```r
# load in all of the packages needed for this hw
library(boot)
library(bootstrap)
library(lattice)

# read the data in
samp <- subset(read.csv("http://stat.pugetsound.edu/hoard/datasets/mms.csv"), type="plain")$diameter

# 1.s.
# generate 1000 bootstrap samples of Plain M&M diameter, compute the mean and median
# of each, generate density plots of both, and compute the standard error of each
# number of bootstrap samples
N <- 1000

# set up empty vectors to store the sample means and sample medians
sampleMeanStar <- numeric(N)
sampleMedianStar <- numeric(N)

# for pedagogical purposes, set the seed
set.seed(100)

# generate N bootstrap samples and compute and store the test statistic(s) for each
for (i in 1:N)
{
  # generate a bootstrap sample
  sampStar <- sample(samp, replace=TRUE)

  # compute and store its sample mean
  sampleMeanStar[i] <- mean(sampStar)
sampleMedianStar[i] <- median(sampStar)
}

# generate a density plot of each test statistic
densityplot(~sampleMeanStar)
densityplot(~sampleMedianStar)

# compute the standard error of each test statistic
sd(sampleMeanStar)
sd(sampleMedianStar)

# 2.s.

# number of bootstrap samples
N <- 1000

# confidence level for this problem
cl <- .90

### percentile
# compute a percentile bootstrap confidence interval using the bootstrap samples from 1.s.
quantile(sampleMedianStar, probs=c((1-cl)/2, (1+cl)/2))

### BCa
# for the BCa bootstrap with boot(), we need to define a median function with other arguments
bMedian <- function(y, id)
{
  median(y[id])
}

# set the seed for pedagogical purposes
set.seed(100)

# compute a BCa bootstrap confidence interval
boot.ci(boot(samp, bMedian, N), conf=cl)
```
### bootstrap-t

# set the seed for pedagogical purposes
set.seed(100)

# use boot to compute a confidence interval for the mean
# use VS (variance stabilization)
# generate 50 bootstrap samples for the variance stabilizing function g
# generate 50 bootstrap samples for estimating the standard deviation
# generate N bootstrap samples and compute the sample mean of each
# select the first piece of the output, [1], to get a confidence interval
boott(samp, mean, VS=TRUE, v.nbootg=50, v.nbootsd=50, v.nboott=N, perc=c((1-cl)/2,(1+cl)/2))[1]

# 3.s.
# confidence level for this problem
cl <- .80

# set the seed for pedagogical purposes
set.seed(100)

### BCa bootstrap
# to use the boot() function, we need a test statistic function
bMean <- function(y, id)
{
  mean(y[id])
}

# now actually compute a BCa bootstrap confidence interval
boot.ci(boot(samp, bMean, N), conf=cl)

### compute a confidence interval using the normal approximation
mean(samp) + c(-1,1)*qt((1+cl)/2, df=length(samp)-1)*sd(samp)/sqrt(length(samp))