Sari Sarieddine – July 11, 2024

Forest Fire Prediction Model

Why this project?

Forest fires have been in the news more frequently in the recent years. With climate change, we are seeing more extreme weather events that lead to lives lost, communities destroyed, and people displaced. This project is a proof of concept of how a prediction model can look like, and the impact it can have to communities that are given the right resources in time to prepare for imminent danger from the spread of wild fires.

Executive Summary

This project aimed to develop a predictive model for forest fires in Canada using historical fire and weather data. The fire dataset, spanning from 1990 to 2022, was cleaned and merged with weather data from 430 selected weather stations. Key features for prediction included mean temperature, total snow, and total precipitation, with additional lag columns to forecast future fire occurrences. Missing data were handled through backfilling and front-filling techniques.

Two machine learning models were employed: a Decision Tree Classifier and a Random Forest Classifier. The decision tree model achieved an accuracy of 88.17% on test data, focusing on minimizing false negatives, which is crucial for health and safety. The random forest model, an ensemble of decision trees, achieved higher accuracy (91.68%) but was less interpretable. Both models showed strong predictive capabilities, with the random forest model offering higher accuracy and the decision tree model providing ease of interpretation. This project demonstrates the potential of machine learning techniques in predicting forest fires, aiding communities in better preparing for such events.

Sourcing the Data

The dataset used for this project were a combination of 3 datasets, sourced from the government of Canada's National Ministry of Forestry and Forest Fire as well as the Ministry of Environment and Climate Change.

Fire Dataset

The fire dataset was retrieved as geospatial *.shp* files from the ministry of Forestry and Forest fire. The dataset spans over 100 years, from 1910 to 2022, and had a total of 59,539 instances of forest fires.

The data collected included:

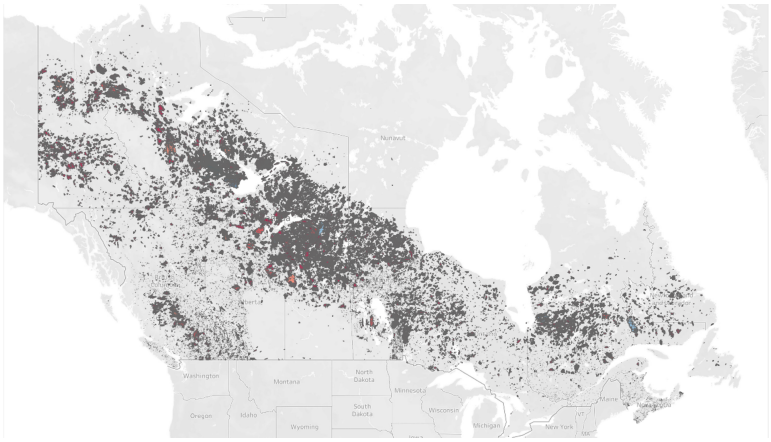
- Cause of fire,
- size of fire damage by hectare,
- Start/End date of fire,
- Geospatial location of fire.

For the purposes of this project, the only data points we needed from this dataset was the **Start/End date of fire**, and **Geospatial location of fire**.

Some cleaning was done on this dataset prior to moving onto merging. Cleaning involved:

- Removing Duplicate Rows,
- Removing rows with corrupt data(Months labelled as 0),
- Removing all rows for dates before 1990.

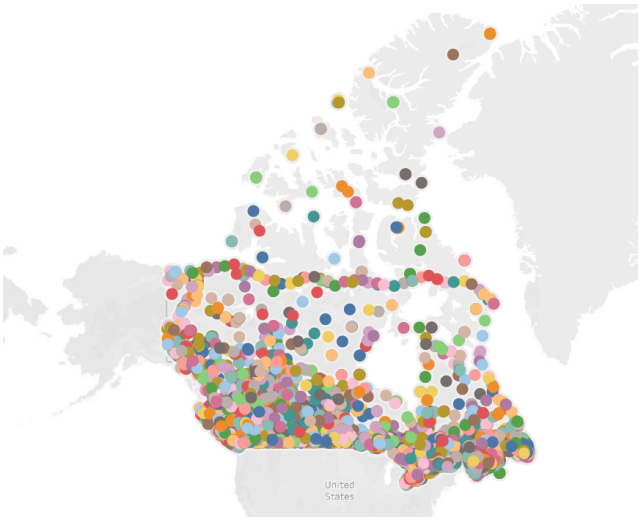
After cleaning, our final dataset has 26,875 unique rows to work with.



Map of forest fires in Canada between 1990 and 2020

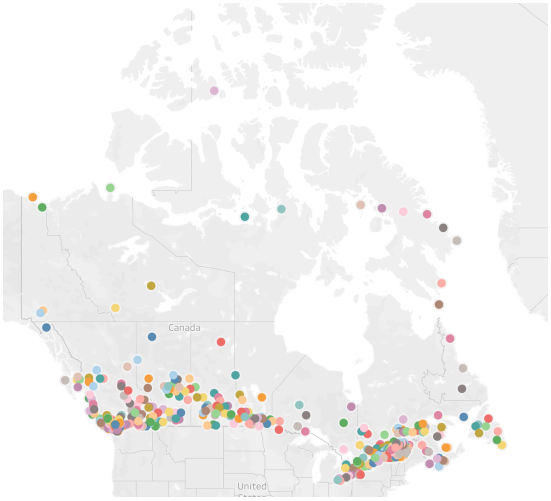
Weather Dataset

To collect the weather data, we first had to determine which stations we were going to select for our model. There are over 8,000 stations all across Canada, but due to the technological and time limitations involved with this project, we can only work with under 500 stations. To first reduce the sample size of our weather station data, we decided to filter out weather stations that don't have data past 1990. Then we proceeded with using **Density-based spatial clustering of applications with noise (DBSCAN)** to decide on the stations we want to use.



Map of weather stations across Canada

DBSCAN is an unsupervised model that works by clustering datapoints based on distance and labeling data determined as noise or outliers. By using DBSCAN, we were able to get 430 clusters which I then picked out 1 station from each cluster, giving me a sample size that is more manageable to work with given the limitations mentioned earlier.



Stations selected for modelling

We now have the stations to work with, the next step is to download the daily climate data from each of those stations from 1990 until 2020.

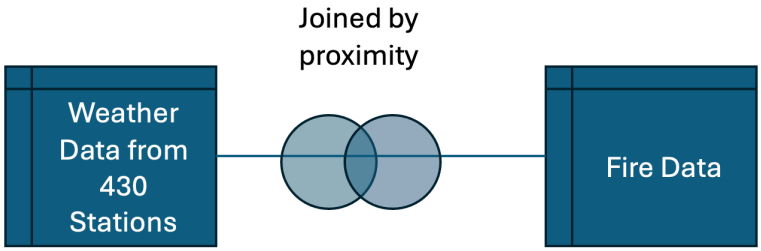
Total files downloaded were 360 *.csv* files.

Code used to download Weather data

```
# downloading weather data
for prov in weather_stations_list: #going through each province

    station= list(weather_stations['Climate ID'][weather_stations['Province']==prov].unique()) #going through each station from the province
    for id in statio (variable) date_range: Index

        for date in date_range: #selecting the date range for each province

            try:
                download=urlretrieve(url=f'https://dd.weather.gc.ca/climate/observations/daily/csv/{prov}/climate_daily_{prov}_{id}_{date}_P1D.csv',
                                     filename=f'Data/Weather_Daily/climate_daily_{prov}_{id}_{date}_P1D.csv')
                print(f'Downloading from {prov}')
            except:
                print('pass')
                continue
```

Combining Datasets

The modelling dataset we had envisioned was going to be a list of weather readings from each station along with a column on whether or not there was a fire near said station at a certain month of a certain year.

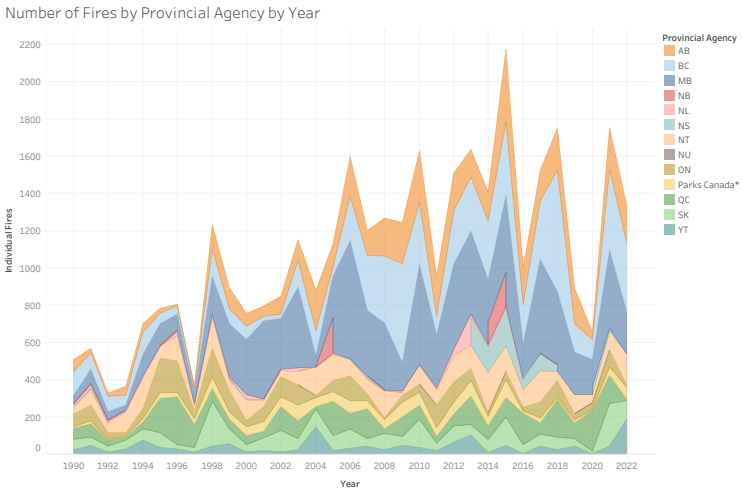


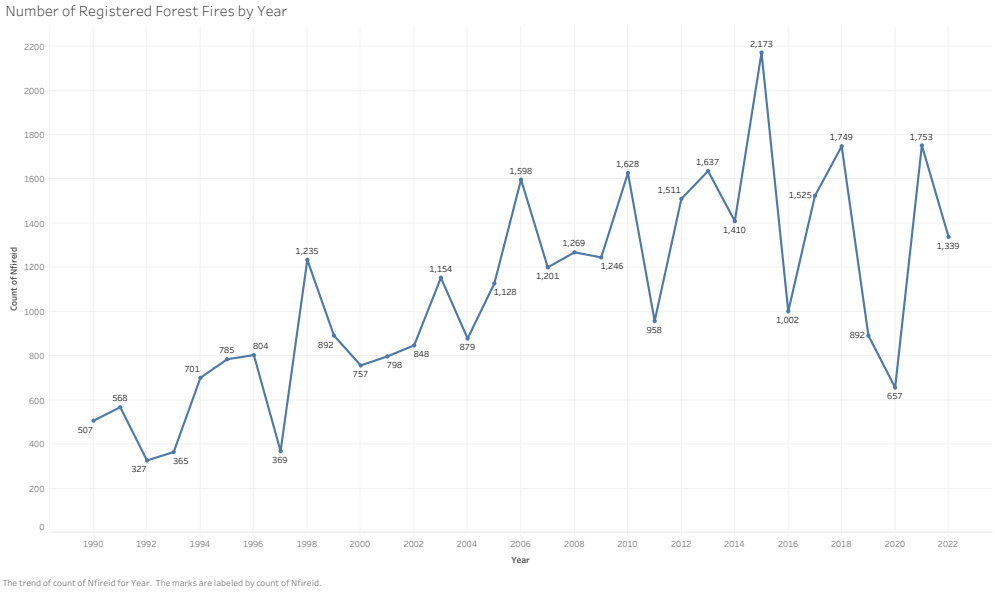
To achieve this, we had to first merge the weather station and fire data set by proximity. We did this by using another model called **k-dimensional tree (KD tree)**. The way this worked was by centring the fire coordinates and finding the 5 closest stations to it. In doing so, we were able to associate our fire data with 5 separate weather stations. This had the added benefit of naturally upscaling our positive result data, as the natural dataset would have had a larger set of negative result(no fire) data points.

Data Findings

While prepping the data for modelling, we looked at some trends relating to the fires and weather data. Here are some of the trends below:

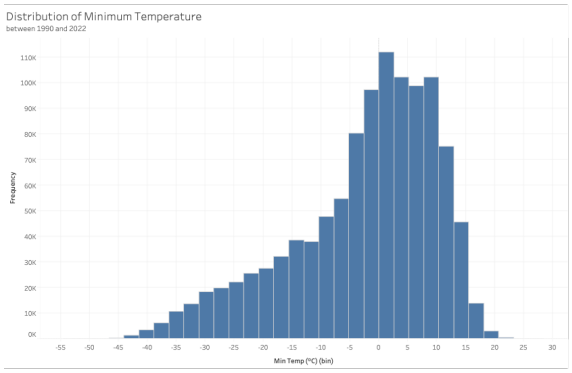
We can see that a majority of the fires occur in the western Provinces. This aligns with the map from above as we can see that the majority of the fires occurred between British Columbia and Saskatchewan.

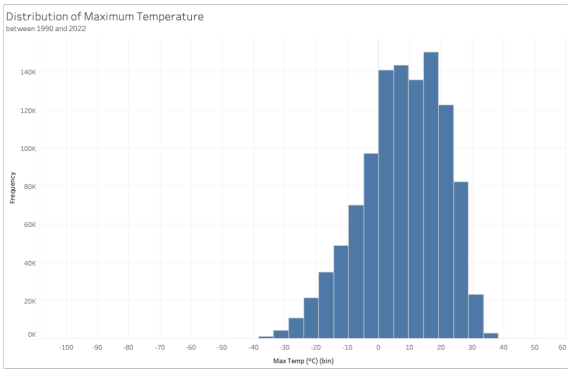




The trend of count of Nfireid for Year. The marks are labeled by count of Nfireid.

We can see an upward trend year over year of the number of fires annually. This is consistent with the increase extreme weather events we are seeing due to climate change.





We can see that there is a left skew of mean min and max temperatures throughout the years. This is also consistent with the increase in temperature we're seeing year over year.

Preparing for Modelling

When preparing for modelling, we decided through feature engineering to settle with these features below for our model to use to predict instances of fire:

Month	Year	Longitude (x)	Latitude (y)	Mean Temp (°C)	Total Snow (cm)	Total Precipitation (mm)

Our target column is labelled fire and simply indicates whether a fire occurred or not in boolean (1 for True/ 0 for False). When looking at our data we can see that we have some missing information.

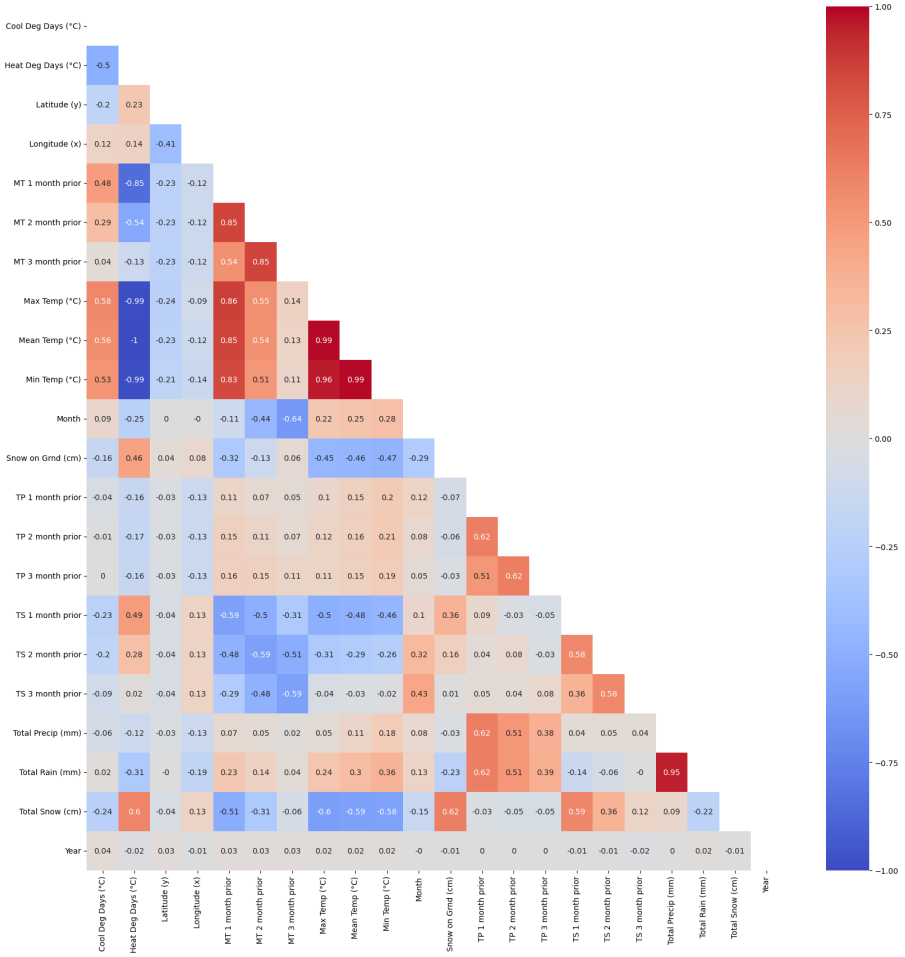
The missing information occurred at the point of collection, so there was no way to impute that data post collection. To account for the missing data, we decided to back fill and front fill - using data from previous and later months to fill missing data.

```
MONTH has 0.0% missing data
YEAR has 0.0% missing data
Longitude (x) has 0.13% missing data
Latitude (y) has 0.13% missing data
Mean Temp (°C) has 11.35% missing data
Total Snow (cm) has 15.33% missing
data
Total Precip (mm) has 9.7% missing
data
Fire has 0.0% missing data
```

Finally, we added 9 more columns with 3 months lag for **Mean Temp**, **Total Snow**, and **Total Precip**. This was done to allow the model to forecast

rather than predict current events. More information on how the preprocessing was done can be found on **daily_weather_preprocessing.ipynb**, **fire_preprocessing.ipynb**, and **Modelling.ipynb.**

Correlation Matrix



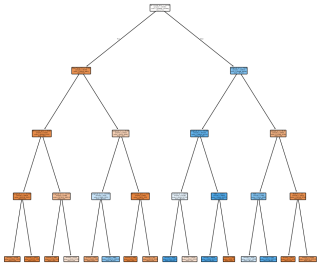
Looking at correlations to determine what features to drop

Modelling

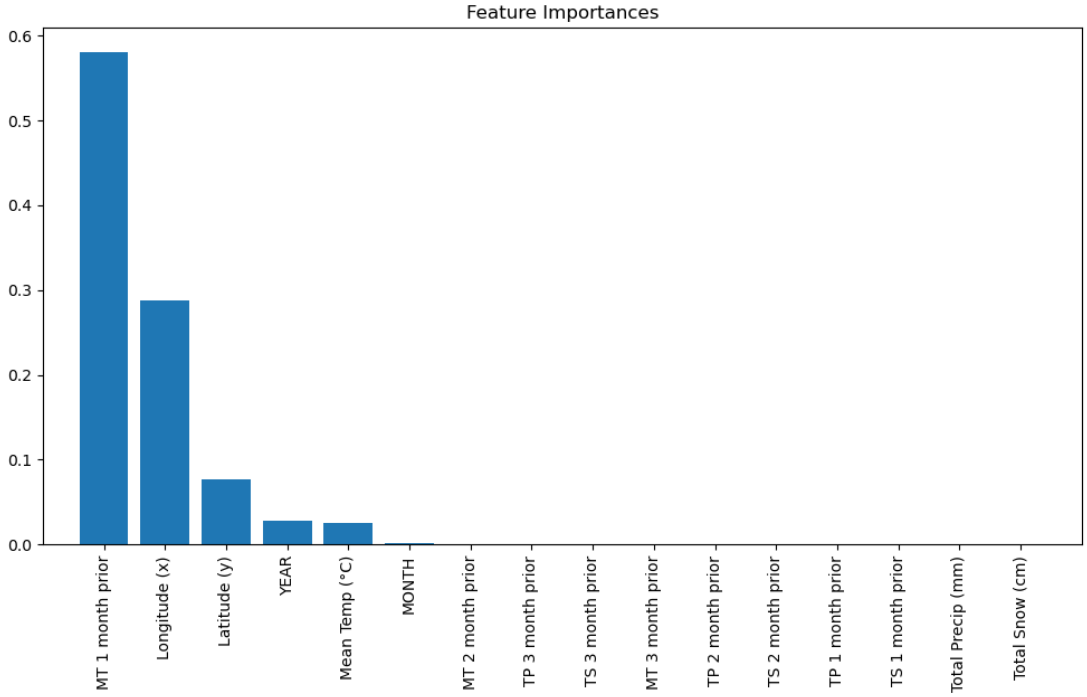
We decided on using two models; Decision Tree Classifier, and the Random Forest Classifier (These models were chosen for the their ability to predict using the data we have, and not for their forest-themed names.)

Decision Tree

We selected the decision tree model for its ability to make non-linear predictions, and find trends in the features that we may otherwise miss. Our decision tree model decided to focus on only 5 features, which is one of the limitations we have with this model. The more features we ask it to focus on, the more **overfitted** it may become. When a model is overfitted, it becomes so well trained on the training material, that it does a very poor job at predicting using new data.



Our model graphed. This is for illustration purposes only.

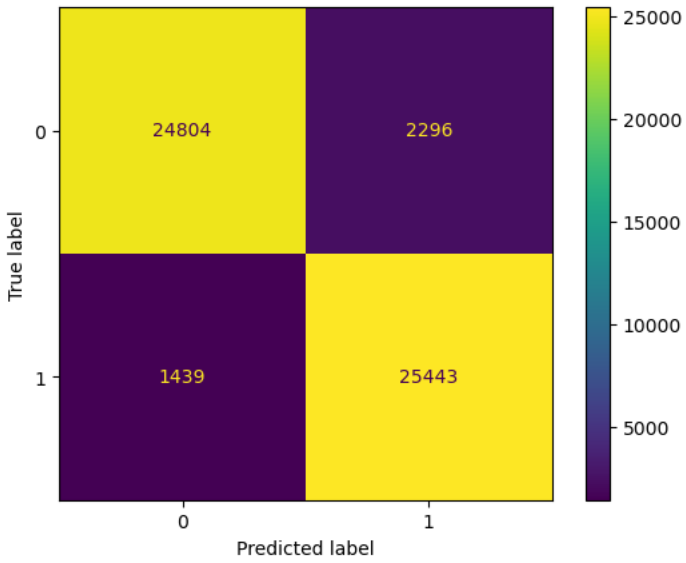


A graph of which features are valued most by our Decision Tree Model

We decided to accept that not all the features will be looked at for our decision tree model, and we were still able to get promising results!

Our model was able to correctly predict **88.02%** of our train data, and **88.17%** of our test data!

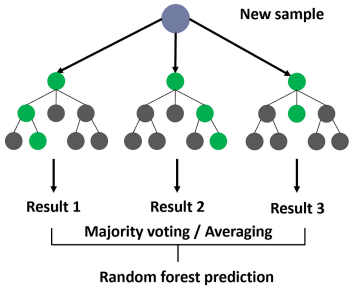
We can see on our confusion matrix on the right, our model correctly predicted **25,443** rows as **True** and **24,804** rows as **False.**



We can also note that the model predicted more **false positives** - predicting that there is a risk of fire when there isn't- than it did false negatives. This is a key feature for a model that is used to tackle health and safety issues, as we would rather the model be wrong in saying there is a fire, and not having one, than to be told there is no fire, and be underprepared when it occurs.

Random Forest Classifier

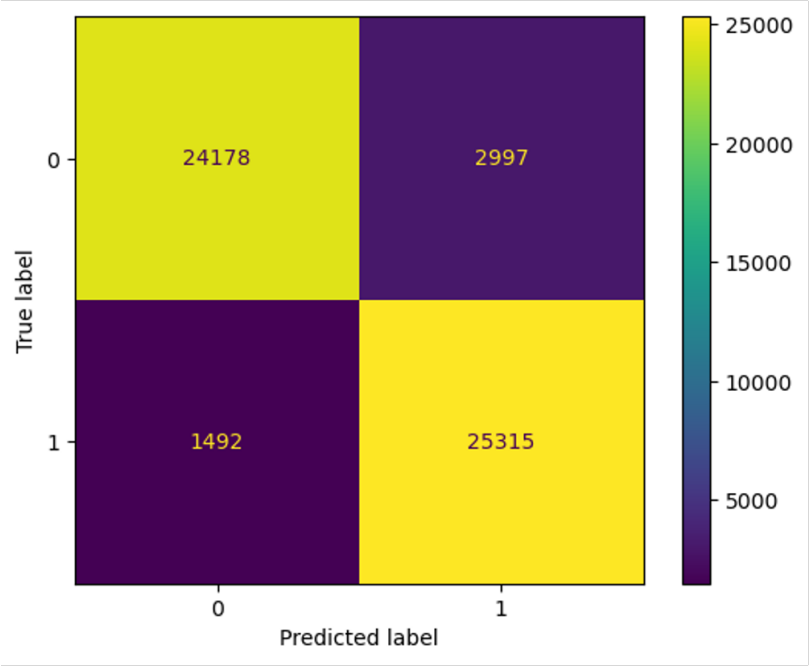
The next model we used was a Random Forest Classifier model. This model is known as a **blackbox model**, due to the difficulty in explaining how it came to its predictions. A Random Forest Classifier is an amalgamation of Decision Trees, using different features and feature combination, that bases its predictions on either



averaging or majority voting from the predictions of each decision tree within it. This model is usually more accurate than a regular decision tree, but due to the lack of explainability these types of models are not ideal for use in health and safety situations. To have accountability for the decisions made by these models, scientists must be able to simply explain what the model looks at when a decision is made.

In any case, we are still using this model to illustrate its effectiveness. This model was able to accurately predict **91.91%** of our training data and **91.68%** of our testing data.

We can see that the Confusion matrix on the right is not much different than the confusion matrix for our Decision Tree model. Despite this minimal change in scores, and because a random forest model takes all features into account, the probability predictions it makes will be more accurate.



Conclusion and Limitations

This project was a proof of concept on a prediction model that can aid communities in anticipating the risk of fire and allow them to plan accordingly. Using just weather related features, geographic data, and date data, our models are accurate enough to predict the environmental conditions that may lead to a forest fire. Adding features like **proximity to forests**, **forest density, humidity, old-growth vs second-growth forest,** and **a lag that exceeds 12 months** will help better predict the risk of a forest fire occurring. Adding lags to our targets will also allow us to predict whether a forest fire will occur further in advance.

Thanks and Gratitude

This project would not have been made possible without the support of my wife, Keisha Deoraj, the educational team at BrainStation, as well as the friends I made during the 3 month bootcamp.

Thank you all for the love and support that you graciously provided me. This has been nothing short of a life changing experience for me, from meeting new, wonderful people, to learning and mastering a technical skill in 3 months! I wish everyone the utmost success in their future!

Friends make the good times better and the hard times easier.