

## # Uber Trip Analysis – Full Step-by-Step Machine Learning Workflow

### ## 1. Project Overview

This project analyzes Uber trip activity using the FOIL dataset and builds a machine learning model to predict hourly demand. The entire workflow—from loading the data to saving the model—is included.

### ## 2. Dataset Details

**\*\*File Path:\*\*** /content/Uber-Jan-Feb-FOIL.csv

Contains:

- date (timestamp of activity)
- dispatching\_base\_number
- active\_vehicles
- trips

### ## 3. Data Loading

The CSV is loaded directly from Google Colab using pandas. The datetime column is automatically detected and parsed.

### ## 4. Data Preprocessing

- Parsed datetime column with error handling
- Dropped invalid or missing timestamps
- Converted timestamps to hourly frequency using floor('H')
- Aggregated trip counts into hourly demand
- Filled missing hours with zeros to maintain continuity

### ## 5. Exploratory Data Analysis (EDA)

Performed initial analysis:

- Hourly pickup time-series plot
- Daily pickup totals
- Summary statistics (min, max, mean, distribution)

## ## 6. Feature Engineering

Created features for model input:

- hour
- dayofweek
- is\_weekend
- month
- day
- weekofyear
- lag\_1 (previous hour demand)
- rolling\_24\_mean (previous day's average demand)

## ## 7. Train-Test Split

Used a time-based split:

- First 70% of data → training
- Remaining 30% → testing

## ## 8. Model Building

Model used: **\*\*RandomForestRegressor\*\***

- Handles nonlinear relationships well
- Robust to missing or noisy data
- Fast to train

## ## 9. Model Evaluation

Computed:

- MAE (Mean Absolute Error)
- RMSE (Root Mean Squared Error)
- $R^2$  (Training & Testing)

Generated:

- Feature importance table
- Actual vs predicted plot for test set

## ## 10. Saving the Model

Model saved as:

- /content/uber\_hourly\_rf\_joblib.pkl

## ## 11. Next Recommended Steps

- Improve with XGBoost or LightGBM
- Add weather, events, and holiday data
- Cluster pickup locations for geospatial modeling
- Build dashboard using Power BI or Streamlit