Logistic Regression

Assignment:

Study the relationship between age and presence (or absence) of coronary heart disease using logistic regression with 5% significance level.

First, I prepared the environment for this assignment and imported the data using the following commands:

```
> rm(list=ls()) #Clear the environment
> setwd("YOUR_PATH") #Set working directory for the assignment
> getwd() #Check working directory
[1] "YOUR_PATH"
```

Then, I input the data from a csv file, that I prepared in advance, and checked that it loaded correctly:

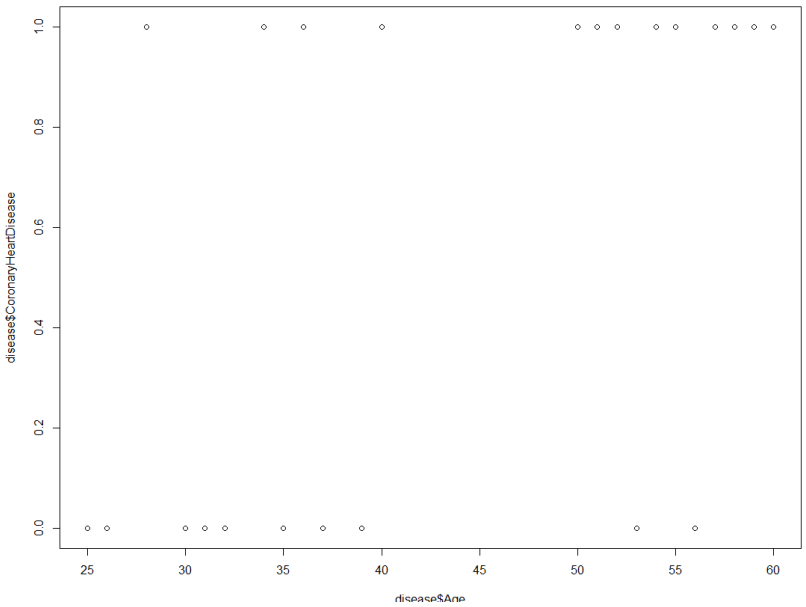
```
> #######Input data from a csv file
>
> disease<-read.csv("heart_disease.csv", header=TRUE)
> disease #Check to make sure it imported correctly
   Patient Age CoronaryHeartDisease
1        1  25                   0
2        2  26                   0
3        3  28                   1
4        4  30                   0
5        5  31                   0
6        6  32                   0
7        7  34                   1
8        8  35                   0
9        9  36                   1
10      10  37                   0
11      11  39                   0
12      12  40                   1
13      13  50                   1
14      14  51                   1
15      15  52                   1
16      16  53                   0
17      17  54                   1
18      18  55                   1
19      19  56                   0
20      20  57                   1
21      21  58                   1
22      22  59                   1
23      23  60                   1
```

I looked at the resulting table structure:

```
> str(disease) #Display internal table structure
'data.frame':  23 obs. of  3 variables:
 $ Patient             : int  1 2 3 4 5 6 7 8 9 10 ...
 $ Age                 : int  25 26 28 30 31 32 34 35 36 37 ...
 $ CoronaryHeartDisease: int  0 0 1 0 0 0 1 0 1 0 ...
```

The data frame contains 23 observations of 3 variables. All columns, including CoronaryHeartDisease, imported as integer type data. It means that we need to convert the CoronaryHeartDisease column to a factor before fitting the model.

```
> #Visually inspect the relationship between variables
> plot(disease$CoronaryHeartDisease ~ disease$Age)
```



While building a logistic regression model, we use the following assumptions:

- Independence – each observation should be independent of other cases. Since we do not have control over data collection in this case, we are going to assume, that this assumption is met.
- Non-linearity – there is no linear relationship between the outcome and the predictor.
- No multicollinearity, - we have just one independent variable in the model.
- No complete separation – We do not have empty cells in the data table or values of zero for the independent variable.

It means that all basic assumptions for the logistical regression hold, and we can proceed with the model.

The **hypothesis** for the logistic regression model:

Ho: the probability of the coronary heart disease is not associated with the age (the coefficient of the Age parameter is zero).

Ha: the probability of the coronary heart disease is associated with the age (the coefficient of the Age parameter is not zero).

The next step is to fit the model.

```
> #Fit the logistic regression model and display results
>
> diseasemodel <- glm(factor(CoronaryHeartDisease) ~ Age, data = disease, family = binomial) #Fit the
model
> summary(diseasemodel) #Display the detailed summary table

Call:
glm(formula = factor(CoronaryHeartDisease) ~ Age, family = binomial,
    data = disease)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.9136  -0.8362   0.5120   0.7284   1.7253

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -4.1212     1.9384  -2.126   0.0335 *
Age           0.1032     0.0451   2.288   0.0222 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 31.492  on 22  degrees of freedom
Residual deviance: 24.844  on 21  degrees of freedom
AIC: 28.844

Number of Fisher Scoring iterations: 4
```


```
> #Fit the logistic regression model and display results
>
> diseasemodel <- glm(factor(CoronaryHeartDisease) ~ Age, data = disease, family = binomial) #Fit the model
> summary(diseasemodel) #Display the detailed summary table

Call:
glm(formula = factor(CoronaryHeartDisease) ~ Age, family = binomial,
    data = disease)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.9136  -0.8362   0.5120   0.7284   1.7253

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -4.1212     1.9384  -2.126   0.0335 *
Age           0.1032     0.0451   2.288   0.0222 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 31.492  on 22  degrees of freedom
Residual deviance: 24.844  on 21  degrees of freedom
AIC: 28.844

Number of Fisher Scoring iterations: 4
```


The above output shows the estimated coefficient for the Age variable of 0.1032. It means that for every additional year of age a patient’s log odds of having coronary heart disease increase by 0.1032 . Since

exp(0.1032) is equal to 1.108713, every additional year of age increase the odds of having coronary heart disease by approximately 1.11 times compared to previous year.

The test returned a Z-value for Age of 2.288 with associated p-value of 0.0222. The p-value is smaller than the significance level of 0.05, it means that we can **reject the null hypothesis** for our model, stating that there is no association between the probability of the coronary heart disease and age, in favor of the alternative hypothesis. The alternative hypothesis states that there is in fact a statistically significant link between the age and presence (or absence) of the coronary heart disease.

Below are the confidence intervals for the coefficient estimate that are based on the log-likelihood function:

```
> confint(diseasemodel) #confidence intervals
Waiting for profiling to be done...
                 2.5 %     97.5 %
(Intercept) -8.43472179 -0.6259744
Age          0.02308053  0.2054780
> exp(diseasemodel$coefficients) #exponentiated coefficients
(Intercept)         Age
 0.01622575  1.10867748
> exp(confint(diseasemodel)) #95% CI for exponentiated coefficeoents
Waiting for profiling to be done...
                  2.5 %    97.5 %
(Intercept) 0.0002171935 0.5347401
Age         1.0233489474 1.2281120
```

Next, we can use anova() to compare the null model with our fitted diseasemodel.

Hypothesis:

Ho: there is no difference in likelihood between the fitted model and the null model.

Ha: There is an improvement in likelihood between the fitted model and the null model.

```
> #Compare the null model with the diseasemodel
> anova(diseasemodel, test='Chisq')
Analysis of Deviance Table

Model: binomial, link: logit

Response: factor(CoronaryHeartDisease)

Terms added sequentially (first to last)


     Df Deviance Resid. Df Resid. Dev Pr(>Chi)
NULL                    22     31.492
Age   1   6.6482        21     24.844 0.009926 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
> #Compare the null model with the diseasemodel
> anova(diseasemodel, test='Chisq')
Analysis of Deviance Table

Model: binomial, link: logit

Response: factor(CoronaryHeartDisease)

Terms added sequentially (first to last)


     Df Deviance Resid. Df Resid. Dev Pr(>Chi)
NULL                    22     31.492
Age   1   6.6482        21     24.844 0.009926 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

This test evaluates deviance as a goodness-of-fit statistic for the model. In our case, deviance measures the discrepancy between the fitted model with the Age variable and the null model excluding Age. The above results show the deviance decreased from 31.492 (for the constant only model) to 24.844 (for our fitted model) when adding the Age variable meaning that the degree of error in prediction decreased. The low p-value of 0.009926 associated with this test statistic which is smaller than the significance level of 0.05 allows us to reject the null hypothesis in favor of the alternative hypothesis. Adding the Age variable to the model significantly improves the model.

Using the model for predicting probabilities of having coronary heart disease for the ages included in the original data set yielded the following results:

```
> ###Predictions
> #Using the diseasemodel to generate predictions for the original dataset
> predict(diseasemodel, disease, type='response')
        1         2         3         4         5         6         7         8         9        10
11        12
0.1762506 0.1917325 0.2257518 0.2638369 0.2843566 0.3058093 0.3512728 0.3751276 0.3996044 0.4245932
0.4756169 0.5013895
       13        14        15        16        17        18        19        20        21        22
23
0.7383164 0.7577538 0.7761852 0.7935961 0.8099842 0.8253575 0.8397331 0.8531359 0.8655973 0.8771536
0.8878449

>

>
```

```
> ###Predictions
> #Using the diseasemodel to generate predictions for the original dataset
> predict(diseasemodel, disease, type='response')
        1         2         3         4         5         6         7         8         9        10        11        12
0.1762506 0.1917325 0.2257518 0.2638369 0.2843566 0.3058093 0.3512728 0.3751276 0.3996044 0.4245932 0.4756169 0.5013895
       13        14        15        16        17        18        19        20        21        22        23
0.7383164 0.7577538 0.7761852 0.7935961 0.8099842 0.8253575 0.8397331 0.8531359 0.8655973 0.8771536 0.8878449
>
```

For example, a 25 year old has about 17.63 percent probability to have coronary heart disease, a 40 year old has 50. 14% probability, a 50 year old 73.83% probability, and a 60 year old has a 88.78% probability of having the disease.

Next, I generated predictions for ages not included in the original dataset. First, for people older than 60:

```
> #Extend prediction interval to people older than 60
> testdata <- data.frame(Age = 61:65)
> predict(diseasemodel, newdata = testdata, type = 'response')
        1         2         3         4         5
0.8977144 0.9068064 0.9151667 0.9228408 0.9298739
```


```
> #Extend prediction interval to people older than 60
> testdata <- data.frame(Age = 61:65)
> predict(diseasemodel, newdata = testdata, type = 'response')
        1         2         3         4         5
0.8977144 0.9068064 0.9151667 0.9228408 0.9298739
```


According to the model, the probability of having coronary heart disease for this group steadily increases from 89.77% to 92.99%.

The original dataset did not include data for 41-49 year old patients. Using the model, I generated the following predictions for the 40 to 50 year old patients:

```
> #Predict probsbilities of coronary heart disease for 40-50 year olds
> testdata2 <-data.frame(Age=40:50)
> predict(diseasemodel, newdata=testdata2, type='response')
        1         2         3         4         5         6         7         8         9        10
11
0.5013895 0.5271547 0.5527761 0.5781202 0.6030594 0.6274744 0.6512561 0.6743075 0.6965453 0.7179000 0
.7383164
```


```
> #Predict probsbilities of coronary heart disease for 40-50 year olds
> testdata2 <-data.frame(Age=40:50)
> predict(diseasemodel, newdata=testdata2, type='response')
        1         2         3         4         5         6         7         8         9        10        11
0.5013895 0.5271547 0.5527761 0.5781202 0.6030594 0.6274744 0.6512561 0.6743075 0.6965453 0.7179000 0.7383164
>
```


According to the model, during this time frame the probability of having coronary heart disease increases dramatically. A 40 year old has 50.14% chance, a 45 year old has 62.75% chance and a 49 year old reaches 71.79%.

Overall, the model created using the existing training data shows strong association between age and presence of coronary heart disease. However, it might be useful to assess predictive power of the model on a different testing data set. Despite calculated statistical significance, some of the resulting values seem to be somewhat unrealistic. It might be caused by some limitations of the training data set and questions about how representative it is for the target population. For example, there is no datapoints for 27, 29, 38, 41-49 year olds. In addition to age, depending on the goals, it might be beneficial to include some other factors into analysis (ex, gender, diet, lifestyle, etc.)