Chapter 18: Logistic Regression

Exercise 2: Low birth weight?

Yêu cầu: Logistic Regression để thực hiện việc xác định trẻ có thiếu cân hay không dựa vào thông tin còn lại.

- Cho dữ liệu birthweight_reduced.csv
- Tạo dataset
- In thông tin head, tail, số dòng, số cột, str, summary
- Vẽ biểu đồ quan sát mối liên hệ giữa các biến (corrplot)
- Tạo train:test từ dữ liệu data với tỉ lệ 80:20
- Áp dụng thuật toán Logistic Regression
- Kiểm tra độ chính xác
- Tìm kết quả Cho dữ liệu Test: c(12, 18, 4.5, 35, 1, 41, 7, 65, 125, 37, 14, 25, 68, 1, 1)

In [1]:
```
library(corrplot)
mydata <- read.csv("birthweight_reduced.csv")
```

In [2]:
```
print(str(mydata))
```

```
'data.frame':	42 obs. of  17 variables:
 $ id              : int  1313 431 808 300 516 321 1363 575 822 1081 ...
 $ headcirumference: int  12 12 13 12 13 13 12 12 13 14 ...
 $ length          : int  17 19 19 18 18 19 19 19 19 21 ...
 $ Birthweight     : num  5.8 4.2 6.4 4.5 5.8 6.8 5.2 6.1 7.5 8 ...
 $ Gestation       : int  33 33 34 35 35 37 37 37 38 38 ...
 $ smoker          : int  0 1 0 1 1 0 1 1 0 0 ...
 $ motherage       : int  24 20 26 41 20 28 20 19 20 18 ...
 $ mnocig          : int  0 7 0 7 35 0 7 7 0 0 ...
 $ mheight         : int  58 63 65 65 67 62 64 65 62 67 ...
 $ mppwt           : int  99 109 140 125 125 118 104 132 103 109 ...
 $ fage            : int  26 20 25 37 23 39 20 20 22 20 ...
 $ fedyrs          : int  16 10 12 14 12 10 10 14 14 12 ...
 $ fnocig          : int  0 35 25 25 50 0 35 0 0 7 ...
 $ fheight         : int  66 71 69 68 73 67 73 72 70 67 ...
 $ lowbwt          : int  1 1 0 1 1 0 1 0 0 0 ...
 $ mage35          : int  0 0 0 1 0 0 0 0 0 0 ...
 $ LowBirthWeight  : Factor w/ 2 levels "Low","Normal": 1 1 2 1 1 2 1 2 2 2 ...
NULL
```

In [3]:
```
## view the first few rows of the data
print(head(mydata))
#print(tail(mydata))
```

```
    id headcirumference length Birthweight Gestation smoker motherage mnocig
1 1313               12     17         5.8        33      0        24      0
2  431               12     19         4.2        33      1        20      7
3  808               13     19         6.4        34      0        26      0
4  300               12     18         4.5        35      1        41      7
5  516               13     18         5.8        35      1        20     35
6  321               13     19         6.8        37      0        28      0
  mheight mppwt fage fedyrs fnocig fheight lowbwt mage35 LowBirthWeight
1      58    99   26     16      0      66      1      0            Low
2      63   109   20     10     35      71      1      0            Low
3      65   140   25     12     25      69      0      0         Normal
4      65   125   37     14     25      68      1      1            Low
5      67   125   23     12     50      73      1      0            Low
6      62   118   39     10      0      67      0      0         Normal
```

In [4]:
```
print(summary(mydata))
```

```
       id         headcirumference     length       Birthweight
 Min.   :  27.0   Min.   :12.00    Min.   :17.00   Min.   : 4.200
 1st Qu.: 537.2   1st Qu.:13.00    1st Qu.:19.00   1st Qu.: 6.450
 Median : 821.0   Median :13.00    Median :20.00   Median : 7.250
 Mean   : 894.1   Mean   :13.26    Mean   :19.93   Mean   : 7.264
 3rd Qu.:1269.5   3rd Qu.:14.00    3rd Qu.:21.00   3rd Qu.: 8.000
 Max.   :1764.0   Max.   :15.00    Max.   :22.00   Max.   :10.000
   Gestation         smoker          motherage         mnocig
 Min.   :33.00   Min.   :0.0000   Min.   :18.00   Min.   : 0.000
 1st Qu.:38.00   1st Qu.:0.0000   1st Qu.:20.25   1st Qu.: 0.000
 Median :39.50   Median :1.0000   Median :24.00   Median : 4.500
 Mean   :39.19   Mean   :0.5238   Mean   :25.55   Mean   : 9.429
 3rd Qu.:41.00   3rd Qu.:1.0000   3rd Qu.:29.00   3rd Qu.:15.750
 Max.   :45.00   Max.   :1.0000   Max.   :41.00   Max.   :50.000
    mheight          mppwt            fage           fedyrs          fnocig
 Min.   :58.0   Min.   : 99.0   Min.   :19.0   Min.   :10.00   Min.   : 0.00
 1st Qu.:63.0   1st Qu.:115.0   1st Qu.:23.0   1st Qu.:12.00   1st Qu.: 0.00
 Median :64.0   Median :125.0   Median :29.5   Median :14.00   Median :18.50
 Mean   :64.4   Mean   :125.9   Mean   :28.9   Mean   :13.67   Mean   :17.19
 3rd Qu.:66.0   3rd Qu.:135.0   3rd Qu.:32.0   3rd Qu.:16.00   3rd Qu.:25.00
 Max.   :71.0   Max.   :170.0   Max.   :46.0   Max.   :16.00   Max.   :50.00
    fheight          lowbwt             mage35        LowBirthWeight
 Min.   :66.00   Min.   :0.0000   Min.   :0.00000   Low   : 6
 1st Qu.:69.00   1st Qu.:0.0000   1st Qu.:0.00000   Normal:36
 Median :71.00   Median :0.0000   Median :0.00000
 Mean   :70.76   Mean   :0.1429   Mean   :0.09524
 3rd Qu.:72.00   3rd Qu.:0.0000   3rd Qu.:0.00000
 Max.   :78.00   Max.   :1.0000   Max.   :1.00000
```

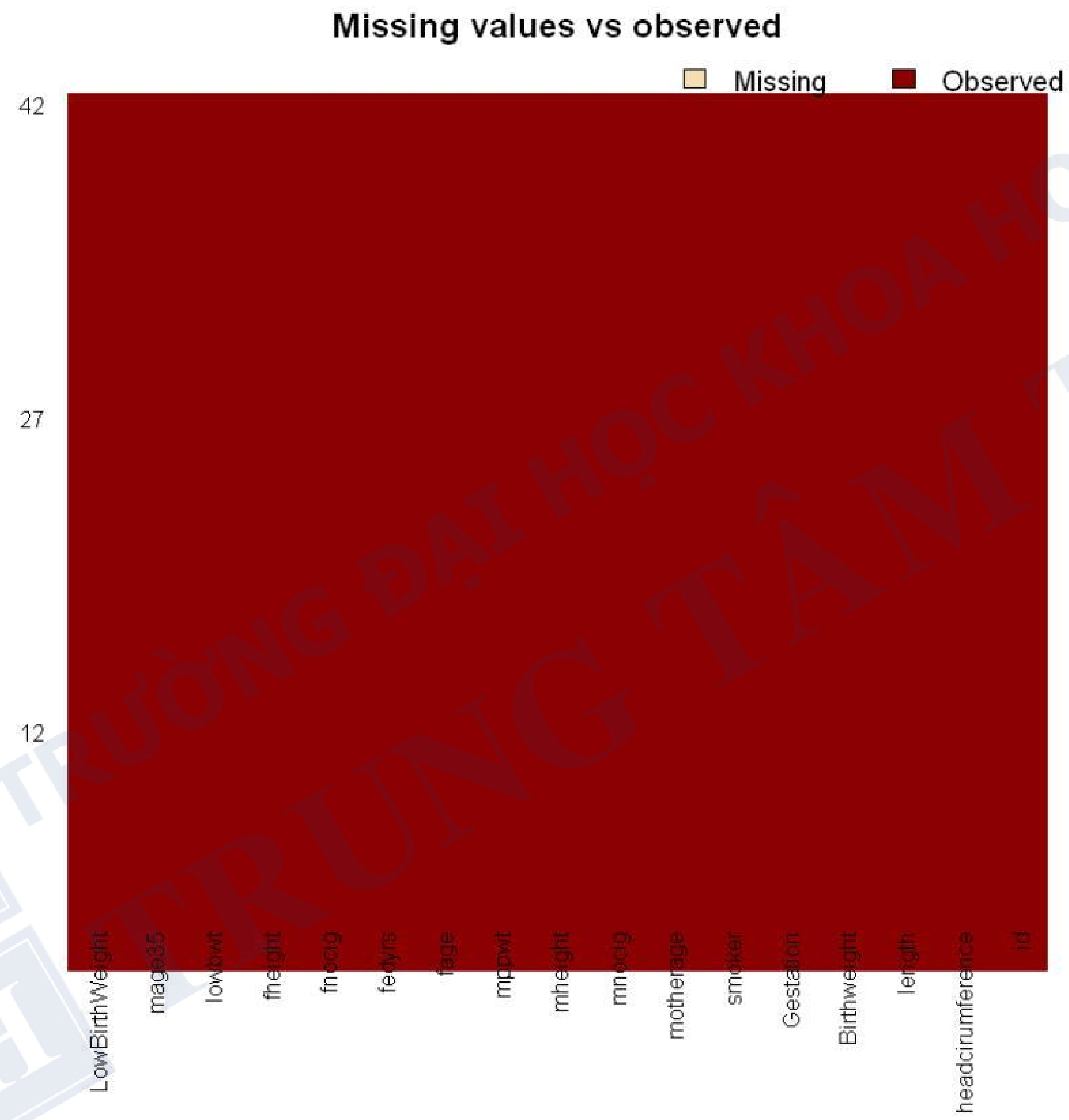
In [5]:
```
print(paste("rows:", ncol(mydata)))
print(paste("cols:", nrow(mydata)))
```

```
[1] "rows: 17"
[1] "cols: 42"
```

In [6]:
```
# check missing value
library(Amelia)
missmap(mydata, main = "Missing values vs observed")
```

```
Loading required package: Rcpp

##
## Amelia II: Multiple Imputation
## (Version 1.7.4, built: 2015-12-05)
## Copyright (C) 2005-2020 James Honaker, Gary King and Matthew Blackwell
## Refer to http://gking.harvard.edu/amelia/ (http://gking.harvard.edu/amelia/)
for more information
##
```



In [7]:
```
# Check Class bias
print(table(mydata$LowBirthWeight))
```

```
   Low Normal
     6     36
```

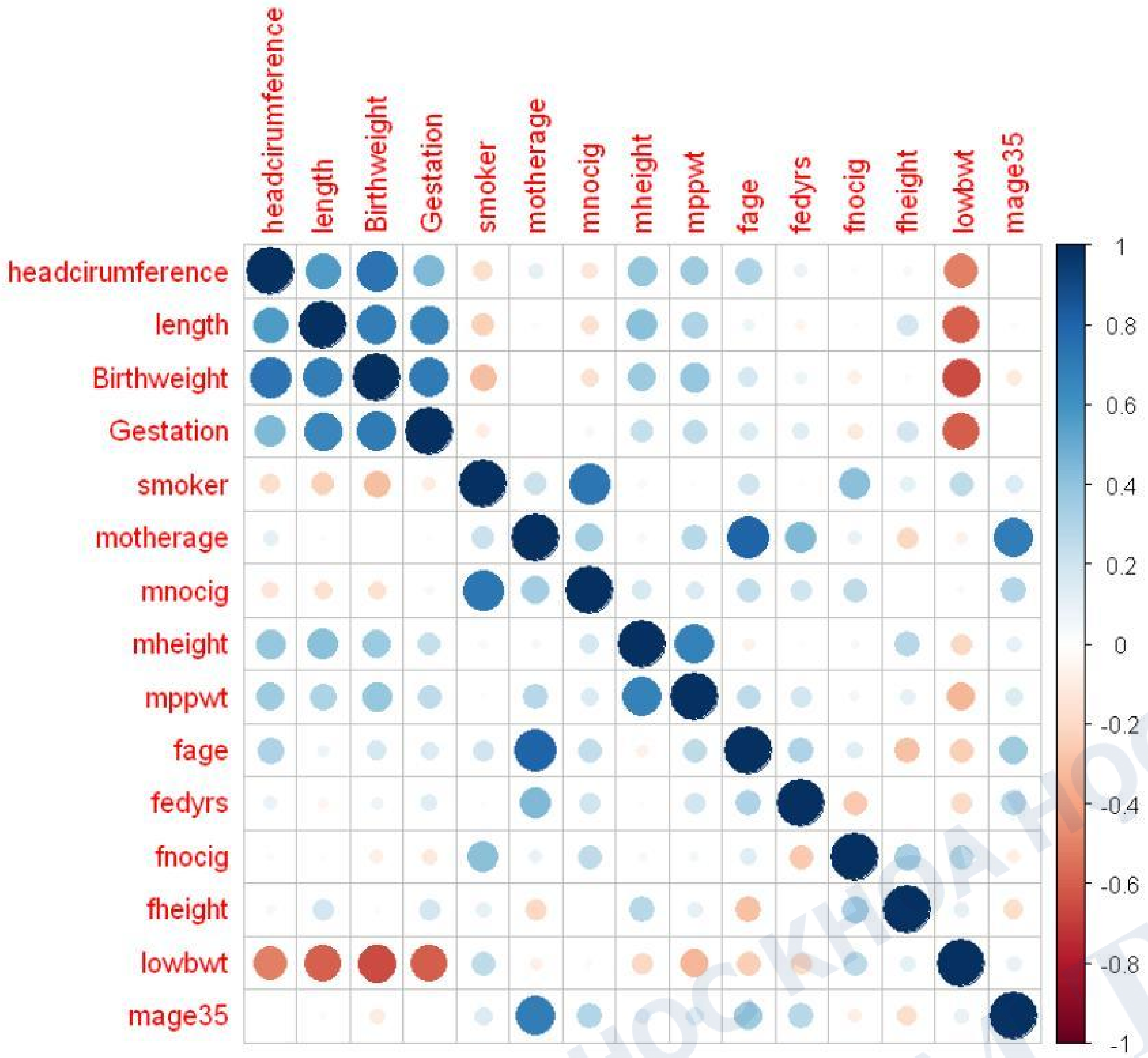
In [8]:

```
# BoxPlot to Check for outliers
# drop rows having outliers

# calculating the correlation between each pair of numeric variables
correlations <- cor(mydata[,2:16])
correlations
```

	headcirumference	length	Birthweight	Gestation	smoker	mothe
headcirumference	1.000000000	0.56532849	0.736396310	0.443974538	-0.17375085	0.11210
length	0.565328491	1.00000000	0.697008279	0.651402769	-0.23534939	-0.02071
Birthweight	0.736396310	0.69700828	1.000000000	0.706291950	-0.30895001	0.00104
Gestation	0.443974538	0.65140277	0.706291950	1.000000000	-0.09474608	0.01077
smoker	-0.173750846	-0.23534939	-0.308950015	-0.094746078	1.00000000	0.21247
motherage	0.112108327	-0.02071895	0.001040475	0.010778455	0.21247879	1.00000
mnocig	-0.131437996	-0.15713803	-0.151227745	0.043194856	0.72721809	0.34029
mheight	0.381293418	0.41473145	0.367947042	0.230929298	0.03968201	0.04678
mppwt	0.357593509	0.30439408	0.389580646	0.250515534	0.01258798	0.27764
fage	0.301363456	0.07890718	0.176790000	0.142175334	0.19750145	0.80658
fedyrs	0.083416559	-0.05072288	0.073869580	0.130986636	-0.01489058	0.44168
fnocig	-0.027734282	0.01971581	-0.088927203	-0.113830614	0.41763296	0.09092
fheight	0.040466392	0.18713730	0.024784274	0.187866905	0.10583531	-0.20360
lowbwt	-0.500246731	-0.59224820	-0.651804466	-0.602934976	0.25301216	-0.07639
mage35	-0.005096869	0.02107483	-0.108480485	0.007394508	0.14693845	0.69266

```
In [9]: corrplot(correlations, method="circle")
```



In [10]:

```
# divided into train and test: 70 - 30
mydata <- mydata[, 2:17]
print(head(mydata))
```

```
  headcirumference length Birthweight Gestation smoker motherage mnocig mheight
1               12     17         5.8        33      0        24      0      58
2               12     19         4.2        33      1        20      7      63
3               13     19         6.4        34      0        26      0      65
4               12     18         4.5        35      1        41      7      65
5               13     18         5.8        35      1        20     35      67
6               13     19         6.8        37      0        28      0      62
  mppwt fage fedyrs fnocig fheight lowbwt mage35 LowBirthWeight
1    99   26     16      0      66      1      0            Low
2   109   20     10     35      71      1      0            Low
3   140   25     12     25      69      0      0         Normal
4   125   37     14     25      68      1      1            Low
5   125   23     12     50      73      1      0            Low
6   118   39     10      0      67      0      0         Normal
```

In [11]:

```
n = nrow(mydata)
trainIndex = sample(1:n, size = round(0.7*n), replace=FALSE)
train = mydata[trainIndex ,]
test = mydata[-trainIndex ,]
print("Rows of training data and test data:")
print(nrow(train))
print(nrow(test))
```

```
[1] "Rows of training data and test data:"
[1] 29
[1] 13
```

In [12]:
```
# estimates a logistic regression model using the glm (generalized linear model)
mylogit <- glm(LowBirthWeight ~ ., data = train, family = "binomial")
print(summary(mylogit))
```

```

Call:
glm(formula = LowBirthWeight ~ ., family = "binomial", data = train)

Deviance Residuals:
       Min          1Q      Median          3Q         Max  
-3.971e-06   3.971e-06   3.971e-06   3.971e-06   3.971e-06  

Coefficients:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)       2.557e+01  2.283e+06       0        1
headcirumference  2.394e-07  1.553e+05       0        1
length           -4.789e-07  7.673e+04       0        1
Birthweight      -7.068e-07  9.882e+04       0        1
Gestation         2.199e-07  3.224e+04       0        1
smoker           -4.695e-07  1.471e+05       0        1
motherage        -6.980e-08  2.662e+04       0        1
mnocig            1.314e-08  5.185e+03       0        1
mheight           4.298e-08  4.016e+04       0        1
mppwt             1.170e-09  6.293e+03       0        1
fage              2.257e-08  2.161e+04       0        1
fedyrs           -1.256e-08  2.780e+04       0        1
fnocig           -1.218e-08  3.734e+03       0        1
fheight           4.529e-09  2.751e+04       0        1
lowbwt           -5.113e+01  2.686e+05       0        1
mage35            1.135e-07  3.308e+05       0        1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 2.6662e+01  on 28  degrees of freedom
Residual deviance: 4.5733e-10  on 13  degrees of freedom
AIC: 32

Number of Fisher Scoring iterations: 24
```

In [13]:

```
pred = predict(mylogit,
               newdata = test,
               type = "response")

pred_value <- ifelse(pred > 0.5, "Normal", "Low")
print("Testdata admit vs predict:")
result <- data.frame(Actual = test$LowBirthWeight, pred_value)
print(result)
```

```
[1] "Testdata admit vs predict:"
   Actual pred_value
6  Normal     Normal
8  Normal     Normal
10 Normal     Normal
15 Normal     Normal
17 Normal     Normal
19    Low        Low
21 Normal     Normal
25 Normal     Normal
27 Normal     Normal
28 Normal     Normal
34 Normal     Normal
37 Normal     Normal
40 Normal     Normal
```

In [14]:

```
# SOLUTION 1
misClasificError <- mean(pred_value != test$LowBirthWeight)
print(paste('Accuracy s2: ',1-misClasificError))
```

```
[1] "Accuracy s2:  1"
```

In [15]:

```
names(test)
```

'headcirumference' 'length' 'Birthweight' 'Gestation' 'smoker' 'motherage' 'mnocig' 'mheight' 'mppwt' 'fage' 'fedyrs' 'fnocig' 'fheight' 'lowbwt' 'mage35' 'LowBirthWeight'

In [16]:

```
# predict new
# sample: (12, 18, 4.5, 35, 1, 41, 7, 65, 125, 37, 14, 25, 68, 1, 1)
y1 <- predict(mylogit,
              newdata = data.frame(headcirumference = c(12),
                                   length = c(18),
                                   Birthweight = c(4.5),
                                   Gestation = c(35),
                                   smoker = c(1),
                                   motherage = c(41),
                                   mnocig = c(7),
                                   mheight = c(65),
                                   mppwt = c(125),
                                   fage = c(37),
                                   fedyrs = c(14),
                                   fnocig = c(25),
                                   fheight = c(68),
                                   lowbwt = c(1),
                                   mage35 = c(1)
                                  ),
              type='response')
y1 <- ifelse(y1 > 0.5, 1, 0)
print("results:")
print(y1)
```

```
[1] "results:"
1
0
```

In [ ]: