Applied Data Analysis for Chinese Lending Data Using R & LLMs

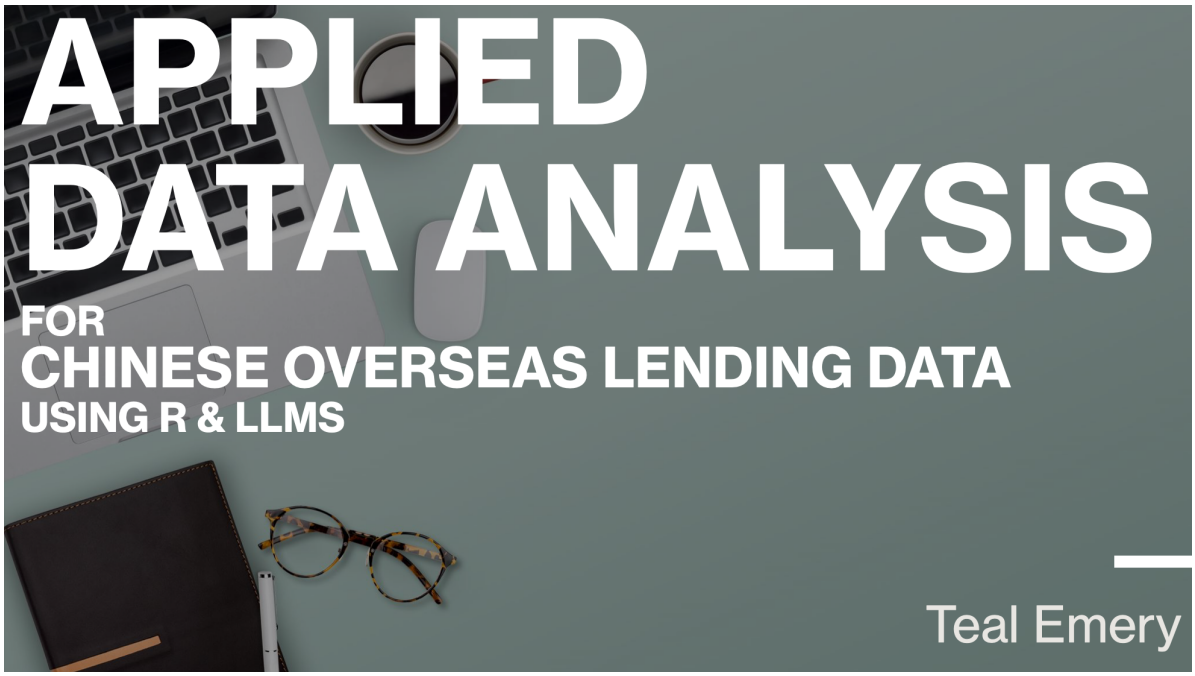
Teal Emery

2025-01-14

Table of contents

Preface

This textbook accompanies the intensive month-long course ***Applied Data Analysis For Chinese Overseas Lending Data Using R & LLMs*** developed for AidData research analysts. The course combines four weekly 90-minute online sessions with a full day of in-person instruction, designed to equip analysts with powerful new tools for data analysis, visualization, and automation.



Our Promise to You

By the end of this course, you will be able to:

- Transform complex datasets into compelling visual stories
- Automate those repetitive tasks that consume hours of your week
- Create analyses that update automatically when new data arrives
- Generate professional reports that combine narrative, code, and visuals
- Leverage AI tools to enhance your analytical capabilities

- Solve common data challenges more efficiently

Our Teaching Philosophy

Think of R not as a programming language to master, but as a powerful toolkit that helps you tell stories with data. Just as you don't need to be a mechanic to drive a car effectively, you don't need to be a programmer to use R powerfully. This course focuses on giving you practical tools that will immediately enhance your analytical capabilities.

We embrace two key advantages that make this possible:

1. **Modern Tools**: The tidyverse ecosystem transforms R from a statistical programming language into an intuitive data analysis toolkit
2. **AI Assistance**: Large Language Models (LLMs) act as your personal guide, helping you find and implement the right tools for each task

What Makes This Course Different

Traditional R courses often get bogged down in programming concepts before getting to practical applications. We flip this approach:

- Start with practical tools you can use immediately
- Focus on real AidData challenges and solutions
- Use AI tools to overcome technical hurdles
- Build from practical application to deeper understanding

Week-by-Week Journey

Week 1: First Steps with R

- Set up R & R Studio on your computer
- Create your first data visualization
- Learn to use AI tools for coding assistance
- Begin working with Quarto for reproducible reports

Week 2: Data Visualization Mastery

- Create publication-ready plots with ggplot2
- Master the grammar of graphics
- Build interactive visualizations
- Design effective data presentations

Week 3: Data Transformation

- Clean and reshape real development finance data
- Master key data manipulation verbs
- Replicate analyses from AidData reports
- Automate repetitive data tasks

Week 4: Advanced Topics

- Handle complex data cleaning challenges
- Create reproducible workflows
- Generate automated reports
- Build functions for common tasks

In-Person Session

- Work on your own projects with expert guidance
- Tackle advanced visualization challenges
- Learn to extract structured data from text using AI
- Build confidence through hands-on practice

Being Realistic About Learning

You won't become an R expert in four weeks—and that's okay. What you will achieve:

- Master enough R to make your daily work easier and more efficient
- Get past the steepest part of the learning curve
- Build confidence in your ability to learn more
- Develop a foundation for continued learning

Your AI Learning Assistant

Learning R in 2025 is fundamentally different from even a few years ago. Modern AI tools serve as 24/7 tutors that can:

- Explain complex code in plain English
- Help debug your problems
- Suggest improvements to your code
- Answer your questions any time

Think of these AI tools as having a knowledgeable colleague[1] always ready to help—they won't do the work for you, but they'll help you learn faster and overcome obstacles more efficiently.

How to Use This Book

This book serves multiple purposes:

1. A reference during the course
2. A guide for self-paced learning
3. A resource for future consultation

Each chapter includes:

- Clear learning objectives
- Practical examples using real data
- Exercises to reinforce learning
- Tips for using AI tools effectively
- Resources for deeper learning

Let's begin this journey together. By the end of the course, you'll have new tools and skills to analyze Chinese development finance data more effectively and efficiently than ever before.

[1] Sometimes that knowledgeable colleague is overconfident and incorrect. Use your human judgment.

1 Introduction

This is a book created from markdown and executable code.

See Knuth (1984) for additional discussion of literate programming.

```
1 + 1
```

```
[1] 2
```

2 Pre-Class Preparation

2.1 Overview

Before our first class meeting, you'll need to install some software and familiarize yourself with a few basic concepts. This preparation will ensure you can participate fully in class activities. While the steps are straightforward, please allow 45-60 minutes to complete everything comfortably.

2.2 Learning Objectives

By completing this pre-class work, you will be able to:

- Install R and RStudio on your computer
- Explain the difference between R and RStudio
- Perform basic calculations in R
- Create simple variables
- Use basic R functions
- Use AI tools to help understand R code

2.3 What is R?

Think of R as two things working together: a powerful calculator designed specifically for data analysis, and a collection of tools that make that calculator more useful. R was created by statisticians in the 1990s to make data analysis more accessible and reproducible. Today, it's one of the most popular tools for data analysis worldwide.

2.4 What is RStudio?

RStudio is like a workshop for R – it's where you'll actually do your work. If R is a powerful calculator, RStudio is the desk, notepad, file organizer, and reference library that makes using that calculator much easier. It's called an IDE (Integrated Development Environment), but you can think of it as your data analysis workspace.

2.5 What are R Packages?

Packages in R are like apps on your phone:

- Your phone comes with some basic apps (like R's built-in functions)
- You can install new apps (packages) to do specific tasks
- Once installed, you need to open (load) an app to use it

2.6 What is the tidyverse?

The tidyverse is like a bundle of the most useful data analysis apps, all designed to work together seamlessly. It includes tools for:

- Creating beautiful visualizations
- Cleaning and organizing data
- Importing data from various sources

These tools are designed to be more intuitive and user-friendly than base R.

2.7 Installation Steps

2.7.1 1. Install R First

1. Go to CRAN
2. Click on your operating system
3. Download and run the installer

2.7.2 2. Install RStudio Second

1. Go to RStudio Download
2. Download and run the installer

2.7.3 Need More Help?

If you run into any issues, you have two great options:

1. Try an interactive tutorial
    - This will walk you through each step with detailed instructions
    - Includes screenshots and troubleshooting tips
2. Ask a friendly LLM
    - Use ChatGPT or Claude
    - Try questions like: "I'm having trouble installing R on [your OS]. Here's what I've tried..."
    - The LLM can provide customized help for your specific situation

2.7.4 Verify Your Installation

After installing both R and RStudio:

1. Open RStudio (not R)
2. Type `2 + 2` in the Console (bottom left)
3. Press Enter

If you see `[1] 4`, you're ready to go!

If not, ask your favorite LLM. Describe what your issue. Upload a screenshot if you are unsure.

2.8 The RStudio Interface

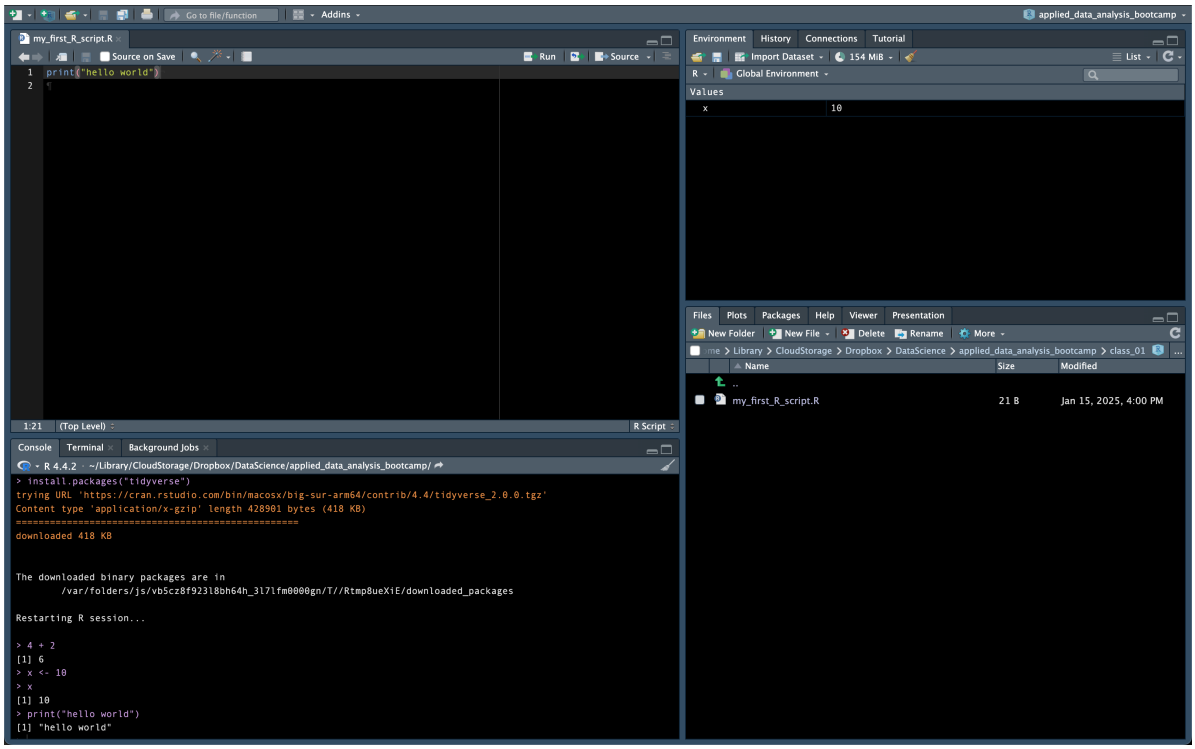


Figure 2.1: R Studio

When you first open RStudio, you'll likely see three panels. To see all four panels, click the "New File" icon in the top left corner and select "R Script". Here's what each panel does:

1. **Source Editor** (top left): This is where you write and edit your R code files
    - Like a text editor for your R scripts
    - Where you'll write code you want to save and reuse
2. **Console** (bottom left): This is where you run R commands
    - Think of it as R's command center
    - Where you can try out code immediately
3. **Environment/History** (top right): Shows your active variables and command history
    - Environment tab lists all variables you've created
    - History tab shows commands you've run
4. **Files/Plots/Packages/Help** (bottom right): A multi-purpose viewing area

- Browse your files
- View plots and visualizations
- Manage R packages
- Access help documentation

> **Tip**
>
> Want to learn more about RStudio's features? Check out the RStudio IDE Cheat Sheet.

2.9 Basic R Console Practice

Once you have R and RStudio installed, try these commands in your RStudio console:

2.9.1 R as a calculator

```
2 * 3 + 4
```

```
[1] 10
```

```
10 / 2
```

```
[1] 5
```

```
2^3   # This means 2 to the power of 3
```

```
[1] 8
```

```
### Creating variables (we call this "assignment")
x <- 10    # The arrow means "assign 10 to x"
y <- 5
x + y
```

```
[1] 15
```

```r
### Using functions
sum(1, 2, 3, 4, 5)
```

```
[1] 15
```

```r
mean(c(1, 2, 3, 4, 5))    # c() combines numbers into a list
```

```
[1] 3
```

2.10 Installing Your First Package: The Tidyverse

Just as installing Microsoft Office gives you a whole suite of programs (Word, Excel, PowerPoint) at once, installing the tidyverse gives you a collection of R packages designed to work together seamlessly for data analysis. Let's install it:

1. Type this command in your console:

```r
install.packages("tidyverse")
```

You'll see quite a bit of text appear as R downloads and installs multiple packages. This is normal! The tidyverse includes packages for:

Making plots (ggplot2) Working with data (dplyr) Reading data files (readr) And several others we'll use throughout the course

After installation completes, load the tidyverse:

```r
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.5.1     v tibble    3.2.1
v lubridate 1.9.4     v tidyr     1.3.1
v purrr     1.0.2
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becor
```

You'll see some messages about which packages were loaded. Don't worry about understanding all of them now - we'll learn about each one as we need it. ::: {.callout-note} You only need to install a package once on your computer (like installing Microsoft Office), but you need to load it with library() each time you start R (like opening Excel when you want to use it). :::

2.11 Try Using AI to Learn R

AI can be your extra-attentive tutor. Here's a piece of code to ask an LLM about:

```
flights |>
  group_by(carrier) |>
  summarize(
    avg_delay = mean(dep_delay, na.rm = TRUE),
    n = n()
  ) |>
  arrange(desc(avg_delay))
```

Try this prompt with ChatGPT or Claude:

> "I'm new to R. Can you explain what each line of this code does? Please explain it like you're talking to someone who has never programmed before."

LLMs' coding ability improves rapidly, so it's worth using the frontier models. If you don't already have access to the paid version of either Claude or ChatGPT, it is $20 a month well spent[1].

2.12 Resources for Learning More

Here are some excellent resources if you want to learn more:

- R for Data Science (2e) - The best comprehensive introduction
- RStudio Primers - Interactive tutorials
- Modern Data Science with R - More academic approach
- R-Ladies Community - Supportive learning community
- #rstats on Bluesky - Active community sharing tips

[1] As of the time of writing in January of 2025, Claude is the best for R, but ChatGPT 4o (and above) aren't too far behind. This will change quickly as new models come out.

2.13 Success Checklist

Before coming to class, you should be able to:

- ☐ Open RStudio
- ☐ Perform a calculation in the console
- ☐ Create a variable
- ☐ Use a basic function
- ☐ Ask an AI to explain R code

3 Summary

In summary, this book has no content whatsoever.

```
1 + 1
```

```
[1] 2
```

References

Knuth, Donald E. 1984. "Literate Programming." *Comput. J.* 27 (2): 97–111. https://doi.org/10.1093/comjnl/27.2.97.