Project Report

Dataset: Heart Failure

Group 3 | DANA 4820

AIM: Predict Mortality Based on Different Factors in case of Heart Failure

Bhargav Kakadiya : 100350875
Gaurav : 100350679
Guneesh Bhatia : 100346420
Ronak Batra : 100348475
Soniya Gosavi : 100343274

The report is based on the "**Heart Failure**" Dataset. The "Heart Failure" Dataset is taken from UCI Machine Learning Repository. The current version of the dataset was elaborated by Davide Chicco (Krembil Research Institute, Toronto, Canada) and donated to the University of California Irvine Machine Learning Repository under the same Attribution 4.0 International (CC BY 4.0) copyright in January 2020. This dataset contains the medical records of 299 patients who had heart failure, collected during their follow-up period, where each patient profile has 13 clinical features. The Dataset Description is as follows:

Data Set characteristics	*Multivariate*
Number Of Instances	*299*
Area	*Life*
Number Of Attributes	*13*
Attribute Characteristics	*Integer, Real*
Missing Values	*N/A*

The description of 13 clinical features (variables) is as follows:

Variable Name	Description	Type	Units
Age	*Age of the paitent*	*Numerical*	*Years*
Anemia	*Decrease in the number of blood cells*	*Categorical*	*Boolean*
High Blood Pressure	*Determines whether paitent has hypertension or not*	*Categorical*	*Boolean*
Creatinine phosphokinase (CPK)	*Level of CPK enzyme in blood*	*Numerical*	*Mcg/L*
Diabetes	*Determines whether paitent has diabetes or not*	*Categorical*	*Boolean*
Ejection fraction	*percentage of blood leaving the heart at each contraction*	*Numerical*	*Percentage*
Platelets	*Platelets in blood*	*Numerical*	*Kiloplatelets/mL*
Sex	*Woman or Male*	*Categorical*	*Binary*
Serum creatinine	*Level of serum creatinine in blood*	*Numerical*	*Mg/dL*
Serum sodium	*Level of serum sodium in blood*	*Numerical*	*mEq/L*
Smoking	*Determines whether paitent smokes or not*	*Categorical*	*Boolean*
Time	*Follow Up Period*	*Numerical*	*Days*
Death Event	*Determines whether paitent was deceased during follow up period*	*Categorical*	*Boolean*

The link for the dataset is: https://archive.ics.uci.edu/ml/datasets/Heart+failure+clinical+records

1. Loading Dataset and libraries:

The Data file has a '.csv' extension. We have used the read.csv function to load the dataset into "RStudio". Also, header = True indicates that the first row of values in the .csv is set as header information (column names). We are loading libraries "dplyr", "ggpubr" and "caTOOLs". The detailed information for the following libraries is as follows:

- Dplyr: It is a grammar of data manipulation, providing a consistent set of verbs that help you solve the most common data manipulation challenges.

- Ggpubr: It is a data visualization library which facilitates the creation of beautiful ggplot2-based graphs.

- caTools: It contains several basic utility functions including moving (rolling, running) window statistic functions, read/write for GIF and ENVI binary files, fast calculation of AUC, etc.

2. Accuracy Issues and Missing Values:

"Str" function is being used to check the structure of the dataset.

```
> str(df)
'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 ...
```

This function provides us a brief overview of the dataset. The structure clearly depicts that we have columns with numeric and integer values in the dataset.
Then, we are using the "summary" function to get the summary of all columns and NA values.

The summary has the following structure:

MIN	1st Quartile	Meidian	Mean	3rd Quartile	MAX

25% values 75% values

```
> summary(df)
      age            anaemia       creatinine_phosphokinase    diabetes      ejection_fraction
 Min.   :40.00   Min.   :0.0000   Min.   :  23.0           Min.   :0.0000   Min.   :14.00
 1st Qu.:51.00   1st Qu.:0.0000   1st Qu.: 116.5           1st Qu.:0.0000   1st Qu.:30.00
 Median :60.00   Median :0.0000   Median : 250.0           Median :0.0000   Median :38.00
 Mean   :60.83   Mean   :0.4314   Mean   : 581.8           Mean   :0.4181   Mean   :38.08
 3rd Qu.:70.00   3rd Qu.:1.0000   3rd Qu.: 582.0           3rd Qu.:1.0000   3rd Qu.:45.00
 Max.   :95.00   Max.   :1.0000   Max.   :7861.0           Max.   :1.0000   Max.   :80.00
 high_blood_pressure   platelets      serum_creatinine  serum_sodium        sex
 Min.   :0.0000      Min.   : 25100   Min.   :0.500    Min.   :113.0   Min.   :0.0000
 1st Qu.:0.0000      1st Qu.:212500   1st Qu.:0.900    1st Qu.:134.0   1st Qu.:0.0000
 Median :0.0000      Median :262000   Median :1.100    Median :137.0   Median :1.0000
 Mean   :0.3512      Mean   :263358   Mean   :1.394    Mean   :136.6   Mean   :0.6488
 3rd Qu.:1.0000      3rd Qu.:303500   3rd Qu.:1.400    3rd Qu.:140.0   3rd Qu.:1.0000
 Max.   :1.0000      Max.   :850000   Max.   :9.400    Max.   :148.0   Max.   :1.0000
    smoking            time        DEATH_EVENT
 Min.   :0.0000   Min.   :  4.0   Min.   :0.0000
 1st Qu.:0.0000   1st Qu.: 73.0   1st Qu.:0.0000
 Median :0.0000   Median :115.0   Median :0.0000
 Mean   :0.3211   Mean   :130.3   Mean   :0.3211
 3rd Qu.:1.0000   3rd Qu.:203.0   3rd Qu.:1.0000
 Max.   :1.0000   Max.   :285.0   Max.   :1.0000
```

The “head” function gives the first 6 values in each column. If we specify the column name along with the data frame name inside the header function them we’ll get the first six values of the specified column.

```
> head(df)
  age anaemia creatinine_phosphokinase diabetes ejection_fraction high_blood_pressure platelets
1  75       0                     582        0                20                    1    265000
2  55       0                    7861        0                38                    0    263358
3  65       0                     146        0                20                    0    162000
4  50       1                     111        0                20                    0    210000
5  65       1                     160        1                20                    0    327000
6  90       1                      47        0                40                    1    204000
  serum_creatinine serum_sodium sex smoking time DEATH_EVENT
1              1.9          130   1       0    4           1
2              1.1          136   1       0    6           1
3              1.3          129   1       1    7           1
4              1.9          137   1       0    7           1
5              2.7          116   0       0    8           1
6              2.1          132   1       1    8           1
```

The function “colnames” gives us the name of all columns in the data frame.

```
> colnames(df)
 [1] "age"                      "anaemia"                  "creatinine_phosphokinase"
 [4] "diabetes"                 "ejection_fraction"        "high_blood_pressure"
 [7] "platelets"                "serum_creatinine"         "serum_sodium"
[10] "sex"                      "smoking"                  "time"
[13] "DEATH_EVENT"
```

The function “colSums(is.na(df))” gives column wise sum of NA values.

```
> colSums(is.na(df))
                      age                   anaemia creatinine_phosphokinase                  diabetes
                        0                         0                         0                         0
       ejection_fraction         high_blood_pressure                 platelets           serum_creatinine
                        0                         0                         0                         0
            serum_sodium                       sex                   smoking                      time
                        0                         0                         0                         0
              DEATH_EVENT
                        0
```

3. Converting Numerical Datatype to Categorical Datatype

We have columns with Datatype as “numeric” or “integer”. That is why we’ll be converting datatypes of columns with binary or boolean values to categorical. In R, the “**as.factor**” function is used to convert the numeric datatype into factorial datatype. Anaemia, diabetes, high_blood_pressure, sex, smoking, and death_event we’ll be converting all the mentioned columns to the categorical data type.

For the age column, we can see that there is one value in decimal. Someone might have included the value by mistake so we’ll be rounding off that value to the nearest integer.

```
> levels(as.factor(df$age))
 [1] "40"     "41"     "42"     "43"     "44"     "45"     "46"     "47"     "48"     "49"
[11] "50"     "51"     "52"     "53"     "54"     "55"     "56"     "57"     "58"     "59"
[21] "60"     "60.667" "61"     "62"     "63"     "64"     "65"     "66"     "67"     "68"
[31] "69"     "70"     "72"     "73"     "75"     "77"     "78"     "79"     "80"     "81"
[41] "82"     "85"     "86"     "87"     "90"     "94"     "95"
> #Round off AGE
> df$age<-round(df$age)
```

4. Significant test (Variance test, Two-sample T-test, Chi-square test)

a. **Age**

Dividing the dataset into the two samples on the basis of Death_Event and performing t-test on age.
Performing F-test to compare the variance of the two samples.
H_0: Variance of two samples are equal
H_a: Variance of two samples are unequal.

```
> var.test(no_death_age, death_age)

        F test to compare two variances

data:  no_death_age and death_age
F = 0.6482, num df = 202, denom df = 95, p-value = 0.01116
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.4533933 0.9066514
sample estimates:
ratio of variances
         0.6481992
```

p = 0.01
=> p < 0.05 (Level of significance)
=> We reject null hypothesis.
=> We conclude that the variance of the two samples are significantly different.

Further for the t-test, we'll be comparing the mean of the two samples.

H_0: $\mu 1 = \mu 2$
H_a: $\mu 1 \neq \mu 2$

```
> t.test(no_death_age, death_age,var.equal = F)

        Welch Two Sample t-test

data:  no_death_age and death_age
t = -4.1876, df = 155.31, p-value = 4.708e-05
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -9.500188 -3.410218
sample estimates:
mean of x mean of y
 58.76355  65.21875
```

** var.equal = F indicates that variance of two samples is not equal**

p = 4.708e-05,
=> p < 0.05 (Level of significance)
=> We reject null hypothesis.
=> We conclude that the mean of the two samples are significantly different.
=> We conclude that **age is a significant variable.**

---

b. Creatinine Phosphokinase

Dividing the dataset into the two samples on the basis of Death_Event and performing t-test on **creatinine_phosphokinase**.
Performing F-test to compare the variance of the two samples.

H_0: Variance of two samples are equal
H_a: Variance of two samples are unequal.

```
> var.test(no_death_phosphokinase, death_phosphokinase)

        F test to compare two variances

data:  no_death_phosphokinase and death_phosphokinase
F = 0.32781, num df = 202, denom df = 95, p-value = 3.354e-11
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.2292891 0.4585098
sample estimates:
ratio of variances
          0.327806
```

p = 3.354e-11
=> p < 0.05 (Level of significance)
=> We reject null hypothesis.
=> We conclude that the variance of the two samples are significantly different.

Further for the t-test, we'll be comparing the mean of the two samples.

H_0: $\mu 1 = \mu 2$
H_a: $\mu 1 \neq \mu 2$

```
> t.test(no_death_phosphokinase, death_phosphokinase,var.equal = F)

        Welch Two Sample t-test

data:  no_death_phosphokinase and death_phosphokinase
t = -0.90119, df = 125.32, p-value = 0.3692
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -415.9482  155.6608
sample estimates:
mean of x mean of y
 540.0542  670.1979
```

** var.equal = F indicates that variance of two samples is not equal**

p = 0.3692,
=> p > 0.05 (Level of significance)
=> We fail to reject null hypothesis.
=> We conclude that the mean of the two samples are not significantly different.
=> We conclude that **creatinine_phosphokinase is not a significant variable**.

---

c. Ejection Fraction

Dividing the dataset into the two samples on the basis of Death_Event and performing t-test on ejection_fraction .
Performing F-test to compare the variance of the two samples.
H_0: Variance of two samples are equal
H_a: Variance of two samples are unequal.

```
> var.test(no_death_ejection_fraction, death_ejection_fraction)

        F test to compare two variances

data:  no_death_ejection_fraction and death_ejection_fraction
F = 0.75176, num df = 202, denom df = 95, p-value = 0.09577
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.5258317 1.0515066
sample estimates:
ratio of variances
         0.7517616
```

p = 0.09577
=> p > 0.05 (Level of significance)
=> We fail to reject null hypothesis.
=> We conclude that the variance of the two samples are not significantly different.

Further for the t-test, we'll be comparing the mean of the two samples.

H_0: $\mu 1 = \mu 2$
H_a: $\mu 1 \neq \mu 2$

```
> t.test(no_death_ejection_fraction, death_ejection_fraction,var.equal = T)

        Two Sample t-test

data:  no_death_ejection_fraction and death_ejection_fraction
t = 4.8056, df = 297, p-value = 2.453e-06
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 4.013671 9.580849
sample estimates:
mean of x mean of y
 40.26601  33.46875
```

** var.equal = T indicates that variance of two samples is equal**

p = 2.453e-06
=> p < 0.05 (Level of significance)
=> We reject null hypothesis.
=> We conclude that the mean of the two samples are significantly different.

=> We conclude that **ejection_fraction is a significant variable.**

---

d. Platelets

Dividing the dataset into the two samples on the basis of Death_Event and performing t-test on platelets .
Performing F-test to compare the variance of the two samples.
H_0: Variance of two samples are equal
H_a: Variance of two samples are unequal.

```
> var.test(no_death_platelets, death_platelets)

        F test to compare two variances

data:  no_death_platelets and death_platelets
F = 0.97991, num df = 202, denom df = 95, p-value = 0.8915
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.6854169 1.3706295
sample estimates:
ratio of variances
         0.9799146
```

p = 0.8915
=> p > 0.05 (Level of significance)
=> We fail to reject null hypothesis.
=> We conclude that the variance of the two samples are not significantly different.

Further for the t-test, we'll be comparing the mean of the two samples.

H_0: $\mu 1 = \mu 2$
H_a: $\mu 1 \neq \mu 2$

```
> t.test(no_death_platelets, death_platelets,var.equal = T)

        Two Sample t-test

data:  no_death_platelets and death_platelets
t = 0.84787, df = 297, p-value = 0.3972
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -13576.17  34129.06
sample estimates:
mean of x mean of y
 266657.5  256381.0
```

** var.equal = T indicates that variance of two samples is equal**

p = 0.3692,
=> p > 0.05 (Level of significance)
=> We fail to reject null hypothesis.
=> We conclude that the mean of the two samples are not significantly different.
=> We conclude that **platelets is not a significant variable**.

---

e. Serum Creatinine

Dividing the dataset into the two samples on the basis of Death_Event and performing t-test on serum_creatinine.
Performing F-test to compare the variance of the two samples.
H_0: Variance of two samples are equal
H_a: Variance of two samples are unequal.

```
> var.test(no_death_serum_creatinine, death_serum_creatinine)

        F test to compare two variances

data:  no_death_serum_creatinine and death_serum_creatinine
F = 0.19837, num df = 202, denom df = 95, p-value < 2.2e-16
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.1387546 0.2774679
sample estimates:
ratio of variances
         0.1983723
```

p = 2.2e-16
=> p < 0.05 (Level of significance)
=> We reject null hypothesis.
=> We conclude that the variance of the two samples are significantly different.

Further for the t-test, we'll be comparing the mean of the two samples.

H_0: $\mu 1 = \mu 2$
H_a: $\mu 1 \neq \mu 2$

```
> t.test(no_death_serum_creatinine, death_serum_creatinine,var.equal = F)

        Welch Two Sample t-test

data:  no_death_serum_creatinine and death_serum_creatinine
t = -4.1526, df = 113.19, p-value = 6.399e-05
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.9615153 -0.3403977
sample estimates:
mean of x mean of y
 1.184877  1.835833
```

** var.equal = F indicates that variance of two samples is not equal**

p = 2.453e-06
=> p < 0.05 (Level of significance)
=> We reject null hypothesis.
=> We conclude that the mean of the two samples are significantly different.
=> We conclude that **serum_creatinine is a significant variable.**

---

f. Serum Sodium

Dividing the dataset into the two samples on the basis of Death_Event and performing t-test on serum_sodium.
Performing F-test to compare the variance of the two samples.
H_0: Variance of two samples are equal
H_a: Variance of two samples are unequal.

```
> var.test(no_death_serum_sodium, death_serum_sodium)

        F test to compare two variances

data:  no_death_serum_sodium and death_serum_sodium
F = 0.63415, num df = 202, denom df = 95, p-value = 0.007646
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.4435640 0.8869957
sample estimates:
ratio of variances
         0.6341466
```

p = 0.007646
=> p < 0.05 (Level of significance)
=> We reject null hypothesis.
=> We conclude that the variance of the two samples are significantly different.

Further for the t-test, we'll be comparing the mean of the two samples.

H_0: $\mu 1 = \mu 2$
H_a: $\mu 1 \neq \mu 2$

```
> t.test(no_death_serum_sodium, death_serum_sodium,var.equal = F)

        Welch Two Sample t-test

data:  no_death_serum_sodium and death_serum_sodium
t = 3.1645, df = 154.01, p-value = 0.001872
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.6920096 2.9914879
sample estimates:
mean of x mean of y
 137.2167  135.3750
```

** var.equal = F indicates that variance of two samples is not equal**

p = 0.001872
=> p < 0.05 (Level of significance)
=> We reject null hypothesis.
=> We conclude that the mean of the two samples are significantly different.
=> We conclude that **serum_sodium is a significant variable.**

---

g. Time

Dividing the dataset into the two samples on the basis of Death_Event and performing t-test on time.
Performing F-test to compare the variance of the two samples.
H_0: Variance of two samples are equal
H_a: Variance of two samples are unequal.

```
> var.test(no_death_time, death_time)

        F test to compare two variances

data:  no_death_time and death_time
F = 1.1794, num df = 202, denom df = 95, p-value = 0.3652
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.8249488 1.6496516
sample estimates:
ratio of variances
          1.179398
```

p = 0.3652

=> p > 0.05 (Level of significance)
=> We fail to reject null hypothesis.
=> We conclude that the variance of the two samples are not significantly different.

Further for the t-test, we'll be comparing the mean of the two samples.

H_0: $\mu 1 = \mu 2$
H_a: $\mu 1 \neq \mu 2$

```
> t.test(no_death_time, death_time,var.equal = T)

        Two Sample t-test

data:  no_death_time and death_time
t = 10.686, df = 297, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
  71.3478 103.5612
sample estimates:
mean of x mean of y
158.33990  70.88542
```

** var.equal = F indicates that variance of two samples is equal**

p = 2.2e-16
=> p < 0.05 (Level of significance)
=> We reject null hypothesis.
=> We conclude that the mean of the two samples are significantly different.
=> We conclude that **time is a significant variable.**

***H*0:** The Death Event is independent of the diabetes
***Ha*:** The Death Event is not independent of the diabetes

```
> chi_diabetes_death <- chisq.test(table(df$DEATH_EVENT, df$diabetes))
> chi_diabetes_death

        Pearson's Chi-squared test with Yates' continuity correction

data:  table(df$DEATH_EVENT, df$diabetes)
X-squared = 2.1617e-30, df = 1, p-value = 1
```

As the p-value 1 is greater than the 0.05 significance level, we fail to reject the null hypothesis that the Death Event is not independent of the diabetes.

***H*0**: The Death Event is independent of the anaemia
Ha: The Death Event is not independent of the anaemia

```
> chi_anemia_death <- chisq.test(table(df$DEATH_EVENT, df$anaemia))
> chi_anemia_death

        Pearson's Chi-squared test with Yates' continuity correction

data:  table(df$DEATH_EVENT, df$anaemia)
X-squared = 1.0422, df = 1, p-value = 0.3073
```

As the p-value 0.3073 is greater than the .05 significance level, we do not reject the null hypothesis that the Death Event is not independent of the anaemia.

***H*0**: The Death Event is independent of the smoking
Ha: The Death Event is not independent of the smoking

```
> chi_smoking_death <- chisq.test(table(df$DEATH_EVENT, df$smoking))
> chi_smoking_death

        Pearson's Chi-squared test with Yates' continuity correction

data:  table(df$DEATH_EVENT, df$smoking)
X-squared = 0.0073315, df = 1, p-value = 0.9318
```

As the p-value 0.9318 is greater than the .05 significance level, we do not reject the null hypothesis that the Death Event is independent of smoking.

***H*0**: The Death Event is independent of the high blood pressure
Ha: The Death Event is not independent of the high blood pressure

```
> chi_bloodP_death <- chisq.test(table(df$DEATH_EVENT, df$high_blood_pressure))
> chi_bloodP_death

        Pearson's Chi-squared test with Yates' continuity correction

data:  table(df$DEATH_EVENT, df$high_blood_pressure)
X-squared = 1.5435, df = 1, p-value = 0.2141
```

As the p-value 0.2141 is greater than the .05 significance level, we do not reject the null hypothesis that the Death Event is not independent of the high blood pressure.

***H*0:** The Death Event is independent of the sex
***Ha*:** The Death Event is not independent of the sex

```
> chi_sex_death <- chisq.test(table(df$DEATH_EVENT, df$sex))
> chi_sex_death

        Pearson's Chi-squared test with Yates' continuity correction

data:  table(df$DEATH_EVENT, df$sex)
X-squared = 0, df = 1, p-value = 1
```

As the p-value 1 is greater than the .05 significance level, we do not reject the null hypothesis that the Death Event is not independent of the Sex.

5. Multicollinearity

Multicollinearity occurs when an independent variable is highly correlated with one or more of the other independent variables.
The results parameter estimates are unstable & the standard errors are large.

```
> formula<-glm(DEATH_EVENT~.,family = binomial,data=df)
> model<-formula
> summary(model)

Call:
glm(formula = DEATH_EVENT ~ ., family = binomial, data = df)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1849  -0.5705  -0.2399   0.4465   2.6671

Coefficients:
                           Estimate Std. Error z value Pr(>|z|)
(Intercept)               1.018e+01  5.657e+00   1.800 0.071866 .
age                       4.748e-02  1.580e-02   3.004 0.002661 **
anaemia1                 -7.741e-03  3.605e-01  -0.021 0.982869
creatinine_phosphokinase  2.224e-04  1.780e-04   1.249 0.211510
diabetes1                 1.451e-01  3.512e-01   0.413 0.679503
ejection_fraction        -7.668e-02  1.633e-02  -4.695 2.66e-06 ***
high_blood_pressure1     -1.028e-01  3.587e-01  -0.286 0.774528
platelets                -1.201e-06  1.889e-06  -0.635 0.525106
serum_creatinine          6.661e-01  1.815e-01   3.670 0.000242 ***
serum_sodium             -6.698e-02  3.974e-02  -1.686 0.091882 .
sex1                     -5.340e-01  4.139e-01  -1.290 0.197051
smoking1                 -1.335e-02  4.127e-01  -0.032 0.974182
time                     -2.105e-02  3.015e-03  -6.982 2.91e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 375.35  on 298  degrees of freedom
Residual deviance: 219.53  on 286  degrees of freedom
AIC: 245.53

Number of Fisher Scoring iterations: 6
```

```
> vif_values <-vif(model)
> vif_values
                      age                 anaemia creatinine_phosphokinase                 diabetes
                 1.099757                1.108486                1.087588                1.040799
       ejection_fraction      high_blood_pressure                platelets         serum_creatinine
                 1.174256                1.061194                1.044412                1.080477
            serum_sodium                     sex                  smoking                    time
                 1.058970                1.377590                1.281855                1.133895
```

The above screenshot clearly depicts that there is ni vif(Variance Inflation Factor) value greater than 5. This clearly indicates that there is no multicollinearity issue. So we are good to go.

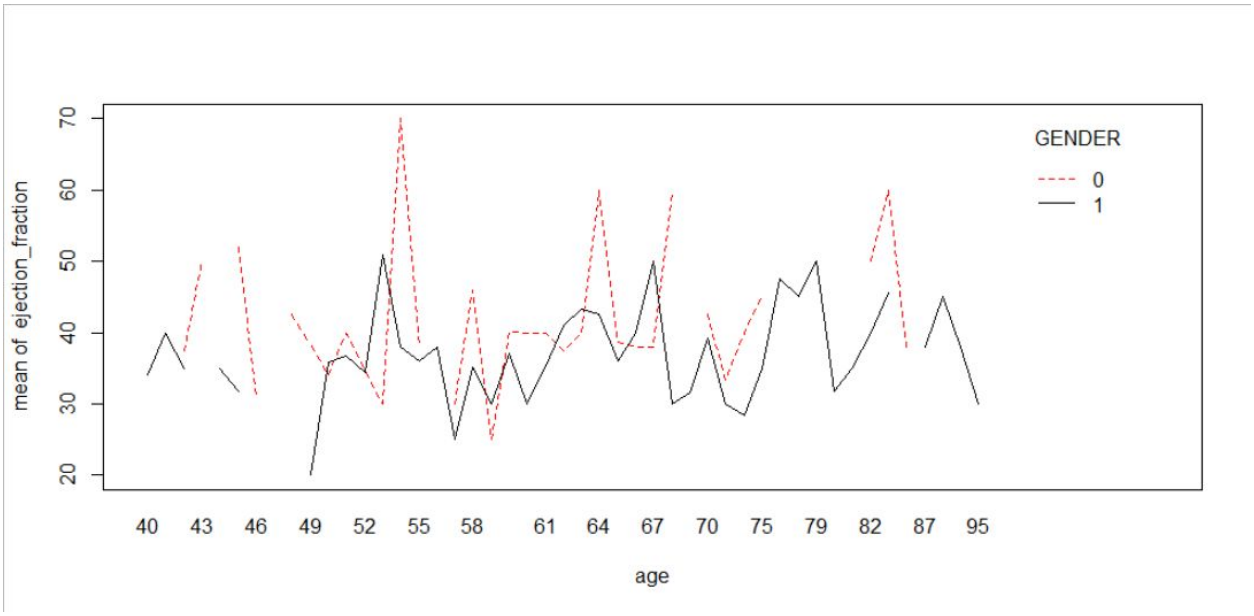
6. Interaction Plot

Interaction plots are used to understand how the value of one variable affects the value of another variable.

Parallel lines - No interaction occurs
Non parallel lines - Interaction occurs

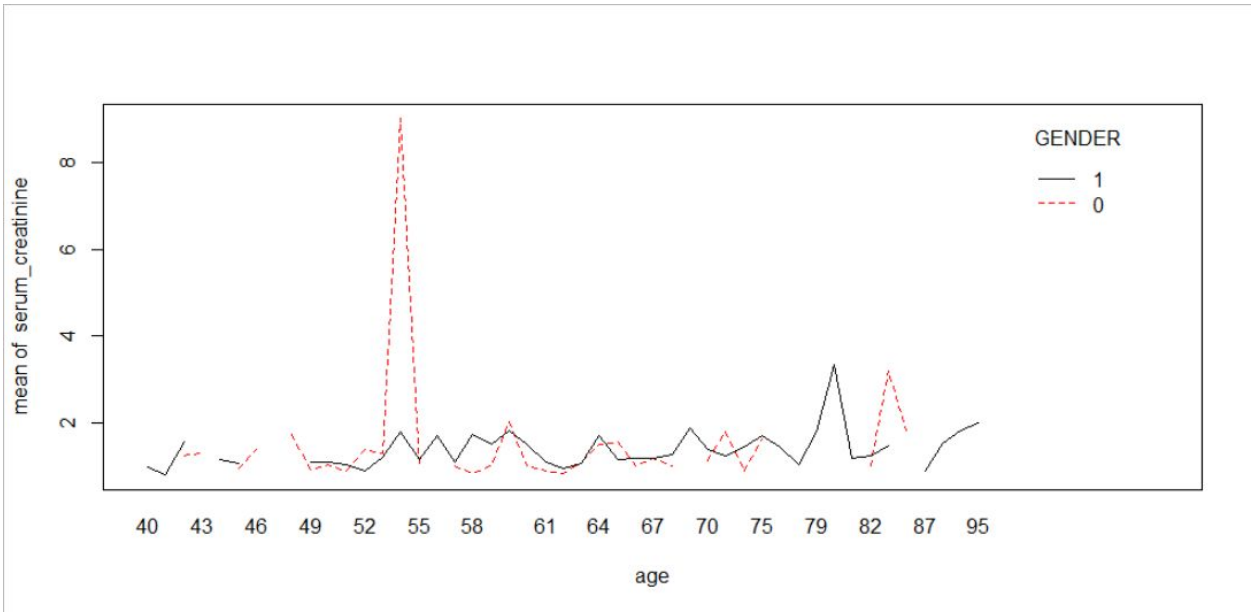
Age VS Ejection Fraction

In this interaction plot, the lines are not parallel. The interaction plot suggests there is an interaction between age and ejection fraction.



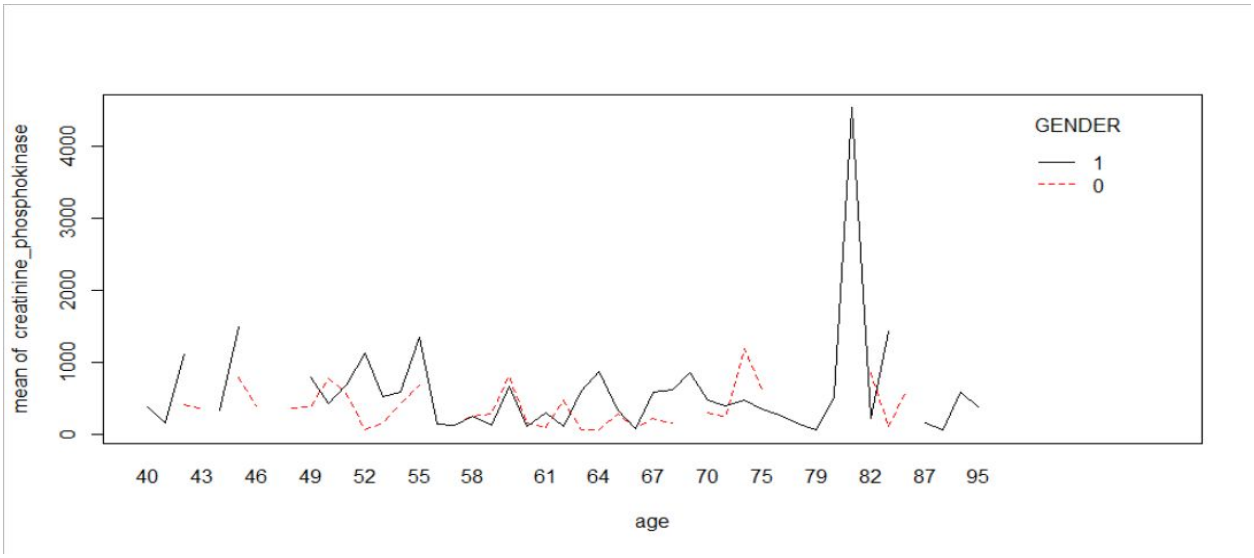
Age VS Serum Creatinine

In this interaction plot, the lines are not parallel. The interaction plot suggests there is an interaction between age and serum creatinine.



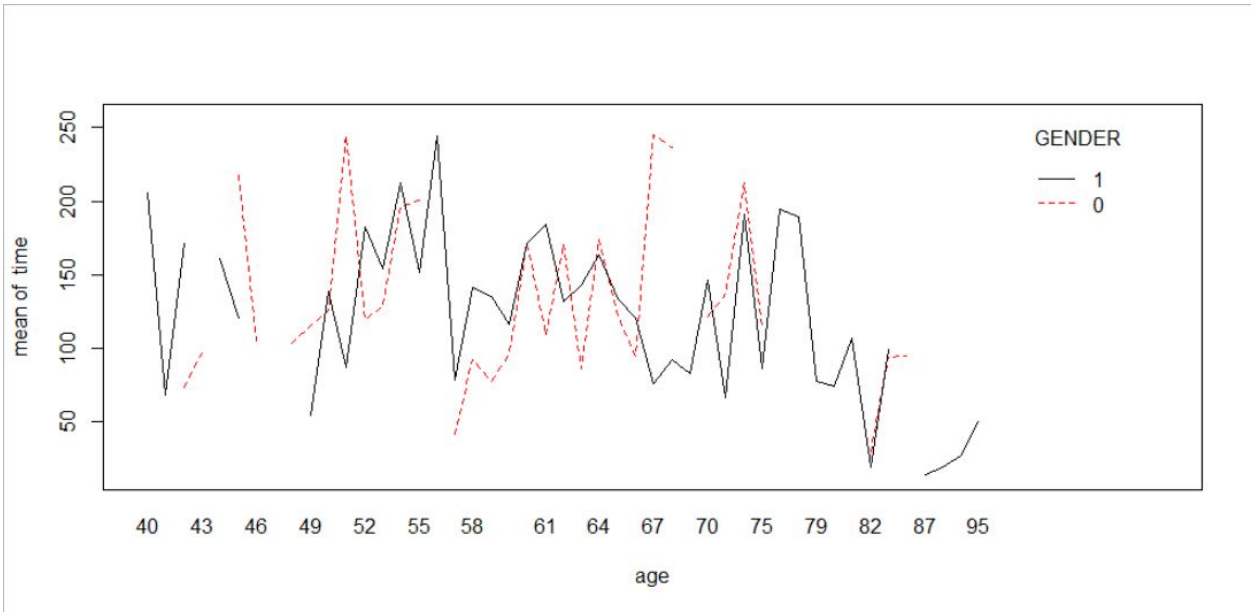
Age VS Creatinine Phosphokinase

In this interaction plot, the lines are not parallel. The interaction plot suggests there is an interaction between age and creatinine phosphokinase.



Age VS Time

In this interaction plot, the lines are not parallel. The interaction plot suggests there is an interaction between age and time.



7. Dividing the Dataset

Partitioning data into training and testing sets helps you to develop highly accurate models that are relevant to data that you collect in the future, not just the data the model was trained on. By training your data and testing it on the holdout set, you get a real sense of how accurate the model's outcomes will be, leading to better decisions and greater confidence in your model's accuracy.

Here we partition our dataset as 70% of values are included in the training set and remaining 30% into the testing set. Training dataset is used to predict the relationship between predictor variables and predicted variables. The training dataset is used to build the model and testing is used to validate the model.

```
set.seed(100)
sample_df<-sample.split(df,SplitRatio = 0.70)
train<-subset(df,sample_df==T)
test<-subset(df,sample_df==F)
```

So, after splitting our data, below image displays number of observations in training dataset and testing dataset

```
> nrow(train)
[1] 207
> nrow(test)
[1] 92
```

8. Stepwise variable selection technique

In this step, we will use the stepwise regression technique to select explanatory variables which are significant.

We begin by fitting the full model (all predictors used) and then we remove one predictor at a time based on the p-values. We repeat the process until we obtain a model with lowest AIC value and only significant predictors are used to fit the model. Detailed process is explained below.

First, we begin by fighting Full model

```
fit<-glm(DEATH_EVENT~.,family = binomial,data=train)
summary(fit)
```

Output:

```
> summary(fit)

Call:
glm(formula = factor(DEATH_EVENT) ~ ., family = binomial, data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0884  -0.6149  -0.2655   0.5164   2.5094

Coefficients:
                           Estimate Std. Error z value Pr(>|z|)
(Intercept)               5.254e+00  6.811e+00   0.771 0.440486
age                       4.498e-02  1.843e-02   2.440 0.014683 *
anaemia1                 -2.836e-01  4.214e-01  -0.673 0.501007
creatinine_phosphokinase  2.739e-04  1.951e-04   1.404 0.160318
diabetes1                 1.042e-01  4.060e-01   0.257 0.797497
ejection_fraction        -6.337e-02  1.834e-02  -3.455 0.000551 ***
high_blood_pressure1      2.600e-01  4.152e-01   0.626 0.531274
platelets                -1.529e-06  2.199e-06  -0.695 0.487025
serum_creatinine          5.736e-01  1.842e-01   3.114 0.001845 **
serum_sodium             -3.316e-02  4.900e-02  -0.677 0.498608
sex1                     -6.087e-01  4.803e-01  -1.267 0.205101
smoking1                 -2.069e-01  5.011e-01  -0.413 0.679721
time                     -1.898e-02  3.412e-03  -5.563 2.65e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 260.66  on 206  degrees of freedom
Residual deviance: 161.70  on 194  degrees of freedom
AIC: 187.7
```

As we can see, there are 8 predictors which are not significant in this model. Moreover, AIC for this model is 187.7. Model with the lowest AIC is considered the best. Observing the p-values, we conclude that the predictor diabetes is the least significant of all other predictors. Hence, we will remove diabetes and check the model again. We remove predictor one by one until the best model is reached.

Below is the code run to achieve the final model.

```
drop1(fit,test = "Chisq")
#removing diabetes as it has largest p-value=0.79
fit.one <- update(fit,. ~ . - diabetes)
summary(fit.one)
#smoking has largest p value
drop1(fit.one,test = "Chisq")
#again smoking has largest p value
```

```
fit.two <- update(fit.one,. ~ . - smoking)
summary(fit.two)
# high blood presuure has largest p-value
drop1(fit.two,test = "Chisq")
#again high blood presuure has largest p-value
fit.three <- update(fit.two,. ~ . - high_blood_pressure)
summary(fit.three)
#removing anaemia
fit.four <- update(fit.three,. ~ . - anaemia)
summary(fit.four)
#removing platelets
fit.five <- update(fit.four,. ~ . - platelets)
summary(fit.five)
#removing serium_sodium
fit.six <- update(fit.five,. ~ . - serum_sodium)
summary(fit.six) #178.01
#removing creatine_phosphokinase
fit.seven <- update(fit.six,. ~ . - creatinine_phosphokinase)
summary(fit.seven)#AIC 178.55
# removing sex
fit.eight <- update(fit.seven,. ~ . - sex)
summary(fit.eight)#AIC 178.27
```

Finally, we see model ‘fit.six’ has the lowest AIC = 178.01, also it has lower p-values of predictors compared to model ‘fit.seven’ and ‘fit.eight’.Below is the output of model ‘fit.six’.

```
> summary(fit.six) #178.01

Call:
glm(formula = factor(DEATH_EVENT) ~ age + creatinine_phosphokinase +
    ejection_fraction + serum_creatinine + sex + time, family = binomial,
    data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1067  -0.6108  -0.2769   0.5370   2.6108

Coefficients:
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)              0.3445774  1.2842949   0.268 0.788468
age                      0.0446994  0.0178508   2.504 0.012278 *
creatinine_phosphokinase 0.0002687  0.0001863   1.442 0.149197
ejection_fraction       -0.0652614  0.0176059  -3.707 0.000210 ***
serum_creatinine         0.5918989  0.1788088   3.310 0.000932 ***
sex1                    -0.6419663  0.4181639  -1.535 0.124734
time                    -0.0186189  0.0032384  -5.749 8.95e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 260.66  on 206  degrees of freedom
Residual deviance: 164.01  on 200  degrees of freedom
AIC: 178.01
```

Hence, in the end, we conclude that predictors age, creatinine_phosphokinase, ejection_fraction, serum_creatinine, sex and time are significant for our analysis.

9. Comparing model with no interaction vs with interaction term

In this part, we will compare our reduced model with different models having different interaction terms. Also, we will check whether our reduced or full model is better.

Initially, we will start with the variables which were claimed to be significant after running stepwise regression .i.e. age, creatinine_phosphokinase, ejection_fraction, serum_creatinine, sex and time.

```
> fit2<-glm(DEATH_EVENT~age+ejection_fraction+serum_creatinine+creatinine_phosphokinase+
+             time+sex,family = binomial,data=train)
> summary(fit2)

Call:
glm(formula = DEATH_EVENT ~ age + ejection_fraction + serum_creatinine +
    creatinine_phosphokinase + time + sex, family = binomial,
    data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1067  -0.6108  -0.2769   0.5370   2.6108

Coefficients:
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)              0.3445774  1.2842949   0.268 0.788468
age                      0.0446994  0.0178508   2.504 0.012278 *
ejection_fraction       -0.0652614  0.0176059  -3.707 0.000210 ***
serum_creatinine         0.5918989  0.1788088   3.310 0.000932 ***
creatinine_phosphokinase 0.0002687  0.0001863   1.442 0.149197
time                    -0.0186189  0.0032384  -5.749 8.95e-09 ***
sex1                    -0.6419663  0.4181639  -1.535 0.124734
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 260.66  on 206  degrees of freedom
Residual deviance: 164.01  on 200  degrees of freedom
AIC: 178.01

Number of Fisher Scoring iterations: 5
```

Fit 2 is our reduced model.

Now, we will create a full model with the interaction terms. The interaction terms are as follow:

1. age*ejection_fraction
2. age*serum_creatinine
3. age*creatinine_phosphokinase
4. age*time

As shown in the screenshot below, the fit3 is our full model which includes all the interaction terms.

```
> fit3<-glm(DEATH_EVENT~age+ejection_fraction+serum_creatinine+creatinine_phosphokinase+time
+           +sex+age*ejection_fraction+ age*serum_creatinine+age*creatinine_phosphokinase+
+             age*time,family = binomial,data=train)
> summary(fit3)

Call:
glm(formula = DEATH_EVENT ~ age + ejection_fraction + serum_creatinine +
    creatinine_phosphokinase + time + sex + age * ejection_fraction +
    age * serum_creatinine + age * creatinine_phosphokinase +
    age * time, family = binomial, data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2044  -0.5917  -0.2782   0.4431   2.5982

Coefficients:
                                Estimate Std. Error z value Pr(>|z|)
(Intercept)                   -3.059e+00  4.466e+00  -0.685    0.493
age                            1.044e-01  7.643e-02   1.366    0.172
ejection_fraction             -1.010e-01  1.099e-01  -0.918    0.358
serum_creatinine               1.550e+00  1.273e+00   1.218    0.223
creatinine_phosphokinase       9.802e-04  7.545e-04   1.299    0.194
time                           7.010e-03  1.747e-02   0.401    0.688
sex1                          -6.876e-01  4.235e-01  -1.624    0.104
age:ejection_fraction          5.082e-04  1.771e-03   0.287    0.774
age:serum_creatinine          -1.650e-02  2.118e-02  -0.779    0.436
age:creatinine_phosphokinase -1.176e-05  1.190e-05  -0.988    0.323
age:time                      -4.177e-04  2.864e-04  -1.458    0.145

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 260.66  on 206  degrees of freedom
Residual deviance: 160.46  on 196  degrees of freedom
AIC: 182.46

Number of Fisher Scoring iterations: 5
```

Now, we'll be comparing the above two models (fit2 and fit3) using anova and likelihood ratio test.

```
> anova(fit2,fit3,test="LRT")
Analysis of Deviance Table

Model 1: DEATH_EVENT ~ age + ejection_fraction + serum_creatinine + creatinine_phosphokinase +
    time + sex
Model 2: DEATH_EVENT ~ age + ejection_fraction + serum_creatinine + creatinine_phosphokinase +
    time + sex + age * ejection_fraction + age * serum_creatinine +
    age * creatinine_phosphokinase + age * time
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1       200     164.01
2       196     160.46  4   3.5432   0.4713
```

H0: Reduced model suits best
Ha: Full model suits better

P-value = 0.4713
=> p < 0.05 (Level of significance)
=> We accept the null hypothesis.
Hence, we conclude that the Reduced model suits better than the full model.

Hence, now that we know that the full model with interaction terms has been rejected. So we'll be comparing reduced model by adding interaction terms to it.

Here, we are taking interaction term age*ejection_fraction along with the reduced model. We'll be calling this model as fit4.

```
> fit4<-glm(DEATH_EVENT~age+ejection_fraction+serum_creatinine+creatinine_phosphokinase+
+             time+sex+age*ejection_fraction,family = binomial,data=train)
> summary(fit4)

Call:
glm(formula = DEATH_EVENT ~ age + ejection_fraction + serum_creatinine +
    creatinine_phosphokinase + time + sex + age * ejection_fraction,
    family = binomial, data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0920  -0.6148  -0.2761   0.5316   2.6349

Coefficients:
                           Estimate Std. Error z value Pr(>|z|)
(Intercept)               1.0988732  3.8117162   0.288 0.773126
age                       0.0319151  0.0632365   0.505 0.613773
ejection_fraction        -0.0868175  0.1042571  -0.833 0.405000
serum_creatinine          0.5960733  0.1800935   3.310 0.000934 ***
creatinine_phosphokinase  0.0002680  0.0001864   1.438 0.150466
time                     -0.0185527  0.0032467  -5.714  1.1e-08 ***
sex1                     -0.6369121  0.4190075  -1.520 0.128499
age:ejection_fraction     0.0003536  0.0016815   0.210 0.833467
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 260.66  on 206  degrees of freedom
Residual deviance: 163.96  on 199  degrees of freedom
AIC: 179.96

Number of Fisher Scoring iterations: 5
```

Now, we'll be comparing the above model with reduced model (fit2 and fit4) using anova and likelihood ratio test.

```
> anova(fit2,fit4,test='LRT')
Analysis of Deviance Table

Model 1: DEATH_EVENT ~ age + ejection_fraction + serum_creatinine + creatinine_phosphokinase +
    time + sex
Model 2: DEATH_EVENT ~ age + ejection_fraction + serum_creatinine + creatinine_phosphokinase +
    time + sex + age * ejection_fraction
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1       200     164.01
2       199     163.96  1 0.044705   0.8325
```

H0: Reduced model suits best
Ha: Reduced model with interaction term age*ejection_fraction suits better

P-value = 0.8325
=> p < 0.05 (Level of significance)
=> We accept the null hypothesis.

Hence, we conclude that the Reduced model suits better than the Reduced model with interaction term age*ejection_fraction.

Hence, now as we know that the reduced model with interaction term age*ejection_fraction has been rejected. Next we'll be considering Reduced model with interaction term age*serum_creatinine. We'll name this model as fit 5.

```
> fit5<-glm(DEATH_EVENT~age+ejection_fraction+serum_creatinine+creatinine_phosphokinase+
+             time+sex+ age*serum_creatinine,family = binomial,data=train)
> summary(fit5)

Call:
glm(formula = DEATH_EVENT ~ age + ejection_fraction + serum_creatinine +
    creatinine_phosphokinase + time + sex + age * serum_creatinine,
    family = binomial, data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1279  -0.5985  -0.2727   0.5326   2.6544

Coefficients:
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)             -0.5758366  2.1548112  -0.267 0.789290
age                      0.0613191  0.0360595   1.700 0.089037 .
ejection_fraction       -0.0663851  0.0178163  -3.726 0.000194 ***
serum_creatinine         1.2344768  1.2146712   1.016 0.309484
creatinine_phosphokinase 0.0002708  0.0001880   1.440 0.149888
time                    -0.0188238  0.0032650  -5.765 8.15e-09 ***
sex1                    -0.6390909  0.4190718  -1.525 0.127255
age:serum_creatinine    -0.0108135  0.0201113  -0.538 0.590796
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 260.66  on 206  degrees of freedom
Residual deviance: 163.73  on 199  degrees of freedom
AIC: 179.73

Number of Fisher Scoring iterations: 5
```

Now, we'll be comparing the above model with the reduced model (fit2 and fit5) using anova and likelihood ratio test.

```
> anova(fit2,fit5,test='LRT')
Analysis of Deviance Table

Model 1: DEATH_EVENT ~ age + ejection_fraction + serum_creatinine + creatinine_phosphokinase +
    time + sex
Model 2: DEATH_EVENT ~ age + ejection_fraction + serum_creatinine + creatinine_phosphokinase +
    time + sex + age * serum_creatinine
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1       200     164.01
2       199     163.73  1  0.27511   0.5999
```

H0: Reduced model suits best
Ha: Reduced model with interaction term age*serum_creatinine suits better

P-value = 0.5999
=> p < 0.05 (Level of significance)
=> We accept the null hypothesis.

Hence, we conclude that the Reduced model suits better than the Reduced model with interaction term age*serum_creatinine.

Hence, now as we know that the reduced model with interaction term age*serum_creatinine. has been rejected. Next we'll be considering a Reduced model with interaction term age*creatinine_phosphokinase. We'll name this model as fit 6.

```
> fit6<-glm(DEATH_EVENT~age+ejection_fraction+serum_creatinine+creatinine_phosphokinase+
+             time+sex+ age*creatinine_phosphokinase,family = binomial,data=train)
> summary(fit6)

Call:
glm(formula = DEATH_EVENT ~ age + ejection_fraction + serum_creatinine +
    creatinine_phosphokinase + time + sex + age * creatinine_phosphokinase,
    family = binomial, data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1187  -0.5940  -0.2771   0.5133   2.6255

Coefficients:
                              Estimate Std. Error z value Pr(>|z|)
(Intercept)                  -1.525e-01  1.414e+00  -0.108 0.914070
age                           5.302e-02  2.066e-02   2.566 0.010288 *
ejection_fraction            -6.581e-02  1.770e-02  -3.719 0.000200 ***
serum_creatinine              5.915e-01  1.790e-01   3.305 0.000949 ***
creatinine_phosphokinase      8.346e-04  7.198e-04   1.160 0.246248
time                         -1.853e-02  3.246e-03  -5.709 1.14e-08 ***
sex1                         -6.512e-01  4.194e-01  -1.553 0.120461
age:creatinine_phosphokinase -9.475e-06  1.115e-05  -0.850 0.395253
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 260.66  on 206  degrees of freedom
Residual deviance: 163.30  on 199  degrees of freedom
AIC: 179.3

Number of Fisher Scoring iterations: 5
```

Now, we'll be comparing the above model with the reduced model (fit2 and fit6) using anova and likelihood ratio test.

```
> anova(fit2,fit6,test='LRT')
Analysis of Deviance Table

Model 1: DEATH_EVENT ~ age + ejection_fraction + serum_creatinine + creatinine_phosphokinase +
    time + sex
Model 2: DEATH_EVENT ~ age + ejection_fraction + serum_creatinine + creatinine_phosphokinase +
    time + sex + age * creatinine_phosphokinase
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1       200     164.01
2       199     163.30  1  0.70581   0.4008
```

H0: Reduced model suits best
Ha: Reduced model with interaction term age*creatinine_phosphokinase suits better

P-value = 0.4008
=> p < 0.05 (Level of significance)
=> We accept the null hypothesis.

Hence, we conclude that the Reduced model suits better than the Reduced model with interaction term age*creatinine_phosphokinase.

Hence, now as we know that the reduced model with interaction term age*creatinine_phosphokinase has been rejected. Next we'll be considering a Reduced model with interaction term age*time. We'll name this model as fit 7.

```
> fit7<-glm(DEATH_EVENT~age+ejection_fraction+serum_creatinine+creatinine_phosphokinase+
+             time+sex+ age*time,family = binomial,data=train)
> summary(fit7)

Call:
glm(formula = DEATH_EVENT ~ age + ejection_fraction + serum_creatinine +
    creatinine_phosphokinase + time + sex + age * time, family = binomial,
    data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1735  -0.6056  -0.2848   0.4369   2.4815

Coefficients:
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)             -2.1035982  2.1317829  -0.987 0.323751
age                      0.0863878  0.0351310   2.459 0.013932 *
ejection_fraction       -0.0674875  0.0179651  -3.757 0.000172 ***
serum_creatinine         0.5743814  0.1779414   3.228 0.001247 **
creatinine_phosphokinase 0.0002874  0.0001848   1.555 0.119940
time                     0.0072154  0.0174297   0.414 0.678898
sex1                    -0.6727552  0.4194713  -1.604 0.108754
age:time                -0.0004216  0.0002883  -1.462 0.143616
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 260.66  on 206  degrees of freedom
Residual deviance: 161.76  on 199  degrees of freedom
AIC: 177.76

Number of Fisher Scoring iterations: 5
```

Now, we'll be comparing the above model with the reduced model (fit2 and fit7) using anova and likelihood ratio test.

```
> anova(fit2,fit7,test='LRT')
Analysis of Deviance Table

Model 1: DEATH_EVENT ~ age + ejection_fraction + serum_creatinine + creatinine_phosphokinase +
    time + sex
Model 2: DEATH_EVENT ~ age + ejection_fraction + serum_creatinine + creatinine_phosphokinase +
    time + sex + age * time
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1       200     164.01
2       199     161.76  1   2.2433   0.1342
```

H0: Reduced model suits best
Ha: Reduced model with interaction term age*time suits better

P-value = 0.1342
=> p < 0.05 (Level of significance)
=> We accept the null hypothesis.

Hence, we conclude that the Reduced model suits better than the Reduced model with interaction term age*time.

Hence, after comparing the Reduce Model with the Model with all the interaction terms we finally conclude that our Reduce Model with variables age, creatinine_phosphokinase, ejection_fraction, serum_creatinine, sex and time is best than other ones.

```
> waldtest(fit2,test = "Chisq")
Wald test

Model 1: DEATH_EVENT ~ age + ejection_fraction + serum_creatinine + creatinine_phosphokinase +
    time + sex
Model 2: DEATH_EVENT ~ 1
  Res.Df Df  Chisq Pr(>Chisq)
1    200
2    206 -6 50.407  3.896e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
```

10. Sensitivity ,Specificity And ROC: -

Sensitivity: - Sensitivity is defined as the ability of modal to identify correctly the patients with disease.
Mathematically it is given as: -

$$P(T^+|D^+) = TP / (TP+FN).$$

Where, **TP**- true positives
FN-false negatives

True Positives is defined as the patients who were tested with a disease and the modal predicted them having the disease.
Similarly, False negatives is defined as patients who do not have the disease, and the modal classified it not having the disease.

Specificity: - The ability of a test to correctly identify people without disease.

Mathematically, it is given as : -

$$\mathbf{P(T^{-}|D^{-}) = TN / (TN + FP).}$$

where **TN-** True Negative

FP- false positives: - False positive is when the modal test positive for the disease, and the disease is actually not present.

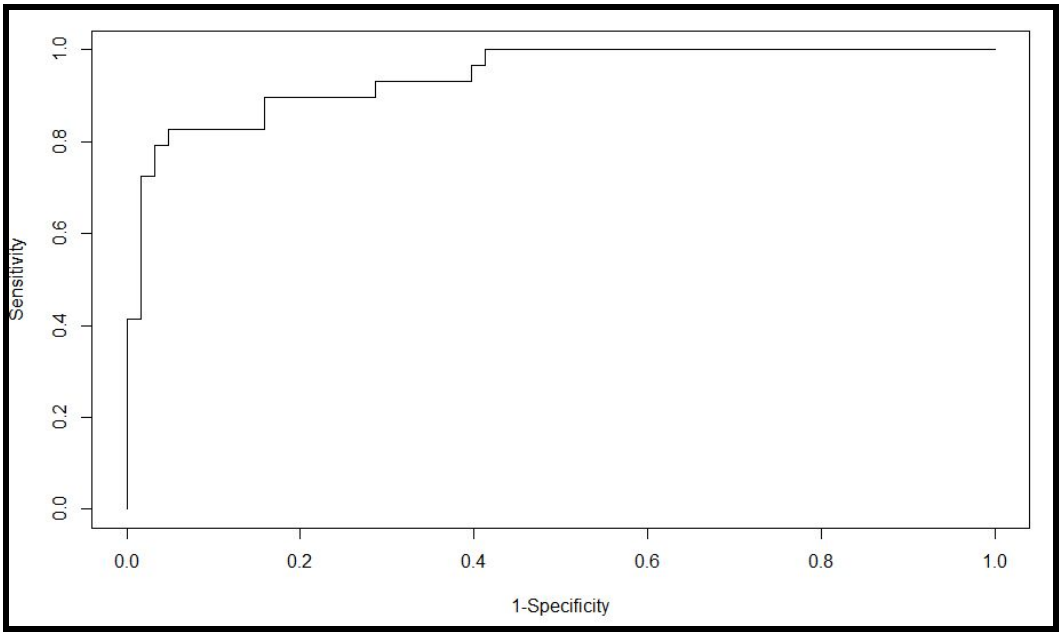
```
> sum(diag(table_mat))/sum(table_mat)
[1] 0.9130435
> 1-sum(diag(table_mat))/sum(table_mat)
[1] 0.08695652
```

Accuracy:-

Accuracy is defined as the ability of the modal to correctly identify the class of the response variable on the hidden data i.e the test data. The accuracy for the data in the dataset was found out to be 91% which is extremely good.

ROC curve :-

ROC curve describes the trade off between the sensitivity and the specificity of the modal. Classifier with top left corner describes a better performance as the one close to the diagonal which is at 45 degree angle.
As seen from the figure above, the ROC curve is fairly good.



11. Hosmer and Lemeshow Test

```
> h1

        Hosmer and Lemeshow goodness of fit (GOF) test

data:  train$DEATH_EVENT, fitted(fit2)
X-squared = 207, df = 8, p-value < 2.2e-16
```