

Computing in Weber Classrooms 205-206:

1. To log in, make sure that the DOMAIN NAME is set to MATHSTAT.
2. Use the class username: st192 The password will be distributed in class.
3. On most of the machines, your work should be saved to the folder C:\temp (on some machines, this folder will be D:\temp) This is a temporary location for class work which is emptied periodically. Work that you would like to keep should be stored on a diskette.
4. Course material (data, homework files, etc.) can be found in the subfolder st192 on drive G:
5. Folders can be examined and navigated using Explore (click the right button over the Start button and select Explore).
6. At the end of class, make sure that you logout from the computer (Start > Shutdown > Close all programs and logon as a different user.)

Running Splus:

1. Click on the Splus icon on the desktop.
2. On some machines, this will be the first time that Splus has been invoked and a setup question will appear inquiring about the location of a work or data directory. Answer yes to the suggested default location (most likely this will be C:\temp).
3. If Splus opens in a blank shell, i.e., no windows are opened, click on the "Commands Window" icon in the toolbar. (The icon looks like two > signs on top of each other followed by an x and a vertical line.)
4. Open an Object Explorer window by clicking on the "Object Explorer" icon in the toolbar. Expand the Data folder (click on the + sign) in the Object Explorer window.
5. Make the Commands Window active (click on the title bar) and try out the following commands:

Basic manipulations:

```

> 2+3
> x <- 2+3      [answer is stored in the vector x]
> x             [displays the contents of x]
> x <- c(10.1, 3.2,4.5, 6.7, 8.9, -1.2)   [overwrites x as a vector containing 6 numbers]
> x[3]         [displays the third entry of x]
> mean(x)      [calculates the mean of x]
> m <- mean(x) [stores the mean of x in m]
> x/var(x)^.5  [computes divided by its standard deviation]
> y <- 1/x     [stores the reciprocal of x in the vector y]
> z <- 2*x + y [computes the vector z consisting of 2*x + y]
> z           [displays z]

```

Plots:

```

> plot(sin(seq(0,pi,.1)))           [plots sin(x) for x from 0 to pi in increments of .1]
> plot(seq(0,2*pi,.1), sin(seq(0,2*pi,.1))) [plots (x, sin(x)) for x from 0 to pi in
                                         increments of .1]
> plot(seq(0,2*pi,.01), sin(seq(0,2*pi,.01)),type="l") [plot with just lines]
> lines(seq(0,2*pi,.01), cos(seq(0,2*pi,.01)))          [appends plot with another]

```

Loops:

```
> sumseries <- 0           [initializes vector sumseries to 0]
> for (i in 1:100) sumseries <- i + sumseries   [sums the numbers 1 to 100]
> sumseries                [displays the result]

> sum(1:100)               [a quick way to sum the numbers 1 to 100]
```

Task 1. The taxi problem.

a) Suppose the mayor claims that there are at least 1000 taxis in NYC and based on a random sample of 5, we observe taxis with serial numbers 405, 280, 73, 440, 179. Based on this data, do you believe the mayor's claim is correct? One approach to answering this question is to observe the frequency of random samples of size 5 with maximum value ≤ 440 . Try to answer this question via simulation.

```
> sample(1000,5)          [produces a sample of 5 numbers from 1 to 1000]
```

Repeat this command 10 times (use up arrow key to get previous command) and see how many times the maximum value ≤ 440 . Use Splus looping facility to repeat this 1000 times.

```
> k <- 0
> for (i in 1:1000) {x <- max(sample(1000,5)); if (x <= 440) k<- k+1}
> k
```

What is your answer for the probability? Repeat using 10000 replications and compute the probability.

In this example, you can compute the probability exactly, $\binom{440}{5} / \binom{1000}{5} = .01628$.

Compare this answer with the ones you computed via simulation. Does the error seem too large?

b) Under the setup of (a), let's compare various estimates of T =total number of taxicabs. If you can't think of any estimates, use the following:

$$\begin{aligned}x_1 &= 6 * \max(y_1, \dots, y_5) / 5 - 1 \\x_2 &= \max(y_1, \dots, y_5) + \min(y_1, \dots, y_5) / 5 \\x_3 &= 2 * \text{median}(y_1, \dots, y_5)\end{aligned}$$

Here we will know the *true value* of T (1000) but we'll pretend as if it is unknown. Here's the Splus code for computing replicates of these statistics.

```

> y <- sample(1000,5)
> x1 <- 6* max(y)/5 -1
> x2 <- max(y)+min(y)/5
> x3 <- 2*median(y)
> for (i in 1:999) { y <- sample(1000,5);
  x1 <-c(x1, 6* max(y)/5 -1);
  x2 <- c(x2,max(y)+min(y)/5);
  x3 <- c(x3,2*median(y))}

```

At the end of the loop each of the x's will contain 1000 replicates of the respective estimates. Look at the mean of these and plot the histograms (> hist(x1)) to see which performed best (closest to 1000).

Task 2. Records.

The objective of this exercise is to see how many records one might expect from certain types of data. To get started, import 99 years (1895-1993) of temperature data into Splus by following the menu options File > Import Data > From File. After the Import Data dialog box opens, Look in folder G, subfolder st192, change the Files of type to Microsoft Excel Files [*.*xl*], and select the Excel file named tundra. A new data structure called tundra will be created. This data structure acts like matrix with the rows corresponding to years and the columns to months.

```

> tundra[,1] [displays average maximum temperature for Jan for the 99 years]
> tundra[,7] [displays average maximum temperature for July for the 99 years]
> tundra[2,] [displays the temperature for the 12 months in 1896]
> tundra[3,7] [displays the temperature for July of 1897]
> tsplot(tundra) [plots the temperature data for each of the 12 months]

```

For each month, count the number of record highs and record lows. Display the results in a table. You might think about an efficient way to "count" these records in Splus.

We will return to records next week.