IJMTST is UGC Approved Journal with Journal Id: 43137

International Journal for Modern Trends in Science and Technology, 8(S03):152-161, 2022

ISSN: 2455-3778 online
DOI: https://doi.org/10.46501/IJMTST08S0335
Available online at: https://www.ijmtst.com/vol8si03.html

Machine Learning Algorithm for Brain Stroke Detection

P. Durga Prasad, Dr. JMSV Ravi Kumar

Department of Computer Science and Engineering, Godavari Institute of Engineering and Technology (A), JNTUK, Kakinada.

To Cite this Article
P. Durga Prasad and Dr. JMSV Ravi Kumar. Identification Machine Learning Algorithm for Brain Stroke Detection. International Journal for Modern Trends in Science and Technology 2022, 8(S03), pp. 152-161.https://doi.org/10.46501/IJMTST08S0335

Article Info
Received: 26 April 2022; Accepted: 24 May 2022; Published: 30 May 2022.

ABSTRACT

Machine learning (ML) has become a powerful health care tool that provides accurate, fast predictive results and personalized clinical care for stroke patients. ML applications in health care are on the rise, but some research areas require further research. Therefore, this study aims to systematically review Machine Learning techniques in stroke and to classify research into two categories based on their function. You can generate predictions using seven machine learning algorithms. These are K-Nearest Neighbor, Naive Bayes, Logistic Regression, Decision Tree, Random Forest, Multilayer Perceptron (MLPNN), and Support Vector Machine.

Keywords: stroke, machine learning, Deep Learning, SVM and image quality

1.INTRODUCTION

Stroke is the leading cause of death and disability worldwide due to the high mortality and morbidity rates linked with it. As a result, getting an accurate diagnosis before starting treatment is critical to ensuring that the right treatment is given. According to the findings of this study, patients who have had an ischemic stroke are separated into two groups based on the results of a CT scan. A third group of patients includes those who have had a hemorrhagic stroke.

A blockage in a blood vessel causes an ischemic stroke, which is the most prevalent cause of this sort of event. Chiun-Li Chin and colleagues [4] investigated the diagnosis and prediction of ischemic stroke and built a method that employs CNN Deep Learning to automatically detect the disease in its early stages. There are two convolution layers in the CNN architecture used in this experiment: one that serves as a pooling layer and another that serves as a linking layer. The pooling layer uses techniques like down-sampling to decrease the amount of data and parameters used in order to avoid over fitting. The results of the classification were determined to be 90% correct, which is considered outstanding accuracy. Strokes were divided into three categories in a research by Marbun, JT. et al. [5]: normal, ischemic, and hemorrhagic. To make their classification, the researchers examined CT scan images from the website www.radiopaedia.org. The researchers also employed CNN's Deep Learning architecture to classify the data collected from the patients.

The exam had a 90 percent pass rate, which was fantastic. Adjusting the hyper parameters of a Deep Learning system, according to some researchers, enhances the accuracy of stroke diagnosis, whereas other researchers utilize the same dataset to get the same conclusion (Badriyah, Tessy et al. [6]). When the random search optimization technique and Bayesian search are used with Deep Learning hyper parameter tuning, the accuracy of Deep Learning hyper parameter tuning can reach 100%. For creating stroke predictions, scientists

developed the Support Vector Machine method, which is based on the performance of multiple kernel functions. Stroke predictions were made using this method. Finally, he and his colleagues were pleased with the outcome.

The Kernel Linear and Polynomial functions, when compared to other functions, provided the most accurate experimental results, with 91.7 and 89.0 percent accuracy, respectively. Gur Amrit Pal Singh [3] used a Multilayer Perceptions Deep Learning architecture, which is a type of neural network, as well as a range of machine learning approaches, including reinforcement learning (MLP). Due to technological restrictions, these technologies made it feasible to detect and classify lung cancer in medical data sets, which was before impossible. There were 15,750 medical pictures identified using a variety of approaches, with the MLP methodology having the highest overall accuracy of 88.55 percent. The pictures were classified using the MLP method.

Hema Haridas and Aswathy Wilson [7] studied three different approaches and discovered that they were most efficient when used in conjunction with one another. The Neural Network (NN) technique is used in the first strategy; however, in the second strategy, the Principal Component Analysis (PCA) technique and the Neural Network approach are integrated (PCA). In the third strategy, a decision tree is used to choose features, while in the second, principle component analysis (PCA) is used to reduce dimension and a neural network is used to predict a stroke before it occurs. According to a review of the three methods, each of the three comparisons uses 95.0 percent, 95.2 percent, and 97.7% of the techniques that produce the best results.

1.1. Problem statement

Using machine learning algorithms such as logistic regression, random forests naive bayes, multi-linear perception decision tree, support vector machine, and k-nearest neighbor to classify stroke patients and determine the most likely candidate for therapy [7].

1.2. Scopes of the research

This application attempts to assess whether or not a system has been hacked by malicious malware by employing machine learning techniques such as support vector machines and principal component analysis [3].

2. RELATED WORK

Badriyah, Tessy et al. (2019).Researchers revealed that data may be evaluated and taken into account for decision-making in "Improving stroke diagnostic accuracy with hyper parameter optimized deep learning." Deep Learning, which can be used to address a wide range of problems and is becoming increasingly popular, is one of the most prominent methods. Kunihiko Fukushima created Recognition, the first Convolution Neural Network model, in 1980. Yann LeCun, Leon Bottou, Joshua Bengio, and Patrick Haffner later developed Recognition. Artificial neural networks are significantly used in Deep Learning (ANNs). These graphic databases need a lot of processing power. A back propagation strategy is used in Deep Learning architecture, as it is in other Neural Networks, to optimize the weight of each perception unit. Deep Learning techniques include the DCNN (Deep Convolution Neural Networks) for image classification, the DBN (Deep Belief - Neural Networks) for speech recognition, RNN (Recurrent Neural Networks) for language translation, and QODE (Query-Oriented Deep Extraction) for multi-document summarization.N. Venkata Subramanian, et al. (2018). As part of a bigger study, the epidemiology of stroke in South, East, and South-East Asia will be examined. Stroke is one of the most common causes of death and long-term disability in the world. Stroke claimed the lives of 6.5 million individuals in 2013, resulting in 113 million disability-adjusted life years (DALYs) lost and 10.3 million new cases. According to the World Health Organization, there were 25.7 million stroke victims in 2013. At least 75.2 percent of all stroke deaths and 81 percent of all DALYs were reported in impoverished nations, which accounted for more than two-thirds of all stroke fatalities and DALYs. Asia, which contains a number of "emerging" economies, is home to more than 60% of the world's population. Except in Japan, stroke deaths are more common in Asia than in Europe, the Americas, or Australasia. In the past, researchers looked into the occurrence of stroke in East and South Asia. Three of the 11 countries in Southeast Asia, where the population is predominantly descended from South and East Asians, were recently included in a stroke study undertaken by researchers in 12 Asian countries. However, no data for any of Southeast Asia's other countries was included.Stroke

epidemiology in Southeast Asia was examined, with a focus on mortality and disability-adjusted life years (DALY), stroke subtypes, and risk factors, based on data from the Global Burden of Disease study and key publications from Asian countries. It would be easier to plan and deploy resources if we had a better grasp of the scope and diversity of stroke in this region.G. A. P. Singh and P. K. Gupta (2018).The use of machine learning to detect and classify lung cancer in people is investigated. Lung cancer is one of the most frequent malignancies, and it is a primary cause of death as a result (Cancer Research UK in Cancer mortality for common cancers). The initial stage in determining the patient's status is a CT scan of the patient's lungs. An X-ray, CT scan, or MRI could be used to accomplish this scan assessment. It's particularly tough to classify lung cancer from images obtained using a variety of imaging modalities. Lung cancer diagnosis and categorization can be greatly improved using image processing and machine learning techniques. Researchers have discovered a method for identifying and classifying CT scan pictures associated with lung cancer. These photos are initially processed using image processing techniques before being used with supervised learning methods. In this case, texture characteristics and statistical variables were merged with other retrieved features and sent to classifiers. K-nearest neighbors, support vector machines, decision trees, the multinomial naive Bayes classifier, stochastic gradient descent, random forest, and multi-layer perceptions were among the classifiers utilized in this study. They were taught and assessed on a collection of 15750 clinical photos, which included 6910 benign and 8840 malignant lung cancer images, respectively. With a result of 88.55 percent, the MLP classifier is demonstrated to be more accurate than the other classifiers.C. L. Chin et al., (2018).The application of a network-based neural network to detect early indicators of ischemic strokes initially piqued my interest, and I began researching it. The number of people who have died as a result of strokes in Taiwan has increased in recent years. After suffering a stroke, those who seek medical assistance as soon as possible have a better chance of recovering. Because a CT scan cannot determine the exact location of an ischemic stroke, doctors must rely on their own interpretation of the image. In this study, a CNN deep learning algorithm was used to build an automated system for recognizing the early indicators of a stroke. Using a CT scan of the brain, the first step will be to begin image pre-processing in order to remove any areas where a stroke is unlikely to occur. As a last step, we'll employ the Data Augmentation approach to include more patch photos. Convolution neural networks will be trained and tested using patch images, which will be fed into the network for testing and training. It took only 256 patch images to train and test a CNN module that could detect an ischemic stroke. The module was successful. According to the results of the testing, the proposed approach has a precision rating of more than 90 percent. Because of this research, doctors will be able to diagnose patients more swiftly and accurately.

3. RESEARCH METHODOLOGY

3.1. Existing method

Despite the fact that Ayurvedic treatments are widely used by stroke victims, there is no documented data to support their effectiveness or safety [5]. Ayurvedic remedies for the detection of strokes are only used to a limited extent in this country due to a lack of precision in the industry.

3.2. Proposed method

The application of machine learning algorithms for the discovery of stroke sickness categorization is becoming more popular these days, and these techniques can improve the precision with which the type of stroke can be identified and classified [7].

3.3. Algorithms

3.3.1. Support vector classifiers algorithm

The Support Vector Machine (SVM) algorithm of Supervised Machine Learning is a simple but effective technique that may be used to create regression and classification models, respectively. If the case calls for it, SVM can be used with datasets that are both linearly and nonlinearly separable, depending on the situation. Despite the fact that there is only a tiny quantity of data available, the support vector machine technique is still capable of performing miracles on the data [9].

Step 1: Load the Pandas library and the dataset using Pandas.

Step 2: Define the characteristics and target audience.

Step 3: Divide the dataset into train and test segments using sklearn before building the SVM algorithm model.

Step 4: Import the support vector classifier function from the Sklearn SVM module (SVC). Utilize the SVC function to create a Support Vector Machine model.

Step 5: Predict values using the SVM algorithm model.

Step 6: Validate the Support Vector Machine model.

3.3.2. Logistic regression

Logistic regression is a machine learning classification technique for estimating the likelihood of a categorical dependent variable occurring [10]. Binary data is used in logistic regression, and this data can be coded as either 1 or 0 depending on the outcome of the experiment (no, failure, etc.). To put it another way, the logistic regression model predicts P(Y=1) as a function of X.

Step 1. Logistic regression hypothesis

As an analogy to logistic regression, it is possible to utilise the sigmoid function to construct a classifier for logistic regression, which is similar to the sigmoid function [11]. Asymptotes are found at 0 and 1 for the logistic function, which crosses the y-axis at 0.5.

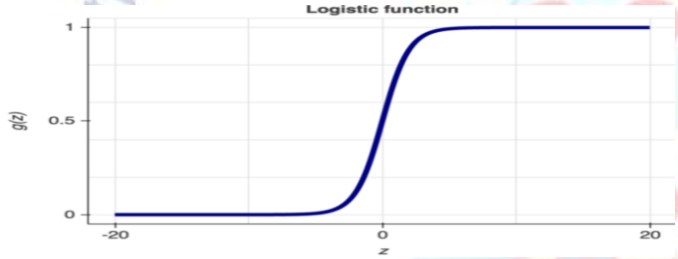

Figure.3.1. Graph of Logistic regression

Step 2. Logistic regression decision boundary

Our data set is anticipated to contain only two variables: height and weight, according to the logistic regression hypothesis, thus we will focus on those.

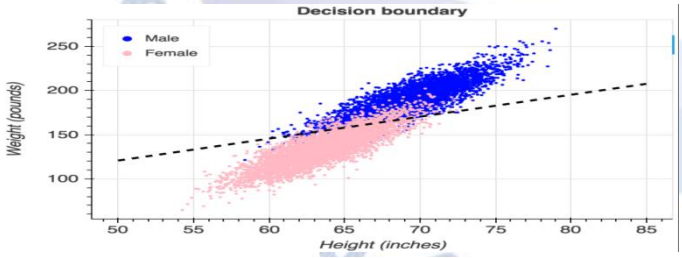

Figure.3.2. The decision boundary is plotted on a graph

3.3.3. Random forest classifier

A.*It is possible to employ random forests for both classification and regression purposes. The most prevalent applications of this technique, on the other hand, are those involving classification challenges and issues [9]. Because it is composed of a greater number of individual trees, a densely packed forest will, on average, be more robust than a sparsely packed forest. This strategy, like the random forest method, constructs decision trees based on data samples, receives forecasts from each of them, and then votes to determine which the most appropriate choice among the options is. Because it averages the outcomes, this ensemble method outperforms a single decision tree in terms of reducing over fitting compared to the single decision tree [8].*

B. Working principle of random forest classifier algorithm

C.*There are four distinct steps to the Random Forest algorithm, which are described further below.*

D. *Step 1. Begin by selecting randomly selected samples from a dataset.*

E. *Step 2. For each sample, this algorithm will create a decision tree.*

F. *Step 3. The forecast result from each decision tree will then be obtained.*

G. *Step 4. Voting will be done for each expected outcome in this step.*

H. *Step 5. Finally, choose the prediction result with the most votes as the final prediction result.*

I. *The process is illustrated in the diagram below.*

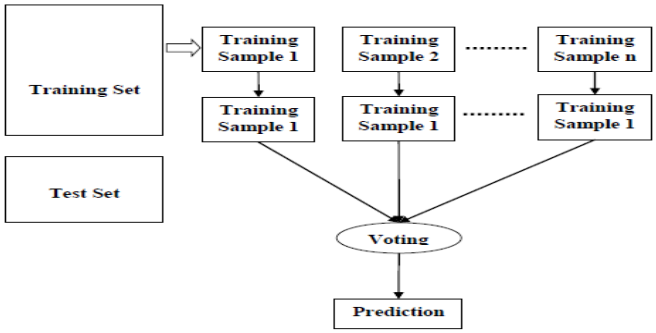

Figure.3.3. working process of random forest classifier algorithm

3.3.4. Naive Bayes algorithm

With the information we already know, we can apply Bayes Theorem to predict the likelihood that an individual piece of data falls into a certain category based on the information we already have. According to Bayes Theorem, there is a 50/50 chance of a positive outcome [10].

$$P(\text{class data}) = (P(\text{data class}) * P(\text{class})) / P(\text{data}) \quad \text{..... (3.1)}$$

Whereas:

P (class data) represents the probability of a class based on the data that has been supplied.

Step 1. Separate by class

To begin, we'll need to organise our practise data into categories. It's a really simple procedure to follow. Class values, for example, can be used as keys in dictionary objects that hold the entire record set, as seen in the following example. The list of records can then be included as a value in the expression [2].

Step 2. Summarized data-set

For every set of data that we have, we require two statistics. As we go along, we'll see how these data are incorporated into probabilistic computations in more detail. The mean and standard deviation of the dataset are the most important statistics to know (average deviation from the mean). The mean is defined as the average of all the values in a set of information.

$$\text{mean} = \text{sum}(x)/n * \text{count}(x) \quad \ldots\ldots\ldots\ldots 3.2)$$

Whereas:

The list of values or column for which we are looking is denoted by the letter x.

Step 3. Summarized data by class

It is necessary to divide down the statistics of our training dataset into categories.

A dataset is divided into rows based on the class to which each row belongs, thanks to the separate by class () function that we defined before. The summarise dataset () function can be used to produce summary statistics for each column in the dataset [5].

By bringing all of this together, we can create a summary of the columns in the dataset based on their class values.

Step 4. Gaussian probability density function

It's impossible to know how common it is to witness a real-value like X1.

You may, for instance, assume that the X1 data points come from a normal or normal-like distribution.

All that is required to sum a Gaussian distribution is the mean and standard deviation. This indicates that a given value's probability can be estimated using only basic math. How do you calculate the Gaussian Probability Distribution Function, or Gaussian PDF?

$$f(x) = (1 / \text{sqrt}(2 * PI) * \text{sigma}) * \exp(-((x-\text{mean})^2 / (2 * \text{sigma}^2))) \quad \ldots\ldots\ldots\ldots (3.3)$$

Sigma represents the standard deviation of x, the mean represents the mean, and pi represents pi.

3.3.5. Support vector classifiers algorithm

Using the basic yet powerful Support Vector Machine (SVM) approach from Supervised Machine Learning, it is possible to create both regression and classification models at the same time. With the SVM method, data sets that can be classified into linear and non-linear categories perform exceptionally well. When dealing with a small amount of data, the support vector machine technique can be used effectively [9].

Step 1. Load the Pandas library and dataset into Pandas.

Step 2. Specify the features and the end aim.

Step 3. Using sklearn, divide the dataset into train and test segments before creating the SVM algorithm model.

Step 4. Import the support vector classifier (SVC) function from the Sklearn SVM module. Create a Support Vector Machine model with the SVC function.

Step 5. Predict values using the SVM algorithm model.

Step 6. Evaluate the Support Vector Machine (SVM) model.

3.3.6. Multi-linear perception

In a multilayer perception, it is possible to have more than one linear layer, which is called a multilayer perception (combinations of neurons). As an example, let us pretend that the input layer, the output layer, and the hidden layer are all components of a three-layer network for the sake of simplicity [7 & 8]. As indicated in the diagram below, the input layer is in charge of receiving data, whilst the output layer is in charge of retrieving data from the input layer. It is possible to increase the complexity of a model by inserting additional hidden layers in the model.

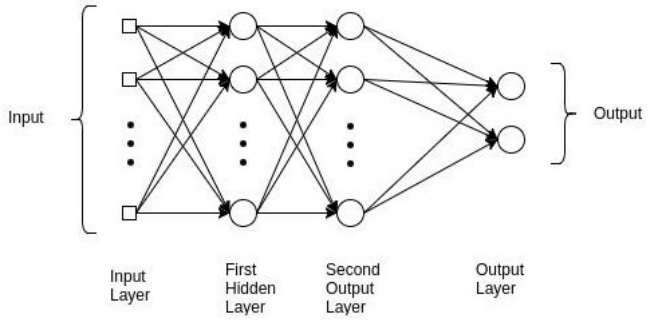

Figure.3.4. Network analysis of Multi-linear perception

This is the most common type of neural network, called a Feed Forward Network. Its goal is to get close to some function f (). This is how an MLP can find the best approximation of that classifier: It starts by learning the best parameters for a mapping that maps an input into an output class. This is done by setting up the mapping f(x) and learning the best parameters for it. The MLP networks are made up of a lot of different things that work together [5].

$$F(x) = f(3)\,(f(2)\,(f(1)\,(x)) \quad \ldots\ldots\ldots\ldots\ldots (3.4)$$

It is done by the units that make up each of these tiers. An affine transformation is done to a linear sum by these units. y = f (W x T + b) is a way to show each layer in the picture. W is the set of parameters, or weights, in the layer; x is the input vector, which can also be the output of the previous layer; and b is the bias vector. W is the set of parameters, or weights, in the layer. An MLP has several fully connected layers in each layer because each layer's units are connected to all the layers before it, so each layer is made up of several fully connected layers. It doesn't matter what the weights of the other units in a layer are [9].

Input data is given labels, which are sometimes referred to as "ground truths," which are applied to each piece of data to describe how it should be classed in the system. The output of the network is used to assign a class score to each input or to generate a prediction about it based on the class score. The classifier's efficacy and efficiency are evaluated using the loss function, which is defined in this section [11]. A huge financial loss will be incurred by the company if the predicted class turns out to be inaccurate. According to the situation, it is possible to be underprepared or over prepared during training sessions. The performance of our model is much improved when compared to the training data. Because of a lack of testing, our model has fallen short of the standards set by the industry. In order to get the most out of network training, it will be necessary to implement a loss function and an optimizer. Calculating the average weights W can aid in reducing the magnitude of the loss function's loss function [5].

Start with a random selection of weights and gradually fine-tune them until losses are kept to an absolute minimum. Throughout this stage, the gradient of the loss function is used as a guide to ensure that everything is done correctly. It is necessary to include an additional learning rate in the equation in order to indicate how far the algorithm has gone since the previous iteration.

3.3.7. K – nearest neighbor algorithm

This tutorial is broken down into three parts:

Step 1. Calculate the Euclidean distance in the binging step.

Step 2. Find Out Who Your Nearest Neighbors Are.

Step 3. Make educated guesses

Step 1. Calculate the Euclidean distance in the binging step

To begin, you must first determine the distance between two sets of data.

Straight lines are one of the easiest ways for measuring the distance between two sets of integers, and they can be used to accomplish this task. However, despite the fact that this is understandable in 2D, it also works in 3D.

$$\text{Euclidean Distance} = \text{sqrt}(\text{sum I to N } (x1\_i - x2\_i)^2) \quad \ldots(3.5)$$

For example, let's imagine that we want to sum all of the columns, in which case we'll have data in columns x1, x2, I, and so on, and so forth.

The greater the drop in Euclidean distance between two records, the lower the value between the two records becomes. There is no differentiation between two records that both have the same value in the database.

Step 2. Find Out Nearest Neighbors

We refer to our dataset's "neighbors" as the new data's k closest counterparts in our dataset.

First, it is important to compute the distance between each record in the dataset and the new data in order to determine which records are the most closely related to the new information. This will be accomplished through the usage of the distance function that we previously developed.

After the distances between the training dataset and the new data have been determined, it is necessary to sort the

records in the training dataset according to their distance from the new data, which is accomplished by sorting the records in the training dataset according to their distance from the new data. This results in the most comparable neighbors being returned as the top k neighbors.

Alternatively, the distance between each dataset item can be tracked as a tuple, and the list of tuples can be sorted by distance (in decreasing order) and retrieved in the different manner

Step 3. Make educated guesses

Identifying the datasets that are most similar to a training dataset's adjacent data sets can aid in the construction of predictions by comparing them to a training dataset. In order to categories oneself, we must choose the category that is most prevalent among our immediate surroundings.

It is necessary to apply the max () method to the output values of neighbors in order to do this. Max () selects a distinct value from the list for each class in the neighborhood and counts the number of distinct values there are in total for each class in the neighborhood.

4. SYSTEM STUDY AND TESTING

It is at this point that the project's feasibility is reviewed, and a business plan with specifics on how much money would be spent on it is provided. Using a system analysis, it will be possible to determine whether or not the suggested system is viable. In this way, the planned method will not impose a financial strain on the company. The most important thing to do before starting a feasibility study is to understand the system's most fundamental requirements [12].

5. TEST STRATEGY AND APPROACH

Manual field testing as well as extensive functional testing are currently being planned.

5.1. Test objectives

1. There must be no errors in any of the data entered.
2. Pages can only be accessed by clicking on the link provided.
3. This includes the entry screen, messages, and responses.

5.2. Feature to be tested

1. Each entry must be formatted correctly;
2. No duplicates are allowed; and
3. Each link must point to a specific page.

5.3. Integration testing

It is the process of merging software components, one at a time, with the goal of identifying and correcting errors in the interfaces that could lead to mistakes. When doing an integration test, the goal is to evaluate the interoperability of software components or applications, such as those found in a software system or those that operate on a corporate level [9].

5.4. Test results

All of the above-mentioned test scenarios were passed with flying colours. There were no issues discovered.

5.5. Acceptance testing

Affiliations with the General Public the testing phase of any project demands a significant amount of feedback from the project's end users. Therefore, it may be safely assumed that the system is up to specification.

5.5.1. Test results

All of the above-mentioned test scenarios were passed with flying colours. There were no issues discovered.

5.6. Result analysis

5.6.1. Home page

Figure.5.1. Illustration of home page

5.6.2. Upload dataset

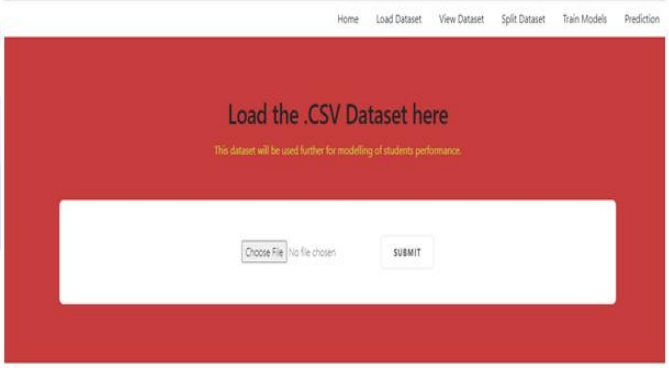

Figure.5.2. Illustration of Upload dataset

5.6.3. View dataset

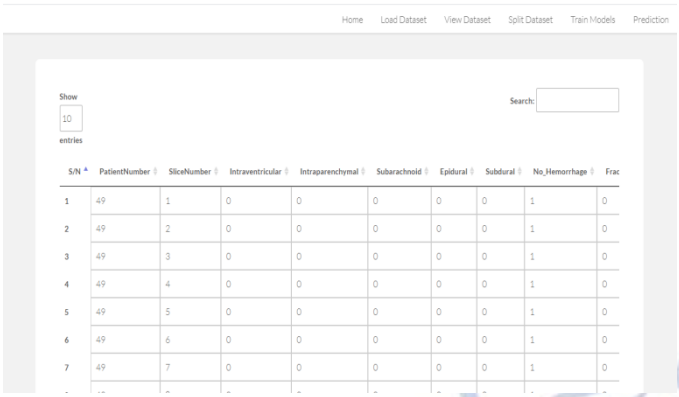

Figure.5.3. Illustration of View dataset

5.6.4. Training

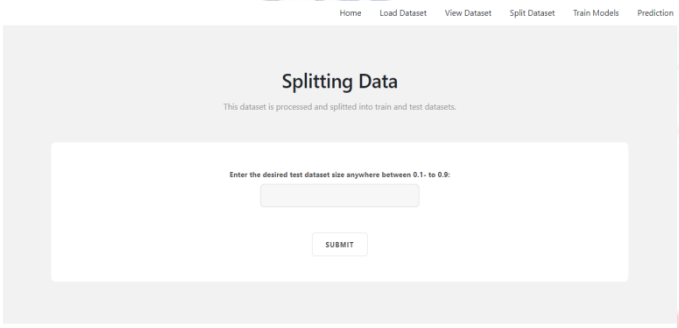

Figure.5.4. Illustration of training

5.6.5. Model performances

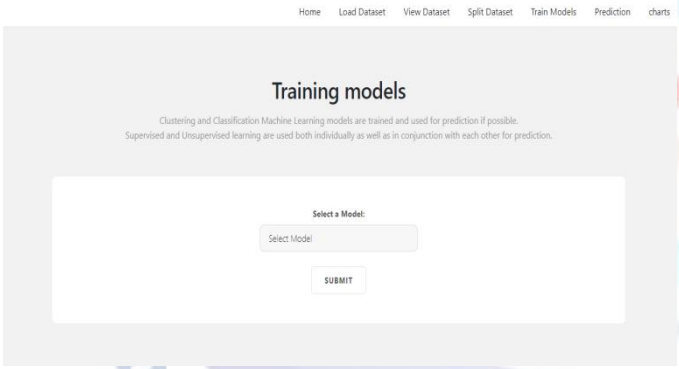

Figure.5.5. Illustration of model performances

5.6.6. Logistic regression accuracy

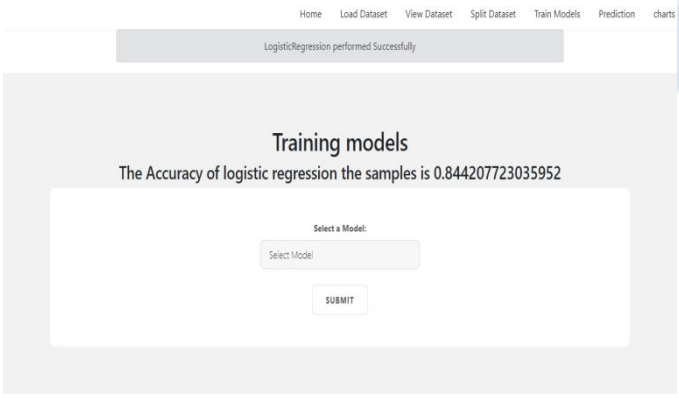

Figure.5.6. Illustration of logistic regression accuracy

5.6.7. Naive Bayes

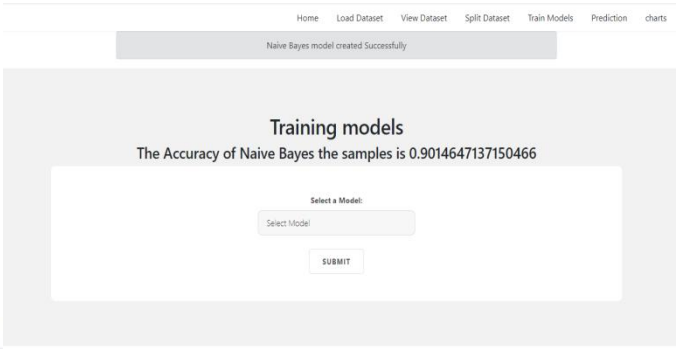

Figure.5.7. Illustration of Naive Bayes

5.6.8. KNN Model

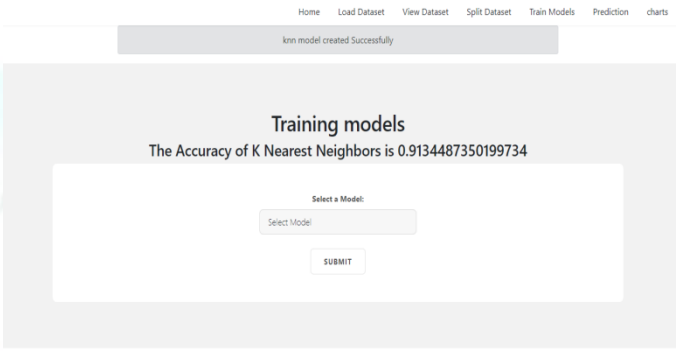

Figure.5.8. Illustration of KNN Model

5.6.9. Decision trees model

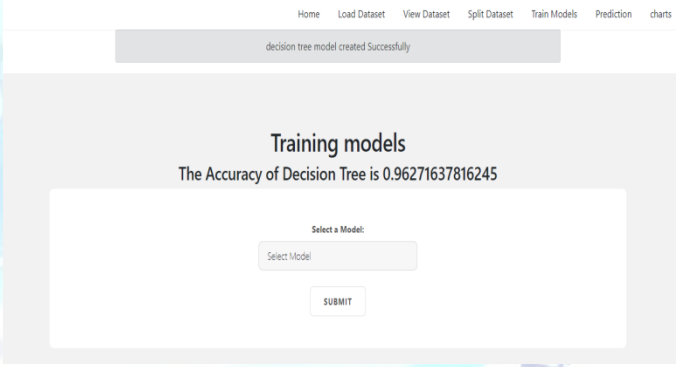

Figure.5.9. Illustration of Decision trees model

5.6.10. Random forest

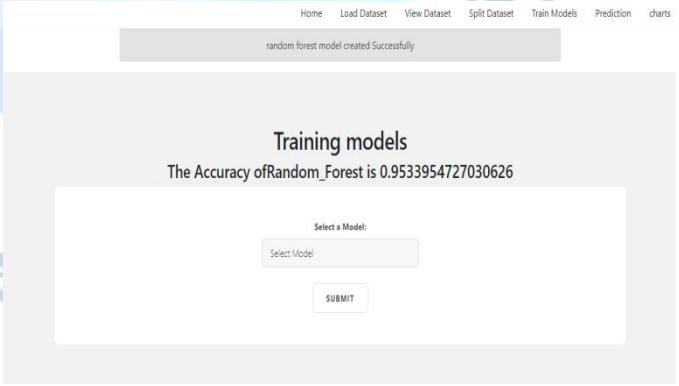

Figure.5.10. Illustration of random forest

5.6.11. MLP Model

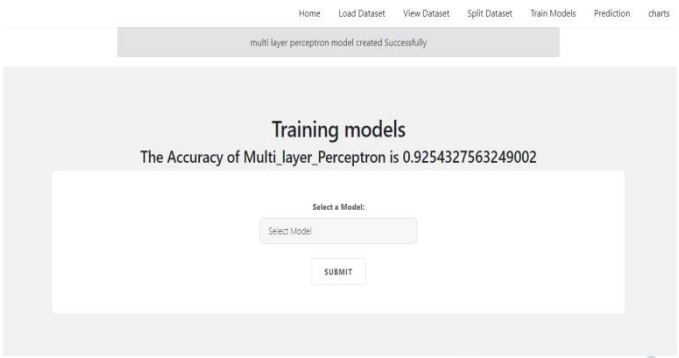

Figure.5.11. Illustration of MLP

5.6.12. SVM Model

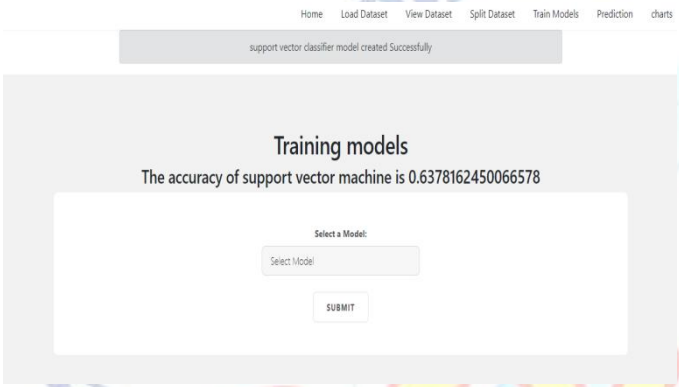

Figure.5.11. Illustration of SVM Model

5.6.13. K_fold_cross validation

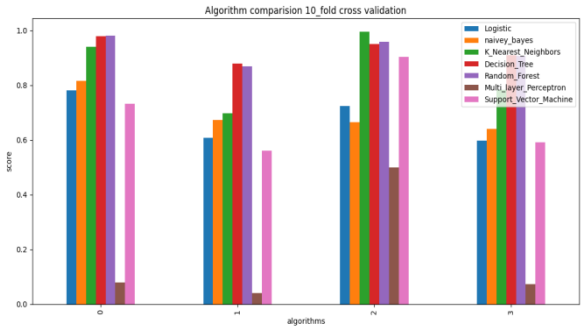

Figure.5.12. Illustration of K_fold_cross validation

5.6.14. Loo validation

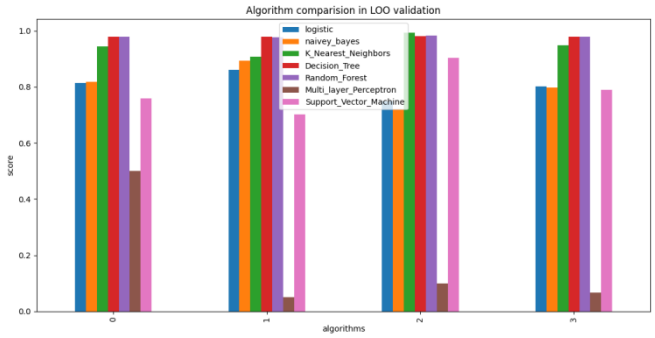

Figure.5.13. Illustration of Loo validation

5.6.15. Predictions

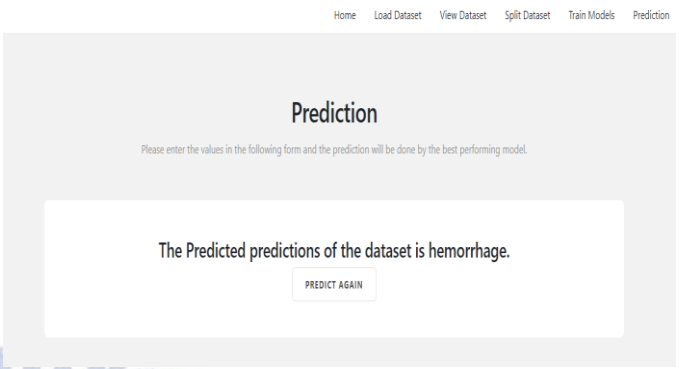

Figure.5.14. Illustration of predictions

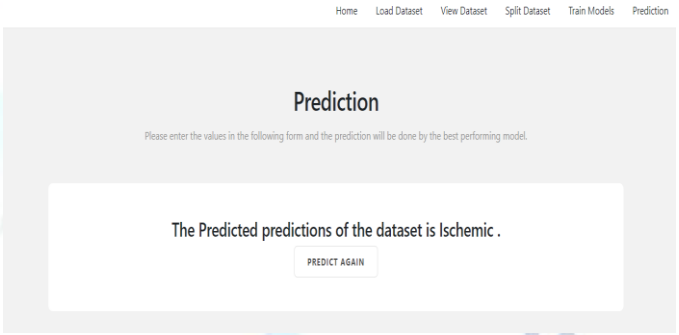

Figure.5.15. Illustration of predictions of dataset

6. CONCLUSION AND FUTURE SCOPES

6.1. Conclusion

The results of CT scans were categorised using machine learning algorithms, which were developed. Before categorization can begin, it is necessary to analyse and extract features from visual data. The classification algorithms K-Nearest Neighbours, Naive Bayes, Logistic Regression, Decision Tree, Random Forest, Multi-layer Perception (MLP-NN), Deep Learning, and Support Vector Machine (SVM) are then compared to each other for comparison. An accurate classification method, when compared to others, the random forest technique is the most accurate. However, the correctness of the default parameter value classification in optimization has not been verified yet. Through further development of the classification model, this goal may be reached in the future. In order to improve the performance of the machine learning algorithm, the parameters of the algorithm must be altered.

6.2. Future scopes

Additionally, in order to increase the effectiveness of the classifier, additional characteristics and analyses that are based on other data sources will be implemented in the near future

Conflict of interest statement

Authors declare that they do not have any conflict of interest.

REFERENCES

[1] V. L. Feigin et al., Update on the global burden of ischemic and hemorrhagic stroke in 1990-2013: The GBD 2013 study, vol. 45, no. 3. 2015.

[2] N. Venketasubramanian, B. W. Yoon, J. Pandian, and J. C. Navarro, Stroke Epidemiology in South, East, and South-East Asia: A Review, vol. 20, no. 1. 2018.

[3] G. A. P. Singh and P. K. Gupta, Performance analysis of various machine learning-based approaches for detection and classification of lung cancer in humans, vol. 3456789. Springer London, 2018.

[4] . L. Chin et al., An automated early ischemic stroke detection system using CNN deep learning algorithm, vol. 2018-Janua, no. iCAST. 2018.

[5] . T. Marbun, Seniman, and U. Andayani, Classification of stroke disease using convolutional neural network, vol. 978, no. 1. 2018.

[6] Badriyah, Tessy et al. Improving stroke diagnosis accuracy using hyperparameter optimized deep learning. International Journal of Advances in Intelligent Informatics, [S.l.], v. 5, n. 3, p. 256-272, nov. 2019. ISSN 2548-3161. doi:https://doi.org/10.26555/ijain.v5i3.427.

[7] .Haridas and A. Wilson, "A Review on Early Stroke Prediction using Artificial Intelligence," Aswathy Wilson Int. J. Innov. Adv. Comput. Sci., vol. 6, no. 11, pp. 47–49, 2017.

[8] C. Y. Hung, W. C. Chen, P. T. Lai, C. H. Lin, and C. C. Lee, Comparing deep neural network and other machine learning algorithms for stroke prediction in a large-scale population-based electronic medical claims database. 2017.

[9] Jenna R.S, Sukesh Kumar, "Stroke Prediction Using SVM", International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), 2016. \

[10] S. Paris, P. Kornprobst, J. Tumblin, and F. Durand, A gentle introduction to bilateral filtering and its applications. 2008.

[11] Alsmadi, M.K.S, Omar, K., & Noah, S. A (2009) Back Propagation Algorithm: The Best Algorithm Among the Multi-layer Perceptron Algoritma. International Journal of Computer Science and Network Security, 9 (4), PP.378-383.

[12] W. J. Powers et al., (2015). American Heart Association/American Stroke Association Focused Update of the 2013 Guidelines for the Early Management of Patients With Acute Ischemic Stroke Regarding Endovascular Treatment," Stroke, vol. 46, no. 10, pp. 3020–3035, 2015.

[13] Pattanayak, R. M., Panigrahi, S., & Behera, H. S. (2020). High-order fuzzy time series forecasting by using membership values along with Data and Support Vector Machine. Arabian Journal for Science and Engineering, 45(12), 10311-10325.

[14] Pattanayak, R. M., Behera, H. S., &Panigrahi, S. (2021). A novel probabilistic intuitionistic fuzzy set based model for high order fuzzy time series forecasting. Engineering Applications of Artificial Intelligence, 99, 104136.

[15] Pattanayak, R. M., Behera, H. S., &Panigrahi, S. (2021). A Non-Probabilistic Neutrosophic Entropy-Based Method For High-Order Fuzzy Time-Series Forecasting. Arabian Journal for Science and Engineering, 1-23.

[16] S. Panigrahi, R.M. Pattanayak, P.K. Sethy, S.K. Behera. (2021). Forecasting of Sunspot Time Series Using a Hybridization of ARIMA, ETS and SVM Methods, Solar Physics, 296(1), 1-19.

[17] Pattanayak, R. M., Behera, H. S., &Panigrahi, S. (2020). A multi-step-ahead fuzzy time series forecasting by using hybrid chemical reaction optimization with pi-sigma higher-order neural network. Computational intelligence in pattern recognition, 1029-1041.

[18] Pattanayak, R. M., Behera, H. S., &Panigrahi, S. (2020). A novel hybrid differential evolution-PSNN for fuzzy time series forecasting. In Computational Intelligence in Data Mining (pp. 675-687). Springer, Singapore.

[19] Pattanayak, R. M., Behera, H. S., & Rath, R.K.: A Higher Order Neuro-Fuzzy Time Series Forecasting Model Based on Un-equal Length of Interval. In: International Conference on Application of Robotics in Industry using Advanced Mechanisms. pp. 34–45 (2020). SPRINGER International Publishing.

[20] Pattanayak R.M., Behera H.S.: Higher order neural network and its applications: A comprehensive survey. In: Advances in Intelligent Systems and Computing. pp. 695–709 (2018), Springer, Singapore. SPRINGER International Publishing.

[21] [21]Parvathi, D. S. L., Leelavathi, N., Ravikumar, J. M. S. V., & Sujatha, B. (2020, July) Emotion Analysis Using Deep Learning. In 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC) (pp. 593-598). IEEE.

[22] Kumar, J. R., Sujatha, B., &Leelavathi, N. (2021, February). Automatic Vehicle Number Plate Recognition System Using Machine Learning. In IOP Conference Series: Materials Science and Engineering (Vol. 1074, No. 1, p. 012012). IOP Publishing."