PROYECTO MACHINE LEARNING I

PREDICCIÓN DE OBESIDAD A PARTIR DE HÁBITOS DE VIDA

DOCENTE

CARLOS ISAAC ZAINEA MAYA

ESTUDIANTES

Báez Bermúdez, Cristian David

Mariño Florez, David José

Rodriguez Simmonds, Santiago

Rodríguez Díaz, Germán Alonso

Universidad EAN

2024

RESUMEN EJECUTIVO

INTRODUCCIÓN

En el presente proyecto abordaremos un problema físico común en la vida cotidiana de muchas personas: la obesidad. Para ello, utilizaremos una base de datos de un repositorio de acceso libre llamada “ObesityDataSet_raw_and_data_sinthetic.csv”, la cual contiene 2,111 registros y 17 columnas.

El objetivo de este ejercicio es, en primer lugar, realizar las limpiezas y transformaciones necesarias y aplicar técnicas de aprendizaje no supervisado para realizar agrupación, asociación y reducción de dimensionalidad de los datos, con el fin de descubrir patrones y estructuras ocultas sin el uso de etiquetas (como la variable "NObeyesdad"). En segundo lugar, utilizaremos aprendizaje supervisado para entrenar nuevamente el modelo, tanto con los resultados obtenidos en la etapa no supervisada como sin ellos. Esto nos permitirá evaluar la eficiencia y confiabilidad del modelo para predecir valores futuros y analizar los datos y atributos que puedan proporcionar una relación significativa con el tipo de obesidad.

METODOLOGÍA

Para la metodología se aplicarán los siguientes pasos:

1. Carga del Dataset y librerias:

En este caso para importar la información de la base de datos se utilizó el repositorio de conjunto de datos (ucimlrepo) para acceder al conjunto de datos del repositorio UCI de Machine Learning y mediante la función fetch_ucirepo poder descargarlo.

Librerías que se utilizaron en el proyecto:
Pandas
numpy
matplotlib
seaborn
sklearn
xgboost
shap

2. Exploración de los datos:

En este paso se realiza una exploración de la información que presenta el dataset, detallando las variables feature y la variable objetivo target.

En la dinámica del ejercicio no se tomará las variables de Altura y Peso del dataset debido a que representan una alta multicolinealidad con el Indice de Masa Corporal que abarca la problemática para clasificar el nivel de obesidad de la persona.

3. Análisis de variables:

Se identifican las variables numéricas y cualitativas. Las numéricas se escalan usando StandardScaler para tener un mejor rendimiento en el análisis posterior de agrupamiento.

4. Aprendizaje No Supervisado

4.1. Definir Clusteres

Para conocer el numero optimo de Clusteres se utilizó dos métodos (Método del codo y método de la silueta), estos dos enfoques permiten evaluar la calidad del agrupamiento y determinar cuál es el valor K para que el método de agrupamiento sea lo más efectivo posible:

4.1.1. Método del Codo

En el método del codo se observó que la inercia cerca de los 8000 concluye que 4 es el numero optimo de clusteres a utilizar, debido a que después de observar k=5 o k= 6, la gráfica tiende a estabilizarse un poco, sin embargo, no es lo suficientemente pronunciada como para asegurar un correcto uso de k=4, para este ejercicio indica que los puntos están más cerca a los centroides en este punto de inercia.

4.1.2. Método de la silueta

En el método de la silueta se observó que efectivamente el valor k = 4 es el mejor para encontrar el numero optimo de clusteres a utilizar en el ejercicio, adicional para asegurarnos de que tuviéramos la mejor calidad de agrupamiento se utilizó el coeficiente de silueta promedio, el cual permite identificar que tan bien está formado el cluster y que tan separados están de los demás, indicándonos un coeficiente de k=4 -> 0.19748266026343453, muy próximo a, k=9-> 0.19836586530640915.

4.2. Reducción de dimensionalidad

Dado que el conjunto de datos tiene varias características, se aplica reducción de dimensionalidad para visualizar los datos en 2D:

PCA (Análisis de Componentes Principales) Reducción de dimensionalidad lineal:

En este método de reducción se visualiza que hay una distribución compacta alrededor del centro con algún que otro dato disperso.

t-SNE (t-Distributed Stochastic Neighbor Embedding): Reducción no lineal más eficaz para visualización:

En el metofo de t-sne se observa que los puntos estan organizados en grupos más dispersos y definidos, por lo cual indica que es el mejor método de los dos presentados para tomar relaciones locales en los datos.

4.3. Visualización de Clusters en PCA y T-SNE

Tras obtener los clusters, usando K-means se representan los resultados en un espacio reducido utilizando PCA y t-SNE, esto permite observar cómo se distribuyen los clusters (k=4) en ambos tipos de reducción. Posteriormente, se valida como la variable target se distribuye en cada uno de los 4 clusteres a partir de los resultados t-sne obtenidos en el paso anterior en ambos planos dimensionales (X_tsne, Y_tsne).

Adicionalmente, se incluye una matriz de correlación y diagramas de cajas (boxplots) para observar la distribución de las variables cuantitativas en cada cluster su correlación.

Por último, mediante un Histograma se observa la distribución de cada uno de los niveles de obesidad en cada cluster, observando que:

el cuarto cluster destaca por una alta prevalencia de individuos con obesity_type_III (obesidad), lo cual indica que este grupo está compuesto predominantemente por personas con un alto índice de obesidad. Este hallazgo sugiere que el cluster 4 podría representar a un segmento de la población que enfrenta mayores riesgos de salud relacionados con la obesidad grave, y su distribución en cuanto a las variables como edad, consumo de alimentos rápidos (FCVC), y actividad física (FAF) podría ser muy diferente a la de los otros clusters.

5. Aprendizaje Supervisado

5.1. Planteamiento Modelo Supervisado:

Se hace un análisis exploratorio de los datos, de tal manera revisar los tipos datos que tenemos. Identificamos variables numéricas con decimales y procedemos a redondearlas a enteros y cambiar los tipos de datos que identificamos a categóricos.

5.2. Codificación y escalamiento de las variables

De las variables que identificamos en la anterior parte del proceso, procedemos a codificar con Label Encoder, concatenamos las columnas y escalamos la variable que tiene el valor más grande.

5.3. Entrenamiento Modelos Supervisados

Se dividen los datos en conjunto de entrenamiento y conjunto de prueba para evitar el sobreajuste. Se prueban varios modelos de clasificación, incluyendo Random Forest, XGBoost, SVM, y Gradient Boosting. Se utiliza validación cruzada para evaluar la precisión de cada modelo.

5.4. Optimización, evaluación y visualización de Resultados

Para el mejor modelo (XGBoost), se realiza una optimización de los hiperparámetros usando GridSearchCV. Después de entrenar el modelo, se realiza una evaluación utilizando matrices de confusión, reportes de clasificación y curvas ROC para cada clase.

5.5. Shap como métrica de validación

Para entender las carácteristicas que afectan mayormente el modelo, diseñamos un shap que contiene 3 vistas, en esta podemos observar cada una de las métricas y su grado de "afectación" al modelo.

RESULTADOS

El análisis muestra que el modelo que incluye clusters como una característica adicional obtiene mejores métricas en precisión y recall comparado con el modelo sin clusters. Los clusters identificados presentan patrones claros de comportamiento, lo que sugiere que representan segmentos significativos de clientes con características comunes. A continuación, presentamos visualizaciones clave:

- **Distribución de Clusters**: Se visualiza cómo se agrupan los diferentes clientes en cada cluster.
- **Desempeño de Modelos**: Gráficos de barras comparativos de precisión, recall y F1-score entre el modelo con y sin clusters.
- **Visualización de Patrones de Clusters**: Gráficas que muestran las características distintivas de cada cluster.

CONCLUSIONES Y RECOMENDACIONES

Conclusiones:

1. **Impacto limitado de la segmentación en el rendimiento del modelo**: Los resultados de las métricas (precisión, recall y f1-score) indican que el modelo sin

clusters y el modelo con clusters tienen un rendimiento casi idéntico. Esto sugiere que, en este caso, la segmentación no aporta un valor significativo al modelo predictivo.

2. **Utilidad potencial de la segmentación en otras aplicaciones**: Aunque en este contexto específico la segmentación no mejoró las métricas de rendimiento, sigue siendo una técnica útil en aplicaciones donde la variabilidad dentro de los grupos es más marcada. Por ejemplo, en modelos donde los clientes tienen comportamientos muy distintos según su segmento, el uso de clusters podría aportar mejoras.

Recomendaciones:

1. **Evitar la inclusión de clusters en el modelo si no mejora el rendimiento**: Dado que la adición de la característica de clusters no mejora las métricas de rendimiento de manera significativa, se recomienda simplificar el modelo y **no incluir los clusters** como variable en futuros entrenamientos, a menos que se observe una ganancia clara en otros contextos o tipos de datos.
2. **Experimentar con otros métodos de segmentación**: En proyectos futuros, se podrían probar otros métodos de clustering, como **DBSCAN** o **clustering jerárquico**, que podrían capturar relaciones diferentes en los datos y quizá mejorar el rendimiento. Además, ajustar el número de clusters o utilizar métodos de segmentación basados en otras técnicas de reducción de dimensionalidad podría cambiar los resultados.
3. **Realizar una prueba de validación adicional para verificar el impacto de la segmentación**: Si en otro contexto se vuelve a considerar la segmentación, se recomienda realizar una prueba de validación inicial (como se hizo aquí) para comprobar si realmente aporta valor al modelo antes de implementarla completamente.

REFERENCIAS

- Scikit-Learn: Herramienta para el preprocesamiento de datos, clustering y modelado supervisado.
- Pandas y Matplotlib: Utilizados para el análisis exploratorio de datos y visualización de resultados.
- Algoritmos de K-means y Árboles de Decisión: Referencias de implementación en documentación de Python.