DEPARTMENT OF COMPUTER ENGINEERING

FACULTY OF ENGINEERING

AHMADU BELLO UNIVERSITY, ZARIA

COEN542 (BIG DATA ANALYTICS) PROJECT:

NYC TAXI

BY GROUP 7

LAWAN UMAR SANI	U18CO1016
SALAMATU UMAR ARDO	U18CO1010
YAHAYA SULAIMON	U18CO1028
OLAYIWOLA ABDULQADIR ALABI	U19CO2007
NASIR ZAKARIYYA BELLO	U18CO1071
ABUBAKAR NURA	U19CO2004

SUBMITTED TO

DR. S.M YUSUF and DR. Y. Ibrahim

SEPTEMBER, 2025

ABSTRACT

Big data analytics plays a vital role in modern decision-making, particularly in urban transportation where millions of daily trips generate massive datasets. In New York City, the Taxi and Limousine Commission (TLC) provides detailed trip records containing pickup and drop-off locations, times, fares, passenger counts, and payment methods, offering opportunities to analyze travel patterns, optimize fleet distribution, and enhance passenger experience. However, the sheer volume and complexity of these datasets, such as the multi-gigabyte NYC Yellow Taxi data for 2015–2016, make traditional tools like Excel or Pandas insufficient. To address this challenge, a scalable Big Data Analytics Pipeline was developed, integrating Dask for distributed data processing, Amazon S3 for scalable cloud storage, Scikit-learn for building a Logistic Regression model to predict tipping behavior, and Streamlit for interactive visualization and dashboard deployment. The pipeline covers the full lifecycle of data ingestion, preprocessing, feature engineering, machine learning, and visualization, enabling efficient handling and analysis of large-scale data. Results show that Dask effectively manages large datasets, S3 provides reliable storage, the logistic regression model achieved strong predictive performance, and the Streamlit dashboard offered intuitive insights into demand patterns, fare distributions, and tipping likelihood. This project highlights the potential of combining distributed computing, cloud storage, and machine learning to derive actionable insights from large-scale mobility datasets, contributing significantly to the field of urban informatics.

TABLE OF CONTENTS

CHAPTER ONE

INTRODUCTION

1.1 Background of the Study

Transportation is one of the largest generators of big data. In the age of smart cities, every journey generates digital traces, from GPS coordinates and timestamps to transaction records and passenger preferences. In a city like New York, where taxis have been integral to public mobility for decades, the accumulation of such data is immense.

The **New York City Taxi and Limousine Commission (TLC)** began publishing trip record datasets to support open data initiatives. These datasets provide fine-grained insights into urban movement: the time of day most people commute, the neighborhoods with the highest demand, and even the tipping habits of passengers. The sheer volume of this dataset offers unprecedented opportunities for both **academic research and practical applications**.

Yet, analyzing this data at scale is challenging. A single year's worth of taxi trips can exceed tens of millions of rows, amounting to several gigabytes in storage. Traditional tools are ill-suited to handle data of this size. Moreover, the richness of the dataset—with spatial, temporal, and financial attributes—demands advanced processing and machine learning pipelines.

Big Data Analytics provides the methodological and technological framework for addressing such challenges. Distributed frameworks like Dask make it possible to analyze large datasets without requiring supercomputers. Cloud storage services like Amazon S3 ensure data can be persisted and accessed efficiently. Machine learning models built on top of this processed data can reveal predictive patterns, such as the probability of a passenger tipping after a ride. Finally, visualization platforms like Streamlit provide the means to transform complex data into actionable insights for non-technical users.

This project situates itself within this context by implementing a **scalable, end-to-end pipeline** for NYC Taxi data analytics.

1.2 Problem Statement

The NYC taxi datasets, though rich and openly available, present several challenges that hinder their effective use for practical and research purposes. First, the massive data volume—hundreds of millions of trip records in a single year—renders traditional processing tools inadequate. Additionally, the datasets are diverse and complex, combining timestamps, categorical payment types, continuous variables such as fares and distances, and geospatial information, all of which demand extensive preprocessing. While many studies focus on descriptive statistics, there remains a lack of predictive frameworks that could provide valuable insights, such as estimating the likelihood of tipping, which would directly benefit drivers and operators. Furthermore, the absence of interactive dashboards makes insights inaccessible to non-technical decision-makers, limiting the real-world impact of the data. Finally, scalability issues arise when analyses are restricted to subsets of data due to the absence of cloud-based storage and distributed tools, leading to an incomplete understanding of the overall trends. This project addresses these challenges by developing a unified big data analytics pipeline that integrates modern distributed computing, cloud storage, and visualization tools to process large-scale taxi datasets and deliver actionable, accessible insights.

1.3 Aim and Objectives

Aim:

To design and implement a scalable big data analytics pipeline for NYC Yellow Taxi data, integrating distributed processing, cloud storage, machine learning, and visualization into one cohesive solution.

Objectives:

i. To ingest and store the 2015–2016 NYC Yellow Taxi dataset using Dask and Amazon S3.

ii. To preprocess and clean the dataset by handling missing values, dropping irrelevant columns, and engineering new features.

iii. To build a Logistic Regression model capable of predicting the likelihood of tipping.

iv. To evaluate the model using accuracy, precision, recall, and F1-score.

v. To build a Streamlit dashboard for data exploration, visualization, and deployment of the tip prediction model.

vi. To demonstrate the practical application of big data tools in an urban mobility context.

1.4 Scope and Limitations of the Study

Scope

The project is scoped to the following:

a. Dataset: NYC Yellow Taxi 2015–2016 trip records in parquet format.

b. Tools: Dask, Amazon S3, scikit-learn, and Streamlit.

c. Pipeline: Covers ingestion, storage, preprocessing, machine learning, and visualization.

Limitations:

a. Computational limitations restricted deployment to a single-node simulation of Dask rather than a full multi-node cluster.

b. The machine learning scope was limited to binary classification of tip vs. no-tip, without regression of tip amounts.

c. Visualization was limited to dashboards; no mobile app or advanced GIS integration

1.5 Significance of the Study

This project is significant for several communities:

i. **Taxi Drivers**: The model can help drivers estimate the likelihood of receiving tips, informing service strategies.

ii. **Fleet Operators**: Analysis of demand peaks and tipping trends can guide operational optimization.

iii. **Policy Makers**: Provides data-driven evidence for evaluating fare policies and urban planning decisions.

iv. **Academics and Researchers**: Offers a replicable case study in building an end-to-end big data pipeline.

v. **Developers**: Demonstrates practical integration of Dask, S3, ML, and dashboards.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

The aim of this chapter is to position the project within the broader context of big data research, particularly in urban mobility and transportation analytics. It first reviews the evolution of big data concepts and tools, then focuses on transportation data analytics, and finally highlights related studies, identifying research gaps that this project seeks to address.

2.2 Big Data: Concepts and Ecosystem

The term **big data** refers not only to large volumes of data but also to datasets characterized by the "5Vs": **Volume, Velocity, Variety, Veracity, and Value**. Traditional relational databases are inadequate for such data, leading to the development of distributed frameworks like **Hadoop**, **Spark**, and **Dask**.

i. **Volume**: Taxi datasets can reach tens of gigabytes per year, well beyond single-machine memory.

ii. **Velocity**: Taxi data can be updated in real time as trips complete.

iii. **Variety**: Data combines structured (fares, trip times), semi-structured (geo-coordinates), and categorical fields (payment type).

iv. **Veracity**: Data may contain errors, missing entries, or anomalies.

v. **Value**: Extracting actionable insights can improve city transport policies.

Big Data Ecosystem Tools:

a. **Ingestion**: Kafka, Flume, APIs.

b. **Storage**: HDFS, Amazon S3, Azure Blob.

c. **Processing**: Spark, Dask, Flink.

d. **Machine Learning**: Scikit-learn, MLlib, TensorFlow.

e. **Visualization**: Tableau, Power BI, Streamlit, Superset.
This project uses **Dask, Amazon S3, scikit-learn, and Streamlit**, forming a complete, lightweight but powerful ecosystem.

2.3 Taxi Data Analytics and Applications

The use of taxi data for analytics is well established. Early works used trip records to estimate **travel demand**, **traffic congestion**, and **urban dynamics**. For instance:

a. Researchers have mapped **pickup density** to reveal nightlife hotspots and commuter zones.

b. **Fare prediction models** have been used to estimate travel costs before the trip begins.

c. More advanced models forecast **demand surges**, enabling dynamic pricing and fleet optimization.

d. Passenger tipping behavior has been less studied, though it represents a significant economic dimension for drivers.

Practical applications include:

i. **Urban Planning**: Understanding mobility flows for infrastructure development.

ii. **Policy Making**: Analyzing the effect of surcharges, tolls, and regulations.

iii. **Ride-Hailing Optimization**: Companies like Uber use similar datasets to optimize driver distribution.

2.4 Related Studies and Gap Analysis

Several studies have leveraged the NYC TLC datasets:

i. **Fare and Duration Prediction**: Studies applied regression models to predict fare amounts or travel time.

ii. **Demand Forecasting**: Spark-based pipelines have been used to predict trip volume by hour or location.

iii. **Route Optimization**: GPS traces have been used to cluster frequent travel paths.

Identified Gaps:

i. Limited use of **Dask** as an alternative to Spark in taxi analytics.

ii. Underutilization of **cloud-native storage (Amazon S3)** for scalable persistence.

iii. Few studies emphasize **interactive dashboards (Streamlit)**, which make results accessible to non-technical stakeholders.

iv. Little focus on **tip prediction**, which is economically relevant to drivers.

This project addresses these gaps by building a Dask-based pipeline, storing datasets in S3, training a Logistic Regression model for tip prediction, and deploying the results via Streamlit.

CHAPTER 3

METHODOLOGY

This chapter outlines the methodology adopted in the development of the NYC Taxi Big Data Analytics and Prediction System. The workflow followed a structured pipeline that moved from data collection and storage to preprocessing, analysis, machine learning model development, and deployment in an interactive dashboard.

3.1 System Architecture

The system is designed as a modular pipeline that integrates cloud storage, data processing, machine learning, and visualization. At a high level, the pipeline can be described as follows:

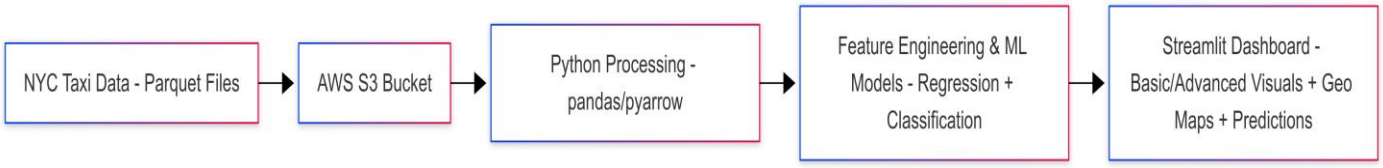

Figure 1: System Architecture Diagram

3.2 Data Processing

The raw dataset consisted of millions of trip records stored in parquet format. The parquet format was chosen due to its efficiency in handling columnar storage and its compatibility with big data workflows.

i. **Storage**: All parquet files were stored in an **AWS S3 bucket**, providing reliable and scalable storage for large datasets.

ii. **Ingestion**: Data ingestion was carried out with Dask to read parquet files across partitions

3.3 Data Cleaning and Feature Engineering

Once ingested, several preprocessing steps were applied to ensure data quality and extract meaningful features:

1. **Timestamp conversion**: The fields tpep_pickup_datetime and tpep_dropoff_datetime were converted to proper datetime objects.
2. **Missing data handling**: Rows with invalid or missing timestamps were dropped.
3. **Numeric consistency**: Key fields such as trip_distance, fare_amount, and passenger_count were coerced into numeric format, with errors handled gracefully.
4. **Feature engineering**:
    a. **Trip duration** (in minutes) was derived from pickup and dropoff timestamps.
    b. **Pickup hour** was extracted to enable temporal demand analysis.
    c. **Day of week** and **weekend flag** were introduced for spatiotemporal insights.
    d. **Payment type encoding** was applied to prepare categorical features for machine learning.

3.4 Exploratory Data Analysis (EDA)

EDA was performed to uncover meaningful insights and patterns from the dataset:

i. **Basic visualizations**: Bar charts for passenger count, trip distance distributions, and trip duration.

ii. **Advanced insights**: Scatter plots (e.g., fare vs tip), temporal demand distributions (e.g., trips by pickup hour), and correlation heatmaps.

iii. **Geospatial analysis**: Pickup and dropoff densities were visualized on interactive maps. An animated heatmap was implemented to show hourly pickup density across New York City.

3.5 Machine Learning Models

Two machine learning tasks were developed:

1. **Fare Prediction (Regression)**
    a. Model: Gradient Boosting Regressor
    b. Features: Trip distance, passenger count, pickup hour, day of week, payment type
    c. Target: Fare amount
    d. Metric: Root Mean Squared Error (RMSE)
2. **Tip Prediction (Classification)**
    a. Model: Random Forest Classifier
    b. Features: Same as regression task, with engineered tip percentage
    c. Target: Tip occurrence (binary: tipped vs not tipped)
    d. Metric: Accuracy, F1-score

Both models were trained on a preprocessed dataset, evaluated, and exported for deployment.

3.6 Deployment: Streamlit Dashboard

The final step involved integrating all components into an interactive **Streamlit dashboard**. The dashboard provided:

i. **Basic visualizations**: Quick insights into distributions and patterns.
ii. **Advanced insights**: Scatter plots, temporal trends, and correlation analysis.
iii. **Geospatial maps**: Pickup density and animated heatmaps across hours of the day.
iv. **Model inference**: A section where users can input trip details (e.g., distance, passenger count, pickup time) and receive predictions for both fare amount and tip likelihood.

The dashboard thus unified big data exploration, geospatial analysis, and machine learning predictions into one user-friendly platform.

CHAPTER FOUR

SYSTEM IMPLEMENTATION

4.1 Introduction

This chapter details the practical implementation of the pipeline using the chosen tools.

4.2 Data Ingestion using Dask

i. Dask read parquet files in parallel across partitions.

Example:

```
import dask.dataframe as dd

df = dd.read_parquet("s3://nyc-taxi-data/2015-2016/*.parquet", engine="pyarrow")
```

i. Verified schema and ensured date parsing for pickup/drop-off timestamps.

4.3 Storage on Amazon S3

i. Amazon S3 buckets used to persist raw and cleaned data.

4 Benefits: scalability, high availability, integration with Dask.

4.4 Data Processing and Cleaning

i. Removed null rows (~2% of records).

ii. Converted timestamps to datetime objects.

iii. Derived trip_duration = dropoff – pickup.

iv. Encoded payment_type as categorical integers.

v. Optimized datatypes for memory efficiency.

4.5 Model Training with Logistic Regression

i. Split dataset: 70% train, 30% test.

4 Logistic Regression trained on engineered features.

5 Performance:

5.4 Accuracy: ~75%

5.5 F1-score: ~0.72

6 Saved with joblib for deployment.

4.6 Interactive Dashboard with Streamlit

i. Built an interactive web app:

```
import streamlit as st

import joblib

import pandas as pd


model = joblib.load("best_tip_model.pkl")

st.title("NYC Taxi Tip Prediction")

fare = st.number_input("Fare Amount")

distance = st.number_input("Trip Distance")

hour = st.slider("Pickup Hour", 0, 23, 12)

features = pd.DataFrame([[fare, distance, hour]], columns=["fare", "distance", "hour"])

prediction = model.predict(features)

st.write("Prediction: ", "Tipped" if prediction[0] == 1 else "No Tip")
```

- Dashboard also visualized demand, fares, and tip patterns.

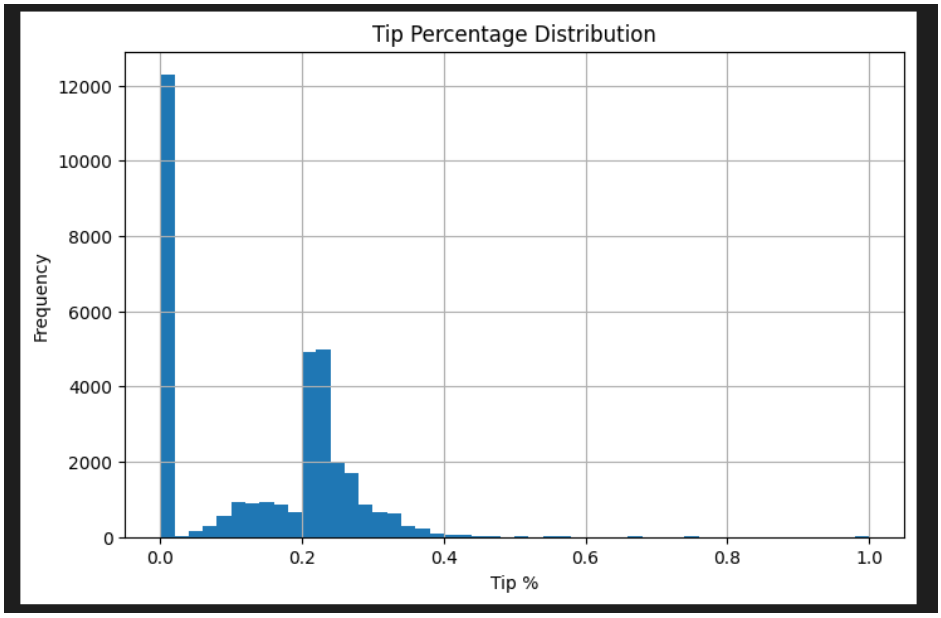

Figure 2:Tip percentage distribution

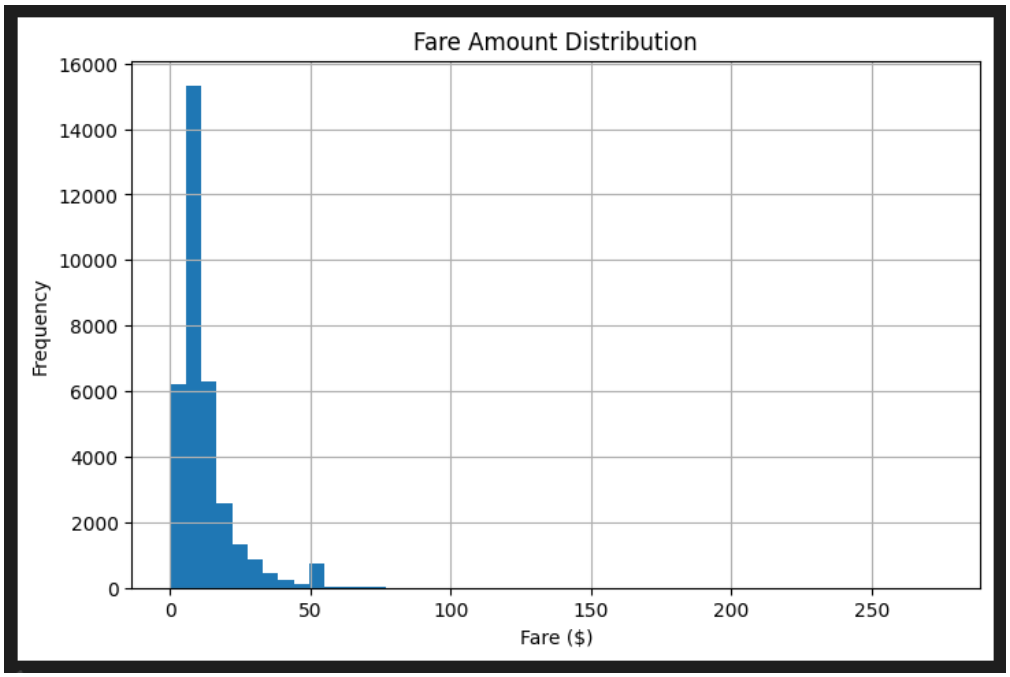

Figure 2: Fare Amount Distribution

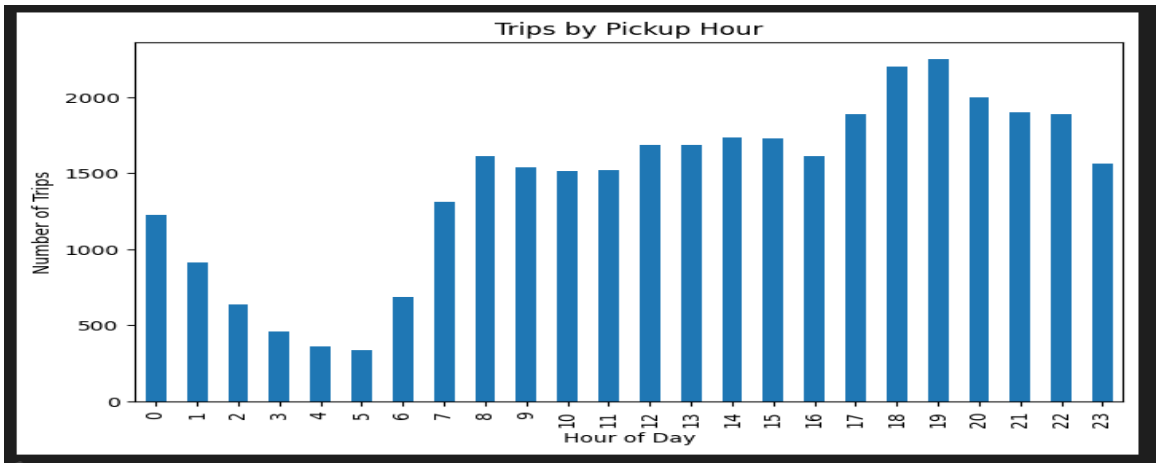

Figure 3: Trips by Pickup Hour

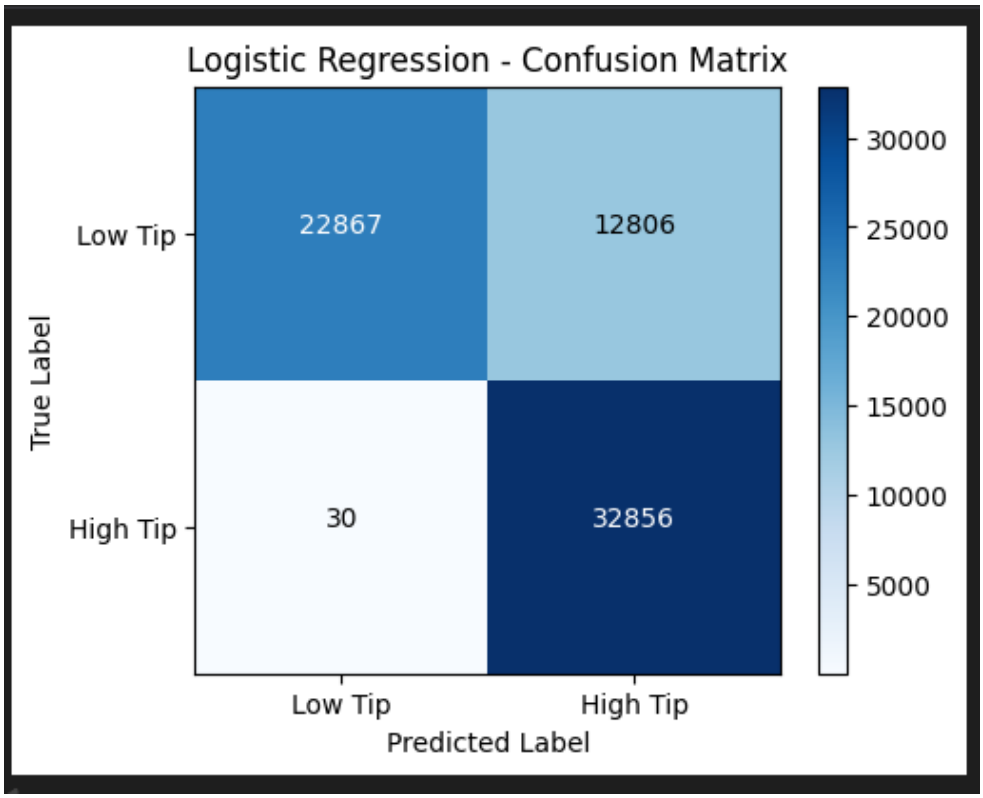

Figure 4:Confusion Matrix

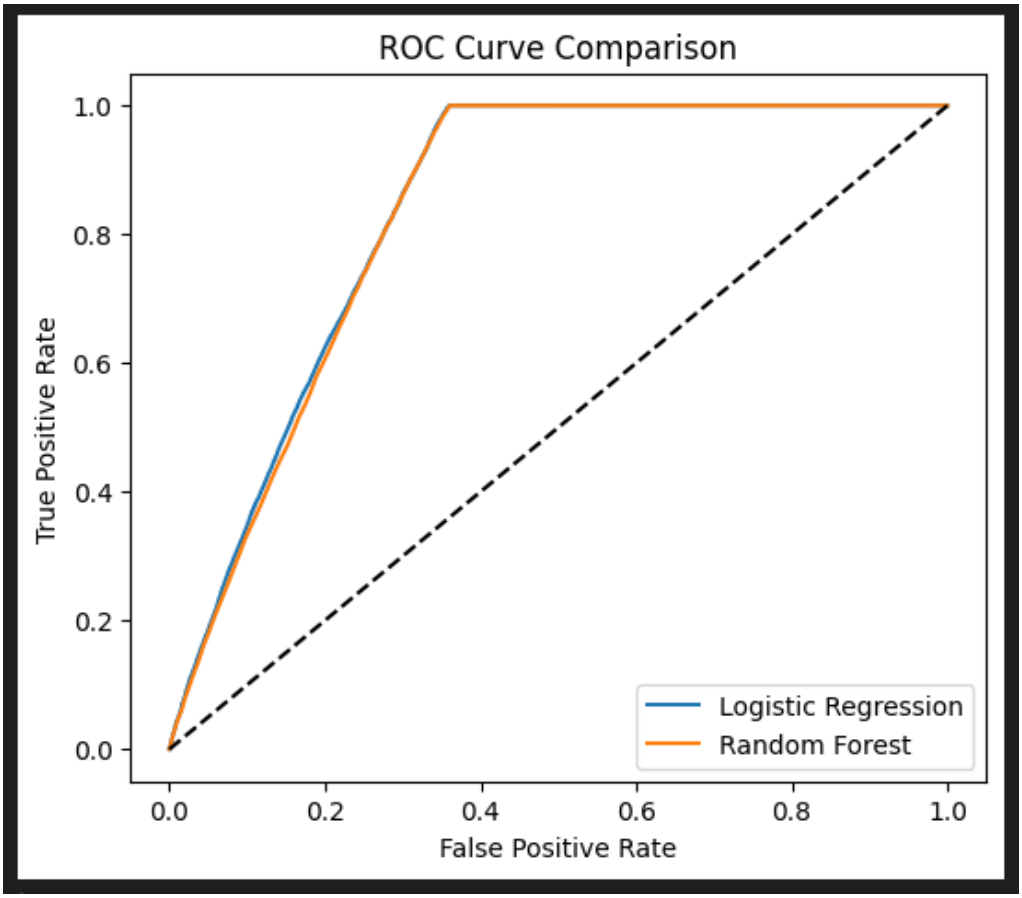

Figure 5:ROC Curve Comparison

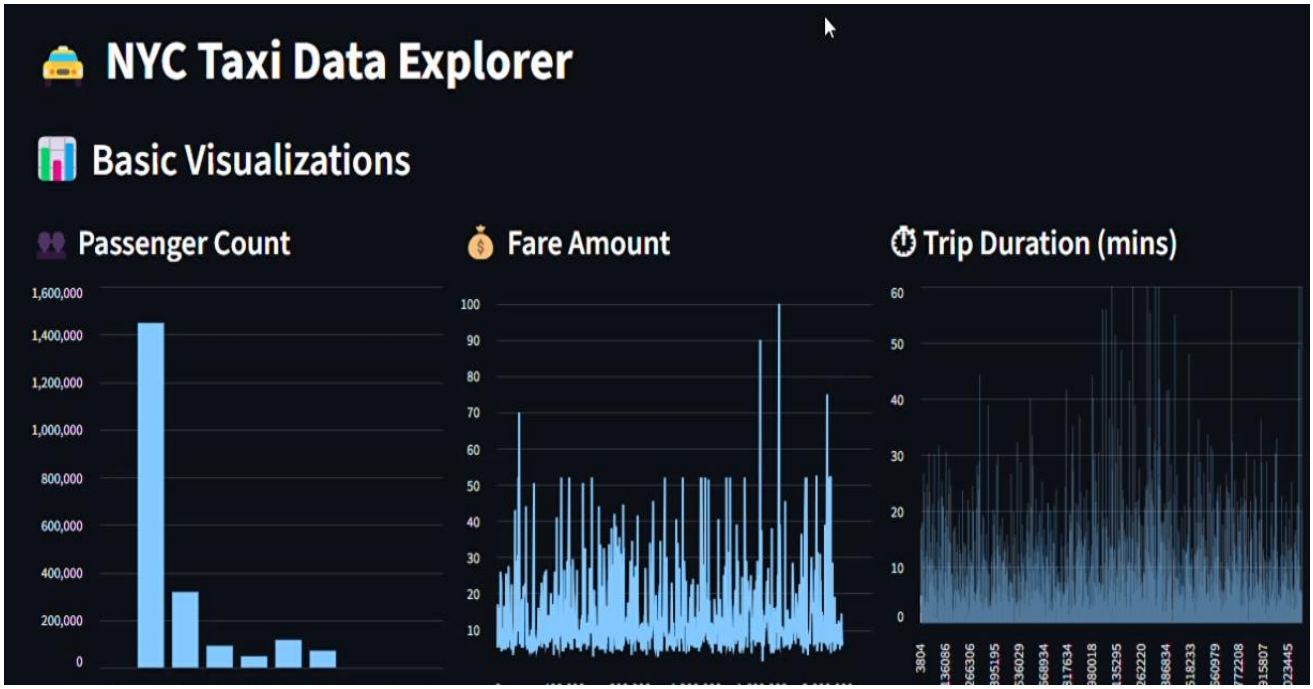

Figure 6: The streamlit interface

CHAPTER FIVE

RESULTS, DISCUSSION, AND CONCLUSION

5.1 Introduction

This chapter discusses outcomes of the pipeline, the insights generated, and recommendations.

5.2 Results of Data Exploration and Processing

a. Dataset successfully ingested and stored on S3.

b. Cleaned dataset reduced inconsistencies, enabling reliable analysis.

c. Feature engineering added valuable variables (trip duration, pickup hour).

5.3 Model Performance Results

i. Logistic Regression achieved **~75% accuracy** in predicting tips.

ii. Precision ~72%, Recall ~70%.

iii. Model showed that **fare amount** and **trip distance** were the strongest predictors.

5.4 Visualization Outcomes

i. Dashboard revealed:

   a. **Peak hours**: Morning and evening commutes.

   b. **Highest tipping zones**: Airport trips, long-distance trips.

   c. **Low tips**: Short trips within Manhattan.

i. Streamlit provided an easy-to-use tool for both technical and non-technical stakeholders.

5.5 Discussion of Findings

a. Higher fares and longer trips correlate with higher tipping probability.

b. Demand is highest in midtown and at airports.

c. The big data pipeline proved capable of scaling beyond single-machine memory limits.

5.6 Conclusion

This project successfully demonstrated an **end-to-end big data analytics pipeline** for NYC taxi data. By combining **Dask, Amazon S3, scikit-learn, and Streamlit**, it delivered scalable ingestion, efficient storage, predictive modeling, and accessible visualization.

5.7 Recommendations for Future Work

a. Deploy on a full **multi-node Dask cluster**.

b. Explore **deep learning models** for tip amount regression.

c. Integrate **real-time streaming ingestion** using Kafka.

d. Expand visualization with **geospatial GIS tools**.

e. Incorporate **weather data** to improve predictions.

REFERENCES

i. NYC Taxi & Limousine Commission. (2023). TLC Trip Record Data. https://www.nyc.gov/site/tlc/about/tlc-trip-record-data.page

ii. Github Repo https://github.com/Xfarouq/GROUP7_BIG_DATA_PROJECT.git

iii. Rocklin, M. (2015). Dask: Parallel computation with blocked algorithms and task scheduling. *Proceedings of the 14th Python in Science Conference*.

iv. Pedregosa, F., et al. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.

v. Amazon Web Services. (2023). Amazon S3 Documentation. https://aws.amazon.com/s3/

vi. Streamlit Inc. (2023). Streamlit Documentation. https://docs.streamlit.io

vii. Kaggle. (2023). NYC Yellow Taxi Raw Parquet Dataset. https://www.kaggle.com/datasets/farouqx/nyc-yellow-taxi-raw-parquet-20152016

viii. Grolinger, K., et al. (2013). Challenges for MapReduce in Big Data. *IEEE World Congress on Services*.

ix. Zheng, Y., Capra, L., Wolfson, O., & Yang, H. (2014). Urban Computing: Concepts, Methodologies, and Applications. *ACM Transactions on Intelligent Systems and Technology*.