Data science for everyone

Prof. Jones-Rooy & Prof. Policastro

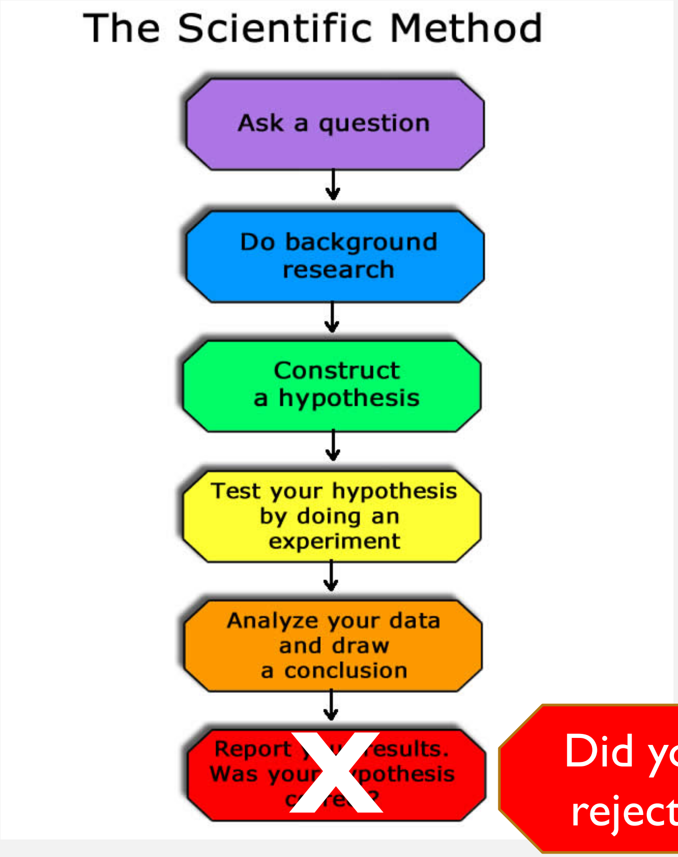
Feb. 26, 2020

5.2: Organizing data in Python

ANNOUNCEMENTS

1. Lab 2 is out
   - Due today, Wed., Feb. 26, 8p
2. Lab 3 out today, Wed., Feb. 26, 8p
   - Due Wed., March 4, 8p
3. Homework 2 is out
   - Due Mon., March 9, 8p

STORY TIME!



The Scientific Method

Ask a question
Do background research
Construct a hypothesis
Test your hypothesis by doing an experiment
Analyze your data and draw a conclusion
Report your results. Was your hypothesis correct?

amazon

Did you reject or fail to reject your hypothesis?

Outline

1. Organizing a dataset
2. Descriptive statistics & variable types
3. Descriptive statistics: concepts & code

See also (and practice with a copy of):
Lecture 5.2 example code!

ORGANIZING A DATASET

- Lecture 5.1:
  - Importing a dataset
  - Inspecting a dataset for overall contents, shape, data types, sort by values, count values of a variable, describe, mean
- Today: More useful code for cleaning & organizing!
  - Columns: change the number shown, rename, re-order, drop, create
  - Rows:
    - change values in an observation (e.g., Bay, Chestnut, Buckskin → brown), drop observations, change the number of rows displayed
    - create a subset of data containing observations that meet certain criteria (e.g., only of horses 15 hands or higher)

ORGANIZING COLUMNS

Action	*Why you might do it*	*Example*	*Syntax*
Change the number of columns shown	You want to see all the variables, or just a few	Your dataset has 100 variables but the default data display doesn't show them all	`pd.set_option('display.max_columns', 100)`
Rename columns	A column has a long name and you plan on referencing it a lot	Your variable is called "GDP per capita" and you change it to "gdp"	`data.rename(columns={'GDP per capita': 'gdp'}, inplace=True)`
Drop columns	You plan to NEVER use this column in your analysis	A survey recorded the IP address of everyone who took it	`test.drop(['IP Address'], axis=1, inplace=True)`

Learning-how-to-learn challenge!!

Create columns

Re-order columns

ORGANIZING ROWS

Action	*Why you might do it*	*Example*	*Syntax*
Change the values for an observation	You have a lot of values you'd like to turn into broader categories	You have data on people from the US and Canada and just want to call it "North America"	`data['country'].replace(['US', 'Canada'], 'North_Am', inplace=True)`
Drop observations	You have observations you are sure you will NEVER want	You have a few survey respondents who were just testers, not real part of sample	`data.drop([0], axis=0, inplace=True)`
Change the number of rows shown	You have a lot of observations and actually want to see them all	Your dataset has 1000 observations and the default display doesn't show them all	`pd.set_option('display.max_rows', 500)`
Create a subset of data based on observations that meet certain criteria	You have a dataset that with observations over time and individual units, and you just want to study one element	You have economic data for 10 countries over 10 years, and just want to see one country's trend over time	`datacan = data[data['country'] == ‘Canada']`

SOME ADVICE ON MANIPULATING DATASETS: WORK FROM A COPY

1. Make sure you are not manipulating the original data
   - If it's stored externally, then whatever you do in the analysis won't affect the original
   - You can also immediately make a copy within your program and just work from the program

ADVICE CONTINUED: DOCUMENT EVERYTHING

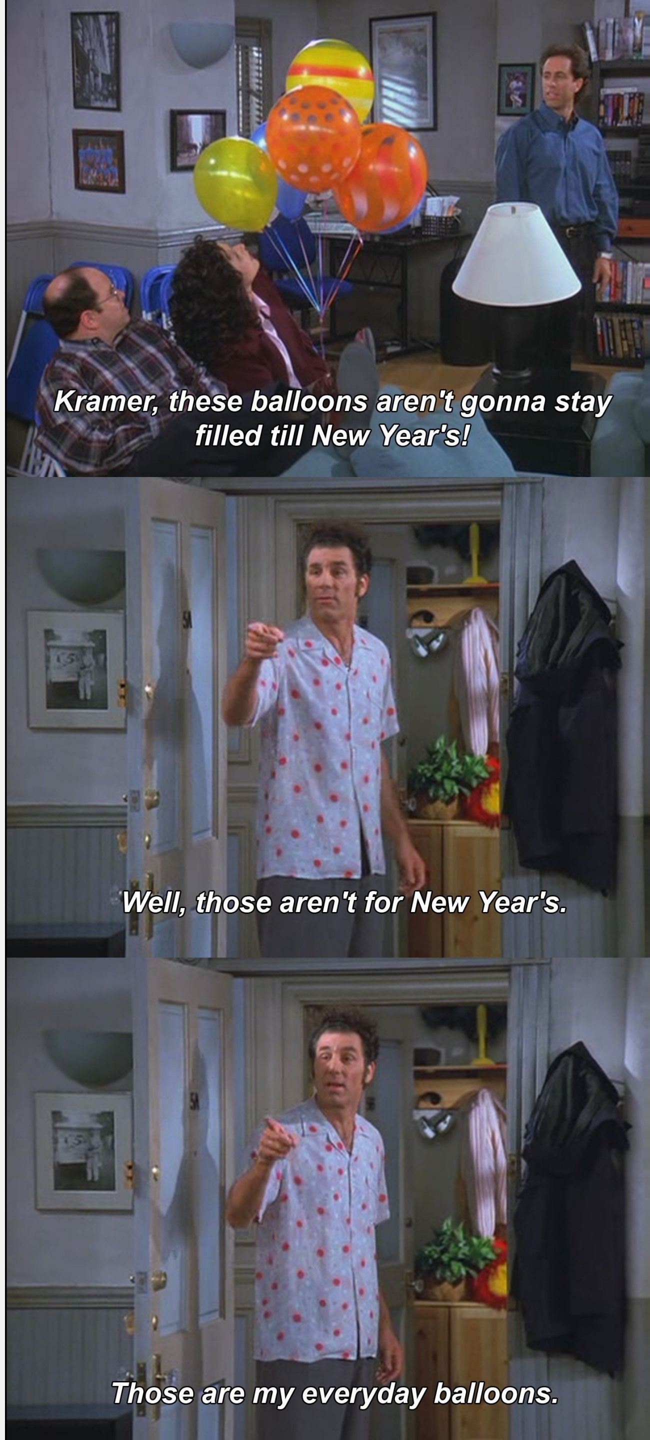
2. Document everything you do, even if (especially if) you are convinced you'll remember
   - It might seem obvious now that "data2" means a subset of data from China from 2008, but when you walk away from your code then return it may be less obvious
   - I like to name subsets of data things that will remind me what it is
     - datachn08 > data2 (But still worth recording this into your own mini codebook)
     - I also totally break these rules (e.g., calling GDP per capita "gdp" because I don't want to type out something like gdp_per_cap)
     - But then I am sure to write this down somewhere! (regular old text files, or whatever, are your friends!)

MORE ADVICE: DROP WITH CAUTION

3. When in doubt, don't drop rows or columns, I prefer to create subsets or rearrange
   - That said, if you have something you FOR SURE want to drop, proceed with caution and remember you did it by **documenting it** in a codebook (or at minimum with # in the code)
   - When working with subsets and you try to make a change that seems universal (like changing a column name), Python will give you a warning, **which is nice!**
   - I get it, it's tempting you want to cut everything you don't need, but worst case scenario you need to use it after all and then it's super annoying to do it all over

LAST ADVICE

4. Have fun out there!!
    - Just kidding, that’s annoying advice (but do have fun)
    - But don’t stress about memorizing cleaning techniques or learning all of the possible options out there
    - This is very much something you learn by working with messed up datasets!
    - It’s much more useful right now to build instincts around being able to think about what sort of format you need the data in and conceptually how you’d get it
        - Then you can look it up – if you can’t imagine what you need to do, you’re stuck
        - Of course, if you memorize enough techniques you could figure it out


Kramer, these balloons aren't gonna stay filled till New Year's!
Well, those aren't for New Year's.
Those are my everyday balloons.

BIG GOAL OF THIS COURSE: LEARN TO LEARN!

Thinking like a data scientist

- Example: Survey data with a column containing the date and time someone started the survey and another with the date and time they completed it
  - How can I get this in a format that tells me the number of minutes each person spent on it?
  - I need to know that I need to probably transform the data into time units, then create a column subtracting one from the other
  - So that's two things to figure out: convert, then create a third column based on a combination of other two

That said, we are teaching the code we absolutely use the most
(I use most of these for every dataset!)

Outline

1. Organizing a dataset
2. Descriptive statistics & variable types
3. Descriptive statistics: concepts & code

DESCRIPTIVE STATISTICS

Statistics: Using data to tell us things we didn’t already know, and to helps us think about how confident or certain we should be about this discovery

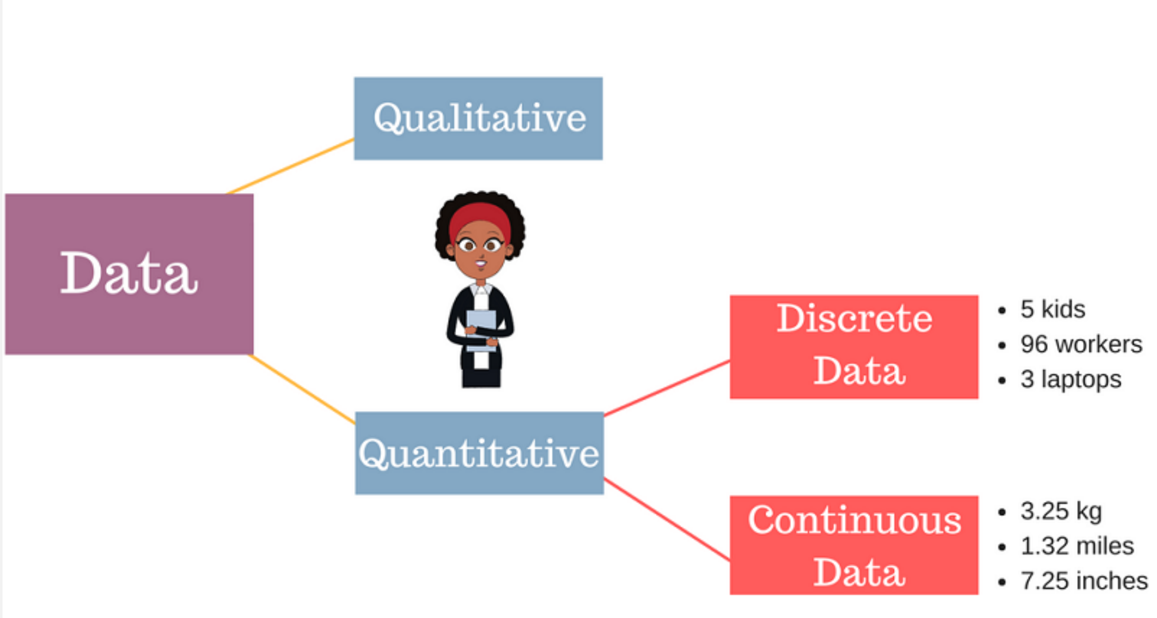
Descriptive Statistic: An individual statistic (often a single number) that summarizes or captures some aspect of a series of numbers (e.g., a variable)

These are all used to characterize data based on its properties.
We can share a lot of information about a topic of interest with just a few “summarizing” numbers

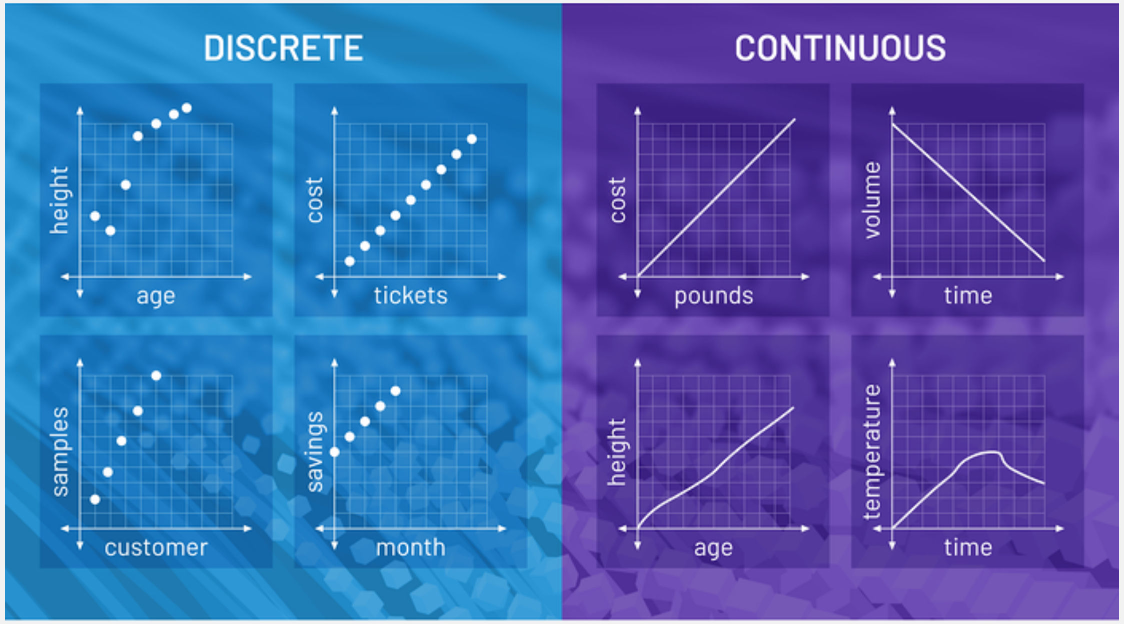
DESCRIPTIVE STATISTICS AND VARIABLE TYPES

- We know about data types:
  - int, float, string, bool
- We can also talk about variable types:
  - Numeric: Continuous or discrete
    - Continuous:Variables that could, in principle, take on any value (e.g., a person's height could be 66" tall or 66.1" or 66.11111, etc.)
    - Discrete:Variables that take on a finite set of values (e.g., the number of people who live in a household, or height if your data is rounded to inches or even 1.5 inches)
      - Floats are allowed, they just have to be finite

DISCRETE VS. CONTINUOUS DATA



Read more here.



Read more here.

DESCRIPTIVE STATISTICS AND VARIABLE TYPES

- Categorical or Qualitative: Objects/strings or other categories with names or numeric assignments that actually represent something else
  - Objects/strings: "color" variable that takes on "bay", "chestnut", etc.
  - Numeric assignments for non-numeric:
    - dummy variables (1 or 0 for True or False)
    - ordinal data: order matters, but not distance between them, e.g. survey responses:
      - 1 = Strongly agree
      - 2 = Slightly agree
      - 3 = Neutral
      - 4 = Slightly disagree
      - 5 = Strongly disagree

Outline

1. Organizing a dataset
2. Descriptive statistics & variable types
3. Descriptive statistics: concepts & code

DESCRIPTIVE STATISTICS

There are lots of different aspects of a variable we could want to summarize.
Four big aspects:

- Measures of frequency
- Measures of central tendency
- Measures of dispersion or variation
- Measures of position

MEASURES OF FREQUENCY

How often something occurs

1. Count:
   1. How many times does the color “bay” appear?
   2. How many respondents are from Canada?

```
data['color'].value_counts()
```

```
Bay    278
```

2. Percent
   1. What percentage of all horses in the sample are bay-colored?
   2. What percentage of respondents are from Canada?

```
data['color'].value_counts(normalize=True)
```

```
Bay    0.289885
```

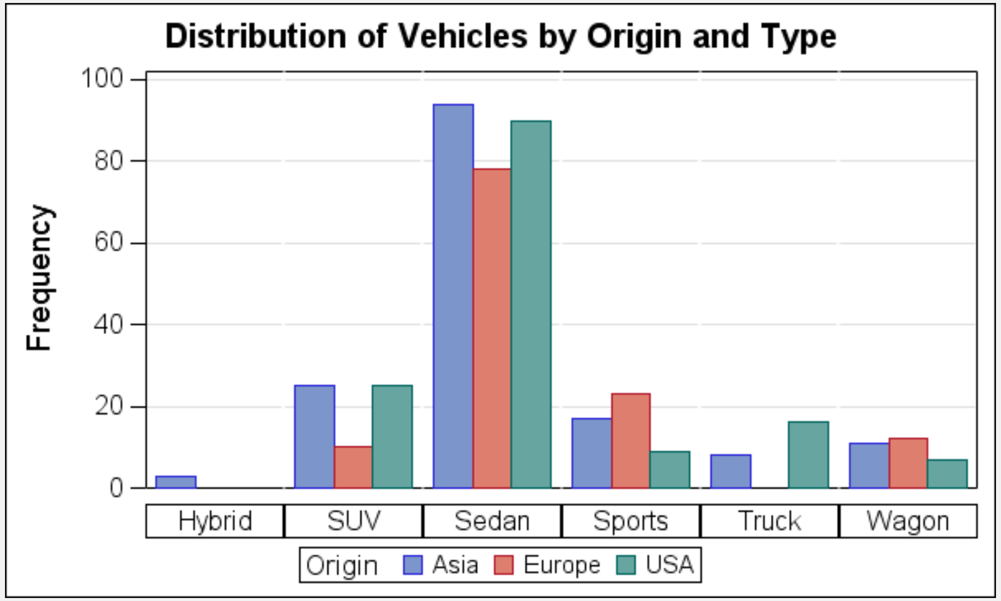
Generally tells us things like: How often something of interest (infections, tall horses) appears in our data

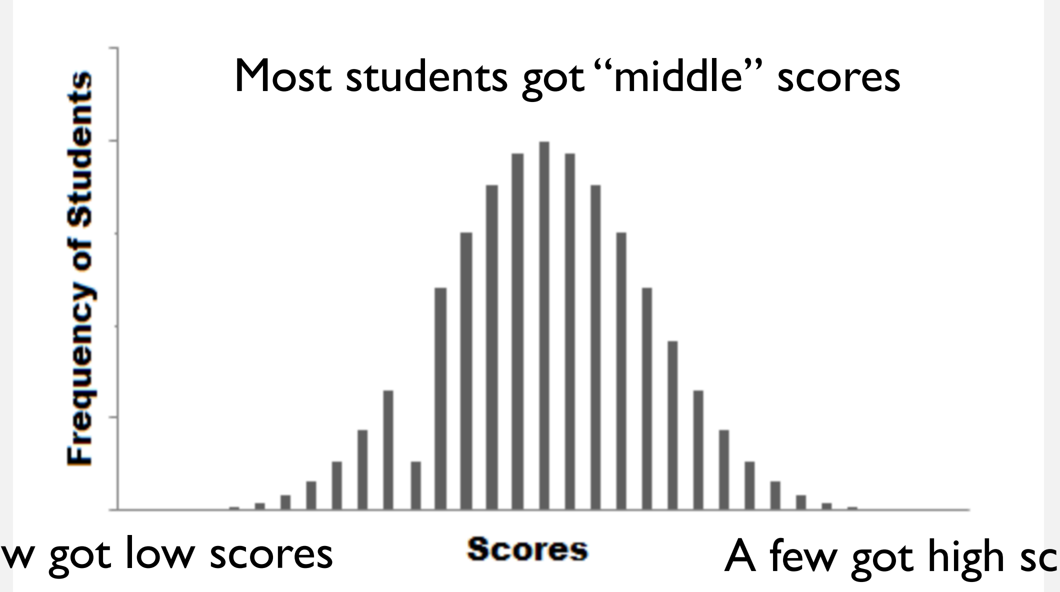
FREQUENCY DISTRIBUTIONS

- We can visualize how often particular values show up in a dataset
- We call these distributions of a variable
- We will talk a lot more about this in the next lecture 6.1

Frequencies of cars by origin and type

Frequency distribution of test scores by students





Most students got “middle” scores

A few got low scores

A few got high scores

DESCRIPTIVE STATISTICS

There are lots of different aspects of a variable we could want to summarize.
Four big aspects:

- Measures of frequency
- Measures of central tendency
- Measures of dispersion or variation
- Measures of position

MEASURES OF CENTRAL TENDENCY

Locates the distribution by various versions of "middle" points

All three of these are types of statistical "averages"

- **Mean**: Sum of numbers in a series / how many numbers

```
data['price'].mean()
```

- **Median**: The number splitting the lower half from the higher half in a series (for even numbers, the mean of the two middlemost

```
data['price'].median()
```

- **Mode**: The number that appears most often in a series

```
data['price'].mode()
```

Generally tells us things like: What the most common category is or what the central point (defined a few ways) of a distribution is

CENTRAL TENDENCY: EXAMPLE

- Group of people with ages:

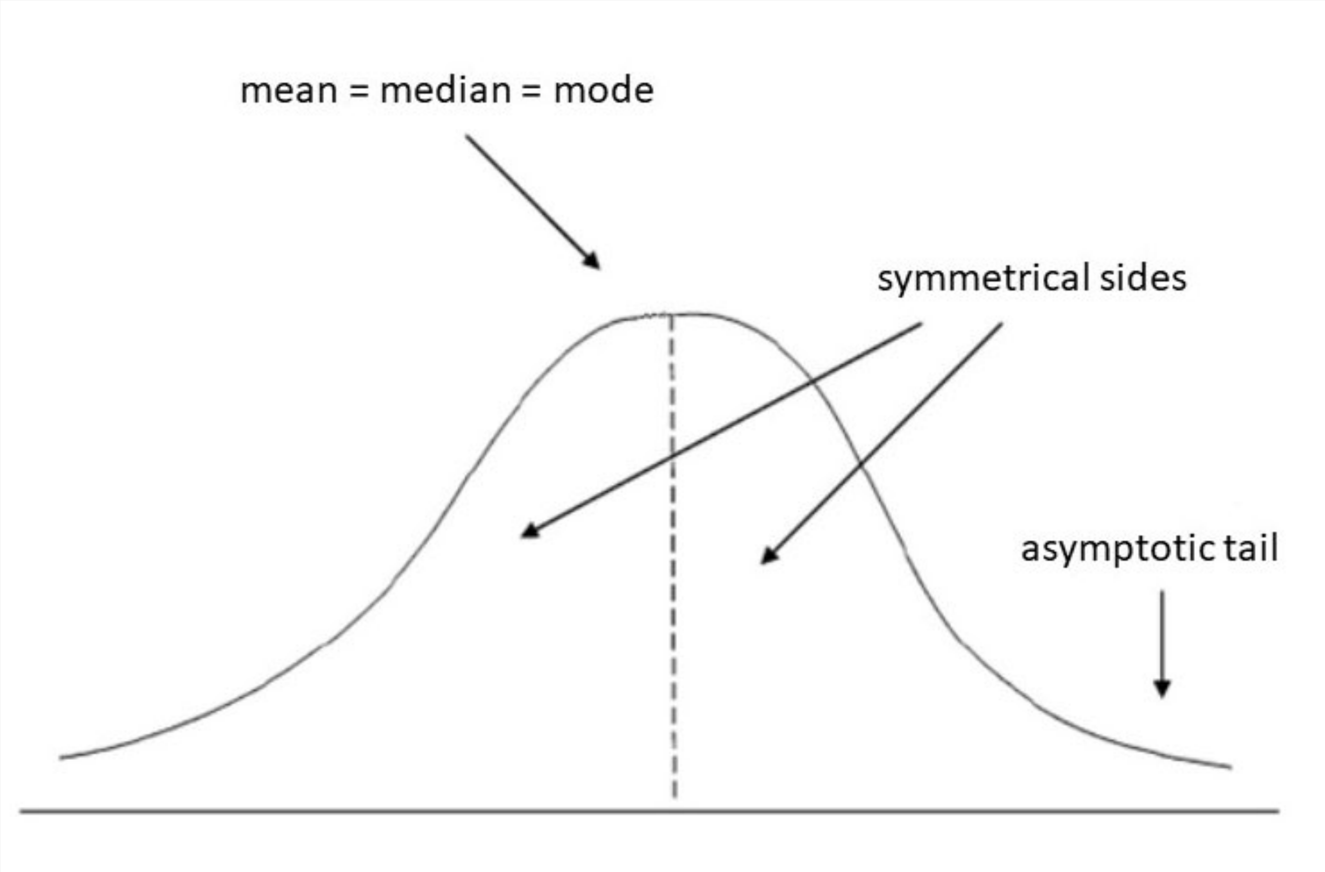
  ages = (21, 21, 21, 23, 25, 28, 29, 29, 30, 44)

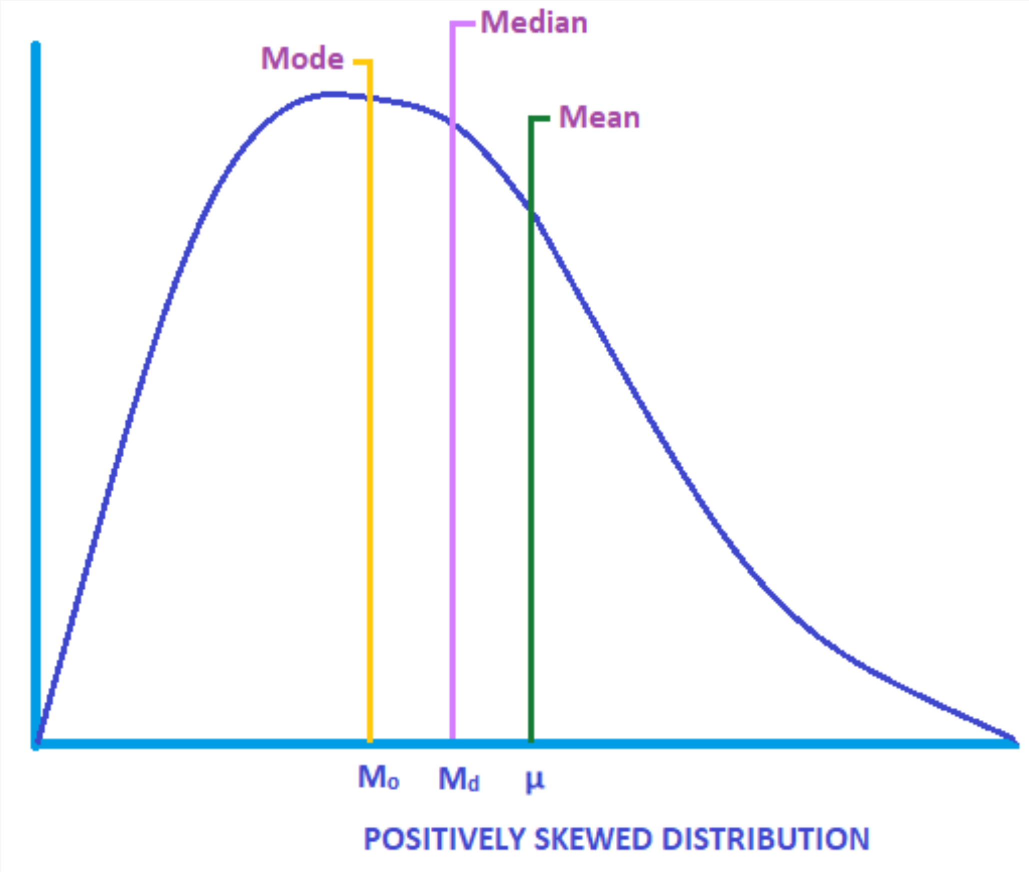
- Mean = sum(ages)/10 = 271/10 = 27.1
- Median = number that splits the series so half are below half are above = 26.5
  - Note, I could change some extremity values (e.g., change 44 to 89) and the median wouldn't change
  - But if I changed 28 to 25 that makes the median 25 as well
- Mode = 21

These are very different ways of summarizing the central tendencies of a variable

The most useful depends on the data and what you're trying to understand

Normal distribution





DESCRIPTIVE STATISTICS

There are lots of different aspects of a variable we could want to summarize.
Four big aspects:

- Measures of frequency
- Measures of central tendency
- Measures of dispersion or variation
- Measures of position

MEASURES OF DISPERSION OR VARIATION

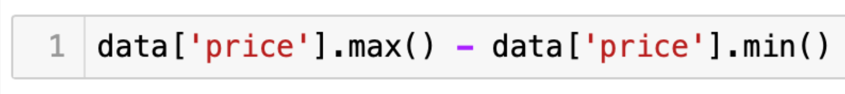
Identifies the spread out-ness of values in a particular variable

- **Range**: the difference between the highest and lowest values in a dataset
- **Quartiles**: Divide the upper and lower halves into their own upper and lower halves (find the median of the halves from the median)
- **Inter-quartile range (IQR)**: Q3-Q1 (cut top and bottom 25%)
- **Variance**: How "spread out" the data are
- **Standard deviation**: The square root of the variance (and the more commonly used of the two)

Generally tells us things like: How widely individuals in a group vary with respect to a certain aspect (e.g., height), and how this affects inference from central tendency statistics

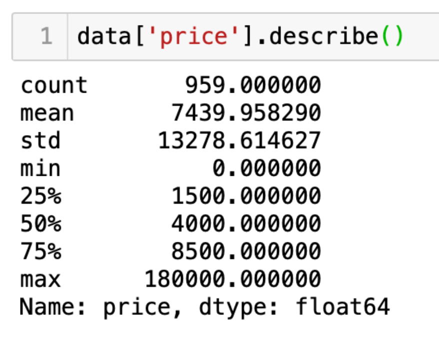
RANGE, QUARTILES, AND IQR

- Range example:
  - Price of most expensive horse – price of least expensive horse
  - Gives you a sense of the possible values this could take on
  - It doesn’t tell you anything about the distribution of prices
  - E.g. the range could be $100 to $500,000, but we don’t know if most are $100 or most are hundreds of thousands, or what



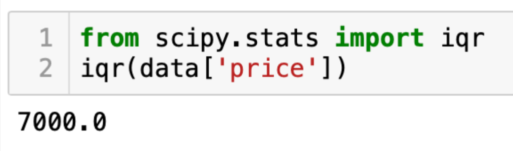
```
data['price'].max() - data['price'].min()
```

- Quartiles: More information about the distribution than just the median. Often we use it in terms of a 5-number summary:
  - Sample minimum, lower (or first) quartile, median (middle value), upper quartile (third quartile), sample maximum
  - Amazing! This is what we get from `describe()`!
- Unlike range, IQR tells us where most of the data lie



```
data['price'].describe()
```

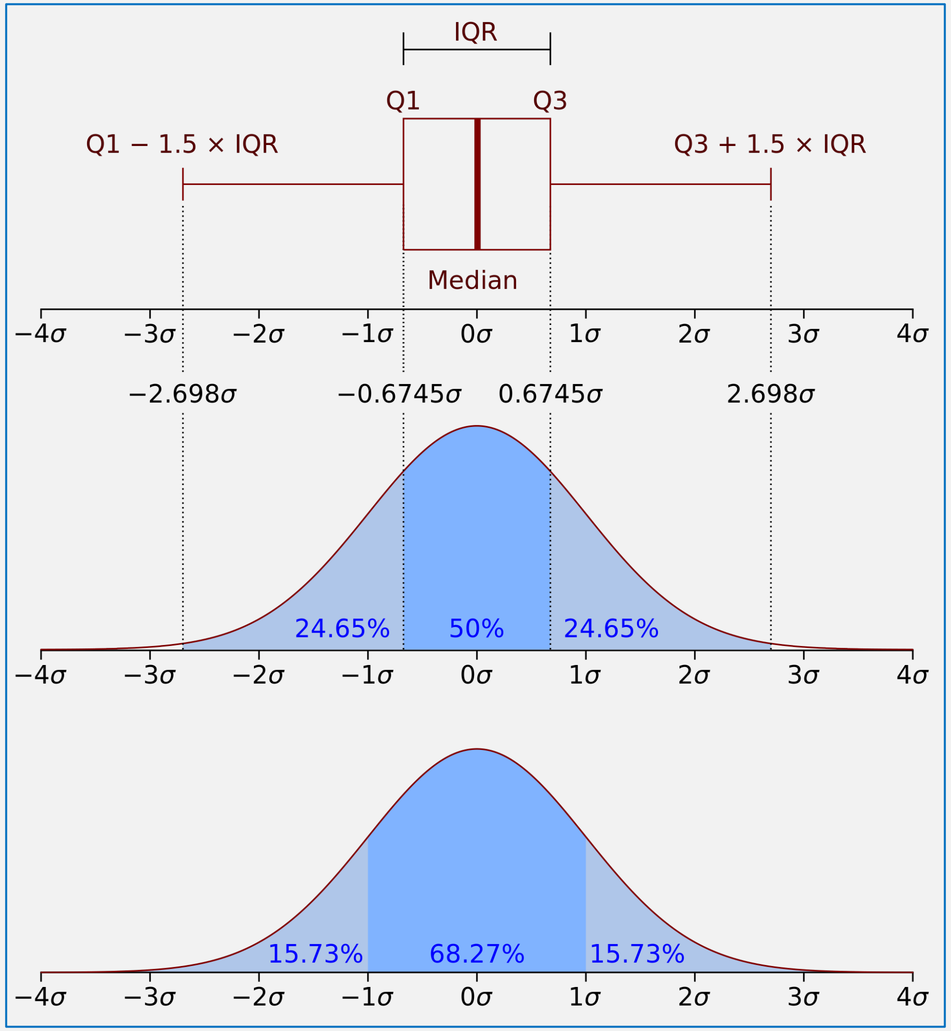
```
count       959.000000
mean       7439.958290
std       13278.614627
min           0.000000
25%        1500.000000
50%        4000.000000
75%        8500.000000
max      180000.000000
Name: price, dtype: float64
```



```
from scipy.stats import iqr
iqr(data['price'])
```

```
7000.0
```

QUARTILES AND IQR



IQR

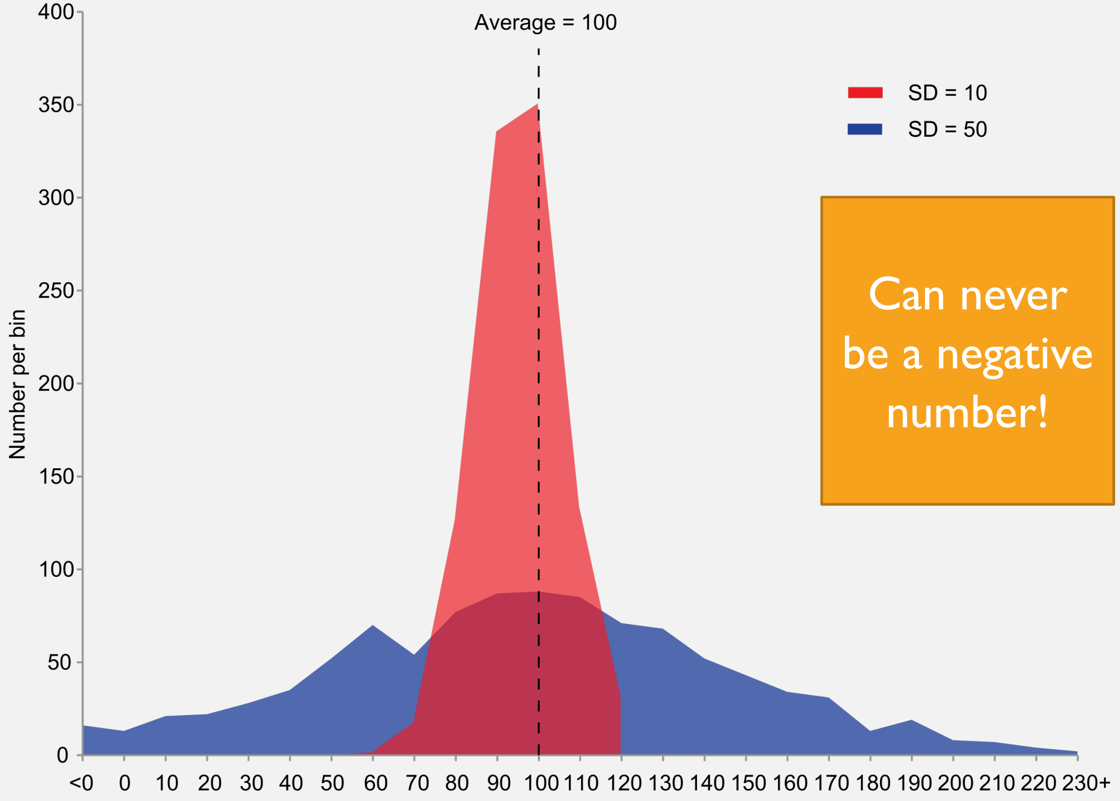
QUARTILES AND QUANTILES

Specialized quantiles	
2-quantiles	Median
3-quantiles	Terciles
4-quantiles	Quartiles
5-quantiles	Quintiles
6-quantiles	Sextiles
7-quantiles	Septiles
8-quantiles	Octiles
10-quantiles	Deciles
20-quantiles	Ventiles
100-quantiles	Percentiles
1000-quantiles	Permilles or milliles

- A quartile is a special (and relatively common) instance of a **quantile**
- **Quantile:** generic name for segments we've broken a variable into

VARIANCE AND STANDARD DEVIATION

Measures of spread out-ness



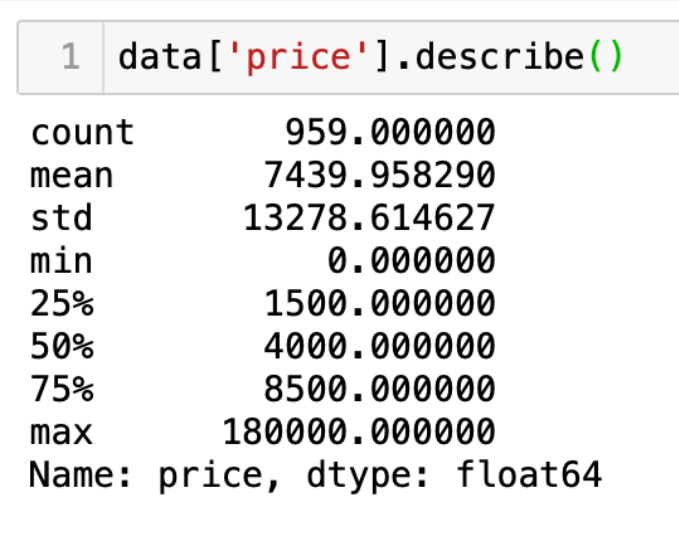
Can never be a negative number!

Variance: How spread out the data is from the mean
The average of the squared differences from the mean

Standard deviation: How spread out the data is from the mean in a format that makes more sense intuitively.
Std. dev. is in the units of the variable
The square root of the variance

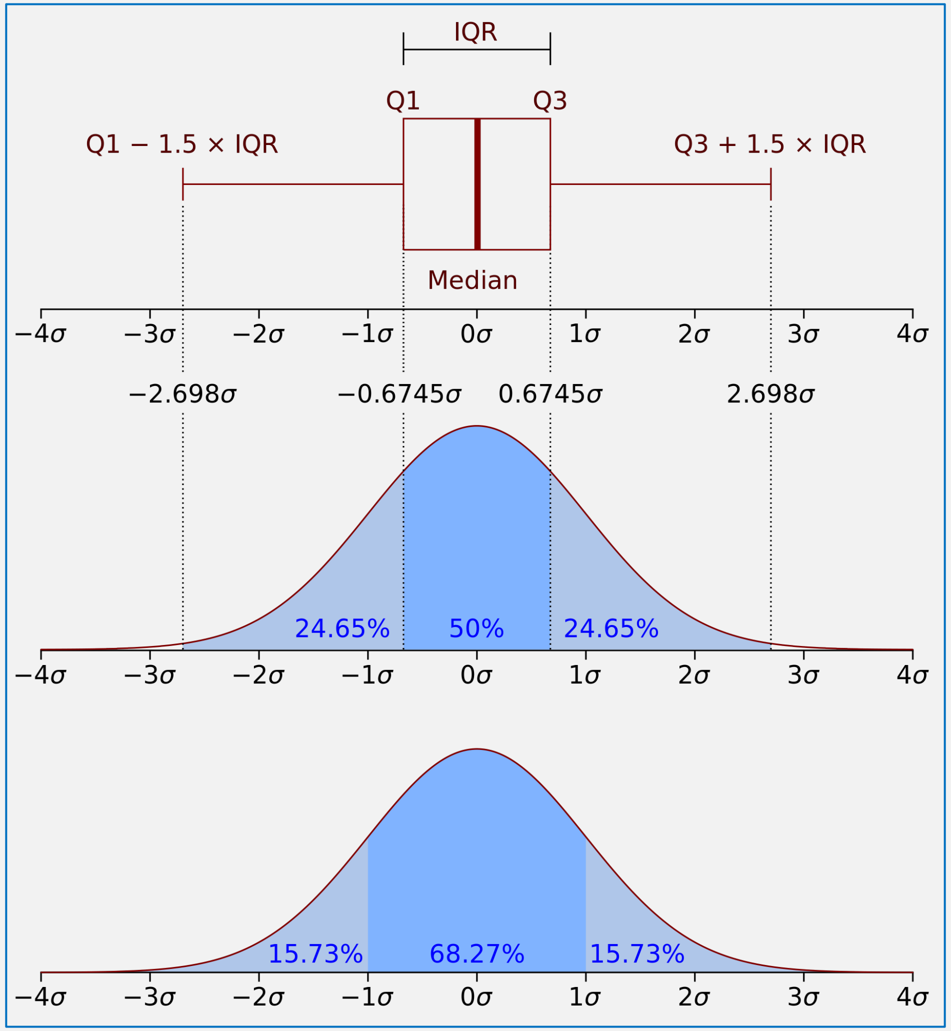
Not particularly informative without context, though larger standard deviations mean the data are more spread out from the mean compared to smaller standard deviations

VARIANCE AND STANDARD DEVIATION



```
data['price'].describe()
```

```
count       959.000000
mean       7439.958290
std       13278.614627
min           0.000000
25%        1500.000000
50%        4000.000000
75%        8500.000000
max      180000.000000
Name: price, dtype: float64
```



Std dev

DESCRIPTIVE STATISTICS

There are lots of different aspects of a variable we could want to summarize.
Four big aspects:

Measures of frequency

Measures of central tendency

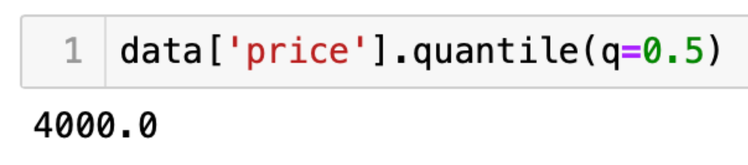
Measures of dispersion or variation

Measures of position

MEASURES OF POSITION

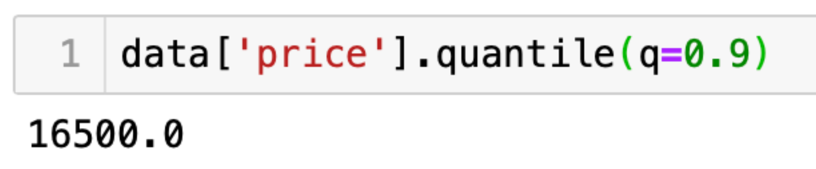
Describes how values fall in relation to one another

- **Rank**: The position of an observation relative to the others with respect to some variable
  - E.g., This person is 5th in their class in terms of GPA
  - E.g., This is the third-most-expensive horse
- **Percentile**: The percentage of scores equal to or lower than a particular score or observation (e.g., a child is in the 90th percentile for height)



```
data['price'].quantile(q=0.5)
```

```
4000.0
```



```
data['price'].quantile(q=0.9)
```

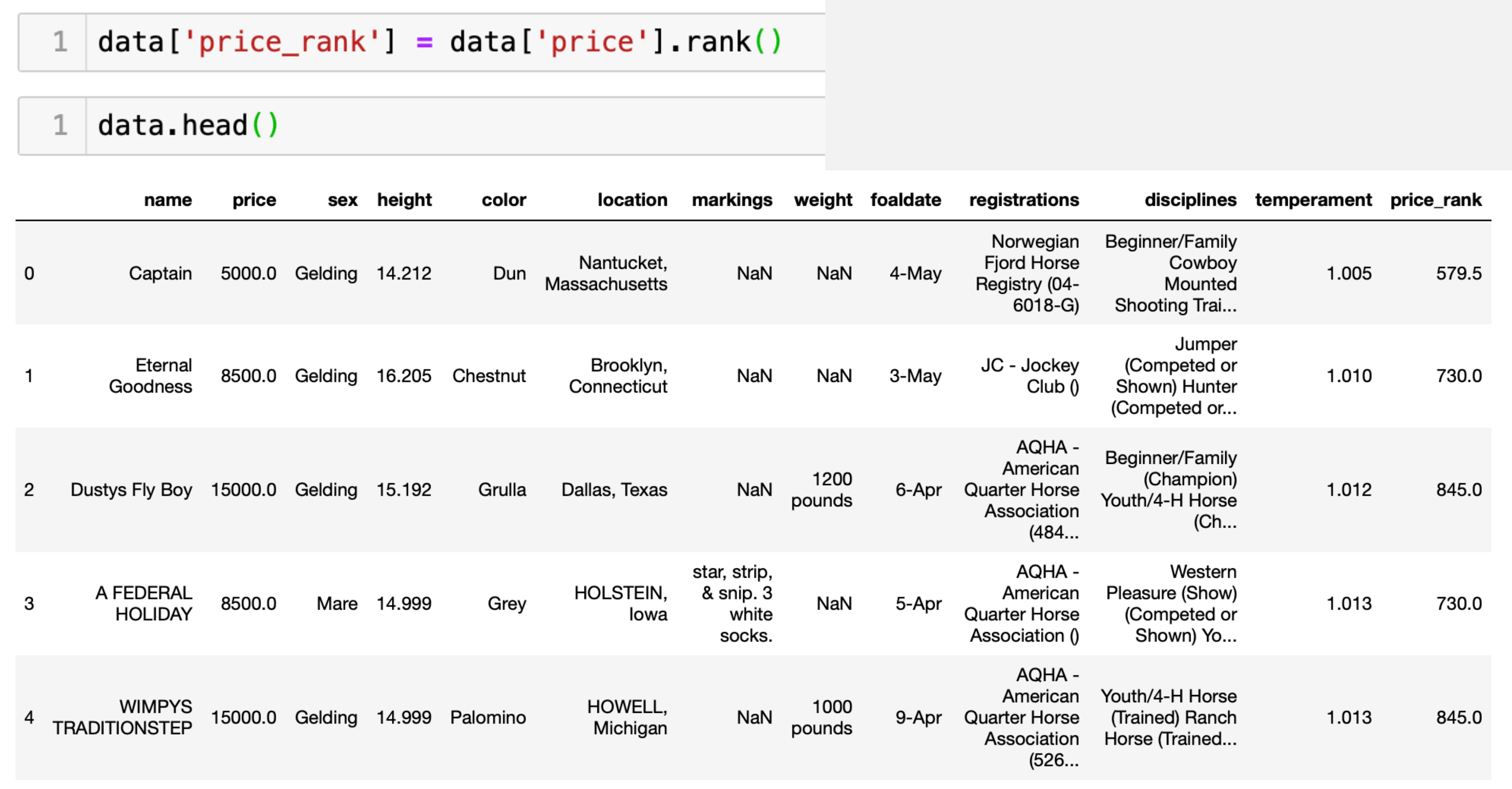
```
16500.0
```

This is the median!

Generally tells us things like: How good a score is compared to other scores

This is the 90th percentile!

RANK EXAMPLE



```
data['price_rank'] = data['price'].rank()
```

```
data.head()
```

	name	price	sex	height	color	location	markings	weight	foaldate	registrations	disciplines	temperament	price_rank
0	Captain	5000.0	Gelding	14.212	Dun	Nantucket, Massachusetts	NaN	NaN	4-May	Norwegian Fjord Horse Registry (04-6018-G)	Beginner/Family Cowboy Mounted Shooting Trai...	1.005	579.5
1	Eternal Goodness	8500.0	Gelding	16.205	Chestnut	Brooklyn, Connecticut	NaN	NaN	3-May	JC - Jockey Club ()	Jumper (Competed or Shown) Hunter (Competed or...	1.010	730.0
2	Dustys Fly Boy	15000.0	Gelding	15.192	Grulla	Dallas, Texas	NaN	1200 pounds	6-Apr	AQHA - American Quarter Horse Association (484...	Beginner/Family (Champion) Youth/4-H Horse (Ch...	1.012	845.0
3	A FEDERAL HOLIDAY	8500.0	Mare	14.999	Grey	HOLSTEIN, Iowa	star, strip, & snip. 3 white socks.	NaN	5-Apr	AQHA - American Quarter Horse Association ()	Western Pleasure (Show) (Competed or Shown) Yo...	1.013	730.0
4	WIMPYS TRADITIONSTEP	15000.0	Gelding	14.999	Palomino	HOWELL, Michigan	NaN	1000 pounds	9-Apr	AQHA - American Quarter Horse Association (526...	Youth/4-H Horse (Trained) Ranch Horse (Trained...	1.013	845.0

Outline

1. Organizing a dataset
2. Descriptive statistics & variable types
3. Descriptive statistics: concepts & code