Capstone Project – 1

1. **Project overview :**

* **Project name :** Crop yield prediction
* **Objective :** Predict the Crop yield (Production per unit area).
* **Problem Type :** Supervised Learning (Regression problem)

This dataset encompasses agricultural data for multiple crops cultivated across various states in India from the year 1997 till 2020. The dataset provides crucial features related to crop yield prediction, including crop types, crop years, cropping seasons, states, areas under cultivation, production quantities, annual rainfall, fertilizer usage, pesticide usage, and calculated yields.

The objective of this project is to develop a predictive model for crop yield prediction using machine learning regression techniques.

The project involves using different regression techniques like Linear Regression, Decision Trees, Random Forest, Ridge Regression, Lasso Regression, and XGBoost, comparing their performance based on metrics such as R² score, Mean Squared Error (MSE), and Root Mean Squared Error (RMSE).

1. **Data Description**

* **Source :** The source of dataset is Kaggle.
* **Time Period :** 1997 - 2020
* **Target Variable:** Crop Yield (Production per unit area)

**Features (Input Variables) :**

* **Crop** : The name of the crop cultivated.
* **Crop\_Year** : The year in which the crop was grown.
* **Season** : The specific cropping season (e.g., Kharif, Rabi, Whole Year).
* **State** : The Indian state where the crop was cultivated.
* **Area** : The total land area (in hectares) under cultivation for the specific crop.
* **Production** : The quantity of crop production (in metric tons).
* **Annual Rainfall** : The annual rainfall received in the crop-growing region (in mm).
* **Fertilizer** : The total amount of fertilizer used for the crop (in kilograms).
* **Pesticide** : The total amount of pesticide used for the crop (in kilograms).
* **Yield** : The calculated crop yield (production per unit area).

1. **Resources:**

* **Data set source:**

The source of the data set is Kaggle

* **Software:**

Python platform jupyter is being used to build the machine learning algorithm

1. **Individual details :**

* **Name :** Chandrapal singh rathore
* **E-mail ID :** [csr19b@gmail.com](mailto:csr19b@gmail.com)
* **Contact details : +91 7014233145**

1. **Project Deliverables :-**

**Data collection and preprocessing**

* 1. Dataset have 19689 rows and 10 columns.
  2. The dataset does not contain any null value.
  3. The dataset does not contain any duplicate row.
  4. There is no such column that needs to be deleted. Every column is useful.
  5. We used label encoder method to convert the object type variables into numeric.
  6. Using the boxplot, we recognized a outlier which was affecting all features. Then we dropped the outlier row.
  7. Then the is being visualized using different charts.

**Data summary**

* The heatmap clearly shows that when rainfall is higher, less fertilizer and pesticide are needed, and vice versa.
* Crops like arecanut and black pepper needs more than 2000 mm rainfall.
* Guar seeds is the crop which require least rainfall about 600 mm.
* The average rainfall is between 1000 and 1500 mm, which meets the needs of all the crops.
* The yield of coconut is the highest, in all the state lots we created.
* In the first state group, Arunachal Pradesh has the highest rainfall, while Delhi has the least.
* In the second group, Haryana receives the minimum rainfall, and Goa experiences the maximum.
* In the third group, Meghalaya has the highest rainfall, while Madhya Pradesh receives the least.
* In the fourth group, Sikkim receives the maximum rainfall, and Punjab has the minimum.
* In the fifth group, Uttar Pradesh receives the least rainfall, and Tripura experiences the maximum.
* Overall meghalaya recieves the maximum rainfall and haryana recieves the least.
* In the first state group, Bihar has the highest use of pesticide, while Delhi has the lowest.
* In the second state group, Gujarat has the highest use of pesticide, while Goa has the lowest.
* In the third state group, Maharashtra has the highest use of pesticide, while Meghalaya has the lowest.
* In the fourth state group, Punjab has the highest use of pesticide, while Puducherry has the lowest.
* In the fifth state group, Uttar Pradesh has the highest use of pesticide, while Tripura has the lowest.
* Overall Uttar Pradesh has the highest use of pesticide, while puducherry has the lowest.
* The use of pesticides gradually increased from 2007 to 2011.

**Data splitting**

Dataset is being splitted into training and testing set.

75% of the dataset is used for training.

25% of the dataset is used for testing.

**Model Selection & Training**

* **Model Selection**:

We selected the following regression models for prediction:

* + **Linear Regression**
  + **Random forest regressor**
  + **Decision tree regressor**
  + **XGBoost regressor**
  + **Gradient boosting**
  + **KNN**
  + **Bagging regressor**
  + **Voting regressor**

**Model Evaluation**

* **Performance Metrics**:  
  We evaluated the performance of the models based on the following metrics:
  + **R² Score**: To measure the proportion of variance explained by the model.
  + **Mean Squared Error (MSE)** : To assess the accuracy of predictions.

#### **Model Improvement**

* **Ensemble Techniques**:

We explored ensemble methods, particularly **Voting Regressor** and **Bagging Regressor** and **Gradient boosting** to combine the strengths of multiple models and improve predictive accuracy.

**Conclusions from regression models**

* We used eight different models for regression prediction, including several ensemble models.
* We also experimented with the hyperparameters of both the Gradient Boosting and Bagging Regressor.
* After evaluating the performance of all the models, we concluded that **XGBoost** was the best performing model, while **Linear Regression** was the least effective.

## **Accuracy scores of the regression models**

|  |  |  |
| --- | --- | --- |
| Model | R2\_score | Mean squared error |
| Linear regression | 0.344685 | 479233.549002 |
| Random forest | 0.976318 | 17318.579027 |
| Decision tree | 0.959757 | 29429.478210 |
| XGBoost | 0.980819 | 14027.084577 |
| Gradient boosting | 0.976445 | 17225.896995 |
| Knn | 0.893544 | 77851.730215 |
| Bagging regressor | 0.978041 | 16058.468885 |
| voting regressor | 0.974227 | 18848.042776 |

**Cross validation**

We applied K-Fold cross-validation to different models and found that Gradient Boosting performed the best, while Linear Regression performed the worst.

# **Accuarcy scores after applying kfold cross validation**

|  |  |
| --- | --- |
| Model | Accuracy |
| Linear regression | 0.3341494951787076 |
| Random forest | 0.9576189800114829 |
| Decision tree | 0.9520780850817383 |
| XGBoost | 0.9627274103272097 |
| Gradient boosting | 0.9681057067594943 |
| Knn | 0.9162536943287367 |
| Bagging regressor | 0.9599420293808769 |

**Conclusion**

After executing all the regression models, both with and without cross-validation, we concluded that XGBoost, with an accuracy of 0.980819 before applying cross-validation, is our final model.