Computer Age Statistical Inference (CASI) chapter 2

honground

10/19/2018

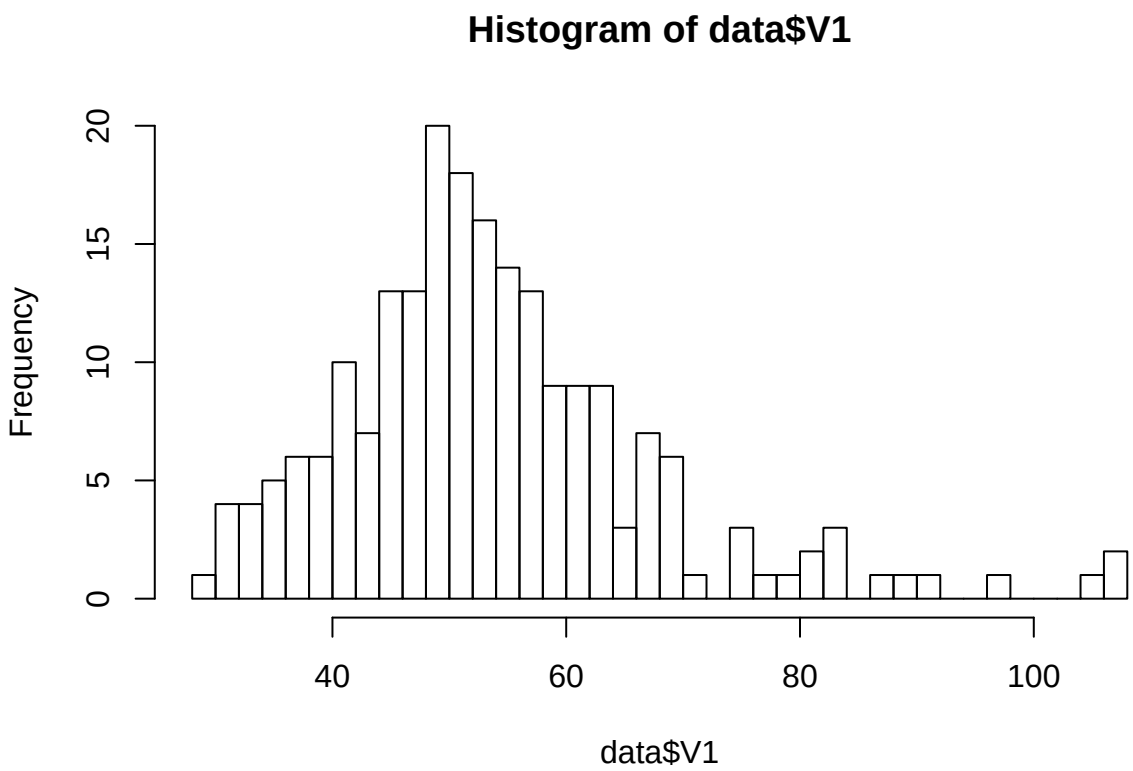
R Markdown

This is an R Markdown document for an implementation of chapter 2 of CASI (computer age statistical inference) R code chunks within the document. You can embed an R code chunk like this:

```
###computer age statistical inference chapter 2
#by honground - suckwon Hong

#install and load required package
install.packages("psych")
```

```
## Installing package into '/home/rstudio-user/R/x86_64-pc-linux-gnu-library/3.5'
## (as 'lib' is unspecified)
```

```
library(psych)

#load data
data<-read.table("gfr.txt",header=F)

#Figure 2.1
hist(data$V1,breaks=30)
```




```
#mean and standard error (Table 2.1)
mean<-mean(data$V1)
```

```r
s.error<-function(x){
  n<-length(x)
  return (sqrt(sum((x-mean(x))^2)/(n*(n-1))))
}
paste("mean: ", mean)
```

```
## [1] "mean:  54.2654028436019"
```

```r
paste("standard error: ",s.error(data$V1))
```

```
## [1] "standard error:  0.944584362171675"
```

```r
#winsorized mean
mean.winsorized<-winsor.mean(data$V1, trim = 0.25, na.rm = TRUE)
paste("winsorized mean: ", mean.winsorized)
```

```
## [1] "winsorized mean:  52.8104265402844"
```

```r
#bootstrap mean
mean.bootstrap<-c()
for(i in 1:1000){
  data.boot<-data$V1[sample(1:length(data$V1),replace=T)]
  mean.bootstrap<-c(mean.bootstrap, winsor.mean(data.boot, trim = 0.25, na.rm = FALSE))
}
s.error.bootmean<-sd(mean.bootstrap)

median.bootstrap<-c()
for(i in 1:1000){
  data.median<-data$V1[sample(1:length(data$V1),replace=T)]
  median.bootstrap<-c(median.bootstrap,median(data.median))
}
s.error.bootstrap<-sd(median.bootstrap)
paste("bootstrap mean_standard deviation: ",s.error.bootmean)
```

```
## [1] "bootstrap mean_standard deviation:  0.898673682734617"
```

```r
paste("bootstrap median_standard deviation: ",s.error.bootstrap)
```

```
## [1] "bootstrap median_standard deviation:  0.842475973405631"
```

```r
#Figure 2.2
n<-10
result<-c()
for(c in seq(-6,6,0.2)){
  alpha<-c()
  beta<-c()
  for(i in 1:1000){
    x<-rnorm(1,0,1/sqrt(n))
    f1x<-dnorm(x,0.5,1/sqrt(n))
    f0x<-dnorm(x,0,1/sqrt(n))
    Lx<-f1x/f0x
    Tx<-log(Lx)
    a<-isTRUE(Tx>=c)
    alpha<-c(alpha,a)

    x<-rnorm(1,0.5,1/sqrt(n))
    f1x<-dnorm(x,0.5,1/sqrt(n))
```

```
    f0x<-dnorm(x,0,1/sqrt(n))
    Lx<-f1x/f0x
    Tx<-log(Lx)
    b<-isTRUE(Tx<c)
    beta<-c(beta,b)
  }
  result<-rbind(result,data.frame(alpha=mean(alpha),beta=mean(beta),c))
}
plot(result$alpha,result$beta,type="l")
```