WILFRID LAURIER UNIVERSITY
LAURIER
Inspiring Lives.

Assignment #1 CP422 – Programming for Big Data

Department of Physics and Computer Science, Faculty of Science, Waterloo Campus

Fall | 2024

Analyzing NYC Taxi Trip Data with Databricks and Apache Spark

Submission instructions:

a) **All code "the ipythonNotebook" must be converted to a single PDF. The output of each cell must be visible.**
b) **All visualization must be attached to your report.**
c) **Merge all your PDFs (including the assignment template that contains the answer to part C) into a *SINGLE* PDF and then submit it to the course page on MyLS.**

Objective: This assignment will help students understand the end-to-end process of analyzing a large dataset using Apache Spark on Databricks. They will perform data ingestion, transformation/cleansing, exploratory data analysis (EDA), and generate insights related to big data.

Dataset:

The dataset comes from the NYC Taxi and Limousine Commission (TLC), which contains trip data for all yellow and green taxi rides in NYC.

Dataset URL: NYC Taxi Data (2010–2023) – Create a Kaggle account if you do not have one already: https://www.kaggle.com/datasets/elemento/nyc-yellow-taxi-trip-data

Part A: Basic Prescriptive Analytics:

Step-by-Step Instructions:

Step 1: Set Up Databricks and Apache Spark

a) Log into Databricks "Community Edition" and create a new **cluster** with Apache Spark.
b) Create a new **notebook** where you will perform all the operations in the following steps.

Step 2: Data Ingestion

a) Use Databricks to import the NYC Taxi dataset from the provided URL.
b) Load the dataset in **CSV** format.

Step 3: Initial Data Exploration

a) Display the schema of the dataset to understand the structure
b) Display basic statistics of numerical columns, such as trip distance, passenger count, and fare amount

Step 4: Data Cleaning

a) Handle missing values by removing them for columns "fare_amount, trip_distance, and passenger_count."
b) Filter out rows with invalid data (e.g., fare_amount < 0 or trip_distance = 0)
c) Convert the pickup_datetime and dropoff_datetime columns to timestamp data types
d) Create new columns, such as trip duration (in minutes) and trip speed (in miles per hour)

Step 5: Exploratory Data Analysis (EDA)

Perform basic grouping and aggregations:

a) Calculate the **average fare** and **average trip distance** for trips grouped by the number of passengers
b) Find the busiest times of day for taxi pickups
c) Which neighbourhoods have the highest average fare amount?

Step 6: Visualizing the Data

Create visualizations to present your findings using Databricks' built-in visualization tools or libraries like Matplotlib or Seaborn.

a) Plot the distribution of trip distances.
b) Visualize average fares by hour of the day.

Step 7: Summary and Insights

Provide a report summarizing key insights from the analysis. This could include:

a) Peak taxi usage times and trip patterns.
b) Relationships between trip duration, speed, and fare.
c) Any geography-related patterns (e.g., specific neighbourhoods having higher fares).

Part B: Advanced Prescriptive Analytics:

Feature Engineering – Compute the following "extraction of new useful features"

1. **Trip Duration**: Calculate the trip duration by subtracting the pickup time from the drop-off time.
2. **Hour and Day**: Extract hour and day from the pickup_datetime to analyze hourly and daily patterns.
3. **Trend Over Years:** Analyze how the trip duration changes over the years. Plot the results.
4. **Hourly Analysis:** Check how the trip duration varies throughout the day. Plot the results

5. **Identify Hotspots:** Determine the most popular pickup and drop-off locations by analyzing their coordinates. An ordered bar or scatter plot or similar is required to visualize the results of this step.
6. Analyze the average fare amount by pickup location to identify which areas have the highest fare charges.
7. Perform correlation analysis to understand the relationships between key features such as trip duration, trip distance, and fare amount.
8. Examine taxi demand by analyzing the number of trips over time monthly to detect seasonality.

Part C: Practice questions:

Answer the following question

1. Define Big Data and explain why traditional data processing methods are insufficient for managing it.
2. Describe the role of Hadoop in solving big data challenges. What are the core strengths of Hadoop compared to traditional relational databases?
3. How do Hadoop's scalability and fault-tolerance features contribute to its popularity in big data processing?
4. What are the key characteristics (the 4Vs) of Big Data?
5. Differentiate between structured, semi-structured, and unstructured data with examples. Explain how Hadoop handles these different types of data.
6. Define the following key components of the HDFS and describe their functions:
   a) NameNode
   b) DataNode
   c) Secondary NameNode
   d) Block Replication
7. Explain the MapReduce programming model. What are the main phases of a MapReduce job, and what are their purposes?
8. Describe the concept of data locality in MapReduce and its importance for efficient processing in distributed systems.
9. Explain what a combiner is in the MapReduce framework and how it optimizes the performance of a MapReduce job.
10. Explain what Hadoop Streaming is and how it allows Python programs to interact with Hadoop's MapReduce framework.
11. Discuss the concept of standard input and standard output in the context of Hadoop Streaming. Why are these important for connecting Python programs to Hadoop?
12. Provide an example of a real-world use case where Hadoop Streaming with Python would be preferable over other methods.