

**Heart failure predictions**

**Using machine learning**

**Title Page**

* Title: Heart failure predictions using machine learning
* Subtitle: "Leveraging Clinical Data and Machine Learning for Early Detection and Prediction of Heart Failure Risk"
* Team Members: Nukapeyyi Latha kumari,

Sathi Sanvitha Reddy,

Galla Jyotshan,

Karakavalasa. Likitha

* Date:30-11-2024

**1.Abstract**

**Background:**

Heart failure (HF) is a leading cause of morbidity and mortality worldwide. Traditional methods for diagnosing and predicting heart failure often involve a combination of clinical symptoms, patient history, and physical examinations, which can be subjective and timeconsuming. In recent years, machine learning (ML) has emerged as a powerful tool in healthcare for improving diagnostic accuracy and predicting the onset and progression of heart failure. By analyzing large, complex datasets, ML algorithms can uncover patterns in clinical, demographic, and medical data that are difficult for humans to detect. These models enable more accurate, early identification of patients at risk for heart failure, facilitating timely interventions. Additionally, ML techniques can improve risk assessment, help guide personalized treatment plans, and reduce hospital readmissions, ultimately improving patient outcomes.

**Dataset Characteristics:**The "Heart Failure Prediction" dataset contains several attributes related to patients' health conditions and clinical measurements, which can be used to predict heart failure. Key attributes in the dataset include:

**Age**: The age of the patient.

* **Gender**: The gender of the patient (e.g., Male, Female).
* **Ejection Fraction**: A measure of how much blood the heart pumps with each contraction, indicating heart function.
* **Serum Creatinine**: The level of creatinine in the blood, used to assess kidney function.
* **Blood Pressure**: The patient's systolic and diastolic blood pressure readings.
* **Platelets**: The concentration of platelets in the blood, related to blood clotting.
* **Smoking History**: Whether the patient has a history of smoking (Yes/No).
* **Diabetes**: Whether the patient has diabetes (Yes/No).
* **Heart Disease History**: The history of heart disease (Yes/No).
* **Target Variable**: A binary variable indicating whether the patient experienced heart failure (1 = Yes, 0 = No).

By applying machine learning models such as Support Vector Machine to this dataset, the goal is to predict the likelihood of heart failure onset in patients. This approach aims to improve early detection, facilitate preventive measures, and support healthcare providers in making more informed decisions, ultimately reducing the burden of heart failure on both patients and the healthcare system.

**2.Methodology**

The methodology for predicting heart failure involves several key steps, from data collection and preprocessing to model training and evaluation. Below is a detailed description of the methodology for building and deploying a heart failure prediction model using machine learning techniques:

# 1. Data Collection

The first step involves gathering relevant data related to heart failure. In this study, we utilize the "Heart Failure Prediction" dataset, which includes clinical attributes and patient history. Key features in this dataset include patient demographics (age, gender), medical history (e.g., smoking, diabetes, heart disease), and clinical measurements (e.g., ejection fraction, blood pressure, serum creatinine). The target variable is the binary classification of heart failure (1 for heart failure, 0 for no heart failure).

# 2. Data Preprocessing

Data preprocessing ensures that the dataset is clean and suitable for machine learning algorithms:

* **Missing Data Handling**: Any missing or incomplete values in the dataset are addressed. Methods like imputation (filling missing values with mean, median, or mode) or deletion of rows/columns with excessive missing data are used.
* **Normalization and Scaling**: Continuous variables such as blood pressure, serum creatinine, and age are scaled or normalized to ensure that all features have comparable ranges. Techniques like Min-Max scaling or Standardization (Z-score) are commonly applied.
* **Encoding Categorical Variables**: Categorical variables such as gender, smoking history, and heart disease history are converted into numerical format using techniques like one-hot encoding or label encoding.
* **Feature Engineering**: Additional features may be created based on domain knowledge, such as creating new binary features for risk categories or aggregating existing features.

# 3. Exploratory Data Analysis (EDA)

EDA is used to understand the dataset's structure and relationships between features:

* **Descriptive Statistics**: Summary statistics like mean, median, variance, and standard deviation are computed for numerical features.
* **Visualizations**: Histograms, scatter plots, and correlation matrices are generated to identify patterns, outliers, and correlations between features.
* **Class Imbalance Check**: The distribution of the target variable (heart failure vs. no heart failure) is examined. If the dataset is imbalanced, techniques like oversampling (SMOTE) or under sampling may be applied to balance the classes.

# 4. Model Selection

Various machine learning algorithms are considered for predicting heart failure:

* **Logistic Regression**: A fundamental model for binary classification problems.
* **Decision Trees**: A tree-based model that splits data based on feature values, making it easy to interpret.
* **Random Forest**: An ensemble learning method that combines multiple decision trees to improve accuracy and reduce overfitting.
* **Gradient Boosting Machines (GBM)**: Techniques like Boost or Light that improve model performance by sequentially correcting errors made by previous trees.
* **Neural Networks**: Deep learning models that capture complex non-linear relationships in the data.

# 5. Model Training

The dataset is split into training and testing sets, typically using an 80/20 or 70/30 split. The training data is used to train the selected models. Cross-validation (e.g., K-fold crossvalidation) is employed to assess the model's generalizability and reduce overfitting. Hyperparameter tuning, such as using grid search or random search, is applied to find the optimal model settings.

# 6. Model Evaluation

The trained models are evaluated based on various performance metrics:

* **Accuracy**: The proportion of correct predictions made by the model.
* **Precision**: The proportion of true positives out of all positive predictions (important in avoiding false positives).
* **Recall (Sensitivity)**: The proportion of true positives out of all actual positives (important in identifying heart failure patients).

**F1 Score**: The harmonic means of precision and recall, providing a balanced measure of model performance.

* **Area Under the Curve (AUC-ROC)**: Measures the model's ability to distinguish between classes, with a higher AUC indicating better model performance.
* **Confusion Matrix**: Provides a summary of prediction results, highlighting false positives, false negatives, true positives, and true negatives.

# 7. Model Comparison and Selection

Several models are compared based on evaluation metrics, and the best-performing model is selected for further use. If necessary, ensemble methods (e.g., voting classifiers) are used to combine the strengths of multiple models.

# 8. Model Deployment

Once the best model is identified, it is deployed in a clinical setting or decision support system. The model is integrated with healthcare systems for real-time prediction and decision-making. A user-friendly interface is developed to allow healthcare providers to input patient data and receive a heart failure risk prediction.

# 9. Continuous Monitoring and Updating

The model's performance is continuously monitored to ensure it remains accurate over time. As new data becomes available or as healthcare practices evolve, the model is retrained and updated periodically to maintain its relevance and effectiveness.

**Outcomes:**

The **SVM (Support Vector Machine)** model demonstrated superior performance with an accuracy of **89.33%** on the test dataset, making it the best-suited algorithm for predicting heart failure.

**The developed application provides:**

* A simple and interactive interface for loan officers and applicants to input data.
* Visualizations to aid in understanding feature distributions.
* Accurate predictions of likelihood.
*  **Vector Machine (SVM)** was chosen due to its effectiveness in handling complex, nonlinear relationships in data. The model was trained using an **RBF kernel**, which allows it to classify patients based on patterns in the high-dimensional feature space.
*  **Tuning**: Key parameters such as **C** (penalty parameterr) And **gamma** (kernel parameter) was optimized using grid search to achieve the best possible performance.

1. **Introduction**

Heart failure is a chronic condition where the heart cannot pump blood effectively, leading to reduced oxygen supply to the body's organs. Predicting heart failure is a vital step in improving patient outcomes, as early intervention can prevent severe complications and improve quality of life. Advances in machine learning and artificial intelligence (AI) have significantly enhanced our ability to predict heart failure with greater accuracy.

This project explores the development and application of ML models, specifically the **Support Vector Machine (SVM)** algorithm, to predict heart failure with high accuracy. By leveraging a carefully pre-processed dataset and advanced modelling techniques, the project demonstrates the potential of ML to enhance efficiency, reduce biases,

**3.2 Objectives**

**Develop a Machine Learning-Based Prediction Model**

a machine learning-based prediction model using **Support Vector Machine (SVM)** for heart failure prediction. The model's performance is evaluated using metrics such as accuracy, precision, recall, F1 score, and AUC-ROC (Area Under the Receiver Operating Characteristic Curve). Hyperparameter tuning ensures the model is optimized for better accuracy. Lastly, the model can be deployed for real-time predictions.

You can extend this approach by exploring other machine learning algorithms like Random Forest, Boost, or Neural Networks, and testing different feature engineering techniques to improve performance further.

* + **Hyperparameter Tuning:** Optimize models to maximize accuracy and minimize false negatives (incorrectly rejecting eligible applications). **Design a User-Friendly Interface**
  + **Web-Based Application:** Develop an interactive application using **Streamlet**, a versatile open-source framework for machine learning projects.
  + **Input Features:** Enable users to input loan application details, such as applicant income, co-applicant income, credit history, loan amount, and property area.
  + **Real-Time Predictions:** Provide real-time loan approval predictions with userfriendly outputs.
  + **Data Visualizations:** Include intuitive visualizations to help users understand data trends, such as applicant income distributions and loan approval ratios.

**3.3 Scope of the Project**

**Using Publicly Available Datasets**

• The project leverages the heart.csv  **dataset**, which contains 918**records** of loan applications. The dataset includes key features such as age, Gender,Pistoling history ejection fraction, Diabetes,taraget value(0=No)(1=Yes) which serve as the basis for training and testing the machine learning models.

**Implementing Four Machine Learning Algorithms**

Four distinct machine learning algorithms are applied to predict loan approval outcomes and ensure a comprehensive approach to prediction:

* + - **Logistic Regression:** A simple statistical model for binary classification, useful for understanding relationships between features and loan approval likelihood. o **Random Forest:** An ensemble learning method that aggregates multiple Accurcy:83.45 o Decision Tree: The Accurcy:84.72%
    - **Support Vector Machine (SVM):** A powerful algorithm that separates data points using hyperplanes, suitable for high-dimensional feature spaces. Accurcy:89.36

• Each model is evaluated based on **accuracy**, **precision**, **recall**, and other metrics to determine the best-performing algorithm for loan approval prediction.

**4. Literature Review**

**Manual Loan Approval Processes**

Traditionally, loan approval processes were largely based on manual methods and predefined rules, which were often subjective and time-consuming. Financial institutions would assess loan applications using a variety of factors, including the applicant's credit history, income, employment status, and other demographic information. However, these manual processes were prone to human error, inconsistent decision-making, and often took longer to complete.

Approval decisions were made by loan officers, who would evaluate each application individually. This process, while accurate in some cases, lacked efficiency and could be biased, especially when evaluating applicants with similar profiles. Additionally, human decision-makers often struggled to identify subtle patterns in large volumes of data that could influence loan approval.

**Challenges with Traditional Credit Scoring**

Traditional credit scoring models, such as FICO, evaluate an applicant's creditworthiness based on factors like credit history, outstanding debt, and payment history. While these models provide a quantitative approach, they do not consider many other potential factors that could impact loan approval decisions, such as an applicant’s overall financial behaviour, co-applicant’s income, or even social factors like geographic location. Moreover, traditional models may not always adapt well to changing financial conditions or new applicant profiles, limiting their accuracy and effectiveness.

**Machine Learning in Heart Failure Prediction :**

Machine learning (ML) plays a transformative role in predicting heart failure (HF), offering advanced methods for analyzing complex datasets, identifying patterns, and improving early detection. This literature review focuses on the theoretical principles and methodologies behind ML applications in HF prediction. Support Vector Machines (SVM) are a popular machine learning algorithm used for classification and regression tasks. In heart failure (HF) prediction, SVMs are particularly effective in handling high-dimensional datasets, identifying nonlinear relationships, and classifying patients into risk categories.

**Limitations of Traditional Heart failure Methods**

their place in clinical practice, they are often slow, subjective, and limited in predictive capability. Advancements in technology, particularly the use of machine learning and AI, can address many of these limitations by providing more accurate, timely, and comprehensive predictions, which could significantly improve patient outcomes. Additionally, **interpretability** of some machine learning models, particularly more complex ones like **Random Forest** and **SVM**, can be difficult. This lack of transparency may limit trust in the model's decisions, especially among loan officers and applicants who may prefer clear, understandable reasons for loan approval or rejection.

**Advantages of Machine Learning Models**

Despite these challenges, machine learning has proven to be an effective tool for automating and improving the process. By utilizing algorithms that evaluate a wider range of variables and learn from past data, financial institutions can make more accurate, objective, and faster decisions, ultimately benefiting both the institution and the applicant. Furthermore, ML models can help in **reducing human biases**, ensure **fairer loan decisions**, and improve the **consistency** of approval processes.

As the data available for loan applications continues to grow, the use of machine learning will likely become more integral in optimizing loan approval workflow and predicting loan defaults more accurately.

**4.2 Use of Machine Learning in Heart failure Prediction**

Machine learning (ML) has become a transformative tool in healthcare, particularly for predicting and managing chronic conditions like heart failure. By leveraging large datasets, advanced algorithms, and computational power, ML can provide accurate, personalized, and timely predictions for heart failure risk. Here's how machine learning is being used to predict heart failure:

# 1. Early Detection of Heart Failure

* **Predicting Onset**: Machine learning models can analyse patient data (e.g., medical history, lab results, imaging data) to identify early signs of heart failure before symptoms become apparent. Early intervention can significantly improve outcomes.
* **Predictive Models**: Algorithms like **Random Forest**, **Support Vector Machine (SVM)**, and **Logistic Regression** can be trained on historical patient data to predict the likelihood of developing heart failure in the future based on existing health conditions.

# 2. Risk Stratification and Prognosis

**Identifying High-Risk Patients**: ML models can stratify patients based on their risk levels for heart failure. For example, they can predict which patients are at a higher risk of hospitalization, re-admission, or worsening of heart failure symptoms.

• **Survival Prediction**: By analyzing factors like ejection fraction, blood pressure, comorbidities, and lab results, ML algorithms can predict patient survival rates and help in tailoring personalized treatment plans.

# 3. Improving Diagnostic Accuracy

* **Clinical Data Integration**: ML can analyse multiple forms of clinical data, such as electronic health records (EHRs), echocardiogram images, ECG signals, and lab results, for a more comprehensive diagnosis. This reduces human error and subjectivity in diagnosis.
* **Image Processing**: ML algorithms, particularly **Convolutional Neural Networks (CNNs)**, are used to process and interpret medical imaging (e.g., echocardiograms, CT scans) to detect structural heart abnormalities indicative of heart failure.

# 4. Personalizing Treatment Plans

* **Tailored Interventions**: Based on a patient's unique profile, ML models can recommend personalized treatment plans. For instance, they can suggest adjustments in medications, lifestyle changes, or monitoring plans that could prevent the progression of heart failure.
* **Medication Management**: ML models can help doctors predict the most effective medications based on a patient's condition, genetics, and medical history, optimizing treatment outcomes.

# 5. Real-Time Monitoring and Predictive Alerts

* **Continuous Monitoring**: ML can be integrated with wearable devices and remote monitoring systems to continuously track patients' vital signs (e.g., heart rate, blood pressure, oxygen levels). ML algorithms can analyse this real-time data and predict potential complications, alerting healthcare providers or patients in time to prevent emergencies.
* **Predicting Decompensated Heart Failure**: Using real-time data, ML models can detect early signs of decompensated heart failure (when the heart can no longer pump enough blood to meet the body's needs) and trigger alerts for immediate intervention.

# 6. Data-Driven Insights

* **Big Data Analysis**: ML can analyse vast amounts of health data from various sources, such as hospital records, patient surveys, sensor data, and genetic information. By finding patterns within this data, ML can identify new risk factors, disease pathways, and potential interventions that were previously unknown.
* **Health Risk Factors Identification**: Through machine learning models, healthcare providers can identify new biomarkers or risk factors that are associated with the development or progression of heart failure.

# 7. Reducing Healthcare Costs

* **Optimizing Resource Allocation**: By predicting which patients are most likely to require hospitalization or intensive care, ML can help healthcare systems allocate resources more efficiently, improving patient outcomes while reducing costs.
* **Preventing Readmissions**: Machine learning can predict the likelihood of patient readmissions, helping healthcare providers implement interventions to reduce these occurrences and lower the overall cost of care.

# 8. Predicting Heart Failure Related Complications

* **Comorbidities Prediction**: Heart failure often occurs alongside other chronic conditions such as diabetes, hypertension, or kidney disease. ML models can predict the likelihood of comorbidities developing and help in preventive care strategies.
* **Monitoring Disease Progression**: Machine learning can predict the progression of heart failure by monitoring key metrics over time and alerting clinicians about the worsening condition.

# 9. Handling Imbalanced Data

⚫ **Improving Predictions with Rare Events**: One challenge in heart failure prediction is that the data is often imbalanced (i.e., the number of patients with heart failure is much lower than those without it). ML techniques like **SMOTE (Synthetic Minority Over-sampling Technique)**, **ensemble methods**, or **anomaly detection** can help mitigate this imbalance, improving the model's ability to predict rare events like heart failure.

# 10. Using Natural Language Processing (NLP)

⚫ **Extracting Information from Unstructured Data**: In many healthcare systems, valuable patient information is stored in unstructured text formats (e.g., clinical notes, discharge summaries). **Natural Language Processing (NLP)** techniques can extract relevant data from these text fields, such as mentions of symptoms, previous treatments, and test results, to help predict heart failure risk.

# Conclusion

Machine learning is revolutionizing heart failure prediction by enabling earlier detection, more accurate risk stratification, personalized treatments, and real-time monitoring. As healthcare systems adopt these technologies, the potential to improve patient outcomes and

reduce the burden of heart failure is substantial. ML can help clinicians make data-driven decisions, ultimately improving the quality of care and enhancing the patient's quality of life

**4.3 Related Work**

**Studies Using Machine Learning for heart failure Prediction**

Several studies have explored the use of machine learning algorithms for predicting These studies demonstrate the effectiveness of machine learning techniques in automating the loan evaluation process and improving decision-making accuracy. Some notable examples include:

**Support Vector Machines (SVM):** A study by R. Sharmila et al, A conceptual method to enhance the prediction of heart diseases using the data techniques. SVM in parallel fashion SVM provides better and efficient accuracy of 85% and 82.35%. SVM in parallel fashion gives better accuracy than sequential SVM.

**Conclusion:** Support Vector Machines (SVM) are particularly well-suited for heart failure approval prediction tasks due to their ability to handle complex, high-dimensional datasets and maintain generalization. In comparison to other algorithms such as Logistic Regression and Random Forest, SVM has demonstrated superior accuracy in predicting loan approval outcomes, particularly when dealing with non-linearly separable data. These findings reinforce the effectiveness of SVM as a preferred model for loan approval prediction, offering robust performance and high accuracy.

**5. Dataset Overview**

### 5.1 Data Source

**Dataset Origin**

The dataset used for this project is the Heart Failure **Prediction Dataset** sourced from the **UCI Machine Learning Repository**, a trusted platform for hosting datasets used in various machine learning applications. This dataset is often employed for binary classification tasks, particularly.

These data sources are essential for training, validating, and testing machine learning models focused on heart failure prediction, readmission risk, or mortality. Access to such diverse datasets supports various research goals and provides valuable insightinto heart disease prediction.

**Ethical Considerations**

The dataset does not contain personally identifiable information (PII), ensuring that it complies with privacy regulations such as GDPR. The dataset is openly available for research and educational purposes and is provided with an open-access license to support transparent and responsible machine learning research.

**Dataset Structure**

* **Rows (Records):** 918 • **Columns (Features):** 12

· Rows **(918):**

* Each row corresponds to an individual patient.
* Represents the patient's features and whether they experienced the outcome (e.g., heart failure, mortality).

· Columns **(12):**

* Iinclude a mix of demographic, medical history, and lab test results.
* A single column is typically reserved for the target variable, which models predict (e.g., mortality or heart failure).

**Target Variable:**

**Loan Status: in**dictates whether a patient developed heart failure.

· 1 (Yes) - Patient experienced heart failure.

· 0 (No) - Patient did not experience heart failure.

**Limitations of the Dataset**

**Class Imbalance:** ·

Issue: If the target variable (e.g., heart failure occurrence) has an uneven distribution (e.g., far fewer cases of heart failure compared to non-cases), the model may become biased toward the majority class.

· Impact**:**

* Poor predictive performance for the minority class.

· Mitigation**:**

* Apply resampling techniques such as oversampling (e.g., SMOTE) or under sampling.
* Use metrics like **F1-score**, **ROC-AUC**, and **precision-recall curves** to evaluate model pperformance.

# Feature Correlation and Redundancy

* **Issue:** Some features may be highly correlated (e.g., ejection fraction and serum creatinine).
* **Impact:**

Redundant features can decrease model interpretability and computational efficiency.

* **Mitigation:**

o Use techniques like **PCA** (Principal Component Analysis) or **correlation analysis** to reduce redundancy.

|  |
| --- |
| Ejection Fraction |

|  |
| --- |
| Hypertension |

|  |
| --- |
| **Attribute** |

|  |
| --- |
| **Gender** |

|  |
| --- |
| Diabetes |

***5.2 Dataset Features***

For a heart failure prediction dataset with **12 features (columns)**, the typical dataset includes a combination of demographic, clinical, and laboratory features. Below is an overview of commonly found features.

|  |
| --- |
| Percentage of blood pumped out of the heart with each contraction, a key measure in heart failure. |

|  |
| --- |
| Platelet count in the blood (kilo platelets/mL), associated with clotting and cardiovascular health. |

|  |
| --- |
| Creatinine level in blood (mg/dL), a measure of kidney function. |

|  |
| --- |
| Indicates if the patient has high blood pressure. |

|  |
| --- |
| **Description** |

|  |
| --- |
| The gender of the applicant. |

|  |
| --- |
| Sodium level in blood (me/L), used to assess severity of heart failure. |

|  |
| --- |
| Indicates if the patient has diabetes. |

|  |
| --- |
| Serum Creatinine |

Here is a table outlining the attributes (features) in the Heart Failure **Prediction Dataset** along with their meanings:

|  |
| --- |
| Resting BP |

|  |
| --- |
| Outcome |

|  |
| --- |
| Platelets |

|  |
| --- |
| Serum Sodium |

|  |
| --- |
| Age |

|  |
| --- |
| Cholesterol |

|  |
| --- |
| Smoking |

**Key Insights into the Features Age:**

|  |
| --- |
| Resting systolic blood pressure (mmHg), an indicator of cardiovascular health. |

|  |
| --- |
| Target variable: Indicates whether the patient experienced heart failure or mortality. |

|  |
| --- |
| Patient's age in years. |

|  |
| --- |
| Total cholesterol in blood (mg/dL), related to cardiovascular risks. |

|  |
| --- |
| Whether the patient is a smoker. |

]1.Older patients typically have a higher risk of heart failure.

2. Can be used for stratification or age-specific predictions.

**Ejection Fraction:**

* 1. A critical feature in diagnosing and managing heart failure.
  2. Low values (<40%) indicate reduced cardiac function.

**Serum Creatinine & Serum Sodium:**

* 1. Reflect kidney function and electrolyte balance, which are often impacted in heart failure.
  2. Abnormal values may indicate worsening conditions.

**Hypertension and Diabetes:**

* 1. Common comorbidities in patients with heart failure.
  2. Important for identifying high-risk groups.

**Smoking:**

1.A major lifestyle factor contributing to cardiovascular disease and heart failure.

2.Helps in risk stratification.

**Platelets:**

1. May indicate coagulopathy or inflammation, relevant in heart failure patients.

**Outcome (Target):**

* 1. Binary indicator of the prediction goal (e.g., heart failure, mortality, or rehospitalization).
  2. The primary focus of machine learning models.

**Target Variable**

The target variable, Heart failure event, is binary:

· 1 (Yes) - Patient experienced a heart failure event.

· 0 (No) - Patient did not experience a heart failure event.

**Description:** Indicates if the patient was diagnosed with heart failure during the study period.

#### 5.3.1 Boxplots for Feature Variations

• Boxplots are useful for visualizing the distribution of data, identifying outliers, and understanding the spread and central tendency of variables. Here's how to interpret the boxplot results and key observations for heart failure prediction.

1. **Ejection Fraction** has a clear separation between those with and without heart failure risk, with lower values being indicative of risk.
2. **Serum Creatinine** values are higher for patients at risk, suggesting kidney dysfunction.
3. **Age** distribution is skewed toward older individuals being at higher risk for heart failure.
4. Outliers for **Serum Creatinine**, **Ejection Fraction**, and other features may point to extreme cases of heart failure, or require further investigation for data anomalies.

Boxplots are a great tool for visualizing feature distributions and relationships with the target variable. They provide a deeper understanding of how features behave and help identify patterns crucial for model training. Would you like to further investigate any specific feature

**6. Data Preprocessing**

Data preprocessing is essential to prepare the dataset for machine learning. It involves cleaning, transforming, and organizing the data to ensure accurate and efficient modelling.

# Steps in Data Preprocessing

# Loading the Dataset

• **Function:**

.load data ()

o Reads the dataset (heart.csv) into a panda DataFrame. o Utilizes @st. cache data to cache the dataset for faster subsequent loads.

def load data ():

file path = 'heart.csv' data = predocs (file path) return data

# 2. Preprocessing the Dataset

• **Function:** preprocess data(data) o **Handling Categorical Variables:**

* + Detects columns with object data type.
  + Applies Label Encoder to convert categorical variables into numerical representations.
  + Example: "Male"/"Female" → 0/1.

o **Handling Missing Values:**

▪ Checks for missing data using .is null (). sum (). ▪ Fills missing values with the column mean using

Datafilms (data. Mean ()). def preprocess data(data):

for column in data. Columns: if data[column].ditype == 'object': encoder = Label Encoder () data[column] = encoder.fit\_transform(data[column]) if Datavian'll (). sum (). any ():

data = Datafilms (data. Mean ()) return data

# 3. Splitting Data into Features and Target

* The features (X) include all columns except the last one.
* The target (y) is the last column, assumed to be the label for prediction.

# 4. Train-Test Split

* Splits the dataset into training and testing sets using
* train\_test\_split from scikit-learn:
  + **80% Training Data** o **20% Testing Data**
  + Stratified splitting ensures that the class distribution is consistent across training and testing sets.

# 5. Standardization of Features

* Standardization ensures that features have a mean of 0 and a standard deviation of 1, which is crucial for algorithms like **SVM**.
* Uses Standards Caler from scikit-learn:

o fit transform is applied to the training set. o transform is applied to the test set and user input to prevent data leakage.

# 6. Model Training

* **Model:** Support Vector Machine (SVM) with an RBF kernel.
* Trains the model on the standardized training set (Train, yttrian).

Enables probability estimates using probability=Ture

def train model(data): X = datafile [: -1]

y = datafile [: -1]

Train, Test, yttrian, yeast = train\_test\_split(X, y, test size=0.2, random state=42, stratify=y) scaler = Standards Caler ()

Train = scaler.fit\_transform(Train) Test = scaler.transform(Test) model = SVC (kernel='ruff', random state=42, probability=True) model. Fit (Train, yttrian) return model, scaler, Columns

# 7. Handling User Input

**Sidebar Input Form:**

* Dynamically generates input fields for all features based on their column names. o Defaults to the mean value of the respective column for user guidance.

**Data Transformation:**

* Converts the user input into a panda DataFrame.
* Applies the pre-trained scaler to standardize the user input data.

# 8. Making Predictions

• The standardized user input is fed into the trained SVM model for prediction:

o model. Predict () returns the class label (e.g., 0 or 1). o model.predict\_proba() returns the confidence/probability for the prediction.

# 9. Displaying Results

* Displays the prediction result (Heart Failure Risk Detected / No Risk) and confidence score.
* Provides an option to view:

o The processed dataset. o Model information (e.g., SVM with RBF kernel).

# Key Considerations

* **Preserve Interpretability:** Feature scaling and encoding should be consistent across the train and test sets.
* **Monitor Class Imbalance:** Address imbalance before training models.
* **Outlier Treatment:** Use domain knowledge to decide whether to keep or remove outliers.

Would you like help implementing this preprocessing pipeline or analysing specific parts of your data'set

**7. Machine Learning Models**

In this section, we explore the four machine learning models—**Logistic Regression**,

**Random Forest**, **K-Nearest Neighbors (KNN)**, and **Support Vector Machine (SVM)**— employed to predict the likelihood of liver disease based on the dataset. Each model has distinct characteristics, strengths, and weaknesses, which are critical for model selection and optimization.

### 7.1 Logistic Regression

* **Mechanics**: Logistic Regression models the probability of a binary outcome using the logistic function (sigmoid function), which maps any real-valued number to a value between 0 and 1. It estimates the probability of the target variable (liver disease presence) given the input features.

**Mathematically**:

P(y=1X) =11+e−(b0+b1X1+b2X2++box)P(y = 1 | X) = \frac {1}{1 + e^ {-(b\_0 + b\_1X\_1 + b\_2X\_2 + \dots + b\_nX\_n)

}}P(y=1X)=1+e−(b0+b1X1+b2X2++bnXn)1

Where: o P(y=1X) P(y = 1 | X) P(y=1X) is the probability that the patient has liver disease.

* + b0b\_0b0 is the intercept term, and b1, b2,…,bnb\_1, b\_2, \dots, b\_nb1, b2,…, bn,bn are the coefficients (weights) for each feature.
  + X1, X2,…,XnX\_1, X\_2, \dots, X\_nX1, X2,,Xn are the features, such as age, bilirubin levels, etc.
* **Strengths**:
  + Simple and interpretable.
  + Efficient for datasets with linear relationships between the features and target.
* **Weaknesses**: o Struggles with complex, non-linear relationships.
* **Use Case**:
  + Serves as a baseline model to assess the influence of features on liver disease prediction.

## 7.2 Decision Tree

* **Mechanics**: A **Decision Tree** is a supervised learning algorithm used for both classification and regression tasks. It splits the data into subsets based on feature values, creating a tree-like structure where each internal node represents a feature test, each branch represents an outcome of the test, and each leaf node represents a class label or continuous value.
* **Strengths**:
  + Simple to understand and interpret, making it easy to visualize and communicate results.
  + Handles both categorical and numerical data effectively.
  + Non-linear relationships between features can be captured easily.
* **Weaknesses**:
* Prone to overfitting, especially when the tree is deep.
  + Sensitive to noisy data and outliers.
* **Use Case**:
  + **Decision Trees** are particularly useful for situations where the decisionmaking process can be represented as a sequence of decisions, such as classifying liver disease based on feature thresholds like bilirubin levels or liver enzymes.

### 7.3 Random Forest

* **Mechanics**: Random Forest is an ensemble method that constructs a collection of decision trees during training. Each tree is trained on a random subset of the data and features (bootstrapping), and the final prediction is the majority vote from all trees (for classification tasks).
* **Strengths**:
  + Handles non-linear relationships effectively. o Resistant to overfitting due to its ensemble approach.
  + Provides feature importance, helping identify key predictors for liver disease.
* **Weaknesses**: o Computationally intensive, especially with a large number of trees.
  + Less interpretable compared to simpler models like Logistic Regression.
* **Use Case**:
  + Highly effective for capturing complex relationships and interactions between features in the liver disease dataset.

### 7.4 Support Vector Machine (SVM)

* **Mechanics**: SVM constructs a hyperplane or a set of hyperplanes in a highdimensional space that separates the classes with maximum margin. It can efficiently perform non-linear classification using kernel methods (e.g., radial basis function) to map data into higher dimensions.
* **Strengths**: o Highly effective for high-dimensional datasets, which is common in medical data.
  + Can capture non-linear relationships using kernels.
* **Weaknesses**: o Computationally expensive, especially with large datasets. o Sensitive to the choice of kernel and parameter tuning.
* **Use Case**:
  + Ideal for liver disease prediction where data has complex, non-linear relationships and high-dimensional feature spaces.

Model performance summary:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| S.No | kernel | Support vector machine | Decision tree | Random forest | Logistic regression |
| 0 | rbf | 88.59 | 78.80 | 87.50 | 86.96 |
| 1 | Linear | 86.96 | 78.80 | 87.50 | 86.96 |
| 2 | polynomal | 89.13 | 78.80 | 87.50 | 86.96 |
| 3 | Sigmoid | 79.89 | 78.80 | 87.50 | 86.96 |



