Feasibility Study for Opening a Restaurant in Pune, India

IBM Data Science Capstone Project

Introduction

- Studying the current scenario and potential competitors is necessary while putting together a business plan
- Factors like location, type, cuisine, menu, ambience, pricing and services provided play an important part in determining the success of the business
- Project aims at analyzing the current landscape of various restaurants and attempts to come up with recommendations for starting/revamping restaurants
- Benefit new potential restaurant owners, existing owners amd allied businesses like food delivery services

Data Acquisition (Sources)

- Geospatial coordinates of Pune city and various localities
    - geopy
- Foursquare API
    - Foursquare - The Trusted Location Data & Intelligence Company
- Zomato API
    - https://developers.zomato.com/api

Data Cleaning and Transformation

- Data imputation for missing pincodes (zip postal codes) missing for few venues
- Prune duplicate records or overlapping records fetched
- Prune data points with missing location or user ratings
- Prune data points with user rating = 0
- Process text data in “highlights” and “cuisine” to add features for services provided
- Merge data from Foursquare and Zomato into single dataframe

Dataframe Used for analysis

```
df.info()
```

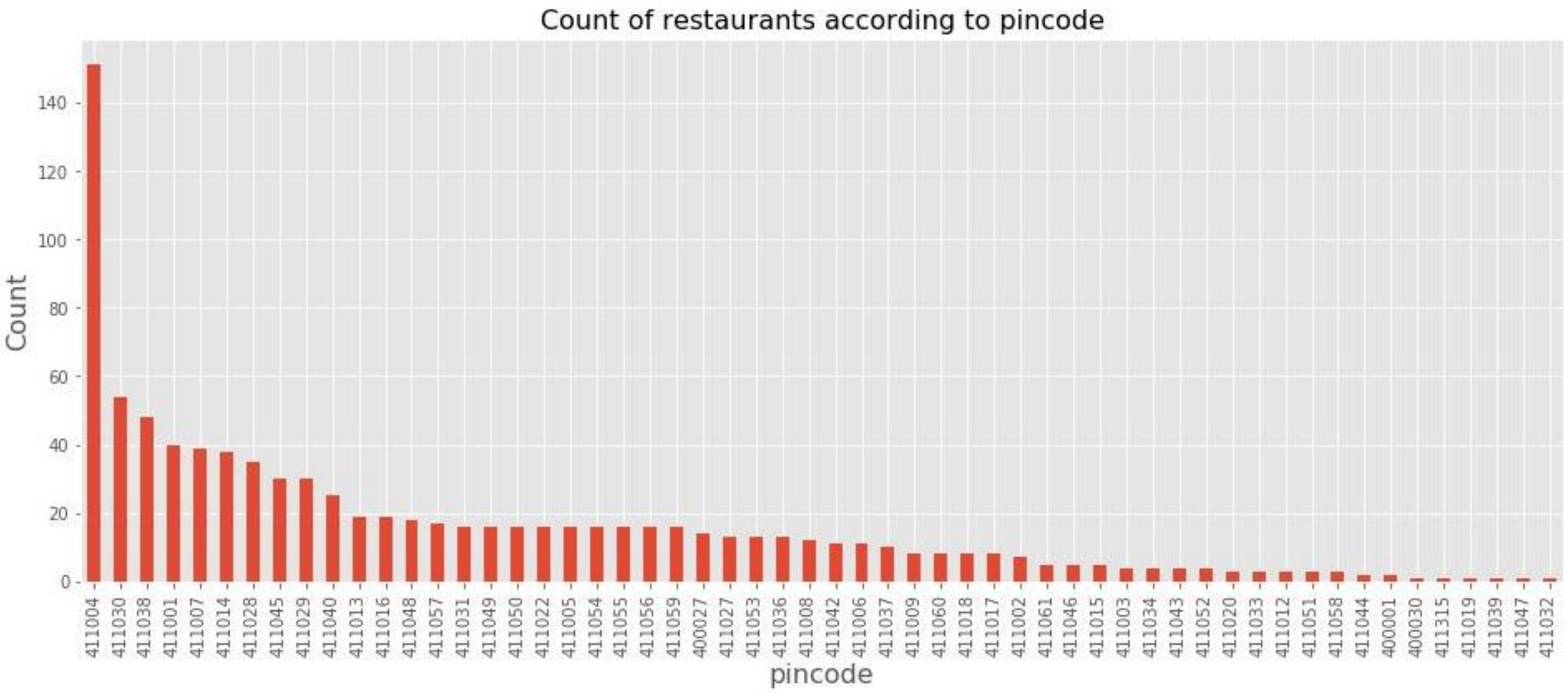
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 928 entries, 0 to 927
Data columns (total 15 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   name                  928 non-null    object
 1   lat                   928 non-null    float64
 2   lng                   928 non-null    float64
 3   pincode               928 non-null    object
 4   type                  928 non-null    object
 5   average_cost_for_two  928 non-null    int64
 6   price_range           928 non-null    int64
 7   aggregate_rating      928 non-null    float64
 8   votes                 928 non-null    int64
 9   cuisines              928 non-null    object
 10  highlights            928 non-null    object
 11  all_reviews_count     928 non-null    int64
 12  photo_count           928 non-null    int64
 13  has_table_booking     928 non-null    int64
 14  has_online_delivery   928 non-null    int64
dtypes: float64(3), int64(7), object(5)
memory usage: 116.0+ KB
```

Exploratory Data Analysis

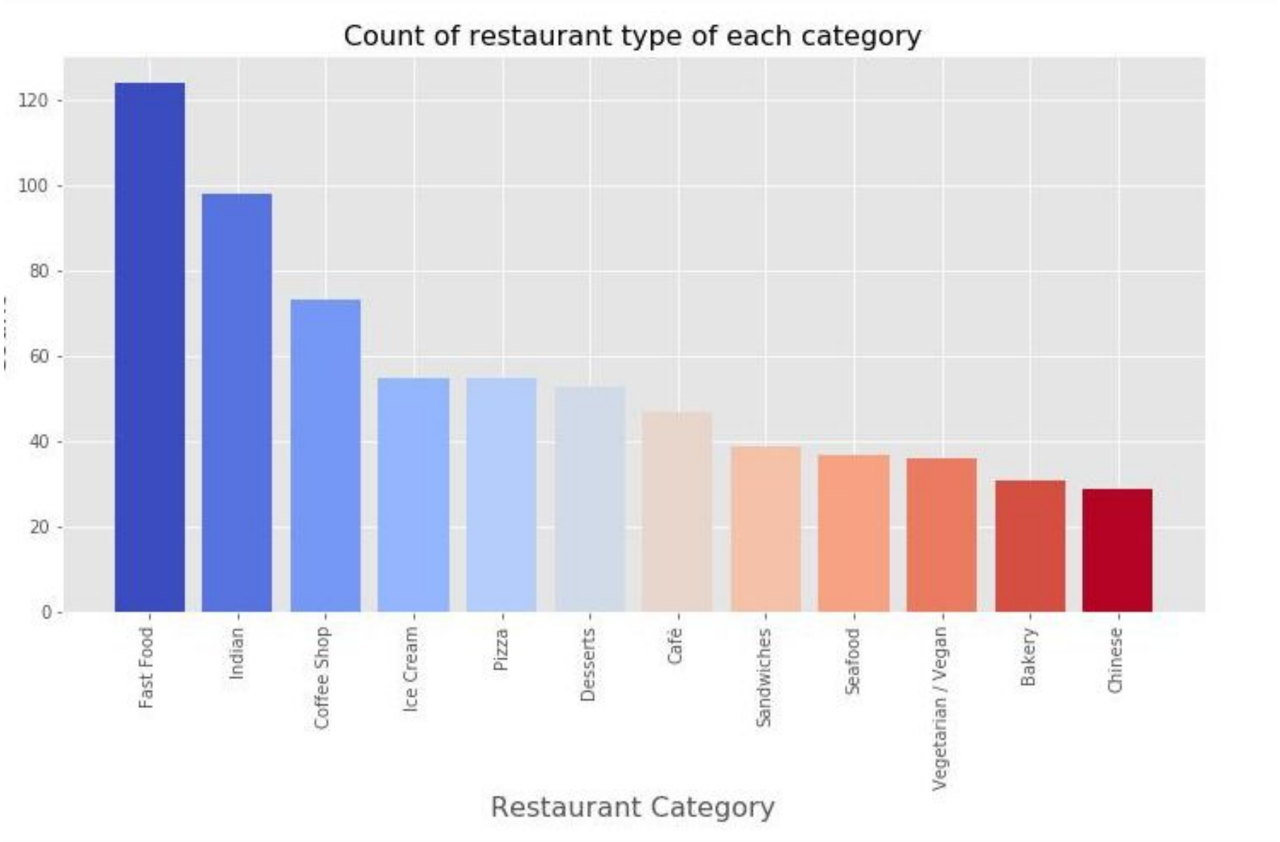
```
df.describe()
```

	lat	lng	average_cost_for_two	price_range	aggregate_rating	votes	all_reviews_count	photo_count	has_table_booking	has_o
count	928.000000	928.000000	928.000000	928.000000	928.000000	928.000000	928.000000	928.000000	928.000000	
mean	18.523253	73.851635	610.021552	1.743534	3.908297	1622.728448	362.492457	435.024784	0.082974	
std	0.028538	0.037079	420.742908	0.763912	0.616884	2172.220883	518.994746	981.523962	0.275992	
min	18.440685	73.766671	100.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	
25%	18.512350	73.833940	350.000000	1.000000	3.700000	262.000000	31.000000	46.000000	0.000000	
50%	18.518451	73.842713	500.000000	2.000000	4.000000	774.000000	198.000000	95.000000	0.000000	
75%	18.524296	73.872360	600.000000	2.000000	4.200000	2434.750000	356.750000	367.000000	0.000000	
max	18.671948	73.937133	2100.000000	4.000000	4.700000	30737.000000	2621.000000	7307.000000	1.000000	

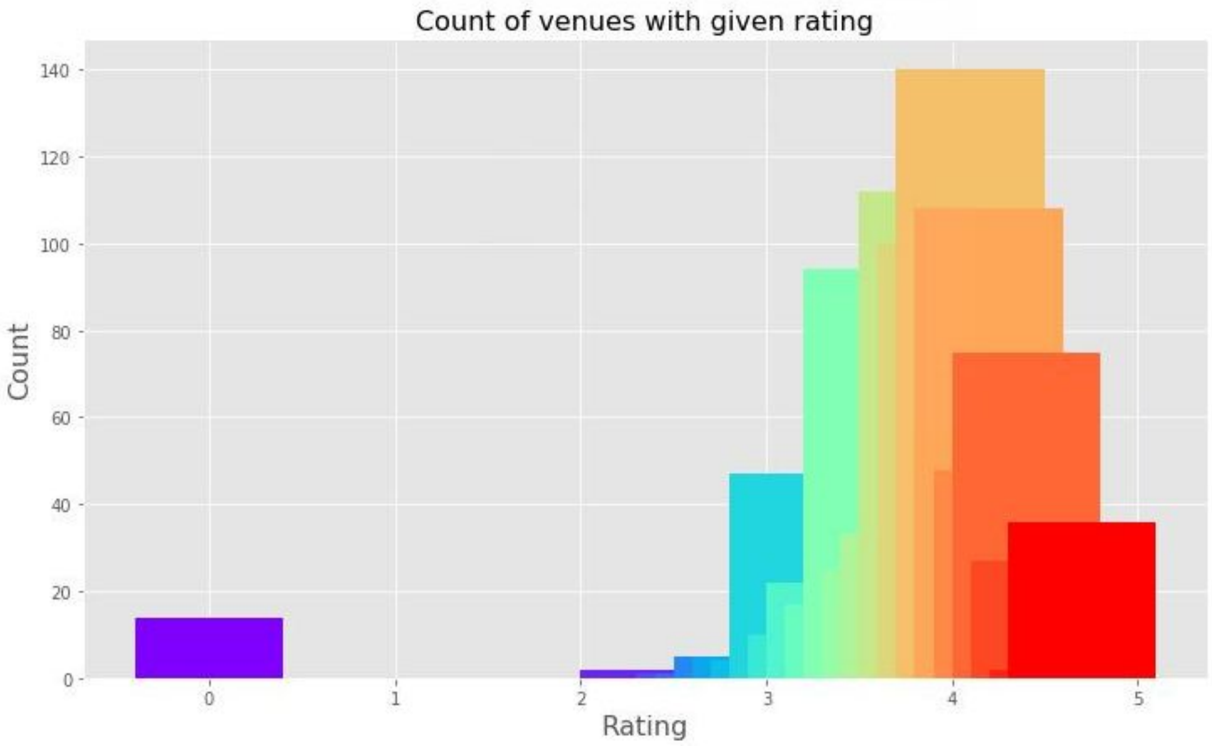
Restaurant Distribution based on pincodes



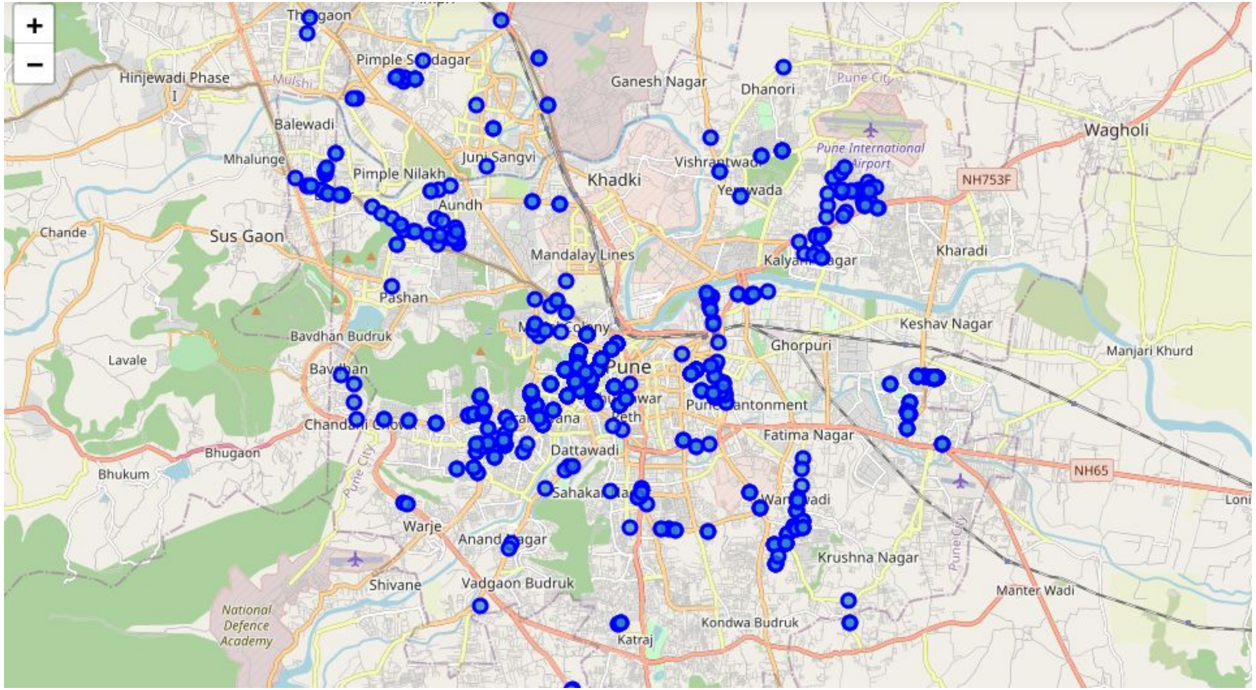
Distribution of Restaurants based on Category



Distribution of Restaurants based on Ratings



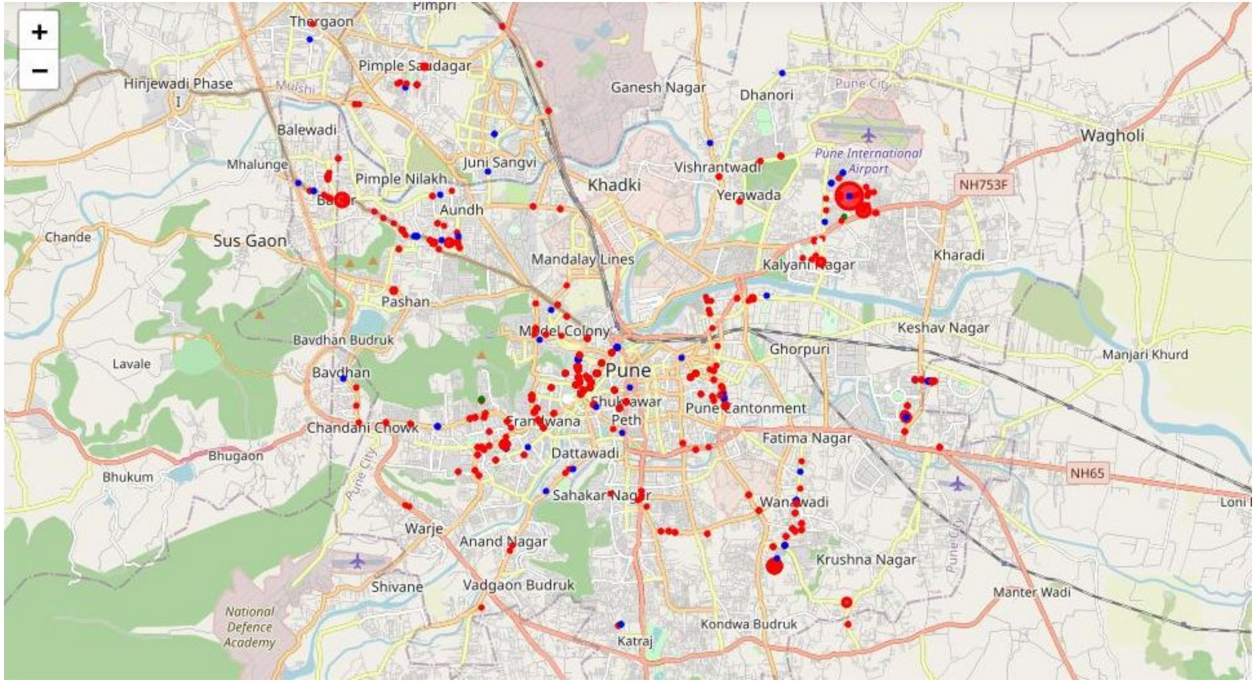
Geographical Distribution of Restaurants



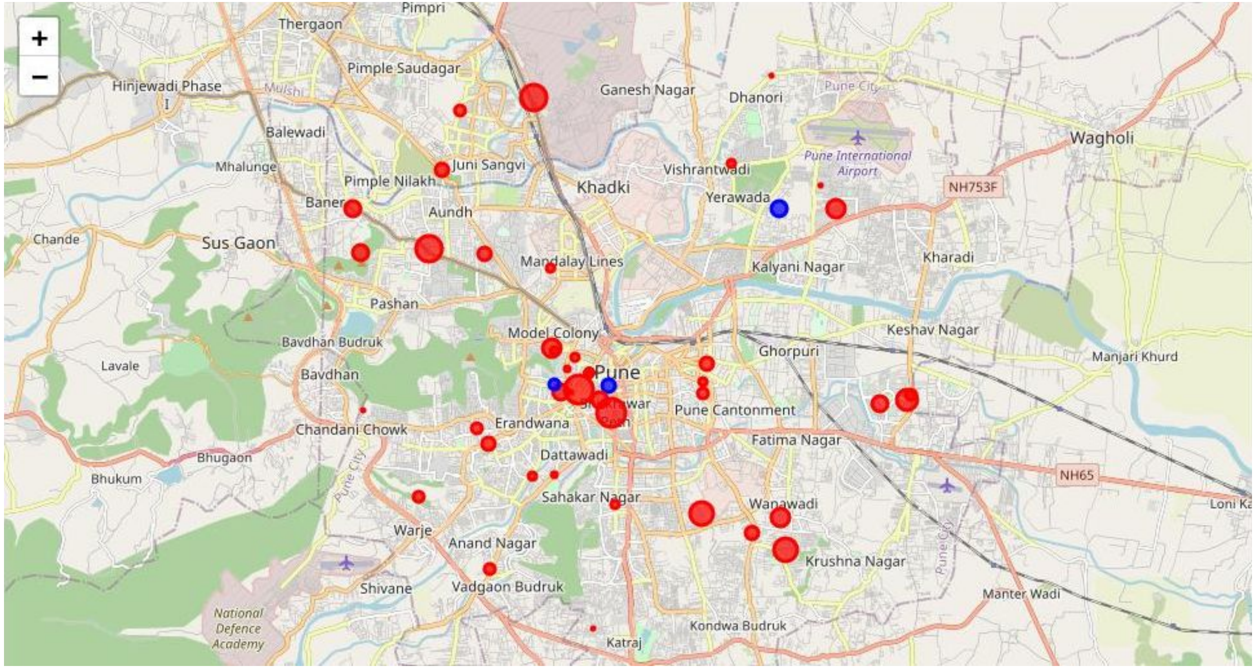
Geographical Representation based on Ratings

Red - Above Average

Blue - Below Average



Popularity of Restaurants by Pincode

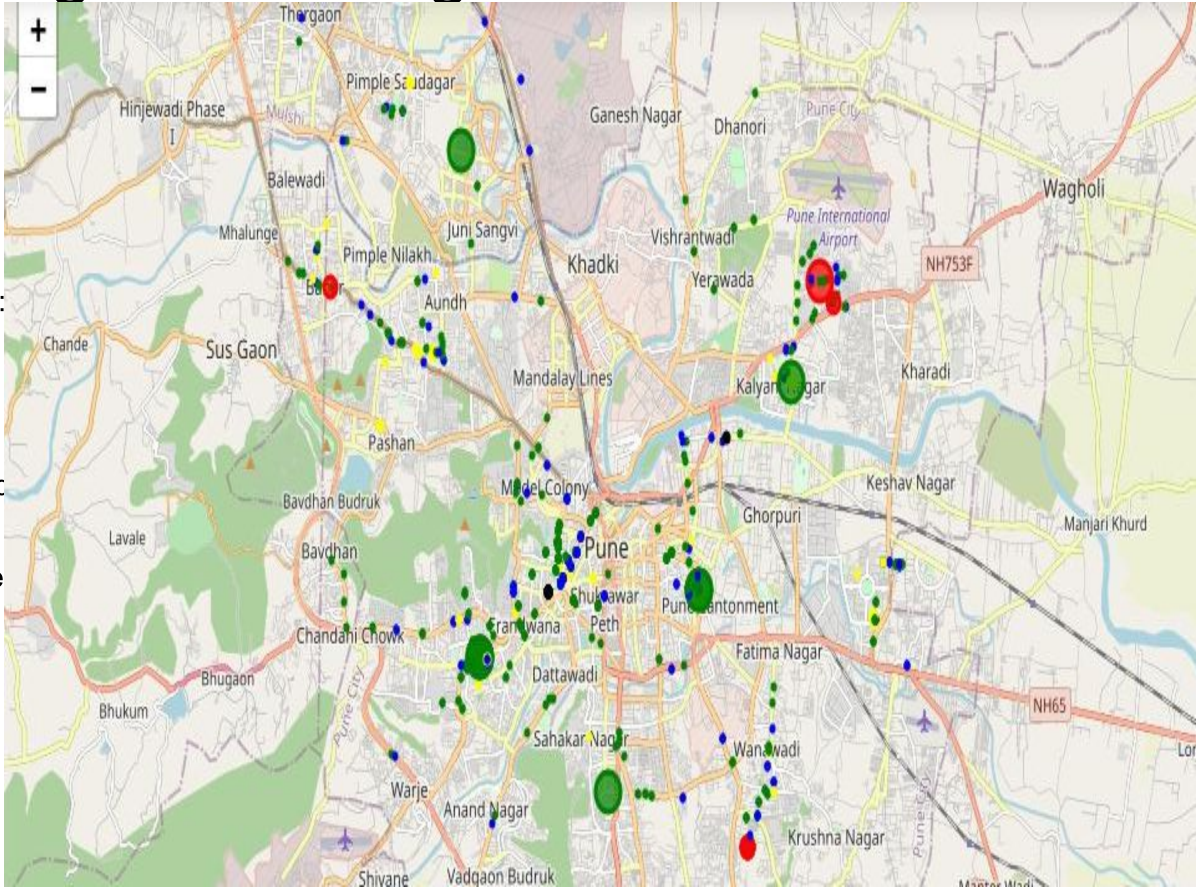


Predictive Modelling: Clustering

KMeans

No of clusters: 5

Cluster label 0:
Cluster label
but
Cluster label
and
Cluster label
affordable and fre
Cluster label
expensive restaurants



Predictive Aggregate Rating

Use multiple linear regression model to predict aggregate rating

Divide data set into training and test sub-sets with 80-20 distribution

Results

```
y_hat = regr.predict(X_test)
x = np.asanyarray(X_test)
y = np.asanyarray(y_test)
print("Residual sum of squares: %.2f"
      % np.mean((y_hat - y) ** 2))

# Explained variance score: 1 is perfect prediction
print('Variance score: %.2f' % regr.score(x, y))
```

```
Residual sum of squares: 0.06
Variance score: 0.88
```

Conclusion and Future Direction

- Built predictive model to cluster current restaurant with similarities
- Used clustering to choose optimum features for thriving restaurants
- Used model in conjunction with statistical metrics to determine location
- Used statistical inference (linear regression) to predict aggregate rating of a restaurant
- Future: use sentimental analysis to mine text within reviews to refine the models further
- Add additional appropriate data to increase the accuracy of the models