SDS 291 Final Project Report

Miya Dang, Mia Tran, Alua Birgebayeva

2025-04-28

Abstract

This project analyzes the Heart Failure Clinical Records Dataset to identify key predictors of mortality among patients with heart failure. Using logistic regression and backward elimination, we developed a final model with five variables: age, ejection fraction, serum creatinine, serum sodium, and follow-up time. Model diagnostics confirmed the robustness of our approach, and performance metrics indicated strong predictive ability (sensitivity: 81.3%, specificity: 79.3%, AUC: 0.89). Our findings suggest that commonly available clinical variables can effectively predict patient outcomes and support early risk assessment in heart failure management.

Introduction

Heart failure is a leading cause of mortality and hospitalization worldwide, especially among older adults. Identifying patients at high risk of death can improve clinical decision-making and patient care. In this study, we aim to determine which clinical and demographic factors are most predictive of mortality among heart failure patients using the Heart Failure Clinical Records Dataset.

Our research question is: ***Which variables best predict death during the follow-up period after a heart failure diagnosis?***

We apply logistic regression with backward elimination to build a parsimonious and interpretable model. Prior studies have highlighted the roles of age, ejection fraction, and kidney function in heart failure outcomes (Choi et al., 2017; Ahmad et al., 2018), but many models lack transparency or validation. This project contributes a validated model using routine clinical measures and emphasizes model diagnostics to ensure reliability. Our goal is to support early risk assessment in heart failure using accessible patient data.

Methods

Dataset Description

The dataset we analyzed in this study is the Heart Failure Clinical Records Dataset, originally sourced from the UCI Machine Learning Repository. It contains the medical records of 299 patients

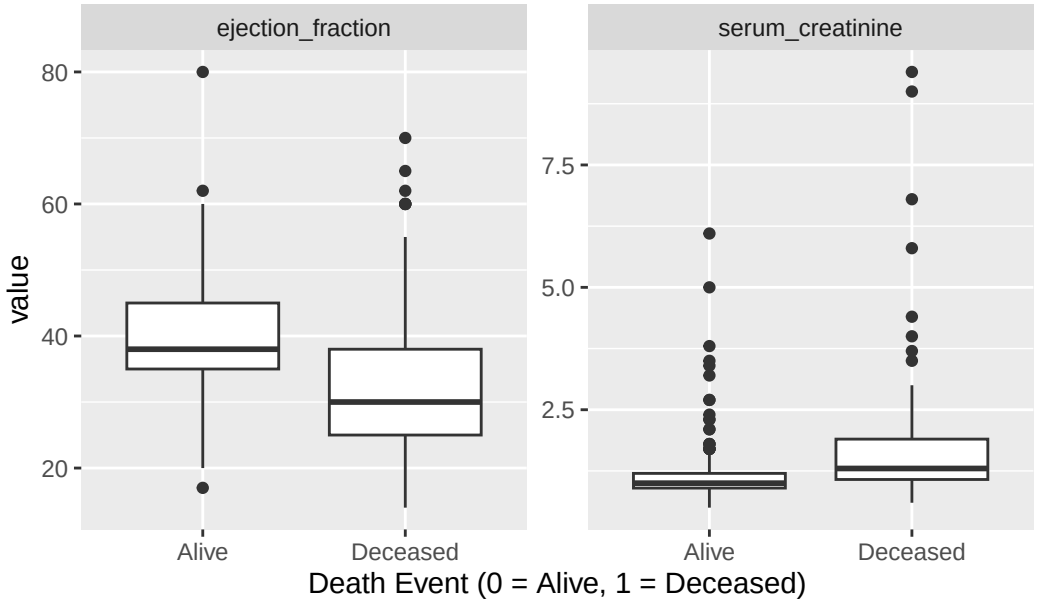
who experienced heart failure, collected during their clinical follow-up period. Each observation is a patient profile. The response variable, DEATH_EVENT, is a binary outcome indicating whether a patient died during the follow-up period (1 = deceased, 0 = alive). All eleven explanatory variables were initially considered, including demographic factors, clinical profile, as well as laboratory blood tests measurements. Details about each variable can be seen in Table 1. No missing data were reported in the dataset.

Data Processing and Exploratory Data Analysis

The dataset was imported into R, and all categorical variables were recoded into factors.

An exploratory data analysis was performed using summary statistics and visualizations. The median age of participants was 60 years (IQR: 51–70), and 32% died during the follow-up period. The median serum creatinine level was 1.10 mg/dL (IQR: 0.90–1.40), and the median ejection fraction was 38% (IQR: 30–45). Two boxplots were created to compare the distributions of ejection fraction (%) and serum creatinine (mg/dL) by death event (Figure 1). These plots indicated that patients who died tended to have lower ejection fractions and higher serum creatinine levels, supporting the inclusion of these variables in subsequent modeling.

Figure 1. Ejection Fraction and Serum Creatinine by Death Eve



Variable Selection

To identify potential predictors of mortality, we applied backward elimination to a full multiple logistic regression model containing 11 predictors. The selection process used nested F-tests with a retention threshold of $p < 0.1$. Variables with the highest p-values were removed sequentially until all remaining predictors had p-values below the threshold. Variables removed sequentially included anaemia, smoking, high_blood_pressure, diabetes, platelets, creatinine_phosphokinase, and sex.

Table 1: Summary of Heart Failure Dataset

Characteristic	**N = 299**
age: age of the patient (years)	60 (51, 70)
anaemia: decrease of red blood cells or hemoglobin (0 = No, 1 = Yes)	
0	170 (57%)
1	129 (43%)
creatinine_phosphokinase: level of the CPK enzyme in the blood (mcg/L)	250 (115, 582)
diabetes: if the patient has diabetes (0 = No, 1 = Yes)	
0	174 (58%)
1	125 (42%)
ejection_fraction: % of blood leaving the heart at each contraction (%)	38 (30, 45)
high_blood_pressure: if the patient has hypertension (0 = No, 1 = Yes)	
0	194 (65%)
1	105 (35%)
platelets: platelets in the blood (kiloplatelets/mL)	262,000 (212,000, 304,00
serum_creatinine: level of serum creatinine in the blood (mg/dL)	1.10 (0.90, 1.40)
serum_sodium: level of serum sodium in the blood (mEq/L)	137 (134, 140)
sex: 0 = Woman, 1 = Man	
0	105 (35%)
1	194 (65%)
smoking: if the patient smokes (0 = No, 1 = Yes)	
0	203 (68%)
1	96 (32%)
time: length of follow-up period (days)	115 (73, 205)
DEATH_EVENT: if the patient died during the follow-up period (0 = No, 1 = Yes)	
Alive	203 (68%)
Deceased	96 (32%)

[1] Median (Q1, Q3); n (%)

The final model retained five predictors: age, ejection_fraction, serum_creatinine, serum_sodium, and time.

The population form of the final model is defined as follows:

$$\begin{aligned}\text{logit}(P(\text{Deceased} = 1)) = \beta_0 + \beta_1\ (\text{Age}) + \beta_2\ (\text{Eject Frac}) + \beta_3\ (\text{Serum Creatinine}) \\ + \beta_4\ (\text{Serum Sodium}) + \beta_5\ (\text{Time})\end{aligned}$$

Where:

- $\text{logit}(P(\text{Deceased} = 1))$: The log-odds of the probability that the binary outcome "Deceased" equals 1.
- β_0: The intercept, the baseline log-odds of death when all predictors are zero. This is not particularly meaningful on its own, but it serves as a baseline for predictions.
- β_1: The slope for Age, the change in the log-odds of death for each 1 year increase in age, holding other variables constant.
- β_2: The slope for Ejection Fraction, the change in the log-odds of death for each 1% increase in ejection fraction (percentage of blood leaving the heart at each contraction), holding other variables constant.
- β_3: The slope for Serum Creatinine, the change in the log-odds of death for each 1 mg/dL increase in the level of serum creatinine in the blood, holding other variables constant.
- β_4: The slope for Serum Sodium, the change in the log-odds of death for each 1 mEq/L increase in the level of serum sodium in the blood, holding other variables constant.
- β_5: The slope for Time, the change in the log-odds of death for each 1 day increase in the length of the follow-up period, holding other variables constant.

Table 2: Coefficient Estimates from Final Model

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	9.4930	5.4058	1.7561	0.07907
age	0.0425	0.0150	2.8255	0.00472
ejection_fraction	-0.0734	0.0158	-4.6518	0.00000
serum_creatinine	0.6860	0.1740	3.9415	0.00008
serum_sodium	-0.0646	0.0384	-1.6822	0.09254
time	-0.0209	0.0029	-7.1657	0.00000

Model Diagnostics: Influential Points and Model Assumptions

We then investigated influential observations and assessed model assumptions to ensure the reliability of our logistic regression model. To identify unusual points, we examined leverage, studentized residuals, and Cook's distance for all observations. Observations with leverage values greater than $\frac{2(k+1)}{n}$ and studentized residuals > 2 or < -2 were flagged as potentially influential. Initially, we used a threshold of $\frac{4}{n}$ for Cook's distance, which flagged 29 observations as influential, which is

nearly 10% of the dataset. However, upon visual inspection of the Cook's distance plot, most of these points clustered closely with the rest of the data and did not appear to exert a disproportionate influence on the model. Only three observations (rows 132, 218, and 229) were clearly distant from the majority, with Cook's distance values exceeding 0.05. We therefore adopted 0.05 as a more appropriate threshold and focused on these three points, which were also consistently flagged across all diagnostic metrics. Further inspection revealed that these three observations had unusually high serum creatinine levels (6.1, 9.0, and 5.0). While these values are high, they are not uniquely extreme within the dataset as several other cases also exhibited elevated serum creatinine levels above 3.0. These high values are clinically plausible and likely reflect cases of severe kidney dysfunction, a condition frequently associated with heart failure and increased mortality risk. Therefore, there is no indication that these values are due to data entry or recording errors. Additionally, serum creatinine remained a statistically significant and clinically meaningful predictor in both the original model and the clean model that excluded them, with consistent effect sizes and p-values. Importantly, model performance metrics such as AIC, deviance, and coefficient estimates did not change substantially after excluding these observations. Based on this evidence, we concluded that the flagged cases represent valid and meaningful variation in the data and chose to retain them in the final model to maintain the integrity and generalizability of our findings.

We assessed multicollinearity among our predictors by calculating Variance Inflation Factors (VIFs). All VIFs fell between 1.03 and 1.13, indicating minimal correlation among the covariates and reassuring us that our coefficient estimates are stable and interpretable. Next, we generated a deviance-residual plot to evaluate model fit and again identify any unusual observations. The residuals are scattered randomly around zero without any clear pattern across the observation index, suggesting the model does not systematically over- or under-predict in different regions of the data. Moreover, all deviance residuals lie within the range of –3 to +3, and there are no points that stand apart from the main cluster. Together, these diagnostics confirm that our model assumptions hold and that no extreme outliers warrant removal.

Model Performance

We evaluated the model's performance using sensitivity, specificity, accuracy, and the area under the ROC curve (AUC). To reduce the risk of missing actual mortality cases, we selected a lower classification threshold of $\pi_0 = 0.3$, prioritizing sensitivity over specificity. Using this cutoff, we calculated the corresponding performance metrics to assess how well the model distinguishes between patients who survived and those who died.

- Sensitivity (0.8125): The model correctly identifies 81.3% of individuals who died (DEATH_EVENT = 1). This indicates strong performance in detecting patients at high risk of death, which is especially important in clinical settings where failing to identify at-risk patients could have serious consequences.
- Specificity (0.7931034): The model correctly classifies 79.3% of individuals who survived (DEATH_EVENT = 0). This means it is also reasonably effective at minimizing false positives, avoiding misclassifying living patients as deceased.
- Accuracy (0.7993311): The model achieves an overall accuracy of approximately 79.9%, meaning it correctly classifies nearly 80% of all cases, whether alive or dead. This suggests strong overall predictive ability.

- AUC (0.8935242): The model has excellent discriminative ability, with an AUC of 0.8935. This indicates that it can distinguish between patients who died and those who survived substantially better than random chance, and approaches the performance of a highly reliable classifier.

Results

The final analysis included all 299 patients in the dataset. Although three observations were flagged as potentially influential, they were retained in the model after further diagnostics. Out of these 299 patients, 32% died during the follow-up period.

The response variable, DEATH_EVENT, indicates whether a patient died during the follow-up period (1 = deceased, 0 = alive). The explanatory variables retained in the final model were age (in years), ejection fraction (percentage of blood leaving the heart per contraction), serum creatinine (mg/dL), serum sodium (mEq/L), and time (length of follow-up in days). These variables were selected through backward elimination using nested F-tests with a retention threshold of $p < 0.1$. The table below summarizes the estimated odds ratios, 95% confidence intervals, and p-values for each predictor in the model.

Table 3: Final Model: Odds Ratios and 95% Confidence Intervals

Predictor	Odds Ratio	95% CI (Low)	95% CI (High)	p-value
Age	1.043	1.014	1.076	0.005
Ejection Fraction	0.929	0.899	0.957	0.000
Serum Creatinine	1.986	1.421	2.874	0.000
Serum Sodium	0.937	0.868	1.011	0.093
Follow-up Time	0.979	0.973	0.985	0.000

Each predictor in the final model had a clear association with the probability of death.

- **Age**: Each additional year increased the odds of death by approximately 4.3% (OR = 1.043, 95% CI: [1.014, 1.076], p = 0.005).
- **Ejection Fraction**: Each 1% increase reduced the odds of death by 7.1% (OR = 0.929, 95% CI: [0.899, 0.957], $p < 0.001$).
- **Serum Creatinine**: Each 1 mg/dL increase more than doubled the odds of death (OR = 1.986, 95% CI: [1.421, 2.874], $p < 0.001$).
- **Serum Sodium**: While not statistically significant, the odds ratio suggests a 6.3% decrease in risk per unit increase (OR = 0.937, 95% CI: [0.868, 1.011], p = 0.093).
- **Follow-up Time**: Each additional day of follow-up was associated with a 2.1% decrease in the odds of death (OR = 0.979, 95% CI: [0.973, 0.985], $p < 0.001$).

To evaluate model performance, we selected a classification threshold of $\pi = 0.3$ to prioritize sensitivity. The model correctly identified 81.3% of patients who died (sensitivity) and 79.3% of patients who survived (specificity). Overall classification accuracy was 79.9%. The area under the ROC curve (AUC) was 0.8935, indicating excellent discriminative ability.

Figure 1 shows boxplots of ejection fraction and serum creatinine by survival outcome. Patients who died had noticeably lower ejection fractions and higher creatinine levels, reinforcing their inclusion in the final model.

These findings address our research question by identifying five routinely available variables that effectively predict mortality among patients with heart failure.

Discussion

This study set out to answer the question: *Which variables best predict death during the follow-up period after a heart failure diagnosis?* Using logistic regression and backward elimination, we identified five key predictors: age, ejection fraction, serum creatinine, serum sodium, and follow-up time.

Our results show that these variables are strong predictors of mortality. Older patients, those with lower ejection fractions, and those with higher serum creatinine levels were significantly more likely to die during follow-up. While serum sodium was not statistically significant at the 5% level ($p = 0.093$), its clinical relevance and borderline confidence interval supported its inclusion. Patients with longer follow-up times were less likely to die, suggesting that time itself may reflect survival resilience.

These findings directly answer our research question and are supported by both statistical metrics and clinical interpretability. As shown in Table 3, odds ratios for the significant predictors were all in directions consistent with medical expectations. Figure 1 further reinforced these relationships, visually demonstrating that patients who died had lower ejection fractions and higher serum creatinine. Model performance metrics, including a high AUC (0.8935), sensitivity (81.3%), and specificity (79.3%), indicate that the model performs well as a classification tool.

Despite these strengths, our analysis has several limitations. First, the dataset contains only 299 observations, which limits the generalizability of our findings. All data came from a single source, and we do not have access to variables such as treatment type, comorbidities, or socioeconomic status. These missing factors may introduce omitted variable bias. Additionally, our model assumes linearity in the log-odds and does not explore potential interactions or non-linear effects.

Still, the study has notable strengths. The final model is interpretable, relies on common clinical measures, and passed all major diagnostic checks (e.g., residuals, Cook's distance, VIFs). Our variable selection process used nested F-tests, and we retained cases flagged as influential only after confirming they did not distort results. These decisions reflect a balance between statistical rigor and real-world applicability.

In summary, this analysis provides strong evidence that five routine clinical measures can meaningfully predict death in heart failure patients. However, our conclusions are specific to the dataset at hand and should not be generalized without further validation. These findings offer a starting point for clinicians or researchers seeking to build early warning tools using accessible patient data.

Data Analysis Appendix

Import dataset, preparing data, load packages

```
# Loading necessary packages
library(kableExtra)
library(gtsummary)
library(ggplot2)
library(tidyr)
library(pROC)
library(car)
library(dplyr)
library(broom)

# Reading csv
heart_data <- read.csv("heart_failure_clinical_records_dataset.csv")

# Factoring categorical variables
heart_data$anaemia <- factor(heart_data$anaemia)
heart_data$diabetes <- factor(heart_data$diabetes)
heart_data$high_blood_pressure <- factor(heart_data$high_blood_pressure)
heart_data$sex <- factor(heart_data$sex)
heart_data$smoking <- factor(heart_data$smoking)
heart_data$DEATH_EVENT <- factor(heart_data$DEATH_EVENT, levels = c(0,1),
                         labels = c("Alive", "Deceased"))
```

EDA

```
# EDA summary table
tbl_summary(heart_data,
            label = list(
              age ~ "age: age of the patient (years)",
              anaemia ~ "anaemia: decrease of red blood cells or hemoglobin (0 =
 ↪  No, 1 = Yes)",
              creatinine_phosphokinase ~ "creatinine_phosphokinase: level of the
 ↪  CPK enzyme in the blood (mcg/L)",
              diabetes ~ "diabetes: if the patient has diabetes (0 = No, 1 =
 ↪  Yes)",
              ejection_fraction ~ "ejection_fraction: % of blood leaving the
 ↪  heart at each contraction (%)",
              high_blood_pressure ~ "high_blood_pressure: if the patient has
 ↪  hypertension (0 = No, 1 = Yes)",
              platelets ~ "platelets: platelets in the blood (kiloplatelets/mL)",
              sex ~ "sex: 0 = Woman, 1 = Man",
```

Table 4: Summary of Heart Failure Dataset

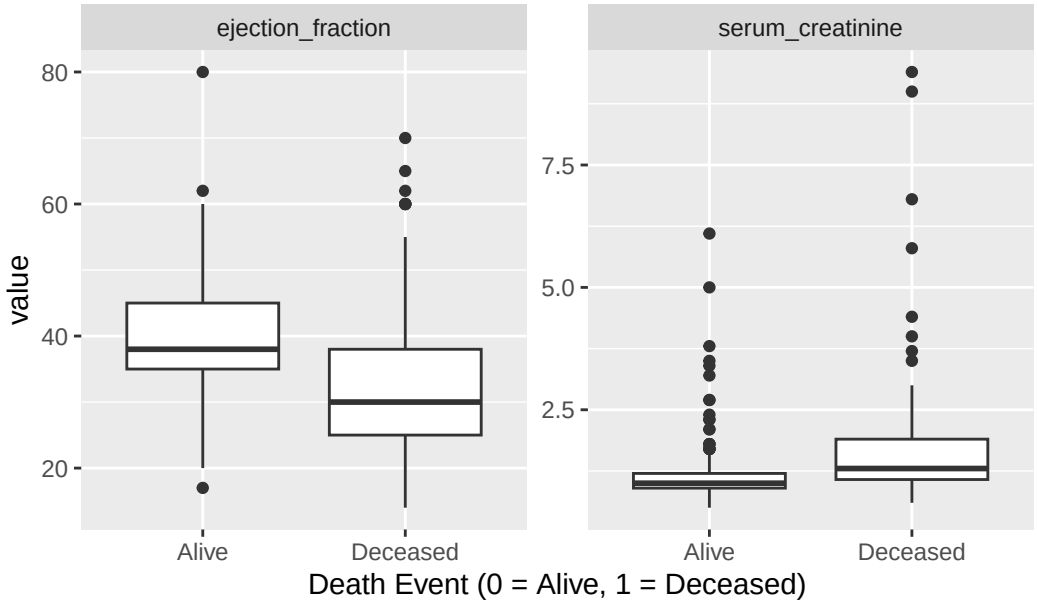
Characteristic	**N = 299**
age: age of the patient (years)	60 (51, 70)
anaemia: decrease of red blood cells or hemoglobin (0 = No, 1 = Yes)	
0	170 (57%)
1	129 (43%)
creatinine_phosphokinase: level of the CPK enzyme in the blood (mcg/L)	250 (115, 582)
diabetes: if the patient has diabetes (0 = No, 1 = Yes)	
0	174 (58%)
1	125 (42%)
ejection_fraction: % of blood leaving the heart at each contraction (%)	38 (30, 45)
high_blood_pressure: if the patient has hypertension (0 = No, 1 = Yes)	
0	194 (65%)
1	105 (35%)
platelets: platelets in the blood (kiloplatelets/mL)	262,000 (212,000, 304,00
serum_creatinine: level of serum creatinine in the blood (mg/dL)	1.10 (0.90, 1.40)
serum_sodium: level of serum sodium in the blood (mEq/L)	137 (134, 140)
sex: 0 = Woman, 1 = Man	
0	105 (35%)
1	194 (65%)
smoking: if the patient smokes (0 = No, 1 = Yes)	
0	203 (68%)
1	96 (32%)
time: length of follow-up period (days)	115 (73, 205)
DEATH_EVENT: if the patient died during the follow-up period (0 = No, 1 = Yes)	
Alive	203 (68%)
Deceased	96 (32%)

[1] Median (Q1, Q3); n (%)

```
            serum_creatinine ~ "serum_creatinine: level of serum creatinine in
↪  the blood (mg/dL)",
            serum_sodium ~ "serum_sodium: level of serum sodium in the blood
↪  (mEq/L)",
            smoking ~ "smoking: if the patient smokes (0 = No, 1 = Yes)",
            time ~ "time: length of follow-up period (days)",
            DEATH_EVENT ~ "DEATH_EVENT: if the patient died during the
↪  follow-up period (0 = No, 1 = Yes)"
          ))  %>%
 as_kable_extra(format = "latex", booktabs = TRUE, caption = "Summary of Heart
  ↪  Failure Dataset")
```

```r
# Accompanying graph
# Reshape data to long format
heart_data_long <- heart_data %>%
  pivot_longer(
    cols = c(ejection_fraction, serum_creatinine),
    names_to = "variable",
    values_to = "value"
  )
# Create labels for faceting
variable_labels <- c(
  "ejection_fraction" = "Ejection Fraction (%)",
  "serum_creatinine" = "Serum Creatinine (mg/dL)"
)
# Create combined plot
ggplot(heart_data_long, aes(x = factor(DEATH_EVENT), y = value)) +
  geom_boxplot() +
  facet_wrap(~ variable, scales = "free_y") +
  labs(x = "Death Event (0 = Alive, 1 = Deceased)",
       title = "Figure 1. Ejection Fraction and Serum Creatinine by Death Event"
  )
```

Figure 1. Ejection Fraction and Serum Creatinine by Death Eve



Select variables

```r
# Selecting variables by doing backward elimination using nested F test (p-value
↪  = 0.1)
```

```r
heart_full <- glm(DEATH_EVENT ~ age + anaemia + creatinine_phosphokinase +
 ↪  diabetes + ejection_fraction + high_blood_pressure + platelets +
 ↪  serum_creatinine + serum_sodium + sex + smoking + time, data = heart_data,
 ↪  family = "binomial")

# drop1(heart_full, test = "F")
drop1_heart <- update(heart_full, . ~ . - anaemia)
# drop1(drop1_heart, test = "F")
drop1_heart <- update(drop1_heart, . ~ . - smoking)
# drop1(drop1_heart, test = "F")
drop1_heart <- update(drop1_heart, . ~ . - high_blood_pressure)
# drop1(drop1_heart, test = "F")
drop1_heart <- update(drop1_heart, . ~ . - diabetes)
#  drop1(drop1_heart, test = "F")
drop1_heart <- update(drop1_heart, . ~ . - platelets)
#  drop1(drop1_heart, test = "F")
drop1_heart <- update(drop1_heart, . ~ . - creatinine_phosphokinase)
# drop1(drop1_heart, test = "F")
drop1_heart <- update(drop1_heart, . ~ . - sex)
# drop1(drop1_heart, test = "F")

# Final model after moving variables
final_model <- drop1_heart

final_model_table <- summary(final_model)$coefficients
final_model_table |>
  kbl(col.names = c("Estimate", "Std. Error", "t value", "Pr(>|t|)"),
      align = "c",
      booktabs = TRUE,
      linesep = "",
      digits = c(4, 4, 4, 5),
      caption = "Coefficient Estimates from Final Model") |>
  kable_classic(full_width = FALSE, latex_options = c("HOLD_position"))
```

Table 5: Coefficient Estimates from Final Model

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	9.4930	5.4058	1.7561	0.07907
age	0.0425	0.0150	2.8255	0.00472
ejection_fraction	-0.0734	0.0158	-4.6518	0.00000
serum_creatinine	0.6860	0.1740	3.9415	0.00008
serum_sodium	-0.0646	0.0384	-1.6822	0.09254
time	-0.0209	0.0029	-7.1657	0.00000

Model diagnose

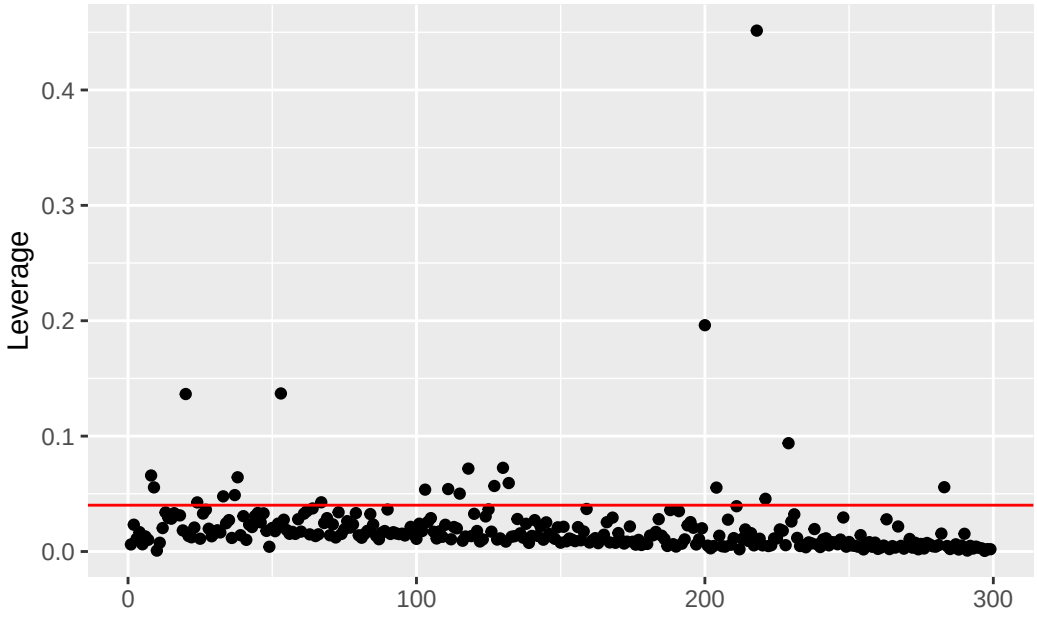
```
# Model diagnostics
# Cooks Distance


# Influential points

# Leverage
case_influence <- final_model |> augment()
case_influence <- case_influence |> mutate(row_id = row_number())

# Calculating the threshold for unusually high leverage
k_plus_one <- length(coef(final_model))
n <- nrow(heart_data)

# Filtering the data to determine which observations have unusually high leverage
leverage_index <- case_influence |> filter(.hat > 2 * k_plus_one/n) |>
 ↪  select(row_id) |> pull()
heart_data[leverage_index, ]
```

```
    age anaemia creatinine_phosphokinase diabetes ejection_fraction
8    60       1                      315        1                60
9    65       0                      157        0                65
20   48       1                      582        1                55
24   53       0                       63        1                60
33   50       1                      249        1                35
37   90       1                       60        1                50
38   82       1                      855        1                50
53   60       0                     3964        1                62
67   42       1                      250        1                15
103  80       0                      898        0                25
111  85       0                      129        0                60
115  60       1                      754        1                40
118  85       1                      102        0                60
127  46       0                      168        1                17
130  53       1                      270        1                35
132  60       1                     1082        1                45
200  60       0                     1211        1                35
204  60       0                       59        0                25
218  54       1                      427        0                70
221  73       0                      582        0                20
229  65       0                       56        0                25
283  42       0                       64        0                30
    high_blood_pressure platelets serum_creatinine serum_sodium sex smoking
8                     0    454000             1.10          131   1       1
```

```
9                   0    263358             1.50          138   0        0
20                  0     87000             1.90          121   0        0
24                  0    368000             0.80          135   1        0
33                  1    319000             1.00          128   0        0
37                  0    226000             1.00          134   1        0
38                  1    321000             1.00          145   0        0
53                  0    263358             6.80          146   0        0
67                  0    213000             1.30          136   0        0
103                 0    149000             1.10          144   1        1
111                 0    306000             1.20          132   1        1
115                 1    328000             1.20          126   1        0
118                 0    507000             3.20          138   0        0
127                 1    271000             2.10          124   0        0
130                 0    227000             3.40          145   1        0
132                 0    250000             6.10          131   1        0
200                 0    263358             1.80          113   1        1
204                 1    212000             3.50          136   1        1
218                 1    151000             9.00          137   0        0
221                 0    263358             1.83          134   1        0
229                 0    237000             5.00          130   0        0
283                 0    215000             3.80          128   1        1
    time DEATH_EVENT
8     10    Deceased
9     10    Deceased
20    15    Deceased
24    22       Alive
33    28    Deceased
37    30    Deceased
38    30    Deceased
53    43    Deceased
67    65    Deceased
103   87       Alive
111   90    Deceased
115   91       Alive
118   94       Alive
127  100    Deceased
130  105       Alive
132  107       Alive
200  186       Alive
204  187       Alive
218  196    Deceased
221  198    Deceased
229  207       Alive
283  250       Alive
```

```
case_influence |> ggplot(aes(x = row_id, y = .hat)) + geom_point() +
  geom_hline(yintercept = 2*k_plus_one/n, col = "red") +
```

```
  xlab("") + ylab("Leverage")
```

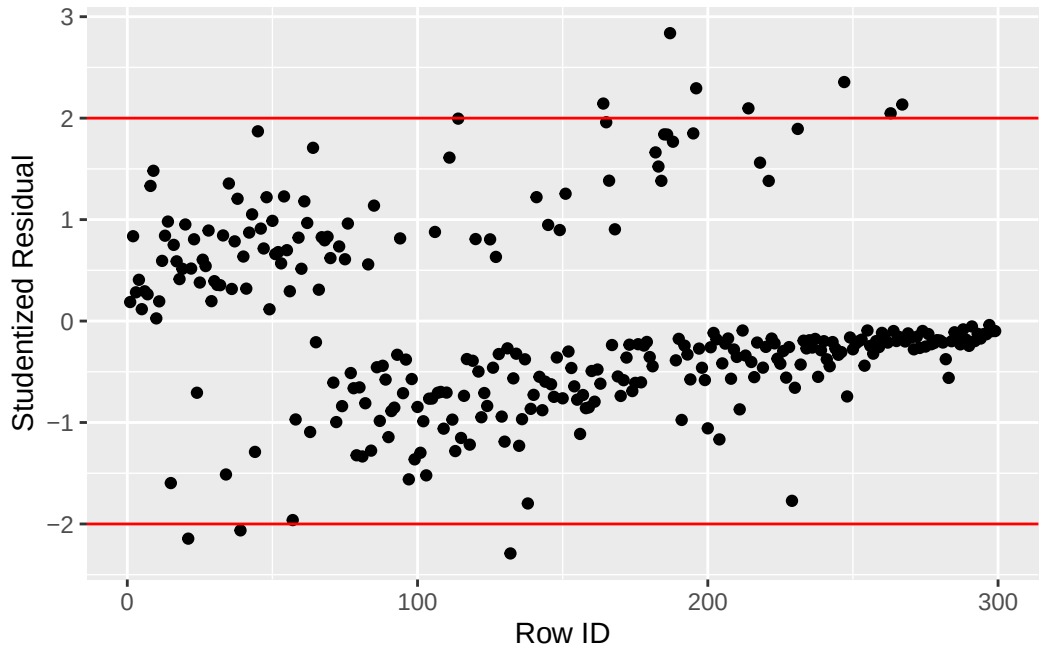



```
# Studentized residuals
case_influence <- case_influence |> mutate(.stu.resid = rstudent(final_model))
stu_resid_id <- case_influence |> filter(.stu.resid < -2 | .stu.resid > 2) |>
 ↪  select(row_id) |> pull()
heart_data[stu_resid_id, ]
```

```
    age anaemia creatinine_phosphokinase diabetes ejection_fraction
21   65       1                      52        0                25
39   60       0                    2656        1                30
132  60       1                    1082        1                45
164  50       1                    2334        1                35
187  50       0                     582        0                50
196  77       1                     418        0                45
214  48       1                     131        1                30
247  55       0                    2017        0                25
263  65       1                     258        1                25
267  55       0                    1199        0                20
    high_blood_pressure platelets serum_creatinine serum_sodium sex smoking
21                    1    276000             1.30          137   0       0
39                    0    305000             2.30          137   1       0
132                   0    250000             6.10          131   1       0
164                   0     75000             0.90          142   0       0
187                   0    153000             0.60          134   0       0
196                   0    223000             1.80          145   1       0
```

```
214                  1    244000              1.60          130    0        0
247                  0    314000              1.10          138    1        0
263                  0    198000              1.40          129    1        0
267                  0    263358              1.83          134    1        1
    time DEATH_EVENT
21    16       Alive
39    30       Alive
132  107       Alive
164  126    Deceased
187  172    Deceased
196  180    Deceased
214  193    Deceased
247  214    Deceased
263  235    Deceased
267  241    Deceased
```

```
# Plotting the studentized residuals against the observation row numbers
case_influence |>
  ggplot(aes(x = row_id, y = .stu.resid)) + geom_point() +
  geom_hline(yintercept = -2, col = "red") +
  geom_hline(yintercept = 2, col = "red") +
  xlab("Row ID") + ylab("Studentized Residual")
```




```
# Determining which observations have unusually large studentized residuals
stu_resid_id <- case_influence |>
  filter(.stu.resid < -2 | .stu.resid > 2) |>
  select(row_id) |>
```

```
  pull()

heart_data[stu_resid_id, ]
```

```
    age anaemia creatinine_phosphokinase diabetes ejection_fraction
21   65       1                      52        0                25
39   60       0                    2656        1                30
132  60       1                    1082        1                45
164  50       1                    2334        1                35
187  50       0                     582        0                50
196  77       1                     418        0                45
214  48       1                     131        1                30
247  55       0                    2017        0                25
263  65       1                     258        1                25
267  55       0                    1199        0                20
    high_blood_pressure platelets serum_creatinine serum_sodium sex smoking
21                    1    276000             1.30          137   0       0
39                    0    305000             2.30          137   1       0
132                   0    250000             6.10          131   1       0
164                   0     75000             0.90          142   0       0
187                   0    153000             0.60          134   0       0
196                   0    223000             1.80          145   1       0
214                   1    244000             1.60          130   0       0
247                   0    314000             1.10          138   1       0
263                   0    198000             1.40          129   1       0
267                   0    263358             1.83          134   1       1
    time DEATH_EVENT
21    16       Alive
39    30       Alive
132  107       Alive
164  126    Deceased
187  172    Deceased
196  180    Deceased
214  193    Deceased
247  214    Deceased
263  235    Deceased
267  241    Deceased
```

```
# Cook's distance
case_influence |> select(.cooksd)
```

```
# A tibble: 299 x 1
        .cooksd
          <dbl>
 1 0.0000183
```

```
 2 0.00167
 3 0.0000777
 4 0.000247
 5 0.00000726
 6 0.0000977
 7 0.0000608
 8 0.0166
 9 0.0191
10 0.0000000448
# i 289 more rows
```

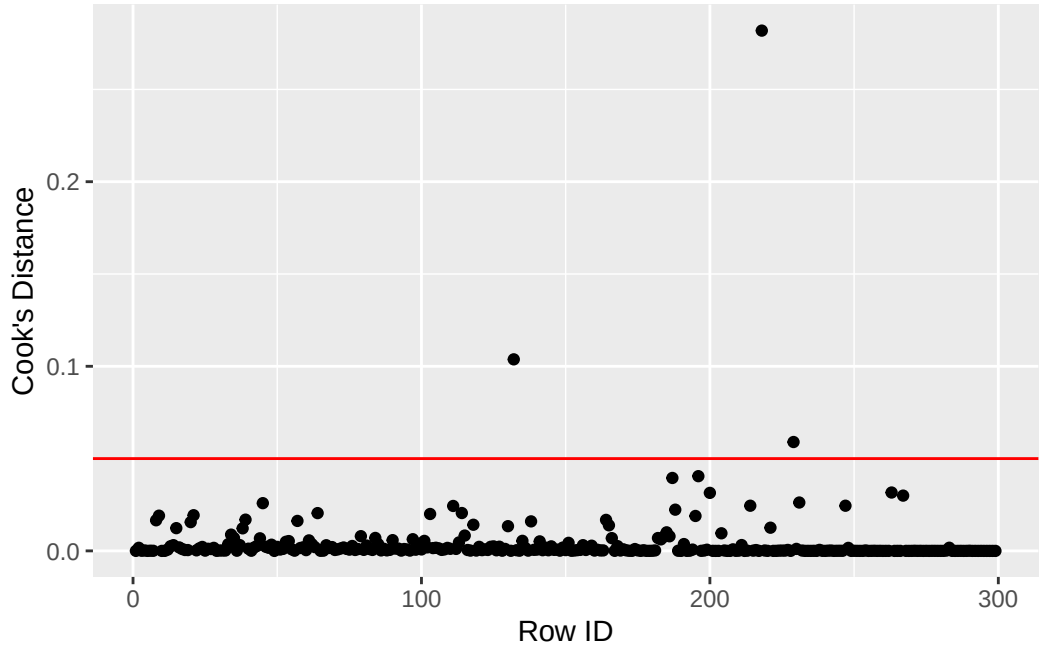
```
case_influence |> filter(.cooksd > 0.05)
```

```
# A tibble: 3 x 14
  DEATH_EVENT   age ejection_fraction serum_creatinine serum_sodium  time
  <fct>       <dbl>             <int>            <dbl>        <int> <int>
1 Alive          60                45              6.1          131   107
2 Deceased       54                70              9            137   196
3 Alive          65                25              5            130   207
# i 8 more variables: .fitted <dbl>, .resid <dbl>, .hat <dbl>, .sigma <dbl>,
#   .cooksd <dbl>, .std.resid <dbl>, row_id <int>, .stu.resid <dbl>
```

```
cooksd_id <- case_influence |> filter(.cooksd > 0.05) |> select(row_id) |> pull()
heart_data[cooksd_id, ]
```

```
    age anaemia creatinine_phosphokinase diabetes ejection_fraction
132  60       1                     1082        1                45
218  54       1                      427        0                70
229  65       0                       56        0                25
    high_blood_pressure platelets serum_creatinine serum_sodium sex smoking
132                   0    250000              6.1          131   1       0
218                   1    151000              9.0          137   0       0
229                   0    237000              5.0          130   0       0
    time DEATH_EVENT
132  107       Alive
218  196    Deceased
229  207       Alive
```

```
# Plotting the Cook's distances against the observation row numbers
case_influence |>
  ggplot(aes(x = row_id, y = .cooksd)) +
  geom_point() +
  geom_hline(yintercept = 0.05, col = "red") +
  xlab("Row ID") + ylab("Cook's Distance")
```




```r
# Define the row indices of the influential observations
influential_rows <- c(132, 218, 229)

# Remove these rows from the heart_data dataset
heart_data_clean <- heart_data[-influential_rows, ]

# Refit the logistic regression model on the cleaned data
final_model_clean <- glm(DEATH_EVENT ~ age + anaemia + creatinine_phosphokinase +
 ↪  diabetes + ejection_fraction + high_blood_pressure + platelets +
 ↪  serum_creatinine + serum_sodium + sex + smoking + time, data =
 ↪  heart_data_clean, family = "binomial")

# Summarize the new model
summary(final_model_clean)
```

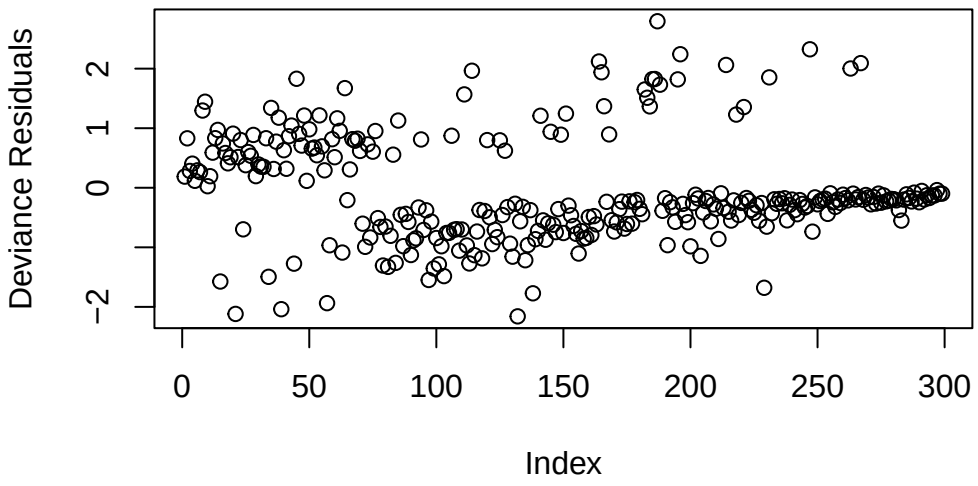
```
Call:
glm(formula = DEATH_EVENT ~ age + anaemia + creatinine_phosphokinase +
    diabetes + ejection_fraction + high_blood_pressure + platelets +
    serum_creatinine + serum_sodium + sex + smoking + time, family = "binomial",
    data = heart_data_clean)

Coefficients:
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)              1.169e+01  5.895e+00   1.984  0.04727 *
age                      4.913e-02  1.642e-02   2.993  0.00276 **
anaemia1                 3.327e-03  3.739e-01   0.009  0.99290
```

```
creatinine_phosphokinase  2.349e-04  1.844e-04   1.273  0.20285
diabetes1                 2.052e-01  3.597e-01   0.570  0.56847
ejection_fraction        -8.256e-02  1.775e-02  -4.652 3.28e-06 ***
high_blood_pressure1     -2.114e-01  3.749e-01  -0.564  0.57286
platelets                -1.010e-06  1.928e-06  -0.524  0.60038
serum_creatinine          8.737e-01  2.931e-01   2.981  0.00287 **
serum_sodium             -7.895e-02  4.104e-02  -1.924  0.05439 .
sex1                     -5.389e-01  4.266e-01  -1.263  0.20659
smoking1                 -6.404e-02  4.229e-01  -0.151  0.87963
time                     -2.144e-02  3.126e-03  -6.860 6.88e-12 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 371.53  on 295  degrees of freedom
Residual deviance: 209.31  on 283  degrees of freedom
AIC: 235.31

Number of Fisher Scoring iterations: 6
```

```r
# Multicollinearity
vif(final_model)
```

```
             age ejection_fraction  serum_creatinine      serum_sodium
        1.053111          1.133484          1.079122          1.028355
            time
        1.096862
```

```r
# Checking conditions using Deviance residual plot
plot(residuals(final_model, type = "deviance"), ylab = "Deviance Residuals")
```



Model performance

```
new_data <- read.csv("heart_failure_clinical_records_dataset.csv")
new_data$anaemia <- factor(new_data$anaemia)
new_data$diabetes <- factor(new_data$diabetes)
new_data$high_blood_pressure <- factor(new_data$high_blood_pressure)
new_data$sex <- factor(new_data$sex)
new_data$smoking <- factor(new_data$smoking)
new_data$DEATH_EVENT <- factor(new_data$DEATH_EVENT, levels = c(0,1),
                          labels = c("Alive", "Deceased"))

# Get predicted probabilities for new data
new_pred <- augment(final_model, newdata = new_data, type.predict = "response")

# Classify using 0.3 threshold
new_classify <- new_pred |>
  mutate(pred = ifelse(.fitted > 0.3, "Deceased", "Alive"))

# Actual vs. Predicted table with margins
compare_table <- new_classify |>
  select(DEATH_EVENT, pred) |>
  table() |>
  addmargins()

# Calculate sensitivity, specificity, accuracy
sensitivity <- compare_table[2, 2] / compare_table[2, 3]
```

```r
specificity <- compare_table[1, 1] / compare_table[1, 3]
accuracy <- (compare_table[1, 1] + compare_table[2, 2]) / compare_table[3, 3]


# Create ROC object
voting_roc <- roc(
  response = new_classify$DEATH_EVENT,
  predictor = new_classify$.fitted,
  quiet = TRUE
)

# Calculate AUC
auc_value <- auc(voting_roc)

# Results
cat("Sensitivity:", sensitivity, "\n")
```

```
Sensitivity: 0.8125
```

```r
cat("Specificity:", specificity, "\n")
```

```
Specificity: 0.7931034
```

```r
cat("Accuracy:", accuracy, "\n")
```

```
Accuracy: 0.7993311
```

```r
cat("AUC:", auc_value, "\n")
```

```
AUC: 0.8935242
```

Results section

```r
broom::tidy(final_model, conf.int = TRUE, exponentiate = TRUE) %>%
  filter(term != "(Intercept)") %>%
mutate(term = dplyr::recode(term,
  "age" = "Age",
  "ejection_fraction" = "Ejection Fraction",
  "serum_creatinine" = "Serum Creatinine",
  "serum_sodium" = "Serum Sodium",
  "time" = "Follow-up Time"
```

```
)) %>%
  select(term, estimate, conf.low, conf.high, p.value) %>%
  knitr::kable(
    col.names = c("Predictor", "Odds Ratio", "95% CI (Low)", "95% CI (High)",
     ↪  "p-value"),
    digits = 3,
    booktabs = TRUE,
    caption = "Final Model: Odds Ratios and 95\\% Confidence Intervals"
  ) %>%
  kable_classic(full_width = FALSE, latex_options = "HOLD_position")
```

Table 6: Final Model: Odds Ratios and 95% Confidence Intervals

Predictor	Odds Ratio	95% CI (Low)	95% CI (High)	p-value
Age	1.043	1.014	1.076	0.005
Ejection Fraction	0.929	0.899	0.957	0.000
Serum Creatinine	1.986	1.421	2.874	0.000
Serum Sodium	0.937	0.868	1.011	0.093
Follow-up Time	0.979	0.973	0.985	0.000