Laboratorio 02: Condensed k-Nearest Neighbors

Por:

Ian Gabriel Cañas Fernández, 1092228

.

Profesor: Juan S. Pérez R.,

Asignatura: INL367L, Secc 01

# Resumen:

El presente laboratorio lo iniciaremos implementando el algoritmo de k Nearest Neigbors (kNN) para varios valores de k y para varios tipos de distancias en el espacio, recordando que este algoritmo consiste en la clasificación automática de varios puestos de muestra respecto a los puntos conocidos más cercanos. Luego de observado el comportamiento del algoritmo ante diferentes tipos de distancias se estará implementando una variación a dicho proceso, llamado condensed k-NN, utilizado para reducir el set de datos para la clasificación.

# Ejercicios previos:

## P2.1 Desempeño de k-NN para diferentes métricas de distancia.

Se nos solicita adaptar el algoritmo desarrollado en el laboratorio 1 para que este calcule las distancias de varios métodos diferentes, los utilizados en el presente serán la distancia euclideana, Manhattan y Mahalanobis,

## P2.1. Preparación de base de datos para condensed Nearest Neighbour.

Para esta sección, se ha procurado simplificar la base de datos a sus datos más representativos, dígase los valores de frontera, pues dan una mejor distinción entre valores intermedios, pues en el centro de cúmulos queda más que claro la clasificación de un valor, para llevar a cabo este proceso se lleva a cabo el siguiente algoritmo.

La razón por la que este algoritmo se trabaja de dicha forma es porque garantiza una simplificación al mínimo de la base de datos, porque desconsidera datos redundantes.

# E2.1 Implementación en Matlab de kNN

En la Ilustración 1 se puede observar una variación que se le ha aplicado al algoritmo visto en el laboratorio 1, donde podemos observar que se le solicita al usuario el tipo de distancia con la que va a trabajar.

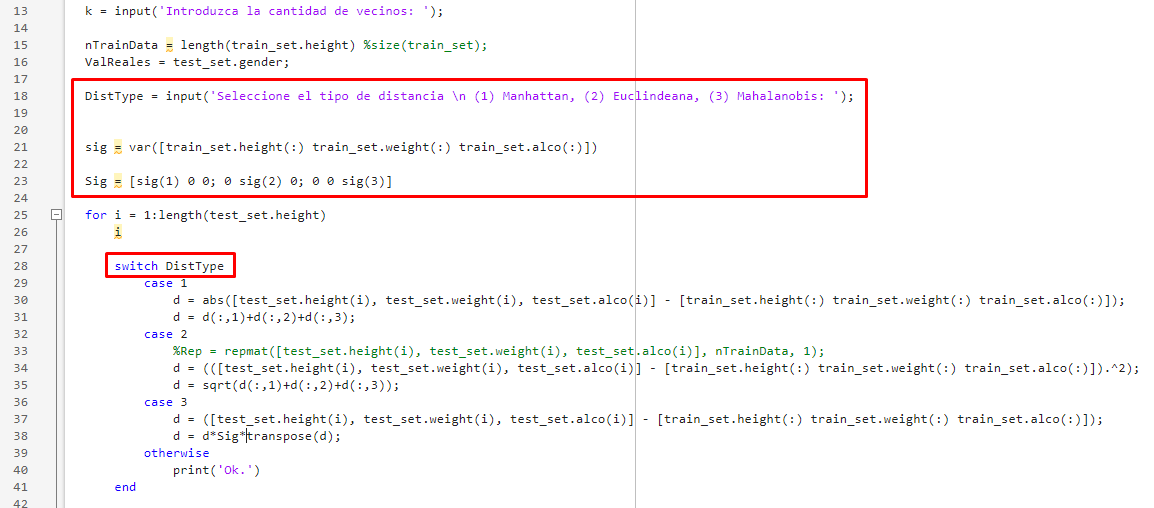


Ilustración 1. Adaptación k-NN a varios tipos de distancias.

# R2.1 Resultados:

En la Tabla 1 se sintetizan los resultados de precisión que se obtuvo con el dataset trabajado anteriormente mientras que en la Tabla 2 se presentan los tiempos de ejecución.

Tabla 1. Precisión mediante varias métricas de distancia y valores de k.

|  |  |  |  |
| --- | --- | --- | --- |
| K | Manhattan | Euclidean | Mahalanobis |
| 1 | 0.7009 | 0.7008 | 0.7010 |
| 3 | 0.7251 | 0.7254 | 0.7249 |
| 5 | 0.7455 | 0.7456 | 0.7451 |
| 9 | 0.7564 | 0.7566 | 0.7560 |
| 15 | 0.7678 | 0.7681 | 0.7668 |
| Promedio | 0.7391 | 0.7393 | 0.7388 |

Tabla 2. Tiempo de ejecución mediante varias métricas de distancia y valores de k.

|  |  |  |  |
| --- | --- | --- | --- |
| K | Manhattan | Euclidean | Mahalanobis |
| 1 | 186.2 | 162.3 | 171.9 |
| 3 | 206.9 | 182.6 | 161.5 |
| 5 | 182.2 | 155.4 | 154.9 |
| 9 | 184.3 | 176.8 | 189.2 |
| 15 | 184.9 | 173.6 | 184.9 |
| Promedio | 188.9 | 170.1 | 172.5 |

# A.2.1 Análisis:

Se ha podido observar el comportamiento del algoritmo de los k vecinos más cercanos, proceso de clasificación mediante el cual se hace una comparación entre los datos a ser analizados y su posición en el plano, tomando en consideración los datos ya reconocidos y clasificados que se encuentren en su entorno.

El algoritmo implementado ha sido probado mediante varios valores de k, de donde se interpreta que, para valores de k mayores, tiende a tenerse mayor nitidez en los cúmulos de un valor determinado. Confirmando este algoritmo con la clasificación de los datos calculados analíticamente para tener garantía de su correcto funcionamiento. Se ha seleccionado para todo caso una distancia euclideana.