# Loading Data into R

Most of the datasets we use are stored at http://www.stat.ucla.edu/~rgould/datasets, and you can find links to this site from the course web page www.stat.ucla.edu/~rgould/110as02.

**Standard Format**

We'll make some assumptions about the format of the files you upload into R. If your file doesn't look this way, you might have to use a word processor or spreadsheet program to edit it.

1. the file is ascii text, tab delimited. (Set this at the "save as" feature in Word or Appleworks.)
2. The first line consists of variable names, separated by tabs.
3. The next row contains the values of each variable for the first unit. Each row represents a separate unit that was measured.
4. If for some reason a value is missing, it should have a special character to denote that it is missing. It can't simply be left blank. R prefers this character to be "NA", but it doesn't really matter what you use. Popular choices are ".", "?", and "-99" (assuming -9999 is not a possible value for these variables.)
5. Each row must have the same number of entries.

Open the data file to check. Here's the risk data set:

| ID | appliances | nuclear | pool | plane | xray | gender |
|---|---|---|---|---|---|---|
| 625 | 0 | 100 | 50 | 50 | 0 | male |
| 526 | 0 | 100 | 10 | 0 | 100 | female |
| 684 | 0 | 95 | 10 | 5 | 80 | female |
| 50 | 0 | 50 | 20 | 40 | 75 | male |
| 535 | 0 | 100 | 50 | 0 | 80 | female |

So the first subject has ID# 625, and rates the risk level of appliances as 0, nuclear as 100, pool as 50, etc.

Here are the first few lines of fsalaray.txt, which contains data for one of the homework problems:

```
77	"Full"
79	"Full"
80	"Full"
85	"Full"
86	"Full"
```

This file has no "header"; no line with variable names. This is less than ideal, but we can live with it.

**Getting the Data onto your Harddrive**

I can't really help you here. It's depends on what browser you use, what type of machine you use, and what operating system you're using. In the lab, it works best if you use Explorer.

- Place your cursor over the link to the dataset you want, but don't click.
- Hold down the "ctrl" key and then press and hold the mouse. A menu should open up.

• Choose “download link to disk”

In the lab, this should put the file in your Documents folder. But at home it’s destination depends on the settings of your browser.

**Final Steps**
1) Start up R.
2) Change the Working Directory
On the menu, select “Tools: Change Working Directory” so that it is the same directory as the one that contains the data file.

OR... if this doesn’t work (sometimes R just crashes if you do this) move the data file into the same folder/directory that contains the application R.

3a) If the data file has a header, from within R type
*riskdata <- read.table(“risk.txt”, header=T)*

Type
*names(riskdata)*
to see the variables available.

3b) If the data file does not have a header, type
*fsalary <- read.table(“fsalary.txt”)*

Type *names(fsalary*). What do you see?

**To refer to the variables**

1. You can see all 600 + appliance entries by typing
*riskdata$appliance*

You can find the average risk rating by type
*mean(riskdata$appliances)*, or the average plane rating: *mean(riskdata$plane)*.

2. Or you can type
*attach(riskdata)*

You can now refer to the variables without the preceding “riskdata$...” For example:
mean(appliances)

**Renaming Variables**
The fsalary dataset has rather dull names. Unfortunately, it is not possible to rename objects. But we can create new objects that are copies of the old, but with the names we want. For

example
*salary <- fsalary$V1*
*rank <- fsalary$V2*

We can now work with and manipulate these variables.

**Fixing Missing Values**
R requires that missing values be coded with NA. If a dataset uses another symbol, you must convert it to "NA". The dataset fsalary has no missing values, but let's suppose that it did, and that they were coded with "."

To convert, type
*salary [salary=="."] <- NA*

What does this do? The statement 'salary == "." ' is TRUE whenever the value of salary is ".", and FALSE if it is anything else. For all of the entries of salary in which there is a TRUE value, we replace whatever was there with the character string NA.

In general, the "<-" is used to assign values on the right-hand side to the object on the left-hand side.

**Examining Variables**
You can index particular values of a variable. For example, the 325th value of appliances is

```
> appliances[325]
[1] 15
```

Or you can refer to a subset. For example, the 1st, 25th, and 503rd entries:

```
> appliances[c(1,25,503)]
[1]  0  0 40
```

Or a range of values:

```
> appliances[313:320]
[1] 13 15 15 15 15 15 15 15
```

But we can get fancier. Suppose we wanted to look only at the men
This command
*mappliance <- appliances[gender=="male"]*
will select only those values of appliances for which the corresponding value of gender is 1.

*Question: Create a variable named fright which contains the appliance ratings only for those who thought that the plane risk was greater than 15.*

**Dealing with Missing Values**
Type
*mean(appliances)*

You'll notice that you don't get the average rating. Why not? Because there are missing values. R cannot compute arithmetic operations on missing values, and so the result is itself a missing value. One quick fix behind this is to type
mean(appliances, na.rm = T)
which tells R to remove the missing values before doing the computation.

Answer to Question:
fright <- appliance[plane > 15]