**R-Reading a CSV file:**

**What is CSV file:** **CSV Files**. A **CSV** is a comma separated values **file**, which allows data to be saved in a table structured format. CSVs look like a garden-variety spreadsheet but with a .**csv** extension (Traditionally they take the form of a text **file** containing information separated by commas, hence the name).

We also assume that the very first row contains a list of labels. The idea is that the labels in the top row are used to refer to the different columns of values.

Three columns of data and six rows. The three columns are labeled “trial,” “mass,” and “velocity.” We can pretend that each row comes from an observation during one of two trials labeled “A” and “B.” A copy of the data file is shown below and is created in defiance of Werner Heisenberg:

The first argument is the name of file. The second argument indicates whether or not the first row is a set of labels. The third argument indicates that there is a comma between each number of each line.

**> x <- read.csv(file = "experiment.csv",head = TRUE,sep = ",")**

**> x**

trial mass velocity

1 A 10.0 12

2 A 11.0 14

3 B 5.0 8

4 B 6.0 10

5 A 10.5 13

6 B 7.0 11

>

**> summary(x)**

trial mass velocity

A:3 Min. : 5.00 Min. : 8.00

B:3 1st Qu.: 6.25 1st Qu.:10.25

Median : 8.50 Median :11.50

Mean : 8.25 Mean :11.33

3rd Qu.:10.38 3rd Qu.:12.75

Max. :11.00 Max. :14.00

**> help(read.csv)**

**Display directory details using:**

**> dir();**

**Get working directory using:**

**>getwd();**

The variable “x” contains the three columns of data. Each column is assigned a name based on the header (the first line in the file). You can now access each individual column using a “$” to separate the two names:

**> names(x)**

**[1] "trial" "mass" "velocity"**

**> x$trail**

**NULL**

**> x$trial**

**[1] A A B B A B**

**Levels: A B**

**> x$mass**

**[1] 10.0 11.0 5.0 6.0 10.5 7.0**

**> x$velocity**

**[1] 12 14 8 10 13 11**

**>**

If the original data is given in an excel spreadsheet. It has been converted into a csv file, this is an option to save within excel. (You should save the file on your computer.) It is a good idea to open this file in a spreadsheet and look at it. This will help you make sense of how R stores the data.

The data is used to indicate an estimate of biomass of ponderosa pine in a study performed by Dale W. Johnson, J. Timothy Ball, and[ROGER[http://cdncache-a.akamaihd.net/items/it/img/arrow-10x10.png](http://www.cyclismo.org/tutorial/R/input.html#60562296)](http://www.cyclismo.org/tutorial/R/input.html#60562296) F. Walker who are associated with the Biological Sciences Center, Desert Research Institute, P.O. Box 60220, Reno, NV 89506 and the Environmental and Resource Sciences College of Agriculture, University of Nevada, Reno, NV 89512. The data is consists of 54 lines, and each line represents an observation. Each observation includes measurements and markers for 28 different measurements of a given tree.

For example, the first number in each row is a number, either 1, 2, 3, or 4, which signifies a different level of exposure to carbon dioxide. The sixth number in every row is an estimate of the biomass of the stems of a tree. Note that the very first line in the file is a list of labels used for the different columns of data.

The data can be read into a variable called “tree” in using the read.csv command:

**tree <- read.csv(file="trees91.csv",header=TRUE,sep=",");**

This will create a new variable called “tree.” If you type in “tree” at the prompt and hit enter, all of the numbers stored in the variable will be printed out. Try this, and you should see that it is difficult to make any sense out of the numbers.

There are many different ways to keep track of data in R. When you use the ***read.csv*** command R uses a specific kind of variable called a “**data frame**”. All of the data are stored within the data frame as separate columns. If you are not sure what kind of variable you have then you can use the **attributes** command. This will list all of the things that R uses to describe the variable:

>**attributes(tree)**

$names

[1] "C" "N" "CHBR" "REP" "LFBM" "STBM" "RTBM" "LFNCC" "STNCC" "RTNCC" "LFBCC" "STBCC" "RTBCC" "LFCACC" "STCACC" "RTCACC" "LFKCC" "STKCC" "RTKCC" "LFMGCC"

[21] "STMGCC" "RTMGCC" "LFPCC" "STPCC" "RTPCC" "LFSCC" "STSCC" "RTSCC"

$class

[1] "data.frame"

$row.names

[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54

The first thing that R stores is a list of names which refer to each column of the data. For example, the first column is called “C”, the second column is called “N.” Tree is of type data.frame. Finally, the rows are numbered consecutively from 1 to 54. Each column has 54 numbers in it.

If you know that a variable is a data frame but are not sure what labels are used to refer to the different columns you can use the names command:

**> names(tree)**

[1] "C" "N" "CHBR" "REP" "LFBM" "STBM" "RTBM" "LFNCC" "STNCC" "RTNCC" "LFBCC" "STBCC" "RTBCC" "LFCACC" "STCACC" "RTCACC" "LFKCC" "STKCC" "RTKCC" "LFMGCC"

[21] "STMGCC" "RTMGCC" "LFPCC" "STPCC" "RTPCC" "LFSCC" "STSCC" "RTSCC"

>If you want to work with the data in one of the columns you give the name of the data frame, a “$” sign, and the label assigned to the column. For example, the first column in tree can be called using “tree$C:”

**> tree$C**

[1] 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4

>

**FIXED WIDTH FILES:**

There are many ways to read data using R. We only give two examples, direct assignment and reading csv files. However, another way deserves a brief mention. It is common to come across data that is organized in flat files and delimited at preset locations on each line. This is often called a “fixed width file.”

The command to deal with these kind of files is *read.fwf*. Examples of how to use this command are not explored here, but a brief example is given. If you would like more information on how to use this command enter the following command:

**> help(read.fwf)**

The *read.fwf* command requires at least two options. The first is the name of the file and the second is a list of numbers that gives the length of each column in the data file. A negative number in the list indicates that the column should be skipped.

In fixedWidth data file there are three columns. The first colum is 17 characters wide, the second column is 15 characters wide, and the last column is 7 characters wide. In the example below we use the optional *col.names* option to specify the names of the columns:

**> a = read.fwf('fixedWidth.txt',widths=c(-17,15,7),col.names=c('temp','offices'))**

**> a**

temp offices

1 17.0 35

2 18.0 117

3 17.5 19

4 17.5 28

>