COMP 4311-FDE
Winter 2021
Final Project Report

Andrew Greer
0680891
Troy Walther
0698382

Introduction

Forest fires are becoming a more common problem in the world. They are a source of widespread ecological damage as well as a localised reduction in air quality due to the amount of carbon released into the atmosphere. Combined with ever warming temperatures on Earth thanks to climate change, we can expect forest fires to increase in severity. This has led to a need to effectively combat them before they can spread. Being able to algorithmically predict the occurrence of a forest fire or it's severity would be a massive advantage, as it would allow efficient allocation of fire fighting resources to fire-prone areas. Through our research, we attempted to find such a classification algorithm and see if it produces meaningful results, which improve upon the current predictions in the fight against forest fires in Canada.

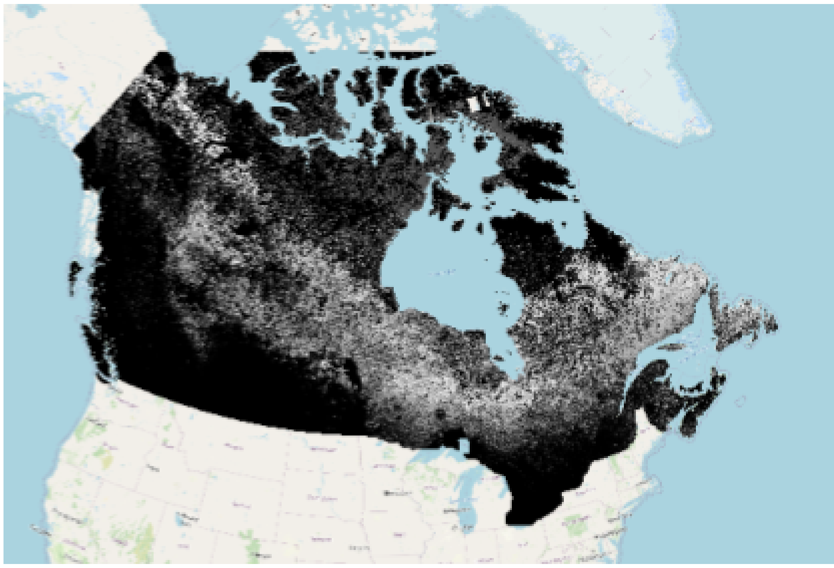
Data

All of our data in this project comes from the Canadian government. The Tree data is made available through Natural Resources Canada and Licensed using the Open Government Licence[1] . The Weather data is made available by Environment and Climate Change Canada using their custom license [2]. The fire data is made available by the Canadian Wildland Fire Information System Datamart.

<u>Tree Data</u>

The data on the populations of trees across Canada comes in the form of GeoTiff files, which are special files designed to be viewed in GIS software such as ARCGIS. Figure 1 shows an image demonstrating how a Geotiff file looks when displayed in a GIS program called QGIS.

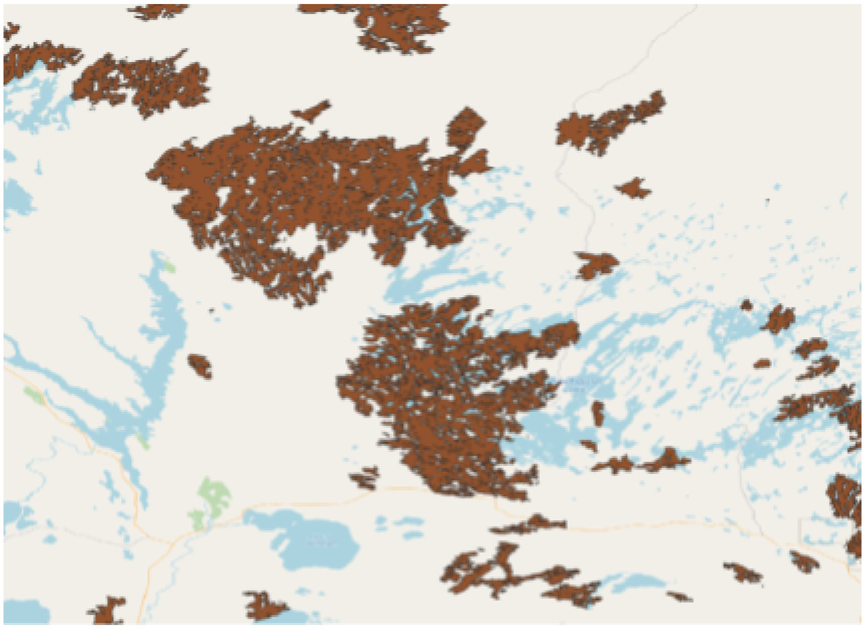
The Geotiff file format is a raster file, meaning each pixel contains a value. Each pixel in the file represents the percentage of the area which that tree species and genus occupies, -9999 represents pixels where no data exists. Since there is a pixel for each point across all of Canada, the number of instances is colossal.



Figure(1) Pice_Mar tree data in GeoTiff format

Fire Data

The fire data comes in another GIS format known as a shape file. Shape files are a vector format which contains information about the shape of a feature. The vectors of a shapefile can contain many features. The number of fires per year varies, however there are most often around 800-2000 fires per year from 2011-2015. As can be seen this format represents the area burned using polygons from the vectors in the table. Each shape has an associated table which contains a series of points needed to build the shape. The X and Y values are incredibly important for tracking the location of the fires.



Figure(2) A shapefile loaded in QGIS

The features of the fires are listed by the CWFIS[4] as follows:

FID- The unique feature ID
SHAPE- The shape of geometry
YEAR - The year of the fire
NFIREID: National fire ID, unique for each fire (important)
BASRC: Burn Area Source (Who provided the fire data)
FIREMAPS: method used to identify fire (field survey, satellite, etc...)
FIREMAPM: how the map data was captured (same as above)
CAUSE- the cause of the fire split into 4 categories

- 1 misc. Known cause but doesn't fit other causes
- 2 Lightning
- 3 Industry (Oil and Gas, prescribed burn, forestry)
- 4 Human Activity (Campfires, Arson, etc...)

BURNCLAS: area burned (1:25%, 2:50%, 3:75%, 4:100%, if not provided assumed 4)
SDATE: Start Date
EDATE: End Date
AFSDATE: Date Reported
AFEDATE: Agency end date(same as end date)

CAPDATE: Capture date of data

POLY_HA: Total Area burned in Hectares

ADJ_HA: Total Adjusted Burn Areas

And a few other misc. Features. Some of these columns have 0 values for the whole file. In particular some start dates are missing, however if other dates are intact other dates can be used. Overall on average we had to drop 50-100 fires due to missing dates.

Weather Data

The weather data comes in a regular CSV format and contains data from weather stations all across Canada split into monthly information. The number of weather station readings per month is normally around 1000-1400. The Features are as follows:

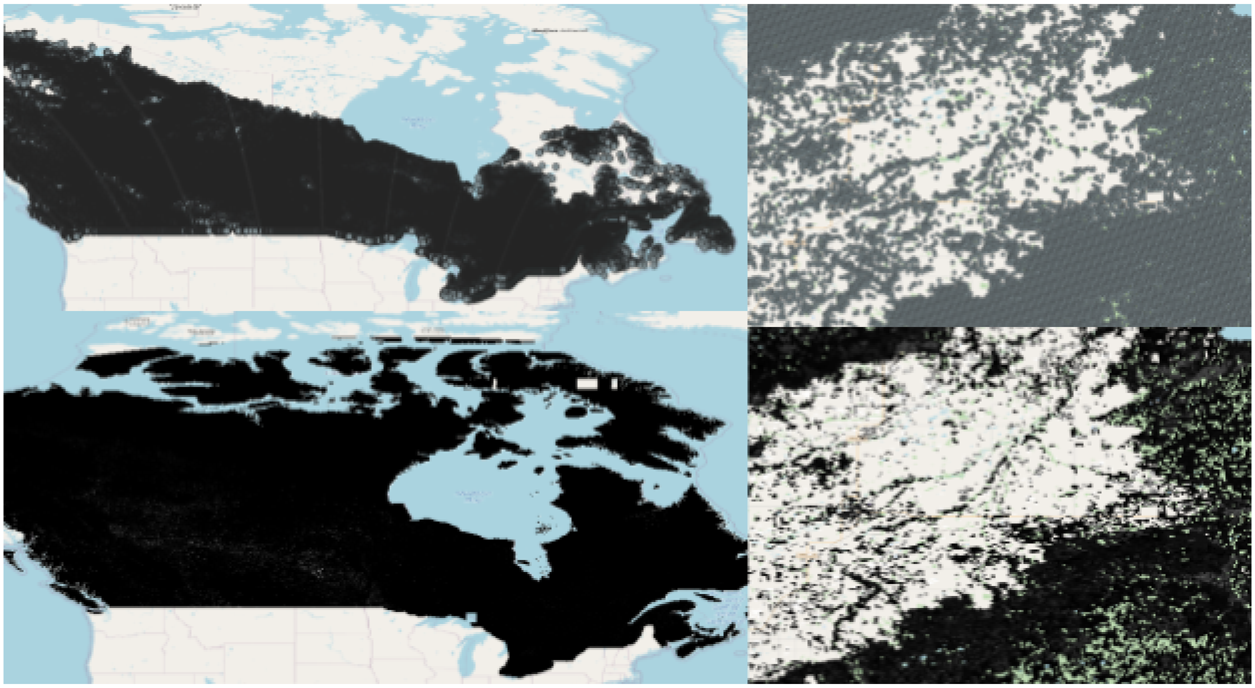
Long:	The longitude in EPSG:4326 format
Lat:	The latitude in EPSG:4326 format
Stn_Name:	Station Name
Clim_ID:	The unique identifier for that climate region
Prov:	The province of the weather station
Tm:	Mean Temperature
DwTm:	Days without Valid Mean Temperature
D:	Mean Temperature difference from Normal (1981-2010) (°C)
Tx:	Highest Monthly Maximum Temperature (°C)
DwTx:	Days without Valid Maximum Temperature
Tn:	Lowest Monthly Minimum Temperature (°C)
DwTn:	Days without Valid Minimum Temperature
S:	Snowfall (cm)
DwS:	Days without Valid Snowfall
S%N:	Percent of Normal (1981-2010) Snowfall
P:	Total Precipitation (mm)
DwP:	Days without Valid Precipitation
P%N:	Percent of Normal (1981-2010) Precipitation
S_G:	Snow on the ground at the end of the month (cm)
Pd:	Number of days with Precipitation 1.0 mm or more
BS;	Bright Sunshine (hours)
DwBS:	Days without Valid Bright Sunshine
BS%:	Percent of Normal (1981-2010) Bright Sunshine
HDD:	Degree Days below 18 °C
CDD:	Degree Days above 18 °C

Other than Longitude, Latitude, Station Name, Clim_ID and Province, All fields contain Null values, with some fields (such as BS%) containing no values at all. This is likely due to the legacy nature of these readings, as they have tracked the same fields since as early as 1840.

Methods & Tools

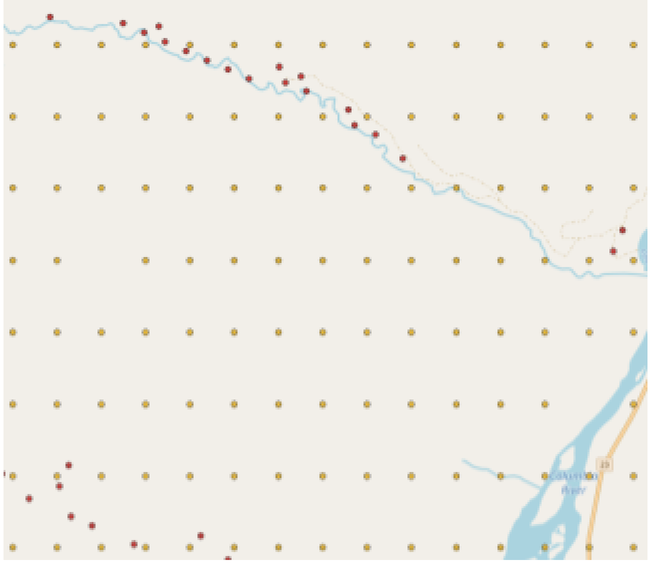
Tree Data:

The first step in processing the tree data was converting from a GeoTiff file to a CSV. The easiest way to do this (And potentially the only way) is Vectorization by using the value of each pixel as a value for the vector. This is done using QGIS Vector creation from a raster tool, This produces a CSV with an X, Y, and percentage value (Note: when converting the value 0% must be Ignored or the resulting CSV will be far too large being as large as 25GB). This must be done by hand for all 71 Genus-Species GeoTiff Files.



Figure(3) The Popu Bal Raster file (bottom) when converted to point vectors (top)

After being converted into a CSV the number of points must be reduced as there are currently too many points to assign each tree to a fire. This simplification is done using a mixture of 5 scripts. First any files over 1GB must be split into slices using split.py, We split them into Slices of 15. Once that is done, all CSV files have there X and Y coordinates rounded to 2 decimal places (Reduced from 12), the percentage values are then averaged across all unique coordinate groups and that data is output to new CSV files using average_simplify.py (Difference Shown in figure 4). Then the large files may be recombined back into one file and reprocessed, using combine_parts.py then average_simplify.py. Then the Genus (first four letters Pice, Popu, Pinu etc...) may be combined using combine_genus.py and then getting the average of that Genus using Average_Simplify.py. After getting these Genus CSV files, it is finally time to combine them all together using Master_Combine.py. The master combine file uses the largest Genus percentage at each coordinate to get the unique tree values at each coordinate pair as shown in figure 4 (Pice_Mar vs Pice). The resulting single file of all this processing is Tree_Data.csv.All of this processing takes 32 Gigabytes of CSV files, and turns it into a 300MB CSV file with 12 459 301 unique coordinates and their associated tree genus and percentage. This data processing was done utilizing a library known as NJIT, which directly compiles python code into C code when written in a special way, this compilation as C code allows for much faster processing of data. We also implemented multiprocessing to further decrease the time. Times for processing the tree files were tracked and the total processing time with multiprocessing was 84 hours. The total compute hours based on a 6 core CPU (12 logical processors) is 505.17 core hours, or 21 core days.

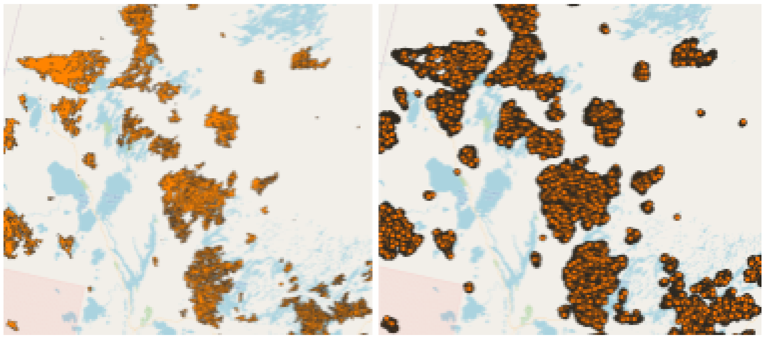


Figure(4) The points resulting from the simplification, And the combining of Genus species. (Red is unprocessed Pice_Mar)

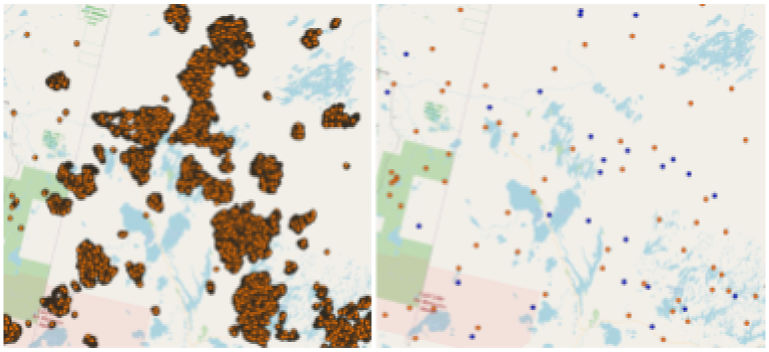
Fire Data:

The first step of processing the fire data is to convert it from a shape file to a regular point map. This is done using QGIS *Vector -> Geometry Tools -> Extract Vertices.* Once these vertices have been extracted you can export the fire with it's coordinates to a CSV file.

Once the fire data is in a CSV, any rows which do not contain at least one of SDATE, AFSDATE, or EDATE, are dropped as fires without dates cannot be merged with weather data. Then the fires are grouped by their Fire ID, and the median coordinate of each fire ID is returned. Any Fires missing a SDATE but having a AFSDATE or EDATE then have those values set as that fires start date (as these values are normally within the same month). Then negative cases



Figure(5) Fire Data before and after Vectorization



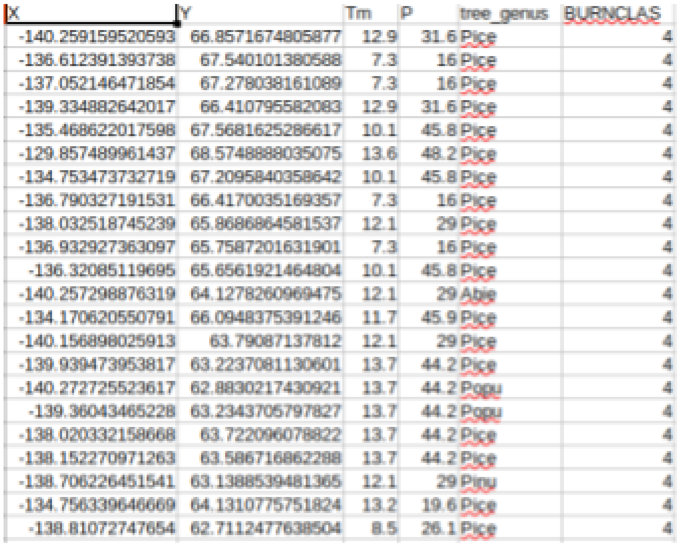
Figure(6) Before and after fire grouping are created using random coordinates across Canada and subtracting the fire coordinates to create a list of negatives equal to the size of the list of fires. All these actions are performed using the process_fire.py script. Once all these steps have finished the resulting DataFrame is exported as fire_processed_year.csv

Weather Data:

The weather data was by far the easiest data to process. First you download one year of monthly summaries of weather, then you run the process_weather.py script. The process weather.py script takes in a year (folder name) and outputs a merged file of that years weather data all in one file sorted by month with missing data imputed. First all twelve months are combined into one DF with a column specifying that readings date. Since weather stations are physical locations, any missing precipitation or mean temperature data is taken from the physically closest weather station within that same month. Once any missing precipitation or Temperature data is imputed, the file is exported as Climate_Sumamries_year.csv.

Final Fire Data:

After completing all the above processing the CSV's are ready to be merged into 1. The file Merge_WTF (merge weather, trees, fire) is run with the desired source files in the same directory. Once a year is provided the program opens the Tree_Data.csv, Climate_Sumamries_year.csv, and fire_processed_year.csv are all opened and stored in memory. First each fire is assigned a tree Genus based on the X and Y coordinates. Then the fire is assigned a mean temperature and precipitation value based on the nearest X and Y in the same month. These files are then exported as fire_data_year.csv and contain the features "X", "Y", "Tm", "P", "tree_genus", "BURNCLAS". These datafiles are what we use for our learning and look as shown in figure (7).

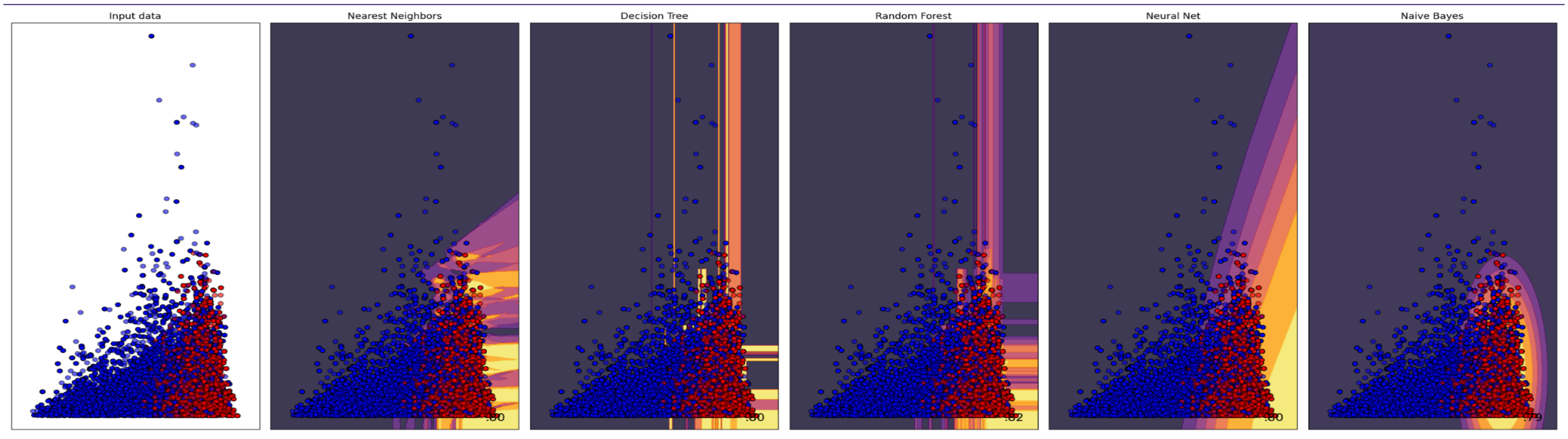


X	Y	Tm	P	tree_genus	BURNCLAS
-140.259159520593	66.8571674805877	12.9	31.6	Pice	4
-136.612391393738	67.540101380588	7.3	16	Pice	4
-137.052146471854	67.278038161089	7.3	16	Pice	4
-139.334882642017	66.410795582083	12.9	31.6	Pice	4
-135.468622017598	67.5681625286617	10.1	45.8	Pice	4
-129.857489961437	68.5748888035075	13.6	48.2	Pice	4
-134.753473732719	67.2095840358642	10.1	45.8	Pice	4
-136.790327191531	66.4170035169357	7.3	16	Pice	4
-138.032518745239	65.8686864581537	12.1	29	Pice	4
-136.932927363097	65.7587201631901	7.3	16	Pice	4
-136.32085119695	65.6561921464804	10.1	45.8	Pice	4
-140.257298876319	64.1278260969475	12.1	29	Abie	4
-134.170620550791	66.0948375391246	11.7	45.9	Pice	4
-140.156898025913	63.79087137812	12.1	29	Pice	4
-139.939473953817	63.2237081130601	13.7	44.2	Pice	4
-140.272725523617	62.8830217430921	13.7	44.2	Popu	4
-139.36043465228	63.2343705797827	13.7	44.2	Popu	4
-138.020332158668	63.722096078822	13.7	44.2	Pice	4
-138.152270971263	63.586716862288	13.7	44.2	Pice	4
-138.706226451541	63.1388539481365	12.1	29	Pinu	4
-134.756339646669	64.1310775751824	13.2	19.6	Pice	4
-138.81072747654	62.7112477638504	8.5	26.1	Pice	4

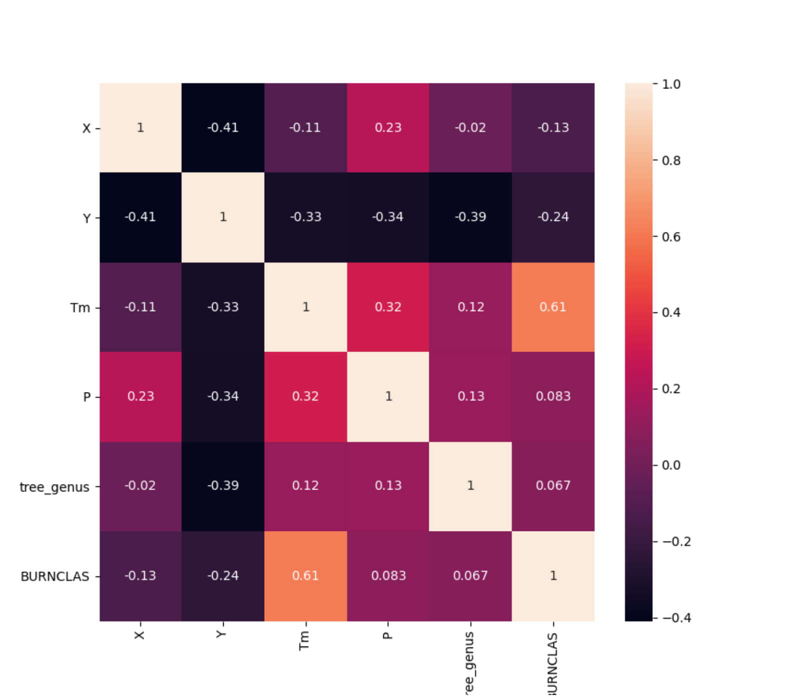
Figure(7) The final data after all pre-processing has finished.

Results and Methodology

For the actual ML on the data, we evaluated a couple of classification algorithms such as K-nearest neighbors, Decision Tree, Random Forest, Neural Net, and Naïve Bayes (see classifier_selection.py). Here, we are using data from 2011 through 2015, created a test-train split, and training the five different classifiers (*x* axis is mean temperature, *y* is precipitation)

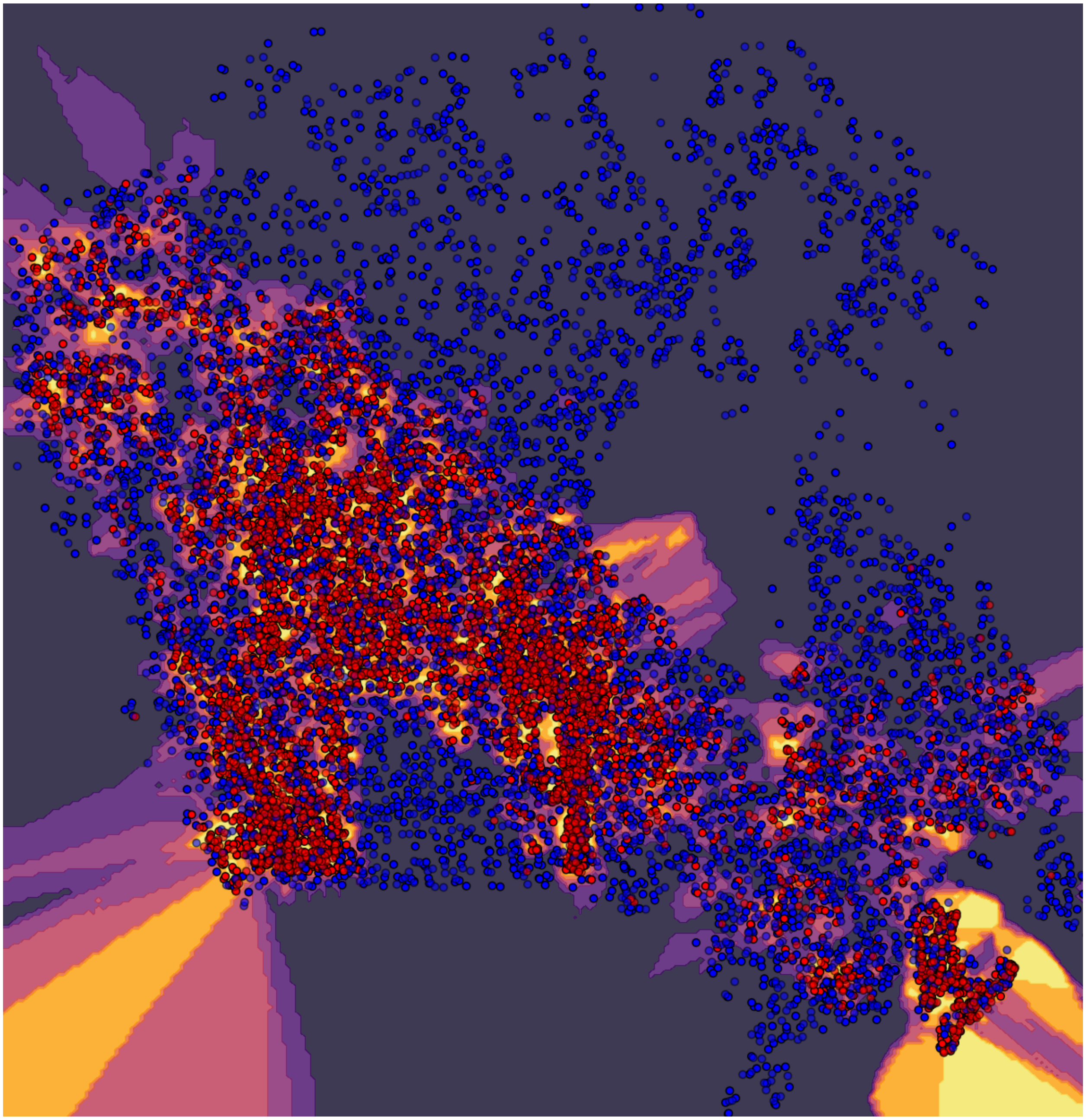


We chose to evaluate the classification algorithms on precipitation and temperature versus incidence of fire as our class variable. We chose these 2 features because matlib.pyplot only lets you plot 2D data (as far as I'm aware, there's probably a way to graph the entire feature-set but for the sake of picking an algorithm, I decided not to bother) and precipitation and mean temperature have the strongest correlation of fire severity (see correlation matrix below).



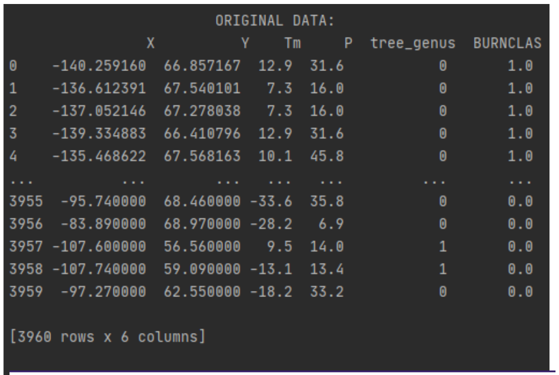
This correlation heatmap shows a strong correlation between mean temperature and risk of fire. Tree genus and precipitation had a much lower correlation.

After running the 5 classification tests on just those 2 variables on different years as well as the combined dataset, we decided that the best approach would be to use Random Forest, as it consistently had the highest accuracy.



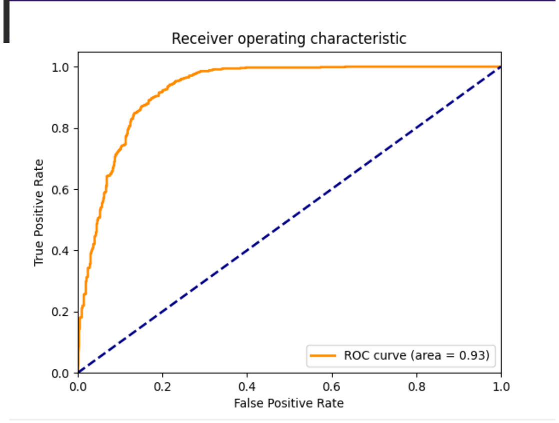
Just for the sake of visualisation, we used K-nearest neighbors using longitude and latitude as our *x* and *y* variables to better visualise the data distribution by geographic location.

For our training, we used the same method of splitting the data using sklearn.train_test_split on the 2015 data. We found that using $m = 2$ features per tree worked best, using a max tree depth of 11 and 20 n-estimators for the number of trees. This yields us an accuracy consistently above 85%.

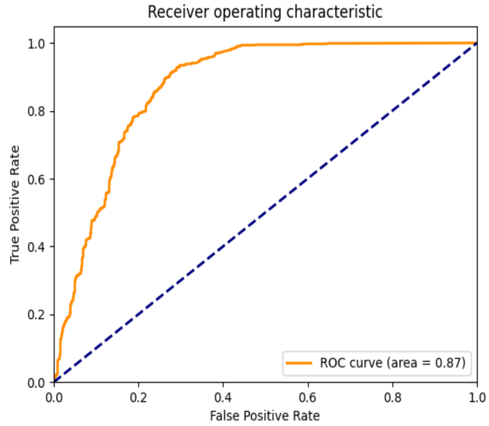


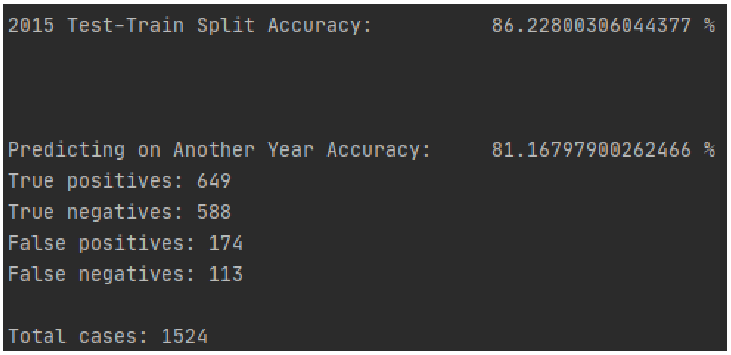
```
                      ORIGINAL DATA:
                X          Y    Tm     P  tree_genus  BURNCLAS
0    -140.259160  66.857167  12.9  31.6           0       1.0
1    -136.612391  67.540101   7.3  16.0           0       1.0
2    -137.052146  67.278038   7.3  16.0           0       1.0
3    -139.334883  66.410796  12.9  31.6           0       1.0
4    -135.468622  67.568163  10.1  45.8           0       1.0
...          ...        ...   ...   ...         ...       ...
3955  -95.740000  68.460000 -33.6  35.8           0       0.0
3956  -83.890000  68.970000 -28.2   6.9           0       0.0
3957 -107.600000  56.560000   9.5  14.0           1       0.0
3958 -107.740000  59.090000 -13.1  13.4           1       0.0
3959  -97.270000  62.550000 -18.2  33.2           0       0.0

[3960 rows x 6 columns]
```

Here is the resulting ROC curve for this methodology.

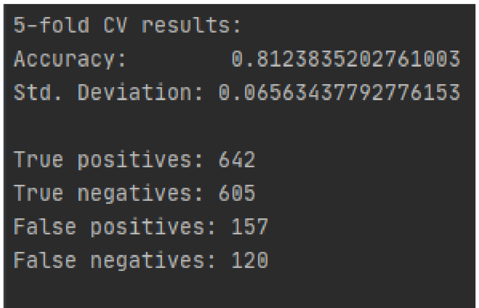


We decided that we should test the applicability of our model by training on 2015's data and predicting on another year, in this case 2011 (see below). The accuracy drops to ~80%, with a somewhat worse but still acceptable false positivity rate. Curiously enough, performing feature scaling hurts the accuracy of this method of prediction by ~5%, so we opted not to scale the training set



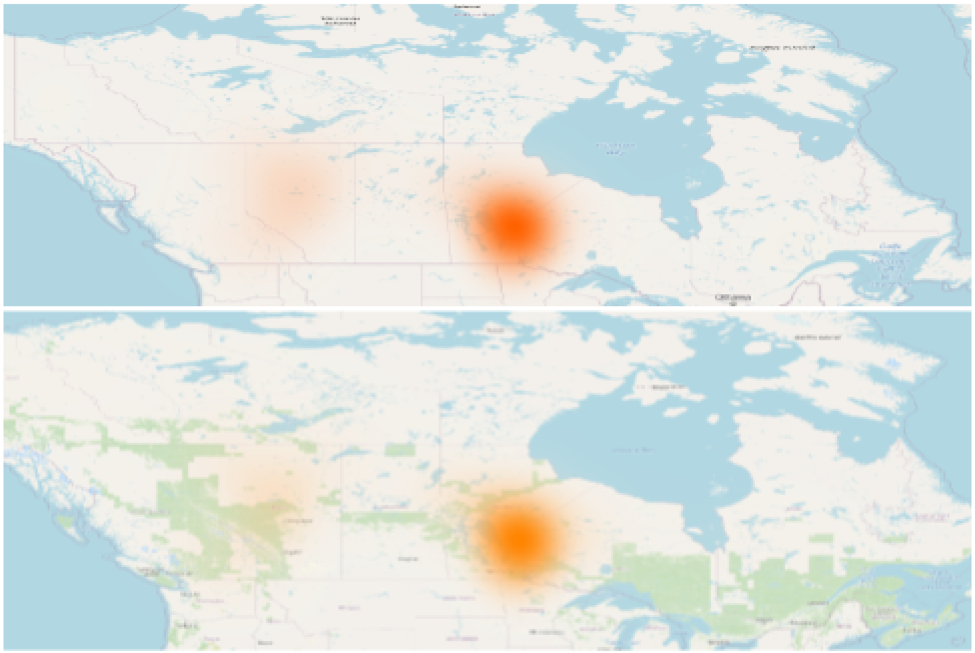


```
2015 Test-Train Split Accuracy:          86.22800306044377 %



Predicting on Another Year Accuracy:     81.16797900262466 %
True positives: 649
True negatives: 588
False positives: 174
False negatives: 113

Total cases: 1524
```



```
5-fold CV results:
Accuracy:        0.8123835202761003
Std. Deviation: 0.06563437792776153

True positives: 642
True negatives: 605
False positives: 157
False negatives: 120
```

We also performed some 5-fold cross validation on the same classifier with these results.



Figure(8) Predicted Top, Actual Bottom; year 2011 heatmap

Tools/Libraries

Scikitlearn
Pandas
Numpy
Seaborn
Matplotlib.pyplot
NJIT

Discussion and Conclusion

In this project we attempted to use a machine learning model and data made available by the Canadian government to predict the probability of a forest fire starting at a given location. We reduced the data on trees across Canada to one file containing the type of tree at 2.5km intervals. We merged and imputed the monthly weather data into yearly intervals and Removed unusable fire data. We then combined all three of the files into one file per year containing the weather data, the fire data, and the tree data. We then ran a program to judge the accuracy of various machine learning models and ultimately decided to use Random Forest for it's probability capabilities. We then tweaked the variables and ran ROC analysis until we were happy with the false positive rates. We then graphed the prediction on a year other than the year the model was trained on and graphed the result in order to compare it to the true fire locations in Canada that year. In our testing, one of the best predictors of forest fires is the mean temperature of a given area. As temperature increases globally due to climate change, we can say with near certainty that the incidence of wildfires in Canada will continue to increase. Of the 5 years of data we used from 2011-2015, 2015 had the most fires at 3960 instances (2011 had 1524 fires, 2013 had 2400 fires). As a result, it is our conclusion that the number of forest fires will trend upwards in the future.

Limitations

One of the major limitations with this work is our access to information on major factors in forest fires. The two primary causes of forest fires are Lightning strikes and Humans. British Columbia found that 60% of forest fires are caused by lightning, and the other 40% are human caused. 40% of fires being caused by humans means that without population data and storm data, the biggest limitation in the prediction of forest fires is the semi random nature of them. Temperature is strongly correlated with forest fires, and low precipitation is a reasonably good predictor, however the causes can strike many similar areas that non-fires and fires can appear near identical. This limitation has led to a high number of false positives.

Future Work

In the future expanding the data to include the aforementioned Storm and Population data could increase the accuracy of the model. This data could potentially lower the number of false positives. As well we would like to do more analysis or comparisons between our model and the Canadian governments publicly available fire predictions. Ultimately this dataset is interesting and there seems to be more work that could be done with it beyond what we covered in this project.

References

1. *Open Government Licence - Canada | Open Government, Government of Canada*. (2019). Open Canada.ca. https://open.canada.ca/en/open-government-licence-canada

2. *Licence Agreement for Use of Environment and Climate Change Canada Data - Climate - Environment and Climate Change Canada*. (2021). Climate Canada. https://climate.weather.gc.ca/prods_servs/attachment1_e.html

3. Wikipedia contributors. (2021, April 14). *Shapefile*. Wikipedia. https://en.wikipedia.org/wiki/Shapefile

4. https://cwfis.cfs.nrcan.gc.ca/downloads/nbac/nbac_2019_r9_20200703.shp.pdf