Use case Study Report
Group Number – 10
Student Names – Shalabh Mittal, Nikhilkumar Mirchandani

US Wildfires
- Where smoke meets sky

Background & Introduction

It's been a year of extreme weather. Hurricanes have devastated Texas and the Caribbean, and monsoon floods have displaced millions and killed more than 1,000 people in South Asia. Meanwhile, one of the worst US wildfire seasons in years has ignited blazes across the west. A staggering amount of land has burned so far this season - more than 8 million acres, along with more than 500 homes and other structures.

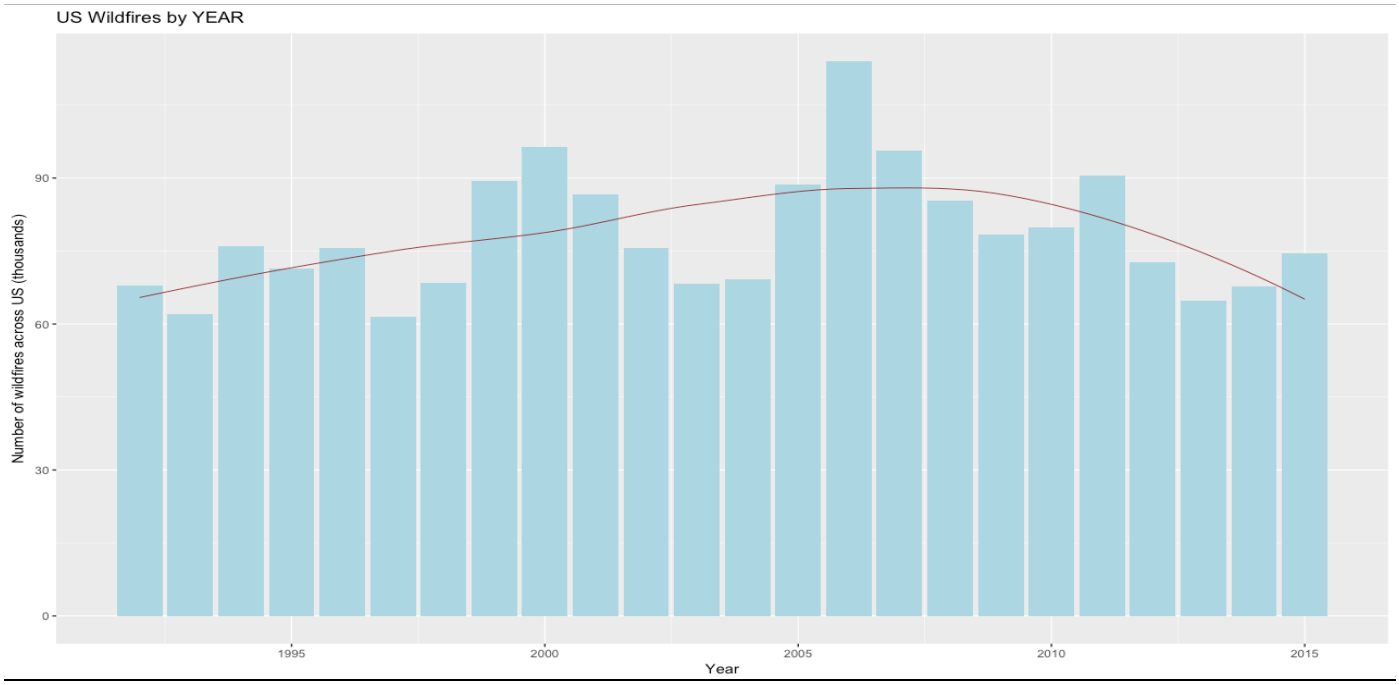
We have obtained the wildfire dataset from Kaggle. The wildfire records were acquired from the reporting systems of federal, state, and local fire organizations. These data were collected using funding from the U.S. Government and can be used without additional permissions or fees. The purpose of this project is to build a model to predict the cause of wildfires in different states to prevent or control such events happening in future. The database has around 2 million observations and 39 variables. Dataset has already been checked for basic errors and redundant records and are removed at the same time. We are visualizing the data on different parameters. We are examining the attributes of fires by cause. Using this, we are trying to find out what causes the most fires and which causes are associated with larger and longer-burning wildfires. We are also examining attributes of fires by size to find out which counties suffered large size fires. Using ggplot2 package, we will visualize number of wildfires by state. Then with the data we have, using regression and predictive analysis, we will use it to build a model to find patterns and predict the future wildfires.

Data Exploration and Visualization

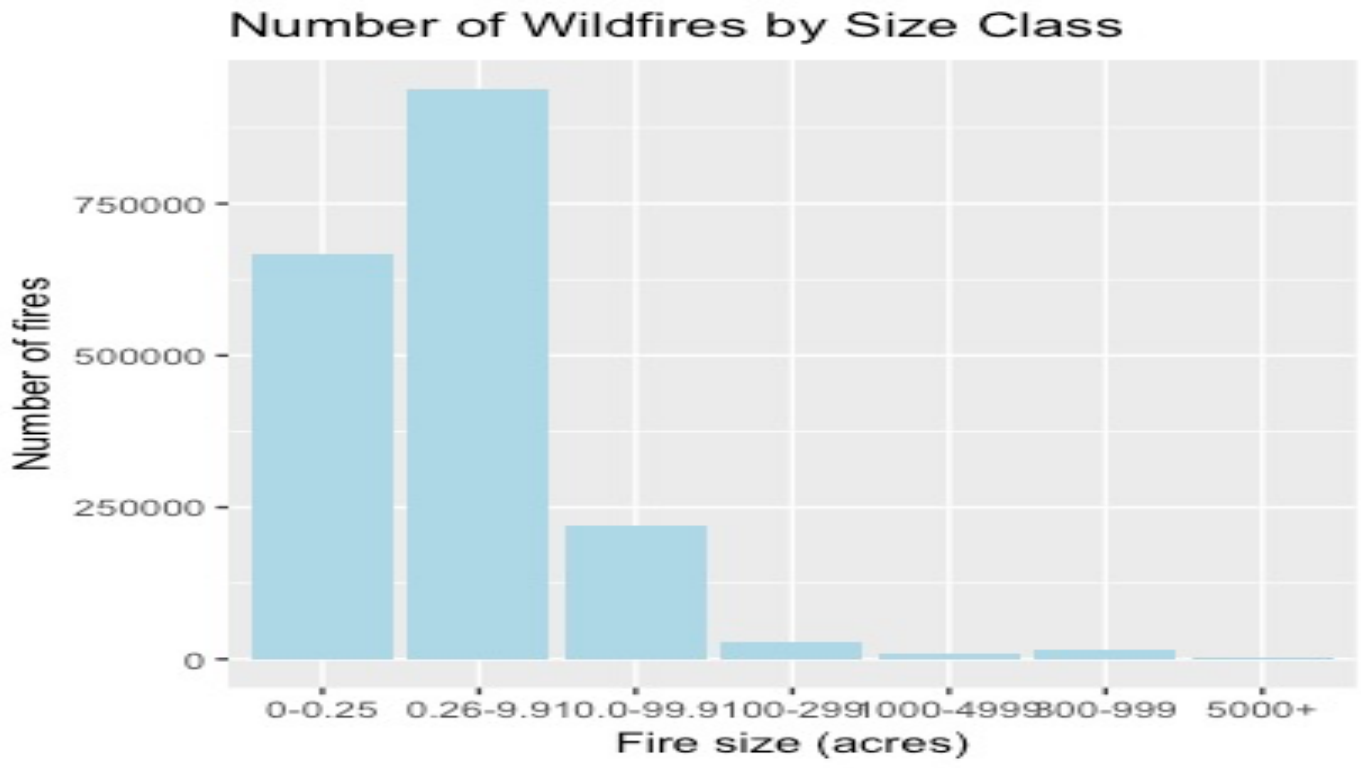
A picture is worth a thousand words.

Data Visualization helps us understand the development of wildfire in a rugged terrain, with a pronounced wind in association with a number of factors that may be responsible for the wildfire. The topography, weather conditions and records of miscellaneous activities in and around a particular area, need to be known for analyzing the extent of wildfire and probability of its future occurrences.

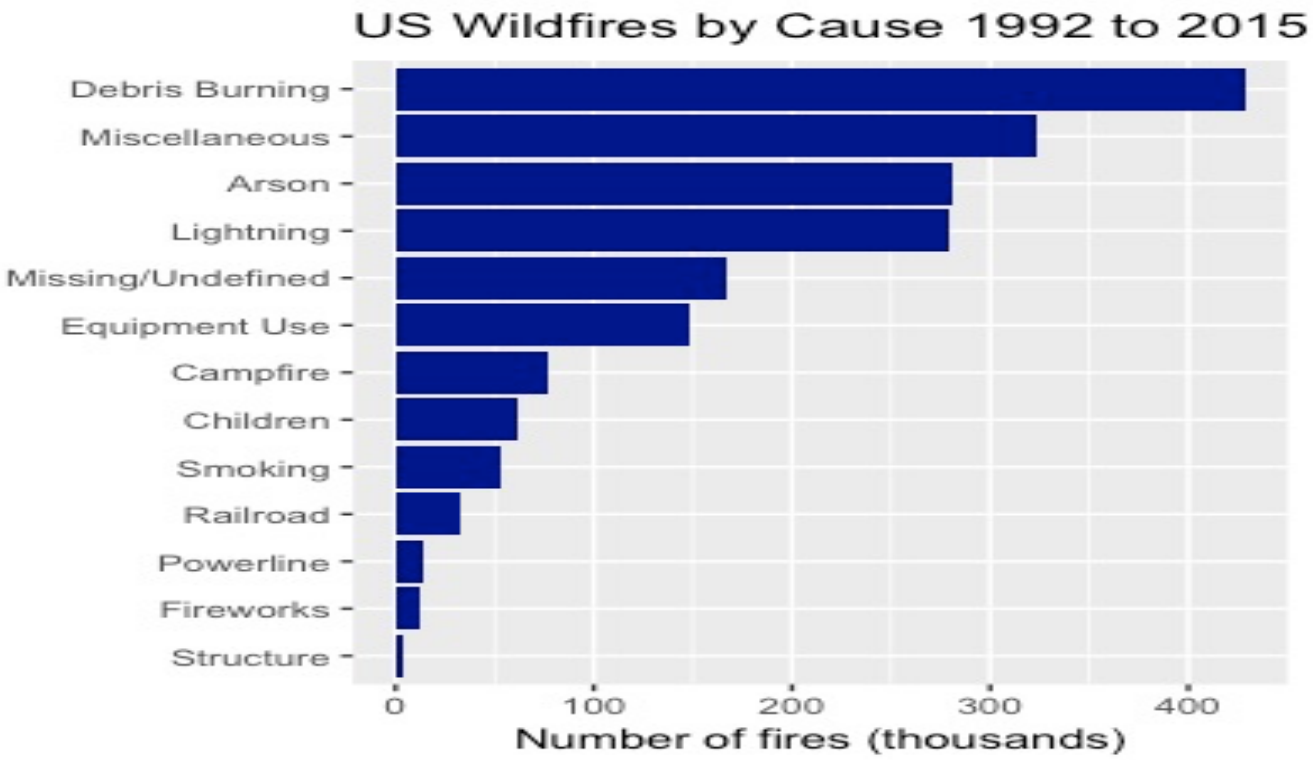
It is possible to analyze and extract plethora of information, considering there are around 2 million observations and 39 variables. The following is a glimpse of a part of the derived information:
A] Causes of wildfire
B] Number of wildfires over time
C] Wildfire size
D] Wildfire geographic distribution throughout the country



From the above graph, it is seen that the number of fires was in the range between 60,000 and 100,000. The number reached the peak of 110,000 only in the year 2006; otherwise it was pretty much in the stated range.



The maximum number of wildfires falls in category B, that is, they range from 0.26 to 9.9 acres.



From the above graph, it is evident that the maximum number of wildfires occurred due to “Debris Burning”.

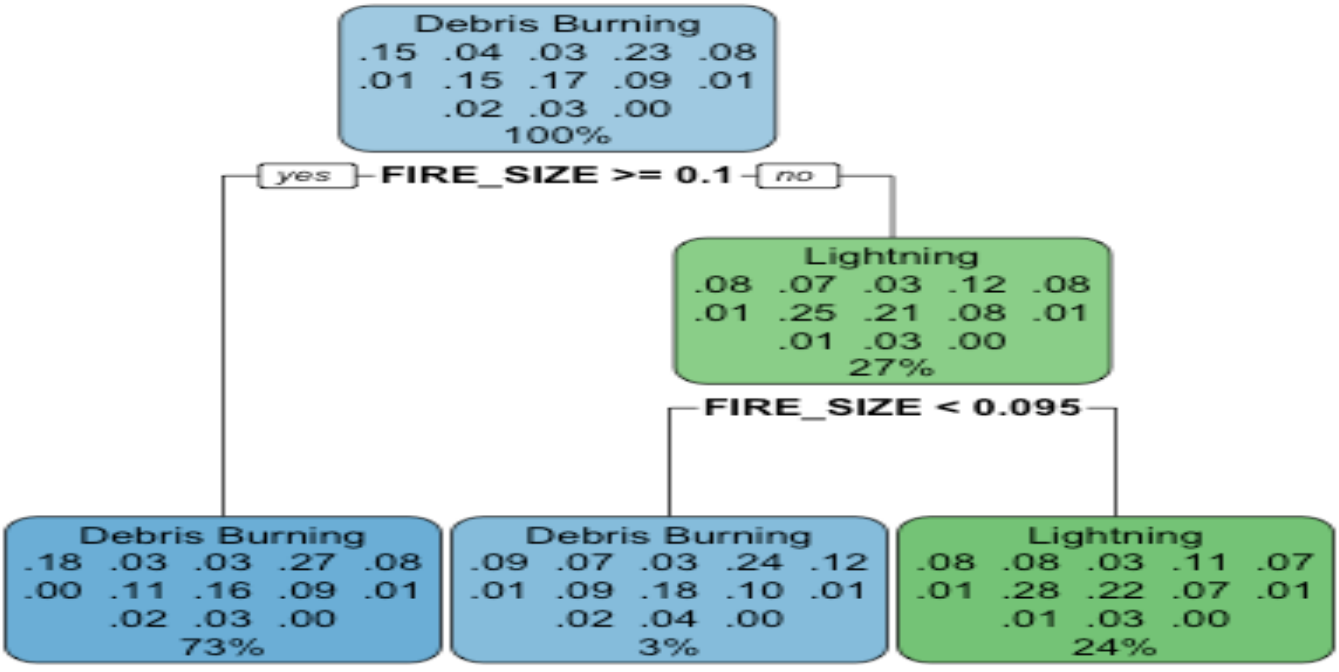
The above figure is the representation of the individual wildfire. We have shown 2010 New York wildfires caused by Campfire. One can know the name and size of the fire by hovering over each leaflet.

Data Preparation and Preprocessing

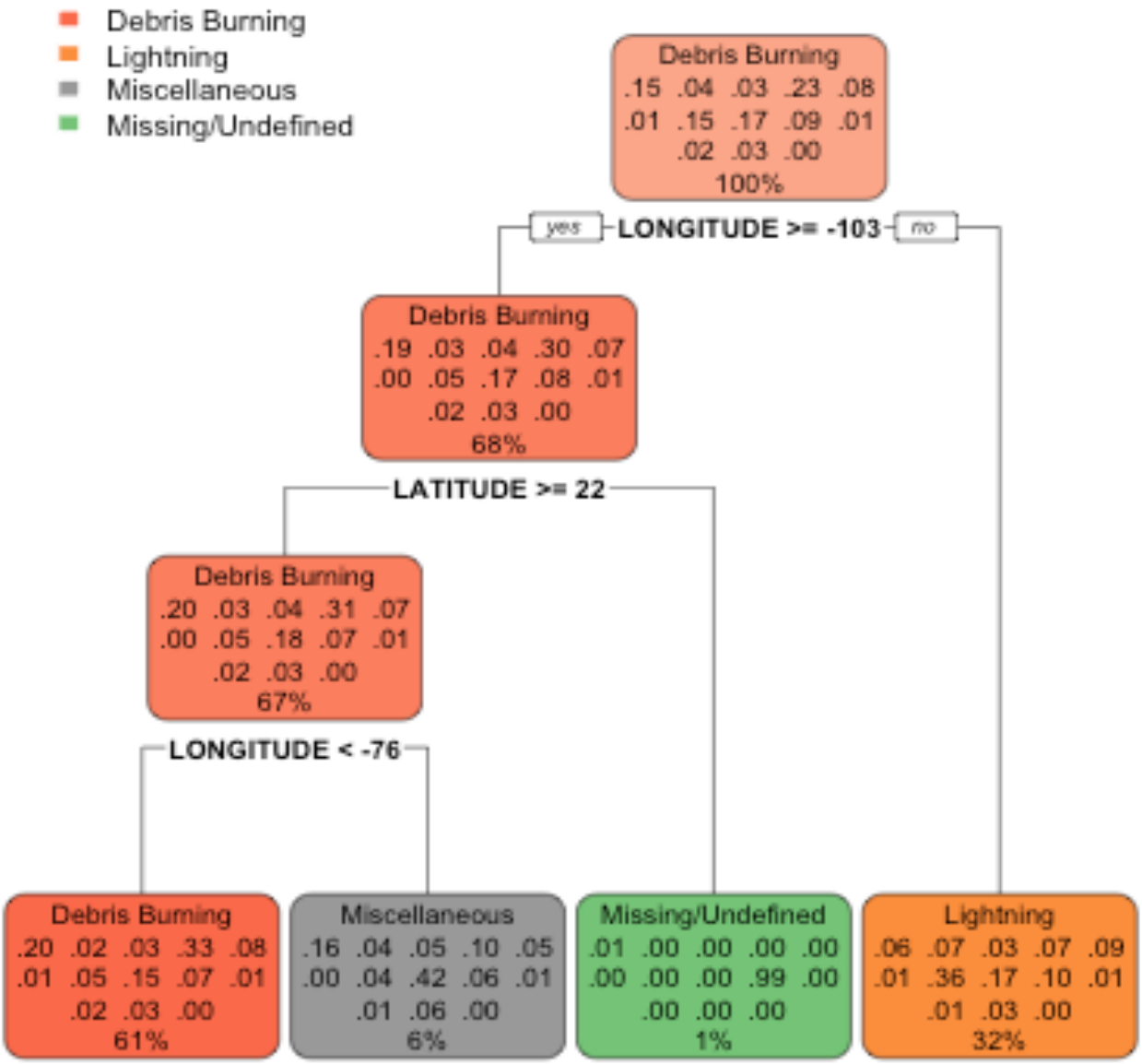
US Wildfire's dataset is already well preprocessed. The variables that we will be working on for visualizing and mining techniques are properly cleaned to ensure consistency and integrity of the data. For instance, variables like CONT_DATE, DISCOVERY_DATE store the date value in Julian format; using the pre-processing techniques, we will be converting the Julian format into standard date format.

Data Mining Techniques and Implementation

By visualizing the data set, we came to know that 'Debris Burning' is most common cause of wildfire as compared to other causes. For predicting, we will be using Classification and Regression Trees method. First, we will choose the features that we want to include in the given model, followed by splitting the parent dataset into training and test data. As a result, we will be able to form a decision tree. A benchmark will be set for accuracy. Eventually, we will introduce more variables like FIRE_YEAR, FIRE_SIZE, LATITUDE, LONGITUDE, etc. for further analysis. At the end, we'll cross validate the techniques in order to be sure about the derived results. We can keep working on the obtained decision tree, but that would keep increasing the number of variables to be analyzed, thereby increasing the complexity. Subsequently, the increasing complexity might lead to over-fitting. We don't want our model to simply 'memorize' the training data'. We want the model to generalize the unseen data and predict the values in later stages. So, we'll move from using a single decision tree to a combination of decision trees. We'll use **random forest** that builds multiple decision trees and combines them together to get a stable and an accurate prediction.



As you can see, the left model chose the first split at FIRE_SIZE >=0.1. From the plot, it is observed that 73% of the fires greater than or equal to 0.1 acres were caused due to debris burning. On the other hand, those with size less than 0.095 acres, majority of the fires were because of lightning while only 3% occurred due to debris burning.



In the second figure (right), you can see that 61% of the areas lying around LATITUDE >=22 & LONGITUDE < -76 experienced wildfires due to debris burning and 6% due to miscellaneous activities. 32% of the wildfires due to lightning were common in areas around the LONGITUDE >= -103.

Performance Evaluation

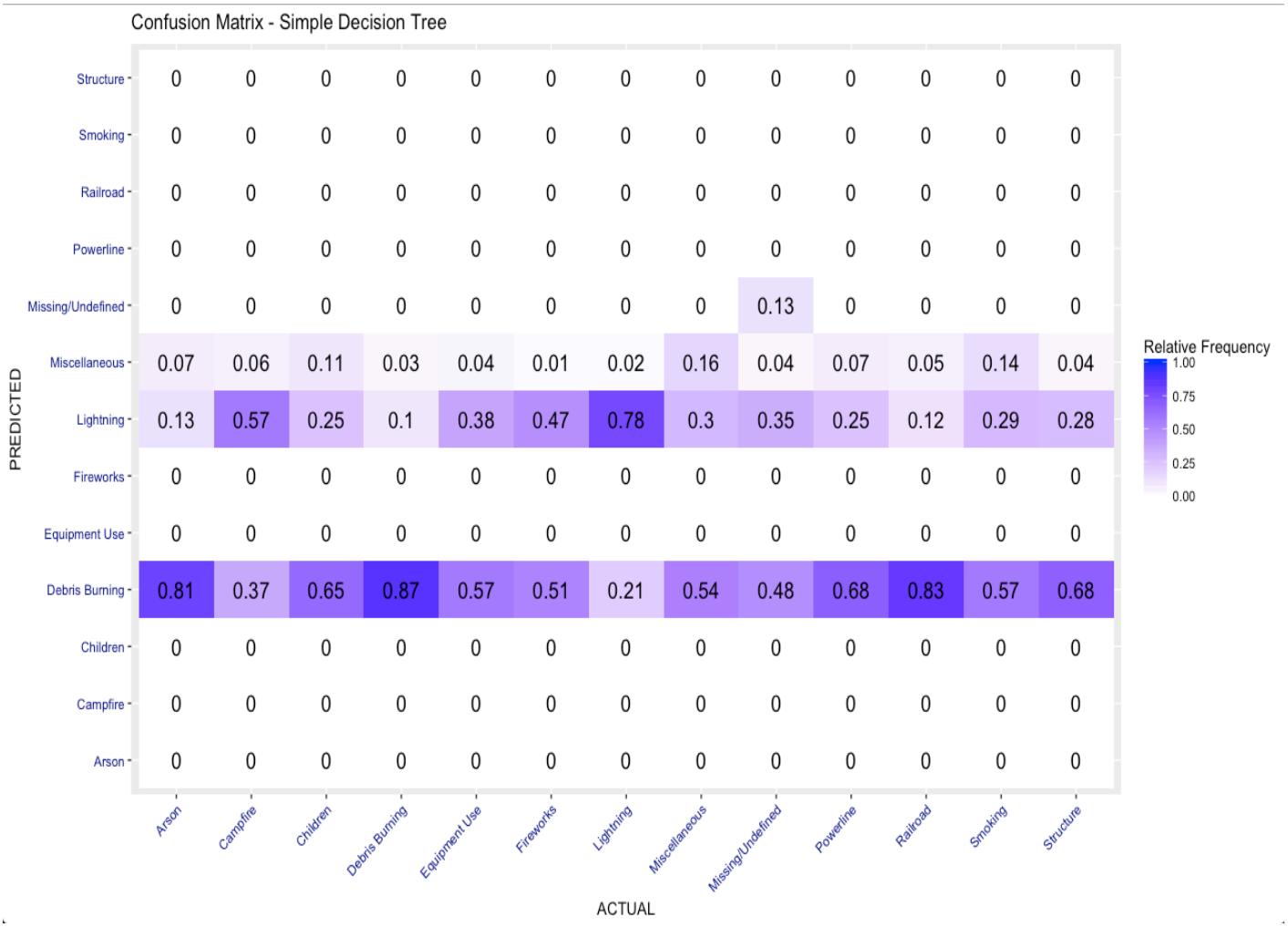
Our main focus of using this data is to predict the cause of wild fires. Using machine-learning algorithm, we want to predict the cause of wild fires with the acceptable level of precision that can be useful for investigators on the field.

Target Feature Analysis

Part 1 – Using a single feature

Firstly, we will select the class, which we want to predict because some classes have low frequency and we may run into difficulty in predicting these classes. After selecting the class, we will divide our data set into training data and test data.

Algorithm used here is a Regression tree. Before using the model, we have set a benchmark. If our model is not more accurate than our benchmark, then our model is useless. After obtaining the benchmark, we created a simple decision tree. This tree was created using only one feature i.e. FIRE_SIZE. Model is trained using training data and applied on test data to find the accuracy. After achieving the accuracy, we created confusion matrix to look deeper. The following is a confusion matrix heat map:



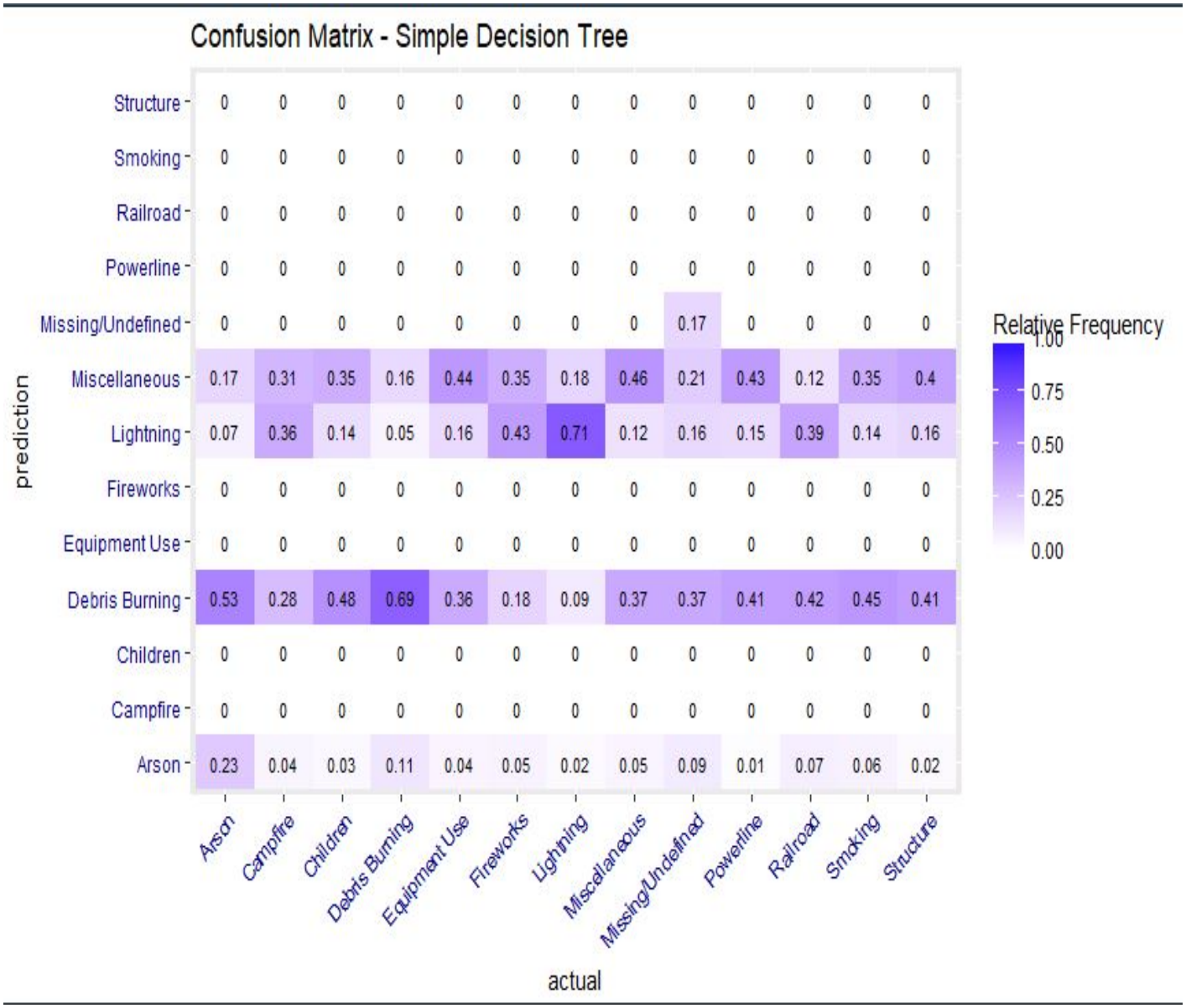
From the heat map, we can see that the actual classes are on the horizontal and the predicted classes are the vertical axis, with each tile showing the percentage of true class. For example, 47% of the observations labeled ‘Campfire’ have been predicted correctly. Similarly, 53% of the observations labeled ‘Campfire’ were incorrectly labeled ‘Lightning’.
The above plot is currently predicting four out of thirteen classes.

	Arson	Campfire	Children	Debris Burning	Equipment Use	Fireworks	Lightning	Miscellaneous	Missing/Undefined	Powerline	Railroad	Smoking	Structure
Arson	0	0	0	6444	0	0	1403	0	0	0	0	0	0
Campfire	0	0	0	987	0	0	1104	0	0	0	0	0	0
Children	0	0	0	1170	0	0	475	0	0	0	0	0	0
Debris Burning	0	0	0	10190	0	0	1902	0	0	0	0	0	0
Equipment Use	0	0	0	2897	0	0	1231	0	0	0	0	0	0
Fireworks	0	0	0	180	0	0	146	0	0	0	0	0	0
Lightning	0	0	0	3750	0	0	4120	0	0	0	0	0	0
Miscellaneous	0	0	0	5754	0	0	3211	0	0	0	0	0	0
Missing/Undefined	0	0	0	3463	0	0	1244	0	0	0	0	0	0
Powerline	0	0	0	268	0	0	129	0	0	0	0	0	0
Railroad	0	0	0	721	0	0	200	0	0	0	0	0	0
Smoking	0	0	0	992	0	0	516	0	0	0	0	0	0
Structure	0	0	0	62	0	0	47	0	0	0	0	0	0

We observe that the model has predicted only two values or classes – Debris Burning and Lightning.
Of the observations where,
A] Debris burning as the actual cause - the model predicted it correctly 7193 times.
B] Lightning as the actual cause – the model predicted it correctly 2952 times.
As a result, the data is partially inaccurate since the instances where the cause of wildfire was something else than the two predicted classes, current model was predicting Lightning at an unacceptable rate. This called for improvisation.

Part 2: Addition of features to improve the accuracy

Finally, we added more features to increase our accuracy. We used variables namely FIRE_SIZE, FIRE_YEAR, LATITUDE, LONGITUDE and DISCOVERY_DOY since they are numeric. As we've added more parameters, we are going to train this model allowing for more values of the complexity parameter. We can do this by increasing the tune Length parameter. After running the model, we found out the accuracy is improved. As you can see, the model is now predicting the 'Missing/Undefined' class along with 'Arson'. As you keep adding the features, the accuracy of the predictive analysis increases accordingly.



Part 3: Using Random Forests

The continuous addition of features increases the complexity of the model, thereby increasing the possibility of over-fitting. It is feasible to use a combination of decision trees that selects a set of features thereby seeking optimal splits at each node. We used the concept of random forests, where the runtime would be set to 100 trees (for instance). On running the model, we found the accuracy to be somewhere close to 53.42%.

Discussion and Recommendation

We have used selective features or variables from the dataset and implemented some machine learning algorithms to predict numerous things like wildfire causes, size category, their distribution, etc. After implementing a few algorithms, we cannot say that the model is cent percent accurate. Thus, there's always a scope of improvisation. The model can be improved in the following ways:

- Adding more features to the model.
- Use more variables/fields for predictive analysis.
- Using different algorithms like Neural Networks, KNN Clustering, etc.
- Try other performance metrics to get a better grasp on within class accuracy.

Summary

Natural calamities are difficult to predict and analyze. But we have tried to visualize and predict the causes of US wildfires. We took a collection of 23 years of wildfires that have been registered. We analyzed the dataset by figuring out the attributes that are correlated to each other and can be used for further research. Subsequently, we visualized different parameters such as causes of wildfire, number of wildfires over time, wildfire size, wildfire geography, etc. We created different plots and histogram to see the variation over a period of time. We also created a leaflet that represents individual fire over an area. Our main goal is to predict the causes of wildfires using machine-learning algorithm and how they might be related to other features. We performed decision tree with different parameters to see the accuracy of prediction. The accuracy increased as we were using more features for predictive analysis. Thereafter, we used random forest algorithm to differ the results with decision tree and analyze the accuracy. We recorded maximum accuracy with random forest and we could increase it by using more parameters, but we didn't perform further for the sake of simplicity. This case study helped us to learn detail knowledge of machine learning algorithm. Although we were able to increase the accuracy to its best possible value, there's always a possibility of improving it. Improvisation is possible by addition of different features. The more the number of features, the better is the accuracy of the model. But still getting accuracy of more than 50% is still an achievement as we are performing prediction on a real data set that was recorded by investigators. This case study can be useful for investigators who might want to dig deeper into the data set for their investigation.

References:

https://www.kaggle.com/
http://oobaloo.co.uk/visualising-classifier- results-with- ggplot2

Appendix

```
#load the required packages
install.packages("RSQLite")
library(RSQLite)
install.packages("dbplyr")
library(dbplyr)
install.packages("dplyr")
library(dplyr)
install.packages("purrr")
library(purrr)
library(ggplot2)
install.packages("xts")
library(xts)
install.packages("ggfortify")
library(ggfortify)
install.packages("ggthemes")
library(ggthemes)
install.packages("maps")
library(maps)
install.packages("mapdata")
library(mapdata)
install.packages("leaflet")
library(leaflet)

#create database connection
connect <- dbConnect(SQLite(), '/Users/nikhilmirchandani/Downloads/FPA_FOD_20170508.sqlite')

#pull the table into RAM
wildfires <- tbl(connect, "Fires") %>% collect()

#size check
print(object.size(wildfires), units = 'Gb')

# disconnect from the database
dbDisconnect(connect)

#quick overview
glimpse(wildfires)
```

```
#number of wildfires over time
wildfires %>%
  group_by(FIRE_YEAR) %>%
  summarise(n_fires = n()) %>%
  ggplot(aes(x=FIRE_YEAR, y=n_fires/1000)) +
  geom_bar(stat = 'identity', fill = 'lightblue') +
  geom_smooth(method = 'auto', se = FALSE, linetype = 'solid', size = 0.3, colour = 'darkred') +
  labs(x='Year', y = 'Number of wildfires across US (thousands)', title = 'US Wildfires by YEAR')
#Day of the year
wildfires %>%
  group_by(DISCOVERY_DOY) %>%
  summarise(NumberOfFires = n()) %>%
  ggplot(aes(x=DISCOVERY_DOY, y=NumberOfFires)) + geom_line(color = "Blue")
  geom_smooth(method = 'lm', se = FALSE, linetype = 'dashed', size = 0.4, color = 'red') +
  labs(x='', y = 'Number of wildfires across US', title = 'US Wildfires by Day of the Year')


#Wildfire Size
SizeCategory <- c('A' = '0-0.25', 'B' = '0.26-9.9', 'C' = '10.0-99.9', 'D' = '100-299', 'E' = '300-999',
                  'F' = '1000-4999', 'G' = '5000+')
wildfires %>%
  group_by(FIRE_SIZE_CLASS) %>%
  summarize(n = n()) %>%
  mutate(FIRE_SIZE_CLASS = SizeCategory[FIRE_SIZE_CLASS]) %>%
  ggplot(aes(x = FIRE_SIZE_CLASS, y= n)) +
  geom_bar(stat = 'identity', fill = 'lightblue') +
  labs(x = 'Fire size (acres)', y = 'Number of fires', title = 'Number of Wildfires by Size Category')


#Cause of wildfire
wildfires %>%
  group_by(STAT_CAUSE_DESCR) %>%
  summarize(cause = n()/1000) %>%
  ggplot(aes(x = reorder(STAT_CAUSE_DESCR, cause), y = cause)) +
  geom_bar(stat = 'identity', fill = 'darkblue') + coord_flip() +
  labs(x = '', y = 'Number of fires (thousands)', title = 'US Wildfires by Cause 1992 to 2015')


# Relationship between the cause and the size of fires
fires %>%
  group_by(STAT_CAUSE_DESCR) %>%
  summarize(MeanSize = mean(FIRE_SIZE, na.rm = TRUE)) %>%
  ggplot(aes(x = reorder(STAT_CAUSE_DESCR, MeanSize), y = MeanSize)) +
  geom_bar(stat = 'identity', fill = 'darkblue') +
  coord_flip() +
  labs(x = '', y = 'Number of fires (thousands)', title = 'US Wildfires by Cause 1992 to 2015')
```

```
#Representation of individual fire
wildfire %>%
  filter(STATE == "NY", FIRE_YEAR == "2010", STAT_CAUSE_DESCR == "Campfire") %>%
  leaflet() %>%
  #setView(lat = -0.900653, lng = -78.467834, zoom = 7) %>%
  addTiles() %>%
  addMarkers(
    ~LONGITUDE,
    ~LATITUDE,
    label = ~paste("Name:", FIRE_NAME, "Size:", FIRE_SIZE, "Acres")
  )

# DATA MINING – Decision Tree
features <- c('FIRE_SIZE', 'FIRE_YEAR', 'LATITUDE', 'LONGITUDE')
wildfire$STAT_CAUSE_DESCR <- as.factor(wildfire$STAT_CAUSE_DESCR)

# index for train/test split
set.seed(123)
train_index <- sample(c(TRUE, FALSE), nrow(wildfire), replace = TRUE, prob = c(0.85, 0.15))
test_index <- !train_index

# Create x/y, train/test data
x_train <- as.data.frame(wildfire[train_index, features])
y_train <- wildfire$STAT_CAUSE_DESCR[train_index]
x_test <- as.data.frame(wildfire[test_index, features])
y_test <- wildfire$STAT_CAUSE_DESCR[test_index]

predict <- rep('Debris Burning', length(y_test))

TestAccuracy <- round(sum(y_test == predict)/length(predict), 4)
print(paste(c("Accuracy:" , TestAccuracy)))
tr_control <- trainControl(method = 'cv', number = 3)

# Train the decision tree model
set.seed(123)
dtree <- train(x = x_train,
               y = y_train,
               method = 'rpart',
               trControl = tr_control)

# make predictions using test set
preds <- predict(dtree, newdata = x_test)

# calculate accuracy on test set
test_set_acc <- round(sum(y_test == preds)/length(preds), 4)
print(paste(c("Accuracy:" , test_set_acc)))
print(dtree$resample)
rpart.plot(dtree$finalModel)
```

```
# Performance Evaluation – Part 1
confusionMatrix(y_test, preds)$table %>%
  prop.table(margin = 1) %>%
  as.data.frame.matrix() %>%
  rownames_to_column(var = 'ACTUAL') %>%
  gather(key = 'PREDICTED', value = 'freq',-ACTUAL) %>%
  ggplot(aes(x = ACTUAL, y = PREDICTED, fill = freq)) +
  geom_tile() +
  geom_text(aes(label = round(freq, 2)), size = 5, color = 'black') +
  scale_fill_gradient(high = 'blue', low = 'white', limits = c(0,1), name = 'Relative Frequency') +
  theme(axis.text.x = element_text(angle = 45, hjust = 1, colour = "darkblue"),
        axis.text.y = element_text(angle = 0, hjust = 1, colour = "darkblue")) +
  ggtitle('Confusion Matrix - Simple Decision Tree')


#  Performance Evaluation – Part 2
features <- c('FIRE_YEAR', 'FIRE_SIZE', 'LATITUDE', 'LONGITUDE')
wildfire$STAT_CAUSE_DESCR <- as.factor(wildfire$STAT_CAUSE_DESCR)

# index for train/test split
set.seed(123)
train_index <- sample(c(TRUE, FALSE), nrow(wildfire), replace = TRUE, prob = c(0.85, 0.15))
test_index <- !train_index

x_train <- as.data.frame(fires[train_index, features])
y_train <- fires$STAT_CAUSE_DESCR[train_index]
x_test <- as.data.frame(fires[test_index, features])
y_test <- fires$STAT_CAUSE_DESCR[test_index]

predict <- rep('Debris Burning', length(y_test))
TestAccuracy <- round(sum(y_test == predict)/length(predict), 4)
print(paste(c("Accuracy:" , TestAccuracy)))
tr_control <- trainControl(method = 'cv', number = 3)

# Train the decision tree model
set.seed(123)
dtree <- train(x = x_train,
               y = y_train,
               method = 'rpart',
               tuneLength = 8,
               trControl = tr_control)

# make predictions using test set
preds <- predict(dtree, newdata = x_test)
```

```
# calculate accuracy on test set
test_set_acc <- sum(y_test == preds)/length(preds)
print(paste(c("Accuracy:" , round(test_set_acc, 4))))

confusionMatrix(y_test, preds)$table %>%
  prop.table(margin = 1) %>%
  as.data.frame.matrix() %>%
  rownames_to_column(var = 'actual') %>%
  gather(key = 'prediction', value = 'freq',-actual) %>%
  ggplot(aes(x = actual, y = prediction, fill = freq)) +
  geom_tile() +
  geom_text(aes(label = round(freq, 2)), size = 3, color = 'black') +
  scale_fill_gradient(low = 'white', high = 'blue', limits = c(0,1), name = 'Relative Frequency') +
  theme(axis.text.x = element_text(angle = 45, hjust = 1, colour = "darkblue"),
        axis.text.y = element_text(angle = 0, hjust = 1, colour = "darkblue")) +
  ggtitle('Confusion Matrix - Simple Decision Tree')
```