

restaurants

September 4, 2019

```
[4]: import pandas as pd
from pandas.plotting import scatter_matrix
import geopandas as gpd
import matplotlib.pyplot as plt
from shapely.geometry import Point, Polygon
import warnings
warnings.filterwarnings("ignore")
%matplotlib inline

def read_csv_file(csvfile, message):
    '''
    Reading in the dataframe from a csv file
    '''
    print()
    print(message)
    print('csvfile: ', csvfile)
    print()
    df = pd.read_csv(csvfile)
    return df

def inspect_data(df, message, head_lines=3, tail_lines=3):
    '''
    Inspecting the (geo)dataframe
    '''
    print()
    print(message); print()
    print('df.shape:')
    print(df.shape); print()
    print('df.info():')
    print(df.info()); print()
    if head_lines > 0:
        print('df.head(head_lines):')
        print(df.head(head_lines)); print()
    if tail_lines > 0:
        print('df.tail(tail_lines):')
        print(df.tail(tail_lines)); print()
    print()
```

```

def stats_on_columns(df, message, columns):
    """
    Obtain simple statistics on selected columns in the (geo)dataframe
    """
    print()
    print(message); print()
    print(df[columns].describe()); print()

def plot_histogram(df, message, column):
    """
    Plot histograms on one selected column in the (geo)dataframe
    """
    print()
    print(message); print()
    print("column: ", column)
    df[column].hist(label = column, bins = 50, figsize=(10,6), alpha=0.5)
#     plt.xlabel('column')
#     plt.ylabel('counts')
    plt.legend()
    plt.show()
    plt.clf()
    plt.cla()
    plt.close()

def plot_kde(df, message, column):
    """
    Plot kde (analogous to an optimized histogram with an automatic bin size) on
    → one selected column in the (geo)dataframe
    """
    print()
    print(message); print()
    print("column: ", column)
    df[column].plot.kde(figsize=(10,6), alpha=0.5)
#     plt.xlabel('column')
#     plt.ylabel('counts')
    plt.legend()
    plt.show()
    plt.clf()
    plt.cla()
    plt.close()

def get_scatter_matrix(df, message, columns):
    """
    Use scatter_matrix to see if there are correlations between the columns in
    → the (geo)dataframe
    """

```

```

print()
print(message); print()
scatter_matrix(df[columns], figsize=(10,10), alpha=0.5)
plt.show()
plt.clf()
plt.cla()
plt.close()

def get_freq(thelist, message):
    """
    Construct a dictionary to count occurrences for each unique key; thelist is a
    →column of a (geo)dataframe
    """
    print()
    print(message); print()
    freq_dict = {}
    for item in thelist:
        if item in freq_dict:
            freq_dict[item] += 1
        else:
            freq_dict[item] = 1
    return freq_dict

def apply_thresholds(df, message, dict):
    """
    Apply thresholds to the (geo)dataframe
    """
    print()
    print(message); print()
    stars = dict.get('stars', 0.0)
    review_count = dict.get('review_count', 0)
    # if df['state'] != 'AR':
    df = df[df['stars'] >= stars]
    df = df[df['review_count'] >= review_count]
    return df

def do_cycle(gdf, cycle, thresh_dict, message):
    """
    A cycle of doing steps of looking at the data after applying threshold(s)
    """
    head_lines=0
    tail_lines=0

    print()
    print(message); print()
    print('cycle: ', cycle)
    message = 'Inspect the data before applying the thresholds:'

```

```

inspect_data(gdf, message, head_lines, tail_lines)

message = 'Apply the thresholds:'
gdf = apply_thresholds(gdf, message, thresh_dict)

message = 'Re-inspect the data after applying the thresholds:'
inspect_data(gdf, message, head_lines, tail_lines)

columns = ['review_count', 'stars']
message = 'Statistics for some columns:'
stats_on_columns(gdf, message, columns)

message = 'Use scatter_matrix to see the histograms and also to see if there_
→are correlations between the columns:'
get_scatter_matrix(gdf, message, columns)
message = 'Get some simple statistics and plot histograms for some columns:'
columns = ['review_count', 'stars']

message = 'Construct a dictionary to count number restaurants in each unique_
→state/province'
# gdf.sort_values(by=['state'])
states_dict = get_freq(gdf['state'], message)
print(states_dict)
print()

message = 'Geographical distribution:'
print(message)
ax = usa1_can1.plot(color='white', edgecolor='black', figsize=(20,12))
gdf.plot(ax=ax, color='red', markersize=50)
plt.show()
plt.clf()
plt.cla()
plt.close()
print()
return gdf

'''
main routine
'''

print()
print()
print('---Start the analysis---')
print()

```

```

http_address = 'https://www.kaggle.com/yelp-dataset/yelp-dataset/version/
↳6#yelp_business.csv'
csvfile = 'business.csv'
message = '''
The purpose of this python tools programming exercise is to extract the
↳restaurant-only
data from the yelp data for USA and Canadian businesses and then analyze it.
The initial data is obtained from
https://www.kaggle.com/yelp-dataset/yelp-dataset/version/6#yelp_business.csv
and saved in a local direcorry as
business.csv
'''
print(message)

message = 'Initial read of the csv file:'
business_df = read_csv_file(csvfile, message)

message = 'Initial inspection of the dataframe:'
inspect_data(business_df, message)

print()
print()
print('---Start removing unnecessary data---')
message = '''
In this process we will reduce the size of the data
from a (192 609 x 15) table (22 MB)
to a (34 392 x 13) table (3.7 MB)
'''
print(message)
print()

head_lines=5
tail_lines=5

message = 'Re-inspect the dataframe after dropping unnecessary column:'
business_df.drop(['Unnamed: 0'], axis=1, inplace=True)
inspect_data(business_df, message, head_lines, tail_lines)

message = 'Inspect the new dataframe containing only restaurants:'
restaurant_df = business_df[business_df['categories'].str.
↳contains('Restaurants') == True]
inspect_data(restaurant_df, message, head_lines, tail_lines)

message = 'Inspect the dataframe containing only restaurants that are still open:
↳'
restaurant_df = restaurant_df[restaurant_df.is_open != 0]
inspect_data(restaurant_df, message, head_lines, tail_lines)

```

```

message = 'Inspect the dataframe after dropping the now unnecessary column_
↳is_open:'
restaurant_df.drop(['is_open'], axis=1, inplace=True)
inspect_data(restaurant_df, message, head_lines, tail_lines)

head_lines=0
tail_lines=0

message = 'Inspect the dataframe containing only restaurants that have a valid_
↳address:'
restaurant_df = restaurant_df[restaurant_df['address'].str.contains('NaN') ==_
↳False]
inspect_data(restaurant_df, message, head_lines, tail_lines)

message = 'Inspect the dataframe containing only restaurants that have valid_
↳hours:'
restaurant_df = restaurant_df[restaurant_df['hours'].str.contains('NaN') ==_
↳False]
inspect_data(restaurant_df, message, head_lines, tail_lines)

message = 'Construct a dictionary to count number restaurants in each unique_
↳state/province'
#restaurant_df.sort_values(by=['state'])
states_dict = get_freq(restaurant_df['state'], message)
print(states_dict)
print()

print()
print()
print('---Finished removing unnecessary data---')
print()

print()
print()
print('---Analyze some columns and see the geographic distribution of the_
↳data---')
cycle = 0
thresh_dict = {'stars': 0.0, 'review_count': 0, 'exclude_states': []}
print('stars >= ', thresh_dict.get('stars', 0.0))
print('review_count >= ', thresh_dict.get('review_count', 0))
print('exclude_states: ', thresh_dict.get('exclude_states', []))
print()

columns = ['review_count', 'stars']
message = 'Statistics for some columns:'

```

```

stats_on_columns(restaurant_df, message, columns)

message = 'Histogram for a column:'
for column in columns:
    print('column: ', column)
    plot_histogram(restaurant_df, message, column)

'''
message = 'kde for a column:'
for column in columns:
    print('column: ', column)
    plot_kde(restaurant_df, message, column)

'''

message = 'Use scatter_matrix to see the histograms and also to see if there are
→correlations between the columns:'
get_scatter_matrix(restaurant_df, message, columns)

message = 'Create a geoDataFrame of the restaurant data and use it to'
print(message)

# https://medium.com/@shakason/4219d2106d
# how-to-convert-latitude-longitude-columns-in-csv-to-geometry-column-using-python-4219d2106d
# creating a geometry column
geometry = [Point(xy) for xy in zip(restaurant_df['longitude'],
→restaurant_df['latitude'])]
# Coordinate reference system : WGS84
crs = {'init': 'epsg:4326'}
# Creating a Geographic data frame
restaurant_gdf = gpd.GeoDataFrame(restaurant_df, crs=crs, geometry=geometry)

print('restaurant_gdf.head():')
print(restaurant_gdf.head()); print()

restaurant_gdf.drop(['longitude', 'latitude'], axis=1, inplace=True)
print('restaurant_gdf.head():')
print(restaurant_gdf.head()); print()

print('restaurant_df.head():')
print(restaurant_df.head()); print()

restaurant_df.drop(['geometry'], axis=1, inplace=True)
print('restaurant_df.head():')
print(restaurant_df.head()); print()

```

```

message = 'From now on will deal with the restaurant data only in the form of_
→the geoDataFrame and not the original DataFrame'
print(message)
print()

message = 'Plot the geographic distribution of the restaurants:'
print(message)
#http://geopandas.org/gallery/create_geopandas_from_pandas.html
world = gpd.read_file(gpd.datasets.get_path('naturalearth_lowres'))
# restrict to North America
ax = world[world.continent == 'North America'].plot(color='white',_
→edgecolor='black', figsize=(20,12))
#cities[cities.continent == 'North America'].plot(ax=ax, color='black')
# now plot the GeoDataFrame
restaurant_gdf.plot(ax=ax, color='red')
plt.show()
print()

message = 'It can be seen that restaurant data only exist for the continental_
→USA (i.e., no Alaska or Hawaii) and a few bordering provinces of Canada'
print(message)
message = 'Thus restrict the map to get a better picture'
print(message)
print()

# downloaded maps in the form of shape files from https://gadm.org/
→download_country_v3.html
# these are the level 1 (_1) files which show the state or province boundaries

# usa
usa1 = gpd.read_file('gadm36_USA_1.shp')
print('usa1.head():')
print(usa1.head()); print
usa1 = usa1[['NAME_1', 'geometry']]
# changing columns using .columns()
usa1.columns = ['state_or_province', 'geometry']
# remove unneeded states
usa1 = usa1[usa1['state_or_province'] != 'Alaska']
usa1 = usa1[usa1['state_or_province'] != 'Hawaii']
print('usa1:')
print(usa1); print

# canada
can1 = gpd.read_file('gadm36_CAN_1.shp')
print('can1.head(3):')
print(can1.head(3)); print
can1 = can1[['NAME_1', 'geometry']]

```



```

# changing columns using .columns()
can1.columns = ['state_or_province', 'geometry']
# remove unneeded provinces
can1 = can1[can1['state_or_province'] != 'Northwest Territories']
can1 = can1[can1['state_or_province'] != 'Nunavut']
can1 = can1[can1['state_or_province'] != 'Nova Scotia']
can1 = can1[can1['state_or_province'] != 'Yukon']
can1 = can1[can1['state_or_province'] != 'Prince Edward Island']
can1 = can1[can1['state_or_province'] != 'Newfoundland and Labrador']
print('can1:')
print(can1); print
# concatenate
usa1_can1 = usa1.append(can1)
print('usa1_can1:')
print(usa1_can1); print

message = 'Construct a dictionary to count number restaurants in each unique_
→state/province'
#restaurant_df.sort_values(by=['state'])
states_dict = get_freq(restaurant_df['state'], message)
print(states_dict)
print()

ax = usa1_can1.plot(color='white', edgecolor='black', figsize=(20,12))
restaurant_gdf.plot(ax=ax, color='red', markersize=50)
plt.show()
print()

'''
for state in states_dict:
    print('state: ', state)
    ax = usa1_can1.plot(color='white', edgecolor='black', figsize=(20,12))
    restaurant_gdf[restaurant_gdf['state']==state].plot(ax=ax, color='red',
→markersize=50)
    plt.show()
    plt.clf()
    plt.cla()
    plt.close()
    print()
'''

print()
print()
print('---It can be seen that about 8k restaurants got very few reviews. Make a_
→decision to recommend based on some thresholds which are:---')
print('Also note that there are locations in 11 states or provinces. Try to_
→retain at least 1 in each.')

```

```

cycle = 1
thresh_dict = {'stars': 4.5, 'review_count': 100, 'exclude_states': []}
print('stars >= ', thresh_dict.get('stars', 0.0))
print('review_count >= ', thresh_dict.get('review_count', 0))
print('exclude_states: ', thresh_dict.get('exclude_states', []))
message = 'Doing a cycle of applying thresholds'
restaurant_gdf = do_cycle(restaurant_gdf, cycle, thresh_dict, message)
print()

cycle = 2
thresh_dict = {'stars': 5.0, 'review_count': 200, 'exclude_states': []}
print('stars >= ', thresh_dict.get('stars', 0.0))
print('review_count >= ', thresh_dict.get('review_count', 0))
print('exclude_states: ', thresh_dict.get('exclude_states', []))
message = 'Doing a cycle of applying thresholds'
restaurant_gdf = do_cycle(restaurant_gdf, cycle, thresh_dict, message)
print()

message = '''
Thus by tightening thresholds we reduced the data to only 19 restaurants in 4
↳states/provinces:
{'AZ': 9, 'NV': 8, 'QC': 1, 'ON': 1}
The user can then select the appropriate restaurant based on cuisine, etc.
'''

print(message)
print()

print('restaurant_gdf:')
print(restaurant_gdf)
print()

```

---Start the analysis---

The purpose of this python tools programming exercise is to extract the restaurant-only data from the yelp data for USA and Canadian businesses and then analyze it. The initial data is obtained from https://www.kaggle.com/yelp-dataset/yelp-dataset/version/6#yelp_business.csv and saved in a local direcorey as business.csv

Initial read of the csv file:

```
csvfile: business.csv
```

```
Initial inspection of the dataframe:
```

```
df.shape:  
(192609, 15)
```

```
df.info():  
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 192609 entries, 0 to 192608  
Data columns (total 15 columns):  
Unnamed: 0      192609 non-null int64  
address         184927 non-null object  
attributes      163773 non-null object  
business_id     192609 non-null object  
categories      192127 non-null object  
city            192608 non-null object  
hours           147779 non-null object  
is_open         192609 non-null int64  
latitude        192609 non-null float64  
longitude       192609 non-null float64  
name            192609 non-null object  
postal_code     191950 non-null object  
review_count    192609 non-null int64  
stars           192609 non-null float64  
state           192609 non-null object  
dtypes: float64(3), int64(3), object(9)  
memory usage: 22.0+ MB  
None
```

```
df.head(head_lines):
```

```
   Unnamed: 0      address \  
0           0  2818 E Camino Acequia Drive  
1           1      30 Eglinton Avenue W  
2           2  10110 Johnston Rd, Ste 15  
  
   attributes      business_id \  
0  {'GoodForKids': 'False'}  1SWheh84yJXfytovILXOAQ  
1  {'RestaurantsReservations': 'True', 'GoodForMe...  QXAEGB4oINsVuTFxEYKFQ  
2  {'GoodForKids': 'True', 'NoiseLevel': 'u'avera...  gnKjwL_1w79qoiV3IC_xQQ  
  
   categories      city \  
0      Golf, Active Life      Phoenix  
1  Specialty Food, Restaurants, Dim Sum, Imported...  Mississauga  
2      Sushi Bars, Restaurants, Japanese      Charlotte  
  
   hours  is_open  latitude \  

```

0		NaN	0	33.522143
1	{'Monday': '9:0-0:0', 'Tuesday': '9:0-0:0', 'W...		1	43.605499
2	{'Monday': '17:30-21:30', 'Wednesday': '17:30-...		1	35.092564

	longitude	name	postal_code	review_count	stars	\
0	-112.018481	Arizona Biltmore Golf Club	85016	5	3.0	
1	-79.652289	Emerald Chinese Restaurant	L5R 3E7	128	2.5	
2	-80.859132	Musashi Japanese Restaurant	28210	170	4.0	

	state
0	AZ
1	ON
2	NC

df.tail(tail_lines):

	Unnamed: 0	address	\
192606	192606	21230 N 22nd St	
192607	192607	4406 E Main St	
192608	192608	3735 S Las Vegas Blvd	

	attributes	\
192606	{'BusinessAcceptsCreditCards': 'True', 'ByAppo...	
192607	{'RestaurantsPriceRange2': '1', 'BusinessAccep...	
192608	{'BikeParking': 'False', 'RestaurantsPriceRang...	

	business_id	\
192606	MiEyUDKTjeci5TMfxVZPpg	
192607	zNMupayB2jEHVD0ji8sxoQ	
192608	c1f_VAX1KIK8-JoVhjbY0w	

	categories	city	\
192606	Home Services, Contractors, Landscaping, Mason...	Phoenix	
192607	Beauty & Spas, Barbers	Mesa	
192608	Massage, Beauty & Spas	Las Vegas	

	hours	is_open	latitude	\
192606	{'Monday': '7:0-15:0', 'Tuesday': '7:0-15:0', ...	1	33.679992	
192607	{'Tuesday': '8:30-17:30', 'Wednesday': '8:30-1...	1	33.416137	
192608	{'Monday': '10:0-0:0', 'Tuesday': '10:0-0:0', ...	0	36.107267	

	longitude	name	postal_code	review_count	stars	\
192606	-112.035569	Phoenix Pavers	85024	14	4.5	
192607	-111.735743	Beasley's Barber Shop	85205	15	4.5	
192608	-115.171920	Oriental Relax	89109	3	4.0	

	state
192606	AZ
192607	AZ

192608 NV

---Start removing unnecessary data---

In this process we will reduce the size of the data
from a (192 609 x 15) table (22 MB)
to a (34 392 x 13) table (3.7 MB)

Re-inspect the dataframe after dropping unnecessary column:

```
df.shape:  
(192609, 14)
```

```
df.info():  
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 192609 entries, 0 to 192608  
Data columns (total 14 columns):  
address           184927 non-null object  
attributes        163773 non-null object  
business_id       192609 non-null object  
categories        192127 non-null object  
city              192608 non-null object  
hours             147779 non-null object  
is_open           192609 non-null int64  
latitude          192609 non-null float64  
longitude         192609 non-null float64  
name              192609 non-null object  
postal_code       191950 non-null object  
review_count      192609 non-null int64  
stars             192609 non-null float64  
state             192609 non-null object  
dtypes: float64(3), int64(2), object(9)  
memory usage: 20.6+ MB  
None
```

```
df.head(head_lines):  
      address \  
0    2818 E Camino Acequia Drive  
1         30 Eglinton Avenue W  
2    10110 Johnston Rd, Ste 15  
3    15655 W Roosevelt St, Ste 237  
4    4209 Stuart Andrew Blvd, Ste F
```

	attributes	business_id \
0	{'GoodForKids': 'False'}	1SWheh84yJXfytovILX0AQ
1	{'RestaurantsReservations': 'True', 'GoodForMe...	QXAEGFB4oINsVuTFxEYKFQ
2	{'GoodForKids': 'True', 'NoiseLevel': 'u'avera...	gnKjwL_1w79qoiV3IC_xQQ
3	NaN	xvX2CttrVhyG2z1dFg_0xw
4	{'BusinessAcceptsBitcoin': 'False', 'ByAppoint...	Hhyx0kGAM07SRyt1Q4wMFQ

	categories	city \
0	Golf, Active Life	Phoenix
1	Specialty Food, Restaurants, Dim Sum, Imported...	Mississauga
2	Sushi Bars, Restaurants, Japanese	Charlotte
3	Insurance, Financial Services	Goodyear
4	Plumbing, Shopping, Local Services, Home Servi...	Charlotte

	hours	is_open	latitude \
0	NaN	0	33.522143
1	{'Monday': '9:0-0:0', 'Tuesday': '9:0-0:0', 'W...	1	43.605499
2	{'Monday': '17:30-21:30', 'Wednesday': '17:30-...	1	35.092564
3	{'Monday': '8:0-17:0', 'Tuesday': '8:0-17:0', ...	1	33.455613
4	{'Monday': '7:0-23:0', 'Tuesday': '7:0-23:0', ...	1	35.190012

	longitude	name	postal_code	review_count \
0	-112.018481	Arizona Biltmore Golf Club	85016	5
1	-79.652289	Emerald Chinese Restaurant	L5R 3E7	128
2	-80.859132	Musashi Japanese Restaurant	28210	170
3	-112.395596	Farmers Insurance - Paul Lorenz	85338	3
4	-80.887223	Queen City Plumbing	28217	4

	stars	state
0	3.0	AZ
1	2.5	ON
2	4.0	NC
3	5.0	AZ
4	4.0	NC

df.tail(tail_lines):

	address \
192604	NaN
192605	17440 Yonge Street
192606	21230 N 22nd St
192607	4406 E Main St
192608	3735 S Las Vegas Blvd

	attributes \
192604	{'BusinessAcceptsCreditCards': 'True'}
192605	{'RestaurantsPriceRange2': '2', 'BikeParking':...
192606	{'BusinessAcceptsCreditCards': 'True', 'ByAppo...
192607	{'RestaurantsPriceRange2': '1', 'BusinessAccep...

```
192608 {'BikeParking': 'False', 'RestaurantsPriceRang...
```

```
business_id \
192604 nqb4kWc0wp8bFxxfvaDpZQ
192605 vY2nLU5K20Pee-FdG0br1g
192606 MiEyUDKTjeci5TMfxVZPpg
192607 zNMupayB2jEHVD0ji8sxoQ
192608 c1f_VAX1KIK8-JoVhjbY0w
```

```
categories city \
192604 Water Purification Services, Water Heater Inst... North Las Vegas
192605 Books, Mags, Music & Video, Shopping Newmarket
192606 Home Services, Contractors, Landscaping, Mason... Phoenix
192607 Beauty & Spas, Barbers Mesa
192608 Massage, Beauty & Spas Las Vegas
```

```
hours is_open latitude \
192604 {'Monday': '0:0-0:0', 'Tuesday': '0:0-0:0', 'W... 1 36.213732
192605 NaN 1 44.052658
192606 {'Monday': '7:0-15:0', 'Tuesday': '7:0-15:0', ... 1 33.679992
192607 {'Tuesday': '8:30-17:30', 'Wednesday': '8:30-1... 1 33.416137
192608 {'Monday': '10:0-0:0', 'Tuesday': '10:0-0:0', ... 0 36.107267
```

```
longitude name postal_code review_count stars \
192604 -115.177059 Sanderson Plumbing 89032 9 5.0
192605 -79.481850 Chapters L3Y 6Y9 3 4.5
192606 -112.035569 Phoenix Pavers 85024 14 4.5
192607 -111.735743 Beasley's Barber Shop 85205 15 4.5
192608 -115.171920 Oriental Relax 89109 3 4.0
```

```
state
192604 NV
192605 ON
192606 AZ
192607 AZ
192608 NV
```

Inspect the new dataframe containing only restaurants:

```
df.shape:
(59371, 14)
```

```
df.info():
<class 'pandas.core.frame.DataFrame'>
Int64Index: 59371 entries, 1 to 192603
Data columns (total 14 columns):
```

```

address          58955 non-null object
attributes       57161 non-null object
business_id     59371 non-null object
categories       59371 non-null object
city            59371 non-null object
hours           45376 non-null object
is_open         59371 non-null int64
latitude        59371 non-null float64
longitude       59371 non-null float64
name            59371 non-null object
postal_code     59269 non-null object
review_count    59371 non-null int64
stars           59371 non-null float64
state           59371 non-null object
dtypes: float64(3), int64(2), object(9)
memory usage: 6.8+ MB
None

```

```
df.head(head_lines):
```

```

              address \
1          30 Eglinton Avenue W
2      10110 Johnston Rd, Ste 15
11     2450 E Indian School Rd
13          5981 Andrews Rd
17  1775 E Tropicana Ave, Ste 29

```

```

              attributes          business_id \
1  {'RestaurantsReservations': 'True', 'GoodForMe...  QXAEgFB4oINsVuTFxEYKFQ
2  {'GoodForKids': 'True', 'NoiseLevel': "u'avera...  gnKjwL_1w79qoiV3IC_xQQ
11 {'RestaurantsTakeOut': 'True', 'BusinessParkin...  1Dfx3zM-rW4n-31KeC8sJg
13 {'RestaurantsPriceRange2': '2', 'BusinessAccep...  fweCYi8FmbJXHCqLnwuk8w
17 {'OutdoorSeating': 'False', 'BusinessAcceptsCr...  PZ-LZzSlhSe9utkQYU8pFg

```

```

              categories          city \
1  Specialty Food, Restaurants, Dim Sum, Imported...  Mississauga
2          Sushi Bars, Restaurants, Japanese        Charlotte
11 Restaurants, Breakfast & Brunch, Mexican, Taco...  Phoenix
13      Italian, Restaurants, Pizza, Chicken Wings  Mentor-on-the-Lake
17          Restaurants, Italian        Las Vegas

```

```

              hours  is_open  latitude \
1  {'Monday': '9:0-0:0', 'Tuesday': '9:0-0:0', 'W...    1  43.605499
2  {'Monday': '17:30-21:30', 'Wednesday': '17:30-...    1  35.092564
11 {'Monday': '7:0-0:0', 'Tuesday': '7:0-0:0', 'W...    1  33.495194
13 {'Monday': '10:0-0:0', 'Tuesday': '10:0-0:0', ...    1  41.708520
17                                NaN    0  36.100016

```

```

longitude          name postal_code  review_count  stars \

```


1	-79.652289	Emerald Chinese Restaurant	L5R 3E7	128	2.5
2	-80.859132	Musashi Japanese Restaurant	28210	170	4.0
11	-112.028588	Taco Bell	85016	18	3.0
13	-81.359556	Marco's Pizza	44060	16	4.0
17	-115.128528	Carluccio's Tivoli Gardens	89119	40	4.0

state	
1	ON
2	NC
11	AZ
13	OH
17	NV

df.tail(tail_lines):

address \	
192587	578 Yonge Street
192589	3863 Medina Rd
192595	1450 Kingston Rd
192596	948 Boulevard de Maisonneuve Est
192603	1911 N Cleveland-Massillon Rd

attributes \	
192587	{'RestaurantsPriceRange2': '2', 'RestaurantsGo...
192589	{'RestaurantsPriceRange2': '2', 'HasTV': 'Fals...
192595	{'BikeParking': 'True', 'RestaurantsAttire': "...
192596	{'GoodForKids': 'True', 'WiFi': "u'no'", 'Rest...
192603	{'OutdoorSeating': 'True', 'BikeParking': 'Tru...

business_id \	
192587	oSOCnUbyv0GUoD3L8_3UPQ
192589	ghovD5ZTGDQ5Q2U4ERddWw
192595	h3QErqS30Zg LJ5Tb6-sLyQ
192596	KnafX7T6qSAmSrLhd709vA
192603	p1eUTUB_5hf1enoKYBV3DA

categories		city \
192587	Restaurants, Thai	Toronto
192589	Burgers, Restaurants, Fast Food, American (New)	Fairlawn
192595	Restaurants, Soup, Chinese, Caribbean	Pickering
192596	Vietnamese, Soup, Restaurants	Montréal
192603	Restaurants, Italian	Bath

hours	is_open	latitude \
192587	{'Monday': '0:0-0:0', 'Tuesday': '11:0-23:0', ...}	1 43.665120
192589	{'Monday': '11:0-22:0', 'Tuesday': '11:0-22:0'...	1 41.136561
192595	{'Monday': '11:0-21:30', 'Tuesday': '11:0-21:3...	1 43.841844
192596	{'Monday': '12:0-21:0', 'Tuesday': '12:0-21:0'...	1 45.517430
192603	{'Monday': '0:0-0:0', 'Tuesday': '17:0-22:0', ...}	1 41.187547

	longitude	name	postal_code	review_count	stars	\
192587	-79.384809	Thai Fantasy	M4Y 1Z3	113	4.0	
192589	-81.639712	Five Guys	44333	22	4.0	
192595	-79.083881	Asia Hut	L1V 1C1	4	4.5	
192596	-73.558873	Pho Maisonneuve	H2L 1Z1	25	4.0	
192603	-81.635791	Ken Stewart's Tre Belle	44210	39	3.5	

	state
192587	ON
192589	OH
192595	ON
192596	QC
192603	OH

Inspect the dataframe containing only restaurants that are still open:

```
df.shape:
(42237, 14)
```

```
df.info():
<class 'pandas.core.frame.DataFrame'>
Int64Index: 42237 entries, 1 to 192603
Data columns (total 14 columns):
address      41913 non-null object
attributes   40477 non-null object
business_id  42237 non-null object
categories   42237 non-null object
city         42237 non-null object
hours        34568 non-null object
is_open      42237 non-null int64
latitude     42237 non-null float64
longitude    42237 non-null float64
name         42237 non-null object
postal_code  42147 non-null object
review_count 42237 non-null int64
stars        42237 non-null float64
state        42237 non-null object
dtypes: float64(3), int64(2), object(9)
memory usage: 4.8+ MB
None
```

```
df.head(head_lines):
```

	address	\
1	30 Eglinton Avenue W	
2	10110 Johnston Rd, Ste 15	

```

11          2450 E Indian School Rd
13          5981 Andrews Rd
23 Center Core - Food Court, Fl 3, Pittsburgh Int...

```

```

          attributes          business_id \
1  {'RestaurantsReservations': 'True', 'GoodForMe... QXAEGFB4oINsVuTFxEYKFQ
2  {'GoodForKids': 'True', 'NoiseLevel': "u'avera... gnKjwL_1w79qoiV3IC_xQQ
11 {'RestaurantsTakeOut': 'True', 'BusinessParkin... 1Dfx3zM-rW4n-31KeC8sJg
13 {'RestaurantsPriceRange2': '2', 'BusinessAccep... fweCYi8FmbJXHCqLnwuk8w
23 {'RestaurantsTakeOut': 'True', 'BusinessParkin... 1RHY4K3BD22FK7Cfftn8Mg

```

```

          categories          city \
1  Specialty Food, Restaurants, Dim Sum, Imported... Mississauga
2          Sushi Bars, Restaurants, Japanese      Charlotte
11 Restaurants, Breakfast & Brunch, Mexican, Taco... Phoenix
13          Italian, Restaurants, Pizza, Chicken Wings Mentor-on-the-Lake
23 Sandwiches, Salad, Restaurants, Burgers, Comfo... Pittsburgh

```

```

          hours  is_open  latitude \
1  {'Monday': '9:0-0:0', 'Tuesday': '9:0-0:0', 'W... 1 43.605499
2  {'Monday': '17:30-21:30', 'Wednesday': '17:30-... 1 35.092564
11 {'Monday': '7:0-0:0', 'Tuesday': '7:0-0:0', 'W... 1 33.495194
13 {'Monday': '10:0-0:0', 'Tuesday': '10:0-0:0', ... 1 41.708520
23          NaN          1 40.496177

```

```

          longitude          name postal_code  review_count  stars \
1  -79.652289  Emerald Chinese Restaurant    L5R 3E7        128    2.5
2  -80.859132  Musashi Japanese Restaurant    28210        170    4.0
11 -112.028588          Taco Bell          85016         18    3.0
13  -81.359556          Marco's Pizza          44060         16    4.0
23  -80.246011          Marathon Diner          15231         35    4.0

```

```

state
1    ON
2    NC
11   AZ
13   OH
23   PA

```

```
df.tail(tail_lines):
```

```

          address \
192587          578 Yonge Street
192589          3863 Medina Rd
192595          1450 Kingston Rd
192596  948 Boulevard de Maisonneuve Est
192603  1911 N Cleveland-Massillon Rd

```

```
attributes \
```

```

192587 {'RestaurantsPriceRange2': '2', 'RestaurantsGo...
192589 {'RestaurantsPriceRange2': '2', 'HasTV': 'Fals...
192595 {'BikeParking': 'True', 'RestaurantsAttire': "...
192596 {'GoodForKids': 'True', 'WiFi': "u'no'", 'Rest...
192603 {'OutdoorSeating': 'True', 'BikeParking': 'Tru...

```

```

                business_id \
192587 oS0CnUbyv0GUoD3L8_3UPQ
192589 ghovD5ZTGDQ5Q2U4ERddWw
192595 h3QErqS30Zg LJ5Tb6-sLyQ
192596 KnafX7T6qSAmSrLhd709vA
192603 p1eUTUB_5hf1enoKYBV3DA

```

```

                                categories      city \
192587                                Restaurants, Thai    Toronto
192589  Burgers, Restaurants, Fast Food, American (New)  Fairlawn
192595                                Restaurants, Soup, Chinese, Caribbean  Pickering
192596                                Vietnamese, Soup, Restaurants  Montréal
192603                                Restaurants, Italian    Bath

```

```

                                hours  is_open  latitude \
192587 {'Monday': '0:0-0:0', 'Tuesday': '11:0-23:0', ...      1  43.665120
192589 {'Monday': '11:0-22:0', 'Tuesday': '11:0-22:0'...      1  41.136561
192595 {'Monday': '11:0-21:30', 'Tuesday': '11:0-21:3...      1  43.841844
192596 {'Monday': '12:0-21:0', 'Tuesday': '12:0-21:0'...      1  45.517430
192603 {'Monday': '0:0-0:0', 'Tuesday': '17:0-22:0', ...      1  41.187547

```

```

                longitude      name postal_code  review_count  stars \
192587 -79.384809      Thai Fantasy    M4Y 1Z3      113    4.0
192589 -81.639712      Five Guys      44333      22    4.0
192595 -79.083881      Asia Hut      L1V 1C1      4    4.5
192596 -73.558873      Pho Maisonneuve  H2L 1Z1      25    4.0
192603 -81.635791  Ken Stewart's Tre Belle    44210      39    3.5

```

```

                state
192587    ON
192589    OH
192595    ON
192596    QC
192603    OH

```

Inspect the dataframe after dropping the now unnecessary column is_open:

```

df.shape:
(42237, 13)

```

```
df.info():
<class 'pandas.core.frame.DataFrame'>
Int64Index: 42237 entries, 1 to 192603
Data columns (total 13 columns):
address          41913 non-null object
attributes        40477 non-null object
business_id      42237 non-null object
categories        42237 non-null object
city             42237 non-null object
hours            34568 non-null object
latitude         42237 non-null float64
longitude        42237 non-null float64
name             42237 non-null object
postal_code      42147 non-null object
review_count     42237 non-null int64
stars           42237 non-null float64
state           42237 non-null object
dtypes: float64(3), int64(1), object(9)
memory usage: 4.5+ MB
None
```

```
df.head(head_lines):
```

	address \
1	30 Eglinton Avenue W
2	10110 Johnston Rd, Ste 15
11	2450 E Indian School Rd
13	5981 Andrews Rd
23	Center Core - Food Court, Fl 3, Pittsburgh Int...

	attributes	business_id \
1	{'RestaurantsReservations': 'True', 'GoodForMe...	QXAEGFB4oINsVuTFxEYKFQ
2	{'GoodForKids': 'True', 'NoiseLevel': 'u'avera...	gnKjwL_1w79qoiV3IC_xQQ
11	{'RestaurantsTakeOut': 'True', 'BusinessParkin...	1Dfx3zM-rW4n-31KeC8sJg
13	{'RestaurantsPriceRange2': '2', 'BusinessAccep...	fweCYi8FmbJXHCqLnwuk8w
23	{'RestaurantsTakeOut': 'True', 'BusinessParkin...	1RHY4K3BD22FK7Cfftn8Mg

	categories	city \
1	Specialty Food, Restaurants, Dim Sum, Imported...	Mississauga
2	Sushi Bars, Restaurants, Japanese	Charlotte
11	Restaurants, Breakfast & Brunch, Mexican, Taco...	Phoenix
13	Italian, Restaurants, Pizza, Chicken Wings	Mentor-on-the-Lake
23	Sandwiches, Salad, Restaurants, Burgers, Comfo...	Pittsburgh

	hours	latitude	longitude \
1	{'Monday': '9:0-0:0', 'Tuesday': '9:0-0:0', 'W...	43.605499	-79.652289
2	{'Monday': '17:30-21:30', 'Wednesday': '17:30-...	35.092564	-80.859132
11	{'Monday': '7:0-0:0', 'Tuesday': '7:0-0:0', 'W...	33.495194	-112.028588
13	{'Monday': '10:0-0:0', 'Tuesday': '10:0-0:0', ...	41.708520	-81.359556

23

NaN 40.496177 -80.246011

	name	postal_code	review_count	stars	state
1	Emerald Chinese Restaurant	L5R 3E7	128	2.5	ON
2	Musashi Japanese Restaurant	28210	170	4.0	NC
11	Taco Bell	85016	18	3.0	AZ
13	Marco's Pizza	44060	16	4.0	OH
23	Marathon Diner	15231	35	4.0	PA

df.tail(tail_lines):

	address
192587	578 Yonge Street
192589	3863 Medina Rd
192595	1450 Kingston Rd
192596	948 Boulevard de Maisonneuve Est
192603	1911 N Cleveland-Massillon Rd

	attributes
192587	{'RestaurantsPriceRange2': '2', 'RestaurantsGo...
192589	{'RestaurantsPriceRange2': '2', 'HasTV': 'Fals...
192595	{'BikeParking': 'True', 'RestaurantsAttire': "...
192596	{'GoodForKids': 'True', 'WiFi': "u'no'", 'Rest...
192603	{'OutdoorSeating': 'True', 'BikeParking': 'Tru...

	business_id
192587	oSOCnUbyv0GUoD3L8_3UPQ
192589	ghovD5ZTGDQ5Q2U4ERddWw
192595	h3QErqS30Zg LJ5Tb6-sLyQ
192596	KnafX7T6qSAmSrLhd709vA
192603	p1eUTUB_5hf1enoKYBV3DA

	categories	city
192587	Restaurants, Thai	Toronto
192589	Burgers, Restaurants, Fast Food, American (New)	Fairlawn
192595	Restaurants, Soup, Chinese, Caribbean	Pickering
192596	Vietnamese, Soup, Restaurants	Montréal
192603	Restaurants, Italian	Bath

	hours	latitude
192587	{'Monday': '0:0-0:0', 'Tuesday': '11:0-23:0', ...	43.665120
192589	{'Monday': '11:0-22:0', 'Tuesday': '11:0-22:0'...	41.136561
192595	{'Monday': '11:0-21:30', 'Tuesday': '11:0-21:3...	43.841844
192596	{'Monday': '12:0-21:0', 'Tuesday': '12:0-21:0'...	45.517430
192603	{'Monday': '0:0-0:0', 'Tuesday': '17:0-22:0', ...	41.187547

	longitude	name	postal_code	review_count	stars
192587	-79.384809	Thai Fantasy	M4Y 1Z3	113	4.0
192589	-81.639712	Five Guys	44333	22	4.0

192595	-79.083881	Asia Hut	L1V 1C1	4	4.5
192596	-73.558873	Pho Maisonneuve	H2L 1Z1	25	4.0
192603	-81.635791	Ken Stewart's Tre Belle	44210	39	3.5

	state
192587	ON
192589	OH
192595	ON
192596	QC
192603	OH

Inspect the dataframe containing only restaurants that have a valid address:

```
df.shape:
(41913, 13)
```

```
df.info():
<class 'pandas.core.frame.DataFrame'>
Int64Index: 41913 entries, 1 to 192603
Data columns (total 13 columns):
address          41913 non-null object
attributes       40185 non-null object
business_id      41913 non-null object
categories       41913 non-null object
city             41913 non-null object
hours            34392 non-null object
latitude         41913 non-null float64
longitude        41913 non-null float64
name             41913 non-null object
postal_code      41877 non-null object
review_count     41913 non-null int64
stars            41913 non-null float64
state            41913 non-null object
dtypes: float64(3), int64(1), object(9)
memory usage: 4.5+ MB
None
```

Inspect the dataframe containing only restaurants that have valid hours:

```
df.shape:
(34392, 13)
```

```
df.info():
<class 'pandas.core.frame.DataFrame'>
```

```

Int64Index: 34392 entries, 1 to 192603
Data columns (total 13 columns):
address      34392 non-null object
attributes   33237 non-null object
business_id  34392 non-null object
categories   34392 non-null object
city         34392 non-null object
hours        34392 non-null object
latitude     34392 non-null float64
longitude    34392 non-null float64
name         34392 non-null object
postal_code  34389 non-null object
review_count 34392 non-null int64
stars        34392 non-null float64
state        34392 non-null object
dtypes: float64(3), int64(1), object(9)
memory usage: 3.7+ MB
None

```

Construct a dictionary to count number restaurants in each unique state/province

```

{'ON': 8399, 'NC': 2593, 'AZ': 6472, 'OH': 3404, 'NV': 4295, 'PA': 2380, 'QC': 3357, 'AB': 1840, 'WI': 1061, 'IL': 368, 'NY': 11, 'SC': 203, 'NM': 1, 'VA': 1, 'NE': 1, 'WA': 1, 'XWY': 1, 'BC': 1, 'VT': 1, 'AR': 1, 'AL': 1}

```

---Finished removing unnecessary data---

---Analyze some columns and see the geographic distribution of the data---

```

stars >= 0.0
review_count >= 0
exclude_states: []

```

Statistics for some columns:

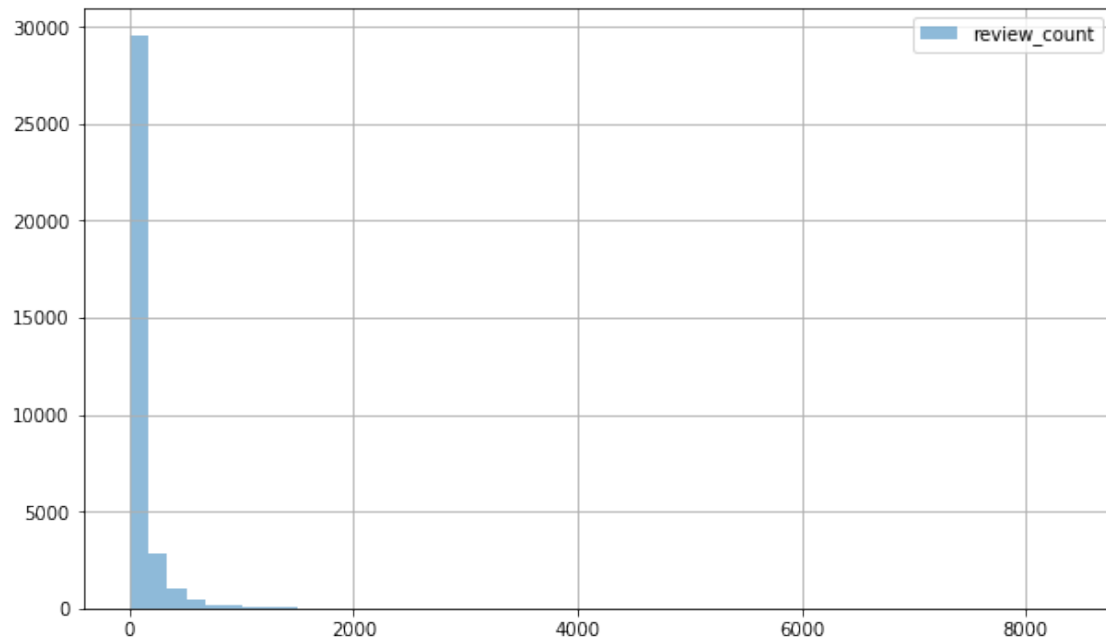
	review_count	stars
count	34392.000000	34392.000000
mean	96.282944	3.524395
std	217.606155	0.784721
min	3.000000	1.000000
25%	12.000000	3.000000
50%	35.000000	3.500000

75%	96.000000	4.000000
max	8348.000000	5.000000

column: review_count

Histogram for a column:

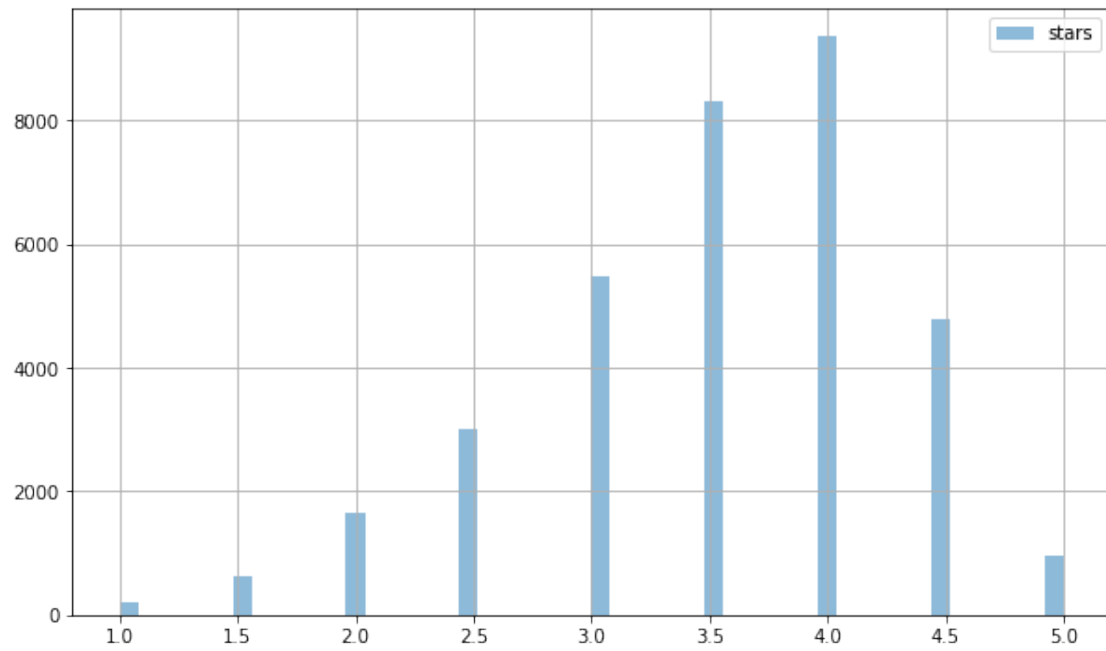
column: review_count



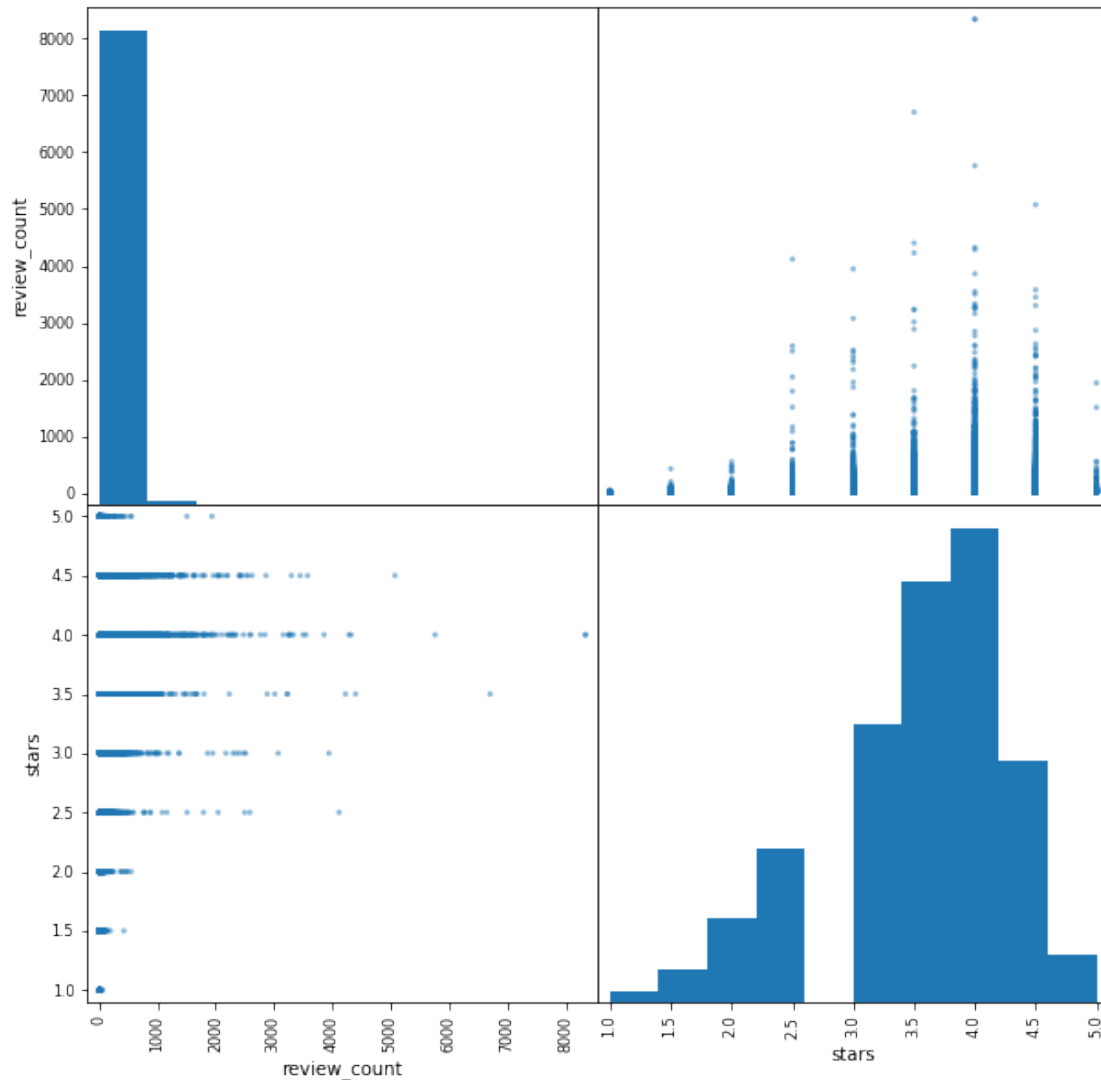
column: stars

Histogram for a column:

column: stars



Use `scatter_matrix` to see the histograms and also to see if there are correlations between the columns:



Create a geoDataFrame of the restaurant data and use it to
restaurant_gdf.head():

	address \	attributes	business_id \
1	30 Eglinton Avenue W		
2	10110 Johnston Rd, Ste 15		
11	2450 E Indian School Rd		
13	5981 Andrews Rd		
25	6055 E Lake Mead Blvd		

	attributes	business_id \
1	{'RestaurantsReservations': 'True', 'GoodForMe...	QXAEGFB4oINsVuTFxEYKFQ
2	{'GoodForKids': 'True', 'NoiseLevel': "u'avera...	gnKjwL_1w79qoiV3IC_xQQ
11	{'RestaurantsTakeOut': 'True', 'BusinessParkin...	1Dfx3zM-rW4n-31KeC8sJg
13	{'RestaurantsPriceRange2': '2', 'BusinessAccep...	fweCYi8FmbJXHCqLnwuk8w

```
25 {'BikeParking': 'True', 'BusinessParking': '{"'... tstimHoMcYbkSC4eBA1wEg
```

```

categories city \
1 Specialty Food, Restaurants, Dim Sum, Imported... Mississauga
2 Sushi Bars, Restaurants, Japanese Charlotte
11 Restaurants, Breakfast & Brunch, Mexican, Taco... Phoenix
13 Italian, Restaurants, Pizza, Chicken Wings Mentor-on-the-Lake
25 Mexican, Restaurants, Patisserie/Cake Shop, Fo... Las Vegas
```

```

hours latitude longitude \
1 {'Monday': '9:0-0:0', 'Tuesday': '9:0-0:0', 'W... 43.605499 -79.652289
2 {'Monday': '17:30-21:30', 'Wednesday': '17:30-... 35.092564 -80.859132
11 {'Monday': '7:0-0:0', 'Tuesday': '7:0-0:0', 'W... 33.495194 -112.028588
13 {'Monday': '10:0-0:0', 'Tuesday': '10:0-0:0', ... 41.708520 -81.359556
25 {'Monday': '11:0-21:0', 'Tuesday': '10:0-21:0'... 36.195615 -115.040529
```

```

name postal_code review_count stars \
1 Emerald Chinese Restaurant L5R 3E7 128 2.5
2 Musashi Japanese Restaurant 28210 170 4.0
11 Taco Bell 85016 18 3.0
13 Marco's Pizza 44060 16 4.0
25 Maria's Mexican Restaurant & Bakery 89156 184 4.5
```

```

state geometry
1 ON POINT (-79.65228890899999 43.6054989743)
2 NC POINT (-80.85913199999999 35.092564)
11 AZ POINT (-112.0285876 33.4951941)
13 OH POINT (-81.35955600000001 41.70852)
25 NV POINT (-115.0405289 36.1956146)
```

```
restaurant_gdf.head():
```

```

address \
1 30 Eglinton Avenue W
2 10110 Johnston Rd, Ste 15
11 2450 E Indian School Rd
13 5981 Andrews Rd
25 6055 E Lake Mead Blvd
```

```

attributes business_id \
1 {'RestaurantsReservations': 'True', 'GoodForMe... QXAEGB4oINsVuTFxEYKFQ
2 {'GoodForKids': 'True', 'NoiseLevel': "u'avera... gnKjwL_1w79qoiV3IC_xQQ
11 {'RestaurantsTakeOut': 'True', 'BusinessParkin... 1Dfx3zM-rW4n-31KeC8sJg
13 {'RestaurantsPriceRange2': '2', 'BusinessAccep... fweCYi8FmbJXHCqLnwuk8w
25 {'BikeParking': 'True', 'BusinessParking': '{"'... tstimHoMcYbkSC4eBA1wEg
```

```

categories city \
1 Specialty Food, Restaurants, Dim Sum, Imported... Mississauga
2 Sushi Bars, Restaurants, Japanese Charlotte
```

11	Restaurants, Breakfast & Brunch, Mexican, Taco...	Phoenix
13	Italian, Restaurants, Pizza, Chicken Wings	Mentor-on-the-Lake
25	Mexican, Restaurants, Patisserie/Cake Shop, Fo...	Las Vegas

	hours \
1	{'Monday': '9:0-0:0', 'Tuesday': '9:0-0:0', 'W...
2	{'Monday': '17:30-21:30', 'Wednesday': '17:30-...
11	{'Monday': '7:0-0:0', 'Tuesday': '7:0-0:0', 'W...
13	{'Monday': '10:0-0:0', 'Tuesday': '10:0-0:0', ...
25	{'Monday': '11:0-21:0', 'Tuesday': '10:0-21:0'...

	name	postal_code	review_count	stars \
1	Emerald Chinese Restaurant	L5R 3E7	128	2.5
2	Musashi Japanese Restaurant	28210	170	4.0
11	Taco Bell	85016	18	3.0
13	Marco's Pizza	44060	16	4.0
25	Maria's Mexican Restaurant & Bakery	89156	184	4.5

	state	geometry
1	ON	POINT (-79.65228890899999 43.6054989743)
2	NC	POINT (-80.85913199999999 35.092564)
11	AZ	POINT (-112.0285876 33.4951941)
13	OH	POINT (-81.35955600000001 41.70852)
25	NV	POINT (-115.0405289 36.1956146)

restaurant_df.head():

	address \
1	30 Eglinton Avenue W
2	10110 Johnston Rd, Ste 15
11	2450 E Indian School Rd
13	5981 Andrews Rd
25	6055 E Lake Mead Blvd

	attributes	business_id \
1	{'RestaurantsReservations': 'True', 'GoodForMe...	QXAEGB4oINsVuTFxEYKFQ
2	{'GoodForKids': 'True', 'NoiseLevel': 'u'avera...	gnKjwL_1w79qoiV3IC_xQQ
11	{'RestaurantsTakeOut': 'True', 'BusinessParkin...	1Dfx3zM-rW4n-31KeC8sJg
13	{'RestaurantsPriceRange2': '2', 'BusinessAccep...	fweCYi8FmbJXHCqLnwuk8w
25	{'BikeParking': 'True', 'BusinessParking': '{'...	tstimHoMcYbkSC4eBA1wEg

	categories	city \
1	Specialty Food, Restaurants, Dim Sum, Imported...	Mississauga
2	Sushi Bars, Restaurants, Japanese	Charlotte
11	Restaurants, Breakfast & Brunch, Mexican, Taco...	Phoenix
13	Italian, Restaurants, Pizza, Chicken Wings	Mentor-on-the-Lake
25	Mexican, Restaurants, Patisserie/Cake Shop, Fo...	Las Vegas

hours	latitude	longitude \
-------	----------	-------------

```

1  {'Monday': '9:0-0:0', 'Tuesday': '9:0-0:0', 'W... 43.605499 -79.652289
2  {'Monday': '17:30-21:30', 'Wednesday': '17:30-... 35.092564 -80.859132
11 {'Monday': '7:0-0:0', 'Tuesday': '7:0-0:0', 'W... 33.495194 -112.028588
13 {'Monday': '10:0-0:0', 'Tuesday': '10:0-0:0', ... 41.708520 -81.359556
25 {'Monday': '11:0-21:0', 'Tuesday': '10:0-21:0'... 36.195615 -115.040529

```

```

                                name postal_code review_count stars \
1          Emerald Chinese Restaurant      L5R 3E7          128    2.5
2          Musashi Japanese Restaurant      28210           170    4.0
11         Taco Bell                        85016            18    3.0
13         Marco's Pizza                    44060            16    4.0
25 Maria's Mexican Restaurant & Bakery      89156           184    4.5

```

```

state geometry
1  ON POINT (-79.65228890899999 43.6054989743)
2  NC POINT (-80.85913199999999 35.092564)
11 AZ POINT (-112.0285876 33.4951941)
13 OH POINT (-81.35955600000001 41.70852)
25 NV POINT (-115.0405289 36.1956146)

```

```
restaurant_df.head():
```

```

address \
1      30 Eglinton Avenue W
2  10110 Johnston Rd, Ste 15
11  2450 E Indian School Rd
13      5981 Andrews Rd
25  6055 E Lake Mead Blvd

```

```

attributes business_id \
1  {'RestaurantsReservations': 'True', 'GoodForMe... QXAEGFB4oINsVuTFxEYKFQ
2  {'GoodForKids': 'True', 'NoiseLevel': "u'avera... gnKjwL_1w79qoiV3IC_xQQ
11 {'RestaurantsTakeOut': 'True', 'BusinessParkin... 1Dfx3zM-rW4n-31KeC8sJg
13 {'RestaurantsPriceRange2': '2', 'BusinessAccep... fweCYi8FmbJXHCqLnwuk8w
25 {'BikeParking': 'True', 'BusinessParking': "{... tstimHoMcYbkSC4eBA1wEg

```

```

categories city \
1  Specialty Food, Restaurants, Dim Sum, Imported... Mississauga
2          Sushi Bars, Restaurants, Japanese      Charlotte
11 Restaurants, Breakfast & Brunch, Mexican, Taco... Phoenix
13      Italian, Restaurants, Pizza, Chicken Wings Mentor-on-the-Lake
25 Mexican, Restaurants, Patisserie/Cake Shop, Fo... Las Vegas

```

```

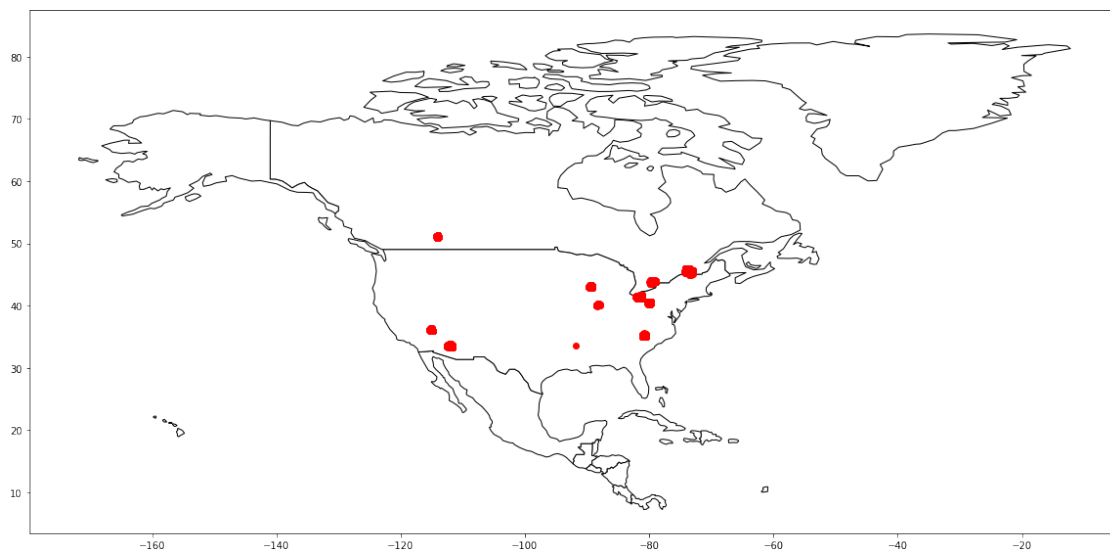
hours latitude longitude \
1  {'Monday': '9:0-0:0', 'Tuesday': '9:0-0:0', 'W... 43.605499 -79.652289
2  {'Monday': '17:30-21:30', 'Wednesday': '17:30-... 35.092564 -80.859132
11 {'Monday': '7:0-0:0', 'Tuesday': '7:0-0:0', 'W... 33.495194 -112.028588
13 {'Monday': '10:0-0:0', 'Tuesday': '10:0-0:0', ... 41.708520 -81.359556
25 {'Monday': '11:0-21:0', 'Tuesday': '10:0-21:0'... 36.195615 -115.040529

```

	name	postal_code	review_count	stars	state
1	Emerald Chinese Restaurant	L5R 3E7	128	2.5	ON
2	Musashi Japanese Restaurant	28210	170	4.0	NC
11	Taco Bell	85016	18	3.0	AZ
13	Marco's Pizza	44060	16	4.0	OH
25	Maria's Mexican Restaurant & Bakery	89156	184	4.5	NV

From now on will deal with the restaurant data only in the form of the geoDataFrame and not the original DataFrame

Plot the geographic distribution of the restaurants:



It can be seen that restaurant data only exist for the continental USA (i.e., no Alaska or Hawaii) and a few bordering provinces of Canada
Thus restrict the map to get a better picture

usa1.head():

	GID_0	NAME_0	GID_1	NAME_1	VARNAME_1	NL_NAME_1	TYPE_1	\
0	USA	United States	USA.1_1	Alabama	AL Ala.	None	State	
1	USA	United States	USA.2_1	Alaska	AK Alaska	None	State	
2	USA	United States	USA.3_1	Arizona	AZ Ariz.	None	State	
3	USA	United States	USA.4_1	Arkansas	AR Ark.	None	State	
4	USA	United States	USA.5_1	California	CA Calif.	None	State	

	ENGTYPE_1	CC_1	HASC_1	geometry
0	State	None	US.AL	(POLYGON ((-87.97000122 30.67666626, -87.96971...
1	State	None	US.AK	(POLYGON ((-179.13146973 51.20972061, -179.132...

```

2      State  None  US.AZ  POLYGON ((-110.82376099 31.33125305, -110.8260...
3      State  None  US.AR  POLYGON ((-92.72460937 33.01473999, -92.861946...
4      State  None  US.CA  (POLYGON ((-118.58975983 33.0386467, -118.5897...

```

usal:

```

      state_or_province      geometry
0      Alabama  (POLYGON ((-87.97000122 30.67666626, -87.96971...
2      Arizona  POLYGON ((-110.82376099 31.33125305, -110.8260...
3      Arkansas POLYGON ((-92.72460937 33.01473999, -92.861946...
4      California (POLYGON ((-118.58975983 33.0386467, -118.5897...
5      Colorado POLYGON ((-102.04303741 37.64289093, -102.0425...
6      Connecticut (POLYGON ((-73.62305449999999 40.98194504, -73...
7      Delaware  (POLYGON ((-75.11148833999999 38.80128479, -75...
8  District of Columbia POLYGON ((-77.00850677 38.96955872, -76.988487...
9      Florida  (POLYGON ((-81.92485809 24.53307915, -81.92485...
10     Georgia  (POLYGON ((-81.45249939 31.01777649, -81.45138...
12     Idaho    POLYGON ((-111.04750061 42.51359177, -111.0476...
13     Illinois POLYGON ((-89.31046295 37.00680923, -89.314926...
14     Indiana  POLYGON ((-85.79228972999999 38.28736877, -85...
15     Iowa     POLYGON ((-91.11219025 40.69557953, -91.113037...
16     Kansas   POLYGON ((-95.96520996 37.00035095, -95.974426...
17     Kentucky (POLYGON ((-89.54174805 36.50033951, -89.54557...
18     Louisiana (POLYGON ((-89.37555695 28.97138977, -89.375 2...
19     Maine    (POLYGON ((-70.41523743 43.3603363, -70.415519...
20     Maryland (POLYGON ((-75.90416718 37.89972305, -75.90305...
21     Massachusetts (POLYGON ((-70.81027985 41.26166534, -70.80999...
22     Michigan (POLYGON ((-83.45072937 41.73384094, -83.46109...
23     Minnesota POLYGON ((-91.85788727000001 44.19422913, -91...
24     Mississippi (POLYGON ((-88.50527954 30.21609116, -88.50499...
25     Missouri  POLYGON ((-94.61815643 36.49837112, -94.618469...
26     Montana  POLYGON ((-108.24507141 45.00112915, -108.2490...
27     Nebraska  POLYGON ((-95.78649901999999 40.00149918, -95...
28     Nevada   POLYGON ((-114.61759186 35.08576202, -114.6152...
29     New Hampshire (POLYGON ((-70.60446930000001 43.00661469, -70...
30     New Jersey (POLYGON ((-74.74200439000001 39.14696121, -74...
31     New Mexico POLYGON ((-104.02654266 32.00101852, -104.0281...
32     New York  (POLYGON ((-74.20204163 40.57395935, -74.20243...
33     North Carolina (POLYGON ((-78.54416655999999 33.84166718, -78...
34     North Dakota POLYGON ((-102.99555206 45.94400024, -104.0442...
35     Ohio      POLYGON ((-82.66564941 38.50551987, -82.674758...
36     Oklahoma  POLYGON ((-97.48200226 33.91559982, -97.489692...
37     Oregon    (POLYGON ((-124.40777588 42.59166718, -124.407...
38     Pennsylvania POLYGON ((-80.42436981 39.72135162, -80.517051...
39     Rhode Island (POLYGON ((-71.57733917 41.23161316, -71.57733...
40     South Carolina (POLYGON ((-80.70610809 32.39250183, -80.70666...
41     South Dakota POLYGON ((-102.99900818 42.99967957, -103.0522...
42     Tennessee POLYGON ((-84.29194640999999 35.20671082, -84...
43     Texas     (POLYGON ((-97.21569823999999 25.97013855, -97...
44     Utah      POLYGON ((-109.0602417 38.49905014, -109.06024...

```



```

45         Vermont POLYGON ((-72.45662689 42.72761154, -72.609992...
46         Virginia (POLYGON ((-75.90833282 36.5597229, -75.907775...
47         Washington (POLYGON ((-122.32476807 45.56042099, -122.322...
48         West Virginia POLYGON ((-80.30742644999999 37.68952942, -80...
49         Wisconsin POLYGON ((-89.92558289 42.50751114, -89.957298...
50         Wyoming POLYGON ((-110.04850006 40.99882889, -110.0572...

```

```
can1.head(3):
```

```

  GID_0 NAME_0  GID_1      NAME_1 \
0   CAN  Canada CAN.1_1      Alberta
1   CAN  Canada CAN.2_1 British Columbia
2   CAN  Canada CAN.3_1      Manitoba

```

```

          VARNAME_1 NL_NAME_1  TYPE_1 ENGTYPE_1 CC_1 \
0              None      None Province Province  48
1  Colombie britannique|New Caledonia      None Province Province  59
2              None      None Province Province  46

```

```

HASC_1      geometry
0  CA.AB POLYGON ((-111.95340729 49.00004959, -111.9608...
1  CA.BC (POLYGON ((-123.54055786 48.31833267, -123.540...
2  CA.MB (POLYGON ((-90.38500214 57.18527603, -90.38500...

```

```
can1:
```

```

  state_or_province      geometry
0      Alberta POLYGON ((-111.95340729 49.00004959, -111.9608...
1  British Columbia (POLYGON ((-123.54055786 48.31833267, -123.540...
2      Manitoba (POLYGON ((-90.38500214 57.18527603, -90.38500...
3  New Brunswick (POLYGON ((-66.84994507 44.48387909, -66.84999...
8      Ontario (POLYGON ((-74.48135376 45.08594894, -74.48263...
10     QuÃbec (POLYGON ((-74.64563751 44.99940491, -74.64547...
11    Saskatchewan POLYGON ((-108.01008606 49.00007629, -108.0210...

```

```
usa1_can1:
```

```

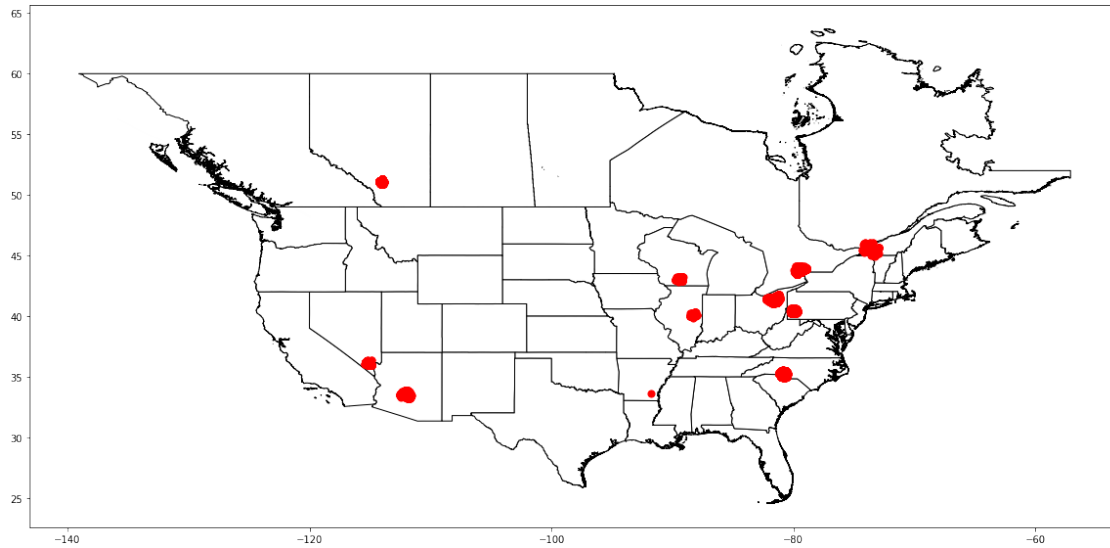
  state_or_province      geometry
0      Alabama (POLYGON ((-87.97000122 30.67666626, -87.96971...
2      Arizona POLYGON ((-110.82376099 31.33125305, -110.8260...
3      Arkansas POLYGON ((-92.72460937 33.01473999, -92.861946...
4      California (POLYGON ((-118.58975983 33.0386467, -118.5897...
5      Colorado POLYGON ((-102.04303741 37.64289093, -102.0425...
6      Connecticut (POLYGON ((-73.62305449999999 40.98194504, -73...
7      Delaware (POLYGON ((-75.11148833999999 38.80128479, -75...
8  District of Columbia POLYGON ((-77.00850677 38.96955872, -76.988487...
9      Florida (POLYGON ((-81.92485809 24.53307915, -81.92485...
10     Georgia (POLYGON ((-81.45249939 31.01777649, -81.45138...
12     Idaho POLYGON ((-111.04750061 42.51359177, -111.0476...
13     Illinois POLYGON ((-89.31046295 37.00680923, -89.314926...
14     Indiana POLYGON ((-85.79228972999999 38.28736877, -85...
15     Iowa POLYGON ((-91.11219025 40.69557953, -91.113037...
16     Kansas POLYGON ((-95.96520996 37.00035095, -95.974426...
17     Kentucky (POLYGON ((-89.54174805 36.50033951, -89.54557...

```

18	Louisiana	(POLYGON ((-89.37555695 28.97138977, -89.375 2...
19	Maine	(POLYGON ((-70.41523743 43.3603363, -70.415519...
20	Maryland	(POLYGON ((-75.90416718 37.89972305, -75.90305...
21	Massachusetts	(POLYGON ((-70.81027985 41.26166534, -70.80999...
22	Michigan	(POLYGON ((-83.45072937 41.73384094, -83.46109...
23	Minnesota	POLYGON ((-91.85788727000001 44.19422913, -91...
24	Mississippi	(POLYGON ((-88.50527954 30.21609116, -88.50499...
25	Missouri	POLYGON ((-94.61815643 36.49837112, -94.618469...
26	Montana	POLYGON ((-108.24507141 45.00112915, -108.2490...
27	Nebraska	POLYGON ((-95.78649901999999 40.00149918, -95...
28	Nevada	POLYGON ((-114.61759186 35.08576202, -114.6152...
29	New Hampshire	(POLYGON ((-70.60446930000001 43.00661469, -70...
30	New Jersey	(POLYGON ((-74.74200439000001 39.14696121, -74...
31	New Mexico	POLYGON ((-104.02654266 32.00101852, -104.0281...
32	New York	(POLYGON ((-74.20204163 40.57395935, -74.20243...
33	North Carolina	(POLYGON ((-78.54416655999999 33.84166718, -78...
34	North Dakota	POLYGON ((-102.99555206 45.94400024, -104.0442...
35	Ohio	POLYGON ((-82.66564941 38.50551987, -82.674758...
36	Oklahoma	POLYGON ((-97.48200226 33.91559982, -97.489692...
37	Oregon	(POLYGON ((-124.40777588 42.59166718, -124.407...
38	Pennsylvania	POLYGON ((-80.42436981 39.72135162, -80.517051...
39	Rhode Island	(POLYGON ((-71.57733917 41.23161316, -71.57733...
40	South Carolina	(POLYGON ((-80.70610809 32.39250183, -80.70666...
41	South Dakota	POLYGON ((-102.99900818 42.99967957, -103.0522...
42	Tennessee	POLYGON ((-84.29194640999999 35.20671082, -84...
43	Texas	(POLYGON ((-97.21569823999999 25.97013855, -97...
44	Utah	POLYGON ((-109.0602417 38.49905014, -109.06024...
45	Vermont	POLYGON ((-72.45662689 42.72761154, -72.609992...
46	Virginia	(POLYGON ((-75.90833282 36.5597229, -75.907775...
47	Washington	(POLYGON ((-122.32476807 45.56042099, -122.322...
48	West Virginia	POLYGON ((-80.30742644999999 37.68952942, -80...
49	Wisconsin	POLYGON ((-89.92558289 42.50751114, -89.957298...
50	Wyoming	POLYGON ((-110.04850006 40.99882889, -110.0572...
0	Alberta	POLYGON ((-111.95340729 49.00004959, -111.9608...
1	British Columbia	(POLYGON ((-123.54055786 48.31833267, -123.540...
2	Manitoba	(POLYGON ((-90.38500214 57.18527603, -90.38500...
3	New Brunswick	(POLYGON ((-66.84994507 44.48387909, -66.84999...
8	Ontario	(POLYGON ((-74.48135376 45.08594894, -74.48263...
10	QuÃbec	(POLYGON ((-74.64563751 44.99940491, -74.64547...
11	Saskatchewan	POLYGON ((-108.01008606 49.00007629, -108.0210...

Construct a dictionary to count number restaurants in each unique state/province

```
{'ON': 8399, 'NC': 2593, 'AZ': 6472, 'OH': 3404, 'NV': 4295, 'PA': 2380, 'QC': 3357, 'AB': 1840, 'WI': 1061, 'IL': 368, 'NY': 11, 'SC': 203, 'NM': 1, 'VA': 1, 'NE': 1, 'WA': 1, 'XWY': 1, 'BC': 1, 'VT': 1, 'AR': 1, 'AL': 1}
```



---It can be seen that about 8k restaurants got very few reviews. Make a decision to recommend based on some thresholds which are:---
Also note that there are locations in 11 states or provinces. Try to retain at least 1 in each.

```
stars >= 4.5
review_count >= 100
exclude_states: []
```

Doing a cycle of applying thresholds

cycle: 1

Inspect the data before applying the thresholds:

```
df.shape:
(34392, 12)
```

```
df.info():
<class 'geopandas.geodataframe.GeoDataFrame'>
Int64Index: 34392 entries, 1 to 192603
Data columns (total 12 columns):
address      34392 non-null object
attributes   33237 non-null object
business_id  34392 non-null object
categories   34392 non-null object
city         34392 non-null object
```

```

hours          34392 non-null object
name           34392 non-null object
postal_code    34389 non-null object
review_count   34392 non-null int64
stars          34392 non-null float64
state          34392 non-null object
geometry       34392 non-null object
dtypes: float64(1), int64(1), object(10)
memory usage: 3.4+ MB
None

```

Apply the thresholds:

Re-inspect the data after applying the thresholds:

```

df.shape:
(1492, 12)

df.info():
<class 'geopandas.geodataframe.GeoDataFrame'>
Int64Index: 1492 entries, 25 to 192571
Data columns (total 12 columns):
address          1492 non-null object
attributes       1489 non-null object
business_id      1492 non-null object
categories       1492 non-null object
city             1492 non-null object
hours           1492 non-null object
name            1492 non-null object
postal_code      1492 non-null object
review_count     1492 non-null int64
stars            1492 non-null float64
state           1492 non-null object
geometry         1492 non-null object
dtypes: float64(1), int64(1), object(10)
memory usage: 151.5+ KB
None

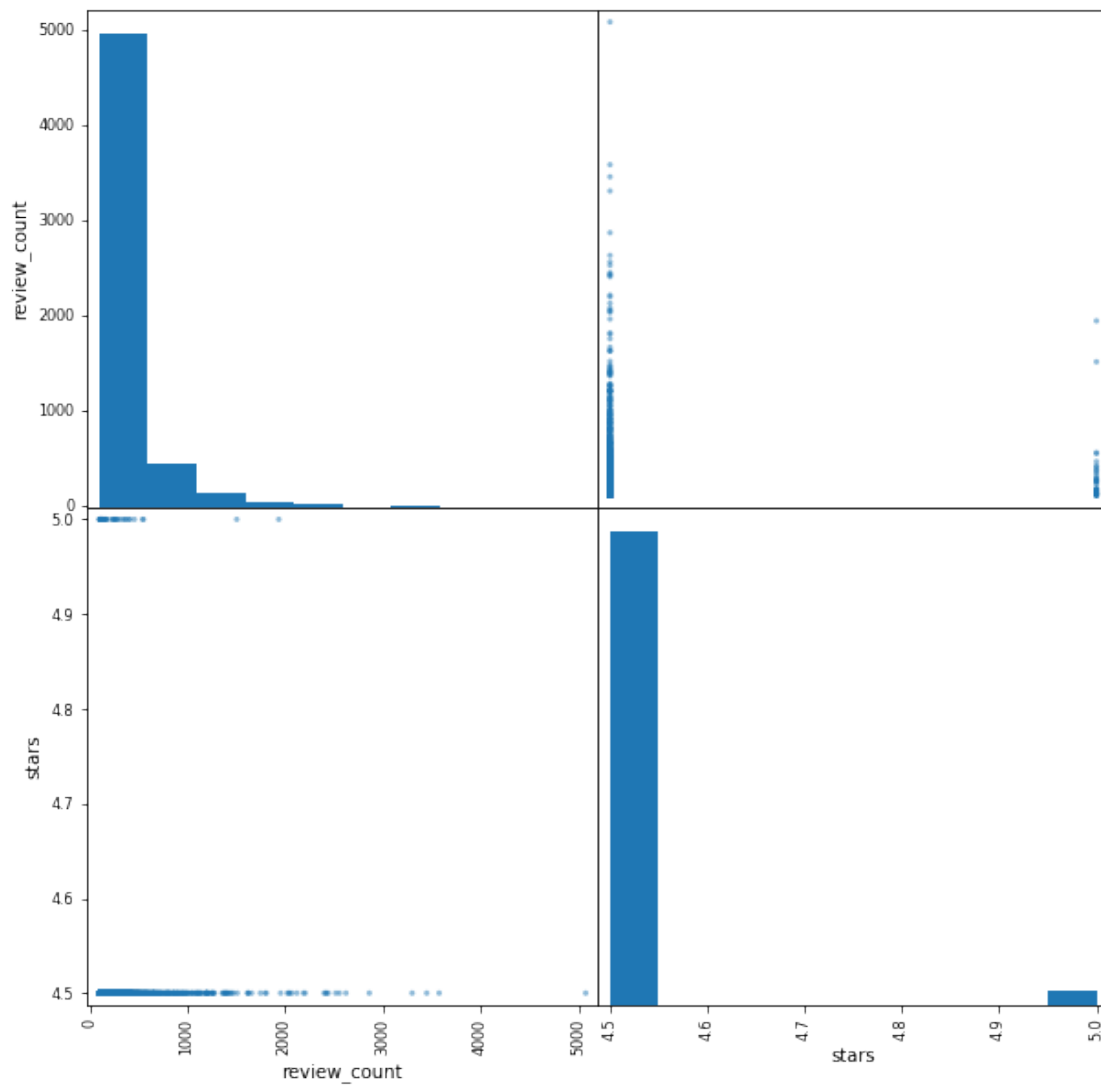
```

Statistics for some columns:

	review_count	stars
count	1492.000000	1492.000000
mean	348.947721	4.514410

std	389.965098	0.083679
min	100.000000	4.500000
25%	143.000000	4.500000
50%	222.500000	4.500000
75%	386.250000	4.500000
max	5075.000000	5.000000

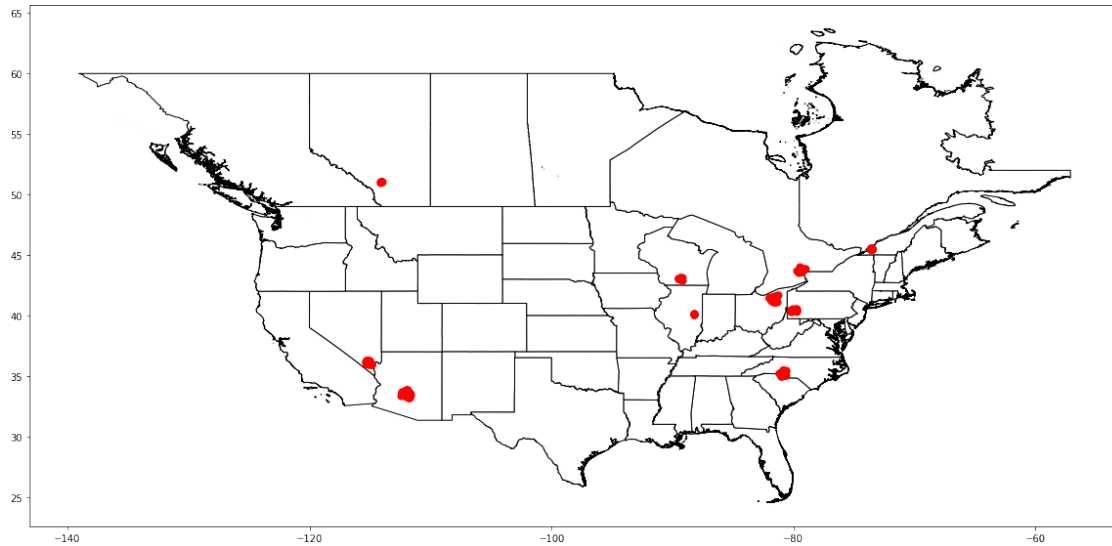
Use `scatter_matrix` to see the histograms and also to see if there are correlations between the columns:



Construct a dictionary to count number restaurants in each unique state/province

```
{'NV': 455, 'PA': 86, 'ON': 93, 'NC': 100, 'AZ': 560, 'QC': 53, 'SC': 7, 'OH': 88, 'WI': 31, 'AB': 12, 'IL': 7}
```

Geographical distribution:



```
stars >= 5.0
review_count >= 200
exclude_states: []
```

Doing a cycle of applying thresholds

cycle: 2

Inspect the data before applying the thresholds:

```
df.shape:
(1492, 12)
```

```
df.info():
<class 'geopandas.geodataframe.GeoDataFrame'>
Int64Index: 1492 entries, 25 to 192571
Data columns (total 12 columns):
address      1492 non-null object
attributes   1489 non-null object
business_id  1492 non-null object
categories   1492 non-null object
```

```

city          1492 non-null object
hours         1492 non-null object
name          1492 non-null object
postal_code   1492 non-null object
review_count  1492 non-null int64
stars         1492 non-null float64
state         1492 non-null object
geometry      1492 non-null object
dtypes: float64(1), int64(1), object(10)
memory usage: 151.5+ KB
None

```

Apply the thresholds:

Re-inspect the data after applying the thresholds:

```

df.shape:
(19, 12)

df.info():
<class 'geopandas.geodataframe.GeoDataFrame'>
Int64Index: 19 entries, 18690 to 163928
Data columns (total 12 columns):
address          19 non-null object
attributes       19 non-null object
business_id      19 non-null object
categories       19 non-null object
city            19 non-null object
hours           19 non-null object
name            19 non-null object
postal_code      19 non-null object
review_count     19 non-null int64
stars           19 non-null float64
state           19 non-null object
geometry         19 non-null object
dtypes: float64(1), int64(1), object(10)
memory usage: 1.9+ KB
None

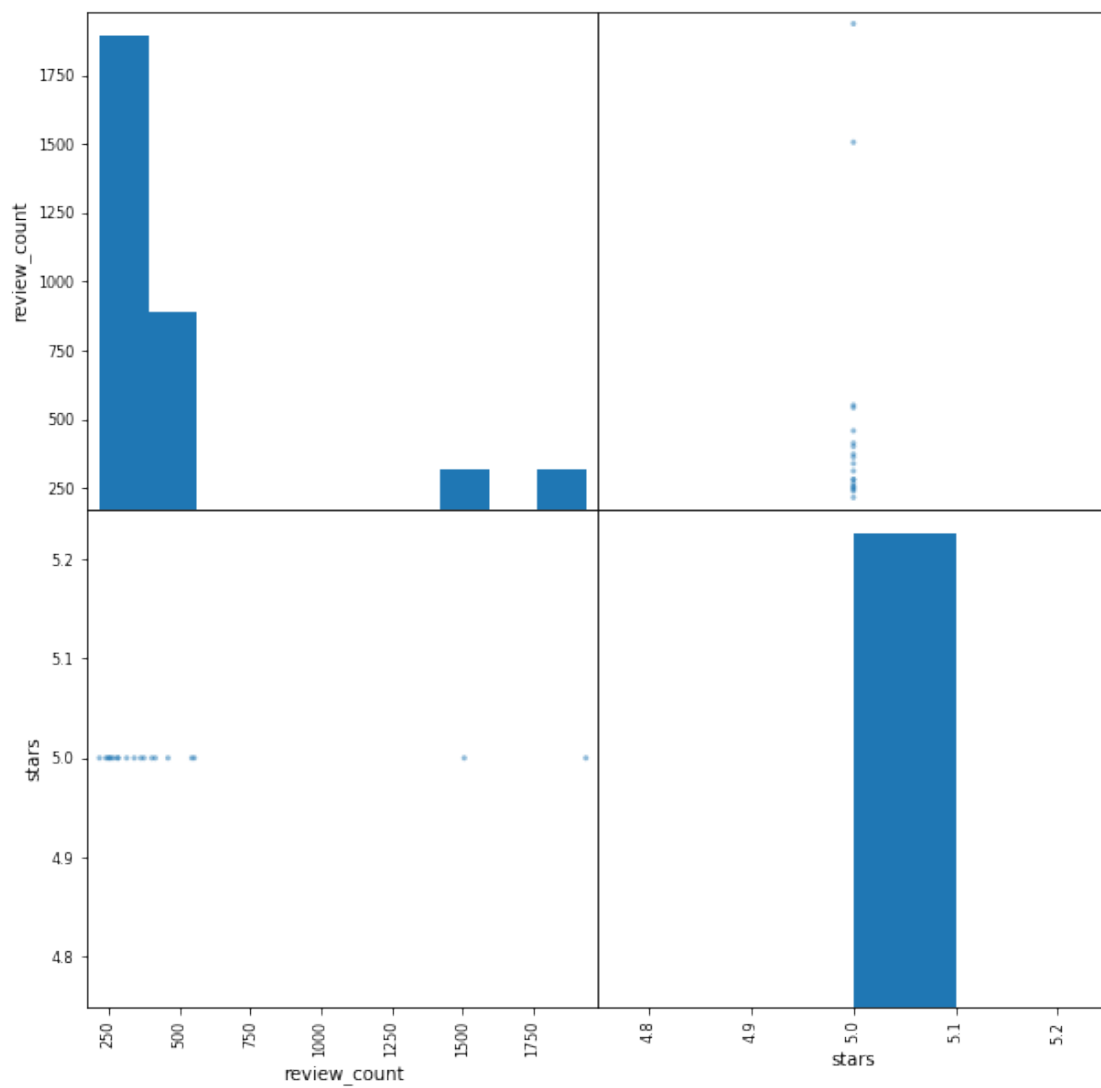
```

Statistics for some columns:

	review_count	stars
count	19.000000	19.0

mean	486.684211	5.0
std	451.559280	0.0
min	217.000000	5.0
25%	262.000000	5.0
50%	340.000000	5.0
75%	436.500000	5.0
max	1936.000000	5.0

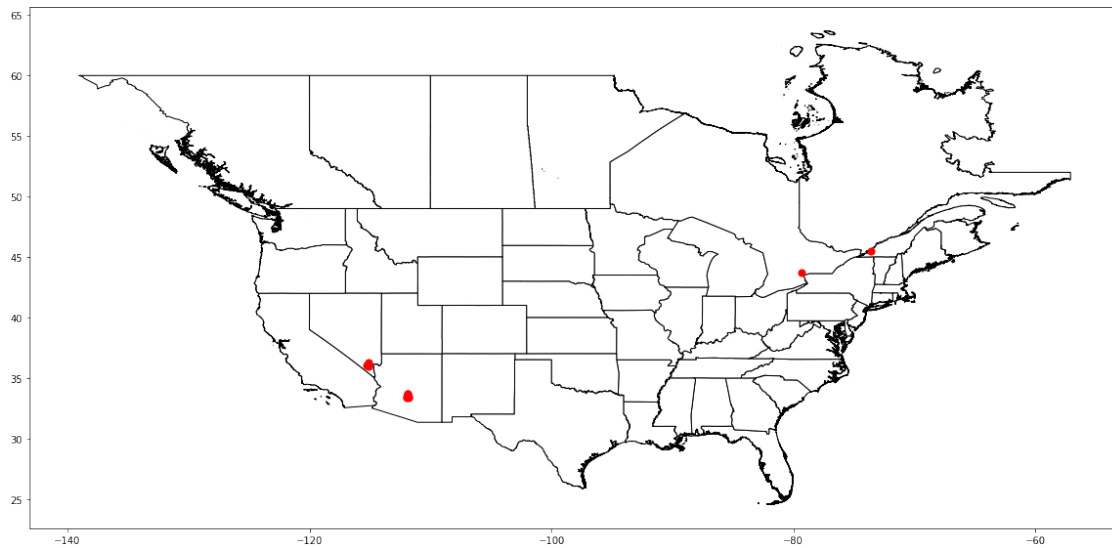
Use `scatter_matrix` to see the histograms and also to see if there are correlations between the columns:



Construct a dictionary to count number restaurants in each unique state/province

```
{'AZ': 9, 'NV': 8, 'QC': 1, 'ON': 1}
```

Geographical distribution:



Thus by tightening thresholds we reduced the data to only 19 restaurants in 4 states/provinces:

```
{'AZ': 9, 'NV': 8, 'QC': 1, 'ON': 1}
```

The user can then select the appropriate restaurant based on cuisine, etc.

restaurant_gdf:

	address \
18690	4301 E University Dr
38760	1615 S Las Vegas Blvd
44484	3430 E Tropicana Ave, Ste 32
50434	18221 N Pima Rd, Ste 100 & 105
52150	1201 Avenue Van Horne
63348	1262 Don Mills Road
92813	7380 S Rainbow Blvd, Ste 101
114022	15040 N Northsight Blvd, Ste 104
114957	10895 S Eastern Ave, Ste 100
119721	7605 E Pinnacle Peak Rd, Ste 3
121204	3957 S Maryland Pkwy
122966	655 W Craig Rd, Ste 118

123429	218 W Main St
125079	3635 Las Vegas Blvd S
150805	1166 South Gilbert Rd, Ste 101
151343	3126 S Mill Ave
157217	8015 E Roosevelt St
159797	4280 Drinkwater Blvd, Ste 200
163928	3899 Spring Mountain Rd

	attributes \
18690	{'BusinessAcceptsCreditCards': 'True', 'Restau...
38760	{'GoodForKids': 'True', 'RestaurantsTakeOut': ...
44484	{'RestaurantsReservations': 'False', 'Business...
50434	{'NoiseLevel': "u'average'", 'BusinessParking'...
52150	{'RestaurantsReservations': 'True', 'Caters': ...
63348	{'Ambience': "{ 'romantic': False, 'intimate': ...
92813	{'Alcohol': "'none'", 'BusinessAcceptsCreditCa...
114022	{'RestaurantsTableService': 'True', 'BusinessA...
114957	{'OutdoorSeating': 'True', 'Alcohol': "u'none'..."
119721	{'Ambience': "{ 'touristy': False, 'hipster': F...
121204	{'HasTV': 'True', 'WiFi': "u'free'", 'Business...
122966	{'RestaurantsAttire': "u'casual'", 'Restaurant...
123429	{'RestaurantsGoodForGroups': 'False', 'GoodFor...
125079	{'WheelchairAccessible': 'True', 'BusinessAcce...
150805	{'RestaurantsDelivery': 'False', 'BusinessPark...
151343	{'BusinessAcceptsCreditCards': 'True', 'WiFi':...
157217	{'GoodForMeal': "{ 'dessert': False, 'latenight...
159797	{'RestaurantsTableService': 'False', 'Wheelcha...
163928	{'BusinessParking': "{ 'garage': False, 'street...

	business_id \
18690	Xg5qEQiB-7L6kGJ5F4K3bQ
38760	G4hjhtA_wQ-tSOGpgGLDjw
44484	8fFTJBhORB2EKG53ibiBKw
50434	EpPOZAG0u7qHP-jv5HjFUg
52150	ARfXk9Sa2vrK36fAGvdmWQ
63348	g6AFW-zY0wDvBl9U82g4zg
92813	IhNASEZ3XnBHmuuVnWdIwA
114022	1qkKfqh08z2XMzLLDFE96Q
114957	q2GzUNQj998GSC8IhkN9hg
119721	kR4NBNFf5SLpeAtZ-uPoPg
121204	cePE3rCuUOVSCCAHSjWxoQ
122966	ewmTwsZqCHH2gvCeDKzOdw
123429	SSCH4Z2gw-hh2KZy7aH4qw
125079	iBPyahdJRP5y0t25fF2W9w
150805	52yWGkwnrQXIjvuMjYxsiA
151343	PhMfnMldVKXo7HqHw6DSrg
157217	2KadiQLTen9_mN_dXgKR5A
159797	xCL38K0oPgK3ydzg4CrvKg

163928 3pSUR_cdrphur06m1HMP9A

	categories	city \
18690	Barbeque, Restaurants	Phoenix
38760	Dive Bars, Restaurants, Bars, Tacos, Fast Food...	Las Vegas
44484	Cafes, Breakfast & Brunch, Restaurants	Las Vegas
50434	Poke, Food, Hawaiian, Sushi Bars, Restaurants	Scottsdale
52150	Restaurants, Middle Eastern, Syrian	Montréal
63348	Restaurants, Italian, Cafes	Toronto
92813	Tea Rooms, Desserts, Cafes, Restaurants, Food,...	Las Vegas
114022	Sushi Bars, Japanese, Restaurants	Scottsdale
114957	Mediterranean, Juice Bars & Smoothies, Food, C...	Henderson
119721	Coffee & Tea, Ice Cream & Frozen Yogurt, Itali...	Scottsdale
121204	Fast Food, Restaurants, Sandwiches, American (...)	Las Vegas
122966	Restaurants, Food, Poke, Hawaiian, Specialty F...	North Las Vegas
123429	American (Traditional), Breakfast & Brunch, Sa...	Mesa
125079	Hotels, Food, Event Planning & Services, Hotel...	Las Vegas
150805	Restaurants, Caterers, Cuban, Sandwiches, Even...	Gilbert
151343	Coffee & Tea, Juice Bars & Smoothies, Taiwanese...	Tempe
157217	Shopping, Wholesale Stores, Seafood, Poke, Spe...	Scottsdale
159797	Vegetarian, Latin American, Vegan, Restaurants...	Scottsdale
163928	Bars, Korean, Restaurants, Asian Fusion, Cockt...	Las Vegas

	hours \
18690	{ 'Monday': '0:0-0:0', 'Tuesday': '11:0-16:0', ...
38760	{ 'Monday': '0:0-0:0', 'Tuesday': '11:0-21:0', ...
44484	{ 'Wednesday': '8:0-14:0', 'Thursday': '8:0-14:...
50434	{ 'Monday': '0:0-0:0', 'Tuesday': '11:0-21:0', ...
52150	{ 'Monday': '17:0-22:0', 'Tuesday': '17:0-22:0'...
63348	{ 'Monday': '7:30-18:0', 'Tuesday': '7:30-18:0'...
92813	{ 'Monday': '0:0-0:0', 'Tuesday': '11:0-22:0', ...
114022	{ 'Monday': '0:0-0:0', 'Tuesday': '16:0-21:0', ...
114957	{ 'Monday': '0:0-0:0', 'Tuesday': '10:0-22:0', ...
119721	{ 'Tuesday': '11:0-22:0', 'Wednesday': '11:0-22...
121204	{ 'Monday': '0:0-0:0', 'Tuesday': '11:0-20:0', ...
122966	{ 'Monday': '10:0-18:0', 'Tuesday': '10:0-18:0'...
123429	{ 'Monday': '0:0-0:0', 'Tuesday': '8:0-21:0', '...
125079	{ 'Monday': '0:0-0:0', 'Tuesday': '0:0-0:0', 'W...
150805	{ 'Monday': '0:0-0:0', 'Tuesday': '11:0-17:0', ...
151343	{ 'Monday': '11:0-22:0', 'Tuesday': '11:0-22:0'...
157217	{ 'Monday': '0:0-0:0', 'Tuesday': '10:0-19:0', ...
159797	{ 'Monday': '0:0-0:0', 'Tuesday': '11:0-20:0', ...
163928	{ 'Monday': '18:0-5:0', 'Tuesday': '18:0-5:0', ...

	name	postal_code	review_count	stars	state \
18690	Little Miss BBQ	85034	1936	5.0	AZ
38760	Bajamar Seafood & Tacos	89104	282	5.0	NV
44484	Zenaida's Cafe	89121	374	5.0	NV

50434	Koi Poke	85255	247	5.0	AZ
52150	Damas	H2V 1K4	217	5.0	QC
63348	Baretto Caffè	M3B 2W7	267	5.0	ON
92813	Brew Tea Bar	89139	1506	5.0	NV
114022	Kodo Sushi Sake	85260	414	5.0	AZ
114957	HUMMUS	89052	459	5.0	NV
119721	That's Amore Gelato	85255	282	5.0	AZ
121204	Karved	89119	313	5.0	NV
122966	Poke Express	89032	543	5.0	NV
123429	Worth Takeaway	85201	552	5.0	AZ
125079	Lip Smacking Foodie Tours	89109	257	5.0	NV
150805	Not Your Typical Deli	85296	402	5.0	AZ
151343	DaYung's Tea	85282	240	5.0	AZ
157217	Chula Seafood	85257	253	5.0	AZ
159797	Simon's Hot Dogs	85251	340	5.0	AZ
163928	J Karaoke Bar	89102	363	5.0	NV

geometry

18690	POINT (-111.9893442839 33.4218771341)
38760	POINT (-115.1518118808 36.1517421042)
44484	POINT (-115.1003590417 36.1017406138)
50434	POINT (-111.886156 33.65154260000001)
52150	POINT (-73.61321170000001 45.5225641)
63348	POINT (-79.34646770000001 43.7447025)
92813	POINT (-115.2424074 36.0542269)
114022	POINT (-111.8976141 33.622219)
114957	POINT (-115.1016894 35.9936882)
119721	POINT (-111.9162816909 33.6969313374)
121204	POINT (-115.138401 36.11709099999999)
122966	POINT (-115.150921382 36.2382459749)
123429	POINT (-111.8374321 33.4154613)
125079	POINT (-115.172678268 36.1145367627)
150805	POINT (-111.794915 33.32878)
151343	POINT (-111.940576007 33.3948182512)
157217	POINT (-111.90824 33.457762)
159797	POINT (-111.923117 33.4978997)
163928	POINT (-115.1917155 36.1261962)