# Topic: Survival Analytics

**Instructions:**

Please share your answers filled in-line in the word document. Submit code separately wherever applicable.

Please ensure you update all the details:

**Name: Anandakrishnan k v**

**Batch Id:** 19042021

**Topic: Survival Analytics**

**Grading Guidelines:**

**1. An assignment submission is considered complete only when correct and executable code(s) are submitted along with the documentation explaining the method and results. Failing to submit either of those will be considered an invalid submission and will not be considered for evaluation.**

**2. Assignments submitted after the deadline will affect your grades.**

**Grading:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ans** | **Date** |  |  | **Ans** | **Date** |
| Correct | On time | A | 100 |  |  |
| 80% & above | On time | B | 85 | Correct | Late |
| 50% & above | On time | C | 75 | 80% & above | Late |
| 50% & below | On time | D | 65 | 50% & above | Late |
|  |  | E | 55 | 50% & below |  |
| Copied/No Submission |  | F | 45 |  |  |

* **Grade A: (>= 90):** When all assignments are submitted on or before the given deadline.
* **Grade B: (>= 80 and < 90):** 
  + When assignments are submitted on time but less than 80% of problems are completed.

(OR)

* + All assignments are submitted after the deadline.
* **Grade C: (>= 70 and < 80):** 
  + When assignments are submitted on time but less than 50% of the problems are completed.

(OR)

* + Less than 80% of problems in the assignments are submitted after the deadline.
* **Grade D: (>= 60 and < 70):**
  + Assignments submitted after the deadline and with 50% or less problems.
* **Grade E: (>= 50 and < 60):** 
  + Less than 30% of problems in the assignments are submitted after the deadline.

(OR)

* + Less than 30% of problems in the assignments are submitted before the deadline.
* **Grade F: (< 50):** No submission (or) malpractice.

**Hints:**

1. **Business Problem**
   1. **What is the business objective?**
   2. **Are there any constraints?**
2. **Work on each feature of the dataset to create a data dictionary as displayed in the below image:**



**2.1 Make a table as shown above and provide information about the features such as its Data type and its relevance to the model building, if not relevant provide reasons and provide description of the feature.**

1. **Exploratory Data Analysis (EDA):**
   1. **Summary.**
   2. **Univariate analysis.**
   3. **Bivariate analysis.**
2. **Model Building**

**4.1 Build the model on the scaled data (try multiple options).**

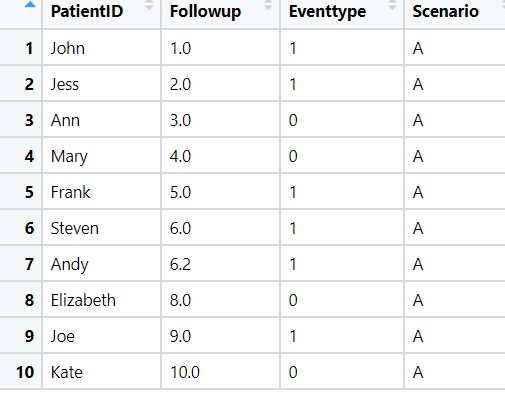
**4.2 Perform survival analytics on the given datasets.**

**4.3 Briefly explain the model output in the documentation.**

1. **Write about the benefits/impact of the solution - in what way does the business (client) benefit from the solution provided?**

**Problem Statement:**

The following dataset contains patient ID, follow up, event type, and scenarios. Build a survival analysis model on the given data.



**Solution:**

**Pyhthon Code:**

# pip install lifelines

# import lifelines

import pandas as pd

import matplotlib.pyplot as plt

# Loading the the survival un-employment data

survival\_ptnt = pd.read\_csv("C://Users//user//Downloads//survival//Patient.csv")

# dropping less informative columns

survival\_ptnt.columns

survival\_ptnt.drop('PatientID',axis=1,inplace= True)

survival\_ptnt.drop('Scenario',axis=1,inplace= True)

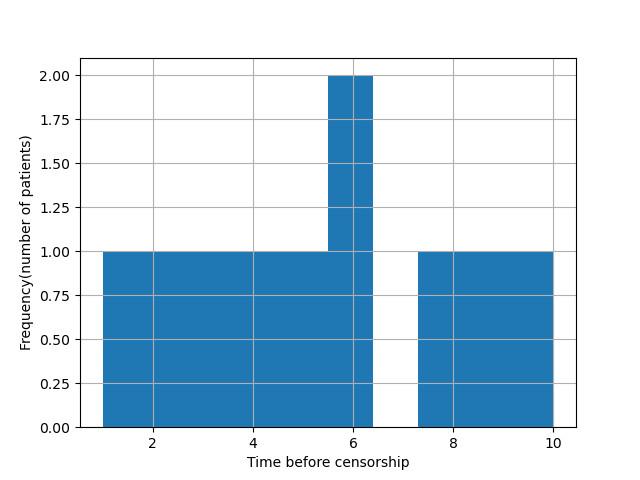
survival\_ptnt.head()

# histogram

survival\_ptnt.Followup.hist();

plt.xlabel('Time before censorship');

plt.ylabel('Frequency(number of patients)');



survival\_ptnt.describe()

survival\_ptnt['Followup'].describe()

# 'Followup' is referring to time

T = survival\_ptnt.Followup

# Importing the KaplanMeierFitter model to fit the survival analysis

from lifelines import KaplanMeierFitter

# Initiating the KaplanMeierFitter model

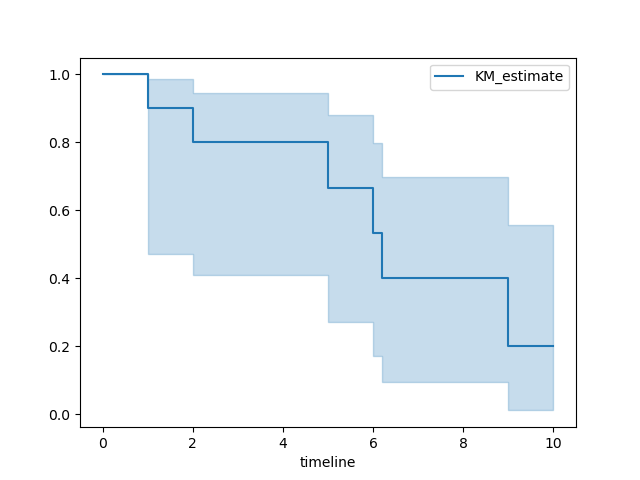
kmf = KaplanMeierFitter()

# Fitting KaplanMeierFitter model on Time and Events for death

kmf.fit(T, event\_observed=survival\_ptnt.Eventtype)

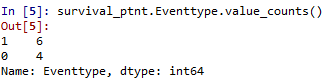
# Time-line estimations plot

kmf.plot()



survival\_ptnt.Eventtype.value\_counts()

# We have 6 non-censored data and 4 censored data.



**Summary:**

* At the end of analysis we have 6 non-censored data and 4 censored data
* Until 5 unit time period there have 80% survival rate, but after that there have happened a sudden fall in the rate
* After 10 unit time period only 20% of total patients got survived

**Problem Statement: -**

ECG of different age groups of people has been recorded. The survival time in hours after the operation is given and the event type is denoted by 1 (if dead) and 0 (if alive). Perform survival analysis on the dataset given below and provide your insights in the documentation.

A large room

Description automatically generated

**Solution:**

**Pyhthon Code:**

**Packages used**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

import statistics

from sklearn.impute import SimpleImputer

# pip install lifelines

# import lifelines

from lifelines import KaplanMeierFitter, CoxPHFitter

from lifelines.statistics import logrank\_test

from scipy import stats

# Data

df = pd.read\_excel("C://Users//user//Downloads//survival//ECG\_Surv.xlsx")

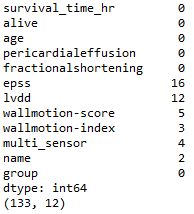
df.head()

df.columns

# Check missing values[¶](https://www.kaggle.com/yukikitayama/survival-analysis#Check-missing-values)

print(df.isnull().sum())

print(df.shape)



**We have 133 observations, but there are many missing values across variables. Gonna implement imputation for missing values with means of each columns.**

# Impute missing values with mean

imp\_mean = SimpleImputer(missing\_values = np.nan, strategy = 'mean')

COLUMNS = ['age', 'pericardialeffusion', 'fractionalshortening', 'epss', 'lvdd', 'wallmotion-score']

X = imp\_mean.fit\_transform(df[COLUMNS])

df\_X = pd.DataFrame(X, columns = COLUMNS)

df\_X.shape

C:\Users\user\Documents\matplot\Untitled.png

COLUMNS\_keep = ['survival\_time\_hr', 'alive']

df\_keep = df[COLUMNS\_keep]

df\_keep.shape

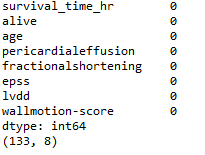
C:\Users\user\Documents\matplot\Untitled.png

df = pd.concat([df\_keep, df\_X], axis = 1)

df = df.dropna() # dropna function applies to survival and alive variables. Not consider imputation for that columns

print(df.isnull().sum())

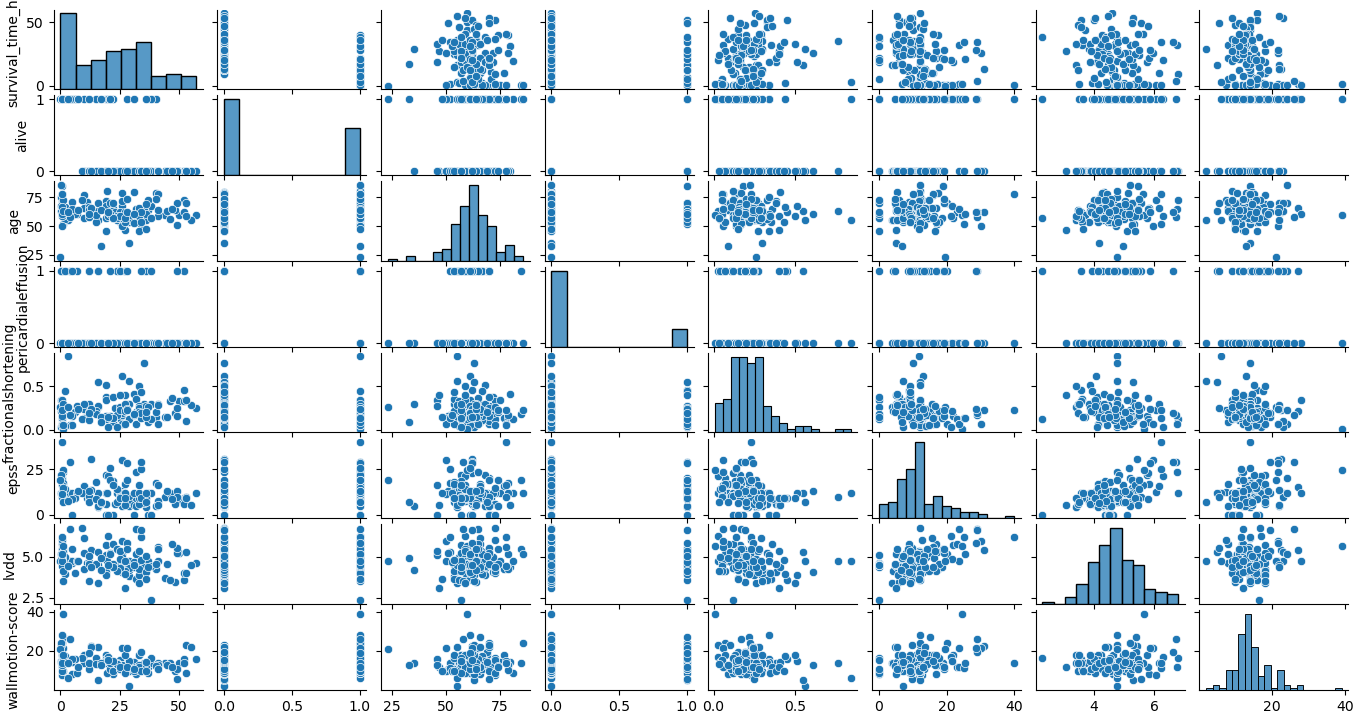
print(df.shape)



**dropna function applies to survival and alive variables which don't consider imputation for. The number of the data for analysis is 130 observations.**

# Scatter plots between survival and covariates

sns.pairplot(df)



# Check censored data

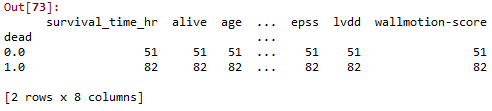
For alive = 1 patients, because they are alive during data collection period and we do not know their survival months after the data collection, they are regarded as censored data. Hence, the following analysis needs to consider the censored data by making dead variable below.

df.loc[df.alive == 1, 'dead'] = 0

df.loc[df.alive == 0, 'dead'] = 1

df.groupby('dead').count()

df.groupby('dead').count()



**We have 82 non-censored data and 51 censored data.**

# Kaplan Meier estimates

kmf = KaplanMeierFitter()

T = df['survival\_time\_hr']

E = df['dead']

kmf.fit(T, event\_observed = E)

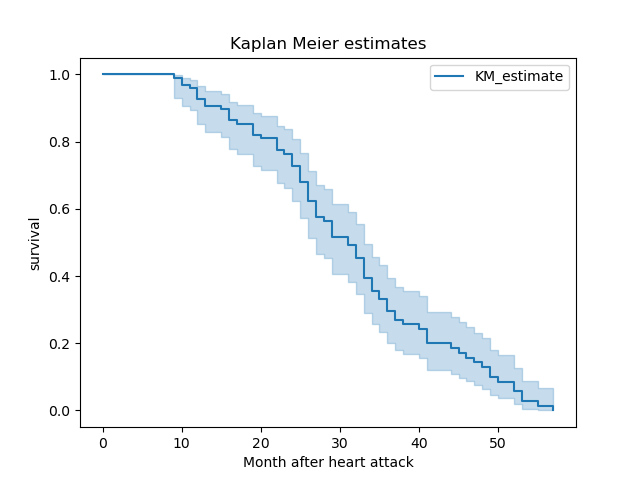
kmf.plot()

plt.title("Kaplan Meier estimates")

plt.xlabel("Month after heart attack")

plt.ylabel("survival")

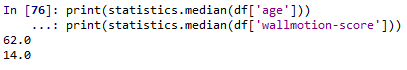
plt.show()



**# slight negative relationship of age and wallmotion-score to survival, so used median to make two groups within each variable to see difference in survival time.**

print(statistics.median(df['age']))

print(statistics.median(df['wallmotion-score']))



age\_group = df['age'] < statistics.median(df['age'])

ax = plt.subplot(111)

kmf.fit(T[age\_group], event\_observed = E[age\_group], label = 'below 62')

kmf.plot(ax = ax)

kmf.fit(T[~age\_group], event\_observed = E[~age\_group], label = 'above 62')

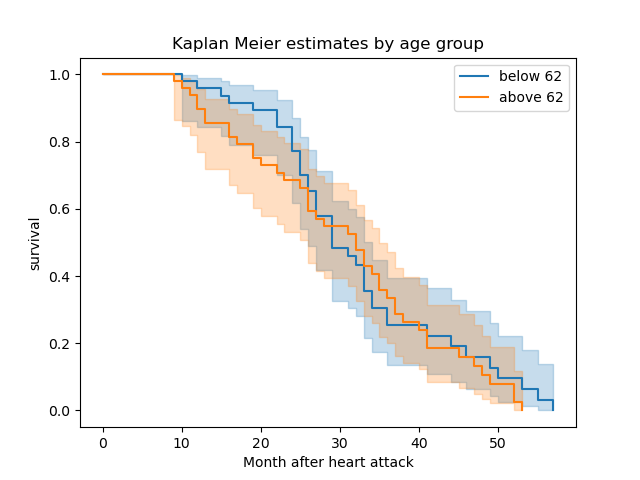
kmf.plot(ax = ax)

plt.title("Kaplan Meier estimates by age group")

plt.xlabel("Month after heart attack")

plt.ylabel("survival")

C:\Users\user\Documents\matplot\Untitled.png



**# The difference by age groups seems to be weak. However, there seems to differ by wallmotion-score group for the first 24 months (2 years) after heart attack. So applied the following analysis based on wallmotion-score group**

# Log-rank test

month\_cut = 24

df.loc[(df.dead == 1) & (df.survival\_time\_hr <= month\_cut), 'censored'] = 1

df.loc[(df.dead == 1) & (df.survival\_time\_hr > month\_cut), 'censored'] = 0

df.loc[df.dead == 0, 'censored'] = 0

E\_v2 = df['censored']

T\_low = T[score\_group]

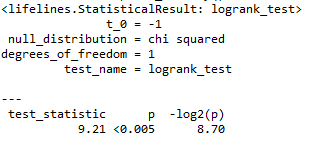
T\_high = T[~score\_group]

E\_low = E\_v2[score\_group]

E\_high = E\_v2[~score\_group]

results = logrank\_test(T\_low, T\_high, event\_observed\_A = E\_low, event\_observed\_B = E\_high)

results.print\_summary()



# "test\_statistic" here is a chi-square statistic. It shows chi-square statistic 9.98, and p-value is less than 5%. Thus confirm that there is a significant difference in suvival time by wallmotion score group for the first 2 year after heart attack.

# 

cph = CoxPHFitter()

df\_score\_group = pd.DataFrame(score\_group)

df\_model = df[['survival\_time\_hr', 'censored', 'age']]

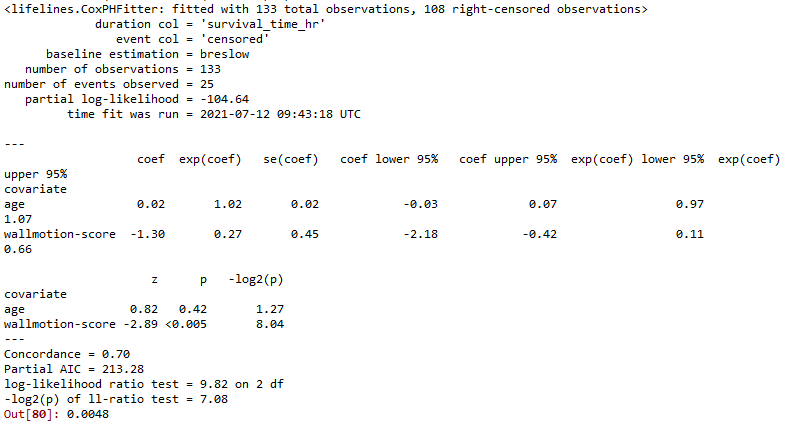
df\_model = pd.concat([df\_model, df\_score\_group], axis = 1)

cph.fit(df\_model, 'survival\_time\_hr', 'censored')

cph.print\_summary()

# p-value of Log-likelihood ratio test

round(stats.chi2.sf(10.68, 2),4)



# p-value of Log-likelihood ratio test

round(stats.chi2.sf(10.68, 2),4)

C:\Users\user\Documents\matplot\Untitled.png

**Summary:**

Whether this model is significant or not depends on the result of Log-likelihood ratio test at the bottom of the summary. This statistic follows Chi-square distribution with 2 degree of freedom, and p-value is 0.0048. It says this cox model is significant so that statistical inference is based on this model. Wallmotion-score group is a risk factor for survival time, but age is not by checking p-values. Negative sign of wallmotion-score variable indicates that the patients with low wallmotion score reduce the risk of death. Hazard ration of wallmotion-score is 0.27, which means it reduce in hazard since it is less than 1 and it reduces the hazard by 73% (1 - 0.23). Thus, I conclude that for the first two years after each patient experiences heart attack, the people with high wallmotion score would have a higher risk of death so that we can pay attention to this group of patients