# Application of Data Science Algorithms on US Fire Data

Appilineni Kushal

June 2021

## Introduction

A fire could originate inside a forest in many ways - a bonfire camping gone wrong, discarded cigarette near dry leaves/grass, a lightning striking a tree, heat of the sun or even ignition at the roots of plants. These fires then spread rapidly throughout the forests/grasslands and prairies causing huge losses in natural habitat of many animals and plant species, human communities etc. It also affects the environment as smokes cover most of the nearby regions causing breathing hazard to people and animals.It is almost impossible to prevent. But not all is bad about these wildland fires, in small numbers it can be benefitial too. Frequent fires can help forests not get too crowded as fire dependent species would disappear. It also helps enriching forest soil with nutrients and fuels and thus promotes growth of native species of plants and trees. It also protects human communities from extreme fires, which could harm human communities and forests. An example of such extreme fires was seen in 2020. One of the factors leading to such an extreme wildfire was the accumulation of dry leaves over recent years, due to fire prevention efforts, according to climate scientists Robert Rohde and Zeke Hausfather. Thus, wildfires have both negative and positive impacts in our ecosystems and is thus important for us to understand the wildfire data and extract information from it.

In today's world, there are many data science techniques available at our disposal to analyze and model systems around us. These data science techniques have found applications in the most diverse of fields, from analyzing genome data in our nucleus to data about stars and quasars far away in the universe and everything in between.

In this project, we make use of some of these data science techniques to analyze the wildland fire data in US. We first use dimension reduction techniques followed by clustering. At its best, this would help us visualize the data, understand general trends (if any) and outliers to the trends. Such clustering efforts if successful would open new questions into exploring what points belonging to a given clusters share in common and what are the differences amongst these clusters. After that, we work on classification problems. In specific, we check whether data science classifier models could be used to classify between human caused and lightning caused fires. If successful, this would be an

| | Year | #human incidents | #human area | #lightining incidents | #lightning area | Geographical Region |
|---|---|---|---|---|---|---|
| **0** | 2020 | 181 | 284 | 168 | 180885 | Alaska |
| **1** | 2019 | 349 | 44061 | 371 | 2454098 | Alaska |
| **2** | 2018 | 229 | 28946 | 138 | 381737 | Alaska |
| **3** | 2017 | 209 | 6890 | 155 | 646133 | Alaska |
| **4** | 2016 | 343 | 10069 | 229 | 486398 | Alaska |
| **...** | ... | ... | ... | ... | ... | ... |
| **195** | 2005 | 28920 | 509082 | 516 | 67982 | Southern Area |
| **196** | 2004 | 27758 | 407456 | 958 | 55341 | Southern Area |
| **197** | 2003 | 16479 | 248412 | 489 | 45048 | Southern Area |
| **198** | 2002 | 31394 | 356204 | 825 | 155530 | Southern Area |
| **199** | 2001 | 34605 | 761605 | 392 | 545983 | Southern Area |

Figure 1: Wildland Fire dataset

important tool in classifying unreported fires, which would improve our understanding of wildland fires.

# Data

Wildland Fires in US can be broadly categorized into Human caused fires and Lightning caused fires. I found this wildland fire data here - Wildland Fire data. The data comprises of 4 factors - no of incidents of fires caused by humans, no of incidents of fire caused by lightning, area (in acres) of land affected by human incidents, area (in acres) of land affected by lightning incidents(see Figure 1). This data is compiled from 2001 - 2020 for different geographic areas across US, namely - Alaska, Northwest, Northern California, Southern California, Northern Rockies, Great Basin, Southwest, Rocky Mountains, Eastern Area and Southern Area (as shown in Figure 2)

## Data Visualization

Since our data consists of 4 variables - no of incidents of fires caused by humans, no of incidents of fire caused by lightning, area (in acres) of land affected by human incidents, area (in acres) of land affected by lightning incidents - it'll be hard to visualize such a data. One could simply plot 2 of the 4 components but it wont capture all the information and variance in the data. Thus, to visualize we use linear and non linear dimension reduction techniques. In particular, we use, the following dimension reduction techniques:


Figure 2: This Figure shows the 10 different geographical regions for which the wildland fire data is available. Source: https://www.nifc.gov/nicc/index.htm

- **Principal Component Analysis** - This is a linear dimension reduction technique which projects the data into a lower dimension space such that maximum information is retained. We look at the covariance matrix $Cov(X)$ for our input matrix $X$ and find the eigenvectors and eigenvalues of this covariance matrix. PCA puts maximum information about the data into its eigenvectors in descending order.

  In our analysis, we first find the covariance matrix and then we find the eigenvalues of the covariance matrix. The eigenvalues were as follows:

  $$\text{eigenvalues} \quad 0.4126 \quad 0.092 \quad 0.271 \quad 0.222$$

  Thus, just using 3 of these eigenvectors associated with the 3 largest eigenvalues contributes 90% of the information in the data (see Figure 3). It's not a significant dimension reduction but atleast we can now visualize the data in 3D.

- **Isomaps** - This is non-linear dimension reduction technique which tries to preserve the distance between the points when projected to a lower dimension. Here, first the neighbors of each point is determined (i.e. all points whose euclidean distance is less than some value $r$) to give a neighborhood map. Each point in the neighborhood is connected with weight dependent on their euclidean distance. Then the shortest path between any two nodes are computed. Following this distance

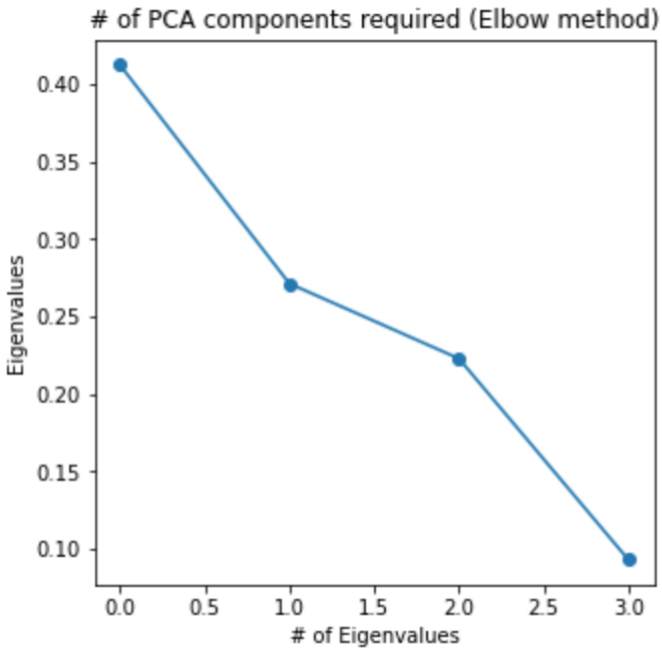


Figure 3: Eigenvalues of the covariance matrix. As seen, 3 eigenvalues are enough to capture most of the variance. It's not significant dimension reduction but atleast we can now visualize data in 3D, without losing any variance.

metric, the data is projected into the lower dimension such that the distances are preserved.

In our application, we used isomap with 5 neighbors in each neighborhood map and we projected the data into 3 dimesions so that we can compare it with PCA results.

- **t-distributed stochastic neighbor embedding** - This is another non-linear dimension reduction technique. It first constructs a probability distribution (in original dimension) such that similar (dissimilar) objects are grouped with a higher (lower) probability and then projects it to a similar probability distribution in lower dimension. Similarity between objects are usually measured by Euclidean distance.

  **In our application, we used tsne to project data into 3 components for comparision with PCA and isomap.**

## Clustering

After reducing the dimension, we implement the following clustering algorithms from unsupervised learning methods. The aim is to visualize any similarities in the data and group them together.Even though we've geographic region labels in the data, **unsupervised clustering is a way to determine similarities between datapoints which are not rooted in their geographic proximity.**

- First, we use **OneHotEncoder** to first turn the geographical region string labels into an indexing which we could later use to compare the clusters from unsupervised learning.

- Then we use data from each of the above dimension reduction methods and applied **Kmeans**. Kmeans is a clustering method which clusters given data into k clusters such that points in each cluster are closer to their respective cluster's centroid than other cluster's centroid. Its an iterative algorithm which starts with k random points as centroids, clusters the data into these k clusters depending on their euclidean distance from each of k centroids and then computes new centroids based on the mean of data in each cluster. It continues this procedure till convergence is reached.

- We also employ **spectral clustering + kmeans** to the original data. Spectral clustering is another way of dimension reduction + clustering method. Here, the smallest eigenvectors of the laplacian matrix is used to embed a high dimensional data into lower dimension. I've used a radial basis function $\phi(\mathbf{x}, \mathbf{y}) = e^{(-\gamma||\mathbf{x}-\mathbf{y}||_2^2)}$, with $\gamma = 1$.

  **Note:** I've tried $\gamma = 0.1, 0.01$ in my spectral clustering, the qualitative results still remain the same.

## Dimension Reduction and Clustering results

Figure 5 shows the plots of the data after dimension reduction and clustering. The plots in the first, second and third row have the first 3 components of Isomap, PCA and t-SNE as axis. The left most column is the raw data i.e. the colors indicate the original geographical labelling which came with the data. The second column shows each of Isomap, PCA and t-SNE data grouped into 3 clusters using Kmeans. The third column represents clustering of the data using Spectral clustering method. As seen in the figure, despite geographical labels, most data appears close to each other, except for a few outliers. This hypothesis about groups and outliers could be supported by the Figure 4. Here, I've plotted total number of fire incidents (left) and total area affected (right) for Alaska, Southern Area and rest of the geographical regions combined. As one can see, the datapoints corresponding to Alaska (0-20 on X axis) have some outliers in it's data for total area affected (see Figure 4 (right)), than rest of the regions. These values of total area affected in Alaska is much higher than rest of the regions, making it a clear outlier. In addition, the number of incidents in Southern Area is much more than the rest of the regions, making it another outlier. Interestingly, the green and yellow points in the spectral clustering plot (see Figure 5) correspond to these outliers. Therefore, we could obtain meaningful clustering from Spectral Clustering method.

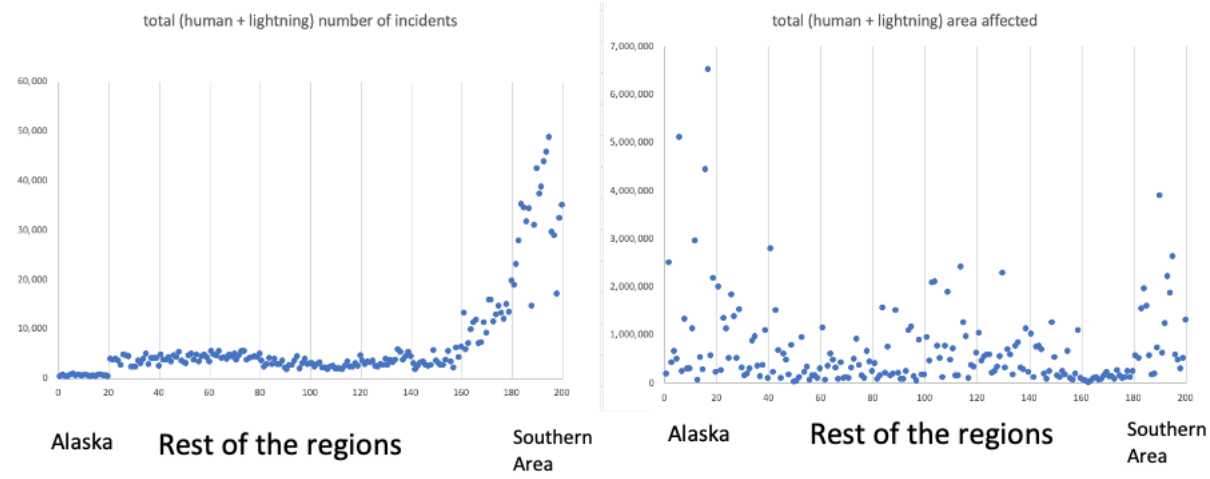


Figure 4: On the left, the total number of incidents are plotted and on the right, the total ares of area affected by the fire is plotted. As we can see, in Alaska, which is the 0-20 on X axis in both the plots, the number of incidents is much lower than rest of the regions, whereas the areas affected is more. Similarly, in Southern Area, the number of incidents is much higher than rest of the regions. This clearly shows us that the points related to Alaska and Southern Area are outliers, as supported by spectral clustering(see Figure 5)

# Binary Classifications

Another data science task one could perform with this wildland dataset is classification between Human caused fires and Lightning caused fires. For each of these classes we've 200 data points (corresponding to the years 2001 - 2020 for 10 geographic regions). Therefore all in all, we have 400 datapoints, where each datapoint (having 2 columns i.e. number of incidents & area of land affected) could belong to either Human caused fire or Lightning caused fire.

## Methods

Here are the simulation methods:

- At first, we split the data into training and test sets, with test set containing about 10% of the total data. The training data would be used to train the classifiers and the test data would be used to evaluate the accuracy of the classifier.
- We further divide the training data into newtraining and validation sets. Since each classification model has a wide range of tunable parameters, we first train the classifier on newtraining set and then test it on validation set. This way, we could trace the tunable parameters which perform the best in our training set.

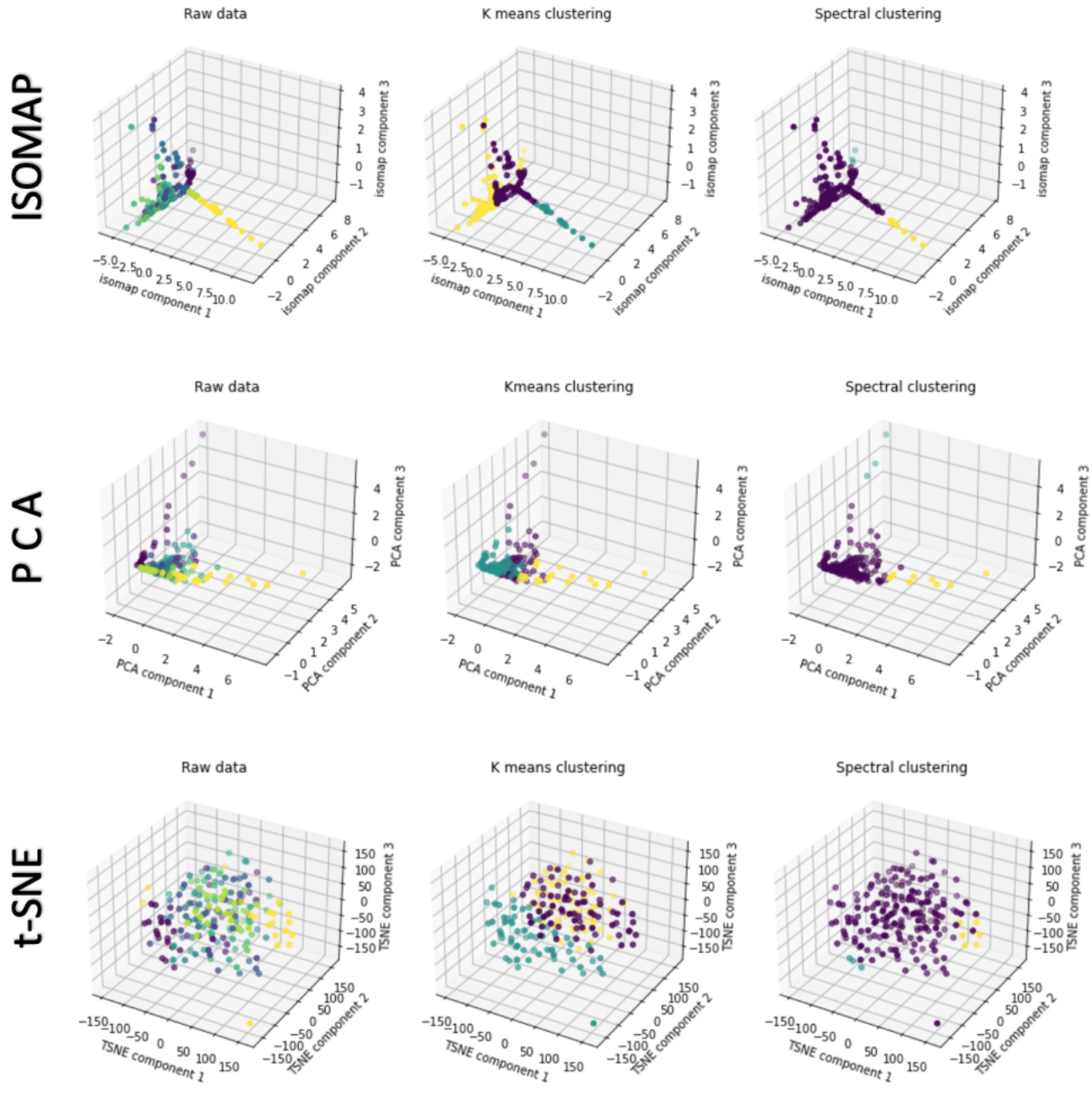


Figure 5: The plots in the first, second and third row have the first 3 components of Isomap, PCA and t-SNE as axis. The left most column is the raw data i.e. the colors indicate the original geographical labelling which came with the data. The second column shows each of Isomap, PCA and t-SNE data grouped into 3 clusters using Kmeans. The third column represents clustering of the data using Spectral clustering method. Visually, in the raw data (first column), most of the data appears to be close to each other (in each of the isomap, pca and t-SNE), except for a few outliers which are far from the group. Spectral clustering does a good job in separating these outliers, which are far from the group.

- Since there are many classifiers, we use Python's pipeline algorithm, which sequentially applies models to a given training dataset. It gives us scores of each of the classifiers on validation sets. It helps us to track the classifier and the set of parameters for which we have the best accuracy.

- We then use the classifier (and its specific parameter sets) which performed the best on validation sets and use that on the test data.

  **Note:** In binary classification, accuracy score predicts the fraction of correct classification. Mathermatically, its given by

  $$A(y, \hat{y}) = \frac{1}{n} \sum \mathbb{1}_{y,\hat{y}}$$

  where $A$ is the accuracy function, $y, \hat{y}$ are true and predicted classes of the data and $n$ is the number of samples to be classified.

Here are some of the algorithms that I used for this classification problem.

- **K Neighbors Classifier** - This is the most simplest form of classifier. For each point, it makes a list of K closest points (also referred to as neighbors). The class to which the point would belong is the class which majority of its neighbors belong to.

  In our analysis, we've used 4 values of K - 2,5,10 and 20. The accuracy scores for each of these are:

  | K | 2 | 5 | 10 | 20 |
  |---|---|---|---|---|
  | accuracy score | 0.71 | 0.69 | 0.70 | 0.72 |

  K Neighbors classifier with 20 neighbors was the 4th best classifier on training data.

- **Support Vector Machines** This classifier forms a hyperplane which separating the training data into classes. Typically, the larger the distance between the separating hyperplane and the nearest training data on either class, the better the classification.

  We used the following kernels for SVM SVC - 'linear', 'rbf'. Here are the accuracy scores:

  | Kernel | linear | rbf |
  |---|---|---|
  | accuracy score | 0.52 | 0.51 |

- **Logistic Classifier with stochastic gradient descent**: Stochastic gradient descent is a iterative technique useful in finding the model with the least loss score. In each step, the gradient of the loss function is calculated and the model

parameters moves towards the direction of minimizing the loss function. When the loss function is given as log loss, the classifier becomes logistic.

Here's a table corresponding to the parameters used and their respective accuracy scores.

| alpha (regularization term) | 0.01 | 0.1 | 1 |
|---|---|---|---|
| accuracy score | 0.51 | 0.51 | 0.51 |

- **Ridge Classifier**: Ridge converts the binary classification problem into -1,1 and applies regression problem. An L2 penalty term with regularisation $\alpha$ ensures that the ridge coefficients are close to 0.

  Here's a table corresponding to the parameters used and their respective accuracy scores.

| alpha (regularization term) | 0.01 | 0.1 | 1 | 10 |
|---|---|---|---|---|
| accuracy score | 0.65 | 0.65 | 0.65 | 0.65 |

- I've also used **Gradient boosting classifier, Random Forest Classifier** and have presented the results in Figure 6. These are decision tree based learning and classification methods.

  The Random Forest Classifier with depth = 2 and 100 estimators gave the best Classification results. Other tree based methods gave similar accuracy.

- **Neural Networks/MLP Classifier**: It uses hidden dense layers and partial derivatives to optimize for the loss function. Between each hidden layers, we used a combination of relu, logistic and sigmoid activation function. All resulting classifications had low accuracy, 0.55 being the maximum achieved accuracy. Possible reasons for such low accuracy could be the scarce amount of data.

### Results on test data

We use our best classifier from training data i.e. Random Forest Classifier and use that on the test data. This classifier had the accuracy score of 0.8 on test data.

## Importance and Conclusion

In this report we mostly worked on 2 problems. First we used dimension reduction and clustering methods to group similar data together. Our data was geographically labelled, but clustering still helped us gain new insights about data, insights which were not rooted in geographic proximity. We used spectral clustering to cluster the data into 3 groups and it grouped most of the data together, with a few different outliers (Figure 5). Looking back at these outliers, we noticed certain behaviors in the data which resulted

| | |
|---|---|
| 0.706944 | {'estimator': GradientBoostingClassifier(), 'estimator__max_depth': 2, 'estimator__n_estimators': 50} |
| 0.713194 | {'estimator': GradientBoostingClassifier(), 'estimator__max_depth': 2, 'estimator__n_estimators': 100} |
| 0.706944 | {'estimator': GradientBoostingClassifier(), 'estimator__max_depth': 2, 'estimator__n_estimators': 200} |
| 0.714583 | {'estimator': GradientBoostingClassifier(), 'estimator__max_depth': 5, 'estimator__n_estimators': 50} |
| 0.700694 | {'estimator': GradientBoostingClassifier(), 'estimator__max_depth': 5, 'estimator__n_estimators': 100} |
| 0.701389 | {'estimator': GradientBoostingClassifier(), 'estimator__max_depth': 5, 'estimator__n_estimators': 200} |
| 0.698611 | {'estimator': GradientBoostingClassifier(), 'estimator__max_depth': None, 'estimator__n_estimators': 50} |
| 0.698611 | {'estimator': GradientBoostingClassifier(), 'estimator__max_depth': None, 'estimator__n_estimators': 100} |
| 0.698611 | {'estimator': GradientBoostingClassifier(), 'estimator__max_depth': None, 'estimator__n_estimators': 200} |
| 0.729861 | {'estimator': RandomForestClassifier(max_depth=2), 'estimator__max_depth': 2, 'estimator__n_estimators': 50} |
| 0.731944 | {'estimator': RandomForestClassifier(max_depth=2), 'estimator__max_depth': 2, 'estimator__n_estimators': 100} |
| 0.730556 | {'estimator': RandomForestClassifier(max_depth=2), 'estimator__max_depth': 2, 'estimator__n_estimators': 200} |
| 0.706944 | {'estimator': RandomForestClassifier(max_depth=2), 'estimator__max_depth': 5, 'estimator__n_estimators': 50} |
| 0.703472 | {'estimator': RandomForestClassifier(max_depth=2), 'estimator__max_depth': 5, 'estimator__n_estimators': 100} |
| 0.704861 | {'estimator': RandomForestClassifier(max_depth=2), 'estimator__max_depth': 5, 'estimator__n_estimators': 200} |
| 0.702083 | {'estimator': RandomForestClassifier(max_depth=2), 'estimator__max_depth': None, 'estimator__n_estimators': 50} |
| 0.700000 | {'estimator': RandomForestClassifier(max_depth=2), 'estimator__max_depth': None, 'estimator__n_estimators': 100} |
| 0.701389 | {'estimator': RandomForestClassifier(max_depth=2), 'estimator__max_depth': None, 'estimator__n_estimators': 200} |

Figure 6: Gradient Boosting classifier and Random Forest Classifier with different maximum depths and no of estimators.

in such outliers. We found that there were several instances of fires burning through large acres of area in Atlanta as compared to areas affected in other regions (Figure 4, right). In addition, looking at the other type of outlier in the spectral clustering plot, we found that it corresponds to the data from Southern area. Looking at the original data, we noticed that the number of incidents of fires were clearly much more than rest of the regions, making it an outlier. Thus, spectral clustering can help trace outliers in fire data. This could later help with understanding the reason behind these outliers. Often huge fire incidents are a result of some marked changes in climate, wind movements or human activities. Therefore it's important to understand the reasons behind frequent fire incidents or unusually large fire incidents, so that we could protect forest and surrounding human communities going further.

We also showed that data science algorithms could be used to a good measure to classify between human caused fires and lightning caused fires. This is very useful in classifying the unreported fire incidents in remote inaccessible forest areas and other locations which are hard to access and hence observe. In additon, such classification of data could help get better estimates of fire incidents and the areas affected. Such information could form the basis of many conservation efforts and policy making to ensure that human communites and ecosystems are safe from hazardous large fires.