

Find us on GitHub

<http://www.genome.duke.edu><https://github.com/Reproducible-Science-Curriculum/>

# Reproducible Science Workshop - Tools, Resources, and Practices

## Duke Marine Lab, Auditorium

September 24-25, 2015

9:00 am - 5:00 pm

**Instructors:** Mine Çetinkaya-Rundel (Duke, Dept. of Statistical Science (<https://stat.duke.edu/>)), Karen Cranston (Duke, Open Tree of Life (<http://opentreeoflife.org>)), Hilmar Lapp (Duke, GCB (<http://genome.duke.edu>)), Dan Leehr (Duke, GCB (<http://genome.duke.edu>))

## General Information

Making science more reproducible has the potential to advance scientific research and make researchers' work more effective and productive. For computational and data-intensive research, which is increasingly pervasive across the sciences, this is particularly true, and yet is often seen as difficult to achieve. In this 2-day bootcamp-style hands-on workshop, we will teach a number of tools, resources, and practices that can be used today to make one's computational science more reproducible.

The course and the curriculum were developed by the participants of the Reproducible Science Curriculum Hackathon (<https://github.com/Reproducible-Science-Curriculum/Reproducible-Science-Hackathon-Dec-08-2014>) held at the National Evolutionary Synthesis Center (NESCent (<http://nescent.org>)) in December 2014. The hackathon and instructor travel are supported by the National Science Foundation (NSF).

**Who:** The course is aimed at graduate students, postdocs, and other researchers who perform computational analysis or work. The material on automation uses basic R for teaching and illustrating the key concepts. Advanced knowledge of R is not needed, but some familiarity with R will make the workshop more enjoyable.

**Where:** 135 Duke Marine Lab Rd, Beaufort, NC 28516. Get directions with OpenStreetMap (<http://www.openstreetmap.org/?mlat=34.7169124&mlon=-76.6721571&zoom=16>) or Google Maps (<http://maps.google.com/maps?q=34.7169124,-76.6721571>).

**Requirements:** Participants must bring a laptop with a few specific software packages installed (listed below). They are also required to abide by our Code of Conduct (<http://software-carpentry.org/conduct.html>), which we have adopted from Software Carpentry.

The course is free but requires registration. We ask that as a courtesy to others you cancel as early as possible if you register and subsequently are prevented from taking your seat.

**Contact:** Please email [hilmar.lapp@duke.edu](mailto:hilmar.lapp@duke.edu) (<mailto:hilmar.lapp@duke.edu>) for more information.

---

## Schedule

### Day 1

---

09:00	Introduction to Reproducible Research
10:30	Coffee/Tea break
10:45	Organizing your project to facilitate Reproducible Research
12:00	Lunch break
13:00	Literate programming
15:00	Coffee/Tea break
15:30	Literate programming
17:00	Wrap-up

### Day 2

---

09:00	Version control
10:30	Coffee/Tea break
10:45	Automating your workflows
12:00	Lunch break
13:00	Automating your workflows
15:00	Coffee/Tea break
15:30	Sharing and publishing your research workflow
16:30	Wrap-up

**Etherpad:** <https://etherpad.mozilla.org/cwfShOVreq> (<https://etherpad.mozilla.org/cwfShOVreq>). We will use this Etherpad for chatting, taking notes, and sharing URLs and bits of code.

# Syllabus

## Introduction to Reproducible Research

- Recognize the problems that reproducible research helps address
- Identify pain points in getting your analysis to be reproducible.
- The role of documentation, sharing, automation, and organization in making your research more reproducible.
- Introducing some tools to solve these problems, specifically R/RStudio/RMarkdown.
- Slides:
  - Intro I slides (intro-slides/intro-01-slides.html)
  - Intro II slides (intro-slides/intro-02-slides.html)

## Organizing your project to facilitate Reproducible Research

- Organize projects and folders to enable reproducibility and reusability
- Understand the structure of data files and the importance of documenting all changes made
- Using these practices, create a reproducible project workflow using knitr in RStudio.
- Slides:
  - File organization (organization-slides/)
  - File naming (naming-slides/)

## Literate programming

- Understand the value of having question, source code, and result side by side
- Use literate programming to create executable documentation
- Create self-documenting data cleaning and quality control reports
- Slides:
  - Literate programming (lit-prog-slides/)

## Version control

- Exposure to the motivations of using version control.
- Learn the basics of Git via Git and Github.
- Slides:
  - Version control, Git, and Github (slides/01-github-slides.html)

## Automating your workflows

- Transforming scripts into functions
- Make files as a concept
- Utilizing free continuous integration tools
- Recreate figures automatically
- Slides:
  - Automation (slides/01-automation-slides.html)
  - Functions (slides/02-functions-slides.html)
  - Functions for data (slides/03-functions-for-data-slides.html)
  - Functions for figures (slides/04-functions-for-figures-slides.html)
  - Testing (slides/05-testing-slides.html)
  - Putting it all together (slides/06-automating-slides.html)

## Sharing and publishing your research workflow

- Ensuring your workflow is permanently archived and citable
  - Documentation and metadata for reusability
  - Choosing licenses for software and data
  - Slides:
    - Sharing, publishing, archiving, licensing (<slides/01-publication-slides.html>)
- 

## Materials

### Introduction

- Download (<https://github.com/Reproducible-Science-Curriculum/rr-intro/archive/data-v0.2.zip>)

### Organization

- Download (<https://github.com/Reproducible-Science-Curriculum/rr-organization1/archive/data-v0.4.zip>)

### Literate Programming

- Download (<https://github.com/Reproducible-Science-Curriculum/rr-literate-programming/archive/data-v0.4.zip>)

### Automation

- Download (<https://github.com/Reproducible-Science-Curriculum/rr-automation/archive/data-v0.2.zip>)

## Reproducibility checklist

A checklist (<checklist.html>) to evaluate and stimulate thoughts about the reproducibility of your project.

## Setup

### Git

Git is a version control system that lets you track who made changes to what when and has options for easily updating a shared or public version of your code on [github.com](https://github.com/) (<https://github.com/>). See the instructions below for your operating system. Windows and Mac Users (with OSX 10.9+) will install the GitHub GUI, while Linux and Mac users with older operating systems will install the command line utility.

If you don't already have a GitHub account, please create one (<https://github.com/join>).

### Windows

Please download the GitHub GUI here (<https://windows.github.com/>).

### Mac OS X

**For OS X 10.9 and higher**, install the GitHub GUI for Mac by downloading and running the installer from here (<https://mac.github.com/>). **For older versions of OS X (10.5-10.8)** use the most recent available installer labelled "snow-leopard" available here (<http://sourceforge.net/projects/git-osx-installer/files/>).

## Linux

If Git is not already available on your machine you can try to install it via your distro's package manager. For Debian/Ubuntu run `sudo apt-get install git` and for Fedora run `sudo yum install git`.

## R

R (<http://www.r-project.org>) is a programming language that is especially powerful for data exploration, visualization, and statistical analysis. To interact with R, we use RStudio (<http://www.rstudio.com/>). After installing both R and RStudio, start RStudio and install some packages that we will need for the workshop (see bottom of the page).

**There is brand new version of RStudio available as of May 26th (v0.99.441), make sure to update if you already have RStudio installed.**

## Windows

Install R by downloading and running this .exe file (<http://cran.r-project.org/bin/windows/base/release.htm>) from CRAN (<http://cran.r-project.org/index.html>). Also, please install the RStudio IDE (<http://www.rstudio.com/ide/download/desktop>).

## Mac OS X

Install R by downloading and running this .pkg file (<http://cran.r-project.org/bin/macosx/R-latest.pkg>) from CRAN (<http://cran.r-project.org/index.html>). Also, please install the RStudio IDE (<http://www.rstudio.com/ide/download/desktop>).

## Linux

You can download the binary files for your distribution from CRAN (<http://cran.r-project.org/index.html>). Or you can use your package manager (e.g. for Debian/Ubuntu run `sudo apt-get install r-base` and for Fedora run `sudo yum install R`). Also, please install the RStudio IDE (<http://www.rstudio.com/ide/download/desktop>).

## After installing R and RStudio

Start RStudio, and type (or copy and paste) at the console: `install.packages(c("knitr", "rmarkdown", "ggplot2", "dplyr"))`

---

We are using Software Carpentry (<https://software-carpentry.org>) workshop template (<https://github.com/swcarpentry/workshop-template>) for this website.

---

[License \(LICENSE.html\)](#)