horizontal line

****

***ZOMATO BANGALORE RESTAURANTS INSIGHTS AND PREDICTIVE MODELING FOR RATING***

***25th July 2020***

# *OVERVIEW:*

We live in an amazing age where we have numerous delivery outlets all around us to choose from a wide range of lip smacking cuisines from all around the world. If you are in the mood to gobble famous Mexican Nachos, just place an order from an app and in a regular time span, your food is rolled-up at your doorstep. Zomato is one such food delivery startup which helps us do this. Zomato has a tie up with most of the restaurants around the world and has rich data of these restaurants which could give us great insights.We will keep our data restricted to Bangalore restaurants.

***BUSINESS PROBLEM TO SOLVE/GOALS:***

* Ratings and reviews play a very important role in attracting new and retaining customers.
* Our target would be improving ratings based on the insights and based on these factors predict ratings using predictive modelling for a prospect restaurant.
* Understand what people like the most in a highly rated restaurant, in a particular locality, which are related to ratings for a prospect restaurant.
* Have an insight of approx\_cost (cost for two), which is based on many factors like neighborhood, restaurant type that can be related to ratings.
* Given a locality, a prospect restaurant can have an insight of the factors to get the best rating.
* Marketing strategies like personalized notifications, discounts etc. can be set up to attract an audience.

***WHO CARES ABOUT THE PROBLEM WE TRYING TO FIX ?***

* Potential clients would be existing restaurant owners and prospects restaurants.
* Having insights on such factors could help the decision makers take actions which would eventually increase the ratings and audience.

***DATA COLLECTION AND WRANGLING:***

The data is scraped from Zomato(<https://www.zomato.com/bangalore>) using the Python package ‘Beautiful soup’ as of Jan 2020.

Ref script: [Web scraping script (Zomato)](https://github.com/Anandpatil412/DSC/tree/master/CapstoneProject1/DataExtraction(WebZomato))

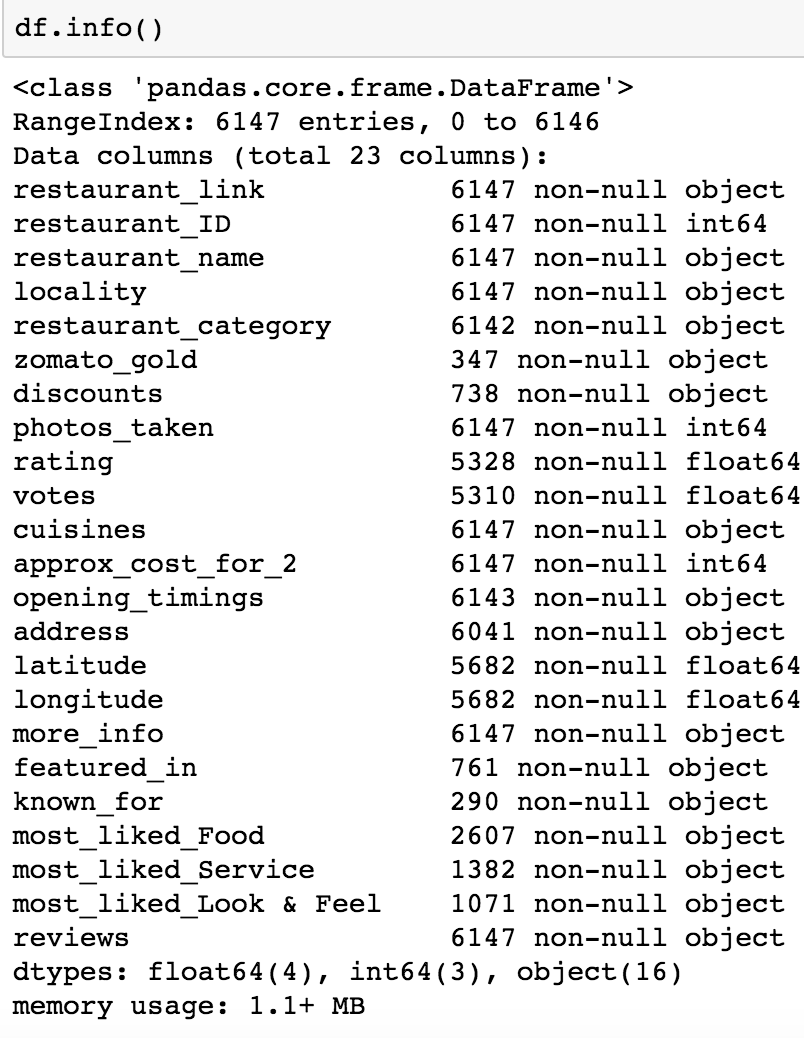
* Most of the data is cleaned/formatted while scraping.
* Some columns are manipulated to tuples.
* Opening and closing timings are transformed to datetime formats.
* Missing values are transformed to np.NAN
* Duplicates rows, if any, are removed based on the restaurant\_id.
* No outliers.

Data csv: [data csv](https://github.com/Anandpatil412/DSC/tree/master/CapstoneProject1/DataFiles/zomato_extracted_data_csv)

***EXPLORATORY DATA ANALYSIS SUMMARY:***

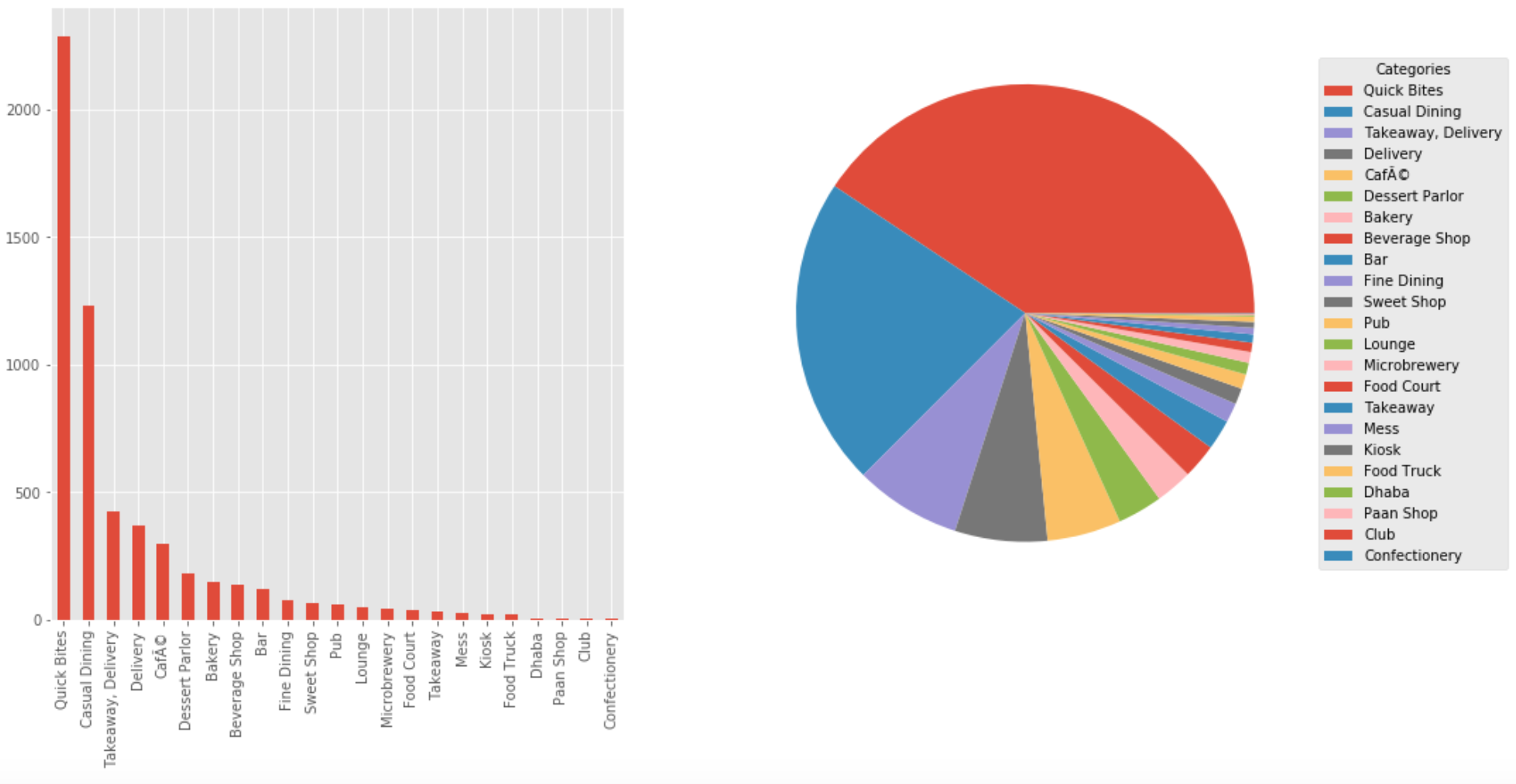
Ref script :[Data cleaning and wrangling script](https://github.com/Anandpatil412/DSC/blob/master/CapstoneProject1/DataWrangling/zomatoDataCleaning.ipynb)

* This dataset is relatively small, having information of around 6k restaurants.
* Let's have a quick glance of the data first and start exploring,

****

****

* ***Different Restaurant categories***

****

Quick bites and Casual dining are the most common of all restaurant categories.

* ***Cuisines***

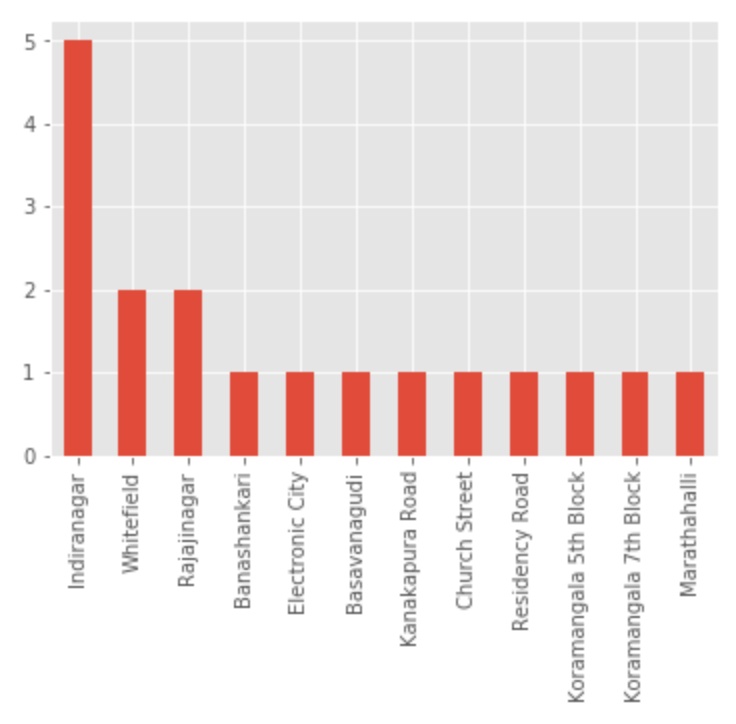
North Indian and Chinese are the most popular whereas Belgium, Portuguese are some of the rare cuisines.

* ***Average Restaurant rating in a locality:***

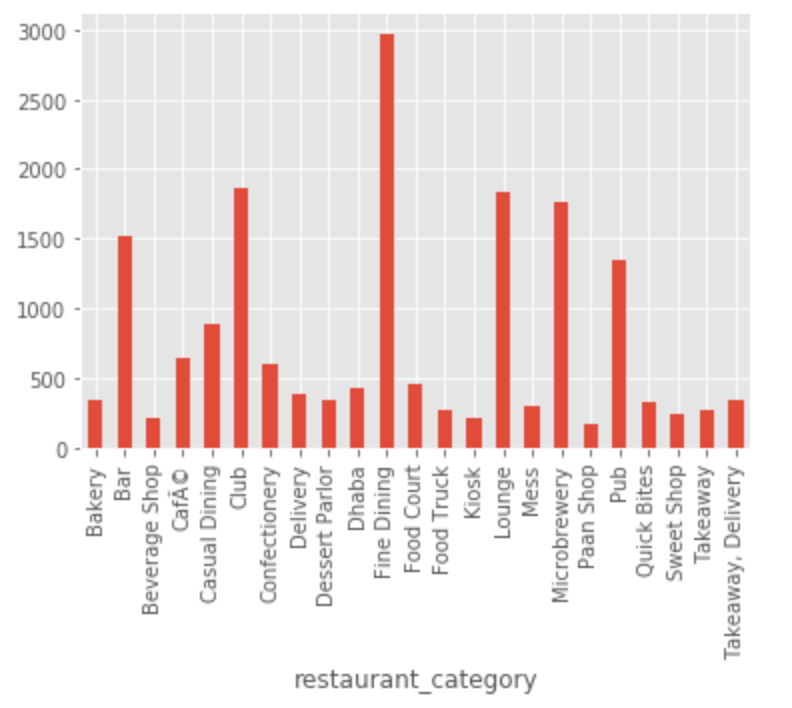
Sankey road, Lavelle road and Church street have highest average ratings.There is definitely locality playing a part in Restaurant rating.

* ***Highest number of newly opened restaurants in a locality***

Indiranagar has the highest number of ‘Newly opened” restaurants.



* ***Restaurant category having highest ‘Cost for 2’***



Fine Dining being the highest, followed by Club and Lounge have the highest ‘Cost for 2’.

* ***Highest and lowest rated restaurant categories :***



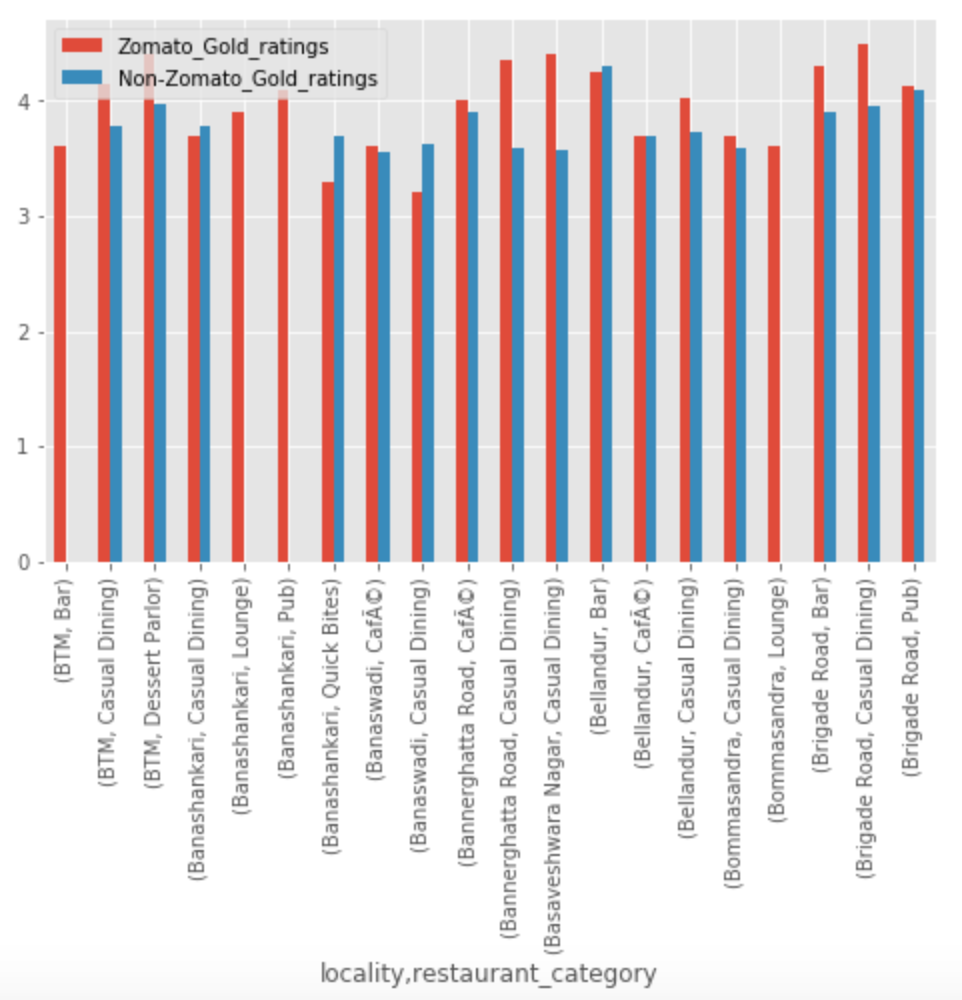
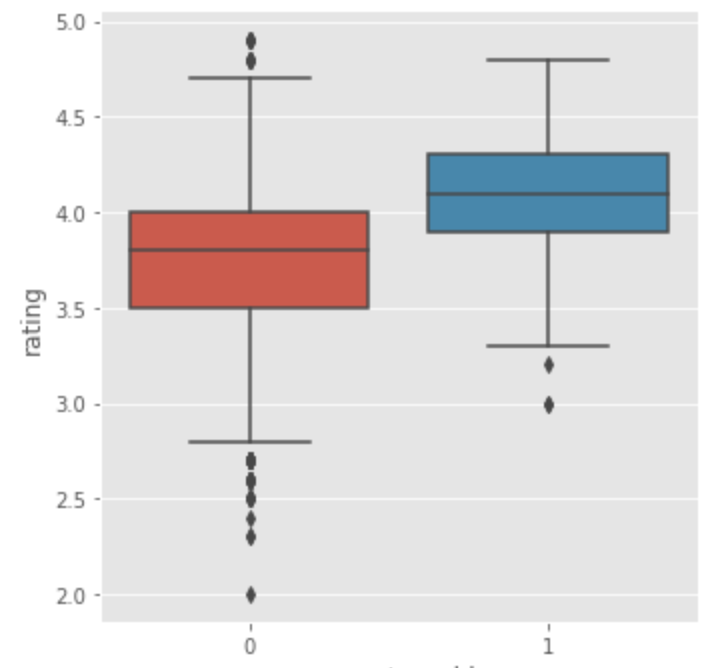
Microbrewery and Club are highest average ratings whereas Confectionery and Dhaba have the lowest average rating.

* ***The following figure summarises the Most liked features in the top rated restaurant categories(microbrewery and club respectively) :***



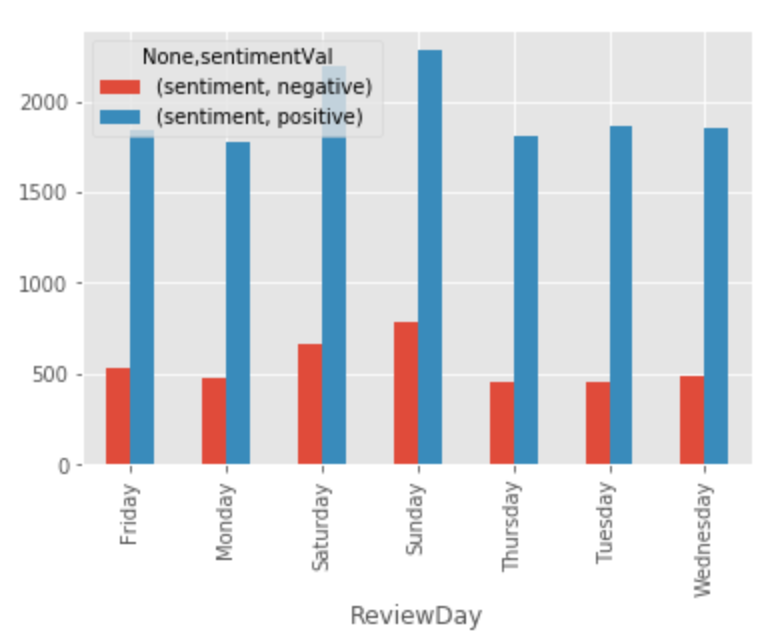
Staff, Chicken, Friendly, Courteous, Beer are some important features for these categories.

* ***Average rating of a Zomato Gold and a Non Zomato Gold restaurant in a locality:***



Majority of the times Zomato gold restaurants have higher ratings compared to a Non Zomato restaurant.

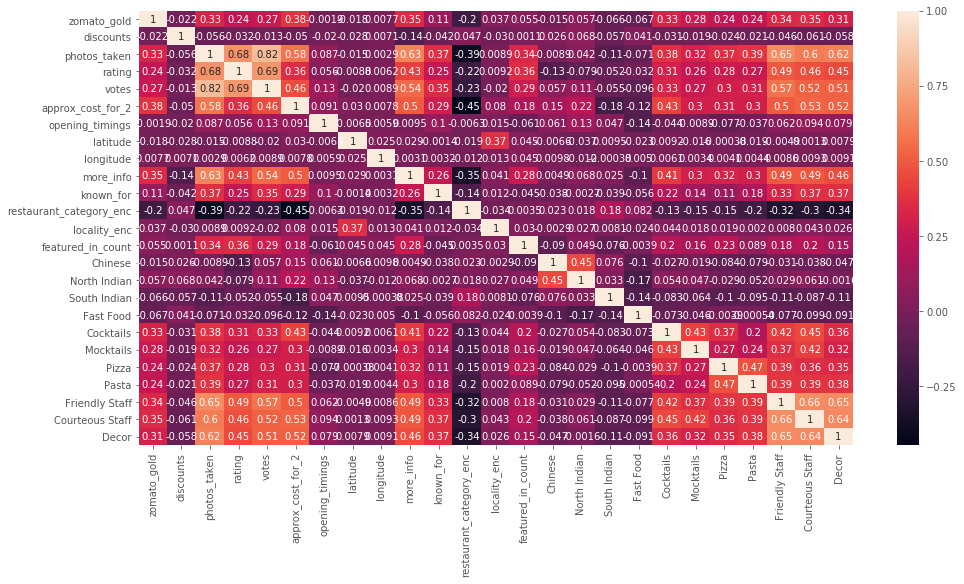
* ***Sentiments over a week:***



Looking at sentiments from latest user reviews, we see relatively high positive sentiments, also on weekends have high sentiment responses.

***EXPLORING CORRELATIONS AND TRENDS:***

To explore correlation and trends between variables we use the pandas correlation and seaborn pairplot visual representations.

******

* *Photos\_taken* are positively correlated with *ratings* and *votes*, and negatively correlated with *approx\_cost\_for\_2*.
* Features like *photos\_taken*, *votes*, *approx\_cost\_for\_2*, *featured\_in\_count*, *zomato\_gold* tend to have positive correlation with restaurant *rating*.
* *Discounts* have a negative correlation with *ratings*.
* Very few localities have high *approx\_cost\_for\_2*.
* From the above restaurant\_category distribution,we see that a very few categories are very popular.
* Only a few certain restaurant categories have high approx\_cost\_for\_2.
* From the distribution of *zomato\_gold*, we see that relatively few restaurants have zomato\_gold.
* From the distribution of discounts, we see that relatively few restaurants have discounts.

******

* Few localities have relatively high *approx\_cost\_for\_2*.
* From the above *restaurant\_category* distribution,we see that a very few categories are very popular.
* Few certain restaurant categories have high *approx\_cost\_for\_2*.
* From the distribution of *zomato\_gold*, we see that relatively few restaurants have zomato\_gold.
* From the distribution of *discounts*, we see that relatively few restaurants have discounts.
* *Photos\_taken* are positively correlated with ratings and votes,and negatively correlated with *approx\_cost\_for\_2*.
* Localities seem not correlated with *rating*.
* *Featured\_in\_count* looks positively correlated with *rating*.

***FEATURE ENGINEERING:***

Keeping in mind getting the best from model learnings, some feature engineering is required on the data.These are some of the techniques performed.

***Imputation:***

* Missing values replaced with NaN.
* *Restaurant\_id* considered as a unique identifier, duplicates removed.
* In some cases, Numeric imputation i.e replacing NaN with a number say ‘0’ made more sense like in case Zomato\_gold, etc.
* Some *Restaurant\_categories* were very low in number, Categorical imputation i.e replacing such categories with ‘Others’ made more sense.

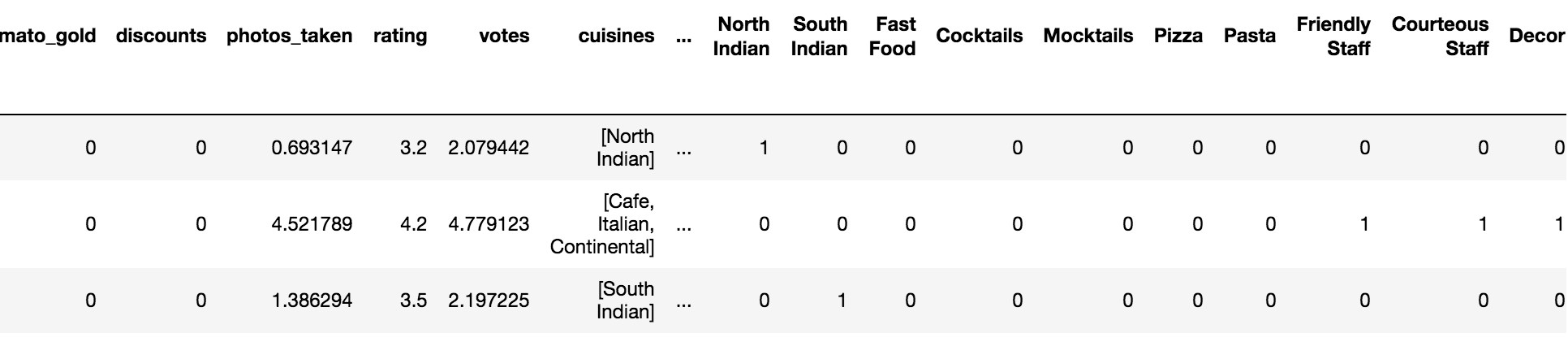
***Log Transform:***

* Log transformation helps to handle highly skewed data.On transforming the distribution is approximated to a normal.
* Log transformation helps to decrease the effect of outliers.
* Referring to the pairplots, we can see the *photos\_taken*, *votes*, *approx\_cost\_for\_2* are highly right skewed which should be log transformed.

***Encoding:***

Encoding converts non-numeric categorical data to numeric data, so as to convert it into a machine-readable form .

* *Label Encoding:* Encodes categories to uniques numbers.Locality and Restaurant\_categories are Label Encoded.
* *MultilabelBinarizer:* Encodes in a binary matrix indicating presence of class.We used this to encode *cuisines,* thenexpanded this to new features having only most prominent cuisines, most liked features.

**

Here, we have expanded to new features like ‘North indian’, ‘Cocktails’ ,etc which are most prominent.

* *Target Encoding:* In this encoding technique, we map or encode the variable, to mean of the target variable. Here, we generally split the data first, encode the variable in the train data then map respectively on the test data.Any unseen category in the test data can be filled with mean target variable. In this dataset, we have encoded *Restaurant\_category* using target encoding.

***Statistical inferences:***

Ref script: [Statistical inference script](https://github.com/Anandpatil412/DSC/blob/master/CapstoneProject1/Statistical_Inference/Statistical_Inference.ipynb)

We have performed some statistical tests using Bootstrap techniques to verify or have confidence of certain observations seen in Exploratory data analysis.

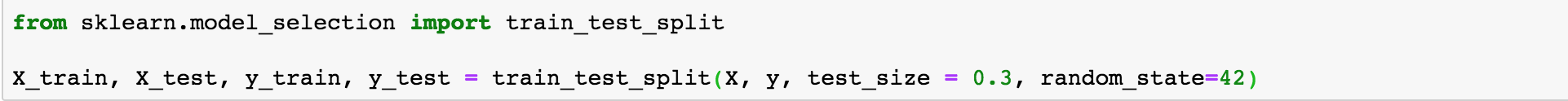
1. Null hypothesis test resulted in positive correlation between approx\_cost\_for\_2 and rating.
2. Null hypothesis test resulted in Zomato Gold have high ratings compared to Non Zomato Gold restaurants.

Now that we have a cleaner data set we can move to predictive modelling of the rating ,

***Predictive Modelling for Restaurant rating:***

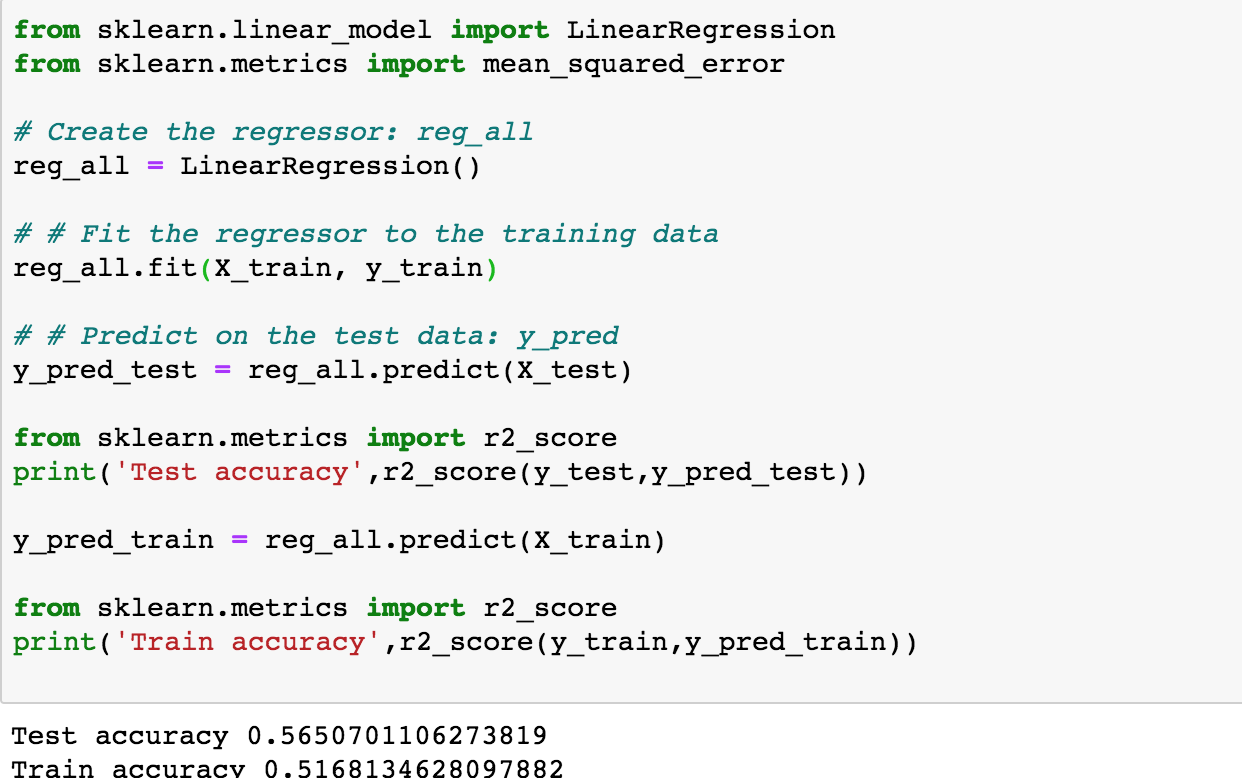
Ref script: [Predictive modeling script](https://github.com/Anandpatil412/DSC/blob/master/CapstoneProject1/IndepthAnalysis/In_depth_analysis.ipynb)

Referring to the pair plots and correlation matrix, we have selected features which show certain relation with the target variable and we split the data into train and test.



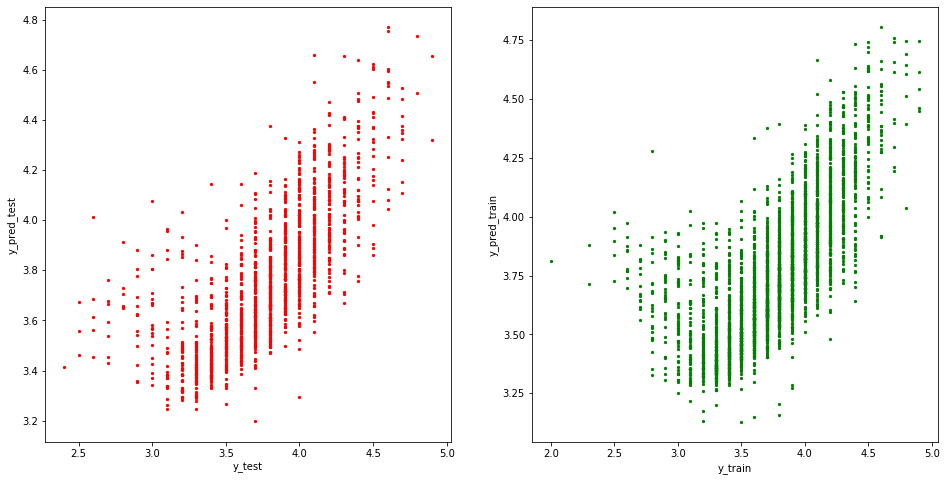


***Linear Regression:***

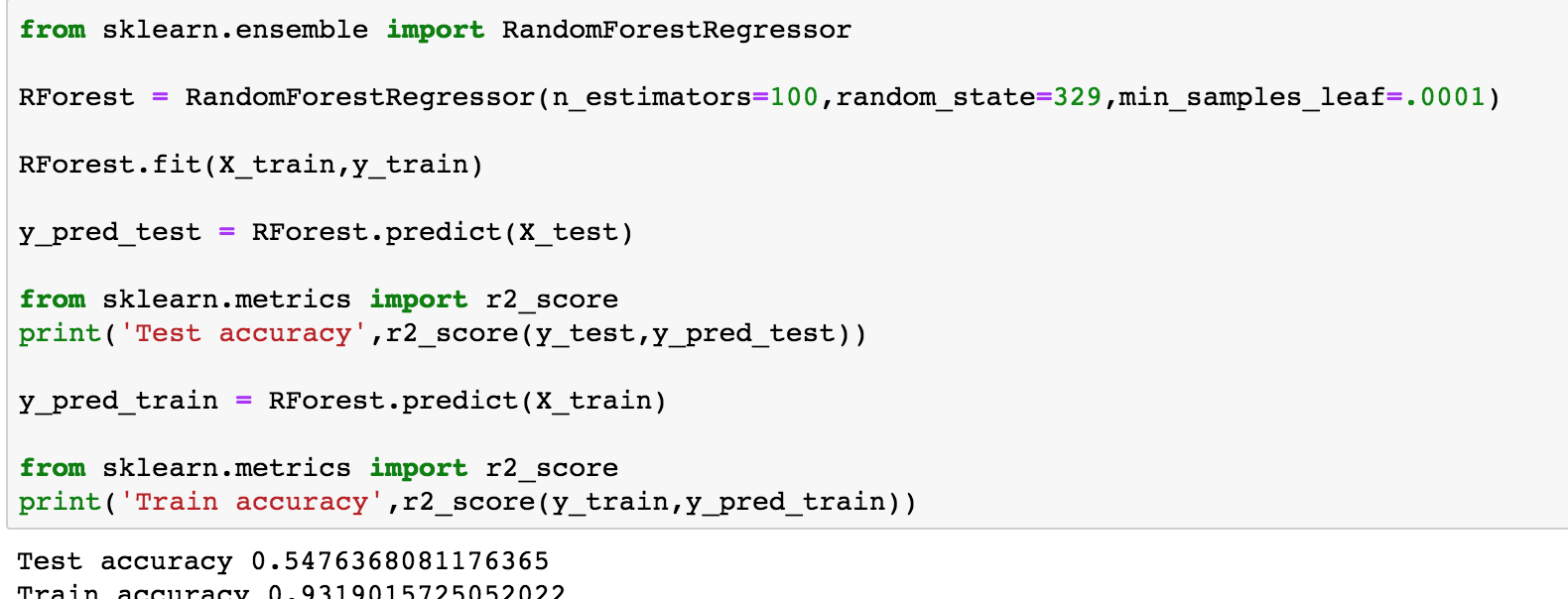
******

Linear regression a test accuracy of **0.565** and train accuracy of **0.516**

Further plotting prediction values against actual values,

******

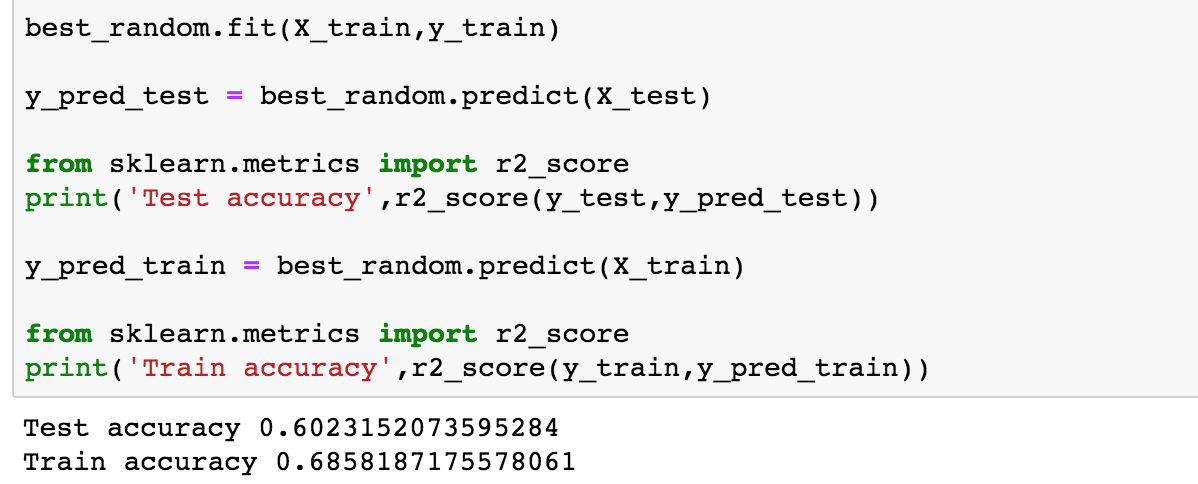
***Random Forest Regressor:***

******

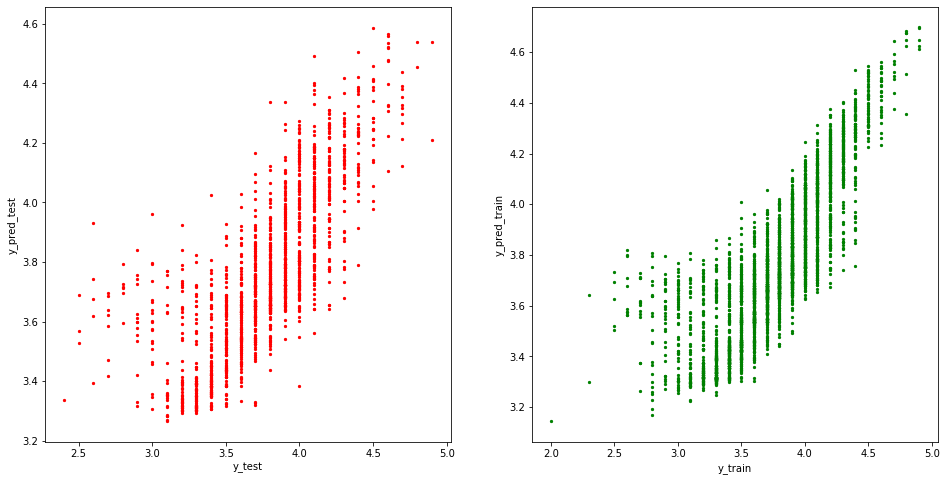
Random forest gives a test accuracy of **0.547** and train accuracy of **0.931** which indicates an overfitting.

Further we regularise and choose the best hyperparameters using *RandomizedSearchCV.*

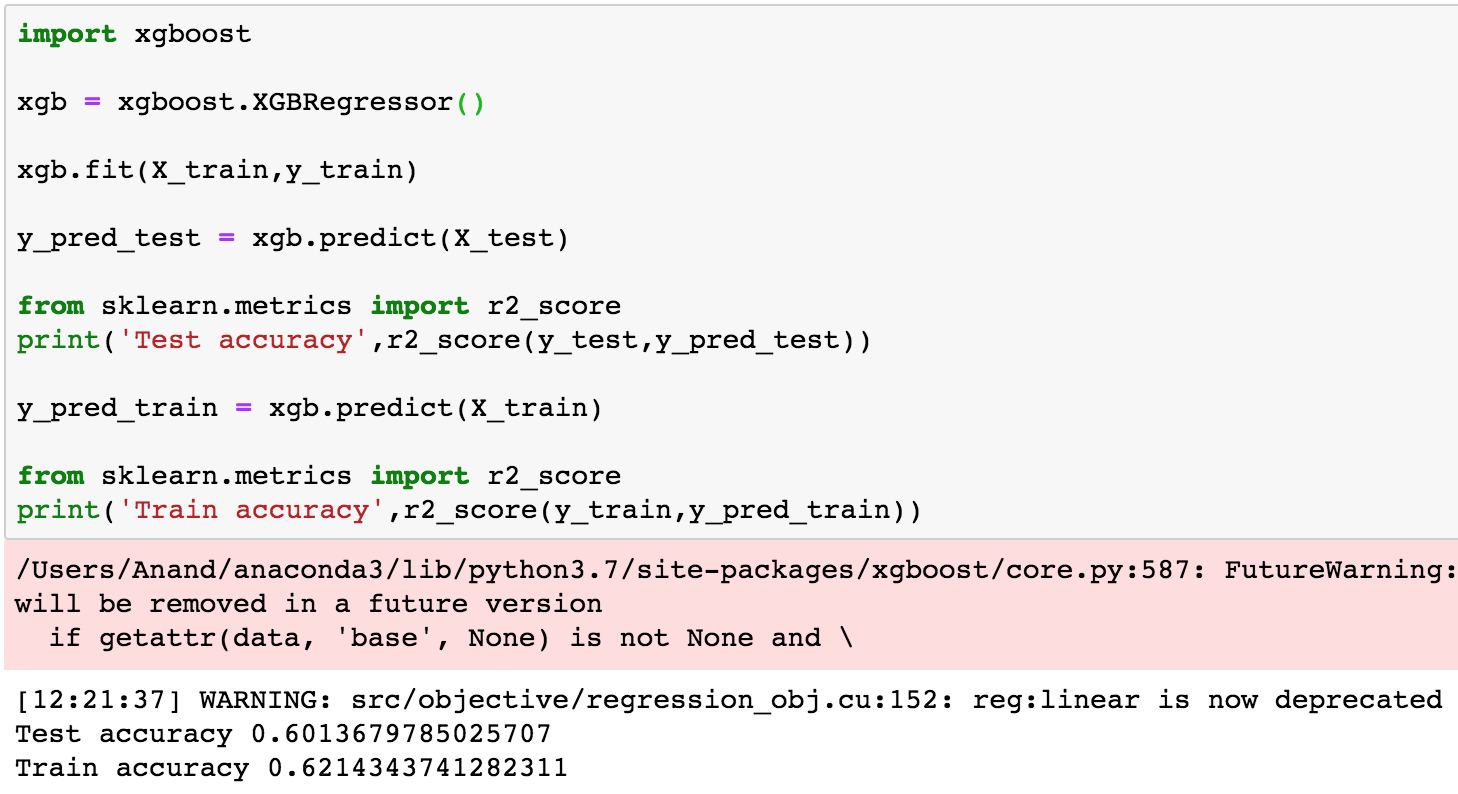
From the best estimator with the best parameters from *RandomizedSearchCV*, we predict,

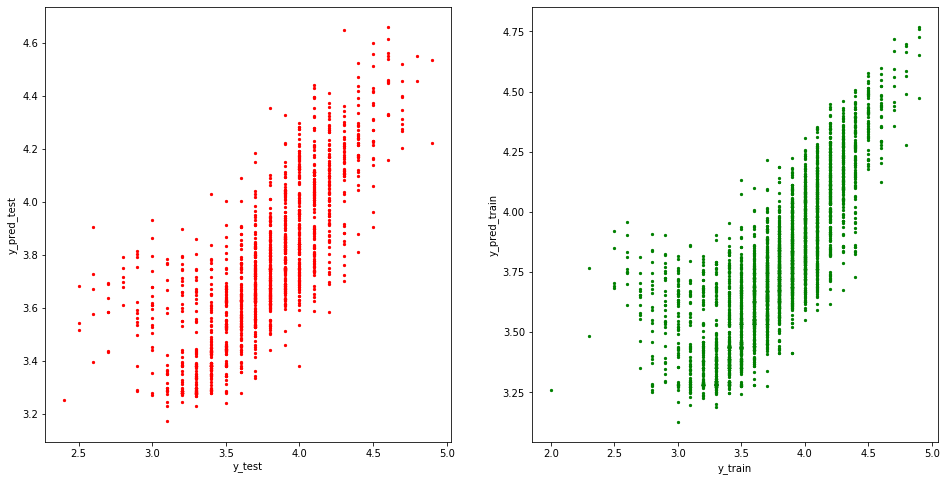
**

So, we have slightly better test accuracy which is **0.602** and train accuracy of **0.685.**

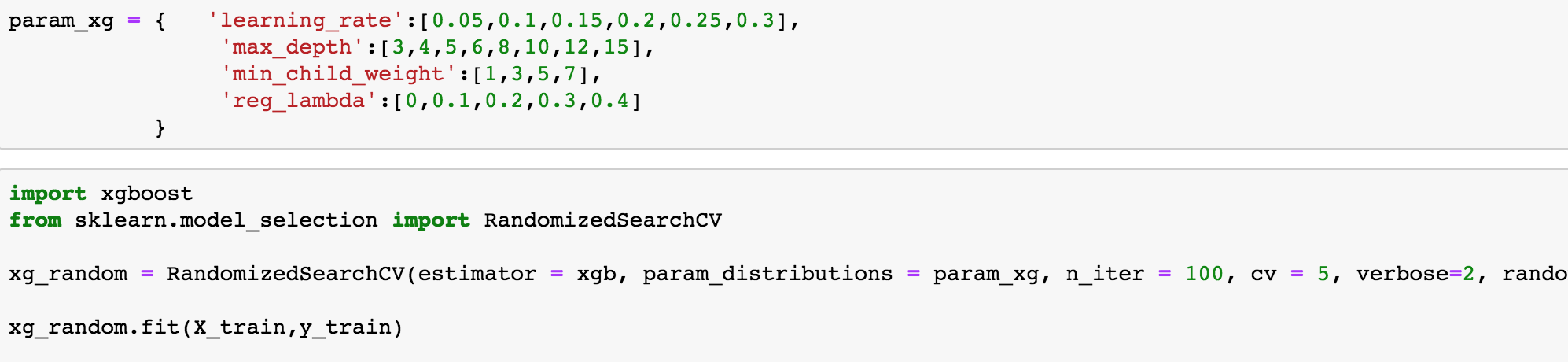


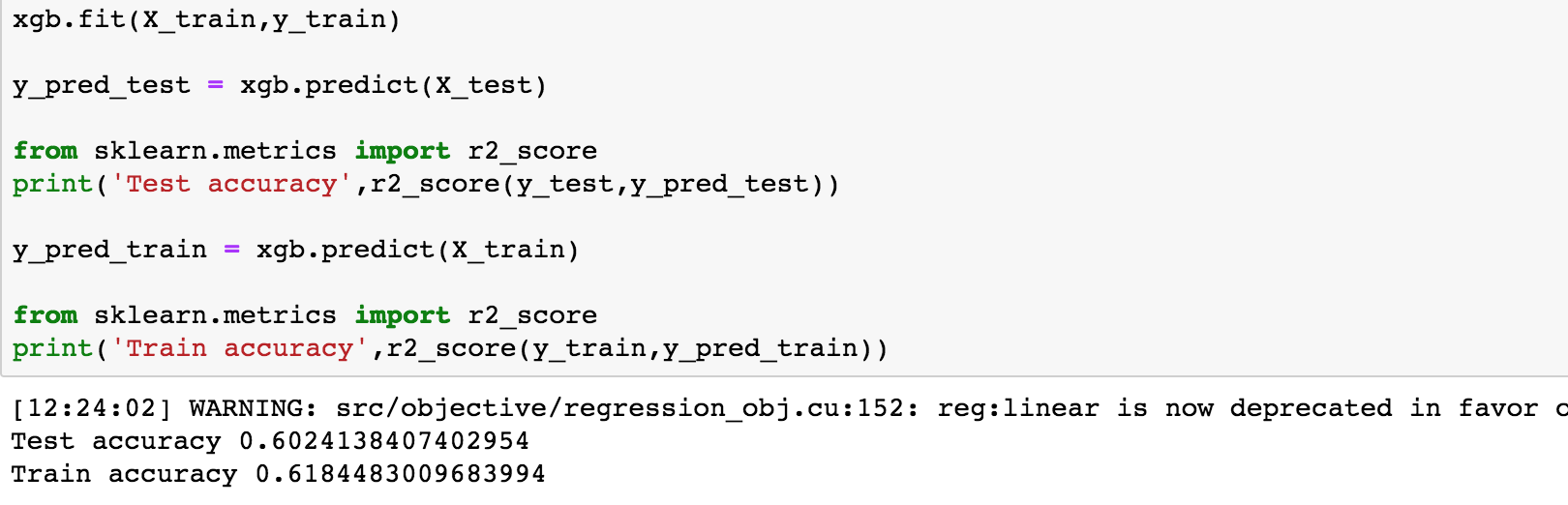
***XG Boost:***

******

XG Boost gives a test accuracy of **0.601** and train accuracy of **0.621.** 

Hyperparameter tuning with *RandomizedSearchCV* gives following results,





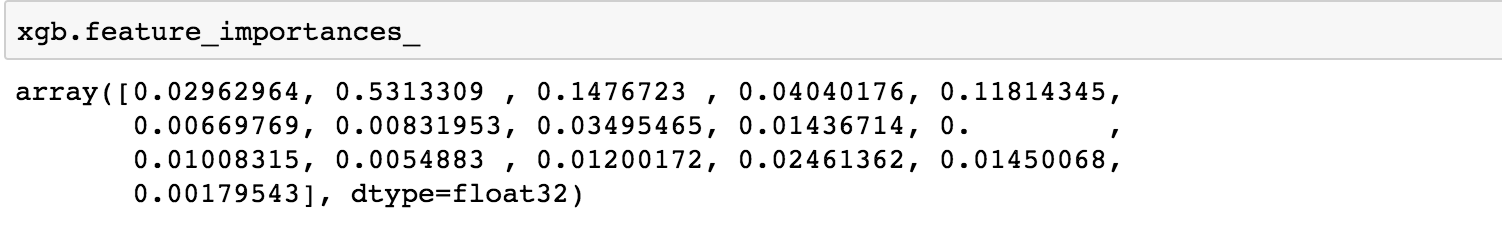
XG boost gives a test accuracy of **0.602** after hyperparameter tuning.

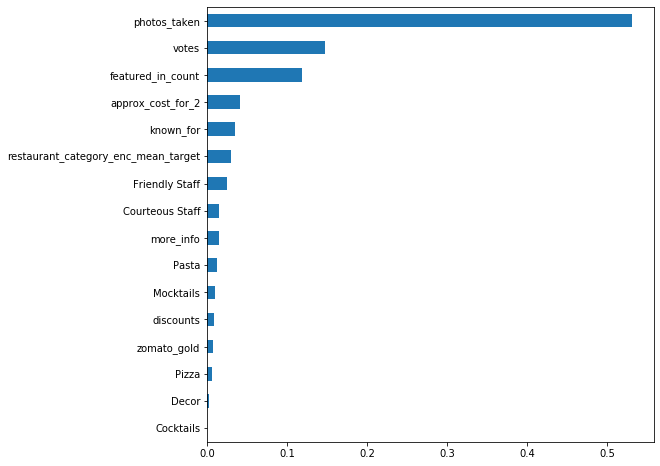
***Test Accuracy from Predictive modeling:***

So far ***RandomForestRegressor*** and ***XGBoost*** have a relatively higher test accuracy of ***0.60*** post hyperparameter tuning.

***Feature importance:***

Let's see the important features of the xgboost model,

******



From the above plot of feature importance, we see that features like ***photos\_taken, votes, featured\_in\_count*** and ***approx\_cost\_for\_2*** are significant features for predicting “***Rating***” of a restaurant.

***Future Development:*** As a part of future development and refinements,

* Update the web scraping to incorporate more data.
* Explore and incorporate more features like commercial rates etc.
* Explore packages like HYPEROPT for hyperparameter tuning.