CS585: Big Data Management
Spring-2015

HW4/Project 3

Total Points: **80**

Release Date: **03/6/2015**

Due Date: **03/24/2013 (4:00pm)**

Teams: **Project to be done in teams of two.**

Short Description

In this project, you will write map-reduce jobs that implement data mining and machine learning techniques in Hadoop. More specifically, you will implement the ***K-Means*** clustering technique and will learn ***RHadoop***.

Problem 1 (K-Means Clustering) [50 points]

K-Means clustering is a popular algorithm for clustering similar objects into *K* groups (clusters). It starts with an initial seed of K points (randomly chosen) as centers, and then the algorithm iteratively tries to enhance these centers. The algorithm terminates either when two consecutive iterations generate the same K centers, i.e., the centers did not change, or a maximum number of iterations is reached.

Hint: You may reference these links to get some ideas (in addition to the course slides):

http://en.wikipedia.org/wiki/K-means_clustering#Standard_algorithm
https://cwiki.apache.org/confluence/display/MAHOUT/K-Means+Clustering

Step 1 (Creation of Dataset) [10 points]:

- Create a dataset that consists of 2-dimenional points, i.e., each point has (x, y) values. X and Y values each range from 0 to 10,000. Each point is in a separate line.
- Scale the dataset such that its size is around 100MB.
- Create another file that will contain K initial seed points. ***Make the "K" value as a parameter to your program***, such that your program will generate these K seeds randomely, and then you upload the generated file to HDFS.

Step 2 (Clustering the Data) [40 points]:

Write map-reduce job(s) that implement the K-Means clustering algorithm as given in the course slides. The algorithm should terminates if either of these two conditions become true:

a) The K centers did not change over two consecutive iterations
b) The maximum number of iterations (make it six (6) iterations) has reached.

- Apply the tricks given in class and in the 2nd link above such as:
  - Use of a combiner
  - Use a single reducer
  - The reducer should indicate in its output file whether centers have changed or not.

Hint: Since the algorithm is iterative, then you need your program that generates the map-reduce jobs to control whether it should start another iteration or not.

Problem 2 (Use of RHadoop) [30 points]

Going back to Project 1, Customers table (ID, Name, Age, CountryCode, Salary). Write an RHadoop script to do the following:

1) creates a map-reduce job that aggregates the records based on the CountryCode, i.e., For each country code, we need the count of customers.
2) Plot the output where country codes on the x-axis, and the count on the y-axis
3) Sort the output descending based on the count, and re-plot the chart.

What to Submit

You will submit a single zip file containing the java code answering Problem 1 as well as the RHadoop script answering Problem 2. Also include a .doc or .pdf report file containing any required documentation, e.g., the two plots that you generate from Problem 2

How to Submit

Use blackboard system to submit your files.