STROKE RISK PREDICTION

Project Report

Submitted by

P.VINEETHA AP22110010580

K.INDU MEGHANA AP22110010540

A.MAHITHA AP22110010576

SAMEER RAJ AP22110010555

RUTIKA AMBADKAR AP22110010559

Under the Supervision of

Mr. B. L. V. SIVA RAMA KRISHNA
Assistant Professor
Department of Computer Science and Engineering
SRM University-AP

In partial fulfilment for the requirements of the project

BACHELOR OF TECHNOLOGY IN

COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SRM UNIVERSITY-AP

NEERUKONDA

MANAGALAGIRI - 522503

ANDHRA PRADESH, INDIA

NOVEMBER-2024

CERTIFICATE

This is to certify that the project work entitled **"Stroke Risk Prediction"** is a Bonafide record of project work carried out by the following students:

- **Ms. P.Vineetha** (Roll No.: AP22110010580)
- **Ms. K.Indu Meghana** (Roll No.: AP22110010540)
- **Ms. A.Mahitha** (Roll No.: AP22110010576)
- **Mr. Sameer Raj** (Roll No.: AP22110010555)
- **Ms. Rutika Ambadkar** (Roll No.: AP22110010559)

from the **Department of Computer Science and Engineering, SRM University-AP.** The students conducted this project work under my supervision during the period **August 2024 to November 2024**.It is further certified that, to the best of my knowledge, this project has not previously formed the basis for the award of any degree or any similar title to this or any other candidate.

This is also to certify that the project work represents the **teamwork** of the candidates.

Station: Mangalagiri
Date: 22-11-2024

Mr. B. L. V. Siva Rama Krishna
Assistant Professor
Department of Computer Science & Engineering
SRM University-AP
Andhra Pradesh

TABLE OF CONTENTS

INTRODUCTION

Stroke is one of the leading causes of disability and death worldwide, posing a significant public health challenge. Early detection and prevention are crucial for reducing its impact on individuals and healthcare systems. However, predicting stroke risk is complex due to the interplay of various factors such as age, lifestyle, medical history, and genetic predisposition.

This project leverages machine learning techniques to develop a predictive model for stroke risk. The project emphasizes the importance of automating stroke prediction through modern computational techniques.By analyzing a dataset containing demographic, lifestyle, and medical information, the model identifies key factors contributing to stroke and provides a probabilistic prediction of stroke occurrence.

The project includes several stages such as data preprocessing, feature selection, data splitting, model development, and evaluation. Multiple algorithms, including Random Forest, Multi-Layer Perceptron, and,Gradient Boosting were employed to identify the most effective approach for stroke risk prediction. Key performance metrics such as accuracy, precision, recall, and F1-score were used to assess the models.

Each algorithm’s performance was compared to ensure the most reliable and accurate model is chosen for deployment. By integrating predictive analytics with healthcare decision-making, this project demonstrates the transformative role of machine learning in preventing life-threatening conditions and supporting better patient care.

This work demonstrates the potential of machine learning to address critical healthcare challenges by improving prediction accuracy, enhancing decision-making, and ultimately contributing to better patient outcomes.

PROBLEM DEFINITION

Our project addresses the problem of identifying individuals at high risk of stroke by developing a predictive model using machine learning techniques.
The primary challenge is to process and analyze a dataset containing diverse features, handle missing or inconsistent data, and identify the most relevant predictors of stroke.
The goal is to create a robust and accurate model that can assist healthcare professionals in making timely and informed decisions, thereby reducing the prevalence of stroke-related complications and fatalities.
By leveraging machine learning, this project aims to bridge the gap between data availability and actionable insights in stroke prediction and prevention.

PROBLEM STATEMENT

Creating a predictive model to identify individuals at risk of heart stroke using demographic, lifestyle and clinical data. The aim is to enable early intervention and lifestyle modifications, reducing heart stroke incidence and improving cardiovascular health outcomes.

We need to predict the stroke risk effectively by using different machine learning algorithms.

OBJECTIVES

The objective is to assist healthcare professionals in early diagnosis and preventive care by offering a reliable and automated decision-support system.

- Correctly predict seizures or non-seizures.
- To use the feature selection for selecting the best features from our dataset.
- To implement machine learning algorithms.
- To enhance the overall performance analysis.

METHODOLOGY

The methodology for this project involves a systematic approach to build a robust machine learning model for stroke risk prediction.

DATA PREPROCESSING:

Data pre-processing is the process of removing the unwanted data from the dataset.
The project begins with loading the stroke dataset, which contains demographic, health, and lifestyle features. Missing values in the dataset are addressed by replacing them with suitable substitutes such as zero's to ensure the data's completeness. Label encoding is applied to transform categorical variables, such as gender, smoking status, and marital status, into numerical formats for compatibility with machine learning algorithms.

FEATURE SELECTION:

To enhance the model's efficiency and accuracy, feature selection is performed using statistical methods such as the Chi-Square test. This step identifies the most relevant predictors of stroke by analyzing the relationship between features and the target variable. The selected features are then used for model training, reducing dimensionality and focusing on the factors most critical to stroke prediction.

DATA SPLITTING:

Data splitting is the act of partitioning available data into two portions, usually for cross-validaton purposes.
The dataset is split into training and testing subsets, typically in an 80:20 ratio. This ensures the model is trained on a majority of the data while reserving a portion for evaluating its performance. The train-test split is essential for assessing how well the model generalizes to unseen data.One Portion of the data is used to develop a predictive model and the other to evaluate the model's performance.

CLASSIFICATION:

Multiple machine learning algorithms, including Random Forest, Gradient Boosting, and Multi-Layer Perceptron (MLP), are implemented to build predictive models. Each model is trained on the training dataset and optimized for accurate classification. Hyperparameters such as the number of estimators in Random Forest or the learning rate in Gradient Boosting are tuned to enhance performance.

- *Random Forest:*
  It is a Supervised Machine Learning Algorithm that is used widely in Classification and Regression problems. It builds decision trees on different samples and takes their majority vote for classification and average in case of regression.

- *Multi-Layer Perceptron(MLP):*
  It is a supplement of feed forward neural network. It consists of three types of layers—the input layer, output layer and hidden layer.

- *Gradient Boosting Trees:*
  The fundamental idea behind GBT is to iteratively train decision trees in a sequential manner, where each new tree aims to correct the errors made by the combination of previously built trees.

PERFORMANCE METRICS:

The performance of each model is evaluated using metrics such as accuracy, precision, recall, and F1-score. A classification report is generated to analyze the model's ability to correctly identify stroke and non-stroke cases.

CODE

```
import pandas as pd
from sklearn import preprocessing
from sklearn.model_selection import train_test_split
from sklearn.feature_selection import SelectKBest
from sklearn.feature_selection import chi2
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.preprocessing import LabelEncoder
dataframe=pd.read_csv("/content/Stroke data.csv")
print("-----------------------------------------------------")
print("Data Selection                  ")
print("-----------------------------------------------------")
print()
print(dataframe.head(15))
print()
print("-----------------------------------------------------")
print("Before Handling Missing values             ")
print("-----------------------------------------------------")
print()
print(dataframe.isnull().sum())

print()
print("-----------------------------------------------------")
print(" After Handling Missing Values             ")
print("-----------------------------------------------------")
print()
dataframe=dataframe.fillna(0)
print(dataframe.isnull().sum())
print()
print("-----------------------------------------------------")
print("Before Label Encoding                 ")
print("-----------------------------------------------------")
print()
print(dataframe['gender'].head(10))

print()
print("-----------------------------------------------------")
print("After Label Encoding                  ")
```

```
print("-----------------------------------------------------")
print()
label_encoder = preprocessing.LabelEncoder()
dataframe['gender']= label_encoder.fit_transform(dataframe['gender'])

print(dataframe['gender'].head(10))

Label=['ever_married','work_type','Residence_type','smoking_status']

dataframe[Label] = dataframe[Label].apply(label_encoder.fit_transform)
X=dataframe.drop(['stroke'],axis=1)
y=dataframe['stroke']

chi_squ = SelectKBest(chi2,k=10)

best_fea= chi_squ.fit_transform(X, y)

print()
print("-----------------------------------------------------")
print("Feature Selection ---> Chi square                 ")
print("-----------------------------------------------------")
print()
print("Total no of original Features :",X.shape[1])
print()
print("Total no of reduced Features  :",best_fea.shape[1])
print()
X_train, X_test, y_train, y_test = train_test_split(best_fea, y,
test_size=0.2)

print()
print("-----------------------------------------------------")
print("Data Splitting                    ")
print("-----------------------------------------------------")
print()

print("Total no of data        :",dataframe.shape[0])
print("Total no of test data   :",X_test.shape[0])
print("Total no of train data  :",X_train.shape[0])
```

```
from sklearn.ensemble import RandomForestClassifier

rf = RandomForestClassifier(n_estimators=20, random_state=0)

rf.fit(X_train, y_train)

pred_rf=rf.predict(X_test)


from sklearn import metrics

acc_rf=metrics.accuracy_score(y_test,pred_rf)*100
print("Random Forest")

print("1. Accuracy =", acc_rf,'%')
print()
print("2. Classification Report:")
print()
print(metrics.classification_report(y_test,pred_rf))


from sklearn.neural_network import MLPClassifier

mlp = MLPClassifier(hidden_layer_sizes=(150,100,50),
max_iter=300,activation = 'relu',solver='adam',random_state=1)


mlp.fit(X_train, y_train)

pred_mlp=mlp.predict(X_test)

acc_mlp=metrics.accuracy_score(y_test,pred_mlp)*100
print("Multi Layer Perceptron")


print("1. Accuracy =", acc_mlp,'%')
print()
print("2. Classification Report:")
print()
print(metrics.classification_report(y_test,pred_rf))
```

```
gb=GradientBoostingClassifier(n_estimators=1000, random_state=0)
gb.fit(X_train, y_train)

pred_gb=gb.predict(X_test)


from sklearn import metrics

acc_gb=metrics.accuracy_score(y_test,pred_rf)*100
print("Gradient Boosting")

print("1. Accuracy =", acc_gb,'%')
print()
print("2. Classification Report:")
print()
print(metrics.classification_report(y_test,pred_gb))
print("----------------------------------------------------")
print("Stroke Prediction")
print("----------------------------------------------------")
print()
import numpy as np
data12=np.array([62980,99,1,0,1,1,0,78.26,41.7,1]).reshape(1,-1)

predictions=mlp.predict(data12)
print(predictions)

print("-----Prediction---")

if(predictions==0):
   print(data12[0][0],"The Patient is not affected by Stroke")
if(predictions==1):
   print(data12[0][0],"The Patient is  affected by Stroke")
data12=np.array([17752,76,0,1,1,2,1,79.05,0,0]).reshape(1,-1)

predictions=rf.predict(data12)
print(predictions)

print("-----Prediction---")
```

```
if(predictions==0):
   print(data12[0][0],"The Patient is not affected by Stroke")
if(predictions==1):
   print(data12[0][0],"The Patient is  affected by Stroke")

data12=np.array([23145,67,1,1,1,2,1,80.6,1,1]).reshape(1,-1)

predictions=gb.predict(data12)
print(predictions)

print("-----Prediction---")

if(predictions==0):
   print(data12[0][0],"The Patient is not affected by Stroke")
if(predictions==1):
   print(data12[0][0],"The Patient is  affected by Stroke")
```

RESULTS

Random Forest
1. Accuracy = 94.9119373776908 %

2. Classification Report:

	precision	recall	f1-score	support
0	0.95	1.00	0.97	968
1	0.75	0.06	0.10	54
accuracy			0.95	1022
macro avg	0.85	0.53	0.54	1022
weighted avg	0.94	0.95	0.93	1022

Multi Layer Perceptron
1. Accuracy = 94.71624266144813 %

2. Classification Report:

	precision	recall	f1-score	support
0	0.95	1.00	0.97	968
1	0.75	0.06	0.10	54
accuracy			0.95	1022
macro avg	0.85	0.53	0.54	1022
weighted avg	0.94	0.95	0.93	1022

```
 Gradient Boosting
 1. Accuracy = 94.9119373776908 %

 2. Classification Report:

              precision    recall  f1-score   support

           0       0.95      0.99      0.97       968
           1       0.41      0.13      0.20        54

    accuracy                           0.94      1022
   macro avg       0.68      0.56      0.58      1022
weighted avg       0.92      0.94      0.93      1022

--------------------------------------------------------
Stroke Prediction
--------------------------------------------------------

[0]
-----Prediction---
62980.0 The Patient is not affected by Stroke
[1]
-----Prediction---
17752.0 The Patient is  affected by Stroke
[0]
-----Prediction---
23145.0 The Patient is not affected by Stroke
```

FUTURE RESEARCH

Future research in stroke risk prediction could focus on integrating real-time data sources such as wearable devices, electronic health records (EHRs), and mobile health applications. This would enable continuous monitoring of key health parameters like blood pressure, heart rate, and physical activity, providing dynamic and personalized risk assessments.

Incorporating advanced techniques like deep learning, particularly models such as Recurrent Neural Networks (RNNs) for time-series data or Convolutional Neural Networks (CNNs) for medical imaging, could further improve the accuracy and reliability of predictions.

Future research should also focus on clinical implementation, validating the model in real-world healthcare settings to assess its practical impact. By addressing these aspects, future advancements could make stroke risk prediction models not only more accurate but also highly applicable in preventive healthcare strategies.

REFERENCES

S. Koton et al., "Report on stroke incidence and mortality in US populations, 1987 to 2011," Journal of the American Medical Association, vol. 312, no. 3. p. 259-268, 2014.

"Heart Disease and Stroke Statistics 2019 Update: American Heart 18 Association Report", E. J. Benjamin, P. Muntner, thiab M. S. Bittencourt. Cycles, cilt. 139, no. 10, p. e56-e528, 2019.