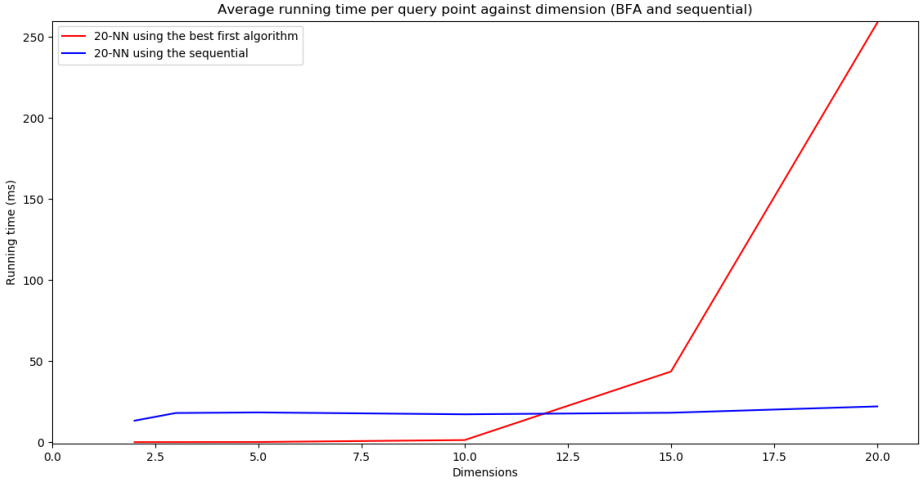
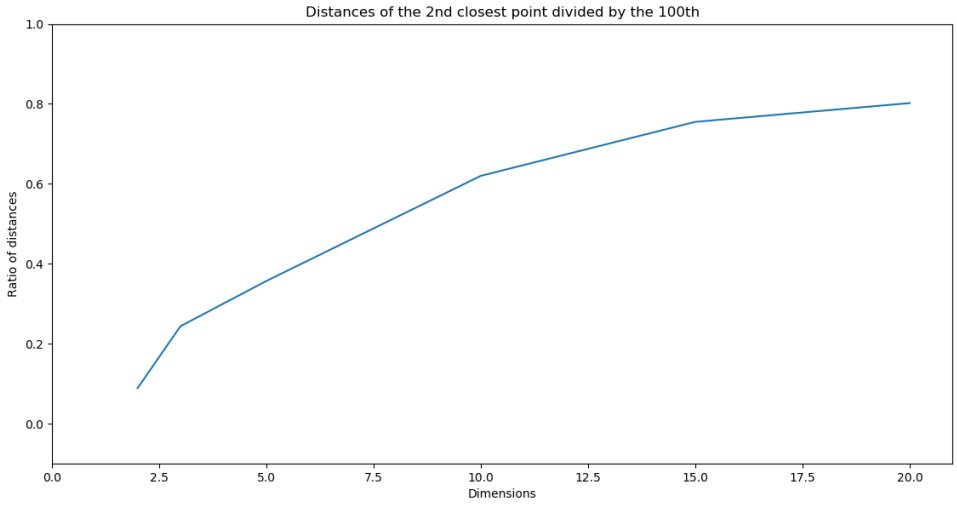
Report – COL362
Assignment 3: Kd-Tree

Quentin BOURGUE – 2017EE19079

Antonin VILLEMIN - 2017EE19080

Laura PICHON - 2017EE19081





c) We can see, thanks to the first graph above that the best-first algorithm is faster than a sequential search for the lowest dimension but become quickly slower. We can explain this difference by how a Kd-Tree is built. Indeed, when we go from a parent to a child in a kd-tree, we split only one dimension. As the number of points doesn't change in the tree, the distance in a tree between the root and a leaf remains the same. So, each dimension is ten times less split in dimension 20 than in dimension 2. So finally when we apply the minimum distance between the mbr and the query point, it is more often higher than first distance in the max heap which mean that we will check more mbr.

For the second graph we observe that the ratio increases with the dimension. Indeed, if we increase the dimension, the space has become larger which means that the average distance between each point has increased. So the 100^{th} nearest point will be farther when we increase the dimension.