INSAID Term 1 and 2 Project

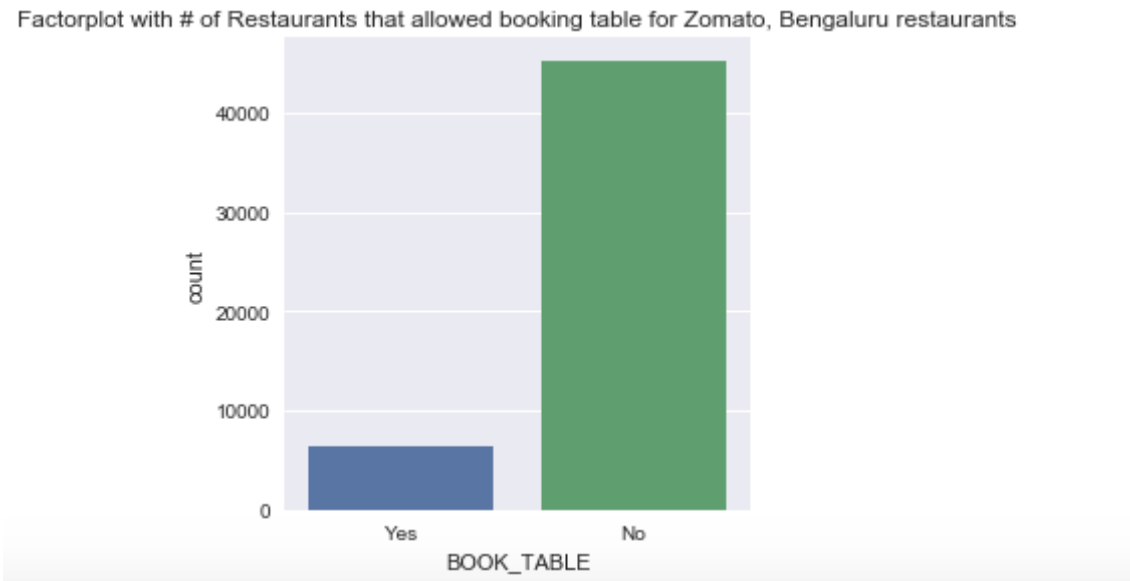
EDA Project
Zomato Restaurants, Bengaluru, India

Prepared by: Simran Khanna

July, 2019

BOOKING a TABLE AHEAD

Only 12.5% of restaurants on Zomato, Bengaluru allow booking a table ahead



BOOK_TABLE
Categorical

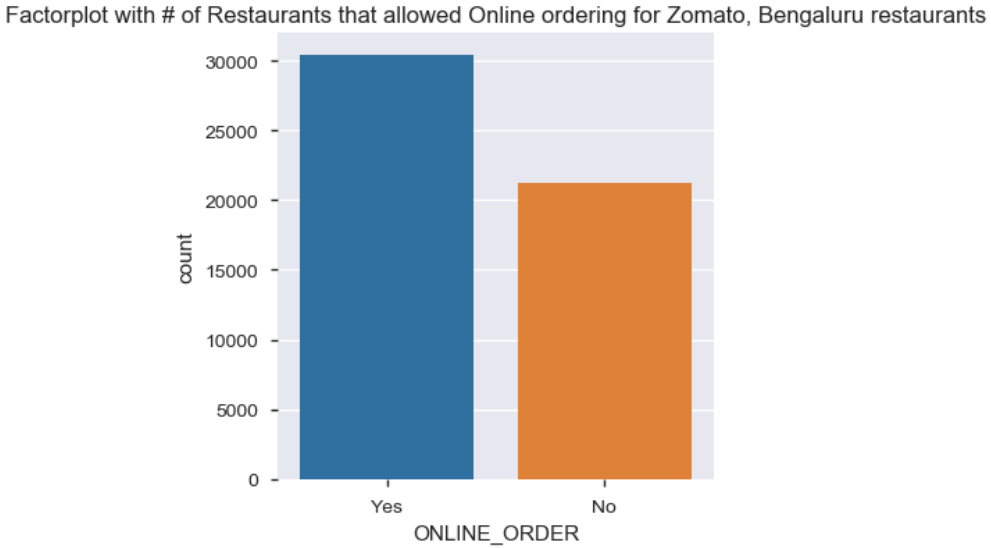
Distinct count	2
Unique (%)	0.0%
Missing (%)	0.0%
Missing (n)	0

Toggle details

Value	Count	Frequency (%)
No	45268	87.5%
Yes	6449	12.5%

ONLINE ORDERING

A Huge majority of ~58.9% of restaurants on Zomato, Bengaluru allow online ordering




```
print("A Huge majority of ~58.9% of restaurants on Zomato, Bengaluru allow online ordering")
```

```
A Huge majority of ~58.9% of restaurants on Zomato, Bengaluru allow online ordering
```

ONLINE_ORDER
Categorical

Distinct count	2
Unique (%)	0.0%
Missing (%)	0.0%
Missing (n)	0

Toggle details

Value	Count	Frequency (%)
Yes	30444	58.9%
No	21273	41.1%

RATINGS

Ratings data shows a Normal distribution with ~22.6% restaurants rated between 3.7-3.9/5 and ~4.3% New restaurants without a valid rating

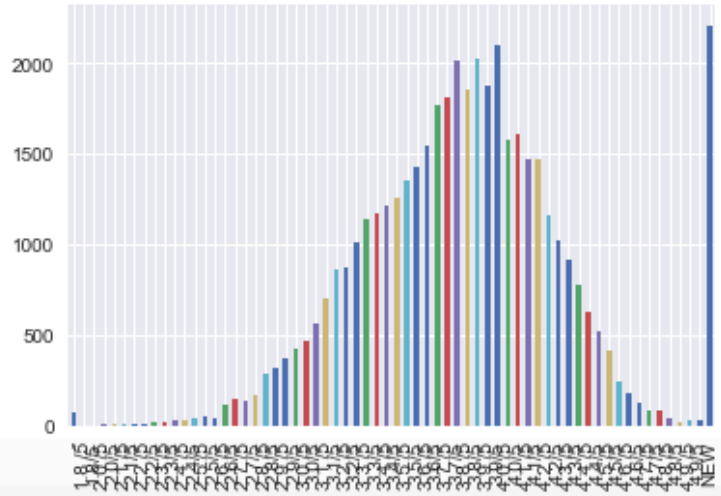
rate
Categorical

Distinct count	65
Unique (%)	0.1%
Missing (%)	15.0%
Missing (n)	7775

NEW	2208
3.9/5	2098
3.8/5	2022
Other values (61)	37614
(Missing)	7775

Toggle details

```
In [97]: zomato['RATE'].value_counts().sort_index().head(100).plot.bar()
Out[97]: <matplotlib.axes._subplots.AxesSubplot at 0x1a23935898>
```




```
In [67]: print("Ratings data shows a Normal distribution with -22.6% restaurants rated between 3.7-3.9/5 and -4.3% New restauran

Ratings data shows a Normal distribution with -22.6% restaurants rated between 3.7-3.9/5 and -4.3% New restaurants wi
thout a valid rating and 15% with missing ratings
```

RATINGS with Mode correction for missing data

Ratings data shows a Normal distribution with ~22.6% restaurants rated between 3.7-3.9/5 and ~19.3% New restaurants without a valid rating. The restaurants with missing ratings (15%) have been treated as new in the absence of any vintage data on restaurants.

RATE
Categorical

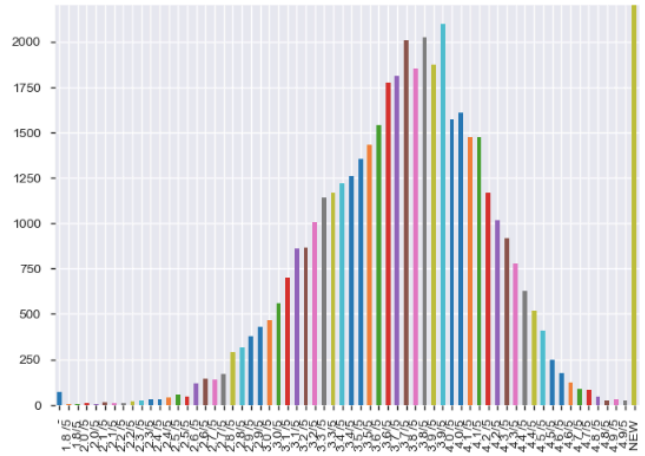
Distinct count	64
Unique (%)	0.1%
Missing (%)	0.0%
Missing (n)	0

Toggle details

Value	Count	Frequency (%)
NEW	9983	19.3%
3.9/5	2098	4.1%
3.8/5	2022	3.9%
3.7/5	2011	3.9%
3.9 /5	1874	3.6%
3.8 /5	1851	3.6%
3.7 /5	1810	3.5%
3.6/5	1773	3.4%
4.0/5	1609	3.1%
4.0 /5	1574	3.0%
Other values (54)	25112	48.6%

In [65]:
```
import pylab
pylab.ylim(0,2200)
zomato_fix['RATE'].value_counts().head(75).sort_index().plot.bar()
```

Out[65]: <matplotlib.axes._subplots.AxesSubplot at 0x1a592ea898>



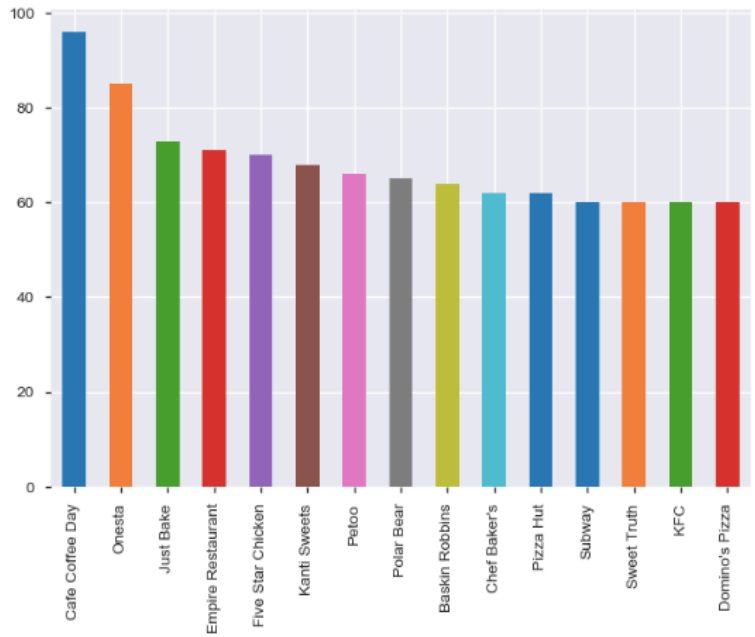
In [66]:
```
print("Ratings data shows a Normal distribution with ~22.6% restaurants rated between 3.7-3.9/5 and ~19.3% New restaura
```

```
Ratings data shows a Normal distribution with ~22.6% restaurants rated between 3.7-3.9/5 and ~19.3% New restaurants w
ithout a valid rating including the Resturants with missing ratings
```

Top 5 Restaurants by frequency

Top 5 restaurants by frequency include 'Cafe Coffee Day', 'Onesta', 'Just Bake','Empire Resturant' and 'Five Star Chicken'

```
In [16]: zomato['NAME'].value_counts().head(15).plot.bar()
Out[16]: <matplotlib.axes._subplots.AxesSubplot at 0x10d8f5860>
```




```
In [18]: "Top 5 restaurants by vote include 'Cafe Coffee Day', 'Onesta', 'Just Bake','Empire Resturant' and 'Five Star Chicken'")
         Top 5 restaurants by vote include 'Cafe Coffee Day', 'Onesta', 'Just Bake','Empire Resturant' and 'Five Star Chicken'
```

Approximate Cost per 2 People – Missing Value treatment -Mode

Approximate cost for 2 people – Before and After fixing missing values through mode replacement

APPROX_COST(FOR TWO PEOPLE)
Categorical

Distinct count	71
Unique (%)	0.1%
Missing (%)	0.7%
Missing (n)	346

Toggle details

Value	Count	Frequency (%)
300	7576	14.6%
400	6562	12.7%
500	4980	9.6%
200	4857	9.4%
600	3714	7.2%
250	2959	5.7%
800	2285	4.4%
150	2066	4.0%
700	1948	3.8%
350	1763	3.4%
Other values (60)	12661	24.5%

APPROX_COST(FOR TWO PEOPLE)
Categorical

Distinct count	70
Unique (%)	0.1%
Missing (%)	0.0%
Missing (n)	0

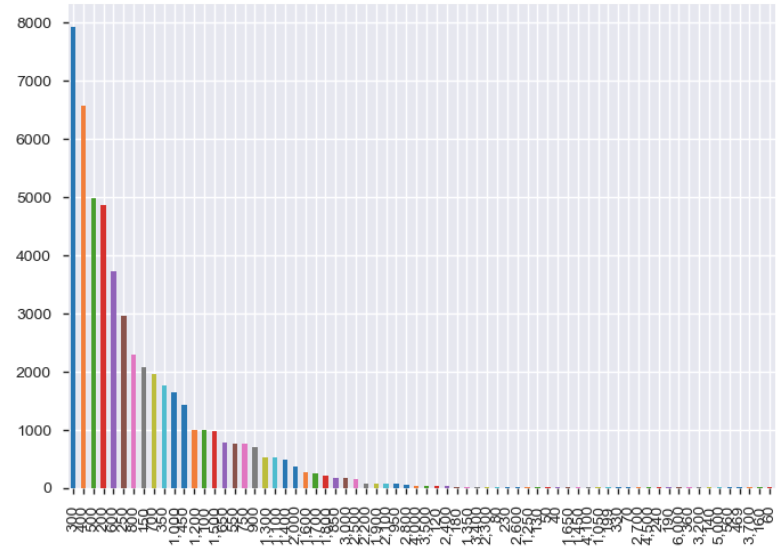
Toggle details

Value	Count	Frequency (%)
300	7922	15.3%
400	6562	12.7%
500	4980	9.6%
200	4857	9.4%
600	3714	7.2%
250	2959	5.7%
800	2285	4.4%
150	2066	4.0%
700	1948	3.8%
350	1763	3.4%
Other values (60)	12661	24.5%

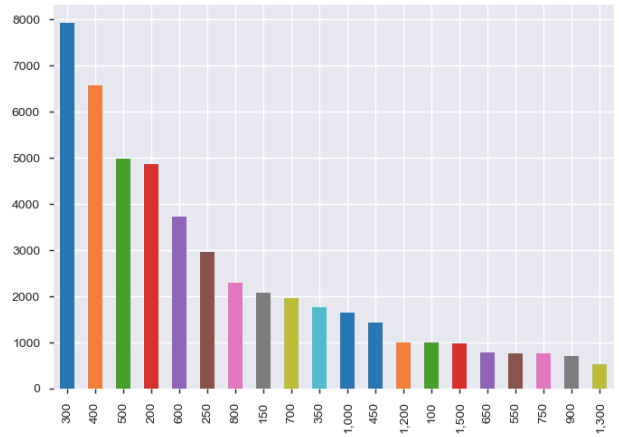
Approximate Cost per 2 People – Missing Value treatment -Mode

Data for Average cost for 2 people is left skewed and borders below INR 1000 for a majority of the restaurants

```
In [100]: zomato_fix['APPROX_COST(FOR TWO PEOPLE)'].value_counts().plot.bar()
Out[100]: <matplotlib.axes._subplots.AxesSubplot at 0x1a5c32da90>
```




```
In [101]: zomato_fix['APPROX_COST(FOR TWO PEOPLE)'].value_counts().head(20).plot.bar()
Out[101]: <matplotlib.axes._subplots.AxesSubplot at 0x1a5c55ef98>
```



CORRELATION ASSESSMENT
Conversion to Numeric

Conversion of data from Category/Character to Numeric to assess correlation across factors

In [110]:
```
zomato.head()
```

Out[110]:

	URL	ADDRESS	NAME	ONLINE_ORDER	BOOK_TABLE	RATE	VOTES	PHONE	LOCATION	REST_T
0	https://www.zomato.com/bangalore/jalsa-banasha...	942, 21st Main Road, 2nd Stage, Banashankari, ...	Jalsa	Yes	Yes	4.1/5	775	080 42297555\r\n+91 9743772233	Banashankari	C D
1	https://www.zomato.com/bangalore/spice-elephan...	2nd Floor, 80 Feet Road, Near Big Bazaar, 6th ...	Spice Elephant	Yes	No	4.1/5	787	080 41714161	Banashankari	C D
2	https://www.zomato.com/SanchurroBangalore?cont...	1112, Next to KIMS Medical College, 17th Cross...	San Churro Cafe	Yes	No	3.8/5	918	+91 9663487993	Banashankari	Cafe, C D
3	https://www.zomato.com/bangalore/addhuri-udupi...	1st Floor, Annakuteera, 3rd Stage, Banashankar...	Addhuri Udupi Bhojana	No	No	3.7/5	88	+91 9620009302	Banashankari	Quick
4	https://www.zomato.com/bangalore/grand-village...	10, 3rd Floor, Lakshmi Associates, Gandhi Baza...	Grand Village	No	No	3.8/5	166	+91 8026612447\r\n+91 9901210005	Basavanagudi	C D

In [108]:
```
zomato_new = zomato.copy(deep=True)   # creating new dataframe to make any datatype changes over it and keep the origin
from sklearn.preprocessing import LabelEncoder                                  # Label encoder is used to transform
number = LabelEncoder()                                                         # Here as our data set consists only
for i in zomato_new.columns:                                                      # in order to find the correlatio
    zomato_new[i] = number.fit_transform(zomato_new[i].astype('str'))
zomato_new.head()
```

Out[108]:

	URL	ADDRESS	NAME	ONLINE_ORDER	BOOK_TABLE	RATE	VOTES	PHONE	LOCATION	REST_TYPE	DISH_LIKED	CUISINES	APPROX_COST(FOR TWO PEOPLE)	R
0	22195	8016	3690	1	1	46	2094	13293	1	27	3651	2159	66	
1	41273	3844	7022	1	0	46	2109	13078	1	27	2964	952	66	
2	92	784	6499	1	0	40	2253	7145	1	22	1416	766	66	
3	1160	2515	199	0	0	38	2208	6770	1	78	2766	2555	42	

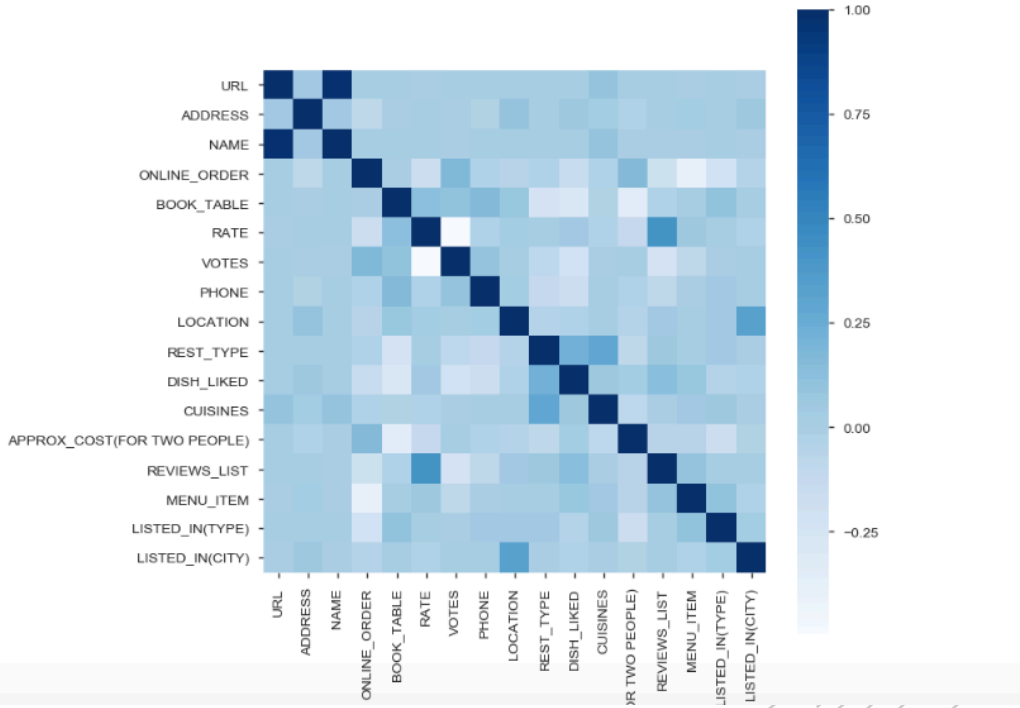
CORRELATION ASSESMENT
Factors driving Ratings

Correlation Assessment indicates that Reviews have high correlation to Ratings (~0.43)

```
In [104]: corr=zomato_new.corr()['RATE']
          corr[np.argsort(corr,axis=0)[::-1]]
```

```
Out[104]: RATE                           1.000000
          REVIEWS_LIST                   0.430843
          BOOK_TABLE                     0.120170
          MENU_ITEM                      0.063947
          DISH_LIKED                     0.043749
          LOCATION                       0.034516
          ADDRESS                        0.020579
          LISTED_IN(TYPE)                0.016577
          REST_TYPE                      0.007890
          NAME                           0.004367
          URL                           -0.000792
          LISTED_IN(CITY)               -0.028311
          CUISINES                      -0.032267
          PHONE                         -0.036015
          APPROX_COST(FOR TWO PEOPLE)   -0.124718
          ONLINE_ORDER                  -0.167062
          VOTES                         -0.493170
          Name: RATE, dtype: float64
```

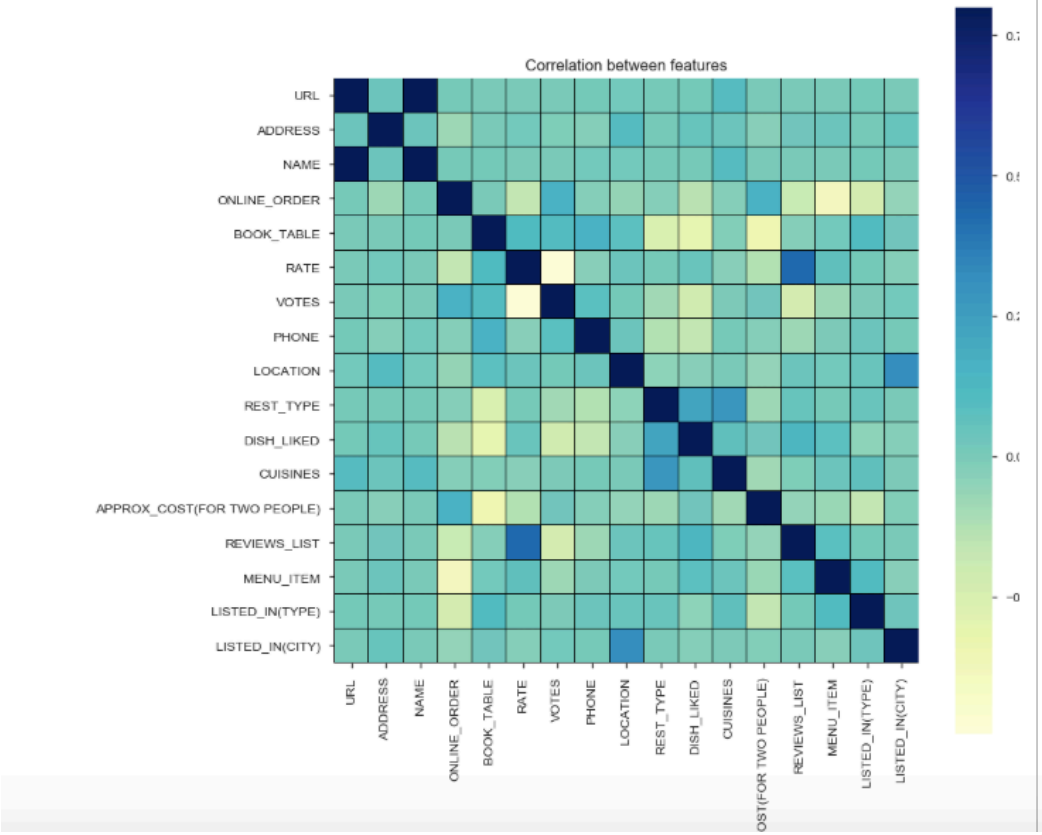
```
In [105]: features_correlation = zomato_new.corr()
          plt.figure(figsize=(8,8))
          sns.heatmap(features_correlation,vmax=1,square=True,annot=False,cmap='Blues')
          plt.show()
```



CORRELATION ASSESMENT
Factors driving Ratings

Correlation Assessment indicates that Reviews have high correlation to Ratings (~0.43)

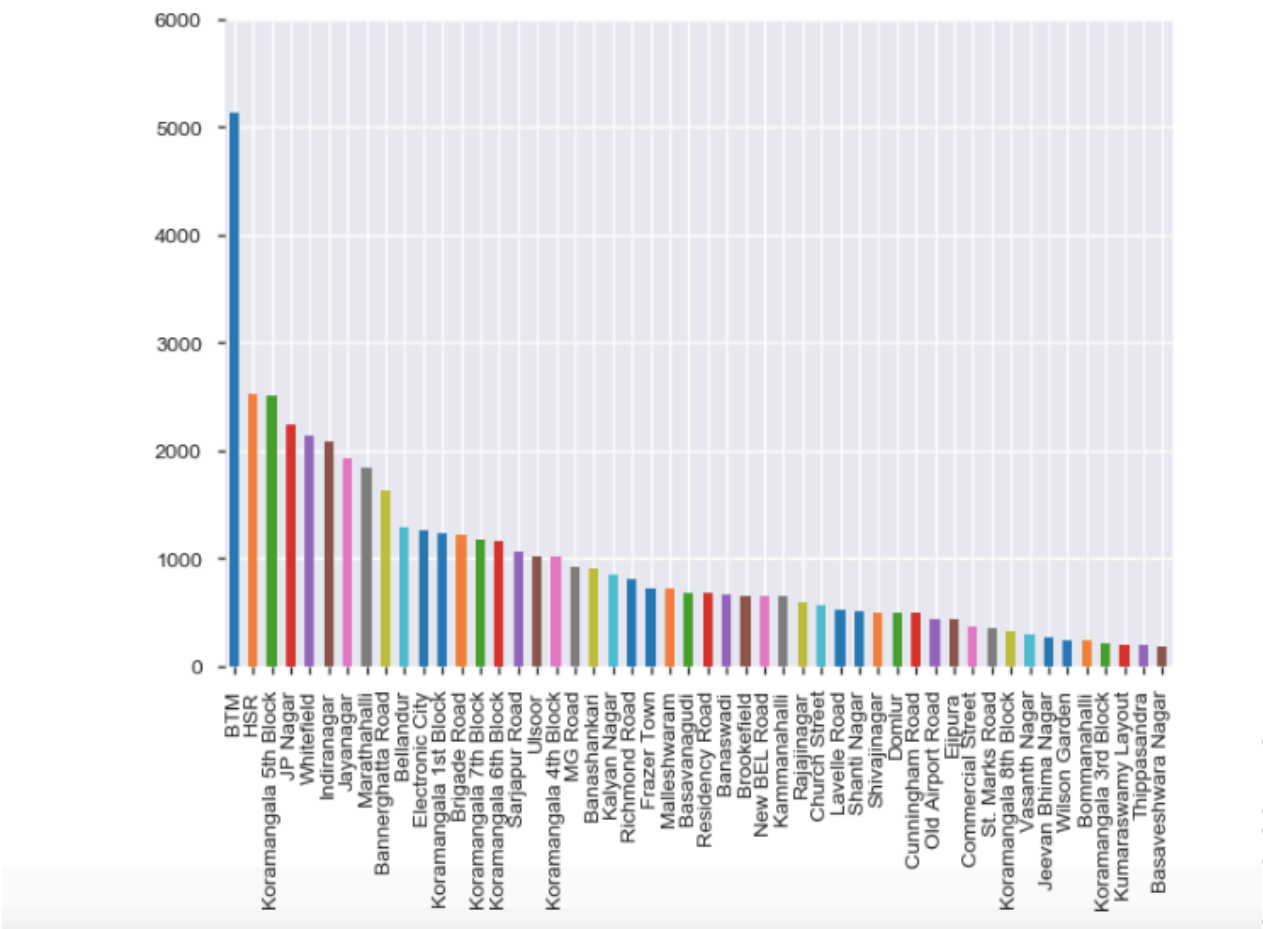
```
In [107]: corr = zomato_new.corr()
          plt.figure(figsize=(10,10))
          sns.heatmap(corr,vmax=.8,linewidth=.01, square = True, cmap='YlGnBu',linecolor ='black')
          plt.title('Correlation between features')

Out[107]: Text(0.5,1,'Correlation between features')
```



Top Restaurant Locations

The top 50 locations for restaurants is as per the list in the graph with BTM, HSR and Koramangala 5thblock topping the list

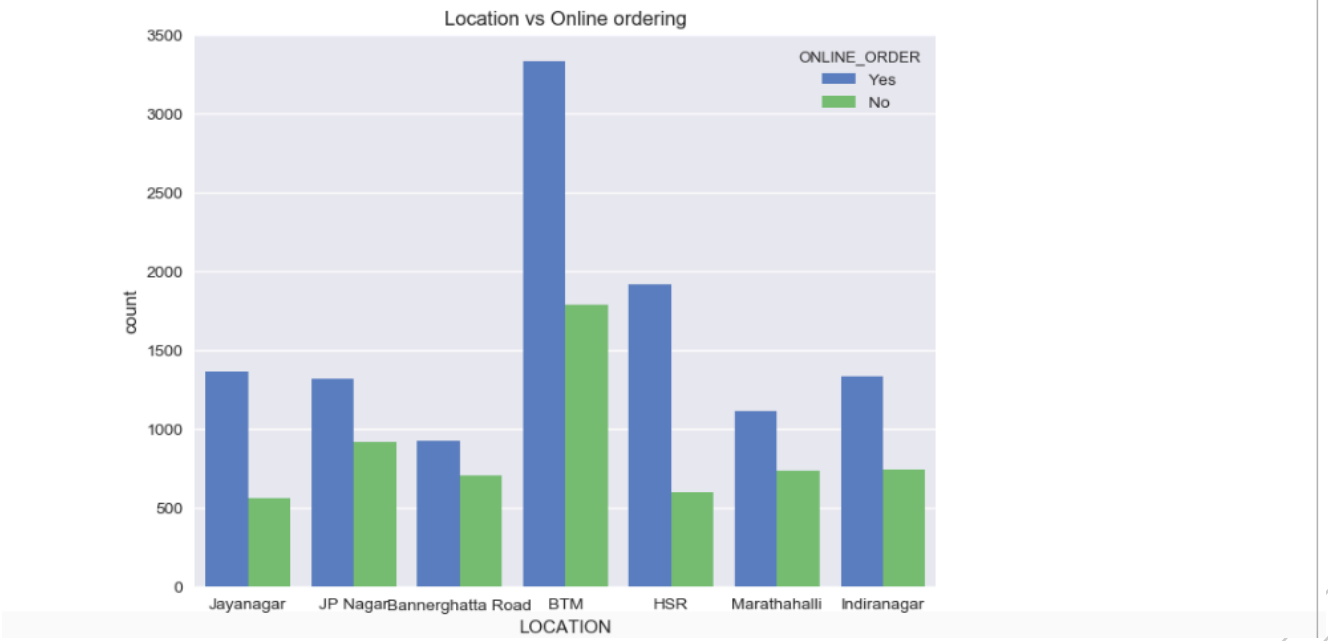
```
In [118]: import pylab
          pylab.ylim(0,6000)
          zomato_fix['LOCATION'].value_counts().head(50).plot.bar()

Out[118]: <matplotlib.axes._subplots.AxesSubplot at 0x1a687082b0>
```



Top Restaurant Locations by Online facility availability

Top restaurant locations typically offer online ordering facility

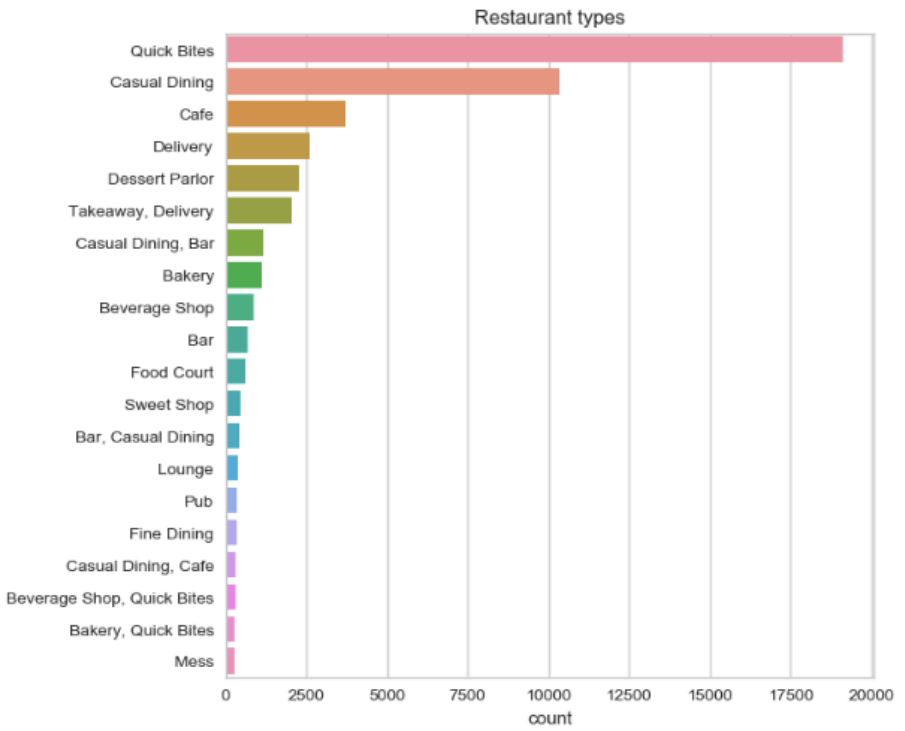
```
In [147]: f = (zomato_fix
                .loc[zomato_fix['LOCATION'].isin(['BTM', 'HSR','JP Nagar','Indiranagar','Jayanagar','Marathahalli','Bannerghat
          fig,ax =plt.subplots(figsize=(8,6))
          sns.countplot(data = f,x = 'LOCATION', hue='ONLINE_ORDER')
          plt.title('Location vs Online ordering')

Out[147]: Text(0.5,1,'Location vs Online ordering')
```



Top Restaurant Types

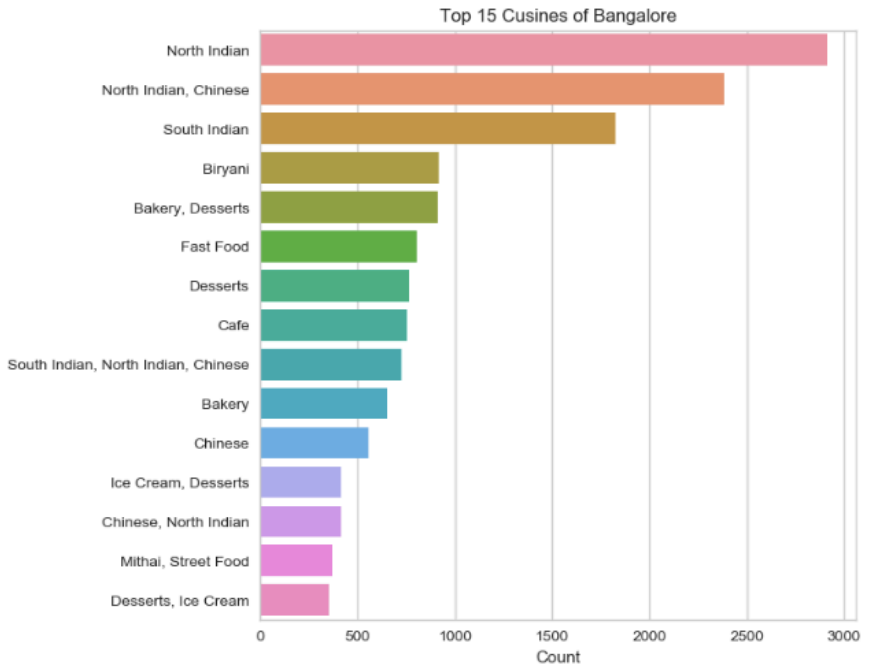
The top 20 types of restaurants are as listed in the graph

```
In [166]: plt.figure(figsize=(7,7))
          rest=zomato_fix['REST_TYPE'].value_counts()[:20]
          sns.barplot(rest,rest.index)
          plt.title("Restaurant types")
          plt.xlabel("count")

Out[166]: Text(0.5,0,'count')
```



Top 15 cuisines at Zomato, Bengaluru

```
In [185]: plt.figure(figsize=(7,7))
          cuisines=zomato_fix['CUISINES'].value_counts()[:15]
          sns.barplot(cuisines,cuisines.index)
          plt.xlabel('Count')
          plt.title("Top 15 Cusines of Bangalore")

Out[185]: Text(0.5,1,'Top 15 Cusines of Bangalore')
```



APPENDIX

Pandas Profiling
Quantitative and Qualitative Summary of the data

Overview

Dataset info

Number of variables	17
Number of observations	51717
Total Missing (%)	4.3%
Total size in memory	6.7 MiB
Average record size in memory	136.0 B

Variables types

Numeric	1
Categorical	15
Boolean	0
Date	0
Text (Unique)	1
Rejected	0
Unsupported	0

Warnings

- ADDRESS has a high cardinality: 11495 distinct values Warning
- APPROX_COST(FOR TWO PEOPLE) has a high cardinality: 71 distinct values Warning
- CUISINES has a high cardinality: 2724 distinct values Warning
- DISH_LIKED has 28078 / 54.3% missing values Missing
- DISH_LIKED has a high cardinality: 5272 distinct values Warning
- LOCATION has a high cardinality: 94 distinct values Warning
- MENU_ITEM has a high cardinality: 9098 distinct values Warning
- NAME has a high cardinality: 8792 distinct values Warning
- PHONE has 1208 / 2.3% missing values Missing
- PHONE has a high cardinality: 14927 distinct values Warning
- RATE has 7775 / 15.0% missing values Missing
- RATE has a high cardinality: 65 distinct values Warning
- REST_TYPE has a high cardinality: 94 distinct values Warning
- REVIEWS_LIST has a high cardinality: 22513 distinct values Warning
- VOTES has 10027 / 19.4% zeros Zeros

http://localhost:8888/view/Documents/Data%20Science/INSAID/zomazomato_before_preprocessing.htmlto_before_preprocessing.html