Storvik Extra 9

a

```
#x=scan("http://www.uio.no/studier/emner/matnat/math/STK2120/v13/illrain.dat",
#       na.strings="*")

#Remove missing data
x <- x[!is.na(x)];n <- length(x)

#Moment estimates
lambda.hat <- mean(x)/var(x);alpha.hat <- mean(x)^2/var(x)
B <- 1000
lambda.star <- rep(NA,B);alpha.star <- rep(NA,B)
for(b in 1:B)
{
#Draw from Gamma distribution
    x.star <- rgamma(n,shape=alpha.hat,rate=lambda.hat)
#Calculate estimates based on bootstrap samples
    lambda.star[b] <- mean(x.star)/var(x.star)
    alpha.star[b] <- mean(x.star)^2/var(x.star)
}


alpha <- 0.05
#Standard bootstrap konfidensintervall
k1 <- as.integer(B*alpha/2);k2 <- as.integer(B*(1-alpha/2))
lambda.sort <- sort(lambda.star)
deltaL <- lambda.sort[k1]-lambda.hat
deltaU <- lambda.sort[k2]-lambda.hat
ki.lambda.basic <- c(lambda.hat-deltaU,lambda.hat-deltaL)
alpha.sort <- sort(alpha.star)
deltaL <- alpha.sort[k1]-alpha.hat
deltaU <- alpha.sort[k2]-alpha.hat
ki.alpha.basic <- c(alpha.hat-deltaU,alpha.hat-deltaL)

print(alpha.hat)
print(ki.alpha.basic)
print(lambda.hat)
print(ki.lambda.basic)
```

b

```
#x=scan("http://www.uio.no/studier/emner/matnat/math/STK2120/v13/illrain.dat",
#       na.strings="*")

#Remove missing data
x <- x[!is.na(x)];n <- length(x)

#Moment estimates
B <- 1000
C.hat = sd(x)/mean(x)
C.star <- rep(NA,B)
for(b in 1:B)
{
```

```
#Draw from Gamma distribution
    x.star <- sample(x,size=n,replace=T)
#Calculate estimates based on bootstrap samples
   C.star[b] <- sd(x.star)/mean(x.star)
}


alpha <- 0.05
#Standard bootstrap konfidensintervall
k1 <- as.integer(B*alpha/2);k2 <- as.integer(B*(1-alpha/2))
C.sort <- sort(C.star)
deltaL <- C.sort[k1]-C.hat
deltaU <- C.sort[k2]-C.hat
ki.C.basic <- c(C.hat-deltaU,C.hat-deltaL)
print(C.hat)
print(ki.C.basic)
```

Storvik Extra 10

a,b

```
#exe12.74=read.table(
#     "http://www.uio.no/studier/emner/matnat/math/STK2120/v16/exe12-74.txt",
#     header=T)
attach(exe12.74)
model = lm(VO2~HR, data=exe12.74)
print(summary(model))

B = 10000
n = length(VO2)
beta0.star = rep(NA, B)
beta1.star = rep(NA, B)
for (b in 1:B)
{
    x.star = HR
    e.star = sample(model$residuals, n, replace=T)
    y.star = model$fitted.values + e.star
    model.star = lm(y.star~x.star)
    beta0.star[b] = model.star$coefficients[1]
    beta1.star[b] = model.star$coefficients[2]
}
print(mean(beta0.star))
print(sd(beta0.star))
print(mean(beta1.star))
print(sd(beta1.star))
```

c

```
#exe12.74=read.table("http://www.uio.no/studier/emner/matnat/math/STK2120/v16/exe12-74.txt
#header=T)
model = lm(VO2~HR, data=exe12.74)
print(summary(model))
x = exe12.74$HR
y = exe12.74$VO2
B = 10000
```

```r
n = length(HR)
beta0.star = rep(NA, B)
beta1.star = rep(NA, B)
for (b in 1:B)
{
    i.star = sample(1:n, n, replace=T)
    x.star = x[i.star]
    y.star = y[i.star]
    model.star = lm(y.star~x.star)
    beta0.star[b] = model.star$coefficients[1]
    beta1.star[b] = model.star$coefficients[2]
}
print(mean(beta0.star))
print(sd(beta0.star))
print(mean(beta1.star))
print(sd(beta1.star))
```

d

b) is more natural when the x_i are considered fixed. c) is more natural when x_i is considered random

e

c) is more robust since the estimated distributions will be more like the true distributions.

Exam 2006, 2
Exam 2009, 2