Assignment 3 Report

k-NN Search using KD-trees

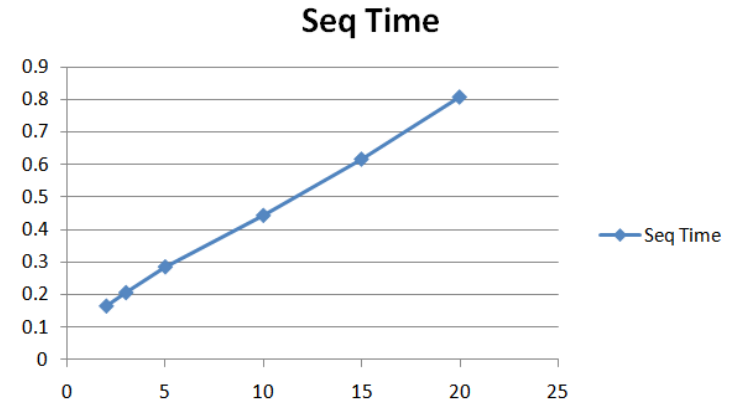
Submitted By:

Ankesh Gupta- 2015CS10435

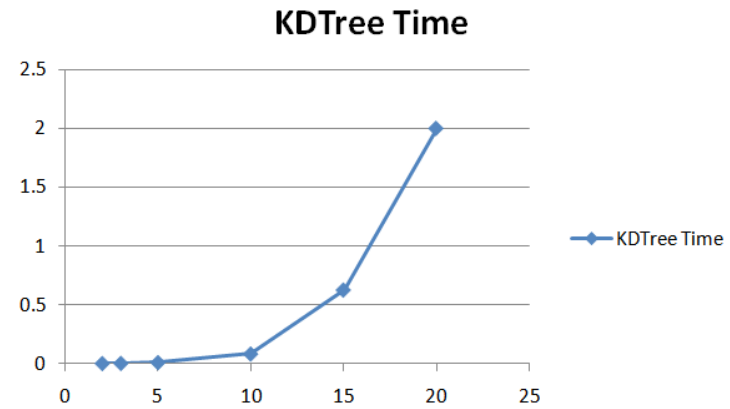
Prakhar Kumar- 2015CS10667

Saransh Goyal- 2015CS50292

a) 20-NN query Sequential Scan v/s KD-trees
We observe the following graphs for running times against dimensions:

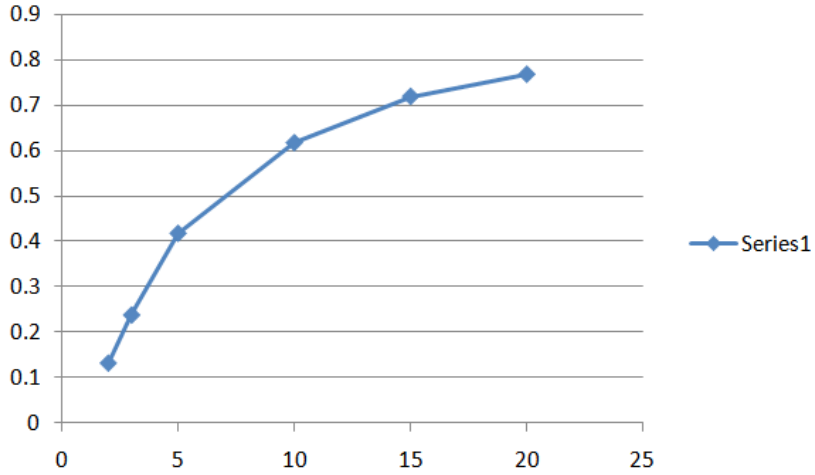


Graph for Sequential scan running time for 100 queries



Graph for KD-tree Algorithm running time for 100 queries

b) 100-NN query
The graph observed for given ratios is:



c) Explanation for our observations:

a. Running time for sequential scan: The graph is linear w.r.t. dimension. The time required for each operation increases linearly with dimension as the size of data points increases. The number of steps for every operation remains the same, so the overall time increases linearly.

b. KD-tree Running Time: The running time for 100 queries grows exponentially with dimension. The main reason is that with increasing dimension, we need to traverse the tree deeper to find the suitable points to be put in the heap. We need more iterations to go through all the dimensions of the data once. In high dimensional spaces, we have to scan more leafs of the tree. For 100,000 data points, $\log_2(n)$ becomes 16.6. So in the best case, the height of the tree becomes at least 17. So in 15 or 20 dimensional spaces, one iteration over all dimensions practically covers the complete tree, so it just becomes a complicated form of linear search. A detailed analysis of the observed running times shows that increasing the dimension by 1 increases the running time by approx. 1.48 times. However, this is not observed for transition from 15 to 20 dimensions as they are just complicated linear searches.

c. Mean $2^{nd}/100^{th}$ nearest neighbour distance ratio: Consider squared distances. Let the std. dev for each dimension be sigma. A d-dimension point is equivalent to summation of d iid variables with std.

dev sigma. So the std dev for d-dimensional point becomes d*sigma. Std. dev. is a measure of variability of a random variable. So the ratio for squared distances should increase linearly with d. Therefore, the graph of ratio of L2 distance increases like y = sqrt(x) graph.