Classify Restaurant Rate

Xiaojun zhu, Jhao-Han Chen, Haiping Sun

Instructor: Amir H Gandomi

STEVENS INSTITUTE of TECHNOLOGY
THE INNOVATION UNIVERSITY®
Business Intelligence & Analytics

Introduction

In our daily life, people often use mobile applications to view a restaurant's rating and decide which restaurant to eat. Therefore, figuring out which variables have a greater impact on restaurant rating is important for entrepreneurs while starting a restaurant. The goal of our project is to find the best classified method to discriminate the good and the bad restaurant according to the variables in the dataset we used.

Experiment

Data_Collection

Our data comes from website Kaggle and the original dataset includes 11 features such as restaurant category, station, Review Number, Dinner Rating, Dinner Price and so on.

Here is the explanation of some features

station_class0	Low density	**C_Cafe**	FirstCategory for Cafe
station_class1	Middle Density	**C_European**	FirstCategory for Europen
station_class2	High Density	**C_Japanese**	FirstCategory for Japanese
C_Asian	FirstCategory for Asian	**C_Noodle**	FirstCategory for Noodle
C_Bar	FirstCategory for Bar		

Data Cleaning

Firstly, as we process the raw data, we found that categories such as BBQ, Mexico and Seafood are rare, so we delete those data.

Then, to make it easier to analysis, we separate variables 'station' and ' First Category' into dummy variables.

	station_class1	station_class2
low density	0	0
median density	1	0
high density	0	1

Finally, as our target cell is dinner rating, we take the rating more than 3.07, which is the median of the rating, as good restaurant and rating less than 3.07 as bad restaurant.

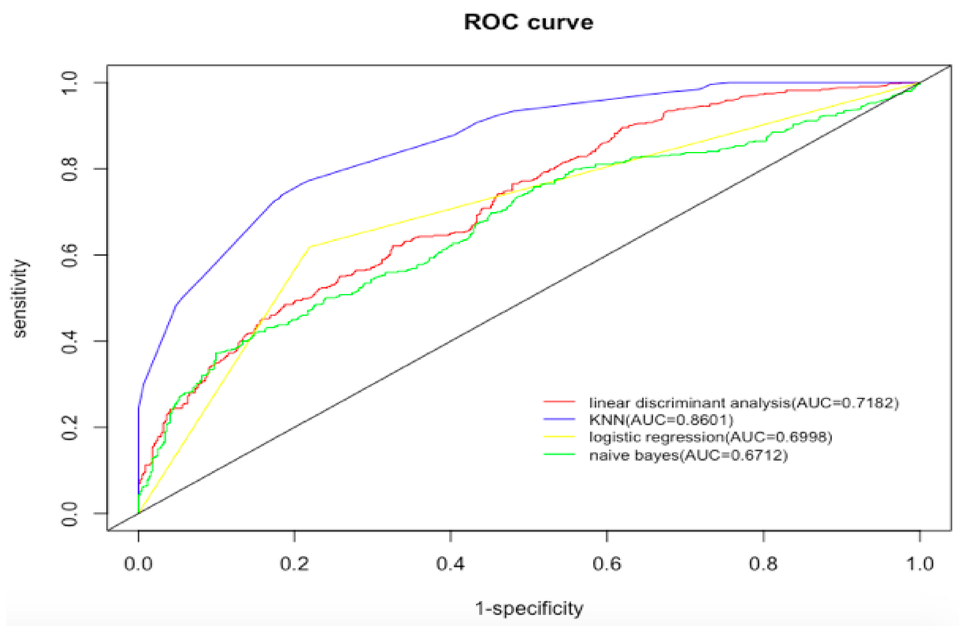
Feature Selection

After cleaning the data, we generate a correlation coefficient matrix to see which factor should we use in our analysis. The result shows that variables "station_class1" and "station_class2", variables "C_Bar" and "C_Japanese" have high correlation which are greater than 0.5, so we decide to drop variables "station_class1" and "C_Japanese".

Attributes	C_Bar	C_Cafe	C_European	C_Japa...	C_Noodle	DinnerPrice	ReviewNum	station_class1	station_class2
C_Bar	1	-0.162	-0.366	-0.620	-0.100	0.008	-0.155	-0.090	0.128
C_Cafe	-0.162	1	-0.068	-0.115	-0.019	-0.024	0.041	0.026	-0.065
C_European	-0.366	-0.068	1	-0.260	-0.042	0.016	0.053	0.011	-0.004
C_Japanese	-0.620	-0.115	-0.260	1	-0.071	0.005	0.086	0.081	-0.106
C_Noodle	-0.100	-0.019	-0.042	-0.071	1	-0.026	0.045	0.024	-0.031
DinnerPrice	0.008	-0.024	0.016	0.005	-0.026	1	-0.008	0.026	0.043
ReviewNum	-0.155	0.041	0.053	0.086	0.045	-0.008	1	-0.008	-0.011
station_class1	-0.090	0.026	0.011	0.081	0.024	0.026	-0.008	1	-0.687
station_class2	0.128	-0.065	-0.004	-0.106	-0.031	0.043	-0.011	-0.687	1

Model

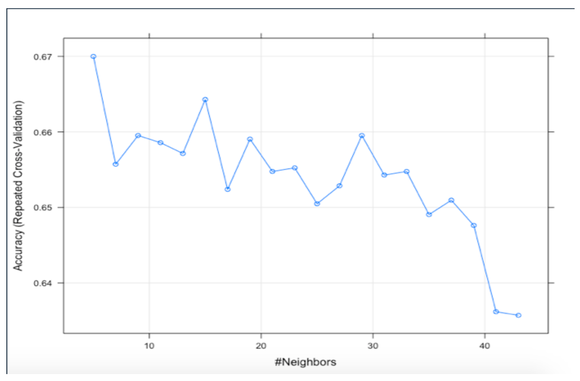
Our group test four classification machine learning algorithms to build model including Logistic Regression, Naïve Bayes, K-Nearest Neighbors and LDA.



model	AUC
Linear Discriminant analysis	0.7182
KNN	0.8601
Logistic Regression	0.6998
Naïve Bayes	0.6712

Result

By testing the accuracy of several models, we finally decide to use KNN to fit our data set. Also while k=5, the Model has the highest accuracy rate.



we randomly choose 10 restaurants to test our model

ID	Name	station_class2	C_Bar	C_Cafe	C_European	C_Noodle	DinnerPrice	ReviewNum
28	Komefuku	1	0	0	0	0	4500	10
39	Zasoudouhigashiyamakyouto	0	0	0	1	0	9000	272
49	Banikuryourisemmontenumayarou	1	1	0	0	0	3500	52
74	Ichishun	1	1	0	0	0	4500	72
102	Yamadaruma	1	1	0	0	0	3500	10
448	Houen	1	0	0	0	0	3500	15
499	Aburaya	1	1	0	0	0	3500	5
524	itariamba-rukimuraya	0	0	0	1	0	4500	28
665	Teuchisobaminagawa	0	0	0	0	1	3500	9
830	RH Cafe	1	0	1	0	0	3500	31

The result of the classification have high accuracy rate based on our model.

			probability	
ID	true classify	predict	bad	good
28	bad	bad	0.6	0.4
39	good	good	0.2	0.8
49	good	good	0.4	0.6
74	good	bad	0.6	0.4
102	bad	bad	1.0	0.0
448	good	good	0.4	0.6
499	bad	bad	1.0	0.0
524	good	good	0.4	0.6
665	bad	bad	1.0	0.0
830	good	bad	0.6	0.4

Conclusion

- We achieve AUC of up to 0.8601 by using KNN model.
- KNN algorithm is sensitive to distance between variables, therefore it is better to normalize the feature