**Predicting Heart Failure Survivability Using Logistic Regression**

**Fred Nagle**

*Montclair State University, Montclair, New Jersey 07043*

Cardiovascular disease kills 659,000 people in the United States annually and 17 million people globally. The main objective of this study was to identify the statistically significant risk factors in the survivability of heart failure. This study has a dataset of 299 individuals who experienced heart failure. We utilized several forms of logistical regression to identify the significant predictors for the survivability of heart failure. Then we created a predictive model to identify the survivability of heart failure based on risk factors found during logistic regression. Based on the analysis, we identified the following factors as significant predictors in the survivability of heart failure: 1) Age of patient, 2) Follow up period, 3) Level of serum creatinine in the blood, and 4) Percentage of blood leaving the heart at each contraction. The predictive model allows us to accurately identify heart failure survivability with 74% accuracy.

**Introduction**

Cardio Vascular Disease (CVD) is a group of disorders of the heart and blood vessels. Some of these disorders include: coronary heart disease (heart attacks), cerebrovascular diseases (strokes), heart failure (HF), and other types of pathology [1]. Each year CVD kills 17 million people globally [2]. Heart failure a symptom of CVD and according to the CDC, 6 million adults are afflicted with heart failure each year in the U.S. [3].

Heart failure is defined by the ejection fraction which is the percentage of blood leaving the heart after each contraction. A normal heart has an ejection fraction between 50% - 70%. An individual is experiencing heart failure if the ejection fraction is less than 40% [4].

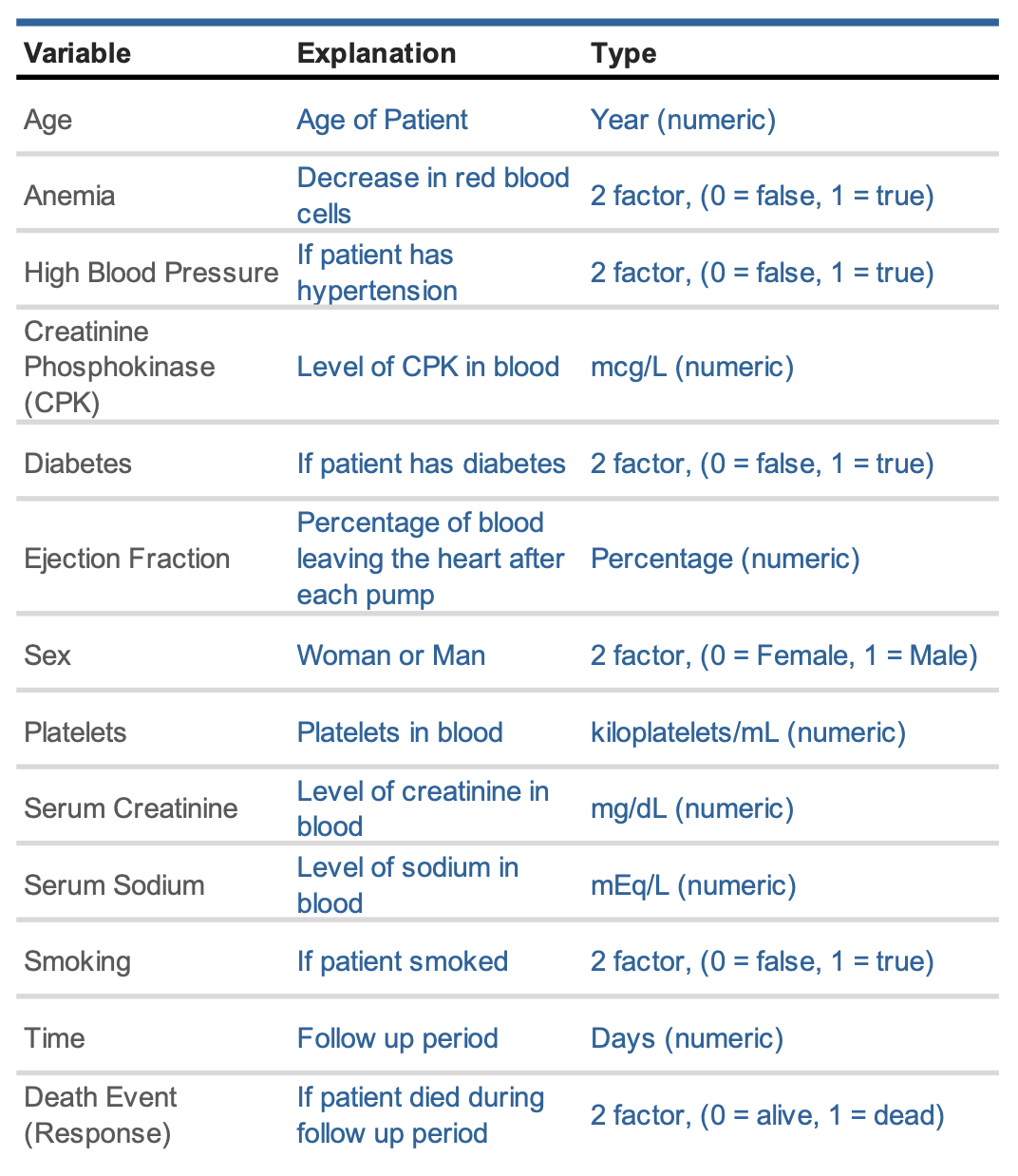
Predicting heart failure can help thwart death from CVD. To date, forecasting heart failure-related events in clinical practice has not been accurate enough to timely apply interventions to reduce death from CVD. Scientists have identified many physiological variables that are associated with heart failure, however, there is no consensus on how impactful they are to survivability [6].

In this paper we analyzed the medical records of 299 individuals who experienced heart failure in Punjab Pakistan. We will be using several logistic regression models to compare and identify the significant predictors in the survivability of heart failure.

**Materials and Methods**

***Data***

We were given a data set of 299 patients that experienced heart failure. The data was collected by the Faisalabad Institute of Cardiology and at the Allied Hospital in Faisalabad (Punjab, Pakistan), during April–December 2015 [7]. The data set contained 13 variables including: clinical, body, and lifestyle descriptives. Some variables were two-factor categorical: anemia, diabetes, high blood pressure, sex, smoking, death event. Other variables were continuous numerical: age, creatinine phosphokinase (CPK), ejection fraction, platelets, serum creatinine, serum sodium, time.



**Table 1:**  Table naming all the variables from the dataset, giving a brief description of each and identifying the variable class.

***Assumptions and Diagnostics***

Since our goal is to predict survivability of heart failure we will use logistic regression because survivability is a two-factor variable (alive = 0, dead = 1). We will be using R to find the logistic regression equation and diagnostics.

We first determine if the numerical variables are correlated to each other.

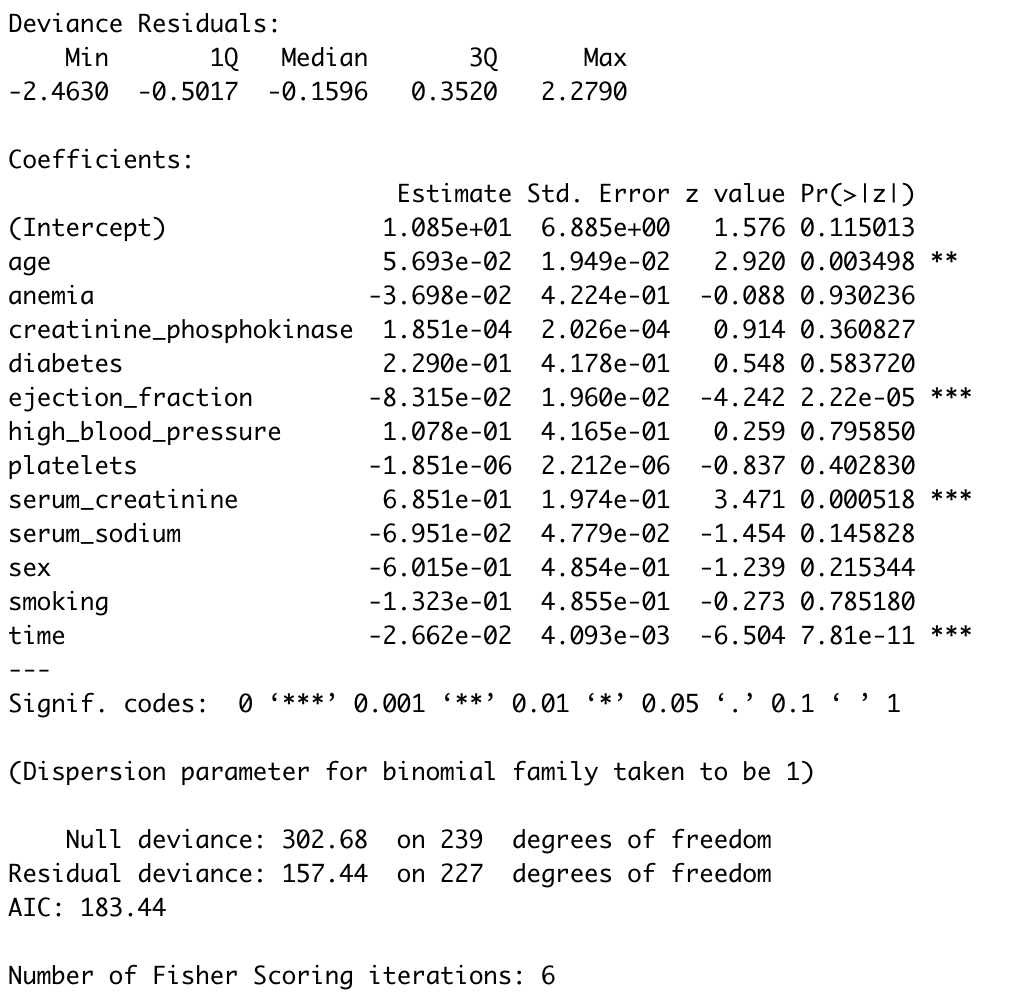


**Figure 1:**  Correlation Plot showing if numerical variables are correlated to themselves and/or other variables.

Here we see that the matrix is symmetrical and that the diagonal is perfectly positively correlated because it shows each variable is correlated with itself. We can now safely assume there is no multicollinearity within the dataset.

***Multiple Logistic Regression***

Next, we performed multiple logistic regression analysis on the full model to find significant predictors in heart failure survivability and compute how accurate the model can predict heart failure survivability.



**Figure 2:**  Full model logistic regression of heart failure dataset. Starred variables are variables of interest.

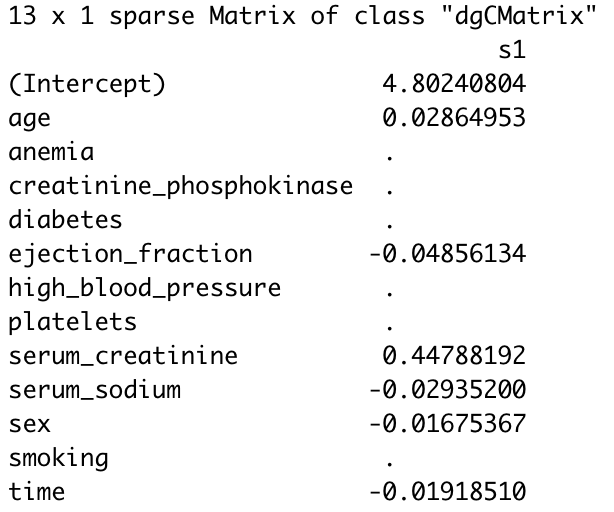
We see there are several significant predictors in predicting heart failure survivability. From the twelve predictors in the above dataset, there are four significant predictors: 1) age, 20 ejection fraction, 3) serum creatinine, and 4) time. These predictors affect the outcome of heart failure survivability.

We now look at how accurate the model can predict heart failure survivability. From our analysis we found a prediction accuracy rating of 0.8135593, or the classification prediction accuracy is about 81% in predicting heart failure survivability.

***Penalized Logistic Regression (Lasso Regression)***

Penalized logistic regression imposes a penalty on logistic regression models with too many variables. We will be specifically using lasso regression which will force less contributive variables coefficients in a model to be exactly zero. Leaving only significant predictor variables in the model. There will be two kinds of lasso regression we will be analyzing, lambda.min and lambda.1se.

***Lambda.min Model***



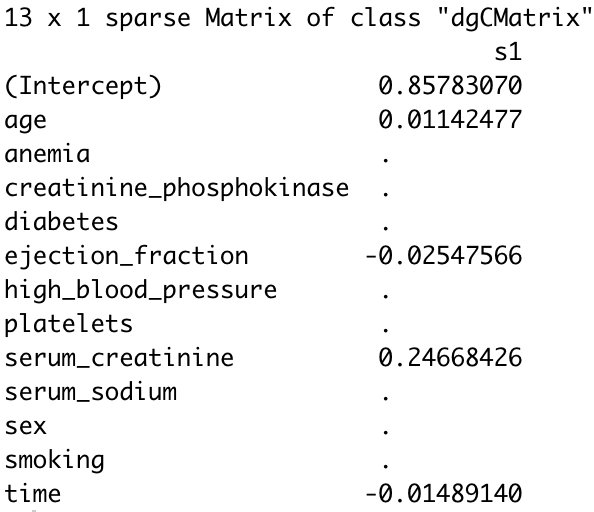
**Figure 3:**  lambda.min model of logistic regression of heart failure dataset.

Here we use lambda.min function in our lasso regression. The lambda.min function uses the value of λ that gives the minimum mean cross-validated error.

We see in the lambda.min lasso regression, the model discarded five of the twelve predictors from the dataset. The model found that there are seven significant predictors that affect the outcome of heart failure: survivability: 1) age, 2) anemia, 3) creatinine phosphokinase, 4) ejection fraction, 5) serum creatinine, 6) serum sodium, and 7) time.

We now look at how accurate the model can predict heart failure survivability. From our analysis we found our model had a prediction accuracy rating of 0.8135593, or the classification prediction accuracy is about 81% in predicting heart failure survivability.

***Lambda.1se Model***



**Figure 4:**  lambda.1se model of logistic regression of heart failure dataset.

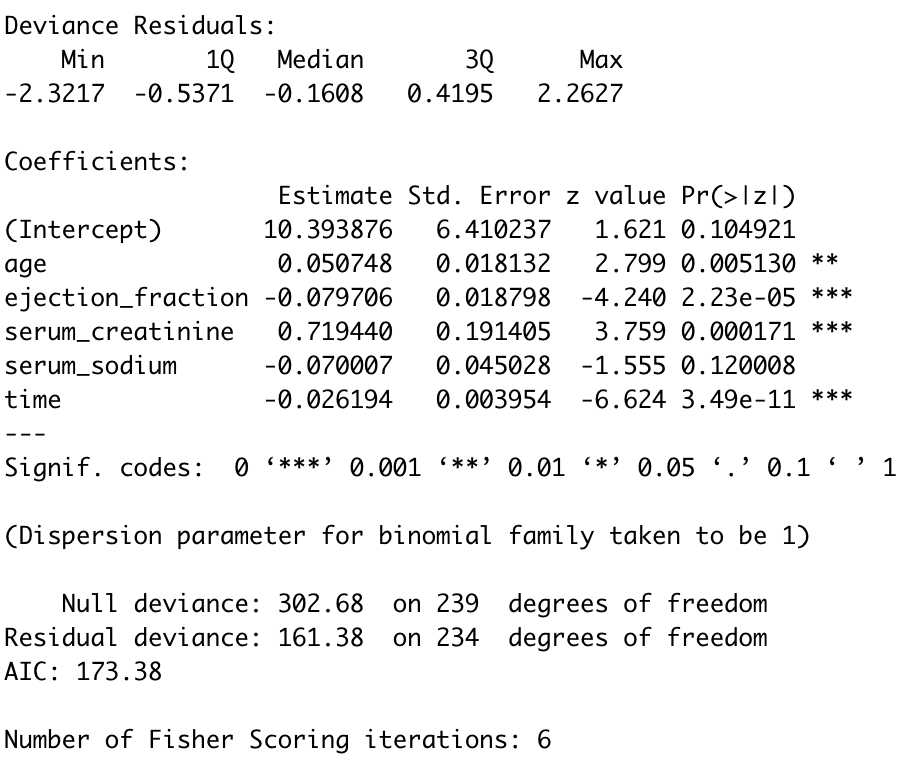
Here we use lambda.1se function in our lasso regression. The lambda.1se function produces the most normalized model such that the error is within one standard deviation of the minimum.

We see in the lambda.1se lasso regression, the model discarded seven of the twelve predictors from the dataset. The model found that there are five significant predictors that affect the outcome of heart failure survivability: 1) age, 2) ejection fraction, 3) serum creatinine, 4) serum sodium, and 5)time.

We now look at how accurate the model can predict heart failure survivability. From our analysis we found our model had a prediction accuracy rating of 0.7627119, or the classification prediction accuracy is about 76% in predicting heart failure survivability.

***Stepwise Logistic Regression***

Here we will perform a different kind of logistic regression known as stepwise logistic regression. Stepwise logistic regression is a model that automatically selects a smaller number of variables to build one of the best performing logistic regression models.



**Figure 5:**  Stepwise model of logistic regression of heart failure dataset.

We see in the stepwise logistic regression the model automatically discarded eight of the twelve predictors from the dataset. The model found that there are four significant predictors that affect the outcome of heart failure survivability: 1) age, 2) ejection fraction, 3) serum creatinine, and 4) time.

We now look at how accurate the model can predict heart failure survivability. From our analysis we found our model had a prediction accuracy rating of 0.8474576, or the classification prediction accuracy is about 85% in predicting heart failure survivability.

We can create a model of the logistic regression:

**Equation 1:**  Equation of stepwise model using significant predictors.

**Results**

Using the step-wise logistic regression model as our guide. We will now look at each statistically significant predictor the model selected: 1) age, 2) ejection fraction, 3) serum creatinine, and 4) time.

***Death Event as related to Age***

Here plotted below is a boxplot of Death event as related to age.



**Figure 6:**  Side by side boxplot of Death Event v. Age of patients from the heart failure Dataset.

Observing this box plot it is obvious from the data that older individuals with heart failure are more likely to die of heart failure. Median age of patients who die of heart failure is 65 years old. The median age of patients who are alive with heart failure is 60 years old.

***Death Event as related to Ejection Fraction***

Here plotted below is a boxplot of Death event as related to Ejection Fraction.



**Figure 7:**  Side by side boxplot of Death Event v. Ejection Fraction of patients from the heart failure Dataset.

Observing this box plot it is obvious from the data that an individual with heart failure is more likely to be in the dead patients’ category if the individual has a lower ejection fraction than survived patients’ category. The median ejection fraction of patients in the alive category is 38% and 30% in the dead category.

***Death Event as related to Serum Creatinine***

Here plotted below is a boxplot of Death event as related to Serum Creatinine.



**Figure 8:**  Side by side boxplot of Death Event v. Serum Creatinine of patients from the heart failure Dataset.

Observing this box plot it is obvious from the data that the more serum creatinine in individuals’ blood with heart failure are more likely to be in the dead patients’ category than survived patients’ category. The median levels of creatinine in the blood of the dead patients is 1.3 mg/dL and is 1.0 mg/dL in the alive patients’ category.

***Death Event as related to Time***

Here plotted below is a boxplot of Death event as related to follow up period.



**Figure 9:**  Side by side boxplot of Death Event v. Time since last follow up of patients from the heart failure Dataset.

Observing this box plot it is obvious from the data that individuals with shorter follow up periods with heart failure are more likely to be in the dead patients’ category than the survived patients’ category. The median days since last follow up of patients in the alive category is 172 days and 44.5 days in the dead patients’ category.

Next, we will perform calculations to accurately identify the effectiveness of our new model. We can use a confusion matrix to show the predicted values model v. the actual values. After some calculation we found a true positive rate (TP) of .7368. Thus, our model with 74% accuracy correctly predicted if an individual can survive heart failure. Our true negative rate (TN) is .9. Thus, our model can predict with 90% accuracy if an individual will die of heart failure. We can also confirm by calculating the area under the curve (AUC) of our model is .8618. Since our AUC of the model is relatively close to 1, this indicates our model does a relatively good job of predicting survivability of heart failure of an individual.

**Discussion/Conclusion**

From the logistic regression analysis, we found the most effective regression analysis was the step-wise multiple logistic regression model. Step-wise logistic regression was able to identify the most statistically significant predictors that can be seen earlier in the full multiple logistic regression model, and create a model with a classification prediction accuracy of about 85%. This was our highest classification prediction accuracy rating for identifying heart failure. The step-wise model was able to create this accuracy with only four of the twelve predictor variables given in the dataset: 1) age of patient, 2) follow up period, 3) level of serum creatinine in the blood, and 4) percentage of blood leaving the heart at each contraction. Our calculated TP rate is 74%. This means our model correctly predicted with 74% accuracy whether an individual can survive heart failure. Our calculated TN Rate is 90%. This means our model correctly predicted with 90% accuracy if an individual will not die of heart failure.

We can confirm how statistically significant the predictors are by observing the box plots and performing a five-number summary for each significant predictor. Specifically, the median values were drastically different between the alive patients and the dead patients for each of the predictors.

This is a relatively small dataset. Therefore, to increase the reliability of the modeling results, I recommend including more patients in the dataset in future analysis.

Using time since last follow up as a significant predictor variable is a probable confounding variable since, most likely, people who go to the doctor more often for follow ups are likely near the end of their life.

**References**

1. World Health Organization, World Heart Day.https://www.who.int/cardiovascular\_diseases/world-heart-day/en/. Accessed 1 April 2022.
2. World Health Organization, Cardiovascular diseases (CVDs). <https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)>. Accessed 5 April 2022.
3. National Heart, Lung, and Blood Institute, What Is Heart Failure?

<https://www.nhlbi.nih.gov/health/heart-failure>. Accessed 5 April 2022.

1. American Heart Association, Ejection Fraction Heart Failure Measurement. <https://www.heart.org/en/health-topics/heart-failure/diagnosing-heart-failure/ejection-fraction-heart-failure-measurement>. Accessed 10 April 2022.
2. Davide Chicco, Giuseppe Jurman. Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone. (BMC Medical Informatics and Decsion Making), 2020 doi:s12911-020-1023-5. Accessed 14 February 2022.
3. Buchan TA, Ross HJ, McDonald M, Billia F, Delgado D, Duero Posada JG, Luk A, Guyatt GH, Alba AC. Physician prediction versus model predicted prognosis in ambulatory patients with heart failure. J Heart Lung Transplant. 2019;38(4):381.
4. Ahmad T, Munir A, Bhatti SH, Aftab M, Raza MA. Survival analysis of heart failure patients: a case study. PLoS ONE. 2017;12(7):0181001.

**Code**

---

title: "Data First Look"

author: "Fred Nagle"

date: "1/29/2022"

output: pdf\_document

editor\_options:

chunk\_output\_type: console

---

```{r setup, include=FALSE}

library(readr)

library(tidyverse)

library(dplyr)

library(caTools)

library(ggplot2)

library(modelr)

library(caret)

library(corrplot)

library(glmnet)

library(broom)

library(InformationValue)

library(ISLR)

library(MASS)

```

```{r data, message=FALSE}

heart\_failure <- read\_csv("~/Desktop/Project/heart\_failure.csv")

heart\_failure <- rename(heart\_failure, anemia = anaemia)

# Convert columns from numeric to factor

heart\_failure\_factor <- heart\_failure

Factors<-c("anemia", "diabetes", "high\_blood\_pressure", "sex", "smoking", "DEATH\_EVENT")

heart\_failure\_factor[Factors]<-lapply(heart\_failure\_factor[Factors],factor)

head(heart\_failure\_factor)

```

# Descriptive Data Analysis

## Full Sample

```{r}

# Summary

summary(heart\_failure\_factor)

# Plots (Death Event v Ejection Fraction)

ggplot(heart\_failure\_factor, aes(DEATH\_EVENT, ejection\_fraction)) +

geom\_boxplot(aes(fill = DEATH\_EVENT)) +

scale\_fill\_manual(name="Death Event",labels=c("Survived Patients", "Dead Patients"), values=c("dodgerblue4", "firebrick4")) +

labs(title = "Death Event v. Ejection Fraction", x = "Death Event", y = "Ejection Fraction")

# Plots (Death Event v age)

ggplot(heart\_failure\_factor, aes(DEATH\_EVENT, age)) +

geom\_boxplot(aes(fill = DEATH\_EVENT)) +

scale\_fill\_manual(name="Death Event",labels=c("Survived Patients", "Dead Patients"), values=c("dodgerblue4", "firebrick4")) +

labs(title = "Death Event v. Age", x = "Death Event", y = "Age")

# Plots (Death Event v Serum Creatinine)

ggplot(heart\_failure\_factor, aes(DEATH\_EVENT, serum\_creatinine)) +

geom\_boxplot(aes(fill = DEATH\_EVENT)) +

scale\_fill\_manual(name="Death Event",labels=c("Survived Patients", "Dead Patients"), values=c("dodgerblue4", "firebrick4")) +

labs(title = "Death Event v. Serum Creatinine", x = "Death Event", y = "Serum Creatinine (mg/gL)")

# Plots (Death Event v Time)

ggplot(heart\_failure\_factor, aes(DEATH\_EVENT, time)) +

geom\_boxplot(aes(fill = DEATH\_EVENT)) +

scale\_fill\_manual(name="Death Event",labels=c("Survived Patients", "Dead Patients"), values=c("dodgerblue4", "firebrick4")) +

labs(title = "Death Event v. Time", x = "Death Event", y = "Time (Days)")

```

## Death Group

```{r}

# Filter dataset only deaths

heart\_failure\_death <- filter(heart\_failure\_factor, DEATH\_EVENT == 1)

# Summary

summary(heart\_failure\_death)

```

## Survival Group

```{r}

# Filter dataset only survival

heart\_failure\_survival <- filter(heart\_failure\_factor, DEATH\_EVENT == 0)

# Summary

summary(heart\_failure\_survival)

```

## Correlation plot

```{r}

correlations <- cor(heart\_failure\_factor[c(1,3,5,7,8,9,12)])

corrplot(correlations, method="circle")

```

# Logistic Regression Analysis

## Create Training Set & Test Set

```{r}

set.seed(123)

training.samples <- heart\_failure\_factor$DEATH\_EVENT %>%

createDataPartition(p = 0.8, list = FALSE)

train.data <- heart\_failure\_factor[training.samples, ]

test.data <- heart\_failure\_factor[-training.samples, ]

```

## Full Logistic Regression Analysis

```{r}

# Fit the model

full.model <- glm(DEATH\_EVENT ~., data = train.data, family = binomial)

# Summarize the model

summary(full.model)

coef(full.model)

car::vif(full.model)

```

#### Prediction accuracy of the full logistic regression model:

```{r}

# Make predictions

probabilities <- full.model %>% predict(test.data, type = "response")

predicted.classes <- ifelse(probabilities > 0.5, 1, 0)

# Prediction accuracy

observed.classes <- test.data$DEATH\_EVENT

mean(predicted.classes == observed.classes)

```

## Step Wise Logistic Regression

```{r}

step.model <- full.model %>% stepAIC(trace = FALSE)

summary(step.model)

coef(step.model)

```

### Compare the stepwise models

#### Prediction accuracy of the stepwise logistic regression model:

```{r}

# Make predictions

probabilities <- predict(step.model, test.data, type = "response")

predicted.classes <- ifelse(probabilities > 0.5, 1, 0)

# Prediction accuracy

observed.classes <- test.data$DEATH\_EVENT

mean(predicted.classes == observed.classes)

```

## Lasso Regression

### Create matrix of predictors

```{r}

# Dummy code categorical predictor variables

x <- model.matrix(DEATH\_EVENT~. , train.data)[,-1]

# Convert the outcome (class) to a numerical variable

y <- train.data$DEATH\_EVENT

```

#### Lasso Model

```{r}

glmnet(x, y, family = "binomial", alpha = 1, lambda = NULL)

```

#### Compute Lasso Regression

```{r}

set.seed(123)

cv.lasso <- cv.glmnet(x, y, alpha = 1, family = "binomial")

plot(cv.lasso)

```

```{r}

coef(cv.lasso, cv.lasso$lambda.min)

coef(cv.lasso, cv.lasso$lambda.1se)

```

#### Prediction accuracy of Lasso Model lambda min

```{r}

# Final model with lambda.min

lasso.model <- glmnet(x, y, alpha = 1, family = "binomial",

lambda = cv.lasso$lambda.min)

# Make prediction on test data

x.test <- model.matrix(DEATH\_EVENT ~., test.data)[,-1]

probabilities <- lasso.model %>% predict(newx = x.test)

predicted.classes <- ifelse(probabilities > 0.5, 1, 0)

# Model accuracy

observed.classes <- test.data$DEATH\_EVENT

mean(predicted.classes == observed.classes)

```

#### Prediction accuracy of Lasso Model lambda.1se

```{r}

# Final model with lambda.1se

lasso.model <- glmnet(x, y, alpha = 1, family = "binomial",

lambda = cv.lasso$lambda.1se)

# Make prediction on test data

x.test <- model.matrix(DEATH\_EVENT ~., test.data)[,-1]

probabilities <- lasso.model %>% predict(newx = x.test)

predicted.classes <- ifelse(probabilities > 0.5, 1, 0)

# Model accuracy rate

observed.classes <- test.data$DEATH\_EVENT

mean(predicted.classes == observed.classes)

```

## New Model

```{r}

#fit multiple regression

new\_multireg <- glm(DEATH\_EVENT~age+ejection\_fraction+serum\_creatinine+time, data = train.data, family = binomial)

#get regression output

summary(new\_multireg)

# Make predictions

probabilities <- new\_multireg %>% predict(test.data, type = "response")

predicted.classes <- ifelse(probabilities > 0.5, 1, 0)

# Prediction accuracy

observed.classes <- test.data$DEATH\_EVENT

mean(predicted.classes == observed.classes)

```

## Compute Accuracy

### Matrix

```{r}

#use model to predict probability of default

predicted <- predict(new\_multireg, test.data, type="response")

#find optimal cutoff probability to use to maximize accuracy

optimal <- optimalCutoff(test.data$DEATH\_EVENT, predicted)[1]

#create confusion matrix

confusionMatrix(test.data$DEATH\_EVENT, predicted)

```

### TP TN Rate

```{r}

#calculate sensitivity (TP Rate)

sensitivity(test.data$DEATH\_EVENT, predicted)

#calculate specificity (TN Rate)

specificity(test.data$DEATH\_EVENT, predicted)

#calculate total misclassification error rate

misClassError(test.data$DEATH\_EVENT, predicted, threshold=optimal)

```

### AUC

```{r}

#calculate probability of default for each individual in test dataset

predicted <- predict(new\_multireg, test.data, type="response")

#calculate AUC

library(pROC)

auc(test.data$DEATH\_EVENT, predicted)

```