Print to PDF

Introduction

Contents

- What is coding?
- Why should economists learn to code?
- Reading guide
- Python for Data Science (py4ds)
- Acknowledgements

DOI 10.5281/zenodo.10865836 release v1.0.1

Welcome to the 1st edition of **Coding for Economists**, a guide for economists on what programming is, why it's useful, and how to do it.

The book aims to give you the skills you need to code for economics, while also giving you tips and tricks about programming more generally that might be useful to you. If you're a complete beginner at coding, we have designed the first chapters to help you get started from scratch. But if you've dabbled before, or even if you're a sophisticated coder, there's likely to be material here relevant to you too.

Naturally, at times, we have made choices regarding what to include, what to omit, and what to recommend. We have given recommendations based on what will serve you best in the long-run. Very occasionally, like the Karate Kid painting Mr Miyagi's fence, you will wonder why such and such a thing is useful for coding in an economics context. But, also like the Karate Kid, you will find that you weren't *just* learning to paint a fence, you were becoming a karate (coding) master.

What is coding?

Computer programming, or coding, is the process of issuing a series of commands that a computer can understand and execute in order to perform a task. Today, almost all research with some quantitative aspect involves coding.

Imagine a computer as being like a particularly mischievous genie; if your instructions aren't completely unambiguous, the computer won't do what you asked for. So we write out our code, our list of instructions for the computer, very explicitly to make sure *we* can check we wrote what we meant, and that the computer will do it in the way we anticipate. Like a genie though, if you can get the instructions just so, you can create magical things.

Why should economists learn to code?

Given so much of economics is quantitative, coding is an essential skill for many economists. Many of the tools and packages that have been developed will help you to do better work more productively. It's Pareto improving.

As a skill, programming is enormously valuable *beyond economics* too because it's used across a wide range of domains (this is especially true for general purpose programming languages). Code is widely used across industry, academia, and the public sector in everything from computer games to websites, data science to software applications. And learning basic programming concepts in any language is useful for almost any other programming language. So learning to code is good for your human capital.

Coding is like a superpower; it's both fun and powerful!

Reading guide

You can read this book in any order depending on your experience and need. If you want to start getting to grips with using code, the first four sections are designed to help you get started quickly while also giving you a solid foundation.

Skip to main content

The rest of the book is split into sections that present topics but differentiate between the level at which they address them; for example, there's "Essential Coding" and "Coding". The idea of these demarcations is to help guide you as to what is accessible and essential versus what is useful but ultimately not strictly necessary. It also points to a possible order for you to arrange your learning by. Of course, the needs of people reading this book will be very heterogeneous, so you should absolutely dip into whatever chapter is most useful to you; likewise use whatever order makes sense to you. Do also note that just because a chapter doesn't appear in an "essential" section, doesn't it mean it's not useful! Although not in an "essential" section, having made use of the concept of a reproducible analytical pipeline (in Reproducible Analysis) in anger in research, we cannot now imagine any other way of working with code in economics.

Python for Data Science (py4ds)

If you're looking for a more narrowly focused and beginner-friendly introduction to *data science*, with less emphasis on research, you might want to check out our sister book *Python for Data Science* (click the link or search "py4ds" on Google.)

Acknowledgements

Contributors:

- Arthur Turrell, who has also contributed to *Python for Data Science* and wrote popular non-fiction book *The Star Builders*.
- lukestein
- Artem Smirnov, who did a wonderful job of improving the figures made using the **Lets-Plot** package.
- Alexander Fischer, who tailored his wonderful **pyfixest** package to be more helpful to the readers of *Coding for Economists*. Do go and give **pyfixest** a star too!
- Igor Alshannikov
- econometrickk
- Andy Banks
- Zeki Akyol

Please note that this book does not represent the views of any employer of its contributors, and the creators accept no liability for any mistakes. We are delighted to accept contributors: if you'd like to help make the book even better, see Contributing.

We hope you enjoy *Coding for Economists*!

Next
Preliminaries ›