Explicación funciones kd-tree

El árbol kd-tree es una estructura de datos que particiona el espacio, para organizar los puntos en este

Este árbol tiene nodos; Estos nodos contienen la data del punto en el espacio, una referencia a su hijo izquierdo, derecho y en que dimensión del árbol se encuentran ()

class Node:

    def \_\_init\_\_(self,point,left\_child,right\_child,dim):

        self.point=point

        self.left\_child=left\_child

        self.right\_child=right\_child

        self.dim=dim

Para Crear el árbol primero le enviamos una lista con todos los puntos que queremos plotear:

points2=[(2,3),(5,4),(9,6),(4,7),(8,1),(7,2)]

root=make\_kdtree(points2)

El código de make\_kdtree es una función recursiva que funciona de la siguiente manera:

* Ordena la lista que le enviamos según su coordenada (parte en x, luego en y , luego x y así sucesivamente), En este caso quedaría así:

[(2,3),(4,7),(5,4),(7,2),(8,1),(9,6)]

* Luego seleccionamos el punto del medio, y cortamos la lista en 2, Aquí seleccionamos el punto [7,2] y la lista de la izquierda son todos los valores donde su x<7 y la lista de la derecha x>7:

1)[(2,3),(4,7),(5,4)] 2)[(7,2)] 3)[(8,1),(9,6)]

* Creamos un nuevo nodo, al cual le insertamos el punto de al medio, y para crear su hijo izquierdo le pasamos la lista 1 y le aplicamos la misma función para el hijo izquierdo, para crear el hijo derecho es exactamente lo mismo pasándole la lista 3

return Node(

   points[median],

   make\_kdtree(points[0:median],depth+1), #left\_child

   make\_kdtree(points[median+ 1:],depth+1), #right\_child

   dim)

* Una vez que vamos repitiendo el proceso llegaremos a tener una lista vacía y es cuando la función retorna

if not points:

       return None