Xtern 2021 Data Science

October 19, 2020

Introduction

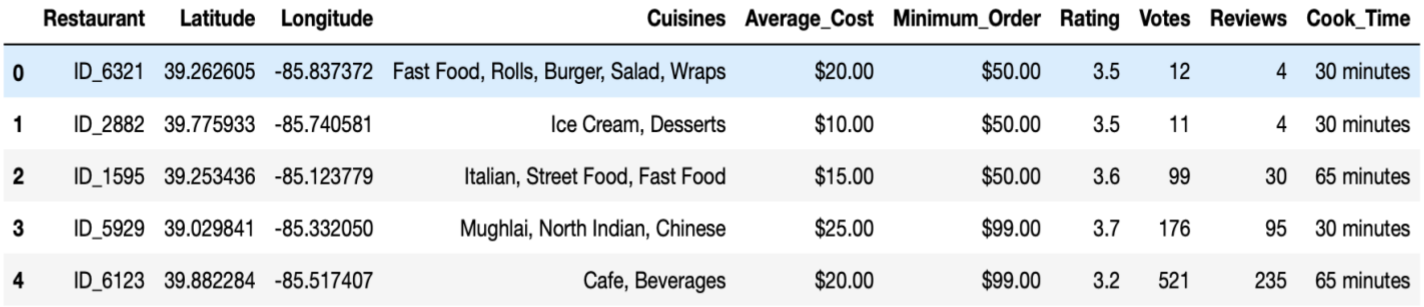
As a member of the data science team, I was responsible for drawing insight from the data available from restaurants around the area. The data provided was in csv format and included various information about restaurants. The list of categories that was available are:

- *Restaurant ID*
- *Latitude*
- *Longitude*
- *Cuisine*
- *Average costs*
- *Minimum costs*
- *Rating*
- *Votes*
- *Reviews*
- *Cook time*

I decided to use python to manipulate data and use some of my visualization and Machine Learning skills to generate some interesting finds about the restaurants in this location. The task was to deduce **FOUR** insights and I have will list them in this report.

Pre-Processing

The data given to us had information of over 2000 restaurants. However, there were a lot of restaurants that lacked information about their rating, reviews, votes and so on. Since there is no way to estimate them from the data available, I decided to drop the rows that lacked this information. An initial look of the data:

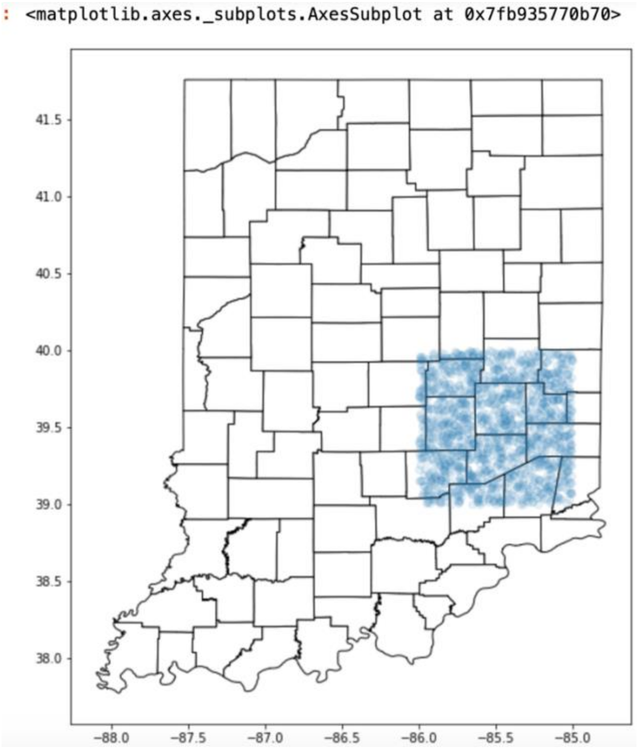


	Restaurant	Latitude	Longitude	Cuisines	Average_Cost	Minimum_Order	Rating	Votes	Reviews	Cook_Time
0	ID_6321	39.262605	-85.837372	Fast Food, Rolls, Burger, Salad, Wraps	$20.00	$50.00	3.5	12	4	30 minutes
1	ID_2882	39.775933	-85.740581	Ice Cream, Desserts	$10.00	$50.00	3.5	11	4	30 minutes
2	ID_1595	39.253436	-85.123779	Italian, Street Food, Fast Food	$15.00	$50.00	3.6	99	30	65 minutes
3	ID_5929	39.029841	-85.332050	Mughlai, North Indian, Chinese	$25.00	$99.00	3.7	176	95	30 minutes
4	ID_6123	39.882284	-85.517407	Cafe, Beverages	$20.00	$99.00	3.2	521	235	65 minutes

I also removed the $ signs in cost columns and converted the data to float data type to do my calculations. I removed the sub-string ‘minutes’ from cooking time and converted the data types for all numerical columns so I can use them for calculations. I used the ***geopandas*** library provided by pandas to use the (lat, long) information so I can plot them in graphical space. I didn’t use the Restaurant ID for any insight as it was ‘invaluable’ data for my project. I use this clean dataset for creating insights, which are discussed below.

1.Location of Restaurants

I used the geographical information provided to plot the restaurant locations in an Indiana map so that I can see where these restaurants are located. This information can be helpful in deciding which locations to employ more delivery drivers to. I used a county level of a base map and plotted the following graph:



```
: <matplotlib.axes._subplots.AxesSubplot at 0x7fb935770b70>
```

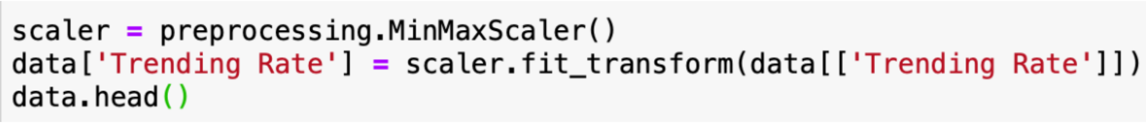
I was able to see which counties the businesses were involved in and also helps me get a sense of the county laws Foodie-X would need to be aware of. I made each point in the graph translucent, so a darker area in the graph suggests locations with densely populated restaurants. We can see which locations are denser and then use that info to hire more drivers in those locations.

2.What affects the trend of Restaurants?

We want to know which restaurants are more favored by customers so that we know which types of restaurants are potentially going to generate more orders. I wanted to create a column that gave me a Trending Rate for each restaurant using rating, votes and reviews. I used the formula:

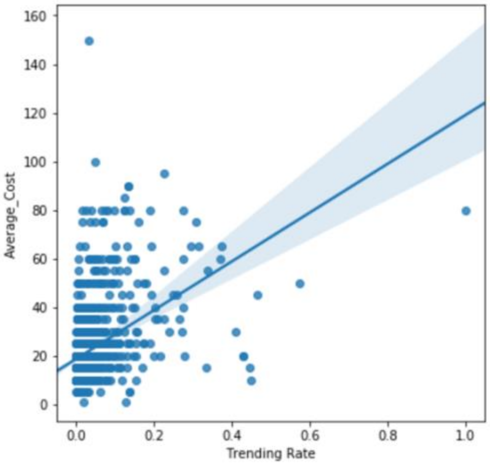
$$\textit{Trending rate} = \textit{Ratings}^2 + \textit{Votes} + \textit{Reviews}$$

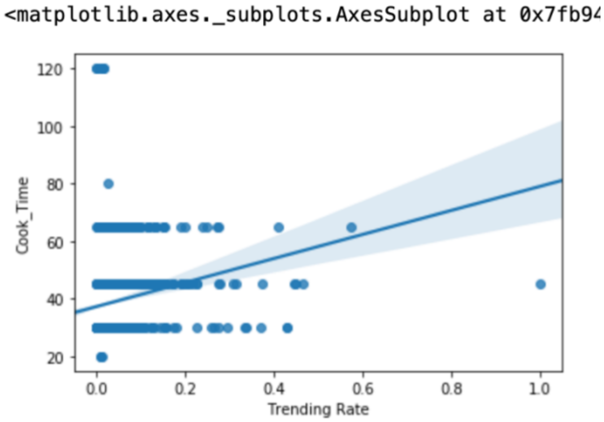
I wanted to square the ratings to magnify its impact on overall score. However, using such a formula would give me a trending rate score that had large range. I wanted to have a bound range for this metrics, so I decided to normalize the values from the above formula. I used the MinMaxScaler() function from sklearn library:



```
scaler = preprocessing.MinMaxScaler()
data['Trending Rate'] = scaler.fit_transform(data[['Trending Rate']])
data.head()
```

I then used this normalized trending rate data and plotted this information against Average costs and Cooking Time. I hypothesized that restaurants with higher costs and longer cooking time would be less trending. I created a scatter plot with a best fit line and got the following visuals:



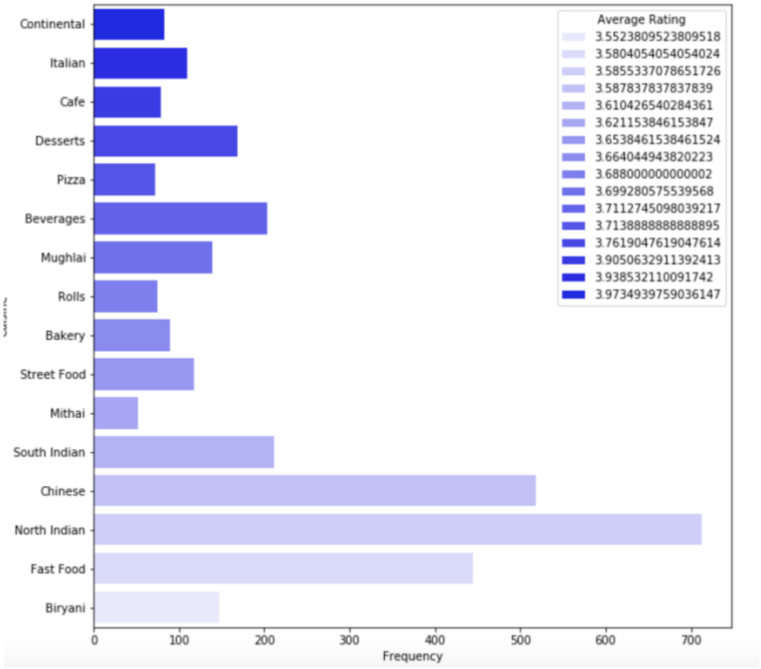


```
<matplotlib.axes._subplots.AxesSubplot at 0x7fb94
```

I was surprised to see the positive sloped lines for both average costs and the cooking time. I am guessing higher costs and better cooking time is resulting in better food and better dining experiences for customers so that could be causing this increase in customer satisfaction with rising costs and cooking time. We can use this information to see what type of restaurants are yielding better customer satisfaction and develop partnerships with such restaurants. However, causality and causation are not the same. So, we should be careful about interpreting these results and overstating our findings.

3. Are certain Cuisines in the Restaurant Market Saturated?

Looking at the data, I noticed that some types of cuisines were more frequently offered than others. I wanted to see if there were any variations in what the customers were liking vs. what was offered in the restaurant business in this location. I broke down each cuisine offered by each restaurant, and then computed the frequency of each type of cuisine. I also assigned ratings to cuisines based on what rating each restaurant had received to compute average rating for each cuisine. I captured all this information in the following bar chart:



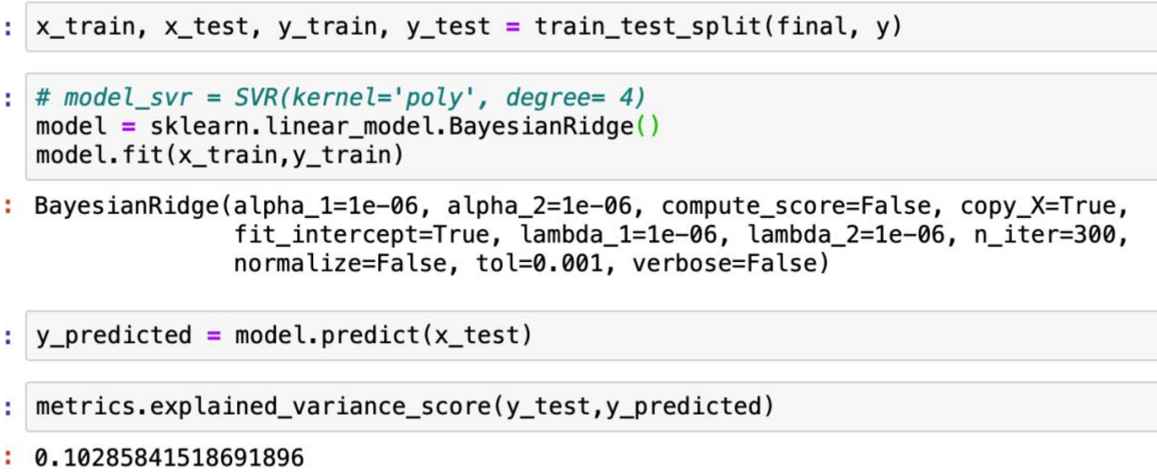
Shorter bars suggest small number of restaurants offering the specific cuisine while taller bars suggest high frequency of that cuisine in the market. Likewise, a dark bar means higher ratings that the lighter bars. I sorted the bar by their hue for easy interpretation. I also only am including cuisines with higher than 50 occurrences so that the bar chart is not very cluttered. From this graph, we can clearly see that, in general, highly rated cuisines are not as widely available as the lower rated cuisines. So these restaurants with niche cuisines are creating a better experience for customers than restaurants that have cuisines that a lot of restaurants offer. This could be because of restaurants trying to create a brand for their unique cuisines.

4.Predict Cooking Time

It would be very useful if we could predict what the cooking time for a restaurant is going to be. This would help us be efficient at allocating resources as we wouldn’t need to make a driver wait at the restaurant until the food is cooked or notify a driver only after the food was cooked and ready to be picked up. For this purpose, I decided to use sklearn’s machine learning algorithm to create regression models to predict cooking time for restaurants. After manipulating the data a little bit, I was able to use the cuisines offered by restaurant, their average cost, the minimum cost and their rating to predict cooking time.

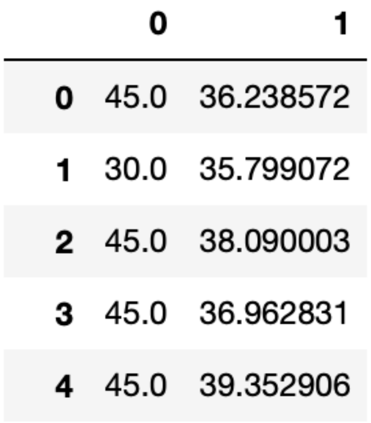
These are not very good independent variables to predict cooking time, however I wanted to see how good of a model I could make with the data available. Due to time limitations, I was only able to compare Linear Regression, SVR (Support Vector Regressor) and the Bayesian Ridge model offered by sklearn library.

All these models were not able to explain a whole lot of change in cooking time. The best model out of the three was Bayesian Ridge and it was able to explain about 10% of the variation in cooking time.



```
: x_train, x_test, y_train, y_test = train_test_split(final, y)

: # model_svr = SVR(kernel='poly', degree= 4)
  model = sklearn.linear_model.BayesianRidge()
  model.fit(x_train,y_train)

: BayesianRidge(alpha_1=1e-06, alpha_2=1e-06, compute_score=False, copy_X=True,
                fit_intercept=True, lambda_1=1e-06, lambda_2=1e-06, n_iter=300,
                normalize=False, tol=0.001, verbose=False)

: y_predicted = model.predict(x_test)

: metrics.explained_variance_score(y_test,y_predicted)

: 0.10285841518691896
```

This is not the best result, nor the type of result I had hoped for, but we could potentially experiment with different regression models to see if there are any other models that can better predict the time. We could also collect other types of input data to train the model to improve its prediction abilities. A sample of actual vs. prediction is given in the table below:



	0	1
0	45.0	36.238572
1	30.0	35.799072
2	45.0	38.090003
3	45.0	36.962831
4	45.0	39.352906

The column ‘0’ has the actual cooking time for the restaurant and ‘1’ has the predicted cooking time.

All the code for this project which I wrote to create the tables, graphs, and the findings are in the python notebook attached with this report.