Introduction

You were introduced to Spark with the claim that it’s 100x faster than MapReduce because it does in-memory processing. In this assignment, you will be expected do a Proof of Concept(POC) on this.

As per Techopedia, a proof of concept (POC) is a demonstration, the purpose of which is to verify that specific concepts or theories have the potential for real-world application. POC is, therefore, a prototype that is designed to determine feasibility but does not represent deliverables. In IT industry POCs are done for various purposes, one of them could be to compare the performance of two different tools designed to perform the same task. The overall objective of POC is to find solutions to technical problems. POCs are purely experimental. Hence, the results achieved during a POC may not be the final one. You might get different results when the solution is implemented on a separate dataset or an entirely different scenario. So, it is encouraged to test and experiment with recurring use cases or scenes and take a decision after analysing all the results.

So for this POC, you will be doing your analysis using the below-mentioned use cases:

lookup of a single row from the entire data set

Filtering multiple rows

Group by and then order by

The above three use cases will be implemented using Pig and Spark RDD. Which means you are expected to develop 3*2 i.e. 6 programs, execute each of them in your Amazon EC2 instance/Cloudera Quickstart VM and generate the output in an HDFS location. Let's say for the use case "lookup of a single row from the entire data set", you will develop and execute a Pig script and Spark code using RDDs. You have to ensure that, the output generated by both the methods are consistent.

Problem Statement

As already mentioned, you will have to perform the POC on three use cases i.e. single record lookup, filter and GroupBy accompanied with ordering in ascending order. So their respective problem statements are mentioned below. The following problems you will have to solve using Pig and Spark RDDs:

Fetch the record having VendorID as '2' AND tpep_pickup_datetime as '2017-10-01 00:15:30' AND tpep_dropoff_datetime as '2017-10-01 00:25:11' AND passenger_count as '1' AND trip_distance as '2.17'

Filter all the records having RatecodeID as 4.

Group By all the records based on payment type and find the count for each group. Sort the payment types in ascending order of their count.