Data management recommendation

- Type R code into a script rather than directly into the R command prompt. Annotate the script with your own notes (R will ignore anything in a line after a # sign). Save the R script in the same folder as the data
- Store data in a nonproprietary software format (e.g., .txt. or .csv). You can still use spreadsheets (e.g., Open Office or Excel). Converting to .csv file generally best way.
- Keep a copy of an uncorrected spreadsheet
- Use descriptive names for your data files. Use _ for spaces in file names. E.g., Melongena_size_survey_2014.csv
- Use short, easy to type names for column headers
- Enter data into spreadsheets using the *long* format, not the *wide* format

Procedure for getting data into R

- Check the column headers in the Excel file (change to something that's easy to type, with no symbols or formatting)
- Save the file as a .csv
- Open R, File > New Script
- Type all commands into the Editor (not the R console)
- Then type into the R Editor: read.csv("insert file path here/filename.csv") Obviously, change the text 'insert file path here' to the actual file path, and "filename.csv" to the actual file name!
- Send the code to R using Cntr R for PC's, Command Return for Mac, or simply cut and paste (for both computer types)
- Check that the data imported correctly in R by typing the filename into R, and also typing summary("filename"), where "filename" is whatever you called it in R.

Tips for plotting in R

- See `?par` for ways to adjust the components of the figure within the plot command
- Set the plot size first, using `dev.new(width = #, height = #)`. This is needed when making plots for a paper and assignments.
- For multiple plots on the same page, use `par(mfrow=c(#,#))` or `par(mfcol=c(#,#))`. Again, see `?par` for what these mean
- Within the plot command, specify `las=1` to make y-axis tick labels horizontal