0.1 1.b. Map traffic speed to Google Plus Codes

Google Plus Codes divide up the world uniformly into rectangular slices (link). Let's use this to segment traffic speeds spatially. Take a moment to answer: **Is this spatial structure effective for summarizing traffic speed?** Before completing this section, substantiate your answer with examples of your expectations (e.g., we expect A to be separated from B). After completing this section, substantiate your answer with observations you've made.

Since we are looking at a uniform rectangular slice, we can take the average of speeds at all the longtidues and latitudes that fall within that slice, so it's convenient to spatially divide area using pluscodes. However, they disregards the traffic flow and do not capture much about the subpopulations since the divisions between neighborhoods, highways, intersections, etc… are abitrary. Before completing this section, we expected pluscodes to not be as accurate as some other method that perhaps takes demographic or divisions between neighborhoods into account. Because of that, it turns out that pluscodes are not too effective at summarizing traffic speeds since they have a high variation between and within each slice.

0.1.1 1.b.v. How well do plus code regions summarize movement speeds?

The following will give us an idea of how well the average represents traffic speed per plus code region. For these questions, we'll refer to a "plus code region" as a "cluster":

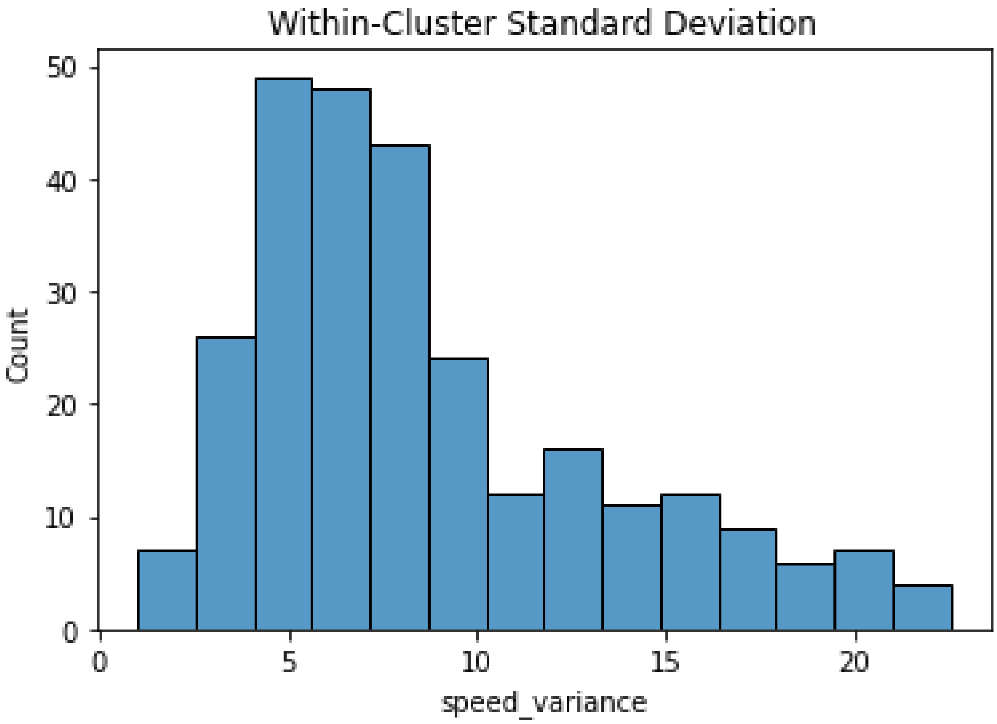
1. **Plot a histogram of the within-cluster standard deviation**.
2. **Compute across-cluster average of within-cluster standard deviation**.
3. **Compute across-cluster standard deviation of within-cluster average speeds**.
4. **Is this average variance reasonable?** To assess what "reasonable" means, consider these questions and how to answer them: (1) Do plus codes capture meaningful subpopulations? (2) Do differences between subpopulations outweigh differences within a subpopulation? Use the statistics above to answer these questions, and compute any additional statistics you need. Additionally explain *why these questions are important to assessing the quality of a spatial clustering.*

Hint: Run the autograder first to ensure your variance average and average variance are correct, before starting to draw conclusions.

In the first cell, write your written answers. In the second cell, complete the code.

The histogram is skewed to the right and the average variance turned out to be ~8.7 mph. The average standard deviation in the whole dataset is higher: ~10.1 mph. While subdividing regions and analyzing traffic by pluscodes did decrease average variance (-8.7 mph variance seems somewhat reasonable, as most people driving on streets tend to stray from the posted speed by about 5-10mph), pluscodes still may not be the best method for analyzing traffic. Due to pluscodes being arbitrary with the way they capture traffic, as they divide regions based solely on latitude and longitude, they may not accurately capture subpopulations. In other words, they do not take into account the divisions between neighborhoods and highways, intersections, etc…which all contribute to the the average speed in that location. This might be why we observe speed variance as high as 15 more commonly than we do in the census tract subdivision method.

```
In [15]: speed_variance_by_pluscode = speeds_to_gps[["speed_mph_mean", "plus_latitude_idx", "plus_longi
         sns.histplot(speed_variance_by_pluscode, x = "speed_variance")
         plt.title("Within-Cluster Standard Deviation")
         average_variance_by_pluscode = speed_variance_by_pluscode["speed_variance"].mean()
         variance_average_by_pluscode = speeds_to_gps[["speed_mph_mean", "plus_latitude_idx", "plus_lon
```



0.1.2 1.c.iv. How well do census tracts summarize movement speeds?

The following will give us an idea of how well the average represents traffic speed per plus code region. For these questions, we'll refer to a "census tract" as a "cluster":

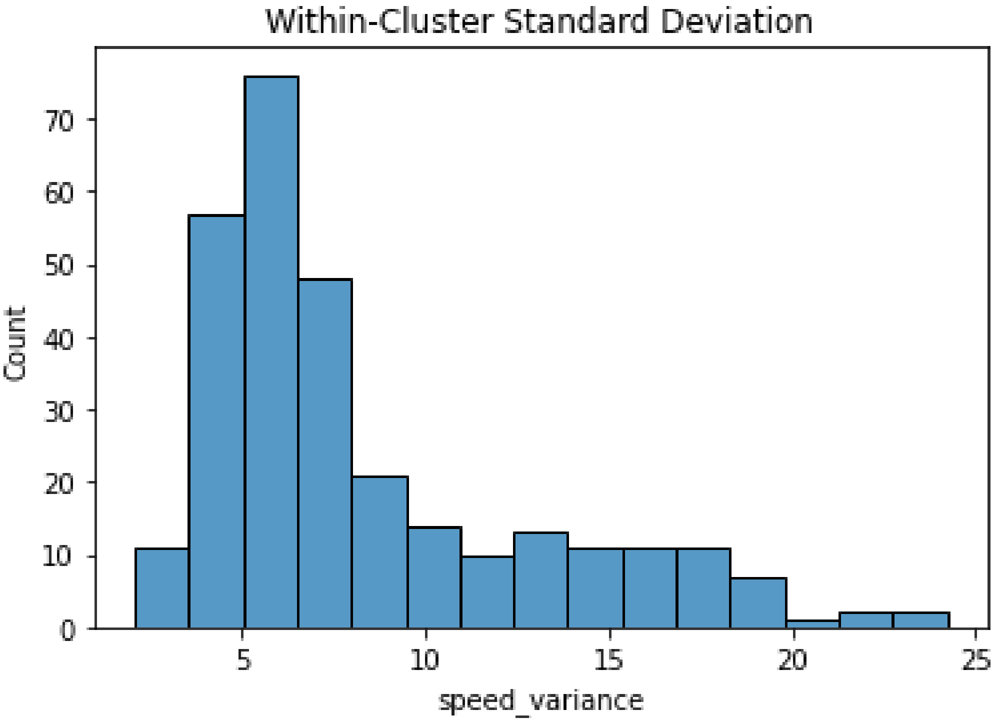
1. **Plot a histogram of the within-cluster standard deviation**.
2. **Compute across-cluster average of within-cluster standard deviation**.
3. **Compute across-cluster standard deviation of within-cluster average speeds**.
4. **Is this average variance reasonable?** To assess what "reasonable" means, consider these questions and how to answer them: (1) Do plus codes capture meaningful subpopulations? (2) Do differences between subpopulations outweigh differences within a subpopulation? Use these ideas to assess whether the average standard deviation is high or not.

Note: We are using the speed metric of miles per hour here.

Just like before, please written answers in the first cell and coding answers in the second cell.

The average variance by tract is ~8.3 mph, which is lower than the average variance by pluscode. Also, the varaince of the average speeds in the whole dataset is ~8.34, which is close to the average variance by tract. This shows that the tracts are representative of the subpopulations as they take into account the populations in that geographical space and areas without street addresses, covering the more rural/suburban outkirts areas. Also, we see that the histogram is skewed to the right, with most variances falling under the 5-10 mph; this may be justified by the fact that some people tend to drive 5-10 miles faster/slower than the posted speed. Given the overall context, the average standard deviation is not high.

```
In [24]: speed_variance_by_tract = speeds_to_tract[["speed_mph_mean", "DISPLAY_NAME"]].groupby("DISPLAY_
         sns.histplot(speed_variance_by_tract, x = "speed_variance")
         plt.title("Within-Cluster Standard Deviation")
         average_variance_by_tract = speed_variance_by_tract["speed_variance"].mean()
         variance_average_by_tract = speeds_to_tract[["speed_mph_mean", "DISPLAY_NAME"]].groupby("DISPLA
```



0.2 1.d. What would be the ideal spatial clustering?

This is an active research problem in many spatiotemporal modeling communities, and there is no single agreed-upon answer. Answer both of the following specifically knowing that you'll need to analyze traffic patterns according to this spatial clustering:

1. **What is a good metric for a spatial structure?** How do we define good? Bad? What information do we expect a spatial structure to yield? Use the above parts and questions to help answer this.
2. **What would you do to optimize your own metric for success in a spatial structure?**

See related articles:

- Uber's H3 link, which divides the world into hexagons
- Traffic Analysis Zones (TAZ) link, which takes census data and additionally accounts for vehicles per household when dividing space

We can assess which spatial structure allows for easier calcuation of neighboring distances; for instance, a hexgaon does not have as many neighboring distances as a square, which requires multiple set of coefficients for tracking due to having two different types of neighbors, one sharing an edge and the other a vertex. H3 can be considered a good spatial structure because it keeps track of the spatial area in a simple manner and is consistent with how it defines each area: a hexagon of the same size. On the other hand, postal codes would be a bad spatial structure because it would localize area into unusual shapes and these areas are subject to change, which is problematic. TAZ, though does not have uniform slices, takes into account demographic information, so it may be better at capturing subpopulations more accurately.

So overall, it is important to be consistent with how each area is defined, so it's easier to add/modify areas in the future, and simplify performing analyses and smoothing over gradients. Moreover, to better capture the subpopulations, it may be useful to incorporate demographics for each slice. Additionally, we can use the variance of speed within each slice/hexagon/space to measure how well the slices capture subpopulations.

Above, we've defined our metrics as variance of speed and consistency of slices. To optimize variance of speed (decrease them), we would want to decrease the size of slices. For example, when comparing the census tracts and the plus codes, we found that the average variance for census tracts was lower. The census tracts were smaller than plus codes; they subdivided the plus codes.

To optimize the consistency of shapes, we can make the area more uniform for census tracts (like the pluscodes) and use a shape, such as a hexagon (like Uber's H3) because that allows for more scalablility and easier analysis. We can also add demograpahic features, such as the number of cars each house has, to better understand the subpopulation and thus estimate traffic flow.

0.2.1 2.a.i. Sort census tracts by average speed, pre-lockdown.

Consider the pre-lockdown period to be March 1 - 13, before the first COVID-related restrictions (travel bans) were announced on March 14, 2020.

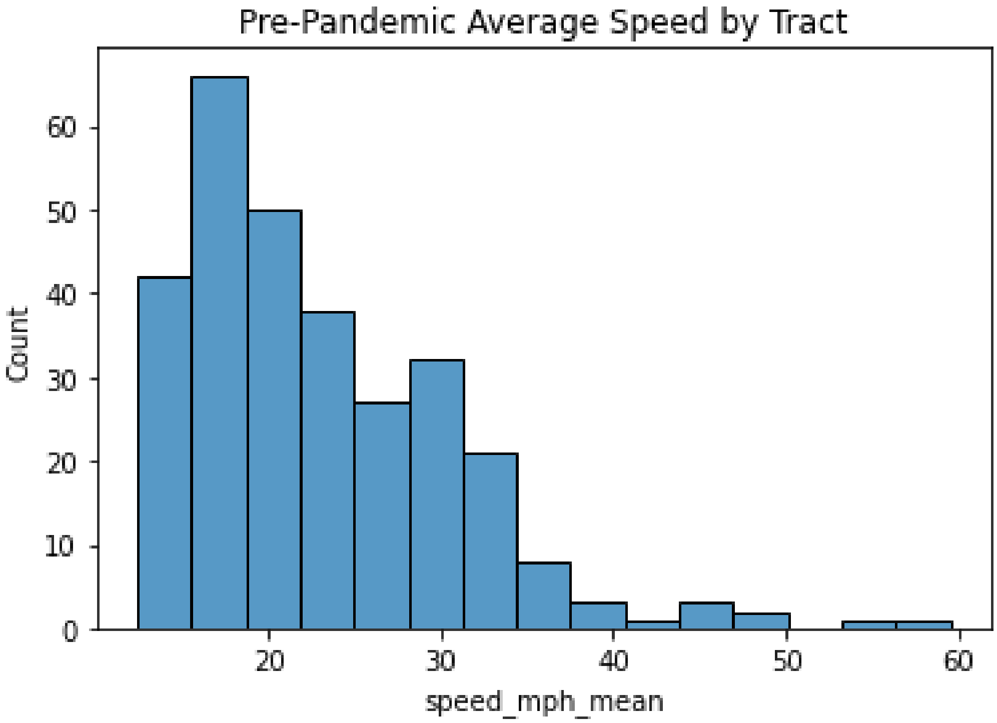
1. **Report a DataFrame which includes the *names* of the 10 census tracts with the lowest average speed**, along with the average speed for each tract.
2. **Report a DataFrame which includes the *names* of the 10 census tracts with the highest average speed**, along with the average speed for each tract.
3. Do these names match your expectations for low speed or high speed traffic pre-lockdown? What relationships do you notice? (What do the low-speed areas have in common? The high-speed areas?) For this specific question, answer qualitatively. No need to quantify. **Hint**: Look up some of the names on a map, to understand where they are.
4. **Plot a histogram for all average speeds, pre-lockdown**.
5. You will notice a long tail distribution of high speed traffic. What do you think this corresponds to in San Francisco? Write down your hypothesis.

Hint: To start off, think about what joins may be useful to get the desired DataFrame.

Our hypothesis is that the high speed traffic corresponds to the highways/freeways in San Francisco, and not the regular streets running through the city, because speeds are usually highest on highways._Type your answer here, replacing this text._

Plot the histogram

```
In [33]: sns.histplot(averages_pre_named, x="speed_mph_mean")
         plt.title("Pre-Pandemic Average Speed by Tract");
```



0.2.2 2.a.ii. Sort census tracts by average speed, post-lockdown.

I suggest checking the top 10 and bottom 10 tracts by average speed, post-lockdown. Consider the post-lockdown period to be March 14 - 31, after the first COVID restrictions were established on March 14, 2020. It's a healthy sanity check. For this question, you should report:

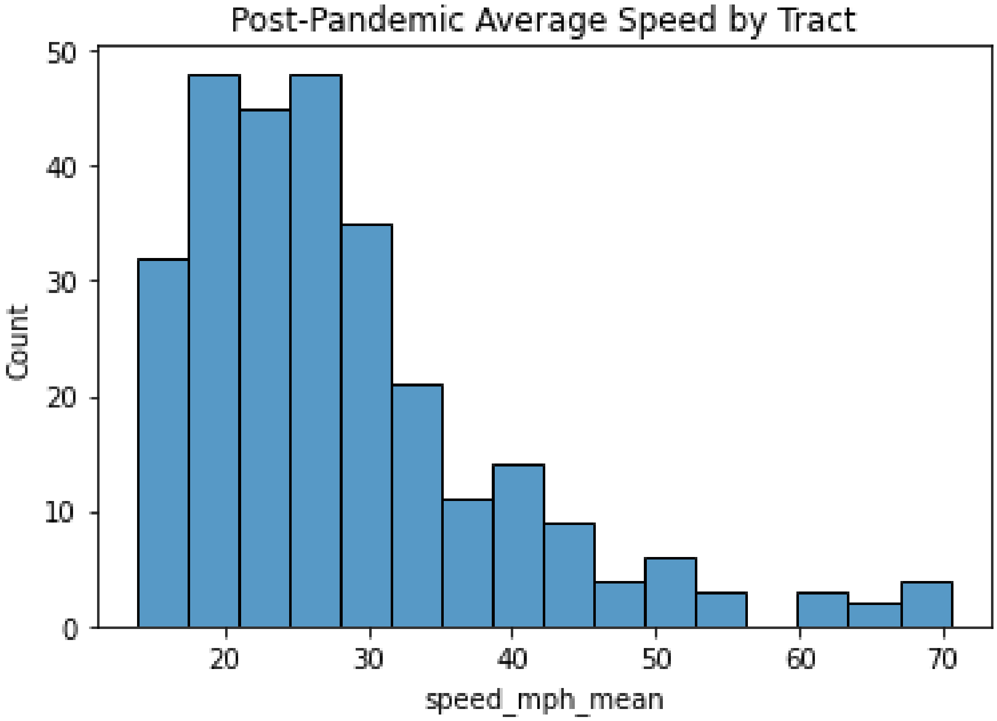
- **Plot a histogram for all average speeds, post-lockdown.**
- **What are the major differences between this post-lockdown histogram relative to the pre-lockdown histogram above**? Anything surprising? What did you expect, and what did you find?

Write the written answers in the cell below, and the coding answers in the cells after that.

Type your answer here, replacing this text.

Plot the histogram

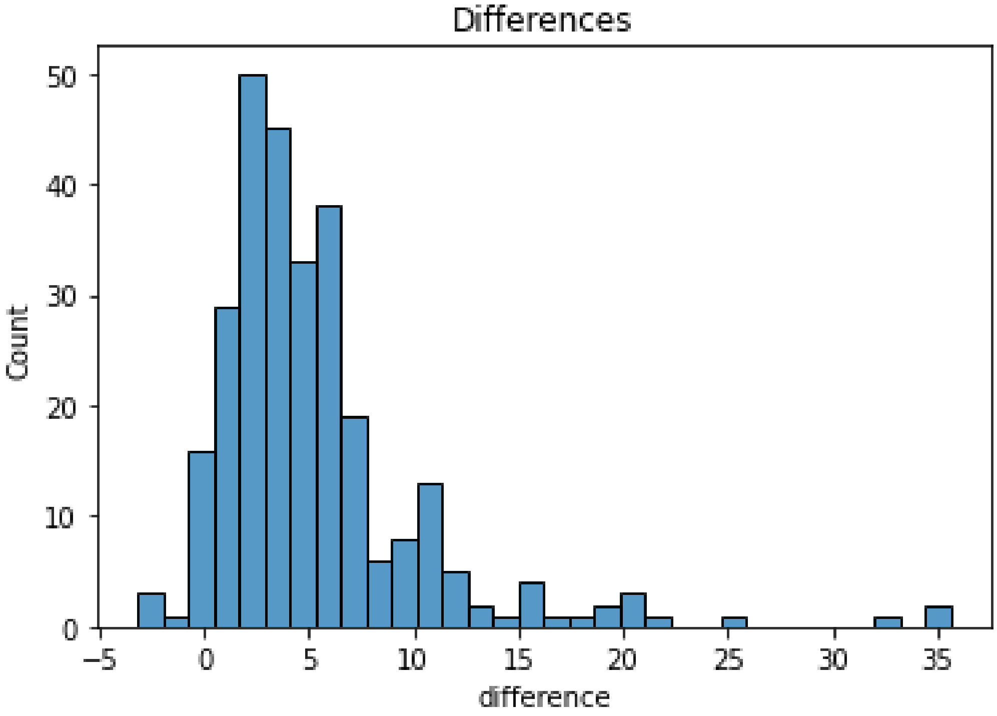
```
In [36]: sns.histplot(averages_post_named, x="speed_mph_mean")
         plt.title("Post-Pandemic Average Speed by Tract");
```



0.2.3 2.a.iii. Sort census tracts by change in traffic speed from pre to post lockdown.

For each segment, compute the difference between the pre-lockdown average speed (March 1 - 13) and the post-lockdown average speed (March 14 - 31). **Plot a histogram of all differences.** Sanity check that the below histogram matches your observations of the histograms above, on your own.

```
In [37]: # The autograder expects differences to be a series object with index
         # MOVEMENT_ID.
         merged_pre_post = averages_post_named.reset_index().merge(averages_pre_named.reset_index(),
                                                                    how="inner", left_on="DISPLAY_NAME",
                                                                    right_on="DISPLAY_NAME")
         merged_pre_post_cleaned = merged_pre_post[["speed_mph_mean_x", "speed_mph_mean_y", "MOVEMENT_II
         merged_pre_post_cleaned["difference"] = merged_pre_post_cleaned["speed_mph_mean_x"] - merged_p
         merged_pre_post_cleaned.reindex(["MOVEMENT_ID"])
         differences = merged_pre_post_cleaned["difference"]
         # plot the differences
         sns.histplot(merged_pre_post_cleaned, x="difference")
         plt.title("Differences");
```



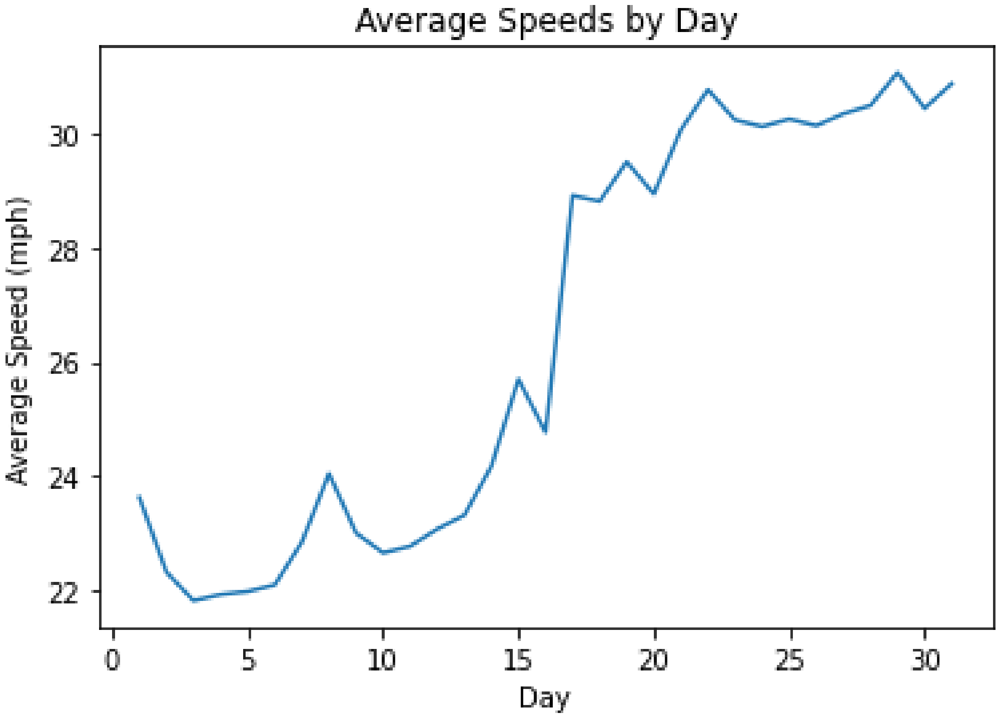
```
In [38]: grader.check("q2aiii")
```

```
Out[38]: q2aiii results: All test cases passed!
```

0.2.4 2.a.iv. Quantify the impact of lockdown on average speeds.

1. **Plot the average speed by day, across all segments**. Be careful not to plot the average of census tract averages instead. Recall the definition of segments from Q1.
2. Is the change in speed smooth and gradually increasing? Or increasing sharply? Why? Use your real-world knowledge of announcements and measures during that time, in your explanation. You can use this list of bay area COVID-related dataes: https://abc7news.com/timeline-of-coronavirus-us-covid-19-bay-area-sf/6047519/

```
In [39]: # Autograder expects this to be a series object containing the
         # data for your line plot -- average speeds per day.
         speeds_daily = speeds_to_tract.groupby("day").agg("mean")["speed_mph_mean"]

         plt.plot(speeds_daily)
         plt.title("Average Speeds by Day")
         plt.xlabel("Day")
         plt.ylabel("Average Speed (mph)");
```



Write your written answer in the cell below

In general, the average speed increases pretty sharply from around day 14-17. However, it increases a little from day 14-15 before very sharply increasing around day 16-17.

Day 14-16: The average speed increased from around 23 mph to 25 mph (9% increase). This may be due to the beginnings of the lockdown. On March 14, 2020, UC Berkeley and Sonoma County announced its first reported cases, while Contra Costa and San Mateo began banning large group gatherings. On March 15, Governor Newsom orders all bars, nightclubs, wineries, and pubs to close, while urging those over 65 to stay home.

Day 16-17: The average speed increased sharply from around 25 mph to 29 mph (16%). This may be due to the shelter-in-place orders being announced on March 16 and taking effect on March 17. Once nearly 7 million Bay Area residents are told to shelter-in-place and restrict activities, traffic would probably decrease sharply, as residents only leave their houses for essential activities. Average speeds probably rose sharply due to the significant decrease in traffic on the road.

As a result of these growing restrictions, traffic began to gradually decrease, leading to higher average speeds, overall. After day 20, the graph begins to settle (since after all, speed limits are still in place)._Type your answer here, replacing this text._

0.2.5 2.a.v. Quantify the impact of pre-lockdown average speed on change in speed.

1. Compute the correlation between change in speed and the *pre*-lockdown average speeds. Do we expect a positive or negative correlation, given our analysis above?
2. Compute the correlation between change in speed and the post-lockdown average speeds.
3. **How does the correlation in Q1 compare with the correlation in Q2?** You should expect a significant change in correlation value. What insight does this provide about traffic?

Written answers in the first cell, coding answerts in the following cell.

The post-lockdown, difference correlation coefficient (.79) is significantly higher (stronger) than the pre-lockdown, difference correlation coefficient (.46). That means that the linear relationship between post-lockdown speeds and the change in speed is much stronger than pre-lockdown speeds and the change in speed. Thus, the changes are more "set" by the post-lockdown speeds than the pre-, which can indicate that pre-lockdown speeds were more clustered. This is consistent with the histograms seen above, as the pre-lockdown average speeds were more skewed and had more density around the 0-20 mph range than post-lockdown.

In a real world sense, this also makes sense. Pre-lockdown, every street/highway had significantly more traffic. This would slow down your speed on a main street and perhaps make it closer to speeds you would see on a side road with stop signs. This would also significantly slow down your speed on highways. As such, traffic could make speeds a lot more uniform across streets/highways. However, post-lockdown, there was significantly less traffic, which would lead to more divergence in speed. For example, with no traffic on main streets, you could go significantly faster than a side road. On highways as well, with less traffic, speeds became much higher than pre-lockdown, when traffic limited average speed. As such, post-lockdown, there was a lot more variance in speed as compared to pre-lockdown. Thus, when calculating correlation coefficients between post-lockdown speeds and change in speed vs pre-lockdown speeds and change in speed, it makes sense that the post-lockdown correlation coefficient would be higher, as the change is more so dictated by the post-lockdown speed.

0.2.6 2.b.i. Visualize spatial heatmap of average traffic speed per census tract, pre-lockdown.

Visualize a spatial heatmap of the grouped average daily speeds per census tract, which you computed in previous parts. Use the geopandas chloropleth maps. **Write your observations, using your visualization, noting down at least 2 areas or patterns of interest**. These may be a local extrema, or a region that is strangely all similar.

Hint: Use `to_crs` and make sure the `epsg` is using the Pseudo-Mercator projection.

Hint: You can use `contextily` to superimpose your chloropleth map on a real geographic map.

Hint You can set a lower opacity for your chloropleth map, to see what's underneath, but be aware that if you plot with too low of an opacity, the map underneath will perturb your chloropleth and meddle with your conclusions.

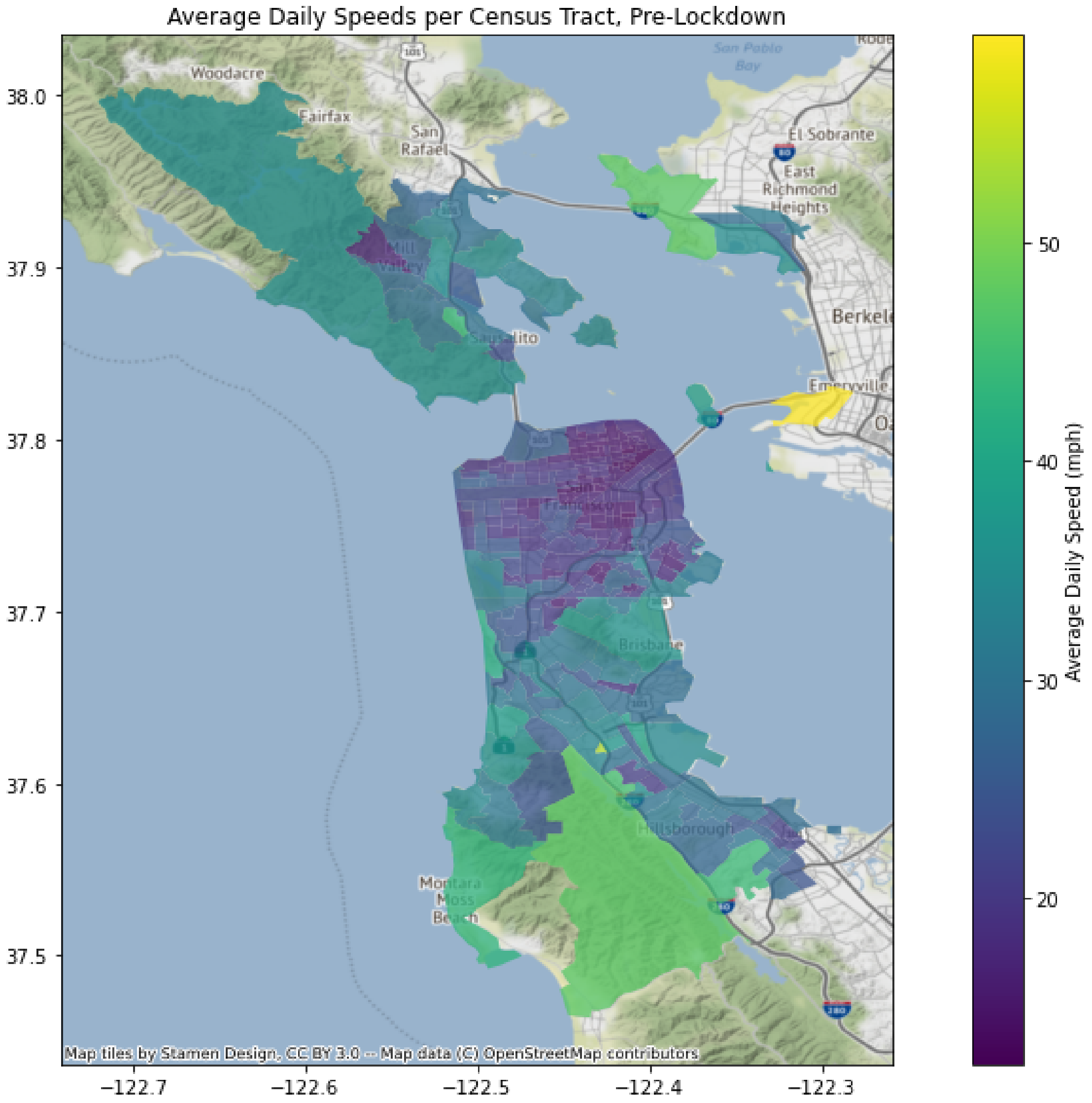
Written answers in the first cell, coding answers in the second cell.

1st pattern: The San Francisco area has the lowest traffic. Most parts of San Francisco are purple (less than 20 mph), and San Francisco has the deepest colors. SF's population is the densest in the Bay and usually has a lot of traffic, so it makes sense that average speeds are lowest in SF.

2nd pattern: The lowest speeds are seen around the bottom right area of the graph, in the light green area (around 53-56 mph). These average speeds are significantly higher than those in the surrounding areas. This may be due to the fact that the area consists of a park (with very few cars) and the 280 highway (which sees a lot of traffic at high speeds). As such, even though the vast majority of the area is a park/mountain range, the average speed in that census tract is above 50 mph._Type your answer here, replacing this text._

```
In [43]: day_tract = speeds_to_tract.groupby("DISPLAY_NAME").agg('mean').reset_index().merge(tract_to_g
         averages_pre_named_geo = averages_pre_named.merge(day_tract, on="DISPLAY_NAME")
         bx = averages_pre_named_geo.plot(column="speed_mph_mean_x", figsize = (15, 10) , legend = True
                              legend_kwds={'label': "Average Daily Speed (mph)",'orientation': "vertical
                             alpha=.75);
         cx.add_basemap(bx, crs=day_tract.crs.to_string())
         plt.title("Average Daily Speeds per Census Tract, Pre-Lockdown")
```

```
Out[43]: Text(0.5, 1.0, 'Average Daily Speeds per Census Tract, Pre-Lockdown')
```



0.2.7 2.b.ii. Visualize change in average daily speeds pre vs. post lockdown.

Visualize a spatial heatmap of the census tract differences in average speeds, that we computed in a previous part. **Write your observations, using your visualization, noting down at least 2 areas or patterns of interest.** Some possible ideas for interesting notes: Which areas saw the most change in average speed? Which areas weren't affected? Why did some areas see *reduced* average speed?

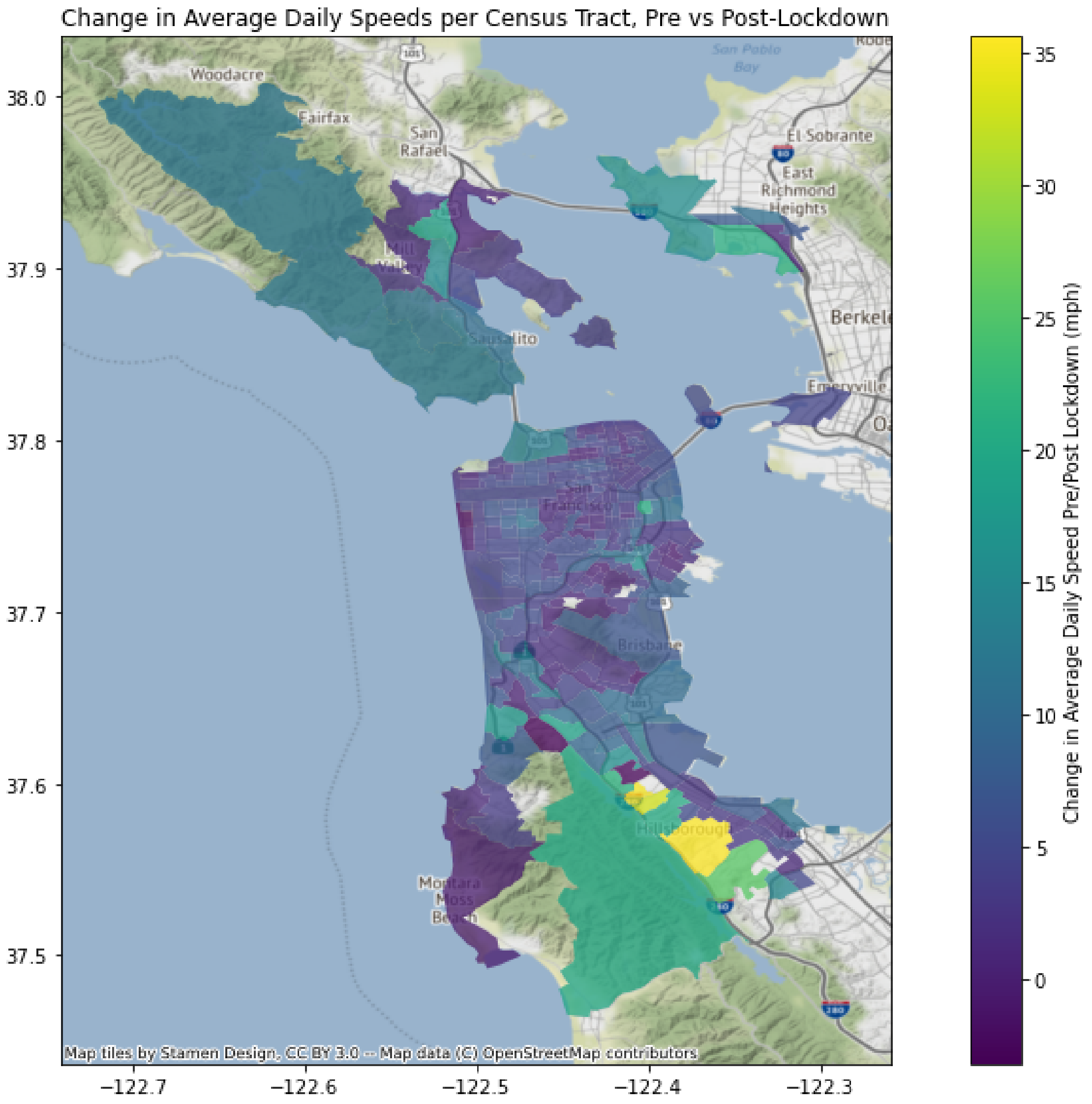
First cell is for the written answers, second cell is for the coding answers.

1st pattern: The Hillsborough/San Mateo area sees the largest change in average speed, increasing around 35 mph. This is probably due to the highway that runs through that census tract. As traffic decreases significantly on the highway, cars can go significantly faster.

2nd pattern: The SF area remains relatively unchanged, though there are some small changes in speed. This may be due to SF's roads being crowded (a lot of stoplights and stop signs). As such, even though traffic may have decreased, the average speed still cannot change much, as they are still limited by stoplights and stop signs.

3rd pattern: The Muir Woods/Golden Gate Recreational area sees a change of about 15 mph. This may be because people were not visiting these areas as often, leading to less traffic. The roads are also not as crowded together as SF's, but not as straightforward as 280 highway (it is more hilly, which restricts speed). As such, it makes sense that the traffic in this area increased more than SF's, but less than the 280 highway.

```
In [44]: merged = averages_post_named.merge(averages_pre_named, on = "DISPLAY_NAME")
         changes = merged.copy()
         changes["differences"] = changes["speed_mph_mean_x"] - changes["speed_mph_mean_y"]
         changes = day_tract.merge(changes, on = "DISPLAY_NAME")

         bx2 = changes.plot(column = "differences", figsize = (15, 10) , legend = True,
                           legend_kwds={'label': "Change in Average Daily Speed Pre/Post Lockdown (mph)
                           alpha=.75);
         cx.add_basemap(bx2, crs=changes.crs.to_string())
         plt.title("Change in Average Daily Speeds per Census Tract, Pre vs Post-Lockdown");
```



0.2.8 4.a.ii. Train and evaluate linear model on pre-lockdown data.

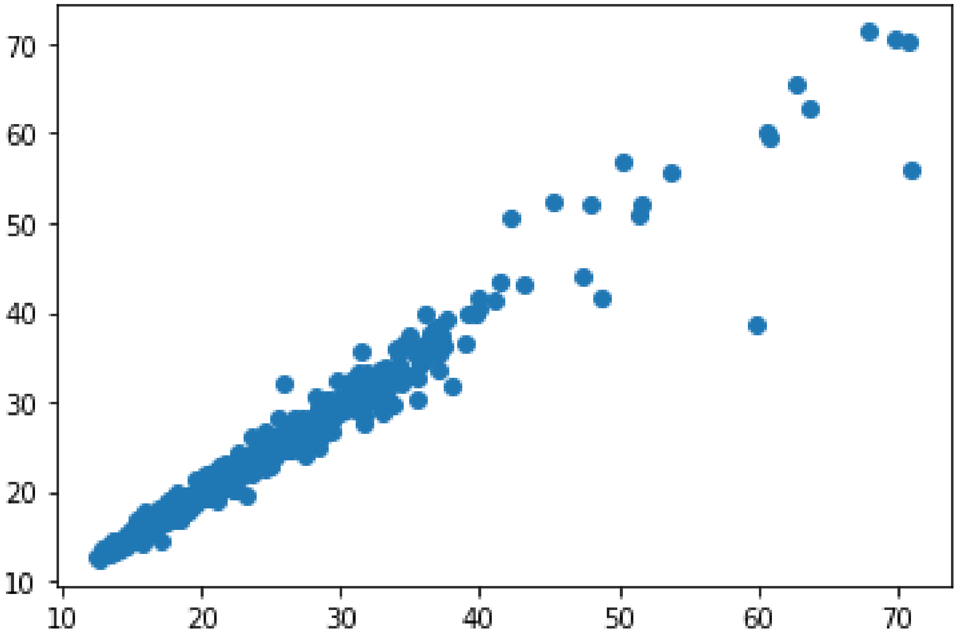
1. **Train a linear model that forecasts the next day's speed average** using your training dataset `X_train`, `y_train`. Specifically, predict $y_{(i,t)}$ from $X_{(i,t)}$, where

- $y_{(i,t)}$ is the daily speed average for day t and census tract i
- $X_{(i,t)}$ is a vector of daily speed averages for days $t-5, t-4, t-3, t-2, t-1$ for census tract i

2. **Evaluate your model** on your validation dataset `X_val`, `y_val`.
3. **Make a scatter plot**, plotting predicted averages against ground truth averages. Note the perfect model would line up all points along the line $y = x$.

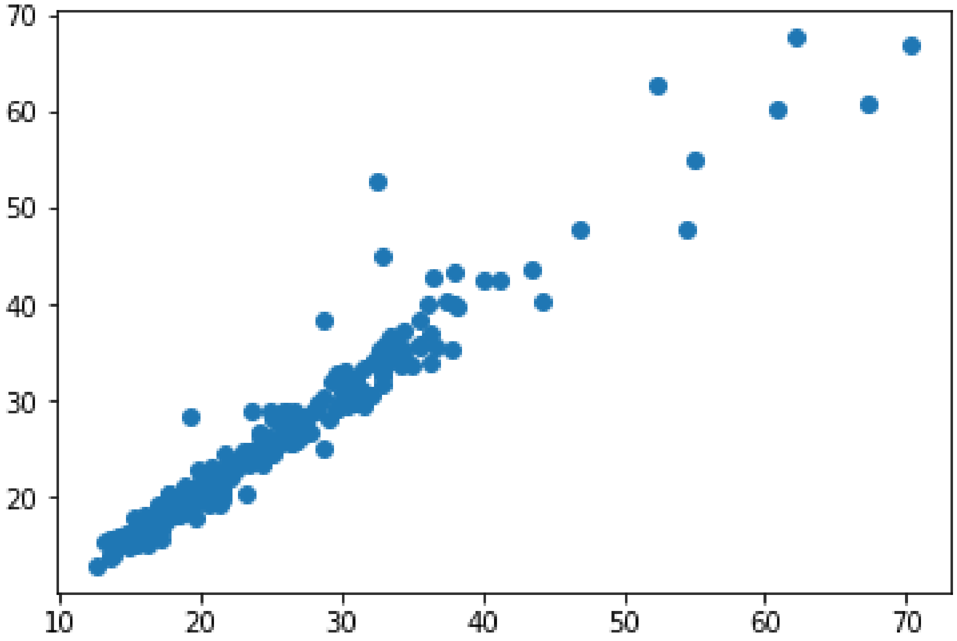
Our model is quantitatively and qualitatively pretty accurate at this point, training and evaluating on pre-lockdown data.

```
In [67]: lm = LinearRegression()
         reg = lm.fit(X_train, y_train) # set to trained linear model

         score = reg.score(X_val, y_val) # report r^2 score

         # create the scatter plot below
         Y_pred = reg.predict(X_val)
         plt.scatter(Y_pred, y_val);
```



Make scatter plot below.

```
In [74]: post_pred = reg.predict(time_series_x_pre)
         plt.scatter(post_pred, time_series_y_post);
```



0.2.9 4.b.ii. Report model performance temporally

1. **Make a line plot** showing performance of the original model throughout all of March 2020.
2. **Report the lowest point on the line plot**, reflecting the lowest model performance.
3. **Why is model performance the worst on the 17th?** Why does it begin to worsen on march 15th? And continue to worsen? Use what you know about covid measures on those dates. You may find this webpage useful: https://abc7news.com/timeline-of-coronavirus-us-covid-19-bay-area-sf/6047519/
4. **Is the dip in performance on the 9th foreshadowed** by any of our EDA?
5. **How does the model miraculously recover on its own?**
6. **Make a scatter plot**, plotting predicted averages against ground truth averages *for model predictions on March 17th.* Note the perfect model would line up all points along the line $y = x$. When compared against previous plots of this nature, this plot looks substantially worse, with points straying far from $y = x$.

Note: Answer questions 2-5 in the Markdown cell below. Q1 and Q6 are answered in the two code cells below.

2) The worst performance is on March 17th; there is a significant dip in the line plot.

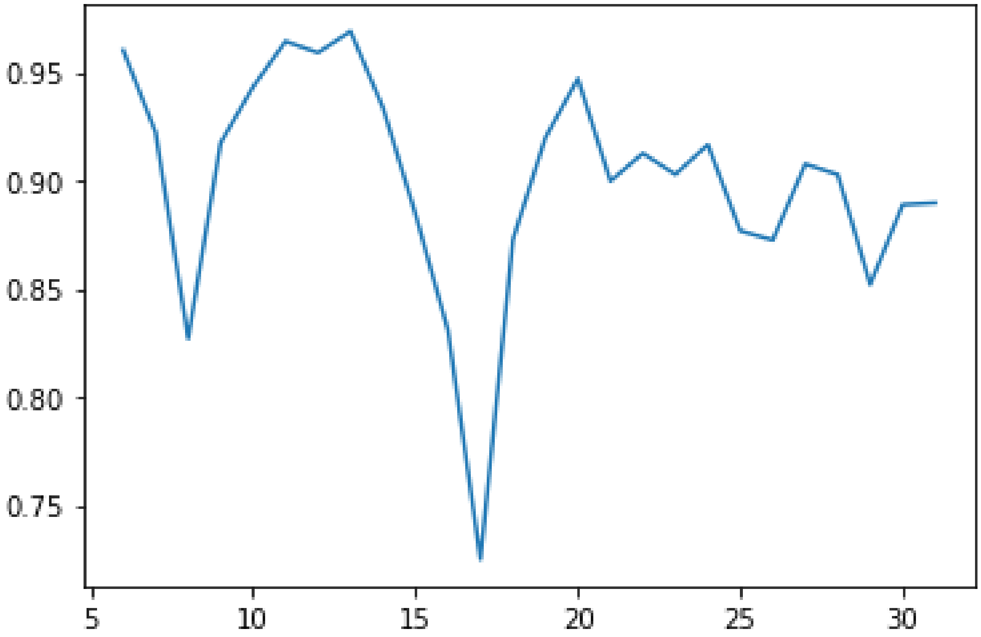
3) The model is a Linear Regression model that is trained on pre-lockdown data and predicts average speeds by tract based on the previous 5 days' average speeds. As such, the model won't do well if traffic suddenly changes significantly from the 5 previous days. Traffic patterns began changing on the 15th when the governor announced that all nightlife would be closing. Traffic then lessened dramatically on the 17th because that was when shelter-in-place orders took effect. Due to the traffic on the 15th and 17th being very different from the previous 5 days, the model's performance began to worsen on the 15th, then dramatically dipped on the 17th.

4) It is. If we go back and look at our line plot for Average Speeds by Day, we see that there is a spike on the 9th.

5) The model recovered because the model predicts average speeds on the average speeds from the previous 5 days. Once the lockdown began to take effect (from the 15th - 17th), our model began taking in post-lockdown data as inputs. As such, our model was able to recover and more accurately predict post-lockdown traffic.

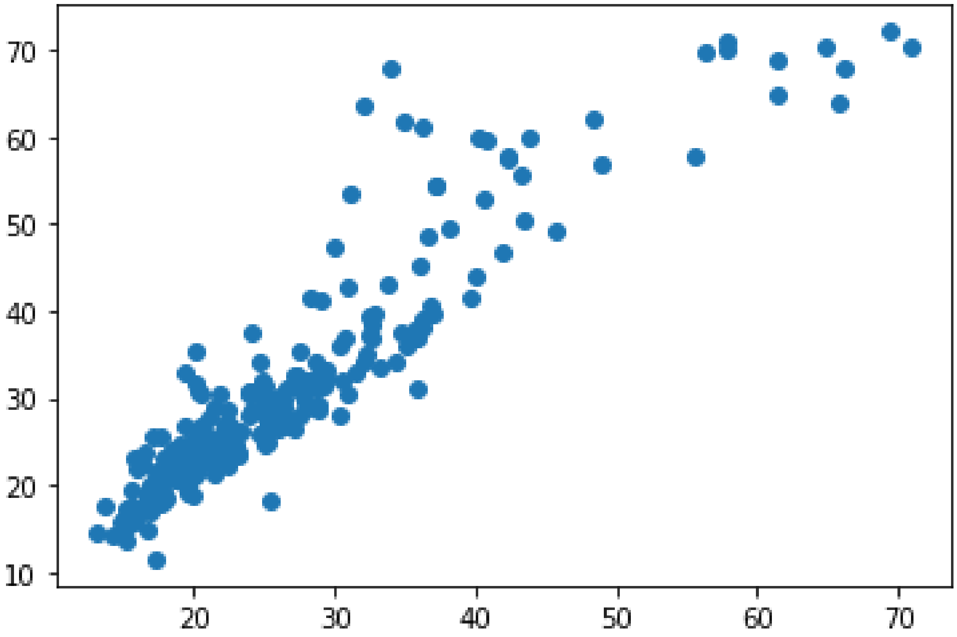
Generate line plot.

```
In [75]: scores = []
         for day in np.arange(1, 27):
             x_train = time_series_to_dataset(time_series[np.arange(day, day + 6)], 5, 0)[0] # get 'tim
             y_pred = time_series_to_dataset(time_series[np.arange(day, day + 6)], 5, 0)[1]
             scores.append(reg.score(x_train, y_pred))
         plt.plot(np.arange(6, 32), scores);
```



Generate a scatter plot.

```
In [76]: time_series_x_pre = time_series_to_dataset(time_series[np.arange(12, 18)], 5, 0)[0] # get 'tim
         Y_actual17 = time_series_to_dataset(time_series[np.arange(12, 18)], 5, 0)[1]
         Y_pred17 = reg.predict(time_series_x_pre)
         plt.scatter(Y_pred17, Y_actual17);
```



0.2.10 4.c.i. Learn delta off of a moving bias

According to our previous work in EDA, the average speed shoots upwards sharply. As a result, our trick to learn delta the around the average and to naively assume that the average of day t is the average for day $t + 1$. We will do this in 4 steps:

1. **Create a dataset for your delta model**.
2. **Train your delta model** on pre-lockdown data.
3. **Evaluate your model on pre-lockdown data**, to ensure that the model has learned to a satisfactory degree, in the nominal case. Remember the naive model achieved 0.97 r^2 on pre-lockdown data.
4. **Evaluate your model on the 17th**, to compare against the naive model also evaluated on that day. Notice that your r^2 score has improved by 10%+. Why is your delta model so effective for the 17th?
5. **Evaluate your model on the 14th**, to compare against the naive model also evaluated on that day. Notice that your r^2 score is now complete garbage. Why is your delta so ineffective for the 14th?

Hint: As you build your datasets, always check to make sure you're using the right days! It's easy to have a one-off error that throws off your results.

Write your written questions in the next cell, then write the code in the following cells.

4) Our delta model predicts average speeds by tract based on deltas (difference between actual sample value and a running average) over the past 5 days. By the 17th, traffic patterns had already begun changing a few days ago (as reflected in our Average Speeds by Day lineplot), and our deltas had already reflected that change. As such, on the 17th, if we take the 16th's average (which already reflects post-lockdown traffic changes), and then change that based on the specific census tract's delta, we should get a prediction that is more accurate than simply predicting based off the past 5 days' averages. Essentially, our delta model "learns" faster than the original one. However, the delta model is a lot more dependent on the previous day's average speed, since that's what we just take as our predicted day's average as well. This is reflected in the r^2 score for day 14th (when the lockdown began).