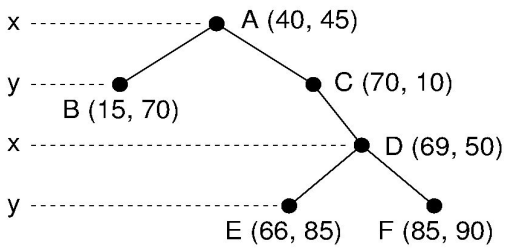
Algoritmos y Estructuras de Ubicación Espacial Implementando un KDTree con KNN

Introducción

Como demostración de lo aprendido en clases respecto al modelamiento de problemas usando estructuras espaciales, se nos pidió crear un programa para Manuel. Este programa debe ser capaz de obtener información de aplicaciones similares a la que él se encuentra desarrollando desde un conjunto de datos que él recopiló. Para ello se escogió el KDTree como estructura espacial y el KNN como algoritmo de búsqueda para encontrar aquellas apps similares a las que Manuel desarrolla.

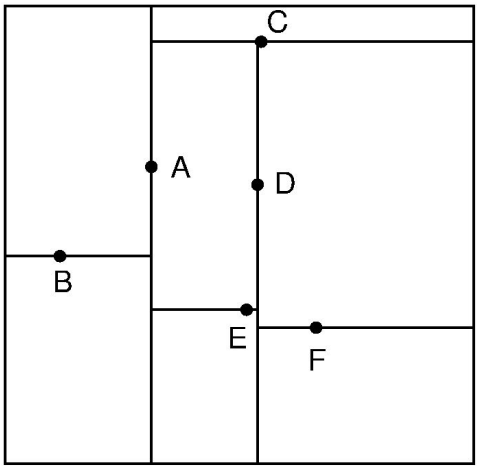
Desarrollo

Como se mencionó anteriormente, la estructura espacial escogida para el desarrollo del desafío fue el de KDTree. El cual ofrece ventajas significativas a la hora de almacenar elementos con múltiples campos para facilitar búsquedas.



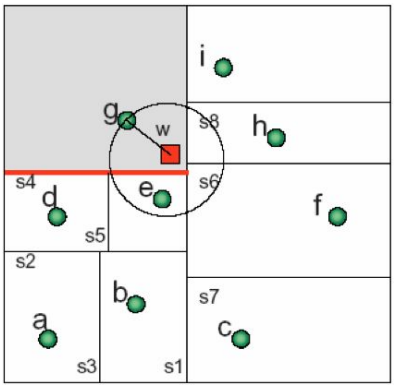
La idea de usar el KDTree es la de aprovechar la multidimensionalidad de los elementos del dataset entregado. Si bien la estructura es casi idéntica a la de un árbol binario normal, el KDTree aprovecha la dimensión de los datos que guarda para crear “k” dimensiones.

Estas dimensiones ayudan a la segmentación de “cuadrantes” (definidos como hiperplanos) en donde se almacenan los datos, los cuales ayudan a la determinación de la distancia entre los elementos almacenados.

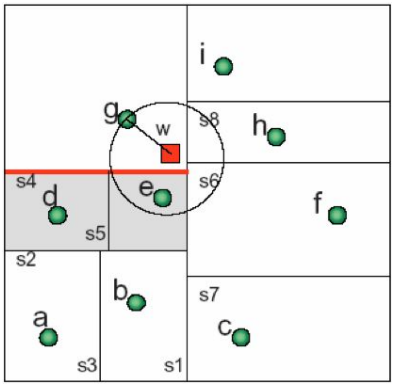


En adición a la implementación del KDTree, también teníamos que implementar un algoritmo de búsqueda para nuestra estructura, a modo de poner en prueba tanto el algoritmo de búsqueda como la estructura. El algoritmo escogido fue el KNN o los “k” vecinos más cercanos.

Para aplicar el KNN en un KDTree, primero se comienza determinando el cuadrante que correspondería al dato de referencia ingresado. Luego se buscan los vecinos más cercanos dentro de ese mismo cuadrante.



Cuando ya se recorrieron todos los vecinos de aquel cuadrante inicial, se continúa la búsqueda en el resto de los cuadrantes. Esto se cumple siempre y cuando dichos cuadrantes se encuentren a una menor distancia que el k-vecino más lejano (el último de los vecinos en términos de distancia), ya que no tendría sentido buscar ahí, esto es porque la distancia al nodo más cercano de ese cuadrante sería en el mejor de los casos igual a la distancia al cuadrante mismo, y si resulta ser más lejano que el k-vecino más lejano, ya no se va a encontrar ningún elemento más cercano al punto de referencia.



Finalmente se muestra una lista de largo “k” con los vecinos más cercanos basado en la función de distancia euclidiana, la cual entrega la información de las aplicaciones similares al dato ingresado por parámetro para la búsqueda, junto con el número de vecinos deseados.