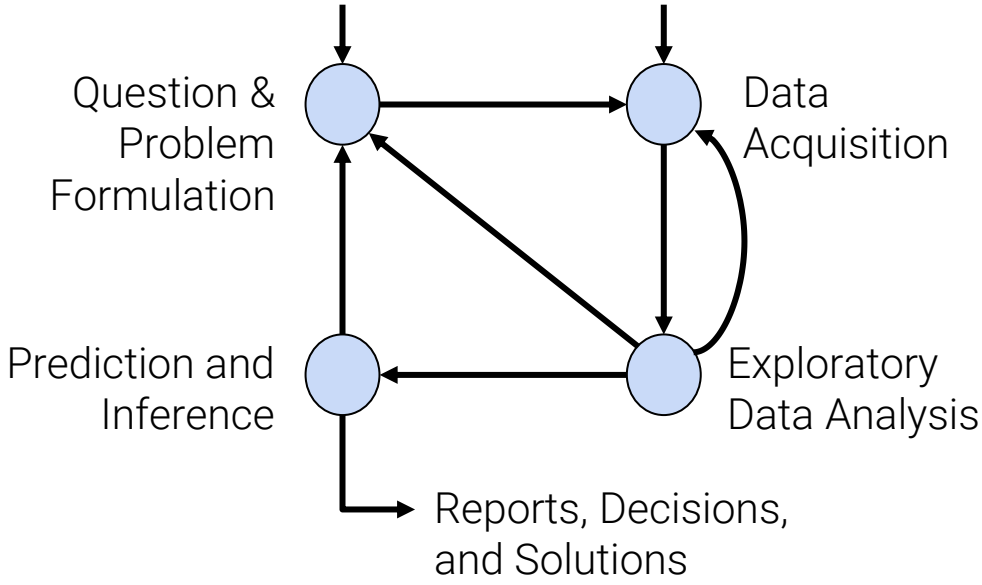
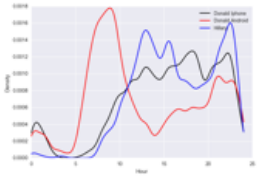
LECTURE 7

Visualization, Part I

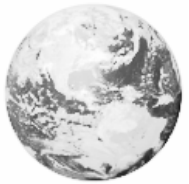
Visualizing Distributions and Relationships Between Quantitative Variables

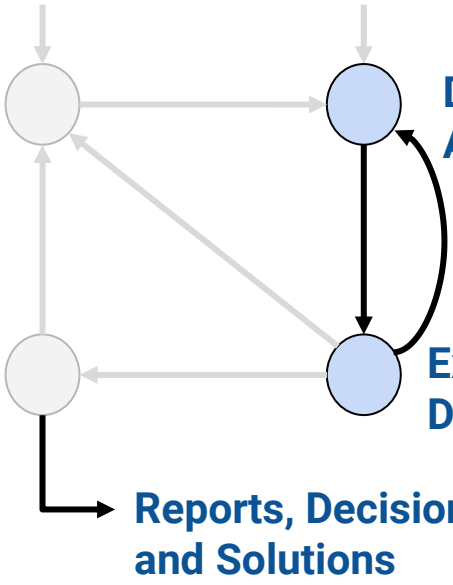
Plan for Weeks 3 and 4

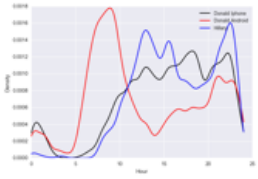




Plan for Weeks 3 and 4







(today)

Data Wrangling Intro to EDA	Working with Text Data Regular Expressions	Plots and variables Seaborn	Viz principles KDE/Transformations

Last week (Part I: Processing Data)

This week (Part II: Visualizing and Reporting Data)

Visualizations

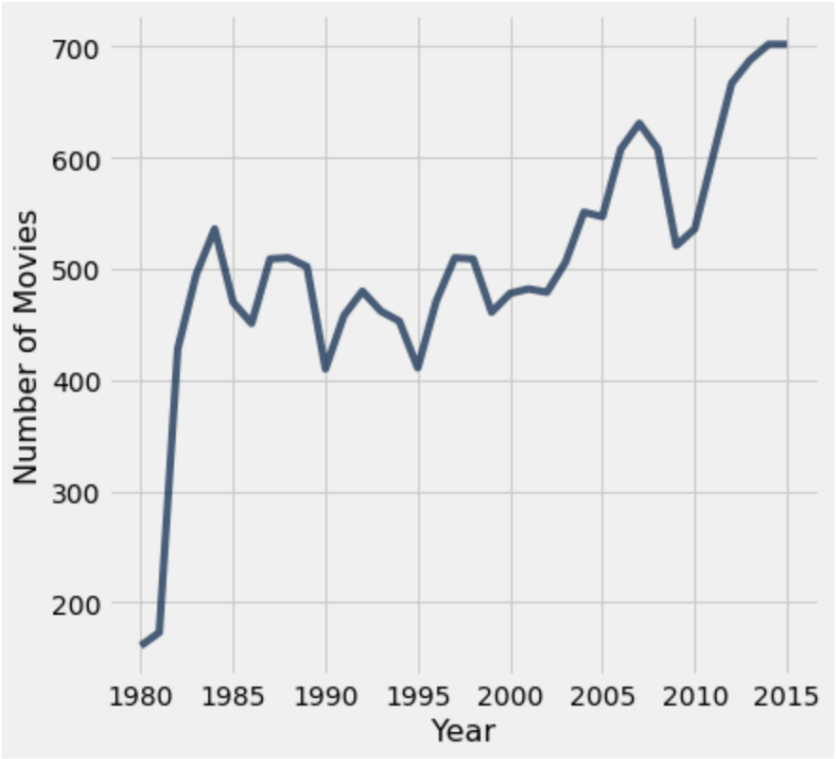
Lecture 07

- **Visualizations**
  - In the Real World, Goals
- Distributions
- Bar Plots for Distributions
- Bar Plot Introspection
- Histograms
- Evaluating Histograms
- Box Plots and Violin Plots
- Comparing Quantitative Distribution

Line and Scatter Charts

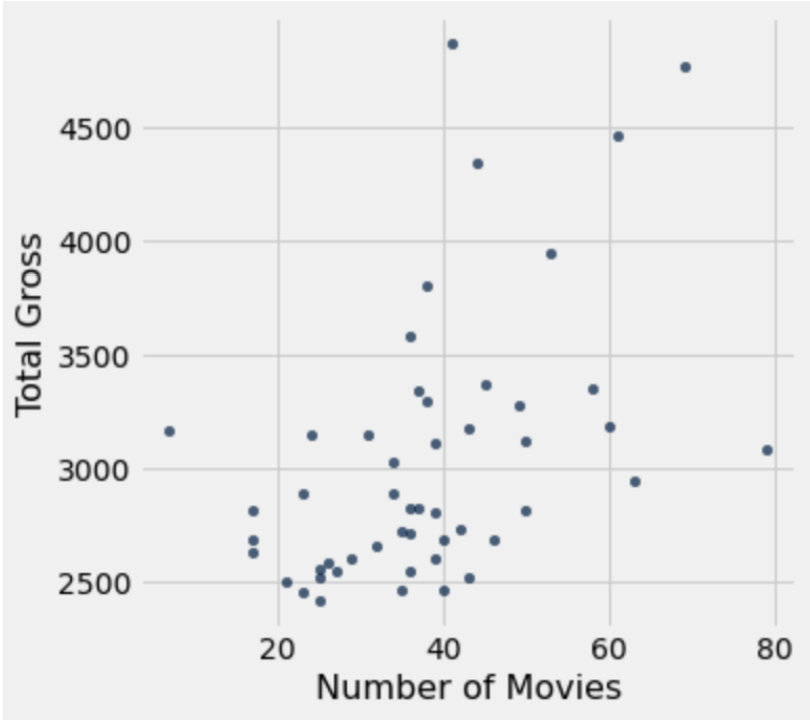
```
# The call is
# t.plot(x_label, y_label)

movies.plot('Year', 'Number of Movies')
```



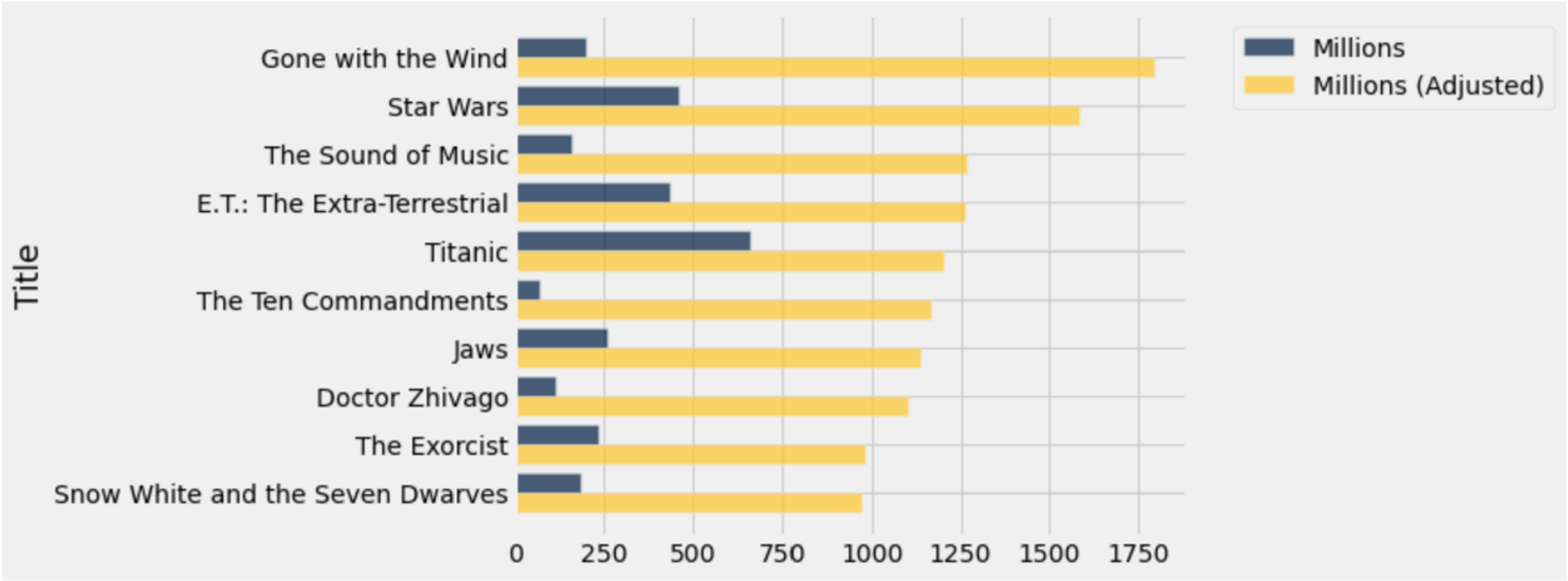

```
# The call is
# t.scatter(x_label, y_label)

actors.scatter('Number of Movies', 'Total Gross')
```



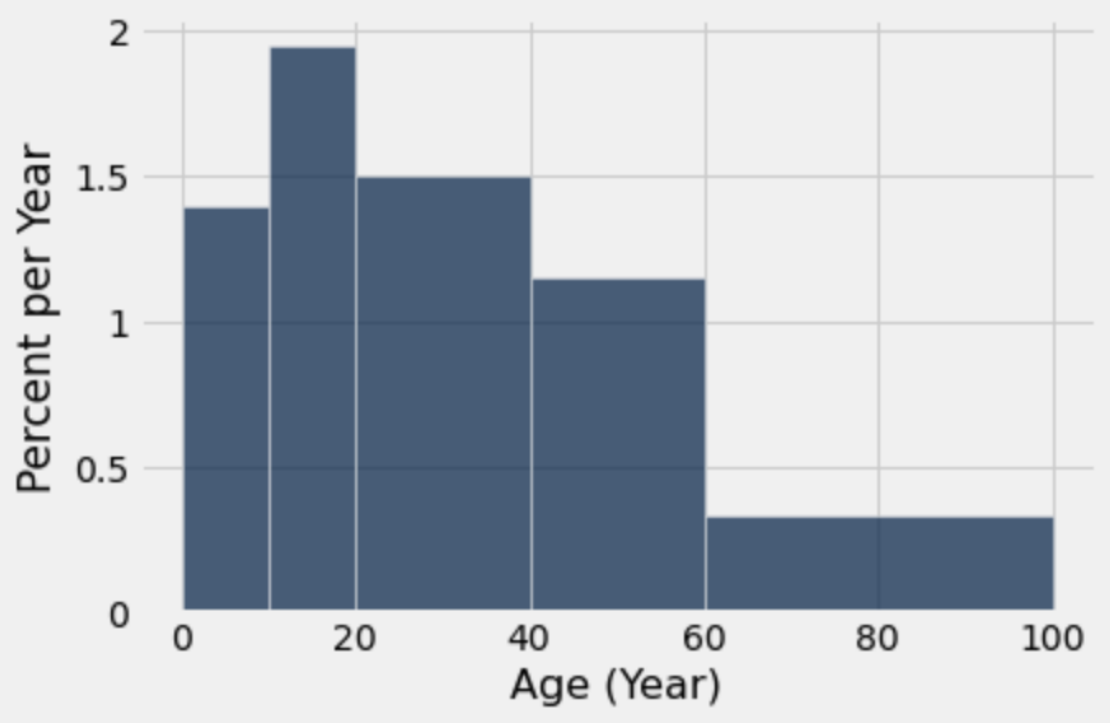
Bar Charts

```
in_millions = top10_adjusted.select('Title', 'Millions', 'Millions (Adjusted)')
in_millions.barh('Title')
```



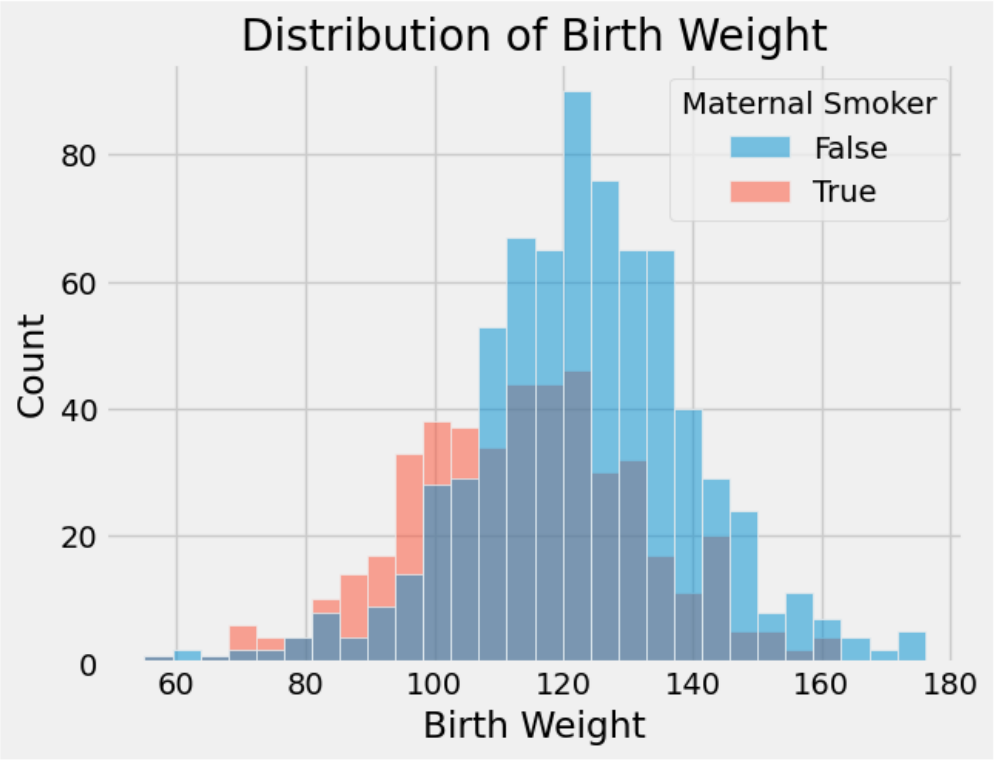
Histograms

```
top_movies.hist('Age', bins = my_bins, unit = 'Year')
```



Visualization

```
import seaborn as sns
sns.histplot(births_df, x="Birth Weight", hue="Maternal Smoker");
plt.title("Distribution of Birth Weight");
```



Visualizations in The Real World, Goals of Visualization

- **Visualizations**
  - **In the Real World, Goals**
- Distributions
- Bar Plots for Distributions
- Bar Plot Introspection
- Histograms
- Evaluating Histograms
- Box Plots and Violin Plots
- Comparing Quantitative Distribution

Boutique Visualizations

Visualizations can also be much more complex.

Examples:

- https://www.nytimes.com/interactive/2021/06/29/upshot/portland-seattle-vancouver-weather.html
- https://www.ft.com/content/a2901ce8-5eb7-4633-b89c-cbdf5b386938
- https://observablehq.com/@johnburnmurdoch/bar-chart-race-the-most-populous-cities-in-the-world
- https://projects.fivethirtyeight.com/2022-election-forecast/
- The John Hunter Excellence in Plotting Contest: https://jhepc.github.io/gallery.html

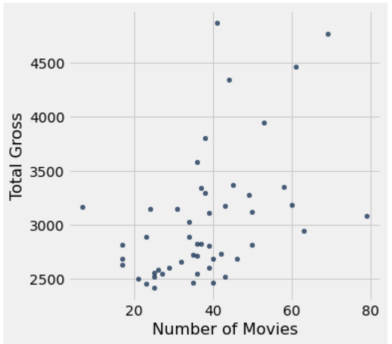
Goals of Data Visualization

Goal 1: To **help your own understanding** of your data/results.

- Key part of exploratory data analysis.
- Useful throughout modeling as well.
- Lightweight, iterative and flexible.

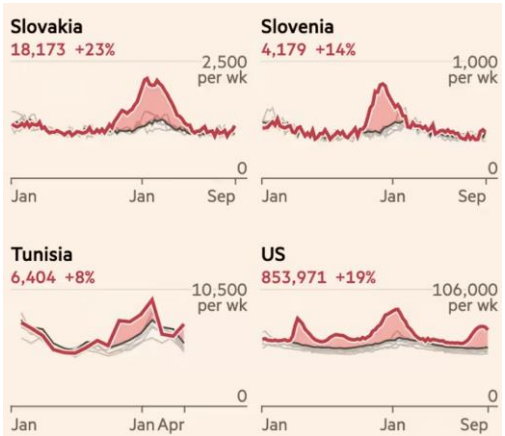
```
# The call is
# t.scatter(x_label, y_label)

actors.scatter('Number of Movies', 'Total Gross')
```



Goal 2: To **communicate results/conclusions to others**.

- Highly editorial and selective.
- Be thoughtful and careful!
- Fine-tuned to achieve a communications goal.
- Often time-consuming: bridges into design, even art.



A constant tool across the lifecycle of data science

Distributions

Lecture 07

- Visualizations
  - In Data 8 and Data 100
  - In the Real World, Goals
- **Distributions**
- Bar Plots for Distributions
- Bar Plot Introspection
- Histograms
- Evaluating Histograms
- Box Plots and Violin Plots
- Comparing Quantitative Distribution

What Is a Distribution?

As a case study, we'll devote a large fraction of our time today towards visualization of **distributions**.

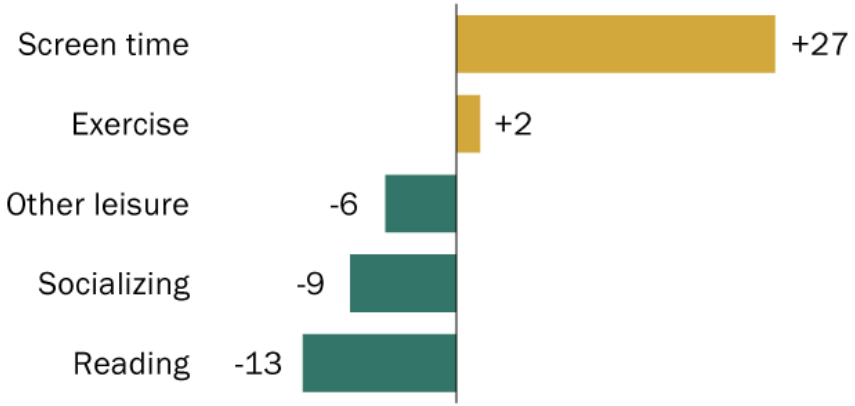
A **distribution** describes the frequency at which values of a variable occur.

- All values must be accounted for **once, and only once**.
- The total frequencies must **add up to 100%**, or to the number of values that we're observing.

Let's look at some examples.

For older Americans, leisure time looks different today than it did a decade ago

Change in daily time use 2005-2015 (minutes), for people 60 and older



Note: Based on non-institutionalized people.
Source: Pew Research Center analysis of 2003-2006 and 2014-2017 American Time Use Survey (IPUMS).

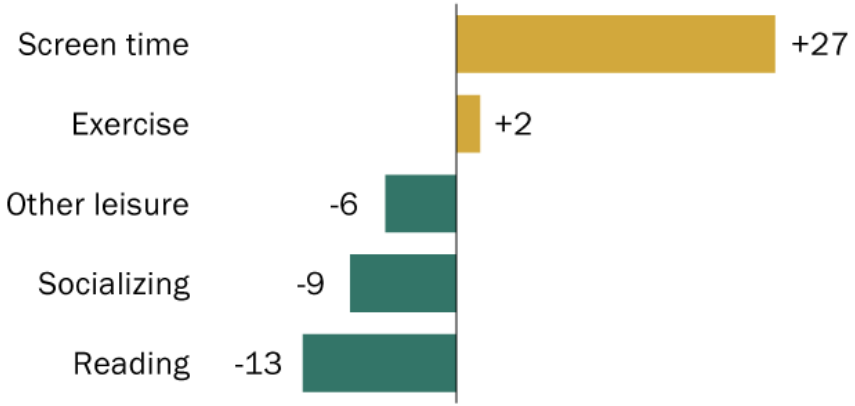
PEW RESEARCH CENTER

Does this chart show a distribution?

- Individuals can be in more than one category.
- The numbers (and bar lengths) correspond to “time”, not the proportion or number of individuals in the category.

For older Americans, leisure time looks different today than it did a decade ago

Change in daily time use 2005-2015 (minutes), for people 60 and older



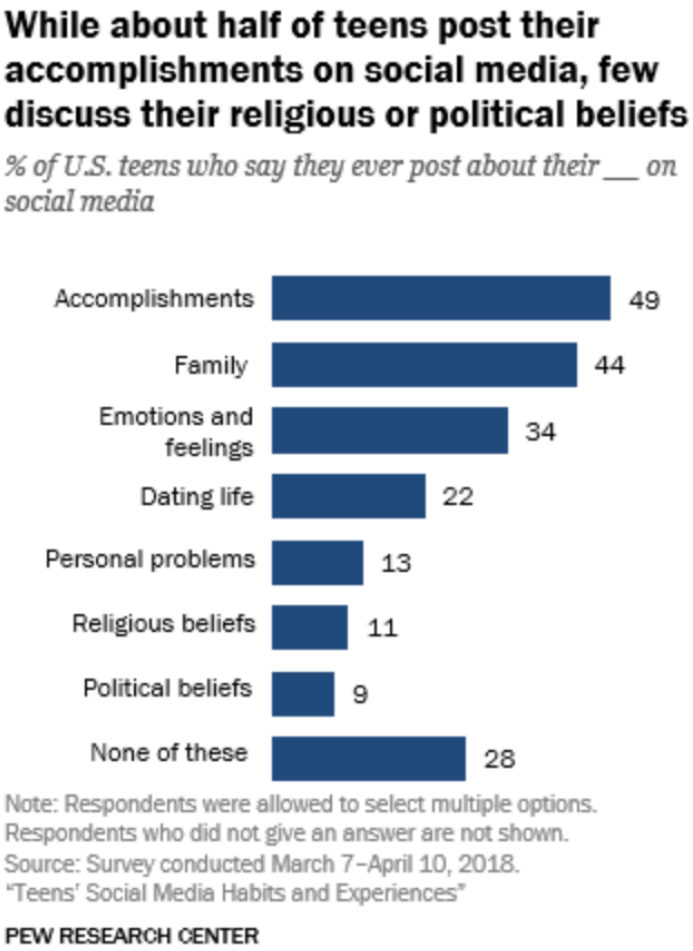
Note: Based on non-institutionalized people.
Source: Pew Research Center analysis of 2003-2006 and 2014-2017 American Time Use Survey (IPUMS).

PEW RESEARCH CENTER

Does this chart show a distribution?

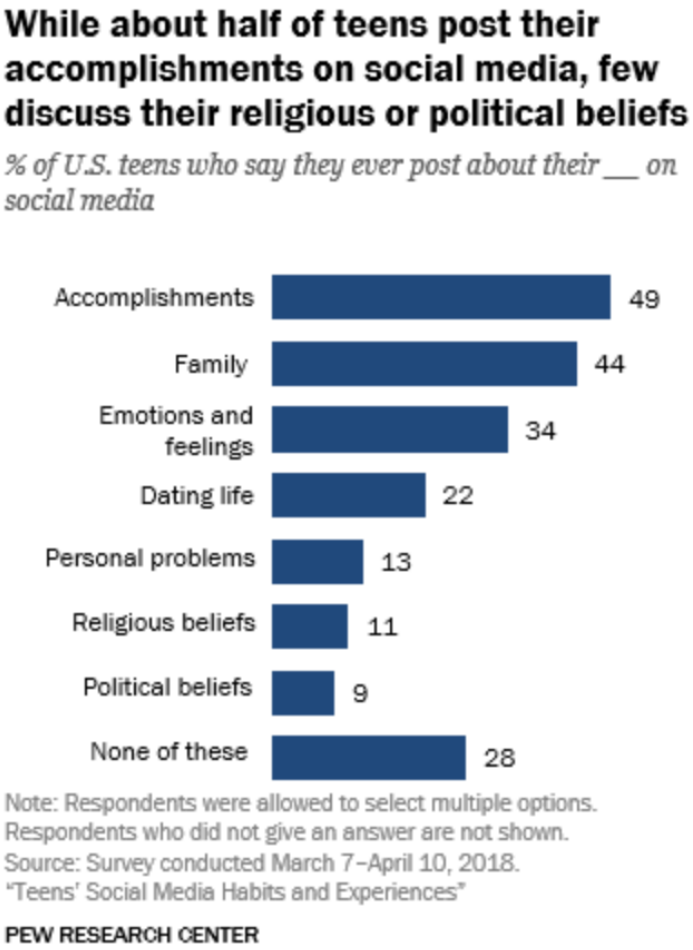
No.

- Individuals can be in more than one category.
- The numbers (and bar lengths) correspond to “time”, not the proportion or number of individuals in the category.



Does this chart show a distribution?

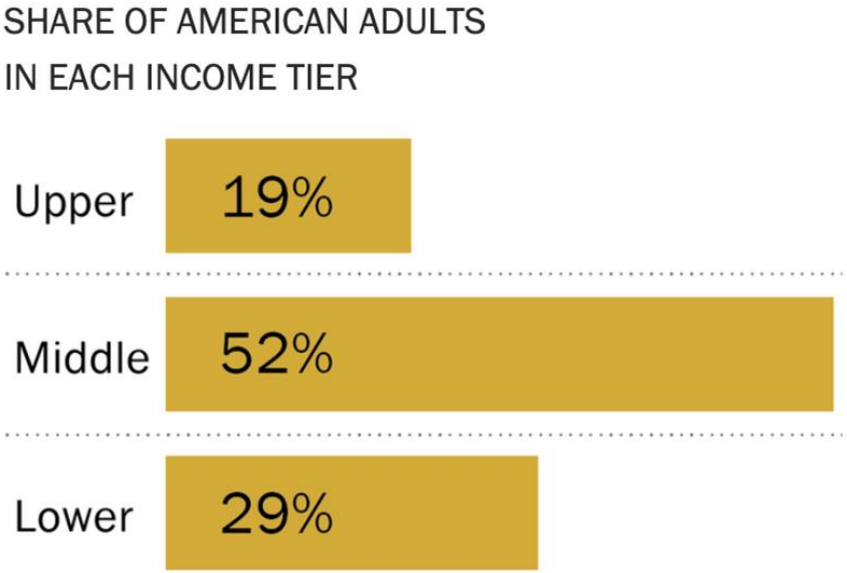
- The chart does show percents of individuals in different categories!



Does this chart show a distribution?

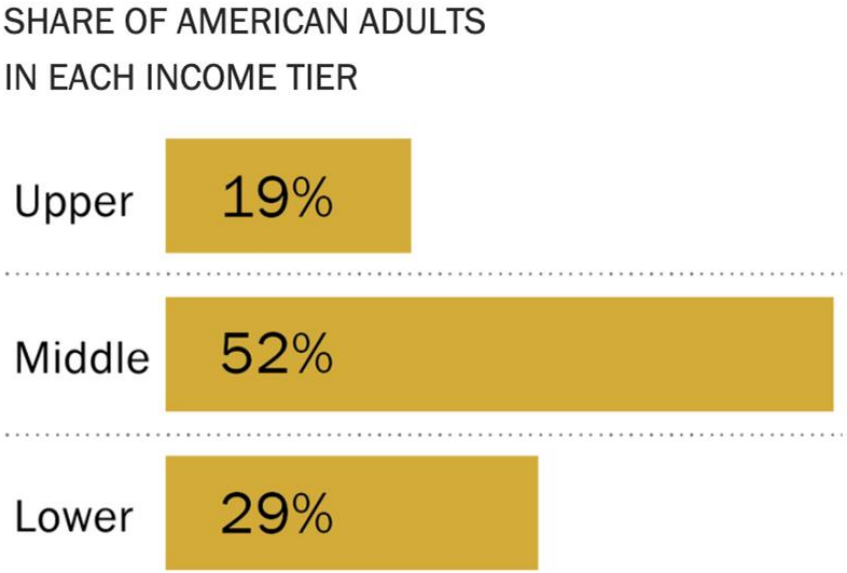
No.

- The chart does show percents of individuals in different categories!
- But, this is not a distribution because individuals can be in more than one category (see the fine print).



Does this chart show a distribution?

- This chart shows the qualitative ordinal variable "income tier."
- Each individual is in exactly one category.
- The values we see are the proportions of individuals in that category.



Does this chart show a distribution?

Yes!

- This chart shows the distribution of the qualitative ordinal variable "income tier."
- Each individual is in exactly one category.
- The values we see are the proportions of individuals in that category.
- Everyone is represented, as the total percentage is 100%.

Bar Plots for Distributions

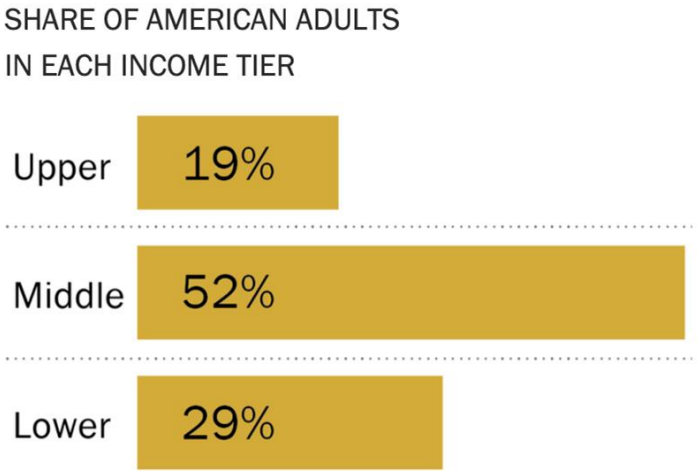
Lecture 07

- Visualizations
  - In Data 8 and Data 100
  - In the Real World, Goals
- Distributions
- **Bar Plots for Distributions**
- Bar Plot Introspection
- Histograms
- Evaluating Histograms
- Box Plots and Violin Plots
- Comparing Quantitative Distribution

Bar Plots

Bar Plots are the most common way of displaying the **distribution** of a **qualitative (categorical)** variable.

- For example, the proportion of adults in the upper, middle, and lower classes.
- **Lengths** encode **values**.
  - *Widths* encode *nothing*!
  - *Color* could indicate a sub-category (but not necessarily).



Example Dataset

We will be using the baby weights dataset from Data 8 for most of our plots today. Here is what that looks like.

```
1 births = pd.read_csv('baby.csv')
```

```
1 births.head()
```

	Birth Weight	Gestational Days	Maternal Age	Maternal Height	Maternal Pregnancy Weight	Maternal Smoker
0	120	284	27	62	100	False
1	113	282	33	64	135	False
2	128	279	28	64	115	True
3	108	282	23	67	125	True
4	136	286	25	62	93	False

```
1 births.shape
```

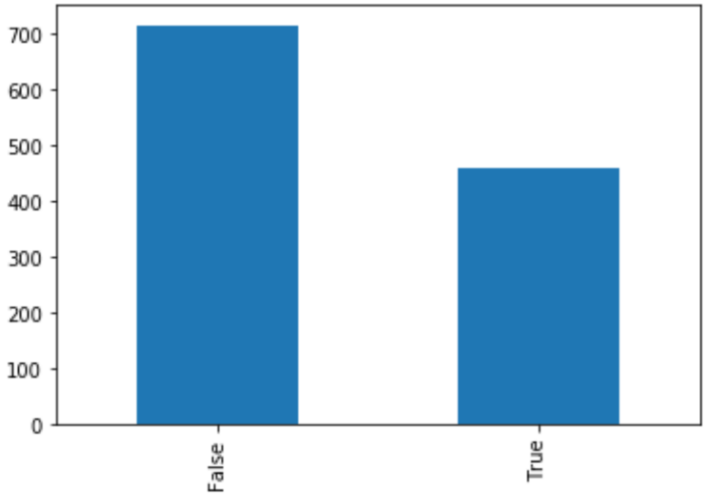
```
(1174, 6)
```

Bar Plots

Suppose **`births['Maternal Smoker']`** is a series containing True and False.

```
births['Maternal Smoker'].value_counts()
```

```
False    715
True     459
Name: Maternal Smoker, dtype: int64
```



We can also visualize with a bar plot.

- What's better about the visualization?
- What's worse?

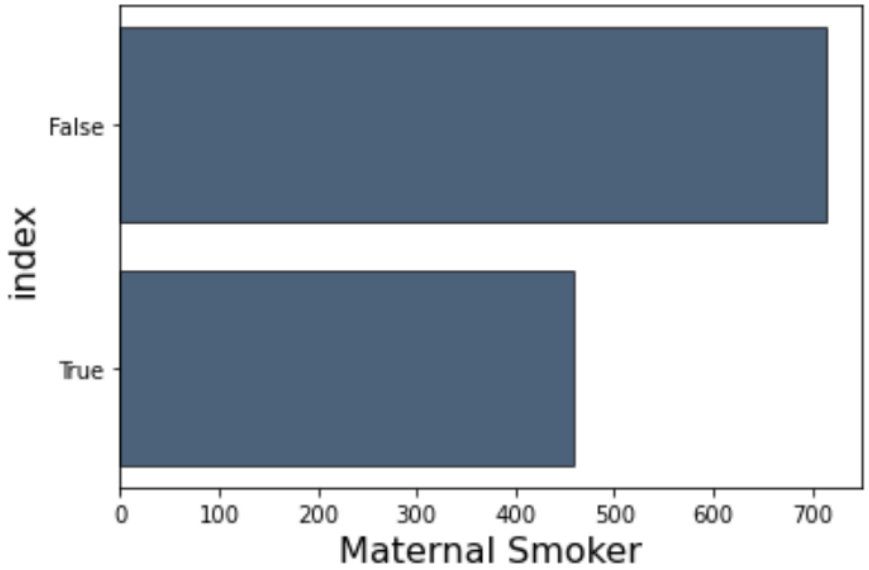
Generating Bar Plots in Python

In our DataHub environment, there are many ways we could generate this bar code.

- See the companion notebook to experiment.

Generating Bar Plots in Python (Data 8 Tables)

Table example:

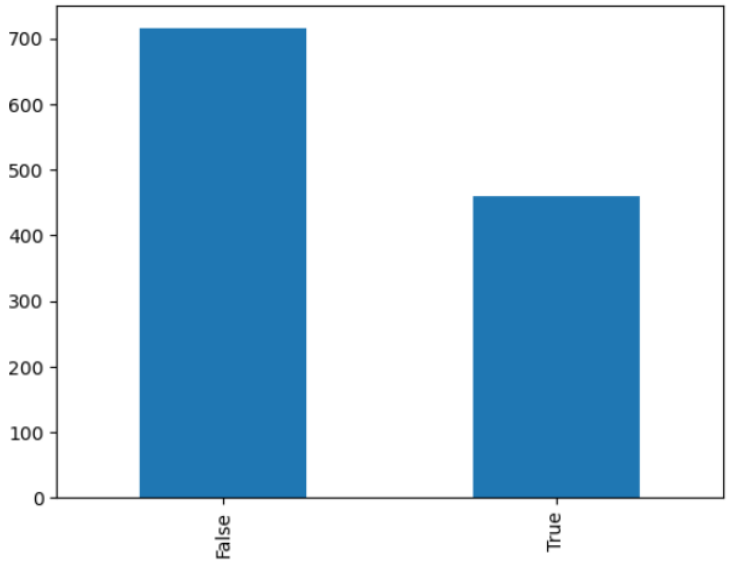


Note: For some unknown reason, the vertical bar version looks bad with the Data8 library.

```
from datascience import Table
t = Table.from_df(births['Maternal Smoker'].value_counts().reset_index())
t.barh("index", "Maternal Smoker")
```

Generating Bar Plots in Python (Pandas Native)

Pandas dataframes and series have plotting methods, similar to the Table class

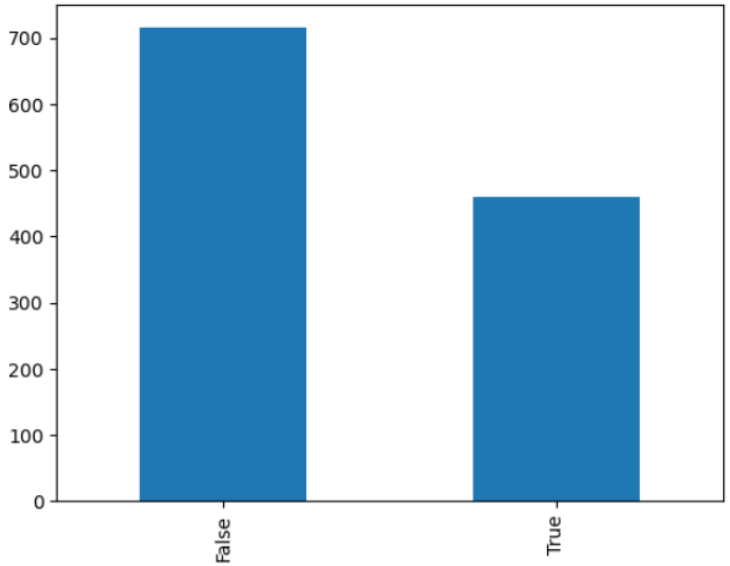



```
births['Maternal Smoker'].value_counts().plot(kind = 'bar')
```

Generating Bar Plots in Python (Matplotlib Manual)

We could also directly invoke the matplotlib's `bar` function.

- We'll do this sometimes in Data 100, but not often.

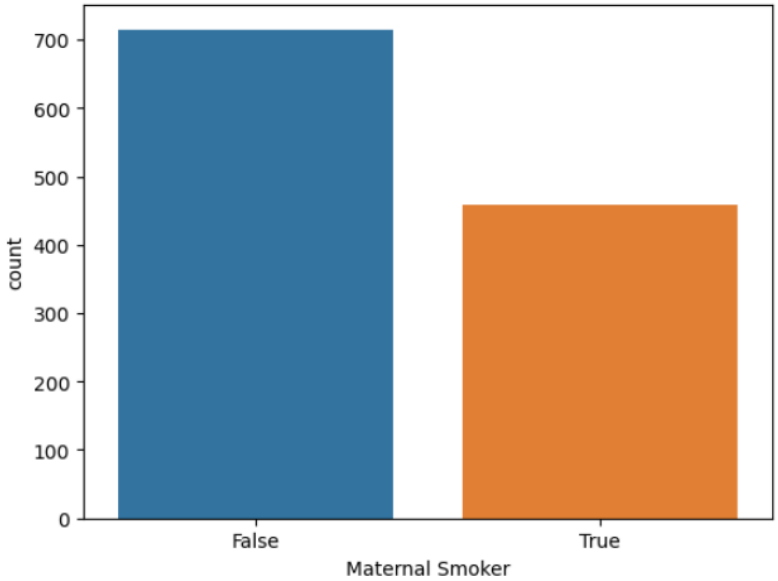



```
ms = births['Maternal Smoker'].value_counts();
plt.bar(ms.index.astype('string'), ms);
```

Generating Bar Plots in Python (Seaborn)

We could use the Seaborn library's `countplot` function.

- Preferred approach



`countplot` operates at a higher level of abstraction!

You give it the entire dataframe and it does the counting for you.

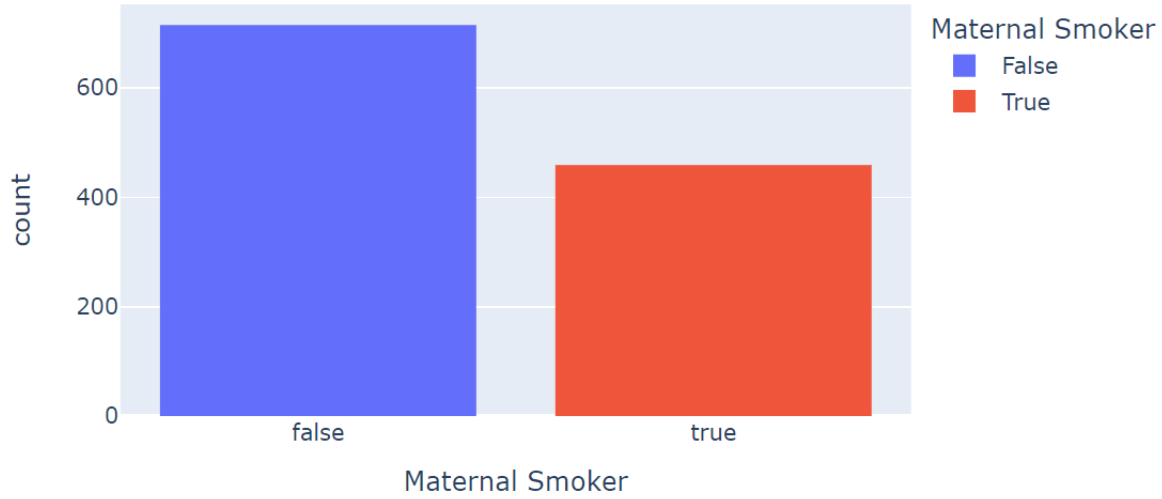
```
import seaborn as sns

sns.countplot(data = births, x = 'Maternal Smoker');
```

Generating Bar Plots in Python (Plotly)

We could use the plotly library's **`histogram`** function.

- it's an important emerging tool to know.
- Interactive and cross-language!




```
import plotly.express as px

px.histogram(births, x = 'Maternal Smoker', color = 'Maternal Smoker')
```

Bar Plot Introspection

Lecture 07

- Visualizations
  - In Data 8 and Data 100
  - In the Real World, Goals
- Distributions
- Bar Plots for Distributions
- **Bar Plot Introspection**
- Histograms
- Evaluating Histograms
- Box Plots and Violin Plots
- Comparing Quantitative Distributions

Generating Bar Plots in Python

We've seen five different ways to generate the same plot. The first four are matplotlib based:

- Data 8 Table library. (`Table.bar`).
- Pandas bar function (`Series.bar`).
- Matplotlib bar function (`plt.bar`).
- **Seaborn countplot function (`sns.countplot`).**
  - Uses matplotlib as its rendering engine.
  - Preferred approach. Simpler to use. Superior aesthetics.

And the last is a non-matplotlib library:

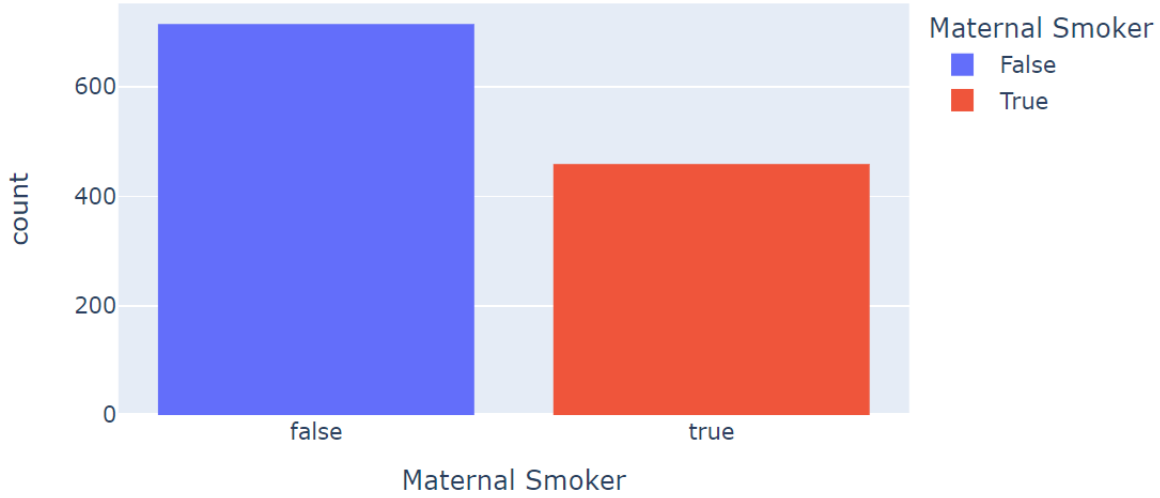
- Plotly histogram function (`px.histogram`).
  - Aesthetics are often superior to seaborn.
  - Plots are interactive.
  - API is cleaner in some areas than matplotlib based libraries.

Why not change to plotly? Extremely time-consuming to change the entire infrastructure of the class.
Also - mpl is the standard in python scientific viz.

Questions About Our Bar Plot

Observations:

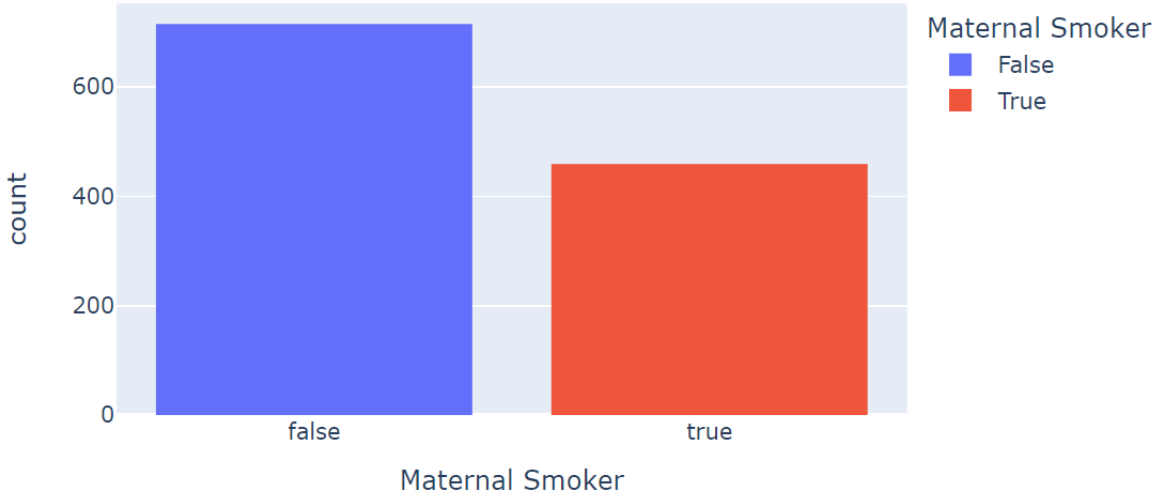
- The colors that were chosen (blue and red) are totally arbitrary and were not intended to convey any specific information, other than that there are two distinct categories.
- The widths of the bars encode no useful information and their arbitrarily chosen width is just to look nice.



Questions About Our Bar Plot

Questions:

- What colors should we use?
- How wide should the bars be?
- Should the legend exist?
- Should the bars and axes have dark borders?



How do we decide? Are these purely aesthetic questions?

Reminder: Goals of Data Visualization

Goal 1: To **help your own understanding** of your data/results.

- Key part of exploratory data analysis.
- Useful throughout modeling as well.
- Lightweight, iterative and flexible.

Goal 2: To **communicate results/conclusions to others**.

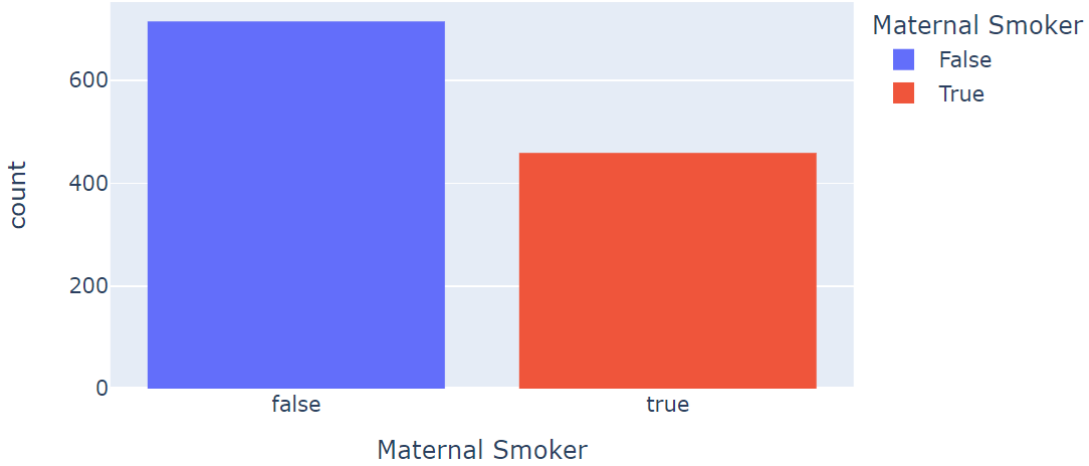
- Highly editorial and selective.
- Be thoughtful and careful!
- Fine tuned to achieve a communications goal.
- Often time-consuming: bridges into design, even art.

A constant tool across the lifecycle of data science

Questions About Our Bar Plot

Questions:

- What colors should we use?
- How wide should the bars be?
- Should the legend exist?
- Should the bars and axes have dark borders?

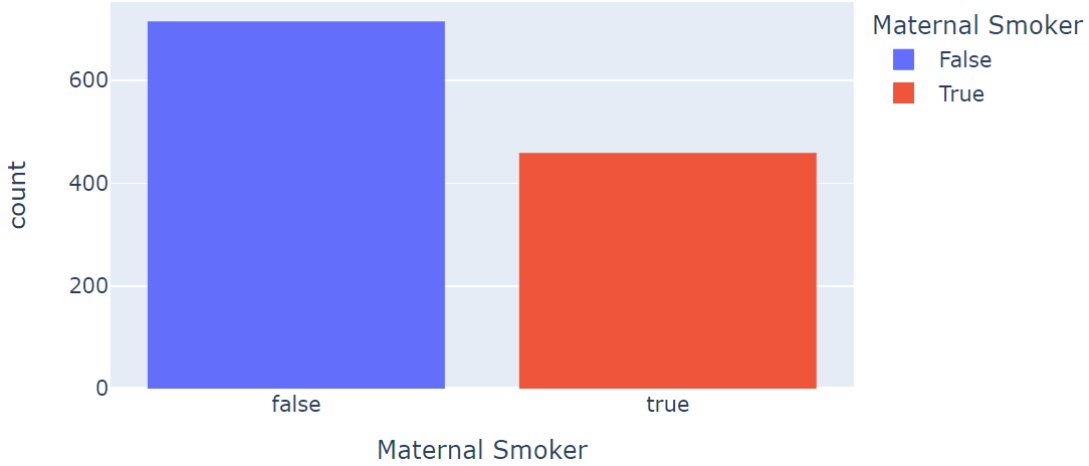


For goal 1, our choices are (IMO) irrelevant.

Questions About Our Bar Plot

Questions:

- What colors should we use?
- How wide should the bars be?
- Should the legend exist?
- Should the bars and axes have dark borders?

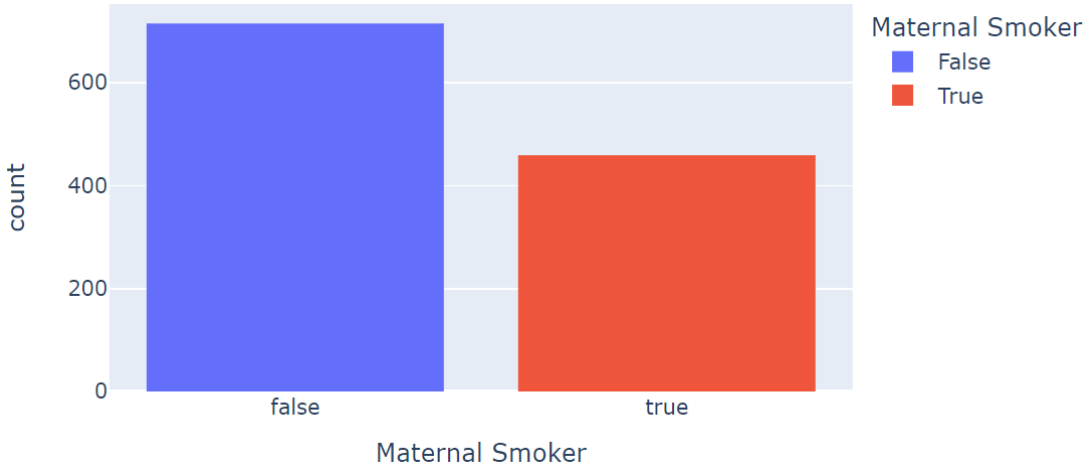


For goal 2, they matter. Let's discuss!

Questions About Our Bar Plot

Questions:

- What colors should we use?
- How wide should the bars be?
- Should the legend exist?
- Should the bars and axes have dark borders?



For goal 2, they matter. Let's discuss!

Thoughts:

- Label the bars
- Increase the font size
- Changing colors: green/red
- Color-blindness
- Adding color gradients

Histograms

Lecture 07

- Visualizations
  - In Data 8 and Data 100
  - In the Real World, Goals
- Distributions
- Bar Plots for Distributions
- Bar Plot Introspection
- **Histograms**
- Evaluating Histograms
- Box Plots and Violin Plots
- Comparing Quantitative Distributions

Questions About Other Features

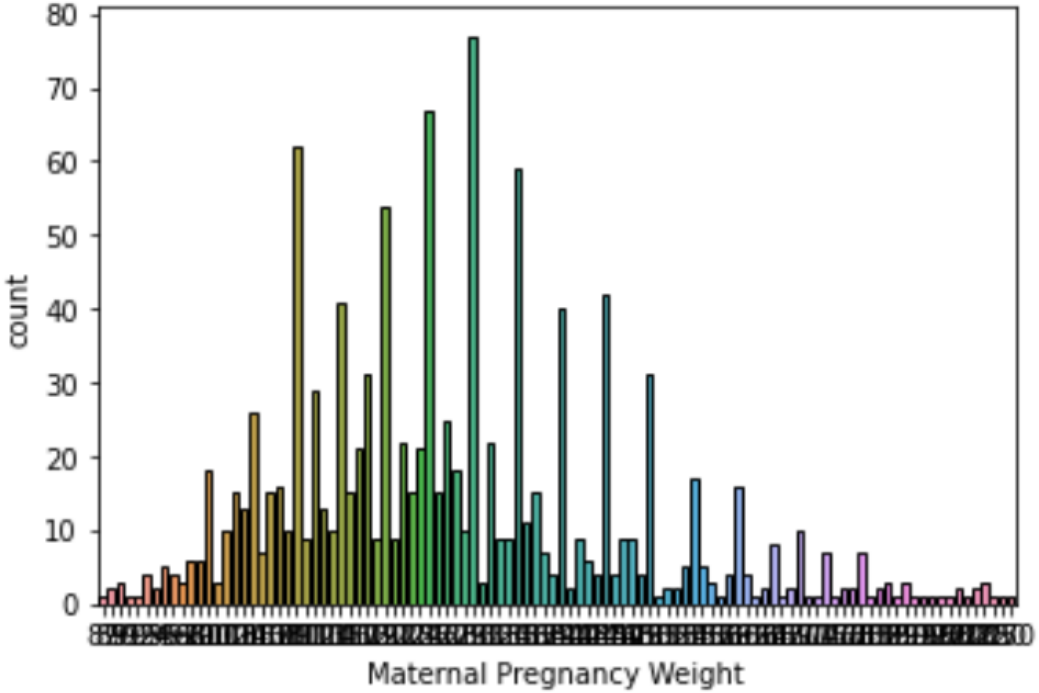
Our data set has many other features.

	Birth Weight	Gestational Days	Maternal Age	Maternal Height	Maternal Pregnancy Weight	Maternal Smoker
0	120	284	27	62	100	False
1	113	282	33	64	135	False
2	128	279	28	64	115	True
3	108	282	23	67	125	True
4	136	286	25	62	93	False
...	...	...	...	...	...	...

Suppose we want to plot the distribution of `"Maternal Pregnancy Weight"` as a bar plot.

- What are our "categories"? One natural choice: Each integer value, e.g. 100 is a category, 135 is a category, etc.

Maternal Pregnancy Weight



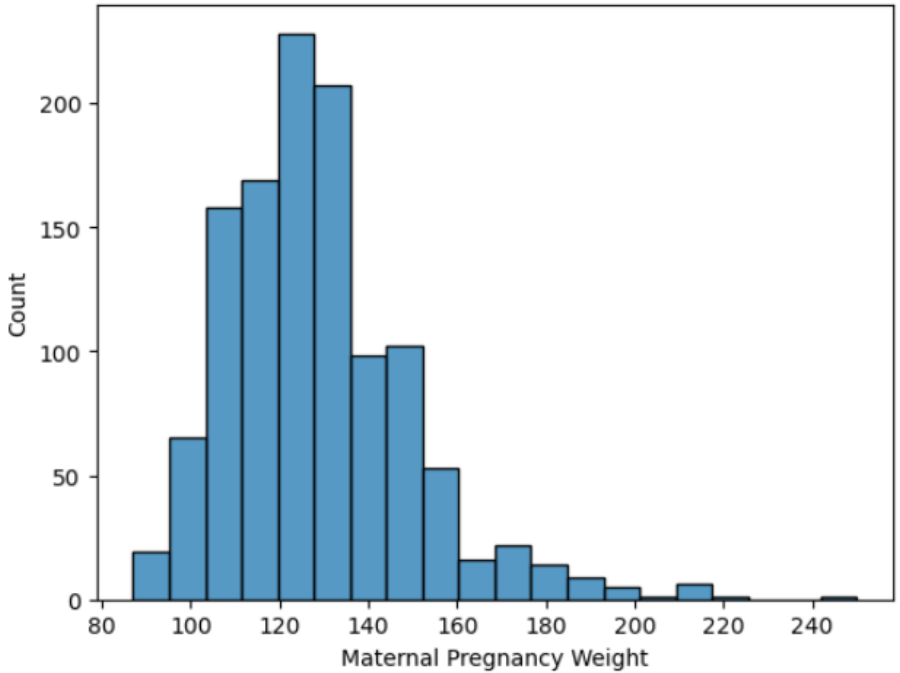
If we use each observed integer value as a category.

What's wrong here?

```
sns.countplot(data = births, x = 'Maternal Pregnancy Weight');
```

Maternal Pregnancy Weight

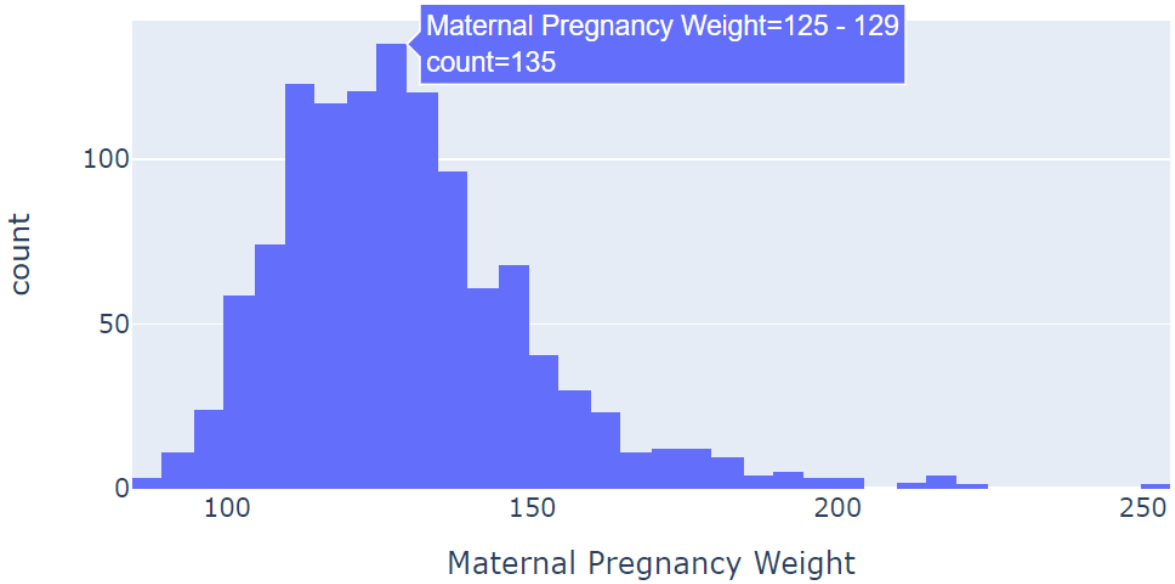
Quick note: These weights are **self-reported pre-pregnancy weights** from the 1960s!



If we use bins of integer values as a category.

```
sns.histplot(data = births, x = 'Maternal Pregnancy Weight', bins = 20);
```

Maternal Pregnancy Weight (in Plotly)



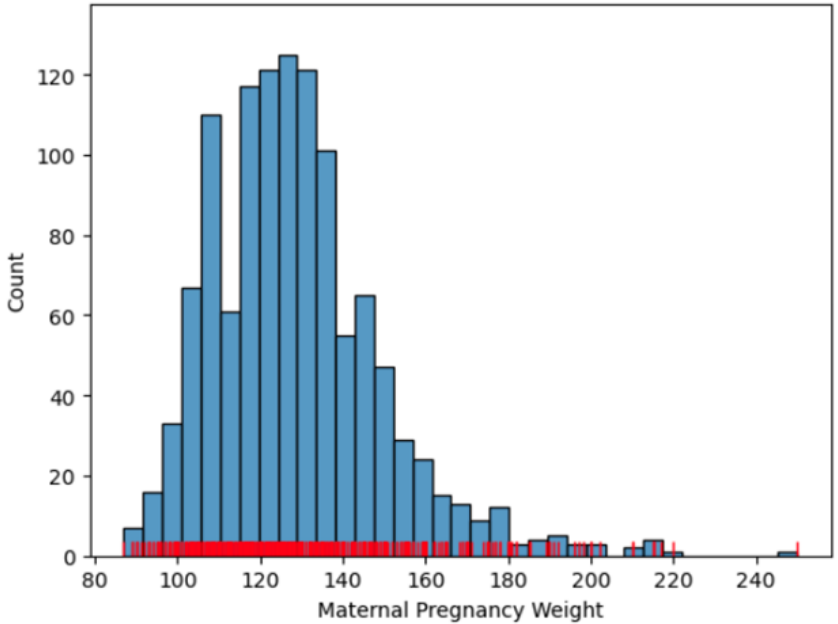
Even though we're not using plotly officially in Data100, being able to mouseover and see bins is so handy.

```
px.histogram(births, x = 'Maternal Pregnancy Weight')
```

Rugplots

We can add even more information to the bar plot.

- An overlaid "**`rug plot`**" lets us see the distribution of data points within each bin.



Not clear to me that the rug plot is useful in this context! But it's an enhancement you might find useful sometimes.

```
sns.histplot(data = births, x = 'Maternal Pregnancy Weight');
sns.rugplot(data = births, x = 'Maternal Pregnancy Weight', color = 'red');
```

Evaluating Histograms

Lecture 07

- Visualizations
  - In Data 8 and Data 100
  - In the Real World, Goals
- Distributions
- Bar Plots for Distributions
- Bar Plot Introspection
- Histograms
- **Evaluating Histograms**
- Box Plots and Violin Plots
- Comparing Quantitative Distributions

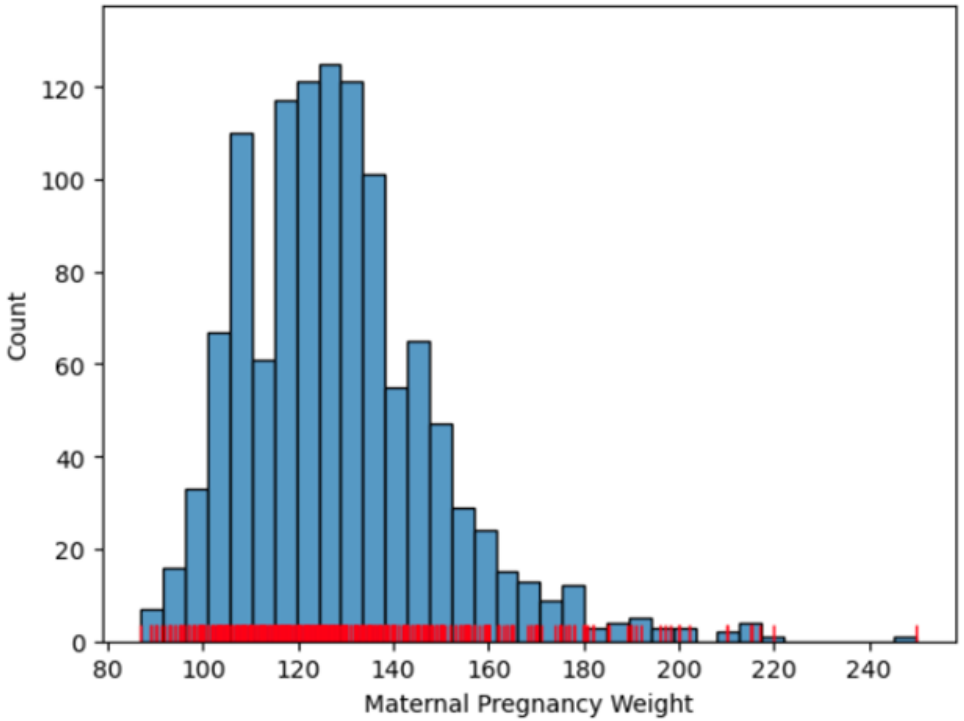
Describing Distributions

Histograms allow us to assess a distribution by their shape.

Some of the terminology we use to describe distributions:

- **Skewness and Tails.**
  - Skewed left vs skewed right.
  - Left tail vs right tail.
- **Outliers.**
  - We'll define these arbitrarily.
- **Modes.**

Skewness and Tails

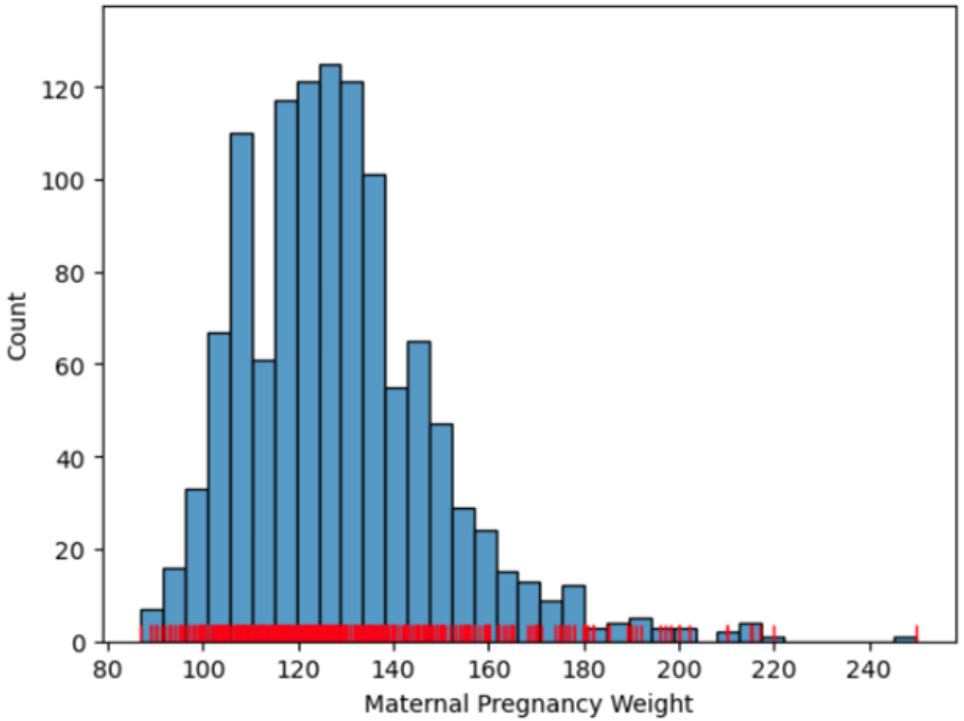


Median: 125, Mean: 128.48

If a distribution has a **long right tail**, we call it **skewed right.**

- Mean is typically to the right of the median.
  - Think of the mean as the "balancing point" of the density.
- If the **tail is on the left**, we say the data is **skewed left**.
- Our distribution can be also **symmetric**, when both tails are of equal size.

Outliers

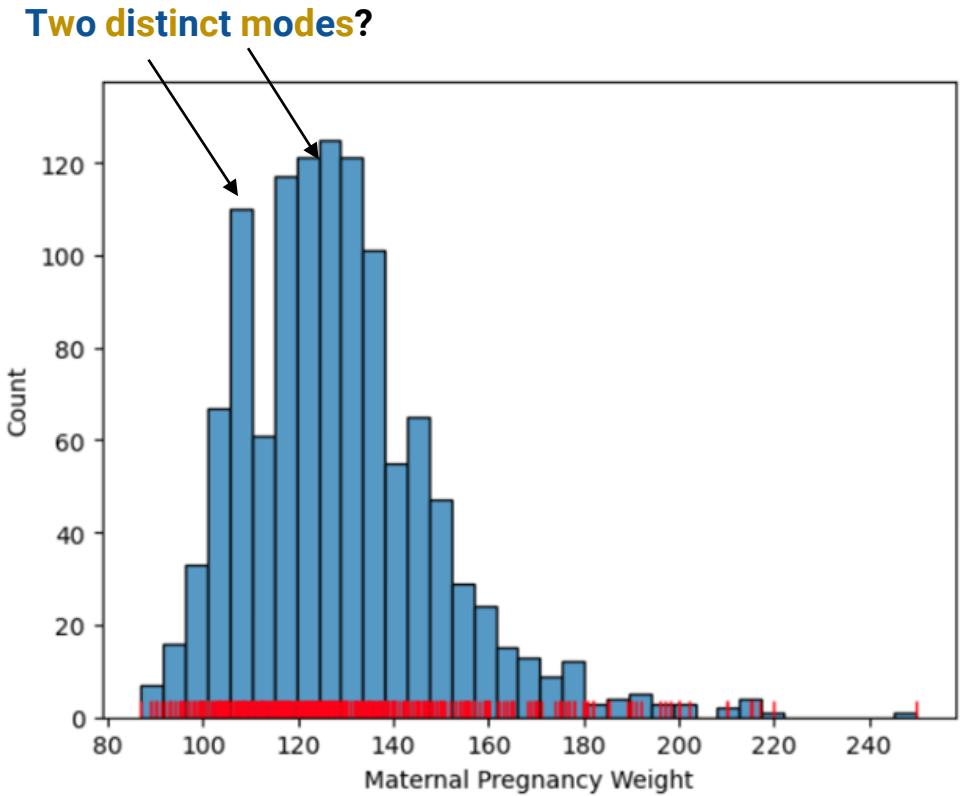


Median: 125, Mean: 128.48

This visualization lets us see **outlier(s)** in this sample on the far right.

- What constitutes an outlier is a choice we have to make.
  - Just the largest point?
  - The rightmost 4 bins?
- Will define outliers more carefully later when we talk about box plots.

Modes



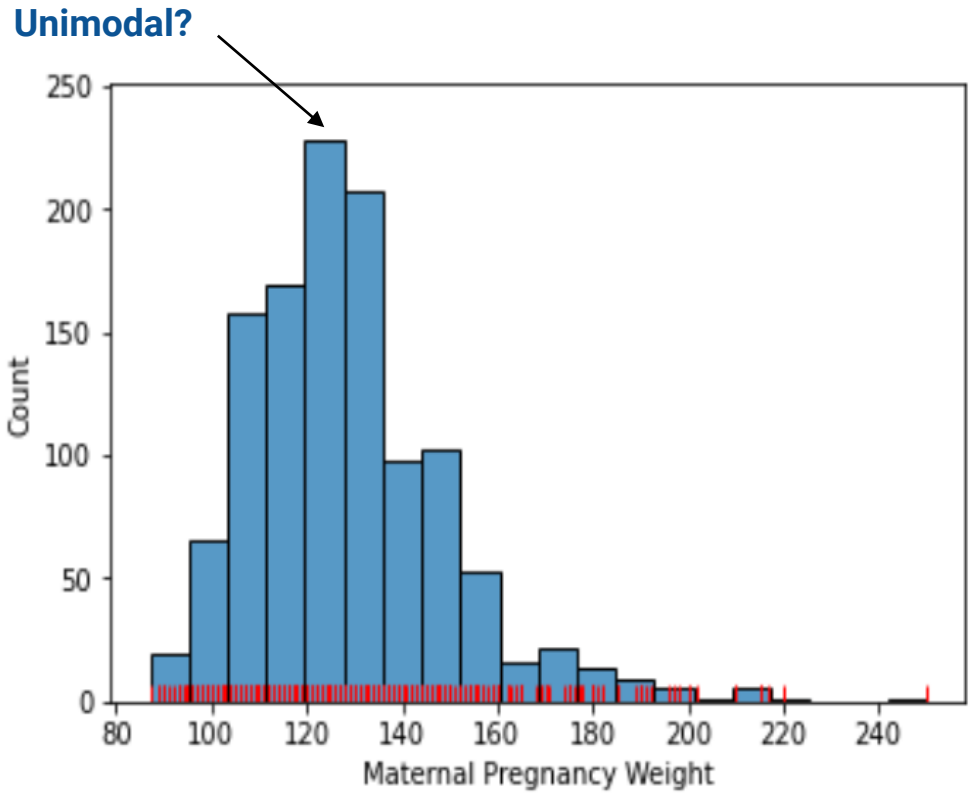
A **mode** of a distribution is a local or global maximum.

- A distribution with a single clear maximum is called unimodal.
- Distributions with two modes are called bimodal.
  - More than two: multimodal.
- Need to distinguish between **modes** and **random noise**.

Any ideas for how we can tell whether or not this is truly bimodal?

```
sns.histplot(data = births, x = 'Maternal Pregnancy Weight');
sns.rugplot(data = births, x = 'Maternal Pregnancy Weight', color = 'red');
```

Modes



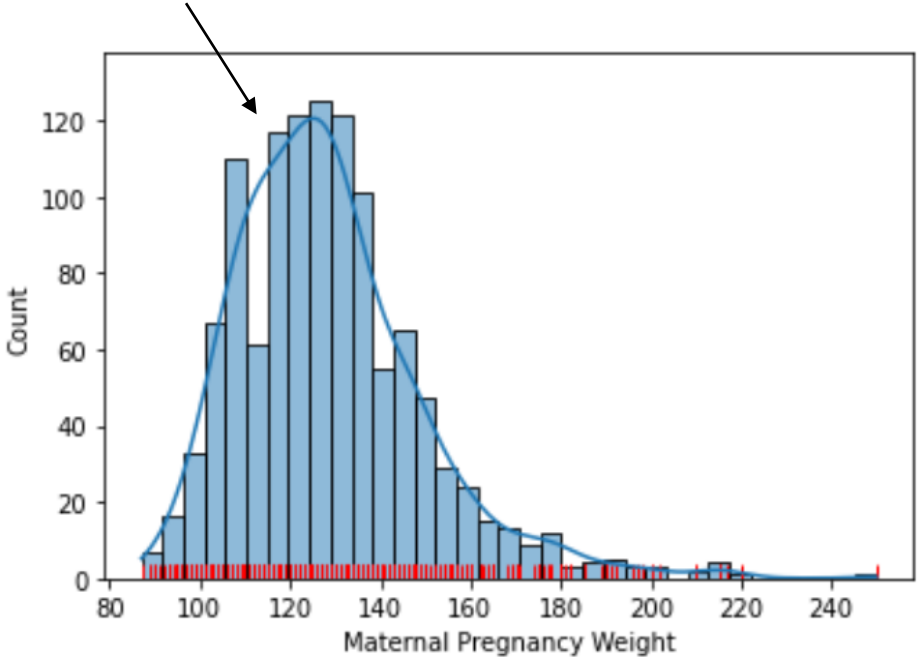
A **mode** of a distribution is a local or global maximum.

- A distribution with a single clear maximum is called unimodal.
- Distributions with two modes are called bimodal.
  - More than two: multimodal.
- Need to distinguish between **modes** and **random noise**.

```
sns.histplot(data = births, x = 'Maternal Pregnancy Weight', bins = 20);
sns.rugplot(data = births, x = 'Maternal Pregnancy Weight', color = 'red');
```

Density Curves

Unimodal



Instead of a discrete histogram, we can visualize what a continuous distribution corresponding to that same histogram could look like...

- The smooth curve drawn on top of the histogram here is called a **density curve**.

In lecture 8, we will study how exactly to compute these density curves (using a technique is called Kernel Density Estimation).

```
sns.histplot(data = births, x = 'Maternal Pregnancy Weight', kde = True);
sns.rugplot(data = births, x = 'Maternal Pregnancy Weight', color = 'red');
```

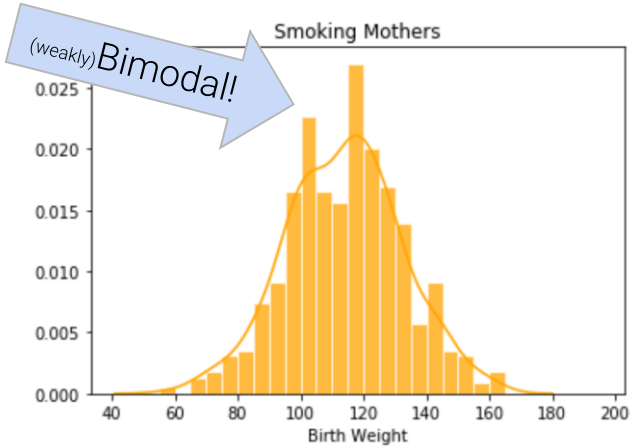
KDEs and Modes

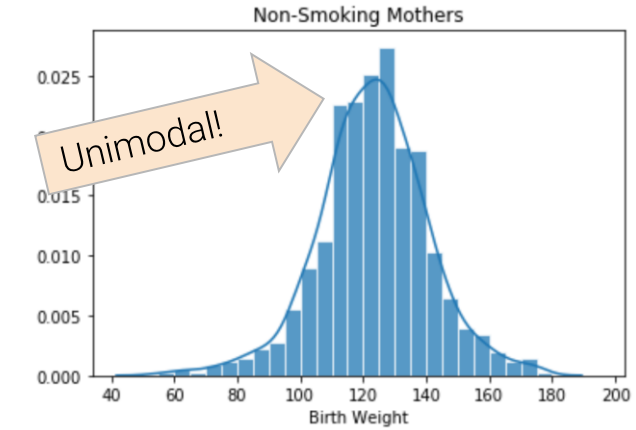
Example: If we plot birth weights of babies of smoking mothers, we get a histogram that appears bimodal.

- Density curve reinforces belief in this bimodality.

From a goal 1 perspective, this is EDA which tells us there may be something interesting here worth pursuing.

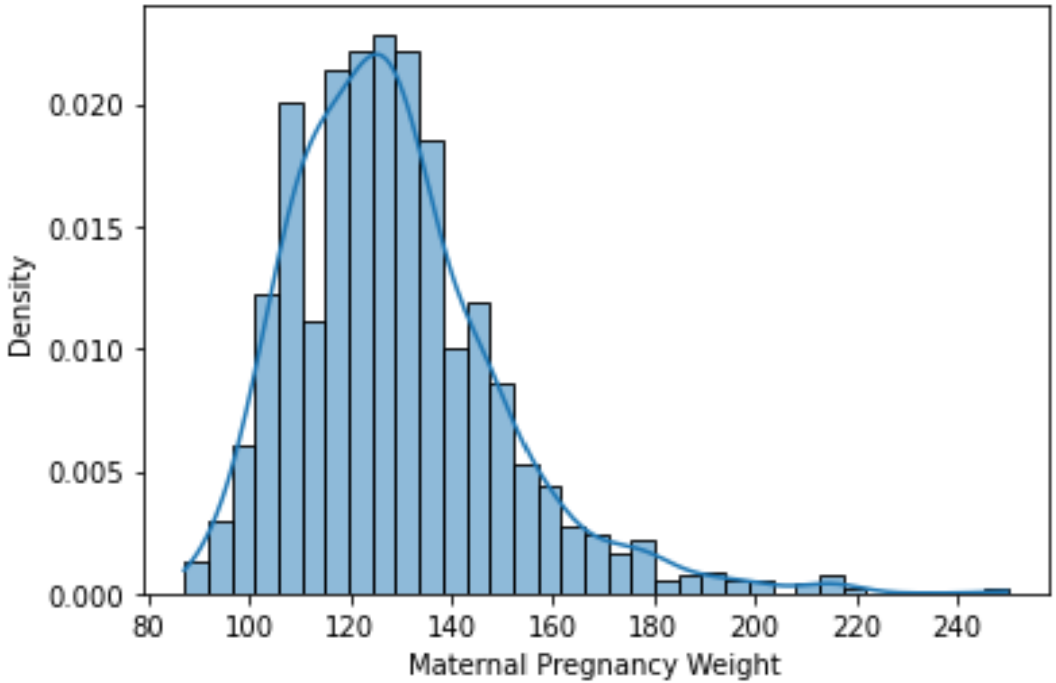
- Deeper analysis necessary!
- If we found something truly interesting, we'd have to cautiously write up an argument and create goal 2 level visualizations.





Reminder: Histograms and Density

Rather than labeling by counts, we can instead plot the density, as shown below:



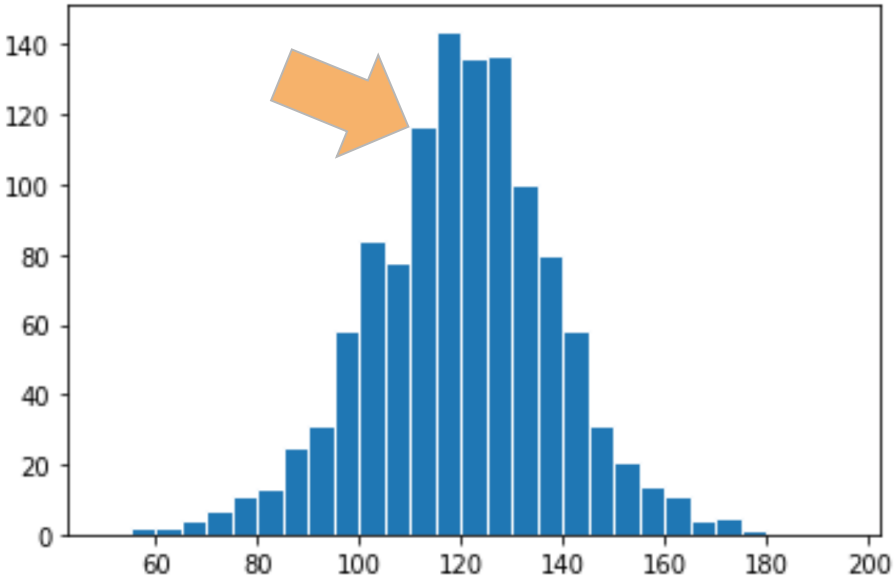

```
sns.histplot(data = births, x = 'Maternal Pregnancy Weight',
             kde = True, stat = 'density');
```

Data 8 Review Calculation: Computing Density from Counts and Bin Size

Approximately ~120 babies were born with a weight between 110 and 115.

There are 1174 observations total.

- Total area of this bin should be:
  - 120/1174 = ~10%
- Density of this bin is therefore:
  - 10% / (115 - 110) = 0.02

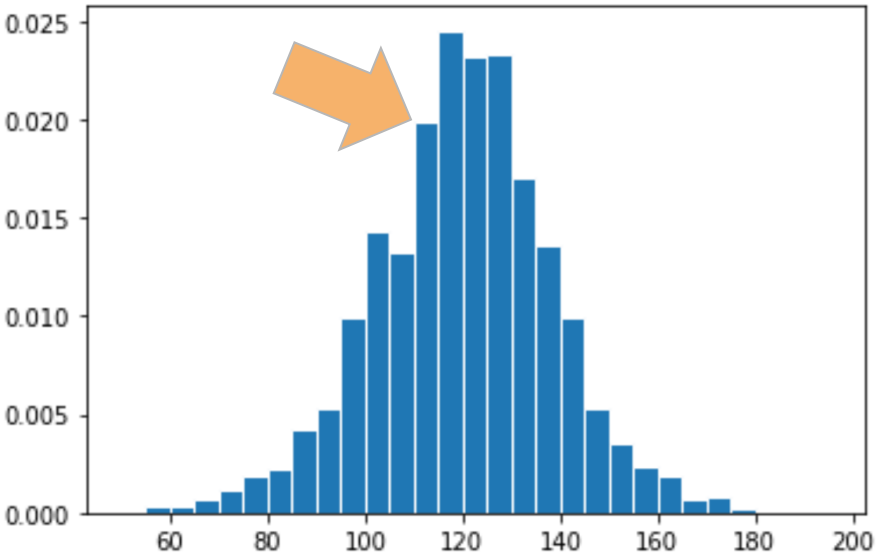


Data 8 Review Calculation: Computing Count from Bin Size and Density

There are 1174 observations total.

- Width of bin [110, 115): 5
- Height of bar [110, 115): 0.02
- Proportion in bin = 5 * 0.02 = 0.1
- Number in bin = 0.1 * 1174 = **117.4**

This is roughly the number we got before (120)!



Box Plots and Violin Plots

Lecture 07, Data 100 Spring 2023

- Visualizations
  - In Data 8 and Data 100
  - In the Real World, Goals
- Distributions
- Bar Plots for Distributions
- Bar Plot Introspection
- Histograms
- Evaluating Histograms
- **Box Plots and Violin Plots**
- Comparing Quantitative Distributions

Quartiles

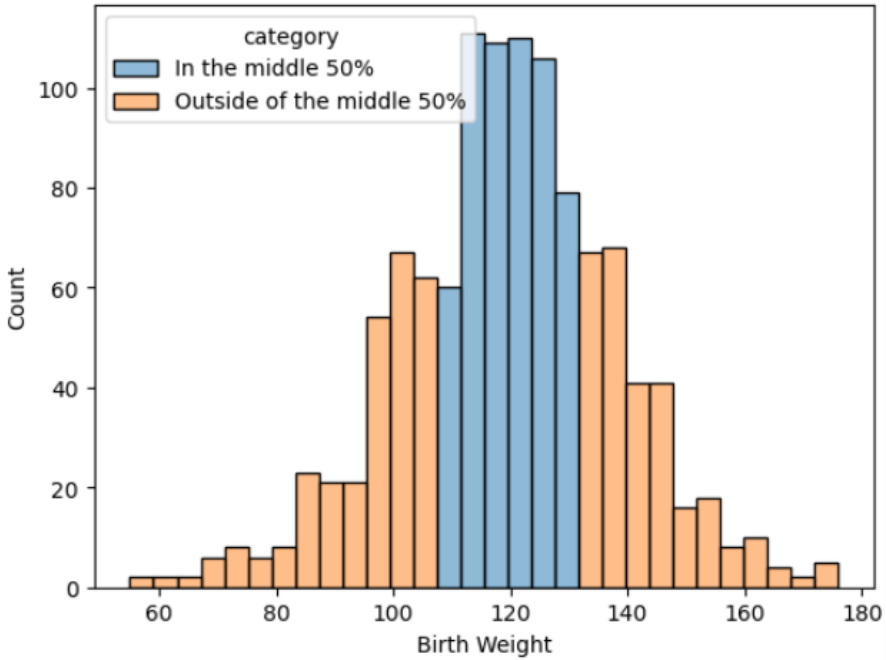
For a quantitative variable:

- First or lower quartile: 25th percentile
- Second quartile: 50th percentile (median)
- Third or upper quartile: 75th percentile

The interval [first quartile, third quartile] contains the "middle 50%" of the data.

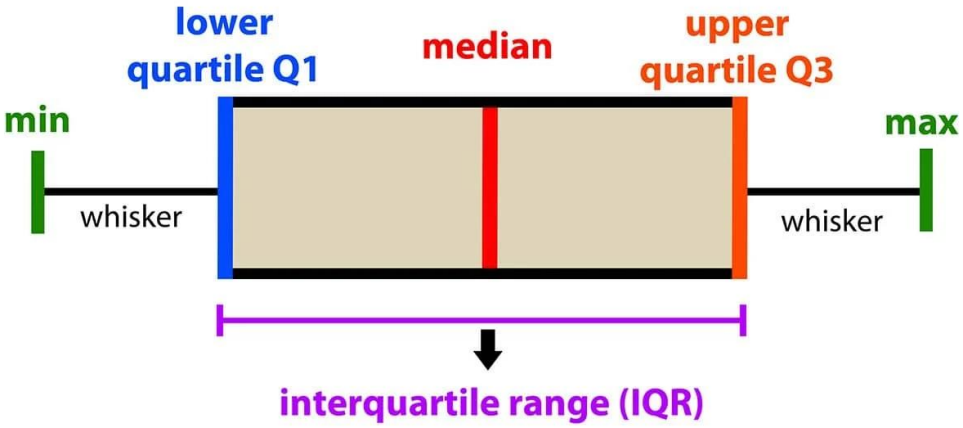
Interquartile range (IQR) measures spread.

- IQR = third quartile – first quartile.

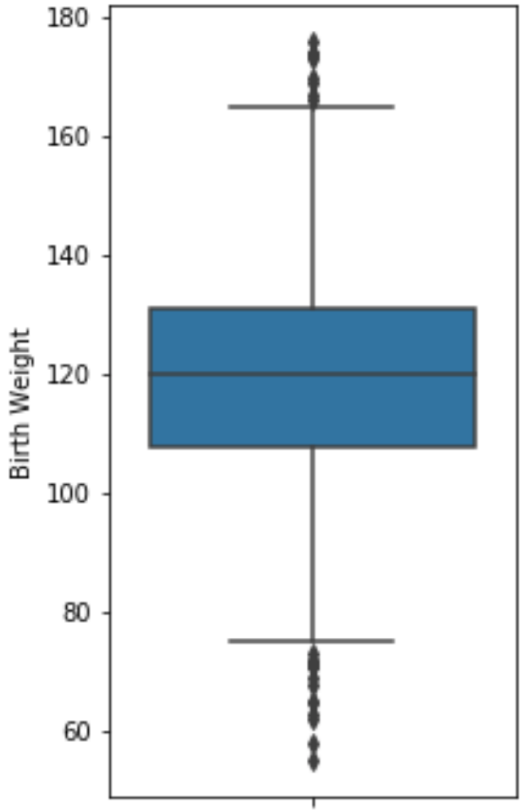


Note: We're now using the baby's weight rather than the mother's.

introduction to data analysis: Box Plot



Box Plots

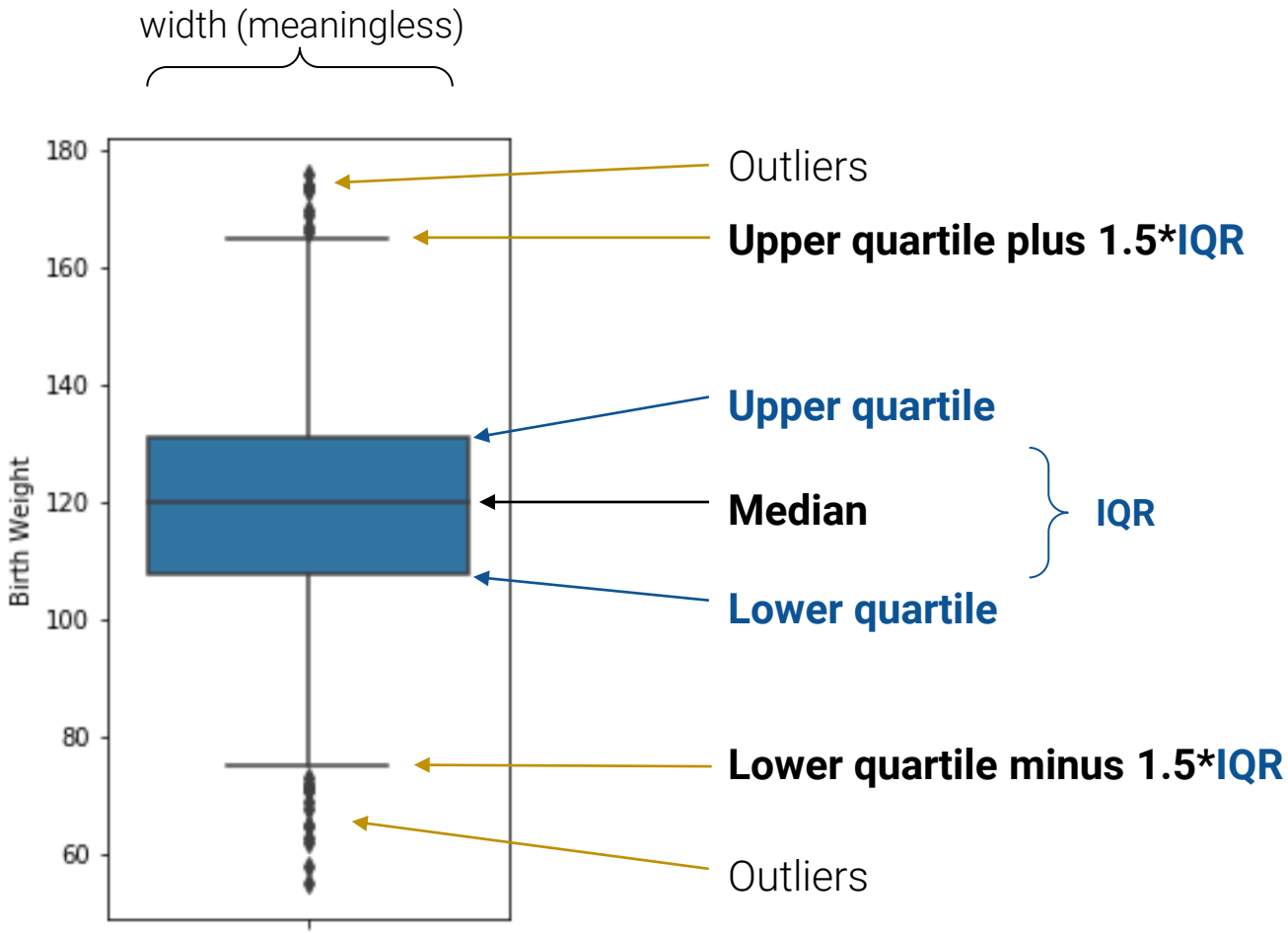


Box plots summarize several characteristics of a numerical distribution. They visualize:

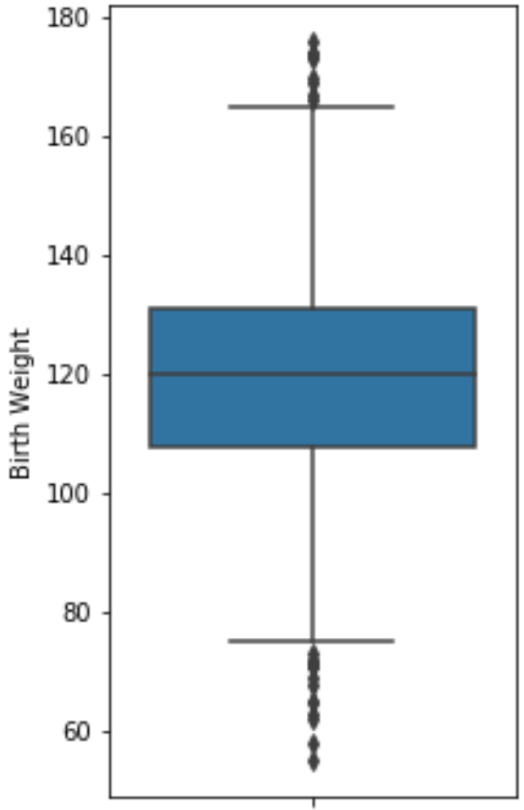
- **Lower quartile**.
- **Median**.
- **Upper quartile**.
- "**Whiskers**", placed at lower quartile minus 1.5*IQR and upper quartile plus 1.5*IQR.
- **Outliers**, which are defined as being further than 1.5*IQR from the extreme quartiles. Arbitrary definition!
- We lose a lot of information, too!

```
sns.boxplot(data = births, y = 'Birth Weight')
```

Box Plots


width (meaningless)
180
160
140
120
100
80
60
Birth Weight
Outliers
Upper quartile plus 1.5*IQR
Upper quartile
Median
IQR
Lower quartile
Lower quartile minus 1.5*IQR
Outliers

Box Plots

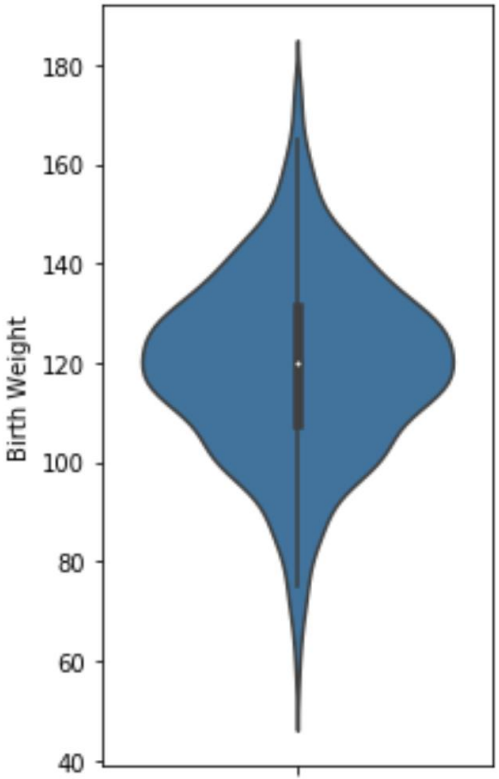



```
1  q1 = np.percentile(bweights, 25)
2  q2 = np.percentile(bweights, 50)
3  q3 = np.percentile(bweights, 75)
4  iqr = q3 - q1
5  whisk1 = q1 - 1.5*iqr
6  whisk2 = q3 + 1.5*iqr
7
8  whisk1, q1, q2, q3, whisk2
```

```
(73.5, 108.0, 120.0, 131.0, 165.5)
```

The five numbers above match what we see on the left.

Violin Plots



Violin plots are similar to box plots, but also show smoothed density curves.

- The "width" of our "box" now has meaning!
- The three quartiles and "whiskers" are still present – look closely.

Next up: Box plots and violin plots are useful for comparing multiple distributions.

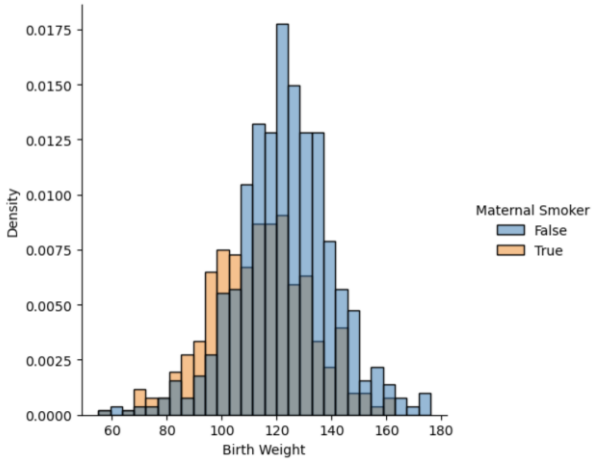
Comparing Quantitative Distributions

Lecture 07

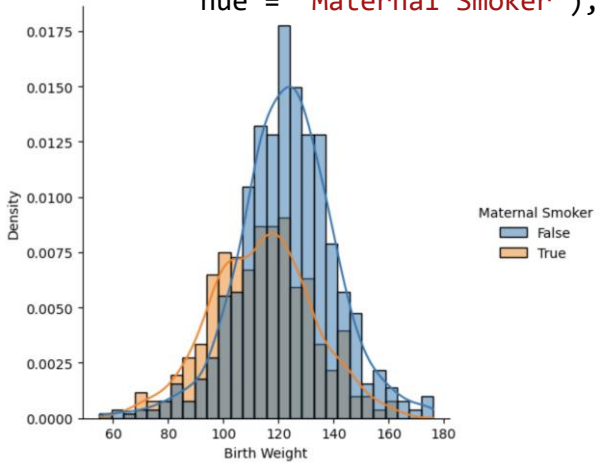
- Visualizations
  - In Data 8 and Data 100
  - In the Real World, Goals
- Distributions
- Bar Plots for Distributions
- Bar Plot Introspection
- Histograms
- Evaluating Histograms
- Box Plots and Violin Plots
- **Comparing Quantitative Distributions**

Overlaid Histograms and Density Curves

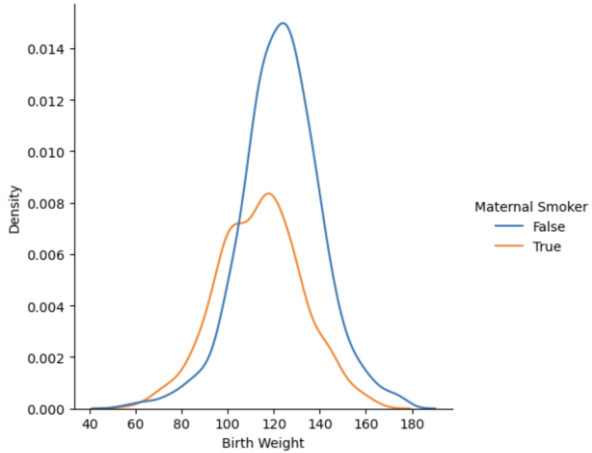
```
sns.displot(data = births,
            x = 'Birth Weight',
            stat = 'density',
            hue = 'Maternal Smoker');
```




```
sns.displot(data = births,
            x = 'Birth Weight',
            kde = True,
            stat = 'density',
            hue = 'Maternal Smoker');
```



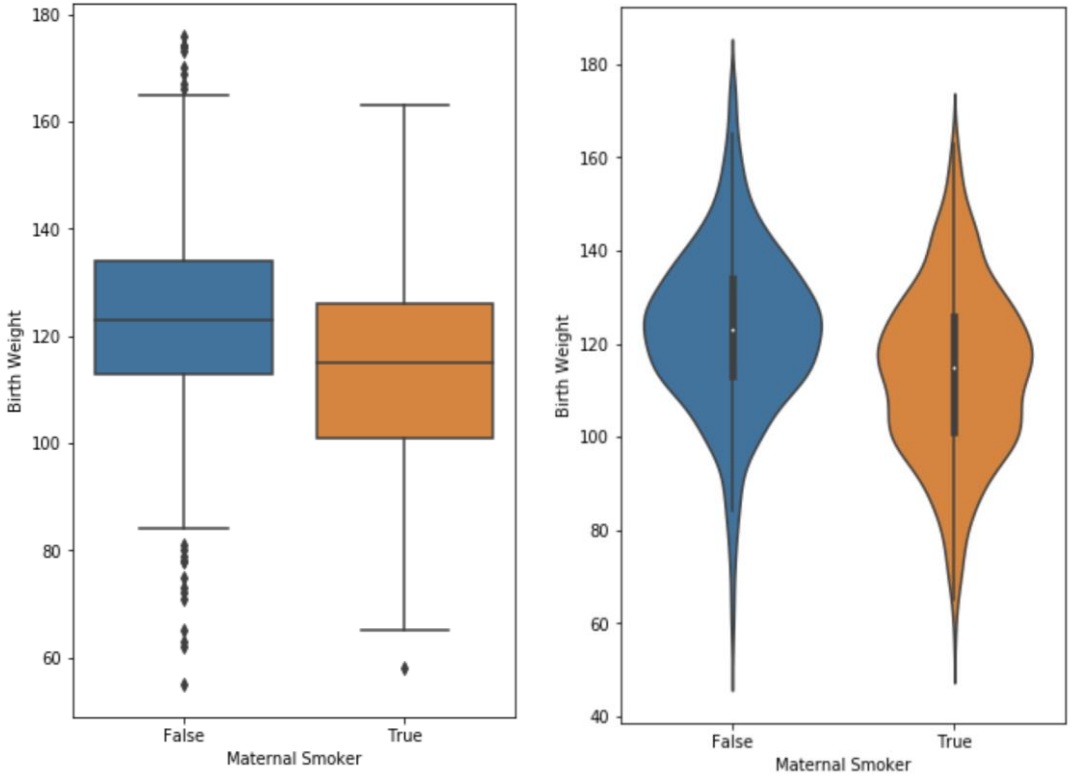

```
sns.displot(data = births,
            x = 'Birth Weight',
            kind = 'kde',
            hue = 'Maternal Smoker');
```



We can overlay multiple histograms and density curves on top of one another.

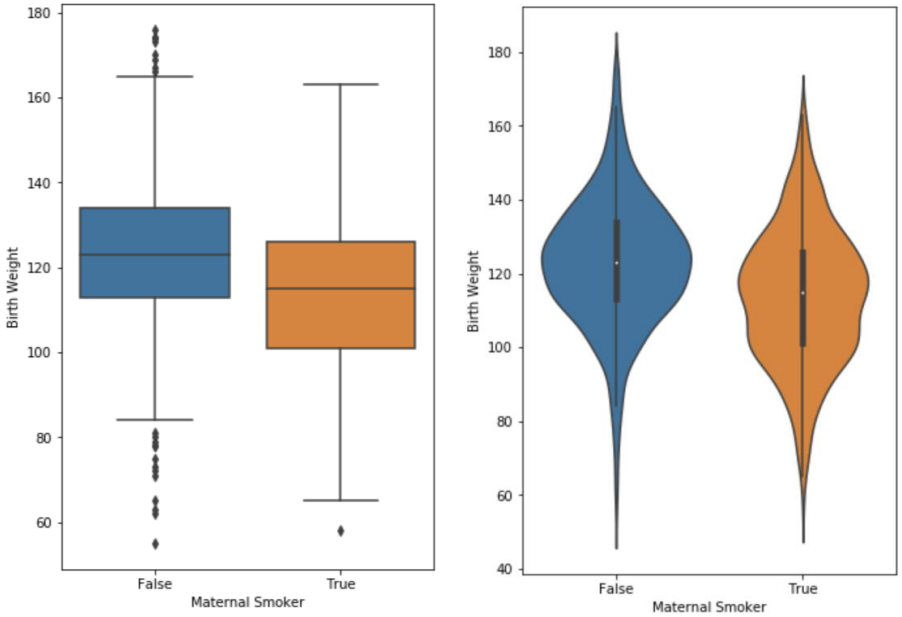
- First: Not terrible, but looks like three separate histograms.
- Second: Has the most information, but isn't very clear!
- Third: Rough estimate of both distributions, but is the most clear by far.
- Neither will generalize well to three or more categories.

Side by Side Box Plots And Violin Plots




```
sns.boxplot(data = births, x = 'Maternal Smoker', y = 'Birth Weight')
sns.violinplot(data = births, x = 'Maternal Smoker', y = 'Birth Weight')
```

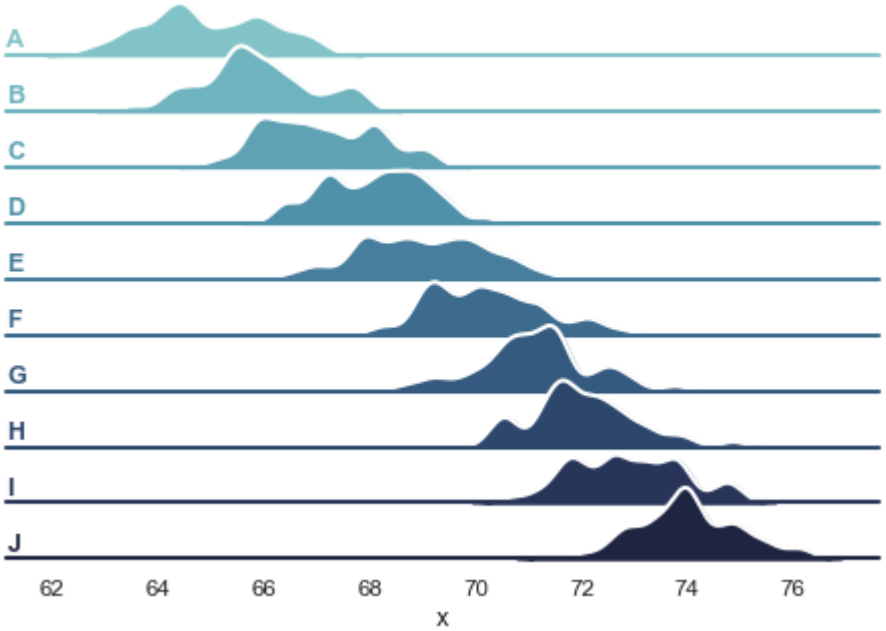
Side by Side Box Plots And Violin Plots



Box plots and violin plots are concise, and thus are well suited to be stacked side by side to compare multiple distributions at once.

- At a glance, we can tell that the median birth weight is higher for babies whose mothers did not smoke while pregnant ("False").
- The violin plot shows us the bimodal nature of the "True" category.

Ridge Plots



Ridge plots show many density curves offset from one another with minimal overlap.

- Useful when the specific *shape* of each curve is important.

Not used in this course, but can be made with seaborn: https://seaborn.pydata.org/examples/kde_ridgeplot.html

LECTURE 7

Visualization, Part I