STAT 2450 Assignment 5 (20 pt)

Alice Liu

Banner: B00783546

1. Plot a histogram of the bootstrap distribution of the estimated slope parameter for the following data.

```
rm(list=ls())
set.seed(999123)
x = c(1,1.5,2,3,4,4.5,5,5.5,6,6.5,7,8,9,10,11,12,13,14,15)
y = c(21.9,27.8,42.8,48.0,52.5,52.0,53.1,46.1,42.0,39.9,38.1,34.0,33.8,30.0,26.1,24.0,20.0,11.1,6.3)
```

- a convenient way to incorporate a bootstrap sample from the pairs (x_i, y_i), is as follows. lm(y~x,subset=sample(1:n,n,replace=T)
- then extract the estimated slope using coef(lm.out)[2]

```
Nboot=1000
temp=NULL
n=length(x)
data=cbind(x,y)

for (i in 1:Nboot){
 ind=sample(1:19,19,replace=T)
   mydata=data.frame(data[ind,])
   lm_out=lm(y~x,data=mydata) #carry out the regression on a bootstrap sample
   estslope=coef(lm_out)[2]#   extract the slope
   temp=c(temp,estslope) #   accumulate the slopes in temp
}
 hist(temp,main = "Histogram of the bootstrap distribution of the estimated Slope",
          xlab = "Estimated slope")
```

- (a) make a histogram of the estimated slopes (7 points)
- (b) use the quantile function to calculate a 90% percentile interval (2 points)

```
interval=quantile(temp,c(0.05, 0.95))
interval
```

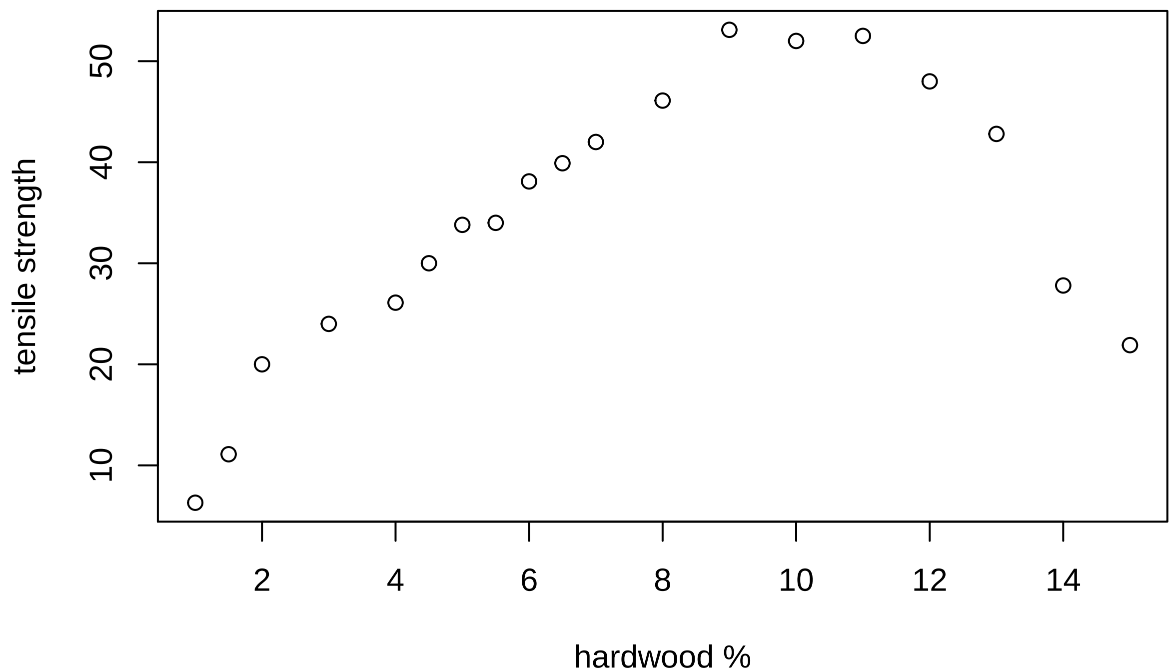
- (c) estimate the variance of $\hat{\beta}_1$. (1 point)

```
varbeta_hat=var(temp)
varbeta_hat
```

2. Explore the sensitivity of the bootstrap estimate of the standard error of the slope estimate using the "boot" procedure, using the hardwood data.

The hardwood data consists of of pairs $(x_i, y_i), i = 1, 2, \ldots, n$, where x is the proportion of hardwood in a piece of lumber, and y is a measure of tensile strength. A plot of x vs y indicates that the relationship is non-linear.

```
rm(list=ls())
x = c(1,1.5,2,3,4,4.5,5,5.5,6,6.5,7,8,9,10,11,12,13,14,15)
y = c(6.3,11.1,20,24,26.1,30,33.8,34.0,38.1,39.9,42,46.1,53.1,52,52.5,48,42.8,27.8,21.9)
n=length(y)
plot(x,y,xlab="hardwood %",ylab="tensile strength")
```



(8 points)

- Run the boot procedure 10 times with a bootstrap sample size of 10.
- Run the boot procedure 10 times with a bootstrap sample size of 100.
- Run the boot procedure 10 times with a bootstrap sample size of 1000.

(2 points)

```
How does the variability of the bootstrap estimate of standard error
change as the number of bootstrap samples increases?
```

ANSWER: According to my data, the variability of bootstrap estimate of standard error decreases as the number of bootstrap samples increases.

Notes:

- You can use the var function to calculate the variance of an array
- The p-% percentile interval is usually centered, i.e. it excludes left and right tails of equal probabilities.
- For the last question you can if you wish show 3 box plots side by side, i.e. one boxplot per sample-size.
- Please note that in order to calculate your bootstrap distributions via a call to the 'boot' function, you will need to use the 't' attribute of the object returned by 'boot'. This field contains the value of the statistics that you have decided to return in your extracting function. See the following code for an example.

```
#boot:used to estimate the std error of any statistic
library(boot)
data=data.frame(x=x,y=y)

bootdata1=function(data,index){
x=data$x[index]
y=data$y[index]
lmout=lm(y~x)
bootdata1=coef(lmout)[2]
return(bootdata1)
}

temp1=NULL
for(i in 1:10){
   b_out=boot(data,bootdata1,R=10)
   temp1=c(temp1,sd(b_out$t))
}
var(temp1)
```

```
## [1] 0.06877412
```

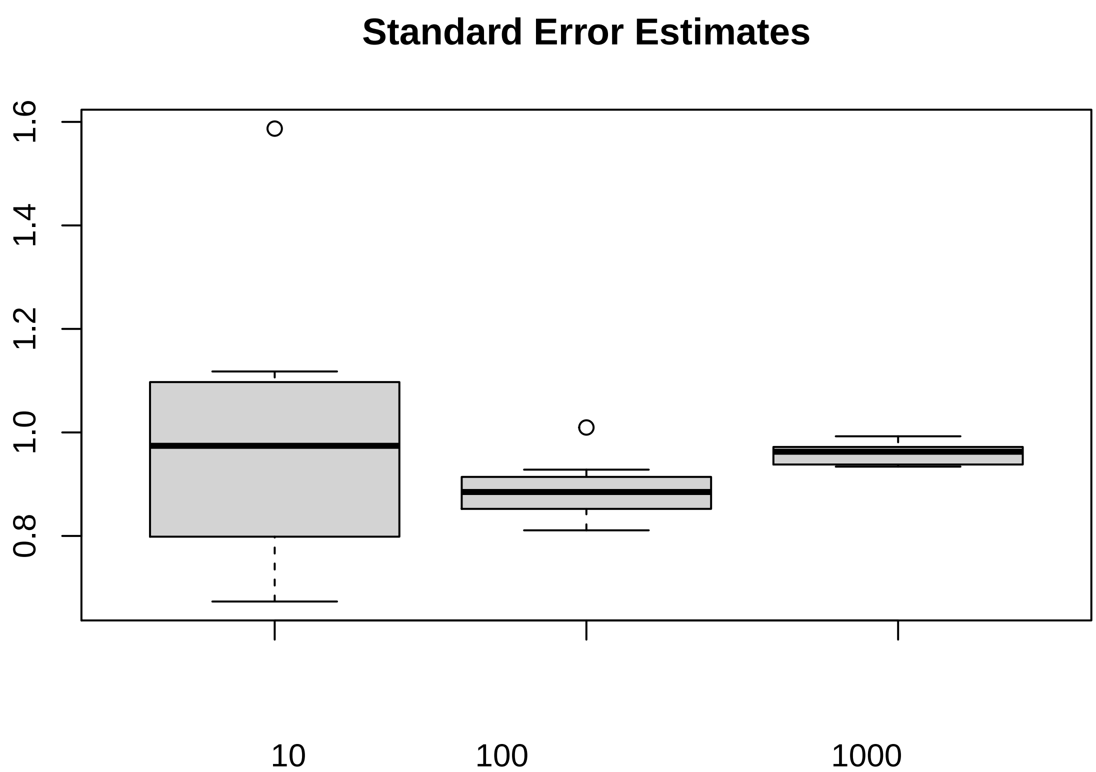
```
temp2=NULL
for(i in 1:10){
   b_out=boot(data,bootdata1,R=100)
   temp2=c(temp2,sd(b_out$t))
}
var(temp2)
```

```
## [1] 0.002914719
```

```
temp3=NULL
for(i in 1:10){
   b_out=boot(data,bootdata1,R=1000)
   temp3=c(temp3,sd(b_out$t))
}
var(temp3)
```

```
## [1] 0.0003814681
```

```r
boxplot(temp1,temp2,temp3,main="Standard Error Estimates",xlab="10                    100
```




```r
# define the extractor function:
#myext <- function(data,index){
# choose what you want to return, for example an estimated regression parameter
# e.g. this might be the estimate of the slope of a linear regression model
# return (this)
#}
# call boot
# bout <- boot(data,myext,..)
# retrieve the bootstrap distribution:
# bout$t
# now you can work on this bootstrap distribution and apply var or sd to e.g. find
# the variance or standard error of the bootstrap distribution
# additional information and examples may be found in the last class lecture
```