STAT 689: Statistical Computing with R and Python, Spring 2018

Instructor: James Long

Lecture Time: M/W/F 12:40pm – 1:30pm

Prerequisites: Some experience with writing code (> 100 lines) in R, python, or Matlab. Experience analyzing data sets from an industrial, engineering, or scientific domain. Basic understanding of fundamental statistical models, e.g. linear regression. Contact the instructor if you are interested in taking the course but have doubts about your background experience.

Course Description: This course covers aspects of numerical analysis for statisticians and data scientists (including matrix inversion, splines, function optimization, and MCMC) with an emphasis on implementing these methods in R and python. Important language specific tools and computation strategies such as vectorization, code profiling, and data visualization will also be covered. Class examples, homework, and projects will be completed in R or python using Jupyter notebooks. Students will have some choice over which language they primarily use for assignments, but they must be willing to code in both.

Comparison with STAT 689 “Databases and Computational Tools used in Big Data” (DCTBD): DCTBD focuses on databases, computing on clusters, and parallel computing. We will not cover databases or computing on clusters at all and will spend little or no time on parallel computing. There will be some course overlap in learning git, python, and code vectorization strategies.

Learning Outcomes: At the end of this course students will be able to:

1. Understand the inner workings of common R and python function (e.g. `lm` in R) and use this knowledge to optimize code.
2. Learn and implement common optimization algorithms (e.g. EM, MM, Newton). Understand their application to common statistical models (e.g. non-linear regression, mixture models).
3. Understand the importance and challenge of numerical matrix inversion for statistical applications. Implement computational strategies to avoid, speed up, and stabilize matrix inversion.
4. Understand common data structures in python and R (vectors, matrices, arrays, lists, dataframes) and their various strengths and weaknesses.
5. Provide several examples of the quote “The form of a mathematical expression and the way the expression should be evaluated in actual practice may be quite different” (J. Gentle)
6. Design and implement simulation studies in R and python.
7. Produce reproducible research reports in clear, well-documented R and python code.
**Textbooks:** There is no required textbook for this course. Lectures will be based on material from the following sources, all of which are available online as pdfs through TAMU library.

- The numerical analysis / modeling portion of this course will primarily be based on material from:
  - Computational Statistics by Gentle (intermediate, mostly parts 1 and 2)
  - Numerical Analysis for Statisticians by Kenneth Lange (advanced, mostly chapters 5, 7, 9–14, 26)
- R and python specific algorithm implementation / data analysis tools will be taught primarily from:
  - The Art of R Programming by Matloff (beginner)
  - Advanced R by Wickham (advanced)
  - Python for Data Analysis by McKinney (intermediate)

**Tentative Course Schedule:** The following is a tentative course schedule. Topics may change based on time and student interest. The topics later in the schedule are the most tentative.

- **Weeks 1–2:** Getting Started—Downloading and setting up R and python. The basic syntax of each language including function creation and important packages (e.g. numpy). Jupyter notebooks and version control with git. The importance of vectorization. C/C++ versus R/python for loops. Timing code. Calling R from python and vice versa.

- **Weeks 3–4:** Linear Regression and Matrix Inversion—Sweeping and the Cholesky decomposition for determining linear regression parameter fits. Implementations in R and python. Applications of linear regression to splines and signal frequency detection.


- **Weeks 7–9:** Optimization and Root Finding—Optimization techniques such as Newton, quasi-Newton, and Hessian approximation. Motivation from maximum likelihood theory and M-estimators. Applications to various statistical models. Implementations in R and python.


- **Weeks 12–15:** Overview of important R and python packages and student presentations.

**Technology**

**github:** Course materials will be posted on github. Please check often.
**Software:** We will be using python 3 and R. Both python and R are free, open source, and available for Windows, Mac, and Linux. Assignments will be completed using Jupyter notebooks. You will need to install python libraries numpy and scipy.

**Getting Help**

**Instructor Office Hours:** TBA

**Instructor email:** jlong@stat.tamu.edu

**Office Hours versus Email:** Questions about course material should be addressed in class or during office hours. Please use email only for administrative issues (eg cannot attend exam, need help but cannot make scheduled office hours, disability issues).

**TA:** TBA

**Grading, Exams, and Assignments**

**Grading Policy:** You will receive a percent correct (0-100) on the homeworks, in–class presentation, projects, and exams. These percentages are weighted:

\[
25\% \text{ homework} + 25 \% \text{ in–class presentation} + 25\% \text{ final project} + 25\% \text{ exam}
\]

The result of this weighting is the percent performance (PP). This is converted to letter grades as follows:

- \( 90\% \leq \text{PP} \leq 100\% \rightarrow A \)
- \( 80\% \leq \text{PP} < 90\% \rightarrow B \)
- \( 70\% \leq \text{PP} < 80\% \rightarrow C \)
- \( 60\% \leq \text{PP} < 70\% \rightarrow D \)
- \( 0\% \leq \text{PP} < 60\% \rightarrow F \)

**Homework:** Posted and completed on github using Jupyter notebooks. About one problem set per week. Students are encouraged to work together, but the answers must be your own. No late homework accepted. Lowest homework grade dropped.

**In-Class Presentation:** Students will give one twenty minute presentation on a statistical package or tool available in either R or python. Presentations will be made in groups.

**Exam:** There will be one in–class exam about 2/3 of the way through the course.

**Final Project:** Students will reproduce and/or extend a simulation or data analysis study previously published in a statistical, engineering, or scientific journal. Code as well as a written report with tables and figures will be submitted for grading.
Course Policies

Absence: You are required to attend class. Only university excused absences will be accepted for missing work. If you know you will miss an exam for a valid reason, please see or email me as soon as possible. See http://student-rules.tamu.edu/rule07 for what constitutes a university excused absence.

Americans with Disabilities Act (ADA) Policy Statement: The Americans with Disabilities Act (ADA) is a federal anti-discrimination statute that provides comprehensive civil rights protection for persons with disabilities. Among other things, this legislation requires that all students with disabilities be guaranteed a learning environment that provides for reasonable accommodation of their disabilities. If you believe you have a disability requiring an accommodation, please contact Disability Services, currently located in the Disability Services building at 701 West Campus Blvd. You can call 979-845-1637 or visit http://disability.tamu.edu.

Copyright Notice: The handouts used in this course are copyrighted. By 'handouts' I mean all materials generated for this class, which include but are not limited to syllabi, quizzes, exams, lab problems, in-class materials, review sheets, and additional problem sets. Because these materials are copyrighted, you do not have the right to copy the handouts, unless I expressly grant permission.

Statement on Plagiarism: As commonly defined, plagiarism consists of passing off as one's own ideas, words, writing, etc., which belong to another. In accordance with this definition, you are committing plagiarism if you copy the work of another person and turn it in as your own, even if you should have the permission of that person. Plagiarism is one of the worst academic sins, for the plagiarist destroys the trust among colleagues without which research cannot be safely communicated. If you have any questions regarding plagiarism, please consult the latest issue of the Texas A&M University Student Rules, under the section “Scholastic Dishonesty.”