**Title:** R for Epidemiologists

**Prerequisites:** None

**Credit Hours:**  3

**Semester:**  Fall, 2017

**Time & Location:**  MW 11:15-12:30, Room MC 2306.

**Instructor of Record:** M. Alan Brookhart, PhD

 Professor, Epidemiology

 2105F McGavran-Greenberg CB #7435

**Facilitators:** Nat MacNell, MSPH macnell@unc.edu

Mike Fliss, MS mike.dolan.fliss@gmail.com
 Sara Levintow, MSPH levintow@email.unc.edu

**Purpose:** This course is an introduction to the R language and RStudio environment for statistical computing and graphics. The use of *R base* software for data management and analysis is emphasized, along with a few specialized packages.

R is a powerful, free, open-source, and community-supported statistical programming language that connects well with growing data science principles.

R for Epidemiologists is a 3-credit class intended to be the most effective and efficient way for UNC Epidemiology students to establish a foundation in the R programming language, RStudio IDE and functional programming modalities. We give special attention to R topics and packages relevant for epidemiology data management, analysis and presentation in the UNC Epidemiology framework. Homework assignments are designed to ease epidemiology students into the language efficiently by building on the UNC Epidemiology core curriculum, using familiar datasets and questions seen in other classes (e.g. NC births from 715/716). Each student completes a project of their own over the course of a semester, and are encouraged to use preexisting projects like those attached to papers, GRAs or dissertations to learn on and get peer support for. The class includes brief introductions to other useful, more advanced epidemiology topics that R facilitates such as spatial GIS, Bayesian statistics and multi-level modeling.

There are no prerequisites, but students who have taken EPID 715/716 will benefit from their past assignments, and students who have taken core epidemiology courses (705, 710, etc.) will benefit from having been introduced to fundamental epidemiology concepts that we'll use in class.

**Lectures:** See course schedule

**Office Hours:** For an appointment, email one of the course facilitators. Office hours are often held after class in the Epi student lounge.

**Learning Objectives:** The objectives of this course are:

1. To familiarize students with R syntax and elementary R programming concepts in preparation for working with R code in advanced epidemiology courses and in real research settings.
2. To learn traditional approaches to data management and other methods made possible by the R language.
3. To gain practice solving complex programming problems in the context of epidemiology by writing new code or by identifying, configuring, troubleshooting, evaluating, and adapting user-submitted R code from the Comprehensive R Archive Network (i.e. R packages).

**Texts & Resources**

Course notes are provided online at sakai.unc.edu.

The following free resources are suggested:

1. *R for Beginners* (<https://cran.r-project.org/doc/contrib/Paradis-rdebuts_en.pdf>)

 A free, short book on elementary R programming

1. *R for Data Science* (<http://r4ds.had.co.nz/>)

A likewise free book on more advanced programming concepts and packages

1. Online “Live” Introduction to R (free course): (<https://www.datacamp.com/courses>)

**Grading**

10% Attendance
 10% Small group & class activity participation

40% Homework problems

40% Final project

Final average Undergraduate Grade Graduate grade

95 or above A H

85-94 B P

75-84 C P

70-74 D L

Below 70 F F

Undergraduate + and – grades will be decided at the discretion of the instructors, based primarily on the final project.

**Homework**

In addition to the project, there are five homework assignments that cover the basics of epidemiologic analysis. These assignments will cover the same topic areas as the class, broadly: (1) nuts & bolts of R, (2) functions, loops and subsetting, (3) data management, (4) graphics, and (5) modelling. To turn in homework, submit the homework assignment word document with answers inline, with your code pasted in at the end. Homework is graded pass/fail, and students are encouraged to update their homework with improvements after homework answers and code are released.

**Final Project**

Students will be asked to complete a data analysis project as part of the course. Students and follow the steps typically used in analysis projects. In the last third of the semester, students will apply R programming skills to a dataset of the their choosing, deciding on a project scope, and then presenting their findings at the end of the semester on a few informal slides.

These analyses should use at least one R package beyond base R, either a package discussed in class or another of the students’ choosing. These special analysis techniques (like GIS or advanced modeling) are covered in the second half of the class, but do not have required homework exercises. Students are (always) encouraged to find an R activity that is useful to them elsewhere; double-dipping is encouraged. Datasets will be available for students without their own. Students must have individual project, but are encouraged to support each other by sharing progress and collaborating (just like in real life). Time will be set aside for project collaboration time in class with small groups.

**Collaboration and the UNC Honor Code**

All academic work in this course, including homework, exams, and projects, is designed to be collaborative - a process that combines a student’s new work with existing work created by others. Indeed, the R language itself is a collaboration between many authors. In this context, it is important to correctly attribute credit to sources used in the course of students’ work and to accurately identify the student’s individual contribution; representing others’ work as your own is a violation of the UNC Honor Code. Copying solution code for a specific problem verbatim (e.g. from a fellow student) defeats the educational purpose of the course, as does posting the problem on a community programming forum (e.g. Stack Exchange).

Students are strongly encouraged to consult their peers and any resources they can find (including programming forums) for other educational purposes, including but not limited to (a) learning about programming concepts that underlie problems or solutions, (b) finding examples of similar or simpler problems with ready solutions that can be adapted to the problem at hand, and (c) reviewing solutions to unrelated problems to learn about the process of finding a solution (i.e. troubleshooting programming errors).