

# CS/ENGR M148 L2: Data Cleaning and Exploratory Data Analysis

Sandra Batista

**Any questions about the syllabus?**

**This week we will begin working on projects during discussion...**

## Course Policies

---

### **Collaboration Policy:**

You may collaborate on ungraded labs and team projects. You may not collaborate on problems sets, quizzes or exams.

### **Generative AI usage Policy:**

You may use generative AI tools on ungraded labs and team projects. You may not use generative AI tools on problems sets, quizzes or exams.

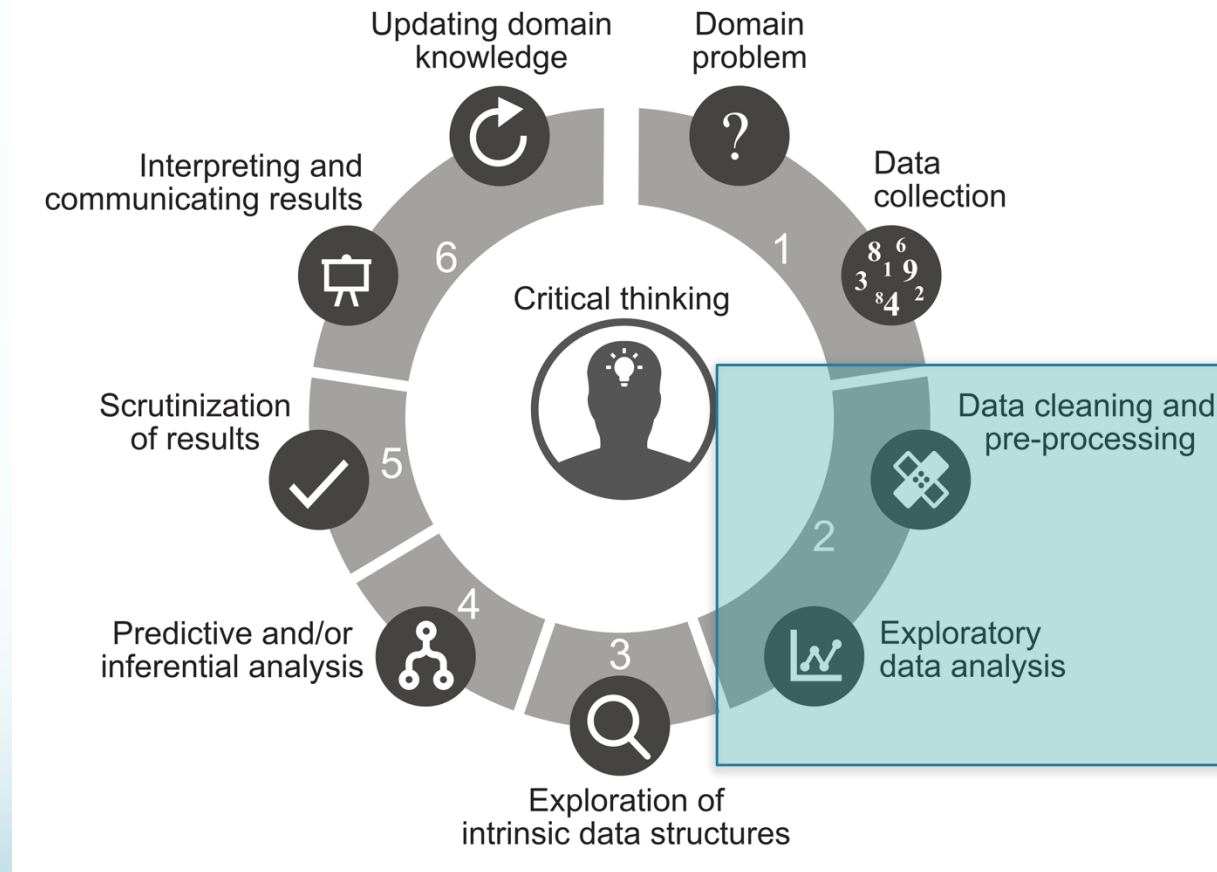
**Late Work Policy:** No late work accepted. However, we will work to create fair accommodations for extenuating circumstances.

# Projects

---

1. Projects will be graded on how well they demonstrate mastery of the methods taught in class and discussions.
2. You may choose your own data set or a data set supported by the course staff.
3. Team contract 5% - This week during discussion. A sample contract will be made available and TAs will reveal data sets.  
Team contracts due by 11:59 pm PT on 10/4/24
4. Project discussion check-ins: 30%, 6x5%
5. Final project code: 25%
6. Final project report: 40%

# Data Science Life Cycle (DSLCL)



*Very critical portion of any project and requires significant portion of time and care*

[Yu, Barter 2024]

# DSLCC Step 2: Data Cleaning and Exploratory Data Analysis (EDA)

**Data cleaning** is the process of modifying a dataset so that it is tidy, appropriately formatted, and unambiguous.

**Preprocessing** is the process of modifying a dataset so that it satisfies the formatting requirements of a particular analysis or algorithm.

This stage deals with possible errors and missing values.

Later we'll deal with transforming data for algorithms by standardizing variables, transforming variables, one hot encoding and **featurization** or constructing new features from data.

# Today's Learning Objectives

Students will be able to:

- ✗ Identify **tabular data**
- ✗ Perform **data cleaning and pre-processing** on a real data set
- ✗ Conduct preliminary **exploratory data analysis** on a real data set

# Tabular Data

Features, variables, attributes, or covariates are columns

Dimension is number of columns.

Table 2.1: US government research and development budget and spending (reported in millions USD). The columns correspond to the year, the budget, the total spending, the spending on climate, the spending on energy, and the political party in office

Year	Budget	Total	Climate	Energy	Party
2000	142,299	1,789,000	2,312	13,350	Democrat
2001	153,197	1,862,800	2,313	14,511	Republican
2002	170,354	2,010,900	2,195	14,718	Republican
2003	192,010	2,159,900	2,689	15,043	Republican
2004	199,104	2,292,800	2,484	15,343	Republican
2005	200,099	2,472,000	2,284	14,717	Republican
2006	199,429	2,655,000	2,004	14,194	Republican
2007	201,827	2,728,700	2,044	14,656	Republican
2008	200,857	2,982,500	2,069	15,298	Republican
2009	201,275	3,517,700	2,346	16,492	Democrat



# Tabular Data

## Feature data types:

Numeric, categorical, dates and times,


Structured (short) text, Unstructured (long) text

Table 2.1: US government research and development budget and spending (reported in millions USD). The columns correspond to the year, the budget, the total spending, the spending on climate, the spending on energy, and the political party in office

Year	Budget	Total	Climate	Energy	Party
2000	142,299	1,789,000	2,312	13,350	Democrat
2001	153,197	1,862,800	2,313	14,511	Republican
2002	170,354	2,010,900	2,195	14,718	Republican
2003	192,010	2,159,900	2,689	15,043	Republican
2004	199,104	2,292,800	2,484	15,343	Republican
2005	200,099	2,472,000	2,284	14,717	Republican
2006	199,429	2,655,000	2,004	14,194	Republican
2007	201,827	2,728,700	2,044	14,656	Republican
2008	200,857	2,982,500	2,069	15,298	Republican
2009	201,275	3,517,700	2,346	16,492	Democrat

# Tabular Data

Each row is an **observation**, **observational unit**, **data unit**, or **data point**

Table 2.1: US government research and development budget and spending (reported in millions USD). The columns correspond to the year, the budget, the total spending, the spending on climate, the spending on energy, and the political party in office 

Year	Budget	Total	Climate	Energy	Party
2000	142,299	1,789,000	2,312	13,350	Democrat
2001	153,197	1,862,800	2,313	14,511	Republican
2002	170,354	2,010,900	2,195	14,718	Republican
2003	192,010	2,159,900	2,689	15,043	Republican
2004	199,104	2,292,800	2,484	15,343	Republican
2005	200,099	2,472,000	2,284	14,717	Republican
2006	199,429	2,655,000	2,004	14,194	Republican
2007	201,827	2,728,700	2,044	14,656	Republican
2008	200,857	2,982,500	2,069	15,298	Republican
2009	201,275	3,517,700	2,346	16,492	Democrat

# Rectangular Data

We often prefer **rectangular data** for data analysis. This is a type of tabular data.

- **Regular structures** are easy to manipulate and analyze
- A big part of **data cleaning** is about transforming data to be more rectangular

Two kinds of rectangular data: **Tables** and **Matrices**.

**Fields/Attributes/  
Features/Columns**

**Records/Rows**


**Tables** (a.k.a. *DataFrames* in R/Python and relations in SQL)

- Named columns with different types
- Manipulated using data transformation languages (map, filter, group by, join, ...)

**Matrices**

- Numeric data of the same type (float, int, etc.)
- Manipulated using linear algebra

What are the differences?  
Why would you use one over the other?

# CSV: Comma Separate Values

Publicly available survey data from the Global Observatory on Donation and Transplantation

<https://drive.google.com/file/d/1T-AsACpKRiFQEIM1PKkENLQPhS00n4ew/view?usp=sharing>

CSV is a very common **tabular file format**.

- × **Records** (rows) are delimited by a newline: `'\n'`, `"\r\n"`
- × **Fields** (columns) are delimited by commas: `' , '`

Pandas: `pd.read_csv(header=...)`  
*Fields/Attributes/Features/Columns*

Records/Rows		Region	Country	...
	0	Europe	Andora	...
	1	Eastern	United Arab Emirates	...

# Variables Are Columns

What does each **column** represent?

A **variable** is a **measurement** of a particular concept.

It has two common properties:

	Region	Country	...
0	Europe	Andora	...
1	Eastern	United Arab Emirates	...

- **Datatype/Storage type:**

How each variable value is stored in memory.

`df[colname].dtype`

- integer, floating point, boolean, object (string-like), etc.

Affects which pandas functions you use.

- **Variable type/Feature type:**

Conceptualized measurement of information (and therefore what values it can take on).

- Use expert knowledge
- Explore data itself
- Consult data codebook (if it exists).

Affects how you visualize and interpret the data.

# Data Formats

---

Are the data in a standard format or encoding?

- ✗ Tabular data: CSV, TSV, Excel, SQL
- ✗ Nested data: JSON or XML

Are the data organized in **records** or nested?

- ✗ Can we define records by parsing the data?
- ✗ Can we reasonably un-nest the data?

Does the data reference other data?

- ✗ Can we join/merge the data?
- ✗ Do we need to merge data?

What are the **fields** in each record?

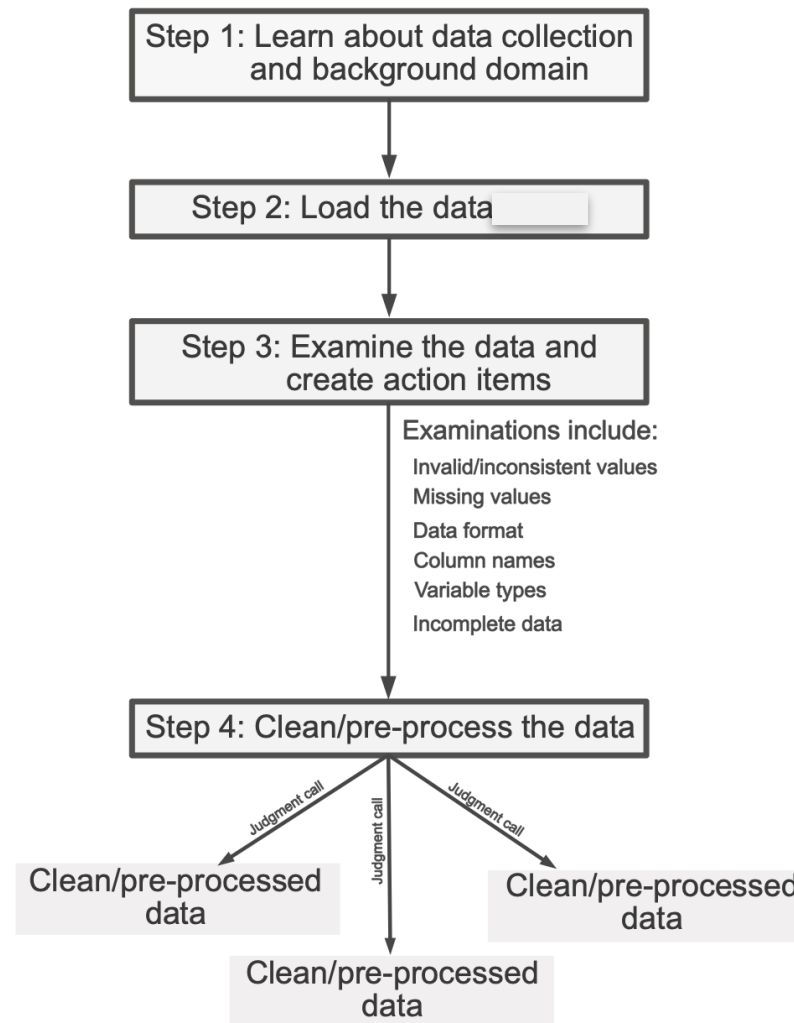
- ✗ How are they encoded? (e.g., strings, numbers, binary, dates ...)
- ✗ What is the type of the data?

# Today's Learning Objectives

Students will be able to:

- ✓ Identify **tabular data**
- ✗ Perform **data cleaning and pre-processing** on a real data set
- ✗ Conduct preliminary **exploratory data analysis** on a real data set

# Data Cleaning Procedure





# 1. Learn about data collection and domain

- What does each variable measure? What real-world quantity is each variable supposed to be capturing?
- How was the data collected? Mentally visualize the real-world data collection procedure. How was each variable physically measured?
- What are the observational units? The observational units correspond to the entities on which the measurements are collected.
- Is the data relevant to my project? Take a moment to verify the data that you have helps answer the question that you are asking.
- What questions do I have, and what assumptions am I making? Write down and answer your questions and assumptions.

## 2. Load your data

- How your data is being stored? .csv file? (.txt) file? A Microsoft Excel spreadsheet with multiple pages? A database in the cloud? Identify how to load the particular file type into your coding environment (e.g., what function to use).
- If your data contains multiple tables, is there a key variable that connects them? For more complex data, you may have several tables with the same set of observational units. Identify whether the tables contain any matching (key) variables to merge observational unit's measurements into a single table.
- What parts of the data are relevant to the question being asked? Is all the data relevant to your question? If not, consider filtering just to the portion of your data that is relevant to your question.

## 2. Checking Data Loading

- Look at subsets of rows of the data
- Verify if the dimension is what you expect
- **Dimension**, in this context, is the number of rows and columns

# Your turn: Global Donation Study

Please get the Jupyter notebook

Go to:

<https://colab.research.google.com/drive/1hacTP78YscYMbTbFVJrLj3oOazDwq0el?usp=sharing>

We'll learn about the data and load it...

Save a copy to your Google Drive and keep notes there...

# 1. Recap on the Global Donation Data

- What does each variable measure? Included in the dictionary in the notebook
- How was the data collected? Documented here: <https://www.transplant-observatory.org/methodology/>
- What are the observational units? For the organ donation data, the donor counts are reported every year for each country, so the observational units are the “country-year” combinations.
- Is the data relevant to my project? Yes we are seeing how donor rates increase per year per country.
- What questions do I have, and what assumptions am I making?
- We assumed every country would have data for every year but this was not true and we don't know why yet...

# 5. Examine Data and Create Action Items

Explore your data for the following:

1. **Invalid or inconsistent values.** Invalid values are usually impossible measurements. Inconsistent values might be identified when measurements disagree with others in the data set
2. **Improperly formatted missing values.** Often when a particular measurement is not available or is not properly entered, it is reported as “missing” (NA) in the data. Missing values should be explicitly formatted as NaN in Python
3. **Nonstandard data format.**
4. **Messy column names.** Your variable column names should be meaningful and clear to humans.
5. **Improper variable types.** Each variable should have an appropriate type (e.g., numeric, character, logical, date-time, etc.) based on measurement and interpretation.
6. **Incomplete data.** Data for which every observational unit appears exactly once (i.e., none are duplicated *and* none are missing from the data) is considered *complete*.

What is wrong  
with this  
data?

#	Country	Alcohol	Deaths	Heart	Liver
1	Australia	2.5	785	211	15.30000019
2	Austria	3.000000095	863	167	45.59999847
3	Belg/Lux	2.900000095	883	131	20.70000076
4	Canada	2.400000095	793	NA	16.39999962
5	Denmark	2.900000095	971	220	23.89999962
6	Finland	0.800000012	970	297	19
7	France	9.100000381	751	11	37.90000153
8	Iceland	-0.800000012	743	211	11.19999981
9	Ireland	0.699999988	1000	300	6.5
10	Israel	0.600000024	-834	183	13.69999981
11	Italy	27.900000095	775	107	42.20000076
12	Japan	1.5	680	36	23.20000076
13	Netherlands	1.799999952	773	167	9.199999809
14	New Zealand	1.899999976	916	266	7.699999809
15	Norway	0.0800000012	806	227	12.19999981
16	Spain	6.5	724	NA	NA
17	Sweden	1.600000024	743	207	11.19999981
18	Switzerland	5.800000191	693	115	20.29999924
19	UK	1.299999952	941	285	10.30000019
20	US	1.200000048	926	199	22.10000038
21	West Germany	2.700000048	861	172	36.70000076

[Shah, 2020]

# 3. Examine Data and Create Action Items

## To check for invalid or inconsistent values:

1. Inspect random rows of the data.
2. Check ranges of values for numeric data.
3. Create histograms for numeric data.
4. Check unique values of categorical variables.

## Action Items:

1. Leave alone.
2. Substitute valid correct value.
3. Replace with NaN
4. Convert values to common unit

[Yu, Barter 2024]



# 3. Examine Data and Create Action Items

## To identify missing values:

1. Check ranges of values for numeric data.
2. Create histograms for numeric data.
3. Check unique values of categorical variables.
4. Calculate proportions of missing values
5. Visualize pattern of missing values as with heatmap

## 3. Examine Data and Create Action Items

### Action Items for missing values:

1. Make sure marked NA
2. Replace known values and justify choice
3. Apply **imputation**. There are many methods for this such as

**Constant-value imputation:** Replacing all missing values with a plausible constant value, **Mean imputation:** Replacing all missing values in a column with the average value of the non-missing values in the column

**Forward/backward fill imputation:** in time-dependent data replace with previous or next data point

[Yu, Barter 2024]

## 3. Examine Data and Create Action Items

### **Action Items for missing values:**

4. Set missing threshold and drop values
5. Remove rows with missing values. Document and justify the decision.

## 5. Examine Data and Create Action Items

Check that the data set is tidy:

A **tidy dataset** satisfies the following criteria:

- Each *row* corresponds to a *single observational unit*.
- Each *column* corresponds to a *single type* of measurement.

## 5. Examine Data and Create Action Items

Check variable names and rename them to more meaningful names.

Check the type of variables in each column. Make sure that type matches the variable and values match column.

Check if you dataset is complete:

A **complete dataset** is one in which each observational unit is *explicitly* represented in the data.

## 5. Examine Data and Create Action Items

### Action Items for complete data set:

1. Choose subset that is complete. Document and justify this choice.
2. Add missing observational units to data and populate them as you would missing values.

## 4. Clean Your Data

**Make sure you save the raw data!!!**

1. Recommendation: Writing a data cleaning function whose input is the raw data and whose output is the clean data
2. In this case it would not be necessary to store cleaned data, but may be preferable for your project if data cleaning is time and computationally intensive.

# Your turn: Global Donation Study

Go to:

<https://colab.research.google.com/drive/1hacTP78YscYMbTbFVJrLj3oOazDwq0el?usp=sharing>

Let's look more at the steps 3 and 4 portion of the notebook...



# Today's Learning Objectives

Students will be able to:

- ✓ Identify **tabular data**
- ✓ Perform **data cleaning and pre-processing** on a real data set
- ✗ Conduct preliminary **exploratory data analysis** on a real data set

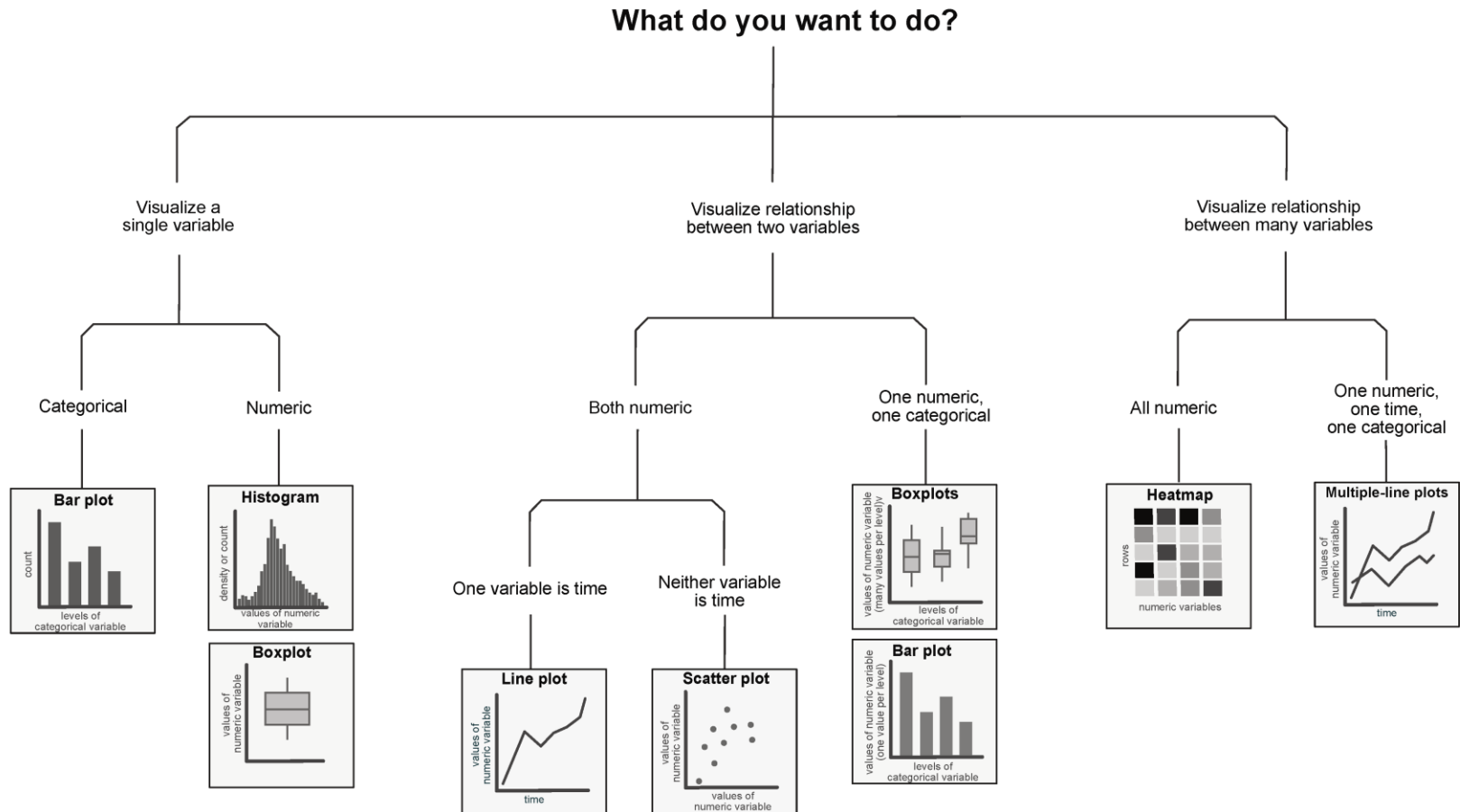
# What is exploratory data analysis (EDA)?

**“Exploratory data analysis (EDA):** the task of visually and numerically summarizing the patterns, trends, and relationships that a dataset contains in the context a domain problem” – Bin Yu

“Exploratory data analysis is an attitude, a state of flexibility, a willingness to look for those things that we believe are not there, as well as those that we believe to be there.”

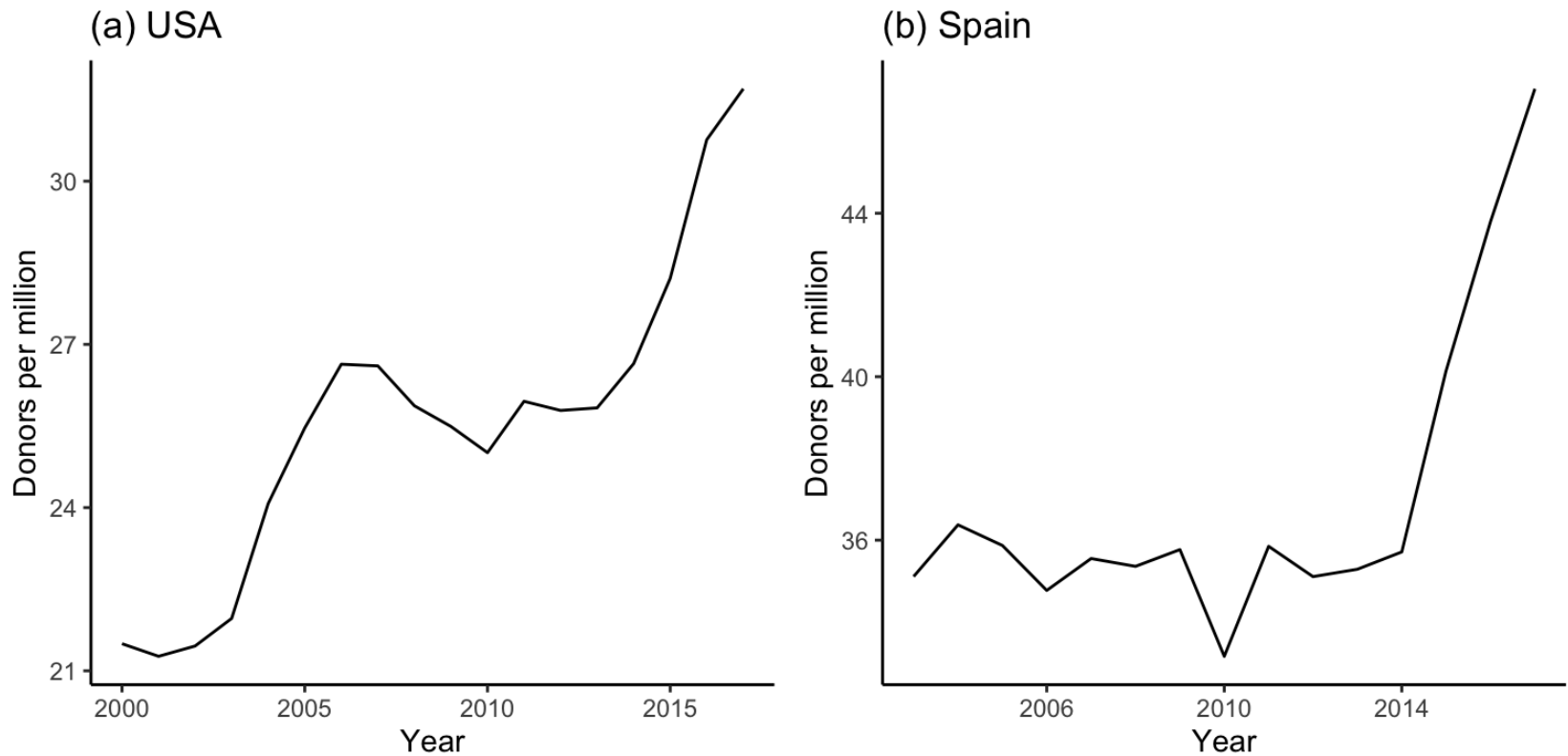
– John Tukey

# Question and Answer EDA Workflow



Make sure variables are **comparable** such as on similar scales

# Comparability



Make sure variables are **comparable**

# Exploratory vs Explanatory Data Analysis

**Exploratory data analysis (EDA)** creates numeric and visual summaries of the data for understanding patterns in it

**Explanatory data analysis** polishes the most informative exploratory tables and graphs to communicate them to external audiences

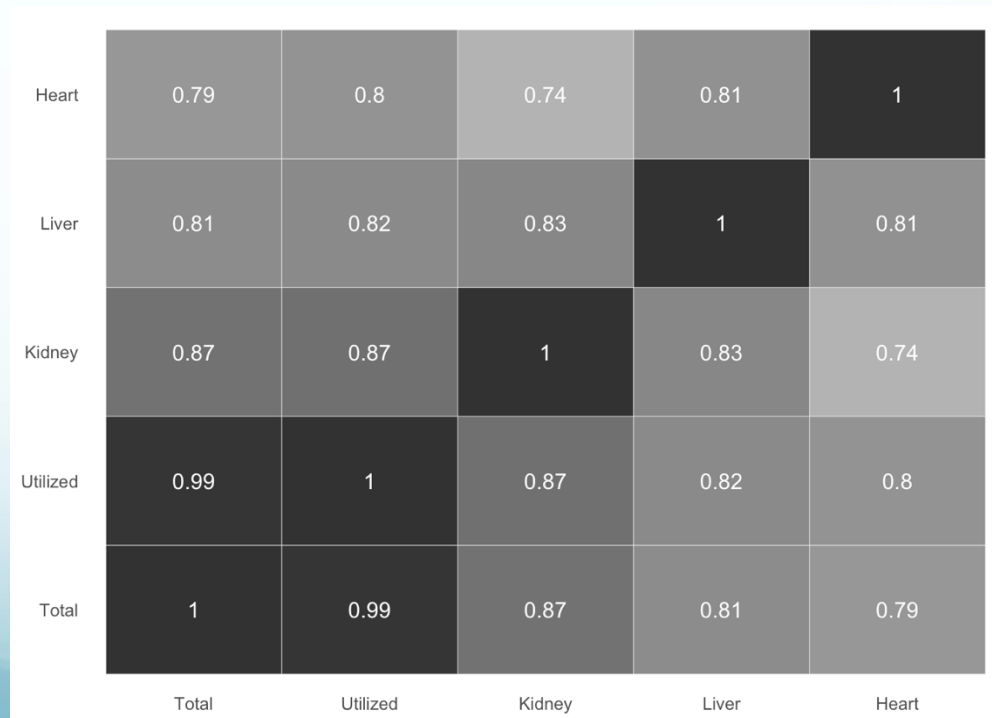


# Nonvisual EDA

1. Calculating summary statistics such as mean, median, variance, and standard deviations of variables
2. Caution: histogram of variable reveals more information
3. Calculate the covariance and correlation to check for linear relationships
4. Linear relationships can also be checked using scatterplots

# Handling correlations in EDA

1. Linear relationships can also be checked using scatterplots in addition to correlation
2. It is possible to check the correlation between every pair of variables in data and create a heatmap



# Global Donation Study EDA

**Question:** How do the *donation rates* change *over time* for each *country*?

To explore this, we will use

1. Line plots
2. Scatterplots
3. Correlations
4. Bar graphs
5. Heatmaps



# Your turn: Global Donation Study

Go to:

<https://colab.research.google.com/drive/1hacTP78YscYMbTbFVJrLj3oOazDwq0el?usp=sharing>

Let's look work on this preliminary EDA for this data set

# Today's Learning Objectives

Students will be able to:

- ✓ Identify **tabular data**
- ✓ Perform **data cleaning and pre-processing** on a real data set
- ✓ Conduct preliminary **exploratory data analysis** on a real data set

*Citations:*

*Yu, B., & Barter, R. L. (2024). Veridical data science: The practice of responsible data analysis and decision making. The MIT Press.*

*Shah. C. (2020) A hands-on introduction to data science. Cambridge University Press.*

*Data 100, Fall 2024, UC Berkeley.*