 **INSTITUTO POLITÉCNICO NACIONAL**

**ESCUELA SUPERIOR DE CÓMPUTO**

Ejercicio de Lab7: Clasificador KNN

**Alumno:** Delgado Acosta Luis Bernardo

Franco Calderas Sergio Alberto

**Grupo**: 5BV1

En esta práctica se reutilizó la parte de los métodos de validación. Solo se cambió la función para clasificar. Un clasificador K-Nearest Neighbors (KNN) predice la clase de cada punto de prueba (X\_test) según sus k vecinos más cercanos en el conjunto de entrenamiento (X\_train).

Interfaz de usuario gráfica, Texto, Chat o mensaje de texto

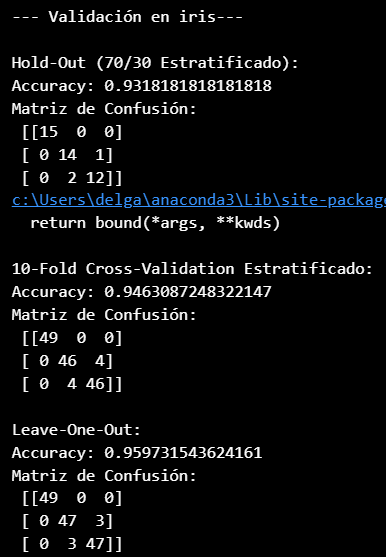
Descripción generada automáticamente

*Fig. 1 Función de KNN*

La función principal contiene una pequeña función la cual calcula la distancia Euclidiana entre dos puntos (x1 y x2). Posteriormente inicializa una lista para almacenar las etiquetas de clase predichas para cada punto en X\_test, se utiliza un ciclo for each, itera sobre cada punto en el conjunto de prueba X\_test, haciendo una predicción para cada uno de ellos. Para cada punto de prueba secalcula la distancia a todos los puntos en X\_train usando la función euclidean\_distance. Se crea una lista de tulas distances, donde cada tupla contiene: la distancia entre x\_test y un punto x\_train, la etiqueta label correspondiente a x\_train.

Se ordena la lista distances en función de la distancia (primer elemento de cada tupla). Luego, se seleccionan los k vecinos más cercanos utilizando [:k], que toma los primeros k elementos de la lista ordenada. El resultado es una lista k\_nearest\_neighbors que contiene las k tuplas con las distancias más pequeñas y sus etiquetas correspondientes. Se extraen las etiquetas (label) de los k vecinos en k\_nearest\_neighbors. Luego, se utiliza Counter para encontrar la etiqueta más frecuente entre vecinos. Most\_common(1)[0][0] devuelve la etiqueta con mayor frecuencia entre los vecinos. Finalmente, la función devuelve y\_pred, que es la lista de etiquetas predichas para todos los puntos de prueba en X\_test.

Resultados de los métodos de validación con el clasificador KNN:

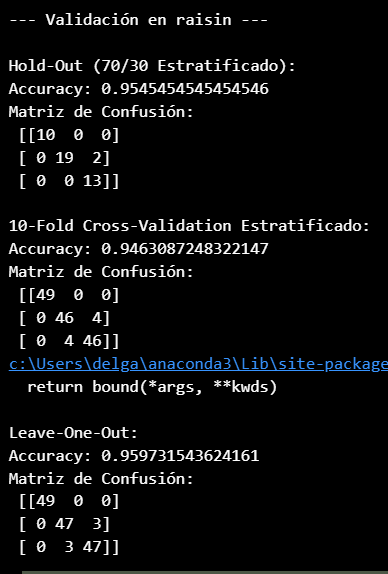


*Fig. 2 Resultados iris con KNN*

Imagen que contiene Texto

Descripción generada automáticamente

*Fig.3 Resultados glass con KNN*



*Fig. 4 Resultados raisin con KNN*

Enlaces a los repositorios

Bernardo: <https://github.com/BernardoD07/Machine-Learning-ESCOM.git>

Sergio: https://github.com/SergioCalderas/Trabajos-Machine-Learning.git