1 Getting Onto S-Plus

1. Log onto your unix account.

2. At the ">" prompt, type 'SPlus' (with a capital S).

3. You will be asked if you want to create a working directory called Data. Answer yes. This is a directory where all of your S-Plus files and datasets will be stored.

4. You are now ready to use S-Plus.

2 Simple Commands

S-Plus treats its commands as functions that take arguments. That is, nearly all S-Plus commands require parentheses to follow, containing any arguments. For example, if you have a set of test scores, called scores, and you want to know the test average, type 'mean(scores)'. The mean function uses the name of a dataset as its argument. Some functions can take more than one argument, although all arguments are not necessary. To find out what arguments a function takes, type 'help(function_name)'.

One command that is necessary for S-Plus is the quit command. The quit command is a '.' but because it is a function, it requires parentheses, but no arguments. So, to exit S-Plus, simply type 'q()'.

Some other helpful commands are:

mean() Finds the mean of a dataset.
var() Finds the variance of a dataset.

stem() Produces a stem and leaf display.

hist() Produces a histogram. (See further info. below)

set.seed() Sets the seed in the random number generator.

rnorm() Generates a sample from a normal distribution.

runif() Generates a sample from a uniform distribution.

read.table() Reads in a dataset from outside S-Plus.

scan() Reads in a dataset from outside S-Plus.

c() Combines datasets.

sqrt() Takes the square root of a number.

postscript() Opens a file for graphics.

Below is a sample S-Plus session where the user inputs 20 test scores,
finds their mean, standard deviation, and generates a histogram.

> scores <- c(98, 97, 87, 67, 100, 76, 77, 76, 92, 93, 88, 83, 81,
           82, 86, 87, 87, 90, 72, 79)
> mean(scores)
[1] 84.9
> sqrt(var(scores))
[1] 8.831165
> stem(scores)

N = 20  Median = 86.5
Quartiles = 78, 91

Decimal point is 1 place to the right of the colon

  6 : 7
  7 : 26679
  8 : 12367778
  9 : 02376
10 : 0

> q()
Notice the use of the ‘c()’ command, which allows me to create a dataset. The ‘<-’ assigns the set of numbers to a variable called scores. The ‘<-‘, which is just a ‘<‘ followed immediately by a ‘-‘, allows assignment.

3 Getting A Hardcopy

This handout assumes that you are simply logging on to jhunix from a terminal, for example from your mom, and that you don’t necessarily have graphics capabilities. However, numerous S-Plus commands produce graphical output, for example, ‘hist()’. This section will discuss how to get printout of your S-Plus session, as well as any graphics required.

First, to get a printout of your session (i.e., the commands and output) follow these instructions.

1. Before getting onto S-Plus, at the jhunix ‘$’ prompt, type ‘script filename’. You will receive a message saying that the script file was started.

2. Get onto S-Plus and take the necessary steps.

3. Quit S-Plus.

4. At the ‘$’ prompt, type ‘exit’. This will not log you off jhunix, it will end the script file.

5. You now have a text file called filename that you can edit. This will be necessary because each time you hit a backspace, the script file records the mistake and the backspace as a `'H'. You should clean up this file before turning it in. You can edit the file using vi, emacs, pico (if you use the pico mailtool), or some other editor.

To obtain graphical output, follow these instructions.

1. While in the middle of your S-Plus session, you create a postscript file to hold the graphics. At the S-Plus prompt ‘>’, type ‘postscript("filename.ps")’. You will not receive any messages after typing this.

2. Now, simply type any commands as you normally would and any commands that yield graphical output will be sent to the postscript file.

3. When you finish your session, quit S-Plus. Doing this also closes the postscript file. Because this is a graphics file, you will not be able to edit it, just print it.
4. If you have done this successfully, you will have to print 2 files (using the lpr command, which prints to the lab in Kreeger), the script file (which can be edited) and the postscript file (which contains the graphics).

4 Before You Begin An Assignment

Make sure that you can get into Splus and execute simple commands, like those above. Please do this well before your first computer assignment, otherwise, you may encounter problems and not have time to get assistance.

When trying these commands, please explore their arguments using the ‘help()’ function. This will greatly help you in the assignments, for example you can provide labels on histograms, or be more specific about what type of distribution you sample from, whether it is $N(0, 1)$ or $N(10, 3)$, etc.

Also, be sure to edit your script file, and at the top include your name and social security number.

5 Teaching Assistants

The following is a list of TAs for the course, their sections, office hours, and email addresses:

- **Chris Wagner (Head TA)**
  - Office Hours: Mon. 9-10
  - Section: Thur. 9-10 (MD 201)
  - email: wagner@brutus.mts.jhu.edu

- **Melissa Agocs**
  - Office Hours: Wed. 1-2
  - Section: Fri. 10:30-11:30 (MD 309)
  - email: mmaagocs@jhuunix.hcf.jhu.edu

- **Max Buot**
  - Office Hours: Fri. 11-12
  - Section: Thur. 1-2 (MD 309)
  - email: buot@brutus.mts.jhu.edu

- **Laura Glaenger**
  - Office Hours: Tues. 2-3
    - Wed. 6-9
Thur. 11:30-12:30
Section: Thur. 10:30-11:30 (MD 309)
Fri. 1-2 (MD 201)
email: glasesge@brutus.mts.jhu.edu

- Maryanne Harkins
  Office Hours: Mwn. 3:30-4:30
  Mwn. 6-9
  Section: Thur. 12-1 (MD 201)
  Thur. 2-3 (MD 201)
  email: harkins@brutus.mts.jhu.edu

- Aleksey Kovalyev
  Office Hours: Thur. 1-3
  Thur. 6-9
  Section: None
  email: kovalyev@brutus.mts.jhu.edu

- Alvin Lim
  Office Hours: Wed. 2-3
  Section: None
  email: lim@brutus.mts.jhu.edu

- Vivian Yuan
  Office Hours: Mwn. 1-2
  Tues. 1-2
  Tues. 6-9
  Section: Wed. 1-2 (Shaffer 101)
  email: yuan@brutus.mts.jhu.edu