AN INTRODUCTION TO STATISTICAL ANALYSIS AND USING R FOR COMPUTING AND GRAPHING

DATA ENTRY AND CREATION • GRAPHING • DESCRIPTIVE STATISTICS • INFERENTIAL STATISTICS

DATA ENTRY AND CREATION

There are many different programs you can use to enter, store, and manipulate your data. For the purposes of this course, we will be dealing with two: Microsoft Excel (hereafter "Excel") and R. We will be entering our data in Excel and analyzing it in R.

First, data entry in Excel. Most of you have probably entered data in Excel before. There are some quirks associated with it (e.g., date formatting), but most of it should be straight forward. Simply use row (horizontal) one to add the headers in each column (vertical) and enter your data below in each column. For example:

	A	B	C	D
1	HEADER1	HEADER2	HEADER3	HEADERN
2	DATA1,1	DATA1,2	DATA1,3	DATA1,N
3	DATA2,1	DATA2,2	DATA2,3	DATA2,N
4	DATAN,1	DATAN,2	DATAN,3	DATAN,N
5				

Next, we will import our data into a computing and graphing environment, R (http://www.r-project.org/index.html). (Note: any Google search of "R + [statistical question] will often yield good results. R is the most widely used program in the biological sciences and there is a great online support community.) There are two ways to do this: (1) import the data straight from the file and (2) paste your data directly into R.

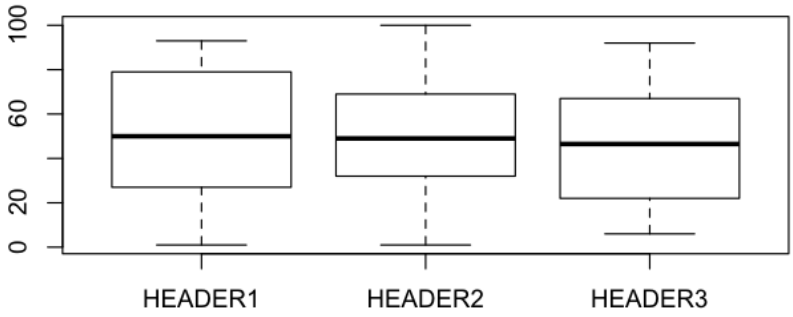
(1) Importing your data into R is straight forward. First, save your Excel spreadsheet as a .csv (stands for comma-separated value). Next, in R, load your data. Do this with the command: **read.csv("[file location]")**. Your data will pop up. If you want to do more than just SEE your data (manipulate, plot, etc.), save it as an *object* by specifying the name you want to save it as

(we will use "sampdat"), and assign it using "<-" (less than-hyphen). For example, **sampdat <-read.csv("/Users/christophermoore/Desktop/SampDat.csv")**. Now, you can just enter **sampdat** in the command line to recall your data. Note that is you want to call an individual column, attach the object using **attach([object])**.

(2) You can also just copy and paste your data in from Excel by creating a *vector*. This is done by assigning values using the **c([enter numbers separated by a comma])** function. Meaning, if you want to import one column from excel, just make a column of commas next to it (except for the last value), copy, and paste. You COULD also just enter your data in this way too.

GRAPHING

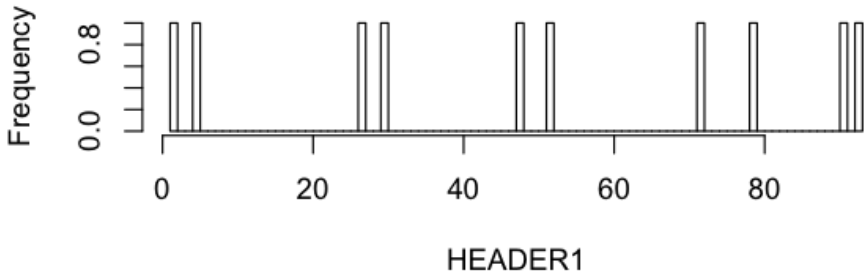
The first graphic we will learn to create is the box-and-whisker plot (commonly called "boxplot". These plots are very informative, as it shows meaningful statistical points on your data on the distribution. Enter **boxplot([object])**, and a graphic will appear like the following:



The dark bar in the center of the box is the *median* value of each variable. The ends of the boxes correspond to the 25 and 75 percentiles of your data, also known as the Interquartile range (IQR). The ends of the whiskers are ± 1.5 × IQR. If you have any outliers, they will appear

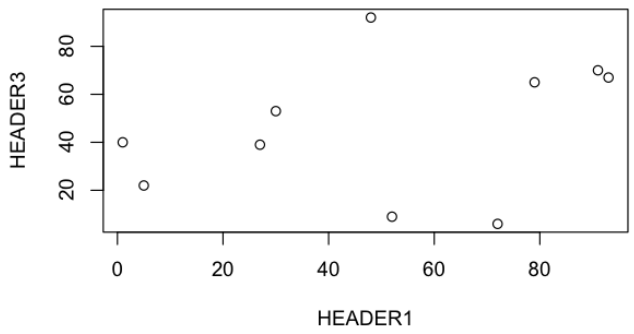
discontinuously beyond the whisker. You can also plot each column individually. This is likely the type of plot that you will use when presenting data from ANOVAs.

Another way to get a feel for your data is through a frequency histogram. Only a single variable can be plotted at once, so call one of your column headers or *vectors*. This is plotted as **hist([vector or column header])**. If you wish to change the bin size, just add breaks to your command. For example **hist([vector or column header] , breaks = [number])**. The histogram will look something like this:

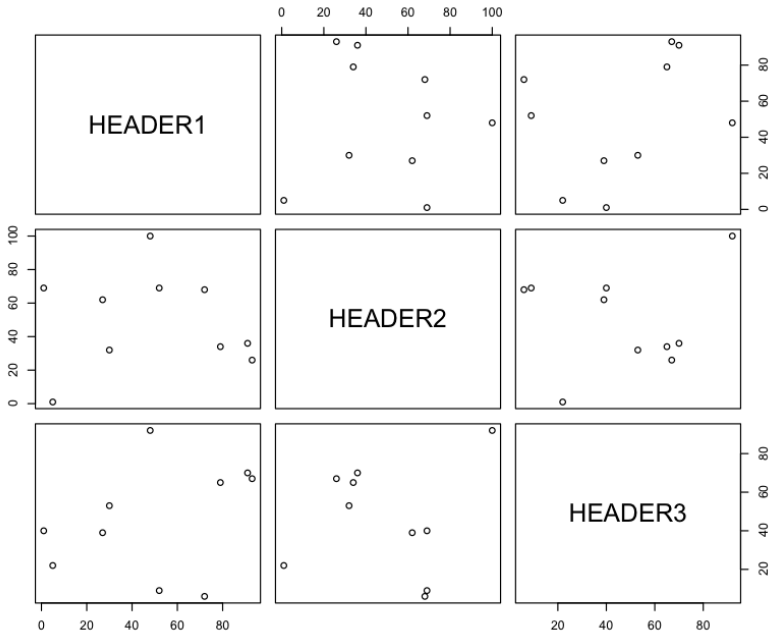


(I randomly chose the 10 data points here from 1 to 100, so there is no apparent pattern, like a Gaussian [bell-shaped] curve.) You should look at all of your data using histograms because this is the rawest way to analyze it graphically.

Lastly, if you wish to plot your continuous data, use **plot([x-axis] , [y-axis])**. This will give you a scatterplot of your data that looks like this:



Later in this handout we will show how to plot your fitted regression line. Another fruitful way to explore your data is to use the **plot()** function on your dataframe (in our case, **sampdat**). You will be presented with a matrix of all of your variables as scatterplots to visualize potential patterns:



(There are no patters here except that there are no patterns, as these data are random.)

DESCRIPTIVE STATISTICS

These are also known as your summary statistics, as they are metrics that simply describe and summarize your data. Here is a list of the basic calculations we will use:

mean([object]) calculates the mean

var([object]) calculates the variance

sd([object]) calculates the standard deviation

summary([object]) calculates min and max (the difference is the range), first and third quartiles (difference is IQR from above), and the median.

For the purposes of this class, these are the basics descriptive statistics that you will need to know for now (you might was to use others depending on what your data look like).

INFERENTIAL STATISTICS

Inferential statistics are metrics that come to conclusions about populations from having incomplete information through sampling and accounting for randomness. We will be making inferences about populations and comparing them to other populations (between treatments) or about correlations between variables in this class. Last week we discussed four ways to statistically analyze your data, depending on the continuity of the data: regression, ANOVA, logistic regression, and tabulation. Again, we will not be covering logistic regression in this course.

Regression can be performed by using the function **lm([formula])**. The "lm" stands for "linear model" which is an assumption we are making of our data and is easiest for this class (remember we were only analyzing simple linear regression?). In the formula, put the two variables of question in the parentheses with a tilda ("~") between them. For example, my example entry looked like this: **regx <- lm(HEADER1 ~ HEADER2)**. The output will give you your intercept and slope (i.e., β_0 and $\beta_1 X$, respectively). If you use the command **summary([object])** you will be presented with relevant statistics, including—and most importantly—coefficient of determination (i.e., R^2; the proportion of variability in a data set that is accounted for by the statistical model) and the probability of attaining the observed data if the null hypothesis is true (i.e., p-value). To plot this line with the data, plot it like normal (see

plot() above) and then type **abline([the name of the object, "regx" in this case])**, and you will see your best fit regression line over the scatterplot of your data points.

Next, ANOVA. Statistically, ANOVA and regression are the same thing (beyond the scope of this class), so we will start by using the same procedure in R. Repeat by creating an object and running a linear model, just like for regression, for example: **anovax <- lm(HEADER1 ~ HEADER3)**. HEADER3, in this instance, should contain the categorical predictor variables of interest. When summarizing (e.g., **anvova(anovax)**), you will notice an *F*-statistic, degrees of freedom values, and a *p*-value. Those are your important statistics to be reported in your final paper. Boxplots are an adequate way to graphically represent your data.