DS5500 Phase 1: Final Report - Team 17

Forest Fire Prediction

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

1. Introduction

Uncontrolled fires that burn forests pose a great threat to the environment and habitat of wildlife as they cause disruption of transportation, communication, and water supply. Approximately 60% of such fires occur naturally and the rest are due to human activities. It is of paramount importance to address this problem as wildfires not only deteriorate the air quality but also the natural ecosystem. The dataset **(*Figure 1*)** contains 12 feature variables such as Temperature, Rain, Fire Weather Index(FWI), Moisture etc. using which we propose to build machine learning models that can predict the area (target variable) of forest cover that would be affected. These models help reduce such hazardous effects and thereby alert wildlife departments to strategize disaster management activities in advance. For this problem, we have compared various regression models and improved models' performance using hyperparameter tuning.

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. During cleaning and preprocessing, we observed the distribution of data using descriptive statistics (***Figure 3***), checked for nulls or missing values and skewness. We observed that our data has no null or missing values. The area variable seemed to show high skewness. We then performed log transformation on the area variable to reduce skewness and normalize it. As a part of EDA, we plotted a few visualizations to understand how the target variable varies with other variables as shown in ***Figure 4***. From ***Figure 5***, we can see that a lot of forest area gets affected during summer when compared to other seasons. This might be due to high temperatures and dry conditions. December also shows a significant loss in forest cover area and is something that we need to look into. We then normalized some of the feature variables using MinMaxScaler() and converted categorical variables to numerical values for better prediction and efficiency. Post train-test split, we proceeded to build 2 baseline models (Linear Regression and Ridge Regression) using RMSE and R-squared as metrics to understand our data better. The train RMSE and R2 scores were 1.35 and 0.035 respectively and the test RMSE and R2 were 1.467 and 0.021 respectively for linear regression. For the ridge regression model; train RMSE, train R2, test RMSE and test R2 were 1.35, 0.033, 1.463 and 0.026 respectively. Our baseline models did not yield the expected results as the models are less complex and there was not much room for tuning hyperparameters.

2.1 Medium Risk: After looking at the baseline results, we decided on Lasso, Decision Tree, Random Forest, XGBoost and LightGBM. The reason we chose the specific models was that Lasso has the ability to do automatic feature selection, and Decision Tree models help capture smaller details that other linear algorithms might miss. Random forest and XGBoost models on the other hand have a greater ability to handle missing values and large datasets efficiently. LightGBM results in smaller and

faster models than XGBoost. Having chosen all the models, we trained them using the training dataset and tested them. With the initial results, we also further tuned the hyperparameters for the models using Gridsearch and Randomised GridSearch. The train RMSE and R-squared for all the models are shown in ***Table 1***. The figure shows that most models have very high RMSE which is not desired. But comparatively, the decision tree model showed good metrics with lower RMSE and high R-squared scores. To see if the same was reciprocated in the test results we tested the models out and plotted the RMSE values as shown in ***Table 1***. Surprisingly the decision tree models had the highest RMSE score with respect to test data. When investigated we found that it was because of overfitting that we see such contrasting results in train and test results. Thus taking both train and test results into consideration and comparing them as shown in ***Table 1*** we found that the Random Forest tuned with RandomizedSearchCV gave the best results. Hence we have finalized that model for further stages.

Model	Train RMSE	Train R2	Test RMSE	Test R2
Random Forest	0.559	0.834	1.523	-0.056
Random Forest using RandomziedSearchCV	1.342	0.045	1.456	0.034
Random Forest using GridSearchCV	1.352	0.032	1.466	0.0216
Lasso	1.374	0.000	1.482	-0.000
Lasso using RandomziedSearchCV	1.374	0.000	1.482	-0.000
Lasso using GridSearchCV	1.364	0.014	1.470	0.016
Decision Tree	0.107	0.993	-2.077	-0.963
Decision Tree using GridSearchCV	1.370	0.005	1.477	0.007
XGBoost	0.938	0.533	1.561	-0.109
XGBoost using GridSearchCV	0.980	0.491	1.487	-0.006
LightGBM	0.665	0.765	1.542	-0.082
LightGBM using GridSearchCV	1.344	0.043	1.542	-0.007

Table 1 Results at the end of Phase 1

2.3 High Risk: As the final step in our project, we would choose the best working algorithm and try to build a Web Application/Graphical User Interface which when provided with input values, the affected area of forest in hectares will be displayed. We also noticed potential areas of improvement which may be addressed by ensemble/multi-task learning techniques with a combination of 2 or more regression models.

3. Conclusion: Future Works and Insights

From the implementation of the algorithms mentioned above, we observed potential points of failure. XGBoost and Random Forest algorithms, being complex, have a higher chance of model overfitting. Random Forest worked better with hyper-parameter tuning using RandomizedSearchCV as shown in ***Table 1*** and we were able to reduce the overfitting issue to some extent. The dataset is comparatively small and there might be imbalances in the datasets which need to be addressed. After any significant findings, we plan to extend our application towards high-risk implementation. With that, we aim to obtain a highly accurate model and a user-friendly application as this would be exceedingly useful to the Forest Fire departments to take strategic actions.

5. Figures

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
8	6	aug	sun	92.3	85.3	488	14.7	22.2	29	5.4	0	0
8	6	aug	mon	92.3	88.9	495.6	8.5	24.1	27	3.1	0	0
8	6	aug	mon	91.5	145.4	608.2	10.7	8	86	2.2	0	0
8	6	sep	tue	91	129.5	692.6	7	13.1	63	5.4	0	0
7	5	sep	sat	92.5	88	698.6	7.1	22.8	40	4	0	0
7	5	sep	sat	92.5	88	698.6	7.1	17.8	51	7.2	0	0
7	5	sep	sat	92.8	73.2	713	22.6	19.3	38	4	0	0
6	5	aug	fri	63.5	70.8	665.3	0.8	17	72	6.7	0	0
6	5	sep	mon	90.9	126.5	686.5	7	21.3	42	2.2	0	0
6	5	sep	wed	92.9	133.3	699.6	9.2	26.4	21	4.5	0	0
6	5	sep	fri	93.3	141.2	713.9	13.9	22.9	44	5.4	0	0
5	5	mar	sat	91.7	35.8	80.8	7.8	15.1	27	5.4	0	0
8	5	oct	mon	84.9	32.8	664.2	3	16.7	47	4.9	0	0
6	4	mar	wed	89.2	27.9	70.8	6.3	15.9	35	4	0	0
6	4	apr	sat	86.3	27.4	97.1	5.1	9.3	44	4.5	0	0
6	4	sep	tue	91	129.5	692.6	7	18.3	40	2.7	0	0
5	4	sep	mon	91.8	78.5	724.3	9.2	19.1	38	2.7	0	0
7	4	jun	sun	94.3	96.3	200	56.1	21	44	4.5	0	0
7	4	aug	sat	90.2	110.9	537.4	6.2	19.5	43	5.8	0	0
7	4	aug	sat	93.5	139.4	594.2	20.3	23.7	32	5.8	0	0
7	4	aug	sun	91.4	142.4	601.4	10.6	16.3	60	5.4	0	0
7	4	sep	fri	92.4	117.9	668	12.2	19	34	5.8	0	0
7	4	sep	mon	90.9	126.5	686.5	7	19.4	48	1.3	0	0
6	3	sep	sat	93.4	145.4	721.4	8.1	30.2	24	2.7	0	0

Figure 1 Sample dataset from the UCI Machine Learning Repository

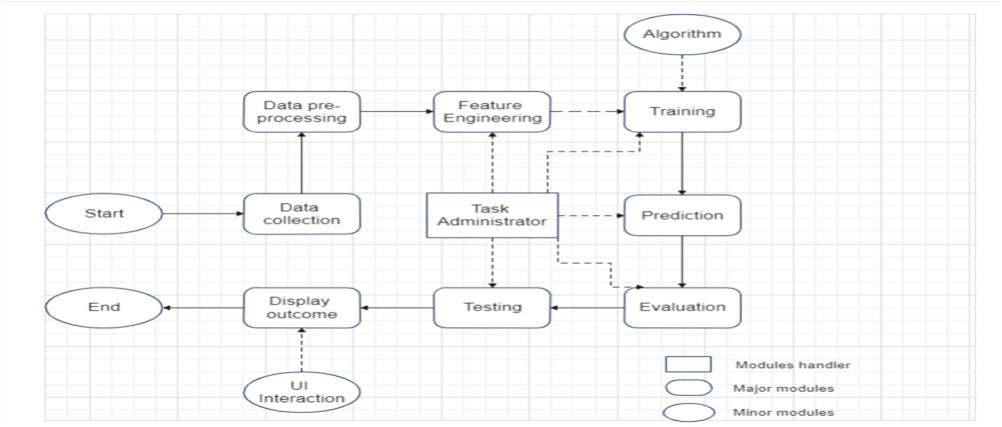

Figure 2 Machine Learning Pipeline(Model Architecture)

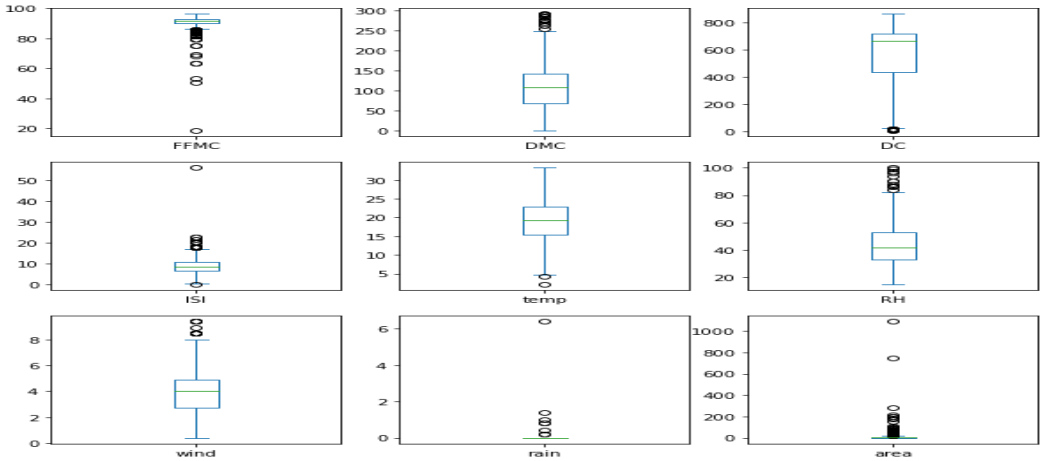

Figure 3 Boxplot showing descriptive statistics of data

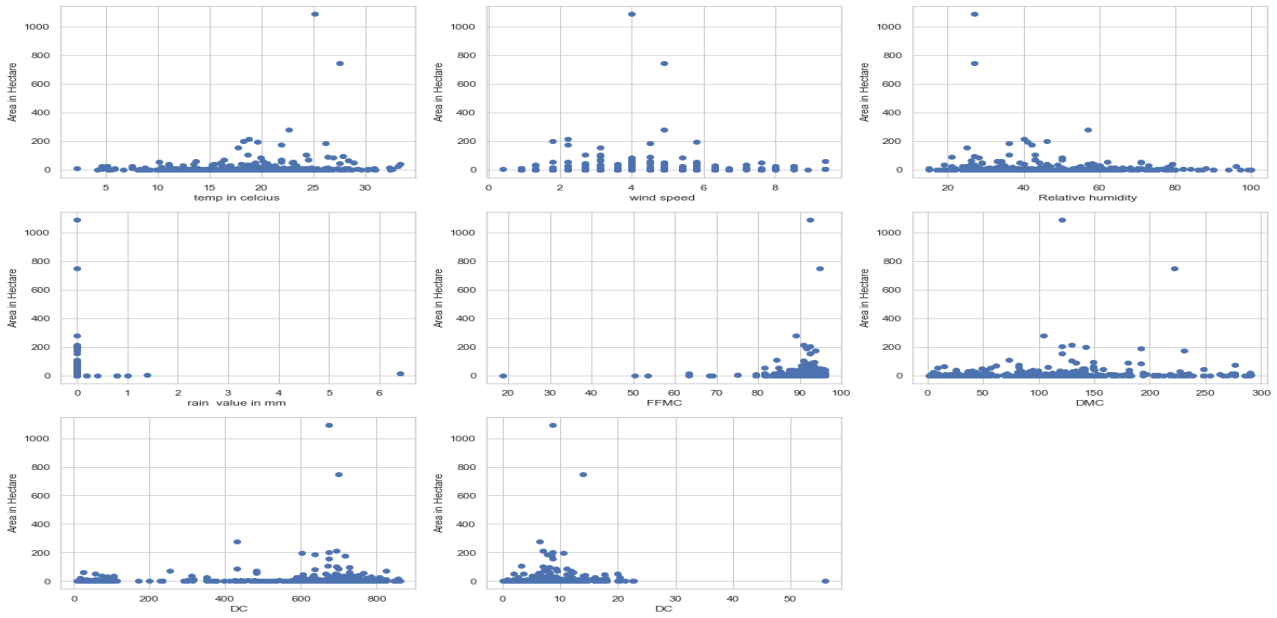

Figure 4 Scatter plot of Feature Variables Vs Area variable

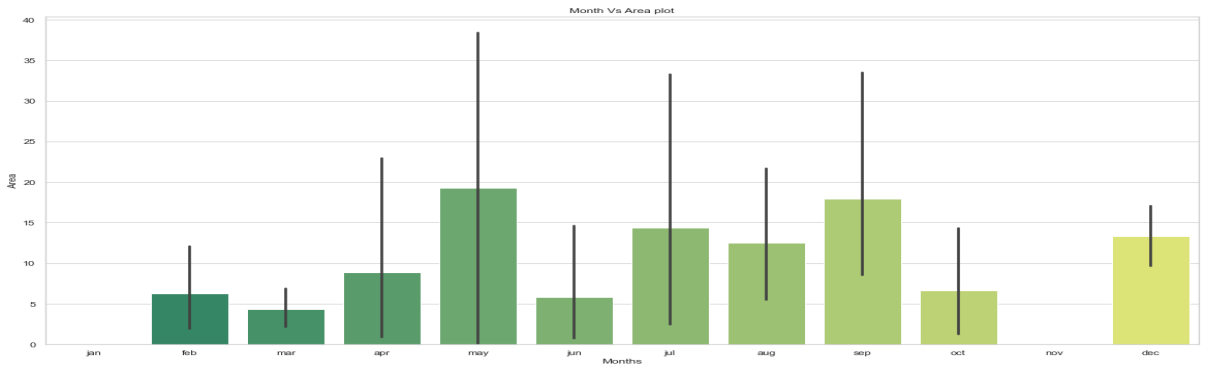

Figure 5 Plot of Month Vs Area

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. Sakr, George & Elhajj, Imad & Mitri, George & Wejinya, Uche. (2010). Artificial intelligence for forest fire prediction. IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM. 1311-1316. 10.1109/AIM.2010.5695809.
6. **GitHub Repo:** https://github.com/ManasaKrishnan/DS5500---Forest-Fire-Prediction