Fire Management Project

Team 2

Deepanshu Bhasin, Gansen Deng, Yanli Li

Zerun Xiao, Kexin Yan

Western
UNIVERSITY · CANADA

Contents

EXECUTIVE SUMMARY

The goal of this project is to optimize the resource allocation by finding distinct populations when dispatching resources to fires and finding the factors that influence them.

The dataset we used for the analysis is about the forest fires in Ontario's North West Region between Fort Frances and Kenora. We first scored the data quality using seven key dimensions - completeness, validity, timeliness, consistency, uniqueness, conformity and accuracy, and the overall score was 79.47 out of 100. After that, we cleaned the data by handling missing values and removing inconsistent observations to prepare data for further analysis.

On performing exploratory data analysis, we found that most of the fires are caused by lightning. Besides this, A&G(Air and Ground attack) and GPP(Ground Attack, Power Pumps) are the two most popular methods for handling fires. We also found that the fuel type present in a geographical area varies over time. Another interesting thing was that human-caused fires have a larger spread rate than lightning-caused fires.

Finally, we tried to fit statistical models on the cleaned data to find distinct populations and corresponding influential factors. We first used Hierarchical Clustering, Partitioning Around Medoid, and K-prototypes to separate the data into three classes. Then we used the Multinomial regression model and the Random Forest model to build multiclassification models by treating the class as the response.

The analysis results showed that the main difference between the three classes is due to the initial attack type, namely weak ground-based action, strong ground-based action and air-ground-combined action, respectively. Moreover, **FUEL_TYPE**, **RESP_GROUP** and **BHE_TO_UCO** are the core factors affecting action types. Therefore, in order to dispatch the resources more efficiently, we can arrange the resources according to **FUEL_TYPE** and **RESP_GROUP**.

1 INTRODUCTION

Wildland fire is an inherent part of the land, and it is also an important indicator of forests' health. Some wildland fires have beneficial effects such as eliminating disease and refreshing the habitation. However, an intensive fire can cause much damage, including soil erosion, death, air pollution, and economic loss [admin, 2018]. In the worst case, an unpredictable fire might get out of control, and the organization or government would fail to manage it. In Ontario, the number of wildland fires varies greatly. The daily count of fires ranges from less than 5 to more than 185 [McFayden, 2020]. Therefore, fire management plays a vital role in forestry as it helps to protect the forest and guides for the usage of the resources.

Ontario manages the fire using an appropriate response found by evaluating conditions such as risk, benefits and the cost. However, there are many variables to be taken into consideration. So, statistical modelling can be used in the evaluation process to find the optimal resource allocation solution.

This project aims to find the pattern of forest fires by performing statistical analysis to dispatch the resources efficiently. There are two main tasks in this project to find the solution to resource allocation. The first one is to find the distinct populations of resources used to handle fires, while the second one is to find the factors that are influential to different populations.

For the first task, it is proposed to perform clustering according to the response characteristics, which means that variables related to dispatching resources are to be taken into consideration. Hierarchical Clustering (HC), Partitioning Around Medoid (PAM) and K-prototypes are used to conduct the clustering task. HC outperformed the other two methods based on the performance of cross-comparison. Thus, the result of HC is used in the second task. To solve the second task, we treated the result of clustering as the response variable and built a classification model to identify the important features influencing it. Three multi-classification methods are implemented to check feature importance-multinomial regression with the LASSO penalty, multinomial regression with BIC, and random forest. Variables **FUEL_TYPE**, **RESP_GROUP** and **BHE_TO_UCO** are significant in affecting the clustering results.

2 DATA AND STUDY AREA

2.1 Data quality

The dataset used for analysis is provided by the Aviation Forest Fire and Emergency Services, Ontario Ministry of Natural Resources to study the forest fires in the Fort Frances and Kenora districts. The provided dataset contains 3465 observations and 93 variables.

The analysis aims to remove bias and use empirical data to create actionable recommendations and predictions for the future. But, data is only meaningful and usable if it is of good quality. If the quality of the data is poor, it will have bad business consequences. Bad data quality can lead to global catastrophes, including misconceptions of strategies, economic loss and reputation risk. To guarantee our data is of good quality, it is necessary to measure the effectiveness of our data. The algorithm for scoring the data quality is based on below key dimensions:

1. Completeness: we measured it by calculating the ratio of missing value. And the completeness score is 1 – missing value ratio. In our dataset, it has approximately 31.5%, so the score for completeness should be 68.5%.

2. Validity: we counted the number of rows that do not exceed allowed ranges. There were some unreasonable data in the timeline of dealing with fires, so lifetime variables are created for different periods in the life of a fire. There were 3.9% of fires with some lifetime variables is out of range. Thus, the validity score is 96.1%.

3. Timeliness: timeliness score is calculated based on the number of correct timelines. The data were collected between 1997 and 2017, and the average of FIRE_YEAR is 2004, which means that our data is 16 years old on average. If we score using the criterion that n-year-old data can get a score of (100 – n) %, then the timeliness score is 84%.

4. Consistency: We checked if the data of a feature follow the same formats. For example, the objective variable has different coding in different periods. We can calculate those consistent formats as consistency scores. We found that variables are highly consistent except OBJECTIVE. Thus, we obtained a consistency score of 98.9%.

5. Uniqueness: It is calculated by the number of non-duplicates we have on the data. For the columns, we found that all the values in PREV_DIST and CUR_DIST are the same, and the columns about the UTM system provide the same geographical information as longitude and latitude. Thus, we think there are four duplicated columns, while there are no duplicated rows in our data. Therefore, the score for the uniqueness dimension is 89/93 = 96%.

6. Conformity: Conformity means the data is following the set of standard data definitions like data type, size and format. We didn't find any variables that were not measured using the standard.

7. Accuracy: Accuracy is the degree to which data correctly reflects the real-world object, and since we didn't find any unexpected outliers in our data. Therefore, we can say the data was accurate.

Since completeness and validity has more impact in our dataset and the influence is irreversible, thus we decided to assign more weight to these two metrics. We assigned a 10% weight to timeliness since timeliness is related to accuracy and decays with time, a 5% to consistency and accuracy, and a 2.5% to both uniqueness and conformity. The reason behind this was because we already knew that inconsistency occurred in only one column, and uniqueness and conformity was not violated. We performed metric measurement for each column and got the average score. The overall score(dq) is the summation of the weighted score for each dimension as outlined below:

$$\begin{aligned}
dq &= 40\% \times (completeness) + 30\% \times (validity) + 10\% \times (timeliness) + 5\% \times (consistency) + \\
&2.5\% \times (uniqueness) + 2.5\% \times (conformity) + 5\% \times (accuracy) \\
&= 40\% \times 68.5 + 30\% \times 96.1 + 10\% \times 84 + 5\% \times 98.9 + 2.5\% \times 96 + 2.5\% \times 100 + 5\% \times 100 \\
&= 79.47
\end{aligned}$$

The final data quality score was 79.47 out of 100, which means our data is at least of good quality, if not excellent. Therefore, meaningful results and conclusions can be drawn from it if we analyze it properly.

2.2 Data cleaning

2.2.1 Handling missing values

A major portion of the data preprocessing process is the handling of missing values, and the most basic approach deployed is to remove variables with more than 50% missing values as we assume that these variables will hardly have any effect on our predictions. The second approach deployed to account for missing values, especially for factor variables, was to replace them with certain default values already defined in the data dictionary. Besides, we replace the missing values of a numerical variable with either median or mean of that variable.

2.2.2 Handling Inconsistent Observations

Typically, the timeline of a fire is Start→Discover→Report→Attack→Beheld→Undercontrol→Out. The inconsistent observations were taken care of by creating new lifetime variables, and the observations that did not follow the general timeline for a fire were removed, finally leaving us with 2998 observations of 53 variables. We tried to explore the relationship between lifetime variables and the OBJECTIVE variable. We find that the fire response level affects the lifetime UCO_TO_OUT most as it is reasonable since it's not wise to monitor the fire before it is under control. Therefore, fires to be monitored turned out to have the longest lifetime.

2.3 Exploratory data analysis

In this section, the goal is to peer into the heart of the process by using the given data as the window for the initial exploratory data analysis. We first visualized the number of fires recorded from 1995 to 2017, as shown in Figure 1. We can see that the number of fires differs quite a lot in different years, and the count is meagre from 2013 to 2017.

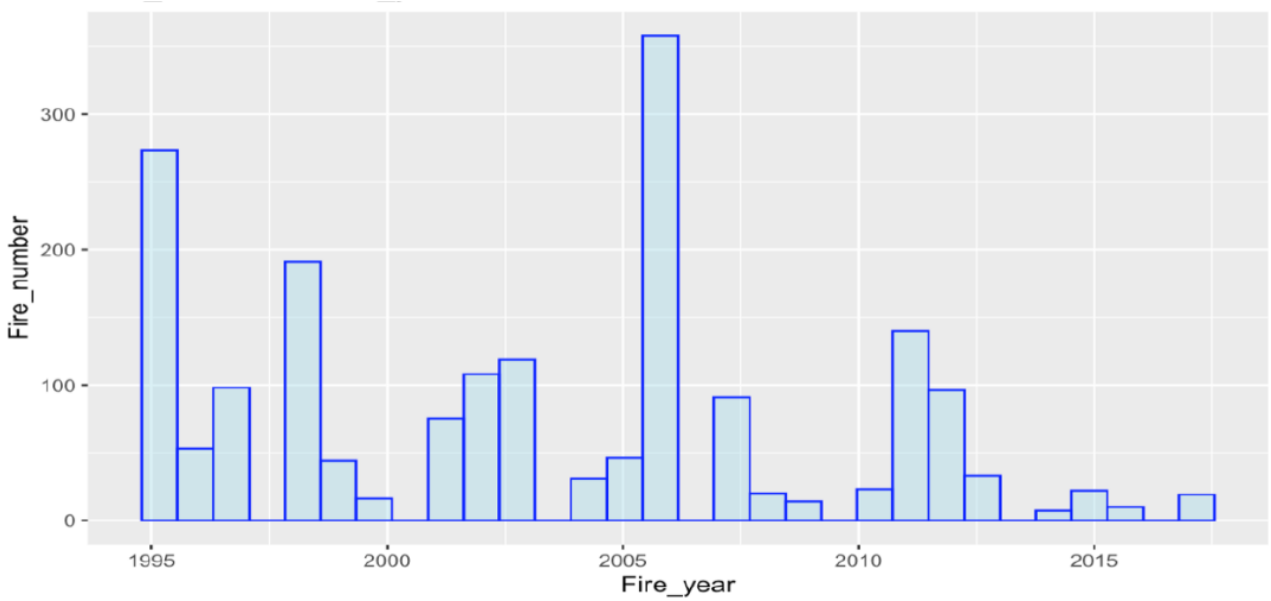

Figure 1: Number of fires from 1995-2017

In this project, we set out to analyze different types of fires by clustering them and use these clusters to identify patterns from data to dispatch resources for fire management efficiently. Thus, the objective of handling a fire is very crucial in our analysis. Based on the information provided by client about **OBJECTIVE** variable, we transformed the **OBJECTIVE** data into the corresponding three levels, *FSUP*, *MON* and *PSUP*. Then we counted the fire cases under each **OBJECTIVE** level. As seen in Figure 2, 99% of the fires are classified in the class *FSUP*.

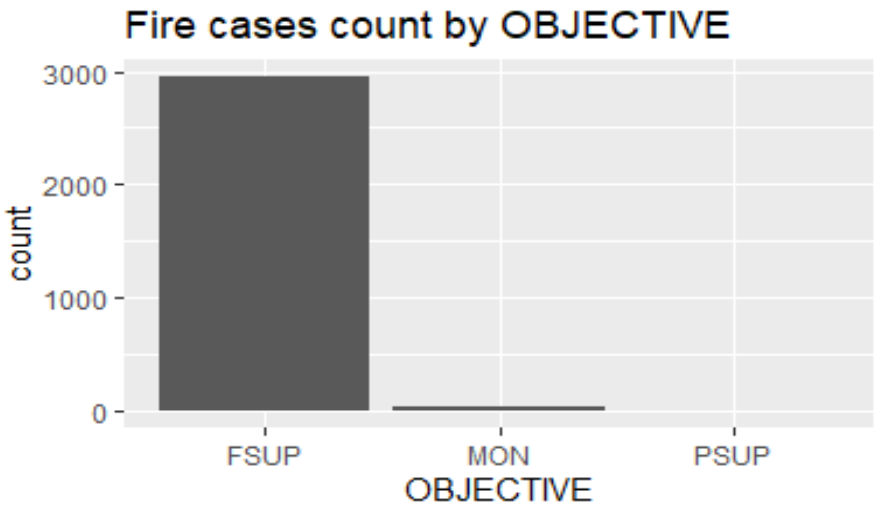

Figure 2: Count of fires with OBJECTIVE

Next, we looked into the causes of fires. Overall, the wildland fires can be classified into two categories according to the causes of the fire. The lightning-caused fires account for approximately 57%

of all fires. We found that A&G (Air and Ground attack) and GPP (Ground Attack, Power Pumps) are the two most used methods to deal with fires, as shown in Figure 3.

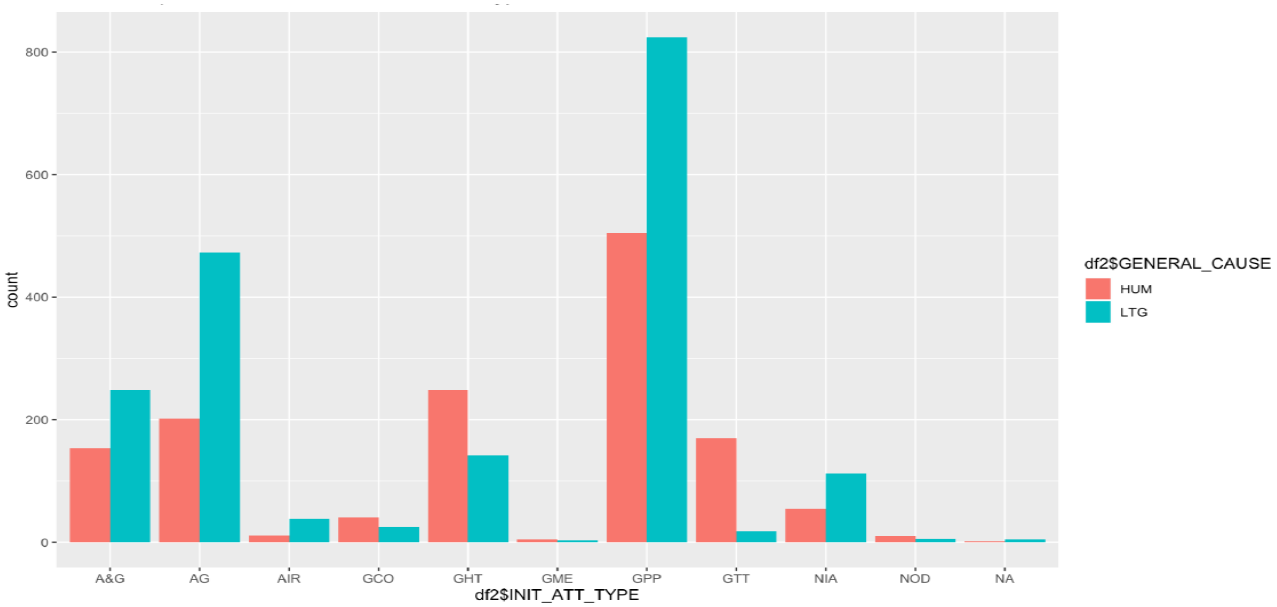

Figure 3: Initial Attack Type for both the categories of fire

The following two tables show the proportion of different types of aircraft and the number of air tankers used, respectively, for both the lightning-caused and human-caused fires.

Cause of fire	COM	FW	HEL	NAC	NOD
Lightning	37.7%	4.49%	45.94%	11.50%	0.37%
Human	20.74%	5.99%	28.75%	43.84%	0.64%

Table 1: Percentage of aircraft used for the major two categories of fire

Cause of fire	0	1	2	3	4	5
Lightning	60.08%	28.93%	9.53%	0.92%	0.33%	0.22%
Human	74.35%	18.42%	6.07%	0.82%	0.27%	0.07%

Table 2: Percentage of airtankers used for the major two categories of fire

Table 1 shows that the usage of a helicopter (HEL) and a combination of fixed-wing and helicopter (COM) for fires caused by lightning is 17% higher than human-caused fires. Table 2 also indicates that the usage of air tankers for lightning fires is about 14% higher than human-caused fires. Compared to human-caused fires, lightning-caused fires are more likely to happen in remote places. Due to the remoteness, the aircraft is necessary for transporting crews and resources to suppress

lightning-caused fires.

One of the primary components of the FBP system is the rate of spread that depends on the initial spread index (ISI), a combined effect of Fine Fuel Moisture Code (FFMC) and wind speed, along with fuel type and topographic slope [Bilgili, 2003]. Figure 4 shows that FWI, a threshold value which is a combination of ISI and BUI, and the number of fires is positively correlated as we have already seen that 2006 was the year that has the largest number of lighting fire occurrence. It was also the year with the highest FWI.

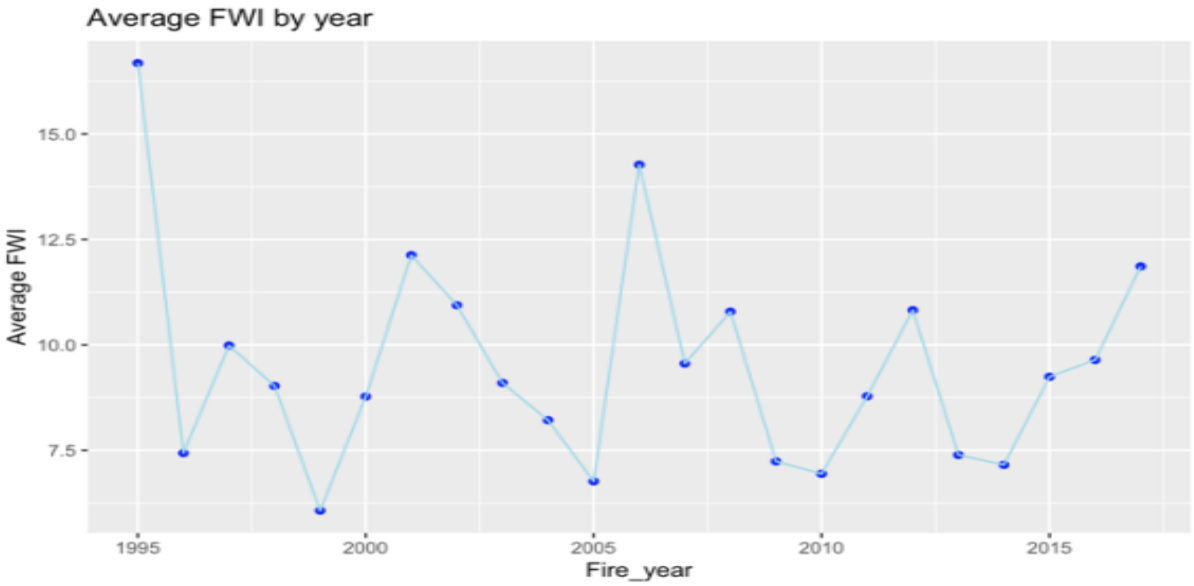

Figure 4: Average FWI by year

Because of the diversity of forest and non-forest ecosystems, describing fuels is a complex task. Currently, the FBP system describes fuel type in qualitative terms rather than quantitative [Stocks et al., 1989]. The number of fuel types currently recognized in the FBP system reflects the amount of empirical fire behaviour data available in Canada. The exploratory analysis showed 42 different levels of fuel types, out of which the C5-Red and White pine and C2-Boreal spruce type of fuel were present in maximum proportions across the Fort Francis and Kenora district respectively, verifying the fact that these vegetations are present in the majority in these regions respectively [Canada, 2019]. The number of times a type of fuel is found burning at the time of the initial attack for the fire is shown in Figure 5. We can conclude that over time the fuel type present in a geographical area varied.

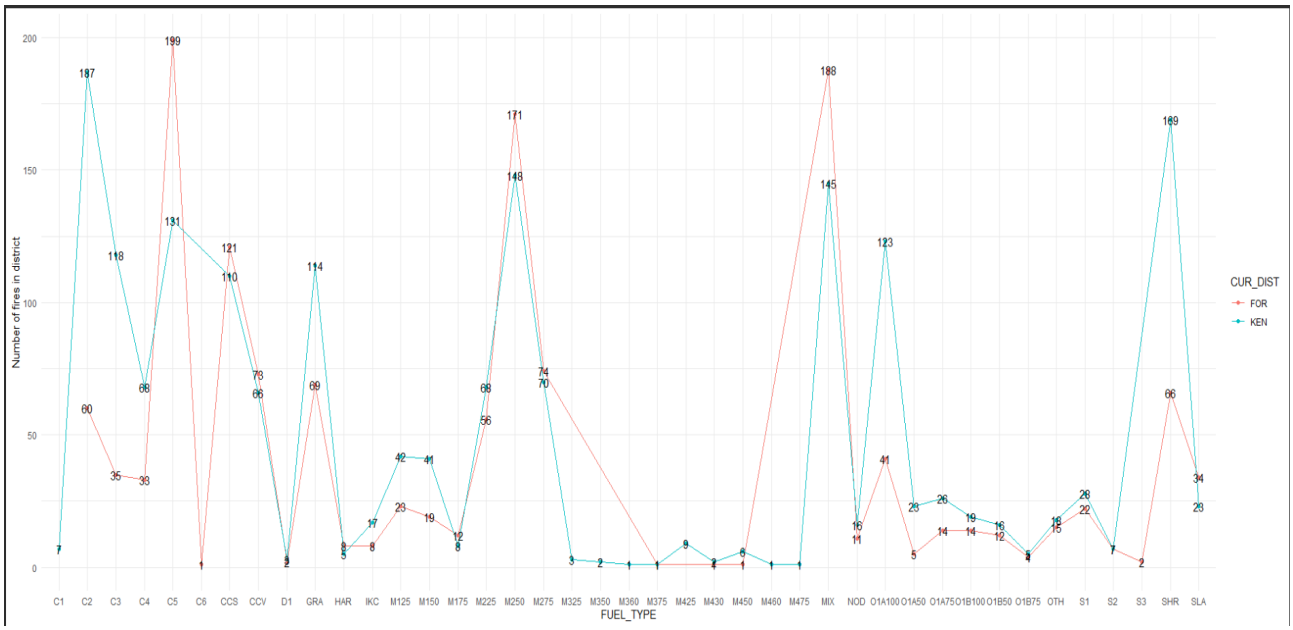

Figure 5: Number of fires that encountered combustion of particular fuel type (for both the districts)

Finally, it was essential to analyze the rate of spread of fires due to the burning of a particular fuel type in an area as it is a good indicator for the calculation of the final size of fires. Also, the attack type and location of attack are highly dependent on the size of a fire. We found that the average lifetime of a fire in the study area is about three days. It takes the longest time for a fire to be discovered, and the second-longest interval is from *Under control* to *Out*. Even though it takes more than one day for a fire to be discovered averagely, most of the fires can be discovered the day they start. We also found that human-caused fires have a bit larger spread than lightning-caused fires, as shown in Figure 6.

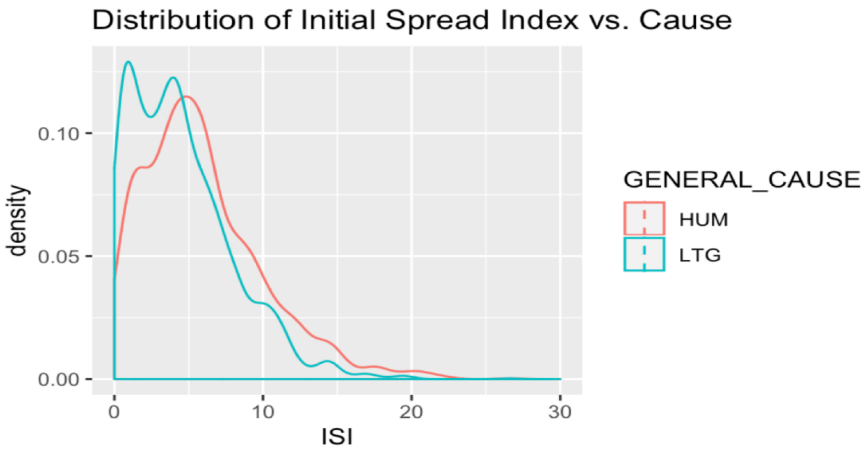

Figure 6: Initial Spread Index with respect to fire cause

3 METHODS

3.1 Clustering method

In order to find sub-populations, we decide to use cluster methods. Because this data set contains both categorical and continuous variables, we selected Hierarchical Clustering (HC) [Rokach and Maimon, 2005], Partitioning Around Medoid (PAM) [Kaufmann and Rousseeuw, 1987] and K-prototypes [Huang, 1998] to make labels for the data. HC treats every data as a class initially, then merge the classes that are close together. Finally, all data will be classified into one class, but we can determine the optimal number of classes by the distance between the classes. PAM is a good method to handle categorical variables. It randomly chooses some data as the center of the classes then updates the centers based on the distance. This process will repeat many times until the center of each class no longer changes. K-prototypes works very similar to PAM. The only difference is that PAM uses the same method to calculate the distance from the data to the center of each class, whether the data is categorical or continuous; however, K-prototypes uses two different methods to calculate the distance. In the clustering problem, we use K to represent the number of clusters applied to the data set, and all three methods give their optimal K value.

3.2 Multi Classification method

After assigning the class to the observations based on the result of clustering, we fitted the data using multinomial regression [Engel, 1988] with the LASSO penalty [Tibshirani, 1996], multinomial regression with BIC (Bayesian Information Criterion) [Schwarz, 1978]. and random forest [Breiman, 2001] to check the variable importance.

Multinomial regression and random forest are two commonly used models for multi-class classification. Multinomial regression is a statistical model, which predicts the probability of an observation belonging to a class based on multiple predictive variables [Kida, 2019]. Random forest is a machine learning model derived from the idea of the decision tree. A decision tree is a diagram that illustrates all possible paths of a decision [Kotu and Deshpande, 2019]. Unlike multiple regression, the random forest does not use a specified function to do the classification. It makes the decision of classification based on its own set of rules.

4 RESULTS

4.1 Distinct populations when dispatching resources

4.1.1 Number of clusters

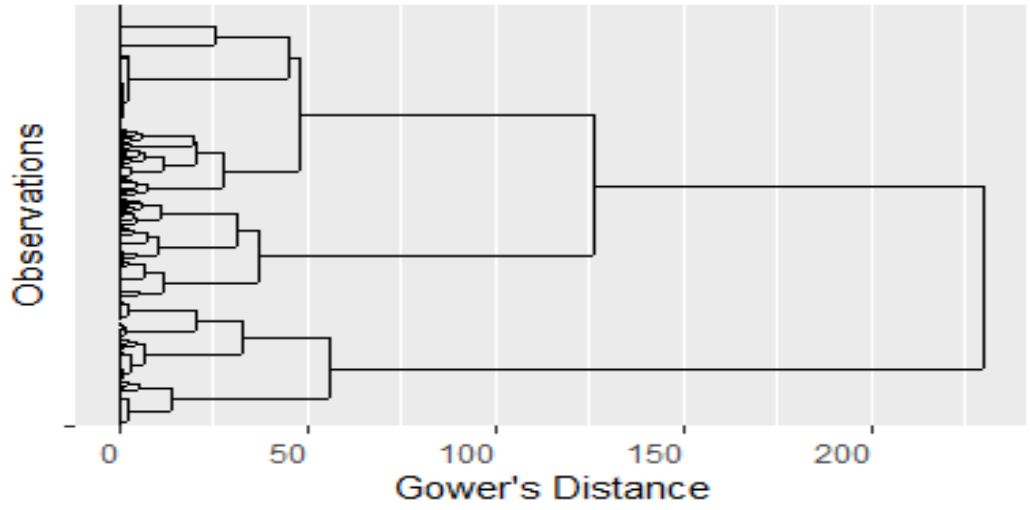

Figure 7: Hierarchical Clustering result

Figure 7 is the clustering result of the Hierarchical Clustering method. Based on this figure, we can see that the data is classified into three classes, and the distance between each class is very far away, which indicates these three classes have a huge difference. Therefore, the optimal K value for the HC method is 3.

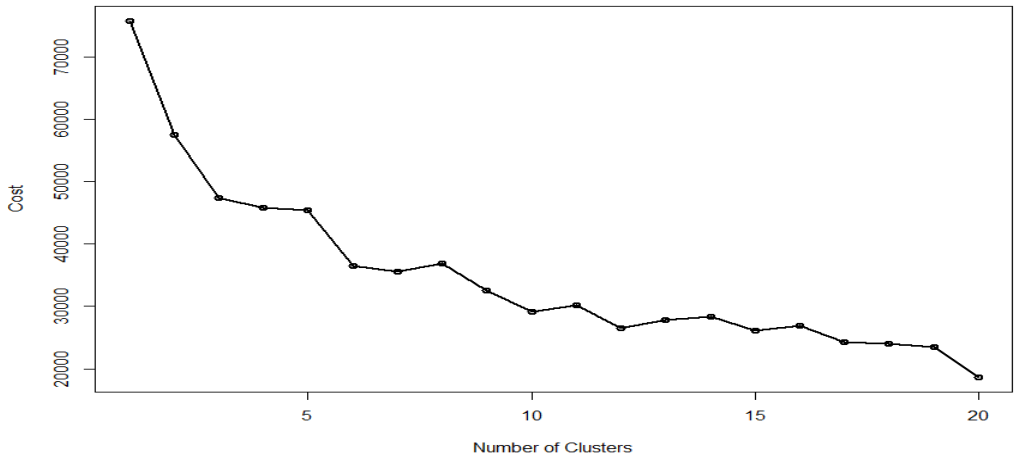

Figure 8: Cost VS Number of Clusters (kproto)

Figure 8 is the result of the Elbow method for the K-prototypes. Elbow method is a popular method to determine the optimal K. Cost is the total distance from data to the center of all classes. When the decrease rate of cost at a K value becomes significantly smaller, that K is the best K value. For example, in Figure 8, when the number of clusters (K) equals to 3, the slope becomes smaller. Thus, 3 is the optimal K for this approach.

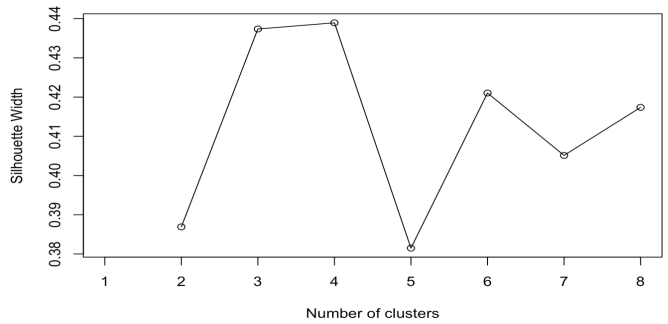

Figure 9: Silhouette Width VS Number of Clusters (PAM)

Figure 9 gives the silhouette width of each K for the PAM method. Silhouette width is a measurement to evaluate the performance of different K. When the silhouette width of the K reaches the largest point, that K is the optimal K value. In figure 9, we can see silhouette width of K = 3 is close to silhouette width of K = 4. Thus, for this data set, there is not much difference whether we classify the data in three classes or four categories. Generally, we should pick four as the best K. However, the best K given by the other two methods is 3. To compare the three approaches, we also selected three as the optimal K for PAM.

4.1.2 Clustering result

As mentioned above, we used three different clustering methods to get distinct populations for dispatching resources. After examining the results, we compared them pair wisely to see how much proportion of them are identical to each other. The result we got is shown in the Table 3.

Pairs	HC VS kproto	HC VS PAM	PAM VS kproto
Identical proportion	67.3%	90.7 %	66.4%

Table 3: Clustering results comparison

From the table, we can see that HC and PAM had very similar clustering results. However, Hierar-

chical Clustering had the highest overall identical proportion compared to the other two methods. Thus, we decided to use the result of Hierarchical Clustering for our further analysis.

4.1.3 Cluster features

By using the Hierarchical Clustering, we divided the data into three different classes. Then, the variables which were considered in clustering against **class** to look into the features of each class were plotted as shown in Figure 10.

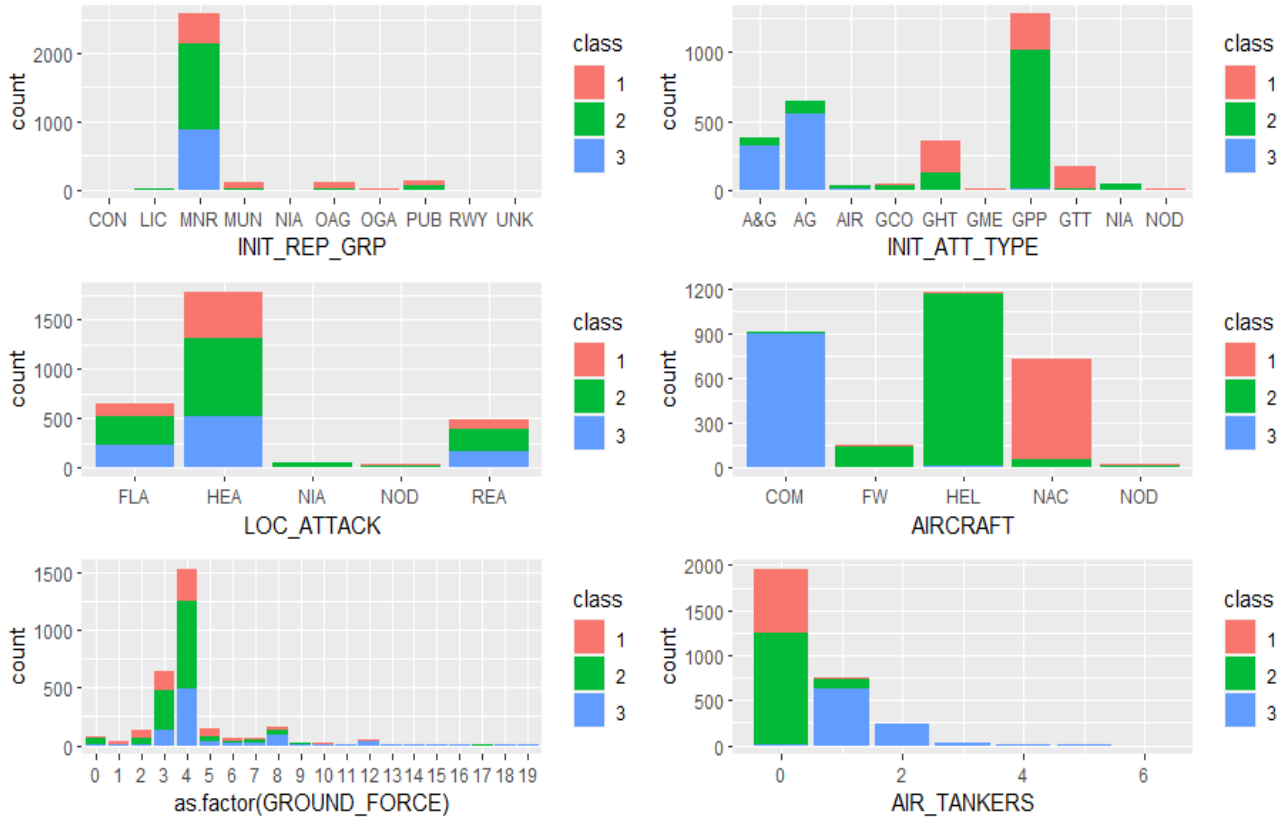

Figure 10: Cluster features

Because the **OBJECTIVE** values are *FSUP* for around 99 % of the observations in our data, we did not compare **OBJECTIVE** for different classes here. From the figure above, we found that the main differences among these three classes lied in the three variables **INIT_ATT_TYPE**, **AIRCRAFT** and **AIR_TANKERS**, as shown in Table 4.

Cluster	INIT_ATT_TYPE	AIRCRAFT	AIR_TANKERS
Type 1	GHT, GTT	NAC	0
Type 2	AIR, GCO, GPP, NIA	FW, HEL	≤ 1
Type 3	A&G, AG	COM	≥ 1

Table 4: Cluster difference

From the table, we can see that Type 1 action is the weak ground-based action. When using the Type 1 action, usually hand tools (GHT) or tank trucks(GTT) are used to attack fires from the ground. In most cases, no air tankers are used for Type 1 action.

For Type 2 action, we can say it is a strong ground-based action because it utilises more powerful tools like Mechanical Equipment and Power Pumps to attack fires. Though in most cases no air tankers are used, sometimes fires might spread so fast that at most one air tanker would be used to control the fires.

The Type 3 action is an air-ground-combined action. It usually requires more than one air tanker, and both fixed-wing and helicopter are used for putting out fires. However, ground force is also necessary in Type 3 action since in most cases, the fire needs to be attacked from air and ground at the same time.

To sum up, there are three different types of actions for dispatching resources to fires, which are weak ground-based action, strong ground-based action and air-ground-combined action. We assumed different types of action might correspond to different sizes of fires. The weak ground-based action is usually used to handle small fires, while the strong ground-based action is often used for big fires, and the air-ground-combined action is used for extremely big fires.

4.2 Factors influencing action types

In order to verify our assumption and to find out the important factors influencing action types further analysis was carried out by fitting multi classification models using action type as the response. After removing 21 variables based on the domain knowledge and preliminary analysis of variables, the models were fit considering only 32 variables.

In multiple regression, we applied two different methods to conduct variable selection. For LASSO regression, we had 13 variables in our final model, while for Stepwise selection using BIC, we had 18 variables in our final model. For the random forest model, we obtained a rank of variables based on the impact of each variable in making classification decisions.

Figure 11, shows the ten most important variables obtained by these three methods, respectively.

Ranking	LASSO	BIC	Random Forest
1	Fuel_Type	Fuel_Type	Attack_To_BHE
2	Resp_Group	Resp_Group	Fuel_Type
3	General_Cause	Fire_MGT_Zone	Gateaway_To_Attack
4	General_IGN	BHE_To_UCO	BHE_To_UCO
5	BHE_To_UCO	Attack_To_BHE	Frep_To_Getaway
6	Rate_of_Spread	Cur_Distken	Resp_Group
7	Ownership	Group_Known_ESTK	UCO_To_Out
8	Fire_MGT_Zone	Frep_To_Getaway	General_IGN
9	FWI	UCO_To_Out	Northing
10	BUI	ISI	BUI

Figure 11: Variable importance ranking

It can be observed from Figure 11 that the top 10 most important variables for the three methods are different. But all three methods have variables **FUEL_TYPE**, **RESP_GROUP** and **BHE_TO_UCO** in common. It is worth noticing that the variable **FUEL_TYPE** received a very high rank among all the used methods, which means it contributes the most when classifying action types.

Fuels include everything in the forest like woods, grasses, and brush. The type of fuel can influence the behaviour and intensity of forest fires. Some types of fuel such as grasses are combustible, and can cause the fire to spread fast and burn intensively. Fuels like hardwood brush have a low moisture content, resulting in fires with slow spread and of low intensity. The allocation of resources for forest fire control is limited and is typically determined by the intensity and severity of the fire. Therefore, the fuel type is an important factor that needs to be considered while allocating the resources.

The variable **RESP_GROUP** also played a significant role in the classification with second-highest-ranking in both methods of multinomial regression. There are roughly 40 response groups that are responsible for the forest fire in Ontario. Each forest fire receives a response from an appropriate response group based on the location and condition of the fire.

The most important variable for the random forest classifier was **ATTACK_TO_BHE**, which is different from the variables selected by LASSO and BIC. It means that different types of action differ a lot in terms of efficiency. It usually takes more time to hold fires by using the air-ground-combined action, while it requires the shortest time to hold fires by using the weak ground-based action.

It is also worth mentioning that the fire size did not show up in the list of important variables. So, the type of action is not dependent on the size of a fire. Instead, the fire management department decides the action to use by fuel type and the group responsible for the fire.

5 DISCUSSION

5.1 Main conclusions

Since the aim of the project was to detect the features of sub-populations when dispatching resources, we first used Hierarchical Clustering, Partitioning Around Medoid and K-prototypes to separate the data into three classes, then used multinomial regression model and random forest to analyze the importance of variables. For multi classification methods, we also applied some variable selection methods like Lasso and BIC to remove some less important variables, so the model becomes more general. The main difference between the three classes was due to the initial attack type. Through exploring the relation between initial attack type and clustering classes, we found type 1, type 2 and type 3 are weak ground-based action, strong ground-based action and air-ground-combined action, respectively. Moreover, we found that **FUEL_TYPE**, **RESP_GROUP** and **BHE_TO_UCO** are there core factors affecting action types.

5.2 Limitation

However, there are still some limitations of our analysis. First of all, in the data cleaning part, we have removed some observations and some variables because either they had missing values or were invalid. So some bias might arise due to the removal of such data.

In terms of the methods used, we did clustering based on Gower's distance. We used Gower's distance [Gower, 1971] to measure the dissimilarity between two observations, but in reality, the two observations can be dissimilar with each other not necessarily the same as their Gower's distance. Thus, bias might arise in clustering if the dissimilarity between two observations is different from the Gower's distance between them.

In addition, we didn't have a specific guideline for comparing different clustering results. In this analysis, we compared the clustering results pair-wise and then chose the one that was most similar to the other two. Even though we might not have selected the best clustering result here, we chose the one with the lowest risk.

5.3 Recommendation

According to the analysis result, we found that **FUEL_TYPE** is the most important variable that affects the action type. More specifically, when **FUEL_TYPE** is *O1A100* or *O1A175*, the action type tends to be a weak ground-based action. Besides, when **FUEL_TYPE** is *C2*, the action type tends to be a air-ground-combined action. Thus, in order to dispatch resources more efficiently, we can arrange more air tankers for the area where **FUEL_TYPE** is *C2* and decrease the air tankers around the area where **FUEL_TYPE** is *O1A100* or *O1A175*.

In terms of **RESP_GROUP**, we found that the action type was possibly an air-ground-combined action when **RESP_GROUP** is *lightning*. So we can also arrange more air tankers around the area where lightning usually happens.

References

[admin, 2018] admin (2018). Pros and cons of forest fires.

[Bilgili, 2003] Bilgili, E. (2003). Fire behavior prediction in canadian slash fuels, based on fuel characteristics.

[Breiman, 2001] Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

[Canada, 2019] Canada, N. R. (2019). Fbp fuel type descriptions.

[Engel, 1988] Engel, J. (1988). Polytomous logistic regression. *Statistica Neerlandica*, 42(4):233–252.

[Gower, 1971] Gower, J. C. (1971). A general coefficient of similarity and some of its properties. *Biometrics*, pages 857–871.

[Huang, 1998] Huang, Z. (1998). Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Mining and Knowledge Discovery*, 2(3):283–304.

[Kaufmann and Rousseeuw, 1987] Kaufmann, L. and Rousseeuw, P. (1987). Clustering by means of medoids. *Data Analysis based on the L1-Norm and Related Methods*, pages 405–416.

[Kida, 2019] Kida, Y. (2019). Generalized linear models.

[Kotu and Deshpande, 2019] Kotu, V. and Deshpande, B. (2019). *In Data Science*. Morgan Kaufmann Publishers.

[McFayden, 2020] McFayden, C. (2020). Fire management introduction for western university, statistical data consulting course.

[Rokach and Maimon, 2005] Rokach, L. and Maimon, O. (2005). *Data Mining and Knowledge Discovery Handbook*. Springer US.

[Schwarz, 1978] Schwarz, G. E. (1978). Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464.

[Stocks et al., 1989] Stocks, B. J., Lynham, T. J., Lawson, B. D., Alexander, M. E., Wagner, C. E. V., McAlpine, R. S., and Dubé, D. E. (1989). Canadian forest fire danger rating system: An overview. *The Forestry Chronicle*, 65(4):258–265.

[Tibshirani, 1996] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–88.

APPENDIX

Appendix A

```
#-----------------------------------------------------------------
# HC
# Read the data
NWR_fire = read.csv('Team2_NWR_W1_W2_FOR_KEN_CLEANDATA.csv')

# Extract the variables to consider
cluster_variable = c('OBJECTIVE', 'INIT_REP_GRP', 'INIT_ATT_TYPE',
                     'LOC_ATTACK', 'GROUND_FORCE', 'AIR_TANKERS',
                     'AIRCRAFT')
NWR_fire_cvar = NWR_fire[, which(names(NWR_fire) %in% cluster_variable)]
#-----------------------------------------------------------------
# Do HC using the cluster package
## Compute the Gower's distance for mixed data
library(StatMatch)
gdist = as.dist(gower.dist(NWR_fire_cvar))
gdist[is.na(gdist)] = max(gdist, na.rm = T)

library(cluster)
## methods to assess
m <- c( "average", "single", "complete", "ward")
names(m) <- c( "average", "single", "complete", "ward")

## function to compute coefficient
ac <- function(x) {
  agnes(gdist, diss = T, method = x)$ac
}

## Find the best linkage method
ac_methods = map_dbl(m, ac)
#-----------------------------------------------------------------
# Do HC using hclust function
## Hierarchical Clustering
NWR_cluster = hclust(gdist, method = 'ward.D')
```

```
## Plot the dendrogram
library(ggplot2)
library(ggdendro)
ggdendrogram(NWR_cluster, rotate = TRUE, theme_dendro = FALSE) +
  xlim('') + xlab('Observations') + ylab("Gower's Distance")

#-----------------------------------------------------------------
## PAM
data= read.csv("Team2_NWR_W1_W2_FOR_KEN_CLEANDATA.csv")
data2= data[, c("OBJECTIVE", "INIT_REP_GRP", "INIT_ATT_TYPE",
                "LOC_ATTACK","GROUND_FORCE","AIR_TANKERS","AIRCRAFT")]
summary(data2)
library(cluster)
library(dplyr)
library(ggplot2)
library(readr)
install.packages("Rtsne")
library(Rtsne)

# Compute Gower distance
gower_dist=daisy(data2, metric = "gower")
gower_mat = as.matrix(gower_dist)

# search for # of clustering
sil_width <- c(NA)
for(i in 2:8){
  pam_fit <- pam(gower_dist, diss = TRUE, k = i)
  sil_width[i] <- pam_fit$silinfo$avg.width
}

# plot Number of clusters
plot(1:8, sil_width,
     xlab = "Number of clusters",
     ylab = "Silhouette Width")
lines(1:8, sil_width)
# 3 or 4 are good
```

```
# clustering
set.seed(100)
k =3
pam_fit = pam(gower_dist, diss = TRUE, k)
pam_results = data2 %>%
  mutate(cluster = pam_fit$clustering) %>%
  group_by(cluster) %>%
  do(the_summary = summary(.))
pam_results$the_summary
# check clustering performance
set.seed(100)
tsne_obj = Rtsne(gower_dist, is_distance = TRUE)
tsne_data = tsne_obj$Y %>%
  data.frame() %>%
  setNames(c("X", "Y")) %>%
  mutate(cluster = factor(pam_fit$clustering))
ggplot(aes(x = X, y = Y), data = tsne_data) +
  geom_point(aes(color = cluster))

#-----------------------------------------------------------------
# kproto
# Read the data
library(clustMixType)
data_cleaned = read.csv("Team2_NWR_W1_W2_FOR_KEN_CLEANDATA.csv")
X = subset(data_cleaned, select = c("OBJECTIVE", "INIT_REP_GRP",
                                    "INIT_ATT_TYPE", "LOC_ATTACK",
                                    "GROUND_FORCE", "AIR_TANKERS", "AIRCRAFT"))
cost = rep(0, 20)
for (i in 1:20) {
  kpres = kproto(X, i)
  cost[i] = kpres$tot.withinss
}
plot(cost, type = "o", lwd = 2, xlab = "Number of Clusters", ylab = "Cost")
kpres = kproto(X, 3)
cluster = predict(kpres, X)$cluster
```

clustering.R

Appendix B

```
#-------------------------------------------------------------------------
# Multiclassification
# Read the data
NWR_fire = read.csv('NWR_fire.csv')[, -1]
summary(NWR_fire)

mean(NWR_fire[NWR_fire$class == 3, ]$ATTACK_TO_BHE)

var_to_remove = c('FIRE_TYPE' , 'REGION', 'FIRE_ZONE', # Only one level
                  'FIRE_NUMBER', # ID of fire
                  'LONGITUDE', 'LATITUDE', # Provide the same information as EASTING
    and NORTHING
                  'START_DATE', 'DISC_DATE', 'F_REP_DATE',
                  'GETAWAY_DATE', 'ATTACK_DATE', 'BHE_DATE', 'UCO_DATE',
                  'OUT_DATE', # Date variable
                  'OBJECTIVE', 'INIT_REP_GRP', 'INIT_ATT_TYPE',
                  'LOC_ATTACK', 'GROUND_FORCE', 'AIR_TANKERS',
                  'AIRCRAFT', # Variables considered in clustering
                  'WX_STATION'# Too many levels and meaningless
)
NWR_fire1 = NWR_fire[, -which(names(NWR_fire) %in% var_to_remove)]
summary(NWR_fire1)
levels(NWR_fire1$GENERAL_CAUSE)
#-------------------------------------------------------------------------
# Multiclassification
## Lasso regression
lasso_mod = glmnet(X, y, family = 'multinomial',
                   type.multinomial = 'grouped')

lasso.beta = lasso_mod$beta
obj = -deviance(lasso_mod)
k = lasso_mod$df
n = lasso_mod$nobs

BIC_lasso = log(n)*k - obj
```

```
lambda.lasso = which.min(BIC_lasso)

lasso.beta1 = lasso.beta[[1]][ ,lambda.lasso]
lasso.beta2 = lasso.beta[[2]][ ,lambda.lasso]
lasso.beta3 = lasso.beta[[3]][ ,lambda.lasso]

mean(names(lasso.beta1[lasso.beta1 != 0]) == names(lasso.beta2[lasso.beta2 != 0]))
mean(names(lasso.beta1[lasso.beta1 != 0]) == names(lasso.beta3[lasso.beta3 != 0]))

lasso.result = data.frame(class1 = lasso.beta1[lasso.beta1 != 0],
                          class2 = lasso.beta2[lasso.beta2 != 0],
                          class3 = lasso.beta3[lasso.beta3 != 0])

library(xlsx)
write.xlsx(lasso.result, 'lasso_result.xlsx')

# Elastic net
elastic_mod = glmnet(X, y, family = 'multinomial',
                     type.multinomial = 'grouped', alpha = 0.5)

elastic.beta = elastic_mod$beta
obj = -deviance(elastic_mod)
k = elastic_mod$df
n = elastic_mod$nobs

BIC_elastic = log(n)*k - obj

lambda.elastic = which.min(BIC_elastic)

elastic.beta1 = elastic.beta[[1]][ ,lambda.elastic]
elastic.beta2 = elastic.beta[[2]][ ,lambda.elastic]
elastic.beta3 = elastic.beta[[3]][ ,lambda.elastic]

mean(names(elastic.beta1[elastic.beta1 != 0]) == names(elastic.beta2[elastic.beta2
    != 0]))
mean(names(elastic.beta1[elastic.beta1 != 0]) == names(elastic.beta3[elastic.beta3
    != 0]))
```

```
elastic.result = data.frame(class1 = elastic.beta1[elastic.beta1 != 0],
                            class2 = elastic.beta2[elastic.beta2 != 0],
                            class3 = elastic.beta3[elastic.beta3 != 0])

#-------------------------------------------------------------------
# Multiclassification
# BIC
# Read the data
cluster_data = read.csv("NWR_fire.csv")
cluster_data = subset(cluster_data, select = -1)

newData = subset(cluster_data, select = -c(FIRE_TYPE, REGION, FIRE_ZONE, FIRE_NUMBER
    , LONGITUDE,
                                           LATITUDE, START_DATE, DISC_DATE,
    F_REP_DATE,
                                           GETAWAY_DATE, ATTACK_DATE, BHE_DATE,
    UCO_DATE, OUT_DATE,
                                           OBJECTIVE, INIT_REP_GRP, INIT_ATT_TYPE,
    LOC_ATTACK,
                                           GROUND_FORCE, AIR_TANKERS, AIRCRAFT,
    WX_STATION))
newData$class = as.factor(newData$class)
n = dim(newData)[1]
p = dim(newData)[2]
library(nnet)
mod = multinom(class ~ ., newData)

step(mod, k = log(n))
mod_BIC = multinom(formula = class ~ CUR_DIST + FIRE_YEAR + EASTING + FIRE_MGT_ZONE
    +
                       GROUP_KNOWN_EST + RESP_GROUP + DC + ISI + BUI + FUEL_TYPE +
                       RATEOFSPREAD + SIZE_INT_ATT + FINAL_SIZE + FREP_TO_GETAWAY +
                       GETAWAY_TO_ATTACK + ATTACK_TO_BHE + BHE_TO_UCO + UCO_TO_OUT,
                     data = newData)
m = summary(mod_BIC)
class2 = m$coefficients[1, ]
class3 = m$coefficients[2, ]
sort(class2, decreasing = TRUE)
```

```
sort(class3, decreasing = TRUE)

#-------------------------------------------------------------------------------
## Randon forest
data=read.csv("NWR_fire.csv")

data2= data[!names(data) %in% c("OBJECTIVE", "INIT_REP_GRP", "INIT_ATT_TYPE","
    LOC_ATTACK",
                                "GROUND_FORCE","AIR_TANKERS","AIRCRAFT","X","REGION"
    ,
                                "FIRE_TYPE","FIRE_ZONE","FIRE_NUMBER","LONGITUDE",
                                "LATITUDE","START_DATE","DISC_DATE","F_REP_DATE","
    GETAWAY_DATE",
                                "ATTACK_DATE","BHE_DATE","UCO_DATE","OUT_DATE","
    WX_STATION")]

set.seed(100)
library(randomForest)
data2$class=as.factor(data2$class)

# random forest
rf = randomForest(class~., data = data2,mtry = 15, importance = TRUE,ntree = 1000)
rf_pred =  predict(rf, fire_ts)
summary(rf)

# check feature importance
importance(rf,type=1)
importance(rf,type=2)

varImpPlot(rf)
```

classification.R