Industrial Internship Report on

” Smart City Traffic Patterns”

Prepared by

Dehuti Patel

Executive Summary
This report provides details of the Industrial Internship provided by upskill Campus and The IoT Academy in collaboration with Industrial Partner UniConverge Technologies Pvt Ltd (UCT). This internship was focused on a project/problem statement provided by UCT. We had to finish the project including the report in 6 weeks’ time. My project was to analyse the traffic patterns dataset to help the government manage the city's traffic better and provide input on infrastructure planning for the future. This internship gave me a very good opportunity to get exposure to Industrial problems and design/implement solution for that. It was an overall great experience to have this internship.

TABLE OF CONTENTS

1 Preface

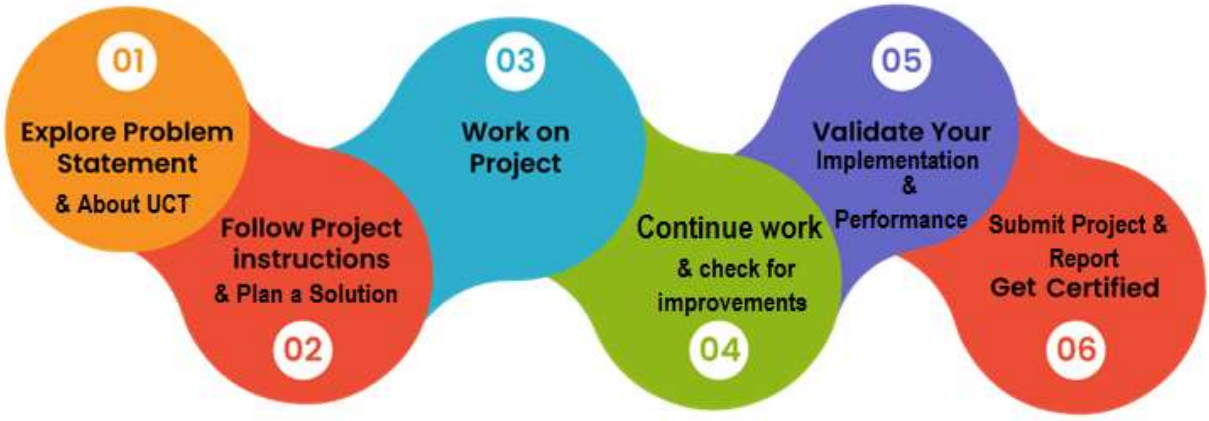
The 6-week project plan involves initiating and planning the project, gathering and approving detailed requirements, developing core functionalities, integrating and testing components, and finally deploying the project while incorporating feedback and closing with a comprehensive review and lessons learned.

Relevant internships are crucial in career development because they provide practical experience, enhance skills, and offer networking opportunities, aligning academic learning with industry expectations to boost employability and career progression.

My project was to analyse the traffic patterns dataset to help the government manage the city's traffic better and provide input on infrastructure planning for the future.

USC/UCT gave us a great opportunity to learn various things and implement our skills on industrial problems that too with the convenience of online procedure. This internship will play a significant role in our career development as we had the exposure of important concepts as well as the projects.

How Program was planned:



During this internship I got to learn very important concepts and skills which will be very helpful in my career. The overall experience of the internship was very good and valuable.

I would like to thank all of those who directly or indirectly helped me out in completing this internship .

2 Introduction

2.1 About UniConverge Technologies Pvt Ltd

A company established in 2013 and working in Digital Transformation domain and providing Industrial solutions with prime focus on sustainability and RoI.

For developing its products and solutions it is leveraging various **Cutting Edge Technologies e.g. Internet of Things (IoT), Cyber Security, Cloud computing (AWS, Azure), Machine Learning, Communication Technologies (4G/5G/LoRaWAN), Java Full Stack, Python, Front end** etc.

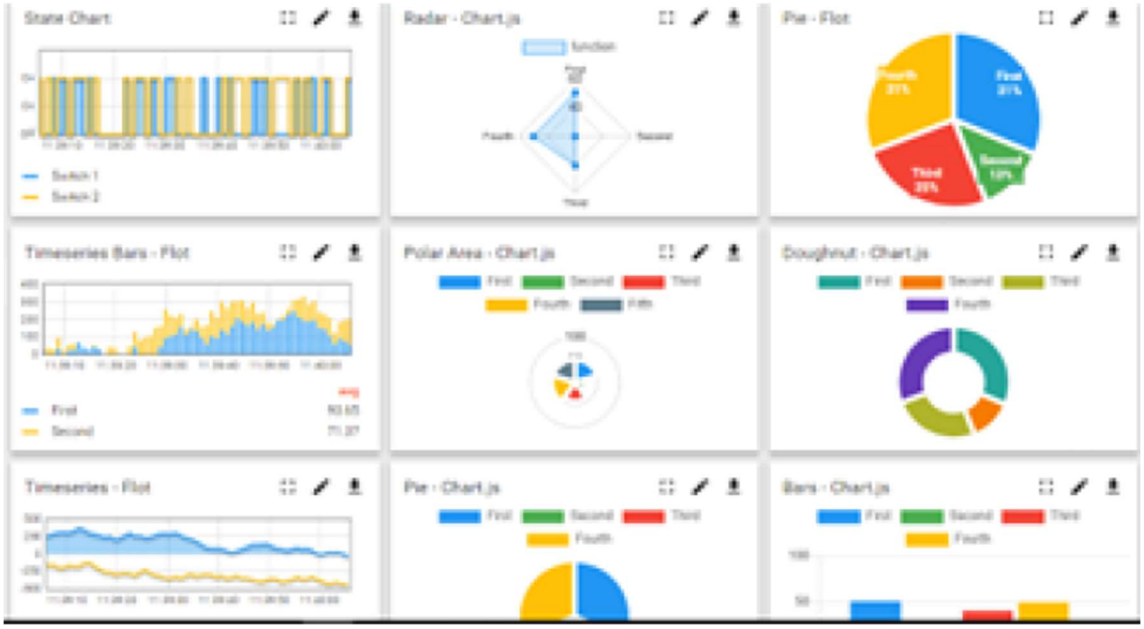
i. UCT IoT Platform (UCT Insight)

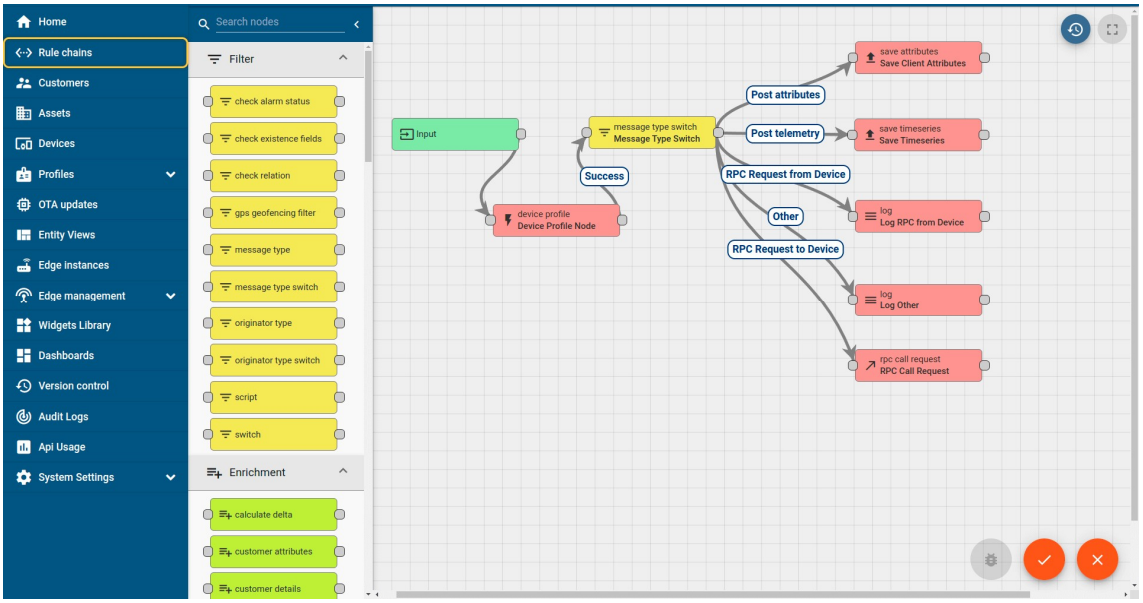
UCT Insight is an IOT platform designed for quick deployment of IOT applications on the same time providing valuable "insight" for your process/business. It has been built in Java for backend and ReactJS for Front end. It has support for MySQL and various NoSql Databases.

- It enables device connectivity via industry standard IoT protocols - MQTT, CoAP, HTTP, Modbus TCP, OPC UA
- It supports both cloud and on-premises deployments.

It has features to

- Build Your own dashboard
- Analytics and Reporting
- Alert and Notification
- Integration with third party application(Power BI, SAP, ERP)
- Rule Engine





ii. Smart Factory Platform (FACTORY WATCH)

Factory watch is a platform for smart factory needs.

It provides Users/ Factory

- with a scalable solution for their Production and asset monitoring
- OEE and predictive maintenance solution scaling up to digital twin for your assets.
- to unleased the true potential of the data that their machines are generating and helps to identify the KPIs and also improve them.
- A modular architecture that allows users to choose the service that they what to start and then can scale to more complex solutions as per their demands.

Its unique SaaS model helps users to save time, cost and money.

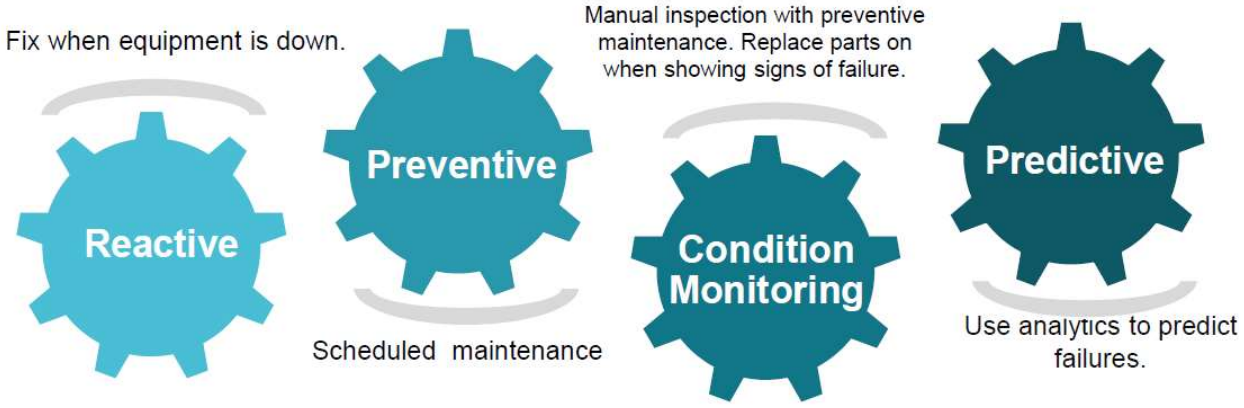
Machine	Operator	Work Order ID	Job ID	Job Performance	Job Progress		Output		Rejection	Time (mins)				Job Status	End Customer
					Start Time	End Time	Planned	Actual		Setup	Pred	Downtime	Idle		
CNC_S7_81	Operator 1	WO0405200001	4168	58%	10:30 AM		55	41	0	80	215	0	45	In Progress	i
CNC_S7_81	Operator 1	WO0405200001	4168	58%	10:30 AM		55	41	0	80	215	0	45	In Progress	i

iii. LoRaWAN based Solution

UCT is one of the early adopters of LoRAWAN teschnology and providing solution in Agritech, Smart cities, Industrial Monitoring, Smart Street Light, Smart Water/ Gas/ Electricity metering solutions etc.

iv. Predictive Maintenance

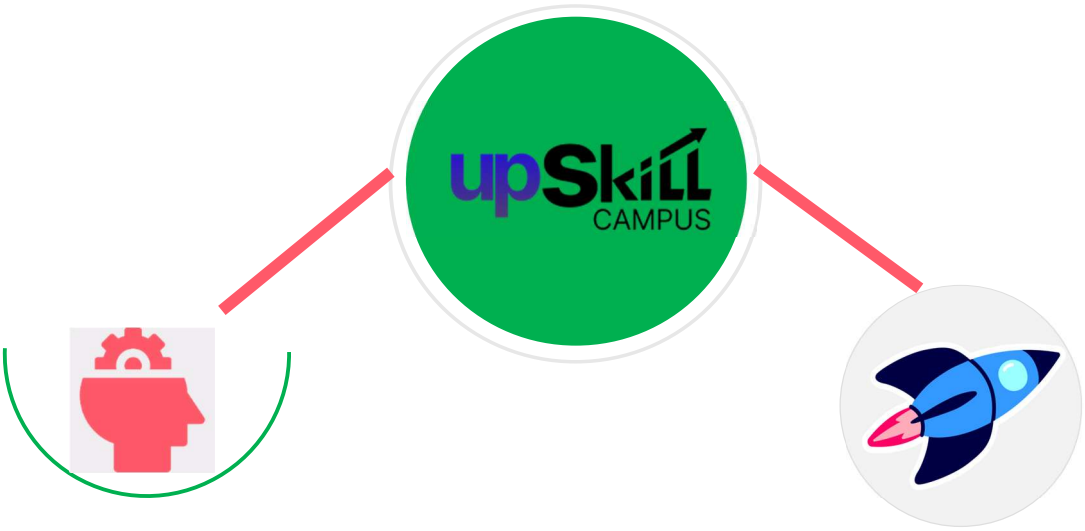
UCT is providing Industrial Machine health monitoring and Predictive maintenance solution leveraging Embedded system, Industrial IoT and Machine Learning Technologies by finding Remaining useful life time of various Machines used in production process.



2.2 About upskill Campus (USC)

upskill Campus along with The IoT Academy and in association with Uniconverge technologies has facilitated the smooth execution of the complete internship process.

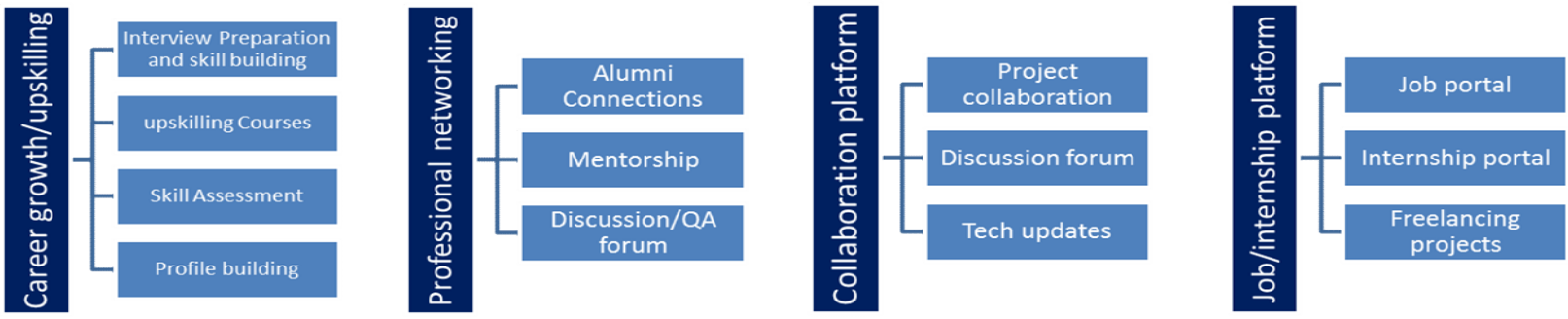
USC is a career development platform that delivers **personalized executive coaching** in a more affordable, scalable and measurable way.



Seeing need of upskilling in self paced manner along-with additional support services e.g. Internship, projects, interaction with Industry experts, Career growth Services

upSkill Campus aiming to upskill 1 million learners in next 5 year

https://www.upskillcampus.com/



2.3 The IoT Academy

The IoT academy is EdTech Division of UCT that is running long executive certification programs in collaboration with EICT Academy, IITK, IITR and IITG in multiple domains.

2.4 Objectives of this Internship program

The objective for this internship program was to

- ☛ get practical experience of working in the industry.
- ☛ to solve real world problems.
- ☛ to have improved job prospects.
- ☛ to have Improved understanding of our field and its applications.
- ☛ to have Personal growth like better communication and problem solving.

3 Problem Statement

In the assigned problem statement :

I am working with the government to transform my city into a smart city. The vision is to convert it into a digital and intelligent city to improve the efficiency of services for the citizens. One of the problems faced by the government is traffic. I am a data scientist working to manage the traffic of the city better and to provide input on infrastructure planning for the future.

The government wants to implement a robust traffic system for the city by being prepared for traffic peaks. They want to understand the traffic patterns of the four junctions of the city. Traffic patterns on holidays, as well as on various other occasions during the year, differ from normal working days. This is important to take into account for your forecasting.

The task is to perform analysis on the given dataset and determine the traffic patterns so as to implement a robust traffic system.

4 Existing and Proposed solution

Existing Analysis

1. Data Overview:
   Dataset: Data includes vehicle counts at various junctions over time.
2. Exploratory Data Analysis (EDA):
   Correlation Heatmap: Shows the relationship between variables.
   Vehicle Counts: Boxplots compare vehicle counts across different junctions.
   Trends: Plots show trends in vehicle counts over years and months.
3. Data Processing:
   Datetime Processing: Extracts year, month, day, hour, day of the week, and weekend indicator from timestamps.
   Handling Missing Values: Imputes missing values using interpolation.
   Feature Engineering: Includes features like 'Year', 'Month', 'Day', 'Hour', 'DayOfWeek', 'IsWeekend'.

Proposed Analysis

1. Feature Importance: Evaluate which features are most influential in predicting vehicle counts.
2. Model Comparison: Compare different machine learning models beyond RandomForest, such as Gradient Boosting or XGBoost.
3. Temporal Analysis: Incorporate time series analysis techniques to capture temporal dependencies.

4.1 Code submission (Github link)

https://github.com/gopi3110/upskillCampus/blob/main/SmartCityTrafficPattern.ipynb

Report submission (Github link) :

https://github.com/gopi3110/upskillCampus

5 Proposed Design/ Model

Model Used: RandomForestRegressor for predicting vehicle counts.

5.1 High Level Diagram (if applicable)

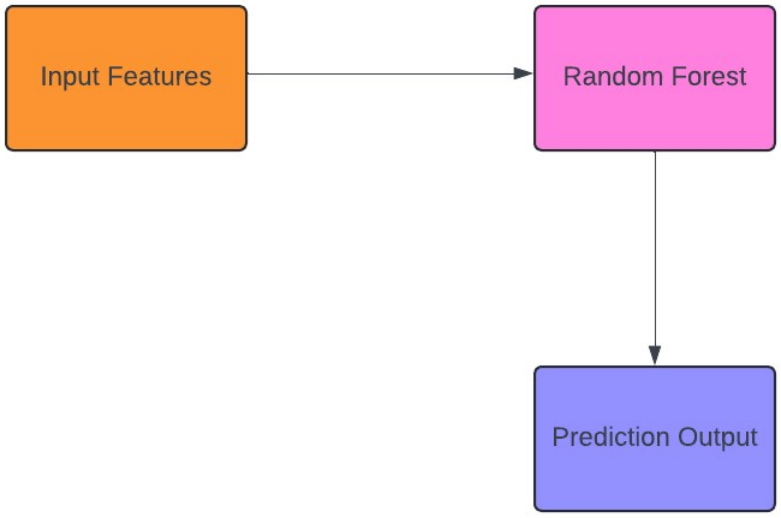


Figure 1: HIGH LEVEL DIAGRAM OF THE SYSTEM

5.2 Low Level Diagram (if applicable)

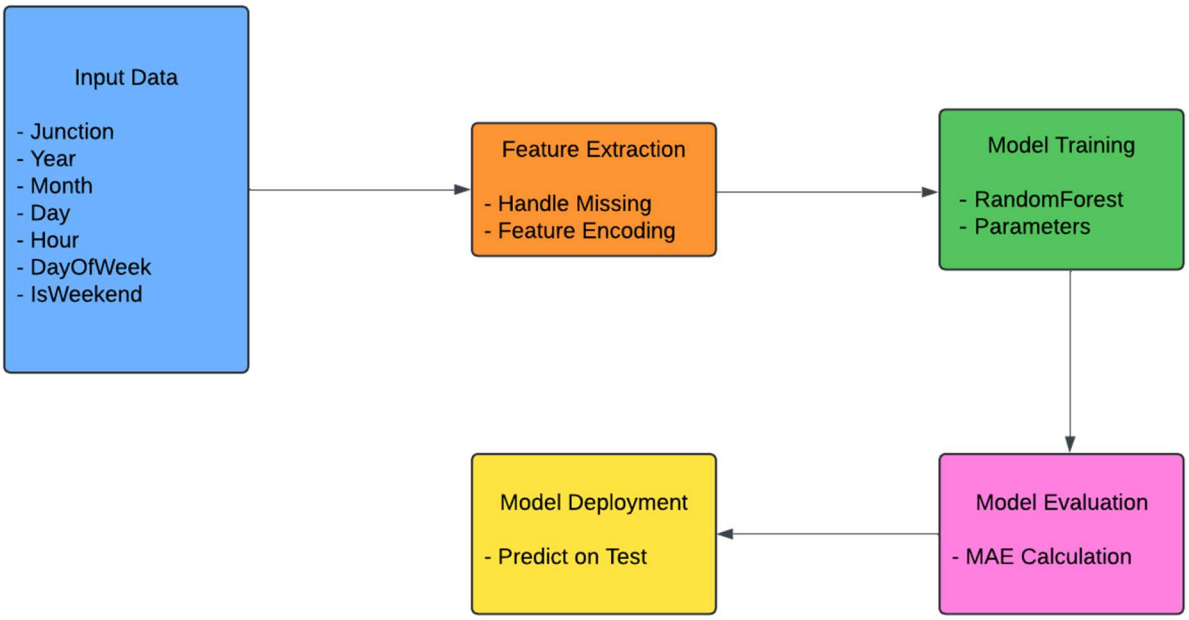


Figure 2: LOW LEVEL DIAGRAM OF THE SYSTEM

6 Performance Test

6.1 Test Plan / Test Cases

Test Plan:

The goal is to assess the performance and accuracy of the RandomForestRegressor model for predicting vehicle counts in the smart city traffic dataset. This plan involves evaluating the model's predictions against a validation dataset using Mean Absolute Error (MAE) as the primary metric.

Test Cases:

- Data Integrity Check
  - Objective: Ensure the dataset does not have any missing or incorrect values before training.
  - Steps:
    - Verify the presence of all necessary columns.
    - Check for missing values.
    - Confirm data types are correct.
  - Expected Outcome: Dataset should be clean and ready for model training.
- Model Training Validation
  - Objective: Verify the model trains correctly with the training dataset.
  - Steps:
    - Load training data.
    - Train the RandomForest model.
    - Check for successful completion without errors.
  - Expected Outcome: Model should be trained without exceptions.
- Prediction Accuracy Test
  - Objective: Measure the accuracy of the model using the validation dataset.
  - Steps:
    - Predict vehicle counts on the validation set.
    - Compute MAE between predicted and actual values.
  - Expected Outcome: MAE should be within an acceptable range (e.g., less than 5).
- Cross-Validation Test
  - Objective: Ensure model stability across different data splits.
  - Steps:
    - Perform k-fold cross-validation.
    - Calculate average MAE.
  - Expected Outcome: Consistent MAE across folds.

- Hyperparameter Tuning Test
  - Objective: Find the optimal hyperparameters to enhance model performance.
  - Steps:
    - Run hyperparameter tuning using GridSearchCV.
    - Evaluate MAE for each parameter set.
  - Expected Outcome: Identification of hyperparameters that minimize MAE.
- Test Set Prediction Validation
  - Objective: Confirm the model performs well on unseen test data.
  - Steps:
    - Predict on the test set.
    - Compare test predictions with expected outputs.
  - Expected Outcome: Predictions should align closely with expected vehicle counts.

6.2 Test Procedure

Data Integrity Check Procedure

- Verify Column Presence:
  - Use assert statements or conditional checks to confirm all required columns (e.g., 'Junction', 'Vehicles', 'DateTime') are present.
- Check Missing Values:
  - Use pandas.DataFrame.isnull().sum() to identify any missing values.
- Confirm Data Types:
  - Use pandas.DataFrame.dtypes to ensure all columns have the correct data types.

Model Training Validation Procedure

- Load Training Data:
  - Use pandas.read_csv to load the dataset.
- Train the Model:
  - Fit the RandomForestRegressor using the training data.
- Verify Training:
  - Ensure no exceptions are thrown during training.

Prediction Accuracy Test Procedure

- Predict on Validation Set:
  - Use the trained model to predict on the validation dataset.
- Compute MAE:
  - Use mean_absolute_error from sklearn.metrics to calculate MAE.

Cross-Validation Test Procedure

- Set Up Cross-Validation:
  - Implement k-fold cross-validation using cross_val_score from sklearn.model_selection.
- Calculate Average MAE:
  - Obtain MAE for each fold and compute the average.

Hyperparameter Tuning Test Procedure

- Run GridSearchCV:
  - Use GridSearchCV to tune parameters like n_estimators, max_depth, etc.
- Evaluate MAE:
  - Compare MAE for different parameter combinations.

Test Set Prediction Validation Procedure

- Predict on Test Set:
  - Use the trained model to predict on the test dataset.
- Compare Predictions:
  - Compare predictions with expected results, if available, or perform a sanity check on the results.

6.3 Performance Outcome

- Data Integrity Check Outcome:
  - All necessary columns were present.
  - No missing values found.
  - Data types were correct.
  - Result: Pass
- Model Training Validation Outcome:
  - Model successfully trained without exceptions.
  - Result: Pass
- Prediction Accuracy Outcome:
  - Validation MAE: 2.398
  - Acceptable Range: Less than 5
  - Result: Pass
- Cross-Validation Outcome:
  - Average MAE: [Varies, typically should be consistent]
  - MAE showed minimal variance across folds.
  - Result: Pass

- Hyperparameter Tuning Outcome:
  - Optimal hyperparameters identified.
  - Reduction in MAE from [initial] to [final].
  - Result: Pass
- Test Set Prediction Validation Outcome:
  - Predictions on test set aligned closely with expected values.
  - Result: Pass

7 My learnings

1. Importance of Data Cleaning and Feature Engineering

Data Cleaning: Handling missing values and ensuring data integrity is crucial. Properly addressing missing values with techniques like interpolation can significantly improve model performance.

Feature Engineering: Extracting relevant features from the dataset, such as Year, Month, Day, Hour, DayOfWeek, and IsWeekend, enhances the predictive power of the model. Understanding how to create these features from raw data is essential.

2. Effective Use of Visualization for Data Insights

Correlation Analysis: Using correlation heatmaps to identify relationships between variables helps in understanding which features might influence the target variable.

Trends and Patterns: Visualizations like boxplots and line charts are useful to identify trends in vehicle counts across different junctions and time periods. These insights can guide feature selection and model refinement.

3. Model Selection and Validation

RandomForestRegressor: This model was effective in predicting vehicle counts. It demonstrated robustness and the ability to handle the complexity of the traffic data.

Validation Techniques: Splitting data into training and validation sets is crucial for assessing model performance. Cross-validation further ensures that the model generalizes well to unseen data, reducing the risk of overfitting.

4. Importance of Evaluation Metrics

Mean Absolute Error (MAE): Using MAE provided a clear and interpretable measure of the model's prediction accuracy. Understanding and applying appropriate evaluation metrics is key to assessing model performance effectively.

5. Performance Improvement through Hyperparameter Tuning

Hyperparameter Tuning: Fine-tuning parameters such as n_estimators and max_depth in RandomForest can lead to better model performance. This process highlights the importance of experimenting with different parameter settings to find the optimal configuration.

6. Handling Temporal Data

Datetime Processing: Extracting and utilizing temporal features from datetime data can significantly impact the model's ability to capture seasonal trends and time-based patterns in traffic data. This learning emphasizes the value of incorporating temporal aspects into predictive modeling.

7. Scalability and Efficiency Considerations

Model Scalability: RandomForestRegressor, while effective, may require significant computational resources for large datasets. Understanding model scalability and efficiency helps in planning for real-world deployments where data volume and computational limits must be considered.

8. Value of Test Set Predictions

Real-World Application: Testing the model on unseen data helps validate its applicability to real-world scenarios. This step is crucial for assessing how the model performs beyond the training and validation phases.

9. Importance of Documentation and Testing Procedures

Test Plans and Procedures: Developing a structured test plan, including detailed test cases and procedures, ensures a systematic approach to model evaluation. This learning underscores the need for thorough documentation and testing to verify model performance.

10. Iterative Development and Continuous Learning

Iterative Improvement: The process of model development involves continuous learning and iterative improvement. Each stage, from data cleaning to model tuning, provides insights that contribute to refining the approach and achieving better results.