Reproducible Research: Automating Your Science

Robert M. Porsch

Overview

Most researchers spend hours making tables and figures to accurately and concisely represent their results. However, this process is often repetitive and must often be repeated when the underlying statistical model or data has changed. This workshop covers how to automate these steps into a coherent data analysis pipeline with dynamic documents while keeping track of a detailed version control, allowing scientists to spend more time on their research.

An analysis pipeline automates the creation of all tables, figures, statistical models and data management tasks into a single command. For example, a reviewer of a recently submitted manuscript suggests using an additional covariate in statistical analysis. Without automation, one has to redo the statistical analysis, generate new tables and figures, and finally modify the corresponding text in the original manuscript. In contrast, an automated data analysis pipeline streamlines this process. One will simply include the new additional variable into the model, and all following tables and figures will automatically react and change accordingly, without any further human interference. Furthermore, it allows the original manuscript to react dynamically to changes in the data and models, saving time and reducing the chances of introducing errors.

The other aspect of this workshop concerns version control, or the management of changes to data analysis and manuscripts. The path from data analysis to the submission of a manuscript is a long one and takes many weeks of exploring the data, testing different hypotheses, and inspecting different aspects of the study. Version control systems, such as git, aim to manage different versions of the same project, thus enabling a coherent handling of different analysis. These systems have a number of benefits for the researcher. For example, such a system allows a scholar to easily compare past and current versions of the results, analysis steps, and manuscripts. It further enables one to maintain multiple versions of the same project and easily merge those versions into a single coherent project. In conclusion, version control enables scientists to keep a detailed record of their work, enabling a flexible handling of multiple versions of the same data analysis as well as the manuscript.

Learning Objectives

This first half of the workshop aims to teach the basics of version control and dynamic documents. We will introduce the basics of version control and how it can be used in a data analysis project. Furthermore, the basics of data analysis pipelines will be introduced. Then, we will make use of the popular version control system, git, to have some practical experience. This includes maintaining and managing multiple versions of the same project simultaneously. Furthermore, the course will cover how to use git to work effectively with other scholars.

The second half of the workshop will concentrate on dynamic documents and analysis pipelines. It will cover in detail how to write complex dynamic documentation for data analysis with the help of RMarkdown for Word, HTML, and PDF. We will go from a raw data file to a publication ready manuscript, while automating all analysis steps and figure and table production. The course will outline, using a real world example, how to use simple scripting to generate reproducible plots and figures. We will further describe and handle potential problems and offer solutions.

Lastly, I will standardize the taught data analysis pipeline for future projects and share them with other scholars. At the end of this course, participants should be able to write their own data analysis pipeline and generate dynamic documents.

Pre-workshop Survey

Participants will be asked to complete a pre-workshop survey in order to answer some basic questions in regard to their previous experience with data analysis. This includes which statistical software tools they have used, programming experience, current forms of data storage and some other basic questions.

Outline of the workshop

The full-day workshop is separated into two different modules, **version control** and **dynamic documents**. Each module will be divided into multiple submodules, each of those containing task oriented practical exercises. Tasks are either performed individually, in pairs or in groups.

Module 1 (3 hours): Version Control and Reproducible Research

The first part of this module will give a general introduction to the topic. On the basis of famous irreproducible results, the importance of open software, data and analysis protocols will be demonstrated. In general, I will first discuss what version control is and how it is helpful to individual scientists as well as science in general. Next, the basics of analysis pipelines will be introduced and how common tasks can be automatised.

Within the practical part of this module I will construct a simple analysis pipeline on a given dataset. During the development of this pipeline I will make use of the popular version control system git in order to maintain multiple versions of the analysis protocols, work with previous versions of the same analysis, as well as collaborate with multiple parties on the same project effectively.

Module 2 (3 hours): Automation and Dynamic Documents

The main aim of this module is to extend data analysis pipelines to dynamic documents. Within this module collaborative exercises will be used to teach participants how to build complex, multi-staged and testable analysis pipeline as well as how to integrate those into dynamic documents. Specifically, participants will be asked to completely automatise the analysis of a given dataset, from data cleaning, to modeling, until the completion of the final analysis report. A special focus will be put on tables and figures. In the end of this module, participants should be able to write a full paper as a dynamic document, automating various tasks.

Requirements

Participants will be asked to bring their laptop. In addition, the following software needs to be installed:

- RStudio: https://www.rstudio.com/products/rstudio/download/
- git: https://git-scm.com/downloads