**SI 618 Lab 5 - SparkSQL**

In this lab, we will practice SparkSQL. Put your NFL team jerseys on because we will look into NFL data (Go Hawks!).

**Question 0: (20 points)**   
Load the json file *hdfs:///var/umsi618/lab5/NFLPlaybyPlay2015.json* (obtained from <https://www.kaggle.com/maxhorowitz/nflplaybyplay2015>) and register it as a table. This dataset includes play-by-play data from all NFL games in 2015.  The dataset contains 46,129 rows and 63 columns. Each row represents one play. Each play is broken down into great detail containing information on; game situation, players involved and results. There is a lot going on here (63 columns!) but in this lab we will only be using a small fraction of them.

**Question 1: (40 points)**

The goal here is to understand how good each team is at progressing the ball, compared to their opponents across all games. Consider the following toy example:

|  |  |  |  |
| --- | --- | --- | --- |
| GameID | Posteam | DefenseTeam | YardsGained |
| 1 | SEA | GB | 50 |
| 1 | SEA | GB | 20 |
| 1 | GB | SEA | 10 |
| 1 | GB | SEA | 10 |
| 2 | NE | SEA | -5 |
| 2 | SEA | NE | 30 |
| 2 | SEA | NE | 30 |

Imagine these rows represent all games played (in reality there will be many more games and many more plays per game. I am also muting a lot of other columns here for ease of reading). You need to first compute, per game, how much better each team is compared to their opponent in progressing the ball. Let’s call this measure *delta-yards*. In our toy example, for Game with GameID 1, SEA has a total of 70 yards (offense denoted by *posteam* gains yards, not the defense) while GB has a total of 20. Therefore, *delta-yards* for SEA is 70-20=50, while it is 20-70=-50 for GB. Similarly, for game with GameID 2, SEA has 60 total yards while NE has -5. Therefore, *delta-yards* for SEA in Game 2 is 60-(-5)=65, while it is -5-60=-65 for NE. Given *delta-yards* per game, you can estimate how good a team is over all games by identifying the mean of *delta-yards* across all games they played. Let’s call this measure *mean-delta-yards*. For our toy example, *mean-delta-yards* for SEA is (50+65/2)=57.5. It is -50 for GB and -65 for NE since they have only one game listed.

Compute *mean-delta-yards* for all teams. Your output should include teamname <tab> *mean-delta-yards* in decreasing order of *mean-delta-yards*.

|  |
| --- |
| Save it as si618\_lab5\_output\_1\_UNIQUENAME.tsv. Your output should match si618\_lab5\_desired\_output\_1.tsv. |

**Question 2: (40 points)**

Some teams have better running backs while others have better receivers. Similarly, some quarterbacks are accurate passers while others are more mobile and hence able to run the ball. So, teams vary in how much they rely on the “run” vs “pass” game. You can learn this from this data. For each offense (denoted by *posteam*) estimate the run to pass ratio across all games (to be obtained from the *PlayType* column—there are other types of plays besides runs and passes, you can ignore those). Output teamname <tab> run-to-pass-ratio where teams are sorted in increasing order of run-to-pass-ratio.

You’re going to need to use nested queries to perform this calculation, as a hint, here’s one of the queries below. How might you perform a self-join to answer this question? (*Think back to HW2 when we found actors who co-stared together.)*

|  |
| --- |
| SELECT posteam, PlayType, count(\*) AS count  FROM nfl  WHERE posteam IS NOT NULL AND PlayType IN ('Pass','Run')  GROUP BY posteam, PlayType |

|  |
| --- |
| Save it as si618\_lab5\_output\_2\_UNIQUENAME.tsv. Your output should match si618\_lab5\_desired\_output\_2.tsv. |

You can view the data guide here for more information:  
<https://github.com/ryurko/nflscrapR-data/blob/master/legacy_data/README.md>

You MUST use **SparkSQL** to do this homework. Other solutions will not get any credit.

**What to submit:**

Submit a zip file named si618\_lab5\_youruniqname.zip containing your Python source code file, si618\_lab5\_uniquename.py, and the 2 tsv files.