RUTGERS
THE STATE UNIVERSITY
OF NEW JERSEY

PROJECT REPORT

Extremely Randomised Trees

Prepared by

Vishnuram Jatin Bangaru(vb440)

Ankeeta Priyam(ap2213)

Noopur Singh(ns1482)

Table of Contents

Introduction

Randomization is a well-known technique that can help to reduce the overall variance of tree-based algorithms. To leverage this fact, Extremely Randomized Trees, also known as Extra Trees, utilize a strong level of randomization for both attribute selection and cut points. One of the primary benefits of this approach is that it can improve the stability and robustness of the model. However, this strong randomization can also result in reduced computational efficiency compared to other models.

The selected paper evaluates the performance of Extra Trees on both classification and regression datasets and compares it with other tree-based models such as DT, Random Forests, Tree Bagging. The results show that Extra Trees perform well in both classification and regression, with ET outperforming other models in all classification datasets. The paper also discusses how the three parameters 'K', nmin, M vary for ExtraTrees. The paper uses three types of ET across the comparisons where K is default, chosen by Cross Validation.

The project considers the findings of the paper and conducts a comparison of various tree and boosting models for both regression and classification on small and large datasets. The project also explores the Bias/Variance tradeoff between these algorithms. The project tries to find how Extra Trees fairs against outliers and how robust it is to outliers. Comparing it with Boosting Algorithms is being done which has not been covered in the Paper. Lastly, the project explores how Extra Trees perform when trying to classify Image Datasets.

What is Extra Trees?

Extra Trees (short for Extremely Randomized Trees) is a machine learning algorithm used for classification, regression, and anomaly detection tasks. It is an ensemble learning method that combines multiple decision trees to make predictions.

The Extra Trees algorithm works by creating a large number of decision trees, each trained on a randomly sampled subset of the training data and a randomly selected subset of features. Unlike Random Forests, which use a majority voting scheme to make predictions, Extra Trees use a weighted average of the predictions of each individual tree to make its final prediction. This makes the algorithm less sensitive to noise and overfitting, and can lead to improved performance compared to other ensemble methods.

It is important to note that Extra Trees differ from other tree ensemble methods in that they randomly select cut points and use the entire learning sample, rather than a bootstrap replica, to grow the trees. The cut point randomness provides a reduce in variation while not using a bootstrap replica helps in reducing bias.

Split_a_node(S)
Input: the local learning subset S corresponding to the node we want to split
Output: a split $[a < a_c]$ or nothing
– If **Stop_split**(S) is TRUE then return nothing.
– Otherwise select K attributes $\{a_1, \ldots, a_K\}$ among all non constant (in S) candidate attributes;
– Draw K splits $\{s_1, \ldots, s_K\}$, where $s_i = \textbf{Pick\_a\_random\_split}(S, a_i), \forall i = 1, \ldots, K$;
– Return a split s_* such that $\text{Score}(s_*, S) = \max_{i=1,\ldots,K} \text{Score}(s_i, S)$.

Pick_a_random_split(S,a)
Inputs: a subset S and an attribute a
Output: a split
– Let $a^S_{\max}$ and $a^S_{\min}$ denote the maximal and minimal value of a in S;
– Draw a random cut-point a_c uniformly in $[a^S_{\min}, a^S_{\max}]$;
– Return the split $[a < a_c]$.

Stop_split(S)
Input: a subset S
Output: a boolean
– If $|S| < n_{\min}$, then return TRUE;
– If all attributes are constant in S, then return TRUE;
– If the output is constant in S, then return TRUE;
– Otherwise, return FALSE.

Extra Trees has several advantages over other tree-based algorithms, including faster training times, lower memory usage, and higher accuracy due to its use of randomization. However, it can be more difficult to interpret than other algorithms, and tuning the hyperparameters can be challenging. The ExtraTrees performs very good compared to the other models however it suffers from a higher bias in classification models.

Machine Learning Models Used

1. Random Forest

It is a type of ensemble learning method used in predictive modeling, where multiple decision trees are trained and combined to make predictions. It works by creating a large number of decision trees that randomly sample the training data and a subset of features. Each tree makes its own prediction, and the final prediction is based on the aggregated results from all the trees. This method is known for its ability to handle high-dimensional data, reduce overfitting, and produce accurate results.

2. Decision Trees

Decision trees are a type of machine learning algorithm that build a hierarchical tree structure to make decisions based on the features of the data. Each node in the tree represents a decision based on a feature, and each branch represents the possible outcomes of that decision. They are commonly used in both classification and regression tasks, and are known for their simplicity, interpretability, and ability to handle nonlinear relationships in data.

3. Bagging

Short for Bootstrap Aggregating, it is a machine learning technique used to improve the stability and accuracy of models by generating multiple subsets of the training data and training a separate model on each subset. The final prediction is then made by aggregating the predictions of all the models. Bagging is commonly used with decision trees, but can be applied to other models as well, and is known for reducing overfitting and increasing model performance.

4. AdaBoost

Short for Adaptive Boosting, it is a machine learning algorithm that combines multiple weak classifiers to create a strong classifier. It works by iteratively training weak classifiers on the misclassified data points, and increasing the weights of these points to focus on the hard-to-classify examples. The final prediction is made by aggregating the predictions of all the weak classifiers weighted by their performance. AdaBoost is commonly used in binary classification problems and is known for its high accuracy and robustness.

5. XGBoost

Short for Extreme Gradient Boosting, it is a machine learning algorithm used for regression, classification, and ranking tasks. It is a variant of the gradient boosting method that uses a more regularized model and a novel tree learning algorithm to improve performance and reduce overfitting. XGBoost is known for its speed, scalability, and accuracy, and is commonly used in machine learning competitions and industry applications.

6. CatBoost

CatBoost is a gradient boosting framework that is designed to handle categorical features in a more efficient and accurate way than other gradient boosting frameworks. CatBoost uses a combination of techniques such as ordered boosting, gradient-based one-hot encoding, and feature combinations to effectively handle categorical features.

Data Preprocessing Techniques

1. Cross Validation Accuracy Score

Cross validation accuracy score is the average accuracy of a machine learning model obtained through cross-validation, representing its generalization performance on unseen data.

2. Cross Validation Negative Mean Squared Error Score

Cross validation negative mean squared error score is the average negative mean squared error of a machine learning model obtained through cross-validation, providing an assessment of its predictive accuracy for regression tasks.

3. Missing Values - Iterative imputing

Iterative imputing involves an iterative procedure that fills in missing values within a dataset by leveraging observed values and data relationships, leading to improved data completeness and accuracy for subsequent analysis and tasks that rely on complete data.

4. Outliers – IQR Technique

- The IQR (Interquartile Range) technique is a statistical method used to detect and handle outliers in a dataset.
- It involves calculating the IQR, which is the range between the 75th percentile (Q3) and the 25th percentile (Q1) of the data.
- Outliers are identified as data points that fall below Q1 - 1.5 * IQR or above Q3 + 1.5 * IQR, and can be treated by either removing them or applying appropriate transformations to mitigate their impact on the analysis.

Datasets

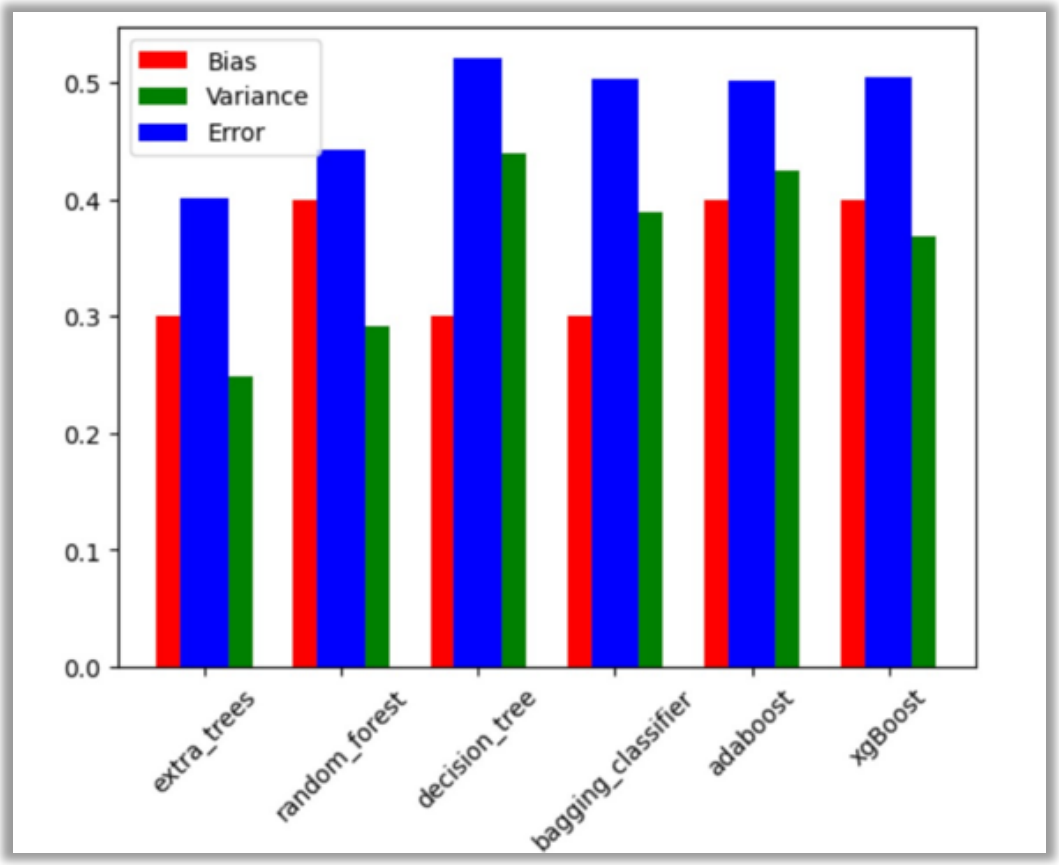
We have picked various classification and regression datasets from the UCI repository. Few datasets are small datasets like Lung, Protein while few are large Portuguese bank datasets. Other Regression datasets include Regression datasets like wine quality, data scientist salaries, auto prices. Images datasets have been picked from Kaggle which include Brain Tumor Images, Satellite Images and Fashion Images.

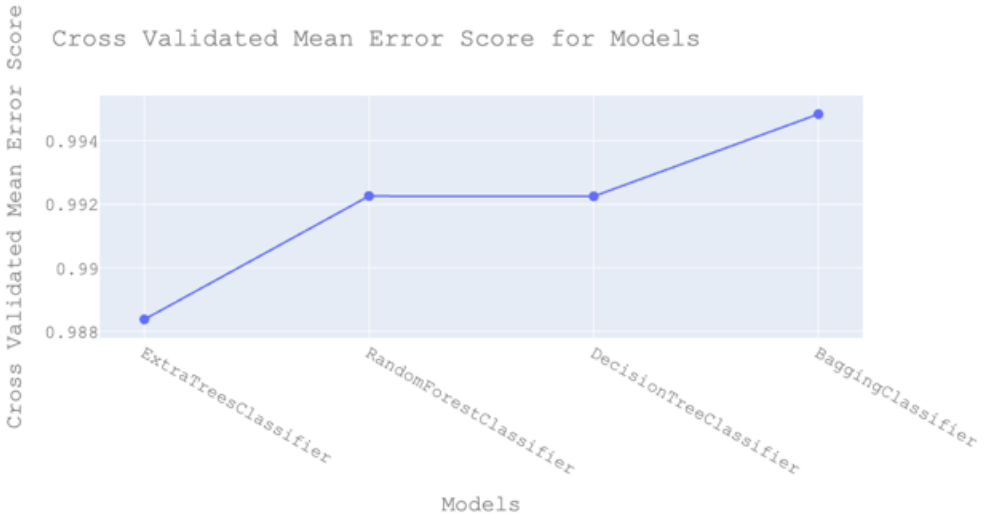
Comparison of Extra Trees Classifier in Small vs Large Datasets

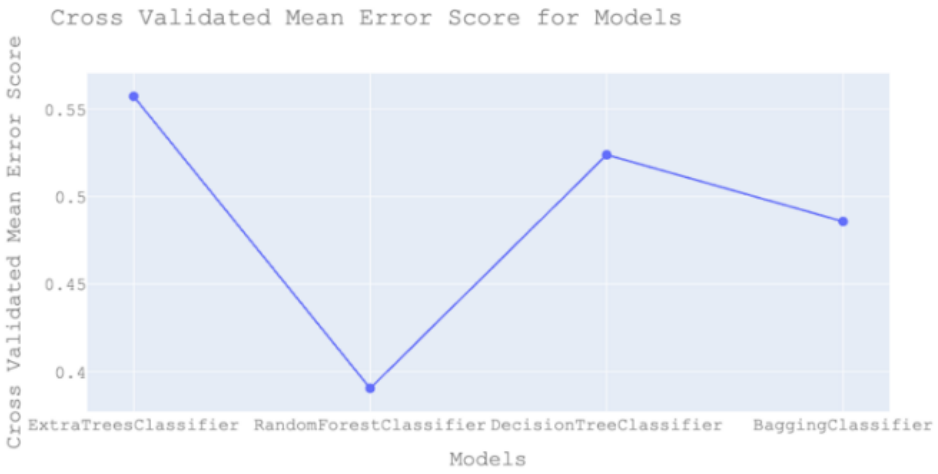
The Project has taken Extra Trees and the other Randomized algorithm to be trained on small and large datasets to see how it performs. We have used KFold Cross Validation technique to train it on datasets. The scoring methodology adopted would be the Cross Validation Mean Accuracy Score. The datasets chosen for small datasets were in the range of 10-100 rows while the large range from 1000 to 10000s

We have compared how it performed against other Tree Methods.

For Small Datasets, On an average we see that Extra Trees performance is comparatively good compared to other Algorithms. We also see that the bias and variance of Extra Trees is better compared to other models even the Random Forests.

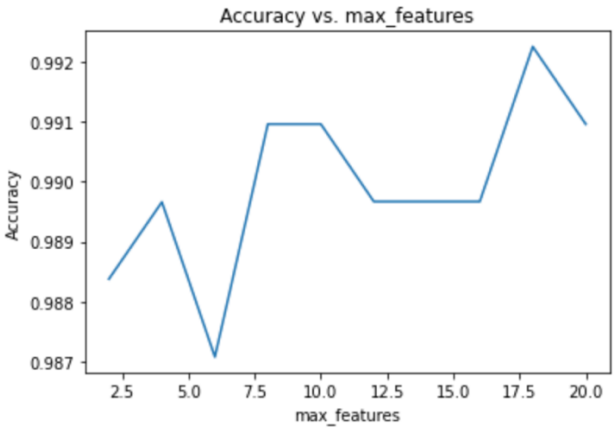


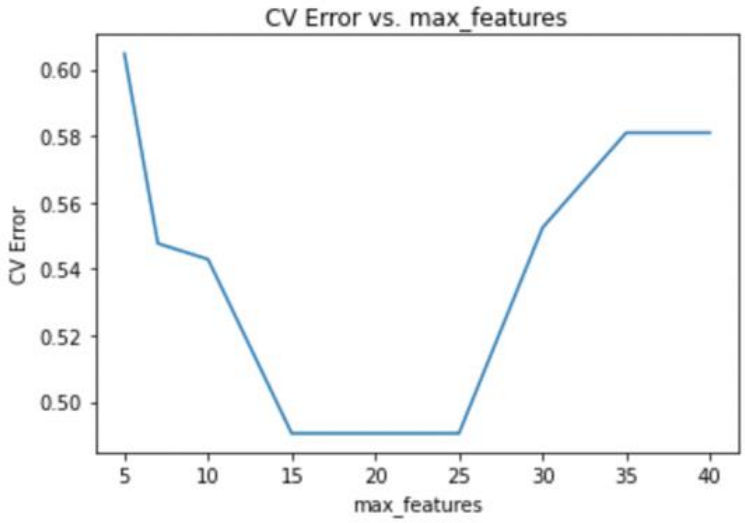




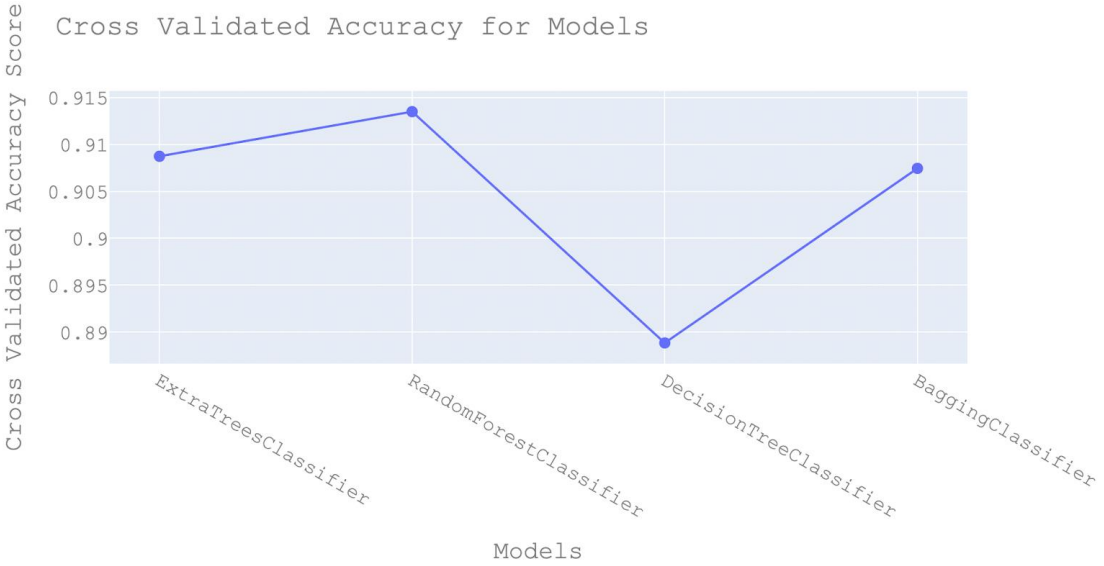
Now varying the K parameter:

The Project also checks how the performance varies on changing the value K. We can clearly see that there is no proper trend with varying K in small datasets, this is mostly due to randomization. This suggests that when the Dataset is small, we cannot pick a particular K value.

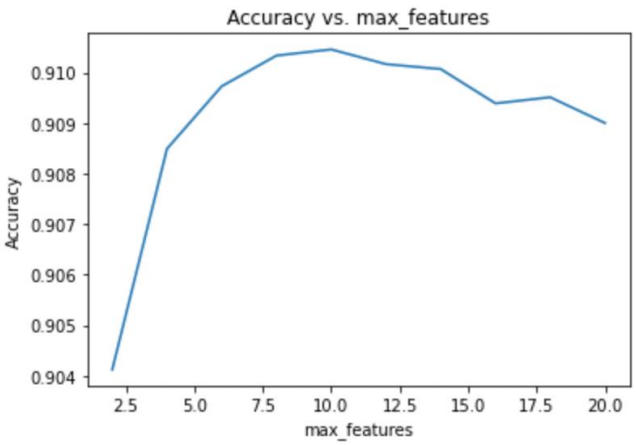


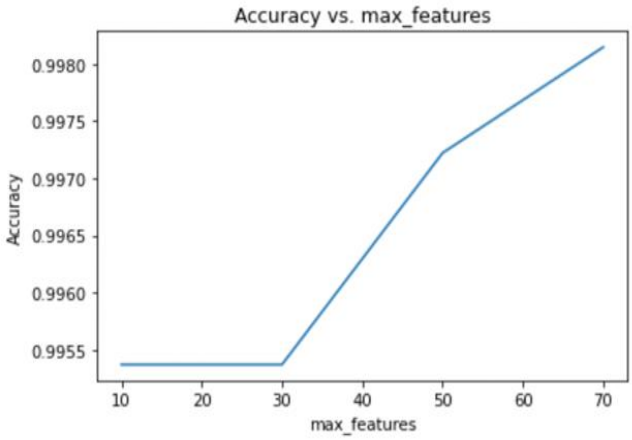


For Large Datasets too, Extra Trees performs comparatively well against the other classifiers.



Although one thing to note for is that we have a particular trend in K when varying the K parameter.

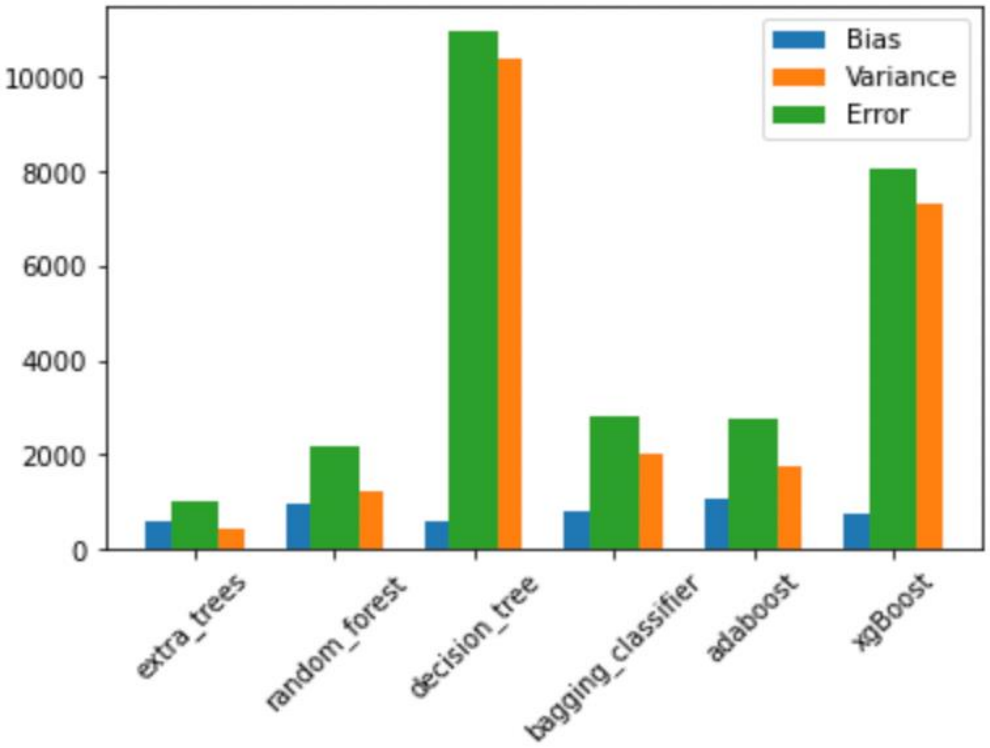




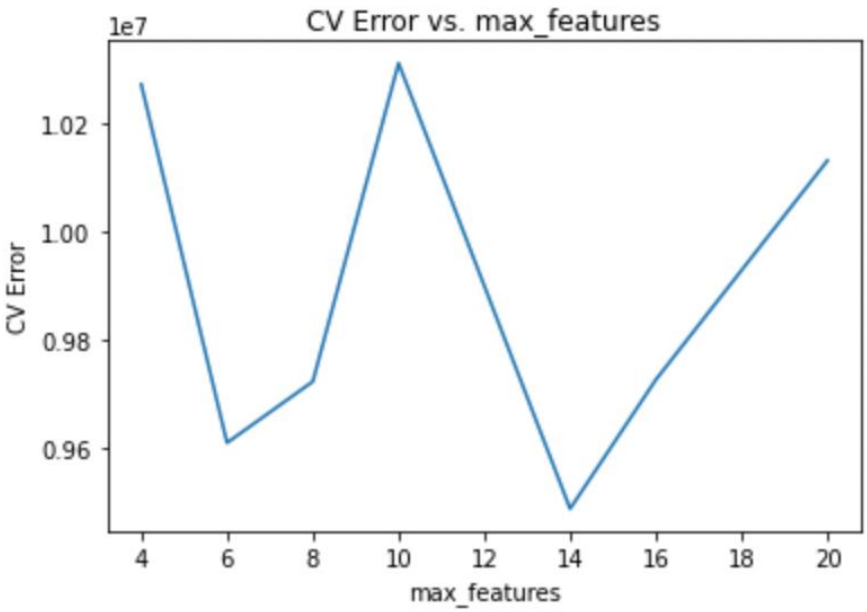
Comparison of Extra Trees Regressor in Small vs Large Datasets

Like classification, The Project tested the performance of Extra Trees Regressor Model against the other Ensemble Regressor Models in Large and Small Datasets.

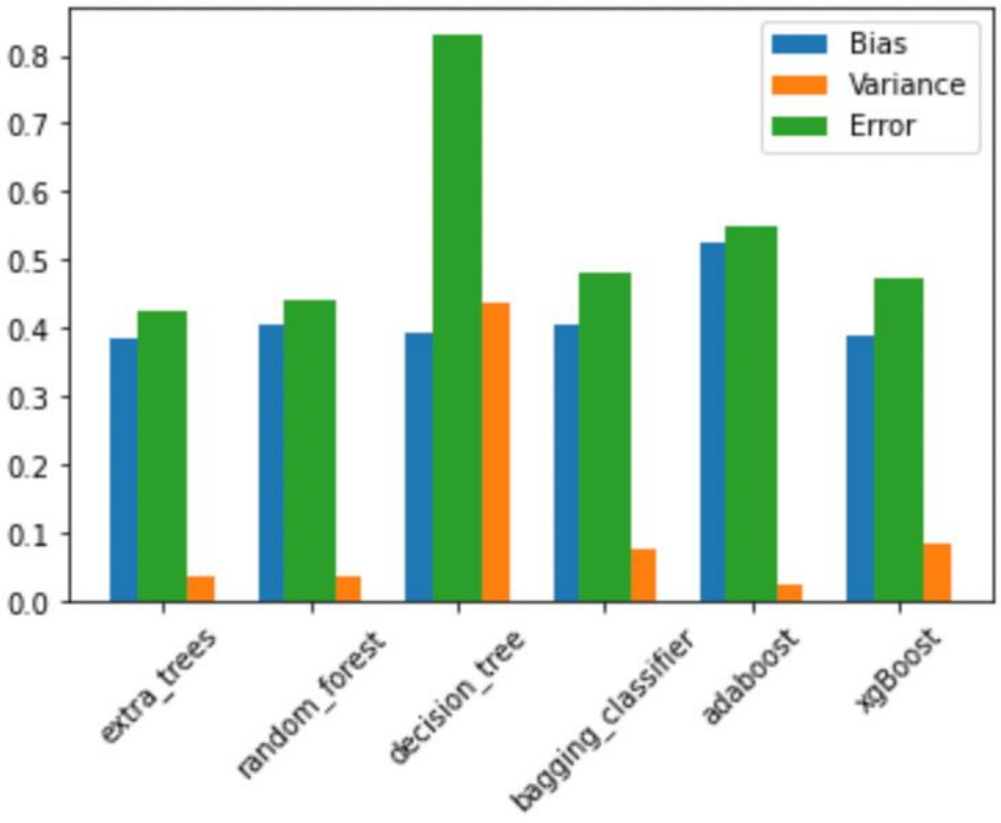
In Small Datasets, the Extra Trees performed really well compared to the other models. We can see the lowest balance of bias variance for Extra Trees compared to Random Forests and other algorithms. This proves the fact from the Paper summary that Extra Trees does perform well in regression than classification probably because it is more robust to noise and absence of decision boundary like classification which can fully exploit the randomness which Extra Trees provide.

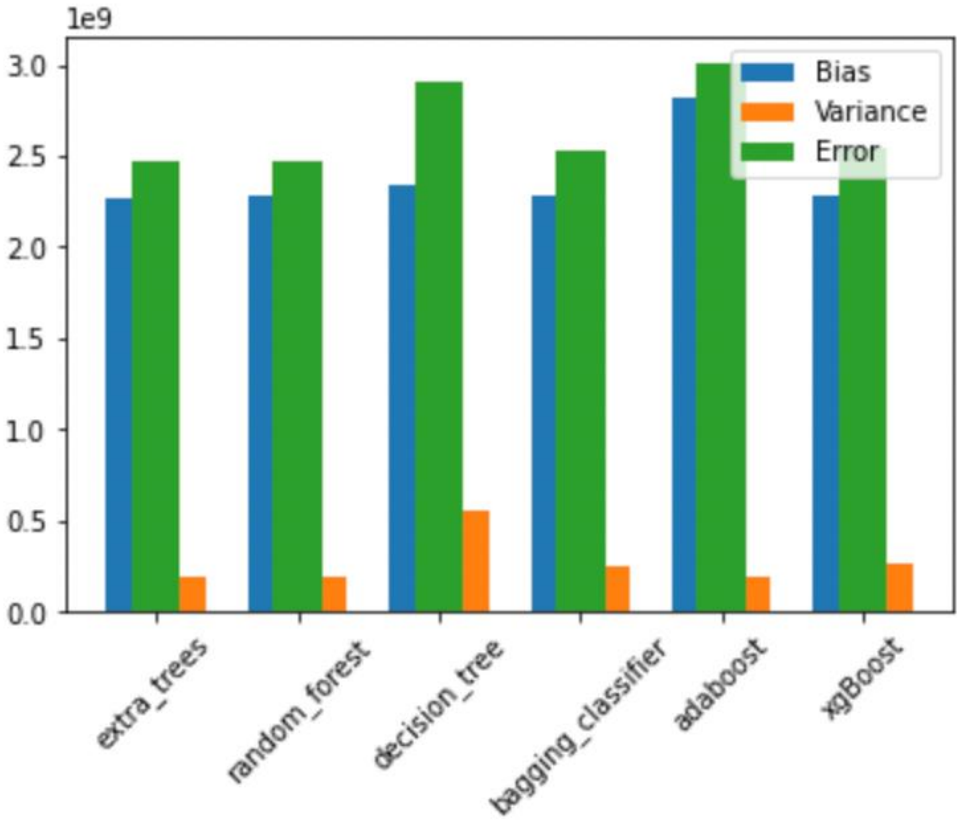


On varying K, we can see the jumpiness in the curve which suggests we probably can't pick a right value of K when regressing on small datasets. This again will be due to the small data and randomization which Extra Trees provide which leads to sensitivity in varying K.

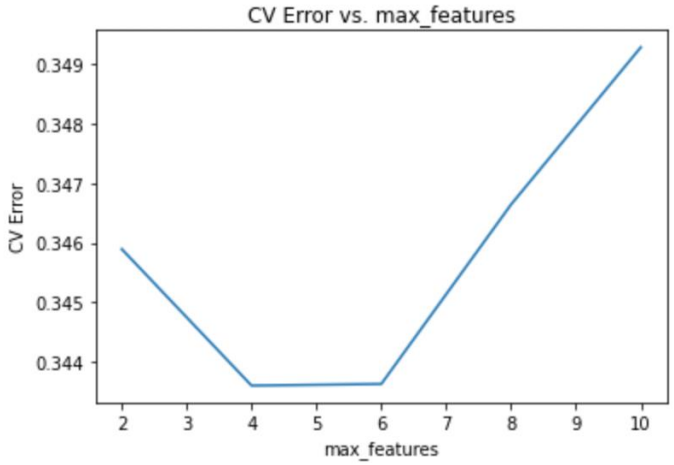


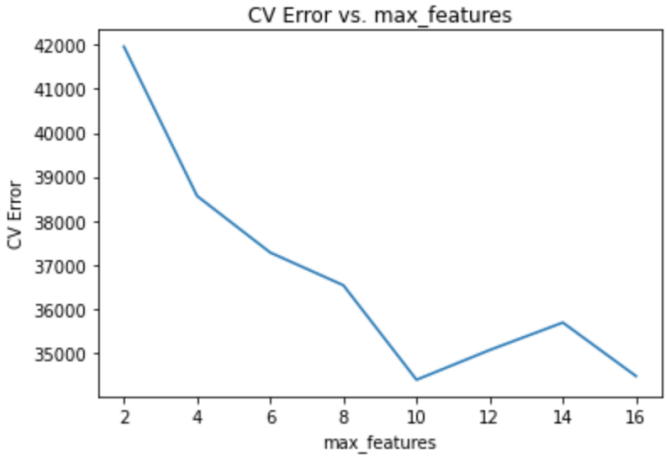
Now for large datasets, we observe that Variance is relatively very low than the bias. However, we notice that for large dataset the performance with Extra Trees and other algorithm although good is not that noticeable.





However, on varying K, we can see a clear pattern however decreasing or increasing depending on the type of dataset.





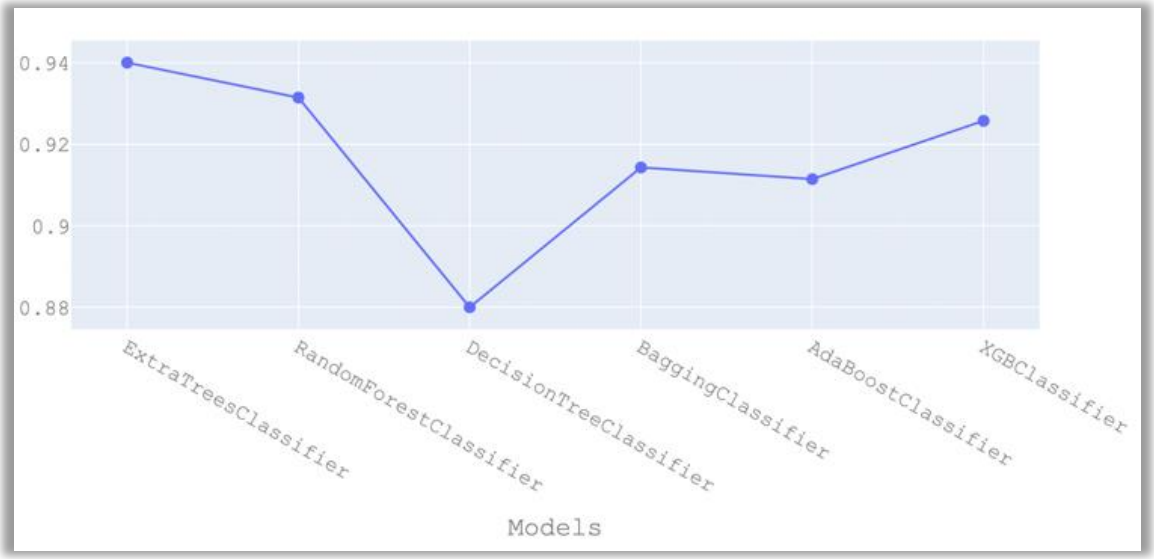
Robustness with outliers

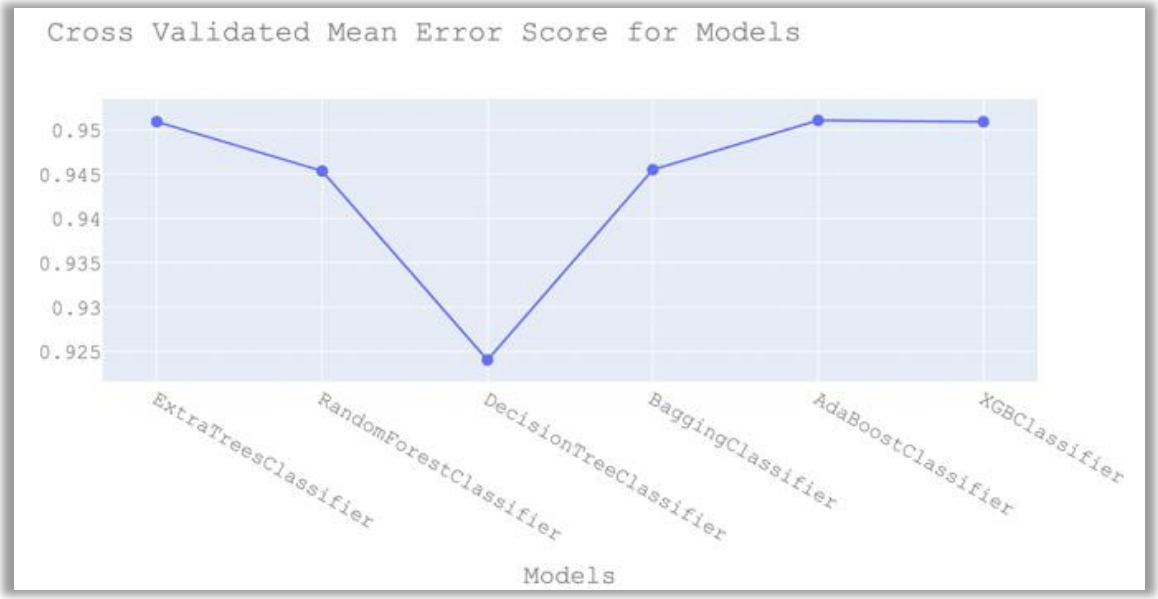
Outliers, which are data points that deviate significantly from the rest of the dataset, can have a significant impact on the performance of the algorithms. The purpose of this section is to investigate the impact of outliers on extra trees performance in comparison to other algorithms.

Random datasets with lot of outliers for classification and regression are used to compare the performance of extra trees with respect to other regression and classification algorithms.

- ## Classification:

Extra Trees along with other ensemble learning algorithms, were used to classify datasets with outliers. Later, the outliers were removed and ran the Extra Trees algorithm again to build new models. Compared the mean accuracy scores of these models before and after the removal of outliers to determine the impact of outliers on the performance of the algorithms.

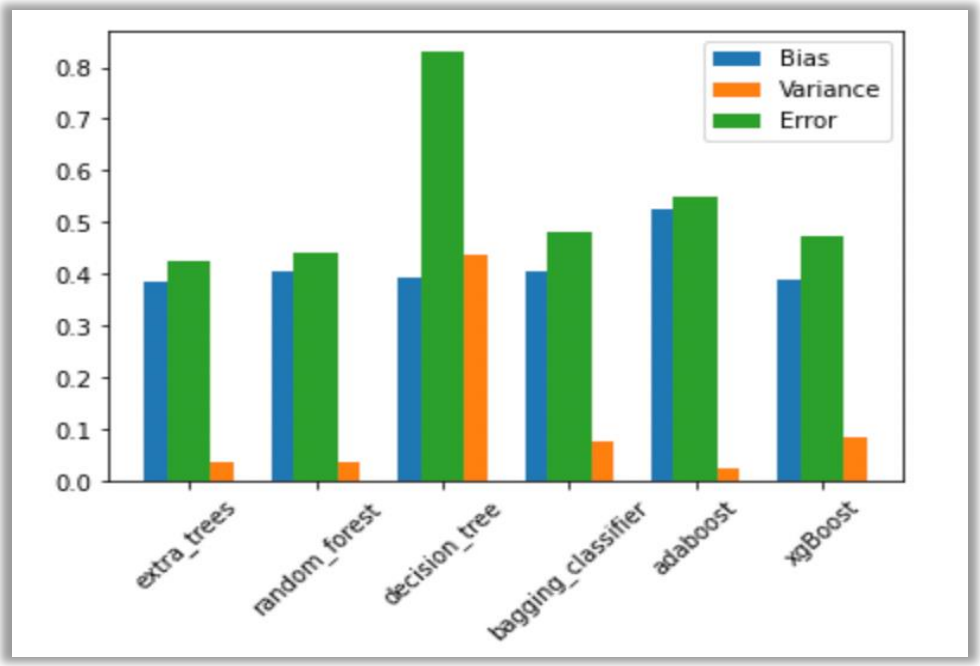


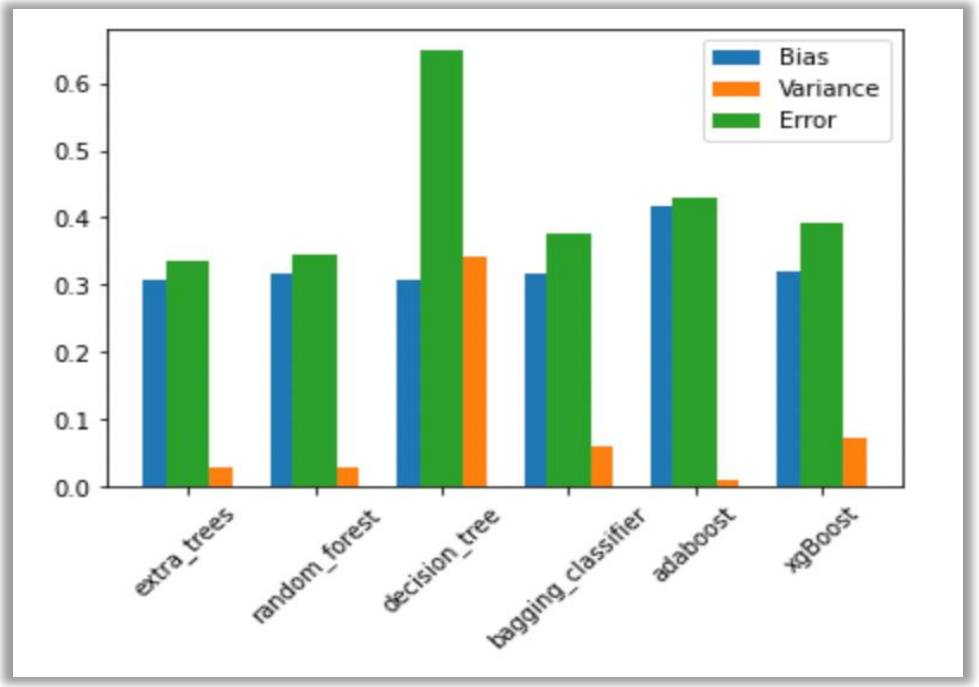


The plot presented displays the performance of various classifiers both before and after the removal of outliers. The mean accuracy score of the Extra Trees classifier shows minimal change before and after outlier removal, whereas other classifiers such as Random Forest, Decision Tree, Bagging, AdaBoost and XGBoost display a slight increase in mean accuracy score. This indicates that the performance of these classifiers improved after removing the outliers, while Extra Trees remained largely unaffected.

- ## Regression:

The comparison as we did in classification, the same thing is followed for regression datasets as well. The models were built using different algorithms like Extra tree, Random Forest, Decision Tree, Bagging, AdaBoost and XGBoost on datasets before and after removal of outliers and their negative RMSE were compared.





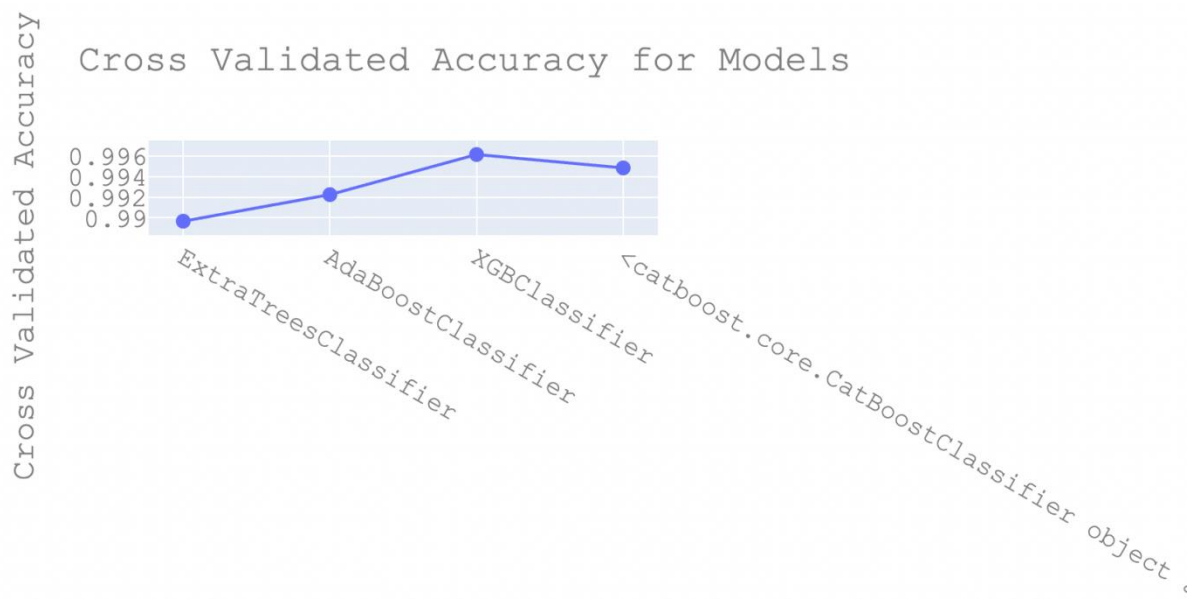
The graph illustrates the performance of different regression models before and after eliminating outliers. The Extra Trees classifier's negative RMSE shows only a slight variation before and after removing outliers, while other models display a slight change in negative RMSE. This suggests that removing outliers led to an improvement in these models' performance, while Extra Trees remained relatively stable.

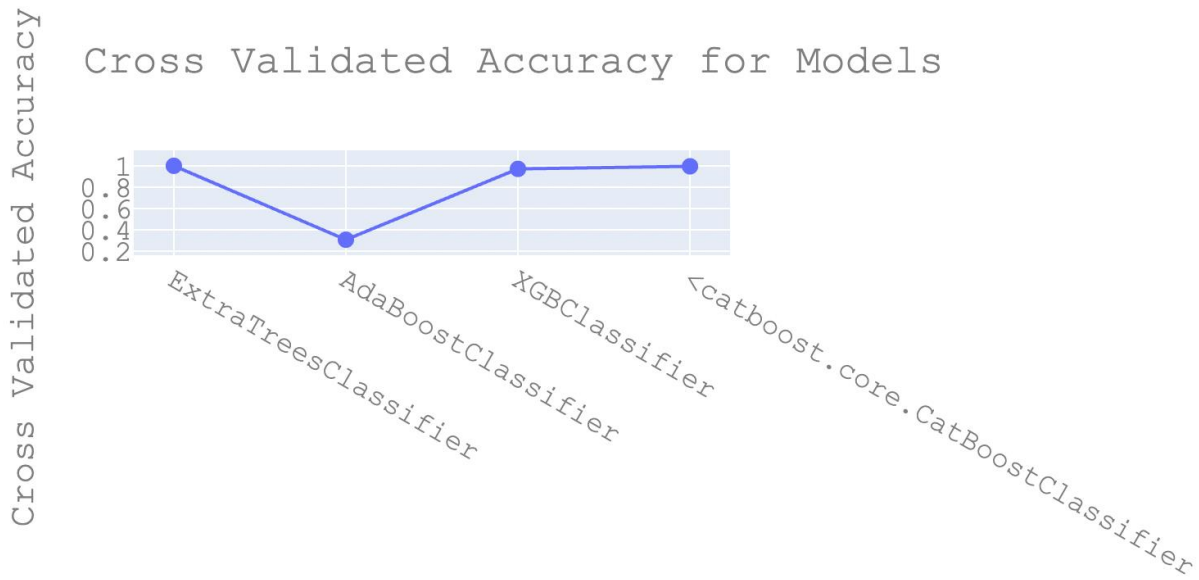
Based on our analysis, it can be inferred that Extra Trees are more resilient to the presence of outliers than other algorithms in both regression and classification problems.

Comparison with Boosting Algorithms

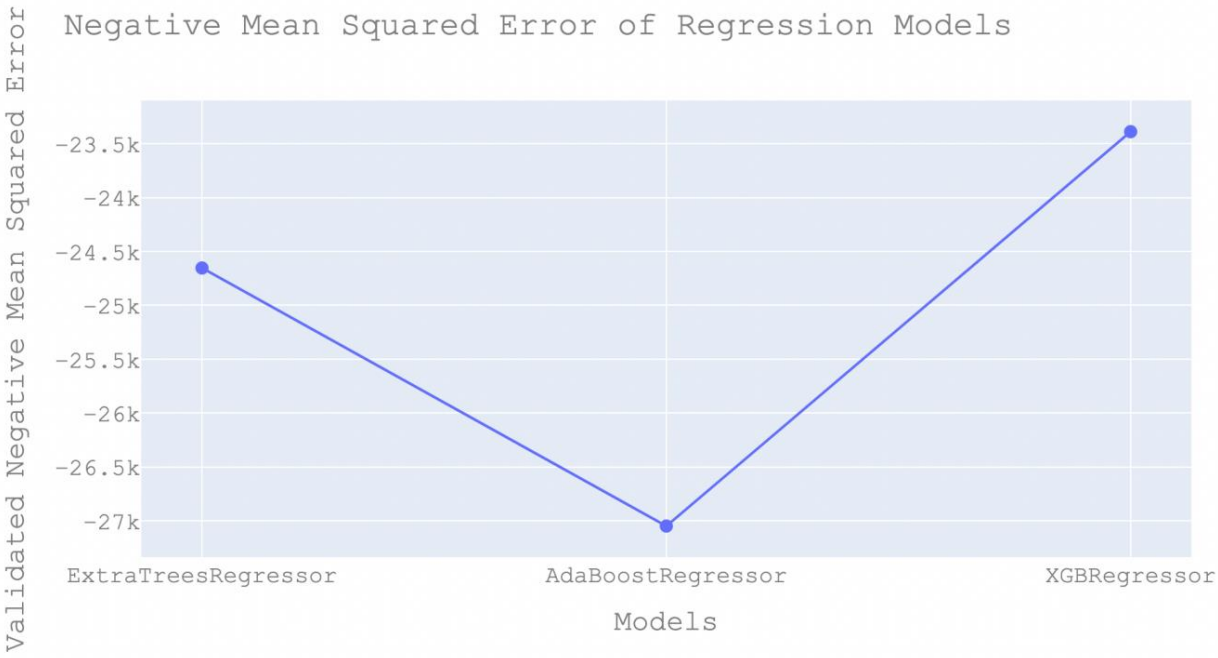
When comparing Extra Trees with boosting algorithms, several important distinctions emerge. Extra Trees exhibit a notable advantage in terms of training speed, as they construct decision trees independently. Conversely, boosting algorithms entail a slower training process as weak learners are sequentially built. However, boosting algorithms frequently outperform Extra Trees, particularly in intricate datasets and demanding classification or regression scenarios. Boosting algorithms excel in their ability to adapt to data and progressively enhance predictions by emphasizing challenging instances. In contrast, Extra Trees rely primarily on randomization for diversification. Ultimately, the choice between the two techniques depends on specific requirements, dataset size, desired accuracy, and the trade-off between training speed and predictive performance. As mentioned above, The Project compared Extra Trees with 4 Boosting Algorithms. The Project uses CatBoost when there were a notable amount of categorical features in the dataset. We all know the advent of XGBoost nowadays so it was interesting to look at how ExtraTrees Perform against XGBoost.

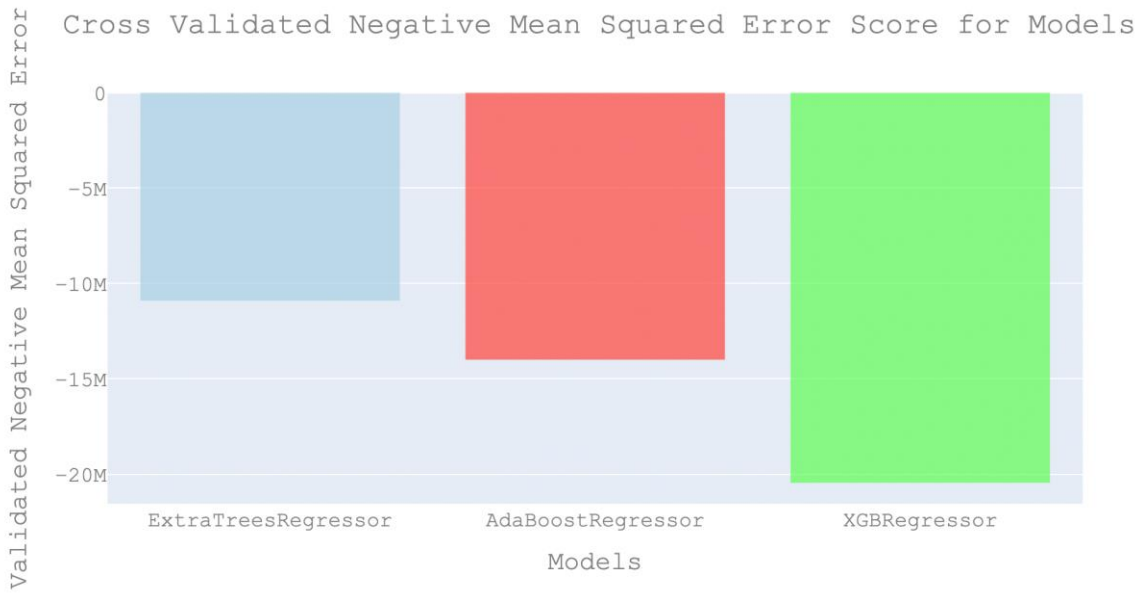
In Classification, We can see on an average Extra Trees perform comparatively good w.r.t XGBoost and other Boosting Algorithms. The accuracy score difference in very slow. Infact in few datasets, we see that Extra Trees performs better than other Boosting Algorithms.





In Regression, We see that ExtraTrees performs comparatively poorly compared to other Boosting Algorithms and the difference is quite large. The poor performance of Extra Trees in regression compared to boosting algorithms can be attributed to the lack of sequential learning, absence of weighted training, and limited ensemble diversity. These factors hinder Extra Trees' ability to capture complex patterns, handle outliers effectively, and adapt to the data during training.




Cross Validated Negative Mean Squared Error Score for Models
Validated Negative Mean Squared Error
0
-5M
-10M
-15M
-20M
ExtraTreesRegressor
AdaBoostRegressor
XGBRegressor
Models

Performance on Image Dataset

The Paper mentions that ExtraTrees would perform well with classifying Images as well. Therefore The Project tries to compare Extra Trees with other Ensemble Algorithms and see how they fair against Image Datasets.

For Image Classification, The Project has taken three datasets Brain Tumor, Satellite Images, Facial Expressions. The Project however uses a Statistical Technique called GrayCoMatrix.

Gray-level co-occurrence matrix (GLCM) is a statistical method used to describe the spatial relationship between pixels in an image. It provides information about the frequency of occurrence of pairs of pixel values at a specified distance and direction in an image. GLCM is a widely used method in image processing and computer vision applications, particularly for feature extraction and texture analysis.

The Project takes the images and converts it into features using 4 Angles, 3 Distances and the following 5 properties to give a total of 60 features:

Contrast: measures the local variations in the gray-level co-occurrence matrix. Higher contrast values indicate a greater difference between the co-occurring gray-level values and therefore greater textural heterogeneity.

Dissimilarity: measures the average difference in the gray-level values between adjacent pixels. Higher dissimilarity values indicate greater textural heterogeneity.

Homogeneity: measures the closeness of the distribution of elements in the GLCM to the GLCM diagonal. Higher homogeneity values indicate a more homogeneous texture.

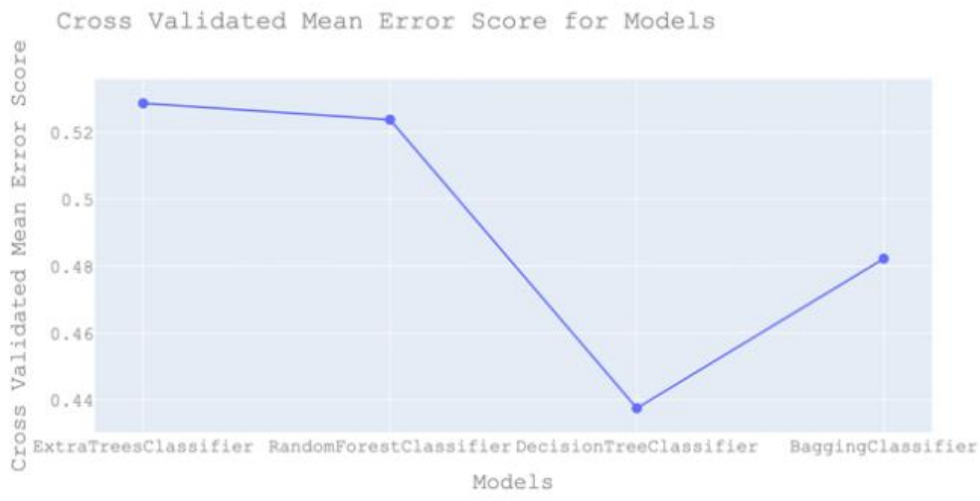
Energy: measures the sum of squared elements in the GLCM. Higher energy values indicate a more homogeneous texture.

Correlation: measures the degree of linear dependency between the gray-level values in the image. Higher correlation values indicate a more linear relationship between the gray-level values.

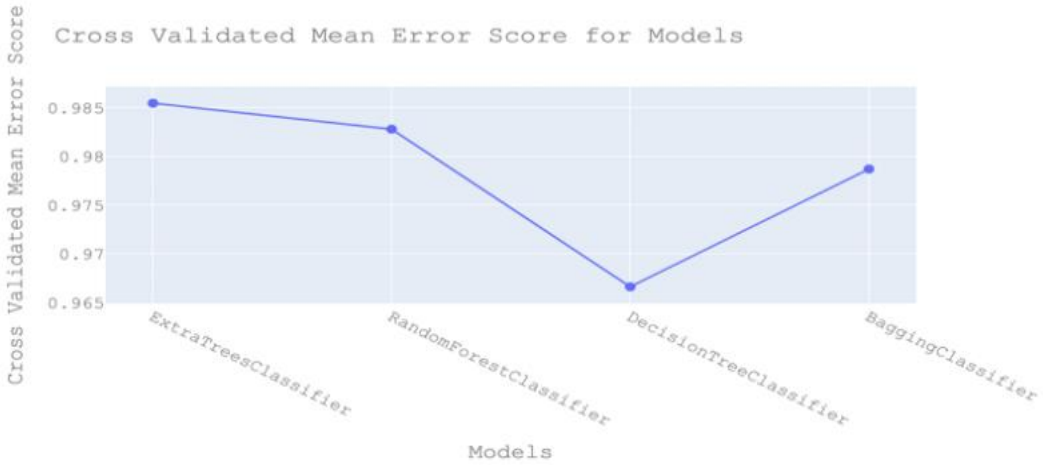
Let’s see how they fair against the images

- Brain Tumor

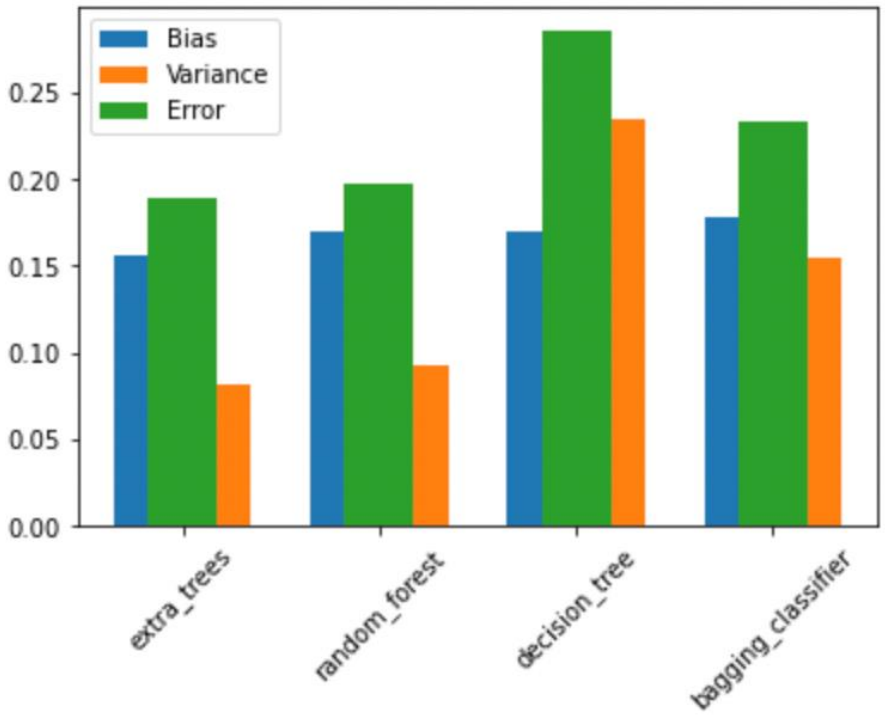
- Facial Images



- Satellite Images



As we can see, Extra have outperformed other models in all the three images datasets. The corresponding test scores margin w.r.t Random Forests is low although Extra Trees have a higher score. Although there is low performance in satellite images, still extra trees have a higher score than Random Forests and other Bagging Classifiers. We can also see the bias variance plots and ExtraTrees have much better bias variance tradeoff compared to other ensemble models, including the Random Forests. This suggests a unique approach for utilizing ExtraTrees compared to DeepNets for Image Classification.



Discussion

In the project, Extra Trees were examined and compared with other algorithms in both classification and regression tasks, using datasets of varying sizes. The robustness of Extra Trees in the presence of outliers was also evaluated. Furthermore, Extra Trees were contrasted with boosting algorithms. The project also extended the ability to test Extra Trees fairness against Image Classification Datasets using a Statistical approach.

The algorithm demonstrated competitive results across both small and large datasets, showcasing its versatility. Additionally, Extra Trees proved to be robust in the presence of outliers, making it a suitable choice for data contaminated with noisy or aberrant points. Comparisons with boosting algorithms, it achieved comparable performance in classification datasets but did not work well in case of large regression datasets. This could probably be influenced by the nature of datasets we had chosen. Coming to image classification tasks revealed that Extra Trees achieved very appreciable performance, highlighting its effectiveness in handling complex image datasets.

Overall, the project findings suggest that Extra Trees are a robust and versatile algorithm that can be effectively employed in various classification and regression tasks, including scenarios with outliers. Instead of using DeepNets for image classification, the project successfully achieved comparable performance by employing the Graycorrelation Statistical concept of feature extraction in conjunction with ExtraTrees. The results underscore the potential of ExtraTrees combined with innovative feature extraction methods as a viable alternative to DeepNets for image classification tasks.

References

1. Extremely Randomized Trees by Pierre Geurts · Damien Ernst · Louis Wehenkel
2. UCI Repository
3. Kaggle
4. https://aws.amazon.com/what-is/boosting/
5. https://www.section.io/engineering-education/boosting-algorithms-python/
6. https://towardsdatascience.com/5-outlier-detection-methods-that-every-data-enthusiast-must-know-f917bf439210
7. Linjun Zhang Data Mining Notes
8. https://github.com/alfianhid/Feature-Extraction-Gray-Level-Co-occurrence-Matrix-GLCM-with-Python
9. ChatGPT
10. https://towardsdatascience.com/an-intuitive-explanation-of-random-forest-and-extra-trees-classifiers-8507ac21d54b