Bayesian Networks for Censored Data

By Cormac Kinsella, Satwik Chandra, Cian Mullarkey, Dermot Nolan, Ali Düzenli, James Friel, Cormac Slattery, Andrew Bermingham and Mary O'Connell.

Table of Contents

Introduction

Cardiovascular disease is the leading cause of death globally [1] and has therefore been the subject of many scientific studies. In this paper, we develop models that aim to improve upon the predictive accuracy attained by Ahmad in 2017 [2] which collected data from patients who had experienced heart failure in a Pakistani hospital to try to understand the relationship between major risk factors and cardiovascular risk in a Pakistani context.

We attempt to build a model that can accurately predict the probability of survival using a given set of covariates, *conditional* on experiencing heart failure. The aim of this analysis is to identify variables that are both *malleable* and predictive of death, which can guide physicians in their choice of treatment plans. It is important to note that our analysis is probably *not* helpful for screening purposes at the population level. The individuals in our sample have *already experienced heart failure,* which means that we don't have a good sense of how predictive the attributes that we're modelling are for predicting the probability of the event occurring. This also means that our analysis does not look to measure the effect size of some intervention on the risk of a bad cardiovascular outcome, but we are instead studying the relationship between certain covariates and cardiovascular risk

Bayesian networks are a powerful method for understanding causal relationships in engineering and medicine. Survival analysis is an area of biostatistics devoted to modelling the time to survival of a person after some event and Cox's Proportional Hazard models and Kaplan-Meier estimators are two methods that attain good performance in settings where there are right-censored observations – i.e. some of the observations in our data set drop out of the analysis before we observe the event, so we don't have information about when the event occurs for those observations, we just know is that the event had *not* occurred up to a certain point. The primary goal of these models is to derive *survival functions,* which estimate the probability of individuals surviving beyond time *t*.

In this paper, we look to apply machine learning algorithms in the field of biostatistics. We evaluate learning algorithms for predicting Bayesian network algorithm performance and study the performance of the model on an observational dataset. We use a Bayesian Network with Inverse Probability of Censoring Weights and a Bayesian interpretation of a Cox's Proportional Hazards model.

The three Bayesian networks that we built unfortunately failed to attain good predictive performance. While our simple best performing model achieved an accuracy of 78.6%, it did so by predicting that almost all of the patients would survive, and miscategorised 13 of the 14 patients who died. When we tried to increase the number of nodes in the network, the network predicted that more patients would die, but it did so at a rate worse than chance. We had more success with our other research direction, implementing the BN-Cox model, which generated survival probabilities that were almost equal to the original Cox model as desired.

Literature Review

Survival analysis, broadly speaking, is concerned with time-to-event data for some event of interest. It has numerous applications in varied domains, from credit risk analysis in econometrics to predictive policing in sociology. One extremely valuable application is the modelling of time-to-death of patients with a specific condition. In this context, survival analysis may estimate time-to-death for a single group or compare time-to-death between several groups. In this project, we consider a data set relating to a single group of patients with that data consisting of measurements of several covariates. We then aim to analyse the relationship between these covariates and time-to-death for these patients.

Censored Data

When performing survival analysis, researchers are confronted with data sets which may come in different forms, often presenting their own challenges. One such example, which this project examines, is the challenge of working with a data set containing censored data. Censored data occurs when certain event times are known to fall only within a given interval, while others occur at known times. In particular, this project focuses on techniques for dealing with right-censored failure time data. Instances of right censoring may appear in data sets for different reasons which may be temporal, financial, or even ethical.

Consider a data set consisting of measurements related to n individuals under study, then we may be confronted with three principal types of right censoring. The first, Type I censoring, is the case where the event is observed only if it occurs prior to some prespecified time. Type II censoring arises when the study continues until the failure of the first r individuals, where r is some predetermined integer $r < n$. Finally, Type III censoring, or competing risks censoring, appears when some individuals under study experience some competing event which leads to their being removed from the study.

Due to their prevalence in the literature, methods for dealing with censored data are invaluable statistical tools. Perhaps the best known of these tools, which this project will treat in detail, is the Kaplan-Meier estimate for the survival function first proposed in 1958 [3]. Another popular approach is the use of inverse probability weighted estimators, such as by Wijeysundera et al. [4] for estimation of health care costs, which this project also makes use of. The use of a modified likelihood function for censored data is a feature of a class of models known as censored regression models, first developed by Tobin in 1958 [5]. These methods stand alongside less refined techniques such as complete case analysis which may discard valuable information.

Cox's Proportional Hazard Model

In 1972, David Cox introduced the groundbreaking Cox's Proportional Hazard (CPH) model for dealing with right-censored failure time data. The CPH model is essentially a regression model for the hazard (or force of mortality) function. Being the first example of a semi-parametric regression model in the literature, the development of the CPH model

represented an important advancement over the nonparametric approach of Kaplan & Meier and the parametric approaches of authors such as Feigl & Zelen [6].

The thrust of any survival analysis technique is the modelling of time-to-event occurrences. By undertaking survival analysis using the CPH model, we may analyse the relationship between variables and the predicted events. The survival function is defined as the probability of an individual surviving beyond a given time t:

$$S(t) = \mathbb{P}(T > t)$$

where T denotes the occurrence of our event of interest.
The hazard function, given by

$$\lambda(t) = \lim_{\Delta t \to 0} \frac{\mathbb{P}(t \leq T < t + \Delta t | T \geq t)}{\Delta t}$$

is related to the survival function $S(t)$ by

$$\lambda(t) = -\frac{d}{dt} log(S(t))$$

and so we may estimate the survival probability from the hazard function.

A simple version of the CPH model with time-independent risk factors has hazard function

$$\lambda(t) = \lambda_0(t) exp(\boldsymbol{\beta}^t \boldsymbol{X})$$

where $\lambda_0(t)$ is called the baseline hazard function and $\boldsymbol{\beta}^t \boldsymbol{X} = \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_n X_n$ where the β_is are coefficients corresponding to the risk factors X_i. The semi-parametric nature of the CPH model is due to the fact that this baseline hazard function $\lambda_0(t)$ may be constrained to follow some distribution or be totally unspecified.

Assuming that the hazard ratio of two observations is constant in time, we may define

$$\gamma = \frac{\lambda(t_2)}{\lambda(t_1)} = \frac{exp(\beta' X_2)}{exp(\beta' X_1)}$$

and thus we obtain an estimate for the survival probability

$$S(t) = S_0(t)^\gamma = S_0(t)^{exp(\boldsymbol{\beta}^t \boldsymbol{X})}$$

Parameter Estimation

In order to perform parameter estimation in the case of right-censored failure times, consider a sample of n individuals in which k individuals fail at times $t_1, t_2, \ldots, t_k$ and the remainder are right-censored. Letting $R(t_i) = l : t_l \geq t_i$ be the risk set at time t_i, then the probability that individual i fails at time t_i given those at risk is

$$\frac{\lambda(t_i; X_i)}{\Sigma_{l \in R(t_i)} \lambda(t_i, X_l)} = \frac{exp(\beta X_i)}{\Sigma_{l \in R(t_i)} exp(\beta X_l)}$$

from which we obtain the log-likelihood

$$\log L(\beta) = \sum_{i=1}^{k} \left(\beta X_i - \log \sum_{l \in R(t_i)} \exp \beta X_l \right)$$

Maximising this function with respect to β, we attain our estimates for β.

Kaplan Meier Estimator

The Kaplan-Meier (KM) estimator is a non-parametric method of estimating the survival function. It is a common technique in medical research [7], used to measure the fraction of patients living for a certain amount of time after treatment and as such is well suited to the aims of this project. The technique is applied by first dividing the time-to-event variable into discrete time intervals.Then, the probability of survival at each interval is calculated based on the number of individuals still at risk and the number of observed events.

The Model

Consider a sample of n individuals, with each individual i having survival time t_i. Assume also that the survival times are in ascending order, with t_i being the ith survival time, i.e., $t_1 \leq t_2 \leq ... \leq t_n$.

We have the initial probability of surviving beyond time 0 as $S(t_0) = 1$. For each distinct time point in the sample, let t_i be the i^{th} smallest survival time, and let k_i be the number of individuals who experience an event at time t_i. Then, the KM estimator of the survival function at time t_i is:

$$S(t_i) = S(t_{i-1}) \cdot (1 - \frac{k_i}{r_i})$$

where $S(t_{i-1})$ is the estimated survival probability at the previous time point, r_i is the number of individuals at risk of experiencing the death event at time . This is the number of individuals who have not experienced the death event before time t_i or whose data had not been right censored before time t_i. Then, k_i is the number of individuals who experience the death event at time . This can be rewritten as:

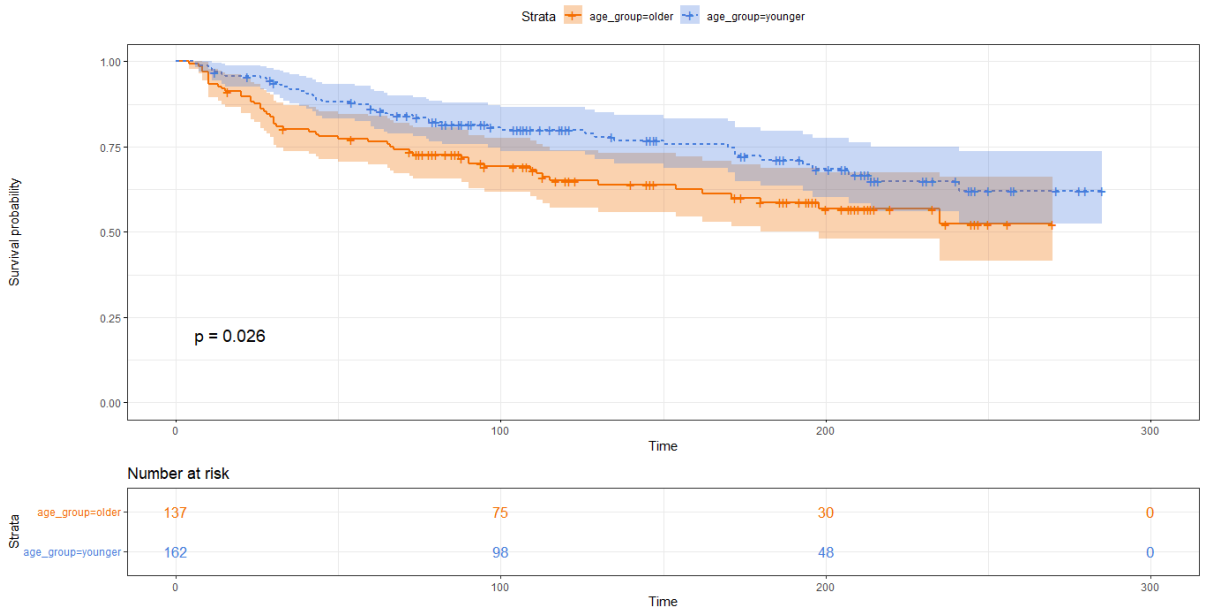
$$S(t_i) = \prod_{j=1}^{i} (1 - \frac{k_j}{r_j})$$

A clear way to visualise this process it to use a 6 column table as shown here for the initial lines of our dataset:

i	t_i	r_i	k_i	$1-\frac{k_i}{r_i}$	$S(t_i)$
1	4	299	1	0.9966555	0.996656
2	6	298	1	0.9966443	0.993311
3	7	297	2	0.993266	0.986622
4	8	295	2	0.9932203	0.979933
5	10	293	6	0.9795222	0.959866

Model Evaluation

Log-rank tests are one way in which we can compare estimated survival curves for different groups. These are carried out by first assuming a null hypothesis H_0 that there is no difference between the groups in the probability of a death event. We then make our decision to accept or reject this hypothesis based on our calculation of the log-rank test statistic [8]. An example of two survival curves from our data is shown below, with the data depending on whether individuals are of age above or below 60 years. It is clear from the plot alone that those older than 60 have a lower survival probability. Calculating the test statistic for these groups results in a p-value of 0.03, leading us to reject H_0 at a 5% level of significance.




```
> print(surv_diff)
Call:
survdiff(formula = Surv(time, DEATH_EVENT) ~ age_group, data = heart_data)

                    N Observed Expected (O-E)^2/E (O-E)^2/V
age_group=older   137       52     41.2      2.81      4.95
age_group=younger 162       44     54.8      2.11      4.95

 Chisq= 5  on 1 degrees of freedom, p= 0.03
```

Bayesian Networks

A directed acyclic graph (DAG) is a set of nodes connected by edges so that no circular paths exist. They are fundamental to several areas of data science and are used to represent the causal dependencies of a set of covariates. Each node represents one of the covariates, and a directed edge between two nodes indicates a correlation between them. The acyclic criteria ensure that there are no circular dependencies that may interfere with the model. DAG's are generally constructed using a combination of expert knowledge and correlation analysis. Purely using human intuition to determine the structure may result in important dependencies being overlooked, while solely using machine-learning algorithms can lead to overfitting. Furthermore, it is often the case [9] that increasing the complexity of the DAG by, for instance, increasing the number of covariates does not lead to a significant increase in accuracy but does increase bias. This is especially true when dealing with a small or heavily censored data set.

Constructing Bayesian Networks

Bayesian networks are a common way to model dependencies and uncertainty among a set of covariates and are commonly used in medical research [10]. In general, the use case for DAGs is that by explicitly modelling causal dependencies, we can account for those dependencies in our research design, which allows us to uncover more accurate estimates of causal relationships. There are various problems with this kind of research design,

primarily in relation to the fact that nonlinear functional forms return biassed estimates when interaction terms are included in the estimating equation, and the presence of time-varying interaction effects is quite common in medical settings [11].

A requirement for a Bayesian Network is that its weights must be learnt. These weights can be thought of the probability of observing a value X_i for node i conditional on its parent nodes like so:

$$\mathbb{P}(X_i|\text{Parents}(i))$$

Now one of the assumptions that a Bayesian Network makes is that nodes on the same level are conditionally independent of one another and thus the joint probability of observing a particular set of values for nodes $1, \ldots, p$ is given by

$$\mathbb{P}(\boldsymbol{X}) = \prod_{i=1}^{p} \mathbb{P}(X_i|\text{Parents}(i))$$

where $\boldsymbol{X} = (X_1, \ldots, X_p)$. We use DAGs to construct a model that can predict the likelihood of death occurring for a given patient based on the information collected upon their admission. In the following sections, we will explore a few methods by which this can be done.

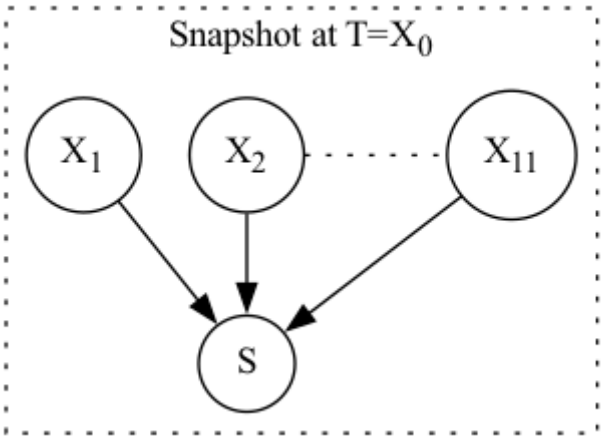
Method 1: Bayesian Network Interpretation of the CPH Model

The first method we examine is the Bayesian Network Interpretation of the CPH Model [12]. To begin, the parameters of the CPH model were estimated from our data set according to the procedure previously outlined. Then we designate random variables $X_1, X_2, ..., X_{11}$ representing risk factors as parents of the survival node S (in our case, this is the binary variable DEATH_EVENT where 0 = alive and 1 = dead). Then, letting X_j be the j^{th} entry from the dataset with $X_j = X_{j1}, X_{j2}, \ldots, X_{j11}$ with $j = 1, \ldots, n$, we define each γ_j by the hazard ratio according to the combination of the parent states. γ_j is equal to the hazard ratio of the conditioning case X_j to the baseline case X_0, i.e., the case in which all risk variables are absent, and thus:

$$\gamma_j = \frac{\exp \boldsymbol{\beta}^t \boldsymbol{X}_j}{\exp \boldsymbol{\beta}^t \boldsymbol{X}_0} \text{ for all } j \in \{1, \ldots, n\} \tag{2}$$

Unlike the CPH model, static Bayesian networks capture a snapshot of a system at a certain time. Therefore, we need to represent time explicitly. We achieve this by adding an indexing variable T for time, capturing each discrete point in time that is of interest, e.g., every day, every two weeks, etc. This time variable can be omitted if we are interested in the prediction at one point in time, say, in one year. The figure below shows an example of such a model

(we will call it the BN-Cox model), showing the relationship between risk factors , the time variable (T), and the survival node (S).



Our Bayesian model, configured with 11 risk factors $X_1, \ldots X_{11}$.
Snapshots can be taken by computing the baseline hazard λ_0 at time $T = t$.

$T = X_0$	X_1	X_2	X_3	X_4	X_5
$\lambda_0(t)$	age	anaemia	creatinine_ phosphokinase	diabetes	ejection_ fraction
X_6	X_7	X_8	X_9	X_{10}	X_{11}
high_blood_ pressure	platelets	serum_ creatinine	serum_sodium	sex	smoking

Next, we compute the conditional probability table “weights” $\beta \in \mathbb{R}^{11}$ for the predictor variables. Recall that we can obtain the survival probabilities $S(t)$ from Equation (1) in the CPH model. Therefore, for each time value T considered, we may assess a set of survival probabilities:

We may determine the survival probabilities directly from the parameters of the CPH model by considering (Eq. 3). First, we configure all risk factor cases in (Eq. 2) to find all hazard ratio values. We then obtain the baseline survival probability at the first point in time from the CPH model and from (Eq. 1) find the survival probability. The survival probability calculated for each combination of risk factors corresponds to the conditional probability of survival. Hence, the conditional probability to be encoded in the CPH model may be estimated by:

$$\mathbb{P}(s|X, T = t) = S_0(t)^{exp(\beta' X)} \text{(Eq. 3)}$$

Where s refers to the survived state, 0 in our case, X is all of the available predictors and S_0 is the baseline survivability.

Method 2: Bayesian Network with Weight Assignments

Structural Learning of a Bayesian Network

Using BIC, and Hill Climbing Algorithm

We now learn the structure of our Bayesian network B using a score-based approach, whereby a score is assigned to each possible structure F. Suppose we have a BN structure F on a dataset D. Define the score by

$$S(F, D) = p(F|D)$$

The goal now is to maximise the score. Applying Bayes law, we derive:

$$\mathrm{S}(F, D) \propto p(D|F)p(F)$$

To make sense of the structural learning, let us for now assume that the prior $p(F)$ is uniform. We are now left with the task of computing $\mathbb{P}(D|F)$ which can be done by marginalising over all possible parameterisations θ of F, hence we can derive

$$p(D|F) = \int p(D|F, \theta)p(\theta|F)\, dp$$

which can be thought of as the averages over all possible parameters. In a large sample space of possible factorisations, the measures $\mathbb{P}(D|G, \theta)$, $\mathbb{P}(\theta|G)$ are multivariate Guassians in law [13]. Approximating the mean of the Guassian $\mathbb{P}(D|F, \theta)$ with MLE $\hat{\theta}$, we propose Schwarz's BIC Score [14] by

$$\mathrm{BIC}(E, D) = \log p(D|\hat{\theta}, F) - d/2 \log N$$

where d coincides with the number of free parameters. $d/2 \log N$ can be thought of as the regularisation "penalty term" to prevent overfitting.

The next goal is to minimise the BIC score across all possible structures F. Since there are an exponential number of possible structures for F, it is impossible to maximise the BIC score using a brute force approach. Instead, we will use a heuristic with the following outline:

Hill Climbing Algorithm

1. Initialise graph B with empty graph set F
2. Iteration:
    a. For each pair of nodes, add, remove or reverse an edge.
    b. Compute BIC score on graph
    c. If structure minimises the BIC score, update F and continue iteration
3. Continue iteration until no single arc change can lower the score any further

One can think of this process as an EM problem called Sufficient Expectation Maximisation with the following steps:

E-Step: Compute the expected statistic, the BIC score using current structure F.
M-Step: Update the structure that minimises the expected statistic, the BIC score.

Using Chow-Liu

An alternative to the Hill climbing algorithm is The Chow-Liu algorithm which consists of the following steps and equations [15]:

1. Compute marginal counts $f_u(i)$ and pairwise counts $f_{uv}(i,j)$ for all variables x_u and x_v in the data set.

2. Compute mutual information $\hat{I}(x_i, x_j)$ for all pairs of variables x_i and x_j using the formula:

$$\hat{I}(x_i, x_j) = \sum_{u,v} f_{uv}(i,j) \log \frac{f_u(i) f_v(j)}{f_{uv}(i,j)}$$

3. Compute the maximum weight spanning tree (MWST) using Kruskal's algorithm. Pick a root and orient the edges away from the root.

4. Set the parameters in the conditional probability tables (CPTs) for each node to be their maximum likelihood estimates:

$$P(x_i \mid x_{\pi(i)}) = \frac{f_{\pi(i)}(j) f_{i\pi(i)}(i,j)}{f_{\pi(i)}(j)}$$

where $\pi(i)$ is the parent of x_i in the tree.

Parameter Learning in Bayesian Network with IPCW & Maximum Likelihood Estimation

In the context of Inverse Probability of Censoring Weighting (IPCW) as in Vock [16], subjects with E = 1 are oversampled if patients with unknown E are excluded. Subsequently, any machine learning technique can be employed for risk prediction, taking the computed weights as weight parameters. The general IPCW method entails the following steps:

1. Utilising the training data, assess the function $G(t) = P(Ci > t)$ signifying the probability that the censoring time exceeds t via the Kaplan-Meier estimator of the censoring times' survival distribution.

2. For each patient i in the dataset, compute an inverse probability of censoring weight. Patients with an event status that remains unknown at τ (i.e., those censored prior to τ) are allocated a weight of $\omega_i = 0$. The other patients receive weights inversely proportional to the estimated likelihood of being censored after their observed follow-up time.

3. Implement a pre-existing prediction method on a weighted version of the training set, in which each member of the training set is assigned a weight factor of ω_i. For instance, if $\omega_i = 3$, the observation is treated as if it appeared three times in the dataset.

By employing this weight assignment technique, conventional machine learning classifiers can be utilised for modelling censored data.

Essential Assumptions for IPCW Implementation

The IPCW method relies on the following assumptions, as given by Vock:

- No unmeasured confounders exist for censoring.
- Upon conditioning the hazard of censoring on the documented history, it no longer depends on X (sequential ignorability of censoring).
- The data is censored at random, meaning the censoring mechanism is independent of the outcome but may rely on the covariates.

Provided these assumptions are met and all prognostic factors are documented, IPCW estimators will entirely rectify the bias resulting from dependent censoring.

Exploratory Data Analysis

The dataset that we are using is a panel of heart disease patients from Pakistan collected between 2015 and 2019. The criteria for inclusion in the dataset is previous admittance to hospital with severe heart problems, which is classified as a stage III or IV heart disease by the New York Heart Association.

The features of the dataset are all of the details that were systematically captured by hospital records about the patients. These range from demographic characteristics such as age and gender to the results of diagnostic tests such as the level of creatinine phosphokinase. Below are the first ten rows of the dataset, along with the variable names for each of the columns.

Heart_failure_dataset

age	anaemia	creatinine_phosphokinase	diabetes	ejection_fraction	high_blood_pressure	platelets	serum_creatinine	serum_sodium	sex	smoking	time	DEATH_EVENT
75	0	582	0	20	1	265000	1.9	130	1	0	4	1
55	0	7861	0	38	0	263358	1.1	136	1	0	6	1
65	0	146	0	20	0	162000	1.3	129	1	1	7	1
50	1	111	0	20	0	210000	1.9	137	1	0	7	1

65	1	160	1	20	0	327000	2.7	116	0	0	8	1
90	1	47	0	40	1	204000	2.1	132	1	1	8	1
75	1	246	0	15	0	127000	1.2	137	1	0	10	1
60	1	315	1	60	0	454000	1.1	131	1	1	10	1
65	0	157	0	65	0	263358	1.5	138	0	0	10	1

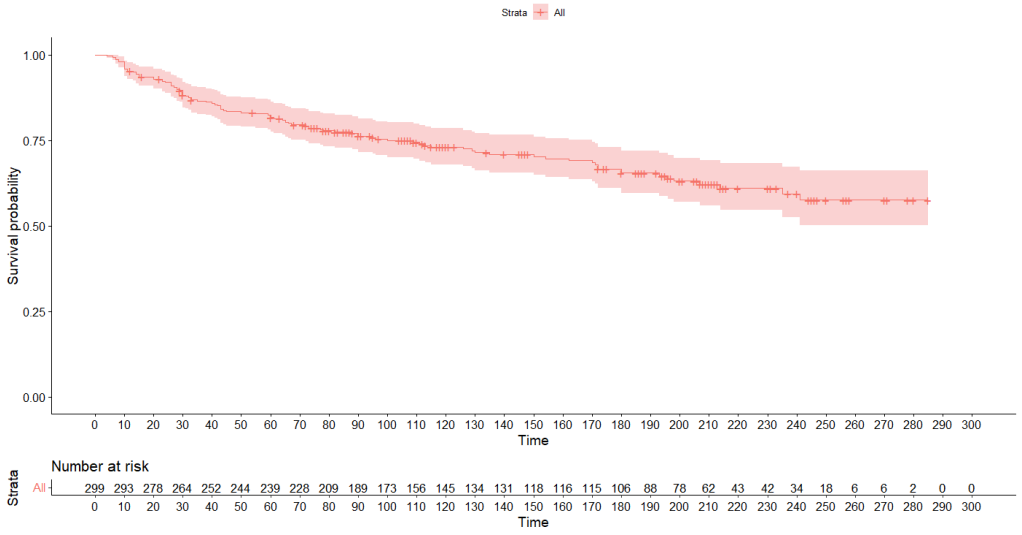
Methodology

Correlation Analysis:

- To investigate the relationships between variables, particularly the dependent variable DEATH_EVENT, we generated a correlation matrix (Appendix D). The results showed that the variables Diabetes and Sex had no correlation with DEATH_EVENT. To reduce the complexity of the Bayesian Network, these variables were excluded from the analysis.
- Furthermore, we observed that Age had a positive correlation with DEATH_EVENT (0.25), as did serum creatinine (0.29), while ejection_fraction had a negative correlation (-0.27).

Inverse Probability of Censoring Weights (IPCW) Calculation:

- To compute the IPCW weights, we first determine whether or not a patient's Death_Event had been censored before Time t=100. This information was then used to assign a value of 1 or 0 to each subject in the dataset, under the column name censoring_indicator.
- Subsequently, we employed the Kaplan-Meier estimator (figure below) to predict the probability of a patient's data being censored at Time=t. We can then use these values to assign weights to each patient, that weight being zero if the patient was censored before Time = 100, and the reciprocal of the Kaplan Meier estimate otherwise.
- This means that patients who died later in the trial receive larger weights since their data was more likely to be censored.



Data Discretization:

From a theoretical perspective, it seemed appealing to us to treat each of the variables in a Bayesian Network as having Gaussian distributions [17]. However, in practice this wasn't feasible. The libraries for learning Bayesian Networks that facilitate the use of continuous variables do not allow for the use of sample weights, and the libraries that do allow for the use of weights only work with discretized datasets. Given that over 20% of the observations in our dataset had been censored after 100 days, we decided to prioritise building models that could account for censoring, and because of this we had to discretize the data.

The primary goal of discretizing the data is to group the continuous values into a certain number of classes. This allows the variable to be treated as categorical, meaning the conditional probabilities can be estimated using empirical frequency analysis. There is a tradeoff between increasing the number of classes and the variance of the model. If we generate lots of categories, then there will be fewer observations in each group, which increases the volatility of the modelling. To combat this, we used a low number of classes with the data roughly evenly distributed among them.

Bayesian Network Structure and Parameter Learning:

Three different network structures were proposed that were found using both machine-learning algorithms and medical research.

Bayesian Network Using Chow-Liu algorithm

The first IPCW model was structured using the Chow-Liu algorithm (see literature review). This fully algorithmic approach yielded the network structure seen in Appendix C.

Bayesian Network with Expert Knowledge

We also sought to estimate the performance of network structures that we defined internally based on the relevant medical literature. To do this, we did a systematic search of scientific papers to identify whether there were relationships between the variables in our dataset. The output from this process was a large DAG (Appendix A), which we believe offers the most complete treatment of the relevant relationships. We also created a smaller and simpler DAG (Appendix B) which was initially intended to be a prototype but ended up yielding the most accurate results of all the models. This can most likely be attributed to the fact that we had very little data available to us. This made more complicated models prone to overfitting, thus rewarding models that use fewer inputs.

Bayesian networks were then created for each of the above structures using the Pomegranate library in Python and employing a 70:30 split of the data into training and test data.

CPH Model and BN-Cox Model:

Many packages were used in creating both models, mainly the *bnlearn* package and the survival package, both in R. The dataset was split into training and testing subsets using a 80:20 ratio to measure the predictiveness of the CPH model. It should be noted that the splitting was done at random, and while the output of the diagnostic tests conducted are subject to change, the average results have been discovered to be favourable upon numerous reruns. The model was constructed by creating a survival object with the variables time and DEATH_EVENT, and 11 risk factors mentioned on Page 10. The baseline survival curve was then derived from this model using survfit (Appendix H). In terms of diagnostics, the coefficients, p-values, and z-values of parameters were evaluated using summary, and the proportional hazards assumption was tested using cox.zph. A function was built to get the survival probabilities of an observation at a specific time, and this function was used on the dataset with its outputs stored for the BN interpretation.

For the BN-Cox model, some risk factors were removed for the sake of simplicity. The decision on which factors to remove was based on the previously mentioned model diagnostics (Results chapter for more information). An empty DAG was defined for the remaining variables and after directing the nodes, the variables were discretized. The conditional probabilities for each node were accurately calculated, normalised where necessary, and used in the fitting process. To evaluate the BN-Cox model, the survival function of a person with similar risk factors was derived from both models.

Model Evaluation:

The performance of the trained Bayesian network was assessed using the test dataset. Various metrics, including model accuracy, classification report, and F1 scores, were calculated to evaluate the model's effectiveness.

Results

Bayesian Network with IPCW weights

We present the model accuracy, which indicates the proportion of accurate predictions made by the Bayesian Network concerning the total samples. This metric is calculated by dividing the number of correct predictions by the total number of predictions. It should be noted that while this is a useful metric for assessing the accuracy of a model, its results can be distorted by the fact that the overall death rate of the sample is relatively low. This means that any model that predicts a low probability of death for each patient will get a high accuracy score, even though the model itself may not be very good at predicting whether someone is at risk of death.

The Bayesian network incorporating the IPCW weights and the chow-liu algorithm achieved an accuracy of 68.9%. This demonstrates the model's effectiveness in predicting the DEATH_EVENT variable based on the given dataset.

Confusion Matrix: BN with IPCW , algorithm: Chow-Liu

		Actual Values	
		1	0
Predicted values	1	0	0
	0	28	62

The confusion matrix serves as a valuable tool for evaluating the performance of classification models, as it displays the number of correctly and incorrectly predicted values. In our analysis, the confusion matrix revealed that the model predicted 62 true positives and 28 false negatives. Given that our objective was to determine the probability of cv_event, this indicates that the model provided accurate predictions 68.9% of the time.

Classification matrices are another useful tool for model evaluation, although we didn't find them very helpful in this context given the poor model performance. Two classification matrices are available in Appendix K for the large model and Chow Liu model.

Bayesian Network with IPCW - Expert Knowledge

The first DAG used was a simple model including only age, anaemia, and blood-pressure as covariates. The structure of this graph is illustrated in Figure 1. This model had an accuracy rate of 78.6%.

For our second model, we did more research into the relevant medical literature to better understand the relationships between the different covariates in the model and their effect on the probability of death in heart failure patients. Our inspiration for this approach was drawn

from Bandyopadhyay et al. [18] ,in which the authors produced a much more complete DAG structure.

We hoped that by modelling more of the relevant covariates explicitly, we would be able to produce a more accurate model. This was not the case. We found that this model drastically *decreased* prediction accuracy to 50.7%. The reason for this seems to be that the low number of observations in the dataset meant that the dataset was too sparse to learn a model with 10 covariates, each of which had between 2 and 5 levels. This led to less stable conditional probability tables which generated poor predictive power.

As discussed, the accuracy metric does a poor job of capturing relevant information about the performance of the model that we care about. For this reason, it's useful to examine the confusion matrix for each of the models, which illustrates the predicted values of the model versus the actual values of the model.

The simple model predicts that almost all patients will still be alive after 100 days. Given that almost 80% of the patients are still alive at that point, it makes sense that the model would classify the majority of observations as more likely to survive than to not survive, if the covariates aren't sufficiently predictive.

Interpreting the confusion matrix for the larger model is difficult. It predicts that 50% of the patients died and 50% of the patients survived, which is much lower than the true survival rate of ~80%. Given that the model narrowly beats flipping a coin for determining the classification, it did not seem helpful to try to draw further inferences.

Confusion Matrix: BN with IPCW , Expert Knowledge: Small

		Actual Values	
		1	0
Predicted values	1	1	0
	0	13	47

Confusion Matrix: BN with IPCW , Expert Knowledge: Large

		Actual Values	
		1	0
Predicted values	1	6	22
	0	8	25

CPH Model

As mentioned in the Methodology section, the quality of the CPH model was measured using the coefficients, p-values and z-values. Coefficients demonstrate how much a unit change in a variable changes the hazard ratio. P-values are test statistics used to calculate whether the null hypothesis H_0 is rejected or failed to be rejected. With regards to the risk factors, the null hypothesis is that factors have no effect on the outcome (DEATH_EVENT). A generally accepted cut-off point is 0.05. The z-value evaluates whether the weights (β) of risk factors are significantly different from 0 or not. A higher level of |z-value| indicates that the variable is significantly different from 0.

Regarding the CPH model built for this project, the variables with acceptable p-values according to the previously mentioned cut-off point were **age**, **ejection_fraction**, **high_blood_pressure**, **serum_creatinine**, and **serum_sodium**. Out of those, **age**, **ejection_fraction**, and **serum_creatinine** stand out with higher levels of |z-value|, while **high_blood_pressure** has the largest exp(coef) value, meaning it contributes to more increase in the hazard function for each unit increase than any other risk factor. For the purposes of simplification and computational efficiency, the BN-Cox model was built using these 4 parent nodes (Appendix I).

Whether the proportional hazards assumption was violated or not was also examined. The proportional hazards assumption is the assumption that the hazard functions of survival curves for different strata are proportional. In a significant majority of reruns, this assumption was not violated by any variable. However, in some cases, **ejection_fraction** has returned a p-value below 0.05, violating the proportional hazards assumption. This finding was ultimately discarded, as it only occurred in a small number of cases, and the p-value in those cases was significantly close to 0.05 (Appendix J).

The predictiveness of the CPH model was also examined using the concordance index. The concordance index, which is a measure of model discrimination level, takes values between 0.5 and 1, with 1 meaning perfect discrimination. As demonstrated in Appendix G, our model had a C-index of 0.79, a highly satisfactory figure.

Bayesian Interpretation of CPH Model

As previously explained, the BN-Cox model was constructed using the aforementioned 4 parent nodes. Continuous variables (age, serum_creatinine) were made discrete where necessary. As explained in [12], Bayesian networks present a snapshot of the system at a given time. Thus, time was specified after variables were properly altered. To find the conditional probabilities at that time, the baseline hazard function was derived at first. Then, each potential value that variables could take were derived by multiplying those values with their respective coefficients(β). The conditional probabilities of the parent nodes, as expressed by Equation 11 on the same paper, were then calculated and stored. Since the sum of these self-derived conditional probabilities were not 1, they had to be normalised before proceeding. The next step was calculating the conditional probability of the child node (DEATH_EVENT| parent nodes). Since the conditional probability of the parent nodes were already individually calculated, the joint probability was calculated and merged in a

straightforward way. The survival function mentioned in the Methodology part was also used when calculating the conditional probability of the child node.

After all conditional probabilities were calculated, and the Bayesian network fitted, the predictiveness of the BN-Cox model was evaluated by comparing it with the original CPH model. The idea behind this approach was to prove that this interpretation of the CPH model would have highly similar results, and this was definitely the case in our example with the CPH model predicting a 80.77% survival chance compared to the 80.69% calculated by BN-Cox for a person with similar characteristics.

Discussion and Conclusion

In this paper, we have tested the performance of different Bayesian Networks for predicting the probability that a patient dies in the first 100 days after experiencing a heart failure event. This work can be broken down into two primary parts. First, we built three networks using a combination of expert knowledge and structure learning algorithms. We then estimated the conditional probability tables produced by each of these models using maximum likelihood estimation, and measured their performance using accuracy scores, confusion matrices, and classification matrices. Overall, the performance of these models was disappointing. Our simple expert knowledge based DAG and the DAG that we learnt using the Chow Liu algorithm both predict that *all* or *almost all* patients will survive, and the only reason for the large difference in their levels of performance was that a much lower proportion of patients in the test data for the smaller DAG died. Since 70.8% of people survive past 100 days, the *a priori* risk of death for each individual is below 25% for all patients unconditional on covariates, and it seems like the variables in our models are not sufficiently predictive to raise this probability above 50%. As discussed in the report section, our larger expert knowledge based network was very inaccurate, likely because the large number of nodes and arcs in that model has led to volatility in the estimation of the conditional probability tables.

The second aspect of this report was the BN-Cox model. Our goal with this model, as explained above, was to see if it was possible to create a model that can generate a similar survival probability to the original CPH model. This was achieved, with the results available above.

One particular research direction that we would be excited to see explored is the use of alternative methods to sample weights for dealing with censored data. This would make it possible to use hybrid and continuous network learning algorithms, which we were not able to utilise. Given the low number of observations in the dataset, we think that Gaussian models of the distribution of each variable might yield better results, especially given the desirable properties of Gaussian networks in general [17].

However, it seems plausible to us that this wouldn't improve performance that much, with the main reason for this being the low number of observations in our dataset. In general, Bayesian Network based approaches to studying health outcomes seem more appropriate for larger, more complex datasets. This is illustrated in the literature by the fact that the most

cited papers on predicting heart failure outcomes using Bayesian networks use large proprietary datasets generated from electronic health records [19]. We were aware of this fact at the outset of our study, but we didn't have much luck with searching for larger datasets to analyse, which meant that we ultimately settled on using a dataset with a relatively low number of observations. Given the issues that we've encountered with building predictive models using this small dataset, and the availability of traditional methods from biostatistics that handle small datasets efficiently, we would be most excited about further work focussed on building bayesian networks from large electronic health record datasets.

Bibliography

[1] Roth, G. A. et al. (2021) “Global Burden of Cardiovascular Diseases and Risk Factors, 1990–2019: Update From the GBD 2019 Study” *Journal of the American College of Cardiology* 77(15): 1958-1959

[2] Ahmad, T. et al. (2017) “Survival analysis of heart failure patients: A case study.” *PLoS ONE* 12(7): e0181001.

[3] Kaplan, E. L. and Meier, P (1958) “Nonparametric Estimation from Incomplete Observations” *Journal of the American Statistical Association* 53 (282): 457–481.

[4] Wijeysundera, H. C. et al. (2012) “Techniques for estimating health care costs with censored data: an overview for the health services researcher” *ClinicoEconomics and Outcomes Research* 4: 145-55

[5] Tobin, J. (1958) “Estimation of Relationships for Limited Dependent Variables” *Econometrica* 26(1): 24-36

[6] Feigl, P. and Zelen, M (1965) “Estimation of Exponential Survival Probabilities with Concomitant Information” *Biometrics* 21 (4): 826-838

[7] Etikan, I. et al. (2017) “The kaplan meier estimate in survival analysis” *Biometrics & Biostatistics International Journal* 5(2): 55-59

[8] Mantel, N. (1966) “Evaluation of survival data and two new rank order statistics arising in its consideration”. *Cancer Chemotherapy Reports* 50(3): 163–70

[9] Tennant, P. W. G. et al. (2021) “Use of directed acyclic graphs (DAGs) to identify confounders in applied health research: review and recommendations” *International Journal of Epidemiology* 50(2): 620-632

[10] McLachlan, S. (2020) “Bayesian networks in healthcare: Distribution by medical condition” *Artificial Intelligence in Medicine* 107

[11] Puhani, P. A. (2012) “The treatment effect, the cross difference, and the interaction term in nonlinear “difference-in-differences” models” *Economics Letters* 115(1): 85-87

[12] Kraisangka, J. and Druzdzel, M. J. (2018)“A Bayesian Network Interpretation of the Cox’s Proportional Hazard Model” *International Journal of Approximate Reasoning* 103: 195-211

[13] Kass, R. E. and Raftery, A. E. (1995) “Bayes Factors” *Journal of the American Statistical Association* 90(430): 773-795

[14] Schwarz, G. (1978) “Estimating the Dimension of a Model” *The Annals of Statistics* 6(2) 461 - 464

[15] Chow, C. K. and Liu, C. N. (1968) “Approximating discrete probability distributions with dependence trees” *IEEE Transactions on Information Theory* IT-14 (3): 462–467

[16] Vock, D. M. et al. (2016) “Adapting machine learning techniques to censored time-to-event health record data: A general-purpose approach using inverse probability of censoring weighting” *Journal of Biomedical Informatics* 61: 119-131

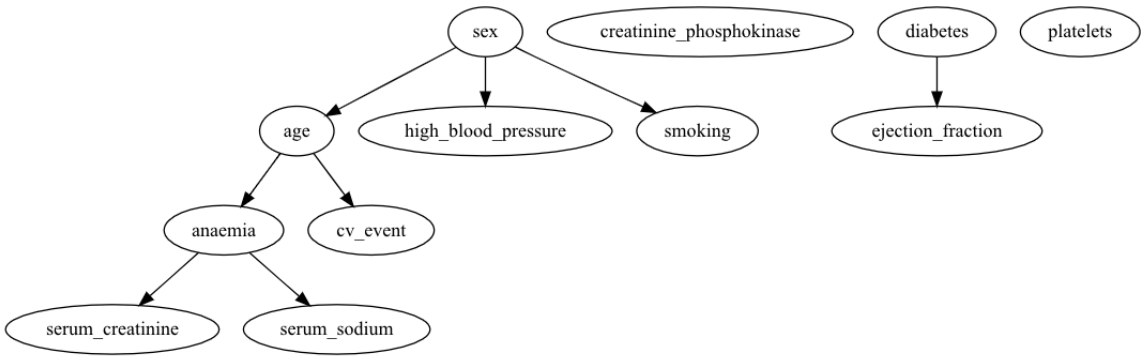
[17] Zhu, W. and Nguyen, N. L. C. (2022) “Structure Learning for Hybrid Bayesian Networks” *arXiv:2206.01356v2*

[18] Bandyopadhyay, S. et al. (2014) “Data mining for censored time-to-event data: a Bayesian network model for predicting cardiovascular risk from electronic health record data” *Data Mining and Knowledge Discovery* 29: 1033–1069

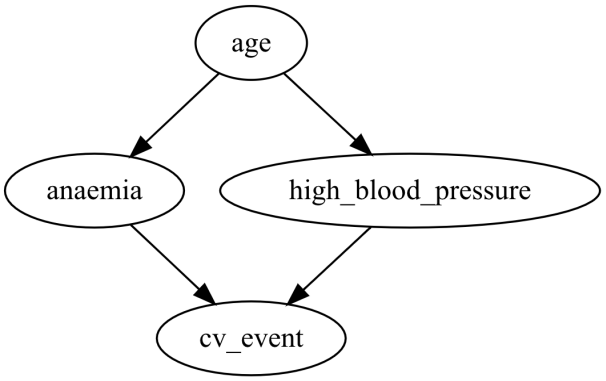
[19] Štajduhar, I. and Dalbelo-Bašić, B. (2010) “Learning Bayesian networks from survival data using weighting censored instances”, *Journal of Biomedical Informatics,* 43(4): 613-622

Appendix

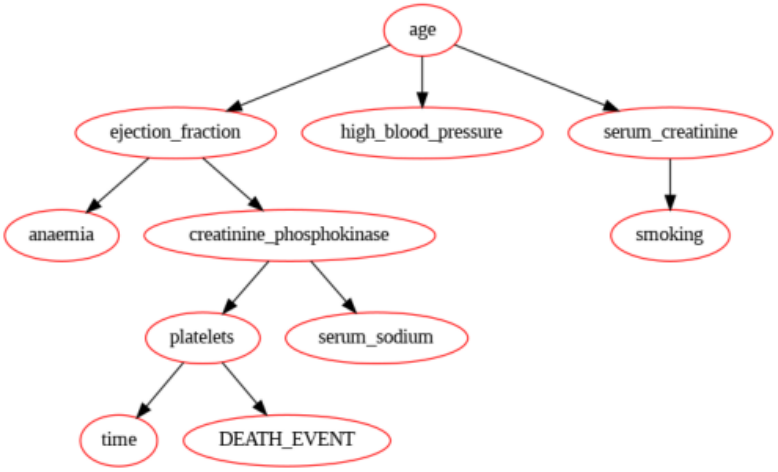
Appendix A:



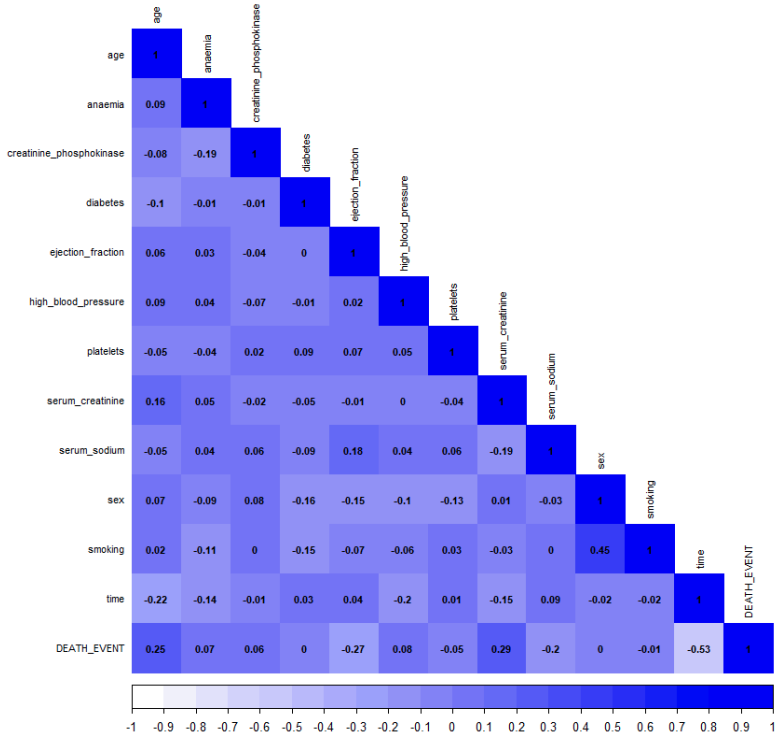
Appendix B:



Appendix C:



Appendix D:



Appendix E:

Variable	Before Discretization	Breaks	Labels	After Discretization
age	75, 55, 65, 50, 65	<= 40, 50, 60, 70, 80, 95	1,2,3,4,5,6	5,3,4,2,4
Creatinine_phosphokinase	582, 7861, 146, 111, 160	<= 10, 120 >=	0,1,2	3,3,3,2,3
ejection_fraction	20,38, 20,20,20	<= 50, 70 >=	0,1,2	1,1,1,1,1
platelets	265000,263358, 162000, 210000, 327000	<= 150000, 450000 >=	0,1,2	2,2,2,2,2

serum_creatinine	1.90, 1.10, 1.30, 1.90, 2.70	<= 0.6, 1.3 >=	0,1,2	3,2,2,3,3
serum_sodium	130, 136, 129, 137, 116	<= 135, 145 >=	0,1,2	1,2,1,2,1

Normal Ranges for Creatinine_phosphokinase, ejection_fraction, platelets, serum_creatinine, serum_sodium

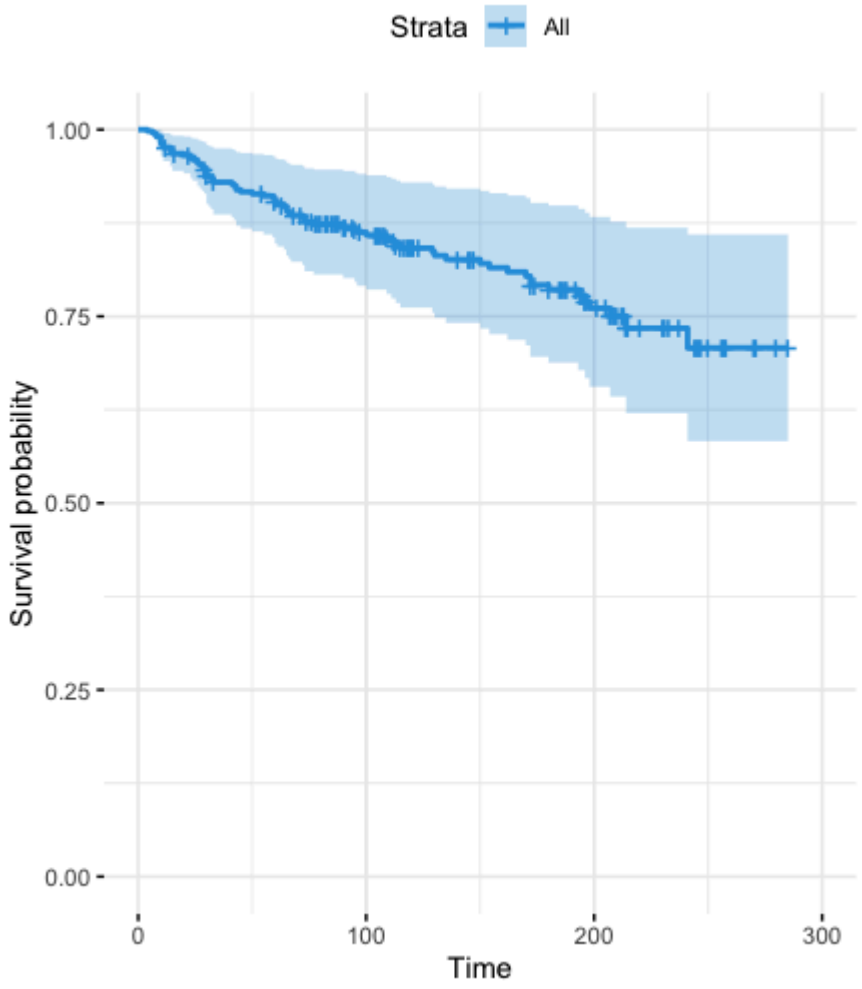
Appendix F:

```
 These results were acquired using censored weights during the training of the bayesian network

The accuracy of the model trained  using weights is
 0.6888888888888889

 The confusion matrix of the model trained  using weights is
 [[62  0]
 [28  0]]

 Other Classification metrics of the model trained  using weights is
              precision    recall  f1-score   support

           0       1.00      0.69      0.82        90
           1       0.00      0.00      0.00         0

    accuracy                           0.69        90
   macro avg       0.50      0.34      0.41        90
weighted avg       1.00      0.69      0.82        90
```

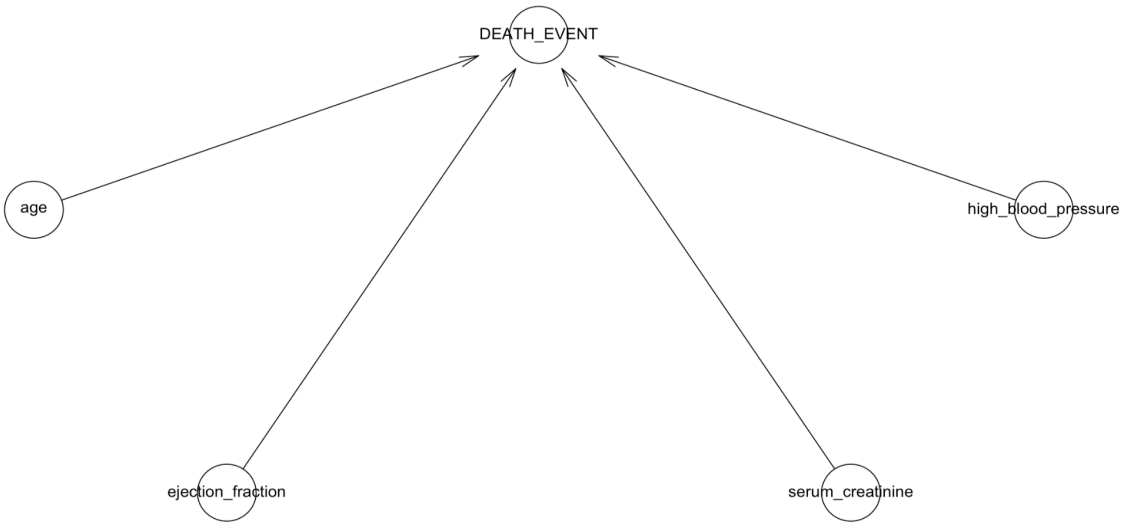
Appendix G:

```
> c_index <- rcorr.cens(-test_set$predicted_risks, Surv(test_set$time,
test_set$DEATH_EVENT))
> print(c_index) #>70%
       C Index            Dxy           S.D.              n
     0.7924081      0.5848161      0.1223364     60.0000000
       missing     uncensored Relevant Pairs     Concordant
     0.0000000     19.0000000   1686.0000000   1336.0000000
     Uncertain
  1848.0000000
```

Appendix H:



Appendix I:



Appendix J:

```
> cox.zph(cox_model)
                            chisq df    p
age                      1.03e-01  1 0.75
anaemia                  1.69e-02  1 0.90
creatinine_phosphokinase 1.02e+00  1 0.31
diabetes                 1.92e-01  1 0.66
ejection_fraction        4.69e+00  1 0.03
high_blood_pressure      8.23e-03  1 0.93
platelets                5.69e-06  1 1.00
serum_creatinine         1.52e+00  1 0.22
serum_sodium             1.10e-01  1 0.74
sex                      7.63e-02  1 0.78
smoking                  4.79e-01  1 0.49
GLOBAL                   1.17e+01 11 0.39
>
```

Appendix K:

Large Network

	precision	recall	f1-score	support
0	0.76	0.53	0.62	47
1	0.21	0.43	0.29	14
accuracy			0.51	61
macro avg	0.49	0.48	0.46	61
weighted avg	0.63	0.51	0.55	61

Chow Liu Network

	precision	recall	f1-score	support
0	1.00	0.69	0.82	90
1	0.00	0.00	0.00	0
accuracy			0.69	90
macro avg	0.50	0.34	0.41	90
weighted avg	1.00	0.69	0.82	90