0.0.1 Question 0a

What is the granularity of the data (i.e. what does each row represent)?

The granularity of data is hour of the day.

0.0.2 Question 0b

For this assignment, we'll be using this data to study bike usage in Washington D.C. Based on the granularity and the variables present in the data, what might some limitations of using this data be? What are two additional data categories/variables that one could collect to address some of these limitations?

The data might only cover a certain period of time. It may hard to tell something about seasonal or long-term trends in bike usage. And also we don't know about geographic scope of the data. The datas is from certain area of Washington D.C. or the whole area.

The first additional data categories/variables I would to add is weather like 'Windy', which could identify seasonal trends and and determine the best times to promote bike usage in the city. The second additional data categories/variables I would to add is location which could show more detailed data about where bikes are being used in the city

0.0.3 Question 2a

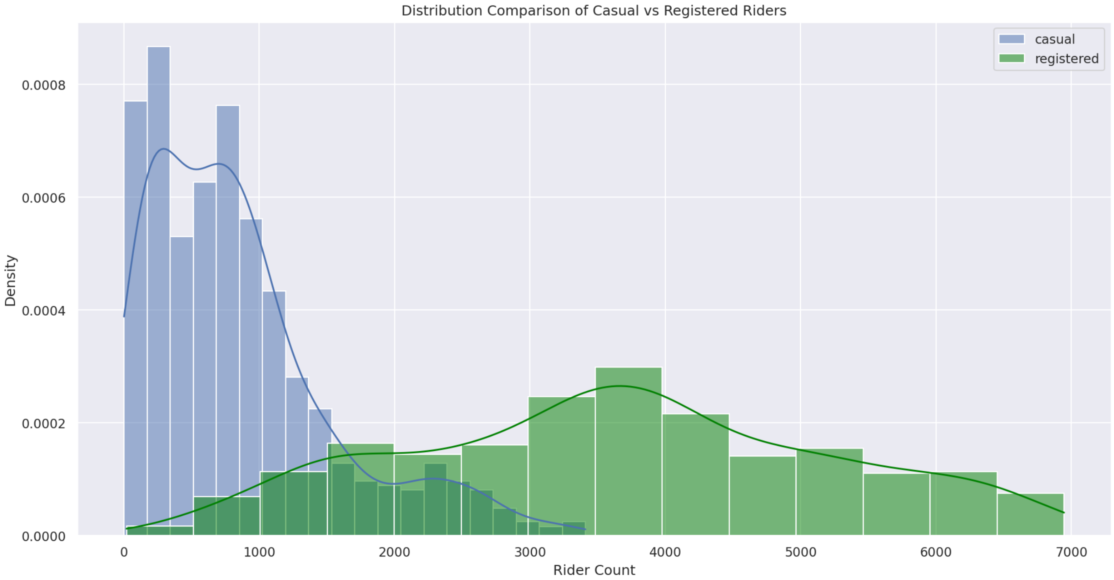
Use the `sns.histplot`(documentation) function to create a plot that overlays the distribution of the daily counts of bike users, using blue to represent `casual` riders, and green to represent `registered` riders. The temporal granularity of the records should be daily counts, which you should have after completing question 1c.

Hint: You will need to set the `stat` parameter appropriately to match the desired plot, and may call `sns.histplot` more than one time.

Include a `legend`, `xlabel`, `ylabel`, and `title`. Read the seaborn plotting tutorial, if you're not sure how to add these. After creating the plot, look at it and make sure you understand what the plot is actually telling us, e.g on a given day, the most likely number of registered riders we expect is ~4000, but it could be anywhere from nearly 0 to 7000.

```
In [35]: sns.histplot(data=daily_counts, x = daily_counts['casual'], kde = True, stat = 'density', labe
         sns.histplot(data=daily_counts, x = daily_counts['registered'], kde = True, color = 'green', s
         plt.title("Distribution Comparison of Casual vs Registered Riders")
         plt.xlabel("Rider Count")
         plt.ylabel("Density")
         plt.legend()
```

```
Out[35]: <matplotlib.legend.Legend at 0x7fd38e6fa400>
```



0.0.4 Question 2b

In the cell below, descibe the differences you notice between the density curves for casual and registered riders. Consider concepts such as modes, symmetry, skewness, tails, gaps and outliers. Include a comment on the spread of the distributions.

The distribution of registered riders is more symmetric distribution centered abourd 3500. It spread put widely from 0 - 7000. The casual riders distribution is skew to left. The peak is around 500. There is a long right tail, sharp dorps from around 850. It is hard to see if there still a value after 5000.

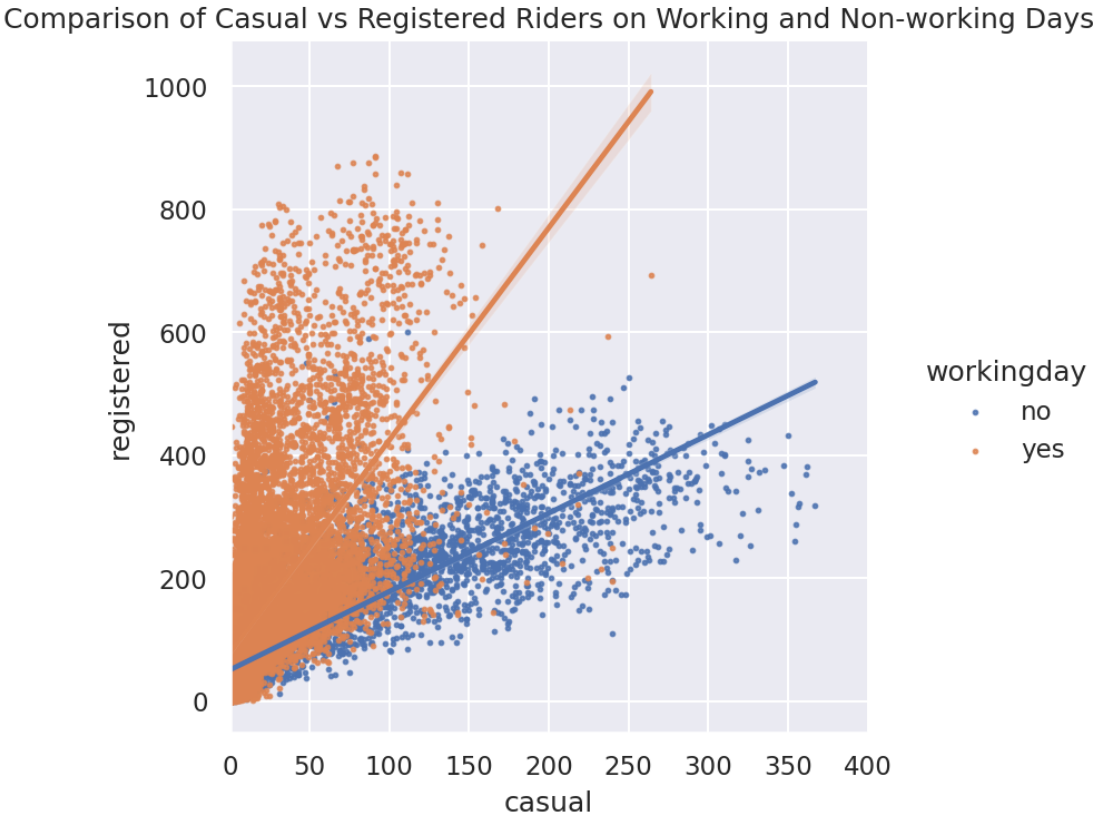
0.0.5 Question 2c

The density plots do not show us how the counts for registered and casual riders vary together. Use `sns.lmplot` (documentation) to make a scatter plot to investigate the relationship between casual and registered counts. This time, let's use the `bike DataFrame` to plot hourly counts instead of daily counts.

The `lmplot` function will also try to draw a linear regression line (just as you saw in Data 8). Color the points in the scatterplot according to whether or not the day is a working day (your colors do not have to match ours exactly, but they should be different based on whether the day is a working day).

Hints: * Checkout this helpful tutorial on `lmplot`. * There are many points in the scatter plot, so make them small to help reduce overplotting. Check out the `scatter_kws` parameter of `lmplot`. * Generate and plot the linear regression line by setting a **paramter** of `lmplot` to `True`. Can you find this in the documentation? We will discuss what is linear regression is more details later. * You can set the `height` parameter if you want to adjust the size of the `lmplot`. * Add a descriptive title and axis labels for your plot.

```
In [36]: # Make the font size a bit bigger
         sns.set(font_scale=1)
         sns.lmplot(data=bike, x = 'casual', y = "registered", hue="workingday", scatter_kws={"s": 3})
         plt.xlim(0, 400)
         plt.title("Comparison of Casual vs Registered Riders on Working and Non-working Days");
```



Comparison of Casual vs Registered Riders on Working and Non-working Days
registered
1000
800
600
400
200
0
0
50
100
150
200
250
300
350
400
casual
workingday
● no
● yes

0.0.6 Question 2d

What does this scatterplot seem to reveal about the relationship (if any) between casual and registered riders and whether or not the day is on the weekend? What effect does overplotting have on your ability to describe this relationship?

There is a linear relationship between casual and registered riders. Maybe people just don't need to go to work on weekend(for registered). So if a day is a working day is important.

We could not distinguish the dots in overploted part. It is hard to see.

0.0.7 Question 3a (Bivariate Kernel Density Plot)

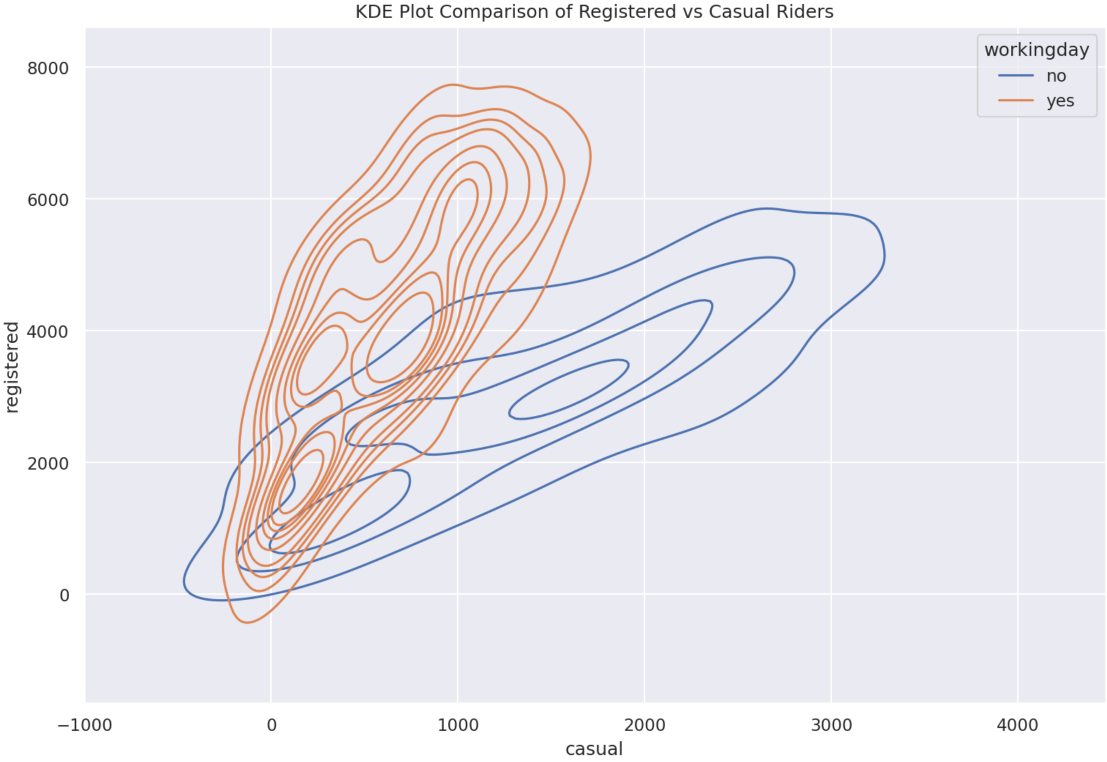
Generating a bivariate kernel density plot with workday and non-workday separated.

Hints: You only need to call `sns.kdeplot` once. Take a look at the `hue` paramter and adjust other inputs as needed.

After you get your plot working, experiment by setting `shade=True` in `kdeplot` to see the difference between the shaded and unshaded version. Please submit your work with `shade=False`.

```
In [38]: # Set the figure size for the plot
         plt.figure(figsize=(12,8))
         sns.kdeplot(data = daily_counts, x='casual', y='registered', hue="workingday", shade=False)
         plt.title('KDE Plot Comparison of Registered vs Casual Riders')
```

```
Out[38]: Text(0.5, 1.0, 'KDE Plot Comparison of Registered vs Casual Riders')
```



0.0.8 Question 3b

With some modification to your 3a code (this modification is not in scope), we can obatined the plot above. In your own words, describe what the lines and the color shades of the lines signify about the data. What does each line and color represent?

Shades represent the level of dencity. If the area is darker, it means there are denser data count. The lines represent difference dencity of value.

0.0.9 Question 3c

What additional details can you identify from this contour plot that were difficult to determine from the scatter plot?

It is easy to see the average(which area is darker), which is hard to see in scatter plot.

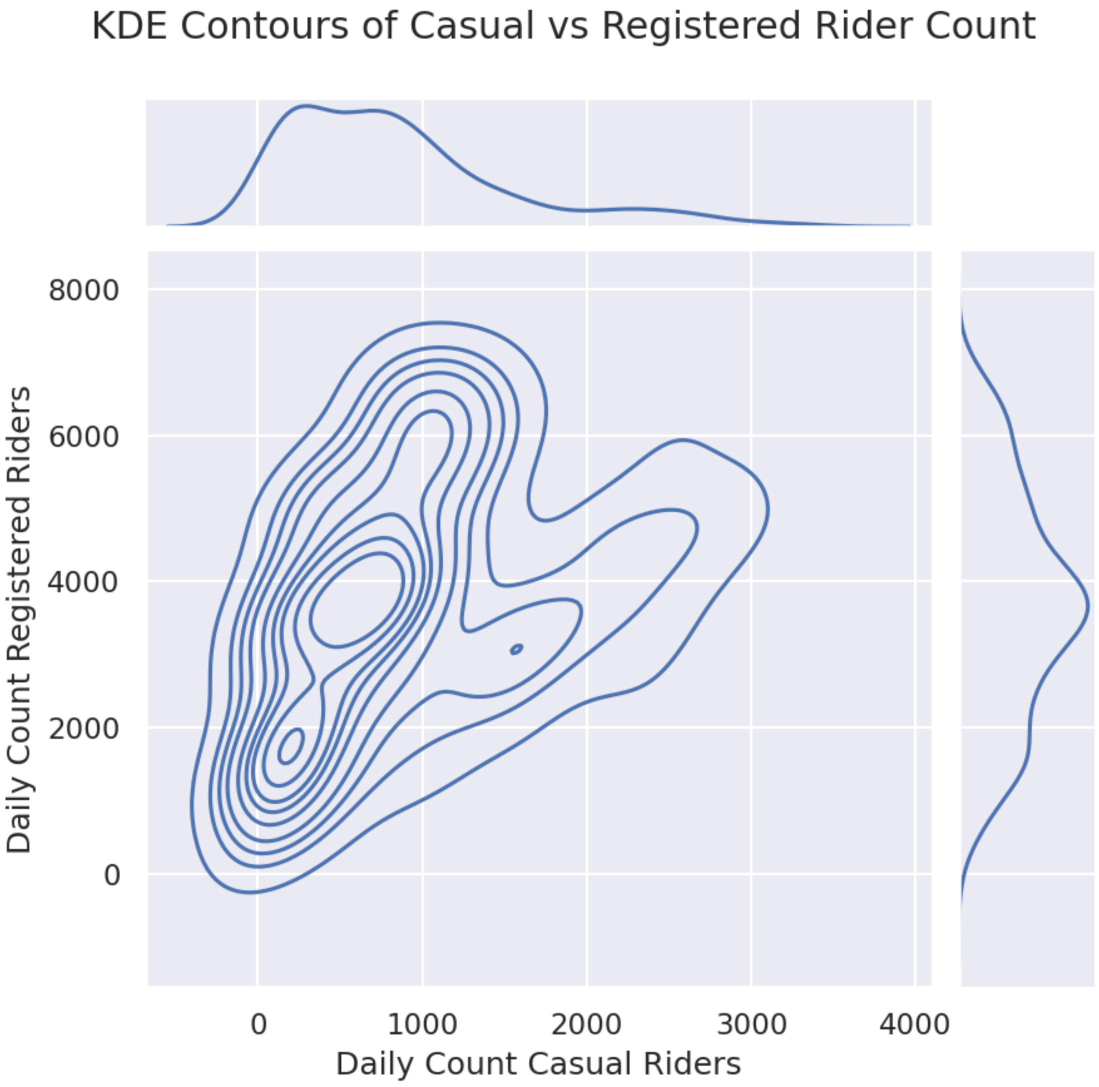
0.1 4: Joint Plot

As an alternative approach to visualizing the data, construct the following set of three plots where the main plot shows the contours of the kernel density estimate of daily counts for registered and casual riders plotted together, and the two "margin" plots (at the top and right of the figure) provide the univariate kernel density estimate of each of these variables. Note that this plot makes it harder see the linear relationships between casual and registered for the two different conditions (weekday vs. weekend).

Hints: * The seaborn plotting tutorial has examples that may be helpful. * Take a look at `sns.jointplot` and its `kind` parameter. * `set_axis_labels` can be used to rename axes on the contour plot.

Note: * At the end of the cell, we called `plt.suptitle` to set a custom location for the title. * We also called `plt.subplots_adjust(top=0.9)` in case your title overlaps with your plot.

```
In [39]: sns.jointplot(data=daily_counts, x="casual", y="registered", kind="kde").set_axis_labels("Daily
         plt.suptitle("KDE Contours of Casual vs Registered Rider Count")
         plt.subplots_adjust(top=0.9);
```


KDE Contours of Casual vs Registered Rider Count
8000
6000
4000
2000
0
Daily Count Registered Riders
0
1000
2000
3000
4000
Daily Count Casual Riders

0.2 5: Understanding Daily Patterns

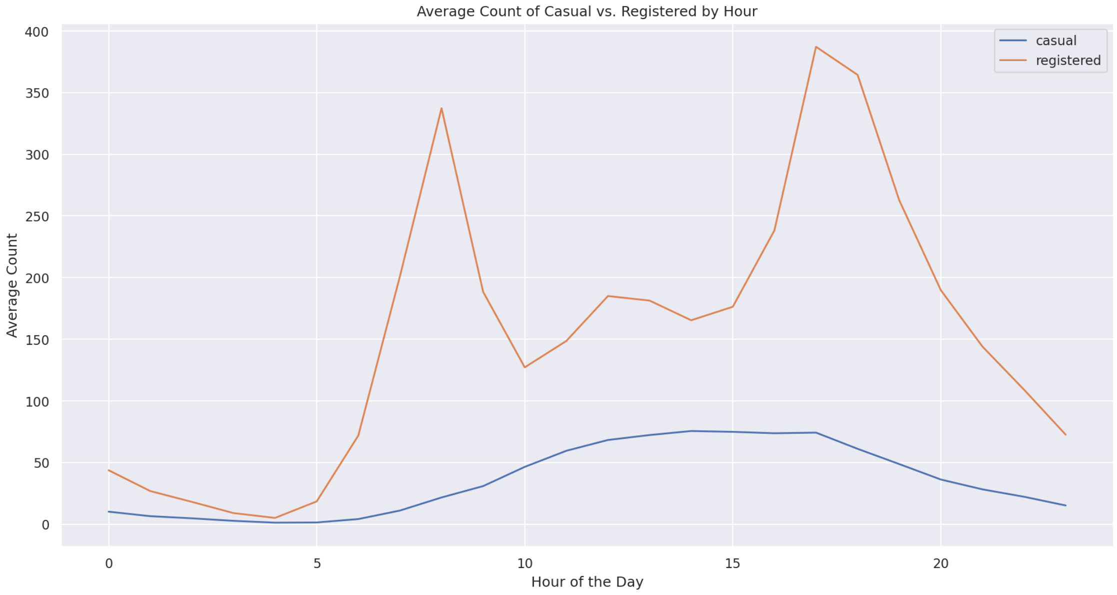
0.2.1 Question 5a

Let's examine the behavior of riders by plotting the average number of riders for each hour of the day over the **entire dataset**, stratified by rider type.

Your plot should look like the plot below. While we don't expect your plot's colors to match ours exactly, your plot should have legend in the plot and different colored lines for different kinds of riders.

```
In [40]: hour_visual = bike.groupby('hr').mean()

         sns.lineplot(data = hour_visual, x=np.arange(24), y = 'casual', label = 'casual')
         sns.lineplot(data = hour_visual, x=np.arange(24), y = 'registered', label = 'registered')
         plt.xlabel('Hour of the Day')
         plt.ylabel('Average Count')
         plt.title('Average Count of Casual vs. Registered by Hour');
```



0.2.2 Question 5b

What can you observe from the plot? Discuss your obseravtion and hypothesize about the meaning of the peaks in the registered riders' distribution.

For registered client, thy minly go to work (Or say peak of rides) around 6am - 10am and get out of work around 15pm - 20pm.
The casual riders ride throughout all day and peak hours are around mid-afternoon.

0.2.3 Question 6b

In our case with the bike ridership data, we want 7 curves, one for each day of the week. The x-axis will be the temperature and the y-axis will be a smoothed version of the proportion of casual riders.

You should use `statsmodels.nonparametric.smoothers_lowess.lowess` just like the example above. Unlike the example above, plot ONLY the lowess curve. Do not plot the actual data, which would result in overplotting. For this problem, the simplest way is to use a loop.

You do not need to match the colors on our sample plot as long as the colors in your plot make it easy to distinguish which day they represent.

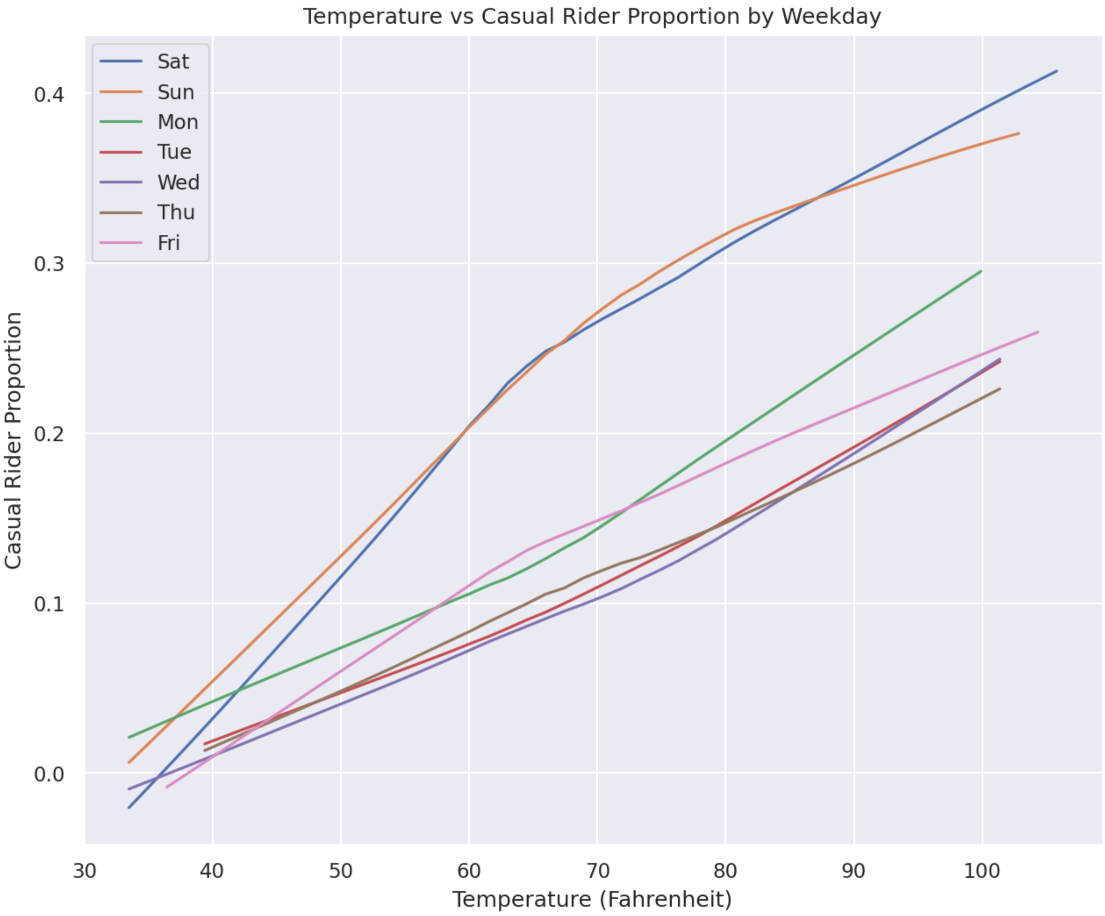
Hints: * Start by just plotting only one day of the week to make sure you can do that first.

- The `lowess` function expects y coordinate first, then x coordinate. You should also set the `return_sorted` field to `False`.
- Look at the top of this homework notebook for a description of the (normalized) temperature field to know how to convert to Fahrenheit. By default, the temperature field ranges from 0.0 to 1.0. In case you need it, Fahrenheit = Celsius × $\frac{9}{5}$ + 32.

Note: If you prefer plotting temperatures in Celsius, that's fine as well!

```
In [46]: from statsmodels.nonparametric.smoothers_lowess import lowess
         plt.figure(figsize=(10,8))
         plt.xlabel("Temperature (Fahrenheit)")
         plt.ylabel("Casual Rider Proportion")
         plt.title("Temperature vs Casual Rider Proportion by Weekday")

         bike['tempFahr'] = bike['temp'] * 41 * 9 / 5 + 32
         for d in bike['weekday'].unique():
             currentDayInLoop = bike[bike['weekday'] == d]
             lows = lowess(currentDayInLoop['prop_casual'], currentDayInLoop['tempFahr'], return_sorted
             sns.lineplot(currentDayInLoop['tempFahr'], lows, label=d)
```


Temperature vs Casual Rider Proportion by Weekday
Sat
Sun
Mon
Tue
Wed
Thu
Fri
Casual Rider Proportion
0.0
0.1
0.2
0.3
0.4
Temperature (Fahrenheit)
30
40
50
60
70
80
90
100

0.2.4 Question 6c

What do you see from the curve plot? How is `prop_casual` changing as a function of temperature? Do you notice anything else interesting?

As temperature increase, the prop_casual also increse.

Interesting thing is the prop_casual is still increse when the temperature goes very high, like from 90 to 100. On Saterday the trend still increasing after 100 Fahrenhei. And also the prop_casual on weekend is heigher than the weed days.

0.2.5 Question 7a

Imagine you are working for a Bike Sharing Company that collaborates with city planners, transportation agencies, and policy makers in order to implement bike sharing in a city. These stakeholders would like to reduce congestion and lower transportation costs. They also want to ensure the bike sharing program is implemented equitably. In this sense, equity is a social value that is informing the deployment and assessment of your bike sharing technology.

Equity in transportation includes: improving the ability of people of different socio-economic classes, genders, races, and neighborhoods to access and afford the transportation services, and assessing how inclusive transportation systems are over time.

Do you think the `bike` data as it is can help you assess equity? If so, please explain. If not, how would you change the dataset? You may discuss how you would change the granularity, what other kinds of variables you'd introduce to it, or anything else that might help you answer this question.

I don't think bike data will help us to assess equity. We basically only get time and temperature directions information. We still need information about people like income, age, genders, races, what block the user live, normal way for transportations. We also need information about location like which block of the city.

0.2.6 Question 7b

Bike sharing is growing in popularity and new cities and regions are making efforts to implement bike sharing systems that complement their other transportation offerings. The goals of these efforts are to have bike sharing serve as an alternate form of transportation in order to alleviate congestion, provide geographic connectivity, reduce carbon emissions, and promote inclusion among communities.

Bike sharing systems have spread to many cities across the country. The company you work for asks you to determine the feasibility of expanding bike sharing to additional cities of the U.S.

Based on your plots in this assignment, what would you recommend and why? Please list at least two reasons why, and mention which plot(s) you drew you analysis from.

Note: There isn't a set right or wrong answer for this question, feel free to come up with your own conclusions based on evidence from your plots!

According to 5a and 6b, I would recommend that prepare more bike during 5- 10 am and 15-20 pm, and allday for weekend. Because data shows there is a heigh demend during those time period. And it the demend will not goes low when temperature goes heigh. So just be sure there are enough bick there when people needed.