STAT 479 (Spring 2017)

Data Science with R

Prerequisite:	333, programming in R	**Credits:**	3.0
Classes:	M 2:25-5:25	**Room:**	SMI 133
Professor:	Karl Rohe	**Office:**	MSC 1239
Phone:	263-8531	**E-mail:**	karlrohe@stat.wisc.edu
Office Hours:	Thur 2-3 or by appointment		
Webpage:	http://pages.stat.wisc.edu/∼karlrohe/ds.html		
TA (321, 322):	Anqi Shi	**Office:**	MSC B248
Phone:	206-403-8940	**E-mail:**	ashi6@wisc.edu
Office Hours:	Wed 1-2PM		

Course Objectives: Applied statistics is an iterative (back and forth) performance of four different types of activities (data collection, data wrangling, data analysis, communication) that require five different types of stances (scientist, coder, mathematician, methodologist, skeptic). Data Science (DS) is applied statistics in the age of the internet; this has led to two major changes from previous forms of applied statistics: easy sharing of software (e.g. CRAN, github, etc) and easy sharing of data (e.g. API data requests).

In class, we will practice the four activities and the five stances. We will share software and data. Group projects will synthesize these things into a performance of DS.

The overarching objective is to develop agile and reproducible code to quickly iterate through the pipeline (i.e. the four steps). As you develop your pipeline, you will necessarily iterate forward *and backward* through the pipeline, developing the separate pieces in non-consecutive order. In order to quickly iterate, we need to develop the ability to *think* and *code* in concise/agile syntax. The base R syntax is excessively broad; the tidyverse (which we will learn) and higher levels of programing more generally aim to streamline 80% of the concepts into short syntax. With agile syntax, it will be easy to update code, incorporate new pieces, etc.
You will develop:

- In the first half of the course, a broad set of computational tools (in R); but not the broadest!
- In the second half of the course, a broad set of statistical tools (machine learning); but not the broadest!

Because 80% of problems are very similar, we will focus on doing these with agility. Zen tip: in "agility" (not "speed"), our aim is to make the software and the coding "transparent." This focuses our concentration on our grand objectives (not on the coding). Analogously, experienced car drivers can navigate complicated directions while driving a car. Due to driving experience, the car has become transparent. Driving the car is like walking or breathing or riding a bike (or "navigating" a familiar path!); it does not require our conscious thought. The coding aspect of data analysis should become similar and agile code is the bridge to get there.

Text:

(r4ds) *R for Data Science* by Garrett Grolemund and Hadley Wickham. Access at r4ds.had.co.nz

(islr) *An Introduction to Statistical Learning with Applications in R* by Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani. You can download the text at www.StatLearning.com. If you enjoy reading words printed on tree carcass (I do), you can order on amazon.

Topics: We are going to start at the beginning of this list and work towards the end. I am hopeful that we will include all topics (at some resolution): importing data, dplyr, `%>%`, tidying, ggplot2, playgrounds and relational data, shiny, writing a thesis statement, prediction vs inference, unsupervised vs supervised learning, network analysis, PCA, topic modeling, nonlinear methods such as splines and generalized additive models, random forests.

Computing: We will make extensive use of the software R. It is assumed that you are comfortable coding in R. You should have comfort writing if statements and for loops. It is assumed that you will have access to a computer to complete the assignments. Can you bring a fully charged laptop to class? You might find these resources helpful `http://cran.r-project.org/doc/manuals/R-intro.pdf` and `http://adv-r.had.co.nz`.

Course Projects: There is one primary project (presentation and written document with a group) and perhaps one smaller (group?) project (shiny app).

Exams: There will be a midterm and a final.

Grading: Projects 41%, "Participation" 39%, Exams 10 + 10%.

Participation: I would prefer to base the course grade on only the class project. However, students often fall behind without an incentive for points. As such, this category is a necessary evil. It is a catchall for homeworks, exercises, quizzes, and attendance. Because we will often be "working" in class (i.e. not lecturing), you need to (i) come to class (ii) prepared and (iii) participate. When `R` code is required for handing in, it needs to be integrated into the text. You are expected to use Rmarkdown; it enables you to keep your notes clean while doing the work and then requires little effort to create the "final" document. See `http://rmarkdown.rstudio.com`. Unedited computer output will not be graded. There will be periodic reading quizzes (sometimes announced beforehand) to check that you have properly prepared for class. Each class should have at least one of these things. You are allotted one "miss" for personal reasons.

Academic Honesty: You are permitted, in fact encouraged, to talk to other students, your teaching assistant, or me about homework. However, you may not present other people's work as your own. If you work with other students solving problems, make sure that you write up your own solution independently. It is not acceptable for one student to write a solution for another student to copy. – On exams, your work is to be entirely your own.

Disability accommodation:

Any student with a documented disability (e.g., physical, learning, psychiatric, vision, hearing, etc.) who needs to arrange reasonable accommodations must contact the instructor and McBurney Disability Resource Center at the beginning of the semester (i.e. within the first two weeks). The instructor needs to keep a copy of the documented disability.