Machine Learning
Proyecto 3:Clustering

Autores

Melanie Alexia Cortez Rojas
Alex Jose Loja Zumaeta
Tania Araceli Barreda Galvez
Melisa Karen Rivera Alagon
Andre Sebastian Segovia Melgarejo
Jean Paul Ycarrayme Valdivia

Docente

Rensso Victor Hugo Mora Colque

Fecha

3 de noviembre del 2024

1 Resumen

Este informe presenta un análisis de clustering aplicado al conjunto de datos "Heart Disease" con el fin de segmentar a pacientes según características clínicas. Se implementaron diversos algoritmos de clustering, incluidos KMeans, Gaussian Mixture Model, MeanShift y DBSCAN, para identificar patrones no supervisados en los datos. El proceso abarcó la normalización de las variables, la evaluación del número óptimo de clusters y la visualización de resultados. Los hallazgos permitieron identificar perfiles de pacientes asociados a riesgos cardiovasculares, proporcionando una base potencial para estrategias de prevención y diagnóstico en el contexto de la salud cardiovascular.

Palabras clave: Clustering, Heart Disease Dataset, KMeans, Gaussian Mixture Model, MeanShift, DBSCAN, Segmentación de pacientes, Análisis exploratorio de datos, Factores de riesgo cardiovascular, Visualización de clusters.

2 Introducción

En este estudio, se implementan técnicas de clustering para identificar segmentos de pacientes en el conjunto de datos "Heart Disease" disponible en Kaggle. El objetivo principal es explorar métodos de agrupamiento no supervisado para descubrir patrones relacionados con el riesgo de enfermedad cardíaca, utilizando indicadores de salud que sirven en su predicción. Para lograr esto, se plantean los siguientes objetivos secundarios:

- Explorar los datos para comprender su estructura y la relación entre las variables, utilizando visualizaciones con T-SNE y UMAP.
- Normalizar o estandarizar las características numéricas que se utilizarán para el clustering.
- Implementar varios algoritmos de clustering, incluyendo KMeans, Gaussian Mixture Model, MeanShift y DBSCAN.
- Determinar el número óptimo de clusters mediante métodos como el método del codo o el índice de silhouette.
- Evaluar los resultados de los diferentes algoritmos de clustering.

El análisis sigue un flujo de trabajo que abarca la exploración y preprocesamiento de los datos, la implementación de los algoritmos de clustering y la evaluación comparativa de los resultados obtenidos. Se presentan visualizaciones que permiten interpretar la segmentación, contribuyendo así a la identificación de perfiles de pacientes diferenciados en términos de factores de riesgo cardiovascular.

3 Descripción del Conjunto de Datos

El conjunto de datos "Heart Disease", disponible en Kaggle, fue recopilado en 1988 y combina información de cuatro bases de datos clínicas: Cleveland, Hungría, Suiza y Long Beach V. Aunque contiene originalmente 76 atributos, los estudios publicados suelen emplear un subconjunto de 14 variables relevantes, incluyendo la variable objetivo que indica la presencia de enfermedad cardíaca.

A continuación, se describen los atributos utilizados en este análisis:

- **Age (edad):** Edad del paciente en años.
- **Sex (sexo):** Género del paciente (1 = masculino, 0 = femenino).
- **Chest pain type (tipo de dolor en el pecho):** Tipo de dolor en el pecho experimentado, clasificado en 4 valores distintos que representan diferentes condiciones clínicas.
- **Resting blood pressure (presión arterial en reposo):** Presión arterial medida en reposo (mmHg).
- **Serum cholestoral (colesterol sérico):** Nivel de colesterol sérico en mg/dl.
- **Fasting blood sugar (glucosa en ayunas):** Glucosa en sangre en ayunas (>120 mg/dl = 1, en caso contrario = 0).
- **Resting electrocardiographic results (resultado del electrocardiograma en reposo):** Resultado del electrocardiograma en reposo, codificado en 3 valores posibles (0, 1 y 2).
- **Maximum heart rate achieved (frecuencia cardíaca máxima alcanzada):** Frecuencia cardíaca máxima lograda durante el ejercicio.
- **Exercise induced angina (angina inducida por ejercicio):** Variable binaria que indica la presencia de angina provocada por el ejer-

cicio (1 = sí, 0 = no).

- **Oldpeak (depresión ST inducida por ejercicio):** Nivel de depresión del segmento ST inducido por ejercicio en comparación con el reposo.
- **Slope of the peak exercise ST segment (pendiente del segmento ST en el ejercicio máximo):** Pendiente del segmento ST durante el esfuerzo, lo que puede indicar anomalías.
- **Number of major vessels colored by fluoroscopy (número de vasos principales coloreados por fluoroscopia):** Número de vasos principales (de 0 a 3) observados a través de fluoroscopia.
- **Thal:** Categoría relacionada con defectos detectados, codificada como 0 = normal; 1 = defecto fijo; 2 = defecto reversible.
- **Target:** Se utiliza para diagnosticar la enfermedad, donde un valor de 1 indica la presencia de enfermedad cardíaca y un valor de 0 su ausencia.

Este conjunto de datos representa una base rica en variables clínicas que pueden relacionarse con el riesgo de enfermedad cardíaca.

4 Gestión y Preparación de Datos

4.1 Carga y Exploración de los Datos

En esta sección se importa el conjunto de datos desde una URL y se almacena en un DataFrame de pandas, convirtiendo cada columna a formato numérico y reemplazando valores inválidos con `NaN` para asegurar la consistencia. Se verifica y se eliminan filas con valores faltantes. Luego, se calculan estadísticas descriptivas para cada variable y se visualizan sus distribuciones mediante histogramas. Finalmente, se genera una matriz de correlación en forma de mapa de calor para analizar relaciones entre las variables y detectar patrones.

4.2 Preprocesamiento

En esta fase, se optimizan los datos para los modelos de aprendizaje automático mediante normalización y reducción de dimensionalidad. Las características numéricas, excluyendo la variable de salida `target`, se norma-

lizan usando `StandardScaler` para asegurar una contribución equitativa al análisis. Posteriormente, se aplican técnicas como T-SNE, UMAP y PCA para reducir las dimensiones a dos componentes, lo que facilita la visualización en gráficos de dispersión coloreados según `target`. Además, para PCA, se visualizan los datos reducidos en un gráfico de dispersión y se calcula la varianza explicada por cada componente. Esto es crucial para evaluar la efectividad del proceso de reducción de dimensionalidad.

4.2.1 Proceso de Reducción de Dimensionalidad con T-SNE

El algoritmo T-SNE sigue un proceso de tres pasos para representar similitudes en alta y baja dimensión [1]. El proceso se divide en tres pasos:

- **Cálculo de Probabilidades Condicionales**: Se centra una distribución gaussiana en cada punto x_i para medir la densidad de otros puntos x_j y se obtienen las probabilidades condicionales $p_{j|i}$, con la desviación estándar σ_i determinada por la perplejidad (indica el número de vecinos) [1].

$$p_{j|i} = \frac{\exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma_i^2}\right)}{\sum_{k \neq i} \exp\left(-\frac{\|x_i - x_k\|^2}{2\sigma_i^2}\right)}$$

- **Distribución Aleatoria**: Se distribuyen aleatoriamente los puntos en un espacio reducido, utilizando distribuciones t-Student para calcular $q_{j|i}$ [1].

$$q_{j|i} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k \neq i}(1 + \|y_i - y_k\|^2)^{-1}}$$

- **Minimización de Divergencias**: Se comparan las similitudes mediante la divergencia Kullback-Leibler (KL) y se minimizan mediante descenso por gradiente para lograr la representación en el espacio reducido [1].

$$\mathrm{KL}(P\|Q) = \sum_i \sum_j p_{j|i} \log\left(\frac{p_{j|i}}{q_{j|i}}\right)$$

4.2.2 Proceso de Reducción de Dimensionalidad con UMAP

El algoritmo UMAP reduce la dimensionalidad de datos complejos construyendo un grafo ponderado que refleja las relaciones entre los puntos de datos [2]. El proceso consta de varias etapas:

- **Construcción del Grafo de Vecinos:** Se crea un grafo ponderado que representa las conexiones locales entre los puntos en el espacio de alta dimensión [2].
- **Optimización de la Proyección:** Se aplican técnicas de optimización para desarrollar una proyección en un espacio de menor dimensión que conserve las relaciones del grafo de vecinos [2] [3].
- **Optimización de la Distancia Cruzada:** Se ajusta la función de distancia cruzada para medir la discrepancia entre las distancias del espacio original y las del espacio proyectado, minimizando una función de costo que refleja la distorsión del mapeo [2] [3].

Al finalizar, los datos se pueden visualizar en el espacio reducido, lo que facilita la aplicación de técnicas de agrupamiento [3].

4.2.3 Proceso de Reducción de Dimensionalidad con PCA

El algoritmo PCA reduce la dimensionalidad de un conjunto de datos, preservando al mismo tiempo las tendencias y patrones más relevantes [4]. El proceso de PCA se lleva a cabo en varios pasos:

- Estandarización de las variables continuas.
- Cálculo de la matriz de covarianza para identificar las correlaciones entre las variables [4]. La matriz de covarianza C se define como:

$$C = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})(x_i - \bar{x})^T$$

  donde x_i son las observaciones y $\bar{x}$ es el vector de medias de las variables.
- Cálculo de vectores propios y los valores propios de la matriz de covarianza.

- Creación de un vector de características que permite decidir qué componentes principales se conservarán, basándose en la proporción de varianza que explican [4].
- Transformación de los datos a lo largo de los ejes de los componentes principales seleccionados mediante la multiplicación de los datos originales por la matriz de los vectores propios correspondientes a los componentes elegidos [4].

5 Metodología de Clustering

5.1 KMeans

El algoritmo K-means es un método no supervisado que, en términos generales, representa la solución a un problema de optimización. El objetivo es obtener clusters compactos, es decir, minimizar la suma de las distancias de los elementos de un cluster a su centroide. Para penalizar estas distancias, se utiliza la distancia al cuadrado [5]. Por lo tanto, la función de error que deseamos optimizar es la siguiente:

$$E = \sum_{i=1}^{K} \sum_{x_j \in S_i} \|x_j - \mu_i\|^2 \tag{1}$$

donde K es el número de clusters, S_i es el conjunto de puntos asignados al cluster i, y μ_i es el centroide del cluster i.

Para encontrar el centroide que minimiza E, derivamos la función con respecto a μ_i:

$$\begin{aligned}
\frac{\partial E}{\partial \mu_i} &= \frac{\partial}{\partial \mu_i} \sum_{x_j \in S_i} \|x_j - \mu_i\|^2 \\
&= \sum_{x_j \in S_i} \frac{\partial}{\partial \mu_i} \left((x_j - \mu_i)^\top (x_j - \mu_i) \right) \\
&= \sum_{x_j \in S_i} (-2(x_j - \mu_i)) \\
&= -2 \sum_{x_j \in S_i} x_j + 2|S_i|\mu_i
\end{aligned}$$

Igualando la derivada a cero para encontrar los extremos, tenemos:

$$-2 \sum_{x_j \in S_i} x_j + 2|S_i|\mu_i = 0 \tag{2}$$

Resolviendo para μ_i, obtenemos:

$$\mu_i = \frac{1}{|S_i|} \sum_{x_j \in S_i} x_j \tag{3}$$

Esto indica que el nuevo centroide μ_i debe ser el promedio de los puntos en el clúster S_i si queremos minimizar la función de error E.

5.1.1 Explicación del algoritmo

Teniendo esto en cuenta, se define el algoritmo K-means. Se inicializan los centroides seleccionando k puntos aleatorios de nuestro conjunto de datos. Luego, los elementos restantes se asignan a los clústeres de acuerdo con el centroide más cercano. A continuación, se calcula el punto promedio de cada clúster y se redefinen los centroides con estos valores. Este proceso se repite, omitiendo la inicialización, hasta que la diferencia entre los centroides anteriores y los actuales no supere un límite definido por la implementación específica; en nuestro caso, consideramos un valor de 1×10^{-4}. Además, para asegurar la rapidez en el entrenamiento, se ha establecido un límite de 100 iteraciones.

La convergencia se garantiza porque, en cada iteración, se minimiza más la función de error; sin embargo, solo se asegura la llegada a un mínimo local.

5.2 Gaussian Mixture Model

El Gaussian Mixture Model (GMM) es un enfoque probabilístico que representa un conjunto de datos como una combinación de múltiples distribuciones gaussianas. Cada gaussiana en la mezcla se define por tres parámetros principales: la media (μ), la covarianza (Σ), y la probabilidad de mezcla (π), que determina la contribución de cada gaussiana al modelo global [6]. Podemos expresar la función de densidad de probabilidad de una mezcla gaussiana de la siguiente manera:

$$p(x) = \sum_{k=1}^{K} \pi_k \cdot \mathcal{N}(x|\mu_k, \Sigma_k) \tag{4}$$

donde K es el número de componentes (o clusters), x son los datos, π_k es la probabilidad de mezcla de la componente k, y $\mathcal{N}(x|\mu_k, \Sigma_k)$ es la función de densidad de la gaussiana, que se define como:

$$\mathcal{N}(x|\mu, \Sigma) = \frac{1}{(2\pi)^{D/2}|\Sigma|^{1/2}} e^{-\frac{1}{2}(x-\mu)^\top \Sigma^{-1}(x-\mu)} \tag{5}$$

donde D es el número de dimensiones, μ es la media y Σ es la covarianza.

Para determinar los parámetros óptimos de las gaussianas, se utiliza el método de máxima verosimilitud (MLE). Dado que el modelo incluye múltiples gaussianas, el proceso de estimación se complica y se recurre al algoritmo de Expectation-Maximization (EM). En la fase E (Expectation), se estima la pertenencia de los datos a cada componente de la mezcla:

$$\gamma_{ik} = \frac{\pi_k \cdot \mathcal{N}(x_i|\mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \cdot \mathcal{N}(x_i|\mu_j, \Sigma_j)} \tag{6}$$

donde γ_{ik} es la probabilidad de que el punto x_i pertenezca al componente k. En la fase M (Maximization), los parámetros se actualizan de la siguiente manera:

$$\pi_k = \frac{N_k}{N} \tag{7}$$

$$\mu_k = \frac{\sum_{i=1}^{N} \gamma_{ik} x_i}{N_k} \tag{8}$$

$$\Sigma_k = \frac{\sum_{i=1}^{N} \gamma_{ik}(x_i - \mu_k)(x_i - \mu_k)^\top}{N_k} \tag{9}$$

donde N_k es el número total de puntos asignados a la componente k y N es el total de datos.

5.2.1 Explicación del algoritmo

El algoritmo GMM se utiliza para modelar distribuciones de datos complejas y se basa en la idea de que los datos pueden generarse a partir de una mezcla de distribuciones gaussianas. Para nuestra implementación, utilizamos la librería `sklearn` en Python, específicamente `GaussianMixture`.

Iteramos en un rango para hallar el número adecuado de componentes K, utilizando el coeficiente de silueta como criterio. Este coeficiente mide la calidad de la agrupación al comparar la distancia media entre los puntos en el mismo cluster y la distancia media a los puntos en el cluster más cercano.

Finalmente, se ajusta el modelo GMM a los datos utilizando el mejor número de componentes identificado. Al concluir, obtenemos los parámetros de cada gaussiana y los centros de los clusters.

5.3 Mean Shift

El algoritmo no supervisado de Mean Shift es un método iterativo de clustering basado en densidades. Utiliza la Estimación de Densidad por Kernel para encontrar las zonas de alta densidad en la data y agruparla en base a estas [7].

5.3.1 Estimación de Densidad por Kernel

Dado un conjunto de puntos $\{x_1, x_2, ...x_n\}$, la estimación de densidad en un punto x usando un kernel K es:

$$\hat{f}(x) = \frac{1}{nh^d} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \tag{10}$$

donde n es el número de puntos de datos, h es el parámetro de ancho de banda que determina la suavidad de la densidad, d es la dimensión del espacio de datos y K es el kernel, usualmente una función gaussiana que pondera la influencia de los puntos vecinos.

5.3.2 Vector de Mean Shift

El cálculo del vector de desplazamiento, vector para mover cada punto hacia la región de mayor densidad. Este vector para un punto x se define

como:

$$m(x) = \frac{\sum_{x_i \in N(x)} K(x_i - x) x_i}{\sum_{x_i \in N(x)} K(x_i - x)} - x \tag{11}$$

donde x es la posición actual del punto, $N(x)$ es el vecindario de puntos dentro de una ventana de ancho h alrededor de x_i.

5.3.3 Algoritmo

1. **Inicialización**: Asignar cada punto de datos x_i como un posible centro de clúster.
2. **Cálculo del vector de desplazamiento**: Para cada punto x_i, calcular el vector Mean Shift $m(x_i)$ y desplazar x_i hacia la media de los puntos en su vecindario.
3. **Actualización iterativa**: Repetir el cálculo y desplazamiento hasta que todos los puntos converjan (es decir, cuando el desplazamiento $m(x_i)$ es cercano a cero para cada punto).

5.4 DBSCAN

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) es un método de clustering basado en la densidad local que encuentra las áreas más densas y las expande para identificar el mayor número de clústeres. Los principales parámetros para la búsqueda son la distancia ϵ, que define el radio de la región de vecindad, y 'Min Points', que representa el número mínimo de puntos que debe tener cada región [8]. Para cada observación o punto del conjunto de datos, la estimación de densidad local se puede describir de la siguiente manera:

- Para cada observación, se establecen los puntos según la distancia máxima, conformando una región circular conocida como "vecindad ϵ de la observación".
- Si dicha observación tiene al menos un cierto número de vecinos, incluida ella misma, se considera una observación central y, por ende, una observación de alta densidad.
- Las observaciones alrededor de esta central se consideran parte del clúster o región. Por lo tanto, un clúster se forma a partir de varias

observaciones centrales.

- Si alguna observación no central no presenta vecinos en su región, se considera una anomalía, lo que es útil para identificar valores atípicos.

Para calcular la distancia entre cada punto y sus vecinos, se utiliza la distancia euclidiana:

$$d(p,q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_n - q_n)^2} = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2} \tag{12}$$

En cada observación, para contar el número de vecinos a una distancia máxima de ϵ, se verifica que la distancia euclidiana sea inferior a ϵ:

$$N_\epsilon(p) = \{q \in \mathbb{R}^n \mid d(p,q) < \epsilon\}$$

6 Resultados y Análisis

6.1 Exploración de los Datos

En esta fase inicial de exploración, se realizaron análisis estadísticos y visualizaciones para entender mejor la distribución y correlación de las variables en el conjunto de datos "Heart Disease". Los gráficos generados incluyen distribuciones de variables individuales, una matriz de correlación y representaciones de los datos en espacios reducidos de dimensionalidad utilizando UMAP, T-SNE y PCA. Estos análisis revelaron patrones importantes que se detallan a continuación.

6.1.1 Métricas Estadísticas

Cuadro 1: Estadísticas descriptivas de las columnas del DataFrame

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
count	1025	1025	1025	1025	1025	1025	1025	1025	1025	1025	1025	1025	1025	1025
mean	54.43	0.70	0.94	131.61	246.00	0.15	0.53	149.11	0.34	1.07	1.39	0.75	2.32	0.51
std	9.07	0.46	1.03	17.52	51.59	0.36	0.53	23.01	0.47	1.18	0.62	1.03	0.62	0.50
min	29	0	0	94	126	0	0	71	0	0.0	0	0	0	0
25 %	48	0	0	120	211	0	0	132	0	0.0	1	0	2	0
50 %	56	1	1	130	240	0	1	152	0	0.8	1	0	2	1
75 %	61	1	2	140	275	0	1	166	1	1.8	2	1	3	1
max	77	1	3	200	564	1	2	202	1	6.2	2	4	3	1

- **count**: En este caso, cada columna contiene 1025 valores, porque el conjunto de datos tiene 1025 filas sin valores faltantes.
- **mean (Media)**: Es el promedio de los valores en cada columna. Este valor es útil para comprender el centro de cada variable. Por ejemplo:
  - La media de la columna `age` (edad) es de 54.43 años.
  - La media en la columna `sex` es de 0.6956, lo que es coherente con la data donde los valores están codificados como 0 para femenino y 1 para masculino. Mostrando una mayor cantidad de personas representadas por el valor 1.
- **std (Desviación Estándar)**: Mide la dispersión de los datos en torno a la media. Una desviación estándar alta indica una mayor variabilidad. Por ejemplo:
  - La columna `trestbps` (presión arterial en reposo) tiene una desviación estándar de 17.52, lo que muestra cierta variabilidad en los valores de presión arterial.
- **min, 25 %, 50 %, 75 %, max**:
  - **min**: Valor mínimo de cada columna.
  - **25 % (Primer Cuartil)**: Valor por debajo del cual se encuentra el 25 % de los datos.
  - **50 % (Mediana)**: Mediana de cada columna, el valor que divide al conjunto de datos en dos partes iguales.
  - **75 % (Tercer Cuartil)**: Valor por debajo del cual se encuentra el 75 % de los datos.
  - **max**: Valor máximo en cada columna.

  Ejemplo en la columna `chol` (colesterol):
  - Valor mínimo: 126
  - Primer cuartil (25 %): 211
  - Mediana (50 %): 240
  - Tercer cuartil (75 %): 275
  - Valor máximo: 564

  Además, en la columna `thal` (tipo de talasemia), los valores varían de 0 a 3, lo cual es coherente con el tipo de variable (categórica).

6.1.2 Distribución de las Variables

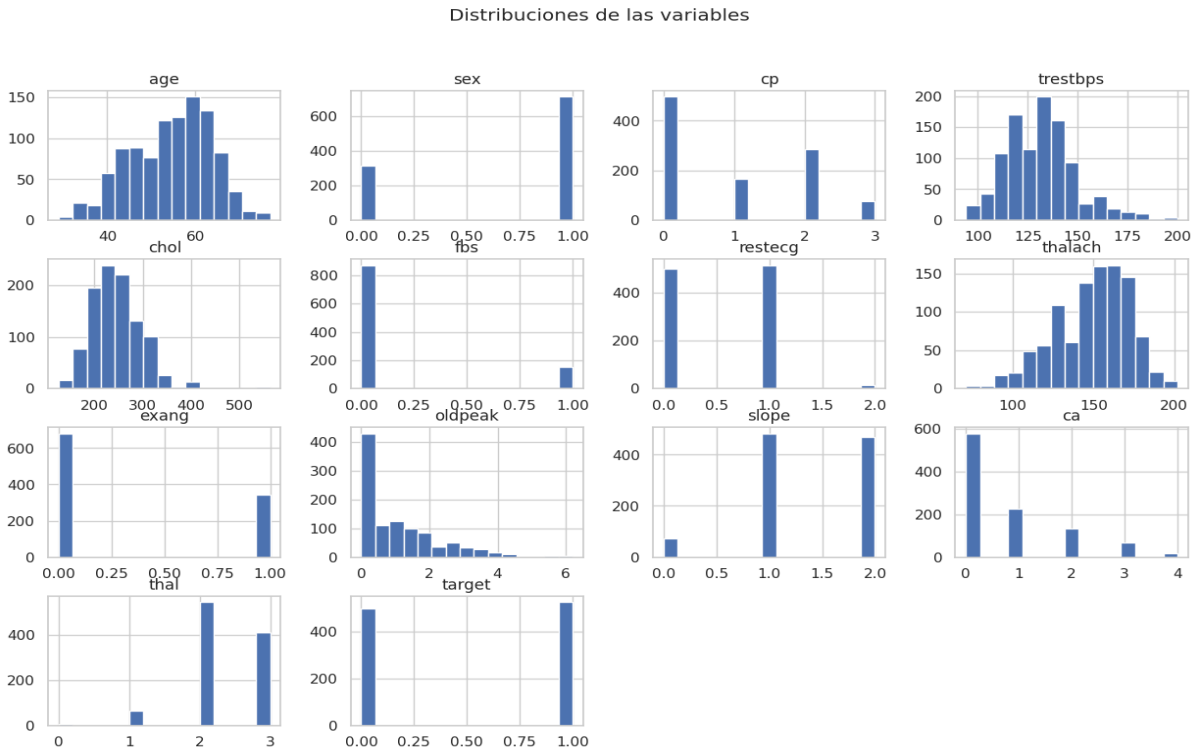

Figura 1: Distribuciones de las variables

La Figura 1 muestra las distribuciones de las variables relevantes, lo cual permite observar cómo se distribuyen los datos y detectar posibles sesgos. Por ejemplo:

- Edad (Age) y colesterol sérico (chol) presentan una distribución aproximadamente normal, mientras que otras variables, como glucosa en ayunas (fbs) y tipo de dolor en el pecho (cp), muestran distribuciones más sesgadas, lo que puede indicar la presencia de valores atípicos o de subgrupos específicos dentro de la población analizada.
- La distribución de frecuencia cardíaca máxima alcanzada (thalach) y presión arterial en reposo (trestbps) refleja una variabilidad que es importante en el contexto de riesgo cardiovascular, sugiriendo que ciertos niveles extremos pueden estar asociados con un mayor riesgo de enfermedad cardíaca.

6.1.3 Matriz de Correlación

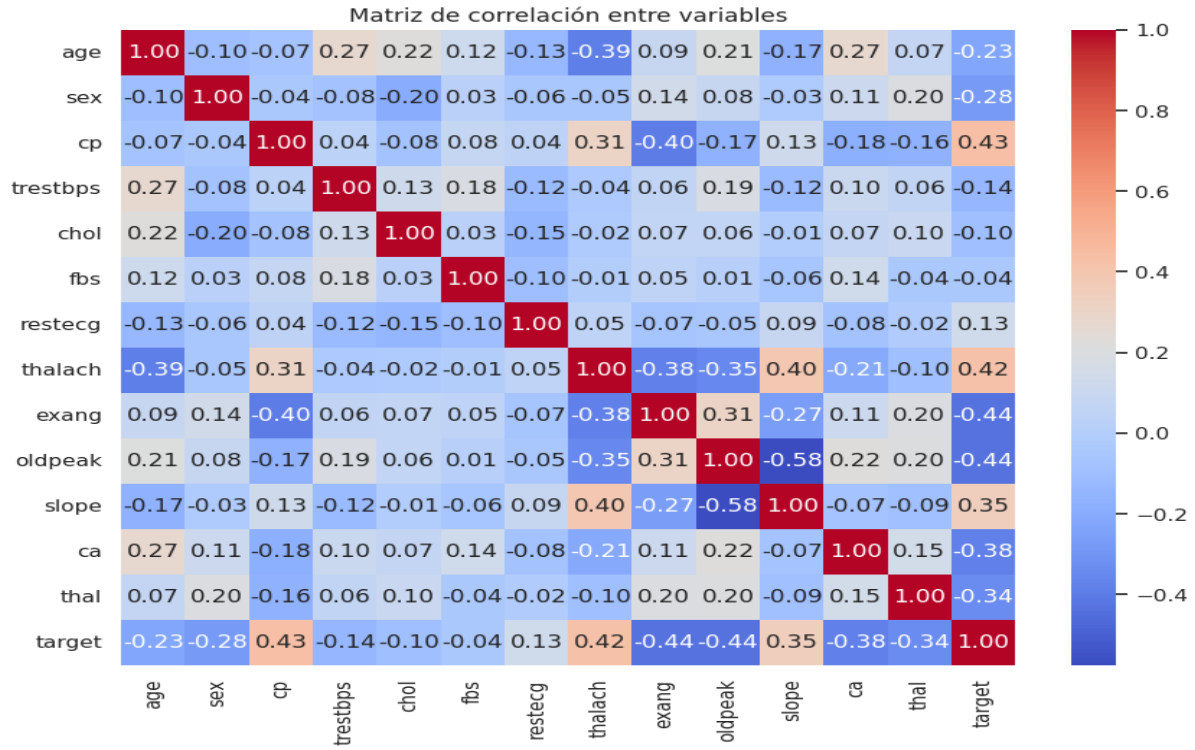

Figura 2: Matriz de correlación entre variable

La matriz de correlación en la Figura 2 proporciona una visión integral de las relaciones entre las variables. Las observaciones más relevantes incluyen:

- Una correlación positiva entre presión arterial en reposo (trestbps) y colesterol sérico (chol), lo que podría indicar que los pacientes con colesterol más alto tienden también a tener presión arterial elevada, ambos factores de riesgo para la enfermedad cardíaca.
- Una fuerte correlación negativa entre frecuencia cardíaca máxima (thalach) y edad (age), lo cual es consistente con la disminución de la frecuencia cardíaca máxima que ocurre naturalmente con la edad.
- Las variables tipo de dolor en el pecho (cp) y frecuencia cardíaca máxima(thalach) tienen correlaciones positivas con el **target** (presencia de enfermedad cardíaca), lo cual subraya su importancia como posibles predictores de riesgo.

6.1.4 Reducción de Dimensionalidad

Para visualizar el conjunto de datos en un espacio reducido y explorar agrupaciones latentes, se aplicaron técnicas de reducción de dimensionalidad como UMAP, T-SNE y PCA (Figuras 3, 4 y 5).

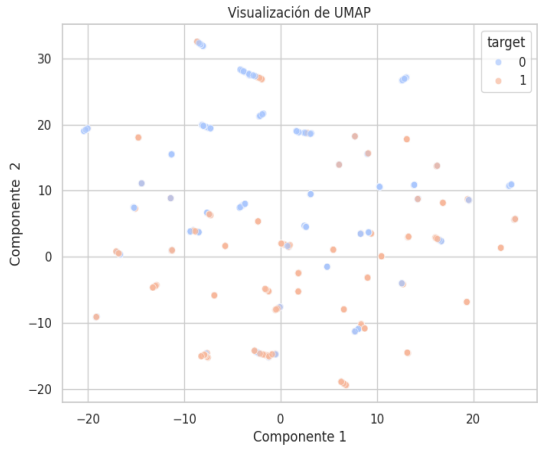

Figura 3: Visualización de UMAP

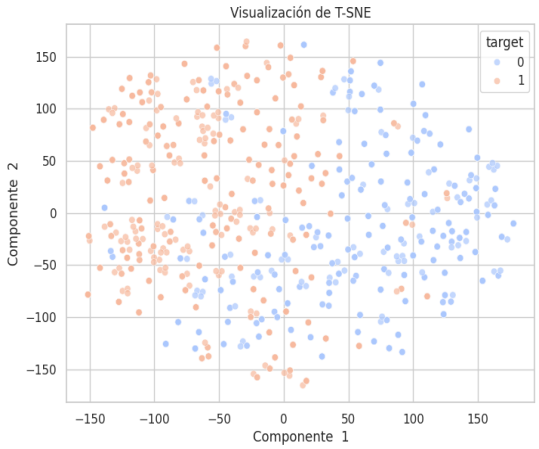

Figura 4: Visualización de T-SNE

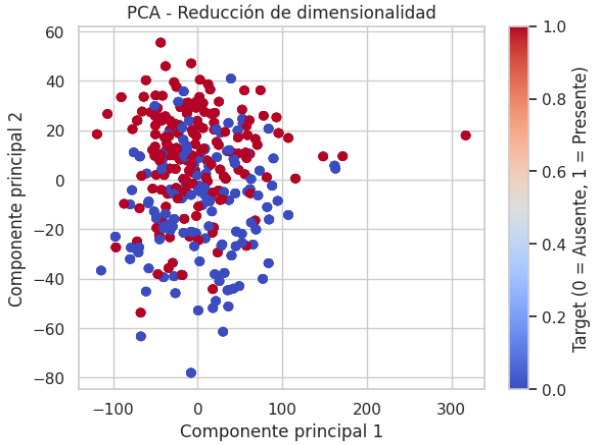

Figura 5: PCA

- **UMAP y T-SNE**: Ambos métodos revelan una estructura en los datos que parece consistente con posibles subgrupos de pacientes, especialmente en los extremos de las distribuciones. La separación en estas visualizaciones podría corresponder a pacientes con diferentes niveles de riesgo cardiovascular, aunque sin una separación clara de clusters bien definida.
- **PCA**: La Figura 5 muestra los datos proyectados en los dos primeros componentes principales, los cuales explican el 74.53 % y 15.20 % de

la varianza, respectivamente, como se indica en el Cuadro 1. Esta alta proporción de varianza explicada sugiere que los datos pueden ser bien representados en este espacio de dos dimensiones, facilitando la aplicación de técnicas de clustering en etapas posteriores.

Cuadro 2: Varianza explicada por cada componente

Componente	**Varianza Explicada**
Componente 1	0.7453
Componente 2	0.1520

6.2 Metodología de Clustering

En esta sección se describen los métodos de clustering utilizados para analizar el conjunto de datos, con el objetivo de segmentar pacientes en grupos homogéneos según sus características clínicas. Se implementaron varios algoritmos de clustering no supervisado, incluidos *K-means*, *Gaussian Mixture Model*, *Mean Shift* y *DBSCAN*, los cuales permiten identificar patrones latentes en los datos que pueden estar asociados con diferentes niveles de riesgo cardiovascular.

6.2.1 Resultados del Clustering con K-means

A continuación, se presentan los resultados obtenidos al implementar el algoritmo de *K-means* para realizar la agrupación de los datos, mediante el análisis de las métricas de *silhouette* y las visualizaciones de los clústeres generados.

Determinación del Número Óptimo de Clústeres

Para identificar el número óptimo de clústeres (K), se utilizó el método del puntaje de *silhouette*. Este puntaje mide la cohesión y separación de los clústeres, proporcionando una métrica cuantitativa para seleccionar el valor de K que maximiza la calidad de la agrupación [9]. La Figura 6 muestra cómo el puntaje de *silhouette* varía en función del número de clústeres.

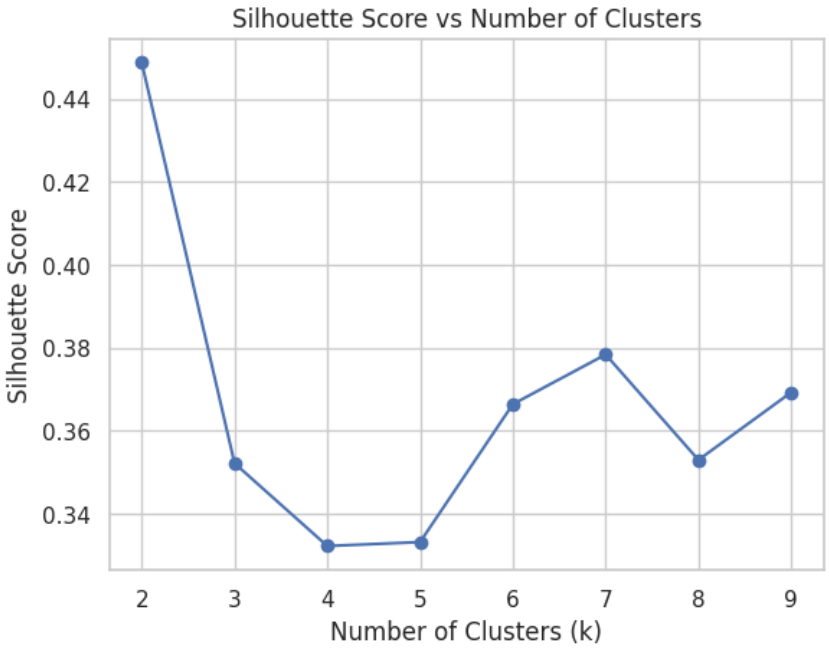

Figura 6: Puntaje de Silueta en función del número de clústeres (K)

En esta gráfica, se observa que el valor de K que maximiza el puntaje de *silhouette* corresponde a $K = \textbf{optimal\_k}$, el cual para este caso toma el valor de 2 e indica el número óptimo de clústeres para la segmentación de los datos.

Visualización de los Clústeres

Una vez determinado el número óptimo de clústeres, se ajustó el algoritmo *K-means* con $K =$ optimal_k. Los datos se proyectaron en el espacio bidimensional de los dos primeros componentes principales, obtenidos mediante *PCA*, lo que permitió visualizar los clústeres en relación con las características más importantes del conjunto de datos.

La Figura 7 muestra los clústeres identificados por *K-means*, donde cada punto representa un paciente y el color indica el clúster al que pertenece. Los centroides de los clústeres se marcan con un símbolo rojo en forma de X, representando la posición central de cada grupo.

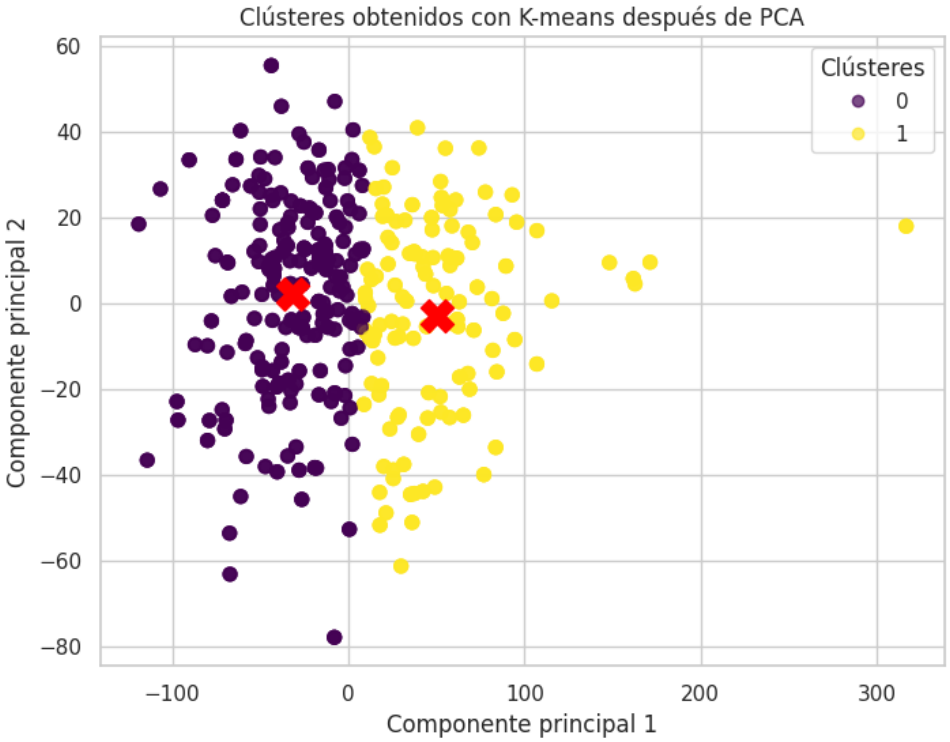

Figura 7: Clústeres obtenidos con K-means después de aplicar PCA

6.2.2 Resultados del Clustering con DBScan

A continuación, se presentan los resultados obtenidos al implementar el algoritmo de *DBScan* para realizar la agrupación de los datos, mediante el análisis de las métricas de *silhouette* y las visualizaciones de los clústeres generados.

Determinación del Valor Óptimo de ϵ

El valor de ϵ determina el radio de vecindad dentro del cual los puntos se consideran vecinos y son agrupados en el mismo clúster si cumplen con la densidad mínima definida por min_samples. Para seleccionar el valor óptimo de ϵ, se evaluó el coeficiente de silueta para una serie de valores de ϵ (de 0.1 a 200 en incrementos de 10), graficando los resultados como se muestra en la Figura 8.

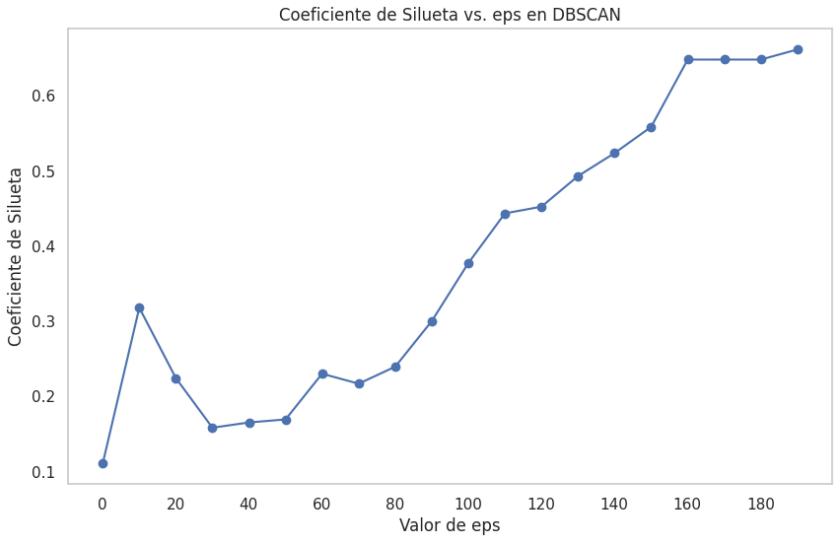

Figura 8: Puntaje de Silueta en función del número de el valor de ϵ

Visualización de los Clústeres

Con el valor óptimo de ϵ seleccionado (190), se realizó el ajuste final de *DBScan* y se visualizó la agrupación en el espacio bidimensional de los dos primeros componentes principales (obtenidos previamente mediante PCA) como se muestra en la Figura 9. Los clústeres encontrados por *DBScan* están representados por puntos de diferentes colores. Cada clúster agrupa puntos cercanos en términos de densidad, formando grupos compactos de pacientes con características clínicas similares.

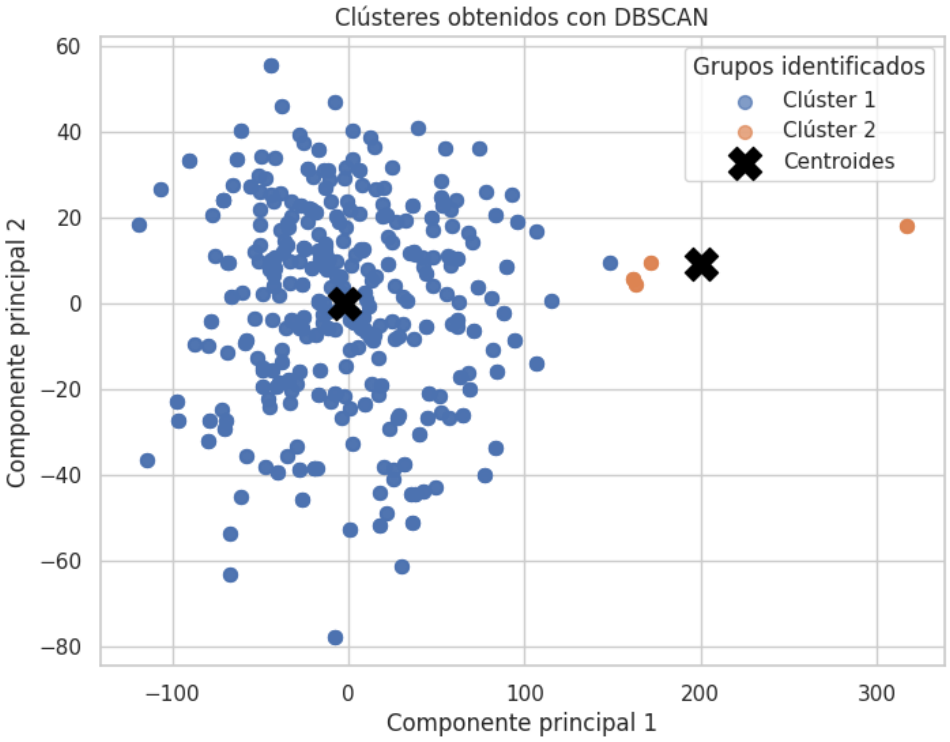

Figura 9: Clústeres obtenidos con DBScan después de aplicar PCA

6.2.3 Resultados del Clustering con Mean Shift

A continuación, se presentan los resultados obtenidos al implementar el algoritmo de *Mean Shift* para realizar la agrupación de los datos, mediante el análisis de las métricas de *silhouette* y las visualizaciones de los clústeres generados.

Optimización del Ancho de Banda

El parámetro clave para *Mean Shift* es el ancho de banda, que determina el radio de influencia de cada punto en el espacio. Para encontrar el valor óptimo, se probaron varios valores de cuantil: 0,1, 0,2, 0,3, 0,4 y 0,5. A continuación, se presentan los resultados obtenidos:

- **Cuantil** 0,1: Ancho de banda de 26,14, con un coeficiente de silueta de 0,33.
- **Cuantil** 0,2: Ancho de banda de 37,03, con un coeficiente de silueta de 0,44.
- **Cuantil** 0,3: Ancho de banda de 45,70, con un coeficiente de silueta de 0,44.

- **Cuantil** 0,4: Ancho de banda de 53,60, con un coeficiente de silueta de 0,66.
- **Cuantil** 0,5: Ancho de banda de 62,00, con un coeficiente de silueta de 0,66.

El mejor valor de cuantil fue 0,4 y 0,5, los cuales produjeron un coeficiente de silueta de 0,66. Esto indica una buena cohesión dentro de los clústeres, así como una separación adecuada entre ellos.

Análisis del Coeficiente de Silueta

El coeficiente de silueta calculado para los diferentes valores de cuantil se presenta en la Figura 10. Se observa que el coeficiente de silueta alcanza su valor máximo en los cuantiles 0,4 y 0,5, con una puntuación de 0,66. A medida que el cuantil disminuye, el coeficiente de silueta también lo hace, lo que indica que un ancho de banda demasiado pequeño no agrupa adecuadamente los puntos.

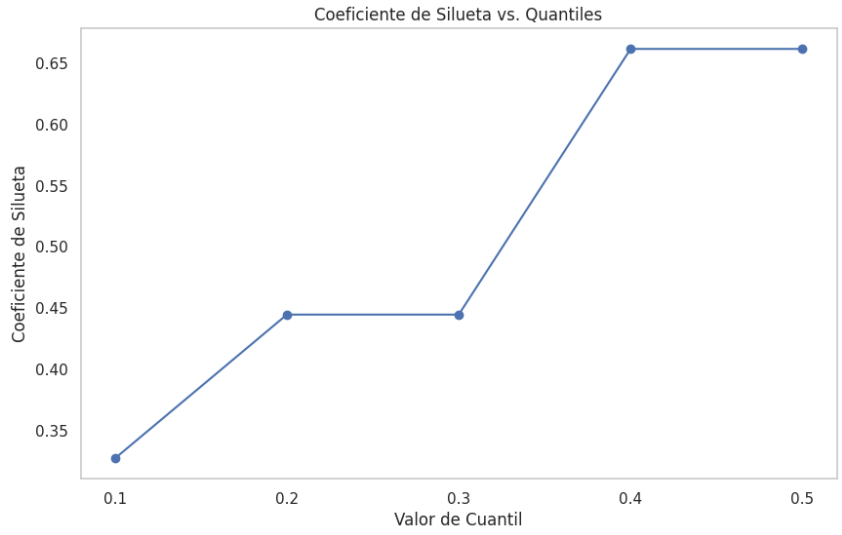

Figura 10: Coeficiente de Silueta vs. Cuantiles

Visualización de los Clústeres

En la Figura 11, se muestran los resultados del clustering utilizando el mejor ancho de banda. Los puntos de datos están coloreados según sus etiquetas de clúster, y los centros de los clústeres se indican con una marca en forma de X de color rojo.

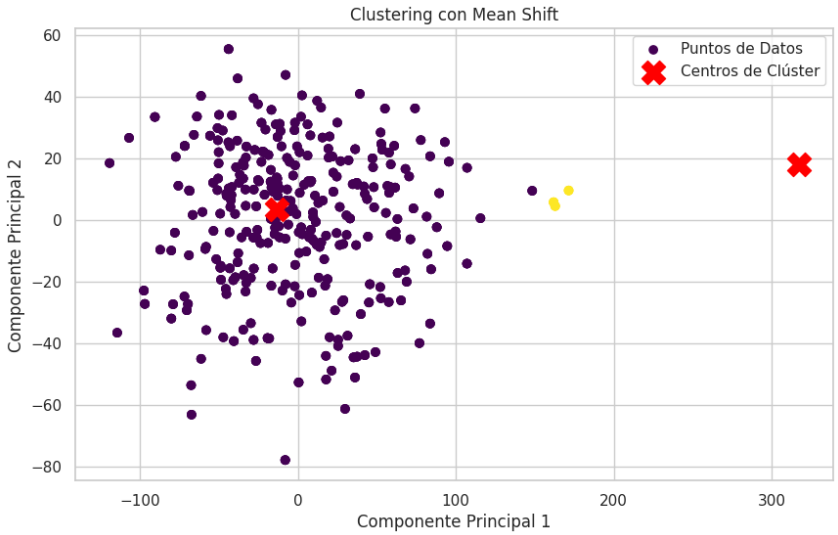

Figura 11: Visualización de Clustering con Mean Shift

6.2.4 Resultados del Clustering con Gaussian Mixture Model

A continuación, se presentan los resultados obtenidos al implementar el algoritmo de *GMM* para realizar la agrupación de los datos, mediante el análisis de las métricas de *silhouette* y las visualizaciones de los clústeres generados.

Optimización del Número de Componentes

El parámetro clave para el modelo *GMM* es el número de componentes, que define la cantidad de distribuciones gaussianas utilizadas para modelar los datos. Se evaluaron diferentes valores de `n_components` dentro del rango de 2 a 9, y para cada valor se calculó el coeficiente de silueta. Los resultados obtenidos son los siguientes:

- **Número de componentes** 2: Coeficiente de silueta de 0,30.
- **Número de componentes** 3: Coeficiente de silueta de 0,27.
- **Número de componentes** 4: Coeficiente de silueta de 0,28.
- **Número de componentes** 5: Coeficiente de silueta de 0,31.
- **Número de componentes** 6: Coeficiente de silueta de 0,31.
- **Número de componentes** 7: Coeficiente de silueta de 0,35.
- **Número de componentes** 8: Coeficiente de silueta de 0,31.
- **Número de componentes** 9: Coeficiente de silueta de 0,35.

El mejor valor de `n_components` fue 7 y 9, ambos con un coeficiente de silueta de 0,35. Este resultado indica que con 7 o 9 componentes, el modelo GMM es capaz de generar clústeres con mejor cohesión y separación en comparación con otras configuraciones.

Visualización de los Resultados

Con el número óptimo de componentes determinado, se ajustó el modelo GMM sobre los datos y se identificaron los centros de los clústeres. Los centros de los clústeres son los siguientes:

$$\text{Centros de Clústeres} = \begin{bmatrix} 33{,}97 & -38{,}80 \\ -37{,}70 & 16{,}52 \\ 43{,}73 & 13{,}21 \\ -73{,}73 & -12{,}34 \\ 78{,}89 & -5{,}34 \\ 316{,}85 & 17{,}91 \\ -34{,}68 & -18{,}25 \\ -0{,}71 & 6{,}51 \\ 161{,}15 & 7{,}14 \end{bmatrix}$$

En la Figura 12, se muestra la visualización de los datos agrupados utilizando el modelo GMM. Cada clúster está representado con un color diferente, y los centros de los clústeres están marcados con una X de color rojo. La visualización confirma una segmentación clara de los datos, con una separación adecuada entre los clústeres.

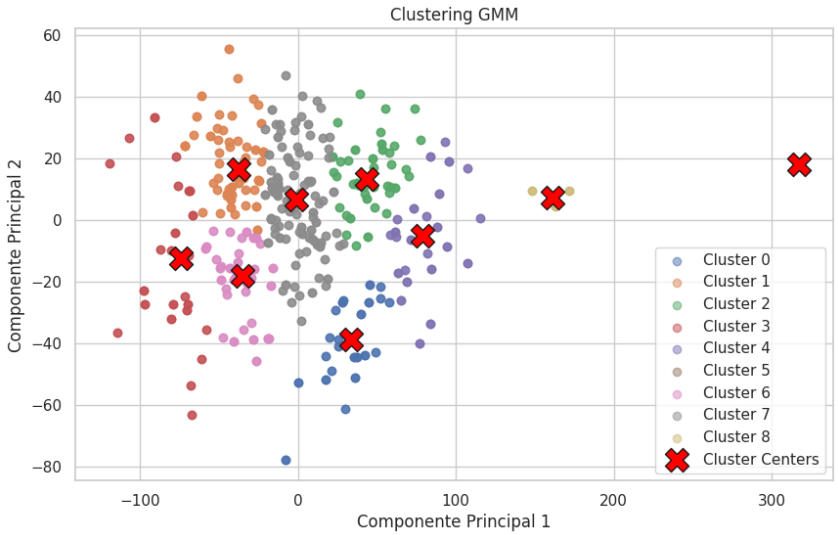

Figura 12: Visualización del Clustering con GMM

Análisis del Coeficiente de Silueta

El gráfico en la Figura 13 muestra la relación entre el número de componentes del modelo GMM y el coeficiente de silueta. Se observa que el coeficiente de silueta alcanza su valor máximo de 0,35 con 7 y 9 componentes. Esto indica que un número moderado de componentes proporciona la mejor segmentación de los datos.

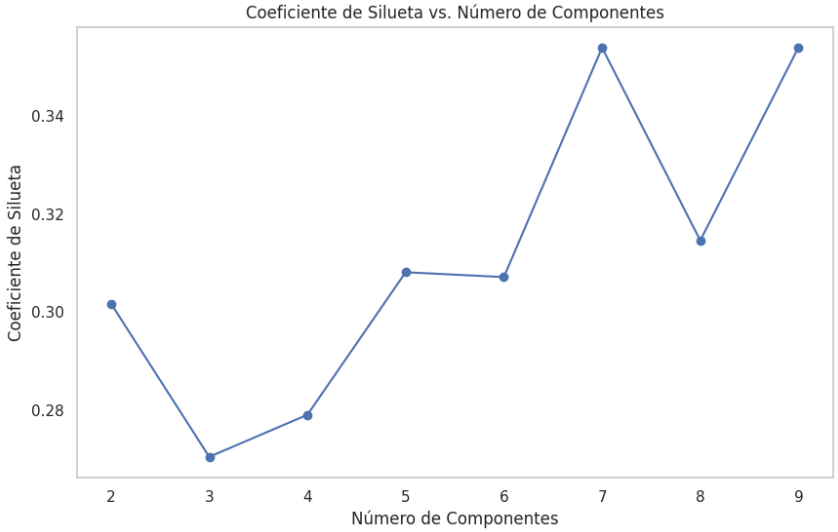

Figura 13: Coeficiente de Silueta vs. Número de Componentes en GMM

Al visualizar los clústeres a través de los diferentes algoritmos de agrupamiento, se identificaron patrones y relaciones importantes entre las características clínicas de los pacientes. Cada algoritmo reveló posibles perfiles de riesgo cardiovascular, con los centroides de los clústeres representando los valores promedio de las características clave en cada grupo. A continuación, se describen los principales clústeres observados para la mayoría de los algoritmos, excepto GMM, donde la distribución fue más compleja.

- **Clúster 1**: Este grupo agrupa a pacientes con características clínicas que indican un **riesgo cardiovascular bajo o moderado**. Los individuos en este clúster tienden a presentar niveles de **presión arterial** y **colesterol sérico** dentro de rangos considerados normales. Además, es menos probable que presenten **angina inducida por ejercicio**, y suelen alcanzar una **frecuencia cardíaca máxima** más alta, lo que sugiere un perfil de salud cardiovascular más favorable.
- **Clúster 2**: Este grupo corresponde a pacientes con un **riesgo cardiovascular elevado**. Los pacientes de este clúster presentan factores de riesgo más pronunciados, como niveles elevados de **presión arterial**

y **colesterol sérico**. También es más común que experimenten **dolor en el pecho** (angina) y tengan una menor **frecuencia cardíaca máxima alcanzada**. Estos indicadores están estrechamente vinculados con un mayor riesgo de enfermedades cardiovasculares, sugiriendo que este grupo podría beneficiarse de estrategias de intervención médica más agresivas.

Para el caso específico del *Gaussian Mixture Model*, la segmentación fue más granular, identificándose varios subgrupos en lugar de dos clústeres principales. Esto sugiere que el GMM puede capturar una mayor complejidad en los datos, reflejando la diversidad de los perfiles clínicos de los pacientes.

En resumen, los algoritmos de clustering aplicados proporcionan una visión clara de los perfiles de riesgo cardiovascular en la población analizada. Los clústeres revelan tanto grupos de bajo riesgo como aquellos con riesgos elevados, lo que puede ser útil para personalizar las estrategias de prevención y tratamiento de enfermedades cardiovasculares.

Enlaces Externos

- **Link del Proyecto:** https://colab.research.google.com/drive/1p4RMBhgU4ytPcVtdTLArJe6UVTdjRaUO?usp=sharing

Referencias

[1] Daniel. "Comprende el algoritmo t-SNE en 3 pasos." (oct. de 2023), dirección: https://datascientest.com/es/comprende-el-algoritmo-t-sne-en-3-pasos.

[2] ASIMOV Ingeniería S. De R.L. De C.V. "¿Que es UMAP?" (Mayo de 2024), dirección: https://asimov.cloud/blog/programacion-5/que-es-umap-328.

[3] Taniwa. "UMAP para descubrir tus datos." (2023), dirección: https://taniwa.es/blog/umap/.

[4] Z. Jaadi. "Principal Component Analysis (PCA): A Step-by-Step Explanation." (feb. de 2024), dirección: https://builtin.com/data-science/step-step-explanation-principal-component-analysis.

[5] Universidad de Oviedo. "El algoritmo k-means aplicado a clasificación y procesamiento de imágenes." (2024), dirección: https://www.unioviedo.es/compnum/laboratorios_py/kmeans/kmeans.html.

[6] O. C. Carrasco. "Gaussian Mixture Model Explained." (2024), dirección: https://builtin.com/articles/gaussian-mixture-model.

[7] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, n.º 8, págs. 790-799, 1995. DOI: 10.1109/34.400568.

[8] DataScientest. "Machine Learning Clustering: el algoritmo DBSCAN." (nov. de 2024), dirección: https://datascientest.com/es/machine-learning-clustering-dbscan.

[9] Scikit-learn. "silhouette_score." (), dirección: https://scikit-learn.org/1.5/modules/generated/sklearn.metrics.silhouette_score.html.