# STROKE RATE PREDICTION USING DEEP NEURAL NETWORK: A COMPARATIVE STUDY


**Submitted by**

**1. Name: Md. Redowan Chowdhury**
**(ID: CSE04180301288)**

**2. Name: Raiyan Bin Noor**
**(ID: CSE04180301281)**

**3. Name: Md. Abu Raihan**
**(ID: CSE04180101225)**


**A Project Report Submitted in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science & Engineering**



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

**NORTHERN UNIVERSITY BANGLADESH**


October 2022

# APPROVAL

The Project Report "**STROKE RATE PREDICTION USING DEEP NEURAL NETWORK: A COMPARATIVE STUDY**" submitted by **Md. Redowan Chowdhury** (ID: **CSE04180301288**), **Raiyan Bin Noor** (ID: **CSE04180301281) and Md. Abu Raihan** (ID: **CSE04180101225**) to the Department of Computer Science and Engineering, Northern University Bangladesh, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering and approved as to its style and contents.

Board of Examiners

1. Nazmin Islam (Supervisor)

2. Riasat Azim (Examiner)

3. Afrin Jubaida (Examiner)

---

Md. Raihan Ul Masood
Associate Professor and Head
Department of Computer Science and Engineering
Northern University Bangladesh

# DECLARATION

We, hereby, declare that the work presented in this Thesis report is the outcome of the investigation performed by us under the supervision of Nazmin Islam, Lecturer, Department of Computer Science and Engineering, Northern University Bangladesh. We also declare that no part of this Thesis has been or is being submitted elsewhere for the award of any degree.

Md. Redowan Chowdhury
ID: 04180301288 ……...

Raiyan Bin Noor
ID: 04180301281 ……...

Md. Abu Raihan
ID: 04180101225 ……...

**Candidates** Signature

# ABSTRACT

The Global Stroke Fact Sheet 2022 published by the World Stroke Organization (WSO) mentioned that 1 of every 4 people would definitely experience stroke in their lifetime, and this risk has increased by 50% over the last 17 years. Apart from specific risk factors, some can be changed which are responsible for 60-80% of the risk. The initial purpose of this research is to take this advantage to study the risk factors and predict stroke to ensure prior treatments, hence saving lives. The goal of this research article is to use Deep Neural Network to create a model capable of predicting Stroke outcomes based on an unbalanced dataset containing information about 5000 individuals whose Stroke outcome is known. Among the applied five activation functions, the study achieved the highest accuracy of 97.05% with the softplus activation function which is higher than previous relevant studies.

Keywords—Dataset, Data Science, Disease Prediction, Machine Learning, Stroke, Unbalanced Data, Deep Neural Network

## ACKNOWLEDGEMENT

First of all, we would like to thank the Almighty ALLAH. Today we are successful in completing our work with such ease because He gave us the ability, chance, and a cooperating supervisor.

We are indebted to a number of individuals in academic circles as well as in university faculties who have contributed to preparing the book. Their contributions are important in so many different ways that we find it difficult to acknowledge them in any other manner but alphabetically. In particular, we wish to extend our appreciation to our respected supervisor **Nazmin Islam**, Lecturer, Dept. of Computer Science & Engineering, Northern University Bangladesh, our honorable coordinator, Faculty of Science and Engineering for her valuable suggestions on preparing and improving the presentation and the book.

Again, we would like to give thanks to our honorable supervisor **Nazmin Islam** for his commitment to excellence in all aspects of the production of this book and completion of the work.

Lastly, we are grateful to our family members; who are always with us in every step of our life.

# TABLE OF CONTENTS

# List of Tables



# List of Figures

# Chapter I

## 1 INTRODUCTION

### 1.1 Background of Study

A stroke, often referred to as a brain attack, occurs when blood supply to the brain is interrupted, depriving it of oxygen and nutrients. As a result, brain cells start to degenerate within minutes [1].

After ischemic heart disease, it is the second leading cause of mortality globally, according to the World Health Organization (WHO). Paralysis, sluggishness, or visual loss are all possible symptoms of stroke. With machine learning algorithms, AI was able to develop beyond just performing the tasks it was programmed to do. Before ML entered the mainstream, AI programs were only used to automate low-level tasks in business and enterprise settings. This included tasks like intelligent automation or simple rule-based classification. This meant that AI algorithms were restricted to only the domain of what they were processed for [2].

However, with machine learning, computers were able to move past doing what they were programmed and began evolving with each iteration. Machine learning is fundamentally set apart from artificial intelligence, as it has the capability to evolve. Using various programming techniques, machine learning algorithms are able to process large amounts of data and extract useful information. In this way, they can improve upon their previous iterations by learning from the data they are provided. We cannot talk about machine learning without speaking about big data, one of the most important aspects of machine learning algorithms. Any type of AI is usually dependent on the quality of its dataset for good results, as the field makes use of statistical methods heavily. Machine learning is no exception, and a good flow of organized, varied data is required for a robust ML solution.

**Supervised Machine Learning** is one of the most basic types of machine learning. In this type, the machine learning algorithm is trained on labeled data. Even though the data needs to be labeled accurately for this method to work, supervised learning is extremely powerful when used in the right circumstances. In supervised learning, the ML algorithm is given a small training dataset to work with. This training dataset is a smaller part of the bigger dataset and serves to give the algorithm a basic idea of the problem, solution, and data points to be dealt with. The training dataset is also very similar to the final dataset in its characteristics and provides the algorithm with the labeled parameters required for the problem. The algorithm

then finds relationships between the parameters given, essentially establishing a cause and effect relationship between the variables in the dataset.

At the end of the training, the algorithm has an idea of how the data works and the relationship between the input and the output. This solution is then deployed for use with the final dataset, which it learns from in the same way as the training dataset. This means that supervised machine learning algorithms will continue to improve even after being deployed, discovering new patterns and relationships as it trains itself on new data.

**Regression** predicts the output values based on input features from the data fed into the system (Real Value Prediction). Linear regression predicts the real value by going straight up to the line, and then moving horizontally to the left to find the value. Also, try to relate between dependent variables & independent variables. If *y* is dependent variable and *x* is independent variable then we know,

$$y = mx + c \tag{1}$$

**Classification** uses the training dataset to get better boundary conditions which could be used to determine each target class. Once the boundary conditions are determined, the next task is to predict the target class.

**Unsupervised Machine Learning** holds the advantage of being able to work with unlabeled data. This means that human labor is not required to make the dataset machine-readable, allowing much larger datasets to be worked on by the program. In supervised learning, the labels allow the algorithm to find the exact nature of the relationship between any two data points. However, unsupervised learning does not have labels to work off of, resulting in the creation of hidden structures. Relationships between data points are perceived by the algorithm in an abstract manner, with no input required from human beings. The creation of these hidden structures is what makes unsupervised learning algorithms versatile. Instead of a defined and set problem statement, unsupervised learning algorithms can adapt to the data by dynamically changing hidden structures. This offers more post-deployment development than supervised learning algorithms.

**Reinforcement Learning** is an area of Machine Learning. It is about taking suitable action to maximize reward in a particular situation. It is employed by various software and machines to find the best possible behavior or path it should take in a specific situation. Reinforcement Learning differs from supervised learning in a way that in supervised learning the training data has the answer key with it so the model is trained with the correct answer itself whereas in

reinforcement learning, there is no answer but the reinforcement agent decides what to do to perform the given task. In the absence of a training dataset, it is bound to learn from its experience.

**Activation Function** determines the output of the neuron. Simply compare the activation function to the biological neurons which fire the signal to other connected neurons. Basically, the activation function maps the output value between *0* and *1*.

$$y = \sum wx + b \tag{2}$$

**Sigmoid Function** is an S shaped monotonic nonlinear function which maps positive value from *+0.5* to *+1* and negative value from *-0.5* to *-1*. This is widely used in shallow neural networks.

$$\phi(a) = \partial(a) = \frac{1}{(1+e^{-a})} \tag{3}$$

**Hyperbolic Tangent (Tanh)** similar to sigmoid function but with different output range. It is the ratio of hyperbolic sine and cosine function which is *0* centered.

$$\phi(a) = tanh(a) = \frac{(e^{a}-e^{-a})}{(e^{a}+e^{-a})} \tag{4}$$

**Rectified Linear Unit (ReLU)** doesn't activate all the neurons at the same time. This means that the neurons will only be activated if the output of the linear transformation is less than *0*.

$$f(x) = max(0, x) \tag{5}$$

**Swish** is as computationally efficient as Relu and shows better performance. The values for swish ranges from negative infinity to infinity.

$$f(x) = x \times sigmoid(x) \tag{6}$$

**Softplus Function** is a smooth approximation to the ReLU activation function. Sometimes it takes the place of ReLU. The equation is,

$$f(x) = ln(1 + (exp^{x})) \tag{7}$$

The derivative of Softplus function is

$$f'(x) = \frac{(exp^{x})}{(1+(exp^{x}))} = \frac{1}{(1+(exp^{-x}))} \tag{8}$$

Which is called logistic function.

**Support Vector Machines (SVMs)** are a set of supervised learning methods used for classification, regression and outlier's detection.

The advantages of support vector machines are: Effective in high dimensional spaces, still effective in cases where number of dimensions is greater than the number of samples, uses a subset of training points in the decision function (called support vectors), so it is also memory efficient. Versatile: different Kernel functions can be specified for the decision function. Common kernels are provided, but it is also possible to specify custom kernels.

The disadvantages of support vector machines include: If the number of features is much greater than the number of samples, avoiding over-fitting in choosing Kernel functions and regularization terms is crucial. SVMs do not directly provide probability estimates, these are calculated using an expensive five-fold cross-validation (see Scores and probabilities, below).

The support vector machines in scikit-learn support both dense (numpy.ndarray and convertible to that by numpy.asarray) and sparse (any scipy.sparse) sample vectors as input. However, to use an SVM to make predictions for sparse data, it must have been fit on such data. For optimal performance, use C-ordered numpy.ndarray (dense) or scipy.sparse.csr_matrix (sparse) with dtype=float64.

**Random Forest** 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.

One of the most important features of the Random Forest Algorithm is that it can handle the data set containing continuous variables as in the case of regression and categorical variables as in the case of classification. It performs better results for classification problems.

**Decision Trees** are a non- parametric supervised learning method used for classification and regression. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. A true can be seen as piecewise constant approximation.

## 1.2 Objective

While certain stroke risk factors, such as age, gender, race, and family history of cerebrovascular illnesses, cannot be changed, others may and are thought to be responsible for between 60% and 80% of all stroke risks in the general population [3].

A stroke happens when a blood artery in the brain bursts and bleeds, preventing blood and oxygen from reaching the brain's tissues. Ischemic Stroke (IS), Transient Ischemic Stroke (TIA), and Hemorrhagic Stroke are three different forms of strokes (HS). If we can identify and predict stroke from symptoms, it will be possible to treat many of our patients prior. There were 6.55 million stroke-related fatalities and 12.2 million event cases in 2021 [4].

Hence, by doing so the stroke rate and the death rate both will decrease. It will contribute a lot in the medical sector. Therefore, the objective is to able to predict the stroke prior to save lives.

## 1.3 Approach of Study

We predicted the stroke in our model using a machine learning method. The patient may get medical care and reduce their risk of stroke with the aid of early stroke prediction. Less persons had strokes and died from them than had strokes and were still alive. In order to extract valuable information from the vast quantity of data, strong data tools are required. Machine learning is used to forecast illness in the healthcare industry. where patient information such as name, age, blood pressure, blood sugar, etc. are kept. Multiple characteristics are used by classification algorithms to identify the illness. The values will be properly predicted using machine learning. We may use a variety of machine learning methods, and we'll choose the one that will provide the highest level of accuracy.

# Chapter II

## 2 LITERATURE REVIEW

To get the necessary understanding of numerous ideas connected to the current study of existing literature were examined. Some of the crucial conclusions drawn are mentioned here, and just a few scientists worked on Machine Learning for Stroke Prediction Some of them from recent years are detailed here.

It used an unbalanced dataset of data on 5110 people whose stroke result is known in order to utilize data analytics and machine learning to build a model that can predict stroke outcome. This study aimed to develop an accurate model for stroke outcome prediction using data science and machine learning (ML), based on historical data and individual characteristics. Supplying pertinent data that will help the medical team administer the required treatment and reduce risks and repercussions. This study demonstrated how the result of strokes might be predicted using Data Science and machine learning algorithms using information about the persons involved. Additionally, the CRISP-DM approach served as a guide for the analysis of the data, making the process easier and more effective while maintaining focus on the business challenge at hand and guiding decision-making accordingly. This study achieved 92.32 percent accuracy using Random Forrest Algorithm [5].

Five machine learning approaches were used in the Cardiovascular Health Study (CHS) dataset to predict strokes. The authors used a mix of the Decision Tree with the C4.5 method, Principal Component Analysis, Artificial Neural Networks, and Support Vector Machine to determine the ideal solution. However, the CHS Dataset used for this study contained fewer input parameters [6].

People's data collected from social media platform were used to predict strokes. The DRFS approach was used by the researchers in this study to identify the different signs and symptoms of a stroke. Using Natural Language Processing (NLP) to extract text from social media postings increases the model's execution time, which is not ideal [7].

The authors used an adapted random forest algorithm to handle the job of stroke prediction. Analyzing stroke risk levels was done this way. This strategy, according to the authors, outperformed the competition in terms of speed and accuracy. Only a small number of strokes can be studied in this way, and it cannot be applied to any additional strokes in the future [8].

Stroke prediction model was developed using Decision Tree, Random Forest, and Multi-layer Perceptron, according to a research article [9]. The three approaches yielded similar results, with very minor changes in accuracies. Decision Tree had a calculated accuracy of 74.31%, Random Forest had a calculated accuracy of 74.53%, and Multi-layer perceptron had a calculated accuracy of 75.02%. According to this study, the Multi-layer Perceptron approach is the most accurate of the three. Using just one metric to measure performance, the accuracy score cannot always offer good results.

On the Cardiovascular Health Study (CHS) dataset, the researchers conducted stroke prediction in [10]. For further effectiveness, they paired this approach with the Support Vector Machine technique. However, this led to the creation of a number of vectors that have the tendency to make the model perform worse.

The prediction of thromboembolic stroke pathology using artificial neural networks is suggested by research in [11]. The Back-propagation algorithm was employed as the prediction approach. Accuracy of around 89 percent might be obtained with this approach. However, due to their complicated structure and growing neuronal population, neural networks take longer to train and analyze information.

Computer Techniques and Applications in Biomedicine - Bora Yoo, Kyunghee Cho, Dong-wook Kim, Soon-ae Shin, Jae-woo Lee, Hyunsun Lim, and This paper's objectives included calculating the 10-year stroke prediction probability and categorizing each user's personal stroke risk into five groups.

Stroke Risk Profile from the Framingham Study: Probability of Stroke - Albert J. Belanger, William B.Kannel, MD, Philip A. Wolf, DO, Ralph B. D'Agostino, PhD This study's Framingham Study cohort was used to build a health risk evaluation function for the prediction of stroke.

According to Tasfia Ismail Shoily et AL comparisons of the Naive Bayes, J48, k-NN, and Random Forest models, the former has higher accuracy. The dataset was compiled from a variety of medical records and cross-referenced by medical professionals using WEKA (Waikato Environment for Knowledge Analysis). The proposed model will assist patients in determining whether they are at risk of having a stroke or not. 4 distinct models, including Naive Bayes, J48, k-NN, and Random Forest, were trained. To verify the models, precision and accuracy were seen. The machine learning models are applied on the dataset [12].

We may learn more about potential limitations caused by stroke using Jaehak Yu et alfavored's C4.5 decision tree method, which employs the NIHSS score and real-time variables to classify stroke severity into four categories. This information aids in predicting the potential timing of a stroke and its associated handicap, allowing for the administration of additional drugs and essential safety measures. Random Forest has a high accuracy of 88.9 percent, whereas Naive Bias has an accuracy rate of 85.4 percent [13].

A Bayesian model known as Bayesian Rule Lists (BRL) was predicted by Benjamin Letham et al., and it builds a distribution of permutations from a huge, processed collection of data. The approach scales with the least amount of the data set with numerous characteristics since the pre-processed data minimizes the model space for different sets of fragments. High accuracy, precision, and tractability may be attained with the aid of the BRL approach [14].

Pei-Wen Huang1 et al. used physiological data to predict stroke using the multimodal analysis approach. This information includes photoplethysmography, arterial blood pressure, and electrocardiography (EKG) (PPG). Each of these signals has been examined for accuracy. The three signals were combined, and they claimed that multi model analysis provides a greater accuracy for stroke prediction [15].

The information gathered from Sugam Multispecialty Hospital was used by Govindarajan et al. [11]. The dataset includes 507 patient records and 22 distinct class labels for the two main kinds of strokes. They used Decision Tree, Logistic Regression, Bagging, and Boosting, as well as Artificial Neural Networks (ANN), Support Vector Machines (SVM), and Decision Trees.

For the categorization of strokes, Sudha et al. [16] employed a Bayesian classifier, a decision tree, and a neural network. The medical institution provides the stroke dataset. Their patient details and history make up their dataset which is designed to be error-free. There are 1000 entries in the dataset. They utilized PCA to lessen the dimensionality. With 92 percent for the neural network, 91 percent for the naive Bayes classifier, and 94 percent for the decision tree, they have the highest accuracy.

Predictive methods for stroke disease span from straightforward models to more intricate ones. Stroke risk factors are intricate and used to identify two distinct complexities of disease and uncertainty from direct and/or indirect sources. Stepwise regression techniques were used for the analysis of stroke patients admitted to the TOAST study. 20 clinical variables were chosen for this study's performance finding and evaluation, which involved 1,266 stroke patients from

a database who had experienced a transient ischemic attack (TIA) or recurrent stroke within three months of their initial stroke. The Cox proportional hazards regression model, which was adjusted for any confounding factors, was used to investigate the predictive importance of blood pressure for stroke risk. Multiple measures' findings demonstrated that the predictive value of home blood pressure grew over time. Compared to traditional blood pressure measurements, the initial home blood pressure values (one measurement) demonstrated a considerably stronger relationship with stroke risk.

On the Cardiovascular Health Study (CHS) dataset, the Cox proportional hazards model and machine learning technique have been compared for stroke prediction. They specifically took into account feature selection, data imputation, and common issues with prediction in medical datasets. This study suggests using a novel feature selection method that selects reliable characteristics based on the conservative mean heuristic. Support vector machines were used in conjunction with this approach (SVMs). Comparing the feature selection process to the Cox proportional hazards model and the L1 regularized Cox model, a larger area under the ROC curve (AUC) is obtained. The technique was also used to forecast various diseases clinically in cases where there were a lot of missing data and unclear risk factors. The MarketScan Medicaid Multi-State Database (MDCD) with Atrial Fibrillation (AF) symptom was used to construct the Bayesian Rule Lists derived stroke prediction model [4]. 12,586 patients were categorized in the database based on their AF diagnoses. A one-year observation before the diagnosis and a one-year observation after the diagnosis made up the observation. According to the findings, 1,786 people experienced a stroke within a year of developing atrial fibrillation.

# Chapter III

## 3 METHODOLOGY

The Deep Neural Network (DNN) approach, which is shown below, was employed in this suggested system to predict strokes using various activation functions.

### 3.1 Deep Neural Network

A Neural Network is a very powerful machine learning mechanism which basically mimics how a human brain learns. The brain receives the stimulus from the outside world, does the processing on the input, and then generates the output. Each neuron is characterized by its weight, bias and activation function.

$$x = (weight * input) + bias \tag{9}$$

$$y = activation\,(\sum (weight * input + bias)) \tag{10}$$

Finally, the output from the activation function moves to the next hidden layer and the same process is repeated. The forward movement of information is known as forward propagation. Based on this error value, the weights and biases of the neurons are updated. This process is known as back propagation.

DNN has become one of the most debated and studied topics in the modern world. This artificial neural algorithm, which was inspired by the human brain, is in charge of several fields. This model starts with multiple layers of weighted input and produces output. The relevance of the input data is dominated by weight [17]. For initializing the weight, Glorot uniform initializer has been introduced. Iterating through the data during forward propagation results in a cost function that specifies the distinction between genuine input and artificial intelligence. Cycle relapse modification of the weights, sometimes referred to as backpropagation, is used to reduce the loss Gradient batch descent.

Deep neural networks (DNNs) yield state-of-the-art performance in numerous applications in the field of machine learning and artificial intelligence. Compared to traditional machine learning algorithms such as support vector machines, perceptron, decision trees, and k-nearest neighbors, DNNs have significant advantages in extracting features at different levels of abstraction and thereby learning more complex patterns. DNNs compute their internal parameters in forward pass and then iteratively refine them during backpropagation to

effectively extract input data features. These advantages give DNNs a greater learning capability, outperforming other methods in tasks such as computer vision, natural language processing, machine translation, speech recognition, genomics, quantitative trading, and self-driving cars. DNNs have become a powerful and valuable tool in many industrial and commercial applications. Recently, DNNs have achieved spectacular success across diverse fields: the AlphaGo and AlphaZero algorithms defeated human world champions in the game of Go [1], online entertainment platforms use DNNs to construct highly effective systems for personalized content recommendation [2], and medical professionals utilize deep learning tools to make diagnoses and discover new drugs [3], [4].

However, as the complexities of the learning tasks and the size of the training data grow, wider utilization of DNNs is challenged by a prohibitively large number of parameters, long training and inference time, and extensive computational and memory resources. For example, the ResNet-50 model contains about 23 million trainable floating-point parameters, with a model size of roughly 100 MB [18]. Such models are not viable for some applications and implementations with strict resource constraints such as edge consumer and industrial devices [19].

The parameters of typical deep learning models often have a certain degree of redundancy (or sparsity in a transform domain), which can be exploited to reduce the model's computational cost and memory footprint. In traditional approaches, there is an efficiency–accuracy trade-off: using fewer parameters reduces the cost, but also compromises the model's accuracy. Prior works considered construction of compact models while maintaining performance accuracy, such as sparsity constraints [20], weight clustering and quantification [21], knowledge distillation [22], and structured projections [23]. There are many existing researches works on addressing these challenges using tensor-based networks [24], [25].

In this chapter, we focus on the utilization of tensor decompositions and tensor networks. To address the aforementioned challenges in deep learning, we utilize tensor networks for efficient representations of high-order tensors. This approach enables compression of the parameter tensors in DNNs while preserving the interactions between features (modes). Tensor operations enable us to conduct forward and backward update calculations efficiently in tensor format. Moreover, we exploit the power of parallelization on GPUs to accelerate neural network training and inference [17]. These techniques will enable DNNs to be used in a broader range of applications by lowering the hardware requirements and computational cost and enable deployment on portable smart devices and embedded systems. There are 3 types of DNN:

- ANN (Artificial Neural Network)- Pretrain Model, Nonlinear, forward direction
- CNN (Convolution Neural Network)- Video & Image, Filters/Kernels, Conventional operation
- RNN (Recurrent Neural Network)- Overcome the looping constraint of ANN in the hidden layer. Capture sequential works parameter sharing.

In RNNs, the signals moving across recurrent connections act as an efficient memory for the network, allowing it to use the data stored there to more accurately forecast the values of a date series in the future..

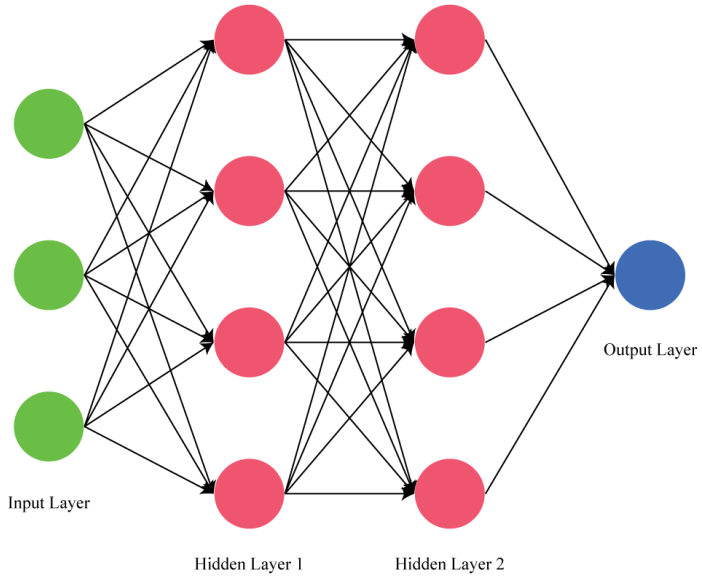


**Figure 1: Deep Neural Network Structure [AI DIARY OF ZNREZA]**

## 3.2 Data Exploration

The information was gathered via Kaggle. Based on the input variables such gender, age, illnesses, and smoking status, this dataset is used to determine whether a patient is likely to get a stroke. The data's rows each provide pertinent information about the patient. 5110 records and 11 clinical characteristics for predicting stroke episodes are included in the collection. Python and the Collaboratory notebook tool were both used by us.

As a language for our model's programming, we chose python. In terms of compiler, an open-source platform called Collaboratory that includes live code, equations, and visualization, was introduced. It can be applied to machine learning, statistical modeling, data visualization, and data cleaning and transformation.

First, we import the libraries needed to build our model in the Data Exploration stage. The software packages Matplotlib, Pandas, and libraries for NumPy. We utilized matplotlib for data analysis and numerical plotting. Pandas is a crucial library since it enables us to deal with data

structures and allows iteration, re-indexing, and sorting. Complex mathematical problems are handled by NumPy. Implementations. The dataset was then read using Panda's library.

Table 1: Attributes Name of our Dataset

| SL No. | Attributes Name | Attributes Type |
|---|---|---|
| 1 | Gender | Male/Female |
| 2 | Age | Numeric |
| 3 | Hypertension | Yes/No |
| 4 | Heart_disease | Yes/No |
| 5 | Ever_married | Yes/No |
| 6 | Residence_type | Qualitative |
| 7 | Avg_glucose_level | Numeric |
| 8 | BMI | Numeric |
| 9 | Stroke | Yes/No |

### 3.3 Processing

Making data more relevant and instructive through the process of changing it from a given form to one that is far more useable and desired

#### 3.3.1 Performance Estimation of Activation Functions

The output of a node is described by an input or group of inputs in its activation function [15]. The output of a neural network model is derived by a numerical identification. Each neuron in the model is subjected to the function, which determines whether to activate it or not by computing a weighted sum depending on whether the input from the neuron is relevant to the system's forecast. The activation function's objective is to introduce irregularity into the output. Five activation functions, including the hyperbolic tangent, sigmoid, soft plus, rectified linear unit, and swish, were assigned to the DNN model in this study. The investigated outcome is estimated to show an activation function, with 10,000 epochs and 0.001 learning rate, reveals the maximum accuracy which is higher than previous studies.

#### 3.3.2 Evaluation Parameters

The mapping of formal and actual parameters was used to evaluate the stroke dataset using the evaluation parameter. A confusion matrix can be seen as a table that shows how a

categorization model operates. Based on the test dataset, the true values are examined as Table *2*: Confusion Matrix

Table 2: Confusion Matrix

| | Predicted 0 | Predicted 1 |
|---|---|---|
| Actual 0 | TN | FP |
| Actual 1 | FN | TP |

It offers metrics parameters for recall, F-measure, accuracy, and precision. Traditionally, these variables provide the final performance evaluation result [26].

$$accuracy = \frac{true\ positive + true\ negative}{true\ positive + true\ negative + false\ positive + false\ negative} \quad (11)$$

$$precision = \frac{true\ positive}{true\ positive + false\ positive} \quad (12)$$

$$recall = \frac{true\ positive}{true\ positive + false\ negative} \quad (13)$$

$$F - measure = \frac{2 \times true\ positive}{2 \times true\ positive + false\ positive + false\ negative} \quad (14)$$

### 3.4 Environment

In the accomplishment of this study, implementing neural networks and modifying the data was the main crucial part. Hence, we incorporated some popular and mandatory tools to achieve the goal. In this part of the book, the tools are being discussed here in detail.

#### 3.4.1 Programming Language

As the programming language, python has been used. Python is one of the most popular programming languages in the current world. Due to its compatibility to multiple paradigms including structured programming, object-oriented programming, and functional

programming. The language itself is so versatile because of its thorough standard library. The language can be read very easily. It compiles English keywords which can be very understandable in any circumstances. When other languages put punctuation, python remains arranged. There is no use of other brackets or semicolons at the end of a line, also the exception of syntax is very low.

Along with if, else, and while, there are keywords such as, raise, def, with, continue, pass, assert, yield, etc. The words itself are explanatory.

Example:

```
n = int (input ('Type a number, and its factorial will be printed: '))
if n < 0:
    raise ValueError ('You must enter a non-negative integer')
factorial = 1
for i in range (2, n + 1):
    factorial *= i
print(factorial)
```

As already mentioned, python is a versatile programming language. Its greatest strength is its largest standard library. The very large package with a wide range includes various functionalities such as, Automation, Data analytics, Databases, Graphical user interfaces, Image processing, Machine learning, Computer networking, Scientific computing, etc. The study of predicting heart stroke was based on machine learning. Which is a primary reason to choose python as the vital tool of the study.

Deep Neural Network (DNN) was employed in this study. Nowadays Python is considered as the best programming language for machine learning. The rationales are, the language is well simple and keeps its consistency. The code of this language is precise and readable at the same time. Machine learning is complicated, yet the language is so simple to write that it is easy to develop and can be relied upon. One does not have to focus on learning the jargon or the technical aspects of a language in depth, rather one can easily learn the basics and solve the machine learning problem directly. The language is evolving, yet easy to master. Anyone who can read will understand the code and build something with that.

### 3.4.2 Compiler

To compile the python code, we need a compiler. As a compiler, we used Google Colaboratory for its live sharing and edition features. The feature was essential as we are a group of 3 and our supervisor used to look into our progress regularly.

Colaboratory, also known as "Colab", is a platform to write and run Python code in the browser. Its main features are, no configuration is required, free access to GPU, and easy sharing with partners. The features make the work very easy and efficient.

Using Colab one can accomplish many things such as importing an image dataset, training an image classifier on it, and evaluating the model, by writing a few lines of python code. The codes are executed in Google's cloud servers so one will get the power of using Google's hardware. Whatever the configuration is of one's device, Colab allows the use of Google's resources including GPUs and TPUs. Only requirement is a browser, which is available to everyone.

Colab is used extensively in the machine learning community with applications including:

- Getting started with TensorFlow
- Developing and training neural networks
- Experimenting with TPUs
- Disseminating AI research
- Creating tutorials

# Chapter V

## 4 RESULT

According to Deep Neural Network, using this model would allow for higher precision and more accuracy. The classification and diagnostic accuracy with softplus activation function as hidden layer and output layer were shown to be more acceptable than others through this investigation. Its accuracy of 97.05 percent was higher than that of previous study [5]. The investigated outcome shows the activation function, with 10,000 epochs and 0.001 learning rate, reveals the maximum accuracy. With the same epochs and learning rate, the hyperbolic tangent (tanh) and sigmoid activation function exhibit the lowest accuracy, which is 67.929 percent. Table *3*: Comparison of Activation Functions the effectiveness of several activation functions.

Table 3: Comparison of Activation Functions

| **Activation Function** | | **Accuracy (for 10,000 epochs and 0.001 learning rate)** |
|---|---|---|
| **Hidden Layer** | **Output Layer** | |
| Hyperbolic Tangent(tanh) | Hyperbolic Tangent(tanh) | 69.34% |
| Sigmoid | Sigmoid | 67.92% |
| Softplus | Softplus | 97.05% |
| Rectified Linear Unit (Relu) | Rectified Linear Unit (Relu) | 67.24% |
| Swish | Swish | 79.72% |

The previous study employed 8 different Machine Learning algorithm including Decision Tree, Neural Network, XGBoost Classifier, etc. Among them, Random Forest provided the best accuracy of 92.32 percent and considered as the best performing algorithm.

The obtained confusion matrix, which illustrates how well the model performed with this accuracy obtained from 0.001 learning rate and 50,000 epochs, is

Table 4: Confusion Matrix Output

| | Predicted 0 | Predicted 1 |
|---|---|---|
| Actual 0 | 12 | 3 |
| Actual 1 | 1 | 41 |

Accuracy, precision, F1-measure, and recall assessment parameters were calculated to have values of 0.929, 0.9318, 0.953, and 0.976, respectively.

The accuracy and loss curve of the model are shown in Figure 2: Accuracy Curve for the Model & Figure *3*: Loss Curve for The Model.

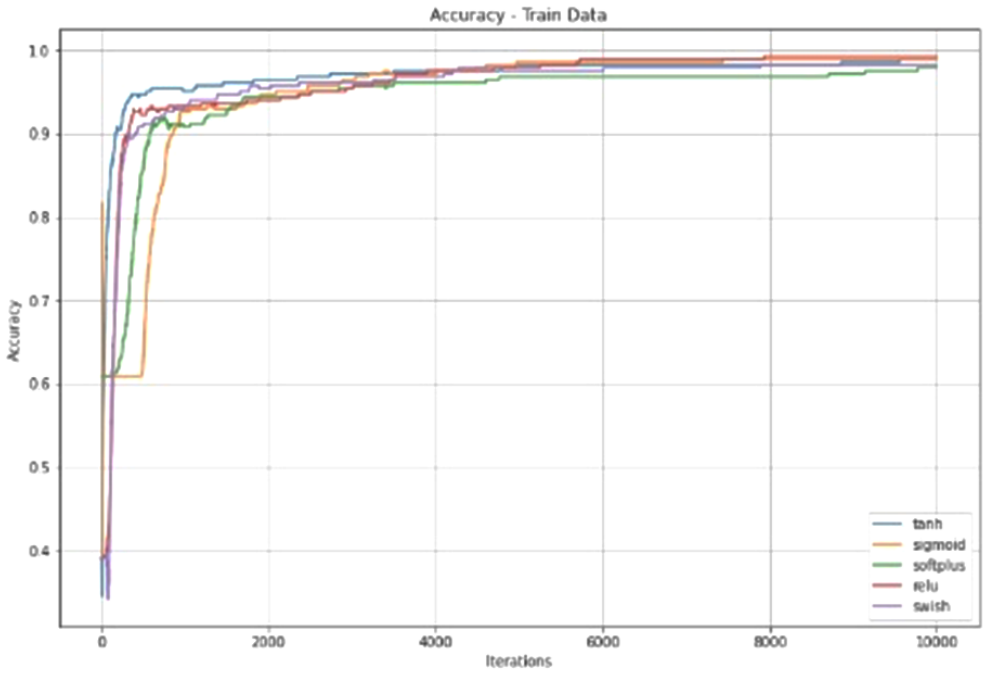


Figure 2: Accuracy Curve for the Model

The curve that is accurate during both training and validation is more significant. Precision Plot (Source: CS231n Convolutional Neural Networks for Visual Recognition) Overfitting is evidently present when there is a discrepancy in accuracy between training and validation. The overfitting increases as the gap widens.

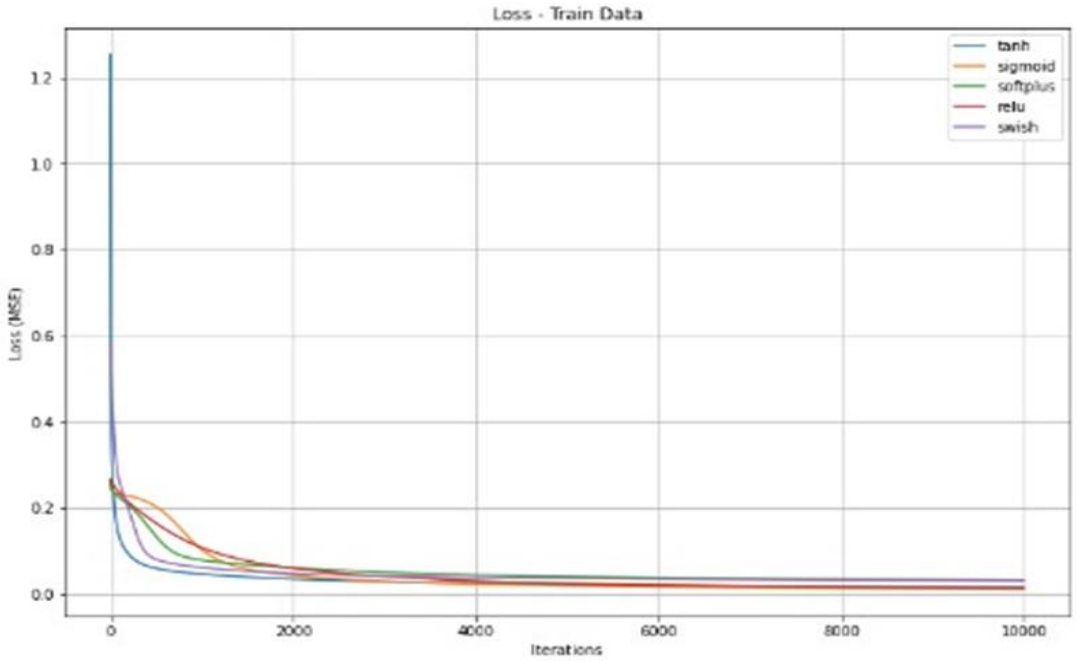


Figure 3: Loss Curve for The Model

The curve that is accurate during both training and validation is more significant. Precision Plot (Source: CS231n Convolutional Neural Networks for Visual Recognition) Overfitting is evidently present when there is a discrepancy in accuracy between training and validation. The overfitting increases as the gap widens.

The Loss curve during training is one of the most used graphs for debugging neural networks. It provides us with an overview of the training procedure and the way the network learns.

# Chapter VI

## 5 Conclusion and Future Work

In Bangladesh, stroke prevalence among the elderly has been exceptionally high. As a result of regular exposure to risk factors including hypertension [26], Bangladesh will soon face a significant challenge in the prevention and treatment of stroke. Between the two classes, the study's data were very unbalanced (stroke vs. non-stroke). As might be predicted, machine learning techniques performed poorly with the unbalanced data set. Meanwhile, the combined effectiveness of various prediction techniques increased. The data balancing process, demonstrating that it is dangerous to do prediction using inaccurate data. Using data balancing methods, excessively unbalanced classes might be successfully remedied. Avoidance is essential for precise prediction [27]. In the balanced ROS, RUS, and SMOTE/SVM data, the AUC for RLR, SVM, and RF compared to that in the unbalanced data set, respectively, sets improved significantly.

The emphasis here is the significance of data balancing methods. only in the unbalanced data set's AUC for RF and Compared to RLR, the SVM's AUC in the ROS-balanced data set exhibited improvement. The remaining models and RLR were found to vary significantly. As shown by demonstrating the application context greatly influenced how well machine learning techniques performed [28]. RLR, a traditional machine learning approach, demonstrated impressive results given its simplicity and versatility. excellent results in our research. SVM is another effective machine learning technique, and it also showed good performance in our study's ROS-balanced data set. RF is an example of a representation. when it comes to ensemble learning, which primarily combines the output from many classifiers to produce more precise projections [29]. RF functioned despite the fact that the initial data set was highly unbalanced. improved over RLR.

Our study's key factors were consistent with those previously reported in other studies. The three machine learning models all included age, hypertension, and bmi as common characteristics methods. Age was recognized as the most significant stroke risk factor in our research into demographic factors, as shown by prior research. An essential factor was hypertension. predictor, and considering how common hypertension is, prevention of it is therefore a crucial endeavor. and detrimental impact on stroke in Bangladesh. Bmi was discovered to be a standalone predictor of early mortality in stroke victims. An essential component shared by RLR and RF was avg glucose. Previous research demonstrated a deficit

in the management of Avg glucose was a crucial indicator of the prognosis of stroke. The following are some benefits of this research.

Our research also had a number of drawbacks. The study's outcome variable was self-reported stroke; As a result, there can be some personal prejudice. Additionally, the population's access to data is restricted, and included in our analysis was not sufficiently big. At the same time, over 50% of the participants dropped. Considering the high percentage of missing outcomes and predictive factors, this might possibly have some unpredictability in our findings. Furthermore, as data balancing methods have advanced, more approaches are developing, but we just covered the three that are most often utilized (ROS, SMOTE/SVM, and RUS) in this investigation. Last but not least, we merely carried out internal validation of our procedures, and future research needs external validation in big populations.

# REFERENCES


[1] S. H. Jones and S. C. Karceski, "What is a stroke?," *Neurology*, vol. 89, no. 4, pp. e43-44, 2017, doi: 10.2307/j.ctvk12shv.4.

[2] WHO, "The top 10 causes of death - Factsheet," *WHO reports*, no. December 2020, pp. 1–9, 2020.

[3] M. A. Moskowitz, E. H. Lo, and C. Iadecola, "The science of stroke: Mechanisms in search of treatments," *Neuron*, vol. 67, no. 2, pp. 181–198, 2010, doi: 10.1016/j.neuron.2010.07.002.

[4] V. L. Feigin *et al.*, "Global, regional, and national burden of stroke and its risk factors, 1990-2019: A systematic analysis for the Global Burden of Disease Study 2019," *Lancet Neurol*, vol. 20, no. 10, pp. 1–26, 2021, doi: 10.1016/S1474-4422(21)00252-0.

[5] J. A. Tavares Rodríguez, "Stroke prediction through Data Science and Machine Learning Algorithms," *School of Engineering and Sciences Tecnológico de Monterrey Monterrey, México*, 2021, doi: 10.13140/RG.2.2.33027.43040.

[6] M. K. Bhuyan, *Computer Vision and Image Processing*. 2019. doi: 10.1201/9781351248396.

[7] S. Pradeepa, K. R. Manjula, S. Vimal, M. S. Khan, N. Chilamkurti, and A. K. Luhach, "DRFS: Detecting Risk Factor of Stroke Disease from Social Media Using Machine Learning Techniques," *Neural Process Lett*, no. 5, 2020, doi: 10.1007/s11063-020-10279-8.

[8] V. Bandi, D. Bhattacharyya, and D. Midhunchakkravarthy, "Prediction of brain stroke severity using machine learning," *Revue d'Intelligence Artificielle*, vol. 34, no. 6, pp. 753–761, 2020, doi: 10.18280/RIA.340609.

[9] C. S. Nwosu, S. Dev, P. Bhardwaj, B. Veeravalli, and D. John, "Predicting Stroke from Electronic Health Records," *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, pp. 5704–5707, 2019, doi: 10.1109/EMBC.2019.8857234.

[10] N. Prentzas, A. Nicolaides, E. Kyriacou, A. Kakas, and C. Pattichis, "Integrating machine learning with symbolic reasoning to build an explainable ai model for stroke prediction," *Proceedings - 2019 IEEE 19th International Conference on Bioinformatics and Bioengineering, BIBE 2019*, pp. 817–821, 2019, doi: 10.1109/BIBE.2019.00152.

[11] D. Shanthi, G. Sahoo, and N. Saravanan, "Designing an Artificial Neural Network Model for the Prediction of Thrombo-embolic Stroke," *International Journal of Biometrics and Bioinformatics*, vol. 3, no. 1, pp. 10–18, 2009.

[12] T. I. Shoily, T. Islam, S. Jannat, S. A. Tanna, T. M. Alif, and R. R. Ema, "Detection of Stroke Disease using Machine Learning Algorithms," *2019 10th International Conference on Computing, Communication and Networking Technologies, ICCCNT 2019*, pp. 1–6, 2019, doi: 10.1109/ICCCNT45670.2019.8944689.

[13] J. Yu *et al.*, "Semantic Analysis of NIH Stroke Scale using Machine Learning Techniques," *2019 International Conference on Platform Technology and Service, PlatCon 2019 - Proceedings*, pp. 1–5, 2019, doi: 10.1109/PlatCon.2019.8668961.

[14] B. Letham, C. Rudin, T. H. McCormick, and D. Madigan, "Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model," *Annals of Applied Statistics*, vol. 9, no. 3, pp. 1350–1371, 2015, doi: 10.1214/15-AOAS848.

[15] Sydney Caulfeild, Sarah Pak, Nathanael Yao, and Hoz Rashid, "Stroke Prediction," *Journal of Mechanics Engineering and Automation*, vol. 11, no. 6. 2021. doi: 10.17265/2159-5275/2021.06.004.

[16] A. Sudha, P. Gayathri, and N. Jaisankar, “Effective Analysis and Predictive Model of Stroke Disease using Classification Methods,” *Int J Comput Appl*, vol. 43, no. 14, pp. 26–31, 2012, doi: 10.5120/6172-8599.

[17] W. Li, H. Liu, P. Yang, and W. Xie, “Supporting regularized logistic regression privately and efficiently,” *PLoS One*, vol. 11, no. 6, pp. 1–19, 2016, doi: 10.1371/journal.pone.0156479.

[18] C. Sammut and G. I. Webb, “Front Matter,” *Encyclopedia of Machine Learning*, 2011.

[19] K. Shankar, P. Manickam, G. Devika, and M. Ilayaraja, “Optimal Feature Selection for Chronic Kidney Disease Classification using Deep Learning Classifier,” *2018 IEEE International Conference on Computational Intelligence and Computing Research, ICCIC 2018*, pp. 1–5, 2018, doi: 10.1109/ICCIC.2018.8782340.

[20] C. B. Kumar, M. V. Kumar, T. Gayathri, and S. R. Kumar, “Data Analysis and Prediction of Hepatitis Using Support Vector Machine (SVM),” *International Journal of Computer Science and Information Technologies*, vol. 5, no. 2, pp. 2235–2237, 2014, [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.662.340&rep=rep1&type=pdf

[21] N. Komal Kumar and D. Vigneswari, “Hepatitis- infectious disease prediction using classification algorithms,” *Res J Pharm Technol*, vol. 12, no. 8, pp. 3720–3725, 2019, doi: 10.5958/0974-360X.2019.00636.X.

[22] W. Luo *et al.*, “Guidelines for developing and reporting machine learning predictive models in biomedical research: A multidisciplinary view,” *J Med Internet Res*, vol. 18, no. 12, pp. 1–10, 2016, doi: 10.2196/jmir.5870.

[23] H. He and E. A. Garcia, “Learning from imbalanced data,” *IEEE Trans Knowl Data Eng*, vol. 21, no. 9, pp. 1263–1284, 2009, doi: 10.1109/TKDE.2008.239.

[24] Q. Jia, L. P. Liu, and Y. J. Wang, “Stroke in China,” *Clin Exp Pharmacol Physiol*, vol. 37, no. 2, pp. 259–264, 2010, doi: 10.1111/j.1440-1681.2009.05290.x.

[25] D. Ron, “An Experimental of Model and Theoretical SeIection Comparison Methods *,” 1995.

[26] F. Z. Dawood *et al.*, “High-Sensitivity C-Reactive Protein and Risk of Stroke in Atrial Fibrillation (from the Reasons for Geographic and Racial Differences in Stroke Study),” *American Journal of Cardiology*, vol. 118, no. 12, pp. 1826–1830, 2016, doi: 10.1016/j.amjcard.2016.08.069.

[27] P. Amarenco *et al.*, “Effects of Intense Low-Density Lipoprotein Cholesterol Reduction in Patients With Stroke or Transient Ischemic Attack,” *Stroke*, vol. 38, no. 12. pp. 3198–3204, 2007. doi: 10.1161/strokeaha.107.493106.

[28] R. Paterson, “Increased Stroke Risk is Related to a Binge Drinking Habit,” *The Journal of Emergency Medicine*, vol. 36, no. 4. p. 436, 2009. doi: 10.1016/j.jemermed.2009.01.017.

[29] “Abnormal Glucose Regulation in Patients With Acute Stroke Across China _ Enhanced Reader.pdf.”