Coursera Capstone Project for IBM Data Science Specialization - Week 2

By Aravindan Natarajan

1. Introduction

1.1. Background

Dallas city is one of the most populous cities in U.S. and is home to many immigrant populations in Texas after San Antonio and Houston. Furthermore, it is the fourth-largest metropolitan area in the U.S. at 7.5 million people as of 2018 with an estimated population of 7,846,293 residents. Being a metropolitan city, Dallas is also home to many restaurants which serves wide variety of cuisines. Owing to significant number of Indian expatriate population, Dallas City and its nearby Suburbs have handful of Indian restaurants.

If someone from India visits Dallas City for the first time, it will be useful if he/she have some prior information about the Indian Restaurants in Dallas City and how good they are. Moreover, prior information on location of other restaurants and their violation history will help in coming to an informed decision.

So, as a part of this project using the Dallas City Inspection Data and FourSquare API Indian restaurants in Dallas City will be listed, visualized and rated.

1.2. Problem Description

By utilizing the Dallas City restaurants inspection data, Indian Restaurants in Dallas City and their risk category will be Analyzed. Secondly, a classsifier model will be built to predict the risk categories of resturants. Furthermore, using the foursqure API the ratings of Indian Restaurants in Dallas City will be obtained.

1.3. Target Audience

- People looking to open new restaurants
- Restaurants
- Travellers who love Indian food

2. Data

For this project I will use the following data :

1. Dallas City restaurants inspection data from 2016-2019
    - Data source : https://www.dallasopendata.com/api/views/dri5-wcct/rows.csv?accessType=DOWNLOAD
    - Description : This data set contains 37876 rows and 114 coulmns with Restaurant Name, Street Name, violation descriptions along with their latitude and longitude. This data will be downloaded and used.

	Restaurant Name	Inspection Type	Inspection Date	Inspection Score	Street Number	Street Name	Street Direction	Street Type	Street Unit	Street Address	Zip Code	Violation Description - 1	Violation Points - 1	Violation Detail - 1	Violation Memo - 1	Violation Description - 2
0	HARVEY'S	Routine	10/03/2016	82	12835	PRESTON	NaN	RD	#306	12835 PRESTON RD #306	75230	*34 Outer door: solid,selfclosing,tightfitting	1.0	228.174 Physical Facilities. Functio...	NaN	*29 Concentration of the sanitizing solution s...
1	7-11	Routine	10/03/2016	86	5123	LOVERS	W	LN	NaN	5123 W LOVERS LN	75209	*18 Chemical sanitizer generated onsite, devic...	3.0	228.111 Equipment, Utensils, and Linens. ...	NaN	*03 Food products not maintained at 135øF or a...
2	MI HONDURAS	Routine	10/03/2016	75	10818	DENNIS	NaN	RD	#101	10818 DENNIS RD #101	75229	*14 When to wash hands after handling soiled e...	3.0	228.38 Management and Personnel. (d)...	All hand sinks were blocked	*21 RFSM - Not On Site
3	TORTILLERIA LA ESPIGA	Routine	10/03/2016	82	1328	JIM MILLER	N	RD	#104	1328 N JIM MILLER RD #104	75217	*39 In-use utensils, between-use storage. Duri...	1.0	228.68 Food. Preventing Contaminatio...	cannot use water containers to keep utensils ...	*20 Grease Trap Tickets
	TMGM @				2323 N									228.35	missing	*22 Handlers-

Fig.1. Snapshot of the Dallas City Restaurants Inspection data loaded into a data frame

This dataset contains most of the information that will be needed for the project such as location information, street name, etc., However, this dataset contains mixed datatypes in all 114 columns and needs extensive cleaning before it can be used for the project.

```
Restaurant Name                object
Inspection Type                object
Inspection Date                object
Inspection Score                int64
Street Number                   int64
Street Name                    object
Street Direction               object
Street Type                    object
Street Unit                    object
Street Address                 object
Zip Code                       object
Violation Description - 1      object
Violation Points - 1          float64
Violation Detail - 1           object
Violation Memo - 1             object
Violation Description - 2      object
Violation Points - 2          float64
Violation Detail - 2           object
Violation Memo - 2             object
Violation Description - 3      object
Violation Points - 3          float64
Violation Detail - 3           object
Violation Memo - 3             object
Violation Description - 4      object
Violation Points - 4          float64
Violation Detail - 4           object
```

Fig.2. A snapshot of datatypes of the columns in the dataframe.

2. Ratings of Indian resturants for selected locality in Dallas City
   - Data source : FourSquare API
   - Description : By using this api we will get all the ratings for Indian restaurants in selected neighbourhood

2.1. Data Pre-processing

The loaded dataframe is further subjected to processing before it has been utilized. So in this regard, a new dataframe with only columns of interest were created. The columns that were dropped are street number, type, unit, address, and other violation based columns.

```
[ ] # Create a new dataframe with necessarry columns to work with
    dallas_mod_df = pd.DataFrame(columns=['Restaurant Name', 'Street Name','Zip Code', 'Inspection Type', 'Inspection Date',
           'Inspection Score', 'Inspection Month', 'Lat Long Location'], data=dallas_df)
    dallas_mod_df.head()
```

	Restaurant Name	Street Name	Zip Code	Inspection Type	Inspection Date	Inspection Score	Inspection Month	Lat Long Location
0	HARVEY'S	PRESTON	75230	Routine	10/03/2016	82	Oct 2016	12835 PRESTON RD #306\n(32.924235, -96.803626)
1	7-11	LOVERS	75209	Routine	10/03/2016	86	Oct 2016	5123 W LOVERS LN\n(32.851252, -96.823033)
2	MI HONDURAS	DENNIS	75229	Routine	10/03/2016	75	Oct 2016	10818 DENNIS RD #101\n(32.895847, -96.881391)
3	TORTILLERIA LA ESPIGA	JIM MILLER	75217	Routine	10/03/2016	82	Oct 2016	1328 N JIM MILLER RD #104\n(32.73556, -96.700079)
4	TMGM @ KEETON PARK GOLF COURSE	JIM MILLER	75227	Routine	10/03/2016	80	Oct 2016	2323 N JIM MILLER RD\n(32.756246, -96.701964)

Fig.3. A snapshot of the modified dataframe.

The shape of the modified dataframe is now reduced to 37875 rows and 8 columns from the original size of 37875 rows and 114 columns.

```
[ ] # Check the shape of the modified dataframe
    dallas_mod_df.shape
```

```
(37875, 8)
```

Fig.4. Size of the modified dataframe

The last column in the above dataframe contains latitude and longitude location. However, it has the address along with that. So a new dataframe is created for ease of handling.

```
[ ] # Create Latitude and Longitude Dataframe
    ll_df = pd.DataFrame(columns = ['Lat Long Location'], data=dallas_mod_df)
    ll_df.head()
```

	Lat Long Location
0	12835 PRESTON RD #306\n(32.924235, -96.803626)
1	5123 W LOVERS LN\n(32.851252, -96.823033)
2	10818 DENNIS RD #101\n(32.895847, -96.881391)
3	1328 N JIM MILLER RD #104\n(32.73556, -96.700079)
4	2323 N JIM MILLER RD\n(32.756246, -96.701964)

Fig.4. LatLong Location Column

This is further cleaned and split into two columns to get only latitude and longitude values.

	Latitude	Longitude
0	32.924235	-96.803626
1	32.851252	-96.823033
2	32.895847	-96.881391
3	32.73556	-96.700079
4	32.756246	-96.701964

Fig.5. Latitude and Longitude cleaned Column

The above dataframe is merged to the dallas_mod_df to get dallas_merged_df

```
[ ] dallas_merged_df = pd.concat([dallas_mod_df, ll_df], axis=1)
    dallas_merged_df.head()
```

	Restaurant Name	Street Name	Zip Code	Inspection Type	Inspection Date	Inspection Score	Inspection Month	Latitude	Longitude
0	HARVEY'S	PRESTON	75230	Routine	10/03/2016	82	Oct 2016	32.924235	-96.803626
1	7-11	LOVERS	75209	Routine	10/03/2016	86	Oct 2016	32.851252	-96.823033
2	MI HONDURAS	DENNIS	75229	Routine	10/03/2016	75	Oct 2016	32.895847	-96.881391
3	TORTILLERIA LA ESPIGA	JIM MILLER	75217	Routine	10/03/2016	82	Oct 2016	32.73556	-96.700079
4	TMGM @ KEETON PARK GOLF COURSE	JIM MILLER	75227	Routine	10/03/2016	80	Oct 2016	32.756246	-96.701964

Fig.6. Dallas merged dataframe with latitude and longitude values

The above dataframe contains most of the necessary columns and hence it will be utilized further. After dropping any null values the final information about the above datframe is.

```
[ ] dallas_merged_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 36554 entries, 0 to 37874
Data columns (total 9 columns):
Restaurant Name     36554 non-null object
Street Name         36554 non-null object
Zip Code            36554 non-null object
Inspection Type     36554 non-null object
Inspection Date     36554 non-null object
Inspection Score    36554 non-null int64
Inspection Month    36554 non-null object
Latitude            36554 non-null object
Longitude           36554 non-null object
dtypes: int64(1), object(8)
memory usage: 2.8+ MB
```

Fig.7. Information about Merged dataframe

The risk categories are explicitly define in the inspection data. Hence, a new column named risk is created in the above dataframe. by binning the inspection scores into following categories using the guidelines in http://www2.dallascityhall.com/FoodInspection/SearchScores.cfm.

Three categories were created for ease of classification:

a. High Risk - Inpection Score between -5 to 60
b. Medium Risk - Inpection Score between 60 to 80
c. Low Risk - Inspection Score between 80 to 100

```
[ ]  # Create risk category bins

     bins = [-5, 60, 80, 100]
     labels =["High Risk","Medium Risk","Low Risk"]
     dallas_merged_df['Risk'] = pd.cut(dallas_merged_df['Inspection Score'], bins,labels=labels)
     dallas_merged_df.head()
```

	Restaurant Name	Street Name	Zip Code	Inspection Type	Inspection Date	Inspection Score	Inspection Month	Latitude	Longitude	Risk
0	HARVEY'S	PRESTON	75230	Routine	10/03/2016	82	Oct 2016	32.924235	-96.803626	Low Risk
1	7-11	LOVERS	75209	Routine	10/03/2016	86	Oct 2016	32.851252	-96.823033	Low Risk
2	MI HONDURAS	DENNIS	75229	Routine	10/03/2016	75	Oct 2016	32.895847	-96.881391	Medium Risk
3	TORTILLERIA LA ESPIGA	JIM MILLER	75217	Routine	10/03/2016	82	Oct 2016	32.73556	-96.700079	Low Risk
4	TMGM @ KEETON PARK GOLF COURSE	JIM MILLER	75227	Routine	10/03/2016	80	Oct 2016	32.756246	-96.701964	Medium Risk

The Inpsection month category column contains both month and year let's split it up further so that it can be used later.

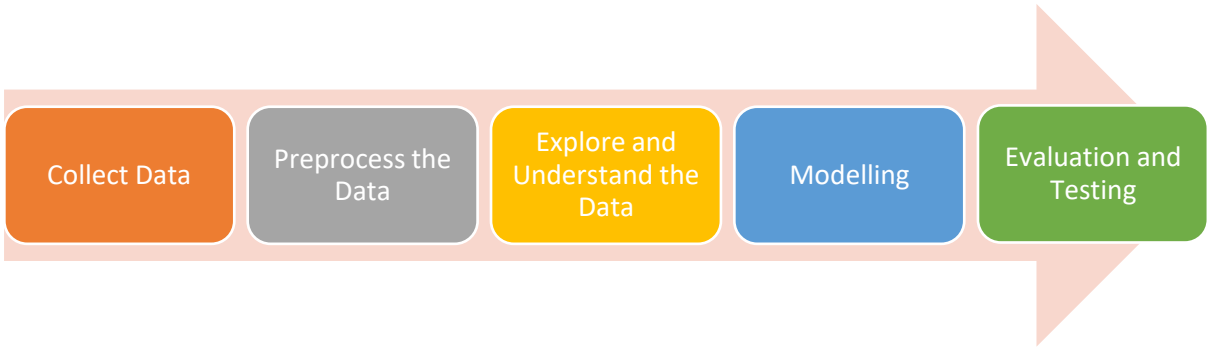
Fig.7. Dataframe after creating risk category column

Now after further cleaning and some type conversion, the final dataframe named dallas_merged_df looks like this.

```
Data columns (total 11 columns):
Restaurant Name     36553 non-null object
Street Name         36553 non-null object
Zip Code            36553 non-null int64
Inspection Type     36553 non-null object
Inspection Date     36553 non-null datetime64[ns]
Inspection Score    36553 non-null int64
Latitude            36553 non-null float64
Longitude           36553 non-null float64
Risk                36553 non-null category
Inspection Month    36553 non-null object
Inspection Year     36553 non-null int64
dtypes: category(1), datetime64[ns](1), float64(2), int64(3), object(4)
memory usage: 3.1+ MB
```

Fig.8. Information about Processed dataframe

3.Methodology



4. Exploratory Data Analysis

3.1.1. Overall performance of Restaurants based on risk category for the period 2016-2019

As we have binned the restaurants that were inspected during the period of 2016-2019 into three categories based on their inspection scores it can be easily visualized the percentage of restaurants in each risk categories.

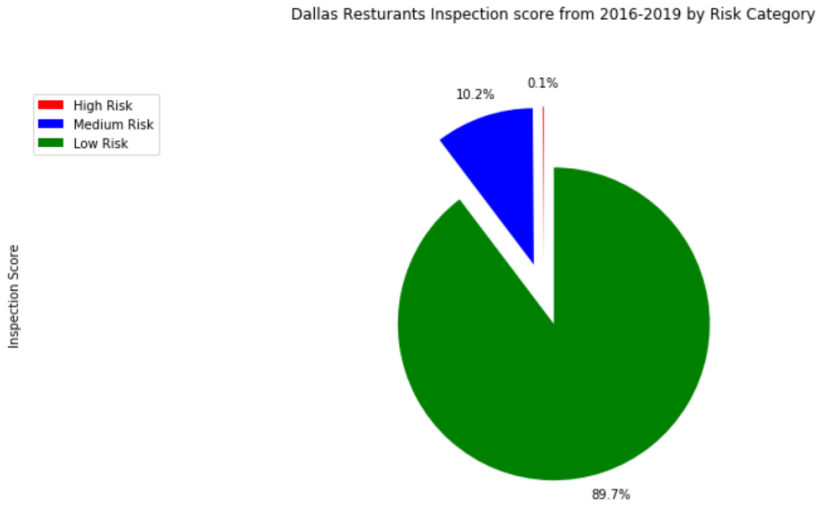


Fig.8. Dallas Restaurants Inspection Score from 2016-2019 by Risk Category

It can be seen from Fig.8. that almost 90% of restaurants that were inspected from 2016-2019 have been placed under low risk category. This shows either the Dallas City officials are very strict or the retsuarants maintain a very high standard!!.

The count of restaurants in each category is given as a bar plot (Fig.9). It is inferred from the same that 32778 restaurants were classified as low risk, 3734 as medium risk and only 41 were classified as high risk.

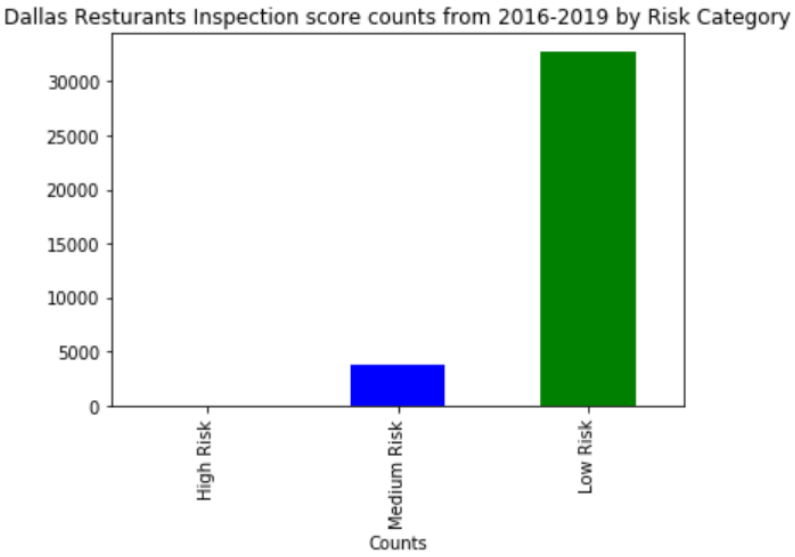


Fig.9. Dallas Restaurants Inspection Score counts from 2016-2019 by Risk Category

Since the overall statistics of restaurants that were placed in each risk category in the period 2016-2019 did convey some meaning, it will be more insightful if we gain some information on the percentage of restaurants that were places in each risk category for individual years. The percentage of restaurants in each risk category for individual years is shown in Fig.10. It is understood from the same that almost 90% of the restaurants were placed in low risk category in each year.

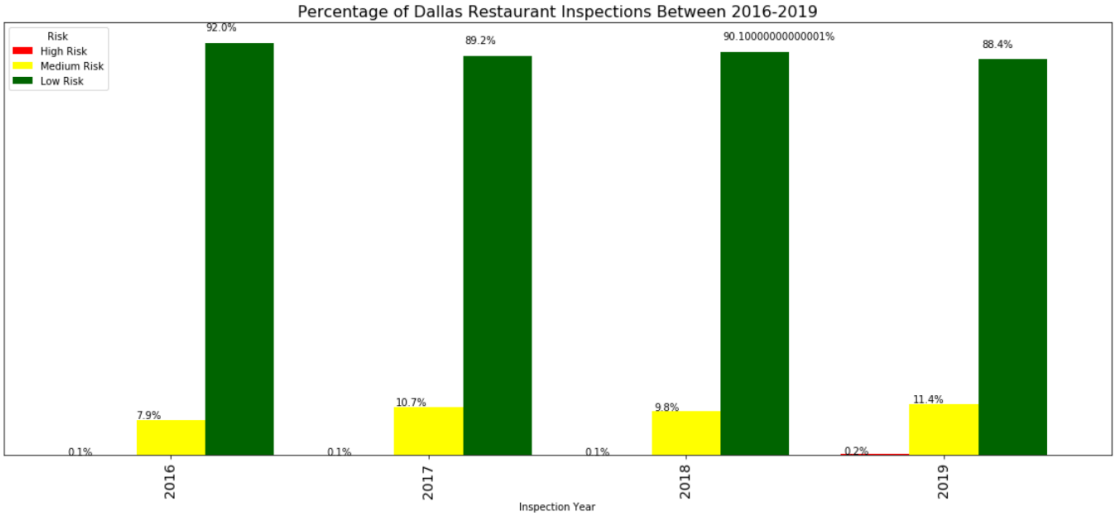


Fig.10. Percentage of Dallas Restaurants Inspections from 2016-2019

The inspection score distribution of the restaurants for each year is shown in Fig.11. It was inferred from the same that most of the restaurants scored above 80 during the inspections in each year.

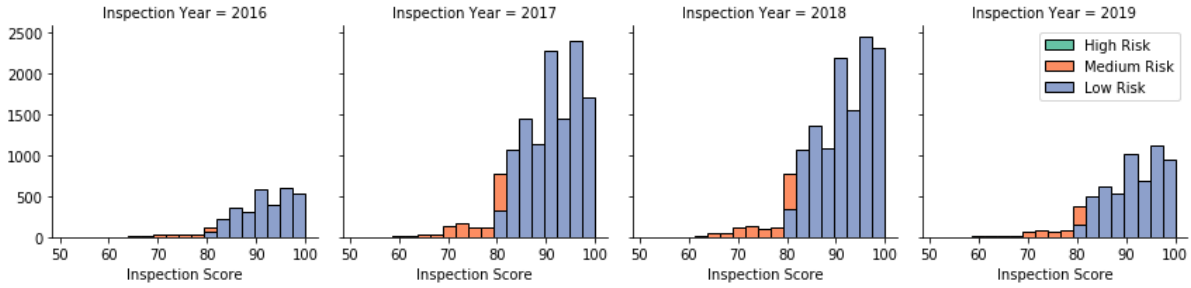


Fig.11. Dallas Restaurants inspection score distribution from 2016-2019

The restaurants will be better prepared if they know in prior on which day, they can expect an inspection. The day in which the restaurants were inspected in each year from 2016-2019 is depicted in Fig.12.

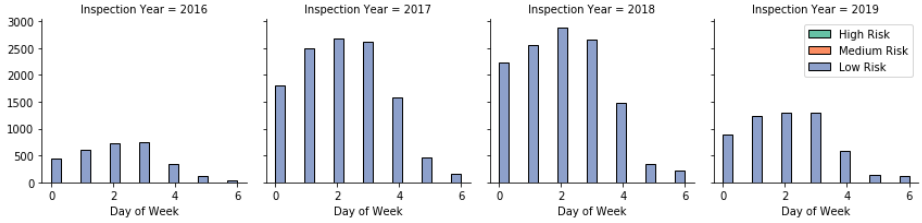


Fig.12. Dallas Restaurants inspection conducted days from 2016-2019

It is inferred from Fig.12. that most of the restaurants were inspected during the weekend. Hence, the restaurants have to be on their toes to cater to the needs of both the customer as well as the inspection officer.

3.2. Overall performance of Indian Restaurants based on risk category for the period 2016-2019

As this project focusses on Indian restaurants in Dallas city, the whole dataframe have to be trimmed to have only Indian restaurants. A simple google search revealed that most of the Indian resturants in Dallas have the keywords such as Shiva, India, Masala, Spice, Mumbai, etc., in their name. Hence the dataframe containing only Indian restaurants were created and named as Indian_rest_df (Fig.13).

```
indian_rest_df = pd.DataFrame(indian_score)
indian_rest_df.reset_index(drop=True, inplace=True)
indian_rest_df.head()
```

	Restaurant Name	Street Name	Zip Code	Inspection Type	Inspection Date	Inspection Score	Latitude	Longitude	Risk	Inspection Month	Inspection Year	Day of Week
0	INDIA PALACE RESTAURANT	PRESTON	75230	Routine	2016-10-12	81	32.923460	-96.803630	Low Risk	Oct	2016	2
1	KALACHANDJI'S	GURLEY	75223	Routine	2016-10-18	95	32.794271	-96.750159	Low Risk	Oct	2016	1
2	INDIA CHAAT CAFE	PRESTON	75252	Routine	2016-12-13	90	32.998852	-96.797507	Low Risk	Dec	2016	1
3	SHIVAS BAR & GRILL	GREENVILLE	75206	Routine	2017-01-09	85	37.531860	-77.471669	Low Risk	Jan	2017	0
4	MUMBAI GRILL	PRESTON	75252	Routine	2017-01-18	89	32.989660	-96.801854	Low Risk	Jan	2017	2

Fig.12. Snapshot of Indian restaurants dataframe that were inspected from 2016-2019

The inspection score received by the Indian restaurants in the same period are given in Fig.14. It is inferred from the same that during the year 2018 few restaurants were placed in Medium risk category.

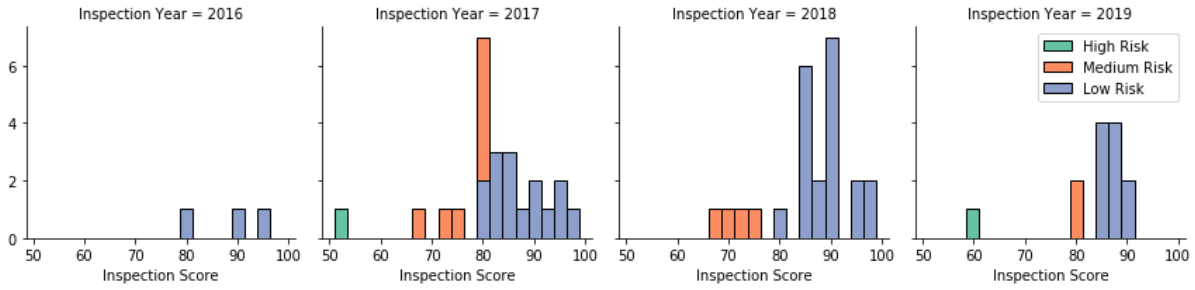


Fig.14. Indian Restaurants inspection score distribution from 2016-2019

The overall percentage of Indian restaurants in each category for the period 2016-2019 is shown in Fig. 15.

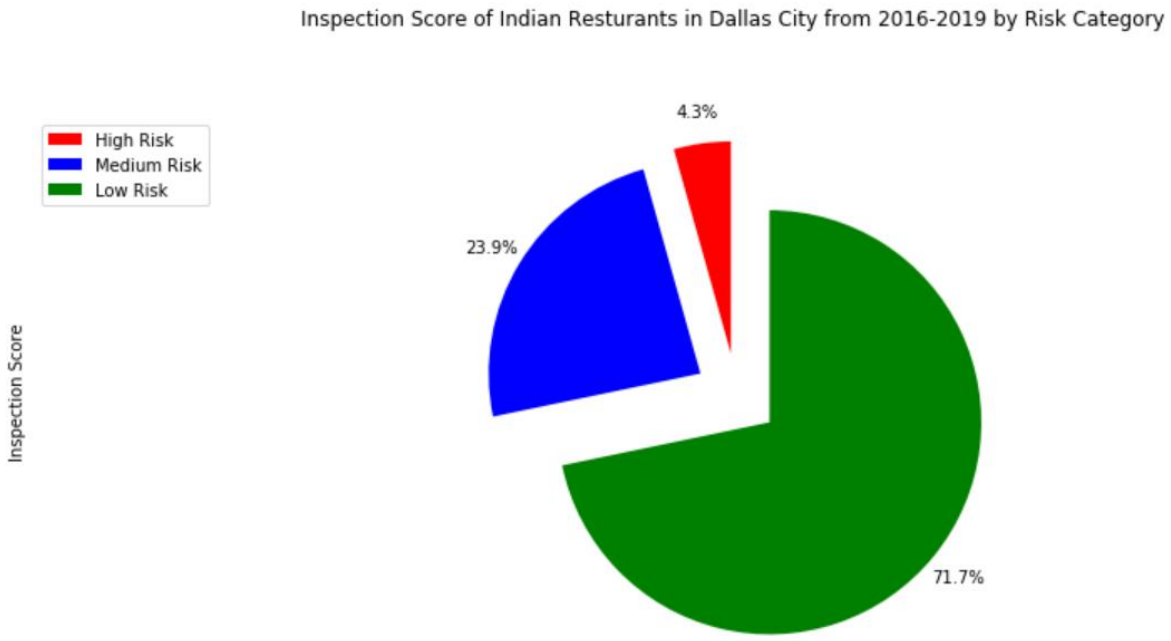


Fig.15. Indian Restaurants Inspection Score from 2016-2019 by Risk Category

The overall percentage of Indian restaurants that were placed in Low risk category is ~72%. This in comparison with the overall percentage of 90 is fairly low. Hence, the Indian restaurants must improve their service quality to match up to the overall performance.

Word Clouds (also known as wordle, word collage or tag cloud) are visual representations of words that give greater prominence to words that appear more frequently. This type of visualization highlights the most common words and present the data in a way that everyone can understand. The violation description of Indian restaurants from 2016-2019 in the form of a word cloud is shown in Fig.16.

Fig.14. Word cloud of violation descriptions of Indian restaurants from 2016-2019

The word cloud shows that most violation descriptions have the words date, food, temp, and marking appears more frequently. This may be due to old food, food not marked and the temperature of the food may be cold.

4. Predictive Modelling

There are two types of models, regression and classification, that can be used to predict the restaurant performance. In the present context, only classification models will be used as they predict the probabilities of the risk categories that the restaurants will be placed.

4.1. Classification models

Since the original dataframe contains 36553 rowas and 11 columns, it is not suitable for predictive modelling. Hence, in the present study the predictive modelling was carried out for those restaurants that were inspected during 2016 only. A new dataframe was created in this regard and named as dallas_2016_df which contains 2974 entries which is sufficient. After necessary one hot encoding and dropping unnecessary columns, the feature correlation between each column was analysed. The corresponding pearson correlation plot is shown in Fig.15.

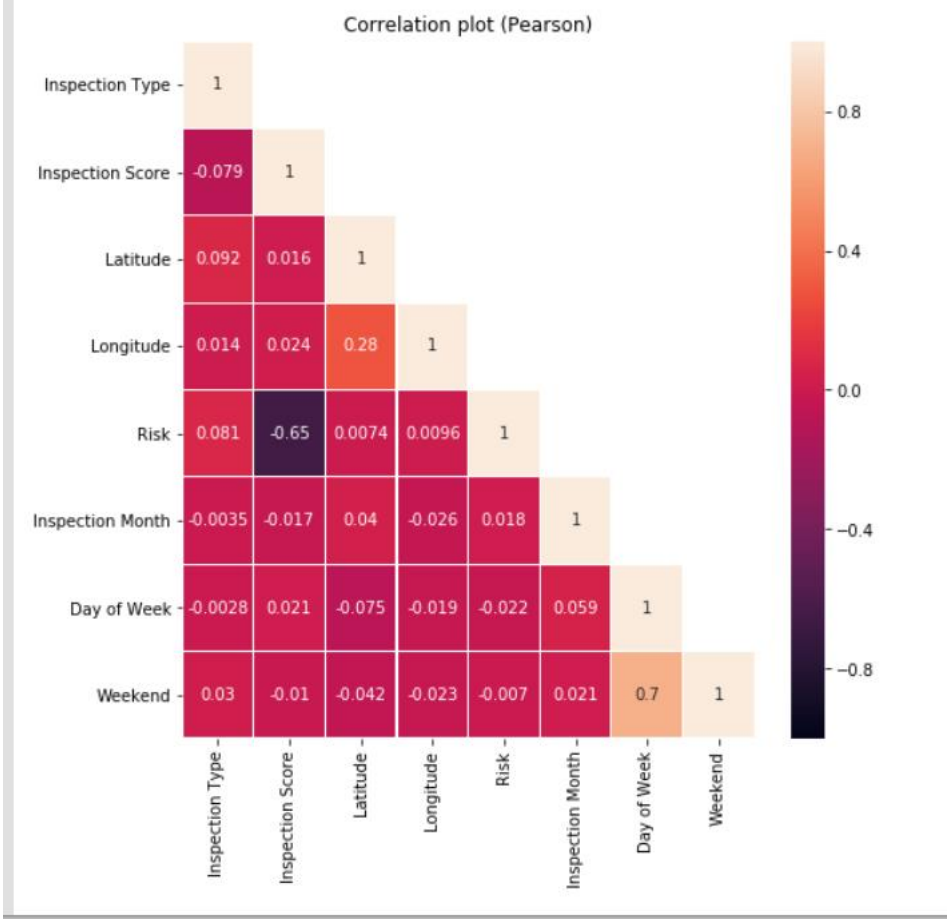


Fig.15. Person correlation plot for the features that will be used in the modelling.

It is inferred from Fig.15 that the day of week and the weekend is highly correlated and the inspection score is negatively correlated with the risk.

Now the dataframe is ready to apply the machine learning algorithm. First, the Decision tree classifier was employed with minimal hyperparameters to predict the risk category and the decision tree classifier employed is shown in Fig. 16.

```
[ ]  rest_tree = DecisionTreeClassifier(criterion="entropy", max_depth = 10, max_leaf_nodes=5)
     rest_tree # it shows the default parameters
```

```
DecisionTreeClassifier(class_weight=None, criterion='entropy', max_depth=10,
                       max_features=None, max_leaf_nodes=5,
                       min_impurity_decrease=0.0, min_impurity_split=None,
                       min_samples_leaf=1, min_samples_split=2,
                       min_weight_fraction_leaf=0.0, presort=False,
                       random_state=None, splitter='best')
```

Fig.16. Default parameters employed in the Decision Tree classifier model.

Then, a random forest model is employed to further improve the model and the default parameters employed are shown in Fig.17.

```
[ ]  # Build a random forest
     rf_tree = RandomForestClassifier(n_estimators = 1000, random_state = 42)
```

Fig.17. Default parameters employed in the Random Forest classifier model.

Usually, the decision tree classifier is prone to overfitting. Hence, the random forest model is employed here.

In order to evaluate the models employed, the accuracy scores were estimated. As inferred from the Table1. Both models have similar high accuracy scores and may be sufficient to predict the given data.

Table.1. Accuracy scores from the classification models.

Machine Learning Algorithm	Accuracy Score
Decision Tree Classifier	**0.9988801791713325**
Random Forest Classifier	**0.9988801791713325**

A sample decision tree from the ransom forest model is shown in Fig. 18.

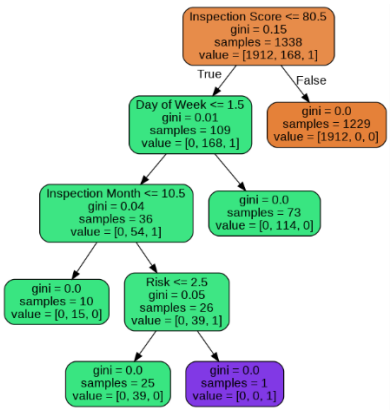


Fig.17. A sample tree from the Random Forest classifier model.

5. Visualization using Folium

In order, to visualize the Indian restaurants in dallas city, folium package was used. For this purpose, Indian_rest_df was used and the Indian restaurants are shown as markers in the Fig. 18.

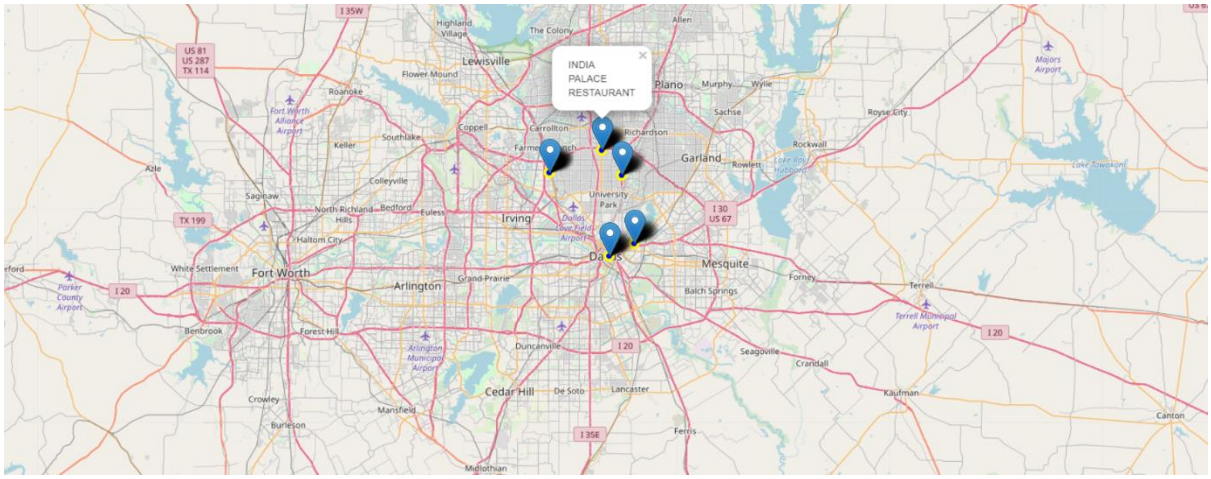


Fig.18. Location of Indian restaurants in Dallas.

6. Ratings of Indian Restaurants using FourSquare API

In order to evaluate the ratings of Indian restaurants in Dallas City, FourSquare API was employed. For this purpose, the locations of Indian restaurants from Indian_rest_df itself was employed to get the nearby venues using FourSquare API by employing getNearbyVenues() function. And from this the Indian restaurants were filtered to get their ID to get their ratings using venue_ratings() and get_venue_details() function.

```
[ ]  def get_Venue_Details(venue_id):
       ratings_list = []

       # Create the API request URL
       url = 'https://api.foursquare.com/v2/venues/{}?&client_id={}&client_secret={}&v={}'.format(
                 venue_id,
                 CLIENT_ID,
                 CLIENT_SECRET,
                 VERSION)
       # make the GET request
       results = requests.get(url).json()
       print(results)
       return(results)
```

Fig.19. get_Venue_Details() function.

```
[ ]  ratings_list=[]
     def venue_ratings():
       for item in id_list:
         rating_details=get_Venue_Details(item)
         venue_data=rating_details['response']
         try:
             venue_id=venue_data['venue']['id']
             venue_name=venue_data['venue']['name']
             venue_likes=venue_data['venue']['likes']['count']
             venue_rating=venue_data['venue']['rating']
             venue_tips=venue_data['venue']['tips']['count']
             ratings_list.append([venue_id,venue_name,venue_likes,venue_rating,venue_tips])
         except KeyError:
           pass
         column_names=['Venue ID','Venue Name','Venue Likes','Venue Rating','Venue Tips']
         df = pd.DataFrame(ratings_list,columns=column_names)
       return(df)
```

Fig.19. venue_ratings() function

Since, the original Indian_rest_df contains restaurants data from 2016-2019. They have duplicate entries. So using the unique IDs of the restaurants, the top five restaurants are sorted on the basis of the number of likes and other ranking values they received.

	Venue ID	Venue Name	Venue Likes	Venue Rating	Venue Tips
0	4a7b7247f964a52010eb1fe3	India Palace	79	8.4	30
1	4b5cf902f964a5203b4d29e3	India Chaat Cafe	61	8.1	19
2	4d94bab574c8236a5956c4fc	Vindu Indian Cuisine	17	6.8	14
3	4b37e960f964a520704825e3	Taj Mahal Indian Restaurant & Bar	23	7.2	17
4	4d3ddd4aa2e4b1f764d0f525	Al-Kabob Grill & Cafe	9	7.6	7

Fig.19. The top 5 Indian restaurants in Dallas City based on the number of likes

It can be seen from Fig. 19 that India Palace is the best Indian restaurant to dine in Dallas city if you like Indian food. The number of likes received by the Indian restaurants in Dallas are shown as bar plot in Fig. 20.

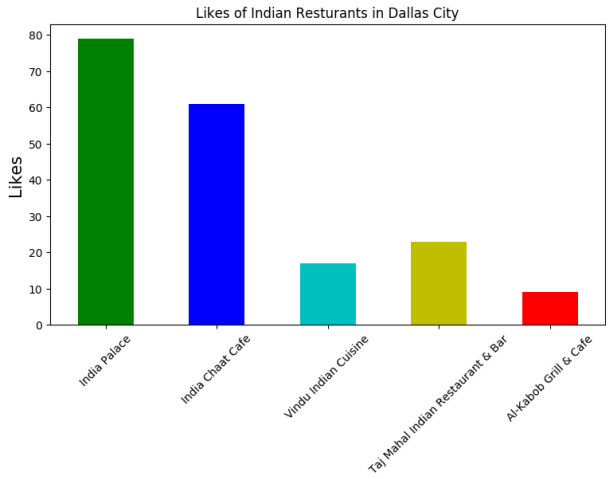


Fig. 20. The number of likes received by the top five Indian restaurants in Dallas City.

7. Conclusions

This project successfully completes my IBM Data Science Professional Certification Training. I am quite new to the data science and I had a steep learning curve during the course. I have really enjoyed doing all the lab exercise and the courses were really informative. The following are the conclusions that I derive from this project:

1. Dallas City have only very few Indian restaurants. Hence, it has a potential market for opening a new Indian restaurant
2. Roughly 80% of Indian restaurants that are currently present in Dallas City are placed in low risk category based on the inspection data from 2016-2019
3. A decision tree classifier model is built for classifying the restaurants into various risk categories and the model performs well for the given data set. This will help the restaurants in predicting their risk category for a given year.
4. The Indian restaurants in the Dallas City were visualized using the folium map rendering library
5. Using FourSquare API, the venue details for the Indian restaurants were analyzed and found that among all the restaurants in Dallas City India Palace is the best place to dine.

8. Limitations and Future Work

1. The restaurants are ranked solely on the data provided by FourSquare API. If data on other demographics are available this can be improved
2. The accuracy of location data depends on Dallas City Inspection Data and FourSquare API. Hence, need to be analyzed further as there are some ambiguous entries.
3. The machine learning model will be further improved as the model developed may be prone to over-fitting