



Alexandria University

Faculty of Computers and Data Science
Department: Computing and Data Science
Course Title: Big Data Analytics 2025-2026

Weather Impact on Urban Traffic Analysis

In

Big Data Analytics

Course Code: 02-24-01401

Name:- *Mazen Shaban Jomaa* ID:- 22010201

Name:- *Osama Mohamed Abd-Elshafy* ID:- 22010047

Name:- *Omar Yahia Ibrahim* ID:- 22010174

Name:- *Fouad Ramzy Fouad* ID:- 22010180

Introduction

This project is about urban traffic analytics under weather conditions aims to design and implement a modern predictive data lake system to analyze how weather conditions affect urban traffic patterns using:

- synthetic data
- MinIO (**Bronze** / Silver / **Gold** layers)
- HDFS (as an additional distributed storage layer)
- Python for data processing
- Monte Carlo simulation
- Factor Analysis

It takes a smart city in London, wants to understand how weather conditions (rain, temperature extremes, humidity, wind, visibility) influence traffic behavior and congestion levels, as its scenario. It applies that through 7 phases as follows.

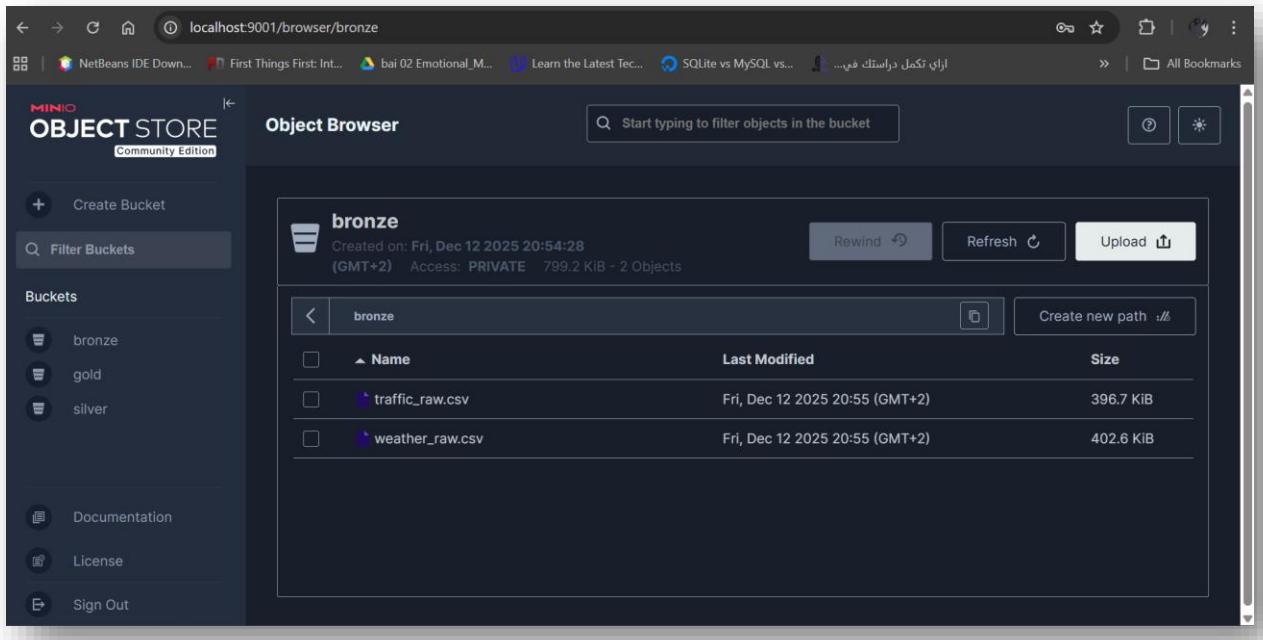
Phase 1: Infrastructure & Data Ingestion (**Bronze Layer**)

Included writing two Python scripts to generate the two synthetic datasets, weather and traffic, such that keeping the dependencies between columns there to mimic the real-life datasets.

Also, a *docker-compose.yml* was created to start MinIO server and create three MinIO buckets:

- ii) bronze → raw data
- iii) silver → cleaned data
- iv) gold → final results

The two generated raw datasets uploaded, then, to the bronze layer through the browser console of MinIO on the local host.



Phase 2: Data Cleaning (Bronze → Silver)

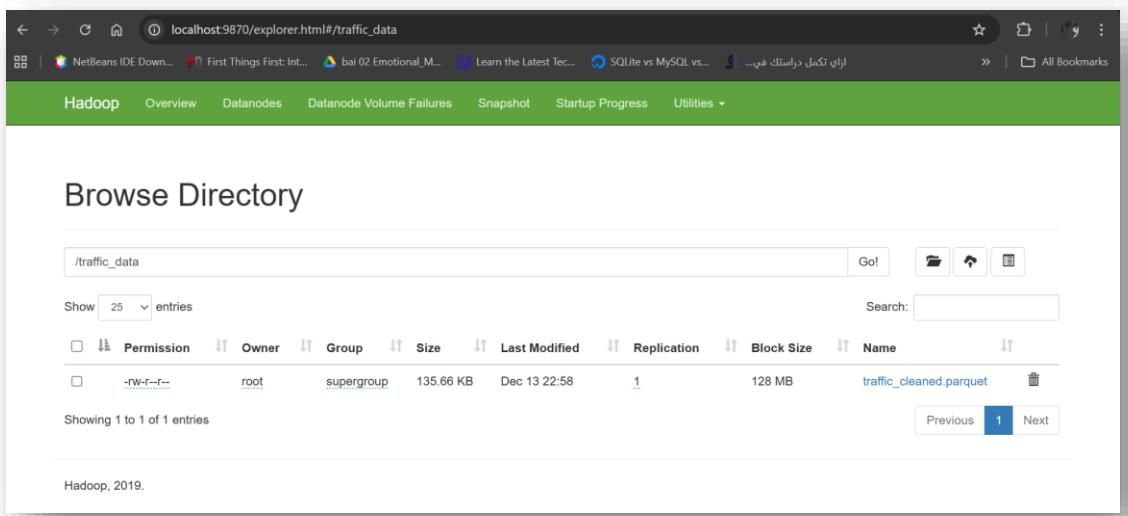
Transformed the messy raw data into clean analytical datasets. For each dataset, the process started by imputing the null values according to the problem with a specific column using pattern-based cleaning, XGBoost, etc. Then, it corrected the wrong format of some columns. After that, checking for duplicates which did not exist because of the imputation in the beginning. It, finally, checked for outliers statistically, using IQR, and graphically, using box plots and histograms to handle.

By finishing the cleaning process with no single deleted row, 5250 in 5250 out, both notebooks ended up connecting to the MinIO server to upload the cleaned version of both datasets into the silver bucket of MinIO as parquet files, in addition to saving other copies locally.

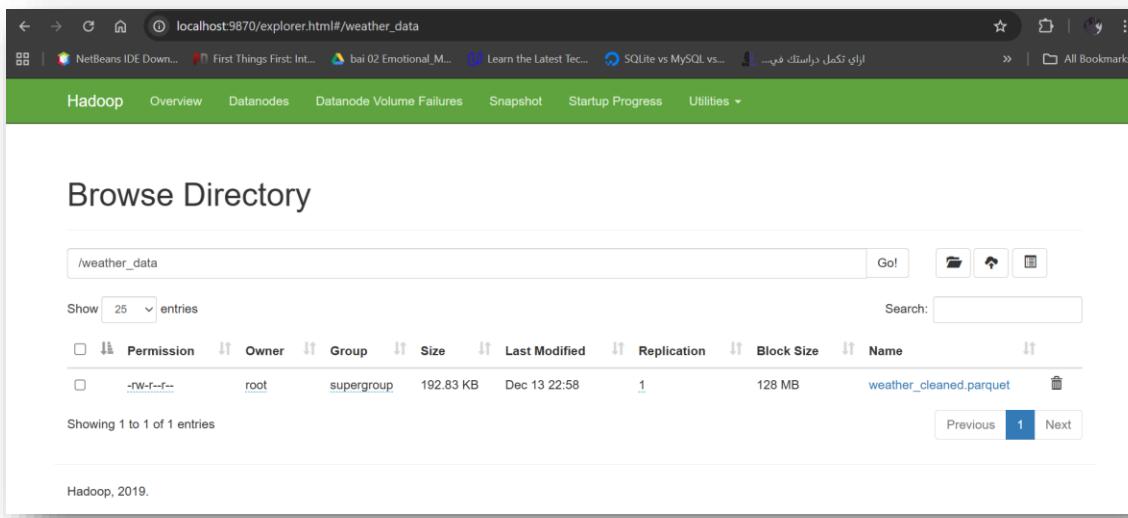
More explanation in detail is shown in the notebooks, *big data project E(1) V(2).ipynb* and *big data project traffic data set.ipynb*, themselves as markdown cells interpreting the code flow.

Phase 3: HDFS Integration (Additional Storage Layer)

Integrated the Hadoop distributed file system with the data lake using Python script file, *ingest_to_hdfs.py*, which connects to both, the MinIO and the HDFS servers, fetch the data files from the silver bucket, and streams the data (read RAM → write RAM). It opens the MinIO file as a stream and pass it directly to HDFS write storing in the *weather_data* and *traffic_data* folders.



The screenshot shows the Hadoop web interface at localhost:9870/explorer.html#/traffic_data. The page title is "Browse Directory". The URL in the address bar is /traffic_data. The search bar contains "/traffic_data". The table header includes columns: Permission, Owner, Group, Size, Last Modified, Replication, Block Size, and Name. There is one entry: "traffic_cleaned.parquet" owned by root, group supergroup, size 135.66 KB, last modified Dec 13 22:58, replication 1, block size 128 MB. The footer says "Hadoop, 2019."



The screenshot shows the Hadoop web interface at localhost:9870/explorer.html#/weather_data. The page title is "Browse Directory". The URL in the address bar is /weather_data. The search bar contains "/weather_data". The table header includes columns: Permission, Owner, Group, Size, Last Modified, Replication, Block Size, and Name. There is one entry: "weather_cleaned.parquet" owned by root, group supergroup, size 192.83 KB, last modified Dec 13 22:58, replication 1, block size 128 MB. The footer says "Hadoop, 2019."

Phase 4: Dataset Merging

To create a unified dataset for analysis, it connects, in the *merge_datasets.py* Python script, to the silver layer on the MinIO server and read the cleaned version parquet files to apply inner join between them on the “date_time” and “city” columns.

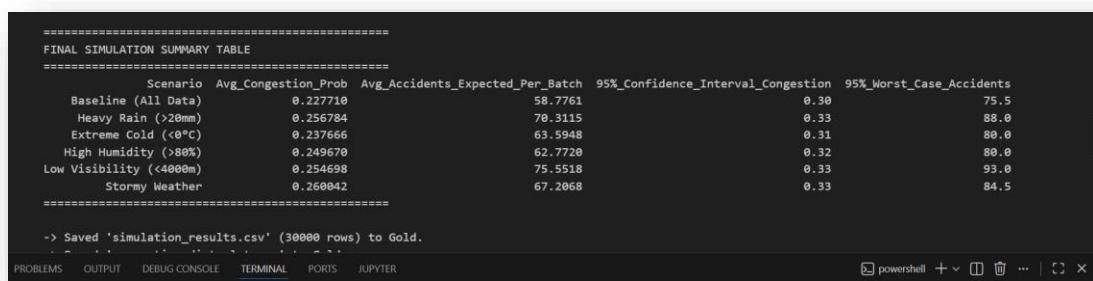
In the end, it uploads the cleaned unified one into the gold bucket on MinIO as a parquet file, as well.

Phase 5: Monte Carlo Simulation (Traffic Risk Prediction)

Simulates the traffic behavior under different weather conditions. It included:

- v) Heavy rain
- vi) Temperature extremes
- vii) High humidity
- viii) Low visibility
- ix) Strong winds

Its output contained the probabilities of each of “avg_congestion_probability”, “Avg_Accidents_Expected_Per_Batch”, “95%_Confidence_Interval_Congestion”, and “95%_Worst_Case_Accidents” for each of the scenarios above aside with the normal state.

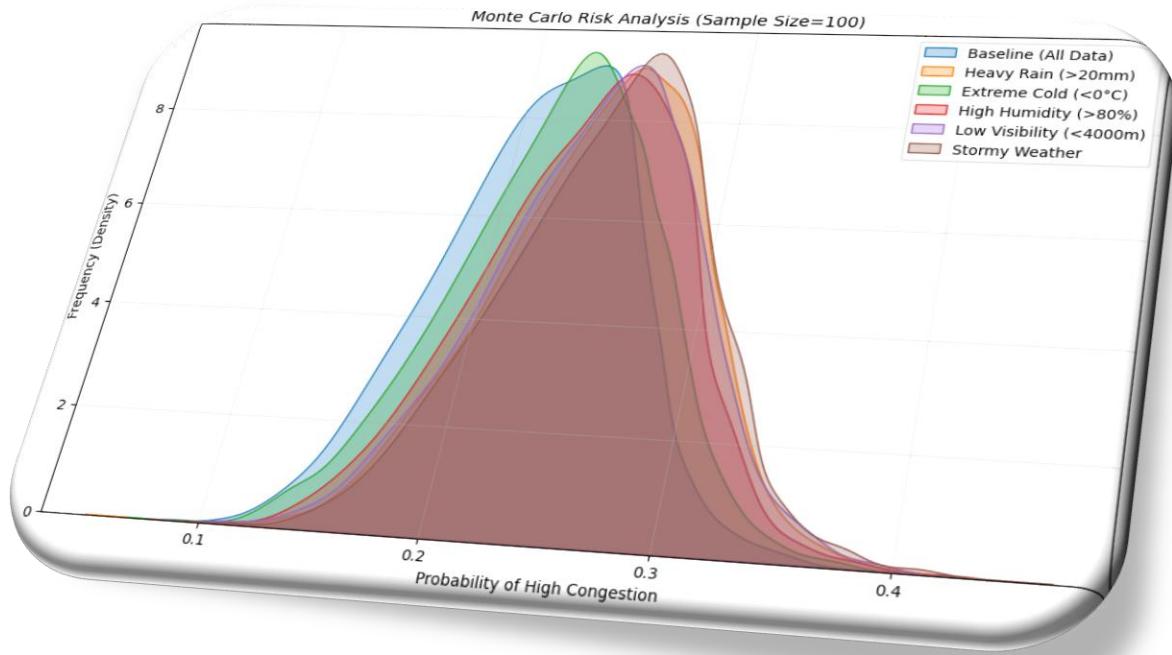


```
=====  
FINAL SIMULATION SUMMARY TABLE  
=====  


| Scenario                | Avg_Congestion_Prob | Avg_Accidents_Expected_Per_Batch | 95%_Confidence_Interval_Congestion | 95%_Worst_Case_Accidents |
|-------------------------|---------------------|----------------------------------|------------------------------------|--------------------------|
| Baseline (All Data)     | 0.227710            | 58.7761                          | 0.30                               | 75.5                     |
| Heavy Rain (>20mm)      | 0.256784            | 70.3115                          | 0.33                               | 88.0                     |
| Extreme Cold (<0°C)     | 0.237666            | 63.5948                          | 0.31                               | 80.0                     |
| High Humidity (>80%)    | 0.249670            | 62.7720                          | 0.32                               | 80.0                     |
| Low Visibility (<4000m) | 0.254698            | 75.5518                          | 0.33                               | 93.0                     |
| Stormy Weather          | 0.260042            | 67.2068                          | 0.33                               | 84.5                     |

  
-> Saved 'simulation_results.csv' (30000 rows) to Gold.  
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS JUPYTER powershell + × ⌂ ⌂ ⌂ ⌂ ⌂ ⌂ ⌂ ⌂ ⌂
```

Also, it uploads the dataset resulting in simulation into the gold layer on MinIO as well as the congestion probability distribution plot shown in the figure below.



Phase 6: Factor Analysis (Weather Impact Detection)

a) Checking factor analysis assumptions:

- i) No outliers in the data set → Done [in phase (2)]
- ii) Data set size (number of rows) > Number of factors → Done (as required number of factors is from 1 to 3 while data set size is 5250)
- iii) Checking whether there is perfect collinearity between features or not → Done [no perfect collinearity], checked in Factor Analysis.ipynb file **[and there wasn't perfect collinearity]**
- iv) Homoscedasticity between data set features:
 - first it wasn't satisfied, there was heteroscedasticity.

- After applying standardization, it is satisfied

b) Stating factor analysis test hypothesis:

- H_0 (Null Hypothesis): The variables in the dataset are uncorrelated = The correlation matrix is an identity matrix
- H_a (Alternative Hypothesis): The variables are correlated and suitable for factor analysis

c) Testing data set adequacy for factor analysis

- It passed Barlett's sphericity test
 - It failed KMO (Kaiser-Meyer-Olkin) test
- We should have stopped Factor analysis but in order to apply phase 7 (Visualization dashboard for factor analysis results & Monte Carlo results) we will continue

d) Determining suitable number of factors for the factor analysis

- Using Kaiser criterion approach (statistical approach)
→ Number of factors=3
- Using scree plot approach (graphical approach)
→ Number of factors=3

e) Factor analysis implementation

- Using PROMAX Rotation (Oblique rotation):

➤ Factor Loadings values table:

	a) FACTOR LOADINGS TABLE:		
	Factor1	Factor2	Factor3
temperature_c	0.0161	-0.2799	0.3639
humidity	-0.0401	0.3426	-0.0295
rain_mm	-0.0800	0.7766	0.3350
wind_speed_kmh	0.0097	0.0253	0.1127
visibility_weather	0.0207	-0.3122	-0.1181
air_pressure_hpa	-0.0057	-0.1112	-0.4556
vehicle_count	0.9365	-0.1354	0.0289
avg_speed_kmh	-0.5695	-0.0415	-0.0605
accident_count	0.0385	0.0809	-0.0209

- Communalities values table & average communality value:

b) COMMUNALITIES:		
Variable	Communality	
temperature_c	0.210976	
humidity	0.119881	
rain_mm	0.721686	
wind_speed_kmh	0.013422	
visibility_weather	0.111837	
air_pressure_hpa	0.219937	
vehicle_count	0.896248	
avg_speed_kmh	0.329754	
accident_count	0.008462	
Average communality: 0.2925		

- Factor (1):

Name: Traffic Density & Speed Factor

most 2 affecting features/variables: vehicle_count
and avg_speed_kmh

- Factor (2):

Name: Precipitation & Moisture Factor

most 2 affecting features/variables: humidity and rain_mm

- Factor (3):

Name: Temperature-Pressure Gradient

most 2 affecting features/variables: air_pressure_hpa
and temperature_c

ii) Using VARIMAX Rotation (Orthogonal rotation):

- Factor Loadings values table:

a) FACTOR LOADINGS TABLE:			
	Factor1	Factor2	Factor3
temperature_c	-0.0349	-0.2558	0.3407
humidity	0.0105	0.3313	-0.0013
rain_mm	0.0233	0.7725	0.3979
wind_speed_kmh	0.0100	0.0318	0.1142
visibility_weather	-0.0211	-0.3112	-0.1431
air_pressure_hpa	-0.0087	-0.1332	-0.4635
vehicle_count	0.9081	-0.0122	-0.0067
avg_speed_kmh	-0.5689	-0.1169	-0.0486
accident_count	0.0505	0.0837	-0.0155

- Communalities values table & average communality value:

b) COMMUNALITIES:	
<hr/>	
Variable	Communality
temperature_c	0.182709
humidity	0.109899
rain_mm	0.755677
wind_speed_kmh	0.014154
visibility_weather	0.117764
air_pressure_hpa	0.232602
vehicle_count	0.824858
avg_speed_kmh	0.339681
accident_count	0.009787
 Average communality: 0.2875	

- Factor (1)

Name: Traffic Density & Speed Factor

most 2 affecting features/variables: vehicle_count
and avg_speed_kmh

- Factor (2):

Name: Precipitation & Moisture Factor

most 2 affecting features/variables: humidity and rain_mm

- Factor (3):

Name: Precipitation-Pressure Factor

most 2 affecting features/variables: air_pressure_hpa
and rain_mm

- ii) Using QUARTIMAX Rotation (Orthogonal rotation):

➤ Factor Loadings
values table:

	Factor1	Factor2	Factor3
temperature_c	-0.0346	-0.0714	-0.4200
humidity	0.0094	0.2940	0.1529
rain_mm	0.0201	0.8691	0.0001
wind_speed_kmh	0.0097	0.0806	-0.0870
visibility_weather	-0.0199	-0.3422	-0.0153
air_pressure_hpa	-0.0075	-0.3307	0.3510
vehicle_count	0.9082	-0.0105	0.0005
avg_speed_kmh	-0.5684	-0.1283	-0.0104
accident_count	0.0502	0.0675	0.0521

➤ Communalities values table & average communality value:

b) COMMUNALITIES:	
Variable	Communality
temperature_c	0.182709
humidity	0.109899
rain_mm	0.755677
wind_speed_kmh	0.014154
visibility_weather	0.117764
air_pressure_hpa	0.232602
vehicle_count	0.824858
avg_speed_kmh	0.339681
accident_count	0.009787
Average communality: 0.2875	

➤ Factor (1)

Name: Traffic Density & Speed Factor most 2 affecting
features/variables: vehicle_count and avg_speed_kmh

➤ Factor (2):

Name: Low-Visibility Rain Factor most 2 affecting
features/variables: visibility_weather and rain_mm

➤ Factor (3):

Name: Temperature-Pressure Gradient Factor

most 2 affecting features/variables: air_pressure_hpa
and temperature_c

Phase 7 (Optional Bonus): Visualization Dashboard

Streamlit has been used to build two simple interactive web dashboards displaying:

- ✚ Cleaned dataset statistics
- ✚ Monte Carlo simulation results
- ✚ Factor Analysis insights

The screenshot shows a Streamlit dashboard titled "Big Data Project: Urban Traffic Risk Analysis". The title includes icons of a car and a cloud. Below the title, it says "A Predictive Data Lake Pipeline using MinIO, HDFS, and Monte Carlo Simulations". There are three tabs at the bottom: "Data Overview" (which is active), "Monte Carlo Risk", and "Factor Analysis". The main content area is titled "The Golden Dataset" and contains a table of data. The table has columns: traffic_id, date_time, city, area, vehicle_count, road_condition, avg_speed_kmh, congestion_level, accident_count, visibility_traffic, weather_. The table shows 10 rows of data. A dropdown menu labeled "Filter by City" is visible above the table, with "All" selected.

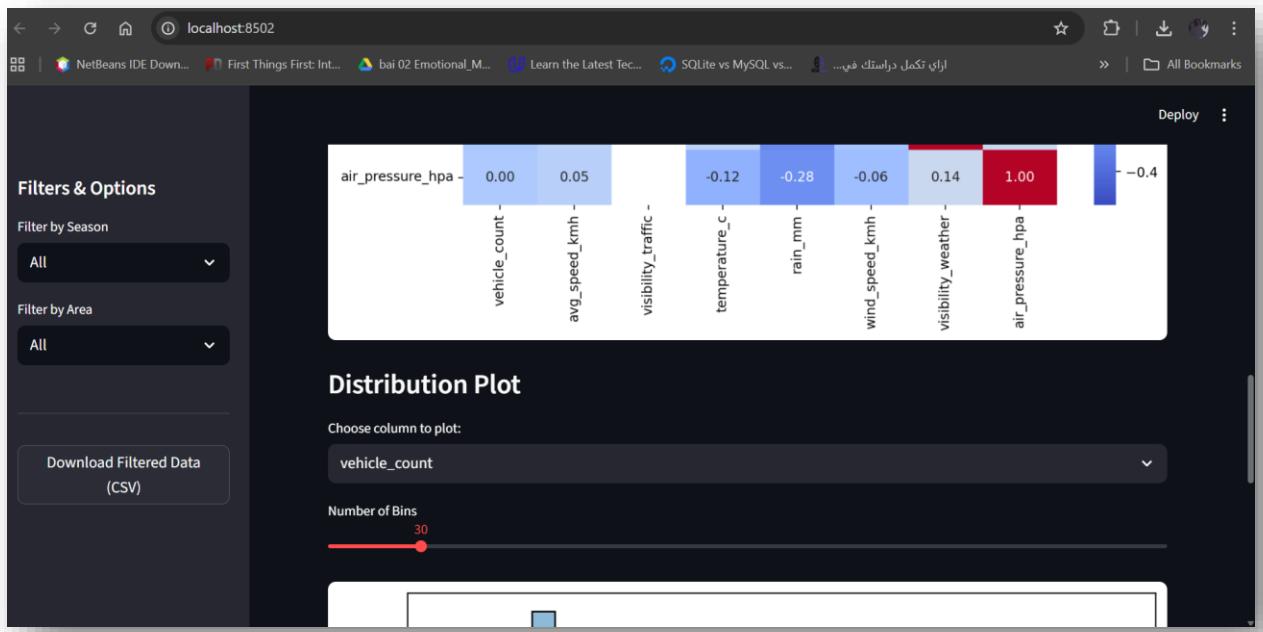
The screenshot shows a Streamlit dashboard titled "Data Overview". It displays a table of data with 10 rows. The columns are: index, traffic_id, date_time, city, area, avg_speed_kmh, road_condition, accident_count, visibility_traffic, and weather_. Below the table, it says "Showing top 1,000 rows. Total Data Points: 5250". Underneath the table, there are three summary statistics: "Total Records" (5250), "Avg Traffic Volume" (1941 cars/hr), and "Avg Temp" (13.1 °C). At the bottom, there is a message: "Pipeline Status: End-to-End Execution Complete." with a checkmark icon.

The two figures above show the data overview tab and the overview tab on the other dashboard allows for filtering in case of needing for more specificity about some feature.

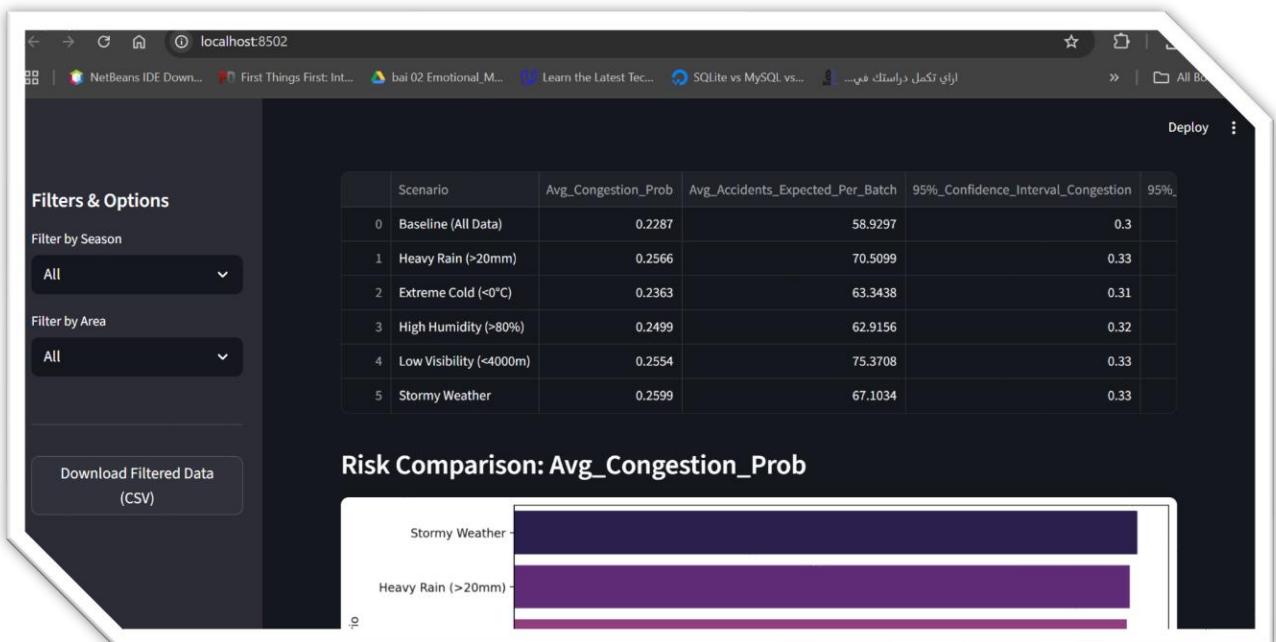
This screenshot shows the 'Dataset Overview & Quick Metrics' section of the dashboard. It displays three key statistics:

- Total Records: 5,250
- Total Vehicles: 10,195,356
- Total Accidents: 3,081

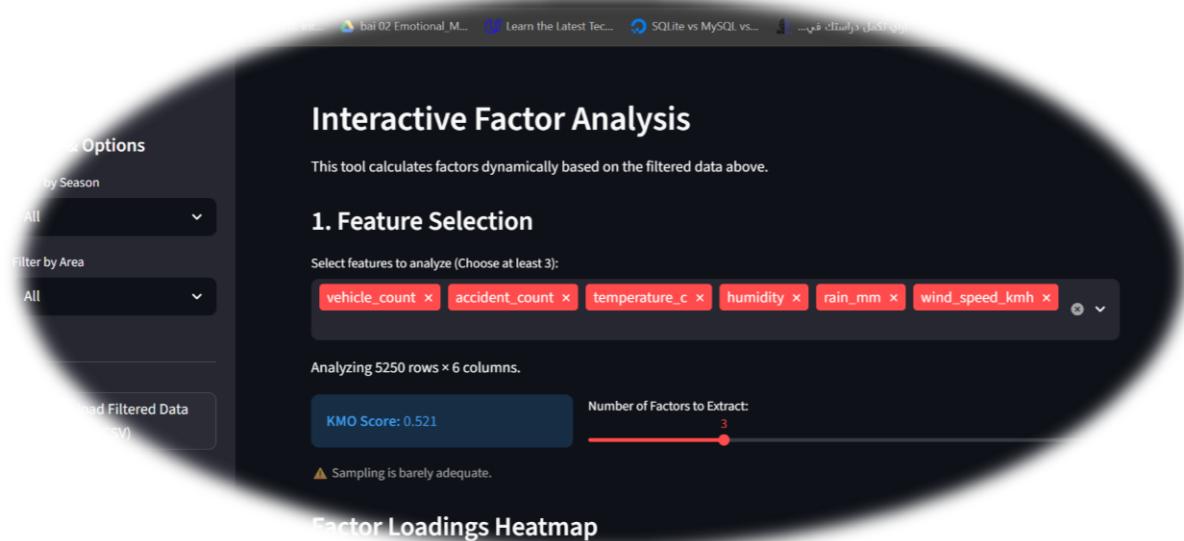
Below these metrics is a table titled 'Sample Data (Top 10 Rows)' with columns including traffic_id, date_time, city, area, vehicle_count, road_condition, avg_speed_kmh, congestion_level, and accident_severity. The first few rows of data are visible.



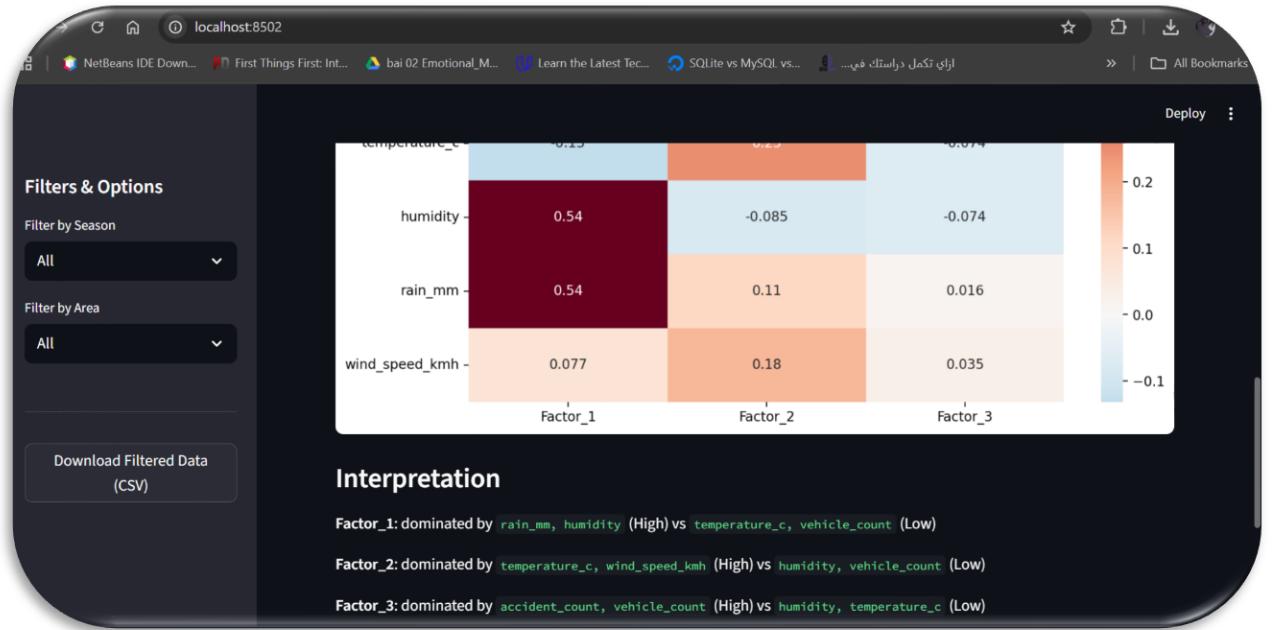
For the Monte Carlo tab, it shows the simulation summary table in addition to a plot for congestion probability in different scenarios.



For factor analysis, one dashboard enables users to apply it to some specific features and change the number of extracted factors.



In addition to showing the relationship between each feature and the factors using heatmap as shown below.



The other dashboard clarifies the eigen value table highlighting the highest three values aside with the scree plot that shows that 3 is the number of factors to extract.



Finally, it shows also each of factor loadings, communalities, and total variance explained tables enabling filtering based on the rotation method (Varimax – Promax – Quartimax)

localhost:8501

Select rotation method:

varimax

a) Factor Loadings (Varimax)

	Unnamed: 0	Factor1	Factor2	Factor3
0	temperature_c	-0.0349	-0.2558	0.3407
1	humidity	0.0105	0.3313	-0.0013
2	rain_mm	0.0233	0.7725	0.3979
3	wind_speed_kmh	0.01	0.0318	0.1142
4	visibility_weather	-0.0211	-0.3112	-0.1431
5	air_pressure_hpa	-0.0087	-0.1332	-0.4635
6	vehicle_count	0.9081	-0.0122	-0.0067
7	sun_rain_kmh	0.0500	0.1160	0.0040

b) Communalities (Varimax)

Variable	Community
0 temperature_c	0.1827
1 humidity	0.1099
2 rain_mm	0.7557
3 wind_speed_kmh	0.0142
4 visibility_weather	0.1178
5 air_pressure_hpa	0.2326
6 vehicle_count	0.8249
7 sun_rain_kmh	0.2207

c) Total Variance Explained (Varimax)

Component	Eigenvalue	Proportion	Cumulative Proportion
Factor1	0.3407	0.0681	0.0681
Factor2	0.1142	0.0229	0.0910
Factor3	0.0067	0.0013	0.0923
Factor4	0.0040	0.0008	0.0931

Thank You