A Course Based Project Report on

FOREST FIRE PREDICTION

Submitted to the

Department of CSE-(CyS, DS) and AI&DS

in partial fulfilment of the requirements for the completion of course

Models for Data Science LABORATORY(22PC2DS301)

BACHELOR OF TECHNOLOGY

IN

CSE-Data Science

Submitted by

P.SHASHANK	23071A6745
P.SREEJA REDDY	23071A6746
P.ABHISHEK	23071A6747
P.NIRANJAN	23071A6748

Under the guidance of

Mrs.N.Madhuri

Assistant Professor

Department of CSE-(CyS, DS) and AI&DS

VALLURUPALLI NAGESWARA RAO VIGNANA JYOTHI INSTITUTE OF ENGINEERING & TECHNOLOGY

An Autonomous Institute, NAAC Accredited with 'A++' Grade, NBA

VignanaJyothi Nagar, Pragathi Nagar, Nizampet (S.O), Hyderabad – 500 090, TS, India

November - 2025

VALLURUPALLI NAGESWARA RAO VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

An Autonomous, ISO 21001:2018& QS I-Gauge Diamond Rated Institute, Accredited by NAAC with 'A++' Grade
NBA Accreditation for B.Tech. CE,EEE,ME,ECE,CSE,EIE,IT,AME, M.Tech. STRE, PE, AMS, SWEProgrammes
Approved by AICTE, New Delhi, Affiliated to JNTUH, NIRF (2024) Rank band:151-200in EngineeringCategory
College with Potential for Excellence by UGC,JNTUH-Recognized Research Centres:CE,EEE,ME,ECE,CSEVignana Jyothi Nagar, Pragathi Nagar, Nizampet (S.O.), Hyderabad – 500 090, TS, India.
Telephone No: 040-2304 2758/59/60, Fax: 040-23042761
E-mail: postbox@vnrvjiet.ac.in, Website: www.vnrvjiet.ac.in

Department of CSE-(CyS, DS) and AI&DS

TAMASOMA JYOTIRGAMAYA
Estd.1995

CERTIFICATE

This is to certify that the project report entitled "**Forest Fire Prediction**" is a bonafide work done under our supervision and is being submitted by **Mr. Shashank (23071A6745), Miss. Sreeja Reddy (23071A6746), Mr. Abhishek (23071A6747), Mr. Niranjan (23071A6748)**in partial fulfillment for the award of the degree of **Bachelor of Technology** in **CSE-Data Science**, of the VNRVJIET, Hyderabad during the academic year 2025-2026.

Mrs.N.Madhuri	**Dr. T. Sunil Kumar**
Assistant Professor	Professor& HOD
Dept of CSE-(CyS, DS) and AI&DS	Dept of CSE-(CyS, DS) and AI&DS

Course based Project Reviewer

VALLURUPALLI NAGESWARA RAO VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

An Autonomous Institute, NAAC Accredited with ‘A++’ Grade,

VignanaJyothi Nagar, Pragathi Nagar, Nizampet(SO), Hyderabad-500090, TS, India

Department of CSE-(CyS, DS) and AI&DS

DECLARATION

We declare that the course based project work entitled “**FOREST FIRE PREDICTION**” submitted in the Department of **CSE-(CyS, DS) and AI&DS**, Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology, Hyderabad, in partial fulfillment of the requirement for the award of the degree of **Bachelor of Technology in CSE-Data Science** is a bonafide record of our own work carried out under the supervision of **Mrs.N.Madhuri, Assistant Professor, Department of CSE-(CyS, DS) and AI&DS, VNRVJIET.** Also, we declare that the matter embodied in this thesis has not been submitted by us in full or in any part thereof for the award of any degree/diploma of any other institution or university previously.

Place: Hyderabad.

P . Shashank	**P . Sreeja Reddy**	**P . Abhishek**	**P . Niranjan**
(23071A6745)	(23071A6746)	(23071A6747)	(23071A6748)

ACKNOWLEDGEMENT

We express our deep sense of gratitude to our beloved President, **Sri. D. Suresh Babu,** VNR Vignana Jyothi Institute of Engineering & Technology for the valuable guidance and for permitting us to carry out this project.

With immense pleasure, we record our deep sense of gratitude to our beloved Principal, **Dr. C.D Naidu**, for permitting us to carry out this project.

We express our deep sense of gratitude to our beloved Professor **Dr. T. Sunil Kumar**, Professor and Head, Department of CSE-(CyS, DS) and AI&DS, VNR Vignana Jyothi Institute of Engineering & Technology, Hyderabad-500090 for the valuable guidance and suggestions, keen interest and through encouragement extended throughout the period of project work.

We take immense pleasure to express our deep sense of gratitude to our beloved Guide, **Mrs.N.Madhuri**, Assistant Professor in CSE-(CyS, DS) and AI&DS, VNR Vignana Jyothi Institute of Engineering & Technology, Hyderabad, for her valuable suggestions and rare insights, for constant source of encouragement and inspiration throughout my project work.

We express our thanks to all those who contributed for the successful completion of our project work.

Mr. P.Shashank	(23071A6745)
Miss. P.Sreeja Reddy	(23071A6746)
Mr. P.Abhishek	(23071A6747)
Mr. P.Niranjan	(23071A6748)

TABLE OF CONTENTS

ABSTRACT

Wildfires have caused devastating consequences to property and human and animal lives,which has become a global problem. Consequently, advanced wildfire prediction models are required to treat complex features and climate conditions. As a result, Machine Learning and Deep Learning models are becoming popular.

However, creating a balanced true and false labelled dataset in the wild-fire domain is often challenging. Hence, One-class classification models are a promising approach to overcome this concern. In this paper, several One-class classification models are investigated; linear models (Principal Component Analysis and One-Class Support Vector Machines), outlier ensemble models (Lightweight On-line Detector of Anomalies and Locally Selective Combination of Parallel Outlier Ensembles), proximity-based models (Histogram-based Outlier Score and Rotation-based Outlier Detection), probabilistic models (Unsupervised Outlier Detection Using Empirical Cumulative Distribution Functions and Copula-Based Outlier Detection),

Neural network-based models (Deep One-class Classification and Adversarially Learned Anomaly Detection) is used in two case studies for California and Western Australian states. In conclusion, it was found that Deep learning-based One-class classification models outperform other models in terms of performance and feature importance of showcasing the effectiveness of deep neural network models in the wildfire prediction domain.

INTRODUCTION

Forest fires, also known as bush or vegetation fires, are uncontrolled and non-prescribed combustion of plants in natural settings such as forests and grasslands. This article delves into the importance of predicting forest fires using machine learning models. Traditional methods rely on the expertise of forest departments, which can only consider a limited number of parameters. In contrast, machine learning can handle numerous parameters, such as latitude, longitude, satellite data, and more, making it a powerful tool for predicting the confidence of forest fires.

The article provides a step-by-step guide on building a forest fire prediction model using Python. It covers data exploration, cleaning, and the use of a Random Forest Regressor for model building. The guide also includes techniques for model tuning and compression using the bz2file module to handle large datasets efficiently.

By leveraging machine learning, we can enhance our ability to predict and manage forest fires, ultimately helping to protect natural resources and communities.

Technologies Involved:

1. Data Collection and Preprocessing

• **Satellite Imagery:** Collecting data from satellites to monitor vegetation, temperature, and other relevant parameters.

• **Meteorological Data:** Using weather data such as temperature, humidity, wind speed, and precipitation.

2. Machine Learning Algorithms

• **Random Forest Regressor:** Used for predicting the confidence of forest fires based on various parameters.

• **Support Vector Machines (SVM):** Utilized for classification tasks in predicting fire

occurrences.

• **K-Nearest Neighbors (KNN):** Applied for predicting fire risk based on historical

data.

3. Model Tuning and Optimization

• **RandomizedSearchCV:** Used for hyperparameter tuning to improve model accuracy.

• **Feature Importance Analysis:** Identifying the most significant factors contributing

to forest fires.

4. Data Compression and Storage

• **bz2file Module:** Used for compressing large datasets and models to save storage

space.

5. Visualization and Analysis

• **Seaborn and Matplotlib:** Libraries for visualizing data correlations and model outputs.

METHOD

Why do we need a forest fire prediction model?

Well, the first question arises as that why we even need Machine learning to predict forest fire in that particular area? So, yes the question is valid that despite having the experienced forest department who have been dealing with these issues for a long time why is there a need for ML, having said that answer is quite simple that the experienced forest department can check on 3-4 parameters from their human mind but ML on other hand can handle the numerous parameters whether it can be latitude, longitude, satellite, version, and whatnot, so dealing with this multi-relationship of a parameter that is responsible for the fire in the forest we do need ML for sure!

Methodology for Predicting Forest Fires Using Machine Learning

1. Data Collection

• **Satellite Imagery:** Utilizing data from satellites like MODIS on NASA's Terra and Aqua satellites.

• **Meteorological Data:** Gathering weather data including temperature, humidity, wind speed, and precipitation.

• **Geospatial Data:** Incorporating information such as latitude, longitude, and elevation.

2. Data Preprocessing

• **Data Cleaning:** Removing irrelevant columns and handling missing values.

• **Feature Engineering:** Creating new features from existing data, such as extracting year, month, and day from date columns.

• **Categorical Data Handling:** Converting categorical data into numerical format using techniques like one-hot encoding and binning.

3. Exploratory Data Analysis (EDA)

• **Correlation Analysis:** Employing heatmaps to understand relationships between variables.

• **Visualization:** Plotting data to identify patterns and trends.

4. Model Building

• **Random Forest Regressor:** Utilizing this algorithm to predict the confidence of forest fires based on various parameters.

• **Train-Test Split:** Dividing the data into training and testing sets to evaluate model performance.

5. Model Evaluation

• **Accuracy Measurement:** Calculating the accuracy of the model on both training and testing data.

• **Model Tuning:** Using techniques like RandomizedSearchCV to optimize hyperparameters and improve model performance.

6. Model Saving and Compression

• **Pickle Module:** Saving the trained model using the pickle module.

• **bz2file Module:** Compressing the model to reduce storage space.

7. Deployment

• **Loading the Model:** Utilizing the saved and compressed model for predictions.

CODE

Importing libraries

import datetime as dt

import pandas as pd

import numpy as np

import seaborn as sns

import matplotlib.pyplot as plt %matplotlib inline

from sklearn.model_selection import train_test_split

from sklearn.metrics import accuracy_score, classification_report

from sklearn.ensemble import RandomForestRegressor

Reading forest fire exploration dataset (.csv)

forest = pd.read_csv('fire_archive.csv')

forest.head()

	latitude	longitude	brightness	scan	track	acq_date	acq_time	satellite	instrument	confidence	version	bright_t31	frp	daynight	type
0	-11.8070	142.0583	313.0	1.0	1.0	2019-08-01	56	Terra	MODIS	48	6.3	297.3	6.6	D	0
1	-11.7924	142.0850	319.3	1.0	1.0	2019-08-01	56	Terra	MODIS	71	6.3	297.3	11.3	D	0
2	-12.8398	132.8744	311.6	3.1	1.7	2019-08-01	57	Terra	MODIS	42	6.3	298.7	23.1	D	0
3	-14.4306	143.3035	310.1	1.1	1.1	2019-08-01	57	Terra	MODIS	33	6.3	296.1	6.5	D	0
4	-12.4953	131.4897	310.3	4.0	1.9	2019-08-01	57	Terra	MODIS	36	6.3	298.8	27.6	D	0

forest.shape

forest.columns

forest.isnull().sum()

forest.describe()

plt.figure(figsize=(10, 10))

sns.heatmap(forest.corr(),annot=True,cmap='viridis',linewidths=.5)

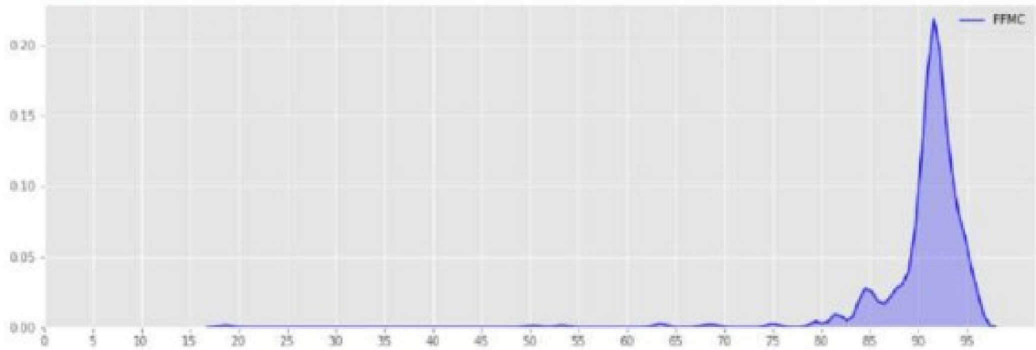
```
forest = forest.drop(['track'], axis = 1)


print("The scan column")
print(forest['scan'].value_counts())
print()
print("The aqc_time column")
print(forest['acq_time'].value_counts())
print()
print("The satellite column")
print(forest['satellite'].value_counts())
print()
print("The instrument column")
print(forest['instrument'].value_counts())
print()
print("The version column")
print(forest['version'].value_counts())
print()
print("The daynight column")
print(forest['daynight'].value_counts())
print()
forest = forest.drop(['instrument', 'version'], axis = 1)
forest.head()
daynight_map = {"D": 1, "N": 0}
satellite_map = {"Terra": 1, "Aqua": 0}
forest['daynight'] = forest['daynight'].map(daynight_map)
```

```
forest['satellite'] = forest['satellite'].map(satellite_map)
forest.head()
forest['type'].value_counts()
types = pd.get_dummies(forest['type'])
forest = pd.concat([forest, types], axis=1)
forest = forest.drop(['type'], axis = 1)
forest.head()
forest = forest.rename(columns={0: 'type_0', 2: 'type_2', 3: 'type_3'})
bins = [0, 1, 2, 3, 4, 5]
labels = [1,2,3,4,5]
forest['scan_binned'] = pd.cut(forest['scan'], bins=bins, labels=labels)
forest.head()
forest['acq_date'] = pd.to_datetime(forest['acq_date'])
forest = forest.drop(['scan'], axis = 1)
forest['year'] = forest['acq_date'].dt.year
forest.head()
forest['month'] = forest['acq_date'].dt.month
forest['day'] = forest['acq_date'].dt.day
forest.shape
y = forest['confidence']
fin = forest.drop(['confidence', 'acq_date', 'acq_time', 'bright_t31', 'type_0'], axis = 1)
plt.figure(figsize=(10, 10))
sns.heatmap(fin.corr(),annot=True,cmap='viridis',linewidths=.5)
fin.head()
Xtrain, Xtest, ytrain, ytest = train_test_split(fin.iloc[:, :500], y, test_size=0.2)
```

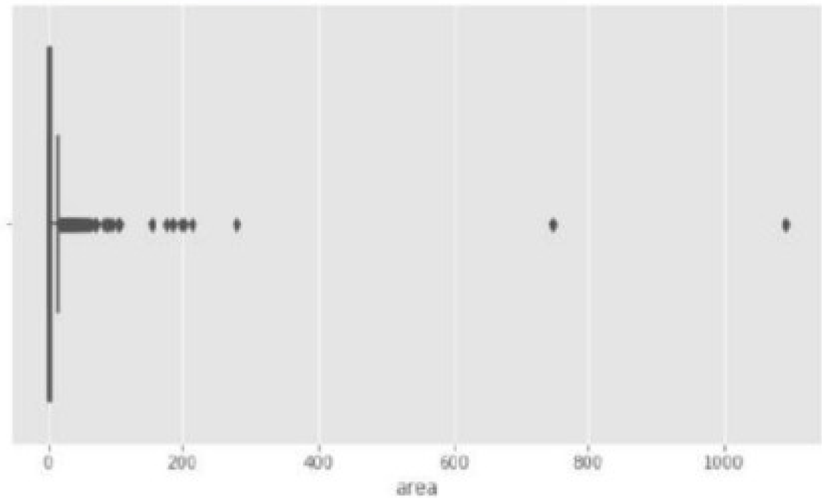
```
random_model = RandomForestRegressor(n_estimators=300, random_state = 42,
n_jobs = -

random_model.fit(Xtrain, ytrain)
y_pred = random_model.predict(Xtest)
random_model_accuracy = round(random_model.score(Xtrain, ytrain)*100,2)
print(round(random_model_accuracy, 2), '%')
random_new.fit(Xtrain, ytrain)
y_pred1 = random_new.predict(Xtest)
random_model_accuracy1 = round(random_new.score(Xtrain, ytrain)*100,2)
print(round(random_model_accuracy1, 2), '%')
random_model_accuracy2 = round(random_new.score(Xtest, ytest)*100,2)
print(round(random_model_accuracy2, 2), '%')
saved_model = pickle.dump(random_new, open('ForestModel.pickle','wb'))
reg_from_pickle = pickle.load(saved_model)
```

TEST CASES/ OUTPUT

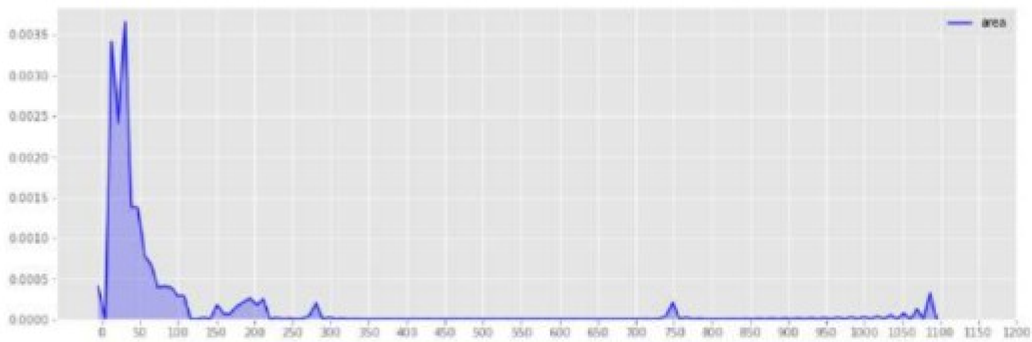
```
plt.figure(figsize=(15,5))
ax = sns.kdeplot(data_df['FFMC'],shade=True,color='b')
plt.xticks([i for i in range(0,100,5)])
plt.show()
```



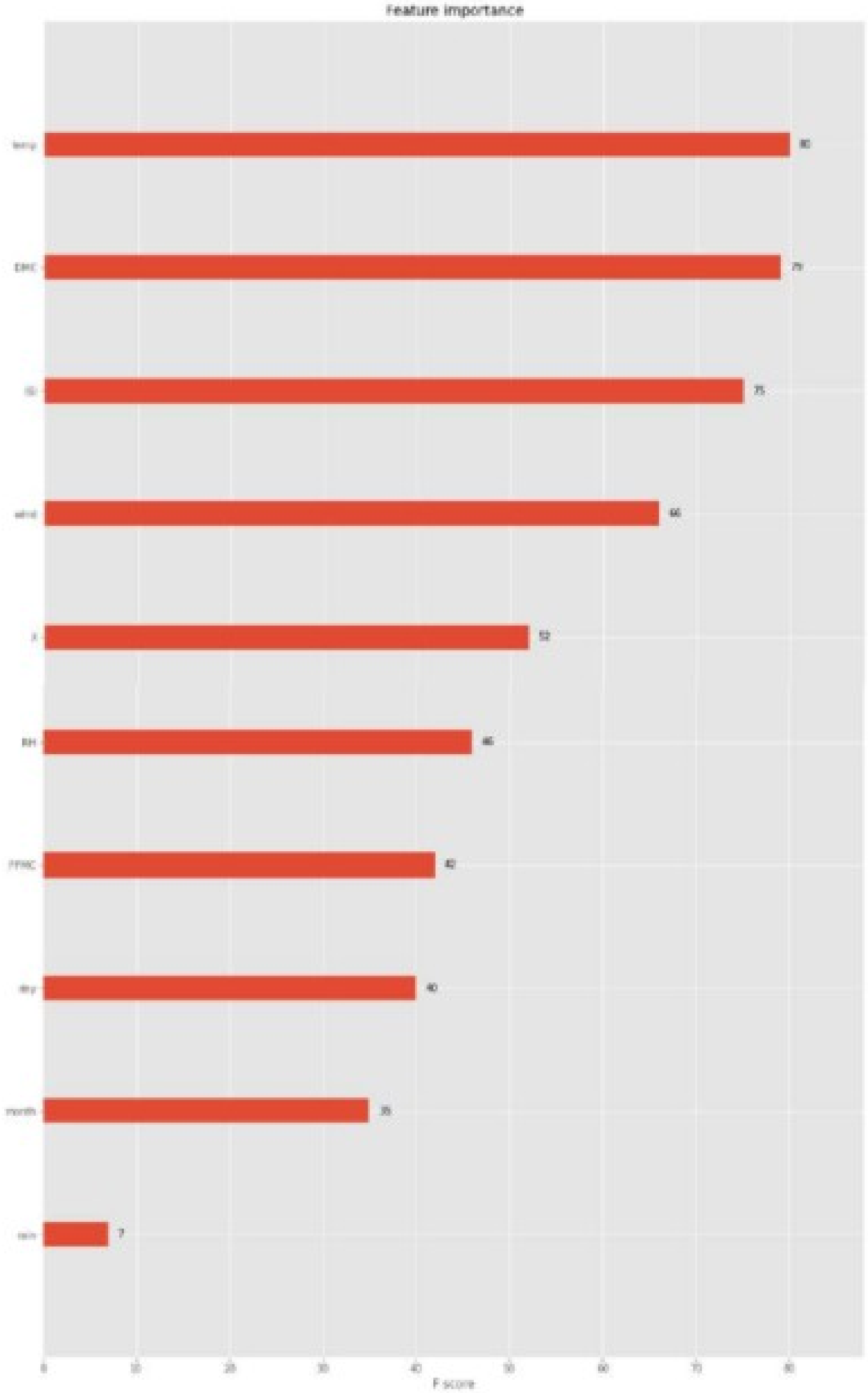
```
outl_dect = sns.boxplot(data_df[target])
```



```
plt.figure(figsize=(15,5))
ax = sns.kdeplot(data_df[target],shade=True,color='b')
plt.xticks([i for i in range(0,1250,50)])
plt.show()
```



Factors effecting the forest fire:



RESULTS

Key Results from the Forest Fire Prediction Model

1. Data Handling

- The dataset used consisted of 36,011 rows and 15 columns.
- No null values were detected in the dataset, ensuring data integrity.

2. Model Accuracy

- The initial Random Forest model achieved an accuracy of 95.32% on the training set but only 65.32% on the test set, indicating overfitting.
- After tuning the model with RandomizedSearchCV, the accuracy improved to 67.39% on the test set.

3. Feature Engineering

- Features such as latitude, longitude, brightness, and satellite data were utilized.
- Categorical data like satellite type and day/night were converted to numerical values for better model performance.

4. Model Compression

- The trained model, initially exceeding 700 MB, was compressed to 93 MB using the bz2file module, making it more storage-efficient.

5. Visualization

- Heatmaps and other visual tools were employed to understand data correlations and feature importance.

CONCLUSION

The application of machine learning in predicting forest fires represents a significant advancement in environmental management and disaster mitigation. By leveraging vast datasets from satellite imagery, meteorological data, and geospatial information, machine learning models can analyze numerous parameters simultaneously, far surpassing the capabilities of traditional methods. The methodology outlined in the Analytics Vidhya article demonstrates a comprehensive approach, from data collection and preprocessing to model building and evaluation. The use of algorithms like Random Forest Regressor, combined with techniques for model tuning and optimization, ensures robust and accurate predictions. Furthermore, the integration of data compression tools like the bz2file module highlights the practical considerations of handling large datasets efficiently. This holistic approach not only enhances our ability to predict and manage forest fires but also underscores the potential of machine learning to address complex environmental challenges. By improving prediction accuracy and enabling proactive measures, these technologies contribute to safeguarding natural resources and protecting communities from the devastating impacts of forest fires.

The use of algorithms like Random Forest Regressor, combined with techniques for model tuning and optimization, ensures robust and accurate predictions. Furthermore, the integration of data compression tools like the bz2file module highlights the practical considerations of handling large datasets efficiently. This holistic approach not only enhances our ability to predict and manage forest fires but also underscores the potential of machine learning to address complex environmental challenges.

REFERENCES

1. *https://www.analyticsvidhya.com/blog/2021/10/forest-fire-prediction-using-machine-learning/*
2. *https://ieeexplore.ieee.org/abstract/document/9498448*
3. *https://www.kaggle.com/datasets/elikplim/forest-fires-data-set*
4. *https://www.kaggle.com/datasets/alik05/forest-fire-dataset*
5. *https://www.researchgate.net/profile/KamalUpreti/publication/359118440_Forest_fire_prediction_using_IoT_and_deep_learning/links/6228eb203c53d31ba4b4926f/Forest-fire-prediction-using-IoT-and-deep-learning.pdf*