FOREST FIRES : BURNED AREA PRIDICTION MODELS

Siddhartha Chilukuri

MS Data Science

Dept of Mathematics and Statistics

TAMUCC.

1. INTRODUCTION

Forest fires are among the most destructive natural events, with devastating impacts on the environment, wildlife, human life, and infrastructure. In recent years, due to climate change and human activities, the frequency and intensity of forest fires have increased globally. This alarming trend underscores the urgent need for effective predictive tools to mitigate fire risks and consequences.

The burned area, or the extent of forested land destroyed by fire, is a critical metric for understanding fire severity and planning response strategies. Accurately predicting the burned area can help:

- Allocate firefighting resources efficiently.
- Protect sensitive ecosystems and biodiversity.
- Minimize economic losses associated with fire damage.
- Improve preparedness for future fire events.

While traditional statistical models have been used in the past to estimate burned areas, they often fall short when capturing the complex relationships between the variables that influence fire spread and intensity. These variables include weather conditions, fire behavior indices, and spatial-temporal factors, which are often non-linear and interdependent.

1.1 PROBLEM STATEMENT

The goal of this project is to predict the burned area of forest fires using machine learning techniques. Specifically, we address the following questions:

1. Which features (e.g., weather, fire indices, or spatial-temporal factors) contribute the most to predicting the burned area?
2. How do machine learning models compare in their ability to predict the burned area of forest fires?
3. Can advanced non-linear models like Gradient Boosting Machines and Random Forests outperform traditional linear regression techniques for this task?

By exploring these questions, this project aims to identify the most effective modeling approach for predicting burned areas. This can assist fire management agencies in decision-making, resource planning, and reducing the ecological and economic impacts of forest fires.

1.2 SIGNIFICANCE OF THE STUDY

Given the increasing global attention on forest fire mitigation and prevention, this project is timely and relevant. By leveraging machine learning, we aim to improve predictive accuracy, providing actionable insights that can support proactive fire management strategies. Additionally, this study demonstrates the potential of applying advanced data-driven methods to real-world environmental challenges, encouraging further research in this field.

2. MOTIVATION

Forest fires are increasingly recognized as a global crisis, with significant repercussions for human life, ecosystems, and the economy. The growing intensity and frequency of these fires, fueled by climate change and land-use changes, highlight the pressing need for effective prevention and mitigation strategies.

2.1 WHY IS PREDICTING BURNED AREA IMPORTANT?

Predicting the burned area is essential for several reasons:

1. **Emergency Preparedness and Response**:

Accurate predictions allow firefighting teams to estimate the potential severity of a fire and allocate resources more effectively. Regions with large potential burned areas can be prioritized for monitoring and evacuation plans.

2. **Ecological Conservation**:

Forests serve as critical carbon sinks, help regulate climate, and support diverse ecosystems. Predicting and mitigating fire damage can help protect these invaluable resources. Post-fire recovery efforts can be better planned with insights into the extent of damage.

3. **Economic Impact Mitigation**:

Fires can destroy infrastructure, agricultural lands, and livelihoods. Predicting burned areas enables more precise cost estimates and resource allocation to minimize economic losses.

4. **Adaptation to Climate Change**:

With global warming leading to hotter and drier conditions, predictive tools can help regions adapt to changing fire patterns and prepare for more intense fire seasons.

2.2 SCIENTIFIC MOTIVATION

Forest fires are complex phenomena influenced by multiple interrelated factors such as weather, fuel conditions, and topography. The burned area, a critical outcome of fire behavior, is shaped by both local (e.g., humidity, wind) and global (e.g., seasonal changes) factors. Traditional models often fall short in addressing this complexity, motivating the application of machine learning (ML) methods, which excel at capturing intricate, non-linear relationships.

2.3 PROJECT GOALS

The primary objective of this study is to evaluate the potential of machine learning models in predicting the burned area of forest fires. This involves:

1. Identifying the features that most strongly influence the burned area.
2. Comparing linear models (e.g., Linear Regression, Ridge Regression) and non-linear models (e.g., Gradient Boosting, Random Forest) to assess their predictive performance.
3. Developing a robust and scalable solution that can be adapted for real-world fire management applications.

3. DATASET OVERVIEW

The dataset contains information about forest fires in a specific region, including geographical data, weather conditions, and the resulting burned area. It has 517 observations and 13 columns. The data types include integers, floats, and categorical variables.

3.1 ATTRIBUTES IN THE DATASET

1. **Geographical Attributes**:
   - X (integer): The x-axis spatial coordinate within the forest map grid (1 to 9).
   - Y (integer): The y-axis spatial coordinate within the forest map grid (2 to 9).
2. **Temporal Attributes**:
   - month (string): The month of the year (e.g., "jan", "feb").
   - day (string): The day of the week (e.g., "mon", "tue").
3. **Fire Weather Index (FWI) System Variables**:
   - **FFMC (Fine Fuel Moisture Code)** (float): An index measuring the moisture content of surface litter. Higher values mean drier conditions.
   - **DMC (Duff Moisture Code)** (float): Represents moisture levels in loosely compacted organic layers.
   - **DC (Drought Code)** (float): Reflects long-term moisture levels, indicating drought severity.
   - **ISI (Initial Spread Index)** (float): A predictor of the initial fire spread rate.
4. **Weather Conditions**:
   - temp (float): Temperature in degrees Celsius.
   - RH (integer): Relative humidity as a percentage.
   - wind (float): Wind speed in km/h.
   - rain (float): Rainfall in mm/m².
5. **Target Variable**:
   - area (float): The total burned area of the forest in hectares. A value of 0 indicates no fire damage.

Below is a overview of the dataset.

	X	Y	month	day	FFMC	DMC	DC	ISI	temp	RH	wind	rain	area
	<int>	<int>	<chr>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<int>	<dbl>	<dbl>	<dbl>
1	7	5	mar	fri	86.2	26.2	94.3	5.1	8.2	51	6.7	0.0	0
2	7	4	oct	tue	90.6	35.4	669.1	6.7	18.0	33	0.9	0.0	0
3	7	4	oct	sat	90.6	43.7	686.9	6.7	14.6	33	1.3	0.0	0
4	8	6	mar	fri	91.7	33.3	77.5	9.0	8.3	97	4.0	0.2	0
5	8	6	mar	sun	89.3	51.3	102.2	9.6	11.4	99	1.8	0.0	0
6	8	6	aug	sun	92.3	85.3	488.0	14.7	22.2	29	5.4	0.0	0

Below is another overview of the data

```
'data.frame':  517 obs. of  13 variables:
 $ X    : int  7 7 7 8 8 8 8 8 8 7 ...
 $ Y    : int  5 4 4 6 6 6 6 6 6 5 ...
 $ month: chr  "mar" "oct" "oct" "mar" ...
 $ day  : chr  "fri" "tue" "sat" "fri" ...
 $ FFMC : num  86.2 90.6 90.6 91.7 89.3 92.3 92.3 91.5 91 92.5 ...
 $ DMC  : num  26.2 35.4 43.7 33.3 51.3 ...
 $ DC   : num  94.3 669.1 686.9 77.5 102.2 ...
 $ ISI  : num  5.1 6.7 6.7 9 9.6 14.7 8.5 10.7 7 7.1 ...
 $ temp : num  8.2 18 14.6 8.3 11.4 22.2 24.1 8 13.1 22.8 ...
 $ RH   : int  51 33 33 97 99 29 27 86 63 40 ...
 $ wind : num  6.7 0.9 1.3 4 1.8 5.4 3.1 2.2 5.4 4 ...
 $ rain : num  0 0 0 0.2 0 0 0 0 0 0 ...
 $ area : num  0 0 0 0 0 0 0 0 0 0 ...
```

Now, we will look at the summary statistics for the data variables

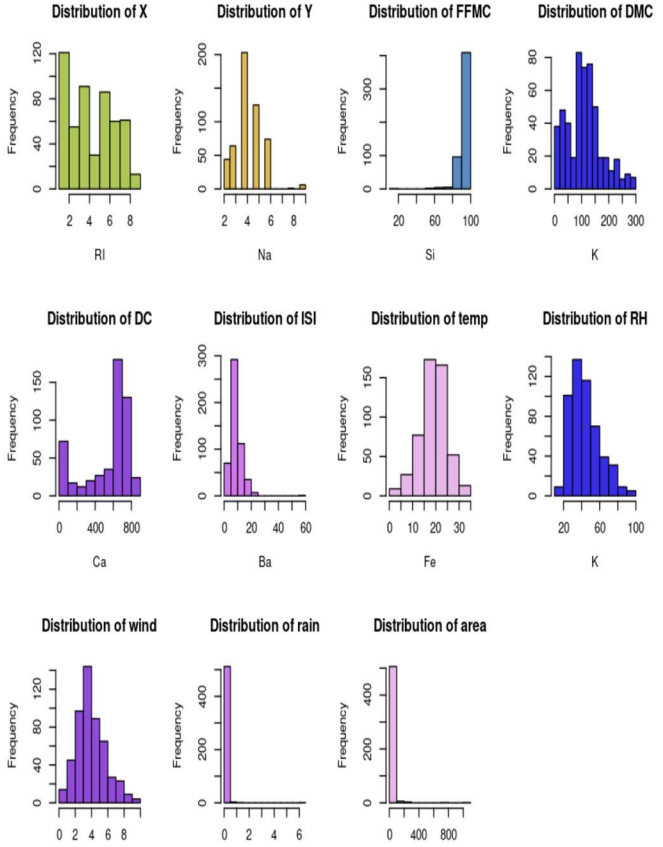
```
       X                Y             month               day           
 Min.   :1.000   Min.   :2.0   Length:517         Length:517        
 1st Qu.:3.000   1st Qu.:4.0   Class :character   Class :character  
 Median :4.000   Median :4.0   Mode  :character   Mode  :character  
 Mean   :4.669   Mean   :4.3                                        
 3rd Qu.:7.000   3rd Qu.:5.0                                        
 Max.   :9.000   Max.   :9.0                                        
      FFMC             DMC              DC              ISI        
 Min.   :18.70   Min.   :  1.1   Min.   :  7.9   Min.   : 0.000  
 1st Qu.:90.20   1st Qu.: 68.6   1st Qu.:437.7   1st Qu.: 6.500  
 Median :91.60   Median :108.3   Median :664.2   Median : 8.400  
 Mean   :90.64   Mean   :110.9   Mean   :547.9   Mean   : 9.022  
 3rd Qu.:92.90   3rd Qu.:142.4   3rd Qu.:713.9   3rd Qu.:10.800  
 Max.   :96.20   Max.   :291.3   Max.   :860.6   Max.   :56.100  
      temp             RH              wind            rain         
 Min.   : 2.20   Min.   : 15.00   Min.   :0.400   Min.   :0.00000  
 1st Qu.:15.50   1st Qu.: 33.00   1st Qu.:2.700   1st Qu.:0.00000  
 Median :19.30   Median : 42.00   Median :4.000   Median :0.00000  
 Mean   :18.89   Mean   : 44.29   Mean   :4.018   Mean   :0.02166  
 3rd Qu.:22.80   3rd Qu.: 53.00   3rd Qu.:4.900   3rd Qu.:0.00000  
 Max.   :33.30   Max.   :100.00   Max.   :9.400   Max.   :6.40000  
      area        
 Min.   :   0.00  
 1st Qu.:   0.00  
 Median :   0.52  
 Mean   :  12.85  
 3rd Qu.:   6.57  
 Max.   :1090.84  
```

From the above summary statistics, it can be observed that some of the variables have skewness and outlier issues. A visualization could help us understand the problem with each of the variables.

4 INITIAL DATA VISUALIZATIONS

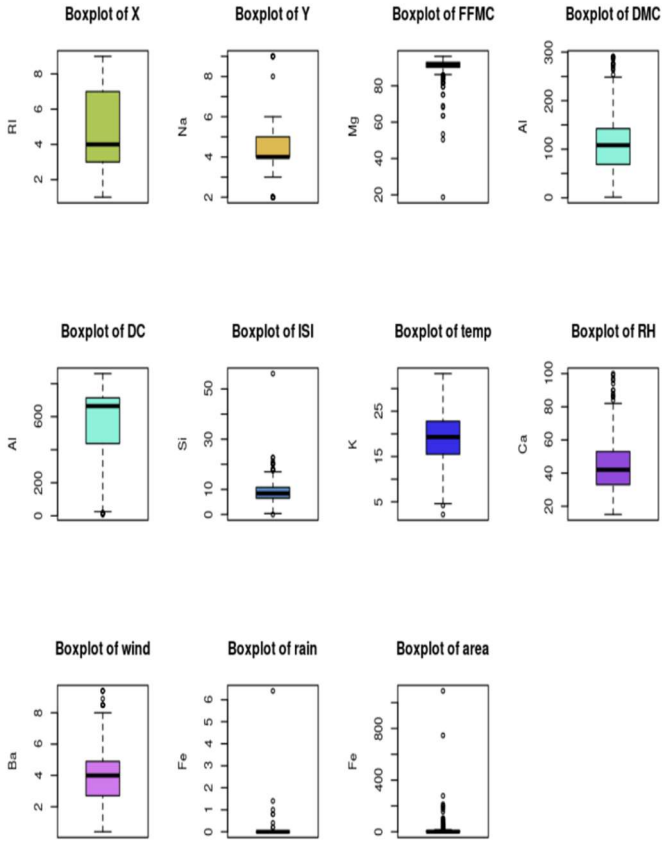
Below are a few of the visualizations of the data.

(i) Histogram



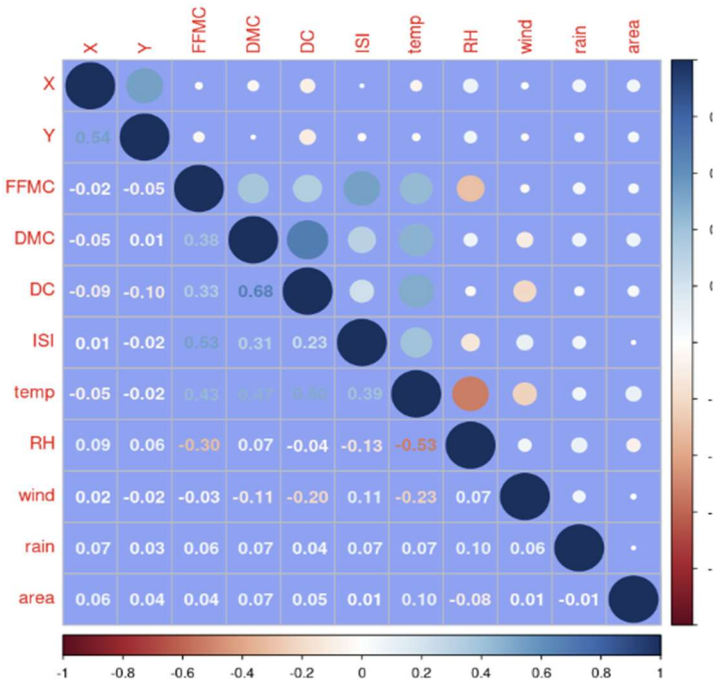
The visualization shows the distributions of various variables, with several exhibiting skewness. Variables like Na (Y), FFMC (Si), DMC (K), DC (Ca), ISI (Ba), rain, and area display positive skewness, indicating most values are clustered at lower ranges with tails extending rightward. RH (K) shows a negative skew, where values are concentrated toward the higher end. The temp (Fe) distribution is fairly symmetric, while wind shows a slight positive skew. Addressing the skewness in these variables, especially the extreme cases like rain and area, may require transformations (e.g., log normalization) for better data representation and analysis.

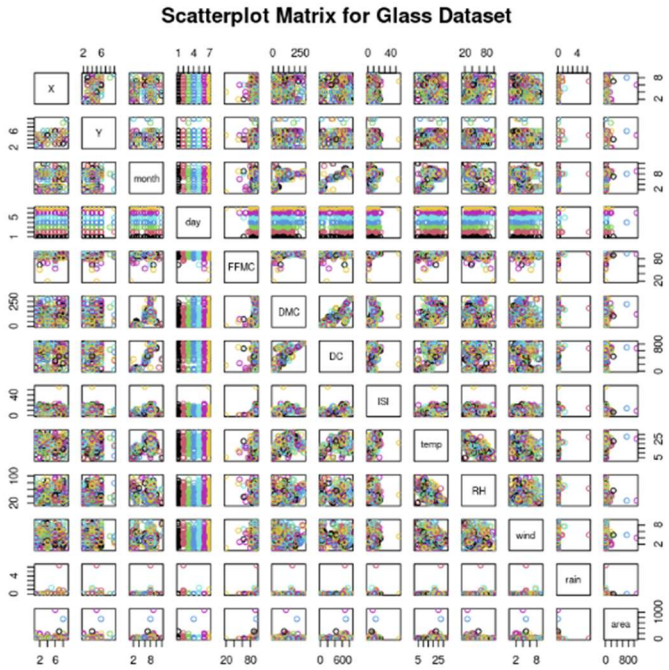
(ii) Boxplots



The boxplots illustrate the distribution and spread of various variables, highlighting the presence of outliers and skewness. **Y (Na)**, **FFMC (Mg)**, **ISI (Si)**, **rain (Fe)**, and **area (Fe)** show a significant number of outliers and exhibit positive skewness, with the data concentrated at lower values and outliers extending upward. **DC (Al)**, **DMC (Al)**, and **RH (Ca)** also display positive skewness with outliers, though the spread is broader. The **temp (K)** variable shows a fairly symmetric distribution, while **X (RI)** and **wind (Ba)** have a balanced spread with fewer outliers. The visualization indicates the need to address outliers and skewed distributions, particularly for **rain**, **area**, and **ISI**, to improve data analysis and modeling accuracy.

(iii) Correlation plots





The correlation matrix and scatterplot matrix provide insights into the relationships between variables. The correlation matrix highlights strong positive correlations, such as DMC and DC (0.68), ISI and FFMC (0.53), and ISI and temp (0.50), while RH and temp (-0.53) show a strong negative correlation. Most other correlations are weak or moderate. The scatterplot matrix visually supports these findings, with some linear patterns, such as DC vs. DMC and ISI vs. FFMC, while variables like area and rain display more scattered, non-linear relationships. These visualizations suggest potential groupings, outliers, and features worth exploring further in analysis and modeling.

(iv) Skewness Calculations.

Rain and area have very strong right skewness issues, FFMC and DC have strong left skewness issues.

```
          X           Y        FFMC         DMC          DC         ISI
 0.03603577  0.41487792 -6.53749886  0.54432492 -1.09406780  2.52162669
       temp          RH        wind        rain        area
-0.32925302  0.85790328  0.56769205 19.70150380 12.77248266
```

5 DATA PREPROCESSING

The preprocessing step is crucial for preparing the dataset for modeling. It involves applying specific transformations to variables to enhance the interpretability, reduce skewness, mitigate outliers, and ensure compatibility with the modeling methods.

Below are a few of the transformations applied on the dataset.

Variable	Transformation(s)	Reason for Transformation
X	None	Spatial location, no skewness or scaling issues.
Y	None	Spatial location, no skewness or scaling issues.
FFMC	Spatial Sign, Winsorized	To cap extreme values and rescale the variable.
DMC	None	No skewness or outliers, retained as-is.
DC	Spatial Sign, Winsorized	Outliers and scale differences needed adjustment.
ISI	Box-Cox, Winsorized	Managed skewness and stabilized variance.
temp	None	Retained as-is; data was well-behaved.
RH	Box-Cox	Addressed significant skewness in distribution.
wind	None	Retained as-is; data showed uniform behavior.
rain	Spatial Sign, Winsorized	Addressed extreme skewness and capped outliers.
area	Spatial Sign, Winsorized	Addressed skewness and capped extreme fire sizes.
month	Removed	Redundant with seasonal variables like temp, RH.
day	Removed	No significant relationship with fire spread.

These transformations helped reduce the outliers and skewness issues. Few of the irrelevant features like month and day are very removed.

```
          X           Y        FFMC         DMC          DC         ISI
 0.03603577  0.41487792 -0.67727545  0.54432492 -0.54474996 -0.05472982
       temp          RH        wind        rain        area
-0.32925302 -0.01043580  0.56769205 -0.54678974 -0.35461139
```

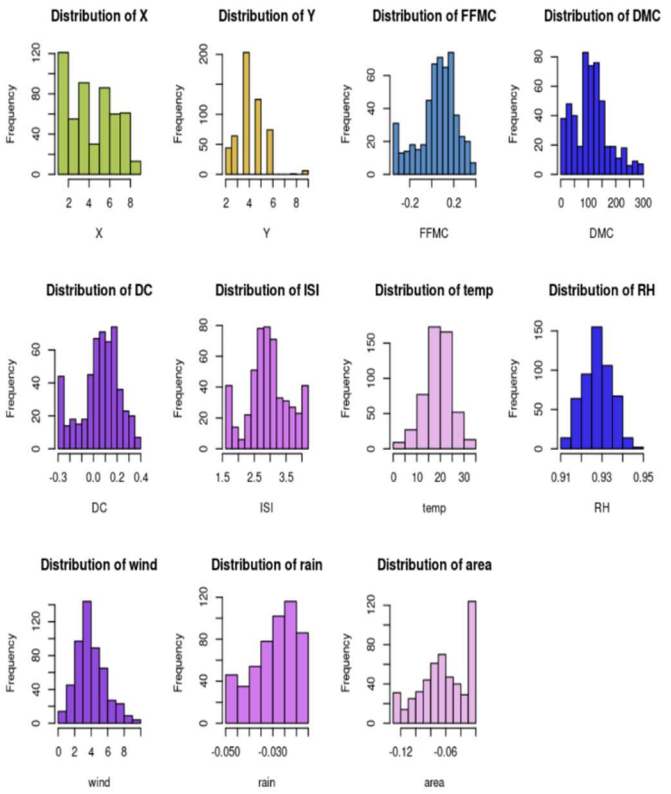
Above are the skewness values after the Data Preprocessing. Below is a preview of the dataset after transformations

	X	Y	FFMC	DMC	DC	ISI	temp	RH	wind	rain	area
	<int>	<int>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	7	5	-0.217819452	26.2	-0.217819452	2.276979	8.2	0.9236419	6.7	-0.01980157	-0.05459808
2	7	4	-0.003165898	35.4	-0.003165898	2.613424	18.0	0.9318005	0.9	-0.02862984	-0.07893993
3	7	4	-0.003272341	43.7	-0.003272341	2.613424	14.6	0.9318005	1.3	-0.02959243	-0.08159402
4	8	6	0.039649818	33.3	0.039649818	3.033478	8.3	0.9117236	4.0	-0.01508361	-0.04185802
5	8	6	-0.051199693	51.3	-0.051199693	3.133984	11.4	0.9113478	1.8	-0.01538479	-0.04241988
6	8	6	0.108162029	85.3	0.108162029	3.886461	22.2	0.9342360	5.4	-0.02640199	-0.07279715

6 VISUALAZATIONS AFTER PREPROCESSING

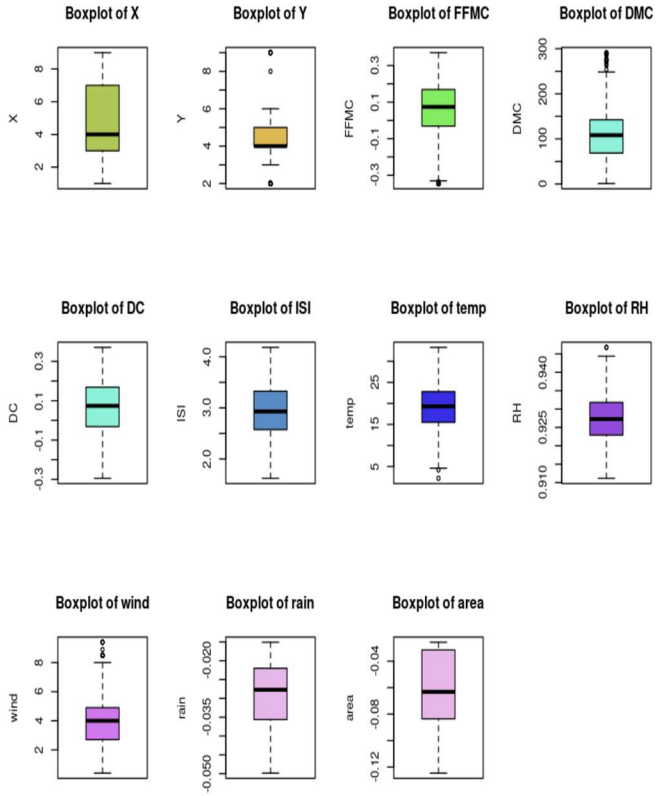
Below are a few of the visualizations of the data.

(i) Histogram



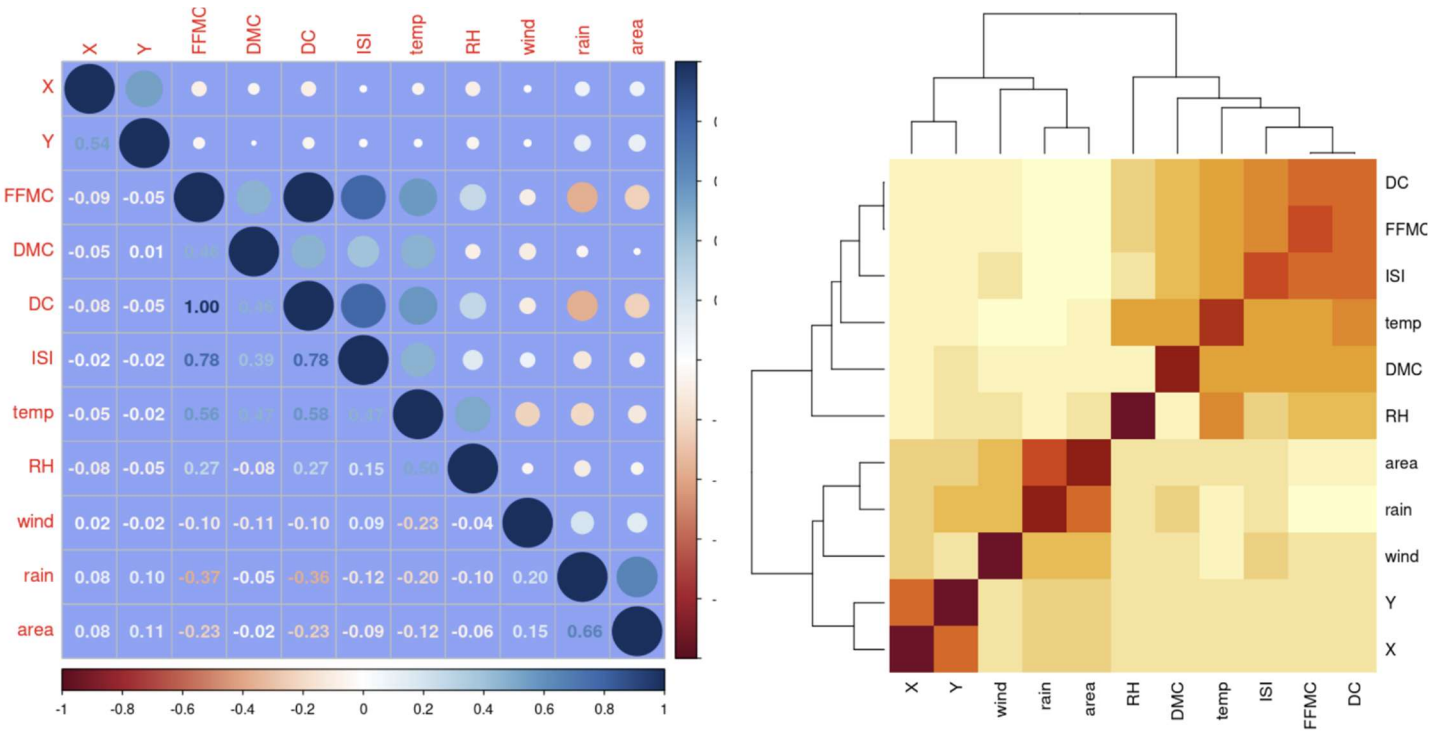
This collection of histograms shows the distributions of various features, revealing key aspects of skewness and potential outlier issues. **X** and **Y** exhibit left-skewness, with a concentration on higher values. **FFMC**, **DMC**, and **DC** are approximately normally distributed, though **DMC** shows slight right-skewness. **ISI** and **wind** display a right-skewed distribution, while **rain** and **area** are notably skewed to the left and exhibit potential outliers on the lower end. **Temp** has a roughly symmetric distribution, while **RH** shows a tight distribution with minor skewness. The **rain** and **area** histograms particularly indicate outliers in the negative range.

(ii) Boxplots



This collection of boxplots highlights skewness and outlier issues for various features. **X** and **Y** show a slight left skew, with **Y** having several outliers on the lower end. **FFMC** exhibits slight left skewness with an outlier below the lower whisker. **DMC** is right-skewed with multiple outliers above the upper whisker. **DC** shows a symmetrical distribution, while **ISI** is relatively symmetrical but slightly skewed to the right. **Temp** is symmetrical but has a few low-end outliers. **RH** is slightly right-skewed with one upper-end outlier. **Wind** demonstrates right-skewness with numerous outliers above the upper whisker. **Rain** has a left-skewed distribution with no apparent outliers, while **area** shows left skewness and potential lower-end outliers.

(iii) Correlation plots



The correlation matrix and dendrogram heatmap collectively highlight the relationships among the various features. The **correlation matrix** shows that **X** and **Y** are moderately correlated (**0.54**), while **FFMC** and **ISI** have a strong positive correlation (**0.78**). Additionally, **DMC**, **DC**, and **temp** have moderate correlations with each other, while **rain** is negatively correlated with **FFMC** and **ISI**. The **dendrogram heatmap** groups the features based on similarity, with **FFMC**, **ISI**, **DMC**, and **temp** clustered together, reflecting their relatedness. Similarly, **X** and **Y** are clustered, and **rain** and **wind** form a separate group. The combination of these visualizations indicates that some features are strongly interrelated, which could be useful for feature reduction or understanding multicollinearity in modeling.

(iv) Skewness Calculations

Skewness for all variables have been reduced significantly. Now data is ready for model implementation.

```
         X          Y       FFMC        DMC         DC        ISI
0.03603577 0.41487792 -0.67727545 0.54432492 -0.54474996 -0.05472982
      temp         RH       wind       rain       area
-0.32925302 -0.01043580 0.56769205 -0.54678974 -0.35461139
```

7 MODEL IMPLEMENTATION

The transformed data is split into 80-20 split for training and testing datasets. Training data will be used to training the models and testing is used for validating the model with unseen data.

In this project, a variety of predictive models were implemented to predict the burned forest area (area) based on the environmental and weather conditions provided in the dataset. Both **linear** and **non-linear models** were utilized to capture the underlying patterns in the data. The models were selected to ensure a robust comparison between simpler models and more complex machine learning algorithms.

7.1 LINEAR MODELS

7.1.1 LINEAR REGRESSION

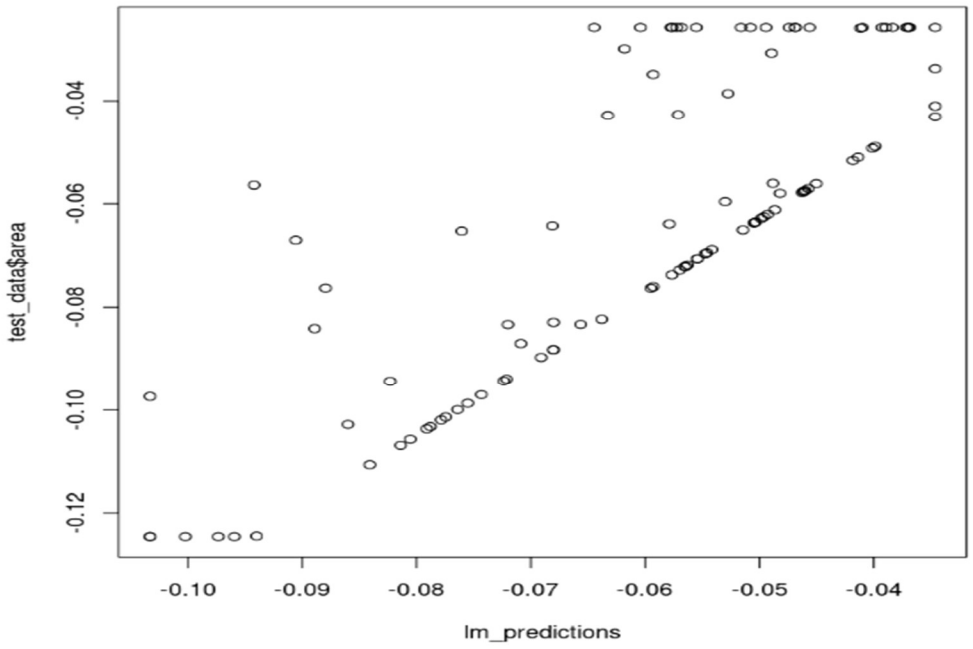
Linear Regression is a fundamental statistical method used to model the relationship between a dependent variable and one or more independent variables. It assumes a linear relationship and finds the best-fit line by minimizing the difference (residuals) between the observed data and the predicted values. In simple linear regression, the model takes the form:

$$y = \beta_0 + \beta_1 x + \epsilon$$

Where y is the dependent variable, x is the independent variable, β0 is the y-intercept, β1 is the slope (coefficient), and ϵ is the error term. Linear regression is widely used for predictive modeling and identifying relationships between variables due to its simplicity and interpretability.

- AIC is used to finalize the most important features for the model. And 'rain' is found to be most important model for the linear regression for predicting 'area' variable.
- 10 – fold cross validation was used for ensuring that the model's performance is not biased toward any single subset of data.
- 'rain' variable was found to be statistically significant.

RMSE: 0.0192276720757619 **Rsquared:** 0.631901237189252 **MAE:** 0.0175770590310894



The accuracy of Linear regression model is 63.19%.

7.1.2 REGULARIZATION MODEL

Regularization models are an extension of linear regression used to prevent **overfitting** by adding a penalty to the model's coefficients. Overfitting occurs when a model captures noise or random fluctuations in the training data, leading to poor performance on new, unseen data. Regularization helps improve the model's generalization by constraining the size of the coefficients, thus simplifying the model.

There are three main types of regularization methods commonly used:

Ridge Regression (L2 Regularization): Adds a penalty proportional to the square of the coefficients, Helps shrink large coefficients but does not reduce them to zero, and Useful when dealing with multicollinearity (highly correlated predictors).

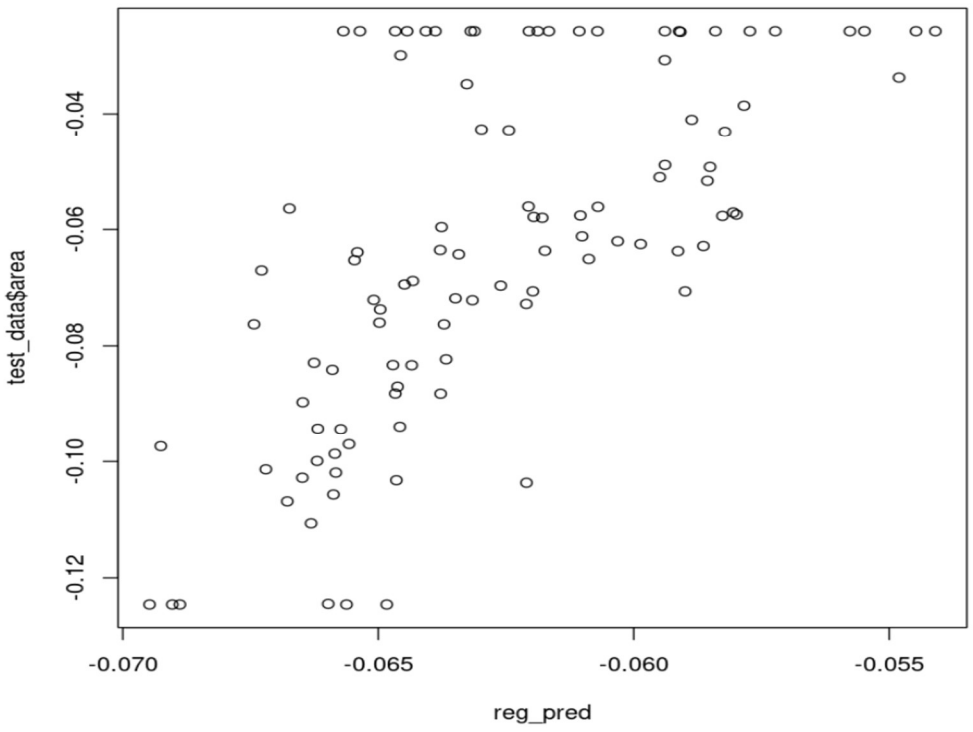
Lasso Regression (L1 Regularization): Adds a penalty proportional to the absolute value of the coefficients, Can shrink some coefficients to zero, effectively performing feature selection, and Suitable when the dataset contains many irrelevant features.

- For this project the best tune for the regularization model was identified as Pure Ridge.

	alpha	lambda
	<dbl>	<dbl>
1	0	0.2

- 10 – fold cross validation was used for ensuring that the model's performance is not biased toward any single subset of data.
- 'glmnet' method is used in the train function.

RMSE: 0.0276104242412958 **Rsquared:** 0.431255930131635 **MAE:** 0.0224409297569756



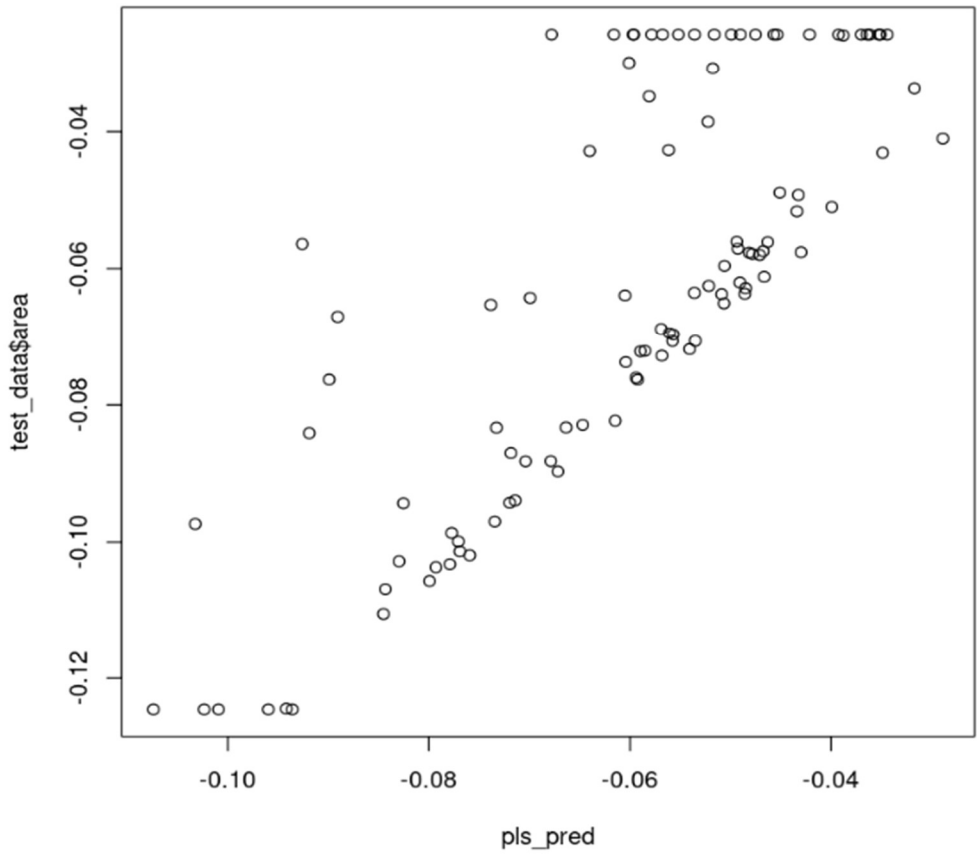
The accuracy of Regularization model regression model is 43.12%.

7.1.3 PARTIAL LEAST SQUARES REGRESSION

Partial Least Squares (PLS) Regression is a statistical method used for modeling relationships between a set of independent variables (predictors) and dependent variables (responses). PLS is particularly useful when predictors are highly collinear or when the number of predictors exceeds the number of observations. It works by projecting both the predictor and response variables onto a new set of latent variables that maximize the covariance between them. This technique reduces dimensionality while retaining the predictive power of the model, making it suitable for applications involving large, complex datasets, such as chemometrics, bioinformatics, and machine learning.

- For this project the model used 9 components. Tune length is used 10.
- 10 – fold cross validation was used for ensuring that the model's performance is not biased toward any single subset of data.
- 'pls' method is used in the train function.
- 'pls' library is used for this model.

RMSE: 0.0193336729736992 **Rsquared:** 0.622025957916158 **MAE:** 0.0174941960994148



The accuracy of Partial Least Squares regression model is 62.20%.

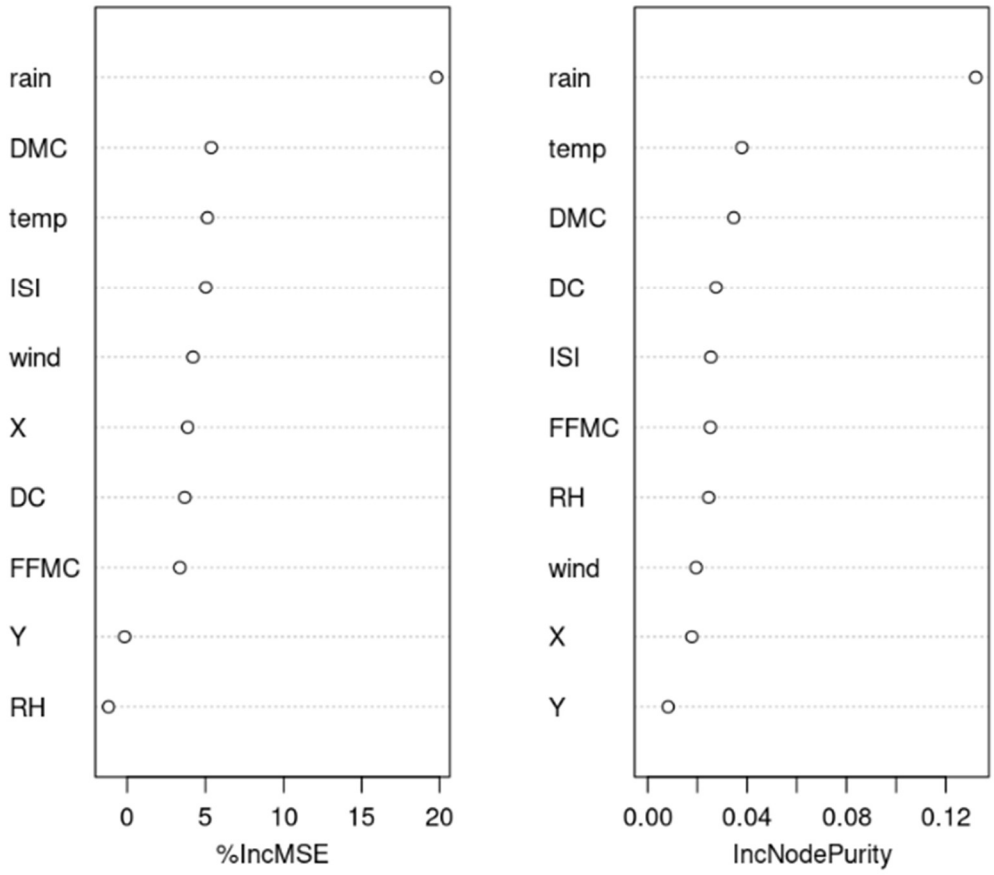
7.2 NON LINEAR MODELS

7.2.1 RANDOM FORESTS

Random Forests is an ensemble learning method used for both classification and regression tasks. It operates by constructing a multitude of decision trees during training and combines their outputs to improve accuracy and prevent overfitting. Each tree is trained on a random subset of the data, and at each split, a random subset of features is considered. The final prediction is determined by averaging the results (for regression) or majority voting (for classification). Random Forests are known for their robustness, scalability, and ability to handle large datasets and high-dimensional feature spaces.

- 'randomForest' library is utilized for this model
- 150 trees are used as tuning parameter
- Based on importance plot, 'rain' variable is identified as the most important predictor contributing to predicting 'area'.

RMSE: 0.0206807980441559 **Rsquared:** 0.568534134883468 **MAE:** 0.0181240705699669



The accuracy of Random Forests model is 56.85%.

7.2.2 GRADIENT BOOSTING METHODS

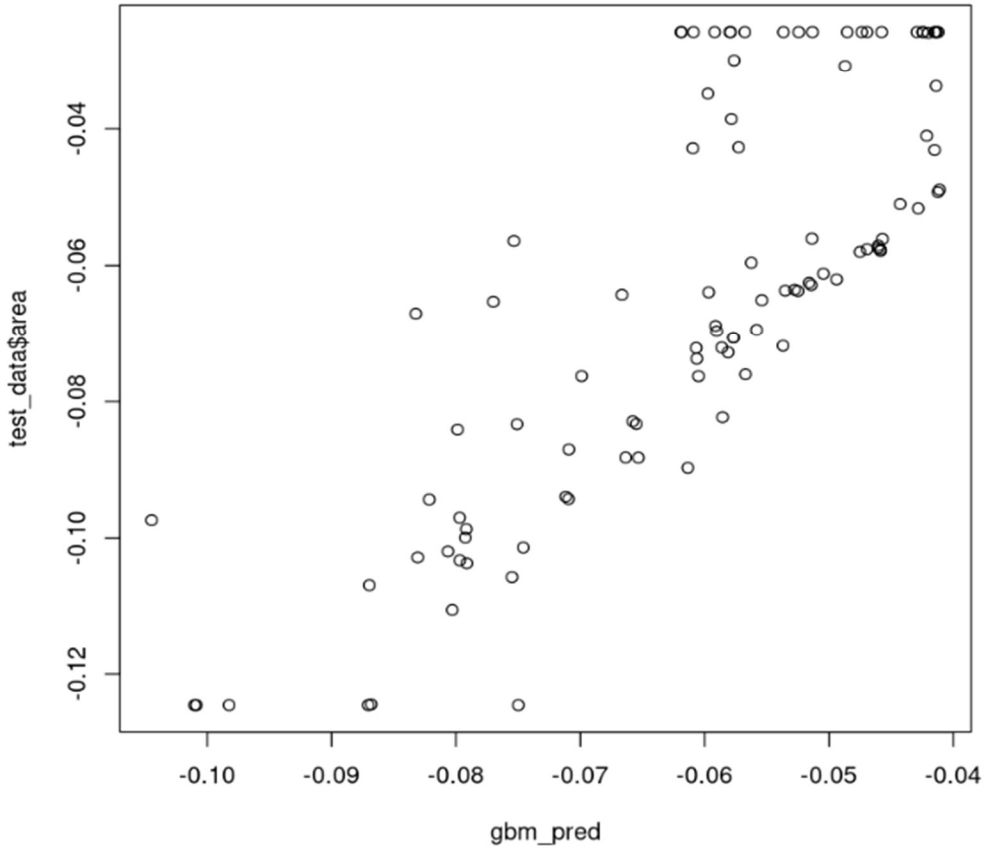
Gradient boosting is a powerful machine learning technique that builds a strong model by combining multiple weak learners, typically decision trees. It works by iteratively training new models to correct the errors of previous ones, resulting in highly accurate predictions. This method is widely used in various domains, including predictive modeling, classification, and ranking, due to its flexibility, robustness, and strong performance.

- The tuning parameters are tuned for best prediction outcome possible

```
gbm_model <- gbm(area ~ ., data = train_data, distribution = "gaussian", n.trees = 100, shrinkage = 0.1,
                 interaction.depth = 3, n.minobsinnode = 10, cv.folds = 5)
```

- The model utilized 21 trees.
- There were 10 predictors of which 9 had non-zero influence.
- 'gbm' library is used for the model.

RMSE: 0.0199226720153965 **Rsquared:** 0.643264667335393 **MAE:** 0.0177118899991812



The accuracy of gradient boosting methods model is 64.36 %.

7.2.3 SUPPORT VECTOR MACHINE

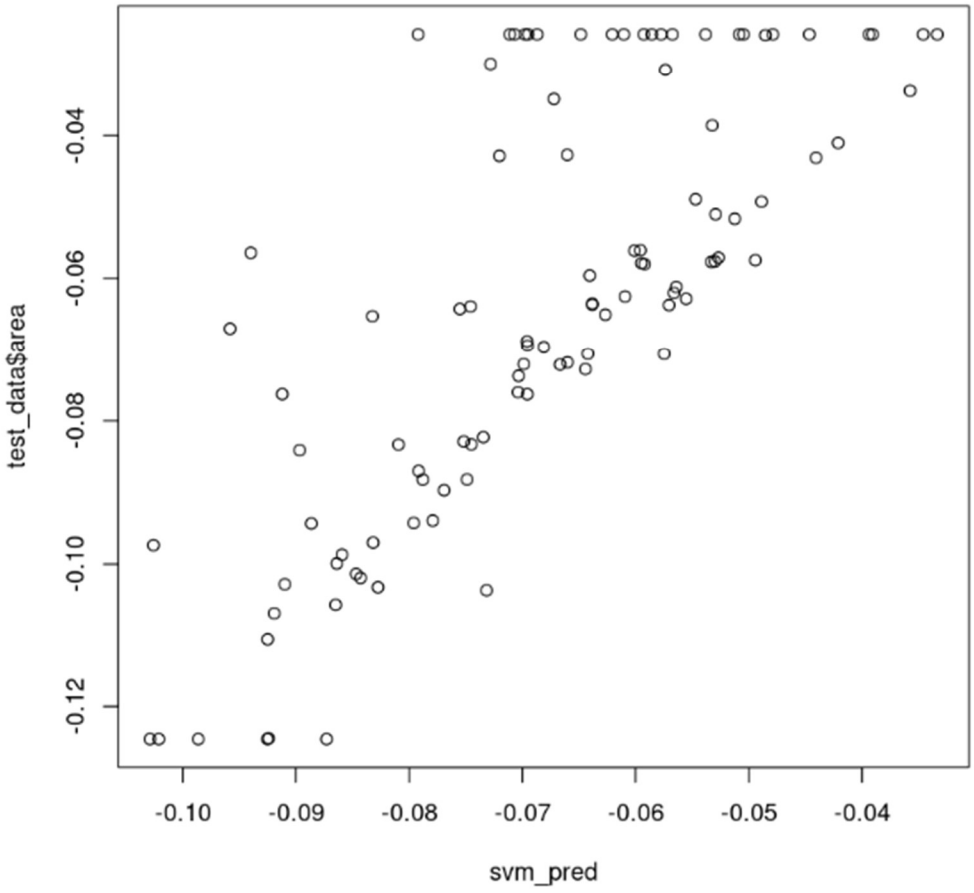
Support Vector Machines (SVMs) are a powerful machine learning algorithm used for classification and regression. They find the optimal hyperplane that maximizes the margin between data points of different classes. SVMs excel in high-dimensional spaces and can handle complex datasets. By focusing on support vectors, they create robust and efficient models.

- The best tune for this model is identified to determine best result.

	sigma	C
	<dbl>	<dbl>
1	0.01	0.25

- 5 – fold cross validation was used for ensuring that the model's performance is not biased toward any single subset of data.
- 'svmRadial' method is used for the model.

RMSE: 0.0208262170977654 **Rsquared:** 0.578775558144919 **MAE:** 0.0159096552297073



The accuracy of Support Vector Machines model is 57.87 %.

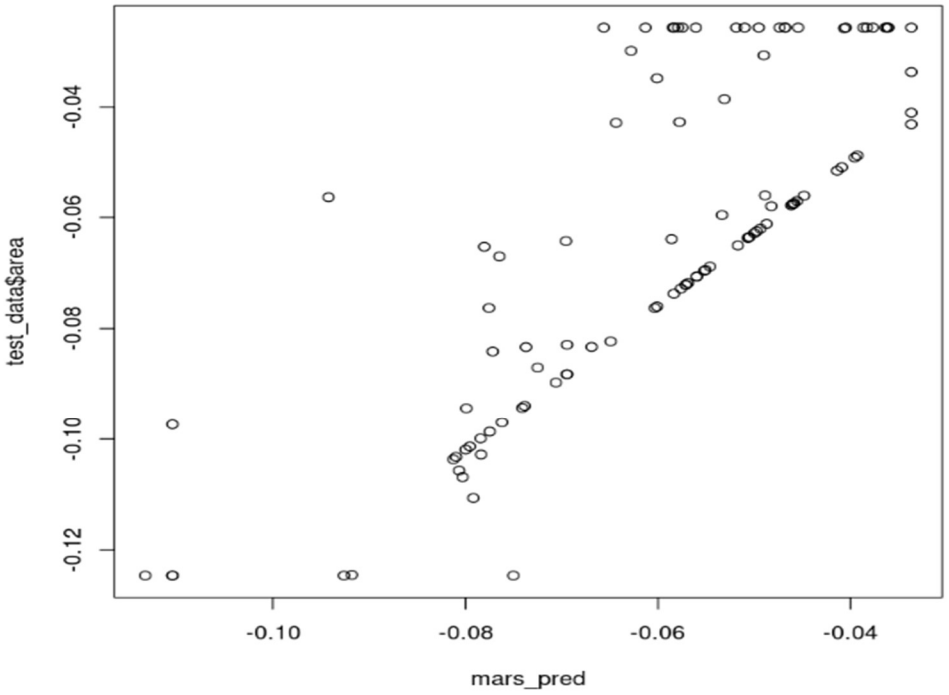
7.2.4 MARS (Multivariate Adaptive Regression Splines)

Multivariate Adaptive Regression Splines (MARS) is a non-parametric regression technique that captures both linear and non-linear relationships in data. It uses piecewise linear basis functions to model complex interactions between the dependent and independent variables. The model is built in two phases: a forward pass, where basis functions are added to maximize fit, and a backward pass, where redundant functions are pruned to prevent overfitting. MARS adapts to the data, automatically identifying significant variables and interactions, making it highly flexible and effective for high-dimensional datasets. Despite its complexity, the additive structure of MARS ensures that the model remains interpretable. Its versatility has made it popular in fields like finance, engineering, and marketing, where capturing non-linear dynamics is crucial.

- 10 – fold cross validation was used for ensuring that the model's performance is not biased toward any single subset of data.
- 'earth' method is used for the model.
- Tuning parameters used are
  RMSE was used to select the optimal model using the smallest value. The final values used for the model were nprune = 10 and degree = 1.

	nprune	degree
	<dbl>	<int>
1	10	1

RMSE: 0.0193462642439133 **Rsquared:** 0.622384216624035 **MAE:** 0.0173049405835712



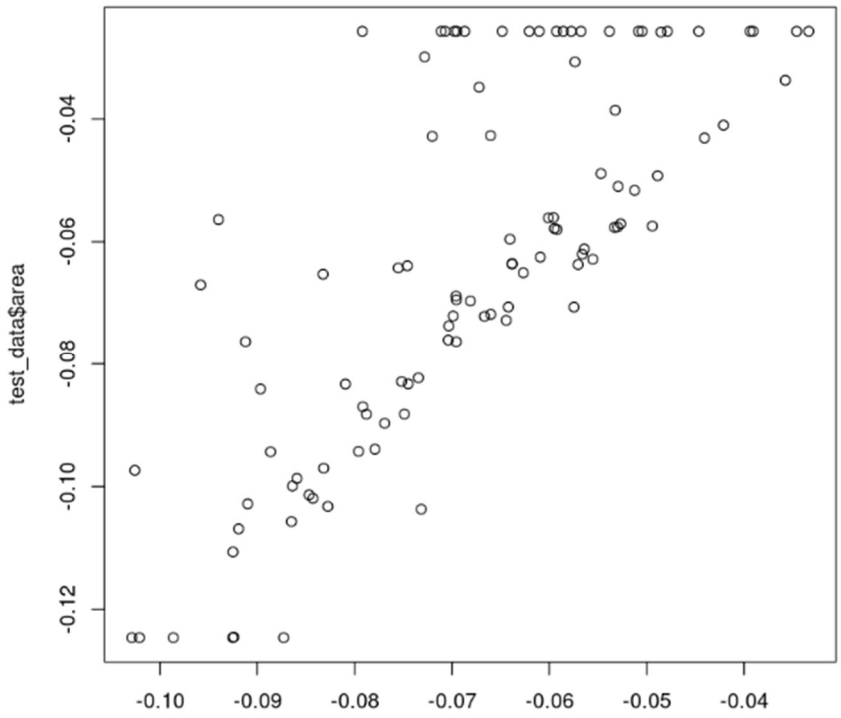
The accuracy of Multivariate Adaptive Regression model is 62.23 %.

7.2.5 k-NN REGRESSION

k-Nearest Neighbors (kNN) Regression is a non-parametric algorithm used for predicting continuous target variables. Instead of learning a model, kNN stores the training data and makes predictions by identifying the kkk nearest data points to a query point based on a similarity measure, such as Euclidean or Manhattan distance. The predicted value is typically calculated as the average (or weighted average) of the target values of these neighbors. This method is simple, intuitive, and effective for capturing local data patterns. However, it can be computationally expensive for large datasets due to the need to compute distances for all training points. The algorithm's performance depends heavily on the choice of kkk, the distance metric, and the presence of noise in the data. Despite these limitations, kNN regression is widely used for its flexibility and ability to model complex relationships without assuming a specific functional form.

- 'knnreg() function is used for the model
- Number of neighbors are 4.

RMSE: 0.0193462642439133 **Rsquared:** 0.622384216624035 **MAE:** 0.0173049405835712



The accuracy of kNN Regression model is 62.23 %.

7.2.6 NEURAL NETWORKS (Ensemble methods-avNNet)

The avNNet model is an ensemble approach that combines multiple artificial neural networks (ANNs) to improve predictive accuracy and reduce overfitting. Each ANN in the ensemble is trained with variations, such as different initial weights or random data samples, and their predictions are averaged to produce a final output. This method leverages the power of ANNs to model complex, non-linear relationships while enhancing robustness and generalization through ensemble averaging. Implemented in R using the caret package, avNNet allows easy customization of hyperparameters like hidden units, regularization (decay), and iterations. It is suitable for both regression and classification tasks and is commonly applied in fields like finance, healthcare, and engineering. The ensemble approach makes avNNet particularly effective in achieving stable and accurate predictions across diverse datasets.\

- The tuning parameters are tuned for best prediction outcome possible

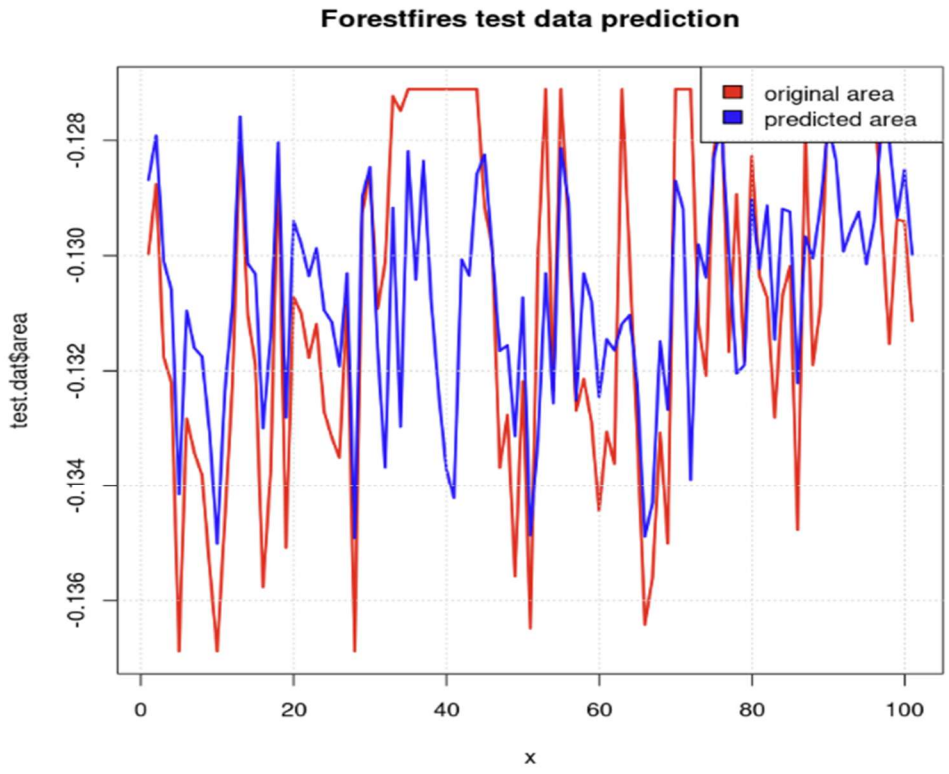
```
avn_model = train(area ~., data = train.dat, method = 'avNNet', tuneGrid = nnet.Grid,
                  trControl = ctrl, preProc = c('center', 'scale'), linout = TRUE,
                  trace = FALSE, maxit = 100, allowParallel = FALSE)
```

- 10 – fold cross validation was used for ensuring that the model's performance is not biased toward any single subset of data.
- Grid is tuned as below

```
nnet.Grid = expand.grid(decay = c(0,0.01,0.05,0.1), size = 1:6, bag = FALSE)
```

- 'avNNet' method is used for the train function.

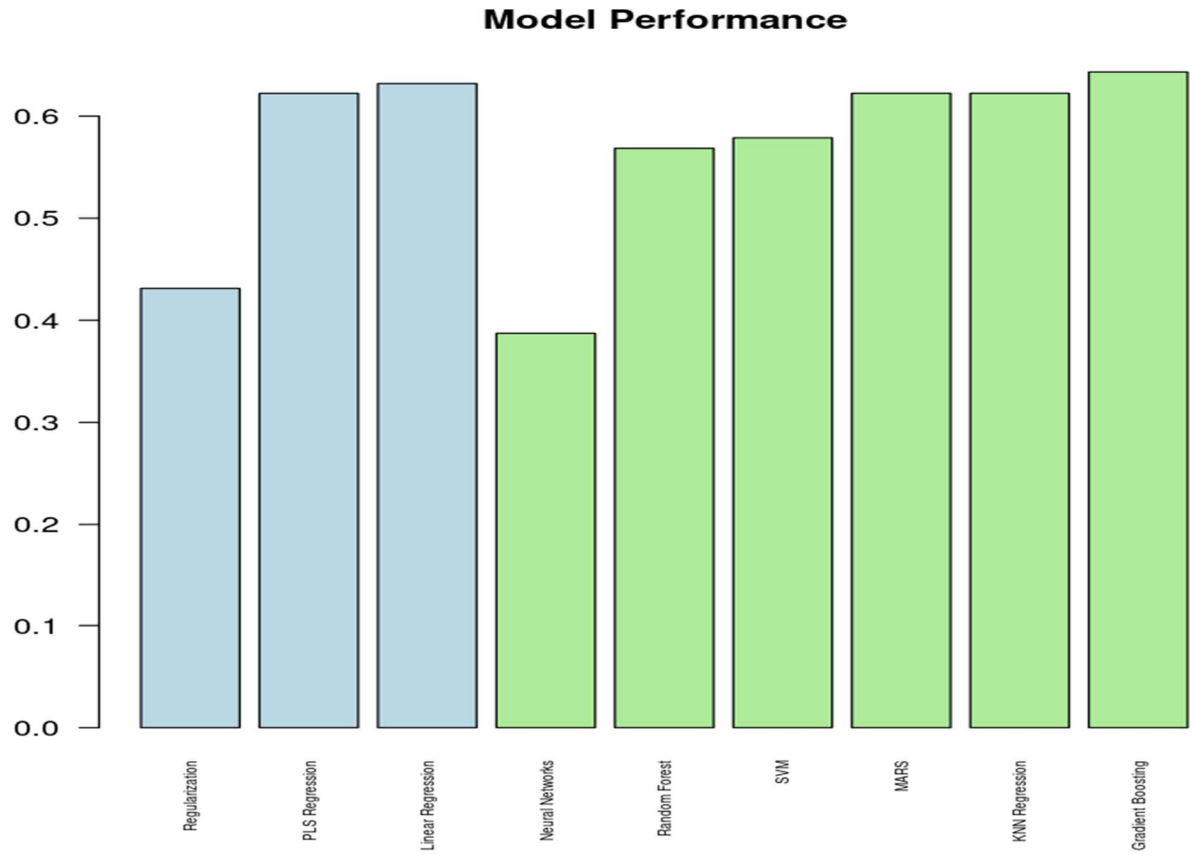
RMSE: 0.0709685454597274 **Rsquared:** 0.387286887171427 **MAE:** 0.0687786189730122



The accuracy of kNN Regression model is 38.72%.

8 RESULT INTERPRETATION

Below is the summarized plots of accuracies of all the models



The project aimed to predict the burned area of forest fires using a range of machine learning models. The models implemented included both linear and non-linear techniques. Here's a summary of the results:

1. **Linear Models**:
    - **Linear Regression**: Achieved an accuracy of **63.19%**. The 'rain' variable was found to be the most significant predictor.
    - **Regularization (Ridge Regression)**: Achieved an accuracy of **43.12%**. Ridge regularization was effective in reducing overfitting but showed lower performance.
    - **Partial Least Squares Regression**: Achieved an accuracy of **62.20%**, showing it can handle multicollinearity effectively.
2. **Non-Linear Models**:
    - **Random Forest**: Accuracy of **56.85%**. The 'rain' variable was identified as the most important feature.
    - **Gradient Boosting**: Achieved the highest accuracy of **64.36%**, indicating its strength in capturing non-linear relationships.
    - **Support Vector Machine (SVM)**: Accuracy of **57.87%**.
    - **MARS (Multivariate Adaptive Regression Splines)**: Accuracy of **62.23%**, effectively handling non-linear interactions.

- **k-Nearest Neighbors (k-NN)**: Accuracy of **62.23%**, showing simplicity and effectiveness in local pattern recognition.
- **Neural Networks (avNNet)**: Accuracy of **38.72%**, indicating challenges in tuning for this dataset.

Key Insights:

1. **Best Performing Models**:
   - **Gradient Boosting (64.36%)** showed the best performance, indicating that ensemble methods are particularly effective for predicting the burned area.
   - **Linear Regression (63.19%)** also performed well, suggesting that a linear approach can be useful if significant predictors like 'rain' are identified.
2. **Feature Importance**:
   - The **'rain'** variable consistently emerged as the most important predictor across multiple models.
   - Non-linear models such as Gradient Boosting and MARS captured complex interactions better than simpler linear models.
3. **Model Comparison**:
   - Non-linear models generally performed better at capturing the intricate relationships in the dataset compared to linear models.
   - Regularization methods like Ridge Regression showed reduced performance, likely due to the complexity and non-linearity in the data.

9 CONCLUSIONS

his project explored the use of various machine learning models to predict the burned area of forest fires based on environmental and weather conditions. Given the increasing frequency and severity of forest fires, accurate prediction models can play a crucial role in improving fire preparedness, resource allocation, and mitigation strategies.

1. **Model Performance**:
   - The models tested included both linear and non-linear approaches:
     - **Linear Regression** provided a strong baseline with **63.19%** accuracy, highlighting its effectiveness for straightforward relationships.
     - **Gradient Boosting** emerged as the best-performing model with an accuracy of **64.36%**, demonstrating its ability to capture complex, non-linear interactions within the data.
     - **Random Forest (56.85%)**, **Support Vector Machine (57.87%)**, and **MARS (62.23%)** also showed competitive performance, emphasizing the importance of ensemble and adaptive methods for predictive accuracy.
     - **Neural Networks (38.72%)** performed poorly, likely due to the complexity of tuning and the limited dataset size.
2. **Key Findings**:
   - **Feature Importance**: The **'rain'** variable consistently appeared as the most significant predictor across different models. This indicates that rainfall plays a critical role in determining the extent of fire damage.
   - **Complex Relationships**: The performance of non-linear models suggests that forest fire data involves intricate, non-linear patterns that linear models struggle to capture fully.

- **Data Challenges**: Issues such as skewness, outliers, and multicollinearity were addressed during preprocessing, significantly improving model performance.

3. **Practical Implications**:

   - **Resource Allocation**: Accurate burned area predictions can help firefighting agencies allocate resources more effectively and prioritize high-risk regions.
   - **Environmental Protection**: Improved models aid in protecting sensitive ecosystems and biodiversity by enabling timely interventions.
   - **Economic Planning**: Predicting the extent of fire damage helps estimate potential economic losses and prepare recovery plans.

In conclusion, leveraging machine learning for predicting forest fire damage holds significant potential. While models like **Gradient Boosting** currently provide the best results, ongoing research and data integration will further enhance predictive capabilities, aiding in more effective fire management and mitigation strategies.