Data Imputation

Mingcheng Hu

Table of contents


```
library(missForest)
library(ggplot2)
library(dplyr)
```

```
Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

```
library(corrplot)
```

```
corrplot 0.92 loaded
```

```r
library(parallel)
library(doParallel)
```

```
Loading required package: foreach
```

```
Loading required package: iterators
```

```r
options(scipen = 999)  # disable scientific notation

setwd("/work/users/y/u/yuukias/BIOS-Material/BIOS992/src/step3_impute_split_data/impute_data")
```

Import and Preprocess Data

```r
data_unimputed <- read.csv("eligible_data.csv")
(dim(data_unimputed))
```

```
[1] 35159    15
```

```r
head(data_unimputed)
```

```
      eid age sex ethnicity     BMI smoking diabetes systolic_bp
1 1000205  40   1         1 21.5595       0        0         149
2 1000239  65   0         1 22.9214       1        0         137
3 1000677  42   0         1 37.8920       2        0         124
4 1000737  52   1         1 22.8374       0        0         148
5 1000779  56   1         1 25.0194       0        0         144
6 1000928  63   0         1 30.9546       1        0         120
  hypertension_treatment total_chol hdl_chol education activity max_workload
1                      0      4.569    1.228         2        0          130
2                      0      5.780    2.221         1        1           60
3                      0      5.874    1.323         3        1           80
4                      0      4.429       NA         4        2          110
5                      0      6.258    1.406         3        2          110
6                      0         NA       NA         1        1           60
  max_heart_rate
1            139
```

```
2               126
3               109
4               112
5               112
6               130
```

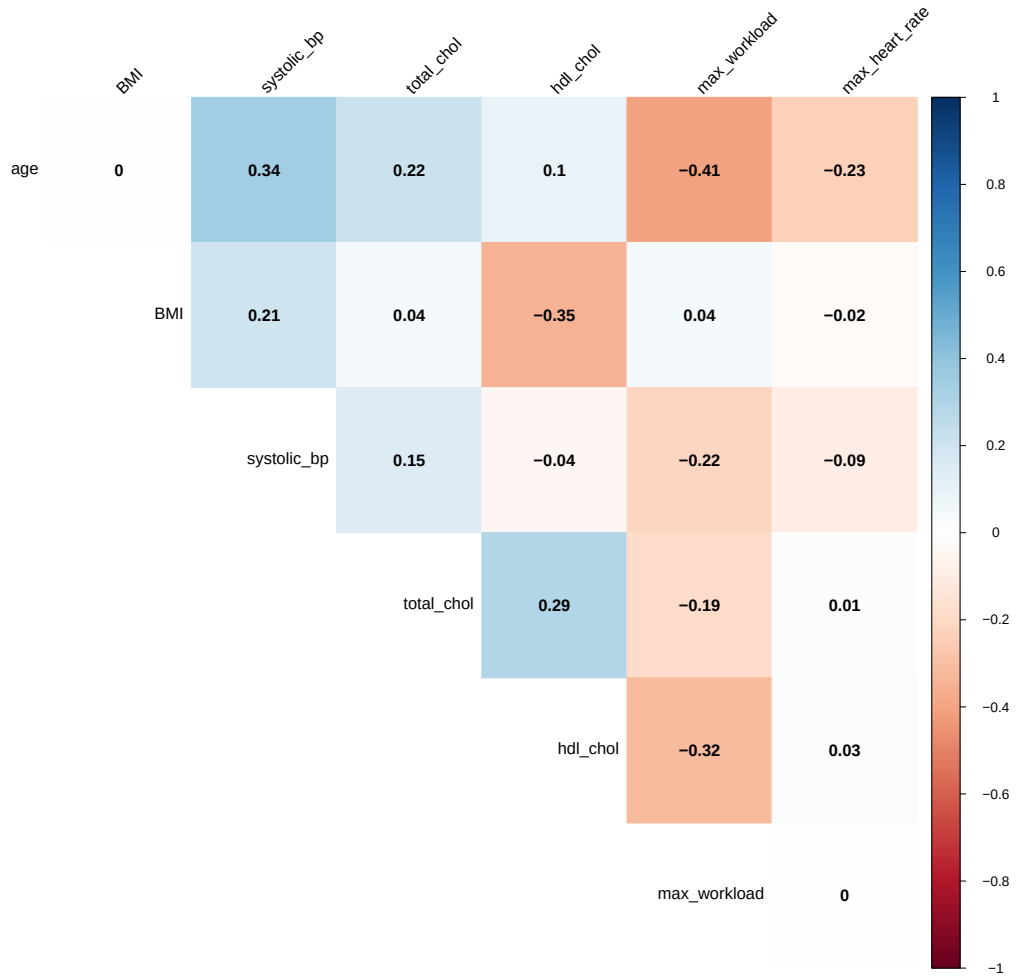
For the missing rate, check `export_data.ipynb`.

Then, we should declare the type of categorical variables

```r
data_unimputed$ethnicity <- as.factor(data_unimputed$ethnicity)
data_unimputed$sex <- as.factor(data_unimputed$sex)
data_unimputed$smoking <- as.factor(data_unimputed$smoking)
data_unimputed$diabetes <- as.factor(data_unimputed$diabetes)
data_unimputed$hypertension_treatment <-
 ↪  as.factor(data_unimputed$hypertension_treatment)
data_unimputed$education <- as.factor(data_unimputed$education)
data_unimputed$activity <- as.factor(data_unimputed$activity)
```

We can also visualize the correlation matrix of the numeric variables.

```r
data_numeric <- data_unimputed[, sapply(data_unimputed, is.numeric)]
# exclude eid
data_numeric <- subset(data_numeric, select = -eid)
cor_matrix <- cor(data_numeric, use = "pairwise.complete.obs")
corrplot(cor_matrix,
         method = "color",
         type = "upper",
         order = "original",
         addCoef.col = "black",
         tl.col = "black",
         tl.srt = 45,
         diag = FALSE,
         na.label = "NA")
```



There are no strong correlations among the numeric variables. Only the cholesterol variables and ECG-related variables are moderately correlated.

MissForest Imputation

Run Imputation

```
start_time <- proc.time()
tryCatch({
    n_cores <- min(detectCores() - 1, 8)
    registerDoParallel(cores = n_cores)
```

```
    data_imputed <- missForest(data_unimputed,
        ntree = 100, maxiter = 10, verbose = TRUE,
        parallel = "variables")

    stopImplicitCluster()
}, error = function(e) {
    print(e)
    return(NULL)
})
```

```
  parallelizing over the variables of the input data matrix 'xmis'
  missForest iteration 1 in progress...

randomForest 4.7-1.1

Type rfNews() to see new features/changes/bug fixes.


Attaching package: 'randomForest'

The following object is masked from 'package:dplyr':

    combine

The following object is masked from 'package:ggplot2':

    margin

Loading required package: rngtools

done!
    estimated error(s): 0.000005463667 0.2756739
    difference(s): 0.00000000000003100323 0.01340035
    time: 1222.071 seconds

  missForest iteration 2 in progress...done!
    estimated error(s): 0.000005414305 0.2734545
    difference(s): 0.00000000000002044821 0.006468573
    time: 1236.834 seconds
```

```
  missForest iteration 3 in progress...done!
    estimated error(s): 0.000005415411 0.2731478
    difference(s): 0.00000000000001719629 0.006346678
    time: 1241.224 seconds

  missForest iteration 4 in progress...done!
    estimated error(s): 0.000005423198 0.2737687
    difference(s): 0.00000000000001632048 0.005985056
    time: 1253.97 seconds

  missForest iteration 5 in progress...done!
    estimated error(s): 0.000005420944 0.2738109
    difference(s): 0.00000000000001574633 0.00638731
    time: 1213.167 seconds

  missForest iteration 6 in progress...done!
    estimated error(s): 0.000005422673 0.2735211
    difference(s): 0.00000000000001369397 0.006127267
    time: 1225.523 seconds

  missForest iteration 7 in progress...done!
    estimated error(s): 0.000005421758 0.2732846
    difference(s): 0.00000000000001526654 0.006033814
    time: 1266.2 seconds

  missForest iteration 8 in progress...done!
    estimated error(s): 0.0000054086 0.2737204
    difference(s): 0.00000000000001630512 0.006196341
    time: 1246.667 seconds
```

```r
end_time <- proc.time()
run_time <- end_time - start_time
print(run_time)
```

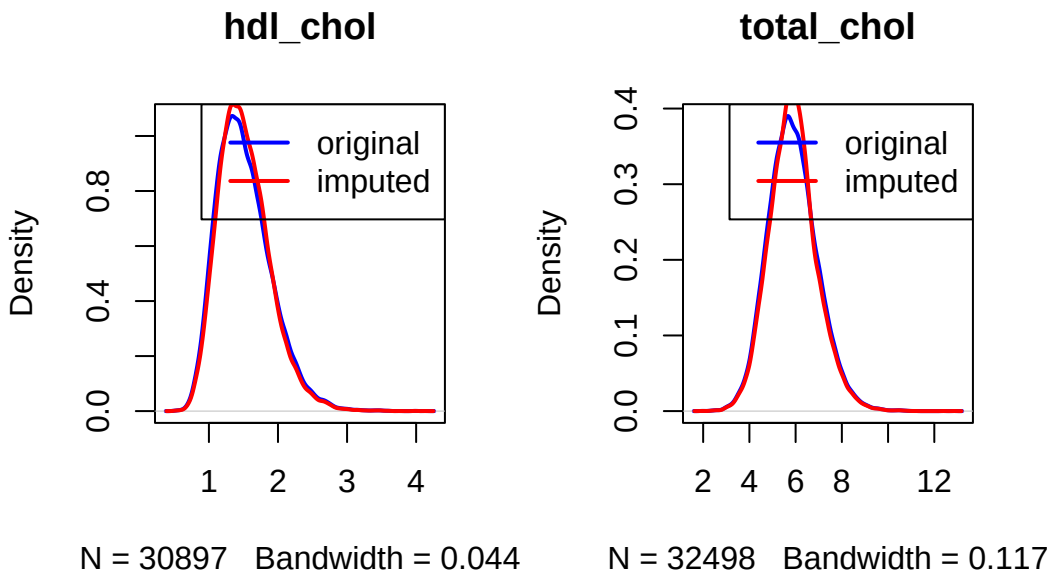
```
     user    system   elapsed
23072.521    20.753  9905.753
```

```r
save(data_imputed, file = "eligible_data_imputed_missForest.RData")
```

Check Imputation Results

We can plot densities of both the observed and imputed values of all variables to see whether the imputations are reasonable. Differences in the densities between the observed and imputed values may suggest a problem that needs to be further checked.

```
par(mfrow = c(1, 2))
for(col in c("hdl_chol", "total_chol")) {
    plot(density(data_unimputed[[col]], na.rm = TRUE),
            main = col,
            col = "blue", lwd = 2)
    lines(density(data_imputed$ximp[[col]]),
            col = "red", lwd = 2)
    legend("topright",
            legend = c("original", "imputed"),
            col = c("blue", "red"),
            lwd = 2)
}
```



Fortunately, there is no significant difference between the densities of the original and imputed values for our case.