Some imputation techniques for missing Data

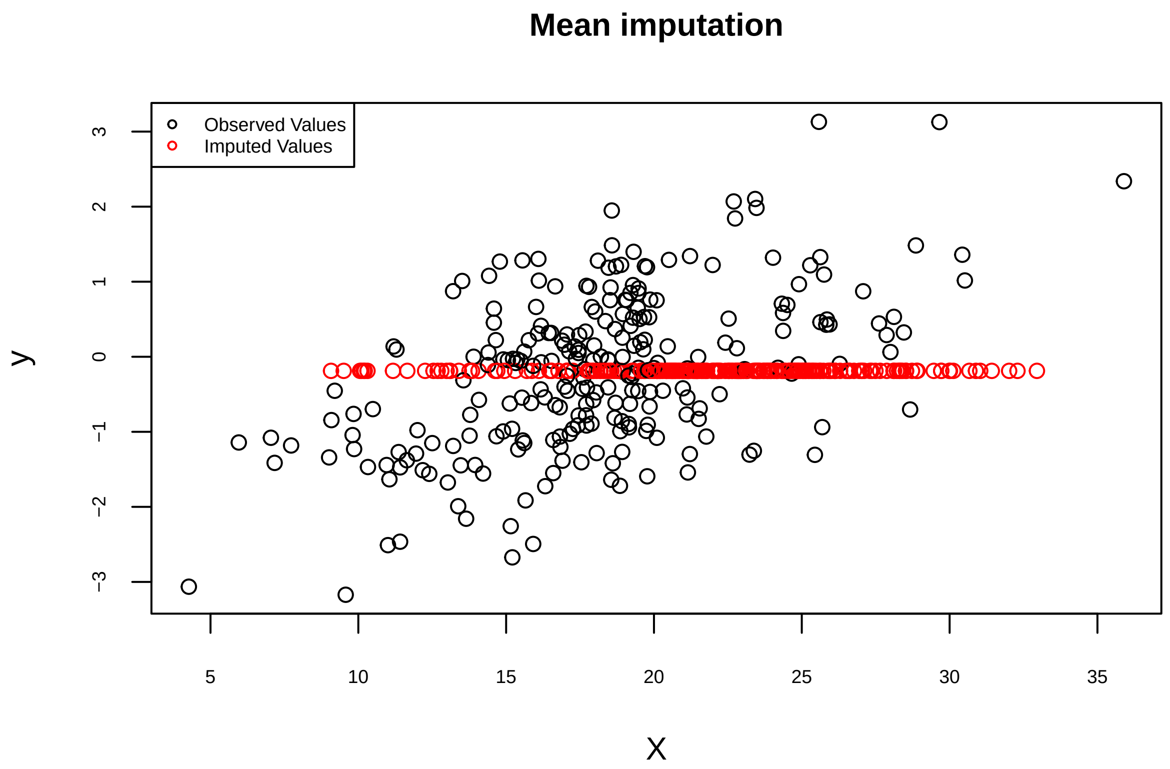
Imane Hadik

2022-08-20

```
library(faux)
dat<- rnorm_multi(n = 500,
                  mu = c(0, 20),
                  sd = c(1, 5),
                  r = 0.5,
                  varnames = c("y", "X"),
                  empirical = FALSE)
library(missMethods)
data_mar <- delete_MAR_1_to_x(dat,0.5,"y","X",3)# to  generate MAR data
```

MEAN imputation

```
#windows(4,4)
mean<-mean(data_mar$y, na.rm = TRUE)
mean_imp<-data_mar
mean_imp$y[is.na(data_mar$y)] <- mean
##plot
plot(mean_imp$X,mean_imp$y,
     xlab='X',ylab='y',main = "Mean imputation",
     col=ifelse(mean_imp$y==mean,'red','black'),
     cex.lab=1,cex.axis=0.6,cex.main=1)
##legend
legend("topleft",                                       # Legend
       c("Observed Values", "Imputed Values"),
       pch = c(1, 1, NA),
       lty = c(NA, NA, 1),
       col = c("black", "red", "#1b98e0"),cex=0.55555)
```



```
##calculate mean, sd, cor,
mean1=mean(mean_imp$y)
sd1=sd(mean_imp$y)
cor1=cor(mean_imp$X,mean_imp$y)
cat(" ","mean=",mean1,"sd=", sd1,"cor=",cor1) # nolint
```

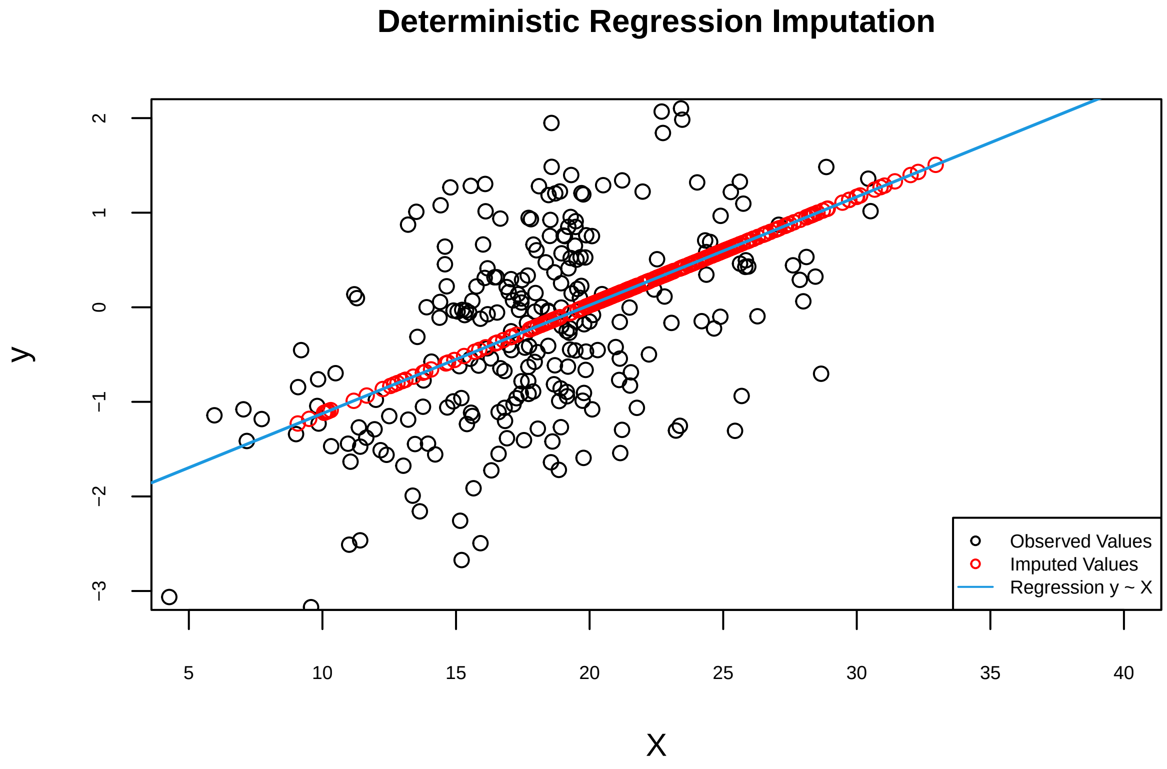
```
##   mean= -0.1878109 sd= 0.7362492 cor= 0.343086
```

Deterministic Regression imputation

```
library(mice)
imp <- mice(data_mar, method = "norm.predict", m = 1) # Impute data
```

```
##
##  iter imp variable
##   1   1  y
##   2   1  y
##   3   1  y
##   4   1  y
##   5   1  y
```

```
data_det <- complete(imp) # Store data
##plot
#windows(4,4)
plot(data_mar$X[!is.na(data_mar$y)], data_mar$y[!is.na(data_mar$y)],     # Plot of observed values
     xlim = c(5, 40), ylim = c(-3, 2),
     main = "Deterministic Regression Imputation",
     xlab = "X", ylab = "y",cex.lab=1,cex.axis=0.6,cex.main=1)
points(data_mar$X[is.na(data_mar$y)], data_det$y[is.na(data_mar$y)],     # Plot of imputed missing valu
       col = "red")
abline(lm(data_det$y ~ data_det$X, data_det), col = "#1b98e0", lwd = 1.5)
##legend
legend("bottomright",                                        # Legend
       c("Observed Values", "Imputed Values", "Regression y ~ X"),
       pch = c(1, 1, NA),
       lty = c(NA, NA, 1),
       col = c("black", "red", "#1b98e0"),cex=0.6)
```



```
##calculate mean, sd, cor
mean2=mean(data_det$y)
sd2=sd(data_det$y)
cor2=cor(data_det$X,data_det$y)
cat(" ","mean=",mean2,"sd=", sd2,"cor=",cor2) # nolint
```

```
##   mean= 0.032745 sd= 0.8565415 cor= 0.6793329
```

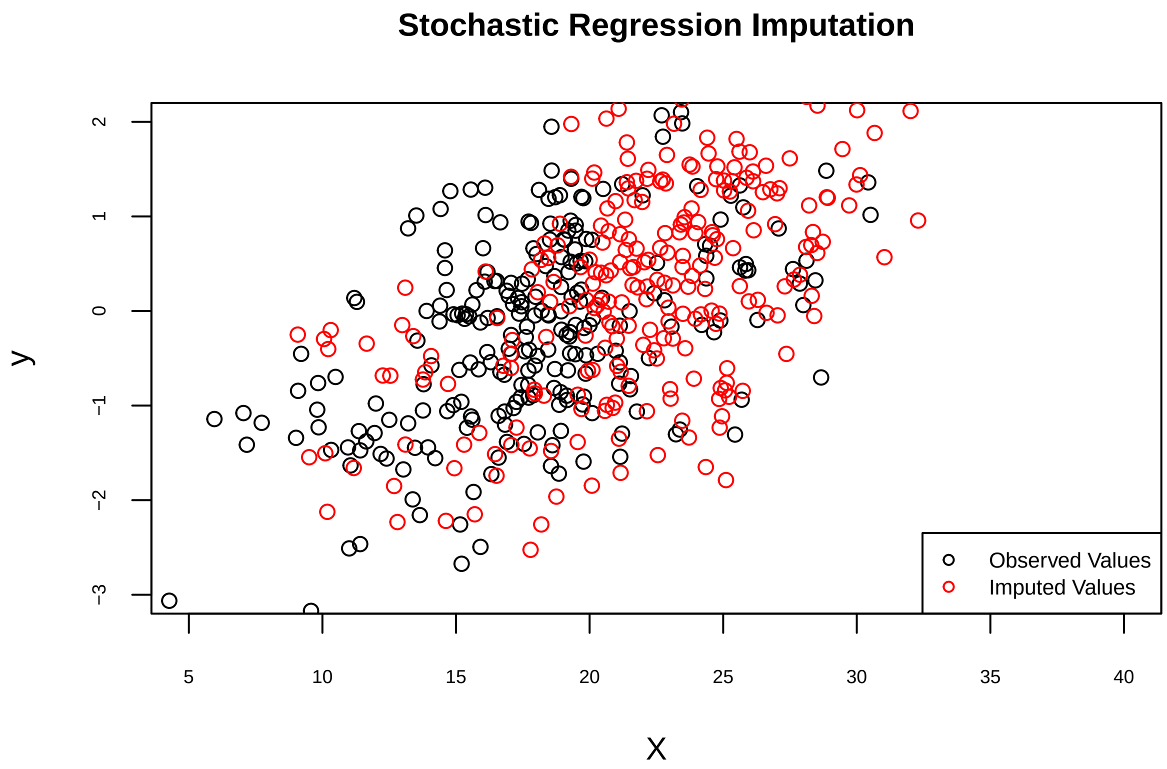
```
# another way to do deterministic regr
#lm(data_mar$y ~ data_mar$X, data_mar)
#datatry=data_mar
#for(i in 1:nrow(datatry))
#{
  #if(is.na(datatry$y[i]))
  #{
    #datatry$y[i] = -1.94762 + 0.09749 *datatry$X[i] # lm coeff
  #}
#}
```

Stochastic regression imputation

```
imp <- mice(data_mar, method = "norm.nob", m = 1) # Impute data
```

```
##
##  iter imp variable
##   1   1  y
##   2   1  y
##   3   1  y
##   4   1  y
##   5   1  y
```

```
data_sto <- complete(imp) # Store data
##plot
#windows(4,4)
plot(data_mar$X[!is.na(data_mar$y)], data_mar$y[!is.na(data_mar$y)],     # Plot of observed values
     xlim = c(5, 40), ylim = c(-3, 2),
     main = "Stochastic Regression Imputation",
     xlab = "X", ylab = "y",cex.axis=0.6,cex.lab=1,cex.main=1)
points(data_mar$X[is.na(data_mar$y)], data_sto$y[is.na(data_mar$y)],     # Plot of imputed missing valu
       col = "red")
##legend
legend("bottomright",
       c("Observed Values", "Imputed Values"),
       pch = c(1, 1, NA),
       lty = c(NA, NA, 1),
       col = c("black", "red"),
       cex=0.7)
```



Stochastic Regression Imputation

```
##calculate mean, sd, cor, IC mu_y put it in a table(or later in latex)
mean3=mean(data_sto$y)
sd3=sd(data_sto$y)
cor3=cor(data_sto$X,data_sto$y)
cat(" ","mean=",mean3,"sd=", sd3,"cor=",cor3) # nolint
```

```
##   mean= 0.0337591 sd= 1.110905 cor= 0.5659903
```

Hot Deck imputation

```
library(missForest)
library(mice)
library(hot.deck)
y_mis <- prodNA(data.frame(dat$y), noNA = 0.5) # to generate MCAR values
data_mcar<-cbind(dat$X,y_mis)
#saveRDS(df,file = "df.rds") #to save the dataset as rds file
res <- hot.deck(data_mcar, m = 6, method = "best.cell")
#let's look at the 2nd imputed dataset
#res$data[2]
#saveRDS(res,file = 'HD.rds') # to save the hot deck results
```