0.0.1 Question 0

Question 0A What is the granularity of the data (i.e. what does each row represent)?

The data record the relative statistics of bike rental, which includes dates,season,temporature,windspeed,humidity,registered bikers,casual bikers and total bikers.It also shows that the whether bikers are rent in holiday or weekday.

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 you can collect to address some of these limitations?

The limitation of using the data may be the ambiguous meaning of some data fields (like yr,mnth, hr.etc). It also hard to distinguish the address of bike rental and the bikers characteristics. The two additional data categories that can be added are location(start point and end point) and rental time/cycling time.

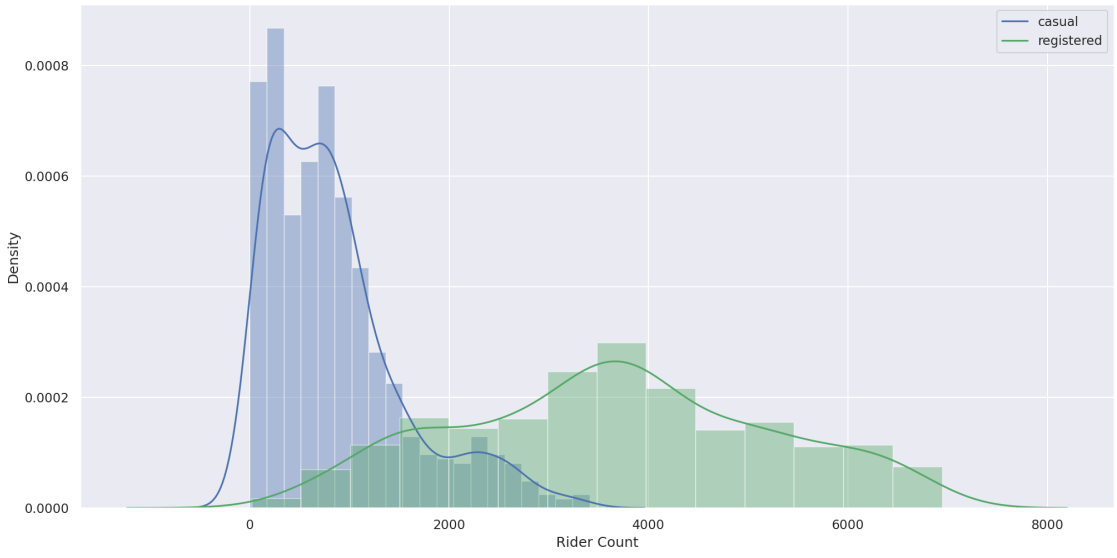
0.0.2 Question 2

Question 2a Use the `sns.histplot` 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.

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 [16]: sns.distplot(daily_counts['casual'], color = 'b')
         sns.distplot(daily_counts['registered'], color = 'g')
         plt.xlabel('Rider Count')
         plt.ylabel('Density')
         plt.legend(['casual', 'registered'])
         plt.show()
```



0.0.3 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 kde of the casual riders is higher than the kde of the registered riders. The registered curve seems to be bimodal and the registered curve more unimodal.The registered curve has a larger standard deviation than casual curve.The distribution of casual riders is skewed right, while the distribution of registered riders is symmetry.

0.0.4 Question 2c

The density plots do not show us how the counts for registered and casual riders vary together. Use `sns.lmplot` 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.

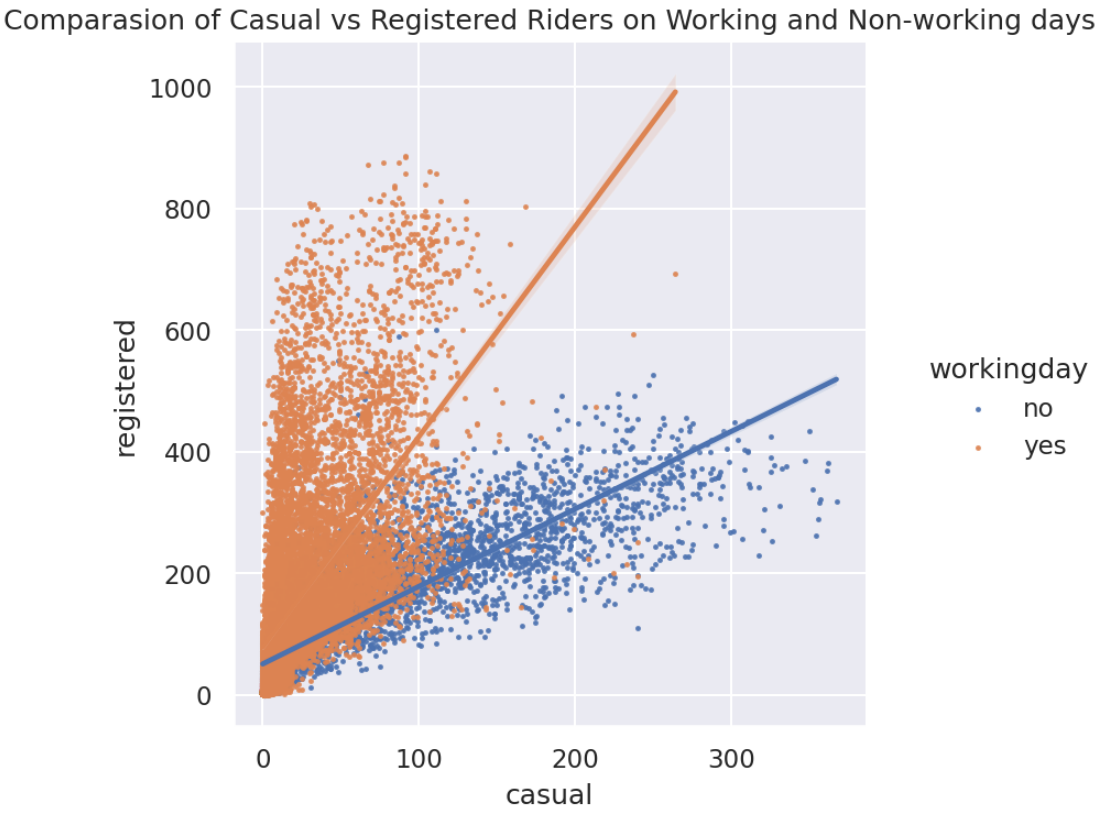
The `lmplot` function will also try to draw a linear regression line (just as you saw in Data 8). Color the points in the scatterplot according to whether or not the day is a working day (your colors do not have to match ours exactly, but they should be different based on whether the day is a working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

There are many points in the scatter plot, so make them small to help reduce overplotting. Also make sure to set `fit_reg=True` to generate the linear regression line. You can set the `height` parameter if you want to adjust the size of the `lmplot`.

Hints: * Checkout this helpful tutorial on `lmplot`.

- You will need to set `x`, `y`, and `hue` and the `scatter_kws` in the `sns.lmplot` call.
- You will need to call `plt.title` to add a title for the graph.

```
In [17]: # 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': 1
         plt.title('Comparasion of Casual vs Registered Riders on Working and Non-working days')
         plt.show()
```


Comparasion of Casual vs Registered Riders on Working and Non-working days
registered
casual
workingday
no
yes
0
200
400
600
800
1000
0
100
200
300

0.0.5 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?

On working day, there are more register riders than casual riders, while in non-working day, there are more casual riders than register.Overplotting makes it hard to see the overlap between the two in the 0-100 range as the orange dots cover much of the blue in this area.

Generating the plot with weekend and weekday separated can be complicated so we will provide a walk-through below, feel free to use whatever method you wish if you do not want to follow the walkthrough.

Hints: * You can use `loc` with a boolean array and column names at the same time * You will need to call kdeplot twice, each time drawing different data from the `daily_counts` table. * Check out this guide to see an example of how to create a legend. In particular, look at how the example in the guide makes use of the `label` argument in the call to `plt.plot()` and what the `plt.legend()` call does. This is a good exercise to learn how to use examples to get the look you want. * You will want to set the `cmap` parameter of `kdeplot` to `"Reds"` and `"Blues"` (or whatever two contrasting colors you'd like), and also set the `label` parameter to address which type of day you want to plot. You are required for this question to use two sets of contrasting colors for your plots.

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`.

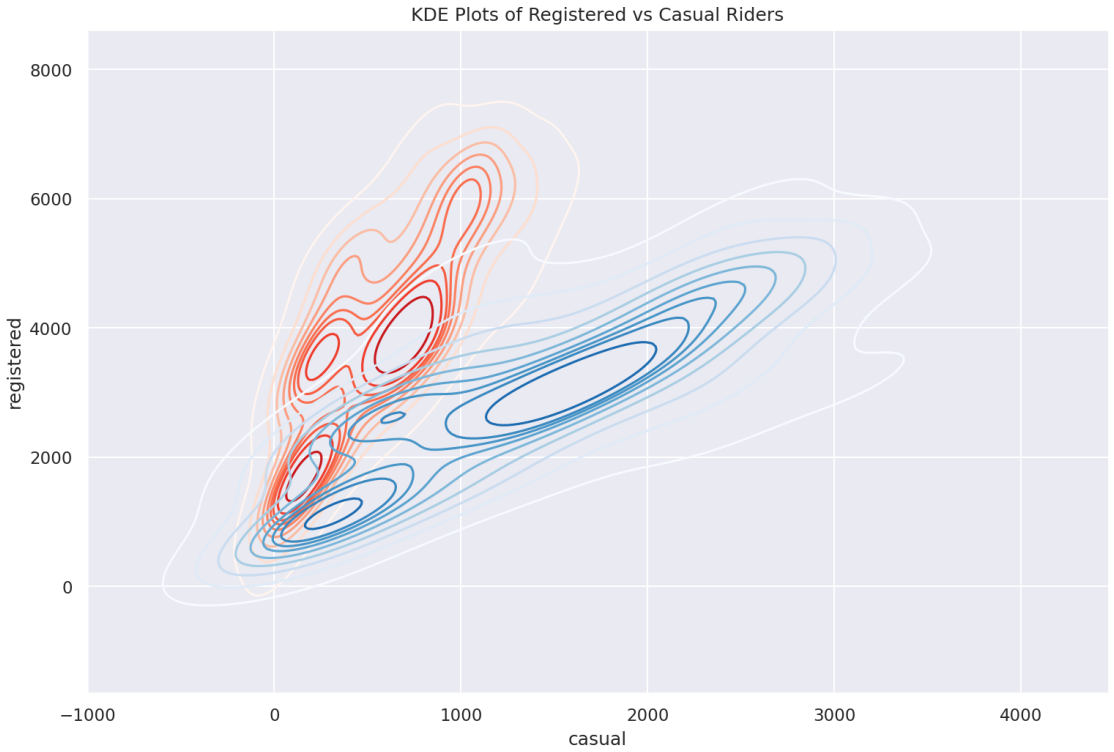
In [19]:
```
# Set the figure size for the plot
plt.figure(figsize=(12,8))

# Set 'is_workingday' to a boolean array that is true for all working_days
is_workingday = bike[bike['workingday'] == 'yes'].set_index('dteday')

# Bivariate KDEs require two data inputs.
# In this case, we will need the daily counts for casual and registered riders on workdays
# Hint: consider using the .loc method here.
casual_workday = is_workingday['casual'].groupby('dteday').sum()
registered_workday = is_workingday['registered'].groupby('dteday').sum()

# Use sns.kdeplot on the two variables above to plot the bivariate KDE for weekday rides
sns.kdeplot(casual_workday, registered_workday, cmap = 'Reds')

not_workingday = bike[bike['workingday'] == 'no'].set_index('dteday')
# Repeat the same steps above but for rows corresponding to non-workingdays
# Hint: Again, consider using the .loc method here.
casual_non_workday = not_workingday['casual'].groupby('dteday').agg('sum')
registered_non_workday = not_workingday['registered'].groupby('dteday').agg('sum')
sns.kdeplot(casual_non_workday, registered_non_workday, cmap = 'Blues')
# Use sns.kdeplot on the two variables above to plot the bivariate KDE for non-workingday ride
plt.title('KDE Plots of Registered vs Casual Riders');
plt.show()
```


KDE Plots of Registered vs Casual Riders
8000
6000
4000
2000
0
registered
−1000
0
1000
2000
3000
4000
casual

Question 3bi In your own words, describe what the lines and the color shades of the lines signify about the data.

The lines describe the distribution of the data,while the shades present the density of the data,the darker the color, the more dense the data is.

Question 3bii What additional details can you identify from this contour plot that were difficult to determine from the scatter plot?

We can see where the data points between the two plots overlap and how the data behaves in those areas.

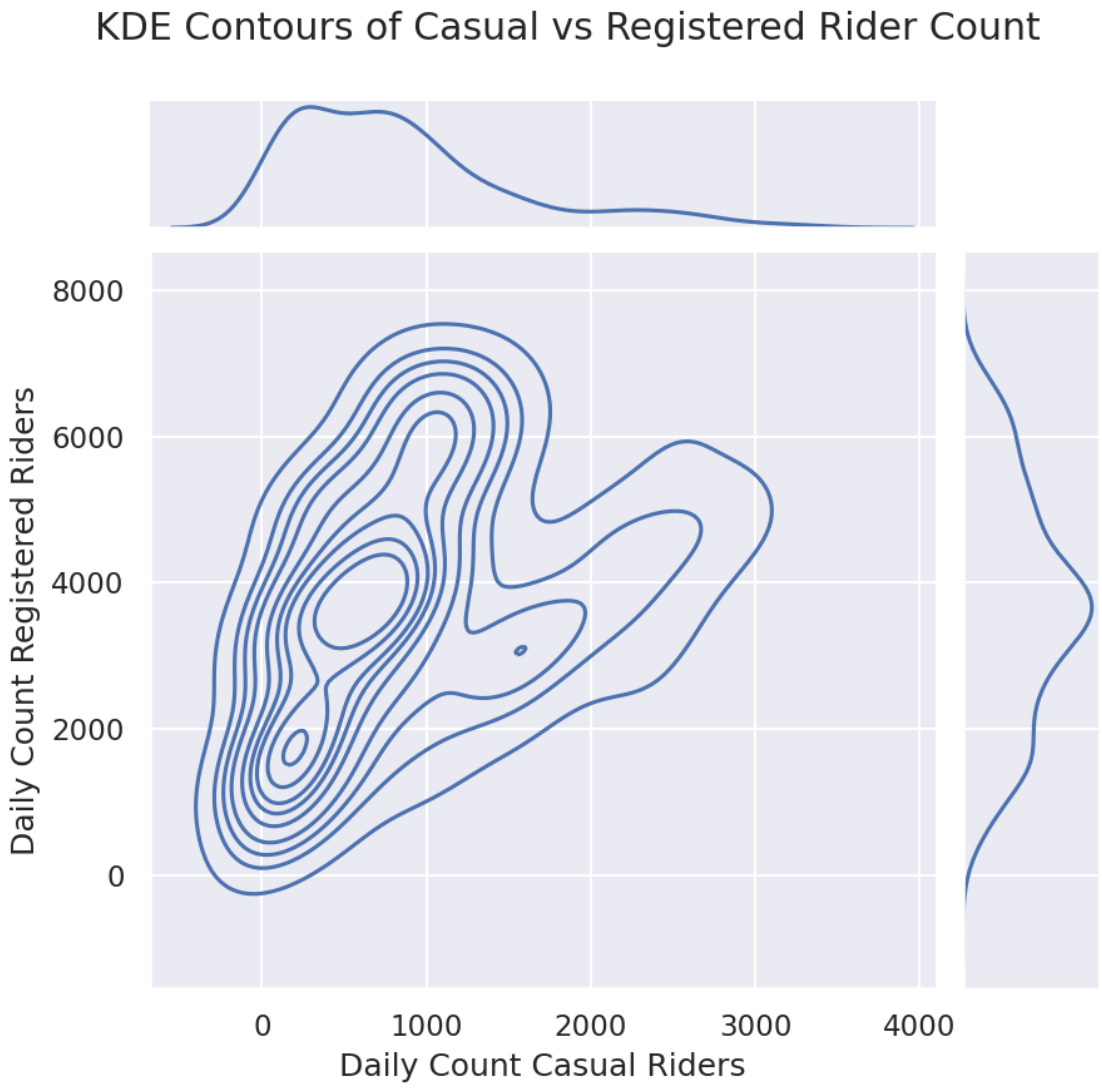
0.1 4: Joint Plot

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


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

0.2 5: Understanding Daily Patterns

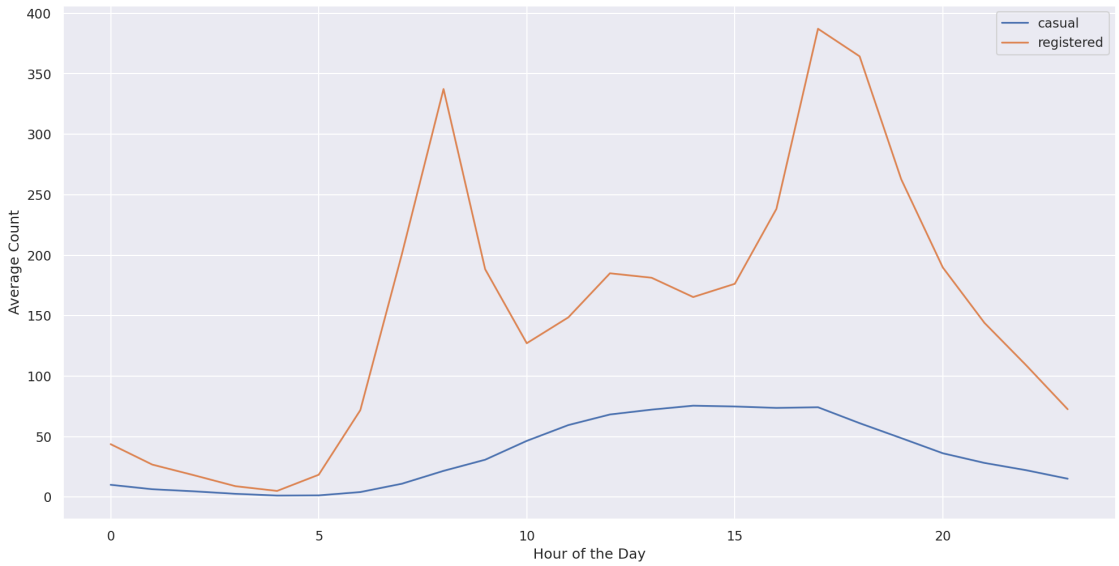
0.2.1 Question 5

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 different colored lines for different kinds of riders.

```
In [21]: new_data = bike.groupby('hr')[['hr', 'casual', 'registered']].mean()
         sns.lineplot(data=new_data, x = 'hr', y = 'casual', label = 'casual')
         sns.lineplot(data=new_data, x = 'hr', y = 'registered', label = 'registered')
         plt.xlabel('Hour of the Day')
         plt.ylabel('Average Count')
         plt.legend()
```

```
Out[21]: <matplotlib.legend.Legend at 0x7fa5d98f7ee0>
```



Question 5b What can you observe from the plot? Hypothesize about the meaning of the peaks in the registered riders' distribution.

The average count of registered riders is larger than casual riders. The peaks of registered are around 6:00am and 20:00pm, reflecting the time period with the largest number of rider, which could be cause by the morning and evening rush hours.

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.

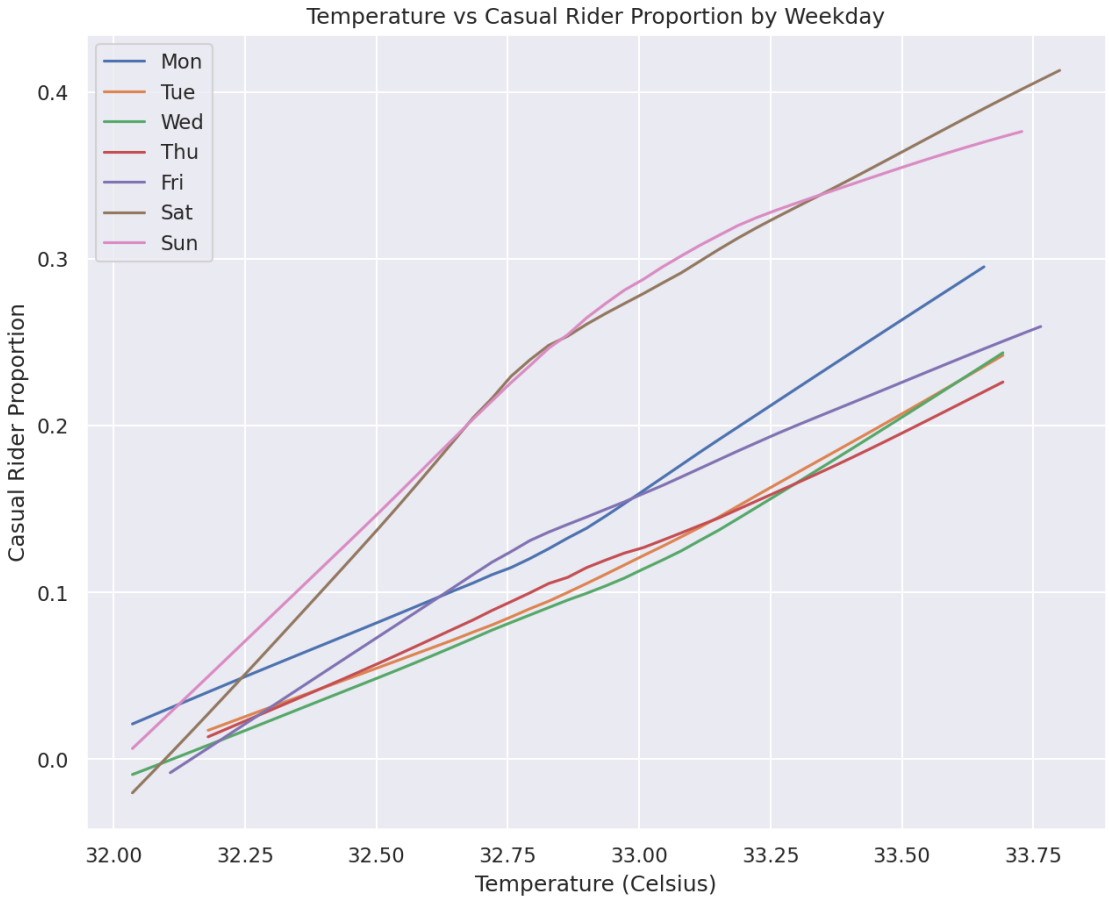
Hints: * Start by just plotting only one day of the week to make sure you can do that first.

- 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 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 [27]:
```
from statsmodels.nonparametric.smoothers_lowess import lowess

        plt.figure(figsize=(10,8))
        for x in ['Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat','Sun', ]:
            a = bike[bike['weekday'] == x]
            yobs = a['prop_casual']
            xobs = a['temp'] * 9 / 5 + 32
            smooth_y = lowess(yobs, xobs, return_sorted = False)
            sns.lineplot(x = xobs, y = smooth_y, label = x)
        plt.title("Temperature vs Casual Rider Proportion by Weekday")
        plt.xlabel('Temperature (Celsius)')
        plt.ylabel('Casual Rider Proportion')
        plt.show()
```


Temperature vs Casual Rider Proportion by Weekday
Mon
Tue
Wed
Thu
Fri
Sat
Sun
Casual Rider Proportion
0.0
0.1
0.2
0.3
0.4
32.00
32.25
32.50
32.75
33.00
33.25
33.50
33.75
Temperature (Celsius)

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?

The prop_casual are increasing with the increasing of temperature every day. I notice that Saturday and Sunday had the highest proportion of casual riders, which are greater than weekdays.

0.2.2 Question 7

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 these data are very helpful in assessing equity , because the equity information goes beyond the granularity of existing dataset. For example, we don't have any information about cyclists to help determine the ability of people of different socioeconomic classes, genders, races and communities to access and afford transportation services.

In order to improve this problem, we need socioeconomic information about riders, such as class, income, home address, etc

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!

From geographic aspect, I suggest expanding into cities (metropolitan areas) with higher temperatures and larger working populations. As can be seen from the chart above, with the temperature, there is a positive ratio with casual Riders.

From time period aspect, as can be seen from the graph, the peak hours tend to be the time when bikes are used the most, so I suggest increasing bike supply between 8am and 6pm.