WEEK 5: DATA WRANGLING

Dr. Kai Li
School of Information Sciences
University of Tennessee, Knoxville
Spring 2025

Review of Week 4

- Data collection approaches
  - *API*
  - *Web scrapping*
- Data formats

- Any questions from last week?
  - *We will spend more time in the demonstration to answer technical questions.*

Requirements for final project

- I don’t have very strict requirements for the final project, because I believe you probably have very different knowledge on this topic.
- But I will ask you share your dataset and topic in one of the classes later (possibly W 9 or 10) to seek feedback from me and others.
- I am expecting a technical report (or a short research paper if you want) that:
  - *has a clear story with clear questions and sufficient results*
  - *a series of questions (around three and at least two inferential questions) that could lead to a meaningful understanding of your topic*
  - *has valid presentation*

Requirements for final project

- While Rpubs (https://rpubs.com/) is a good source to find many technical reports written in R Markdown, I would say most of the documents are lower than my expectation for our final project.
  - *Mostly because insufficient presentation, i.e., telling the story in a reader-friendly way*
- I will try to find some good examples in the next weeks and share with you!
  - *But feel free to share anything that you came across!*
  - *And I will also use the next assignment to give you some preparation!*

Overview of this week

- Techniques and procedures of data wrangling
- Tidyverse

- “Data scientists spend 90% of their time to clean the data.”

Concepts related to wrangling

- Data wrangling can be generally defined as the whole process of converting the raw data into a usable form.
- It can have many other names that may or may not be totally identical:
  - *Data cleaning*
  - *Data preprocessing*
- Data wrangling is both an art and a science.

The Six Steps in Data Wrangling

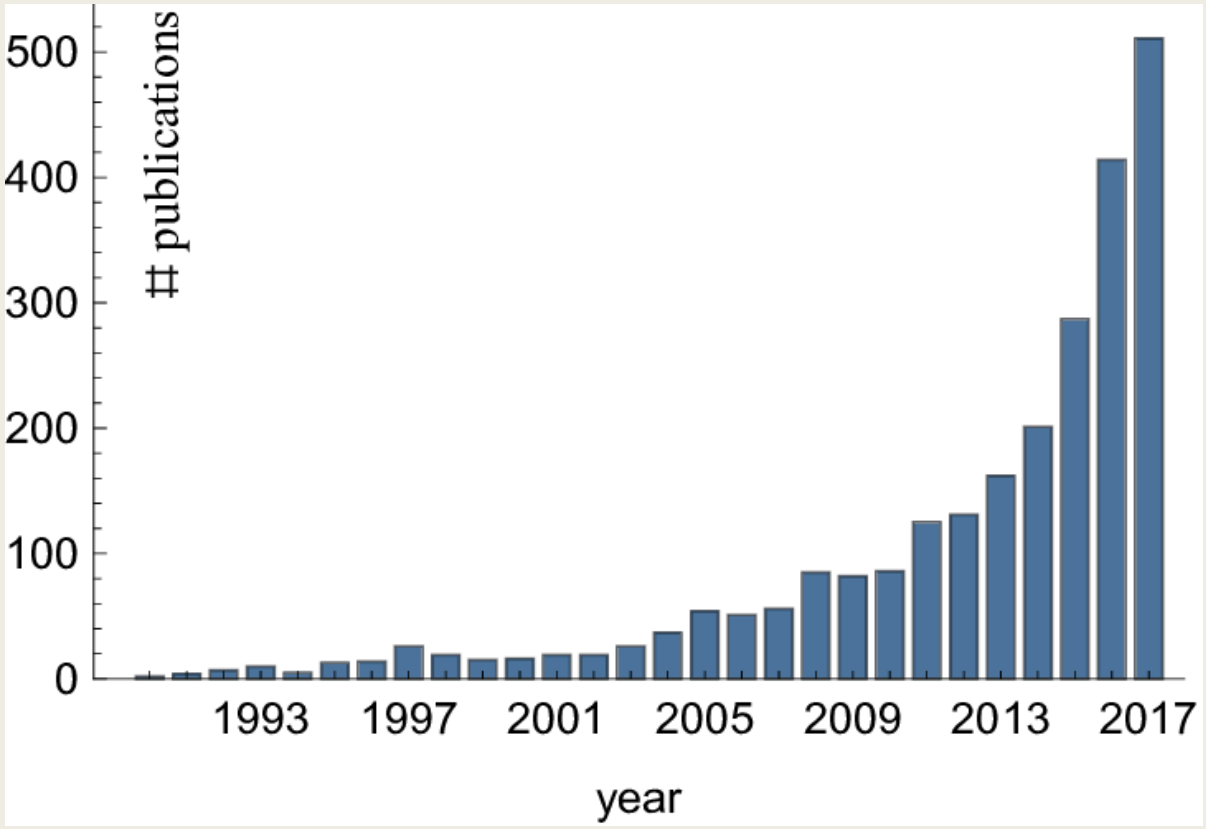
Discovering	Data exploration to familiarize with source data in preparation
Structuring data	To transform features to uniform formats, units, and scales
Cleaning data	To remove or replace missing and outlier data (!!)
Enriching data	To derive new features/measurements from existing data
Validating data	To check the dataset for internal consistency and accuracy
Publishing/sharing	To make dataset available to other researchers in a database

First, some higher-level issue

- Familiarity of the data is critical (i.e., *exploration*)!
  - *Granularity of data: (We can generally know this by reading the data documentation, but it is still important to confirm through exploration.)*
    - What each row of a dataset is describing?
  - *How are different sub-categories are represented?*
    - Are all categories represented sufficiently?
    - For example: time

Scope of time

- When was the data collected or updated?
- Does the data cover every time period sufficiently?
  - *Balancedness of data is a requirement if we want to compare different categories.*



Cleaning data: general issues

- Real-world data is always messy, such as the following issues:
  - *Duplicated data (rows & IDs)*
  - *Errors in the data*
  - *Inconsistent formats*
  - *Outliers and missing values*
- How do we find them?

ID	DEPARTMENT	PHONE NUMBER	ZIP	CITY	STATE
1	Fire Department	718-999-FDNY	10004	New York	NY
2	Community Affairs	718-999-1438	60611	Chicago	IL
3	EMS Command	718-999-2770/1753	60611	Chicago	IL
4	Human Resources	718-999-2164	90054	Los Angeles	CA
5	HR	718-999-2164	90054	LA	CA
6	Intern Program	718-999-2181		SF	CA

Data standardization

- Different formats of values
  - *For example, “HR” vs. “Human Resources”*
- Statistically standardization
  - *We may need to standardize some variables for statistical models, as will discuss in the next few weeks.*

Date

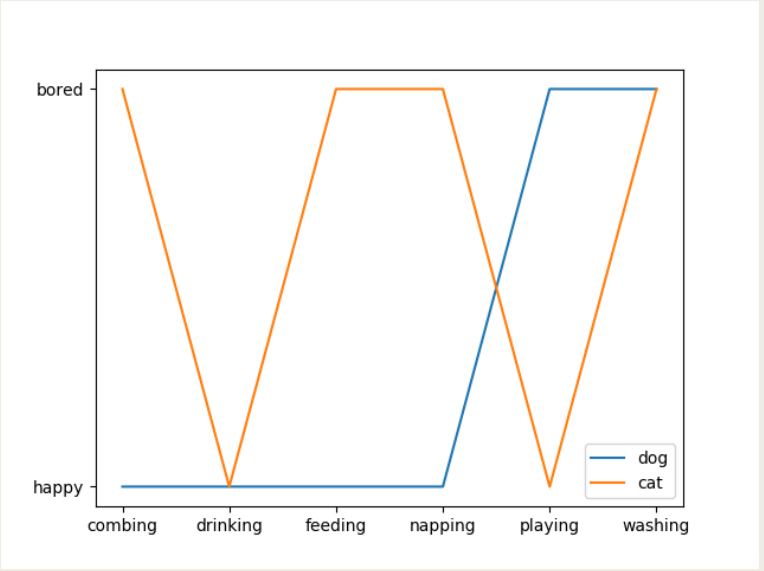
- For example, one type of information that often needs standardization is time. There are many different formats for date/time information:
  - *2024/12/6*
  - *2024-12-06*
  - *12.06.2024*
  - *https://en.wikipedia.org/wiki/List_of_date_formats_by_country#:~:text=For%20English%20speakers%2C%20MDY%20(mmmm,%2Fle%209%20avril%202019)*.

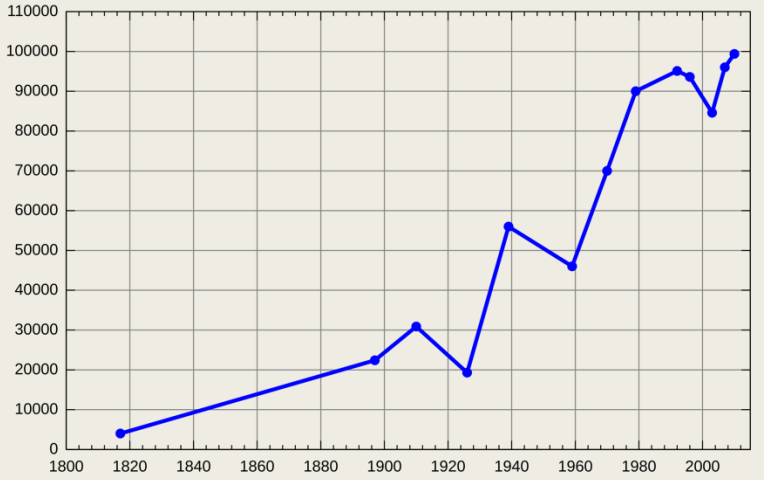
Date/time visualization

- We will most likely use line charts to visualize date/time data, with the date/time information presented in the x-axis.
  - *What is the difference between line chart and bar chart?*
- That said, there are many other possibilities of visualizing date/time information.

Line chart vs. bar chart

- Both are very commonly used 2-d graphs.
- In the line chart, the slope of the line matters, so we must make sure that the x-axis is a meaningful data series.
  - *For example, time series*
  - *Instead, we cannot use a categorical variable in the x-axis of line charts (in this case, we should use a bar chart).*
- Another example





Missing values

- There can be two types of missing values:
  - *Explicit missing values: individual values in a row that are missing*
    - NA: data point not available (but should); similar to "NaN"
    - NULL: data point that is not possible (such as no answer or undefined value)
    - They are treated slightly differently in R (see our demonstration).
    - We can spot these values relatively easily from descriptive analysis.
  - *Implicit missing values: the whole row is missing*
    - For example, we have the data for every month of the year, except for 2024/2.
    - We must pay special attention to this situation!

Missing values: solutions

- There are a few solutions to missing values:
  - *Drop the records with missing values*
  - *Drop the whole variable*
    - if most of the values in a variable are missing data
  - *Imputation, or inferring the missing values*
    - Replace the missing values with the average value (in the whole population or in a sample)
    - However, its usage could have an impact on the shape of data. So we should be super careful to use this approach!

Outliers

- An outlier is a value that is significantly deviating from other observations.
- Outliers can be real values or from error in the data collection or processing pipeline.
  - *So the challenge is to (1) figure out the reason for the value, whether the value is genuine or not and (2) the impact of the outliers to the results.*
- A major part of this assessment will need our domain knowledge and common sense.

Outliers

- For real outliers, we can use statistical transformation to reduce them into ”normal” values.
- For outliers that are errors, we can of course remove them.
  - *But, we still need to consider the consequence of the removal.*
  - *For example, the outlier may be in a case that is very special in the whole population.*

Reshape tables

- There are two types of tables that we may use: long tables and wide tables.
- Different functions may need a certain type of table.
  - *Generally, long tables are more useful for most visualization applications.*
- We can use the reshape function to manipulate them.

Wide Format

Team	Points	Assists	Rebounds
A	88	12	22
B	91	17	28
C	99	24	30
D	94	28	31

Long Format

Team	Variable	Value
A	Points	88
A	Assists	12
A	Rebounds	22
B	Points	91
B	Assists	17
B	Rebounds	28
C	Points	99
C	Assists	24
C	Rebounds	30
D	Points	94
D	Assists	28
D	Rebounds	31

Cleaning data: solutions

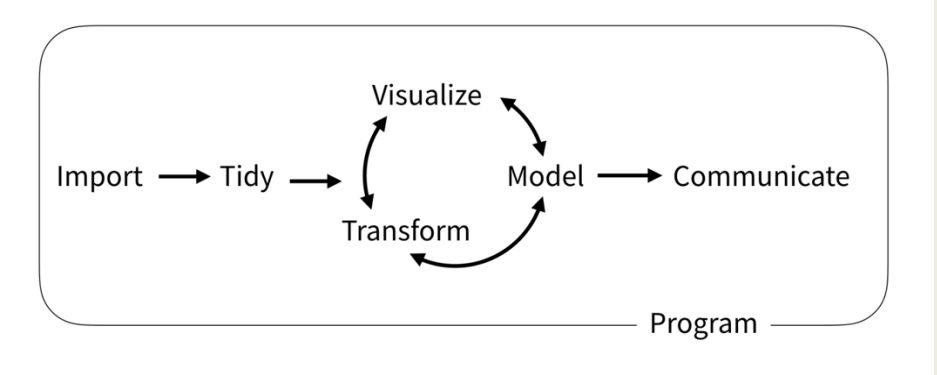
- Real-world data is always messy, such as the following issues:
  - *Duplicated data* → *remove data*
  - *Errors* → *fix or remove errors*
  - *Inconsistent formats* → *standardization*
  - *Outliers and missing values* → *really depends*

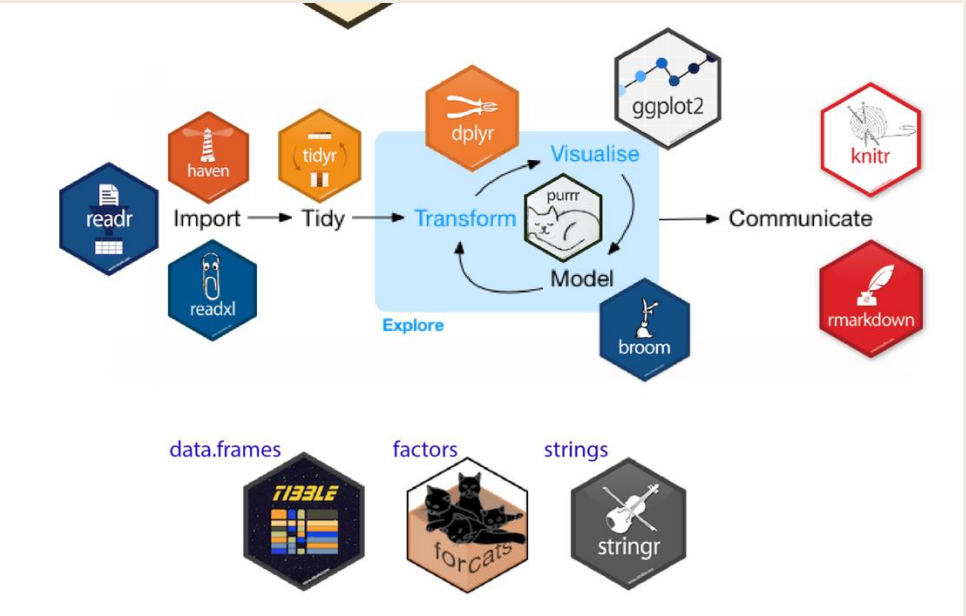
Additional comments on data wrangling

- Even though it is hard, try to have a plan for data wrangling before doing it.
  - *Having too many ad hoc decisions and steps can introduce unintended inconsistencies to the data, because there may be interferences between some actions.*
- Document every step of your data wrangling step.

tidyverse

- Tidyverse is a collection of R packages (or *framework*) using the same pipeline used for data cleaning and preparation.
  - *Instead of relying on individual functions, we can express the whole lifecycle as a series of connected tidyverse-supporting functions.*
  - *These functions are connected by "Subject %>% Verb" structure.*
- https://www.tidyverse.org/





Demonstration

- 1. Data cleaning steps
- 2. Tidyverse
  - *Many examples are taken from: https://oliviergimenez.github.io/intro_tidyverse/#1*