MATH48091 Statistical Computing Coursework 3

07 December, 2019

Question 1.

(i) For this part, the clinician has proposed the following model for the response variable, new-born baby's length in inches, denoted as y:

$$y_i = \alpha + \beta m_i + \gamma f_i + \epsilon_i, \quad i = 1, \ldots, 42$$

If we fit this model in R using the `lm()` function, we can then find estimates for the coefficients α, β and γ. Hence, once the data from "birth length.txt" has been read into the command window, we can specify variables accordingly and create the fitted model using the proposed model from above.

```
data <- read.table("birth length.txt", header=TRUE, sep="")

y <- data$length
m <- data$mheight
f <- data$fheight
s <- data$smoker

model <- lm(y ~ m + f, data)

estimates <- model$coefficients
names(estimates) <- c("alpha", "beta", "gamma")
estimates
```

```
##      alpha       beta      gamma
## 6.65996234 0.17164827 0.03128298
```

Therefore,

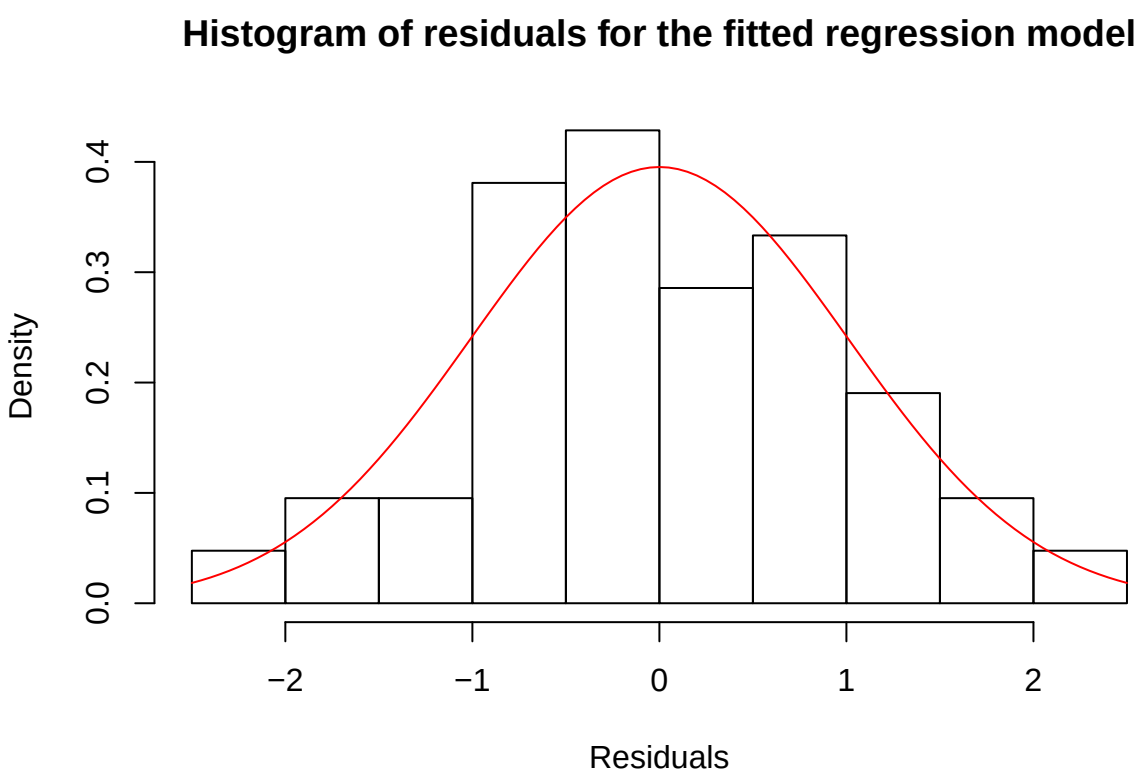
$$\hat{y}_i = 6.65996234 + 0.17164827 m_i + 0.03128298 f_i$$

is the fitted model for our data.

(ii) We can work out the residuals of the fitted model above by computing the difference between estimated values of the baby's length and the real value of the baby's length for given values of mother's and father's height. Hence $r_i = y_i - \hat{y}_i$ for $i = 1, \ldots, 42$ where r_i is the residual.

Then a histogram is plotted of the residuals we have found.

```
yest <- estimates[1] + estimates[2]*m + estimates[3]*f
res <- y - yest
hist(res, freq=FALSE, main="Histogram of residuals for the fitted regression model",
     xlab="Residuals")
curve(dnorm(x, mean=mean(res), sd=sd(res)), col=2, add=T)
```



The pdf of a Normal distribution with the mean set to be the mean of the residuals and the variance equal to the variance of the residuals has been superimposed onto the histogram. There seems to be little goodness of fit to help suggest that the residuals are normally distributed. However, since the data set is relatively small, we cannot rule out that residuals are not normally distributed.

(iii) The function below implements the bootstrap residual method to generate $B = 1000$ bootstrap estimates of the coefficients of the linear model that the clinician proposed earlier. We would like to only look at the coefficient of the father's height for this part, and hence we just call the third column of the bootstrap sample in the code given.

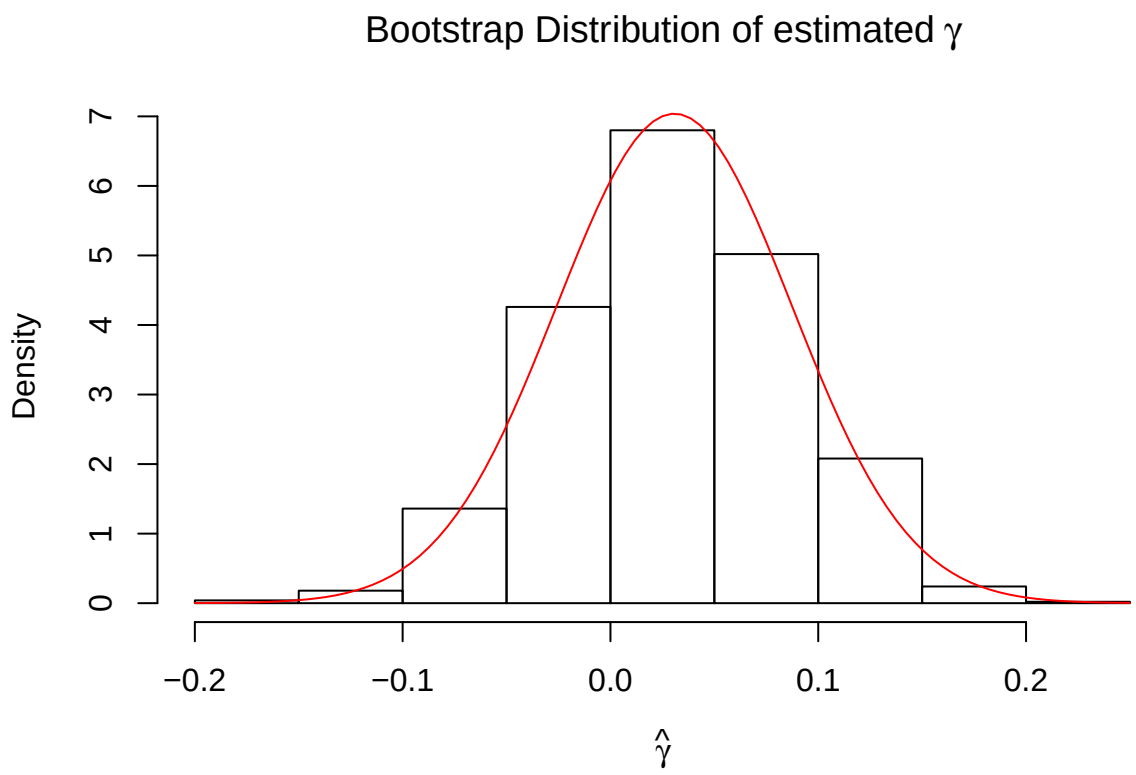
```
bs.reg <- function(data, B)
{
  output= matrix(0, ncol=3, nrow=B)
  y= data[,1] # birth length
  m= data[,2] # mother height
  f= data[,3] # father height
  n= length(data[,1]) # number of observed values, n=42
  model= lm(y ~ m + f, data) # assumed model
  estimates= model$coefficients # estimated coef's of alpha, beta and gamma
  yest= estimates[1] + estimates[2]*m + estimates[3]*f # fitted model
  res= y - yest # residuals

  for(i in 1:B)
    {
    error= sample(res, n, replace=TRUE) # random sample from residuals
    ystar= estimates[1] + estimates[2]*m + estimates[3]*f + error # bootstrap responses
    new.model= lm(ystar ~ m + f, data) # bootstrap model
    output[i,]= new.model$coefficients # new estimates for alpha, beta and gamma
```

```
    }
  output
}
```

This function is then run and a histogram is produced to help determine the sampling distribution of $\hat{\gamma}$. We can see that when a suitable Normal distribution that uses the mean and standard error of the bootstrap coefficients for γ is superimposed over the plot, there is evidence that the distribution of the simulated γ values are in fact normally distributed.

```
bootstrap <- bs.reg(data, B=1000)
hist(bootstrap[,3], freq=FALSE, main=expression(paste("Bootstrap Distribution of estimated"
      ~gamma)), xlab=expression(hat(gamma)))
curve(dnorm(x, mean=mean(bootstrap[,3]), sd=sd(bootstrap[,3])), col=2, add=T)
```



We can work out the *bias*, *standard error* and the $P(\gamma > 0)$ easily in R.

```
# Bias for gamma:
bias <- mean(bootstrap[,3])-estimates[3]
bias
```

```
##         gamma
## -0.0005357739
```

We can see from above that the bias for $\hat{\gamma}$ is $-5.3577387 \times 10^{-4}$ and is very small so could be argued to be unbiased estimator for γ.

```
# Standard error for gamma:
std.error <- sqrt(var(bootstrap[,3]))
std.error
```

```
## [1] 0.05667702
```

Similarly for the standard error, we get a value of 0.056677 which is relatively small.

```
# P(gamma>0):
prob.gamma <- sum(bootstrap[,3]>0)/length(bootstrap[,3])
prob.gamma
```

```
## [1] 0.708
```

Our estimate for $P(\gamma > 0)$ suggests that 70.8% of the data for estimates of γ are greater than 0. If the true value of γ was equal to 0, then we would expect the proportion of bootstrap estimates that are greater than 0 to be equal to 0.5. However, since we had an answer bigger than 0.5, this suggests it is likely that $\gamma \neq 0$ and so significant in the model.

(iv) For the parameter β, we wish to estimate a 95% confidence interval. This will involve using the bootstrap-t methodology that was explored in lectures.

The theory behind this is to generate B bootstrap samples and then for each bootstrap estimate, compute a corresponding z-score. The formula for this is, for $i = 1, \ldots, B$,

$$z^*(i) = \frac{\hat{\beta}^*(i) - \hat{\beta}}{\hat{\sigma}_i^*},$$

where $\hat{\beta}^*(i)$ is the estimated value for β in the i^{th} bootstrap sample, and $\hat{\sigma}_i^*$ is the corresponding estimated standard error.
These z-scores are then ordered and we select an appropriate percentile given by some value $\hat{t}^{(\alpha)}$ such that for a specified α,

$$\frac{\#\{z^*(i) \leq \hat{t}^{(\alpha)}\}}{B} = \alpha$$

Since we have $B = 1000$ bootstrap samples and in this case, $\alpha = 0.025$ since we are computing a $100(1-2\alpha)\% = 95\%$ confidence interval, we get that $\hat{t}^{(0.025)}$ is the 25^{th} largest value and $\hat{t}^{(0.975)}$ is the 975^{th} largest value. Hence, the bootstrap confidence interval is

$$(\hat{\beta} - \hat{t}^{(1-\alpha)}\hat{\sigma}, \hat{\beta} - \hat{t}^{(\alpha)}\hat{\sigma})$$

where $\hat{\sigma}$ is the standard error of $\hat{\beta}$.

If we take the bootstrap function we wrote earlier and add an extra line that will calculate the z-score for each bootstrap estimate we sample for β, we can then implement the steps above to find a suitable 95% confidence interval.

```
bs.ci <- function(data, B)
{
  output= matrix(0, ncol=3, nrow=B)
  z= 0
  y= data[,1] # birth length
  m= data[,2] # mother height
  f= data[,3] # father height
  n= length(data[,1]) # length of data = 42
  model= lm(y ~ m + f, data) # assumed model
  estimates= model$coefficients # estimates for alpha, beta and gamma
  yest= estimates[1] + estimates[2]*m + estimates[3]*f # fitted model
  res= y - yest # residuals

  for(i in 1:B)
    {
    error= sample(res, n, replace=TRUE) # random errors
```

```
    ystar= estimates[1] + estimates[2]*m + estimates[3]*f + error # bootstrap fitted model
    new.model= lm(ystar ~ m + f, data)
    std.error= summary(new.model)$coef[2,2] # standard error for beta from the summary table
    output[i,]= new.model$coefficients # new model estimates for alpha, beta and gamma
    z[i]= (output[i,2]-estimates[2])/std.error # z scores for beta (second variable coefficient)
    }
  z
}
```

Like in the procedure above, these computed z-scores then need to reordered into ascending ordered so that percentiles can be obtained. The code below shows how this is done in R, and how we pick out the appropriate $\hat{t}^{(\alpha)}$ values that will be used to find the end points of the confidence interval.

```
z.scores <- bs.ci(data, B=1000)
sorted <- sort(z.scores) # re-ordered z scores (increasing size)
# estimated t(0.025) value
t.lower <- sorted[25]
t.lower
```

```
## [1] -2.071332
```

```
# estimated t(0.975) value
t.upper <- sorted[975]
t.upper
```

```
## [1] 1.918291
```

```
# original data beta standard error
sigma <- summary(model)$coef[2,2]
sigma
```

```
## [1] 0.06592566
```

```
conf.int <- c(estimates[2] - t.upper*sigma, estimates[2] - t.lower*sigma)
conf.int
```

```
##       beta       beta
## 0.04518369 0.30820222
```

Thus, our 95% confidence interval for β is

$$\begin{aligned}\text{C.I} &= (\hat{\beta} - \hat{t}^{(1-\alpha)}\hat{\sigma},\ \hat{\beta} - \hat{t}^{(\alpha)}\hat{\sigma}) \\ &= (0.1716483 - (1.9182909 \times 0.0659257),\ 0.1716483 - (-2.0713325 \times 0.0659257)) \\ &= (0.0451837,\ 0.3082022)\end{aligned}$$

We can see that this is not symmetric around the β estimate since we have not used the traditional method for calculating t-scores. However, it can be observed that 0 does not lie in the confidence interval and therefore it does not seem plausible that $\beta = 0$ in this case.

(v) Now we are considering the mother's and father's height as a single bivariate set of paired data. Hence if we wish to estimate a 95% confidence interval for the difference in the means of the mother's and father's heights i.e. $E[M] - E[F]$, we can implement the same methodology as above when we worked out z-scores for β except we are doing this for a difference of means.
Let M and F be the set of mother's and father's heights respectively and let their difference $M-F = diff$, $\mathrm{E}[diff] = \mathrm{E}[M - F] = \mathrm{E}[M] - \mathrm{E}[F] = D$. Then, for $i = 1, \ldots, B$, the z-scores that are needed to help construct a 95% confidence interval for $E[M] - E[F]$ are

$$z^*(i) = \frac{D^*(i) - D}{s_i^*/\sqrt{n}},$$

where $D^*(i)$ is the i^{th} bootstrap estimated value for the difference in means of mother's and father's height, and s_i^* is the estimated standard error for the i^{th} bootstrap.

The function below will be run for $B = 1000$ to obtain z-scores using the formula above.

```
bs.diff <- function(data, B)
{
  diff= data[,2] - data[,3]
  mean.diff= mean(diff) # mean difference between mother's and father's heights
  n= length(data[,2])
  x= matrix(0, ncol=2, nrow=n)
  z= 0
  for(i in 1:B)
  {
    s= sample(n, n, replace=TRUE) # sample of length n, from n, with replacement
    for(j in 1:n)
    {
      x[j,1]= data[s[j],2]
      x[j,2]= data[s[j],3]
    }
    new.diff= x[,1]-x[,2] # diff between new bootstrap columns of x
    new.mean= mean(new.diff) # new mean difference
    z[i]= (new.mean-mean.diff)/(sd(new.diff)/sqrt(n)) # z-scores for the bootstrap differences
  }
  z
}
```

Once again, these z-scores need to be reordered and the correct percentiles for $\alpha = 0.025$ and $1 - \alpha = 0.975$ be found.

```
diff.z.scores <- bs.diff(data, B=1000)
diff.sorted <- sort(diff.z.scores) # re-ordered z scores (increasing size)
# estimated t(0.025) value
t.lower2 <- diff.sorted[25]
t.lower2
```

```
## [1] -1.848122
```

```
# estimated t(0.975) value
t.upper2 <- diff.sorted[975]
t.upper2
```

```
## [1] 2.155662
```

```
# standard error of the original difference in means
diff.sigma <- sd(data[,2]-data[,3])/sqrt(length(data[,2]))
diff.sigma
```

```
## [1] 0.5029777
```

```
# mean of the difference between mother's and father's heights
mean.diff <- mean(data[,2]-data[,3])

conf.int2 <- c(mean.diff - t.upper2*diff.sigma,  mean.diff - t.lower2*diff.sigma)
conf.int2
```

```
## [1] -7.441393 -5.427579
```

Hence, the 95% confidence interval for $E[M] - E[F]$ is

$$\begin{aligned}\text{C.I} &= ((E[M] - E[F]) - \hat{t}^{(1-\alpha)} \times \frac{s}{\sqrt{n}},\ (E[M] - E[F]) - \hat{t}^{\alpha)} \times \frac{s}{\sqrt{n}}) \\ &= (-6.3571429 - (2.1556622 \times 0.5029777),\ -6.3571429 - (-1.848122 \times 0.5029777)) \\ &= (-7.4413929,\ -5.4275787)\end{aligned}$$

Since our entire confidence interval is below 0, there is no evidence to suggest that $E[M] = E[F]$. In context of the question, this means that it does not seem plausible that the mean of the mother's heights and the mean of the father's heights are equal. In fact, the confidence interval that has been produced suggests that on average, the height of the father is greater than the height of the mother and hence the negative difference in means.

iv) Finally, we wish to consider only the length of new born babies and the smoking status of their mothers. Therefore we only take the first and fourth columns of the data provided. We also wish to split this data into two so that we have a set of lengths of babies whose mothers are nonsmokers (S=0) and another set of lengths of babies with mothers who do smoke (S=1). The code below shows how this was done.

```
smoker <- subset(data, smoker==1, select=c(length, smoker))
nonsmoker <- subset(data, smoker==0, select=c(length, smoker))
```

Let's define $\theta = E[Y|S = 0] - E[Y|S = 1]$.

```
theta <- mean(nonsmoker$length)-mean(smoker$length)
theta
```

```
## [1] 0.5181818
```

In order to estimate $P(\hat{\theta} > 0)$, we can use bootstrap methodology to produce a sample of estimates for θ and then calculate the proportion of those values that are greater than 0. Below, the function written simulates such a bootstrap. The function is then run to obtain $B = 1000$ estimates for θ.

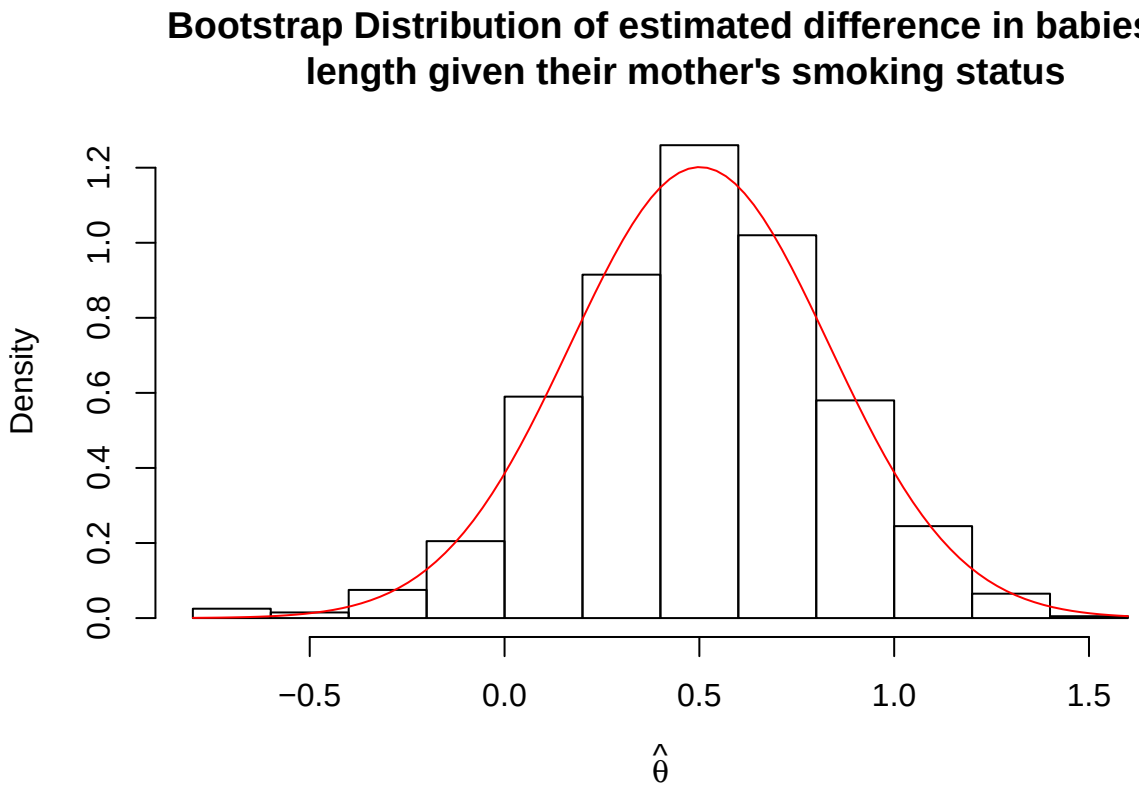
```
bs.theta <- function(ns, s, B)
{
  theta= 0
  for(i in 1:B)
  {
    nsboot= sample(ns, length(ns), replace=TRUE)
    sboot= sample(s, length(s), replace=TRUE)
    theta[i]= mean(nsboot) - mean(sboot)
  }
  theta
}

theta.hat <- bs.theta(nonsmoker[,1], smoker[,1], B=1000)
mean(theta.hat)
```

```
## [1] 0.5008955
```

A histogram of the 1000 bootstrap estimates for θ has been created to help us draw any conclusions about what $\hat{\theta}$ is describing.

```
hist(theta.hat, freq=FALSE, main="Bootstrap Distribution of estimated difference in babies
     length given their mother's smoking status", xlab= expression(hat(theta)))
curve(dnorm(x, mean=mean(theta.hat), sd=sd(theta.hat)), col=2, add=T)
```



Like with previous histogram plots, the pdf of a Normal distribution with parameters that use the mean and standard error of $\hat{\theta}$ has been superimposed over the top. We can observe that this red line suggests that the data for $\hat{\theta}$ could be normally distributed since there appears to be a close fit with the shape of the histogram. We can also see that the majority of the density is to the right of 0, so this implies that $\hat{\theta}$ is mostly a positive number so babies with mothers who do not smoke are bigger in length than babies whose mothers were recorded to smoke.

```
# P(theta>0)
prob.theta <- sum(theta.hat>0)/length(theta.hat)
prob.theta
```

```
## [1] 0.936
```

Our conclusions from the histogram are further cemented once we actually calculate $P(\hat{\theta} > 0)$. From the code we can see that $P(\hat{\theta} > 0) = 0.936$. If there didn't appear to be any correlation between the length of babies and the smoking status of the mother, then only around half of our estimates would be greater than 0. Since this probability is greater than 0.5, this implies that the lengths of babies whose mothers don't smoke are longer than babies with mothers who do smoke.