Forest fire prediction using K - Nearest Neighbour Model

Dr. Nagagopiraju Vullam[1]. Dr. Subhani Shaik[2], Dr. Gondi Konda Reddy[3], Korivi Vamshee Krishna[4], and Aseena Babu Shaik[5]*

[1]Dept. of AIML, Chalapathi College of Engineering & Technology (A), Guntur, A.P., India.
[2]Dept. of IT, Sreenidhi Institute of Science & Technology (A), Hyderabad, India.
[3]Dept. of ME, Sreenidhi Institute of Science & Technology (A), Hyderabad, India.
[4]Dept. of CSE, Samskruthi College of Engineering and Technology (A), Hyderabad, India.
[5]Dept. of AIML, Samskruthi College of Engineering and Technology (A), Hyderabad, India.

Abstract. Forest fire most dangerous threat of wildfires, this learning, explores the efficiency of the K NN technique for forest fire prediction. With different types of datasets encircling meteorological data, our ML model includes crucial environmental variables to different patterns leading to fire accidents. The KNN model's nonparametric and spatially aware characteristics make it a compelling choice for capturing local dependencies within the dataset. The KNN algorithm's simplicity and flexibility make it convenient for handling diverse datasets and redesigning to changing environmental conditions. However, it's pivotal to acknowledge the importance of feature selection and data preprocessing in enriching the model's performance. Additionally, continuous monitoring and updating of the model with real-time data are essential for ensuring its reliability in predicting forest fire occurrences.

1 Introduction

Forest Fire Prediction is of supreme importance due to its wide implications on the environment, human safety, and economic stability. By using KNN, we can manage and anticipate the occurrence of forest fires, thereby safeguarding wildlife habitats and preserving ecosystems. The significance extends to the safety of mankind, allowing for timely evacuations and strategic deployment of firefighting resources

K-Nearest Neighbor is applied in forest fire prediction by utilizing historical data encompassing various environmental features. The parameters considered are temperature, Relative Humidity, wind, and rain. The Proposed System can be used for the occurrence of a fire given its parametersThis emphasis on interpretability is pivotal

in empowering stakeholders, including firefighting agencies and decision-makers, with actionable insights into the influential features steering our predictions.

ML algorithms may acquire data and utilize it to learn on their own. So how exactly does the ml method operate? Just by looking at the numbers. In supervised learning, models are prepared on labeled data sets, where each input category's characteristics are taught to the algorithm.

In the following sections, we presented the methodology, results and analysis, and implications of our research.By scrutinizing the KNN model's efficacy in predicting forest fires and elucidating the practical applications of machine learning in wildfire management, this research strives to contribute meaningfully to the evolving landscape of proactive wildfire mitigation strategies.

2 Literature Survey

They surveyed five techniques for gentle estimation under Artificial Neural Networks andproposed a model that would be the most effective way to predict forest fires. The outcomes were compiled from the UCI knowledge system table, which was gathered over time. There are 517ways to enter the MNP [1]. Preprocessing the records collection was the first step. Using the Principal Component Analysis approach, the fire domains (clusters) were selected and significant patterns were labeled. Network spreadsheet electronics that determined the best choice techniques for imagining thicket fires [2].

Five consistency verifications—the RMSE, the MSE, the RAE, the MAE, and the Information Gain- were used to determine the final step. The final step involved judging the predictors. All verification predictors should be more productive and effective, according to the data [3]. The outcomes once more demonstrate that, in comparison to other predictors, the SVM technique delivers prediction awareness along with an additional dependable calculation error. According to the data, the SVM forecasts more accurately than other methodsand is the best option for predicting forest fires [4]. [5] Demonstrate that the SVM can be executed in parallel to condense SVM faster in terms ofmachine rapidity when dealing with large data sets that increase the number of training vectors.

First, MapReduce is used to manage the massive data collection process, together with integratedsoftware tools like Hadoop and Twisters. Job reduction in the reiterative map is not supported by Hadoop's MapReduce framework. In both graphs, twisters assist in reducing and combining the jobs. Parallelization is used to divide training samples into smaller groups called subparagraphs. Every subchapter makes use of a libSVM model [6].

3 Proposed Work

Forest Fire Prediction is of supreme importance due to its wide implications on the environment, human safety, and economic stability. By using KNN, we can manage and anticipate the occurrence of forest fires, thereby safeguarding wildlife habitats and preserving ecosystems. The significance extends to the safety of mankind, allowing for timely evacuations and strategic deployment of firefighting resources [7].

K-Nearest Neighbor is applied in forest fire prediction by utilizing historical data encompassing various environmental features. The parameters considered are temperature, Relative Humidity, wind, and rain. The Proposed System can be used for the occurrence of a fire given its parameters[8].

1. Dataset Preparation
Prepare the dataset-linked features and the target variable.

2. Data Normalization
Preprocess the dataset by managing the missing principles, encrypting categorical variables, and measuring numerical countenance if needed.

3. KNN Model Training
Train the KNN model and use your entire dataset. The algorithm will store the entire dataset as its knowledge base.

4. Make Predictions
Use the prepared model to form indicators on new knowledge points or the same dataset [9]. In this scenario, when making predictions for new data facts, the algorithm will discover the k-nearest neighbors within the dataset and classify the novel data point based on the mainstream class amongst these neighbors. The effectiveness of the current model is contingent on the characteristics of the dataset, the choice of features, and the various parameters of the KNN algorithm (such as the number of neighbors, k).

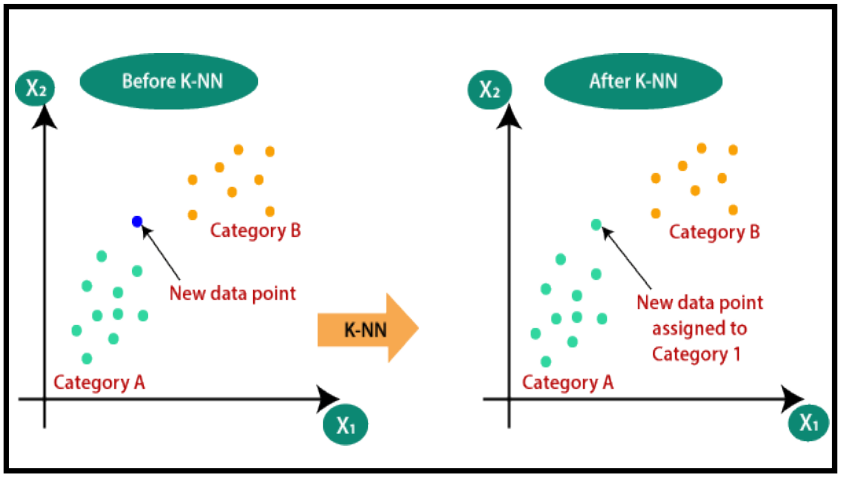

Fig. 1. sample KNN algorithm [2]

4 Methodology

Without being expressly designed, ML algorithms may acquire data and utilize it to learn on their own. So how exactly does the ml method operate? Just by looking at the numbers. Thethree main groups of machine learning algorithms are: - Supervised machine learning -Task-oriented (classification and regression):

ML without supervision - Driven by data (clustering) Strengthening computer learning - taking lessons from errors (reward or punishment) monitoring ml.

In supervised learning, models are prepared on labeled data sets, where each input category's characteristics are taught to the algorithm. Two categories of supervised machine learning are recognized.

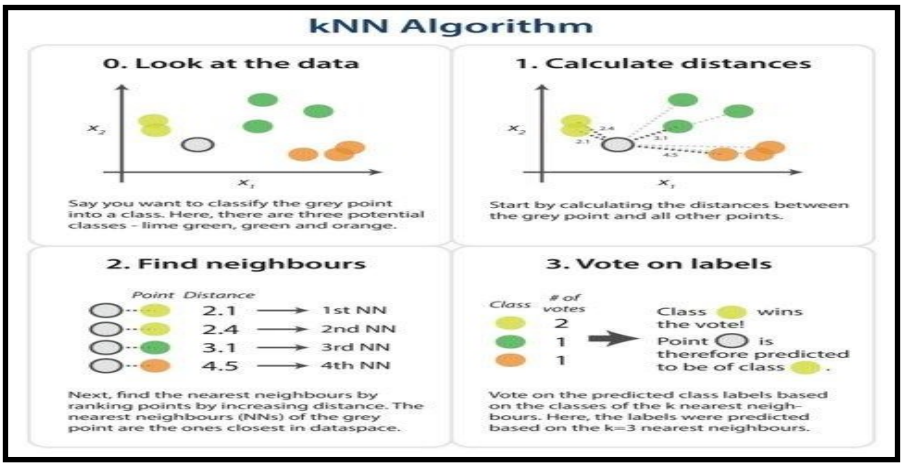

Fig. 2. Sample KNN algorithm

KNN: K-Nearest Neighbors is an individual of the most natural machine learning supervised methods.

The K-NN algorithm is mostly used for the classification of the data, and also it can beused for Regression (secondhand).

The K-NN algorithm does not make any assumptions on basic data, on account of this, itis a non-parametric algorithm.

The following steps will help you understand how the KNN algorithm operates:

Step 1: Decide the center with K.
Step 2: Calculate neighbors using Euclidean distance from the chosen K.
Step 3: Evaluate the closest neighbors from K.
Step 4: Sum the data points from every class of the K neighbors.
Step 5: Determine which group has the greatest number of neighbors. Add more data points to that category now.

Step 6: The completed model has all the necessary features.

Euclidean Distance [10]: This formula is generally used for distance calculation. The length of the line segment that separates two locations in Euclidean space. Therefore, the formula of Euclidean distance is as such:

$$d = \sqrt{(a2 - a1)^2 + (b2 - b1)^2} \quad (1)$$

Each of the variables represented is given below. D-Euclidean distance and (a1, b1) are coordinates of the first point. (b2, b2) coordinates of the second point.

Prerequisites for a functioning KNN Algorithm:

To effectively implement and use the K-Nearest Neighbors algorithm, there are several prerequisites and considerations:

KNN is a distance-based algorithm, so the data needs to be numeric.
KNN doesn't handle missing values well. Addressable of any missing values in the dataset needs to be done before applying KNN.

The choice of the number of neighbors (k) is crucial. If the cost of K is minor, the model can be more prone to noise. On the other hand, if the cost of K is big, the model is considered too generalized.

Utilize the complete set of training and testing data. Opt for an appropriate worth for K, denoting the nearest data points, where K is an integer typically chosen within the range of 3 to 10 for optimal results [11].

For each data point, execute the following steps:

1. Evaluate the distance amid the test data and every row in the training data utilizing similarity metrics such as distance measures (Euclidean, Manhattan, Murkowski) or specific metrics like overlap/Hamming Distance for discrete variables. The commonly employed distancemetric is Euclidean distance.

2. Arrange the data points in ascending order based on their computed distances.

3. Choice the topmost K rows from the arranged array.

4. Attribute to the test data by determining the most frequently occurring class among these selected rows.

Selection of the K value

The value of k **= sqrt(N)**
Here total number of samples (N)=518

k=sqrt (518) ~=22

5 Development of Research Work

The pre-processed data is separated into training and testing data. Generally, most of the research people considered data 80:20 [12,13] for training and testing. In this data 13 types of properties are considered. The following Table 1 shows.

Table 1. Dataset

	X	Y	month	day	FFMC	DMC	DC	ISI	temp	RH	wind	rain	area
2	7	5	mar	fri	86.2	26.2	94.3	5.1	8.2	51	6.7	0	0
3	7	4	oct	tue	90.6	35.4	669.1	6.7	18	33	0.9	0	0
4	7	4	oct	sat	90.6	43.7	686.9	6.7	14.6	33	1.3	0	0
5	8	6	mar	fri	91.7	33.3	77.5	9	8.3	97	4	0.2	0
6	8	6	mar	sun	89.3	51.3	102.2	9.6	11.4	99	1.8	0	0
7	8	6	aug	sun	92.3	85.3	488	14.7	22.2	29	5.4	0	0
8	8	6	aug	mon	92.3	88.9	495.6	8.5	24.1	27	3.1	0	0
9	8	6	aug	mon	91.5	145.4	608.2	10.7	8	86	2.2	0	0
10	8	6	sep	tue	91	129.5	692.6	7	13.1	63	5.4	0	0
11	7	5	sep	sat	92.5	88	698.6	7.1	22.8	40	4	0	0
12	7	5	sep	sat	92.5	88	698.6	7.1	17.8	51	7.2	0	0
13	7	5	sep	sat	92.8	73.2	713	22.6	19.3	38	4	0	0
14	6	5	aug	fri	63.5	70.8	665.3	0.8	17	72	6.7	0	0
15	6	5	sep	mon	90.9	126.5	686.5	7	21.3	42	2.2	0	0
16	6	5	sep	wed	92.9	133.3	699.6	9.2	26.4	21	4.5	0	0
17	6	5	sep	fri	93.3	141.2	713.9	13.9	22.9	44	5.4	0	0
18	5	5	mar	sat	91.7	35.8	80.8	7.8	15.1	27	5.4	0	0
19	8	5	oct	mon	84.9	32.8	664.2	3	16.7	47	4.9	0	0
20	6	4	mar	wed	89.2	27.9	70.8	6.3	15.9	35	4	0	0

Implementation steps
Importing essential libraries
Load the dataset

Importation of essential libraries
We need to import numpy, pandas, matplotlib to use them while building the model.

```
# Importing necessary libraries
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score, classification_report
```

5.1 Loading dataset

The forest fire dataset is loaded in the form of data frames.

```
[2]: file_path = r'C:\Users\abhin\Downloads\forestfires.csv'

     # Load the dataset into a Pandas DataFrame
     data = pd.read_csv(file_path)

     # Display the first few rows of the dataset
     data.head()
```

[2]:

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

```
# Create a KNN classifier with Euclidean distance
knn_classifier = KNeighborsClassifier(n_neighbors=5, metric='euclidean')

# Train the classifier
knn_classifier.fit(X_train, y_train)

# Make predictions on the test set
y_pred = knn_classifier.predict(X_test)

# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
report = classification_report(y_test, y_pred)

print(f"Accuracy: {accuracy}")
print("Classification Report:\n", report)
```

```
# Separate features and labels
X = dataset[['FFMC', 'DMC', 'DC', 'ISI', 'temp', 'RH', 'wind', 'rain']].values  # Features
y = (dataset['area'] > 0).astype(int)  # Binary classification: 1 if fire occurred, 0 otherwise

# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
# Standardize the features (optional but recommended for KNN)
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
```

```
# Apply preprocessing to the training set
X_train_preprocessed = preprocessor.fit_transform(X_train)

# Apply preprocessing to the testing set
X_test_preprocessed = preprocessor.transform(X_test)

# Display the shapes of the training and testing sets after preprocessing
print("Training set shape (after preprocessing):", X_train_preprocessed.shape, y_train.shape)
print("Testing set shape (after preprocessing):", X_test_preprocessed.shape, y_test.shape)
```

```
Training set shape (after preprocessing): (413, 29) (413,)
Testing set shape (after preprocessing): (104, 29) (104,)
```

Table 2. Results of KNN

Accuracy: 0.5673076923076923
Classification Report:

	precision	recall	f1-score	support
0	0.57	0.51	0.54	51
1	0.57	0.62	0.59	53
accuracy			0.57	104
macro avg	0.57	0.57	0.57	104
weighted avg	0.57	0.57	0.57	104

Table 2 generates an accuracy rate of 56.7%. It provides information on the classification report, including precision value, recall, f1 score, and support value.

6 Conclusion

By imposing the proximity-based classification mechanism of KNN, the model can effectively analyze historical data, considering factors such as weather conditions, topography, and vegetation types. This ensures accurate identification of patterns associated with past forest fires,rooting for predicting potential fire outbreaks. The KNN algorithm's simplicity and flexibility make it convenient for handling diverse datasets and redesigning to changing environmental conditions. However, it's pivotal to acknowledge the importance of feature selection and data preprocessing in enriching the model's performance. Additionally, continuous monitoring and updating of the model with real-time data are essential for ensuring its reliability in predicting forest fire occurrences.

References

1. Babu, Suresh and Kabdulova, G and Kabzhanova, G. “Developing the Forest Fire Danger Index for Country Kazakhstan by Using Geospatial Techniques”, 2019.

2. T. Preeti, S. Kanakaraddi, A. Beelagi, S. Malagi, and A. Sudi, "Forest Fire Prediction Using Machine Learning Techniques," 2021 International Conference on Intelligent Technologies (CONIT), Hubli, India, 2021.

3. K. Zhu, H. Wang, H. Bai, J. Li, Z. Qiu, H. Cui, E.Y. Chang Parallelizing Support VectorMachines on Distributed computers Adv. Neural Inf. Process. September 2008.

4. Subhani Shaik, “DM Algorithms Based Clustering for Road Accident Data Analysis”, International Journal of Computer Sciences and Engineering, Vol.-6, Issue-9, Sept. 2018.

5. Vijayalakshmi K and Subhani Shaik, “Predicting employee attrition methodologies ofK-fold technique”, I. J. Mathematical Sciences and Computing, March 2023, 1, 23-36.

6. Jagan Chowhaan, D. Nitish, G. Akash, Nelli Sreevidya, Subhani Shaik, “Machine Learning Approach for House Price Prediction”, Asian Journal of Research in Computer

Science,Volume 16, Issue 2, Page 54-61, June- 2023.

7. Mani Chandra, D. Teja, S. Kiran Kumar, N Sreevidya, Dr. Subhani Shaik," WhatsApp Chat analysis and spam message detection using Machine learning algorithms", 2nd International Conference on Data Science and Artificial Intelligence (ICDSAI) 24-25 April 2023, Lendi College of Engineering (A), Vizianagaram, AP.

8. Mamatha, Srinivasa Datta and Subhani Shaik," Fake Profile Identification using Machine Learning Algorithms", International Journal of Engineering Research and Applications, Vol.11, Series-2, July-2021.

9. Dr. Sunil Bhutada and Subhani Shaik, "IPL Match Prediction using Machine Learning", IJAST, Vol.29, Issue 5, April-2020.

10. R. Vijaya Kumar Reddy, Shaik Subhani, G. Rajesh Chandra, B. Srinivasa Rao," Breast Cancer Prediction using Classification Techniques", International Journal of Emerging Trends in Engineering Research, Vol. 8, No.9,2020.

11. R. Vijaya Kumar Reddy, Subhani Shaik, B. Srinivasa Rao, "Machine learning based outlier detection for medical data", Indonesian Journal of Electrical Engineering and Computer Science, Vol. 24, No. 1, October 2021.

12. Subhani Shaik, P. Santhosh Kumar S. Vikram Reddy K. Sai Srinivas Reddy, and Sunil Bhutada," Machine Learning based Employee Attrition Predicting", Asian Journal of Research in Computer Science, Volume 15, Issue 3, March 2023.

13. Neeraja, Anupam, Sriram, and Subhani Shaik," Fraud Detection of AD Clicks Using Machine Learning Techniques", Journal of Scientific Research and Reports, Volume 29, Issue 7, June-2023.