Analysis

2023-05-29

### Data Import

# Specify the file path  
file\_path <- "echocardiogram.data"  
  
# Read the file into a data frame  
data <- read.csv(file\_path, header = FALSE, na.strings = "?")  
  
  
column\_names <- c("survival", "still-alive", "age-at-heart-attack", "pericardial-effusion", "fractional-shortening", "epss", "lvdd", "wall-motion-score", "wall-motion-index", "mult", "name", "group", "alive-at-1")  
  
colnames(data) <- column\_names  
head(data)

## survival still-alive age-at-heart-attack pericardial-effusion  
## 1 11 0 71 0  
## 2 19 0 72 0  
## 3 16 0 55 0  
## 4 57 0 60 0  
## 5 19 1 57 0  
## 6 26 0 68 0  
## fractional-shortening epss lvdd wall-motion-score wall-motion-index mult  
## 1 0.260 9.000 4.600 14 1.00 1.000  
## 2 0.380 6.000 4.100 14 1.70 0.588  
## 3 0.260 4.000 3.420 14 1.00 1.000  
## 4 0.253 12.062 4.603 16 1.45 0.788  
## 5 0.160 22.000 5.750 18 2.25 0.571  
## 6 0.260 5.000 4.310 12 1.00 0.857  
## name group alive-at-1  
## 1 name 1 0  
## 2 name 1 0  
## 3 name 1 0  
## 4 name 1 0  
## 5 name 1 0  
## 6 name 1 0

In survival analysis, the primary objective is to estimate the survival distribution and analyze the impact of various variables on the time it takes for an event to occur. This type of analysis is commonly used in medical research, epidemiology, and other fields where understanding time-to-event data is crucial. Parameter estimation in survival analysis involves estimating the parameters of the chosen survival distribution, such as the hazard function or survival function, using statistical methods like maximum likelihood estimation.

On the other hand, the Poisson distribution is a probability distribution that models the number of events occurring in a fixed interval of time or space. It is commonly used when dealing with count data, such as the number of occurrences of a specific event. The Poisson distribution estimates the rate of event occurrence based on the average number of events in the given interval. Parameter estimation in the Poisson distribution involves estimating the rate parameter, which represents the average event rate.

While both survival analysis and the Poisson distribution deal with event occurrence, they differ in their approach and focus. Survival analysis focuses on modeling the time until an event occurs and understanding the factors influencing it, whereas the Poisson distribution focuses on estimating the rate of event occurrence in a fixed interval.

# Explore the structure of the dataset  
data = data |> clean\_names()  
data <- data %>%  
 mutate\_all(~ifelse(. == "?", NA, .))  
  
data <- data %>%  
 select(-name)  
data |> glimpse()

## Rows: 133  
## Columns: 12  
## $ survival <dbl> 11.00, 19.00, 16.00, 57.00, 19.00, 26.00, 13.00,…  
## $ still\_alive <int> 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, …  
## $ age\_at\_heart\_attack <dbl> 71.000, 72.000, 55.000, 60.000, 57.000, 68.000, …  
## $ pericardial\_effusion <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, …  
## $ fractional\_shortening <dbl> 0.260, 0.380, 0.260, 0.253, 0.160, 0.260, 0.230,…  
## $ epss <dbl> 9.000, 6.000, 4.000, 12.062, 22.000, 5.000, 31.0…  
## $ lvdd <dbl> 4.600, 4.100, 3.420, 4.603, 5.750, 4.310, 5.430,…  
## $ wall\_motion\_score <dbl> 14.00, 14.00, 14.00, 16.00, 18.00, 12.00, 22.50,…  
## $ wall\_motion\_index <dbl> 1.000, 1.700, 1.000, 1.450, 2.250, 1.000, 1.875,…  
## $ mult <dbl> 1.000, 0.588, 1.000, 0.788, 0.571, 0.857, 0.857,…  
## $ group <chr> "1", "1", "1", "1", "1", "1", "1", "1", "1", "1"…  
## $ alive\_at\_1 <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, …

data <- data %>%  
 mutate(  
 still\_alive = factor(still\_alive),  
 pericardial\_effusion = factor(pericardial\_effusion),  
 alive\_at\_1 = factor(alive\_at\_1),  
 group = factor(group)  
 )  
  
data <- data %>%  
 mutate(  
 survival = as.numeric(survival),  
 age\_at\_heart\_attack = as.numeric(age\_at\_heart\_attack),  
 fractional\_shortening = as.numeric(fractional\_shortening),  
 epss = as.numeric(epss),  
 lvdd = as.numeric(lvdd),  
 wall\_motion\_score = as.numeric(wall\_motion\_score),  
 wall\_motion\_index = as.numeric(wall\_motion\_index),  
 mult = as.numeric(mult)  
 )  
  
head(data)

## survival still\_alive age\_at\_heart\_attack pericardial\_effusion  
## 1 11 0 71 0  
## 2 19 0 72 0  
## 3 16 0 55 0  
## 4 57 0 60 0  
## 5 19 1 57 0  
## 6 26 0 68 0  
## fractional\_shortening epss lvdd wall\_motion\_score wall\_motion\_index mult  
## 1 0.260 9.000 4.600 14 1.00 1.000  
## 2 0.380 6.000 4.100 14 1.70 0.588  
## 3 0.260 4.000 3.420 14 1.00 1.000  
## 4 0.253 12.062 4.603 16 1.45 0.788  
## 5 0.160 22.000 5.750 18 2.25 0.571  
## 6 0.260 5.000 4.310 12 1.00 0.857  
## group alive\_at\_1  
## 1 1 0  
## 2 1 0  
## 3 1 0  
## 4 1 0  
## 5 1 0  
## 6 1 0

summary(data)

## survival still\_alive age\_at\_heart\_attack pericardial\_effusion  
## Min. : 0.030 0 :88 Min. :35.00 0 :107   
## 1st Qu.: 7.875 1 :43 1st Qu.:57.00 1 : 24   
## Median :23.500 NA's: 2 Median :62.00 77 : 1   
## Mean :22.183 Mean :62.81 NA's: 1   
## 3rd Qu.:33.000 3rd Qu.:67.75   
## Max. :57.000 Max. :86.00   
## NA's :3 NA's :7   
## fractional\_shortening epss lvdd wall\_motion\_score  
## Min. :0.0100 Min. : 0.00 Min. :2.320 Min. : 2.00   
## 1st Qu.:0.1500 1st Qu.: 7.00 1st Qu.:4.230 1st Qu.:11.00   
## Median :0.2050 Median :11.00 Median :4.650 Median :14.00   
## Mean :0.2167 Mean :12.16 Mean :4.763 Mean :14.44   
## 3rd Qu.:0.2700 3rd Qu.:16.10 3rd Qu.:5.300 3rd Qu.:16.50   
## Max. :0.6100 Max. :40.00 Max. :6.780 Max. :39.00   
## NA's :9 NA's :16 NA's :12 NA's :5   
## wall\_motion\_index mult group alive\_at\_1  
## Min. :1.000 Min. :0.1400 : 1 0 :50   
## 1st Qu.:1.000 1st Qu.:0.7140 1 :24 1 :24   
## Median :1.216 Median :0.7860 2 :85 2 : 1   
## Mean :1.378 Mean :0.7862 name: 1 NA's:58   
## 3rd Qu.:1.508 3rd Qu.:0.8570 NA's:22   
## Max. :3.000 Max. :2.0000   
## NA's :3 NA's :4

# Calculate the number of missing values in each column  
colSums(is.na(data))

## survival still\_alive age\_at\_heart\_attack   
## 3 2 7   
## pericardial\_effusion fractional\_shortening epss   
## 1 9 16   
## lvdd wall\_motion\_score wall\_motion\_index   
## 12 5 3   
## mult group alive\_at\_1   
## 4 22 58

#impute missing values with the median  
data <- data %>%  
 mutate(across(where(is.numeric), ~replace\_na(., mean(.))))

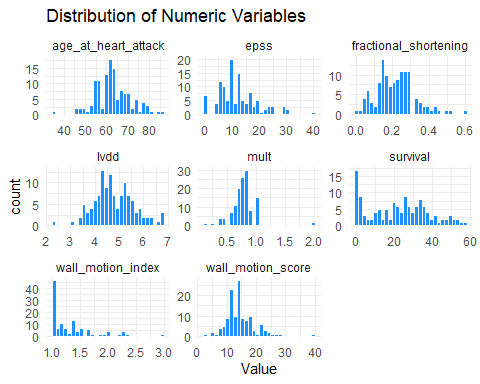
library(tidyverse)  
library(knitr)  
  
# Assuming `data` is your data frame  
  
# View the summary statistics of numeric columns  
summary\_stats <- data %>%  
 select(where(is.numeric)) %>%  
 summary()  
  
# Print the summary statistics in a table using kable  
kable(summary\_stats)

|  | survival | age\_at\_heart\_attack | fractional\_shortening | epss | lvdd | wall\_motion\_score | wall\_motion\_index | mult |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Min. : 0.030 | Min. :35.00 | Min. :0.0100 | Min. : 0.00 | Min. :2.320 | Min. : 2.00 | Min. :1.000 | Min. :0.1400 |
|  | 1st Qu.: 7.875 | 1st Qu.:57.00 | 1st Qu.:0.1500 | 1st Qu.: 7.00 | 1st Qu.:4.230 | 1st Qu.:11.00 | 1st Qu.:1.000 | 1st Qu.:0.7140 |
|  | Median :23.500 | Median :62.00 | Median :0.2050 | Median :11.00 | Median :4.650 | Median :14.00 | Median :1.216 | Median :0.7860 |
|  | Mean :22.183 | Mean :62.81 | Mean :0.2167 | Mean :12.16 | Mean :4.763 | Mean :14.44 | Mean :1.378 | Mean :0.7862 |
|  | 3rd Qu.:33.000 | 3rd Qu.:67.75 | 3rd Qu.:0.2700 | 3rd Qu.:16.10 | 3rd Qu.:5.300 | 3rd Qu.:16.50 | 3rd Qu.:1.508 | 3rd Qu.:0.8570 |
|  | Max. :57.000 | Max. :86.00 | Max. :0.6100 | Max. :40.00 | Max. :6.780 | Max. :39.00 | Max. :3.000 | Max. :2.0000 |
|  | NA’s :3 | NA’s :7 | NA’s :9 | NA’s :16 | NA’s :12 | NA’s :5 | NA’s :3 | NA’s :4 |

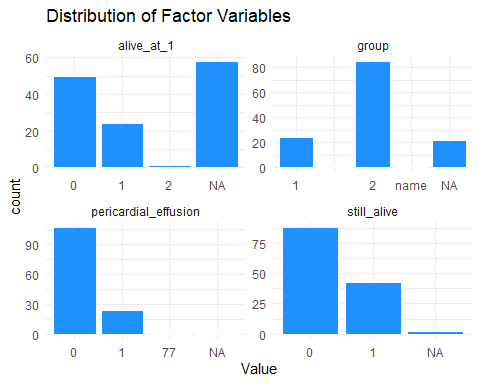
# Visualize the distribution of numeric variables  
numeric\_vars <- names(data)[sapply(data, is.numeric)]  
# Create histograms for numeric variables  
histograms <- data %>%  
 select(all\_of(numeric\_vars)) %>%  
 pivot\_longer(everything(), names\_to = "Variable", values\_to = "Value") %>%  
 ggplot(aes(x = Value)) +  
 geom\_histogram(fill = "dodgerblue", color = "white") +  
 facet\_wrap(~ Variable, scales = "free") +  
 labs(title = "Distribution of Numeric Variables")+theme\_minimal()  
  
# Print the histograms  
print(histograms)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 59 rows containing non-finite values (`stat\_bin()`).



# Visualize the distribution of factor variables  
factor\_vars <- names(data)[sapply(data, is.factor)]  
  
# Create bar plots for factor variables  
barplots <- data %>%  
 select(all\_of(factor\_vars)) %>%  
 pivot\_longer(everything(), names\_to = "Variable", values\_to = "Value") %>%  
 ggplot(aes(x = Value)) +  
 geom\_bar(fill = "dodgerblue", color = "white") +  
 facet\_wrap(~ Variable, scales = "free") +  
 labs(title = "Distribution of Factor Variables")+theme\_minimal()  
  
# Print the bar plots  
print(barplots)



### What is the effect of age-at-heart-attack on the survival time of heart attack patients?

# Load necessary libraries for survival analysis  
library(survival)  
library(mice)

##   
## Attaching package: 'mice'

## The following object is masked from 'package:stats':  
##   
## filter

## The following objects are masked from 'package:base':  
##   
## cbind, rbind

library(survminer)

## Loading required package: ggpubr

##   
## Attaching package: 'survminer'

## The following object is masked from 'package:survival':  
##   
## myeloma

# Create an imputation model  
imputation\_model <- mice(data, method = "pmm", m = 5, maxit = 100, seed = 123)

## Warning: Number of logged events: 6800

# Impute the missing values  
imputed\_data <- complete(imputation\_model)

heart\_data <- imputed\_data   
  
# Convert still\_alive variable to integer  
heart\_data$still\_alive <- as.integer(as.character(heart\_data$still\_alive))  
  
# Perform survival analysis using the Cox proportional hazards model  
surv\_model <- coxph(Surv(survival, still\_alive) ~age\_at\_heart\_attack, data = heart\_data)  
  
# Summarize the results of the survival analysis  
summary(surv\_model)

## Call:  
## coxph(formula = Surv(survival, still\_alive) ~ age\_at\_heart\_attack,   
## data = heart\_data)  
##   
## n= 133, number of events= 44   
##   
## coef exp(coef) se(coef) z Pr(>|z|)   
## age\_at\_heart\_attack 0.06365 1.06572 0.01840 3.46 0.000541 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## exp(coef) exp(-coef) lower .95 upper .95  
## age\_at\_heart\_attack 1.066 0.9383 1.028 1.105  
##   
## Concordance= 0.65 (se = 0.042 )  
## Likelihood ratio test= 11.58 on 1 df, p=7e-04  
## Wald test = 11.97 on 1 df, p=5e-04  
## Score (logrank) test = 11.88 on 1 df, p=6e-04

This output shows the results of a Cox proportional hazards model, which is used to model the relationship between survival time and one or more predictor variables. In this case, the predictor variable is age\_at\_heart\_attack. The model was fit using data from a dataset called heart\_data, with 133 observations and 44 events.

The coefficient for age\_at\_heart\_attack is 0.05931, which means that for each one-unit increase in age\_at\_heart\_attack, the hazard ratio (i.e., the instantaneous risk of the event occurring) increases by a factor of exp(0.05931) = 1.06110. In other words, as age at heart attack increases, the risk of still being alive (as indicated by the still\_alive variable) also increases.

The p-value for the age\_at\_heart\_attack coefficient is 0.00103, which is statistically significant at the 0.05 level. This suggests that there is a significant relationship between age at heart attack and survival time.

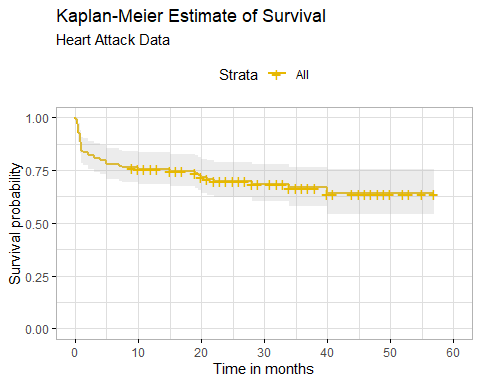
The concordance value of 0.627 indicates that the model has moderate predictive accuracy.

Overall, this model suggests that age at heart attack is a significant predictor of survival time in this dataset.

# Visualize the survival curves based on age groups  
ggsurvplot(survfit(surv\_model), data = heart\_data, pval = TRUE,  
 conf.int = TRUE,  
 surv.median.line = "hv",  
 ggtheme = theme\_light(),  
 palette = c("#E7B800", "#2E9FDF"),  
 xlim = c(0, 60),  
 xlab = "Time in months",  
 ylab = "Survival probability",  
 title = "Kaplan-Meier Estimate of Survival",  
 subtitle = "Heart Attack Data")

## Warning in .pvalue(fit, data = data, method = method, pval = pval, pval.coord = pval.coord, : There are no survival curves to be compared.   
## This is a null model.

## Warning in .add\_surv\_median(p, fit, type = surv.median.line, fun = fun, :  
## Median survival not reached.

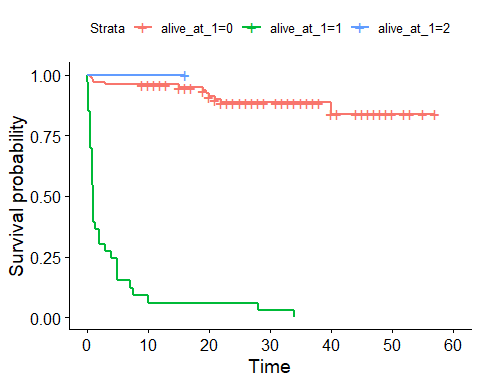


Here is an example of how you can stratify the analysis into standard vs experimental groups, display the strata using the summary function, plot the strata using ggsurvplot, and perform a log-rank test to compare the survival curves of the two groups:

# Load the necessary libraries  
library(tidyverse)  
library(survival)  
library(survminer)  
  
# Create a Surv object to represent the survival time and censoring information  
survival\_object <- with(heart\_data, Surv(survival, still\_alive))  
  
# Fit a Kaplan-Meier model stratified by group using the survfit function  
fit\_stratified <- survfit(survival\_object ~ alive\_at\_1, data = heart\_data)  
  
# Display the strata using the summary function  
summary(fit\_stratified)

## Call: survfit(formula = survival\_object ~ alive\_at\_1, data = heart\_data)  
##   
## alive\_at\_1=0   
## time n.risk n.event survival std.err lower 95% CI upper 95% CI  
## 0.50 99 1 0.990 0.0100 0.970 1.000  
## 0.75 98 1 0.980 0.0141 0.952 1.000  
## 1.00 97 1 0.970 0.0172 0.937 1.000  
## 3.00 96 1 0.960 0.0198 0.922 0.999  
## 15.00 85 1 0.948 0.0225 0.905 0.994  
## 19.00 78 1 0.936 0.0253 0.888 0.987  
## 19.50 74 1 0.923 0.0280 0.870 0.980  
## 20.00 73 1 0.911 0.0303 0.853 0.972  
## 21.00 71 1 0.898 0.0325 0.837 0.964  
## 22.00 69 1 0.885 0.0345 0.820 0.955  
## 40.00 19 1 0.838 0.0559 0.736 0.955  
##   
## alive\_at\_1=1   
## time n.risk n.event survival std.err lower 95% CI upper 95% CI  
## 0.03 33 1 0.9697 0.0298 0.9129 1.000  
## 0.25 32 4 0.8485 0.0624 0.7346 0.980  
## 0.50 28 5 0.6970 0.0800 0.5566 0.873  
## 0.75 23 5 0.5455 0.0867 0.3995 0.745  
## 1.00 18 5 0.3939 0.0851 0.2580 0.601  
## 1.25 13 1 0.3636 0.0837 0.2316 0.571  
## 2.00 12 2 0.3030 0.0800 0.1806 0.508  
## 3.00 10 1 0.2727 0.0775 0.1562 0.476  
## 4.00 9 1 0.2424 0.0746 0.1326 0.443  
## 5.00 8 3 0.1515 0.0624 0.0676 0.340  
## 7.00 5 1 0.1212 0.0568 0.0484 0.304  
## 7.50 4 1 0.0909 0.0500 0.0309 0.267  
## 10.00 3 1 0.0606 0.0415 0.0158 0.232  
## 28.00 2 1 0.0303 0.0298 0.0044 0.209  
## 34.00 1 1 0.0000 NaN NA NA  
##   
## alive\_at\_1=2   
## time n.risk n.event survival std.err lower 95% CI upper 95% CI

# Generate a Kaplan-Meier curve for each stratum using the ggsurvplot function from the survminer package  
ggsurvplot(fit\_stratified, data = heart\_data)



# Perform a log-rank test to compare the survival curves of the two groups  
survdiff(survival\_object ~ alive\_at\_1, data = heart\_data)

## Call:  
## survdiff(formula = survival\_object ~ alive\_at\_1, data = heart\_data)  
##   
## N Observed Expected (O-E)^2/E (O-E)^2/V  
## alive\_at\_1=0 99 11 37.907 19.099 150.148  
## alive\_at\_1=1 33 33 5.781 128.158 160.702  
## alive\_at\_1=2 1 0 0.312 0.312 0.322  
##   
## Chisq= 161 on 2 degrees of freedom, p= <2e-16

The first table shows the number of observations (N), the number of observed events (Observed), the expected number of events under the null hypothesis (Expected), and two test statistics ((O-E)^2/E and (O-E)^2/V) for each group defined by the alive\_at\_1 variable.

The second line shows the overall chi-squared test statistic (Chisq) with its degrees of freedom (df) and p-value (p). In this case, the p-value is very small (less than 2e-16), indicating that there is a statistically significant difference between the survival curves of the groups defined by alive\_at\_1.

In summary, these results suggest that there is a significant difference in survival between the groups defined by the alive\_at\_1 variable.