Research Software Engineering with Python

- Damien Irving
- Kate Hertweck
- Luke Johnston
- Sara Mahallati
- Joel Ostblom
- Charlotte Wickham
- Greg Wilson

1. Aims

Software is now as essential to research as telescopes, test tubes, and reference libraries, which means that researchers now need to know how to build, check, use, and share programs. However, most introductions to programming focus on developing commercial applications, not on exploring problems whose answers aren't yet known. Our goal is show readers how to do that, both on their own and as part of a team.

We believe every researcher should know how to write short programs that analyze data in a reproducible way, and how to use version control to keep track of what they have done. But just as some astronomers spend their careers designing telescopes, some researchers focus on building the software that makes research possible. People who do this are called *research software engineers*, and the aim of this book is to get you ready for this role, i.e., to help you go from writing code for yourself to creating tools to help your entire field advance.

2. Synopsis

This book introduces the core skills researchers need to develop robust software that others can use, to share their work with others, and to be productive as part of a research software team. We assume readers have some basic programming knowledge, and build on that to cover:

- using the Unix shell to manage work and make it repeatable
- effective use of version control
- building reusable software tools
- automating workflows
- configuring software
- testing and error handling
- creating productive, inclusive teams
- documenting, packaging, and releasing software for use by the wider research community

Our book is a ready-to-go university semester course. All of this material has been used and refined in workshops, some of it by multiple instructors over many years. Please see the end of this proposal for a detailed table of contents.

3. Target Market

At the beginning of the book, we introduce three personas who characterize our audience:

Amira Khan completed a master's in library science five years ago and has since worked for a small aid organization. She did some statistics during her degree, and has learned some R and Python by doing data science courses online, but has no formal training in programming. Amira would like to tidy up the scripts, data sets, and reports she has created in order to share them with her colleagues. These lessons will show her how to do this and what "done" looks like.

Jun Hsu completed a 4-month data science bootcamp last year after doing a PhD in Geology and now works for a company that does forensic audits. He uses a variety of machine learning and visualization packages, and would now like to turn some of his own work into an open source project. This book will show him how such a project should be organized and how to encourage people to contribute to it.

Sami Virtanen became a competent programmer during a bachelor's degree in applied math and was then hired by the university's research computing center. The kinds of applications they are being asked to support have shifted from fluid dynamics to data analysis; this guide will teach them how to build and run data pipelines so that they can pass those skills on to their users.

4. Competing Titles

The Turing Way (https://the-turing-way.netlify.app/), produced by the Turing Institute in the UK, is the only up-to-date resource we know of with similar breadth. Other books cover only one topic in more depth than our audience needs (e.g., Ray & Ray's *Unix and Linux: Visual QuickStart Guide*), are aimed at commercial software developers (any number of books on testing, coding style, and dev ops), or go into data science rather than telling readers how to build software (e.g., Zhang's upcoming *A Tour of Data Science*).

Our book has broader and deeper coverage, and includes exercises with solutions. Our focus on programming best practices (e.g. principles like writing modular, reusable, testable code). Others teach basic Python and R syntax and how to complete isolated tasks with particular libraries, but today, people can pick up that information from a Google search. Our content is the bigger picture you can't glean from online code snippets or Stack Overflow answers.

We go through the process of building a data science workflow/package from scratch. Most other texts simply present information (like a reference book): we tackle a real data science problem from beginning to end, weaving in relevant information as we go.

5. Format and Timeline

Our manuscript is approximately 470 pages long, with 40 diagrams and other graphics and 150 code listing.

6. Other Information

- Dr. Damien B. Irving: post-doctoral researcher in climate science at the University of New South Wales. https://damienirving.github.io/
- Dr. Kate L. Hertweck: bioinformatics training manager at the Fred Hutchinson Cancer Research Center. http://katehertweck.com/
- Dr. Luke Johnston: post-doctoral researcher in bioinformatics at Aarhus University. http://lukewjohnston.com/
- Sara Mahallati: graduate student in biomedical engineering at the University of Toronto. https://scholar.google.ca/citations?user=8zCoa-UAAAAJ
- Joel Ostblom: graduate student in biomedical engineering at the University of Toronto. https://joelostblom.com/
- Dr. Charlotte Wickham: a data scientist and instructor at Oregon State University. https://www.cwick.co.nz/
- Dr. Gregory V. Wilson: co-founder of Software Carpentry, now part of the education group at RStudio. http://third-bit.com/

Table of Contents

The current table of contents is given below. Please note that we are currently planning to divide the material in Chapter 15 (Publishing) among the other chapters so that the book will build toward creating and distributing a reusable research software package. Please also note that we have started work on a companion volume that uses R instead of Python for examples. If this proposal is accepted, we expect to be able to complete the second book some time in 2021.