INTRODUCTION

Disparate datasets can be joined together to form a more comprehensive, statistically relevant dataset that can be used to make and validate complex hypotheses that could not be explored or answered using exclusively the UFO sightings data. This in turn increases the value and volume and potentially the veracity and variety. In the past 2 weeks, our group has explored and added several features to the UFO dataset, making the resultant dataset an exhaustive source from which we explored, clustered, and drew conclusions from.

Datasets

The group explored several datasets of various MIME types including weather on day of sighting, proximity to a holiday, and even considered creating our own dataset of city images to create a neural network that can classify whether the image inputted is a densely populated or unpopulated area (binary classification) -- we would've fetched and classified the first google image result of each UFO sightings' location.

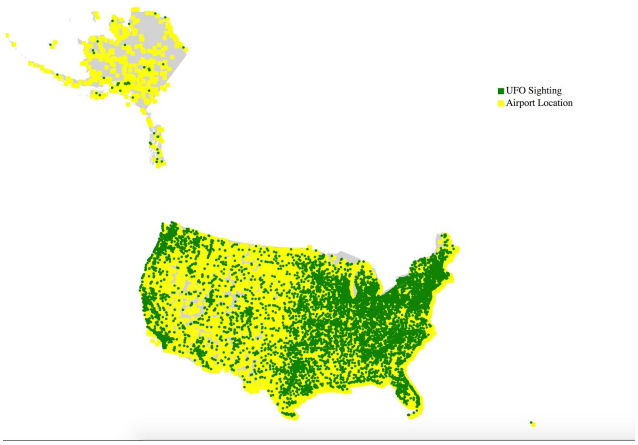
We were also able to extract shapes of about 1500 UFO sightings from their description.

The following are the datasets from which we appended features to the UFO dataset:

1) **Airports (txt/csv)**

6 features added: Closest distance/name to large, medium, and small airport for each UFO sighting

Close proximity to large U.S. airports is a fair reason why one could mistake a new or unseen aircraft (i.e. a blimp or new jet) for a UFO. Densely populated areas like Los Angeles or New York would also contain many medium and small sized airports, so we considered the distance to those airports as well. A rural area would have a large distance to its closest airport, but may have a small distance to its closest small airport -- because rural areas are more likely to be close to a small airport than a large airport, so we thought these 3 features would tell us a lot about what type of city a UFO sighting took place in -- an unintended consequence (deliberately used in this case) that adds substantial value.

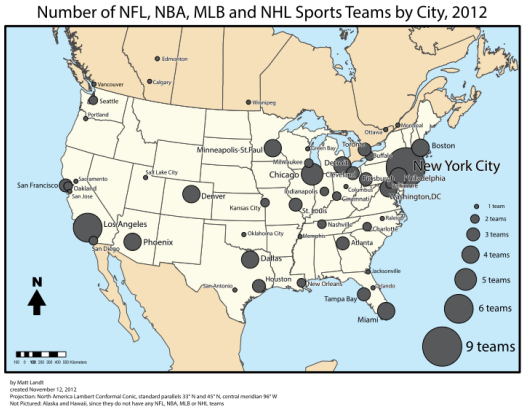


In the U.S. there are 172 large airports, 678 medium sized airports, and 13865 small sized airports.

This picture shows UFO sighting locations and airport coverage in the U.S. -- it is clear that there is an airport (large, medium, or small) near every single UFO sighting reported in the dataset. It is worth noting, however, that airports didn't become this prevalent until a few decades ago, so there are many sightings from early/mid 1900s that don't fall under this assumption, meaning they could warrant some exploration - but we'll wait till more features are added before exploring this. There are 858 closed airports that we considered exploring, but could not get around to; it would've been interesting to see if older UFO sightings had any correlation with proximity to closed airports.

2) **Sports (txt/csv)**

Context: The largest sports leagues in the U.S. by popularity are: NFL, MLB, NBA, NHL, MLS, and CFL. Major sports leagues, due to their interest in making money, are only located in major markets - with the larger markets having greater presence in various leagues. The following picture captures this idea:



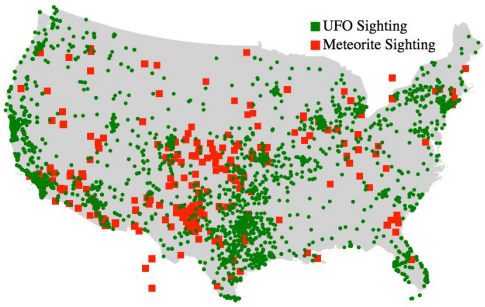
From the map it is clear that the count of the major sports teams (bigger circles = larger count) is indicative of population density of that area (another unintended consequence). This is a better indicator of what type of area a sighting is than the closest airport distance because there are less sports teams than airports (due to necessity of airports). Therefore, if the UFO sighting is far away from the closest sports team, it is likely that the sighting took place in a rural area.

5 features added: closest sports metro distance/name, closest sports metro population, closest metro major 4 sports leagues count, closest metro major 6 sports leagues count -- note that since CFL is in Canada the last feature is basically a count of major 5 sports leagues.

This dataset also inspired us to consider adding closest university and closest NCAA team, but major sports teams' metros added more value.

3) Meteorite (txt/csv)

Inspired by the SpaceX launch in December 2017 that got LA residents believing aliens are here and Wisconsin's meteor madness, we determined that a meteor falling can cause similar reactions in people believing they saw a UFO. We acquired a meteor dataset from Kaggle and joined with our original UFO sighting dataset. There was a limitation to the meteor dataset - only the year and location was specified, so we joined based on year and location instead of an ideal join on day, month, year and location. Location here was state - determined if the closest meteorite sighting in that year occurred in the same state as the UFO sighting.

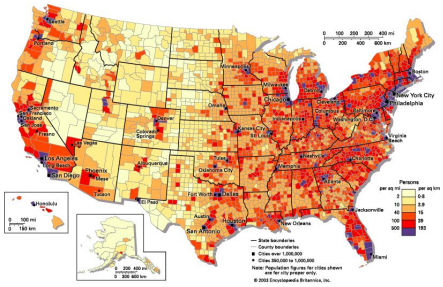


This map shows the overlay of joined UFO and meteorite sightings. Although not as dense as the other maps, there are still a lot of data points. We did consider that all meteor sightings in the past have not been recorded due to many meteors originating from ancient comets and meteor showers that are unpredictable in nature. On top of that, this may not be a complete dataset of ***all*** meteorites.

1 feature added: meteor sighting in year (boolean value)

We added a boolean flag due to the fact that we did not have granular details about the meteorite, else we would've added distance to meteorite from UFO sighting location as a feature as well.

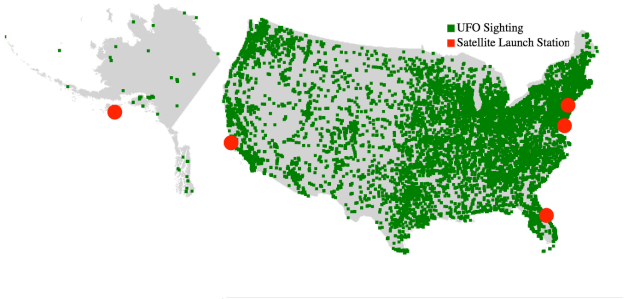
4) Population (application/pdf)



1 features added: population for sighting's state for given year

The sports features added earlier depend on populations from 2010 and we are aware that populations fluctuate quite a bit in the U.S. (i.e. recent increase in Austin, TX or decrease in Detroit). To account for this, we wanted to gather population of the city on the date of the sighting. We could not acquire a dataset with city census from early 1900s. The best we could find is the population of every state for every year since it was annexed till 2010 - so we went with this. Although it may seem a bit odd to add the state's population when we want to measure the city's population, it was the best we could find and our attempt to justify this uses the heatmap to the right - states with high population have many cities with high population and states with low population do **not** have many or any cities with high population, so this lack of granularity only affects small cities in states with large populations.

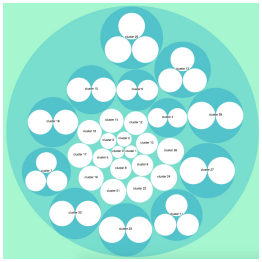
5) Satellite (text/csv) -- explored, not used



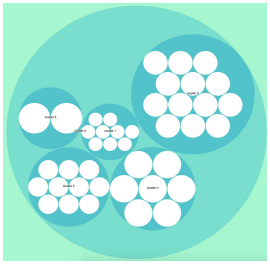
Also inspired the above-mentioned SpaceX incident, we explored a satellite dataset that detailed the launch dates and locations for satellite launches. Although the number of unique launch locations is very low, all are located in densely populated areas, which we hypothesized can explain some UFO sightings in the areas close to the launch locations. We ended up not using this dataset due to the fact that not many satellite launch dates coincided with UFO sighting report dates. If we had found a satellite trajectory or navy testing dataset, we may have had more luck.

Clustering

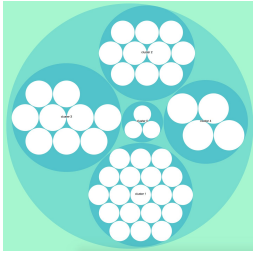
Cosine Similarity



Edit-Distance Similarity



Jaccard Distance Similarity



Circle packing clustering on a sample of 10 rows - produced using group-modified tika-similarity code

Similarity Functions	Sample Size	Number of clusters formed	Featurization	Value type
Cosine Similarity	10	28	Column wise min/max normalized features	Continuous
Edit-distance	10	6	Categorization	Discrete
Jaccard-Similarity	10	4	Categorization	Discrete

After appending new features to the dataset, we explored the above similarity metrics from tika-similarity. We conducted the experiment on a sample of 10 sightings - for the purpose of comparing/contrasting the 3 similarity functions.

Observations: Cosine similarity between almost all the files was really high (above 0.85) - this was due to the fact that the scales of the features were all of different magnitude. Large number of clusters were produced when computing edit-distance and jaccard similarity - which was due to most of the features being continuous rather than discrete values.

To address these issues, we attempted the following techniques -

1) Column-wise min-max **normalization:** Each column is normalized as: $\frac{df - df.min()}{df.max() - df.min()}$

2) Feature **Categorization:** Each column value is discretized based on below mentioned conditions:

Category	Population	Distance Metric(Miles)	CLASS
SMALL	< 200000	<50	SMALL = 1
MEDIUM	200000 - 600000	50 - 100	MEDIUM = 2
LARGE	> 600000	> 100	LARGE = 3

Cosine-Similarity with normalized feature vectors

Unique id	Large Airport Distance	Medium Airport Distance	Small Airport Distance	Distance to closest Metro	Metro Population	M4	M6	Population by State	Meteor Sightings	Cosine similarity
23587	0.00117463	0.00250376	0.000969536	0.000583419	0.073206442	0.222222222	0.1	0.211720188	0	1
53575	0.00140884	0.000655664	0.001830727	0.005862298	0.101512933	0.222222222	0.1	0.213183305	0	

Because of normalization, we were able to reduce the impact of large magnitude features - this way all features are given a fair chance to make an impact in whether two vectors are dissimilar.

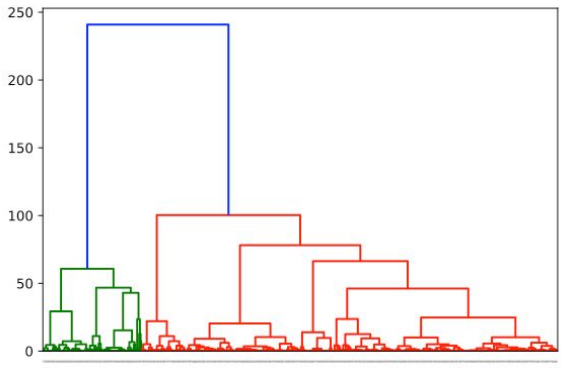
Edit Distance and Jaccard-Similarity with categorized variables

Unique id	Large Airport Distance	Medium Airport Distance	Small Airport Distance	Distance to closest Metro	Metro Population	Meteor Sightings	M4	M6	Edit-Distance	Jaccard Similarity
30094	1	1	1	1	3	0	2	2	1	0.78
60102	1	1	1	1	3	1	2	2		

Below is the output of edit-distance and jaccard similarity based on discretized features. These tuples share 7 out of 8 features in common, which is why the edit distance is 1 and jaccard similarity is $\frac{7}{8} \approx 0.78$

Edit distance seems to be a flawed metric in this sense: the edit distance between '11' and '12' is 1 and the edit distance between '11' and '110' is also 1. Categorizing variables overcomes this problem to a certain extent, but does not warrant edit distance to be the best distance metric for our appended dataset.

Using jaccard similarity with categorized variables makes much more sense than with continuous values. This is due to the fact that jaccard similarity relies on set intersection and union, meaning if two continuous values aren't exactly the same, they would not be considered the same - but if you convert those real numbers into discrete values, they can likelihood of intersection increases.



Using scipy, we were able to cluster normalized similarity scores on the **cosine** similarity matrix of a sample of 1000 rows. Unlike the clustering visualizations (circle-packing and dendrogram) found in tika-similarity that clustered on the similarity score rather than using an approach such as hierarchical clustering, we used hierarchical agglomerative analysis with defined proximity being Ward's method - the minimal increase of sum of squares. This approach place each row into **only one** cluster (unlike the approaches found in tika-similarity) that cluster on a (row1,row2) tuple. This means that using this approach, rows that are placed in a cluster are similar - meaning all tuple combinations of rows in that cluster would also be similar.

The results of this clustering are very interesting - the rural UFO sightings are densely located to the left most top-level green cluster. Within the green cluster you can see skinny clusters with small numbers of IDs per sub-cluster. After manual inspection of these rows, we hypothesize that UFO sightings within such tall, skinny clusters should be further explored. Furthermore, we did not consider the timestamp of the sighting when clustering. If we had, then we expect to see even skinnier and taller clusters that would reveal sightings that were similar and occurred around the same time This may, however, result in many more total clusters because an event like the SpaceX launch in December 2017 could warrant many UFO sighting reports from Los Angeles within the same time frame.

Conclusion

We added several features to the original dataset that resulted in a dataset that one can analyze using statistical approaches or manually inspect and make sense of -- allowing one to determine the veracity of a sighting. We used features that add a lot of value and volume, and by using an 'application' MIME type, we added variety to the dataset too. The resultant dataset allowed us to explore and cluster the data, learning that densely populated areas (especially near sports metros) result in more UFO sightings. We also were able to compare and contrast various similarity metrics and ultimately decided that cosine similarity was the best one to use given our normalized dataset.

All in all, we were able to explore various datasets, modify tika-similarity to work in the context of this project, and read a lot of engaging UFO reports.

Technologies Used

We used the following while packages/technologies for the assignment: scikit-learn, jupyter notebook, D3, tika, tika-similarity.

We used pandas to read the datasets and merge individual datasets to the original one based on the features that we discuss above for each dataset. The detailed information regarding the method of merging and the columns used for merging them is found in the readme of individual datasets in the /data folder in the git repository.

D3: Used D3 to visualize various UFO sightings, airports, metro cities, etc on US map. Also, visualized the clustered output using Dendrograms, Circle packing, etc.

We also tried visualizing the clusters in a more simple and readable way, we used indented tree representation of clustered output. We also made a pull request for the same in tika-similarity so that new visualizations can be added to it.

ABOUT TIKA:

Tika is easy to use. We could get the metadata, content related information just by calling the parser file. Also, the MIME type information was accurate. Alos, it is fast. It is great that we are able to use the tool for metadata, content, as well as language detection. Various API built upon Tika (eg. vision API) makes it a more certified

ADDITIONAL NOTES:

1. Images can be found in respective dataset folders and also in the google drive. "https://drive.google.com/drive/folders/1FAx8p1Nzzvev7BOH2dsFubH2JLsuA4zS?usp=sharing"