Proyecto de aprendizaje no supervisado

Proyecto Final

Índice

Contenido

Introducción

Describe los datos, es decir, que información contiene y de donde obtuviste la base de datos que vas a utilizar. Incluye una descripción del problema (preguntas que quieren resolver).

La base de datos utilizada fue descargada de la plataforma kaggle (Figura 1) y los datos provienen originalmente del Instituto Nacional de Diabetes y Enfermedades Digestivas y Renales de Estados Unidos. El objetivo es predecir en base a medidas diagnósticas si un paciente tiene diabetes, por lo que se incluyen 768 observaciones (con una usabilidad del 100%) constituidas por 8 características y dos etiquetas o posibles resultados. Adicionalmente, el proveedor de los datos puntualiza que se impusieron ciertas restricciones para la selección de las instancias de esta base de datos de una más grande. En particular, todos los pacientes son mujeres de al menos 21 años, de ascendencia indígena pima.

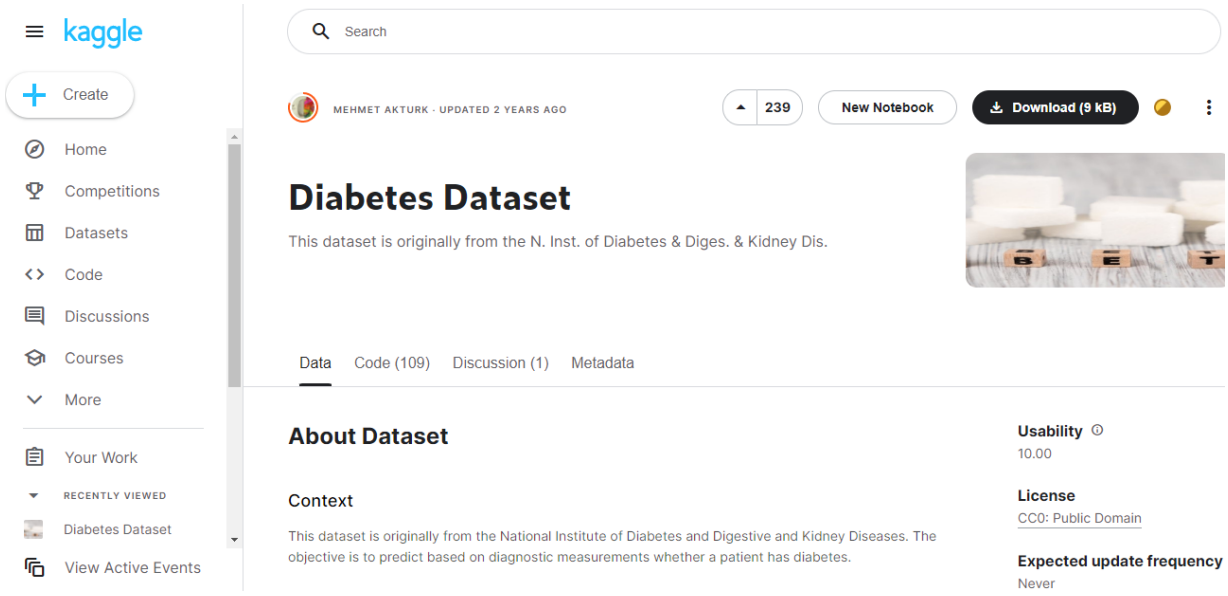


Figura 1. Base de datos de diabetes obtenida en kaggle

Características

Embarazos: Número de veces embarazada

Glucosa: Concentración de glucosa en plasma a las 2 horas en una prueba de tolerancia oral a la glucosa

Presión arterial: Presión arterial diastólica (mm Hg)

Grosor de la piel: Grosor del pliegue cutáneo del tríceps (mm)

Insulina: insulina sérica de 2 horas (mu U/ml)

IMC: Índice de masa corporal (peso en kg/(altura en m)^2)

DiabetesPedigreeFunction: Función de pedigrí de diabetes, puntuación basada en el historial familiar

Edad: Edad (años)

Resultado: Variable de clase dependiendo si se tiene diabetes o no (0 o 1)

Descripción del problema

La diabetes mellitus constituye uno de los principales problemas de salud en el mundo, ya que hay cerca de 100 millones de diabéticos en el planeta. La prevalencia de esta enfermedad está incrementándose de forma importante en las poblaciones en vía de desarrollo debido al envejecimiento de la población, el cambio de hábitos dietéticos (mayor consumo de azúcares refinados) y un descenso de la actividad física, lo que también conlleva a un aumento de las personas obesas (González, 2011). Además, muchas de las personas que sufren la enfermedad desconocen su situación, como ocurre con muchas personas obesas.

En el presente trabajo se busca entrenar modelos de machine learning para poder ser aplicados en el campo de la medicina, específicamente facilitar el diagnóstico de personas con diabetes. Se busca llegar a un diagnóstico lo más preciso posible dadas las medidas diagnósticas de un paciente.

Créditos

Nombre y puesto de los integrantes. (Analista, programador, data scientist…)

Gerardo Moreno Zizumbo: Programador

Cristina López Ontiveros: Data Scientist

Diego Alejandro Flores Meza: Data Scientist

Sofía Ingigerth Cañas Urbina: Analista

Método 1: K-meidas

K-medias - Diego Flores

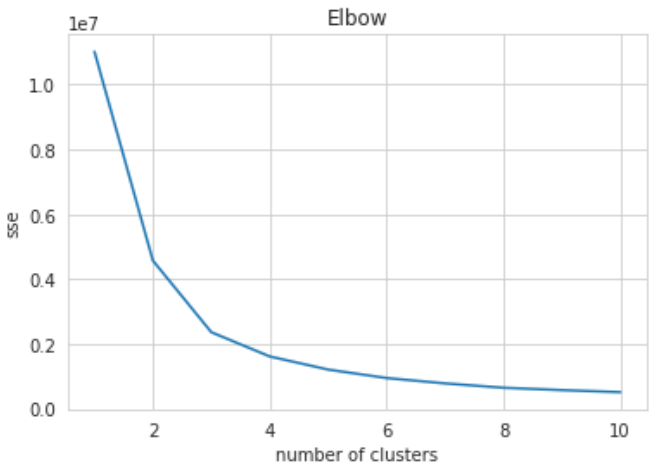
Este modelo de aprendizaje no supervisado tiene la función de clasificar agrupando en k número de grupos según son las características proporcionadas. El algoritmo cuenta de tres pasos; la inicialización donde se escoge el número de grupos k y los centroides k en el espacio de los datos, el segundo paso es la asignación de los objetos al centroide que se encuentre más cercano y el último paso es la actualización del centroide al nuevo promedio de los objetos. Estos dos últimos pasos se repiten hasta que el centroide deje de moverse o su movimiento sea casi nulo.

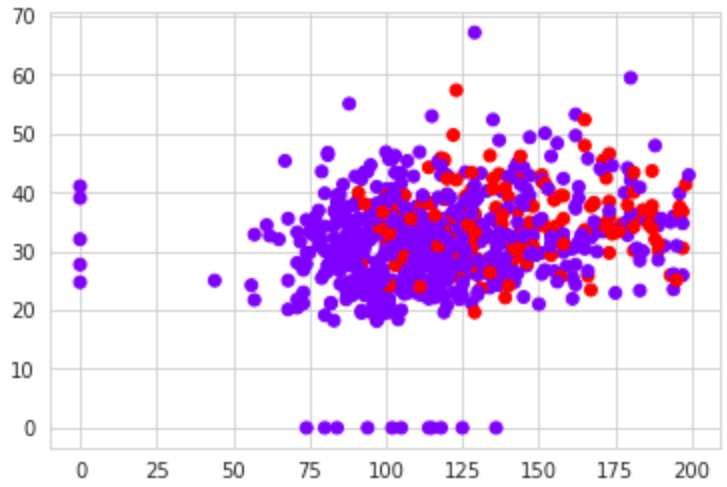
Objetivo

El objetivo principal del algoritmo K-Means es minimizar la suma de distancias entre los puntos y su respectivo centroide de clúster para clasificar los objetivos.

Implementación

Para poder implementar este algoritmo fue necesario cargar las librerías de pandas, numpy y de la librería de sklearn importe KMeans desde el cual se establece el número de clusters y el random state de la función. Después procedí a seleccionar las columnas que usaría en el modelo e hice una escalada de los datos de las mismas columnas. Realice la gráfica de codo para poder determinar el número de clusters que serían necesarios para el modelo y por último grafique el modelo k-medias.





Evaluación

	Precisión	Sensibilidad	Score F1
0	0.66	0.79	0.72
1	0.40	0.26	0.31
Exactitud			0.60

```
from sklearn.metrics import adjusted_rand_score
adjusted_rand_score (data1.Outcome, kmeans.labels_)
```

```
0.04752166384071964
```

```
9] from sklearn.metrics import silhouette_score
silhouette_score(X, Y)
```

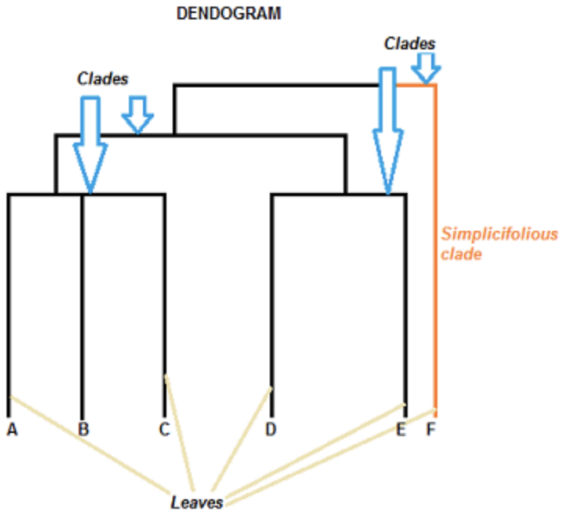
```
0.1151358594875247
```

Método 2: Dendrogramas

Dendrogramas- Cristina López Ontiveros

Un dendrograma es un tipo de aprendizaje por refuerzo comúnmente utilizado por la biología computacional, esto para poder mostrar la agrupación de genes o muestras. Esto se debe a que, como menciona correctamente la página de Statistics How To; “Un dendrograma es un tipo de diagrama de árbol que muestra un agrupamiento jerárquico: relaciones entre conjuntos de datos similares. Se utilizan con frecuencia en biología para mostrar la agrupación entre genes o muestras, pero pueden representar cualquier tipo de datos agrupados.” Asimismo cabe recalcar que un dendrograma es un agrupamiento jerárquico aglomerativo.

Las principales partes de un dendrograma son las siguientes;



En un dendrograma se tienen 2 partes principales:

- Codos: Estas son las ramas, y comúnmente se les etiqueta con letras griegas como α, β o δ. Este tipo de codos se pueden dividir en diversos tipos dependiendo del número de hojas con el que este cuente, en este caso tenemos de 3 tipos.
    - Simple (simplicifolius): Con una sola rama que es la F
    - Doble (bífolio): Con 2 ramas que son la D y E
    - Triple (trifolio): Que es el ABC
- Resultados

Cabe mencionar que también en un dendrograma entre más larga sea la distancia entre esquinas, mayor es la diferencia en términos de características.

Objetivo

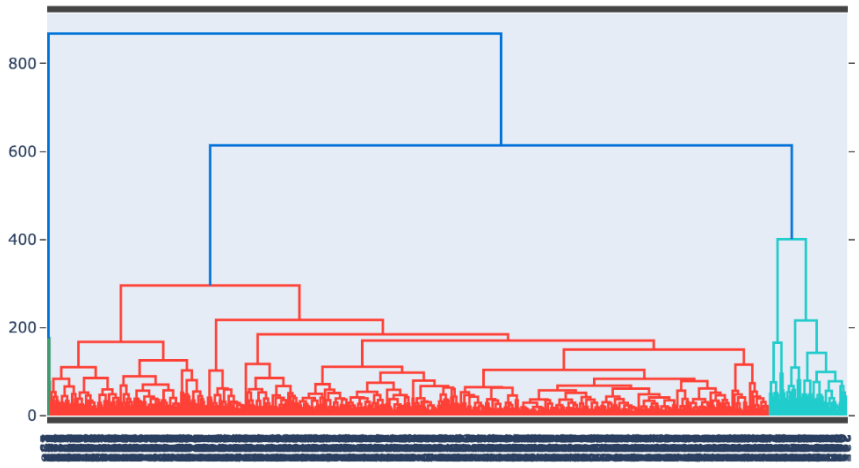
Visualización para el agrupamiento jerárquico de nuestra base de datos

Implementación

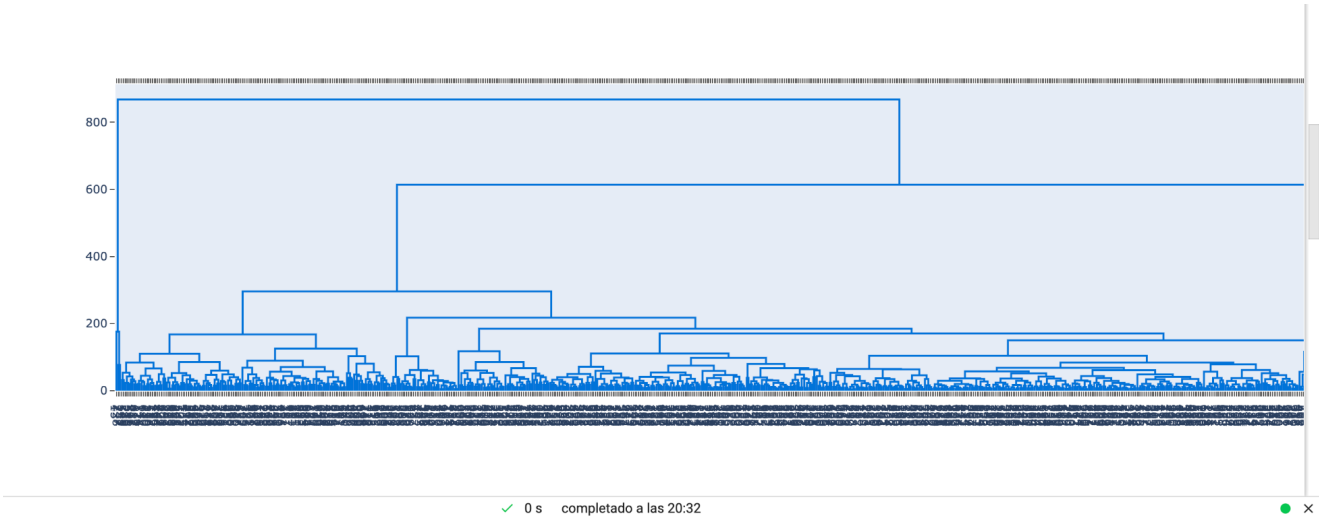
En este proyecto para poder jerarquizar la información de nuestra base de datos de la diabetes utilizando dendrogramas, pero en nuestro caso utilizamos tres tipos de dendrogramas los cuales fueron; diagrama básico, dendograma con colores y dendograma con etiquetas. Para la elaboración de este código se utilizaron para todos los dendrogramas las librerías de ***plotly.figure_factory as ff,*** ***numpy as np*** y ***pandas as pd***. Posteriormente importamos los datos al programa para poder darles un parámetro X y Y, para poder usar luego la función ***ff.create_dendrogram*** para crear cada una de las figuras para cada uno de los casos y en los paréntesis vamos a poner las especificaciones de tamaño, color, etc.

Finalmente, nos da esto para cada una de nuestras especificaciones:

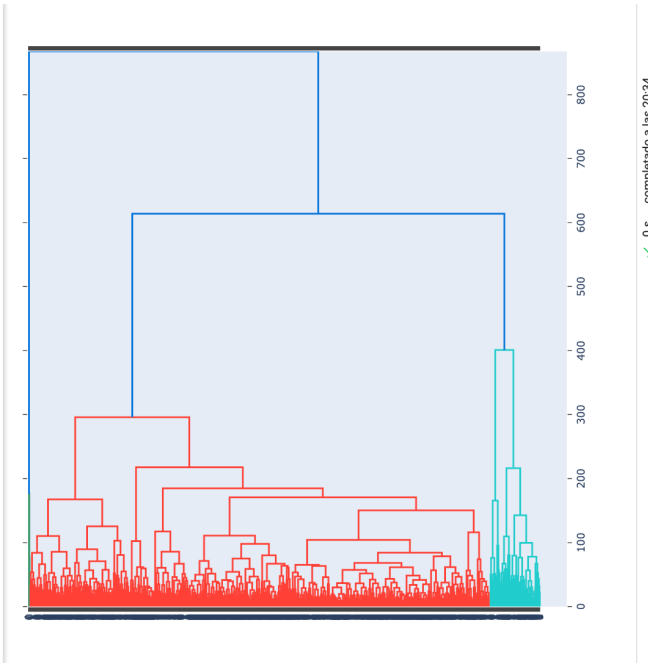
- Dendrograma básico



- Dendrograma con colores



➢ Dendrograma con etiquetas



Evaluación

Evaluación

```
from sklearn.metrics import silhouette_score
silhouette_score(X, Y)
```

```
0.10426011347558613
```

Método 3: Spectral Clustering

Explicación del método - Sofía Cañas

Spectral clustering es una técnica con raíces en la teoría de grafos, donde el enfoque que se utiliza es identificar comunidades de nodos en un grafo en función de los bordes que los conectan, por medio del aprendizaje no supervisado. El método es flexible y también nos permite agrupar datos no graficables. Específicamente, spectral clustering utiliza información de los eigenvalores o valores propios (espectro) de matrices especiales construidas a partir del gráfico o el conjunto de datos (Fleshman, 2019).

Cabe mencionar que técnicas de agrupación como K-Means, que se basan en la proximidad euclidiana, pueden no ser capaces de agrupar datos que se encuentran en una variedad de alta dimensión, como se ilustra en la Figura 3. Es por eso que spectral clustering es prometedor para descubrir agrupamientos ocultos a partir de datos que están conectados pero no necesariamente compactos dentro de límites convexos (IBM, 2018).

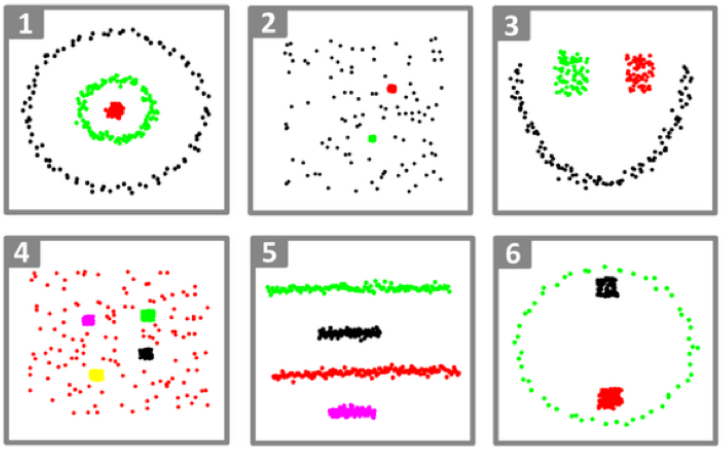


Figura 3. Datos que se encuentran en variedades (espacios topológicos) de muchas dimensiones

Internamente, para realizar spectral clustering hay tres pasos principales (Figura 4):

1. Crear un grafo de similitud entre nuestros N objetos a agrupar.
2. Calcular los primeros k vectores propios de su matriz laplaciana para definir un vector de características para cada objeto.
3. Ejecutar k-means en estas características para separar objetos en k clases.

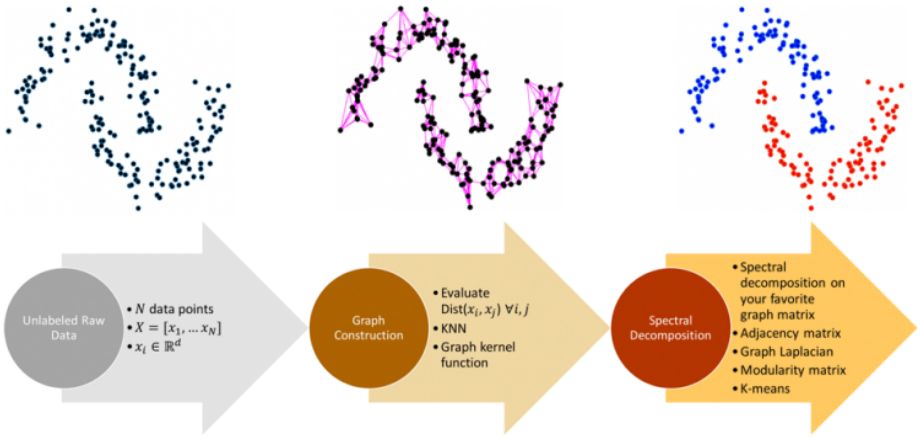


Figura 4. Pasos de spectral clustering

Hay diferentes formas de construir un grafo que represente las relaciones entre puntos de datos:

- Grafo de vecindad ε: cada vértice está conectado a vértices que caen dentro de una bola de radio ε donde ε es un valor real que debe ajustarse para capturar la estructura local de los datos.

- Grafo de k-vecinos más cercano: cada vértice está conectado a sus k-vecinos más cercanos, donde k es un número entero que controla las relaciones locales de los datos.

Objetivo

El objetivo es la agrupación de los datos (clustering), que es una de las tareas principales en el aprendizaje automático no supervisado. De modo que el modelo asigne datos sin etiquetar a grupos, donde se espera que puntos de datos similares se asignen al mismo grupo, en este caso serían dos grupos porque nuestra pregunta es si la persona tiene diabetes o no entonces se espera que los grupos resultantes nos puedan ayudar a responder esto.

Implementación

Para el entrenamiento de este modelo se importa la librería Scikit-Learn. Como preprocesamiento con StandardScaler() se implementa una estandarización, escalando los valores de cada característica numérica en su conjunto de datos para que las características tengan una media de 0 y una desviación estándar de 1.

Con SpectralClustering en scikit-learn se pueden establecer los parámetros del algoritmo antes de ajustar el estimador a los datos. La implementación de scikit-learn es flexible y proporciona varios parámetros que se pueden ajustar.

Por último, el algoritmo se evalúa con adjusted_rand_score.

Evaluación

Adjusted Rand Score	Valores sin procesamiento	0.0025
	Valores con escala estándar	0.9668

Método 4: BIRCH

Explicación del método - Gerardo Moreno

“Balanced iterative reducing and clustering using hierarchies” o BIRCH por sus siglas en inglés, es un modelo de aprendizaje no supervisado del tipo clustering jerárquico. El funcionamiento del modelo está basado en guardar información en una estructura de tipo árbol, dicho árbol se le conoce como Clustering Feature Tree (CF-Tree) y la información que almacena se le conoce como Clustering Features (CF), un CF es un triplete de valores representativos del Cluster que dicho componente del árbol alberga, es decir, una nodo raíz tendrá un CF representativo de todos sus subnodos y los respectivos clusters que albergan ellos y por el otro lado, un nodo de tipo hoja tendrá un CF representativo únicamente de los clusters que albergan. El guardar los CF de esta manera nos permite leer los puntos de datos una sola vez y no necesitar continuar leyéndolos cada que se hace una iteración de mejora de clusters. Esto es posible matemáticamente debido a que se pueden calcular las métricas necesarias para evaluar la efectividad de los clusters con la información guardada en los CF.

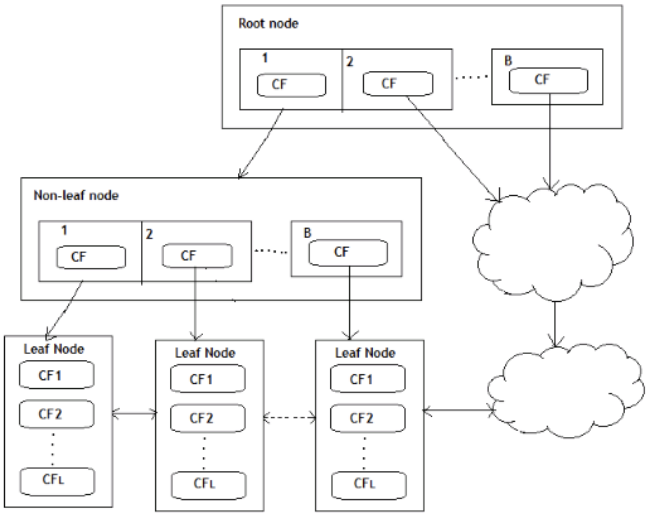


Figura 5. Arquitectura de modelo Birch

Objetivo

El objetivo es usar este algoritmo para agrupar la base de datos. Al probar con diferentes hiper parámetros, entre ellos el radio máximo permitido en subcluster (T) y el número máximo de CF for nodos (B). Nuestra base de datos tiene una variable de predicción binaria, por lo tanto, se espera poder explorar diferentes opciones y obtener una que clasifique en dos clusters como si estuviera clasificando el tener la enfermedad o no tenerla.

Implementación

Para la implementación utilice la librería de Scikit-Learn, modelo “Birch”, de igual forma, utilice Scikit-Learn para importar una librería que hiciera una normalización estándar de la base de datos (StandardScaler) y me tomé la libertad de explorar cuáles serían los resultados si hiciera una reducción de dimensiones con análisis de componentes principales antes de hacer el clustering, de igual forma importando desde Scikit-Learn (PCA).

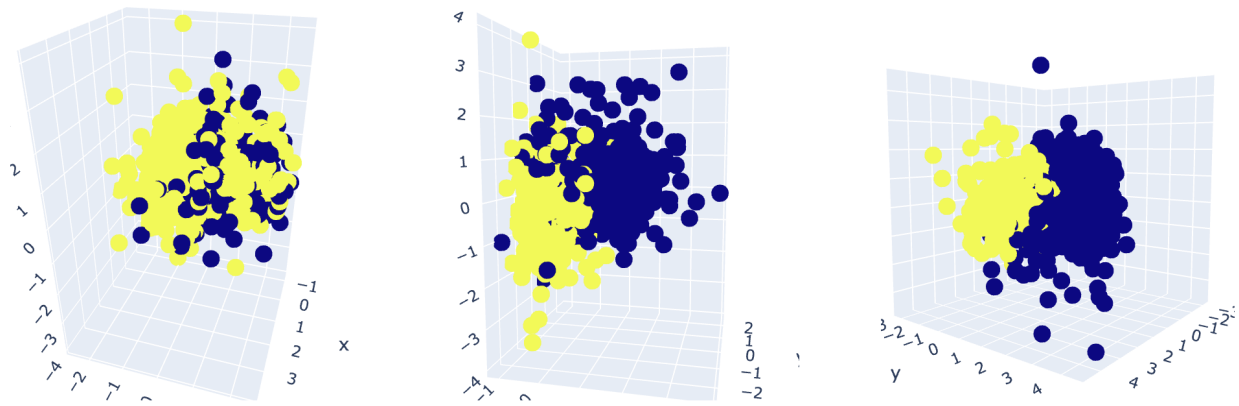


Figura 6. Gráficas al implementar Birch con diferente escalas, de izquierda a derecha: (Sin escalar, Escala estándar, Componentes reducidos (PCA))

Evaluación

Utilizando como métrica de evaluación el Adjusted Rand Score, los resultados fueron los siguientes.

DataSet	Adjusted Rand Score	Silhouette Score
Valores sin procesamiento	0.0773	0.1111
Valores escala estándar	0.4135	0.2053
Valores con reducción de componentes (8 componentes)	0.7417	0.2090

Cabe recalcar que si bien se utilizó análisis de componentes principales, donde el propósito sería reducir el número de dimensiones, en este caso, debido a que gran parte de las variables aportan valor a la variable respuesta, se observó que el implementar PCA con 8 dimensiones (sin reducir dimensiones), mejora la métrica por 34 centésimas por arriba de solo utilizar los datos con escala normal.

Comparación

1) K-Means
    a) Ventajas:
        i) Fácil de implementar
        ii) Velocidad de convergencia rápida
    b) Desventajas
        i) K no es fácil de elegir
        ii) Sensible a puntos anormales

2) Dendrograma:
    a) Ventajas:
        i) Fácil de implementar
        ii) Si se tienen pocos datos es visualmente atractivo
        iii) No se tiene que poner el número de clusters por hacer
    b) Desventajas:
        i) Poco preciso

ii) En caso de tener muchos datos será muy difícil de visualizar el dendograma
iii) Entre más datos, más lento se vuelve

3) Spectral Clustering
    a) Ventajas:
        i) Puede encontrar clusters ocultos por espacios topológicos complejos.
        ii) Es un modelo flexible y ajustable con diferentes parámetros.
        iii) En comparación con otros algoritmos, es computacionalmente rápido para conjuntos de datos dispersos de varios cientos de observaciones.
    b) Desventajas:
        i) Llega a ser computacionalmente complejo como para problemas a gran escala.
        ii) El número de grupos (k) debe especificarse antes de iniciar el procedimiento.

4) Birch
    a) Ventajas:
        i) Eficiente en términos de memoria y tiempo
        ii) Funciona con grandes cantidades de datos
    b) Desventajas
        i) Debido a su estructura de árbol, la solución de clustering depende del orden en el que los puntos de datos son introducidos.
        ii) Solo trabaja con datos numéricos

Conclusiones

Posibles Mejoras

Hemos rascado la superficie de los modelos no supervisados, ni siquiera hemos llegado al fondo del potencial de los modelos implementados en el trabajo presente. Sin embargo, antes de experimentar con ello, antes de llegar a la parte de implementar modelos, una posible mejora sería probar los datos introducidos. Se ha observado que el escalar los datos mejora considerablemente el rendimiento de los modelos no supervisados presentados, de ahí viene la pregunta, ¿Qué otros procesos de procesamiento se pueden aplicar?

Algoritmo seleccionado

Se ha seleccionado el spectral clustering como el algoritmo que ha superado en la métrica de "Adjusted Rand Score" o Adjusted Rand Index", este algoritmo alcanzó un valor de 0.96 lo que lo coloca notablemente por arriba de los demás implementados. Cabe mencionar que esta métrica se llevó a cabo con los datos con escala estándar, ya que al no estar escalados, el modelo regresa una métrica de prácticamente 0. Sin embargo, esto no es un problema, ya que en la práctica, el escalar los datos se puede hacer de manera fácil y rápida. Este proceso de escalamiento resulta esencial para múltiples modelos, no es sorpresa que el spectral clustering tenga un buen resultado una vez que los datos han sido escalados.

Reflexiones

Diego Flores

Considero que trabajar con estos modelos de aprendizaje no supervisado es algo muy interesante ya que la forma en la que trabajan los algoritmos no es algo que se pueda pensar fácilmente, el poder computacional de estos es algo impresionante y la implementación de estos en la medicina podría llegar a ser un gran avance para este ámbito. La realización de este proyecto me ayudó a comprender un poco más cómo funcionan estos algoritmos de aprendizaje no supervisado para la clasificación.

Cristina López Ontiveros

Debo de mencionar que trabajar en este tipo de proyectos ayuda mucho a conocer un poco más a fondo el tipo de aprendizaje por refuerzo que vimos en las clases, y esto te pone a pensar seriamente en preguntas como "¿Cómo se le ocurrió a la máquina hacer eso?" o "¿Qué hay detrás de cómo funciona la máquina?", y cada vez me impresiona más cómo una máquina puede optimizar en tiempo y forma lo que nosotros comúnmente podríamos hacer en días.

Asimismo, debo de admitir que hubiera sido interesante conocer más tipos de métodos de aprendizaje por refuerzo.

Sofia Cañas

Con este proyecto pude entender mejor la diferencia entre aprendizaje supervisado y no supervisado, que honestamente me costó entender de manera profunda en un principio, no sabiendo cómo un algoritmo iba a separar datos si no se le especificaba la etiqueta a utilizar. La situación médica fue ideal porque nuestro objetivo es poder acercarnos a determinar un diagnóstico binario y de este modo ya sabemos el número de clusters a utilizar. Tampoco tenía mucho conocimiento de cómo se podían evaluar métodos con este tipo de aprendizaje porque por definición no tienen una referencia. Algo muy interesante fue poder relacionar los contenidos de esta materia con otras como fundamentos matemáticos para la física y modelación con ecuaciones diferenciales, porque el algoritmo de spectral clustering tiene una base muy fuerte en conceptos matemáticos. Ahora puedo ver claramente cómo ciertas ramas de las matemáticas nos ayudan a fortalecer a la inteligencia artificial.

Gerardo Moreno

Una perspectiva diferente de resolver un problema. El utilizar los modelos de aprendizaje no supervisado personalmente me ha parecido más asombroso que los supervisados. Evidentemente los supervisados son elementales, sin embargo, hay algo detrás del hecho de no darle ninguna información, etiquetas o feedback directo de si está haciendo bien el trabajo y aun así el modelo llegar a regresar una búsqueda satisfactoria. De igual forma, aprecié combinar dos mundos del aprendizaje no supervisado siendo uno el de la reducción de dimensiones y el otro el de clusters, al combinarlos, pude estudiar el comportamiento de datos al cómo el utilizar componentes con mayor relevancia en los datos, pueden ayudar al modelo de clusters. Sin duda es enriquecedor el poder combinar diferentes herramientas para un mismo objetivo.

Referencias

- Dendrograms. (s. f.). plotly. Recuperado 4 de mayo de 2022, de https://plotly.com/python/dendrogram/

- Fleshman, W. (2019) *Spectral Clustering*. Towards Data Science. Recuperado de : https://towardsdatascience.com/spectral-clustering-aba2640c0d5b#:~:text=Spectral%20clustering%20is%20a%20technique,non%20graph%20data%20as%20well.

- IBM (2018) *An End-to-End Approach for Scaling Up Spectral Clustering*. IBM. Recuperado de: https://www.ibm.com/blogs/research/2018/08/spectral-clustering/

- Mong, H. (s.f) Colector. Wiki. Recuperado de: https://hmong.es/wiki/Manifold

- Soni, M. (2020). What is Dendrogram? Maniksonituts. Recuperado 4 de mayo de 2022, de https://maniksonituts.medium.com/what-is-dendrogram-260df8b9d076.

- S. (2021). Hierarchical Clustering / Dendrogram: Simple Definition, Examples. Statistics How To. Recuperado 4 de mayo de 2022, de https://www.statisticshowto.com/hierarchical-clustering/

- Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.

- Keerthana, V. (2021) What, why and how of Spectral Clustering!. Analytics Vidhya. Recuperado de: https://www.analyticsvidhya.com/blog/2021/05/what-why-and-how-of-spectral-clustering/

- Wikipedia contributors. (2022, 30 abril). BIRCH. Wikipedia

  https://en.wikipedia.org/wiki/BIRCH

- Real Python. (2021, 8 enero). K-Means Clustering in Python: A Practical Guide.

  https://realpython.com/k-means-clustering-python/

Anexos

Datos

Base de datos: Diabetes Dataset
https://www.kaggle.com/datasets/mathchi/diabetes-data-set

Código

1. https://colab.research.google.com/drive/1x8FShoDPcxgCpQdj8jdrZ-SBUtVi1ZT8?usp=sharing
2. https://colab.research.google.com/drive/1gcO3hMqWxHudmZQD3Asb4nFd9WQK03Wa?usp=sharing
3. https://colab.research.google.com/drive/1eNfaBFd7aHIsSxw21ub9AOVdOJmrrUAh?usp=sharing
4. https://colab.research.google.com/drive/108BBUm9bPE1EpkKxunNRrI6wqpo_azPz?usp=sharing

Evidencias de trabajo en equipo