

## Exercise1

### Creating subplots with col and row

We've seen in prior exercises that students with more absences ("absences") tend to have lower final grades ("G3"). Does this relationship hold regardless of how much time students study each week?

To answer this, we'll look at the relationship between the number of absences that a student has in school and their final grade in the course, creating separate subplots based on each student's weekly study time ("study\_time").

Seaborn has been imported as sns and matplotlib.pyplot has been imported as plt.

### Instructions

- Modify the code to use relplot() instead of scatterplot().
- Modify the code to create one scatter plot for each level of the variable "study\_time", arranged in columns.
- Adapt your code to create one scatter plot for each level of a student's weekly study time, this time arranged in rows.

```
In [ ]: # Change to use relplot() instead of scatterplot()
sns.scatterplot(x="absences", y="G3",
               data=student_data)

# Change to make subplots based on study time
sns.relplot(x="absences", y="G3",
            data=student_data,
            kind="scatter")

# Change this scatter plot to arrange the plots in rows instead of columns
sns.relplot(x="absences", y="G3",
            data=student_data,
            kind="scatter",
            col="study_time")

# Show plot
plt.show()

# _____#
#Solutions

# Change to use relplot() instead of scatterplot()
sns.relplot(x="absences", y="G3",
            data=student_data, kind='scatter')

# Change to make subplots based on study time
sns.relplot(x="absences", y="G3",
            data=student_data,
            kind="scatter", col='study_time')

# Change this scatter plot to arrange the plots in rows instead of columns
sns.relplot(x="absences", y="G3",
            data=student_data,
            kind="scatter",
            row="study_time")

# Show plot
plt.show()
```

## Exercise2

### Creating two-factor subplots

Let's continue looking at the student\_data dataset of students in secondary school. Here, we want to answer the following question: does a student's first semester grade ("G1") tend to correlate with their final grade ("G3")?

There are many aspects of a student's life that could result in a higher or lower final grade in the class. For example, some students receive extra educational support from their school ("schoolsup") or from their family ("famsup"), which could result in higher grades. Let's try to control for these two factors by creating subplots based on whether the student received extra educational support from their school or family.

Seaborn has been imported as sns and matplotlib.pyplot has been imported as plt.

### Instructions

- Use relplot() to create a scatter plot with "G1" on the x-axis and "G3" on the y-axis, using the student\_data DataFrame.
- Create column subplots based on whether the student received support from the school ("schoolsup"), ordered so that "yes" comes before "no".
- Add row subplots based on whether the student received support from the family ("famsup"), ordered so that "yes" comes before "no". This will result in subplots based on two factors.

```
In [ ]: # Create a scatter plot of G1 vs. G3

# Adjust to add subplots based on school support

# Adjust further to add subplots based on family support

# Show plot
plt.show()

#_____#
#Solutions

# Create a scatter plot of G1 vs. G3
sns.relplot(x='G1',y='G3',data=student_data,kind='scatter')

# Adjust to add subplots based on school support
sns.relplot(x="G1", y="G3",
            data=student_data,
            kind="scatter",col='schoolsup',col_order=['yes','no'])

# Adjust further to add subplots based on family support
sns.relplot(x="G1", y="G3",
            data=student_data,
            kind="scatter",
            col="schoolsup",
            col_order=["yes", "no"],row='famsup',row_order=['yes','no'])

# Show plot
plt.show()
```

## Exercise3

# Changing the size of scatter plot points

In this exercise, we'll explore Seaborn's mpg dataset, which contains one row per car model and includes information such as the year the car was made, the number of miles per gallon ("M.P.G.") it achieves, the power of its engine (measured in "horsepower"), and its country of origin.

What is the relationship between the power of a car's engine ("horsepower") and its fuel efficiency ("mpg")? And how does this relationship vary by the number of cylinders ("cylinders") the car has? Let's find out.

Let's continue to use relplot() instead of scatterplot() since it offers more flexibility.

## Instructions

- Use relplot() and the mpg DataFrame to create a scatter plot with "horsepower" on the x-axis and "mpg" on the y-axis. Vary the size of the points by the number of cylinders in the car ("cylinders").
- To make this plot easier to read, use hue to vary the color of the points by the number of cylinders in the car ("cylinders").

```
In [ ]: # Import Matplotlib and Seaborn

# Create scatter plot of horsepower vs. mpg

# Create scatter plot of horsepower vs. mpg

# Show plot

#_____#
#Solutions

# Import Matplotlib and Seaborn
import matplotlib.pyplot as plt
import seaborn as sns

# Create scatter plot of horsepower vs. mpg
sns.relplot(x='horsepower', y='mpg', data=mpg,kind='scatter',size='cylinders')

# Create scatter plot of horsepower vs. mpg
sns.relplot(x="horsepower", y="mpg",
            data=mpg, kind="scatter",
            size="cylinders",hue='cylinders')

# Show plot
plt.show()
```

## Exercise4

### Changing the style of scatter plot points

Let's continue exploring Seaborn's mpg dataset by looking at the relationship between how fast a car can accelerate ("acceleration") and its fuel efficiency ("mpg"). Do these properties vary by country of origin ("origin")?

Note that the "acceleration" variable is the time to accelerate from 0 to 60 miles per hour, in seconds. Higher values indicate slower acceleration.

#### Instructions

- Use relplot() and the mpg DataFrame to create a scatter plot with "acceleration" on the x-axis and "mpg" on the y-axis. Vary the style and color of the plot points by country of origin ("origin").

```
In [ ]: # Import Matplotlib and Seaborn

# Create a scatter plot of acceleration vs. mpg

# Show plot

# _____#
#Solutions

# Import Matplotlib and Seaborn
import matplotlib.pyplot as plt
import seaborn as sns

# Create a scatter plot of acceleration vs. mpg
sns.relplot(x='acceleration', y='mpg', data=mpg, kind='scatter', style='origin', hue='origin')

# Show plot
plt.show()
```

## Exercise5

### Interpreting line plots

In this exercise, we'll continue to explore Seaborn's mpg dataset, which contains one row per car model and includes information such as the year the car was made, its fuel efficiency (measured in "miles per gallon" or "M.P.G"), and its country of origin (USA, Europe, or Japan).

How has the average miles per gallon achieved by these cars changed over time? Let's use line plots to find out!

#### Instructions

- Use relplot() and the mpg DataFrame to create a line plot with "model\_year" on the x-axis and "mpg" on the y-axis.

```
In [ ]: # Import Matplotlib and Seaborn

# Create line plot

# Show plot

# _____#
#Solutions

# Import Matplotlib and Seaborn
import matplotlib.pyplot as plt
import seaborn as sns

# Create line plot
sns.relplot(x='model_year', y='mpg', data=mpg, kind='line')

# Show plot
plt.show()
```

## Exercise6

### Visualizing standard deviation with line plots

In the last exercise, we looked at how the average miles per gallon achieved by cars has changed over time. Now let's use a line plot to visualize how the distribution of miles per gallon has changed over time.

Seaborn has been imported as sns and matplotlib.pyplot has been imported as plt.

## Instructions

- Change the plot so the shaded area shows the standard deviation instead of the confidence interval for the mean.

```
In [ ]: # Make the shaded area show the standard deviation
sns.relplot(x="model_year", y="mpg",
            data=mpg, kind="line")

# Show plot

#_____#
#Solutions

# Make the shaded area show the standard deviation
sns.relplot(x="model_year", y="mpg",
            data=mpg, kind="line",ci='sd')

# Show plot
plt.show()
```

## Exercise7

## Plotting subgroups in line plots

Let's continue to look at the mpg dataset. We've seen that the average miles per gallon for cars has increased over time, but how has the average horsepower for cars changed over time? And does this trend differ by country of origin?

## Instructions

- Use relplot() and the mpg DataFrame to create a line plot with "model\_year" on the x-axis and "horsepower" on the y-axis. Turn off the confidence intervals on the plot.
- Create different lines for each country of origin ("origin") that vary in both line style and color.
- Add markers for each data point to the lines.
- Use the dashes parameter to use solid lines for all countries, while still allowing for different marker styles for each line.

```
In [ ]: # Import Matplotlib and Seaborn

# Create line plot of model year vs. horsepower

# Change to create subgroups for country of origin

# Add markers and make each line have the same style

# Show plot

#_____#
#Solutions

# Import Matplotlib and Seaborn
import matplotlib.pyplot as plt
import seaborn as sns

# Create line plot of model year vs. horsepower
sns.relplot(x='model_year',y='horsepower',data=mpg,dashes=False,kind='line',ci=None)

# Change to create subgroups for country of origin
sns.relplot(x="model_year", y="horsepower",
            data=mpg, kind="line",
            ci=None,style='origin')

# Add markers and make each line have the same style
sns.relplot(x="model_year", y="horsepower",
            data=mpg, kind="line",
            ci=None, style="origin",
            hue="origin",markers=True,dashes=False)

# Show plot
plt.show()
```