DecsionTreeRegrssor

October 17, 2024

[2]:

```
import seaborn as sns, matplotlib.pyplot as plt
import pandas as pd, numpy as np
```

[3]:

```
zomato = pd.read_csv('data.csv', on_bad_lines='skip', encoding='utf-8',␣
  ↪skiprows=[4271])
zomato.head()
```

[3]:

```
                                                 url  \
0  https://www.zomato.com/bangalore/jalsa-banasha…
1  https://www.zomato.com/bangalore/spice-elephan…
2  https://www.zomato.com/SanchurroBangalore?cont…
3  https://www.zomato.com/bangalore/addhuri-udupi…
4  https://www.zomato.com/bangalore/grand-village…

                                             address                   name  \
0  942, 21st Main Road, 2nd Stage, Banashankari, …                  Jalsa
1  2nd Floor, 80 Feet Road, Near Big Bazaar, 6th …         Spice Elephant
2  1112, Next to KIMS Medical College, 17th Cross…        San Churro Cafe
3  1st Floor, Annakuteera, 3rd Stage, Banashankar…  Addhuri Udupi Bhojana
4  10, 3rd Floor, Lakshmi Associates, Gandhi Baza…          Grand Village

  online_order book_table   rate  votes                             phone  \
0          Yes        Yes  4.1/5    775    080 42297555\r\n+91 9743772233
1          Yes         No  4.1/5    787                      080 41714161
2          Yes         No  3.8/5    918                    +91 9663487993
3           No         No  3.7/5     88                    +91 9620009302
4           No         No  3.8/5    166  +91 8026612447\r\n+91 9901210005

       location            rest_type  \
0  Banashankari        Casual Dining
1  Banashankari        Casual Dining
2  Banashankari  Cafe, Casual Dining
3  Banashankari          Quick Bites
4  Basavanagudi        Casual Dining

                                          dish_liked  \
0  Pasta, Lunch Buffet, Masala Papad, Paneer Laja…
```

```
1  Momos, Lunch Buffet, Chocolate Nirvana, Thai G…
2  Churros, Cannelloni, Minestrone Soup, Hot Choc…
3                                           Masala Dosa
4                                   Panipuri, Gol Gappe

                         cuisines approx_cost(for two people)  \
0  North Indian, Mughlai, Chinese                         800
1     Chinese, North Indian, Thai                         800
2          Cafe, Mexican, Italian                         800
3      South Indian, North Indian                         300
4        North Indian, Rajasthani                         600

                                        reviews_list menu_item  \
0  [('Rated 4.0', 'RATED\n  A beautiful place to …        []
1  [('Rated 4.0', 'RATED\n  Had been here for din…        []
2  [('Rated 3.0', "RATED\n  Ambience is not that …        []
3  [('Rated 4.0', "RATED\n  Great food and proper…        []
4  [('Rated 4.0', 'RATED\n  Very good restaurant …        []

  listed_in(type) listed_in(city)
0          Buffet    Banashankari
1          Buffet    Banashankari
2          Buffet    Banashankari
3          Buffet    Banashankari
4          Buffet    Banashankari
```

```
[4]: zomato.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4270 entries, 0 to 4269
Data columns (total 17 columns):
 #   Column                       Non-Null Count  Dtype
---  ------                       --------------  -----
 0   url                          4270 non-null   object
 1   address                      4270 non-null   object
 2   name                         4270 non-null   object
 3   online_order                 4270 non-null   object
 4   book_table                   4270 non-null   object
 5   rate                         3736 non-null   object
 6   votes                        4270 non-null   int64
 7   phone                        4207 non-null   object
 8   location                     4269 non-null   object
 9   rest_type                    4252 non-null   object
 10  dish_liked                   1806 non-null   object
 11  cuisines                     4265 non-null   object
 12  approx_cost(for two people)  4263 non-null   object
 13  reviews_list                 4270 non-null   object
 14  menu_item                    4270 non-null   object
 15  listed_in(type)              4270 non-null   object
 16  listed_in(city)              4270 non-null   object
dtypes: int64(1), object(16)
memory usage: 567.2+ KB
```

[5]:
```
zomato.isnull().sum()
```

[5]:
```
url                               0
address                           0
name                              0
online_order                      0
book_table                        0
rate                            534
votes                             0
phone                            63
location                          1
rest_type                        18
dish_liked                     2464
cuisines                          5
approx_cost(for two people)       7
reviews_list                      0
menu_item                         0
listed_in(type)                   0
listed_in(city)                   0
dtype: int64
```

[6]:
```
zomato.shape
```

[6]:
```
(4270, 17)
```

[7]:
```
zomato.dropna(how = 'any', inplace = True)
```

[8]:
```
zomato.shape
```

[8]:
```
(1782, 17)
```

[9]:
```
zomato.isnull().sum()
```

[9]:
```
url                            0
address                        0
name                           0
online_order                   0
book_table                     0
rate                           0
votes                          0
phone                          0
```

```
location                       0
rest_type                      0
dish_liked                     0
cuisines                       0
approx_cost(for two people)    0
reviews_list                   0
menu_item                      0
listed_in(type)                0
listed_in(city)                0
dtype: int64
```

[10]:

```
zomato = zomato.
 ↪drop(['url','address','phone','dish_liked','reviews_list','menu_item','listed_in(type)'],␣
 ↪axis = 1)
zomato.head()
```

[10]:

```
                    name online_order book_table   rate  votes      location  \
0                  Jalsa          Yes        Yes  4.1/5    775  Banashankari
1         Spice Elephant          Yes         No  4.1/5    787  Banashankari
2        San Churro Cafe          Yes         No  3.8/5    918  Banashankari
3  Addhuri Udupi Bhojana           No         No  3.7/5     88  Banashankari
4          Grand Village           No         No  3.8/5    166  Basavanagudi

              rest_type                        cuisines  \
0         Casual Dining  North Indian, Mughlai, Chinese
1         Casual Dining     Chinese, North Indian, Thai
2   Cafe, Casual Dining         Cafe, Mexican, Italian
3           Quick Bites      South Indian, North Indian
4         Casual Dining        North Indian, Rajasthani

  approx_cost(for two people) listed_in(city)
0                         800    Banashankari
1                         800    Banashankari
2                         800    Banashankari
3                         300    Banashankari
4                         600    Banashankari
```

[11]:

```
# check for duplicates

zomato.duplicated().sum()
```

[11]:

```
512
```

[12]:

```
zomato.drop_duplicates(inplace = True)
```

[13]:

```
zomato.shape
```

```
[13]: (1270, 10)
```

```
[14]: zomato = zomato.rename(columns = {'approx_cost(for two people)':
       ↪'cost','listed_in(city)':'city'})
```

```
[15]: zomato.head()
```

```
[15]:                    name online_order book_table   rate  votes      location  \
      0                 Jalsa          Yes        Yes  4.1/5    775  Banashankari
      1        Spice Elephant          Yes         No  4.1/5    787  Banashankari
      2       San Churro Cafe          Yes         No  3.8/5    918  Banashankari
      3 Addhuri Udupi Bhojana           No         No  3.7/5     88  Banashankari
      4         Grand Village           No         No  3.8/5    166  Basavanagudi

                   rest_type                        cuisines cost          city
      0        Casual Dining  North Indian, Mughlai, Chinese  800  Banashankari
      1        Casual Dining     Chinese, North Indian, Thai  800  Banashankari
      2  Cafe, Casual Dining         Cafe, Mexican, Italian  800  Banashankari
      3          Quick Bites      South Indian, North Indian  300  Banashankari
      4        Casual Dining        North Indian, Rajasthani  600  Banashankari
```

```
[16]: zomato.cost.value_counts()
```

```
[16]: 400      183
      600      125
      500      125
      300      114
      800      103
      200       62
      700       54
      250       54
      450       49
      750       47
      1,000     43
      650       41
      150       39
      1,200     34
      550       30
      350       29
      100       27
      900       22
      1,500     20
      1,300     13
      1,100     12
      850        9
      1,600      7
      1,400      6
```

```
950        5
1,700      4
1,800      4
1,350      2
2,000      2
230        1
2,200      1
1,900      1
180        1
330        1
Name: cost, dtype: int64
```

[17]:

```
zomato['cost'] = zomato['cost'].astype(str)
zomato['cost'] = zomato['cost'].apply(lambda x :x.replace(',',''))
zomato['cost'] = zomato['cost'].astype(float)
```

[18]:

```
zomato.cost.dtype
```

[18]:

```
dtype('float64')
```

[19]:

```
zomato['rate'].unique()
```

[19]:

```
array(['4.1/5', '3.8/5', '3.7/5', '4.6/5', '4.0/5', '4.2/5', '3.9/5',
       '3.0/5', '3.6/5', '2.8/5', '4.4/5', '3.1/5', '4.3/5', '2.6/5',
       '3.3/5', '3.5/5', '3.8 /5', '3.2/5', '4.5/5', '2.5/5', '2.9/5',
       '3.4/5', '2.7/5', '4.7/5', 'NEW', '2.4/5', '2.2/5', '2.3/5',
       '4.8/5', '3.9 /5', '4.2 /5', '4.0 /5', '4.1 /5', '2.9 /5',
       '2.7 /5', '2.5 /5', '2.6 /5', '4.5 /5', '4.3 /5', '3.7 /5',
       '4.4 /5', '4.9/5'], dtype=object)
```

[20]:

```
zomato.rate.value_counts()
```

[20]:

```
3.9/5     188
3.8/5     178
4.0/5     148
4.1/5     141
3.7/5     105
4.2/5      90
3.6/5      56
4.3/5      48
4.4/5      36
3.5/5      29
3.0/5      28
2.9/5      25
3.2/5      23
2.7/5      14
3.4/5      14
```

```
3.3/5      14
2.8/5      14
2.6/5      12
4.5/5      12
3.1/5      12
4.1 /5     12
4.2 /5      8
4.6/5       8
NEW         7
4.0 /5      7
2.4/5       5
2.5/5       5
4.7/5       5
2.3/5       4
3.8 /5      4
3.9 /5      4
4.3 /5      3
4.8/5       2
2.9 /5      1
2.7 /5      1
2.5 /5      1
2.6 /5      1
4.5 /5      1
2.2/5       1
3.7 /5      1
4.4 /5      1
4.9/5       1
Name: rate, dtype: int64
```

[42]:
```
zomato = zomato.loc[zomato.rate != 'NEW']
```

[48]:
```
remove_slash = lambda x:x.replace('/5','')
```

[50]:
```
zomato.rate  = zomato.rate.apply(remove_slash).str.strip().astype('float')
```

[52]:
```
zomato.head()
```

[52]:
```
                     name online_order book_table  rate  votes     location  \
0                   Jalsa          Yes        Yes   4.1    775  Banashankari
1          Spice Elephant          Yes         No   4.1    787  Banashankari
2         San Churro Cafe          Yes         No   3.8    918  Banashankari
3   Addhuri Udupi Bhojana           No         No   3.7     88  Banashankari
4           Grand Village           No         No   3.8    166  Basavanagudi

            rest_type                        cuisines   cost         city
0       Casual Dining  North Indian, Mughlai, Chinese  800.0  Banashankari
1       Casual Dining     Chinese, North Indian, Thai  800.0  Banashankari
```

```
2  Cafe, Casual Dining          Cafe, Mexican, Italian  800.0  Banashankari
3          Quick Bites      South Indian, North Indian  300.0  Banashankari
4        Casual Dining        North Indian, Rajasthani  600.0  Banashankari
```

```
[54]: zomato.online_order.value_counts()
```

```
[54]: Yes    1005
      No      258
      Name: online_order, dtype: int64
```

```
[56]: zomato.online_order.replace(('Yes','No'), (1,0), inplace = True)
```

```
[58]: zomato.online_order.value_counts()
```

```
[58]: 1    1005
      0     258
      Name: online_order, dtype: int64
```

```
[60]: zomato.book_table.value_counts()
```

```
[60]: No     1056
      Yes     207
      Name: book_table, dtype: int64
```

```
[62]: zomato.book_table.replace(('Yes','No'), (1,0), inplace = True)
```

```
[64]: zomato.book_table.value_counts()
```

```
[64]: 0    1056
      1     207
      Name: book_table, dtype: int64
```

```
[66]: zomato.head()
```

```
[66]:                    name  online_order  book_table  rate  votes      location  \
      0                  Jalsa             1           1   4.1    775  Banashankari
      1         Spice Elephant             1           0   4.1    787  Banashankari
      2        San Churro Cafe             1           0   3.8    918  Banashankari
      3  Addhuri Udupi Bhojana             0           0   3.7     88  Banashankari
      4          Grand Village             0           0   3.8    166  Basavanagudi

                   rest_type                        cuisines   cost         city
      0        Casual Dining  North Indian, Mughlai, Chinese  800.0  Banashankari
      1        Casual Dining     Chinese, North Indian, Thai  800.0  Banashankari
      2  Cafe, Casual Dining          Cafe, Mexican, Italian  800.0  Banashankari
      3          Quick Bites      South Indian, North Indian  300.0  Banashankari
      4        Casual Dining        North Indian, Rajasthani  600.0  Banashankari
```

[131]:

```
zomato.city.unique()
```

[131]:

```
array(['Banashankari', 'Bannerghatta Road', 'Basavanagudi', 'Bellandur'],
      dtype=object)
```

[72]:

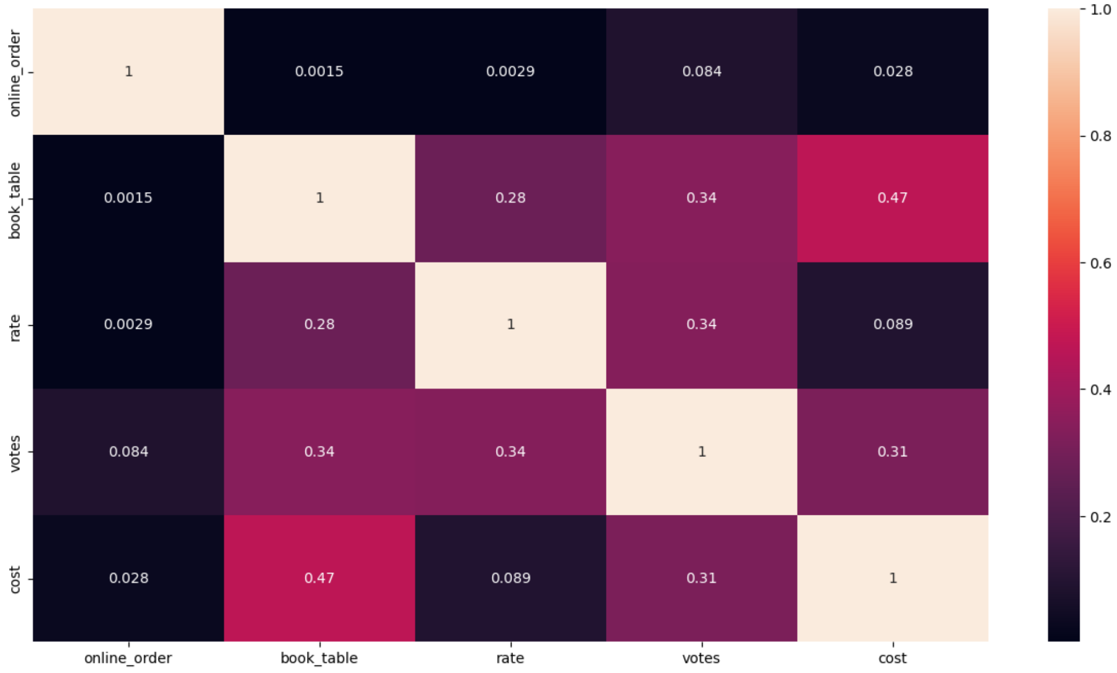
```
zomato1 = zomato[['online_order','book_table','rate','votes','cost']]
corr = zomato1.corr(method = 'kendall')
```

[74]:

```
plt.figure(figsize = (15,8))
sns.heatmap(corr, annot = True)
```

[74]:

```
<Axes: >
```



[80]:

```
from sklearn.model_selection import train_test_split

X = zomato1.drop(['cost'], axis = 1)
y = zomato1['cost']
```

[82]:

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2,␣
 ↪random_state = 42)
X_train.shape, X_test.shape, y_train.shape, y_test.shape
```

[82]:

```
((1010, 4), (253, 4), (1010,), (253,))
```

[88]:

```
from sklearn.linear_model import LinearRegression
reg = LinearRegression()
reg.fit(X_train, y_train)
```

[88]:

```
LinearRegression()
```

[92]:

```
y_pred = reg.predict(X_test)
y_pred[:10]
```

[92]:

```
array([ 492.59352764, 1041.02515609,  464.25906727,  428.40310357,
        480.02195083, 1113.97866557,  454.5465854 ,  458.29936619,
       1079.71302373,  471.14487333])
```

[94]:

```
from sklearn.metrics import r2_score
r2_score(y_test, y_pred)
```

[94]:

```
0.44503765283086094
```

[98]:

```
from sklearn.tree import DecisionTreeRegressor
dt = DecisionTreeRegressor(min_samples_leaf = 0.0001)
dt.fit(X_train, y_train)
```

[98]:

```
DecisionTreeRegressor(min_samples_leaf=0.0001)
```

[100]:

```
y_predit = dt.predict(X_test)
y_predit[:10]
```

[100]:

```
array([ 400.,  800.,  600., 1500.,  300., 1200.,  750.,  600.,  800.,
        750.])
```

[104]:

```
r2_score(y_test,y_predit)
```

[104]:

```
0.22074361227896666
```

[ ]:

```
from sklearn.ensemble import RandomForestRegressor
rf = RandomForestRegressor(n_estimators = 500, random_state = 300,␣
 ↪min_samples_leaf = 0.001)
rf.fit(X_train, y_train)
```

[111]:

```
y_predited = rf.predict(X_test)
y_predited[:10]
```

[111]:

```
array([ 484.19519841,  939.82293651,  575.0147619 ,  507.98746032,
        485.74130952, 1151.94849206,  545.54107143,  574.3420671 ,
        986.7784127 ,  621.04396465])
```

[113]:

```
r2_score(y_test,y_predited)
```

```
[113]: 0.49938015287202986
```

[115]:

```python
from sklearn.ensemble import ExtraTreesRegressor
et = ExtraTreesRegressor(n_estimators = 100)
et.fit(X_train, y_train)
```

```
[115]: ExtraTreesRegressor()
```

[119]:

```python
y_predited1 = et.predict(X_test)
y_predited1[:10]
```

```
[119]: array([ 506. ,  800. ,  620. ,  530.5,  329. , 1200. ,  746.5,  530.5,
               960. ,  750. ])
```

[121]:

```python
r2_score(y_test,y_predited1)
```

```
[121]: 0.3506691285210082
```

[ ]:

```python

```