Installing Simulation Software

Jinko Graham

11/04/2019

- NOTE 1: This document gives instructions on installing software. There is a second document called testInst.Rmd that you can use to test the installed software.

- NOTE 2: Before working through this installation document, you must copy, or "clone" the class GitHub repository to your computer. The simplest way to do this is to create an RStudio project linked to the git repository https://github.com/SFUStatgen/SFUStat840. See the instructions on the class Canvas page on Computing: https://canvas.sfu.ca/courses/47316/pages/computing for instructions on getting started with git and version control in RStudio.

1. Overview

We will work with simulation software called `msprime` and `SLiM`. `msprime` simulates ancestries of chromosomes according to a backward-in-time model called the coalescent that we will study in class. `msprime` is a Python program that we will interface with through R, using the R package `reticulate`. `SLiM` simulates genomic data forwards-in-time using a Wright-Fisher reproductive model that we will also study in class. `SLiM` is a C program that can be run from either the command line or a graphical user interface. Both `msprime` and `SLiM` return a data structure called a succinct tree sequence (STS) that represents ancestral relationships amongst chromosomes. We will use the Python package `tskit` to work with STSs. We will use the Python package `pyslim` to import `SLiM` STSs into Python for processing by `tskit`. The majority of this document contains instructions for installing Python and the packages `msprime`, `tskit` and `pyslim`. At the end is a short section describing how to install the C program `SLiM`. The C program `SLiM` needs to be installed separately from the Python packages, with an installer program that is downloaded from the developer's website.

2. Python and Python packages

We will be using the R package **reticulate** to interface between R and Python. Specifically, we will use `reticulate` to install the Python packages `msprime`,`tskit` and `pyslim`. The instructions in this section will lead you through the installation of Python, `reticulate`, `msprime`, `tskit` and `pyslim`.

2.1. Installing Python and conda

Conda is a package-management system used by the Python community. Python and conda are installed simultaneously by a program called miniconda, which you can find at https://docs.conda.io/en/latest/miniconda.html

Choose the installer for Python version 3.7 that is appropriate for your operating system and follow the instructions to install. [Note for Mac users: You can choose either a "bash installer" or a ".pkg installer". The .pkg installer may be more familiar to most Mac users. Download the .pkg file to your computer, double-click to start the installer and you will be guided through the installation.]

- NB: The installation of conda changes the configuration of the shell used in your terminal windows. You need to open a *new* terminal after the installation to be able to use conda. From a new terminal, type `which python` and `which conda` and record the paths to these executables that were installed by miniconda (you will need them in the next step). On my Mac the paths are `/Users/jgraham/opt/miniconda3/bin/python` and `/Users/jgraham/opt/miniconda3/bin/conda`.

2.2. Installing the R package `reticulate`

As an R package, `reticulate` and can be installed from CRAN in the usual way. After installing, use `library()` to load the package and the function `use_python()` to set the path to your Python executable, as in the following R code chunk:

```r
library(reticulate)
use_python("~/opt/miniconda3/bin/python") # change if necessary
```

2.3. Installing `msprime` and `tskit` into a Python "environment" from R

Python environments are like RStudio projects in that they are meant to encapsulate all code and data for a specific project. However, they take the encapsulation idea a step further than R and also include any Python packages that you need for the project. In the following, I create an environment called `Stat840` and install `msprime` and `tskit` into it with `conda_install()`. Since `msprime` depends on `tskit` (in much the same way as R packages can depend on each other), `tskit` will automatically install when we install `msprime`. To use the new `Stat840` environment, we call `use_condaenv()`. The option `required=TRUE` of `use_condaenv()` ensures that we are using the right version of Python.

```r
envname <- "Stat840"
conda <- "~/opt/miniconda3/bin/conda" # change if necessary
conda_install(envname,"msprime",conda=conda)
use_condaenv("Stat840",required=TRUE,conda=conda)
```

2.4. Installing `pyslim` from R

Unfortunately `pyslim` is not available on the `conda-forge` website that hosts conda-installable packages. However, the class has its own conda-installable version of `pyslim` along with a bespoke function `conda_install_local()` to simplify its installation from within R. Both the package and R function are in the class GitHub repository (https://github.com/SFUStatgen/SFUStat840). Before proceeding, you **must** have a copy of the repository on your computer. In what follows, replace `/path/to/SFUStat840` with the **full** path to the copy of the class repository on your computer. Note that the tilde shortcut ~ doesn't work in R and so you have to specify the full path.

```r
pkgname <- "pyslim"
dir <- "/path/to/SFUStat840/PythonPackages/"
source(paste0(dir,"conda_install_local.R"))
conda_install_local(envname,pkgname,dir,conda)
```

2.5. Using the Stat840 Python environment in future R sessions

To use the `Stat840` Python environment and all the packages that you've installed into it, include the following code chunk at the top of your RMarkdown documents.

```r
library(reticulate)
conda <- "~/opt/miniconda3/bin/conda"
use_condaenv("Stat840",required=TRUE,conda=conda)
```

3. Installing `SLiM`

SLiM is a C program available from https://messerlab.org/slim/

Unfortunately the program is only available for Mac OS X and Linux users. As a Mac user, I haven't installed SLiM on Linux. I know some of you are keen to try installing on Linux and so please help eachother out. If you are a Mac user, download and run the OS X installer from the above website. This should install both a command-line version

of SLiM (e.g., in /usr/local/bin/slim) and a graphical-user-interface (GUI) version of SLiM (e.g., SLiMgui in your Applications folder).

If you do not have access to a Mac and can't get SLiM running on Linux, you are welcome to use the Mac Pro housed in Brad McNeney's office. To use this server, all you have to do is log in from the console with your SFU ID the first time. This takes 30 seconds, so stop by Brad's office in K10565 immediately after class on Monday or Thursday. After this initial login from the console, you will be able to log in remotely. You will not be able to use the SLiM GUI when you log in remotely.