Statistics Project

Ambika Prasad Swain

Table of contents

# 1. SMDM 2 Project

import warnings ; warnings.warn = lambda \*args,\*\*kwargs: None  
import numpy as np  
import scipy.stats as spst  
import pandas as pd  
import matplotlib.pyplot as plt  
import seaborn as sns  
import statsmodels.api as sm  
import statsmodels.formula.api as smf  
import duckdb as duc  
from sklearn.model\_selection import train\_test\_split  
from sklearn.ensemble import RandomForestRegressor  
from sklearn.metrics import mean\_squared\_error, r2\_score  
from scipy.stats import chi2\_contingency

We are going to use duckdb as our database engine. You can refer to the documentation here: [Documentation](https://duckdb.org/docs/). First, we will create a database called yelp.db in duckdb. We will persist this db on the harddrive because it might be too big to handle in your system RAM. That said, duckdb has in-RAM database capabilities as well. Of course, if you have a lot of RAM, you can use Pandas directly or use the in-RAM capabilities of duckdb. Once we execute the command below, we will have created the required db and connected to it as well. Once we have created the db, it will appear in the root folder of your project.

con = duc.connect("yelp.db")

Now, we will import the csv files into four different tables namely restos, resto\_reviews, users and user\_friends. This is a onetime execution. You don’t need to execute the codecell below everytime you do the analysis. Once the tables are created, they will persist on the db in your root folder.

restos table comes from restuarants\_train.csv (There is also a restuarants\_test.csv. You do not need to worry about this test data now. This will be used to conduct a hackathon to predict star ratings later if we get time. ). The columns correspond to business.json in https://www.yelp.com/dataset/documentation/main

resto\_reviews table comes from restaurant\_reviews.csv. The columns correspond to review.json in https://www.yelp.com/dataset/documentation/main

users table comes from user.csv. The columns correspond to user.json in https://www.yelp.com/dataset/documentation/main

user\_friends table comes from user\_friends\_full.csv. It was created by me. It shows the number of friends of each user in the social network of yelp.

con.sql("""  
 CREATE TABLE restos AS  
 FROM read\_csv('yelp.restaurants\_train.csv', sample\_size = -1);  
""")  
  
con.sql("""  
 CREATE TABLE resto\_reviews AS  
 FROM read\_csv('yelp.restaurant\_reviews.csv', sample\_size = -1);  
""")  
  
con.sql("""  
 CREATE TABLE users AS  
 FROM read\_csv('yelp.user.csv', sample\_size = -1);  
""")  
  
con.sql("""  
 CREATE TABLE user\_friends AS  
 FROM read\_csv('yelp.user\_friends\_full.csv', sample\_size = -1);  
""")

CatalogException: Catalog Error: Table with name "restos" already exists!  
[1;31m---------------------------------------------------------------------------[0m  
[1;31mCatalogException[0m Traceback (most recent call last)  
Cell [1;32mIn[28], line 1[0m  
[1;32m----> 1[0m con[38;5;241m.[39msql([38;5;124m"""[39m  
[0;32m 2[0m [38;5;124m CREATE TABLE restos AS[39m  
[0;32m 3[0m [38;5;124m FROM read\_csv([39m[38;5;124m'[39m[38;5;124myelp.restaurants\_train.csv[39m[38;5;124m'[39m[38;5;124m, sample\_size = -1);[39m  
[0;32m 4[0m [38;5;124m"""[39m)  
[0;32m 6[0m con[38;5;241m.[39msql([38;5;124m"""[39m  
[0;32m 7[0m [38;5;124m CREATE TABLE resto\_reviews AS[39m  
[0;32m 8[0m [38;5;124m FROM read\_csv([39m[38;5;124m'[39m[38;5;124myelp.restaurant\_reviews.csv[39m[38;5;124m'[39m[38;5;124m, sample\_size = -1);[39m  
[0;32m 9[0m [38;5;124m"""[39m)  
[0;32m 11[0m con[38;5;241m.[39msql([38;5;124m"""[39m  
[0;32m 12[0m [38;5;124m CREATE TABLE users AS[39m  
[0;32m 13[0m [38;5;124m FROM read\_csv([39m[38;5;124m'[39m[38;5;124myelp.user.csv[39m[38;5;124m'[39m[38;5;124m, sample\_size = -1);[39m  
[0;32m 14[0m [38;5;124m"""[39m)  
  
[1;31mCatalogException[0m: Catalog Error: Table with name "restos" already exists!

Let us look at the first 5 rows of the restos table.

con.sql("""  
 SELECT \* FROM restos   
 LIMIT 5;  
""")

┌──────────────────────┬─────────────────────┬───────────┬───┬──────────────────────┬──────────────────────┐  
│ name │ address │ city │ … │ attributes.Accepts… │ attributes.HairSpe… │  
│ varchar │ varchar │ varchar │ │ boolean │ varchar │  
├──────────────────────┼─────────────────────┼───────────┼───┼──────────────────────┼──────────────────────┤  
│ Oskar Blues Taproom │ 921 Pearl St │ Boulder │ … │ NULL │ NULL │  
│ Flying Elephants a… │ 7000 NE Airport Way │ Portland │ … │ NULL │ NULL │  
│ Bob Likes Thai Food │ 3755 Main St │ Vancouver │ … │ NULL │ NULL │  
│ Boxwood Biscuit │ 740 S High St │ Columbus │ … │ NULL │ NULL │  
│ Mr G's Pizza & Subs │ 474 Lowell St │ Peabody │ … │ NULL │ NULL │  
├──────────────────────┴─────────────────────┴───────────┴───┴──────────────────────┴──────────────────────┤  
│ 5 rows 60 columns (5 shown) │  
└──────────────────────────────────────────────────────────────────────────────────────────────────────────┘

We can convert the results from duckdb into a pandas dataframe by appending .df() function as given below. This pandas dataframe can be used for further analysis in python.

df\_restos = con.sql("""  
 SELECT \* FROM restos   
 LIMIT 5;  
""").df()  
df\_restos

|  | name | address | city | state | postal\_code | latitude | longitude | stars | review\_count | is\_open | ... | attributes.Smoking | attributes.DriveThru | attributes.BYOBCorkage | attributes.Corkage | attributes.RestaurantsCounterService | attributes.DietaryRestrictions | attributes.AgesAllowed | attributes.Open24Hours | attributes.AcceptsInsurance | attributes.HairSpecializesIn |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | Oskar Blues Taproom | 921 Pearl St | Boulder | CO | 80302 | 40.017544 | -105.283348 | 4.0 | 86 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 1 | Flying Elephants at PDX | 7000 NE Airport Way | Portland | OR | 97218 | 45.588906 | -122.593331 | 4.0 | 126 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 2 | Bob Likes Thai Food | 3755 Main St | Vancouver | BC | V5V | 49.251342 | -123.101333 | 3.5 | 169 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 3 | Boxwood Biscuit | 740 S High St | Columbus | OH | 43206 | 39.947007 | -82.997471 | 4.5 | 11 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 4 | Mr G's Pizza & Subs | 474 Lowell St | Peabody | MA | 01960 | 42.541155 | -70.973438 | 4.0 | 39 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |

Let us look at the columns in each of the tables.

con.sql("""  
 DESCRIBE restos  
""").df()

|  | column\_name | column\_type | null | key | default | extra |
| --- | --- | --- | --- | --- | --- | --- |
| 0 | name | VARCHAR | YES | None | None | None |
| 1 | address | VARCHAR | YES | None | None | None |
| 2 | city | VARCHAR | YES | None | None | None |
| 3 | state | VARCHAR | YES | None | None | None |
| 4 | postal\_code | VARCHAR | YES | None | None | None |
| 5 | latitude | DOUBLE | YES | None | None | None |
| 6 | longitude | DOUBLE | YES | None | None | None |
| 7 | stars | DOUBLE | YES | None | None | None |
| 8 | review\_count | BIGINT | YES | None | None | None |
| 9 | is\_open | BIGINT | YES | None | None | None |
| 10 | attributes.RestaurantsTableService | VARCHAR | YES | None | None | None |
| 11 | attributes.WiFi | VARCHAR | YES | None | None | None |
| 12 | attributes.BikeParking | VARCHAR | YES | None | None | None |
| 13 | attributes.BusinessParking | VARCHAR | YES | None | None | None |
| 14 | attributes.BusinessAcceptsCreditCards | VARCHAR | YES | None | None | None |
| 15 | attributes.RestaurantsReservations | VARCHAR | YES | None | None | None |
| 16 | attributes.WheelchairAccessible | VARCHAR | YES | None | None | None |
| 17 | attributes.Caters | VARCHAR | YES | None | None | None |
| 18 | attributes.OutdoorSeating | VARCHAR | YES | None | None | None |
| 19 | attributes.RestaurantsGoodForGroups | VARCHAR | YES | None | None | None |
| 20 | attributes.HappyHour | VARCHAR | YES | None | None | None |
| 21 | attributes.BusinessAcceptsBitcoin | BOOLEAN | YES | None | None | None |
| 22 | attributes.RestaurantsPriceRange2 | VARCHAR | YES | None | None | None |
| 23 | attributes.Ambience | VARCHAR | YES | None | None | None |
| 24 | attributes.HasTV | VARCHAR | YES | None | None | None |
| 25 | attributes.Alcohol | VARCHAR | YES | None | None | None |
| 26 | attributes.GoodForMeal | VARCHAR | YES | None | None | None |
| 27 | attributes.DogsAllowed | VARCHAR | YES | None | None | None |
| 28 | attributes.RestaurantsTakeOut | VARCHAR | YES | None | None | None |
| 29 | attributes.NoiseLevel | VARCHAR | YES | None | None | None |
| 30 | attributes.RestaurantsAttire | VARCHAR | YES | None | None | None |
| 31 | attributes.RestaurantsDelivery | VARCHAR | YES | None | None | None |
| 32 | categories | VARCHAR | YES | None | None | None |
| 33 | hours.Monday | VARCHAR | YES | None | None | None |
| 34 | hours.Tuesday | VARCHAR | YES | None | None | None |
| 35 | hours.Wednesday | VARCHAR | YES | None | None | None |
| 36 | hours.Thursday | VARCHAR | YES | None | None | None |
| 37 | hours.Friday | VARCHAR | YES | None | None | None |
| 38 | hours.Saturday | VARCHAR | YES | None | None | None |
| 39 | hours.Sunday | VARCHAR | YES | None | None | None |
| 40 | int\_business\_id | BIGINT | YES | None | None | None |
| 41 | attributes.GoodForKids | VARCHAR | YES | None | None | None |
| 42 | attributes.ByAppointmentOnly | BOOLEAN | YES | None | None | None |
| 43 | attributes | VARCHAR | YES | None | None | None |
| 44 | hours | VARCHAR | YES | None | None | None |
| 45 | attributes.Music | VARCHAR | YES | None | None | None |
| 46 | attributes.GoodForDancing | VARCHAR | YES | None | None | None |
| 47 | attributes.BestNights | VARCHAR | YES | None | None | None |
| 48 | attributes.BYOB | VARCHAR | YES | None | None | None |
| 49 | attributes.CoatCheck | VARCHAR | YES | None | None | None |
| 50 | attributes.Smoking | VARCHAR | YES | None | None | None |
| 51 | attributes.DriveThru | VARCHAR | YES | None | None | None |
| 52 | attributes.BYOBCorkage | VARCHAR | YES | None | None | None |
| 53 | attributes.Corkage | VARCHAR | YES | None | None | None |
| 54 | attributes.RestaurantsCounterService | BOOLEAN | YES | None | None | None |
| 55 | attributes.DietaryRestrictions | VARCHAR | YES | None | None | None |
| 56 | attributes.AgesAllowed | VARCHAR | YES | None | None | None |
| 57 | attributes.Open24Hours | BOOLEAN | YES | None | None | None |
| 58 | attributes.AcceptsInsurance | BOOLEAN | YES | None | None | None |
| 59 | attributes.HairSpecializesIn | VARCHAR | YES | None | None | None |

con.sql("""  
 DESCRIBE resto\_reviews  
""")

┌────────────────────┬─────────────┬─────────┬─────────┬─────────┬─────────┐  
│ column\_name │ column\_type │ null │ key │ default │ extra │  
│ varchar │ varchar │ varchar │ varchar │ varchar │ varchar │  
├────────────────────┼─────────────┼─────────┼─────────┼─────────┼─────────┤  
│ stars │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ useful │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ funny │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ cool │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ text │ VARCHAR │ YES │ NULL │ NULL │ NULL │  
│ date │ TIMESTAMP │ YES │ NULL │ NULL │ NULL │  
│ int\_business\_id │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ int\_user\_id │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ int\_rest\_review\_id │ BIGINT │ YES │ NULL │ NULL │ NULL │  
└────────────────────┴─────────────┴─────────┴─────────┴─────────┴─────────┘

con.sql("""  
 DESCRIBE users  
""")

┌────────────────────┬─────────────┬─────────┬─────────┬─────────┬─────────┐  
│ column\_name │ column\_type │ null │ key │ default │ extra │  
│ varchar │ varchar │ varchar │ varchar │ varchar │ varchar │  
├────────────────────┼─────────────┼─────────┼─────────┼─────────┼─────────┤  
│ review\_count │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ yelping\_since │ TIMESTAMP │ YES │ NULL │ NULL │ NULL │  
│ useful │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ funny │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ cool │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ elite │ VARCHAR │ YES │ NULL │ NULL │ NULL │  
│ fans │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ average\_stars │ DOUBLE │ YES │ NULL │ NULL │ NULL │  
│ compliment\_hot │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ compliment\_more │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ compliment\_profile │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ compliment\_cute │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ compliment\_list │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ compliment\_note │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ compliment\_plain │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ compliment\_cool │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ compliment\_funny │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ compliment\_writer │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ compliment\_photos │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ int\_user\_id │ BIGINT │ YES │ NULL │ NULL │ NULL │  
├────────────────────┴─────────────┴─────────┴─────────┴─────────┴─────────┤  
│ 20 rows 6 columns │  
└──────────────────────────────────────────────────────────────────────────┘

con.sql("""  
 DESCRIBE user\_friends  
""")

┌─────────────┬─────────────┬─────────┬─────────┬─────────┬─────────┐  
│ column\_name │ column\_type │ null │ key │ default │ extra │  
│ varchar │ varchar │ varchar │ varchar │ varchar │ varchar │  
├─────────────┼─────────────┼─────────┼─────────┼─────────┼─────────┤  
│ int\_user\_id │ BIGINT │ YES │ NULL │ NULL │ NULL │  
│ num\_friends │ BIGINT │ YES │ NULL │ NULL │ NULL │  
└─────────────┴─────────────┴─────────┴─────────┴─────────┴─────────┘

# 2. Descriptive analysis

Average stars received by restaurants.

con.sql("""  
 SELECT avg(stars) FROM restos  
""")

┌───────────────────┐  
│ avg(stars) │  
│ double │  
├───────────────────┤  
│ 3.527858606557377 │  
└───────────────────┘

We will select the stars column from restos and plot a histogram of stars.

restos\_stats = con.sql('''  
 SELECT AVG(stars) as mean\_rating,   
 MEDIAN(stars) as median\_rating,   
 STDDEV(stars) as std\_rating,   
 MIN(stars) as min\_rating,   
 MAX(stars) as max\_rating   
 FROM restos;  
''').df()  
restos\_stats

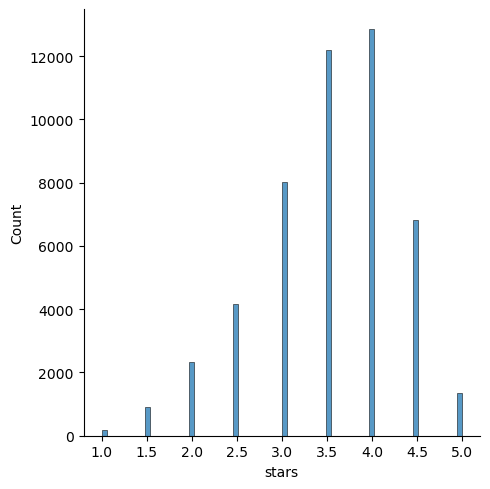
|  | mean\_rating | median\_rating | std\_rating | min\_rating | max\_rating |
| --- | --- | --- | --- | --- | --- |
| 0 | 3.527859 | 3.5 | 0.773849 | 1.0 | 5.0 |

df\_temp = con.sql("""  
 SELECT stars FROM restos  
""").df()  
df\_temp

|  | stars |
| --- | --- |
| 0 | 4.0 |
| 1 | 4.0 |
| 2 | 3.5 |
| 3 | 4.5 |
| 4 | 4.0 |
| ... | ... |
| 48795 | 2.0 |
| 48796 | 3.0 |
| 48797 | 3.0 |
| 48798 | 4.0 |
| 48799 | 4.5 |

sns.displot(data = df\_temp, x = "stars", kind = "hist")

F:\anaconda3\envs\sandbox\Lib\site-packages\seaborn\\_oldcore.py:1119: FutureWarning: use\_inf\_as\_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.  
 with pd.option\_context('mode.use\_inf\_as\_na', True):



Average stars on yelp received by restaurants grouped by state.

df\_temp = con.sql("""  
 SELECT state, avg(stars) as avg\_stars FROM restos   
 GROUP BY state  
 ORDER BY avg\_stars DESC  
""").df()  
df\_temp

|  | state | avg\_stars |
| --- | --- | --- |
| 0 | ABE | 4.500000 |
| 1 | NH | 4.000000 |
| 2 | OR | 3.752074 |
| 3 | TX | 3.656262 |
| 4 | CO | 3.587216 |
| 5 | FL | 3.521030 |
| 6 | MN | 3.500000 |
| 7 | MA | 3.480819 |
| 8 | BC | 3.454463 |
| 9 | WA | 3.450202 |
| 10 | OH | 3.421853 |
| 11 | GA | 3.399796 |
| 12 | VA | 2.000000 |
| 13 | KS | 2.000000 |
| 14 | KY | 2.000000 |
| 15 | WY | 1.500000 |

Let us look at how many restaurants provide WiFi.

con.sql("""  
 SELECT DISTINCT "attributes.WiFi" FROM restos  
""").df()

|  | attributes.WiFi |
| --- | --- |
| 0 | None |
| 1 | u'no' |
| 2 | u'paid' |
| 3 | 'no' |
| 4 | 'paid' |
| 5 | None |
| 6 | u'free' |
| 7 | 'free' |

df\_temp = con.sql("""  
 SELECT "attributes.WiFi" AS wifi, count("attributes.WiFi") AS count FROM restos  
 GROUP BY wifi  
""").df()  
df\_temp

|  | wifi | count |
| --- | --- | --- |
| 0 | None | 0 |
| 1 | u'free' | 14994 |
| 2 | 'no' | 4394 |
| 3 | 'free' | 5496 |
| 4 | u'paid' | 184 |
| 5 | u'no' | 11706 |
| 6 | 'paid' | 81 |
| 7 | None | 29 |

Of course we will need to correct the data because, as you can see, the values of the variable are not clean. There seems to be repetitions of the same thing mentioned in two different ways.

df\_new = pd.DataFrame({  
 "wifi": [  
 "None",  
 "Free",  
 "No",  
 "Paid"  
 ],  
 "count": [  
 (df\_temp["count"][0] + df\_temp["count"][7]),  
 (df\_temp["count"][1] + df\_temp["count"][3]),  
 (df\_temp["count"][2] + df\_temp["count"][5]),  
 (df\_temp["count"][4] + df\_temp["count"][6])  
 ]  
})  
df\_new

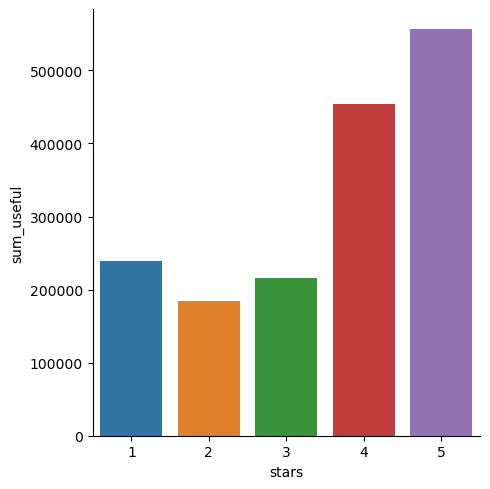
|  | wifi | count |
| --- | --- | --- |
| 0 | None | 29 |
| 1 | Free | 20490 |
| 2 | No | 16100 |
| 3 | Paid | 265 |

Let us find aggregates of useful votes recevied by reviews grouped by star ratings.

df\_temp = con.sql("""  
 SELECT stars, sum(useful) as sum\_useful, avg(useful) as avg\_useful, max(useful) AS max\_useful FROM resto\_reviews  
 GROUP BY stars  
 ORDER BY stars ASC  
""").df()  
df\_temp

|  | stars | sum\_useful | avg\_useful | max\_useful |
| --- | --- | --- | --- | --- |
| 0 | 1 | 239163.0 | 1.256445 | 146 |
| 1 | 2 | 184718.0 | 1.192214 | 260 |
| 2 | 3 | 216230.0 | 1.000694 | 177 |
| 3 | 4 | 454666.0 | 1.036895 | 411 |
| 4 | 5 | 556257.0 | 0.825011 | 223 |

sns.catplot(data = df\_temp, x = "stars", y = "sum\_useful", kind = "bar")



# 3. Data Preparation

df\_resto = con.sql("""  
 SELECT \* FROM restos   
 LIMIT 50;  
""").df()  
df\_resto

|  | name | address | city | state | postal\_code | latitude | longitude | stars | review\_count | is\_open | ... | attributes.Smoking | attributes.DriveThru | attributes.BYOBCorkage | attributes.Corkage | attributes.RestaurantsCounterService | attributes.DietaryRestrictions | attributes.AgesAllowed | attributes.Open24Hours | attributes.AcceptsInsurance | attributes.HairSpecializesIn |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | Oskar Blues Taproom | 921 Pearl St | Boulder | CO | 80302 | 40.017544 | -105.283348 | 4.0 | 86 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 1 | Flying Elephants at PDX | 7000 NE Airport Way | Portland | OR | 97218 | 45.588906 | -122.593331 | 4.0 | 126 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 2 | Bob Likes Thai Food | 3755 Main St | Vancouver | BC | V5V | 49.251342 | -123.101333 | 3.5 | 169 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 3 | Boxwood Biscuit | 740 S High St | Columbus | OH | 43206 | 39.947007 | -82.997471 | 4.5 | 11 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 4 | Mr G's Pizza & Subs | 474 Lowell St | Peabody | MA | 01960 | 42.541155 | -70.973438 | 4.0 | 39 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 5 | Sister Honey's | 247 E Michigan St | Orlando | FL | 32806 | 28.513265 | -81.374707 | 4.5 | 135 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 6 | Everything POP Shopping & Dining | 1050 Century Dr | Orlando | FL | 32830 | 28.350498 | -81.542819 | 3.0 | 7 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 7 | RaceTrac | 350 W Sand Lake Rd | Pine Castle | FL | 32809 | 28.450302 | -81.380587 | 3.5 | 5 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 8 | Cascade Restaurant | Hyatt Regency Grand Cypress, 1 Grand Cypress Blvd | Orlando | FL | 32836 | 28.381945 | -81.510327 | 3.5 | 18 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 9 | Longwood Galleria | 340-350 Longwood Ave | Boston | MA | 02215 | 42.338544 | -71.106842 | 2.5 | 24 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 10 | Legal Sea Foods | 1 Harborside Dr | Boston | MA | 02128 | 42.363442 | -71.025781 | 3.5 | 856 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 11 | ARGO | 3790 Canada Way, Unit 108 | Burnaby | BC | V5G 1G4 | 49.254436 | -123.020050 | 4.0 | 76 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 12 | Espresso Minute | 334 Mass Ave | Boston | MA | 02115 | 42.342673 | -71.084239 | 4.5 | 7 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 13 | Cleary's Restaurant & Spirits | 12429 NE Glisan St | Portland | OR | 97230 | 45.526473 | -122.535323 | 3.5 | 19 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 14 | El Pollo Rey | 1725 E Riverside Dr | Austin | TX | 78741 | 30.243493 | -97.730141 | 5.0 | 11 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 15 | Eurasia Sushi Bar & Seafood | 7101 W Hwy 71, Ste C-13 | Austin | TX | 78735 | 30.234533 | -97.877262 | 4.5 | 395 | 1 | ... | u'no' | None | None | None | NaN | None | None | NaN | NaN | None |
| 16 | Texas Roadhouse | 2457 S Highway 27 | Clermont | FL | 34711 | 28.538452 | -81.737625 | 3.5 | 102 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 17 | Kerala Kitchen | 103- 9386 120 Street | Surrey | BC | V3V 4B9 | 49.173723 | -122.889641 | 4.0 | 9 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 18 | Cafe Yumm! | 301 SW Morrison St | Portland | OR | 97204 | 45.518450 | -122.675723 | 3.5 | 158 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 19 | Magical Pizza Express | 3262 Vineland Rd, Ste 106 | Orlando | FL | 32811 | 28.520500 | -81.443950 | 1.0 | 8 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 20 | Taco Bell | 3255 28th Street | Boulder | CO | 80301 | 40.033678 | -105.259103 | 1.5 | 44 | 1 | ... | None | True | None | None | NaN | None | None | NaN | NaN | None |
| 21 | Panda Express | 1460 Rinehart Rd | Sanford | FL | 32771 | 28.800686 | -81.331712 | 2.5 | 47 | 1 | ... | None | True | None | None | NaN | None | None | NaN | NaN | None |
| 22 | Gigi's Roast Beef & Pizza | 5 Center St | Burlington | MA | 01803 | 42.506935 | -71.195854 | 3.0 | 16 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 23 | Wendy's | 6247 Mcneil Drive | Austin | TX | 78729 | 30.441875 | -97.746581 | 2.0 | 46 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 24 | Great Wraps | 3393 Peachtree Rd NE | Atlanta | GA | 30326 | 33.846335 | -84.363578 | 3.0 | 11 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 25 | Chuck E. Cheese | 7456 W Colonial Dr | Orlando | FL | 32818 | 28.551335 | -81.483167 | 2.0 | 15 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 26 | Olmecas Mexican Restaurant | 2121 E Oltorf St | Austin | TX | 78741 | 30.230016 | -97.732639 | 3.0 | 21 | 0 | ... | None | None | 'no' | None | NaN | None | None | NaN | NaN | None |
| 27 | JP Licks | 611 Assembly Row | Somerville | MA | 02145 | 42.395406 | -71.079491 | 3.5 | 131 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 28 | Los Potrillos Mexican Restaurant & Bar | 10722 NW Lakeshore Ave | Vancouver | WA | 98685 | 45.698889 | -122.703543 | 3.5 | 63 | 1 | ... | None | None | None | True | NaN | None | None | NaN | NaN | None |
| 29 | Yan's China Bistro | 146 Humphrey St | Swampscott | MA | 01907 | 42.468081 | -70.916752 | 4.0 | 74 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 30 | Piece of Cake | 3215 Roswell Rd NE | Atlanta | GA | 30305 | 33.842875 | -84.378539 | 4.0 | 231 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 31 | Andy's Frozen Custard | 5381 International Dr | Orlando | FL | 32819 | 28.463278 | -81.451181 | 4.5 | 36 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 32 | Sabor Latino - Puerto Rican Cuisine & Gallery | 777 Deltona Blvd | Deltona | FL | 32725 | 28.882545 | -81.279028 | 3.5 | 26 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 33 | Whole Time Chicken | None | Portland | OR | 97215 | 45.516241 | -122.597527 | 5.0 | 7 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 34 | B Street Coffee House | 2190 W Burnside St | Portland | OR | 97210 | 45.523074 | -122.696206 | 4.5 | 131 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 35 | Viuda Bistro | 108 N Main St | Buda | TX | 78610 | 30.081542 | -97.844353 | 4.5 | 21 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 36 | Urban Pantry | 1128 NW Lovejoy St | Portland | OR | 97209 | 45.529939 | -122.682932 | 3.5 | 26 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 37 | First China Restaurant | 5295 Buford Hwy NE | Atlanta | GA | 30340 | 33.896640 | -84.280918 | 3.0 | 11 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 38 | Taqueria So Mexican | SW 1st And Columbia | Portland | OR | 97230 | 45.513199 | -122.676381 | 4.0 | 28 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 39 | Pho Oanh | 2121 E Oltorf St, Ste 12B | Austin | TX | 78741 | 30.229913 | -97.732649 | 3.5 | 15 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 40 | Sapporo Ramen | 1815 Massachusetts Ave | Cambridge | MA | 02140 | 42.387212 | -71.118532 | 3.5 | 635 | 1 | ... | None | None | 'no' | None | NaN | None | None | NaN | NaN | None |
| 41 | Long John Silvers | 720 Old San Antonio Rd | Buda | TX | 78610 | 30.087588 | -97.822439 | 1.0 | 7 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 42 | Something Fishy | 2107 A E Semoran Blvd | Apopka | FL | 32703 | 28.673554 | -81.471320 | 4.5 | 242 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 43 | VietRiver | 1817 S Washington Ave | Titusville | FL | 32780 | 28.592881 | -80.803729 | 4.0 | 105 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 44 | Fukami Sushiya | 4246 SE Belmont | Portland | OR | 97215 | 45.516382 | -122.618593 | 4.0 | 17 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 45 | Let's Roll Custom Sushi Bar | 1184 Denman Street, Suite 104 | Vancouver | BC | V6G 2M9 | 49.286360 | -123.140261 | 4.5 | 31 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 46 | Hunan Lion | 2038 Crown Plaza | Columbus | OH | 43220 | 40.065378 | -83.071415 | 4.0 | 162 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 47 | Teriyaki House | 868 Broadway | Somerville | MA | 02144 | 42.400746 | -71.117803 | 4.0 | 52 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 48 | The Primavera Restaurant | 289 Walk Hill St | Roslindale | MA | 02131 | 42.287845 | -71.107567 | 3.5 | 19 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 49 | Deja Vu Creperie | 480 N Orlando Ave, Ste 126 | Winter Park | FL | 32789 | 28.602128 | -81.363952 | 3.5 | 19 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |

df\_resto.drop\_duplicates()

|  | name | address | city | state | postal\_code | latitude | longitude | stars | review\_count | is\_open | ... | attributes.Smoking | attributes.DriveThru | attributes.BYOBCorkage | attributes.Corkage | attributes.RestaurantsCounterService | attributes.DietaryRestrictions | attributes.AgesAllowed | attributes.Open24Hours | attributes.AcceptsInsurance | attributes.HairSpecializesIn |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | Oskar Blues Taproom | 921 Pearl St | Boulder | CO | 80302 | 40.017544 | -105.283348 | 4.0 | 86 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 1 | Flying Elephants at PDX | 7000 NE Airport Way | Portland | OR | 97218 | 45.588906 | -122.593331 | 4.0 | 126 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 2 | Bob Likes Thai Food | 3755 Main St | Vancouver | BC | V5V | 49.251342 | -123.101333 | 3.5 | 169 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 3 | Boxwood Biscuit | 740 S High St | Columbus | OH | 43206 | 39.947007 | -82.997471 | 4.5 | 11 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 4 | Mr G's Pizza & Subs | 474 Lowell St | Peabody | MA | 01960 | 42.541155 | -70.973438 | 4.0 | 39 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 5 | Sister Honey's | 247 E Michigan St | Orlando | FL | 32806 | 28.513265 | -81.374707 | 4.5 | 135 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 6 | Everything POP Shopping & Dining | 1050 Century Dr | Orlando | FL | 32830 | 28.350498 | -81.542819 | 3.0 | 7 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 7 | RaceTrac | 350 W Sand Lake Rd | Pine Castle | FL | 32809 | 28.450302 | -81.380587 | 3.5 | 5 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 8 | Cascade Restaurant | Hyatt Regency Grand Cypress, 1 Grand Cypress Blvd | Orlando | FL | 32836 | 28.381945 | -81.510327 | 3.5 | 18 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 9 | Longwood Galleria | 340-350 Longwood Ave | Boston | MA | 02215 | 42.338544 | -71.106842 | 2.5 | 24 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 10 | Legal Sea Foods | 1 Harborside Dr | Boston | MA | 02128 | 42.363442 | -71.025781 | 3.5 | 856 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 11 | ARGO | 3790 Canada Way, Unit 108 | Burnaby | BC | V5G 1G4 | 49.254436 | -123.020050 | 4.0 | 76 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 12 | Espresso Minute | 334 Mass Ave | Boston | MA | 02115 | 42.342673 | -71.084239 | 4.5 | 7 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 13 | Cleary's Restaurant & Spirits | 12429 NE Glisan St | Portland | OR | 97230 | 45.526473 | -122.535323 | 3.5 | 19 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 14 | El Pollo Rey | 1725 E Riverside Dr | Austin | TX | 78741 | 30.243493 | -97.730141 | 5.0 | 11 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 15 | Eurasia Sushi Bar & Seafood | 7101 W Hwy 71, Ste C-13 | Austin | TX | 78735 | 30.234533 | -97.877262 | 4.5 | 395 | 1 | ... | u'no' | None | None | None | NaN | None | None | NaN | NaN | None |
| 16 | Texas Roadhouse | 2457 S Highway 27 | Clermont | FL | 34711 | 28.538452 | -81.737625 | 3.5 | 102 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 17 | Kerala Kitchen | 103- 9386 120 Street | Surrey | BC | V3V 4B9 | 49.173723 | -122.889641 | 4.0 | 9 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 18 | Cafe Yumm! | 301 SW Morrison St | Portland | OR | 97204 | 45.518450 | -122.675723 | 3.5 | 158 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 19 | Magical Pizza Express | 3262 Vineland Rd, Ste 106 | Orlando | FL | 32811 | 28.520500 | -81.443950 | 1.0 | 8 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 20 | Taco Bell | 3255 28th Street | Boulder | CO | 80301 | 40.033678 | -105.259103 | 1.5 | 44 | 1 | ... | None | True | None | None | NaN | None | None | NaN | NaN | None |
| 21 | Panda Express | 1460 Rinehart Rd | Sanford | FL | 32771 | 28.800686 | -81.331712 | 2.5 | 47 | 1 | ... | None | True | None | None | NaN | None | None | NaN | NaN | None |
| 22 | Gigi's Roast Beef & Pizza | 5 Center St | Burlington | MA | 01803 | 42.506935 | -71.195854 | 3.0 | 16 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 23 | Wendy's | 6247 Mcneil Drive | Austin | TX | 78729 | 30.441875 | -97.746581 | 2.0 | 46 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 24 | Great Wraps | 3393 Peachtree Rd NE | Atlanta | GA | 30326 | 33.846335 | -84.363578 | 3.0 | 11 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 25 | Chuck E. Cheese | 7456 W Colonial Dr | Orlando | FL | 32818 | 28.551335 | -81.483167 | 2.0 | 15 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 26 | Olmecas Mexican Restaurant | 2121 E Oltorf St | Austin | TX | 78741 | 30.230016 | -97.732639 | 3.0 | 21 | 0 | ... | None | None | 'no' | None | NaN | None | None | NaN | NaN | None |
| 27 | JP Licks | 611 Assembly Row | Somerville | MA | 02145 | 42.395406 | -71.079491 | 3.5 | 131 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 28 | Los Potrillos Mexican Restaurant & Bar | 10722 NW Lakeshore Ave | Vancouver | WA | 98685 | 45.698889 | -122.703543 | 3.5 | 63 | 1 | ... | None | None | None | True | NaN | None | None | NaN | NaN | None |
| 29 | Yan's China Bistro | 146 Humphrey St | Swampscott | MA | 01907 | 42.468081 | -70.916752 | 4.0 | 74 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 30 | Piece of Cake | 3215 Roswell Rd NE | Atlanta | GA | 30305 | 33.842875 | -84.378539 | 4.0 | 231 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 31 | Andy's Frozen Custard | 5381 International Dr | Orlando | FL | 32819 | 28.463278 | -81.451181 | 4.5 | 36 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 32 | Sabor Latino - Puerto Rican Cuisine & Gallery | 777 Deltona Blvd | Deltona | FL | 32725 | 28.882545 | -81.279028 | 3.5 | 26 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 33 | Whole Time Chicken | None | Portland | OR | 97215 | 45.516241 | -122.597527 | 5.0 | 7 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 34 | B Street Coffee House | 2190 W Burnside St | Portland | OR | 97210 | 45.523074 | -122.696206 | 4.5 | 131 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 35 | Viuda Bistro | 108 N Main St | Buda | TX | 78610 | 30.081542 | -97.844353 | 4.5 | 21 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 36 | Urban Pantry | 1128 NW Lovejoy St | Portland | OR | 97209 | 45.529939 | -122.682932 | 3.5 | 26 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 37 | First China Restaurant | 5295 Buford Hwy NE | Atlanta | GA | 30340 | 33.896640 | -84.280918 | 3.0 | 11 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 38 | Taqueria So Mexican | SW 1st And Columbia | Portland | OR | 97230 | 45.513199 | -122.676381 | 4.0 | 28 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 39 | Pho Oanh | 2121 E Oltorf St, Ste 12B | Austin | TX | 78741 | 30.229913 | -97.732649 | 3.5 | 15 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 40 | Sapporo Ramen | 1815 Massachusetts Ave | Cambridge | MA | 02140 | 42.387212 | -71.118532 | 3.5 | 635 | 1 | ... | None | None | 'no' | None | NaN | None | None | NaN | NaN | None |
| 41 | Long John Silvers | 720 Old San Antonio Rd | Buda | TX | 78610 | 30.087588 | -97.822439 | 1.0 | 7 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 42 | Something Fishy | 2107 A E Semoran Blvd | Apopka | FL | 32703 | 28.673554 | -81.471320 | 4.5 | 242 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 43 | VietRiver | 1817 S Washington Ave | Titusville | FL | 32780 | 28.592881 | -80.803729 | 4.0 | 105 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 44 | Fukami Sushiya | 4246 SE Belmont | Portland | OR | 97215 | 45.516382 | -122.618593 | 4.0 | 17 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 45 | Let's Roll Custom Sushi Bar | 1184 Denman Street, Suite 104 | Vancouver | BC | V6G 2M9 | 49.286360 | -123.140261 | 4.5 | 31 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 46 | Hunan Lion | 2038 Crown Plaza | Columbus | OH | 43220 | 40.065378 | -83.071415 | 4.0 | 162 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 47 | Teriyaki House | 868 Broadway | Somerville | MA | 02144 | 42.400746 | -71.117803 | 4.0 | 52 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 48 | The Primavera Restaurant | 289 Walk Hill St | Roslindale | MA | 02131 | 42.287845 | -71.107567 | 3.5 | 19 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 49 | Deja Vu Creperie | 480 N Orlando Ave, Ste 126 | Winter Park | FL | 32789 | 28.602128 | -81.363952 | 3.5 | 19 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |

df\_resto.describe()

|  | latitude | longitude | stars | review\_count | is\_open | int\_business\_id |
| --- | --- | --- | --- | --- | --- | --- |
| count | 50.000000 | 50.000000 | 50.000000 | 50.000000 | 50.000000 | 50.000000 |
| mean | 37.508985 | -93.579034 | 3.570000 | 91.000000 | 0.640000 | 68.740000 |
| std | 7.625686 | 19.662369 | 0.909193 | 156.602187 | 0.484873 | 41.658872 |
| min | 28.350498 | -123.140261 | 1.000000 | 5.000000 | 0.000000 | 1.000000 |
| 25% | 29.182294 | -118.222329 | 3.500000 | 15.250000 | 0.000000 | 36.250000 |
| 50% | 40.025611 | -83.676167 | 3.500000 | 29.500000 | 1.000000 | 63.500000 |
| 75% | 44.770188 | -81.339772 | 4.000000 | 104.250000 | 1.000000 | 105.750000 |
| max | 49.286360 | -70.916752 | 5.000000 | 856.000000 | 1.000000 | 137.000000 |

features = df\_resto[[ 'review\_count']]   
target = df\_resto['stars']  
  
X\_train, X\_test, y\_train, y\_test = train\_test\_split(features, target, test\_size=0.2, random\_state=42)  
model = RandomForestRegressor(n\_estimators=100, random\_state=42)  
  
model.fit(X\_train, y\_train)  
y\_pred = model.predict(X\_test)  
mse = mean\_squared\_error(y\_test, y\_pred)  
r2 = r2\_score(y\_test, y\_pred)  
print(f"Mean Squared Error: {mse}")  
print(f"R^2 Score: {r2}")  
  
importance = model.feature\_importances\_  
feature\_importance\_df = pd.DataFrame({'Feature': features.columns, 'Importance': importance})  
print(feature\_importance\_df.sort\_values(by='Importance', ascending=False))

Mean Squared Error: 0.9690963900513131  
R^2 Score: -0.006853392261104485  
 Feature Importance  
0 review\_count 1.0

correlation\_review\_rating = spst.spearmanr(df\_resto['review\_count'], df\_resto['stars'])  
print(f"correlation between review count and rating: r={correlation\_review\_rating.correlation}, p={correlation\_review\_rating.pvalue}")

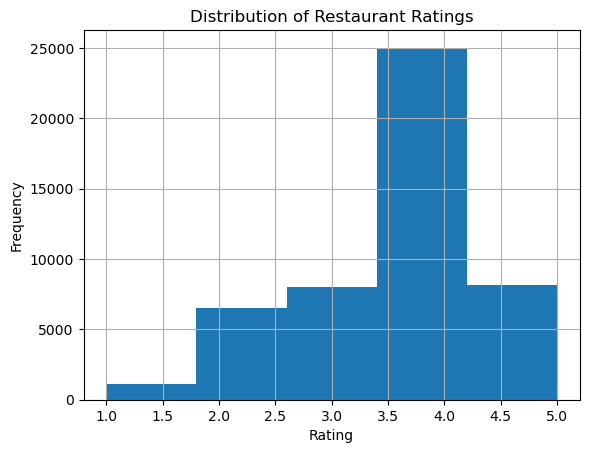
correlation between review count and rating: r=0.21767275128966765, p=0.12888288086635213

con.sql("""  
 SELECT \* FROM resto\_reviews  
 LIMIT 20;  
""")

┌───────┬────────┬───────┬───────┬───┬─────────────────────┬─────────────────┬─────────────┬────────────────────┐  
│ stars │ useful │ funny │ cool │ … │ date │ int\_business\_id │ int\_user\_id │ int\_rest\_review\_id │  
│ int64 │ int64 │ int64 │ int64 │ │ timestamp │ int64 │ int64 │ int64 │  
├───────┼────────┼───────┼───────┼───┼─────────────────────┼─────────────────┼─────────────┼────────────────────┤  
│ 2 │ 1 │ 1 │ 1 │ … │ 2010-01-08 02:29:15 │ 4954 │ 6319642 │ 2 │  
│ 2 │ 0 │ 0 │ 0 │ … │ 2006-04-16 02:58:44 │ 14180 │ 292901 │ 5 │  
│ 5 │ 0 │ 0 │ 0 │ … │ 2014-05-07 18:10:21 │ 11779 │ 6336225 │ 7 │  
│ 5 │ 5 │ 3 │ 3 │ … │ 2014-02-05 21:09:05 │ 3216 │ 552519 │ 12 │  
│ 4 │ 0 │ 0 │ 0 │ … │ 2017-05-26 03:05:46 │ 8748 │ 544027 │ 16 │  
│ 5 │ 0 │ 0 │ 0 │ … │ 2016-01-30 02:39:22 │ 2498 │ 6222487 │ 17 │  
│ 1 │ 0 │ 1 │ 0 │ … │ 2014-07-13 18:59:44 │ 2933 │ 75453 │ 21 │  
│ 1 │ 1 │ 0 │ 0 │ … │ 2015-12-20 19:31:34 │ 9068 │ 3448079 │ 22 │  
│ 5 │ 1 │ 1 │ 0 │ … │ 2011-06-29 01:26:18 │ 9607 │ 28981 │ 26 │  
│ 5 │ 0 │ 0 │ 0 │ … │ 2016-09-18 17:06:01 │ 6290 │ 1793741 │ 31 │  
│ 3 │ 0 │ 0 │ 0 │ … │ 2018-09-25 03:22:50 │ 5210 │ 6345751 │ 33 │  
│ 1 │ 0 │ 0 │ 0 │ … │ 2017-11-06 19:27:53 │ 14459 │ 6360015 │ 36 │  
│ 5 │ 0 │ 0 │ 0 │ … │ 2013-10-24 02:48:36 │ 15026 │ 3629711 │ 37 │  
│ 5 │ 0 │ 0 │ 0 │ … │ 2018-05-08 17:14:19 │ 435 │ 6299381 │ 40 │  
│ 1 │ 0 │ 0 │ 0 │ … │ 2018-04-15 08:49:03 │ 3557 │ 6378417 │ 43 │  
│ 5 │ 0 │ 0 │ 0 │ … │ 2008-09-29 09:33:04 │ 1800 │ 6336931 │ 44 │  
│ 4 │ 1 │ 0 │ 0 │ … │ 2005-03-14 19:50:32 │ 13432 │ 65933 │ 51 │  
│ 4 │ 1 │ 0 │ 0 │ … │ 2015-01-03 22:55:27 │ 15225 │ 83929 │ 52 │  
│ 5 │ 3 │ 1 │ 2 │ … │ 2016-10-26 15:45:52 │ 11621 │ 35210 │ 61 │  
│ 1 │ 2 │ 0 │ 0 │ … │ 2014-11-12 12:18:00 │ 8500 │ 40687 │ 64 │  
├───────┴────────┴───────┴───────┴───┴─────────────────────┴─────────────────┴─────────────┴────────────────────┤  
│ 20 rows 9 columns (8 shown) │  
└───────────────────────────────────────────────────────────────────────────────────────────────────────────────┘

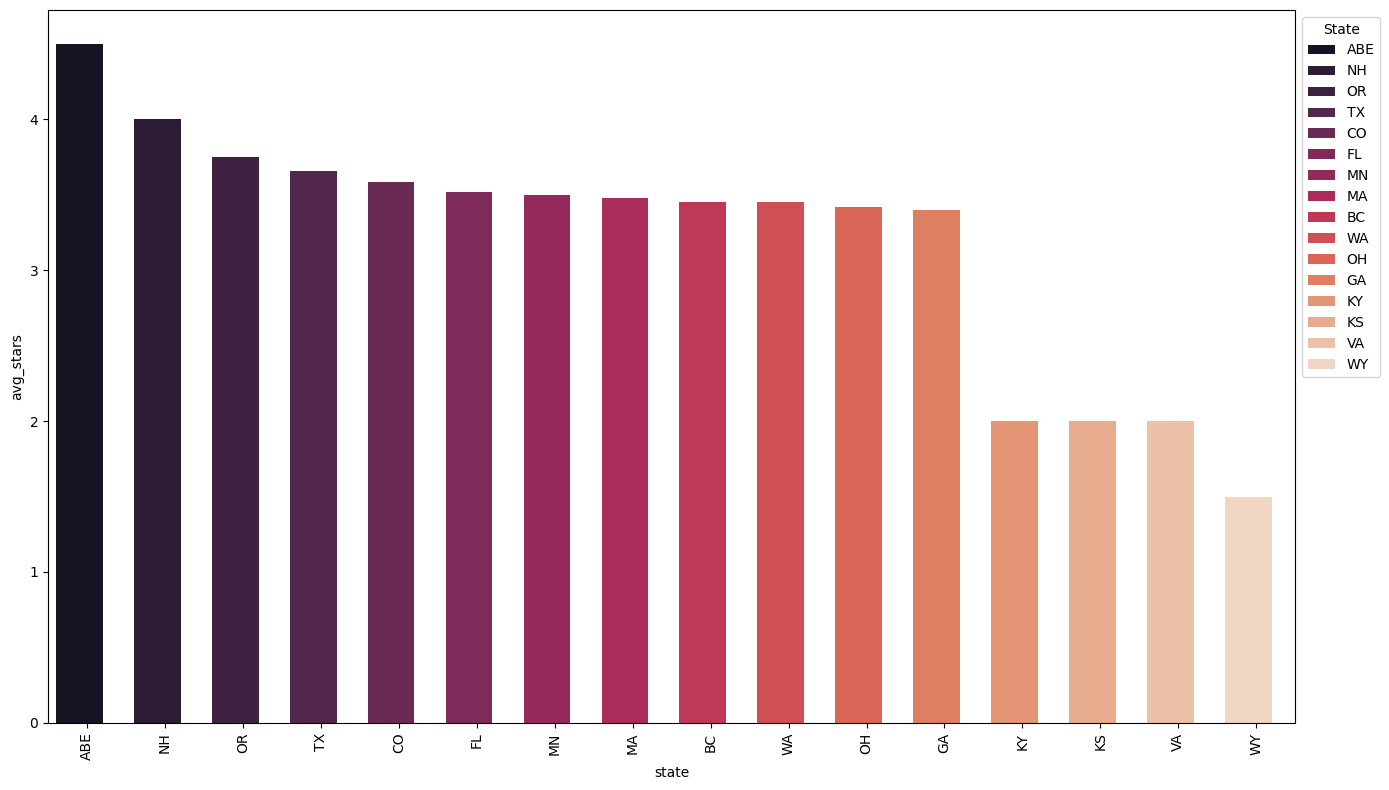
## 3.1 Exploratory Data Analysis

con.sql('''  
 SELECT stars   
 FROM restos;  
''').df().hist(bins=5)  
plt.title('Distribution of Restaurant Ratings')  
plt.xlabel('Rating')  
plt.ylabel('Frequency')  
plt.show()



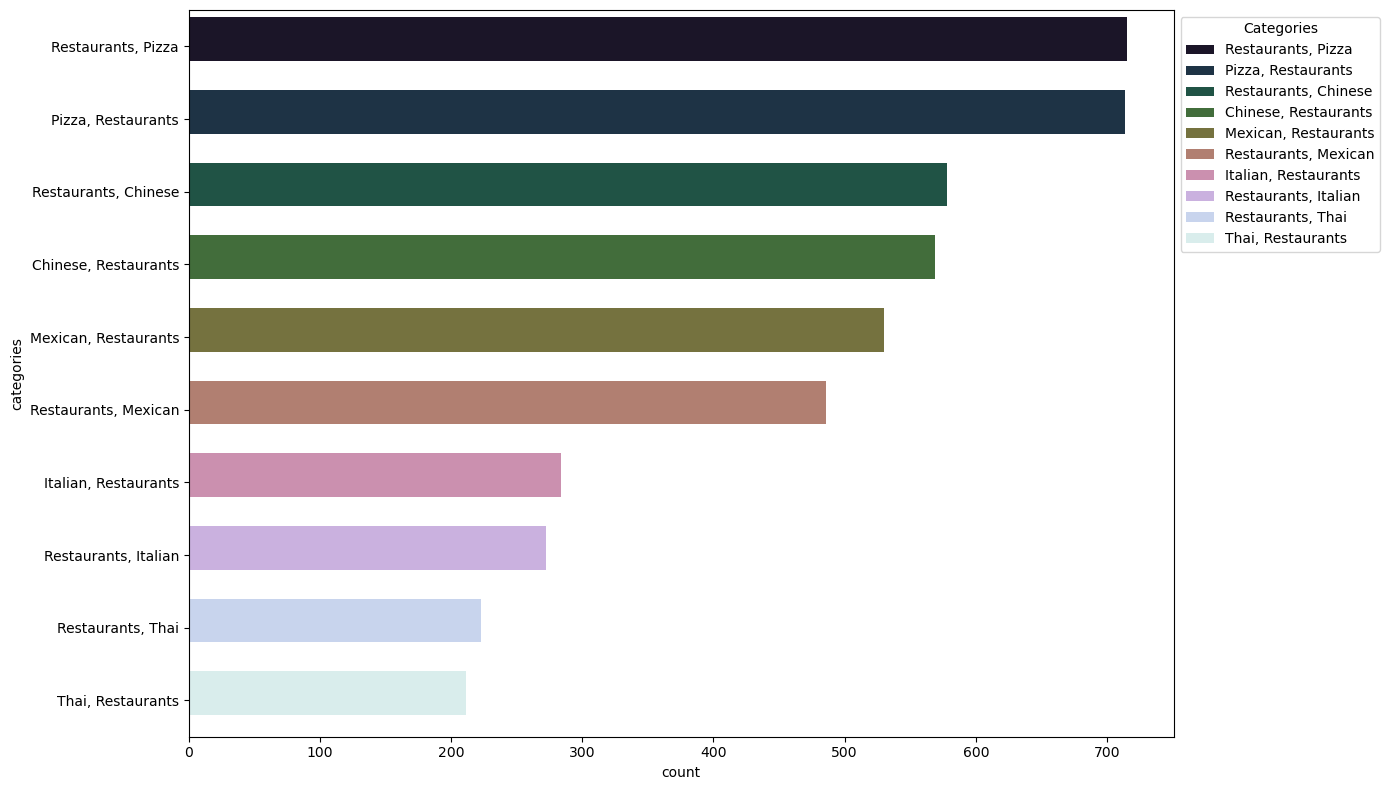
#### 3.1.0.1 Location Analysis

avg\_stars\_state = con.sql("""  
 SELECT state, avg(stars) as avg\_stars  
 FROM restos  
 GROUP BY state  
 ORDER BY avg\_stars DESC  
""").df()  
  
plt.figure(figsize=(14, 8))   
ax = sns.barplot(data=df\_avg\_stars\_state, x='state', y='avg\_stars', hue='state', palette='rocket', dodge=False)  
  
for patch in ax.patches:  
 patch.set\_width(0.6)   
plt.xticks(rotation=90)  
plt.legend(loc='upper left', bbox\_to\_anchor=(1, 1), title='State')  
plt.tight\_layout()  
plt.show()



#### 3.1.0.2 Restaurant Type

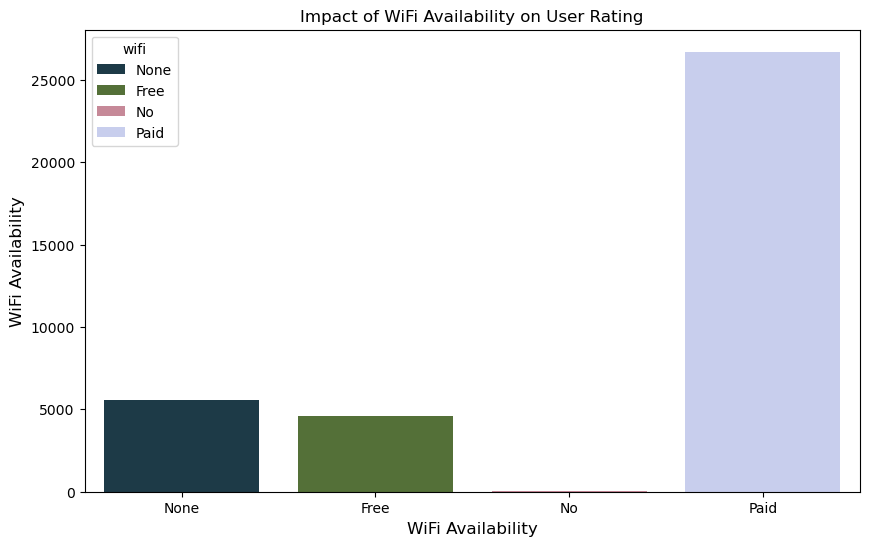
df\_categories = con.sql("""  
 SELECT categories, count(\*) as count  
 FROM restos  
 GROUP BY categories  
 ORDER BY count DESC  
 LIMIT 10  
""").df()  
  
plt.figure(figsize=(14, 8))   
ax = sns.barplot(data=df\_categories, x='count', y='categories', hue='categories', palette='cubehelix', dodge=False)  
  
for patch in ax.patches:  
 patch.set\_height(0.6)   
  
plt.legend(loc='upper left', bbox\_to\_anchor=(1, 1), title='Categories')  
plt.tight\_layout()  
plt.show()



#### 3.1.0.3 Inner join restos and resto\_reviews tables

df\_merged = con.sql("""  
 SELECT r.int\_business\_id, r.name, r.stars as resto\_avg\_stars,   
 rr.stars as review\_stars, r."attributes.WiFi", r."attributes.RestaurantsTakeOut"  
 FROM restos r  
 JOIN resto\_reviews rr ON r.int\_business\_id = rr.int\_business\_id  
""").df()  
  
# Analyze average ratings based on WiFi availability  
df\_wifi = df\_merged.groupby('attributes.WiFi').agg({  
 'review\_stars': 'mean'  
}).reset\_index()

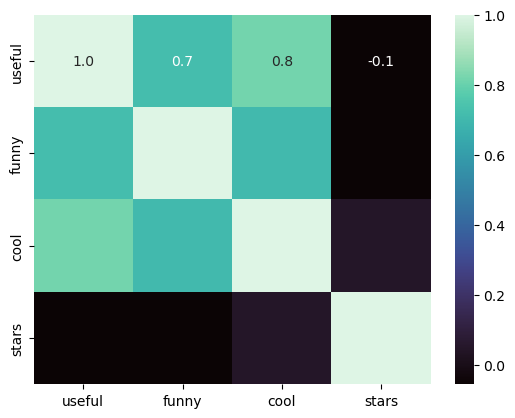
df\_temp = con.sql("""  
 SELECT "attributes.WiFi" AS wifi, count("attributes.WiFi") AS count FROM restos  
 GROUP BY wifi  
""").df()  
df\_temp  
df\_new = pd.DataFrame({  
 "wifi": [  
 "None",  
 "Free",  
 "No",  
 "Paid"  
 ],  
 "count": [  
 (df\_temp["count"][0] + df\_temp["count"][7]),  
 (df\_temp["count"][1] + df\_temp["count"][3]),  
 (df\_temp["count"][2] + df\_temp["count"][5]),  
 (df\_temp["count"][4] + df\_temp["count"][6])  
 ]  
})  
df\_new  
plt.figure(figsize = (10,6))  
sns.barplot(data = df\_new, x = "wifi", y = "count", hue = "wifi", dodge = False, palette = 'cubehelix', linewidth=0)  
plt.xlabel('WiFi Availability', fontsize=12)  
plt.ylabel('WiFi Availability', fontsize=12)  
plt.title('Impact of WiFi Availability on User Rating', fontsize=12)  
plt.show()



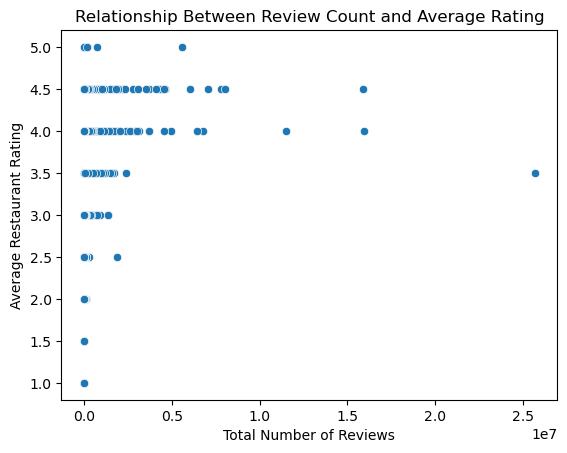
df\_amenities = con.sql("""  
 SELECT   
 "attributes.OutdoorSeating", "attributes.HappyHour",  
 "attributes.Alcohol", "attributes.BusinessAcceptsCreditCards",  
 avg(stars) as avg\_stars  
 FROM restos  
 GROUP BY   
 "attributes.OutdoorSeating", "attributes.HappyHour",  
 "attributes.Alcohol", "attributes.BusinessAcceptsCreditCards"  
 ORDER BY avg\_stars DESC  
""").df()  
df\_amenities.head(10)

|  | attributes.OutdoorSeating | attributes.HappyHour | attributes.Alcohol | attributes.BusinessAcceptsCreditCards | avg\_stars |
| --- | --- | --- | --- | --- | --- |
| 0 | True | False | None | False | 5.0 |
| 1 | True | False | u'none' | False | 4.7 |
| 2 | None | True | 'full\_bar' | None | 4.5 |
| 3 | None | True | 'none' | True | 4.5 |
| 4 | True | False | u'beer\_and\_wine' | False | 4.5 |
| 5 | None | False | u'beer\_and\_wine' | False | 4.5 |
| 6 | None | True | u'none' | True | 4.5 |
| 7 | None | None | u'none' | None | 4.5 |
| 8 | True | False | 'beer\_and\_wine' | False | 4.5 |
| 9 | True | True | u'none' | False | 4.5 |

df\_review\_chars = con.sql("""  
 SELECT useful, funny, cool, stars  
 FROM resto\_reviews  
""").df()  
  
sns.heatmap(df\_review\_chars.corr(), cmap = 'mako', annot = True, fmt = '.1f')

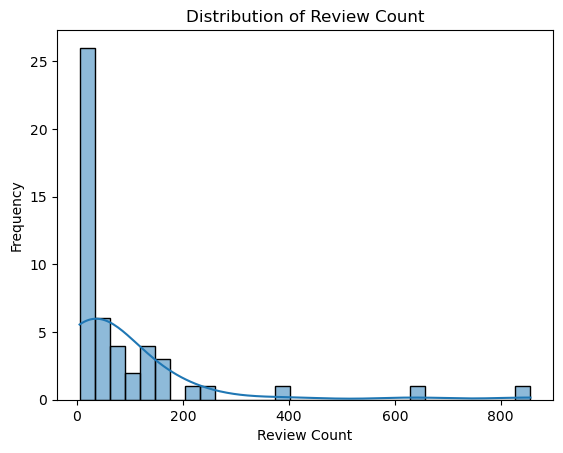


df\_merged = con.sql("""  
 SELECT r.int\_business\_id, r.name, r.stars as resto\_avg\_stars,   
 r.review\_count, rr.stars as review\_stars  
 FROM restos r  
 JOIN resto\_reviews rr ON r.int\_business\_id = rr.int\_business\_id  
""").df()  
  
df\_grouped = df\_merged.groupby('int\_business\_id').agg({  
 'resto\_avg\_stars': 'mean',  
 'review\_count': 'sum'  
}).reset\_index()  
  
sns.scatterplot(data=df\_grouped, x='review\_count', y='resto\_avg\_stars')  
plt.xlabel('Total Number of Reviews')  
plt.ylabel('Average Restaurant Rating')  
plt.title('Relationship Between Review Count and Average Rating')  
plt.show()



sns.histplot(df\_resto['review\_count'], bins=30, kde=True)  
plt.title('Distribution of Review Count')  
plt.xlabel('Review Count')  
plt.ylabel('Frequency')  
plt.show()

C:\Users\saswa\anaconda3\Lib\site-packages\seaborn\\_oldcore.py:1119: FutureWarning: use\_inf\_as\_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.  
 with pd.option\_context('mode.use\_inf\_as\_na', True):



df\_ratings\_state = con.sql("""  
 SELECT state, stars   
 FROM restos  
""").df()

sns.set(style="whitegrid")  
  
plt.figure(figsize=(12, 8))  
sns.boxplot(data=df\_ratings\_state, x='state', y='stars', palette='muted')  
plt.xlabel('State', fontsize=14, labelpad=10)  
plt.ylabel('Restaurant Ratings', fontsize=14, labelpad=10)  
plt.title('Restaurant Ratings by State', fontsize=16, weight='bold', pad=15)  
plt.xticks(rotation=90, fontsize=12)  
plt.yticks(fontsize=12)  
sns.despine()  
plt.grid(True, which='both', linestyle='--', linewidth=0.5)  
plt.show()



# 4. Hypothesis Testing

def check\_normality(data):  
 test\_stat\_normality, p\_value\_normality=spst.shapiro(data)  
 print("p value:%.4f" % p\_value\_normality)  
 if p\_value\_normality <0.05:  
 print("Reject null hypothesis >> The data is not normally distributed")  
 else:  
 print("Fail to reject null hypothesis >> The data is normally distributed")

def convert\_none\_to\_zero(data):  
 return 0 if pd.isna(data) else 1

## 4.1 Hypothesis Test 1 - Chi Square Test

### 4.1.1 Step 1 : Form Hypothesis

In this case, we are checking if the reviews are dependent on categories of the restaurants.

Ho ( Null Hypothesis ) : Review Count is independent of Category Count

Ha ( Alternate Hypothesis ) : Review Count is not independent of Category Count

### 4.1.2 Step 2 : Collect Data

df\_hypothesis1 = con.sql("""  
select review\_count, categories from restos;  
"""  
).df()  
df\_hypothesis1

|  | review\_count | categories |
| --- | --- | --- |
| 0 | 86 | Gastropubs, Food, Beer Gardens, Restaurants, B... |
| 1 | 126 | Salad, Soup, Sandwiches, Delis, Restaurants, C... |
| 2 | 169 | Restaurants, Thai |
| 3 | 11 | Breakfast & Brunch, Restaurants |
| 4 | 39 | Food, Pizza, Restaurants |
| ... | ... | ... |
| 48795 | 48 | American (Traditional), Desserts, American (Ne... |
| 48796 | 310 | Mexican, Bars, Nightlife, Tex-Mex, Restaurants |
| 48797 | 185 | Restaurants, Mexican, Latin American |
| 48798 | 10 | Cuban, Sandwiches, Restaurants, Cafes |
| 48799 | 82 | Restaurants, Middle Eastern, Mediterranean, Pe... |

### 4.1.3 Converting the categories to numerical values

templ = []  
for x in df\_hypothesis1['categories']:  
 a = len(x.split(','))  
 templ.append(a)  
df\_hypothesis1['Category Count'] = templ  
df\_hypothesis1

|  | review\_count | categories | Category Count |
| --- | --- | --- | --- |
| 0 | 86 | Gastropubs, Food, Beer Gardens, Restaurants, B... | 9 |
| 1 | 126 | Salad, Soup, Sandwiches, Delis, Restaurants, C... | 7 |
| 2 | 169 | Restaurants, Thai | 2 |
| 3 | 11 | Breakfast & Brunch, Restaurants | 2 |
| 4 | 39 | Food, Pizza, Restaurants | 3 |
| ... | ... | ... | ... |
| 48795 | 48 | American (Traditional), Desserts, American (Ne... | 8 |
| 48796 | 310 | Mexican, Bars, Nightlife, Tex-Mex, Restaurants | 5 |
| 48797 | 185 | Restaurants, Mexican, Latin American | 3 |
| 48798 | 10 | Cuban, Sandwiches, Restaurants, Cafes | 4 |
| 48799 | 82 | Restaurants, Middle Eastern, Mediterranean, Pe... | 5 |

contingency\_table = pd.crosstab(df\_hypothesis1['review\_count'], df\_hypothesis1['Category Count'])  
contingency\_table

| Category Count | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | ... | 17 | 18 | 19 | 20 | 21 | 23 | 26 | 27 | 28 | 37 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| review\_count |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 | 31 | 489 | 324 | 276 | 224 | 120 | 86 | 27 | 18 | 12 | ... | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 6 | 21 | 390 | 293 | 267 | 227 | 118 | 63 | 31 | 13 | 7 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | 18 | 375 | 284 | 241 | 179 | 97 | 69 | 42 | 19 | 7 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8 | 17 | 308 | 247 | 221 | 165 | 107 | 47 | 38 | 12 | 8 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 9 | 13 | 279 | 215 | 212 | 163 | 97 | 45 | 34 | 13 | 3 | ... | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 5115 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6221 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7219 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7298 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 9185 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

### 4.1.4 Step 3 : Compute Chi Square Test Statistic

stat, p, dof, expected = chi2\_contingency(df\_hypothesis1[['review\_count', 'Category Count']])

### 4.1.5 Step 4: Analyse the Chi Square-Stat and compare it with alpha

print("\nChi-square Test Results:")  
print("Chi-square value:", stat)  
print("P-value:", p)  
print("Degrees of freedom:", dof)  
print("Expected frequencies table:")  
expected\_frequencies = pd.DataFrame(expected)  
expected\_frequencies

Chi-square Test Results:  
Chi-square value: 566898.516679833  
P-value: 0.0  
Degrees of freedom: 48799  
Expected frequencies table:

|  | 0 | 1 |
| --- | --- | --- |
| 0 | 91.011223 | 3.988777 |
| 1 | 127.415712 | 5.584288 |
| 2 | 163.820202 | 7.179798 |
| 3 | 12.454167 | 0.545833 |
| 4 | 40.236541 | 1.763459 |
| ... | ... | ... |
| 48795 | 53.648721 | 2.351279 |
| 48796 | 301.774056 | 13.225944 |
| 48797 | 180.106421 | 7.893579 |
| 48798 | 13.412180 | 0.587820 |
| 48799 | 83.347120 | 3.652880 |

Since, the p-value is 0 which is lesser than alpha ( 0.05 ), we have enough statistical evidence to reject Null Hypothesis. This means that reviews is not independent of categories.

## 4.2 Hypothesis Test 2 - Multiple Linear Regression

### 4.2.1 Step 1 : Form Hypothesis

Objective : To understand how well Useful, Funny, and Cool predict the Stars rating. This helps assess the collective impact of these variables on Stars.

Independent variables : Useful, funny and cool Depenedent variable: stars

H0 ( Null Hypothesis ) : The coefficients of Useful, Funny, and Cool are equal to zero

Ha ( Alternate Hypothesis ) : At least one of the coefficients of useful, funny and cool is not qual to zero

### 4.2.2 Step 2 : Collect Data

df\_hypothesis2 = con.sql("""  
select \* from resto\_reviews  
LIMIT 100000;  
"""  
).df()  
df\_hypothesis2

|  | stars | useful | funny | cool | text | date | int\_business\_id | int\_user\_id | int\_rest\_review\_id |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 2 | 1 | 1 | 1 | I've stayed at many Marriott and Renaissance M... | 2010-01-08 02:29:15 | 4954 | 6319642 | 2 |
| 1 | 2 | 0 | 0 | 0 | The setting is perfectly adequate, and the foo... | 2006-04-16 02:58:44 | 14180 | 292901 | 5 |
| 2 | 5 | 0 | 0 | 0 | I work in the Pru and this is the most afforda... | 2014-05-07 18:10:21 | 11779 | 6336225 | 7 |
| 3 | 5 | 5 | 3 | 3 | I loved everything about this place. I've only... | 2014-02-05 21:09:05 | 3216 | 552519 | 12 |
| 4 | 4 | 0 | 0 | 0 | I think their rice dishes are way better than ... | 2017-05-26 03:05:46 | 8748 | 544027 | 16 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 99995 | 5 | 0 | 0 | 0 | Just wait in line! Don't get discouraged! Ther... | 2016-03-27 20:46:08 | 1851 | 402967 | 333977 |
| 99996 | 5 | 0 | 0 | 0 | Excellent service, food and overall atmosphere... | 2014-12-13 18:45:46 | 14551 | 9686596 | 333980 |
| 99997 | 5 | 0 | 0 | 0 | Whenever restaurants have house-made veggie bu... | 2017-09-30 13:43:07 | 1646 | 9876374 | 333981 |
| 99998 | 2 | 0 | 0 | 0 | Of course the view is amazing...and SO much be... | 2016-11-07 01:49:43 | 15925 | 9825763 | 333985 |
| 99999 | 5 | 0 | 0 | 1 | Edit: servers name is BATMAN aka David\n\nNot ... | 2017-09-24 20:05:17 | 13497 | 9916684 | 333992 |

x = df\_hypothesis2[['useful', 'funny', 'cool']]  
y = df\_hypothesis2['stars']

### 4.2.3 Step 3: Compute Test Statistic

x = sm.add\_constant(x)  
model = sm.OLS(y, x).fit()  
model.summary()

OLS Regression Results

|  |  |  |  |
| --- | --- | --- | --- |
| Dep. Variable: | stars | R-squared: | 0.053 |
| Model: | OLS | Adj. R-squared: | 0.053 |
| Method: | Least Squares | F-statistic: | 1858. |
| Date: | Sat, 31 Aug 2024 | Prob (F-statistic): | 0.00 |
| Time: | 17:24:06 | Log-Likelihood: | -1.6729e+05 |
| No. Observations: | 100000 | AIC: | 3.346e+05 |
| Df Residuals: | 99996 | BIC: | 3.346e+05 |
| Df Model: | 3 |  |  |
| Covariance Type: | nonrobust |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | coef | std err | t | P>|t| | [0.025 | 0.975] |
| const | 3.8170 | 0.004 | 849.751 | 0.000 | 3.808 | 3.826 |
| useful | -0.1779 | 0.003 | -51.716 | 0.000 | -0.185 | -0.171 |
| funny | -0.1816 | 0.005 | -37.561 | 0.000 | -0.191 | -0.172 |
| cool | 0.3829 | 0.006 | 68.444 | 0.000 | 0.372 | 0.394 |

|  |  |  |  |
| --- | --- | --- | --- |
| Omnibus: | 8624.380 | Durbin-Watson: | 1.998 |
| Prob(Omnibus): | 0.000 | Jarque-Bera (JB): | 10205.729 |
| Skew: | -0.758 | Prob(JB): | 0.00 |
| Kurtosis: | 2.615 | Cond. No. | 3.85 |

### 4.2.4 Step 4: Analyse the F-Stat and compare it with alpha

Alpha = 0.05

Since, p-value = 0, it means we have enough statistical evidence to reject Null Hypothesis. This implies that there is atleast one coefficient which is not equal to 0 which concludes that there is at least one independent variable significant enough that impact stars ( dependedent variable ). If we keep everything else constant, then if that variable is increased by 1 unit, there is an increase of that particular coefficient’s units to stars variable.

### 4.2.5 Interpretation of R-Square

R-square describes how well the variability in the data is explained by the regression model. So, 5.3% of the variability is explained by the regression model

### 4.2.6 95 % Confidence Interval

confidence\_intervals = model.conf\_int(alpha=0.05)  
confidence\_intervals

|  | 0 | 1 |
| --- | --- | --- |
| const | 3.808213 | 3.825821 |
| useful | -0.184689 | -0.171201 |
| funny | -0.191097 | -0.172143 |
| cool | 0.371892 | 0.393819 |

## 4.3 Hypothesis Test 3 - Simple Linear Regression

Independent Variable : Number of friends

Dependent Variable : Number of reviews

H0 ( Null Hypothesis ) : The coefficient for number of friends makes a significant impact on number of reviews is 0.

Ha ( Alternate Hypothesis ) : The coefficient for number of friends makes a signicant impact on number of reviews is not equal to 0.

### 4.3.1 Step 2 : Collect Data

joined\_table = con.sql("""  
SELECT u.review\_count as Num\_Reviews, f.num\_friends as Num\_Friends  
FROM users u  
LEFT JOIN user\_friends f  
ON u.int\_user\_id = f.int\_user\_id;  
""").df()  
joined\_table

|  | Num\_Reviews | Num\_Friends |
| --- | --- | --- |
| 0 | 187 | 2 |
| 1 | 13 | 3 |
| 2 | 90 | 1 |
| 3 | 2 | 1 |
| 4 | 3 | 1 |
| ... | ... | ... |
| 2189452 | 1 | 0 |
| 2189453 | 3 | 0 |
| 2189454 | 2 | 0 |
| 2189455 | 1 | 0 |
| 2189456 | 3 | 0 |

### 4.3.2 Step 3 : Compute Linear Regression ( Test Statistic )

x1 = joined\_table['Num\_Friends']  
y1 = joined\_table['Num\_Reviews']  
  
x1 = sm.add\_constant(x1)  
model1 = sm.OLS(y1, x1).fit()  
model1.summary()

OLS Regression Results

|  |  |  |  |
| --- | --- | --- | --- |
| Dep. Variable: | Num\_Reviews | R-squared: | 0.141 |
| Model: | OLS | Adj. R-squared: | 0.141 |
| Method: | Least Squares | F-statistic: | 3.597e+05 |
| Date: | Sat, 31 Aug 2024 | Prob (F-statistic): | 0.00 |
| Time: | 17:26:51 | Log-Likelihood: | -1.2423e+07 |
| No. Observations: | 2189457 | AIC: | 2.485e+07 |
| Df Residuals: | 2189455 | BIC: | 2.485e+07 |
| Df Model: | 1 |  |  |
| Covariance Type: | nonrobust |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | coef | std err | t | P>|t| | [0.025 | 0.975] |
| const | 11.2056 | 0.051 | 220.923 | 0.000 | 11.106 | 11.305 |
| Num\_Friends | 0.1996 | 0.000 | 599.714 | 0.000 | 0.199 | 0.200 |

|  |  |  |  |
| --- | --- | --- | --- |
| Omnibus: | 5603470.749 | Durbin-Watson: | 1.788 |
| Prob(Omnibus): | 0.000 | Jarque-Bera (JB): | 1039125953855.817 |
| Skew: | 28.010 | Prob(JB): | 0.00 |
| Kurtosis: | 3377.517 | Cond. No. | 162. |

### 4.3.3 Step 4: Analyse the F-Stat and compare it with alpha

Alpha = 0.05

Since, p-value << alpha, it means we have enough statistical evidence to reject Null Hypothesis. This implies that coefficient of number of friends makes a signicant impact on the number of reviews.

### 4.3.4 Interepretation of R Square

In this case, R square is coming approx 15%. This means that 15% of the variability in the data is explained by the Regression model. It describes how well the data is fitted into the model.

check2 = con.sql("""  
select \* from restos limit 10;  
"""   
).df()  
check2

|  | name | address | city | state | postal\_code | latitude | longitude | stars | review\_count | is\_open | ... | attributes.Smoking | attributes.DriveThru | attributes.BYOBCorkage | attributes.Corkage | attributes.RestaurantsCounterService | attributes.DietaryRestrictions | attributes.AgesAllowed | attributes.Open24Hours | attributes.AcceptsInsurance | attributes.HairSpecializesIn |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | Oskar Blues Taproom | 921 Pearl St | Boulder | CO | 80302 | 40.017544 | -105.283348 | 4.0 | 86 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 1 | Flying Elephants at PDX | 7000 NE Airport Way | Portland | OR | 97218 | 45.588906 | -122.593331 | 4.0 | 126 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 2 | Bob Likes Thai Food | 3755 Main St | Vancouver | BC | V5V | 49.251342 | -123.101333 | 3.5 | 169 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 3 | Boxwood Biscuit | 740 S High St | Columbus | OH | 43206 | 39.947007 | -82.997471 | 4.5 | 11 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 4 | Mr G's Pizza & Subs | 474 Lowell St | Peabody | MA | 01960 | 42.541155 | -70.973438 | 4.0 | 39 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 5 | Sister Honey's | 247 E Michigan St | Orlando | FL | 32806 | 28.513265 | -81.374707 | 4.5 | 135 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 6 | Everything POP Shopping & Dining | 1050 Century Dr | Orlando | FL | 32830 | 28.350498 | -81.542819 | 3.0 | 7 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 7 | RaceTrac | 350 W Sand Lake Rd | Pine Castle | FL | 32809 | 28.450302 | -81.380587 | 3.5 | 5 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 8 | Cascade Restaurant | Hyatt Regency Grand Cypress, 1 Grand Cypress Blvd | Orlando | FL | 32836 | 28.381945 | -81.510327 | 3.5 | 18 | 0 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |
| 9 | Longwood Galleria | 340-350 Longwood Ave | Boston | MA | 02215 | 42.338544 | -71.106842 | 2.5 | 24 | 1 | ... | None | None | None | None | NaN | None | None | NaN | NaN | None |

city = con.sql("""  
select city,  
"attributes.RestaurantsTableService",   
"attributes.WiFi",   
"attributes.OutdoorSeating",  
"attributes.RestaurantsGoodForGroups",  
"attributes.HappyHour",  
"attributes.Ambience",  
"attributes.HasTv",  
"attributes.Alcohol",  
from restos limit 10;  
"""   
).df()  
city

|  | city | attributes.RestaurantsTableService | attributes.WiFi | attributes.OutdoorSeating | attributes.RestaurantsGoodForGroups | attributes.HappyHour | attributes.Ambience | attributes.HasTV | attributes.Alcohol |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | Boulder | True | u'free' | True | True | True | {'touristy': False, 'hipster': False, 'romanti... | True | 'beer\_and\_wine' |
| 1 | Portland | None | u'free' | False | False | None | {'romantic': False, 'intimate': False, 'touris... | False | u'beer\_and\_wine' |
| 2 | Vancouver | None | u'no' | False | True | None | {'romantic': False, 'intimate': False, 'classy... | False | u'none' |
| 3 | Columbus | None | None | None | None | None | None | None | None |
| 4 | Peabody | False | u'free' | True | True | None | {'romantic': False, 'intimate': False, 'classy... | True | u'none' |
| 5 | Orlando | None | None | None | None | None | None | None | None |
| 6 | Orlando | None | u'free' | None | True | None | None | False | None |
| 7 | Pine Castle | None | None | False | True | None | {'touristy': False, 'hipster': False, 'romanti... | True | 'none' |
| 8 | Orlando | None | 'free' | True | True | None | {'romantic': False, 'intimate': False, 'classy... | True | 'full\_bar' |
| 9 | Boston | None | 'free' | False | True | None | {'romantic': False, 'intimate': False, 'classy... | False | 'full\_bar' |

### 4.3.5 Hypothesis Test 4

### 4.3.6 Ho: Population mean rating of restaurants serving alcohol is less than or equal to Population mean of restaurants not serving alcohol.

### 4.3.7 Ha: Population mean rating of restaurants serving alcohol is greater than the Population mean of restaurants not serving alcohol .

ratings\_serving\_alcohol=con.sql(""" SELECT stars as rating from restos WHERE "attributes.Alcohol" not like '%none%' AND "attributes.Alcohol" not like '%None%' """).df()  
ratings\_not\_serving\_alcohol=con.sql(""" SELECT stars as rating from restos WHERE "attributes.Alcohol" like '%none%' OR "attributes.Alcohol" like '%None%' """).df()  
  
ratings\_serving\_alcohol = ratings\_serving\_alcohol['rating'].dropna().to\_numpy()  
ratings\_not\_serving\_alcohol = ratings\_not\_serving\_alcohol['rating'].dropna().to\_numpy()  
  
if len(ratings\_serving\_alcohol) == 0 or len(ratings\_not\_serving\_alcohol) == 0:  
 print("One of the arrays is empty. Cannot perform t-test.")  
elif np.var(ratings\_serving\_alcohol) == 0 or np.var(ratings\_not\_serving\_alcohol) == 0:  
 print("One of the arrays has no variance. Cannot perform t-test.")  
else:  
 # two-tailed t-test  
 t\_statistic, p\_value\_two\_tailed = spst.ttest\_ind(ratings\_serving\_alcohol, ratings\_not\_serving\_alcohol)  
  
 # Convert to one-tailed p-value  
 if t\_statistic > 0:  
 p\_value\_one\_tailed = p\_value\_two\_tailed / 2  
 else:  
 p\_value\_one\_tailed = 1 - (p\_value\_two\_tailed / 2)  
   
 print(f"T-statistic: {t\_statistic}")  
 print(f"One-tailed P-value: {p\_value\_one\_tailed}")  
  
  
 alpha = 0.05  
 if p\_value\_one\_tailed < alpha:  
 print("Reject the null hypothesis: The mean rating of restaurants serving alcohol is significantly greater.")  
 else:  
 print("Fail to reject the null hypothesis: There is no significant difference in mean ratings.")

T-statistic: 12.800173368523778  
One-tailed P-value: 9.712319877930074e-38  
Reject the null hypothesis: The mean rating of restaurants serving alcohol is significantly greater.

## 4.4 Confidence interval for proportion of restaurants which has outdoor dining area

df\_temp\_3 = con.sql("""  
 SELECT count(\*), "attributes.OutdoorSeating" from restos  
 group by "attributes.OutdoorSeating" ;  
""").df()  
df\_temp\_3

|  | count\_star() | attributes.OutdoorSeating |
| --- | --- | --- |
| 0 | 20619 | False |
| 1 | 2159 | None |
| 2 | 19858 | True |
| 3 | 6164 | None |

proportion\_of\_outdoor\_eating=19858/(20619+19858)  
n=20619+19858;  
se\_proportion\_outdoor = np.sqrt(proportion\_of\_outdoor\_eating \* (1 - proportion\_of\_outdoor\_eating) / n)  
se\_proportion\_outdoor

0.0024847864912894793

margin\_of\_error = 1.96 \* se\_proportion\_outdoor  
confidence\_interval = (proportion\_of\_outdoor\_eating - margin\_of\_error, proportion\_of\_outdoor\_eating + margin\_of\_error)  
confidence\_interval

(0.48572941824978305, 0.49546978129563785)

#### 4.4.0.1 we can say that, the proportion of restaurants having outdoor eating is between the CI 0.48 and 0.49

### 4.4.1 Confidence interval for the mean rating of restaurants where alcholol is served

df\_CI=con.sql(""" SELECT stars as rating from restos WHERE "attributes.Alcohol" not like '%none%' AND "attributes.Alcohol" not like '%None%' """).df()  
  
mean\_rating = df\_CI['rating'].mean()  
sem\_rating = spst.sem(df\_CI['rating'], nan\_policy='omit')   
  
confidence\_level = 0.95  
degrees\_freedom = len(df\_CI['rating']) - 1  
confidence\_interval = spst.t.interval(confidence\_level, degrees\_freedom, loc=mean\_rating, scale=sem\_rating)  
  
print(f"Mean Rating: {mean\_rating}")  
print(f"{confidence\_level\*100}% Confidence Interval for the Mean Rating: {confidence\_interval}")

Mean Rating: 3.583017437452616  
95.0% Confidence Interval for the Mean Rating: (3.5747438580283766, 3.591291016876855)

df1 = con.sql("""  
 SELECT stars, review\_count  
   
 FROM restos  
 ORDER BY stars ASC  
""").df()  
  
X = df1['review\_count']   
y = df1['stars']  
X = sm.add\_constant(X)  
model = sm.OLS(y, X).fit()  
  
print(model.summary())

OLS Regression Results   
==============================================================================  
Dep. Variable: stars R-squared: 0.022  
Model: OLS Adj. R-squared: 0.022  
Method: Least Squares F-statistic: 1091.  
Date: Sat, 31 Aug 2024 Prob (F-statistic): 1.22e-236  
Time: 17:34:14 Log-Likelihood: -56193.  
No. Observations: 48800 AIC: 1.124e+05  
Df Residuals: 48798 BIC: 1.124e+05  
Df Model: 1   
Covariance Type: nonrobust   
================================================================================  
 coef std err t P>|t| [0.025 0.975]  
--------------------------------------------------------------------------------  
const 3.4691 0.004 890.918 0.000 3.462 3.477  
review\_count 0.0005 1.66e-05 33.030 0.000 0.001 0.001  
==============================================================================  
Omnibus: 1761.746 Durbin-Watson: 0.043  
Prob(Omnibus): 0.000 Jarque-Bera (JB): 1957.536  
Skew: -0.488 Prob(JB): 0.00  
Kurtosis: 3.107 Cond. No. 263.  
==============================================================================  
  
Notes:  
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

### 4.4.2 Interpretation of the Linear Regression Analysis

The provided output represents the results of an OLS (Ordinary Least Squares) regression analysis, where the dependent variable is the rating given to a restaurant, and the independent variable is review\_length, which presumably measures the number of words in a review.

#### 4.4.2.1 1. **Regression Coefficients**

* **Intercept (3.4691)**: This value indicates that when the review\_count is zero (i.e., when a review has no words), the expected average rating is approximately 3.4691. This serves as the baseline rating.
* **review\_count (0.0005)**: The coefficient for review\_count is 0.0005, meaning that for every additional word in a review, the rating increases by approximately 0.0005 points.

#### 4.4.2.2 2. **Statistical Significance**

* **P>|t| for review\_length (0.000)**: The p-value for review\_count is extremely small (0.000), which indicates that the relationship between review count and rating is statistically significant. In other words, there’s a very low probability that this relationship is due to random chance.
* **t-statistic for review\_length (33.030)**: The t-value suggests that the effect of review length on rating is strong and positive.

#### 4.4.2.3 3. **Model Fit**

* **R-squared (0.022)**: The R-squared value of 0.022 indicates that only 2.2% of the variability in restaurant ratings can be explained by the length of the review. This suggests that while the relationship between review length and rating is statistically significant, it is not practically significant, as review length only explains a small fraction of the variation in ratings.
* **Adj. R-squared (0.022)**: The adjusted R-squared is nearly identical to the R-squared value, which is typical in models with a single predictor. This confirms that adding more variables to the model might be necessary to better explain the variability in ratings.

#### 4.4.2.4 4. **Overall Model Significance**

* **F-statistic (1091) and Prob (F-statistic) (1.22e-236)**: The F-statistic and its associated p-value (0.00) suggest that the model as a whole is statistically significant. This means that the regression model provides a better fit to the data than a model with no predictors.

#### 4.4.2.5 5. **Diagnostic Tests**

* **Omnibus, Prob(Omnibus), Skew, Kurtosis**: These statistics assess the normality of residuals. The large values and low p-values here suggest that the residuals deviate from normality, which might imply potential issues in the model fit or the need for data transformation.
* **Durbin-Watson (0.043)**: This value indicates no significant autocorrelation in the residuals.
* **Jarque-Bera (1957.536) and Prob(JB) (0.00)**: These values also test for normality. The very low p-value indicates that the residuals are not normally distributed.

### 4.4.3 Summary

* **Relationship**: There is a statistically significant but practically weak inverse relationship between the length of a review and the rating given. Longer reviews are associated with slightly lower ratings.
* **Model Fit**: The model explains only 2.4% of the variability in ratings, suggesting that other factors not included in the model may better explain the ratings.
* **Model Significance**: Despite the weak practical significance, the overall model is statistically significant.
* **Residual Analysis**: There might be concerns regarding the normality of residuals, which could suggest potential issues with the model or the need for further analysis.

In conclusion, while there is a measurable relationship between review length and rating, its impact is minimal, and additional variables may need to be considered to build a more predictive model.