
0.0.1 Question 1a

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

Hint: Examine all variables present in the dataset carefully before answering this question! Pay special attention to the time-based columns.

Each row represents an hour for which the number of bikes rented by registered and casual users are recorded. Information about weather (temperature, wind, conditions), what day of the week it is, and whether or not this day is a holiday, a weekend, or a working day is also included. The actual date, season, year, and month from which the hour is from is also recorded.

0.0.2 Question 1b

For this assignment, we'll be using this data to study bike usage in Washington, DC. 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?

1. While there is a `dteday` variable to encode the full date, there are also separate columns for `yr`, `mnth`, and `hr`. However, we are missing the corresponding `day` column, which may make it difficult to filter for certain dates across the year (for example, searching for information from the first day of every month). **I would add another variable to collect the `day` of the data point**
2. Additionally, while there is information for every hour and what season the hour is from, it is difficult to determine whether or not there is still sunlight present during that hour (sunrise and sunset hours vary across the year). Therefore, **I would add another column `sun` which stores a boolean corresponding to whether the sun is still out for this hour**. This would make it easier to determine the number of rentals that occur during the day or during the night.

0.0.3 Question 3a

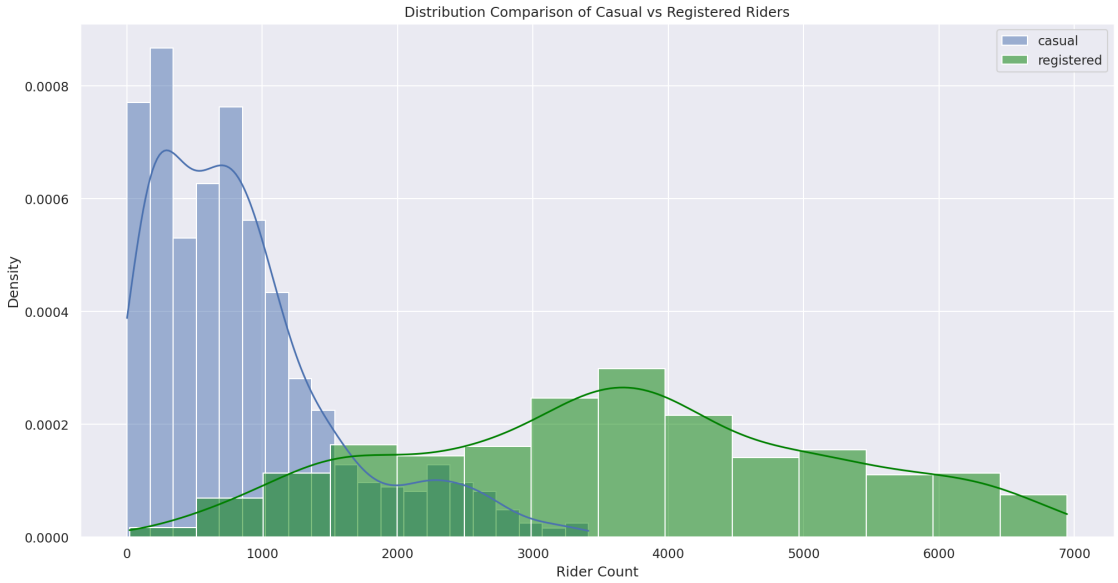
Use the `sns.histplot`(documentation) function to create a plot that overlays the distribution of the daily counts of bike users, using blue to represent `casual` riders, and green to represent `registered` riders. The temporal granularity of the records should be daily counts, which you should have after completing question 2.c. In other words, you should be using `daily_counts` to answer this question.

Hints: - You will need to set the `stat` parameter appropriately to match the desired plot. - The `label` parameter of `sns.histplot` allows you to specify, as a string, how the plot should be labeled in the legend. For example, passing in `label="My data"` would give your plot the label "My data" in the legend. - You will need to make two calls to `sns.histplot`.

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.

For all visualizations in Data 100, our grading team will evaluate your plot based on its similarity to the provided example. While your plot does not need to be *identical* to the example shown, we do expect it to capture its main features, such as the **general shape of the distribution**, the **axis labels**, the **legend**, and the **title**. It is okay if your plot contains small stylistic differences, such as differences in color, line weight, font, or size/scale.

```
In [15]: sns.histplot(data=daily_counts['casual'], stat='density', kde=True, legend=True, label='casual
         sns.histplot(data=daily_counts['registered'], stat='density', kde=True, color='green', legend='
         plt.xlabel('Rider Count')
         plt.legend()
         plt.title('Distribution Comparison of Casual vs Registered Riders');
```


Distribution Comparison of Casual vs Registered Riders
casual
registered
Density
0.0000
0.0002
0.0004
0.0006
0.0008
0
1000
2000
3000
4000
5000
6000
7000
Rider Count

0.0.4 Question 3b

In the cell below, describe 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.

For the casual riders, the distribution has a right tail, meaning it is right-skewed. Meanwhile, the registered riders distribution is more symmetric and is more akin to a normal distribution. The modes of the registered riders are 1707, 4841, and 6248, all of which are far greater than the modes of the casual riders: 120, 968. This makes sense as the registered riders are normally distributed around a center of ~3800, whereas the casual riders distribution is right skewed.

The spread of the registered rider counts is wider, as there are rider counts from 0 up to ~7000. The unregistered rider counts are not as spread out, as the distribution is right skewed with its data points mainly concentrated in the sub 1000 counts, with a few counts reaching a maximum of the ~3500. This is reflected in their standard deviations, as the registered riders distribution has a much larger standard devia

```
In [16]: print(daily_counts['registered'].mode())
         daily_counts['registered'].std()
```

```
0    1707
1    4841
2    6248
Name: registered, dtype: int64
```

```
Out[16]: 1560.2563770194536
```

```
In [17]: print(daily_counts['casual'].mode())
         daily_counts['casual'].std()
```

```
0    120
1    968
Name: casual, dtype: int64
```

```
Out[17]: 686.622488284655
```

0.0.5 Question 3c

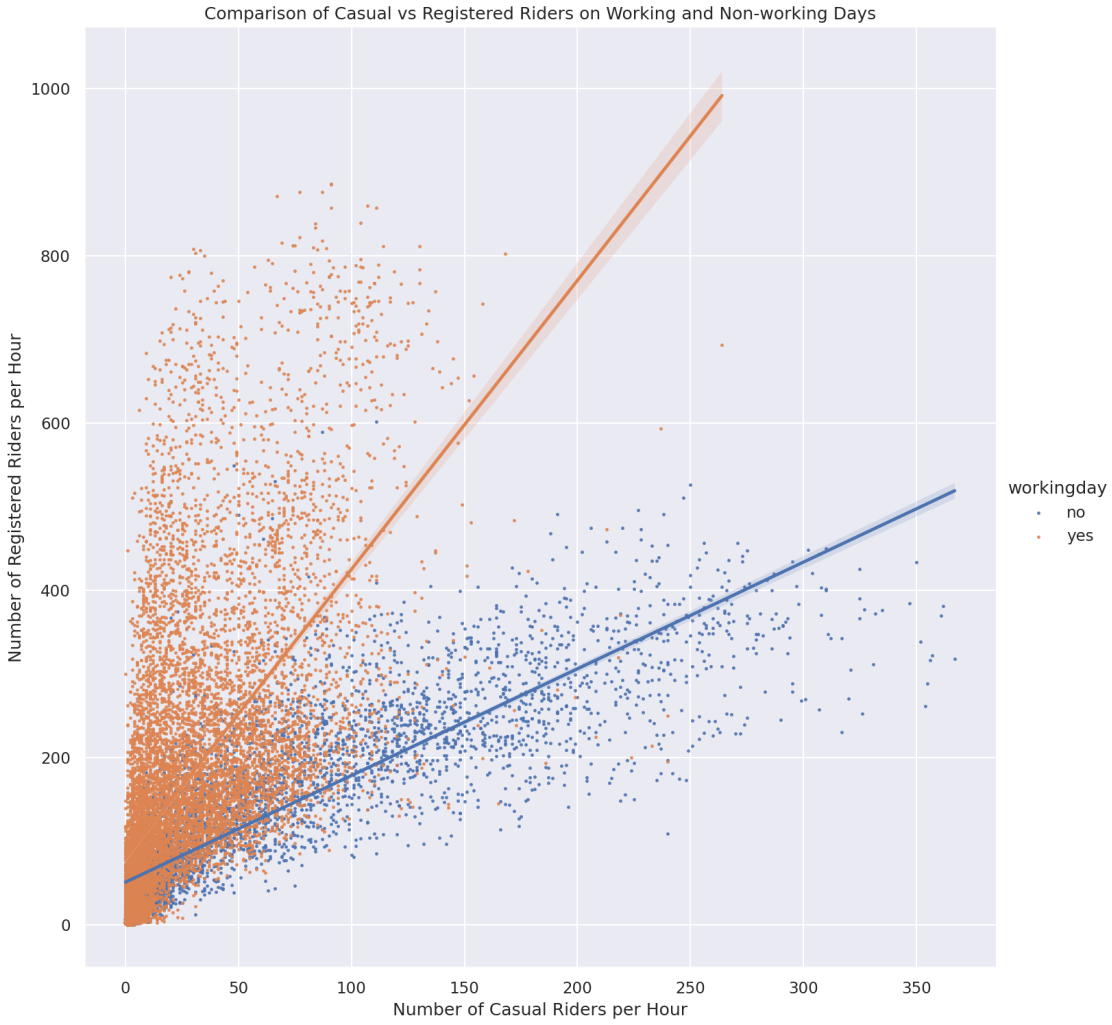
The density plots do not show us how the counts for `registered` and `casual` riders vary together. Use `sns.lmplot` (documentation) to make a scatter plot to investigate the relationship between casual and registered counts. This time, let's use the `bike DataFrame` to plot hourly counts instead of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

The `lmplot` function will also try and 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: * Check out 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`. * 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. * You should be using the `bike DataFrame` to create your plot. * It is okay if the scales of your `x` and `y` axis (i.e., the numbers labeled on the two axes) are different from those used in the provided example.

In [18]:
```
sns.set(font_scale=1) # This line automatically makes the font size a bit bigger on the plot.
sns.lmplot(data=bike, x="casual", y="registered", hue='workingday', scatter_kws={'s':2}, heigh
plt.title('Comparison of Casual vs Registered Riders on Working and Non-working Days')
plt.xlabel('Number of Casual Riders per Hour')
plt.ylabel('Number of Registered Riders per Hour')
```

Out[18]:
```
Text(66.00877083333337, 0.5, 'Number of Registered Riders per Hour')
```


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

0.0.6 Question 3d

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?

If the day is a workday, the ratio of registered riders to casual riders is greater. If the day is not a workday, that ratio decreases; in otherwords, on non-workdays, there are less registered riders for every casual rider per hour, as indicated by a lesser slope of the linear regression for working days compared to non-working days of casual versus registered riders.

Due to overplotting at low numbers of riders per hour (< 50) for both casual and registered riders, it is difficult to see if the relationship or the linear regression holds for low numbers of riders.

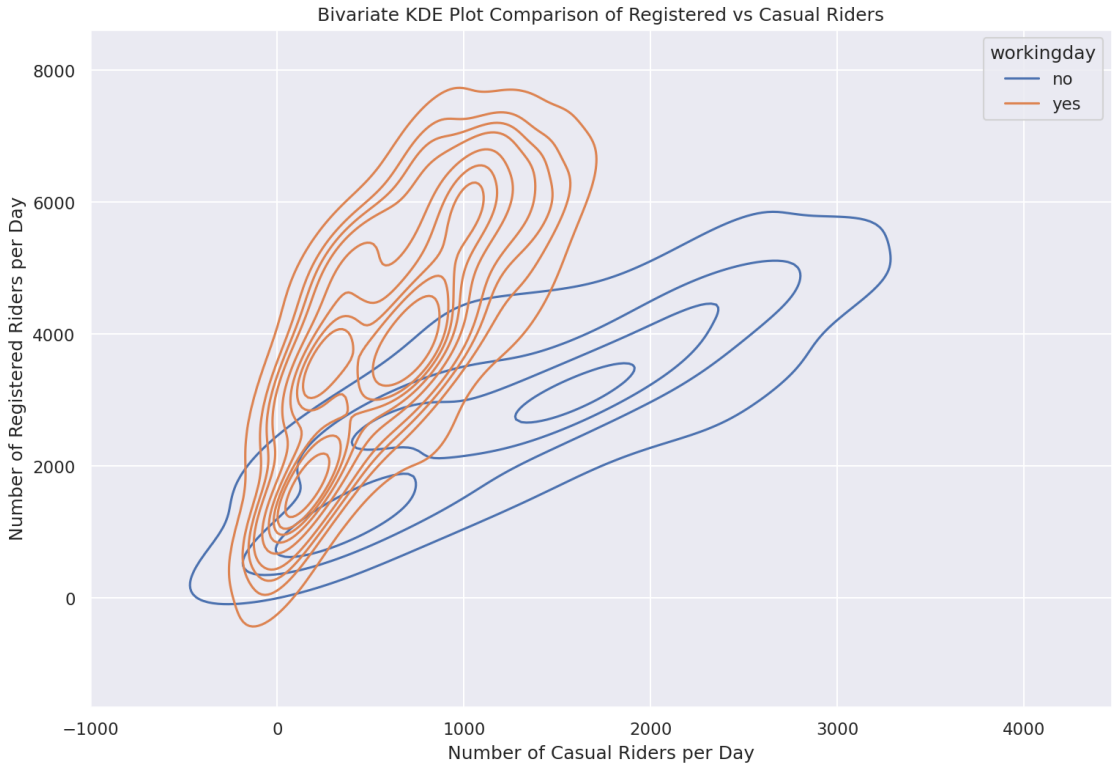
0.0.7 Question 4a (Bivariate Kernel Density Plot)

Generate a bivariate kernel density plot with workday and non-workday separated using the `daily_counts` `DataFrame`.

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

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

```
In [20]: # Set the figure size for the plot
         plt.figure(figsize=(12,8))
         sns.kdeplot(data=daily_counts, x='casual', y='registered', hue='workingday', fill=False)
         plt.title('Bivariate KDE Plot Comparison of Registered vs Casual Riders')
         plt.xlabel('Number of Casual Riders per Day')
         plt.ylabel('Number of Registered Riders per Day');
```



0.0.8 Question 4b

With some modification to your 4a code (this modification is not in scope), we can generate 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?

Hint: You may find it helpful to compare it to a contour or topographical map as shown here.

The color shades signify the density of the data; in other words, how many data points exist within that range. The lines delineate the different subgroups of data points; in other words, the lines connect data points that have the same densities.

0.0.9 Question 4c

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

On Non-Workdays, there are a lot of data points for days where about 1000 to 2000 casual riders will rent bikes, and about 2000-4000 registered riders will rent bikes. There are also a lot of data points for days on non-working days where very few casual riders will rent bikes (<500), but still ~1000 registered riders will rent bikes. This contour plot thus gives us more information about the overall trends between the registered and casual riders for the whole of non-working versus working days.

In [21]:
```
# sns.set(font_scale=1) # This line automatically makes the font size a bit bigger on the plot
# sns.lmplot(data=daily_counts, x="casual", y="registered", hue='workingday', scatter_kws={'s'
# plt.title('Comparison of Casual vs Registered Riders on Working and Non-working Days')
# plt.xlabel('Number of Casual Riders per Day')
# plt.ylabel('Number of Registered Riders per Day')
```

0.1 5: 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 to see the linear relationships between casual and registered for the two different conditions (weekday vs. weekend). You should be making use of `daily_counts`.

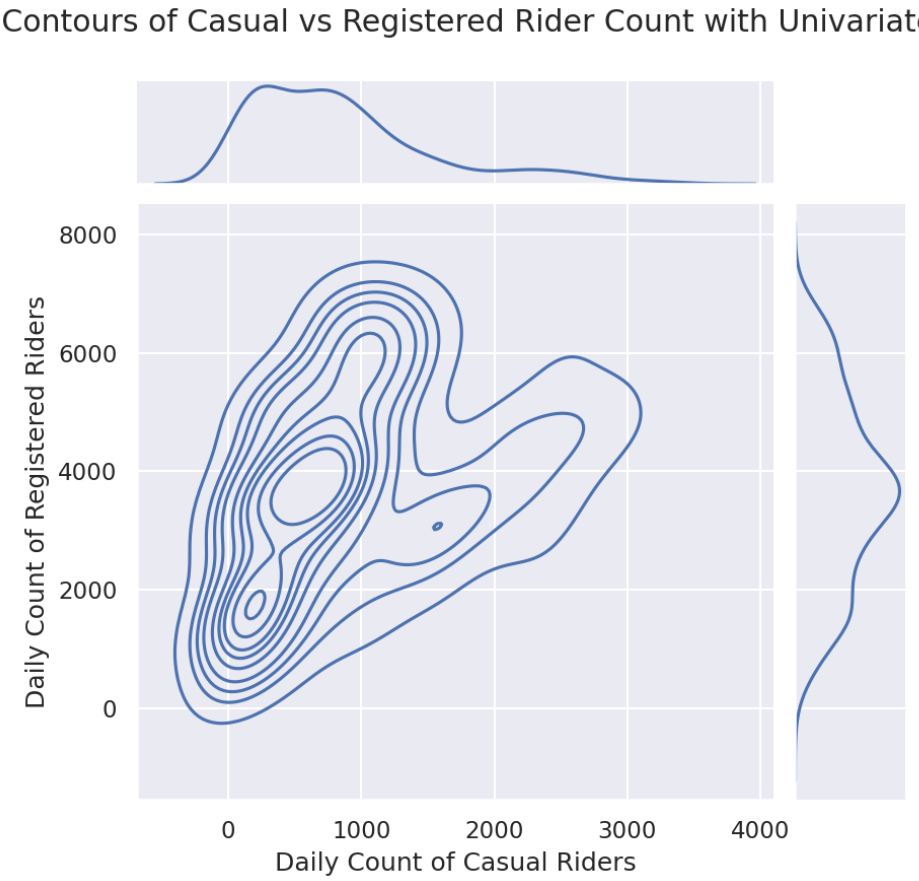
Hints: * The seaborn plotting tutorial has examples that may be helpful. * Take a look at `sns.jointplot` and its `kind` parameter. * `set_axis_labels` can be used to rename axes on a `seaborn` plot. For example, if we wanted to plot a scatterplot with 'Height' on the x-axis and 'Weight' on the y-axis from some dataset `stats_df`, we could write the following:

```
graph = sns.scatterplot(data=stats_df, x='Height', y='Weight')

graph.set_axis_labels("Height (cm)", "Weight (kg)")
```

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


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

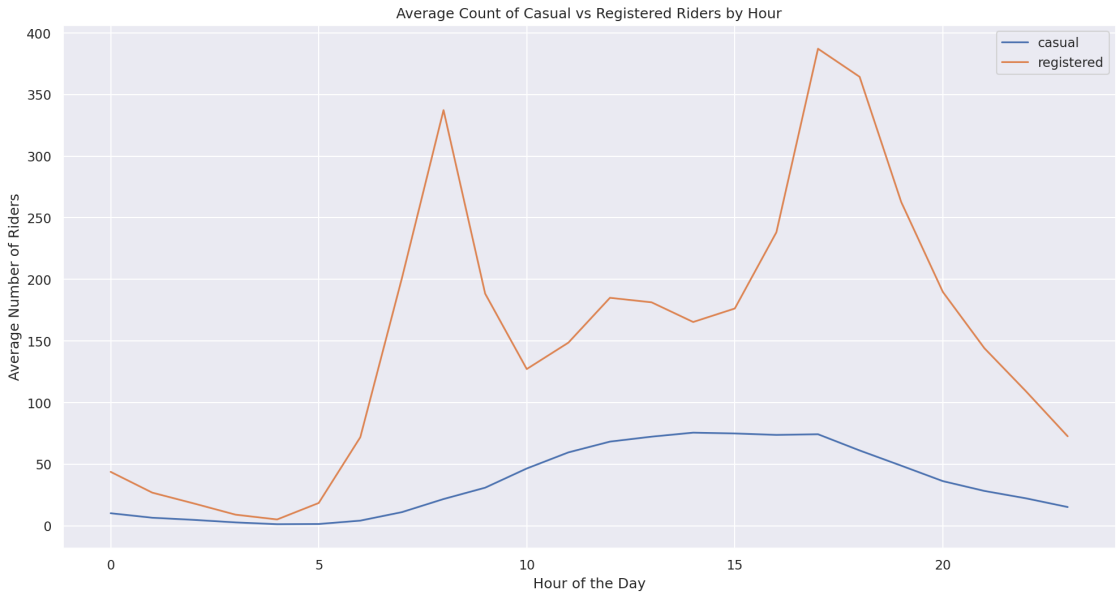
0.2 6: Understanding Daily Patterns

0.2.1 Question 6a

Let's examine the behavior of riders by plotting the average number of riders for each hour of the day over the **entire dataset** (that is, `bike DataFrame`), 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 a legend in the plot and different colored lines for different kinds of riders, in addition to the title and axis labels.

```
In [23]: per_hour = (bike[['hr', 'casual', 'registered']]
                     .groupby('hr')
                      .mean()
                     )
         # per_hour.head()
         sns.lineplot(per_hour, x=per_hour.index, y='casual', label='casual')
         sns.lineplot(per_hour, x=per_hour.index, y='registered', label='registered')
         plt.title("Average Count of Casual vs Registered Riders by Hour")
         plt.xlabel("Hour of the Day")
         plt.ylabel("Average Number of Riders");
```



0.2.2 Question 6b

What can you observe from the plot? Discuss your observations for both types of riders, and hypothesize about the meaning of the peaks in the registered riders' distribution.

For casual riders, the peak of riders is between 10 hours and about 17 hours, or between 10:00AM and 5:00PM, with a relatively steep descent after 5PM. This unimodal distribution would make sense if most casual riders ride on non-workdays and ride for leisure or travel; if this is the case, most likely they would be out during the daytime hours (10-5PM is a good bet) and would rent a bike for leisure during those times. Most likely they are not out and about for leisure too early (before 9AM) or too late when it gets dark (past 5PM). The peak is not as steep/defined because casual riders are more likely to bike throughout the whole day, not necessarily at specific hours.

For registered riders, the distribution is more bimodal, or perhaps even multimodal with a small peak around 12:00. This makes more sense if registered riders, who more routinely ride, would have schedules where they consistently ride. Thus, they would ride at the same times, resulting in these peaks. Additionally, riders may have groups that they ride together with at the same time. Finally, registered riders would need to have consistent schedules to work around their workdays, so they would ride early in the mornings (the peak around 7:00), during their lunch break (the small peak around 12:00), or after work (the peak around 17:00).

0.2.3 Question 7b

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 (as given in the `'temp'` column), 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 plotting only one day of the week to make sure you can do that first. Then, consider using a `for` loop to repeat this plotting operation for all days of the week.

- The `lowess` function expects the `y` coordinate first, then the `x` coordinate. You should also set the `return_sorted` field to `False`.
- **You will need to rescale the normalized temperatures stored in this dataset to Fahrenheit values.** Look at the section of this notebook titled 'Loading Bike Sharing Data' for a description of the (normalized) temperature field to know how to convert back to Celsius first. After doing so, convert it to Fahrenheit. By default, the temperature field ranges from 0.0 to 1.0. In case you need it, Fahrenheit = Celsius $\times \frac{9}{5} + 32$.

Note: If you prefer plotting temperatures in Celsius, that's fine as well! Just remember to convert accordingly so the graph is still interpretable.

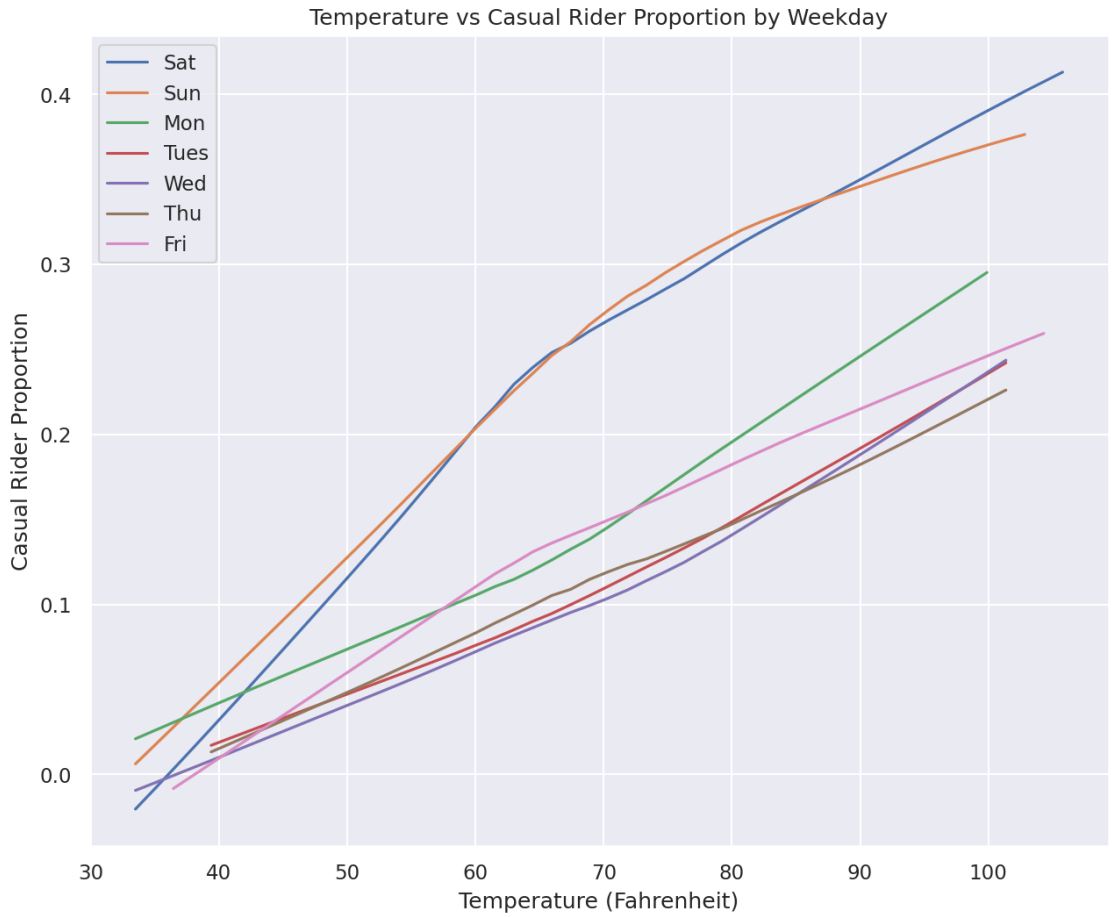
In [39]:
```
from statsmodels.nonparametric.smoothers_lowess import lowess

plt.figure(figsize=(10,8))
temp = bike['temp']
prop_casual = bike['prop_casual']
prop_casual_smoothed = lowess(prop_casual, temp, return_sorted=False)


weekday_list = ['Sat','Sun','Mon','Tues','Wed','Thu','Fri']
for day in weekday_list:
    day_df = bike[bike['weekday']==day]
    temp = (day_df['temp'] * 41) * (9/5) + 32
    prop_casual_smoothed = lowess(day_df['prop_casual'], temp, return_sorted=False)
    sns.lineplot(x=temp, y=prop_casual_smoothed, label=day)
```

```
plt.title('Temperature vs Casual Rider Proportion by Weekday')
plt.xlabel('Temperature (Fahrenheit)')
plt.ylabel('Casual Rider Proportion')
```

```
Out[39]: Text(0, 0.5, 'Casual Rider Proportion')
```



0.2.4 Question 7c

What do you observe in the above plot? How is `prop_casual` changing as a function of temperature? Do you notice anything else interesting?

As the temperature increases, the `prop_casual` also increases. Additionally, this trend is more apparent especially on the weekends (Saturday and Sunday). The slope of the temperature vs casual proportion line is much higher for Saturday and Sunday versus for the weekdays.

0.2.5 Question 8a

Imagine you are working for a bike-sharing company that collaborates with city planners, transportation agencies, and policymakers 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 informs 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 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.

Note: There is no single "right" answer to this question – we are looking for thoughtful reflection and commentary on whether or not this dataset, in its current form, encodes information about equity.

The `bike` data set as is does not have enough information to help us assess equity. Currently, it only contains information regarding the weather and whether or not the rider is registered or not. While one can make assumptions about the socio-economic status of registered versus casual riders, this information is definitely not enough and such assumptions would definitely be inaccurate when attempting to determine the equity of the bike sharing system. In order to assess equity properly, we need more information about the demographics of the people who are renting bikes, as well as perhaps location data correlating to where the bikes are actually being rented out from or returned to.

Some helpful information to include would be: 1. The gender of the person renting the bike 2. The race of the person renting the bike 3. The income level of the person renting the bike 4. The location where the person is living 5. The location where the bike is being rented 6. The location where the bike is being returned

In terms of changing the granularity of the data, hour by hour information is probably not as necessary. Instead, having a weekly or monthly summary of the bike renting statistics, along with more information about the demographics of the bike renters, would make this data set more reliable for formulating hypotheses about the equity of the bike sharing system.

0.2.6 Question 8b

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 in the US.

Based on your plots in this assignment, would you recommend expanding bike sharing to additional cities in the US? If so, what cities (or types of cities) would you suggest? Please list at least two reasons why, and mention which plot(s) you drew your 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!

I would suggest that more cities implement a bike-sharing system. Based on graph Q3a and Q3b, there are thousands of registered and casual riders renting bikes daily, and hundreds of registered and casual riders renting bikes during the most popular hours. This means that the bike-sharing system is definitely being utilized, and more cities adopting this program would allow more people access to bikes as an alternative and green form of transportation.

In terms of which type of cities I would suggest the bike-sharing system for, I would choose cities that are relatively warm and generally have higher temperatures. Based on graph 7b, as the temperature increases, the proportion of casual riders increases. This means that more casual riders (the vast majority of people) are likely to use the bike-sharing system when it is warmer, and thus bike-sharing systems likely will perform better among the general population in warmer cities.

Cities that implement a 4-day workweek as well would benefit from having a bike-sharing system. In the graph generated in Q4a, it is apparent that more casual riders rented bikes on non-working days. Thus, cities with more non-working days would also likely benefit from having more casual riders utilize the bike-sharing system.