How to install R on mac

Sigurður Haukur Birgisson

How to Install R on Mac

Prerequisites

First check if Xcode is installed by running:

```
xcode-select -p
```

If it is not installed, you will need to install it.

To install Xcode command line tools by running the following command in the terminal:

```
sudo xcode-select --install
```

Then install XQuartz from http://xquartz.macosforge.org/, look for a link to a file ending with `.pkg` (e.g. `XQuartz-2.8.5.pkg`). This is needed for some R packages that require X11. Remember to reinstall XQuartz after every macOS update.

Next, download and install the GNU Fortran compiler from https://mac.r-project.org/tools/, look for a link to a file ending with `.pkg` (e.g. `gfortran-14.2-universal.pkg`). Make sure to choose the version that matches your macOS version. This is necessary for compiling certain R packages.

Installing R and RStudio

Download and install R from CRAN at https://cran.r-project.org/bin/macosx/, look for a link to a file ending with `.pkg` (e.g. `R-4.5.1-arm64.pkg`). Choose the latest version available for your macOS, and be carefull to choose the correct version depending on if you have an Apple Silicone mac or Intell

After installing R, you can install RStudio by downloading it from https://posit.co/download/rstudio-desktop/. RStudio provides a user-friendly interface for R programming.

Getting Started

Once R and RStudio are installed, you can open RStudio and start using R. You can install additional R packages using the `install.packages("package_name")` command in the R console.

You can also run R scripts directly from the terminal by using the `Rscript` command followed by the script name:

```
Rscript your_script.R
```

Optional Shell Configurations

To make working with R more efficient, here are common aliases you can add to your shell configuration file (`.bashrc`, `.zshrc`):

To edit your `.zshrc` file, you can use:

```
nano ~/.zshrc  # or use your preferred text editor (e.g., vim, code)
```

Then add the following aliases:

```
# Lowercase R commands
alias r='R'
alias rscript='Rscript'

# Open RStudio from terminal
alias rstudio='open -a RStudio'

# Common R CMD commands
alias rrun='Rscript'
alias rcheck='R CMD check'
alias rinstall='R CMD INSTALL'
alias rbuild='R CMD build'
```

Remember to source your shell configuration file after adding the aliases:

```
source ~/.zshrc  # or source ~/.bashrc depending on your shell
```

Now, you can use these aliases to quickly access R and RStudio from the terminal. For example, you can open RStudio by simply typing `rstudio` in the terminal.

RStudio Configuration

Open RStudio and go to **Preferences** (on macOS) or **Tools > Global Options** (on Windows/Linux).

General Settings

Change the default working directory: In the **General** tab, set the **Default working directory** to your preferred folder where you want to save your R projects.

Code Editor

Enable vim keybindings: Go to the **Code** tab, and under **Editing**, select **Vim** from the **Keybindings** dropdown menu.

Enable code formatting on save: In the **Code** tab, select the **Formatting** sub-tab and select a code formatter like `styler` or `formatR`, then check the box for **Format code on save**.

Terminal

Select bash/zsh as the terminal shell: Go to the **Terminal** tab, and in the **New terminals open with** dropdown, select **bash** or **zsh**, depending on your preference.

Accessibility

Remove animations: Go to the **Accessibility** tab and check the box for **Reduce motion** to minimize animations in the RStudio interface.