MVA_Assignment_7

Aman

10/28/2020

Assignment 7 - Linear regression

This document checks the assumptios of Linear regression on the Heart Failure Prediction dataset. We know we have a classification problem at hand and modeling with linear regression would not serve our purpose however we perform a theoretical exercise of checking assumptions, multi-collinearity as well as some other interesting results.

Let us load libraries and data

```r
# clear environment
rm(list = ls())

# defining libraries

library(ggplot2)
library(dplyr)
library(PerformanceAnalytics)
library(data.table)
library(sqldf)
library(nortest)
library(MASS)
library(rpart)
library(class)
library(ISLR)
library(scales)
library(ClustOfVar)
library(GGally)
library(reticulate)
library(ggthemes)
library(RColorBrewer)
library(gridExtra)
library(kableExtra)
library(Hmisc)
library(corrplot)
library(energy)
library(nnet)
library(Hotelling)
library(car)
library(devtools)
library(ggbiplot)
library(factoextra)
library(rgl)
library(FactoMineR)
library(psych)
library(nFactors)
library(scatterplot3d)
library(lmtest)
library(mctest)
```

```r
# reading data
data <- read.csv('/Users/mac/Downloads/heart_failure_clinical_records_dataset.csv')
str(data)
```

```
## 'data.frame':    299 obs. of  13 variables:
##  $ age                     : num  75 55 65 50 65 90 75 60 65 80 ...
##  $ anaemia                 : int  0 0 0 1 1 1 1 1 0 1 ...
##  $ creatinine_phosphokinase: int  582 7861 146 111 160 47 246 315 157 123 ...
##  $ diabetes                : int  0 0 0 0 1 0 0 1 0 0 ...
##  $ ejection_fraction       : int  20 38 20 20 20 40 15 60 65 35 ...
##  $ high_blood_pressure     : int  1 0 0 0 0 1 0 0 0 1 ...
##  $ platelets               : num  265000 263358 162000 210000 327000 ...
##  $ serum_creatinine        : num  1.9 1.1 1.3 1.9 2.7 2.1 1.2 1.1 1.5 9.4 ...
##  $ serum_sodium            : int  130 136 129 137 116 132 137 131 138 133 ...
##  $ sex                     : int  1 1 1 1 0 1 1 1 0 1 ...
##  $ smoking                 : int  0 0 1 0 0 1 0 1 0 1 ...
##  $ time                    : int  4 6 7 7 8 8 10 10 10 10 ...
##  $ DEATH_EVENT             : int  1 1 1 1 1 1 1 1 1 1 ...
```

Fitting a linear regression model

```r
mod <-  lm( DEATH_EVENT ~ age+anaemia+creatinine_phosphokinase+
    diabetes+ejection_fraction+high_blood_pressure+platelets+
    serum_creatinine+serum_sodium+sex+smoking+time, data)
summary(mod)
```

```
## 
## Call:
## lm(formula = DEATH_EVENT ~ age + anaemia + creatinine_phosphokinase + 
##     diabetes + ejection_fraction + high_blood_pressure + platelets + 
##     serum_creatinine + serum_sodium + sex + smoking + time, data = data)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -0.80866 -0.28041 -0.04205  0.24742  0.96983 
## 
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)   
## (Intercept)               1.664e+00  6.954e-01   2.392  0.01738 * 
## age                       5.767e-03  1.867e-03   3.088  0.00221 **
## anaemia                  -2.766e-03  4.438e-02  -0.062  0.95035   
## creatinine_phosphokinase  3.427e-05  2.247e-05   1.525  0.12840   
## diabetes                  1.928e-02  4.410e-02   0.437  0.66236    
## ejection_fraction        -9.834e-03  1.844e-03  -5.333 1.96e-07 ***
## high_blood_pressure      -1.430e-02  4.565e-02  -0.313  0.75438    
## platelets                -8.370e-08  2.208e-07  -0.379  0.70492    
## serum_creatinine          8.527e-02  2.123e-02   4.017 7.54e-05 ***
## serum_sodium             -7.599e-03  5.024e-03  -1.513  0.13149    
## sex                      -6.369e-02  5.108e-02  -1.247  0.21353    
## smoking                  -5.733e-03  5.119e-02  -0.112  0.91091    
## time                     -2.733e-03  2.903e-04  -9.415  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3646 on 286 degrees of freedom
## Multiple R-squared:  0.4168, Adjusted R-squared:  0.3924 
## F-statistic: 17.04 on 12 and 286 DF,  p-value: < 2.2e-16
```

We cannot really interpret the results here.

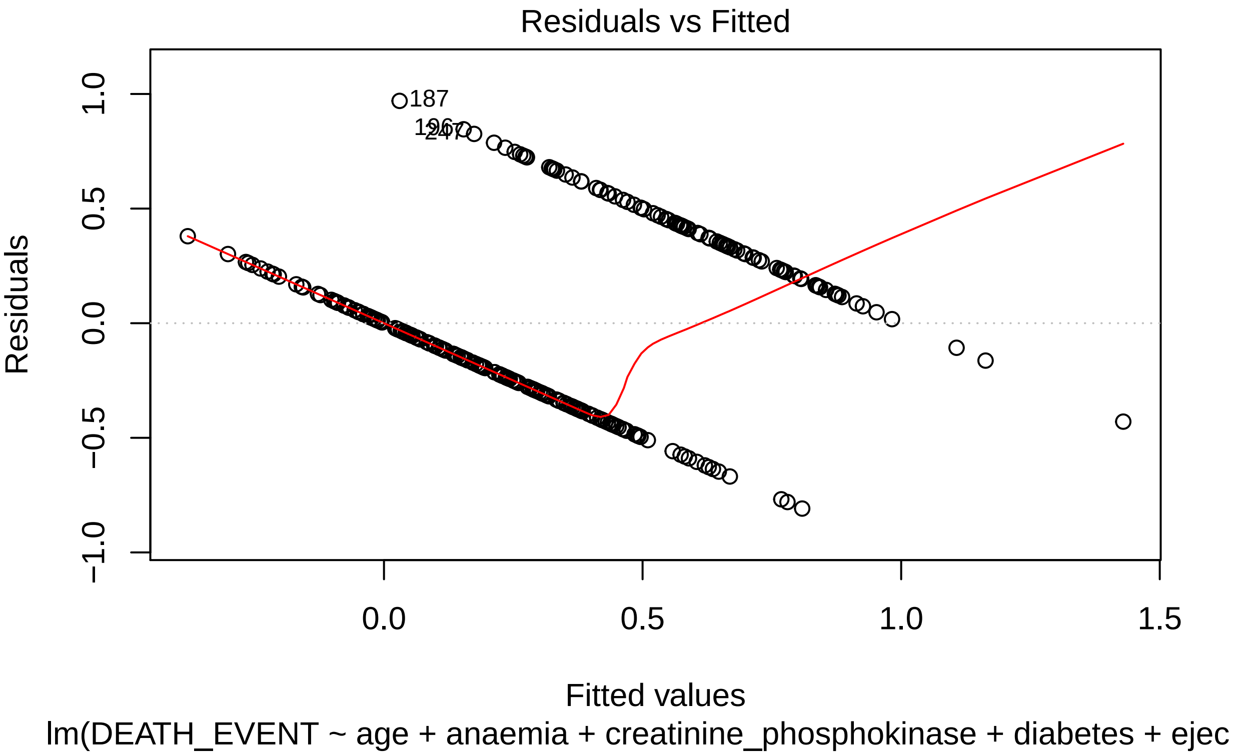
Testing assumptions of linear regression

Recall the assumptions of linear regression as
1. Linear relationship
2. Normality of residuals
3. Homoscedasticity
4. No auto-correlation
5. No or little multicollinearity
6. Normality of the dependent variable.

Linear relationship

The linearity assumption can be checked by inspecting the Residuals vs Fitted plot.

```
plot(mod,which=1)
```

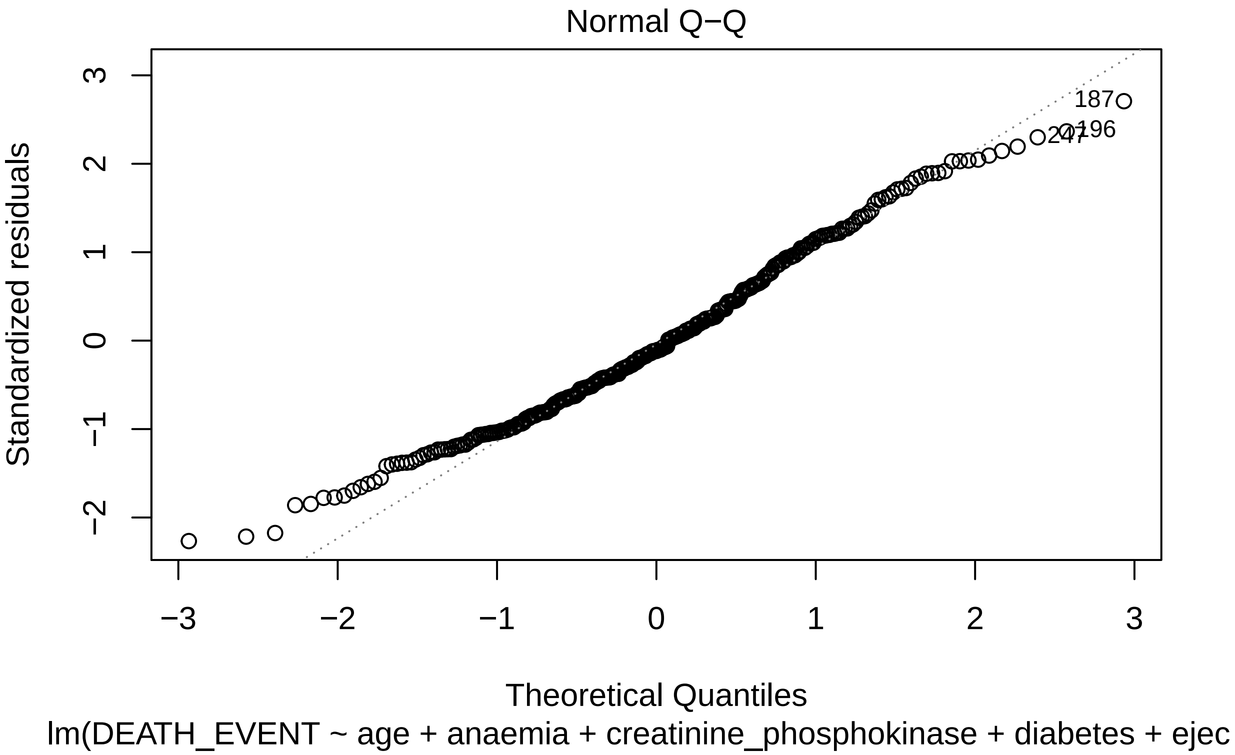


In this plot, we clearly see a pattern for residuals We see them decreasing below 0.5 (fitted values) and increasing above 0.5 (fitted values). This indicates we don't have linear relationship between our dependent and independent variables.

Normality of residuals

The QQ plot of residuals can be used to check the normality assumption. The normal probability plot of residuals should approximately follow a straight line.

```
plot(mod,which=2)
```



Surprisingly we see points falling along reference line however we also see some falling outside so we dig deeper

Shapiro-Wilk Normality Test

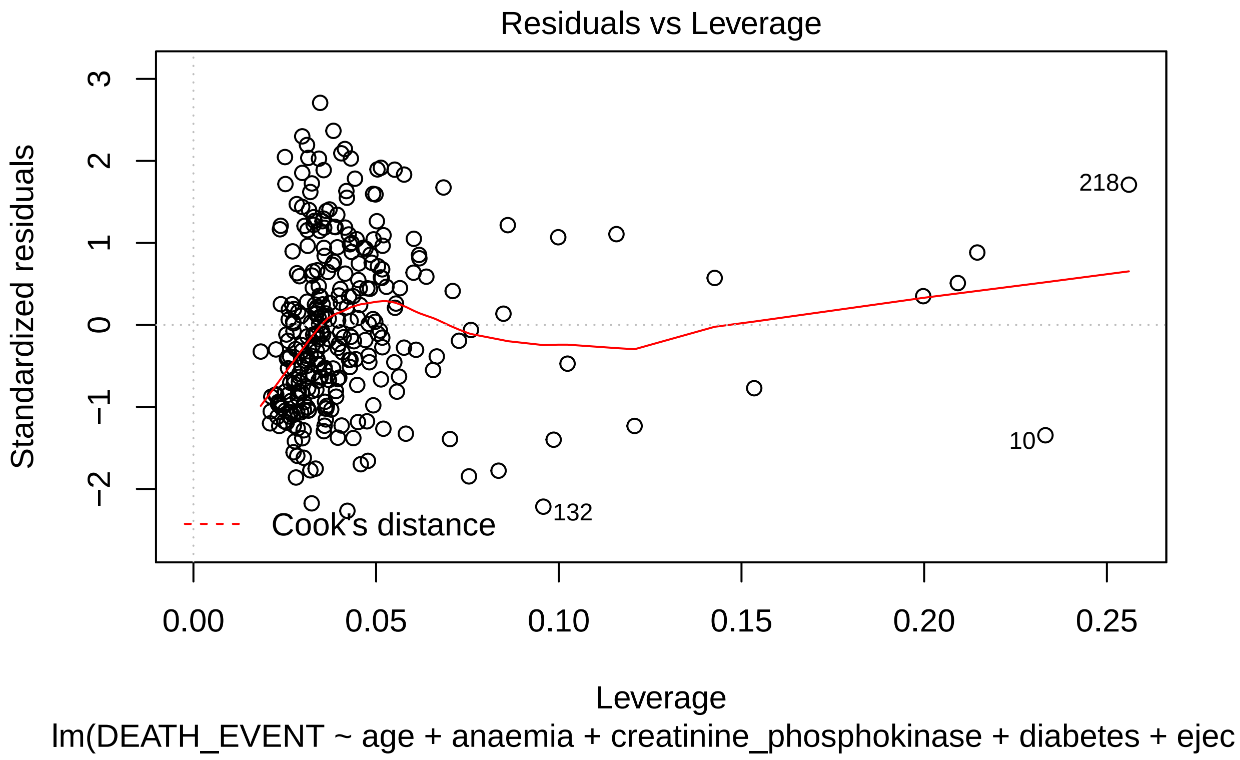
```
resid <- studres(mod)
shapiro.test(resid)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid
## W = 0.98477, p-value = 0.002927
```

From the p-value $= 0.002927 < 0.05$, we can see that the residuals are not normally distributed

High leverage points

```
# High leverage points
plot(mod, which=5)
```



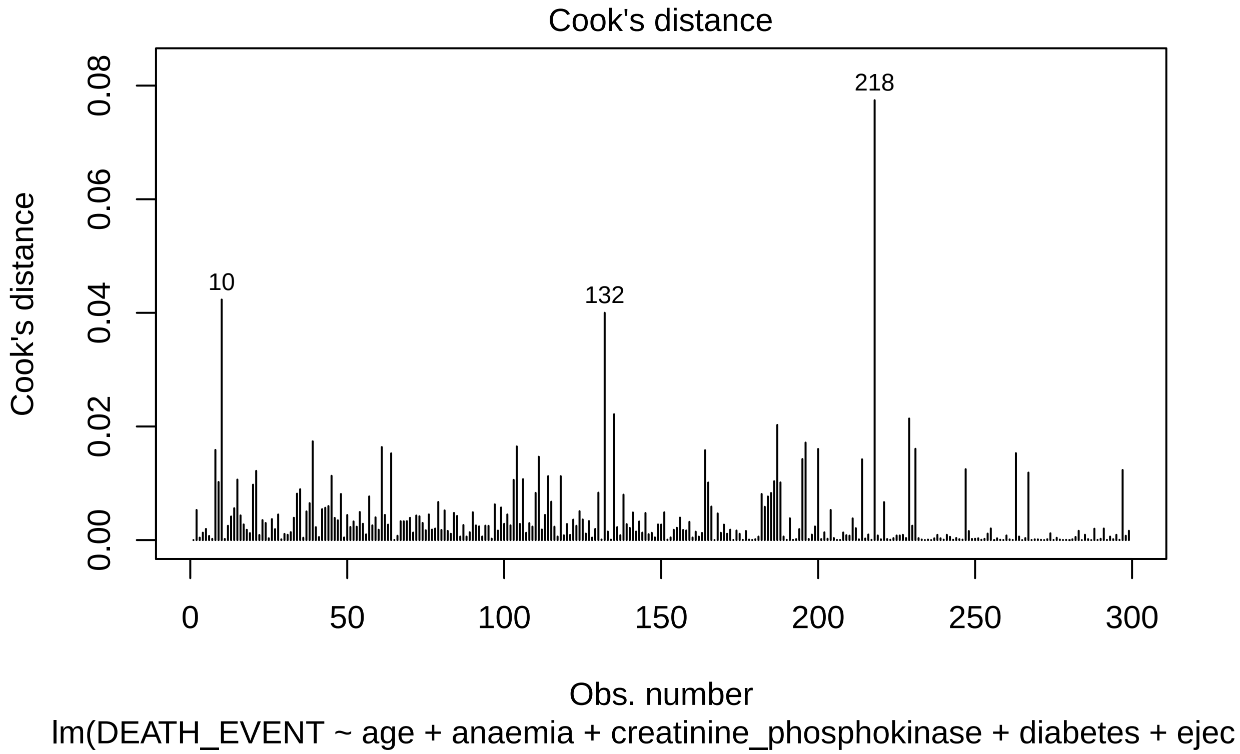
Leverage statistic is defined as -

$\hat{L} = \frac{2(p+1)}{n}$ where p is number of predictors and n is number of observations

So for us $\hat{L} = 0.0869$

Cook's distance

```
#Cook's distance
plot(mod, 4)
```

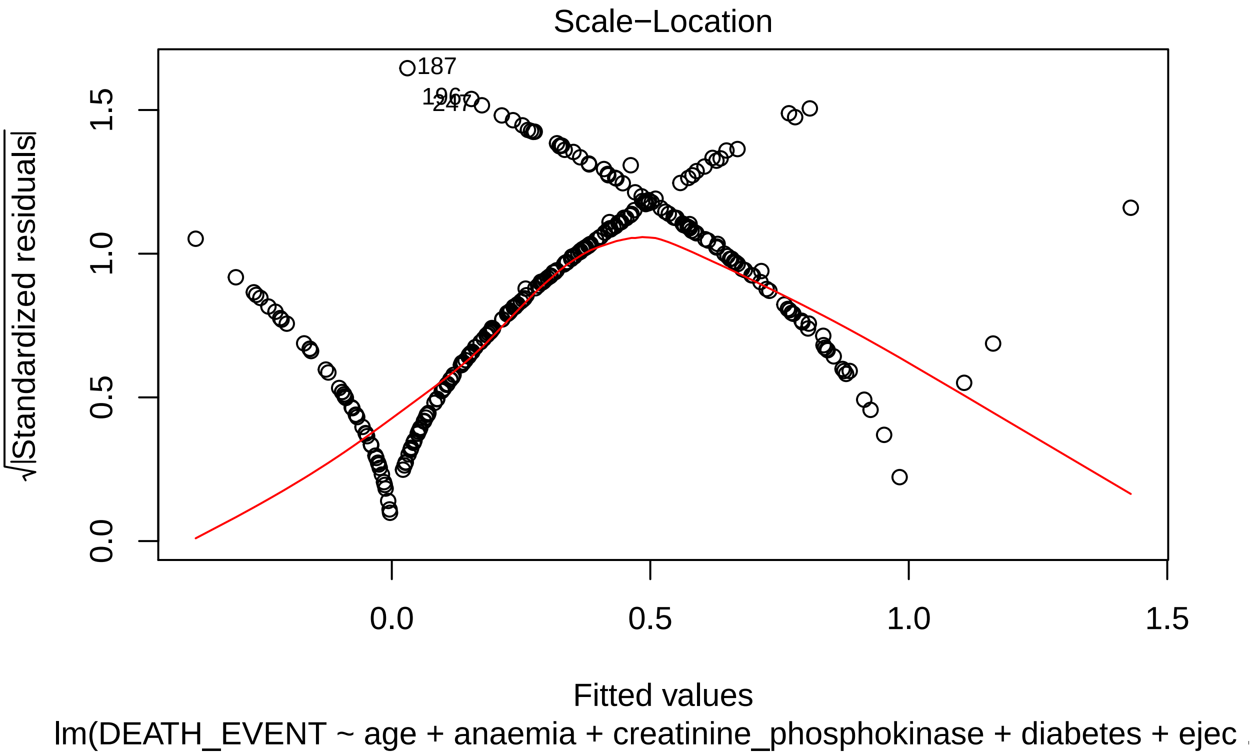


A rule of thumb is that an observation has high influence if Cook's distance exceeds $\frac{4}{(n-p-1)}$

So from the above plots we see cook's plot shows 10, 132, 218 as values of extreme nature and we see no influential points. All points seem to fall under Cook's distance lines (missing dashed lines in residuals vs leverage plot indicates the same)

Homoskedasticity

```
plot(mod,which=3)
```



The spread-location or scale-location plot helps us assess homoskedasticity. We see clearly what is not a horizontal line indicating residuals are not spread equally around the range of fitted values.

ncvTest() For Homoscedasticity

```
ncvTest(mod)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 17.61125, Df = 1, p = 2.7098e-05
```

We see a p-value $<$ .05, indicating that our data is not homoscedastic.

Breusch-Pagan Test For Homoscedasticity

```
bptest(mod)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  mod
## BP = 27.51, df = 12, p-value = 0.00652
```

We see a p-value $<$ .05, indicating that our data is not homoscedastic.

Autocorrelation Assumption

The Durbin Watson examines whether the errors are autocorrelated with themselves. The null states that they are not autocorrelated.

```
durbinWatsonTest(mod)
```

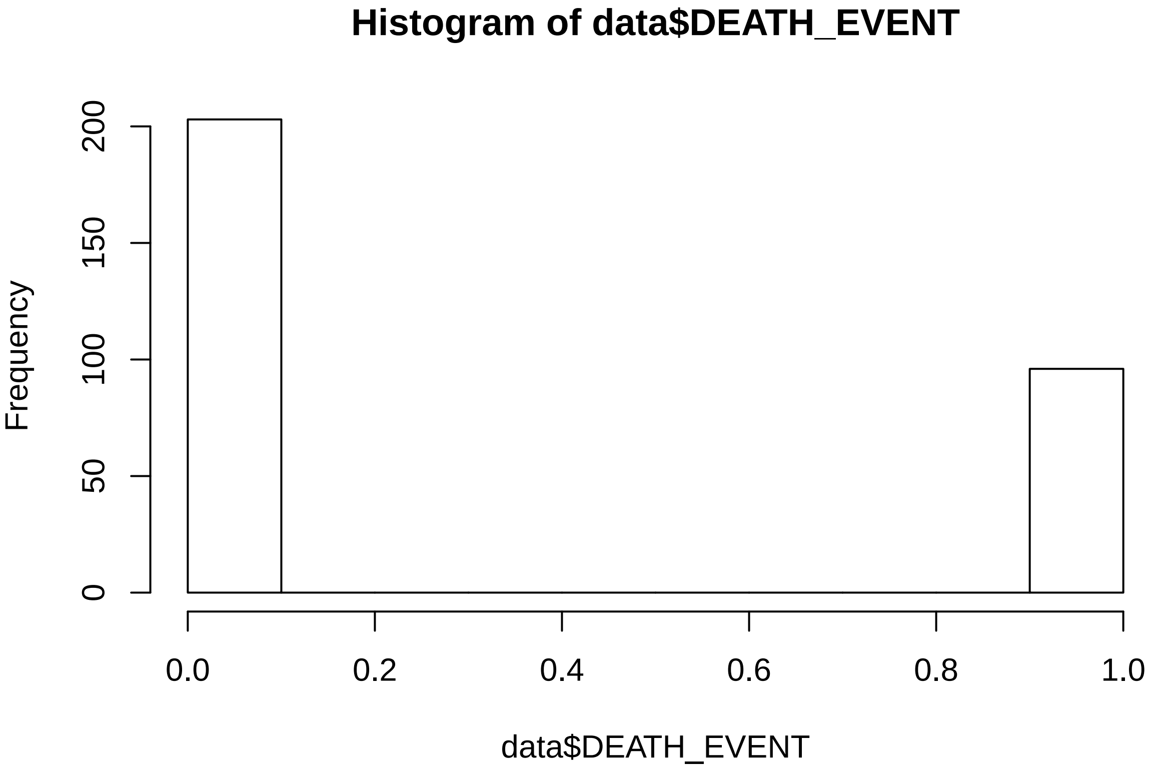
```
##  lag Autocorrelation D-W Statistic p-value
##    1       0.2102177      1.577746       0
##  Alternative hypothesis: rho != 0
```

We see that p-value < 0.05, so the errors are autocorrelated.

Normality of y

We can check the normality of the dependent variable by plotting a histogram.

```
hist(data$DEATH_EVENT)
```



Our histogram doesn't indicate normality of dependent variable.

Assessing multicollinearity

VIF method

```
vif(mod)
```

```
##                      age                  anaemia creatinine_phosphokinase 
##                 1.106067                 1.087163                 1.066014 
##                 diabetes        ejection_fraction      high_blood_pressure 
##                 1.064324                 1.067758                 1.068377 
##                platelets         serum_creatinine             serum_sodium 
##                 1.045809                 1.081241                 1.101927 
##                      sex                  smoking                     time 
##                 1.337716                 1.285049                 1.138009
```

We note that VIF is below 2 for all independent variables indicating there is no multi-collinearity problem in our data.

Note: We see our dependent variable isn't ideal for Linear regression and requires classification techniques to model the same. We do test for assumptions however above and see most of them failing except for little multi-collinearity in our data.

This concludes our approach to Linear regression in our dataset