STAT 517: Takehome 1

1. (30 pts) Load the **mtcars** data. Use the data to make a plot as similar to the one on the web page as possible. Provide the commands you used as well as the finished plot. Include your name in the lower right hand corner. We have already used most of the commands you will need–`help(legend` should help with the rest.

2. (30 pts) Consider the hit-or-miss simulation we studied in class.

   (a) Convert the commands from the hit-or-miss simulation into a function; the function arguments should include the number of random draws, the upper and lower endpoints of x, the function maximum over the domain of x, and a user-defined function (Graduate students should not provide the function maximum, but compute it within the body of the function). The function maximum should simply be a numeric value you determined beforehand. The user-defined function needs to be created beforehand–just like we created `ex1.fcn` in class. The output should include the estimated integral and the proportion of points under your curve.

   (b) Test your program by estimating:

   $$h(x) = \int_0^2 \exp(-x)\, dx$$

   Use 10000 draws and compare your answer to the actual value of this integral. Remember that this is a simulation and there is no single right answer here.

   (c) Grad students (or EC for undergrads) Add the necessary commands to generate a graphic of the simulation similar to the one I generated in class.

3. (10 pts) Heights and weights for 10 males and 15 females appear below:

```
MaleHt=c(68,72,74,66,70,70,71,72,69,70)
FemaleHt=c(66,64,64,68,63,65,66,65,67,70,69,64,66,66,62)
MaleWt=c(171,191,193,168,181,181,183,178,180,178)
FemaleWt=c(135,130,131,139,127,136,132,138,146,144,127,135,136,123,130)
```

   (a) Generate side-by-side boxplots of Male Height/Female Height and a side-by-side boxplots of Male Weight/Female Weight.

4. (30 pts) Here is code for Classic Monte Carlo Integration (using 10000 iterations) of the function we studied in class:

```
ex1.fcn <- function(x){
h <- 4/(1 + x^2)
return(h)
}
n = 10000
```

```
a = 0
b = 1
my.rand.x = runif(n, min=a, max=b)
pi.MC = ((b - a)/n)*sum(ex1.fcn(my.rand.x))
```

I want you to create a **for** loop that saves the estimate of π from each of 100 separate iterations and stores it in a numeric vector (of length 100). Construct a histogram of your estimates and include a red vertical reference line at π. Comment on the accuracy of the Classic Monte Carlo integration approach. Graduate students should place the **for** loop in a function that allows the user to vary the sample size, the simulation size, the integration limits and the input function. Graduate students should demonstrate their function for at least one additional input function.