A Comparative Study of Forest Fire Prediction using Machine Learning models

Sachin Malego st125171 | Sila N Mahoot st125127

29 November 2024

For the fulfillment project of
AT82.01 Computer Programming for Data Science and Artificial Intelligence
Submitted to: Dr. Chantri Polprasert

Agenda

Asian Institute of Technology

Introduction

Why Are Forest Fires a Problem?

Global Tree Cover Loss (2001–2023)

- 2001-2021: The world lost approximately 437 million hectares of tree cover, representing around an **11% decrease since 2000**.

- Annual Average: On average, around **25 million hectares** of tree cover have been lost each year since 2000.

Introduction

Why Are Forest Fires a Problem?

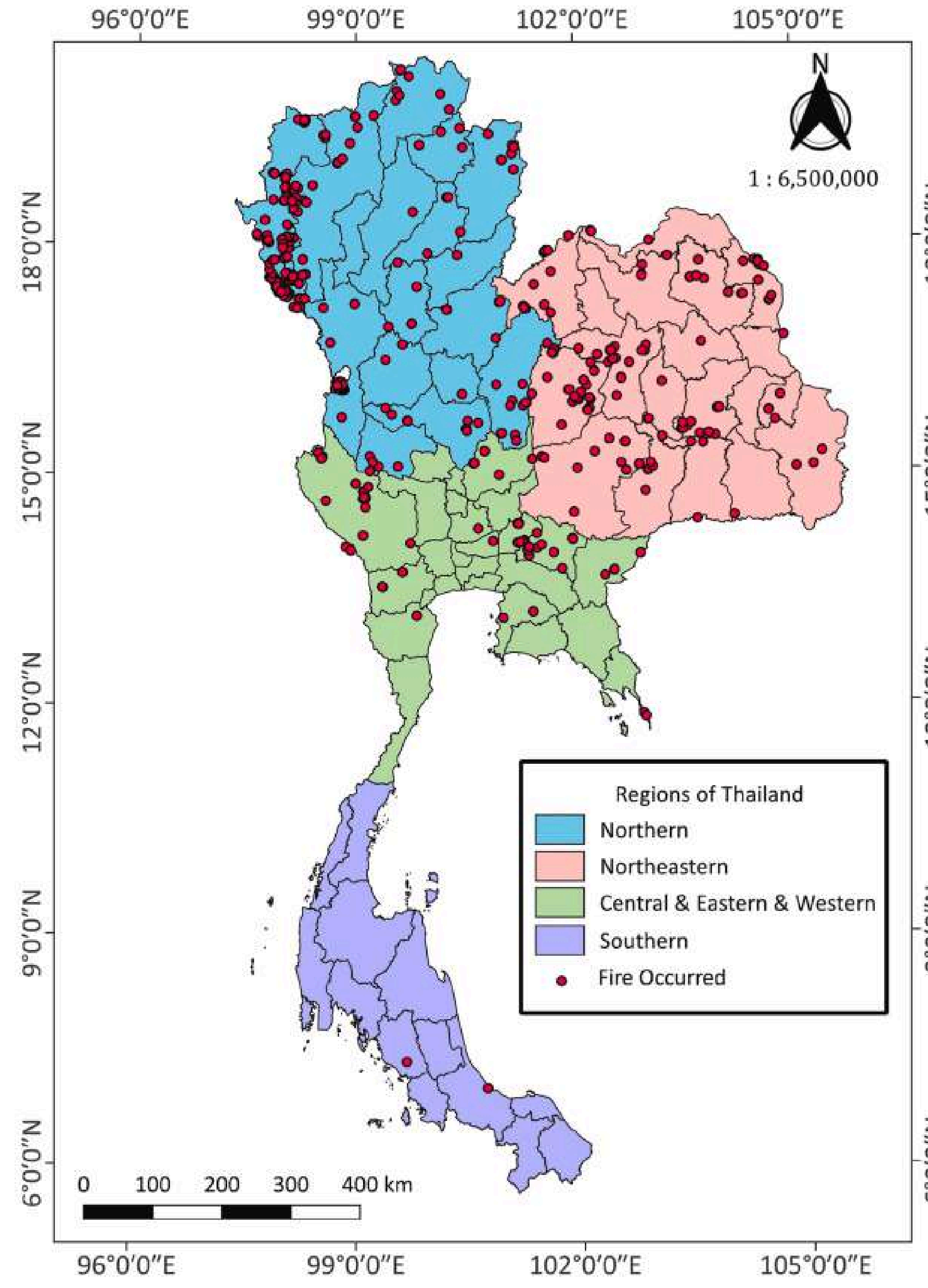
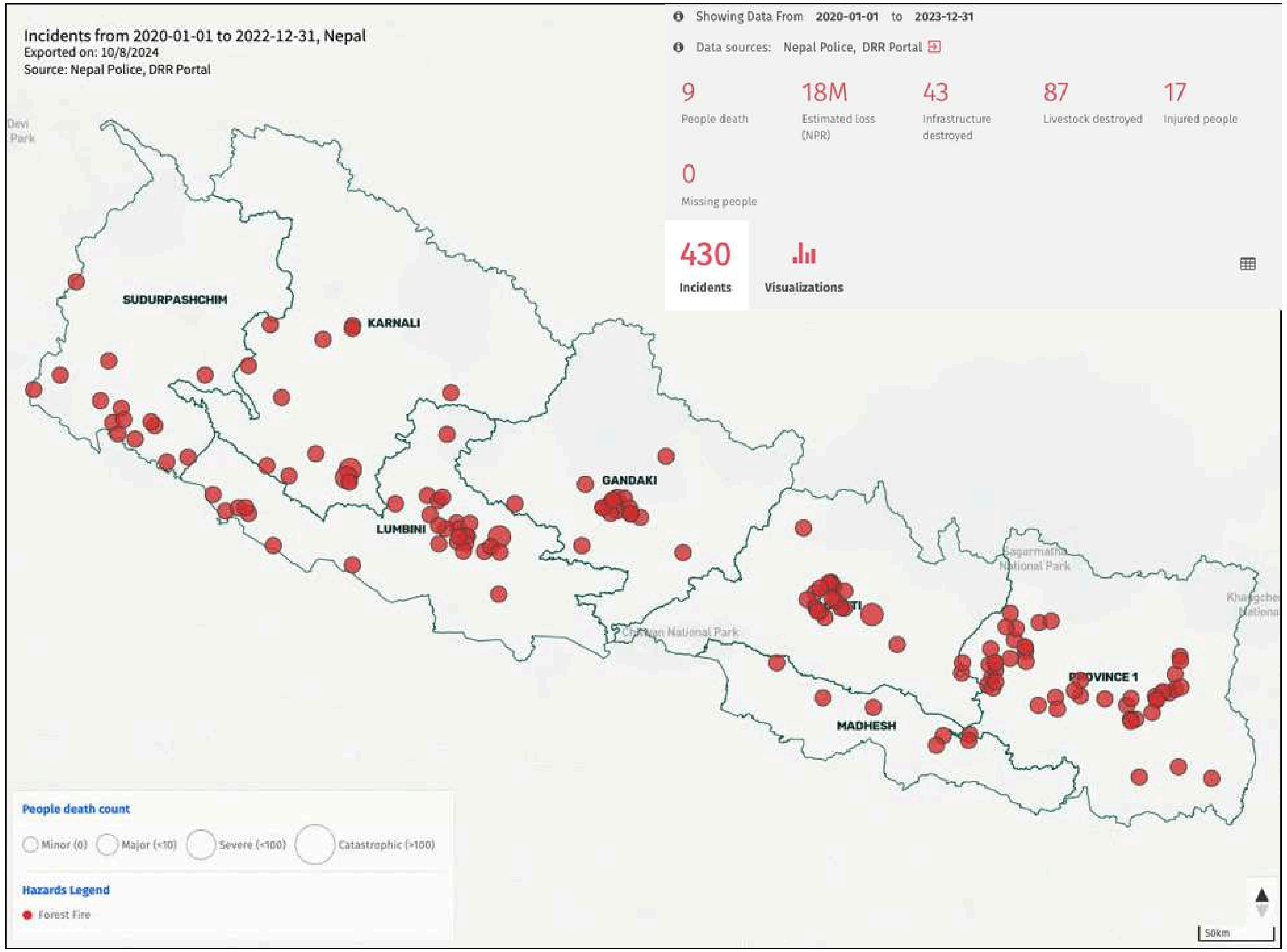


Image Source: https://doi.org/10.1016/j.heliyon.2024.e34021



Source: https://bipadportal.gov.np/incidents/

Problem Statements

Challenges in Fire Prediction

- Weather Impact Analysis
- Drought-Induced Fire Risk
- Fire Spread
- Difficulty in accurately predicting fires
- Current systems lack precision for early warnings
- Need for better environmental data handling

Related Works

- Previous studies used machine learning and satellite data.
- Our model includes climate data like El Niño for improved accuracy.

Toward a More Resilient Thailand Developing a Machine Learning-Powered Forest Fire Warning System

doi.org/10.1016/j.heliyon.2024.e34021

Developed a machine learning-powered forest fire warning system using satellite data and gas measurements. The **XGBoost model achieved 99.6% accuracy**.

Predicting Wildfires in Algerian Forests Using Machine Learning Models

doi.org/10.1016/j.heliyon.2023.e18064

Used PCA for reducing data complexity and developed an **ANN for predicting wildfires, achieving an accuracy of 96.7%**. It highlighted key features like relative humidity and drought code. The dataset included various weather features collected from Algeria.

Comparison of Forest Fire Prediction System Using Machine Learning Algorithms

doi.org/10.1109/ICACITE57410.2023.10182818

Compared several machine learning models, including logistic regression and random forest, to predict forest fires. It discussed the strengths and weaknesses of each model. The dataset included temperature, wind speed, and humidity, and the authors suggested **integrating climate patterns like El Niño for better predictions.**

Datasets

Data Sources Used

Forest Fire Occurrences
in Algeria (2012) and Portugal (2017

Historical Weather Data
from Meteostat & Weather Underground

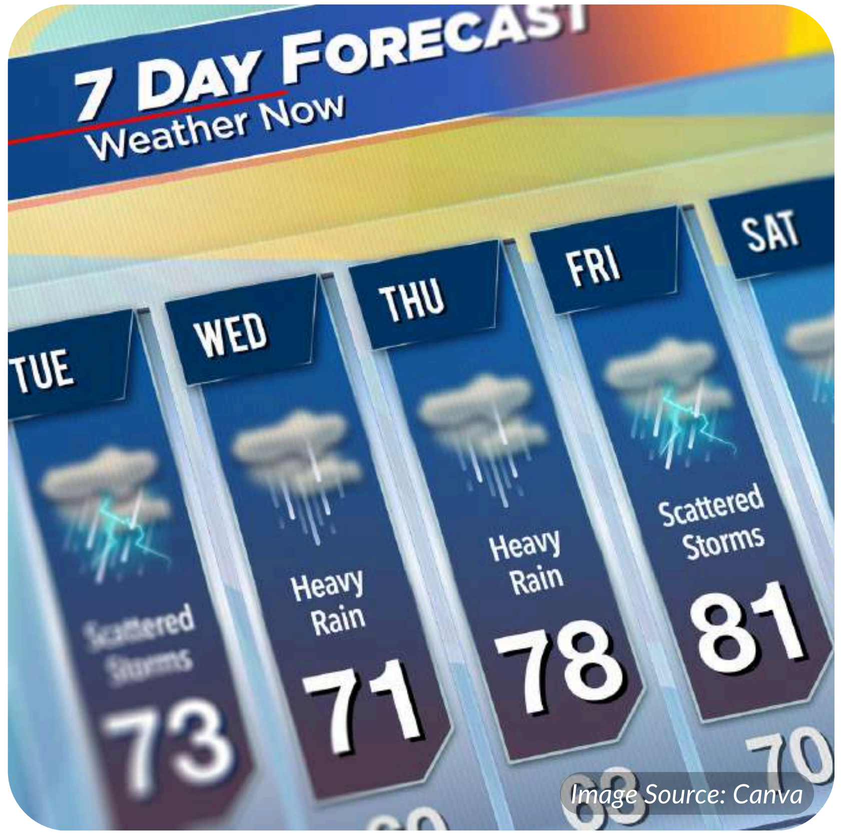


Image Source: Canva

Sea Surface Temperature
from Climate Prediction Center

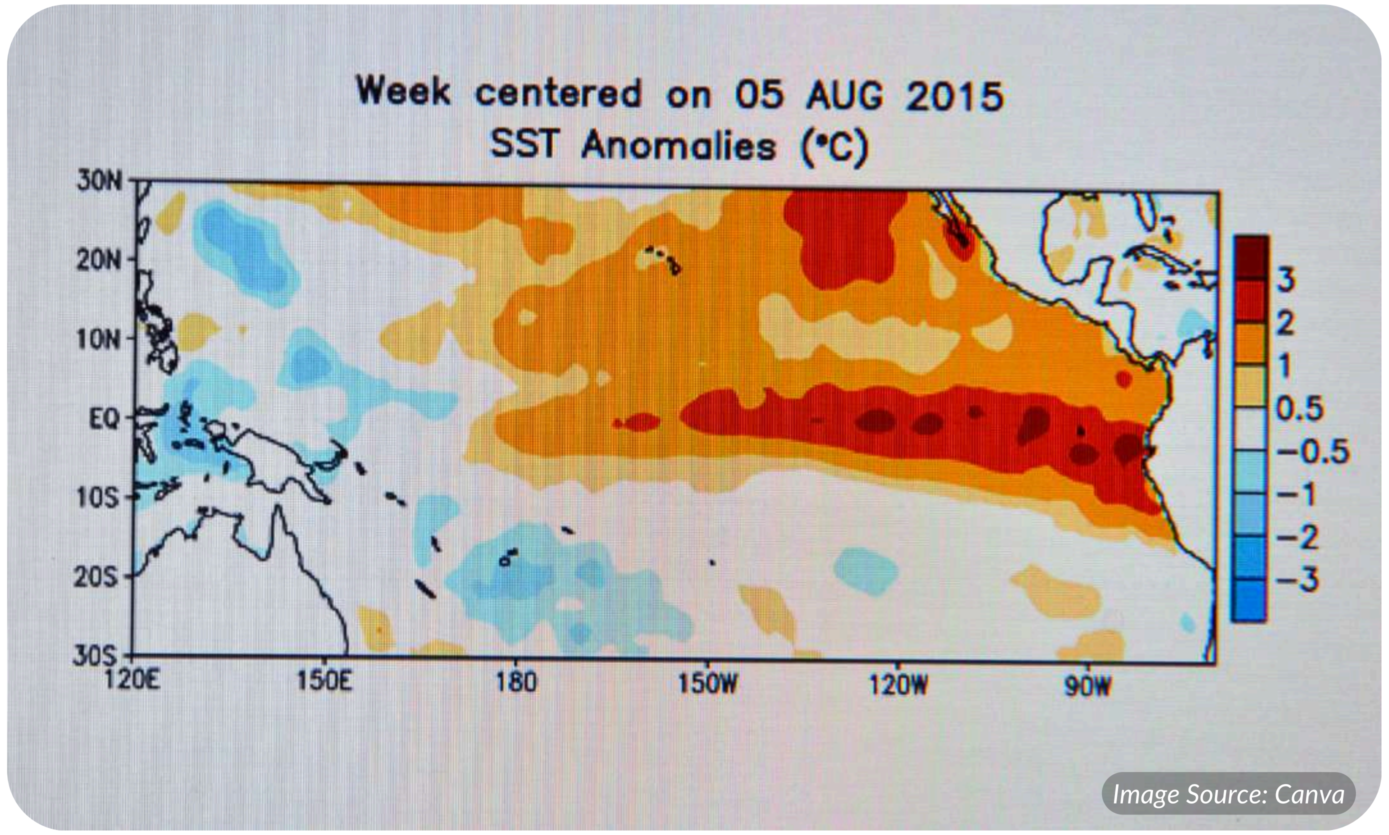


Image Source: Canva

Datasets

Data Sources Used

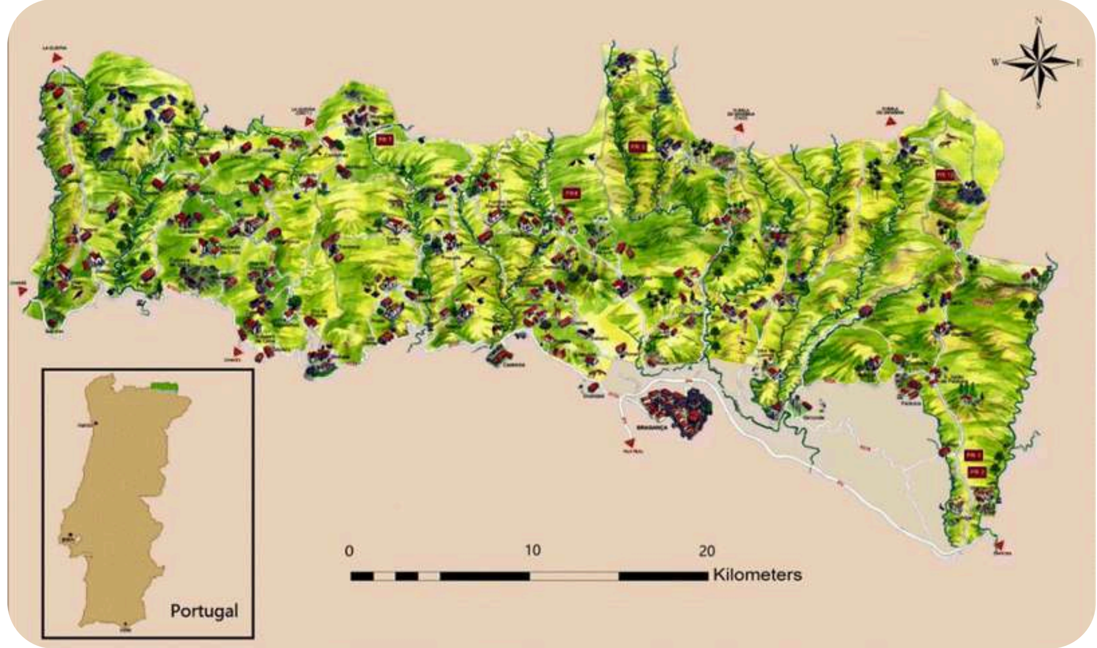
Forest Fire Occurrences in Algeria (2012) and Portugal (2017)

Variables	Description
X	*X*-axis spatial coordinate (from 1 to 9)
Y	*Y*-axis spatial coordinate (from 1 to 9)
Month	Month of the year (from "January" to "December")
Day	Day of the week (from "Monday" to "Sunday")
FFMC	FFMC code from the FWI system (from 18.7 to 96.20)
DMC	DMC code from the FWI system (from 1.1 to 291.3)
DC	DC code from the FWI system (from 7.9 to 860.6)
ISI	ISI code from the FWI system (from 0 to 56.10)
Temp	Temperature in degrees Celsius (from 2.2 to 33.30)
RH	Relative humidity in percentage (from 15.0 to 100)
Wind	Wind speed in km/h (from 0.40 to 9.40)
Rain	Outside rain in mm/m^2 (from 0.0 to 6.40)
Area	Total burned area of the forest (in *ha*) (from 0.00 to 1090.84)

#	Column	Non-Null Count	Dtype
0	day	243 non-null	int32
1	month	243 non-null	int32
2	year	243 non-null	int32
3	Temperature	243 non-null	int32
4	RH	243 non-null	int32
5	Ws	243 non-null	int32
6	Rain	243 non-null	float64
7	FFMC	243 non-null	float64
8	DMC	243 non-null	float64
9	DC	243 non-null	float64
10	ISI	243 non-null	float64
11	BUI	243 non-null	float64
12	FWI	243 non-null	float64
13	Classes	243 non-null	object
14	Region	243 non-null	int32

Features: Date, Temperature, RH, WindSpeed, Rain, Precipitation, FFMC, DMC, DC, FWI, Burn Area

Data Quality Checklist: Completeness, Accuracy, Documentation, Anomaly Detection, others



Datasets

Data Sources Used

Historical Weather Data
from Meteostat & Weather Underground

date	temp_min	temp_avg	temp_max	dew_min	dew_avg	dew_max	hum_min	hum_avg	hum_max	wind_speed_min
2012-01-01	18.888889	11.000000	7.222222	10.000000	7.555556	2.777778	94	80.6	45	22.53076
2012-01-02	20.000000	13.111111	7.777778	12.222222	8.333333	6.111111	93	74.0	49	22.53076
2012-01-03	16.111111	13.222222	8.888889	10.000000	8.777778	7.777778	100	76.7	59	19.31208
2012-01-04	17.222222	11.111111	7.222222	10.000000	7.666667	6.111111	94	80.6	52	22.53076
2012-01-05	18.888889	12.055556	8.888889	12.777778	9.055556	7.222222	94	83.2	52	14.48406

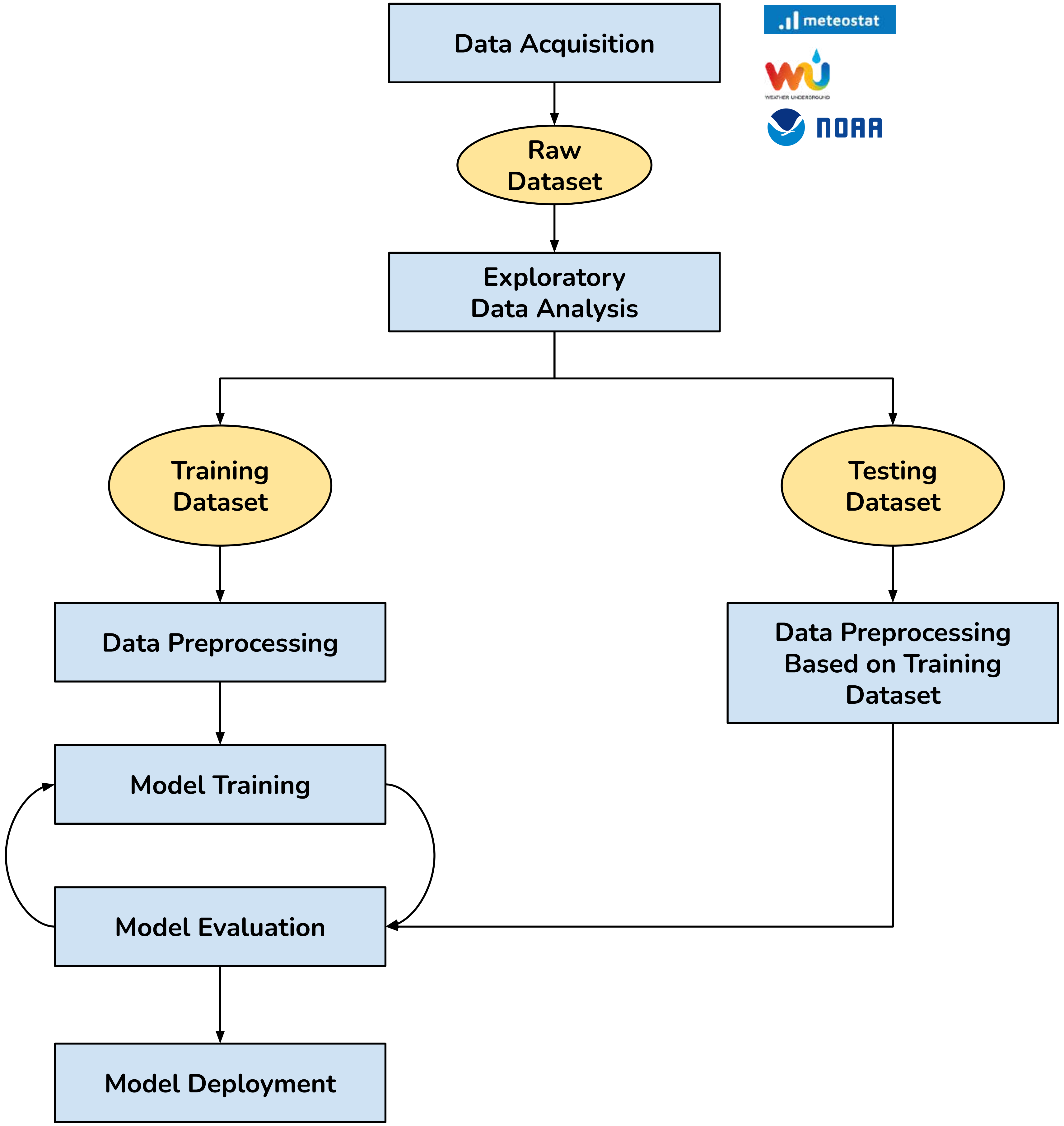
Sea Surface Temperature
from Climate Prediction Center

	nino12_sst	nono12_ssta	nino3_sst	nino3_ssta	nino34_sst	nino34_ssta	nino4_sst	nino4_ssta
1981-09-02	0.186275	0.022222	0.333333	0.030303	0.421053	0.066667	0.512195	0.15
1981-09-03	0.186275	0.022222	0.333333	0.030303	0.421053	0.066667	0.512195	0.15
1981-09-04	0.186275	0.022222	0.333333	0.030303	0.421053	0.066667	0.512195	0.15
1981-09-05	0.186275	0.022222	0.333333	0.030303	0.421053	0.066667	0.512195	0.15
1981-09-06	0.186275	0.022222	0.333333	0.030303	0.421053	0.066667	0.512195	0.15
1981-09-07	0.186275	0.022222	0.333333	0.030303	0.421053	0.066667	0.512195	0.15
1981-09-08	0.186275	0.022222	0.333333	0.030303	0.421053	0.066667	0.512195	0.15
1981-09-09	0.137255	0.133333	0.317460	0.060606	0.421053	0.066667	0.536585	0.10
1981-09-10	0.137255	0.133333	0.317460	0.060606	0.421053	0.066667	0.536585	0.10
1981-09-11	0.137255	0.133333	0.317460	0.060606	0.421053	0.066667	0.536585	0.10

Threshold for SST/SSTA (NINO 3.4): The most recent three-month average for the area is computed, and if the region is more than **0.5 °C** (**0.9 °F**) above (or below) normal for that period, then an El Niño (or La Niña) is considered in progress.

Methodology

How We Build Our Model



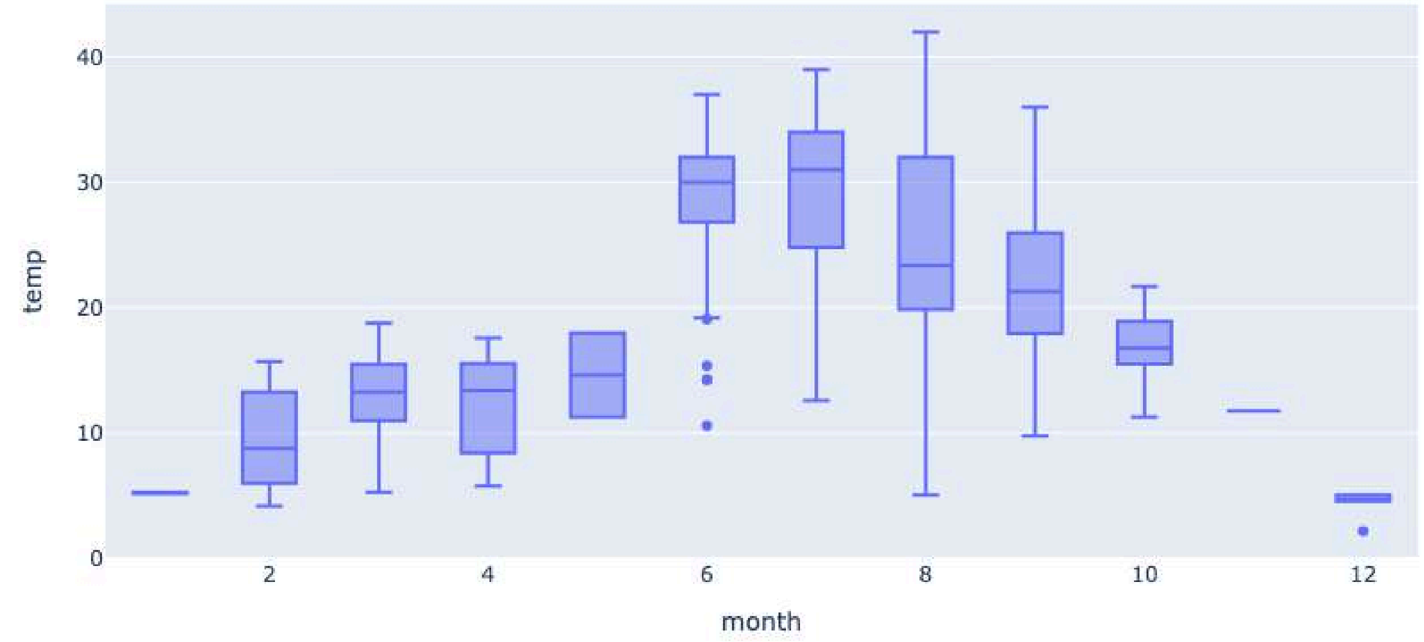
Missing Values
Data Scaling
Data Imbalance

Random Forest
Gradient Boosting
Extreme Gradient Boosting

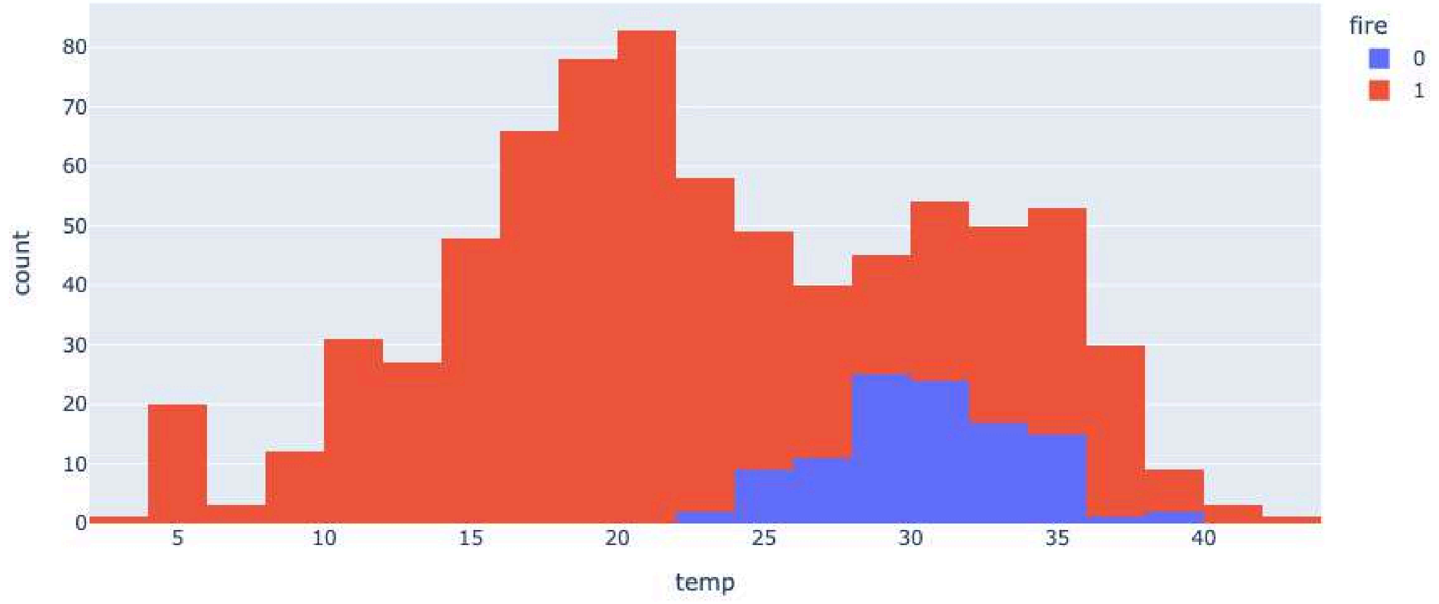
Exploratory Data Analysis

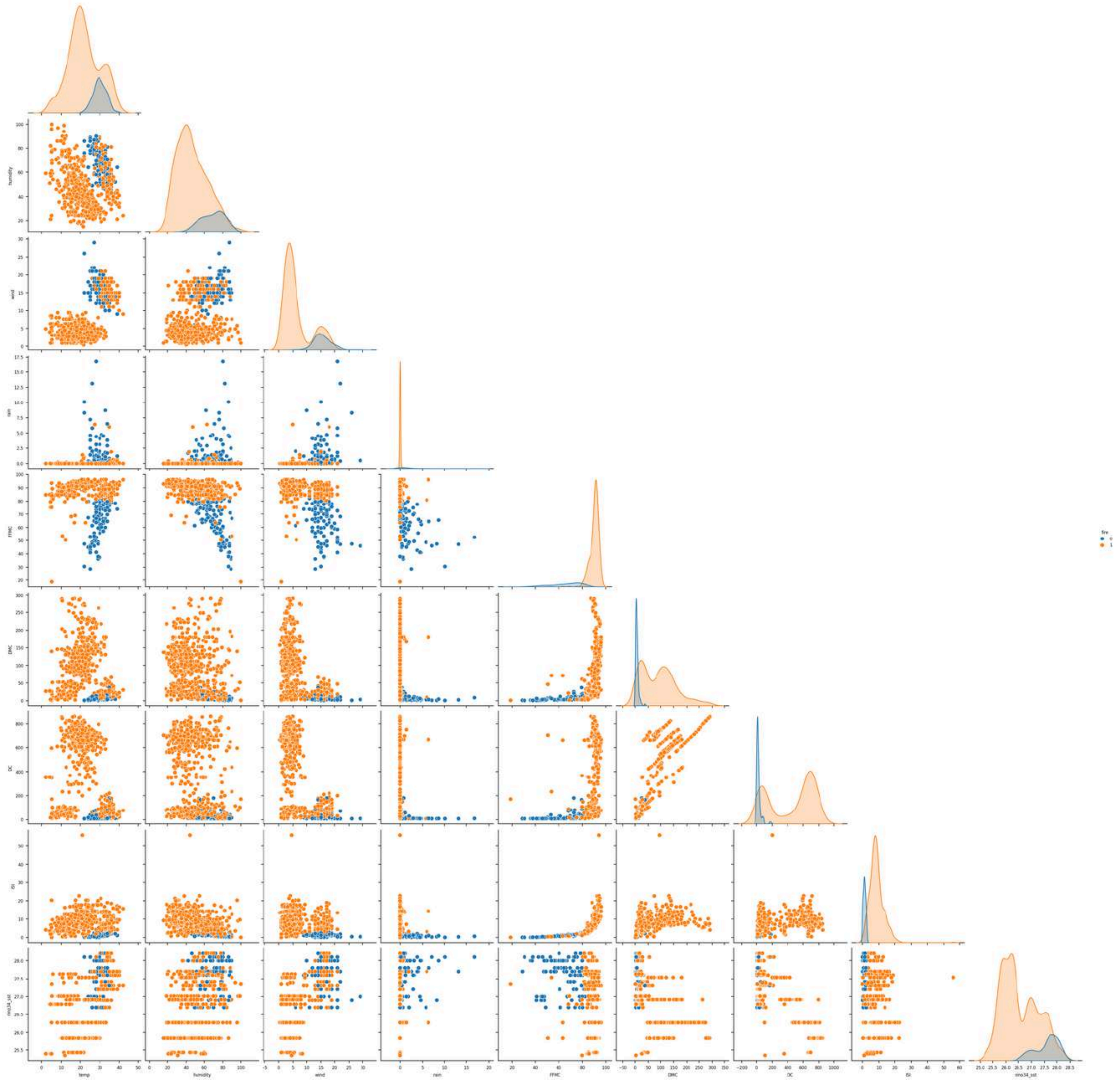
Exploring the Data

Temperature distribution over months



Temperature Vs Fire count



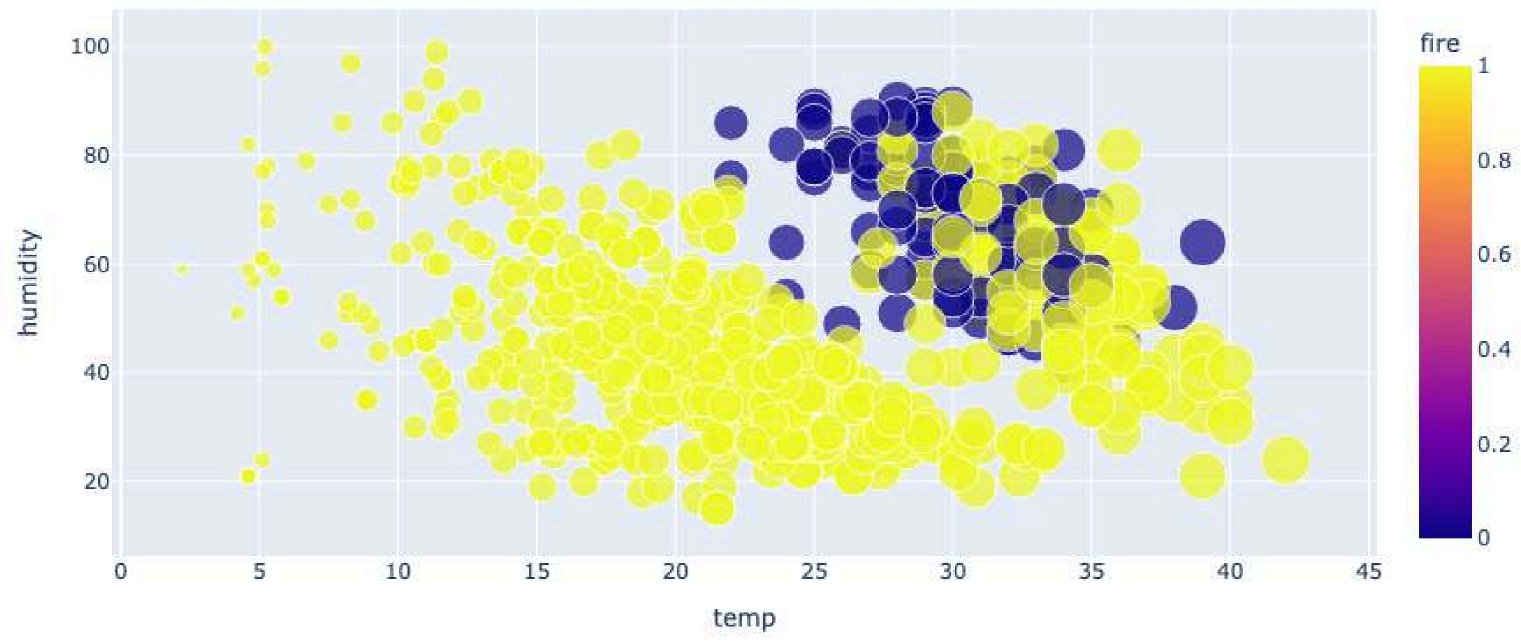


Pairplot between different features

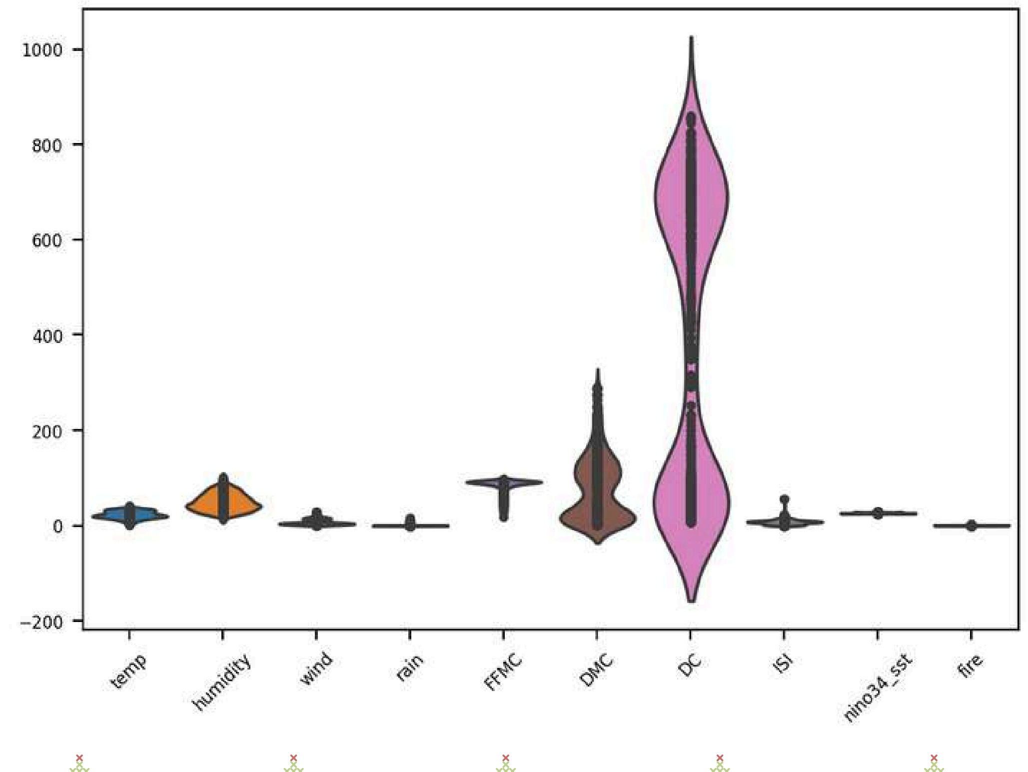
Exploratory Data Analysis

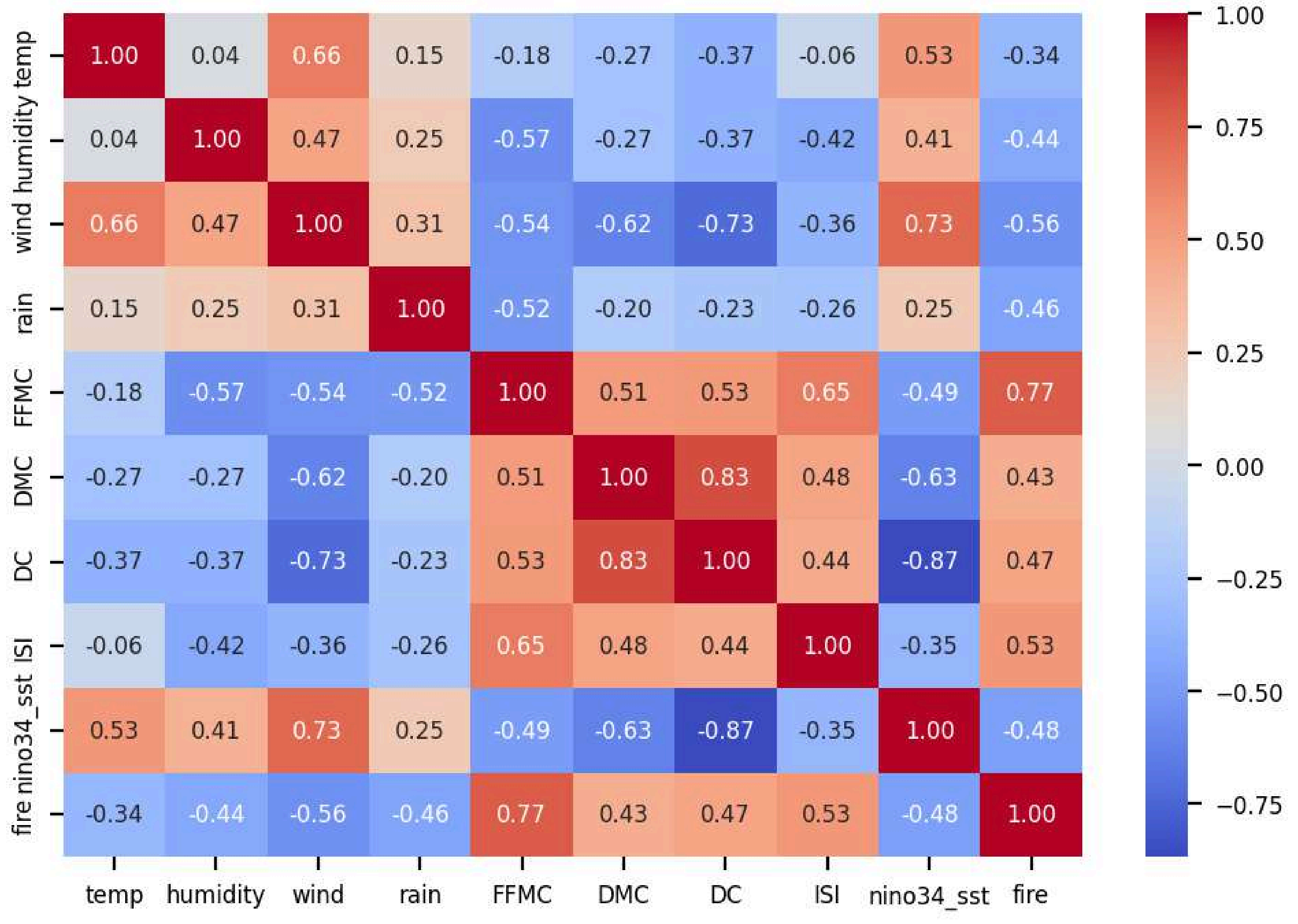
Exploring the Data

Temperature Vs Humidity with fire occurrences



Violin distribution plot of different features



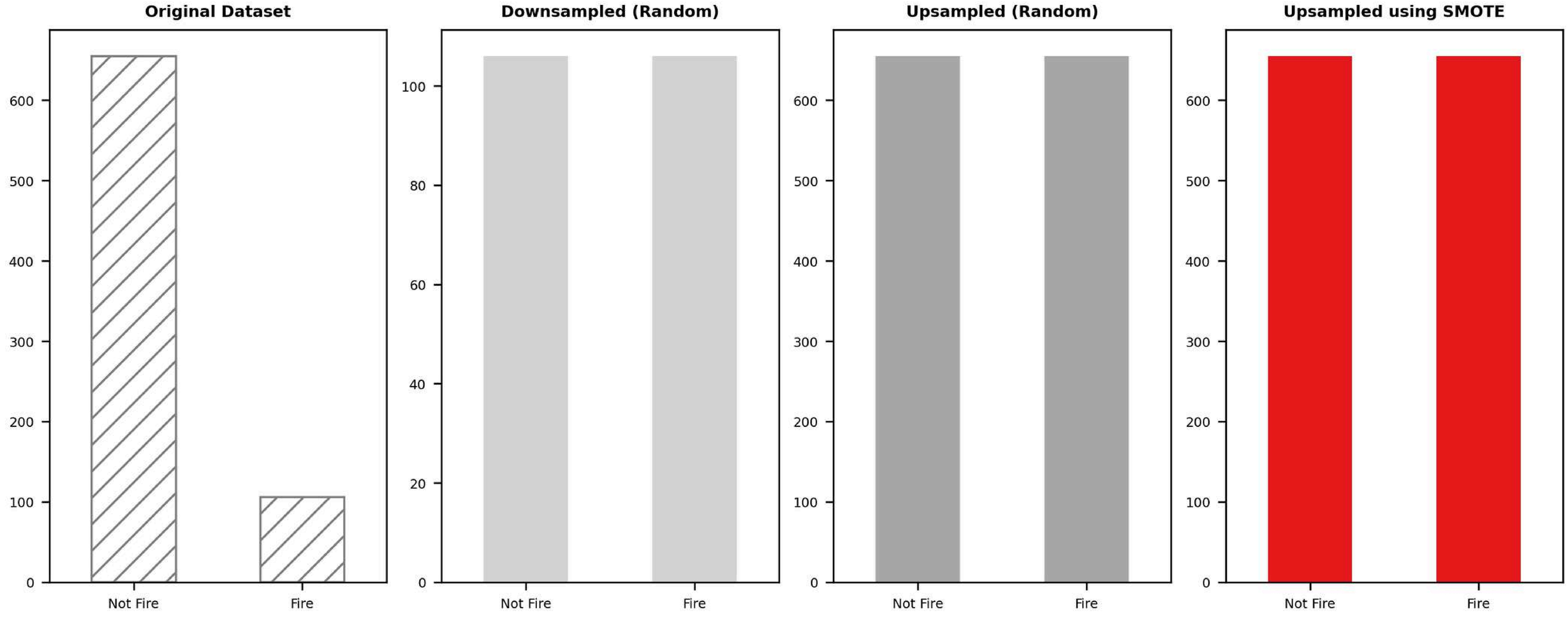


Multicolinearity Heatmap

Data Pre-processing

Exploring the Data

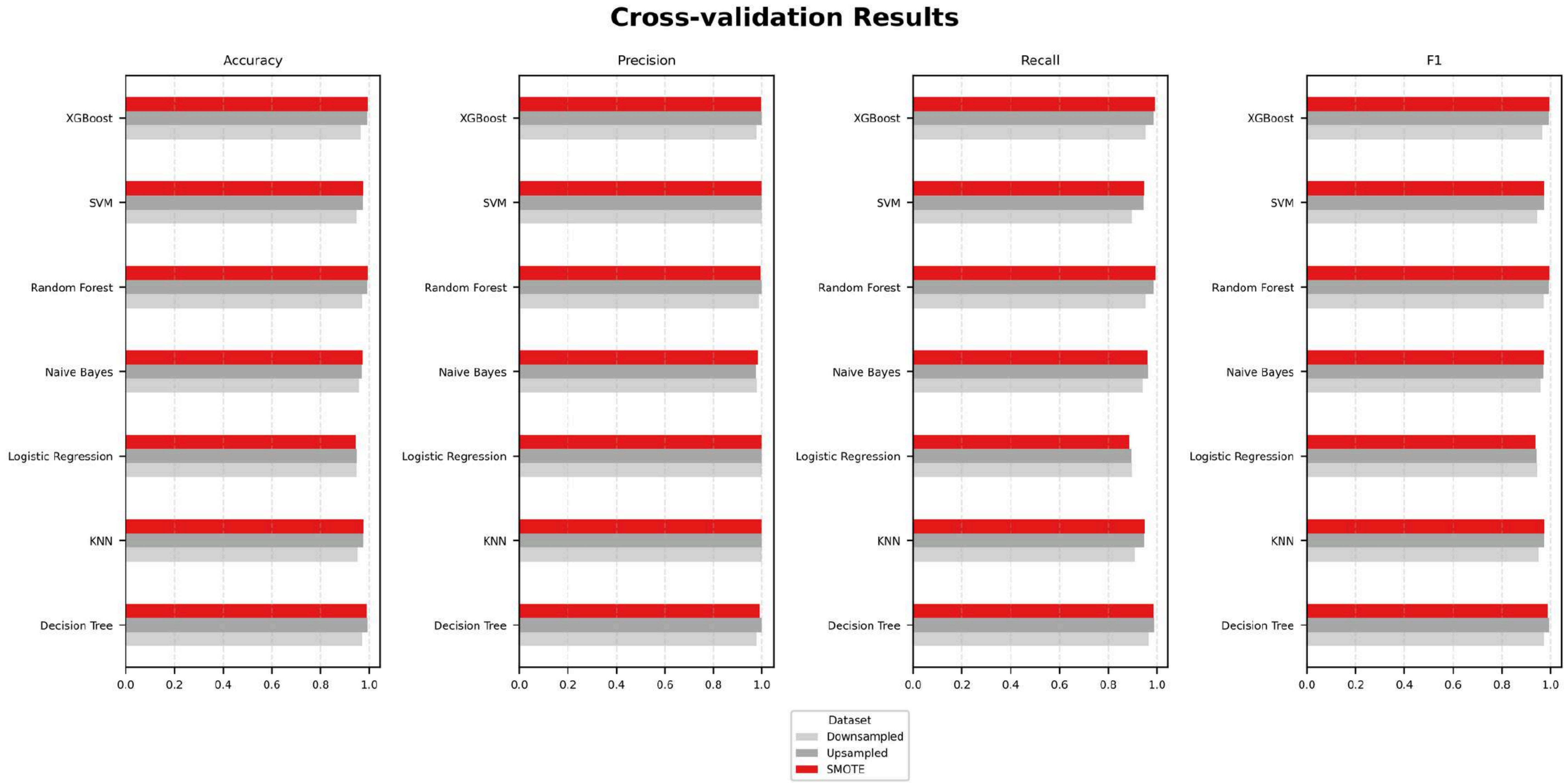
Using Downsampling, Upsampling and SMOTE upsampling to mitigate data imbalance



Machine Learning Model

Exploring the Data

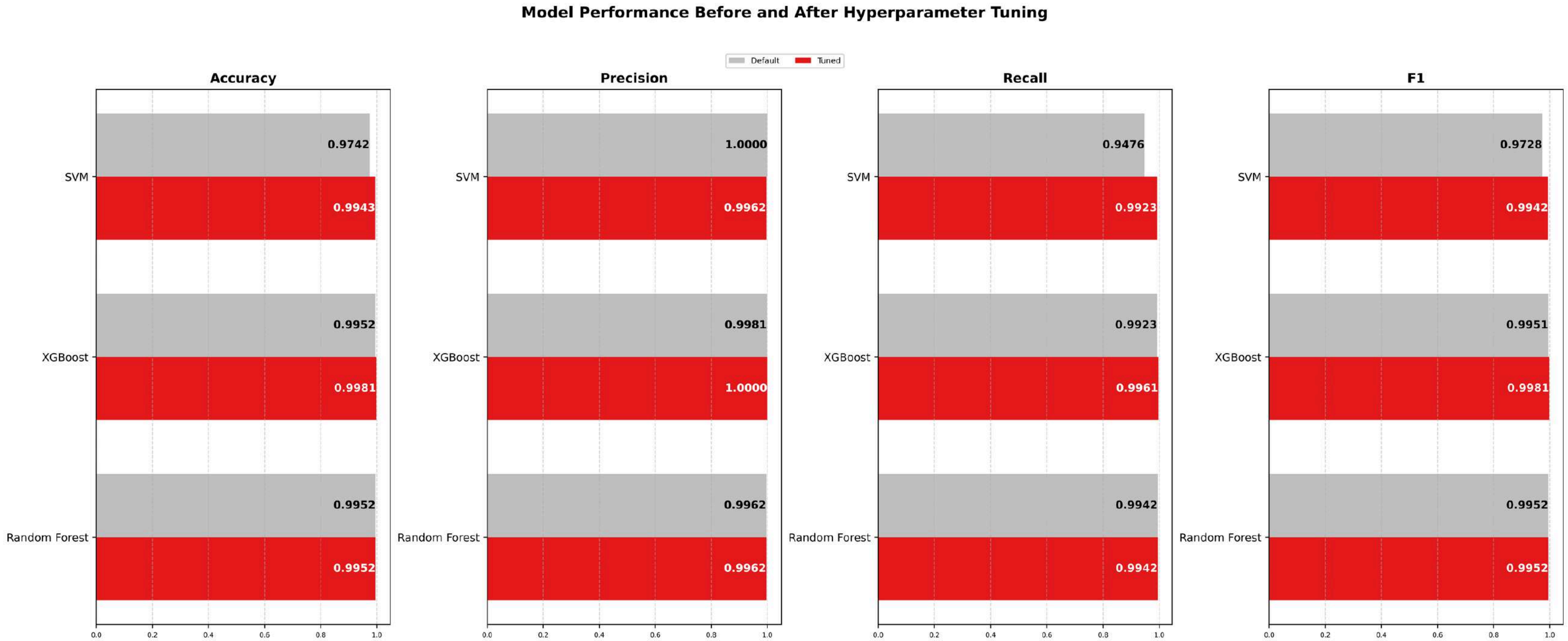
Each model was trained and validated using a combination of cross-validation techniques to ensure robustness and reduce the risk of overfitting. Performance metrics such as Accuracy, Precision, Recall, and F1-score for classification scenarios were used to evaluate the models.



Machine Learning Model

Exploring the Data

Each model was trained and validated using a combination of cross-validation techniques to ensure robustness and reduce the risk of overfitting. Performance metrics such as Accuracy, Precision, Recall, and F1-score for classification scenarios were used to evaluate the models.



Machine Learning Model

Exploring the Data

- Initial results revealed that **while Linear Regression offered insights into general trends, it was insufficient for capturing the intricate, non-linear, and interactive relationships inherent in forest fire datasets**.

- **Decision Trees provided better interpretability but were prone to overfitting** without additional constraints. Similarly, **Random Forest and XGBoost showed strong predictive capabilities; however, they were resource-intensive and required extensive tuning.**

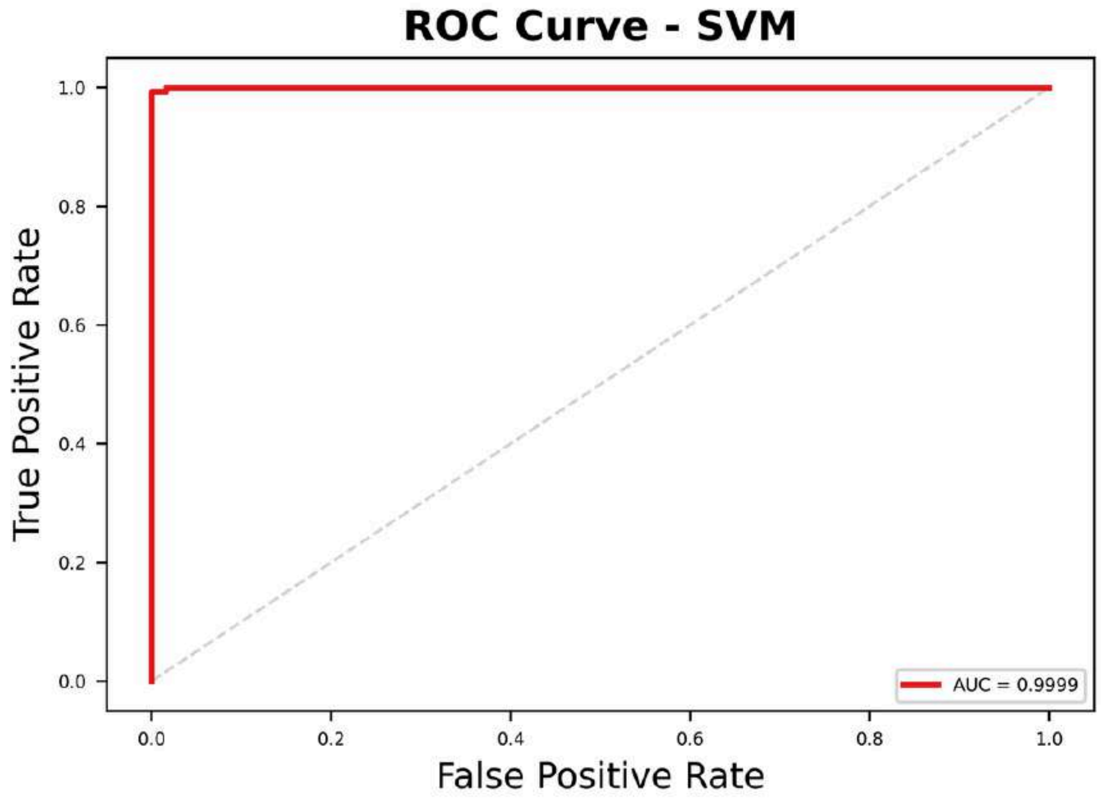
- Based on our analysis, **SVM emerged as the most suitable model for this scenario due to the following reasons**:
    1. Simplicity and Interpretability
    2. Less Tuning Needed
    3. Efficient and Fast
    4. Lower Risk of Overfitting
    5. Good Generalization

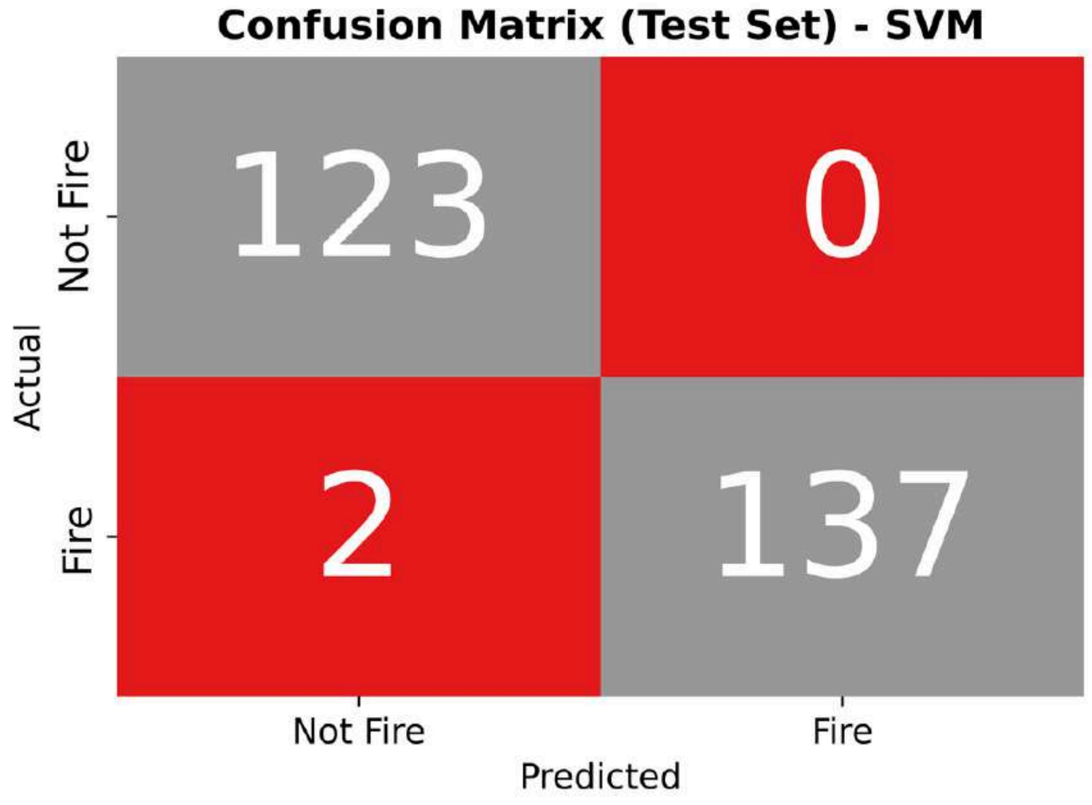
Default hyperparameters

	Accuracy	Precision	Recall	F1
Random Forest	0.995229	0.996154	0.994212	0.995150
XGBoost	0.995224	0.998058	0.992252	0.995141
SVM	0.974204	1.000000	0.947647	0.972771

Best hyperparameters

	Accuracy	Precision	Recall	F1
Random Forest	0.996186	0.998077	0.994212	0.996116
XGBoost	0.997134	0.998077	0.996135	0.997092
SVM	0.994267	0.996190	0.992270	0.994174





Pipelining

Exploring the Data

```python
# Define the classifiers (default hyperparameters)
classifiers = {
    'Logistic Regression': LogisticRegression(),
    'Random Forest': RandomForestClassifier(),
    'SVM': SVC(),
    'KNN': KNeighborsClassifier(),
    'Decision Tree': DecisionTreeClassifier(),
    'Naive Bayes': GaussianNB(),
    'XGBoost': XGBClassifier()
}

# Define the pipeline
pipeline = Pipeline([
    ('scaler', MinMaxScaler()),
    ('classifier', None)
])

# Perform cross-validation
cv_results = {}

kf = KFold(n_splits=5, shuffle=True, random_state=42)

for name, clf in classifiers.items():
    pipeline.set_params(classifier=clf)
    scores_downsampled = cross_validate(pipeline, X_train_downsampled, y_train_downsampled, cv=5, scoring=['accuracy', 'precision', 'recall', 'f1'])
    scores_upsampled = cross_validate(pipeline, X_train_upsampled, y_train_upsampled, cv=5, scoring=['accuracy', 'precision', 'recall', 'f1'])
    scores_smote = cross_validate(pipeline, X_train_smote, y_train_smote, cv=5, scoring=['accuracy', 'precision', 'recall', 'f1'])
    cv_results[(name, 'Downsampled')] = {
            'Accuracy': scores_downsampled['test_accuracy'].mean(),
            'Precision': scores_downsampled['test_precision'].mean(),
            'Recall': scores_downsampled['test_recall'].mean(),
            'F1': scores_downsampled['test_f1'].mean()
        }
    cv_results[(name, 'Upsampled')] = {
            'Accuracy': scores_upsampled['test_accuracy'].mean(),
            'Precision': scores_upsampled['test_precision'].mean(),
            'Recall': scores_upsampled['test_recall'].mean(),
            'F1': scores_upsampled['test_f1'].mean()
        }
    cv_results[(name, 'SMOTE')] = {
            'Accuracy': scores_smote['test_accuracy'].mean(),
            'Precision': scores_smote['test_precision'].mean(),
            'Recall': scores_smote['test_recall'].mean(),
            'F1': scores_smote['test_f1'].mean()
        }
```

Pipelining

Exploring the Data

```python
# Evaluate default hyperparameters for later comparison
cv_results_default = {}

for name, clf in classifiers.items():
    pipeline.set_params(classifier=clf)
    scores = cross_validate(pipeline, X_train_smote, y_train_smote, cv=5, scoring=['accuracy', 'precision', 'recall', 'f1'])
    cv_results_default[name] = {
            'Accuracy': scores['test_accuracy'].mean(),
            'Precision': scores['test_precision'].mean(),
            'Recall': scores['test_recall'].mean(),
            'F1': scores['test_f1'].mean()
        }
```

```python
# Perform hyperparameter tuning
best_models = {}

for name, clf in classifiers.items():
    pipeline.set_params(classifier=clf)
    grid_search = GridSearchCV(pipeline, param_grid=params[name], cv=5, scoring='f1', n_jobs=-1)
    grid_search.fit(X_train_smote, y_train_smote)
    best_models[name] = grid_search.best_estimator_
    print(f'{name} best hyperparameters: {grid_search.best_params_}')

# Compare the best models using cross-validation
cv_results_best = {}

for name, clf in best_models.items():
    pipeline.set_params(classifier=clf)
    scores = cross_validate(pipeline, X_train_smote, y_train_smote, cv=5, scoring=['accuracy', 'precision', 'recall', 'f1'])
    cv_results_best[name] = {
            'Accuracy': scores['test_accuracy'].mean(),
            'Precision': scores['test_precision'].mean(),
            'Recall': scores['test_recall'].mean(),
            'F1': scores['test_f1'].mean()
        }

# Put the results in a DataFrame
cv_results_best = pd.DataFrame(cv_results_best).T
```

Discussion

Introduction

- Study aimed to investigate and generalize a model with the potential comparison of machine learning models in predicting forest fires specifically with dataset from Portugal and Algeria.
- Our analysis indicated that the Support Vector Machine (SVM) model significantly generalized better than other machine learning algorithms, including Linear Regression, Decision Trees, Random Forest, and XGBoost, with respect to accuracy, precision, and recall metrics.
- By implementing machine learning models like SVM into existing early warning systems, forest managers can allocate resources more efficiently and improve response times during critical wildfire season.
- Our results align with previous research that has explored the use of machine learning for forest fire prediction. This consistency across different geographic contexts suggests that machine learning techniques can serve as a universal tool in forest fire prediction.
- Our findings complement the work of Zaidi (2023), which **highlighted the utility of machine learning in predicting wildfires in the Algerian landscape**, reinforcing the notion that such **methodologies can be adapted to diverse ecological environments**.

Discussion

Issues and Challenges

- The major challenges in the application development are **acquiring additional datasets** for testing and **integrating an interactive map**. The **lack of diverse, high-quality datasets** is a significant hurdle, as more data is needed to ensure the model's accuracy across different regions, conditions, and environmental factors.
- Integrating an interactive map that displays fire predictions and environmental data is complex, however **the application was able to integrate the map with fire occurrence data**.

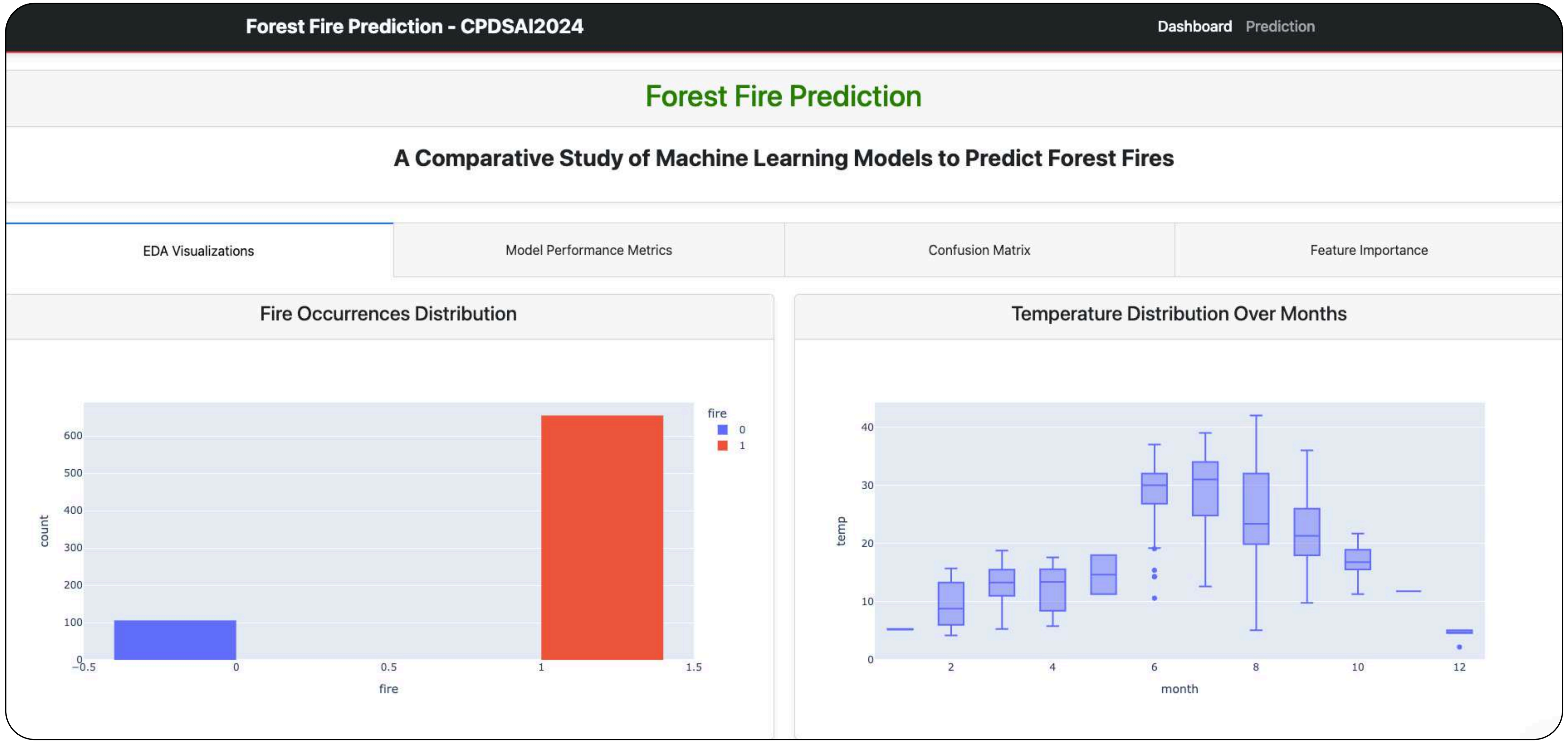
Discussion

Limitations and Future Research Dimensions

- The analysis was **constrained to specific geographic regions**, which **may limit the generalizability of the findings**.
- Future studies **should aim to validate these results across different climates and ecosystems** to assess the robustness of the SVM model.
- The dataset utilized did not account for certain variables, such as **human-induced factors like arson or land-use changes since lack of data**, which can significantly influence fire occurrence. Future research should incorporate these elements to enhance the predictive accuracy of machine learning models.
- Future research **could explore the incorporation of additional data sources, such as satellite imagery and real-time environmental sensors**, to augment the predictive capabilities of the models.
- Integrating **human activity data** could provide a more holistic understanding of fire dynamics, allowing for improved model performance.
- Integration of **Real-time API** can enhance predictive systems reliability.

Deployment and Conclusion

Deploying the Model



Link to app: http://ec2-98-84-247-124.compute-1.amazonaws.com:8050

Python Dash Plotly AWS Web App

Thank You!