Predictive Modeling Approach for California's Wildfire Threat

Final Project Report

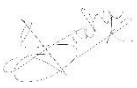
Group 33

Student 1: Kruthika Srinivas Vasisht
Student 2: Sneha Manjunath Chakrabhavi

vasisht.k@northeastern.edu
chakrabhavi.s@northeastern.edu

Percentage of Effort Contributed by Student 1: 50%
Percentage of Effort Contributed by Student 2: 50%

Signature of Student 1: 

Signature of Student 2: 

Submission Date: 04/12/2024

Table of Contents

1. Problem Setting

California has suffered greatly from wildfires in recent years, with serious repercussions for its people, environment, and economy. The diverse terrain of the state, along with climate change and human activities, has left it susceptible to regular and severe wildfires. These fires not only act as direct threats to life and property but also cause enduring damage to the environment, atmospheric conditions, and community health. It's essential to grasp the underlying patterns and reasons behind these wildfires to manage, prevent, and reduce their effects more effectively.

2. Problem Definition

The objective is to create a predictive model that can analyze past wildfire data in California. We aim to track how often and how severely wildfires occur, pinpoint their main causes, understand where they're most likely to happen, and evaluate their effects on people and property. By using data mining techniques, this model is critical given California's ongoing wildfire challenges. Ultimately, the aim is to offer quick and useful information that helps both communities and officials make smarter decisions and reduce risks associated with wildfires.

3. Data Sources

The data source is an extensive compilation of historical records of wildfire incidents in California, available on Kaggle. It provides insightful information about the dynamics of wildfires in the area.
Kaggle/California_Wildframes/2013-2020
Fire.CA.gov/incidents

4. Data Description

The "California Wildfires" dataset is impressively detailed, containing 40 columns and nearly 40,780 entries that document wildfire incidents from 2013 to 2020. This rich dataset captures six distinct wildfire events, providing an in-depth look at multiple facets of each incident. It features data on the size of fires, the administrative regions affected, the area in acres burned, current fire status, involved counties, and fatalities. This information highlights the geographic and meteorological variety found in the dataset, which is essential for pinpointing the exact locations of the fires.

After cleaning and processing the data, we refined our dataset to contain 18 columns and 10,988 entries, each representing a distinct wildfire incident. We kept essential categorical variables like the Date, County, and Names of the fires to provide important context. Among the 15 numerical variables, crucial data points for prediction included Maximum temperature, Minimum temperature, Fire Cause, Latitude, Longitude, and Acres burned. This streamlined dataset allows us to use predictive modeling to estimate the chances or potential impacts of future wildfires, drawing on historical data and other pertinent factors.

4.1 Description of Variables

The table gives a summary of the type and data format for each column, along with a description of what the information in each column represents in the final dataset.

Columns	Type	Meaning
Date	Object	The month and year of when the fire took place.
Count	Object	The county the fire started in.
Maxtemp	Float	The average maximum temperature of that month (°F).
Mintemp	Float	The average minimum temperature of that month (°F).
Avgtemp	Float	The average temperature of that month (°F).
Snow	Float	The total snow for that month.
Humid	Float	The average humidity for that month.
Wind	Float	The average wind for that month.
Precip	Float	The average precipitation for that month.
q_avgtemp	Float	The quarterly average temperature (°F).
q_avghumid	Float	The quarterly average humidity.
q_sumprecip	Float	The quarterly average precipitation.
Sunhour	Float	The average hours of sun for that month.
Name	Object	The name of the fire.
Cause	Float	The cause of the fire.
Latitude	Float	The latitude coordinate of the fire's location.
Longitude	Float	The longitude coordinate of the fire's location.
Acresburned	Float	The total number of acres burned.

5. Data Exploration

In this project, data exploration is required for several reasons. Initially, it gives a foundational understanding of the dataset's structure, quality, and the variables involved, which is crucial for any predictive modeling. We can understand the patterns, developments, and correlations between different variables, like weather conditions, geographic locations, and the occurrence and severity of wildfires. This insight is vital for identifying the most relevant factors that influence wildfire risks and guiding the selection of features for modeling.

This helps in detecting anomalies, outliers, or missing values that could bias the analysis or predictive performance if they are not properly addressed. A thorough exploratory analysis of statistical and visualisation methods ensures that the predictive models are built on a solid, well-understood foundation, ultimately leading to more reliable, accurate insights for wildfire prevention and management strategies.

5.1 Data Cleaning and Preparation

Initial Dataset Loading: The dataset is initially loaded with all its original columns. This ensures a comprehensive understanding of the available data before making any modifications.

Row/Column Removal: Irrelevant columns such as `fire_name`, `lat`, and `long` are removed. The removal of `fire_name` eliminates redundant or non-informative data, while excluding latitude (`lat`) and longitude (`long`) prevents the model from overfitting specific geographic locations, particularly areas with abundant historical data which might not be representative of broader patterns.

Target Variable Creation: A binary target variable, `fire_occurred`, is introduced. This variable is derived from the `acres_burned` field; it is set to 1 if any fire occurred (acres burned > 0) and 0 otherwise. This simplifies the model's output to a yes/no format, which is more straightforward for binary classification tasks.

5.2 Dimension Reduction

Feature Engineering: Instead of applying dimensionality reduction techniques like PCA, which could obscure interpretability, the focus is on feature engineering. New features are constructed from the existing data to enhance the model's ability to make predictions.

Handling Variables: Dummy variables are created for categorical data such as the month of the year, and 'year' is retained as a feature. These modifications allow the model to capture seasonal and annual trends in wildfire occurrences, reflecting how wildfire risks vary with time due to environmental changes and human activities.

5.3 Data Transformation

Temporal Features: The dataset is enriched by transforming the date column into separate 'year' and 'month' columns. This transformation leverages temporal information, enabling the predictive models to account for seasonality and year-over-year trends in wildfire occurrences, crucial for forecasting given the seasonal nature of wildfires.

Spatial Information Handling: Detailed spatial coordinates are omitted to keep the model generalized, the inclusion of the `County` variable retains a level of geographical specificity. This approach uses county-level data as a proxy for regional characteristics (e.g., climate patterns, and historical wildfire data), thereby providing spatial context without the granularity of exact coordinates.

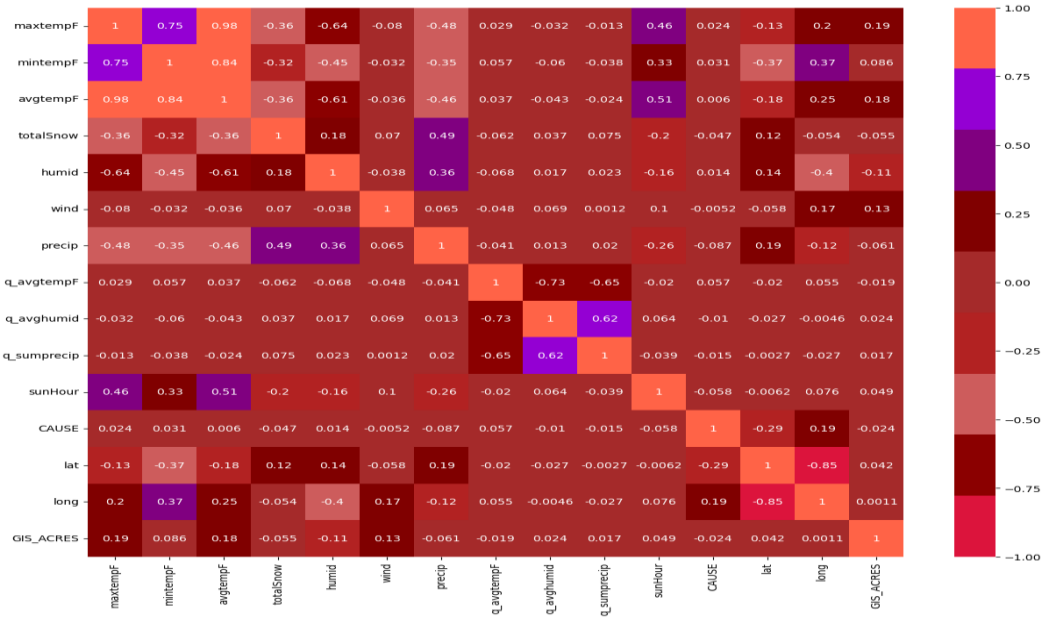
5.4 Exploratory Data Analysis

A variety of visualization techniques are employed to identify and understand patterns and relationships within the data. Heatmaps are used to visualize correlations between variables, bar graphs and pair plots help in assessing the distribution and relationship of categorical data, boxplots are utilized for observing distributions and spotting outliers, and scatter plots are applied to explore relationships between continuous variables. These tools collectively reveal insights into the factors influencing wildfires and help guide further data processing and model refinement.

Heatmap: Visualizing the Correlation Matrix

In this analysis, a correlation matrix was calculated based on the numeric columns of the data frame. The correlation matrix represents the pairwise correlations between all pairs of numeric variables in the dataset. Each cell in the matrix contains the correlation coefficient, which indicates the strength and direction of the linear relationship between two variables.

After calculating the correlation matrix, a heatmap with annotations was created using the Seaborn library. The heatmap provides a visual representation of the correlation matrix, with colors indicating the magnitude and direction of the correlations. Positive and negative correlations are shown in shades of red. The intensity of the color reflects the strength of the correlation.



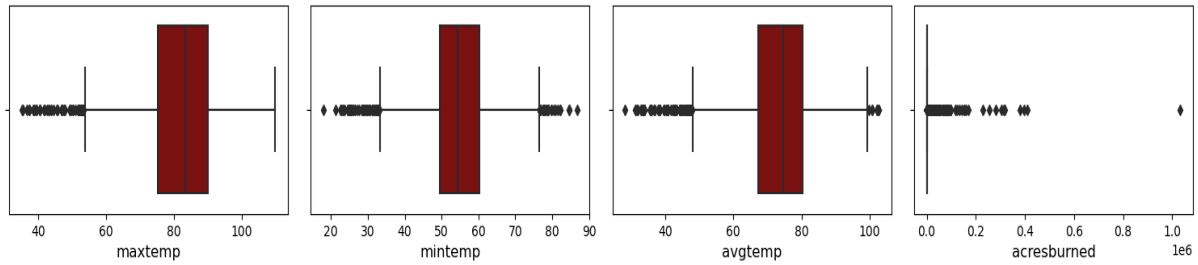
Heatmaps illuminated the correlation between environmental variables (like temperature, humidity, and wind speed) and wildfire characteristics (such as acres burned). High correlations indicated variables that might significantly impact the likelihood and severity of wildfires, guiding feature selection for the predictive model.

By identifying pairs of highly correlated variables, heatmaps helped in mitigating multicollinearity in the dataset. Removing or combining redundant variables ensured that the model's performance wasn't compromised by unnecessary complexity or misleading interpretations of feature importance.

Boxplots: Distribution and outliers in each variable

The boxplots for maxtemp, mintemp, and avgtemp represent the distribution of temperatures during wildfire incidents. Understanding the distribution of temperatures can help determine if higher temperatures correlate with increased wildfire occurrences or severity.

The acresburned boxplot provides a visual summary of the severity of wildfires in terms of the area affected. This can be instrumental in understanding the typical scale of wildfire events and identifying any extreme cases.

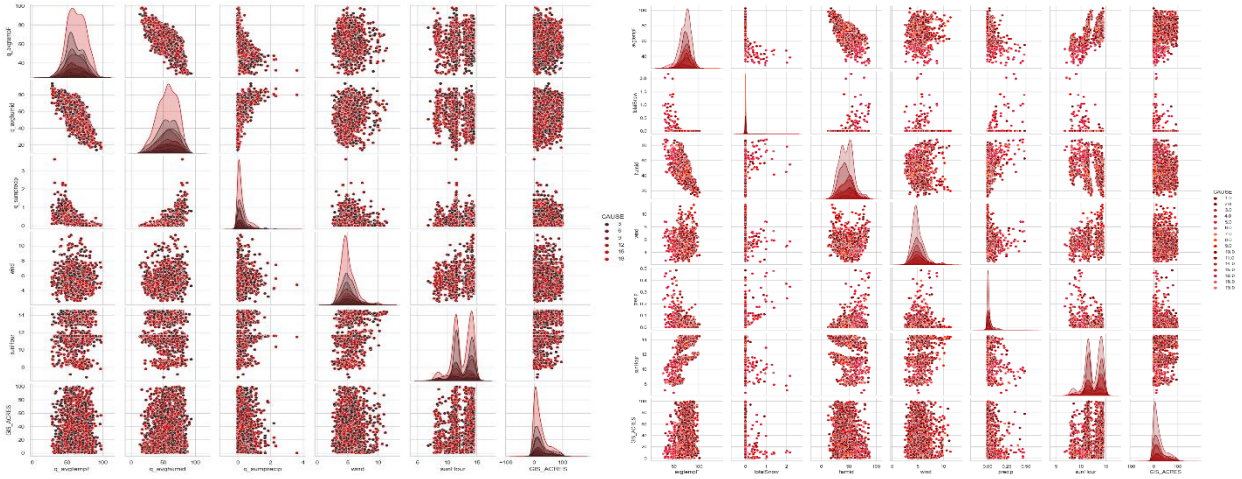


Helping to assess if temperature should be a key variable in the predictive model. Identifying outliers that could either represent data entry errors or actual extreme wildfire events that may need to be considered separately.

Pair plots: Monthly/Quarterly weather and fire data, by the cause of fire

This contains two pair plots, which are grid-based visualizations that enable one to see the distribution of a single variable as well as the relationship between two variables. The diagonals in each plot show the univariate distribution of the variables, typically as histograms or kernel density estimates. The off-diagonals show scatter plots for the pairwise relationship between the variables.

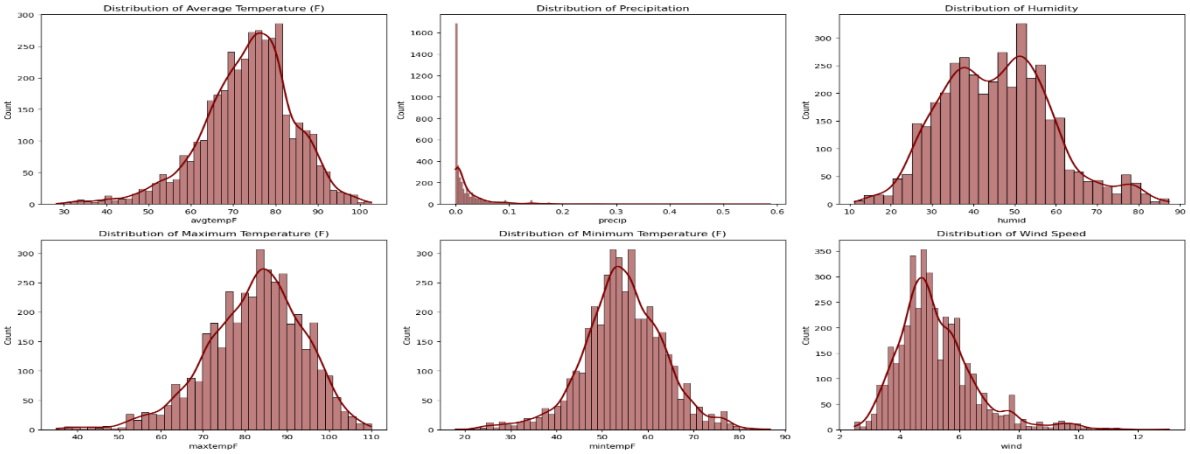
The pair plots represent the relationships and distributions of various variables related to wildfire occurrences. Variables included in this type of analysis typically range from environmental data like temperature, humidity, and wind to geographical data like latitude and longitude. These plots are color-coded to differentiate data points by a specific category, such as the cause of a wildfire, making it easier to identify patterns.



They analyze the relationships between variables like temperature, humidity, wind, and their quarterly averages, and correlation with wildfire occurrences.

Histogram: Distribution of numeric variables

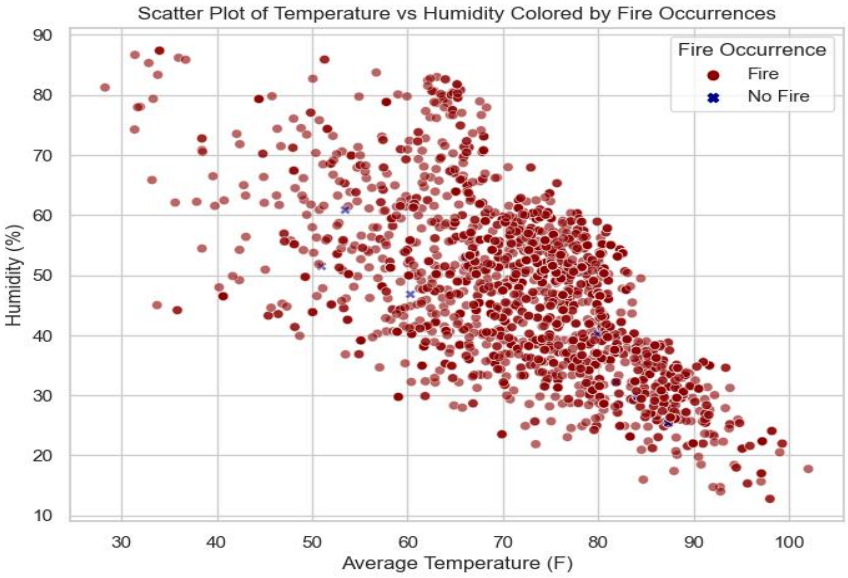
The provided histograms with overlaid kernel density plots represent various environmental factors from the California wildfire dataset. The average temperature shows a normal distribution, centered around the 60-70°F mark, indicating that this is the most common temperature range in the dataset. Precipitation is right-skewed, with the majority of values clustered near zero, suggesting infrequent and low precipitation.



Humidity also displays a roughly normal distribution but with a skew towards lower values, which points to a generally drier climate. Lastly, the wind speed histogram exhibits a right skew, with most of the data concentrated at lower wind speeds, though there is a long tail indicating occasional higher wind speed events.

Scatter Plot: Relationship between average temperature and humidity by fire occurrence

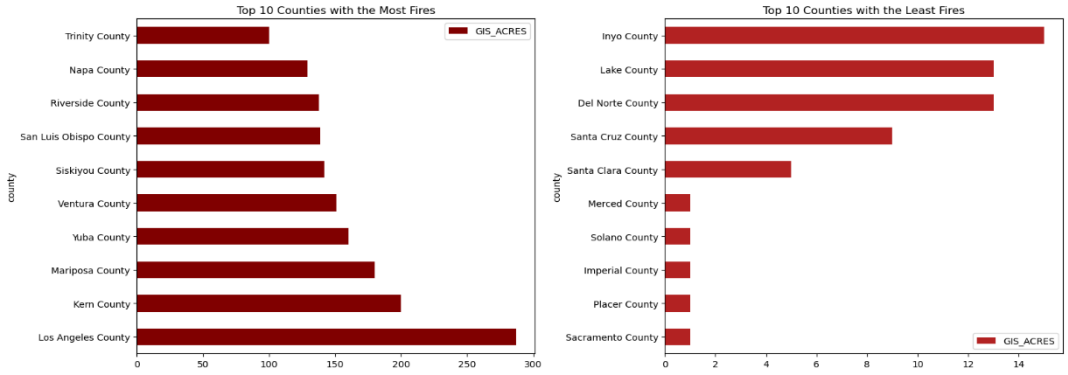
The scatter plot depicts a clear relationship between average temperature, humidity, and wildfire occurrences. Most wildfires (indicated by red dots) have happened at higher temperatures, regardless of humidity levels, suggesting that temperature may be a significant factor in wildfire outbreaks.



While fires are scattered across a range of humidities, there are very few fire occurrences (blue dots) at lower temperatures, highlighting the role temperature plays in the risk of a wildfire. This visual evidence implies that higher temperatures might be more influential than humidity in the likelihood of a fire starting. Additionally, the clustering of red dots at mid-range humidity levels indicates that there is a range of humidity where fires are more likely to occur, even if it is not as strong a predictor as temperature. The absence of fires in the high temperature and high humidity region may suggest that extremely humid conditions could help mitigate the risk of fires, despite high temperatures.

Bar Chart: Top 10 counties with the most and least fires per county

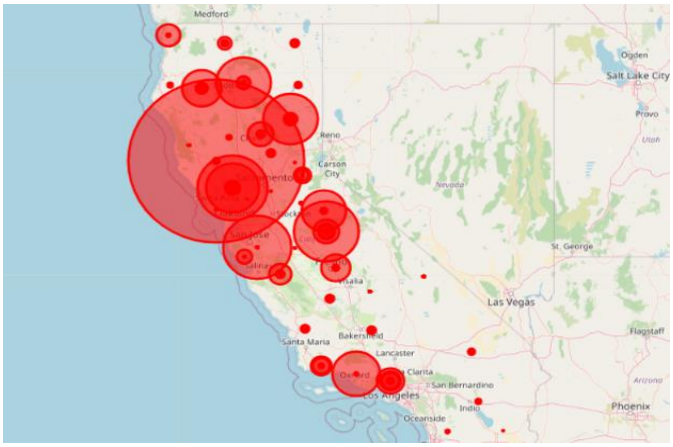
The Bar chart shows the top 10 counties with most and least acres burned in wildfires. By visualizing the number of wildfires attributed to different causes like the highest wildfires in a county in a bar chart, we could identify the most common sources of wildfires.



Bar charts showing the number of wildfires over months or years highlight temporal trends in wildfire occurrence. Identifying periods with higher wildfire frequencies could guide seasonal preparedness and emergency response planning.

Map Chart: Hotspots of Wildfire Incidents in California

Maps are indispensable in projects involving geographical data. They provide spatial context and can highlight patterns that are not apparent in tabular data alone: Mapping the locations of wildfires, with markers sized or colored by severity (e.g., acres burned), offers immediate insight into high-risk areas. This spatial analysis helps in identifying patterns such as proximity to urban areas or natural features and mitigation strategies.



Overlaying maps with environmental data (temperature, humidity, vegetation dryness) can reveal areas at high risk of wildfires. These visualizations support the targeting of preventive measures, such as controlled burns or clearing vegetation in vulnerable areas.

Maps showing both wildfire risk predictions and the location of firefighting resources (fire stations, water sources) facilitate strategic planning for resource allocation. They ensure rapid response capabilities are optimized for areas with the highest predicted risks.

6. Data Mining Tasks

6.1 Data Splitting and Training

When preparing data for modeling wildfire incidents in California after the data preprocessing, the dataset is initially split into a feature matrix, X, and a target vector, y. This separation is crucial for distinguishing between input features, which are predictors, and the outcome variable to be predicted. Here, training and testing split alone is adequate given the constraints of data size. The feature matrix X initially includes various columns that may or may not be relevant to the prediction of wildfires. For this dataset, non-predictive columns such as 'acres_burned', 'date', and 'q_avgtemp' are removed from X. The target vector y is assigned to the column 'fire_occurred', which indicates whether a wildfire occurred.

The data is then divided into training and testing sets using the `train_test_split()` function from the `sklearn.model_selection` module, typically allocating 80% of the data for training and 20% for testing. This split ensures that the models are trained on a substantial portion of the data, allowing them to learn the underlying patterns, while the test set is reserved for evaluating the model's predictive performance on unseen data.

6.2 Model Selection and Preparation

The objective is to perform binary classification, specifically to predict the occurrence of wildfires. Four algorithms are selected based on their suitability for this task.

- **Decision Tree:** Employs a tree-like structure to perform classification and regression by recursively splitting data along attributes that result in the significant differentiation of outcomes.
- **Logistic Regression:** This model is used for binary classification tasks, calculating the probability that an instance belongs to a particular class, with outputs ranging from 0 to 1.
- **Random Forest:** An ensemble method that enhances accuracy and stability by combining the predictions of multiple decision trees, suitable for both classification and regression tasks.
- **K-Nearest Neighbors (KNN):** Algorithm that assigns class labels or predicts values based on the attributes of the K-nearest training examples, applicable to both classification and regression.

The StandardScaler is applied to normalize the features, ensuring that the distance-based calculations reflect equally scaled features, which is vital for the performance of KNN. Once the models are selected, we train and evaluate them on various performance metrices.

6.3 Model Training and Evaluation

Each model is trained using its respective training data. These models are evaluated using various metrics to understand their performance. Model evaluation is performed using the `score()` method, which provides the accuracy of the model predictions. These models were likely selected for their diverse approaches to predictive tasks, allowing for a comprehensive analysis of the dataset's

characteristics. Each model contributes a unique perspective to the problem, and their combined use provides a multi-faceted view of the predictive task at hand. Further metrics include:

Accuracy: The ratio of correctly predicted instances to the total instances.

Precision: The ratio of true positive predictions to the total positive predictions.

Recall: The ratio of true positive predictions to all actual positive instances.

F1 Score: The harmonic mean of precision and recall, providing a balance between the two.

ROC AUC: The area under the Receiver Operating Characteristic curve, a graphical plot illustrating the diagnostic ability of a binary classifier system.

These metrics provide insights into how well each model performs, particularly in terms of correctly classifying whether a wildfire will occur that will help us chose the best performer for deployment.

6.4 Hyperparameter Tuning and Validation

In scenarios where model development involves relatively straightforward objectives, the primary focus is typically on assessing basic performance rather than on intensive hyperparameter tuning. For such applications, a simple train-test split is often sufficient to evaluate whether the model can generalize beyond its training data. Consequently, we have chosen to utilize only training and testing datasets, without incorporating a separate validation set.

Typically, a validation dataset is used to provide an unbiased evaluation of a model trained on the training dataset. It plays a crucial role in fine-tuning hyperparameters and minimizing overfitting, which occurs when a model is overly fitted to the training data and performs poorly on new, unseen data. However, in straightforward model applications focused on initial performance evaluation, the use of a validation dataset can be omitted. This omission does not significantly compromise the assessment of the model’s ability to generalize, making it a viable approach for simpler tasks.

The models' varied capabilities in handling different types of data and learning tasks make them collectively advantageous for creating a robust and reliable wildfire prediction system. Although we discuss the use of tuning techniques like `GridSearchCV` or `RandomizedSearchCV` for systematic hyperparameter tuning and validation, these tools are particularly useful when the development process requires precise model optimization. In cases where models are simple and the primary goal is to check performance, extensive hyperparameter tuning need not be conducted initially.

7. Data Mining Models

The models chosen for data mining tasks for the prediction of wildfires are given below:

7.1 Decision Tree

7.2 K Nearest Neighbours (KNN)

7.3 Random Forest

7.4 Logistic Regression

8. Performance Evaluation

8.1 Decision tree

A decision tree model is a type of machine learning model used for both classification and regression tasks. It works by recursively splitting the dataset into subsets based on the most significant attribute at each step, resulting in a tree-like structure where each internal node represents a "decision" based on an attribute, each branch represents the outcome of the decision, and each leaf node represents a class label or a numerical value.

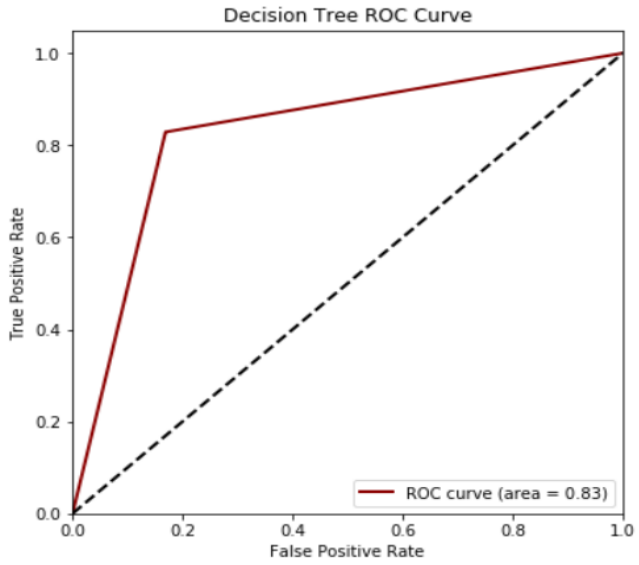
Advantages:

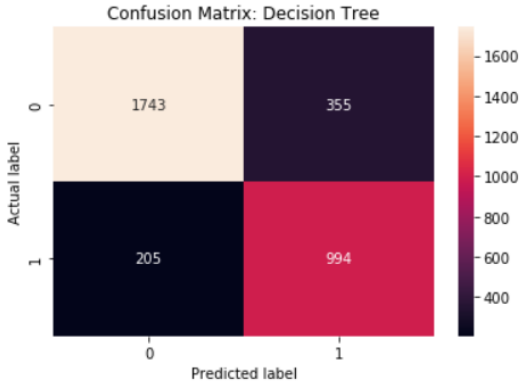
- Decision trees do not require extensive data preprocessing such as normalization, making them suitable for datasets with mixed data types.
- They can handle missing values in the dataset.
- They are easy to interpret and visualize, making them useful for understanding the underlying logic of the model.

Disadvantages:

- Decision trees are prone to overfitting, especially when the tree is deep and complex.
- They are sensitive to minor changes in the data, which can lead to different tree structures.
- They can become overly complex, especially with large datasets and many features, leading to longer training times and less interpretable models.

Decision Tree Performance Metrics:





Accuracy:	0.8301
Precision:	0.7368
Recall:	0.8290
F1 Score:	0.7802
ROC AUC:	0.8298

8.2 K-Nearest Neighbors (KNN)

The K-Nearest Neighbors (KNN) model is a simple, instance-based learning algorithm used for classification and regression tasks. In classification, it assigns a class label to a new data point based on the majority class of its K nearest neighbors in the training dataset. In regression, it predicts the value of a new data point by averaging the values of its K nearest neighbors.

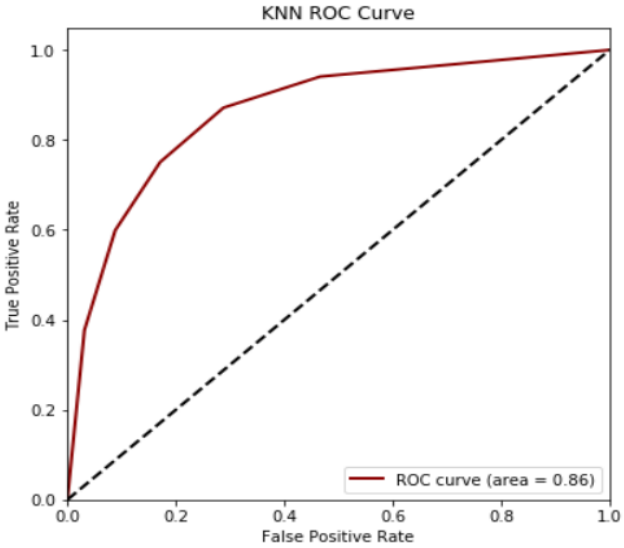
Advantages:

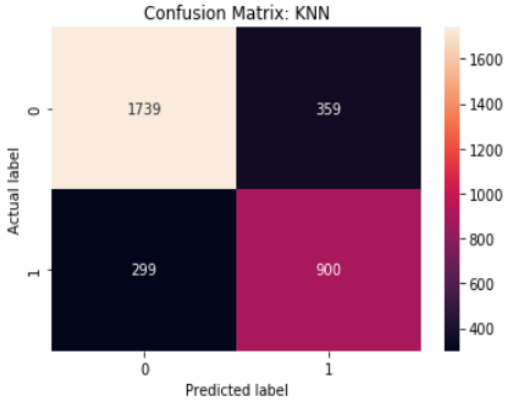
- KNN is easy to understand and implement
- It does not make any assumptions about the underlying data distribution.
- It does not require a training phase. It simply stores the training instances and uses them for predictions.
- It can be used for both classification and regression tasks, making it a versatile algorithm.

Disadvantages:

- As the size of the training dataset grows, the time taken to find the nearest neighbors increases, making KNN computationally expensive.
- KNN can perform poorly if the dataset contains noisy data, outliers, or irrelevant features, as these can affect the distance calculations.
- Selecting the K value can be challenging and may require experimentation and validation.
- KNN may not perform well on imbalanced datasets, where one class is significantly more frequent than the others, as it tends to favor the majority class.

KNN Performance Metrics:





Accuracy:	0.8004
Precision:	0.7148
Recall:	0.7506
F1 Score:	0.7323
ROC AUC:	0.8638

8.3 Random Forest

Random Forest is an ensemble learning method that operates by constructing a multitude of decision trees during training and outputting the mode of the classes (classification) or mean prediction (regression) of the individual trees. It builds multiple decision trees and merges them to get a more accurate and stable prediction.

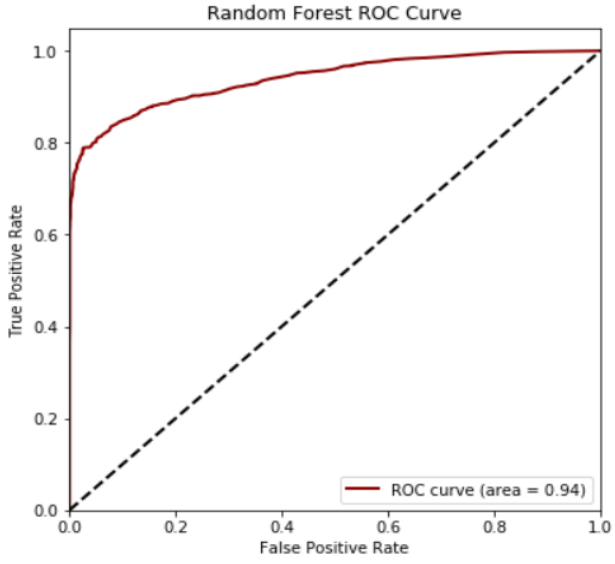
Advantages:

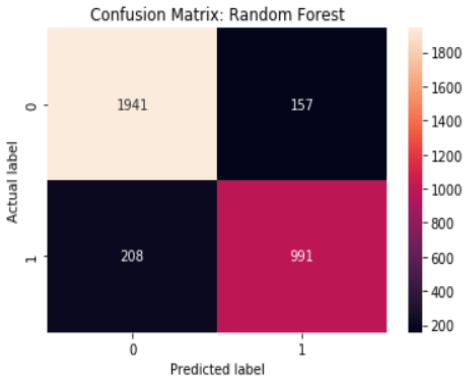
- Reduced Overfitting: By combining multiple decision trees, Random Forest reduces overfitting, especially when the individual trees are pruned properly.
- Robustness to Noise: Random Forest is robust to noise in the training data, as it averages out biases from individual trees.
- Handles Missing Values and Outliers: Random Forest can handle missing values and outliers well, without the need for imputation or extensive preprocessing.

Disadvantages:

- Complexity and Interpretability: While Random Forest provides high accuracy, the resulting model can be complex and difficult to interpret, especially when dealing with many trees.
- Computationally Intensive: Training a Random Forest model can be computationally intensive, especially for large datasets with many features, as it requires building multiple decision trees.
- Memory Consumption: Random forests can consume more memory due to storing multiple decision trees, especially when dealing with large datasets.

Random Forest Performance Metrics:





Accuracy:	0.8892
Precision:	0.8632
Recall:	0.8265
F1 Score:	0.8444
ROC AUC:	0.9535

8.4 Logistic Regression

Logistic Regression is a popular statistical model used for binary classification tasks. Despite its name, it's used for classification, not regression. It predicts the probability that an instance belongs to a particular class. The output is a probability score between 0 and 1, which can be converted into class predictions using a threshold (typically 0.5).

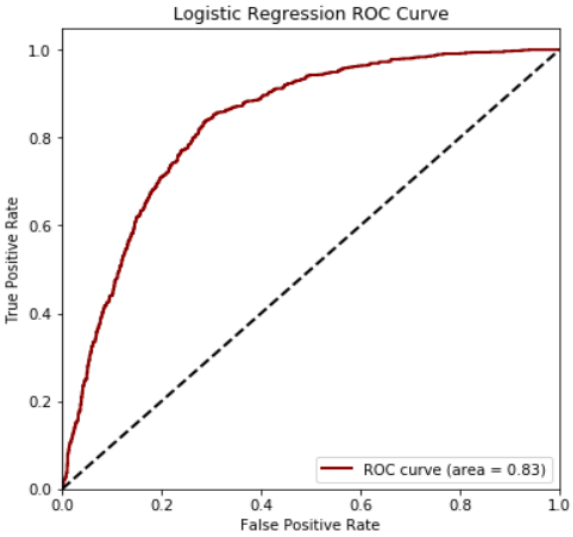
Advantages:

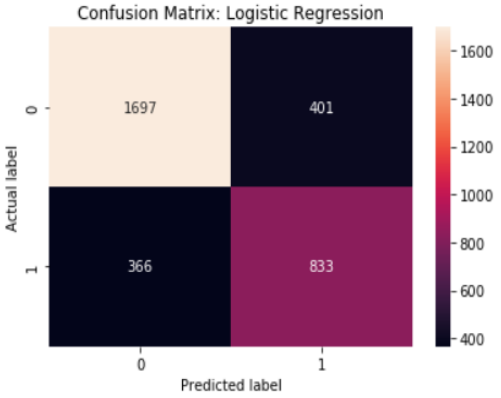
- It's efficient to train and can handle large datasets. This makes it suitable for situations where quick results are needed or where computational resources are limited.
- The coefficients in logistic regression provide insights into the impact of each feature on the outcome. Analysts can understand which variables are influential in predicting the outcome.
- Techniques like L1 and L2 regularization can be applied to prevent overfitting, allowing for better generalization to unseen data. This helps in reducing the variance of the model and making it more robust.

Disadvantages:

- Assumes a linear relationship between the features and the log odds of the response. It may not capture complex relationships in the data, such as interactions between variables.
- When the decision boundary is not linear, logistic regression may not perform well. It may struggle with datasets where the relationship between features and target is more complex.
- Outliers can disproportionately influence the coefficients and predictions in logistic regression. This can result in a model that is skewed towards these outliers.

Logistic Regression Performance Metrics:





Accuracy:	0.7673
Precision:	0.6750
Recall:	0.6947
F1 Score:	0.6847
ROC AUC:	0.8322

9. Project Results

9.1 Model Comparison

The table shows that the Random Forest classifier has the highest Accuracy and F1 Score, which suggests it balances Precision and Recall better than the other models.

It also has the highest ROC AUC, indicating it's superior in distinguishing between the classes. The Decision Tree and KNN models show comparable accuracy, but KNN has a better F1 Score and ROC AUC, making it the second-best model overall.

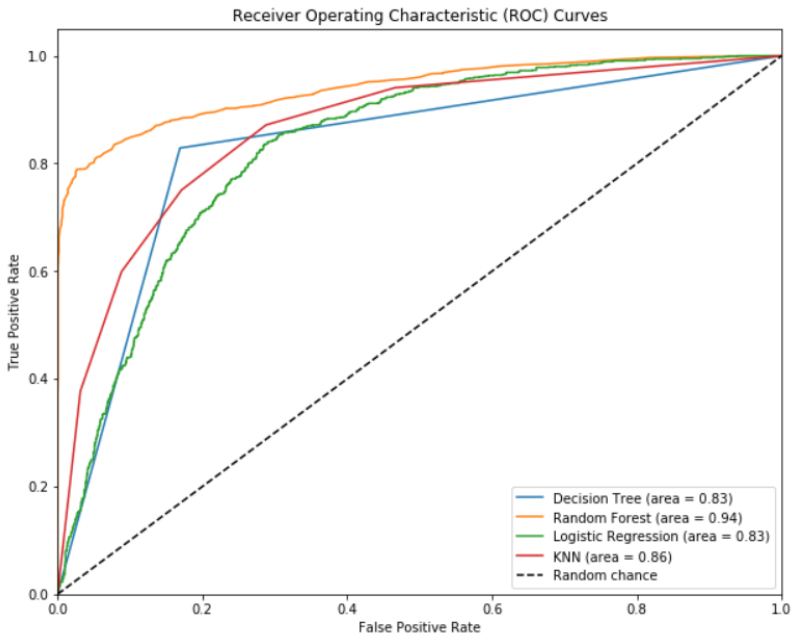
Logistic Regression has the lowest scores across all metrics, suggesting it may struggle with this dataset's complexity compared to the other models. These insights can guide model selection and further tuning for the task at hand.

	Model	Accuracy	Precision	Recall	F1 Score	ROC AUC
0	Decision Tree	0.830149	0.736842	0.829024	0.780220	0.829842
1	Random Forest	0.889293	0.863240	0.826522	0.844482	0.939592
2	Logistic Regression	0.767364	0.675041	0.694746	0.684751	0.832250
3	KNN	0.800425	0.714853	0.750626	0.732303	0.863856

9.2 Top Performer

This shows ROC curves comparing the performance of four different classification models: Decision Tree, Random Forest, Logistic Regression, and KNN.

The ROC curve plots the True Positive Rate against the False Positive Rate at various thresholds. The closer a curve is to the top left corner, the better the model's performance. The Random Forest model has the highest area under the curve (AUC), indicating the best performance among the models.



10. Conclusion

- Successfully developed a predictive model to analyze the presence and trends of wildfires. This was accomplished using data mining techniques and machine learning models.
- The model identified primary causes and geographical patterns affecting wildfires. This includes natural factors like temperature and human-induced factors, thus providing critical insights for managing and mitigating wildfire risks.

Challenges faced

- The project faced challenges in data cleaning and preparation, including dealing with a large initial dataset that required significant refinement to focus on relevant variables for the model.
- Selecting and tuning appropriate models to accurately predict wildfires presented difficulties. We tested various algorithms (Random Forest, Logistic Regression, KNN, Decision Trees), each with its own set of challenges, including handling overfitting and ensuring good generalization.
- While models like Random Forest showed high accuracy, they were computationally intensive and less interpretable, which can complicate their implementation in real-world settings where explainability is crucial for decision-makers.

11. Insights for Decision Making

- The Random Forest model, with its high ROC AUC, has proven to be exceptional in classification tasks, providing reliable performance for predictive purposes.
- These insights allow for the strategic allocation of resources to high-risk areas and the crafting of proactive measures, enhancing both the focus of risk-reduction strategies and the efficacy of resource distribution.

Impact of Project Outcomes

- Predictive models have shown early warning capabilities, facilitating timely evacuation and response, while the integration of new data ensures their continuous refinement to address evolving environmental conditions.
- This advancement aids operational decisions, equipping wildfire management to better prepare for the diverse challenges presented by varying geographic and environmental scenarios.

References

(https://www.ncdc.noaa.gov/)
NOAA's National Centers for Environmental Information (NCEI) offers climate and weather data for research.

(https://public.wmo.int/en/our-mandate/climate/weather-and-climate-information)
"World Weather and Climate Information" by World Meteorological Organization (WMO) global weather data and reports.

Kaggle/California_Wildframes/2013-2020
Fire.CA.gov/incidents
https://gis.data.ca.gov/