LECTURE 04

Arrays and Variables

Storing many values in a single name.

Data 6 Summer 2022

Developed by students and faculty at UC Berkeley and Tuskegee University

data6.org/su22/syllabus/#acknowledgements-

Announcements!

Week 1

- **Office hours** start today
  - Tuesdays and Thursdays 1-3 PM in Evans 6
- **Homework 1** will be released today and due on 7/14 @ 11PM

Today's Roadmap

Lecture 04, Data 6 Summer 2022

1. NoneType
    a. `print()`
    b. Display vs. Output
2. Arrays
    a. Array Operations
3. Variables in Data Science

1. **NoneType**
2. Arrays
3. Variables in Data Science

`print()`

A very useful function for human display: `print()`

The **`print()`** function **displays** values.

- Works even if it's not the last line of a cell!
- Strings are displayed without quotes
- Can take multiple arguments of different types
- Sub-expressions are evaluated before display

```
In [18]: print(2)
         print("Hello, world!")

2
Hello, world!
```

```
In [19]: x = 3
         y = 4
         print(x, "+", y, "is equal to", x + y)

3 + 4 is equal to 7
```

A very useful function for human display: `print()`

The **`print()`** function **displays** values.

- Works even if it's not the last line of a cell!
- Strings are displayed without quotes
- Can take multiple arguments of different types
- Sub-expressions are evaluated before display

```
In [18]: print(2)
         print("Hello, world!")

2
Hello, world!
```

```
In [19]: x = 3
         y = 4
         print(x, "+", y, "is equal to", x + y)

3 + 4 is equal to 7
```

⚠️ **Note** ⚠️

Print **displays** values.
It **does not** produce cell output!

```
In [20]: print("10x Biggest number:")
         10 * max(5, 2, -1)

10x Biggest number:

Out[20]: 50
```

Terminology going forward:
Print means **display**
Output means **cell output**

Quick Check 1

What happens when we run the cell below?

```
print(15)
x = 3 + 4
x
print(14)
-3
```

Answer on Ed!

Quick Check

“Hello, World!”

This is a common “first” program to test that a programming language works as expected.

Now you understand it!!!

A **"Hello, world!" program** is generally a computer program that outputs or displays the message "Hello, world!". This program is very simple to write in many programming languages, and is often used to illustrate a language's basic syntax. "Hello, world!" programs are often the first a student learns to write in a given language,[1] and they can also be used as a sanity test to ensure computer software intended to compile or run source code is correctly installed, and that its operator understands how to use it.

```
print("Hello, World!")
```

https://en.wikipedia.org/wiki/%22Hello,_World!%22_program#Python

Review of `print()` and Typecasting

There are two common ways to print strings:

Multiple arguments

```
In [19]: x = 3
         y = 4
         print(x, "+", y, "is equal to", x + y)

3 + 4 is equal to 7
```

- Python inserts a space character for display
- Arguments can be different data types

String concatenation

```
In [2]: polygon = "square"
        s = 4
        print("The area of a " + polygon + \
              " with side length " + str(s) + \
              " is " + str(s ** 2) + ".")

The area of a square with side length 4 is 16.
```

- Programmer must insert space character for display
- **One** string argument, so all values must be cast to string

If your lines of code are too long, use the `\` character to break code into multiple lines.

NoneType

NoneType

There are infinitely* many integers, floating point numbers, and strings.

However, for the **NoneType** data type, there is only one value: **None**.

None is strange:

- Cells will **not** output expressions that evaluate to **None**.
- **None** **can** be displayed (i.e., printed).
- **None** is also referred to as the "null value."

```
In [6]: my_var = None
        type(my_var)
Out[6]: NoneType

In [7]: # No cell output!
        my_var

In [8]: # But it can be printed.
        print(my_var)
        None
```

*Actually a finite number because of how computers store information; take CS61C to learn more!

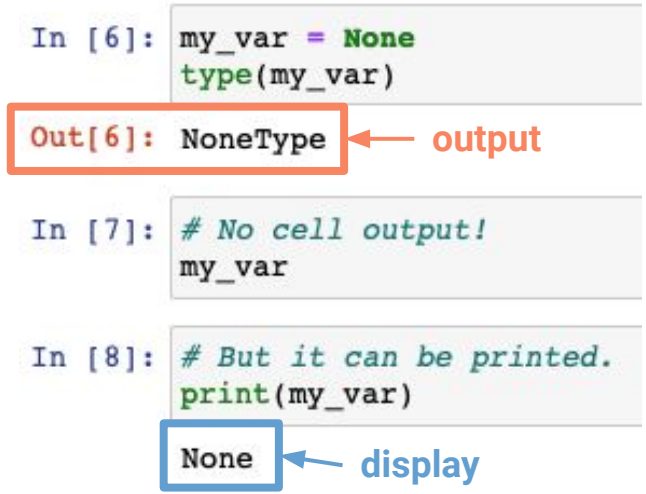
NoneType

There are infinitely* many integers, floating point numbers, and strings.

However, for the **NoneType** data type, there is only one value: **None**.

None is strange:

- Cells will **not** output expressions that evaluate to **None**.
- **None** **can** be displayed (i.e., printed).
- **None** is also referred to as the "null value."



print() returns **None**, therefore when evaluated as the last line in a cell:

- Print **displays** the value of the evaluated argument
- But the cell **does not output** anything!

*Actually a finite number because of how computers store information; take CS61C to learn more!

Quick Check 2

What is output and/or displayed when we run the cell below?

```
In [ ]: print("This value is", print(1))
```

Answer on Ed!

Quick Check

Python Data Types: Summary

Data Type	Category	Example Value(s)
`int`	Numeric	`3, -1`
`float`	Numeric	`3.4, -1.3,`
`str`	Text Sequence	`“”, “Hello, World!”, ‘234’`
`NoneType`	Special	`None`
`bool`	Numeric	`True, False`
`array`	NumPy Sequence	`array([1, 2, 3]), array([])`
`Table`	datascience	`Table()`
…	…	…

so far

coming up

our next topics

Arrays

Arrays = More Values!

An Array Is a Sequential Collection of Values

arranged like a line/queue

multiple values organized together

Use **`make_array()`** to create arrays. Values in an array must all be of the same data type, and Python will cast appropriately.

```
In [2]: make_array(5, -1, 0, 5)
Out[2]: array([ 5, -1,  0,  5])
```

Array with 4 **int**s

```
In [3]: make_array(5, -1, 0.3, 5)
Out[3]: array([ 5. , -1. ,  0.3,  5. ])
```

Array with 4 **float**s

```
In [4]: make_array(4, -4.5, "not a number")
Out[4]: array(['4', '-4.5', 'not a number'], dtype='<U32')
```

Array with 3 **str**s

An Array Is a Sequential Collection of Values

arranged like a line/queue

multiple values organized together

Use **`make_array()`** to create arrays. Values in an array must all be of the same data type, and Python will cast appropriately.

```
In [2]: make_array(5, -1, 0, 5)
Out[2]: array([ 5, -1,  0,  5])
```

Array with 4 **`int`**s

```
In [3]: make_array(5, -1, 0.3, 5)
Out[3]: array([ 5. , -1. ,  0.3,  5. ])
```

Array with 4 **`float`**s

```
In [4]: make_array(4, -4.5, "not a number")
Out[4]: array(['4', '-4.5', 'not a number'], dtype='<U32')
```

Array with 3 **`str`**s

Python can assign an entire array of values to a single name.

The order of a list is fixed (i.e., they will be arranged in the order specified when building the array), and values can be repeated.

```
In [5]: arr = make_array("hello",
                         "world",
                         "!")
        arr
Out[5]: array(['hello', 'world', '!'], dtype='<U5')
```

Arrays allow us to write code that performs computation on many pieces of data at once.

Side Note: `datascience` Package

The **`datascience`** Python package was written by UC Berkeley specifically for data science education.

We generally put the **import statement** in a cell at the top of our notebook.

- After running the import statement, we can then call package functions without *prepending* **`datascience`**.
- The **`make_array()`** function is from this package!

```
from datascience import *
```

"Import everything from the data science package"

```
In [1]: from datascience import *

In [2]: sq_array = make_array(1, 4, 9, 16, 25)
        sq_array

Out[2]: array([ 1,  4,  9, 16, 25])

In [3]: type(sq_array)

Out[3]: numpy.ndarray
```

Array Operations

American Community Survey (ACS) 2020

The following table is drawn from the **American Community Survey** (ACS) of 2020. It shows education levels of adults 25 years or higher by state.

We show AL, CA, FL, NY, TX.

(Now) How can we use **arrays** to analyze this data?

(Later) How is this data presented, and in what **societal context** was it analyzed?

	Estimated total state population	Estimated high school graduate or higher (%)	Estimated bachelor's degree or higher (%)
Alabama	3,344,006	86.9	26.2
California	26,665,143	83.9	34.7
Florida	15,255,326	88.5	30.5
New York	13,649,157	87.2	37.5
Texas	18,449,851	84.4	30.7

Compute % of Non-HS Graduates by State

	Estimated total state population	Estimated high school graduate or higher (%)	Estimated bachelor's degree or higher (%)
Alabama	3,344,006	86.9	26.2
California	26,665,143	83.9	34.7
Florida	15,255,326	88.5	30.5
New York	13,649,157	87.2	37.5
Texas	18,449,851	84.4	30.7

Demo

```
hs_or_higher = make_array(86.9, 83.9, 88.5, 87.2, 84.4)
```

```
below_hs = 100 - hs_or_higher
below_hs
```

Arithmetic on Arrays: Evaluation Returns a New Array

⚠️ Evaluating array expressions returns a **new array**; it does **not** change the original array.

```
100 - hs_or_higher
hs_or_higher
```

```
array([86.9, 83.9, 88.5, 87.2, 84.4])
```

Demo

Arithmetic on Arrays: Evaluation Returns a New Array

⚠️ Evaluating array expressions returns a **new array**; it does **not** change the original array.

```
100 - hs_or_higher
hs_or_higher
```

```
array([86.9, 83.9, 88.5, 87.2, 84.4])
```

Array Arithmetic is Element-Wise

1) Arithmetic with an array and a **numeric value**

```
below_hs = 100 - hs_or_higher
below_hs
```

```
array([13.1, 16.1, 11.5, 12.8, 15.6])
```

Demo

Element-Wise Arithmetic

This **element-wise** behavior works with all of the arithmetic operations you expect!

```
numbers_arr
```

```
array([ 5,  4,  9, 12, 18])
```

```
numbers_arr - 5
```

```
array([ 0, -1,  4,  7, 13])
```

```
numbers_arr // 2
```

```
array([2, 2, 4, 6, 9])
```

```
numbers_arr ** 2 - 1
```

```
array([ 24,  15,  80, 143, 323])
```

Estimate # Bachelor Degrees by State

	Estimated total state population	Estimated high school graduate or higher (%)	Estimated bachelor's degree or higher (%)
Alabama	3,344,006	86.9	26.2
California	26,665,143	83.9	34.7
Florida	15,255,326	88.5	30.5
New York	13,649,157	87.2	37.5
Texas	18,449,851	84.4	30.7

Demo

```
bs_or_higher = make_array(26.2, 34.7, 30.5, 37.5, 30.7)
state_pop = make_array(...) # see demo
```

```
bs_or_higher / 100 * state_pop
```

Arithmetic on Arrays: Evaluation Returns a New Array

⚠️ Evaluating array expressions returns a **new array**; it does **not** change the original array.

```
100 - hs_or_higher
hs_or_higher
```

```
array([86.9, 83.9, 88.5, 87.2, 84.4])
```

Array Arithmetic is Element-Wise

1) Arithmetic with an array and a **numeric value**

```
below_hs = 100 - hs_or_higher
below_hs
```

```
array([13.1, 16.1, 11.5, 12.8, 15.6])
```

2) Arithmetic with two **arrays of equal length** (same number of values).

```
bs_or_higher/state_pop*100
```

```
array([26.22734529, 34.71525729, 30.54635476, 37.45866503, 30.6895595 ])
```

Demo

Quick Check 3

1. Assign `f_temps` to the result of converting all celsius temperatures in the array `c_temps` to fahrenheit.

```
c_temps = make_array(30, 18, -4.5, 0, 3)
f_temps = ...
```

*Hint: Fahrenheit is Celsius * 9/5 + 32*

2. How many elements are in `empty_array`?

```
empty_array = make_array()
```

Answer on Ed!

Quick Check

Variables in Data Science

Terminology: Data, Tabular Data, and Records

> "[**Data**] are systematically collected elements of information about the world."
>
> – King, Keohane, and Verba. *Designing Social Inquiry* (1994).

A **dataset** is a collection of **data**.

In this course we will often use **tabular data**:

- **Columns**
- **Rows** (also known as **records**).

	Estimated total state population	Estimated high school graduate or higher (%)	Estimated bachelor's degree or higher (%)
Alabama	3,344,006	86.9	26.2
California	26,665,143	83.9	34.7
Florida	15,255,326	88.5	30.5
New York	13,649,157	87.2	37.5
Texas	18,449,851	84.4	30.7

row

column

Terminology: Variables

In data science, a **variable** is a **measurable** feature, attribute, and/or representation of a concept. It can have different values for different individuals.

variable

	Estimated total state population	Estimated high school graduate or higher (%)	Estimated bachelor's degree or higher (%)
Alabama	3,344,006	86.9	26.2
California	26,665,143	83.9	34.7
Florida	15,255,326	88.5	30.5
New York	13,649,157	87.2	37.5
Texas	18,449,851	84.4	30.7

Terminology: Variables

In data science, a **variable** is a **measurable** feature, attribute, and/or representation of a concept. It can have different values for different individuals.

A **column** contains/captures all **measurements** for a particular variable from our dataset.

Example 1: This column represents the estimated number of people 25 years or older in each state. Values are positive integers.

variable

	Estimated total state population	Estimated high school graduate or higher (%)	Estimated bachelor's degree or higher (%)
Alabama	3,344,006	86.9	26.2
California	26,665,143	83.9	34.7
Florida	15,255,326	88.5	30.5
New York	13,649,157	87.2	37.5
Texas	18,449,851	84.4	30.7

Terminology: Variables

In data science, a **variable** is a **measurable** feature, attribute, and/or representation of a concept. It can have different values for different individuals.

A **column** contains/captures all **measurements** for a particular variable from our dataset.

Example 2: This column represents the recorded sex of each respondent. Values are binary categories.

variable

Address	Person	Sex	Birthdate	In school?	Highest degree or level of school completed?	...
..., CA	1	F	mm/dd/yyyy	N	...	...
..., CA	2	M	mm/dd/yyyy	N	...	...
...	...	...	...	...	...	...

Terminology: Variables

In data science, a **variable** is a **measurable** feature, attribute, and/or representation of a concept. It can have different values for different individuals.

Variables are **empirical measurements**; they are often **metrics** that researchers create to approximate the specific dimensions of an abstract concept of a research question.

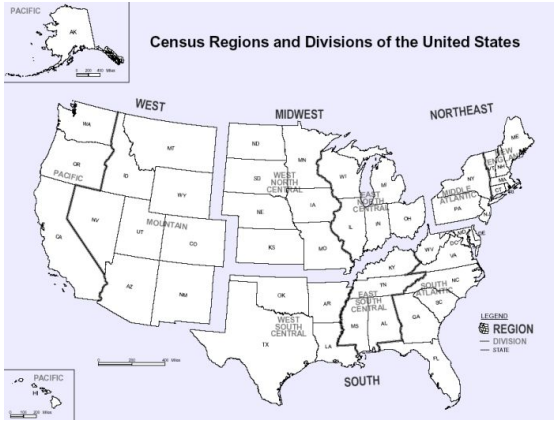
Side note: In computer science, Python names are also called *variables*. However, in this class we'll always try to refer to Python names as names to avoid confusing it with the data science definition of a **variable**.

variable

	Estimated total state population	Estimated high school graduate or higher (%)	Estimated bachelor's degree or higher (%)
Alabama	3,344,006	86.9	26.2
California	26,665,143	83.9	34.7
Florida	15,255,326	88.5	30.5
New York	13,649,157	87.2	37.5
Texas	18,449,851	84.4	30.7

Case Study: American Community Survey (2020)

Variables are **empirical measurements**; they are often **metrics** that researchers create to approximate the specific dimensions of an abstract concept of a research question.



About the ACS

Conducted annually by the US Census Bureau.

- Mailed to a sample of **~3.5 million household addresses** in 50 states + DC + Puerto Rico.
- ACS builds on Census questions and also asks questions on education, employment, internet access, and transportation.

Important uses at local and national level:

- Distribution of federal/state funds
- Economic development (infrastructure, hospitals, schools, bridges, etc.)
- Emergency management

Research Question and Data Collection

Pose the Research Question

What is the level of educational attainment in 2020 among US resident adults?

Household survey → **Private dataset by household** → ***aggregation*** → **Public dataset by geographic region**

Address	Person	Sex	Birthdate	In school?	Highest degree or level of school completed?	...
..., CA	1	F	mm/dd/yyyy	N	...	...
..., CA	2	M	mm/dd/yyyy	N	...	...
...	...	...	...	...	...	...

(this lecture)

	Estimated total state population	Estimated high school graduate or higher (%)	Estimated bachelor's degree or higher (%)
Alabama	3,344,006	86.9	26.2
California	26,665,143	83.9	34.7
Florida	15,255,326	88.5	30.5
New York	13,649,157	87.2	37.5
Texas	18,449,851	84.4	30.7

The definition of variables is impacted not only by **researchers' interests**, but also the process and limitations of **data collection**.

Defining Concepts in the Research Question

Pose the Research Question

What is the level of **educational attainment** in 2020 among US resident adults?

Translate a concept into a variable

Concept: Education

Variable: Highest degree received by academic institution

Define the variable domain (i.e., all possible values)

- Less than high school graduate
- High school graduate (includes equivalency)
- Some college or associate's degree
- Bachelor's degree
- Graduate or professional degree

Consider other factors that may influence the outcomes of the study, and repeat

Variable domains can be categorical, numeric, or qualitative! In this course we focus on the first two.

Quick Check 4

What is the level of educational attainment in 2020 among **US resident adults**?

How can we define variable(s) that capture the concept of "**US resident adults**"? (Open Ended)

Hint: Defining multiple variables can help capture a single concept!

Consider "US resident" and "adult" separately.

Answer on Ed!

Quick Check

Defining Concepts in the Research Question

Pose the Research Question

What is the level of educational attainment in 2020 among **US resident adults**?

Translate a concept into a variable

Concept: US resident	Concept: Adult
Variable: Address	Variable: Age bracket

Define the variable domain (i.e., all possible values)

US Postal address of a residence *[where the survey is mailed]*	Below 18 18 to 25 (“young adult”) 25 and up *[recorded from age]*

Consider other factors that may influence the outcomes of the study, and repeat

Identifying Confounding Variables

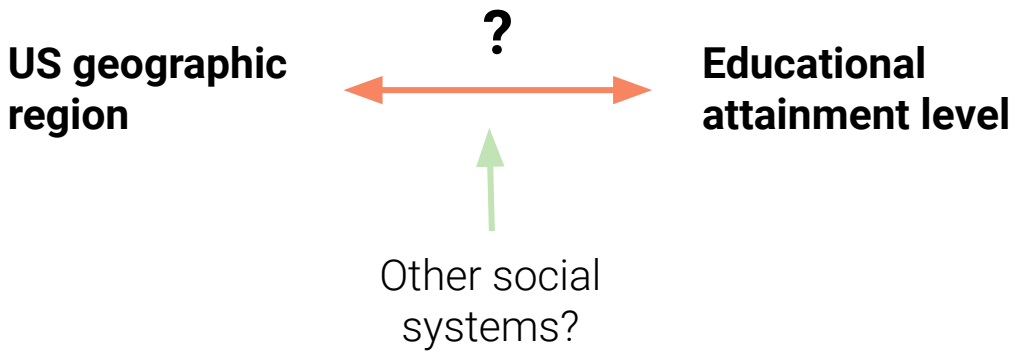
Pose the Research Question

What is the level of educational attainment in 2020 among **US resident adults**?

Translate a concept into a variable

Define the variable domain (i.e., all possible values)

Consider other factors that may influence the outcomes of the study, and repeat



Confounding variables can be linked to other concepts in a way that makes two concepts appear related (even when they are not).

Identifying Confounding Variables

Pose the Research Question

What is the level of educational attainment in 2020 among **US resident adults**?

US geographic region ? **Educational attainment level**

<u>Race/Ethnicity</u>
White alone
White alone, not Hispanic or Latino
Black alone
American Indian or Alaska Native alone
Asian alone
Native Hawaiian and Other Pacific Islander Alone
Some other race alone
Two or more races
Hispanic or Latino Origin

<u>Sex</u>
Male
Female

<u>Income/Earnings</u>
(numeric amount)

These are just a few **confounding variables** measured in the ACS. **What kinds of confounding variables would likely *not* be measured in a government study?**

Data and the Government

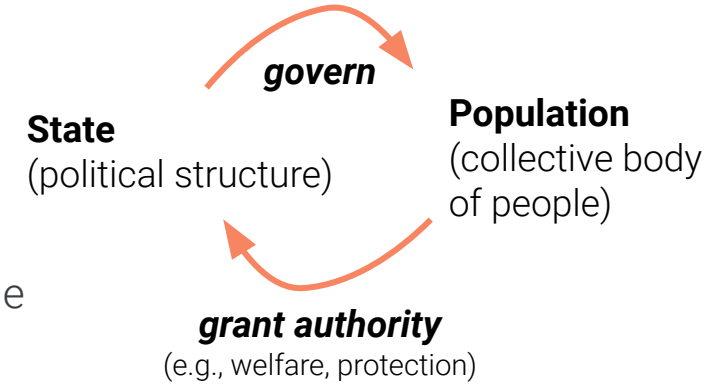
Variable Definitions Have Human Contexts!

Variables are empirical measurements; they are often **metrics that researchers create** to approximate the specific dimensions of an abstract concept of a research question.

Take **Data 104**: Human Contexts and Ethics to learn more!

The ACS is a United States **government survey**. The agency's goals influence the construction, collection and interpretation of data.

- First introduced in 2005, the ACS was a product of the expanding US administrative state.
- Administered by the Census Bureau



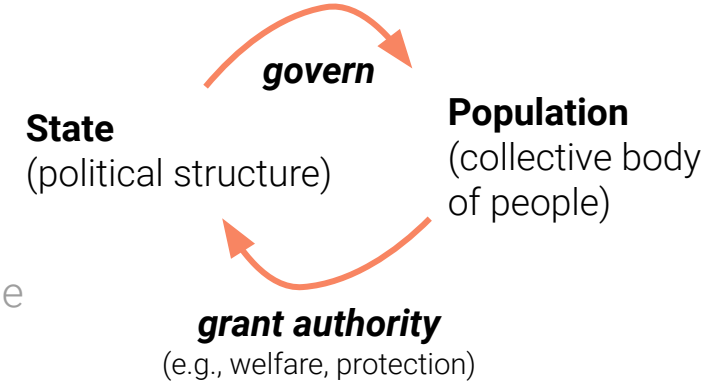
Variable Definitions Have Human Contexts!

Variables are empirical measurements; they are often **metrics that researchers create** to approximate the specific dimensions of an abstract concept of a research question.

Take **Data 104**: Human Contexts and Ethics to learn more!

The ACS is a United States **government survey**. The Census Bureau's goals influence the construction, collection and interpretation of data.

- First introduced in 2005, the ACS was a product of the expanding US administrative state.



Federal government agencies, businesses, and local agencies all use ACS data. Some **applications**:

- Labor statistics, social welfare
- Agriculture, education, economic production
- Insurance, disease control, health surveillance

US government studies like the ACS and the Census **normalize and categorize** for the state's purpose of administrative **management** of a population.

Data *Represent* the World to *Do Work* in the World

Variables are empirical measurements; they are often **metrics that researchers create** to approximate the specific dimensions of an abstract concept of a research question.

Take **Data 104**: Human Contexts and Ethics to learn more!

Classification: Implicit and explicit social organization of things, people, and knowledge into discrete categories governed by identifiable principles.

- When people classify, their judgments (**perspectives**, **biases**) enter into and are reproduced and embedded in infrastructures, systems, and devices.

“What’s Counted, Counts.”

Official **categories** map onto social distinctions and hierarchies—and, in turn, normalize them.

- Everything that does not fit is excluded/deviant.

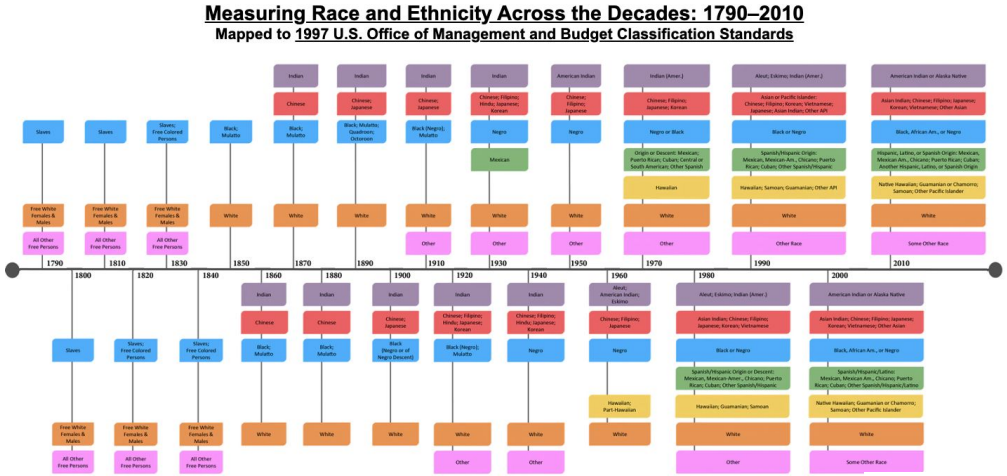
3 **What is Person 1’s sex?** *Mark (X) ONE box.*

☐ Male ☐ Female

- Sex as a binary category
- Male listed before Female
- Gender not measured

- **Categories** and **classifications** could change over time.

  In government data, these changes often reflect social changes in population management.



Link

Redefining Data Science

There is **no such thing as "impartial data."** 🙂 All data are collected by inherently human systems.

As data scientists, we often obtain data from existing sources.

- We search for **data** and understand the **contexts** for how and why the data were collected
- We tweak the scientific methodology for analysis
- And we present the original contexts and their caveats as part of our results

(more next lecture)

If we cannot answer our originally posed question:

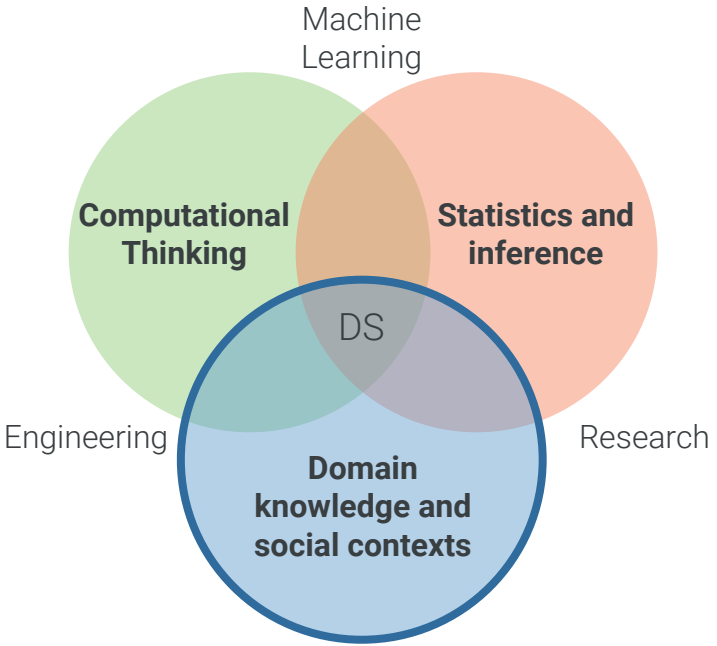
- We can look for **more data** (and understand those new contexts)
- We can seek **more contexts** for our existing data (e.g., by engaging with the communities of interest, or field experts)
- Or we can reframe our question

Data Science necessitates dialogue!

Bioethics
A Closing Note on Historical Context

This course is at the intersection of data, computing and society; as a result, we believe it is important to acknowledge these historical and contemporary social context.

The **Bioethics Center at Tuskegee University** defines **bioethics** as not just the study of ethical issues around biology, medicine and technologies, but as the **promotion of "life,"** which itself is interconnected and interdependent—bioethics therefore incorporates communities when discussing health disparities, business ethics, public health, engineering, and more. Like this course, bioethics is interdisciplinary!



Questions to Ask

- **How was this data collected?**
  - Who was included and who wasn't?
  - Is personal data being used with informed consent from participants?
- **How is this data being used?**
  - Who may be impacted by this data?
  - What conclusions can be drawn from the data? What are the impacts of these conclusions on policy, etc.

Data science is all about the context in which data are collected and used.

In Conclusion...

Summary

- The **`print()`** function is used to **display** things, but does have an output itself
- The Python **NoneType** indicates a “null” value — It represents the **absence** of a value.
  - **`None`** can be printed, but is not outputted
- We use **arrays** to store and work with multiple values at once
  - We can perform (element-wise) arithmetic operations and other functions on arrays
- In data science, a **variable** is a measurable feature, attribute, and/or representation of a concept
  - We will use “name” to refer to Python names and “variable” to refer to data science variables
- The **context** of our data matter a lot, and we should always examine the impact of data science on society

Recap

- NoneType
- Arrays
- Variables in Data Science
- Social Contexts and Implications

Next Time

- NumPy
- Array Indexing
- Working with Data