STAT 405/705: STATISTICAL COMPUTING WITH R

QUARTER 1, FALL 2020

Meetings.

- Section 1: Tu/Th 9–10:20 am
- Section 2: Tu/Th 10:30 am–11:50 pm

The lectures will be given virtually via Zoom. Our first meeting is on September 1 and the last meeting is on October 20. We will not meet on September 29 in order to balance Q1 and Q2.

Instructor. Weijie Su (`suw@wharton.upenn.edu`)

Office hours**:** TBD

Teaching assistant. Shuxiao Chen (`shuxiaoc@wharton.upenn.edu`)

Office hours**:** TBD

Course website. All course related materials will be distributed via the Canvas website and grades can be checked during the quarter using Canvas. You are encouraged to use Piazza on Canvas to ask questions regarding the course material, assignments and any scheduling issues, which allows everyone to benefit from the Q&A.

Background. The R statistical programming environment has long been the platform of choice for quantitative statistical research activities. Literally, thousands of add-on packages are available providing for a comprehensive suite of data analysis tools. Recently, R has made a major push into the business environment where it is frequently used by data scientists.

The goal of this course is to introduce students to the R programming environment and related ecosystem and thus provide them with an in-demand skill-set, in both the research and business environments. Further, R is a platform that is used in other advanced classes taught at Wharton, so that this class will prepare students for these higher level classes and electives.

One key feature of R is that it is open source and is freely distributed. Consequently, it is a platform, that once learnt, will remain universally available to students.

Prerequisites. Any of Stat 102, Stat 112, Stat 431, Stat 613, Stat 621 (or waiving the MBA statistics course). These classes all take students to the level of regression modeling.

Course overview. This one quarter course will expose students to the R statistical programming language. No previous programming experience is assumed. There will be an assumption that students have completed prior statistics courses to the level of multiple regression analysis.

Students are expected to have access to a computer on which they have installed R, the R studio IDE, any necessary support packages, and the video conference tool Zoom. Classes typically have a four part structure:

(1) A topic overview
(2) Instructor demonstration
(3) Student hands-on group-based project activity
(4) Wrap-up

By the end of the course students should be able to code statistical functions in R. They should be able to extend the functionality of R by using add-on packages and they should be able to use R to perform the work-horse statistical tasks such as multiple-regression and simulation analyses.

Course materials. The R statistical software program, which is available at https://www.r-project.org/; RStudio is an Integrated Development Environment (IDE) for R, which is available from https://www.rstudio.com/. Course notes will be available from Canvas.

There is no required textbook, though there are optional texts available for students to refer to. A recommended one is *R for Data Science: Import, Tidy, Transform, Visualize, and Model Data*, 1st Edition by Hadley Wickham, and Garrett Grolemund. Another useful book would be *Modern Data Science with R*, 1st Edition by Benjamin Baumer, Daniel Kaplan, and Nicholas Horton.

There are many quality free on-line tutorials and resources for R such as DataCamp and Coursera that students may find useful too. Jenny Bryan's course at UBC is also very popular and the materials are available at https://stat545.com/.

Homework. There will be 5 homeworks during the quarter. These homeworks will be prescriptive and involve performing a set of programming related tasks in R. The deliverables will be R code, output and related discussions. Submit your homework via Canvas. Homework should be submitted to Canvas as text files for code and PDF files for any required output. There is no final exam but rather a take-home final project. You may discuss the homeworks with other students, but you must write your own code. If you use code from any outside source, then it must be attributed in your homework code itself. When appropriate, homeworks will be run through Turnitin to determine originality. Plagiarized code will receive a 0. Homework should be submitted to Canvas as text files for code and PDF files for any required output. Late homeworks are penalized by 25% up to 1 day late and 50% up to two days late. Homeworks more than two days late will receive a 0. Homeworks will be due 11:59pm eastern time on the dates below.

Deliverable	Due date
Homework 1	9/12
Homework 2	9/19
Homework 3	9/26
Homework 4	10/10
Homework 5	10/17
Take-home final project	10/29

Quizzes. There will be 5 *in-class* quizzes. Each quiz has 5 multiple choice questions and will take 10 minutes (you can leave after 5 minutes). You can drop the lowest quiz score. There will be no make-up quizzes.

Deliverable	Date
Quiz 1	9/8
Quiz 2	9/15
Quiz 3	9/22
Quiz 4	10/6
Quiz 5	10/15

The quizzes will be given at the end of the classes. Once you have opened the quiz, you will have exactly 10 minutes to complete the quiz. The timer is not stopped if you close the browser window or navigate away. If you attempt to do this – i.e. open the quiz, close it, and then come back later to finish it – it is considered cheating. Note that we can follow everything you do on Canvas, so there is no way to skirt this prohibition.

Class content. The class content is structured along the lines of using R for a project in a business setting. Specifically, there needs to be a problem definition and planning stage. Data needs to be identified and read into the analysis platform. The analysis occurs. Results are reported to interested parties.

MODULE 1. Introduction to R and RStudio. In this class we will get to know R. This involves first of all installing R and RStudio. The basic functionality of R will be demonstrated. Using R for calculations. Using R to calculate summary statistics on data. Using R to generate random numbers. Variable types in R. Numeric variables, strings and factors. Accessing the help system.

MODULE 2. Data structures: vectors, matrices, lists and data frames. R makes extensive use of various data structures. The core data structures are vectors, matrices, arrays, lists and dataframes. We will discuss accessing elements of these data structures, sub-setting vectors, slicing arrays and drilling down on lists. We will also take a first look at the apply and lapply functions, that allow you to apply functions to arrays and lists.

MODULE 3. Reading data into R from various data sources. Merging data across data sources. R has many options for bringing in data for analysis. These include reading from flat files (plain text), reading from database connections and reading from web sources. Many problems involve multiple data sources, so we will discuss merging data sources in R.

MODULE 4. Statistical modeling functions: lm and glm. Linear and generalized linear models (for example, logistic regression) are the workhorses of modern analytics. This class will illustrate the implementation of these functions in R and requires the use of the formula syntax for model specification. We will discuss prediction and model checking.

MODULE 5. Writing your own functions (I). One of the most powerful features of R is the ability to write your own functions. These functions may help pre-process data or implement specific computational algorithms. In this class we will introduce the R function syntax and in particular the passing of variables into the function, and argument handling.

MODULE 6. Writing your own functions (II). There are always elegant ways to write functions and more brute force approaches. We will describe approaches that make R more efficient in function evaluations, in particular vectorization. We will discuss the "..." notation that allows arguments to be passed on to other functions. We will discuss functions that themselves take other functions as arguments.

MODULE 7. Iterating with R. Logic and flow control. In order to prepare for simulation modeling it is important to be able to control the flow of an algorithm. These functions include the if, for, while and break constructs.

MODULE 8. Simulation I. This class will introduce Bootstrapping and Monte-Carlo simulation in R. We will investigate the sample command and be introduced to random number generation from the canonical probability distribution functions.

MODULE 9. Simulation II. Expanding on the simulation ideas from the previous class we will investigate some permutation tests as alternatives to classical hypothesis tests .We will also use simulation to check whether or not modeling assumptions appear reasonable, for example whether normality for a quantity such as the maximum likelihood estimator is reasonable. Finally we will compare the efficiency of different sampling methodologies, simple random sampling and stratified random sampling in terms of the efficiency of the estimates they produce.

MODULE 10. Extending R with add-on packages and the R ecosystem. One of the great benefits of using R is the access to hundreds of add-ion packages. This class will discuss the R ecosystem, and illustrate R's extensibility. We will illustrate this extensibility through an example using the randomForest package and various other packages that work in conjunction with it.

MODULE 11. Graphics. A picture is worth a thousand words and any statistical analysis with no graphics tends to fall a little flat. R has a wide range of graphic abilities, from the low level graphical primitives that can be used to build more complicated graphics to high level routines such as ggplot2. This class will explore some of these graphical capabilities.

MODULE 12. Dynamic and web reporting: Knitr and RMarkdown. The sincerest form of flattery is implementation. In this class we will look at the facilities available to create living reports and on-line presentations driven by the R language.

Grading. The final grade will be weighted using 55% from the five assignments (each counts as 11%), 20% from the final project, and 25% from the quizzes. All assignments will be included in the final grade. There is *no* "drop the lowest score" policy for the assignments, but you can drop the lowest quiz score. There will be no extra credit opportunities at the end of the course. Grade queries/re-grade requests must be submitted to the TA within one week of the homework being graded. Any request will result in regrading the whole assignment.