Notebook

February 24, 2020

0.0.1 Question 1a

Let's look at the distribution of inspection scores. As we saw before when we called head on this data frame, inspection scores appear to be integer values. The discreteness of this variable means that we can use a barplot to visualize the distribution of the inspection score. Make a bar plot of the counts of the number of inspections receiving each score.

It should look like the image below. It does not need to look exactly the same (e.g., no grid), but make sure that all labels and axes are correct.
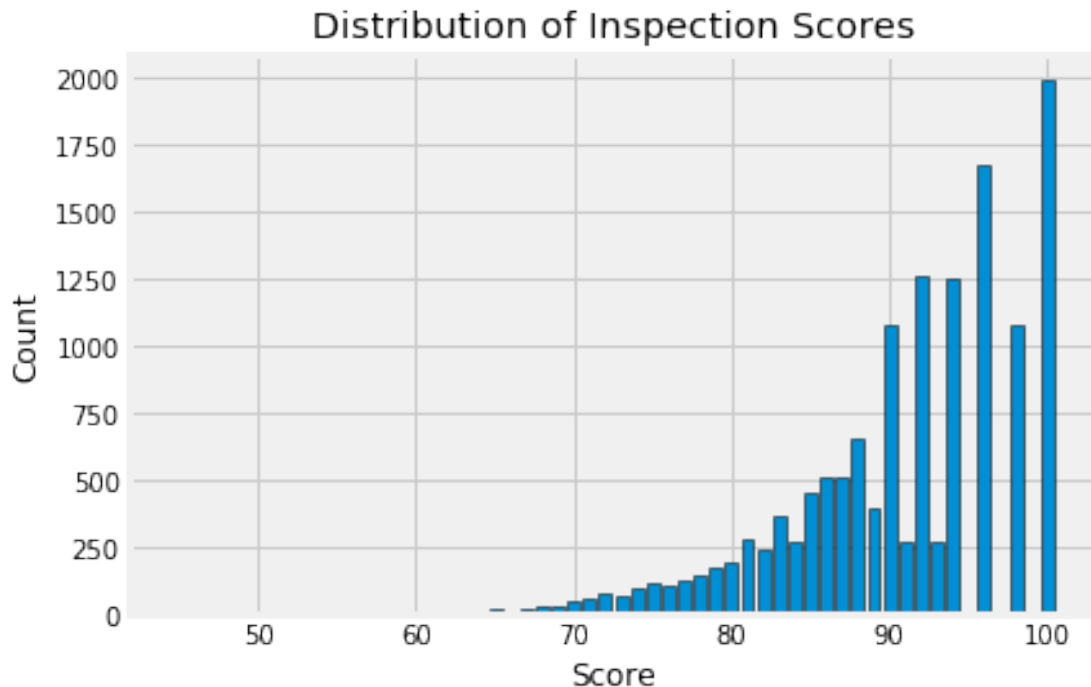

You might find this matplotlib.pyplot tutorial useful. Key syntax that you'll need:

```
plt.bar
plt.xlabel
plt.ylabel
plt.title
```

Note: If you want to use another plotting library for your plots (e.g. plotly, sns) you are welcome to use that library instead so long as it works on DataHub. If you use seaborn sns.countplot(), you may need to manually set what to display on xticks.

```
In [4]: plt.bar(ins.score.value_counts().drop(labels=-1).index, ins.score.value_counts().drop(labels=-1)
        plt.xlabel('Score')
        plt.ylabel('Count')
        plt.title('Distribution of Inspection Scores')
```

```
Out[4]: Text(0.5, 1.0, 'Distribution of Inspection Scores')
```

Distribution of Inspection Scores
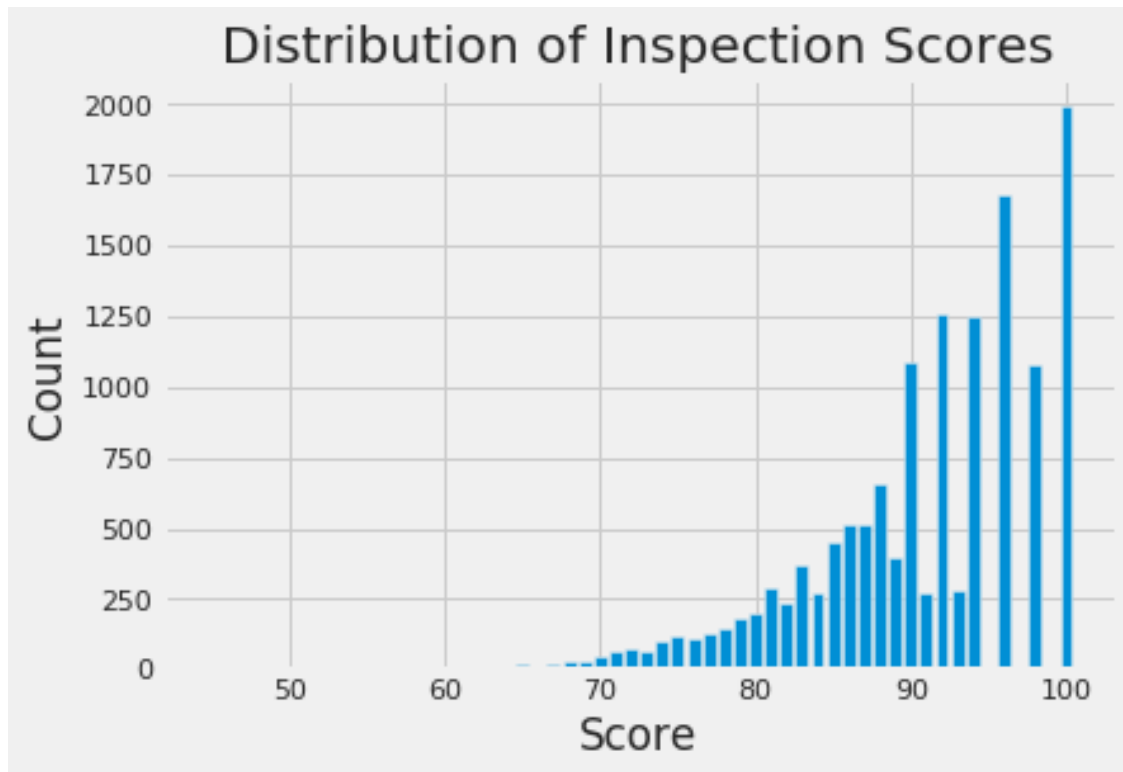
0.0.2 Question 1b

Describe the qualities of the distribution of the inspections scores based on your bar plot. Consider the mode(s), symmetry, tails, gaps, and anomalous values. Are there any unusual features of this distribution? What do your observations imply about the scores?

The most common scores are quite high, and no scores are given below 40 (barely any below 65). The scores are skewed right by quite a bit.

Use the cell above to identify the restaurant with the lowest inspection scores ever. Be sure to include the name of the restaurant as part of your answer in the cell below. You can also head to yelp.com and look up the reviews page for this restaurant. Feel free to add anything interesting you want to share.

The restaurant with the lowest inspection score is called Lollipot, located on 890 Taraval Street.

Now, create your scatter plot in the cell below. It does not need to look exactly the same (e.g., no grid) as the sample below, but make sure that all labels, axes and data itself are correct.
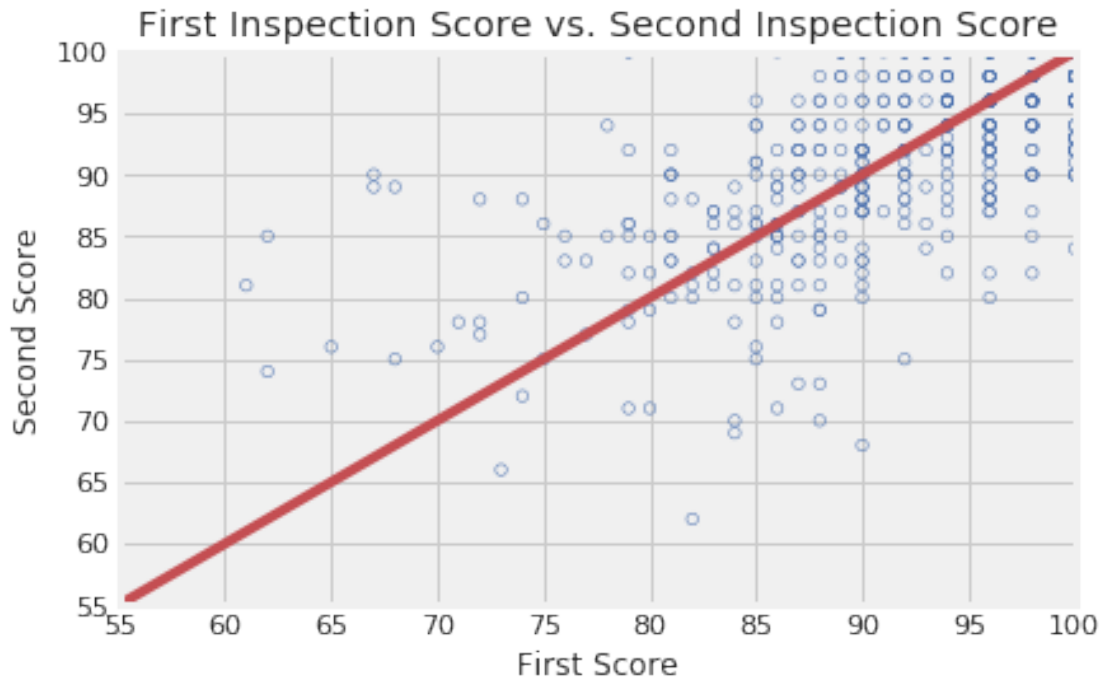
Key pieces of syntax you'll need:

`plt.scatter` plots a set of points. Use `facecolors='none'` and `edgecolors=b` to make circle markers with blue borders.

`plt.plot` for the reference line.

`plt.xlabel`, `plt.ylabel`, `plt.axis`, and `plt.title`.

Note: If you want to use another plotting library for your plots (e.g. `plotly`, `sns`) you are welcome to use that library instead so long as it works on DataHub.

Hint: You may find it convenient to use the `zip()` function to unzip scores in the list.

```
In [92]: first_second_score = list(zip(*scores_pairs_by_business['score_pair'].values))

         plt.plot(np.arange(45, 110), np.arange(45, 110), c='r')
         plt.scatter(first_second_score[0], first_second_score[1], facecolors='none', edgecolors='b')
         plt.axis(xmin=55, xmax=100, ymin=55, ymax=100)
         plt.xlabel("First Score")
         plt.ylabel("Second Score")
         plt.title("First Inspection Score vs. Second Inspection Score");
```

9

First Inspection Score vs. Second Inspection Score
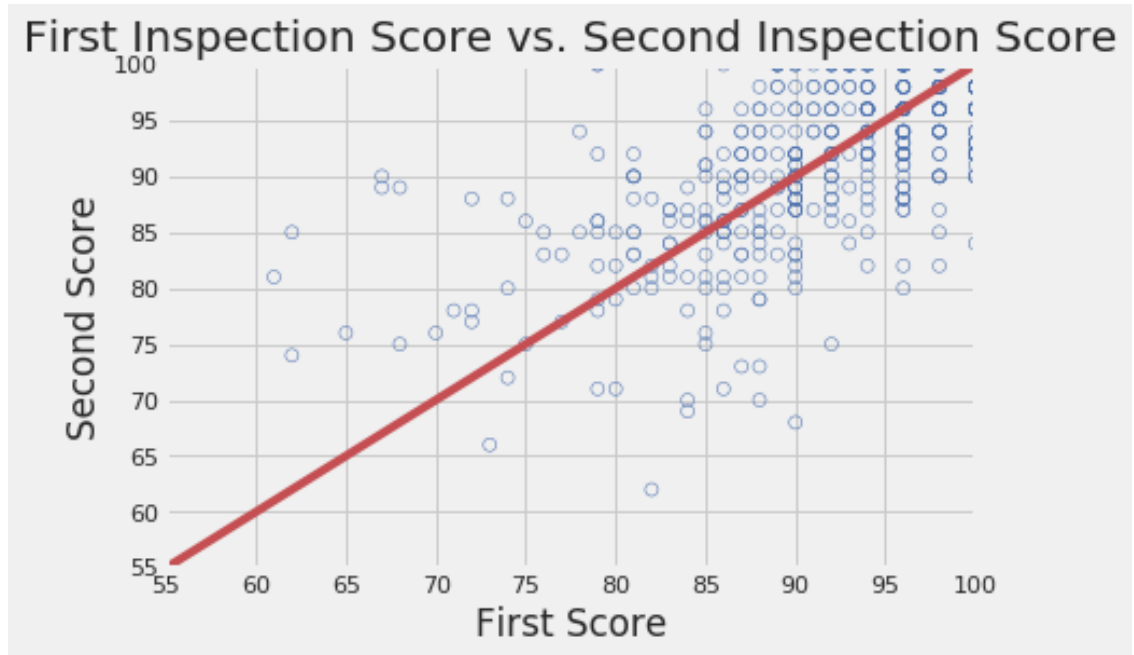
0.0.3 Question 2d

Another way to compare the scores from the two inspections is to examine the difference in scores. Subtract the first score from the second in `scores_pairs_by_business`. Make a histogram of these differences in the scores. We might expect these differences to be positive, indicating an improvement from the first to the second inspection.

The histogram should look like this:
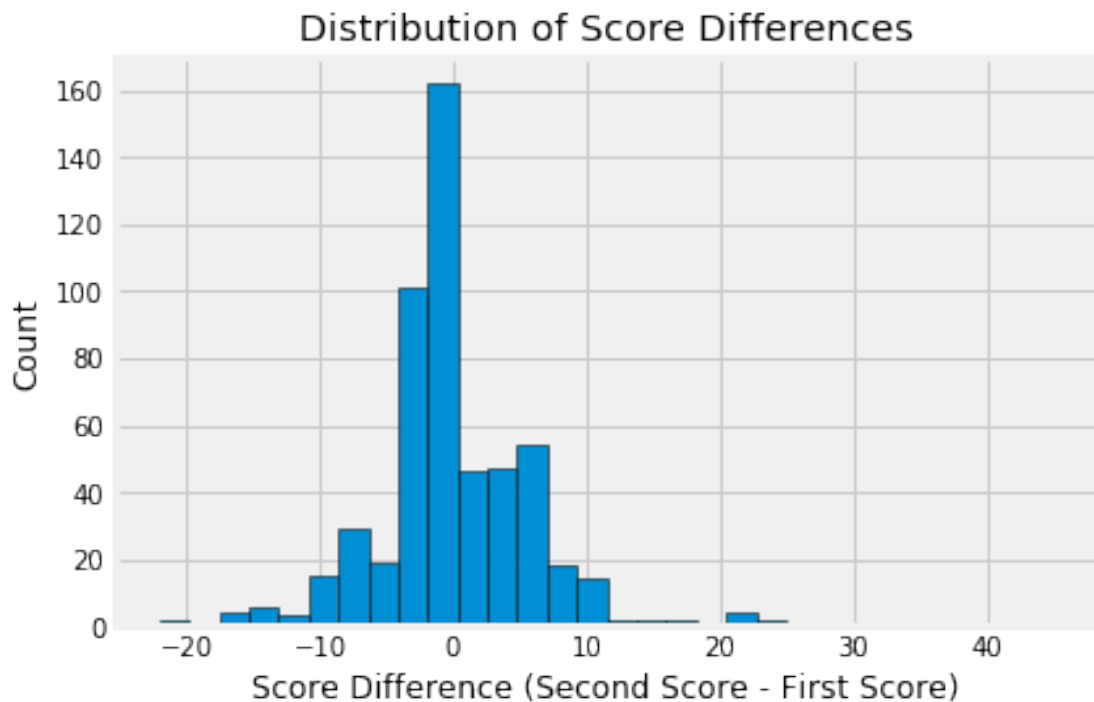
Hint: Use `second_score` and `first_score` created in the scatter plot code above.
Hint: Convert the scores into numpy arrays to make them easier to deal with.
Hint: Use `plt.hist()` Try changing the number of bins when you call `plt.hist()`.

```
In [99]: difference = np.subtract(first_second_score[1], first_second_score[0])
         plt.hist(difference, bins=40)
         plt.xlabel("Score Difference (Second Score - First Score)")
         plt.ylabel("Count")
         plt.title("Distribution of Score Differences");
```

Distribution of Score Differences
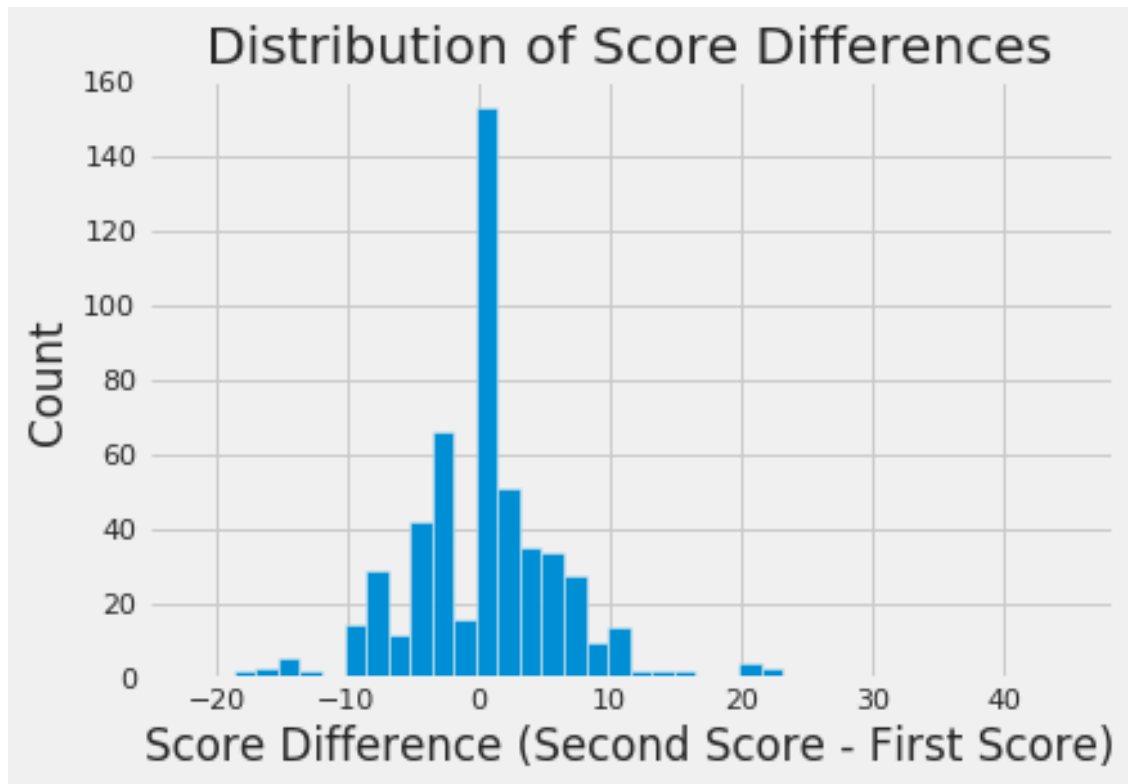
0.0.4 Question 2e

If restaurants' scores tend to improve from the first to the second inspection, what do you expect to see in the scatter plot that you made in question 2c? What do you oberve from the plot? Are your observations consistent with your expectations?

Hint: What does the slope represent?

If the majority of restaurants improved, I would expect the majority of scatter points to lie above the red line, which represents no change in score. The plot shows a large amount of restaurants have improved, but plenty have also gotten worse. This is consistent with my expectations, as most restaurants care about inspection scores, but many do not.

0.0.5 Question 2f

If a restaurant's score improves from the first to the second inspection, how would this be reflected in the histogram of the difference in the scores that you made in question 2d? What do you oberve from the plot? Are your observations consistent with your expectations? Explain your observations in the language of Statistics: for instance, the center, the spread, the deviation etc.

This would be reflected in the histogram by data points lying to the right of the center 0, as in the second score is larger than the firs score. Based on the histogram, there are a large number of restaurants at the center - these restaurants score relatively the same in both inspections, either improving or worsening slightly. This is again consistent with expectations, as I do not expect most restaurants to change significantly in their score between inspections.

0.0.6 Question 2g

To wrap up our analysis of the restaurant ratings over time, one final metric we will be looking at is the distribution of restaurant scores over time. Create a side-by-side boxplot that shows the distribution of these scores for each different risk category from 2017 to 2019. Use a figure size of at least 12 by 8.

The boxplot should look similar to the sample below:
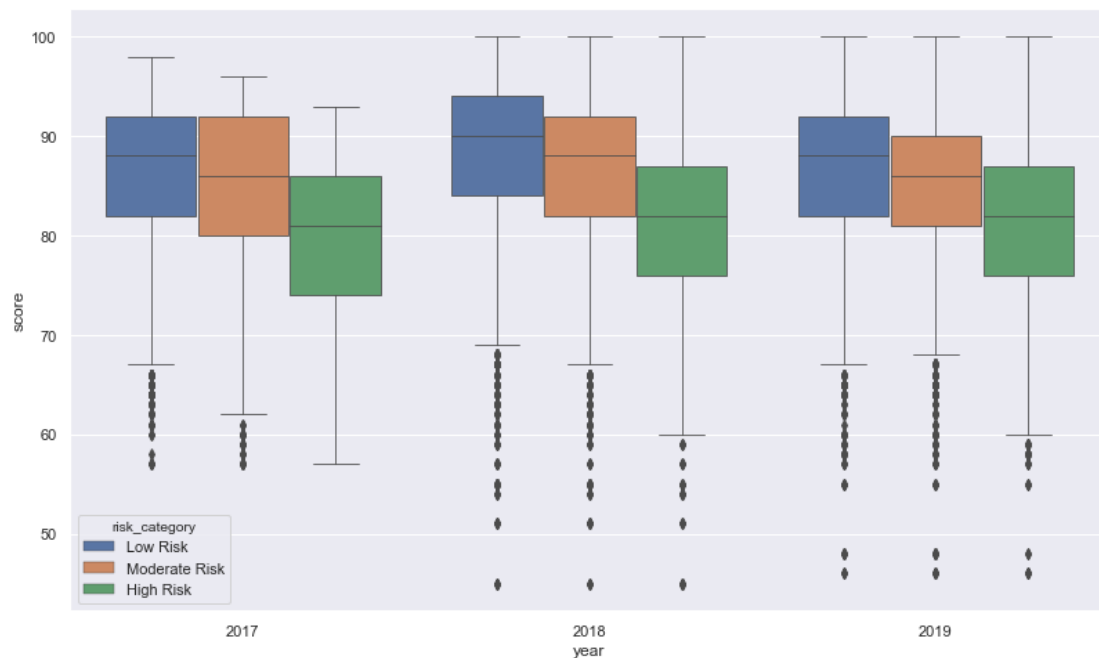
Hint: Use `sns.boxplot()`. Try taking a look at the first several parameters.
Hint: Use `plt.figure()` to adjust the figure size of your plot.

```
In [115]: # Do not modify this line
          sns.set()

          data = ins.merge(ins2vio).merge(vio)
          box_plot_data = data[(data.year.isin([2017, 2018, 2019])) & (data['Missing Score'] == 'False']

          plt.figure(figsize=(12, 8))
          sns.boxplot(x=box_plot_data['year'], y=box_plot_data['score'], hue=box_plot_data['risk_categor
```
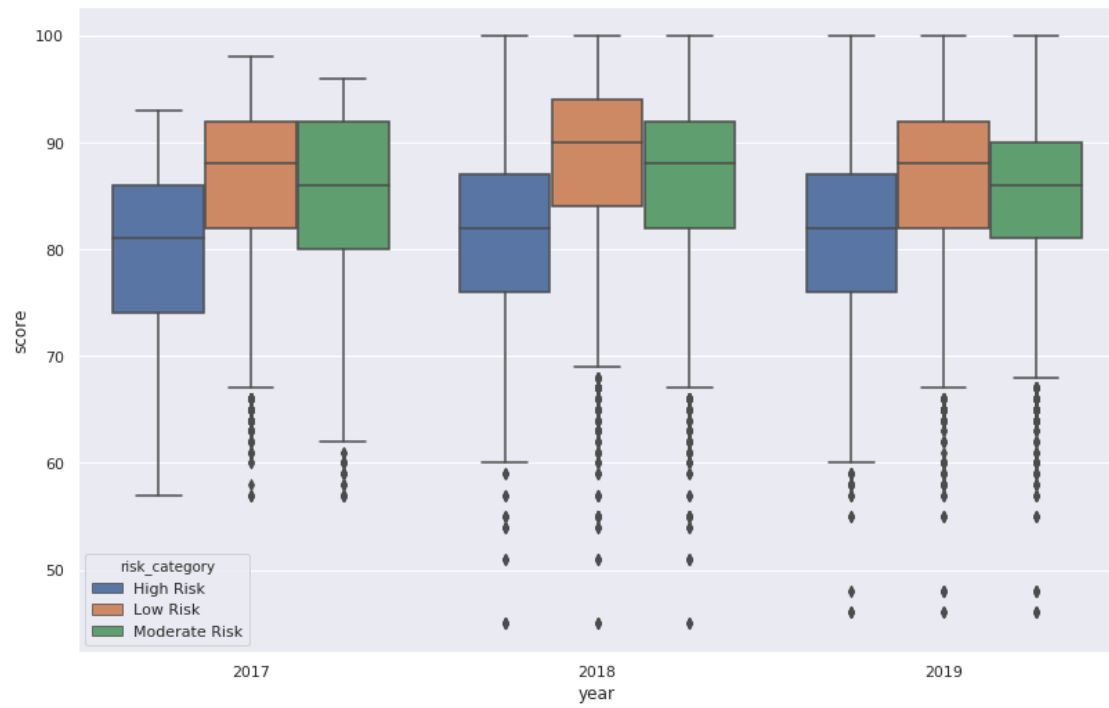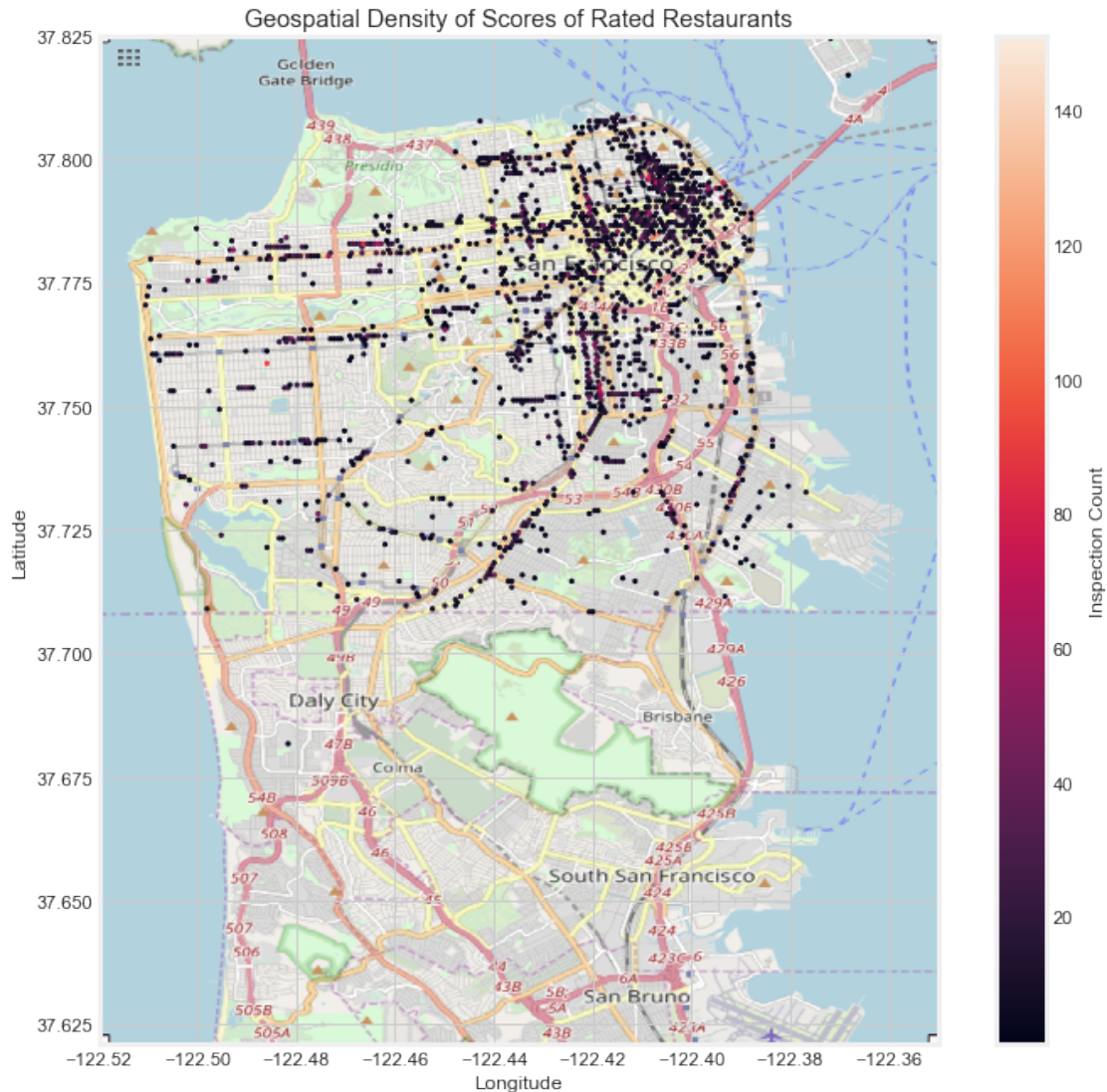
0.0.7 Question 3b

Now that we have our DataFrame ready, we can start creating our geospatial hexbin plot.

Using the `rated_geo` DataFrame from 3a, produce a geospatial hexbin plot that shows the inspection count for all restaurant locations in San Francisco.

Your plot should look similar to the one below:

Geospatial Density of Scores of Rated Restaurants

Hint: Use `pd.DataFrame.plot.hexbin()` or `plt.hexbin()` to create the hexbin plot.

Hint: For the 2 functions we mentioned above, try looking at the parameter `reduce_C_function`, which determines the aggregate function for the hexbin plot.

Hint: Use `fig.colorbar()` to create the color bar to the right of the hexbin plot.

Hint: Try using a `gridsize` of 200 when creating your hexbin plot; it makes the plot cleaner.

```
In [151]: from matplotlib import cm
          # DO NOT MODIFY THIS BLOCK
          min_lon = rated_geo['longitude'].min()
          max_lon = rated_geo['longitude'].max()
```

19

```python
min_lat = rated_geo['latitude'].min()
max_lat = rated_geo['latitude'].max()
max_score = rated_geo['score'].max()
min_score = rated_geo['score'].min()
bound = ((min_lon, max_lon, min_lat, max_lat))
min_lon, max_lon, min_lat, max_lat
map_bound = ((-122.5200, -122.3500, 37.6209, 37.8249))
# DO NOT MODIFY THIS BLOCK

# Read in the base map and setting up subplot
# DO NOT MODIFY THESE LINES
basemap = plt.imread('./data/sf.png')
fig, ax = plt.subplots(figsize = (11,11))
ax.set_xlim(map_bound[0],map_bound[1])
ax.set_ylim(map_bound[2],map_bound[3])
# DO NOT MODIFY THESE LINES


# Create the hexbin plot
magma = cm.get_cmap('magma')
rated_geo.plot.hexbin('longitude', 'latitude', 'score', reduce_C_function=np.size, gridsize=2
plt.title("Geospatial Density of Scores of Rated Restaurants")
plt.xlabel("Longitude")
plt.ylabel("Latitude")
fig.get_axes()[1].set_ylabel("Inspection Count")

# Setting aspect ratio and plotting the hexbins on top of the base map layer
# DO NOT MODIFY THIS LINE
ax.imshow(basemap, zorder=0, extent = map_bound, aspect= 'equal');
# DO NOT MODIFY THIS LINE
```

Geospatial Density of Scores of Rated Restaurants
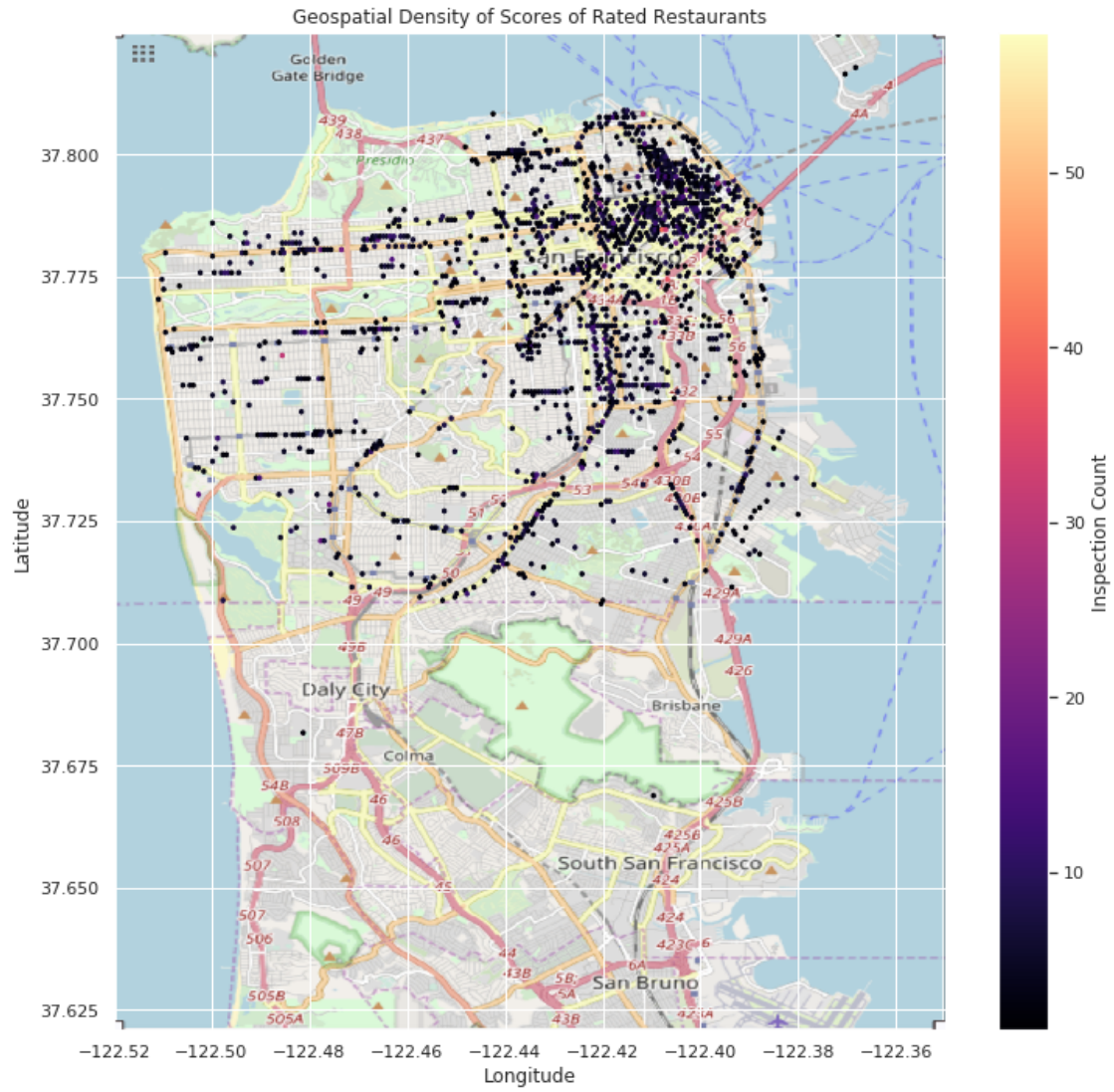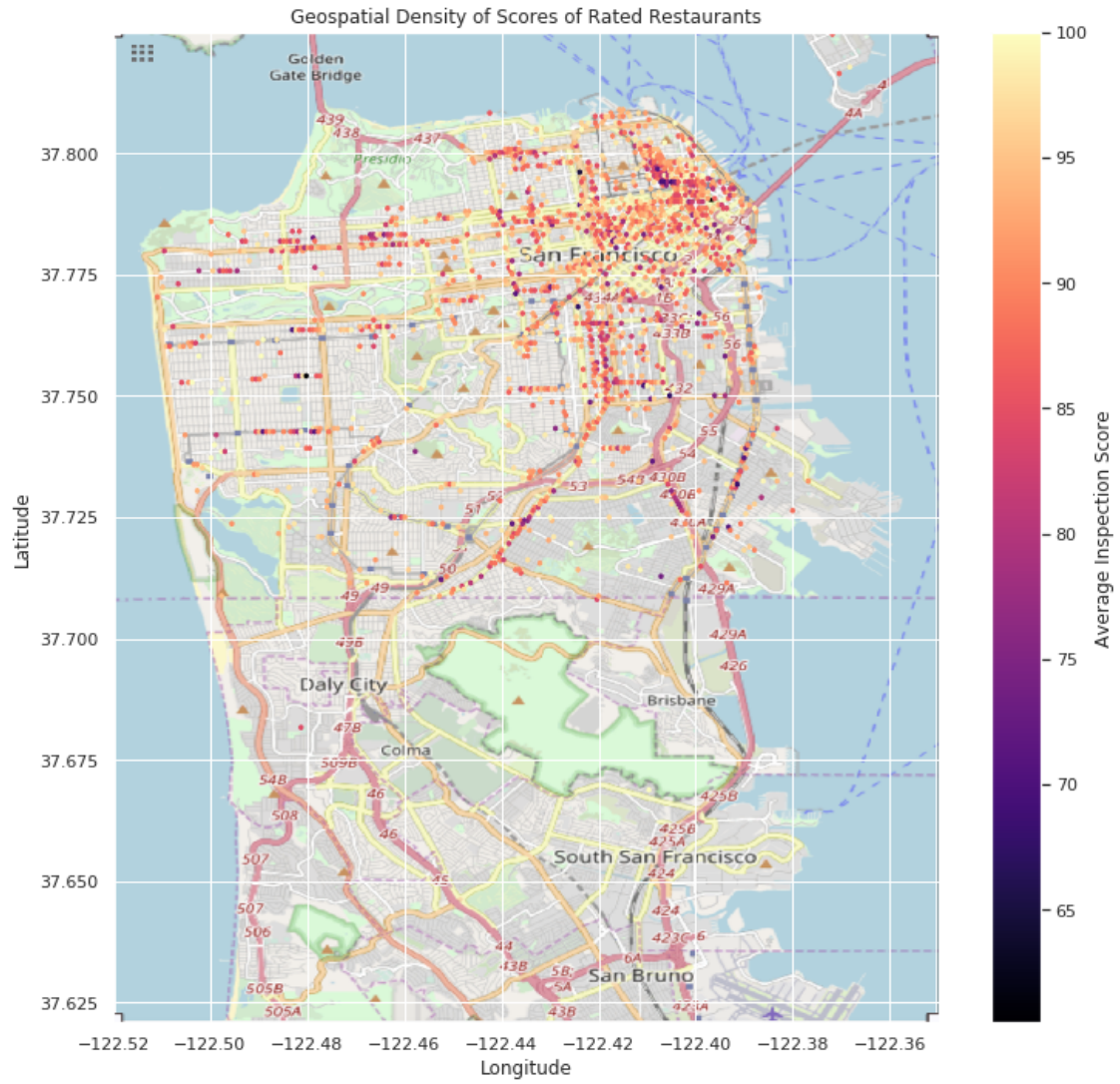
0.0.8 Question 3c

Now that we've created our geospatial hexbin plot for the density of inspection scores for restaurants in San Francisco, let's also create another hexbin plot that visualizes the **average inspection scores** for restaurants in San Francisco.

Hint: If you set up everything correctly in 3b, you should only need to change 1 parameter here to produce the plot.

```
In [161]:  # Read in the base map and setting up subplot
           # DO NOT MODIFY THESE LINES
           basemap = plt.imread('./data/sf.png')
           fig, ax = plt.subplots(figsize = (11,11))
           ax.set_xlim(map_bound[0],map_bound[1])
           ax.set_ylim(map_bound[2],map_bound[3])
           # DO NOT MODIFY THESE LINES


           # Create the hexbin plot
           rated_geo.plot.hexbin('longitude', 'latitude', 'score', reduce_C_function=np.mean, gridsize=2(
           plt.title("Geospatial Density of Scores of Rated Restaurants")
           plt.xlabel("Longitude")
           plt.ylabel("Latitude")
           fig.get_axes()[1].set_ylabel("Average Inspection Score")



           # Setting aspect ratio and plotting the hexbins on top of the base map layer
           # DO NOT MODIFY THIS LINE
           ax.imshow(basemap, zorder=0, extent = map_bound, aspect= 'equal');
           # DO NOT MODIFY THIS LINE
```

Geospatial Density of Scores of Rated Restaurants

0.0.9 Question 3d

Given the 2 hexbin plots you have just created above, did you notice any connection between the first plot where we aggregate over the **inspection count** and the second plot where we aggregate over the **inspection mean**? In several sentences, comment your observations in the cell below.

Here're some of the questions that might be interesting to address in your response:

- Roughly speaking, did you notice any of the actual locations (districts/places of interest) where inspection tends to be more frequent? What about the locations where the average inspection score tends to be low?
- Is there any connection between the locations where there are more inspections and the locations where the average inspection score is low?
- What have might led to the connections that you've identified?

Inspections seem to be more frequent in more populated areas, as expected. Especially near major roads and bart lines, there appear to be the most inspections. This makes sense, as there are a high number of people, and consequently, restaurants in these areas. Low average inspection scores appear to be fairly distributed, but are more concentrated in areas with a high number of inspections.

There appears to be an inverse relation between frequency of inspections and average inspection score - areas with a high number of inspections tend to have lower average inspection scores. Chinatown, Union Square, and the Mission District, for instance, represent this trend.

This most likely occurs due to the high prevalence of cheap/fast food options in this location. Additionally, restaurants that do not score well are more likely to be inspected often. Coupled with the general lack of improvement in inspection scores, it makes sense for these areas to have both a high frequency and low average of inspection scores.

0.0.10 Grading

Since the assignment is more open ended, we will have a more relaxed rubric, classifying your answers into the following three categories:

- **Great** (4-5 points): The chart is well designed, and the data computation is correct. The text written articulates a reasonable metric and correctly describes the relevant insight and answer to the question you are interested in.
- **Passing** (3-4 points): A chart is produced but with some flaws such as bad encoding. The text written is incomplete but makes some sense.
- **Unsatisfactory** ($<= 2$ points): No chart is created, or a chart with completely wrong results.

We will lean towards being generous with the grading. We might also either discuss in discussion or post on Piazza some examplar analysis you have done (with your permission)!

You should have the following in your answers: * a few visualizations; Please limit your visualizations to 5 plots. * a few sentences (not too long please!)

Please note that you will only receive support in OH and Piazza for Matplotlib and seaborn questions. However, you may use some other Python libraries to help you create you visualizations. If you do so, make sure it is compatible with the PDF export (e.g., Plotly does not create PDFs properly, which we need for Gradescope).

```
In [162]: # YOUR DATA PROCESSING AND PLOTTING HERE
          time_geo = ins_and_bus[['latitude', 'longitude', 'timestamp']]
          time_geo.timestamp = time_geo.timestamp.astype(int)

          basemap = plt.imread('./data/sf.png')
          fig, ax = plt.subplots(figsize = (11,11))
          ax.set_xlim(map_bound[0],map_bound[1])
          ax.set_ylim(map_bound[2],map_bound[3])

          time_geo.plot.hexbin('longitude', 'latitude', 'timestamp', reduce_C_function=np.mean, gridsiz
          plt.title("Geospatial Density of Inspection Times of Rated Restaurants")
          plt.xlabel("Longitude")
          plt.ylabel("Latitude")
          fig.get_axes()[1].set_ylabel("Average Time of Inspection")

          ax.imshow(basemap, zorder=0, extent = map_bound, aspect= 'equal');

          # YOUR EXPLANATION HERE (in a comment)
          # This visualization shows the average time at which an inspection for a given restaurant hap
          # difficult to understand,
```

Geospatial Density of Inspection Times of Rated Restaurants