**EXERCISES**

**Changing style and palette**

Let's return to our dataset containing the results of a survey given to young people about their habits and preferences. We've provided the code to create a count plot of their responses to the question "How often do you listen to your parents' advice?". Now let's change the style and palette to make this plot easier to interpret.

We've already imported Seaborn as sns and matplotlib.pyplot as plt.

**Instructions 1/3**

* Set the style to "whitegrid" to help the audience determine the number of responses in each category.

**script.py**

# Set the color palette to "Purples"

sns.set\_style("whitegrid")

# Create a count plot of survey responses

category\_order = ["Never", "Rarely", "Sometimes",

                  "Often", "Always"]

sns.catplot(x="Parents Advice",

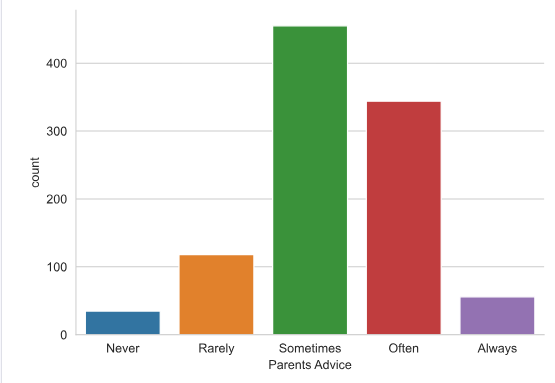
            data=survey\_data,

            kind="count",

            order=category\_order)

# Show plot

plt.show()



* Set the color palette to the sequential palette named "Purples".

**script.py**

# Set the color palette to "Purples"

sns.set\_style("whitegrid")

sns.set\_palette("PuRd")

# Create a count plot of survey responses

category\_order = ["Never", "Rarely", "Sometimes",

                  "Often", "Always"]

sns.catplot(x="Parents Advice",

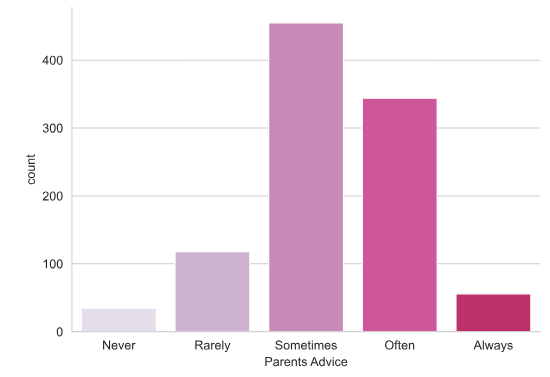
            data=survey\_data,

            kind="count",

            order=category\_order)

# Show plot

plt.show()



* Change the color palette to the diverging palette named "RdBu".

**script.py**

# Change the color palette to "RdBu"

sns.set\_style("whitegrid")

sns.set\_palette("RdBu")

# Create a count plot of survey responses

category\_order = ["Never", "Rarely", "Sometimes",

                  "Often", "Always"]

sns.catplot(x="Parents Advice",

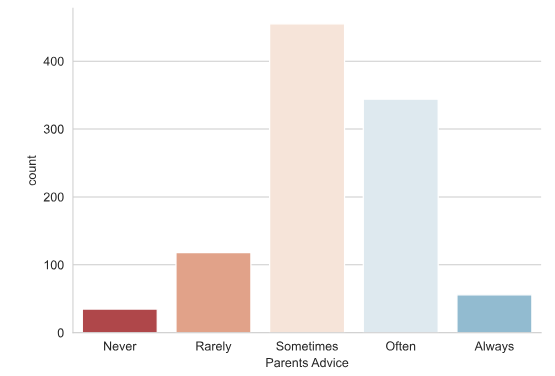
            data=survey\_data,

            kind="count",

            order=category\_order)

# Show plot

plt.show()



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Changing the scale**

In this exercise, we'll continue to look at the dataset containing responses from a survey of young people. Does the percentage of people reporting that they feel lonely vary depending on how many siblings they have? Let's find out using a bar plot, while also exploring Seaborn's four different plot scales ("contexts").

We've already imported Seaborn as sns and matplotlib.pyplot as plt.

**Instructions 1/4**

Set the scale ("context") to "paper", which is the smallest of the scale options.

**script.py**

# Set the context to "paper"

sns.set\_context("paper")

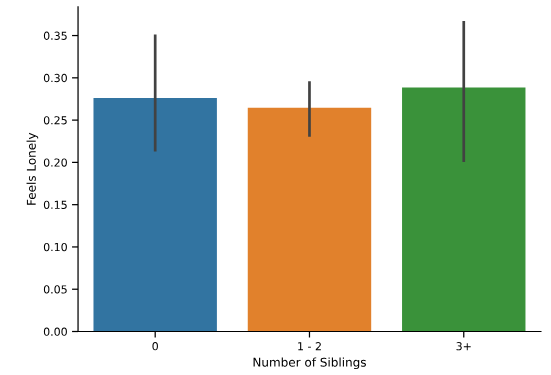
# Create bar plot

sns.catplot(x="Number of Siblings", y="Feels Lonely",

            data=survey\_data, kind="bar")

# Show plot

plt.show()



Change the context to "notebook" to increase the scale.

**script.py**

# Change the context to "talk"

sns.set\_context("notebook")

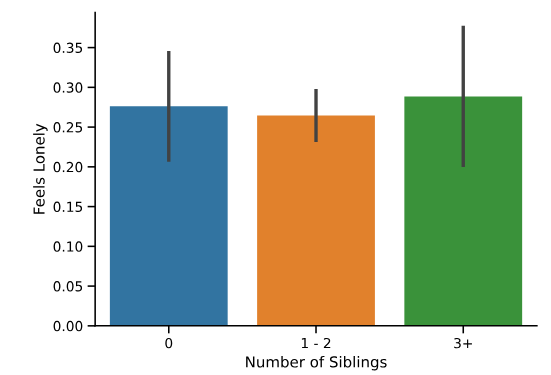
# Create bar plot

sns.catplot(x="Number of Siblings", y="Feels Lonely",

            data=survey\_data, kind="bar")

# Show plot

plt.show()



Change the context to "talk" to increase the scale.

**script.py**

# Change the context to "poster"

sns.set\_context("talk")

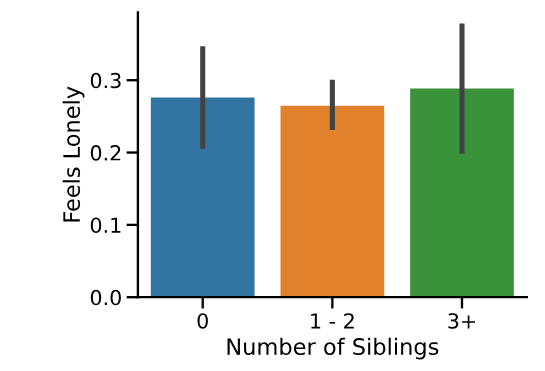
# Create bar plot

sns.catplot(x="Number of Siblings", y="Feels Lonely",

            data=survey\_data, kind="bar")

# Show plot

plt.show()



Change the context to "poster", which is the largest scale available.

**script.py**

# Change the context to "poster"

sns.set\_context("poster")

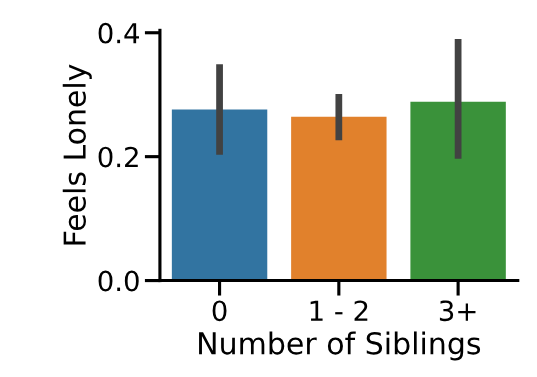
# Create bar plot

sns.catplot(x="Number of Siblings", y="Feels Lonely",

            data=survey\_data, kind="bar")

# Show plot

plt.show()



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Exercise**

**Exercise**

**Using a custom palette**

So far, we've looked at several things in the dataset of survey responses from young people, including their internet usage, how often they listen to their parents, and how many of them report feeling lonely. However, one thing we haven't done is a basic summary of the type of people answering this survey, including their age and gender. Providing these basic summaries is always a good practice when dealing with an unfamiliar dataset.

The code provided will create a box plot showing the distribution of ages for male versus female respondents. Let's adjust the code to customize the appearance, this time using a custom color palette.

We've already imported Seaborn as sns and matplotlib.pyplot as plt.

**Instructions**

**100 XP**

* Set the style to "darkgrid".
* Set a custom color palette with the hex color codes "#39A7D0" and "#36ADA4".

**script.py**

# Set the style to "darkgrid"

sns.set\_style("darkgrid")

# Set a custom color palette

custom\_palette = ["#39A7D0", "#36ADA4"]

sns.set\_palette(custom\_palette)

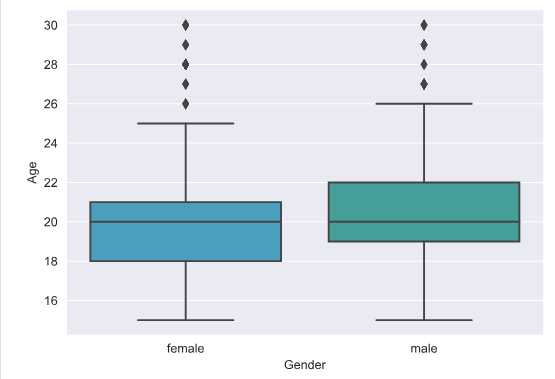
# Create the box plot of age distribution by gender

sns.catplot(x="Gender", y="Age",

            data=survey\_data, kind="box")

# Show plot

plt.show()



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_