BUSN 5000

Data Fundamentals

Chris Cornwell

Terry College of Business

Fall 2023

Section 1

Preliminaries

Week 1 :: Reading and listening

Reading: Bekes and Kedzi, ch 1; Bueno de Mesquita and Fowler, ch 1; Healy, ch 2

IRL: "The Data Coach", Against the Rules, Season 2 Episode 5.

R highlights

Packages

- `readr` for `read_csv` and `write_csv` (part of the tidyverse)
- `knitr` for report generation
- `skimr` for the `skim` function
- `alfred` for accessing data from FRED

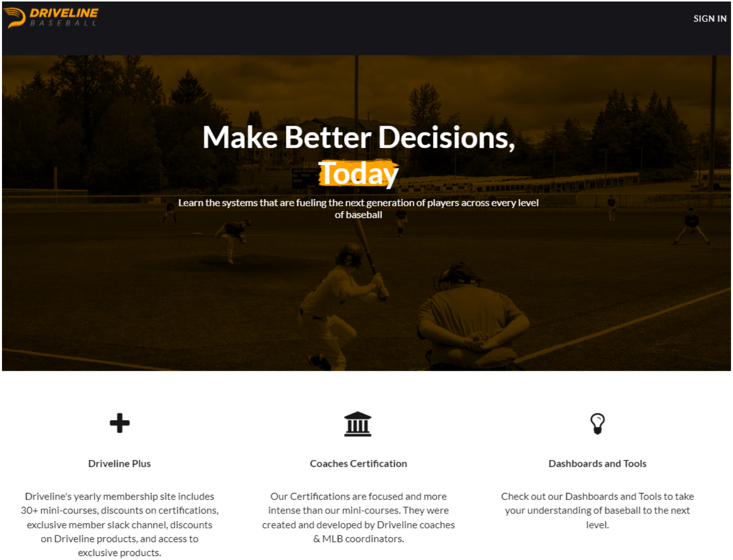
Functions

- `read_csv` for reading csv files
- `write_csv` for write csv files
- `here` for easy file referencing in project-oriented workflows
- `str` for showing the internal structure of an object
- `skim` for an overview of a data frame
- `obj.size` for determining the memory used by an object

The main things

1. Data tables and data sets
2. Tidy data
3. Data value chain
4. Aspects of data quality
5. Reproducibility
6. Notebooks and reproducibility
7. Data documentation

It’s about making better decisions



Make Better Decisions, Today

Learn the systems that are fueling the next generation of players across every level of baseball

Driveline Plus

Driveline's yearly membership site includes 30+ mini-courses, discounts on certifications, exclusive member slack channel, discounts on Driveline products, and access to exclusive products.

Coaches Certification

Our Certifications are focused and more intense than our mini-courses. They were created and developed by Driveline coaches & MLB coordinators.

Dashboards and Tools

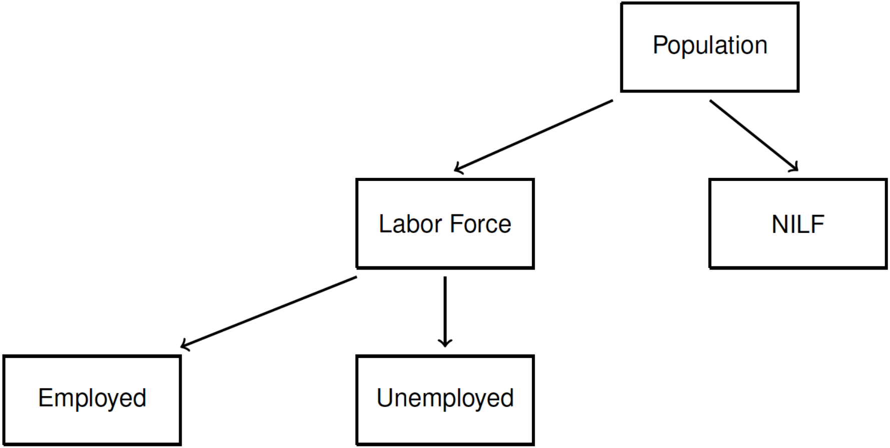
Check out our Dashboards and Tools to take your understanding of baseball to the next level.

Source: Driveline Baseball

Section 2

Labor Force Participation

Basic measurement concepts



Current Population Survey (CPS)

“The CPS is administered to a scientifically selected multistage probability-based sample of households designed to represent the civilian noninstitutional population of each state and the United States as a whole.

“The CPS starts with a probability sample of about 74,000 assigned housing units each month. Approximately 62,000 housing units are eligible for interview each month, after excluding those which are destroyed, vacant, converted to nonresidential use, contain only people whose usual place of residence is elsewhere, or are ineligible for other reasons.

Interviews are completed for about 54,000 housing units. In a given month, about 13 percent of housing units are not interviewed, because of the temporary absence of the occupants, other failures to make contact after repeated attempts, inability of people contacted to respond, unavailability for other reasons, and refusals to cooperate. Information is obtained each month on about 105,000 people ages 16 and older.”

More on the survey design here.

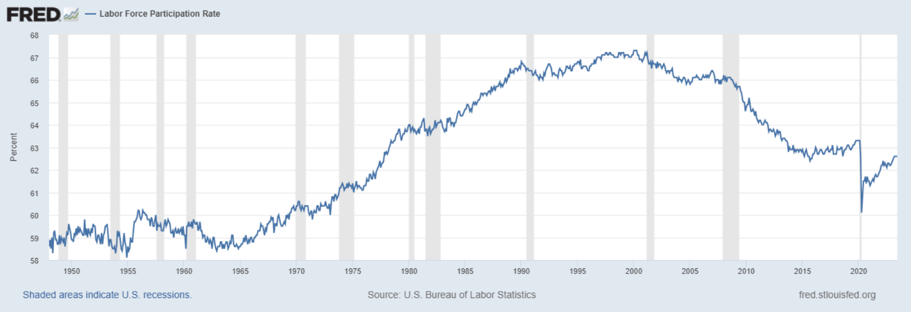
Annual Social and Economic supplement (ASEC)

“This Annual Social and Economic (ASEC) Supplement provides the usual monthly labor force data, but in addition, provides supplemental data on work experience, income, noncash benefits, and migration. Comprehensive work experience information is given on the employment status, occupation, and industry of persons 15 years old and over. Additional data for persons 15 years old and older are available concerning weeks worked and hours per week worked, reason not working full time, total income and income components. Data on employment and income refer to the preceding year, although demographic data refer to the time of the survey.”

ASEC 2022 technical documentation and data dictionary

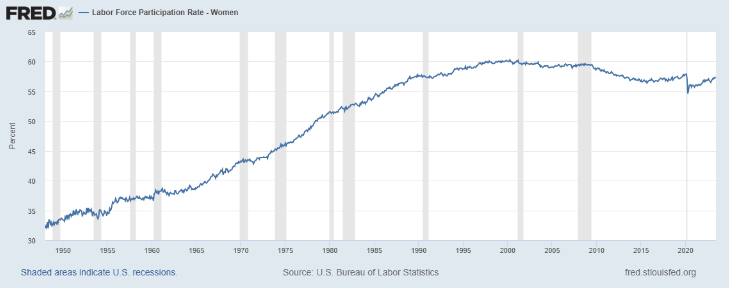
LFPR 1948-2023

What is the question?

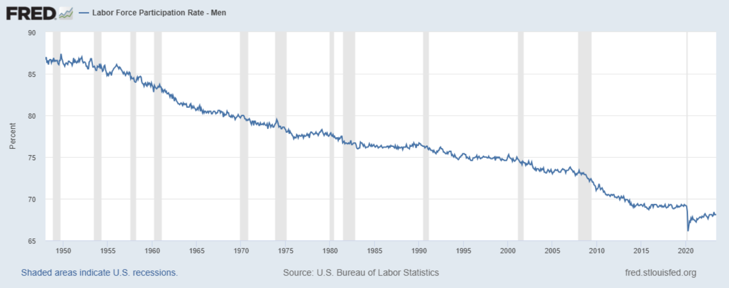


Source: FRED

Women and men



Source: U.S. Bureau of Labor Statistics



Source: U.S. Bureau of Labor Statistics

What’s the matter with the men?

March 10, 2023

The Men — and Boys — Are Not Alright

Richard Reeves breaks down the evidence that many American males are falling behind in education, employment and health.

Podcast link

Section 3

Talking Data

What do we mean by “data”?

The term *data* refers generically to any collection of discrete symbols that convey information.

Data can be either *structured* (organized in a table of records and fields) or *unstructured* (such as text, image and video files).

Structured data are stored in *data warehouses* using a predefined *data model* and format, while unstructured data are stored in *data lakes* in their raw form. (IBM’s take on the distinction.)

When we speak of a *data set*, we are talking about a collection of data that is restricted to a known and specific structure and type of content.

“Data” and “data set” are often used interchangeably, but this is wrong because **“data” is plural** and “data set” is singular. The singular form of data is *datum*.

Records and fields

A data set is made up of *records* that contain information on a specific type of entity (e.g. a person).

The *unit of record* is the type of entity to which the records in a data set correspond.

Each record is made up of a set of known *fields* that contain measurements of known types.

For example, in a data set that has information on the demographic characteristics of individual people, the unit of record is a person, and each record might contain measurements of a person's race, gender, and age.

Observations and variables

We will generally not speak in terms of records and fields. For us, a record is an *observation* and a field is a *variable*.

Notationally, we will indicate an observation on a particular variable by tacking on a subscript i.

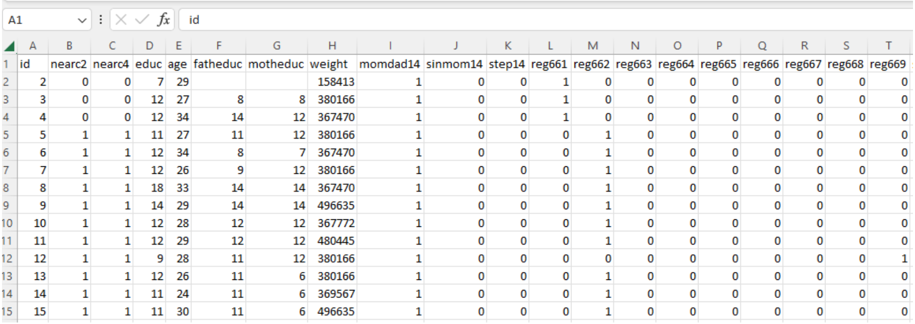
Let's say *educ* is a variable in our data set that measures years of schooling. We will say that the variable $educ_i$ contains the years of schooling for person (observation) i. In general, x_i indicates the observation i on the variable x.

The observation index i will have values that run from 1 to the number of observations in the data set, which we will call N.

Tabular data

We are going to restrict our attention to *tabular data*, i.e. data that can be organized in a single *data table*. Think a single sheet in an Excel workbook, or, more simply, a `csv` file.

Here is a snip of a `csv` file containing the Card data, which will be featured prominently early in Part II.



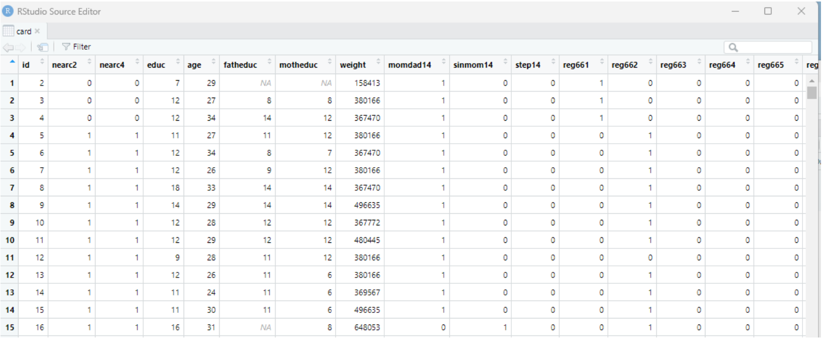
A1 | *fx* | id

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1	id	nearc2	nearc4	educ	age	fatheduc	motheduc	weight	momdad14	sinmom14	step14	reg661	reg662	reg663	reg664	reg665	reg666	reg667	reg668	reg669
2	2	0	0	7	29			158413	1	0	0	1	0	0	0	0	0	0	0	0
3	3	0	0	12	27	8	8	380166	1	0	0	1	0	0	0	0	0	0	0	0
4	4	0	0	12	34	14	12	367470	1	0	0	1	0	0	0	0	0	0	0	0
5	5	1	1	11	27	11	12	380166	1	0	0	0	1	0	0	0	0	0	0	0
6	6	1	1	12	34	8	7	367470	1	0	0	0	1	0	0	0	0	0	0	0
7	7	1	1	12	26	9	12	380166	1	0	0	0	1	0	0	0	0	0	0	0
8	8	1	1	18	33	14	14	367470	1	0	0	0	1	0	0	0	0	0	0	0
9	9	1	1	14	29	14	14	496635	1	0	0	0	1	0	0	0	0	0	0	0
10	10	1	1	12	28	12	12	367772	1	0	0	0	1	0	0	0	0	0	0	0
11	11	1	1	12	29	12	12	480445	1	0	0	0	1	0	0	0	0	0	0	0
12	12	1	1	9	28	11	12	380166	1	0	0	0	0	0	0	0	0	0	0	1
13	13	1	1	12	26	11	6	380166	1	0	0	0	1	0	0	0	0	0	0	0
14	14	1	1	11	24	11	6	369567	1	0	0	0	1	0	0	0	0	0	0	0
15	15	1	1	11	30	11	6	496635	1	0	0	0	1	0	0	0	0	0	0	0

Reading csv files

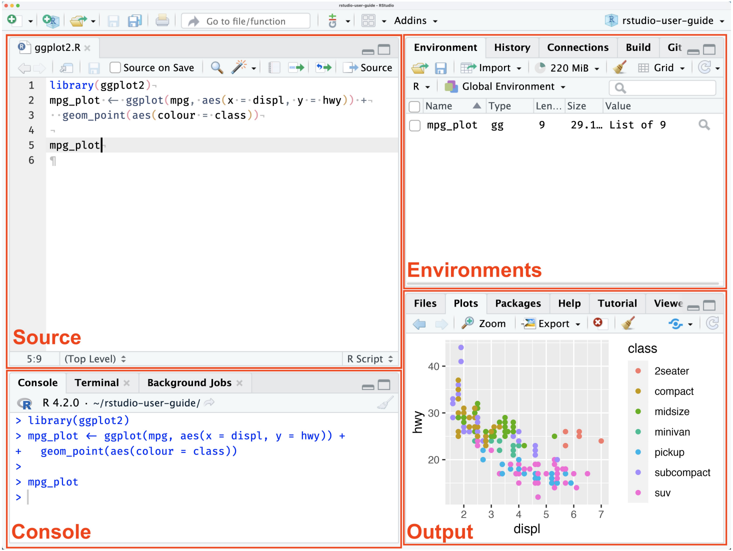
We will use the function `read_csv` to read `csv` files into R. The `View` function opens the data browser in RStudio which presents the contents of the file in a spreadsheet-like format.

```
card <- read_csv("data/card.csv")
View(card)
```



	id	nearc2	nearc4	educ	age	fatheduc	motheduc	weight	momdad14	sinmom14	step14	reg661	reg662	reg663	reg664	reg665	reg
1	2	0	0	7	29	NA	NA	158413	1	0	0	1	0	0	0	0	
2	3	0	0	12	27	8	8	380166	1	0	0	1	0	0	0	0	
3	4	0	0	12	34	14	12	367470	1	0	0	1	0	0	0	0	
4	5	1	1	11	27	11	12	380166	1	0	0	0	1	0	0	0	
5	6	1	1	12	34	8	7	367470	1	0	0	0	1	0	0	0	
6	7	1	1	12	26	9	12	380166	1	0	0	0	1	0	0	0	
7	8	1	1	18	33	14	14	367470	1	0	0	0	1	0	0	0	
8	9	1	1	14	29	14	14	496635	1	0	0	0	1	0	0	0	
9	10	1	1	12	28	12	12	367772	1	0	0	0	1	0	0	0	
10	11	1	1	12	29	12	12	480445	1	0	0	0	1	0	0	0	
11	12	1	1	9	28	11	12	380166	1	0	0	0	0	0	0	0	
12	13	1	1	12	26	11	6	380166	1	0	0	0	1	0	0	0	
13	14	1	1	11	24	11	6	369567	1	0	0	0	1	0	0	0	
14	15	1	1	11	30	11	6	496635	1	0	0	0	1	0	0	0	
15	16	1	1	16	31	NA	8	648053	0	1	0	0	1	0	0	0	

The panes of RStudio



```r
library(ggplot2)
mpg_plot <- ggplot(mpg, aes(x = displ, y = hwy)) +
  geom_point(aes(colour = class))

mpg_plot
```

Name	Type	Len...	Size	Value
mpg_plot	gg	9	29.1…	List of 9

```
> library(ggplot2)
> mpg_plot <- ggplot(mpg, aes(x = displ, y = hwy)) +
+   geom_point(aes(colour = class))
>
> mpg_plot
>
```

Full Posit guide to RStudio

“Tidy” data

Hadley Wickham, creator of the tidyverse, introduced the notion of *tidy* data.

We will say that data are *tidy* if

1. each variable corresponds to a column
2. each observation corresponds to a row
3. every entry (cell) corresponds to a single value

In all of our work, we will presume the data are tidy or we will make them so.

Cross sections, time series, and panels

Cross section

- many units observed at a particular time
- the order of observations does not matter
- CPS

Time series

- a single unit observed over multiple time periods
- order matters
- FRED labor force participation rate

Panel

- same units observed over multiple time periods
- order matters within unit
- National Longitudinal Study of Young Men (NLSYM)

Data sets vs databases

A *data set* may comprise multiple data tables structured for a particular analysis. When a data set is a single table, the terms “data set” and “data table” can refer to the same thing.

The term *database* refers to a collection of tables where each table has some known and meaningful relationship to the other tables and is usually managed with a database management system (DBMS) like MySQL. For example, UGA has a database that links student application, financial aid, and course enrollment information.

Big data

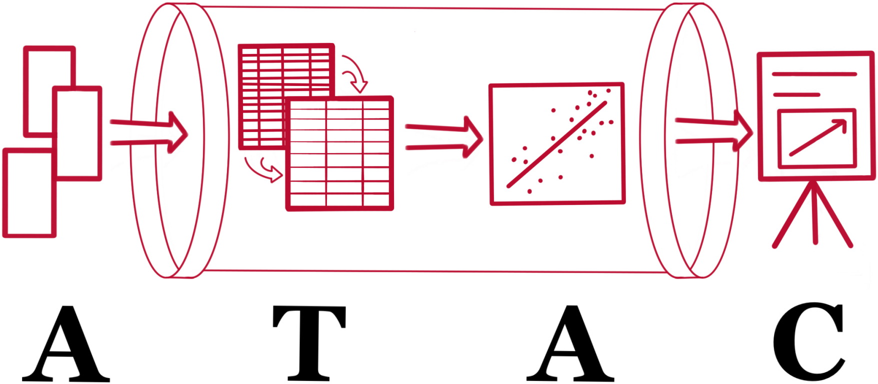
"Big data" is a fluid concept, but is generally defined in terms of

- Volume – Literal size and scale as in the number of petabytes
- Velocity – Speed of generation, collection and storage
- Variety – Complexity of sources and forms
- Veracity – Degree of consistency and completeness

Section 4

The Data Value Chain

Acquire, Transform, Analyze, Communicate



Acquisition

Internally or externally sourced?

Actively or passively generated?

Structured or unstructured form?

Mechanisms and modes:

- Sampling
- Surveys
- Administrative
- Automated systems and sensors (web scraping, APIs, GPS tracking)

What is the question?

Transformation

Cleaning and tidying

- Structuring data sets to facilitate analysis
- `tidyr` is the tidyverse way
- Time intensive

Filtering, selecting and mutating

- Data manipulation relevant to analysis
- `dplyr` provides the tidyverse verbs to do it
- `filter` selects observations that satisfy certain conditions
- `select` keeps or drops variables based on their names
- `mutate` creates new variables as functions of existing variables

Analysis

The workhorse of data science is the CEF

$$E(outcome|factors) = E(y|x\text{s})$$

Descriptive

- Summarize and describe the main features of y and the xs
- Measures and visualizations of central tendencies and variation

Predictive

- Predict y based on a set of xs
- Machine-learning methods

Causal

- Determine the effect of some x – call it a "treatment" – on y
- Causal inference methods

Communication

Reporting - like the monthly communications about the labor force

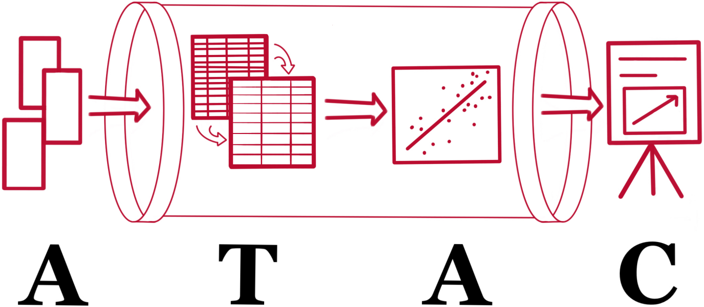
Packaging

- Expanding the usefulness of analysis outputs by creating a “wrapper” or interface that makes them more accessible to the end user
- Like credit agencies’ analysis of consumer banking and spending for financial services companies

Selling

- Own use – for internal process/product optimization
- Trading in – like selling data or aggregated market intelligence
- Trading on – like using data to sell targeted advertising

ATACing labor force participation



Survey –> cleaned & tidy'd March CPS –> labor force characteristics –> report

Mo money mo problems

The point of this is to learn **learn something about the world**. But, the world is messy and so are the data we are usually able to collect.

When we move from the classroom to the real world, there are a bunch of areas where data analysis can break down.

Even worse – it is not always obvious that there is a problem.

One goal of this class is to recognize where problems might crop up and what to do about them.

Data quality

The key aspects of data quality include

- Content – what a variable measures
- Validity – whether a variable measures what it is supposed to
- Reliability – whether repeated measurements return the same value
- Comparability – whether a variable is measured the same way across units
- Coverage – whether all units intended for inclusion are included
- Selection – whether selected units are representative of those not covered

Common issues

Entity resolution

- Duplicate observations
- Ambiguous entity ids across tables
- Non-entity rows (like summaries of variable contents)

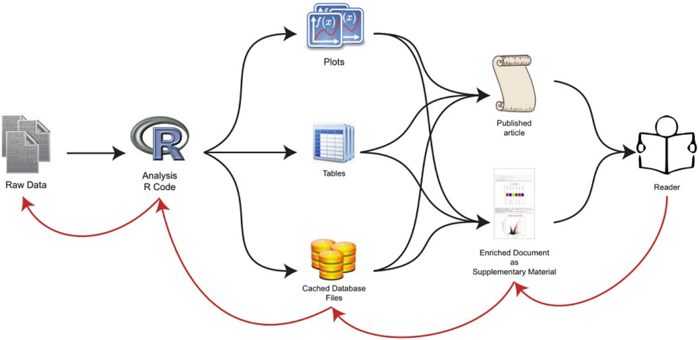
Missing data

- Discovery – how are missing values recorded?
- Coverage implications
  - Missing at random – effect on sample size and power
  - Missing endogenously – concern about selection bias

Section 5

Reproducibility

Same data, same results



ATAC as a production pipeline

In data analysis it is not enough to just focus on the set of final outputs, like a set of figures and tables or a report describing them. Data analysis consists of *all the resources used in the analysis*, including the raw data and the set of instructions that generated the output. Without this information, it is not possible to fully understand what the data analysis is telling us.

A data analysis is *reproducible* if

1. The results or outputs of the original analysis can be generated using the same inputs
2. The inputs used for the original analysis are readily available

Reproducibility at each stage of production

A reproducible analysis just requires the raw data and all of the code or instructions needed to get from the raw data to the output, but it is advisable to separate the Acquisition, Transformation and Analysis tasks.

	Acquisition	*Transformation*	*Analysis*
Inputs	code to acquire raw data	raw input data files code to transform data	analysis data code for analysis
Outputs	raw input data files	data ready for analysis	results of analysis

Each step of the ATAC chain has different inputs and outputs, with the outputs from the prior stage becoming inputs for the next. You should make a record of the inputs and the outputs from each stage of the analysis.

Given the wide range of exchanges possible at the Communication stage, those tasks are necessarily separate.

Why reproducibility matters

Reproducibility matters for important practical and conceptual reasons:

- For your *future self* and others who will use your work
- For transparency to guard against error and fraud related to *confirmation* and *publication biases*
- As a response to the *replication crisis* in the social and behavioral sciences

Automating the production stages

Reproducibility involves developing scripts to *automate* the ATA stages. The ideal is to

1. automate everything that can be automated and
2. have a single script that executes the full analysis from beginning to end.

This way the results can always be re-generated by executing of a single bit of code ("one click") on the same raw data.

The alternative is to carry out your analysis *interactively*, one step at a time. Think cutting and pasting from one spreadsheet to another, restricting range of cases, and applying some function to the restricted set. Or, executing commands one after another at the RStudio console. **Don't do this.**

Documenting the inputs

Reproducibility involves

- Identifying what each input is for and how it relates to the other inputs
- Commenting scripts to explain what is going on at each step
- Describing the exact *provenance* (source) of your raw input data

A reproducible data analysis is a *product* that you should be able to produce over and over.

For detailed guidance for managing code and data in analytics projects, see Gentzkow and Shapiro (2014), "Code and Data for the Social Sciences: A Practitioner's Guide".

Beyond input-based reproducibility

Apart from the inputs, your computing environment (OS, version of R, new packages) can affect reproducibility.

With cloud-based computing, virtual machines, and containerization, an analyst can write a set of instructions that will:

- Request resources for a specific computing environment
- Obtain and install all the software needed to perform the analysis, including specific versions of the operating system, the statistical programming languages involved, and any packages.
- Install the input data and analysis files into a particular location
- Execute the analysis

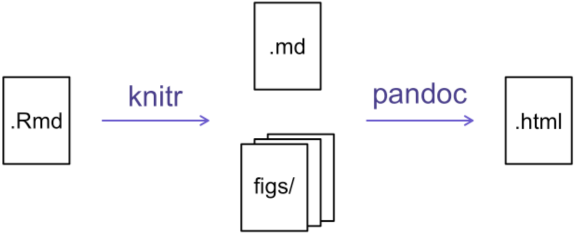
These computing environment issues are far outside the scope of this class, but these tools are increasingly common in industry and academic settings. Here is an introduction if you are interested.

Using R Markdown for reproducible analysis

R Markdown facilitates literate programming using a notebook interface that allows you to “knit” code and text together to create reproducible analysis reports. Quarto and Jupyter Notebook are other examples.

R Markdown is a combination of R and Markdown, a simple mark-up language for creating web pages.

When you `knit` a `.Rmd` document, a plain `.md` document is created, along with any created figures. The `.md` document and figures are processed by pandoc, which converts everything into whatever format you chose.



Source: Reproducible reports with R Markdown. (You are not restricted to html output; `knitr` can produce PDF and Word documents as well.)

Section 6

Data Documentation

Best practices

Best practices would have you create a document for each data set that records

- Data provenance
- Data format and file structure
- Data schema
- Use restrictions and licensing
- Known measurement issues

Provenance

How were the data originally generated? Where and when did you originally obtain the data? What has happened to the data since then?

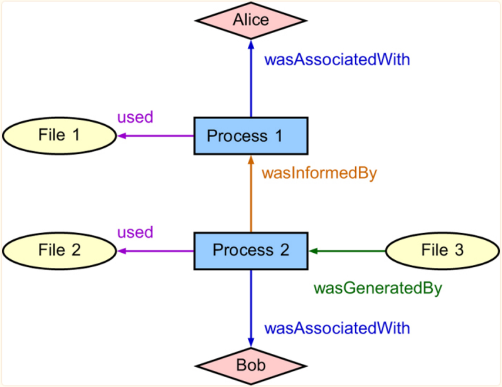


Figure 1

A simple W3C PROV-DM compliant provenance graph.

Source: Pasquier et al. (2017)

Data format and file structure

- What are the names of the files containing the raw data?
- How large are the files in bytes?
- In what format are the data encoded (e.g. `csv`, `xlsx`, `rds`)?

Data schema

A *data schema* is a representation of the data structure and comprises all the attributes of the data and their data types, including

- The number and name of each data table
- For each table
  - The number of observations that appear in the table
  - The list of variables
  - The format of the information in each variable contains (e.g. numeric, string, etc)
- The *unit of record* or entity to which each observation pertains (e.g. year, person, person-year)
- The *key variables* that uniquely identify an observation (e.g. `year`, `person_id`, `year` + `person_id`)

Use restrictions and known measurement issues

The list of questions here is open-ended but include

- Can the data be freely used and redistributed?
- Is there an open-source license that must be respected?
- What steps do users need to take to obtain the data?

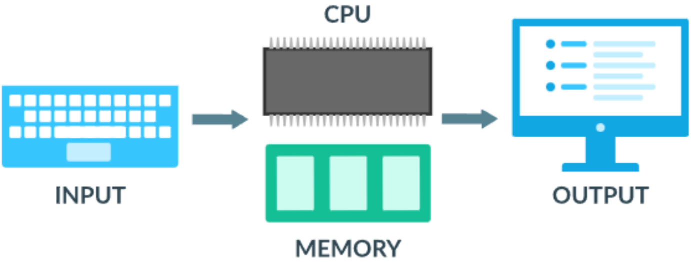
- What is missing?
- Are there sample weights to consider?
- Is there a disclosure limitation or privacy protection?

Section 7

Managing computing resources

How computers work

Your data are *in storage*, either on a local disk (SSD or HDD) or remotely in the cloud, until you execute a command to read them. When you do, they are moved to your central memory (RAM) where they can be processed by the CPU.



Storage unit measures

Unit	Bytes	Example
Bit	1/8	1 or 0 (on/off)
Byte	1	a single text character
Kilobyte (KB)	1024 (2^{10})	a very short text document
Megabyte (MB)	1024^2	a decent quality photo
Gigabyte (GB)	1024^3	2 hrs of SD video
Terabyte (TB)	1024^4	human genome sequence
Petabyte (PB)	1024^5	20m filing cabinets of text

Everything is an object

In R, *everything* is stored as an *object*, which can have attributes, such as name, dimension, class, etc.

When you create an object, R allocates space in your computer's memory to hold it. Your computer's memory is a fixed resource that you will have to manage.

The amount of memory R allocates for storing data depends on the *data type* and *data structure*.

The function `object.size` estimates how many KB an object "weighs". If you want to get into the weeds of memory management, check out Hadley Wickham's Advanced R text. He recommends using `obj_size` from the `lobstr` package to "weigh" an object.

R data types

Data Type	Description	Allocation
Numeric (`num`)	real numbers	8 bytes/number
Integer (`int`)	integer values	4 bytes/number
Character (`chr`)	strings and text	1 byte/letter
Logical (`lgl`)	boolean values	4 bytes/value*
Complex (`cpl`)	complex numbers	16 bytes/number

Unlike some programming languages, you do not have to tell R what kind of data you are loading into it. It will basically guess what the data type should be based on a scan of the first few instances.

R data structures

Data Structure	Description
Vector	A sequence of data elements of the same type
Matrix	A two-dimensional array of data elements of the same type
Data Frame	A tabular data structure
List	An ordered collection of objects
Factor	A vector that can contain only predefined values, and is used to store categorical data
Array	A multidimensional collection of same-type data elements

A couple of examples

Suppose your data set has 1m observations on 10 variables that only contain integer values. What are the storage requirements if the data are stored as `num`? As `int`?

How much space is need to store the Card data, which contains 3010 observations on 34 `num` variables?

Section 8

Back to the labor force

Computing the LFPR

Let $E =$ number employed and $U =$ number unemployed. Then, the labor force (LF) is defined as

$$LF = E + U,$$

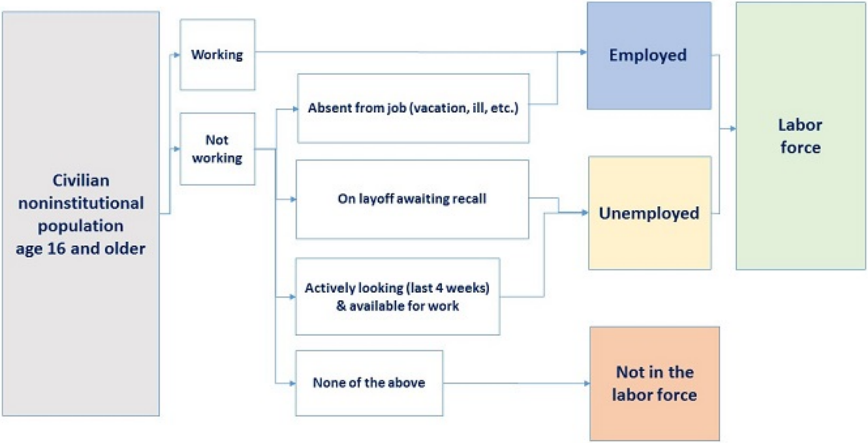
and those not in the labor force ($NILF$) are just the residual of the overall population (Pop),

$$NILF = Pop - LF.$$

The labor force participation rate ($LFPR$) is just the ratio of the labor force to the population:

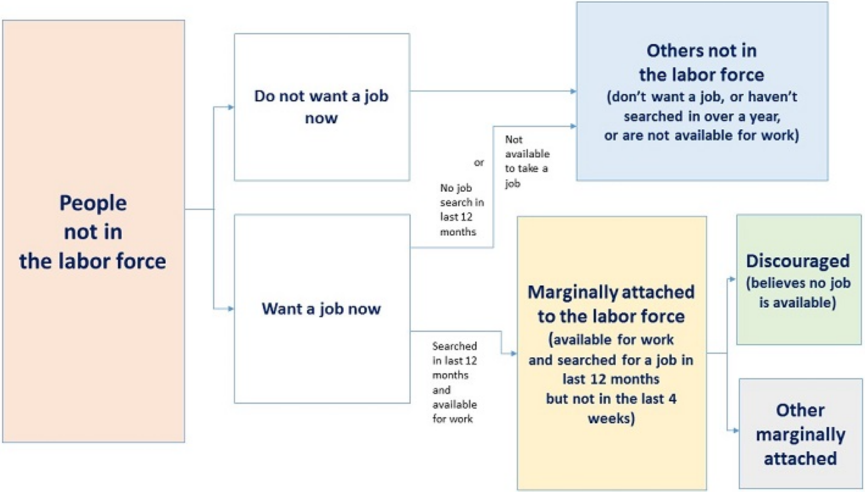
$$LFPR = \frac{LF}{Pop}.$$

Who is in the labor force?


Civilian noninstitutional population age 16 and older
Working
Not working
Absent from job (vacation, ill, etc.)
On layoff awaiting recall
Actively looking (last 4 weeks) & available for work
None of the above
Employed
Unemployed
Not in the labor force
Labor force

Source: BLS Concepts and Definitions

Who is not in the labor force?


People not in the labor force
Do not want a job now
Want a job now
No job search in last 12 months
or
Not available to take a job
Others not in the labor force (don't want a job, or haven't searched in over a year, or are not available for work)
Searched in last 12 months and available for work
Marginally attached to the labor force (available for work and searched for a job in last 12 months but not in the last 4 weeks)
Discouraged (believes no job is available)
Other marginally attached

Source: BLS Concepts and Definitions

Calculating the LFPR from the March 2022 CPS

```
cpsmar <- read_csv("../../Project I/data/pppub22.csv")
```

```
## Rows: 152732 Columns: 832
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr   (1): PERIDNUM
## dbl (831): PH_SEQ, P_SEQ, A_LINENO, PF_SEQ, PHF_SEQ, OED_TYP1, OED_TYP2, OED...
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
# PEMLR
# 1 = Employed - at work
# 2 = Employed - absent
# 3 = Unemployed - on layoff
# 4 = Unemployed - looking
# 5 = Not in labor force - retired
# 6 = Not in labor force - disabled
# 7 = Not in labor force - other
E    <- nrow(cpsmar[cpsmar$PEMLR==1 | cpsmar$PEMLR==2 & cpsmar$A_AGE>15,])
U    <- nrow(cpsmar[cpsmar$PEMLR==3 | cpsmar$PEMLR==4 & cpsmar$A_AGE>15,])
LF   <- E + U
NIFL <- nrow(cpsmar[cpsmar$PEMLR==5 | cpsmar$PEMLR==6 | cpsmar$PEMLR==7 & cpsmar$A_AGE>15,])
Pop  <- NIFL + LF
LFPR <- (LF/Pop)*100
print(paste("March 2022 LFPR =", LFPR))
```

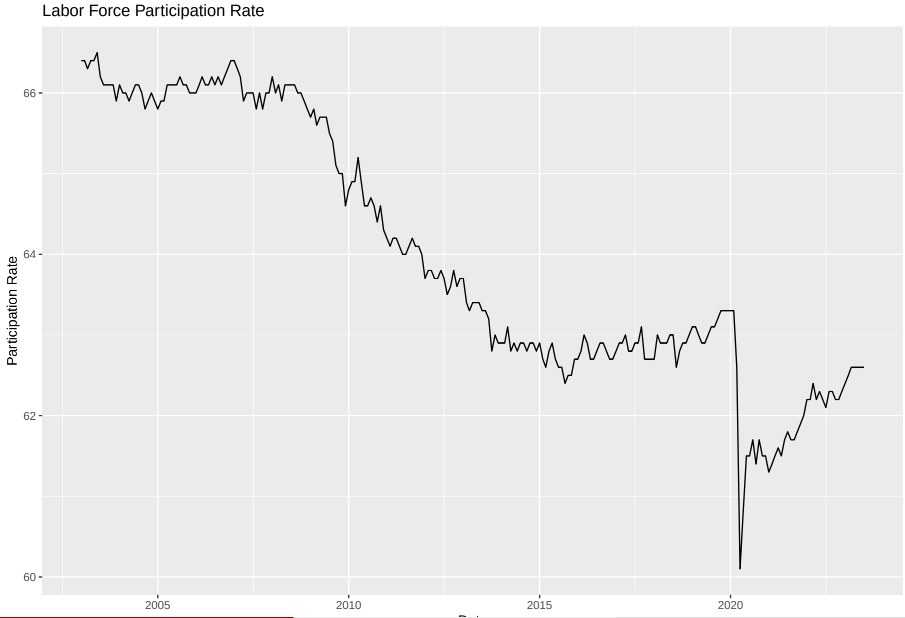
```
## [1] "March 2022 LFPR = 62.3510534982899"
```

Alfred

The `alfred` package allows you to read data directly from FRED. Let's use it to highlight recent trends in the LFPR.

```r
library(alfred)
start_date <- "2003-01-01"
lfpr <- get_fred_series(series_id = "CIVPART",
                        observation_start = start_date)
lfpr_plot <- ggplot(data=lfpr) +
  geom_line(aes(x=date,y=CIVPART)) +
  ggtitle("Labor Force Participation Rate") +
  ylab("Participation Rate") +
  xlab("Date")
```

Recent trends in labor force participation



Potential explanations

- Aging population
- Expansion of disability insurance
- Video games
- Mass incarceration
- Decline in marriage
- Declining opportunities for non-college educated