Data Collection

CSC 380 - Principles of Data Science

Lecture 3.2

Administrative

- Reading for this week is posted on D2L
- Everything related to the course will be at the newly launched class website. csc380.beingenfa.com

In the last lecture,

- Introduction to Pandas

In this lecture,

- Research Design for Statistical Analysis
  - Causation versus Correlation
  - Sampling
- Revisit the Data Science Process
- Data Collection
- Data Processing

1 .Statistical Analysis Pipeline

1. Plan research design
2. Collect data from a sample
3. Visualize and summarize the data (plots and summary stats)
4. Make inferences from data (i.e. estimate stuff, test hypotheses, …)
5. Interpret results

Credit : Much of the material of slides under this topic is From fall 22 slides by Prof.Jason Pacheco

1.1.1 Research Design

- Observational & Natural Experiment
- Case Studies
- Surveys
- Randomized Control : control, randomise, replicate

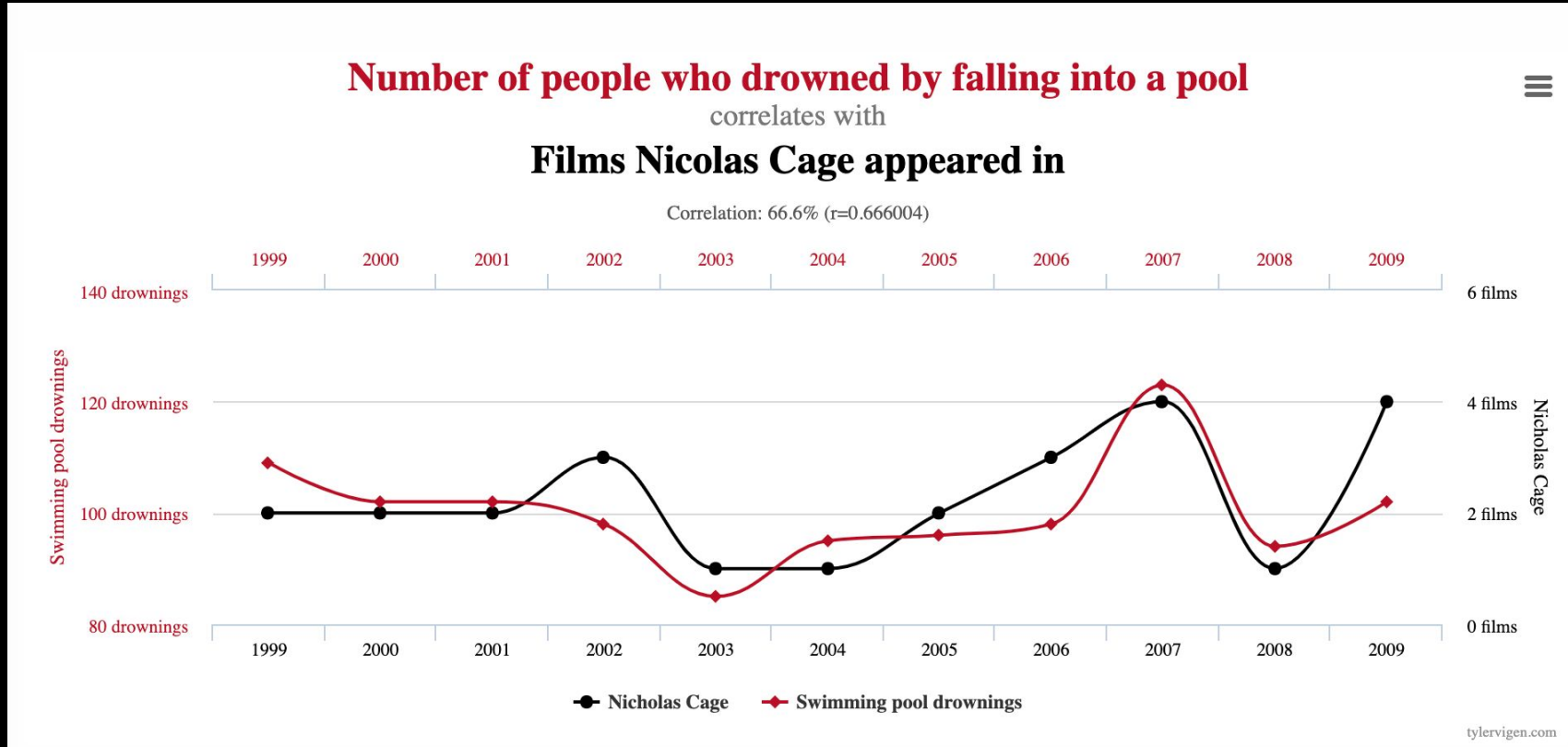
1.1.2 Causation versus Correlation

Covariance : how two random variables in a data set will change together

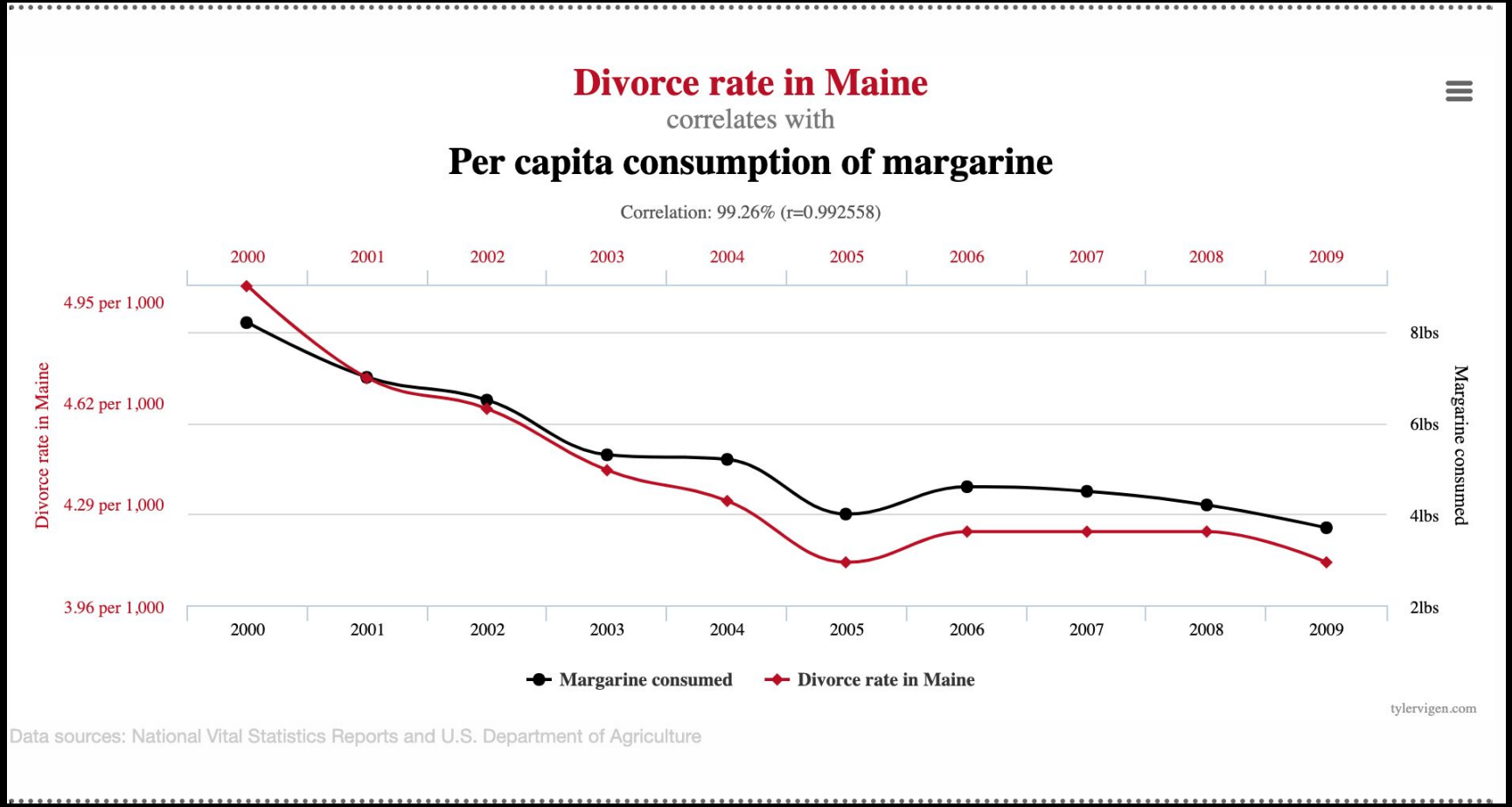
Correlation : how two random variables are **related**

Causation : how one variable causes an effect on another variable

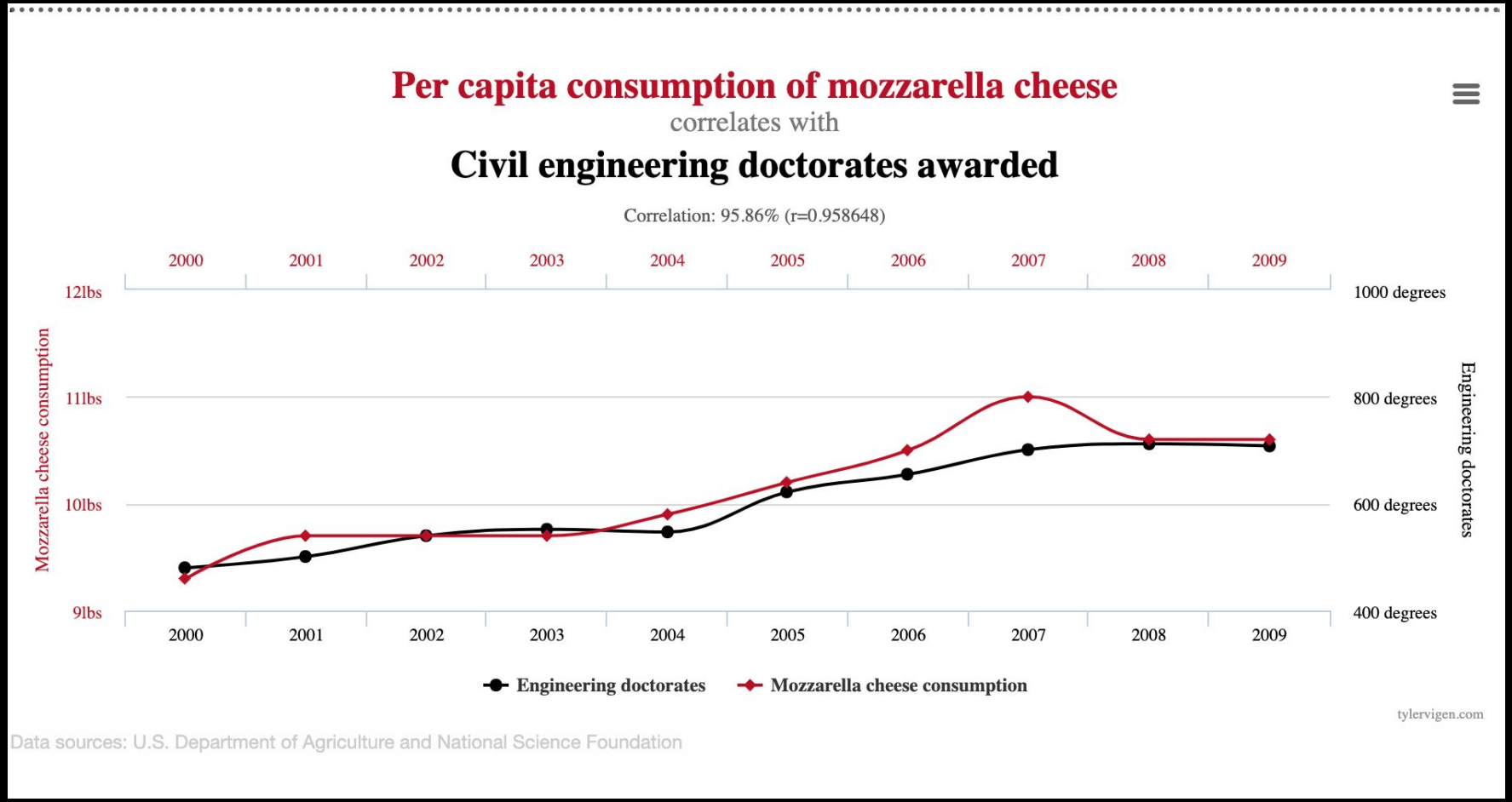
Correlation does not imply causation



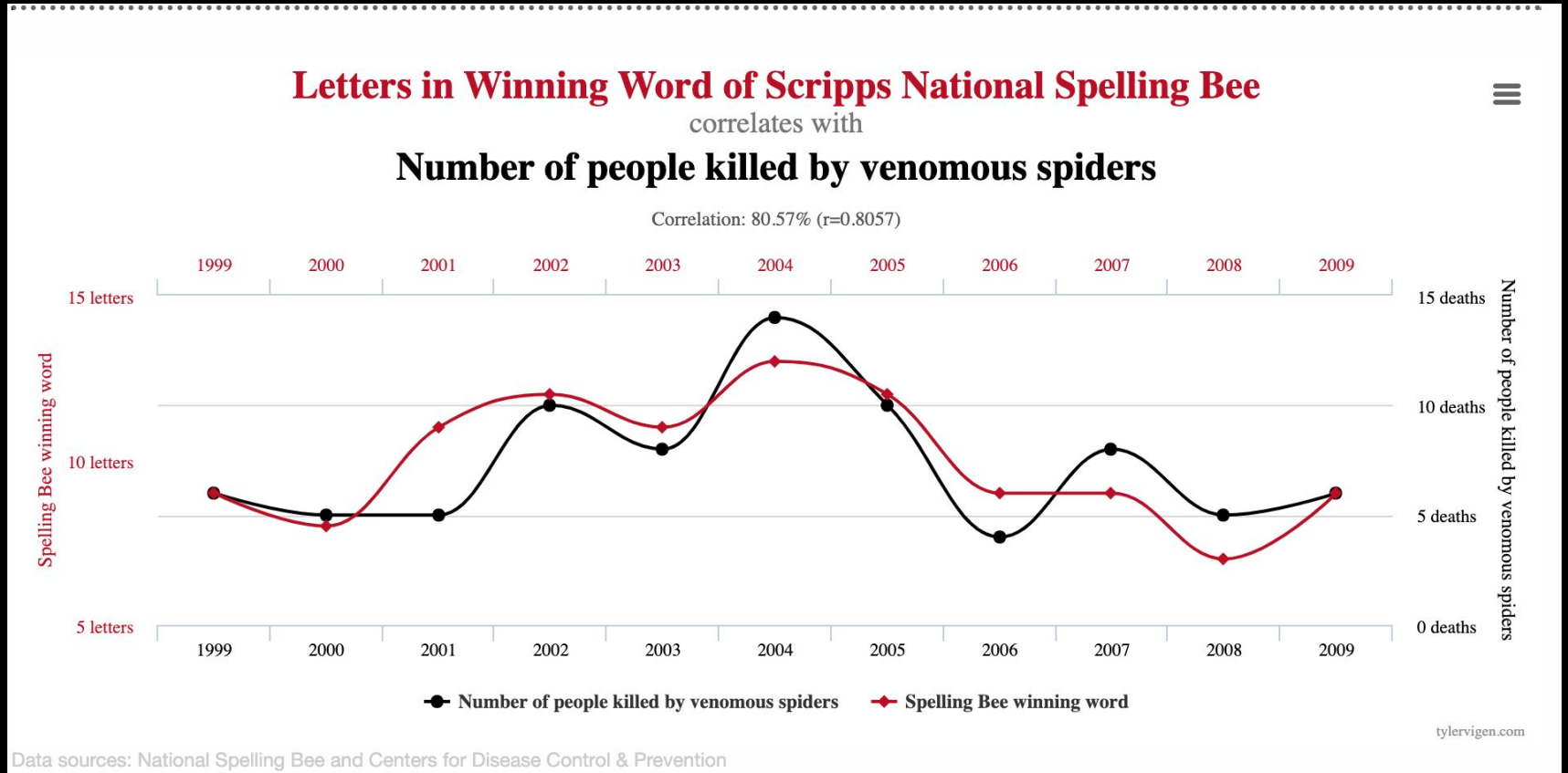
Credit : https://www.tylervigen.com/spurious-correlations



Credit : https://www.tylervigen.com/spurious-correlations



Credit : https://www.tylervigen.com/spurious-correlations



Credit : https://www.tylervigen.com/spurious-correlations

1.1.3 Confounding Variables

They are like extra independent variables that are having a hidden effect on your dependent variables.

1.1.5 Controlling for Confounders

- Stratified Sampling

- Probabilistic Model

1.1.6 Randomized Controlled Experiments

1. Control for effects of confounders by comparing several treatments

2. Randomize the assignment of subjects to treatments to eliminate bias due to systematic differences in categories

3. Replicate experiment on many subjects, to reduce chance of variation in the results

Example : Vaccine trials

1. Placebo : Control Subjects are randomly selected to receive either the vaccine or an injection of saline solution

2. Randomize Stratified sampling with age strata: 12-15yrs, 16-55yrs, 55+yrs with ~40% in the latter strata

3. Replicate Experiment is repeated at multiple sites in several countries

Trial Enrollment

The landmark phase 3 clinical trial enrolled **46,331** participants at **153** clinical trial sites around the world.

Trial Geography

Our trial sites are located in **Argentina, Brazil, Germany, Turkey, South Africa** and the **United States.**

Participant Diversity

Approximately **42%** of overall and **30%** of U.S. participants have diverse backgrounds.

Participants	Overall Study	U.S. Only
Asian	5%	6%
Black	10%	10%
Hispanic/Latinx	26%	13%
Native American	1.0%	1.3%

49.1% of participants are male and **50.9%** are female

Participant Age

Ages 12-15	2,260
Ages 16-17	754
Ages 18-55	25,427
Ages 56+	17,879

https://www.pfizer.com/science/coronavirus/vaccine/about-our-landmark-trial

1.2.1 Data Collection

- What can I measure?
- What shall I measure?
- How shall I measure it?
- How frequently shall I measure it?
- What obstacles prevent reliable measurement?

1.2.2 Reasons for Sampling

- Necessity
- Practicality
- Cost-effectiveness
- Manageability

1.2.3 Population Parameter vs. Sample Statistic

Population parameter: A measure that describes the whole population.

Sample statistic: A measure that describes the sample and reflects the population parameter.

Example: Political Leaning

1.2.4 Sampling Error

The sampling error is the difference between the population parameter and the sample statistic

1.2.5 Sample bias

When the sample is not representative of the population

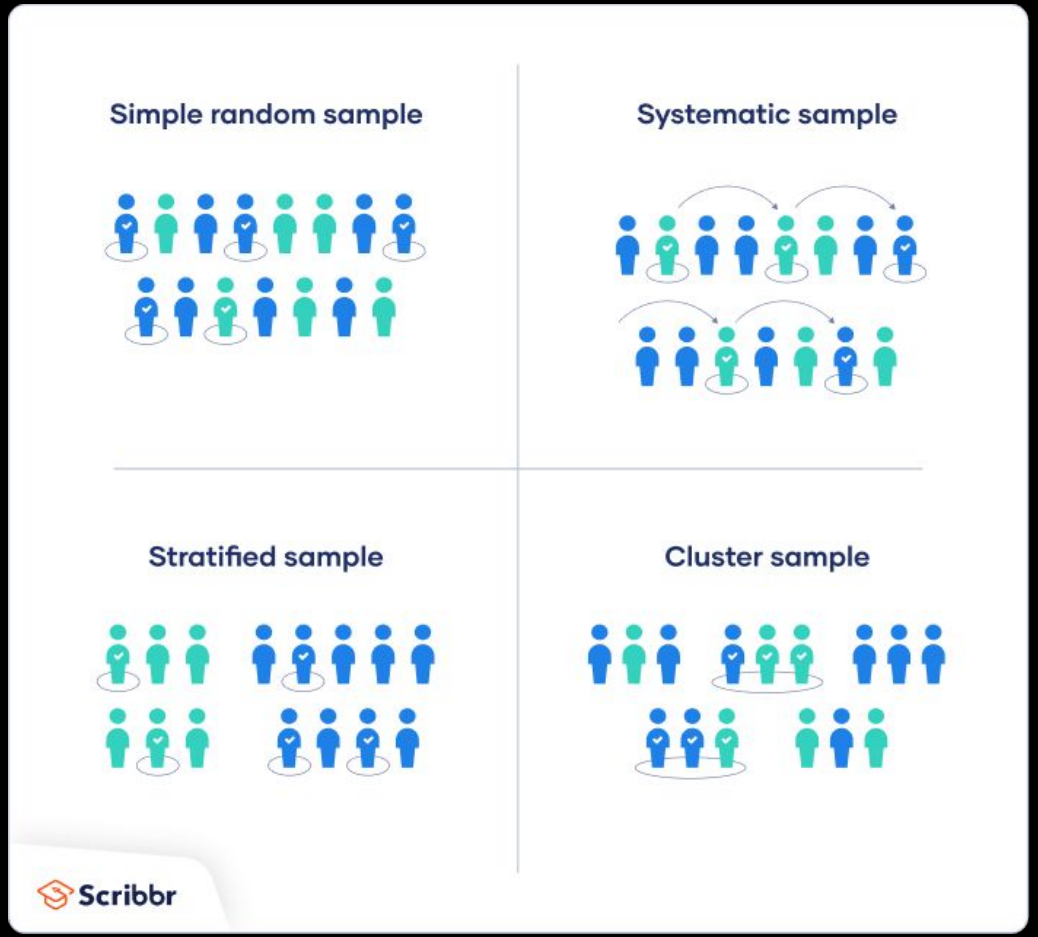
The Crash Test Bias: How Male-Focused Testing Puts Female Drivers at Risk

https://www.consumerreports.org/car-safety/crash-test-bias-how-male-focused-testing-puts-female-drivers-at-risk/

1.2.6 Sampling Methods

Probability Sampling :Random selection allowing strong statistical inferences about the population

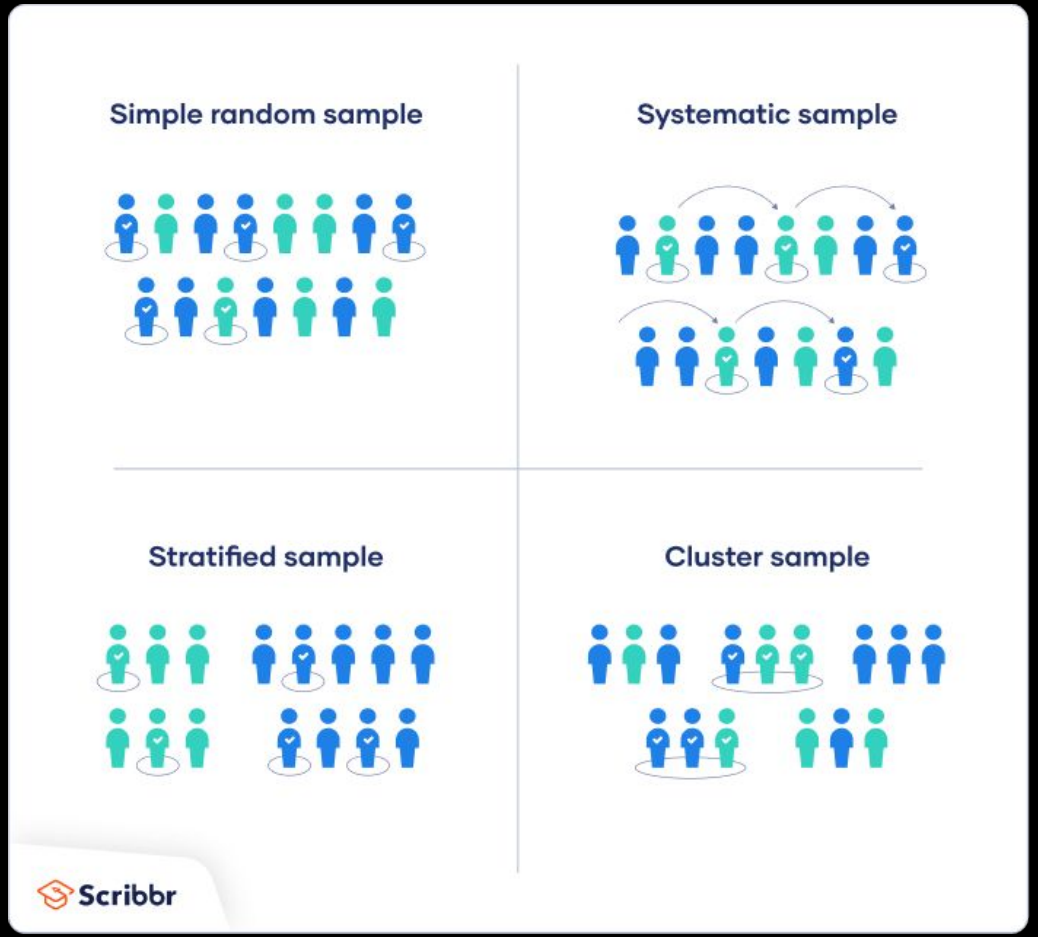
Non-Probability Sampling: Based on convenience or other criteria to easily collect data (but no random sampling)



1.2.6.1 Types of Probability Sampling

Simple Random Sample:

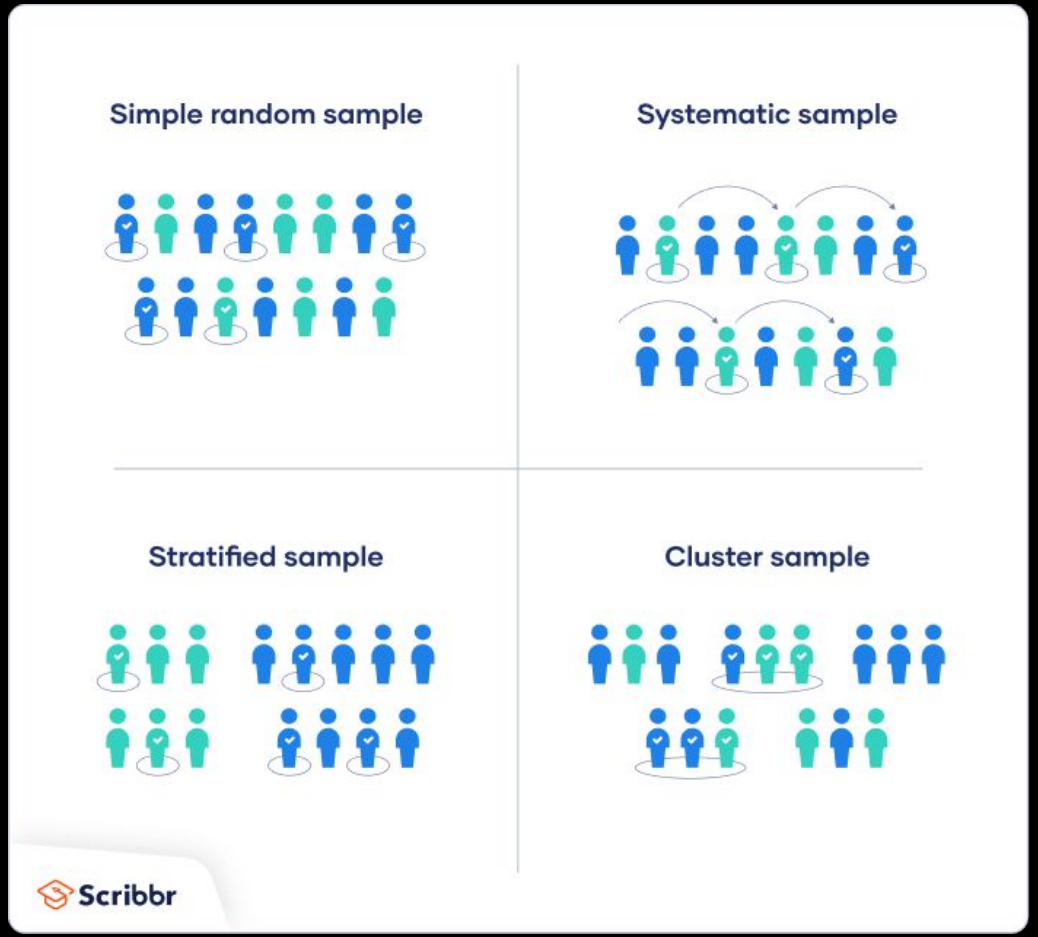
Each member of the population has the same chance of being selected (i.e. uniform over the population)



Types of Probability Sampling

Systematic Sample Select:

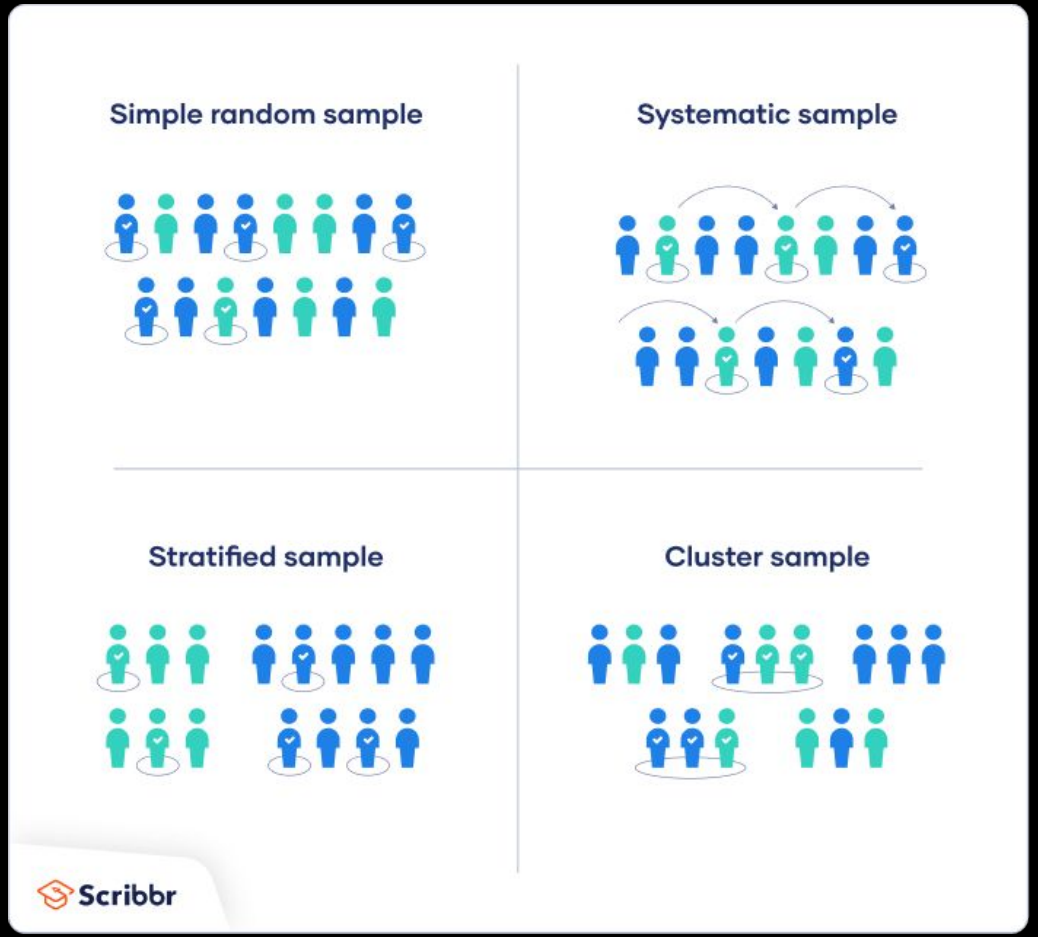
members of population at a regular interval, determined in advance



Types of Probability Sampling

Stratified Sample Divide : population into homogeneous subpopulations (strata).

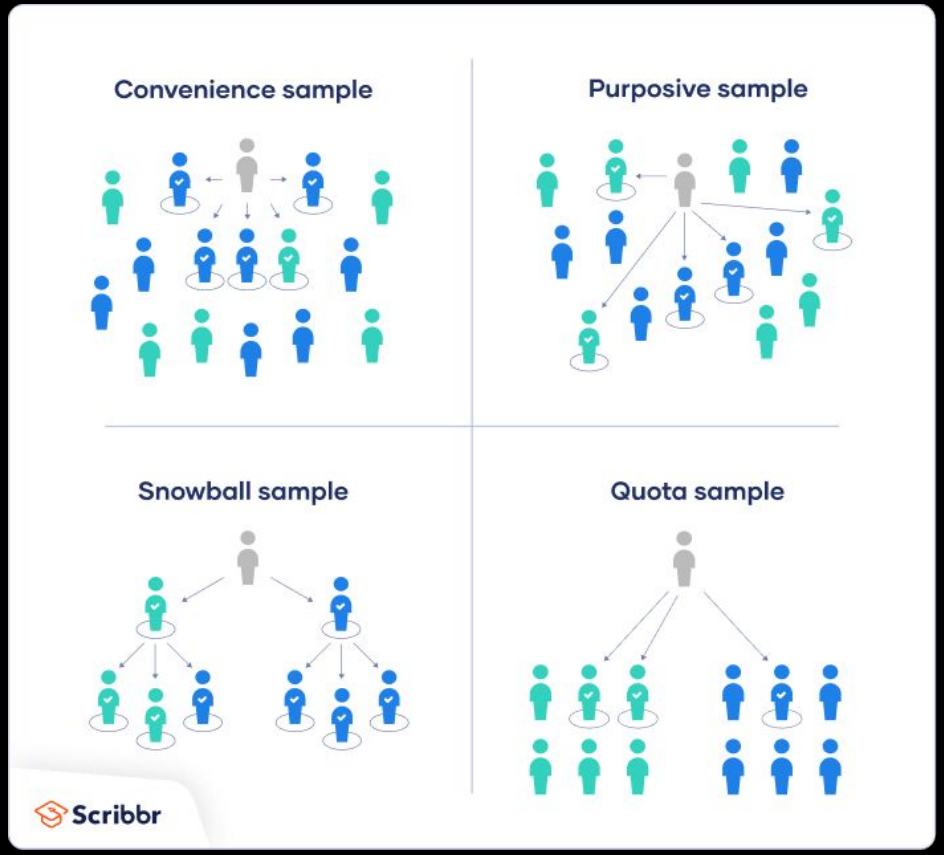
Probability sample the strata



Types of Probability Sampling

Cluster Sample Divide:

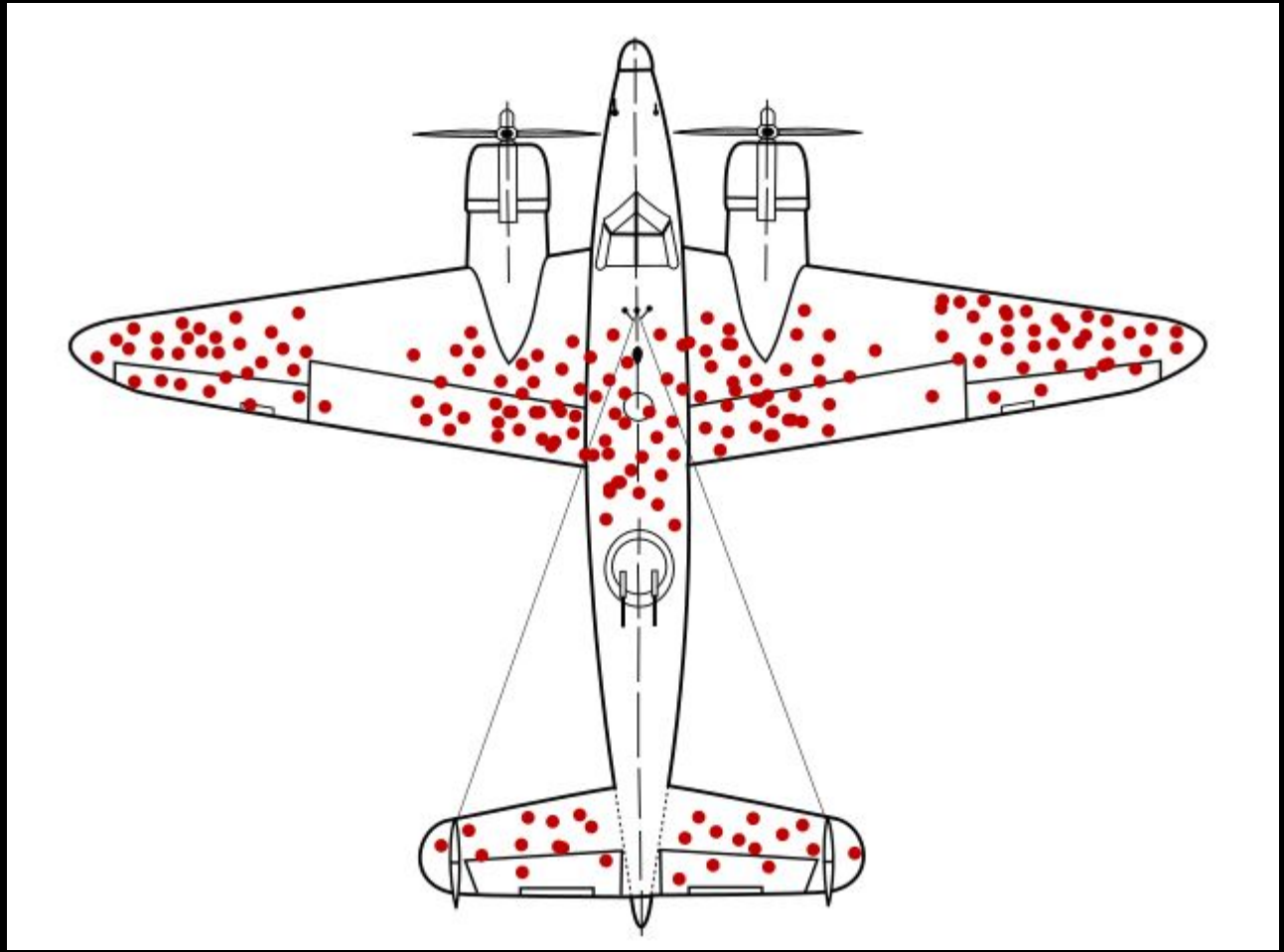
population into subgroups (clusters). Randomly select entire clusters.

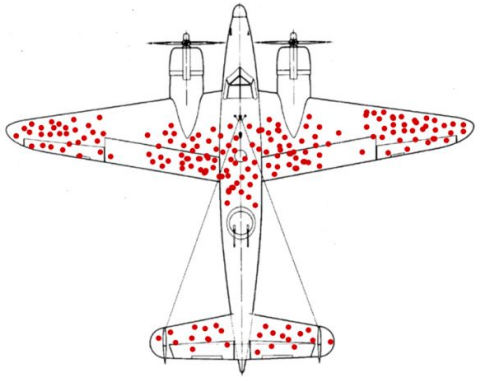


1.2.6.2 Types of Non-probability Sampling

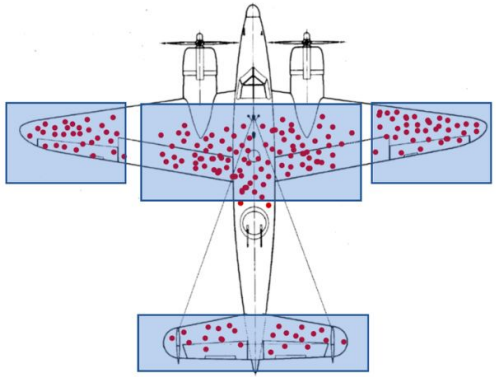
1.2.7 Common Types of Sampling Bias

- Self-selection
- Exclusion
- Survivorship

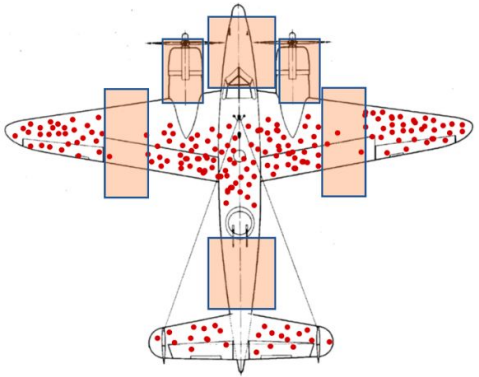




Our data if only from returning flights. Here we is a visualization of the places that bullet holes were observed.



And initial guess at how to fix this might be to apply additional armor platting to the parts of the plane with the most holes...



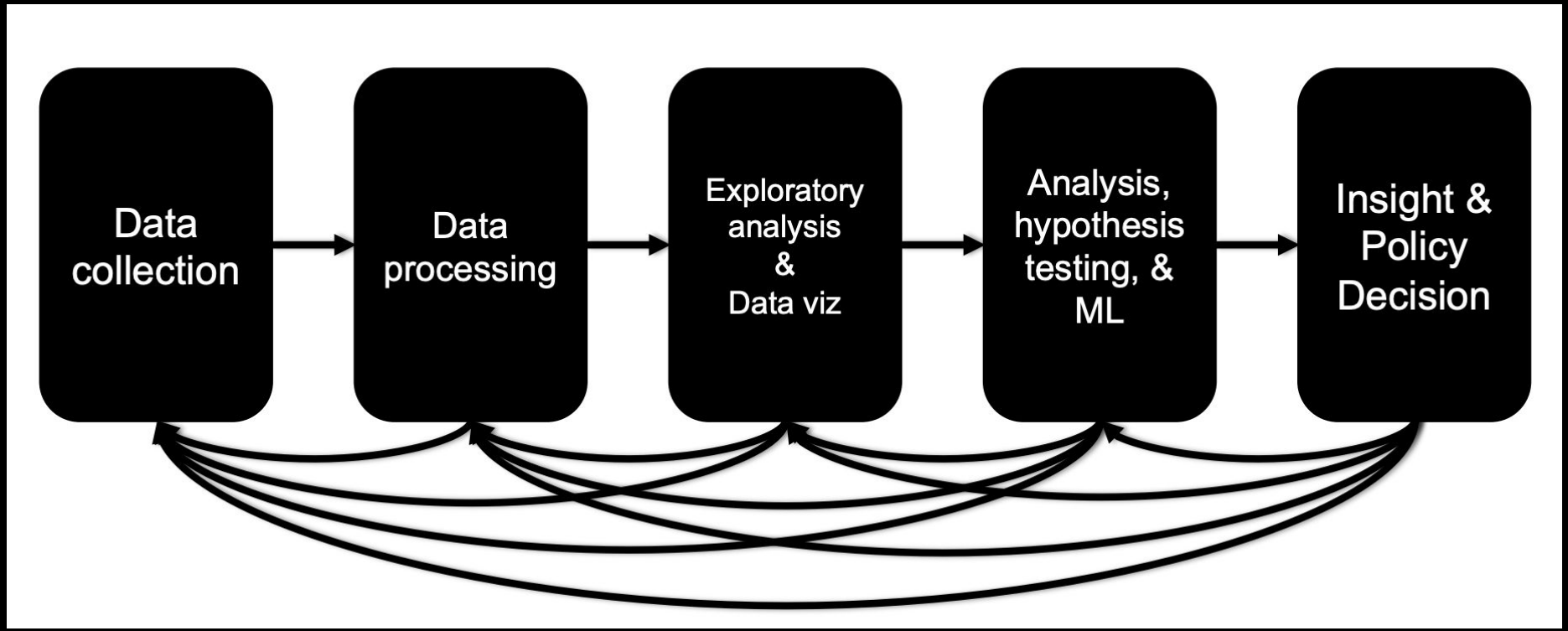
.... However this is where planes that *returned* had bullet holes. The planes we want to protect are the ones that did *not* return, so we should place armor there.

Source : https://www.countbayesie.com/blog/2020/11/5/survivorship-bias-in-house-hunting-a-practical-modeling-example-using-jax

Connecting to Data Science Process

Lecture content/topics in this section is derived from Prof.John P Dickerson’s lectures in CMSC641

Data Science Process



Credit : Prof.John P Dickerson’s lectures in CMSC641

How to collect data?

- Direct download and load from local storage
- Generate locally via downloaded code (e.g., simulation)
- Query data from a database.
- Query an API from the intra/internet
- Scrape data from a webpage

How to collect data?

- Direct download and load from local storage
- Generate locally via downloaded code (e.g., simulation)
- Query data from a database.
- Query an API from the intra/internet
- Scrape data from a webpage

Databases

- Organized collection of structured information, or data
- Database Management Systems (DBMS) are database management tools, a uniform technology that helps businesses optimize, manage, store, and retrieve data

https://www.oracle.com/database/what-is-database/
https://improvado.io/blog/top-25-best-database-management-software

Types of Database

- Relational database:
  stores information in tables. Often, these tables have shared information between them, causing a relationship to form between tables.

- Non-relational database (NoSQL (Not Only SQL)) :
  - Any kind of database that doesn't use the tables, fields, and columns structured data concept from relational databases.
  - They look more like JSON

https://www.mongodb.com/compare/relational-vs-non-relational-databases


MySQL


MongoDB


PostgreSQL


Oracle Database


Microsoft SQL Server


SQLite


Microsoft Access


IBM Db2


MariaDB


Redis


Amazon RDS


Airtable


Apache Cassandra


Neo4j


FileMaker


Google Cloud


DBeaver


Amazon DynamoDB


Informix


Oracle Corporation


ArangoDB


OrientDB

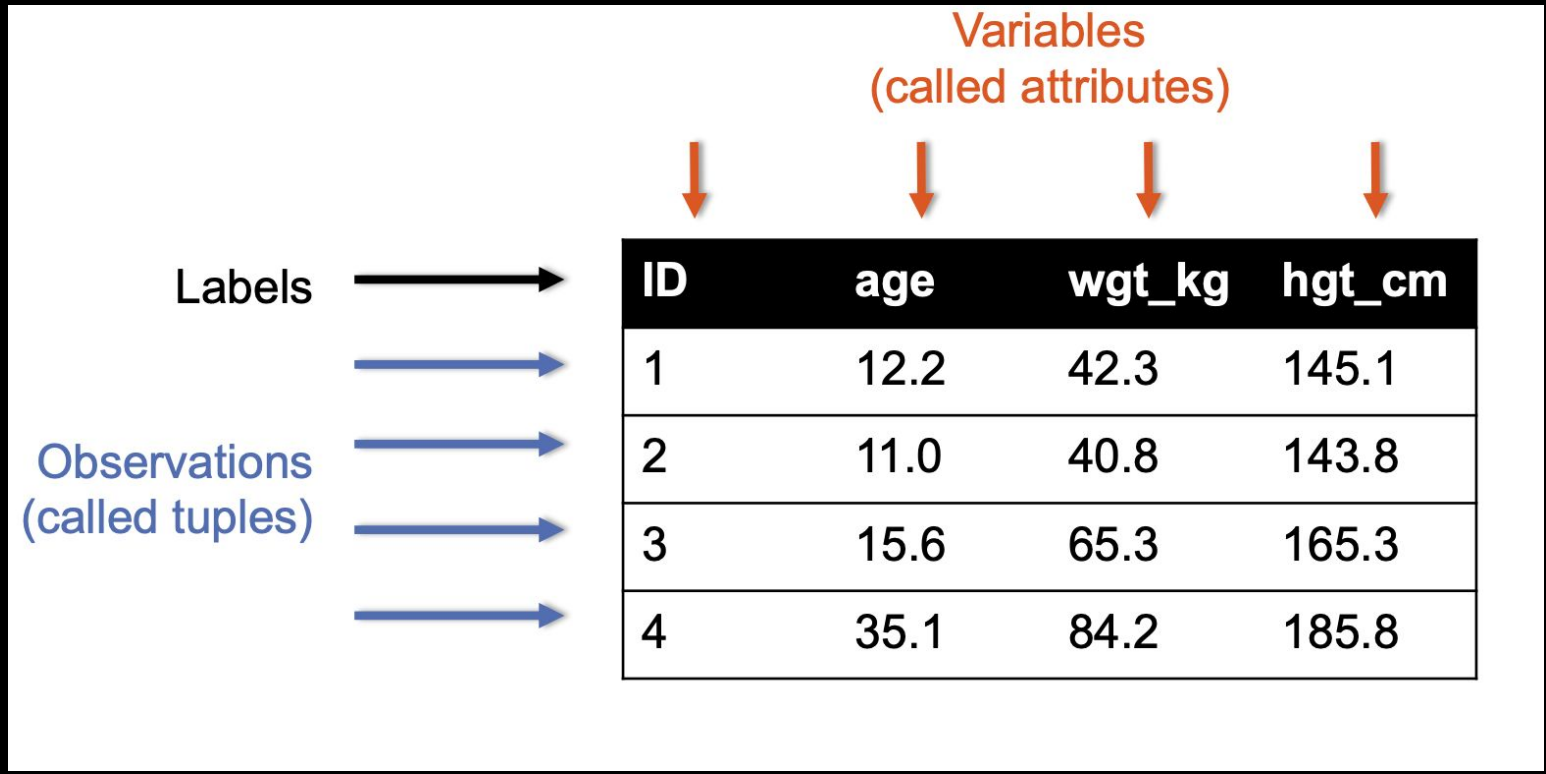

Kintone


DbVisualizer

SQL

Language used by nearly all relational databases to query, manipulate, and define data, and to provide access control.

Relation



ID	age	wgt_kg	hgt_cm
1	12.2	42.3	145.1
2	11.0	40.8	143.8
3	15.6	65.3	165.3
4	35.1	84.2	185.8

Credit : Prof.John P Dickerson's lectures in CMSC641

PRIMARY KEYS

ID	age	wgt_kg	hgt_cm	nat_id
1	12.2	42.3	145.1	1
2	11.0	40.8	143.8	1
3	15.6	65.3	165.3	2
4	35.1	84.2	185.8	1
5	18.1	62.2	176.2	3
6	19.6	82.1	180.1	1

ID	Nationality
1	USA
2	Canada
3	Mexico

The primary key is a unique identifier for every tuple in a relation

- **Each tuple has exactly one primary key**

Credit : Prof.John P Dickerson's lectures in CMSC641

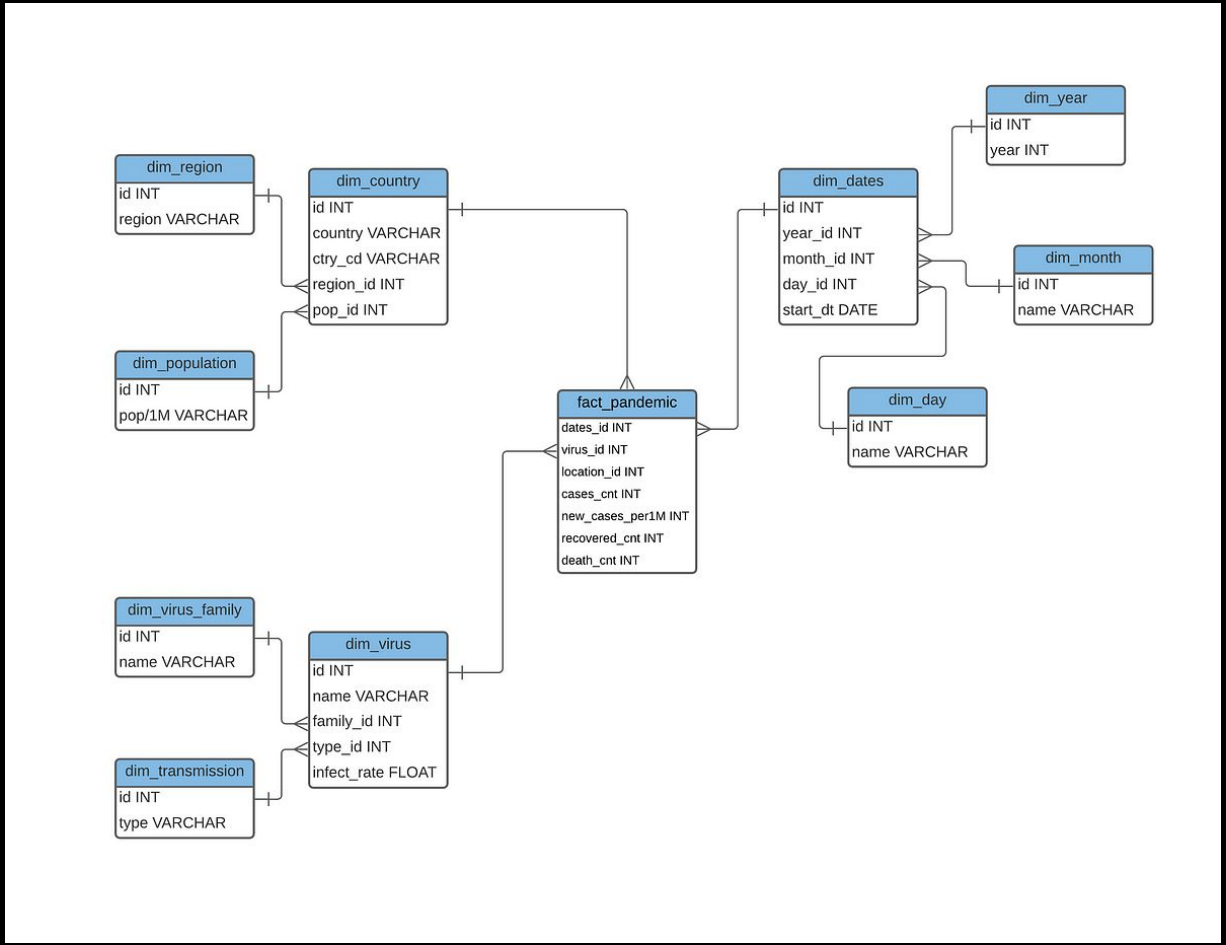
FOREIGN KEYS

ID	age	wgt_kg	hgt_cm	nat_id
1	12.2	42.3	145.1	1
2	11.0	40.8	143.8	1
3	15.6	65.3	165.3	2
4	35.1	84.2	185.8	1
5	18.1	62.2	176.2	3
6	19.6	82.1	180.1	1

ID	Nationality
1	USA
2	Canada
3	Mexico

Foreign keys are attributes (columns) that point to a different table's primary key

Credit : Prof.John P Dickerson's lectures in CMSC641



Credit : https://towardsdatascience.com/designing-your-database-schema-best-practices-31843dc78a8d

Types of Relationships

- One-to-one
- One-to-one-or-none
- One-to-many and many-to-one
- Many-to-many

```sql
CREATE TABLE users (
  id INTEGER PRIMARY KEY NOT NULL AUTO_INCREMENT,
  email VARCHAR(255) NOT NULL,
  `password` VARCHAR(255) NOT NULL,
  phone_number VARCHAR(15),
  created TIMESTAMP NOT NULL DEFAULT NOW()
);
```

Credit : https://www.sammeechward.com/a-quick-guide-to-creating-tables

SQL Tutorial

SQL HOME
SQL Intro
SQL Syntax
SQL Select
SQL Select Distinct
SQL Where
SQL And, Or, Not
SQL Order By
SQL Insert Into
SQL Null Values
SQL Update
SQL Delete
SQL Select Top
SQL Min and Max
SQL Count, Avg, Sum
SQL Like
SQL Wildcards
SQL In
SQL Between
SQL Aliases
SQL Joins
SQL Inner Join
SQL Left Join
SQL Right Join

SQL Right Join
SQL Full Join
SQL Self Join
SQL Union
SQL Group By
SQL Having
SQL Exists
SQL Any, All
SQL Select Into
SQL Insert Into Select
SQL Case
SQL Null Functions
SQL Stored Procedures
SQL Comments
SQL Operators

SQL Database

SQL Create DB
SQL Drop DB
SQL Backup DB
SQL Create Table
SQL Drop Table
SQL Alter Table
SQL Constraints
SQL Not Null
SQL Unique

SQL Primary Key
SQL Foreign Key
SQL Check
SQL Default
SQL Index
SQL Auto Increment
SQL Dates
SQL Views
SQL Injection
SQL Hosting
SQL Data Types

SQL References

SQL Keywords
MySQL Functions
SQL Server Functions
MS Access Functions
SQL Quick Ref

Tutorial:
https://www.w3schools.com/sql/

Examples

- SELECT PRODUCT_NAME, PRICE FROM PRODUCT WHERE PRODUCT _ID = 23;

- SELECT MIN(Price) AS SmallestPrice FROM Products;

What about non-relational databases?

```
db.product.find({"_id": 23}, {productName: 1, price: 1})
```

Usage in Python

```
import sqlite3
con = sqlite3.connect("tutorial.db")
```

```
cur = con.cursor()
```

```
cur.execute("CREATE TABLE movie(title, year, score)")
```

```
>>> res = cur.execute("SELECT name FROM sqlite_master")
>>> res.fetchone()
```

```
>>> for row in cur.execute("SELECT year, title FROM movie ORDER BY year"):
...     print(row)
(1971, 'And Now for Something Completely Different')
(1975, 'Monty Python and the Holy Grail')
(1979, "Monty Python's Life of Brian")
(1982, 'Monty Python Live at the Hollywood Bowl')
(1983, "Monty Python's The Meaning of Life")
```

Credit : https://docs.python.org/3/library/sqlite3.html

this kinda feels like pandas …

Credit : Prof.John P Dickerson’s lectures in CMSC641

this kinda feels like pandas …

Rule of thumb: do heavy, rough lifting at the relational DB level, then fine-grained slicing and dicing and viz with pandas

Credit : Prof.John P Dickerson’s lectures in CMSC641

Not going into these:

Except for one topic : Joins

JOINING DATA

A join operation merges two or more tables into a single relation. Different ways of doing this:

- Inner
- Left
- Right
- Full Outer

Join operations are done on columns that explicitly link the tables together

Credit : Prof.John P Dickerson's lectures in CMSC641

INNER JOINS

id	name
1	Megabyte
2	Meowly Cyrus
3	Fuzz Aldrin
4	Chairman Meow
5	Anderson Pooper
6	Gigabyte

`cats`

cat_id	last_visit
1	02-16-2017
2	02-14-2017
5	02-03-2017

`visits`

Inner join returns merged rows that share the same value in the column they are being joined on (`id` and `cat_id`).

id	name	last_visit
1	Megabyte	02-16-2017
2	Meowly Cyrus	02-14-2017
5	Anderson Pooper	02-03-2017

Credit : Prof.John P Dickerson’s lectures in CMSC641

LEFT JOINS

Inner joins are the most common type of joins (get results that appear in both tables)

Left joins: all the results from the left table, only some matching results from the right table

Left join (`cats`, `visits`) on (`id`, `cat_id`) ??????????

id	name	last_visit
1	Megabyte	02-16-2017
2	Meowly Cyrus	02-14-2017
3	Fuzz Aldrin	NULL
4	Chairman Meow	NULL
5	Anderson Pooper	02-03-2017
6	Gigabyte	NULL

Credit : Prof.John P Dickerson's lectures in CMSC641

RIGHT JOINS

Take a guess!

Right join
(cats, visits)
on
(id, cat_id)
??????????

id	name
1	Megabyte
2	Meowly Cyrus
3	Fuzz Aldrin
4	Chairman Meow
5	Anderson Pooper
6	Gigabyte

cats

cat_id	last_visit
1	02-16-2017
2	02-14-2017
5	02-03-2017
7	02-19-2017
12	02-21-2017

visits

id	name	last_visit
1	Megabyte	02-16-2017
2	Meowly Cyrus	02-14-2017
5	Anderson Pooper	02-03-2017
7	NULL	02-19-2017
12	NULL	02-21-2017

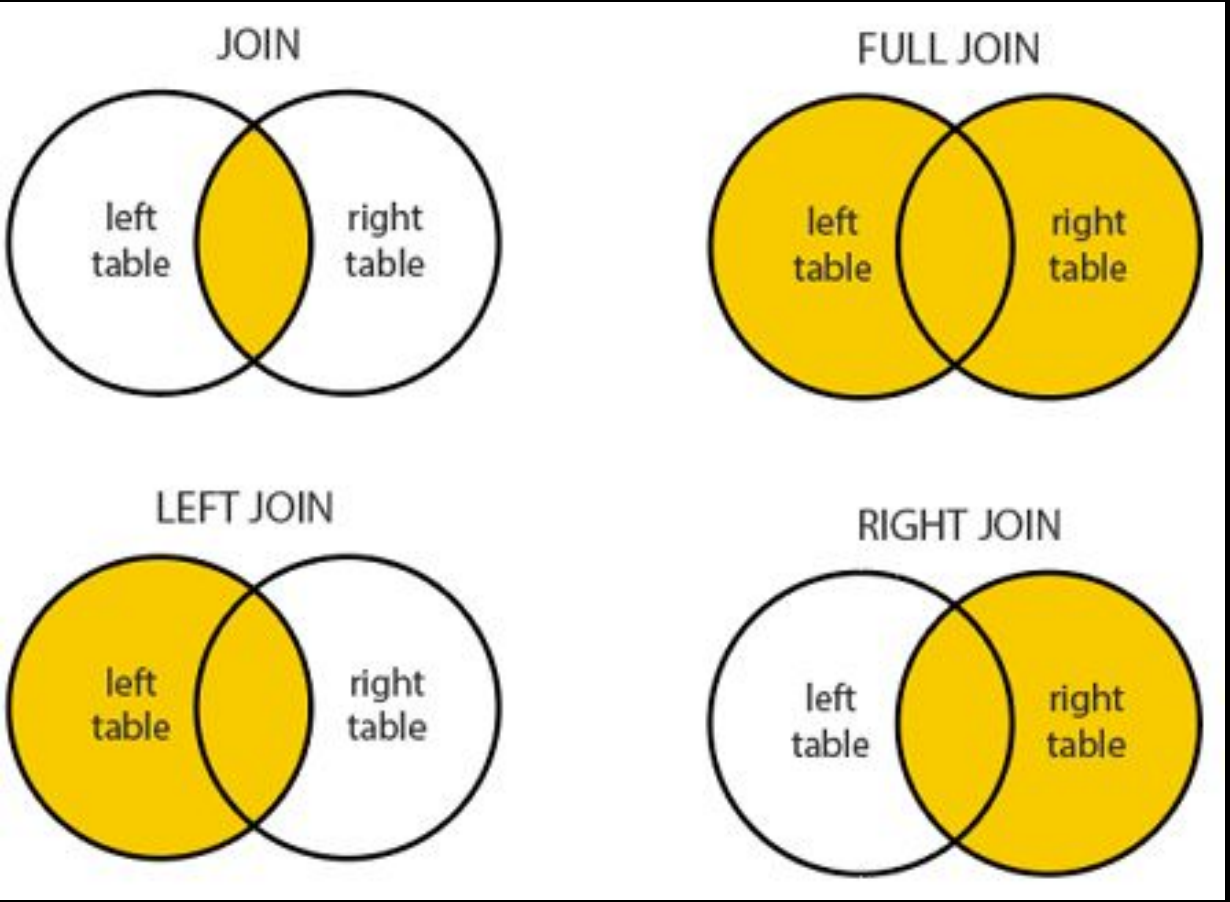
Credit : Prof.John P Dickerson's lectures in CMSC641

FULL OUTER JOIN

Combines the left and the right join ??????????

id	name	last_visit
1	Megabyte	02-16-2017
2	Meowly Cyrus	02-14-2017
3	Fuzz Aldrin	NULL
4	Chairman Meow	NULL
5	Anderson Pooper	02-03-2017
6	Gigabyte	NULL
7	NULL	02-19-2017
12	NULL	02-21-2017

Credit : Prof.John P Dickerson's lectures in CMSC641



Credit : https://www.dofactory.com/sql/join

We will be learning later how to do these in pandas in an upcoming lecture

In this lecture,

- Research Design for Statistical Analysis
  - Causation versus Correlation
  - Sampling
- Revisit the Data Science Process
- Data Collection
  - Till SQL
- Data Processing