Michigan Neighbourhood Segmentation

Md Asif Shahjalal
M.S.E in Computer Engineering
University of Michigan-Dearborn

Abstract- This paper describes the techniques used to provide neighbourhood recommendations to people with the help of machine learning clustering algorithms. One such algorithm used in this project is the k-means clustering. The basic idea behind this paper, is to firstly create a tool that helps us extract the zip-codes given a location within Michigan. Once the zip-codes and its corresponding latitudes & longitudes are available, we had to identify the venues popular in those areas. Associating the features/amenities for the various locations gives us our dataset and with this data we can run the clustering algorithm to find what cities in Michigan are similar to one another, thus providing us with a neighbourhood recommendation system. All the Jupyter notebooks, javascript files and csv files can be downloaded from the following github link:
https://github.com/OnlyRefat/neighbourhoodSegmentation

Keywords: *clustering, K-means, Foursquare API, zip-codes, latitudes & longitudes, elbow method, PCA.*

INTRODUCTION

We humans get very accustomed to the locality around our home such that we struggle when it comes to choosing a new location to stay in. The main reason for this difficulty is because we try to find a neighbourhood that is a close match to where we had lived previously and finding an identical neighbourhood is a major task by itself. Hence, this project focuses on this problem and gives us a recommendation system telling which other areas are very similar to the old neighbourhood so that it is easier to choose among the many different areas. Hence making his/her moving-in task less stressful.

The main objective here is to use FourSquare API to get the data of top venues in a particular area and for obtaining all the areas of a particular state (in this case Michigan only) we use the Google API tool to extract area zip-codes along with its corresponding latitudes and longitudes. We then send queries to FourSquare API so that we receive venues details as per the extracted zip-codes and hence generate our dataset. As the dataset would be quite huge, we perform a dimensionality reduction to reduce the time and storage space required. It becomes easier to visualize the data when reduced to very low dimensions such as 2D or 3D. We then try to cluster our dataset using clustering techniques like K-means algorithm. Also an exploratory analysis on different segments will be conducted and discussed.

LITERATURE SURVEY

There are many technical papers written on clustering algorithms used for several different industrial applications. One such paper discusses that clustering algorithms are very popular in fields like image segmentation, market segmentation, image compression etc. [1] The clustering algorithms have also shown very good results in segmenting high crime rate areas or even in segmenting drivers (drunk, rash, careful) based on driving habit [2,3,4]. Particularly in paper [2] crime analysis was done by performing k-means clustering on crime dataset using rapid miner tool. The following steps were taken to cluster the crime data: take crime dataset, filter dataset according to requirement, open Rapid miner tool and read excel file of crime dataset, apply replace missing value operator, perform k-means clustering on resultant dataset, perform normalization operator on resultant dataset, perform plot view to get cluster and finally perform crime analysis on cluster formed. From the clustered results the authors of the paper were able to easily identify crime trend over a given range of years and hence used it to design precaution methods for future.

One another paper [5] used satellite images to analyse the various objects and quantity of resource availability in different land areas. Particularly in India where there is drastic change in the land use, water body, environment etc., due to population increases this study was helpful to segment the urban areas into land use and land cover regions & compare it in year wise manner to get the land use changes. The proposed technique was used to study the growth and status of urban sprawl in a particular Indian city called Salem. Initially pre-processing was done with mean shift filtering to reduce noise and for region smoothing. After that K-means clustering technique was applied to segment the images into different regions like vegetation area, building area and water body area then study the land use changes in Salem region during the period of 1973-2014.

One of the other important challenges while clustering the dataset using k-means is that of choosing the cluster number *k.* Several papers have been written on this very topic and how to efficiently choose the cluster number. One such paper [6] does the following for news content classification: News headlines can be used to categorize different news types. The appropriate type of news can make it easier for people to choose the particular topic they want. Similarity in a title was used to cluster the various news types. This paper [6] used TFIDF as document preprocessing method, K-Means as clustering method, and elbow method to optimize number of cluster. Purity method was applied to evaluate news title clustering as internal evaluation. SSE (Sum Square Error) of each cluster are calculated and compared to optimize number of cluster in the elbow method. The paper concludes that elbow method can be used to optimize number of clusters in K-Means clustering method.

All these above papers provided us with a good foundation of clustering algorithms and how it can be applied to various datasets to cluster the data as per our requirements.

METHODOLOGY

This section gives the detailed description of our work. The whole task was divided into two parts. They are (i) Data pre-processing, (ii) Model building. Figure 1 illustrates our overall workflow.

I. Data Preprocessing:

The main challenge in this section is to generate an organized data set, which is not available online. The data processing task consists of fetching data from two APIs. That is Google Geocode API and

FourSquare API. The following part contains the description of our Data preprocessing.

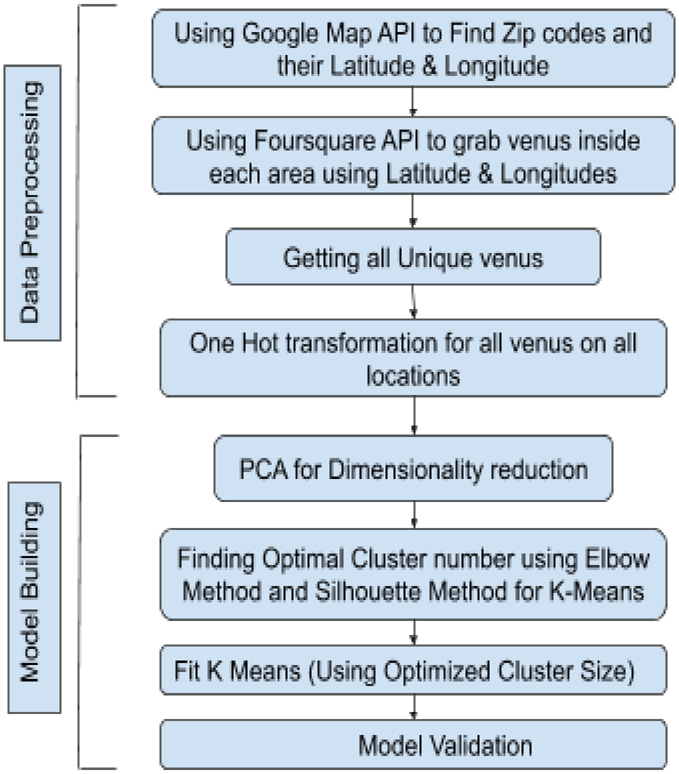
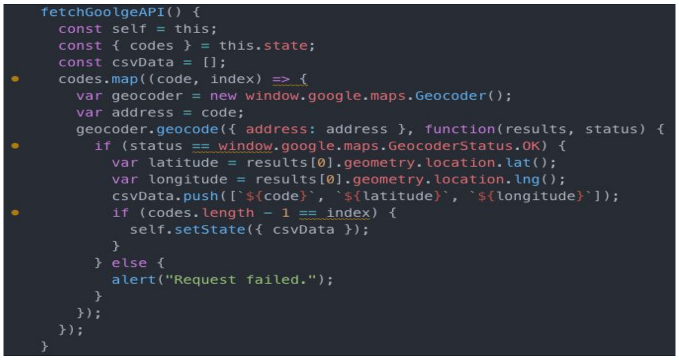

Data Preprocessing
Using Google Map API to Find Zip codes and their Latitude & Longitude
Using Foursquare API to grab venus inside each area using Latitude & Longitudes
Getting all Unique venus
One Hot transformation for all venus on all locations
Model Building
PCA for Dimensionality reduction
Finding Optimal Cluster number using Elbow Method and Silhouette Method for K-Means
Fit K Means (Using Optimized Cluster Size)
Model Validation

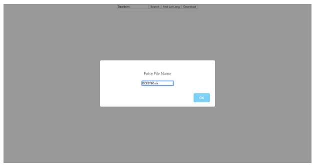
Figure 1. Neighbourhood segmentation work flow chart

(i) Use of Google map API to find zip codes and corresponding area latitudes and longitudes: here we came up with a tool that can be used to find out latitudes and longitudes of a given zip code. We used google Geocode Javascript API to find out the zip codes and their corresponding latitude and longitude. Finally we provided options for saving the data in a CSV file. Figure 2 shows the code snippet of fetching latitude and longitude. Once we have the zip codes, city names and their corresponding latitude & longitude we can download them in a csv file. Figure 3a and 3b shows the UI of our tool. From the downloaded CSV file, we use the latitudes and longitudes to fetch venues from FourSquare API.



```
fetchGoolgeAPI() {
  const self = this;
  const { codes } = this.state;
  const csvData = [];
  codes.map((code, index) => {
    var geocoder = new window.google.maps.Geocoder();
    var address = code;
    geocoder.geocode({ address: address }, function(results, status) {
      if (status == window.google.maps.GeocoderStatus.OK) {
        var latitude = results[0].geometry.location.lat();
        var longitude = results[0].geometry.location.lng();
        csvData.push([`${code}`, `${latitude}`, `${longitude}`]);
        if (codes.length - 1 == index) {
          self.setState({ csvData });
        }
      } else {
        alert("Request failed.");
      }
    });
  });
}
```

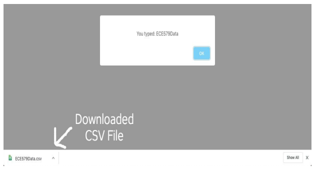
Figure 2. Fetching the latitudes & longitudes given the zip code using Google Geocode API



Enter File Name

ECE579Data

Figure 3a. Before downloading CSV



You typed: ECE579Data

Figure 3b. After Downloading CSV

(ii) Using FourSquare API to identify venues related to zipcode: We use FourSquare API to search for a particular place and filter by food, shops and outdoor places. Foursquare API (Python API)gives us access to a huge database of different venues from all around the world. It includes a rich variety of information such as the place's addresses, popularity, tips and photos. The access to the API is available for free and provides an easy setup. It also enables the user to search for places using a simple input box and this information is fetched from the Foursquare API and displayed.

Before getting started, we need to create account for accessing the APIs: in order to access the places information. To get the API key from FourSquare, we first need to create an account on their website, click on "Create a new app" and fill in the details. At the new page, we will be granted with a Client ID and a Client Secret, both needed in order to be able to perform request to the API. We start by implementing the search function. First, we need to figure out what kind of request we have to do in order to get the data we desire. We will do a simple HTTP request which will return a response. The URL is in the form:

https://api.foursquare.com/v2/venues/4ccc574a063a721e83df8d9a?client_id=E3IEHLNNW5CRC3WA2K5DLRZBNVP2LKNTIOD4GN0YHWUKUBA5&client_secret=FHVGA3OGTSWQ4ORRHKYN1VQTWRT4GPIWTMACTMBFLYH0CNWJ&v=20190421

The parameters are as following: **client_id**: the generated client ID from FourSquare. **client_secret**: the generated secret from Foursquare. **v**: the version of the API that we are using. The venue ID is retrieved from the search query executed in the mainview controller. When we push the page, the ID is given as a parameter and used inside the details controller. After reading the ID, a query to the API will be executed. After the query has finished, we can parse the desired data out of the returned JSON object. Following this above query method, we collected top 50 venues within 2 miles radius of each of the zip-codes.

(iii) Getting all unique venues: However, all the zipcodes didn't have 50 venues around it's 2 miles of radius and also the venues weren't similar enough. Hence we separated all the unique venues from our dataset and got 425 different categories of venues from the overall dataset.

(iv) One Hot transformation: A one hot encoding or transformation is a representation of categorical variables as binary vectors.This first requires that the categorical values be mapped to integer values. Then, each integer value is represented as a binary vector that is all zero values except the index of the integer, which is marked with a 1.

II. Model Building:

(i) PCA for dimensionality reduction: Principal Component Analysis is a dimension-reduction tool that can be used to reduce a large set of variables to a small set that still contains most of the information from the large set. Principal component analysis (PCA) is a mathematical procedure that transforms a number of (possibly) correlated variables into a (smaller) number of uncorrelated variables called principal components. The first principal component accounts for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible. Principal components analysis is similar to another multivariate procedure called Factor Analysis. Traditionally, principal component analysis is performed on a square symmetric matrix. It can be a SSCP matrix (pure sums of squares and cross products), Covariance matrix (scaled sums of squares and cross products), or Correlation matrix (sums of squares and cross products from standardized data). The analysis results for objects of type SSCP and covariance do not differ, since these objects only differ in a global scaling factor. A correlation matrix is used if the variances of individual variants differ much, or if the units of measurement of the individual variants differ.

The objectives of principal component analysis are it reduces attribute space from a larger number of variables to a smaller number of factors and as such is a "non-dependent" procedure (that is, it does not assume a dependent variable is specified). PCA is a dimensionality reduction or data compression method. The goal is dimension reduction and there is no guarantee that the dimensions are interpretable. To select a subset of variables from a larger set, based on which original variables have the highest correlations with the principal component.

(ii) Optimal Cluster Number: K-means is a simple unsupervised machine learning algorithm that groups a dataset into a user-specified number (*k*) of clusters. The algorithm is somewhat naive--it clusters the data into *k* clusters, even if *k* is not the right number of clusters to use. Therefore, when using k-means clustering, users need some way to determine whether they are using the right number of clusters.

One method to validate the number of clusters is the ***elbow method***. The idea of the elbow method is to run k-means clustering on the dataset for a range of values of *k* (say, *k* from 1 to 20), and for each value of *k* calculate the sum of squared errors (SSE). Then, plot a line chart of the SSE for each value of *k*. If the line chart looks like an arm, then the "elbow" on the arm is the value of *k* to be used.The idea is that we want a small SSE, but the SSE tends to decrease toward 0 as we increase *k* (the SSE is 0 when *k* is equal to the number of data points in the dataset, because then each data point is its own cluster, and there is no error between it and the center of its cluster). So our goal is to choose a small value of *k* that still has a low SSE, and the elbow usually represents where we start to have diminishing returns by increasing *k*.

The second method to find our optimal cluster number is silhouette analysis. ***Silhouette analysis*** is a way to measure how close each point in a cluster is to the points in its neighboring clusters. It is a neat way to find out the optimum value for k during k-means clustering. Silhouette values lies in the range of [-1, 1]. A value of +1 indicates that the sample is far away from its neighboring cluster and very close to the cluster it is assigned. Similarly, value of -1 indicates that the point is close to its neighboring cluster than to the cluster it is assigned. And, a value of 0 means its at the boundary of the distance between the two clusters. Value of +1 is ideal and -1 is least preferred. Hence, higher the value better is the cluster configuration.

(iii) Run K-means clustering algorithm:
Once we have the optimal number of clusters. We use python scikit learn API and k-means to fit our processed data. Our initial feature number was 425 and by using PCA we have reduced those features to 200. Using all those information we fit our model.

(iv) Model Validation: To validate the model, we followed a different approach. We seperated a portion of the dataset for testing. After getting the K-means model, we find the average distance of centroid to points for each cluster. And then we calculated the distance of each test data from the 16 clusters. Comparing the calculated distance with the average distance, we get an idea of which test point belongs to which cluster. So, we have manually calculated the clusters of the test points. Finally, we again used our model to predict the clusters of the test point. Comparing the manually calculated clusters for the test point with the predicted clusters by our model, we got an overview of the strength of our model.

RESULTS

In this section we describe our experimental work. Figure 4 shows the correlation between the 425 features of our dataset.

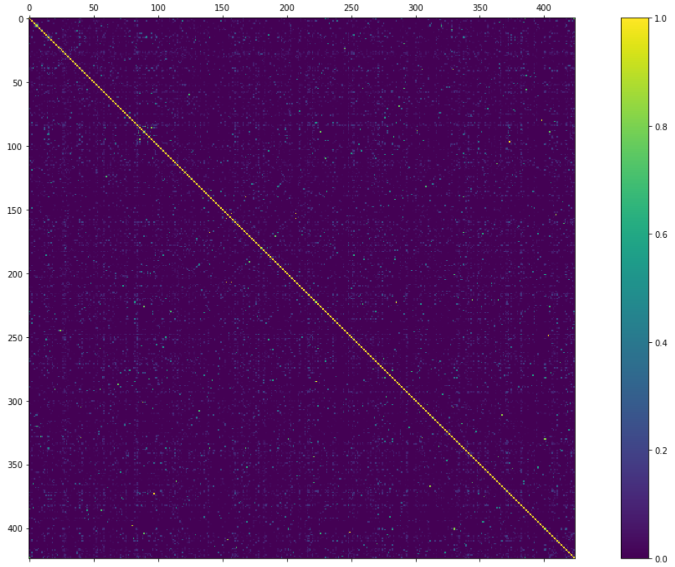


Figure 4. Feature correlation heatmap (425 features). Violet means no correlation, whereas yellow means high correlation

We filtered out all the zipcodes which have less than 5 venues. We filtered 598 zip codes from 904 zip codes. So our dataset was converted into a 598 / 425 matrices. All the features were converted to one hot encoding and then for each venue the frequency was calculated, like how frequent is a particular venue in a particular zipcode. However, to reduce the overfitting of the data we utilized PCA algorithm to reduce the dimensionality of our dataset. Figure 5 shows the cumulative explained variance vs number of principal components for our dataset. It is evident that at 200 principal components we retain almost 95% of the variance. So we chose 200 principal components for further analysis.

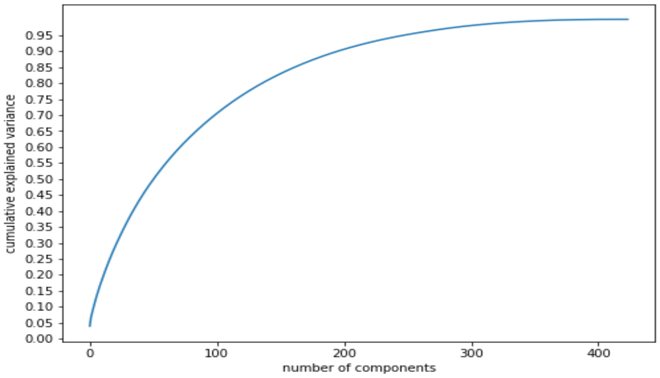


Figure 5. Selecting Principal Components

For selecting the number of cluster for our dataset, we used the elbow method and silhouette technique. Figure 6 clearly demonstrates that for 15 clusters we get the maximum silhouette score.

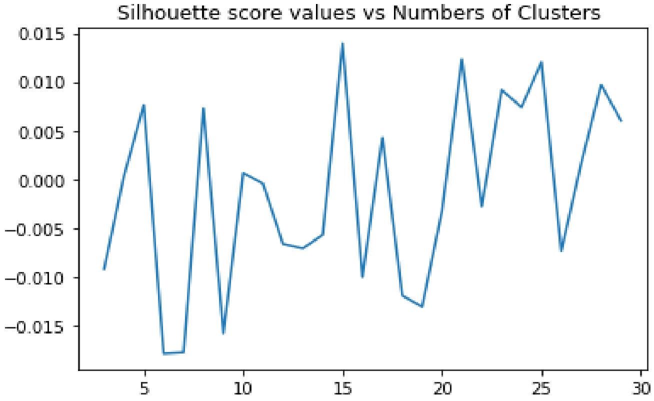


Figure 6. Silhouette Score confirms optimal Cluster Number 15

The Silhouette Coefficient is calculated using the mean intra-cluster distance and the mean nearest-cluster distance for each sample. Figure 7 shows the elbow method, in which case, we can't see a distinctive elbow, but we can still come to a conclusion that 15 clusters would be a reasonable choice.

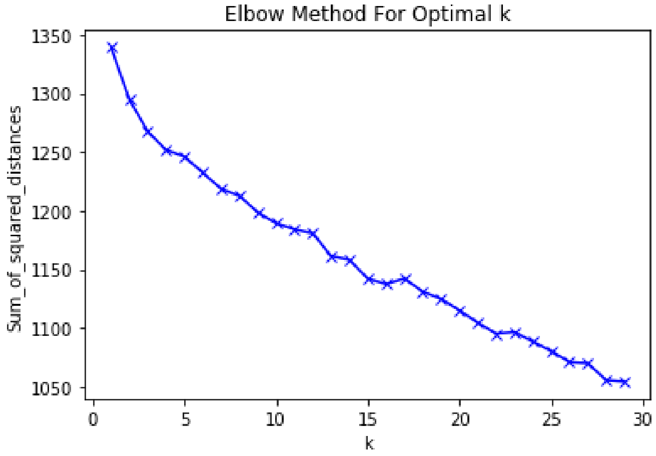


Figure 7. Elbow found at k =15 (K is number of Clusters)

Further we clustered our dataset using k-means clustering and calculated mean distance of each cluster points to their cluster centroids. Table 1 shows each of the clusters and the mean distance in each cluster points and centroids.

Table 1. Mean distance of Each Centroid

Cluster Number	Mean Distance from Centroid
0	1.4674097913956954
1	1.2290248668269355
2	1.187682170454529
3	1.331539408556313
4	1.472422636079081
5	1.4630449508813668
6	0.7141100449356619
7	0.24130453820208125
8	1.4862804220207164
9	1.1482065960616394
10	1.4058538902861646
11	0.8623951532178163
12	1.6498054446663795
13	1.2050351215452504
14	1.6358328059053404

We again utilized PCA to reduce our dataset into a 2 dimensional space to visualize the cluster data. Figure 8 shows the visualization in 2-D space.

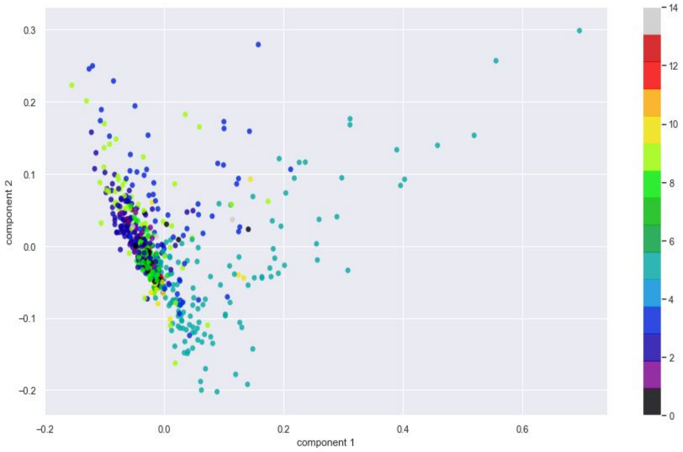
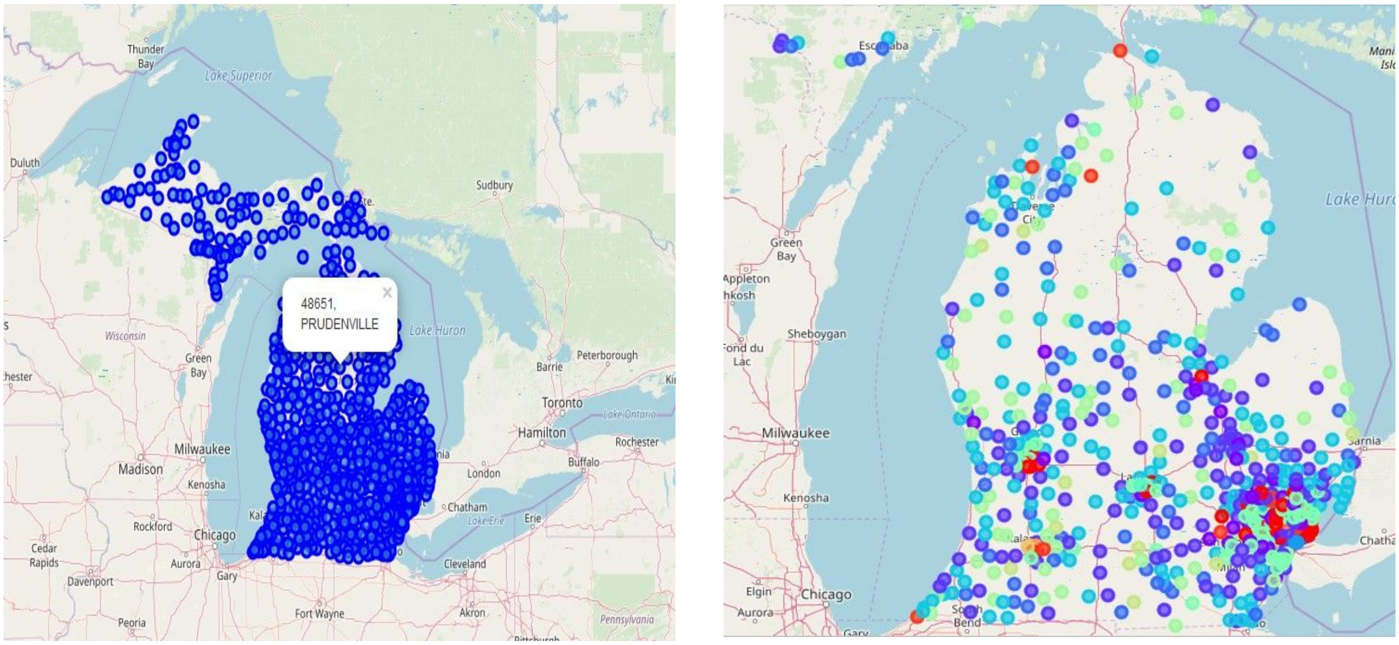


Figure 8. Visualizing the data in 2-D space using PCA

We plotted the clusters in actual map using folium library available in python. Figure 9 shows the Michigan map with all the available zip codes before and after the clustering.

We tested on some data points which we did not use for clustering and calculated the distance from each of the centers. We finally allocated them to the cluster for which it has the minimum distance. Table 2 shows the distance and assigned clusters.



(a) Before Clustering

(b) After Clustering

Figure 9. Michigan Zip codes with the assigned cluster label

We identified the top 5 venues in each zip code under each cluster. Figure 10 shows the top 5 venues for several zipcodes for 2 of the clusters.

Discussion

First the zip codes and their corresponding latitude and longitude of a city needed to be identified in a csv. The first challenge was to find the zip codes of a certain city. And we found this from http://federalgovernmentzipcodes.us/ website. Although they have latitude and longitude for the zipcodes in their database, but the problem was that latitude and longitude was truncated to 2 decimal places and many zip codes have same latitude and longitude. So, we had to use Google Geocode API for fetching the latitude and longitude for the corresponding zip codes. Fetching one request in the Geocode API at a time worked fine but when we tried to make batch request to Google Geocode API, we faced maximum request limit (10 per second). So, it had to programmatically handle the time between two API request. Finally, we came up with a web app that takes a city name as input and downloads a CSV with city name latitude and longitude. Figure 3 shows that.

Then we worked on collecting data from foursquare api. Foursquare api allows only 100 premium calls and 95000 regular calls per day. As we searched for 50 venues per zip code and there were 904 zip codes so we could not collect the data on a single attempt. We had to divide the data gathered from zipcode.csv into 4 segments and had to collect the data from 4 attempts in 4 different days. Then we had to merge the 4 dataset into a single 1 for further analysis. At first the queries were made for the venues within 500 meters of each zip codes. However, that did not return any venues for around 30% of the zipcodes and overall data gathered was very few. So, we had to increase the radius to 2 miles and collect the data again for a more sensible dataset.

We worked on clustering the data. One of the main challenges faced was to decide how to categorize the venues. After categorizing they needed to be converted to one hot encoding for clustering. The next challenge was to eliminate the zipcodes which did not have enough venues within 2 miles of radius. The most challenging one was to determine how many clusters should be used. The initial approach was to use the elbow method for determining the cluster number. However, the elbow method was not showing any definitive elbow. So, it was decided to use Silhouette score to find appropriate cluster number. Final challenge was to visualize the sample points from the clusters and PCA was used to reduce the dimension into a 2-D space for visualizing.

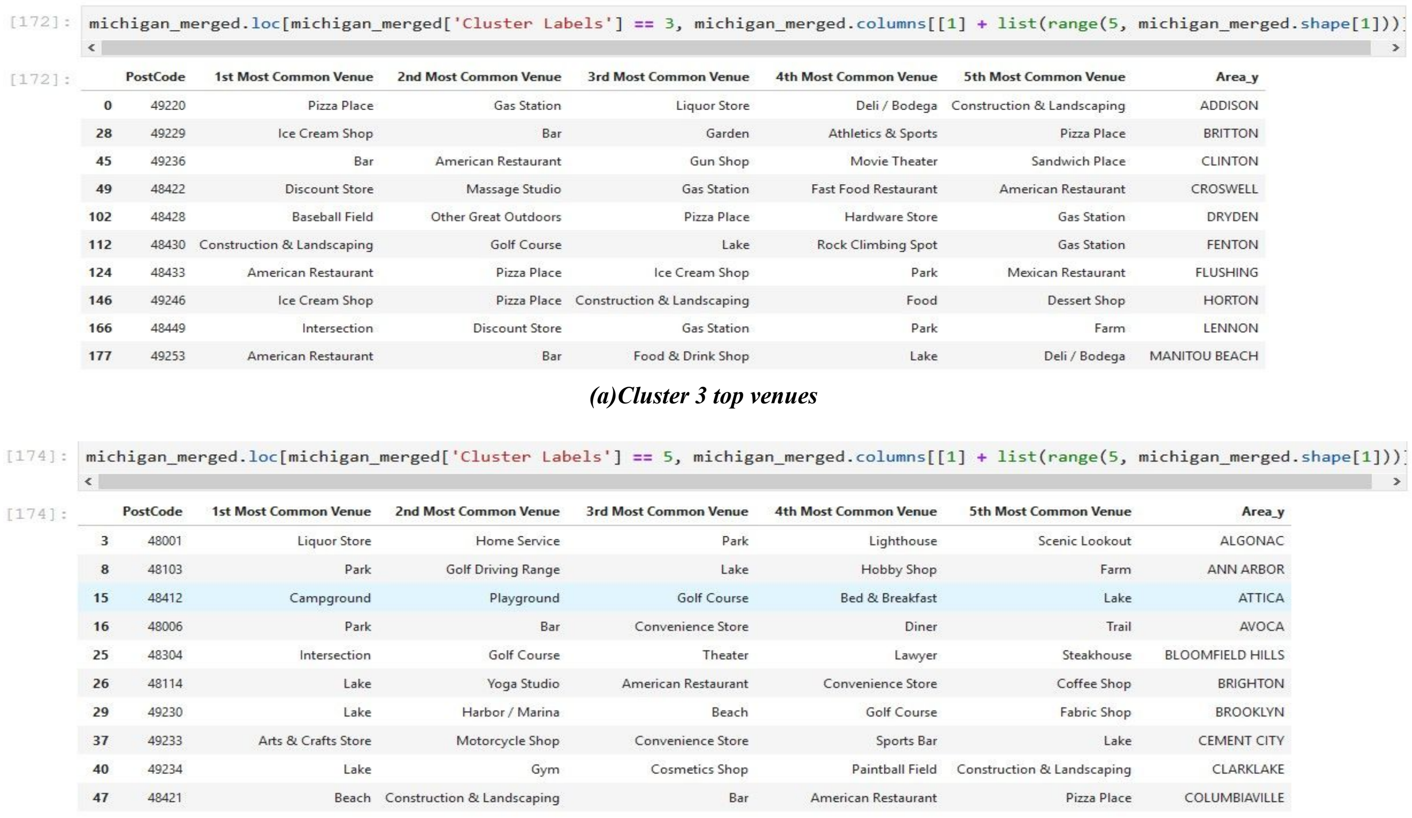
We calculated the mean cluster distances between each cluster point and their associated centoids. Then the new points distance to each cluster centroids were measured and they were assigned to the clusters with minimum distance.

CONCLUSION & FUTURE WORK

1. The project work was only done on the zip codes of Michigan State, which includes 598 zip codes each having 200 features even after dimensionality reduction with PCA. The problem is that we have a

Table 2. Distance of Each New Points to All the Centroids

N	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C 12	C13	C14	Label
1	1.41	1.49	1.26	1.21	2.46	1.29	2.52	2.46	1.29	1.26	1.35	2.51	1.81	2.24	1.28	3
2	1.51	1.59	1.39	1.28	2.40	1.33	2.58	2.43	1.40	1.42	1.33	2.53	1.98	2.41	1.50	3
3	1.84	1.89	1.84	1.94	2.80	1.95	2.792	2.50	1.72	1.98	2.01	2.63	2.26	2.69	1.97	8
4	2.05	2.15	2.05	2.07	2.83	2.06	2.94	2.82	1.87	2.08	2.11	2.68	2.42	2.73	2.09	8
5	1.78	2.00	1.89	2.02	2.66	2.03	2.90	2.86	1.86	2.00	2.16	2.77	2.36	2.84	2.03	0
6	2.02	2.11	2.01	2.08	2.73	2.09	3.00	2.86	1.89	2.11	2.17	2.70	2.47	2.85	2.08	8
7	1.31	1.39	1.17	1.31	2.45	1.24	2.41	2.40	1.19	1.31	1.36	2.39	1.87	2.30	1.32	2
8	1.25	1.28	0.99	1.05	2.41	1.03	2.50	2.38	1.11	1.08	1.31	2.37	1.84	2.25	1.26	2
9	1.80	1.85	1.66	1.62	2.66	1.50	2.73	2.62	1.71	1.64	1.76	2.73	2.19	2.55	1.76	5



```
[172]: michigan_merged.loc[michigan_merged['Cluster Labels'] == 3, michigan_merged.columns[[1] + list(range(5, michigan_merged.shape[1]))
```

[172]:

	PostCode	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	Area_y
0	49220	Pizza Place	Gas Station	Liquor Store	Deli / Bodega	Construction & Landscaping	ADDISON
28	49229	Ice Cream Shop	Bar	Garden	Athletics & Sports	Pizza Place	BRITTON
45	49236	Bar	American Restaurant	Gun Shop	Movie Theater	Sandwich Place	CLINTON
49	48422	Discount Store	Massage Studio	Gas Station	Fast Food Restaurant	American Restaurant	CROSWELL
102	48428	Baseball Field	Other Great Outdoors	Pizza Place	Hardware Store	Gas Station	DRYDEN
112	48430	Construction & Landscaping	Golf Course	Lake	Rock Climbing Spot	Gas Station	FENTON
124	48433	American Restaurant	Pizza Place	Ice Cream Shop	Park	Mexican Restaurant	FLUSHING
146	49246	Ice Cream Shop	Pizza Place	Construction & Landscaping	Food	Dessert Shop	HORTON
166	48449	Intersection	Discount Store	Gas Station	Park	Farm	LENNON
177	49253	American Restaurant	Bar	Food & Drink Shop	Lake	Deli / Bodega	MANITOU BEACH

(a)Cluster 3 top venues

```
[174]: michigan_merged.loc[michigan_merged['Cluster Labels'] == 5, michigan_merged.columns[[1] + list(range(5, michigan_merged.shape[1]))
```

[174]:

	PostCode	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	Area_y
3	48001	Liquor Store	Home Service	Park	Lighthouse	Scenic Lookout	ALGONAC
8	48103	Park	Golf Driving Range	Lake	Hobby Shop	Farm	ANN ARBOR
15	48412	Campground	Playground	Golf Course	Bed & Breakfast	Lake	ATTICA
16	48006	Park	Bar	Convenience Store	Diner	Trail	AVOCA
25	48304	Intersection	Golf Course	Theater	Lawyer	Steakhouse	BLOOMFIELD HILLS
26	48114	Lake	Yoga Studio	American Restaurant	Convenience Store	Coffee Shop	BRIGHTON
29	49230	Lake	Harbor / Marina	Beach	Golf Course	Fabric Shop	BROOKLYN
37	49233	Arts & Crafts Store	Motorcycle Shop	Convenience Store	Sports Bar	Lake	CEMENT CITY
40	49234	Lake	Gym	Cosmetics Shop	Paintball Field	Construction & Landscaping	CLARKLAKE
47	48421	Beach	Construction & Landscaping	Bar	American Restaurant	Pizza Place	COLUMBIAVILLE

(b)Cluster 5 top venues

Figure 10: Top 5 venues in each zip code for cluster 3 and 5

huge feature space but limited number of samples. We can collect data from entire United States, which will make our dataset well balanced.

2. When we started our work, we suffered for lack of organized dataset. We plan to make the dataset open source with our tool. So that others can use it for academic purpose and collaborate with this.

3. From figure 8, we can spot certain outlier in our data. In future we will try to filter out those outliers for more robust clustering.

4. There could be other clustering algorithms that can work better. In future, DBSCAN seems to be a good fit for our data.

5. We can sum everything, and convert to a neighbourhood recommendation APP.

REFERENCES

[1] Amandeep Kaur Mann and Navneet Kaur, “Survey Paper on Clustering”, International Journal of Science, Engineering and Technology Research (IJSETR) Volume 2, Issue 4, April 2013.

[2] Jyoti Agarwal, Renuka Nagpal and Rajni Sehgal, “Crime Analysis using K-Means Clustering”, International Journal of Computer Applications (0975 – 8887) Volume 83 – No 4, December 2013.

[3] Paper on Analyzing Crime Data: https://s3.amazonaws.com/academia.edu.documents/37598046/2115mlaij01.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1552707115&Signature=72AaCctyBPcn2%2FAnaVu8N9zbNlY%3D&response-content-disposition=inline%3B%20filename%3DUSING_MACHINE_LEARNING_ALGORITHMS_TO_ANA.pdf

[4] San Francisco 2014 Crime Data Clustering Application: http://www.sarahmakesmaps.com/blog/2015/1/clustering-san-francisco-crime-data

[5] K.Nithya, R.Shanmugasundaram and N.Santhiyakumari, “Study of Salem City Resource Management Using K-Means Clustering”, Proc. IEEE Conference on Emerging Devices and Smart Systems (ICEDSS 2017),Tamil Nadu, India.

[6] Dhendra Marutho, Sunarna Hendra Handaka, Ekaprana Wijaya, Muljono, “The Determination of Cluster Number at k-mean using Elbow Method and Purity Evaluation on Headline News ”, International Seminar on Application for Technology of Information and Communication (iSemantic), 2018.