# 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: Vishvash C Batch Id:** 23012024

**Topic: Survival Analytics**

**Guidelines:**

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

**2. Ensure that you submit your assignments correctly and in full. Resubmission is not allowed.**

**3. Post the submission you can evaluate your work by referring to the keys provided. (will be available only post the submission).**

**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:

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

1. Data Pre-processing

3.1 Data Cleaning, Feature Engineering, etc.

3.2 Outlier Treatment.

1. Exploratory Data Analysis (EDA):
   1. Summary.
   2. Univariate analysis.
   3. Bivariate analysis.
2. Model Building
   1. Build the model on the scaled data (try multiple options).
   2. Perform survival analytics on the given datasets.
   3. Briefly explain the model output in the documentation.
3. Deployment
4. Share the benefits/impact of the solution - how or in what way the business (client) gets to benefit from the solution provided.

**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



**Code:**

# Problem Statement

'''

Problem Statement: ECG of different pericardialeffusion 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.

Objective: Maximize the survival rate.

Constraints: Improve the overall treatmet for the patient.

'''

# pip install lifelines

import pandas as pd

# Loading the the survival un-employment data

survival\_dead = pd.read\_excel(r"C:/Users/Lenovo/Downloads/Study material/Data Science/Survival\_Analysis/Assignment/Survival Analytics/ECG\_Surv.xlsx")

survival\_dead.head()

survival\_dead.describe()

survival\_dead["survival\_time\_hr"].describe()

# survival\_time\_hr is referring to time

T = survival\_dead.survival\_time\_hr

# 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\_dead.alive)

# Time-line estimations plot

kmf.plot()

# Over Multiple groups

# For each group, here group is pericardialeffusion

survival\_dead.pericardialeffusion.value\_counts()

# Applying KaplanMeierFitter model on Time and Events for the group "0"

kmf.fit(T[survival\_dead.pericardialeffusion == 0], survival\_dead.alive[survival\_dead.pericardialeffusion == 0], label = 'Absence of pericardial effusion')

ax = kmf.plot()

# Applying KaplanMeierFitter model on Time and Events for the group "1"

kmf.fit(T[survival\_dead.pericardialeffusion == 1], survival\_dead.alive[survival\_dead.pericardialeffusion == 1], label = 'Presence of pericardial effusion')

kmf.plot(ax=ax)

'''

If there is Presence of pericardial effusion the probability of patient dies increases after the surgery.

Whereas if there is no presence of pericardical effusion the probability of patient dies is comparitively lesser.

'''

**Output:**

survival\_dead.head()

Out[22]:

survival\_time\_hr alive age ... wallmotion-index multi\_sensor group

0 11.0 0 71 ... 1.00 1.000 1

1 19.0 0 72 ... 1.70 0.588 1

2 16.0 0 55 ... 1.00 1.000 1

3 57.0 0 60 ... 1.45 0.788 1

4 19.0 1 57 ... 2.25 0.571 1

[5 rows x 11 columns]

survival\_dead["survival\_time\_hr"].describe()

Out[24]:

count 133.000000

mean 21.795338

std 15.885313

min 0.030000

25% 6.000000

50% 22.000000

75% 33.000000

max 57.000000

Name: survival\_time\_hr, dtype: float64

survival\_dead.pericardialeffusion.value\_counts()

Out[35]:

pericardialeffusion

0 108

1 25

Name: count, dtype: int64

survival\_dead.alive.value\_counts()

Out[36]:

alive

0 82 (alive)

1 51 (dead)

Name: count, dtype: int64



