Predictive Analytics for Precision Agriculture

Abstract:

- The project focuses on estimating crop yield in Buldhana using machine learning techniques.
- The dataset includes various features such as NDVI, LAI, climate parameters, and crop conditions.
- The goal is to develop a model that accurately predicts crop yield based on these features.

Introduction:

- Overview of Buldhana Yield Estimation ML Project.
- Utilize machine learning to predict crop yield.
- Explore the impact of various features on crop yield.

Problem Statement:

- Predicting crop yield in Buldhana.

Dataset:

The dataset used in this project contains information related to

- Experimental weight
- NDVI (Normalized Difference Vegetation Index)
- LAI (Leaf Area Index)
- Rainfall
- Temperature
- Humidity
- Sun hours
- FAPAR (Fraction of Absorbed Photosynthetically Active Radiation)
- Dry matter (biomass)
- Ground cover (FCover)

Data Cleaning and Preprocessing:

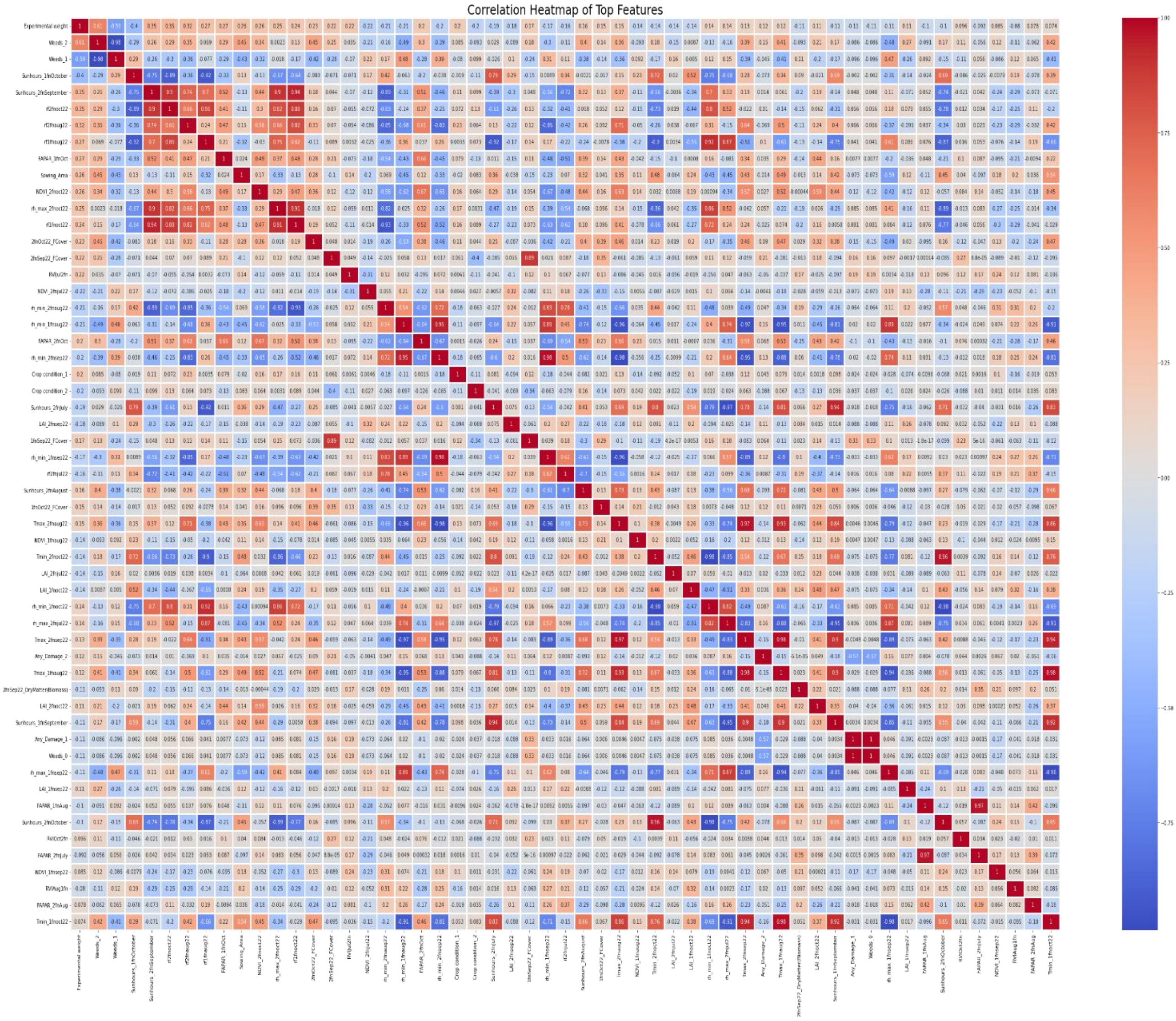
- Replaced **'No data'** values with **Nan**.
- Imputed missing values using **mean** or specific values for certain features.
- One-hot encoded categorical variables **like ('Any Damage,' 'Weeds,' and 'Crop condition').**
- The cleaned dataset contains **127 samples with 95 features**.
- The data has been cleaned to improve the quality of the data outliers must not present in the dataset.
- So, I used IsolationForest module to remove the outliers with contamination **0.05**
- After removing the outliers the dataset contains **120 samples and 95 features.**

Feature Selection Techniques:

- **Correlation Analysis:** The correlation analysis has been performed to extract best features which are highly correlated with the dependent variable. According to our data the best **55** features has been extracted,
- Those features are:

['Experimental weight', 'Weeds_2', 'Weeds_1', 'Sunhours_1fnOctober', 'Sunhours_2fnSeptember', 'rf2fnoct22', 'rf2fnaug22', 'rf1fnaug22', 'FAPAR_1fnOct', 'Sowing_Area', 'NDVI_2fnoct22', 'rh_max_2fnoct22', 'rf1fnoct22', '2fnOct22_FCover', '2fnSep22_FCover', 'RVIJul2fn', 'NDVI_2fnjul22', 'rh_min_2fnaug22', 'rh_min_1fnaug22', 'FAPAR_2fnOct', 'rh_min_2fnsep22', 'Crop condition_1', 'Crop condition_2', 'Sunhours_2fnJuly', 'LAI_2fnsep22', '1fnSep22_FCover', 'rh_min_1fnsep22', 'rf2fnjul22', 'Sunhours_2fnAugust', '1fnOct22_FCover', 'Tmax_2fnaug22', 'NDVI_1fnaug22', 'Tmin_2fnoct22', 'LAI_2fnjul22', 'LAI_1fnoct22', 'rh_min_1fnoct22', 'rh_max_2fnjul22', 'Tmax_2fnsep22', 'Any_Damage_2', 'Tmax_1fnaug22', '2fnSep22_DryMatter(Biomass)', 'LAI_2fnoct22', 'Sunhours_1fnSeptember', 'Any_Damage_1', 'Weeds_0', 'rh_max_1fnsep22', 'LAI_1fnsep22', 'FAPAR_1fnAug', 'Sunhours_2fnOctober', 'RVIOct2fn', 'FAPAR_2fnJuly', 'NDVI_1fnsep22', 'RVIAug1fn', 'FAPAR_2fnAug', 'Tmin_1fnoct22']

- Here, is the heatmap for the correlation:



- **OLS (Ordinary Least Squares):** The another feature selection to extract the best features based on the **significance value(0.000 to 0.05)** which is nothing but **p-value.** The Backward-Elimination method has been used to eliminate the column based on p-value.

['Weeds_2', 'Weeds_1', 'rf2fnoct22', 'rf2fnaug22', 'rf1fnaug22',

'rh_max_2fnoct22', 'RVIJul2fn', 'NDVI_2fnjul22', 'rh_min_2fnaug22', 'Crop condition_1', 'Sunhours_2fnJuly', '1fnSep22_FCover', 'rf2fnjul22', 'Sunhours_2fnAugust', 'Tmin_2fnoct22', 'rh_min_1fnoct22',

'Any_Damage_1', 'Weeds_0', 'rh_max_1fnsep22', 'FAPAR_1fnAug',
'Sunhours_2fnOctober', 'FAPAR_2fnJuly', 'FAPAR_2fnAug',
'Experimental weight']

- After performing the features selection techniques the data contains 120 samples and 24 features.

Diagnostic tests for regression analysis:

- **Durbin-Watson:** I got the Durbin-watson value as 0.09158803851713097. The range of the durbin-watson must be **0 to 4.** The result which is obtained also placed between 0 to 4 so the data is suitable for Linear regression.

Pre-Processing The Data:

- **Standardization:** Standard scaling is used to transform the features of a dataset so that they have a mean of 0 and a standard deviation of 1. It helps in improving the performance and convergence of certain machine learning algorithms. This process is commonly applied before feeding the data into machine learning models to ensure that features with different scales do not disproportionately influence the model**.**

Model Selection and Cross-Validation:

- **Train-Test-Split:** The data must split into train data and test to perform the models here, we split the data with test_size of 0.2 means 80% of the data has been declared as training data and 20% of data has been declared as testing data with the random_state of 1234 .The data has been splited like this
- The xtrain data has been shaped like (96, 23)
- The xtest data has been shaped like (24, 23)
- The ytrain data has been shaped like (96, 1)
- The ytest data has been shaped like (24, 1)

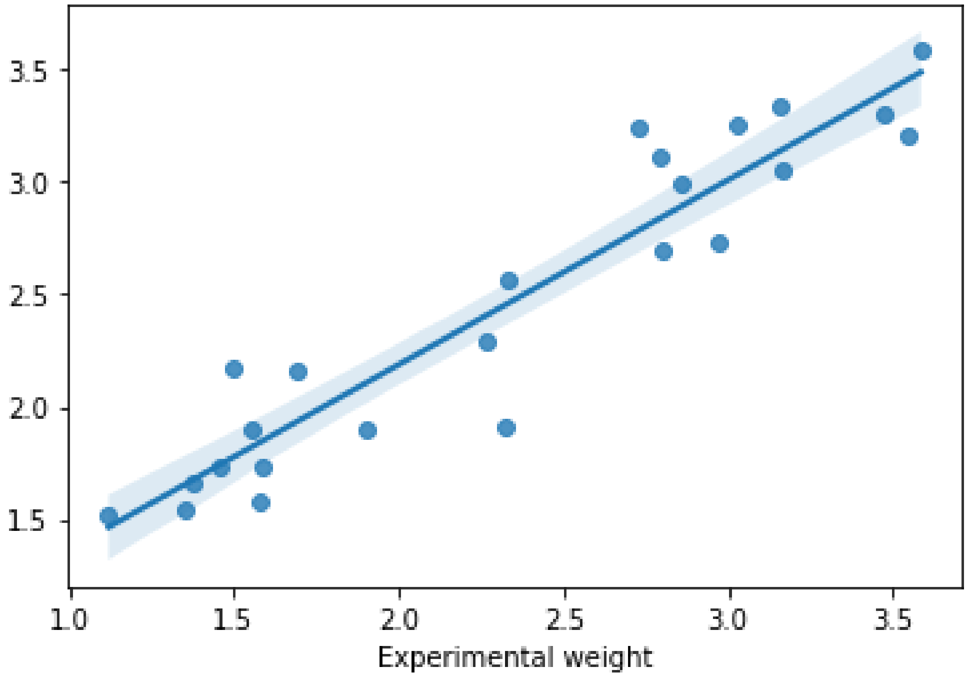
- First, I have been performing the linear regressor because of the Durbin Watson has been in the range.

- **Linear regressor performance:**

MSE VALUE (Linear Regression): **0.08665998272687653**

MAE VALUE (Linear Regression): **0.24186614805211626**

Accuracy (Linear Regression): **0.8577619601563689**



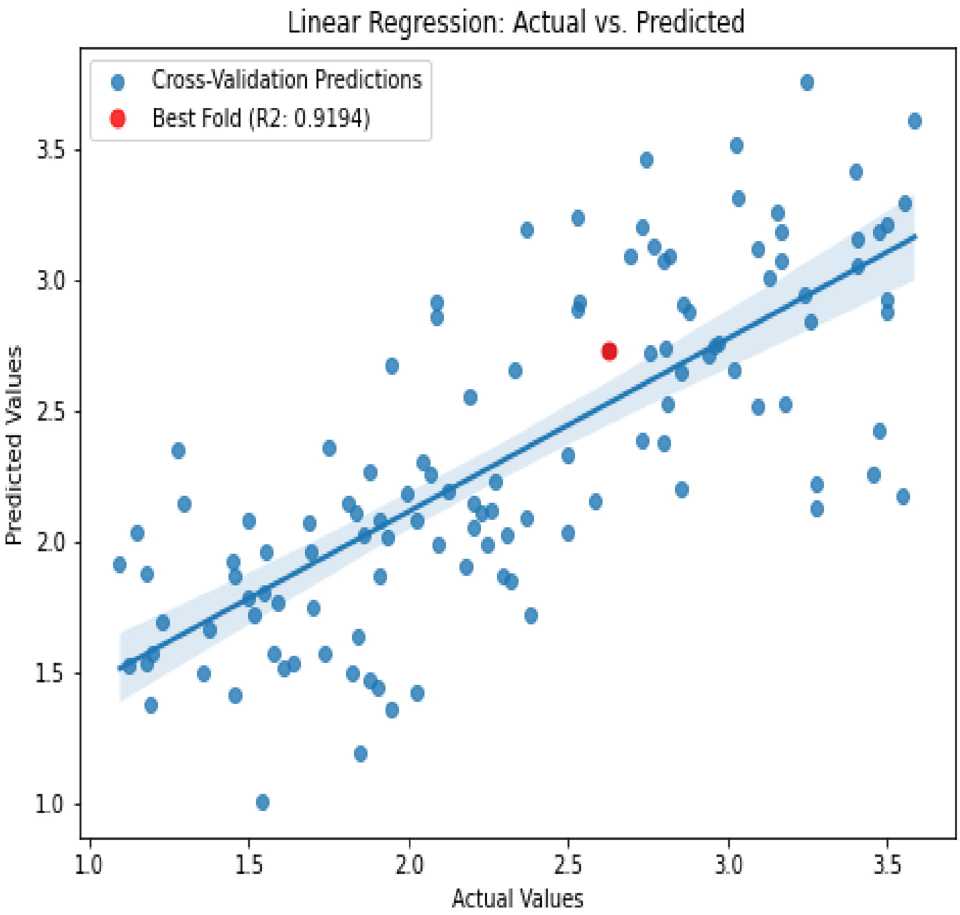
- The accuracy is good to improve the accuracy I performed cross validations to linear regressor cross_val_score, cross_val_predict with **cv=7** and to evaluate the model's performance and compute R2 scores for each fold. To best fold apply the cross validation to the entire dataset and visualize the regression results using Seaborn. Plot the overall predictions and highlight the predictions from the best fold in red and display the scores of cross validation scores and mean cross validation score.

- The results after the performing the cross validation for linear regressor are:

- Cross-Validation Scores (R2): [0.57856003 0.5103502 0.29704371 0.41147993 0.10710294 0.40491925, **0.91941199**]

- By performing the cross validation the accuracy has been improved to **85%** to **91%**

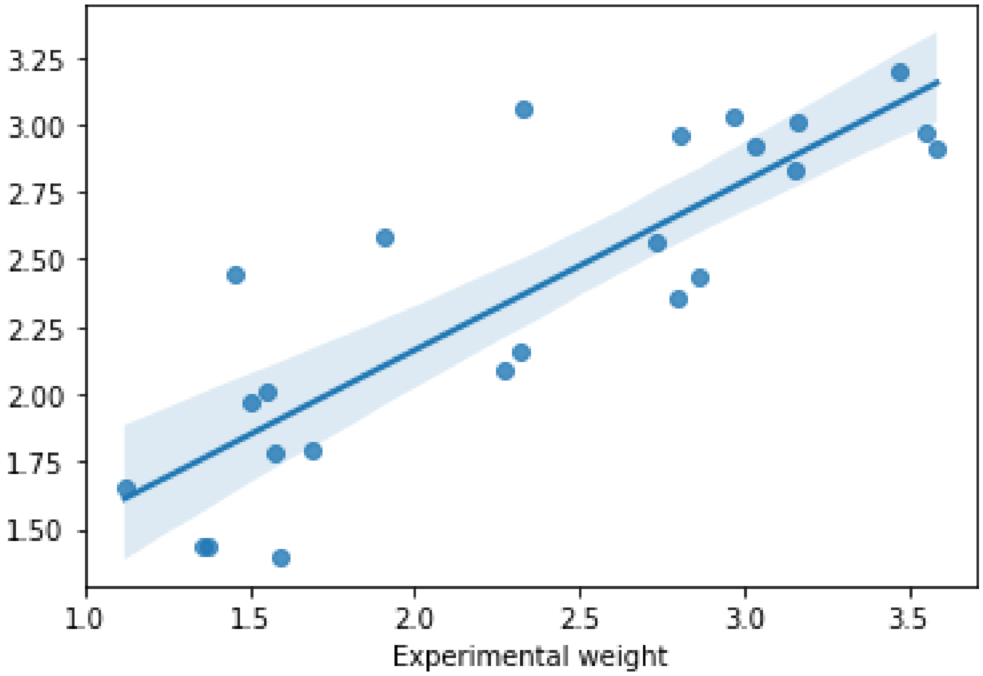
- Mean Cross-Validation Score: 0.46126686448631865



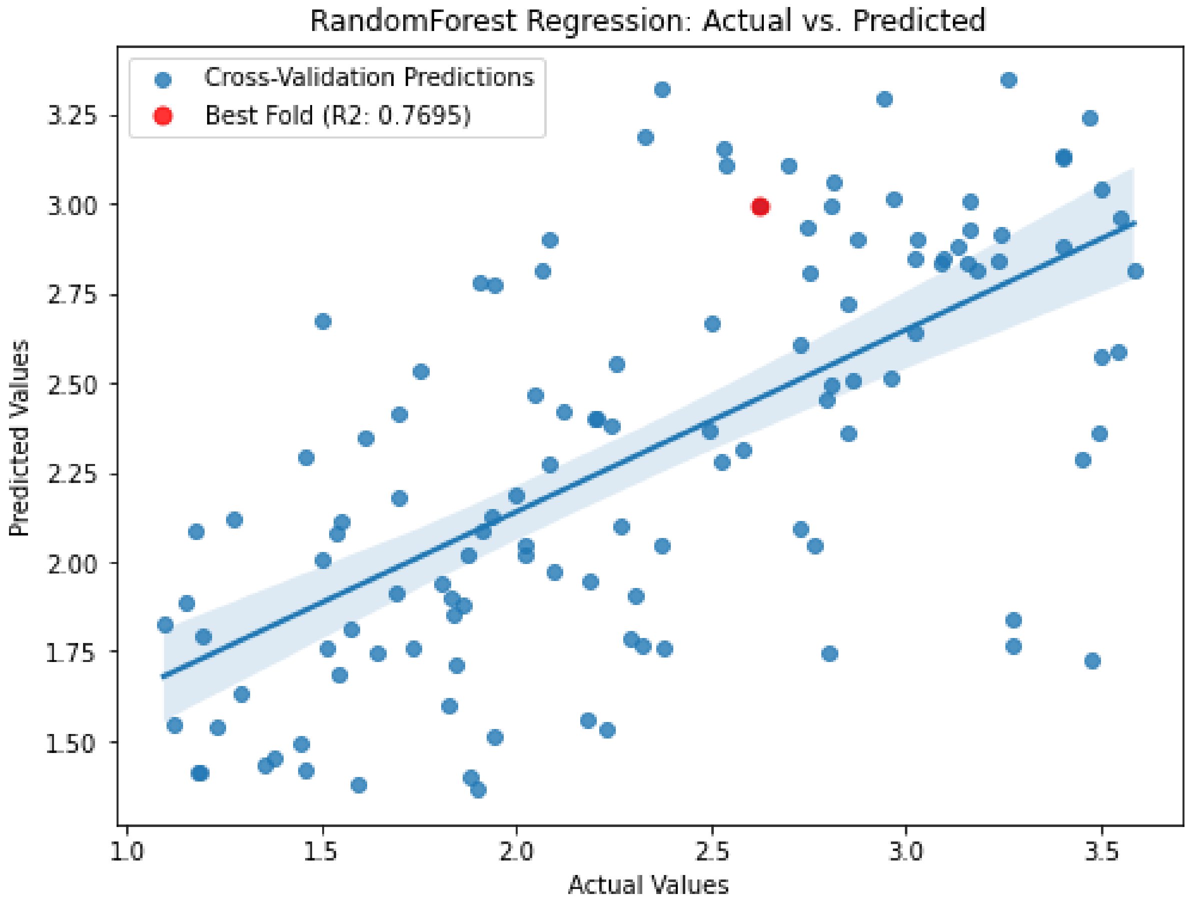
- The aiming accuracy has been achieved but how the other models like random forest regressor, decision tree regressor, boosting techniques like xgb, gradient boosting.

- **Random Forest Regressor:**

- MSE VALUE (RandomForest Regressor) 0.17755749701961646
- MAE VALUE (RandomForest Regressor) 0.34101308035714295
- Accuracy (RandomForest Regressor): **0.7085687125601172**



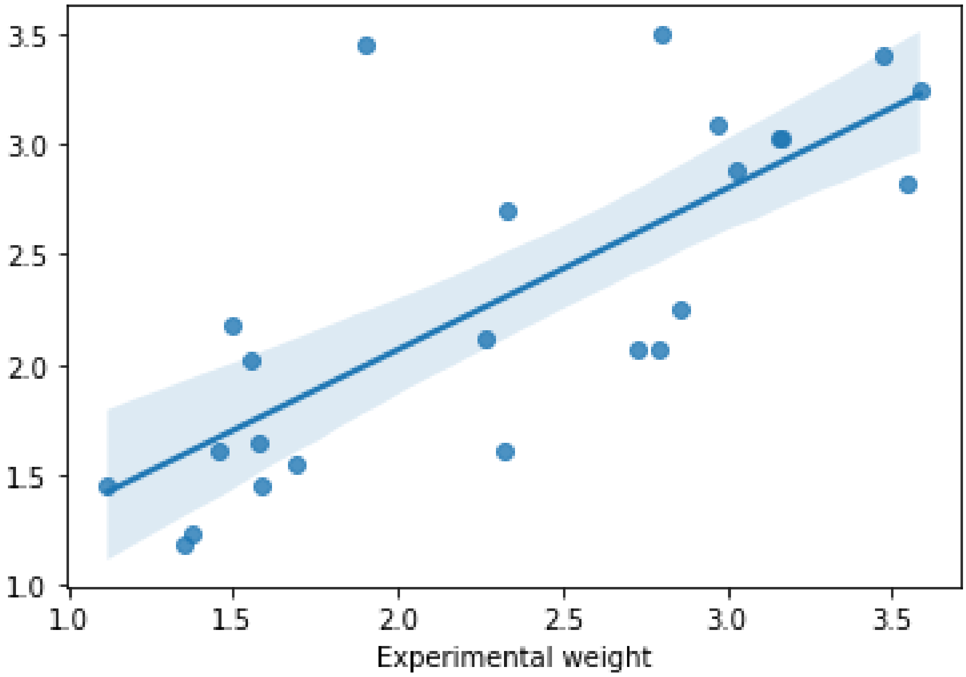
- The random forest has achieved the 70% accuracy lets apply the cross validation
- **Random Forest with CV:**



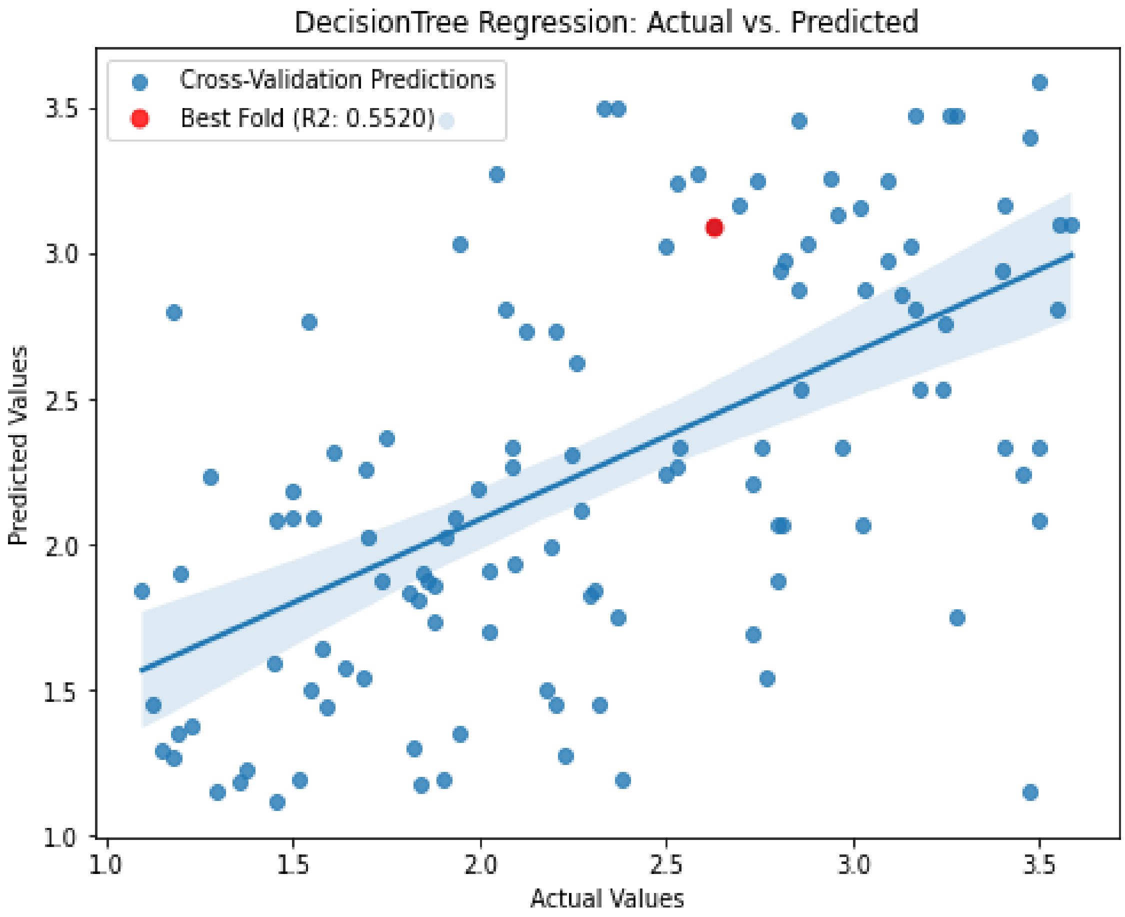
- Cross-Validation Scores (R2): [0.24989712, 0.33879777, 0.30155282, 0.44407119 ,0.05767121 0.19809187, **0.76949555**]
- The accuracy has been improved from **70%** to **76%** when we applying the cross validation.
- Mean Cross-Validation Score: 0.32060501540770403

- **DecisionTreeRegressor:**

- MSE VALUE (DecisionTreeRegressor) 0.2727374999999999
- MAE VALUE (DecisionTreeRegressor) 0.39499999999999996
- Accuracy (DecisionTreeRegressor): **0.552346467525651**

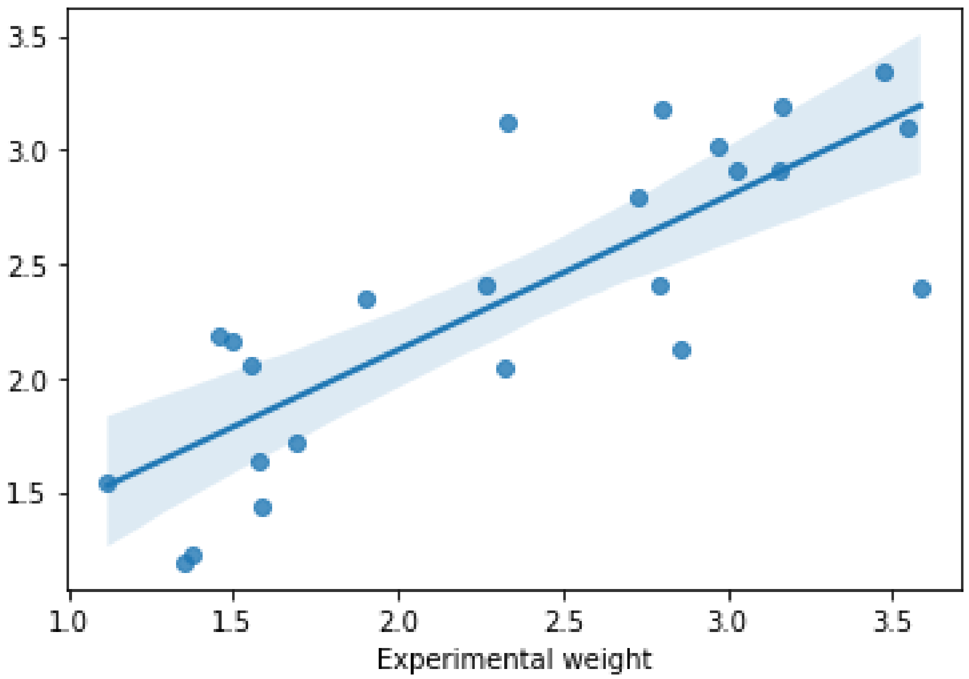


- The decision tree achieved the accuracy of 55% lets apply the cross validation
- **DecisionTreeRegressor with cv:**
- Cross-Validation Scores (R2): [-0.23059543 -0.05853176 -0.04558426 -0.13020669 -0.25952179 -0.4383837, **0.55198223**]
- The accuracy has been same with accuracy of 55% no change in the accuracy.
- Mean Cross-Validation Score: -0.08726305638607967
- The mean cross validation score has been got the negative value .

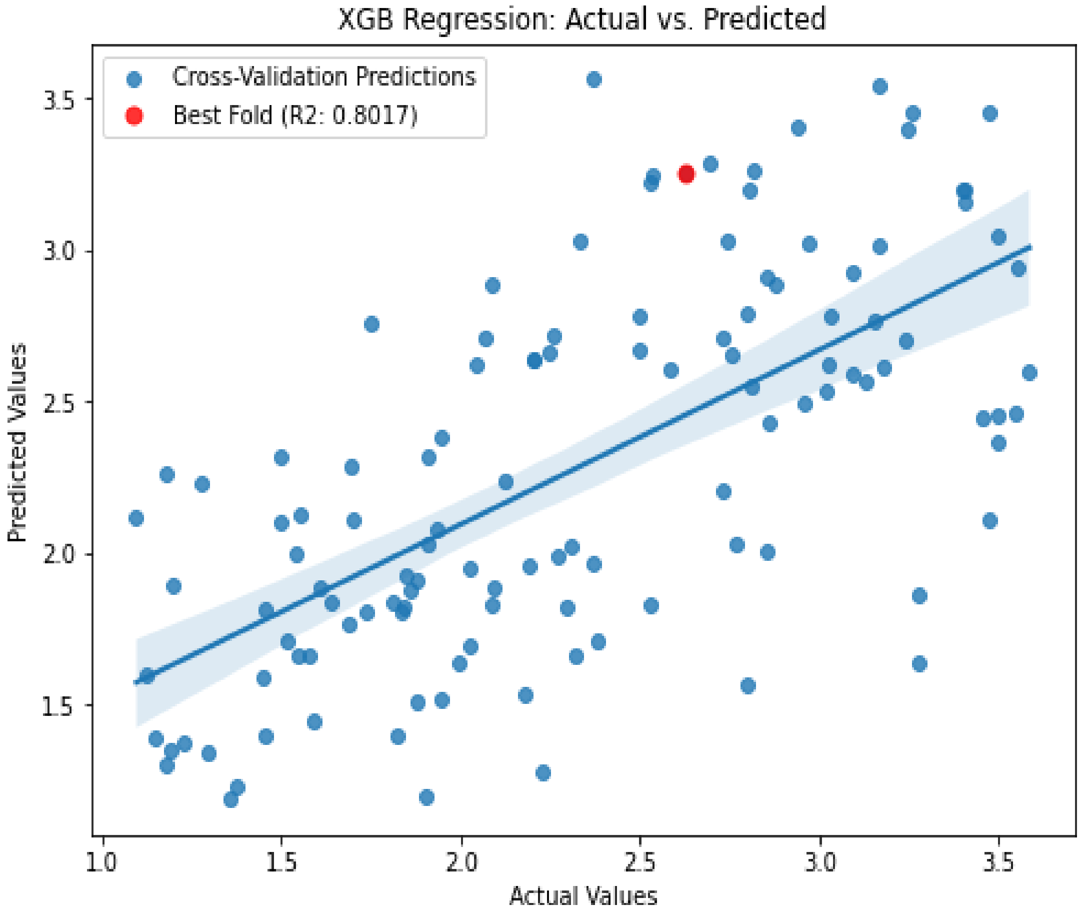


- **XGBRegressor:**

- MSE VALUE (XGBRegressor) 0.20609408394687248
- MAE VALUE (XGBRegressor) 0.3452239392201106
- Accuracy (XGBRegressor): **0.6617306211984691**



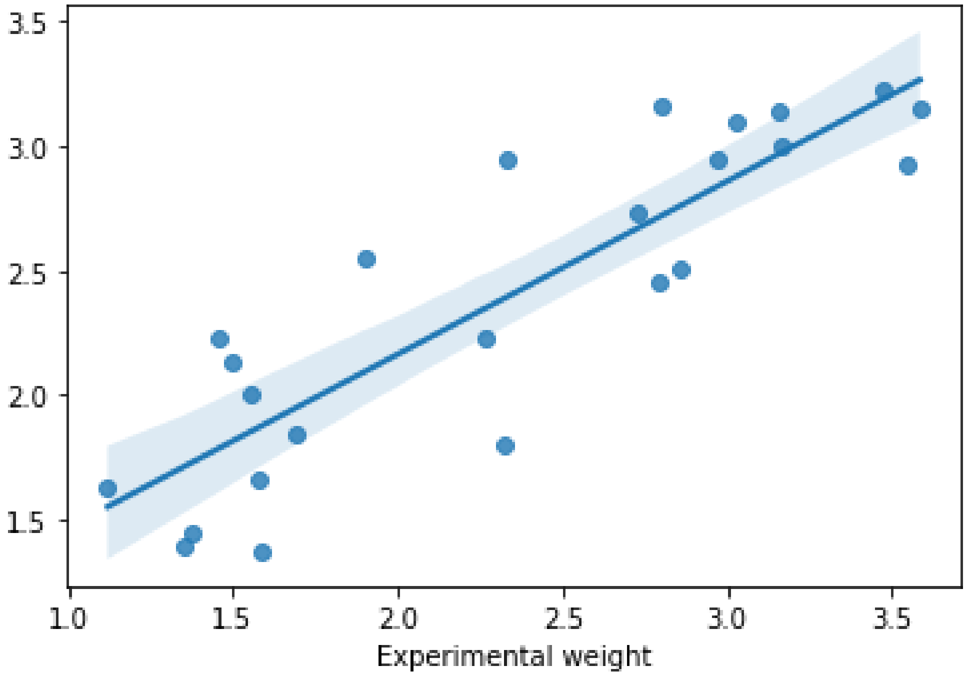
- The xgb regressor achieved the accuracy of 66% let's apply the cross validation to xgb.
- **XGBRegressor with CV:**



- Cross-Validation Scores (R2): [0.17442693, 0.46389482, -0.07239256, 0.07750235, 0.04743424 0.04868224**, 0.80165027**]
- The accuracy has been improved from **66%** to **80%** when we applying the cross validation.
- The xgb regressor also achieved the accuracy 80%.
- Mean Cross-Validation Score: 0.2201711850761083.

- **GradientBoostingRegressor:**

- MSE VALUE (GradientBoostingRegressor) 0.15268532051843461
- MAE VALUE (GradientBoostingRegressor) 0.3080163610659594
- Accuracy (GradientBoostingRegressor): **0.7493922798036373**

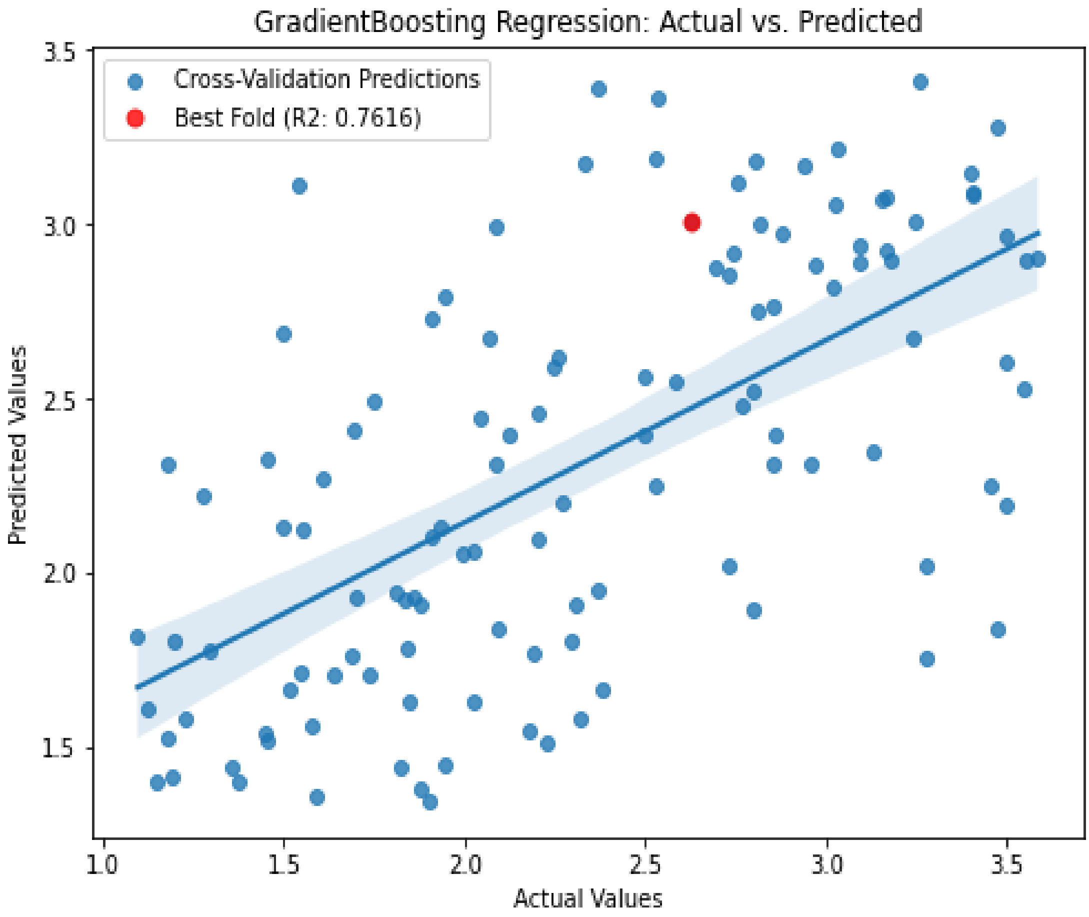


- The gradient boosting regressor achieved the accuracy of 74% let's apply the cross validation to xgb.

- **GradientBoostingRegressor with CV:**

- Cross-Validation Scores (R2): [0.15775274, 0.16740939, 0.34975788, 0.26392372 ,0.20974946 0.13282827, **0.76159505**]

- Mean Cross-Validation Score: 0.2319310845395406
- The accuracy has been improved from **74%** to **76%** when we applying the cross validation.



- # CONCLUSION:

- The aim is to achieve the 80% of accuracy using machine learning models while the dependent variable is experimental weight with the various independent variables like environmental factors. We have achieved the accuracy of 91% with linear regressor and 80% accuracy with xgb regressor.