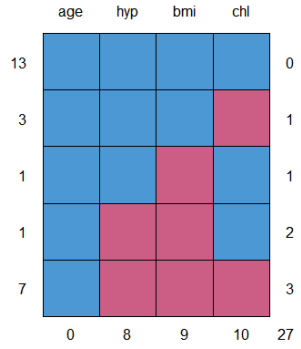
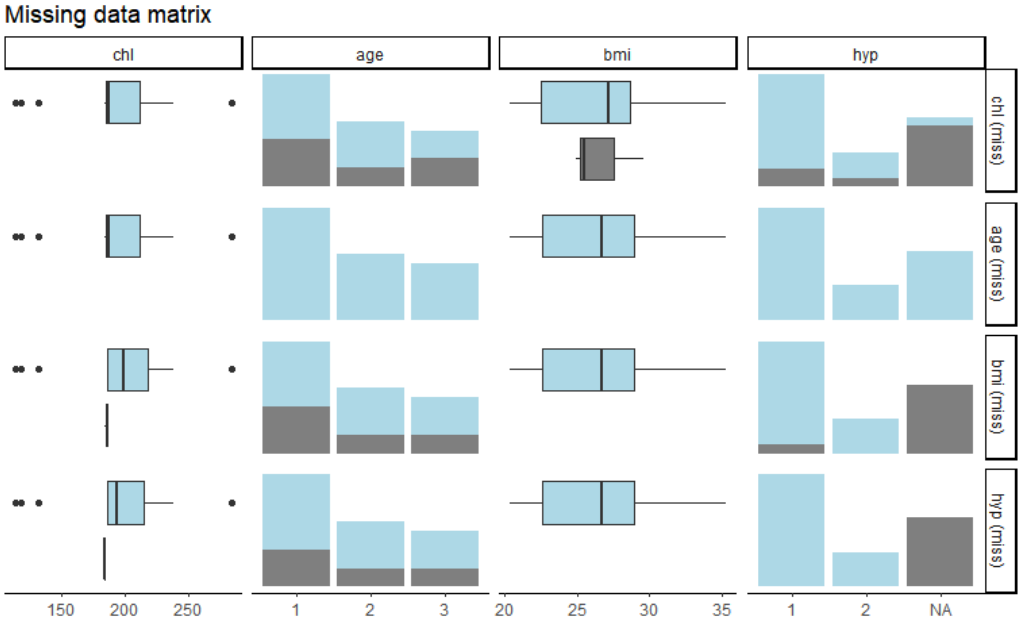
```
> ###############################################
> # Statistical Modelling & Machine Learning #
> #             R Example2                      #
> ###############################################
>
> #################
> # Missing value #
> #################
>
> # mice package: Multivariate imputation by chained equation.
> # install.packages('mice')
> library(mice)
>
> # install.packages('rms')
> library(rms)
>
> # install.packages('finalfit)
> library(finalfit)
>
> # Data
> head(nhanes)
  age  bmi hyp chl
1   1   NA  NA  NA
2   2 22.7   1 187
3   1   NA   1 187
4   3   NA  NA  NA
5   1 20.4   1 113
6   3   NA  NA 184
>
> nhanes$age = as.factor(nhanes$age)
> nhanes$hyp = as.factor(nhanes$hyp)
>
> # Description of data
> describe(nhanes)
nhanes

 4  Variables     25  Observations
---------------------------------------------------------------------------
age
      n  missing distinct
     25       0       3

Value        1    2    3
Frequency   12    7    6
Proportion 0.48 0.28 0.24
---------------------------------------------------------------------------
bmi
      n  missing distinct    Info    Mean     Gmd     .05     .10     .25
     16       9      16       1   26.56   4.897   21.38   21.85   22.65
    .50     .75     .90     .95
  26.75   28.92   31.65   33.73

lowest : 20.4 21.7 22.0 22.5 22.7, highest: 28.7 29.6 30.1 33.2 35.3

Value      20.4  21.7  22.0  22.5  22.7  24.9  25.5  26.3  27.2  27.4  27.5
Frequency     1     1     1     1     1     1     1     1     1     1     1
Proportion 0.062 0.062 0.062 0.062 0.062 0.062 0.062 0.062 0.062 0.062 0.062

Value      28.7  29.6  30.1  33.2  35.3
Frequency     1     1     1     1     1
Proportion 0.062 0.062 0.062 0.062 0.062
---------------------------------------------------------------------------
hyp
      n  missing distinct
     17       8       2

Value        1    2
Frequency   13    4
Proportion 0.765 0.235
```

```
--------------------------------------------------------------------------------
chl
      n  missing distinct     Info     Mean      Gmd      .05      .10      .25
     15       10       13    0.993    191.4    50.46    116.5    123.2    185.0
    .50      .75      .90      .95
  187.0    212.0    234.4    251.8

lowest : 113 118 131 184 186, highest: 206 218 229 238 284

Value       113   118   131   184   186   187   199   204   206   218   229
Frequency     1     1     1     1     1     3     1     1     1     1     1
Proportion 0.067 0.067 0.067 0.067 0.067 0.200 0.067 0.067 0.067 0.067 0.067

Value       238   284
Frequency     1     1
Proportion 0.067 0.067
--------------------------------------------------------------------------------
>
> # Missing pattern
> md.pattern(nhanes)
   age hyp bmi chl
13   1   1   1   1  0
3    1   1   1   0  1
1    1   1   0   1  1
1    1   0   0   1  2
7    1   0   0   0  3
     0   8   9  10 27
```

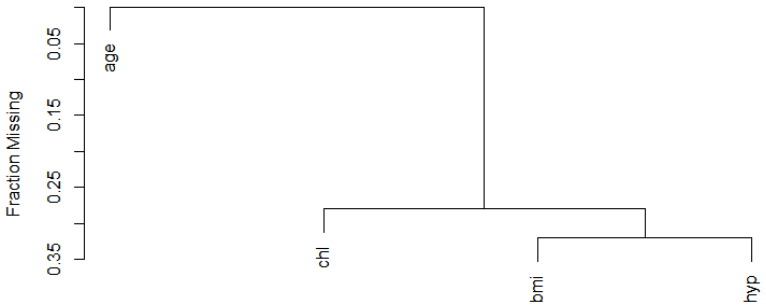



```
>
> missing_pairs(nhanes,'chl',c('age','bmi','hyp') )
```



```
> # Clustering for variables with missing values for the same obs.
>
> # Missing pattern between variables
> # (rr: both observed, mm: both missing,
> #  mr: row variable is missing & column variable is observed)
> md.pairs(nhanes)
$rr
    age bmi hyp chl
age  25  16  17  15
bmi  16  16  16  13
hyp  17  16  17  14
chl  15  13  14  15

$rm
    age bmi hyp chl
age   0   9   8  10
bmi   0   0   0   3
hyp   0   1   0   3
chl   0   2   1   0

$mr
    age bmi hyp chl
age   0   0   0   0
bmi   9   0   1   2
hyp   8   0   0   1
chl  10   3   3   0

$mm
    age bmi hyp chl
age   0   0   0   0
bmi   0   9   8   7
hyp   0   8   8   7
chl   0   7   7  10

>
> missing.clus = naclus(nhanes,method='average')
> missing.clus
varclus(x = sim, similarity = "Fraction Missing", type = "similarity.matrix",
    method = method)


Similarity matrix (NA)

    age  bmi  hyp  chl
age   0 0.00 0.00 0.00
bmi   0 0.36 0.32 0.28
hyp   0 0.32 0.32 0.28
chl   0 0.28 0.28 0.40

hclust results (method=average)


Call:
hclust(d = as.dist(1 - x), method = method)

Cluster method   : average
Number of objects: 4
```

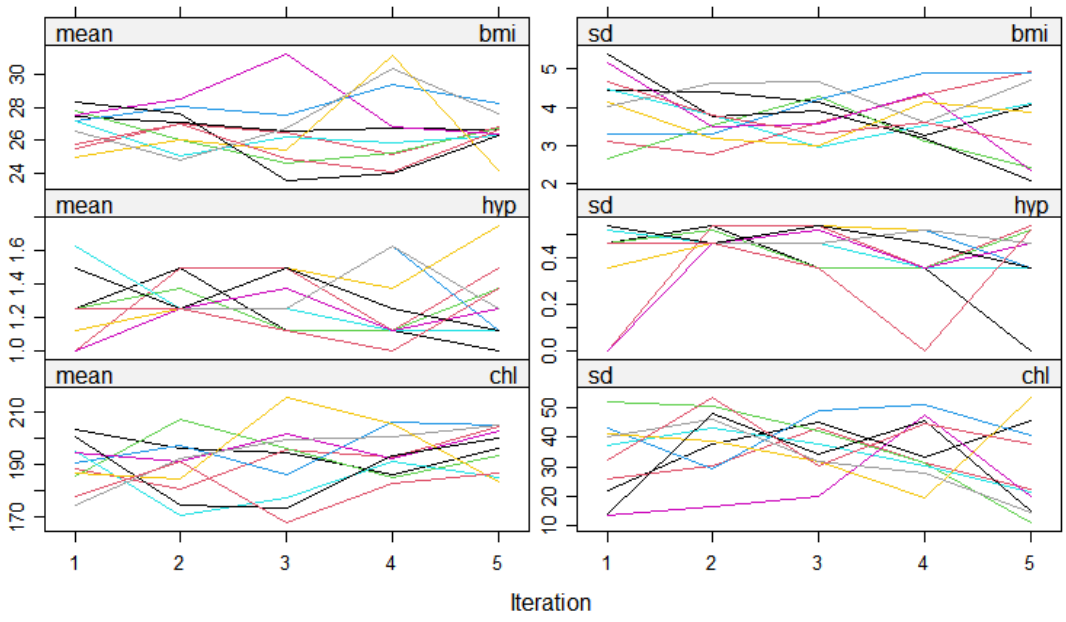
```
> plot(missing.clus)
```




```
>
> # Check missing pattern of Y variable.
> fit = glm(is.na(chl) ~ age + bmi + hyp, data=nhanes, family=binomial )
> summary(fit)

Call:
glm(formula = is.na(chl) ~ age + bmi + hyp, family = binomial,
    data = nhanes)

Deviance Residuals:
     Min       1Q   Median       3Q      Max
-1.21113  -0.59806  -0.18868  -0.00007   1.95545

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   -4.50273    6.94920  -0.648    0.517
age2         -17.32404 4794.25157  -0.004    0.997
age3           2.33743    2.34884   0.995    0.320
bmi            0.09293    0.22928   0.405    0.685
hyp2          -0.28344    2.12208  -0.134    0.894

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 15.442  on 15  degrees of freedom
Residual deviance: 11.109  on 11  degrees of freedom
  (9 observations deleted due to missingness)
AIC: 21.109

Number of Fisher Scoring iterations: 18

> # MICE
>
> ?mice
>
> # Missing values should be coded as NA.
> # m: The number of imputed datasets
> # method: Imputation methods for each column.
> # predictorMatrix: A matrix containing 0/1 data specifying
> #               the set of predictors to be used for each target column.
> #
>
> set.seed(0)
>
> imp = mice(nhanes, m=10, method=c('','pmm','logreg','pmm'), print=F)
>
> # predictors for imputation of each column
> imp$predictorMatrix
    age bmi hyp chl
age   0   1   1   1
```

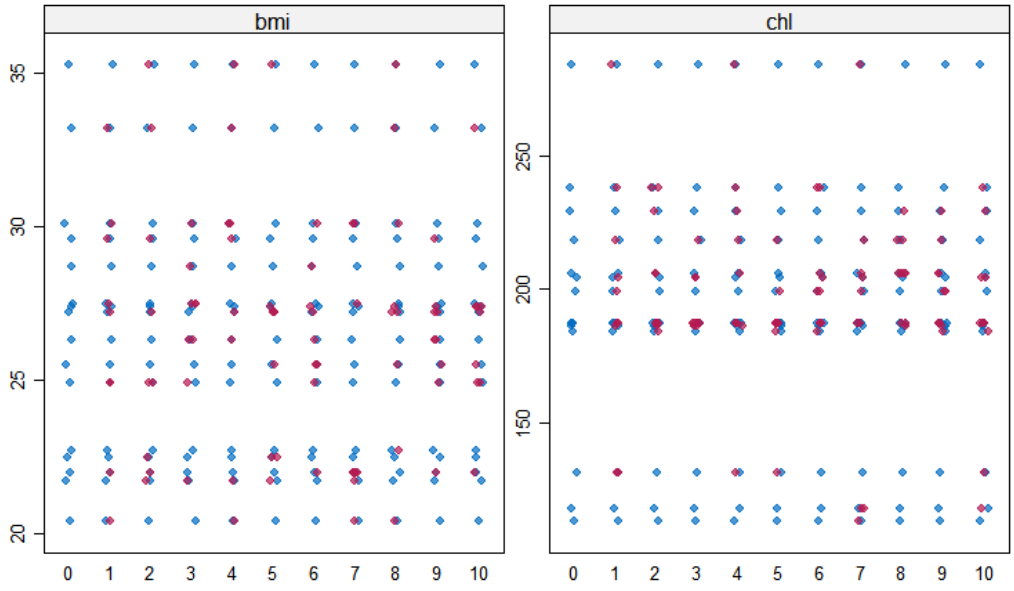
```
bmi   1   0   1   1
hyp   1   1   0   1
chl   1   1   1   0
>
> # If you don't want to use 'chl' variable to predict other variables,
> pred = imp$predictorMatrix
> pred[,'chl'] = 0
> pred
    age bmi hyp chl
age   0   1   1   0
bmi   1   0   1   0
hyp   1   1   0   0
chl   1   1   1   0
>
> imp1 = mice(nhanes, m=10, method=c('','pmm','logreg','pmm'),
+           predictorMatrix = pred, print=F)
>
> imp1$predictorMatrix
    age bmi hyp chl
age   0   1   1   0
bmi   1   0   1   0
hyp   1   1   0   0
chl   1   1   1   0
>
> # list the actual imputations for BMI
> imp$imp$bmi
     1    2    3    4    5    6    7    8    9   10
1  29.6 22.5 30.1 27.2 27.2 27.2 22.0 35.3 25.5 33.2
3  33.2 27.2 28.7 30.1 27.2 26.3 22.0 33.2 27.2 27.2
4  27.5 21.7 24.9 35.3 21.7 27.4 27.5 22.7 22.0 27.4
6  24.9 24.9 21.7 26.3 27.4 24.9 21.7 20.4 26.3 24.9
10 20.4 22.0 27.5 21.7 22.5 25.5 30.1 25.5 27.4 27.4
11 22.0 29.6 26.3 33.2 27.2 25.5 22.0 30.1 29.6 24.9
12 24.9 24.9 27.5 20.4 22.5 28.7 20.4 27.4 24.9 25.5
16 27.2 33.2 26.3 29.6 25.5 30.1 30.1 27.2 27.2 22.0
21 30.1 35.3 27.5 30.1 35.3 22.0 22.0 27.2 26.3 27.4
>
> # The first imputed dataset
> complete(imp,1)
   age  bmi hyp chl
1    1 29.6   1 131
2    2 22.7   1 187
3    1 33.2   1 187
4    3 27.5   1 186
5    1 20.4   1 113
6    3 24.9   1 184
7    1 22.5   1 118
8    1 30.1   1 187
9    2 22.0   1 238
10   2 20.4   1 218
11   1 22.0   1 187
12   2 24.9   1 199
13   3 21.7   1 206
14   2 28.7   2 204
15   1 29.6   1 131
16   1 27.2   1 187
17   3 27.2   2 284
18   2 26.3   2 199
19   1 35.3   1 218
20   3 25.5   2 204
21   1 30.1   1 238
22   1 33.2   1 229
23   1 27.5   1 131
24   3 24.9   1 284
25   2 27.4   1 186
>
> # The second imputed dataset
> complete(imp,2)
   age  bmi hyp chl
```

```
1    1 22.5   1 187
2    2 22.7   1 187
3    1 27.2   1 187
4    3 21.7   2 206
5    1 20.4   1 113
6    3 24.9   2 184
7    1 22.5   1 118
8    1 30.1   1 187
9    2 22.0   1 238
10   2 22.0   2 187
11   1 29.6   1 187
12   2 24.9   2 184
13   3 21.7   1 206
14   2 28.7   2 204
15   1 29.6   1 238
16   1 33.2   1 238
17   3 27.2   2 284
18   2 26.3   2 199
19   1 35.3   1 218
20   3 25.5   2 206
21   1 35.3   1 229
22   1 33.2   1 229
23   1 27.5   1 131
24   3 24.9   1 186
25   2 27.4   1 186
>
> # Checking the convergence of MICE.
> plot(imp, c('bmi','hyp','chl'))
```

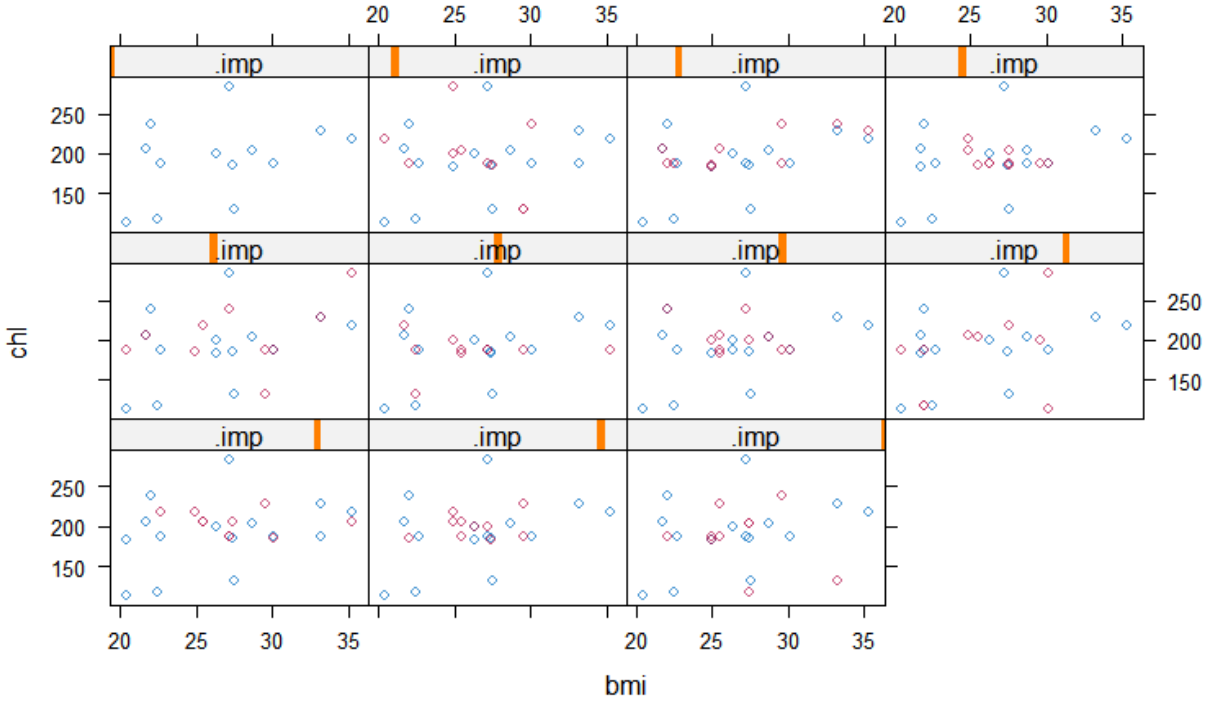



```
>
```

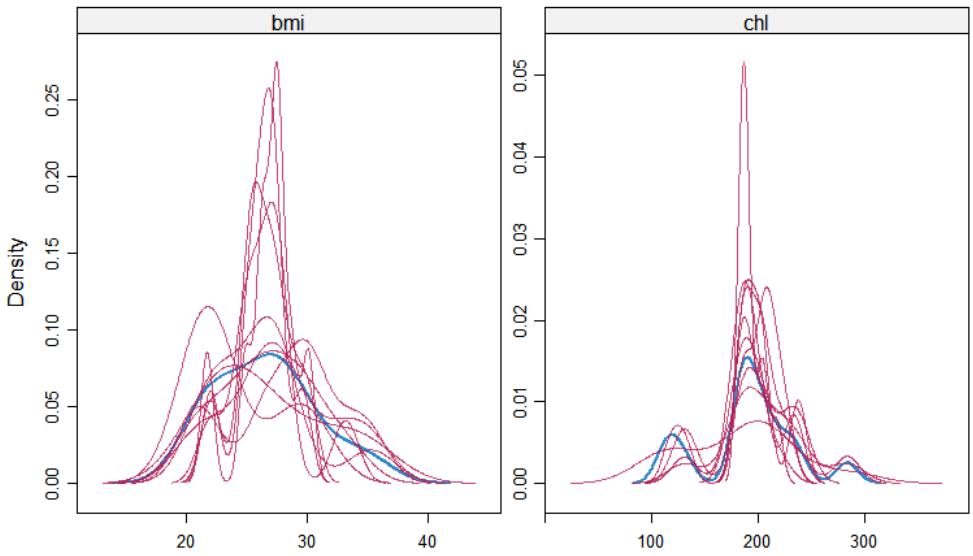
```
> # Compare the imputed data and observed data.
>
> stripplot(imp, pch=20, cex=1.2)
> # blue point: observed, red point: imputed
```




```
>
> xyplot(imp,chl ~ bmi | .imp)
> # The first plot: original complete set.
```



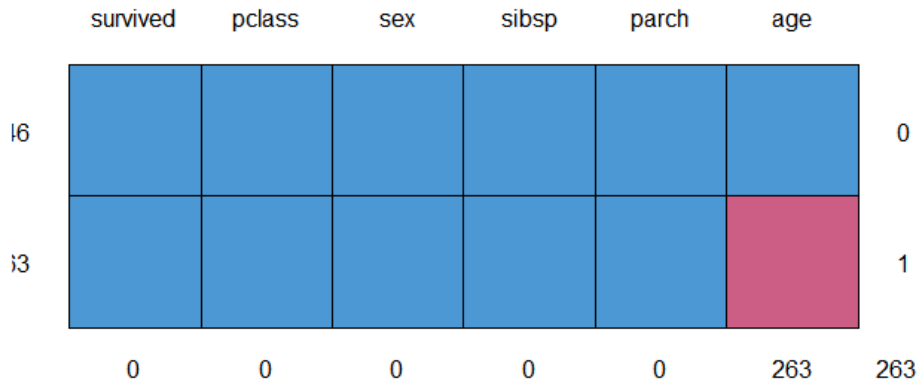
```
>
> densityplot(imp, scales=list(relation='free'),layout=c(2,1))
> # blue line: density for observed data, red line: density for imputed data
```




```
> # Prediction model with MICE
> # Goal: predict chl based on age, hyp, bmi variables
>
> set.seed(0)
>
> imp = mice(nhanes, m=10, method=c('','pmm','logreg','pmm'), print=F)
> # In mice, all variables with missing values should be imputed,
> # even if it is Y variable.
>
> # To delete obs with missing Y value, imputed Y is replaced with NA.
> md = dim(imp$imp$chl)
> iy = as.data.frame(matrix(NA,md[1],md[2]))
> colnames(iy) = colnames(imp$imp$chl)
> rownames(iy) = rownames(imp$imp$chl)
>
> imp$imp$chl = iy
>
> # Apply prediction model to each imputed dataset.
> # E.g., prediction model => linear regression model
>
> fit = with(imp, lm(chl ~ age + bmi + hyp))
>
> # Model averaging.
> summary(pool(fit))
        term   estimate std.error  statistic      df    p.value
1 (Intercept) -20.438722 67.240191 -0.3039659 7.863403 0.76904162
2       age2  53.767873 24.282857  2.2142318 7.836746 0.05838258
3       age3  84.616450 30.273159  2.7950981 7.325918 0.02550782
4        bmi   6.721977  2.340181  2.8724172 7.898132 0.02103059
5       hyp2  -3.504540 28.638268 -0.1223726 6.390101 0.90636255
>
>
> # Checking imputation effect for significant X variables
> # E.g., to impute bmi variable, we used chl (Y) variable.
>
> comp.dat = na.omit(nhanes)
> fit1 = lm(chl ~ age + bmi + hyp, data=comp.dat)
> summary(fit1)

Call:
lm(formula = chl ~ age + bmi + hyp, data = comp.dat)
```

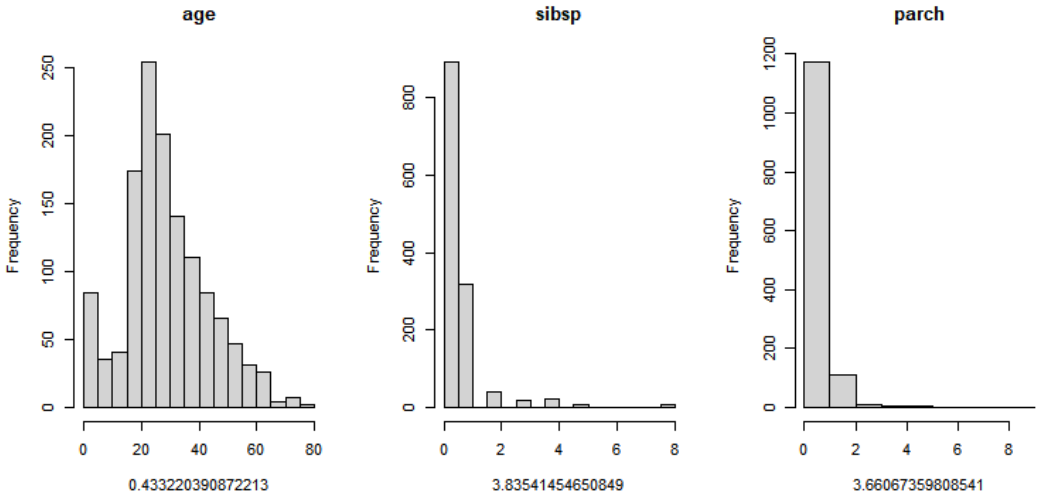
```
Residuals:
    Min      1Q  Median      3Q     Max
-34.054 -17.670   0.599   7.157  56.611

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -35.677     63.245  -0.564  0.58815
age2          59.543     22.947   2.595  0.03187 *
age3         109.458     30.437   3.596  0.00702 **
bmi            7.160      2.201   3.253  0.01164 *
hyp2          -7.692     25.179  -0.305  0.76779
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 30.69 on 8 degrees of freedom
Multiple R-squared:  0.736,    Adjusted R-squared:  0.604
F-statistic: 5.575 on 4 and 8 DF,  p-value: 0.01916

>
>
> # Predict test obs.
> M = imp$m
> imp.dat = vector(mode='list',length=M)
> for (m in 1:M) imp.dat[[m]] = complete(imp,m)
>
> p.model = function(dat) lm(chl ~ age + bmi + hyp, data=dat)
>
> fit.imp = lapply(imp.dat, p.model)
>
> test.obs = data.frame(age=c('2','1'),bmi = c(23.3,21.5),hyp=c('1','1'))
>
> yhat = lapply(fit.imp, predict, newdata=test.obs)
> yhat = matrix(unlist(yhat),nrow(test.obs),M)
>
> apply(yhat,1,mean)
[1] 189.9512 124.0838

> ########################
> # Data transformation #
> ########################
>
> # install.packages('rms')
> library(rms)
> library(e1071)
>
> getHdata(titanic3)
> dat = titanic3[,c('survived','pclass','age','sex','sibsp','parch')]
>
> describe(dat)
dat

 6  Variables     1309  Observations
---------------------------------------------------------------------------
survived : Survived
      n  missing distinct    Info     Sum    Mean     Gmd
   1309       0       2   0.708     500   0.382   0.4725

---------------------------------------------------------------------------
pclass
      n  missing distinct
   1309       0       3

Value       1st   2nd   3rd
Frequency   323   277   709
Proportion 0.247 0.212 0.542
---------------------------------------------------------------------------
age : Age [Year]
      n  missing distinct    Info    Mean     Gmd     .05     .10     .25
   1046     263      98   0.999   29.88   16.06       5      14      21
```

```
    .50     .75     .90     .95
     28      39      50      57

lowest :  0.1667  0.3333  0.4167  0.6667  0.7500, highest: 70.5000 71.0000 74.0000 76.0000 80.0
---------------------------------------------------------------------------
sex
      n  missing distinct
   1309       0       2

Value     female   male
Frequency    466    843
Proportion  0.356  0.644
---------------------------------------------------------------------------
sibsp : Number of Siblings/Spouses Aboard
      n  missing distinct     Info     Mean      Gmd
   1309       0       7    0.67   0.4989    0.777

lowest : 0 1 2 3 4, highest: 2 3 4 5 8

Value         0     1     2     3     4     5     8
Frequency   891   319    42    20    22     6     9
Proportion 0.681 0.244 0.032 0.015 0.017 0.005 0.007
---------------------------------------------------------------------------
parch : Number of Parents/Children Aboard
      n  missing distinct     Info     Mean      Gmd
   1309       0       8    0.549    0.385   0.6375

lowest : 0 1 2 3 4, highest: 3 4 5 6 9

Value         0     1     2     3     4     5     6     9
Frequency  1002   170   113     8     6     6     2     2
Proportion 0.765 0.130 0.086 0.006 0.005 0.005 0.002 0.002
---------------------------------------------------------------------------
>
> md.pattern(dat)
     survived pclass sex sibsp parch age
1046        1      1   1     1     1   1   0
263         1      1   1     1     1   0   1
            0      0   0     0     0 263 263
```

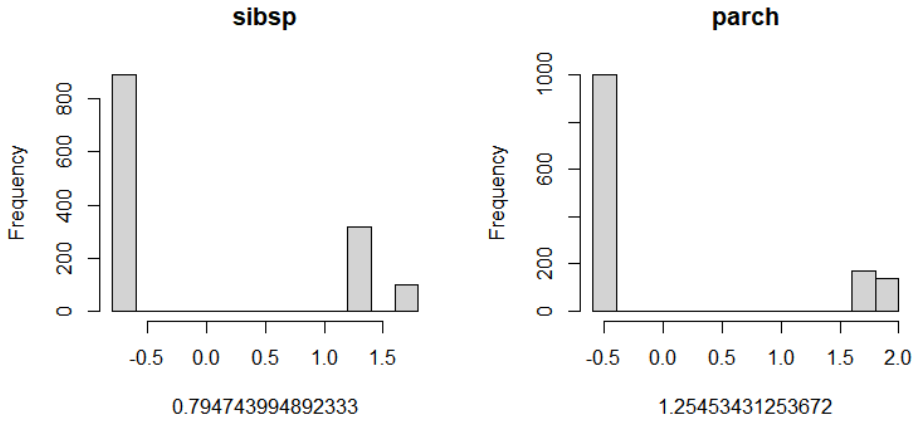



```
>
> imp = mice(dat,m=1,method='pmm')
```

```
 iter imp variable
  1   1  age
  2   1  age
  3   1  age
  4   1  age
  5   1  age
> imp.dat = complete(imp)
>
> par(mfrow=c(1,3))
> for (j in c('age','sibsp','parch'))
+ {
+   hist(imp.dat[,j], main=j,xlab = skewness(imp.dat[,j]))
+ }
```




```
> # Standardization or centering: Use 'scale()' function.
> # Yeo-Johnson transformation
> # install.packages('bestNormalize')
> library(bestNormalize)
>
> imp.dat1 = imp.dat
> imp.dat1$sibsp = yeojohnson(imp.dat$sibsp)$x.t
> imp.dat1$parch = yeojohnson(imp.dat$parch)$x.t
>
> par(mfrow=c(1,2))
> for (j in c('sibsp','parch'))
+ {
+   hist(imp.dat1[,j], main=j,xlab = skewness(imp.dat1[,j]))
+ }
```




```
>
> # Discretization of continuous variable
```

```
>
> imp.dat2 = transform(imp.dat,
+    agec = ifelse(age < 21, 'child','adult'),
+    sibsp = ifelse(sibsp==0, 'no sibsp','sibsp'),
+    parch = ifelse(parch==0, 'no parch','parch'))
>
> head(imp.dat2)
  survived pclass    age    sex    sibsp    parch  agec
1        1    1st 29.0000 female no sibsp no parch adult
2        1    1st  0.9167   male    sibsp    parch child
3        0    1st  2.0000 female    sibsp    parch child
4        0    1st 30.0000   male    sibsp    parch adult
5        0    1st 25.0000 female    sibsp    parch adult
6        1    1st 48.0000   male no sibsp no parch adult
```