IDA Assignment 1

Johnny Lee, s1687781

Q1.

Let us suppose that on a (hypothetical) survey there is a question about alcohol consumption and that the variable ALQ records the respondent's answer to the following specific question: "In the past year, have you had at least 12 drinks of any type of alcoholic beverage?". The possible answers are 'Yes' or 'No'. Not all participants respond to this question, that is, the ALQ variable has some missing values. Further, and again hypothetically, suppose that we only have additional data on the gender of the participants in the survey (which is fully observed). For each of the following situations, choose, justifying, the correct answer.

a)

Suppose that ALQ is MCAR. The probability of ALQ being missing for those with ALQ=Yes is 0.3. What is the probability of ALQ being missing for those with ALQ=No?

i) 0.03
ii) 0.3
iii) 0.33

Answer : (ii)

Given the probability of ALQ being missing for those with ALQ=Yes is 0.3, and ALQ is MCAR, we can have the equation as below.

$$\begin{aligned} &f(0|ALQ = Yes, gender, \psi) = f(0|\psi) = 0.3 \\ &= f(0|ALQ = No, gender, \psi) = f(0|\psi),\ \forall ALQ, gender, \psi \end{aligned} \tag{1}$$

where under MCAR, the missing values are completely irrelevant to the existing data, observed values and missing values. Therefore, the probability of ALQ being missing for those with ALQ=No has the same probability as of ALQ being missing for those with ALQ=Yes. Thus, (ii) is the correct option and (i) and (iii) are incorrect options.

b)

ALQ being MAR given gender means,

i) The probability of ALQ being missing depends on the Yes/No value of ALQ even after adjusting for gender.
ii) The probability of ALQ being missing is independent of the Yes/No value of ALQ after adjusting for gender.
iii) The probability of ALQ being missing is independent of the Yes/No value of ALQ100 and gender.

Answer : (ii)

Given that ALQ is MAR, it implies that the probability of ALQ being missing depends on the individual gender. However, from [**pg9, Lecture w1**], within groups defined by individuals with similar gender then the probability of a subject having a missing ALQ is the same as for any other subject. Therefore we conclude that the probability of ALQ being missing is independent of the Yes/No value of ALQ after adjusting for gender. Thus, (ii) is the correct answer and (i) and (iii) are incorrect.

c)

Suppose again that ALQ is MAR given gender, and that the probability of ALQ being missing for men is 0.1. What is the probability of ALQ being missing for women?

i) 0.1
ii) 0.9
iii) It is impossible to conclude from the information given

Answer : (iii)

Given that ALQ is MAR,the probability of ALQ being missing for women cannot be concluded from the given information. The option (i) suggest ALQ to be MCAR as the probability for being missing is same as men and it violates the assumption that ALQ is MAR. The option (ii) requires additional information to determine its probability to be equal to 0.9. Thus, (iii) is the correct answer and (i) and (ii) are incorrect.

Q2.

Suppose that a dataset consists of 100 subjects and 10 variables. Each variable contains 10% of missing values. What is the largest possible subsample under a complete case analysis? What is the smallest? Justify.

Answer : 90 **(largest),** 0**(smallest)**

We have 100 subjects and 10 variables, and the dataset can also be interpreted as a matrix with dimension 100×10. Overall, it contains 1000 values in the dataset. Suppose each column contains 10% of missing values, each column has 10 missing values.

Case 1 : Largest possible subsample under a complete case analysis

We can consider the case when each column contains the missing value for the same subjects. As each column needs to have 10 missing values, we can select 10 random subjects out of 100 subjects. Now, to perform complete case analysis and removing the missing values, the largest possible subsample under a complete case analysis is **90**.

Case 2 : Smallest possible subsample under a complete case analysis

We can consider the case when each column contains the missing value for the different subjects. In other words, no column contains the same missing value for a particular subject. For instance, if the first column contains missing values from subject 1 to 10, the second column contains missing values from subject 11 to 20. If this continues, we can have all subjects with missing variable. Now, to perform complete case analysis and removing the missing values, the smallest possible subsample under a complete case analysis is **0**.

Q3.

Consider a two variable (Y_1, Y_2) problem, with each variable defined as follows:

$$Y_1 = 1 + Z_1$$
$$Y_2 = 5 + 2Z_1 + Z_2$$

where Y_1 is fully observed but Y_2 is subject to missingness. Further consider that Y_2 is missing if $a(Y_1 - 1) + b(Y_2 - 5) + Z_3 < 0$, where Z_1, Z_2, and Z_3 follow independent standard normal (that is, mean 0 and variance 1) distributions. Important: Please use `set.seed(1)` in R when simulating the data, for reproducibility reasons.

a)

Start by simulating a (complete) dataset of size 500 on (Y_1, Y_2). Then, and considering $a = 2$ and $b = 0$, simulate the corresponding observed dataset (by imposing missingness on Y_2 as instructed above). Is this mechanism **MCAR, MAR,** or **MNAR**? Display the marginal distribution of Y_2 for the complete (as originally simulated) and observed (after imposing missingness) data. Comment.

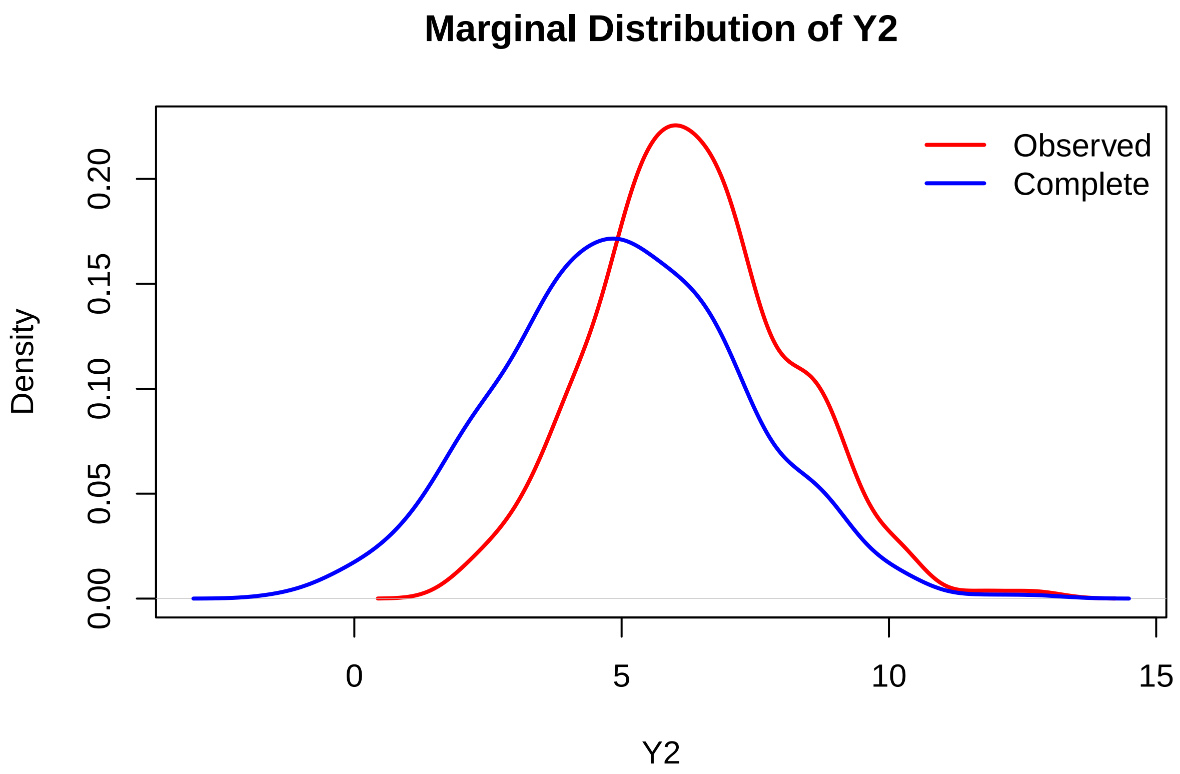
Answer :

```r
set.seed(1)
#generating each Z
n <- 500
z1 <- rnorm(n, 0, 1)
z2 <- rnorm(n, 0, 1)
z3 <- rnorm(n, 0, 1)

#generating Y_1 and Y_2
y1 <- 1 + z1
y2 <- 5 + 2*z1 + z2
y2.missing <- 5 + 2*z1 + z2

#generating missing values
mn <- function(a,b){
  a * (y1-1) + b* (y2-5) + z3 >= 0
}
r <- mn(a=2,b=0)

#imposing missing ness
y2.missing[!r] <- NA
y2.observed <- y2[r]

y1y2 <- data.frame(y1 = y1, y2 = y2.missing)
```

After simulating 500 samples of (Y_1, Y_2) and imposing missingness on Y_2, we can conclude that this mechanism is **MAR**. This is because the missingness of Y_2 is dependent on Y_1 where Y_1 is fully observed and independent of Y_2 as $a = 2$ and $b = 0$. Therfore when the value of Y_1 is less than 1, the value of Y_2 in the same index becomes missing. There is no information given for Z_3 other than it is randomly generated. Therefore we assume that Z_3 is fully observed.

Now, we continue our analysis by plotting the marginal distribution of Y_2.

```
#Plotting marginal distribution of Y2
plot(density(y2.observed),
     xlim = c(min(density(y2)$x, density(y2.observed)$x), max(density(y2)$x, density(y2.observed)$x)),
     ylim = c(min(density(y2)$y, density(y2.observed)$y), max(density(y2)$y, density(y2.observed)$y)),
     col= "red", lwd = 2,
     main="Marginal Distribution of Y2",
     xlab = "Y2")
legend("topright", legend = c("Observed", "Complete"),
       col = c("red", "blue"), lty = c(1, 1), lwd = c(2, 2), bty = 'n')
lines(density(y2), col="blue", lwd = 2)
```



By imposing the missingness in Y_2, we can notice the significant difference in the density between Observed and Complete. As the missingness occurs for the lower values so we the observed distribution shifted to the right.

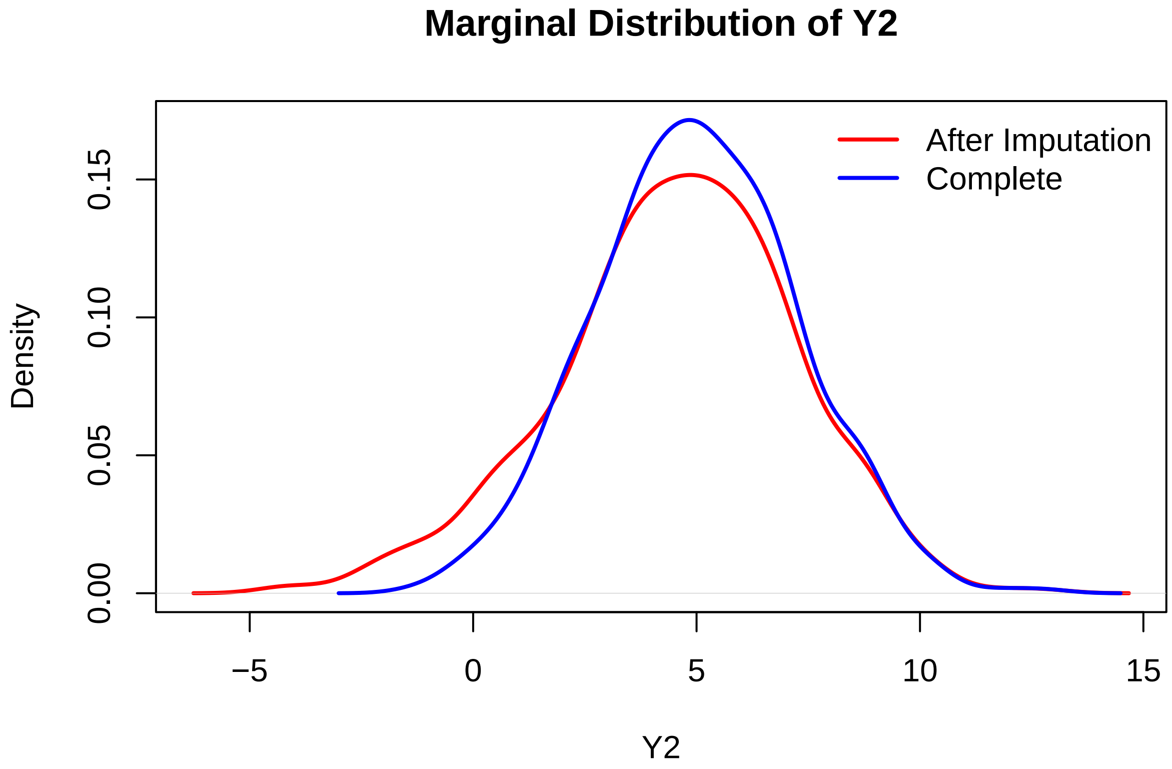
b)

For the observed dataset simulated in (a), impute the missing values using stochastic regression imputation. Display the marginal distribution of Y_2 for the complete (as originally simulated) and completed (after imputation) data. Comment.

Answer :

```r
set.seed(1)
#Fitting Linear Model
fit <- lm(y2 ~ y1, data = y1y2)

#generating noise
noise <- rnorm(nrow(y1y2), mean = 0, sd = sigma(fit))

#Stochastic Regression Imputation
predsri <- predict(fit, newdata = y1y2[1]) + noise
y1y2$y2 <- ifelse(is.na(y1y2$y2)==TRUE, predsri, y1y2$y2)
```

```r
plot(density(y1y2$y2),
     xlim = c(min(density(y2)$x, density(y1y2$y2)$x), max(density(y2)$x, density(y1y2$y2)$x)),
     ylim = c(min(density(y2)$y, density(y1y2$y2)$y), max(density(y2)$y, density(y1y2$y2)$y)),
     col= "red", lwd = 2,
     main="Marginal Distribution of Y2", xlab = "Y2")
legend("topright", legend = c("After Imputation", "Complete"),
       col = c("red", "blue"), lty = c(1, 1), lwd = c(2, 2), bty = 'n')
lines(density(y2), col="blue", lwd = 2)
```



```r
cat("CCA mean:",  mean(y2), "CCA standard error:", sd(y2) / sqrt(length(y2)), "\n")
```

```
## CCA mean: 4.999348 CCA standard error: 0.1003737
```

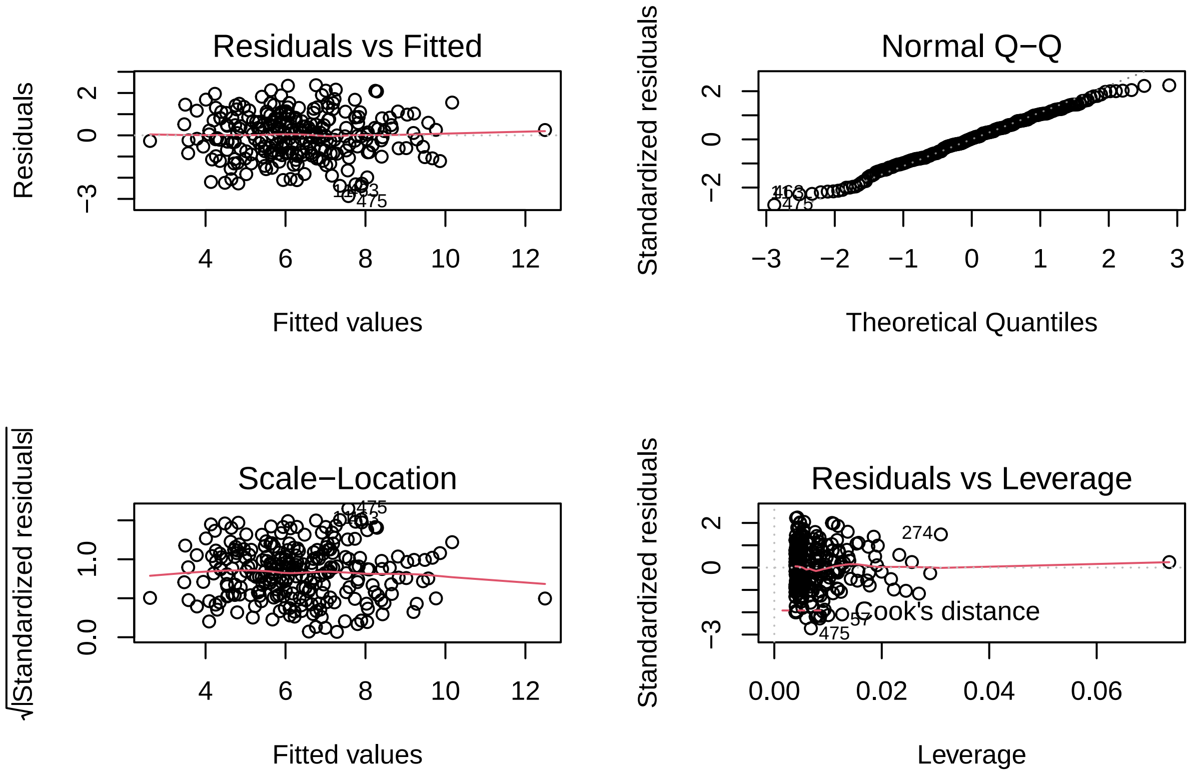
```
cat("SRI mean:",  mean(y1y2$y2, na.rm=TRUE),
    "SRI standard error:", sd(y1y2$y2, na.rm = TRUE) / sqrt(nrow(y1y2)))
```

```
## SRI mean: 4.550714 SRI standard error: 0.1195328
```

```
summary(fit)
```

```
##
## Call:
## lm(formula = y2 ~ y1, data = y1y2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.86708 -0.78476  0.04138  0.79522  2.36619
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.87014    0.16924   16.96   <2e-16 ***
## y1           2.00000    0.09033   22.14   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.057 on 252 degrees of freedom
##   (      246         .)
## Multiple R-squared:  0.6605, Adjusted R-squared:  0.6591
## F-statistic: 490.2 on 1 and 252 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(2,2))
plot(fit)
```



After the stochastic regression imputation, we noticed that the peak of the density reduced and formed a similar shape to the complete dataset. Also the difference between the two means are 0.449. However, we observed that the standard error increased.

By looking at the summary, the regression equation is as below.

$$Y_2 = 2.87014 + 2 * Y_1 \tag{2}$$

We can see that the the coefficient of Y_1 is significant and its p-value is less than 0.05. Then we can conclude that the a linear model can be used to fit for the missing values. To further check the assumptions let us look at the plots above.

The top left is the residual plot to check the linear fit. There is no significant pattern within the residual and a critical outlier. Thus, this further solidfies the linear relationship between y2 and y1 values. To check the normality, we look at the Normal Q-Q plot on the top right. The residuals generally follow the straight line. Therefore, we can also conclude that the normality assumption was not violated. To check if the residuals are equally spreaded along the ranges of predictors, we look at the Scale-Location plot. We can see sufficient variation among the plots without any noticeable trend. At the bottom right, the Cook's distance lines are not seen in a large portion within the plots. We can conclude that there is no influential outlier that affects the model. Then, homoscedasticity and independence of residual error terms shown in scale-location plot and residual vs leverage plot.

c)

Using the complete dataset simulated in (a), now impose missingness on Y_2 by considering $a = 0$ and $b = 2$. Is this mechanism **MCAR, MAR,** or **MNAR**? Display the marginal distribution of Y_2 for the complete (as originally simulated) and observed (after imposing missingness) data. Comment.

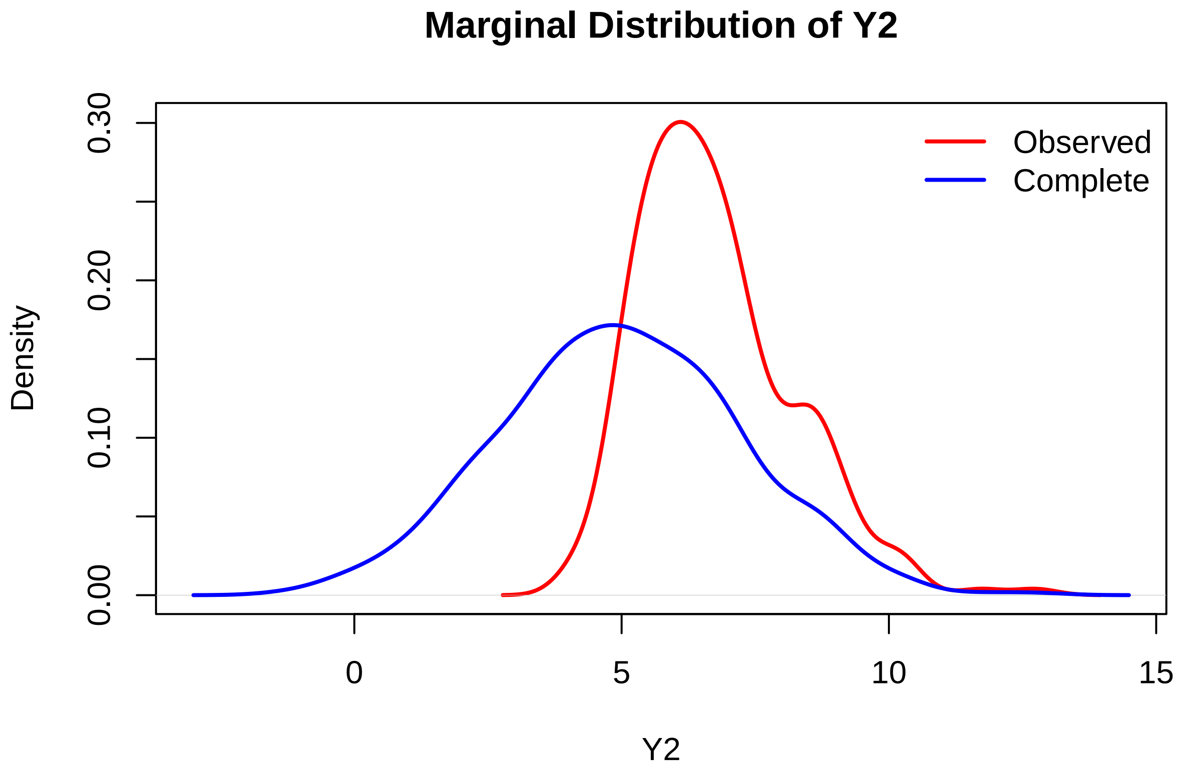
Answer :

```
set.seed(1)
#generating each Z
n <- 500
z1 <- rnorm(n, 0, 1)
z2 <- rnorm(n, 0, 1)
z3 <- rnorm(n, 0, 1)

#generating Y_1 and Y_2
y1 <- 1 + z1
y2 <- 5 + 2*z1 + z2
y2.missing <- 5 + 2*z1 + z2

#generating missing values
mn <- function(a,b){
  a * (y1-1) + b* (y2-5) + z3 >= 0
}
r <- mn(a=0,b=2)

#imposing missing ness
y2.missing[!r] <- NA
y2.observed <- y2[r]

y1y2 <- data.frame(y1 = y1, y2 = y2.missing)
```

After simulating 500 samples of (Y_1, Y_2) and imposing missingness on Y_2 with $a = 0$ and $b = 2$, we can conclude that this mechanism is **MNAR**. Given that Z_3 is generated together, the missingness of Y_2 is only dependent on Y_2 itself. Thus, the value of Y_2 is missing when the actual $2(Y_2 - 5) + Z_3 < 0$. Again, the distribution of the observed data are shifted to the right as the missingness occurs below 5.

Now, we continue our analysis by plotting the marginal distribution of Y_2.

```
#Plotting marginal distribution of Y2
plot(density(y2.observed),
     xlim = c(min(density(y2)$x, density(y2.observed)$x), max(density(y2)$x, density(y2.observed)$x)),
     ylim = c(min(density(y2)$y, density(y2.observed)$y), max(density(y2)$y, density(y2.observed)$y)),
     col= "red", lwd = 2,
     main="Marginal Distribution of Y2", xlab = "Y2")
legend("topright", legend = c("Observed", "Complete"),
       col = c("red", "blue"),
       lty = c(1, 1), lwd = c(2, 2), bty = 'n')
lines(density(y2), col="blue", lwd = 2)
```



We can clearly see that the marginal distribution between the observed and complete is different. The observed distribution seem to have two separate peak whereas the complete data has a general shape.

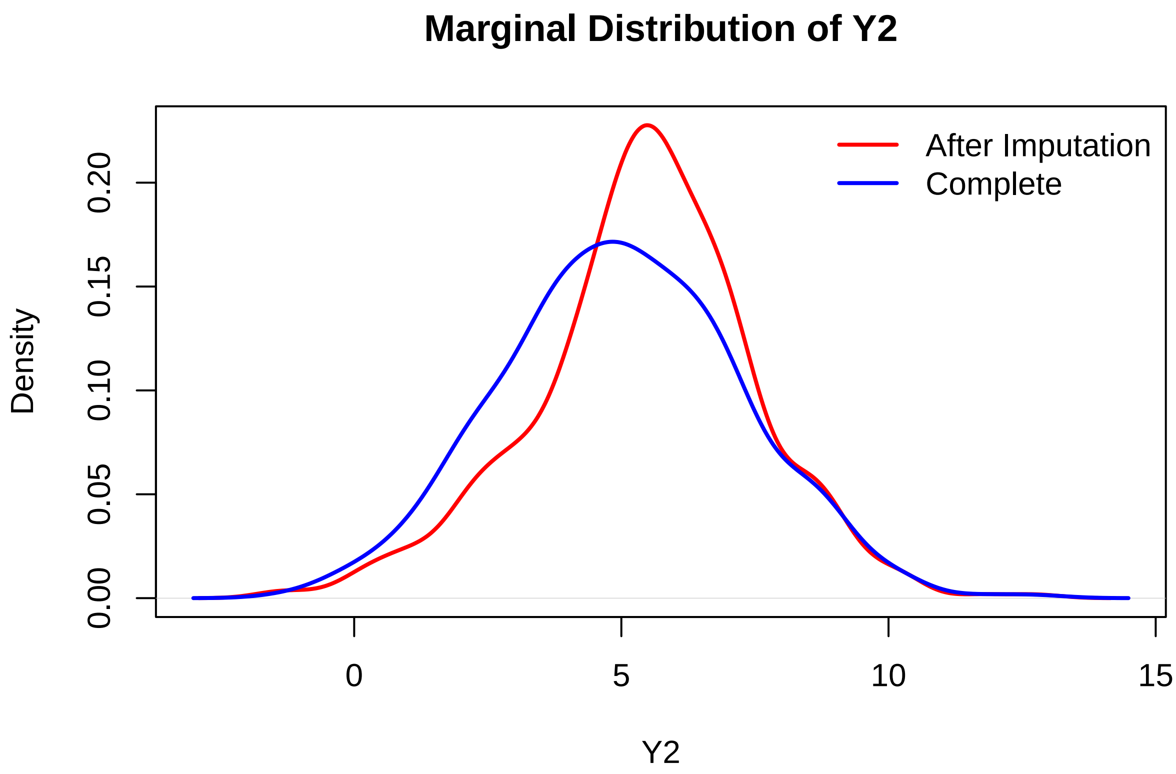
d)

The same as in (b) but for the observed data generated in (c).

Answer :

```
set.seed(1)

#Fitting Linear Model
fit <- lm(y2 ~ y1, data=y1y2)

#generating noise
noise <- rnorm(nrow(y1y2), mean = 0, sd = sigma(fit))

#Stochastic Regression Imputation
predsri <- predict(fit, newdata = y1y2[1]) + noise
y1y2$y2 <- ifelse(is.na(y1y2$y2)==TRUE, predsri, y1y2$y2)
```

```
plot(density(y1y2$y2),
     xlim = c(min(density(y2)$x, density(y1y2$y2)$x), max(density(y2)$x, density(y1y2$y2)$x)),
     ylim = c(min(density(y2)$y, density(y1y2$y2)$y), max(density(y2)$y, density(y1y2$y2)$y)),
     col= "red", lwd = 2,
     main="Marginal Distribution of Y2", xlab = "Y2")
legend("topright",
       legend = c("After Imputation", "Complete"),
       col = c("red", "blue"),
       lty = c(1, 1), lwd = c(2, 2), bty = 'n')
lines(density(y2), col="blue", lwd = 2)
```



```
cat("CCA mean:",  mean(y2), "CCA standard error:", sd(y2) / sqrt(length(y2)), "\n")
```

```
## CCA mean: 4.999348 CCA standard error: 0.1003737
```

```
cat("SRI mean:",  mean(y1y2$y2, na.rm=TRUE),
    "SRI standard error:", sd(y1y2$y2, na.rm = TRUE) / sqrt(nrow(y1y2)))
```

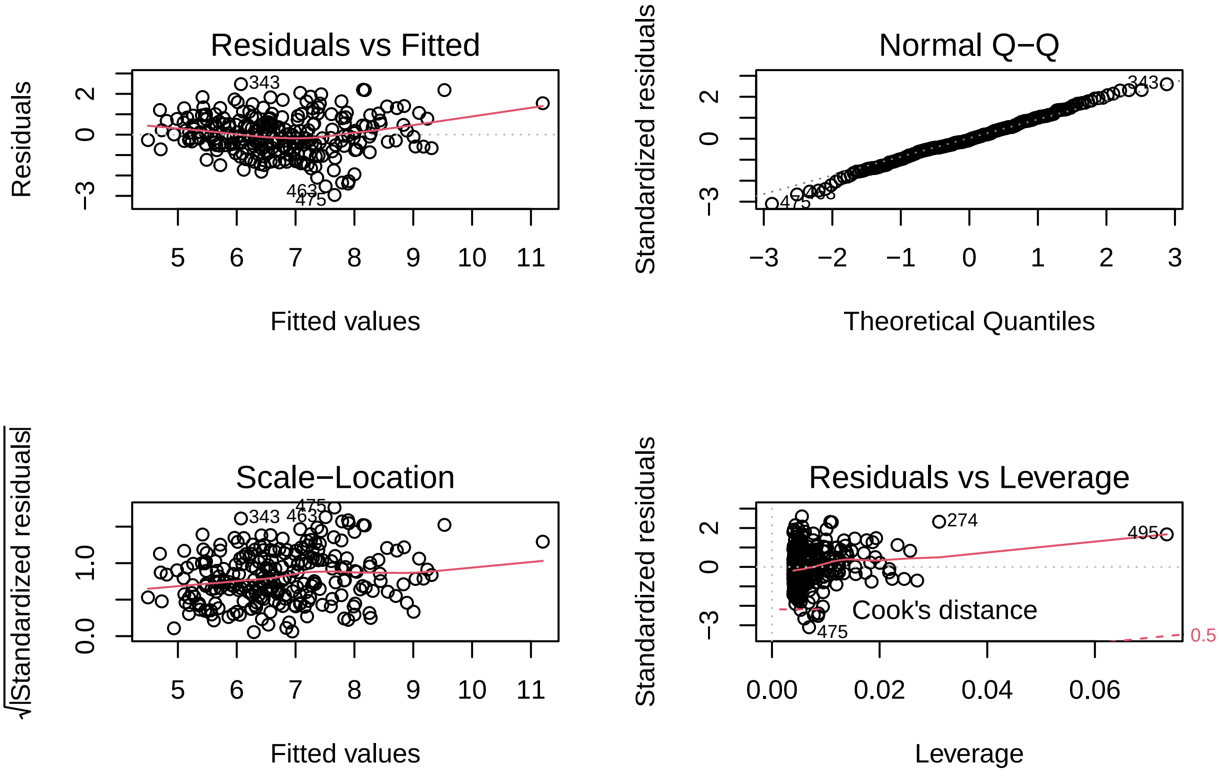
```
## SRI mean: 5.426805 SRI standard error: 0.0926159
```

```
summary(fit)
```

```
##
## Call:
## lm(formula = y2 ~ y1, data = y1y2)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.95667 -0.53998 -0.05132  0.60061  2.47652
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.27803    0.15200   28.14   <2e-16 ***
## y1           1.43925    0.08136   17.69   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9559 on 250 degrees of freedom
##   (      248          .)
## Multiple R-squared:  0.5559, Adjusted R-squared:  0.5541
## F-statistic:   313 on 1 and 250 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(2,2))
plot(fit)
```



After the stochastic regression imputation, we notice that the peak of the density reduced and the density plot shifted back. Unlike in (a), the mean value of the observed data has increased after the imputation with difference of 0.427. However, we observed that the standard error decreased.

By looking at the summary, the regression equation is as below.

$$Y_2 = 4.27803 + 1.43925 * Y_1 \tag{3}$$

We can see that the the coefficient of Y_1 is significant and its p-value is less than 0.05. Then we can conclude that the a linear model can be used to fit for the missing values. To further check the assumptions let us look at the plots above.

The top left is the residual plot to check the linear fit. There is no significant pattern within the residual and a critical outlier. Thus, this further solidfies the linear relationship between Y_2 and Y_1 values. To check the normality, we look at the Normal Q-Q plot on the top right. The residuals generally follow the straight line. Therefore, we can also conclude that the normality assumption was not violated. To check if the residuals are equally spread along the ranges of predictors, we look at the Scale-Location plot at bottom left. To that, we can see sufficient variation among the plots without any noticeable trend. At the bottom right, the Cook's distance lines are not seen in a large portion within the plots. We can conclude that there is no influential outlier that affects the model. Then, homoscedasticity and independence of residual error terms shown in scale-location plot and residual vs leverage plot.

Overall, stochastic regression imputation is not a good imputation method for the case when $a = 0$ & $b = 2$. Since, the mechanism is **MNAR** it is hard to impute and definitely not desirable with single imputation.

Q4.

It is sometimes necessary to lower a patient's blood pressure during surgery, using a hypotensive drug. Such drugs are administrated continuously during the relevant phase of the operation; because the duration of this phase varies, so does the total amount of drug administered. Patients also vary in the extent to which the drugs succeed in lowering blood pressure. The sooner the blood pressure rises again to normal after the drug is discontinued, the better. The dataset `databp.Rdata` available on Learn, a partial missing value version of the data presented by Robertson and Armitage (1959), relate to a particular hypotensive drug and give the time in minutes before the patient's systolic blood pressure returned to $1000mm$ of mercury (the recovery time), the logarithm (base 10) of the dose of drug in milligrams (you can use this variable as is, no need to transform it to the original scale), and the average systolic blood pressure achieved while the drug was being administered.

```
load("databp.Rdata")
```

a)

Carry out a complete case analysis to find the mean value of the recovery time (and associated standard error) and to find also the (Pearson) correlations between the recovery time and the dose and between the recovery time and blood pressure.

Answer :

```
#Complete Case Analysis
#Computing mean and standard error
mean <- mean(databp$recovtime, na.rm=TRUE)
stderr <- sd(databp$recovtime, na.rm=TRUE) / sqrt(sum(!is.na(databp$recovtime)))

#Correlation between the variables
cor.dose <- cor(databp$recovtime, databp$logdose,
                use = "complete", method = "pearson")
cor.bp <- cor(databp$recovtime, databp$bloodp,
              use = "complete",method = "pearson")

#Printing the computed values.
cat("Mean value:", mean, "\n")
```

```
## Mean value: 19.27273
```

```
cat("Standard Error:", stderr, "\n")
```

```
## Standard Error: 2.603013
```

```
cat("Correlation of recovery time and dose of drug", cor.dose, "\n")
```

```
## Correlation of recovery time and dose of drug 0.2391256
```

```
cat("Correlation of recovery time and blood pressure", cor.bp, "\n")
```

```
## Correlation of recovery time and blood pressure -0.01952862
```

Firstly, we set `set.seed(1)` for consistency. The total number of data is 25 where there are 22 observed and 3 missing. We will benchmark the mean and standard error value to compare during the analysis. To briefly describe the correlation, the correlation between recovery time and drug dosage is weakly positive. On the other hand, the correlation between recovery time and blood pressure is nearly zero but with slight negative correlation.

b)

The same as in (a) but using mean imputation.

Answer :

```
#Implementing Mean Imputation

databp.mi <- databp
databp.mi[is.na(databp$recovtime), 3] <- mean

#Computing mean and standard error
mean.mi <- mean(databp.mi$recovtime, na.rm=TRUE)
stderr.mi <- sd(databp.mi$recovtime, na.rm=TRUE) / sqrt(nrow(databp.mi))

#Correlation between the variables
cor.dose.mi <- cor(databp.mi$recovtime, databp.mi$logdose,
                   use = "complete", method = "pearson")
cor.bp.mi <- cor(databp.mi$recovtime, databp.mi$bloodp,
                 use = "complete",method = "pearson")

cat("Mean value:", mean.mi, "\n")
```

```
## Mean value: 19.27273
```

```
cat("Standard Error:", stderr.mi, "\n")
```

```
## Standard Error: 2.284135
```

```
cat("Correlation of recovery time and dose of drug", cor.dose.mi, "\n")
```

```
## Correlation of recovery time and dose of drug 0.2150612
```

```
cat("Correlation of recovery time and blood pressure", cor.bp.mi, "\n")
```

```
## Correlation of recovery time and blood pressure -0.01934126
```

As the mean value of the 22 observed values are 19.27273, we imputed the mean value to the index where the missing values are found. This ensures the mean of the dataset to remain the same but decrease in standard error. Also, we observed slight decrease in the correlation between dose of drug and blood pressure. However, the correlation did not change significantly.

c)

The same as in (a) but using mean regression imputation. (5 marks)

Answer :

```
#Implementing Regression Imputation
databp.ri <- databp
fit.ri <- lm(recovtime ~ logdose + bloodp, data = databp.ri)
pred.ri <- predict(fit.ri, newdata = databp.ri)
databp.ri$recovtime <- ifelse(is.na(databp.ri$recovtime)==TRUE,
                              pred.ri, databp.ri$recovtime)
```

```
#Computing mean and standard error
mean.ri <- mean(databp.ri$recovtime, na.rm=TRUE)
stderr.ri <- sd(databp.ri$recovtime, na.rm=TRUE) / sqrt(nrow(databp.ri))

#Correlation between the variables
cor.dose.ri <- cor(databp.ri$recovtime, databp.ri$logdose,
                   use = "complete.obs", method = "pearson")
cor.bp.ri <- cor(databp.ri$recovtime, databp.ri$bloodp,
                 use = "complete.obs",method = "pearson")

cat("Mean value:", mean.ri, "\n")
```

```
## Mean value: 19.44428
```

```
cat("Standard Error:", stderr.ri, "\n")
```

```
## Standard Error: 2.312845
```

```
cat("Correlation of recovery time and dose of drug", cor.dose.ri, "\n")
```

```
## Correlation of recovery time and dose of drug 0.2801835
```

```
cat("Correlation of recovery time and blood pressure", cor.bp.ri, "\n")
```

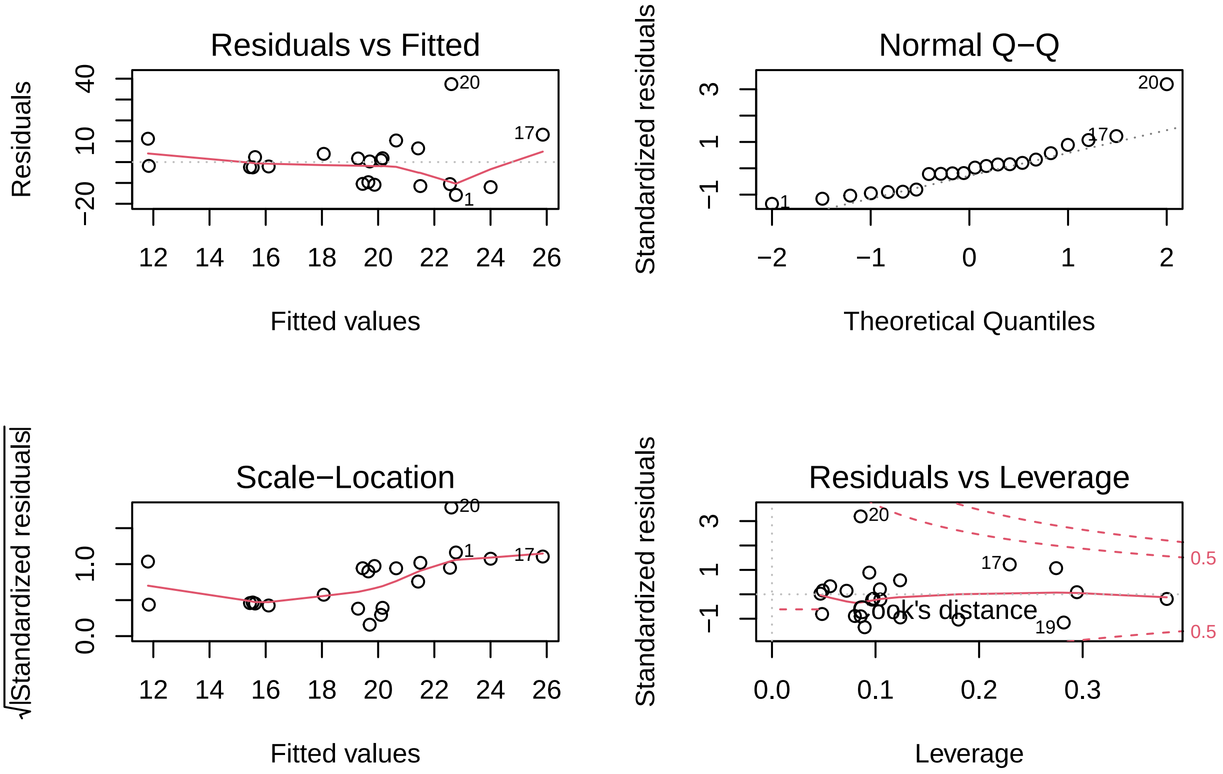
```
## Correlation of recovery time and blood pressure -0.0111364
```

```
summary(fit.ri)
```

```
##
## Call:
## lm(formula = recovtime ~ logdose + bloodp, data = databp.ri)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.768 -10.250  -0.770   3.546  37.394
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  15.2159    19.8203   0.768    0.452
## logdose      11.4290     8.4178   1.358    0.190
## bloodp       -0.2769     0.3411  -0.812    0.427
##
## Residual standard error: 12.25 on 19 degrees of freedom
##   (      3          .)
## Multiple R-squared:  0.08879,    Adjusted R-squared:  -0.007129
## F-statistic: 0.9257 on 2 and 19 DF,  p-value: 0.4134
```

```
par(mfrow=c(2,2))
plot(fit.ri)
```



We performed regression imputation on the `databp` dataset where `recovtime` is the response variable and `logdose` and `bloodp` are the explanatory variables. After imputation, we observed both mean and standard error increased to 19.4428 and 2.312845 respectively. We observed a slight increase in the correlations but the change is insignificant to compare.

By looking at the summary, the regression equation is as below.

$$\text{RecoveryTime} = 15.2159 + 11.4290 * \log(\text{dose}) - 0.2769 * \text{bloodpressure} \tag{4}$$

We can see that all of the coefficients are insignificant and its p-values are greater than 0.05. Then we can conclude that the a linear model is not a good fit. To further check the assumptions let us look at the plots above.

The top left is the residual plot to check the linear fit. There is no significant pattern within the residual and a critical outlier. However, the zero line spikes up at 23 and with the coefficients being insignificant, we suggest that linear assumption was not met. To check the normality, we look at the Normal Q-Q plot on the top right. The residuals generally follow the straight line. Therefore, we can also conclude that the normality assumption was not violated. To check if the residuals are equally spread along the ranges of predictors, we look at the Scale-Location plot at bottom left. To that, we can see sufficient variation among the plots without any noticeable trend. At the bottom right, the Cook's distance lines are not seen in a large portion within the plots. We can conclude that there is no influential outlier that affects the model. Then, homoscedasticity and independence of residual error terms shown in scale-location plot and residual vs leverage plot.

d)

The same as in (a) but using stochastic regression imputation. Do you need any extra care when conducting stochastic regression imputation in this example? (5 marks)

Answer :

```
#Implementing Stochastic Regression Imputation
set.seed(1)
databp.sri <- databp
fit.sri <- lm(recovtime ~ logdose + bloodp, data = databp.sri)
noise <- rnorm(n = nrow(databp), mean = 0, sd = sigma(fit.sri))
# pred.sri <- predict(fit.sri, newdata = databp.sri) + noise
pred.sri <- predict(fit.sri, newdata = databp.sri) + noise
databp.sri$recovtime <- ifelse(is.na(databp.sri$recovtime)==TRUE,
                               pred.sri, databp.sri$recovtime)
```

```
mean.sri <- mean(databp.sri$recovtime, na.rm=TRUE)
stderr.sri <- sd(databp.sri$recovtime, na.rm=TRUE) / sqrt(nrow(databp.sri))

#Correlation between the variables
cor.dose.sri <- cor(databp.sri$recovtime, databp.sri$logdose,
                    use = "complete.obs", method = "pearson")
cor.bp.sri <- cor(databp.sri$recovtime, databp.sri$bloodp,
                  use = "complete.obs",method = "pearson")

cat("Mean value:", mean.sri, "\n")
```

```
## Mean value: 20.4598
```

```
cat("Standard Error:", stderr.sri, "\n")
```

```
## Standard Error: 2.444571
```

```
cat("Correlation of recovery time and dose of drug", cor.dose.sri, "\n")
```

```
## Correlation of recovery time and dose of drug 0.2284537
```

```
cat("Correlation of recovery time and blood pressure", cor.bp.sri, "\n")
```

```
## Correlation of recovery time and blood pressure -0.01786944
```

We performed stochastic regression imputation on the `databp` dataset where `recovtime` is the response variable and `logdose` and `bloodp` are the explanatory variables. After imputation, we observed both mean and standard error increased to 20.4598 and 2.444571 respectively. These values are even higher than in part (b) and (c). We observed a slight decrease in the correlations but the change is insignificant to compare.

We did not perform the model check here as we are using the same linear model as part (c). However, we need to note that the prediction value generated with noise should not be negative as time is a measured value which cannot be expressed in negative values. Thus, it will not make sense to impute a negative value for the recovery time.

e)

You will now conduct the same analysis but applying another technique called predictive mean matching (Little, 1988), which is a special type of hot deck imputation. In the simplest form of this method (and the one you will use here), a regression model is used to predict the variables with missing values from the other (complete) variables. For each subject with a missing value, the donor is chosen to be the subject with a predicted value of her or his own that is closest (to be measured by the squared difference) to the prediction for the subject with the missing value.

Answer :

```
databp.hd <- databp
databp.hd[,3] <- predict(fit.sri, databp.hd)

r <- is.na(databp$recovtime)
pred.hd <- NULL

#predictive mean matchning imputation / hot deck imputation
for (missing in which(r)) {
  #
  recov.time <- databp.hd[missing, 3]
  #Initialising distances vector
  distances <- rep(TRUE, nrow(databp.hd))

  #Preventing duplication
  distances[r] <- NA

  #Computing the distances for observed values
  distances[!r] = (rep(recov.time, sum(!r)) - databp.hd[!r, 3])^2
  donor <- which.min(distances)

  # Storing original value of donor
  pred.hd <- c(pred.hd, databp[donor, 3])
  cat(sprintf('Donor for subject %s is %s with the value of %s\n',
              missing, donor, databp[donor, 3]))
}
```

```
## Donor for subject 4 is 6 with the value of 13
## Donor for subject 10 is 2 with the value of 10
## Donor for subject 22 is 17 with the value of 39
```

```
#hot deck imputation
databp.hd <- databp
databp.hd[is.na(databp$recovtime),3] <- pred.hd
```

```
mean.hd <- mean(databp.hd$recovtime, na.rm=TRUE)
stderr.hd <- sd(databp.hd$recovtime, na.rm=TRUE) / sqrt(nrow(databp.hd))

#Correlation between the variables
cor.dose.hd <- cor(databp.hd$recovtime, databp.hd$logdose,
                   use = "complete.obs", method = "pearson")
cor.bp.hd <- cor(databp.hd$recovtime, databp.hd$bloodp,
                 use = "complete.obs",method = "pearson")

cat("Mean value:", mean.hd, "\n")
```

```
## Mean value: 19.44
```

```
cat("Standard Error:", stderr.hd, "\n")
```

```
## Standard Error: 2.464467
```

```
cat("Correlation of recovery time and dose of drug", cor.dose.hd, "\n")
```

```
## Correlation of recovery time and dose of drug 0.3037945
```

```
cat("Correlation of recovery time and blood pressure", cor.bp.hd, "\n")
```

```
## Correlation of recovery time and blood pressure -0.03208685
```

We performed predictive mean matching imputation on the `databp` dataset where `recovtime` is the response variable and `logdose` and `bloodp` are the explanatory variables. After imputation, we observed both mean and standard error increased to 19.44 and 2.464467 respectively. We observed the magnitude of the correlations increased greatly for both bloodpressure and dosage.

Now, let us compare the overall performance, by referring to the table below,

```
data.frame(method=c("CCA", "MI", "RI", "SRI", "PMM"),
           mean = c(mean, mean.mi, mean.ri, mean.sri, mean.hd),
           S.E.=c(stderr, stderr.mi, stderr.ri, stderr.sri, stderr.hd),
           cor.bp=c(cor.bp, cor.bp.mi, cor.bp.ri, cor.bp.sri, cor.bp.hd),
           cor.dose=c(cor.dose, cor.dose.mi, cor.dose.ri, cor.dose.sri, cor.dose.hd))
```

```
##   method     mean     S.E.      cor.bp  cor.dose
## 1    CCA 19.27273 2.603013 -0.01952862 0.2391256
## 2     MI 19.27273 2.284135 -0.01934126 0.2150612
## 3     RI 19.44428 2.312845 -0.01113640 0.2801835
## 4    SRI 20.45980 2.444571 -0.01786944 0.2284537
## 5    PMM 19.44000 2.464467 -0.03208685 0.3037945
```

Generally, we noticed that there is a decrease in the standard error after imputation. The mean value of stochastic regression imputation was especially higher than others as the missing values could not be fitted with linear model and the noise term could cause to differ from the original mean value. However, due to the noise value, the correlation seem to remain unchanged

We can see that the complete case analysis and mean imputation have the same mean value but lower standard error for mean imputation as the denominator increased from 22 to 25 to calculate the standard error. Also, the correlation between the `recovtime` and `logdose` decreased due to attenuation. However this effect is not seen clearly between the `recovtime` and `bloodp` as the correlation itself is near zero.

We noticed significant change in the correlation using meaning imputation and predictive mean matching. For the mean imputation, the fitted value was not a good fit thus resulted in the decrease in the correlation. However for the predictive mean matching we noticed that the missing values seems to follow the same distribution as the observed value. Therefore this will both increase the correlation in positive and negative direction respectively.

f)

What is an advantage of predictive mean matching over stochastic regression imputation? Based on your analysis, can you foresee any potential problem of predictive mean matching?

Answer :

Advantage The advantage of predictive mean matching is that the linear model is only used for matching the missing subject to potential donor. When the missing values is imputed, it uses the observed value of the donor. Thus, it is less sensitive towards linearity and normality assumptions compared to stochastic regression imputation. Also, it ensures that the distribution of the missing values are the same as the distribution of the observed values of the donor.

Disadvantage However, predictive mean matching can produce significant bias for smaller dataset with relatively larger missing values. In other words, the same donor will be used repetitively for certain missing values. If the assumption above is not validated, it may not be a good replacement for stochastic regression imputation.