Data clean-up and data curation

Núria Jolis Orriols

2 de juny, 2022

Contents


```
libraries <- c("readr", "tidyverse", "naniar" ,"ggplot2", "car", "dplyr", "reshape2", "patchwork", "rsta
check.libraries <- is.element(libraries, installed.packages()[, 1])==FALSE
libraries.to.install <- libraries[check.libraries]
if (length(libraries.to.install!=0)) {
  install.packages(libraries.to.install)
}

success <- sapply(libraries,require, quietly = FALSE,  character.only = TRUE)
if(length(success) != length(libraries)) {stop("A package failed to return a success in require() functi
```

Step 1 - Obtaining the data

```
data<- read.csv("data/data.Li21.csv", sep=";")
```

Step 2 - Data clean-up and data curation

The dataset consists of 1176 observations and 51 variables, all numeric being 36 of type “double” and 15 of type “integer”.

Dataset modifications:

1. The first two variables are to be discarded. The variable, "group", is removed as it was created by Li 2021 to separate the data for training and testing their models, and the ID is the patient's identification which will not be useful to predict the outcome.
2. A new variable is created to combine systolic and diastolic blood pressure. It is called mean arterial pressure (MAP) and it follows the next equation: MAP = [Systolic + 2*Diastolic]/3.
3. 11 of the variables are numerical binary, they will be converted into factors: "outcome", "gender", "hypertensive", "atrialfibrillation", "CHD.with.no.MI", "diabetes", "deficiencyanemias", "depression", "hyperlipemia", "renal.failure" and "COPD".

```
## 'data.frame':    1176 obs. of  48 variables:
##  $ outcome           : Factor w/ 2 levels "Survivor","Non-survivor": 1 1 1 1 1 1 1 1 1 1 ...
##  $ age               : int  72 75 83 43 75 76 72 83 61 67 ...
##  $ gender            : Factor w/ 2 levels "M","F": 1 2 2 2 2 1 1 2 2 1 ...
##  $ BMI               : num  37.6 NA 26.6 83.3 31.8 ...
##  $ hypertensive      : Factor w/ 2 levels "No","Yes": 1 1 1 1 2 2 2 2 2 2 ...
##  $ atrialfibrillation: Factor w/ 2 levels "No","Yes": 1 1 1 1 1 2 1 2 2 1 ...
##  $ CHD.with.no.MI    : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 1 1 1 ...
##  $ diabetes          : Factor w/ 2 levels "No","Yes": 2 1 1 1 1 1 1 2 2 2 ...
##  $ deficiencyanemias : Factor w/ 2 levels "No","Yes": 2 2 2 1 2 2 1 2 1 1 ...
##  $ depression        : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 1 1 1 ...
##  $ hyperlipemia      : Factor w/ 2 levels "No","Yes": 2 1 1 1 1 2 2 1 1 1 ...
##  $ renal.failure     : Factor w/ 2 levels "No","Yes": 2 1 2 1 2 2 2 1 2 1 ...
##  $ COPD              : Factor w/ 2 levels "No","Yes": 1 2 1 1 2 2 2 1 1 1 ...
##  $ heart.rate        : num  68.8 101.4 72.3 94.5 67.9 ...
##  $ respiratory.rate  : num  16.6 20.9 23.6 21.9 21.4 ...
##  $ temperature       : num  36.7 36.7 36.5 36.3 36.8 ...
##  $ SP.O2             : num  98.4 96.9 95.3 93.8 99.3 ...
##  $ urine.output      : num  2155 1425 2425 8760 4455 ...
##  $ hematocrit        : num  26.3 30.8 27.7 36.6 29.9 ...
##  $ RBC               : num  2.96 3.14 2.62 4.28 3.29 ...
##  $ MCH               : num  28.2 31.1 34.3 26.1 30.7 ...
##  $ MCHC              : num  31.5 31.7 31.3 30.4 33.7 ...
##  $ MCV               : num  89.9 98.2 109.8 85.6 91 ...
##  $ RDW               : num  16.2 14.3 23.8 17 16.3 ...
##  $ leucocyte         : num  7.65 12.74 5.48 8.22 8.83 ...
##  $ platelets         : num  305 246 204 216 251 ...
##  $ neutrophils       : num  74.7 NA 68.1 81.8 NA ...
##  $ basophils         : num  0.4 NA 0.55 0.15 NA 0.3 0.2 NA 0.55 NA ...
##  $ lymphocyte        : num  13.3 NA 24.5 14.5 NA ...
##  $ PT                : num  10.6 NA 11.3 27.1 NA ...
##  $ INR               : num  1 NA 0.95 2.67 NA ...
##  $ NT.proBNP         : num  1956 2384 4081 668 30802 ...
##  $ creatine.kinase   : num  148 60.6 16 85 111.7 ...
##  $ creatinine        : num  1.958 1.122 1.871 0.586 1.95 ...
##  $ urea.nitrogen     : num  50 20.3 33.9 15.3 43 ...
##  $ glucose           : num  115 148 149 128 146 ...
##  $ blood.potassium   : num  4.82 4.45 5.83 4.39 4.78 ...
##  $ blood.sodium      : num  139 139 141 138 137 ...
##  $ blood.calcium     : num  7.46 8.16 8.27 9.48 8.73 ...
##  $ chloride          : num  109.2 98.4 105.9 92.1 104.5 ...
##  $ anion.gap         : num  13.2 11.4 10 12.4 15.2 ...
##  $ magnesium.ion     : num  2.62 1.89 2.16 1.94 1.65 ...
```

```
##  $ pH                : num  7.23 7.22 7.27 7.37 7.25 ...
##  $ bicarbonate       : num  21.2 33.4 30.6 38.6 22 ...
##  $ lactic.acid       : num  0.5 0.5 0.5 0.6 0.6 ...
##  $ PCO2              : num  40 78 71.5 75 50 ...
##  $ EF                : int  55 55 35 55 55 35 55 75 50 55 ...
##  $ m.a.p             : num  97.5 90 86 90.9 90.9 ...
```

Finally, we got a dataset of 1176 observations and 48 variables being 39 numeric and 11 factors. Outcome is the response variable whose behavior shall be modeled and the 49 variables left are considered to be candidate predictors.

In order to perform models for outcome prediction an exploratory data analysis is going to be performed first.

Checking for missing values

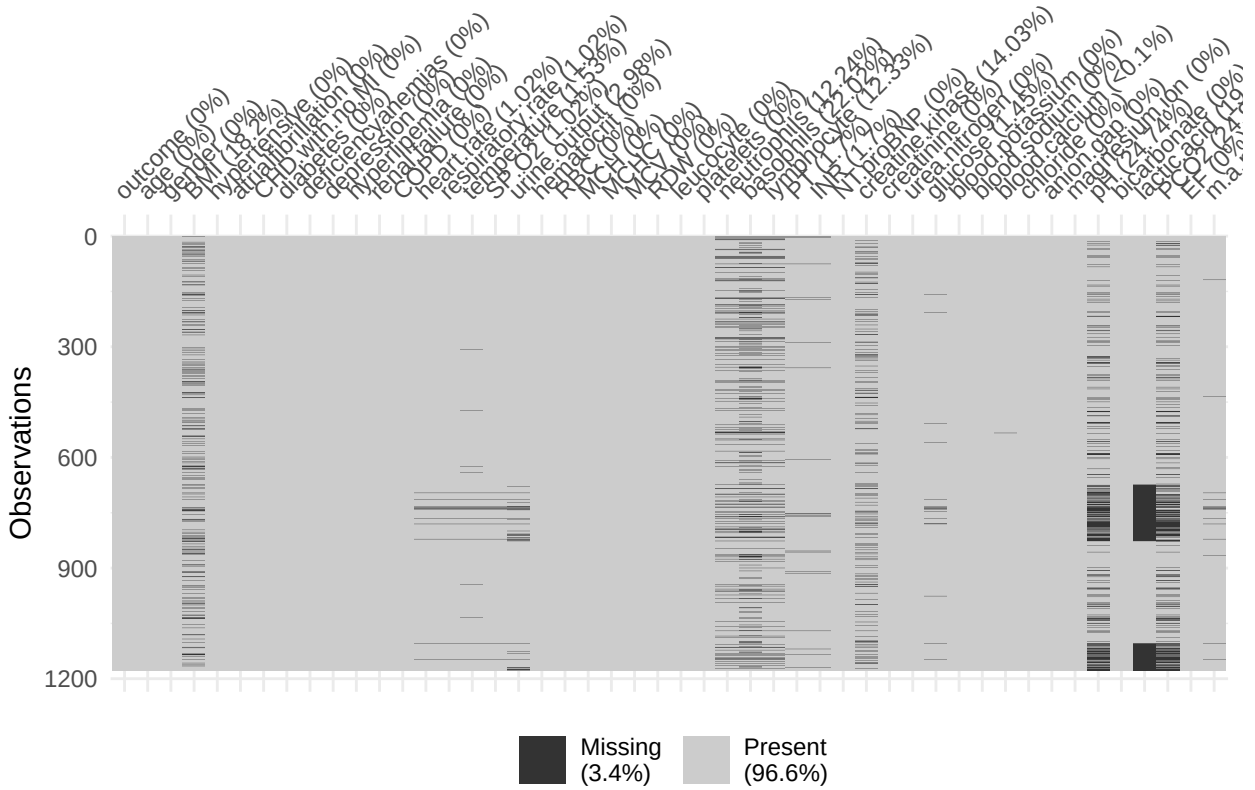
```
#Determine the total number of missing values
sum(is.na(data))
```

```
## [1] 1901
```

```
#In percentage and graphic visualization
round(mean(is.na(data))*100, 1)
```

```
## [1] 3.4
```

```
vis_miss(data)
```



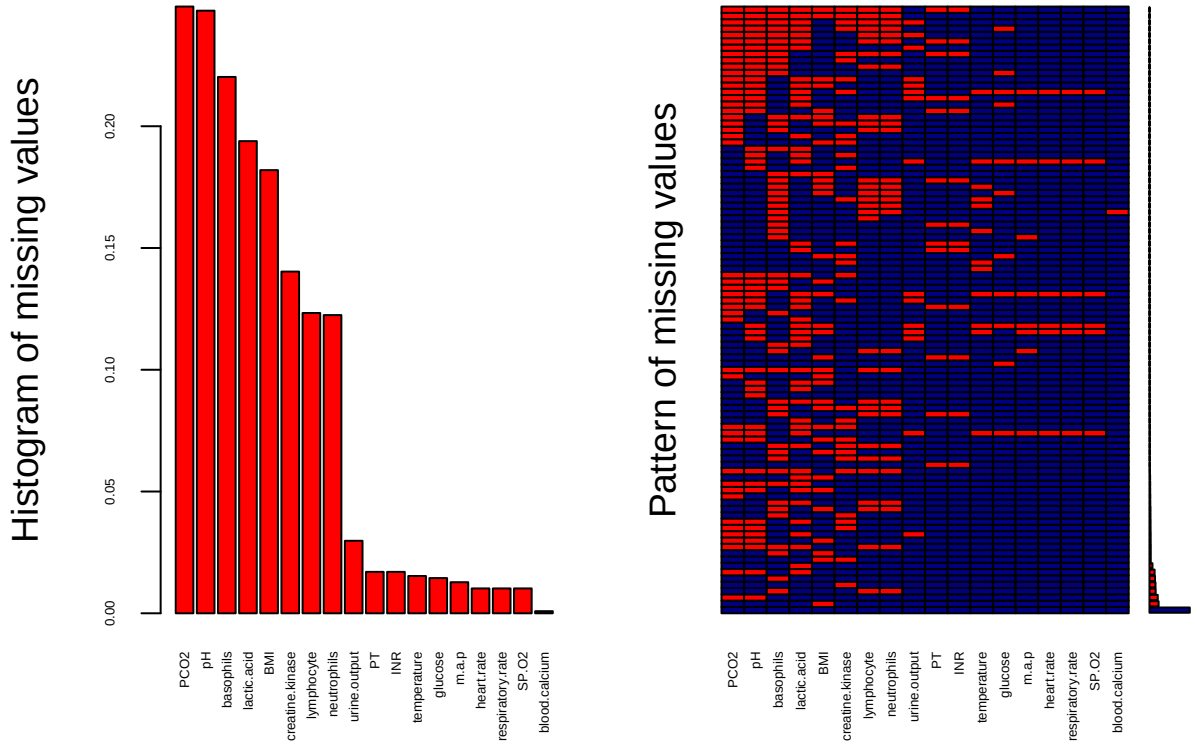
```r
#Variables with % of missing values
x1<-apply(is.na(data), 2, mean)
x2<-round(x1[x1>0], 3)*100 #%NA per variable
x2
```

```
##              BMI       heart.rate respiratory.rate      temperature
##             18.2              1.0              1.0              1.5
##           SP.O2     urine.output      neutrophils       basophils
##             1.0              3.0             12.2             22.0
##       lymphocyte               PT              INR  creatine.kinase
##             12.3              1.7              1.7             14.0
##          glucose    blood.calcium               pH      lactic.acid
##             1.4              0.1             24.7             19.4
##             PCO2            m.a.p
##             24.9              1.3
```

The 3.4% of the values are missing and are concentrated in 21 of the 50 variables. Of those 21 variables, 8 of them present more than 10% of missing values: "basophils", "creatine.kinase", "lactic.acid", "BMI", "neutrophils", "lymphocyte", "pH" and "PCO2".

```r
#To analyze the patron of the missing values we create a dataset with those variables and then we plot
var.na<-dplyr::select(data, c("PCO2", "pH", "basophils", "lactic.acid", "BMI", "creatine.kinase", "lympl

aggr_plot<-aggr(var.na, col=c("navyblue", "red"), numbers=TRUE, labels=names(var.na), ylab=c("Histogram
```




```r
ggsave(filename="results_missingvalues_pattern.png", path="~/Desktop/UOC/TFM/R/Figures", width = 5, hei
```

The histogram on the left side shows the proportion of missing values in each variables. The graphic on the right side shows the pattern of missing values, in navy blue the observed values and in red color the missing

values. It seems that some of the features have a pattern of missing data (there are several red cells in the same row). Because of that, MCAR gets discarded.

The prediction models are very sensitive to missing values, so we would have to take measures in order to make predictions.

In order to study the models performance we will create five different datasets:

Dataset A = data with no methodologies applied

```
dataA<-data
dim(dataA) #Dimensions of the original dataset
```

```
## [1] 1176   48
```

```
round(mean(is.na(dataA))*100, 2)#% of the missing values
```

```
## [1] 3.37
```

```
table(dataA$outcome)
```

```
##
##     Survivor Non-survivor
##         1017          159
```

The original dataset contains 1176 observations, 48 variables and 3,4% of missing values.

Dataset B = listwise deletion

This method creates a subset with the complete observations.

```
sum(complete.cases(dataA))#Determine the complete observations
```

```
## [1] 428
```

```
nrow(dataA)-sum(complete.cases(dataA))#Determine the observations with missing values
```

```
## [1] 748
```

```
dataB<-na.omit(dataA)#Create the dataset omitting the missing values
sum(is.na(dataB))#Checking for missing values
```

```
## [1] 0
```

```
dim(dataB)# Dimensions of the new dataset
```

```
## [1] 428  48
```

```
table(dataB$outcome)
```

```
##
##     Survivor Non-survivor
##          363           65
```

The dataset B after appying the listwise deletion consist in 428 complete observations and 48 variables.

Dataset C = deletion of variables with >15% NA

This method is another kind of deletion method where the features with >15% of missing values are deleted.

```
dataC<- dataA[, which((apply(is.na(dataA), 2, mean)*100)<15)]
round(mean(is.na(dataC))*100, 2)#Checking for missing values
```

```
## [1] 1.22
```

```
dim(dataC)# Dimensions of the new dataset
```

```
## [1] 1176   43
```

```
table(dataC$outcome)
```

```
##
##     Survivor Non-survivor
##         1017          159
```

This second method creates a data set with 1177 observations, 43 variables and an 1,22% of missing values. The omitted variables are PCO2, PH, Basophils, Lactic.acid and BMI.

Dataset D = KNN imputations

This third method is a type of imputation method of handling missing values. It consists in a machine learning-based method that uses a Euclidean distance to find the nearest neighbors.

```
k<- round(sqrt(nrow(dataA))) #Determine the best k
dataD<-kNN(dataA, variable = colnames(dataA), k = 34, imp_var = FALSE) #Generate the imputation with k=
round(mean(is.na(dataD)*100), 2)#Checking for missing values
```

```
## [1] 0
```

```
dim(dataD)# Dimensions of the new dataset
```

```
## [1] 1176   48
```

```
table(dataD$outcome)
```

```
##
##     Survivor Non-survivor
##         1017          159
```

This imputation method creates a data set with the same dimensions as dataset A but without missing values (1176 observations and 48 variables).

Dataset E = multiple imputation

Another imputation method which consist in generating multiple imputed values from the observed data.

```
columns<- c("PCO2", "pH", "basophils", "lactic.acid", "BMI", "creatine.kinase", "lymphocyte", "neutrophi
pmm.data<- mice(dataA[,names(dataA) %in% columns], seed=12345, printFlag = FALSE, m = 30)#Generate the
imputed.data<- mice::complete(pmm.data)
complete.data<-dataA[, which((apply(is.na(dataA), 2, mean)*100)<0.01)]
dataE<-cbind(complete.data, imputed.data)#Create the new dataset
round(mean(is.na(dataE)*100), 2)#Checking for missing values
```

```
## [1] 0
```

```
dim(dataE)# Dimensions of the new dataset
```

```
## [1] 1176   48
```

```
table(dataE$outcome)
```

```
##
##     Survivor Non-survivor
##         1017          159
```

The multiple imputation method also creates a data set with the same dimensions as dataset A but without missing values (1176 observations and 48 variables).