US Wildfire Analysis

INFO 6105 - Data Science Engineering Methods and Tools

Rutuja Kale [NU ID- 001592244]

OUTLINE

Introduction

Every year, the wildfires are destroying the areas, houses near the forest in the United States and around the world. It's high time to study the causes of wildfires in the area to avoid the situation in the future. Which will benefit the government to prevent the forest area from such activities. I will be working on wildfire data, how to use ML algorithm to predict the wildfire causes in the United States. I am taking the 'fire' column of this wildfire dataset and calculating the percentage accuracy for all possible causes that could initiate fires in top states.

Objective

Objective? Analyzing and predicting the wildfire causes?

Data Analysis Data visualization

Figuring out the top fire horizontal locations

Apply Machine Learning Algorithms to explore and predict the causes of wildfires in United States.

Dataset Description & Source Link

Data Source: https://www.kaggle.com/rtatman/188-million-us-wildfires

Reading and Pre-processing Data

Implemented sqlite3 to import the dataset.

```
print("Head:")
print(df.head())
```

```
Head:
  FIRE_CODE FIRE_NAME  FIRE_YEAR STAT_CAUSE_DESCR   LATITUDE   LONGITUDE  \
0      BJ8K  FOUNTAIN       2005    Miscellaneous  40.036944 -121.005833
1      AAC0    PIGEON       2004        Lightning  38.933056 -120.404444
2      A32W     SLACK       2004   Debris Burning  38.984167 -120.735556
3      None      DEER       2004        Lightning  38.559167 -119.913333
4      None  STEVENOT       2004        Lightning  38.559167 -119.933056

  STATE  FIRE_SIZE FIRE_SIZE_CLASS
0    CA       0.10               A
1    CA       0.25               A
2    CA       0.10               A
3    CA       0.10               A
4    CA       0.10               A
```

Importing data from Kaggle

```
conn = sqlite3.connect('/Users/rutuja/Downloads/FPA_FOD_20170508 (1).sqlite')
data = pd.read_sql(
                   """

                    SELECT *
                    from fires

                   """, con=conn)
```

1.Importing data

Reading and preprocessing the data

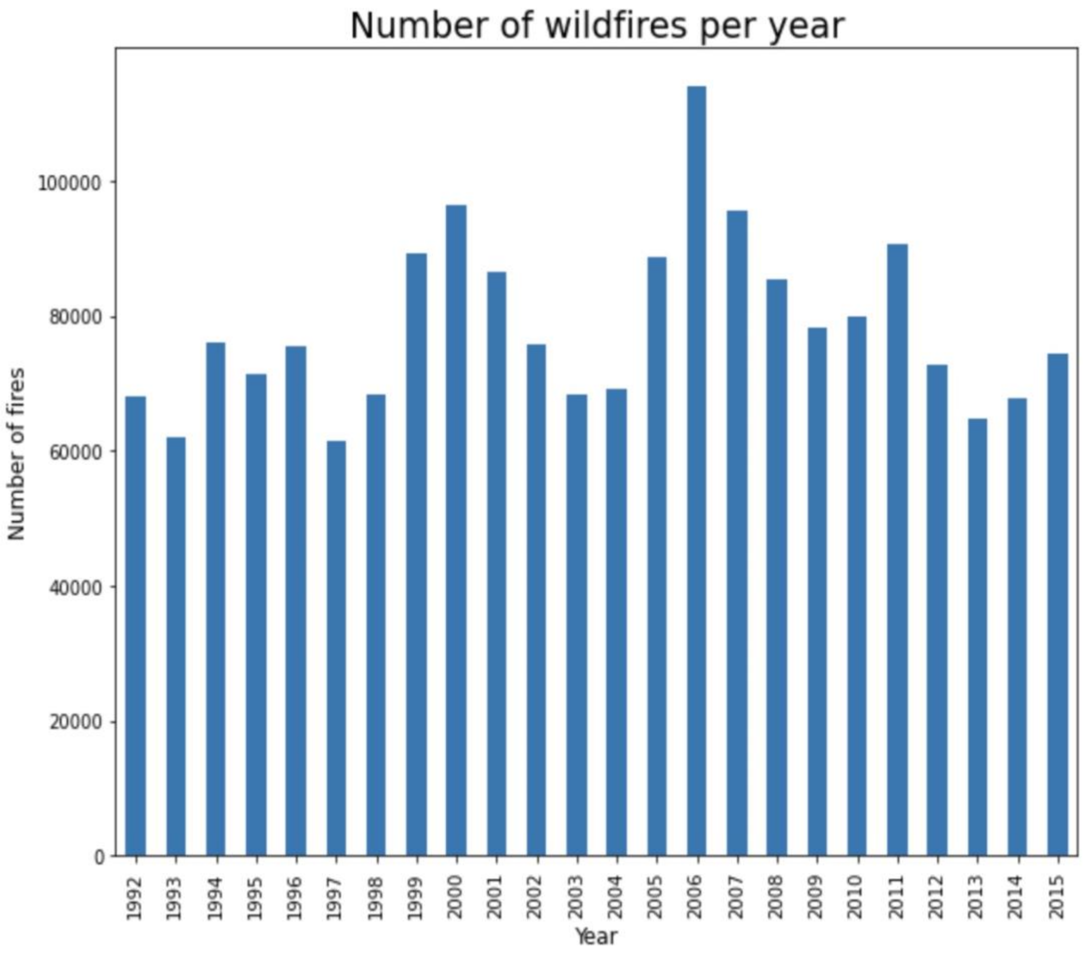
```
  FIRE_CODE FIRE_NAME  FIRE_YEAR STAT_CAUSE_DESCR   LATITUDE   LONGITUDE  \
0      BJ8K  FOUNTAIN       2005    Miscellaneous  40.036944 -121.005833
1      AAC0    PIGEON       2004        Lightning  38.933056 -120.404444
2      A32W     SLACK       2004   Debris Burning  38.984167 -120.735556
3      None      DEER       2004        Lightning  38.559167 -119.913333
4      None  STEVENOT       2004        Lightning  38.559167 -119.933056

  STATE  FIRE_SIZE FIRE_SIZE_CLASS DISCOVER_DATE CONTROL_DATE
0    CA       0.10               A    2005-02-02   2005-02-02
1    CA       0.25               A    2004-05-12   2004-05-12
2    CA       0.10               A    2004-05-31   2004-05-31
3    CA       0.10               A    2004-06-28   2004-07-03
4    CA       0.10               A    2004-06-28   2004-07-03
```

2. Data Columns

Visualization

1.Number of Wildfires per year

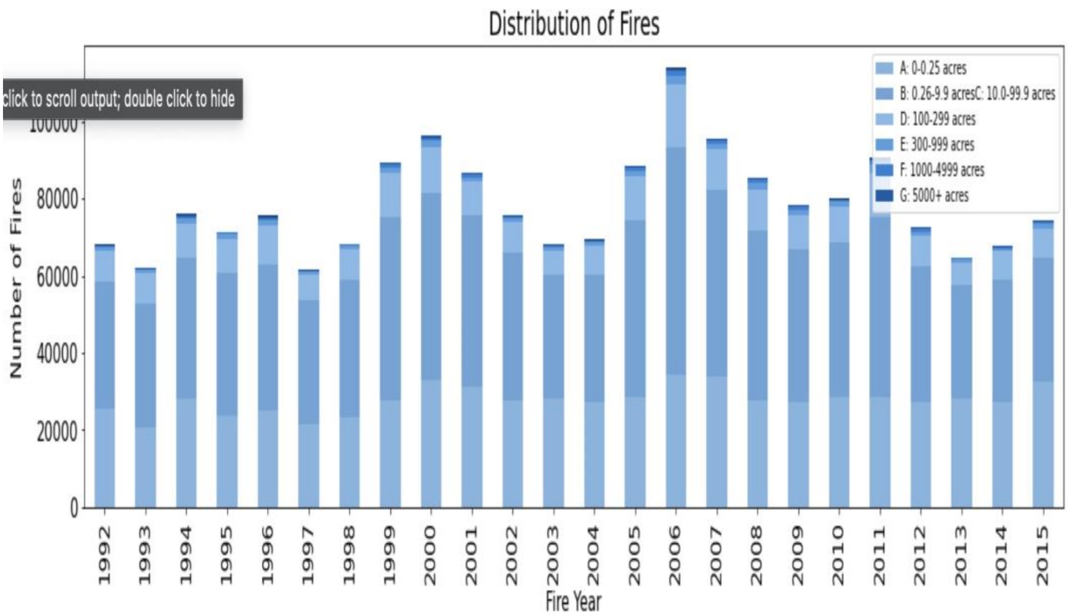
I have created a bar plot of fires per year. In 2006 maximum incidents of wildfires took place. Around 10,000 - 15,000 incidents of wildfire take place every year.



3: Timeline and Fires Per Year

Number of fires for each class per year

Analyze the wildfire occurrences based on fire size class.



4: Distribution of Fires

Consider the wildfire incidences based on fire size class. Taking the following Fire size column to predict the causes.

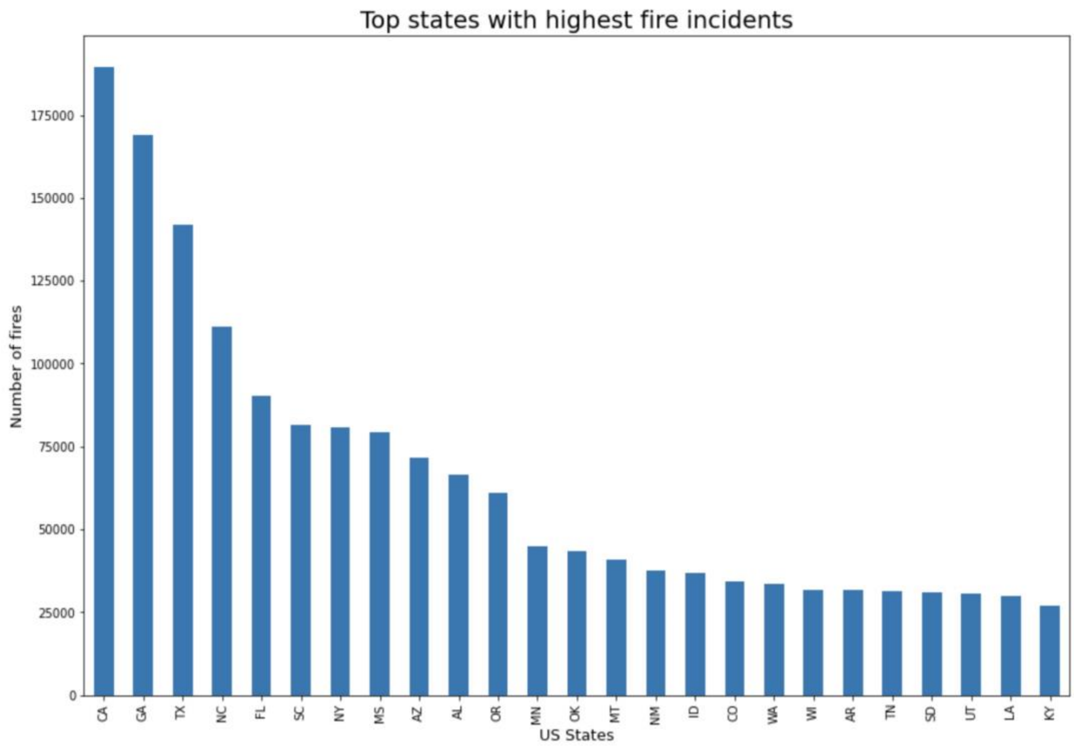
A= 0- 0.25 acres
B= 0.26-9.9 acres,
C=10.0-99.9 acres,
D=100-299 acres,
E=300 to 999 acres,
F=1000 to 4999 acres, and
G=5000+ acres

Analyzing the incidence of large fires are more than 5000 which is class G.

Top states with highest fire incidents

Result: CA, GA, TX, are more top three more susceptible to fire. So, I have decided to use this data to predict the causes of wildfires.

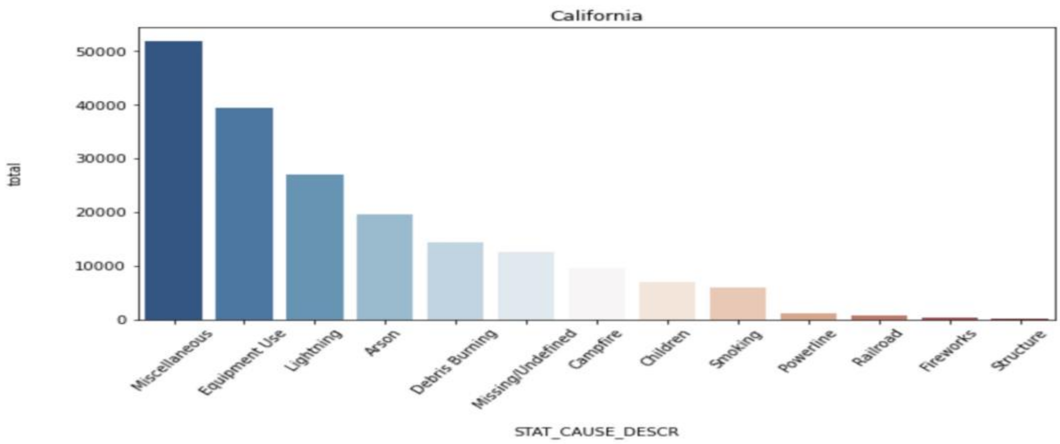
Next question comes is what caused this high probability of wildfire, so here I have picked top three states to identify the causes of wildfire.



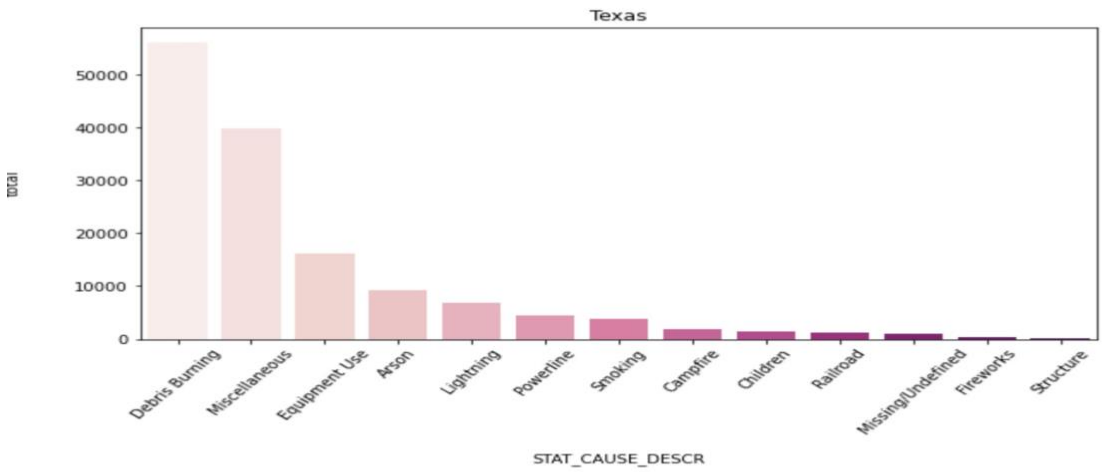
5: Top states with highest fire incidents

There is a very small section of natural source for wildfire. Most of the wildfire is started because of human action such as Debris Burning, Arson (malicious)

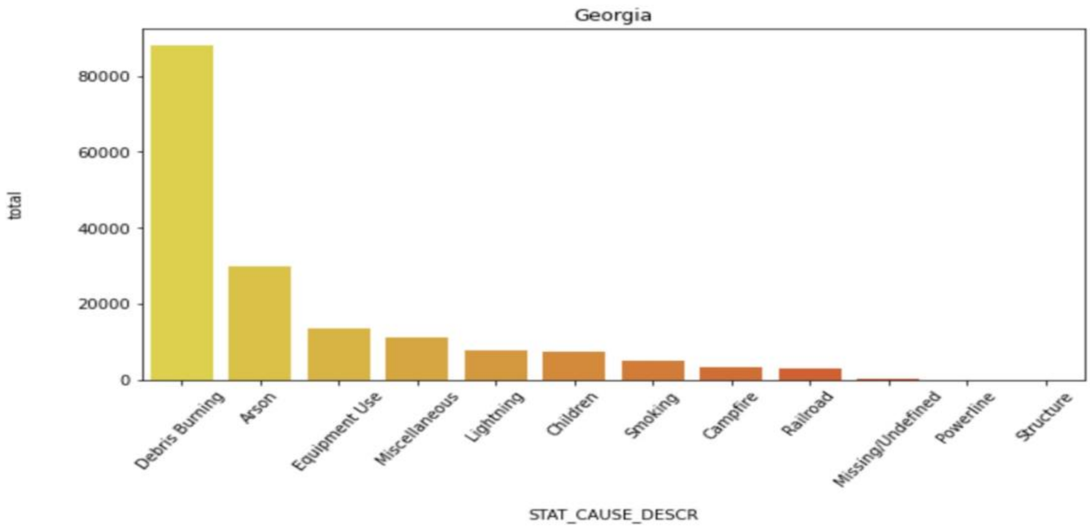
Number of Wildfires with Causes CA, TX, GA



6: Number of Wildfires with Causes CA



7: Number of Wildfires with Causes TX



8: Number of Wildfires with Causes GA

Transforming the states and causes to numeric records.

```
fire_data = df.copy()
```

```
#Coverting the states and causes to numeric numbers.

le = preprocessing.LabelEncoder()
fire_data['STAT_CAUSE_DESCR'] = le.fit_transform(df['STAT_CAUSE_DESCR'])
fire_data['STATE'] = le.fit_transform(df['STATE'])
```

Number of Causes of Wildfire are as below

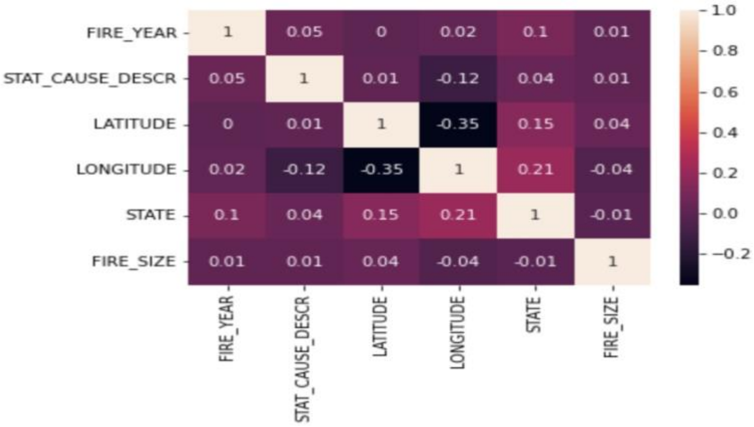
```
df.STAT_CAUSE_DESCR.unique()
```

```
array(['Miscellaneous', 'Lightning', 'Debris Burning', 'Campfire',
       'Equipment Use', 'Arson', 'Children', 'Railroad', 'Smoking',
       'Powerline', 'Structure', 'Fireworks', 'Missing/Undefined'],
      dtype=object)
```

Correlation

I have moved the states and causes to numeric digits. calculated the correlations but the dataset didn’t show a strong correlation.

```
<AxesSubplot:>
```



It is not possible to predict the fire causes now, I used ML algorithms on the dataset to evaluate, and predict accuracy.

Machine Learning Algorithm

List of machine algorithm I am using in this project:

1. Gaussian Naïve Bayes
2. Decision Tree
3. Random Forest

Gaussian Naive Bayes

Here, I have Imported the Gaussian Naive Bayes model, created a Gaussian Classifier.

```python
# Initialize Gaussian from SkLearn
from sklearn.metrics import accuracy_score
from sklearn.naive_bayes import GaussianNB
```

```python
# Fit the model on the X_train and y_train data
X=test.drop(['STAT_CAUSE_DESCR'],axis=1).values
y=test['STAT_CAUSE_DESCR'].values
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=0)
nb_classifier = GaussianNB()
nb_classifier.fit(X_train, y_train)
y_pred_nb = nb_classifier.predict(X_test)
predicted_result=nb_classifier.predict(X_test)
```

Trained and tested the model and projected the response for the test dataset.

```
              precision    recall  f1-score   support

           0       0.25      0.00      0.00     84170
           1       0.00      0.00      0.00     22818
k to expand output; double click to hide output ).07      0.04     18240
           3       0.26      0.93      0.41    129099
           4       0.32      0.00      0.01     44329
           5       0.03      0.06      0.04      3415
           6       0.50      0.02      0.04     83316
           7       0.21      0.07      0.10     97026
           8       0.75      0.20      0.31     50300
           9       0.00      0.00      0.00      4289
          10       0.00      0.00      0.00     10053
          11       0.00      0.00      0.00     15925
          12       0.00      0.00      0.00      1160

    accuracy                           0.25    564140
   macro avg       0.18      0.10      0.07    564140
weighted avg       0.30      0.25      0.15    564140
```

And finally came to an accuracy of **25%,** which is not good. So, moving to the next ML algorithm Decision tree.

Decision Tree

Collected the data required and converted data to numeric numbers.

```python
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=0.3, random_state=0)
```

```python
d_tree=DecisionTreeClassifier(max_depth=2,random_state=0)
```

```python
d_tree.fit(X_train,y_train)
```

```
DecisionTreeClassifier(max_depth=2, random_state=0)
```

```python
dt_tree=DecisionTreeClassifier(max_depth=2,random_state=0)
```

```python
dt_tree.fit(X_train,y_train)
```

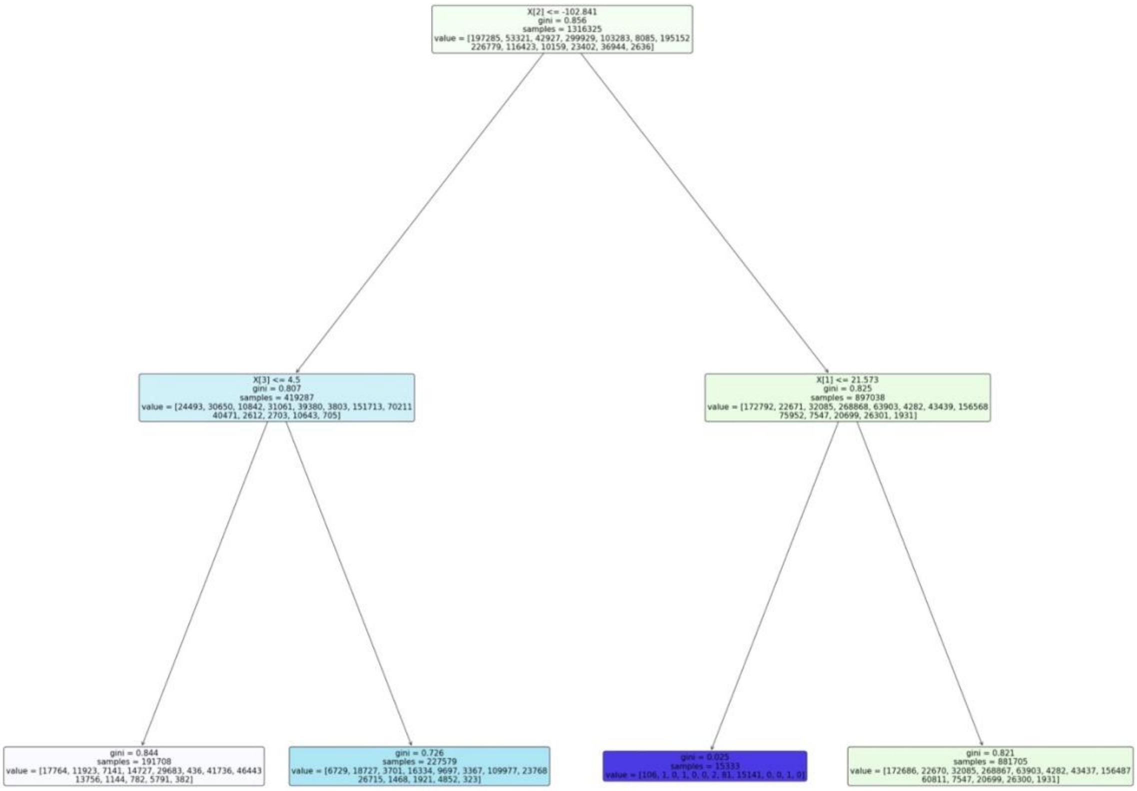
```
DecisionTreeClassifier(max_depth=2, random_state=0)
```

After training and testing data, I got accuracy score of 33%.

```
#decision tree socre
score=dt_tree.score(X_test,y_test)
score
```

```
0.3342042755344418
```

The accuracy is not what we wanted so trying to predict the accuracy using last algorithm Random Forest.



Random forest

Imported the Random Forest, created a Random Forest Classifier.

```
rf_classifier = ske.RandomForestClassifier(n_estimators=50)
rf_classifier = rf_classifier.fit(X_train, y_train)
print(rf_classifier.score(X_test,y_test))
```

```
0.5787020952245896
```

The 58% score is better than the Gaussian and Decision tree accuracy results.

I have decided to classify the fire causes into 4 classes: natural causes, accidental causes, malicious causes, and other causes.

```
def set_label(fire_cause):
    cause = 0
    natural = ['Lightning']
    accidental = ['Structure','Fireworks','Powerline','Railroad','Smoking','Children','Campfire','Equipment Use','Debri
    malicious = ['Arson']
    other = ['Missing/Undefined','Miscellaneous']
    if fire_cause in natural:
        cause = 1
    elif fire_cause in accidental:
        cause = 2
    elif fire_cause in malicious:
        cause = 3
    else:
        cause = 4
    return cause


dt['LABEL'] = dt_orig['STAT_CAUSE_DESCR'].apply(lambda x: set_label(x))
dt = dt.drop('STAT_CAUSE_DESCR',axis=1)
print(dt.head())
```

Four classes of wildfire

```
   FIRE_YEAR   LATITUDE   LONGITUDE  STATE  DISCOVERY_DATE  FIRE_SIZE  MONTH  \
0       2005  40.036944 -121.005833      4       2453403.5       0.10      2
1       2004  38.933056 -120.404444      4       2453137.5       0.25      5
2       2004  38.984167 -120.735556      4       2453156.5       0.10      5
3       2004  38.559167 -119.913333      4       2453184.5       0.10      6
4       2004  38.559167 -119.933056      4       2453184.5       0.10      6

   DAY_OF_WEEK  LABEL
0            6      4
1            6      1
2            1      2
3            1      1
4            1      1
```

four classes of wildifre reasons

Then, did trained and tested new dataset, which gave better accuracy is 70%.

```
#random forest test based on the new dataset gave a 70% score.
X = dt.drop(['LABEL'], axis=1).values
y = dt['LABEL'].values
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=0.3, random_state=0)
rf_classifier = ske.RandomForestClassifier(n_estimators=50)
rf_classifier = rf_classifier.fit(X_train, y_train)
print(rf_classifier.score(X_test,y_test))

0.7010706562200872
```

Model Improvement

After reducing the number of elements model gave the accuracy of 70% using a random forest algorithm. But the accuracy is not good for malicious part which is Arson.

The previous model has the US wildfires and try to classify the causes into four classes. So, I have decided to perform random forest algorithm on single state with one fire cause, malicious.

Next step is to predict malicious fires on top state. CA, GA, and TX. To improve the precision, drop some columns to do that.

Predict malicious fires in CA, TX, GA

CA

```
X = dt_CA.drop(['ARSON'], axis=1).values
y = dt_CA['ARSON'].values
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=0.3, random_state=0)
rf_classifier = ske.RandomForestClassifier(n_estimators=200)
rf_classifier = rf_classifier.fit(X_train, y_train)
print(rf_classifier.score(X_test,y_test))
```

```
0.92151587092236
```

CA (California) State: accuracy is 91%

GA

```
X = dt_GA.drop(['ARSON'], axis=1).values
y = dt_GA['ARSON'].values
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=0.3, random_state=0)
rf_classifier= ske.RandomForestClassifier(n_estimators=200)
rf_classifier = rf_classifier.fit(X_train, y_train)
```

```
print(rf_classifier.score(X_test,y_test))
```

```
0.8574840607173171
```

GA (Georgia) State: accuracy is 85%

TX

```
X = dt_TX.drop(['ARSON'], axis=1).values
y = dt_TX['ARSON'].values
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=0.3, random_state=0)
rf_classifier = ske.RandomForestClassifier(n_estimators=200)
rf_classifier = rf_classifier.fit(X_train, y_train)
print(rf_classifier.score(X_test,y_test))
```

```
0.9438120496632009
```

TX (Texas) State: accuracy is 94% which the highest accuracy to predict the fire causes

Conclusion

Implemented required data pre-processing, apply numerous classification and regression models to predict the causes for wildfires. Also, increase the accuracy of the predictions using modification and validation methods. In this project, I have implemented multiple machine learning methods to predict wildfire causes. The models I have attempted to apply to include Decision Tree, Gaussian, and Random Forest machine learning algorithms. But observed partial accuracy amongst all models on the present dataset. Random forest came out to be the finest model and gave an accuracy of around 70 % but didn't work well for the Arson portion. Hence, in the next model, I drop a few columns and work on only 1 state to check the wildfire caused in that state is Arson or not. Random Forest overall percentage accuracy is 91 % which is inordinate to predict causes of wildfires in the United States.

References

k, R. (2021, December 15). *Wildfires*. NASA. Retrieved December 15, 2021, from https://earthdata.nasa.gov/learn/toolkits/wildfires

1.88 million US wildfires. Kaggle. (n.d.). Retrieved December 15, 2021, from https://www.kaggle.com/rtatman/188-million-us-wildfires

Sklearn naive Bayes classifier python: Gaussian naive Bayes Scikit-Learn tutorial. DataCamp Community. (n.d.). Retrieved December 15, 2021, from https://www.datacamp.com/community/tutorials/naive-bayes-scikit-learn

1.10. decision trees. scikit. (n.d.). Retrieved December 15, 2021, from https://scikit-learn.org/stable/modules/tree.html

Tyagi, N. (2020, September 30). *Understanding the Gini index and information gain in decision trees*. Medium. Retrieved December 15, 2021, from https://medium.com/analytics-steps/understanding-the-gini-index-and-information-gain-in-decision-trees-ab4720518ba8

Sklearn Random Forest classifiers in Python. DataCamp Community. (n.d.). Retrieved December 15, 2021, from https://www.datacamp.com/community/tutorials/random-forests-classifier-python

Science, M. D. (2021, August 17). *How to predict the cause of wildfires using a random forest classifier*. Mr. Data Science. Retrieved December 15, 2021, from https://mrdatascience.com/how-to-predict-the-cause-of-wildfires-using-a-random-forest-classifier