Restaurant Rating Prediction

Tool: python, jupyter notebook, machine learning model, ms sql, power bi
Source: Kaggle
Duration: 3 weeks

Problem Statement

My friend, Nicole has a restaurant at BTM in Bangalore, India. She is thinking of joining Zomato as new restaurant to increase her sales. However, before joining she would like to know if her restaurant will be able to get at least rate 4 over 5. She doesn't think of moving her restaurant since she just renovated. Other than that, she is open for any suggestion that can achieve her target.

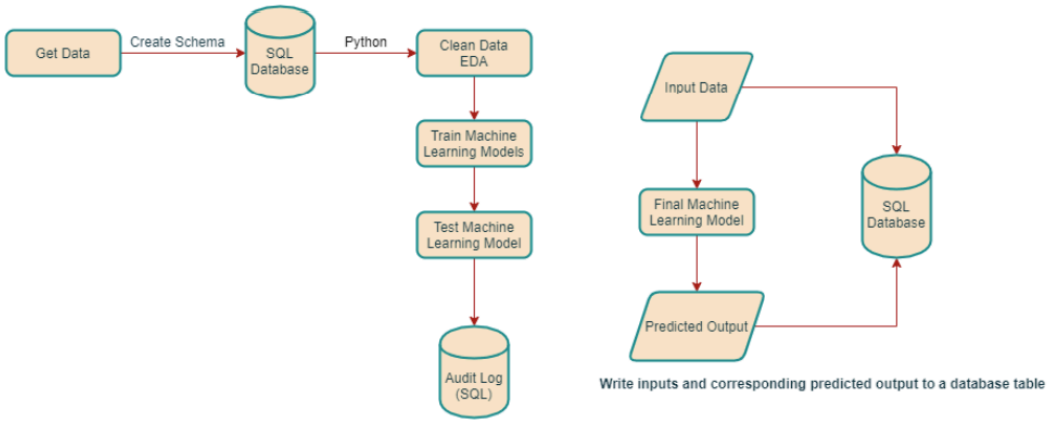
Objective

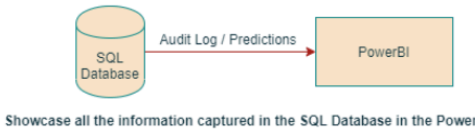
The objective is to predict her restaurant rating at Zomato platform by using Machines Learning techniques and identify the important features that for high rating.

Data Description

Zomato is a platform that provides information, menu and user-reviews of restaurant as well as food delivery option from partner restaurant in select cities. This data is scraped and accurate to that available on the Zomato website until 15 March 2019. This is a supervised Machine Learning and I treat it as Regression problem. I will be using Linear Regression, Lasso, Decision Tree Regression and Random Forest Regression as training models. The metrics will be in R square, MSE and MAE.

Workflow





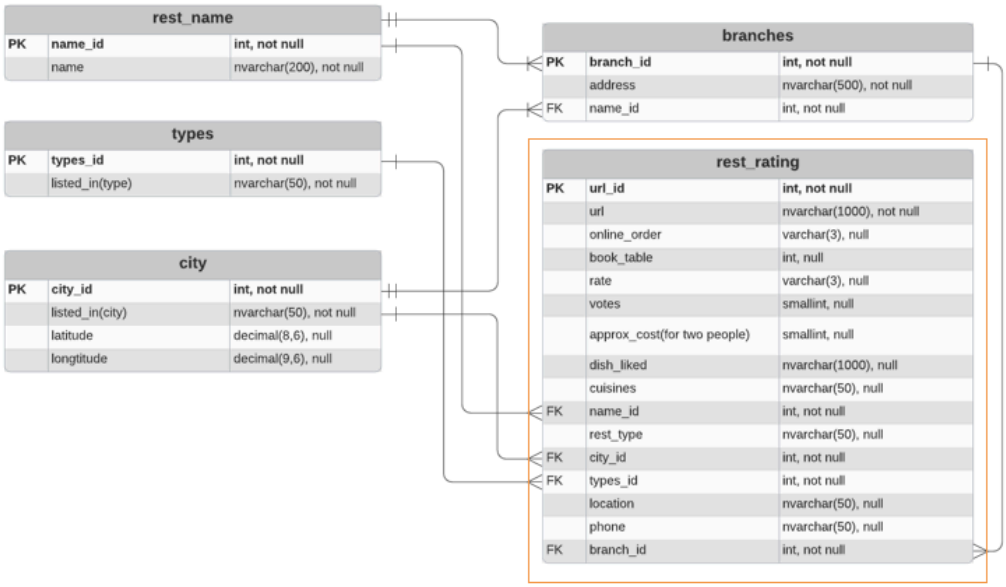
Extract Transform Load (ETL)

	feature	null	dtype	uni_value	max_len
0	url	0	object	51717	538.0
1	address	0	object	11495	346.0
2	name	0	object	8792	159.0
3	online_order	0	object	2	3.0
4	book_table	0	object	2	3.0
5	rate	7775	object	64	6.0
6	votes	0	int64	2328	0.0
7	phone	1208	object	14926	34.0
8	location	21	object	93	29.0
9	rest_type	227	object	93	29.0
10	dish_liked	28078	object	5271	134.0
11	cuisines	45	object	2723	86.0
12	approx_cost(for two people)	346	object	70	5.0
13	reviews_list	0	object	22513	1284117.0
14	menu_item	0	object	9098	24897.0
15	listed_in(type)	0	object	7	18.0
16	listed_in(city)	0	object	30	21.0

This is the information of the initial data. From here, feature is the columns. There are 17 columns and 51,717 rows in this data. Url is the unique value because every rating come from a url. In the 'rate' column, the initial value looks like '4.1/5', so I will remove the '/5'. And the column contained '-' and 'New' value which I think it is the new join in restaurants that without the rating. There for I will remove the value and treat it as new restaurant that will be predicted later. I will drop 'review' column as well, since I already

have the 'rate' column. The maximum length column is giving me the idea that the number of nvarchar that I need for setting up the schema later.

Entity Relationship Diagram (ERD)



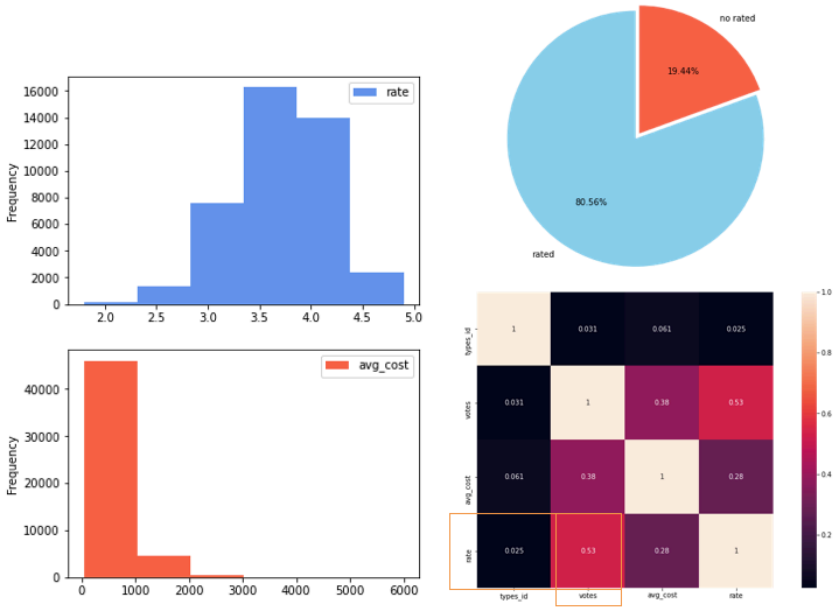
After created an ER Diagram, use pyodbc to store the tables in Ms SQL database. For the rating prediction, I will use rest_rating table.

Exploratory Data Analysis (EDA)

This is the information of the rest_rating table.

	feature	null	dtype	uni_value	max_len
0	url_id	0	int64	51717	0.0
1	url	0	object	51717	538.0
2	online_order	0	object	2	3.0
3	book_table	0	object	2	3.0
4	rate	10052	float64	31	0.0
5	votes	10027	float64	2327	0.0
6	phone	0	object	14927	34.0
7	location	21	object	93	29.0
8	rest_type	227	object	93	29.0
9	dish_liked	28078	object	5271	134.0
10	cuisines	45	object	2723	86.0
11	avg_cost	346	float64	70	0.0
12	menu_item	0	object	9095	8000.0
13	name_id	0	int64	8743	0.0
14	types_id	0	int64	7	0.0
15	city_id	0	int64	30	0.0
16	branch_id	0	int64	11466	0.0

First of all, I don't need all the columns as features. I will remove url, phone, name, branch, because they are just information about the restaurant. I will remove those location and city also, since Nicole has no intention to change location. I will also drop the menu and dish liked column. After removed unnecessary data. There are some null value which I will treat it later.



Let's look at the distribution of the target. From the histogram, the distribution is quite normalized. To me it is good to continue. The url without rating is about 20%. For this portion I will keep it for prediction. In the average cost, most of the price are within 1k rupee, so the null value I will replace with median. I will also remove the votes column. In fact, I'm predicting the rating of new restaurant. However votes is a information that will be known after launching a restaurant. So I think the column is not necessary.

types_id	meal_type
1	Buffet
2	Cafes
3	Delivery
4	Desserts
5	Dine-out
6	Drinks & nightlife
7	Pubs and bars

	online_order	book_table	types_id	avg_cost	rest_type	cuisines	rate
0	No	Yes	1	3000.0	Fine Dining	Continental, North Indian, Italian, Chinese	4.0
1	No	Yes	5	3000.0	Fine Dining	Continental, North Indian, Italian, Chinese	4.0
2	Yes	No	3	250.0	Quick Bites	North Indian, Fast Food, Street Food	3.9
3	Yes	No	5	250.0	Quick Bites	North Indian, Fast Food, Street Food	3.9
4	Yes	No	3	250.0	Quick Bites	North Indian, Fast Food, Street Food	3.9
...	...	...	...	...	...	...	...
51712	No	No	3	NaN	Takeaway, Delivery	Continental, Italian, Steak, American	4.1
51713	No	No	3	200.0	Food Truck	Fast Food	3.4
51714	No	No	3	200.0	Food Truck	Fast Food	3.4
51715	No	No	5	200.0	Food Truck	Fast Food	3.4
51716	No	No	5	200.0	Food Truck	Fast Food	3.4

51717 rows × 7 columns

This is the table after extracting the attributes for machines learning. There are 7 types. Types means Buffet, Café, Delivery, Dessert, Dine-out, Drink & nightlife and

Pubs and bars. Restaurant type is more detail, for example, Quick Bites, Bakery, Casual Dining, Food Court, Kiosk, Sweet Shop, etc. And in the 'cuisines' columns, its value is a list.

	types_id	avg_cost	rate
count	51717.000000	51371.000000	41665.000000
mean	3.807375	555.431566	3.700449
std	1.140839	438.850728	0.440513
min	1.000000	40.000000	1.800000
25%	3.000000	300.000000	3.400000
50%	3.000000	400.000000	3.700000
75%	5.000000	650.000000	4.000000
max	7.000000	6000.000000	4.900000

From the statistic view, the average price are within 40 – 6000 rupee and the average is around 555 rupee. For rate are within 1.8 – 4.9 and the mean of the rate is around 3.7. So, I will said above 3.7 rate is consider a good rating.

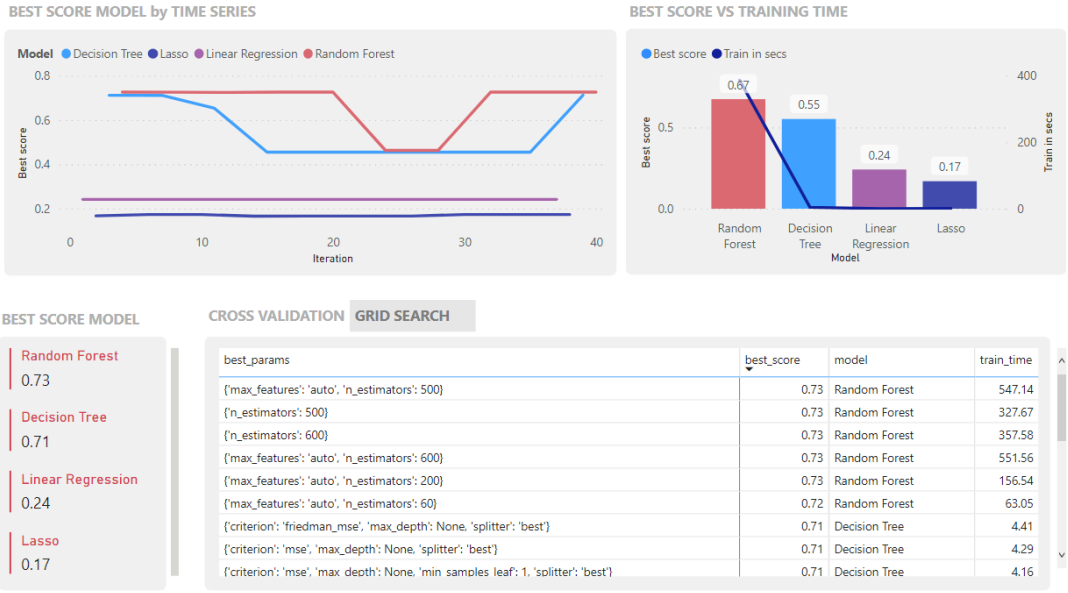
Data Preparation

- Replaced the null value in the 'avg_cost' column by median value.
- Replace null in 'rest_type' and 'cuisines' column by random value.
- Using label encoding for categorical data.
- After excluded new restaurant from the training set, the remaining data is 41,665.
- Then I allocated 80% for training and 20% for testing.

Machine learning model training/evaluation

Before training, I use the total training data for grid searching the best parameter and R square scoreLinear SVC. For the models, I used:

1. Random Forest Classifier
2. MLP Nueral Networks
3. Using SelectBest to select 5 best features.



best_params	best_score	model	train_time
{'max_features': 'auto', 'n_estimators': 500}	0.73	Random Forest	547.14
{'n_estimators': 500}	0.73	Random Forest	327.67
{'n_estimators': 600}	0.73	Random Forest	357.58
{'max_features': 'auto', 'n_estimators': 600}	0.73	Random Forest	551.56
{'max_features': 'auto', 'n_estimators': 200}	0.73	Random Forest	156.54
{'max_features': 'auto', 'n_estimators': 60}	0.72	Random Forest	63.05
{'criterion': 'friedman_mse', 'max_depth': None, 'splitter': 'best'}	0.71	Decision Tree	4.41
{'criterion': 'mse', 'max_depth': None, 'splitter': 'best'}	0.71	Decision Tree	4.29
{'criterion': 'mse', 'max_depth': None, 'min_samples_leaf': 1, 'splitter': 'best'}	0.71	Decision Tree	4.16

by looking at the iteration searching time, Random Forest is remaining the highest score across the 4 models. However, its training time is also the longest.

After splitting the 80%, 20% to train data and test data, 33,332 data is the training set. By using the best parameters for each of model, Random Forest and Decision Tree are the top 2 accurate models. Although Random Forest training time is much longer than Decision Tree, but within 1 min I think is ok, so I will choose random forest as my best model for prediction.

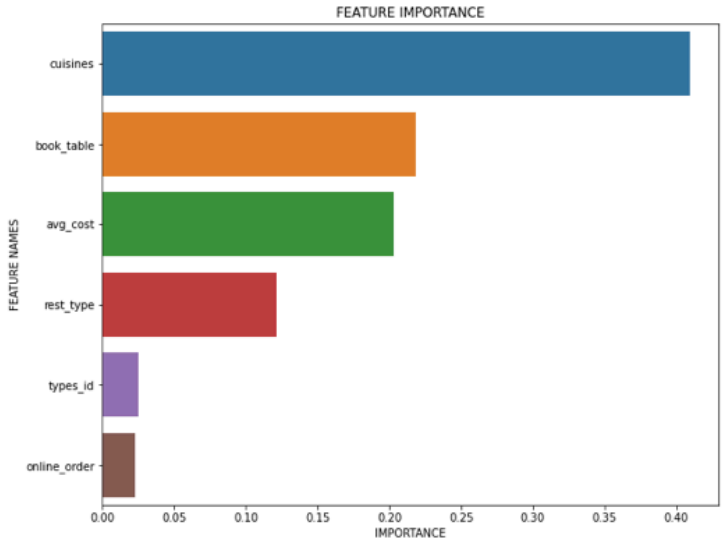
NICOLE's RESTAURANT INFO.

cuisines	online_order	book_table	type	avg_cost	rest_type
American, Cafe, Continental, French, Burger, Mexican, Desserts, Pizza	No	0	Buffet	450.00	Bakery, Sweet Shop

NICOLE's RESTAURANT RATE PREDICTION

By using Random forest as the best model, after tested the 8,333 data and here is the comparison between the actual and the predicted rating. I continue to predict the rating of those new restaurants and I get the average of 3.5 rating for each restaurants. I process to input my friend's restaurant data into the model and it predicted the rating of 3.7 which is actually same as the mean value.

Conclusion



The predicted rate is 3.7 which is lower than what Nicole expect. In order to get a higher rate, I suggest her to reference the feature important and make some adjustment to her restaurant's data. Like for example, she can fine tweak her restaurant cuisines; consider to provide the 'book table' service or even increase her food price since the mean of the avg_cost is around 555 rupees.

Future Opportunities

For the future opportunity, I will do inverse encoding. Because now everything is number which is difficult to understand. Coming up will look into more detail on the existing features, there is 2723 unique value in the 'cuisines' columns, I wonder if label encoding is the best approach to encoding my categorical data. And I will also look for more data. Scrape the data from different country so that I can analyst by region. Last but not least, I would explore more on the review column in order to better understand the customer feedback.