

**Q2 Final Report**

By:

Travis Tran

Htin-Aung (Hiroki) Masudathaya

Xueru (Wilson) Xie

Eric Wang

3/9/2021

University of California, San Diego

DSC 180A

Dr. Aaron Fraenkel / Dr. Ilkay Altintas

## Table of Contents

<b>Background</b>	<b>3</b>
<b>Overview</b>	<b>3</b>
<b>Proposed Statement</b>	<b>4</b>
<b>Data</b>	<b>5</b>
<b>EDA</b>	<b>8</b>
<b>Clustering Model</b>	<b>15</b>
<b>Conclusion: Model Result</b>	<b>17</b>
<b>Conclusion: Model Discussion</b>	<b>19</b>
<b>Individual Contributions</b>	<b>19</b>
<b>References</b>	<b>33</b>

## **Q2 Final Report**

### **Background**

We are working with the WiFIRE team's data where their team is using emerging AI techniques to create an end-to-end cyberinfrastructure for real-time data which could be used for simulations, predictions, and visualizations of wildfire properties. The WiFIRE data was created with the intention that their data would be used to assist with the prediction of wildfire rate of spread and predictions in the future. The WiFIRE team created a synthetic data source called FastFuels which is what we will mainly be working with for this project.

### **Overview**

The Congressional Research Service released statistics on wildfires, revealing that 2020 alone had 59 wildfires that burned over 10 million acres (Congressional Research Service, 2021, p. 1). This devastation included properties, farmland, animals, endangered ecosystems, and even resulted in fatalities. With this statistic, it is also revealed that wildfire occurrences are becoming more common due to a variety of factors such as climate change. With that being said, the negative impacts of wildfires have been increasing, resulting in many resources being spent on wildfire damage control and rebuilding.

Although there is no sole cause of wildfires, fire science narrows down three main factors that influence the ignition of wildfires: weather/wind, landscape, and most importantly fuel. Fuel is defined to be the material that a fire can use to grow, so an example of fuel in a forest setting would be dead branches and leaves on the ground. Traditional fire-fighting methods limit the build up of fuels using prescribed fires, in which controlled fires are used to burn fuel, preventing a build up of fuel that a wildfire could use to grow. However, this technique is most effective

with knowledge and prediction of prime fuel build-up locations. As the fire seasons continue to last longer due to droughts, climate change, and a variety of other factors, it is important to pinpoint locations that are most flammable and use prescribed fires to limit the amount of fuel build up.

### **Proposed Statement**

Wildfires are fires that occur by accident or natural causes and are unintended and uncontrolled. After the 2018 wildfire season, one of the most devastating and deadly fire seasons, wildfires have been becoming more common even during the most unlikely times of the year. With more resources, and more importantly, lives lost to wildfires, the need for proactive fire fighting methods is more important than ever. One such technique is prescribed fires, which are fires purposely started and controlled to burn certain areas and their fuel (e.g. dry grass, dead leaves, broken branches and timber, etc.). However, it's difficult to pinpoint locations that require this treatment.

The San Diego Supercomputer Center WiFIRE lab has tackled this problem, utilizing land-survey data on fuel across the United States. This data is collected in two year increments and 30 x 30m areas, leaving a margin of error in the analysis done. A new synthetic data source, FastFuels, is currently being generated as 1 x 1 x 128 meter areas (currently generated on the western side of the United States).

In order to improve the development of proactive fire fighting methods, how can we predict and classify fuel types based on the FastFuels 3D surface area to volume (SAV), bulk-density, and elevation data values, using machine learning methods?

For us to be able to create our predictions, we will begin implementing the Fastfuels voxel data to try to create the labels with Machine Learning techniques. We will utilize clustering algorithms to make the most accurate labels.

## Data

We used the following datasets of [Landfire](#) and [Fastfuels](#) data to generate most of the data we used. They served as invaluable resources for us to build our clustering model. They allowed us to make important observations of the correlations between the variables.

**FastFuels** is a synthetic 3-dimensional data source which is currently being generated by the WiFIRE lab in the San Diego Supercomputer Center. The WiFIRE lab uses existing fuels and spatial data sources, such as fia plots, landfire labels, and landfire fuel type extrapolation, along with emerging AI techniques to model the 3-dimensional fuel data across the United States, although it has currently only plotted the Western portion of the United States, where wildfires are most prevalent. FastFuels interpolates from LandFire, which is another spatial data source that has surface type data for 30 meter by 30 meter areas (pixel) in the US. FastFuels' data is 1 meter by 1 meter by 128 meter areas (voxel) data that contains the properties of the terrain such as bulk density, surface area to volume ratio, fuel moisture content, and elevation, which is all stored in a data dictionary.

Feature	Data structure	Empty value denotation	Data Description
Surface area to volume (SAV)	Dictionary with list of varying size depending on query bounds	{-1, 0, 1}	The proportion of surface area per unit volume of the object. Typically a good representation of how

			fast something will burn.
Moisture	Dictionary with list of varying size depending on query bounds	{0}	Fuel moisture content. Unknown units and unknown collection process.
Bulk Density	Dictionary with list of varying size depending on query bounds	{-1, 0}	Density of canopy and fuels. Typically a representation of how long something will burn for.
Elevation	Dictionary with list of 128 elements	None	Height above sea level, in meters.

Table 1. Data dictionary of features

Surface area to volume is our primary focus, as it represents the ratio of surface area for all vegetation/fuel in the area to the volume of vegetation in the area which helps identify fuel types based on how much potential flammable material is exposed in the voxel. Due to the lack of clarity regarding how moisture data is collected, we are excluding moisture content from the modeling process. Bulk density shows the density of canopy covering, and elevation shows how far above surface level each voxel is, in meters.

These are all valuable features in classifying vegetation and fuel types of these specific pixels. For example, one observation we noticed was that FastFuels' elevation and SAV data allowed us to identify the difference between taller trees and shorter trees, which is significant because taller trees are less flammable than shorter trees. Therefore, we can classify any pixel with tall trees to be slightly less flammable than pixels with short trees. One aspect of FastFuels that differentiates from existing spatial data sources is that it is generated in real-time which allows for much more accurate and up-to-date predictions. However, because of this, we limit the

scope of this project to match any regions with available FastFuels data, as we cannot confidently generalize our predictive modeling to regions without FastFuels data available.

With Fastfuels we also used the web app to obtain species data. The newest version of the **Fastfuels web API** allows us to gain species data of fuel types that was inaccessible through queries with the Python SDK. Linking species data with our existing fastfuels data provides more context and information on our modeling and predictive analysis. This data provides individual fuel units where the represented trees are all composed of real data from the FIA database. This web API contains the same scope of data as the SDK and provides information of 3D fuels across the contingent US.

The Fastfuels web app allows us to get information by creating a workspace, selecting a bounding box, and generating the fuels from that area. Bounding boxes can be selected through specified coordinates or through manually selecting a box on the map. The limitations of collecting this species data is that manual selections of bounding boxes and workspaces remain necessary to collect the data. Coordinate or LiDAR data can also be used to further improve accuracy and clarity when querying data.

The **Landfire** dataset and API consists of attributes pertaining to forest canopy data and fire behavior fuel model data from various years. Landfire data helps us obtain additional data complementing the Fastfuels we are already using. Landfire data may help us in improving our modeling and predictive output. The attributes contained in the Landfire dataset include canopy height, canopy cover, canopy bulk density, canopy base height, FBFM40, FBFM13, and existing vegetation type. Canopy data serves as an important source of information for fuel vegetation data. Dependencies used to access Landfire data include xarray, rioxarray, dask, rasterio,

geopandas, requests, s3fs==0.5.1 and fsspec==0.8.3. Versioning of s3fs and fsspec are specified for version compatibility.

FBFM references the different standard fire behavior fuel models differentiated by the types of fuel loading found in both dead and living fuel distinctions as well as their size and classifications. These categorized fuel models were evaluated from experts on fire and fuel from the US. FBFM 13 refers to the original fire classification and the more updated and more nuanced version in the FBFM 40.

Accessing Landfire data is similar to that of Fastfuels. Coordinates in latitude, longitude, and the radius of the bounding box (in meters) are used to specify the location. The type of data indicated by the data code abbreviation as well as the frame of the data is used to query the data.

The output data of queries from Landfire is a 2d array of the specified data type with the selected bounding box. As Landfire is 2d it represents vertical columns in the 1m squared area as opposed to the 3d array and voxel representations of Fastfuels. When combining these data sources, each data value from the Landfire model will represent the vertical column in the corresponding Fastfuels plot.

## **EDA**

The first step of any project is defining the data source and accessibility. FastFuels data is accessed through the FastFuels API. Data can be accessed for a specified area by entering its latitudinal and longitudinal coordinates as well as the radius of the area. Returned data comes in a data dictionary format, where each key represents a characteristic, and each value represents the characteristic's values in a raster data format with a size dependent on the query boundaries.



```
lat = 39.465  
lon = -106.768  
roi = fio.query(lon, lat, 500)
```

Figure 1. Query methods to access FastFuels API

FastFuels data can also be visualized in 3 dimensions with its query method as seen in Figure 2 below. This visualization utilizes the ipywidget known as ipyvtk, which works with server side pyvista render windows (Kitware).

```
roi.view('sav')
```

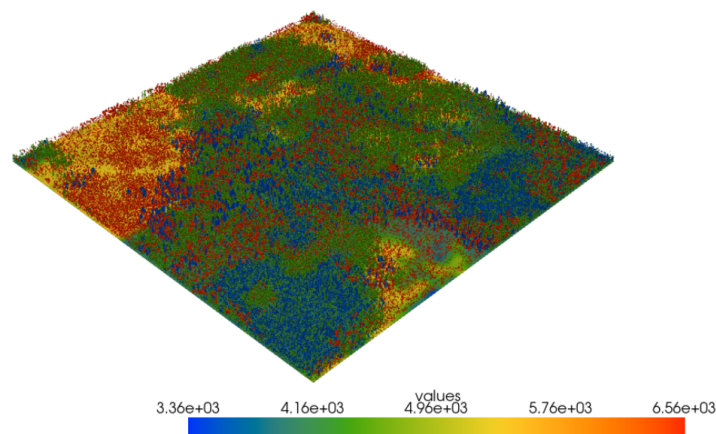


Figure 2. FastFuels query function to visualize its 3-D SAV data

Data can be highly detailed depending on the location of the query. In areas with tree/wildlife data there is voxelized representation of the vegetation (Figure 3). In areas where this does not exist, we see flat voxelized areas (Figure 4).

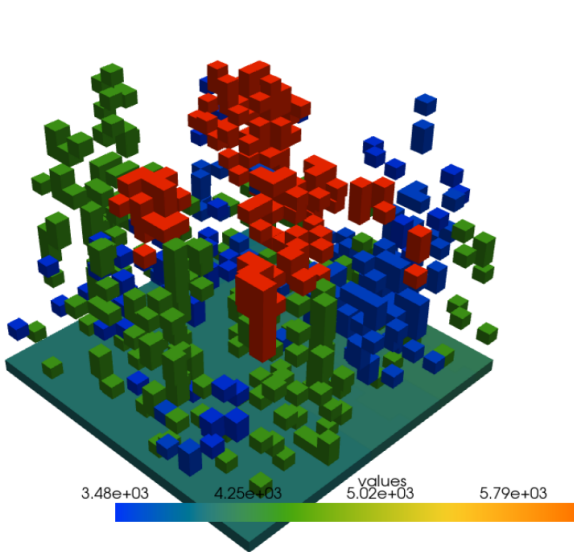


Figure 3. 3D voxel representations of vegetation

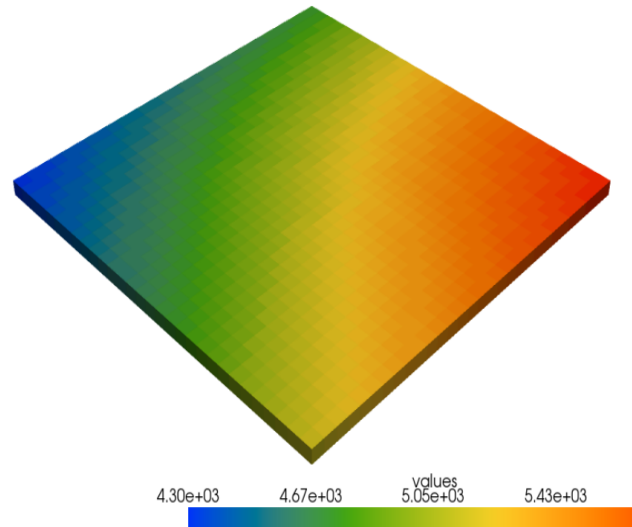


Figure 4. 2D flat representation of voxels

At this point of our EDA, we knew that SAV would be crucial to our analysis, so we decided to investigate other characteristics provided by the FastFuels data sources. In each voxel we have attributes including bulk density, SAV, moisture, fuel depth, and elevation by accessing the properties of the query as seen below in Figure 5.

```
print(roi.get_properties())
['bulk_density', 'sav', 'moisture', 'fuel_depth', 'elevation']
```

Figure 5. Voxel attributes accessed through query properties

Investigating bulk density, we discovered that the data was in a format very similar to the SAV values. Each value, excluding the air values, represents the density of the specific location,

calculated through a specific density formula derived from the standard “density = mass / volume” formula. As can be seen in Figure 6 below, there is typically the greatest density at base elevation on the ground.

```
[ 1.5764706 -1.      -1.      -1.      -1.      -1.
  -1.      -1.      -1.      -1.      -1.      -1.
  -1.      -1.      -1.      -1.      -1.      -1.
  -1.      -1.      -1.      -1.      -1.      1.082353
  1.1764706 -1.      0.83137256 1.0352942 -1.      -1.]
```

Figure 6. Example bulk density data array from the Bootleg Fire site in Oregon spanning a vertical column space

When viewing the distribution of SAV values, pixels with elements above ground level will appear with SAV's in higher elevations. Here, there are SAV values which most likely represent some form of vegetation or fuel, separated by air (Figure 7). Knowing this data formatting, we can begin designing an algorithm to classify pixel labels. Taking the average SAV of each pixel and plotting them on a histogram (Figure 8), we see a clear distinction between pixels with only ground-level SAV and pixels with multiple SAV values at different elevations.

```
[ 5.6156865e+03 -1.0000000e+00 -1.0000000e+00 -1.0000000e+00
 -1.0000000e+00 -1.0000000e+00 -1.0000000e+00 -1.0000000e+00
 6.5568628e+03 -1.0000000e+00 -1.0000000e+00 -1.0000000e+00
 -1.0000000e+00 -1.0000000e+00 -1.0000000e+00 -1.0000000e+00
 -1.0000000e+00 -1.0000000e+00 -1.0000000e+00 -1.0000000e+00
 -1.0000000e+00 -1.0000000e+00 -1.0000000e+00 -1.0000000e+00
 -1.0000000e+00 3.4196079e+03 -1.0000000e+00 -1.0000000e+00]
```

Figure 7. Sample SAV values

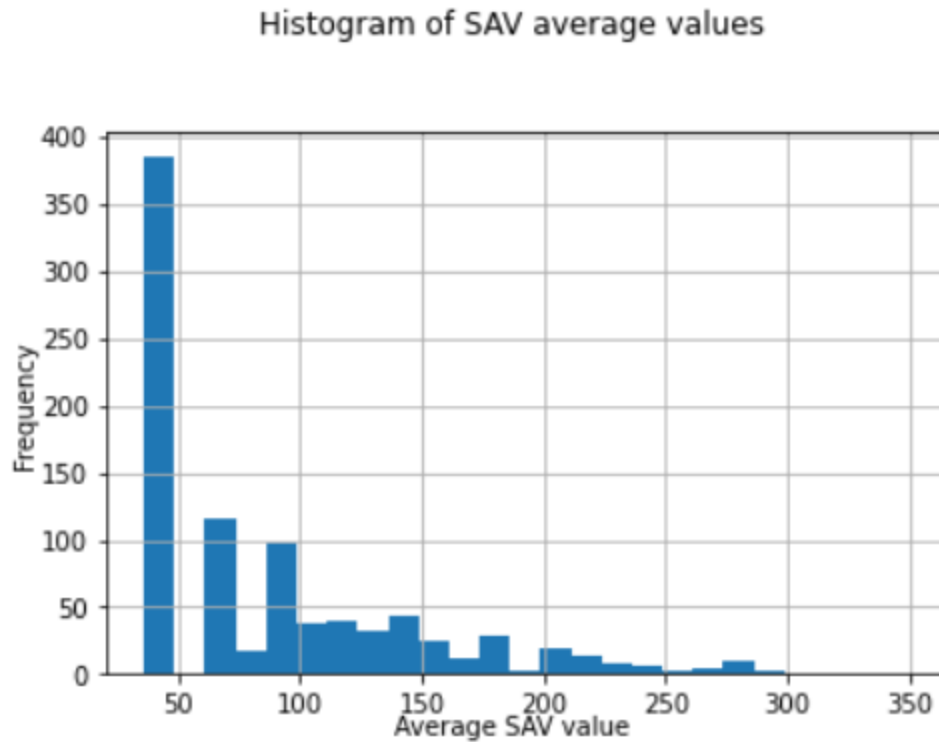


Figure 8. Histogram of average surface area to volume values in a 30 sq. meter area

Investigating moisture, raw moisture data appears to come exclusively in values of 0.2, 0, and 1 (Figure 9, 10). When consulting the WiFIRE team, we ran into additional confusion regarding how exactly moisture data was generated and even whether it has a purpose. This led us to avoid using moisture content in our analysis.

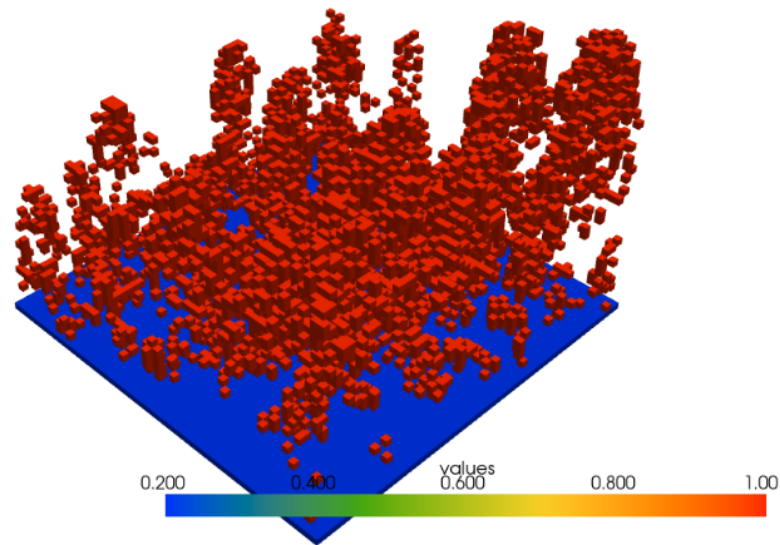


Figure 9. Example moisture content data visualization

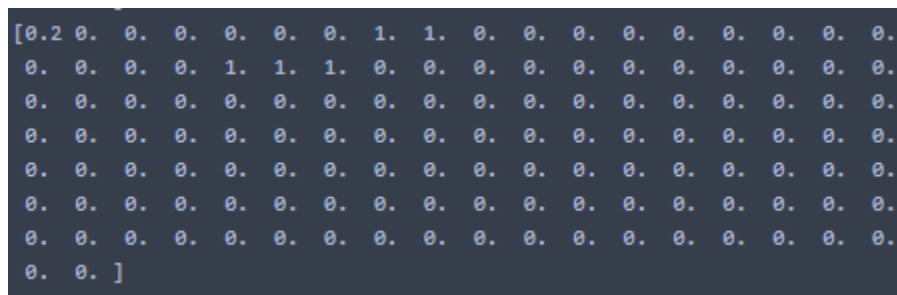


Figure 10. Example moisture content data array

Investigating fuel depth and elevation, all the locations that were queried returned 2 dimensional data, only ground level values for fuel depth and elevation. Two dimensional fuel depth data values are likely due to the nature of the feature itself. Since fuel depth represents how deep the fuel reaches, it becomes clear how a high fuel depth value indicates fuel existing deeper in the ground than areas with a low fuel depth value. Through this reasoning, we can see how fuel depth only needs to be two dimensional to provide additional context on the fuel content of a

location. Elevation follows similar reasoning, where each value represents how many meters above sea level the location is. Example queries can be found below in Figure 11.

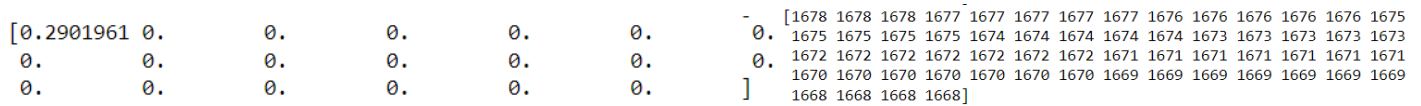


Figure 11. Example fuel depth (vertical column space) and elevation values (plane space)  
for a query of the Bootleg Fire site in Oregon

Now that we have a sufficient understanding of the composition of values provided for each characteristic, we can investigate points of interest in queries. In all of our queries, we see that air values in each query account for a majority of the data in the array, which takes up a lot of memory. In Figure 12, for example, empty values make up 98.5% of data for arrays; non-empty values take up 1.5% as seen below.

```
total_num = np.size(raw_sav_data)
print('nonzeros%: ', 100 * np.count_nonzero(raw_sav_data) / total_num)
print('zeros% : ', 100 * (1 - (np.count_nonzero(raw_sav_data) / total_num)))

nonzeros%:  1.517578125
zeros% :  98.482421875
```

Figure 12. Percentage of zero and nonzero values in an example query

After achieving a greater understanding of the data source, we proceeded with investigating the difference in information provided between the older two dimensional LandFire labels and the newer FastFuels three dimensional data with the increased resolution. In order to do this, we extracted a 30 square meter area and identified its fuel labeling according to the LANDFIRE Model 40 Attribute Data Dictionary ([landfire.gov](http://landfire.gov)). We then queried the exact same

location using the FastFuels API and assigned our own labels to each voxel within the 30 square meter area, and reported the composition of voxel labeling for that area. However, due to the range of SAV values that are potentially similar across each label type, we were only able to label differences between areas with trees and areas without trees. See Figure 13 below.

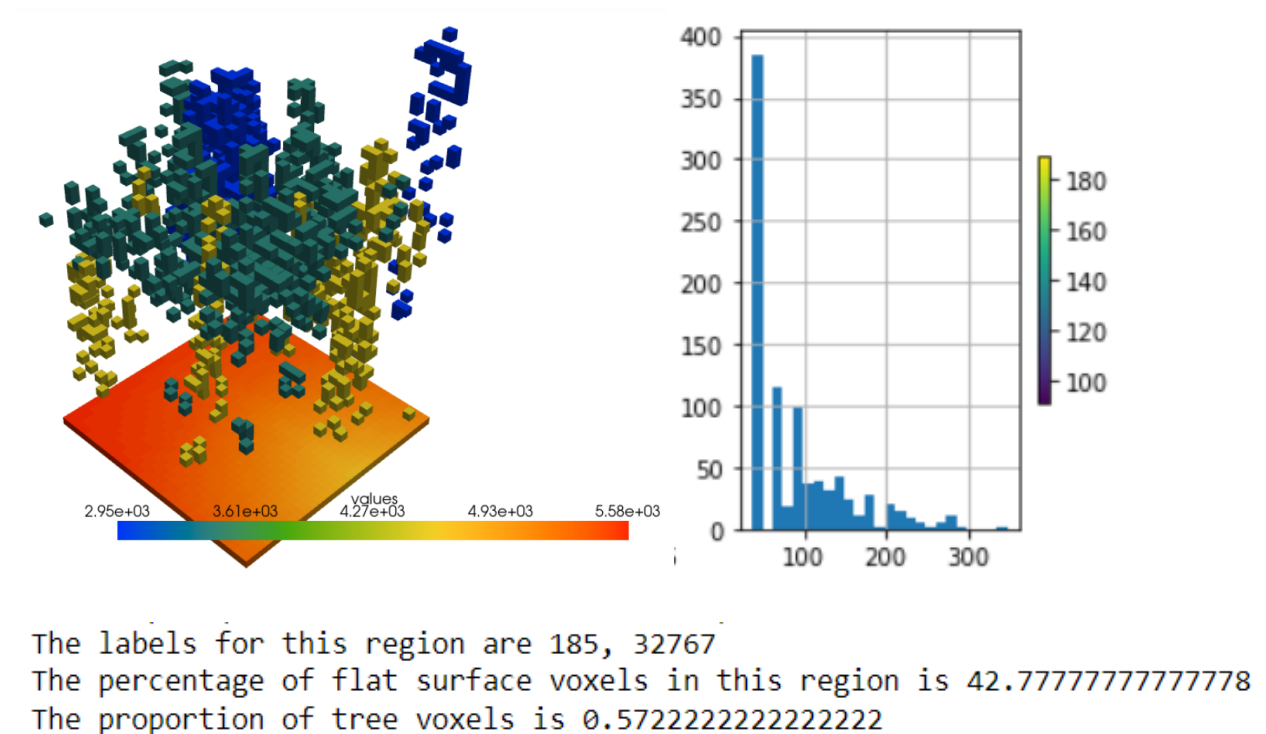


Figure 13. Output of a function that renders the three dimensional SAV view, plots a histogram of average SAV values, prints the LANDFIRE Model 40 fuel labels, creates tree/flat labels for the voxels, and prints the proportion of tree voxels in a specified location

## Clustering Model

At first, we were comparing the characteristics between two geographical features: forest and grassland. We wanted to find out if we could differentiate between the two fuel types using a clustering algorithm. To compare the characteristics between two geographical features, we were using the SAV values from the WiFIRE dataset. The general approach to this problem contains 4

steps. The first step would be to find training data. Since the goal of this model is to compare two features, then we have to find training data to train the model. Our training data would be extracted from the locations across the United States. For each training data, we will be taking one GPS coordinate on the map that is either from a forest or a grassland. We find about 25 data points for each geographical feature as training points. Next, for each of the inputted GPS coordinates, there will be a  $30 \times 30 \times 128$  SAV array.

Knowing that the  $30 \times 30 \times 128$  SAV array was too large for the clustering model, we immediately thought of doing dimensionality reduction to the data. Not every data point was meaningful in this array, therefore we had to extract 2 numbers from the array to perform the 2-dimensional k-means clustering. We wanted to use PCA to reduce the array dimension, then take two statistical values (mean & standard deviation) to perform the clustering model. However, when we tested this method it did not perform as well as we expected.

Then we decided to go back to our initial observation for ideas. The first number we chose was the average of the  $30 \times 30 \times 128$  SAV array because an average of a group of numbers is a good statistic. Secondly, we noticed that there was a clear distinction between the histogram of SAV values between a grass and a forest (Figure 14). The grass SAV values had a uniform distribution of a number approximately equalled to 5960, and the forest SAV values had a right skewed distribution. Therefore, our second number was the count of SAV values larger than 5960. Using this combination of average and count gave the clustering model the best performance.



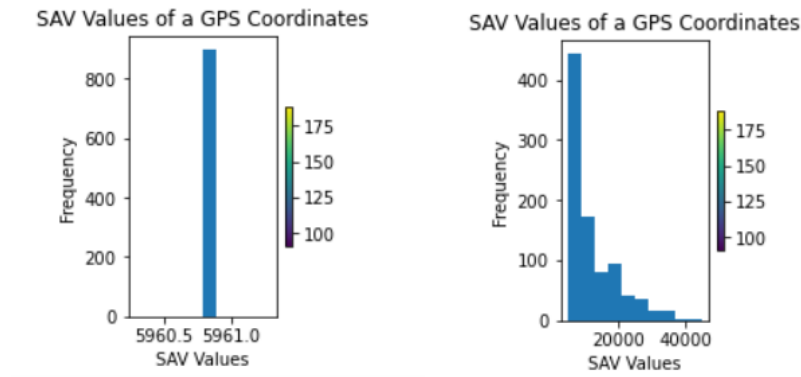


Figure 14. Histogram distribution of SAV values of a grass (left) and a forest (right)

## Conclusion

### Model Result

The result of the k-means clustering demonstrated a clear distinction between the SAV values of forest and grassland (Figure 15). Using this result from the clustering model, we were able to answer our original hypothesis question. The 50 training points for each fuel type were sufficient for the model because the result was promising. However, if we were to increase the training points, then there would be a better distinction between the two fuel types.

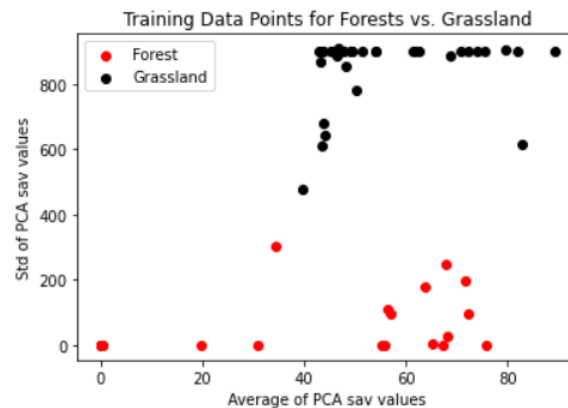


Figure 15. Clustering result of 50 forests and grassland data points

The clustering model would also be seen as a classification model because the training points were pre-labeled. In other words, we were using labeled data points to verify our accuracy in the clustering model. The result below showed a high accuracy for predicting the grassland (Table 2). However, the accuracy was not great for predicting the forest. We came to a conclusion that predicting forest was not accurate because the forest coordinates often contained grass. That means the forest coordinates might have had a lot of grass around the trees, leading the clustering model to believe that the coordinates had more grass than trees. Nevertheless, the clustering model was able to distinguish between coordinates with a lot of grass and a lot of forests.

Predict/Actual	Grass	Forest
Grass	43	43
Forest	7	15

Table 2. Confusion matrix of the clustering/classification model

After building the model, we wanted to expand it with more fuel types. We added fuel types shrub and grass-shrub to the model and the results were also promising (Figure 16). By adding more fuel types, the clustering model could distinguish coordinates that had many fuel types simultaneously in the area.

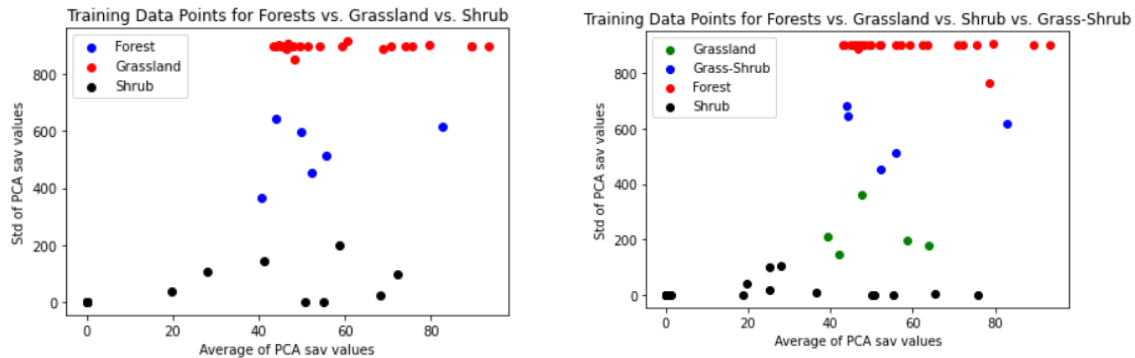


Figure 16. Clustering result of 25 data points for 3 fuel types (left) & 4 fuel types (right)

## Model Discussion

With the promising result of 2, 3, and 4 fuel types of clustering, we were hoping that future scientists could expand our clustering model as much as possible. In total, there were 12 fuel types available, and our model currently only supported up to 4 fuel types. Moreover, our clustering model could also expand itself to a formal classification model. An unlabeled GPS coordinate can be served as an input, and the classification model could predict its fuel type as the output.

## Individual Contributions

Eric Wang was in charge of keeping the team updated on revisions to code, notebooks, presentations, and any other artifacts in our project. Additionally, he researched fire science background and motivation for the project, created visualizations, and worked closely with Wilson to create the data collection process and changing model parameters.

Xueru Xie (Wilson) was responsible for developing the pipeline, which included creating and writing the data collection, investigation, hypothesis, data aggregation, clustering model, and the conclusion.


Travis Tran was responsible for mainly the report, website, and the sites this quarter along with providing help anywhere it was needed and making sure everyone was on the same page.

Htin-aung Masudathaya (Hiroki) was responsible for maintaining the Github repository and evaluating the PCA dimensionality reduction process. He also created the project structure that the team used to handle data and run notebooks and scripts.

## References

FastFuels, <https://fastfuels.io/>

LandFire, <https://landfire.gov/index.php>

Kitware. (n.d.). *Kitware/ipyvtklink*:  *minimalist ipywidget to interface with any python vtkrenderwindow*. GitHub. Retrieved December 14, 2021, from <https://github.com/Kitware/ipyvtklink>

Week 3, Week 4, Week 5. Retrieved from

<https://github.com/wilson5207/DSC-180A-Work-Samples/>

[https://github.com/hiroki-mtg/DSC180\\_Progress](https://github.com/hiroki-mtg/DSC180_Progress)

“Wildfire Statistics.” *Congressional Research Service*, <https://sgp.fas.org/crs/misc/IF10244.pdf>