Preparation workshop Zurich 14. - 16. July

Software installation and else

Sonja Hartnack Valerie Hungerbühler

2021-07-11

Overview

This document (thanks to Matt Denwood and Giles Innocent, I benefitted from their document prepared for an earlier workshop) will help you to prepare for the HARMONY Training workshop "Bayesian Latent Class Models to evaluate diagnostic tests in the absence of a gold standard" held in Zurich 14-16 July 2021. The workshop will be oriented around practical exercises, so it is very important that you bring a laptop with the necessary software installed. You should also be familiar enough with R that you can use it for basic tasks, although we will try to keep the programming to a minimum during the workshop. You will also need to download the teaching material from our GitHub site, so you might find it helpful to have created an account with GitHub before the course so that you can access the material more easily during the week. The purpose of this document is to ensure that you have the necessary software installed, access to the GitHub repository where the teaching material will be made available, and to give you a brief refresher in the basics of Bayesian statistics.

Software installation

You need to install R (the current version is 4.1.0) from https://cran.r-project.org/ and we recommend that you also use Rstudio which can be downloaded separately from https://www.rstudio.com/products/rstudio/download/

Please also install the latest versions of the following R packages: PriorGen, rjags, runjags, coda, TeachingDemos, MCMCpack, mada

You will also need the standalone JAGS software (version 4.3.0 or later) for the course - download the installer for your platform from here: https://sourceforge.net/projects/mcmc-jags/files/JAGS/4.x/

To check that you have installed the software correctly please run the following code within R (or RStudio) and make sure that no errors are produced:

```
stopifnot(getRversion() >= "4.0.0")
stopifnot(require('PriorGen'))
stopifnot(require('rjags'))
stopifnot(require('runjags'))
stopifnot(require('coda'))
stopifnot(require('TeachingDemos'))
stopifnot(require('MCMCpack'))
stopifnot(require('mada'))
stopifnot(testjags()$JAGS.available)
stopifnot(numeric_version(testjags()$JAGS.version) >= "4.3.0")
stopifnot(testjags()$rjags.found)
stopifnot(numeric_version(testjags()$rjags.version) >= "4-8")
```

GitHub

Basics

GitHub is an online code repository that in it's most basic form stores the version history of any code project you are working on, and allows you to go back in time to a previous version before someone introduced a bug. It is particularly useful when collaborating with others because it allows you to see who made a change, when it was made, and what the code looked like before that. It also allows changes from different people to be merged into the same central repository to ensure that nobody gets out of sync with everybody else's code.

We will primarily be using GitHub as a way to disseminate the lecture notes and R/JAGS code for the exercises on course, so you only need to use the most basic features of GitHub (but it is a good thing to learn).

Simple Web Usage

We have created a public repository containing the teaching material for the training workshop. This means that anyone can view the teaching material at any time via the following website: https://github.com/shartn/COST.BLCM.Zurich.git

You should see a number of files (we will add files during the course) in an .Rmd and an .pdf format. You can click on any of these files to view them, although of the different file formats the .pdf version is probably easiest to download directly from the website. There are two problems with this:

1 - You will likely encounter problems when copy/pasting R code from the PDF files. Here it is better to use the .Rmd file.

2 - If/when any of the files are updated, you will not get an automatic update of the new version.

We therefore recommend that you follow the instructions below to clone the repository directly to your computer.

Creating an Account

If you have never used GitHub before, you should create an account via http://github.com - it is free and easy. Remember to make a note of the username and password that you just created!

GitHub Desktop

The easiest way to use GitHub is via GitHub desktop. Go to https://desktop.github.com and download/install the software. Then open it and sign in using your GitHub account (the username and password that you just created). This should become the primary way in which you interact with GitHub, rather than via your browser.

The first step is to clone the `COST.BLCM.Zurich` repository on to your computer. Click on the green button `code`. Select "Open with GitHub Desktop". A suggested local path folder will be created automatically at the bottom but feel free to change this. Then click on Clone and wait for the files to download.

Creating the clone copies all of the files currently on the GitHub repository to your computer at the local path specified above (if you can't remember what this was choose 'Show in Windows/Finder' or similar, depending on your OS, under the Respository menu). This is where all of the course material will (eventually) be located. For example, open the `'Preparation.Rmd'` file - that is a local copy of the same document you are currently reading! But now you also have the PDF version - use this if you want to print the document for some reason, but it is a good idea to stick to using the Rmd version if you want to copy/paste R code (you will probably encounter problems with quotation marks if you copy/paste R code from PDF files). The two versions should be identical.

Modifying Files

Once you have set up the local copy of the repository, you can then add, delete and modify any of the files within your local copy of the repository. Try doing this by deleting the `'Preparation.Rmd'` file. Now go back to GitHub Desktop where you should see a line appear on the left with a red minus symbol in a box to the right hand side - this is telling you that a file has been deleted locally (if you had modified the file rather than deleting it, the box would be orange and contain a dot). However, you don't want to delete or modify any of the files we are providing in case we update them later. If you do this by mistake, just right-click the relevant line in GitHub desktop and choose "Discard changes" - the file should

then be restored. If you do want to modify a file for some reason, we suggest that you copy it and modify the copied version. If you keep the copy inside the same folder then GitHub desktop will show the new file (green plus in the box) but you can safely ignore these, or move the copied file outside of the repository folder if you want to keep things simple.

Fetching Updates

An important advantage of GitHub is that we can update a file online (for example to fix a typo or add new material discussed during the workshop), and that update will be immediately available to all of you. But in order to receive the updated files you need to tell GitHub desktop to look for updates. Open GitHub desktop, then click on 'Fetch origin' and then 'Pull' to make sure that any recent changes are synced to your computer. As long as you remember to pull changes regularly then you will always have an up-to-date copy - so do this regularly. Forgetting to do this is the only real potential downside of Git.

More information

That is pretty much all you need to know for the training workshop but there are some good tutorials available, for example: https://happygitwithr.com.