Sardar Patel University

Vallabh Vidyanagar, Anand – 388120

Department Of Statistics

Project Report

(PSO4CSTA54)

On

Statistical and Survival Analysis of Key Risk Factors in Heart Failure Patients.

PROJECT GUIDE

Prof. (Dr.) Jyoti M. Divecha

Dr. Dharmeh P. Raykundaliya

Dr. Khimya Tinnani

SUBMITTED BY

Harshdeep Krishnat Gharge

M.Sc Statistics

2024-2025

Certificate

We certify that **Mr. Harshdeep Krishnat Gharge**, a student of M.Sc. Statistics, Semester IV (Exam No. 04), has successfully completed his project titled **“Statistical and Survival Analysis of Key Risk Factors in Heart Failure Patient”** as part of the course PS04CSTA54 for the academic term ending March 2025.

His work has been thoroughly reviewed and found to be satisfactory.

PLACE : Vallabh Vidyanagar

Date :

Project Guide	Project Guide
Dr. Dharmesh P. Raykundaliya	**Dr. Khimya S. Tinani**

Project Guide

Prof (Dr.) Jyoti M. Divecha

(Head of Department)

ACKNOWLEDGEMENT

We extend our heartfelt gratitude to all those who contributed to the completion of this MSc Statistics project titled **“Statistical and Survival Analysis of Key Risk Factors in Heart Failure Patient.”**

We are also deeply thankful to **Ms. Dr. Jyoti Divecha**, Head of the Department, for her continuous encouragement and support, which provided us with the conducive environment necessary for conducting this research.

Our gratitude extends to the esteemed faculty members, **Dr. D.P. Raykundaliya, and Dr. Khimya Tinnani**, **Mr. Tanaji Mane** for their valuable inputs, constructive feedback, and scholarly guidance, which enriched our understanding and enhanced the quality of our work.

Additionally, we would like to acknowledge the contributions of all non-teaching staff members whose efforts behind the scenes facilitated the smooth execution of this project.

Last but certainly not least, we express our deepest appreciation to our parents for their unwavering love, encouragement, and support throughout our academic journey. Their belief in our abilities has been a constant source of motivation and strength.

I acknowledge with gratitude the collective support and encouragement of all those mentioned above, without whom this project would not have been possible.

Harshdeep Krishnat Gharge

INDEX

ABSTRACT

Heart failure is a critical cardiovascular condition that significantly contributes to global mortality, affecting millions of individuals each year. This project aims to predict DEATH_EVENT in heart failure patients using clinical data from the UCI Machine Learning Repository. The dataset comprises 299 patient records with various medical attributes, including age, ejection fraction, creatinine levels, blood pressure, and comorbid conditions such as diabetes and anemia. Understanding these factors can help in early diagnosis and better treatment planning for heart failure patients.

To achieve these objectives, we employ various statistical techniques, including chi-square tests, Kaplan-Meier survival analysis, and the log-rank test for group comparisons. A binary logistic regression model is used to determine significant predictors of mortality, while the Cox proportional hazards model evaluates the effect of independent variables on patient survival over time. The necessary computations and modeling are carried out using R programming, utilizing libraries such as survival, ggplot2, caret, and visualization.

The findings of this project can help healthcare professionals make data-driven decisions for managing heart failure patients. By identifying critical risk factors, the study provides insights into improving patient monitoring, prioritizing high-risk individuals, and optimizing treatment strategies. The integration of statistical methods enhances predictive accuracy, contributing to better clinical decision-making and potentially reducing heart failure-related deaths.

OBJECTIVES

1. **Assess statistical association between (Smoking status, Anemia, Diabetes, Sex and High blood pressure) and DEATH_EVENT**.

2. **To cluster age groups and assess their association with patient death events for a better understanding of death events patterns.**

3. **To evaluate the impact of independent variables on survival or Death Event outcomes using survival analysis methods.(log-Rank Test)**

4. **To identify which clinical factors significantly affect patient death event time using COX PROPORTIONAL HAZARDS MODEL.**

DATA SHEET

The dataset has been taken from **University of California, Irvine (UCI)** containing 399 observations and 13 features.

Sr No	age	anaemia	creatinine_phosphokinase	diabetes	ejection_fraction	high_blood_pressure
1.	75	No	582	No	20	Yes
2.	55	No	7861	No	38	No
3.	65	No	146	No	20	No
4.	50	Yes	111	No	20	No
5.	65	Yes	160	Yes	20	No
6.	90	Yes	47	No	40	Yes
7.	75	Yes	246	No	15	No
8.	60	Yes	315	Yes	60	No
9.	65	No	157	No	65	No
10.	80	Yes	123	No	35	Yes
11.	75	Yes	81	No	38	Yes
12.	62	No	231	No	25	Yes
13.	45	Yes	981	No	30	No
14.	50	Yes	168	No	38	Yes
15.	49	Yes	80	No	30	Yes
16.	82	Yes	379	No	50	No
17.	87	Yes	149	No	38	No
18.	45	No	582	No	14	No
19.	70	Yes	125	No	25	Yes
20.	48	Yes	582	Yes	55	No

Sr No	platelets	serum_creatinine	serum_sodium	sex	smoking	time	DEATH_EVENT
1.	265000	1.9	130	Male	Non_Smoker	4	Death
2.	263358	1.1	136	Male	Non_Smoker	6	Death
3.	162000	1.3	129	Male	Smoker	7	Death
4.	210000	1.9	137	Male	Non_Smoker	7	Death
5.	327000	2.7	116	Female	Non_Smoker	8	Death
6.	204000	2.1	132	Male	Smoker	8	Death
7.	127000	1.2	137	Male	Non_Smoker	10	Death
8.	454000	1.1	131	Male	Smoker	10	Death
9.	263358	1.5	138	Female	Non_Smoker	10	Death
10.	388000	9.4	133	Male	Smoker	10	Death
11.	368000	4	131	Male	Smoker	10	Death
12.	253000	0.9	140	Male	Smoker	10	Death
13.	136000	1.1	137	Male	Non_Smoker	11	Death
14.	276000	1.1	137	Male	Non_Smoker	11	Death
15.	427000	1	138	Female	Non_Smoker	12	Alive
16.	47000	1.3	136	Male	Non_Smoker	13	Death
17.	262000	0.9	140	Male	Non_Smoker	14	Death
18.	166000	0.8	127	Male	Non_Smoker	14	Death
19.	237000	1	140	Female	Non_Smoker	15	Death
20.	87000	1.9	121	Female	Non_Smoker	15	Death

INTRODUCTION

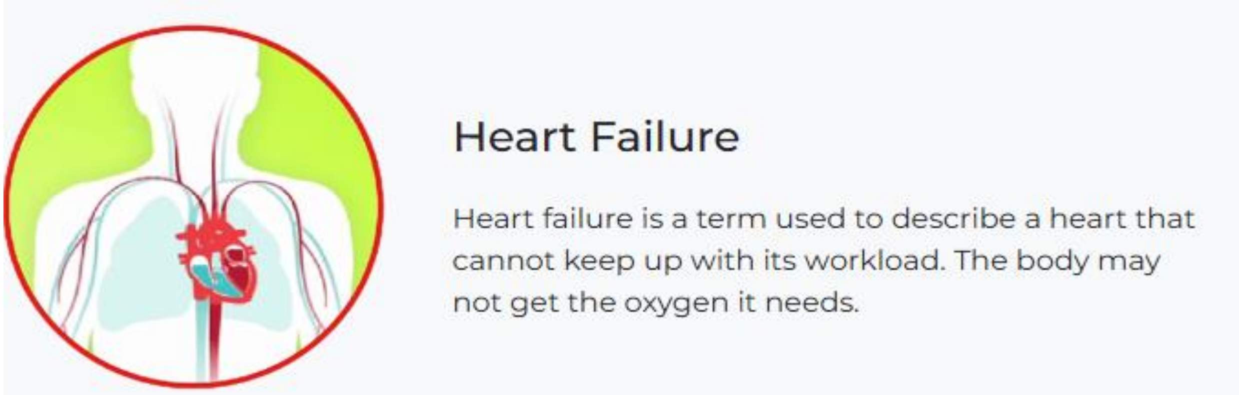
Heart failure is a life-threatening cardiovascular condition that occurs when the heart is unable to pump sufficient blood to meet the body's needs. It is one of the leading causes of mortality worldwide, contributing to approximately 17 million deaths annually. Identifying the key factors that influence survival in heart failure patients is crucial for improving diagnosis, treatment, and patient management. This project focuses on predicting mortality in heart failure patients using clinical data and evaluating the significance of various medical risk factors.

The dataset used in this study is sourced from the UCI Machine Learning Repository and consists of 299 patient records collected in 2015. It includes variables such as age, ejection fraction, creatinine levels, blood pressure, smoking status, diabetes, anemia, and other clinical parameters. By analyzing this dataset, we aim to identify the most critical factors affecting patient survival and assess their impact using statistical techniques.

To achieve this, we employ various statistical techniques, including chi-square tests for association analysis, Kaplan-Meier survival curves for visualizing survival trends, and the log-rank test to compare survival distributions. Additionally, we utilize binary logistic regression to determine significant predictors of mortality and the Cox proportional hazards model to evaluate the impact of risk factors over time. These analyses are conducted using R programming, leveraging libraries such as survival, ggplot2, caret for computation and visualization.

❖ Heart failure

Heart failure is a chronic condition that occurs when the heart can't pump enough blood to meet the body's needs. It can affect one or both sides of the heart.



Heart Failure

Heart failure is a term used to describe a heart that cannot keep up with its workload. The body may not get the oxygen it needs.

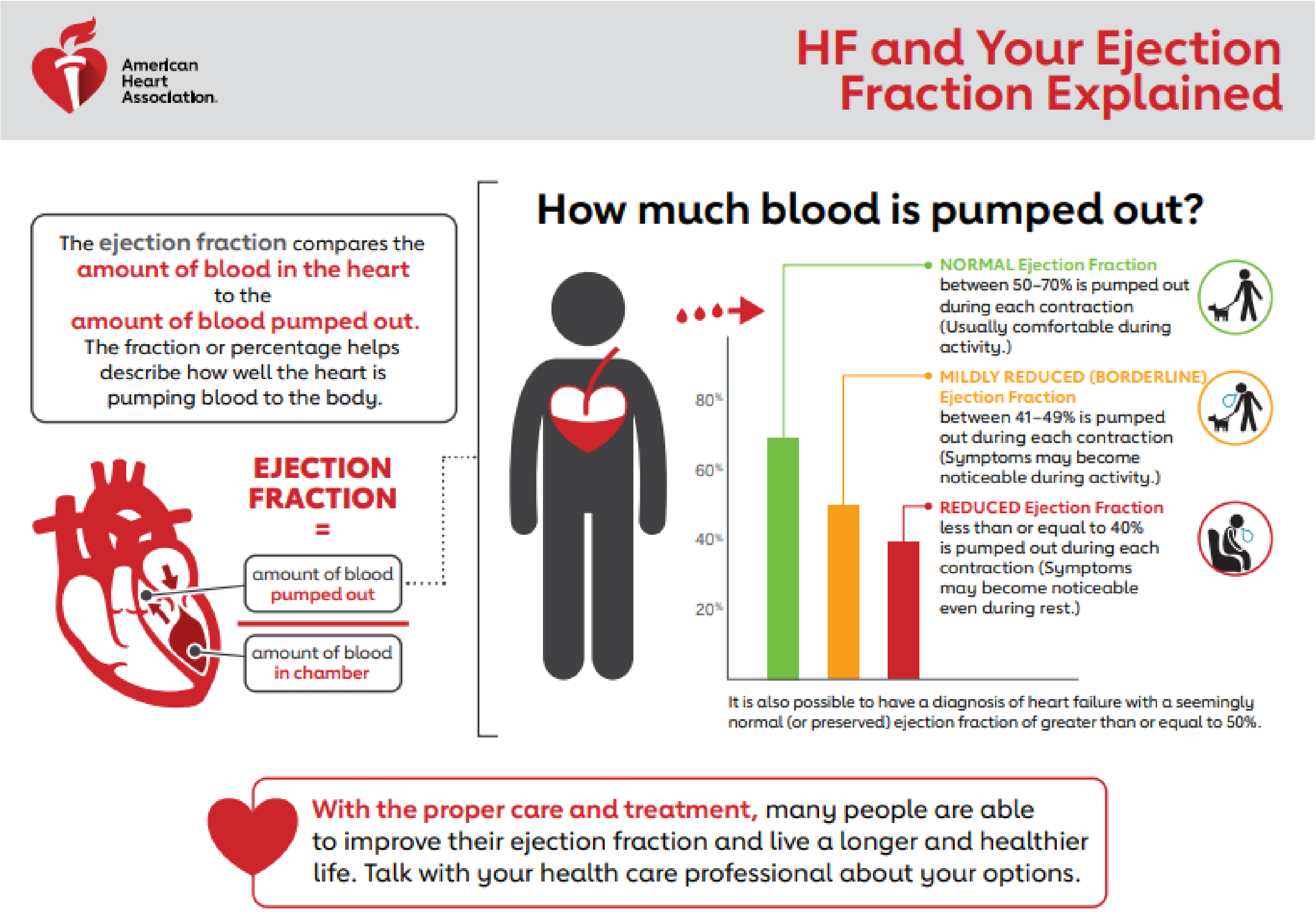
The term heart failure sounds like the heart is no longer working at all. Actually, heart failure, sometimes called HF, means that the heart isn’t pumping as well as it should. Congestive heart failure is a type of heart failure that requires timely medical attention, although sometimes the two terms are used interchangeably.

Symptoms

- Shortness of breath
- Weight gain
- Nausea and lack of appetite
- Chest pain
- Fatigue and weakness
- Swelling in the legs, ankles, and feet
- Rapid or irregular heartbeat

What is “ejection fraction”?

Ejection fraction (EF) is a measurement, expressed as a percentage, of how much blood the left ventricle pumps out with each contraction. An ejection fraction of 60 percent means that 60 percent of the total amount of blood in the left ventricle is pushed out with each heartbeat. A normal heart’s ejection fraction is between 55 and 70 percent.



American Heart Association.

HF and Your Ejection Fraction Explained

The **ejection fraction** compares the **amount of blood in the heart** to the **amount of blood pumped out.** The fraction or percentage helps describe how well the heart is pumping blood to the body.

EJECTION FRACTION = amount of blood **pumped out** / amount of blood **in chamber**

How much blood is pumped out?

NORMAL Ejection Fraction between 50–70% is pumped out during each contraction (Usually comfortable during activity.)

MILDLY REDUCED (BORDERLINE) Ejection Fraction between 41–49% is pumped out during each contraction (Symptoms may become noticeable during activity.)

REDUCED Ejection Fraction less than or equal to 40% is pumped out during each contraction (Symptoms may become noticeable even during rest.)

It is also possible to have a diagnosis of heart failure with a seemingly normal (or preserved) ejection fraction of greater than or equal to 50%.

With the proper care and treatment, many people are able to improve their ejection fraction and live a longer and healthier life. Talk with your health care professional about your options.

IMFORMATION ABOUT PARAMETERS

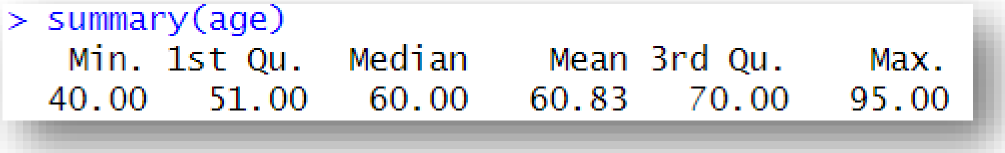
Sr.No	Variable	Description	Type
1	age	Age of the patient (in years).	Numeric
2	anaemia	Indicating whether the patient has anemia	Categorical (Binary)
		(1 = Yes, 0 = No).	
3	creatinine_phosphokinase	Level of creatinine phosphokinase (CPK) enzyme in the blood (mcg/L)	Numeric
4	diabetes	Indicating Whether the patient has diabetes	Categorical (Binary)
		(1 = Yes, 0 = No).	
5	ejection_fraction	Percentage of blood leaving the heart each time it contracts.	Numeric
6	high_blood_pressure	Indicating Whether the patient has high blood pressure	Categorical (Binary)
		(1 = Yes, 0 = No).	
7	platelets	Platelet count in the blood (kiloplatelets/mL).	Numeric
8	serum_creatinine	Level of creatinine in the blood (mg/dL).	Continuous
9..	serum_sodium	Level of sodium in the blood (mEq/L).	Numeric
10..	sex	Gender of the patient (1 = Male, 0 = Female).	Categorical (Binary)

11	smoking	Whether the patient smokes (1 = Smoker, 0 = Non_smoker).	Categorical (Binary)
12	time	Follow-up period (in days).	Numeric
13	DEATH_EVENT	Target variable: whether the patient died during follow-up (1 = Death, 0 = Alive)	Categorical (Binary)

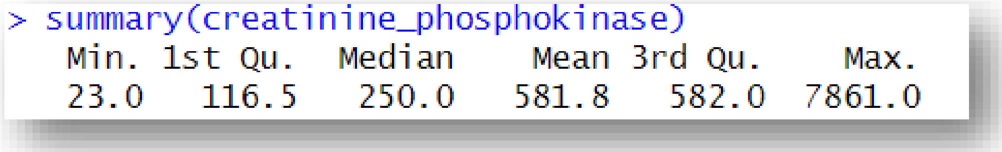
DATA STRUCTURE

```
str(data)
'data.frame':                 299 obs. of  13 variables:
 $ age                      : num  75 55 65 50 65 90 75 60 65 80 ...
 $ anaemia                  : chr  "No" "No" "No" "Yes" ...
 $ creatinine_phosphokinase : int  582 7861 146 111 160 47 246 315 15
 $ diabetes                 : chr  "No" "No" "No" "No" ...
 $ ejection_fraction        : int  20 38 20 20 20 40 15 60 65 35 ...
 $ high_blood_pressure      : chr  "Yes" "No" "No" "No" ...
 $ platelets                : num  265000 263358 162000 210000
 $ serum_creatinine         : num  1.9 1.1 1.3 1.9 2.7 2.1 1.2 1.1 1.5 9.
 $ serum_sodium             : int  130 136 129 137 116 132 137 131 13
 $ sex                       : chr  "Male" "Male" "Male" "Male" ...
 $ smoking                  : chr  "Non_Smoker" "Non_Smoker"
 $ time                     : int  4 6 7 7 8 8 10 10 10 10 ...
 $ DEATH_EVENT              : int  1 1 1 1 1 1 1 1 1 1 ...
```

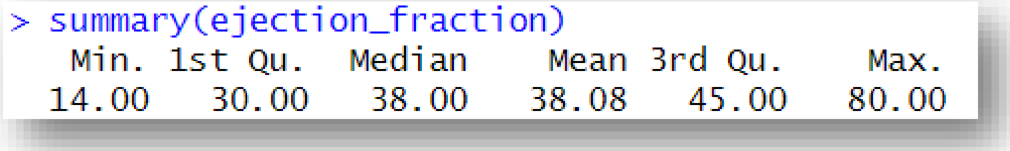
DESCRIPTIVE STATISTICS



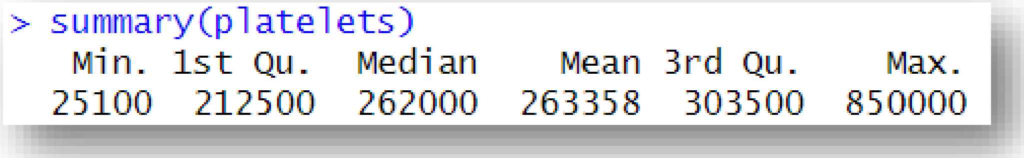
```
> summary(age)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  40.00   51.00   60.00   60.83   70.00   95.00 
```



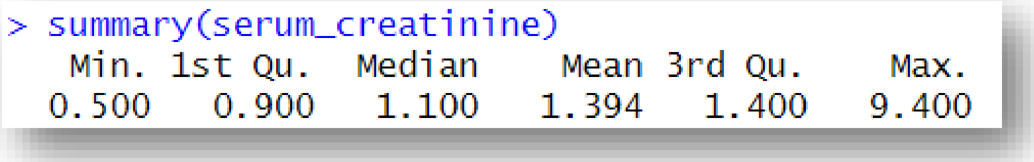
```
> summary(creatinine_phosphokinase)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
   23.0   116.5   250.0   581.8   582.0  7861.0 
```



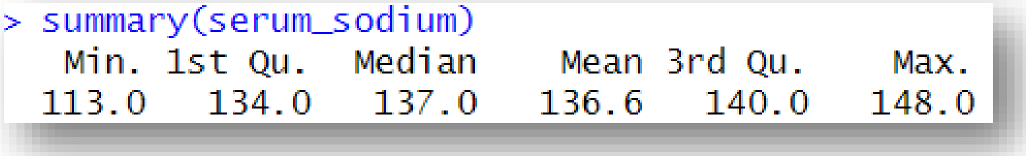
```
> summary(ejection_fraction)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  14.00   30.00   38.00   38.08   45.00   80.00 
```



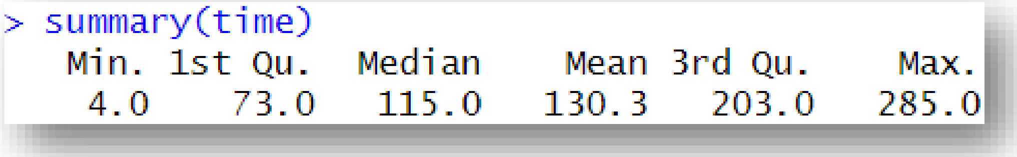
```
> summary(platelets)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  25100  212500  262000  263358  303500  850000 
```



```
> summary(serum_creatinine)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  0.500   0.900   1.100   1.394   1.400   9.400 
```



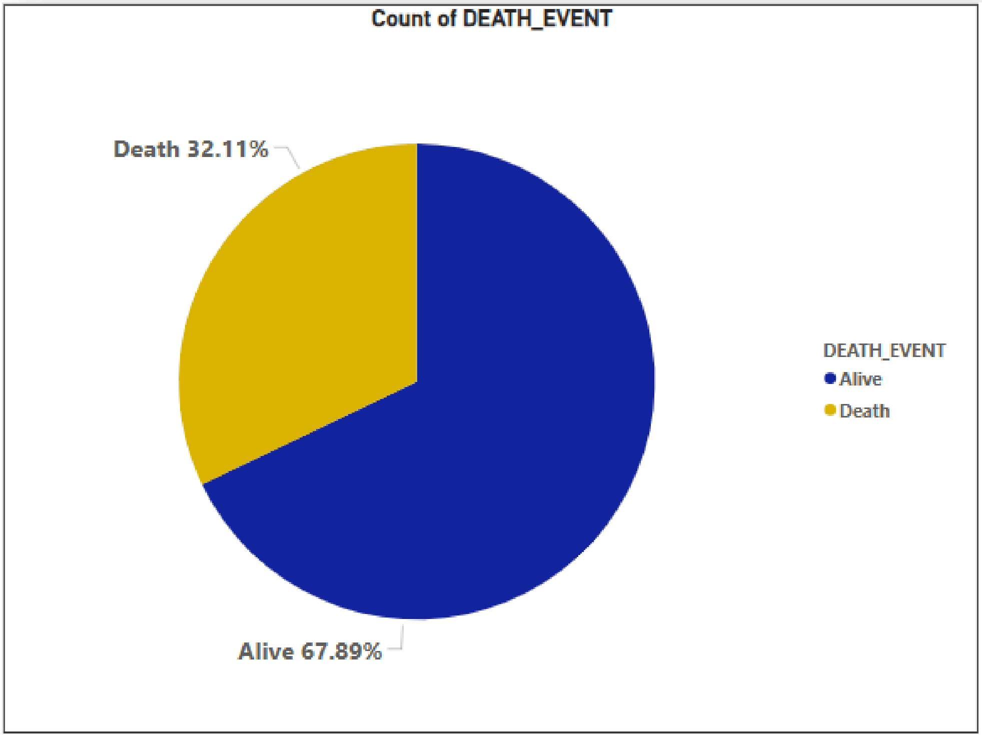
```
> summary(serum_sodium)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  113.0   134.0   137.0   136.6   140.0   148.0 
```



```
> summary(time)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
    4.0    73.0   115.0   130.3   203.0   285.0 
```

Data Visualization

Count of Death event

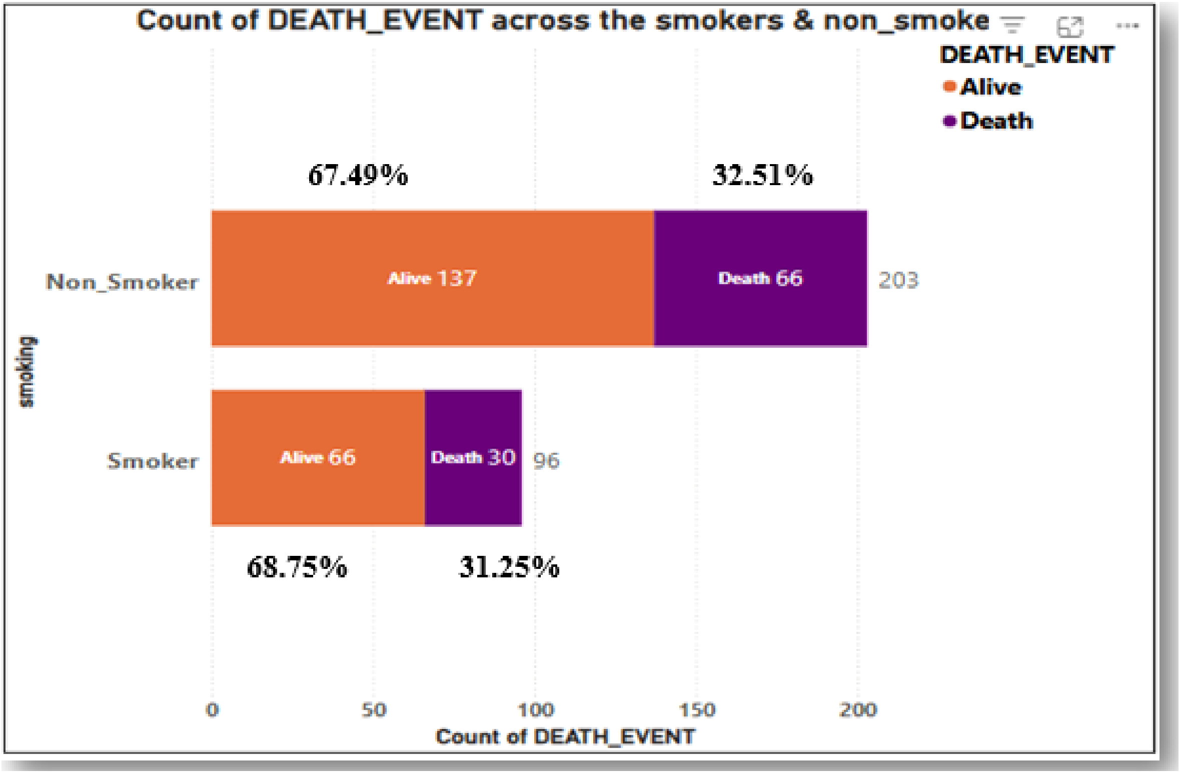


✓ Interpretation:

The pie chart represents the distribution of DEATH_EVENT, which categorizes patients into two groups: Alive and Death.

- The blue portion (67.89%) represents patients who survived.
- The yellow portion (32.11%) represents patients who did not survive.
- The chart shows that 67.89% of patients survived, while 32.11% died, indicating a significant DEATH_EVENT

Comparision Death event across Smoking Status

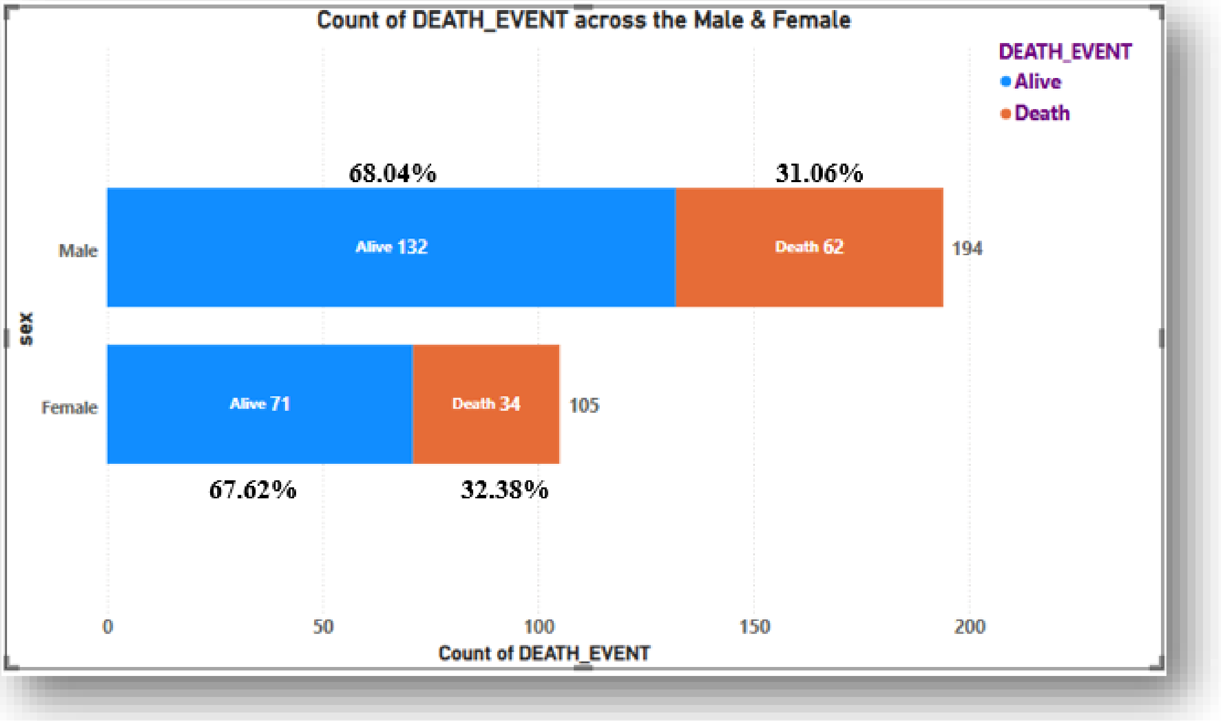


✓ Interpretation:

This bar chart compares the count and proportion of DEATH_EVENT (Alive vs. Death) between smokers and non-smokers. Here's the interpretation:

- The survival rate is slightly higher among smokers (68.75%) than non-smokers (67.49%), but the difference is small.
- Similarly, the death rate is marginally higher in non-smokers (32.51%) compared to smokers (31.25

Comparision Death event across Sex (Gender)



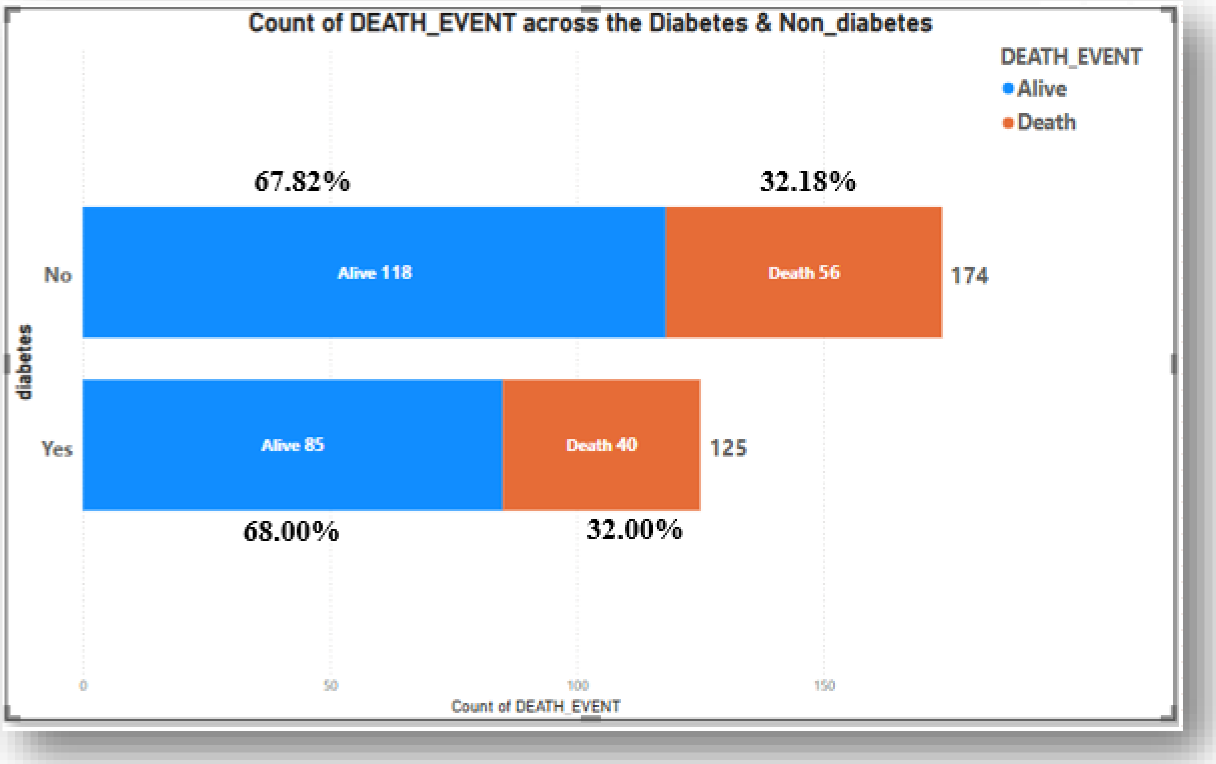
✓ Interpretation:

This bar chart compares the count and proportion of DEATH_EVENT (Alive vs. Death) between males and females.

Here’s the interpretation:

- The survival rate is slightly higher among males (68.04%) compared to females (67.62%), but the difference is very small.
- Similarly, the death rate is slightly higher in females (32.38%) than in males (31.06%)

Comparision Death event across Diabetes



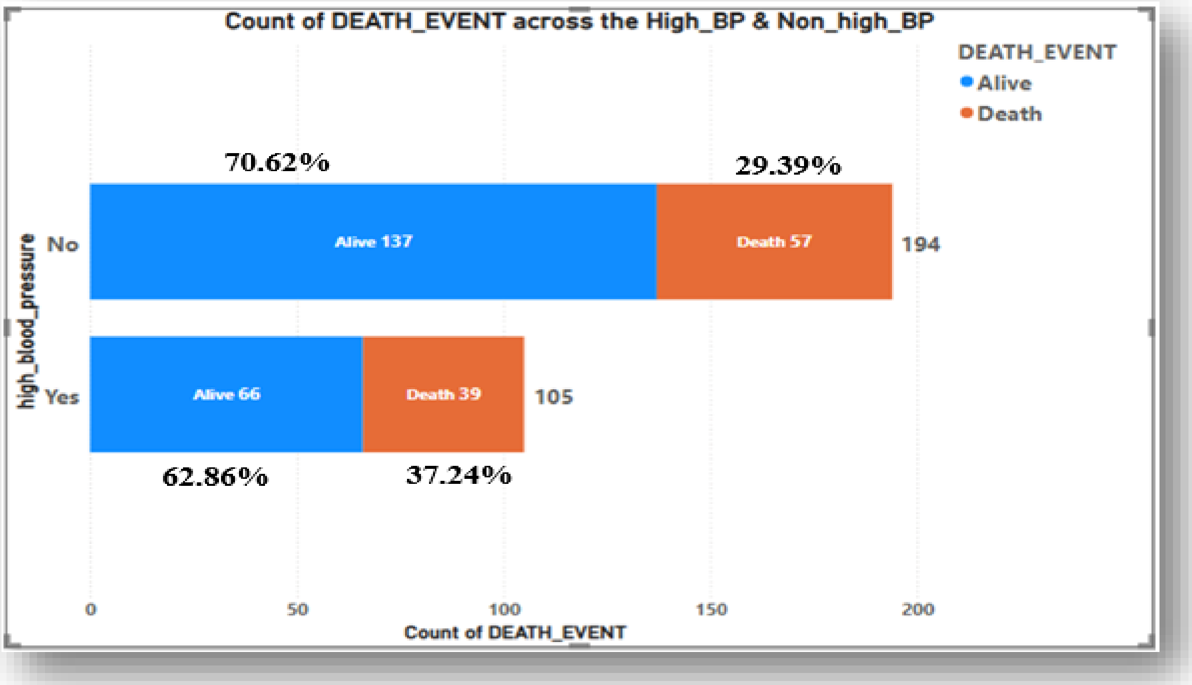
✓ Interpretation:

Here is the interpretation of the bar chart comparing DEATH_EVENT (Alive vs. Death) between Diabetic and Non-Diabetic patients in the same format as before:

- The survival rate is nearly identical for both groups (67.82% for non-diabetics vs. 68.00% for diabetics).
- The death rate is also nearly the same (32.18% for non-diabetics vs. 32.00% for diabetics).

- This suggests that diabetes does not appear to significantly affect survival in this dataset.

Comparision Death event across High Blood Pressure



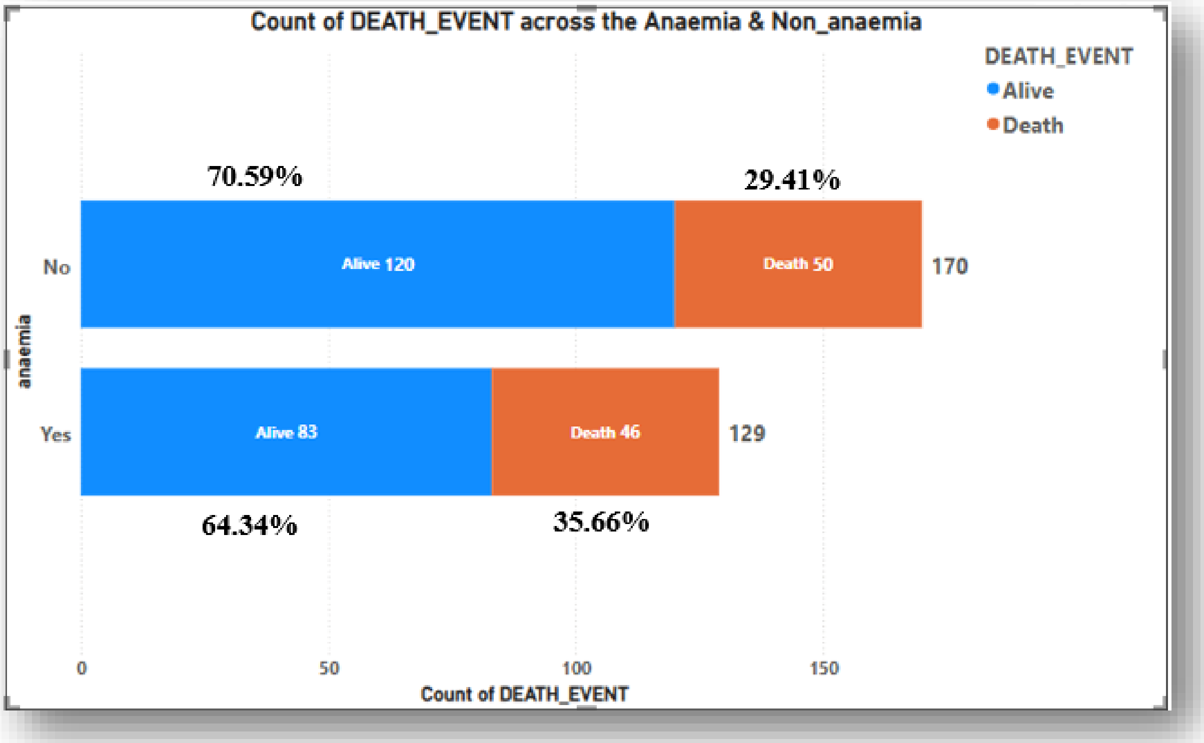
✓ Interpretation:

Here is the interpretation of the DEATH_EVENT (Alive vs. Death) between patients with and Normal High Blood Pressure (High_BP**)**

- The survival rate is lower in patients with high blood pressure (62.86%) compared to those without it (70.62%).
- The death rate is higher in high blood pressure patients (37.24%) than in those without high blood pressure (29.39%).

This suggests that high blood pressure may be associated with a higher risk of mortality in this dataset.

Comparision Death event across Anemia



✓ Interpretation:

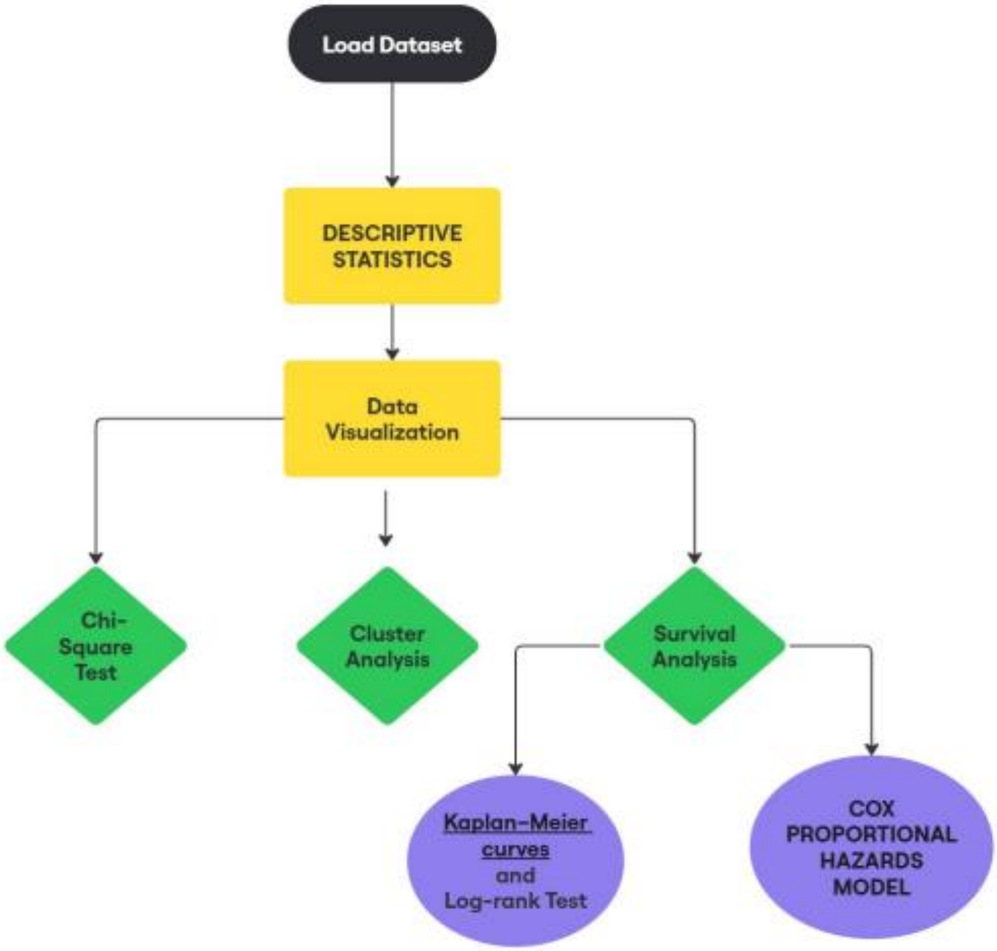
Here is the interpretation of the DEATH_EVENT (Alive vs. Death) between patients with and without Anemia in a structured format:

- The survival rate is lower in patients with anemia (64.34%) compared to those without anemia (70.59%).
- The death rate is higher in anemic patients (35.66%) than in those without anemia (29.41%).

This suggests that anemia may be associated with a higher risk of mortality in this dataset.

METHODOLOGY

Project: ["Statistical and Survival Analysis of Key Risk Factors in Heart Failure Patients".]



Statistical Software's used to Perform statistical Analysis

- **MS-Excel**
- **RStudio**
- **Power BI**
- **Python**

Chi-Square Test

The **Chi-Square Test** is a statistical method used to check the association between categorical variables. In machine learning, it helps select relevant features by identifying variables that have a significant relationship with the target variable.

Formula: The chi-square statistic is calculated as:

$$\mathrm{x}^2 = \sum \frac{(\mathrm{O} - \mathrm{E})^2}{\mathrm{E}}$$

Where:

O = Observed frequency

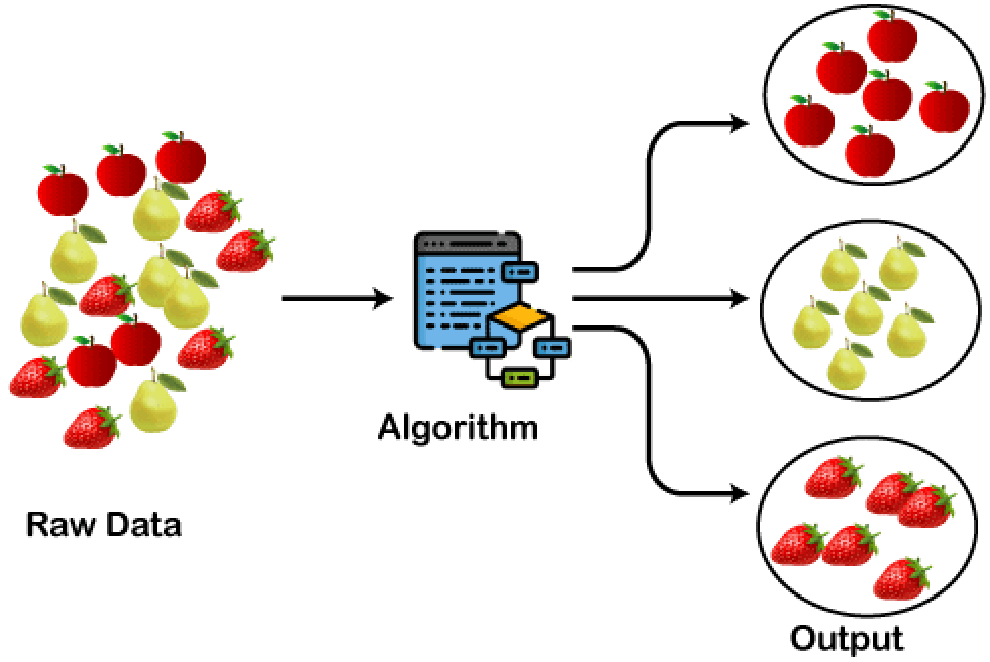
E = Expected frequency

Steps in R Programing:

- ✓ **Prepare Data :** Ensure that the variables are categorical. Convert continuous variables if needed.
- ✓ **Compute Chi-Square Statistic :** Compare **observed** vs. **expected** frequencies.
- ✓ **Determine Significance :** Use a **p-value threshold of 0.05**.
- ✓ **Select Features :** Keep only significant categorical variables for model building.

Cluster Analysis

Cluster Analysis is an unsupervised machine learning technique used to group similar data points into clusters. The goal is to maximize intra-cluster similarity (data points within a cluster are similar) and minimize inter-cluster similarity (data points from different clusters are dissimilar).



Applications of Cluster Analysis

- ✓ **Marketing Segmentation:** Group customers based on purchasing behavior.
- ✓ **Anomaly Detection:** Identify fraudulent transactions in banking.
- ✓ **Image Segmentation:** Recognize patterns in medical imaging.
- ✓ **Biology & Genetics:** Group genes with similar expression patterns.
- ✓ **Recommendation Systems:** Personalize user recommendations.

➢ K-Modes Clustering

K-Modes is an extension of the K-Means clustering algorithm specifically designed for categorical data. Since K-Means relies on Euclidean distance, which is not meaningful for categorical attributes, K-Modes uses a different approach to measure similarity and update cluster centroids.

➢ How K-Modes Works

Initialization: Select K initial cluster centroids randomly from the dataset. Each centroid is represented by a mode (most frequent category in each feature).

Assignment of Data Points: Compute the Hamming distance (number of mismatched categorical attributes) between each data point and the cluster centroids. Assign each point to the nearest cluster.

Updating Cluster Centroids: The new centroid for each cluster is updated by selecting the most frequent category for each attribute (the mode of the cluster).

Repeat Steps 2 and 3 until the centroids no longer change or a stopping criterion is met.

➢ Advantages of K-Modes

✓ Works well for categorical data.

✓ Efficient and simple to implement.

✓ Scales well for large datasets

- **Disadvantages of K-Modes**

  ✕ Sensitive to the choice of initial centroids.

  ✕ May not handle noise and outliers well.

  ✕ The number of clusters (K) must be predefined.

- **How DBSCAN Works**

Choose Parameters: Set ε (radius of neighborhood) and MinPts (minimum points required to form a dense region). Find Core Points: Identify core points with at least MinPts neighbors.

Expand Clusters: Start with a core point and expand to its density-reachable points. Continue growing until all reachable points are included.

Classify Remaining Points: Assign border points to nearest clusters. Mark outliers as noise.

- **Advantages of DBSCAN**

  ✓ Does not require specifying K (number of clusters).

  ✓ Handles clusters of arbitrary shapes (non-spherical).

  ✓ Detects outliers automatically.

➢ Disadvantages of DBSCAN

❌ Choosing ε and MinPts can be tricky, and wrong values may lead to poor clustering.

❌ Fails in datasets with varying densities (may incorrectly merge or split clusters).

❌ Computationally expensive for large datasets ($O(n^2)$ complexity).

➢ Hierarchical Clustering

Hierarchical clustering is a hierarchy-based clustering method that organizes data into a tree-like structure (dendrogram). It can be categorized into:

Agglomerative (Bottom-Up): Start with individual points and merge clusters iteratively.

Divisive (Top-Down): Start with all points in one cluster and split them iteratively.

How Agglomerative Hierarchical Clustering (AHC) Works

- ✓ **Start with N clusters:** Each data point is its own cluster.
- ✓ **Compute Distance Matrix:** Measure similarity using distance metrics (e.g., Euclidean, Manhattan).
- ✓ **Merge Closest Clusters:** Find and merge two closest clusters based on linkage criteria:
  - **Single Linkage:** Minimum distance between clusters.
  - **Complete Linkage:** Maximum distance between clusters.
  - **Average Linkage:** Average distance between clusters.

- ✓ **Repeat Until One Cluster Remains**: Merge until all data points form a single cluster.

How Divisive Hierarchical Clustering Works

- ✓ Start with all data points in one cluster.
- ✓ Use a clustering algorithm (e.g., K-Means) to split into smaller clusters.
- ✓ Repeat splitting until each data point is its own cluster.

Advantages of Hierarchical Clustering

✓ No need to specify K (number of clusters).

✓ Dendrogram provides a visual representation of data relationships.

✓ Works well for small to medium datasets.

Disadvantages of Hierarchical Clustering

✕ Computationally expensive (O (n^2 log n)), making it inefficient for large datasets.

✕ Sensitive to noise and outliers, especially with single-linkage clustering.

✕ No flexibility to modify clusters once formed.

➢ Mahalanobis Distance

Mahalanobis distance is a distance metric that measures how far a point is from a distribution, considering correlations between variables. Unlike Euclidean distance, which treats all variables equally, Mahalanobis distance accounts for the variance and correlation of the data.

Formula

The Mahalanobis distance D_M between a point x and the mean μ of a distribution is given by:

$$\boldsymbol{D_M}(\boldsymbol{x}) = \sqrt{(\boldsymbol{x} - \mu)^T \boldsymbol{S}^{-1}(\boldsymbol{x} - \mu)}$$

Where:

x = data point (vector)

μ = mean of the dataset

S = covariance matrix of the dataset

S^{-1} = inverse of the covariance matrix

T = transpose operation

Mahalanobis distance is used in clustering to measure similarity while considering correlations between features. It helps in detecting non-spherical clusters, making it useful in DBSCAN, hierarchical clustering, and modified K-Moad's. This distance metric improves clustering accuracy in datasets with correlated variables.

Survival analysis

Survival analysis is a branch of statistics that deals with time-to-event data. It is used to analyse the expected duration of time until one or more events happen, such as death, disease progression, equipment failure, or customer churn. The key feature of survival data is **censoring**, which occurs when the event of interest has not been observed for some subjects by the end of the study.

Key Concepts in Survival Analysis

1. **Survival Time (T)**: The time from a defined starting point to the occurrence of a given event.
2. **Censoring**: When the exact survival time is unknown. Common types:
   - **Right censoring**: The event has not occurred by the end of the study.
   - **Left censoring**: The event occurred before the study started.
   - **Interval censoring**: The event happened between two known time points.
3. **Survival Function (S(t))**: The probability that the event has not occurred by time *t*:

   $$S(t) = P(T > t)$$

   It is a non-increasing function, starting at 1 and approaching 0 over time.

4. **Hazard Function (λ(t))**: The instantaneous risk of the event occurring at time tt, given that it has not yet occurred:

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

5. Cumulative Hazard Function (H(t)): Represents the accumulated risk over time:

$$H(t) = \int_{0}^{t} \lambda(u)du$$

✚Kaplan–Meier curves

The **Kaplan-Meier (KM) estimator** is a non-parametric method used to estimate the **survival function** from time-to-event data, especially when censoring is present. It provides a stepwise estimate of the probability that an individual survives beyond a given time *t*.

- Plots survival curves for visual comparison of group survival rates over time.

$$\boldsymbol{S(t) = \prod_{ti \leq t} \left(1 - \frac{d_i}{n_i}\right)}$$

where,

- $\boldsymbol{S(t)}$= estimated survival probability at time $\boldsymbol{t}$
- $\boldsymbol{t_i}$ = time of each event (death or failure)
- $\boldsymbol{d_i}$ =number of events (deaths) at time $\boldsymbol{t_i}$
- $\boldsymbol{n_i}$ = number of individuals **at risk** just before time $\boldsymbol{t_i}$
- The product $\prod$ means that the survival probabilityis **updated at each event time**, multiplying the probability from previous time points

Each time an event occurs, the probability of surviving is updated by multiplying the previous survival probability by the conditional probability of surviving past that event time.

❖ Step of Kap Kaplan–Meier curves

- **Load Required Libraries**

- **Prepare Data :** Ensure survival time and event variables are correctly formatted.
- **reate the Survival Object :** Define the survival time and event status.
- **Fit the Kaplan–Meier Model :** Fit an overall **Kaplan–Meier estimator** without grouping.
- **Determine Statistical Significance :** Check the log-rank test for differences between groups.

❖ **Visualize Kaplan–Meier Curves**Plot the survival curve with confidence intervals. Stratified Kaplan–Meier plot.

❖ **Select Features for Model Building :** Use significant variables (p-value < 0.05) for further modeling (e.g., Cox regression).

❖ Assumptions of the Kaplan-Meier Estimator

1. **Independent Censoring**: The probability of being censored is unrelated to the survival probability.
2. **Consistent Event Definition**: All subjects experience the same type of event.
3. **Accurate Time Measurement**: The exact event times must be known.

✓ Advantages of the Kaplan-Meier Method

- Handles censored data effectively.
- Provides an empirical survival curve without assuming a specific distribution.
- Useful for comparing survival distributions between groups using statistical tests (e.g., the **log-rank test**).

❖Kaplan-Meier survival curve:

A **Kaplan-Meier survival curve** is a graphical representation of the estimated survival function over time. It is widely used in medical research, reliability analysis, and other fields dealing with time-to-event data. The curve is a step function that decreases at observed event times and remains constant between events.

Log-rank Test

The **log-rank test** is a statistical test used to compare survival distributions between two or more groups. It is commonly applied in **Kaplan-Meier survival analysis** to determine if there is a significant difference in survival times among groups.

❖ Hypothesis

Null Hypothesis (H_0): There is no difference in survival distributions between the groups.

Alternative Hypothesis (H_1): There is a significant difference in survival distributions between the group.

❖Assumption

1. The survival times are independent between subjects.
2. The censoring is non-informative (i.e., subjects are lost to follow-up or censored randomly and not due to a systematic reason).
3. The proportional hazards assumption holds (i.e., the ratio of hazards between groups remains constant over time).

Test Statistics

$$\chi^2 = \sum \frac{\left(O_j - E_j\right)^2}{V_j}$$

where,

$$O_j = Total\ observed\ deaths\ in\ group\ j.$$

$$E_j = Total\ expected\ deaths\ in\ group\ j.$$

$$V_j = Variance\ of\ \left(O_j - E_j\right)$$

Decision Criteria: $\chi^2 > \chi^2{}_{(\propto,k-1)}$ Reject H_0

$\chi^2 \leq \chi^2{}_{(\propto,k-1)}$ Accept H_0

COX PROPORTIONAL HAZARDS MODEL

The Cox model evaluates the hazard function, which describes the risk of the event happening at time t, given survival up to that time. The model doesn't assume any specific baseline hazard shape (non-parametric for time) but assumes that covariates have a multiplicative effect on the hazard.

Hazard Function and proportional Hazards Assumption:

The **hazard function**, $h(t)$, represents the **instantaneous risk of an event occurring at time $\boldsymbol{t}$** given that the subject has survived up to that time.

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

Model:

The hazard function at time t is given by

$$h(t|x) = h_0(t)\exp(\beta_1 X_1 + \beta_2 X_2 + \_\_\_\_\_\_\_\_\_\_ + \beta_n X_n)$$

Where,

$h(t|x) = hazard\ at\ time\ t\ given\ covariates\ X.$

$h_0(t)\ = baseline\ hazard\ function$

$X_1, X_2, \dots \dots, X_n =$ predictor variables

$\beta_1 + \beta_2 + \cdots \dots . + \beta_p = regression\ coefficients.$

- $If\ HR > 1.\quad Higher\ risk\ of\ Event\ occuring$
- $If\ HR < 1\quad Lower\ Risk\ of\ Event\ occuring$
- $If\ HR = 1:\ \ NO\ effect$

- **Objective 1 - Assess statistical association** between (Smoking status, Anemia, Diabetes, Sex and High blood pressure) and **DEATH_EVENT**.

❖ **Chi-Square Test -** We performed the Chi-Square test to check the association between categorical variables and risk_segment.

Hypothesis

H0 : There is no association between DEATH_EVENT and (Smoking status, Anemia, Diabetes Sex and High blood pressure)

H1 : There is association between DEATH_EVENT and (Smoking status, Anemia, Diabetes Sex and High blood pressure)

Chi- square Test	P_value	Result
Smoking status	0.9318 > 0.05	No association
Anemia	0.3073 > 0.05	No association
Diabetes	1.0000 > 0.05	No association
Sex	1.0000 > 0.05	No association
High blood pressure	0.000062 < 0.05	Association

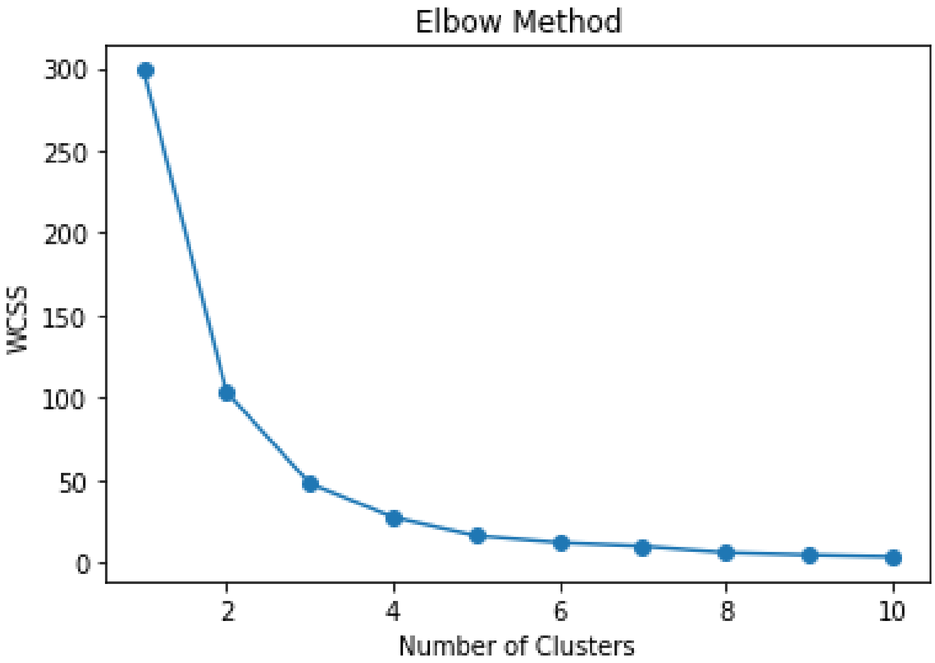
Since the **p-value**(0.3073,0.9318,1.0000 and 1.0000 > 0.05), (**P-value is greter than 0.05**),We Accept the **null hypothesis.**

This means that there is **no association** between (Smoking ,Anaemia, Diabetes and Sex) and death event.Here **Association** only one present in High blood pressure and death event.

High blood pressure has an impact on survival.

- **Objective 2 - To cluster age groups and assess their association with patient death events for a better understanding of death events patterns.**

Elbow Method

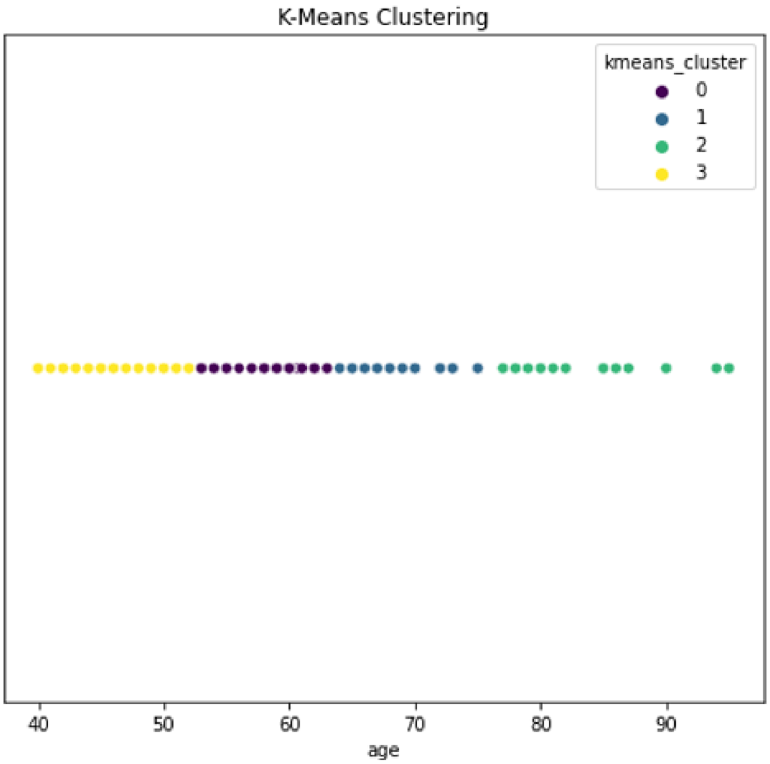


✓ Interpretation:

- The "elbow" point is where the curve **bends** and starts to level off. This is the optimal **k** because adding more clusters beyond this point yields **diminishing returns**.
- In this case, the elbow seems to be around **k = 4**.

K- Means

We first perform **K-Means clustering** to group patients based on age and assess the association between these clusters and patient death events. This helps in gaining a better understanding of death events patterns, allowing for effective categorization of patients into different groups based on their risk status related to death events.

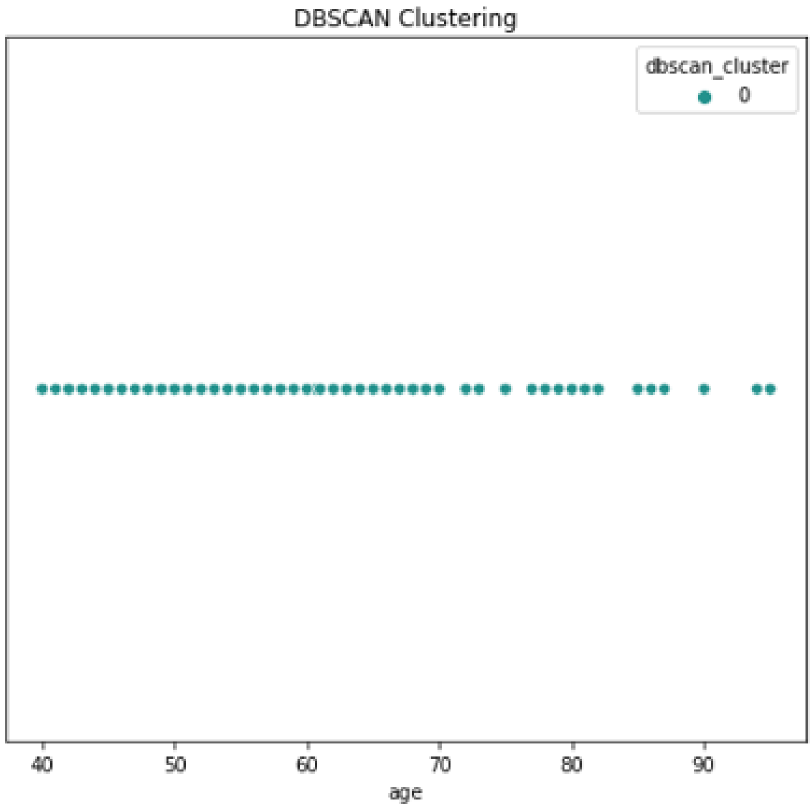


✓ Interpretation:

The graph shows that K-Means clustering segments age into four distinct groups. This segmentation helps in identifying patterns among different age groups, allowing for a clearer understanding of their association with death event

DBSCAN

Now we apply **DBSCAN clustering** to group patients based on age and analyze its association with patient death events. This approach helps in effectively categorizing patients into different risk groups based on their likelihood of experiencing a death event.

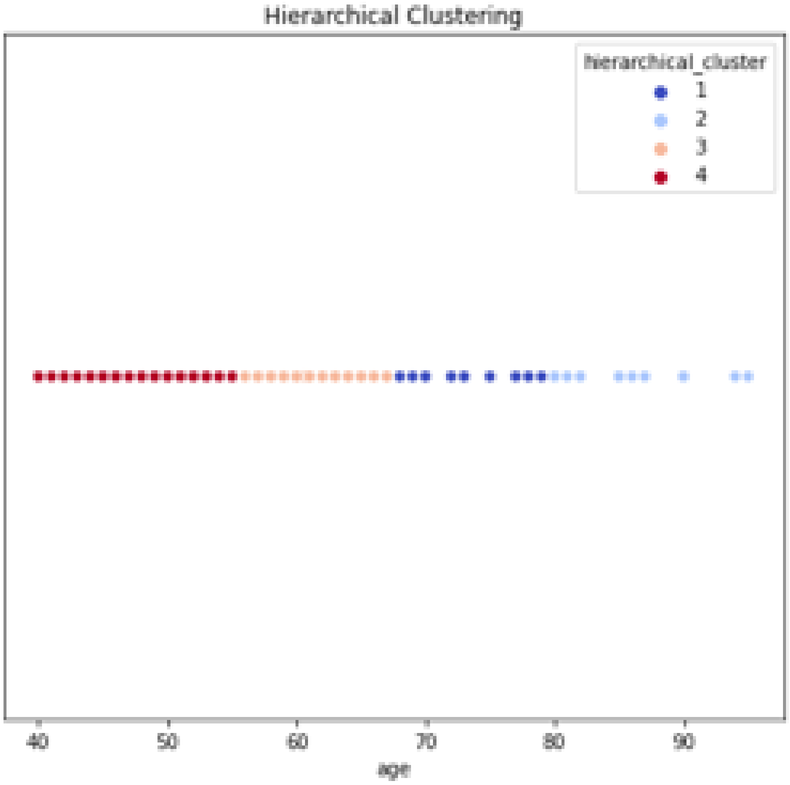


✓ Interpretation:

In the graph above, we applied DBSCAN clustering, but it did not perform well. Instead of forming distinct age-based clusters, it grouped the entire dataset into a single cluster, indicating that DBSCAN may not be suitable for segmenting age in this case.

Hierarchical Clustering

We first apply Hierarchical Clustering to group patients based on age and analyze its association with patient death events. This method provides a tree-like structure (dendrogram) that helps identify natural groupings within the data. By leveraging hierarchical clustering, we can effectively categorize patients into different risk groups, enhancing our understanding of death patterns and the relationship between age and death events.



✓ Interpretation:

we applied Hierarchical Clustering, which successfully segmented the age data into 4 distinct clusters. This indicates that hierarchical clustering effectively identifies natural groupings within the age distribution.

Mahalanobis Distance

After performing K means hierarchical and division clustering we evaluate the segmentation quality using the Mahalanobis distancto determine which clustering technique provides a most effective patient’s segmentation age groups.

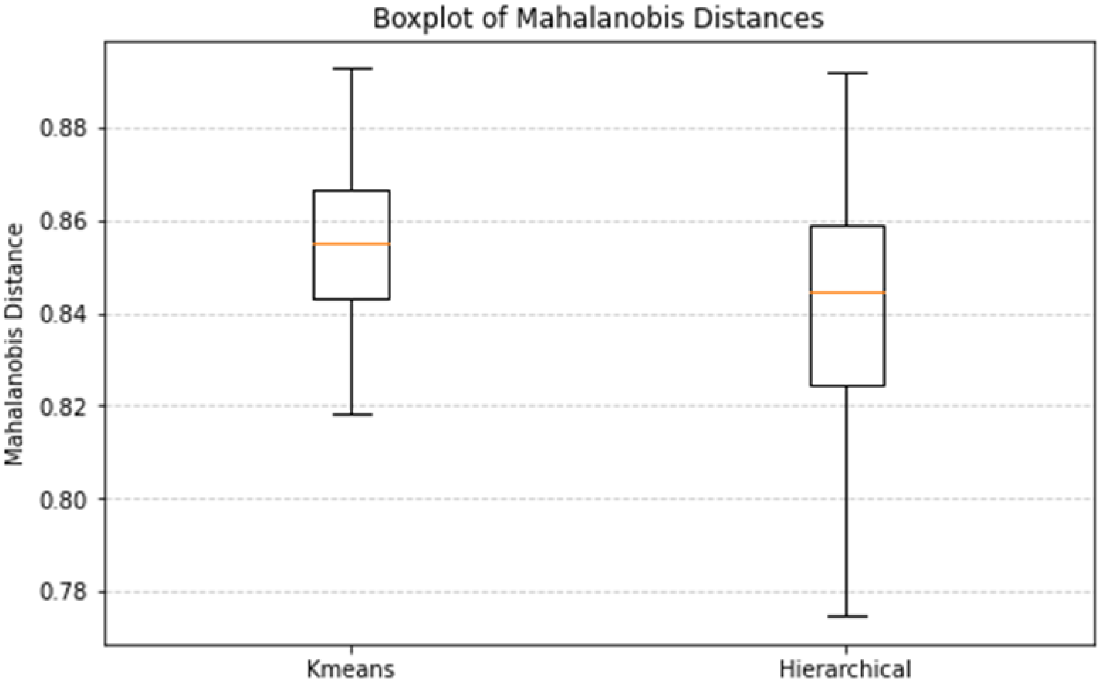
Mahalanobis Distance :

K-Means: {1: 0.8517722587666068, 0: 0.8579676503142867, 3: 0.8926307987853063, 2: 0.8181064446092193}

DBSCAN: {0: 0.8010151854636127}

Hierarchical: {1: 0.8412678249295291, 4: 0.8475969587104876, 3: 0.892005949764283, 2: 0.7746636271602741}

Here we observe that Dbiscan clustering is not performing well so we compare K means and hierarchical clustering distances using box plot to determine better clustering technique for patient segmentation.

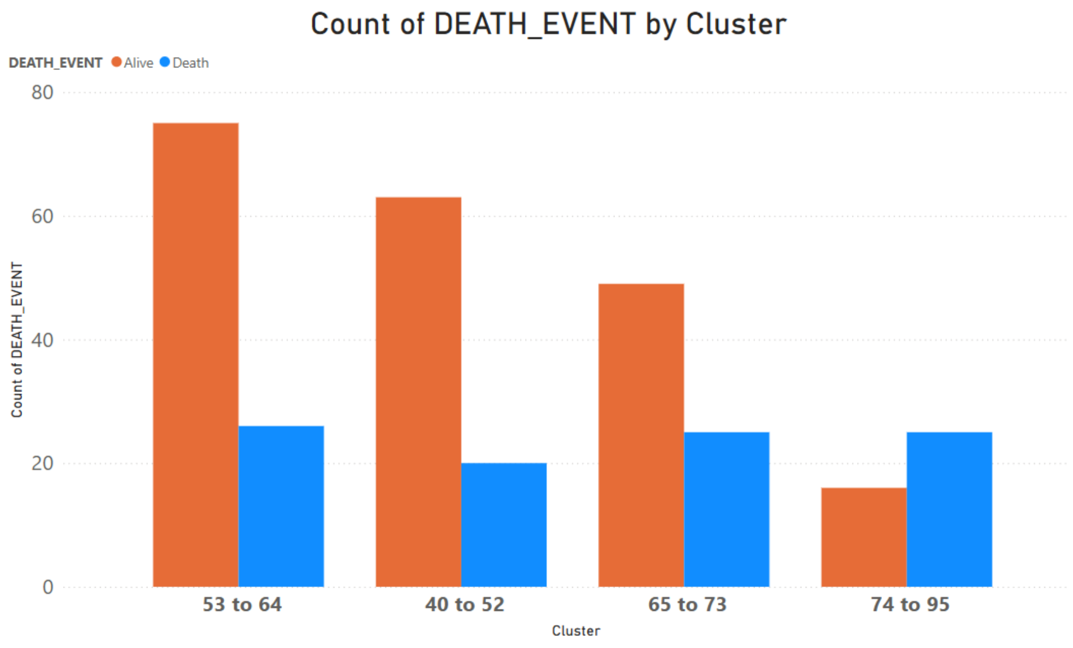


✓ Interpretation:

Here we can see in above box plot K means clustering has lower distances resulting in more compact and tightly distributed clusters as compared to hierarchical clustering That’s why K means clustering is preferred in this case.

We segment patients based on the four identified clusters and analyze their death event.

❖ Count of Death event by Cluster (Age_Group)



✓ Interpretation:

- **Cluster: 40 to 52 (25.74% Death Rate)**

✓ Although this is the youngest age group, over a quarter (25.74%) of patients have died.

✓ This suggests that even younger individuals in this dataset face significant health risks leading to mortality.

- **Cluster: 53 to 64 (24.01% Death Rate)**

✓ Despite being slightly older, this group has a lower death rate (24.01%) than the 40–52 cluster.

✓ This could indicate that middle-aged patients may have better medical management or resilience in this dataset.

- **Cluster: 65 to 73 (33.78% Death Rate)**

✓ The mortality rate increases significantly to 33.78%, showing that this age range is at a higher risk of death.

✓ This suggests a possible turning point where aging contributes more strongly to mortality outcomes.

- **Cluster: 74 to 95 (60.98% Death Rate)**

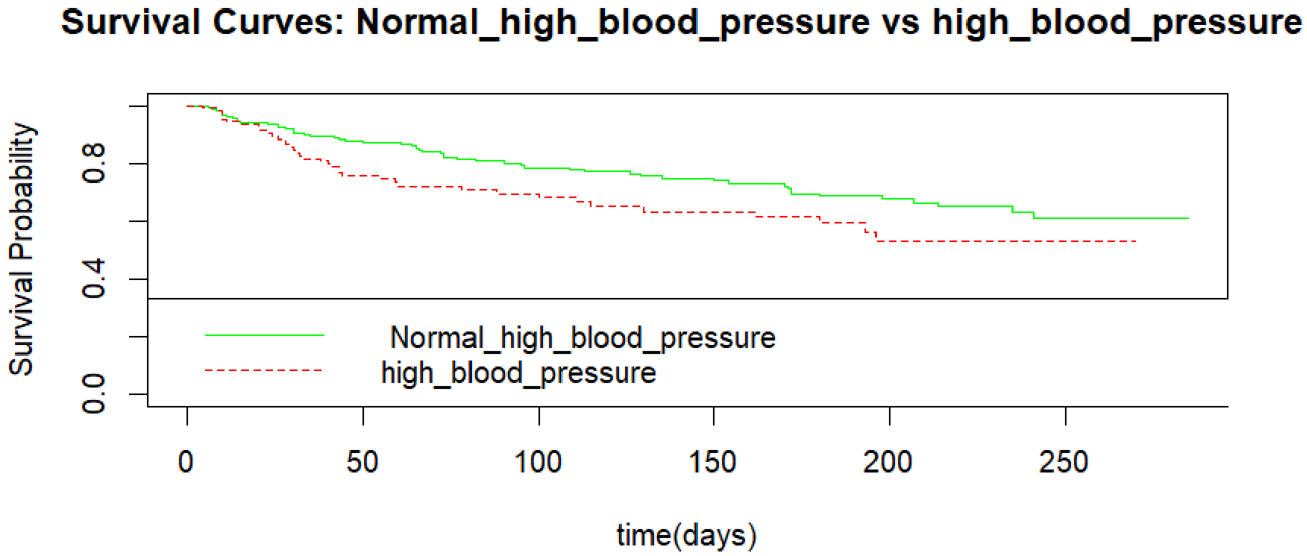
✓ This group has the highest mortality rate, with nearly two-thirds (60.98%) of patients dying.

✓ The drastic increase in deaths highlights advanced age as a major mortality risk factor, requiring focused healthcare and intervention strategies.

Objective 3 - To evaluate the impact of independent variables on survival or Death Event outcomes using survival analysis methods.

# Plot the survival curves

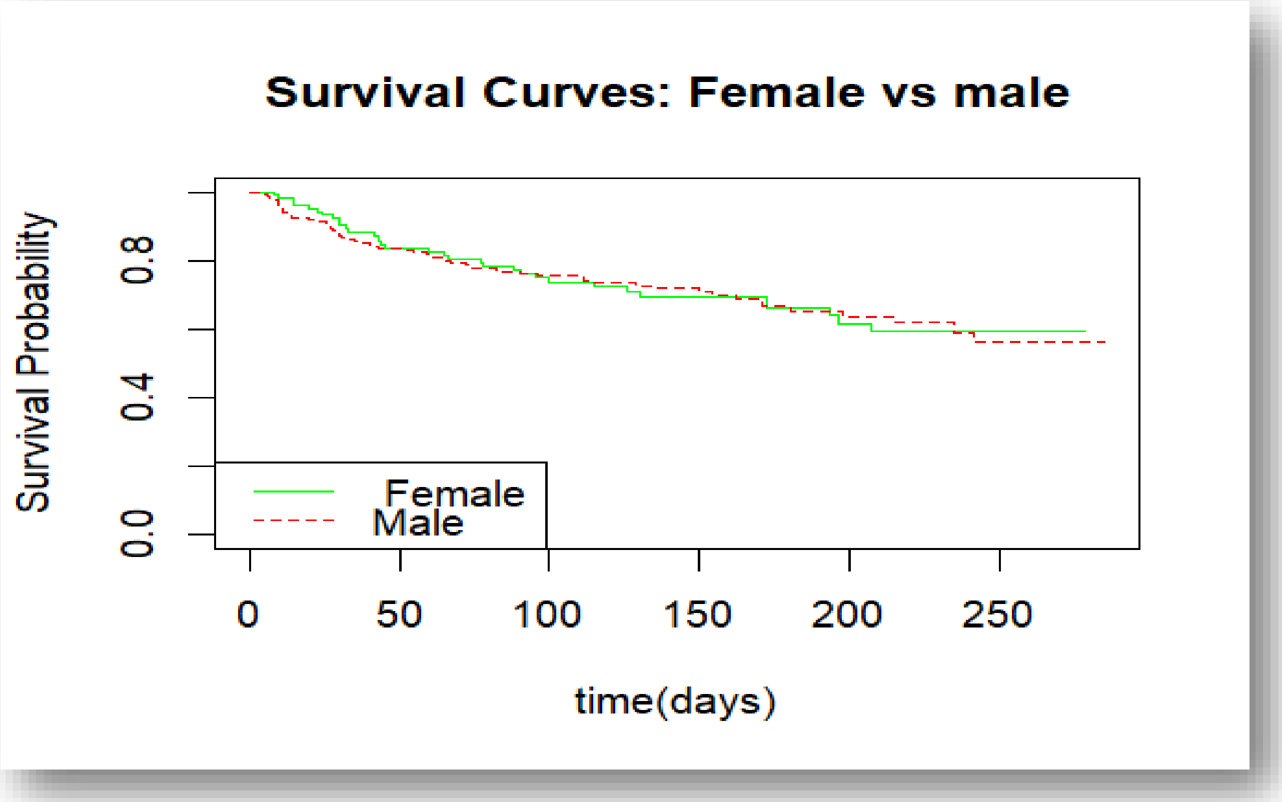
1) Fit survival curves for high_blood_pressure vs. Normal _high_blood_pressure



✓ Interpretation:

The green line (Normal high blood pressure) generally higher than the red line (high blood pressure), indicating that individuals norma high blood pressure have a better survival probability over time.

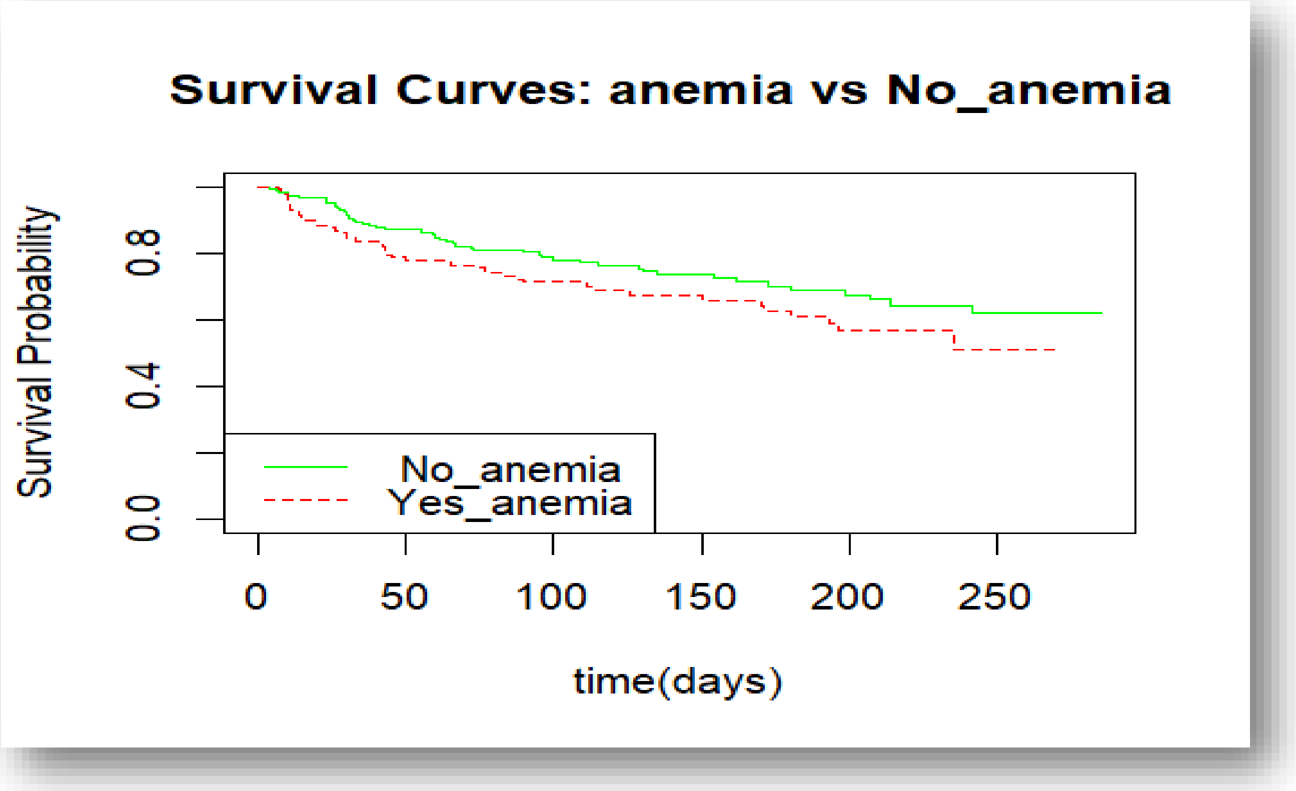
2) Fit survival curves for Male vs. Female



✓ Interpretation:

The survival probabilities for **females** and **males** are nearly identical throughout the follow-up period, indicating no significant difference in survival between the sexes.

3) Fit survival curves for anemia vs. No_anemia

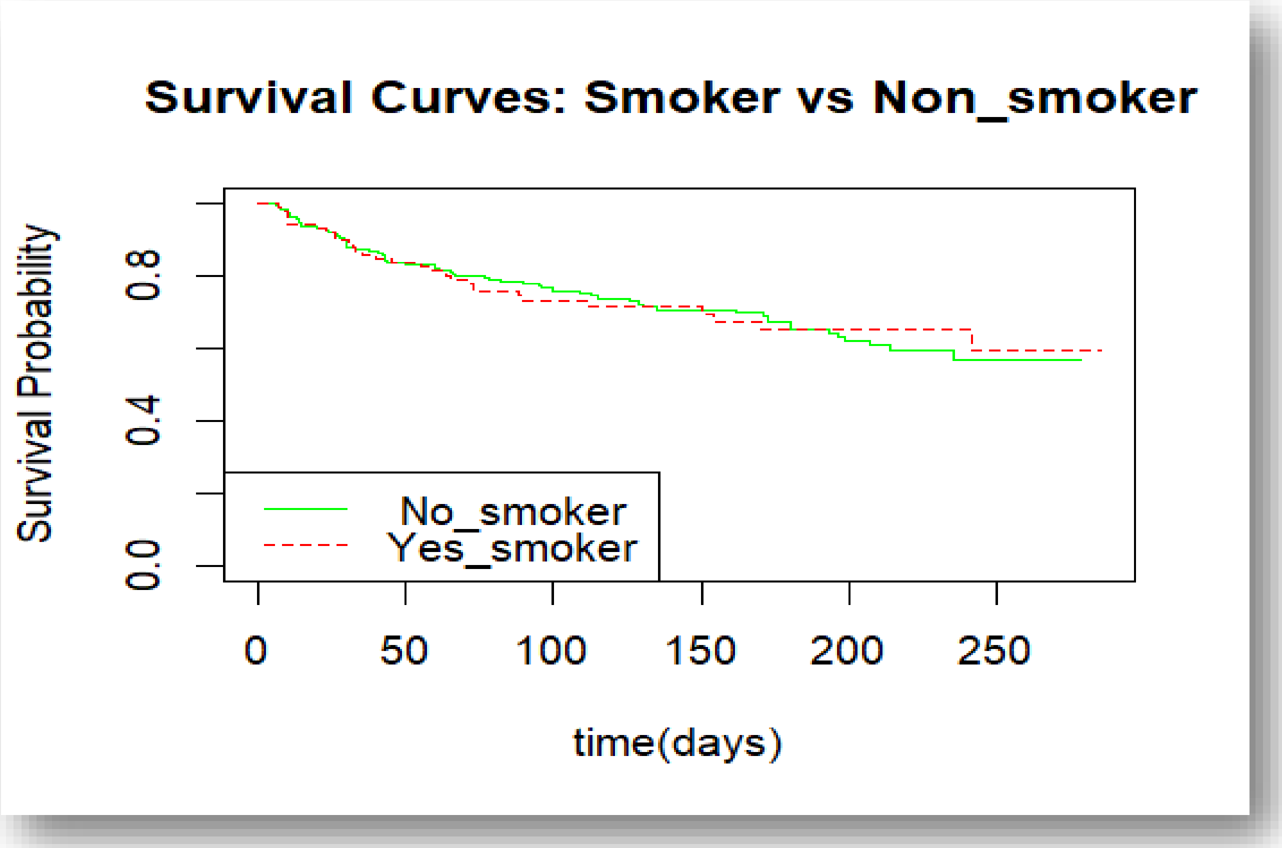


✓ Interpretation:

The green line (No anemia) generally higher than the red line (Yes anemia), indicating that without anemia have a better survival probability over time.

Patients with anemia have lower survival probabilities than those without anemia.

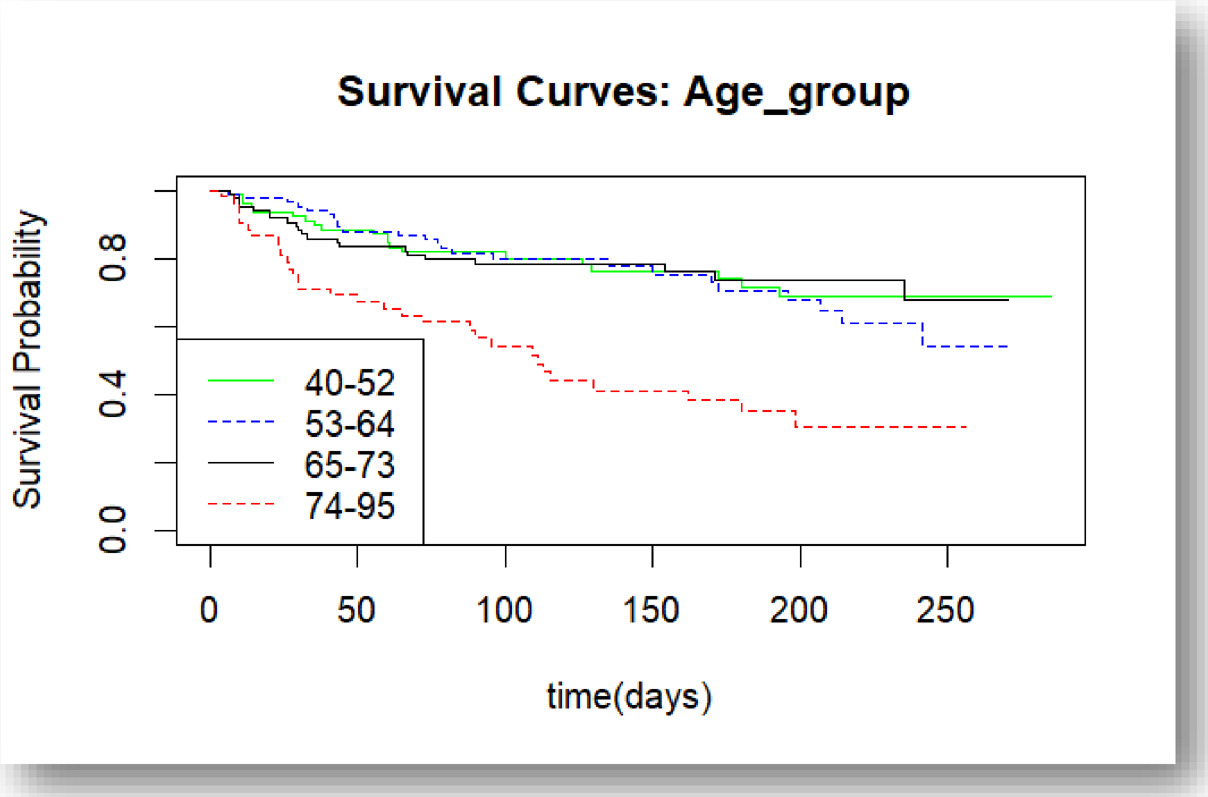
4) Fit survival curves for smokers vs. Non_smokers



✓ Interpretation:

The survival probabilities for No_smoker and smoker are nearly identical throughout the follow-up period, indicating no significant difference in survival between the No_smoker and smoker.

5) Fit survival curves for Age_group



✓ Interpretation:

Survival probability decreases as age increases, confirming that older individuals have a higher risk of death event.

The 40-52 and 53-64 groups show better long-term survival, while the 74-95 group has the lowest survival probability.

These findings suggest the need for age-specific medical interventions, with greater focus on high-risk elderly patients (74-95).

Log-Rank Test

Log-Rank Test	P_value	Result
high blood pressure vs. No high blood pressure	0.04<0.05	significant difference in survival distributions
Male vs. Female	0.9>0.05	No significant difference in survival distributions.
Anemia vs. No_anemia	0.1>0.05	No significant difference in survival distributions.
Smokers vs. Non_smokers	1.0000 > 0.05	No significant difference in survival distributions.
Age_group	0.0002 < 0.05	significant difference in survival distributions

✓ Interpretation:

Since high blood pressure and Age_group the p-value is less than 0.05 so we reject H0 and say that the result is **significant difference** in survival distributions.

Since Sex, Anemia and Smokers status the p-value is greater than 0.05, so we do not reject H0 and say that the result is **no significant difference** in survival distributions

❖ **Conclusion :** High blood pressure and Age_group has an impact on survival.

Objective 4 - To identify which clinical factors significantly affect patient death event time using COX PROPORTIONAL HAZARDS MODEL

❖ TEST COX PROPORTIONAL HAZARDS MODEL

```
Cox_model= coxph( surv_obj ~ Age_group + diabetes + platelets +
                    sex + high_blood_pressure + creatinine + sodium + smoking +
              anaemia + creatinine_phosphokinase + ejection_fraction, data = data)
```

```
> summary(cox_model)
Call:
coxph(formula = surv_obj ~ Age_group + diabetes + platelets +
   sex + high_blood_pressure + creatinine + sodium + smoking +
   anaemia + creatinine_phosphokinase + ejection_fraction, data = data)

  n= 299, number of events= 96

                               coef  exp(coef)   se(coef)      z Pr(>|z|)
Age_group53-64            6.347e-02  1.066e+00  3.207e-01  0.198 0.843101
Age_group65-73            1.290e-01  1.138e+00  3.208e-01  0.402 0.687627
Age_group74-95            1.175e+00  3.239e+00  3.107e-01  3.782 0.000156 ***
diabetesYes               1.585e-01  1.172e+00  2.271e-01  0.698 0.485295
platelets                -5.385e-07  1.000e+00  1.144e-06 -0.471 0.637749
sexMale                  -2.324e-01  7.926e-01  2.514e-01 -0.924 0.355284
high_blood_pressureYes    3.629e-01  1.438e+00  2.199e-01  1.651 0.098762 .
creatinine                3.240e-01  1.383e+00  7.167e-02  4.521 6.15e-06 ***
sodium                   -4.565e-02  9.554e-01  2.352e-02 -1.941 0.052293 .
smokingSmoker             8.964e-02  1.094e+00  2.502e-01  0.358 0.720078
anaemiaYes                4.465e-01  1.563e+00  2.181e-01  2.047 0.040691 *
creatinine_phosphokinase  1.967e-04  1.000e+00  1.006e-04  1.954 0.050678 .
ejection_fraction        -4.643e-02  9.546e-01  1.032e-02 -4.499 6.84e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


Concordance= 0.735  (se = 0.027 )
Likelihood ratio test= 76.63  on 13 df,   p=5e-11
Wald test            = 86.08  on 13 df,   p=8e-13
Score (logrank) test = 90.96  on 13 df,   p=9e-14
```

Cox_model1 =coxph(surv_obj ~ Age_group + high_blood_pressure + creatinine + sodium + anaemia + creatinine_phosphokinase + ejection_fraction, data = data)

```
> summary(cox_model1)
Call:
coxph(formula = surv_obj ~ Age_group + high_blood_pressure + 
    creatinine + sodium + anaemia + creatinine_phosphokinase + 
    ejection_fraction, data = data)

  n= 299, number of events= 96 

                               coef  exp(coef)   se(coef)      z Pr(>|z|)    
Age_group53-64            4.478e-02  1.046e+00  3.162e-01  0.142 0.887367    
Age_group65-73            1.100e-01  1.116e+00  3.190e-01  0.345 0.730281    
Age_group74-95            1.074e+00  2.927e+00  2.976e-01  3.608 0.000308 ***
high_blood_pressureYes    3.925e-01  1.481e+00  2.165e-01  1.813 0.069895 .  
creatinine                3.163e-01  1.372e+00  7.018e-02  4.507 6.57e-06 ***
sodium                   -4.844e-02  9.527e-01  2.350e-02 -2.061 0.039268 *  
anaemiaYes                4.500e-01  1.568e+00  2.172e-01  2.072 0.038277 *  
creatinine_phosphokinase  1.873e-04  1.000e+00  9.945e-05  1.884 0.059624 .  
ejection_fraction        -4.511e-02  9.559e-01  1.014e-02 -4.448 8.66e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


Concordance= 0.73  (se = 0.028 )
Likelihood ratio test= 75.15  on 9 df,   p=1e-12
Wald test            = 86.58  on 9 df,   p=8e-15
Score (logrank) test = 89.89  on 9 df,   p=2e-15
```

✓ Interpretation:

1. **Age Groups :**
   - **Age 74-95**: Hazard ratio (HR) = **2.93** (**p = 0.0003**), indicating that patients in this age group have **2.93 times higher risk** of mortality compared to the reference group (40-52).
   - **Age 53-64 and 65-73**: Not statistically significant (**p > 0.05**), meaning they do not significantly affect survival compared to the reference group.(40-52).

2. **High Blood Pressure :**
    - HR = **1.48** (**p = 0.0699**), suggesting an increased risk, but it is not statistically significant at the 5% level but is significant for 1% level.
3. **Creatinine (Kidney Function Marker)**
    - HR = **1.37** (**p = 6.57e-06**), indicating a **significant increase in hazard risk** 37% higher For every 1-unit increase in creatinine levels
4. **Sodium:**
    - HR = **0.95** (**p = 0.0393**), suggesting that **higher sodium levels are associated with a small but significant protective effect**
5. **Anaemia**
    - HR = **1.57** (**p = 0.0383**), indicating that patients with anaemia have a significantly higher risk of death event **(56.8% higher than non-anaemic patients).**
6. **Creatinine Phosphokinase (CPK) :**
    - HR = **1.0002** (**p = 0.0596**), showing a very minor effect, but not statistically significant at 5% but is statistically significant for 1% level.
7. **Ejection Fraction (Heart Pumping Ability) :**
    - HR = **0.9559** (**p = 8.66e-06**), meaning higher ejection fraction is significantly associated with lower death event risk. **1% increase in ejection fraction reduces the risk by approximately 4.4%.**

❖ Conclusion :

- Age 74-95, high creatinine levels, low sodium levels, presence of anaemia, and reduced ejection fraction significantly impact survival.
- Creatinine and ejection fraction are the most influential variables, with higher creatinine increasing risk and better ejection fraction reducing risk.
- High blood pressure and CPK levels do not show strong statistical significance, but their potential impact should be explored further.
- The model performs well (C-index = 0.73), indicating good discrimination between high- and low-risk patients.

This analysis suggests that managing kidney function (creatinine), improving heart efficiency (ejection fraction), and addressing anaemia could improve survival outcomes for patients.

OVERALL CONCLUSION

1. Log Rank Test :

- High blood pressure and Age_group significantly affects in human survival.

2. Kaplan–Meier curve :

- Patients with high blood pressure have lower survival probabilities than those with normal blood pressure.
- Patients with anemia have lower survival probabilities than those without anemia.
- Older age groups have lower survival probabilities, with the 74-95 age group experiencing the worst survival outcomes.

3. Cluster Analysis:

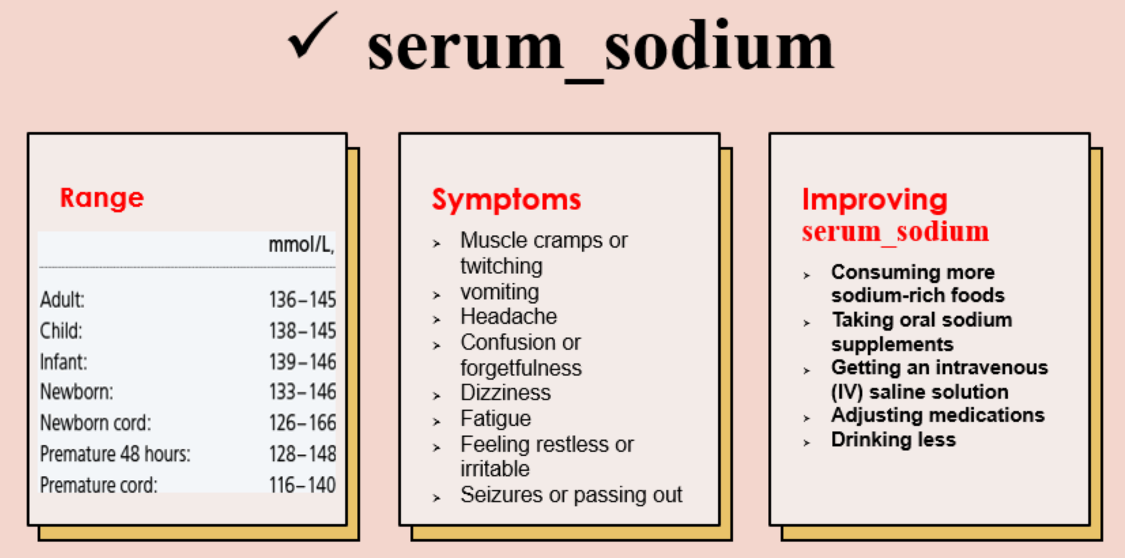
- In the age group 74 to 95 we can seee that death rate is significantly high.
- While for age groups 40 to 52 and 53 to 64 have similar death rates, i.e. middle-aged patients have similar death rate.
- Now moving further upto 65 age death rate dose not show any significant fluctuation but just after 65 age we can see a significant rise in death rate this indicates that patients of age more than 65 years are more likely to die and thus require medical attention as compared to patients less than 65 years of age.

4. Survival Analysis:

Here we see how likely is a patient to survive based on the available variables:

- We considered (40-52) as our refrence age group on basis of this we can clealy see that patients in age group (74-95) have high risk of death(higher death rate) i.e. increase in age leads to increase in death risk.
- Increasing high blood pressure will lead to incrase in death risk.
- Here high creatine level sugesst higher death risk i.e. higher the creatine level higher the death rate.
- Patients suffering from anemia will have higher death risk as compared to the ones not suffering from it.
- Here as Creatinine Phosphokinase level increases this leads to increase in the death rate meaning patients with higher Creatinine Phosphokinase levels will have higher death risk.
- Now **ejection fraction** is a variable who has a **protective effect** meaning that higher the fraction lower the death rate. This ejection fraction can be **increased** by following a healthy life,eating healthy and organic food along with daily excersise and keeping ones heart healthy.
- **Sodium level** also have a **protective effect** meaning higher the sodium level lower the death rate.

VARIOUS SUGGESTIONS TO INCREASE SODIUM LEVELS AND EJEECTION FRACTION.



✓ serum_sodium

Range

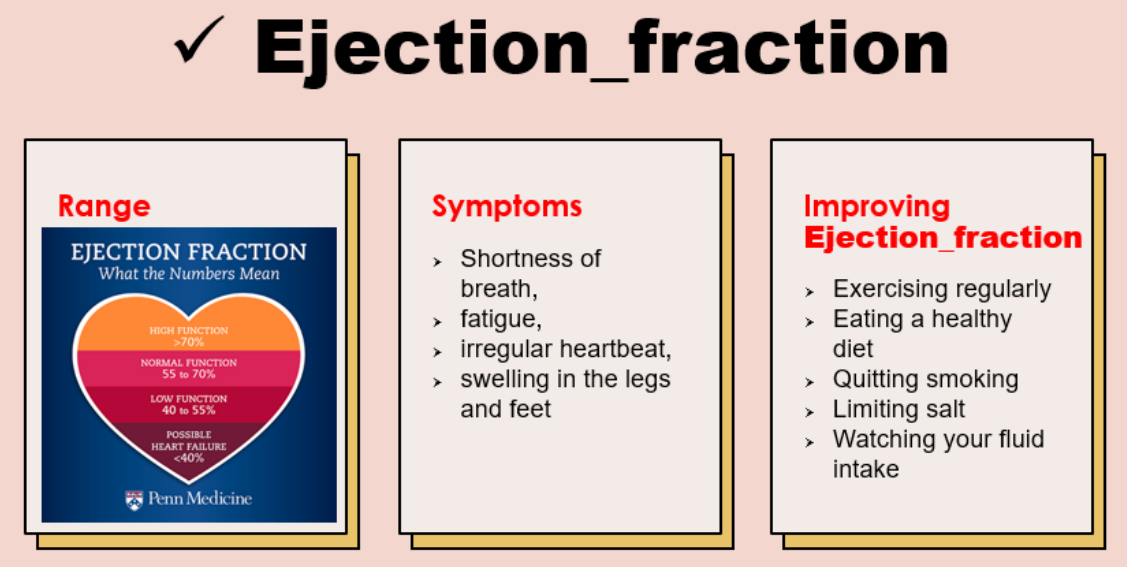
	mmol/L,
Adult:	136–145
Child:	138–145
Infant:	139–146
Newborn:	133–146
Newborn cord:	126–166
Premature 48 hours:	128–148
Premature cord:	116–140

Symptoms

- Muscle cramps or twitching
- vomiting
- Headache
- Confusion or forgetfulness
- Dizziness
- Fatigue
- Feeling restless or irritable
- Seizures or passing out

Improving serum_sodium

- **Consuming more sodium-rich foods**
- **Taking oral sodium supplements**
- **Getting an intravenous (IV) saline solution**
- **Adjusting medications**
- **Drinking less**



✓ Ejection_fraction

Range

Symptoms

- Shortness of breath,
- fatigue,
- irregular heartbeat,
- swelling in the legs and feet

Improving Ejection_fraction

- Exercising regularly
- Eating a healthy diet
- Quitting smoking
- Limiting salt
- Watching your fluid intake

Refrenced by-

SUMMARY

The study concludes that key factors such as high blood pressure, anemia, and older age significantly impact survival probabilities. Kaplan-Meier survival analysis demonstrates that patients with these risk factors have lower survival rates over time.

Cox proportional hazards models confirm these findings, emphasizing their statistical significance. The results suggest that early intervention and proper management of these conditions could improve patient outcomes. Additionally, model comparisons ensure the reliability of the analysis, reinforcing the validity of conclusions. Overall, the study highlights critical risk factors influencing survival and provides a foundation for future research and clinical decision-making.

RECOMMENDATIONS

1. **Early Detection and Management** – Regular screening for anemia and hypertension in at-risk populations can help improve survival rates through early intervention.
2. **Targeted Treatment Plans** – Personalized treatment strategies should be developed for older patients and those with high-risk conditions to improve their long-term survival.
3. **Lifestyle Modifications** – Encouraging healthy lifestyle choices, including a balanced diet, regular exercise, and smoking cessation, can help reduce risk factors affecting survival.
4. **Improved Patient Monitoring** – Implementing regular follow-ups and monitoring systems for high-risk patients can enhance early diagnosis and timely treatment adjustments.
5. **Clinical Decision Support** – Healthcare providers should use predictive models, such as logistic regression and Cox proportional hazards models, to assess patient risk and guide treatment strategies.
6. **Public Health Awareness** – Educational programs should be conducted to increase awareness about the impact of anemia, hypertension, and aging on survival outcomes.
7. **Further Research** – Future studies should focus on additional risk factors, potential interventions, and the effectiveness of treatment protocols in improving patient survival.

REFERENCES

Data Source: University of California, Irvine (UCI) Machine Learning Repository

License

This dataset is licensed under a Creative Commons Attribution 4.0 International (CC BY 4.0)

1. Dataset Link:

https://archive.ics.uci.edu/dataset/519/heart+failure+clinical+records

2. Survival Analysis A Self-Learning Text, Third Edition: (BOOK)

http://www.uop.edu.pk/ocontents/survival-analysis-self-learning-book.pdf

3. Cox Proportional Theory:

https://www.sthda.com/english/wiki/cox-proportional-hazards-model

4. Cox Model Assumptions:

https://www.sthda.com/english/wiki/cox-model-assumptions

5. Log Rank Test: https://encr.pw/MtuTs

6. Cox Proportional Hazard Rate Model: https://encr.pw/tyUll

APPEINDIX

R code :

```
data=read.csv(file.choose(),header = TRUE)
data
head(data)
install.packages("survival")
install.packages("survminer")
#load necessary library
library(survival)
library(survminer)

age=data$age
age
anemia=data$anaemia
anemia
creatinine_phosphokinase=data$creatinine_phosphokinase
creatinine_phosphokinase
diabetes=data$diabetes
diabetes
ejection_fraction=data$ejection_fraction
ejection_fraction
high_blood_pressure=data$high_blood_pressure
high_blood_pressure
platelets=data$platelets
platelets
creatinine=data$serum_creatinine
creatinine
sodium=data$serum_sodium
sodium
sex=data$sex
sex
```

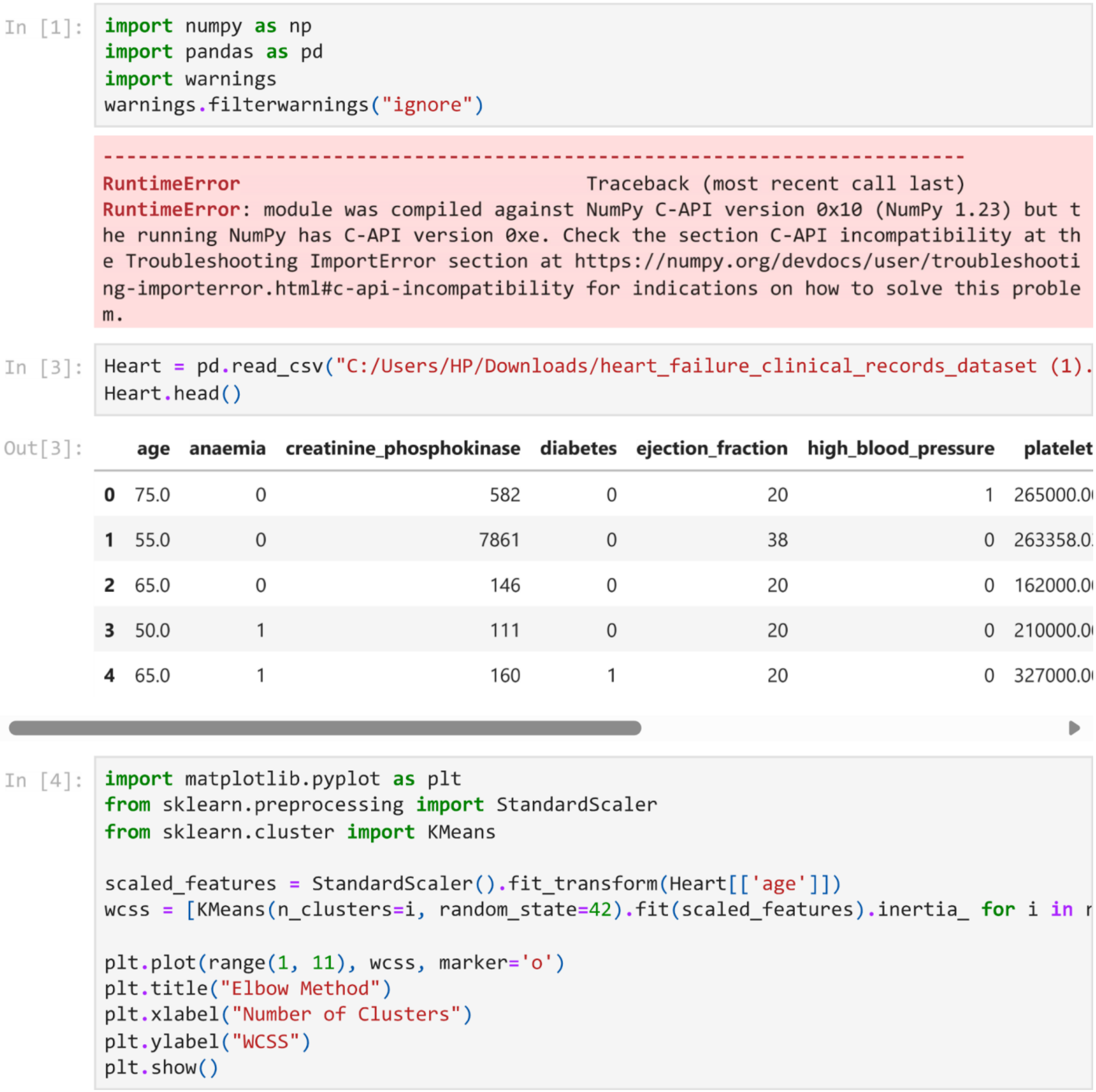
```
smoking=data$smoking
smoking
time=data$time
time
DEATH_EVENT=data$DEATH_EVENT
DEATH_EVENT
summary(dada)

#load necessary library
library(dplyr)
quartiles=quantile(age,probs = c(0,0.25,0.5,0.75,1),na.rm = T)
quartiles
Age_group=cut(data$age,breaks = quartiles,include.lowest = T,labels = c("40-52","53-64","65-73","74-95"))
Age_group
print(data)

##non parametric test of data
# To check association of SMOING AND DEATH_EVENT
# Convert relevant variables to factors
data$smoking = as.factor(data$smoking)
data$smoking
data$DEATH_EVENT = as.factor(data$DEATH_EVENT)
data$DEATH_EVENT
# Create a contingency table for Smoking and Death Event
contingency_table=table(data$smoking,data$DEATH_EVEN)
contingency_table
# Perform the Chi-Square test
chi_square_test = chisq.test(contingency_table)
chi_square_test
--------------------------------------------------------------------------------
# To check association of anaemia AND DEATH_EVENT
# Convert relevant variables to factors
data$anaemia = as.factor(data$anaemia)
data$anaemia
data$DEATH_EVENT = as.factor(data$DEATH_EVENT)
data$DEATH_EVENT
# Create a contingency table for Anaemia and Death Event
contingency_table= table(data$anaemia,data$DEATH_EVENT)
contingency_table
# Perform the Chi-Square test
chi_square_test = chisq.test(contingency_table)
chi_square_test
---------------------------------------------------------------------------
# To check association of diabetes AND DEATH_EVENT
# Convert relevant variables to factors
data$diabetes = as.factor(data$diabetes)
data$diabetes
data$DEATH_EVENT = as.factor(data$DEATH_EVENT)
data$DEATH_EVENT
# Create a contingency table for diabetes and Death Event
contingency_table= table(data$diabetes,data$DEATH_EVENT)
contingency_table
# Perform the Chi-Square test
chi_square_test = chisq.test(contingency_table)
chi_square_test
---------------------------------------------------------------------------
# To check association of sex AND DEATH_EVENT
# Convert relevant variables to factors
data$sex = as.factor(data$sex)
data$sex
data$DEATH_EVENT = as.factor(data$DEATH_EVENT)
data$DEATH_EVENT
# Create a contingency table for sex and Death Event
```

```
contingency_table= table(data$sex,data$DEATH_EVENT)
contingency_table
# Perform the Chi-Square test
chi_square_test = chisq.test(contingency_table)
chi_square_test
-------------------------------------------------------------------------------------

# To check association of HBP AND DEATH_EVENT
# Convert relevant variables to factors
data$high_blood_pressure = as.factor(data$high_blood_pressure)
data$high_blood_pressure
data$DEATH_EVENT = as.factor(data$DEATH_EVENT)
data$DEATH_EVENT
# Create a contingency table for HBP and Death Event
contingency_table1=
table(data$high_blood_pressure,data$DEATH_EVENT)
contingency_table1
# Perform the Chi-Square test
chi_square_test1 = chisq.test(contingency_table1)
chi_square_test1
-------------------------------------------------------------------------------------

# To check association of AGE GROUP AND DEATH_EVENT
# Convert relevant variables to factors
data$Age_group = as.factor(Age_group)
data$Age_group
data$DEATH_EVENT = as.factor(data$DEATH_EVENT)
data$DEATH_EVENT
# Create a contingency table for Age_group and Death Event
contingency_table1=
table(data$Age_group,data$DEATH_EVENT)
contingency_table1
```

```
# Perform the Chi-Square test
chi_square_test1 = chisq.test(contingency_table1)
chi_square_test1
--------------------------------------------------------------------------------
# Create a survival object
surv_obj = Surv(time, DEATH_EVENT)
surv_obj

# 1) Fit survival curves for anemia vs. No_anemia
fit_ane = survfit(surv_obj ~ anemia, data = data)
fit_ane
# Plot the survival curves
plot(
  fit_ane ,
  col = c("green", "red"),
  lty = 1:2,                  # Line types for the groups
  xlab = "time(days)",
  ylab = "Survival Probability",
  main = "Survival Curves: anemia vs No_anemia ")
# Add a legend
legend(
  "bottomleft",
  legend = c(" No_anemia", "Yes_anemia"),
  col = c("green", "red"),
  lty = 1:2)
# Perform the log-rank test for anemia
logrank_ane = survdiff(surv_obj ~ anemia, data = data)
logrank_ane
-------------------------------------------------------------------------------
# 2) Fit survival curves for smokers vs. Non_smokers
fit_smo = survfit(surv_obj ~ smoking, data = data)
fit_smo
```

```
# Plot the survival curves
plot( fit_smo ,
  col = c("green", "red"),
  lty = 1:2,                # Line types for the groups
  xlab = "time(days)",
  ylab = "Survival Probability",
  main = "Survival Curves: Smoker vs Non_smoker ")
# Add a legend
legend("bottomleft",
  legend = c(" No_smoker", "Yes_smoker"),
  col = c("green", "red"),
  lty = 1:2)
# Perform the log-rank test for smoking
logrank_smo = survdiff(surv_obj ~ smoking, data = data)
logrank_smo
---------------------------------------------------------------------------------
# 3) Fit survival curves for high_blood_pressure vs. Normal_high_blood_pressure
fit_HBP = survfit(surv_obj ~ high_blood_pressure, data = data)
fit_HBP
# Plot the survival curves
plot(fit_HBP ,
  col = c("green", "red"), # Colors for the groups
  lty = 1:2,                # Line types for the groups
  xlab = "time(days)",
  ylab = "Survival Probability",
  main = "Survival Curves: Normal_high_blood_pressure vs high_blood_pressure ")
# Add a legend
legend( "bottomleft",
  legend=c("Normal_high_blood_pressure",
"high_blood_pressure"),
  col = c("green", "red"),
  lty = 1:2)
# Perform the log-rank test for high_blood_pressure
logrank_HBP = survdiff(surv_obj ~ high_blood_pressure, data =
data)
logrank_HBP
-------------------------------------------------------------------------------
# 4) Fit survival curves for Male vs. Female
fit_sex = survfit(surv_obj ~ sex, data = data)
fit_sex
# Plot the survival curves
plot(fit_sex ,
  col = c("green", "red"), # Colors for the groups
  lty = 1:2,                  # Line types for the groups
  xlab = "time(days)",
  ylab = "Survival Probability",
  main = "Survival Curves: Female vs male ")
# Add a legend
legend("bottomleft",
  legend = c(" Female", "Male"),
  col = c("green", "red"),
  lty = 1:2)
# Perform the log-rank test for Sex
logrank_sex = survdiff(surv_obj ~ sex, data = data)
logrank_sex
-------------------------------------------------------------------------------
# 5) Fit survival curves for
fit_Age_group = survfit(surv_obj ~ Age_group, data = data)
fit_Age_group
# Plot the survival curves Age_group
plot(fit_Age_group ,
  col = c("green", "blue","black","red"), # Colors for the groups
  lty = 1:2,                # Line types for the groups
  xlab = "time(days)",
  ylab = "Survival Probability",
  main = "Survival Curves: Age_group ")
# Add a legend
legend( "bottomleft",
  legend = c("40-52 ", "53-64","65-73","74-95"),
  col = c("green","blue","black","red" ),
  lty = 1:2)
# Perform the log-rank test for Age_group
logrank_Age_group = survdiff(surv_obj ~ Age_group, data = data)
logrank_Age_group
--------------------------------------------------------------------------------
# Test proportional hazards assumption
cox_model = coxph(surv_obj ~ Age_group+ diabetes+platelets +
sex+high_blood_pressure+creatinine+sodium+smoking+anaemia+
creatinine_phosphokinase+ejection_fraction, data = data)
cox_model
#summary of model
summary(cox_model)
# Schoenfeld residuals test
test_ph=cox.zph(cox_model)
test_ph
cox_model1=coxph(surv_obj ~ Age_group + high_blood_pressure
+ creatinine + sodium + anaemia +
 creatinine_phosphokinase + ejection_fraction,  data = data)
cox_model1

#summary of model1
summary(cox_model1)
```

Python



In [1]:

```
import numpy as np
import pandas as pd
import warnings
warnings.filterwarnings("ignore")
```

```
---------------------------------------------------------------------------
RuntimeError                              Traceback (most recent call last)
RuntimeError: module was compiled against NumPy C-API version 0x10 (NumPy 1.23) but t
he running NumPy has C-API version 0xe. Check the section C-API incompatibility at th
e Troubleshooting ImportError section at https://numpy.org/devdocs/user/troubleshooti
ng-importerror.html#c-api-incompatibility for indications on how to solve this proble
m.
```

In [3]:

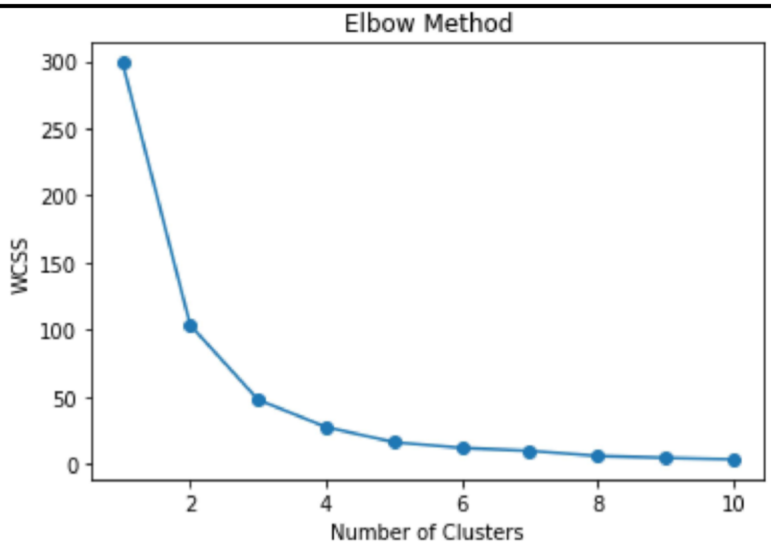
```
Heart = pd.read_csv("C:/Users/HP/Downloads/heart_failure_clinical_records_dataset (1).
Heart.head()
```

Out[3]:

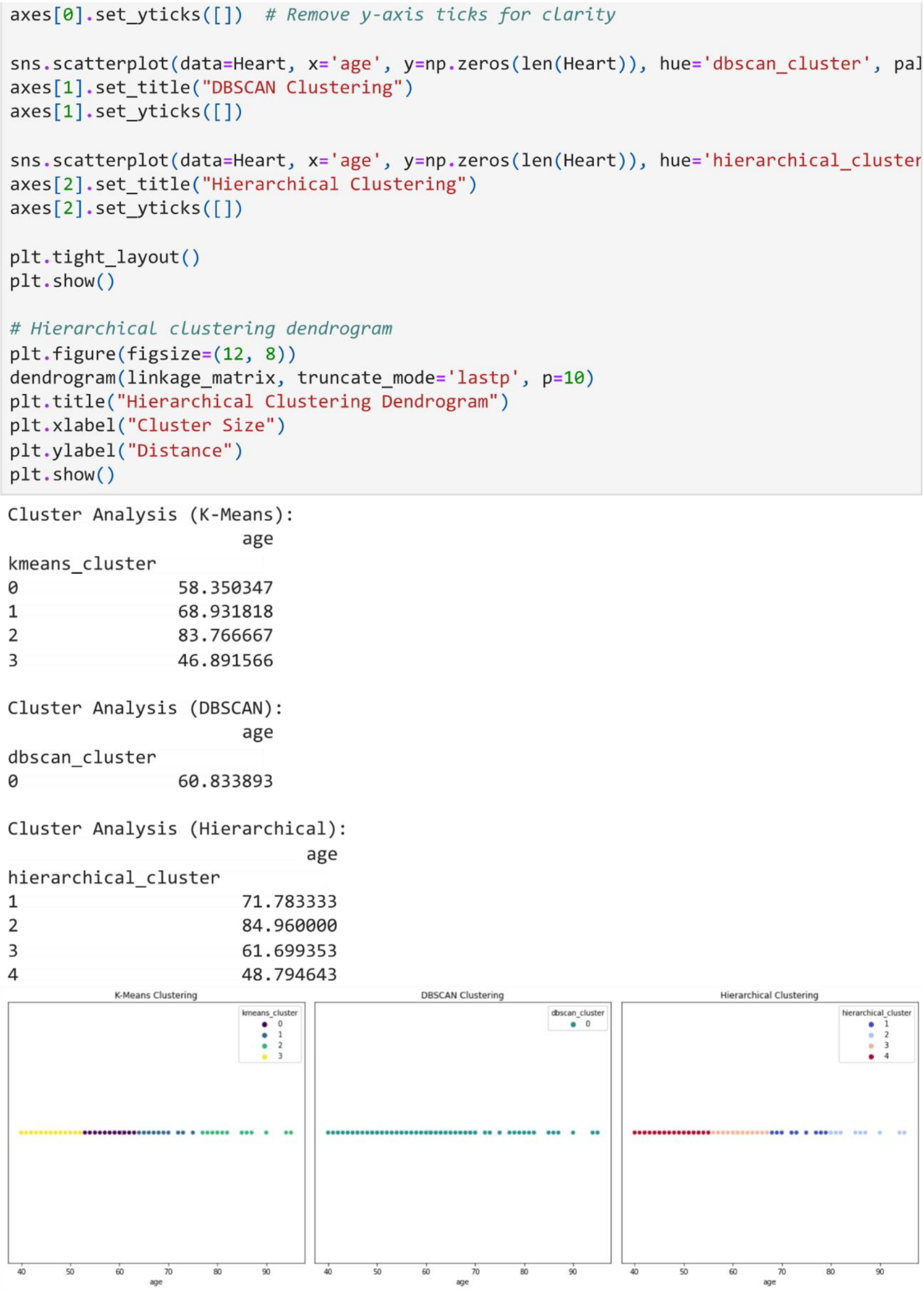
	age	anaemia	creatinine_phosphokinase	diabetes	ejection_fraction	high_blood_pressure	platelet
0	75.0	0	582	0	20	1	265000.0
1	55.0	0	7861	0	38	0	263358.0
2	65.0	0	146	0	20	0	162000.0
3	50.0	1	111	0	20	0	210000.0
4	65.0	1	160	1	20	0	327000.0

In [4]:

```
import matplotlib.pyplot as plt
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import KMeans

scaled_features = StandardScaler().fit_transform(Heart[['age']])
wcss = [KMeans(n_clusters=i, random_state=42).fit(scaled_features).inertia_ for i in r

plt.plot(range(1, 11), wcss, marker='o')
plt.title("Elbow Method")
plt.xlabel("Number of Clusters")
plt.ylabel("WCSS")
plt.show()
```

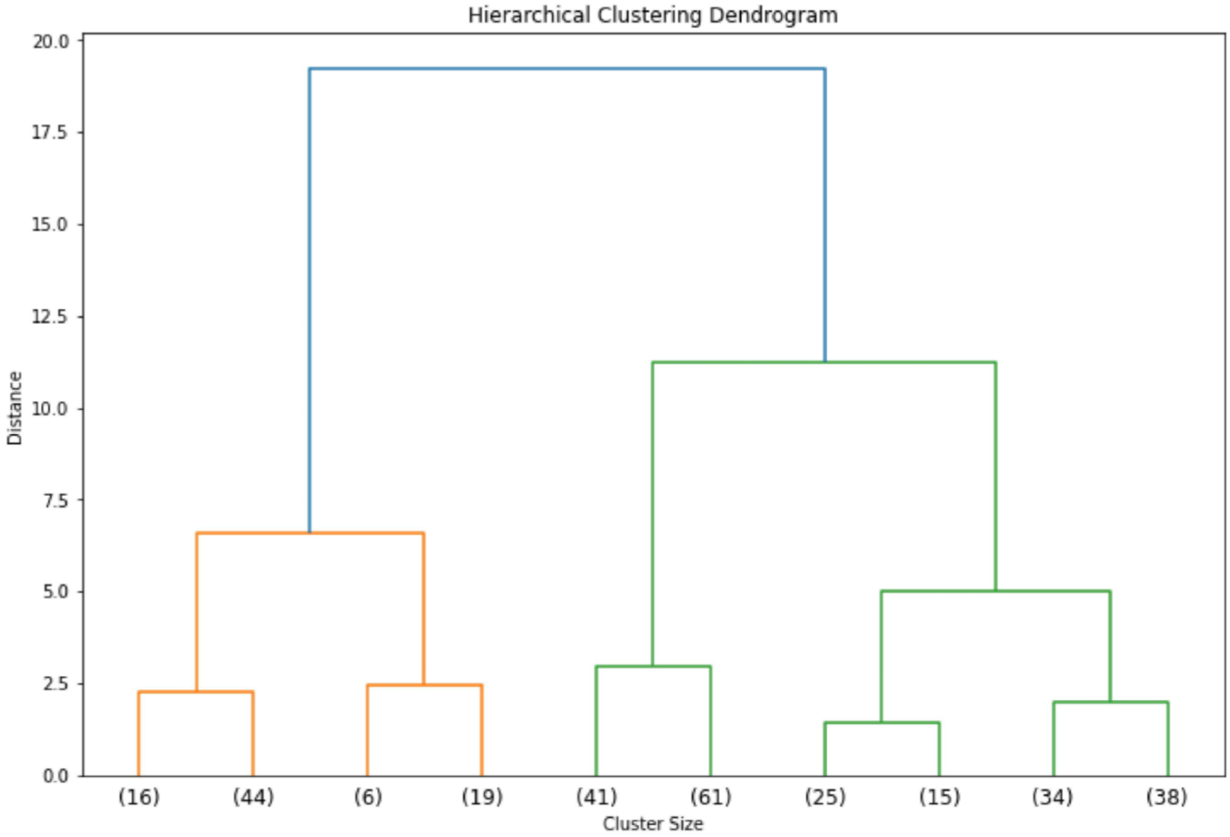


In [5]:
```
import pandas as pd
import numpy as np
from sklearn.cluster import KMeans, DBSCAN
from sklearn.preprocessing import StandardScaler
from scipy.cluster.hierarchy import dendrogram, linkage, fcluster
import matplotlib.pyplot as plt
import seaborn as sns

# Assuming Heart is a valid DataFrame
features = Heart[['age']]

# Standardizing the feature
scaler = StandardScaler()
scaled_features = scaler.fit_transform(features)

# K-Means Clustering
kmeans = KMeans(n_clusters=4, random_state=42, n_init=10)
Heart['kmeans_cluster'] = kmeans.fit_predict(scaled_features)

# DBSCAN Clustering
dbscan = DBSCAN(eps=0.5, min_samples=5)
Heart['dbscan_cluster'] = dbscan.fit_predict(scaled_features)

# Hierarchical Clustering
linkage_matrix = linkage(scaled_features, method='ward')
Heart['hierarchical_cluster'] = fcluster(linkage_matrix, t=4, criterion='maxclust')

# Display cluster analysis
print("Cluster Analysis (K-Means):")
print(Heart.groupby('kmeans_cluster')[['age']].mean())

print("\nCluster Analysis (DBSCAN):")
print(Heart.groupby('dbscan_cluster')[['age']].mean())

print("\nCluster Analysis (Hierarchical):")
print(Heart.groupby('hierarchical_cluster')[['age']].mean())

# Visualization of clustering results
fig, axes = plt.subplots(1, 3, figsize=(18, 6))

sns.scatterplot(data=Heart, x='age', y=np.zeros(len(Heart)), hue='kmeans_cluster', pal
axes[0].set_title("K-Means Clustering")
```



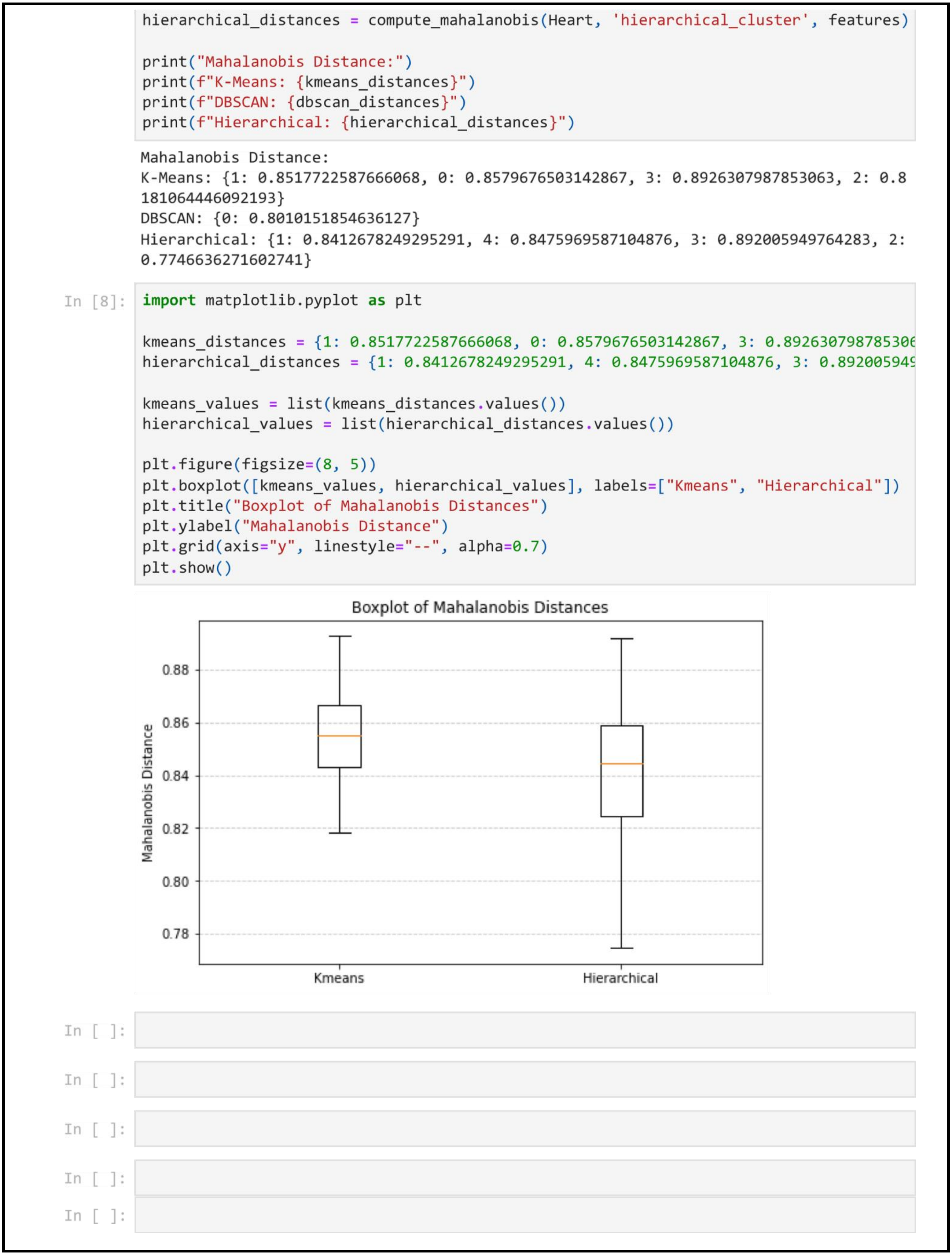
```
axes[0].set_yticks([])  # Remove y-axis ticks for clarity

sns.scatterplot(data=Heart, x='age', y=np.zeros(len(Heart)), hue='dbscan_cluster', pal
axes[1].set_title("DBSCAN Clustering")
axes[1].set_yticks([])

sns.scatterplot(data=Heart, x='age', y=np.zeros(len(Heart)), hue='hierarchical_cluster
axes[2].set_title("Hierarchical Clustering")
axes[2].set_yticks([])

plt.tight_layout()
plt.show()

# Hierarchical clustering dendrogram
plt.figure(figsize=(12, 8))
dendrogram(linkage_matrix, truncate_mode='lastp', p=10)
plt.title("Hierarchical Clustering Dendrogram")
plt.xlabel("Cluster Size")
plt.ylabel("Distance")
plt.show()
```

```
Cluster Analysis (K-Means):
                      age
kmeans_cluster
0               58.350347
1               68.931818
2               83.766667
3               46.891566

Cluster Analysis (DBSCAN):
                      age
dbscan_cluster
0               60.833893

Cluster Analysis (Hierarchical):
                            age
hierarchical_cluster
1                     71.783333
2                     84.960000
3                     61.699353
4                     48.794643
```



In [6]:

```
from scipy.spatial.distance import mahalanobis
from numpy.linalg import pinv
import numpy as np

features = ['age']  # Updating to reflect clustering on 'age'

# Ensure the features exist in the DataFrame
if not set(features).issubset(Heart.columns):
    raise ValueError(f"Missing columns in the DataFrame: {set(features) - set(Heart.co

# Compute Mahalanobis distance with pseudo-inverse
def compute_mahalanobis(df, cluster_column, features, reg=1e-6):
    results = {}
    for cluster in df[cluster_column].unique():
        if cluster == -1:  # Ignore DBSCAN outliers
            continue

        cluster_data = df[df[cluster_column] == cluster][features]
        mean_vector = cluster_data.mean().values
        covariance_matrix = np.cov(cluster_data.values.T) + np.eye(len(features)) * re

        # Compute Mahalanobis distance for each point in the cluster
        distances = cluster_data.apply(lambda row: mahalanobis(row.values, mean_vector

        # Store the mean distance per cluster
        results[cluster] = distances.mean()

    return results

# Compute Mahalanobis distances for clusters
kmeans_distances = compute_mahalanobis(Heart, 'kmeans_cluster', features)
dbscan_distances = compute_mahalanobis(Heart, 'dbscan_cluster', features)
```



```
hierarchical_distances = compute_mahalanobis(Heart, 'hierarchical_cluster', features)

print("Mahalanobis Distance:")
print(f"K-Means: {kmeans_distances}")
print(f"DBSCAN: {dbscan_distances}")
print(f"Hierarchical: {hierarchical_distances}")
```

```
Mahalanobis Distance:
K-Means: {1: 0.8517722587666068, 0: 0.8579676503142867, 3: 0.8926307987853063, 2: 0.8
181064446092193}
DBSCAN: {0: 0.8010151854636127}
Hierarchical: {1: 0.8412678249295291, 4: 0.8475969587104876, 3: 0.892005949764283, 2:
0.7746636271602741}
```

In [8]:

```
import matplotlib.pyplot as plt

kmeans_distances = {1: 0.8517722587666068, 0: 0.8579676503142867, 3: 0.892630798785306
hierarchical_distances = {1: 0.8412678249295291, 4: 0.8475969587104876, 3: 0.892005949

kmeans_values = list(kmeans_distances.values())
hierarchical_values = list(hierarchical_distances.values())

plt.figure(figsize=(8, 5))
plt.boxplot([kmeans_values, hierarchical_values], labels=["Kmeans", "Hierarchical"])
plt.title("Boxplot of Mahalanobis Distances")
plt.ylabel("Mahalanobis Distance")
plt.grid(axis="y", linestyle="--", alpha=0.7)
plt.show()
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: