Stat 133: Concepts in Computing with Data

THEME:
Use the computer expressively to conduct statistical analysis of data.

We will use existing software rather than build routines from the ground up.

We focus on various aspects of computing to conduct statistical analysis,
NOT the computational aspects of statistical methods.

Statistical Thinking in the context of computing with data.

DATA Technologies – Statisticians work includes interfacing/working closely with the original data and those who own it.

What are DATA ?

Numbers

EXAMPLE: Daily precipitation amounts from a network of stations from the Colorado Front Range

- 56 weather stations
- Daily precipitation – hundredths of an inch (4̃00,000 measurements)
- Date – year (1948 to 2001) and day
- Location of weather station – latitude, longitude, and elevation

```
  [1]   0  10  11   1   0   0   0   0   0   0   0   0  10   0   0   0   0   0
 [19]   0   0   0   7  18   0   0   0   0   0   0   0   0   0   0   0   0   0
 [37]   4  15   0   0   0   0   0   0   0   0   0   0   0   3   0   0   0   0
 [55]   0   0   0  45   0   3  28   0   0   0   0  41   2   0   0   0   0   3
 [73]   0   0   0   0 112   0   0   0   0   0   0   0   0   0   0   0   9   2
 [91]   0   0   2  18   0   0   0   0   0   8   7   3   0   0  14  53   0   0
[109]   0   0   0  10   0   0   0   0   0  63   5   0   0   6   0   0   0   0
[127]  66  76   5  13   2   2 103   8  25   0   1   2   0   0   0   0   0   0
[145]   0   0   0   0   4   0   0   0   0   4   6  90 257   2 159   6  18  30
[163]  55   5  33  16   1   0   0   0   0   0   0   0   0   0   0   1  11   4
[181]   0   0   0   0   0  18  22  31  16  25  42   0   2   9   0   0   0  19
[199]   0   0  16   0   0   0   0   0  30   0   0   0   0   0   0   0   0   0
[217]   9   0   9   0   0  25  32   1   9   5   0   0   0   0   4   0   0   0
```

Statistical problem:

GOAL:

- Plan for floods – how should land and roadways be developed?
- Agriculture and vegetation – does precipitation come in a limited series of intense events or is more evenly distributed over many days?
- Climate change – global warming, how will extreme precipitation events change?

Statistical Investigations

- What is the distribution of large precipitation events and how does this distribution vary over space?
- How can irregular station observations be extrapolated to locations where measures are not made?
- How well does a climate model simulation reproduce the features in the observed meteorology?

Text

EXAMPLE: SPAM = Unsolicited, mass, junk email

- $> 50\%$ of electronic mail is SPAM
- Offensive, time-consuming

```
Return-Path: whisper@oz.net
Delivery-Date: Fri Sep  6 20:53:36 2002
From: whisper@oz.net (David LeBlanc)
Date: Fri, 6 Sep 2002 12:53:36 -0700
Subject: [Spambayes] Deployment
In-Reply-To: <LNBBLJKPBEHFEDALKOLCIEJABCAB.tim.one@comcast.net>
Message-ID: <GCEDKONBLEFPPADDJCOECEHJENAA.whisper@oz.net>

You missed the part that said that spam is kept in the "eThunk" and was
viewable by a simple viewer for final disposition?

Of course, with Outbloat, you could fire up PythonWin and stuff the spam
into the Junk Email folder... but then you loose the ability to retrain on
the user classified ham/spam.

David LeBlanc
Seattle, WA USA
```

```
> -----Original Message-----
> From: spambayes-bounces+whisper=oz.net@python.org
> [mailto:spambayes-bounces+whisper=oz.net@python.org]On Behalf Of Tim
> Peters
> Sent: Friday, September 06, 2002 12:24
> To: spambayes@python.org
> Subject: RE: [Spambayes] Deployment
>
> [Guido]
> > ...
> > - A program that acts both as a pop client and a pop server.  You
> >   configure it by telling it about your real pop servers.  You then
> >   point your mail reader to the pop server at localhost.  When it
> >   receives a connection, it connects to the remote pop servers, reads
> >   your mail, and gives you only the non-spam.
>
> FYI, I'll never trust such a scheme:  I have no tolerance for false
> positives, and indeed do nothing to try to block spam on any of my email
> accounts now for that reason.  Deliver all suspected spam to a Spam folder
> instead and I'd love it.
> _______________________________________________
```

Statistical problem:

GOAL: Identify SPAM before we read it.

Use statistical methodology to filter our electronic mail.

- Get sample, classified messages
- Convert or transduce text to response and predictor variables
- Fit statistical model to data
  - use information from mail headers (i.e. sender, routing information, date, return address, etc.)
  - use information in the content of the message body
- Tune the algorithm/model
  - how often do we reject regular message as SPAM?
  - accept SPAM as regular message?
- Deploy classifier as filter.

Images, Sound, Video

EXAMPLE: Traffic flow on highways in California
Video recordings 24-7; Loop detectors at 22,000 locations, transmit data every 30 seconds, collect 2GB a day, and store 4TB

Statistical Thinking and the Data Analysis Cycle

Learn how to think about the *data process*

- Data ACQUISITION – Input/output, regular expressions
- Data CLEANING, verification, and manipulation – graphics, exploratory data analysis
- Data ORGANIZATION – data frames, XML, databases
- MODEL the data – fit statistical models to the data
- Data as a PSEUDO-POPULATION – assess the fit of the model via the bootstrap, cross-validation
- SIMULATED data – simulation studies

In this cycle we encounter:

- Statistical Concepts
- Computing Concepts
- Software

Computing Concepts

- Programming concepts - e.g. loops, recursion, trees
- Regular expressions and text manipulation
- Relational Databases
- Random number generation
- Representation of numbers in the computer
- Event handling and GUI development

Statistical problem:

GOAL: Understand how traffic flows under various road conditions

- What is the distribution of lane occupancy and how does occupancy in different lanes relate to each other?
- When traffic flow breaks down and then recovers at a later time, is the level at which it breaks down higher than the flow level at which it recovers? This phenomena is called hysteresis.
- Researchers validate theories such as hysteresis and calibrate simulation models.

Statistical Concepts

- Graphics
  - elements of graphing data
  - grammar of graphics
  - advanced plotting
- Computationally intensive methods
  - Classification and Regression Trees
  - Kth Nearest Neighbor clustering
  - Thin plate splines
- Simulation tools
  - Bootstrap
  - Cross-validation
  - Monte Carlo Markov Chain

Software

- R - statistical software
- Unix - shell commands
- SQL - Structured Query Language for relational databases
- XML - Extensible Markup language
- Gtk - GNU Toolkit for creating graphical user interfaces

This course will NOT address:

- A course in Computational Statistics,
  For example, we will not focus various algorithms for computing least squares solutions and inverting matrices
- A course in Applied Statistics
  For example, we will not learn a comprehensive set of statistical methods such as ordinary least squares, weighted least squares, and general linear models, etc.
- A course in Mathematical Statistics
  For example we will not cover the expectation, variance, and large sample properties of least squares estimators.

Goals of the Course

- Focus: use existing software and functionality for context-specific analyses.
- Learn about: box of tools and how to use them to create things, and even build new tools.
  Learn about currently emerging technologies
- De-emphasize: understanding the existing algorithms.
  Be able to intelligently discuss different technologies and tools, knowing when to use them and what are the trade-offs
- Understanding fundamental algorithms is important if you need to
  - recreate them in a new language
  - use them in new ways when developing new algorithms.
- Practical: how statistical methodology is used in Industry, Laboratory, Research
- Focus: overall task not just on the application of specific statistical methodology but on how to think about approaching problems related to computing on data

General Information

- Instructor
  - Deborah Nolan
  - Office: 395 Evans
  - Email: nolan@stat.berkeley.edu
  - Office Hours: Mon 4:00-5:00, Fri 1:00-2:00
- GSI
  - Joel Hanson
  - Office: 397 Evans Hall
  - Email: jhanson@stat.berkeley.edu
  - Office Hours: TBA
- USI
  - TBA
- Lab meets on Fridays: 3-4 or 4-5 in 342 Evans

Course Materials

- There is NO textbook for the class.
  No single book covers it all!
- I'll prepare detailed notes or chapters that go into more details than in class.
- Distributed via the Web at
  `www.stat.berkeley.edu/users/nolan/stat133/Fall05`
- Many links to resources on the Web as you need them.
- We will use R as the primary computational environment
- R manuals
  - An Introduction to R `cran.r-project.org/doc/manuals/R-intro.pdf`
  - R Data Import/Export `cran.r-project.org/doc/manuals/R-data.pdf`
  - R Language Definition `cran.r-project.org/doc/manuals/R-lang.pdf`
  - On-line user guides to R, on-line help

Computing Resources

- Statistical Computing Facilities (SCF) – networked computers running Unix
- Undergraduate computer laboratory
  - 342 Evans and 432 Evans
  - Open 8am to 6pm Monday through Friday.
  - Remote access through ssh
- Account from GSI on Friday
- Mailing lists: archive, post.
  - Class mailing list

FIRST ASSIGNMENT: If you have a laptop or computer at home, install R on it by Friday. `cran.us.r-project.org/`

Grading

Participation	5%	Class mailing list and in class
Homeworks	35%	About 7 Short computing assignments
Projects	40%	2 Parts each worth 20% Must be done in groups of 2 or 3 Traffic, GUI development, Rainfall
Final	20%	Written final exam

Expectations: Although there is no computing, probability, or statistics prerequisite for this course, there is an expectation that you have the Curiosity, Initiative and Motivation to Explore on your own and Learn as needed.

There will be the opportunity to learn and receive help from many sources – instructor, GSI, student assistants, fellow students.

Academic Integrity

Code of Student Conduct is available on the web at
`http://students.berkeley.edu/sas/rights.shtml`

- Free to discuss course matters with instructor, GSI, USIs, and fellow students
- Keep the code you write to yourself.
- Make a significant contribution to your group's work.
- Questions: If you are uncertain as to whether something may be a violation of the student code of conduct, ask the instructor.

Writing a program is like writing a paper – your code should be your original work.

Plagiarism

The use of intellectual material produced by another person without acknowledging its source

- Copying from the writings or works of others into one's academic assignment without attribution
- Submitting work of others as if it were one's own
- Using the views, opinions, or insights of another without acknowledgment

Other violations:

- Writing an exam, paper, assignment for another student
- Representing oneself as another person
- Representing, explicitly or implicitly, that work obtained from another source was produced by oneself
- Failure to comply with the instructions or directives of the course instructor

Cheating

Fraud, deceit, or dishonesty in an academic assignment

- Providing answers to or receiving answers from others for any academic assignment.
- In group assignments it is the responsibility of the student to ascertain from the instructor to what degree the work must be done exclusively by the student or may be done in collaboration with others;
- Improperly obtaining or using improperly obtained information about an assignment or assisting others in doing so;
- Putting one's name on another student's assignment
- Altering previously graded work for the purpose of seeking a grade appeal