# Assignment 1

Arturo Esquivel

**1**

**a)** The correct answer is 0.3 (ii) because since ALQ is MCAR the probability for an observation to be missing is independent of the ALQ value.

$$\Pr(R = 1 \mid ALQ = YES) = \Pr(R = 1 \mid ALQ = NO) = \Pr(R = 1)$$

So, the probability of an observation with ALQ=Yes being missing is the same for when ALQ=No, 0.03.

**b)** The correct answer is (ii). When ALQ is MAR given gender, the probability of an observation being missing changes for different gender values. Gender is assumed to be the only variable affecting the probability of missingness. If gender is adjusted for then the probability of missingness becomes completely at random and independent of what value ALQ takes.

**c)** The correct answer is (iii). It is not possible to compute a probability for women without knowing the probability of missing values without accounting for gender and the proportions of men and women in the sample.

**2**

Under a complete case analysis only subjects with all the variables observed are considered. The biggest possible subsample will occur when the missingness occurs for the same subjects along all variables. All variables have a 10% (10 for a sample of 100) of their observations missing. If the 10 subjects with missing values for $Y_i$ are the same than those for $Y_j$ and for all other variables, then there are 10 subjects with missing values and the subsample will contain 90 subjects.

The smallest subsample occurs when the missing values occur for different subjects along all variables. When values missing for $Y_i$ occur for 10 subjects different to the 10 subjects for which variable $Y_j$ is missing and different from all subjects with missing values in any other variables rather than $Y_i$. If that happens, there would be 100 (10*10) subjects, each with only one missing value. So there would be no complete cases, subsample of 0 subjects.

**3**

**a)** $b = 0$ and so missingness in $Y_2$ depends only on the completely observed variable $Y_1$ and a random component ($Z_3$). However, $Y_1$ and $Y_2$ are correlated through $Z_1$. Small values of $Y_1$, which make more probable missingness in $Y_2$ (because $a(Y_1 - 1)$ becomes negative), are due to negative values in $Z_1$. If $Z_1$ is negative, it is more probable for $Y_2$ to be smaller (because $2Z_1$ is negative). And so, cases in which $Y_2$ tends to be smaller relate to cases in which its value tends to be missing.

Looking at the distributios we can see the effect of the relation between the value of $Y_2$ and missingness. Since observations with higher probability of $Y_2$ being small are related to higher probability of the value

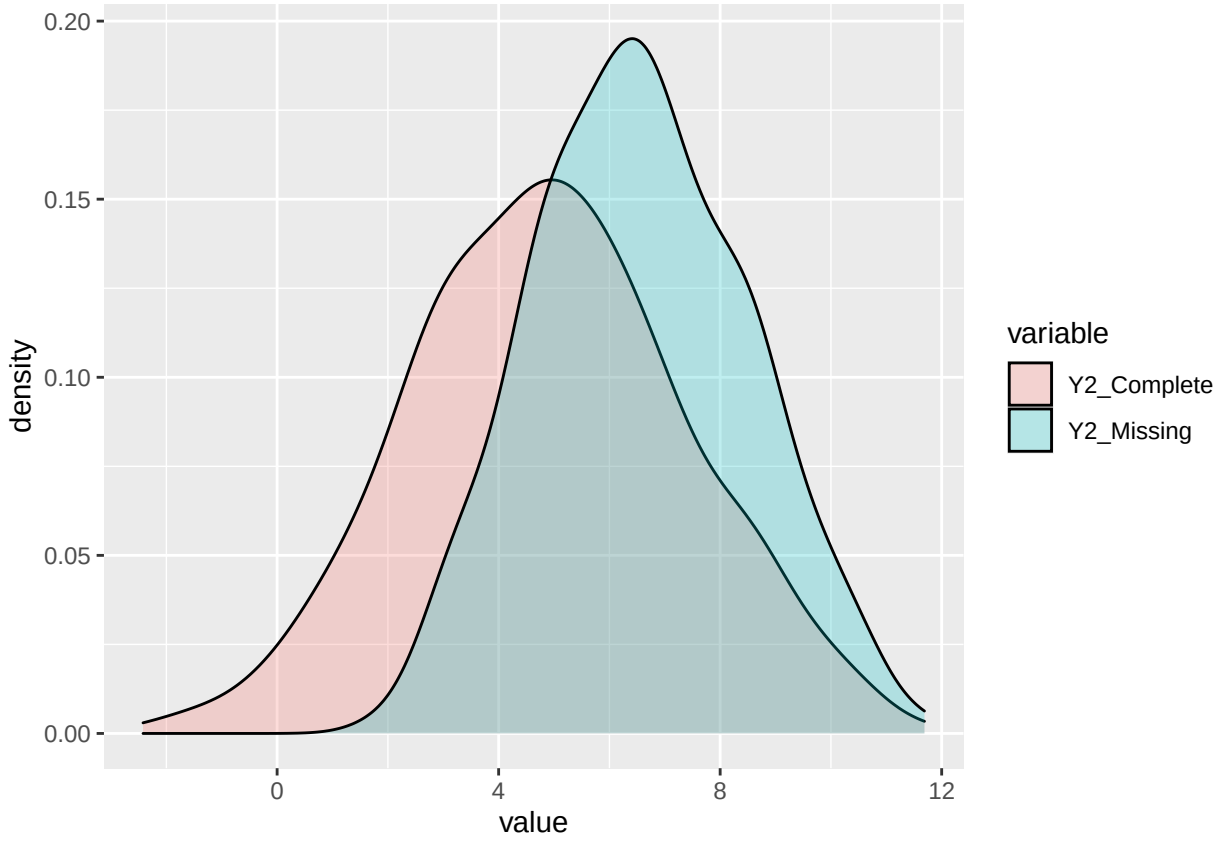
being missing, we can see that the distribution for Y2_Missing is to the right of that for Y2_Complete. For that reason I have decided to classify it as MNAR and not MAR (although MAR would still be ok).

```
library(reshape)
library(ggplot2)
set.seed(29)
mu <- 0; sigma <- 1; a<-2; b<-0;
z1<-rnorm(500,mu,sigma)
z2<-rnorm(500,mu,sigma)
z3<-rnorm(500,mu,sigma)
y1<-1+z1
y2_C<-5+2*z1+z2
y2_M<-ifelse(a*(y1-1)+b*(y2_C-5)+z3 < 0,NA,y2_C)
Disp<- data.frame(Y2_Complete=y2_C,Y2_Missing=y2_M)
Disp<-melt(Disp)
```

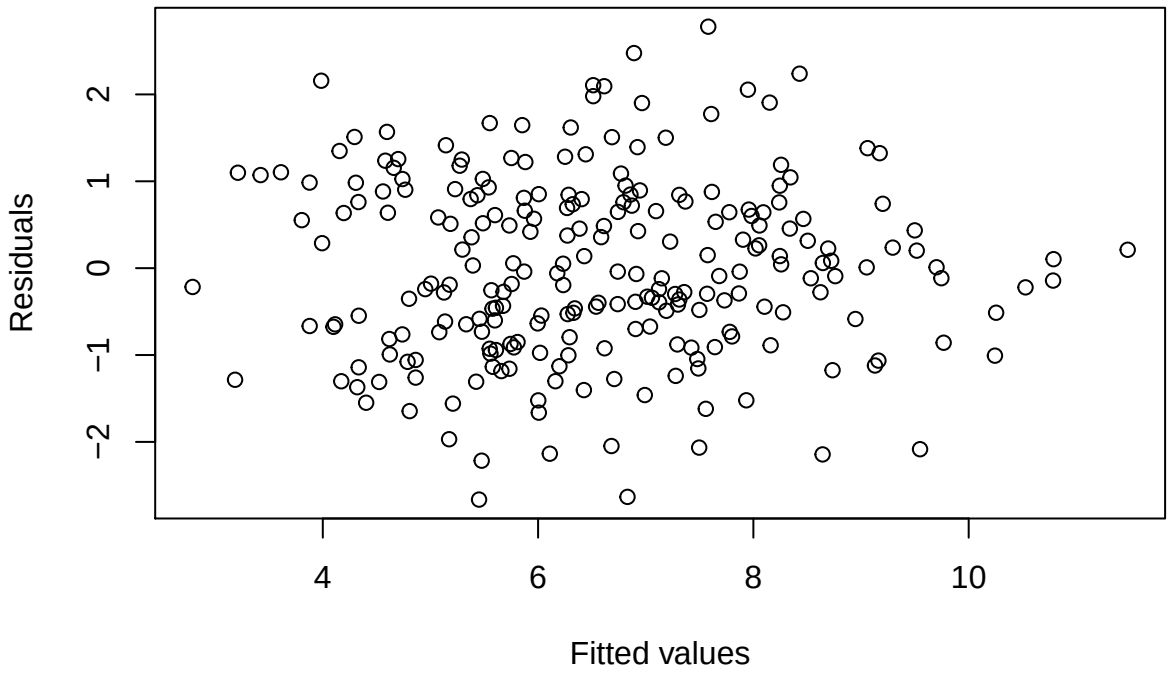
```
## Using  as id variables
```

```
ggplot(Disp,aes(x=value, fill=variable)) +
  geom_density(alpha=0.25)
```

```
## Warning: Removed 259 rows containing non-finite values (stat_density).
```

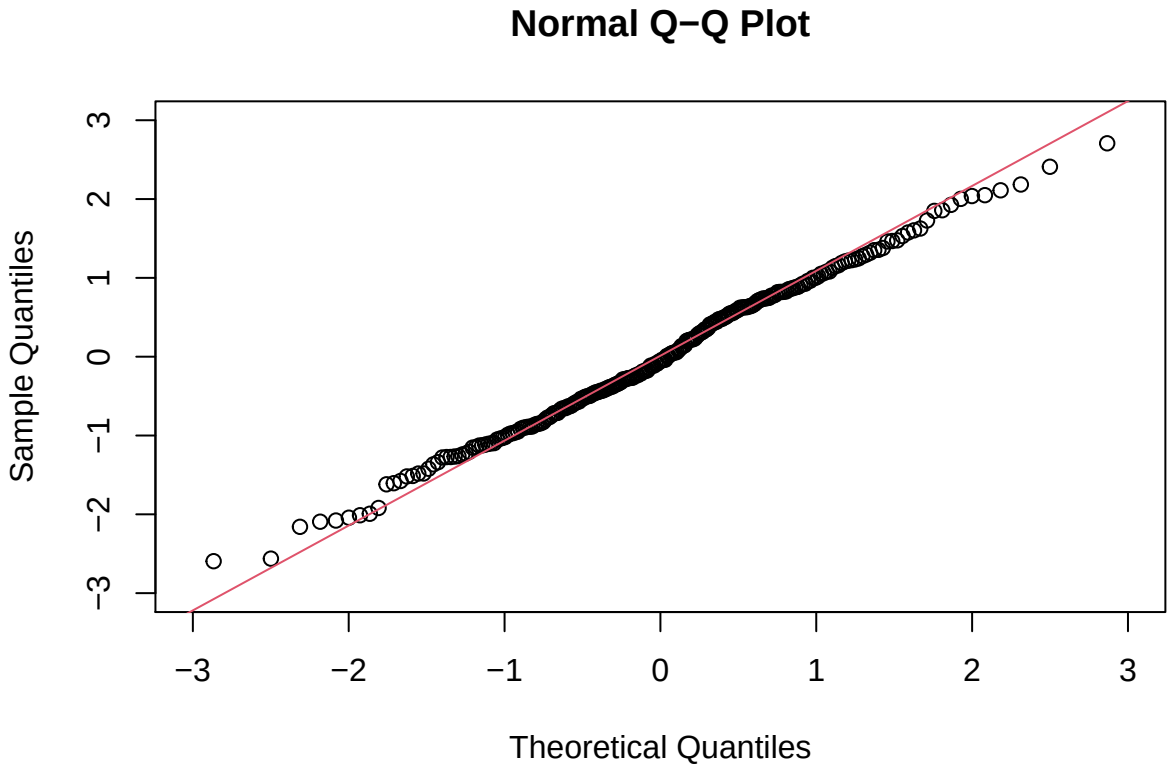



```
fit <- lm(y2_M ~ y1)
plot(fit$fitted.values, residuals(fit), xlab = "Fitted values", ylab = "Residuals")
```




b)

```
qqnorm(rstandard(fit), xlim = c(-3,3), ylim = c(-3,3))
qqline(rstandard(fit),col=2)
```



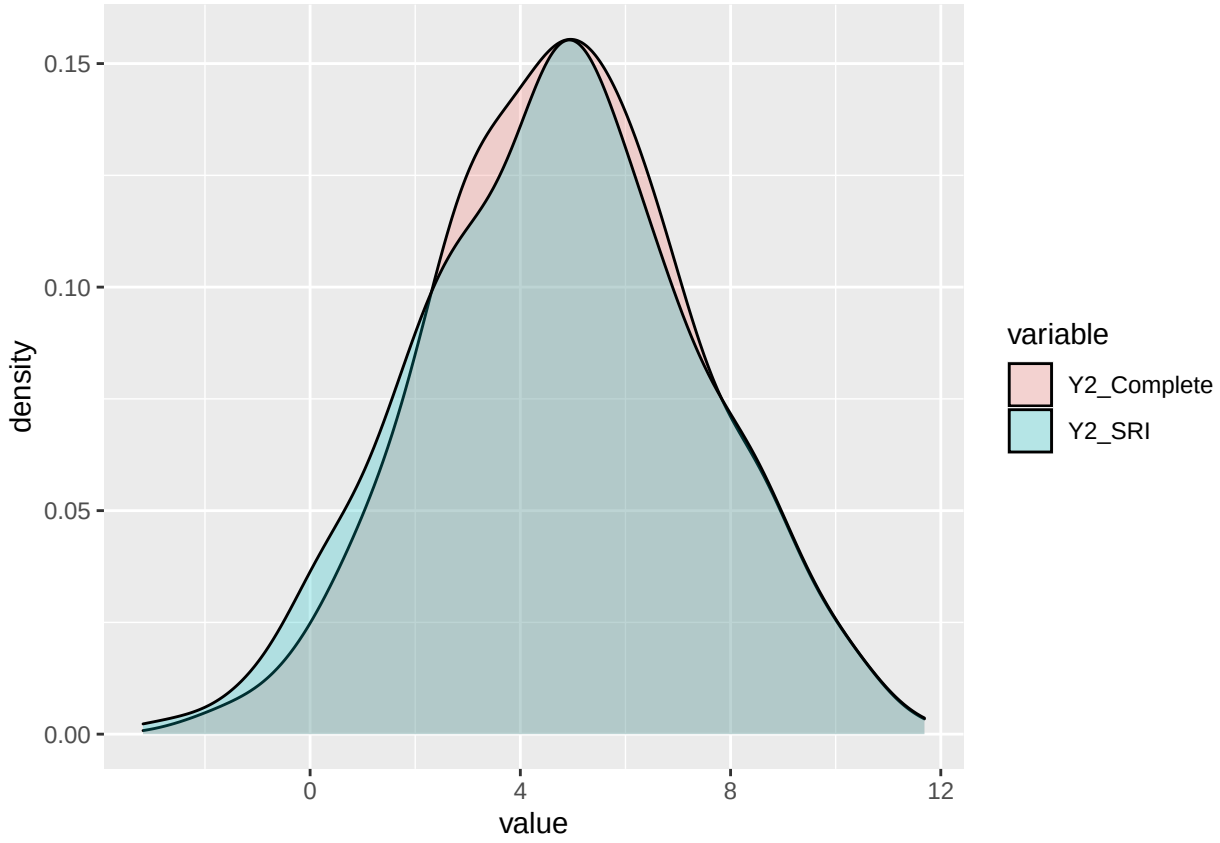

We informally check the linearity and homoscedasticity assumptions looking at residuals and QQ-plot.

```
sigmaest <- sigma(fit)
pred <- fit$coefficients[1] + y1*fit$coefficients[2] + rnorm(500, 0, sigmaest)
y2_SRI<-ifelse(is.na(y2_M)==TRUE,pred,y2_M)
```

```
Disp2<- data.frame(Y2_Complete=y2_C,Y2_SRI=y2_SRI)
Disp2<-melt(Disp2)
```

```
## Using  as id variables
```

```
ggplot(Disp2,aes(x=value, fill=variable)) +
  geom_density(alpha=0.25)
```




The resemble between both distributions is remarkable. After imputing $Y_2$ missing values with SRI, the distribution is around the actual mean for complete values. It probably yielded such good results due to the fact that (as stated previously) $Y_2$ and $Y_1$ are highly correlated. We see a bit more of variation for smaller values, which makes sense since most of the imputations were needed for smaller values.

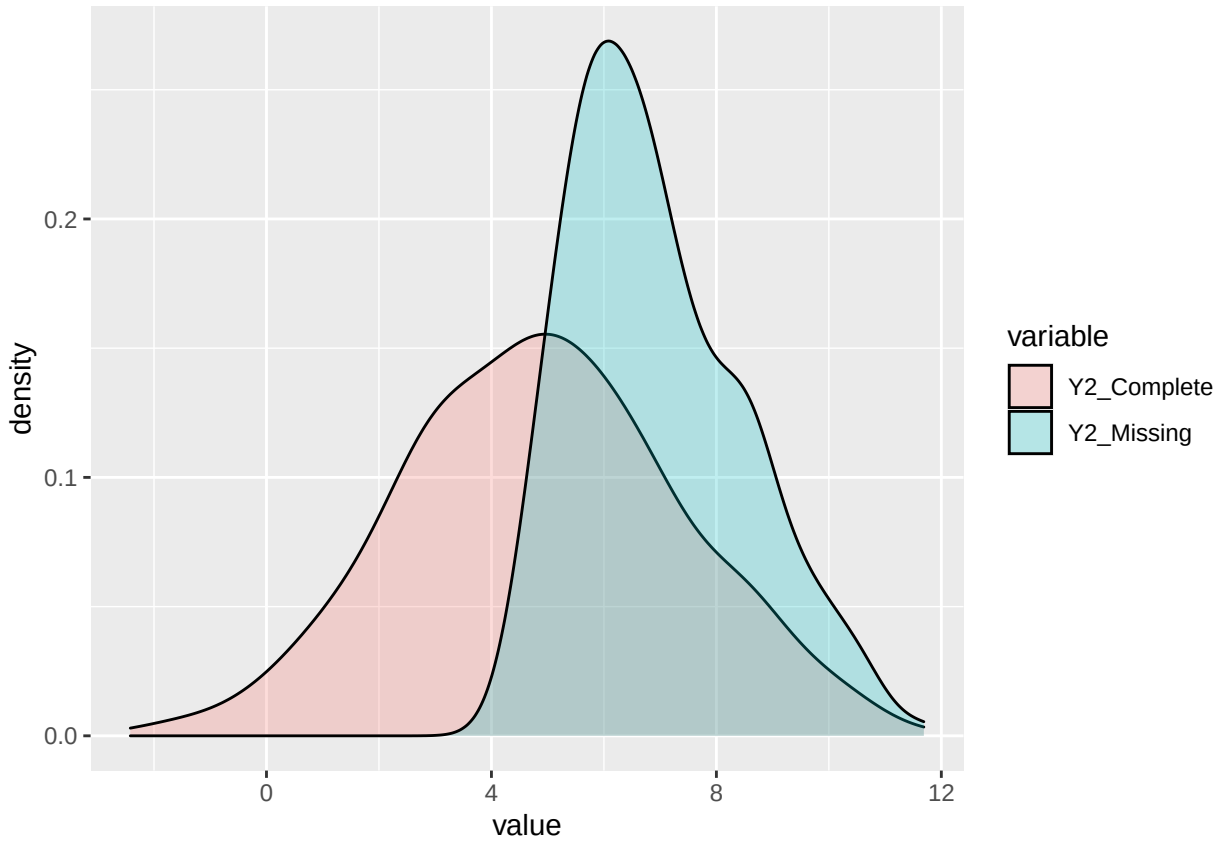
**c)** In this case $b = 2$ and so missingness in $Y_2$ depends only on the actual value of $Y_2$ and a random component ($Z_3$). It is clear that it is MNAR. Missingness occurs almost entirely for small ($> 5$) values of $Y_2$ (because $(Y_2 - 5)$ becomes negative). That is why the missing distribution drops rapidly for values smaller than 5 and is not symmetric.

```
a<-0; b<-2;
y2_M2<-ifelse(a*(y1-1)+b*(y2_C-5)+z3 < 0,NA,y2_C)
Disp3<- data.frame(Y2_Complete=y2_C,Y2_Missing=y2_M2)
Disp3<-melt(Disp3)
```

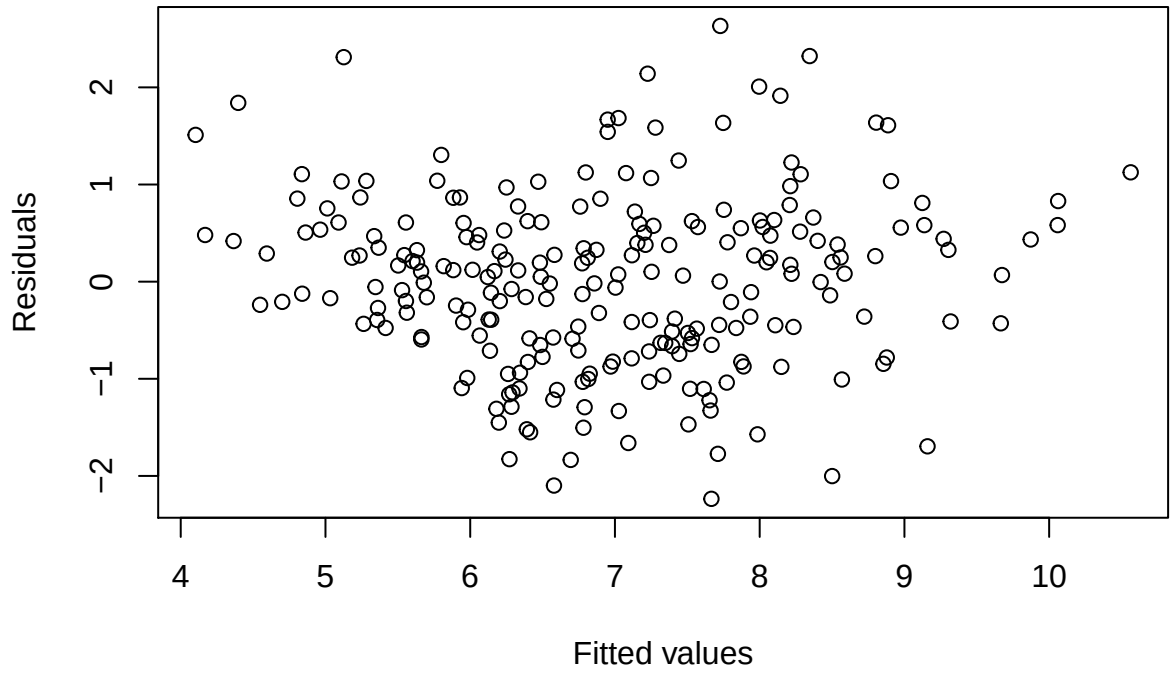
```
## Using  as id variables
```

```
ggplot(Disp3,aes(x=value, fill=variable)) +
  geom_density(alpha=0.25)
```

```
## Warning: Removed 265 rows containing non-finite values (stat_density).
```

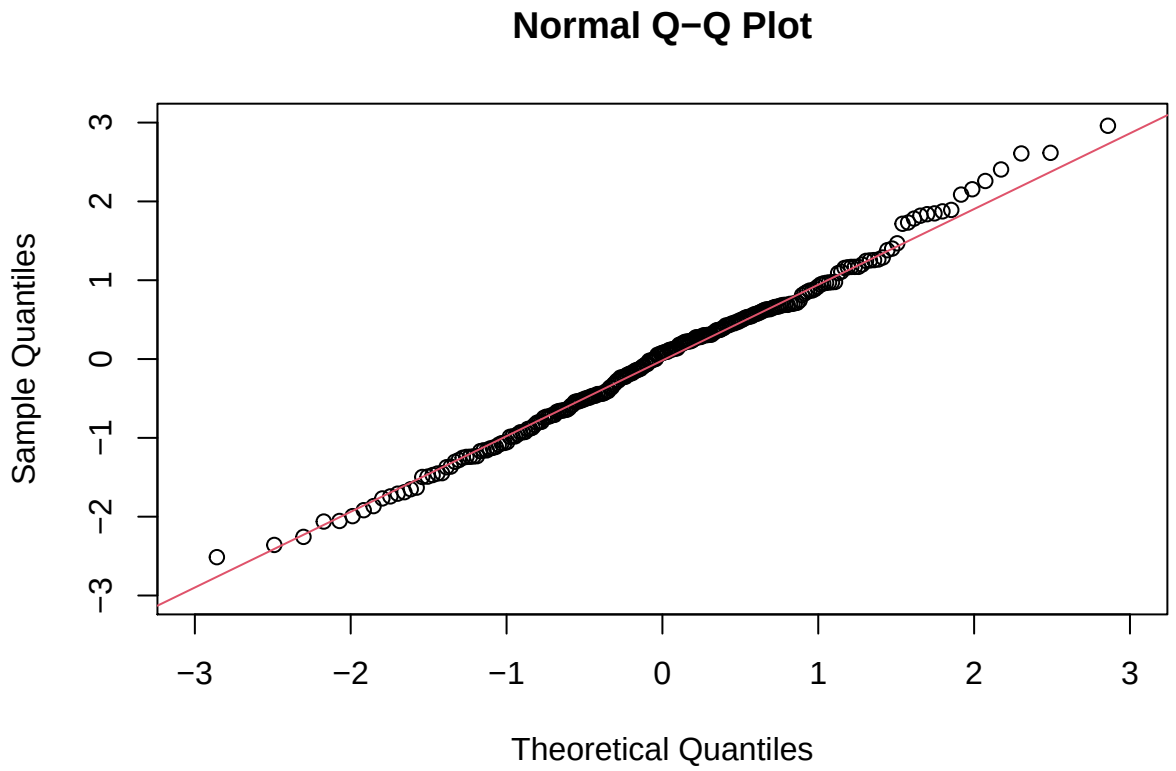



```
fit2 <- lm(y2_M2 ~ y1)
plot(fit2$fitted.values, residuals(fit2), xlab = "Fitted values", ylab = "Residuals")
```




d)

```
qqnorm(rstandard(fit2), xlim = c(-3,3), ylim = c(-3,3))
qqline(rstandard(fit2),col=2)
```

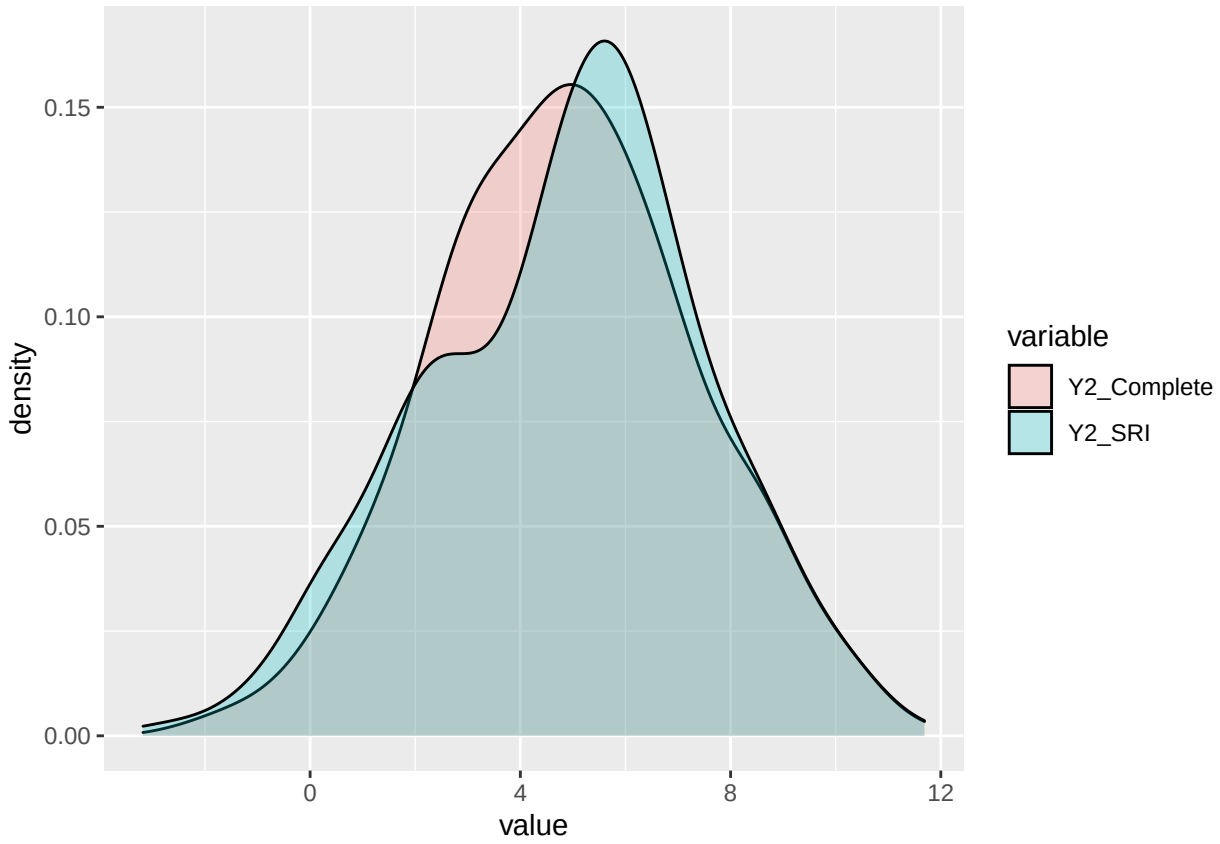



We informally check the linearity and homoscedasticity assumptions looking at residuals and QQ-plot.

```
sigmaest2 <- sigma(fit2)
pred2 <- fit2$coefficients[1] + y1*fit2$coefficients[2] + rnorm(500, 0, sigmaest2)
y2_SRI2<-ifelse(is.na(y2_M2)==TRUE,pred,y2_M2)
Disp4<- data.frame(Y2_Complete=y2_C,Y2_SRI=y2_SRI2)
Disp4<-melt(Disp4)
```

```
## Using  as id variables
```

```
ggplot(Disp4,aes(x=value, fill=variable)) +
  geom_density(alpha=0.25)
```




Once again, due to the high correlation between $Y_1$ and $Y_2$, SRI does a great job estimating the actual marginal distribution for $Y_2$. The distributions are fairly similar and pretty close to each other. Much of the symmetry of the complete distribution is restored.

**4**

**a)** Complete Case Analysis

```
load("/cloud/project/databp.Rdata")
ind <- which(is.na(databp$recovtime) == FALSE)
mcc <- mean(databp$recovtime, na.rm = TRUE)
secc <- sd(databp$recovtime, na.rm = TRUE)/sqrt(length(ind))
CorbDose<-cor(databp$logdose[ind],databp$recovtime[ind])
CorbBP<-cor(databp$bloodp[ind],databp$recovtime[ind])
Output<-data.frame(Mean=mcc,SE=secc,Dose_Correlation=CorbDose,Blood_P_Correlation=CorbBP)
Output
```

```
##       Mean       SE Dose_Correlation Blood_P_Correlation
## 1 19.27273 2.603013        0.2391256         -0.01952862
```

We have the estimate of the mean to be 19.27 with a standard error of 2.6. The correlation with the dose is considerable (0.24). Recovery time doesn't seem to be related to the blood pressure, since their correlations is -0.02.

**b)** Mean Imputation

```
recovtimeMI<-ifelse(is.na(databp$recovtime) == TRUE, mcc, databp$recovtime)
mmi <- mean(recovtimeMI)
semi <- sd(recovtimeMI)/sqrt(length(recovtimeMI))
CorbDoseMI<-cor(databp$logdose,recovtimeMI)
CorbBPMI<-cor(databp$bloodp,recovtimeMI)
Output2<-data.frame(Mean=mmi,SE=semi,Dose_Correlation=CorbDoseMI,Blood_P_Correlation=CorbBPMI)
Output2
```

```
##       Mean       SE Dose_Correlation Blood_P_Correlation
## 1 19.27273 2.284135        0.2150612         -0.01934126
```

The inputted values were:

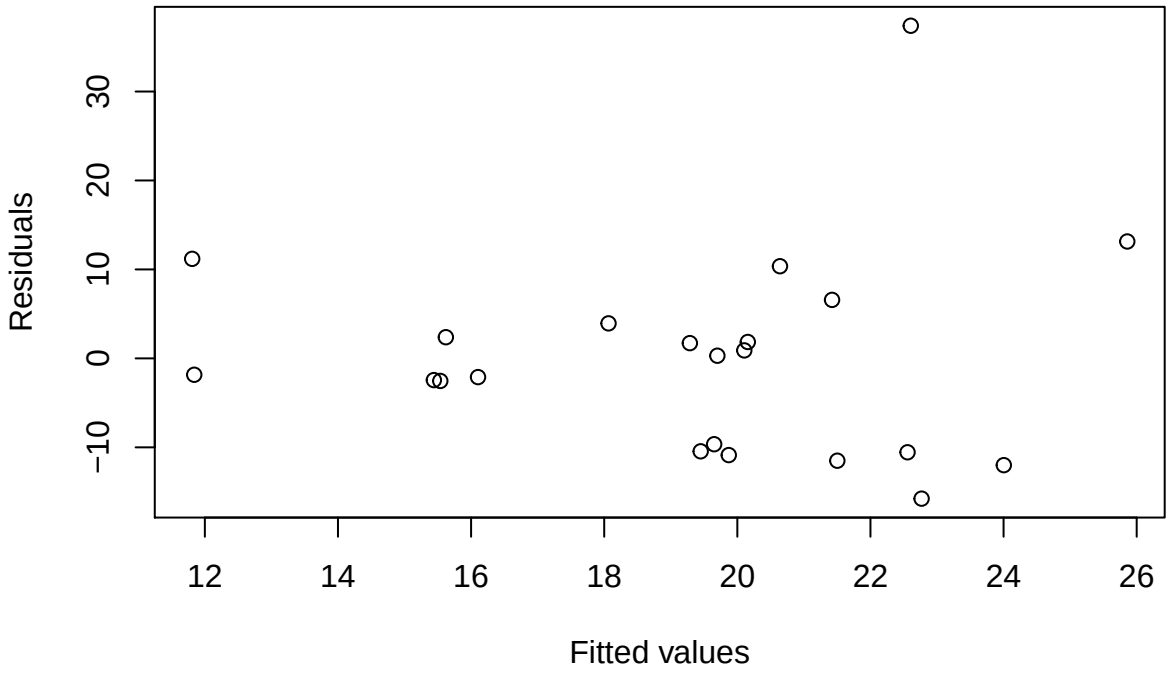
```
recovtimeMI[is.na(databp$recovtime)==TRUE]
```

```
## [1] 19.27273 19.27273 19.27273
```

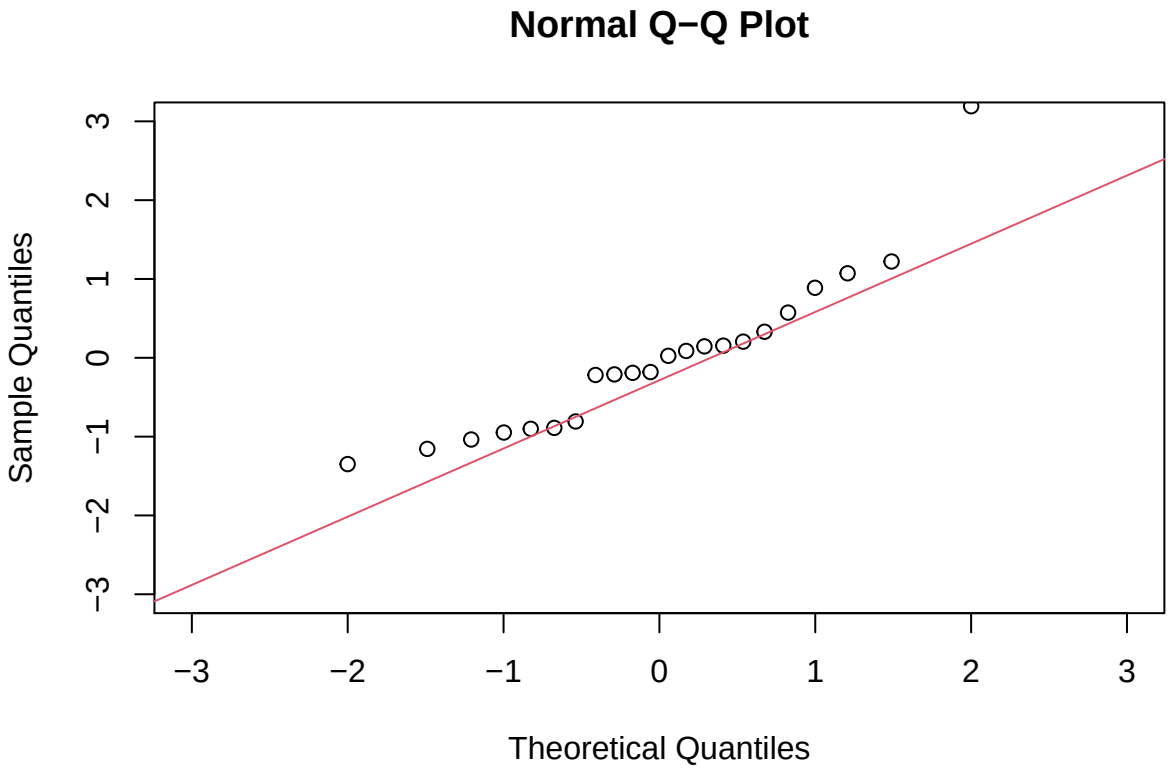
Here we have the same estimate of the mean as before (19.27) given that we only inputted that same mean to the missing observations. The standard error is lower because of that (2.28). The correlation with the dose is a bit smaller (0.22). And the correlation with the blood pressure remains insignificant (-0.02).

**c)** Mean Regression Imputation

```
fitRecovtime<-lm(recovtime ~ logdose + bloodp, data = databp)
plot(fitRecovtime$fitted.values, residuals(fitRecovtime), xlab = "Fitted values", ylab = "Residuals")
```




```
qqnorm(rstandard(fitRecovtime), xlim = c(-3,3), ylim = c(-3,3))
qqline(rstandard(fitRecovtime),col=2)
```




We informally check the linearity and homoscedasticity assumptions looking at residuals and QQ-plot. There are few observations so is difficult to know for sure. But, looking at the QQ-plot the regression is probably not the best way to go.

```
predri <- predict(fitRecovtime,newdata=databp)
recovtimeMRI<-ifelse(is.na(databp$recovtime) == TRUE, predri, databp$recovtime)
mri <- mean(recovtimeMRI)
semri <- sd(recovtimeMRI)/sqrt(length(recovtimeMRI))
CorbDoseMRI<-cor(databp$logdose,recovtimeMRI)
CorbBPMRI<-cor(databp$bloodp,recovtimeMRI)
Output3<-data.frame(Mean=mri,SE=semri,Dose_Correlation=CorbDoseMRI,Blood_P_Correlation=CorbBPMRI)
Output3
```

```
##       Mean       SE Dose_Correlation Blood_P_Correlation
## 1 19.44428 2.312845        0.2801835          -0.0111364
```

The inputted values were:

```
predri[is.na(databp$recovtime)==TRUE]
```

```
##        4       10       22
## 14.26254 21.51562 26.32896
```

Now we have an estimate for the mean of 19.44. Since the inputted values are in a straight line, the standard error is lower than for the CCA (2.31). For the same reason, the correlation with the dose is higher (0.28). And the correlation with the blood pressure remains insignificant (-0.01).

**d)** Stochastic Regression Imputation

In this case we need to be careful, because due to the stochastic component we could get a negative value for the recovery time. And of course, a negative recovery time makes no sense. With the coefficients obtained, the sigma from the regression and the seed set we have no negative value and no further adjustment is needed.

```
set.seed(29)
predsri <- predict(fitRecovtime,newdata=databp)+ rnorm(length(databp$recovtime),0,sigma(fitRecovtime))
recovtimeSRI<-ifelse(is.na(databp$recovtime) == TRUE, predsri, databp$recovtime)
msri <- mean(recovtimeSRI)
sesri <- sd(recovtimeSRI)/sqrt(length(recovtimeSRI))
CorbDoseSRI<-cor(databp$logdose,recovtimeSRI)
CorbBPSRI<-cor(databp$bloodp,recovtimeSRI)
Output4<-data.frame(Mean=msri,SE=sesri,Dose_Correlation=CorbDoseSRI,Blood_P_Correlation=CorbBPSRI)
Output4
```

```
##       Mean       SE Dose_Correlation Blood_P_Correlation
## 1 21.53037 2.724354        0.3604275          0.02389744
```

The inputted values were:

```
predsri[is.na(databp$recovtime)==TRUE]
```

```
##        4       10       22
## 25.86518 35.13001 53.26402
```

Now we have an estimate for the mean of 21.53, which is higher than all others. Due to the stochastic component, the standard error is higher than (2.72). For the same reason as for RI, the correlation with the dose is higher (0.36). The correlation with the blood pressure remains insignificant (0.02).

**e)** Predictive Mean Matching

```
recovtimePMM<-databp$recovtime
for (i in which(is.na(recovtimePMM) == TRUE)){
  ini_pred<-predri[i]
  dif<-ini_pred^2
  for (j in which(is.na(databp$recovtime) == FALSE)){
    if((ini_pred-predri[j])^2<dif){dif<-(ini_pred-predri[j])^2;recovtimePMM[i]<-recovtimePMM[j]}
  }
}
mpmm <- mean(recovtimePMM)
sepmm <- sd(recovtimePMM)/sqrt(length(recovtimePMM))
CorbDosePMM<-cor(databp$logdose,recovtimePMM)
CorbBPPMM<-cor(databp$bloodp,recovtimePMM)
Output5<-data.frame(Mean=mpmm,SE=sepmm,Dose_Correlation=CorbDosePMM,Blood_P_Correlation=CorbBPPMM)
Output5
```

```
##    Mean       SE Dose_Correlation Blood_P_Correlation
## 1 19.44 2.464467        0.3037945         -0.03208685
```

The inputted values were:

```
recovtimePMM[is.na(databp$recovtime)==TRUE]
```

```
## [1] 13 10 39
```

With this method we have an estimate for the mean of 19.44. Since other observed values are used, the standard error is lower than for other methods (2.46). The correlation with the dose is 0.3. The correlation with the blood pressure remains insignificant (-0.03).

**f)** It is less dependent on linearity assumptions. It is better at dealing with issues such as heteroscedasticity because it uses real values from other subjects and that helps to account for variance changes. Finally, using only real values avoids problems like negative values.

The main problem that I can think of for this method is that ita can happen that many predictions for missing values are closest to the same predicted value of the donor. That could cause for the same value to be inputted multiple times. So it may lead to biased estimations and of course to underestimation of the standard errors.

Github Repository:

https://github.com/Arturo-Esquivel/Incomplete-Data-Analysis