Course Syllabus

Fall 2018 Programming in R for Analytics, 94-842

• R programming
• RStudio IDE
• Shiny
• Data Manipulation – ingestion, aggregation, visualization, reporting
• Data analysis
• Linear Statistical modeling and Statistical Hypothesis Testing

Instructor: Prahlad G Menon, PhD
Office Location: TBD
Email Address:
pgmenon@andrew.cmu.edu

Office Hours: After Class, Thursdays, Location HBH 1002 (Mini-1), HBH 1206 (Mini-2).

Phone: +1 (412) 259-3031

Teaching Assistant(s): Alton Lu, Youngjoo Sun, Xinmei Zhang
Email Address:
lualton@cmu.edu
yson@andrew.cmu.edu
xinmeiz@andrew.cmu.edu

Office Hours: Monday (1:30-3:30pm), HBH 1111, Thursday after class (6-8 pm), Friday 3-5 pm, HBH 1002

NOTE:
1) For email communication, please include CMU 94-842 in the subject line.
2) The responsibility of the TA will be limited to grading, proctoring tests / exams and conducting office hours for the purpose of resolving basic inquiries. The TA is not expected to resolve student questions on a technical level without the support of the course instructor.

Course Description:
This course introduces students to R, a widely used statistical programming language, and Shiny – a popular R-based web-application development framework that offers a front-end interface to the statistical prowess of the libraries accessible through R. Students will learn to manipulate data objects, produce graphics, analyze data using common statistical methods, and generate reproducible statistical reports. They will also gain experience in applying these acquired skills in various public policy areas.

By the end of the class, students learn to:
• Use RStudio, read R documentation, and write R scripts.
• Import, export and manipulate data.
• Produce statistical summaries of continuous and categorical data.
• Produce basic graphics using standard functions, and produce more advanced graphics using the ggplot2 library.
• Perform common hypothesis tests, and run simple regression models in R
• Produce reports of statistical analyses in R Markdown.
Prerequisites:
- Basic vector calculus and linear algebra which will underpin the implementation and exploration of the key mathematical concepts learned in this class;
- A laptop with admin access to install R, RStudio and relevant libraries needed through the course;
- Students must be enrolled in a graduate program in Heinz College. Special permission can be granted by the College.

Course Level, Number of Units: Graduate, 6 Credit Units.

Class Schedule:
- Lecture:
  4:30 pm – 5:50 pm, Tuesdays and Thursdays, 27 Aug 2018 – 12 Oct 2018

Venue:
Hamburg Hall 1002 (Mini-1), 1206 (Mini-2), Carnegie Mellon University.

Required Textbook(s):
Core text: There is no core textbook for this course. However, a series of reference texts will be used during this course:
- Garrett Grolemund and Hadley Wickham, R for Data Science
- Phil Spector, Data Manipulation with R
- Paul Teetor, The R Cookbook
- Winston Chang, The R Graphics Cookbook
- Norman Matloff, The Art of R Programming: A Tour of Statistical Software Design
- Roger D. Peng, R Programming for Data Science

Note: Selected chapters from the above-mentioned textbooks will be made available to the enrolled students, as PDFs, via the course website on Blackboard.

Other Resources:
There are many resources online that may help you to learn R. A few that are particularly relevant for this course are listed below.
- R Style guide
- An Introduction to Factors in R
- A brief introduction to apply in R
- ggplot2 cheatsheet
- The odds ratio: calculation, usage, and interpretation
- Fisher's exact test
- Pearson's Chi-squared test
In addition, the following are good on-line resources on R related information:

- R studio cheat sheet  https://www.rstudio.com/resources/cheatsheets/
- R Project homepage https://www.r-project.org # Official news from the R project, including links to documentations mailing lists, and more
- StackOverflow http://stackoverflow.com/questions/tagged/r : An active R community for coding questions. You can search whether someone has already asked similar questions.
- R bloggers http://www.r-bloggers.com # Tutorials about R and interesting posts. Skill levels may vary
- CRAN Task Views add-on packages https://cran.r-project.org/web/views/OfficialStatistics.html
- Crantastic.org http://crantastic.org #A community site for R packages. You can search, review, and tag CRAN packages
- Books about R https://www.r-project.org/doc/bib/R-books.html #Extensive list of R books from the R project

Canvas – The New Blackboard:

Course Website: To access the course Canvas page, login at: https://canvas.cmu.edu/courses/6129. You should check the course Canvas page daily for announcements and handouts. PDF copies of lecture notes and all related course content, including R / RMD files with code written up in class, will be available via the Canvas portal.

Other Supplemental Materials:
Supplemental reading assignments and additional material, including but not limited to code and instructional tutorials, will be uploaded for student access as the course progresses, via the course website on Blackboard.

Grading Algorithm:

Quizzes: 20%
- Not present / not taken = 0
- Your course grade is equally affected whether you miss 1 point on a 3 point quiz, or you miss 1 point on a 10 point quiz.
- Lowest two quiz scores are dropped (i.e. the 2 on which you missed the most points) in order to be accommodating in the event that you are missing for a week-long conference.

Homework: 60%

Final Project (teams of 2-3): 20%
- Final project report, code (including code-documentation) and final project presentation, each weighted for 10% of the total grade.
**Final Letter Grade**
While lower cutoffs may be used based to relax the expectations based on principles of grading the class on a *bell curve*, the following maximum grade cutoffs are guaranteed:

- \( \geq 93.5 \) A
- \( \geq 90.0 \) A-
- \( \geq 87.5 \) B+
- \( \geq 83.5 \) B
- \( \geq 80.0 \) B

**Tentative Course Content**
- 14 lectures, total + 4 Quizzes + 3 Homework Assignments + 2 Project Deliverables (i.e. a proposal and a final project submission).

**Course Calendar (Tentative)**

<table>
<thead>
<tr>
<th>Date</th>
<th>Day</th>
<th>Class Activity</th>
</tr>
</thead>
</table>
| August 27 | Mon. | Canvas Open –  
- Syllabus available |
| 28 | Tue. | CLASSES BEGIN. Introduction to R for Programming in Data Science + Examples for R, R/Shiny Applications that students may look forward to building during / after this course.  
Assignment 0 uploaded with Reading Assignments (links / PDFs) |

**Week 1:** Getting Started, Setting Up a Development Environment, Retrieving Packages:
- R Console Input and Evaluation  
- Data Types - R Objects and Attributes, Vectors and Lists, Matrices, Factors, Missing Values, Data Frames, Names Attribute, Summary  
- Reading Tabular Data  
- Reading Large Tables  
- Textual Data Formats  
- Connections: Interfaces to the Outside World  
- Subsetting – Basics: Lists, Matrices, Partial Matching, Removing Missing Values  
- Vectorized Operations  
- Introduction to swirl  
- Introduction to AWS R/Shiny Server for this class

**Week 2-3:** Data frames, Functions, Loops, `apply()`, `sapply()`, `lapply()` etc., Conditional Statements, Debugging
- Control Structures - If-else  
- Control Structures - For loops  
- Control Structures - While loops  
- Control Structures - Repeat, Next, Break  
- Your First R Function  
- Functions (part 1)  
- Functions (part 2)  
- Scoping Rules - Symbol Binding  
- Scoping Rules - R Scoping Rules  
- Scoping Rules - Optimization Example
- Coding Standards
- Dates and Times, timing code blocks
- Reading: Practical R Exercises in swirl

**Week 3-4: Data aggregation, dplyr, Visualization / Graphics using ggplot, plotly**

- Loop Functions and Debugging
- Loop Functions – lapply, apply, mapply, tapply, vapply, split
- dplyr
- Debugging Tools - Diagnosing the Problem
- Debugging Tools - Basic Tools
- Debugging Tools - Using the Tools
- Reading: Practical R Exercises in swirl
- Basic Graphics

**Week 4-5: Statistical testing, generalized linear modeling for continuous & dichotomous response variables**

- Simulation - Generating Random Numbers
- Simulating a Linear Model
- Random Sampling

**Week 6: Natural language processing, regular expressions and text / string manipulation, web-scraping**

- The str Function, stringr, stringi, grepl, regexp
- Stringi and stringr libraries
- httr and curl

**Week 7: Shiny and Interactive Graphical Widgets, Integrating R with Python / Java JDBC / ODBC connectors**

**October**

<table>
<thead>
<tr>
<th>Date</th>
<th>Day</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>Tue.</td>
<td>Final Lecture + Final Project Presentations.</td>
</tr>
<tr>
<td>11</td>
<td>Thur.</td>
<td>Final Project Presentations.</td>
</tr>
</tbody>
</table>

**Symposium of Final Projects**

The course will culminate with a symposium of student projects which will leverage the hypothesis testing, quantitative analysis and R/Shiny App building techniques applied to real datasets from the UCI dataset library: [https://archive.ics.uci.edu/ml/datasets.html](https://archive.ics.uci.edu/ml/datasets.html).

**Education Objectives**

(a) **An ability to apply knowledge of mathematics, science, and engineering:**
Theoretical lectures and tutorials are combined with practical exercises in programming and mathematical derivation / problem solving) to prepare students for a large-scale science/engineering final project relevant to data mining, data analysis and data science in general.

(b) **An ability to design and conduct experiments, as well as to analyze and interpret data:**
Class assignments and project work will require students to set up a development environment and within this environment design, build, and run data-processing experiments in software to implement empirical or numerical models (through various
libraries available open-source) which embody data-processing pipeline architectures to explore and deriving value from data.

(c) An ability to identify, formulate, and solve engineering problems:
As a project-based course with several assignments and quizzes along the way, students will constantly be solving engineering problems by hand (i.e. pencil and paper) or in terms of code (i.e. programming). For the final projects further, students will both identify measurable problems as well as formulate and then proceed to implement (at least partially) the engineering approach to address the identified problem.

(g) An ability to communicate effectively:
Ongoing participation of students during the classroom discussions as well as the final projects will require both articulation of complex mathematical and conceptual information and individual student presentations. The latter will have presentation format equivalent to an oral conference presentation. Although students will be graded primarily on technical content, they will also be graded on the clarity, effectiveness and length of their presentations.

(i) A recognition of the need for, and an ability to engage in life-long learning:
Students are shown from the beginning of the course that building useful data processing pipelines requires awareness of the cutting edge in the field, the inherent limitations in having to deal with big data, as well as the constraints of user-requirements. Further, students are made aware of the fact that a working knowledge of tools or techniques employed in contemporary workflows during data science practice is required in order to define today’s limitations and engineer tomorrow’s novel technology solutions. These ideals are aligned with the goals of recognizing the importance of life-long learning as it relates to data science and data engineering practice.

(j) A knowledge of contemporary issues:
The course work and syllabus is based upon tools that have seen extensive application in academic literature and commercial embodiments that engage in the use of programming in R for data science. Students will require to consult with the latest scientific literature as part of their regular reading assignments as well as in order to address the project requirement in this course i.e. in regard to both project topic selection and implementation, including potentially leveraging big-data tools like Hadoop / Hive, Spark, Apache Drill etc. in order to process particularly large / unwieldy datasets for their project work. The readings selected for this course as well as the lecture notes (i.e. slides, code and code notebooks) provide references to active research and their work in the current literature.

(k) An ability to use the techniques, skills, and modern engineering tools necessary for engineering practice:
As a project based course, students are taught and required to make use of a wide variety of algorithms, open-source code and software libraries in addition to basic mathematical skills and programming in order to address data analysis and visualization problems. Therefore, the course exposes students to a wide range of techniques, skills, and modern engineering tools necessary for engineering practice in the industry.
University Policies:

*Academic Integrity*

Students in this course will be expected to comply with the CMU Policy for Academic Integrity: [http://www.cmu.edu/academic-integrity/](http://www.cmu.edu/academic-integrity/). Any student suspected of violating this obligation for any reason during the semester will be required to participate in the procedural process, initiated at the instructor level, as outlined in the University Guidelines on Academic Integrity. This may include, but is not limited to, the confiscation of the examination of any individual suspected of violating University Policy.

*Disability Services*

If you have a disability for which you are or may be requesting an accommodation, you are encouraged to contact both your instructor the Disability Resources office at 8-2013.

*Copyright Notice*

Course materials may be protected by copyright. United States copyright law, 17 USC section 101, et seq., in addition to University policy and procedures, prohibit unauthorized duplication or retransmission of course materials. See [Library of Congress Copyright Office](https://www.loc.gov/copyright/) and the [University Copyright Policy](https://www.cmu.edu/copyright/).

*Statement on Classroom Recording*

To ensure the free and open discussion of ideas, students may not record classroom lectures, discussion and/or activities without the advance written permission of the instructor, and any such recording properly approved in advance can be used solely for the student’s own private use.