
Instituto Politécnico
Nacional.
Escuela Superior de Computo.

Asignatura: Aprendizaje de máquina

Profesor: Abdiel Reyes Vera.

# Práctica 6: Aprendizaje no supervisado

Alumnos:
Rodríguez Juárez Héctor Sajoni
Velásquez Arrieta Uriel Eduardo

Grupo: 5BM1

22 de septiembre del 2025

# Índice



# Índice de imágenes

# Introducción

La presente práctica tiene como objetivo introducirnos en el aprendizaje no supervisado, para ello, se desarrollan dos técnicas dentro de este paradigma, que son la reducción dimensional y el clustering.

La reducción dimensional es una técnica que reduce el número de características (dimensiones) en un conjunto de datos, transformándolo de un espacio de alta dimensión a uno de menor dimensión, mientras conserva la mayor parte de su información significativa. Sus principales beneficios incluyen la reducción del tiempo de entrenamiento, menores requerimientos computacionales y la facilidad para visualizar los datos. Algunas de sus técnicas más conocidas son el Análisis de Componentes Principales (PCA), el Análisis Discriminante Lineal (LDA) y el t-SNE.

El clustering o agrupamiento es la tarea de agrupar objetos por similitud, en grupos o conjuntos de manera que los miembros del mismo grupo tengan características similares. Es la principal tarea en minería de datos exploratoria y es una técnica común en el análisis de datos estadísticos. Esta tarea es considerada una técnica de aprendizaje no supervisado puesto que busca encontrar relaciones entre variables descriptivas, pero no la que guardan con respecto a una variable objetivo.

Con el objetivo de desarrollar la primera parte de esta práctica, se implementó el algoritmo PCA a 2 diferentes datasets, que son: iris dataset y mall customers dataset. Datests bastante simples, pero que cuentan ambos con 4 características, lo que los hace imposibles de visualizar sin una reducción dimensional.

El algoritmo PCA, o Análisis de Componentes Principales, es una técnica de reducción de dimensionalidad que simplifica conjuntos de datos grandes al transformarlos en un nuevo sistema de coordenadas con menos dimensiones, manteniendo la mayor parte de la información original. Su objetivo es encontrar las direcciones de máxima varianza en los datos y proyectarlos en un subespacio de menor dimensión para facilitar la visualización y el análisis, eliminar la redundancia y mejorar el rendimiento de otros algoritmos de aprendizaje automático.

Para trabajar la segunda parte de esta práctica se implementaron 5 algoritmos de agrupamiento (KMeans, SpectralClustering, DBSCAN, Birch, AgglomerativeClustering) sobre 4 datasets diversos (iris, mall customers, blob dataset y moon dataset.

Para realizar la comparación de desempeño entre las diferentes técnicas de agrupamiento que se utilizan en esta práctica se utiliza una métrica que en español recibe el nombre de coeficiente de silueta, la cual es fue calculada utilizando la función de scikit-learn silhouette_score

El coeficiente de silueta es una métrica utilizada para evaluar la calidad de la agrupación en clústeres. Mide qué tan bien un objeto se ajusta a su propio clúster en comparación con otros clústeres. Los valores oscilan entre -1 y 1; un valor cercano a 1 indica clústeres

bien separados y densos, mientras que un valor negativo sugiere que el objeto podría haber sido asignado incorrectamente al clúster equivocado.

# Desarrollo

El desarrollo de este estudio comparativo se estructura en varias fases metodológicas que garantizan una evaluación sistemática y rigurosa de los algoritmos de machine learning seleccionados. El flujo de trabajo implemento algunas prácticas asegurando la reproducibilidad y validez de los resultados obtenidos a través de un proceso metodológico robusto y bien documentado.

## Datasets Utilizados

La investigación emplea 4 datasets diversos que representan diferentes dominios de aplicación y características estructurales, proporcionando un marco comprehensivo para la evaluación de algoritmos. El primer conjunto corresponde al **Iris Dataset**, un dataset clásico de clasificación con 150 muestras de flores iris, 4 características numéricas y 3 clases, obtenido directamente de scikit-learn. Le sigue el **Mall Customers Dataset**, un conjunto de datos de clientes de un centro comercial con 200 objetos de 4 características de interés, pero sin una variable objetivo definida.

Los otros 2 datasets tiene en común que no están definidos, sino que son generados bajo demanda de manera aleatoria siguiendo unas directrices, ambos se encuentran disponibles en scikit-learn, el primero de ellos es el de **Make Blobs**, el cual genera puntos alrededor de clusters aleatorios siguiendo una distribución gausiana. El último dataset es **Make Moons** que crea secuencias de puntos que conforman 2 semicírculos intercalados.

## Preprocesamiento de Datos

Al ser conjuntos de datos extremadamente simples solo requieren un preprocesamiento simple, para el dataset iris basta con eliminar la variable objetivo, lo que lo hace apto para tareas de aprendizaje no supervisado. En el caso de mall customers, tan solo hace falta eliminar la columna de id de cliente, puesto que es un número que no aporta nada y dejarlo solo entorpecería el entrenamiento del modelo. Para los otros 2 dataset no se realiza un preprocesamiento puesto que ya están pensados para ser utilizados en experimentos de agrupamiento.

Es justo mencionar que el aprendizaje supervisado es en ocasiones utilizado para tareas de preprocesamiento, por ejemplo, este es el principal uso que se le da al análisis de componentes principales; lo que explica porque en esta ocasión no se requiere un gran procesamiento previo de los datos. Puesto que los algoritmos implementados ya tienen esta tarea de por sí.

# Modelos de agrupamiento

## KMeans

divide un conjunto de datos en \(k\) grupos o clústeres. Su objetivo es minimizar la suma de las distancias entre cada punto de dato y el centroide de su clúster. El proceso implica inicializar \(k\) centroides aleatorios, asignar cada punto al centroide más cercano y luego re-calcular los centroides como la media de los puntos asignados, repitiendo esto hasta que las asignaciones de clústeres no cambien.

## Spectral clustering

El clustering espectral es una técnica de agrupación que trata los datos como nodos en un grafo y utiliza sus propiedades espectrales (autovalores y autovectores) para transformar los datos en un espacio de menor dimensión, donde los grupos son más fáciles de separar. Es especialmente útil para datos no linealmente separables, ya que no asume una forma de clúster predefinida.

## DBSCAN

El algoritmo DBSCAN (Density-Based Spatial Clustering of Applications with Noise) es un método de agrupamiento no supervisado que agrupa puntos de datos cercanos en regiones de alta densidad y marca como valores atípicos los puntos aislados en regiones de baja densidad. A diferencia de otros algoritmos, no requiere predefinir el número de clústeres y puede identificar clústeres de formas arbitrarias.

## Birch

El agrupamiento BIRCH es un algoritmo de aprendizaje no supervisado para agrupar grandes conjuntos de datos de manera eficiente, creando un resumen de los datos mediante un árbol CF (Characteristic Feature) (Árbol de características de agrupamiento) y luego realizando un agrupamiento en este resumen. Su principal ventaja es que es eficiente en memoria y tiempo, lo que le permite manejar conjuntos de datos masivos que no cabrían en la memoria RAM para otros algoritmos como K-Means.

## Agglomerative clustering

La agrupación aglomerativa es un enfoque de agrupación ascendente y jerárquica que parte de cada punto de datos como un clúster independiente y fusiona iterativamente los dos clústeres más similares hasta que solo queda uno. Es un proceso ascendente porque se construye a partir de puntos de datos individuales hasta formar clústeres más grandes. El resultado es una estructura arborizada llamada dendrograma, que visualiza la jerarquía y puede cortarse a una altura específica para obtener un conjunto plano de clústeres.

# Estrategia de Evaluación

La estrategia de evaluación se diseña para proporcionar una comparación justa y estadísticamente válida entre todos los algoritmos. Esta empieza utilizar la misma

instancia de los datasets de blob y moon para todos los modelos, las cuales constan de 300 muestras, y 3 centroides para el caso de blobs dataset.

La métrica de evaluación seleccionada proporciona una visión multidimensional del rendimiento de cada modelo. El coeficiente de silueta tiene como propósito evaluar la calidad de la agrupación en clústeres, los modelos que tienen este parámetro más cercano a 1 son los que tuvieron un mejor rendimiento, mientras que los que tienen este valor más bajo son considerados como los peores.

## Visualización y Análisis

Los resultados se grafican mediante la librería matplotlib. El sistema aprovecha los gráficos de dispersión para presentar la clasificación de los datos, facilitando el análisis cuantitativo de los mismos.

## Resultados

La ejecución de los 5 algoritmos sobre los 4 datasets nos permite apreciar el resultado de las distintas heurísticas de clasificación utilizadas. Algunos de los modelos empleados muestran variaciones considerables en el rendimiento, sin embargo, sería incorrecto tomar esta evaluación como una medida de la calidad general de los modelos puesto que eso depende completamente de dataset a estudiar y el propósito de este procesamiento.

### Dataset Iris

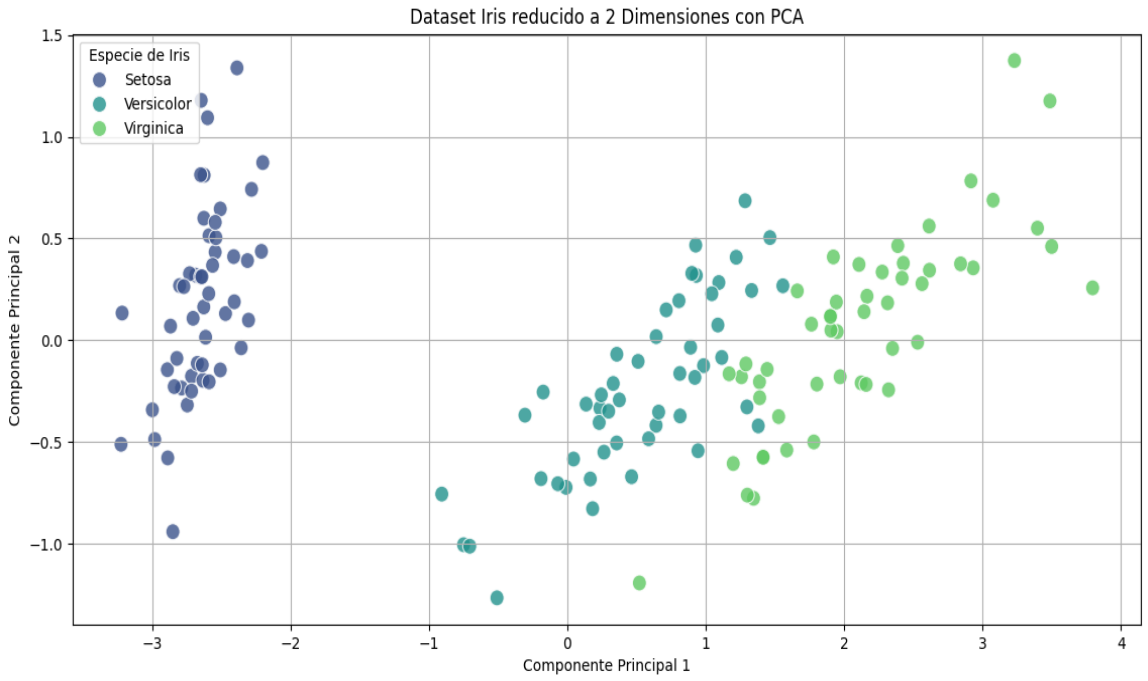


*Ilustración 1. Dataset Iris reducido a 2 dimensiones con PCA*

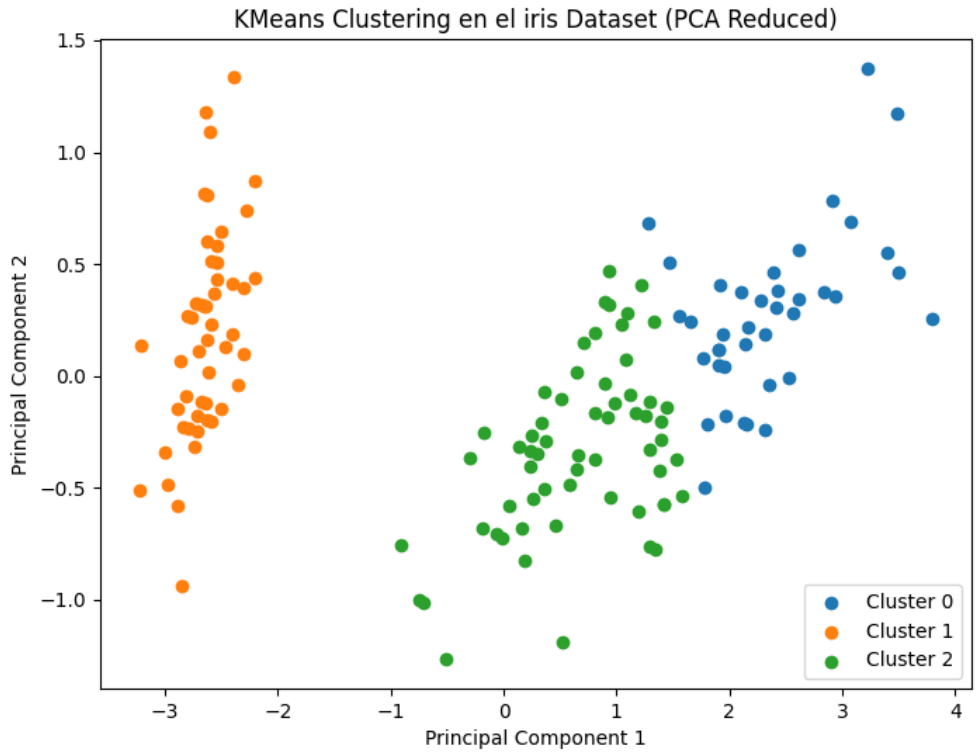


*Ilustración 2. KMeans en Iris Dataset*


*Ilustración 3. Spectral Clustering en Iris Dataset*

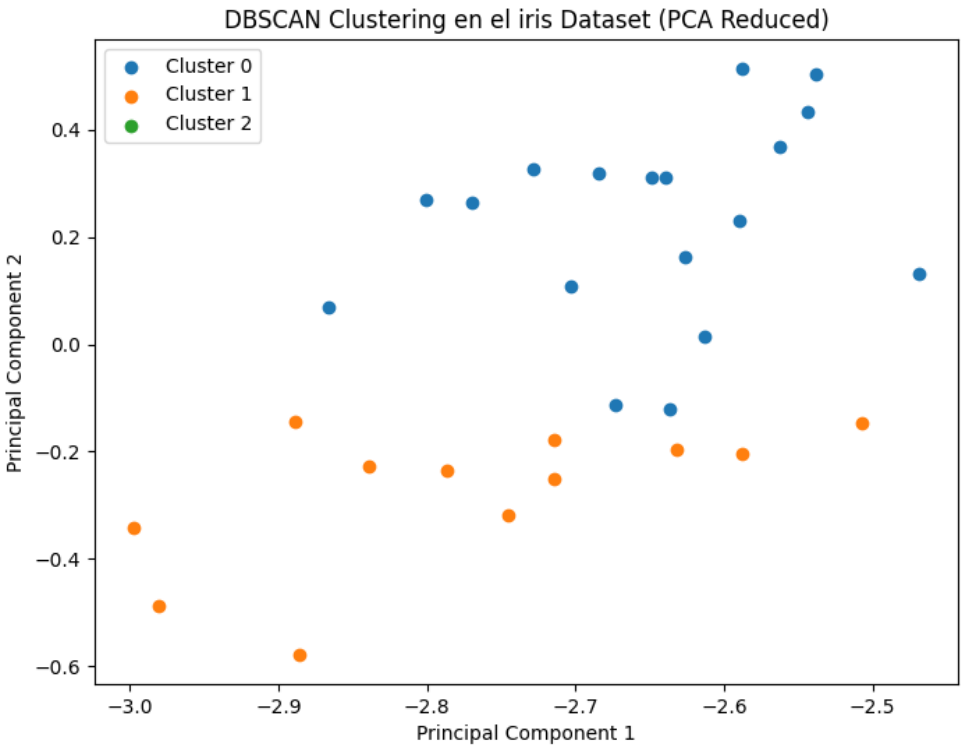


*Ilustración 4. DBSCAN Clustering en Iris Dataset*

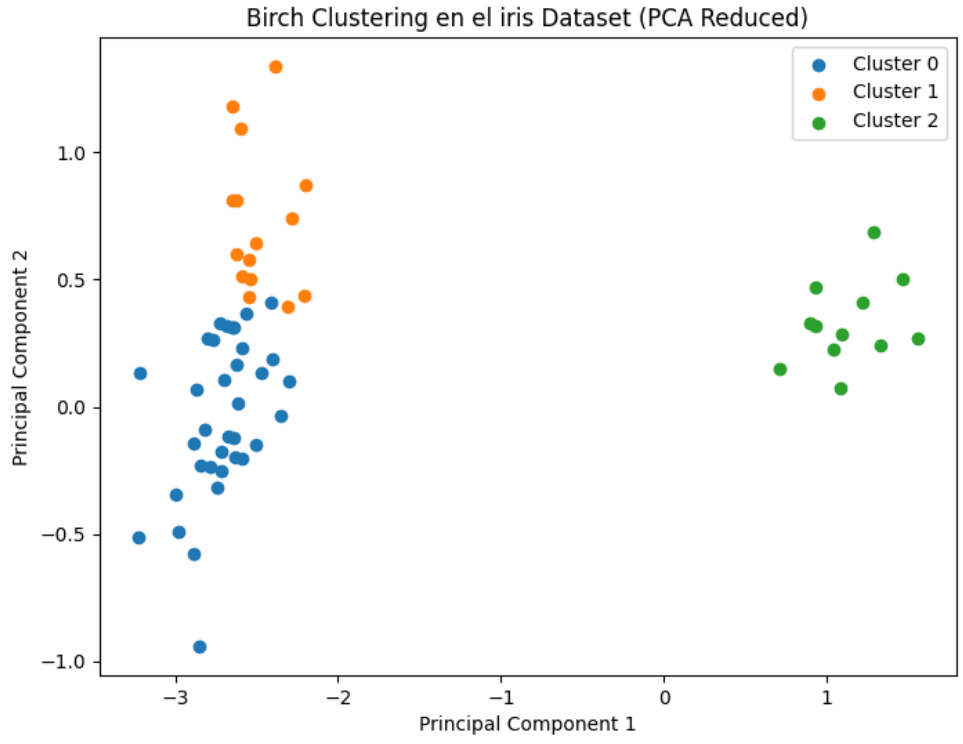


*Ilustración 5. Birch Clustering en Iris Dataset*

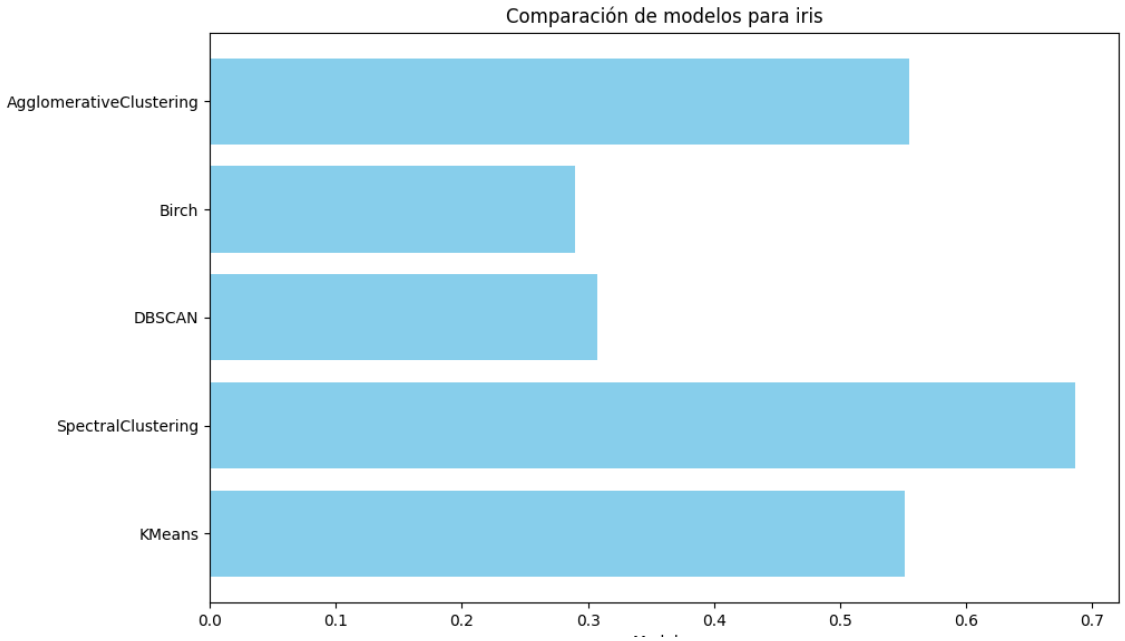


*Ilustración 6. Comparación de modelos en Iris Dataset con silhouette_score*

Como se puede apreciar en el gráfico de barras anterior, el mejor modelo de agrupamiento según la métrica utilizada fue spectral clustering, mientras que kmeans y DBSCAN fueron los peores.

## Mall customers dataset


*Ilustración 7. Mall customers reducido con PCA*

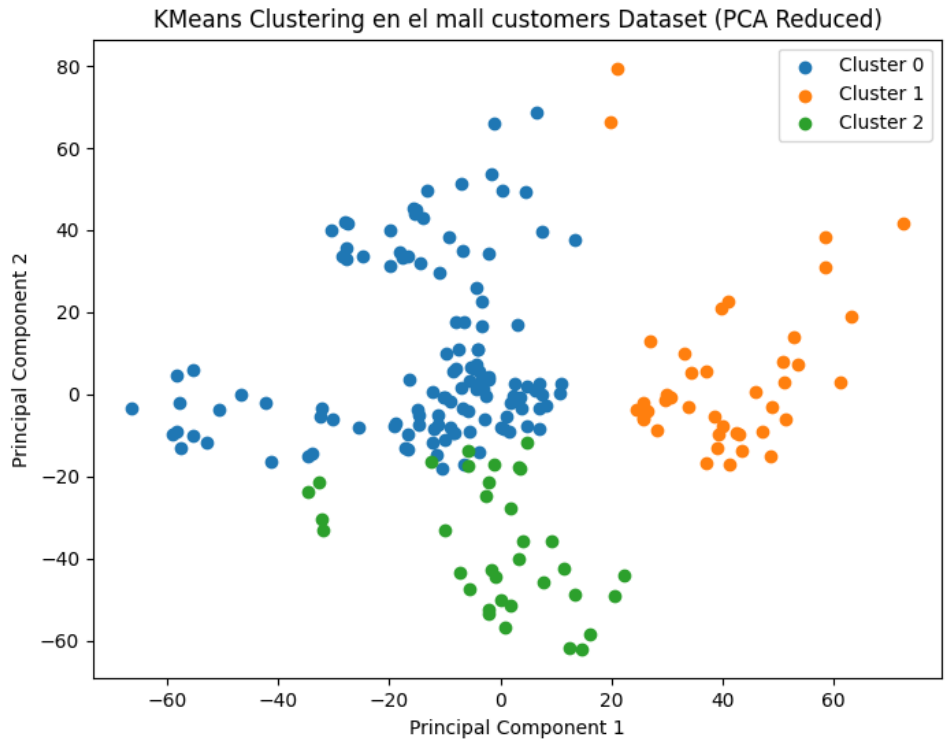


*Ilustración 8. Kmeans en mall customers Dataset*

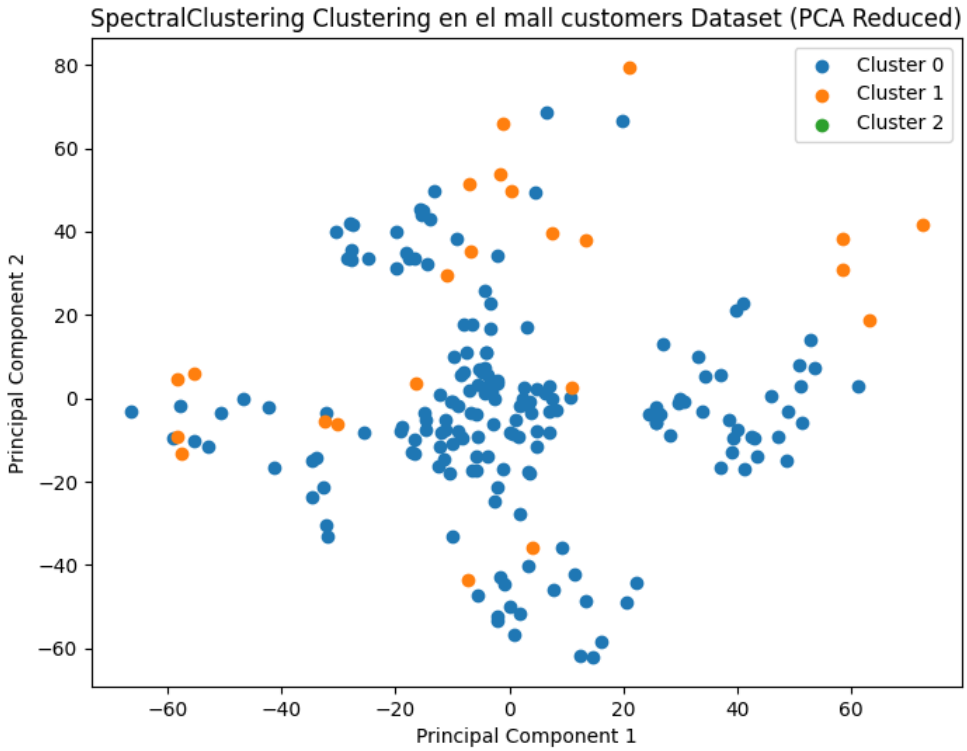


*Ilustración 9. Spectral Clustering en Mall Customers Dataset*

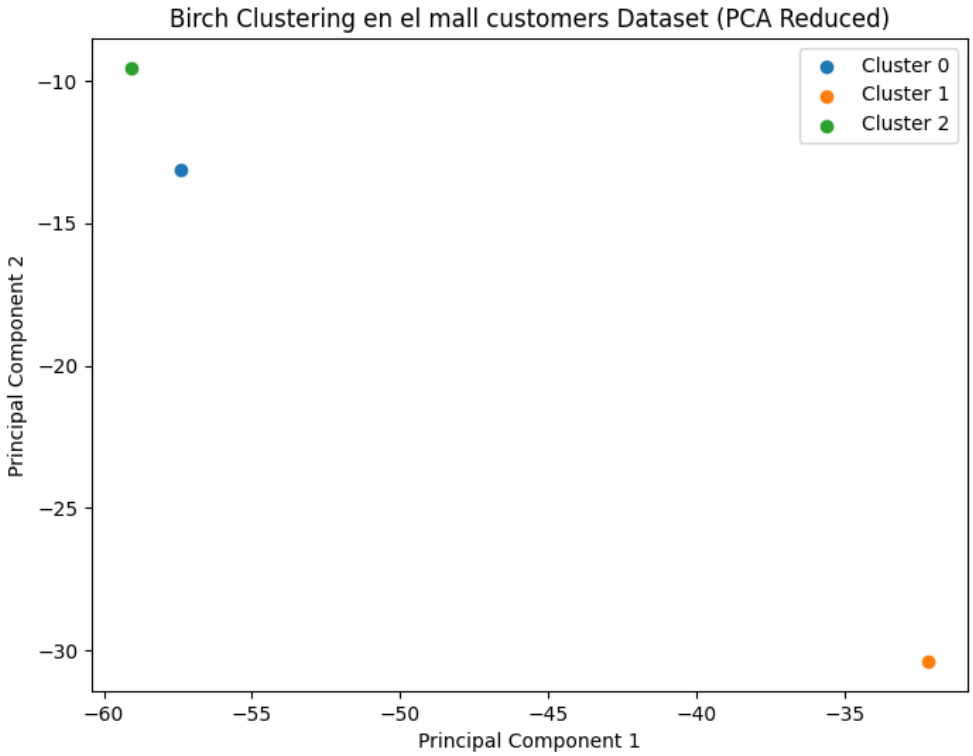


*Ilustración 10. Birch clustering en Mall Customers Dataset*

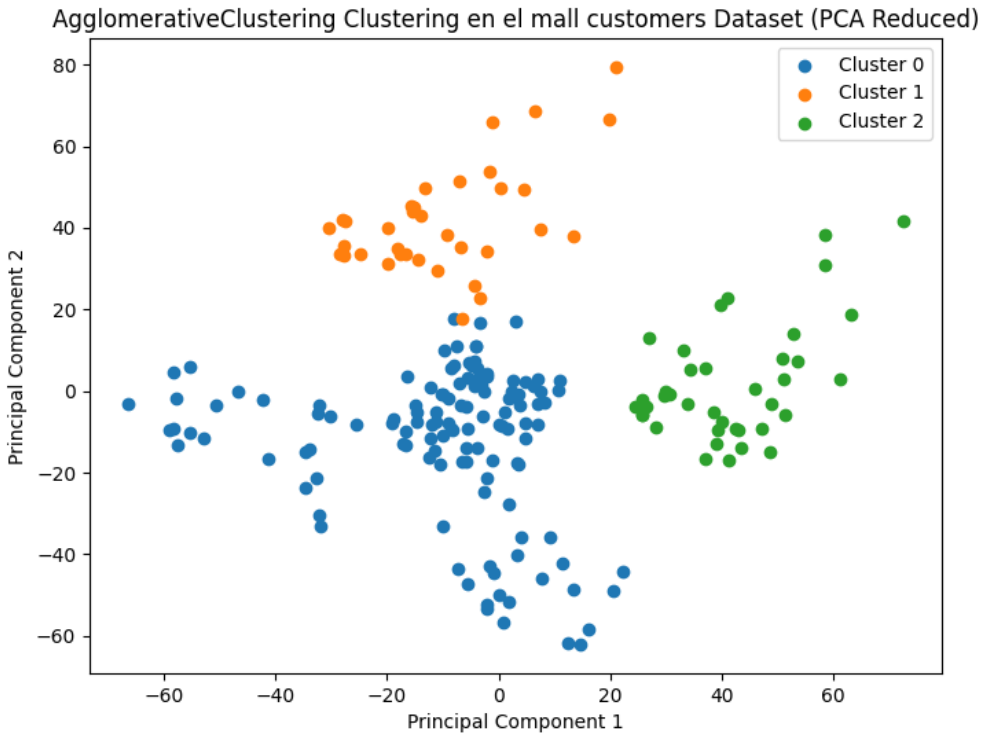


*Ilustración 11. Agglomerative clustering en Mall Dataset*

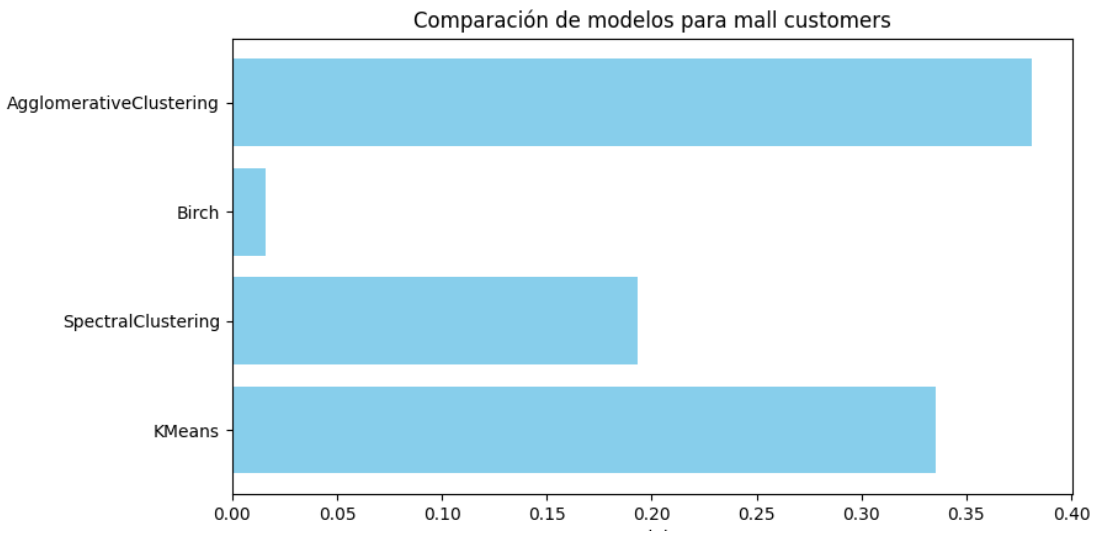


*Ilustración 12. Comparación de resultados en Mall Customers*

Para este dataset el agrupamiento por DBSCAN no fue considerado porque no logró identificar varias categorías, dejando dicho modelo de lado, el algoritmo de k-means tuvo un excelente desempeño mientras que birch dejó mucho que desear.

## Blobs dataset

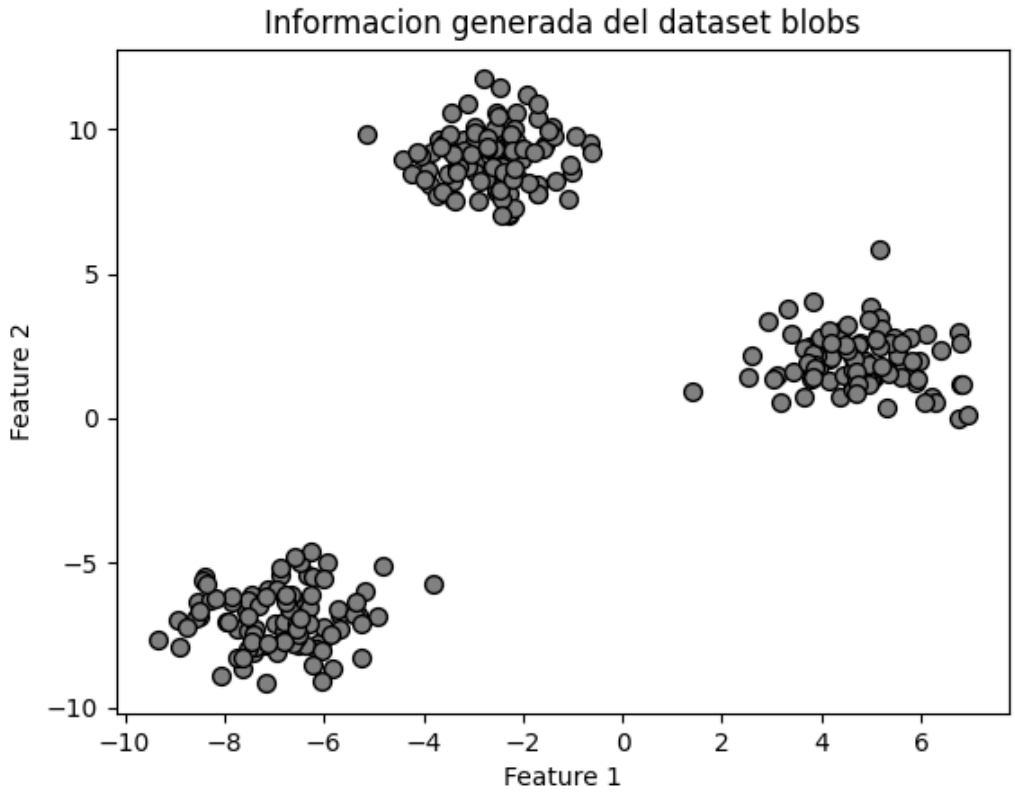


*Ilustración 13. Dataset generado por make_blobs*

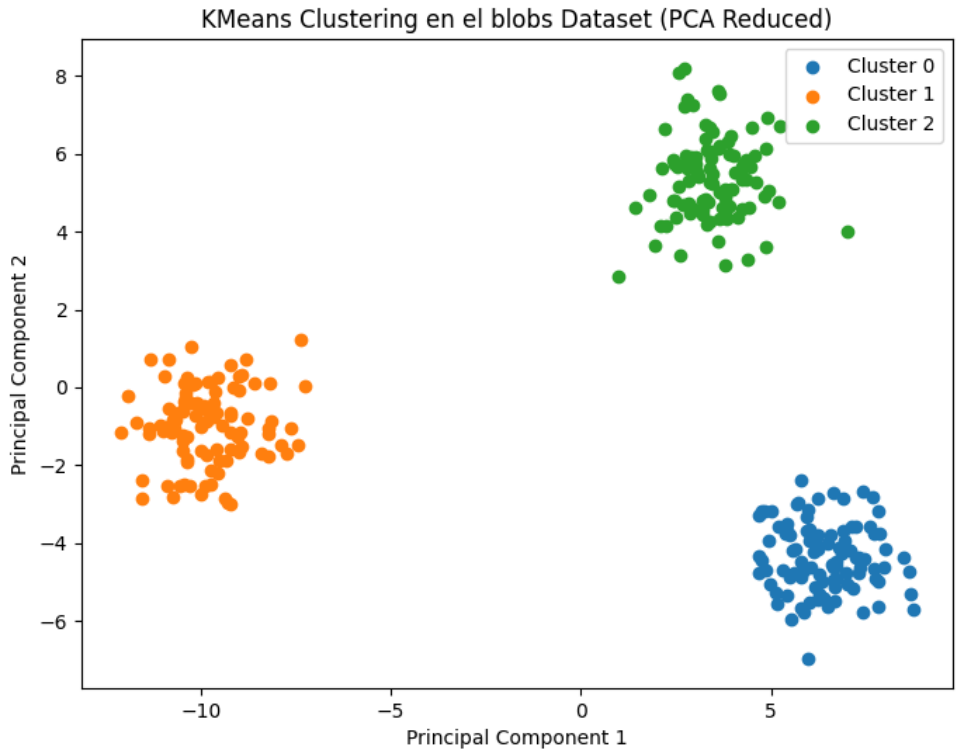


*Ilustración 14. Kmeans en blobs dataset*

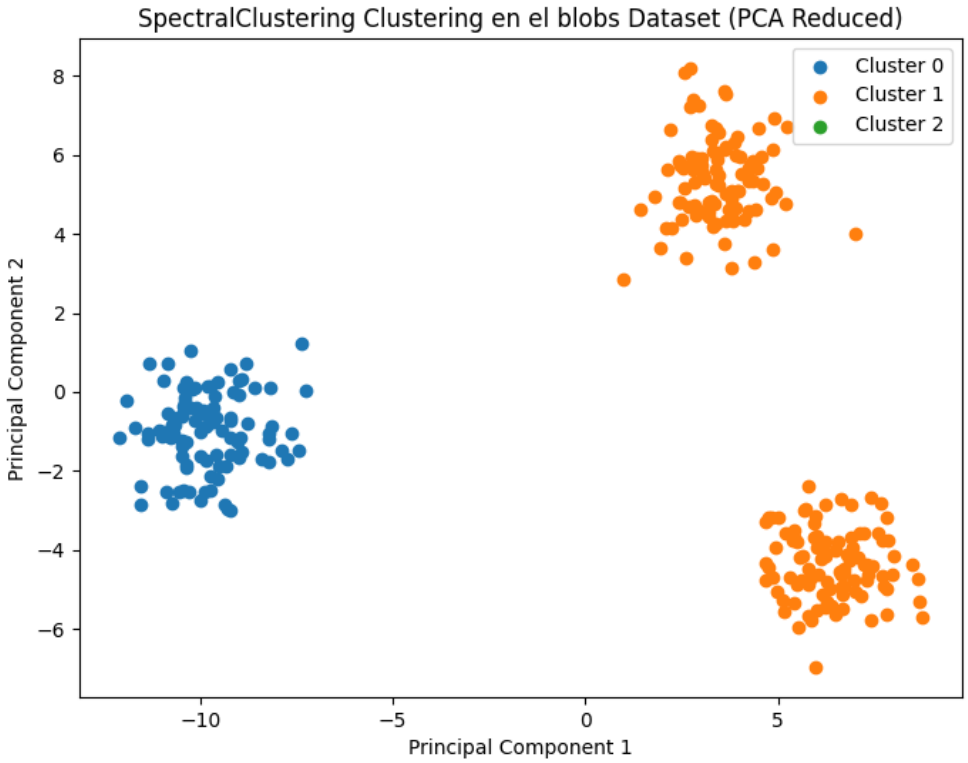


*Ilustración 15. Spectral clustering en Blobs Dataset*

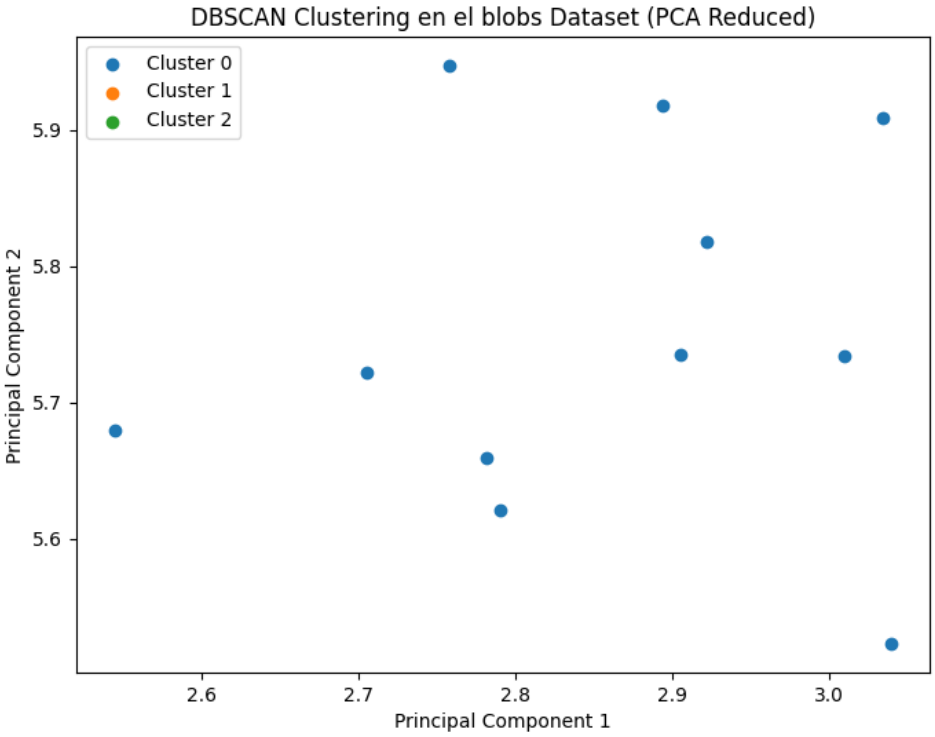


*Ilustración 16. BSCAN clustering en Blobs Dataset*

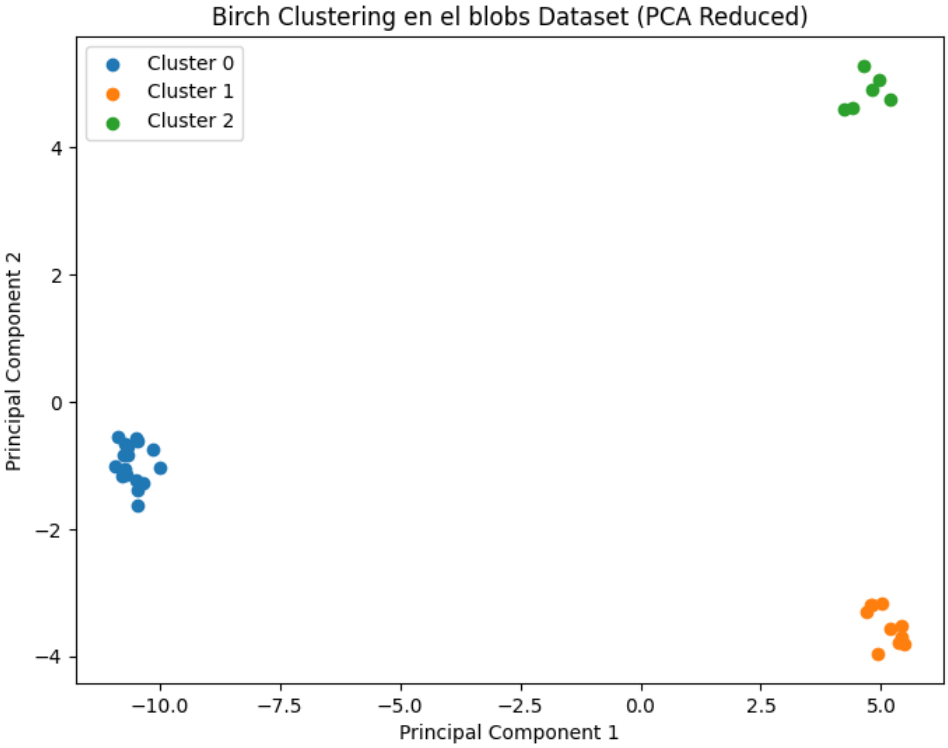


*Ilustración 17. Birch clustering en Blobs Dataset*

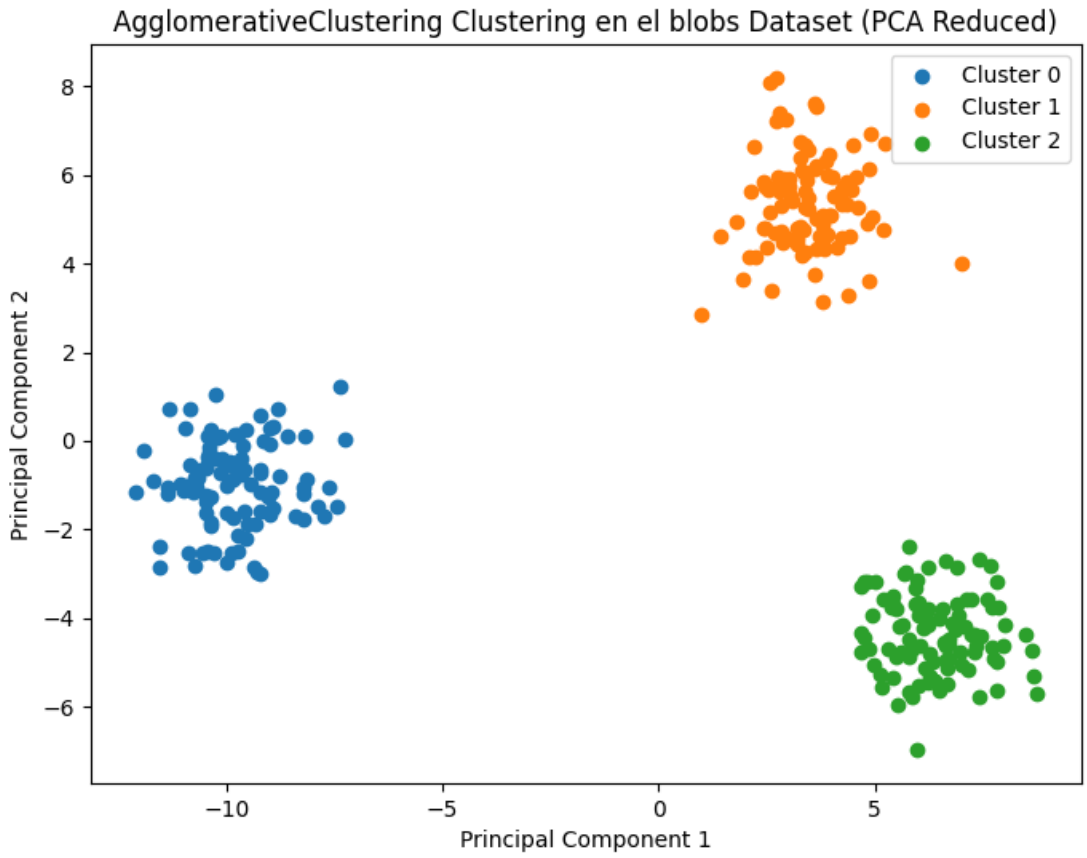


*Ilustración 18. Agglomerative clustering en blobs Dataset*

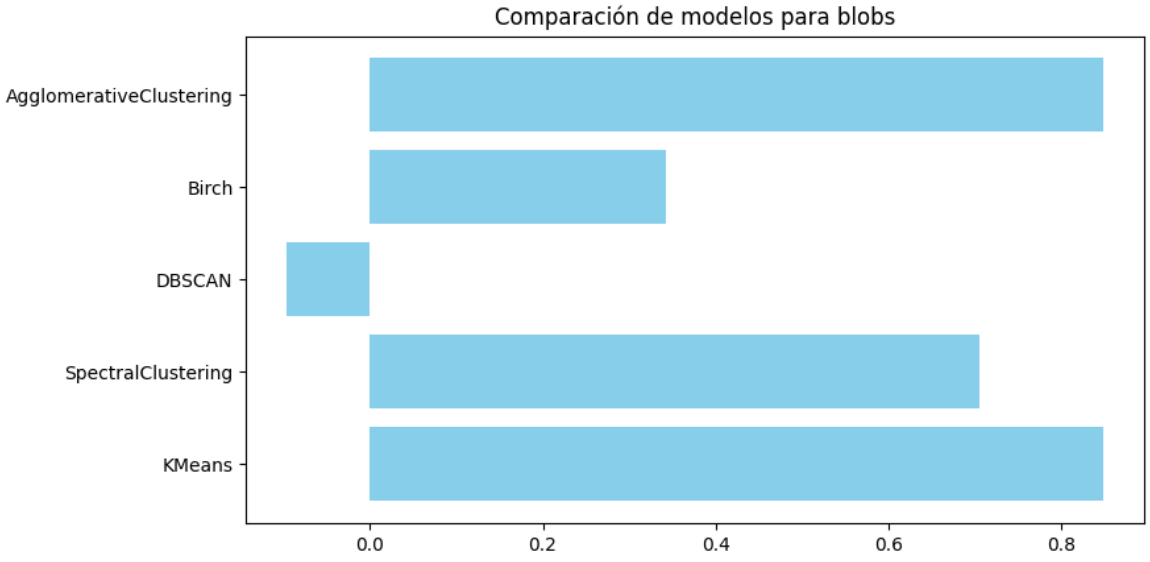


*Ilustración 19. Comparación de modelos en Blobs Dataset*

Para este dataset DBSCAN fue claramente el algoritmo de agrupamiento menos apropiado, sobre todo porque lo comparamos con k-means y algomerative clustering que tuvieron desempeños bastante positivos. Esto último se debe en gran medida a que la desviación estándar que se le dio a la función make_blobs fue bastante pequeña (de 1), por lo que los puntos no se separan mucho dentro de la misma clase.

## Moon dataset

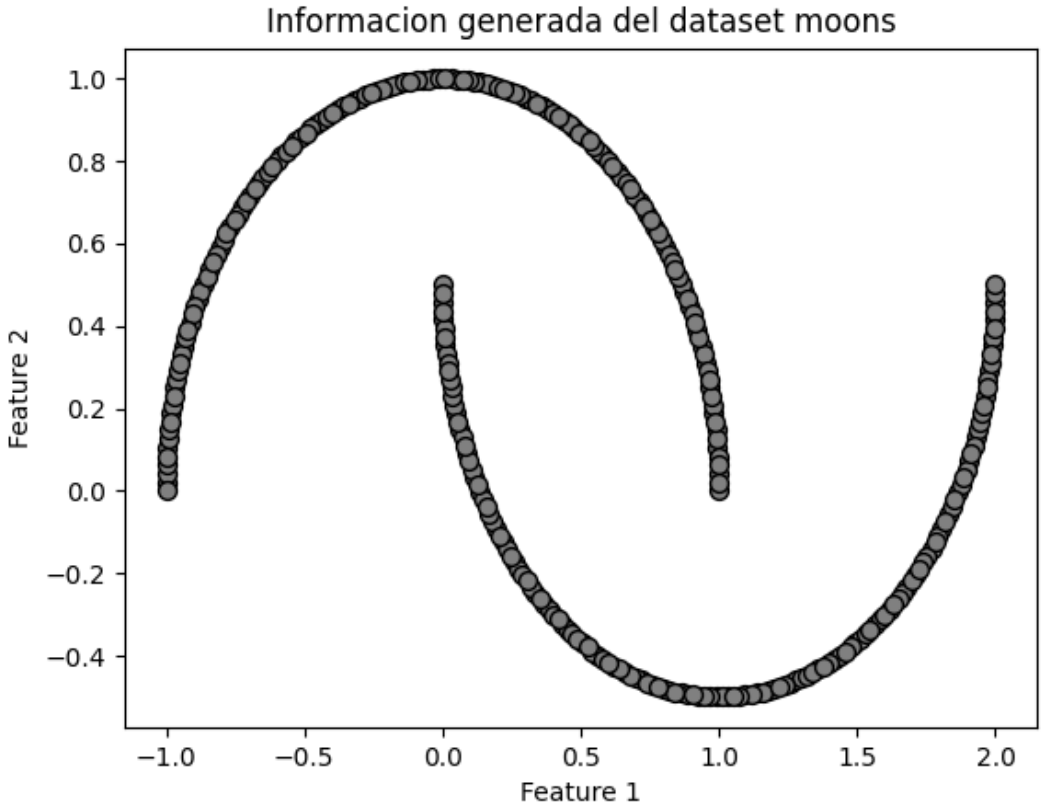


*Ilustración 20. Dataset generado por Moons Dataset*

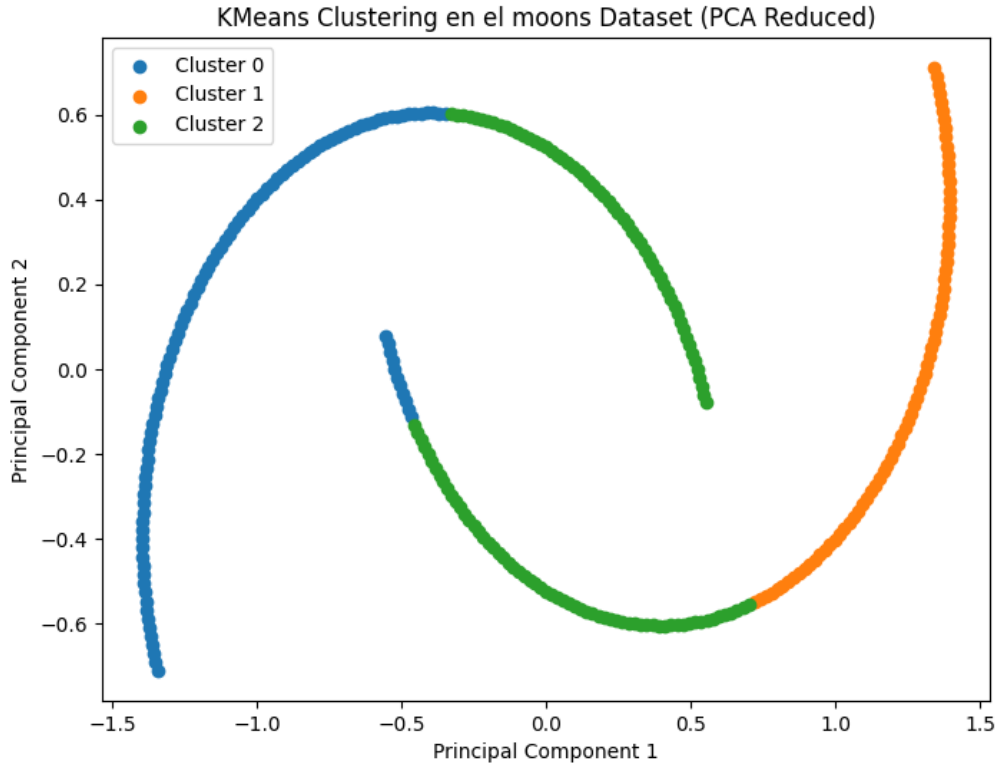


*Ilustración 21. KMeans en Moons Dataset*

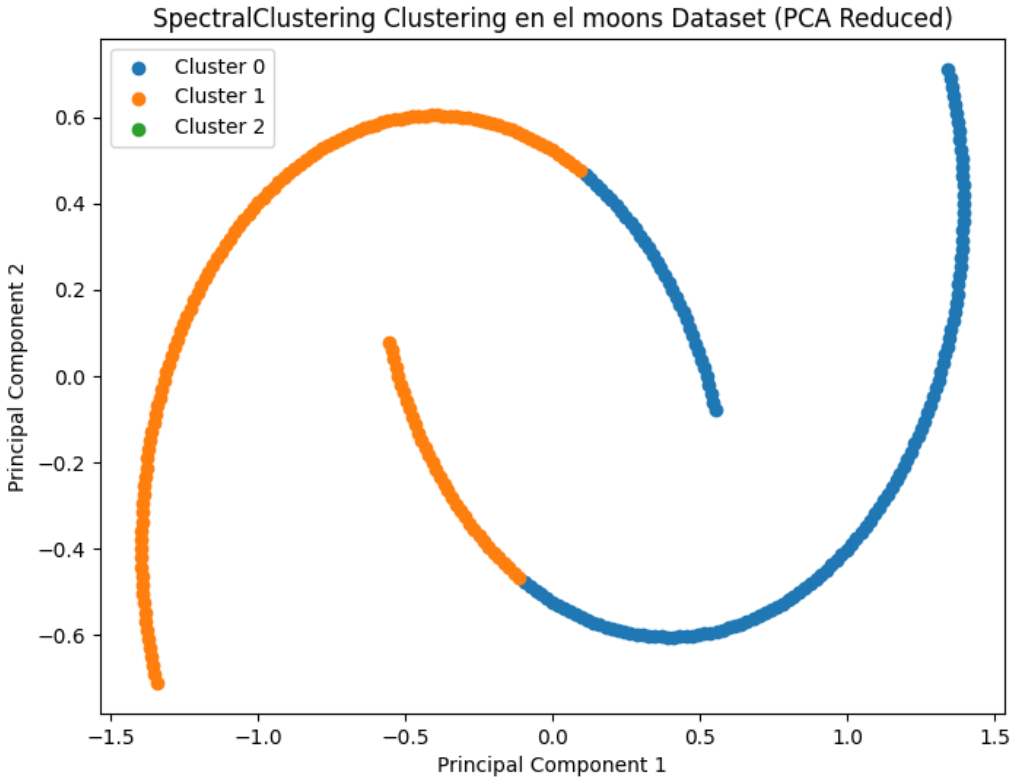


*Ilustración 22. Spectral clustering en Moons Dataset*

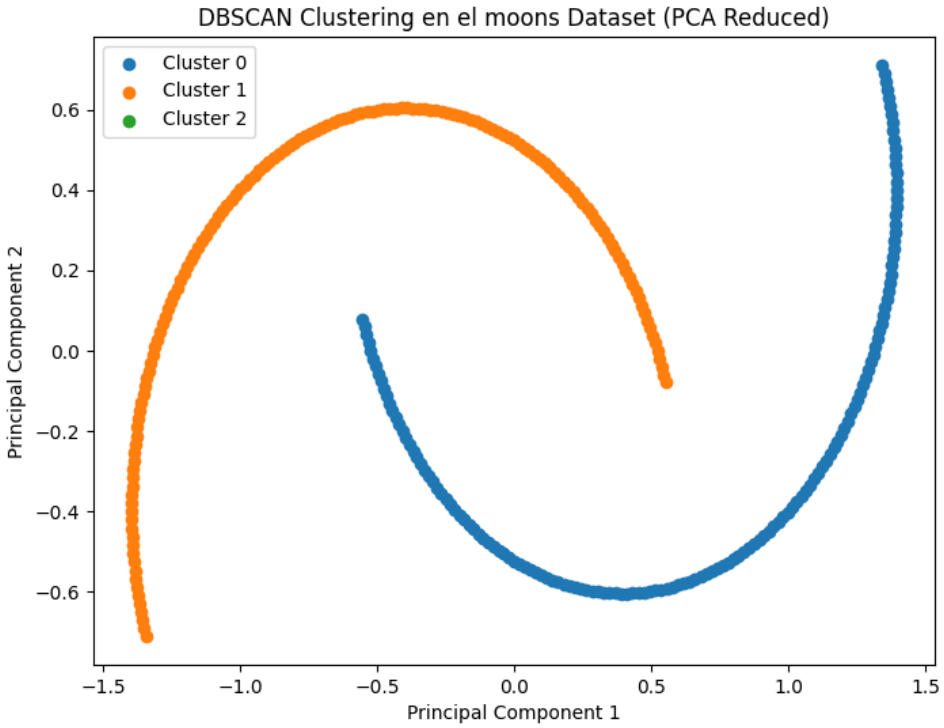


*Ilustración 23. DBSCAN clusterin en Moons Dataset*

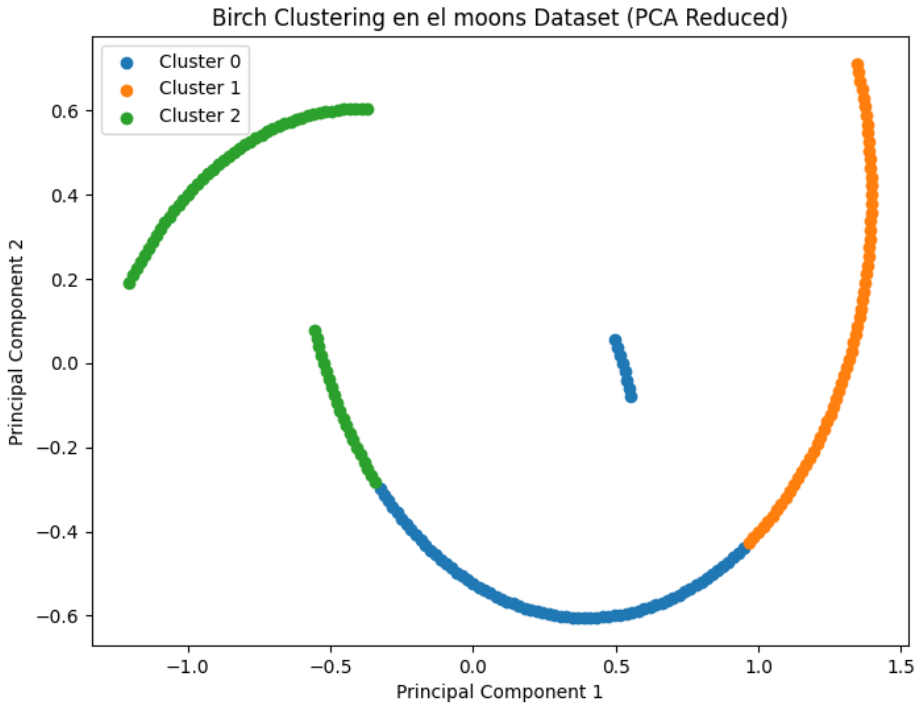


*Ilustración 24. Birch clustering en Moons Dataset*

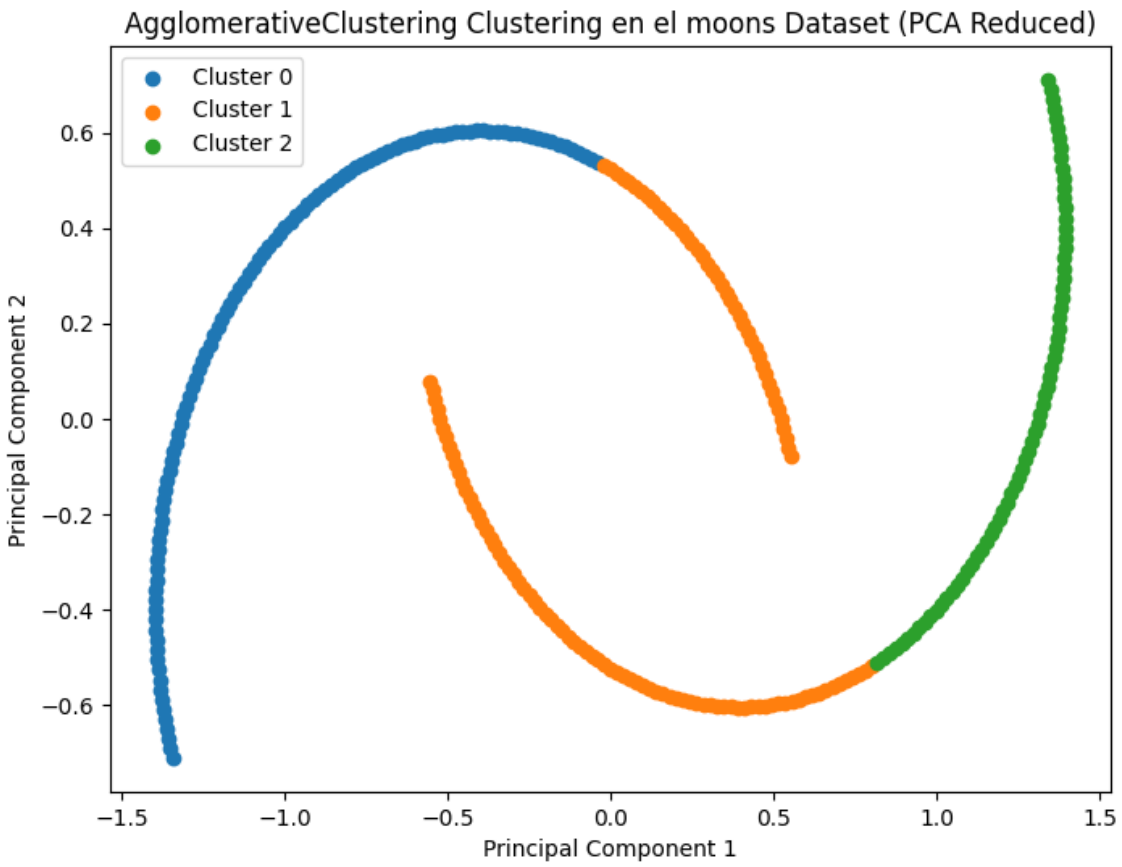


*Ilustración 25. Agglomerative clustering en Moons Dataset*

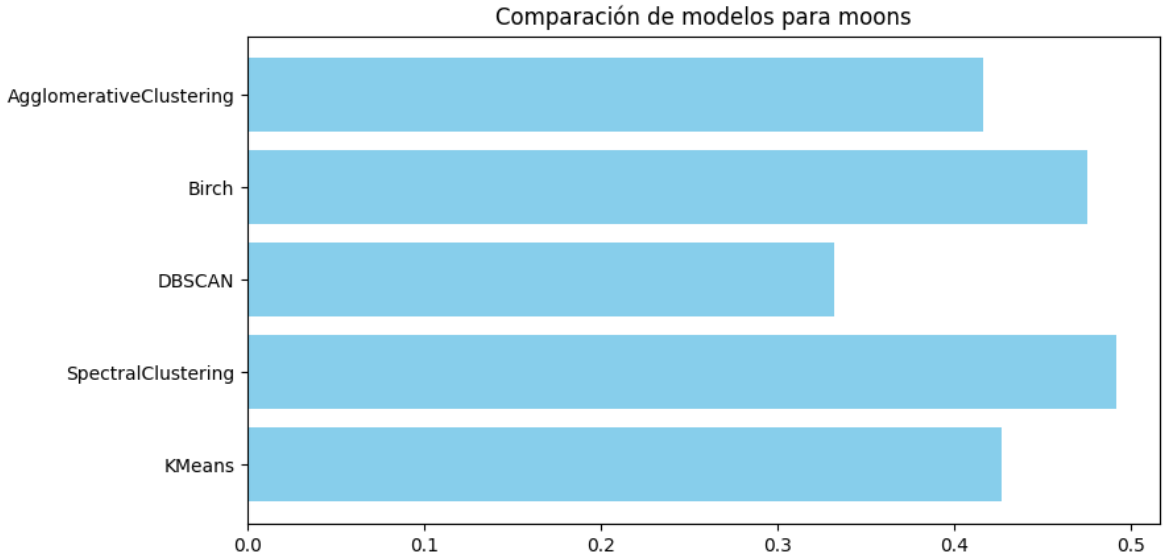


*Ilustración 26. Comparación de modelos para Moons Dataset*

# Conclusiones

La presente práctica nos introdujo en el aprendizaje no supervisado con scikit-learn, más específicamente en reducción de dimensionalidad y clustering. A grandes rasgos podemos decir que un modelo de agrupamiento es un algoritmo de aprendizaje no supervisado que divide un conjunto de datos basándose en las similitudes y diferencias que tienen los datos. Para medir el rendimiento de esta clase de modelos tomamos una métrica que nos indica, de algún modo, que tan similares son los datos agrupados y que tan diferente es cada dato con respecto a los que se encuentran en otras categrorías.

La reducción de dimensionalidad se mostró como una técnica fundamental en la visualización de datos, el uso del algoritmo de análisis de componentes principales nos permitió visualizar la información, así como disminuir el tiempo necesario para ejecutar los subsecuentes algoritmos.

La reducción de dimensionalidad es una técnica para simplificar conjuntos de datos al reducir el número de características o variables, preservando al mismo tiempo la mayor parte de la información relevante. Los beneficios de aplicar esta técnica fueron claramente visibles en el desarrollo de esta práctica, nosotros identificamos que en primer lugar mejora el rendimiento, ya que reduce la complejidad y el tiempo de entrenamiento de los modelos, lo que lleva a un mejor rendimiento; el segundo punto es que facilita la visualización, poque permite visualizar datos complejos en 2D o 3D, lo que facilita su análisis y comprensión. Además de lo anterior también reduce la redundancia al eliminar características irrelevantes o redundantes, lo que simplifica el conjunto de datos.

Pero el paradigma anterior también tiene algunas desventajas, entre estas están la pérdida de información (puede haber una pérdida de datos durante el proceso, lo que podría afectar el funcionamiento de algunos algoritmos), y otra es la dificultad de interpretación, es decir, puede ser complicado comprender la relación entre las características originales y las nuevas dimensiones reducida.

El clustering sirve para agrupar datos similares en grupos (clústeres) basados en sus características comunes, sin conocer previamente las etiquetas de los datos. Su objetivo es descubrir patrones ocultos, maximizando la similitud dentro de cada grupo y minimizándola entre grupos distintos. Los algoritmos de agrupamiento, al trabajar con datos no etiquetados, no tienen una respuesta correcta predefinida. En estos algoritmos para agrupar los datos, lo primero que se debe de hacer es definir cómo se medirá la similitud entre ellos. Esto se hace a través de métricas de distancia, como la distancia euclidiana o la distancia de Manhattan, para después continuar con la formación de grupos, en donde los datos se organizan en grupos o clusteros donde los puntos dentro de cada grupo son lo más parecidos posible entre sí, mientras que los puntos de diferentes grupos son lo más diferentes posible.cEl clustering se muestra como una técnica eficaz en: segmentación de clientes, procesamiento de documentos, biología, detección de anomalías y análisis de datos, siendo una herramienta para descubrir patrones y obtener información útil de datos sin procesar.

# Referencias

2.3. Clustering. (s. f.). Scikit-learn. https://scikit-learn.org/stable/modules/clustering.html

Murel, J., PhD, & Kavlakoglu, E. (2025, 18 febrero). Reducción de la dimensionalidad. *Artificial Inteligence*. https://www.ibm.com/mx-es/think/topics/dimensionality-reduction

P, P. (2024, 27 septiembre). *What is Dimensionality Reduction? A Guide.* Roboflow Blog. https://blog.roboflow.com/what-is-dimensionality-reduction

Smolic, H. (2024, 5 septiembre). Clustering Model in ML. *Graphite Note*. https://graphite-note.com/clustering-model-in-ml

Navarro, S. (2024, 18 abril). ¿Qué es el clustering o agrupamiento en machine learning? *KeepCoding Bootcamps*. https://keepcoding.io/blog/que-es-clustering-o-agrupamiento/