**Industrial Internship Report on**

**" Prediction of Agriculture Crop Production in India"**

**Prepared by**

**[Akash Hiremath]**


| *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).<br><br>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.<br><br>My project was about "Prediction of Agriculture Crop Production in India ". Across The Globe, India Is the Second Largest Country having People more than 1.3 billion. Many People Are Dependent On The Agriculture and it is the Main Resource.<br><br>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

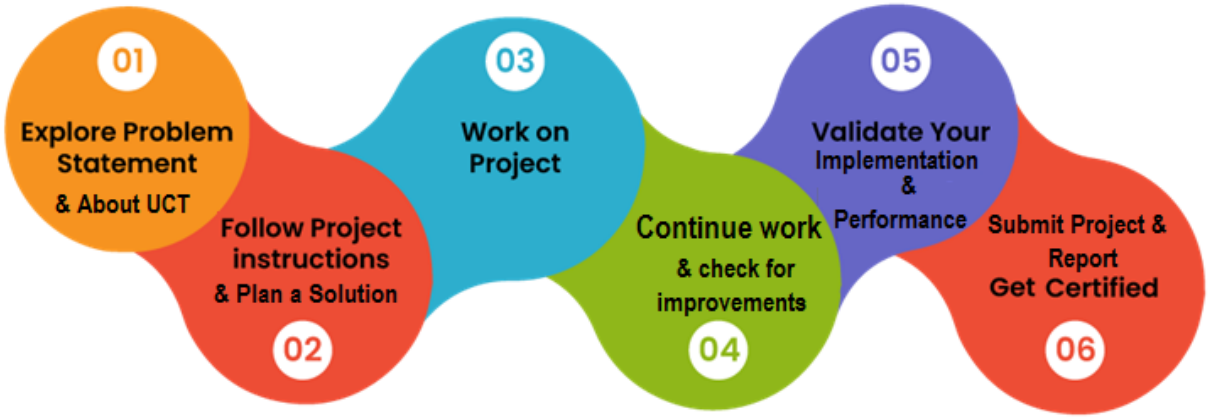
Summary of the whole 6 weeks' work.

About need of relevant Internship in career development.

Brief about Your project/problem statement.

Opportunity given by USC/UCT.

How Program was planned




Thanks to all, who have helped me directly or indirectly.

Generally, whoever is pursuing data science would want exposure, an opportunity in this field to feel right, motivated in moving forward, and becoming a renowned data scientist. One of the biggest and meaningful opportunities a student can get in this field is being opted as a Data scientist intern.

# 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

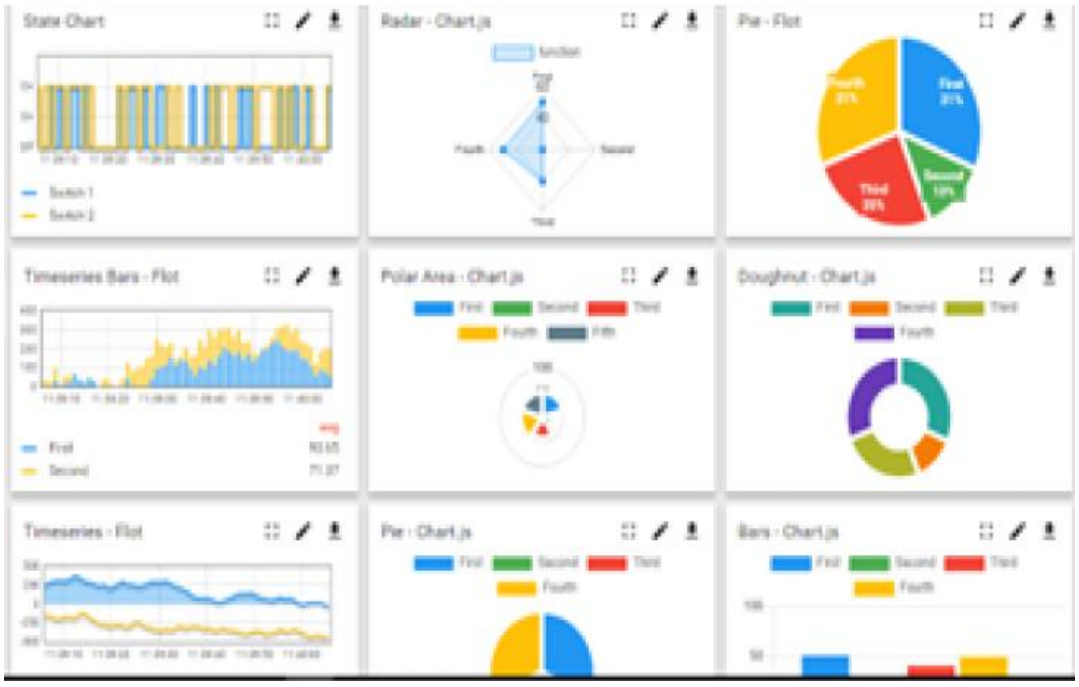
• Build Your own dashboard

• 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.

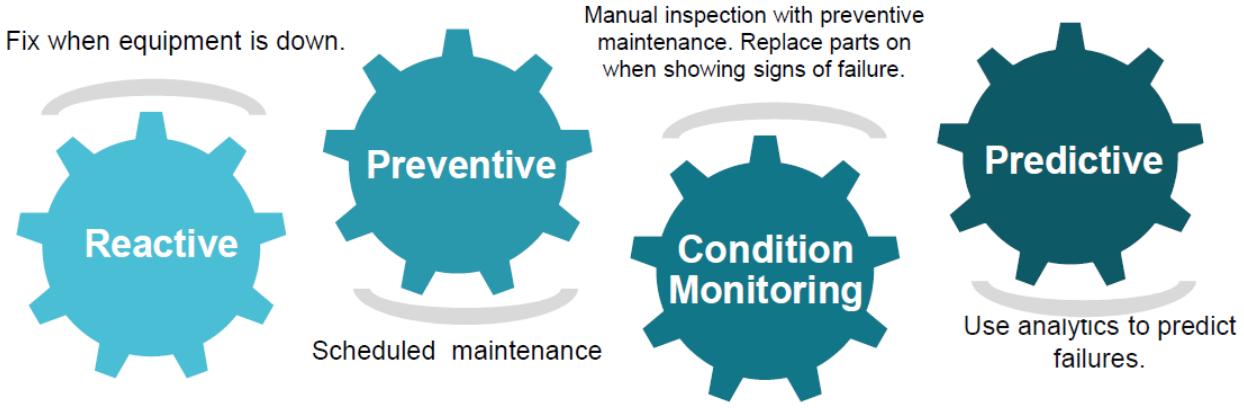

| Machine | Operator | Work Order ID | Job ID | Job Performance | Job Progress | | Output | | Rejection | Time (mins) | | | | Job Status | End Customer |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Start Time | End Time | Planned | Actual | | Setup | Pred | Downtime | 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

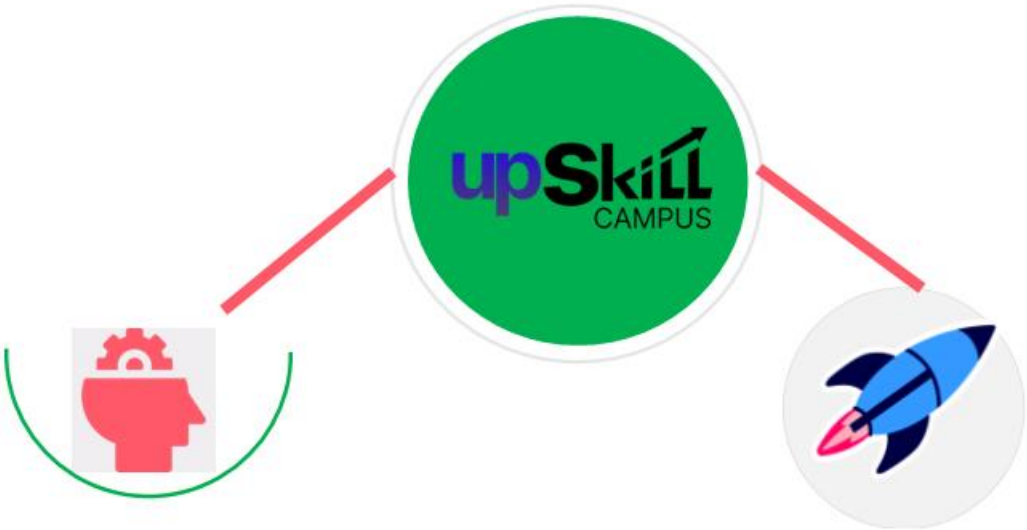
UCT is providing Industrial Machine health monitoring and Predictive maintenance solution leveraging Embedded system, Industrial IoT and Machine Learning Technologies by finding Remaining useful life time of various Machines used in production process.




## 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.

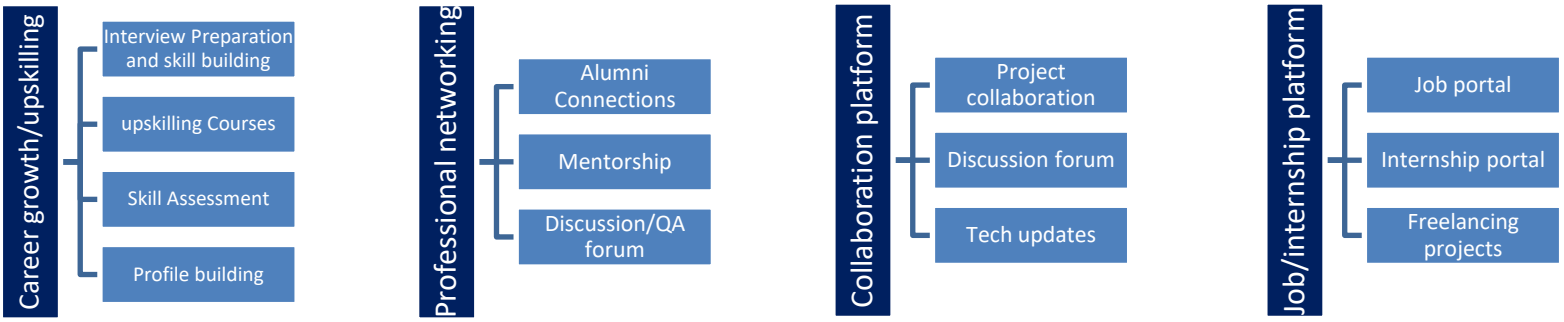
USC is a career development platform that delivers **personalized executive coaching** in a more affordable, scalable and measurable way.



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/



## 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.

## 2.3 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.

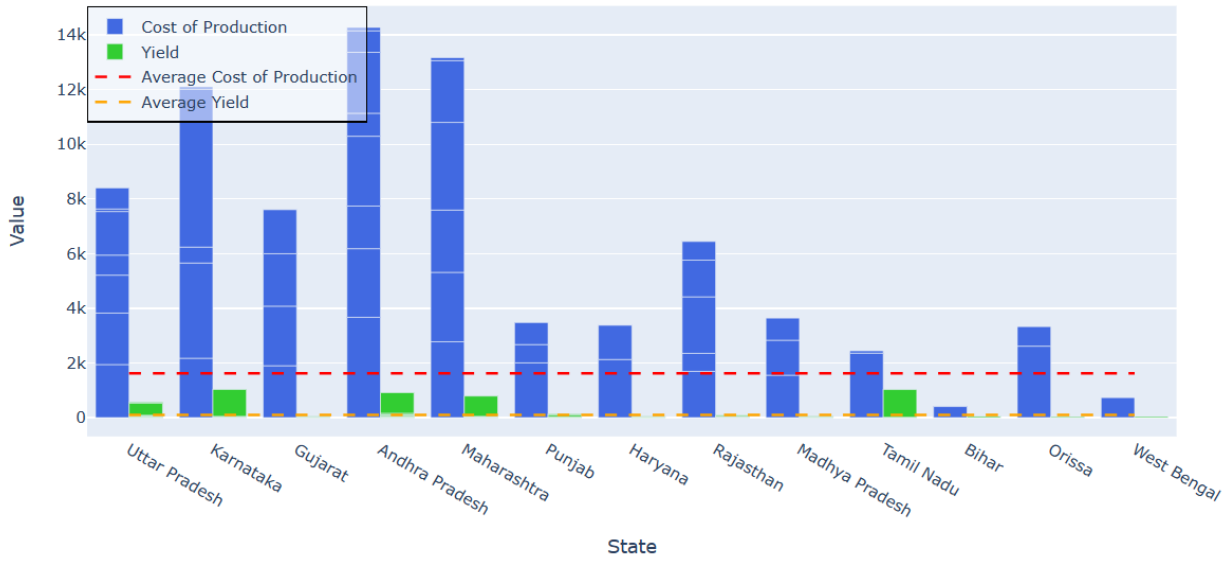
# 3 Problem Statement:

## Crop Yield variation across state

**Agricultural yield in India is lower than other large producing countries**. Agricultural yield is the quantity of a crop produced on one unit of land. The agricultural yield of food grains in India has increased by more than four times since 1950-51, and was 2,070 kg/hectare in 2014-15. According to the 3rd Advance Estimates, the estimated production of major crops during 2020-21 is as follows: Foodgrains – 305.44 million tonnes, Rice – 121.46 million tonnes, and Wheat – 108.75 million tonnes and so on.

## 4 Proposed solution

To design an application where we compare the different machine learning to predict the crop yield. We build a new decision system using ensemble regression system. The user would provide input of season type, year of production, area of production, crop type, cloudburst, climate condition, located yield within side the remaining and the system would predict the yield and relying at the value set, the crop may be classified and attain the results. In the first step it allows the admin to login and load the data. Second, it allows the admin to perform analysis by considering all the input conditions. Finally, a report is generated for the crop yield and the accuracy of the models are also generated. The accuracy which is near to 1 is considered as an ideal model and the model which has accuracy near to 0 are considered as unideal model. The input for the system will be a season, rainfall, area of production, crop type, district name, state name and output will be the production of crop yield and accuracy of each model.




### 4.1 Code submission (Github link):

https://github.com/akashhiremath608/upskillcampus/tree/main

### 4.2 Report submission (Github link) :

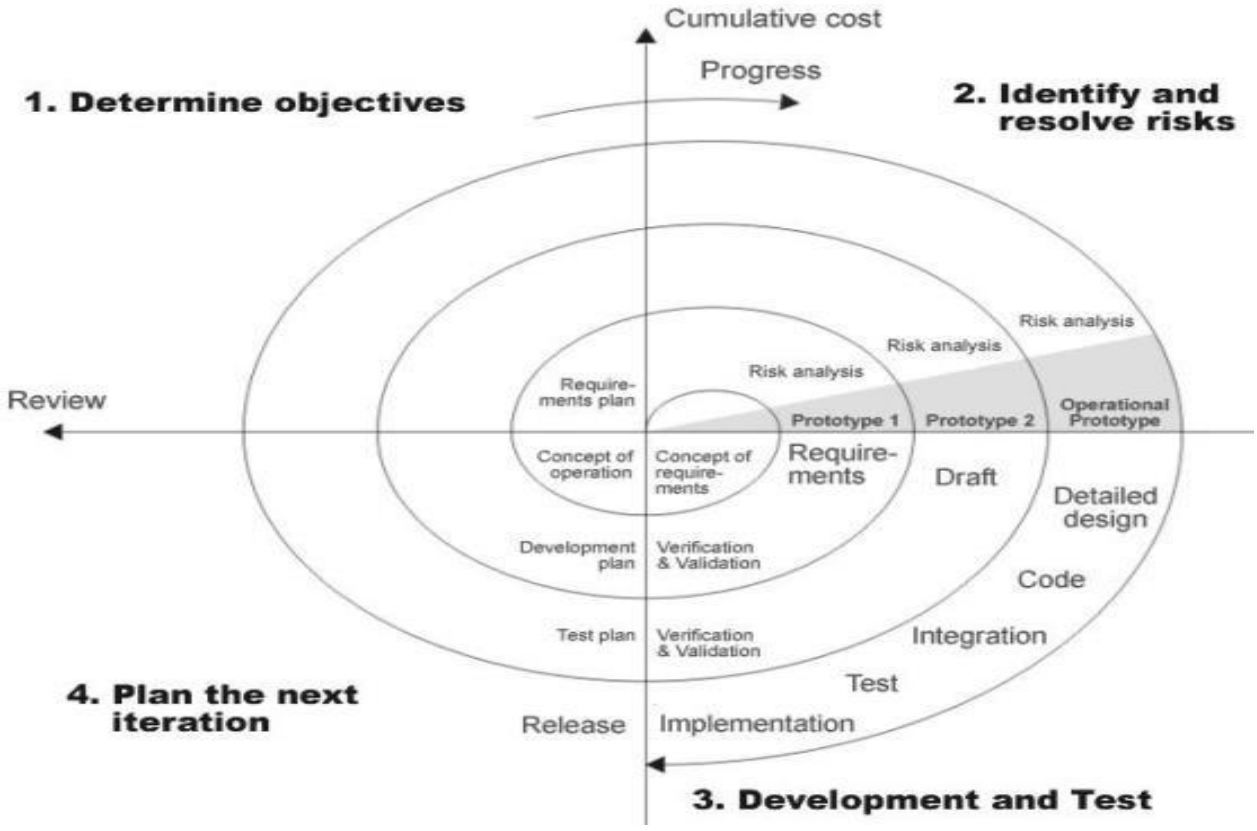
## 5 Proposed Design/ Model

This is achieved through these concepts.

(i) Random Forest regressor.
(ii) Gradient Boost regressor.
(iii) Decision
(iv) Tree regressor.

**System architecture consists of 6 modules namely,**

(i) Requirement gathering
(ii) Analysis
(iii) Design
(iv) Coding
(v) Testing
(vi) Maintenance



## (i) Requirement gathering stage:

This stage takes as its input the objectives defined within the high-level portion of the project set up. Each target will be broken down into one or extra specifications. These specifications describe the intended application's key functions, operational information areas, and reference in areas, as well as the first data entities. The key roles include managing sensitive processes as well as mission crucial inputs, output, and reports. These core functions, information regions, and data entities are organized according to a user class hierarchy. Each of these definitions is referred to as a Prerequisite. Requirements are diagnosed via means of precise requirement identifies and, at minimum, include a requirement identity and textual description. In this stage, all the necessities are well specified withinside the primary deliverable: the RTM, and an updated project set up.

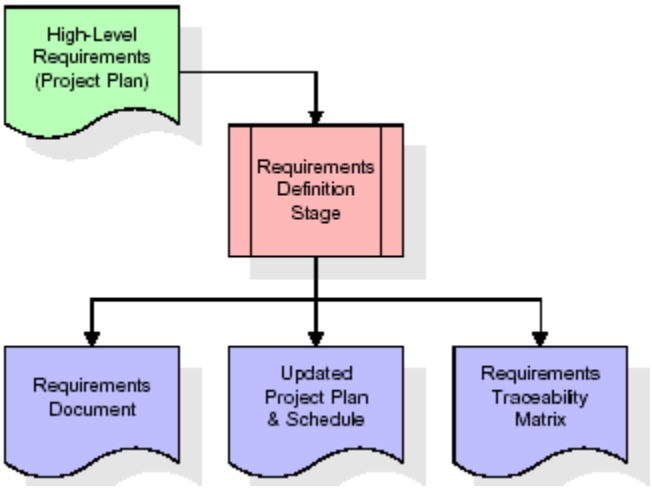



**Fig. 2:** Requirement gathering stage

## (ii) Analysis stage:

This planning stage establishes a bird's eye view of the intended software product and makes use of this to set up the fundamental challenge structure, examine feasibility and dangers related to the project, and describe suitable management and technical approaches. The maximum critical phase of the project pan is a list of high- degree product necessities, additionally known as goals. During the requirements specification stage, all the software program product necessities so as to be created a glide from one or more of these objectives. The minimal records and references to outside files can be included. The configuration control plan, the fine guarantee plan, and the project plan and schedule are the outputs of the project planning stage, with a complete list of deliberate

responsibilities for the approaching necessities degree and high- degree estimates of attempt for the out stages.

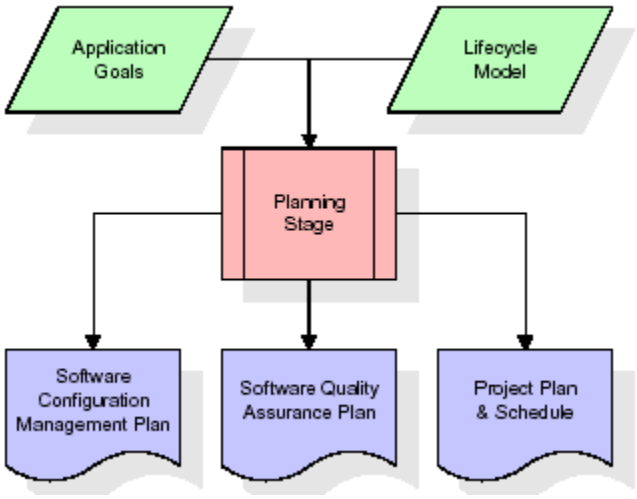



**Fig. 3:** Analysis stage

## (iii) Design stage:

It takes as its preliminary input the necessities recognized withinside the accredited necessities report. For every requirement, a set of one or more layout factors can be produced due to interviews, workshops, and/ or prototype efforts. Functional hierarchy diagrams, display format diagrams, tables of enterprise rules, enterprise process diagrams, pseudo code, and a entire entity dating diagram with a complete records dictionary are all examples of layout factors that designate the preferred software program functions in detail. These design elements are intended to provide enough information about the software so that professional programmers can create it with minimal help. The RTM is revised after the layout report is completed and accredited to signify that every layout characteristic is formally aligned with a precise requirement. The outputs of the design degree are the layout report, an up-to-date RTM, and an up to date venture plan.

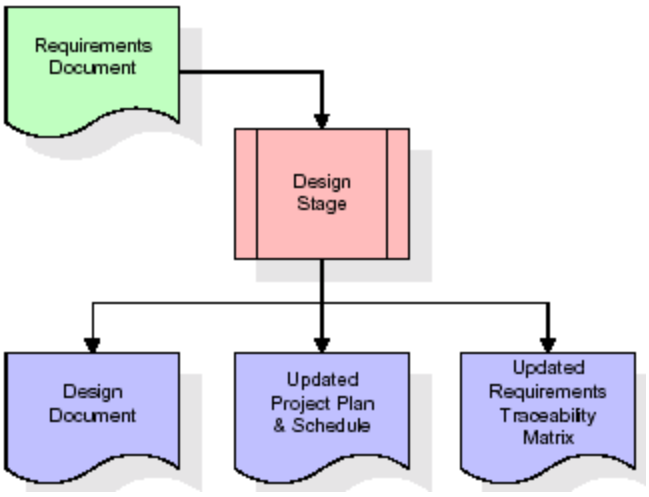



**Fig. 4:** Design stage

## (iv) Coding Stage:

The Software package artifacts, on-line facilitate and test records migrated from the event surroundings to a separate test environment. All test cases are run at this point to confirm that the program is true and complete. The test suite's prosperous completion demonstrates a stable and full migration capability. Production users' square measures are outlined and connected to their acceptable roles throughout this time, and reference information is finalized for production use. The Production Initiation Plan contains the ultimate reference knowledge and production user list. An associate integrated assortment of tools, an online support system, an implementation map, a development plan that identifies reference knowledge and production users, an approval arrangement that has the ultimate suite of test cases, and an updated project plan are all products of this level.

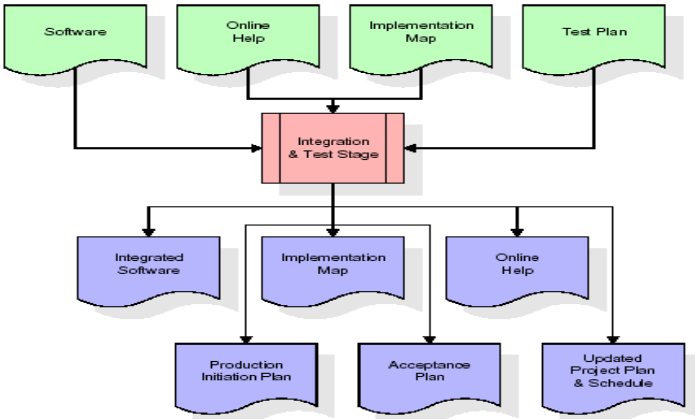



**Fig. 5:** Coding stage

## (v) Testing stage:

The software program artifacts, on-line help, and preliminary production statistics are loaded onto the production server. All test cases are run at this point to ensure that the program is right and complete. The test suite must be completed successfully before the program can be accepted by the customer. The customer formally approves the delivery of this system after customer workers have checked that the preliminary production statistics load is correct and that the test suite has been achieved with perfect results. A production application, a accomplished acceptance test suite, and a memorandum of customer acceptance of the system are the primary outputs of this level. Finally, the PDR enters the very last piece of real labor statistics into the project agenda and saves it as a everlasting undertaking record. The PDR "locks" the project at this factor via means of archiving all application objects, the implementation map, the supply code, and the documentation for future use.

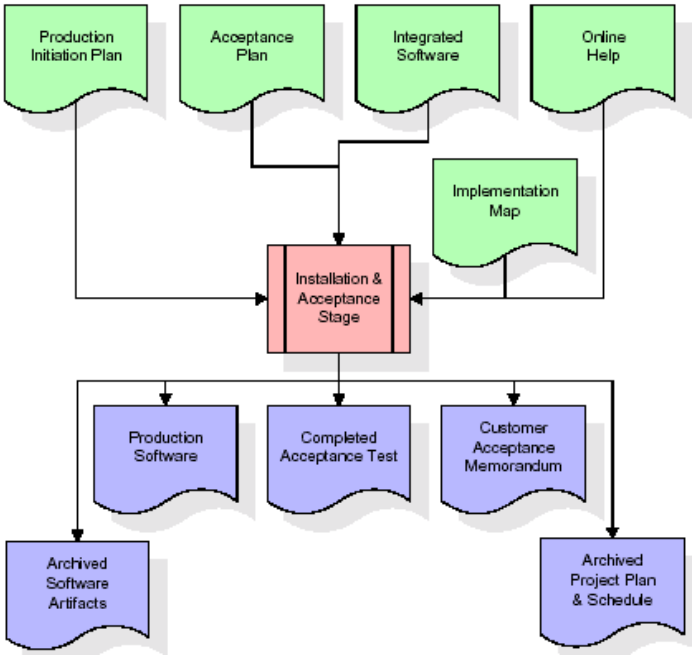



**Fig. 6:** Testing stage

### (vi) Maintenance:

The outer rectangle represents project maintenance; the maintenance team will begin by studying specifications and understanding documentation; after that, employees will be allotted work and receive coaching within the class to which they have been allotted.

**Training our model:**

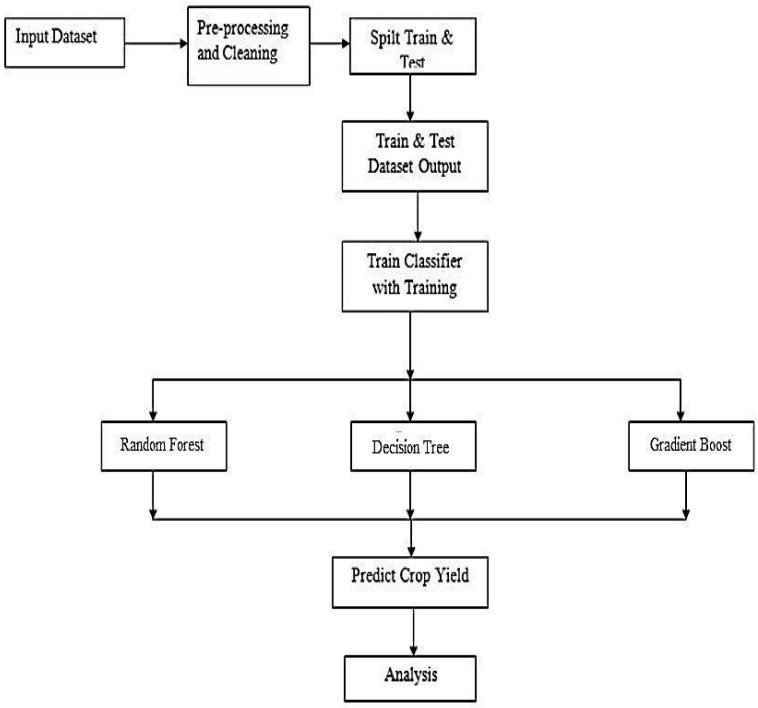



Fig. 7: Architecture Diagram

## 6 Methodology:

To train our model we need data. The collected data can have little 'NA' values filtered out in Python. Since the data set is composed of digital data, the robust scaling we use is very similar to normalization, but it uses interquartile range, and normalization compresses the data in units of 0 and 1.

**Ensemble Algorithms**

❖ On average, this is an improvement. Here, we add a meta model and use creases in addition to creases. The predictions of other models are used to train the basic meta-model.

❖ The entire training set is divided into two different sets (train and test/holdout datasets).

❖ We train the chosen base models with initial part (train dataset).

❖ Then we check them with the second half (holdout set).

❖ Now, the predictions obtained from the check half are inputs to the train higher level learner referred to as meta-model.

## Algorithms:

❖ **Random Forest Regression:** It generates multi decision trees from which each decision tress uses a part of data sample and predicts the result. Then, the result which was achieved by maximum number of trees is considered as the final prediction.

❖ **Decision Tree Regression:** Trees are constructed through an algorithmic approach that identifies ways to split the data set based on different conditions. It is one of the most widely used practical methods for supervised learning. These are non-parametric method used for both classification and regression.

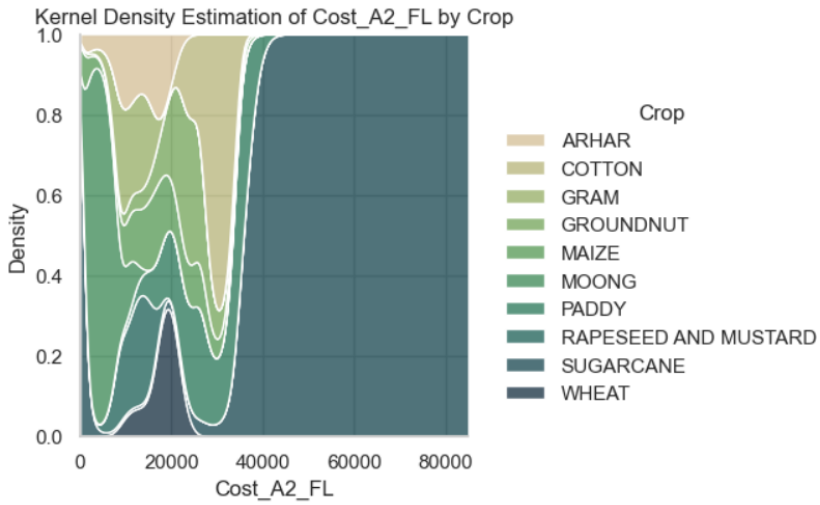
❖ **Gradient Boost Regression:** This method converts the weak learners into strong learners by boosting their capability. It is a sequential process of learning from the previous trees and increases the model accuracy.

### 6.1 Top 5 Data Representation

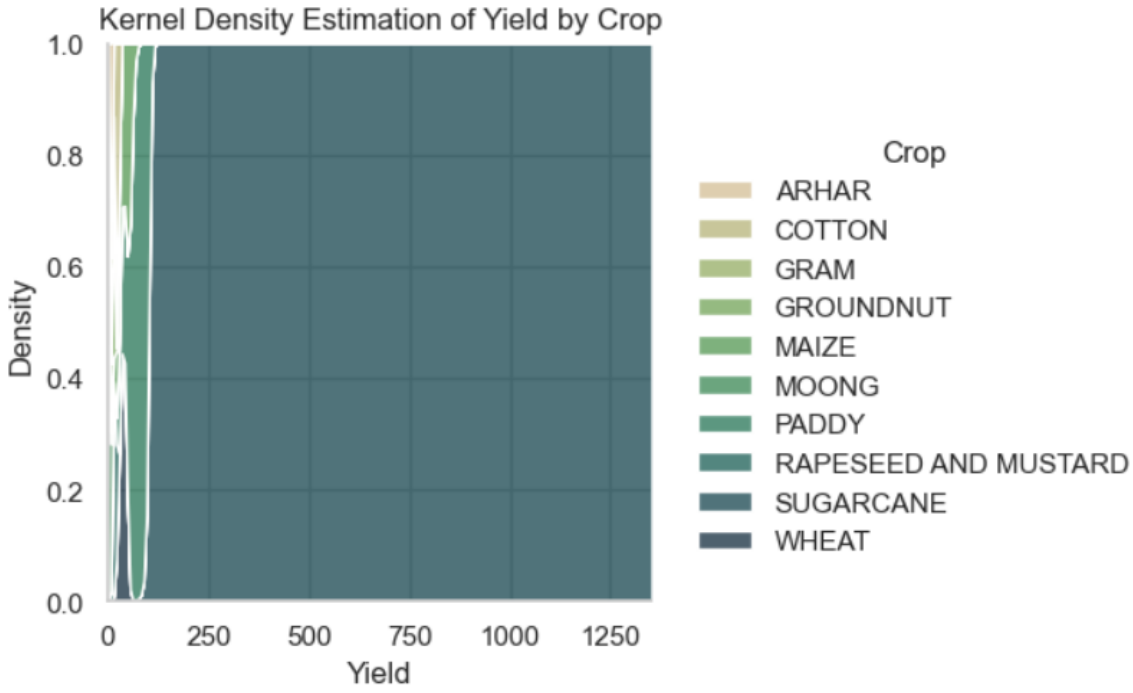
| | Crop | State | Cost_A2_FL | Cost_C2 | Cost_Production | Yield |
|---|---|---|---|---|---|---|
| 0 | ARHAR | Uttar Pradesh | 9794.050000 | 23076.740000 | 1941.550000 | 9.830000 |
| 1 | ARHAR | Karnataka | 10593.150000 | 16528.680000 | 2172.460000 | 7.470000 |
| 2 | ARHAR | Gujarat | 13468.820000 | 19551.900000 | 1898.300000 | 9.590000 |
| 3 | ARHAR | Andhra Pradesh | 17051.660000 | 24171.650000 | 3670.540000 | 6.420000 |
| 4 | ARHAR | Maharashtra | 17130.550000 | 25270.260000 | 2775.800000 | 8.720000 |

**Figure 1:Data**

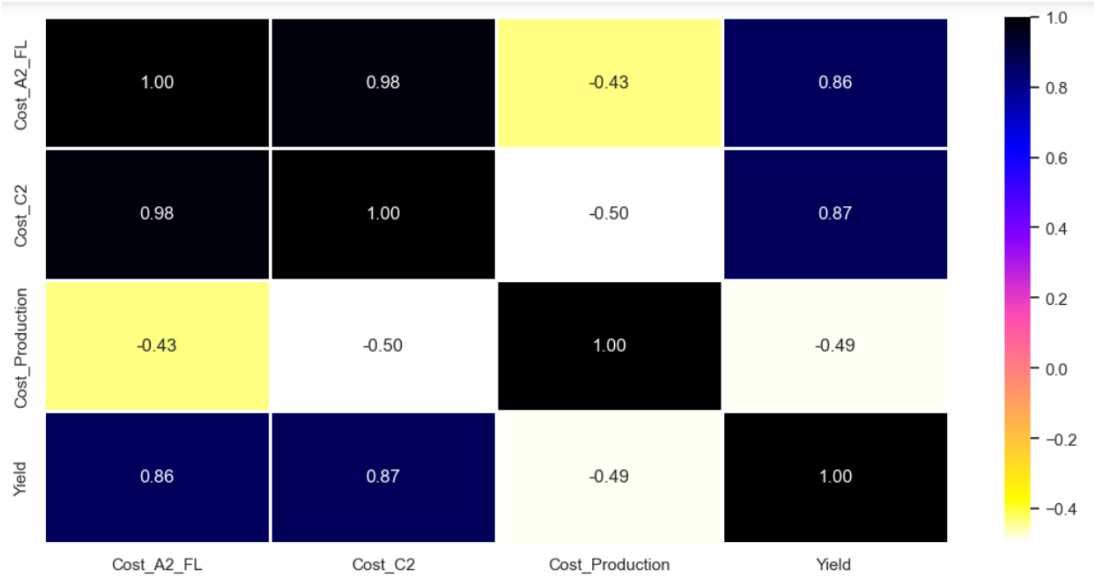
## 6.2 Kernal Density Estimation of cost_A2_FL by Crop




## 6.3 Kernal Density Estimation of Yield by Crop



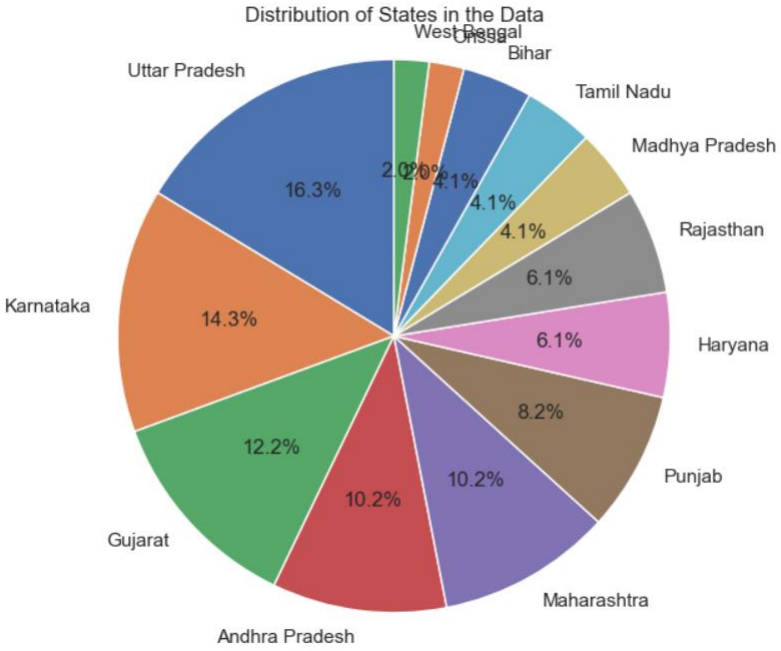
## 6.4 Correlation Map



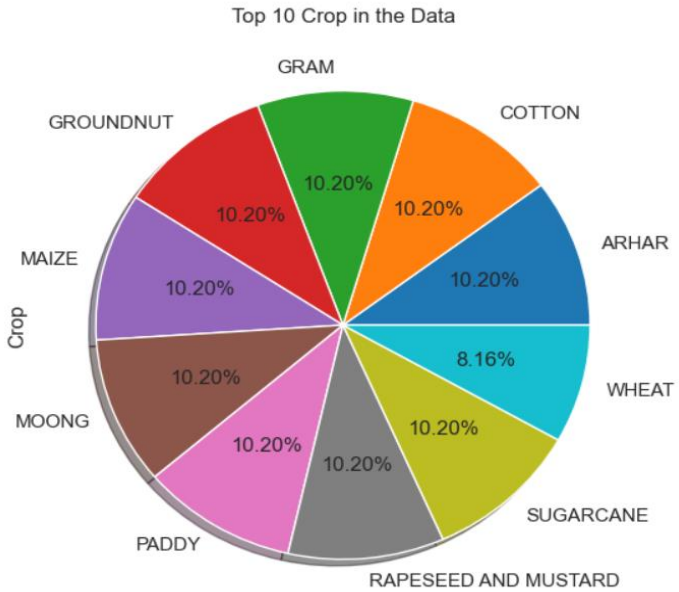

## 6.5 Average Yield

| Yield | Cost_A2_FL | Cost_C2 | Cost_Production |
|---|---|---|---|
| 1.320000 | 6440.640000 | 7868.640000 | 5777.480000 |
| 3.010000 | 5483.540000 | 8266.980000 | 2614.140000 |
| 4.050000 | 6204.230000 | 9165.590000 | 2068.670000 |
| 4.710000 | 13647.100000 | 17314.200000 | 3484.010000 |
| 5.900000 | 6684.180000 | 13209.320000 | 2228.970000 |
| 6.420000 | 17051.660000 | 24171.650000 | 3670.540000 |
| 6.700000 | 10780.760000 | 15371.450000 | 2261.240000 |
| 6.830000 | 8552.690000 | 12610.850000 | 1691.660000 |
| 7.470000 | 10593.150000 | 16528.680000 | 2172.460000 |
| 8.050000 | 12985.950000 | 18679.330000 | 2277.680000 |

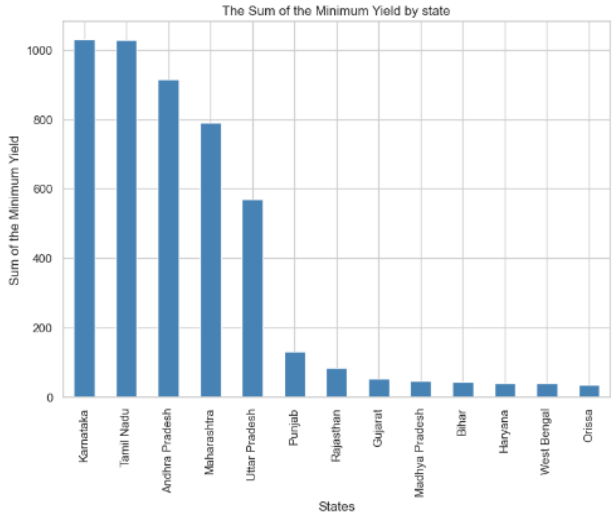
## 6.6 Distribution of States in Data

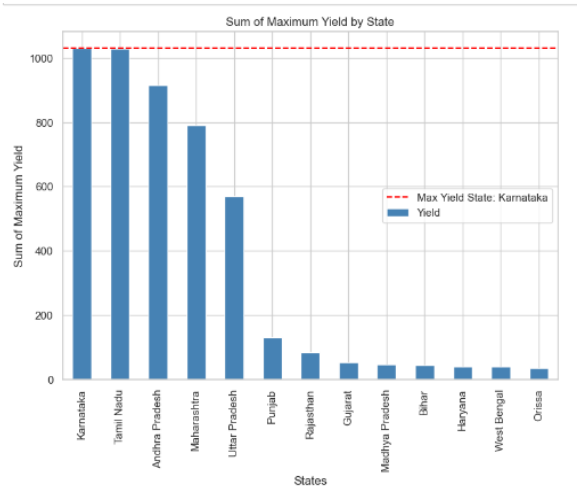



## 5.7 To 10 Crops in Data



## 6.8 Minimum and Maximum Yield by Crops



# 7 Performance Test

To implement Machine Learning for performance testing, we will discuss the stages involved to create and evaluate an ML model:

- Defining the parameters
- Generating training data
- Choosing an ML model
- Evaluating the model

**Defining the Parameters**

To begin, we must identify the inputs and define the expected outputs. Response Time is a critical parameter that needs to be closely monitored during performance testing. So, in this example, we will predict the Response Time of a page/request by feeding Total Samples, Sent Bytes, Received Bytes, and Request Name as inputs.

Output (Label) a Response Time

Input (Feature) a Threads, Sent Bytes, Received Bytes, Request/Page

**Generating Training Data**

The first and most essential step is to collect relevant data corresponding to our problem statement. We generated Input data by executing performance tests using JMeter in a controlled test environment (to eliminate noise due to server performance or additional network traffic).

Performance reports generated from these tests contain the following data: Sent Bytes, Bytes, Threads Request Name, and Response Time.

The performance test documented below generated 490 records for each page request.

| Threads | Ramp-Up(sec) | Total Pages per iteration | Iterations |
|---|---|---|---|
| 10 | 1 | 4 | 10 |
| 20 | 1 | 4 | 10 |
| 30 | 1 | 4 | 10 |
| ... | ... | ... | ... |
| ... | ... | ... | ... |
| 490 | 1 | 4 | 10 |

**Choosing an ML Model**

The process of training an ML model involves assigning an ML algorithm (the learning algorithm) to learn from the training data. ML model is the outcome created by the training process.

We installed the Machine Learning library and uploaded our performance test data in it. The library evaluated various algorithms with varying computations and settings to identify the best - performing model.

For our test data, the following algorithms were evaluated. The FastTreeRegression algorithm returned the highest accuracy at 82.3%.

| Trainer | RSquared | Absolute-loss | Squared-loss | RMS-loss |
|---|---|---|---|---|
| FastTreeRegression | 0.7834 | 78.65 | 13771.64 | 117.35 |
| FastTreeTweedieRegression | 0.7815 | 97.52 | 26585.33 | 163.05 |
| FastTreeRegression | -0.8185 | 309.62 | 113991.15 | 337.63 |
| FastTreeRegression | 0.823 | 75.78 | 11252.6 | 106.08 |
| FastTreeRegression | 0.8276 | 94.1 | 28917.49 | 170.05 |

**Evaluating the Model**

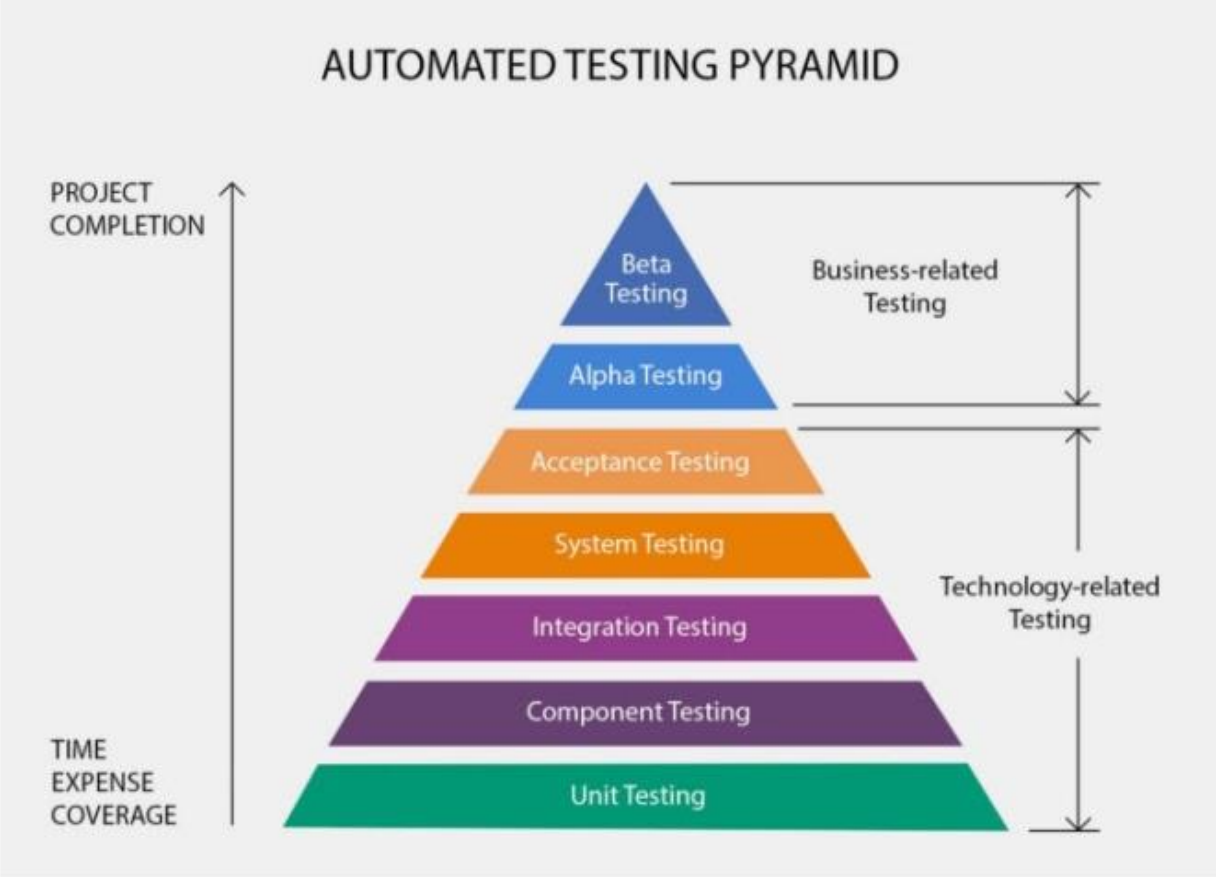
The model evaluation aims to estimate the generalization accuracy of a model based on future (unseen/out-of-sample) data. Now, that we had an ML Model, we used it to predict the response time for a new set of data (which was not used for training the model). The results of ML Model were tested against actual data, and the deviation was within acceptable limits.

## 7.1 Test Plan/ Test Cases

When executing a test case, you need to deal with five test case parameters:

- **The product's initial state or preconditions**– This is the required state of a test item and its environment before test case execution.
- **Data organization**– This involves selecting data from existing databases or creating, generating, and editing data for testing. This data must be high quality to produce high-quality results.
- **Dataset for input**– This is the data required for test execution.
- **Forecasted output**– This is the predicted behaviour of a model and what it should predict.
- **Expected output**– This parameter is the observable predicted behaviour of a test item, under specified conditions.

## 7.2 Test Procedure

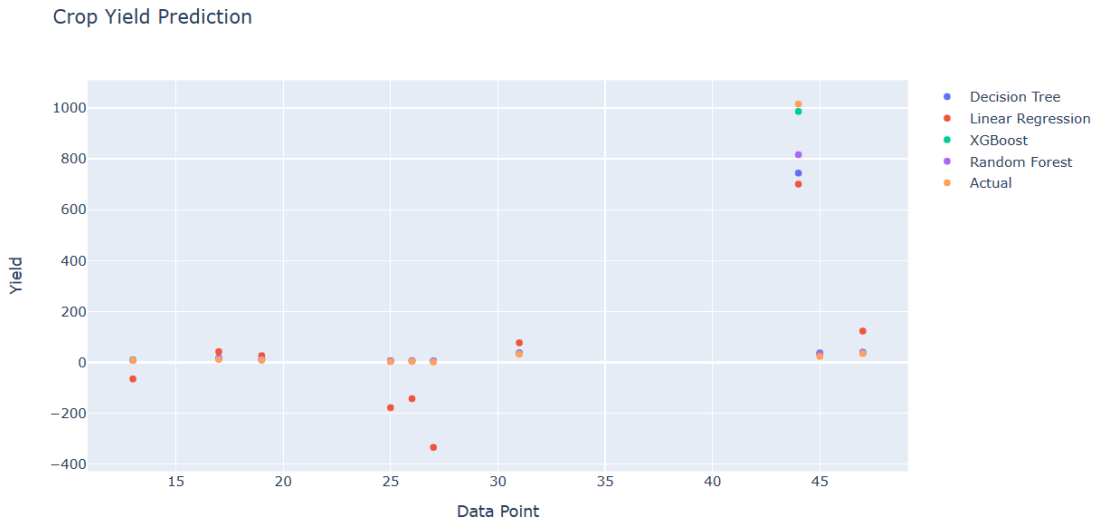



```
# Train the models
models = {
    'Decision Tree': DecisionTreeRegressor(),
    'Linear Regression': LinearRegression(),
    'XGBoost': XGBRegressor(),
    'Random Forest': RandomForestRegressor()
}

predictions = {}
for model_name, model in models.items():
    model.fit(X_train, y_train)
    predictions[model_name] = model.predict(X_test)
```

```
# Evaluate the models
evaluation = {}
for model_name, y_pred in predictions.items():
    mse = mean_squared_error(y_test, y_pred)
    mae = mean_absolute_error(y_test, y_pred)
    evaluation[model_name] = {'MSE': mse, 'MAE': mae}
```

## 7.3 Performance Outcome




```
Evaluation Results:
Decision Tree:
MSE: 130.98577000000006
MAE: 7.212999999999999
---
Linear Regression:
MSE: 28230.05404690465
MAE: 124.16507037394788
---
XGBoost:
MSE: 113.11858814325804
MAE: 6.5234736404419
---
Random Forest:
MSE: 4490.103665335023
MAE: 24.990350000000053
---
```

# 8 My learnings

A data science and machine learning internship offer's students and aspiring professionals a valuable opportunity to gain practical experience in the field of data-driven technologies. During such internships, participants typically work alongside experienced data scientists and machine learning engineers, applying theoretical knowledge to real-world projects and challenges.

As Intern I learned about various responsibilities, including:

1) Data Collection and Cleaning
2) Exploratory Data Analysis (EDA
3) Feature Engineering
4) Model Development
5) Data Visualization
6) Evaluation and Optimization
7) Machine Learning Pipelines
8) Collaboration and Communication
9) Domain-specific Projects

Overall, a data science and machine learning internship serve as a bridge between academic learning and practical application, allowing participants to develop essential skills and a deeper understanding of the data science workflow. It's a valuable stepping stone for those aspiring to build a career in the rapidly evolving field of data science and machine learning.

# 9 Future work scope

The future work scope for data science and machine learning is exceptionally promising, with increasing demand and opportunities across various industries. As technology continues to evolve and generate massive amounts of data, the role of data scientists and machine learning engineers becomes even more critical

Here are some key trends and areas of growth in this field:

- Artificial Intelligence (AI) Integration
- Automation and Decision Support
- Natural Language Processing (NLP
- Computer Vision
- IoT and Sensor Data Analysis
- Personalization and Recommender Systems
- Healthcare Analytics
- Finance and Fintech
- Continuous Learning and Upskilling