Heart Failure Survival Study

Mailman School of Public Health at Columbia University

Yi Huang, Runze Cui, Xuesen Zhao, Huanyu Chen, Jiahe Deng
P8108 Final Project Report: Group 6

Dec 11, 2023

Abstract

Heart failure (HF) results from weakened heart muscles, impairing blood pumping and causing symptoms like breathlessness (Ahmad et al., 2017). Statistics show HF affects 1-2% of adults, especially those over 70, potentially higher due to misdiagnosis (Jones et al., 2019). HF prevalence increased by 25% since 2002 due to aging, improved survival, and risk factors. This study utilizes data from the Institute of Cardiology and Allied Hospital in Faisalabad, Pakistan, which previously investigated the impact of key physiological and clinical factors on the prognosis of heart failure (HF) patients between April and December 2015. Our research employs a comprehensive range of statistical methodologies, including exploratory data analysis, non-parametric approaches, semi-parametric model, parametric models, hypothesis testing, model checking, and validation procedures, aiming to identify significant predictors and investigate their influences on heart failure. The significant factors identified in our study include creatinine levels, age, blood pressure, and serum sodium. Additionally, the model's discriminative ability across varied samples and conditions is validated using rigorous bootstrap validation methods such as c-index and calibration slope. Our findings contribute to refining risk assessment models, enhancing clinical decision-making, and optimizing patient care for heart failure in the future. The insights gained from our modeling process offer a deeper understanding of HF progression and its risk factors, paving the way for more personalized treatment approaches and preventive strategies that could profoundly impact patient outcomes and quality of life.

Contents

1 Introduction

1.1 Background

Heart failure (HF) occurs when the muscles in the heart wall weaken and enlarge, impairing the heart's ability to pump blood effectively. This condition can cause the heart's ventricles to become stiff, hindering their ability to fill properly between beats. Over time, the heart becomes less capable of meeting the body's demand for blood, leading to symptoms like difficulty in breathing as the heart struggles to function efficiently.(Ahmad et al., 2017).

According to the statistics, heart failure affects 1-2% of adults in the general population and is more common in older individuals, with over 10% of those aged over 70 years being diagnosed. The actual prevalence might be as high as 4%, as heart failure is often undiagnosed or misdiagnosed, especially in the elderly. Since 2002, the prevalence of heart failure has increased by nearly 25%, driven by factors such as an aging population, better survival rates post-coronary events, and a rise in risk factors like hypertension and atrial fibrillation. (Jones et al., 2019).

1.2 Objective

Our project aims to assess the influence of key physiological and clinical factors on the outcomes of heart failure patients at the Institute of Cardiology and Allied Hospital, Faisalabad, Pakistan, during April-December 2015. We will examine variables such as creatinine levels, gender, age, ejection fraction, blood pressure, anemia, and serum sodium to determine their impact on patient prognosis. Utilizing a range of analytic techniques including exploratory data analysis, nonparametric methods, hypothesis testing, semi-parametric modeling, parametric survival models, model checking, and validation procedures, our study is designed to identify crucial predictors of heart failure and to investigate their influences on the outcome. The insights gained from this analysis are expected to contribute significantly to the development of tailored treatment strategies and improved risk stratification models, thereby enhancing clinical decision-making and patient care for heart failure management.

2 Exploratory Data Analysis (EDA)

The present study focuses on 299 heart failure patients, including 105 women and 194 men. All participants were over 40 years old and diagnosed with left ventricular systolic dysfunction, classified under NYHA classes III and IV. The follow-up duration ranged from 4 to 285 days, with an average of 130 days. Diagnosis of the disease was confirmed through cardiac echo-cardiogram reports or physician's notes. A brief description of variables in the dataset is shown below:

- **age**: Age in years
- **time**: Survival time in days
- **event**: Event binary indicator (0 = Censored, 1 = Event)
- **gender**: Sex binary indicator (0 = Female, 1 = Male)
- **smoking**: Smoking status (0 = No smoking, 1 = Smoking)
- **diabetes**: Diabetes status (0 = No diabetes, 1 = Diabetes)
- **bp**: Blood pressure status (0 = Normal, 1 = Hypertension)
- **anemia**: Anemia status (0 = No anemia, 1 = Anemia: patients with haematocrit < 36)
- **EF_cat**: Ejection fraction (Low: EF ≤ 30, Medium: $30 <$ EF ≤ 45 and High: EF > 45)
- **sodium**: Sodium in mEq/L
- **creatinine**: Serum creatinine in mg/dL
- **platelets**: Platelets in mcL
- **cpk**: Creatinine phosphokinase in U/L

This study falls into the category of Overall Survival (OS), where event indicator equals 1 indicates the death of the subject and is the endpoint of survival. Specifically, 203 subjects were right-censored and 96 subjects have event. Detailed descriptive statistics table stratified by survival status are presented below.

Table 1: Descriptive Statistics Table for Variable Characteristic

	Censored	Event	Overall	P-value
	(N=203)	(N=96)	(N=299)	
Survival time (days)				
Mean (SD)	158 (67.7)	70.9 (62.4)	130 (77.6)	<0.001
Median [Min, Max]	172 [12.0, 285]	44.5 [4.00, 241]	115 [4.00, 285]	
Age (years)				
Mean (SD)	58.8 (10.6)	65.2 (13.2)	60.8 (11.9)	<0.001
Median [Min, Max]	60.0 [40.0, 90.0]	65.0 [42.0, 95.0]	60.0 [40.0, 95.0]	
Gender				
0	71 (35.0%)	34 (35.4%)	105 (35.1%)	1
1	132 (65.0%)	62 (64.6%)	194 (64.9%)	
Smoking status				
0	137 (67.5%)	66 (68.8%)	203 (67.9%)	0.932
1	66 (32.5%)	30 (31.3%)	96 (32.1%)	
Diabetes				
0	118 (58.1%)	56 (58.3%)	174 (58.2%)	1
1	85 (41.9%)	40 (41.7%)	125 (41.8%)	
Blood Pressure				
0	137 (67.5%)	57 (59.4%)	194 (64.9%)	0.214
1	66 (32.5%)	39 (40.6%)	105 (35.1%)	
Ejection Fraction (EF_cat)				
Low	42 (20.7%)	51 (53.1%)	93 (31.1%)	<0.001
Medium	115 (56.7%)	31 (32.3%)	146 (48.8%)	
High	46 (22.7%)	14 (14.6%)	60 (20.1%)	
Anemia				
0	120 (59.1%)	50 (52.1%)	170 (56.9%)	0.307
1	83 (40.9%)	46 (47.9%)	129 (43.1%)	
Serum Sodium (mEq/L)				
Mean (SD)	137 (3.98)	135 (5.00)	137 (4.41)	0.002
Median [Min, Max]	137 [113, 148]	136 [116, 146]	137 [113, 148]	
Serum creatinine (mg/dL)				
Mean (SD)	1.18 (0.654)	1.84 (1.47)	1.39 (1.03)	<0.001
Median [Min, Max]	1.00 [0.500, 6.10]	1.30 [0.600, 9.40]	1.10 [0.500, 9.40]	
Creatinine phosphokinase (U/L)				
Mean (SD)	540 (754)	670 (1320)	582 (970)	0.369
Median [Min, Max]	245 [30.0, 5210]	259 [23.0, 7860]	250 [23.0, 7860]	
Plateletes (mcL)				
Mean (SD)	267000 (97500)	256000 (98500)	263000 (97800)	0.399
Median [Min, Max]	263000 [25100, 850000]	259000 [47000, 621000]	262000 [25100, 850000]	

Based on the descriptive table, we observe that the mean survival time for deletions and events is 158 days and 70.9 days, respectively. Since we have the complete dataset, there is no need to worry about missing values issue. The table also lists the p-values for each variable with some having relatively large p-values. However, we still need to check the distribution of each variable [1] and determine which variables need to be transformed.

[1] Histograms for continuous variables and bar charts for categorical variables

Figure 1.1: Histograms of Continuous Variables

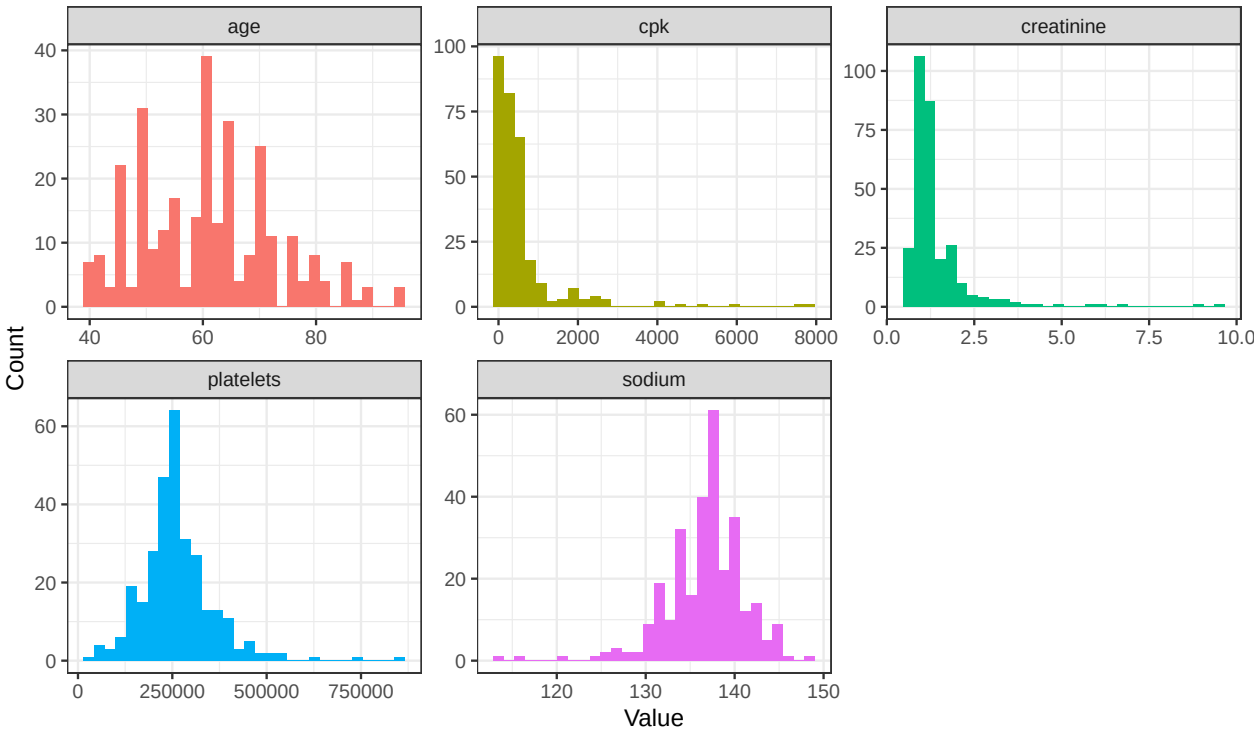
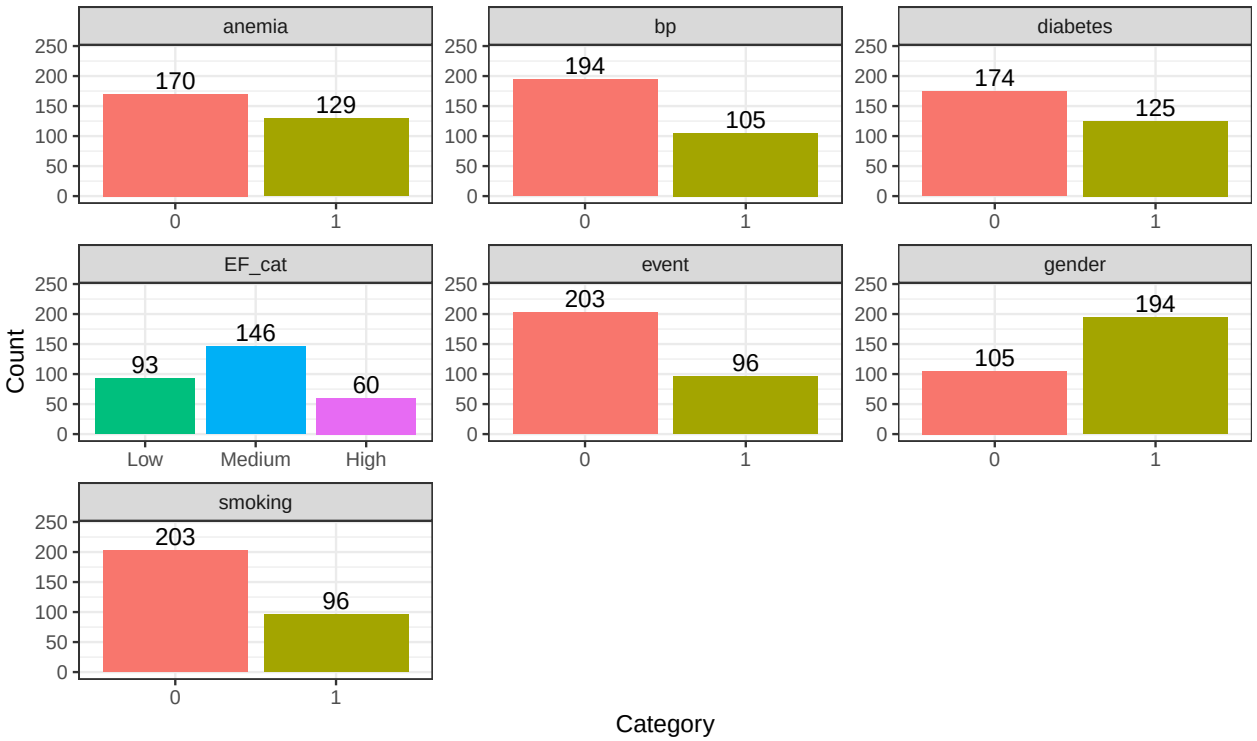


Figure 1.2: Bar Charts of Categorical Variables



After checking the histograms, two continuous variables creatinine phosphokinase (cpk) and serum creatinine are right-skewed. We decide to log-transformed both of them for further model fitting. The new log-transformed variables follow a little more symmetric-like distribution and are stored as `logcre` and `logcpk`.

3 Methods

3.1 Nonparametric Methods

In our analysis, we applied life table, Kaplan-Meier and Fleming-Harrington Curves to estimate the survival function. All three approaches can handle censored data.

The life table is particularly useful for larger sample sizes and when the data are grouped into intervals. The survival probability at each interval is estimated as:

$$\hat{S}_L(t_i) = \prod_{t_{i-1}<t} (1 - \frac{d_i}{n_i'})$$

where d_i is the number of events in interval[2] $[t_{i-1}, t_i]$, n_i' is the average number at risk in the interval $[t_{i-1}, t_i]$.

The Kaplan-Meier curve allows for varying follow-up times and censored data, making it versatile for smaller samples and individual subject data. It provides a visual representation of the survival experience of the cohort over time. The Kaplan-Meier estimate of survival function can be mathematically expressed as:

$$\hat{S}_K(t) = \prod_{t_i \leq t} [1 - \frac{d_i}{n_i}]$$

where d_i is the number of events at time t_i and n_i is the number at risk at t_i^-, and c_i is the number of censored during the interval $[t_i, t_{i+1}]$

Unlike the Kaplan-Meier estimator, the Fleming-Harrington estimator[3] is designed to weight events differently over time in survival analysis. It focuses on estimating the cumulative hazard function. The estimated survival probability can be computed as:

$$\hat{S}_F(t) = \prod_{t_i \leq t} \exp[-\frac{d_i}{n_i}]$$

where the d_i, n_i and c_i conditions are the same as K-M estimator above, It is true that $\hat{S}_F(t) \geq \hat{S}_K(t)$ because $\exp[-\frac{d_i}{n_i}]$ is always great and equal to $1 - \frac{d_i}{n_i}$.

Both K-M and F-H survival curves are presented in the **Result** section for further comparison.

3.2 Hypothesis Testing

The Log-Rank test focuses on comparing the number of observed to expected events across the groups at each time point. The test statistic[4] is calculated as:

$$\frac{L}{\sqrt{\text{Var}(L)}} = \frac{\sum_{i=1}^{k}(d_{0i} - e_{0i})}{\sum_{i=1}^{k}\sqrt{\frac{n_{0i}n_{1i}d_i(n_i - d_i)}{n_i^2(n_i - 1)}}} \sim N(0,1)$$

The Gehan's Wilcoxon test[5] gives more weight to events at earlier time points. It achieves a greater sensitivity to differences in survival that manifest at the beginning of the observation period. The test statistic is calculated as:

[2]This is the survival function at the end of interval. However, R often reports the survival function at the beginning of the interval

[3]Also known as Nelson-Aalen estimator

[4]This is the log-rank test statistics with tie

[5]Also known as the Breslow test

$$\frac{L}{\sqrt{\text{Var}(L)}} = \frac{\sum_{i=1}^{k} n_i(d_{0i} - e_{0i})}{\sum_{i=1}^{k} \sqrt{\frac{n_{0i} n_{1i} d_i (n_i - d_i)}{n_i - 1}}} \sim N(0,1)$$

The Gehan's Wilcoxon test is actually a special case of weighted log-rank test for weight equals n_i.

3.3 Proportional Hazard Models

We use two proportional hazard models to evaluate the effect of several factors on survival time. It allows us to examine how specified factors influence the rate of the event that we are interested in at a particular point in time. The rate here is the hazard rate. Proportional hazard model is the primary regression model to investigate the effectiveness of treatment X over survival time T, where the i_{th} patient at a time t is

$$h_i(t) = h_0(t) \exp[\beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_p X_{ip}]$$

where

- $h_0(t)$ is the baseline hazard function
- $h_i(t)$ is the hazard function determined by a set of p covariates $(X_{i1},\ X_{i2},\ ...,\ X_{ip})$
- $(\beta_1,\ \beta_2,\ ...,\ \beta_p)$ are the coefficients which measure the impact of covariates.

The **proportional hazard** can be expressed as ratio of two hazard functions at time t in two individuals or groups with covariates X and X', and does not depend on t.

$$\frac{h(t|X = x)}{h(t|X = x')} = e^{\beta(x - x')}$$

There are different ways to formulate the baseline hazard function $h_0(t)$, which lead to different models and estimations.

Cox Proportational Hazard Model

The Cox proportional hazard model is a semi-parametric model which does not assume a particular baseline hazard function $\tilde{h}_0(t)$. In contrast to parametric PH models which use the full likelihood, the Cox PH model is formulated by partial likelihood because there is very limited information on β beyond $L_p(\beta)$. The model is given by:

$$h_i(t) = \tilde{h}_0(t) \exp[\beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_p X_{ip}]$$

Weibull Proportional Hazard Model

The Weibull model is a popular parametric proportional hazard model which assumes a specific functional form for the hazard rate, which can either increase or decrease over time. Its parameters are intuitively interpretable, with the shape parameter distinctly indicating whether the hazard rate is increasing, decreasing, or constant.

$$h_i(t) = \lambda\gamma t^{\gamma - 1} \exp(\beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_p X_{ip})$$

where λ is the scale parameter, and γ is the shape parameter.

3.4 Model Selection

Survival Tree

The survival tree method is a non-parametric approach used in survival analysis for model selection and identifying significant predictors of time-to-event outcomes. It involves segmenting the data into homogeneous subgroups based on the values of explanatory variables. The method constructs a tree structure where each node represents a subset of the dataset, and each split is based on the value of a predictor variable that best separates the data in terms of survival. Mathematically, this process involves recursively partitioning the data to maximize a criterion like the log-rank statistic. At each node, the split is chosen to maximize the difference in survival between the resulting subgroups. This can be represented as:

$$\text{Max}_{v,s}[\chi^2_{LR}(v,s)]$$

where $\chi^2_{LR}(v,s)$ is the log-rank test statistic computed for a split s on variable v. The process continues until a stopping criterion is met, typically based on the minimum number of observations in a node or a minimum improvement in the survival difference. The result is a tree where the paths from the root to the leaves represent rules for predicting survival, offering a visual and interpretable model of the factors affecting the time to an event like death in this study.

Stepwise Selection In model selection process, we incorporated bidirectional stepwise selection, alongside using various model assessment criteria such as Akaike Information Criterion (AIC), Corrected Akaike Information Criterion (AICc), and Schwarz Bayesian Criterion (SBC). This comprehensive approach enhanced our ability to identify the most appropriate model for our data.

We chose the final model based on AIC in our survival analysis due to its effective balance between model complexity and fit, particularly valuable in preventing overfitting in complex models. Additionally, given our sufficiently large sample size, AIC provided a more appropriate measure compared to AICc, and its less stringent penalty compared to SBC was better suited for our data's scale and complexity. The AIC for the survival model is as follows (Collett et al., 1999):

The formula for the Cox model being:

$$AIC_{cox} = -2\partial\mathcal{L}(\theta; x) + 2k$$

In this equation, k represents the number of parameters in the model. The term $2k$ serves as a penalty to discourage overfitting by complex models and $\mathcal{L}(\theta; x)$ indicates log-likelihood of parameter θ in the sample x.

For other survival models, the AIC adapts as follows with different penalty term:

$$AIC = -2\mathcal{L}(\theta; x) + 2(p + 2 + k)$$

where $k = 0$ for the exponential model, $k = 1$ for the Weibull, log-logistic and log-normal models, and $k = 2$ for the generalized gamma model.

3.5 Model Checking in the Proportional Hazard Model

Graphical Approach

The Cox Proportional Hazard model has two main assumptions. One is that the that the hazard functions of the survival curves of the different strata are proportional at time t. The other assumption is that the relationship between the $\log h(t)$, and each covariate is linear. We can compare the survival curves visually to check the PH assumption. One of the most commonly used plot is $\log(-\log \hat{S}(t|Z = z))$ over $\log t$. Since we know:

$$\log(-\log \hat{S}(t|Z=z)) - \log(-\log \hat{S}_0(t)) = \beta$$

where Z is a binary group indicator and the survival probability is usually estimated by K-M estimator. The parallel curves in the plot indicates the hazard ratio across the variable of interest is proportional at time t.

Another graphical method is to compare the differences between the fitted survival functions and observed K-M estimates from our PH model. If the fitted survival curse is closed to the observed K-M estimated survival curve, our PH assumption holds.

In the parametric proportional hazard models, we can use both $-\log \hat{S}(t)$ plot and $\log(-\log \hat{S}(t))$ to confirm if the hazard rate is constant. In other words, it can help us choose a proper distribution before fitting the models. For $-\log \hat{S}(t)$ plot, a straight line means constant hazard rate and the distribution should be exponential. Otherwise, Weibull model is better. In $\log(-\log \hat{S}(t))$ plot, if the slope of the straight line equals 1, then hazard rate should be constant. And if it is not, we choose Weibull model.

Schoenfeld residuals

The Schoenfeld residuals test evaluating the proportional hazards assumption in Cox regression models. It involves plotting the residuals against time; a lack of systematic trends in this plot generally indicates that the assumption holds. A smoothed curve can also be added to the plot to aid in interpretation. Conversely, if the residuals display a distinct trend over time, it suggests a violation of the proportional hazards assumption. Such a violation may occur if the effect of a covariate on the hazard rate changes over time, leading to non-constant hazard ratios. This could be attributed to time-varying covariates or time-dependent effects, which alter the relationship between the covariates and the survival outcome throughout the study period.

3.6 Model Validation

ROC Curves and AUC

In our analysis, we employed time-dependent Receiver Operating Characteristic (ROC) curves and Area Under the Curve (AUC) values to evaluate the discriminative ability of our Cox Proportional Hazards model over time. Specifically, we focused on two clinically relevant time points: 50 days and 250 days (Ahmad et al., 2017). The ROC curve is a graphical representation that illustrates the diagnostic ability of a binary classifier system by plotting the true positive rate (sensitivity) against the false positive rate (1 - specificity) at various threshold settings. AUC, a key summary measure of the ROC curve, quantifies the overall ability of the model to discriminate between individuals who will experience the event and those who will not, irrespective of the chosen probability threshold (Heagerty & Zheng, 2005). Generally, higher AUC values indicate better discriminative ability.

C-Index

The concordance index (C-index) was calculated to assess the predictive accuracy of our model. This metric is a measure of the model's ability to correctly rank the survival times of pairs of individuals, considering censored data (Steyerberg & Vergouwe, 2014). The C-index is calculated through pairwise comparisons, where a pair is concordant if the individual predicted to have a shorter survival time indeed experiences the event earlier than the other individual in the pair (Steyerberg & Vergouwe, 2014). A C-index of 0.5 suggests no better predictive accuracy than random chance, while a value of 1 indicates perfect prediction.

Calibration Slope

To evaluate the calibration of our model, we focused on the calibration slope. Calibration reflects the agreement between observed outcomes and predicted probabilities and the calibration slope assesses whether the predicted risks are of the correct magnitude (Steyerberg & Vergouwe, 2014). A slope of 1 indicates perfect calibration, meaning the model's predicted probabilities are accurately scaled. We calculated the calibration slope using logistic regression within a bootstrap framework, which allowed us to robustly assess the scale of the predicted risks relative to the actual event occurrences. The bootstrap approach, involving resampling the dataset 400 times, provided a more comprehensive understanding of the model's calibration under varying sample conditions.

4 Results

4.1 Nonparametric Methods

4.1.1 Life Table

Table 2.1: Heart Failure Life Table (Male)

	tstart	tstop	nsubs	nlost	nrisk	nevent	surv	pdf	hazard	se.surv	se.pdf	se.hazard
0-30	0	30	194	1	193.5	23	1.0000000	0.0039621	0.0042125	0.0000000	0.0007755	0.0008766
30-60	30	60	170	4	168.0	12	0.8811370	0.0020979	0.0024691	0.0232651	0.0005862	0.0007123
60-90	60	90	154	25	141.5	9	0.8181986	0.0017347	0.0021898	0.0278070	0.0005626	0.0007295
90-120	90	120	120	20	110.0	5	0.7661577	0.0011608	0.0015504	0.0309802	0.0005094	0.0006932
120-150	120	150	95	18	86.0	2	0.7313323	0.0005669	0.0007843	0.0332571	0.0003970	0.0005546
150-180	150	180	75	2	74.0	5	0.7143246	0.0016088	0.0023310	0.0345899	0.0006991	0.0010418
180-210	180	210	68	26	55.0	3	0.6660594	0.0012110	0.0018692	0.0384014	0.0006834	0.0010787
210-240	210	240	39	16	31.0	2	0.6297289	0.0013543	0.0022222	0.0416431	0.0009305	0.0015705
240-270	240	270	21	16	13.0	1	0.5891013	0.0015105	0.0026667	0.0478504	0.0014564	0.0026645
270-300	270	300	4	4	2.0	0	0.5437858	0.0000000	0.0000000	0.0620201	NaN	NaN
300-Inf	300	Inf	0	0	0.0	0	0.5437858	NA	NA	0.0620201	NA	NA

Table 2.2: Heart Failure Life Table (Female)

	tstart	tstop	nsubs	nlost	nrisk	nevent	surv	pdf	hazard	se.surv	se.pdf	se.hazard
0-30	0	30	105	3	103.5	8	1.0000000	0.0025765	0.0026801	0.0000000	0.0008750	0.0009468
30-60	30	60	94	0	94.0	9	0.9227053	0.0029448	0.0033520	0.0262504	0.0009372	0.0011159
60-90	60	90	85	10	80.0	6	0.8343612	0.0020859	0.0025974	0.0367098	0.0008241	0.0010596
90-120	90	120	69	15	61.5	4	0.7717841	0.0016732	0.0022409	0.0419136	0.0008140	0.0011198
120-150	120	150	50	5	47.5	2	0.7215868	0.0010128	0.0014337	0.0460937	0.0007039	0.0010135
150-180	150	180	43	3	41.5	2	0.6912042	0.0011104	0.0016461	0.0489040	0.0007700	0.0011636
180-210	180	210	38	12	32.0	3	0.6578931	0.0020559	0.0032787	0.0519106	0.0011416	0.0018907
210-240	210	240	23	10	18.0	0	0.5962156	0.0000000	0.0000000	0.0579853	NaN	NaN
240-270	240	270	13	11	7.5	0	0.5962156	0.0000000	0.0000000	0.0579853	NaN	NaN
270-300	270	300	2	2	1.0	0	0.5962156	0.0000000	0.0000000	0.0579853	NaN	NaN
300-Inf	300	Inf	0	0	0.0	0	0.5962156	NA	NA	0.0579853	NA	NA

Table 2.1 and **Table 2.2** represent the lifetable with a time break of 30 days (one month), stratified by gender. According to the table, we find that the last line (270-300 days) shows a survival probability larger than 0.5 for both genders (0.54 for male and 0.59 for female). This may indicate a high life expectancy and improved health care, where a significant proportion of individuals are expected to live longer than 300 days (10 months). Moreover, males have a relatively shorter survival time than females based on the life table. This hypothesis needs future testing in the following modeling fit.

4.1.2 The Kaplan-Meier and Fleming-Harrington Model

Figure 2.1. Overall Kaplan−Meier Sur

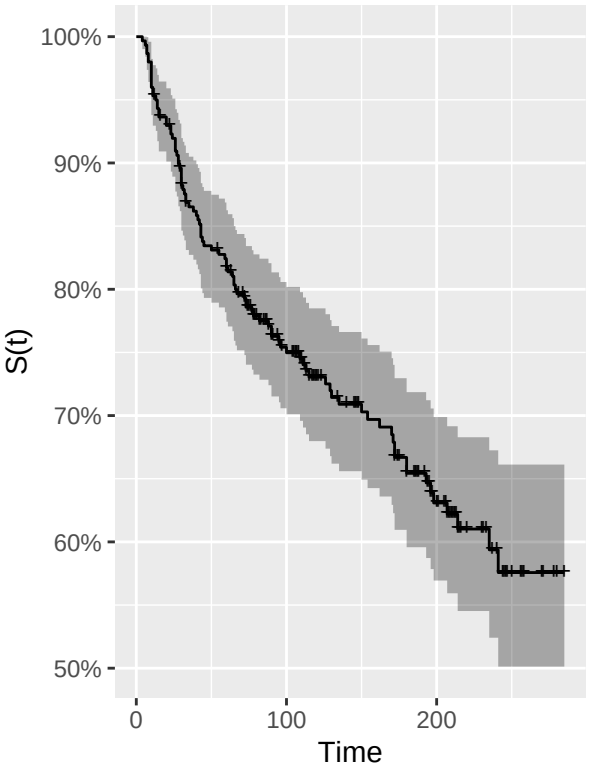


Figure 2.2. Kaplan−Meier Survival Cur

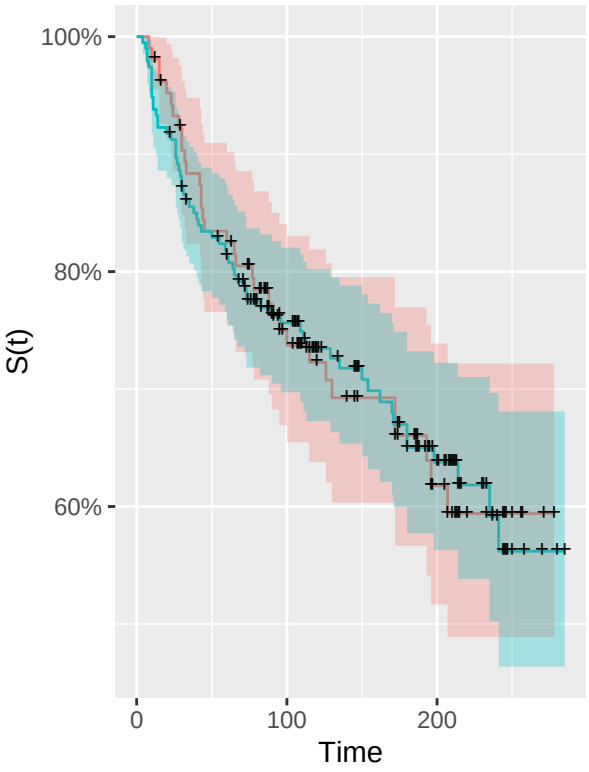


Figure 2.3. Overall Fleming−Harrington Survival Curve

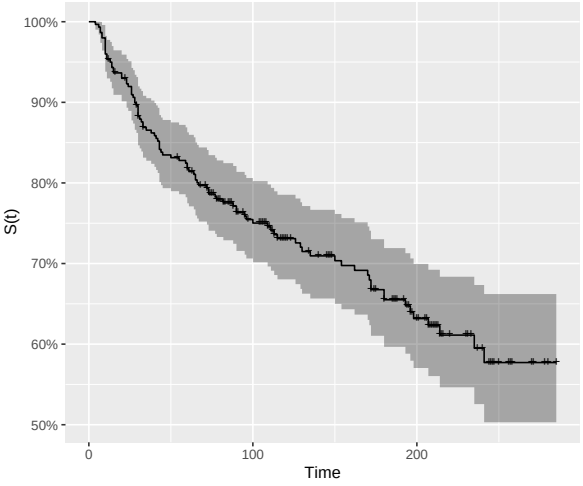
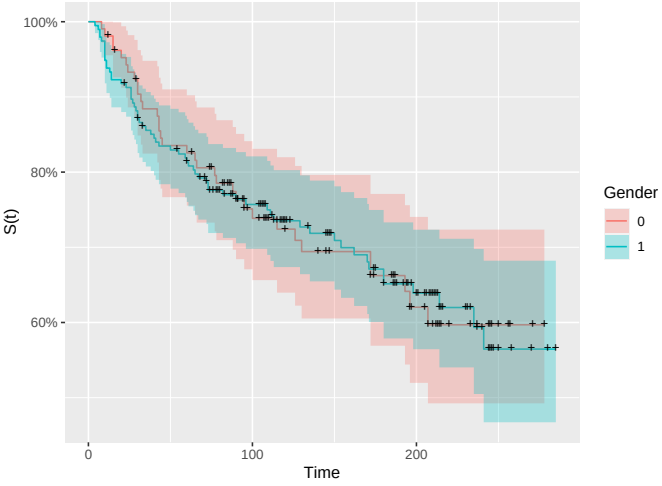


Figure 2.4. Fleming−Harrington Survival Curve By Gender



From the graph we noticed that the median survival time is larger than 300 months. In other words, more than half of the individuals in this study have not experienced the event in the study period. Moreover, the Kaplan-Meier and Fleming-Harrington have similar trends and show no significant difference between genders. Given that the p-value is relatively high (p-value = 0.95) for both Kaplan-Meier and Fleming-Harrington estimators, we can further make sure that no significant difference appears in the survival experience between males and females.

4.2 Hypothesis Testing

4.2.1 Log-Rank test and Gehan's Wilcoxon test

Figure 3.1: Comparison of Survival Experience Between Males and Females by Log-Rank Test

Gender	N	Observed	Expected	$\frac{(O-E)^2}{E}$	$\frac{(O-E)^2}{V}$
0	105	34	34.3	0.00254	0.00397
1	194	62	61.7	0.00141	0.00397

Figure 3.2: P-values for the Log-rank and Gehan's Wilcoxon test

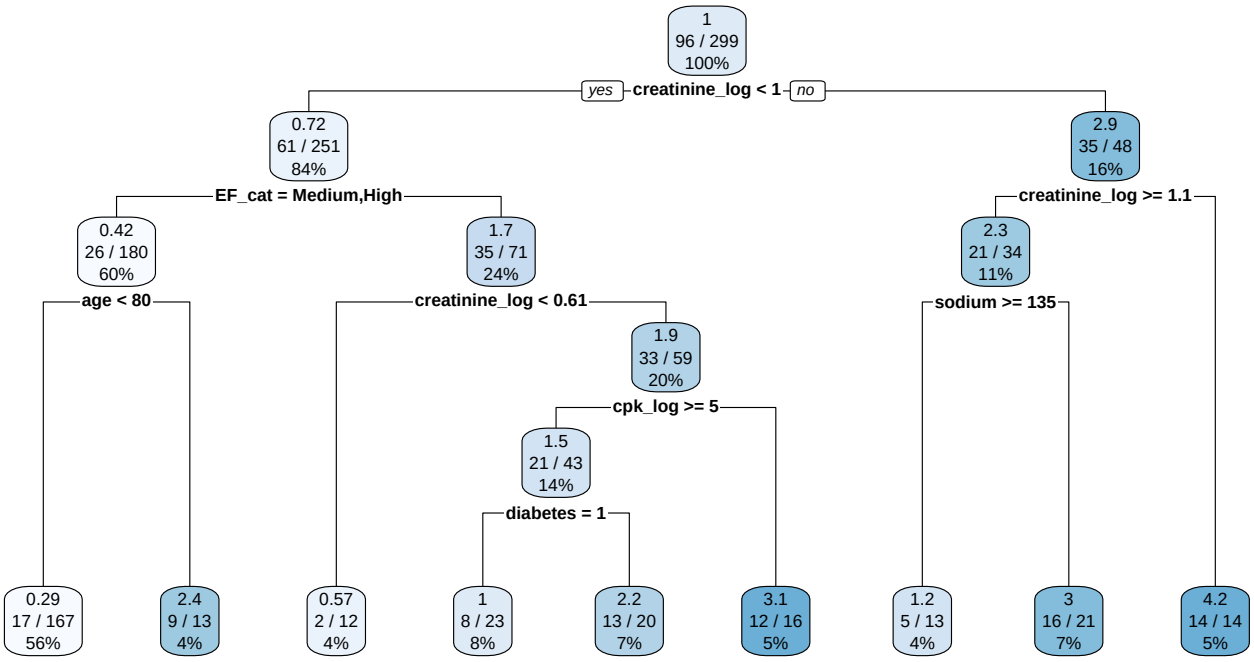
Test	Chi square	df	p-value
Log Rank Test	0	1	0.950
Wilcoxon Test	0	1	0.765

The Log-Rank test and the Gehan's Wilcoxon test can be used to test for differences in survival experience between genders. Reviewing the results from the table, we find that both the Log-Rank test and the Gehan's Wilcoxon test have provided a similar result and gave a p-value greater than 0.05. This indicates that we failed to reject the null hypothesis at the significance level of 0.05, and we can state there is no statistically significant difference in survival experiences between males and females.

4.3 Model Selection

4.3.1 Survival Tree

Figure 3: Results of Survival Tree



The survival tree depicted in the image stratifies patients based on factors affecting survival. At the first level, patients are divided by `EF_cat`, suggesting that EF is a significant factor in survival. For patients with medium to high EF, age is the next discriminator, indicating its importance in survival for this subgroup. Among those with lower EF, log-transformed creatinine levels further split the cohort, underscoring renal function's role in survival. The tree branches into additional factors such as log-transformed cpk levels, the presence of diabetes, and sodium levels, highlighting their contribution to survival outcomes. Terminal nodes provide risk ratios and the proportion of patients experiencing the event, illustrating the combined effect of these variables on patient survival. This tree model, therefore, offers a nuanced view of patient risk profiles, emphasizing the multifactorial nature of survival in this patient population.

4.3.2 Stepwise Selection

Table 3: Summary Table of Model Selection

Step	EnteredEffect	SL	AIC	AICc	SBC
1	logcre	0	993.05	991.09	995.61
2	age	0	975.5	971.62	980.62
3	EF_catMedium	2e-04	964.02	958.28	971.71
4	EF_catHigh	5e-04	953.71	946.15	963.97
5	bp	0.0185	950.16	940.83	962.98
6	sodium	0.1228	949.78	938.72	-
7	anemia	0.1922	-	937.35	-
8	logcpk	-	-	936.72	-
9	diabetes	-	-	936.33	-

The **Table 3** summarizes the Cox model selection process, indicating `logcre`, `age`, `EF` categories, and `BP` as persistent predictors across the top four models. The AIC favors `logcre` and `age` for their significant contributions to model fit relative to added complexity, emphasizing model fit over parameter count.

However, with large sample size, the AICc modification slightly adjusts these values, shifting the preference toward other variables like `diabetes`, which shows the lowest AICc, suggesting a strong impact on the model when accounting for sample size. The SBC, with its more substantial penalty for complexity, particularly at larger sample sizes, selects `BP` as the best variable, underscoring its contribution to the model's explanatory power without excessive complexity.

We choose AIC for model selection, given its consistency with SL findings, which jointly highlight `logcre` and `age` as key predictors. This consistent identification underscores their substantial impact on model accuracy while maintaining simplicity, justifying their selection.

4.4 Semi-parametric Model

The Cox model obtained through the Stepwise Selection using Akaike Information Criterion (AIC) contains creatinine, age, ejection fraction, blood pressure status, sodium covariates, the model as follows:

$$h(t) = h_0(t) \exp[\beta_1 Age + \beta_2 EF_{Medium} + \beta_3 EF_{High} + \beta_4 BP + \beta_5 Sodium + \beta_6 \log(Creatinine + 1)]$$

4.4.1 Model Checking

Before fitting the Cox Proportional Hazard model, we need to check whether our variables hold the PH assumption[6]. The $\log(-\log \hat{S}(t|Z = z))$ over $\log t$ plot and observed survival versus estimated survival curves are provided below. We have multiple variables remained after model selection, so we randomly took two of them (`EF_cat` and `logcre`) for the PH assumption check. Since `logcre`[7] is continuous variable, we categorized it into two groups and recoded as "Low" and "High", respectively.

Figure 4.1: Log of Negative Log of Estimated Survival Functions by EF Group

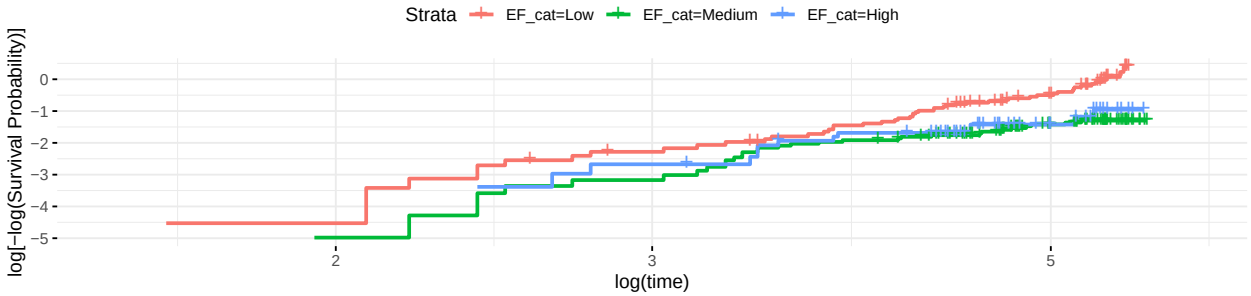
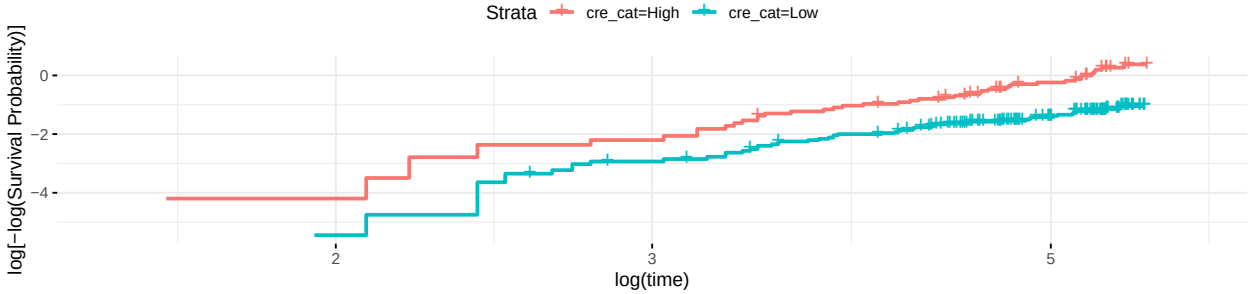
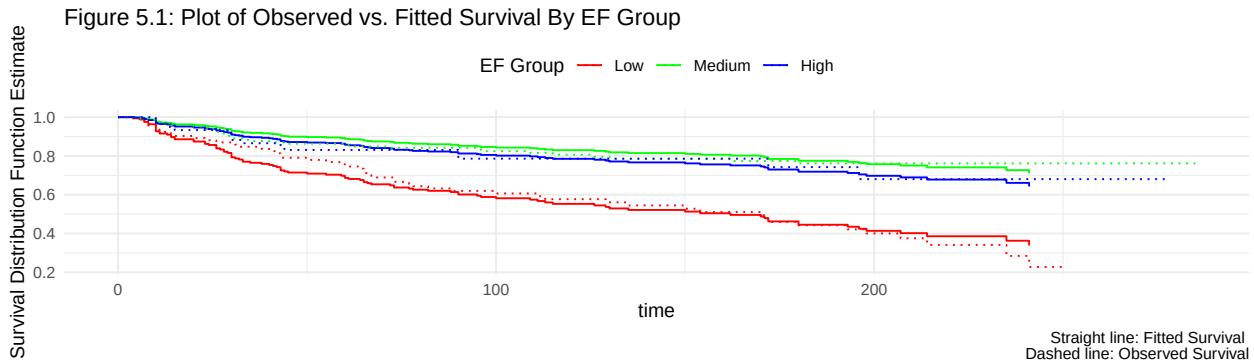


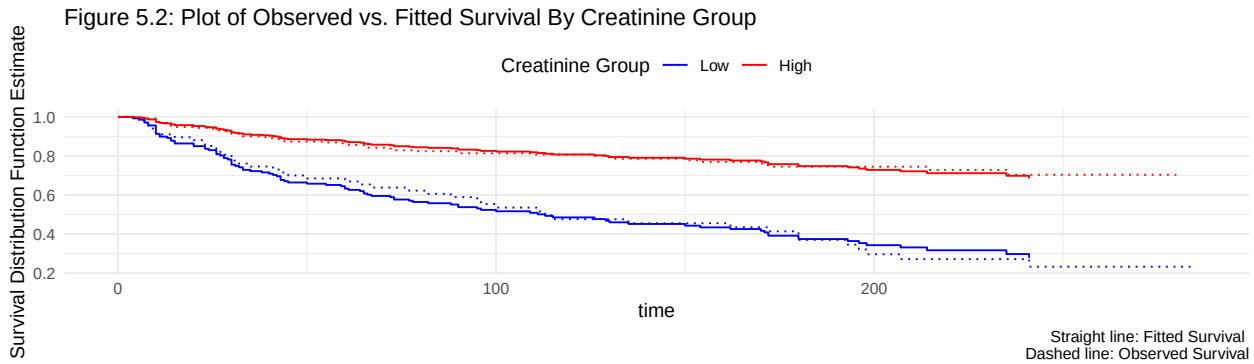
Figure 4.2: Log of Negative Log of Estimated Survival Functions by Log−transformed Creatinine Group



[6]See the Method Section for details

[7]Since 1.5 mg/dL is the clinical threshold of creatinine, so we categorized $\log(\text{creatinine} + 1) > \log(2.5)$ as "High", otherwise, it's in "Low" group. And people in "High" group should be regarded as renal dysfunction.





Global Schoenfeld Test p: 0.0895

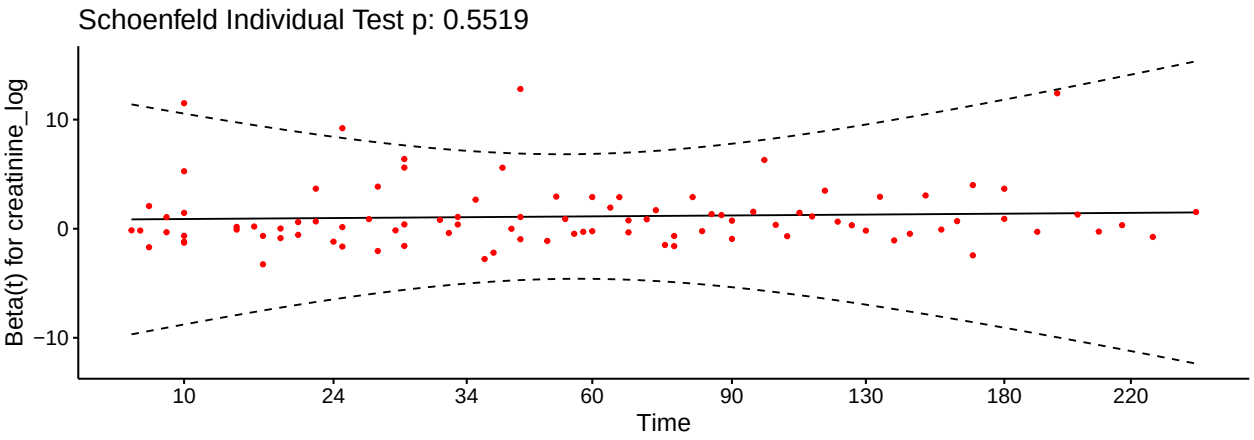


Figure 6.1: Schoenfeld Test For Log-transformed Creatinine

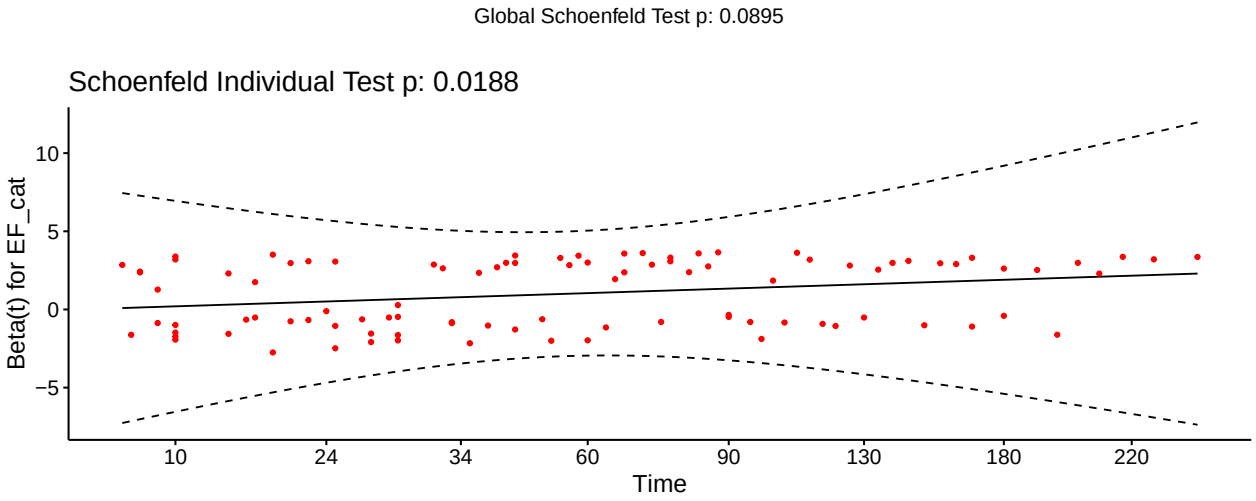


Figure 6.2: Schoenfeld Test For EF_cat

Obviously, the curves in **Figure 4.2** are parallel with each other, indicating the proportional hazard assumption hold. For the observed survival versus estimated survival plot **Figure 5.2**, the straight curve and dashed curve are close to each other, which means the proportional hazard ratio is proportional for creatinine groups. These findings are also consistent with what we observed in the Schoenfeld residual plots **Figure 6.1** of the fitted model. We chose to input `logcre` variables and found a regression lines with a slope close to zero.

However, after checking the `EF_cat` variable, the curves in **Figure 4.1** are not parallel, the straight curve and dashed curve in **Figure 5.1** are not close to each other, especically at the beginning of the study. And there is an obvious slope for the regression line in Globall Schoenfeld test in **Figure 6.2**. The PH assumption is violated and we decide to remove this variable from our models.

4.4.2 Cox Model Fitting and Results

Table 4: Summary of Cox Proportional Hazard Model

	Variable	β	$exp(\beta)$	$SE(\beta)$	Z test statistics	P-value
1	logcre	1.1175	3.0573	0.2931	3.8124	0.0001
2	age	0.0462	1.0473	0.0091	5.0891	0.0000
3	EF_catMedium	-1.0758	0.3410	0.2353	-4.5718	0.0000
4	EF_catHigh	-0.9783	0.3759	0.3124	-3.1320	0.0017
5	bp	0.5457	1.7258	0.2138	2.5517	0.0107
6	sodium	-0.0379	0.9628	0.0241	-1.5718	0.1160

Table 4 indicates that the first five variables are statistically significant at the $\alpha = 0.05$. Among these, `logcre`, `age`, and `bp` exhibit positive hazard ratios, suggesting an increased risk associated with higher values of these variables. On the other hand, `EF_cat` demonstrates a negative hazard ratio, indicating patient with high or medium ejection fractions value have a lower risk compared to those with low ejection fraction values.

4.5 Parametric Models

4.5.1 Parametric Model Checking

Figure 7.1: Negative Log of Estimated Survival Functions For Age Groups

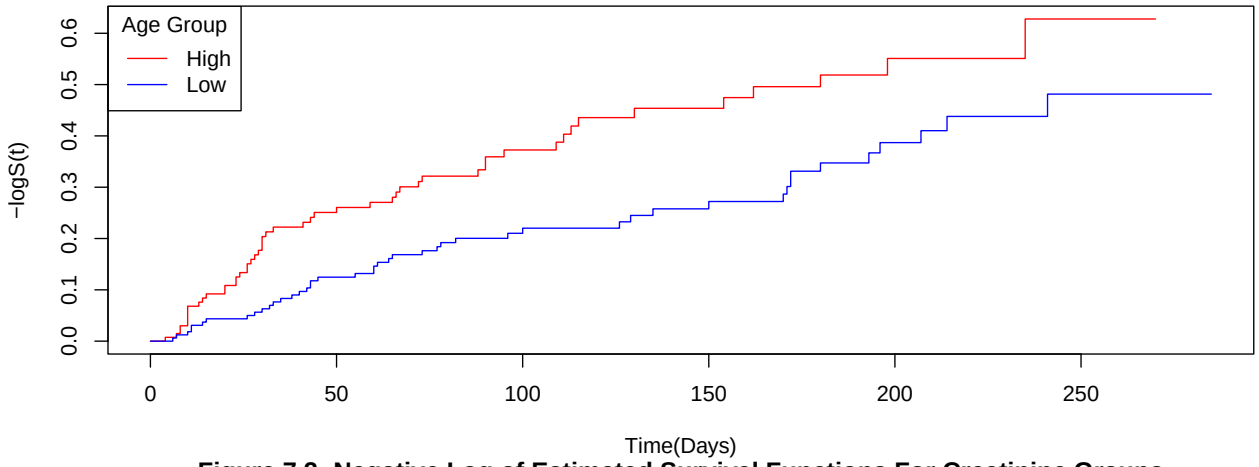
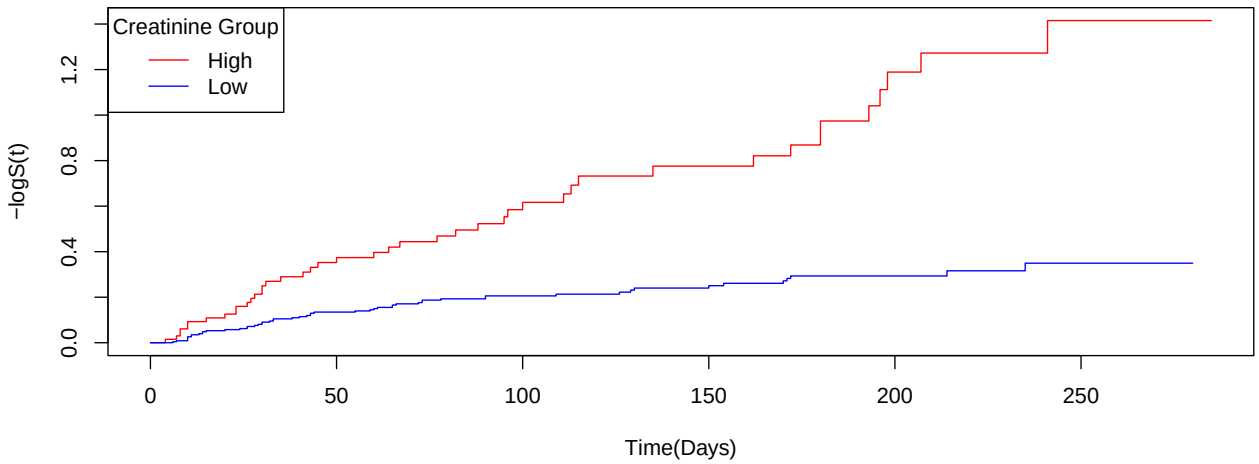


Figure 7.2: Negative Log of Estimated Survival Functions For Creatinine Groups



Since we have already created the $\log(-\log \hat{S}(t)$ (**Figure 4.1** and **Figure 4.2**) for `age` and `logcre`, and the slopes do not strictly equal to 1. Now we also check the $-\log \hat{S}(t)$ plot (**Figure 7.1** and **Figure 7.2**) for same variables. The curves for both two plots seems not to be straight and nonlinear. We suggest Weibull distributions for parametric proportional hazard models.

4.5.2 Weibull Model Fitting and Results

Table 5: Summary of Weibull PH Model Fitting

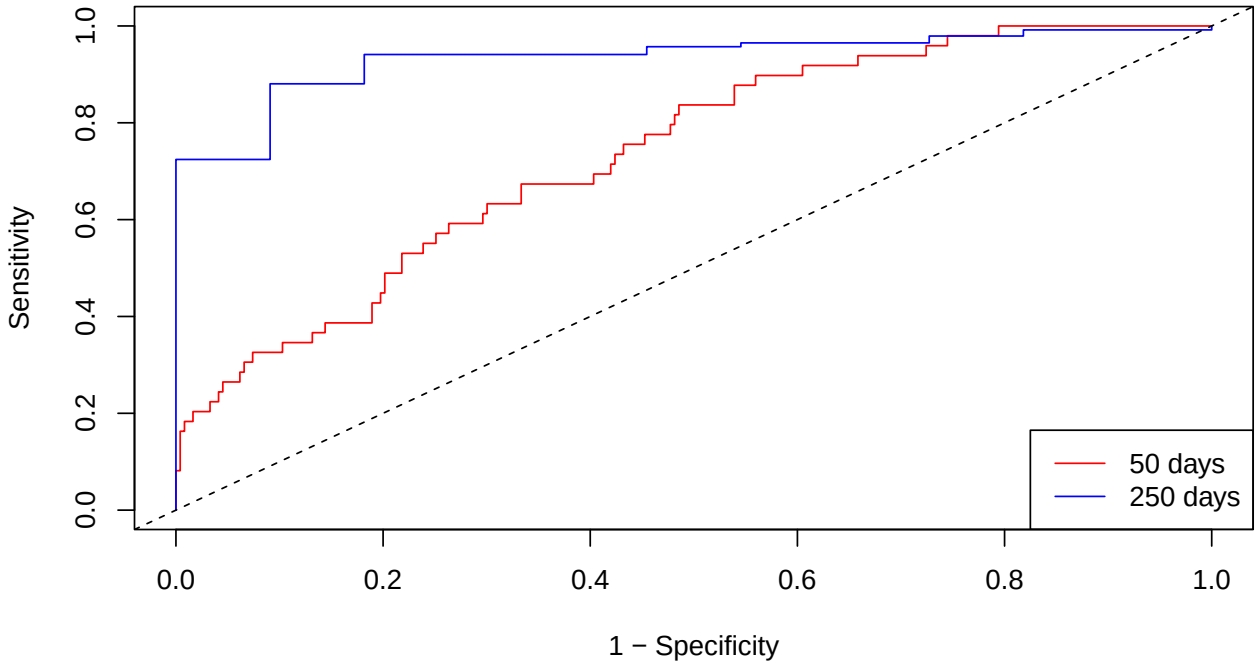
Variable	β	$exp(\beta)$	$SE(\beta)$	Wald P-value
bp	0.4907	1.6335	0.2110	0.0200
age	0.0409	1.0418	0.0089	0.0000
sodium	-0.0519	0.9494	0.0211	0.0140
logcre	1.2681	3.5539	0.2579	0.0000
log(scale)	2.5470	12.7691	3.2353	0.4311
log(shape)	-0.0875	0.9162	0.0891	0.3262

The Weibull proportional hazard model demonstrates that in the, the first four variables are statistically significant at $\alpha = 0.05$, which is the same as the significance variables in the Cox proportional hazard model. Additionally, the log scale value of 2.547 implies that the baseline hazard function is scaled by exp(2.547), influencing the time to event and indicating a higher baseline hazard rate. Meanwhile, the log shape value of -0.0875, when exponentiated to approximately 0.9162, suggests a decreasing hazard rate over time, meaning that the risk of the event occurring diminishes as time progresses.

4.6 Model Validation

4.6.1 ROC Curves and AUC

Figure 8: Time−Dependent ROC Curves for 50 and 250 Days



Using the Cox Proportional model with the 5 selected predictors, our time-dependent ROC analysis at 50 days yielded an AUC of approximately 0.74, while at 250 days, the AUC was 0.93. These values indicate that the model's ability to discriminate between those who will experience the event and those who will not

improves over time. The ROC curves further visually demonstrate this improvement, with the curve for 250 days being closer to the top left corner, indicating better performance.

4.6.2 C-Index

The average C-index calculated through bootstrapping (n = 400) was 0.737 This suggests that in about 74% of pairwise comparisons, our model correctly ranks the survival times. A C-index of around 0.74 is generally indicative of good predictive ability, especially in clinical settings where accurate risk stratification is crucial for treatment planning.

4.6.3 Calibration Slope

Our calculated mean calibration slope came to approximately 1.33. This value, slightly above the ideal of 1, is significant in understanding the model's performance. The calibration slope measures the extent to which the model's predicted risks are proportionate to the observed risks. A value of 1 would indicate perfect calibration, meaning the model's predictions are perfectly aligned with the actual observed risks. Our finding of a calibration slope above 1 suggests that our model may be mildly overfitting the data, predicting slightly higher risks than what is observed.

5 Discussion

According to the life table, we find that there is a slight difference in survival probability between male and female. However, the hypothesis test and model selection show that there is actually no difference. Similarities in heart failure presentation, treatment regimen, and sample size may explain the absence of sex-based differences in survival. Specifically, if the severity of heart failure was not related to gender or if patients received appropriate gender-based treatment and the sample size or characteristics of the study limited gender-specific analyses, potential differences could be masked.

Both the **Cox PH model** and the **Weibull PH Model** reveals significant predictors of survival outcomes in heart failure patients. Key among these is `logcre`, with the Cox model showing a HR of 3.06, indicating a substantial risk increase with elevated levels. `Age` with the Cox model suggesting a 5 increase in risk per year, implies people with advancing age face higher hazard. The `EF_cat` shows importance of cardiac function, as higher ejection fractions (`EF_catMedium` and `EF_catHigh`) correlate with reduced risk. `bp` similarly shows a significant impact on hazard, with a 72% increase in risk for higher levels in the Cox model, a finding mirrored in the Weibull analysis.

The Weibull model, in addition to confirming these findings, brings additional insights due to its parametric nature. The model emphasis on `bp`, `age`, and `EF_cat` with notably small p-values highlights their robustness as predictors. For `sodium`, the relationship with survival outcomes is more complex. In the Cox model, it is approach to significance level $\alpha = 0.1$, but remains statistically non-significant. The consistency between the Cox and Weibull models in identifying the key predictors provides credibility to these findings.

Supporting this, A study published in Frontiers in Cardiovascular Medicine found that postoperative serum creatinine is a significant prognostic factor for cardiac surgery patients (Zhong et al., 2021). This study showed that higher levels of postoperative serum creatinine were linked to increased hospital mortality and longer stays in the intensive care unit. Specifically, it found that patients who did not survive had significantly higher postoperative serum creatinine levels, and there was a positive correlation between these levels and lengths of ICU stay.

The nuanced findings from our model validation analysis provide a comprehensive view of our Cox model's performance. While the model exhibits strong discriminative ability, as indicated by the AUC values and C-index, our calibration assessment, particularly the calibration slope, suggests areas where improvement is needed. Notably, the calibration slope, slightly over the ideal value of one, implies a mild overestimation in

risk predictions. This indicates a complex calibration scenario where the model might be overfitting to some extent.

Such overestimation, although modest, is critical in clinical settings. Accurate risk prediction is vital for informed decision-making and effective patient management. Overestimated risks might lead to more aggressive interventions than necessary, affecting patient care and resource allocation. Conversely, underestimating risks could result in missed opportunities for timely intervention. This highlights the importance of achieving a balance in predictive accuracy, ensuring that the model neither overestimates nor underestimates risks.

The observed improvement in the model's discriminative ability over time, with increasing AUC values from 50 to 250 days, underscores the dynamic nature of risk factors and their evolving impact on patient outcomes. However, the calibration results emphasize the need to focus not just on the model's ability to discriminate but also on the accuracy of its probability predictions.

Future work should, therefore, focus on refining the model's complexity and variable selection. Re-evaluating the model's components and considering alternative modeling approaches might help in aligning the predicted probabilities more closely with actual outcomes. Applying more advanced calibration techniques could also address the observed overfitting, enhancing the model's reliability. Additionally, external validation on an independent dataset is essential to confirm the model's effectiveness and applicability in different clinical contexts. Such efforts will be crucial in enhancing the model's utility and ensuring its robustness in real-world clinical applications, where precise risk assessment directly informs patient care strategies.

6 Conclusions

The non-parametric method showed that more than 50% of both genders (54% for males and 59% for females) can survive for over 300 days. This may indicate a high life expectancy and improved health care. Moreover, the non-parametric method showed no significant difference in survival probability between genders. This result was later consistent with the results of the model selection. In conclusion, incorporating the insights from both semi-parametric and parametric models, particularly the Cox Proportional model, our study offers a holistic view of heart failure prognosis. The findings not only reinforce the critical role of established risk factors but also highlight the evolving nature of these risks over time. While the Cox model demonstrates strong discriminative ability, it also suggests the need for refinement in risk predictions and addressing slight over-estimation. Future work will focus on refining the model's complexity, exploring alternative approaches, and conducting external validation to enhance its accuracy and applicability in diverse clinical settings. This study, by identifying significant predictors such as creatinine levels, age, ejection fraction, and blood pressure, contributes to a deeper understanding of heart failure progression and lays the groundwork for more personalized and effective patient management strategies.

7 References

1. Ahmad, T., Munir, A., Bhatti, S. H., Aftab, M., & Raza, M. A. (2017). Survival analysis of heart failure patients: A case study. PLOS ONE, 12(7), e0181001. https://doi.org/10.1371/journal.pone.0181001

2. Collett, D. (1999). Modelling survival data in medical research. Chapman & Hall/CRC.

3. Heagerty, P. J., & Zheng, Y. (2005). Survival Model Predictive Accuracy and ROC Curves. Biometrics, 61(1), 92–105. https://doi.org/10.1111/j.0006-341X.2005.030814.x

4. Jones, N., Ak, R., Adoki, I., Fdr, H., & Cj, T. (2019). Survival of patients with chronic heart failure in the community: a systematic review and meta-analysis. European Journal of Heart Failure, 21(11), 1306–1325. https://doi.org/10.1002/ejhf.1594

5. Pavlou, M., Ambler, G., Seaman, S. R., Guttmann, O., Elliott, P., King, M., & Omar, R. Z. (2015). How to develop a more accurate risk prediction model when there are few events. BMJ, h3868. https://doi.org/10.1136/bmj.h3868

6. Steyerberg, E. W., & Vergouwe, Y. (2014). Towards better clinical prediction models: Seven steps for development and an ABCD for validation. European Heart Journal, 35(29), 1925–1931. https://doi.org/10.1093/eurheartj/ehu207

7. Zhong, J., Gao, J., Luo, J. C., Zheng, J. L., Tu, G. W., & Xue, Y. (2021). Serum creatinine as a predictor of mortality in patients readmitted to the intensive care unit after cardiac surgery: a retrospective cohort study in China. Journal of thoracic disease, 13(3), 1728–1736. https://doi.org/10.21037/jtd-20-3205

8 Appendix

8.1 Code

```
knitr::opts_chunk$set(echo = FALSE, message = FALSE, warning = FALSE)
library(biostat3)
library(tidyverse)
library(knitr)
library(kableExtra)
library(survival)
library(survminer)
library(ggfortify)
library(ggsurvfit)
library(patchwork)
library(writexl)
library(readxl)
library(table1)
library(rmarkdown)
library(KMsurv)
library(StepReg)
library(ggplot2)
library(timeROC)
library(boot)
library(rms)
library(survivalROC)
library(rpart)
library(eha)
library(pec)
data = read_csv("./data/heart_failure.csv")
dat <- data |>
  arrange(TIME) |> janitor::clean_names() |>
  mutate(ejection_fraction_cat = case_when(ejection_fraction <= 30 ~ "Low",
                                           ejection_fraction > 30
                                           & ejection_fraction <= 45 ~ "Medium",
                                           ejection_fraction > 45 ~ "High")) |>
  mutate(gender = factor(gender),
         smoking = factor(smoking),
         diabetes = factor(diabetes),
         bp = factor(bp),
         event = factor(event),
         anaemia = factor(anaemia),
         ejection_fraction_cat = factor(ejection_fraction_cat,
                                        levels = c("Low", "Medium", "High"))) |>
  rename(platelets = pletelets,
         anemia = anaemia,
         EF = ejection_fraction,
         EF_cat = ejection_fraction_cat)

# Calculate the number of right-censored:
number_censored <- sum(dat$event == 0)
# Calculate the number of event:
number_event <- sum(dat$event == 1)
```

```r
dat_table = dat
label(dat_table$time) = "Survival time (days)"
label(dat_table$gender) = "Gender"
label(dat_table$smoking) = "Smoking status"
label(dat_table$diabetes) = "Diabetes"
label(dat_table$bp) = "Blood Pressure"
label(dat_table$anemia) = "Anemia"
label(dat_table$age) = "Age (years)"
label(dat_table$EF_cat) = "Ejection Fraction (EF_cat)"
label(dat_table$sodium) = "Serum Sodium (mEq/L)"
label(dat_table$creatinine) = "Serum creatinine (mg/dL)"
label(dat_table$platelets) = "Plateletes (mcL)"
label(dat_table$cpk) = "Creatinine phosphokinase (U/L)"
dat_table$event <- factor(dat$event, levels = c(0, 1),
                          labels = c("Censored", "Event"))

pvalue <- function(x, ...) {
  # Remove the "overall" column
    x <- x[names(x) != "overall"]
    # Construct vectors of data y, and groups (strata) g
    y <- unlist(x)
    g <- factor(rep(1:length(x), times = sapply(x, length)))
    if (is.numeric(y)) {
        # For numeric variables, perform a standard 2-sample t-test
        p <- t.test(y ~ g)$p.value
    } else {
        # For categorical variables, perform a chi-squared test of independence
        p <- chisq.test(table(y, g))$p.value
    }
    # Format the p-value, using an HTML entity for the less-than sign.
    # The initial empty string places the output on the line below the variable label.
    c("", sub("<", "<", format.pval(p, digits = 3, eps = 0.001)))
}
caption = "Table 1: Descriptive Statistics Table for Variable Characteristic"

table1 = table1(~ time + age + gender + smoking + diabetes + bp + EF_cat +
                  anemia + sodium + creatinine + cpk + platelets | event,
                  data = dat_table,
                extra.col = list(`P-value` = pvalue), caption = caption)
t1kable(table1) |> kable_styling(font_size = 8, latex_options = "HOLD_position")
# Data contains the continuous vars only
cont_dat = dat |>
  dplyr::select(age, sodium, creatinine, platelets, cpk)
# Long format
cont_dat.long = cont_dat |>
  pivot_longer(cols = c(age, sodium, creatinine, platelets, cpk))
# Plot the continuous variable histograms
cont_hist = ggplot(data = cont_dat.long, aes(x = value)) +
  geom_histogram(aes(fill = name), bins = 30) +
  facet_wrap(~name, scales = "free") +
  labs(x = "Value", y = "Count",
       title = "Figure 1.1: Histograms of Continuous Variables") +
  theme_bw() +
```

```
  theme(legend.position = "none")
cont_hist

# Data contains the categorical vars only
cate_data = dat |>
  dplyr::select(event, gender, smoking, diabetes, bp, anemia, EF_cat)
# Long format
cate_dat.long = cate_data |>
  pivot_longer(cols = c(event, gender, smoking, diabetes, bp, anemia, EF_cat))
# Plot the categorical variable barplots
cate_barplot = ggplot(cate_dat.long, aes(x = value, fill = value)) +
  geom_bar() +
  geom_text(stat = 'count', aes(label = ..count..), vjust = -0.3) +
  facet_wrap(~name, scales = "free") +
  labs(x = "Category", y = "Count", fill = "Category",
       title = "Figure 1.2: Bar Charts of Categorical Variables") +
  theme_bw() +
  theme(legend.position = "none") +
  ylim(0, 240)
cate_barplot
dat_log = dat |>
  mutate(cpk_log = log(cpk + 1),
         creatinine_log = log(creatinine + 1))
nonpara_dat = dat_log |>
  dplyr::select(-c(EF, cpk, creatinine)) |>
  relocate(time, event, EF_cat, smoking, everything()) |>
  mutate(event = as.numeric(event) - 1)
nonpara_male = nonpara_dat |> filter(gender == 1)
nonpara_female = nonpara_dat |> filter(gender == 0)
life_table_male <- lifetab2(Surv(time, event) ~ 1, data = nonpara_male,
                            breaks = seq(0, 300, 30))
life_table_female <- lifetab2(Surv(time, event) ~ 1, data = nonpara_female,
                              breaks = seq(0, 300, 30))
life_table_male |> kable(booktabs = T,
                         caption = "Table 2.1: Heart Failure Life Table (Male)") |>
  kable_styling(latex_options = c("HOLD_position"), font_size = 6)
life_table_female |> kable(booktabs = T,
                           caption = "Table 2.2: Heart Failure Life Table (Female)") |>
  kable_styling(latex_options = c("HOLD_position"), font_size = 6)
km_overall = survfit(Surv(time, event) ~ 1, data = nonpara_dat)
km_overall_plot =
  km_overall |> autoplot() +
  labs(y = "S(t)",
       x = "Time",
       subtitle = "Figure 2.1. Overall Kaplan-Meier Survival Curve") +
  theme(legend.position = "none")

km = survfit(Surv(time, event) ~ gender, data = nonpara_dat)
km_plot =
  km |> autoplot() +
  labs(y = "S(t)",
       x = "Time",
       subtitle = "Figure 2.2. Kaplan-Meier Survival Curve By Gender",
```

```
       color = "Gender", fill = 'Gender') + theme(legend.position = "none")
km_overall_plot + km_plot
fh_overall = survfit(Surv(time, event) ~ 1, data = nonpara_dat, type = "fh")
fh_overall_plot =
  fh_overall |> autoplot() +
  labs(y = "S(t)",
       x = "Time",
       subtitle = "Figure 2.3. Overall Fleming-Harrington Survival Curve") +
  theme(legend.position = "none")
fh <- survfit(Surv(time, event) ~ gender, data = nonpara_dat, type = "fh")
fh_plot =
  fh |> autoplot() +
  labs(y = "S(t)",
       x = "Time",
       subtitle = "Figure 2.4. Fleming-Harrington Survival Curve By Gender",
       color = "Gender", fill = 'Gender')
fh_overall_plot + fh_plot
logrank_test <- survdiff(Surv(time, event) ~ gender, data = nonpara_dat)
wilcox_result <- wilcox.test(time ~ gender, data = nonpara_dat)
results1 <- tibble(
  Gender = c(0, 1),
  N = c(105, 194),
  Observed = c(34,62),
  Expected = c(34.3, 61.7),
  '11' = c(0.00254, 0.00141),
  '22' = c(0.00397, 0.00397)
)

results2 <- tibble(
  Test = c("Log Rank Test", "Wilcoxon Test"),
  "Chi square" = c(0, 0),
  df = c(1, 1),
  "p-value" = c(0.950, 0.765)
)

results_table1 <- kable(results1, align = "c", booktabs = T, escape = F,
caption = "Figure 3.1: Comparison of Survival Experience Between Males and Females by Log-Rank Test",
col.names = c("Gender", "N", "Observed", "Expected", "$\\frac{(O-E)^2}{E}$", "$\\frac{(O-E)^2}{V}$")) |>
  kable_styling(latex_options = c("HOLD_position"))

results_table2 <- kable(results2, align = "c", booktabs = T,
caption = "Figure 3.2: P-values for the Log-rank and Gehan's Wilcoxon test") |>
  kable_styling(latex_options = c("HOLD_position"))

results_table1
results_table2
surv_obj <- Surv(time = nonpara_dat$time, event = nonpara_dat$event)
surv_tree <- rpart(surv_obj ~ gender + smoking + diabetes + bp + anemia + age + EF_cat + sodium + creati
                   data = nonpara_dat, method = "exp")
rpart.plot::rpart.plot(surv_tree, main = "Figure 3: Results of Survival Tree")
# raw data
raw_data <- data |>
  arrange(TIME) |>
```

```r
  janitor::clean_names() |>
  mutate(platelets = pletelets,
         anemia = anaemia)

# model data
model_data <- dat |>
  mutate(event = as.numeric(event))

## create dummy variable for categorical variable
heart_data <- model.matrix(EF~ EF_cat, data = model_data)[,-1] |>
  as.data.frame()

## create data frame for stepwiseCox
heart_data <- cbind(raw_data, heart_data)
stepwise_data <- heart_data |>
  mutate(logcre = log(creatinine + 1),
         logcpk = log(cpk + 1)) |>
  dplyr::select(-creatinine, -cpk, -ejection_fraction)
# Variable selection using stepwise Cox model using Sl
stepwise_model1 <- stepwiseCox(Surv(time, event) ~ gender + smoking + diabetes + bp +
                                 anemia + age + sodium + platelets + logcre + logcpk +
                                 EF_catMedium + EF_catHigh,
                               data = stepwise_data,
                               select = "SL",
                               # significant level for entry
                               sle = 0.25,
                               # significant level for stay
                               sls = 0.15,
                               method = "efron",
                               weights = NULL,
                               best = NULL)
# 7 variables are selected: logcre, age, EF_catMedium, EF_catHigh, bp, sodium, anemia

## Variable selection using stepwise Cox model using AIC
stepwise_model2 <- stepwiseCox(Surv(time, event) ~ gender + smoking + diabetes + bp +
                                 anemia + age + sodium + platelets + logcre + logcpk +
                                 EF_catMedium + EF_catHigh,
                               data = stepwise_data,
                               selection = "bidirection",
                               select = "AIC",
                               # significant level for entry
                               sle = 0.25,
                               # significant level for stay
                               sls = 0.15,
                               method = "efron",
                               weights = NULL,
                               best = NULL)
# 6 variables are selected: logcre, age, EF_catMedium, EF_catHigh, bp, sodium

### Variable selection using stepwise Cox model using AICc
stepwise_model3 <- stepwiseCox(Surv(time, event) ~ gender + smoking + diabetes + bp +
                                 anemia + age + sodium + platelets + logcre + logcpk +
                                 EF_catMedium + EF_catHigh,
```

```
                                data = stepwise_data,
                                selection = "bidirection",
                                select = "AICc",
                                # significant level for entry
                                sle = 0.25,
                                # significant level for stay
                                sls = 0.15,
                                method = "efron",
                                weights = NULL,
                                best = NULL)
# AICc 9 variables: logcre, age, EF_catMedium, EF_catHigh, bp, sodium, anemia, logcpk, diabetes

### Variable selection using stepwise Cox model using SBC
stepwise_model4 <- stepwiseCox(Surv(time, event) ~ gender + smoking + diabetes + bp +
                                  anemia + age + sodium + platelets + logcre + logcpk +
                                  EF_catMedium + EF_catHigh,
                                data = stepwise_data,
                                selection = "bidirection",
                                select = "SBC",
                                # significant level for entry
                                sle = 0.25,
                                # significant level for stay
                                sls = 0.15,
                                method = "efron",
                                weights = NULL,
                                best = NULL)
# SBC 5 variables: logcre, age, EF_catMedium, EF_catHigh, bp
# Extract data from the models
steps2 <- stepwise_model3$`Process of Selection`[, "Step"]
enteredEffect1 <- stepwise_model3$`Process of Selection`[, "EnteredEffect"]
sl1 <- stepwise_model1$`Process of Selection`[, "SL"]
aic2 <- stepwise_model2$`Process of Selection`[, "AIC"]
aic3 <- stepwise_model3$`Process of Selection`[, "AICc"]
sbc4 <- stepwise_model4$`Process of Selection`[, "SBC"]

# Determine the maximum length
max_len <- max(sapply(list(steps2, enteredEffect1, sl1, aic2, aic3, sbc4), length))

# Function to pad vectors with NA to make their length equal to max_len
pad_vector <- function(vec, max_len) {
  length(vec) <- max_len
  return(vec)
}

# Apply the function to each vector
steps2 <- pad_vector(steps2, max_len)
enteredEffect1 <- pad_vector(enteredEffect1, max_len)
sl1 <- pad_vector(sl1, max_len)
aic2 <- pad_vector(aic2, max_len)
aic3 <- pad_vector(aic3, max_len)
sbc4 <- pad_vector(sbc4, max_len)

# Create the data frame
```

```r
model_selection <- data.frame(
  Step = steps2,
  EnteredEffect = enteredEffect1,
  SL = round(as.numeric(sl1),4),
  AIC = round(as.numeric(aic2), 2),
  AICc = round(as.numeric(aic3), 2),
  SBC = round(as.numeric(sbc4), 2)
)
model_selection[is.na(model_selection)] <- c("-")

# Create table using kable
model_selection |> kable(booktabs = T,
                         caption = "Table 3: Summary Table of Model Selection", digits = 4) |>
  kable_styling(latex_options = c("HOLD_position"), font_size = 10)
mc_dat = nonpara_dat %>% mutate(age_cat = case_when(age <= mean(age) ~ "Low", age > mean(age) ~ "High"))
  mutate(cre_cat = case_when(creatinine_log <= log(2.5) ~ "Low",
                             creatinine_log > log(2.5) ~ "High")) %>%
  mutate(age_cat = factor(age_cat)) %>%
  mutate(cre_cat = factor(cre_cat)) %>%
  mutate(EF_cat = factor(EF_cat, levels = c("Low", "Medium", "High"))) %>%
  as.data.frame()
mc_surv_ef = survfit(Surv(time, event == 1) ~ EF_cat, data = mc_dat)
mc_surv_cre = survfit(Surv(time, event == 1) ~ cre_cat, data = mc_dat)
mc_ef_surv_log = survfit(Surv(log(time + 1), event == 1) ~ EF_cat, data = mc_dat)
mc_cre_surv_log = survfit(Surv(log(time + 1), event == 1) ~ cre_cat, data = mc_dat)

g1 = ggsurvplot(mc_ef_surv_log, data = mc_dat, fun = "cloglog", risk.table = FALSE,
                xlab = "log(time)", ylab = "log[-log(Survival Probability)]",
                ggtheme = theme_minimal(), xlim = c(1.5, 6)) +
  labs(title = "Figure 4.1: Log of Negative Log of Estimated Survival Functions \nby EF Group")
g2 = ggsurvplot(mc_cre_surv_log, data = mc_dat, fun = "cloglog", risk.table = FALSE,
                xlab = "log(time)", ylab = "log[-log(Survival Probability)]",
                ggtheme = theme_minimal(), xlim = c(1.5,6)) +
  labs(title = "Figure 4.2: Log of Negative Log of Estimated Survival Functions \nby Log-transformed Cr
gridExtra::grid.arrange(g1$plot, g2$plot, nrow = 2)

g3.1 = ggsurvplot(mc_surv_ef, data = mc_dat, risk.table = FALSE, ggtheme = theme_minimal())
g3.2 = ggadjustedcurves(coxph(Surv(time, event == 1) ~ EF_cat, data = mc_dat), variable = "EF_cat",
                        data = mc_dat, ggtheme = theme_minimal())
km_fit = g3.1$plot$data
cox_fit = g3.2$data
cox_fit$EF_cat = cox_fit$variable

g3 = ggplot(cox_fit, aes(x = time, y = surv, group = EF_cat, color = EF_cat)) + geom_step() +
  geom_step(data = km_fit, aes(x = time, y = surv, group = EF_cat, color = EF_cat), lty = 3) +
  labs(title = "Figure 5.1: Plot of Observed vs. Fitted Survival By EF Group", y = "Survival Distributi
  caption = "Straight line: Fitted Survival \nDashed line: Observed Survival") +
  theme_minimal() +
  theme(legend.position = "top") +
  scale_color_manual(
    name = "EF Group",
    values = c("red", "green","blue"),
    labels = c("Low", "Medium","High")
```

```
  )


g4.1 = ggsurvplot(mc_surv_cre, data = mc_dat, risk.table = FALSE, ggtheme = theme_minimal())
g4.2 = ggadjustedcurves(coxph(Surv(time, event == 1) ~ cre_cat, data = mc_dat), variable = "cre_cat",
                        data = mc_dat, ggtheme = theme_minimal())
km_fit = g4.1$plot$data
cox_fit = g4.2$data
cox_fit$cre_cat = cox_fit$variable

g4 = ggplot(cox_fit, aes(x = time, y = surv, group = cre_cat, color = cre_cat)) + geom_step() +
  geom_step(data = km_fit, aes(x = time, y = surv, group = cre_cat, color = cre_cat), lty = 3) +
  labs(title = "Figure 5.2: Plot of Observed vs. Fitted Survival By Creatinine Group", y = "Survival Di
  caption = "Straight line: Fitted Survival \nDashed line: Observed Survival") +
  theme_minimal() +
  theme(legend.position = "top") +
  scale_color_manual(
    name = "Creatinine Group",
    values = c("blue", "red"), # Change these colors based on your data
    labels = c("Low", "High") # Custom labels for the legend
  )

g3 / g4
scho <- coxph(Surv(time, event == 1) ~ EF_cat + bp + creatinine_log + age + sodium, data = mc_dat)
ggcoxzph(cox.zph(scho), var = c("creatinine_log"), df = 2, nsmo = 1000)
ggcoxzph(cox.zph(scho), var = c("EF_cat"), df = 2, nsmo = 1000)
# Create table using kable
model_summary <- tibble(stepwise_model2$`Coefficients of the Selected Variables`) %>% select(-Variable)

model_summary <- model_summary |>
  mutate(var = c("logcre", "age", "EF\\_catMedium", "EF\\_catHigh", "bp", "sodium"),
         coef = as.numeric(coef),
        `exp(coef)` = as.numeric(`exp(coef)`),
        `se(coef)` = as.numeric(`se(coef)`),
        z = as.numeric(z),
        `Pr(>|z|)` = as.numeric(`Pr(>|z|)`)) %>%
  relocate(var, coef, `exp(coef)`, `se(coef)`, z, `Pr(>|z|)`)

model_summary |> kable(booktabs = T, escape = F,
      caption = "Table 4: Summary of Cox Proportional Hazard Model",
      digits = 4, row.names = TRUE,
      col.names = c("Variable", "$\\beta$", "$exp(\\beta)$", "$SE(\\beta)$", "Z test statistics", "P-val
  kable_styling(latex_options = c("HOLD_position"), font_size = 10)
mc_surv_age = survfit(Surv(time, event == 1) ~ age_cat, data = mc_dat)
mc_surv_cre = survfit(Surv(time, event == 1) ~ cre_cat, data = mc_dat)

plot(mc_surv_age, col = c("red", "blue"),
     fun = "cumhaz", xlab = "Time(Days)", ylab = "-logS(t)",
     main = "Figure 7.1: Negative Log of Estimated Survival Functions For Age Groups")
legend("topleft", legend = c("High", "Low"),
       title = "Age Group", col = c("red", "blue"), lty = 1)

plot(mc_surv_cre, col = c("red", "blue"),
```

```
     fun = "cumhaz", xlab = "Time(Days)", ylab = "-logS(t)",
     main = "Figure 7.2: Negative Log of Estimated Survival Functions For Creatinine Groups")
legend("topleft", legend = c("High", "Low"),
       title = "Creatinine Group", col = c("red", "blue"), lty = 1)
fit_weibull = eha::phreg(Surv(time, event) ~ gender + smoking + diabetes + bp + anemia +
                  age + sodium + platelets + log(creatinine+1) + log(cpk+1),
                data = model_data, dist = "weibull")

# AIC Both direction
library(MASS)
# stepAIC(fit_weibull)

final_weibull <- eha::phreg(formula = Surv(time, event) ~ bp + age + sodium + logcre, data = stepwise_da

# Extract relevant information
weibull_summary <- tibble(
  Variable = c("bp", "age", "sodium", "logcre", "log(scale)", "log(shape)"),
  Coef = round(as.vector(final_weibull$coefficients),4),
  `Exp(Coef)` = round(exp(as.vector(final_weibull$coefficients)),4),
  `se(Coef)` = round(sqrt(as.vector(diag(final_weibull[["var"]]))),4),
  `Wald p` = round(1 - pchisq((as.vector(final_weibull$coefficients)/sqrt(as.vector(diag(final_weibull[

# Create table using kable
weibull_summary |> kable(booktabs = T, escape = F, caption = "Table 5: Summary of Weibull PH Model Fitt
                         digits = 4, col.names = c("Variable", "$\\beta$", "$exp(\\beta)$", "$SE(\\beta
  kable_styling(latex_options = c("HOLD_position"), font_size = 10)
step_model_final <- coxph(Surv(time, event == 1) ~ EF_catHigh +EF_catMedium + bp + logcre + age + sodiu

# Calculate predicted risks
predicted_risks <- predict(step_model_final, newdata = stepwise_data, type = "risk")
 # Time points for ROC analysis
time_points <- c(50, 250)

# Calculate ROC curves at specified times
roc_50 <- timeROC(T = stepwise_data$time, delta = stepwise_data$event, marker = predicted_risks, times =
roc_250 <- timeROC(T = stepwise_data$time, delta = stepwise_data$event, marker = predicted_risks, times

# Extract AUC values
auc_50 <- roc_50$AUC
auc_250 <- roc_250$AUC

# Print AUC values
#print(paste("AUC at 50 days:", auc_50))
#print(paste("AUC at 250 days:", auc_250))

# Plot ROC curves
plot(roc_50$FP, roc_50$TP, type = "l", col = "red", xlab = "1 - Specificity", ylab = "Sensitivity", mai
lines(roc_250$FP, roc_250$TP, type = "l", col = "blue")
legend("bottomright", legend = c("50 days", "250 days"), col = c("red", "blue"), lty = 1)
abline(0, 1, col = "black", lty = 2)
# Define the function for calculating C-statistic
boot_c_statistic <- function(original_data, indices) {
 # Creating a bootstrap sample
```

```r
 boot_data <- original_data[indices, ]

 # Fit the Cox model to the bootstrap sample
 fit <- coxph(Surv(time, event == 1) ~ EF_catHigh + EF_catMedium + bp + logcre + age + sodium, data = s

   # Calculate the concordance statistic using the updated function
   concordance <- concordance(fit)$concordance
   return(concordance)
 }

 # Perform bootstrapping for C-statistic
 set.seed(123) # for reproducibility
 boot_results_c_stat <- boot(data = stepwise_data, statistic = boot_c_statistic, R = 400)

 # Calculate the average C-statistic
 mean_c_stat <- mean(boot_results_c_stat$t)
 #print(mean_c_stat)
 # Define the bootstrap function for calibration metrics using logistic regression
boot_calibration_logistic <- function(original_data, indices) {
 boot_data <- original_data[indices, ]
fit <- coxph(Surv(time, event == 1) ~ EF_catHigh + EF_catMedium + bp + logcre + age + sodium, data = st


  # Predicted risks for the original dataset
   predicted_risks <- predict(fit, newdata = original_data, type = "risk")

  # Fit a logistic model for calibration
   calibration_model_logistic <- glm(event ~ predicted_risks, data = original_data, family = "binomial"

   # Calibration slope (coefficient of predicted_risks)
   calibration_slope_logistic <- coef(calibration_model_logistic)["predicted_risks"]

   return(calibration_slope_logistic)
 }

 # Perform bootstrap
 set.seed(123)
 boot_results_logistic <- boot(data = stepwise_data, statistic = boot_calibration_logistic, R = 400)

 # Calculate the average calibration slope
mean_calibration_slope_logistic <- mean(boot_results_logistic$t)
# print(mean_calibration_slope_logistic)
```