**COURSE: Introduction to R (Part I)**

Instructor: Mason Murphy

TAs: Abby Valachovic and Keaka Farleigh

**COURSE OVERVIEW:** Part I of an Introduction to coding in R and RStudio, a coding language and environment highly used by biologists. No experience or background necessary in R or any coding language, and a great opportunity for review for those with basic R skills.

**LOGISTICS:**

Software Needed:

1. R and RStudio (will be teaching/working in RStudio, which requires installation of R)

* Download R: <https://cloud.r-project.org/>
* Download RStudio: <https://rstudio.com/products/rstudio/download/>
* I have a one-page document with instructions on how to install R and RStudio that I can provide in advance if needed

1. Packages “dplyr” and “tidyr”
   * Not vital to install before workshop, as discussing packages is part of the lesson plan
   * If not installed prior, however, participant will need internet connection to download during lesson (assume will have internet to connect over Zoom?)

Preparation Needed:

1. Install R, RStudio, and possibly packages on personal computers
2. Create pre-made scripts and datasets
   * Start with clean scripts, add more messy scripts later on
3. Create problem sets for students to work with
   * Short exercises at the end of most major topics
   * Prepare more detailed practice set for end of Day 2 to test all topics and have short presentation if time
4. Share scripts, datasets, and practice problems with students before workshop (flash drive, or Google Drive folder?)

**LESSON PLAN:**

Day 1:

1. **Introduction to R and packages: 30 minutes**
   1. Introduction to R, layout of RStudio, how to write and run code
   2. Introduction to packages (install required packages if have not already done so), package manuals, help files, and forums (StackOverflow)
2. **Basics in R from a prepared script: 50 min**
   1. Objects, functions, arguments, assignment operators, comments
   2. Types of objects and types of data storage in R
   3. Warnings and errors (built into script so they can see how RStudio handles them)
   4. General coding tips and style (case sensitive, naming objects, parentheses vs. brackets vs. braces, missing commas or quotations)
   5. Getting data into R and related functions (setwd, getwd, read.csv, head, View)
   6. Practice similar exercise with a new script that has a couple errors (can they troubleshoot and modify things?)
3. **BREAK (10 minutes)**
4. **Working with data in R: 30 minutes**
   1. Indexing, selecting columns by name
   2. Summarizing data frames
5. **Adjusting a script from a colleague: 30 minutes**
   1. Introduce a scenario when you ask a colleague for a script and they send you one, and then you need to make it work with your own data
   2. What errors are there? How do you fix them? How do you adjust things to fit the data you are working with (give them a similar, but not exact, dataset to practice with)?
   3. Include a couple exercises/goals to achieve with the “borrowed” script and your “new” dataset so they can have a final answer to work towards

Day 2:

1. **Quick review from Day 1: 15 min**
   1. Overview of RStudio, objects, functions, data in R, etc.
2. **Manipulating data: 45 minutes**
   1. Introduce “dplyr” (filter, select, mutate, summarize) and “tidyr” (gather, spread)
   2. Using the pipe function to combine data commands
   3. Practice exercises with new dataset
3. **Basic analyses and plots: 30 minutes**
   1. Introduction to basic plotting in base-R
   2. Overview of simple analyses in R (correlation, t-test, linear regression)
   3. Practice exercises with new dataset
4. **BREAK (10 minutes)**
5. **Practice and presentations: 50 minutes**
   1. Prepare advanced practice questions and exercises with new dataset that they will have to figure out how to import data, format it, do analysis, and/or create a specific type of plot
   2. Split into small groups to work together and troubleshoot (break-out rooms?)
   3. End with short presentations of their results, if time (screen sharing?)