Brain stroke severity prediction and analysis

Amisha Aggarwal
amisha19016@iiitd.ac.in

Harman Singh
harman19042@iiitd.ac.in

Meenal Gurbaxani
meenal19434@iiitd.ac.in

Yash Tanwar
yash19130@iiitd.ac.in

Abstract

Stroke is a disease that affects the arteries leading to and within the brain. A stroke occurs when a blood vessel that carries oxygen and nutrients to the brain is either blocked by a clot or ruptures. According to the WHO, stroke is the 2nd leading cause of death worldwide. Globally, 3% of the population are affected by subarachnoid hemorrhage, 10% with intracerebral hemorrhage, and the majority of 87% with ischemic stroke. 80% of the time these strokes can be prevented, so putting in place proper education on the signs of stroke is very important. The existing research is limited in predicting risk factors pertained to various types of strokes.This research work proposes an early prediction of stroke diseases by using different machine learning approaches with the occurrence of hypertension, body mass index level, average glucose level, smoking status, previous stroke and age. Machine Learning techniques including Logistic Regression, Random Forest, Decision Trees and Naive Bayes are used to predict the severity of future stroke occurrence on a scale of 0 to 3.

Code can be found here https://github.com/harmansingh25/ML_Project_2021

1. Introduction

In 2018, 1 in every six deaths from cardiovascular disease was due to stroke[1]. Someone in the United States has a stroke every 40 seconds. Every 4 minutes, someone dies of a stroke. Every year, more than 795,000 people in the United States have a stroke. About 610,000 of these are first or new strokes[2]. Stroke is a leading cause of serious long-term disability. Stroke reduces mobility in more than half of stroke survivors age 65 and over[3].

1.1. Brain Stroke

According to the definition proposed by the World Health Organization in 1970, “stroke is rapidly developing clinical signs of focal (or global) disturbance of cerebral function, with symptoms lasting 24 hours or longer, or leading to death, with no apparent cause other than of vascular origin”. A stroke occurs when blood flow to different areas of the brain gets disrupted and the cells in those regions do not get nutrients and oxygen, and as a result, start to die. Stroke is a medical emergency requiring immediate care. Early detection can help minimize further damage to the affected areas of the brain and avoid other complications in the body.

Strokes are broadly of two types. If the flow of blood among the blood tissues decreases, it is a case of ischemic stroke. On the other hand, internal bleeding among the brain tissues results in a hemorrhagic stroke.

1.2. Machine Learning for Stroke Prediction

Machine learning techniques can be used to predict the occurrence and risk of stroke in a human being.The existing research is limited in predicting whether a stroke will occur or not.

Our work attempts to predict the risk of stroke-based upon a ranking scale determined with the following criteria: 0:Low risk, 1: Moderate Risk, 2: High Risk, 3: Severe risk. This is a multiclass classification in contrast to the binary classification done by most authors earlier.

We have used features including hypertension, body mass index level, average glucose level, smoking status, previous stroke severity(Nihss score), age and gender to predict the risk of stroke for an individual. Machine Learning techniques including Logistic Regression, Random Forest, Decision Trees and Naive Bayes have been used for prediction.Our work also determines the importance of the characteristics available and determined by the dataset.Our contribution can help predict early signs and prevention of this deadly disease.

2. Literature Review

[1] Jeena et al. predict the chances of a stroke via Support Vector Machine(SVM) using the biological risk factors. After preprocessing, 12 risk factors were given

as input with 350 samples. SVM had been implemented through MATLAB with multiple different kernel functions. The error rate was used to assess the classifier's efficiency, whereas validation necessitated the calculation of sensitivity, specificity, accuracy, precision, and F1 score. They obtained the best accuracy through Linear SVM Classifier (accuracy of 91%), with the results being evaluated on a spectrum of patients of different age groups.

[2] Minhaz et al. collected data of 5110 healthy and unhealthy subjects and considered various features, including age, hypertension, BMI level, heart disease and smoking status to predict stroke incidence as a binary classification problem. They employed ten classifiers including Logistic Regression, Stochastic Gradient Descent, Decision Tree Classifier etc and finally aggregated the results of the base classifiers by using the weighted voting approach to reach the highest accuracy. Performance measures including accuracy, area under the curve, precision and recall were the highest for weighted voting approach.

[3] Shoily et al. combed through multiple datasets for a sample of size 1058 for different types of strokes(ischemic, hemorrhagic, brain stem and mini-stroke) with a total of 28 features and prepared them to use with the WEKA toolkit. They then used inbuilt algorithms from the toolset like Naive Bayes, Random Forest and J48 (The popular decision tree algorithm C4.5 is implemented in WEKA as a classifier named J48) and k-NN. Finally, they compared the accuracy, precision, recall and F1-score to conclude that Naive Bayes gave the best result with an accuracy of 85.6

3. Dataset Details and Preprocessing

3.1. Dataset Details

The dataset consists of 4798 records of patients out of which 3122 are males and 1676 are females. There are 12 primary features describing the dataset with one feature being the target variable. The description about the primary features is given in the following table.

Name	Min	Max	Mean
Age	1.0	90.0	47.116
Gender	-	-	-
NIHSS Score	0.0	45.0	18.124
mRS	-1.0	6.0	3.764
Systolic$_BP$	100.0	195.0	153.091
Diastolic$_BP$	59.0	135.0	103.655
Paralysis	0.0	3.0	1.362
Smoking	0.0	3.0	0.884
BMI	18.0	45.0	33.739
Cholesterol	160.0	253.0	217.531
TOS Score	-1.0	3.0	1.988

1. Age: Denoting the age of subjects
2. Gender: Male/Female
3. NIHSS Score: It evaluates the severity of the stroke. 0 - No stroke symptoms to ¿21 - Severe stroke
4. mRS : Modified Rankin scale -1: No stroke 0: No symptoms 1: No disability 2: Slight disability 3: Moderate disability 4: Moderate severe disability 5: Severe disability 6:Expired
5. Systolic BP: Denoting the systolic BP of subjects in mmHg
6. Diastolic BP: Denoting the diastolic BP of subjects in mmHg
7. Paralysis:0: Normal 1: Minor paralysis 2: Partial paralysis 3: Complete paralysis
8. Smoking: 0: Doesn't smoke 1: Formerly smoked 2: Frequently smokes 3: Regularly Smokes
9. BMI: Denoting the BMI of subjects in kg/m2
10. Cholestrol: Denoting the total cholesterol level of subjects in mg/dL
11. TOS Score: Treatment outcome score(for symptom severity) -1: No symptoms to 3: Severe symptoms

The target variable is 'Risk'- 0: Low risk 1: Moderate risk 2: High risk 3: Severe risk

Distribution of the target variable: 2 - 3852, 1 - 632, 3 - 227, 0 - 87

Distribution of gender: Male - 3122, Female 1676

Exploratory data analysis can be viewed in detail by clicking the link.

Some example plots:

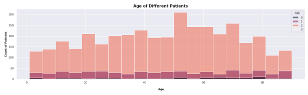


Figure 1. The severity of strokes age-wise

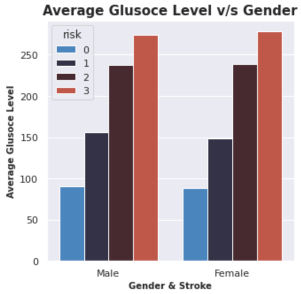


Figure 2. Average glucose level vs gender

- The age of the subjects is uniformly distributed with more patients having age greater than or equal to 50
- The risk associated with different ages is also uniformly distributed with the subjects having age greater

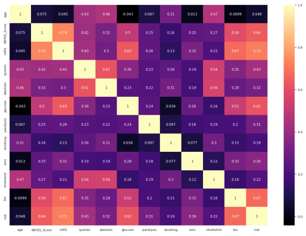


Figure 3. Getting correlation of variables

than or equal to 50 having slightly higher count amidst all risk scores

- All age brackets have similar number of risk labels

- It can be clearly seen that the average glucose levels, systolic and diastolic blood pressure levels, BMI and cholesterol are highest for risk 3 and lowest for risk 0, across both males and females.

- From the correlation table, it can be seen that the mRS , TOS, and NIHSS Score are strongly correlated with the risk score. Glucose levels and risk are also strongly correlated. NIHSS Score is strongly correlated with mRS, and glucose is also strongly correlated to mRS. More correlations can be seen from the plot.

3.2. Preprocessing Techniques

Performed data scaling using MinMax Scaler. This estimator scales and translates each feature independently to fall inside the training set's given range [0,1].
One hot encoded gender into Male and Female.One hot encoding generates new (binary) columns that indicate the presence of each potential value from the source data.

The data of risk 2 is many times higher than the that of risk 0. To overcome this, SMOTETomek was used to oversample data. Now all labels have approximately 3850 data points. SMOTE Tomek is a hybrid method that combines the two techniques above, combining an under-sampling method (Tomek) with an oversampling method (SMOTE)
Dropped columns "Unnamed: 0" and "pid" as they were irrelevant to the result.

4. Methodology

The flow of the study was as follows:
* Data Cleaning and Preprocessing
* Analysis and Visualization
* Training and Testing models
* Result Analysis

4.1. Logistic Regression

Logistic Regression is an excellent algorithm when dealing with classification problems. GridSearchCV was done to find the best parameters for Logistic Regression. The parameters included: cv(number of folds) maxIter (maximum iterations for convergence)

LogisticRegressionCV was used to perform cross-validation Logistic regression with 5 folds, max iterations = 1000 and random state = 0.

4.2. Naive Bayes Classifier

For the Naive Bayes Training Classifier, we considered a number of different algorithms to be applied until we finally selected Gaussian Naive Bayes algorithm with default parameters

This variant of Naive Bayes supports continuous values and assumes that each class is normally distributed. A Gaussian distribution describes the data with no co-variance (independent dimensions) between dimensions. At every data point, the z-score distance between that point and each class-mean is calculated

Classification Report and Confusion Matrix were used to predict accuracy. Information about probability and log-probability estimates was also provided. Cohen Kappa Score was also calculated

4.3. Decision Trees

Decision Tree algorithm: We start from the root of the tree when using Decision Trees to forecast a class label for a record. The values of the root attribute and the record's attribute are compared. We follow the branch that corresponds to that value and jump to the next node based on the comparison. The tree is constructed using criterion : Entropy or Gini impurity. Information gain and gini impurity are further calculated to split the node After this, GridSearchCV() is used to find the best model parameters in order to check if there is improvement in the evaluation metrics.

4.4. Random Forest Classifier

This algorithm creates decision trees on data samples, then gets predictions from each of them before voting on the best solution. GridSearchCV was used to find the best parameters for the Random Forest Classifier. Optimal values for these parameters included criterion (Gini/Entropy), bootstrap , Max depth (The maximum depth of the tree) and finally Random State Random Forest Classification was done for both the default value of the parameters and the best parameters. Graphs of max depth vs accuracy and number of trees vs accuracy were plotted.

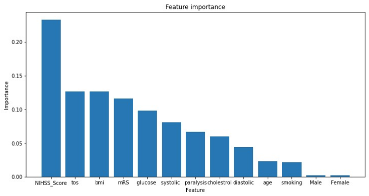


Figure 4. Feature Importance using Random Forest Model

4.5. AdaBoost Classifier

This is a meta-estimator that starts by fitting a classifier on the original dataset, then fits further copies of the classifier on the same dataset, but adjusts the weights of inaccurately classified instances so that future classifiers focus more on difficult cases.GridSearchCV was used to find the best parameters for the AdaBoost Classifier. Optimal values for these parameters included n_estimators (number of estimators) and learning rate

4.6. Support Vector Machine

The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space(N — the number of features) that distinctly classifies the data points.SVMs are also called kernelized SVM due to their kernel that converts the input data space into a higher-dimensional space.The most popular kernel functions are linear, polynomial, radial basis function and sigmoid.

We performed SVM classification for these GridSearchCV was used to find the best parameters - Kernel, Gamma(Kernel coefficient) and random state. Graphs of Kernels vs accuracy also were plotted.

4.7. K-Nearest Neighbour

It is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure (e.g., distance functions). We have used 2 different distance functions which are Euclidean distance (the shortest distance between two real-valued vectors) and Manhattan distance(the sum of the absolute differences between the two vectors);

KNeighborsClassifier was used for creating the various models and the model with minimum error and maximum accuracy is obtained at K = 1. Graphs of Error Rate vs K-Value and Accuracy vs K-Value were also plotted for both simultaneously.

4.8. Multilayer Perceptron (MLP)

This is a class of feedforward artificial neural networks (ANN). There are at least three levels of nodes in an MLP: an input layer, a hidden layer, and an output layer. Each node, with the exception of the input nodes, is a neuron with a nonlinear activation function. GridSearchCV was used to find the best parameters for the Random Forest Classifier. MLPClassifier was used to create the model from sklearn's implementation Graphs shows Accuracy vs Learning Rates were also plotted

Model	Accuracy	Precision	Recall	F1 Score
Logistic	0.9311	0.933	0.929	0.930
Gaussian NB	0.89896	0.689	0.888	0.751
Decision Tree	0.9776	0.9775	0.9776	0.9775
Random forest	0.987	0.988	0.987	0.987
SVM	0.93	0.93	0.93	0.93
KNN(Euclidean)	0.983	0.983	0.983	0.983
KNN(Manhattan)	0.981	0.981	0.981	0.981
AdaBoost	0.866	0.872	0.865	0.864
MLP	0.9617	0.9638	0.9615	0.9618

Table 1. Results

5. Results and Analysis

This table shows the accuracy, precision, recall and F1-Score for the different ML techniques using the optimal parameters. We have used Macro weightage for accuracy, precision etc. which means each label has been given equal weight.The accuracy and other metrics can be seen from Table 1.

5.1. Random Forest Classifier

The optimal parameteres using GridSearchCV are:'bootstrap': False, 'criterion': 'gini', 'maxDepth': 30, 'randomState': 23

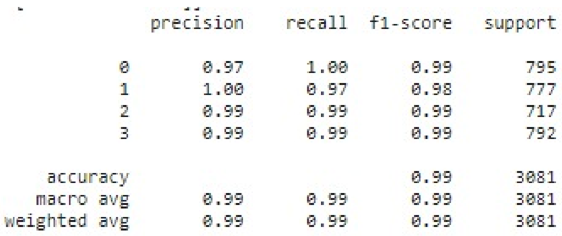

```
              precision    recall  f1-score   support

           0       0.97      1.00      0.99       795
           1       1.00      0.97      0.98       777
           2       0.99      0.99      0.99       717
           3       0.99      0.99      0.99       792

    accuracy                           0.99      3081
   macro avg       0.99      0.99      0.99      3081
weighted avg       0.99      0.99      0.99      3081
```

Figure 5. Classification Report of Random Forest Model

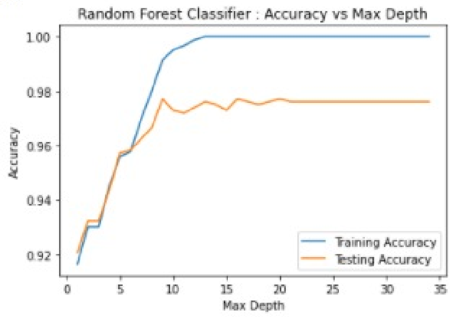
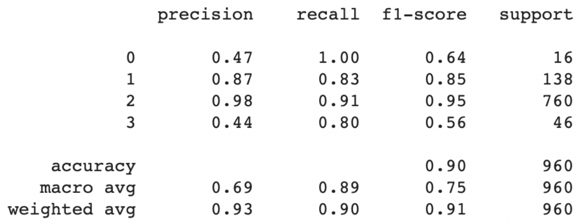


Figure 6. Accuracy vs Max Depth

5.2. Gaussian Naive Bayes Classifier

The optimal Parameters used are the default ones: ’priors’: None, ’var_smoothing’: 1e-09. We have also found out log-probability and probability estimates along with the Cohen Kappa Scores



```
              precision    recall  f1-score   support

           0       0.47      1.00      0.64        16
           1       0.87      0.83      0.85       138
           2       0.98      0.91      0.95       760
           3       0.44      0.80      0.56        46

    accuracy                           0.90       960
   macro avg       0.69      0.89      0.75       960
weighted avg       0.93      0.90      0.91       960
```

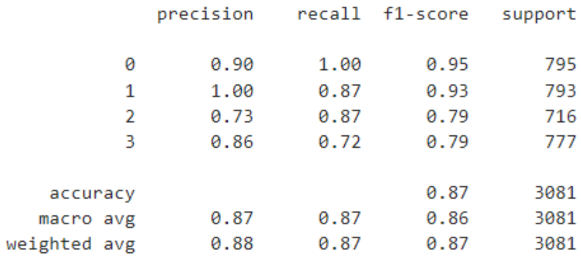
Figure 7. Classification Report of Gaussian Naive Bayes Model

5.3. Logistic Regression

GridSearchCv gave the best parameters as ’randomState’:[30], ’maxIter’:[10000]

5.4. Adaboost Classifier

The optimal parameters are : ’learning_rate’: 0.9, ’n_estimators’: 30

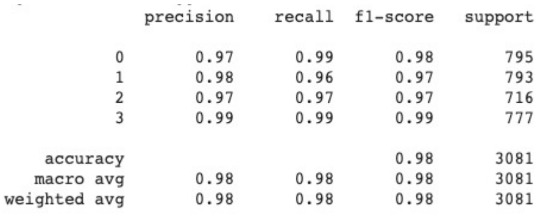


```
              precision    recall  f1-score   support

           0       0.90      1.00      0.95       795
           1       1.00      0.87      0.93       793
           2       0.73      0.87      0.79       716
           3       0.86      0.72      0.79       777

    accuracy                           0.87      3081
   macro avg       0.87      0.87      0.86      3081
weighted avg       0.88      0.87      0.87      3081
```

Figure 8. Classification Report of Adaboost Classifier Model

5.5. Decision Tree Classifier

The optimal Parameters are : ’criterion’: ’entropy’, ’max_depth’: 30, ’max_features’: ’auto’, ’max_leaf_nodes’: None, ’random_state’: 2



```
              precision    recall  f1-score   support

           0       0.97      0.99      0.98       795
           1       0.98      0.96      0.97       793
           2       0.97      0.97      0.97       716
           3       0.99      0.99      0.99       777

    accuracy                           0.98      3081
   macro avg       0.98      0.98      0.98      3081
weighted avg       0.98      0.98      0.98      3081
```

Figure 9. Classification Report of Decision Tree Model

5.6. Support Vector Machine

The optimal parameters are: ’gamma’: ’scale’, ’kernel’: ’poly’, ’random_state’: 1. In figure 10, we can see the classification report after training the model with the optimal parameters found above. In figure 11, we can see the variation of accuracy vs. kernel and it is evident that the polynomial kernel performs the best.

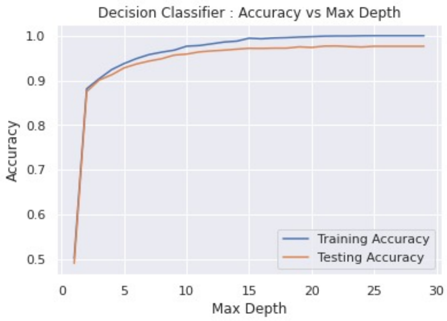
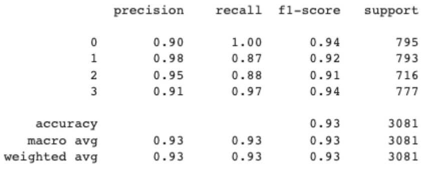


Figure 10. Accuracy vs Max Depth



```
              precision    recall  f1-score   support

           0       0.90      1.00      0.94       795
           1       0.98      0.87      0.92       793
           2       0.95      0.88      0.91       716
           3       0.91      0.97      0.94       777

    accuracy                           0.93      3081
   macro avg       0.93      0.93      0.93      3081
weighted avg       0.93      0.93      0.93      3081
```

Figure 11. Classification report of SVM

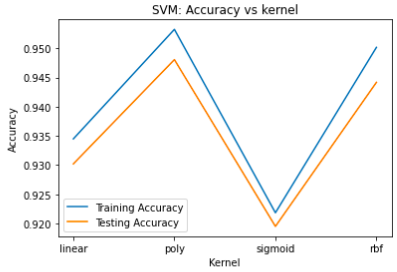


Figure 12. Accuracy vs. Kernel for SVM

5.7. K-Nearest Neighbour

KNeighborsClassifier was used for creating the various models, and the model with minimum error and maximum accuracy is obtained at K = 1. The preferred distance function between Euclidean and Manhattan distances was the Euclidean Distance function. Optimal Parameters were the default parameters used ’weights’: ’uniform’, ’algorithm’:’auto’, ’leaf_size’: 30,’metric’: ’minkowski’, ’metric_params’: None, ’n_jobs’=None.

5.8. Multi Layer Perceptron

The optimal paramenters are activation=’relu’, alpha=0.008, learning_rate_init=0.01, max_iter=100

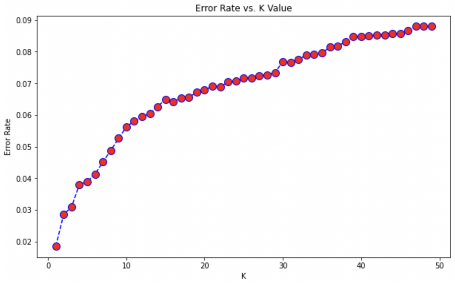


Figure 13. Error Rate vs. K - Value for KNN

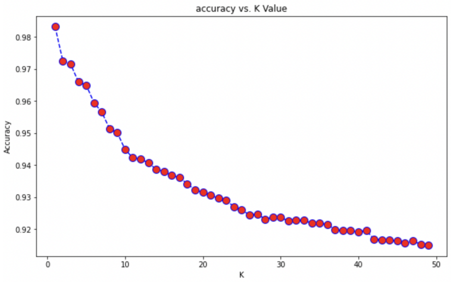
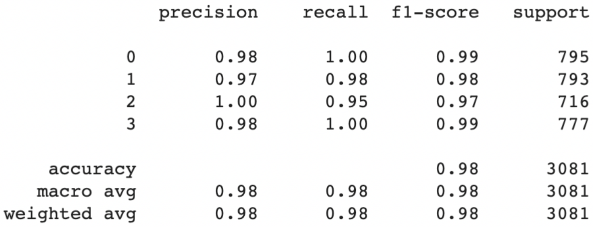
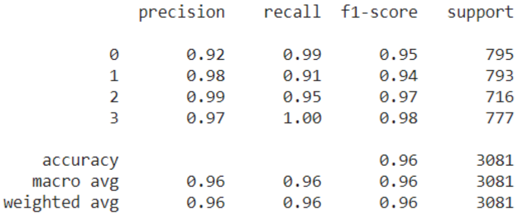


Figure 14. Accuracy Score vs. K - Value for KNN



```
              precision    recall  f1-score   support

           0       0.98      1.00      0.99       795
           1       0.97      0.98      0.98       793
           2       1.00      0.95      0.97       716
           3       0.98      1.00      0.99       777

    accuracy                           0.98      3081
   macro avg       0.98      0.98      0.98      3081
weighted avg       0.98      0.98      0.98      3081
```

Figure 15. Classification Report of Best KNN Model



```
              precision    recall  f1-score   support

           0       0.92      0.99      0.95       795
           1       0.98      0.91      0.94       793
           2       0.99      0.95      0.97       716
           3       0.97      1.00      0.98       777

    accuracy                           0.96      3081
   macro avg       0.96      0.96      0.96      3081
weighted avg       0.96      0.96      0.96      3081
```

Figure 16. Classification Report of MLP Model

6. Conclusion

We learned the workings of several Machine Learning Techniques including logistic regression,Gaussian Naive Bayes, Decision Trees and Random Forests etc.

So far the most efficient model we have found is the Random Forest Model with an accuracy of 98.7% while K-Nearest Neighbour comes a close second with 98.3% as can be seen in Table 1

We have also learned a great deal of information on strokes and it's various types(ischemic and hemorrhagic and mini strokes) and it's after effects on a human body This project has also provided us with a new perspective on human suffering and disability

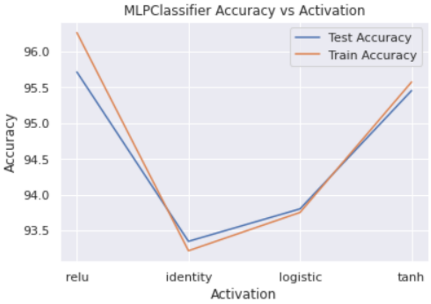


Figure 17. MLPClassifier Accuracy vs Learning Rate

7. Individual Contributions

Amisha Aggarwal - Literature Reviews, Rf, AdaBoost
Harman Singh - EDA, Decision Trees, svm
Meenal Gurbaxani - Literature Reviews, NB, KNN
Yash Tanwar - Data Preprocessing, Logistic Regression, MLP

8. Future Work

This project takes upon the idea of illness prediction via Machine learning and uses the stroke prediction models as something to build upon and improve. Certain improvements can be made to this project by focusing on important features in the Random Forest Model(the most accurate one) so as to help the at-risk individuals. Furthermore, Using these models, we can further branch out to other such diseases, including heart attacks etc., depending upon available datasets.

9. References

[1] Donkor E. S. (2018). Stroke in the 21st Century: A Snapshot of the Burden, Epidemiology, and Quality of Life. Stroke research and treatment, 2018, 3238165. https://doi.org/10.1155/2018/3238165

[2] Tianyu Liu, Wenhui Fan, Cheng Wu(2019),A hybrid machine learning approach to cerebral stroke prediction based on imbalanced medical dataset,Artificial Intelligence in Medicine,Volume 101, https://doi.org/10.1016/j.artmed.2019.101723

[3]https://www.ninds.nih.gov/Disorders/All-Disorders/Stroke-Information-Page

[4] Bandi, V., Bhattacharyya, D., Midhunchakkravarthy, D. (2020). Prediction of brain stroke severity using machine learning. Revue d'Intelligence Artificielle, Vol. 34, No. 6, pp. 753-761. https://doi.org/10.18280/ria.340609