DS5500 Phase 2: Final Report - Team 17

Forest Fire Prediction

Team Members: Akshaya Mahesh, Sakthi Kripa Selvan, Manasa Krishnan

1. Introduction

Uncontrolled forest fires have the potential to seriously destroy the environment and wildlife, disrupting traffic, communications, and the water supply. While 60% of these fires are caused by natural causes, the remaining 40% are caused by human activity. Since wildfires impair the natural ecology in addition to the air quality, action is necessary to reduce this issue. For our project, the dataset we have chosen has 12 feature variables, including temperature, rain, fire weather index (FWI), moisture, and others as given in ***Figure 1***. We utilized this information to build machine learning models that can predict the affected area of forest cover in hectares. These models enable wildlife agencies to plan ahead for disaster relief initiatives. In the previous phase, we implemented fine-tuned Tree and XGBoost regressors to solve this problem. In this phase, we experimented with Recursive Feature Elimination(RFE), a feature engineering technique to select the most relevant features and we improved the model's performance by building stacked regression models. We also deployed a GUI to display the affected area of the forest given feature variables' values.

2. Methodology and Results

The workflow of our project is shown in ***Figure 2***

2.1 Low Risk: To understand the data and its attributes better, we performed exploratory data analysis(Eg: ***Figure 4***) in the previous phase. In Phase 2, we tried to improve the model and make the algorithm effective against noise. We removed possible outliers from FFMC, ISI and Rain using the ranges from ***Figure 3*** and corrected the skewness in the Area variable. We also utilized Recursive Feature Elimination(RFE), a feature engineering technique which works in parallel with the chosen machine learning algorithm to improve the model by selecting the most relevant features. Though proven to solve overfitting in many cases, the RFE did not work out for our use case as it further decreased the performance of the model. The possible reasons for this could be that we had only 12 features and few records and eliminating them will further decrease the information needed for efficient prediction. Hence, we proceeded further without using RFE and removed the possible outliers from the features.

2.1 Medium Risk: At the end of phase 1, the Random Forest tuned with RandomizedSearchCV gave the best results as shown in ***Table 2***. In phase 2, we further improved the model by utilizing ensemble techniques. Stacking regression is an ensemble technique to combine multiple regression models via a meta-regressor. Though this technique makes it difficult to interpret the model results, it increased our model performance significantly. We used linear regression as a meta model here. Since these are regression models, we will be using R squared and RMSE to evaluate them. We tried different combinations of models as shown in ***Table 1*** out of which XGBoost + Gradient Boosting gave the best results. It had a higher R2 score and lesser RMSE score than the other models. Since this was a significant improvement, we finalized upon this model to build a GUI.

Model	RMSE	R2
Linear Regression + Random Forest	1.423	0.18
Lasso + Gradient Boosting	1.424	0.13
XGBoost + Gradient Boosting	1.417	0.19

Table 1 Results at the end of Phase 2

2.3 High Risk: We developed a simple GUI as shown in ***Figure 5*** using ipywidgets where the inputs are the input features like temperature, rain and other FWI factors. The best model which is a stacking regressor of XGBoost and Gradient Boosting regressors then predicts the affected area in hectares and displays the output.

3. Conclusion: Future Works and Insights

The project, being a difficult regression task, owing to very less number of datapoints and tough interpretation of climatic factors and outliers, proves to work efficiently well compared to related works as the implemented stacking technique does a good job in understanding the relative importance of the features. We performed high level ablation settings with algorithms and low level ablation settings with pre-processing and feature engineering which provided us with a good learning source for the point of failure and how to rectify it. In the future, the project can be extended incorporating a large dataset which may make the model more efficient. The deployed real-time application will be very useful to the Forest Fire Department to monitor the affected forest cover and take strategic actions.

5. Figures and Tables

X	Y	month	day	FFMC	DMC	DC	ISI	temp	RH	wind	rain	area
7	5	mar	fri	86.2	26.2	94.3	5.1	8.2	51	6.7	0	0
7	4	oct	tue	90.6	35.4	669.1	6.7	18	33	0.9	0	0
7	4	oct	sat	90.6	43.7	686.9	6.7	14.6	33	1.3	0	0
8	6	mar	fri	91.7	33.3	77.5	9	8.3	97	4	0.2	0
8	6	mar	sun	89.3	51.3	102.2	9.6	11.4	99	1.8	0	0

Figure 1 Sample dataset from the UCI Machine Learning Repository

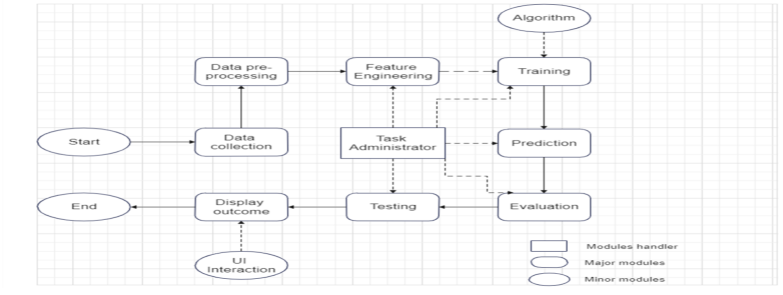

Figure 2 Machine Learning Pipeline(Model Architecture)

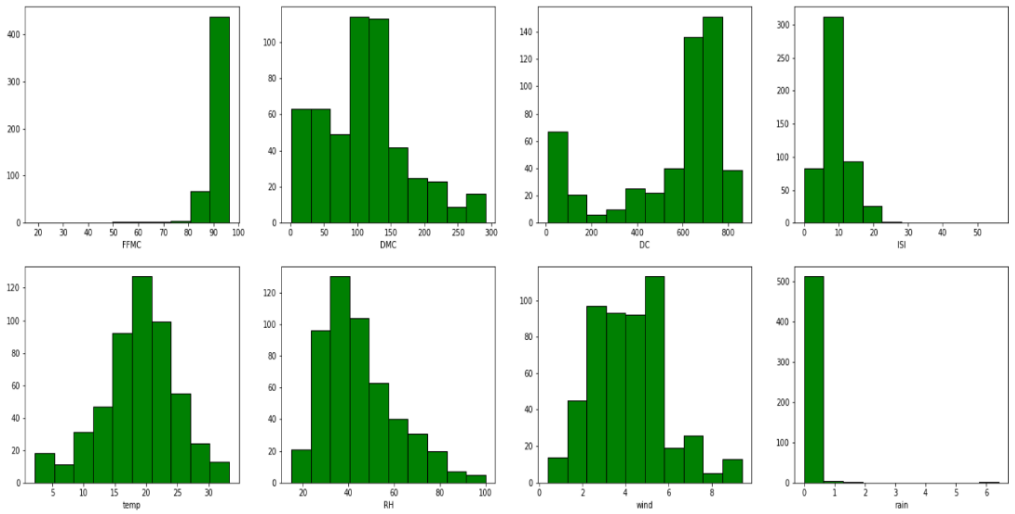

Figure 3 Histograms showing descriptive statistics of data

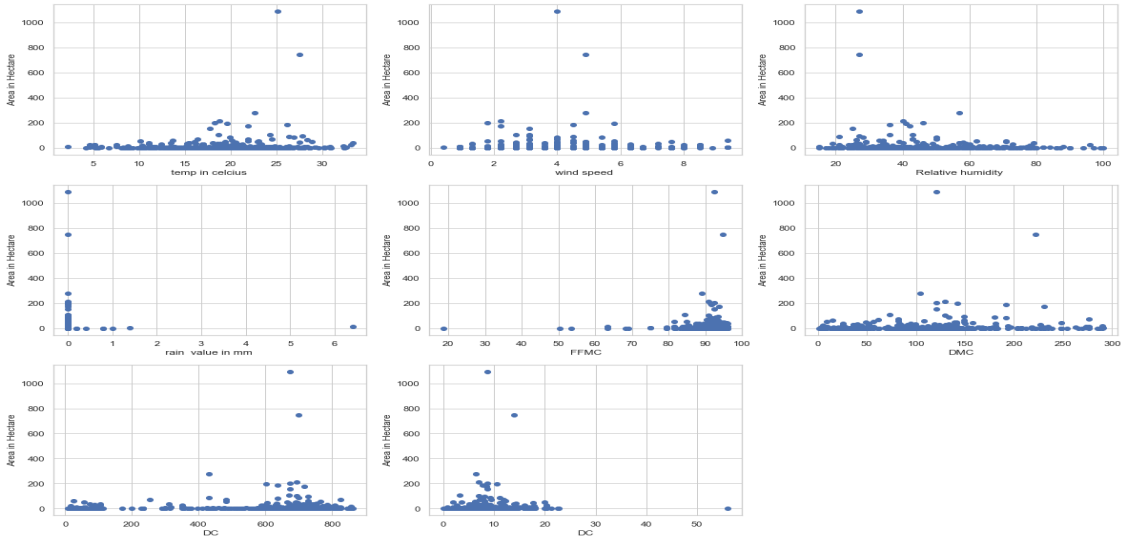

Figure 4 Scatter plots of Feature Variables Vs Area

	Model	RMSE_Train	R2_Train	RMSE_Test	R2_Test
0	Random Forest	0.559291	0.834460	1.523916	-0.056628
1	Random Forest using RandomizedSearchCV	1.342861	0.045691	1.456634	0.034615
2	Random Forest using GridSearchCV	1.352192	0.032382	1.466362	0.021677
3	Lasso	1.374632	0.000000	1.482978	-0.000620
4	Lasso using RandomizedSearchCV	1.374632	0.000000	1.482978	-0.000620
5	Lasso using GridSearchCV	1.364672	0.014439	1.470188	0.016566
6	Decision Tree	0.107154	0.993924	2.077282	-0.963317
7	Decision Tree using GridSearchCV	1.370647	0.005790	1.477236	0.007113
8	XGBoost	0.938812	0.533572	1.561262	-0.109050
9	XGBoost using GridSearchCV	0.980310	0.491427	1.487504	-0.006737
10	LightGBM	0.665730	0.765457	1.542533	-0.082602
11	LightGBM using GridSearchCV	1.344707	0.043066	1.542533	-0.007612

Table 2 Results at the end of phase 1

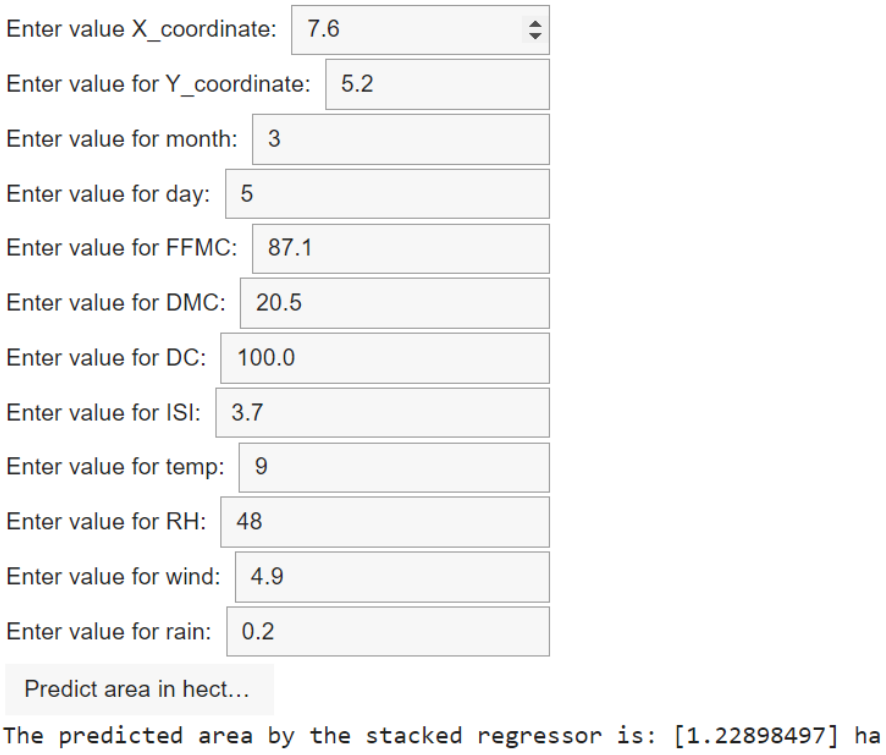

Figure 5 Sample Input and Output using GUI developed by ipywidgets

6. References

1. **Dataset:** https://archive.ics.uci.edu/ml/index.php
2. **ML Algorithms:** https://scikit-learn.org/stable/supervised_learning.html#supervised-learning
3. **Metrics:** https://scikit-learn.org/stable/modules/model_evaluation.html
4. P. Rakshit *et al*., "Prediction of Forest Fire Using Machine Learning Algorithms: The Search for the Better Algorithm," *2021 6th International Conference on Innovative Technology in Intelligent System and Industrial Applications (CITISIA)*, Sydney, Australia, 2021, pp. 1-6, doi: 10.1109/CITISIA53721.2021.9719887.
5. X. Zeng, Y. -W. Chen and C. Tao, "Feature Selection Using Recursive Feature Elimination for Handwritten Digit Recognition," *2009 Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, Kyoto, Japan, 2009, pp. 1205-1208, doi: 10.1109/IIH-MSP.2009.145.
6. B. Pavlyshenko, "Using Stacking Approaches for Machine Learning Models," *2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP)*, Lviv, Ukraine, 2018, pp. 255-258, doi: 10.1109/DSMP.2018.8478522.
7. **GitHub Repo:** https://github.com/ManasaKrishnan/DS5500---Forest-Fire-Prediction