# Introduction to R — Fall 2007
## Homework Set # 3

Begin the homework session by executing the following command in *R*:

```
source("http://www.nd.edu/~steve/Rcourse/hmwrkData/hmwrk3Data.R")
```

This will define the variable x1. Note that each student will have a different x1. Also read in the .CSV file of clinical data and store as a data.frame as follows.

```
clinUpps <- read.csv(file=
"http://www.nd.edu/~steve/Rcourse/hmwrkData/Clinical_Upps.csv")
```

**Learning Set 3.A.** Look up the help on `order`. Understand how to sort by multiple columns.

**Exer. 3.1.** Make the rownames of the data.frame `clinUpps` the patient IDs.

(1) Find the first quartile of the `age_at_diagnosis` variable.

(2) Create the data frame `early` of the data from original data frame restricted to patients with age ≤ the first quartile.

(3) Sort the `early` data frame by grade (`Grd`) and then by age.

(4) Save the resulting data frame as a .CSV file named "yourNameEarly.csv". Submit this with your homework.

**Exer. 3.2.** Create a Q-Q normal plot of x1. Add the Q-Q line in blue. Title the plot as "Your Name's Q-Q Normal Plot". Save this graphic with a title including your name and e-mail with your homework. Would you consider x1 to be normally distributed?

**Exer. 3.3.** Create a function that, given a numeric vector v returns the subvector of values between the first and third quartiles.

**Exer. 3.4.** Create a function that, given a character vector a and a character s, returns the number of entries in a that are equal to s. NOTE: the function should have one parameter for a and one for s.

**Exer. 3.5.** Create a data frame `dat` by reading in the .CSV file

```
http://www.nd.edu/~steve/Rcourse/hmwrkData/sampleDataFrame.csv
```

All values in this data.frame are either A, B, or C. Think of each row as a sample and each column as a reading of some sample trait at a different time. Create an integer vector `numB`, with names the rownames in `dat`, such that for each i, `numB[i]` is the number of B's in row i of `dat`. (You may use either a loop or `lapply`. If you use `lapply` be sure to use `unlist` to create an integer vector as the final result.)