Data Science for Social Scientists

An applied course using IPUMS data

Daniel E. Ehrlich, IPUMS, University of Minnesota
Anna Tremblay, Dept of Soc, Anth, & CJ, Clemson University

2022-09-07

Contents

Preface

An open-source book using open-source tools and nearly open-source data.

What is IPUMS

IPUMS started as a project to digitize the historical records of the US census. It has expanded to include 9 data collections, which are united in their methods and principles of making soial science research easier. IPUMS data consists of individual-level census and survey data from more than 100 countries around the world. Notably:

- IPUMS **harmonizes** these data, ensuring consistently coded values across time and space.
- IPUMS provides harmonized **GIS Shapefiles** for most census and survey data
- IPUMS provides extensive **metadata**, including
  - Original questionnaire text
  - Alerts about notable changes in variable definition, universe, or coding

IPUMS data is free to use for education and research purposes. Researchers just need to register with an **email address** and brief project description. Nothing too formal - we're just trying to understand what kinds of questions researchers are interested in. For educators, we have additional resources to set up classroom accounts, making it easy to get your students registered and share IPUMS data with them.

Why make this course

While we (DEE) may be slightly biased, we think IPUms is a fantastic resource for **Education** and **Research**. Real-world example datasets provide the bulk of the content for this course, providing an **applied context** we hope students (and instructors) will find engaging. We also know meany instructors may be teaching across multiple disciplines, in large departments, or be the only "data person" at their instutition. We think IPUMS data will be useful to virtually any social science field. We provide some example lessons, and encourage instructors to develop their own, using our `template`, to tailor this course to their subject or interest.

Course Description

This course is broken down into 3, 5-week units. Unit 1 focuses on familiarizing yourself with R and the IPUMS dataset. In Unit 2, each week will showcase a method/analysis using preselected variables. In class, students will walk through a given problem set and produce a lab report by the end of class. In Unit 3, students will work towards answering a research question that they pose, creating a research paper with literature review, data analysis, conclusion, and data outputs.

Course Aims

Provide students with relevant, hands on, methodological training in data literacy and visualization.

LEARNING OUTCOMES

LEARNING OUTCOMES:

- Understand the depth of the IPUMS database and the variables it has to offer
- Compose R code to analyze the IPUMS data
- Produce visually pleasing data outputs in R
- Synthesize the information in a written report
- Present the analysis in a poster format for other students

Guiding Principles

- phenomenon-based learning
  - try to start the class with a **question** or **problem**
  - *why* does the data look the way it does
  - structure class so students work towards solving the problem
- RELEVANT examples
  - try to touch on 2 or more disciplines (eg, economics, demography)

Syllabus

This syllabus is initially envisioned as 3 5-week sections. However, compilation and content are intended to be modular with templates for instructors to include their own specialties.

The basic structure of this course is:

Unit 1 (Weeks 1-5): Understanding and Testing Data

- Students use simple datasets
  - Data provided by the course/instructor
  - simplified data to illustrate trends
    - EG: Age distributions ; Age by Sex
    - potential usecase for synthesized data

Unit 2 (Weeks 6-10): Finding Data and Asking Questions * Students begin to analyze more complex (IPUMS) datasets + Data provided by course/instructor + IPUMS data testing/demonstrating real world effects + EG: SEX ~ EDUATTAIN ; Sex by eduattain by some SES indicators + Students are given a medium/large set of IPUMS variables + Students learn to perform exploratory analysis to guide hypotheses * Students learn to navigate IPUMS website + Students learn to develop a hypothesis and find relevant data

Unit 3 (Weeks 11-15): Discussing Data and Student Research * Students develop a research question to be answered with IPUMS data * Students are encouraged to fit it to their interests/major/disciplie * Course time should be devoted to individual/small-group research * Instructor/class present on recent research * Instructor models constructive / scholarly criticism * Encourage students to critque published work - responsibly

Unit 1 (5 weeks) Understanding and Testing Data

Week 1: Intro to R, data types, data structures

Week 2: Plotting Data, Distributions

Week 3: Statisitcal testing of simple data sets

Week 4: Correlation and Relationships of simple data sets

Week 5: (TBD)

Addl Details

Intro to data/ simple analysis

Students will be able to:

Technical:

- Download R and RStudio
- Read data into R and
- Write (save) data out of R.
- Summarize data visually
  - Using base R
  - Using ggplot (tidyverse)
- Summarize data tabularly
  - Using base R
  - Using gttable / tidyverse
- Formally state and test assumptions of data
  - *EG:* t-test, anova, (maybe) correlations

Conceptual:

- Understand main types of data
  - *EG:* logical, numeric, character, etc
  - R specic vs general terms
- Recgonize various data distributions
  - *EG:* normal, poisson, etc
- Know which types statistical tests are appropriate for a given set of data.

Unit 2 Finding Data and Asking Questions (Using IPUMS Data)

Week 6 Exploratory analysis

Week 7 Hypothesis Testing

Week 8 Statistical Inference

Week 9 (TBD)

Week 10 (TBD)

0.0.0.1 Addl Details

Here we demonstrate two **different** approaches to conducting research. Students become familar writing up short lab reports detailing their findings. For unit 0.0.0.1.1, we/instructor provides students with simple datasets from IPUMS (or other real-world data). Students will learn exploratory data analysis techniques and how to create lab reports to summarize key findings.

For unit 0.0.0.1.2, students will learn to develop their own simple research questions or social-science hypotheses. They will seek out data to answer these questions, learning to navigate ipums.org, and create **data extracts**, as well as hypothesis-testing statistical methods. Again, lab reports to summarize findings.

0.0.0.1.1 Exploratory Analysis If you've just collected a survey, or other raw data, you may not know what you're looking for. This is perfectly ok but goes against *the scientific method* most people learned in grade school (More on that to follow(***include_link***)).

This unit begins by presenting data/distributions and asking students to begin interpreting the data . visual exploration is encouraged and basic of data manipulation are taught * *EG:* how to subset data, how to reshape data, how to recode data, how to convert from one `data type` to another.

Example lab exercise:

Students given a data set (xls, csv, etc) * load data, perform manipulations, basic summaries + cross tabs + group means by a covariate * inspect data visually + *DESCRIBE* the distribution - is it normal? significant? * *FIND* aquestion in the spread of the data + how can you test this (maybe small group work) * write up/ present results + think on confounding factors / biases

0.0.0.1.2 Hypotethsis Driven If, on the other hand you have an a pre-exisiting idea you want to test. We can follow the traditional *scientific method*. With a question in mind, the first question is: where to look. What better place than IPUMS!

Begin introducing navigation of web resources - mainly IPUMS international

Students should become comfortable working through lab exercises: * Define a question (or be presented with one) * Download variables from IPUMS (course downloads possible) * Perform a basic analysis (discussed in Unit 1) * Generate a **visual argument** for your analysis + Include explanation/interpretation/reflection on the question at hand, and the data used + Any obvious biases + Any obvious confounding factors

Unit 3 Discussing Data and Student Research

Week 11: Students develop research Question

Week 12: Students find relevant variables from IPUMS

Week 13: Students test and evaluate results

Week 14: Students prepare presentations of results

Week 15: Students present work (slides, poster, podium, etc)

0.0.0.2 Addl Details

Students will select their own research question that can be answered with the IPUMS data set and will spend five weeks producing a research paper complete with data analysis, visualization, and interpretation.

In this section we encourage the instructor to provide ample time for independent student/small-group research. Some class time should be devoted to modelling healthy discussion and critique of methods.

We provide some examples here but encourage instructors (or students) to bring in recent journal/popular articles that do (or do not) apply data science methods well.

DEV NOTES

TO DO

- Make chapter 1 chapter 2
- Anna Adds chapter con data science intro exclusive of R/IPUMS
- discuss style
  - key terms section for each chapter?
  - key terms in **bold**
  - italics for *emphasis*
  - are we pro-hyphens, or are they pedantic?

MISC IDEAS

- Application forward
- Present research/ analysis/results FIRST, then explain the mathematical principals behind it
- daily/weekly "i'm stuck on…"
  - Students send in questions (night before class) and instructor spends 10-15 mins talking through (or collaboratively working through with class) solutions
  - Alternatively, once a month maybe a longer class covering "common problems asked this month" daily/weekly "recent research"
- pick out a recent article with good visualization (or bad) and spend 5-10 mins discussing what makes it good (or bad)
  - Encourage students to find articles for extra credit

Documentation This function grabs any packages in your project and adds them to a local list that can be referenced using `R-pacakgename` * **NOTE** in practice, that needs to be wrapped in markdown syntax, eg: `[@R-bookdown]` * See help files for more info - might be able to create/add a `citation` file

Chapter 1

Unit 1: The Basics

1.1 Week 1: Intro to R, data types, data structures

1.2 Week 2: Plotting Data, Distributions

1.2.1 Normal Distributions

First we'll generate a normal distribution with the `rnorm()` function. This takes 3 arguments: `n, mean, sd`, which you can see filled in below. While we could print out a list of all these values, it's not easy to *understand* a list of numbers

```
normal_dist <- rnorm(n = 100, ## 100 samples
                     mean = 10, ## with a mean of 10
                     sd = 1 ## and a standard deviation of 1
                     )


normal_dist
```

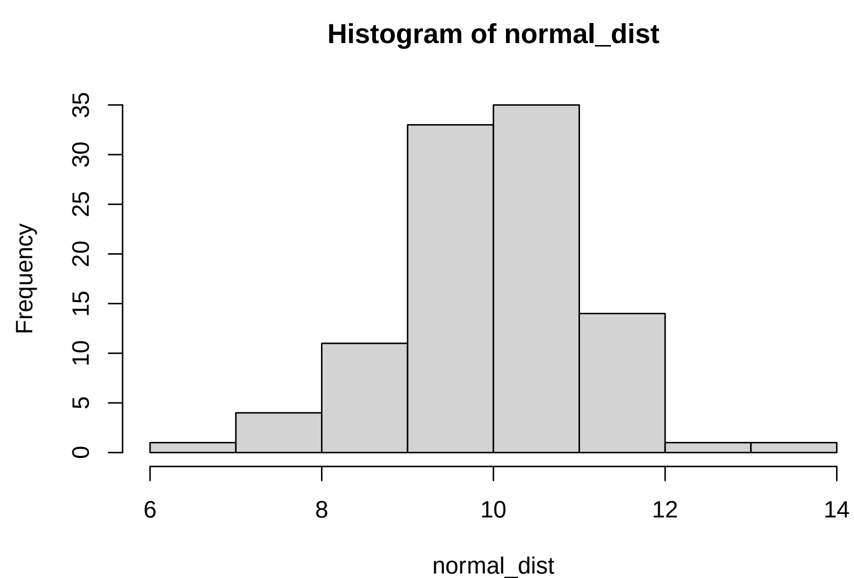
```
##   [1] 13.033154  9.706548  7.657098  9.863114 11.094736  9.474891  9.631290
##   [8] 11.323917  9.365660  8.288121 11.146634 10.434830 11.589687 11.856376
##  [15] 10.910982  9.492497 10.270983  8.957959  8.836700  9.349669 10.775143
##  [22] 10.135217  9.725308  9.597938  9.729460 11.627957 10.437467 10.062037
##  [29]  9.710761  9.052624 10.757093 10.368510  9.279179 11.141923 10.468264
##  [36] 10.380980  8.316040  8.748622 10.402271  9.014280  8.701523  9.957140
##  [43]  9.525941  9.869269  6.986774 10.906572  9.906199 10.552217 10.424223
##  [50]  9.794316 10.579508 10.315345  9.977534  7.739517 11.138667 10.075000
```

```
## [57] 11.111265  8.965880  9.348072 10.182019 10.892381 10.129555 10.440351
## [64]  8.840116  9.842284 10.235484 10.484269  9.871027 11.956154  7.750249
## [71] 11.112196 10.007758 11.634157  8.262024 10.410810  7.780116 11.234180
## [78] 10.998700  9.053740 10.543187  9.886425 12.190800  9.926524 10.095864
## [85]  9.268190 11.804283  9.225618 10.892000 10.308673  9.660945  9.489733
## [92]  8.974609  9.510372 10.743717  9.587023 10.861527 10.173602 10.732303
## [99]  9.746346  8.529163
```

Another better way to look at data would be to **visualize** or **plot** it. One way to to that is with a **histogram**, which groups **continuous values** into **bins**, then plots the **frequency** for each bin.

In R, we use the `hist()` function to plot a histogram of data. We can (try to) control the number of bins with the `breaks` argument, but note that it doesn't always match up. The `hist()` function will adjust based on the distribution of the data.

```
hist(normal_dist,breaks = 5)
```



Another way to visualize this would be with a d

1.2.2 What *is* normal?

1.2.2.1 Quantitative summaries

5num summary * Min, 25th percentile, median, 75th percentile, Max

```
tab_normal_dist <- summary(normal_dist)
```

We can print the table in R by calling its name.

```
tab_normal_dist
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   6.987   9.448  10.035  10.012  10.735  13.033
```

Mean, standard deviation

1.2.2.2 Meaningful Comparisons

How to compare apples to oranges? Standardize the units / standardize the data

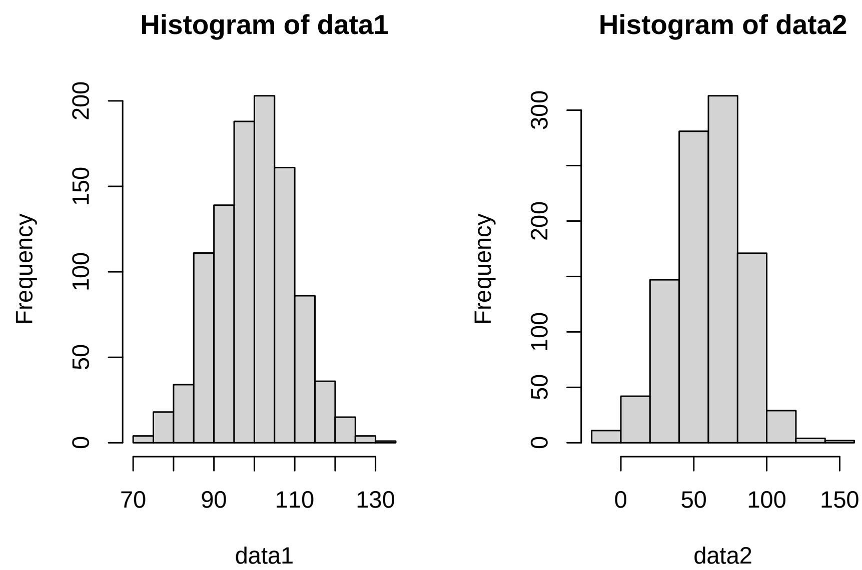
```
data1 <- rnorm(n=1000,
              mean = 100,
              sd = 10)

data2 <- rnorm(n=1000,
               mean = 60,
               sd = 25)
```

Are these the same distribution?

Any issues??

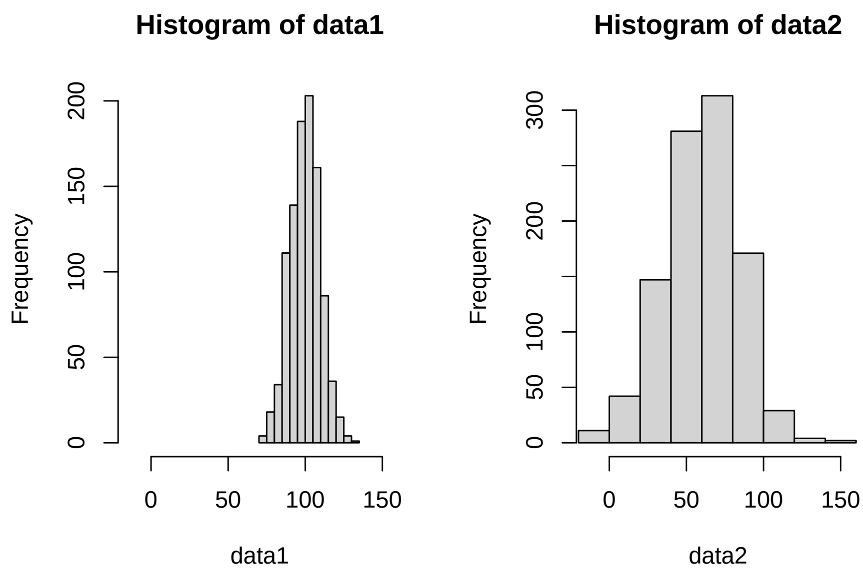
```
layout(matrix(1:2, ncol = 2))
hist(data1)
hist(data2)
```




```
total_range <- range(data1, data2)
```

Are they the same?

```
layout(matrix(1:2, ncol = 2))
hist(data1, xlim = total_range)
hist(data2, xlim = total_range)
```



Numerically / tabularly

Often times its important to tables of **summary statistics**

```
norm_comp_tab <- rbind(summary(data1),
                       summary(data2))

norm_comp_tab
```

```
##            Min.  1st Qu.    Median     Mean   3rd Qu.     Max.
## [1,]  71.26983 93.15937 100.10482 99.92318 106.51155 132.7447
## [2,] -14.53694 42.85820  60.91629 59.92596  77.18745 155.6257
```

Making the table a little nicer. Also an example of **conditional programming**.

```
rownames(norm_comp_tab) ## they're null
```

```
## NULL
```

```
if(is.null(rownames(norm_comp_tab))){
  rownames(norm_comp_tab) <- c("data1", "data2")
}
```

When working with **Rmarkdown** we can take advantage of `knitr` and `pandoc` to nice looking tables even easier.

```
knitr::kable(norm_comp_tab)
```

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
data1	71.26983	93.15937	100.10482	99.92318	106.51155	132.7447
data2	-14.53694	42.85820	60.91629	59.92596	77.18745	155.6257

How transform the data

Simple transformation (multiply all values by 100) * to convert units * other examples?

Complex transformations * log-transformation (*DEE: not a fan*) * z-scores (*DEE: a better option*)

Why transform the data? * Real world applications? * Is it always appropriate to transform data?

1.2.3 Skews

What to do if the data are **not** normal?

1.3 Week 3: Statisitcal testing of simple data sets

1.3.1 t-tests, ANOVA, chi2

1.4 Week 4: Relationships between variables in simple data sets

1.4.1 Correlation, Linear Regression

1.4.1.1 Simple LM

1.4.1.2 Complex LM

1.4.2 Genearlized Linear Model

1.5 Week 5:

For now, I have 3 main chapters for each of the main sections: * Basics of data science / R 1 * Applications/critiques using IPUMS data 2 * Student-driven projects 3

Each of these **Chapters** contains multiple sections. We'll likely want to break these sections out into their own `.Rmd` files as they get fleshed out. For now, I'll try to keep the abundance of files limited.

NOTE: As these actually get filled out, we will probably want to insert different `parts` to the book (EG, the content of Unit 1 is covered in `Part I`). * Declare parts with `# (PART) Part I {-}` immediately before the first chapter `#` it contains.

Topics to include: * What is data? * Everything can be data * How do we interpret data * Tables * Plots * Univariate distributions * What can they tell us * Multi-modality in distributions * Categorical vs continuous data * Don't need to get ahead of this yet * Add in a grouping category - multi state/multinational dataset * Ttest / anova

Type of Data: Age distributions Specifically generate a dataset with old/young folks over-represented to highlight a bimodal distribution

Start with single state/country Add a second state/country to demo ttest Add more to demo anova

Alternatively, income by education level - may be more interesting/relevant to college students (or depressing)

1.6 Intro to R/RStudio

1.7 Reading Data / Distributions

1.7.1 What *is* a normal distribution

1.7.1.1 How normal is it?

show increasingly unclear examples of normal vs not

introduce tests of normality

1.7.1.2 Measuring normality - single sample

reinforce [concept of statistical] **normality**

is a value from a sample? - one way ttest something about tails

1.7.1.3 comparing normality - two saples

standard / two-way t test

1.7.1.4 comparing more than two - ANOVA

Chapter 2

IPUMS

2.1 Week 6 Exploratory analysis

2.2 Week 7 Hypothesis Testing

2.3 Week 8 Statistical Inference

2.4 Week 9 (TBD)

2.5 Week 10 (TBD)

Some text to break up the sub-section headers

2.6 Intro to IPUMS website

2.6.1 background on ipums

2.6.2 navigating website

Find certain (very common) variables to answer (common) social science questions.

We describe our methods in this chapter.

Math can be added in body using usual syntax as follows. This may be useful, particularly for explaining the math side of things.

2.7 math example

p is unknown but expected to be around 1/3. Standard error will be approximated

$$SE = \sqrt(\frac{p(1-p)}{n}) \approx \sqrt{\frac{1/3(1-1/3)}{300}} = 0.027$$

You can also use math in footnotes like this[1]. Footnotes are helpful because they re-link to where you left off.

We will approximate standard error to 0.027[2]

The `longnote` footnote seems particularly useful.

[1] where we mention $p = \frac{a}{b}$

[2] p is unknown but expected to be around 1/3. Standard error will be approximated

$$SE = \sqrt(\frac{p(1-p)}{n}) \approx \sqrt{\frac{1/3(1-1/3)}{300}} = 0.027$$

Chapter 3

Unit 3: Independent Research

3.1 Week 11: Students develop research Question

3.2 Week 12: Students find relevant variables from IPUMS

3.3 Week 13: Students test and evaluate results

3.4 Week 14: Students prepare presentations of results

3.5 Week 15: Students present work (slides, poster, podium, etc)

By this point, students should be familiar with basic concepts from Chapter 1. These include:

- Basic Coding
  - read/write data in/out of R
  - basic manipulations

- Theoretical Basis
  - looking at data distributions
  - formal assessment of distributions

Students will also be familiar with how these concepts are applied from Chapter 2. Hopefully students will be able to:

- Come up with a social science question they are interested in
  - Critically think about target variable(s) of interest. Any *a priori* covariates? confounders?
  - Acquire relevant data from IPUMS
  - Analyze, Summarize, Visualize Data
    - scope and complexity at student/teach discretion
  - Present research to class
    - **potentially** critically discuss/evaluate each others work.
    - **science is collaborative** everyone should be out to do their best work and represent the data as best we can. We all have conscious and unconscious biases, and the best way to confront them is share and receive (respectful) feedback.

During this Unit, we suggest giving ample class time for independent student research, peer-to-peer collaboration, and basic R/stats troubleshooting. This would also be a great time to model how to give respectful criticism by discussing recent research papers. * We could maybe come up with 1-2 seed examples, with a few talking points

3.6 Example one

3.7 Example two

Chapter 4

Example RMD code

For now, this chapter is a bit of a placeholder. I'm not sure what/how the `references.Rmd` file actually fits in to the code/construction (it looks automatic) so I want to keep that in place and need a section to note that.

I also want a more centralized reference point to put any example code I find helpful while working in R/bookdown. This section could get really unrully really fast, but oh well.

4.1 Core

`index.Rmd` is required and treated as file `00`. Chapters *should* be numbered for ease of sorting but custom orders are possible by specifying filenames somewhere **in this file**

Remember each Rmd file contains one and only one chapter, and a chapter is defined by the first-level heading `#`. + **IE** beyond the YAML header this file functions as a normal chapter since it starts with a top level header. + Note that `index.Rmd` has its own YMAL in addition to the various .yml files…not sure exactly how these relate.

Reference a figure by its code chunk label with the `fig:` prefix, e.g., see Figure @ref(fig:norm_dist_plot). Similarly, you can reference tables generated from `knitr::kable()`, e.g., see Table @ref(tab:norm_summary_tab). * Again, this prints an auto-generated numeral * also leaving this in the context of the plots in Chapter 2

You can write citations, too. See `knitr::write_bib()` for more on this. Quick example from demo/index (may not work without write_bib() though): we are using the **bookdown** package (Xie, 2022) in this sample book, which was built on top of R Markdown and **knitr** (Xie, 2015). * If included, "Refernces" section gets added to each chapter. * Not exactly sure where

Embed html renders (EG, fancy tables (IPUMS_var_desc), or any shiny app) with `webshot` R package and `phantomJS`.

```
install.packages("webshot")
webshot::install_phantomjs()
```

4.2 Tips

***Autonumber sections** Note the `{-}` used to indicate "do not number this section" eg: preface.

LABEL EVERYTHING you'll likely want to reference it later * code chunks that produce figures can be referenced via `@\ref(fig:[LABEL])`

You can label chapter and section titles using `{#label}` after them, e.g., we can reference Chapter 1. If you do not manually label them, there will be automatic labels anyway, * No idea how the automatic references work, so always be sure to declare them. * **NOTE** these display as the relevant Chapter `numeral`.

4.3 Syntax

italics or *italics* (can handle spaces) **bold** `code` *equations*

4.3.1 Math

Randal Pruim features an extensive list of common math expression on their github page. Here are some quick notes:

In-line equations can be written within `$` and will be displayed right there: $a^2 + b^2 = c^2$. In contrast, you can also add equation chunks by using `$$`

This can be coded in-line,

$$\sum_{n=1}^{10} n^2$$

, but will result in a page break.

Alternatively, a more "classic" equation chunk:

$$ Plain text doesnt get spaces

how

very

odd

$$

To compile this example to PDF, you need XeLaTeX. You are recommended to install TinyTeX (which includes XeLaTeX): https://yihui.name/tinytex/.

Bibliography

Xie, Y. (2015). *Dynamic Documents with R and knitr*. Chapman and Hall/CRC, Boca Raton, Florida, 2nd edition. ISBN 978-1498716963.

Xie, Y. (2022). *bookdown: Authoring Books and Technical Documents with R Markdown*. R package version 0.27.