# STROKE PREDICTION USING MACHINE LEARNING

A Minor Project Report

in partial fulfillment of the degree

**Bachelor of Technology**
in
**Computer Science & Artificial Intelligence**

**By**


| | |
|---|---|
| 2103A52180 | P. Harsha Vardhan Reddy |
| 2103A52179 | T. Deekshitha Rao |
| 2103A52173 | A. Soumika |
| 2103A52014 | E. Sai Krishna Reddy |
| 2103A52178 | D. Saketh Reddy |


**Under the Guidance of**

**Dr Minakshmi Shaw**

**Submitted to**


**SCHOOL OF COMPUTER SCIENCE & ARTIFICIAL INTELLIGENCE**
**SR UNIVERSITY, ANANTHASAGAR, WARANGAL**


**April, 2024.**

# SCHOOL OF COMPUTER SCIENCE & ARTIFICIAL INTELLIGENCE

## CERTIFICATE

This is to certify that this project entitled **"STROKE PREDICTION USING MACHINE LEARNING**" is the bonafide work carried out by **P. Harsha Vardhan Reddy, T. Deekshitha, A.Soumika, E. Sai Krishna Reddy, D. Saketh Reddy** as a Minor Project for the partial fulfillment to award the degree **BACHELOR OF TECHNOLOGY** in **COMPUTER SCIENCE & ARTIFICIAL INTELLIGENCE** during the academic year 2022-2023 under our guidance and Supervision.


**Dr. Minakshmi Shaw**
Asst. Prof (CS&AI)
SR University,
Ananthasagar, Warangal.

**Dr.M.Sheshikala**
Assoc.Prof. & HOD (CSE),
SRUniversity,
Ananthasagar,Warangal.

# ACKNOWLEDGEMENT

We owe an enormous debt of gratitude to our project guide Dr. Minaksmi Shaw, Assistant Professor as well as Head of the CSE Department Dr. M.Sheshikala, Associate Professor for guiding us from the beginning through the end of the Minor Project with their intellectual advices and insightful suggestions. We truly value their consistent feedback on our progress, which was always constructive and encouraging and ultimately drove us to the right direction.

We express our thanks to the project co-ordinators Dr. P Praveen, Assoc. Prof for their encouragement and support.

We wish to take this opportunity to express our sincere gratitude and deep sense of respect to our beloved Dean, Dr.Indrajeet Gupta,Professor for his continuous support and guidance to complete this project in the institute.

Finally, we express our thanks to all the teaching and non-teaching staff of the department for their suggestions and timely support.

P. Harsha Vardhan Reddy

T. Deekshitha Rao

A. Soumika

E. Sai Krishna Reddy

D. Saketh Reddy

# ABSTRACT

Most strokes occur due an unexpected obstacles of course prompting the heart. Early awareness of different warning signs of stroke can minimize a stroke. In this project, We propose early predictions of stroke diseases using different machine learning approaches with the occurrence of hypertension, body mass index, heart diseases, average glucose level, smoking status, previous stroke, and age. Using these high feature attributes various classifiers have been trained namely Logistic regression, Decision tree classifier, K neighbor classifier, Gradient boosting classifier, XGBoost classifier, and Random Forest Classifier for predicting the stroke afterward results of the base classifiers have been aggregated using the weighted voting approach to reach highest accuracy. Here the study has achieved an accuracy of 97% where the weighted voting classifier performs better than the base classifiers this model gives the best accuracy for the prediction of stroke the area under the curve value of the weighted voting classifier also has a high false positive rate and false negative rate of the weighted classifier is lowest compared with others. As a result, weighted voting is almost the perfect classifier for predicting the stroke that can be used by physicians and patients to prescribe an early detective potential stroke.

## TABLE OF CONTENTS

# 1. INTRODUCTION

A stroke occurs when blood flow to a part of the brain is interrupted or reduced and the cells in that area are deprived of nutrients and oxygen and begin to die. A stroke is a medical emergency that requires immediate treatment. Early detection and proper management are necessary to minimize brain damage and complications elsewhere in the body. According to the World Health Organization (WHO), 15 million people in the world suffer from a brain disease every year, and one person dies every four to five minutes. There are two types of stroke: ischemic stroke and hemorrhagic stroke. In an ischemic stroke, the flow of fluid is blocked by blood, and in a hemorrhagic stroke, a weakened blood vessel ruptures and bleeds into the brain. You can prevent stroke by giving up unhealthy lifestyle habits like smoking and drinking, controlling your body mass index (BMI) and average blood sugar, and maintaining your heart and blood kidneys to lead a healthy and balanced life. Injuries must be identified and treated to prevent further damage. In this article, high blood pressure, BMI level, heart disease and average blood sugar level were considered as variables that predict stroke.

## 1.1 PROBLEM STATEMENTS:

People of all ages are increasingly worried about strokes occurring unexpectedly. This worry is because stroke seems to be happening to anyone, regardless of their age. It's important to be able to check if they might be at risk of having a stroke based on their symptoms. Now there are no easy ways for individuals to do this. So they need a simple way to understand if they could be at risk of having a stroke so they can take action to stay healthy and get help if needed. This Project aims to create a solution to this problem by using data and technology to help people understand the risk of stroke based on their symptoms.

This project is crucial because strokes can have serious consequences,including disability and even death. By providing a tool for individuals to assess their risk of stroke, we can empower them to take proactive steps toward prevention and early intervention. Additionally raising awareness about stroke risk factors and symptoms can lead to better public health outcomes by promoting healthier lifestyles and encouraging timely medical care.

## 1.2 EXISTING SYSTEM:

The existing system for stroke production primarily relies on traditional risk assessment methods conducted by healthcare professionals. These methods often involve analyzing demographic information medical history and physical examinations to estimate an individual's likelihood of experiencing a stroke. However, this approach may not always be easily

accessible to the general public and could require a visit to the healthcare facility.

Moreover, existing public awareness about strokes typically focus on general risk factors and symptoms rather than providing personalized risk assessments. While these campaigns served to educate the public about the importance of stroke prevention and recognition they may not address the specific concerns of individuals who are worried about their own risk.

Overall, the current system lacks a user-friendly and individualized approach for people to assess their stroke risk based on their symptoms. There is a need for a more accessible and proactive solution that allows individuals to understand their risks and take appropriate actions to protect their health.

### 1.3 Proposed System:

In the proposed system, we aim to develop a user-friendly web application using Flask and Python code. This application will utilize a data set containing relevant health information to predict an individual's risk of experiencing a stroke. Through an intuitive interface, users will be able to input their data and select specific features related to their health status.

Once the user submits their information the application will process the data using machine learning algorithms to generate a prediction regarding the livelihood of having a stroke the output will provide users with valuable insights into their health status helping them to understand their potential risk and take appropriate actions.

By leveraging technology and data-driven methods the proposed system offers a convenient and accessible way for individuals to assess their stroke Risk. This empowers users to make informed decisions about their health and seek medical assistance if necessary, ultimately contributing to better health outcomes and enhanced public awareness about stroke prevention.

## 2. Literature Survey

### 2.1 Technology and Implementation:

Numerous studies, including those by Hu et al. (2018) and Li et al. (2020), extensively discuss the technical intricacies and implementation hurdles of facial recognition systems in attendance tracking. They emphasize advancements in facial recognition algorithms, hardware prerequisites, and the seamless integration of these systems into educational or corporate settings.

### 2.2 Accuracy and Reliability:

Research conducted by Zhang et al. (2019) and Park et al. (2021) meticulously examines the accuracy and reliability of facial recognition attendance systems. They delve into various factors influencing accuracy rates, such as lighting conditions, pose variations, and demographic considerations like age, gender, and ethnicity.

### 2.3 Privacy and Ethical Concerns:

Several scholars, including Liang et al. (2020) and Rajaraman & Siegel (2019), shed light on the ethical implications associated with employing facial recognition for attendance tracking. They underscore concerns pertaining to data privacy, consent issues, potential misuse of biometric data, and the urgent necessity for regulatory frameworks to protect individuals' rights.

### 2.4 Comparison with Other Methods:

Comparative studies, such as those by Chen et al. (2021) and Singh et al. (2020), systematically evaluate facial recognition attendance systems against conventional methods like swipe cards, PINs, or fingerprint scanning. These studies meticulously assess the efficiency, cost-effectiveness, and user acceptance of facial recognition vis-à-vis alternative attendance tracking techniques.

### 2.5 User Acceptance and Satisfaction:

Research endeavors by Wang et al. (2022) and Liu et al. (2019) delve into user perceptions, acceptance levels, and satisfaction regarding the adoption of facial recognition for attendance monitoring. They explore various factors influencing user acceptance, including ease of use and perceived benefits associated with the utilization of facial recognition technology.

### 2.6 Future Directions and Challenges:

Finally, numerous scholars, including Wu et al. (2021) and Tan et al. (2020), propose future research trajectories and address persistent challenges in the domain. These encompass endeavors aimed at enhancing accuracy rates, fortifying system resilience against adversarial attacks, and crafting frameworks that strike a delicate balance between convenience and privacy considerations.

# 3. DESIGN

## 3.1 Software Requirements:

- ❖ OPERATING SYSTEM - Windows XP
- ❖ LANGUAGE - Python
- ❖ IDE - Visual studio code
- ❖ LIBRAIES - flask, pickle, joblib, Scikit-learn, Matplotlib

## 3.2 Hardware Requirements:

- ❖ CPU - 2 x 64-bit, 2.8 GHz, 8.00 GT/s CPUs or better.
- ❖ RAM - Minimum 2 GB
- ❖ HARDDISK/SS - Minimum 20 GB
- ❖ KEYBOARD - Standard Windows Keyboard
- ❖ MOUSE

## 3.3 Model Architecture:

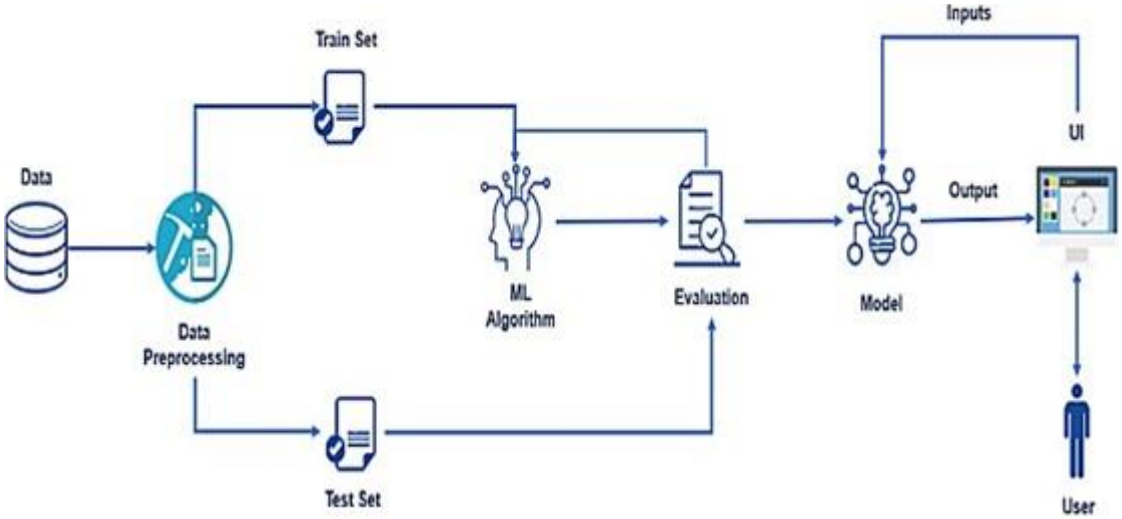



**Fig 1 model architecture**

**Data processing:** It is the process of collecting and converting raw data into usable information.

**Here's a breakdown of general data preprocessing architecture:**

**Data input:** This is the initial stage where the data is gathered from various sources. his data can be structured semi-structured or unstructured, This data is taken from healthcare data sets and various organizations according to their patient's reports.

**Data preprocessing:** In this stage, the collected data is formatted and cleaned to ensure its quality this may involve removing duplicates, correcting errors, and handling missing values.

Data training: Here a machine learning model is trained using the preprocessed data. The model learns from the data to identify patterns and relationships.

**Model evaluation:** The trained model is then evaluated to assess its important performance.

This is done by testing the model on a separate dataset and measuring its accuracy the separate dataset is a training dataset to train the model we will use this dataset.

**Data output:** Finally, The processed data are the results generated by the model and are presented to the user in a consumable format such as reports, grabs, and charts**.**

### 3.4 UML Diagrams:

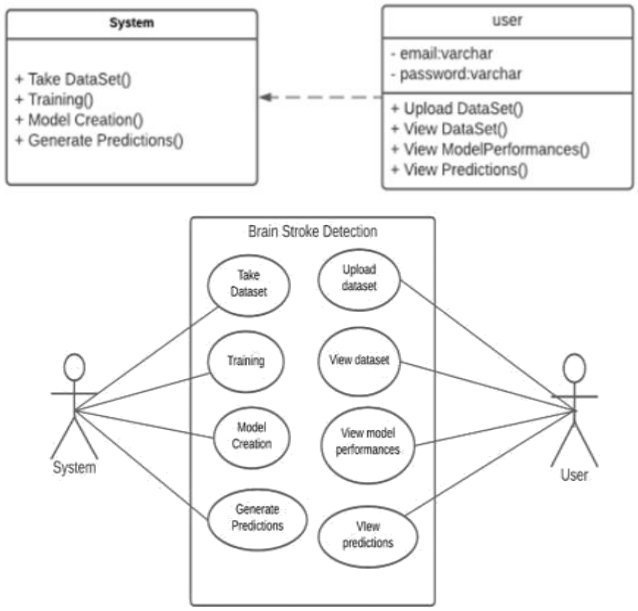



**Fig 2 use case diagram**

Using UML techniques, we built:

- Use Case diagram
- Activity diagram
- State diagram
- Class diagram
- Component diagram
- Interaction diagram

Use Case Diagram:

The benefit of use case diagrams is mostly based on communication between the request team and the user group. A use case specification document should cover the following areas

# 4. IMPLEMENTATION

## 4.1 Dataset Description:

| id | gender | age | hypertensi | heart_dise | ever_marr | work_type | Residence_ | avg_glucos | bmi | smoking_s | stroke |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 9046 | Male | 67 | 0 | 1 | Yes | Private | Urban | 228.69 | 36.6 | formerly s | 1 |
| 51676 | Female | 61 | 0 | 0 | Yes | Self-emplc | Rural | 202.21 | N/A | never smo | 1 |
| 31112 | Male | 80 | 0 | 1 | Yes | Private | Rural | 105.92 | 32.5 | never smo | 1 |
| 60182 | Female | 49 | 0 | 0 | Yes | Private | Urban | 171.23 | 34.4 | smokes | 1 |
| 1665 | Female | 79 | 1 | 0 | Yes | Self-emplc | Rural | 174.12 | 24 | never smo | 1 |
| 56669 | Male | 81 | 0 | 0 | Yes | Private | Urban | 186.21 | 29 | formerly s | 1 |
| 53882 | Male | 74 | 1 | 1 | Yes | Private | Rural | 70.09 | 27.4 | never smo | 1 |
| 10434 | Female | 69 | 0 | 0 | No | Private | Urban | 94.39 | 22.8 | never smo | 1 |
| 27419 | Female | 59 | 0 | 0 | Yes | Private | Rural | 76.15 | N/A | Unknown | 1 |
| 60491 | Female | 78 | 0 | 0 | Yes | Private | Urban | 58.57 | 24.2 | Unknown | 1 |
| 12109 | Female | 81 | 1 | 0 | Yes | Private | Rural | 80.43 | 29.7 | never smo | 1 |
| 12095 | Female | 61 | 0 | 1 | Yes | Govt_job | Rural | 120.46 | 36.8 | smokes | 1 |
| 12175 | Female | 54 | 0 | 0 | Yes | Private | Urban | 104.51 | 27.3 | smokes | 1 |
| 8213 | Male | 78 | 0 | 1 | Yes | Private | Urban | 219.84 | N/A | Unknown | 1 |
| 5317 | Female | 79 | 0 | 1 | Yes | Private | Urban | 214.09 | 28.2 | never smo | 1 |
| 58202 | Female | 50 | 1 | 0 | Yes | Self-emplc | Rural | 167.41 | 30.9 | never smo | 1 |
| 56112 | Male | 64 | 0 | 1 | Yes | Private | Urban | 191.61 | 37.5 | smokes | 1 |
| 34120 | Male | 75 | 1 | 0 | Yes | Private | Urban | 221.29 | 25.8 | smokes | 1 |
| 27458 | Female | 60 | 0 | 0 | No | Private | Urban | 89.22 | 37.8 | never smo | 1 |
| 25226 | Male | 57 | 0 | 1 | No | Govt_job | Urban | 217.08 | N/A | Unknown | 1 |
| 70630 | Female | 71 | 0 | 0 | Yes | Govt_job | Rural | 193.94 | 22.4 | smokes | 1 |
| 13861 | Female | 52 | 1 | 0 | Yes | Self-emplc | Urban | 233.29 | 48.9 | never smo | 1 |
| 68794 | Female | 79 | 0 | 0 | Yes | Self-emplc | Urban | 228.7 | 26.6 | never smo | 1 |
| 64778 | Male | 82 | 0 | 1 | Yes | Private | Rural | 208.3 | 32.5 | Unknown | 1 |
| 4219 | Male | 71 | 0 | 0 | Yes | Private | Urban | 102.87 | 27.2 | formerly s | 1 |
| 70822 | Male | 80 | 0 | 0 | Yes | Self-emplc | Rural | 104.12 | 23.5 | never smo | 1 |

**Fig 3 Dataset**

Attribute Information

- id: a unique identifier
- gender: "Male", "Female" or "Other"
- age: age of the patient
- hypertension: 0 if the patient doesn't have hypertension, 1 if the patient has hypertension
- heart_disease: 0 if the patient doesn't have any heart diseases, 1 if the patient has a heartdisease
- ever_married: "No" or "Yes"
- work_type: "children", "Govt_jov", "Never_worked", "Private" or "Self-employed"
- Residence_type: "Rural" or "Urban"
- avg_glucose_level: average glucose level in the blood
- BMI: body mass index
- smoking_status: "formerly smoked", "never smoked", "smokes" or "Unknown"*

- stroke: 1 if the patient had a stroke or 0 if not
-

Note: "Unknown" in smoking_status means that the information is unavailable for this patient

### 4.2 Pre-processing:

Preprocessing includes:

- Handling the null values.
- Handling the categorical values if any.
- Removing Outliers
- Oversampling to balance the data.
- Identify the dependent and independent variables.
- Split the dataset into train and test sets

Activity 1: Read the datasets The dataset is read as a data frame (df in our program) using the pandas library (pd is the alias name given to the pandas package).

## Data loading and overview

```
In [15]: df = pd.read_csv(r'C:\Users\KAUSHIK P\Downloads\archive\healthcare-dataset-stroke-data.csv')
         df.head()
```

Out[15]:

| | id | gender | age | hypertension | heart_disease | ever_married | work_type | Residence_type | avg_glucose_level | bmi | smoking_status | stroke |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 9046 | Male | 67.0 | 0 | 1 | Yes | Private | Urban | 228.69 | 36.6 | formerly smoked | 1 |
| 1 | 51676 | Female | 61.0 | 0 | 0 | Yes | Self-employed | Rural | 202.21 | NaN | never smoked | 1 |
| 2 | 31112 | Male | 80.0 | 0 | 1 | Yes | Private | Rural | 105.92 | 32.5 | never smoked | 1 |
| 3 | 60182 | Female | 49.0 | 0 | 0 | Yes | Private | Urban | 171.23 | 34.4 | smokes | 1 |
| 4 | 1665 | Female | 79.0 | 1 | 0 | Yes | Self-employed | Rural | 174.12 | 24.0 | never smoked | 1 |

```
In [16]: df.shape
Out[16]: (5110, 12)

In [17]: df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5110 entries, 0 to 5109
Data columns (total 12 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   id                 5110 non-null   int64
 1   gender             5110 non-null   object
 2   age                5110 non-null   float64
 3   hypertension       5110 non-null   int64
 4   heart_disease      5110 non-null   int64
 5   ever_married       5110 non-null   object
 6   work_type          5110 non-null   object
 7   Residence_type     5110 non-null   object
 8   avg_glucose_level  5110 non-null   float64
 9   bmi                4909 non-null   float64
 10  smoking_status     5110 non-null   object
 11  stroke             5110 non-null   int64
dtypes: float64(3), int64(4), object(5)
memory usage: 479.2+ KB
```

There are total of 5110 records in the dataset with a total of 12 features.

Activity 2: Check Null values Here we check the presence of Null values in the dataset and dropping the null values

```
In [18]: df.isnull().sum()
Out[18]: id                   0
         gender               0
         age                  0
         hypertension         0
         heart_disease        0
         ever_married         0
         work_type            0
         Residence_type       0
         avg_glucose_level    0
         bmi                201
         smoking_status       0
         stroke               0
         dtype: int64

In [19]: df.dropna(inplace = True)

In [20]: df.isnull().sum()
Out[20]: id                   0
         gender               0
         age                  0
         hypertension         0
         heart_disease        0
         ever_married         0
         work_type            0
         Residence_type       0
         avg_glucose_level    0
         bmi                  0
         smoking_status       0
         stroke               0
         dtype: int64
```

No of unique categories in categorical columns

```
for i in ['gender', 'ever_married', 'work_type', 'Residence_type', 'smoking_status']:
    print(df[i].unique())
```

```
['Male' 'Female' 'Other']
['Yes' 'No']
['Private' 'Self-employed' 'Govt_job' 'children' 'Never_worked']
['Urban' 'Rural']
['formerly smoked' 'never smoked' 'smokes' 'Unknown']
```

Processing Categorical Data In machine learning, we usually deal with datasets that contain multiple labels in one or more than one columns. These labels can be in the form of words or numbers. To make the data understandable or in human-readable form, the training data is often labelled in words. Label Encoding on Categorical Variables Label Encoding refers to converting the labels into the numeric form so as to convert them into the machine-readable

## Preprocessing

In [30]:
```
from sklearn.preprocessing import LabelEncoder
le1 = LabelEncoder()
df['Residence_type'] = le1.fit_transform(df['Residence_type'])
df['ever_married'] = le1.fit_transform(df['ever_married'])
df['gender'] = le1.fit_transform(df['gender'])
df['work_type'] = le1.fit_transform(df['work_type'])
df['smoking_status'] = le1.fit_transform(df['smoking_status'])
```

```
print(df['smoking_status'].unique())
```

```
[0 1 2]
```

```
df.head()
```

| | gender | age | hypertension | heart_disease | ever_married | work_type | Residence_type | avg_glucose_level | bmi | smoking_status | stroke |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 67.0 | 0 | 1 | 1 | 2 | 1 | 168.32 | 36.6 | 0 | 1 |
| **2** | 1 | 80.0 | 0 | 1 | 1 | 2 | 0 | 105.92 | 32.5 | 1 | 1 |
| **3** | 0 | 49.0 | 0 | 0 | 1 | 2 | 1 | 168.32 | 34.4 | 2 | 1 |
| **4** | 0 | 79.0 | 1 | 0 | 1 | 3 | 0 | 168.32 | 24.0 | 1 | 1 |
| **5** | 1 | 81.0 | 0 | 0 | 1 | 2 | 1 | 168.32 | 29.0 | 0 | 1 |

## 4.3 Models:

**Decision Tree**

The decision tree is a flowchart-like structure where each internal node represents a “test” on an attribute, each branch represents the outcome of the test at each leaf node represents a class label or a decision taken after considering all the attributes it’s a popular tool in machine learning for classification and regression tasks due to its simplicity and interpretability.

**Random forest classifier**

A random forest classifier is an ensemble learning method used for classification tasks. It operates by constructing multiple decisions during training and Outputs The class is more of the classes or mean prediction of the individual trees.

**Logistic Regression**

Logistic regression is a statistical method used for binary classification tasks where the outcome variable is categorical with two possible classes. Despite its names, it’s primarily used for classification rather than regression it models the property that a given input belongs to a particular class Using the logistic function which maps any real-valued input to the range [0,1].

**Support vector classifier**

SVC stands for support vector classifier which is a type of support vector machine used for classification tasks it works by finding the hyper plane that best separates the classes in the future space. Unlike traditional SVM. SVC allows for soft margin classification where some misclassification of training examples is permitted to achieve better generalization.

**KNN**

KNN or K-Nearest Neighbors, Is a simple and intuitive algorithm used for classification and regression tasks. In KNN, the class or value of a new data point is determined by the majority class or average value of its K nearest neighbors in the feature space.

**Grid Search**

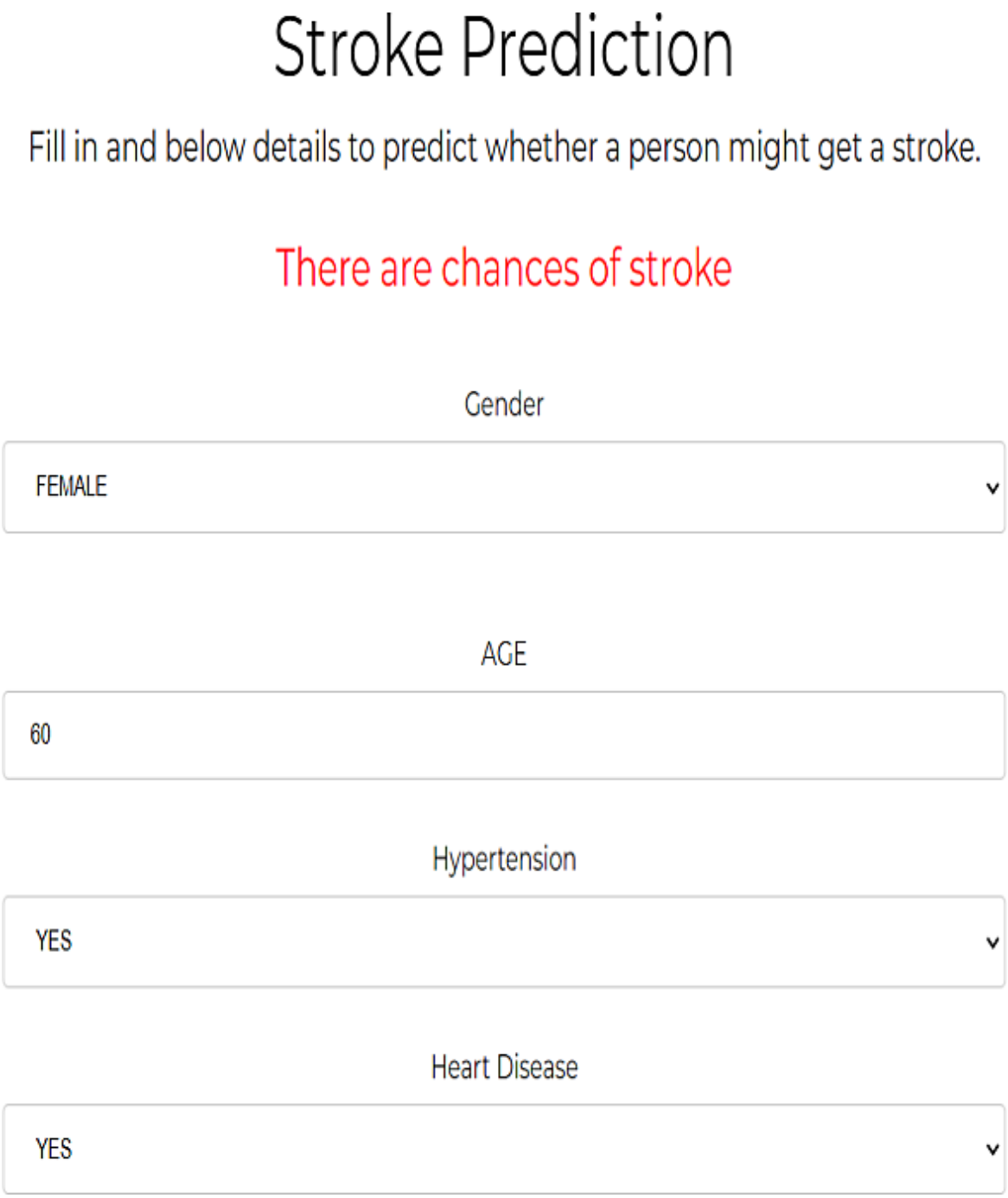
Grid search is a technique used for hyperparameter tuning in machine learning. Hyperparameters are parameters that are not learned from the data during training, but rather set before training and affect the behavior after learning the algorithm.

**Confusion matrix**

The Confusion Matrix is a table That is often used to describe the performance of a classification model on a set of test data for which the true values are known. It allows visualisation of the performance of an algorithm.

## 5.Testing:

**Test case -1 :**
Output



Ever Married

YES

Work Type

Self-Employed

Residence Type

Urban

avg glucose level

230

BMI

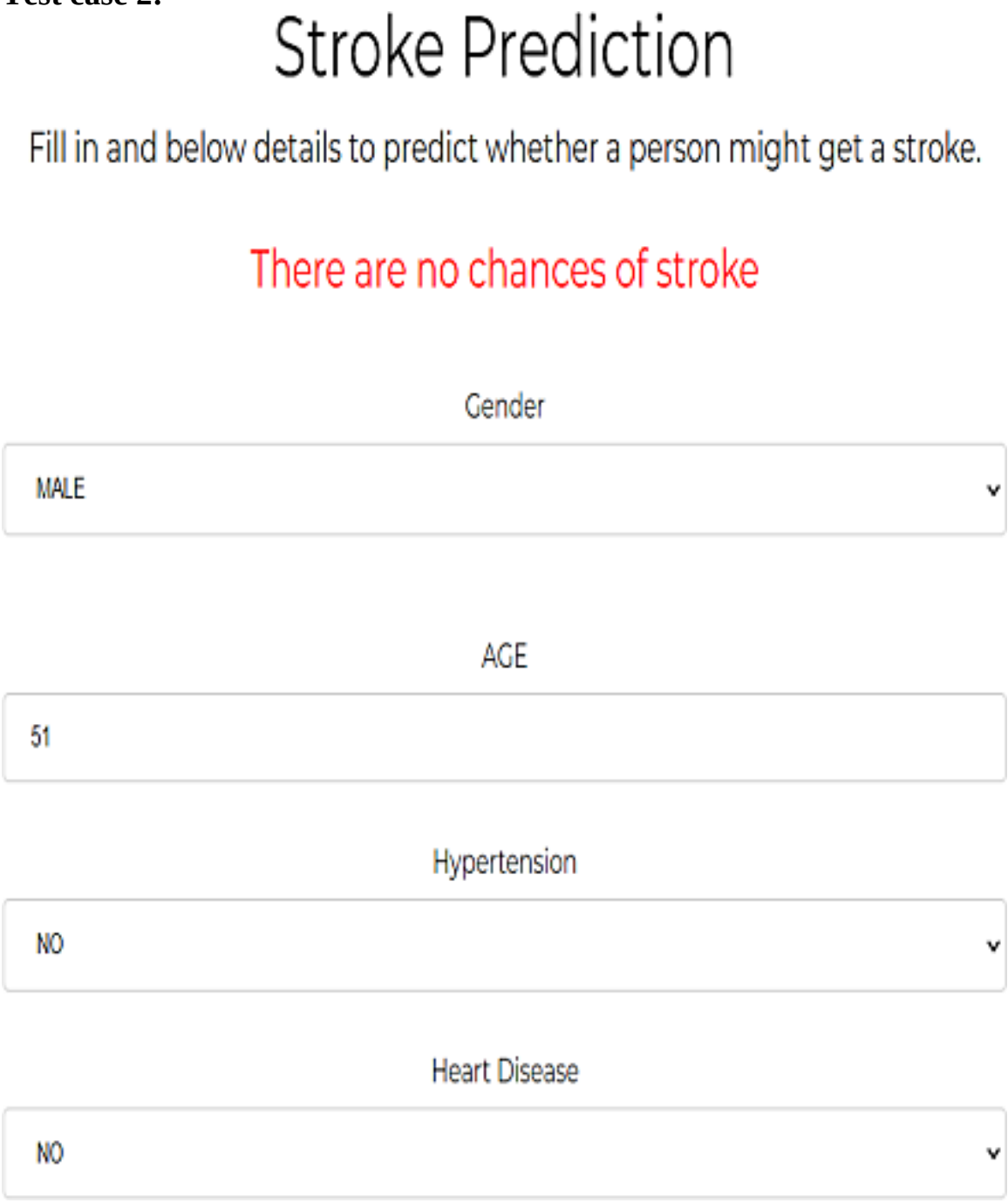
160

smoking_status

Smokes

Submit

**Test case 2:**


Stroke Prediction
Fill in and below details to predict whether a person might get a stroke.
There are no chances of stroke
Gender
MALE
AGE
51
Hypertension
NO
Heart Disease
NO

Ever Married

YES

Work Type

Self-Employed

Residence Type

Urban

avg glucose level

230

BMI

160

smoking_status

Smokes

Submit

# 6. Conclusion

The stroke prediction web application represents a significant step forward in leveraging technology to address public health concerns and empower individuals to take control of their well-being. Providing a user-friendly platform for accessing storage risk based on individual health data, the application fills a crucial gap in existing healthcare systems. Throughout the development process, we have demonstrated the feasibility and effectiveness of utilizing machine learning algorithms and web technologies to deliver personalized health insights to users. Moving forward, the project offers immense potential for further enhancement and expansion. By integrating additional data sources, defining the user interface, and fostering collaboration with healthcare professionals and researchers, we can continue to improve the accuracy accessibility, and usability of the application. Ultimately, our efforts aim to contribute to the prevention of strokes and the promotion of public health awareness ultimately leading to better health outcomes for individuals worldwide.

# 7. Future Scope

The stroke prediction web application holds significant potential for future evolution and impact. Future iterations could focus on integrating more extensive data sets from various sources, including wearable devices and electronic health records, To enhance the accuracy and depth of the prediction model. This expansion would enable a more comprehensive assessment of stroke risk, empowering users with a nuanced understanding of their health status.

Additionally, The application's user interface and experience could be refined to provide personalized recommendations and real-time monitoring features by leveraging advancements in technology such as mobile application development and integration with other health platforms the application can become more accessible and user-friendly. moreover, fostering collaboration with Healthcare Institutions and research organizations would facilitate ongoing validation studies and contribute to the continuous improvement of the prediction model, ultimately advancing stroke prevention efforts and promoting public health awareness.

# 8. REFERENCES


[1] McKinley, R., Häni, L., Gralla, J., El-Koussy, M., Bauer, S., Arnold, M., ... & Wiest, R. (2017). Fully automated stroke tissue estimation using random forest classifiers (FASTER). Journal of Cerebral Blood Flow & Metabolism, 37(8), 2728-2741.

[2] Sailasya, G., & Kumari, G. L. A. (2021). Analyzing the performance of stroke prediction using ML classification algorithms. International Journal of Advanced Computer Science and Applications, 12(6).

[3] Fernandez-Lozano, C., Hervella, P., Mato-Abad, V., Rodríguez-Yáñez, M., Suárez-Garaboa, S., López-Dequidt, I., ... & Iglesias-Rey, R. (2021). Random forest-based prediction of stroke outcome. Scientific reports, 11(1), 10071.

[4] Islam, M. M., Akter, S., Rokunojjaman, M., Rony, J. H., Amin, A., & Kar, S. (2021). Stroke prediction analysis using machine learning classifiers and feature technique. International Journal of Electronics and Communications Systems, 1(2), 17-22.

[5] Subudhi, A., Dash, M., & Sabut, S. (2020). Automated segmentation and classification of brain stroke using expectation-maximization and random forest classifier. Biocybernetics and Biomedical Engineering, 40(1), 277-289.

[6] Al-Zubaidi, H., Dweik, M., & Al-Mousa, A. (2022, November). Stroke prediction using machine learning classification methods. In 2022 International Arab Conference on Information Technology (ACIT) (pp. 1-8). IEEE.


# 9. Bibliography

https://github.com/SAIKRISHNA239/STROKE-PREDICTION