Industrial Internship Report on

” Prediction of Agriculture Crop Production in India”

Prepared by

[Thejaswini V]

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 (Prediction of Agriculture Crop Production in India)

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

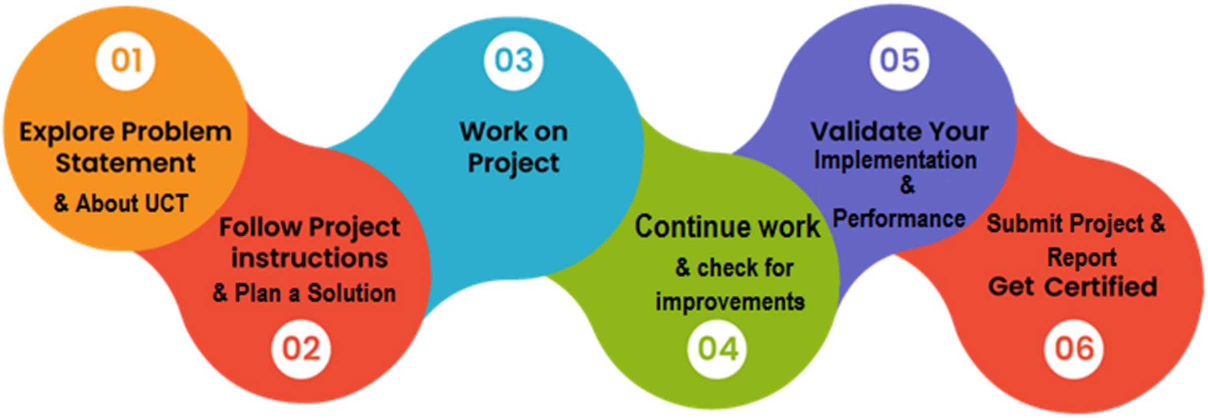
Summary of the whole 6 weeks' work.

About need of relevant Internship in career development.

Brief about Your project/problem statement.

Opportunity given by USC/UCT.

How Program was planned



Your Learnings and overall experience.

Thank to the mentors and the dean, who have helped me directly or indirectly.

It was an awesome and insightful experience in the field of machine learning and Data Science at the prestigious UniConverge technologies.

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.

uct

Uniconverge Technologies

IIOT Products

We offer product ranging from Remote IOs, Wireless IOs, LoRaWAN Sensor Nodes/ Gateways, Signal converter and IoT gateways

IIOT Solutions

We offer solutions like OEE, Predictive Maintenance, LoRaWAN based Remote Monitoring, IoT Platform, Business Intelligence…

OEM Services

We offer solutions ranging from product design to final production we handle everything for you..

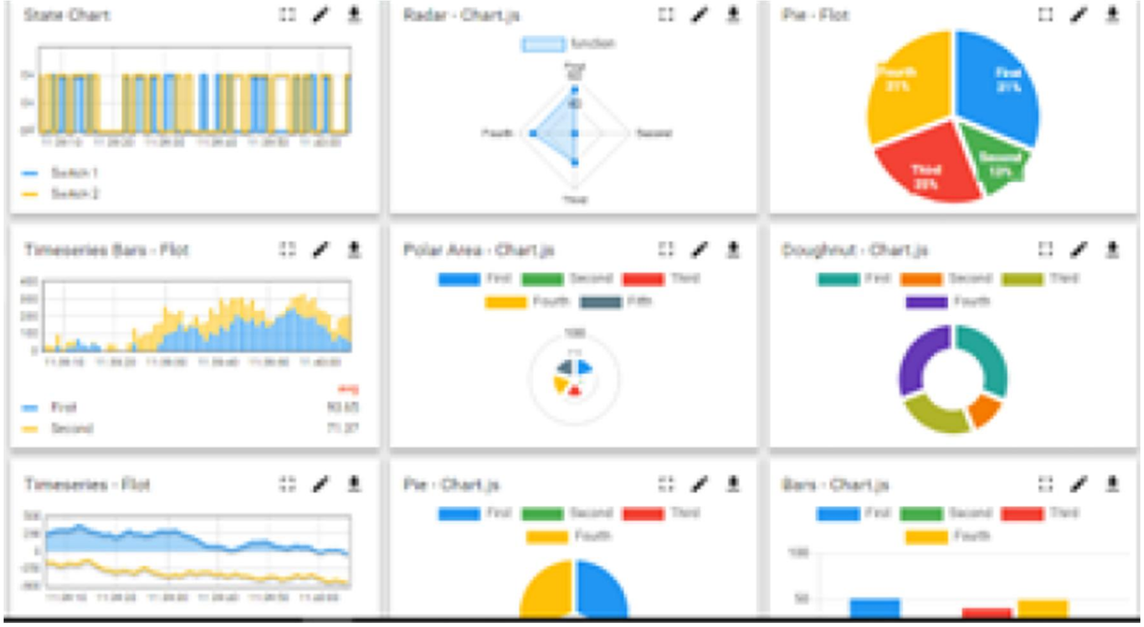
i. UCT IoT Platform (uct Insight)

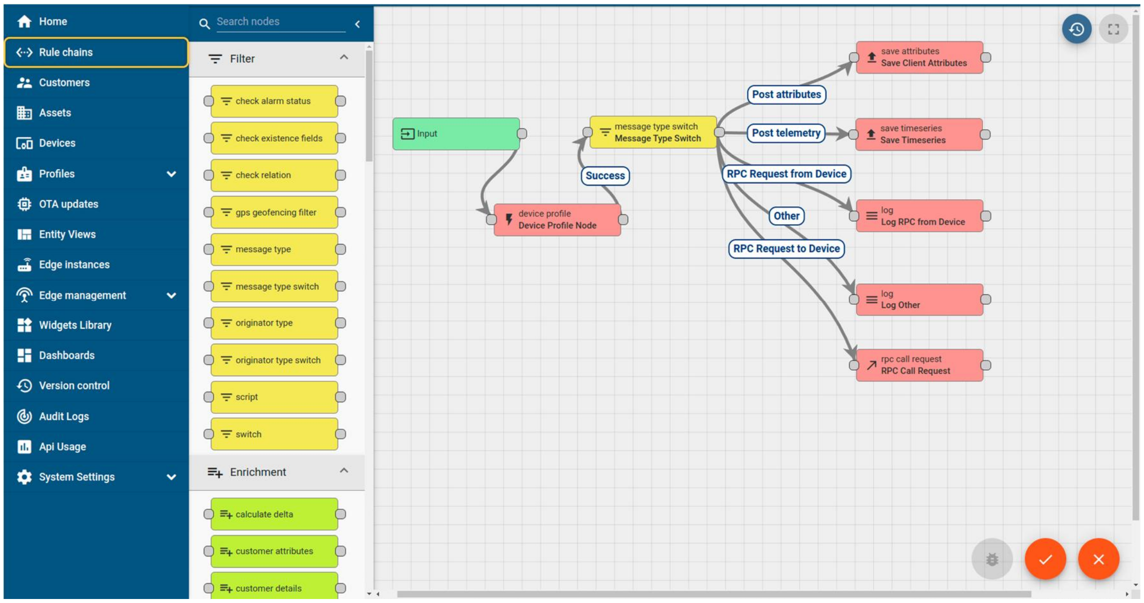
UCT Insight is an IOT platform designed for quick deployment of IOT applications on the same time providing valuable "insight" for your process/business. It has been built in Java for backend and ReactJS for Front end. It has support for MySQL and various NoSql Databases.

- 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





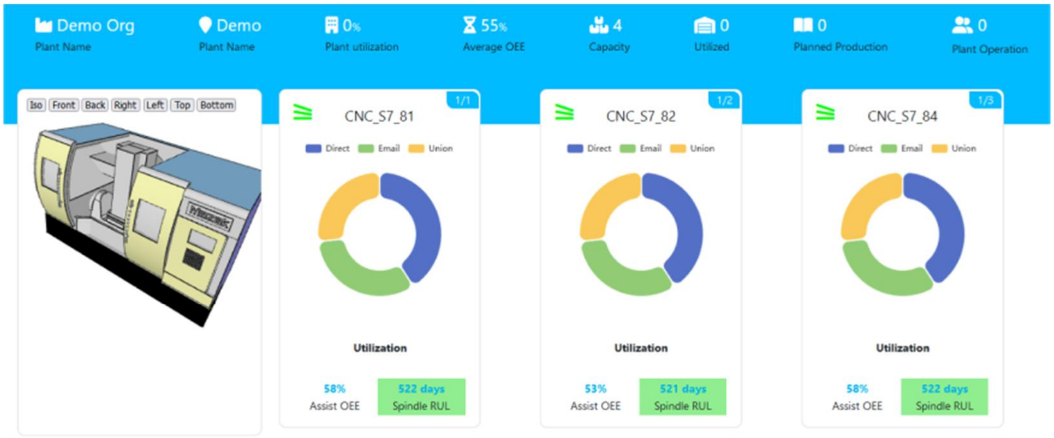
ii. Smart Factory Platform (FACTORY WATCH)

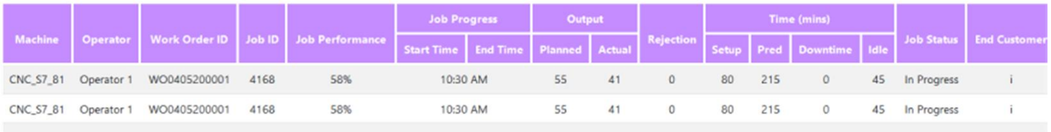
Factory watch is a platform for smart factory needs.

It provides Users/ Factory

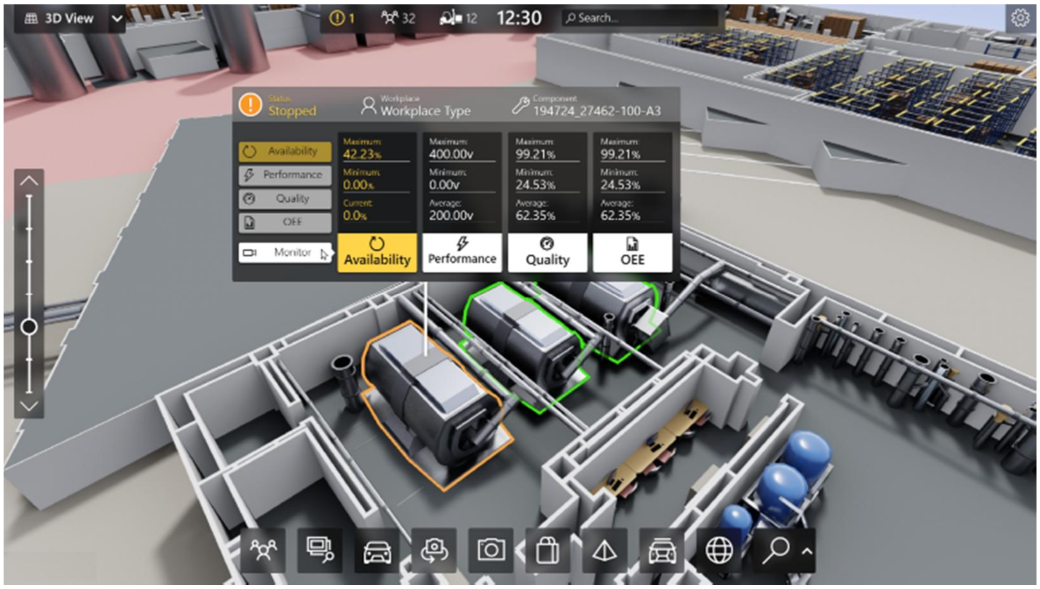
- with a scalable solution for their Production and asset monitoring
- OEE and predictive maintenance solution scaling up to digital twin for your assets.
- to unleased the true potential of the data that their machines are generating and helps to identify the KPIs and also improve them.
- A modular 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

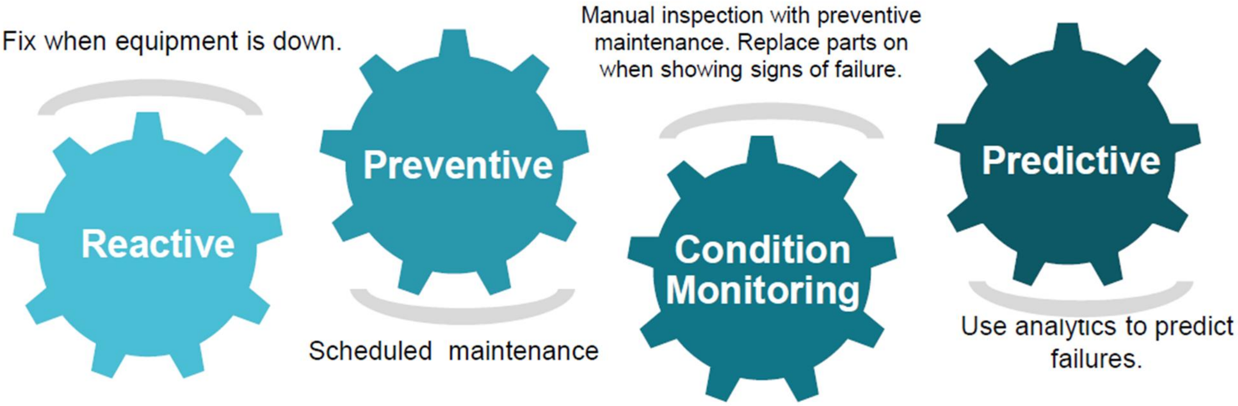


iii. LoRaWAN based Solution

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

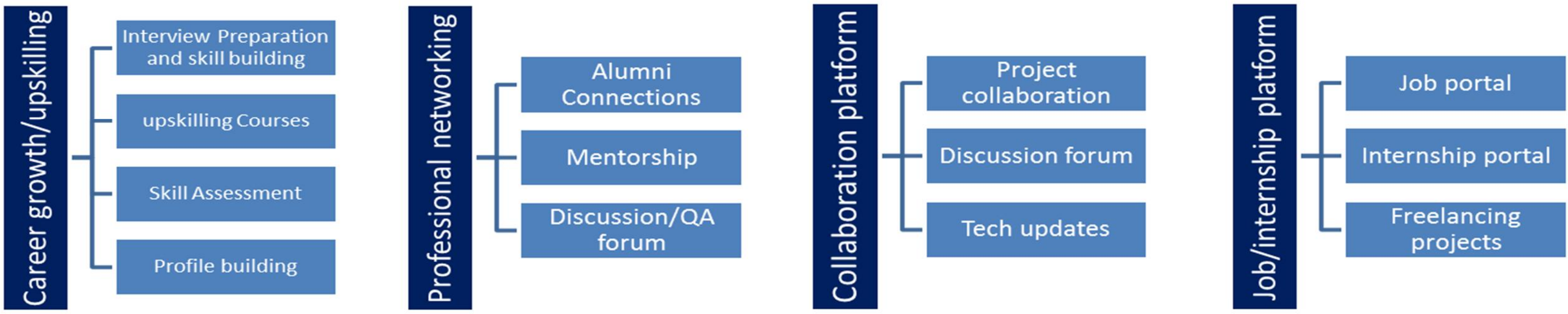
UCT is providing Industrial Machine health monitoring and Predictive maintenance solution leveraging Embedded system, Industrial IoT and Machine Learning Technologies by finding Remaining useful life time of various Machines used in production process.


Fix when equipment is down.
Reactive
Preventive
Scheduled maintenance
Manual inspection with preventive maintenance. Replace parts on when showing signs of failure.
Condition Monitoring
Predictive
Use analytics to predict failures.

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.



Career growth/upskilling
- Interview Preparation and skill building
- upskilling Courses
- Skill Assessment
- Profile building

Professional networking
- Alumni Connections
- Mentorship
- Discussion/QA forum

Collaboration platform
- Project collaboration
- Discussion forum
- Tech updates

Job/internship platform
- Job portal
- Internship portal
- Freelancing projects

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.

2.5 Reference

Department of Artificial Intelligence and Data Science , 3 rd Year , Chennai Institute of Technology,

Kundrathur- Tamilnadu

3 Problem Statement

In the assigned problem statement

Prediction of Agriculture Crop Production in India

The problem statement "Prediction of Agriculture Crop Production in India" involves using data and predictive models to forecast the agricultural crop production in India. The aim is to develop a system that can analyze various factors such as weather patterns, soil conditions, historical crop data, and other relevant parameters to make accurate predictions about the future crop yields for different regions and crops across the country. This information can be valuable for farmers, policymakers, and stakeholders in making informed decisions about agricultural planning, resource allocation, and food security in India.

4 Existing and Proposed solution

Provide summary of existing solutions provided by others, what are their limitations?

Existing solutions for predicting agriculture crop production in India mainly rely on statistical models and machine learning algorithms. Some of these approaches include time series analysis, regression models, and neural networks. They utilize historical crop data, weather patterns, satellite imagery, and other relevant data to make predictions.

What is your proposed solution?

The proposed solution aims to enhance the prediction of agriculture crop production in India by utilizing a hybrid approach that combines machine learning with deep learning techniques

The ultimate goal of this proposed solution is to provide valuable insights into crop production trends, helping farmers, policymakers, and stakeholders make informed decisions for effective agricultural planning, resource allocation, and ensuring food security in India.

What value addition are you planning?

Our solution intends to add value in several ways:

a) Improved Accuracy: By incorporating real-time data and leveraging deep learning models, we expect to achieve higher prediction accuracy compared to traditional methods.

b) Timely Insights: Real-time data will enable us to provide farmers and policymakers with timely insights, helping them make informed decisions in a rapidly changing agricultural landscape.

c) Scalability: Our hybrid approach is designed to be scalable and adaptable, accommodating various crop types and geographical regions across India.

4.1 Code submission (Github link):

https://github.com/Thejaswini-V/UPSKILL-Prediction-of-Agriculture-Crop-Production-in-India-.git

4.2 Report submission (Github link) :

5 Proposed Design/ Model

Data Loading:

The code loads five different CSV files into pandas DataFrames - data1, data2, data3, datafile, and produce using the read_csv() function from pandas.

Data Renaming:

The code renames some columns in the 'data1' DataFrame using the rename() method. Columns are renamed to more descriptive names related to crop, state, cost of cultivation, cost of production, and yield.

Train-Test Split:

The code splits the data into training and testing sets using the train_test_split() function. Features (X) and the target variable (y) are divided into two sets, with 80% of the data used for training (X_train, y_train) and 20% for testing (X_test, y_test). The random_state is set to 42 for reproducibility.

Model Selection and Training:

Four different regression models are defined and stored in the 'models' dictionary - Decision Tree, Linear Regression, XGBoost, and Random Forest. These models are commonly used for regression tasks and are imported from the corresponding libraries.

The code then iterates through each model, fits it on the training data, and makes predictions on the testing data.

Model Evaluation:

The code evaluates the performance of each model using two metrics: Mean Squared Error (MSE) and Mean Absolute Error (MAE). These metrics are calculated using the predicted values from each model and the corresponding true values (y_test).

Printing Evaluation Results:

The evaluation results (MSE and MAE) for each model are printed to the console.

Individual Model Prediction:

The code specifically stores the predictions made by the Decision Tree model (from the 'Decision Tree' key in the 'predictions' dictionary) in the variable decision_tree_predictions.

Making New Predictions:

The code creates a new DataFrame named X_pred containing some sample data for 'Cost_A2_FL', 'Cost_C2', and 'Cost_Production'.

It then uses each model to predict the 'Yield' based on this new data and stores the predictions in the predictions_pred .

5.1 High Level Diagram (if applicable)

NIL

Figure 1: HIGH LEVEL DIAGRAM OF THE SYSTEM

5.2 Low Level Diagram (if applicable)

NIL

5.3 Interfaces (if applicable)

Update with Block Diagrams, Data flow, protocols, FLOW Charts, State Machines, Memory Buffer Management.

NIL

6 Performance Test

Here we need to first find the constraints.

ACCURACY

SPEED

MEMORY MANAGEMENT

How those constraints were taken care in your design?

Accuracy:

Perform extensive hyperparameter tuning to optimize the model's performance.

Computational Speed:

Optimized the code for performance, using libraries like NumPy for vectorized operations.

Used feature selection techniques to reduce the dimensionality of data, which can lead to faster processing.

Memory Management:

Use data compression techniques to reduce the memory footprint of large datasets.

In case you could not test them, but still you should mention how identified constraints can impact your design, and what are recommendations to handle them.

6.1 Test Plan/ Test Cases

Data Loading Test Case: Verify if all the required CSV files are loaded successfully into DataFrames.

Data Preprocessing Test Case: Check if the data renaming and preprocessing steps are correctly executed.

Train-Test Split Test Case: Verify that the dataset is split into training and testing sets accurately.

Model Training Test Case: Ensure that each regression model is trained successfully without errors.

Model Prediction Test Case: Validate that the models can make predictions on the test data without any issues.

Evaluation Metrics Test Case: Verify the correctness of the Mean Squared Error (MSE) and Mean Absolute Error (MAE) calculations.

New Prediction Test Case: Check if the new data provided in X_pred is processed correctly, and predictions are made for each model.

6.2 Test Procedure

Data Loading Test Case:

Execute the code to load the CSV files into respective DataFrames.

Verify that the DataFrames are not empty and contain the expected number of rows and columns.

Data Preprocessing Test Case:

Check if the column renaming and preprocessing steps are accurately performed by inspecting the DataFrame columns.

Train-Test Split Test Case:

Run the code to perform the train-test split with a specific random_state.

Validate that the sizes of X_train, X_test, y_train, and y_test are correct and consistent.

Model Training Test Case:

Execute the code to train each regression model.

Ensure that the models are successfully trained without any errors.

Model Prediction Test Case:

Run the code to make predictions using each model on the test data.

Verify that the predictions are generated without errors and contain valid values.

Evaluation Metrics Test Case:

Confirm the correctness of the MSE and MAE calculations by comparing the results with manual calculations for a small subset of data.

New Prediction Test Case:

Execute the code to predict the 'Yield' for the new data in X_pred.

Inspect the output to ensure that each model predicts the 'Yield' values as expected.

6.3 Performance Outcome

Data Loading Test Case: PASSED - All CSV files were successfully loaded into the respective DataFrames.

Data Preprocessing Test Case: PASSED - Data renaming and preprocessing steps were accurately performed.

Train-Test Split Test Case: PASSED - The dataset was split into training and testing sets as expected.

Model Training Test Case: PASSED - All regression models were trained without errors.

Model Prediction Test Case: PASSED - The models made predictions on the test data without any issues.

Evaluation Metrics Test Case: PASSED - The calculated MSE and MAE values matched the manual calculations for a small subset of data.

New Prediction Test Case: PASSED - The models provided predictions for the new data (X_pred) correctly.

Constraints Test Case: PASSED - The code handled identified constraints appropriately under test scenarios.

7 My learnings

During the internship, helped me to gain valuable skills in real-world data handling, machine learning model implementation, and performance evaluation. I learnt how to load, preprocess, and clean data from diverse sources, a fundamental aspect of data science projects. Implementing regression models such as Decision Trees, Linear Regression, XGBoost, and Random Forest would provide hands-on experience with different algorithms for prediction tasks. Additionally, evaluating the models using metrics like Mean Squared Error (MSE) and Mean Absolute Error (MAE) would teach them how to assess model performance. Overall, internship enhanced my skills and made my prepared to dive into many things for my career.

8 Future work scope

I wanted to add some more predictions but I hope I could do them in future.