Independent Study in Idaho

STAT 427:
R Programming

Providing independent study opportunities for more than 40 years.

The University of Idaho in statewide cooperation with
Boise State University — Idaho State University — Lewis-Clark State College

Course Guide

Independent Study in Idaho

Self-paced study. Anytime. Anywhere!

STAT 427
R Programming
University of Idaho
3 Credits

Prepared by:
Dr. Brian Dennis
Professor, Department of Statistical Sciences
University of Idaho

1-STAT 427

Table of Contents

Stat 427 R Programming 3 Semester-Hour Credits: UI

Welcome!

Whether you are a new or returning student, welcome to the Independent Study in Idaho (ISI) program. Below, you will find information pertinent to your course including the course description, course materials, course objectives, as well as information about assignments, exams, and grading. If you have any questions or concerns, please contact the ISI office for clarification before beginning your course.

Policies and Procedures

Refer to the ISI website at **www.uidaho.edu/isi** and select *Students* for the most current policies and procedures, including information on setting up accounts, student confidentiality, exams, proctors, transcripts, course exchanges, refunds, academic integrity, library resources, and disability support and other services.

Course Description

This course is an introduction to the R computer language for scientific graphics, statistical analysis, simulation, and mathematical modeling. Topics include assignment statements, scripts, objects, functions, data management and manipulation, loops and logical structures, vector and matrix calculations, contemporary graphical displays, probability and simulation, dynamic models, numerical optimization, and standard methods of statistical analysis.

Credit not awarded for Stat 424 after Stat 419.

Prerequisite: Stat 251 (3 credits), or Stat 301 (3 credits), or Stat 416 (3 credits) or any equivalent undergraduate introductory statistics course.

15 graded assignments,1 project paper, 0 (Zero) proctored exams
Available online only.

Students may submit up to two assignments per week; however, **assignments and exams must be submitted consecutively, in the order outlined in the course.**

ALL assignments plus project paper must be submitted to receive a final passing grade for the course.

Course Materials

Required Course Materials

- Dennis, B. 2013. *The R Student Companion.* CRC Press, Boca Raton, Florida USA. ISBN: 978-1-4398-7540-7.
- Wickham, H. 2016. *ggplot2: Elegant Graphics for Data Analysis, Second Edition*. Springer, Switzerland. ISBN: 978-3-319-24275-0.
- The student will need regular access to a semi-contemporary laptop or desktop computer (Windows, Mac, or Linux) on which a semi-recent version of R is installed and running.

Course Delivery

All ISI courses are delivered through Canvas, an online management system that hosts the course lessons and assignments and other items that are essential to the course. Upon registration, the student will receive a *Registration Confirmation Email* with information on how to access ISI courses online.

Course Introduction

R is a computer programming language and package for scientific graphs, calculations, statistical analysis, and mathematical modeling. It was written by scientists, for scientists to use in their work. It is easy to use yet is extraordinarily powerful. R is spreading rapidly throughout the science and technology world (see the article in the New York Times Jan 7 2009 linked in Lesson 1), and it is setting the standards for graphical data displays in science publications.

R is free. It is an open-source product that is easy to install on most computers. It is available for Windows or Mac machines, as well as for Linux/Unix machines. One simply downloads and installs it from the R website (http://www.r-project.org/). At present, there are no R versions for chromebooks or tablets (although there are websites with R servers on which one can run small R programs).

While students in all STEM fields will benefit from knowing R, STEM teachers especially will find that R opens up many possibilities for making science and math exciting for students. Calculating in R is intuitive and fun. Fantastic, publication-quality graphs of data, equations, or both can be produced with little effort. Every high school or college course currently relying on graphing calculators or spreadsheet programs would benefit from using R instead.

I believe R will revolutionize the teaching of introductory college and high school courses in science and mathematics ("science" interpreted broadly to include research-based social sciences, and "mathematics" interpreted broadly to include statistics). This course is intended to get that revolution started! Indeed, R is easy and fun enough to use in middle school and even elementary school.

This course provides an introduction to programming and analysis with R. The material in the course blends programming concepts, scientific graphs, mathematical ideas, and statistical methods toward solving real scientific problems. Applications are drawn from many areas of science. The science and math content are middle- and high-school level, but the approaches are commensurate with daily tasks of researchers all over the world. The skills acquired in this course will carry over into future college courses and professional work and will help to gold-plate any professional resume.

The course is an active course! Readings are not meant to be read; they are meant to be *done*! You should progress through each lesson with a computer at hand, typing/running every R command and evaluating the result.

There are no tests (life will test you plenty on these topics!). Instead, there are work-at-home computational assignments accompanying each lesson as well as a final project paper.

Course Objectives

Learning outcomes:

- Students completing this course will be able to solve most statistical analysis problems appearing in standard undergraduate statistics textbooks with the R software.
- Using the R software, students will be able to prepare excellent publication-quality graphical displays of data as well as undertake substantial scientific calculations and simulations in support of scientific work.
- Students will be able to understand and use basic computer programming concepts such as assignment statements, scripts, logic and control, loops, data management and manipulation, objects.

Lessons

Overview

Each numbered lesson includes a reading and a computational/written assignment. The assignments consist of computational tasks to be performed with R that illustrate, develop, or extend the lesson. You should hand in, via Canvas, the R code you used for the assignment, the output of the code, as well as typed written responses to any questions posed in the assignment. The assignment portions should be assembled and submitted to Canvas in the form of a either an MS-Word file (docx or doc file), a pdf file, or an html file. Each lesson is the rough equivalent of 1 week of material in the on-campus semester version of the course (usually 3 class meetings per week, plus homework time), use this fact to scale your personal pacing.

The course project can take many forms, including a data analysis that heavily relies on R (with a report that focuses on the results of the analysis), or a project that explores an unfamiliar aspect of R not covered in the course, such as a contributed function for a specialized analysis or some more advanced programming feature (with a mostly expository report). More details can be found in the Final Project Information section.

The course will require considerable independent study in the form of learning by reading and doing, without the benefit of lectures. However, I welcome questions! I can usually get to answering questions within 24 hours, unless I am traveling away from computers.

Each lesson includes the following components:

- lesson objectives
- reading assignment(s)
- written computational assignment to be submitted

Study Hints:

- Keep a copy of every assignment submitted.
- Complete all reading assignments.
- Set a schedule allowing for course completion one month before your personal deadline. An *Assignment Submission Log* is provided for this purpose.
- Web pages and URL links in the World Wide Web are continuously changing. Contact your instructor if you find a broken Web page or URL.
- The concepts in the course are cumulative; each successive lesson builds on previous lessons.
- This is a course for beginners at computing; concepts will likely be new and unfamiliar. You can expect frustration at first, but you also can expect your learning to snowball as we go!

Refer to the ***Course Rules*** in Canvas for further details on assignment requirements and submission.

Grading

The course grade will be based upon the following considerations:

Assignments	Points	Percentage
Each assignment	10	5
Total	150	75

Project	Points	Percentage
Project	50	25
Total	50	25

A = 90% to 100%
B = 80% to 89%
C = 70% to 79%
D = 60% to 69%
F = 59% or less

The final course grade is issued after all assignments and the project paper have been graded.

Acts of academic dishonesty, including cheating or plagiarism, are considered a very serious transgression and may result in a grade of F for the course.

About the Course Developer

Your course developer and instructor is Brian Dennis, a professor at the University of Idaho with a joint appointment in the Department of Fish and Wildlife Sciences and the Department of Statistical Science. Dr. Dennis has a master's degree in statistics and a Ph.D. in ecology, both degrees from Pennsylvania State University. His research interests include mathematical modeling, biometrics, statistical ecology, stochastic processes, and computer simulation, especially toward solving pressing problems in ecology and natural resource management. He has authored or co-authored more than 100 scientific publications, including papers in Science, Nature, Proceedings of the National Academy of Science USA, Proceedings of the Royal Society, Biometrics, Ecology Letters, Ecology, and Ecological Monographs. More information here:
https://www.uidaho.edu/cnr/faculty/dennis

Contacting Your Instructor

Instructor contact information is posted on your Canvas site under *Course Rules.*

Assignment Submission Log Template

Lesson	Projected Date for Completion	Date Submitted	Grade Received	Cumulative Point Totals
1				
2				
3				
4				
5				
6				
7				
Begin discussion of ideas for your project with the instructor around midway through course or earlier. Conduct project, document results, outline report, draft report, revise report, simultaneously with subsequent lessons.				
8				
9				
10				
11				
12				
13				
14				
15				
Be putting final touches on project paper, with time to spare. Submit final project report.				
Project Report				

Lesson 1
About R, and Getting Started in R

Lesson Objectives

In Lesson 1, you will:

- Learn the origins of R and gain an overview of its capabilities.
- Understand arithmetic calculations, assignment statements, variable names, vectors, vector arithmetic, and simple graphs (equation, scatterplot).

Reading Assignment

- New York Times article, *"Data analysts captivated by R's power"* Jan 7 2009:
- http://webpages.uidaho.edu/~brian/R_NYT_article.pdf
- Presentation about R: http://webpages.uidaho.edu/~brian/what_is_R.pdf
- *The R Student Companion* (abbreviated hereafter as *RSC*): Preface, and Chapter 1.

Reading notes:

There is a typo near the bottom of RSC p. 11. The R statement given by "k=b*m/(a+m)" should be "k=a*m/(b+m)" (a and b are interchanged).

Important Terms

Arithmetic symbols
Assignment statements
Adding to plots with point()
Order of operations
Vectors
R console
Variable names
Simple graphs with plot()

Written Assignment

Before beginning the first written assignment, refer to the **Lessons** section above for your instructor's assignment requirements. If emailing assignments to your instructor, please copy the ISI office at *indepst@uidaho.edu.*

Assignment for Lesson 1:

- Do Computational Challenges 1.1 & 1.2 (*RSC* pp 14-15), and your choice of 1 additional CC in the set.
- Copy/paste your R commands & the resulting output from the console into a word processor document. Submit the document to Canvas in the form of a Word file (doc or docx), pdf file, or html file.

Lesson 2
R Scripts

Lesson Objectives

In Lesson 2, you will:

- Write, run, debug, and save R scripts!
- Optionally, learn the math of interest payments!

Reading Assignment

- *RSC*, Chapter 2.

Reading notes:

- All scripts used in *RSC* are available at the book website:
  http://webpages.uidaho.edu/~brian/rsc/RStudentCompanion.html

- A free shell or environment for working with R called RStudio has become popular in recent years among R users. It has windows for managing scripts, plots, and files. This course does not cover RStudio, in order to focus better on R programming basics. You are welcome to use RStudio for this course, but realize there are differences between working in RStudio and working in "vanilla" R. Those who begin working with R a lot after this course will likely want to pick up RStudio. It can be downloaded at: https://www.rstudio.com/

Important Terms

R script	R editor	objects()
rm()	source()	Loan principal
Loan interest	Geometric series	

Written Assignment

Assignment for Lesson 2:

- Re-do each of the three parts of the Lesson 1 assignment as R scripts.
- Pick one additional CC at the end of Lesson 2 (*RSC*, pp 36-39) and write an R script to solve it.

Lesson 3
Functions in R

Lesson Objectives

In Lesson 3, you will:

- Learn about the many pre-programmed functions existing in R for convenient calculation of numerous mathematical and statistical quantities.
- Know how to build your own functions in R!

Reading Assignment

- *RSC*, Chapter 3.

Reading notes:

- One of the extraordinary features of R is the enormous and fast-growing library of user-contributed functions for use in R for various specialized and super-sophisticated analyses. Whether you need to perform a Black-Scholes pricing for your next stock futures purchase, or need a generalized linear mixed model analysis, or need to numerically solve a system of nonlinear differential equations, there is an R function that has been written for whatever you want to do. You will find your function in the "Comprehensive R Archive Network" (CRAN): https://cran.r-project.org/

Important Terms

R function	Argument	NaN
function() { }	return()	Local and global variables

Written Assignment

Assignment for Lesson 3:

- CCs 3.1 and 3.2 (*RSC*, pp 49-50).
- Pick one additional CC at the end Chapter 3 and use R to solve it.

Lesson 4
Basic Graphs

Lesson Objectives

In Lesson 4, you will:

- Learn to draw various types of graphs in R.

Reading Assignment

- *RSC*, Chapter 4.

Reading notes:

- This lesson covers "Base R Graphics", which are simple and quite serviceable for many tasks. An extraordinary set of graphical resources has been added to R in the form of the "ggplot2" package. Later lessons in this course will introduce some capabilities of ggplot2, which is based on a unified philosophy of good graphical portrayal.

Important Terms

Stripchart: stripchart()
Stem-and-leaf plot: stem()
Pie chart: pie()
Logical vector
Histogram: hist()
Boxplot: boxplot()
Quantitative variable (or vector)
Scatterplot: plot()
Bar graph: barplot()
Categorical variable

Written Assignment

Assignment for Lesson 4:

- CCs 4.1, 4.2,and 4.4 (*RSC*, pp 74-75).

Lesson 5
Data Input and Output

Lesson Objectives

In Lesson 5, you will:

- Bring data from an external file into R for plotting or analysis.
- Learn about "data frames" in R.
- Write data created in R into an external file.

Reading Assignment

- *RSC*, Chapter 5.

Reading notes:

- Since R was created, the data-analytics revolution has swept the world. Years of experience have taught analysts to pare down data management into simple and easily-followed operations. Data sets in a format like a data frame that are easily manageable are called "tidy" data, and a suite of R packages for producing and working with tidy data are contained in the "tidyverse", a super-package that you will install and work with starting in Lesson 7.

Important Terms

Data frame
read.table()
working directory
attach()
data.frame()
detach()
write.table()

Written Assignment

Assignment for Lesson 5:

- CCs 5.1, 5.2, and 5.3 (*RSC*, p 88).

Lesson 6
Loops

Lesson Objectives

In Lesson 6, you will:

- Learn to use a fundamental programming concept: loops!
- Work with an important application of loops in mathematical modeling: projection of quantities in a dynamic system through time!

Reading Assignment

- *RSC*, Chapter 6.

Reading notes:

The last line in the data table in CC 6.3 (p 102) should read:

lg adult	.991	--	479

Important Terms

For-loop	for () { }	numeric()
Fibonacci	Mathematical recursion	Population model
Northern Spotted Owl		

Written Assignment

Assignment for Lesson 6:

- CC 6.1 (*RSC*, pp 100-102).
- Pick one additional CC at the end Chapter 6 and use R (and a loop!) to solve it.

Lesson 7
Logic, Control, and Data Manipulation

Lesson Objectives

In Lesson 7, you will:

- Learn how to use logical comparison operators.
- Learn how to use Boolean operations.
- Learn how to use conditional statements.
- Combine the above in order to pick out subsets of a data frame and construct new data frames.
- Be introduced to the "tidyverse"!

Reading Assignment

- *RSC*, Chapter 7.

Once you are familiar with the Chapter 7 material (the concepts of which remain important if you learn other programming languages), you are invited/urged to start reading about and working with the "tidyverse", a collection of R packages which make "data wrangling" much easier. Install the tidyverse on your system (if it is not already there) with the following R command: **install.packages("tidyverse")**
Then, at the beginning of an R session, or at the beginning of a script, issue the following R command to load the tidyverse into your work session: **library(tidyverse)**
The tidyverse contains the ggplot2 package for graphics which you will utilize in Lessons 12 and 13 as well as the dplyr package which contains handy functions for data management.

- An easy tutorial (& there are many others online as well) for the tidyverse can be found here: https://www.datacamp.com/community/tutorials/tidyverse-tutorial-r
- An online free book about working with data in R using the tidyverse can be found here: https://r4ds.had.co.nz/

Reading notes:

- In Table 7.1 (p 116), the header should read:
  Biofeed   Drug   Special Diet No   Special Diet Yes

- In CC 7.1 (p 125), the 1988 winning time in the data table should read 216.0 (not 16.0).

- Do not be overwhelmed by the sheer volume of possibilities contained in R. Learn the basics of logic and control, and you can then add skills and packages to your repertoire as needed in your own work. Part of the fun of using R is being continually amazed at the new stuff that is constantly being added to R.

Important Terms

Logical comparison operator
Inf and -Inf
if ( ) { } else { }
Profile plot

Boolean operations
Indexes
Uniform random numbers
tidyverse

Missing data: NA
Conditional statements
runif()

Written Assignment

Assignment for Lesson 7:

- CCs 7.1 and 7.2 (*RSC*, p 125).
- Pick one additional CC at the end Chapter 7 and use R to solve it.
- Optionally learn and use the dplyr package (within the tidyverse) to do the Lesson 7 assignment, if you are comfortable already with basic logic and control statements.

Lesson 8
Exploring Quadratic and Trigonometric Functions

Lesson Objectives

In Lesson 8, you will:

- Review some special functions from high school algebra, using R to explore and visualize their properties.
- Learn some important scientific applications of these functions.

Reading Assignment

- *RSC*, Chapters 8 and 9. (While the volume of reading is larger than usual, the math concepts are a review from high school math. Hopefully you will find the concepts to be retold in scientifically compelling ways.)

Reading notes:

Arrg, in Figure 8.2 (p 128) , the label of $1-x$ in the lower right of the rectangle should be $x-1$.

Important Terms

Golden rectangle
Parabola
Right triangle
Polar coordinates
Eccentricity

Golden ratio
Yield-effort curve
sin, cos, tan, csc, sec, cot
polar curve
parallax

Golden section
Pythagorean theorem
Radians
Periapsis

Written Assignment

Assignment for Lesson 8:

- CC 8.1 (*RSC*, p 140), 9.1, 9.2 (*RSC*, p 161-162).

Lesson 9
Exploring Exponential and Logarithmic Functions

Lesson Objectives

In Lesson 9, you will:

- Review some more special functions from high school algebra, using R to explore and visualize their properties.
- Learn some important scientific applications of these functions.

Reading Assignment

- *RSC*, Chapter 10.

Reading notes:

Exponential and logarithmic functions are key concepts. Many applications of mathematics for solving important problems in the sciences use exponential and logarithmic functions. The author's experience is that the exponential and logarithmic functions (along with power functions) are insufficiently covered in high school and even college algebra, thus denying students the opportunity to understand the quantitative basis of science.
The many compelling uses of exponential and logarithmic functions answer the students' perennial question: "why do we have to know this?".

Important Terms

Power function	The number e	Exponential function
exp(), log(), log10()	Logarithmic function	Exponential growth
Exponential decay	Logarithmic scale	Richter scale
pH	Star magnitude	Radionuclide dating
Logistic growth	Peak oil	

Written Assignment

Assignment for Lesson 9:

- CC 10.2, 10.3, and 10.4 (*RSC*, p 190).
- Pick one additional CC at the end Chapter 10 and use R to solve it.

Lesson 10
Matrix Arithmetic

Lesson Objectives

In Lesson 10, you will:

- Learn to add, subtract, and multiply matrices!
- Learn to use matrices to solve systems of linear equations and to fit lines to data!

Reading Assignment

- *RSC*, Chapters 11, 12.

Reading notes:

- In CC 12.3 (p 227), the 1988 winning time in the data table should read 216.0 (not 16.0).

- In Appendix C of *RSC* (p 323), the first entry in the "Matrices" section should read:

  A=matrix(x,m,n): Vector x is read into an m by n matrix A, column by column.
  Values in x are recycled.

  Matrix math might be new to you, but the basics of it are quite easy: matrices are just a way of organizing vast amounts of ordinary arithmetic calculations. Matrices are hugely important in science for allowing numerically intensive computations to be represented and communicated symbolically. The point is: R is *designed* to do matrix calculations, and so why not join in the fun of performing millions of calculations with a single line of R code? As a bonus to learning some matrix math, you will see the inner workings of linear regression.

Important Terms

Dot product
Scalar multiplication
Northern Spotted Owl
solve()
Sum of squared errors
Matrix
matrix()
Projection matrix
Response variable
Least squares
Matrix product: %*%
Matrix addition, subtraction
Matrix inverse
Predictor variable
Cepheid star

Written Assignment

Assignment for Lesson 10:

- CC 11.2 plus choice of 11.3 or 11.4 (*RSC*, p 205-206).
- CC 12.2 (RSC p 226-227) plus choice of 12.3, 12.4, or 12.5.

Lesson 11
Probability and Simulation

Lesson Objectives

In Lesson 11, you will:

- Learn about random variables and use random numbers generated in R to model processes that have randomly varying outcomes.
- Simulate results in probability theory such as the law of large numbers and the central limit theorem.
- Learn about the binomial, uniform, and normal probability distributions and the functions for working with them in R.

Reading Assignment

- *RSC*, Chapter 14.

Important Terms

Probability
Statistics
Law of large numbers
Binomial distribution
dbinom(), pbinom(), rbinom()
Uniform distribution
runif()
Normal distribution
dnorm(), pnorm(), rnorm()
randomly fluctuating prices

Written Assignment

Assignment for Lesson 11:

- CC 14.1 and 14.2 (*RSC*, p 273).

Lesson 12
Advanced Graphs with ggplot2

Lesson Objectives

In Lesson 12, you will:

- Learn to use the ggplot() function (contained in the ggplot2 R package) to produce a variety of graph types.
- Be intrigued by the "grammar of graphics" as a unifying philosophy of data portrayal.

Reading Assignment

- *ggplot2*: *Elegant Graphics for Data Analysis* (abbreviated hereafter as *ggplot2*) Preface, Chapters 1 and 2.

Reading notes:

- The ggplot2 package is one of the many packages contained in the tidyverse. You can install ggplot2 separately if you have not installed the tidyverse. At the beginning of your R session, either of the statements **library("tidyverse")** or **library("ggplot2")** will load ggpplot2 for your use.

- The grammar of graphics finds that different graph types, such as boxplots, histograms, scatterplots, etc., have fundamental aspects in common. These aspects are exploited in the ggplot2 package, so that one needs just one function, ggplot(), to produce many kinds of graphs. By contrast, Base Graphics in R require separate functions (boxplot(), hist(), plot(), etc) for each graph type.

Important Terms

Grammar of graphics
coord
aesthetic mapping
Layers, geoms, stats
facet
ggplot()
scales
theme

Written Assignment

Assignment for Lesson 12:

- *ggplot2* exercises 2.3.1.3 (p 14), 2.5.1.2 (p 17), 2.6.6.3 and 2.6.6.4 (p 27).

Lesson 13
Advanced Graphs with ggplot2, Part 2

Lesson Objectives

In Lesson 13, you will:

- Extend your skills in using the ggplot() function.
- Learn more about the grammar of graphics and how one builds different graph qualities in ggplot().

Reading Assignment

- *ggplot2*: Read Chapter 3 in detail, and then look through Chapters 4 and 5 for future reference for your own work.

Reading notes:

- A whole course could be taught around ggplot2. We can only get your feet wet in ggplot2 within the scope of this course. Hopefully this introduction will provide enough background to allow you to explore the capabilities of ggplot2 more fully on your own.

- Chapters 9 and 10 of *ggplot2* provide an introduction to data manipulation using the tidyverse and the dplyr package.

Important Terms

Grammar of graphics
Coord
aesthetic mapping

Layers, geoms, stats
facet
ggplot()

scales
theme

Written Assignment

Assignment for Lesson 13:

- Repeat the assignment for Lesson 4, using ggplot()!.

Lesson 14
Statistical Methods: Inferences for Means and Proportions

Lesson Objectives

In Lesson 14, you will:

- Start using R for statistical analysis!
- Obtain numerical summaries of variables.
- Obtain confidence intervals and perform hypothesis tests for means using t statistics in R.
- Obtain confidence intervals and perform hypothesis tests for proportions using z statistics in R.

Reading Assignment

- Review material on inferences for means (t-based confidence intervals and hypothesis tests) and proportions (binomially-distributed data) in any undergraduate statistics textbook. An excellent, and free, online text, *OpenIntro Statistics, Third Edition*, can be downloaded (pdf) here: https://www.openintro.org/

Reading notes:

- Statistical methods are coded as functions in R. For instance, t.test() provides a variety of inferences for means based on t statistics. Variables (columns of data frames) in the form of vectors are the basic argument in the statistical method functions. Typically the statistical method functions do not print anything automatically. Rather, the functions return a list of output items. The user must decide what is to be done with the list. This non-printing may sound weird at first, but realize that R is meant for flexibility. For instance, you might be simulating the performance of a statistical method (perhaps for a brute force power calculation) by generating thousands of data sets—you would only want to pick out of the output those part of interest and would not want everything printed. The summary() statement, when the output of a statistical method function is the argument, usually prints to the console most of the things you want to see in an individual analysis.

The analyses in this lesson can be performed with the following scripts (often accompanied by example data, which will need to be transferred as files to your computer):

R Scripts for Statistical Analysis

Summarizing quantitative data
http://www.webpages.uidaho.edu/~brian/R_summary_statistics.pdf

Summarizing categorical data in tables
http://webpages.uidaho.edu/~brian/categorical_data_summaries.txt
Data: http://webpages.uidaho.edu/~brian/student_survey.txt (social survey of student life)

Inferences for one and two means (the script pieces for each analysis can be run separately)
http://webpages.uidaho.edu/~brian/one_and_two_means.txt
Data:
http://webpages.uidaho.edu/~brian/textbooks.txt (textbook prices: UCLA and Amazon)

http://webpages.uidaho.edu/~brian/androsterone.txt (Margolese 1970 data on androsterone (testosterone) levels in straight and gay males)

One proportion
http://webpages.uidaho.edu/~brian/one_proportion_test.txt

Two proportions
http://webpages.uidaho.edu/~brian/two_proportions_test.txt

Important Terms

Distribution mean (μ)
Sample mean ($\overline{x}$)
Student t distribution
Null and alternative
Success probability p

Distribution variance (σ^2)
Sample variance (s^2)
Confidence interval
Test size α
P-value of a test

Random sample of size n
Normal distribution
Hypothesis test
Binomial distribution

Written Assignment

Assignment for Lesson 14:

- Run the above scripts. Organize: script, data (header + first few observations), output, for each analysis method into a Word file (for your future reference!). Insert notes/comments as desired for your future reference. Submit Word file or pdf.

Lesson 15
Statistical Methods: Linear Regression

Lesson Objectives

In Lesson 15, you will:

- Start using R for regression analysis!

Reading Assignment

- Review material on linear regression and its accompanying inferences (least squares estimates, confidence intervals and hypothesis tests for intercept and slope coefficients in the regression model, confidence interval for the regression line, prediction interval for a new observation) in any undergraduate statistics textbook. An excellent, and free, online text, *OpenIntro Statistics, Third Edition*, can be downloaded (pdf) here:
  https://www.openintro.org/

Reading notes:

- You have seen already how the estimates of the intercept and slope parameters in a linear regression model are the least squares estimates, found by solving a system of two linear equations in two unknowns. You have also seen how to perform those calculations using matrices in R. Conveniently, all of the calculations in linear regression are contained in the lm() (linear model) function in R. With two variables, Y (response variable, or variable that is to be predicted) and X (predictor variable, or variable used to help predict the response variable, sometimes called a covariate of Y), regression is easy. The R statement

  lm(Y~X)

  fits the linear equation of the form

  $$Y = \beta_0 + \beta_1 X_1$$

  to the data, where β_0 and β_1 are respectively the intercept and slope parameters.

  But lm() does much, much more, including multiple regression. Why stop with one predictor variable when you could incorporate two or more? If X1 and X2 are potential predictor variables (say, temperature and humidity), then lm(Y~X1+X2) fits a linear model of the form

  $$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2$$

  to the data.

  More discussion about fitting linear and nonlinear model equations to data using R is contained in Chapter 15 of *RSC*. This is strictly optional reading for the course which you might find useful in future work.

The analyses in this lesson can be performed with the following scripts (accompanied by example data, which will need to be transferred as files to your computer):

R Scripts for Statistical Analysis

Linear regression
http://webpages.uidaho.edu/~brian/linear_regression.txt
Example data are already contained in R as the "faithful" data set.

Multiple regression: each sub-script can be run individually. The last function, regsubsets(), fits all possible models (all possible combinations of predictor variables) and requires the installation of the "leaps" package from CRAN.

http://webpages.uidaho.edu/~brian/multiple_regression.txt
Data: http://webpages.uidaho.edu/~brian/lungpressure.txt

Important Terms (Notation may vary from stat text to stat text)

Parameters β_0 and β_1
Predicted value
R^2
Least squares
Residual
Outlier
LS estimates b_0 and b_1
Correlation R
Leverage

Written Assignment

Assignment for Lesson 15:

- Run the above scripts. Organize: script, data (header + first few observations), output, for each analysis method into a Word file (for your future reference!). Insert notes/comments as desired for your future reference. Submit Word file or pdf.

Final Project Information

Final product to be handed in:
Typed project report, double spaced, 12 point font, 1 inch margins, no more than 8 pages of text (not including figures, tables, references, appendix). Report should be submitted at least two weeks before you need a grade reported for the course.

- Report can be in any technical format familiar in your field.
- Figures & tables, if used, should have descriptive captions and be numbered consecutively, and placed after the list of references (if any).
- Excessive technical details, such as lengthy R code, can be put in an appendix (at the very end of the document).
- Write at a technical level appropriate for peers in your field.
- Report should be submitted as an MS-Word file (doc or docx) or as a pdf file.
- If the report involves many equations & math symbols, you are encouraged to prepare the report using Latex (but submit the report as a pdf file).

Project scope:
Projects can take many forms. The main charge is for the project to use R in some significant way.

- Example: project could involve an analysis of an existing substantial data set. Report would focus on the analysis results, with an appendix showing code for analyses & supplemental figures.
- Example: project could involve an exposition of an advanced or specialized aspect of R not covered in the course (such as genetics statistics, financial analyses, mapping, mark-recapture, etc.). Report would be instructional/expository, with excellent examples, showing the reader how to use R for such tasks.
- You are encouraged to discuss ideas for your project with the instructor.
- Graded reports will be returned to you with comments. Grading will be a 50-50 blend of project execution and communication quality.