BIG DATA ANALYTICS Project

Team members:

ID	Name
22010383	مازن محمد محمد احمد
22010107	سامي عادل
22010317	احمد فكري عبد الحميد
22011460	عبد الرحمن محمد محمود
22010332	خالد ربيع عبد الجواد

1. Introduction

Urban traffic systems are highly sensitive to weather conditions such as rain, temperature extremes, humidity, wind, and visibility. Understanding how these factors influence traffic congestion and accident risks is critical for smart city planning and traffic management.

This project aims to design and implement a **modern predictive Big Data Lake system** to analyze the impact of weather on urban traffic in **London**. The system follows a **Medallion Architecture (Bronze → Silver → Gold)** using **MinIO**, with **HDFS** as an additional distributed storage layer. Advanced analytics techniques such as **Monte Carlo Simulation** and **Factor Analysis** are applied to generate actionable insights.

---

2. Project Architecture Overview

The pipeline is implemented as follows:

1. **Bronze Layer (Raw Data)**
   - Raw synthetic traffic and weather datasets stored in MinIO.
2. **Silver Layer (Cleaned Data)**
   - Data cleaning, validation, and transformation using Python.
   - Cleaned datasets stored in Parquet format.
3. **HDFS Integration**
   - Cleaned Parquet files copied to HDFS for distributed storage.
4. **Gold Layer (Analytics & Insights)** Monte Carlo Simulation results.
   - Factor Analysis outputs (scores and loadings).
   -

---

3. Data Description

3.1 Traffic Dataset

- Approximately **5,050 rows** before cleaning.
- Contains vehicle counts, congestion levels, accidents, road conditions, and visibility.
- Includes duplicates, missing values, outliers, and inconsistent date formats.

3.2 Weather Dataset

- Approximately **5,050 rows** before cleaning.
- Includes temperature, humidity, rain, wind speed, visibility, and air pressure.
- Contains missing values, extreme outliers, and invalid ranges.

4. Data Cleaning & Quality Report (Bronze → Silver)

Summary of Fixes Applied

remove 50 duplicate rows

replace missing traffic id with index +1
The dataset contained multiple inconsistent date formats such as:

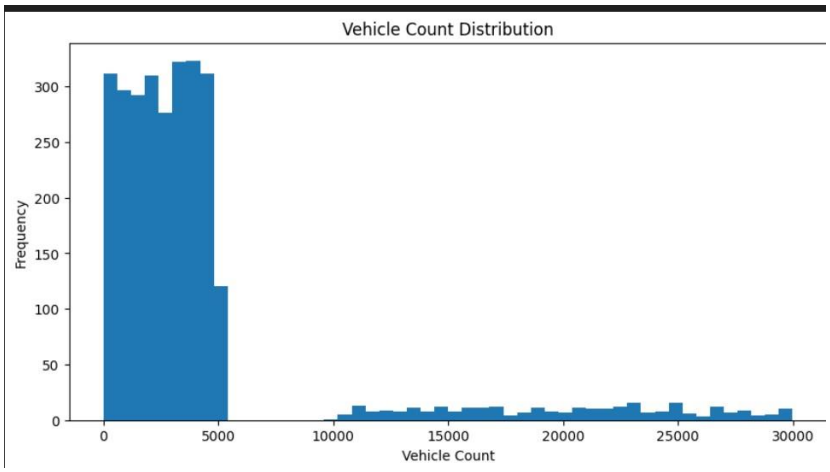
- `15/01/2024 8AM`
- `2024-01-15T08:00Z`
- Invalid entries like `TBD` or `2099-00-00 99:99`

All valid timestamps were converted into a consistent format:
YYYY-MM-DD HH:MM
Invalid timestamps were removed.

Handling Missing Values

- Missing `area` values were imputed using the most frequent category (mode).
- Missing values in categorical fields like `congestion_level` and `road_condition` were filled.

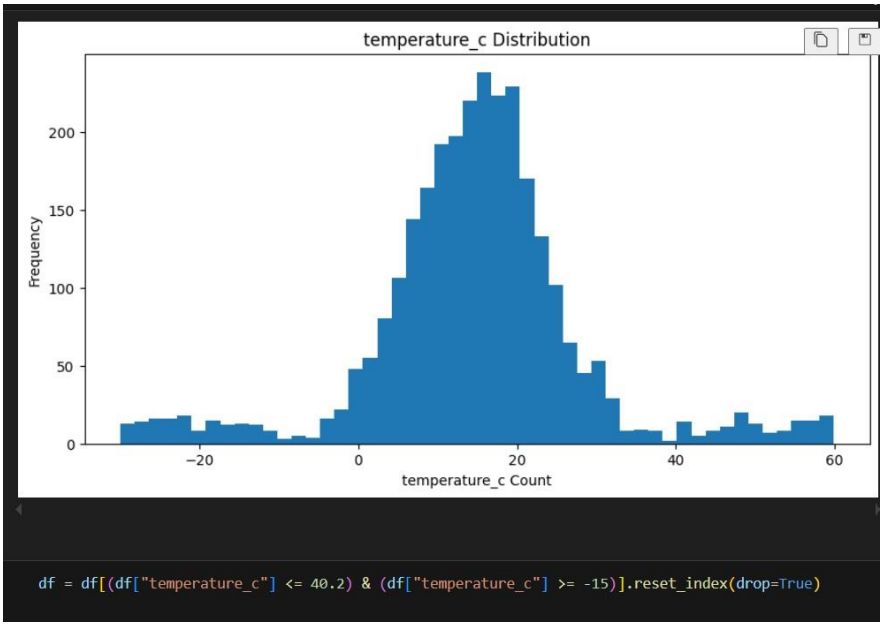
in city we replace nan with London because there is only value on that column in vehicle_count .
we draw histogram use IQR to get the outlier and replace it with median



and finally we convert ["traffic_id", "vehicle_count", "accident_count", "visibility_m"] data type into integer

weather data:

Number of rows before cleaning 5050 Number of rows after cleaning 2599
weather id ,date ,city we did like we did with traffic season column we replace missing value with mode.



we remove rows that temperature greater then 40.2 because this is the greater degree in weather of London was record and less then -15 also and replace missing value with median.

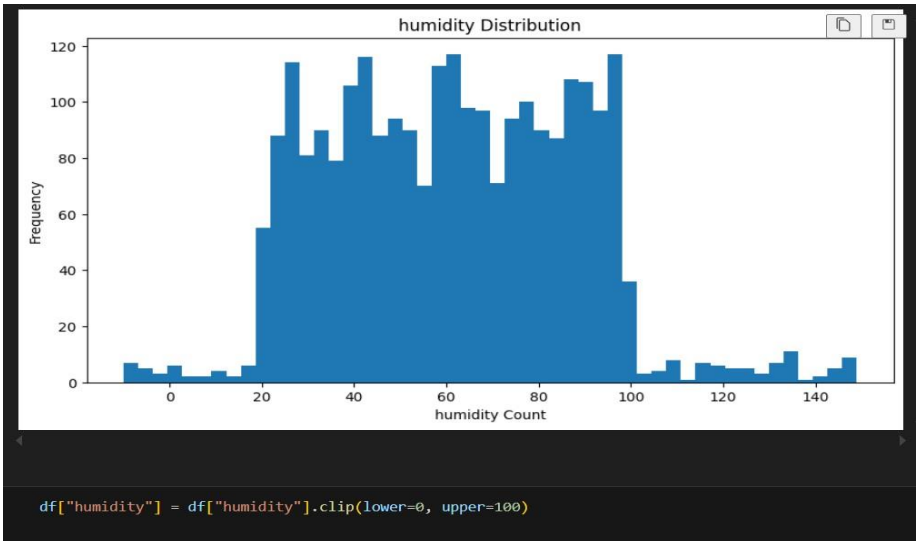
---

humidity range is 0 100 so any values greater than 100 or less that 0 is invalid and value greater than 100 will be replace with 100 and less than 0 will replace with 0

because we want to save rows and if remove those rows, data will decrease

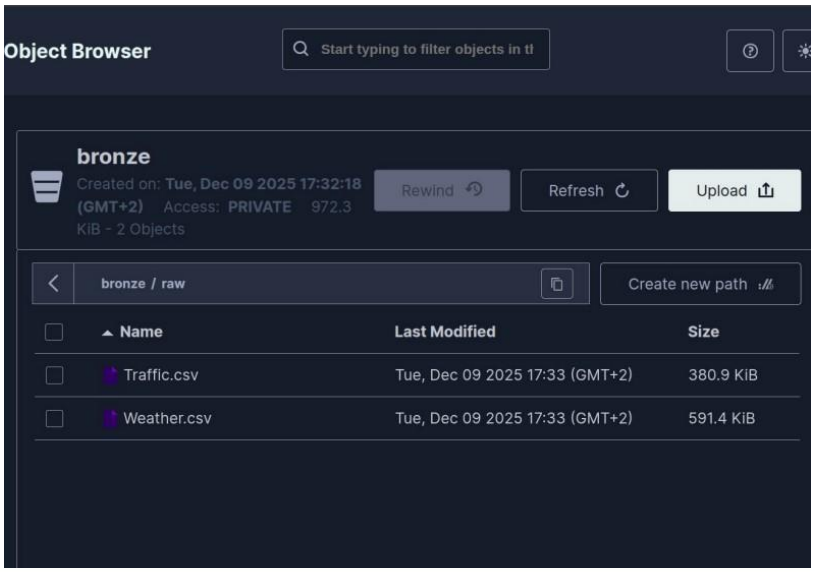
in rain and wind

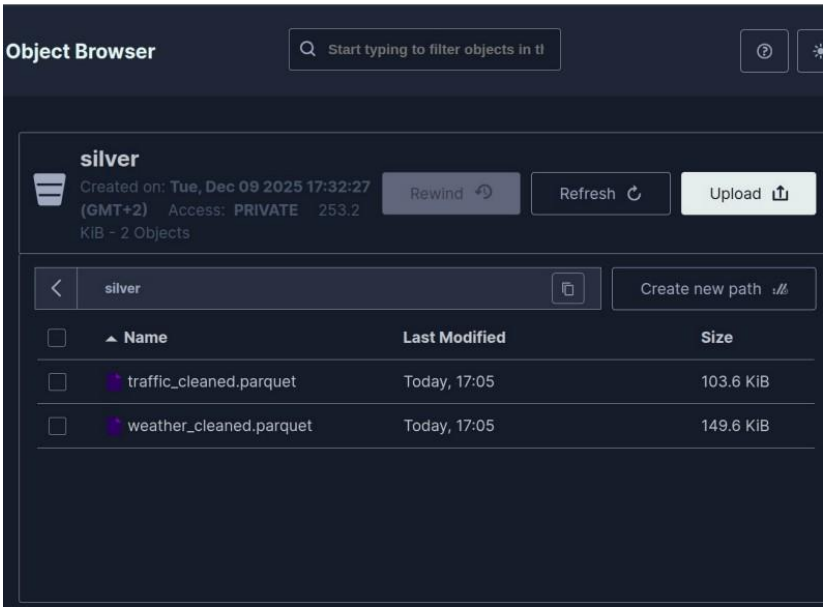
we replace missing with median also
["weather_id", "visibility_m"] convert data type into integer

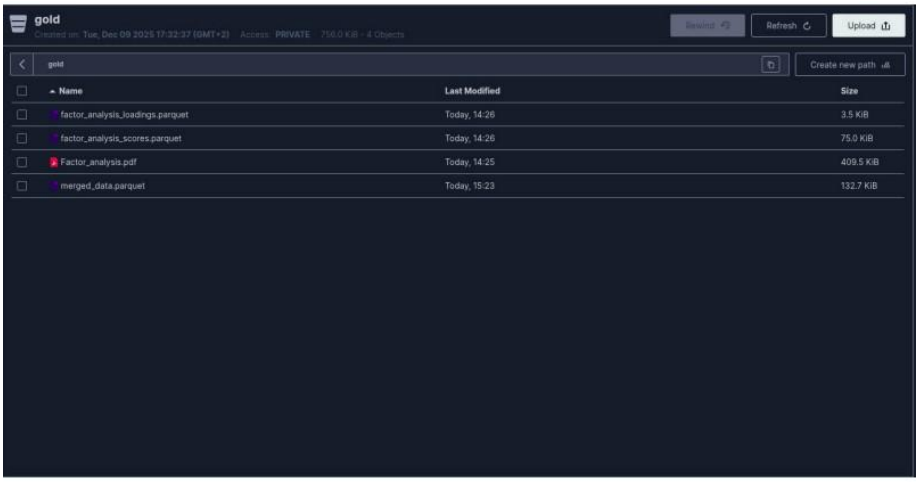


6. MINIO

we created three buckets (bronze, silver,gold)



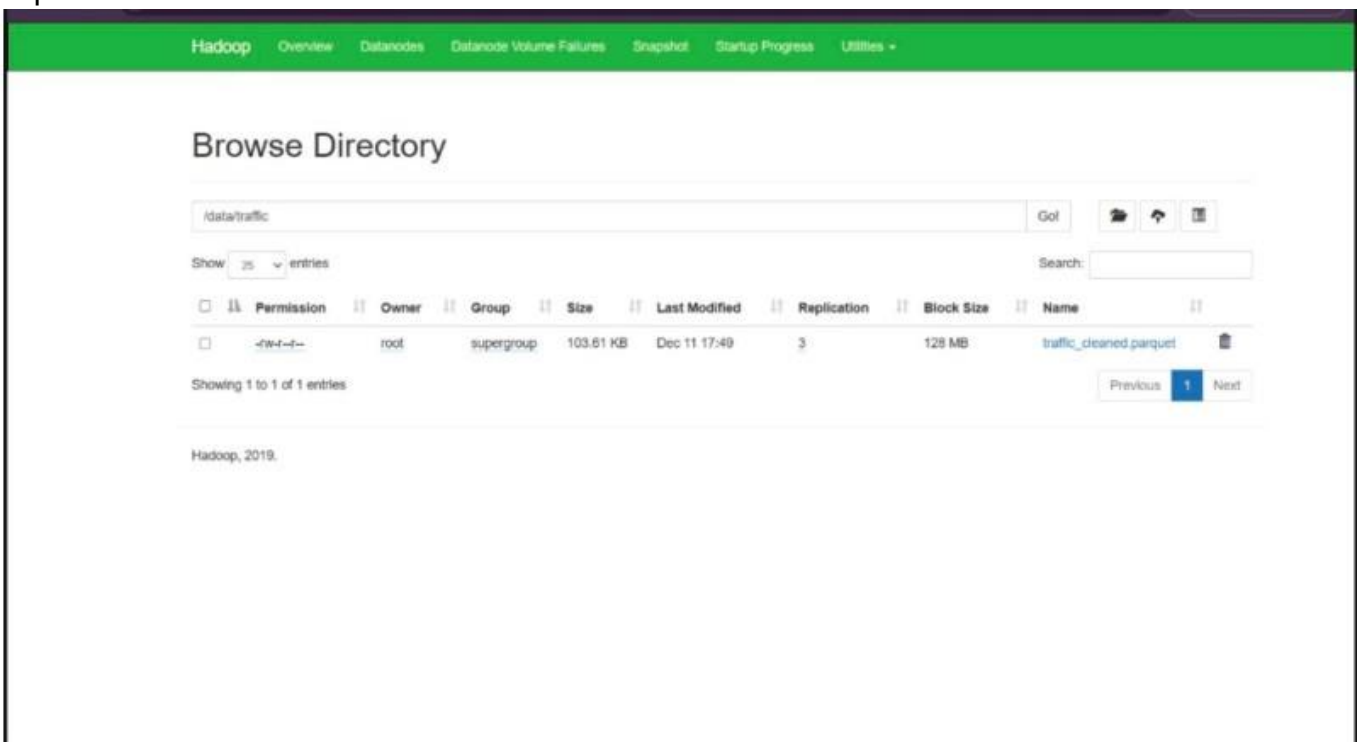




5. HDFS Integration

After cleaning, both datasets were saved as **Parquet files** in the **Silver layer** and then copied into **HDFS**.

This step adds scalability and fault tolerance, enabling distributed processing for future expansion.



The cleaned traffic and weather datasets were merged using:

- date_time
- city

This produced a unified analytical dataset used for simulations and statistical analysis.

7. Factor Analysis – Weather Impact Detection

7.1 introduction

Factor Analysis (FA) is a statistical technique used to uncover latent structures—referred to as factors—that explain correlations among observed numerical variables. By reducing the dimensionality of a dataset, FA highlights hidden relationships, simplifies complex data, and supports more efficient downstream analytics. In this project, Factor Analysis was applied to the cleaned dataset that merged and stored at merged data. The resulting analytical outputs were stored in the Gold layer, following the project's Medallion Architecture (Bronze → Silver → Gold), ensuring reliable, curated, and high-quality data for further analysis.

7.2 Methodology

7.2.1 Load Cleaned Data

The cleaned Silver-layer dataset was loaded for statistical analysis

7.2.2 Select Numeric Variables

Only numerical features were retained, as Factor Analysis requires continuous numeric inputs.

7.2.3 Standardize the Data

Standardization transformed all numeric features to have zero mean and unit variance, ensuring each variable contributes equally to the factor extraction process.

7.2.4 Apply Factor Analysis

A Factor Analysis model was configured to extract three latent factors. The model decomposed the standardized dataset into: • Factor Scores: Representation of each observation in terms of latent factors. • Factor Loadings: Contribution strength of each variable to each factor.

7.2.5 Save Outputs

Factor scores and loadings were saved to the Gold layer for downstream analytics.

7.3 Results

7.3.1 Factor Scores

• File: gold/factor analysis scores.parquet • Description: Numerical representation of each observation in terms of the three extracted factors.

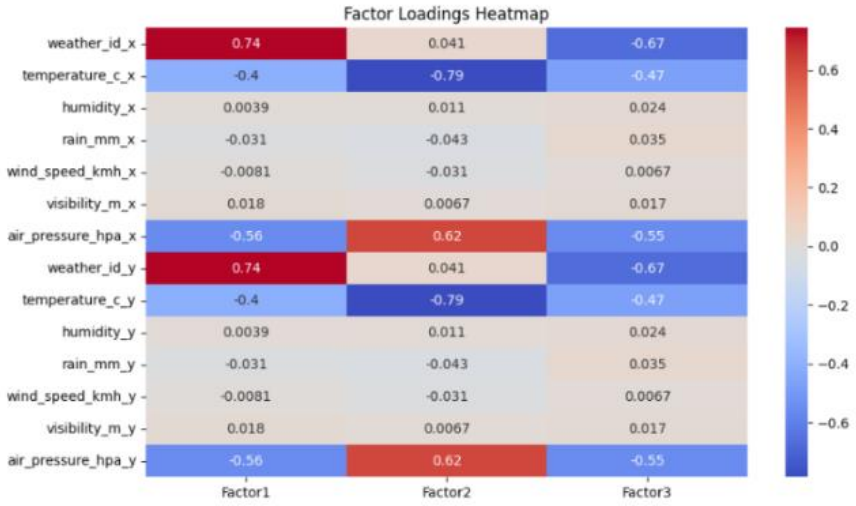
7.3.2 Factor Loadings

• File: gold/factor analysis loadings.parquet
• Description: Contribution strength of each variable to each factor.

7.4 Visualizations

7.4.1 Factor Loadings Heatmap

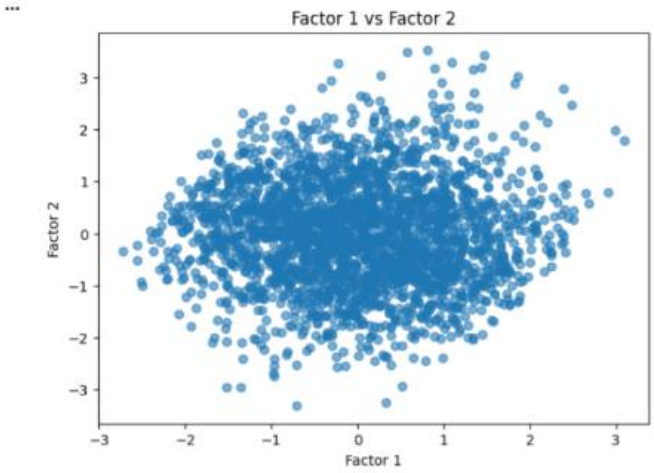
Other variables (humidity, wind speed, rain, visibility) have low loadings, indicating minor influence.



7.4.2 Scatter Plots of Factor Scores

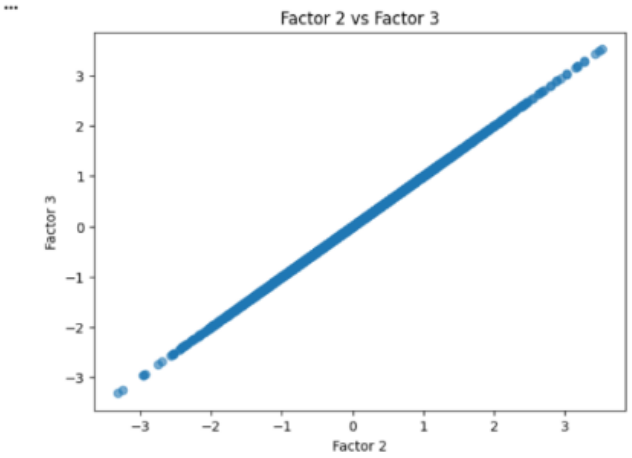
7.4.2.1 Factor 1 vs Factor 2

Points are dispersed, indicating independence between these factors. Useful for clustering and dimensionality reduction.



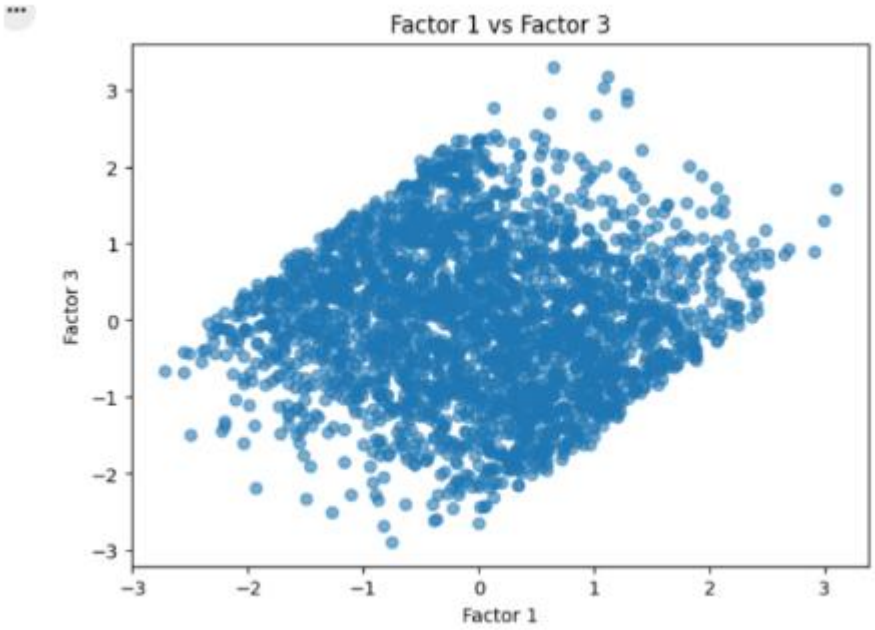
7.4.2.2 Factor 2 vs Factor 3

Points show a strong linear trend, indicating high correlation. Suggests potential redun dancy; these factors may represent overlapping constructs



7. 4.2.3 Factor 1 vs Factor 3

Points are moderately dispersed, indicating partial independence. Factor 1 contributes unique information relative to Factor 3.



The Factor Analysis revealed three latent factors capturing key patterns in the weather related dataset: 1. General weather conditions 2. Pressure-temperature dynamics 3. Secondary weather patterns Scatter plots indicate that Factor 2 and Factor 3 are highly correlated, which may require further consideration in downstream modeling. Factors 1 and 3 provide comple mentary information, supporting dimensionality reduction and exploratory insights. All factor scores and loadings are saved in the Gold layer, ensuring availability for predictive modeling, clustering, or further statistical analysis.

8. Monte Carlo Simulation – Traffic Risk Assessment

To quantify traffic risks using a **Monte Carlo Simulation** based on historical data. The key required outputs were:

- Probability of Traffic Jams.
- Accident Risk specifically under "Bad Weather" conditions.
- The distribution of congestion severity across 10,000 simulated scenarios.

Methodology: The Empirical Monte Carlo Approach

Instead of using Machine Learning models (which were prone to accuracy issues due to class imbalances we have), we implemented a **Statistical Lookup Model**. This approach uses historical base rates to drive the simulation, ensuring results are grounded in reality.

We created a "Risk Table" by grouping historical data by `road_condition`. This established the baseline probability for Jams and Accidents for every possible scenario.

Distributions:

- **Bernoulli Distribution:** Used for binary events (Did a jam happen? Yes/No).
- **Normal Distribution:** Used for continuous severity (How bad is the congestion?), defined by historical Mean and Standard Deviation.

Generate Samples: We generated **100,000 synthetic days**, simulating various weather and road conditions. Also, we made sure that the generation was chained, meaning that some generation of attributes of a day, depends on another generated attributes on the same day. To make sure that we don't have a sunny day with extreme winds and rain.

Run Simulation: We merged the historical risks into the synthetic days and applied random sampling (`np.random.rand` and `np.random.normal`) to determine specific daily outcomes.

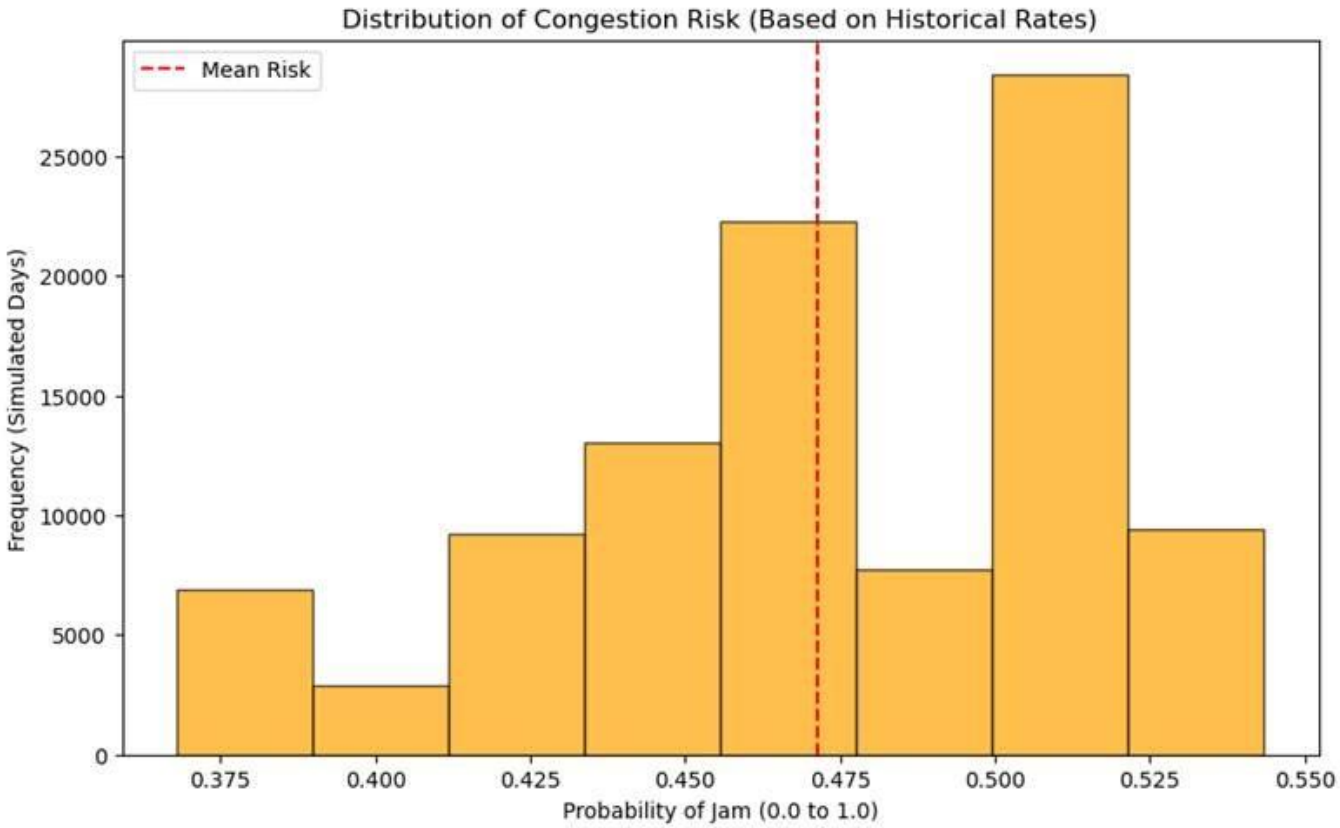
Analyze Results: We visualized the probability distributions and calculated aggregate risks.

Key Findings & Analysis

The Risk Profile (Histogram Analysis)

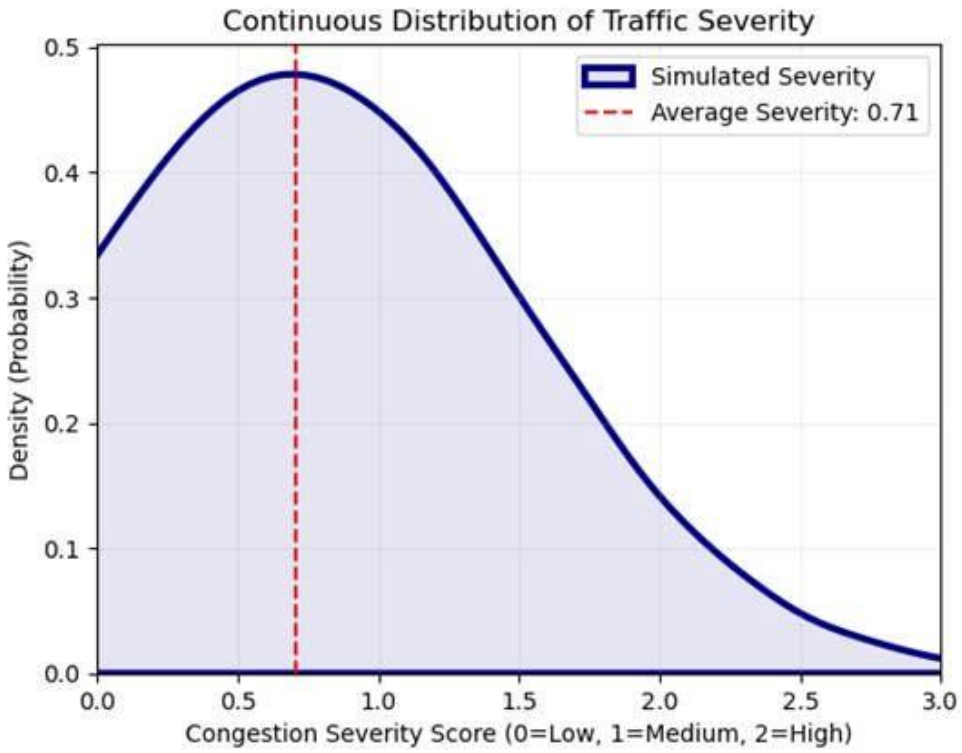
With baseline 37.5%, Even under ideal conditions, the system remains fragile with a high chance of congestion.

Bad weather acts as a "multiplier," pushing the probability of a jam over **50%**.



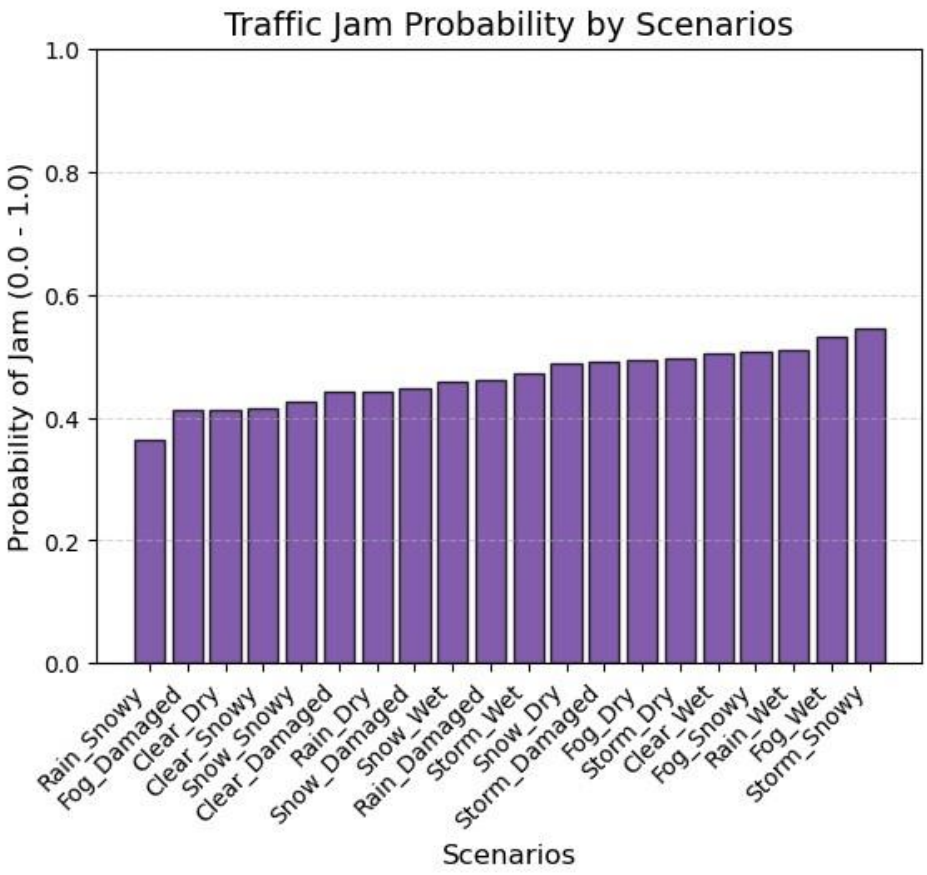
Continuous Distribution of Traffic Severity

The severity curve peaks at **0.65** (between "Low" and "Medium") but has a long tail extending past **2.0** ("High"). While the "average" day sees manageable traffic, the system is prone to extreme tail-risk event.



Visualizing our probabilities against different scenarios, we find that the peak is Storm_Snowy. This condition represents the absolute peak risk, with a jam probability of **~55%**.

Also, there are some anomalies in data that makes Rain_Snowy is surprisingly the **lowest** risk (~37%), even lower than `Clear_Dry` .



9. Final Insights & Conclusion

This project successfully implemented a **full Big Data analytical pipeline** from raw ingestion to advanced analytics. Key outcomes include:

- A robust **data cleaning and validation process**.
- A scalable **data lake architecture** using MinIO and HDFS.
- Realistic **traffic risk estimation** using Monte Carlo Simulation.
- Identification of **key weather drivers** affecting traffic through Factor Analysis.

Final Recommendation:

Weather conditions significantly amplify traffic congestion risks in London. Urban planners and traffic authorities should integrate probabilistic simulations and weather-aware strategies into traffic management systems to mitigate congestion and accident risks.