Restaurant Success Exploration with Respect to Demographics and Economics

Nicholai L’Esperance

# Abstract

The service industry is a major sector in the US economy, and restaurants make up a significant portion of this. The success of a restaurant depends on a number of factors; the quality of the food, the location, the types of food served, and much more. This project aims to model the success of a restaurant (yelp dataset “is open” and “stars”), using restaurant data such as types of food served and location, employment data, housing data, and crime data. The objective of this project is to create a model that can aid either location selection or restaurant ‘category’ selection for prospective new restaurants, giving a prediction of the ‘star’ rating of a prospective restaurant. Decision tree analysis is used as a white-box tool to show what features are important for restaurant success. Differing tastes per region are explored via the supervised learning model’s prediction capabilities.

# Data Sources and EDA

The primary dataset used in this project is the restaurant information dataset aggregated and published by Yelp. This dataset contains information about restaurant location, foods, as well as a rating of the restaurant. Supplementing this data, I also include cost of living data from worldpopulationreview.com, heath care and income information from kff.org, crime data from ucrdatatool.gov, and some county level health and food security information from countyhealthrankings.org. These datasets are not joined easily, and I will go into depth on the EDA and data manipulations required to join these datasets successfully.

## Yelp Data

Getting the yelp dataset into a usable form required a significant amount of pivoting, filtering, rearranging, and imputation. The scope of this project is aimed at analyzing businesses that served food. In order to create a proper filter, I created a distinct count of all of the categories within the dataset. To filter only food, I iteratively removed any businesses that contained a category label that was obviously related to food. After removing any businesses that were tagged with either ‘food’ or ‘restaurants’ labels, there were no more frequent categories that appeared food related, so this filter was deemed sufficient. The original dataset contained 209,393 records, which included all business types, some of which not in the US. After filtering, I had 52,930 food related businesses in the United States. Additionally, I removed any businesses that did not have any category labels; an additional 405 records were dropped.

Next, additional attributes associated with the restaurant were parsed. These attributes describe the services available at the restaurant, such as parking, if there is delivery, or if there are TVs in the restaurant. Many of these attributes are blank for most of the dataset, so some must be dropped. The ten most common attributes for restaurants are shown below (note, this includes all restaurants, even those outside of the US):

|  |  |
| --- | --- |
| **Attribute** | **Count** |
| RestaurantsPriceRange2 | 72456 |
| BusinessParking | 72277 |
| RestaurantsTakeOut | 65991 |
| OutdoorSeating | 57275 |
| RestaurantsDelivery | 55655 |
| BikeParking | 55478 |
| RestaurantsGoodForGroups | 55249 |
| RestaurantsReservations | 55053 |
| GoodForKids | 54573 |
| Ambience | 53062 |

Most of these attributes are used as features in the final dataset, while some are removed. For example, the BikeParking attribute. This attribute is mostly either True or unknown. This data is not terribly useful for our model. Additionlly, subjectively, I do not expect this feature to be important in our predictions.

Other attributes must be cleaned and imputed. An example of this is ‘NoiseLevel’. The noise level data contains a lot of unclean data, such as “u’quiet’” instead of “quiet”. This data is cleaned, and missing values were imputed with the mean/mode of the dataset, which is “Average”. The values for this attribute are categorical but they are ordered, so attributes such as this one are mapped to integer values. The before cleanup and after imputation and integer mapping can be seen below. We can see there was not a big change to the overall distribution.

A picture containing drawing

Description automatically generatedA picture containing drawing

Description automatically generated

This process of imputation we repeated for all of the attribute and category values to be used as features. Categories, unlike the attribute columns, are one-hot-encoded, giving the yelp data set a very wide shape.

## Supplementary Data Sources

### Cost of Living

These data are simple to import. In order to map this data to the yelp data, we simply needed to map state names to the initialisms, allowing us to join to the Yelp data directly. No imputation required.

### Health Care Cost, Household Income

These data are also simple to import. In additional to the state name mapping, we additionally have to strip and ‘$’ and ‘,’ from the cost column, to convert to numerics. No imputation required.

### Crime Data

This data contains crime data over several years, the newest being 2014. In addition to crime counts, this dataset also includes population measures. Several new features are created from this data. First, all crime counts are converted to per capita counts, simply by dividing counts by the population. This is done for the most recent year only. Secondly, for each type of crime, a new ‘year to year’ change feature set is created, which encapsulates the change of per capita crime rates in each location from 2012 to 2014. No imputation was required for this data.

## County Level Data

County level data for every county was manually combined (datasets are by state). This dataset contains a huge number of possible features. Features of interest included in the model are diabetes\_rate, food\_insecurity rates, and the rate of kids receiving free lunch in school.

Joining this data to our larger dataset is much more difficult, as Yelp does not have county level location data, but rather just latitude and longitude. To do this, I used a reverse lookup geolocator tool “Nominatim” to look up every single latitude and longitude value in the dataset, to then resolve it to a county. However, this results in a lot of odd cases, such as states without counties and towns that belong to zero or more than one counties. For these cases, we can impute with state rates, for simplicity.

## Additional EDA, Correlations

With the dataset engineered, a few correlations were run to see if there were obvious, interesting relationships. First, I took a look at latitude and longitude correlations to see if there were any apparent food relationships.

A screenshot of a social media post

Description automatically generated

There are no strong first order correlations between latitude and longitude to any or our food categories. Interestingly, there is a strong correlation between both latitude and longitude to health care spending and violent crimes.

Next, we check if there are any strong food related correlations to increasing violent crimes.

A screenshot of a cell phone

Description automatically generated

Again, we don’t see any food categories or attributes standing out here, but there are a lot of cost of living correlations. Areas with higher costs of livings are correlated to increasing violent crime rates.

Finally we can take a look at how our output ‘stars’ column correlates to our other features.

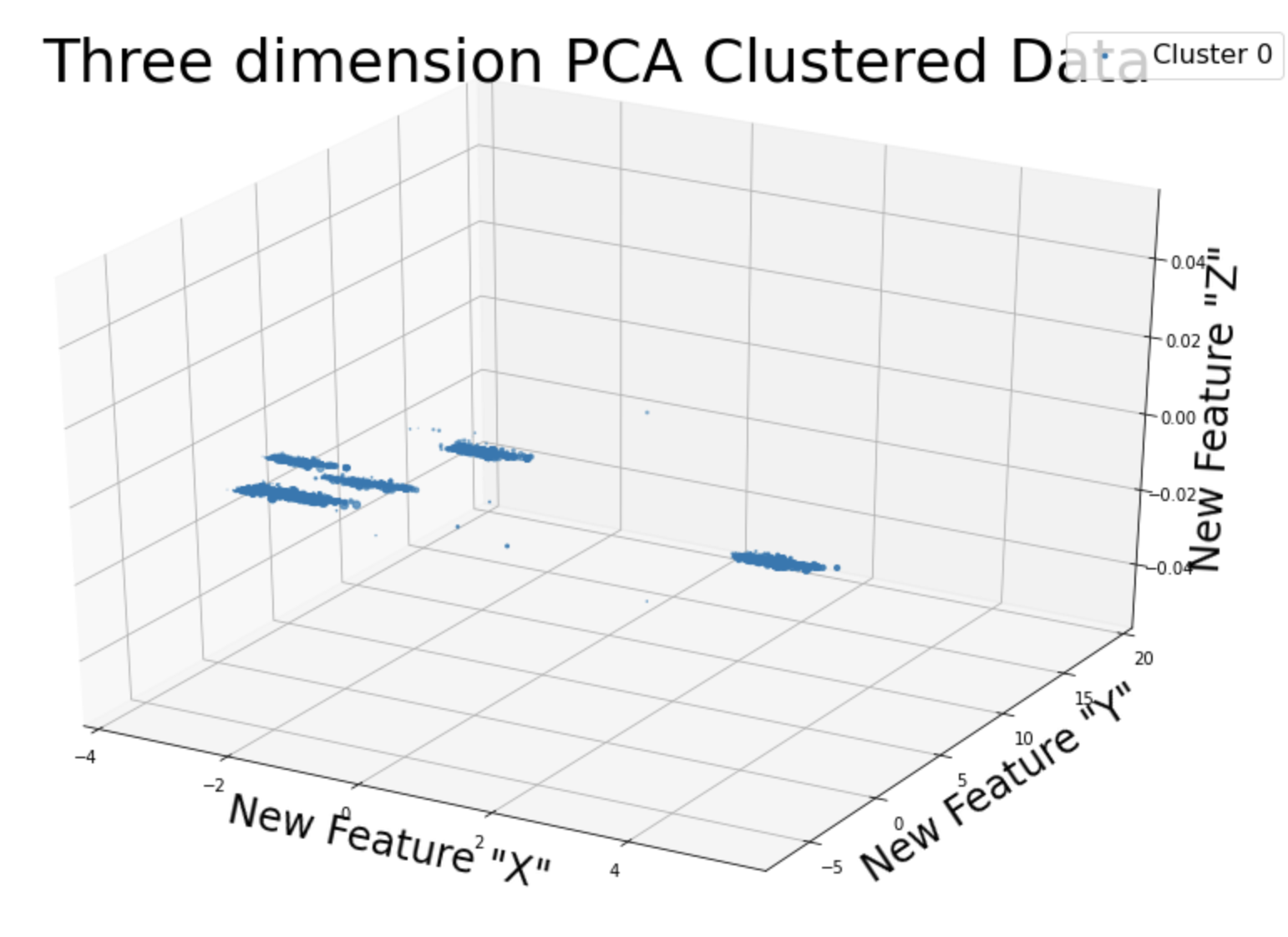
A screenshot of a cell phone

Description automatically generated

No strong correlations here. It will be interesting to see what features end up standing out as ‘important’, once we start training our more complex models. From a correlation standpoint, we don’t see much connection between our yelp dataset and the other joined datasets.

# Unsupervised Learning

In order to better understand our dataset, we can try a clustering algorithm. However, first, I would like to do a 3-d visualization using PCA. We can take a look at our data in three dimensions, to see if there are obvious clusters.



There are five fairly strong clusters in this data. Additionally, it looks like these clusters are likely separable in only two dimensions, so we will try PCA in 2D shortly. We can verify our five clusters using a different method. Below, we use the elbow method and Kmeans to identify that five clusters is in fact our ideal number of clusters.

A picture containing game

Description automatically generated

Given how well these two methods agree, I will use PCA to generate a new two feature dataset, and train a Kmeans clustering model with k=5. We can then take these cluster labels and create a new feature, which can aid our supervised model. Below is the five clusters visualized, the 2D PCA space.

A close up of a map

Description automatically generated

Examining the five clusters, we can get a rough idea to see if there were any obvious defining features.

* Cluster 0: juice bars and smoothies
* Cluster 1: bars, nightlife, pizza
* Cluster 2: juice bars and smoothies, lounges
* Cluster 3: deserts + ice cream & frozen yoghurt + bakeries
* Cluster 4: ethnic food + specialty food + imported food, beauty spas, barbers, cosmetics and stores.

# Supervised Learning

With the additional learning from the un-supervised models, we can now look at selecting a supervised model to perform our ‘rating’ prediction. Our target variable is a continue 1-5 star rating of a restaurant, so we will focus on regression models. The three models evaluated for this project are a gradient boost regressor, a random forest regressor, and an SVM based regressor. Prior to any modeling, we split our data into train and test, at 25% test, 75% train.

For each of the three supervised models, a grid search cross-validation is run, in order to tune for the best hyper-parameters. The search space for the three models can be seen below:

Gradient Boost: {'n\_estimators': [50, 100, 500],

'max\_depth': [3, 5, 7],

'min\_samples\_split': [2, 4, 6]}

Random Forest: {'n\_estimators': [50, 100, 500],

'max\_depth': [None, 10, 50, 500],

'min\_samples\_split': [2, 4, 6],

'min\_samples\_leaf': [1, 5, 8]}

SVM: {'C': [0.5, 1, 2],

'epsilon': [0.01, 0.1, 1, 10]}

Running the grid search, we can select the best model from each type, and then compare the results on the test set. Below, the best parameters for each model are shown.

Gradient Boost: {'max\_depth': 5,

'min\_samples\_split': 6,

'n\_estimators': 500}

Random Forest: {'max\_depth': 7,

'min\_samples\_split': 6,

'n\_estimators': 500}

SVM: {'C': 0.5,

'epsilon': 0.1}

Now, we can run each of the three models on the test set, and compare performance. To evaluate the models, we can look at both MSE and MAE scores.

|  |  |  |
| --- | --- | --- |
| Model | MAE | MSE |
| Gradient Boost | 0.573 | 0.529 |
| Random Forest | 0.621 | 0.599 |
| SVM | 0.568 | 0.534 |

The gradient boost and SVM models had near identical performance; SVM had marginally better MAE and marginally worse MSE. However, the SVM model takes significantly longer to train, so I will select the Gradient Boost model going forward.

# Data Analysis

## Restaurant Rating By Location

With our trained model, we can now explore some additional learnings. As an experiment, we can try to see where in the United States we can expect to find the best Italian food. For this experiment, we can take an example row from our dataset, and alter the latitude and longitude. At each point, we can use our model to create a prediction of the star rating of this restaurant. Below, we can see a heatmap of Italian restaurant rating by location.

A screenshot of a cell phone

Description automatically generated

We can see there are two main ‘hotspots’, in the north west, and in the south east. In both of these locations, Italian restaurants tend towards higher ratings. After running several different types of food, these locations tend to have higher ratings no matter the type of food. While there are small differences, the north west and south east continually are predicted to have higher rated restaurants. Below is a similar heatmap, for ethnic food. There is one key different I want to note. It appears that we can expect higher rated ethnic restaurants farther south from the north west than we can Italian (see the extended green section at 25 latitude, -101 longitude).

A screenshot of a cell phone

Description automatically generated

## The making of a good restaurant

Another good question might be, what makes a restaurant good? We can attempt to learn this using a decision tree, looking at feature importances. I’ll use GridSearchCV to tune hyperparameters. The decision tree results on the full dataset give us the following feature importances:

['fast\_food', 'convenience\_stores', 'longitude', 'latitude', 'coffee\_&\_tea']

Latitude and longitude actually do play an important role in our tree. We can visualize the tree to see how some of these factors come to play. As we expected from the feature importances, we can see fast food is the very first feature. The tree shows us that, if a restaurant is considered fast food, it tends to have a worse star rating. Intuitively this makes sense. Note, even though latitude and longitude are considered important features, they are not shown on the first three levels of the tree, they are likely closer to the leaves (this tree is 68 degrees).

A close up of a piece of paper

Description automatically generated

## Regional Preferences

Given a fixed location, can we gleam information about preferred foods in that region? To do this experiment, we take an example row from two locations: Texas and Vermont. Each row contains untouched information about that location, such as costs of living and crime rates. Then, we make a prediction on the rating given a number of different possible foods served. For each region, this gives us a way to compare food preferences against each one another. Then, scaling the results to the highest rated food, we can then compare two regions against one another directly.

A close up of a fence

Description automatically generated

Overall, the food preferences in the regions appear quite similar. There are 2-3 places where the preferences appear to differ some. First, ice cream and frozen yoghurt is a bit more popular in Vermont than in Texas, as is fast food. Vietnamese food is slightly more popular in Texas than it is in Vermont. Overall, however, the general rating of these different foods does appear to match between these two locations.

# Conclusions and Findings

There is a lot of intertwined relationships between restaurant success, demographics, and economics. While this modeling project barely scratches the surface of signals to be analyzed, it is clear through both the unsupervised and supervised models that there are strong relationships in the data. Our unsupervised analysis found there are five very distinct clusters in our data, a finding that would be difficult to surmise directly from the data itself. The supervised modeling was able to predict a restaurant’s success within half a star consistently. Using the supervised model, we can see geographic areas that tend to higher rated restaurants in general. More interestingly, we can see how some regional food preferences change by geographic location.

There is a near endless amount of future work to continue, such as working to model local crime rates as a target variable based on restaurant information in the region, or even trying to predict the rate of diabetes in a region based on food preferences; this list goes on.