BIG DATA COMPUTING 2021/22 - HOMEWORK 3

PYTHON VERSION

Run your algorithm on the cluster on CloudVeneto using the following datasets: **HIGGS-REDUCED-7D.txt** (about 1.2M points in 7 dimensions), and **artificial9000.txt** (9200 points in 2 dimensions). The datasets are in the **directory /data/BDC2122** of the HDFS. You must fill the two tables below, one for each dataset, where the headers of the rows indicate the values to report, and the headers of the columns indicate the configurations of parameters to be used.

The first table collects results aimed at assessing the **scalability** of the algorithm.

HIGGS-REDUCED-7D.txt	2 executors k=10, z=150, L=2	4 executors k=10, z=150, L=4	8 executors k=10, z=150, L=8	16 executors k=10, z=150, L=16
Time to read input from file (in ms)	11156.631708145142	7430.318355560303	5485.337734222412	5754.363536834717
Time of ROUND 1 (in ms)	369891.97516441345	187617.8948879242	96774.56498146057	48003.894090652466
Time of ROUND 2 (in ms)	49.288034439086914	150.64716339111328	431.1201572418213	1829.7722339630127
Time to compute objective function (in ms)	2348.7865924835205	1363.9881610870361	634.9282264709473	492.1143054962158
Value of objective function	9.12239051009	7.56819300974	6.53216055326	6.06278890865

The second table collects results aimed at comparing the **accuracy** attained by the algorithm against the one attained by the sequential algorithm from Homework 2 on the entire dataset.

Artificial9000.txt	2 executors k=9, z=200, L=2	4 executors k=9, z=200, L=4	8 executors k=9, z=200, L=8	16 executors k=9, z=200, L=16	Sequential algorithm from Homework 2 with k=9, z=200
Value of objective function	13.2424381818	12.8557916131	11.8414916712	11.7176606027	11.576939707884812

Provide below a brief comment to justify the scalability and accuracy observed (your answer should be of at most 6 lines, font 12 points):

The scalability of the algorithm is shown by the fact that by increasing the number of partitions the total execution time decreases. The execution time of Round 1 halves accordingly by doubling the executors and the Round 2 increases because the sequential algorithm has to handle more points. By increasing the number of partitions the value of the objective function converges to the score of the sequential algorithm, that's because increasing the partitions leads to a better representation of the initial full pointset, which is where Sequential computes its solution.