Homework-4: Data Engineering

Deadline: October 3rd, 11:59PM ET.

In this homework, you are expected to practice the implementation of data engineering related tasks.

Your homework submission should be on GitHub. Use the following GitHub classroom to access the assignment and create your assignment repository: https://classroom.github.com/a/6i_s1N8f

You should submit the URL for your GitHub repository on Canvas. Grading penalty will be applied if otherwise.

Cite any external sources you use. External sources shouldn't exceed more than 30% of the final solution.

Submit your answers in one (or more) Jupyter notebook(s) **but clearly identify in your ReadMe file the location of the answers for every question**. If you need to include output or screenshots, you may include them in your notebook or attach them as separate image files using the naming convention qx.png (where x is the question number). If you have more than one image for a given question, name the files as qx_1.png, qx_2.png, etc.

A sample structure of your notebook is shown below.

Q1

Refer to q1.sql and q1.png

Q2

```
from pyspark.sql.functions import monotonically_increasing_id


### Some code here
## For output, refer to q2.png
```

Use the NSL-KDD dataset to answer the first four questions:

Q1. (15%): Display the count of each protocol_type for logged in and non-logged in users.

- Submit both code and output screenshot (or print it on your Jupyter notebook).

Q2. (15%): Create a new data frame that includes all network traffic with normal traffic (i.e., traffic with no attacks identified) and override the protocol_type to be “tcp” for all of them. Show a sample output.

- Submit both code and output screenshot (or print it on your Jupyter notebook).

Q3. (20%): Repeat the first two questions on Spark cluster hosted on the cloud. Provide a screenshot of the final output displayed on the cloud with the cloud URL clearly visible in the screenshot.

- Submit your notebook that ran on the cloud and the output screenshot showing the cloud URL (printing the output in your jupyter notebook is not sufficient and a screenshot showing the cloud URL is required).

Q4. (20%): Ingest both of your training and testing NSL-KDD dataset into one Postgres Database table. Add one column to differentiate between the train and test data elements. Note: Don’t include any changes you made in Q.1 to this dataset, and you don’t have to run this question’s solution on the cloud. Show a sample output when you ingest the data from a Postgres table.

- Submit both code and output screenshot (or print it on your Jupyter notebook).

Q5. (30%): Use the Machine Learning Process Flow chart shown in the lecture to conduct all the feature engineering steps that are required to be done on the NFL-Pro-Bowl dataset. You may download the dataset from the following repository https://github.com/nfl-football-ops/Big-Data-Bowl/tree/master/Data

- Limit your data cleaning and engineering to Plays.csv
    - Hint: consider the ML problem that eventually needs to be addressed after the feature engineering phase is the prediction of the result of a given play (i.e. prediction of the values in the PlayResult column)
- Stop at the data scaling stage of the process flow chart in Phase-2. (Make sure to conduct phase-1 as well).
- The following file shows the details of the various data points in the dataset: https://github.com/nfl-football-ops/Big-Data-Bowl/blob/master/schema.md
- Use pipelines when appropriate.
- Submit both code and output screenshot of the data frame that resulted from the data scaling phase (or print it on your Jupyter notebook).