1. Data Cleaning

- Addressed missing values by imputing them with the average of the respective columns.
- Identified and handled outliers to ensure data integrity.
- Verified and corrected data types to ensure consistency across the dataset.

2. Feature Selection

- Conducted multi-collinearity and correlation analysis among the features to identify and remove redundant variables.
- Evaluated feature importance using statistical methods, retaining those with significant contributions to the model.
- Checked for class imbalance in the target variable and applied Synthetic Minority Over-sampling Technique (SMOTE) to address this issue.

3. Feature Engineering

- Temporal Features: Created additional features such as the day of the week, hour of the day, and a 'busy hour' indicator to capture temporal patterns.
- Behavioral Features: Added driver-specific metrics like rolling completion rates and day/hour-specific performance averages.
- Distance Calculation: Calculated the Haversine distance between the driver and pickup locations and included it as a new feature.
- Geographical Clustering: Generated clustering features for both drivers and customers based on their locations to capture geographical influences.
- Additionally, engineered domain-specific features such as the average number of rides completed in the past five days, driver and customer clusters, and others based on domain knowledge.

4. Model Building

- Developed a model pipeline incorporating Logistic Regression, Random Forest, and XGBoost to compare performance.
- Random Forest emerged as the best model, demonstrating superior metrics, including F1 score, accuracy, precision, and recall compared to the other models.

5. Hyperparameter Tuning

- Implemented Grid Search to optimize hyperparameters for the Random Forest model.
- Identified the best parameters through this process, which were then used in the final model deployment.

6. Model Validation & Testing

- After obtaining the optimal parameters from Grid Search, these were applied in the final Random Forest model for deployment.

7. Model Evaluation

- Initially, the Random Forest model exhibited suboptimal performance with an accuracy and F1 score around 30%, alongside low precision and recall.
- After tuning and feature enhancements, the model's performance improved significantly: accuracy reached 96%, F1 score increased to 98%, and precision and recall rose to 97% and 98%, respectively.