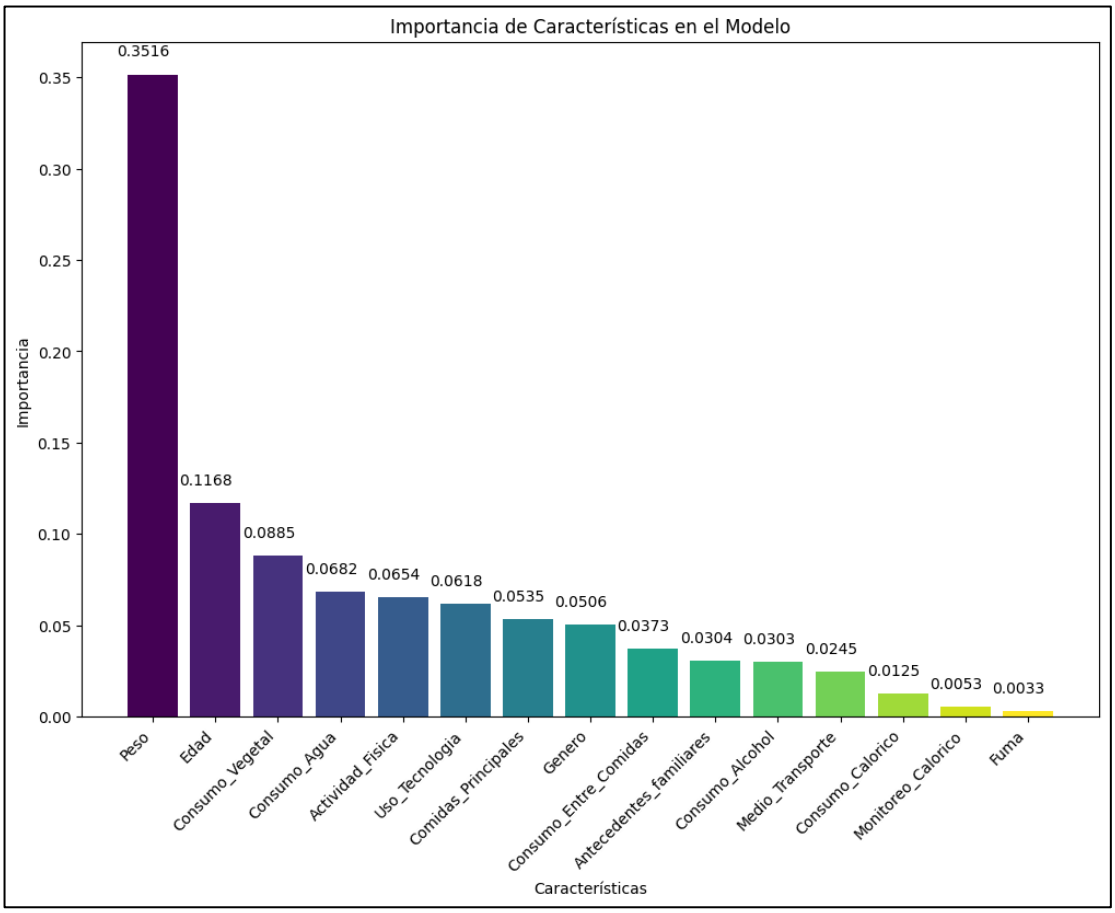
RANDOM FOREST:

Se va a utilizar el algoritmo Random Forest para determinar la importancia de características al predecir los factores que influyen en el desarrollo de la obesidad.

Random Forest calcula la importancia de cada característica durante el proceso de entrenamiento de los árboles de decisión. Esta importancia se basa en cuánto contribuye cada característica a la reducción de la impureza o el error en las predicciones. Cuanto más se utiliza una característica para tomar decisiones en los árboles y cuánto mejora la precisión del modelo, mayor será su importancia.

Una vez que Random Forest ha sido entrenado, puedes obtener un ranking de importancia de las características. Esto te proporciona información sobre cuáles características son las más influyentes en la tarea de predicción. Puedes usar este ranking para identificar las características más relevantes como se muestra en la **Figura 3.3.5.2.2.**

Figura 3.3.5.2.2.



3.3.5.2. Separar en datos de entrenamiento y datos de prueba utilizando las características más relevantes:

Se va a separar las columnas del dataframe de la siguiente forma:

Variables de entrada X: Array de Arrays de numpy en el que cada array tiene las variables de entrada de un elemento: **'Peso', 'Edad', 'Actividad_Fisica', 'Antecedentes_familiares', 'Consumo_Agua', 'Uso_Tecnologia', 'Consumo_Entre_Comidas', 'Comidas_Principales', 'Consumo_Calorico'**. Estas

serían las características más relevantes que se seleccionaron para entrenar los modelos.

Variable de salida y: Array de numpy en el que cada posición del array contiene la salida o el valor esperado del elemento del Dataset: 'Nivel_Obesidad'

3.3.5.3.Modelos de Machine Learning aplicados:

En todos los modelos se utilizó **Grid Search** para ajustar los hiperparametros y poder obtener el mejor modelo (Scikit-learn).

Cada modelo justado con los mejores hiperparametros se van a evaluar en el conjunto de prueba (X_test, y_test) y se mostraran los resultados obteniendos.

A continuación se detallan los modelos que se aplicaron en este proyecto:

- **MODELO KNN:**
  K-Nearest-Neighbor es un algoritmo basado en instancia de tipo supervisado de Machine Learning. Puede usarse para clasificar nuevas muestras (valores discretos) o para predecir (regresión, valores continuos). Sirve esencialmente para clasificar valores buscando los puntos de datos "más similares" (por cercanía) aprendidos en la etapa de entrenamiento y haciendo conjeturas de nuevos puntos basado en esa clasificación (Aprende Machine Learning).

- **SVM**
  Support Vector Machine (SVM) es un tipo de algoritmo de clasificación y regresión en aprendizaje supervisado contenido en el aprendizaje automático. El concepto SVM se denomina intento de encontrar el mejor hiperplano que dividirá los datos en dos clases en el espacio de entrada. El principal objetivo del proceso de formación sobre el concepto SVM es encontrar la ubicación del hiperplano. El método SVM utiliza la función del producto escalar. El hiperplano es la línea utilizada para separar el conjunto de datos. El hiperplano puede ser una línea en dos dimensiones y puede ser un plano en varios planos (Medium).

El método SVM se divide en dos tipos según sus características: SVM lineal y SVM no lineal.

KERNEL LINEAR: sirve para clasificar datos que se pueden separar linealmente en dos clases utilizando márgenes suaves. La clasificación lineal generalmente se aplica a conjuntos de datos que tienen dimensiones más bajas donde el conjunto de datos tiene pocas características para clasificar.

KERNEL NO-LINEAR: utiliza el concepto de kernel en un espacio de trabajo de alta dimensión. El concepto de kernel es una función que se utiliza modificando el algoritmo SVM para resolver problemas no lineales.

Kernels no lineales aplicados:

- **SVM CON KERNEL RBF:**
  El kernel RBF es el kernel más utilizado para resolver el problema de clasificar conjuntos de datos que no se pueden separar linealmente. Se sabe que este kernel tiene un buen rendimiento con ciertos parámetros y los resultados del entrenamiento tienen un valor de error pequeño en comparación con otros kernels. El RBF del núcleo gaussiano tiene dos parámetros, a saber, gamma y sigma. Cuando gamma es alta, es probable que los puntos alrededor de los datos se consideren en el cálculo. El parámetro sigma se utiliza para encontrar el valor óptimo para cada conjunto de datos.
- **SVM CON KERNEL POLY:** es una función del núcleo adecuada para su uso en máquinas de vectores de soporte (SVM) y otras kernelizaciones, donde el núcleo representa la similitud de los vectores de muestra de entrenamiento en un espacio de características. Los núcleos polinomiales también son adecuados para resolver problemas de clasificación en conjuntos de datos de entrenamiento normalizados. El núcleo polinómico tiene un parámetro de grado (d) que funciona para encontrar el valor óptimo en cada conjunto de datos. El parámetro d es el grado de la función kernel polinomial con un valor predeterminado de d = 2. Cuanto mayor sea el valor d, la precisión del sistema

resultante fluctuará y será menos estable. Esto sucede porque cuanto mayor es el valor del parámetro d, más curvada será la línea del hiperplano resultante.

- **SVM CON KERNEL SIGMOID:** El núcleo sigmoideo se aplica ampliamente en redes neuronales para procesos de clasificación. La clasificación SVM con el núcleo sigmoideo tiene una estructura compleja y es difícil para los humanos interpretar y comprender cómo el núcleo sigmoideo toma decisiones de clasificación. El interés en estos núcleos surge de su éxito en la clasificación con la red neuronal y la regresión logística, propiedades específicas, linealidad y distribución acumulativa. El núcleo sigmoideo es generalmente problemático o inválido porque es difícil tener parámetros positivos. La función sigmoidea ahora no se usa ampliamente en la investigación porque tiene un gran inconveniente, a saber, que el rango de valores de salida de la función sigmoidea no está centrado en cero. Esto provoca que se produzca el proceso de retropropagación que no es ideal, de modo que el peso de la ANN no se distribuye uniformemente entre valores positivos y negativos y tiende a acercarse a los valores extremos 0 y 1.

- **ARBOL DE DESICIÓN**

  Los arboles de decisión son representaciones gráficas de posibles soluciones a una decisión basadas en ciertas condiciones, es uno de los algoritmos de aprendizaje supervisado más utilizados en machine learning y pueden realizar tareas de clasificación o regresión. Los árboles de decisión tienen un primer nodo llamado raíz (root) y luego se descomponen el resto de atributos de entrada en dos ramas (podrían ser más, pero no nos meteremos en eso ahora) planteando una condición que puede ser cierta o falsa. Se bifurca cada nodo en 2 y vuelven a subdividirse hasta llegar a las hojas que son los nodos finales (Aprende Machine Learning).

- **NAIVE BAYES**

  El clasificador Naive Bayes es un algoritmo de aprendizaje automático simple pero potente que se puede utilizar para tareas de clasificación. Se basa en el teorema de Bayes, que es una fórmula matemática que describe la probabilidad de que ocurra un evento dado el conocimiento de otros eventos.El clasificador Naive Bayes funciona calculando la probabilidad de cada clase dadas las características de

entrada. Luego se predice como resultado la clase con la mayor probabilidad (Artificial Intelligence in Plain English).

Clasificadores Naive Bayes aplicados en este proyecto:

- **GAUSSIANNB:** Este clasificador se utiliza para tareas de clasificación donde las características de entrada son continuas y normalmente distribuidas.
- **MULTINOMIALNB:** Este clasificador se utiliza para tareas de clasificación donde las características de entrada son recuentos discretos. Se supone que las características de entrada se generan a partir de una distribución multinomial, donde cada característica representa un recuento de un evento o categoría particular.

- **REGRESION LOGISTICA**

  La regresión logística es un modelo estadístico para estudiar las relaciones entre un conjunto de variables cualitativas Xi y una variable cualitativa Y. Se trata de un modelo lineal generalizado que utiliza una función logística como función de enlace. Un modelo de regresión logística también permite predecir la probabilidad de que ocurra un evento (valor de 1) o no (valor de 0) a partir de la optimización de los coeficientes de regresión. Este resultado siempre varía entre 0 y 1. Cuando el valor predicho supera un umbral, es probable que ocurra el evento, mientras que cuando ese valor está por debajo del mismo umbral, no es así (DataScientest).

- **XGBOOST**

  XGBoost (potenciación del gradiente eXtreme) es una implementación de código abierto popular y eficiente del algoritmo de árboles aumentados de gradientes. La potenciación de gradientes es un algoritmo de aprendizaje supervisado que intenta predecir de forma apropiada una variable de destino mediante la combinación de un conjunto de estimaciones a partir de un conjunto de modelos más simples y más débiles. El algoritmo XGBoost funciona bien en competiciones de aprendizaje automático debido a su manejo robusto de una variedad de tipos de datos, relaciones, distribuciones y la variedad de hiperparámetros que puede ajustar. Puede usar XGBoost para problemas de regresión, de clasificación (binaria y multiclase) y de ranking (Amazon Web Services).

3.3.6. Ensamble de los Mejores Modelos

El ensamblado es el arte de combinar un conjunto de diversos de modelos para improvisar sobre la estabilidad y el poder predictivo del modelo. Los modelos pueden ser diferentes entre sí por una variedad de razones. Esta es la idea básica de un conjunto: combinar predicciones de varios modelos, promedia errores idiosincráticos y produce mejores predicciones generales (AprendeIA).

Ensambles que se aplicaron en este proyecto:

- **Ensamble por Votación:** El ensamble de votación con voto duro es una técnica de aprendizaje automático que se utiliza para combinar las predicciones de varios modelos base con el objetivo de mejorar el rendimiento general del modelo. En los problemas de clasificación existen dos variantes: Hard Voting y Soft Voting. En el esquema de Hard Voting la clase final es la que tenga mayor cantidad de votos. Es un sistema democrático. Cuando se hace uso de Soft Voting lo que tomamos en cuenta son las probabilidades de cada clase. Para cada clasificador obtenemos la probabilidad de la clase1, la de la clase2, etc. Luego se promedian los valores de probabilidad de cada clase y se selecciona la que tenga el mayor valor promedio (Medium).
- **Ensamble apilado (stacking):** El ensamble de apilado, también conocido como ensamble stacking, es una técnica avanzada de ensamblado que combina las predicciones de varios modelos base utilizando otro modelo denominado "meta-modelo" o "modelo de nivel superior". La idea clave detrás del ensamble de apilado es permitir que el meta-modelo aprenda cómo combinar las fortalezas y debilidades de los modelos base para hacer predicciones más precisas. Este enfoque puede ser particularmente útil cuando los modelos base son diversos y pueden aportar diferentes perspectivas al problema (Medium).

4. RESULTADOS

En cada modelo aplicado se realizó una evaluación simple del rendimiento utilizando la visualización de la **Matriz de Confusión**. La matriz de confusión es una herramienta que se utiliza en problemas de clasificación para evaluar el rendimiento de un modelo. Es especialmente útil cuando se tiene un conjunto de datos etiquetado, y se desea comprender cómo el modelo clasifica las instancias en comparación con las etiquetas reales (IMC Smart Education).

La matriz de confusión muestra la cantidad de verdaderos positivos (TP), falsos positivos (FP), verdaderos negativos (TN) y falsos negativos (FN) de un modelo de clasificación. Estos términos se definen de la siguiente manera:

Verdaderos Positivos (TP): Instancias que fueron clasificadas correctamente como positivas.

Falsos Positivos (FP): Instancias que fueron clasificadas incorrectamente como positivas.

Verdaderos Negativos (TN): Instancias que fueron clasificadas correctamente como negativas.

Falsos Negativos (FN): Instancias que fueron clasificadas incorrectamente como negativas.

Para una evaluación más detallada del rendimiento del modelo se calculó las de métricas a partir de las etiquetas reales (y_test) y las predicciones del modelo (y_pred) en el conjunto de prueba utilizando la librería sklearn.metrics (Scikit-learn):

Exactitud (Accuracy): La exactitud es una medida general de qué tan bien el modelo clasifica las instancias. Se calcula como el número de predicciones correctas (verdaderos positivos y verdaderos negativos) dividido por el número total de instancias.

Precisión (Precision): La precisión mide la proporción de instancias positivas predichas correctamente entre todas las instancias clasificadas como positivas (falsos positivos).

Recall (Sensibilidad): El recall mide la proporción de instancias positivas que fueron clasificadas correctamente entre todas las instancias realmente positivas. Es útil cuando se quiere asegurar que no se pierdan instancias positivas.
F1-score: El F1-score es la media armónica entre precisión y recall. Proporciona un equilibrio entre ambas métricas. Es útil cuando hay un desequilibrio entre las clases.

También se utilizó la **Validación Cruzada K-Fold** que es una técnica comúnmente utilizada para evaluar el rendimiento de un modelo de aprendizaje automático. Permite garantizar que todas las observaciones de la serie de datos original tengan la oportunidad de aparecer en la serie de entrenamiento y en la serie de prueba. En caso de datos de entrada limitados, resulta uno de los mejores enfoques. Primero se empieza separando la serie de datos de manera aleatoria en K folds. El procedimiento tiene un parámetro único llamado “K” que hace referencia al número de grupos en el se dividirá la muestra. El valor de K no debe ser ni demasiado bajo ni demasiado alto y, por lo general, se elige un valor comprendido entre 5 y 10 en función de la envergadura de la serie de datos. Por ejemplo, si K=10, la serie de datos se dividirá en 10 partes. Un valor K más alto lleva a un modelo con menos sesgo, pero una varianza demasiado amplia puede llevar a un ajuste excesivo. Un valor más bajo es prácticamente lo mismo que utilizar el método Train-Test Split. Después se ajusta el modelo utilizando los folds K-1 (K menos 1). El modelo se valida usando el K-fold restante. Las puntuaciones y los errores se deben anotar. El proceso se repite hasta que cada K-fold sirva dentro de la serie de entrenamiento. La media de las puntuaciones registradas es la métrica de rendimiento del modelo (DataScientest).

4.1. Presentación e Interpretación de los Resultados de los Modelos

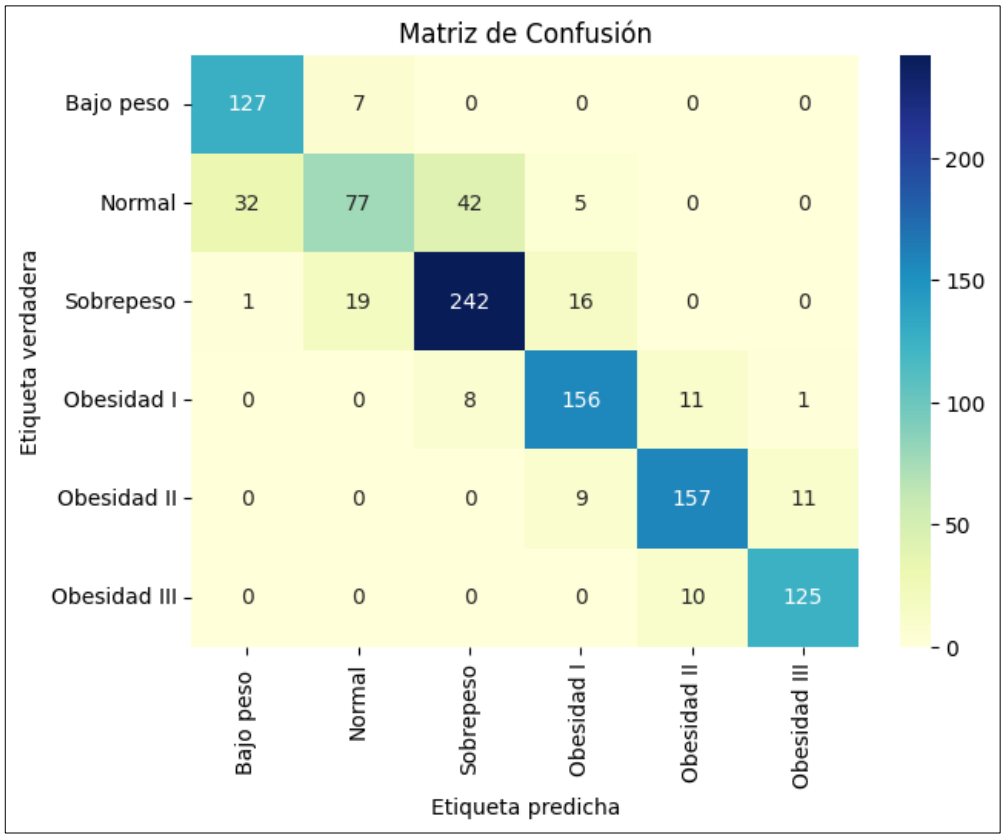
- **MODELO KNN:**

  Los resultados obtenidos a partir del mejor modelo KNN son los siguientes:

  Mejor modelo: KNeighborsClassifier(n_neighbors=3, weights='distance')
  Mejores hiperparámetros: {'n_neighbors': 3, 'weights': 'distance'}
  Exactitud en el conjunto de prueba: 0.8371212121212122

  En la **Figura 4.1.1** se muestra la evaluación del rendimiento del modelo a través de la Matriz de Confusión.

Figura 4.1.1



Métricas de SKLEARN obtenidas:
Exactitud (Accuracy): 0.8371212121212122
Precisión (Precision): 0.8336052866789031
Recall (Sensibilidad): 0.8371212121212122
F1-score: 0.8304079935201215

Métricas de Validación cruzada (Cross-Validation) con la técnica K-Fold
Scores de validación cruzada: [0.8463357 0.84834123 0.83886256 0.87677725 0.86729858]
Media de los scores: 0.8555230636505214
Desviación estándar de los scores: 0.014170123775050926
Exactitud promedio: 0.8555230636505214

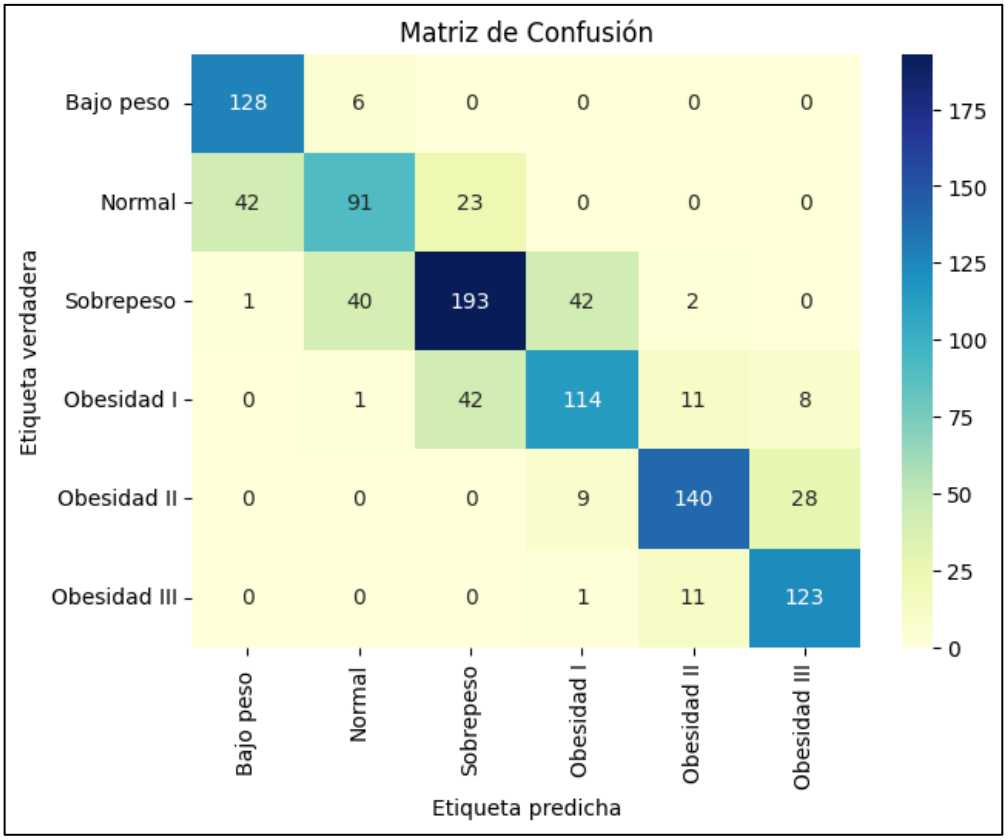
- **SVM CON KERNEL LINEAR**

  Los resultados obtenidos a partir del mejor modelo **SVM con kernel linear** son los siguientes:

  Mejor modelo: SVC(C=0.1, class_weight='balanced', kernel='linear', max_iter=1000, random_state=42, tol=0.0001)
  Mejores hiperparámetros: {'C': 0.1, 'class_weight': 'balanced', 'max_iter': 1000, 'tol': 0.0001}
  Exactitud en el conjunto de prueba: 0.7471590909090909

En la **Figura 4.1.2** se muestra la evaluación del rendimiento del modelo a través de la Matriz de Confusión.

Figura 4.1.2



Métricas de SKLEARN obtenidas:
Exactitud (Accuracy): 0.7471590909090909
Precisión (Precision): 0.7457707312632568
Recall (Sensibilidad): 0.7471590909090909
F1-score: 0.7432526635083763

Métricas de Validación cruzada (Cross-Validation) con la técnica K-Fold
Scores de validación cruzada: [0.75177305 0.76777251 0.72985782 0.73933649 0.74407583]
Media de los scores: 0.7465631407347653
Desviación estándar de los scores: 0.012762910911878542
Exactitud promedio: 0.7465631407347653

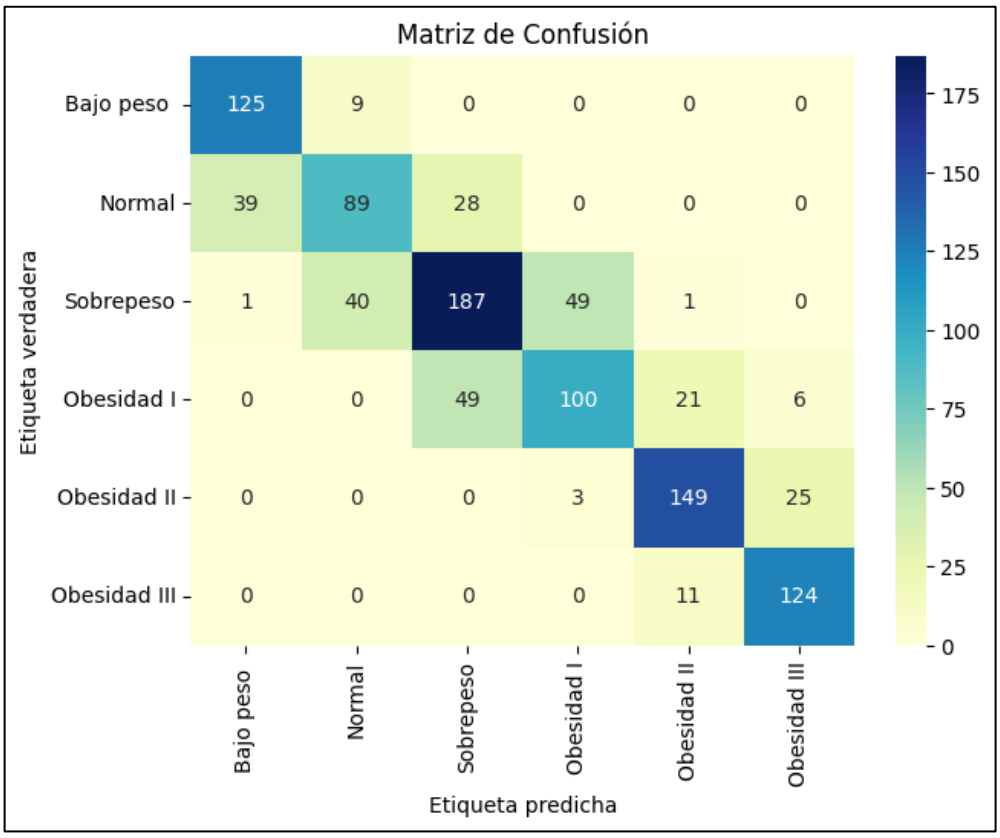
- **SVM CON KERNEL RBF**

  Los resultados obtenidos a partir del mejor modelo **SVM con kernel RBF** son los siguientes:

Mejor modelo: SVC(C=100, class_weight='balanced', max_iter=500, random_state=42, tol=0.01)
Mejores hiperparámetros: {'C': 100, 'class_weight': 'balanced', 'max_iter': 500, 'tol': 0.01}
Exactitud en el conjunto de prueba: 0.7329545454545454

En la **Figura 4.1.3** se muestra la evaluación del rendimiento del modelo a través de la Matriz de Confusión.

Figura 4.1.3



Métricas de SKLEARN obtenidas:
Exactitud (Accuracy): 0.7329545454545454
Precisión (Precision): 0.7270232773909053
Recall (Sensibilidad): 0.7329545454545454
F1-score: 0.7272817346523065

Métricas de Validación cruzada (Cross-Validation) con la técnica K-Fold
Scores de validación cruzada: [0.77068558 0.73222749 0.74407583 0.72748815 0.75829384]
Media de los scores: 0.7465541774506179
Desviación estándar de los scores: 0.016093811078665164
Exactitud promedio: 0.7465541774506179

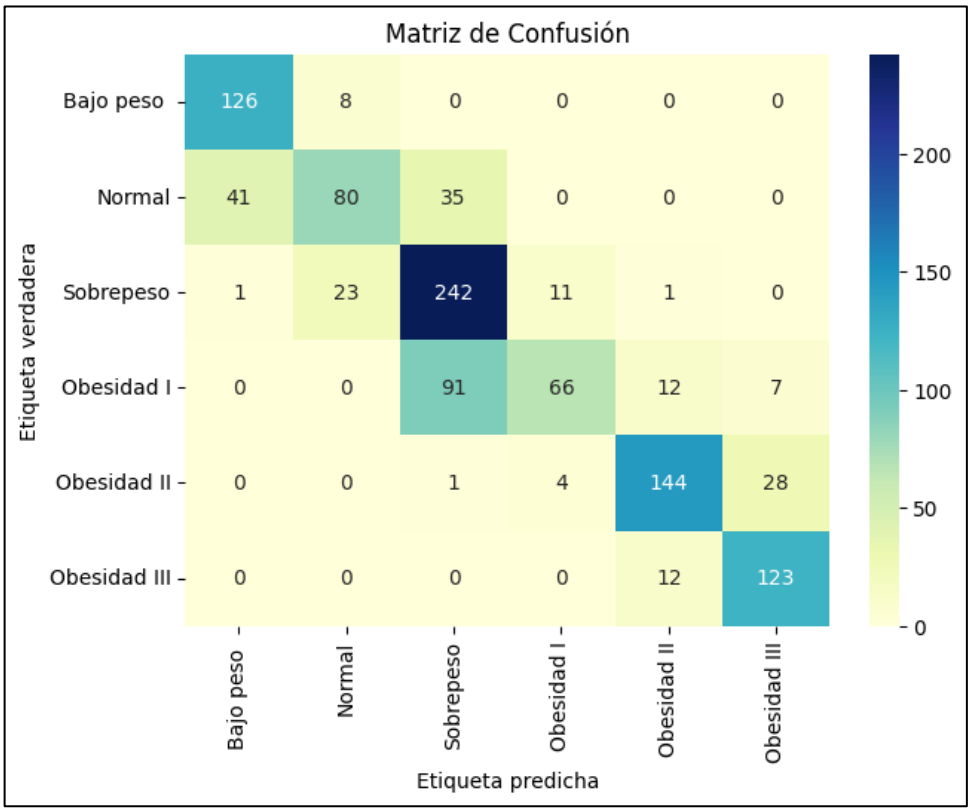
- **SVM CON KERNEL POLY**

  Los resultados obtenidos a partir del mejor modelo **SVM con kernel Poly** son los siguientes:

  Mejor modelo: SVC(C=100, coef0=0.5, degree=2, kernel='poly', max_iter=500, random_state=42)
  Mejores hiperparámetros: {'C': 100, 'coef0': 0.5, 'degree': 2, 'max_iter': 500}
  Exactitud en el conjunto de prueba: 0.7395833333333334

  En la **Figura 4.1.4** se muestra la evaluación del rendimiento del modelo a través de la Matriz de Confusión.

  Figura 4.1.4



  Métricas de SKLEARN obtenidas:
  Exactitud (Accuracy): 0.7395833333333334
  Precisión (Precision): 0.7524347801078314
  Recall (Sensibilidad): 0.7395833333333334
  F1-score: 0.723798956579992

  Métricas de Validación cruzada (Cross-Validation) con la técnica K-Fold
  Scores de validación cruzada: [0.71867612 0.6943128 0.74170616 0.63744076 0.72037915]
  Media de los scores: 0.7025029970981367
  Desviación estándar de los scores: 0.03582789168004579
  Exactitud promedio: 0.7025029970981367

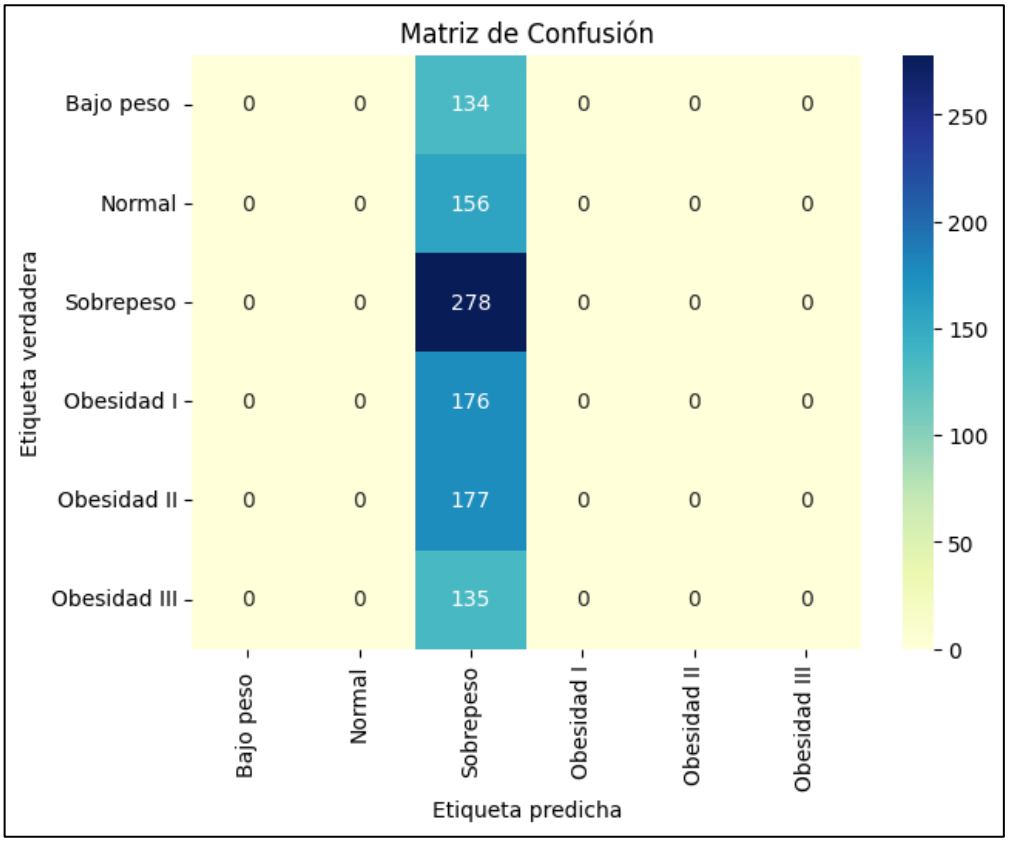
- **SVM CON KERNEL SIGMOID**

  Los resultados obtenidos a partir del mejor modelo **SVM con kernel Sigmoid** son los siguientes:

  Mejor modelo: SVC(C=0.001, kernel='sigmoid', max_iter=500, random_state=42)
  Mejores hiperparámetros: {'C': 0.001, 'class_weight': None, 'coef0': 0.0, 'max_iter': 500}
  Exactitud en el conjunto de prueba: 0.26325757575757575

  En la **Figura 4.1.5** se muestra la evaluación del rendimiento del modelo a través de la Matriz de Confusión.

  Figura 4.1.5



  Métricas de SKLEARN obtenidas:
  Exactitud (Accuracy): 0.26325757575757575
  Precisión (Precision): 0.06930455119375574
  Recall (Sensibilidad): 0.26325757575757575
  F1-score: 0.10972354731725047

  Métricas de Validación cruzada (Cross-Validation) con la técnica K-Fold
  Scores de validación cruzada: [0.25768322 0.28909953 0.27488152 0.27488152 0.24407583]

Media de los scores: 0.2681243207511232
Desviación estándar de los scores: 0.015611695953957564
Exactitud promedio: 0.2681243207511232

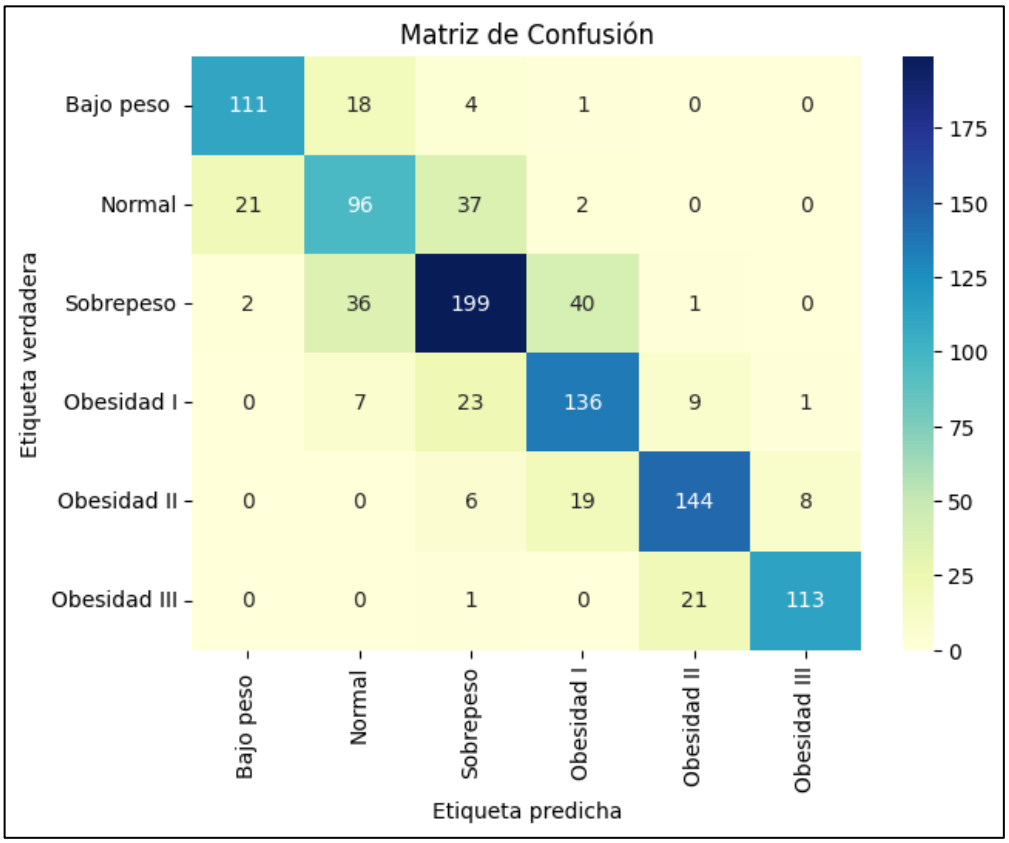
- **ARBOL DE DESICIÓN**

  Los resultados obtenidos a partir del mejor modelo **Árbol de decisión** son los siguientes:

  Mejor modelo: DecisionTreeClassifier(max_features='auto', min_samples_split=5, random_state=42)
  Mejores hiperparámetros: {'max_depth': None, 'max_features': 'auto', 'min_samples_leaf': 1, 'min_samples_split': 5}
  Exactitud en el conjunto de prueba: 0.7566287878787878

  En la **Figura 4.1.6** se muestra la evaluación del rendimiento del modelo a través de la Matriz de Confusión.

Figura 4.1.6



Métricas de SKLEARN obtenidas:
Exactitud (Accuracy): 0.7566287878787878
Precisión (Precision): 0.7602844868459318
Recall (Sensibilidad): 0.7566287878787878
F1-score: 0.7577012645354642

Métricas de Validación cruzada (Cross-Validation) con la técnica K-Fold
Scores de validación cruzada: [0.75886525 0.81516588 0.78199052 0.78672986 0.80805687]
Media de los scores: 0.790161675237807
Desviación estándar de los scores: 0.020017290694843207
Exactitud promedio: 0.790161675237807

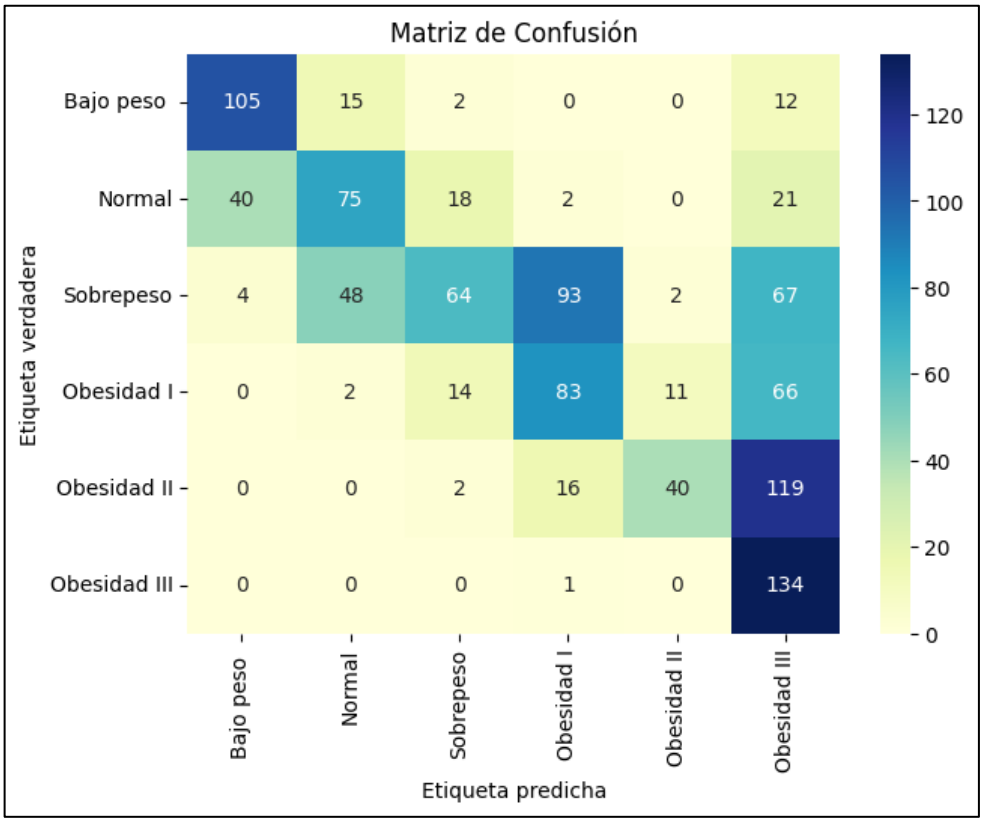
- **NAIVE BAYES GAUSSIANNB**

  El clasificador GaussianNB en scikit-learn está diseñado para ser fácil de usar y no requiere una configuración extensa de hiperparámetros, ya que se encarga de estimar los parámetros automáticamente a partir de los datos de entrenamiento:

  gnb_model = GaussianNB()

  En la **Figura 4.1.7** se muestra la evaluación del rendimiento del modelo a través de la Matriz de Confusión.

Figura 4.1.7



Métricas de SKLEARN obtenidas:
Exactitud (Accuracy): 0.4744318181818182
Precisión (Precision): 0.5753720731909509
Recall (Sensibilidad): 0.4744318181818182
F1-score: 0.45288603400522565

Métricas de Validación cruzada (Cross-Validation) con la técnica K-Fold
Scores de validación cruzada: [0.5106383 0.48341232 0.51184834 0.44075829 0.50236967]
Media de los scores: 0.4898053846929515
Desviación estándar de los scores: 0.026549018038622658
Exactitud promedio: 0.4898053846929515

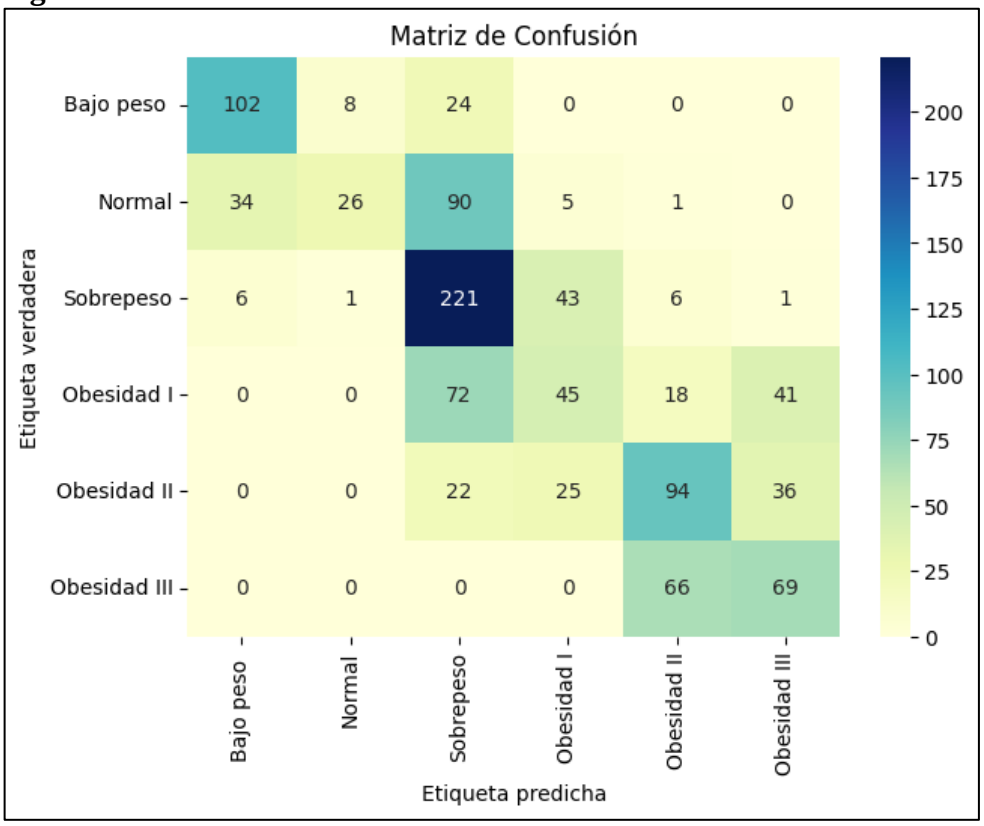
- **NAIVE BAYES MULTINOMIALNB**

  Los resultados obtenidos a partir del mejor modelo **Naive Bayes MultinomialNB** son los siguientes:

  Mejor modelo: MultinomialNB(alpha=0.1)
  Mejores hiperparámetros: {'alpha': 0.1}
  Exactitud en el conjunto de prueba: 0.5265151515151515

  En la **Figura 4.1.8** se muestra la evaluación del rendimiento del modelo a través de la Matriz de Confusión.

Figura 4.1.8



Métricas de SKLEARN obtenidas:
Exactitud (Accuracy): 0.5274621212121212
Precisión (Precision): 0.5452390935286964
Recall (Sensibilidad): 0.5274621212121212
F1-score: 0.49922146452121297

Métricas de Validación cruzada (Cross-Validation) con la técnica K-Fold
Scores de validación cruzada: [0.5106383 0.48341232 0.51184834 0.44075829 0.50236967]
Media de los scores: 0.4898053846929515
Desviación estándar de los scores: 0.026549018038622658
Exactitud promedio: 0.4898053846929515

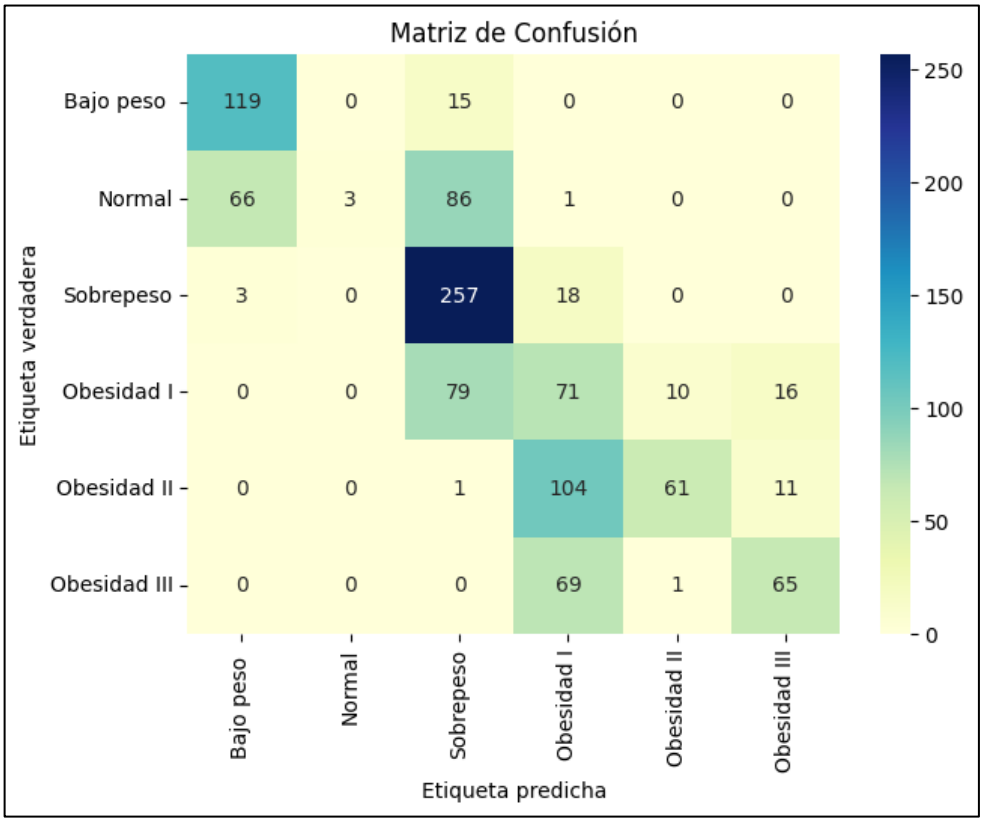
- **REGRESION LOGISTICA**

  Los resultados obtenidos a partir del mejor modelo **Regresión Logística** son los siguientes:

  Mejor modelo: LogisticRegression(C=0.001)
  Mejores hiperparámetros: {'C': 0.001}
  Exactitud en el conjunto de prueba: 0.5454545454545454

  En la **Figura 4.1.9** se muestra la evaluación del rendimiento del modelo a través de la Matriz de Confusión.

Figura 4.1.9



Métricas de SKLEARN obtenidas:
Exactitud (Accuracy): 0.5454545454545454
Precisión (Precision): 0.659838964486828
Recall (Sensibilidad): 0.5454545454545454
F1-score: 0.4975995785780562

Métricas de Validación cruzada (Cross-Validation) con la técnica K-Fold
Scores de validación cruzada: [0.54846336 0.57582938 0.4478673 0.5450237 0.48341232]
Media de los scores: 0.5201192116791592
Desviación estándar de los scores: 0.04710622444986559
Exactitud promedio: 0.5201192116791592

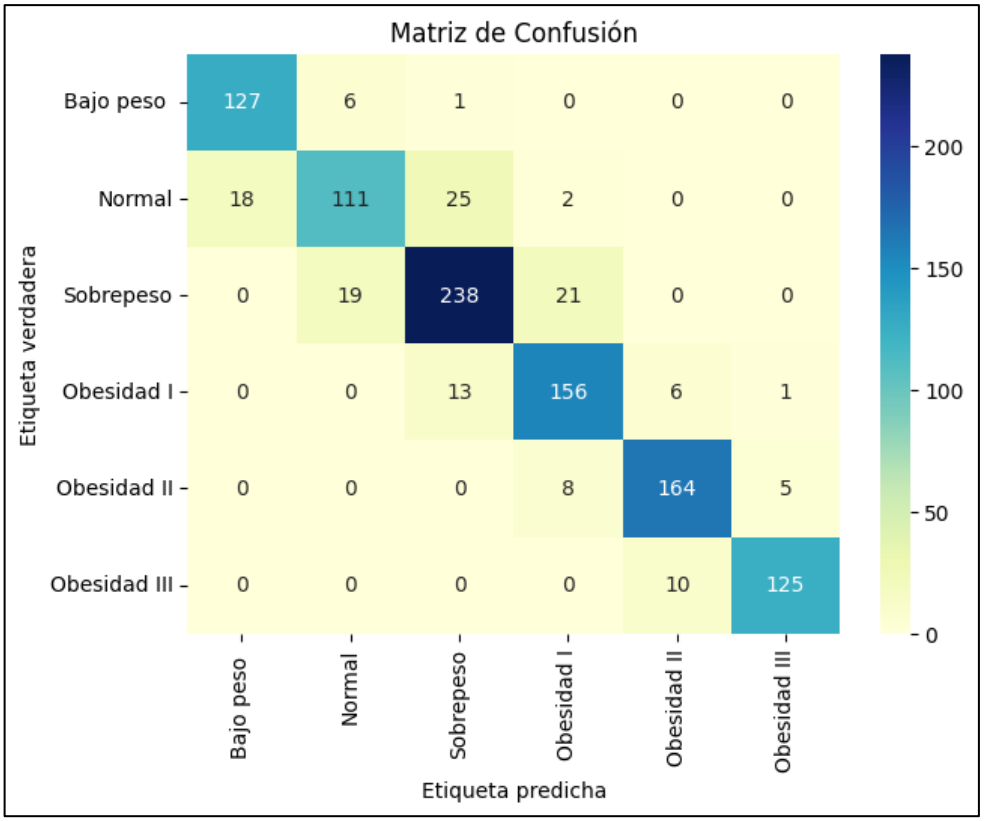
- **XGBOOST**

  Los resultados obtenidos a partir del mejor modelo **XGBOOST** son los siguientes:

  Mejor modelo: XGBClassifier(learning_rate=0.2, min_child_weight=2, n_estimators=200)
  Mejores hiperparámetros: {'learning_rate': 0.2, 'min_child_weight': 2, 'n_estimators': 200}
  Exactitud en el conjunto de prueba: 0.8721590909090909

  En la **Figura 4.1.10** se muestra la evaluación del rendimiento del modelo a través de la Matriz de Confusión.

Figura 4.1.10



Métricas de SKLEARN obtenidas:
Exactitud (Accuracy): 0.8721590909090909
Precisión (Precision): 0.8716432219705289
Recall (Sensibilidad): 0.8721590909090909
F1-score: 0.8710214728753962

Métricas de Validación cruzada (Cross-Validation) con la técnica K-Fold
Scores de validación cruzada: [0.91489362 0.86018957 0.88625592 0.87440758 0.88625592]
Media de los scores: 0.8844005243521226
Desviación estándar de los scores: 0.01801523099333119
Exactitud promedio: 0.8844005243521226

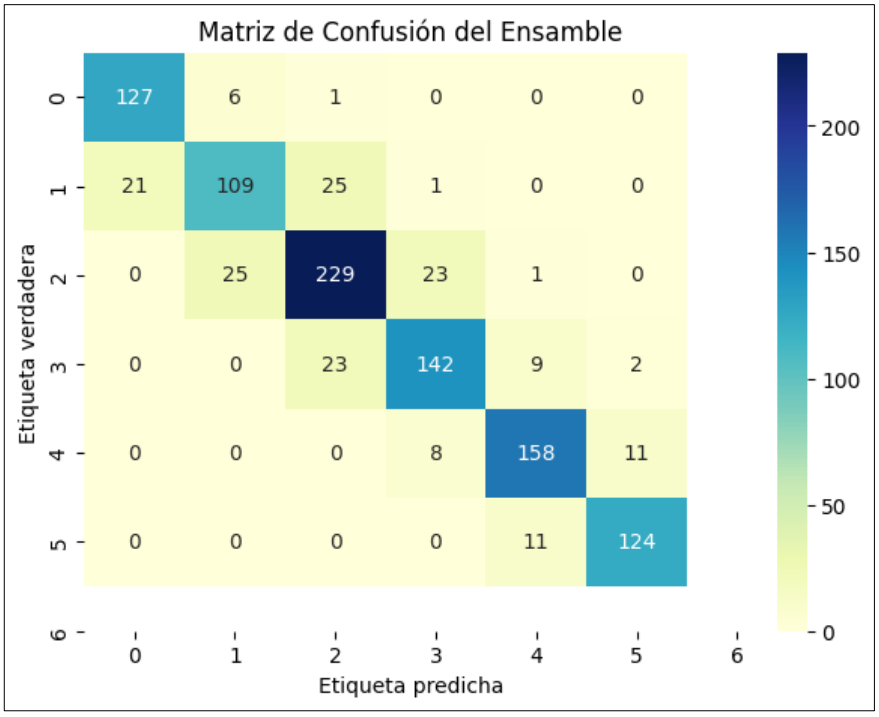
4.2. Presentación e Interpretación de los Resultados del Ensamble de Modelos

- **Ensamble por Votación de la mayoría**

  Se realiza un ensamble simple conocido como "Votación de la mayoría", combinando las predicciones de tres modelos base: KNN, DecisionTreeClassifier, y XGBClassifier (XGBoost) y asignando un voto a la clase predicha por cada modelo. Luego, la clase que recibe la mayoría de votos se considera la predicción final del ensamble.

  En la **Figura 4.2.1** se muestra la evaluación del rendimiento del modelo a través de la Matriz de Confusión.

Figura 4.2.1



Métricas de SKLEARN obtenidas:
Exactitud (Accuracy): 0.8683712121212122
Precisión (Precision): 0.8671096493357582
Recall(Sensibilidad): 0.8683712121212122
F1-score: 0.8661601320024686

Se optó por utilizar un ensamble de tres modelos base: KNN, DecisionTreeClassifier, y XGBClassifier (XGBoost), ya que este enfoque proporcionó valores equilibrados en las métricas de evaluación. Con una exactitud del 86.83%, una precisión del 87.71%, un recall del 86.83%, y un F1-score del 86.61%, el ensamble demostró un rendimiento sólido al lograr un equilibrio entre la capacidad de clasificar correctamente las muestras positivas y negativas.

Exactitud (Accuracy): La exactitud mide la proporción de predicciones correctas en el total de predicciones. Es una métrica común para evaluar el rendimiento global del modelo. En este caso, el modelo logra una exactitud del 86.83%, lo que significa que el 86.83% de las predicciones son correctas.

Precisión (Precision): La precisión mide la proporción de verdaderos positivos (predicciones correctas) entre todas las predicciones positivas. En otras palabras, es la capacidad del modelo para predecir correctamente los positivos. En este caso, el modelo tiene una precisión del 86.71%, lo que indica que el 86.71% de las predicciones positivas son correctas.

Recall (Recuperación o Sensibilidad): El recall mide la proporción de verdaderos positivos entre todas las instancias reales de una clase. Es importante cuando el coste de los falsos negativos es alto, ya que se centra en la capacidad del modelo para encontrar y capturar los positivos. El recall es del 86.83%, lo que significa que el modelo captura el 86.83% de las instancias positivas.

F1-score: El F1-score es la media armónica de precisión y recall. Combina la capacidad del modelo para predecir correctamente los positivos y para capturar los positivos en un solo valor. En este caso, el F1-score es del 86.61%, lo que indica un equilibrio entre precisión y recall.

Métricas de Validación cruzada (Cross-Validation) con la técnica K-Fold
Scores de validación cruzada: [0.74231678 0.91706161 0.91706161 0.87677725 0.77488152]
Media de los scores: 0.8456197550782605
Desviación estándar de los scores: 0.0732858366457176
Exactitud promedio: 0.8456197550782605

En el contexto del ensamble por Votación de la mayoría y las métricas de validación cruzada con la técnica K-Fold, se puede realizar la siguiente interpretación de los valores:

- La exactitud promedio del ensamble por Votación de la mayoría es del 84.56%, lo que sugiere un rendimiento promedio en la clasificación.
- La desviación estándar es relativamente alta (0.0733), indicando que puede haber variabilidad en el rendimiento del modelo entre los diferentes pliegues.
- Los scores individuales proporcionan información detallada sobre el rendimiento en cada fold, lo que puede ser útil para identificar posibles variaciones en el rendimiento del modelo en diferentes conjuntos de datos.

En resumen, estos resultados indican que el ensamble por Votación de la mayoría tiene un rendimiento promedio en la clasificación, pero hay cierta variabilidad en el rendimiento entre los diferentes pliegues de la validación cruzada. Puede ser útil investigar más a fondo para comprender las razones detrás de esta variabilidad y posiblemente ajustar el modelo o considerar otros enfoques.

- **Ensamble apilado**

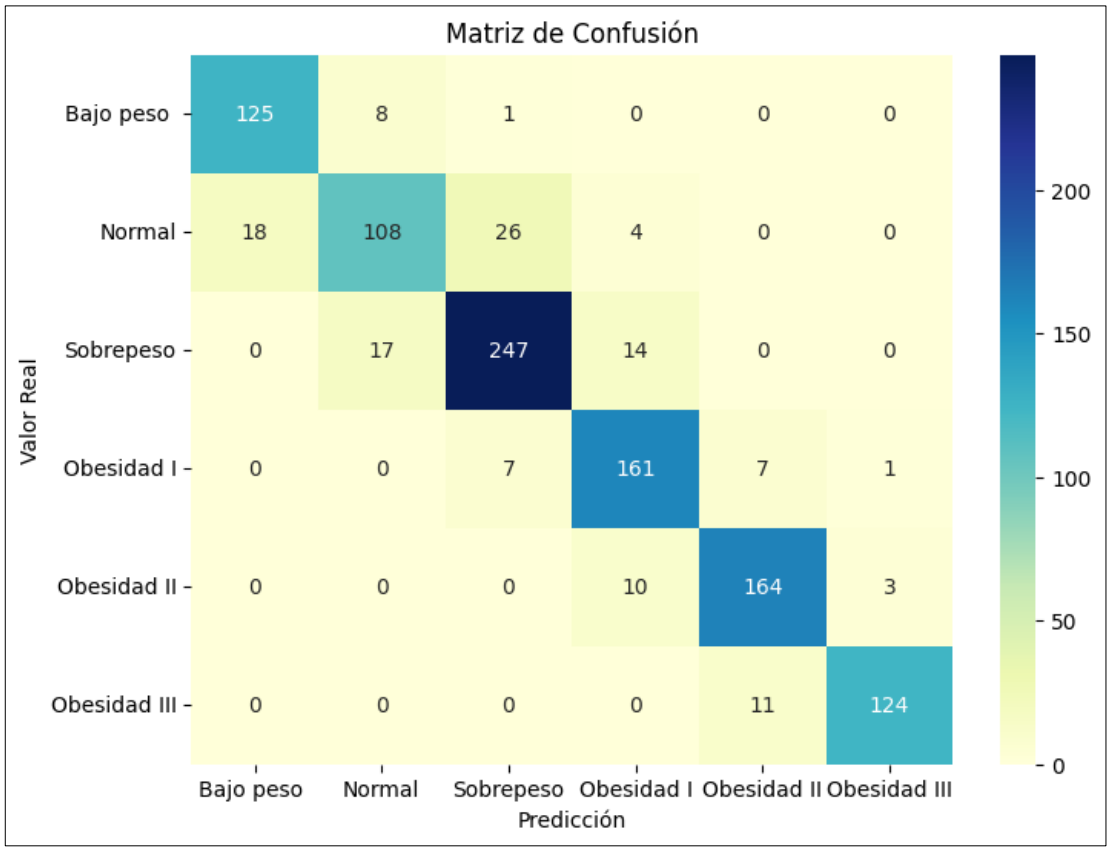
  Para este ensamble hemos definido tres modelos base (KNN, Árbol de Decisión y XGBoost) y un modelo meta (Regresión Logística) que se utiliza para combinar las predicciones de los modelos base.

  Estos tres modelos ofrecen enfoques y características únicas, lo que aumenta la capacidad del ensamble para capturar patrones complejos y mejorar la generalización del modelo.

  Al utilizar la Regresión Logística como modelo meta, se busca lograr una síntesis inteligente de las predicciones individuales de los modelos base. Este enfoque permite aprovechar las fortalezas de cada modelo base y compensar sus debilidades, generando así un ensamble que exhibe un rendimiento robusto y generalizable en la tarea de predicción de niveles de obesidad.

  En la **Figura 4.2.2** se muestra la evaluación del rendimiento del modelo a través de la Matriz de Confusión.

Figura 4.2.2



Métricas de SKLEARN obtenidas:
Exactitud (Accuracy): 0.8797348484848485
Precisión (Precision): 0.8791422628849103
Recall(Sensibilidad): 0.8797348484848485
F1-score: 0.8783051264800746

Se optó por utilizar un ensamble de tres modelos base: KNN, DecisionTreeClassifier, y XGBClassifier (XGBoost), ya que este enfoque proporcionó valores equilibrados en las métricas de evaluación. Con una exactitud del 87.98%, una precisión del 87.91%, un recall del 87.97%, y un F1-score del 87.83%, el ensamble demostró un rendimiento sólido al lograr un equilibrio entre la capacidad de clasificar correctamente las muestras positivas y negativas.

Exactitud (Accuracy): La exactitud mide la proporción de predicciones correctas en el total de predicciones. Es una métrica común para evaluar el rendimiento global del modelo. En este caso, el modelo logra una exactitud del 87.98%, lo que significa que el 87.98% de las predicciones son correctas.

Precisión (Precision): La precisión mide la proporción de verdaderos positivos (predicciones correctas) entre todas las predicciones positivas. En otras palabras, es la capacidad del modelo para predecir correctamente los positivos. En este caso, el modelo tiene una precisión del 87.91%, lo que indica que el 87.91% de las predicciones positivas son correctas.

Recall (Recuperación o Sensibilidad): El recall mide la proporción de verdaderos positivos entre todas las instancias reales de una clase. Es importante cuando el coste de los falsos negativos es alto, ya que se centra en la capacidad del modelo para encontrar y capturar los positivos. El recall es del 87.97%, lo que significa que el modelo captura el 87.97% de las instancias positivas.

F1-score: El F1-score es la media armónica de precisión y recall. Combina la capacidad del modelo para predecir correctamente los positivos y para capturar los positivos en un solo valor. En este caso, el F1-score es del 87.83%, lo que indica un equilibrio entre precisión y recall.

Métricas de Validación cruzada (Cross-Validation) con la técnica K-Fold
Scores de validación cruzada para ensemble apilado: [0.90307329 0.89810427 0.8957346 0.91232227 0.87203791]
Media de los scores: 0.8962544676369422
Desviación estándar de los scores: 0.013377624362013429
Exactitud promedio: 0.8962544676369422

En el contexto de la validación cruzada con la técnica K-Fold y las métricas específicas proporcionadas se puede realizar la siguiente interpretación de los valores:
La media de los scores indica el rendimiento promedio del ensamble apilado a lo largo de todos los pliegues. En este caso, es la media de la exactitud en los diferentes folds. La exactitud promedio es simplemente otra forma de expresar la media de los scores y representa la precisión promedio del modelo sobre todos los pliegues de la validación cruzada. La exactitud promedio del ensamble apilado es del 89.63%, lo que sugiere un buen rendimiento promedio en la clasificación.
La desviación estándar proporciona una medida de la variabilidad de los scores entre los diferentes pliegues. En este caso, es la desviación estándar de la exactitud. La

desviación estándar es relativamente baja (0.0134), indicando que el rendimiento es consistente entre los diferentes pliegues.

Los scores individuales proporcionan información detallada sobre el rendimiento en cada fold, lo que puede ser útil para identificar posibles variaciones en el rendimiento del modelo en diferentes conjuntos de datos. Cada número representa la métrica de evaluación (posiblemente la exactitud) para un fold específico.

En resumen, estos resultados sugieren que el ensamble apilado es efectivo y generaliza bien en diferentes subconjuntos de datos.

- **Ensamble apilado: Random Forest y Gradient Boosting**

  En esta sección, implementamos un ensamble avanzado utilizando la técnica de apilamiento (stacking). El apilamiento combina las predicciones de varios modelos base más avanzados, en este caso, Random Forest y Gradient Boosting, utilizando un modelo meta (también Gradient Boosting en este ejemplo).

  El propósito de utilizar un ensamble avanzado es capitalizar las fortalezas individuales de los modelos base, mejorando así la capacidad de generalización y rendimiento predictivo del modelo global. Los modelos base, RandomForest y Gradient Boosting, son conocidos por su capacidad para manejar relaciones complejas en los datos y capturar patrones no lineales.

  El modelo meta (también conocido como clasificador final) se entrena para combinar las salidas de los modelos base y ajustar el ensamble para mejorar aún más la precisión. La elección de Gradient Boosting como modelo meta se basa en su capacidad para capturar patrones y relaciones complejas en los datos, lo que lo hace adecuado para aprender de las predicciones de los modelos base.

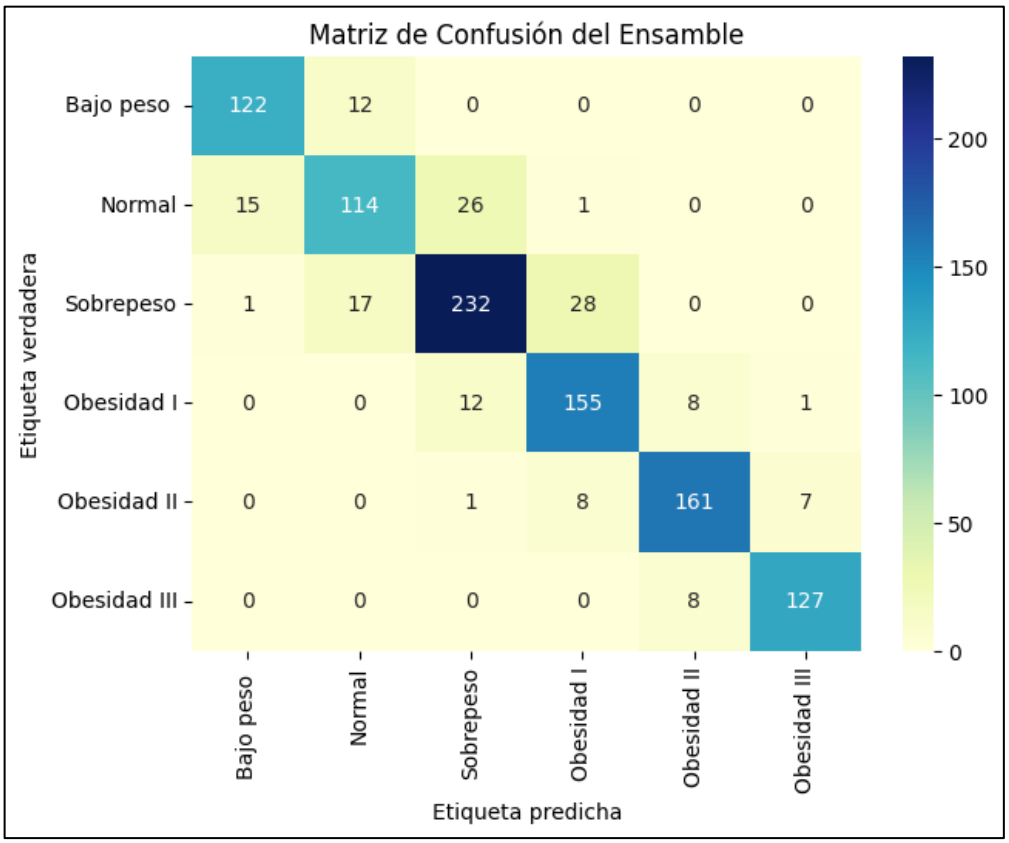
  Después de entrenar el ensamble apilado, evaluamos su rendimiento en un conjunto de prueba y realizamos una validación cruzada para garantizar su robustez y capacidad de generalización. Los resultados de la validación cruzada, junto con la exactitud promedio, nos proporcionan métricas objetivas para comparar y seleccionar el modelo final.

Esta estrategia de ensamble avanzado puede ser particularmente efectiva cuando los modelos base son diversos y pueden abordar diferentes aspectos del problema. Además, el ajuste cuidadoso de los hiperparámetros de los modelos base y del modelo meta puede contribuir a optimizar el rendimiento del ensamble.

Sintetizando, la implementación de este ensamble avanzado busca mejorar la calidad predictiva del modelo combinando las fortalezas de modelos más avanzados, proporcionando así una solución más robusta y precisa para nuestro problema específico.

En la **Figura 4.2.3** se muestra la evaluación del rendimiento del modelo a través de la Matriz de Confusión.

Figura 4.2.3



Métricas de SKLEARN obtenidas:
Exactitud (Accuracy): 0.8626893939393939
Precisión (Precision): 0.8625978766468855
Recall (Sensibilidad): 0.8626893939393939
F1-score: 0.8621035788862871

En general, las métricas son bastante consistentes entre sí, lo que indica un buen rendimiento general del ensamble apilado de Random Forest y Gradient Boosting.

Exactitud (Accuracy): La exactitud mide la proporción de predicciones correctas en relación con el total de predicciones. En este caso, alrededor del 86.26% de las predicciones del modelo son correctas. La alta exactitud sugiere que el modelo está haciendo predicciones correctas en una proporción significativa de casos.

Precisión (Precision): La precisión se refiere a la proporción de predicciones positivas correctas entre todas las predicciones positivas. En este caso, alrededor del 86.25% de las predicciones positivas del modelo son correctas.

Recall (Sensibilidad): Recall (también conocido como sensibilidad o tasa positiva real) mide la proporción de instancias positivas que fueron correctamente identificadas por el modelo. En este caso, alrededor del 86.26% de las instancias positivas fueron correctamente identificadas.
La precisión y el recall son bastante similares, indicando que el modelo es capaz de identificar positivos de manera precisa y recuperar la mayoría de los verdaderos positivos.

F1-score: El F1-score es la media armónica entre precisión y recall. Es útil cuando se busca un equilibrio entre ambas métricas. Un F1-score del 86.21% sugiere un equilibrio entre precisión y recall, indicando un rendimiento general sólido del modelo.

Métricas de Validación cruzada (Cross-Validation) con la técnica K-Fold
Scores de validación cruzada: [0.68321513 0.92890995 0.93364929 0.8957346 0.77251185]
Media de los scores: 0.8428041634454864
Desviación estándar de los scores: 0.09887198161888174
Exactitud promedio: 0.8428041634454864

En el contexto de la validación cruzada con la técnica K-Fold para un ensamble apilado de Random Forest y Gradient Boosting, se puede realizar la siguiente interpretación de los valores:

La media de los scores sugiere que, en promedio, el modelo logra una exactitud del 84.28% en los datos de validación. Este valor proporciona una indicación general del rendimiento del modelo. Cuanto más cercano a 1, mejor es el rendimiento del modelo en términos de exactitud.

La desviación estándar mide la variabilidad o dispersión de los scores. En este caso, la desviación estándar es 0.09887198161888174. Una desviación estándar baja indica consistencia en el rendimiento entre los pliegues.

Los scores individuales en los pliegues varían, lo que puede deberse a diferencias en las características de los conjuntos de validación.

En resumen, el modelo tiende a ser preciso, pero hay cierta variabilidad en su rendimiento en diferentes conjuntos de datos de validación. Estas métricas son útiles para evaluar la estabilidad y la robustez del modelo a través de la validación cruzada.

5. DISCUSIÓN

5.1. Comparación con objetivos

La evaluación de la comparación con los objetivos revela un cumplimiento sustancial de los objetivos planteados en nuestro análisis exploratorio de datos y modelado predictivo. La implementación de visualizaciones interactivas, como los gráficos de dispersión y las distribuciones segmentadas por género, proporciona una experiencia dinámica para explorar las relaciones entre variables clave. Además, la incorporación de controles deslizantes y menús desplegables permite una personalización eficiente de los análisis.

En términos de modelado predictivo, la presentación de resultados incluye matrices de confusión y métricas de evaluación para cada modelo y ensamble, brindando una visión completa del rendimiento. La interactividad se mantiene, permitiendo a los usuarios ajustar parámetros y explorar cómo afectan las predicciones.

En general, la implementación demuestra un enfoque efectivo para lograr los objetivos establecidos, proporcionando herramientas visuales intuitivas y modelos predictivos sólidos que respaldan el análisis y la toma de decisiones.

5.2. Desafíos y Limitaciones

Durante el desarrollo de este proyecto, nos enfrentamos a varios desafíos y reconocemos ciertas limitaciones que afectaron la ejecución y los resultados obtenidos. Uno de los principales desafíos fue la disponibilidad y calidad de los datos, ya que, en algunos casos, la falta de información completa podría haber afectado la precisión de nuestros modelos predictivos y visualizaciones.

Uno de los desafíos significativos que enfrentamos durante el desarrollo del proyecto fue la demora en la ejecución del programa. La complejidad inherente de los modelos predictivos y la extensión del conjunto de datos contribuyeron a tiempos de ejecución más largos de lo ideal. Este factor puede afectar la eficiencia y la velocidad de respuesta, especialmente al interactuar con visualizaciones interactivas.

La demora en la ejecución plantea consideraciones importantes en entornos donde el tiempo de procesamiento es crítico. Aunque hemos optimizado el código en la medida de lo posible, es esencial tener en cuenta este desafío al planificar el uso y la implementación del proyecto en entornos con restricciones temporales. Este factor también puede influir en la experiencia del usuario al interactuar con la aplicación.*

Además, la interpretación de las relaciones causales entre variables puede ser un desafío, ya que los modelos predictivos y las visualizaciones basadas en correlaciones no garantizan causalidad. Es importante tener en cuenta estas limitaciones al interpretar los resultados y tomar decisiones basadas en el análisis.

Otro desafío radica en la complejidad inherente de los modelos predictivos utilizados. Aunque hemos implementado modelos robustos, existe el riesgo de sobreajuste y la necesidad de una validación cuidadosa en entornos del mundo real.

En resumen, aunque hemos abordado estos desafíos y limitaciones en la medida de lo posible, es crucial tener en cuenta estos aspectos al interpretar los resultados y considerar futuras mejoras y refinamientos en el proyecto

6. CONCLUCIONES

6.1. Reflexiones Finales

A lo largo de este proyecto de análisis y desarrollo centrado en la obesidad, hemos alcanzado una comprensión profunda de los factores clave que influyen en esta condición de salud crítica. Reflexionando sobre los resultados y hallazgos más destacados, se destacan varias conclusiones fundamentales:

Importancia del Género:

Nuestro análisis revela una relación significativa entre el género y los niveles de obesidad. Las mujeres, en su mayoría, ocupan posiciones más elevadas en varios factores evaluados, subrayando la necesidad de abordar cuestiones específicas de género en futuras intervenciones y estrategias de salud.

Visualizaciones Interactivas:

La implementación de visualizaciones interactivas, como gráficos de dispersión y distribuciones segmentadas por género, ha mejorado drásticamente la experiencia de exploración de datos. Estas herramientas no solo facilitan la comprensión de las relaciones entre variables, sino que también permiten una personalización eficiente de los análisis, brindando un enfoque dinámico para la toma de decisiones informadas.

Modelado Predictivo y Factores Relacionados con la Obesidad:

Los modelos predictivos desarrollados proporcionan información valiosa para anticipar niveles de obesidad. Es esencial reconocer las limitaciones inherentes a estos modelos, como el riesgo de sobreajuste y la importancia de una validación cuidadosa en entornos del mundo real.

En nuestro proyecto, identificamos las variables más influyentes relacionadas con el desarrollo de la obesidad. Estas variables clave incluyen: Peso, Edad, Actividad Física, Antecedentes Familiares, Consumo de Agua, Uso de Tecnología, Consumo entre Comidas y Consumo Calórico.

Los algoritmos de machine learning que arrojaron los resultados más destacados fueron KNN, Árbol de Decisión y XCBoost. Estos modelos se utilizaron para construir dos tipos de ensambles: un ensamble por votación y un ensamble apilado (stacking). También se aplicó un ensamble apilado de Random Forest y Gradient Boosting. De manera notable, el ensamble apilado de KNN, Árbol de Decisión y XCBoost demostró ser el más efectivo, exhibiendo métricas superiores en comparación con otras configuraciones.

Es importante destacar que, si bien estos modelos ofrecen una perspectiva valiosa, su implementación práctica debe considerar cuidadosamente las condiciones del entorno real y las posibles limitaciones. El entendimiento profundo de los factores clave relacionados con la obesidad permite una toma de decisiones más informada en el desarrollo de estrategias preventivas y de intervención.

Desafíos y Limitaciones:

Hemos enfrentado desafíos notables, desde la disponibilidad de datos hasta la demora en la ejecución del programa. Estos desafíos subrayan la necesidad continua de mejorar y refinar el proyecto, así como la importancia de considerar estas limitaciones al interpretar los resultados.

En última instancia, este proyecto no solo avanza en nuestra comprensión de la obesidad, sino que también establece una base sólida para futuras investigaciones y mejoras. La combinación de análisis exploratorio, visualizaciones interactivas y modelado predictivo ofrece una herramienta integral para abordar y prevenir la obesidad, contribuyendo a la gestión efectiva de la salud en la sociedad.

6.2. Aplicación de Conocimientos

La realización de este proyecto ofreció una valiosa oportunidad para aplicar los conceptos fundamentales de big data y machine learning aprendidos durante el curso. A continuación, se destacan las principales aplicaciones de estos conocimientos en el desarrollo y análisis del proyecto sobre obesidad:

Manejo de Grandes Conjuntos de Datos:

El proyecto implicó trabajar con un conjunto de datos extenso y complejo. Los conceptos de big data aprendidos en el curso fueron esenciales para abordar eficientemente la importación, manipulación y análisis de grandes volúmenes de datos. La habilidad para estructurar y limpiar datos a esta escala fue fundamental para obtener resultados precisos.

Transformación de Datos para Machine Learning:
La transformación de datos, incluida la conversión de variables categóricas a numéricas, fue una fase crucial para preparar el conjunto de datos para el modelado predictivo. La comprensión de cómo los algoritmos de machine learning interpretan diferentes tipos de datos fue esencial para optimizar la entrada de datos y mejorar la efectividad de los modelos.

Análisis Exploratorio y Visualización Avanzada:
La aplicación de técnicas de big data permitió realizar un análisis exploratorio profundo del conjunto de datos. La capacidad para identificar patrones, correlaciones y distribuciones a través de visualizaciones avanzadas mejoró significativamente gracias a los conceptos aprendidos sobre manejo de grandes conjuntos de datos.

Desarrollo de Modelos Predictivos:

La implementación de modelos predictivos para predecir niveles de obesidad se benefició directamente de los conceptos de machine learning. La comprensión de algoritmos como regresión logística y ensambles de modelos contribuyó a la creación de herramientas predictivas sólidas, fundamentales para el éxito del proyecto.

La aplicación práctica de los conceptos de big data y machine learning fue integral en cada etapa del proyecto. Desde la importación y limpieza de datos hasta el análisis exploratorio y el desarrollo de modelos predictivos, estos conocimientos fueron la columna vertebral que permitió abordar los desafíos del mundo real asociados con conjuntos de datos complejos y la toma de decisiones basada en modelos predictivos avanzados.

6.3. Sugerencias para Futuras Investigaciones

El proyecto sobre obesidad ha proporcionado una base sólida, pero existen oportunidades para expandir y mejorar el trabajo realizado. A continuación, se presentan sugerencias para futuras investigaciones que podrían enriquecer y ampliar la comprensión de los factores asociados a la obesidad:

- **Integración de Datos Adicionales:**
  Considerar la incorporación de conjuntos de datos adicionales relacionados con la salud, como información genética, datos clínicos más detallados y resultados de pruebas específicas. Esto podría ofrecer una visión más completa de los factores que contribuyen a la obesidad y mejorar la precisión de los modelos predictivos.

- **Validación Externa de Modelos Predictivos:**
  Realizar validaciones externas de los modelos predictivos en diferentes poblaciones o conjuntos de datos. Esto ayudaría a evaluar la generalización de los modelos más allá de los datos utilizados en este proyecto y fortalecer la confianza en la capacidad predictiva en diversos contextos.

- **Exploración de Causalidad:**
  Profundizar en la investigación de relaciones causales entre variables identificadas en el análisis exploratorio. Aplicar métodos estadísticos o experimentos controlados para comprender mejor la naturaleza de estas relaciones y ofrecer información más precisa sobre los factores que influyen en la obesidad.

- **Explorar ensambles con modelos más avanzados:**
  Profundizar en la exploración de ensambles que incorporen modelos más avanzados, como redes neuronales o modelos de aprendizaje profundo. Estos modelos pueden ofrecer una capacidad de aprendizaje más compleja y capturar patrones sutiles en conjuntos de datos complejos.

7. CÓDIGO Y DOCUMENTACIÓN

7.1. Enlace a Google Colab

https://github.com/msaramayo/Trabajo-Final/blob/main/trabajoFinalGrupo22_301223.ipynb

8. REFERENCIAS

Amazon Web Services. (n.d.). *https://docs.aws.amazon.com/es_es/sagemaker/latest/dg/xgboost.html.*

Aprende Machine Learning. (n.d.). *https://www.aprendemachinelearning.com/arbol-de-decision-en-python-clasificacion-y-prediccion/.*

Aprende Machine Learning. (n.d.). *https://www.aprendemachinelearning.com/clasificar-con-k-nearest-neighbor-ejemplo-en-python/.*

AprendeIA. (n.d.). *https://aprendeia.com/metodos-de-ensamble-de-modelos-machine-learning-ensemble-methods-en-espanol/#:~:text=El%20ensamblado%20es%20el%20arte,t%C3%A9cnica%20de%20modelado%20diferente%20utilizada.*

Artificial Intelligence in Plain English. (n.d.). *https://ai.plainenglish.io/naive-bayes-classifier-achieving-100-accuracy-on-iris-dataset-d6df3e927096.*

DataScientest. (n.d.). *https://datascientest.com/es/cross-validation-definicion-e-importancia.*

DataScientest. (n.d.). *https://datascientest.com/es/que-es-la-regresion-logistica.*

IMC Smart Education. (n.d.). *https://blogs.imf-formacion.com/blog/tecnologia/matriz-confusion-como-interpretarla-202106/.*

Medium. (n.d.). *https://medium.com/@nicolasarrioja/gu%C3%ADa-definitiva-a-las-t%C3%A9cnicas-de-ensemble-7a4bb1203bcb.*

Medium. (n.d.). *https://medium.com/analytics-vidhya/introduction-to-svm-and-kernel-trick-part-1-theory-d990e2872ace.*

Scikit-learn. (n.d.). *https://scikit-learn.org/stable/modules/classes.html#module-sklearn.metrics.*

Scikit-learn. (n.d.). *https://scikit-learn.org/stable/modules/grid_search.html.*

UTEC. (n.d.). *Universidad Tecnológica*. Retrieved Diciembre 27, 2023, from https://aichallenge.utec.edu.uy/community/postid/124/

World Health Organization. (n.d.). *https://www.who.int/europe/news-room/fact-sheets/item/a-healthy-lifestyle---who-recommendations.*