FIT5197 2018 S1 Assignment 2

Chuangfu Xie, 27771539

22/05/2018

1. Task A

A.1. Handle MVs

First, let's load data from the current working directory, then we have a peek at the data.

```
data_A <- read.csv('./auto_mpg_train.csv', sep = ',')
dim(data_A)
```

```
## [1] 348   9
```

By following the instruction, there are some missing value (MV) listed as `'?'` , let's find out these MVs at which col and how many of them:

```
check_MVs <- function(df){
    for (i in c(1:ncol(df))){
        MV_count = sum(df[i]=='?')
        if (MV_count!=0) print(paste("Col:",i,", Mv:",MV_count))
}}
check_MVs(data_A)
```

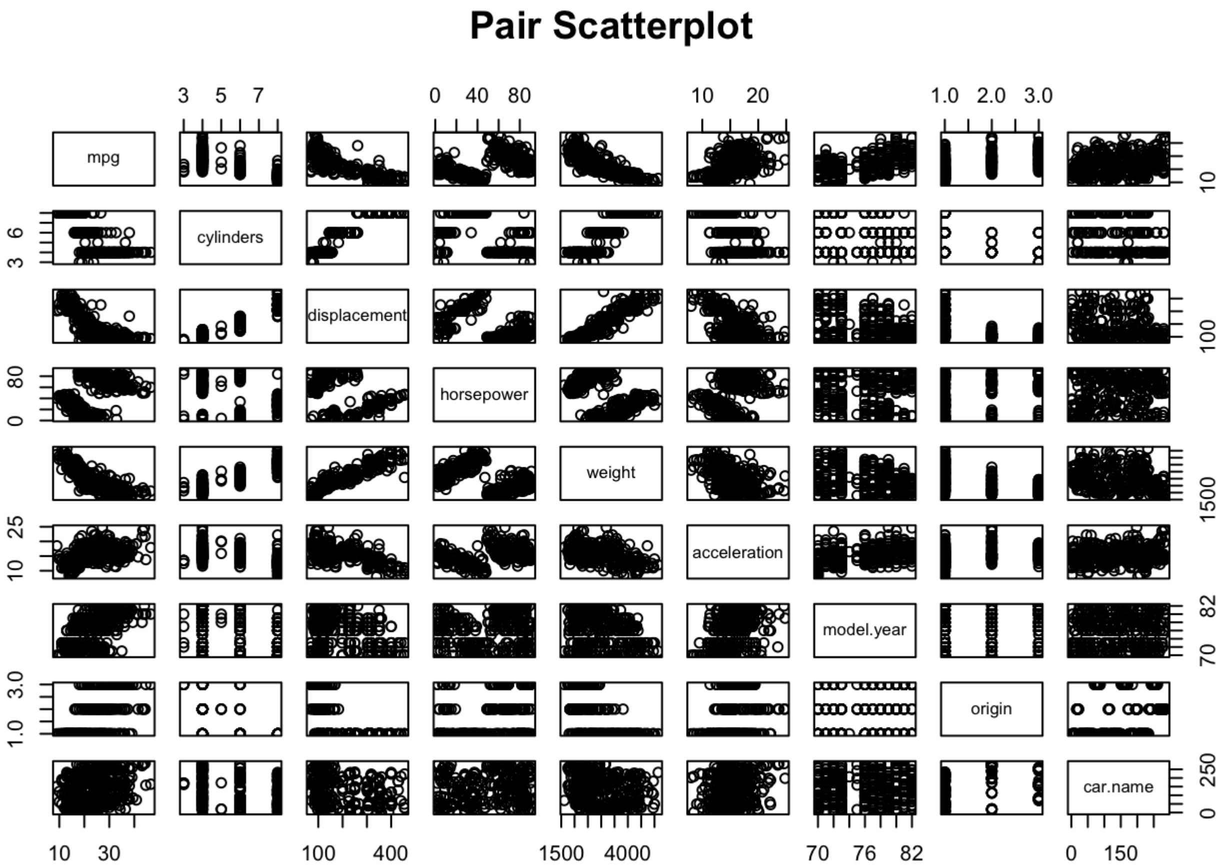
```
## [1] "Col: 4 , Mv: 6"
```

As there are only 6 row containing MVs ($\frac{6}{348} = 1.72\%$), we just drop them since train data cannot contains any MVs:

```
data_A <- subset(data_A, horsepower!='?')
write.csv(data_A, file = "auto_mpg_train_modified.csv")
check_MVs(data_A)
```

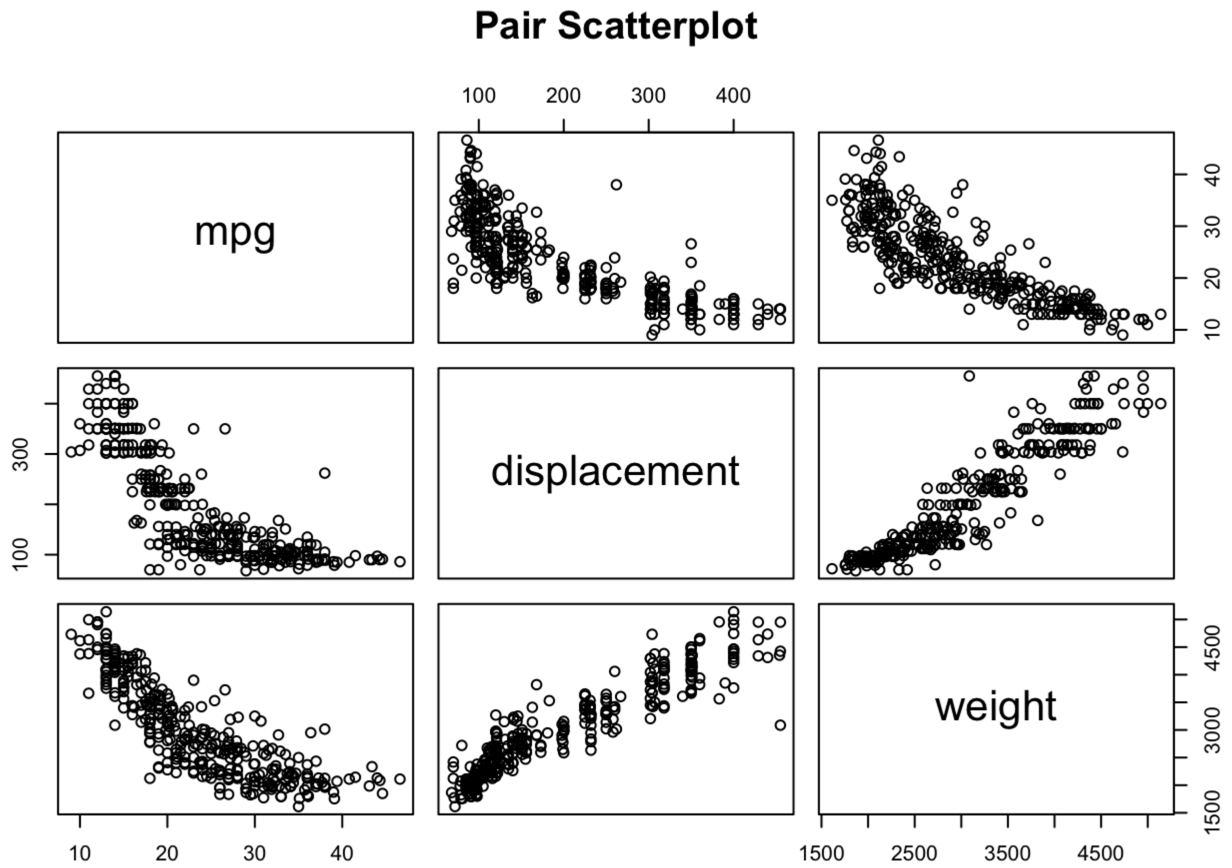
A.2 Visualisation

```
pairs(~mpg+cylinders+displacement+horsepower+weight+acceleration+model.year+origin+ca
r.name,
      data=data_A,
   main="Pair Scatterplot")
```



From above, we can clearly find that only the following columns are relative to **mpg**: **displacement** and **weight.** Since their pair scatter plot shows that their relationship with mpg are nearly linear, we can use it for fitting our linear regression model. Let's have a closer look at these data:

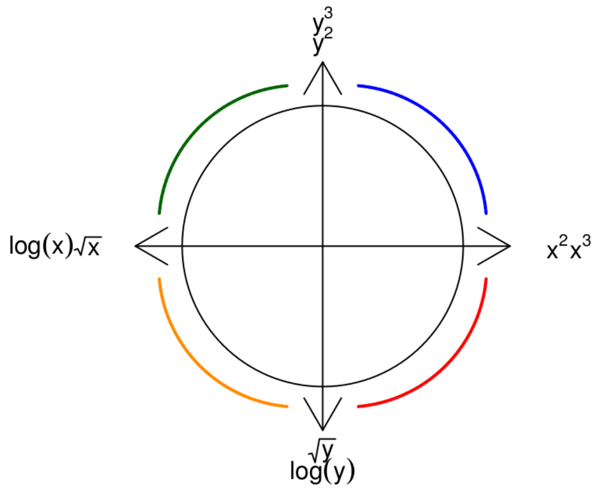
```
pairs(~mpg+displacement+weight,
      data=data_A,
   main="Pair Scatterplot")
```



A.3 Preprocessiong

Since their relation are nearly linear, we need to employ **power transformation** to improve their linearity beforehand.

According to **Tukey's Ladder of Powers**:

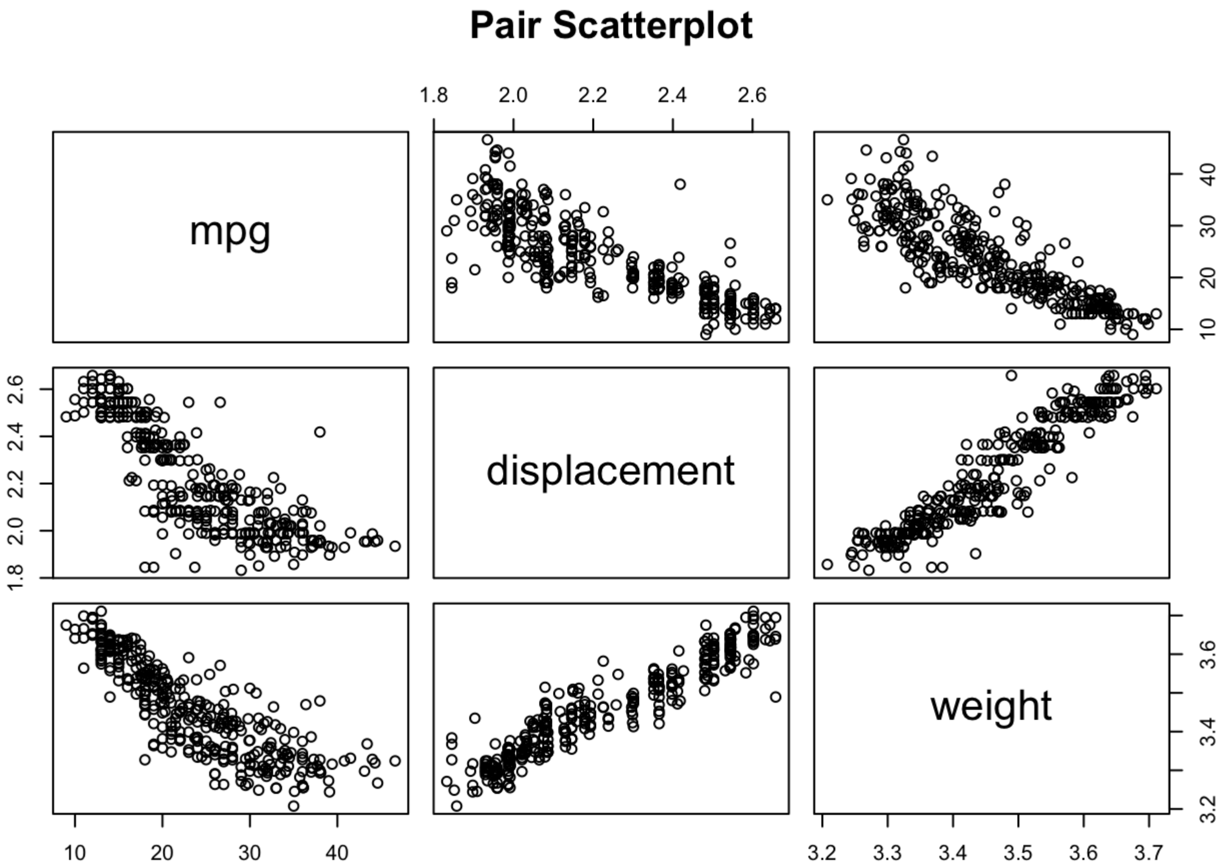


We should apply $log_{10}(x)$ on **displacement** and **weight**:

```
data_A['displacement'] <- lapply(data_A['displacement'], function(x){log(x,10)})
data_A['weight'] <- lapply(data_A['weight'], function(x){log(x,10)})
```

Let's check their linearity:

```
pairs(~mpg+displacement+weight,
      data=data_A,
   main="Pair Scatterplot")
```



Now we have linearise these data, we can extract from **mpg**, **displacement** and **weight** to form a new train dataset for our linear regression model:

```
train_A <- data_A[,c('mpg','displacement','weight'),]
```

A.4 Fitting

```
model <-lm(mpg~displacement+weight, train_A)
summary(model)
```

```
##
## Call:
## lm(formula = mpg ~ displacement + weight, data = train_A)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.2755  -2.7433  -0.1921   2.1148  16.9611
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   178.328     13.447  13.262  < 2e-16 ***
## displacement   -9.325      2.935  -3.177  0.00163 **
## weight        -38.726      5.617  -6.894 2.65e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.268 on 339 degrees of freedom
## Multiple R-squared:  0.7167, Adjusted R-squared:  0.715
## F-statistic: 428.8 on 2 and 339 DF,  p-value: < 2.2e-16
```

```
Call:
lm(formula = mpg ~ displacement + weight, data = train_A)
```

The first part of the output just shows the arguments we just call.

```
Residuals:
     Min       1Q   Median       3Q      Max
-14.2755  -2.7433  -0.1921   2.1148  16.9611
```

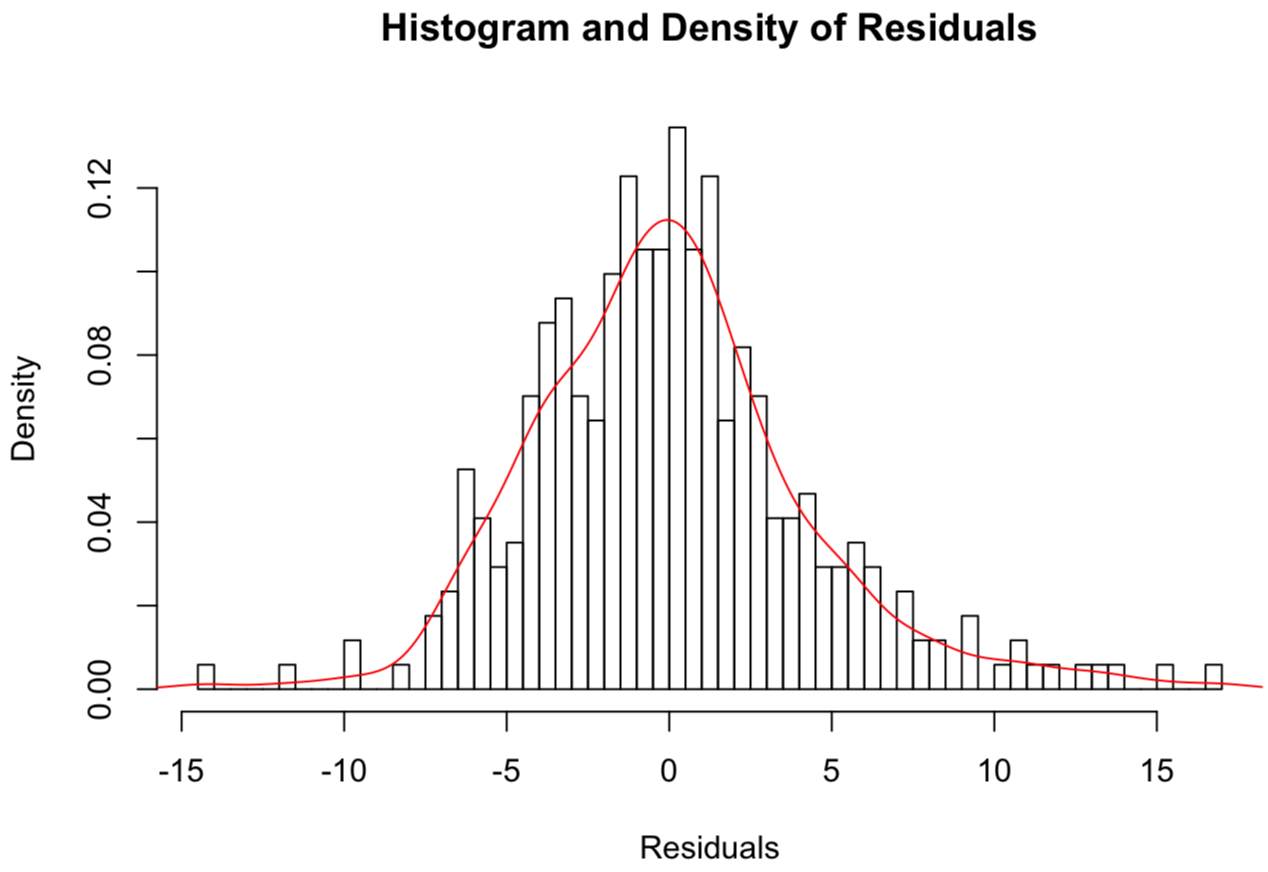
The second part show the quartiles of the residuals. It helps us to identify wether the residual look normally distributed around zero or not. The meadian of the residuals should be close to 0 (`-0.1921`), and the abolute value of 1Q and 3Q should be close (`2.7433` v.s `2.1148`). It seems that our model is fairly well.

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   178.328     13.447  13.262  < 2e-16 ***
displacement   -9.325      2.935  -3.177  0.00163 **
weight        -38.726      5.617  -6.894 2.65e-11 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

The third part is about the coefficients ($\hat{\beta}$). Here is some point to understand these number: 1. Intercept ($\hat{\beta}_0$) and slope($\hat{\beta}_i$). We can use these value to plot the regression line. 2. The standard error measure all estimated coefficients’ variability. The lower the error and better the fit. 3. t-values are not imformative, but we can use it for p-values. The lesser the p-value, the more descriptive the predictor variable is. 4. The last line also gives the idea of the significance of relevance. The more star you get, the more relevant the variable is.

Also, we can find that it provides R^2 score for checking the goodness of fitting. **0.7167** is high, that means our model fit well with the train dataset. However, we shouldn’t so sure before checking the residual plot. let’s check:

```
hist(model$residuals,
     breaks = 50,
     main = 'Histogram and Density of Residuals',
     xlab='Residuals',
     probability=T)
lines(density(model$residuals),
      col = 'red'
     )
```



The above plot confirms that our linear model fitted to the dataset as both histogram and density curve show a nearly normal distribution of residuals.

A.5 Predict values and evaluate model

```
data_A_test <- read.csv('./auto_mpg_test.csv',sep = ',')
data_A_test['displacement'] <- lapply(data_A_test['displacement'], function(x){log(x,
10)})
data_A_test['weight'] <- lapply(data_A_test['weight'], function(x){log(x,10)})
test_A <- data_A_test[, c('displacement','weight'),]
```

Now, we calculate the Mean Standard error on our test data, since we have the formula:

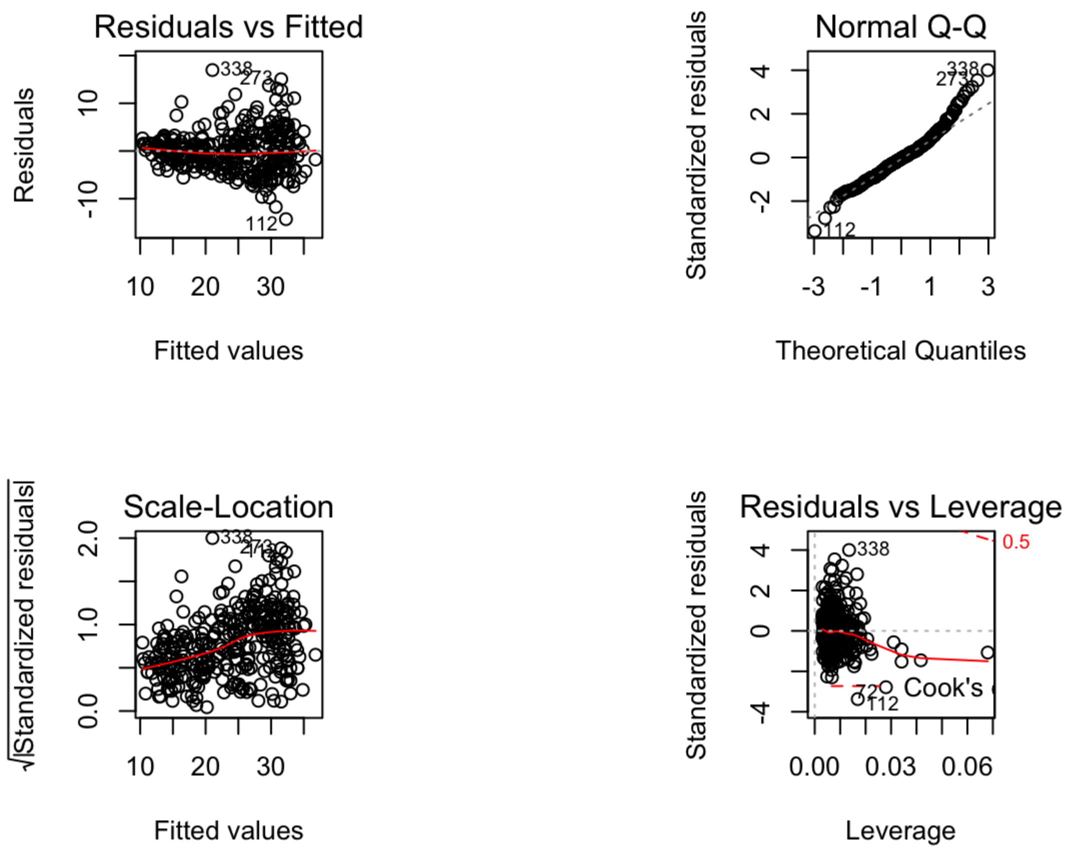
$$MSE = \frac{1}{N} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

```
predict_data <- predict(model, newdata = test_A)
mean((data_A_test$mpg - predict_data)**2)
```

```
## [1] 9.613281
```

The mean squared error tells us how close the regression line is to a set of points. The smaller the means squared error, the closer you are to finding the line of best fit. However, the interpretation depends on the pattern of the scatter plot. Then we use another method to evaluate our mode:

```
par(mfrow=c(2,2), pty = 's')
plot(model)
```



Here we can find that in **Residuals *versus* Fitted** plot, horizontal trend line in the plot **indicates a linear pattern between response and predictors**. For a good fit, residuals should be almost evenly distributed around zero line without any visible pattern. In this case, our model may suffer from the effect of outliers.

Task B

B.1 Handle MVs

First, let's load data from the current working directory, then we have a peek at the data.

```
data_B <- read.csv('./adult_income_train.csv', sep = ',')
```

First, let's check how many MVs in this dataset:

```
check_MVs(data_B)
```

```
## [1] "Col: 2 , Mv: 2799"
## [1] "Col: 7 , Mv: 2809"
## [1] "Col: 14 , Mv: 857"
```

Now we have spotted some MVs in **workclass**, **occupation** and **native_country**. let's take a look what kind of value they have:

```
summary(data_B[c(2,7,14)])
```

```
##             workclass                occupation          native_country
##  Private          :30258   Prof-specialty : 5502   United-States:39240
##  Self-emp-not-inc: 3448   Craft-repair   : 5456   ?            :  857
##  Local-gov        : 2810   Exec-managerial: 5410   Mexico       :  844
##  ?                : 2799   Adm-clerical   : 5010   Philippines  :  266
##  State-gov        : 1726   Sales          : 4894   Germany      :  184
##  Self-emp-inc     : 1495   Other-service  : 4388   Canada       :  165
##  (Other)          : 1306   (Other)        :13182   (Other)      : 2286
```

From summary above, we can find that the reason why **workclass**, **occupation** contain MVs: these information might be difficult for people to answer. So does the **native_country**. Hence, we just replace this `"?"` into some proper value as `"Unknown"`, `"Not Given"`.

```
# edit data_B level "?" as "Unknown"
levels(data_B$workclass)[1] <- "Unknown"
# edit level "?" as "Not Given"
levels(data_B$occupation)[1] <- "Not Given"
levels(data_B$native_country)[1] <- "Not Given"
```

B.2

We need to make sure that all train data should not contain any MVs, let's check:

```
levels(data_B$income)[1] <- 0
levels(data_B$income)[2] <- 1
model <- glm(income~., family = binomial, data = data_B)
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(model)
```

```
##
## Call:
## glm(formula = income ~ ., family = binomial, data = data_B)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -5.1131  -0.5027  -0.1823  -0.0336   3.8667
##
## Coefficients: (2 not defined because of singularities)
##                                        Estimate Std. Error z value
## (Intercept)                          -8.983e+00  3.817e-01 -23.534
## age                                   2.493e-02  1.421e-03  17.547
## workclassFederal-gov                  1.199e+00  1.312e-01   9.142
## workclassLocal-gov                    5.072e-01  1.191e-01   4.258
## workclassNever-worked                -8.058e+00  8.524e+01  -0.095
## workclassPrivate                      6.753e-01  1.056e-01   6.396
## workclassSelf-emp-inc                 8.304e-01  1.272e-01   6.528
## workclassSelf-emp-not-inc             1.356e-01  1.160e-01   1.169
## workclassState-gov                    3.109e-01  1.295e-01   2.401
## workclassWithout-pay                 -2.029e-01  7.916e-01  -0.256
## fnlwgt                                7.803e-07  1.473e-07   5.298
## education11th                         4.803e-02  1.834e-01   0.262
## education12th                         4.517e-01  2.274e-01   1.987
## education1st-4th                     -6.381e-01  4.437e-01  -1.438
## education5th-6th                     -3.744e-01  2.842e-01  -1.317
## education7th-8th                     -4.565e-01  2.001e-01  -2.281
## education9th                         -1.979e-01  2.244e-01  -0.882
## educationAssoc-acdm                   1.370e+00  1.525e-01   8.985
## educationAssoc-voc                    1.297e+00  1.473e-01   8.808
## educationBachelors                    1.930e+00  1.368e-01  14.113
## educationDoctorate                    2.871e+00  1.849e-01  15.524
## educationHS-grad                      8.032e-01  1.333e-01   6.027
## educationMasters                      2.265e+00  1.454e-01  15.578
## educationPreschool                   -5.110e+00  3.713e+00  -1.376
## educationProf-school                  2.782e+00  1.739e-01  16.001
## educationSome-college                 1.168e+00  1.352e-01   8.642
## educational_num                              NA         NA      NA
## marital_statusMarried-AF-spouse       2.484e+00  4.762e-01   5.216
## marital_statusMarried-civ-spouse      2.324e+00  2.318e-01  10.028
## marital_statusMarried-spouse-absent   1.221e-01  1.898e-01   0.643
## marital_statusNever-married          -4.299e-01  7.584e-02  -5.668
## marital_statusSeparated              -1.449e-01  1.446e-01  -1.002
## marital_statusWidowed                 8.315e-02  1.355e-01   0.613
## occupationAdm-clerical                9.467e-02  8.477e-02   1.117
## occupationArmed-Forces                3.553e-01  9.076e-01   0.391
## occupationCraft-repair                1.303e-01  7.292e-02   1.787
## occupationExec-managerial             8.604e-01  7.496e-02  11.477
## occupationFarming-fishing            -8.822e-01  1.231e-01  -7.166
## occupationHandlers-cleaners          -6.438e-01  1.247e-01  -5.161
## occupationMachine-op-inspct          -1.953e-01  9.134e-02  -2.138
## occupationOther-service              -7.800e-01  1.071e-01  -7.280
## occupationPriv-house-serv            -2.508e+00  1.007e+00  -2.490
## occupationProf-specialty              6.173e-01  8.039e-02   7.679
## occupationProtective-serv             5.778e-01  1.130e-01   5.115
## occupationSales                       3.531e-01  7.737e-02   4.564
## occupationTech-support                6.917e-01  1.018e-01   6.797
## occupationTransport-moving                   NA         NA      NA
```

```
## relationshipNot-in-family                    5.902e-01  2.294e-01   2.573
## relationshipOther-relative                  -4.475e-01  2.163e-01  -2.069
## relationshipOwn-child                       -5.141e-01  2.249e-01  -2.286
## relationshipUnmarried                        4.186e-01  2.437e-01   1.718
## relationshipWife                             1.207e+00  8.796e-02  13.727
## raceAsian-Pac-Islander                       8.455e-01  2.338e-01   3.616
## raceBlack                                    4.001e-01  2.033e-01   1.968
## raceOther                                    4.883e-01  2.904e-01   1.681
## raceWhite                                    6.173e-01  1.934e-01   3.192
## genderMale                                   7.743e-01  6.793e-02  11.398
## capital_gain                                 3.231e-04  9.041e-06  35.736
## capital_loss                                 6.397e-04  3.199e-05  19.999
## hours_per_week                               2.867e-02  1.382e-03  20.745
## native_countryCambodia                       9.898e-01  5.506e-01   1.798
## native_countryCanada                         6.812e-01  2.420e-01   2.815
## native_countryChina                         -7.025e-01  3.243e-01  -2.167
## native_countryColumbia                      -2.253e+00  7.956e-01  -2.832
## native_countryCuba                           3.318e-01  2.955e-01   1.123
## native_countryDominican-Republic            -1.551e+00  7.610e-01  -2.038
## native_countryEcuador                       -4.960e-01  6.298e-01  -0.788
## native_countryEl-Salvador                   -6.264e-01  4.477e-01  -1.399
## native_countryEngland                        5.152e-01  2.980e-01   1.729
## native_countryFrance                         8.185e-01  4.584e-01   1.786
## native_countryGermany                        2.507e-01  2.520e-01   0.995
## native_countryGreece                        -2.283e-01  4.052e-01  -0.563
## native_countryGuatemala                     -3.188e-01  7.477e-01  -0.426
## native_countryHaiti                          1.927e-01  5.078e-01   0.379
## native_countryHoland-Netherlands            -8.348e+00  3.247e+02  -0.026
## native_countryHonduras                      -1.413e+00  2.105e+00  -0.671
## native_countryHong                          -4.326e-01  5.976e-01  -0.724
## native_countryHungary                        4.144e-01  6.326e-01   0.655
## native_countryIndia                         -2.766e-01  2.836e-01  -0.975
## native_countryIran                           2.927e-01  4.009e-01   0.730
## native_countryIreland                        1.276e+00  5.018e-01   2.542
## native_countryItaly                          7.790e-01  2.991e-01   2.605
## native_countryJamaica                        2.015e-01  4.142e-01   0.486
## native_countryJapan                         -6.937e-02  3.474e-01  -0.200
## native_countryLaos                          -1.304e+00  8.638e-01  -1.510
## native_countryMexico                        -5.924e-01  2.234e-01  -2.651
## native_countryNicaragua                     -9.403e-01  7.831e-01  -1.201
## native_countryOutlying-US(Guam-USVI-etc) -7.466e-01  1.080e+00  -0.692
## native_countryPeru                          -6.493e-01  6.353e-01  -1.022
## native_countryPhilippines                    2.360e-01  2.417e-01   0.976
## native_countryPoland                        -2.304e-02  3.639e-01  -0.063
## native_countryPortugal                       6.013e-01  4.451e-01   1.351
## native_countryPuerto-Rico                   -1.232e-01  3.323e-01  -0.371
## native_countryScotland                      -1.595e-01  7.555e-01  -0.211
## native_countrySouth                         -1.147e+00  3.835e-01  -2.991
## native_countryTaiwan                        -2.788e-02  4.148e-01  -0.067
## native_countryThailand                      -7.829e-01  6.973e-01  -1.123
## native_countryTrinadad&Tobago               -1.156e+00  8.340e-01  -1.386
## native_countryUnited-States                  2.445e-01  1.135e-01   2.155
## native_countryVietnam                       -9.150e-01  5.077e-01  -1.802
## native_countryYugoslavia                     7.909e-01  6.123e-01   1.292
##                                             Pr(>|z|)
## (Intercept)                                  < 2e-16 ***
## age                                          < 2e-16 ***
## workclassFederal-gov                         < 2e-16 ***
```

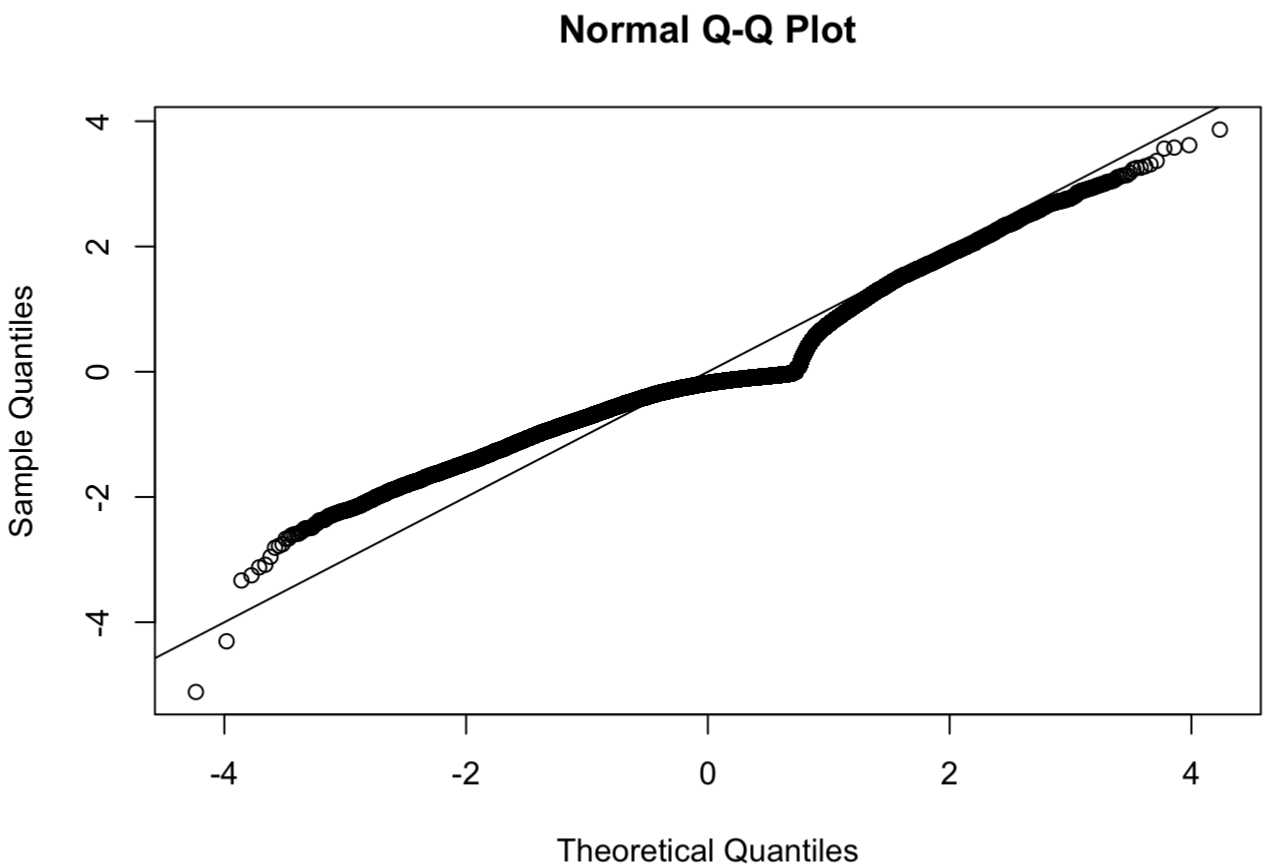
```
## workclassLocal-gov                     2.06e-05 ***
## workclassNever-worked                  0.924684
## workclassPrivate                       1.60e-10 ***
## workclassSelf-emp-inc                  6.68e-11 ***
## workclassSelf-emp-not-inc              0.242303
## workclassState-gov                     0.016340 *
## workclassWithout-pay                   0.797719
## fnlwgt                                 1.17e-07 ***
## education11th                          0.793465
## education12th                          0.046958 *
## education1st-4th                       0.150368
## education5th-6th                       0.187776
## education7th-8th                       0.022567 *
## education9th                           0.377763
## educationAssoc-acdm                     < 2e-16 ***
## educationAssoc-voc                      < 2e-16 ***
## educationBachelors                      < 2e-16 ***
## educationDoctorate                      < 2e-16 ***
## educationHS-grad                       1.67e-09 ***
## educationMasters                        < 2e-16 ***
## educationPreschool                     0.168785
## educationProf-school                    < 2e-16 ***
## educationSome-college                   < 2e-16 ***
## educational_num                              NA
## marital_statusMarried-AF-spouse        1.83e-07 ***
## marital_statusMarried-civ-spouse        < 2e-16 ***
## marital_statusMarried-spouse-absent    0.520216
## marital_statusNever-married            1.44e-08 ***
## marital_statusSeparated                0.316373
## marital_statusWidowed                  0.539548
## occupationAdm-clerical                 0.264077
## occupationArmed-Forces                 0.695457
## occupationCraft-repair                 0.074013 .
## occupationExec-managerial               < 2e-16 ***
## occupationFarming-fishing              7.74e-13 ***
## occupationHandlers-cleaners            2.46e-07 ***
## occupationMachine-op-inspct            0.032506 *
## occupationOther-service                3.34e-13 ***
## occupationPriv-house-serv              0.012767 *
## occupationProf-specialty               1.61e-14 ***
## occupationProtective-serv              3.14e-07 ***
## occupationSales                        5.03e-06 ***
## occupationTech-support                 1.07e-11 ***
## occupationTransport-moving                   NA
## relationshipNot-in-family              0.010078 *
## relationshipOther-relative             0.038592 *
## relationshipOwn-child                  0.022251 *
## relationshipUnmarried                  0.085822 .
## relationshipWife                        < 2e-16 ***
## raceAsian-Pac-Islander                 0.000299 ***
## raceBlack                              0.049111 *
## raceOther                              0.092677 .
## raceWhite                              0.001414 **
## genderMale                              < 2e-16 ***
## capital_gain                            < 2e-16 ***
## capital_loss                            < 2e-16 ***
## hours_per_week                          < 2e-16 ***
## native_countryCambodia                 0.072240 .
```

```
## native_countryCanada                        0.004880 **
## native_countryChina                         0.030269 *
## native_countryColumbia                      0.004622 **
## native_countryCuba                          0.261368
## native_countryDominican-Republic            0.041571 *
## native_countryEcuador                       0.430972
## native_countryEl-Salvador                   0.161737
## native_countryEngland                       0.083770 .
## native_countryFrance                        0.074147 .
## native_countryGermany                       0.319802
## native_countryGreece                        0.573207
## native_countryGuatemala                     0.669817
## native_countryHaiti                         0.704402
## native_countryHoland-Netherlands            0.979492
## native_countryHonduras                      0.501972
## native_countryHong                          0.469151
## native_countryHungary                       0.512433
## native_countryIndia                         0.329360
## native_countryIran                          0.465257
## native_countryIreland                       0.011025 *
## native_countryItaly                         0.009196 **
## native_countryJamaica                       0.626736
## native_countryJapan                         0.841702
## native_countryLaos                          0.131047
## native_countryMexico                        0.008020 **
## native_countryNicaragua                     0.229859
## native_countryOutlying-US(Guam-USVI-etc) 0.489229
## native_countryPeru                          0.306739
## native_countryPhilippines                   0.328968
## native_countryPoland                        0.949509
## native_countryPortugal                      0.176764
## native_countryPuerto-Rico                   0.710764
## native_countryScotland                      0.832756
## native_countrySouth                         0.002779 **
## native_countryTaiwan                        0.946418
## native_countryThailand                      0.261537
## native_countryTrinadad&Tobago               0.165765
## native_countryUnited-States                 0.031189 *
## native_countryVietnam                       0.071501 .
## native_countryYugoslavia                    0.196480
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 48173  on 43841  degrees of freedom
## Residual deviance: 27615  on 43743  degrees of freedom
## AIC: 27813
##
## Number of Fisher Scoring iterations: 11
```

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-5.1131  -0.5027  -0.1823  -0.0336   3.8667
```

The meadian of the residuals close to 0 (-0.1823). The distribution is slightly left tailed as median are more closer to 3Q but not 1Q (0.0336 v.s -0.5027). However, we only need to check its normality if the model is Gaussian. For logistic model, we need to check **Q-Q plot**:

```
qqnorm(residuals(model, type="deviance"))
abline(a=0,b=1)
```



In this plot, dots are nearly over the line `y=x`, we can regard the residuals as normally distributed.

With respect to the coefficients significance, we can find that almost all the coefficient have been given 3 stars. It means that most of the variables are relevent to `income`.

B.3

```
test_B <- read.csv("./adult_income_test.csv")
predict_data <- predict(model, newdata = test_B)
```

```
## Warning in predict.lm(object, newdata, se.fit, scale = 1, type =
## ifelse(type == : prediction from a rank-deficient fit may be misleading
```

```
predict_data[predict_data<0] <- '0'
predict_data[predict_data>0] <- '1'
levels(test_B$income)[1] <- '0'
levels(test_B$income)[2] <- '1'
```

Then, we calculate at what percentage this model correctly predict:

```
match <- sum(predict_data==test_B$income)
notmatch <- sum(predict_data!=test_B$income)
paste(round(match/(match+notmatch)*100,2), "%")
```

```
## [1] "84.86 %"
```

Let’s create a **confusion matrix**:

```
confusion.matrix <- as.matrix(table('Actual'=model$y, 'Prediction'=round(model$fitte
d.values)))
confusion.matrix
```

```
##       Prediction
## Actual     0     1
##      0 31138  2246
##      1  4148  6310
```

From above we can find that: * **TP**=31138 * **FP**=2246 * **FN**=4148 * **TN**=6310

Then we can calculate the **accuracy**:

```
N <- nrow(data_B) # number of observations
diag <- diag(confusion.matrix) # TN and TP
#accuracy = (TP + TN)/N
Accuracy <- sum(diag)/N
paste(round(Accuracy*100,2),"%") # get percentage of accuracy
```

```
## [1] "85.42 %"
```

Also, we can calculate the **Precision** and **Recall**

```
# number of observations per class
rowsums = apply(confusion.matrix, 1, sum)
# number of predictions per class
colsums = apply(confusion.matrix, 2, sum)
# Calculate precision
Precision = diag / colsums
Recall = diag / rowsums
F1 = 2 * Precision * Recall / (Precision + Recall)
round(data.frame(Precision, Recall)*100,2)
```

```
##   Precision Recall
## 0     88.24  93.27
## 1     73.75  60.34
```