LECTURE 1: INTRODUCTION

STAT 545: Introduction to Computational Statistics

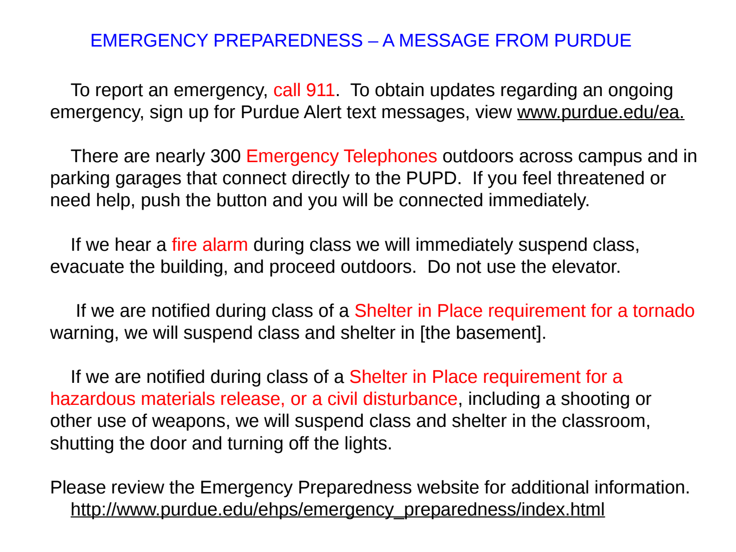
Vinayak Rao

Department of Statistics, Purdue University

August 20, 2019

Logistics

• **Class**	Tue/Thu 1030-1145AM, UNIV 119
• **Class Website**:	`https://varao.github.io/stat545_fall19/`
	`http://piazza.com/purdue/fall2019/stat545`
• **Class Email**:	`purduestat545@gmail.com`
	Send all homework here
• **Instructor**:	Vinayak Rao (varao@purdue.edu)
	If you email me, include STAT545 in the subject
	E.g. ‘STAT545: My dog ate my homework’
Office:	Math212
Office Hours	1245 - 145 PM Tuesdays or by appointment
• **TA**:	Zizhuang Wu
Office Hours	315-415 PM Mondays (MATH G143)



EMERGENCY PREPAREDNESS – A MESSAGE FROM PURDUE

To report an emergency, call 911. To obtain updates regarding an ongoing emergency, sign up for Purdue Alert text messages, view www.purdue.edu/ea.

There are nearly 300 Emergency Telephones outdoors across campus and in parking garages that connect directly to the PUPD. If you feel threatened or need help, push the button and you will be connected immediately.

If we hear a fire alarm during class we will immediately suspend class, evacuate the building, and proceed outdoors. Do not use the elevator.

If we are notified during class of a Shelter in Place requirement for a tornado warning, we will suspend class and shelter in [the basement].

If we are notified during class of a Shelter in Place requirement for a hazardous materials release, or a civil disturbance, including a shooting or other use of weapons, we will suspend class and shelter in the classroom, shutting the door and turning off the lights.

Please review the Emergency Preparedness website for additional information.
http://www.purdue.edu/ehps/emergency_preparedness/index.html

Grading

Your grade will be curved

Homework:	30%
Midterm 1:	25%
Midterm 2:	25%
Project:	20%

If you do well, you will get an A

- you are not competing with your classmates

If you do poorly, you will get an C

- don't compare with your laziest classmates

Homework

Six assignments involving reading, writing and programming

Are vital to doing well in the exams

Late homework will not be accepted

One (worst) homework will be dropped

Homework

Six assignments involving reading, writing and programming

Are vital to doing well in the exams

Late homework will not be accepted

One (worst) homework will be dropped

You may discuss problems with other students, but must:

- write your own solution independently
- name students you had significant discussions with

I take academic integrity very seriously

Academic integrity:

`www.purdue.edu/odos/osrr/academic-integrity/index.html`

Project

A nontrivial real world problem

Read and implement an algorithm from a paper

Groups of 2 (contact me first for groups of size 3)

- Complexity $\approx$ # groupmembers × one homework

Must submit:

- A proposal explaining problem, goals and distribution of work (mid-October)
- A report (each group member must submit their own report)
- A short presentation

Start thinking about this early!

Prerequisites

We will use R for homework/project (more later)

- Need to know R or a language like Matlab/Python etc
- *This is not a course to learn programming in R* (see STAT598z)
- Don't need to be an expert but willing to learn as you go

Some (undergrad engineering-level) math

- Probability and statistics: conditional densities, Bayes rule, maximum likelihood.
- Linear algebra.
- Basic multivariate calculus

Prerequisites

- What is a positive-definite matrix?
- What is the Hessian of a function? What is a convex function? Plot a convex function.
- Write down the pdf of the multivariate normal.
- What is the law of large numbers? Central limit theorem?
- What is the Kullback-Leibler divergence?
- What is the trace of a matrix?
- For a matrix X, what is $\frac{\mathrm{d}|X|}{\mathrm{d}X}$?

Books

We will not use a fixed textbook for this course.
Will link to relevant documents over the course. For now:

- Math cribsheets:
  `http://homepages.inf.ed.ac.uk/imurray2/pub/cribsheet.pdf`
  `http://www.cs.nyu.edu/~roweis/notes/matrixid.pdf`
- Old and new matrix algebra (T. Minka):
  `http://research.microsoft.com/en-us/um/people/minka/papers/matrix/`
- R-manual
  `http://cran.r-project.org/doc/manuals/R-intro.pdf`

Books (contd.)

That said, there are relevant textbooks for reference:

- G. James, D. Witten, T. Hastie, and R. Tibshirani. *An Introduction to Statistical Learning: with Applications in R.* Springer Texts in Statistics. Springer New York, 2014
  `http://www-bcf.usc.edu/~gareth/ISL/`
- G.H. Givens and J.A. Hoeting. *Computational Statistics.* Wiley Series in Computational Statistics. Wiley, 2012
- Paul Teetor. *R cookbook.*
  O'Reilly, Beijing, 2011
- N. Matloff and N.S. Matloff. *The Art of R Programming: A Tour of Statistical Software Design.*
  No Starch Press, 2011

Programming in R:

- Don’t need to be an expert, but be willing to learn.
- Won’t be graded for programming elegance.
- Reading assignments and homeworks will guide you.
- But you need to experiment yourself!
- If you’re stuck, the internet is your friend.

Programming in R:

- Don’t need to be an expert, but be willing to learn.
- Won’t be graded for programming elegance.
- Reading assignments and homeworks will guide you.
- But you need to experiment yourself!
- If you’re stuck, the internet is your friend.

Setting up your computing environment:

- Important you have access to R/a text-editor/a compiler.
- Your Purdue account should have all of this.

Probability, statistics and computation

Probability:

- analysis of random phenomena (properties of probability distributions and models).

Statistics:

- the study of the collection, organization, analysis, interpretation and presentation of data

Computation:

- Vital for stat. analysis of large datasets/complex models
- Storage/representation/manipulation of data
- Development and analysis of algorithms

Computational Statistics

Comp. Statistics vs Stat. Computing

[1] C. Lauro. Computational statistics or statistical computing, is that the question? *Comp. Stat. and Data Analysis*, 23(1):191–193, 1996

Comp. Statistics vs Stat. Computing

- One view: ‘Who cares?’ But[1]:
- Statistical Computing: ‘Application of Comp. Sci. to Statistics’

  Tools: programming, software, data structures and their manipulation, hardware (GPUs, parallel architectures)

  E.g. Releasing software to the world?

[1] C. Lauro. Computational statistics or statistical computing, is that the question? *Comp. Stat. and Data Analysis*, 23(1):191–193, 1996

Comp. Statistics vs Stat. Computing

- One view: 'Who cares?' But[1]:

- Statistical Computing: 'Application of Comp. Sci. to Statistics'

  Tools: programming, software, data structures and their manipulation, hardware (GPUs, parallel architectures)

  E.g. Releasing software to the world?
- Computational Statistics: 'Design of algorithms for implementing statistical methods on computers'

  Statistical methodology/algorithms E.g. Writing a paper?

[1] C. Lauro. Computational statistics or statistical computing, is that the question? *Comp. Stat. and Data Analysis*, 23(1):191–193, 1996

Comp. Statistics vs Stat. Computing

- One view: 'Who cares?' But[1]:
- Statistical Computing: 'Application of Comp. Sci. to Statistics'

  Tools: programming, software, data structures and their manipulation, hardware (GPUs, parallel architectures)

  E.g. Releasing software to the world?
- Computational Statistics: 'Design of algorithms for implementing statistical methods on computers'

  Statistical methodology/algorithms E.g. Writing a paper?

This course: a bit of both (but mostly latter)

[1] C. Lauro. Computational statistics or statistical computing, is that the question? *Comp. Stat. and Data Analysis*, 23(1):191–193, 1996

Computational Statistics

We will look at algorithms to:

- optimize loss/utility functions
- integrate functions
- generate/use random numbers (Monte Carlo methods)

Often these attempt to solve the same statistical problems

We will look at

- general purpose algorithms
- algorithms that exploit specific structure (e.g. Gaussianity, conditional independence etc.)
- frameworks for developing new algorithms

The formal goal of this course is (with STAT546) to prepare PhD students for the Computaitonal Statistics qualifying exam.

Topics covered (tentative)

- Overview of R (very brief!)
- Intro. to gradient-based optimization
- Numerical linear algebra and algorithms
- Data structures, sorting and basic complexity analysis
- Dynamic programming, hidden Markov models, Kalman filter
- Greedy heuristics like the k-means algorithm
- Exponential family distributions and conjugate priors
- Intro. to the EM algorithm
- Intro. to Monte Carlo methods and numerical integration
- Intro. to Markov Chain Monte Carlo methods
- More optimization and root-finding

Programming and R

R: a programming environment for statistical computing.

Based on Bell Labs' S language by John Chambers

Started by Ihaka and Gentleman at the Univ. of Auckland *R: A Language for Data Analysis and Graphics*, (1996)

A high-level interpreted language with convenient features for loading, manipulating and plotting data

- Free, open source.
- A huge collection of user-contributed packages to perform a wide variety of tasks
- Widely used in academia, and increasingly popular in industry

Programming with R

Install R from http://cran.r-project.org/

Interact with R via the prompt, GUI or scripts.

```
> print('Hello world')
[1] ”Hello world”
```

RStudio provides a more convenient *Integrated Development Environment* (IDE) to interact with R

Layout includes editor, console, workspace/history/plots/packages/files tabs

Convenient user interface with e.g. point-and-click options

You should install RStudio Desktop (available at rstudio.org)

Command prompt

```
> x <- rgamma(3,2,1) # Generate Gamma(2,1) variables
[1] 0.6768685 1.5953583 0.7012949
> z <- sum(x)
[1] 2.973522
> p <- x / z  # Normalize by sum
[1] 0.2276319 0.5365215 0.2358466
> # A random (Dirichlet distributed) prob. vector
> sum(p) == 1
[1] TRUE
```

Command prompt

```
> x <- rgamma(3,2,1) # Generate Gamma(2,1) variables
[1] 0.6768685 1.5953583 0.7012949
> z <- sum(x)
[1] 2.973522
> p <- x / z  # Normalize by sum
[1] 0.2276319 0.5365215 0.2358466
> # A random (Dirichlet distributed) prob. vector
> sum(p) == 1
[1] TRUE
```

```
> # Careful
> 1.2 == 3 *.4
[1] FALSE
> all.equal(1.2, 0.3*4) # Can specify tolerance
[1] TRUE
```

MY_DIRICHLET.R

```r
# Script: Sequence of commands stored in a file.
# Repeatability, releasing code (and submitting homework!)

my_dirichlet <- function(n, shape, rate=1) {
  len <- length(shape)
  x <- matrix(rgamma(n*len, shape, rate),len,n)
  z <- colSums(x)
  p <- t(x) / z   # Column-major ordering
  return(p)
}
```

If you're stuck:

```
> help(rgamma)
> example(rgamma)
```

- R-manual:
  `http://cran.r-project.org/doc/manuals/R-intro.pdf`
- If you've used other languages:
  `www.johndcook.com/R_language_for_programmers.html`
- The internet (google, stackexchange).

R Markdown

Write up homework using `knitr` and R Markdown
(see demo and reading material)

Render Markdown into .pdf and submit to course email

Questions about course/homework:

- First option: post on piazza
- For specific problems, email me or discuss at office hours

G.H. Givens and J.A. Hoeting.
Computational Statistics.
Wiley Series in Computational Statistics. Wiley, 2012.

G. James, D. Witten, T. Hastie, and R. Tibshirani.
An Introduction to Statistical Learning: with Applications in R.
Springer Texts in Statistics. Springer New York, 2014.

C. Lauro.
Computational statistics or statistical computing, is that the question?
Comp. Stat. and Data Analysis, 23(1):191–193, 1996.

N. Matloff and N.S. Matloff.
The Art of R Programming: A Tour of Statistical Software Design.
No Starch Press, 2011.

Paul Teetor.
R cookbook.
O'Reilly, Beijing, 2011.