

GUI BASED HEART STROKE PREDICTION USING MACHINE LEARNING ALGORITHMS

Sharma. V¹, Surya Prakash. S², Mrs. M. Nivethitha Devi, M.E.,³

UG Scholar, Department of Electronics and Instrumentation Engineering, St. Joseph's College Of Engineering,
Chennai^{1,2}

Associate Professor, Department of Electronics and Instrumentation Engineering, St. Joseph's College Of Engineering,
Chennai³

Abstract: Many predictive techniques have been widely applied in clinical decision making such as predicting occurrence of a disease or diagnosis, evaluating prognosis or outcome of diseases and assisting clinicians to recommend treatment of diseases. However, the conventional predictive models or techniques are still not effective enough in capturing the underlying knowledge because it is incapable of simulating the complexity on feature representation of the medical problem domains. To overcome this problem, predictive analytical techniques for heart stroke using machine learning model applied on given hospital dataset. The atrial fibrillation symptoms in heart patients are a major risk factor of stroke and share common variables to predict stroke and the analysis of given dataset by supervised machine learning algorithm to capture several information's like, variable identification, uni-variate analysis, bi-variate and multi-variate analysis, missing value treatments etc. The main objective is to predictive analytics model to diagnose heart stroke stages of patients. Additionally, discuss the performance from the given hospital dataset with evaluation of classification report and identify the confusion matrix. To propose a machine learning-based method to accurately predict the heart stroke by given attributes in the form of best accuracy from comparing supervise classification machine learning algorithms. Additionally, to compare and discuss the performance of various machine learning algorithms from the given healthcare department dataset with evaluation classification report, identify the confusion matrix and to categorizing data from priority and the result shows that the effectiveness of GUI based the proposed machine learning algorithm technique can be compared with best accuracy with precision, Recall and F1 Score.

Keywords: Dataset, python, Prediction of Accuracy result.

1. INTRODUCTION

1.1 Domain overview

Machine learning is to predict the future from past data. Machine learning (ML) is a type of artificial intelligence (AI) that provides computers with the ability to learn without being explicitly programmed. Machine learning focuses on the development of Computer Programs that can change when exposed to new data and the basics of Machine Learning, implementation of a simple machine learning algorithm using python. Process of training and prediction involves use of specialized algorithms.

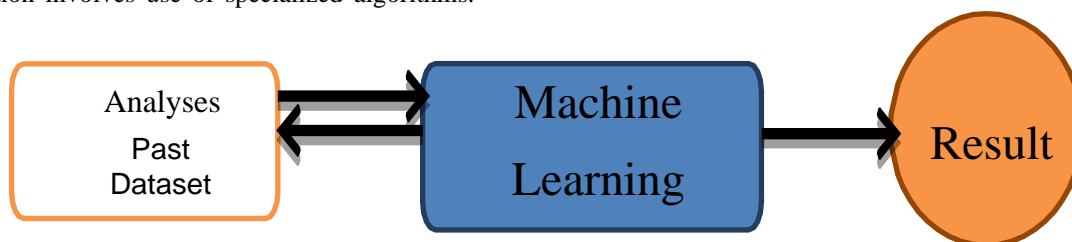


Fig: Process of Machine learning

1.2 Objectives

This analysis aims to observe which features are most helpful in predicting the patient diseases by given attribute symptoms of heart disease or not and to see the general trends that may help us in model selection and hyper parameter

selection. To achieve used machine learning classification methods to fit a function that can predict the discrete class of new input.

The repository is a learning exercise to:

- Apply the fundamental concepts of machine learning from an available dataset and Evaluate and interpret my results and justify my interpretation based on observed dataset.
- Create notebooks that serve as computational records and document my thought process and investigate the patient details whether patient affected by disease or not to analyses the data set.
- Evaluate and analyses statistical and visualized results, which find the standard patterns for all regiments.

1.3 Overview of the system

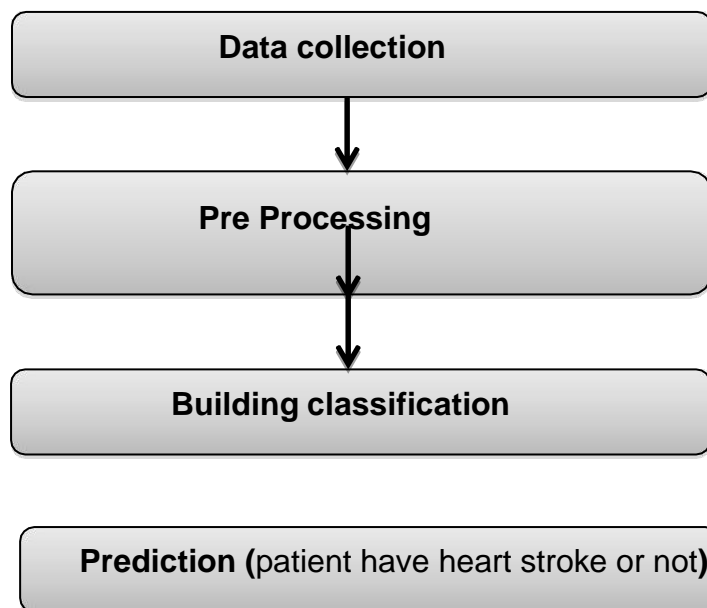


Fig: data flow diagram for Machine learning model

1.4 Existing system

Limitations in available diagnostic metrics restrict the efficacy of managing therapies for cardiogenic shock. In current clinical practice, cardiovascular state is inferred through measurement of pulmonary capillary wedge pressure and reliance on linear approximations between pressure and flow to estimate peripheral vascular resistance. Mechanical circulatory support devices residing within the left ventricle and aorta provide an opportunity for both determining cardiac and vascular state and offering therapeutic benefit.

1.5 Proposed system

Exploratory Data Analysis:

It will be using Jupyter notebook to work on this dataset and will first go with importing the necessary libraries and import our dataset to Jupyter notebook:

Process of functional steps,

- Problem defines
- Preparing data
- Evaluating algorithms
- Improving results
- Prediction the result

2. SOFTWARE REQUIREMENTS

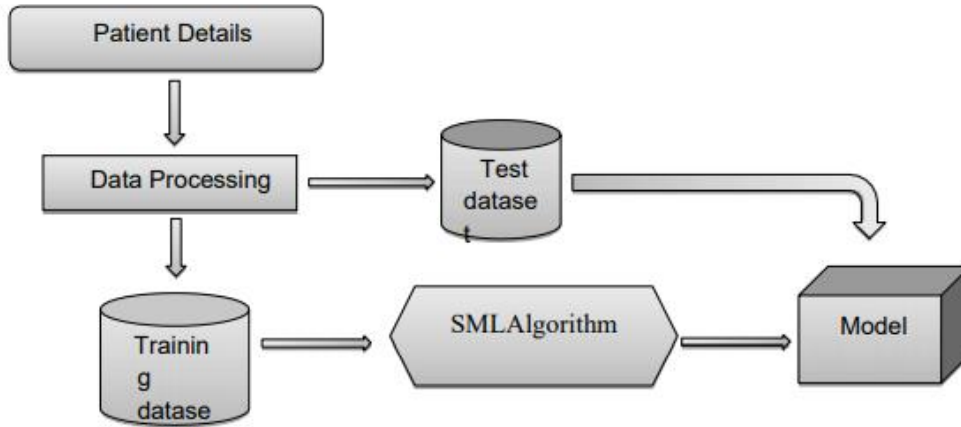


Fig: Architecture of Proposed model

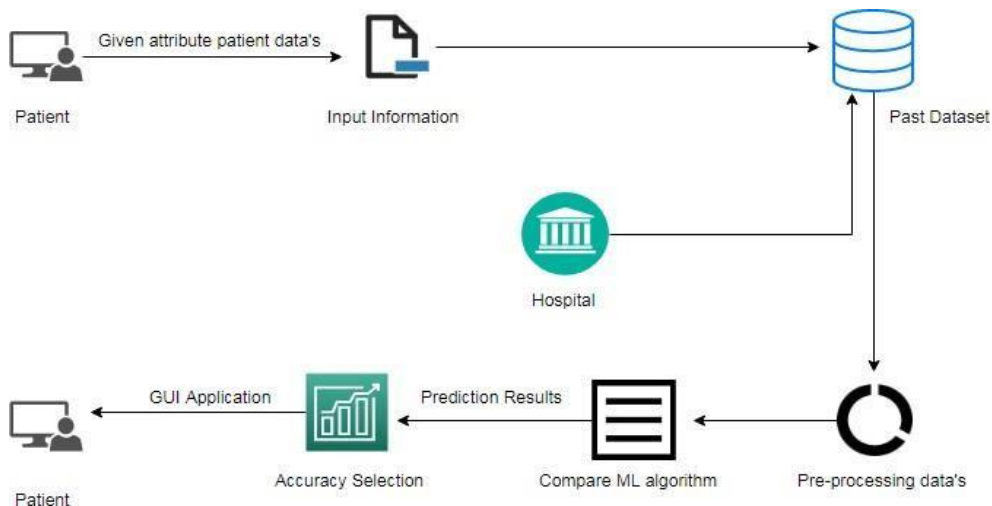
Anaconda is a free and open-source distribution of the Python and R programming languages for scientific computing (data science, machine learning applications, large-scale data processing, predictive analytics, etc.), that aims to simplify package management and deployment.

The following applications are available by default in Navigator:

- JupyterLab
- Jupyter Notebook
- QtConsole
- Spyder
- Glueviz
- Orange
- Rstudio
- Visual Studio Code

2.1 Working Process

- Download and install anaconda and get the most useful package for machine learning in Python.
- Load a dataset and understand its structure using statistical summaries and data visualization.
- machine learning models, pick the best and build confidence that the accuracy is reliable.



2.2 Algorithm Explanation

In machine learning and statistics, classification is a supervised learning approach in which the computer program learns from the data input given to it and then uses this learning to classify new observation. This data set may simply be bi-class (like identifying whether the person is male or female or that the mail is spam or non-spam) or it may be multi-class too.

Some examples of classification problems are: speech recognition, handwriting recognition, bio metric identification, document classification etc. In Supervised Learning, algorithms learn from labeled data. After understanding the data, the algorithm determines which label should be given to new data based on pattern and associating the patterns to the unlabeled new data.

Used Python Packages

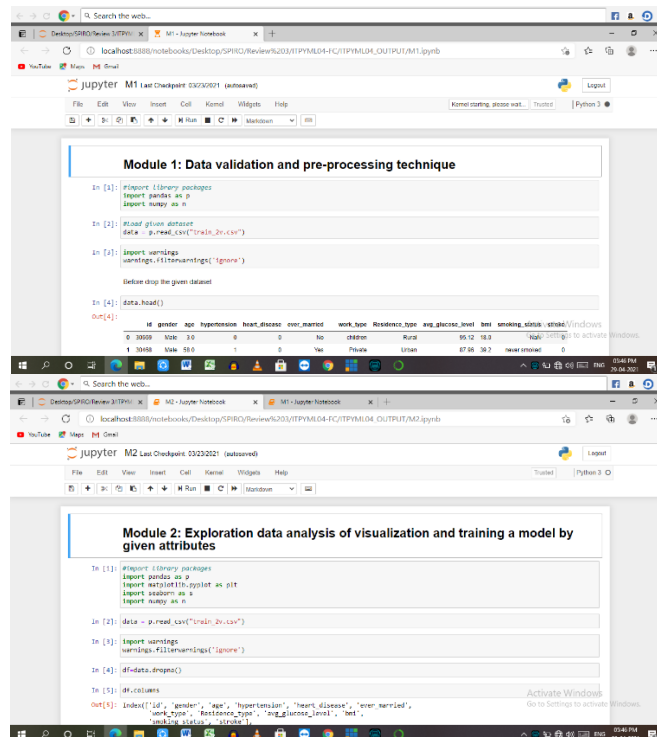
sklearn:

- In python, sklearn is a machine learning package which include a lot of ML algorithms.
- Here, we are using some of its modules like train_test_split, DecisionTreeClassifier or Logistic Regression and accuracy_score.]

NumPy:

- It is a numeric python module which provides fast maths functions for calculations.
- It is used to read data in numpy arrays and for manipulation purpose. Pandas:
- Used to read and write different files.
- Data manipulation can be done easily with data frames. Matplotlib:
- Data visualization is a useful way to help with identify the patterns from given dataset.
- Data manipulation can be done easily with data frames.

2.3 Source Code



```
Module 1: Data validation and pre-processing technique

In [1]: #Import library packages
import pandas as p
import numpy as n

In [2]: #Load given dataset
data = p.read_csv("train_2v.csv")

In [3]: import warnings
warnings.filterwarnings("ignore")

#Delete drop the given dataset

In [4]: data.head()

Out[4]:
   id  gender  age  hypertension  heart_disease  ever_married  work_type  Residence_type  avg_glucose_level  bmi  smoking_status  stroke
0  30659  Male  39.0  0  0  0  No  Other  Rural  95.12  18.9  Never smoked  0
1  10153  Male  51.0  1  1  0  Yes  Private  Urban  87.66  38.7  Never smoked  0

Module 2: Exploration data analysis of visualization and training a model by given attributes

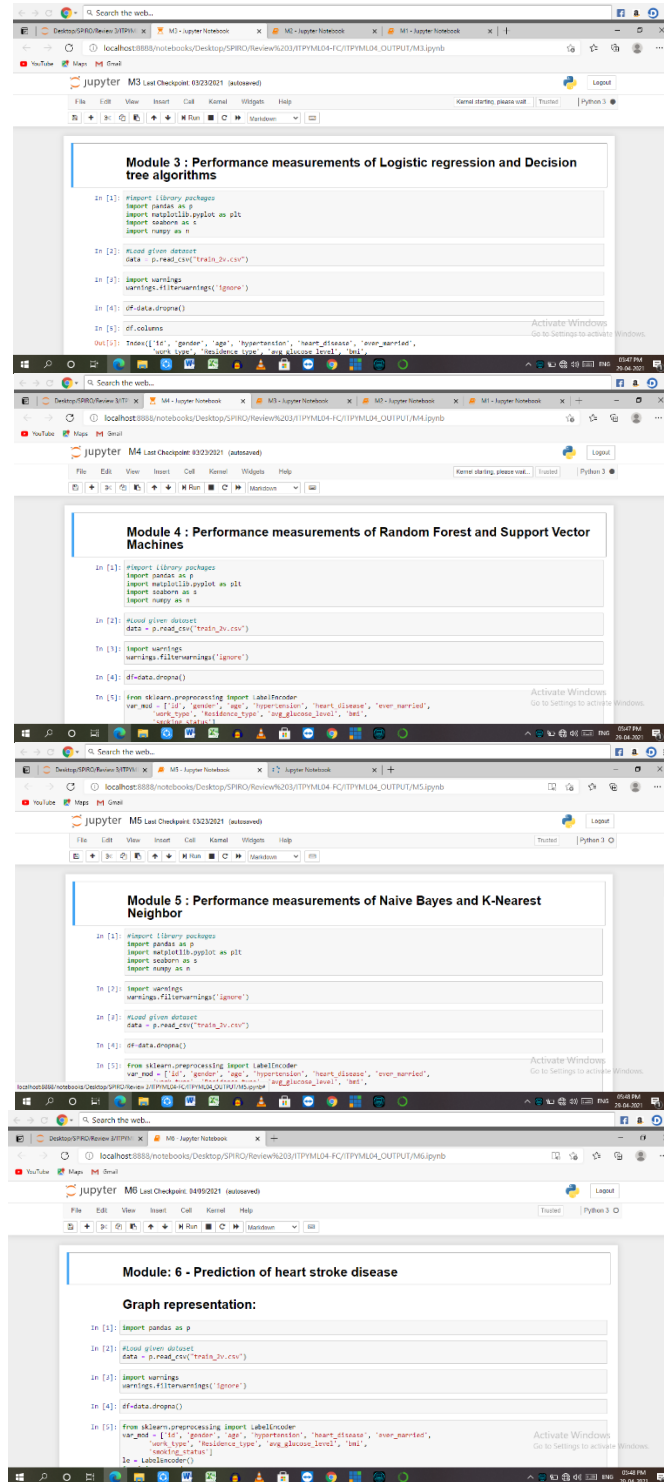
In [1]: #Import library packages
import pandas as p
import matplotlib.pyplot as plt
import seaborn as s
import numpy as n

In [2]: data = p.read_csv("train_2v.csv")

In [3]: import warnings
warnings.filterwarnings("ignore")

In [4]: df=data.dropna()

In [5]: #columns
Out[5]: index(['id', 'gender', 'age', 'hypertension', 'heart_disease', 'ever_married', 'work_type', 'Residence_type', 'avg_glucose_level', 'bmi', 'smoking_status', 'stroke'])
```



The image displays four sequential screenshots of a Jupyter Notebook interface, each showing a different module of code. The code in each cell is as follows:

Module 3 : Performance measurements of Logistic regression and Decision tree algorithms

```
In [1]: #Import library packages
import pandas as p
import matplotlib.pyplot as plt
import seaborn as s
import numpy as n

In [2]: #Load given dataset
data = p.read_csv("train_2v.csv")

In [3]: #Import warnings
warnings.filterwarnings('ignore')

In [4]: #df.data.dropna()

Out[1]: Index(['sex', 'gender', 'age', 'hypertenstion', 'heart_disease', 'ever_married',
              'work_type', 'residence_type', 'avg_glucose_level', 'bmi',
              dtype=object)]
```

Module 4 : Performance measurements of Random Forest and Support Vector Machines

```
In [1]: #Import library packages
import pandas as p
import matplotlib.pyplot as plt
import seaborn as s
import numpy as n

In [2]: #Load given dataset
data = p.read_csv("train_2v.csv")

In [3]: #Import warnings
warnings.filterwarnings('ignore')

In [4]: #df.data.dropna()

In [5]: from sklearn.preprocessing import LabelEncoder
var_map = ['sex', 'gender', 'age', 'hypertenstion', 'heart_disease', 'ever_married',
           'work_type', 'residence_type', 'avg_glucose_level', 'bmi']
```

Module 5 : Performance measurements of Naive Bayes and K-Nearest Neighbor

```
In [1]: #Import library packages
import pandas as p
import matplotlib.pyplot as plt
import seaborn as s
import numpy as n

In [2]: #Import warnings
warnings.filterwarnings('ignore')

In [3]: #Load given dataset
data = p.read_csv("train_2v.csv")

In [4]: #df.data.dropna()

In [5]: from sklearn.preprocessing import LabelEncoder
var_map = ['sex', 'gender', 'age', 'hypertenstion', 'heart_disease', 'ever_married',
           'work_type', 'residence_type', 'avg_glucose_level', 'bmi']
```

Module: 6 - Prediction of heart stroke disease

Graph representation:

```
In [1]: import pandas as p

In [2]: #Load given dataset
data = p.read_csv("train_2v.csv")

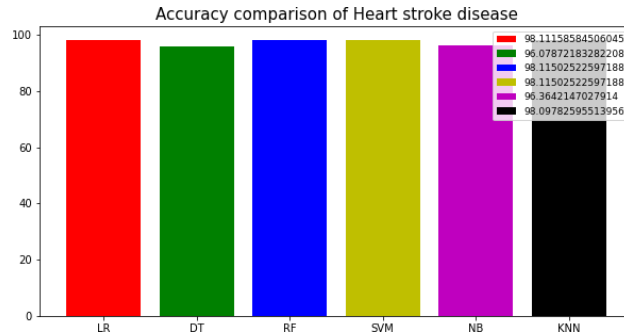
In [3]: #Import warnings
warnings.filterwarnings('ignore')

In [4]: #df.data.dropna()

In [5]: from sklearn.preprocessing import LabelEncoder
var_map = ['sex', 'gender', 'age', 'hypertenstion', 'heart_disease', 'ever_married',
           'work_type', 'residence_type', 'avg_glucose_level', 'bmi']
X = data.drop('status', axis=1)
y = data['status']
```

3. RESULT AND DISCUSSION

Comparison Result



GUI Results



REFERENCES

- [1]. A Machine Learning Approach to Classifying Self-Reported Health Status in a cohort of Patients with Heart Disease using Activity Tracker Data., Yiwen Meng, William Speier, Member, ChrisandraShufelt, Sandy Joung, Jennifer E
- [2]. Human Heart Disease Prediction System using Data Mining Techniques., Theresa Princy. R, J. Thomas
- [3]. Predicting heart failure class using a sequence prediction algorithm., Carine BouRjeily , Georges Badr , Amir Hajjam Al Hassani , Emmanuel Andres
- [4]. Changes in Daily Measures of Blood Pressure and Heart Rate Improve Weight-based Detection of Heart Failure Deterioration in Patients on Telemonitoring., Rohan Joshi and Illapha Cuba Gyllensten
- [5]. Heart Disease Prediction using Evolutionary Rule Learning., Aakash Chauhan, Aditya Jain, Purushottam Sharma, Vikas Deep
- [6]. Sparse Support Vector Machine for Intrapartum Fetal Heart Rate Classification., Jordan Frecon , Roberto Leonarduzzi , Nelly Pustelnik, Patrice Abry , Muriel Doret