DATA SCIENCE PROJECT REPORT:
Restaurant Recommendation System:

Mohit Zanwar – 111903145 Rohaan Advani 111903151 Shilpa Mohan – 111903155

PROJECT OBJECTIVES:

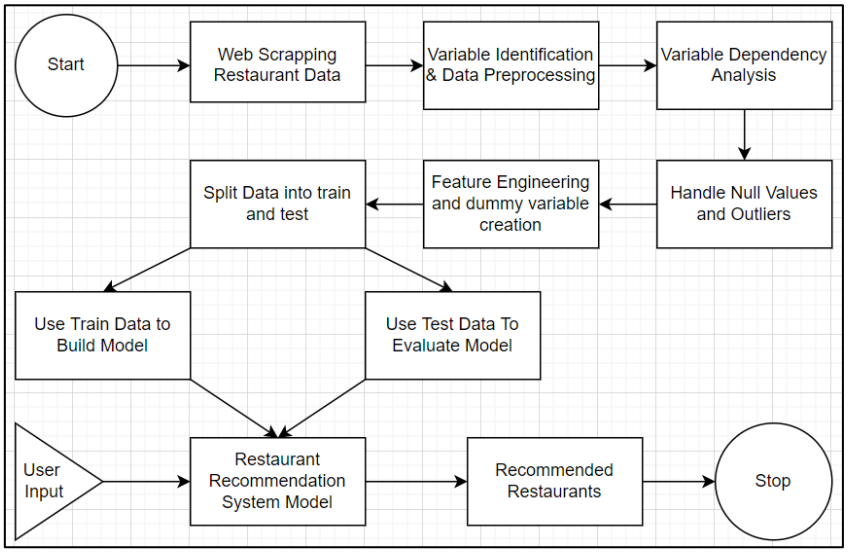
1. The system is intended to serve recommendation to a user by calculating his likeliness towards a particular restaurants and similarity between various other users by collaborative filtering which can further improve the user's experience by giving it more accurate results.

2. Perform a location-based analysis to check which restaurants are feasible for dining / delivery.

3. Perform an economic analysis of general city population to check which type of crowd can afford dining prices in restaurants.

4. Perform area-based analysis to analyse the highest and lowest rated restaurants in the locality.

DESIGN MODEL:



STEP 1 – WEB SCRAPPING RESTAURANT DATA:

Scrape Restaurant data from known sources such as Swiggy / Zomato / Yelp.
The restaurant data collected was as follows:

1. Restaurant Name – Name of restaurant
2. Cuisine Served – Food served
3. Stars / Rating – Reviews of customers
4. Average price for two – For budget and economic analysis
5. Location – For location-based feasibility
6. Fine Dine – For executive dining
7. Night Life – For young crowd / event areas
8. Food Type – For nutritional analysis
9. Meal Type – Snack/Lunch/Dinner
10. Student Area – Places with maximum student population.

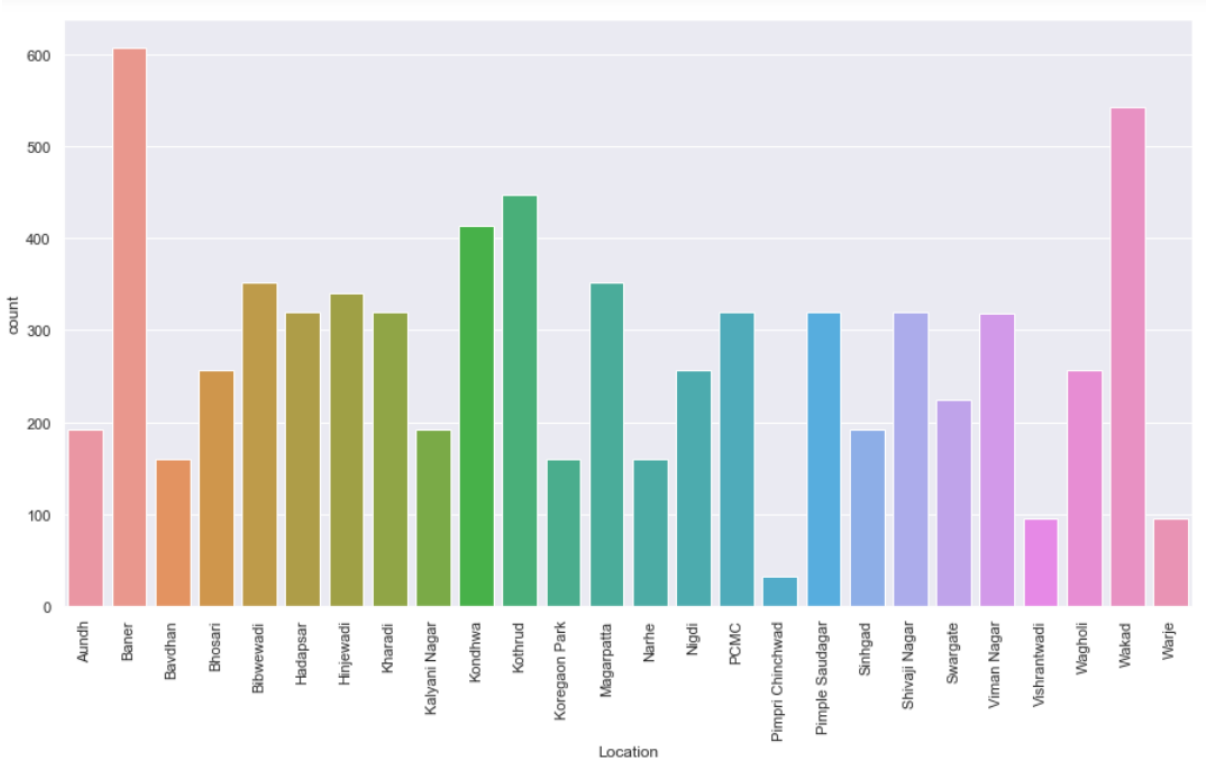
STEP 2 – VARIABLE IDENTIFICATION:

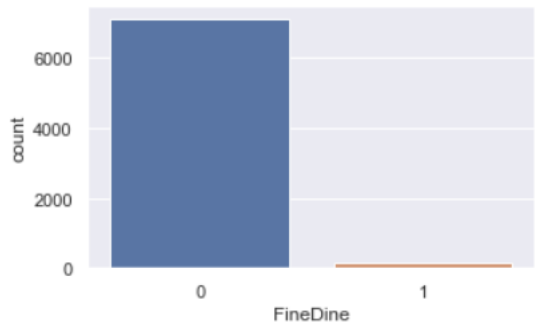
Categorical Variable:
1. Location
2. Fine Dine
3. Night Life
4. Food Type
5. Meal Type
6. Student area

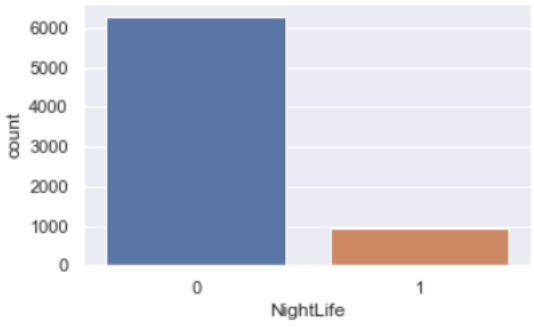
Continuous Variables:
1. Stars
2. Price for two

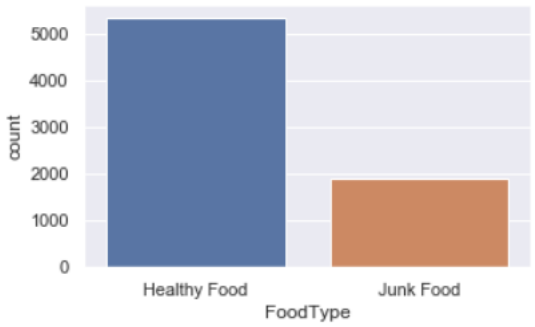
Descriptive Variables:
1. Restaurant Name
2. Cuisine Type

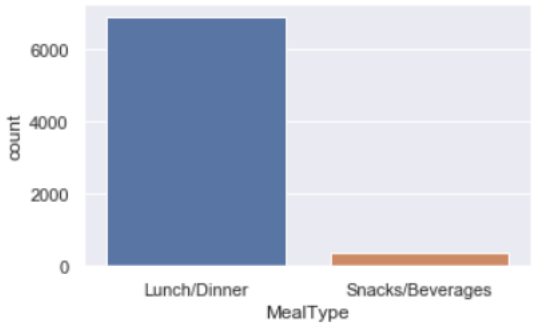
STEP 3 – VARIABLE ANALYSIS:

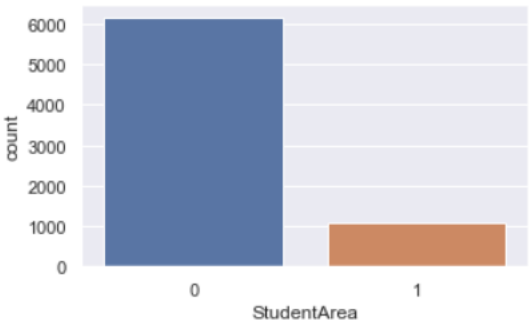


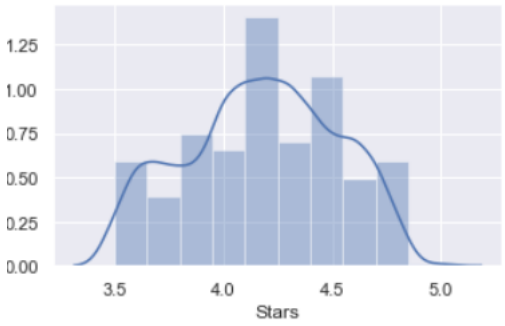


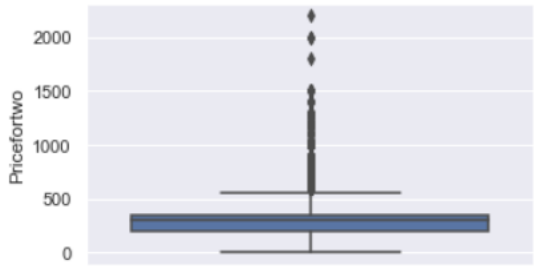












STEP 4 – ENCODING CATEGORICAL VARIABLES:

```
      Baner  Bavdhan  Bhosari  Bibwewadi  Hadapsar  Hinjewadi  Kalyani Nagar  \
0         0        0        0          0         0          0              0
1         0        0        0          0         0          0              0
2         0        0        0          0         0          0              0
3         0        0        0          0         0          0              0
4         0        0        0          0         0          0              0
...     ...      ...      ...        ...       ...        ...            ...
7240      0        0        0          0         0          0              0
7241      0        0        0          0         0          0              0
7242      0        0        0          0         0          0              0
7243      0        0        0          0         0          0              0
7244      0        0        0          0         0          0              0

      Kharadi  Kondhwa  Koregaon Park  ...  Shivaji Nagar  Sinhgad  Swargate  \
0           0        0              0  ...              0        0         0
1           0        0              0  ...              0        0         0
2           0        0              0  ...              0        0         0
3           0        0              0  ...              0        0         0
4           0        0              0  ...              0        0         0
...       ...      ...            ...  ...            ...      ...       ...
7240        0        0              0  ...              0        0         0
7241        0        0              0  ...              0        0         0
7242        0        0              0  ...              0        0         0
7243        0        0              0  ...              0        0         0
7244        0        0              0  ...              0        0         0

      Viman Nagar  Vishrantwadi  Wagholi  Wakad  Warje  Junk Food  \
0               0             0        0      0      0          0
1               0             0        0      0      0          1
2               0             0        0      0      0          0
3               0             0        0      0      0          0
4               0             0        0      0      0          0
...           ...           ...      ...    ...    ...        ...
7240            0             0        0      0      1          1
7241            0             0        0      0      1          0
7242            0             0        0      0      1          1
7243            0             0        0      0      1          1
7244            0             0        0      0      1          1

      Snacks/Beverages
0                    0
1                    0
2                    0
3                    0
4                    0
...                ...
7240                 0
7241                 0
7242                 0
7243                 0
7244                 0

[7245 rows x 27 columns]
```

STEP 5 – HANDLE NULL VALUES AND OUTLIERS:

There are no null values in the dataset thus null values need not be handled.

In order to handle outliers, we have to calculate:
1. IQR = Q3 – Q1.
2. Upper Limit = Q3 + 1.5 * IQR.
3. Lower Limit = Q1 – 1.5 * IQR.
4. All data points Above the Upper Limit are set to Upper Limit and all data points Below the Lower Limit are set to Lower Limit.

STEP 6 – PREPARATION OF LOCATION DISTANCE DATA:

The distance dataset has the following columns:
1. Location – Represents the area of residence
2. Latitude – Latitude coordinate of area of residence.
3. Longitude – Longitude coordinate of area of residence.
4. X – Longitude * 88Km (Average distances between longitudes is 88 Km @ 40N / 40S.
5. Y – Latitude * 110Km (Distance between each latitude is 110Km).

STEP 7 – PREPARATION OF JOB DATA:

The job dataset has the following columns:
1. Job – List of all occupations
2. Average Monthly Salary of occupation.

STEP 8 – DROP UNREQUIRED COLUMNS:

The location encoded and other categorical unencoded columns can be dropped. Since we are using the distance table for location-based preference calculation.

STEP 9 – TAKE USER INPUT:

Following input is taken:
1. Age - determines sa(student area) is age between 10 and 25.
2. Area of Residence - determines location and location-based restaurant preferences.
3. Occupation - determines job-based restaurant preferences.
4. Cuisine Preferences (Enter Multiple Cuisines separated by comma(,)).
5. User's interest in Fine Dining.
6. User's interest in Night Life.
7. User's interest in Food Type.
8. Take current time to determine whether it is time for lunch/dinner or snacks/beverages.

STEP 10 – CALCULATE PREFERENCE SCORE:

Following are the steps of calculating preference score:
1. Take a count of the number of common cuisines between the user’s cuisine preferences and the cuisines offered by each restaurant and set preference score as this count.
2. If restaurant is more than 4.5 Star / 5 increment preference score by 1.
3. If price for two is too expensive (>1000) or too cheap (<300) decrement preference score by 1.
4. If fine dine, night life, snacks / beverages, junk food, student area preferences of user as same as that of restaurant increment preference score by 1 for each of the variables matching.

STEP 11 – CALCULATE LOCATION PREFERENCE SCORE:

Following are the steps of calculating location preference score:
1. Import the distance database and calc Z score which is the root mean square distance between the user's area of residence's X and Y coordinates and the X and Y coordinates of each location.
2. Sort the location with respect to the Z scores this is the order of location preferences. Set this order as the location-based preference wherein the closer locations have a higher preference score.

STEP 12 – CALCULATE JOB PREFERENCE SCORE:

Following are the steps of calculating job preference score:
1. Import the job database.
2. Calculate the daily avg salary = month avg salary / 30 for all professions.
3. Calculate the estimated meal cost = daily avg salary / 2 for all professions.
4. Calculate difference between users' profession and all professions.
5. Sort the difference for all professions and set the job preferences score for all restaurants comparing the estimate meal cost with price for two of all restaurants.

STEP 13 – CALCULATE AVERAGE PREFERENCE SCORE:

Average preference score for all restaurants is the average of the preference score, location preference score and job preference score.

STEP 14 – CALCULATE USERSCORE:

User score = Average of all average preference scores set to all restaurants.

STEP 15 – SORT DATABASE:

Sort database by the 3 preference scores in descending order in order to ensure that the higher preferred restaurants are set to the beginning of the dataset.
The top 10 restaurants are taken from the sorted database and output file is created for the top 10 recommended restaurants.

STEP 16 – SET RESTAURANTS AS RECOMMENDED:

1. Top 10 restaurants are set as recommended by default.
2. Set restaurant as recommended if preference score is greater than 4.
3. Set restaurant as recommended if location preference score is greater than 23.
4. Set restaurant as recommended if job preference score is greater than 15.
5. Save top ten restaurants of user in a list to put in user table.
6. Import user table and compare user score of all users to current user's user score and sort the differences in order to choose users whose preferences are closest to current users' preferences.
7. Take top 5 users top ten restaurant list saved and recommend it to the current user.

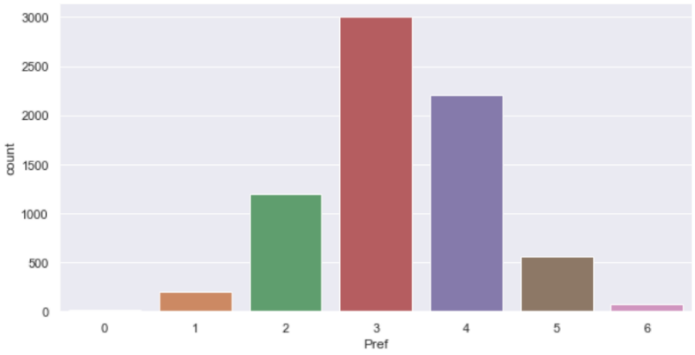
STEP 17 – ADD DATA TO USER TABLE:

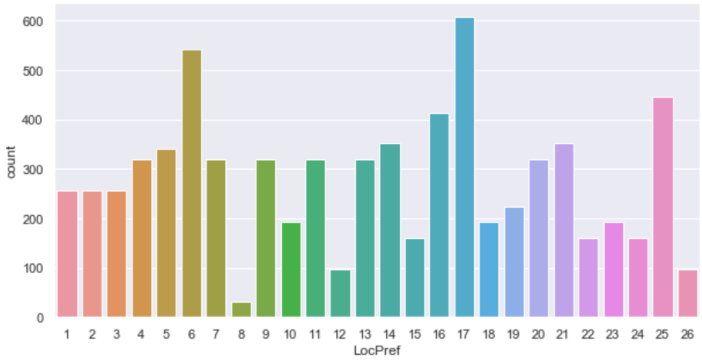
Add current user's user score and top ten list to user table for subsequent users.

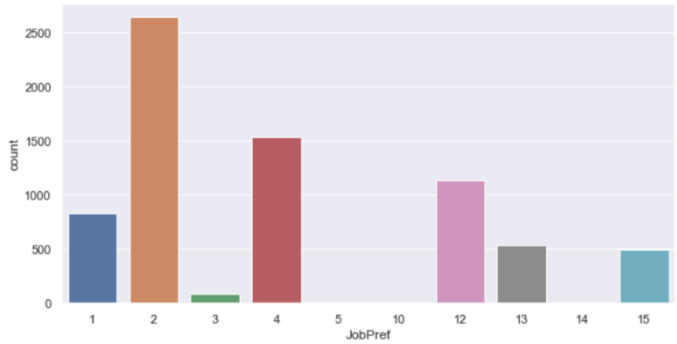
STEP 18 – SAVE ALL RECOMMENDED RESTAURANT DATA:

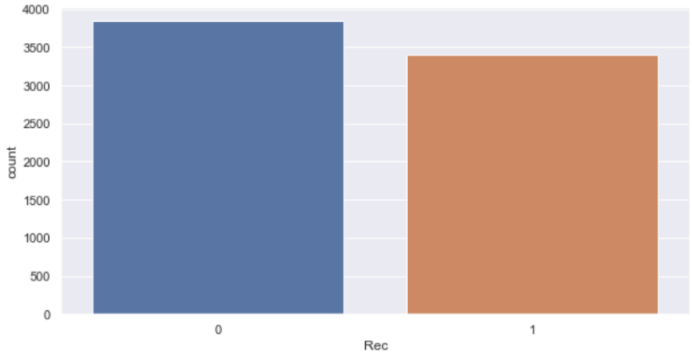
The restaurant names of the restaurants recommended after step 16 are added to a csv file as all restaurants recommended output.

STEP 19 – ANALYSE PREFERENCE SCORES AND RECOMMENDED:









STEP 20 – USERS' PREFERENCE SCORES AND RECOMMENDED VARIABLE NOW USED FOR MODELS:

1. Treat Outliers of the preference score variables.
2. Scale the preference score variables using the standard scaler.

STEP 21 – X/Y AND TRAIN/TEST SPLIT:

1. The Preference score are taken as X and the recommended variable is taken as Y.
2. Perform train test split set test size at 25%.

STEP 22 – BUILD DIFFERENT CLASSIFICATION MODELS:

1. KNN: K Nearest Neighbor algorithm falls under the Supervised Learning category and is used for classification (most commonly) and regression. It is a versatile algorithm also used for imputing missing values and resampling datasets. As the name (K Nearest Neighbor) suggests it considers K Nearest Neighbors (Data points) to predict the class or continuous value for the new Datapoint.

2. Decision Tree Classifier: It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome. In a Decision tree, there are two nodes, which are the Decision Node and Leaf Node.

3. Logistic Regression: Logistic regression is a supervised learning algorithm used to predict a dependent categorical target variable. In essence, if you have a large set of data that you want to categorize, logistic regression may be able to help. A logistic regression model predicts a dependent data variable by analyzing the relationship between one or more existing independent variables.

4. Random Forest Classifier: Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset. Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

STEP 23 – MODEL EVALUATION:

Evaluation of the model:

1. Accuracy: This is the rate of the classifier being correct, so basically take a sum of True Positive and True Negative values and then divide by total.

2. Precision: Precision evaluates how precise your model was at making predictions. This is a good metric to pay attention to if you want your model to be conservative in its flagging of data.

3. Recall: Recall evaluates the sensitivity of your model. Basically it checks how successful your model was at flagging the relevant samples.

4. F1 Score: The F-Score, or also referred to as the F-Measure, is the harmonic mean of precision and recall.

5. Confusion Matrix: A confusion matrix is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known.

KNN:

```
Accuracy: 0.9950331125827815
Precision: 0.9975669099756691
Recall: 0.9915356711003628
F1 Score: 0.9945421467556095
Confusion Matrix:
 [[983   2]
 [  7 820]]
```

Decision Tree:

```
Accuracy: 0.9983443708609272
Precision: 1.0
Recall: 0.9963724304715841
F1 Score: 0.9981829194427619
Confusion Matrix:
 [[985   0]
 [  3 824]]
```

Logistic Regression:

```
Accuracy: 0.9337748344370861
Precision: 0.9491740787801779
Recall: 0.9032648125755743
F1 Score: 0.9256505576208178
Confusion Matrix:
 [[945  40]
 [ 80 747]]
```

Random Forest:

```
Accuracy: 0.9983443708609272
Precision: 1.0
Recall: 0.9963724304715841
F1 Score: 0.9981829194427619
Confusion Matrix:
 [[985   0]
 [  3 824]]
```