

Conditional Inference Trees & Cox Regression to Predict Heart Failure Survival Time

Antonio Pano

11/10/2022

Dataset found at this link
(<https://archive.ics.uci.edu/ml/datasets/Heart+failure+clinical+records>)

- All 299 patients had left ventricular systolic dysfunction

Initial Variables:

- age: age of the patient (years)
- anaemia: presence of critically low haematocrit levels (boolean)
- high blood pressure: if the patient has hypertension (boolean)
- creatinine phosphokinase (CPK): level of the CPK enzyme in the blood (mcg/L)
- diabetes: if the patient has diabetes (boolean)
- ejection fraction: percentage of blood leaving the heart at each contraction (percentage)
- platelets: platelets in the blood (kiloplatelets/mL)
- sex: woman or man (binary)
- serum creatinine: level of serum creatinine in the blood (mg/dL)
- serum sodium: level of serum sodium in the blood (mEq/L)
- smoking: if the patient smokes or not (boolean)
- time: follow-up period (days)
- [target] death event: if the patient deceased during the follow-up period (boolean)

```
library(skimmr)
library(ggplot2)
library(dplyr)
library(tidyr)
library(survival)
library(survminer)
library(partykit)
library(coin)
library(survminer)
library(flexsurv)
library(randomForestSRC)
library(broom)
library(gtsummary)
```

Loading in the data

Creating Left Ventricular Ejection Fraction Groups set by Cardiology Experts (<https://www.ncbi.nlm.nih.gov/books/NBK459131/>). Rounding for averages instead of only using data for men and women.

```
HF <- read.csv("heart_failure_clinical_records_dataset.csv")

HF$anaemia = as.factor(HF$anaemia)
HF$diabetes = factor(HF$diabetes,levels=c(0,1),labels=c("Absent","Present"))
HF$hypertension = factor(HF$high_blood_pressure,levels=c(0,1),labels=c("Absent","Present"))

HF$sex = factor(HF$sex,levels=c(0,1),labels=c("Female","Male"))
HF$smoking = as.factor(HF$smoking)
HF$DEATH_EVENT = as.factor(HF$DEATH_EVENT)

HF <- select(HF, -high_blood_pressure)

skim(HF)
```

Data summary

Name	HF
Number of rows	299
Number of columns	13

Column type frequency:








factor	6
numeric	7

Group variables None

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
anaemia	0	1	FALSE	2	0: 170, 1: 129
diabetes	0	1	FALSE	2	Abs: 174, Pre: 125
sex	0	1	FALSE	2	Mal: 194, Fem: 105
smoking	0	1	FALSE	2	0: 203, 1: 96
DEATH_EVENT	0	1	FALSE	2	0: 203, 1: 96
hypertension	0	1	FALSE	2	Abs: 194, Pre: 105

Variable type: numeric

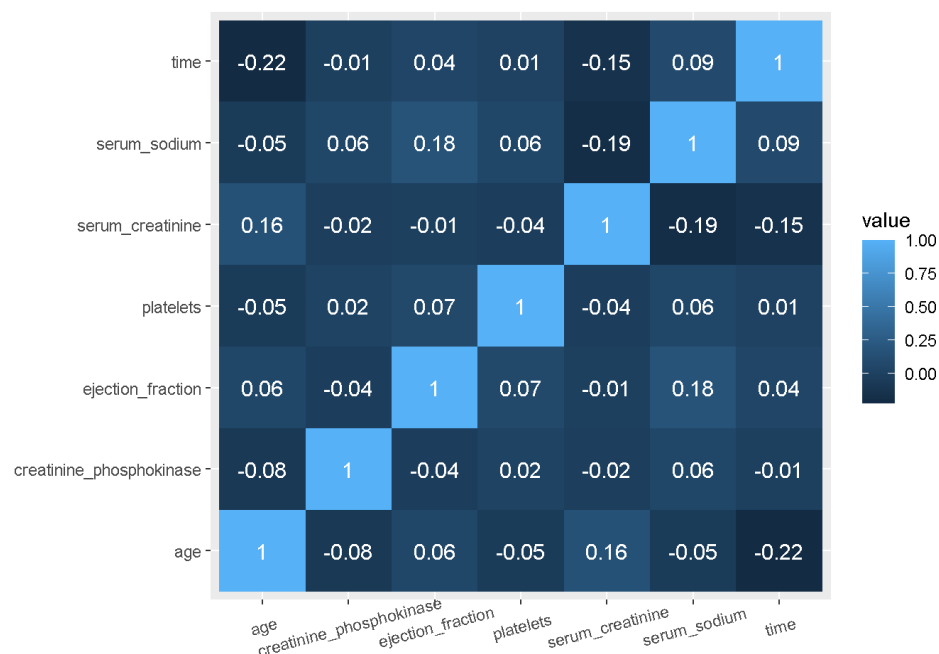
skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
age	0	1	60.83	11.89	40.0	51.0	60.0	70.0	95.0	
creatinine_phosphokinase	0	1	581.84	970.29	23.0	116.5	250.0	582.0	7861.0	
ejection_fraction	0	1	38.08	11.83	14.0	30.0	38.0	45.0	80.0	
platelets	0	1	263358.03	97804.24	25100.0	212500.0	262000.0	303500.0	850000.0	
serum_creatinine	0	1	1.39	1.03	0.5	0.9	1.1	1.4	9.4	
serum_sodium	0	1	136.63	4.41	113.0	134.0	137.0	140.0	148.0	
time	0	1	130.26	77.61	4.0	73.0	115.0	203.0	285.0	

Correlation

Time and Serum_Creatinine have a correlation to Serum_Sodium of 0.15 & 0.19, respectively.

```
cormat <- HF %>% select(where(is.numeric)) %>% cor() %>% round(2)
melted_cormat <- reshape2::melt(cormat)

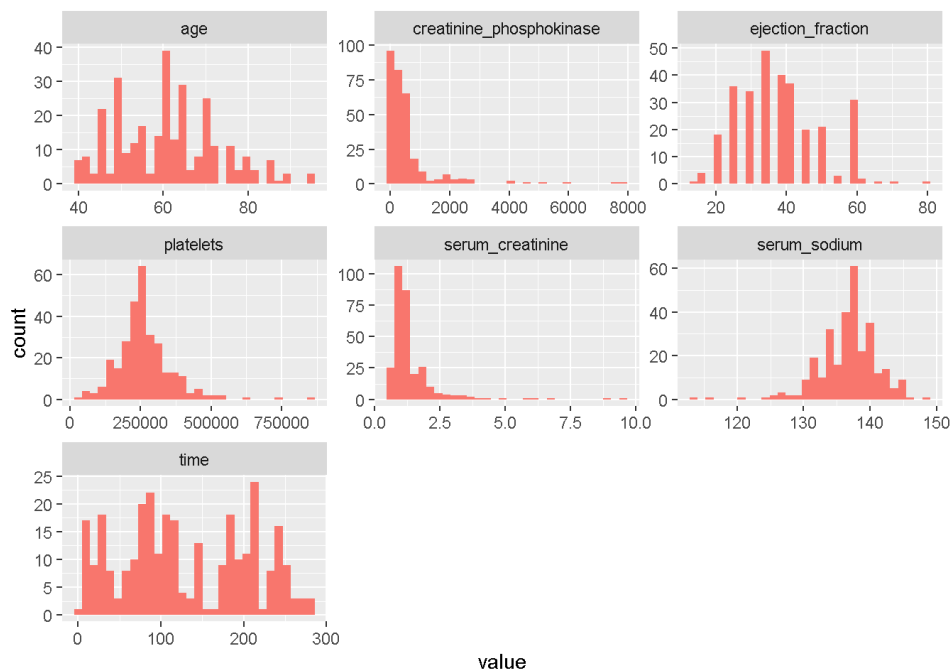
ggplot(data = melted_cormat, aes(x=Var1, y=Var2, fill=value)) +
  geom_tile() +
  geom_text(aes(Var2, Var1, label = value), color = "white", size = 4) +
  theme(axis.title.x=element_blank(),
        axis.title.y=element_blank(),
        axis.text.x = element_text(angle = 15, vjust = 0.8)
  )
```



Choosing to grab distributions based on having hypertension– what's traditionally seen as a good indicator of heart failure.

Doing so to look at, specifically, Ejection Fraction right after to see if there is correlation.

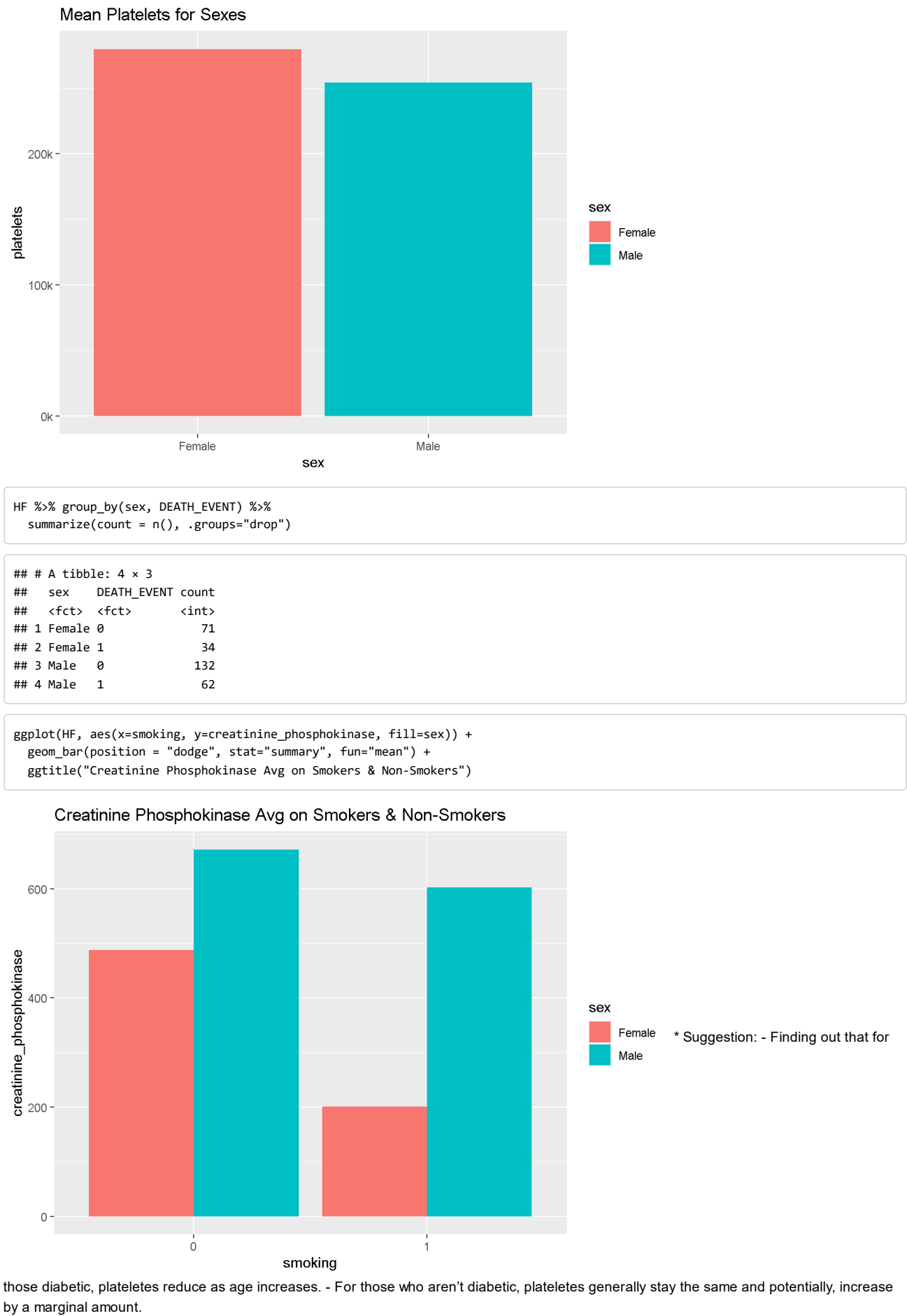
```
HF %>%
  purrr::keep(is.numeric) %>%
  gather() %>%
  ggplot(aes(value)) +
    facet_wrap(~ key, scales = "free") +
    geom_histogram(aes(fill="orange"), show.legend = FALSE)
```



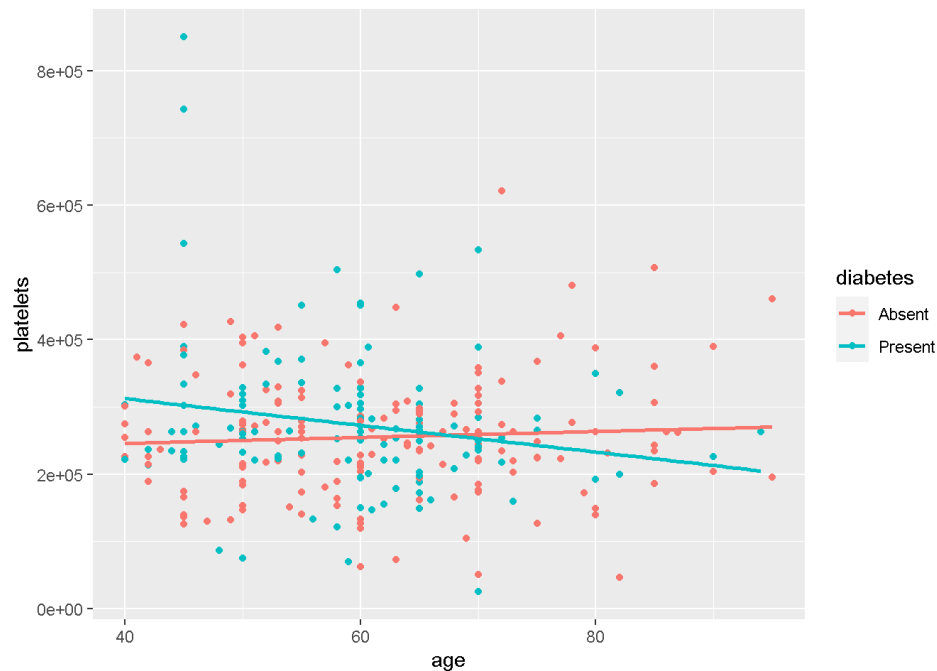
Comparing creatinine_phosphokinase to Men & Women– those who smoke and those who do not.

- Noticing that the average creatinine_phosphokinase is higher for non-smokers.

```
ggplot(HF, aes(x=sex, y=platelets, fill=sex)) +
  geom_bar(position = "dodge", stat="summary", fun="mean") +
  scale_y_continuous(labels = scales::label_number(suffix = "k", scale = 1e-3)) +
  ggtitle("Mean Platelets for Sexes")
```



```
ggplot(HF, aes(x=age, y=platelets,color=diabetes)) + geom_point() +
  geom_smooth(method='lm', se = FALSE)
```

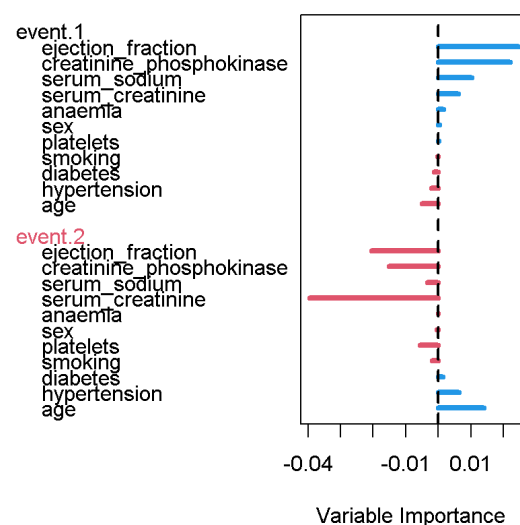
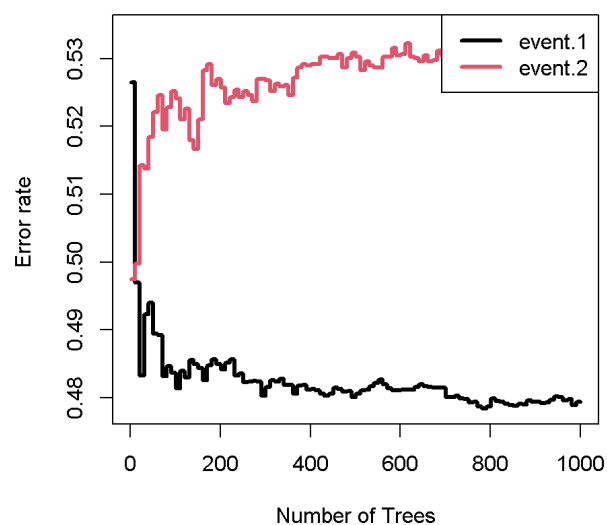


Random Forest Survival

Used to get variable importance chart.

```
# mtry means how many nodes at each split
fit <- rfsrc(Surv(time, DEATH_EVENT) ~ .,
  data = HF,
  ntree = 1000,
  importance = TRUE,
  nsplit = 5)

#fit
plot(fit)
```



```
##
##
##      event.1  event.2
## ejection_fraction 0.0248 -0.0204
## creatinine_phosphokinase 0.0220 -0.0151
## serum_sodium 0.0104 -0.0034
## serum_creatinine 0.0062 -0.0396
## anaemia 0.0016 -0.0001
## sex 0.0003 -0.0004
## platelets 0.0002 -0.0057
## smoking -0.0002 -0.0019
## diabetes -0.0014 0.0013
## hypertension -0.0021 0.0064
## age -0.0050 0.0139
```

Conditional Inference Trees - Kaplan Meier Curves

We can see we have remaining cases in which the person did not declared deceased due to the ending of the curve not dropping down to 0%.

Insights from this graph include: * Serum Creatinine is highly significant with the showcased split at 1.8 for survival prediction.

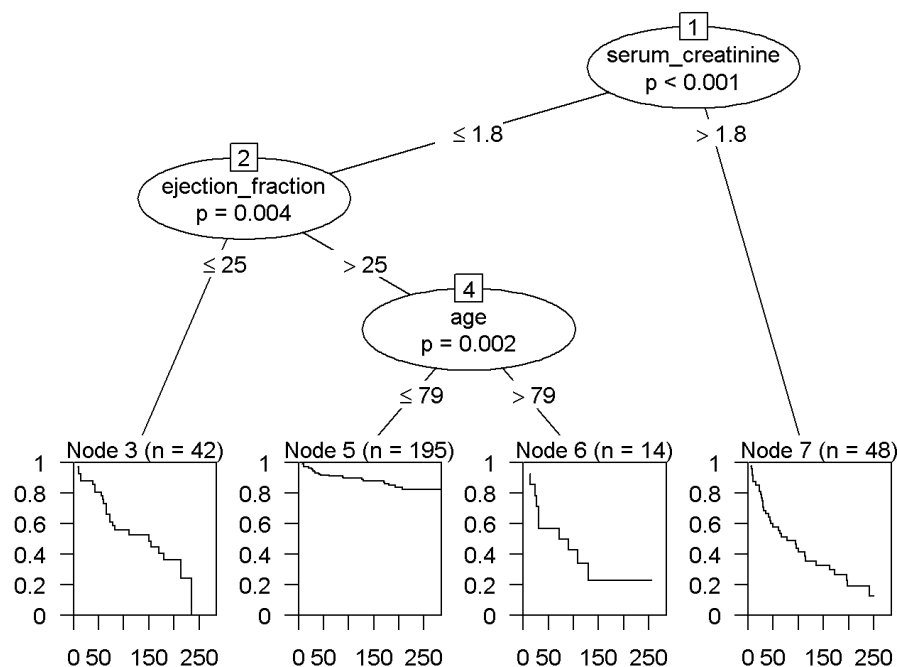
```
set.seed(0)

# Won't directly go from factor to numeric. Needed for Survival Analysis.
HF$DEATH_EVENT = as.numeric(as.character(HF$DEATH_EVENT))

# Dropping categorical Ejection Fraction.

# Creating a Conditional Inference Tree for descriptive analytics
CondInfTree <- ctree(Surv(time, DEATH_EVENT) ~ .,
                     data = HF,
                     control = ctree_control(alpha = 0.05))

plot(CondInfTree)
```

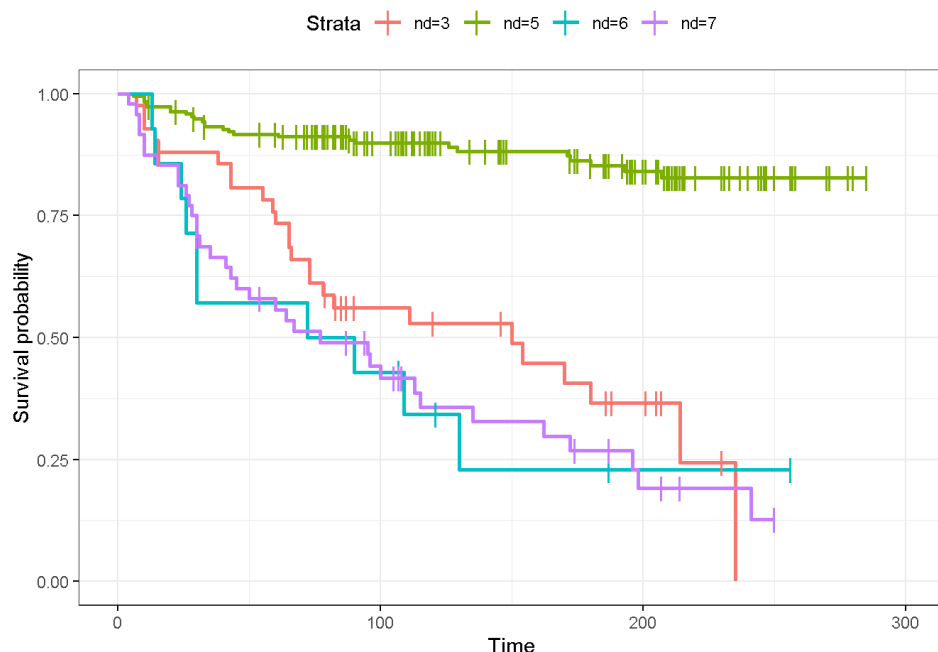


Plotting all node distributions/curves in one plot.

```
nd <- factor(predict(CondInfTree, type = "node"))

all_nd <- survfit(Surv(time, DEATH_EVENT) ~ nd, data = HF)

ggsurvplot(all_nd, data = HF,
  censor.shape = "|",
  conf.int = FALSE, #surv.median.line = "hv",
  ggtheme = theme_bw())
```



```
# Extracting survival curve for only one observation from the ctree. Perhaps an outlier.
#nd1 <- predict(CondInfTree, type = "prob")[[10]]
#summary(nd1, times=c(20, 45, 60, 80, 100, 10*(11:15)))
```

Constructing an exponential curve for previous graph's second node. * 48% probability of survival after t=150 days for patients older that have less than or equal to 1.8 in serum creatine, and an ejection fraction under 25.

```
K <- HF %>%
  filter(serum_creatinine <= 1.8, ejection_fraction <= 25)

# This one is best.
# The ~ 1 is our way of letting R know that we aren't using any x variables. Just time and whether event occurred which are both y variables.
pred_k_surv <- survfit(Surv(time, DEATH_EVENT) ~ 1, data = K)

summary(pred_k_surv, times=c(20, 45, 60, 80, 100, 10*(11:15)))
```

```
## Call: survfit(formula = Surv(time, DEATH_EVENT) ~ 1, data = K)
##
##   time n.risk n.event survival std.err lower 95% CI upper 95% CI
##   20     36      5   0.881  0.0500    0.788    0.985
##   45     33      3   0.808  0.0612    0.696    0.937
##   60     31      3   0.734  0.0688    0.611    0.882
##   80     23      6   0.587  0.0768    0.454    0.759
##  100     17      1   0.562  0.0776    0.429    0.736
##  110     17      0   0.562  0.0776    0.429    0.736
##  120     16      1   0.529  0.0798    0.393    0.711
##  130     14      0   0.529  0.0798    0.393    0.711
##  140     14      0   0.529  0.0798    0.393    0.711
##  150     13      1   0.488  0.0834    0.349    0.682
```

- No pruning was done since most trees found revolve around the same 3 variables.
- Probability of survival after 150 days for those younger than 70 is 77%.
- Probability of survival after 200 days for those younger than 70 is 70%.

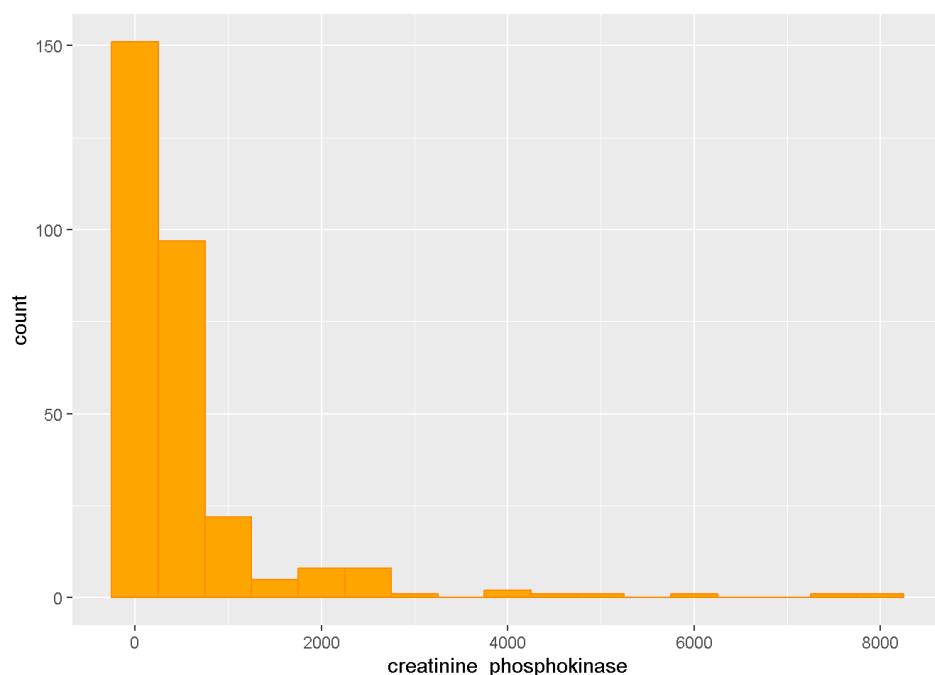
```
survfit(Surv(time, DEATH_EVENT) ~ 1, data = HF %>% filter(age <= 70)) %>%
tbl_survfit(
  times = c(150,200),
  label_header = "***{time} Day Survival (95% CI) For Those Younger Than 70**"
)
```

Characteristic	150 Day Survival (95% CI) For Those Younger Than 70	200 Day Survival (95% CI) For Those Younger Than 70
----------------	---	---

Overall	77% (71%, 82%)	70% (64%, 77%)
---------	----------------	----------------

- Creatine_Phosphokinase not being a split variable in the conditional inference tree lead me to look at in closer.

```
ggplot(HF, aes(x=creatinine_phosphokinase)) + geom_histogram(binwidth = 500, fill = "orange", color = "darkorange")
```



```
survfit(Surv(time, DEATH_EVENT) ~ 1, data = HF %>% filter(creatinine_phosphokinase <= 1000)) %>%
tbl_survfit(
  times = c(150,200),
  label_header = "***{time} Day Survival (95% CI) For Those with less than 1000 in Creatine Phosphokinase**"
)
```

Characteristic	150 Day Survival (95% CI) For Those with less than 1000 in Creatine Phosphokinase	200 Day Survival (95% CI) For Those with less than 1000 in Creatine Phosphokinase
----------------	---	---

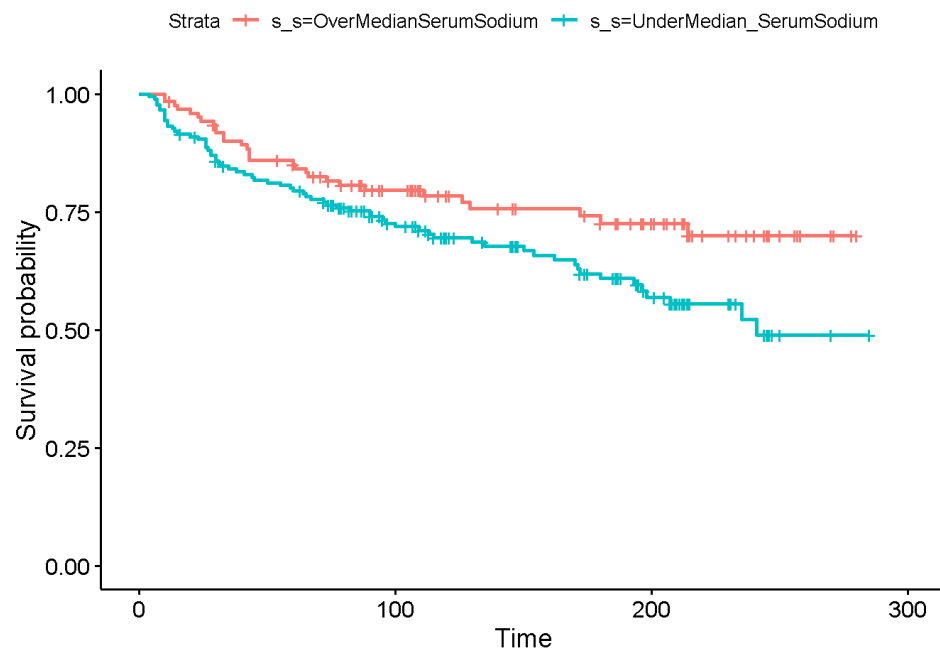
Overall	69% (64%, 76%)	62% (55%, 69%)
---------	----------------	----------------

- Looking at Serum Sodium Splitting at the median in case the lower-end outliers have too much weight.
- Finding that those with higher serum sodium have better survival rates, on average.

```
ss <- HF %>%
  mutate(s_s = ifelse((serum_sodium <= median(serum_sodium)), "UnderMedian_SerumSodium", "OverMedianSerumSodium"))

ss_fit <- survfit(Surv(time, DEATH_EVENT) ~ s_s, data=ss)

ggsurvplot(ss_fit, data = ss)
```

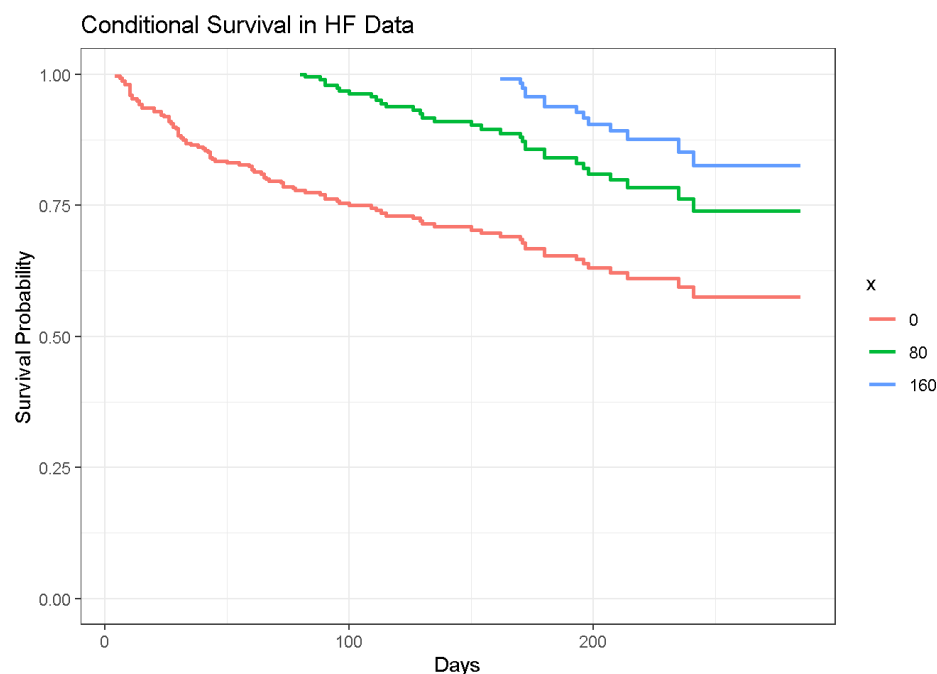
#CLEARLY, HAVING HIGHER SERUM_SODIUM MEANS HIGHER RATE OF SURVIVAL.

Conditional Survival: The probability of surviving further 't' years, given that a patient has already survived 's' years.

```
remotes::install_github("zabore/condsurv")
library(condsurv)

fit1 <- survfit(Surv(time, DEATH_EVENT) ~ 1, data = HF)

gg_conditional_surv(
  basekm = fit1,
  at = seq(0, 160, 80),
  main = "Conditional Survival in HF Data",
  xlab = "Days",
  ylab = "Survival Probability"
)
```



Cox Proportional Hazards Model (Cox Regression)

KM will make the curve based on event & time but that's all. We need to include the rest of the variables.

- At a given instance in time, someone who has hypertension is 0.42 times as likely to die as someone without hypertension adjusting for age.
- At any given instance in time, someone who does *not* have hypertension is 0.65 times as likely to die as someone who does, adjusting for age.
- Concordance: Goodness of fit for survival analysis.

```
# hypertension useful bc tree didn't output it. i paired it w/ age bc why not?
coxMod2 <- coxph(Surv(time, DEATH_EVENT) ~ hypertension + age, data=HF)
summary(coxMod2)
```

```
## Call:
## coxph(formula = Surv(time, DEATH_EVENT) ~ hypertension + age,
##       data = HF)
##
## n= 299, number of events= 96
##
##               coef exp(coef) se(coef)      z Pr(>|z|)
## hypertensionPresent 0.417717  1.518491 0.209708  1.992  0.0464 *
## age                 0.042424  1.043336 0.008693  4.880 1.06e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##               exp(coef) exp(-coef) lower .95 upper .95
## hypertensionPresent    1.518     0.6585    1.007    2.290
## age                    1.043     0.9585    1.026    1.061
##
## Concordance= 0.638 (se = 0.031 )
## Likelihood ratio test= 27.36 on 2 df,  p=1e-06
## Wald test               = 27.52 on 2 df,  p=1e-06
## Score (logrank) test = 28.25 on 2 df,  p=7e-07
```

Checking Linearity of Model * Linearity of the final cox regression is sufficient. * Anaemia is not statistically significant.

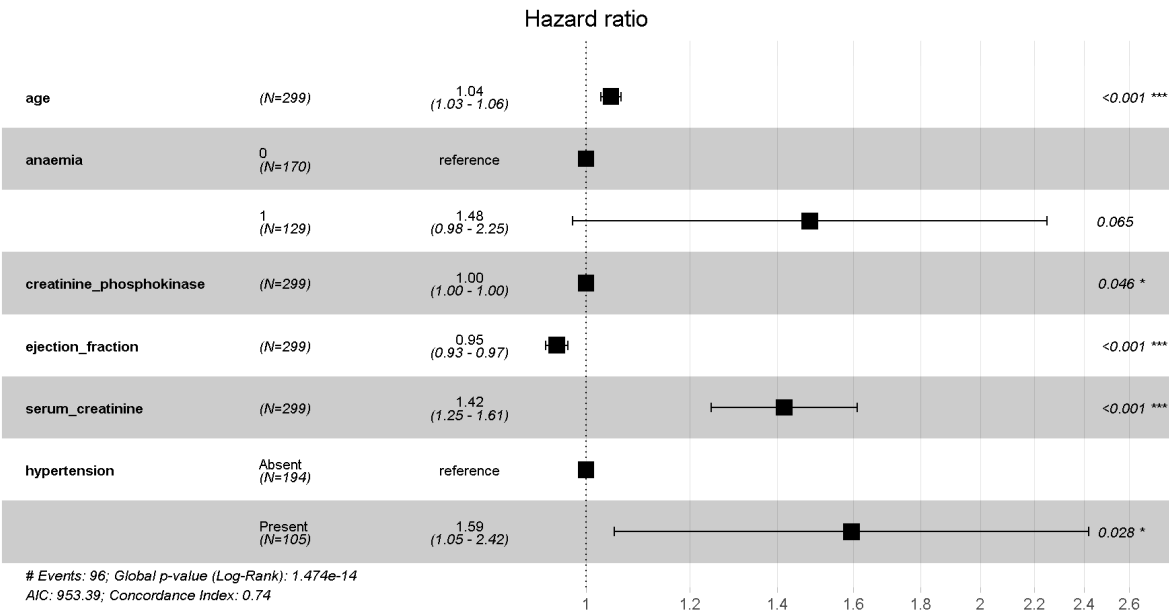
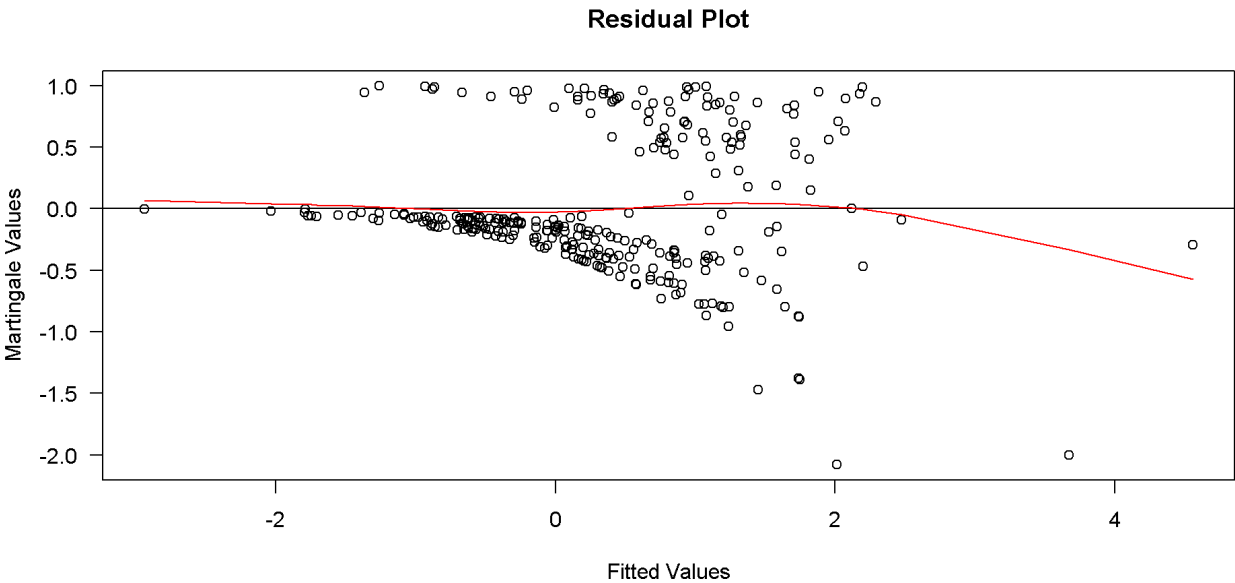
```
sigMod = coxph(Surv(time, DEATH_EVENT) ~ age+anaemia+creatinine_phosphokinase+ejection_fraction+
               serum_creatinine+hypertension, data=HF)
summary(sigMod)
```

```
## Call:
## coxph(formula = Surv(time, DEATH_EVENT) ~ age + anaemia + creatinine_phosphokinase +
##       ejection_fraction + serum_creatinine + hypertension, data = HF)
##
## n= 299, number of events= 96
##
##               coef exp(coef) se(coef)      z Pr(>|z|)
## age              4.361e-02  1.045e+00  8.853e-03  4.926 8.41e-07 ***
## anaemia1         3.933e-01  1.482e+00  2.129e-01  1.847  0.0648 .
## creatinine_phosphokinase 1.965e-04  1.000e+00  9.856e-05  1.993  0.0462 *
## ejection_fraction -5.179e-02  9.495e-01  1.005e-02 -5.152 2.57e-07 ***
## serum_creatinine   3.483e-01  1.417e+00  6.550e-02  5.318 1.05e-07 ***
## hypertensionPresent  4.668e-01  1.595e+00  2.129e-01  2.192  0.0284 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##               exp(coef) exp(-coef) lower .95 upper .95
## age              1.0446     0.9573    1.0266    1.0629
## anaemia1         1.4818     0.6749    0.9762    2.2493
## creatinine_phosphokinase 1.0002     0.9998    1.0000    1.0004
## ejection_fraction  0.9495     1.0531    0.9310    0.9684
## serum_creatinine   1.4167     0.7059    1.2460    1.6108
## hypertensionPresent  1.5948     0.6270    1.0506    2.4209
##
## Concordance= 0.738 (se = 0.028 )
## Likelihood ratio test= 77.02 on 6 df,  p=1e-14
## Wald test              = 85.82 on 6 df,  p=2e-16
## Score (logrank) test = 83.51 on 6 df,  p=7e-16
```

```
plot(predict(sigMod), residuals(sigMod, type = "martingale"), xlab = "Fitted Values",
      ylab = "Martingale Values", main = "Residual Plot", las = 1) +
  abline(h=0) +
  lines(smooth.spline(predict(sigMod), residuals(sigMod, type="martingale")), col="red")
```

```
## integer(0)
```

```
ggforest(sigMod, data = HF)
```



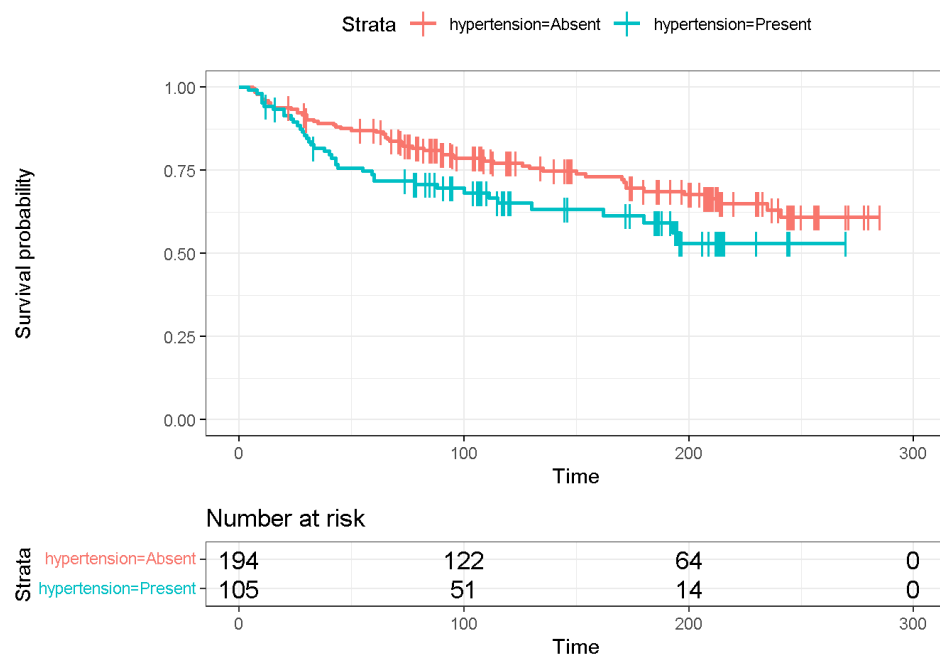
```
library(ggthemes)
finMod <- sigMod %>% tidy()

# finMod %>% mutate(upper = estimate + 1.96 * std.error,
#                   lower = estimate - 1.96 * std.error) %>%
#   mutate(across(all_of(c("estimate", "lower", "upper")), exp)) %>%
#   ggplot(aes(estimate, term, color = estimate > 1)) +
#   geom_vline(xintercept = 1, color = "gray75") +
#   geom_linerange(aes(xmin = lower, xmax = upper), size = 2.25, alpha = 0.28) +
#   geom_point(size = 4) +
#   theme_gdocs(base_size = 16) +
#   scale_color_manual(values = c("green4", "red3"), guide = "none") +
#   xlim(c(0, 3)) +
#   labs(title = "Hazard Ratios for Significant Variables", y = NULL,
#        x = "Hazard Ratio Estimates (95% C.I.)") +
#   theme(axis.text.y = element_text(hjust = 0, size = 18)) +
#   geom_text(label = exp(finMod$estimate) %>% round(2),
#             nudge_y = .2, nudge_x = .15)
```

Performing the Log-Rank Test on select variables to extract significance.

•

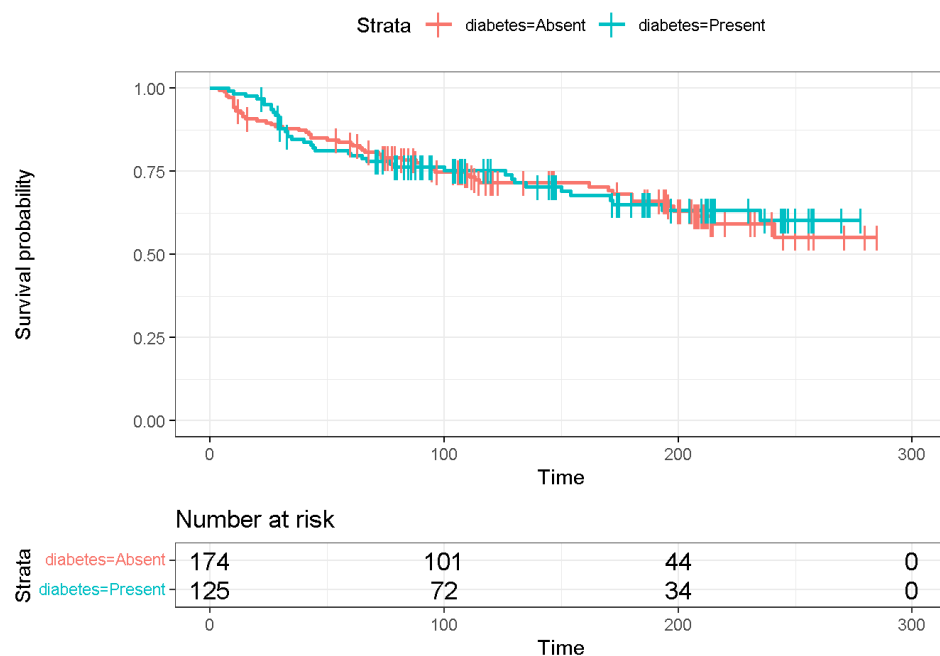
```
#Hypertension
ggsurvplot(survfit(Surv(time, DEATH_EVENT) ~ hypertension, data=HF),
  data = HF,
  censor.shape="|",
  conf.int = FALSE, #surv.median.Line = "hv",
  risk.table = TRUE,
  ggtheme = theme_bw())
```



```
survdifff(Surv(time, DEATH_EVENT) ~ hypertension, data=HF)
```

```
## Call:
## survdifff(formula = Surv(time, DEATH_EVENT) ~ hypertension, data = HF)
##
##              N Observed Expected (O-E)^2/E (O-E)^2/V
## hypertension=Absent 194      57   66.4      1.34    4.41
## hypertension=Present 105      39   29.6      3.00    4.41
##
## Chisq= 4.4 on 1 degrees of freedom, p= 0.04
```

```
#Diabetes
ggsurvplot(survfit(Surv(time,DEATH_EVENT) ~ diabetes, data=HF),
  data = HF,
  censor.shape="|",
  conf.int = FALSE, #surv.median.Line = "hv",
  risk.table = TRUE,
  ggtheme = theme_bw())
```



```
survdif(Surv(time,DEATH_EVENT) ~ diabetes, data=HF)
```

```
## Call:
## survdiff(formula = Surv(time, DEATH_EVENT) ~ diabetes, data = HF)
##
##               N Observed Expected (O-E)^2/E (O-E)^2/V
## diabetes=Absent 174      56      55   0.0172   0.0405
## diabetes=Present 125      40      41   0.0231   0.0405
##
## Chisq= 0 on 1 degrees of freedom, p= 0.8
```

Binary Logistic Regression

Creating category variables for Serum Creatinine & Creatinine Phosphokinase due to their heavy right skewness.

Using One (https://labs.selfdecode.com/blog/creatine-kinase/#:~:text=The%20low%20normal%20limit%20for,3%2C%204%2C%205%5D./)) and Two (<https://www.mayoclinic.org/tests-procedures/creatinine-test/about/pac-20384646#:~:text=The%20typical%20range%20for%20serum,52.2%20to%2091.9%20micromoles%2FL>).

```
HF <- HF %>%
  mutate(SC_Condition = cut(HF$serum_creatinine, breaks = c(0, 0.7, 1.25, Inf),
    labels = c("Low", "Normal", "High"), include.lowest = TRUE),
    CPK_Condition = cut(HF$creatinine_phosphokinase, breaks = c(0, 30, 200, 300, Inf),
    labels = c("Low", "Normal", "High", "Severely High"), include.lowest = TRUE)) %>%
  select(-serum_creatinine, -creatinine_phosphokinase)
```

```
set.seed(0)
library(caTools)

split_log <- sample.split(HF, SplitRatio = 0.7)
train_log <- subset(HF, split_log == TRUE) %>% select(-time)
test_log <- subset(HF, split_log == FALSE)

logit1 <- glm(DEATH_EVENT~., family = binomial,data = train_log)
summary(logit1)$coefficients[,4] %>% round(digits = 5)
```

```
##          (Intercept)              age
##          0.90491         0.00005
##          anaemia1         diabetesPresent
##          0.24233         0.78826
##          ejection_fraction         platelets
##          0.00002         0.58928
##          serum_sodium         sexMale
##          0.73230         0.20661
##          smoking1         hypertensionPresent
##          0.97265         0.10369
##          SC_ConditionNormal         SC_ConditionHigh
##          0.37492         0.01592
##          CPK_ConditionNormal         CPK_ConditionHigh
##          0.60066         0.27251
## CPK_ConditionSeverely High
##          0.46066
```

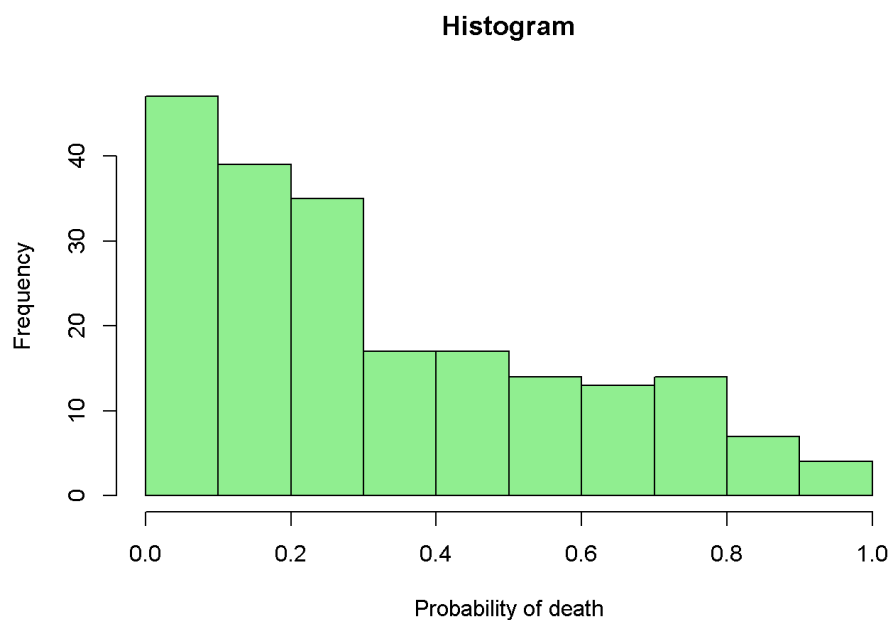
```
summary(logit1)$aic
```

```
## [1] 217.2686
```

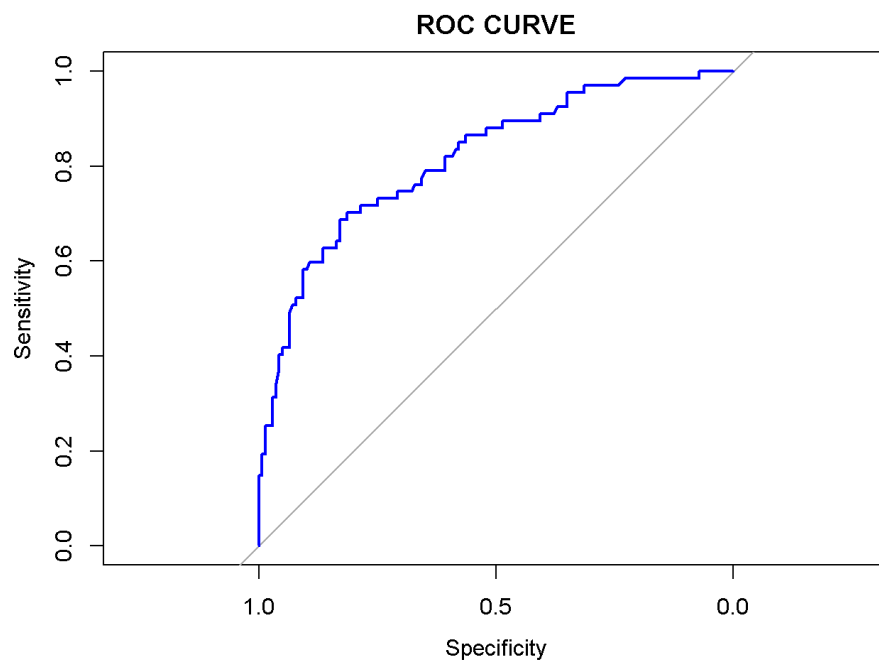
```
logit2 <- step(logit1, direction = "backward", trace = FALSE)
summary(logit2)$coefficients[,4] %>% round(digits = 5)
```

```
##          (Intercept)              age ejection_fraction hypertensionPresent
##          0.00349         0.00003         0.00010         0.04829
## SC_ConditionNormal         SC_ConditionHigh
##          0.40556         0.01190
```

```
hist(logit2$fitted.values, main=" Histogram ",xlab="Probability of death", col='light green')
```



```
r <- pROC::roc(DEATH_EVENT~logit2$fitted.values, data = train_log, plot = TRUE, main = "ROC CURVE", col= "blue")
```



```
optimal_roc <- r$thresholds[which.max(r$sensitivities + r$specificities)]

test_predicted_data <- test_log %>%
  mutate(p1=predict(logit2, newdata=test_log, type="response")) %>%
  mutate(Predict=ifelse(p1 < optimal_roc,0,1))

cm <- table(test_predicted_data$DEATH_EVENT,test_predicted_data$Predict) %>% prop.table()
rownames(cm) <- c("Obs. neg", "Obs. pos")
colnames(cm) <- c("Pred. neg", "Pred. pos")

ERROR.RESULTS <- tibble(
  Sensitivity=c(cm[1,1]/sum(cm[1,])),
  Specificity=c(cm[2,2]/sum(cm[2,])),
  FalsePositives=c(cm[2,1]/sum(cm[2,])),
  FalseNegatives=c(cm[1,2]/sum(cm[1,]))
)

efficiency <- sum(diag(cm))/sum(cm)

ERROR.RESULTS
```

```
## # A tibble: 1 × 4
##   Sensitivity Specificity FalsePositives FalseNegatives
##   <dbl>         <dbl>         <dbl>         <dbl>
## 1     0.698         0.586         0.414         0.302
```

```
efficiency
```

```
## [1] 0.6630435
```

Findings:

- At a given instance in time, someone who has hypertension is 0.42 times as likely to die as someone without hypertension adjusting for age.
- Probability of survival after 150 days for those younger than 70 is 77%.
- Probability of survival after 200 days for those younger than 70 is 70%.

- 24% probability of survival after $t=130$ days for patients older than 79, that have less than or equal to 1.8 in serum creatine, and an ejection fraction over 25.
- For those diabetic, platelets seem to reduce as age increases.
- On average, creatinine_phosphokinase is higher for non-smokers.
- Men, on average, have higher creatinine_phosphokinase .
- Women, on average, have a higher platelets count.
- age , ejection fraction , the presence of hypertension , and a value of serum creatinine greater than 1.25 are the variables that contribute most to an accurate prediction of mortality.
- age , creatinine_phosphokinase , ejection_fraction , serum_creatinine , and the presence of hypertension are what most impact the survival rate probability.
- anemia , smoking , sex status, and diabetes , and are the fields that contribute the least to survival rates, in that order.