Instituto Tecnológico de Costa Rica

Área Académica de Ingeniería en Computadores
Programa de Licenciatura en Ingeniería en Computadores

CE1102-Taller de introducción a la programación

Tarea Programada 2: Clasificación K-medias

Profesor: Saúl Calderón Ramírez

Estudiantes:
Federico Alfaro Chaverri 2022051002
Valentin Tissera Doncini 2022145010

Primer semestre 2022

Indice

1 Introducción

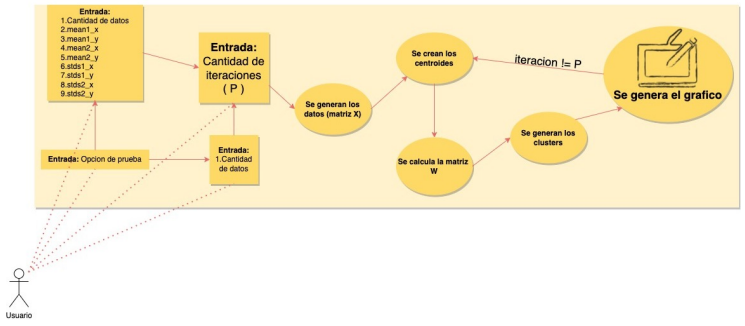
En la siguiente documentación se mostrara como se llevo a cabo el proyecto desarrollado en el mes de mayo. El proyecto consisitió de realizar un programa el cual permita visualizar la cantidad de datos N, la cual fue asignada por el usuario y dada una posicion aleatoria por el programa dos clase con un los cuales van a contar con signos diferentes cada uno, según el arreglo de etiquetas previamente explicada.

Ya una vez tengamos entendido esto lo único que queda es que el usuario le asigne valores a la desviación estándar y a la iteración para implementar en el algoritmo. Si todos estos valores fueron inicializados de una forma correcta el programa presentara el resultado final de la clasificación automática despues de usar el algoritmo de K-medias, coloreando los puntos según la membresía estimada por su algoritmo y los centroides para que se pueda ver de una manera eficiente su cambio.

2 Análisis del problema

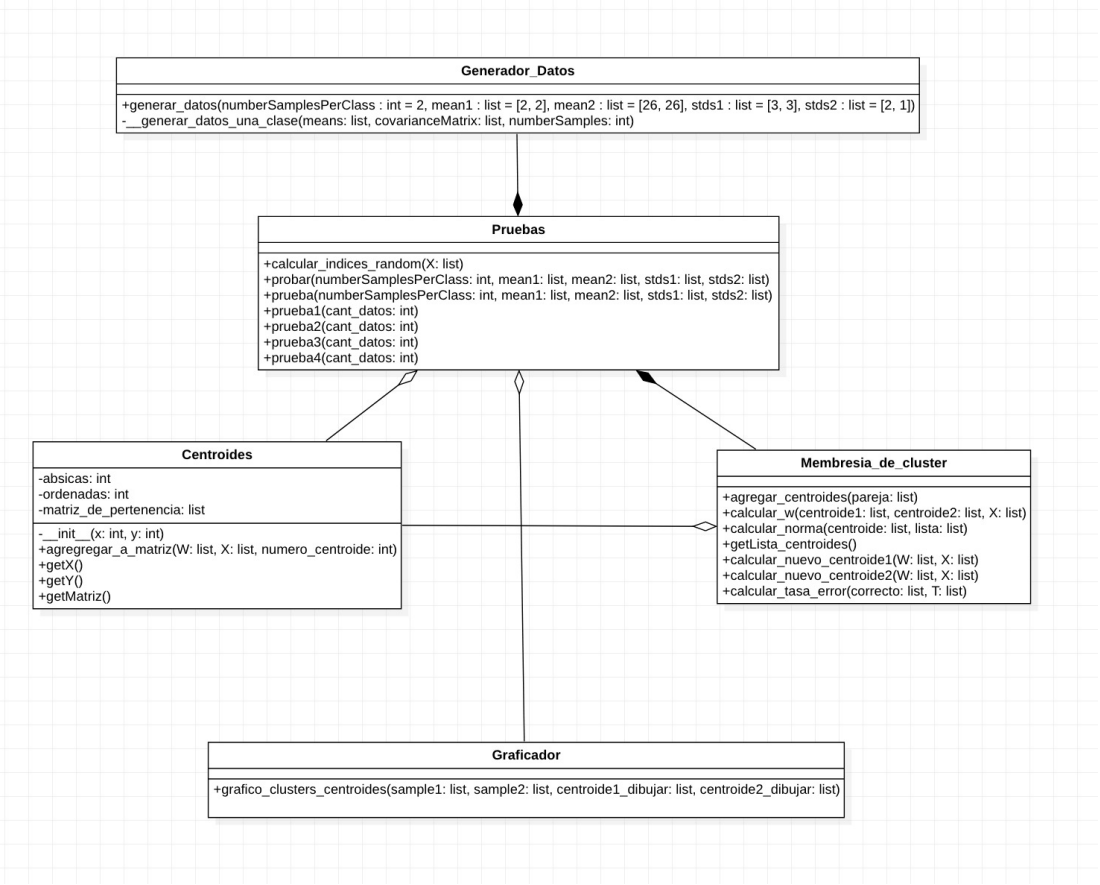
- Entradas: Las entradas de este programa serian la cantidad de datos N a generar aleatoriamente,los centroides de las clases 1 y 2. Las desviaciones estándar y por el ultimo la iteraciones del programa a llevarse a cabo.
- Salidas: El programa debe devolver una interfaz gráfica la cual permita visualizar todos los datos ingresados por el usuario después de que se le haya aplicado el algoritmo de k-medias y el calculo de los nuevos centroides.
- Restricciones:
  - Los centroides tienen que tener mínimo un dato asignado a cada uno de estos.
- Subproblemas:
  - Generar una cantidad N de datos con cierta desviación estándar.
  - Crear los primeros centroides.
  - Calcular la norma de cada punto con cada centroide.
  - Crear las matrices binaria (W) la cual va a ser usada.
  - Presentar el resultado de la clasificación automática usando el algoritmo de K-medias.
  - Graficar los datos separados por cluster.
  - Recalcular los nuevos centroides.
  - Iterar una cantidad $P \in \mathbb{N}^+$ de veces.

Figure 1: Diagrama de casos de uso



3 Diseño de la solución

Figure 2: Diagrama de clases

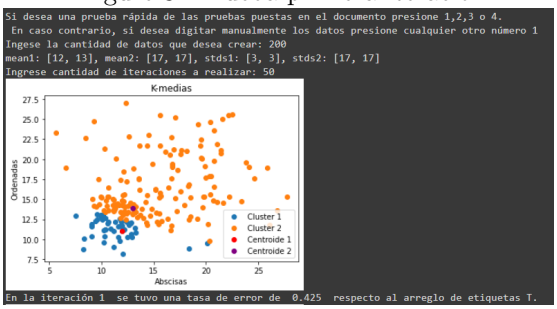


4 Implementación y pruebas

Para hacer este proyecto se hizo uso de la clase **Generador_Datos()** creada por el doctor Saúl Calderon en clase, la cual hace uso de la librería PyTorch y Numpy para generar una cierta cantidad de datos dada una desviación estándar y una media de los datos. Además, se importó el paquete KMeans para generar automáticamente los clusters de los datos para compararlos con los generados con el programa implementado en este proyecto. Para poder generar los gráficos de los datos se hizo uso de la librería matplotlib.

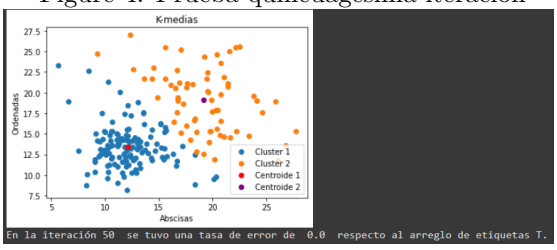
Para poder visualizar lo implementado en el programa se presentan un par de imágenes que muestran un ejemplo de como funciona el programa. Las siguientes dos figuras se muestran pruebas que muestran una prueba en la primera iteración y en la quincuagésima iteración. Los parámetros para la generación de los datos se muestran en la Figura 3.

Figure 3: Prueba primera iteración



Se puede ver en la Figura 4 la reposición de los centroides y de nueva distribución de los datos, la distribución de los datos a cada cluster es más equilibrada.

Figure 4: Prueba quincuagésima iteración



Al final de cada gráfico el programa muestra la tasa de error de nuestro programa con respecto a la solución brindada por el paquete KMeans en cada iteración.

5 Conclusiones y recomendaciones

Después de lograr implementar el método de K-medias podemos concluir que es un buen método de agrupamiento de datos y una buena forma de iniciarse en los mecanismos de aprendizaje no supervisados para las computadoras. Pudimos notar que existe la limitante dependiendo de la implementación, la forma en que se agrupan los datos cuando un centroide se encuentra muy separado del resto de los datos mientras que el otro centroide se encuentra cercano al resto. Este algoritmo puede tener errores cuando hay datos que no tienen cierta distribución.

Se recomienda que a en el momento de generar los datos se tome en cuenta la distribución estimada de los mismos, y descartar los datos que se salgan de cierto margen de distribución. Se recomienda también estudiar sobre graficación e interfaces gráficas en Python para planificar debidamente una forma de crear un objeto que sea compatible con las interfaces gráficas de forma más simple en vez de hacer el gráfico y luego intentar modificarlo para ponerlo en la interfaz gráfica.

Otra recomendación es separar en una clase todo lo que sea pertinente a la graficación de los datos y en caso de la interfaz gráfica también agregarlo en una nueva clase para tener una mejor modularización del programa y poder hacer pruebas de manera más sencilla.

6 Referencias

TP2. (s/f). Dropbox. https://www.dropbox.com/sh/f5hp6sqkw3qqral/AAAepMOgHmWgWUpxQ_I-BThya/Trabajos_Practicos/TP2?dl=0&preview=clustering.pdf&subfolder_nav_tracking=1

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. Computing in science engineering, 9(03), 90-95.

Wu, B. (2021, August). K-means clustering algorithm and Python implementation. In 2021 IEEE International Conference on Computer Science, Artificial Intelligence and Electronic Engineering (CSAIEE) (pp. 55-59). IEEE.