Analysis on Bangalore food culture using Zomato dataset

1st Madhusmita Kalita
Master's in Computer Science
Lakehead University
ID:1115467
mkalita@lakeheadu.ca
Section: FB

2nd Mina Armanyous
Master's in Computer Science
Lakehead University
ID: 1117659
mmakram@lakeheadu.ca
Section: FB

3rd Mahesh Dhaka
Master's in Computer Science
Lakehead University
ID: 1099371
mdhaka@lakeheadu.ca
Section: FB

***Abstract*—**

A. BACKGROUND

The food culture in Bangalore, India is really fascinating. A plethora of restaurants from all over the world can be found here. Recent years has seen a tremendous growth in the use of online platforms like Zomato to assess a restaurant's food, ambiance or quality of service. Numerous factors like these add to the overall "Ratings" of a restaurant. Hence, a good customer rating is very crucial for both existing and new restaurants to sustain in today's tough market. Our solution aims at classifying the restaurants in Bangalore into 3 major categories – 'Highly rated', 'Mediumly rated' and 'Lowly rated' through the use of Machine learning techniques.

B. OBJECTIVE

The intuition behind analyzing the dataset from Zomato is to infer knowledge about some very important factors that impact the overall operations and future growth of a restaurant through their ability to maintain a good 'Rating'. The proposed solution uses some of the Machine Learning classification algorithms like – Random Forest, Decision Tree, Support Vector Machines(SVM), etc. to segregate restaurants into the aforementioned categories. The optimal solution is presented to the users by evaluating all the classifiers through metrics like Accuracy, Precision score, f1-score. Our solution also allows a new restaurant to predict their probable 'rating' category by using some of the important features like location of the establishment, type of restaurant,book table etc

C. Methods

In order to assess the 'Ratings' of restaurants and categorize them into the aforementioned classes, we leveraged the dataset of 'Zomato' available on Kaggle. In order to propose an optimal solution, we evaluated 6 different classification algorithms viz. Support Vector Machines (SVM), Naïve byes, Decision tree, Random forest ,logistic regression & k-Nearest neighbour(kNN) algorithm. Several visualizations have also been presented to provide with exploratory data analysis.

D. Results

Out of the six trained Machine learning models, Random forest outperformed with a best accuracy of 81%. The evaluation metrics was based on 4 metrics: accuracy, precision, recall and f1-score. Each of these evaluation techniques is important as one single method cannot decide the performance of a classifier. The reason for having this combination is to not judge a model by its accuracy only because at times although the accuracy may be high, it does not necessarily mean that the predictive power will be higher.

E. Conclusion

Our solution serves as a platform to understand the classification of restaurants as High/Average/Low rated. It also provides ways to understand importance of some of the crucial features, that may be helpful for any existing or new establishment

I. Introduction

A. PROBlEM DEFINATION

The objective of this Project is to understand the impact of factors like –location, type of a restaurant, basic cost and popularity on a Restaurant's ratings and reviews. The project will also focus on analyzing these features and try to find some hidden patterns through visualizations. Furthermore, the aim would be to train a classification model to classify restaurants as 'Highly rated', 'Average rated' and 'Low rated'. Once we have achieved a quality model (with good accuracy etc.), we will be using it to predict the same on new restaurants.

B. SIGNIFICANCE OF THE PROBLEM

Understanding how an existing or new restaurant can aim for higher customer satisfaction(ratings) by focusing on the factors such as – location, type of a restaurant, basic cost and popularity.

C. LITERATURE REVIEW

Online food delivery services are a growing trend all across the world. With the emergence of smartphones and the internet at our fingertips, life has become easier and more convenient. Especially for people who are working most of the time and don't have enough time to prepare their own meals, it has become a challenge for them to manage time for all of this. This is where online food delivery services come in and make their life easier, especially in a city that is called the IT hub of India known as Bangalore.

(Saxena, 2019) said that the Indian online food market is worth about 350billion dollar and recent growth in mobile technology has helped them grow their business to various markets in the country. The paper mentions that due to the increase of young generations working in big metro cities and having a hectic work culture makes these kinds of ideas more feasible and beneficial. It states that customers are leaning more towards the online food ordering apps like Zomato and Swiggy because it requires very less human intervention and that increases privacy for the customers. It is also opined that people tend to lean towards such services due to the ease of handling such mobile applications and usefulness and innovativeness against information technology in recent years.

(Gupta, 2019) states that the food delivery market's penetration broke 30 percent in 2016, and they believe it will still mature and go until 65 percent in recent years. And Asian countries having a larger population on average is a good market to run such a business and can be highly profitable.

Some in-depth details about the market in India were also talked about where it is mentioned that the market size for India is expected to reach about 42 lakh crores by the year 2020. And ideas, like providing people with health consciousness an option for healthier food and targeting different sections of the population, are the hot trends in the food business. Options like loyalty points, flexible payments and door to door delivery are said to be some of the various reasons these businesses are blooming and with smartphones and computers growing day by day, and more people having access to these services, it is expected to grow at a linear rate.

D. PREVIOUS WORK

The dataset that was chosen for this project, is a very recent dataset published around April 2019. This dataset has been used in an article published on a website where they use different python plotting libraries to visualize the data and find relationships between the features and do some analyzing based on what is visible from the graphs.

Most of the online resources that are using this dataset or similar dataset are representing the visual aspects of the dataset and doesn't follow many data science techniques when it comes to analyzing. So, for that, we can take a different approach and use algorithms to predict and cluster parts of the dataset and see how the results vary from the analyzed results that are mentioned in those articles. In one of the cases, they have even removed all the instances and features that has missing values and made predictions based on the complete values, so that can be one opportunity for us to take a different approach and compare results. Most of the papers published on this topic of online food delivery systems are talking about business statistics and the scope of such innovations and ideas. It seems like this is a hot topic for entrepreneurs and start-ups for which most of the topic of discussion is on market availability and finding newer opportunities.

So, this project will also be doing the analyzing part that others have done but after some data science techniques have been used to reach a conclusion.

E. Our Achievement

We are able to achieve around 81 % accuracy with random forest classification, with 5 fold cross validation method. We are also able to achieve more than 70% accuracy, precision, F1-score, recall in all the other methods i.e Support vector machine, Naïve Bayes, logistic regression ,decision tree & KNN. For unsupervised learning we used K-means algorithm, for which we received 0.39 Silhouette width.

II. Data

A. DATA SOURCE

https://www.kaggle.com/himanshupoddar/zomato-bangalore-restaurants

B. DATA DESCRIPTION

The “Zomato” dataset is an extract of the data from Zomato app for Bangalore restaurants. Zomato is an aggregator and food delivery web application in India. It provides information such as location, cuisine type, menus etc. and have an ability to rate and write reviews for restaurants. Bangalore being the hub of India’s IT Industry and a city with 12000 restaurants people are dependent on restaurant foods more than home cooked food due to busy schedule. Every cuisine such as Japanese, US, Chinese, European, Russia to Antarctica all kinds are available in this city. The data set is a representation of the restaurant food culture of Bangalore. It contains 51.7 K instances with 17 features

C. FEATURES

- URL - Contains the URL of the restaurant’s website. It contains total of 51717 unique value. Feature type is URL
- address - Contains the address of the restaurant. It contains 11495 unique value
- name - Contains the name of the restaurant It contains 8792 unique values. The data type is String.
- online order -This feature is a categorical feature which contains whether the online order is available or not. The data type is Boolean have values “Yes” or “No”.
- book table - This is a categorical feature, which says booking option available or not. The data type is Boolean have values “Yes” or “No”.
- rate - Contains overall rating of the restaurants out of 5. The datatype is “String”. It has a total 43.9 K records.
- phone - contains the phone number of the restaurant. The data type is String. It has total 14923 unique values.
- Location - contains the neighborhood in which the restaurant is located. The feature has only 21 missing value with 93 unique value
- rest type - contains the restaurant type such as casual dining, café etc. the data type of the feature is String. The feature has 227 missing values with 93 unique values. The most common being “Quick Bite”
- Dish liked - contains dishes people liked in the restaurant. It has 54 percent missing values and 5271 unique values. Most common being “Biriyani”.
- cuisines: contains the food style like Chinese, north Indian etc. it has 45 missing values with 2723 unique values. Most common being North Indian.
- approx cost (for two people)- contains the approximate cost of a meal for two people. The data type is Decimal. And it has 346 missing value
- reviews list - containing reviews for the restaurant, each instance consists of two values, rating and review by the customer. The data type is string. It has total 22.5 K unique values. Most common being “[]”.
- menu item - contains the list of menus available in the restaurant. The data type is string. It has total 9098 unique values. Most common being “[]”.
- listed in(type): contains the type of the meal like delivery, dine in etc. The data type is string. It has total 7 unique values. Most common being “Delivery”.
- listed in(city): contains the neighborhood in which the restaurant is listed. The data type is string. It has total 30 unique values. Most common being “BTM”.

D. Missing Values

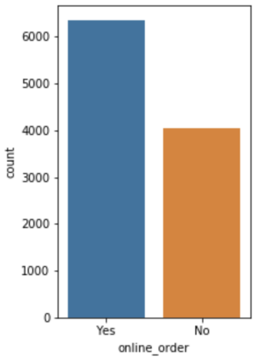
Feature name	**Count**
rate	7775(15%)
dish_liked	28.1k(54%)
rest_type	227
cuisines	45
approx_cost (for two people)	346
location	21
phone	1208(2%)

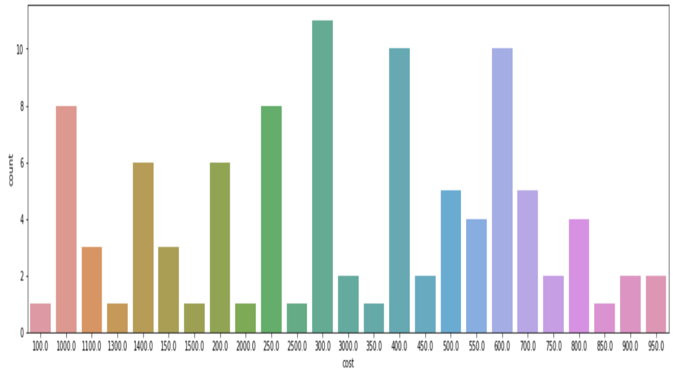
E. CLASS INFORMATION

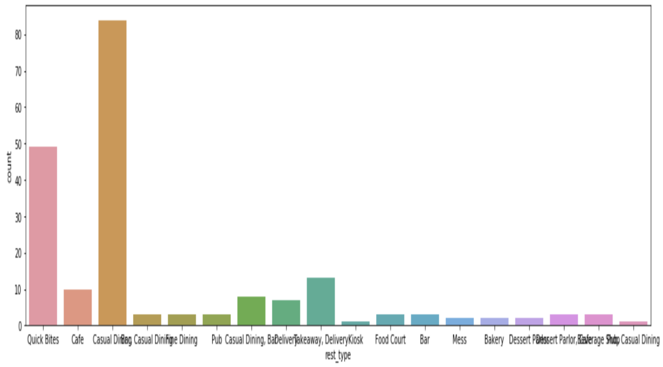
The Machine learning model will classify all our test and new dataset into the following classes

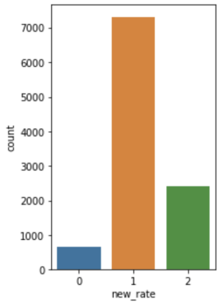
- Low rated -This class will contain all the restaurants which are rated below 3.00/5.00
- Average rated - This class will contain all restaurants which are rated in the range of 3.00-4.00/5.00
- High rated - This class will contain all restaurants which are rated above 4.00/5.00

F. Graphical representation for some important features









III. Methods

A. Data Pre-processing

- Irrelevant features removal - With our domain knowledge we figured out some features are irrelevant to rating prediction. Such as URL of a restaurant does not have anything to do with rating. Also, some features contain repetitive information such as location and listed_in(city) both contains the location or neighborhood of the restaurant. Feature "dish_liked" have 54 percent missing value hence this column is not considered as more than half of the data have missing data. "reviews_list' 'menu_item ' both this feature contains text hence we have not considered these attribute as well. In future this can be extended using Natural language processing techniques to analyze. Below features have been removed:
  -url
  -phone
  -listed_in(city)
  -listed_in(type)
  -dish_liked
  -reviews_list
  -menu_item

- Missing Values- Deleting all the instances which have missing rates, have value as 'New'. After this we have below number of missing values

  rest type – 149
  approx_cost – 247
  cuisines -11

  As the above numbers are not that significant, we have imputed the missing value as below

  rest_type – The missing values are imputed with the mode, which is "quick bytes"
  approx_cost - The missing values are imputed with the mean.
  cuisines - The missing values are imputed with the mode, which is "North Indian"

- Changing the data type for vote and Cost from string to numeric. Parsing the Rate value to extract the actual value (i.e. it had the format "4.1/5" which should be only 4.1). New columns is added to represent the class of rate and it has the following classes:

  - Low rated (Class: 0)
  - Average rated (Class: 1)
  - High rated (Class: 2)

- Data correction - In our data set, same restaurant with the same location has multiple instances. So, we have grouped each resturant and considered the mean of rate and vote for a restaurant for each location. A restaurant having a same name, but different location is considered as two different instances. This is done so that classification is accurate and every restaurant with a location have only 1 rating.

- Final number of instances: 10386

B. FEATURE SELECTION

For feature selection we have used SFFS (Sequential floating forward selection) algorithm. This algorithm is an extension of LRS with flexible backtracking capabilities. Rather than fixing the values of L and R, these floating methods allow those values to be determined from the data. The dimensionality of the subset during the search can be thought to be "floating", up and down. In SFFS after each forward step, SFFS performs backward steps as long as the objective function increases. We have used Random forest algorithm and K-fold Cross validation with SFFS approach to get the subset of features. We have received the below attributes to be most important

- location
- book_table
- rest_type
- cuisines
- votes

C. Scaling and Encoding feature

As we have features with different scales such as cost and votes, we used standard scaler for normalizing values which helped in getting correct accuracy in the different used models. Also, label encoding have been used as many of the used methods libaries (Sklearn) can only perform with numerical columns. Thus label encoding has been used for string data type columns such as rest type and cuisines.

D. Supervised Learning methods (Classification)

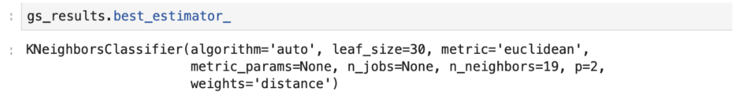
The problem of this project is a classification problem, as we are trying to classify restaurant into 3 classes Low rated, average rated, high rated. Below algorithms have been used to perform the same -

1) Naïve Bayes: A Naive Bayes classifier is a probabilistic machine learning model that's used for classification task. The crux of the classifier is based on the Bayes theorem. It is fast and easy to implement but its biggest disadvantage is that the requirement of predictors to be independent. In most of the real-life cases, the predictors are dependent, this hinders the performance of the classifier. We used Gaussian naïve Bayes because it assumes that features follow a normal distribution. We first split the testing and training samples and classify by Naïve Bayes algorithm. We calculated Accuracy, Precision, Recall, f1_score to evaluate performance of the model.

2) Support Vector Machine:: The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space (N — the number of features) that distinctly classifies the data points. Hyperplanes are decision boundaries that help classify the data points. Data points falling on either side of the hyperplane can be attributed to different classes. We used radial basis function kernel (rbf) SVM model and calculated Accuracy, Precision, Recall, f1_score to evaluate performance of the model.

3) Logistic Regression: Logistic Regression is a 'Statistical Learning' technique and one of the most popular ways to fit models for categorical data, especially for binary response data in Data Modeling. It is the most important (and probably most used) member of a class of models called generalized linear models. logistics regression majorly makes predictions to handle problems which require a probability estimate as output, in the form of 0/1. Logistic regression is an extremely efficient mechanism for calculating probabilities.
We have used maximum 100 iteration in the model and calculated Accuracy, Precision, Recall, f1_score to evaluate performance of the model

4) KNN: Lazy learning method that predicts the data class based on the nearest k-neighbors. KNN can be used for classification — the output is a class membership (predicts a class — a discrete value). An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors. We split the dataset in testing and training and used 5 neighbors(K=5) for the model. Also, hyper-parameter tuning with K-fold cross validation(K=5) has been used and it gave the following parameter set:



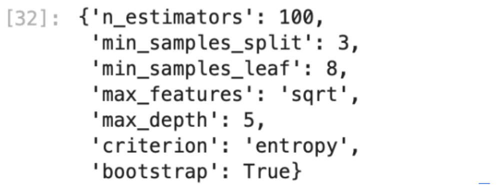
```
: gs_results.best_estimator_

: KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='euclidean',
                       metric_params=None, n_jobs=None, n_neighbors=19, p=2,
                       weights='distance')
```

We calculated Accuracy, Precision, Recall, f1_score to evaluate our performance in both the ways.

5) Decision tree: A decision tree is a decision support tool that uses a tree-like model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility. A decision tree is a flowchart-like structure in which each internal node represents a "test" on an attribute, each branch represents the outcome of the test, and each leaf node represents a class label. The paths from root to leaf represent classification rules. We used 'entropy' as criterion for our model and we calculated Accuracy, Precision, Recall, f1_score to evaluate our performance

6) Random Forest: Random forest, like its name implies, consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model's prediction. We split the dataset in testing and training sample and used the below parameters.

Also, hyper-parameter tuning with K-fold cross validation has been used and it gave the following parameter set:

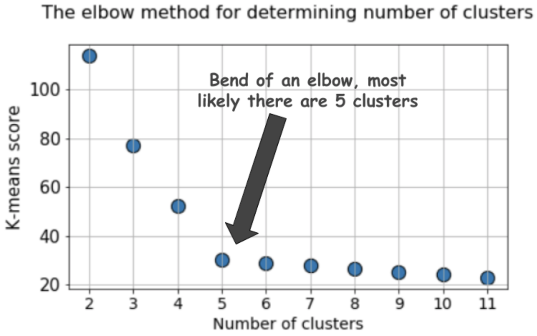


```
[32]: {'n_estimators': 100,
        'min_samples_split': 3,
        'min_samples_leaf': 8,
        'max_features': 'sqrt',
        'max_depth': 5,
        'criterion': 'entropy',
        'bootstrap': True}
```

E. *Unsupervised Learning methods*

We wanted to analyze our dataset in unsupervised learning methods and hence used K-means clustering algorithm. K-means tries to find homogeneous subgroups within the data such that data points in each cluster are as similar as possible according to a similarity measure such as Euclidean-based distance or correlation-based distance. For calculating the number of clusters, we used Elbow method.

- The Elbow Method: It involves running the algorithm multiple times over a loop, with an increasing number of cluster choice and then plotting a clustering score as a function of the number of clusters.



F. *Validation Methods*

1) Train_Test Split approach: In this approach randomly split the complete data into training and test sets. Then Perform the model training on the training set and use the test set for validation purpose. We have taken 25% as testing data and used for all the classification methods

2) K-fold cross validation: We used a K-fold cross validation technique to validate our training dataset for KNN & Random Forest methods. The reason for choosing a K-fold technique is to use all of our training data and make sure the Machine learning model is not biased towards a normal test-train split. We used K value as 5 both the time.

IV. TOOLS

A. *Programming language:*

Python 3.7

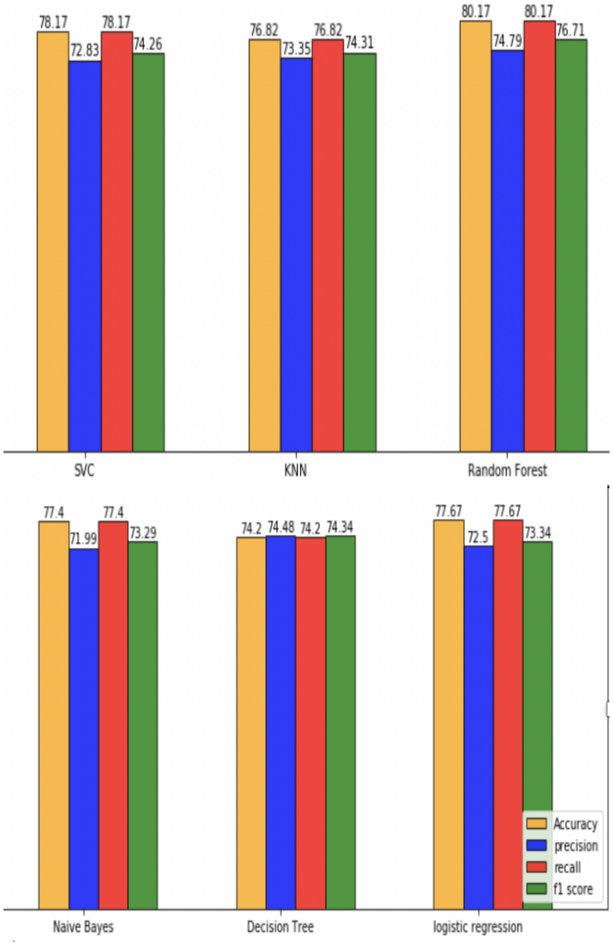
B. *libraries used:*

- NumPy
- Sklearn
- Math
- Pandas
- matplotlib
- seaborn
- mlxtend
- pprint
- yellowbrick

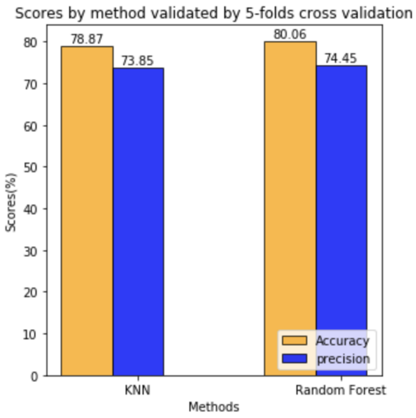
V. Results

A. Comparative Analysis

- Performance calculation with Train_Test Split validation



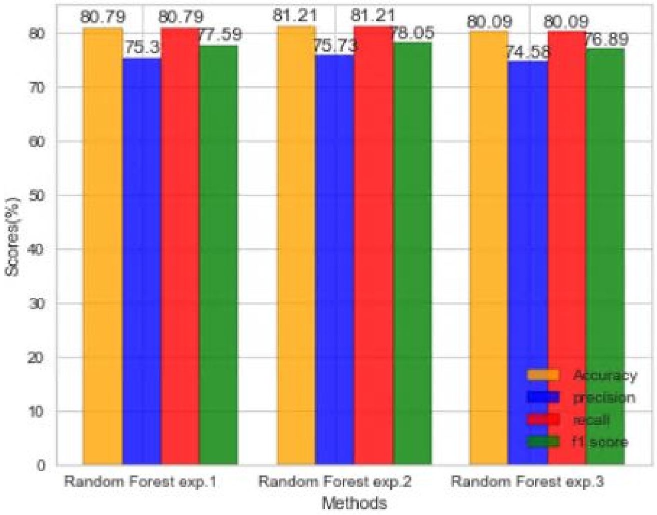
- Performance calculation with K-fold cross validation



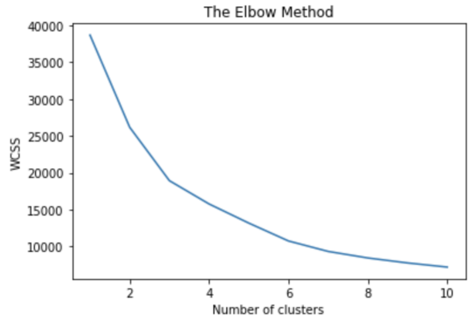
We have performed our experiments on the grouped restaurants dataset resulted from the data preprocessing which has 10386 instances Random Forest method gave the best performance in predicting the rating based on the selected features and it could achieve more than 80% accuracy. All other methods achieved similar results ranging from 74% to 78%. Same for other scores such as precision, recall and F1-score. Using some supporting methods could lead to achieving this good performance such as using hyperparameter tuning for selecting the efficient parameters besides feature scaling that gives weighted importance for all the selected features.

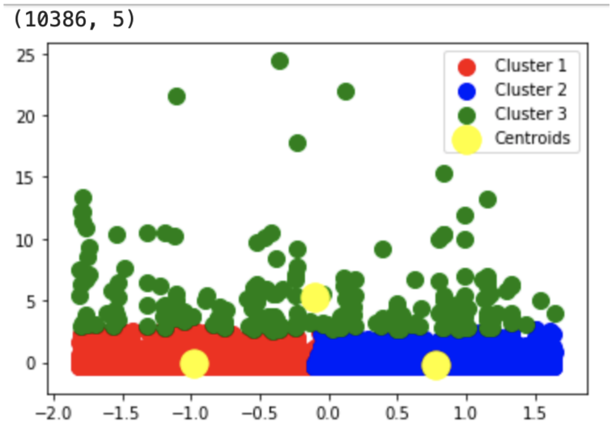
Method	Accuracy	Precision	Recall	F1-score
Random Forest	80.17	74.79	80.17	76.71
KNN	76.82	73.35	76.82	74.31
Logistic Regression	77.67	72.5	77.67	73.34
Naïve Bayes	77.4	71.99	77.4	73.29
Support Vector Machine	78.17	72.83	78.17	74.26
Decision Tree	74.2	74.48	74.2	74.34

Also, to confirm the consistent high performance of Random Forest, we performed a set of experiments using the same selected set hyperparameters and we achieved more than 80% accuracy in all them as shown in the figure below so we had 81.21 as a top-1 accuracy.

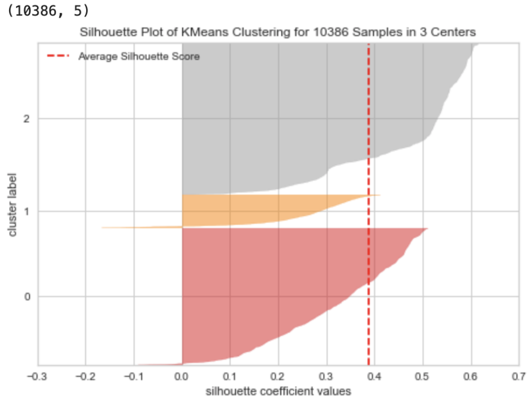


- K-means clustering
  We performed k-means clustering too to see how data is scattered across different clusters so first we used Elbow method to get the optimum number of k then we performed the clustering using k=3 and we got data scattered across clustered as shown in the second figure below.





```
(10386, 5)
```



```
(10386, 5)
```

We calculated the Silhouette width of this clustering solution and we achieved around 0.39.

VI. Discussion and Conclusion

A. Work Summary

We prepared our dataset and performed data pre-processing by handling missing values, formatting necessary data and correcting the wrong values. The best features for the Model are selected using the Sequential Floating Forward Selection (SFFS) method. We then trained our Machine learning model using a variety of classifiers like SVM, Naïve byes, Decision tree, Random forest & k-Nearest Neighbour(kNN). To select the best parameters for two of our classifiers - kNN and Random Forest, we performed hyperparameter tuning as well . Also, to learn from Unsupervised methods, we used the k-Means clustering algorithm.

B. Results summary

A comparative analysis of all the classification algorithms was performed. The best result was acquired from Random Forest, with accuracy of around 81%, precision of 75%.Also location, book_table,rest_type are found to be the three most important feature as per the model. In the case of k-means clustering, we got a silhouette width of 0.39. The optimal value of the Random Forest algorithm helped us use the model as the baseline for our prediction of ‘Ratings’.

C. Limitations

Our solution excludes ‘review list’, ‘menu items’ and a few other attributes which might contribute to more insights.

D. Future Work

In the future, we are planning to extend the project by adding NLP and deep learning methods to analyze features like ‘review list’, ‘menu items’

References

[1] Gupta, D. M. (2019). A Study on Impact of Online Food delivery app on Restaurant Business special reference to zomato and swiggy. IJRAR-International Journal of Research and Analytical Reviews, 5.

[2] Kumar, N. (2019, July 13). Exploratory Data Analysis Tutorial: Analyzing the Food Culture of Bangalore. Retrieved from Marktechpost: https://www.marktechpost.com/2019/07/13/exploratory-data-analysis-tutorial-analyzing-the-food-culture-of-bangalore/

[3] Saxena, A. (2019). An Analysis of Online Food Ordering Applications in India: Zomato and Swiggy. International Journal of Research in Engineering, IT and Social Sciences, ISSN 2250-0588, 21.

[4] URL-”https://towardsdatascience.com/logistic-regression-for-dummies-a-detailed-explanation-9597f76edf46”

Appendix

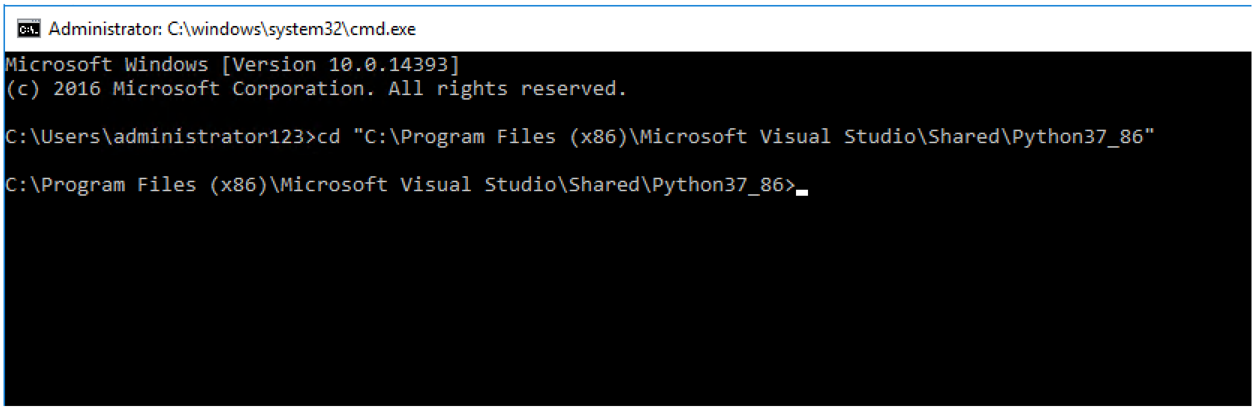
1- Create new folder for the project then copy the files from D2L to this new folder.

2- Install Python 3.7 if it is not already installed.

Python installers can be downloaded from this link: https://www.python.org/downloads/

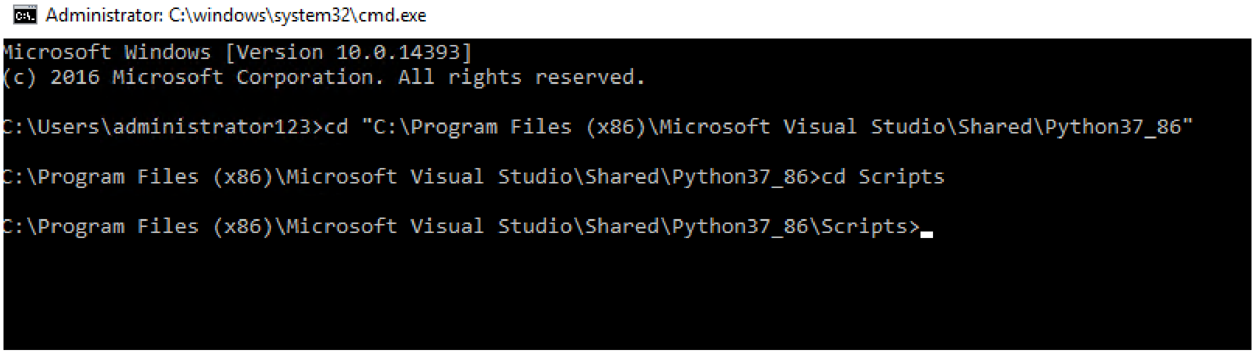
3- Open Windows command prompt.

4- Browse to the Python installation folder.



```
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\administrator123>cd "C:\Program Files (x86)\Microsoft Visual Studio\Shared\Python37_86"

C:\Program Files (x86)\Microsoft Visual Studio\Shared\Python37_86>_
```

5- From the installation folder, browse to the scripts folder.



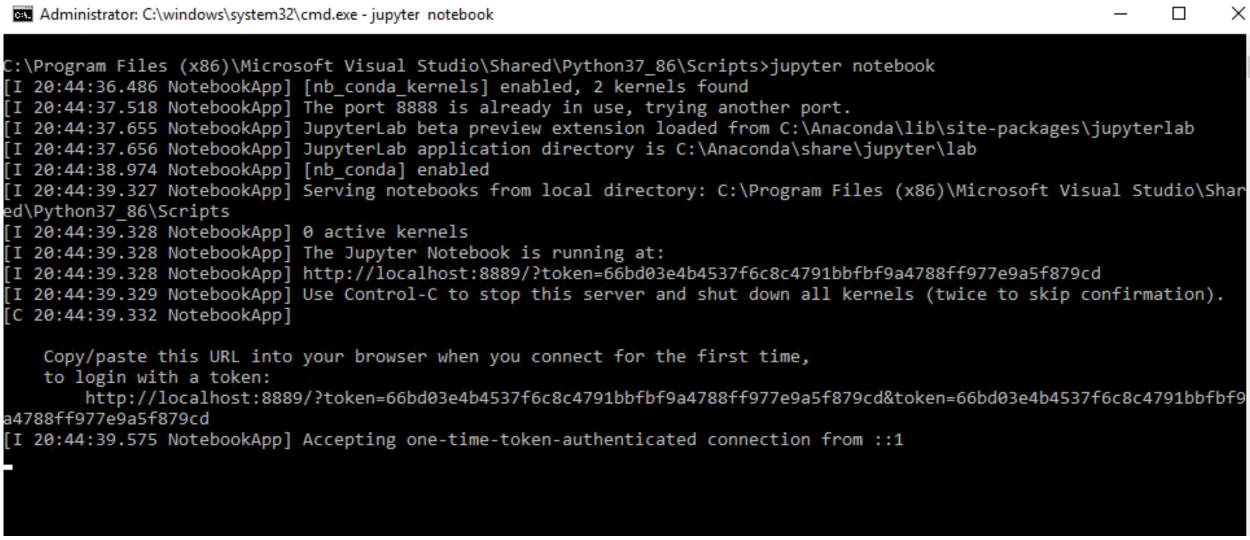
```
icrosoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

:\Users\administrator123>cd "C:\Program Files (x86)\Microsoft Visual Studio\Shared\Python37_86"

:\Program Files (x86)\Microsoft Visual Studio\Shared\Python37_86>cd Scripts

:\Program Files (x86)\Microsoft Visual Studio\Shared\Python37_86\Scripts>_
```

6- If jupyter is not installed, execute this command “pip install jupyterlab”, otherwise, skip this step.

7- Similarly, run the following commands (one by one) to install the required libraries:
Pip install numpy
Pip install pandas
Pip install matplotlib
Pip install seaborn
Pip install sklearn
Pip install mlxtend
Pip install pprint
Pip install yellowbrick

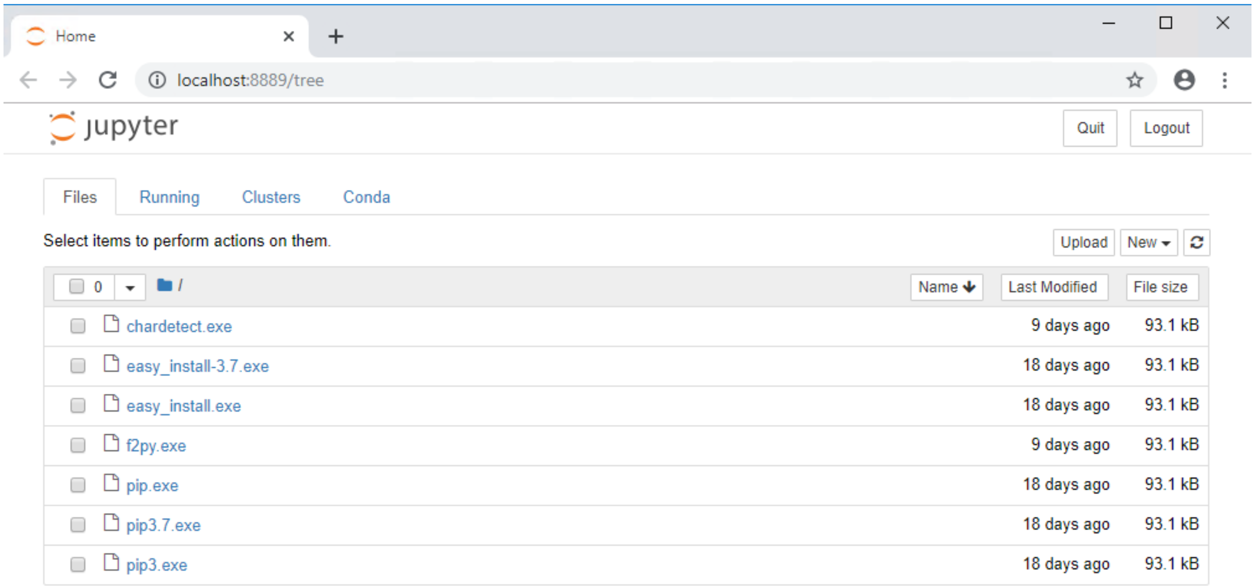
Note: Python might give some warnings if libraries are already installed.

8- After jupyter is installed, start jupyter with executing this command “jupyter notebook”


Administrator: C:\windows\system32\cmd.exe - jupyter notebook

```
C:\Program Files (x86)\Microsoft Visual Studio\Shared\Python37_86\Scripts>jupyter notebook
[I 20:44:36.486 NotebookApp] [nb_conda_kernels] enabled, 2 kernels found
[I 20:44:37.518 NotebookApp] The port 8888 is already in use, trying another port.
[I 20:44:37.655 NotebookApp] JupyterLab beta preview extension loaded from C:\Anaconda\lib\site-packages\jupyterlab
[I 20:44:37.656 NotebookApp] JupyterLab application directory is C:\Anaconda\share\jupyter\lab
[I 20:44:38.974 NotebookApp] [nb_conda] enabled
[I 20:44:39.327 NotebookApp] Serving notebooks from local directory: C:\Program Files (x86)\Microsoft Visual Studio\Shar
ed\Python37_86\Scripts
[I 20:44:39.328 NotebookApp] 0 active kernels
[I 20:44:39.328 NotebookApp] The Jupyter Notebook is running at:
[I 20:44:39.328 NotebookApp] http://localhost:8889/?token=66bd03e4b4537f6c8c4791bbfbf9a4788ff977e9a5f879cd
[I 20:44:39.329 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 20:44:39.332 NotebookApp]

    Copy/paste this URL into your browser when you connect for the first time,
    to login with a token:
        http://localhost:8889/?token=66bd03e4b4537f6c8c4791bbfbf9a4788ff977e9a5f879cd&token=66bd03e4b4537f6c8c4791bbfbf9
a4788ff977e9a5f879cd
[I 20:44:39.575 NotebookApp] Accepting one-time-token-authenticated connection from ::1
```

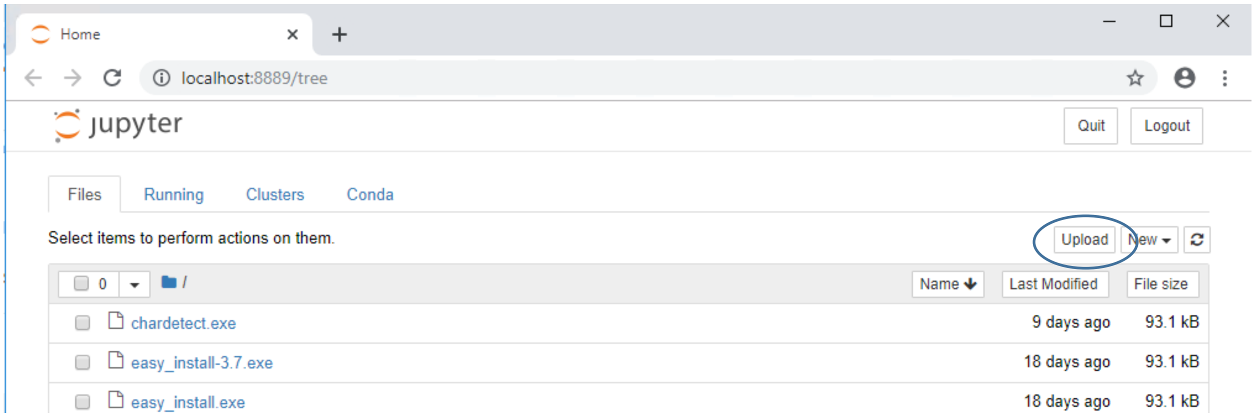
9- Make sure that jupyter starts without errors and finally, a browser window will open with the jupyter home page as shown below:



Select items to perform actions on them.

Name	Last Modified	File size
chardetect.exe	9 days ago	93.1 kB
easy_install-3.7.exe	18 days ago	93.1 kB
easy_install.exe	18 days ago	93.1 kB
f2py.exe	9 days ago	93.1 kB
pip.exe	18 days ago	93.1 kB
pip3.7.exe	18 days ago	93.1 kB
pip3.exe	18 days ago	93.1 kB

10- Click on “upload” button

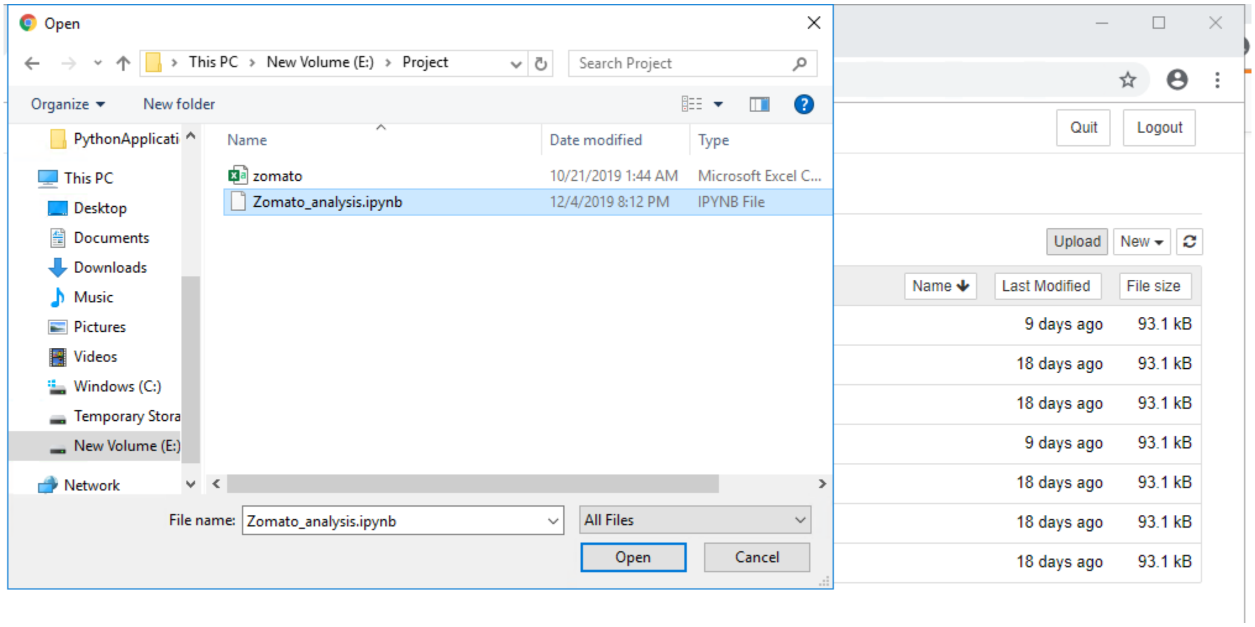


Select items to perform actions on them.

Name	Last Modified	File size
chardetect.exe	9 days ago	93.1 kB
easy_install-3.7.exe	18 days ago	93.1 kB
easy_install.exe	18 days ago	93.1 kB

11- File dialog will open.

12- Browse to the project folder created in step#1



Name	Date modified	Type
zomato	10/21/2019 1:44 AM	Microsoft Excel C...
Zomato_analysis.ipynb	12/4/2019 8:12 PM	IPYNB File

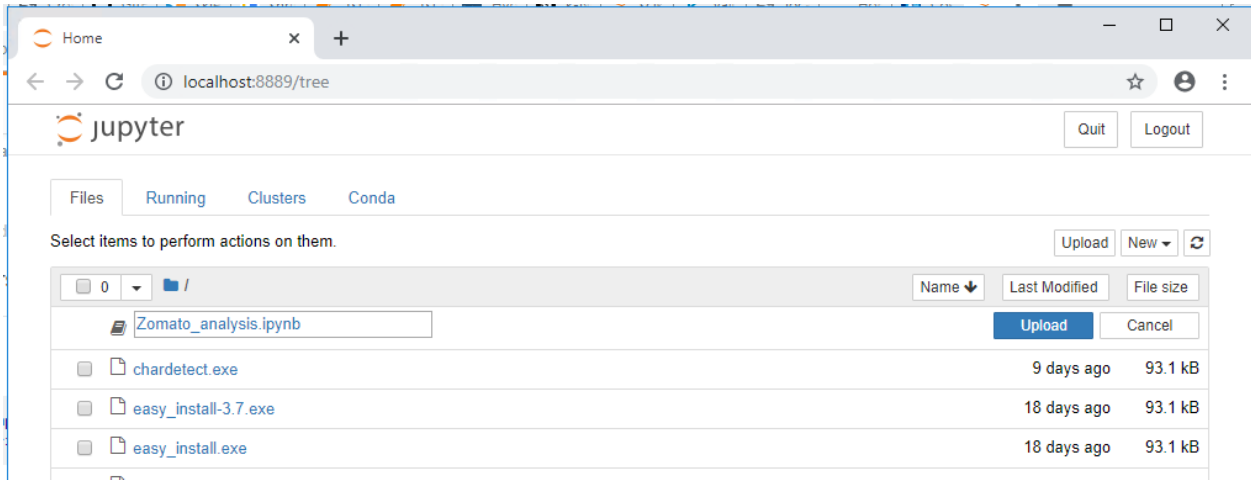
File name: Zomato_analysis.ipynb

All Files

Last Modified	File size
9 days ago	93.1 kB
18 days ago	93.1 kB
18 days ago	93.1 kB
9 days ago	93.1 kB
18 days ago	93.1 kB
18 days ago	93.1 kB
18 days ago	93.1 kB

13- Select Zomato_analysis.ipynb then click open.

14- Make sure file is added as shown below.

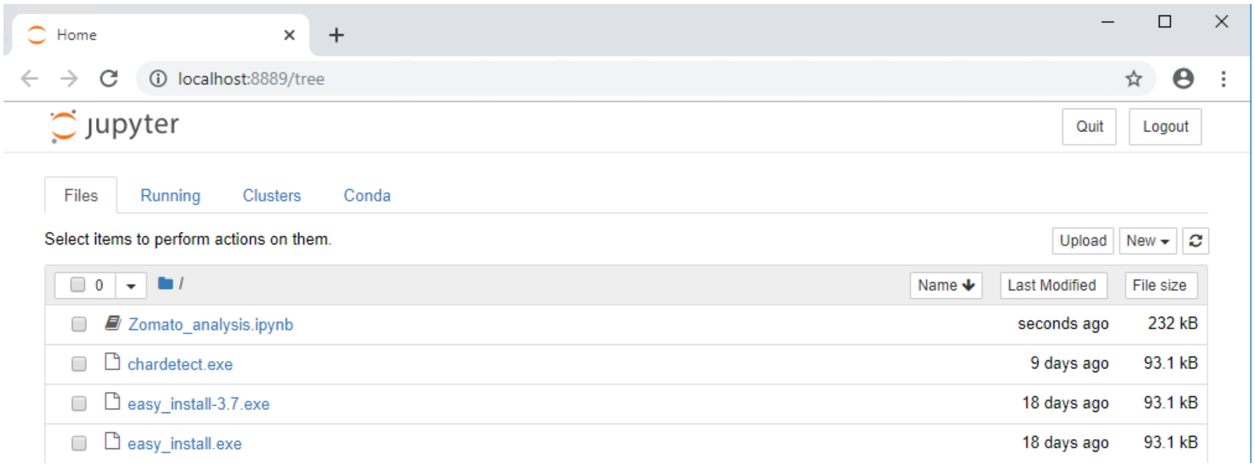


Select items to perform actions on them.

Zomato_analysis.ipynb		
chardetect.exe	9 days ago	93.1 kB
easy_install-3.7.exe	18 days ago	93.1 kB
easy_install.exe	18 days ago	93.1 kB

15- Click on the blue Upload button listed in the same row of the file.

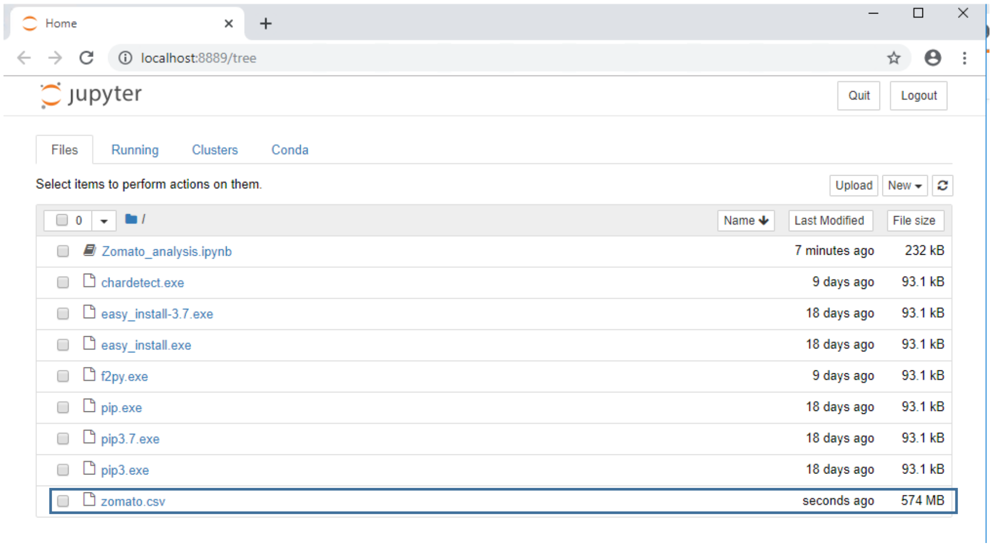
16- Make sure that file is uploaded successfully as shown below:



Select items to perform actions on them.

Zomato_analysis.ipynb	seconds ago	232 kB
chardetect.exe	9 days ago	93.1 kB
easy_install-3.7.exe	18 days ago	93.1 kB
easy_install.exe	18 days ago	93.1 kB

17- Repeat steps 10 to 16 to upload the dataset file zomato.csv then make sure that dataset file is uploaded successfully.

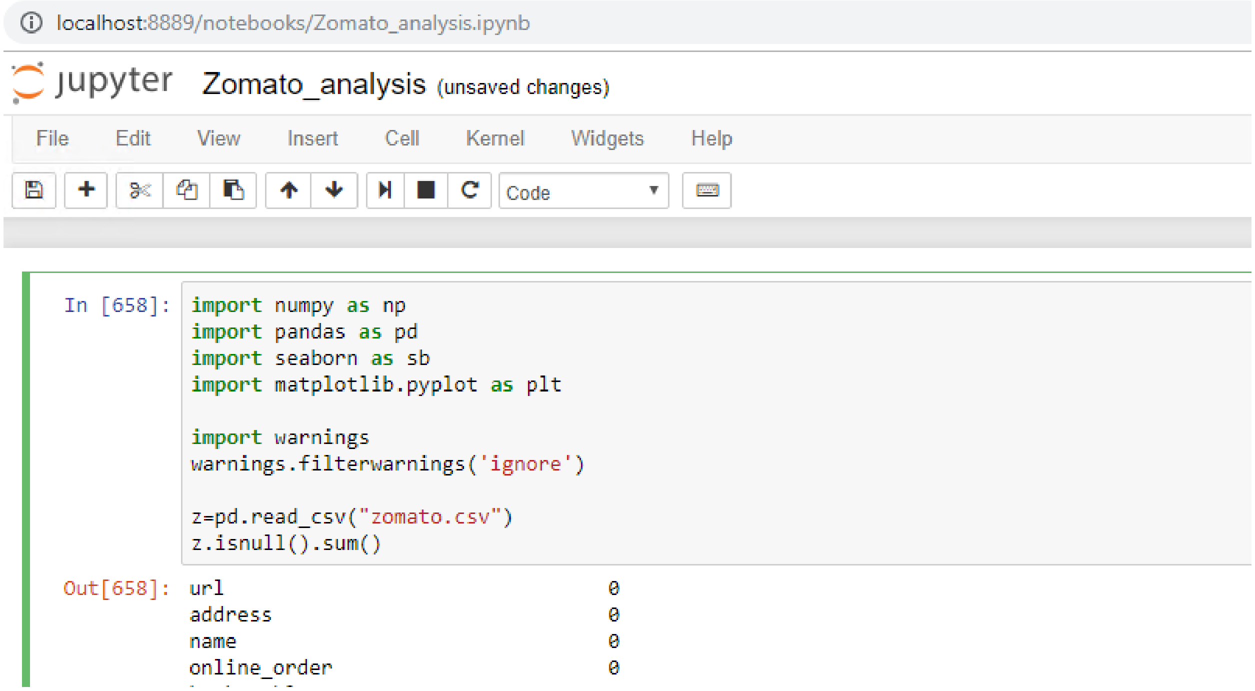


Select items to perform actions on them.

Zomato_analysis.ipynb	7 minutes ago	232 kB
chardetect.exe	9 days ago	93.1 kB
easy_install-3.7.exe	18 days ago	93.1 kB
easy_install.exe	18 days ago	93.1 kB
f2py.exe	9 days ago	93.1 kB
pip.exe	18 days ago	93.1 kB
pip3.7.exe	18 days ago	93.1 kB
pip3.exe	18 days ago	93.1 kB
zomato.csv	seconds ago	574 MB

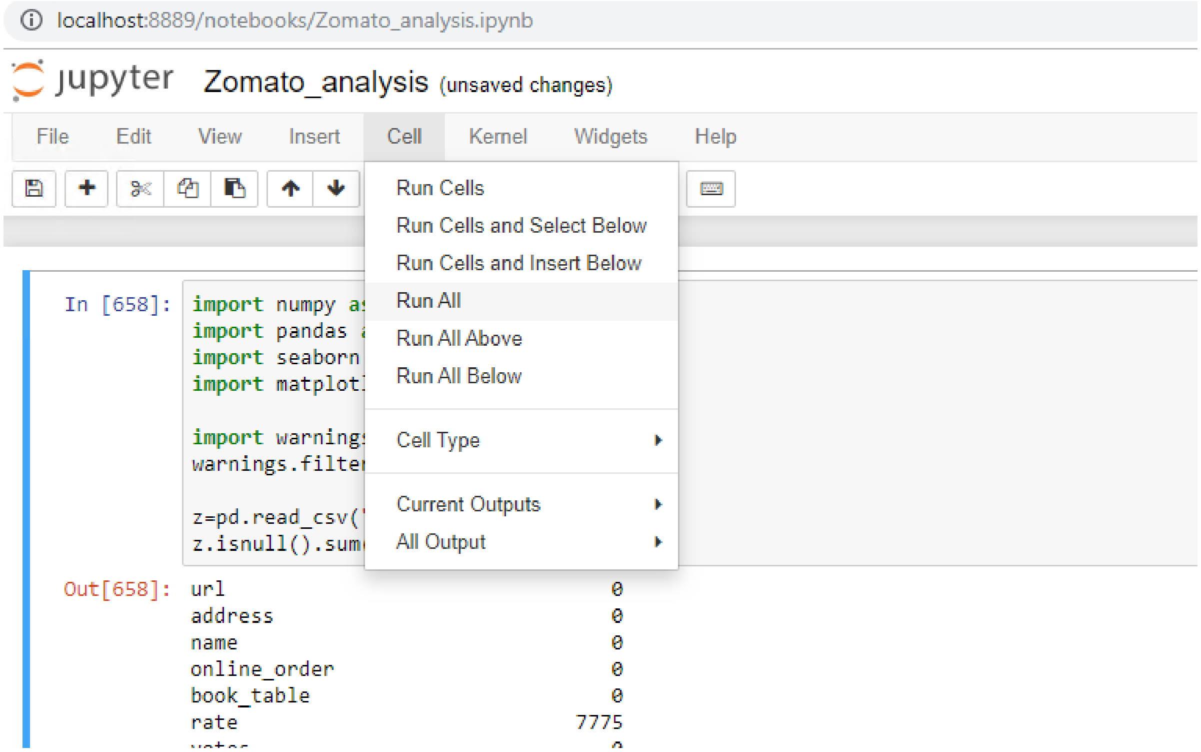
18- Click on Zomato_analysis.ipynb

19- The file will open as shown below:


localhost:8889/notebooks/Zomato_analysis.ipynb
jupyter
Zomato_analysis (unsaved changes)
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Code

```
In [658]: import numpy as np
          import pandas as pd
          import seaborn as sb
          import matplotlib.pyplot as plt

          import warnings
          warnings.filterwarnings('ignore')

          z=pd.read_csv("zomato.csv")
          z.isnull().sum()

Out[658]: url                                  0
          address                              0
          name                                 0
          online_order                         0
```

20- From the menu bar, select cell then “Run All”


localhost:8889/notebooks/Zomato_analysis.ipynb
jupyter
Zomato_analysis (unsaved changes)
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Run Cells
Run Cells and Select Below
Run Cells and Insert Below
Run All
Run All Above
Run All Below
Cell Type
Current Outputs
All Output

```
In [658]: import numpy as
          import pandas a
          import seaborn
          import matplotl

          import warning
          warnings.filte

          z=pd.read_csv(
          z.isnull().sum

Out[658]: url                                  0
          address                              0
          name                                 0
          online_order                         0
          book_table                           0
          rate                              7775
```

Jupyter will run all program steps and this process will take upto 5 minutes to complete.