Analyzing Survey Data with SQL & Python

Vatsal Parikh

This project analyses data from a survey about the growth of Finnish companies. The data reports the perceptions of top managers on growth, innovativeness, and the ability for renewal.

Where is the data from?

- Suominen & Pihlajamaa, 2022
- The dataset

What will I learn today?

- How to summarize and visualize questions with a numeric response using a histogram.
- How to determine whether there is a difference between two groups of numeric responses using a Mann-Whitney U test.
- How to summarize and visualize questions with a categorical response using a bar plot.

Setup

For this analysis we need the `plotly.express` package for drawing histograms and bar plots.

We'll also need the `mannwhitneyu` function from the `scipy.stats` package to perform the Mann-Whitney U test.

Instructions

Import the following packages.

- Import `plotly.express` using the alias `px`.
- From `scipy.stats` import the `mannwhitneyu` function.

```
# Import plotly.express using the alias px
import plotly.express as px

# From scipy.stats import the mannwhitneyu function
from scipy.stats import mannwhitneyu
```

Task 1: Import the Survey Dataset

The survey data is contained in a CSV file named `"What_does_it_take_to_generate_new_growth_Survey_data.csv"`.

Data dictionary

The dataset contains the following columns.

- `Growth_Firm`: Is the company (firm) *currently* classified as a growth company under OECD definitions?
- `question_2_row_1_transformed`: The responses to question 2, part 1 (with some pre-applied transformation).
- `question_2_row_2_transformed`: The responses to question 2, part 2 (with some pre-applied transformation).
- `question_3_row_1`: The responses to question 3, part 1.
- ...
- `question_7_row_1`: The responses to question 7, part 1.

The details of each question are fully described in `survey_questions.csv`, and we'll cover the details of the specific questions that we look at as we come to them in the tasks here.

Instructions

Use SQL to import the survey data.

- Select everything from `survey_data.csv`.
  - This uses European style CSV settings, so you can't use the default CSV reading settings.
  - Set the column delimiter to a semi-colon.
  - Set the decimal separator to a comma.
  - Set the null string to a space.
- Assign to a DataFrame named `survey`.

▶ Code hints

DataFrames and CSVs DataFrame as `survey`

```
-- Select everything from survey_data.csv
SELECT *
    FROM read_csv_auto("survey_data.csv", delim=";", decimal_separator=",", nullstr=" ")
```

	Gr...	question_2_row_1_transformed	question_2_row_2_transformed	question_3_...	question_3_...
0	0	35.1351351351	50.750939132	4	
1	0	23.0180426463	51.1822003413	5	
2	0	86.6404715128	62.9326385265	3	
3	0	17.6470588235	39.1304347826	3	
4	0	60	32.802124834	4	
5	0	-1.295496607	17.7106351559	5	
6	0	12.2754491018	64.6231446972	4	
7	0	66.6666666667	68.6814731515	3	
8	0	9.375	34.4537815126	5	
9	0	506.0606060606	689.2659826361	4	
10	0	26.9841269841	29.9241243114	4	
11	0	20	16.5501165501	4	
12	0	16.0714285714	49.4470459301	3	
13	0	81.8181818182	28.7995878413	4	
14	0	50	28.0409731114	5	
15	0	76.4705882353	60.6425702811	3	

Rows: 120

The dataset doesn't contain the actual questions that were asked. To find out what the questions are, we can look up the column titles in the data dictionary contained in `survey_questions.csv`.

Instructions

Use SQL to import the data dictionary for the survey questions.

- Select everything from `survey_questions.csv`.
  - This uses the default read CSV settings.

DataFrames and CSVs DataFrame as

```
-- Select everything from survey_data.csv
SELECT *
    FROM 'survey_questions.csv'
```

	column			section	title
0	question_2_row_1_transformed	2	1	estimated growth	Expected employee count in five yea
1	question_2_row_2_transformed	2	2	estimated growth	Expected revenue in five years (as a
2	question_3_row_1	3	1	company culture	Employees are encouraged to be cre
3	question_3_row_2	3	2	company culture	Managers are expected to be creativ
4	question_3_row_3	3	3	company culture	Employees' ability to function creativ
5	question_3_row_4	3	4	company culture	We are constantly looking for ways t
6	question_3_row_5	3	5	company culture	Assistance in developing new ideas i
7	question_3_row_6	3	6	company culture	Our organization is open and respons
8	question_3_row_7	3	7	company culture	Managers here are always searching
9	question_3_row_8	3	8	company culture	Our organization has a clear and insp
10	question_3_row_9	3	9	company culture	We have ensured that all managers c
11	question_3_row_10	3	10	company culture	All departments and employees shar
12	question_3_row_11	3	11	company culture	We believe that higher risks are wort
13	question_3_row_12	3	12	company culture	We encourage innovative initiatives,
14	question_3_row_13	3	13	company culture	We do not like to “play it safe”
15	question_3_row_14	3	14	company culture	Managers are constantly seeking new

Rows: 35

Task 2: Visualizing Numeric Responses

Question 2 asks

> If the firm develops the way you would like it to, how much revenue would the firm receive, and how many employees would it have five years ahead? Disregard possible inflation.

In this task we'll consider the first part, about employee count.

The responses are numeric, and so it's natural to visualize the distribution as a histogram.

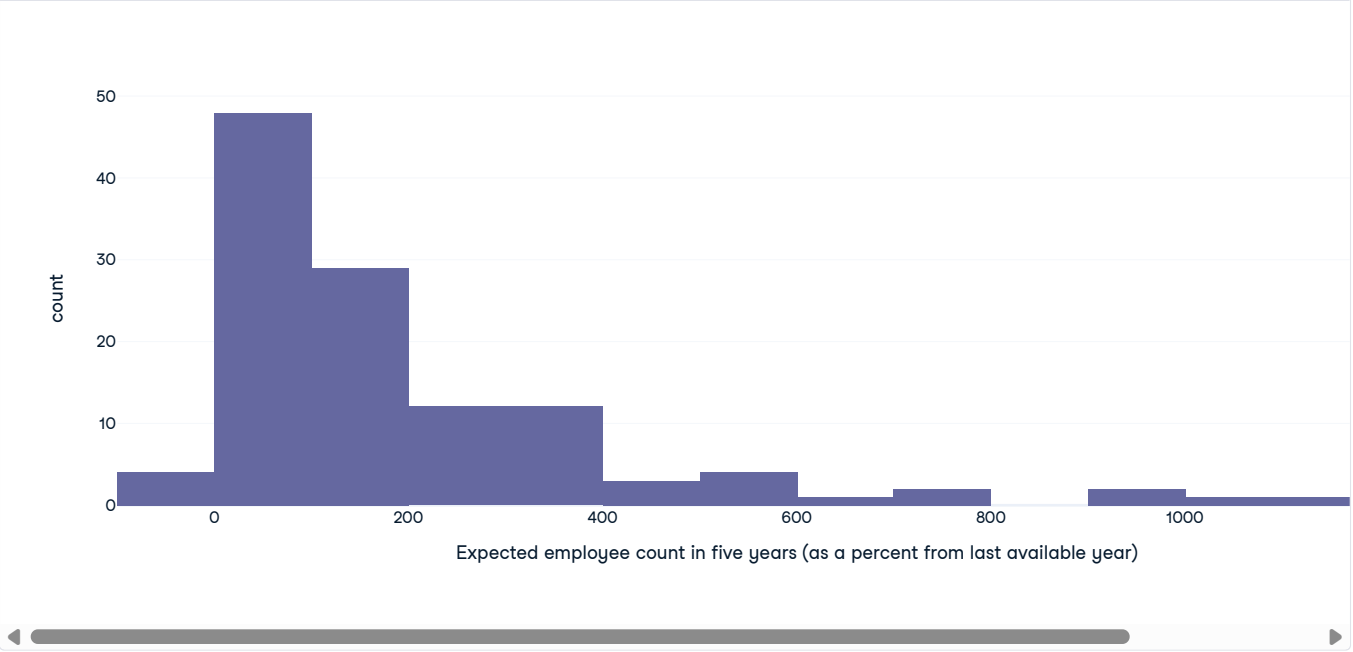
Instructions

Draw a histogram of expected employee count in five years.

- Draw a histogram of the `survey` data.
- On the x-axis, plot `question_2_row_1_transformed`.
- Set the x-axis label to `"Expected employee count in five years (as a percent from last available year)"`.

▶ Code hints

```
# Draw a histogram of the survey data
# On the x-axis, plot question_2_row_1_transformed
# Facet the plot in rows by growth firm status.
px.histogram(
    survey,
    x="question_2_row_1_transformed",
    labels={
        "question_2_row_1_transformed": "Expected employee count in five years (as a percent from last available year)"
    }
)
```



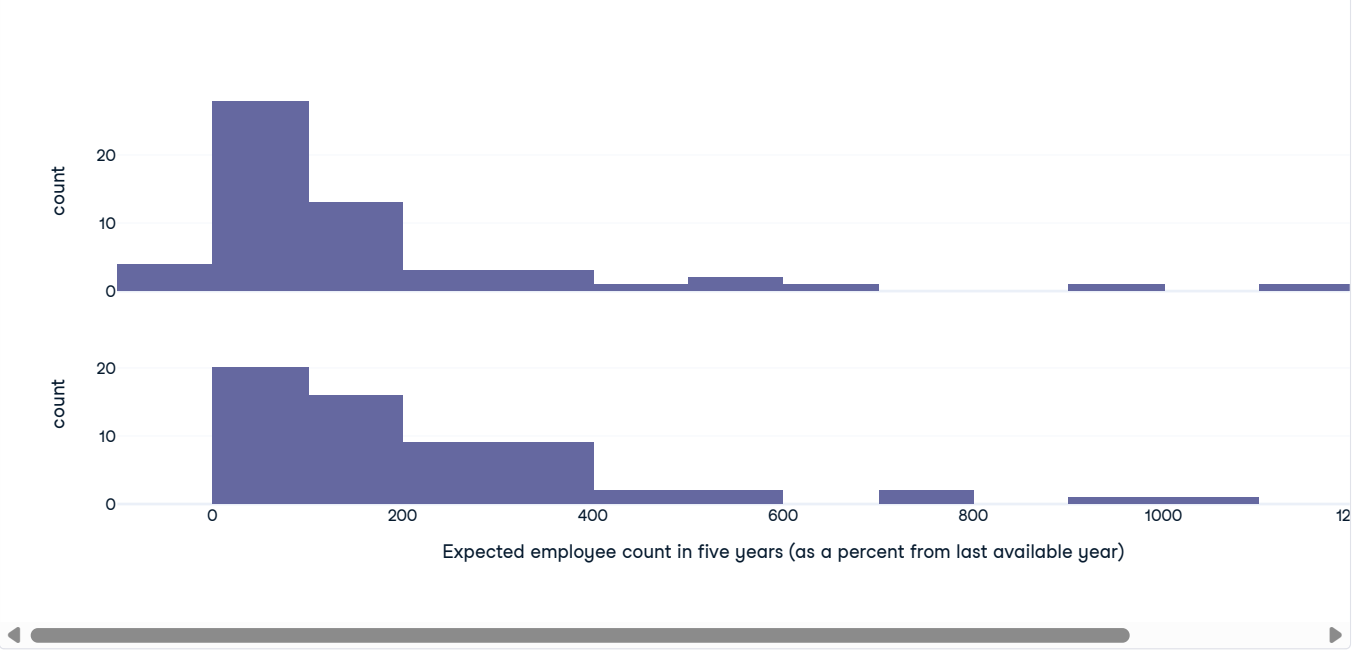
An interesting question is whether companies that are currently classified as *growth* have different expectations of how many more employees they will add over the next five years compared to *non-growth* companies. We can draw a histogram for each.

Instructions

Update the histogram of expected employee count in five years.

- Copy and paste your previous histogram code.
- Facet the plot in rows by growth status.

```
# Copy and paste your previous histogram code.
# On the x-axis, plot question_2_row_1_transformed
# Facet the plot in rows by growth status.
px.histogram(
    survey,
    x="question_2_row_1_transformed",
    facet_row="Growth_Firm",
    labels={
        "question_2_row_1_transformed": "Expected employee count in five years (as a percent from last available year)"
    }
)
```



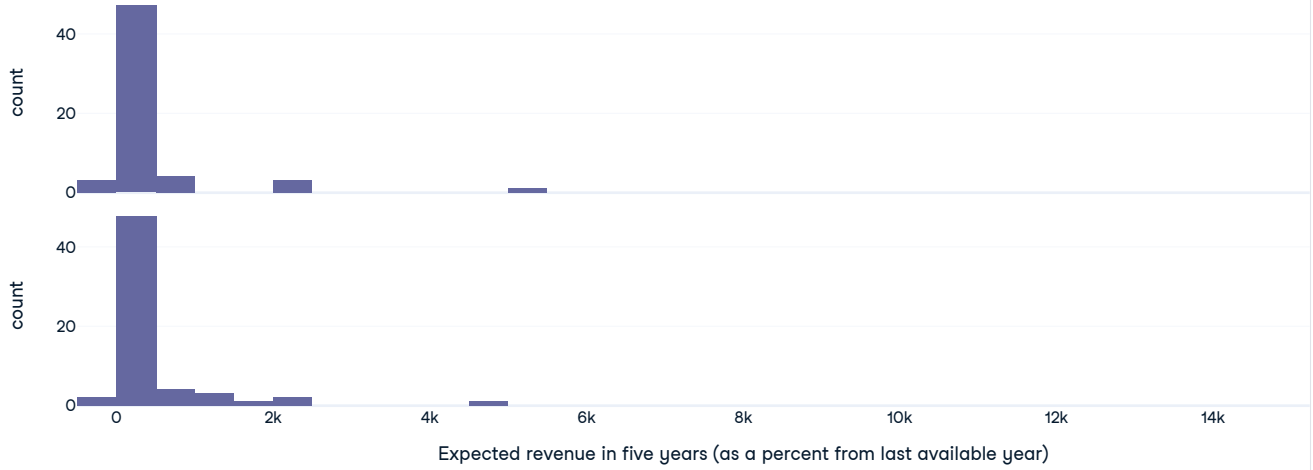
Visualize Another Question With Numeric Reponses

Instructions

Draw the last histogram again, this time with the results of question 2, part 2.

- Copy and paste your previous code.
- Change the column to `question_2_row_1_transformed`.
- Change the x-axis title to `"Expected revenue in five years (as a percent from last available year)"`.

```
# Visualize question 2, part 2
px.histogram(
    survey,
    x="question_2_row_2_transformed",
    facet_row="Growth_Firm",
    labels={
        "question_2_row_2_transformed": "Expected revenue in five years (as a percent from last available year)"
    }
)
```



Task 3: Calculating Statistical Significance Between Groups of Numeric Responses

The two histograms look pretty similar. However, there may be a statistically significant difference between the two groups.

We data don't have a bell-shaped normal distribution curve, so we use a Mann-Whitney U test (a.k.a. Wilcoxon Rank Sum test) to compare them.

Instructions

Get the non-growth rows for question 2, part 1.

- Select the `question_2_row_1_transformed` column from the survey CSV.
- Get rows where growth firm status is `0`.
- Assign to a dataframe named `q2_1_non_growth`.

DataFrames and CSVs DataFrame as

```
-- Select the question_2_row_1_transformed column from the survey CSV
-- Get rows where growth firm status is 0
SELECT question_2_row_1_transformed
    FROM read_csv_auto("survey_data.csv", delim=";", decimal_separator=",", nullstr=" ")
    WHERE Growth_Firm = 0
```

	question_2_row_1_transformed
0	35.1351351351
1	23.0180426463
2	86.6404715128
3	17.6470588235
4	60
5	-1.295496607
6	12.2754491018
7	66.6666666667
8	9.375
9	506.0606060606
10	26.9841269841
11	20
12	16.0714285714
13	81.8181818182
14	50
15	76.4705882353
16	96.8503937008

Rows: 58

Instructions

Get the growth rows for question 2, part 1.

- Do the same again, this time getting rows where growth firm status is `1`.
- Assign to `q2_1_growth`.

DataFrames and CSVs DataFrame as

```
-- Select the question_2_row_1_transformed column from the survey CSV
-- Get rows where growth firm status is 1
SELECT question_2_row_1_transformed
    FROM read_csv_auto("survey_data.csv", delim=";", decimal_separator=",", nullstr=" ")
    WHERE Growth_Firm = 1
```

	question_2_row_1_transformed
0	580.2721088435
1	166.6666666667
2	400
3	7.2961373391
4	25
5	372.972972973
6	284.6153846154
7	153.5211267606
8	108.3333333333
9	200
10	20
11	1076.4705882353
12	126.4150943396
13	334.7826086957
14	92.3076923077
15	102.7027027027
16	31.8181818182

Rows: 62

Instructions

- Perform a Mann-Whitney U test on `q2_1_non_growth` and `q2_1_growth`.
- Look at the p-value. Is it more or less than `0.05`?

```
# Perform a Mann-Whitney U test on q2_1_non_growth and q2_1_growth
mannwhitneyu(q2_1_non_growth, q2_1_growth)
```

```
MannwhitneyuResult(statistic=array([1299.]), pvalue=array([0.00884359]))
```

Task 4: Visualizing Categorical Responses

Many of the questions in the survey dataset have categorical responses with 5 options from "Strongly disagree" to "Strongly agree".

The values are encoded as `1` for `Strongly disagree` through to `5` for `Strongly agree`. For visualizing the responses, it is better to have explicit labels rather than numbers.

We'll gradually build up the SQL query to get the counts for each response type then draw a bar plot.

Useful jargon

These sorts of survey responses where answer is a level of agreement to a statement are called **Likert scales** (or rating scales).

Instructions

- Import everything from `agree_disagree.csv` as `lookup`.

▶ Code hints

DataFrames and CSVs DataFrame as

```
-- Import everything from agree_disagree.csv as lookup
SELECT *
    FROM 'agree_disagree.csv' AS lookup
```

	response
0	1 Strongly disagree
1	2 Disagree
2	3 Neither agree or disagree
3	4 Agree
4	5 Strongly agree

Rows: 5

We're working towards getting the counts for each of the five responses, even if they aren't all present in the dataset. That means that we want zero counts to be allowed. To achieve this, we need a left join.

Instructions

Extend the previous code to join the lookup to the survey data.

- Copy and paste the previous code.
- Left join lookup to the survey data on `lookup` `code` equal to `survey` `question_3_row_1`.
- Select the `lookup` `response` and the `survey` `question_3_row_1` columns.

▶ Code hints

DataFrames and CSVs DataFrame as

```
-- Copy and paste the previous code
-- Left join lookup to the survey data on lookup code equal to survey question_3_row_1
-- Select the lookup response and the survey question_3_row_1 columns
SELECT
    lookup.response,
    survey.question_3_row_1
    FROM 'agree_disagree.csv' AS lookup
    LEFT JOIN read_csv_auto("survey_data.csv", delim=";", decimal_separator=",", nullstr=" ") AS survey
        ON lookup.code = survey.question_3_row_1
```

	response	question_3_...
0	Agree	4
1	Strongly agree	5
2	Neither agree or disagree	3
3	Neither agree or disagree	3
4	Agree	4
5	Strongly agree	5
6	Agree	4
7	Neither agree or disagree	3
8	Strongly agree	5
9	Agree	4
10	Agree	4
11	Agree	4
12	Neither agree or disagree	3
13	Agree	4
14	Strongly agree	5
15	Neither agree or disagree	3
16	Strongly agree	5

Rows: 121

Instructions

Extend the previous code to get counts.

- Copy and paste the previous code.
- Change the selection from `survey.question_3_row_1` to the count of that column, naming the result as `n`.
- Group by the `lookup` `response`.

DataFrames and CSVs DataFrame as

```
-- Copy and paste the previous code
-- Change the selection from survey.question_3_row_1 to the count of that column, naming the result as n
-- Group by the lookup response
SELECT
    lookup.response,
    COUNT(survey.question_3_row_1) AS n
    FROM 'agree_disagree.csv' AS lookup
    LEFT JOIN read_csv_auto("survey_data.csv", delim=";", decimal_separator=",", nullstr=" ") AS survey
        ON lookup.code = survey.question_3_row_1
    GROUP BY lookup.response
```

	response	
0	Strongly disagree	0
1	Neither agree or disagree	18
2	Agree	67
3	Strongly agree	29
4	Disagree	6

Rows: 5

In order to draw an easy to interpret plot, we want to include a color scheme based on the level of agreement with the statement.

Using `lookup.code - 3` gives us a range from `-2` (Strongly disagree) to `2` (Strongly agree).

Instructions

Extend the previous code to include the level of agreement, and order the results.

- Copy and paste the previous code.
- Calculate the `lookup` `code` minus 3, naming the result as `agreement`.
- Order the result by `lookup` `code`.
- Assign the result to a DataFrame named `q3_1_counts`.

DataFrames and CSVs DataFrame as

```
-- Copy and paste the previous code
-- Calculate the lookup code minus 3, naming the result as agreement
-- Order the result by lookup code
SELECT
    lookup.response,
    COUNT(survey.question_3_row_1) AS n,
    lookup.code - 3 AS agreement
    FROM 'agree_disagree.csv' AS lookup
    LEFT JOIN read_csv_auto("survey_data.csv", delim=";", decimal_separator=",", nullstr=" ") AS survey
        ON lookup.code = survey.question_3_row_1
    GROUP BY lookup.code, lookup.response
    ORDER BY lookup.code
```

	response		a
0	Strongly disagree	0	-2
1	Disagree	6	-1
2	Neither agree or disagree	18	0
3	Agree	67	1
4	Strongly agree	29	2

Rows: 5

Now we are (finally) ready to plot the questions 3 part 1 responses.

These types of categorical variables where you have a neutral response and two sets of responses going in opposite directions (agreeing and disagreeing) are best visualized using a diverging color scale.

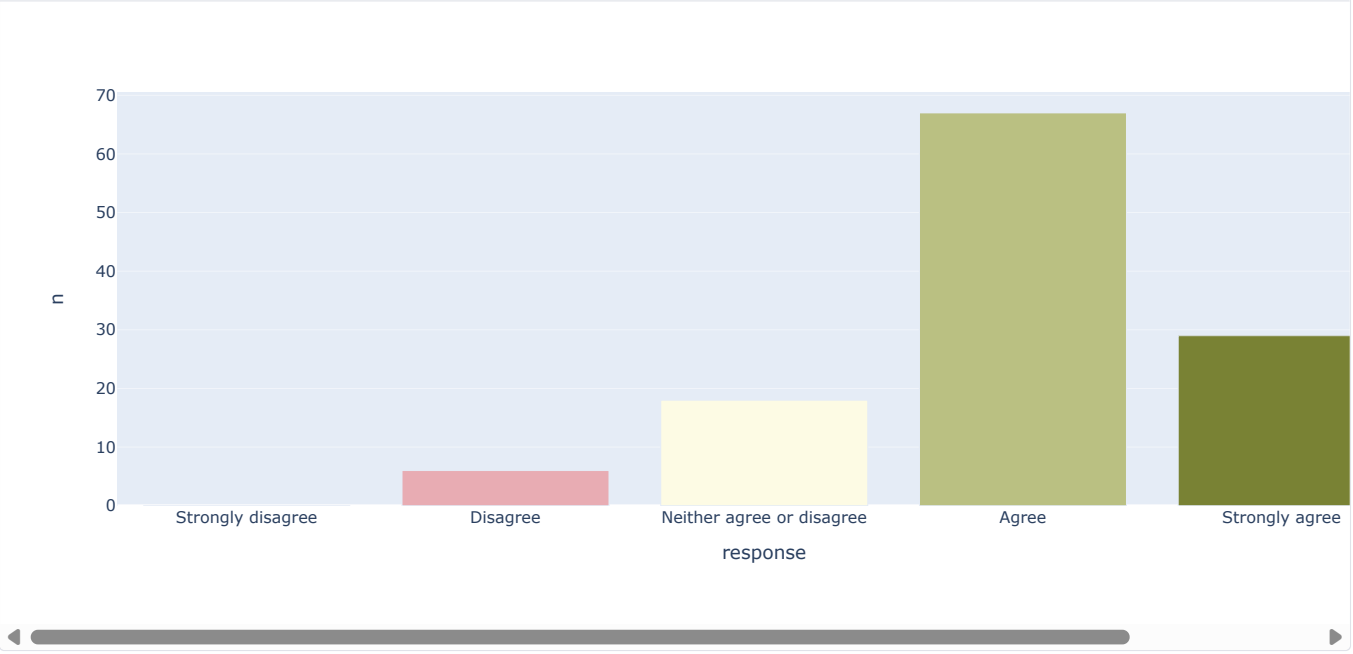
Instructions

Draw a bar plot of the response counts.

- Draw a bar plot of `q3_1_counts`.
- On the x axis, plot `response`.
- On the y axis, plot `n`.
- Color the bars by `agreement`.
- Use the diverging continuous color scale `px.colors.diverging.Armyrose_r`.

▶ Code hints

```
# Draw a bar plot of q3_1_counts
# On the x axis, plot response
# On the y axis, plot n
# Color the bars by agreement
# Use the diverging continuous color scale px.colors.diverging.Armyrose_r
px.bar(
    q3_1_counts,
    x="response",
    y="n",
    color="agreement",
    color_continuous_scale=px.colors.diverging.Armyrose_r
)
```



YOUR TURN: Visualize Another Question with Categorical Responses

Instructions

Choose another agree-disagree question (any part of q3 to q6), then get the counts of the responses.

- Copy and paste your previous SQL query.
- Change the column to one one for your new question. (The code needs changing in 2 places.)
- Assign the results to a DataFrame with a meaningful name.

DataFrames and CSVs DataFrame as

```
-- Get the counts for your new categorical question
SELECT
    lookup.response,
    COUNT(survey.question_3_row_13) AS n,
    lookup.code - 3 AS agreement
    FROM 'agree_disagree.csv' AS lookup
    LEFT JOIN read_csv_auto("survey_data.csv", delim=";", decimal_separator=",", nullstr=" ") AS survey
        ON lookup.code = survey.question_3_row_13
    GROUP BY lookup.code, lookup.response
    ORDER BY lookup.code
```

	response	n	a
0	Strongly disagree	4	-2
1	Disagree	25	-1
2	Neither agree or disagree	41	0
3	Agree	36	1
4	Strongly agree	14	2

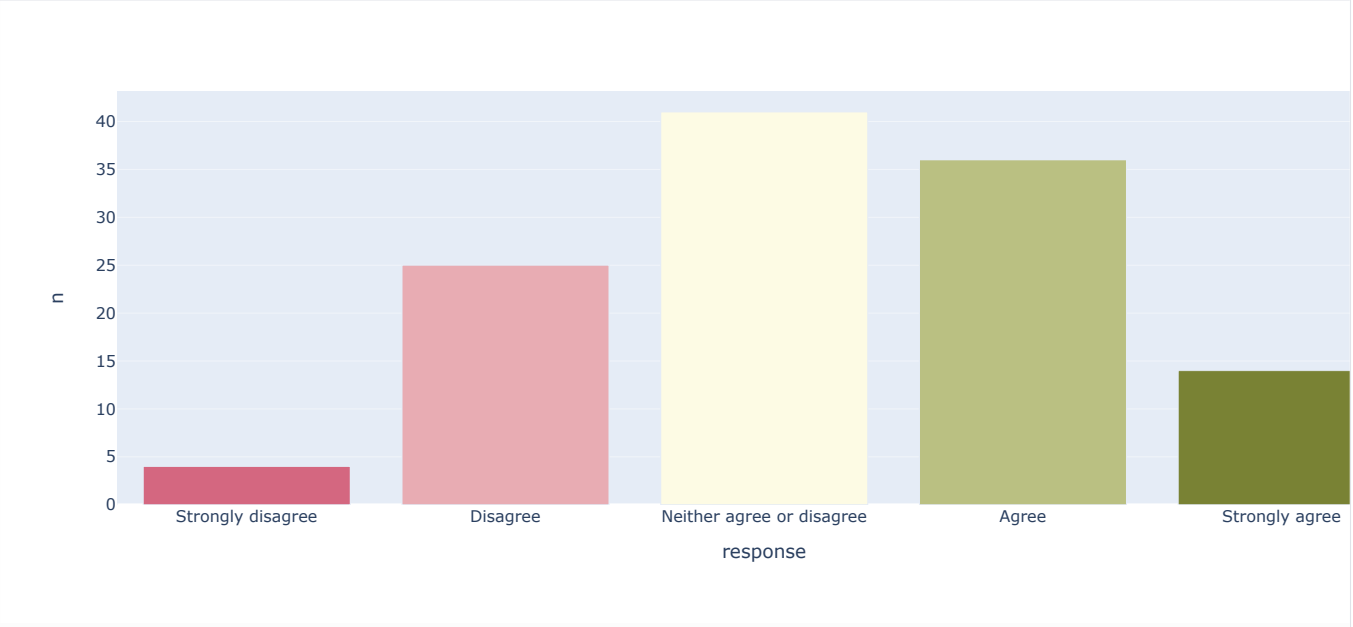
Rows: 5

Instructions

Draw a bar plot of the response counts for your new question.

- Copy and paste your previous plotting code.
- Change the dataset to your new DataFrame of counts.

```
# Visualize the responses from your new categorical question
px.bar(
    q3_13_counts,
    x="response",
    y="n",
    color="agreement",
    color_continuous_scale=px.colors.diverging.Armyrose_r
)
```



- Visualize the relationship between responses for both the numeric questions using a scatter plot, with points colored by growth status.
- Visualize the relationship between responses for two categorical questions using a heatmap, with cells colored by count. How might you extend this to display the growth statuses?
- Find out which questions had the strongest agreement with the statement. That is, calculate which questions had the highest average numeric score for the responses.
- Find out which questions had the strongest level of feeling in the responses. That is, calculate which questions had more "Strongly agree" and "Strongly disagree" responses. Think of a way to weight the different responses and calculate an average for each question.