

LECTURE 7

Visualization

Summary Statistics & Visualizing distributions

CSCI 3022, CU Boulder

Maribeth Oscamou

- HW 3 due tomorrow night (11:59pm MT)
- Graded HW 2 released tonight (with detailed feedback in the Gradescope rubric)
- Quiz 3 at beginning of class on Friday:
 - Scope: HW 2, nb 2, Lectures 2 & 3, Calculus and Discrete concepts that were covered in HW 2
 - What to bring:
 - Pencil
 - (optional) 8.5" x 11" one sided hand written crib sheet
 - (optional) Calculator. If you don't bring a calculator, you will be required to simplify any shorthand notation such as $P(5, 2)$ or $C(4, 2)$ that you use in terms of factorials and quotients.

Goals for this Lecture

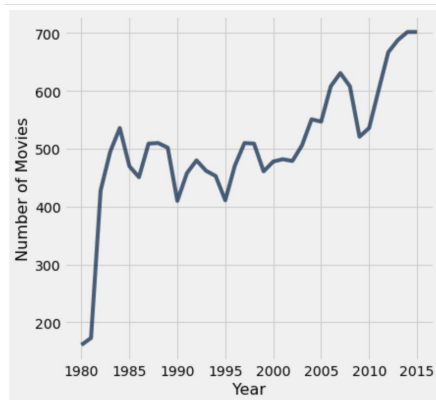
Understand the theories behind effective visualizations and start to generate plots of our own

- The necessary "pre-thinking" before creating a plot
- Python libraries for visualizing data

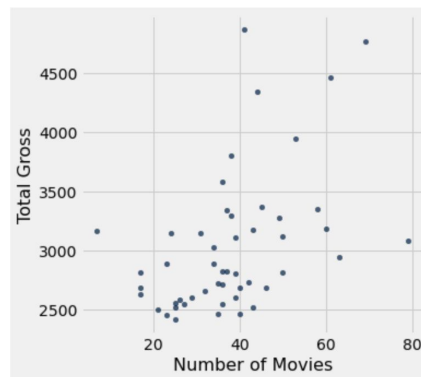
Goals of Visualization

- **Visualization**
 - **Goals of visualization**
 - Distributions
 - Visualizing qualitative data
 - Summarizing quantitative data
 - Visualizing quantitative data
 - Describing Distributions

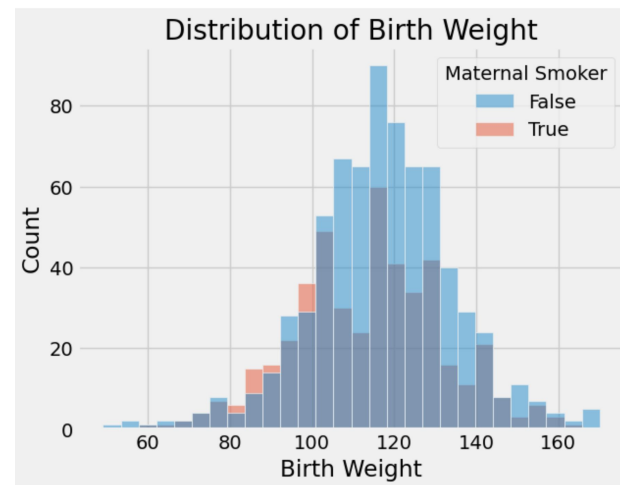
You have likely worked with many types of visualizations before.....



Line plot



Scatter plot



Histogram

What do these achieve?

- Provide a high-level overview of a complex dataset.
- Communicate trends to viewers.

Goals of Data Visualization

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

- Key part of exploratory data analysis.
- Summarize trends visually before in-depth analysis.
- 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.
- Considerations: clarity, accessibility, and necessary context.

What do these goals imply?

Visualizations aren't a matter of making "pretty" pictures.

We need to do a lot of thinking about what stylistic choices communicate ideas most effectively.

Distributions

- **Visualization**
 - Goals of visualization
 - **Distributions**
 - Visualizing qualitative data
 - Summarizing quantitative data
 - Visualizing quantitative data
 - Describing Distributions

A distribution describes...

- The set of values that a variable can possibly take.
- The frequency with which each value occurs.

...for a **single** variable

Example: Distribution of students across recitation sections in CSCI 1300.

- The list of recitation sections (10-11 am, 11-12 pm, etc.)
- The number of students enrolled in each section

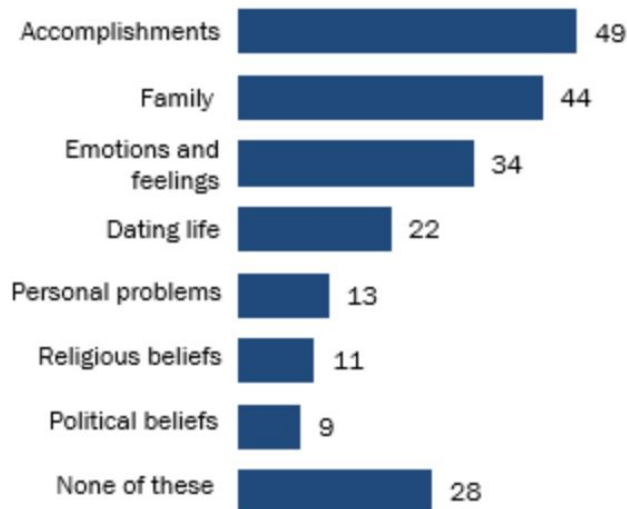
In other words: How is the variable distributed across all of its possible values?

This means that percentages **should sum to 100%** (if using proportions) and counts should **sum to the total number of datapoints** (if using raw counts).

Let's see some examples.

While about half of teens post their accomplishments on social media, few discuss their religious or political beliefs

% of U.S. teens who say they ever post about their ___ on social media



Note: Respondents were allowed to select multiple options.

Respondents who did not give an answer are not shown.

Source: Survey conducted March 7–April 10, 2018.

"Teens' Social Media Habits and Experiences"

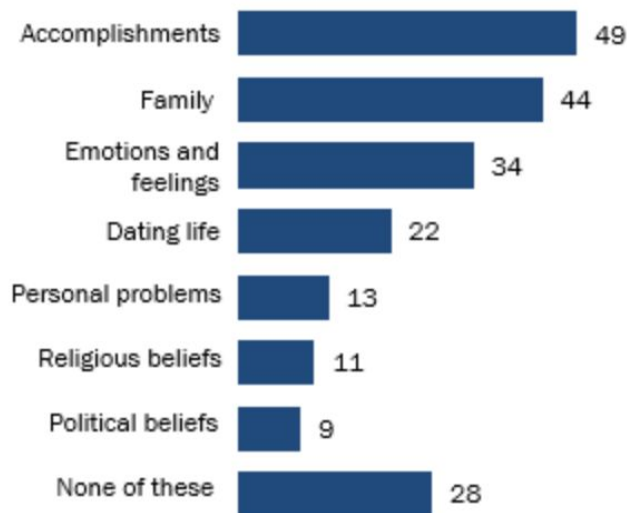
PEW RESEARCH CENTER

Does this chart show a distribution?

- A). yes
- B). no
- C). Need more information

While about half of teens post their accomplishments on social media, few discuss their religious or political beliefs

% of U.S. teens who say they ever post about their ___ on social media



Note: Respondents were allowed to select multiple options.

Respondents who did not give an answer are not shown.

Source: Survey conducted March 7–April 10, 2018.

"Teens' Social Media Habits and Experiences"

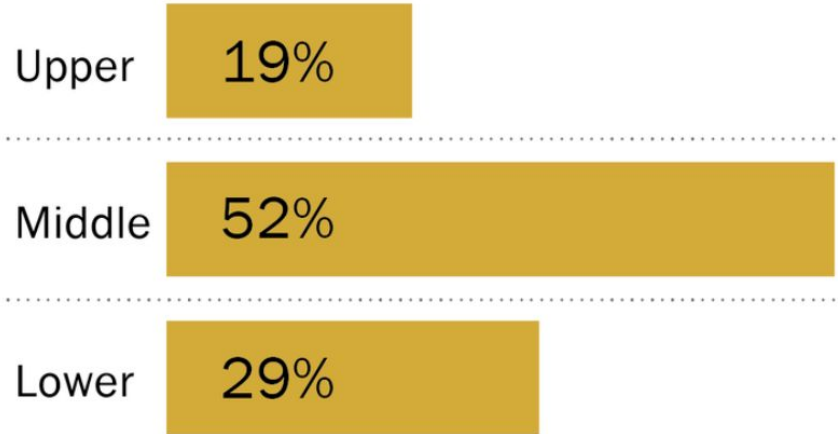
PEW RESEARCH CENTER

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).

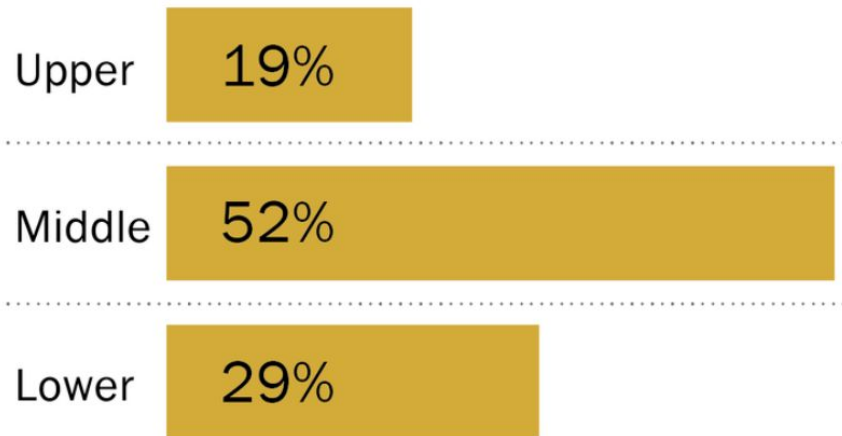
SHARE OF AMERICAN ADULTS
IN EACH INCOME TIER



Does this chart show a distribution?

- A). yes
- B). no
- C). Need more information

SHARE OF AMERICAN ADULTS
IN EACH INCOME TIER



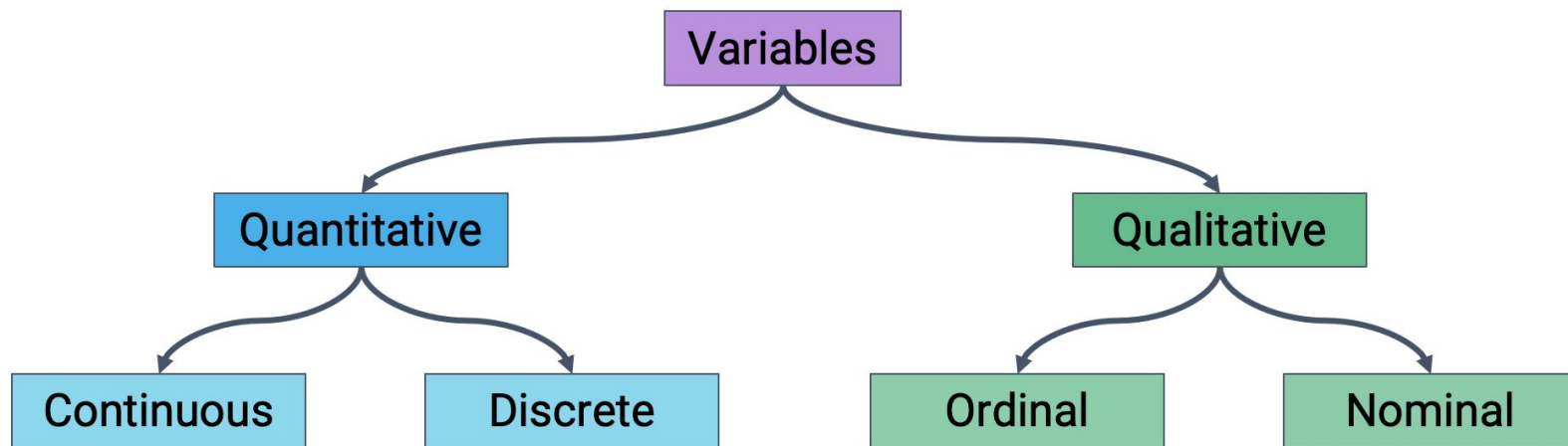
Does this chart show a distribution?

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%.

Variable Types Should Inform Plot Choice

Different plots are more or less suited for displaying particular types of variables.



First step of visualization: Identify the variables being visualized. Then, select a plot type accordingly.

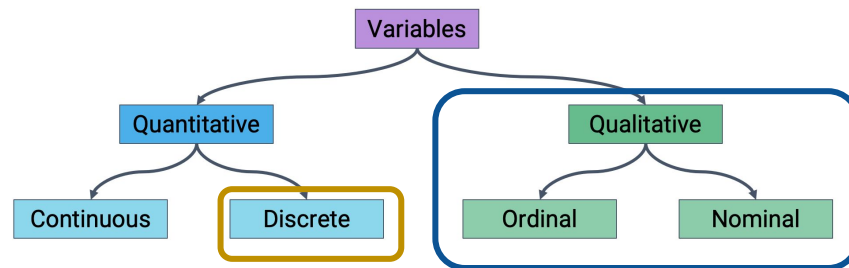
Visualizing Qualitative Data

- **Visualization**
 - Goals of visualization
 - Distributions
 - **Visualizing qualitative data**
 - Summarizing quantitative data
 - Visualizing quantitative data
 - Describing Distributions

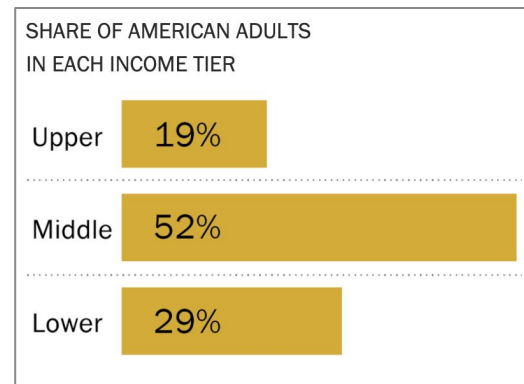
Bar Plots: Distributions of Qualitative Variables

Bar plots are the most common way of displaying the **distribution** of a **qualitative** variable.

*Sometimes quantitative discrete data too, if there are few unique values.



- 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).



In this class we will focus on these 3 plotting libraries:

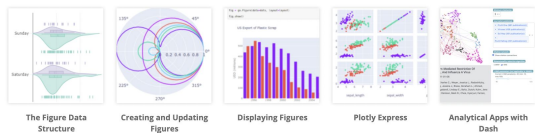


<https://plotly.com/python/>

<https://matplotlib.org/gallery.html>

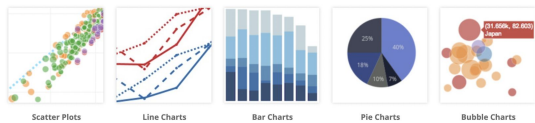
<https://seaborn.pydata.org/examples/index.html>

Fundamentals



More Fundamentals »

Basic Charts

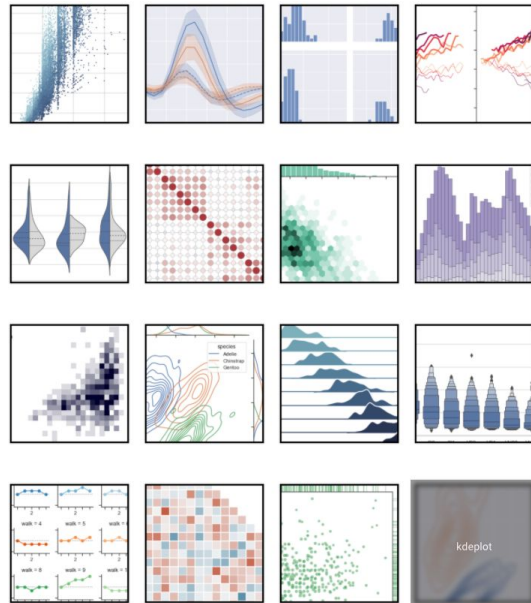
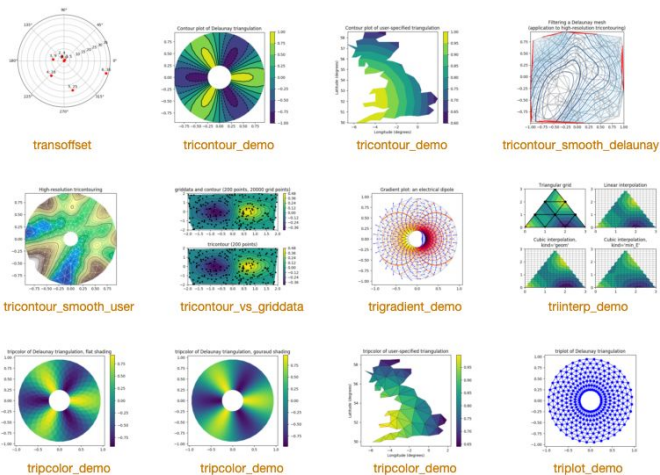


More Basic Charts »

Statistical Charts

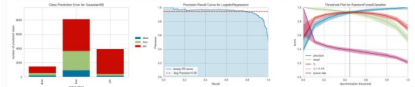


More Statistical Charts »



The rich Python plotting ecosystem - this is not all!

Yellowbrick: Machine Learning Visualization



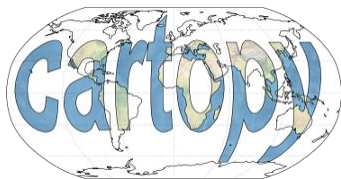
geoplot: geospatial data visualization



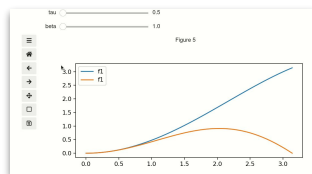
bokkeh



Altair



mpl_interactions



bqplot - Jupyter widgets




See demo notebook

Generating Bar Plots: Matplotlib

Most Matplotlib plotting functions follow the same structure: We pass in a sequence (**list**, **array**, or **Series**) of values to be plotted on the x-axis, and a second sequence of values to be plotted on the y-axis.

```
import matplotlib.pyplot as plt  
plt.plotting_function(x_values, y_values)
```

Matplotlib is typically
given the alias **plt**



To add labels and a title:

```
plt.xlabel("x axis label")  
plt.ylabel("y axis label")  
plt.title("Title of the plot");
```

Generating Bar Plots: Matplotlib

To create a bar plot in Matplotlib: `plt.bar()`

```
continents = wb["Continent"].value_counts()
```

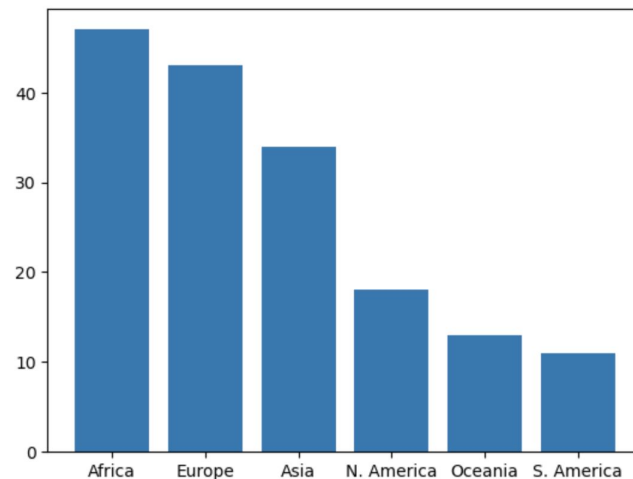
Africa	47
Europe	43
Asia	34
N. America	18
Oceania	13
S. America	11

Name: Continent, dtype: int64

```
plt.bar(continents.index, continents.values);
```

x values

y values




Generating Bar Plots: Seaborn

Seaborn plotting functions use a different structure: Pass in an entire **DataFrame**, then specify what column(s) to plot.

```
import seaborn as sns  
sns.plotting_function(data=df, x="x_col", y="y_col")
```

Seaborn is typically given the alias `sns`

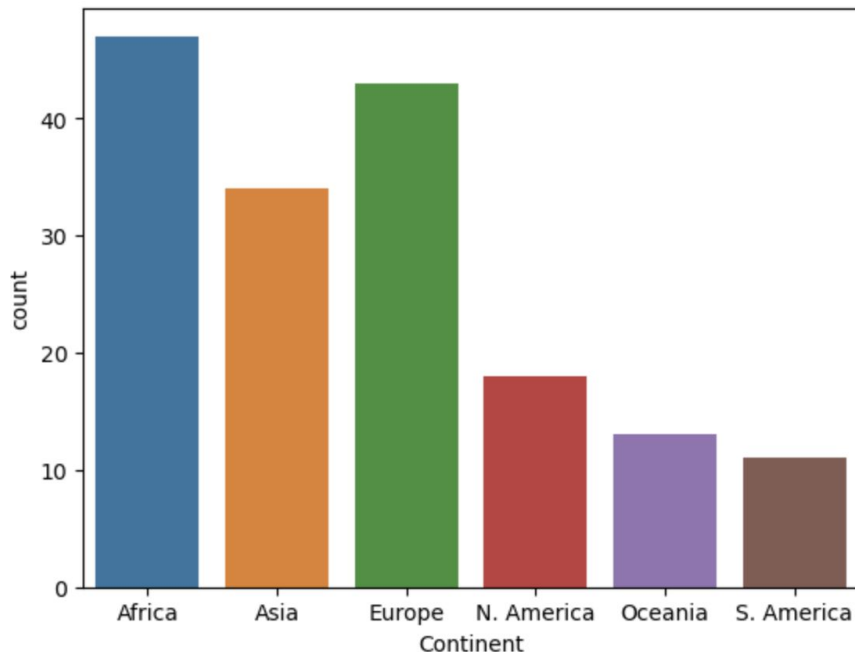


To add labels and a title, use the same syntax as before:

```
plt.xlabel("x axis label")  
plt.ylabel("y axis label")  
plt.title("Title of the plot");
```

Generating Bar Plots: Seaborn

To create a bar plot in Seaborn: `sns.countplot()`



`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
```

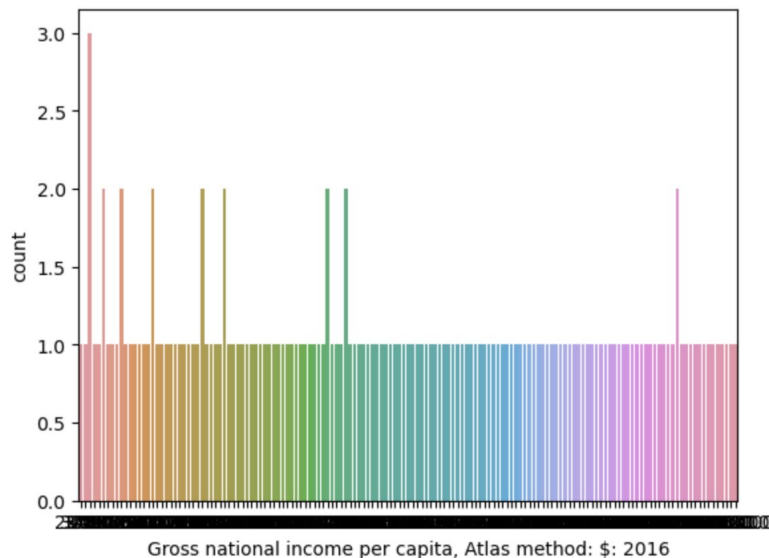
```
sns.countplot(data=wb, x="Continent");
```

Distributions of Quantitative Variables

Earlier, we said that bar plots are appropriate for distributions of qualitative variables.

Why only qualitative? Why not quantitative as well?

- For example: The distribution of gross national income per capita.



A bar plot will create a separate bar for each unique value. This leads to too many bars for continuous data!

Summarizing Quantitative Data

- **Visualization**
 - Goals of visualization
 - Distributions
 - Visualizing qualitative data
 - **Summarizing & Visualizing quantitative data**
 - **Histograms**
 - Describing Distributions

Summarizing the “center” of the sample is a popular way to characterize quantitative data.

Mean:
$$\bar{x} = \frac{x_1 + x_2 + \dots + x_n}{n} = \frac{1}{n} \sum_{k=1}^n x_k$$

Median: The *middle element* of the dataset when it is put in ascending order.

Note: When n is even, take the average of the middle two elements.

Mode: Most commonly occurring value in a data set.

Characterizing Quantitative Data: Quartiles

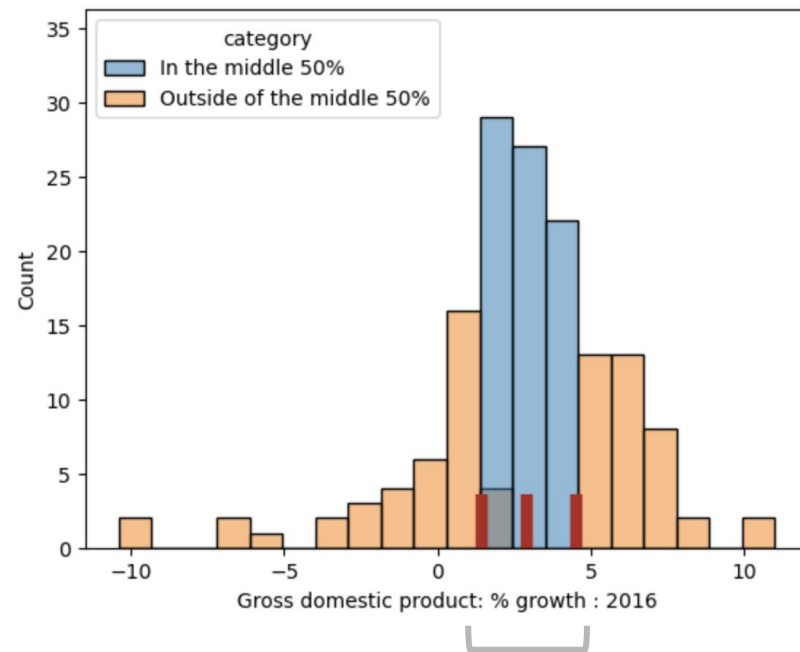
For a quantitative variable:

- 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.

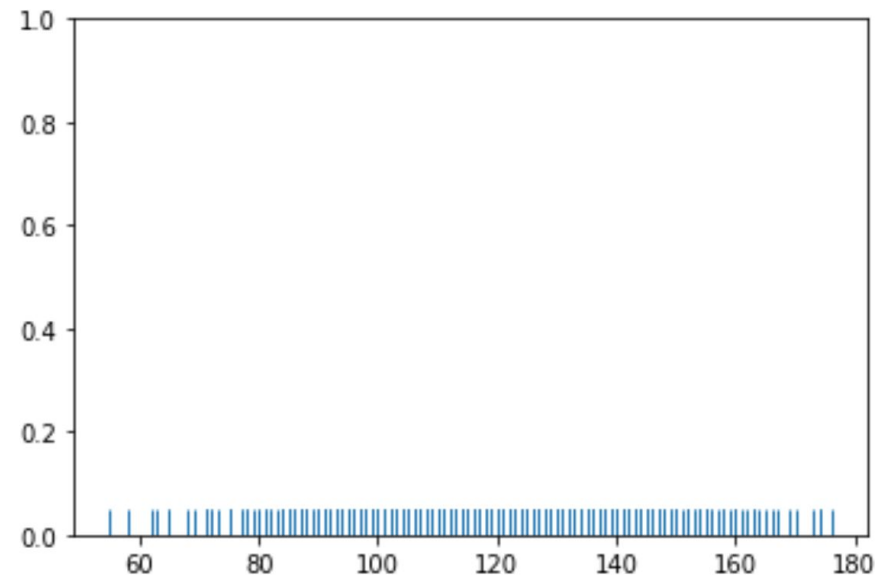
- $\text{IQR} = \text{third quartile} - \text{first quartile}$.



The length of this region is the IQR

Visualizing Quantitative Data: Rug plot

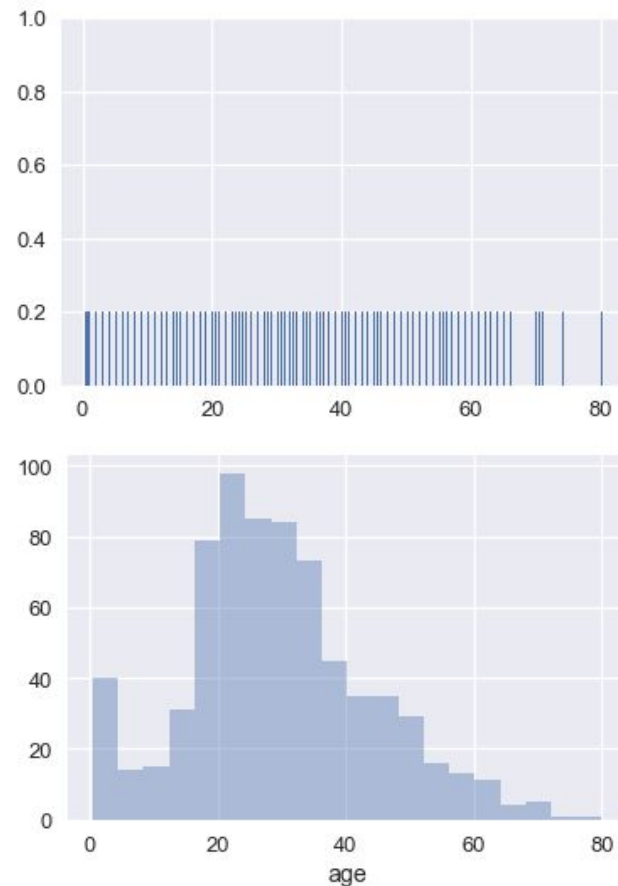
- Rug plots are used to show the distribution of a single quantitative (**numerical**) variable.
- They show us each and every value!
- Issues with rug plots:
 - Too much detail.
 - Hard to see the bigger picture.
 - **Overplotting.**
 - How many birth weights were at 120?
 - Can't tell – they're all on top of each other.



`sns.rugplot(bweights)`

- HW 4 released tonight
- Nb4 (EDA) released Monday; Live Zoom session Tuesday 5pm-6pm

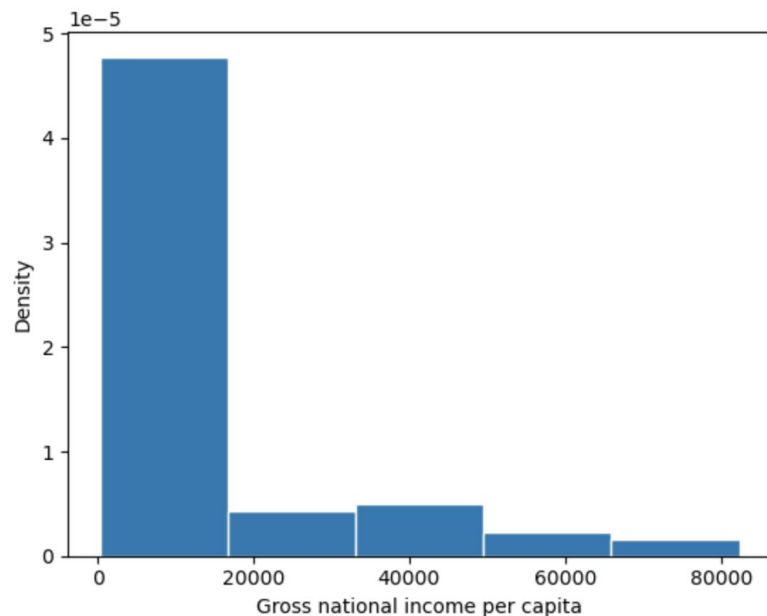
- Histograms are a smoothed version of rug plots.
- We smooth if we want to focus on general structure rather than individual observations.



Histograms

A histogram:

- Collects datapoints with similar values into a shared "bin".
- Scales the bins such that the **area** of each bin is equal to the **percentage** of datapoints it contains



The first bin has a width of \$16410
height of 4.77×10^{-5}

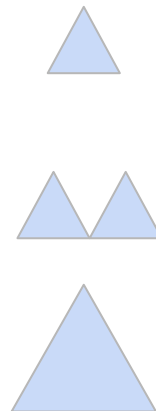
This means that it contains $16410 \times (4.77 \times 10^{-5}) = 78.3\%$
of all datapoints in the dataset.

Area Principle for Histograms

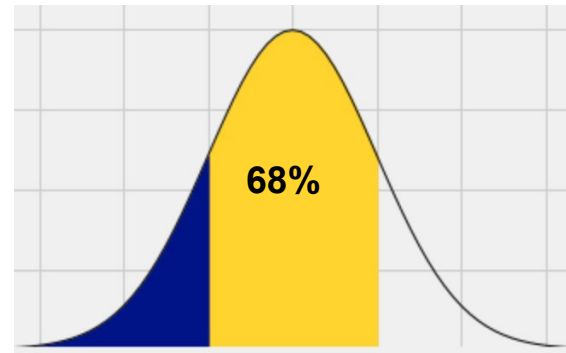
Areas should be proportional to the values they represent.

For example

- If you represent 20% of a population by
- Then 40% can be represented by:
- But not by:



- Displays the distribution of a numerical variable
- One bar corresponding to each bin
- Uses the area principle:
 - The **area** of each bar is the **percent** of individuals in the corresponding bin
- The vertical axis is a rate (e.g., percent per year)
- We can also approximate histograms by smooth curves.
- Areas will still represent percents.



How to Calculate Height in a Density Histogram

Example: Let's say you randomly sample 5 music students and ask them how many years they've played an instrument. Suppose we get the following data points:

Points: [2.2, 2.8, 3.7, 5.3, 5.7]

Suppose we choose the following bins:

Bins: [0, 2), [2, 4), [4, 6), [6, 8]

The [2, 4) bin contains 3 out of the 5 total points:

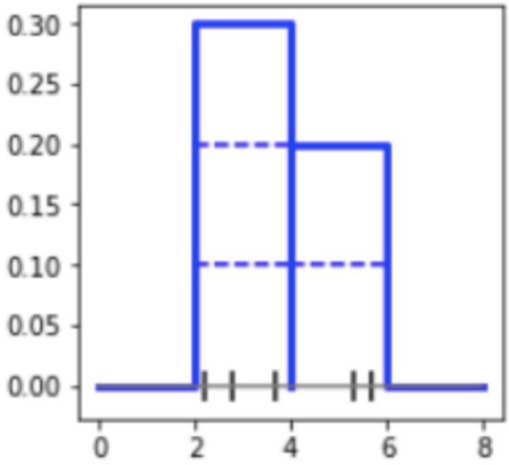
- 3 out of 5 is 60%
- The bin is $4 - 2 = 2$ years wide

60 percent

Height of bar above [2,4) bin = -----

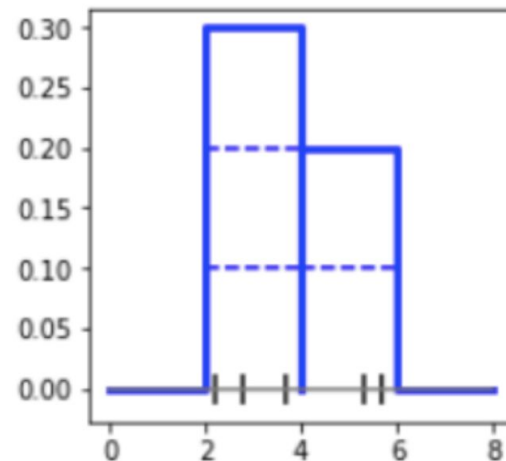
2 years

= 30 percent per year



Height Measures Density

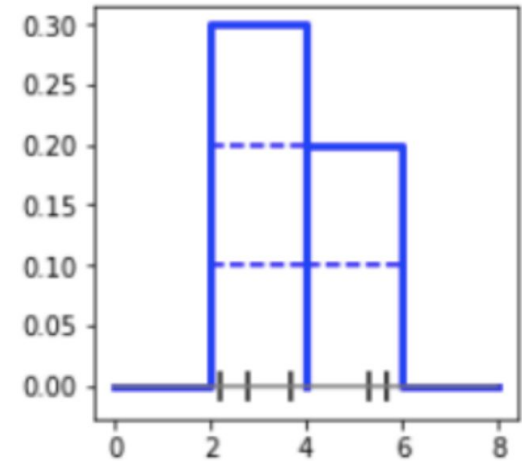
$$\text{Height} = \frac{\text{\% in bin}}{\text{width of bin}}$$



- The height measures the percent of data in the bin ***relative to the amount of space in the bin.***
- Height measures crowdedness, or density.
- Units: percent per unit on the horizontal axis

Area Measures Percent

Area of bar = % in bin = Height x width of bin



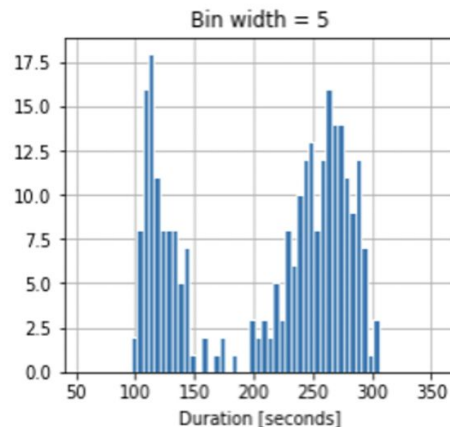
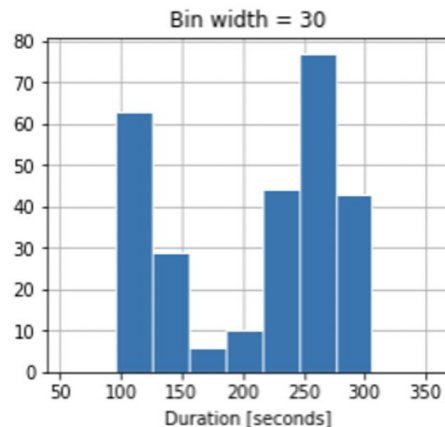
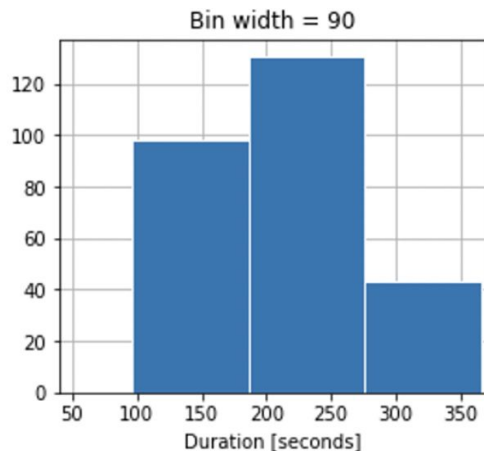
- “How many individuals in the bin?” Use area.
- “How crowded is the bin (i.e. what is its density)?” Use **height**.

Histograms in Code

Bin Size

Caution: The shape of your histogram can be easily influenced by your bin size and the starting value of your bins.

Example: The following are 3 different histograms of the SAME Old Faithful eruption duration data. These 3 different bin choices tell 3 very different stories about the data:

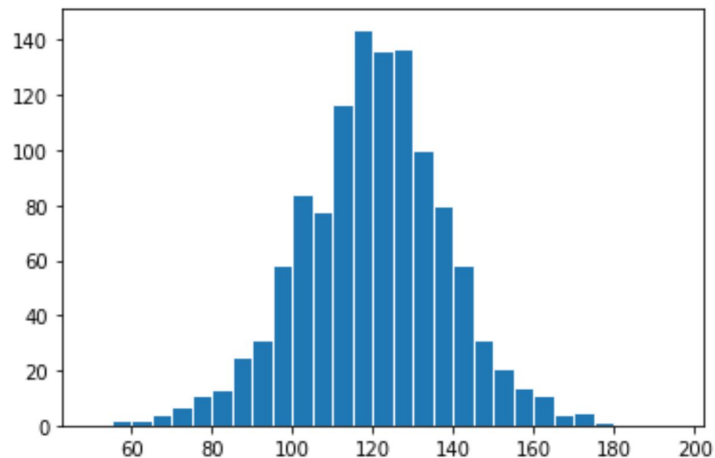


There are a variety of rules of thumb for how to pick the best interval width. (See here for some of them: https://onlinestatbook.com/2/graphing_distributions/histograms.html)

The best advice is to experiment with different choices of width, and to choose a histogram according to how well it communicates 36

Histograms in Code

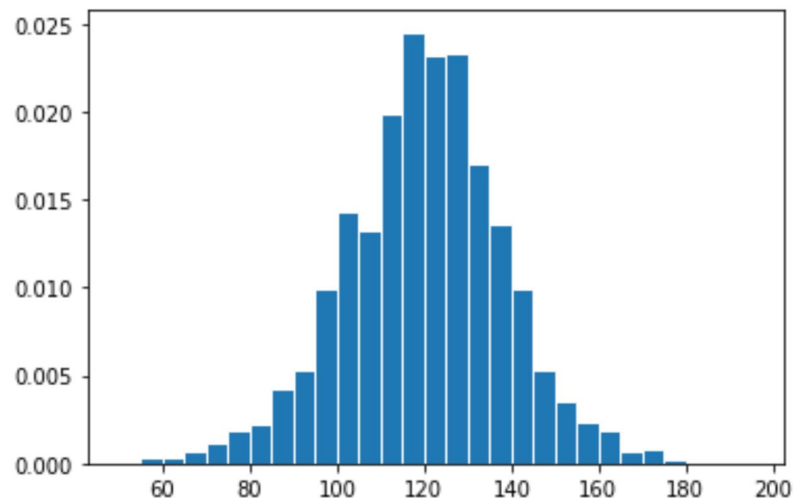
By default, **matplotlib** histograms show *counts* on the y-axis, *not* *proportions* per unit.



```
plt.hist(bweights, bins=bw_bins, ec='w')
```

where `bw_bins = range(50, 200, 5)`

We use the optional **density** parameter to fix the y-axis. After doing this, the total area sums to 1.



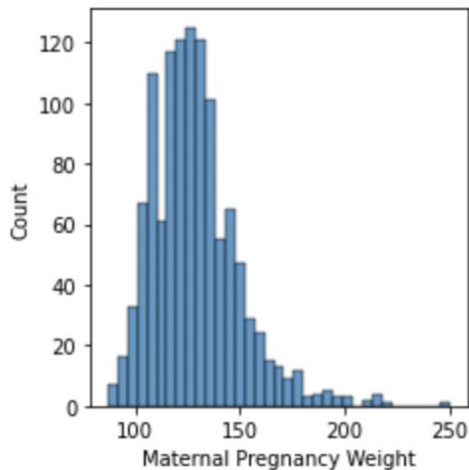
```
plt.hist(bweights, density=True,  
bins=bw_bins, ec='w')
```

Density Histograms in Code

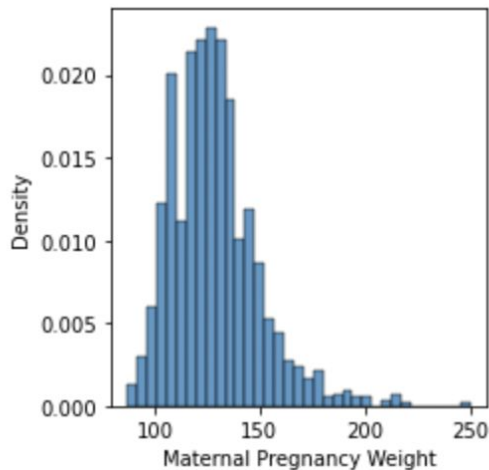
In Matplotlib: `plt.hist(x_values, density=True,)`

In Plotly: `px.histogram(births, x = 'Maternal Pregnancy Weight', histnorm="probability density", nbins=20)`

In Seaborn: `sns.histplot(data=births, x="Maternal Pregnancy Weight", stat="density",)`

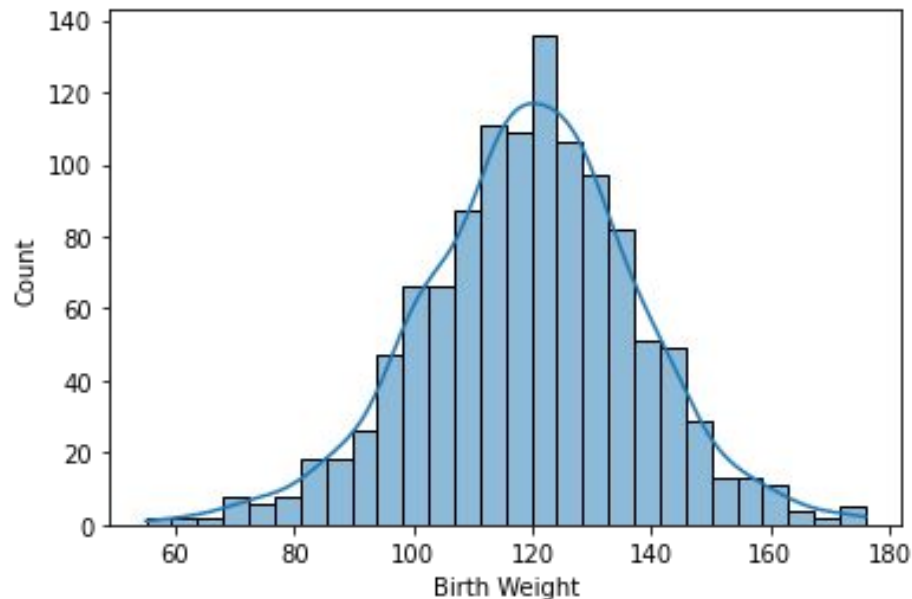


Frequency
Histogram



Density Histogram

Density curves



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**.

- Density curves are a smoothed versions of histograms.

```
sns.histplot(bweights, kde=True)
```

Kernel density estimation (KDE)

Kernel Density Estimation is used to estimate a **density curve** (aka a probability density function) from a set of data.

- Just like a histogram, a density function's total area must sum to 1.

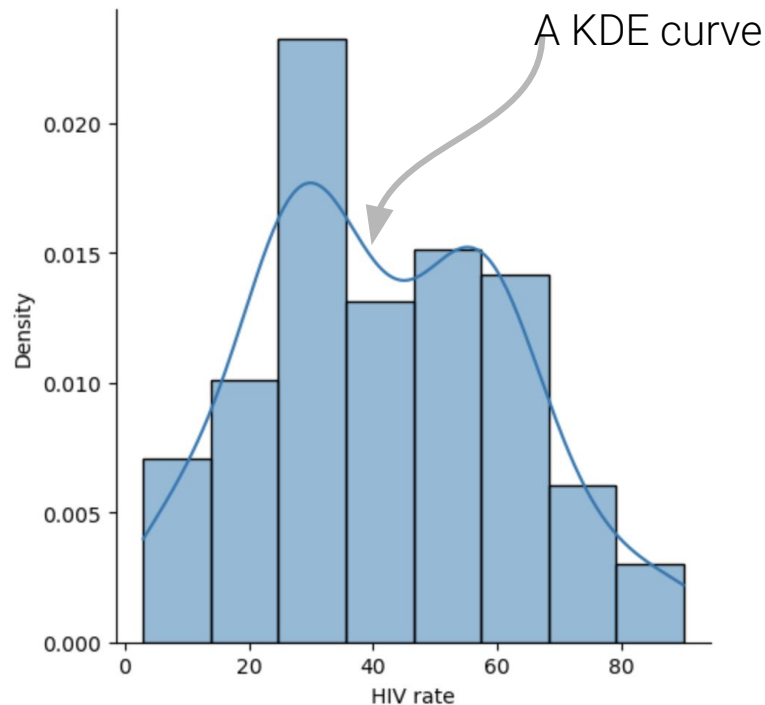
There's a built-in function in Python that will automatically overlay a KDE on top of a histogram:

```
sns.histplot(bweights, kde=True)
```

We can also plot a density curve by itself, by appropriately setting the parameters of

sns.displot or calling directly `sns.kdeplot`:

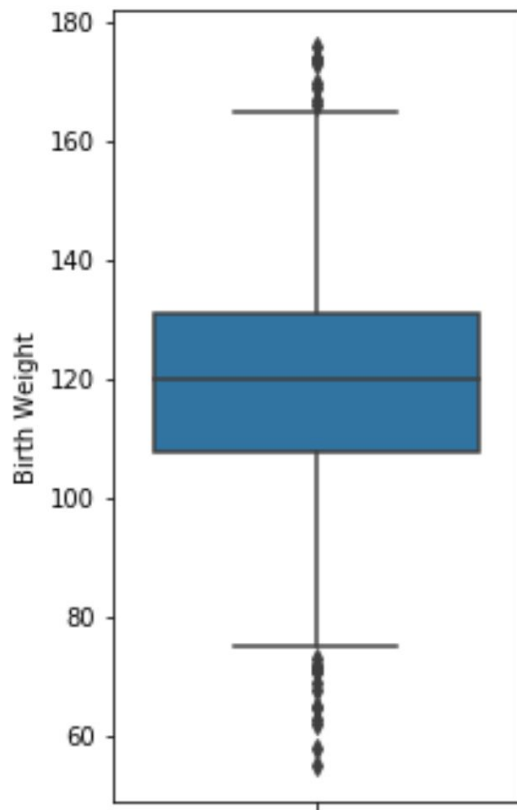
```
sns.displot(bweights, kind='kde')  
sns.kdeplot(bweights)
```



Visualizing Quantitative Data

- **Visualization**
 - Goals of visualization
 - Distributions
 - Visualizing qualitative data
 - **Summarizing & Visualizing quantitative data**
 - Histograms
 - **Box Plots and Violin Plots**
 - Describing Distributions

Box plots

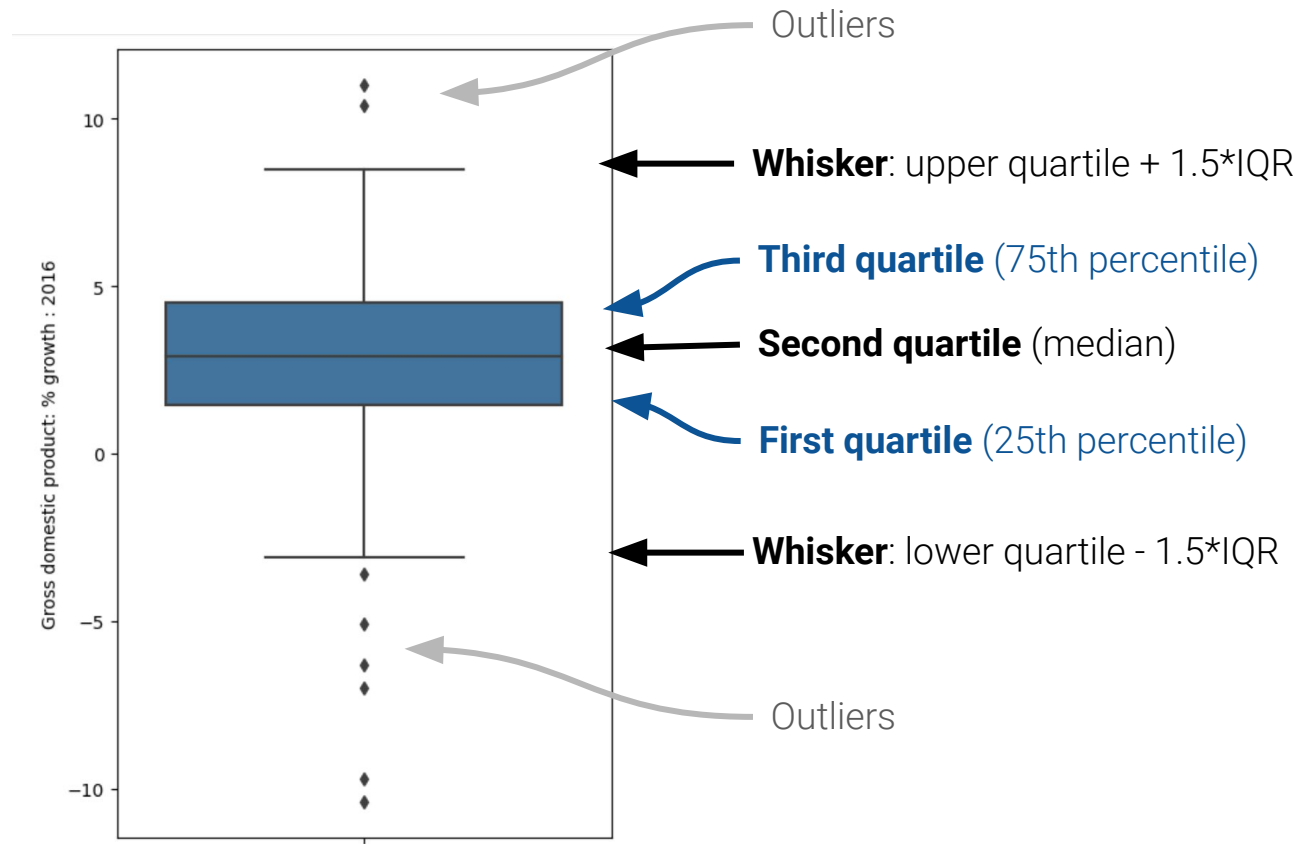


Box plots summarize several characteristics of a numerical distribution. They visualize:

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

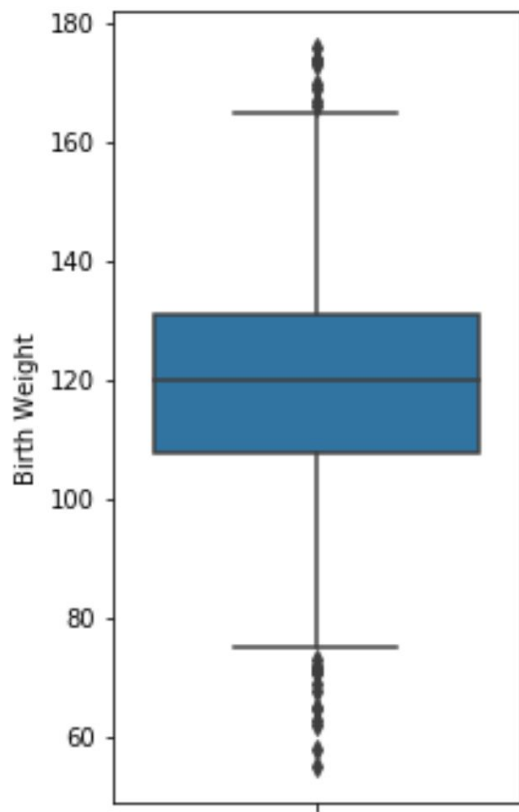
`sns.boxplot(bweights)`

Box Plots



```
sns.boxplot(data=wb, y="Gross domestic product: % growth : 2016")
```

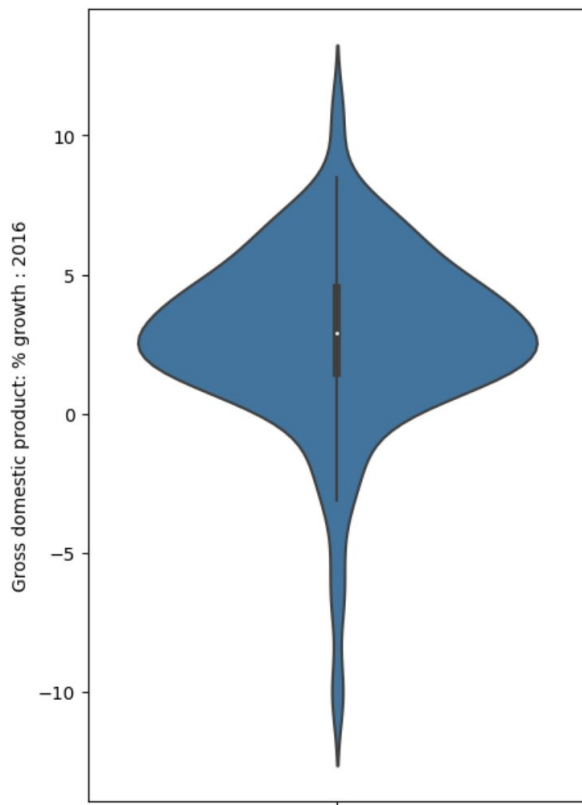
Box plots



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

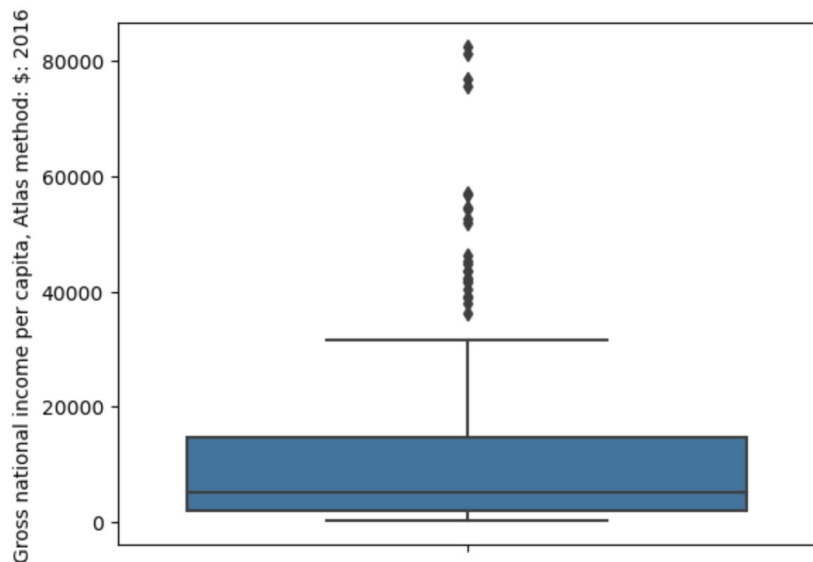
- The "width" of our "box" now has meaning!
- The three quartiles and "whiskers" are still present – look closely.

```
sns.violinplot(data=wb, y="Gross domestic product: % growth : 2016")
```

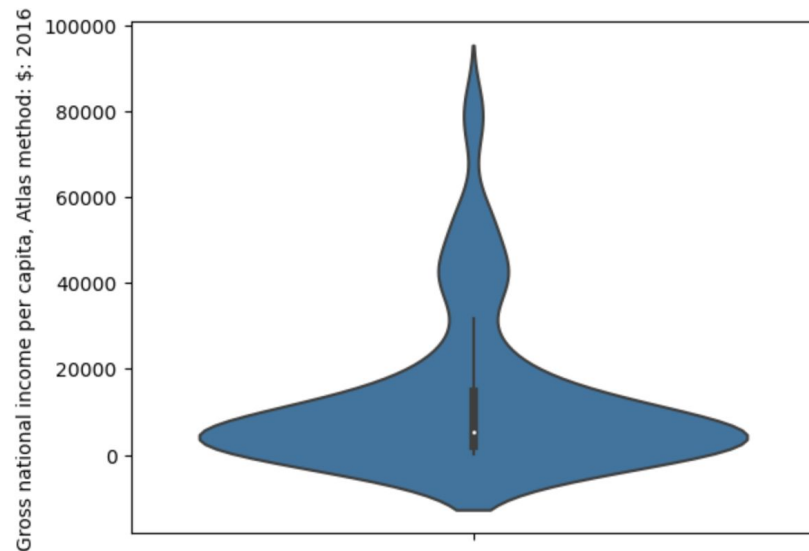
Box plots and Violin Plots

Box plots and violin plots display distributions using information about **quartiles**.

- In a box plot, the width of the box encodes no meaning.
- In a violin plot, the width of the "violin" indicates the density of datapoints at each value.



```
sns.boxplot(data=df, y="y_variable");
```



```
sns.violinplot(data=df, y="y_variable");
```

Describing Distributions

- **Visualization**
 - Goals of visualization
 - Distributions
 - Visualizing qualitative data
 - **Summarizing & Visualizing quantitative data**
 - Histograms
 - Box Plots and Violin Plots
- **Describing Distributions**

Describing distributions

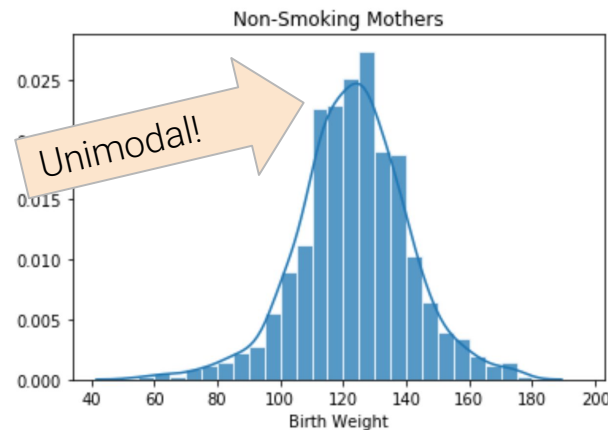
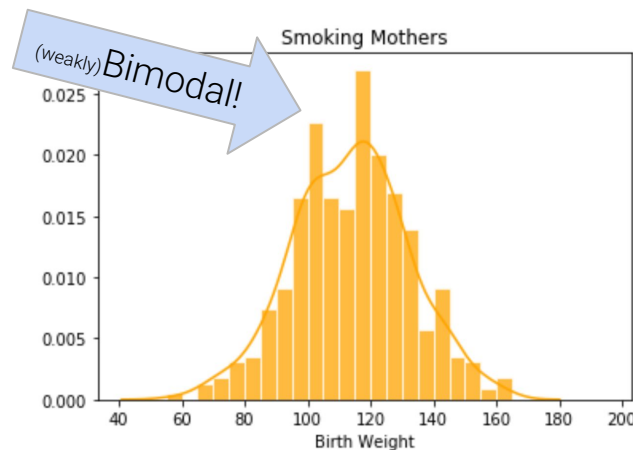
One of the benefits of a histogram or density curve is that they show us the “bigger picture” of our distribution (something we don’t get with a rug plot).

Some of the terminology we use to describe distributions:

- **Modes.**
- **Skewness.**
 - Skewed left vs skewed right.
- **Tails.**
 - Left tail vs right tail.
- **Outliers.**
 - Define these arbitrarily.
 - Will see one definition in the next section.

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.

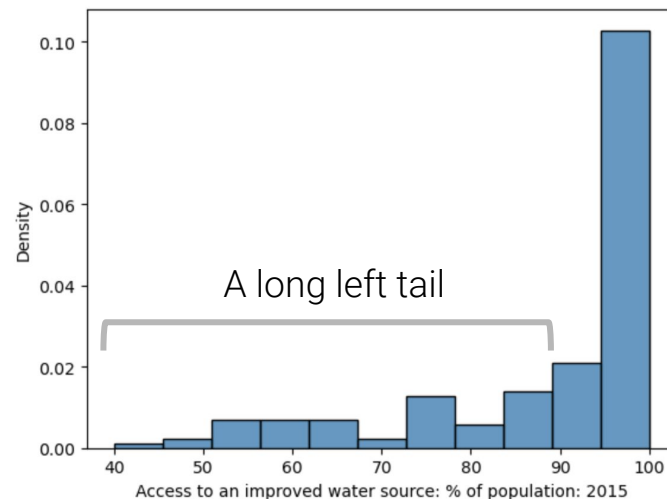
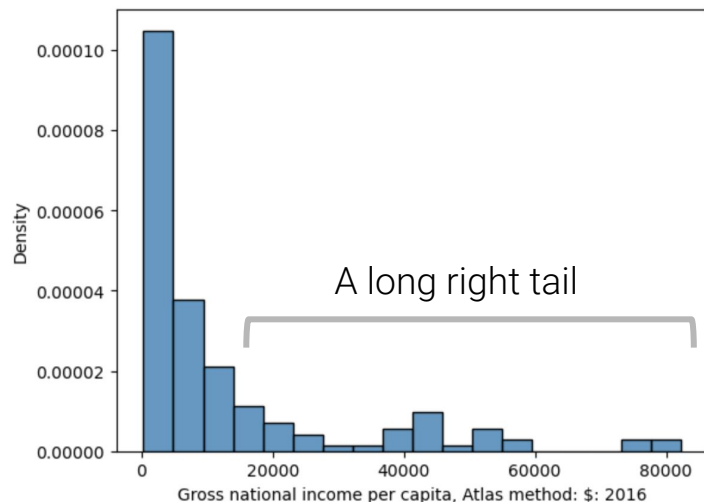


Interpreting Histograms

The **skew** of a histogram describes the direction in which its "tail" extends.

- A distribution with a long right tail is skewed right.
- A distribution with a long left tail is skewed left.

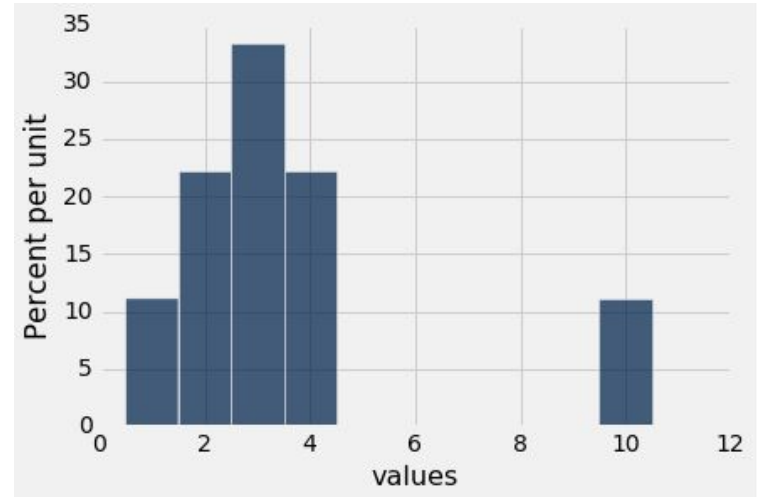
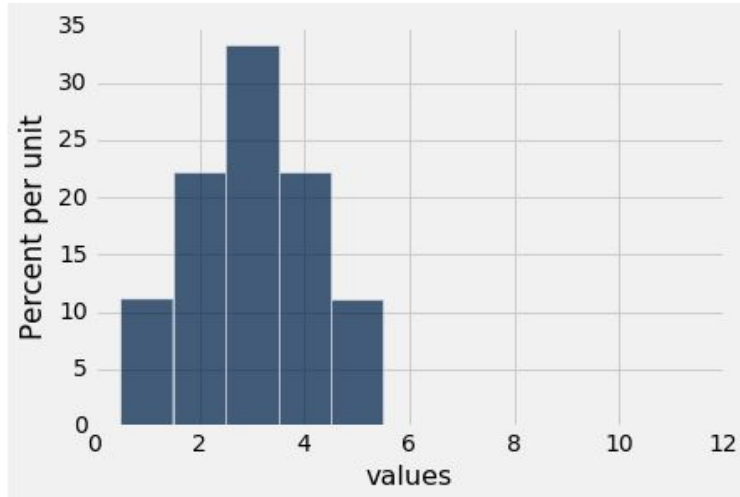
A histogram with no clear skew is called symmetric.



Discussion Question: Mean vs Median

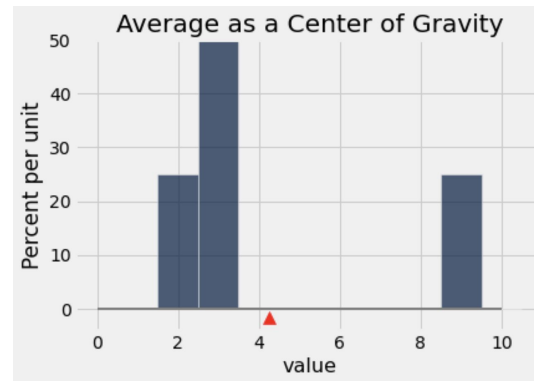
Are the medians of these two distributions the same or different? Are the means the same or different?
If you say “different,” then say which one is bigger.

- A). Means Equal B). Medians Equal C). Left mean > Right mean d). Left mean < right mean
e). Median on left > median on right

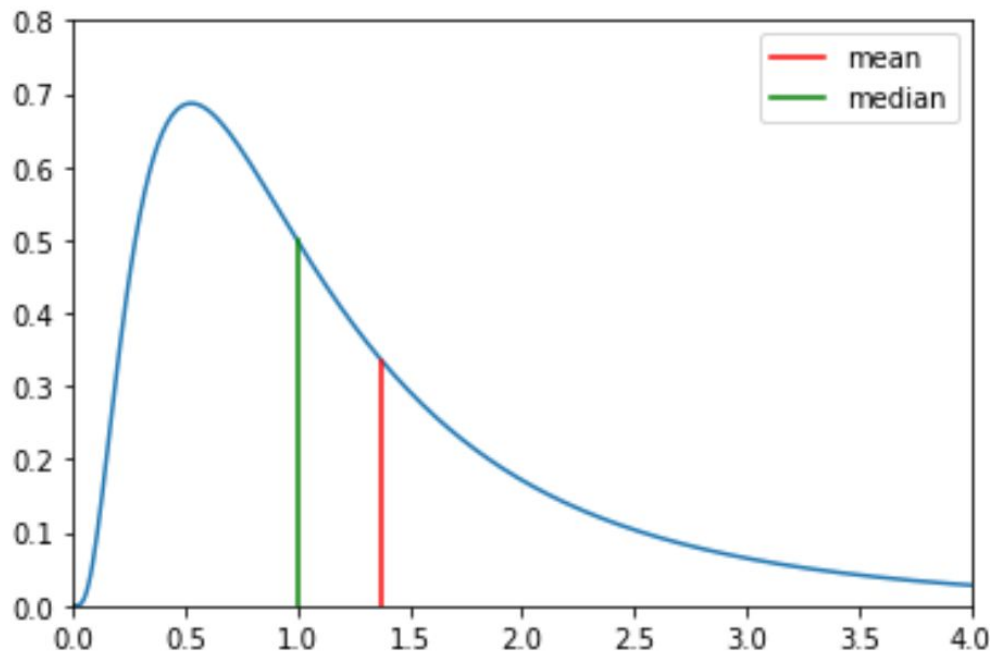


Comparing Mean and Median

- **Mean:** Balance point of the histogram
 - Physics Analogy: Center of Gravity
- **Median:** 50th percentile of the data
- If the distribution is symmetric about a value, then that value is both the average and the median
- If the histogram is skewed, then the mean is pulled away from the median in the direction of the skew (tail)



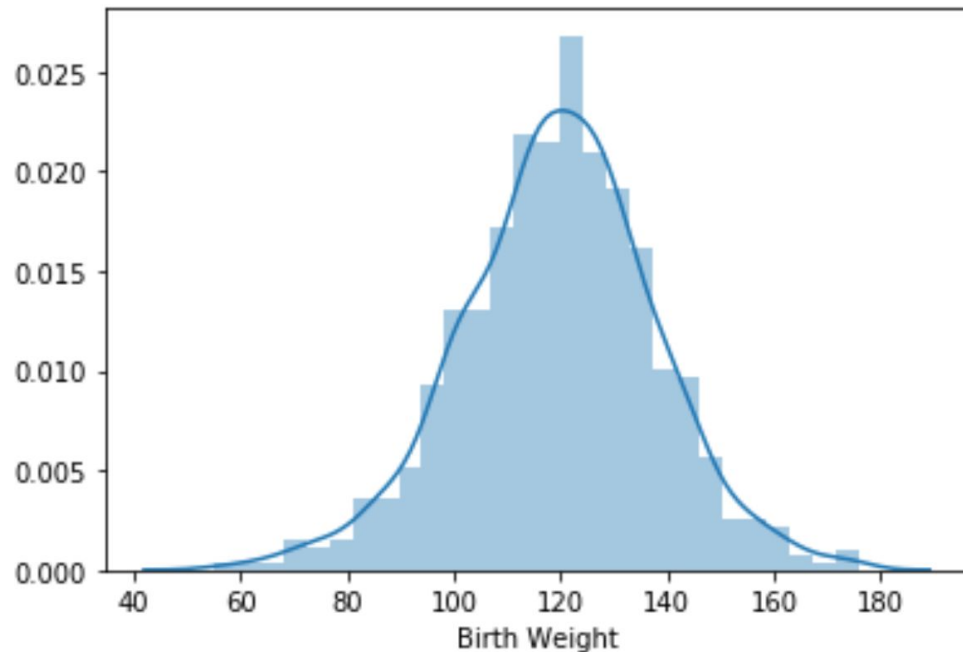
Skew and tails



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

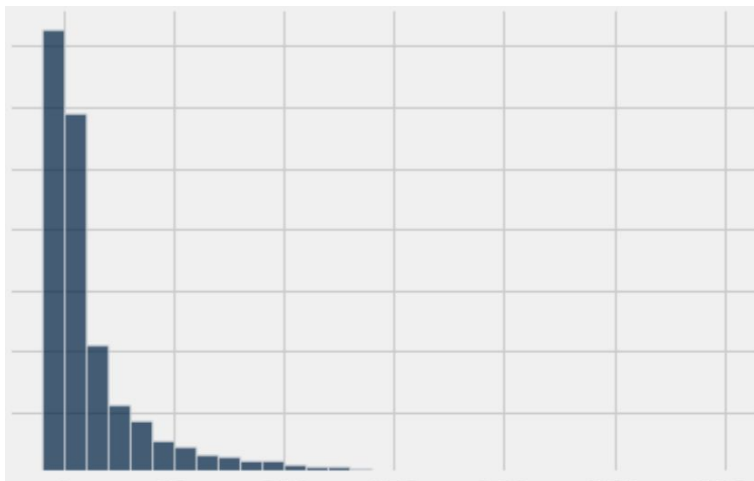
- Such an example is on the left.
- In such cases, the mean is typically to the right of the median.
 - Think of the mean as the “balancing point” of the density.
- In the event that the tail is on the left, we say the data is skewed left.
- Our distribution can be symmetric, when both tails are of equal size.

Describing Distributions: Example



Consider the distribution of birth weights shown to the left. We might describe this as being:

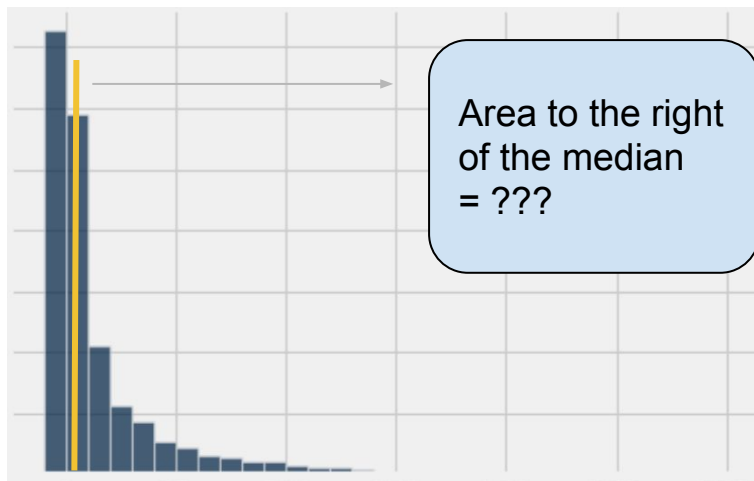
- Unimodal. There is a single clear peak.
- Symmetric. It doesn't appear to be skewed in any direction.
 - Mean is very close to the median.
- Roughly bell shaped.



The histogram shows the distribution of values contained in an array **x**.

Which of the following is **True**?

- (A) `sum(x > np.average(x)) / len(x) < 0.5`
- (B) `sum(x > np.average(x)) / len(x) == 0.5`
- (C) `sum(x > np.average(x)) / len(x) > 0.5`



The histogram shows the distribution of values contained in an array **x**.

The gold line is at the median.

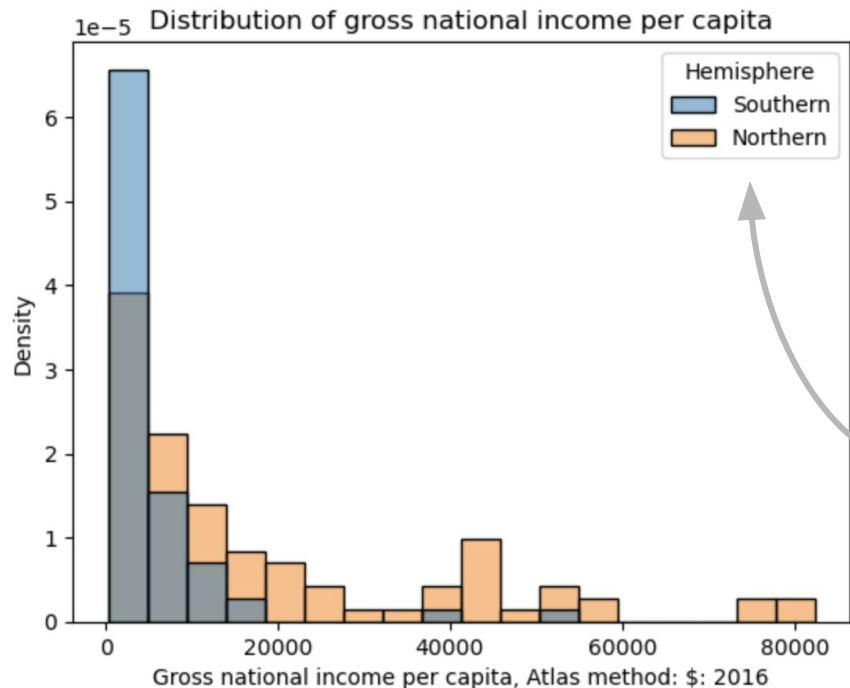
Which of the following is **True**?

- (A) `sum(x > np.average(x)) / len(x) < 0.5`
- (B) `sum(x > np.average(x)) / len(x) == 0.5`
- (C) `sum(x > np.average(x)) / len(x) > 0.5`

Comparing quantitative distributions

Overlaid Histograms

To compare a quantitative variable's distribution across qualitative categories, overlay histograms on top of one another.

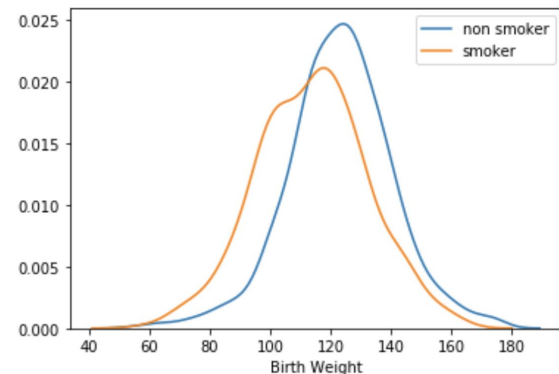
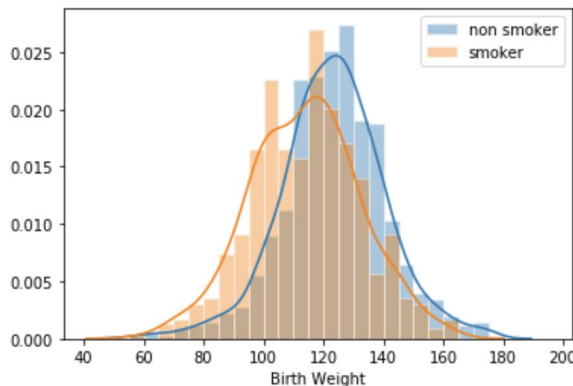
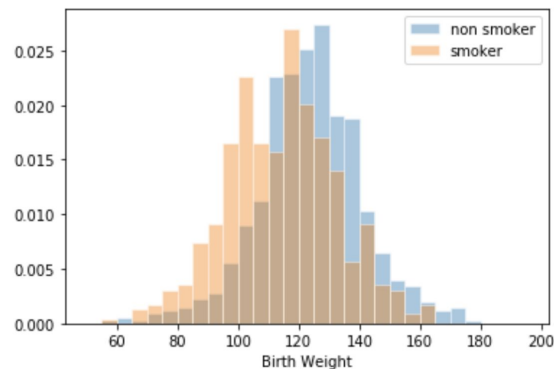


The **hue** parameter of Seaborn plotting functions sets the column that should be used to determine color.

```
sns.histplot(data=wb, hue="Hemisphere",  
x="Gross national income...")
```

Always include a legend when color is used to encode information!

Overlaid histograms and density curves



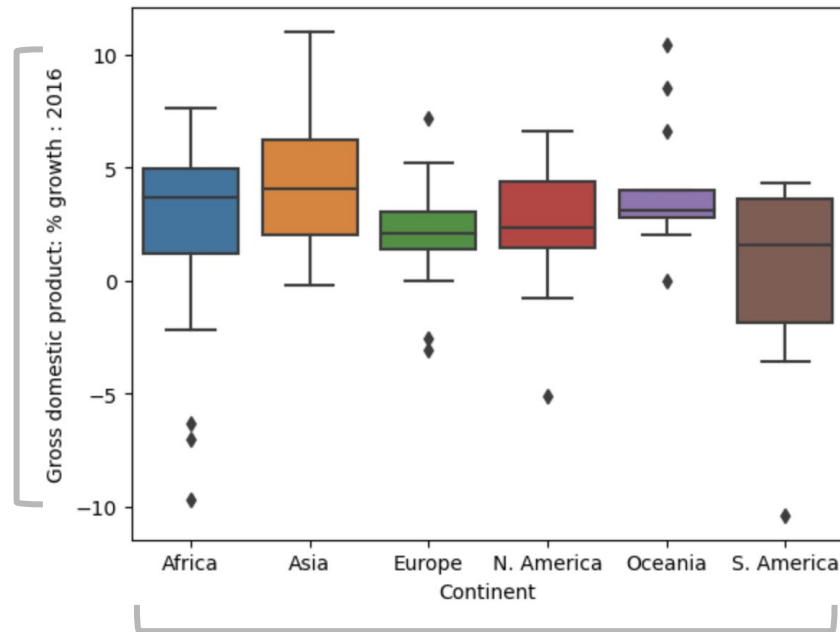
We can overlay multiple histograms and density curves on top of one another.

- 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 and Violin Plots

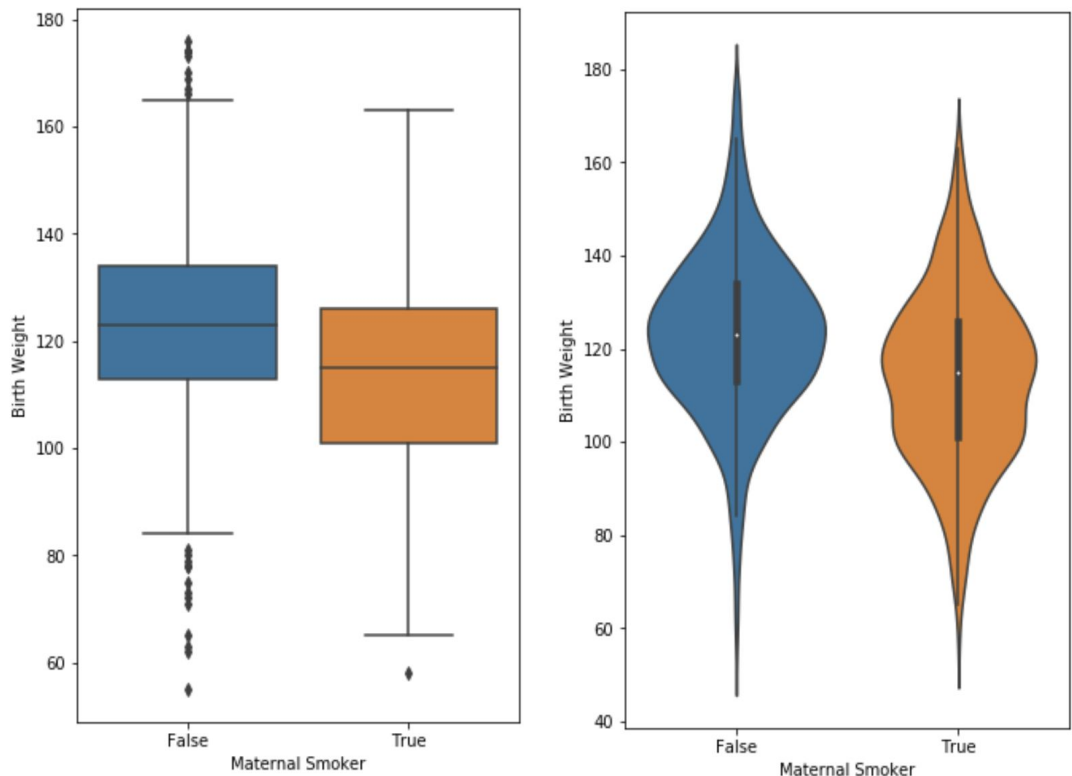
What if we wanted to incorporate a *qualitative* variable as well? For example, compare the distribution of a quantitative continuous variable *across* different qualitative categories.

```
sns.boxplot(data=wb, x="Continent", y="Gross domestic product: % growth : 2016");
```



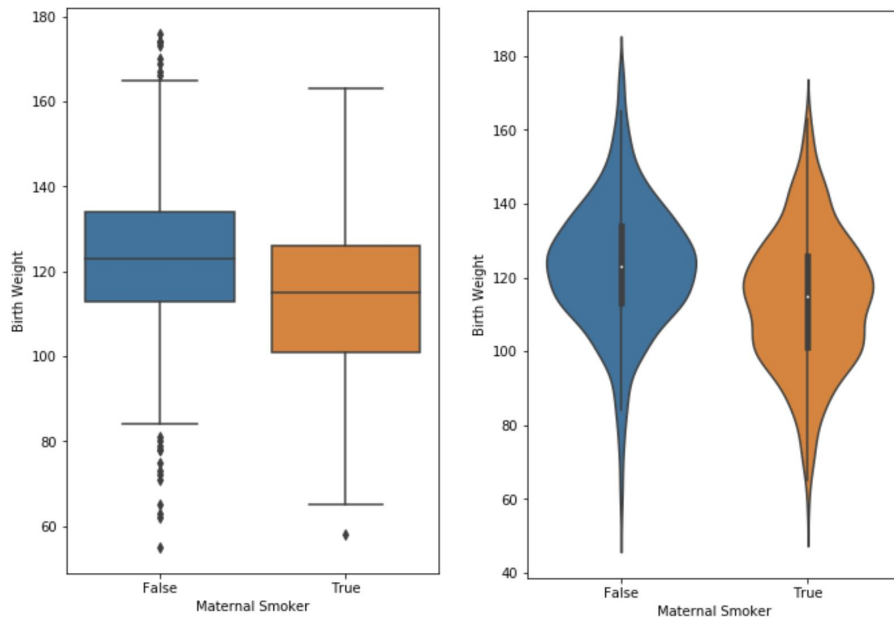
Continent: qualitative nominal

Side by side box plots and violin plots



Code is in the speaker's notes.

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.

Mean as Minimum of Mean Squared Error

Median as Minimum of Mean Absolute Error

LECTURE 7

Visualization

Content credit: [Acknowledgments](#)