C S-5593-995-999, Fall, 2022

Predicting the Risk of Patient Having a Stroke Using Data Mining

Authors:

Amina Asghar, amina.asghar-1@ou.edu

Nasri Binsaleh, nasri.binsaleh-1@ou.edu

Onintsoa Ramananandroniaina, ony@ou.edu

ABSTRACT

According to the World Health Organization (WHO), between the years 2000 and 2019, stroke was the leading cause of death globally. Preventing strokes and providing proper care to stroke patients can save people's lives and be a cost-effective measure for healthcare-related costs. One way to achieve that is to identify their risk of stroke early on in order to take the necessary precautions to prevent it. This project aims to create an application that can accurately predict whether a person is at risk of stroke based on their basic health and personal metrics input. To ensure the most accurate predictions, three different classification models were implemented and then compared, and the best performing model based on accuracy, precision, recall, f-1 score, and runtime is selected. The three models to be used are complement naïve bayes, decision trees, and random forest. The results of the comparison show that naïve bayes outperforms the other methods, which will be implemented in the application where the user can check their stroke risk.

SECTION 1 Introduction

This project aims to create an application that can accurately predict whether a person is at risk of stroke based on their basic health and personal metrics input. All the algorithms used in this project will be for classification. The objectives of the project are to understand the different algorithms used for classification and how they interact with the features, implement from scratch three different classification algorithms and compare the different algorithms and how they affect the outcomes of the predictions, and create an application that will take in the user's health attributes as inputs and output the risk of stroke using the best classification algorithm identified based on the performance metrics.

Prediction of severe health issues is a massive area of interest in data mining because of its wide-ranging real-life implications. According to the World Health Organization (WHO), between the years 2000 and 2019, stroke was the leading cause of death globally [WHO, 2020]. Moreover, according to the Centers for Disease Control and Prevention, 1 in 6 deaths from cardiovascular disease was due to stroke in 2020. Between 2017 and 20Amin18, stroke-related costs in the US rose to nearly 53 billion dollars [CDC, 2022]. Considering the facts mentioned earlier, preventing strokes, and providing proper care to stroke patients can save people's lives and be a cost-effective measure for healthcare-related costs.

When the project is concluded, the functionality of the application will include taking in user input data to predict the risk of the patient having a stroke based on that data and, subsequently, outputting the prediction to the user.

SECTION 2 Related Works

Several research papers employ different data mining algorithms to predict the risk of severe diseases like stroke. In an article published in the International Journal of Preventative Medicine, the authors aim to predict stroke incidence by considering factors such as a history

of cardiovascular disease, smoking, and diabetes. The authors concluded that the k-nearest neighbor and C4.5 decision tree algorithm had a high accuracy in predicting the occurrence of a stroke.

Data mining tasks predict and detect other deadly diseases, such as cancer. In an article titled, *Cancer Classification Using Gaussian Naive Bayes algorithm*, the authors aim to use Gaussian Naïve Bayes Algorithm for cancer classification. Early cancer detection is critical because early detection can drastically increase a person's survival rate. Kamel concluded that using Gaussian Naïve Bayes in combination with an efficient technique for normalization;

the Algorithm had 90% accuracy when predicting lung cancer and an even higher accuracy of 98% when predicting breast cancer. Based on the results in the article mentioned earlier, it can be concluded that Naïve Bayes Algorithm is a suitable candidate for classification when aiming to predict the cancer class.

A decision tree is another method used in data mining for classification and prediction. In an article titled "*Decision tree methods: application for classification and prediction*" by Yan-Yan Song and Ying Lu, the authors established that the decision tree model is a powerful method that can be utilized for classification, prediction, data manipulation as well as interpretation. The decision tree models enable us to handle missing values without utilizing imputation, making it easier to understand and interpret our results. Song et al. further explains the potential of decision tree methods by stating that decision tree models are robust to outliers which is crucial for analyzing data with many outliers. According to the authors, decision tree models are also an excellent method to perform classification and statistical analysis for data that is heavily skewed if we don't want to deal with the heavily skewed data by using data transformation. Another significant advantage of decision tree models is that this method doesn't rely on distributional assumptions as it is a non-parametric approach. Song at el. further explain that a significant advantage of using decision tree methods is that they enable us to easily interpret and understand the relationship between the target variable and input variables as the decision tree models work by reiteratively dividing the input variables into smaller and smaller groups.

Since our dataset is quite imbalanced, it would be interesting to see the algorithms typically used to predict severe disease. In an article titled "*An Imbalanced-Data Processing Algorithm for the Prediction of Heart Attack in Stroke Patients*," published in IEEE Access, the authors aim to predict heart attack in stroke patients. The data set used highly imbalanced because stroke patients with a heart attack account for a smaller percentage of the total patients who suffer from stroke. To deal with the imbalanced data set, the researchers used the under-sampling-clustering-oversampling Algorithm (shortly, UCO algorithm) to predict the risk of a heart attack in stroke patients. The authors can obtain a more balanced data set using the UCO algorithm. Then, five different methods are combined with the UCO technique to predict the risk of a heart attack, which leads to the conclusion that the random forest classifier achieved the best predicting performance with an accuracy of 70.29% and a precision of 70.05%. These results conclude that using this approach, UCO, in combination with a random forest algorithm, will provide reliable results when predicting whether a stroke patient is at risk of a stroke.

A similar article, titled, "Prediction for cardiovascular diseases based on laboratory data: An analysis of the random forest model" published by Xi et al. the researchers aim to develop a model that can be used for cardiovascular diseases in the general population using the random forest algorithm. The authors considered factors such as age, body mass index (BMI), fasting blood glucose (FBG), diastolic blood pressure (DBP), triglyceride (TG), systolic blood pressure (SBP), total cholesterol (TC), waist circumference, and high-density lipoprotein-cholesterol (HDL-C) were considered to predict the risk of cardiovascular disease using the random forest algorithm. When using the random forest model, the authors used 335 training samples to develop the random forest model and a total of 163 testing samples were used to analyze the performance of the established model. The developed model has an accuracy of 72.89% when aiming to predict the occurrence of fatal or non-fatal cardiovascular events.

Data mining techniques have been used to identify risk factors for stroke, the best medical care for stroke patients, and the mortality rate of stroke patients. Mortality in patients with stroke was predicted using data mining techniques such as support vector machines (SVM), decision trees, and logistics Regression (LR) using a dataset containing 4149 records, and it was found that LR outperformed the other methods [Hadianfard,2022]. Using administrative data, a stroke severity index was developed using an appropriate prescription, laboratory, procedure, and service claims where the KNN model outperformed other models [Sung,2015]. Moreover, classification methods have been used to classify strokes based on symptoms and different factors using techniques such as SVM and ANN, where ANN outperformed other algorithms. [Govindarajan, 2020] Data mining techniques have been used to predict short-term versus short/intermediate-term post-stroke mortality considering risk factors such as frailty, socio-demographics, medication, and medical history [Easton, 2014].

SECTION 3 The Proposed Work And Results

SECTION 3.1 Stroke Dataset

The dataset used in this project is a dataset from Kaggle [Fedesoriano, 2020] that can be used in training classification models to predict stroke events. This dataset has 5110 observations with 11 clinical attributes and 1 class attribute as the label.

id	gender	age	hypertension	heart_disease	ever_married	work_type	Residence_type	avg_glucose_level	bmi	smoking_status	stroke
9046	Male	67	0	1	Yes	Private	Urban	228.69	36.6	formerly smoked	1

Figure 1: Sample Data

From the dataset, one of the attributes, ' id,' may be redundant since the tuple's index can already identify each tuple. Also, 'id' is irrelevant to building a classification model; thus, this attribute can be removed from the dataset. And the rest of the data remains the same since all other attributes are significant for predicting strokes.

SECTION 3.2 Naïve Bayes Classifier

SECTION 3.2.1 Data Preprocessing

The naïve bayes classifier implemented accepted only categorical data, so all variables were converted to objects. Age was discretized in intervals of 5, with infants under age 2 being its own bin. The bmi was discretized using intervals in the bmi chart, and labeled: ['underweight', 'healthy weight', 'overweight', 'obesity class 1', 'obesity class 2', 'obesity class 3']. As for the average glucose level, it was discretized into 8 bins according to CDC guide for healthy glucose levels [CDC].

MICE Imputation was then used to impute the missing values from the BMI. The train and test ratio were then set to 70/30.

SECTION 3.2.2 Constructing Complement Naïve Bayes Classifier Model

Naïve Bayes is a probabilistic classifier that assigns the most likely class given a series of features. This works on the assumption that each feature is an independent variable. Bayes principle is used for the classification by calculating the probability of the class given the features:

$$P(C_i|X = (x_1, x_2, .. x_{\{n-1\}}, x_n)) = \frac{P(X|C_i)\,P(C_i)}{P(X)}$$

The Naïve Bayes classifier predicts that the tuple $\boldsymbol{X}$ belongs to the class $\boldsymbol{Ci}$ having the highest posterior probability. The Naïve Bayes algorithm is as follows:

$$argmax\ P(y) = \prod P(c|x)^{f_i}$$

Where f_i is the frequency of the condition.

A frequency table is created to find the likelihood of each condition, and the probabilities of each result (stroke or no stroke) is calculated based on the condition. There is, however, an imbalance between the dataset that result in stroke and that don't, as such, the model tends to overfit and predict no stroke despite the patient conditions leaning towards a stroke. To counter this, a complement naïve bayes was used, where the inverse of the naïve bayes is used, and the minimum probability is selected.

$$argmin\ P(y) = \prod \frac{1}{P(c|x)^{f_i}}$$

SECTION 3.2.3 Evaluation

The confusion matrix for the Complement Naïve Bayes shows a large proportion of the data having a false label that has been correctly predicted, but there is still a significant proportion of True labels that were predicted as false despite using complement naïve bayes to counter the imbalance. As such, while the accuracy of the model is quite high at 0.925, its precision, recall and f-1 score are respectively 0.210, 0.237, 0.273. The runtime for training the model is 5 seconds for the entire dataset.

SECTION 3.3 ID3 Decision Tree Classifier

SECTION 3.3.1 Data Preprocessing

A total of 201 missing values were found in the BMI attribute. These missing values can be imputed using MICE imputation as it predicts the BMI based on other attributes. An imputation was needed as the tuples with missing values consisted of 4% of the total dataset; removing these tuples or keeping the missing values will affect the accuracy of the classification task.

In the smoking_status attribute, the missing values were largely present, as many patients have an unknown smoking status. Therefore, instead of imputing missing values in this attribute, these missing values will be a factor level of their own called 'Unknown.'

Changing the non-numeric attributes into factor type. Some attributes were recorded as string, including gender, ever_married, work_type, residence_type, and smoking status. Changing them into factor types will work better for building the classification model.

SECTION 3.3.2 Constructing Decision Tree Classifier Model

Decision trees consist of a root node, internal node(s), and leaf nodes. Each decision tree has one and only one root node, and each leaf node has a class label associated with it. The internal nodes and the root node contain test conditions for attributes of the data set.

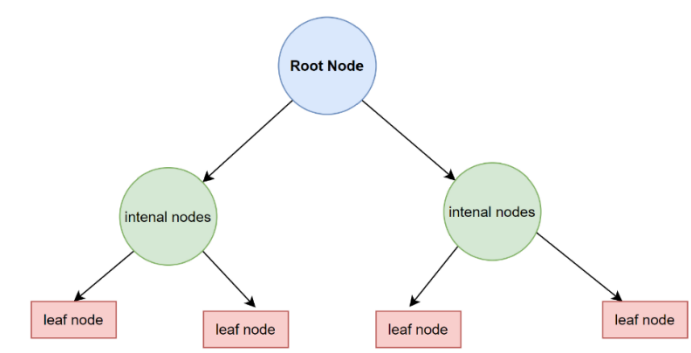

Figure 2: Decision Tree

When constructing a decision tree, a stopping condition for the splitting position must also be selected. Also, the measures needed to decide the best split for the records must also be chosen. In this case, our stopping criterion is maximum depth of the tree. The algorithm will only build a tree of the size specified with maximum depth of the tree. After fine tuning, the maximum depth of tree is 5. When the depth is more than 5, it does not add anymore performance to the model, but adds more run-

time. In this paper, an ID3 decision tree is constructed where the entropy and information gain are used to split the branches of the tree.

When constructing decision trees, different measures can be used to select the best possible split. The two popular measures for deciding the best split for decision trees are entropy and information gain. Both measures reflect the degree of impurity of the child's nodes.

A higher degree of impurity implies a higher degree of skewness of the class distribution. If a class is normally distributed, then that class will have the highest impurity. A lower value for both these measures is preferred over higher values.

Entropy can be represented by the following equation:

$$Entropy(t) = -\sum_{i=0}^{c-1} p(i|t)\log_2 p(i|t)$$

where c is the number of classes.

When using any impurity measure, it would be helpful to determine the performance of a test condition which can be done by comparing the impurity before and after splitting the node, i.e., the impurity of parent and child nodes. It is desirable to have a more significant difference in impurity between the parent and child nodes. This measure, also known as the gain, reflects a particular split's goodness. The following equation can represent this:

$$\Delta = I(parent) - \sum_{j=1}^{k} \frac{N(v_j)}{N} I(v_j)$$

Since the entropy of the parent node remains the same for all test conditions, then, to maximize the gain, the weighted average impurity measures of the child nodes will be minimized. This is also known as Information GainΔ_{info} (the difference in entropy between parents and child nodes) and provides information about how good a test condition is.

SECTION 3.3.3 Evaluation

The accuracy score of decision tree classifier is at 0.92 which may seem to be a good value for the model. But it falls short with other metrics. The precision, f-1, and recall scores of the model are 0.14, 0.14, and 0.14 respectively. These scores are when the model is trained with 5% of the dataset, and improving the dataset only resulted in overfitting. The runtime for training decision tree model is about 1 second for 5% training data.

SECTION 3.4 Random Forest Classifier

SECTION 3.4.1 Data Preprocessing

The random forest classifier uses the same pre-processed data that was done for Decision Tree classifier, since our Random Forest classifier uses ID3 trees in the previous section as their trees in the forest.

SECTION 3.4.2 Constructing Random Forest Classifier Model

Random Forest is an ensemble method that basically combines many decision tree classifiers. The idea behind Random Forest is to have the trees in the forest looking at the data from 'different angle'. This means that the decision trees in Random Forest will be fed and trained with data that are randomly sampled from the whole dataset. In our Random Forest classifier, the stopping criterion is the number of trees in the forest. In our random forest, the number of trees was chosen to be 25 trees in the forest after fine tuning for the best result. Thus, the algorithm will construct the forest until it reaches the specified number of trees. The algorithm uses

bootstrap sampling to sample the data for each tree in the forest.

SECTION 3.4.3 Evaluation

The accuracy score of random forest classifier is 0.92 which is the best among the three models. But it also falls short with other metrics. The precision, f-1, and recall scores of the model are 0.19, 0.18, and 0.17 respectively. These scores are when the model is trained with 2% of the dataset, and improving the dataset only resulted in overfitting. The runtime for training a random forest model is about 7 seconds for 2% training data. But this runtime increases drastically when feeding more data to the model.

SECTION 3.5 Comparison of Classification Models

Comparison table for the three models used with the full data for Complement Naïve Bayes and 5% of the dataset for Decision Tree and Random Forest:

	Complement Naïve Bayes	Decision Tree (5%)	Random Forest (5%)
accuracy	0.9254	0.9176	0.9440
precision	0.2103	0.1416	0.08511
recall	0.2728	0.1416	0.0171
f1-score	0.2375	0.1416	0.0286
runtime	5 sec	1s	30 sec

Decision Trees perform the worst both in terms of metrics and runtime, as it needs 1 second to run the model with only 5% of the data, compared to the complement naïve bayes model that uses the whole dataset. While random forest seems to outperform both complement naïve bayes and decision trees in accuracy and precision, its recall is extremely low, and it has a runtime that is almost 6 times larger than complement naïve bayes despite using only a fraction of the data. When all the data is used, the runtime increases to 4970 seconds (82 minutes), while the precision, recall and f1-score is reduced to 0 due to the large imbalance of the predictors and the Random Forest algorithm's overfitting. As such, complement naïve bayes will be used in the stroke application for patients to predict their risk of stroke.

SECTION 3.6 Stroke Predictor Application

SECTION 3.6.1 Application Construction

The Stroke Predictor Application interface was constructed using python package called 'tkinter' (TK interface) that allows us to create a GUI that can be seen in Figure 4.

SECTION 3.6.2 System Architecture

The user interface was created using tkinter python standard interface to take in the input data from the users. Once the user inputs the data to the application, the program will process the input data and transform it into data types that match the data types required by Naïve Bayes classifier constructed in Section 3.2. The transformed data is then put into the classifier and the classifier then predicts the class (1:have or 0:have no stroke) of the patient. Once the class is predicted, the resulting class is processed in the program to write an output and then display an output message to the user.

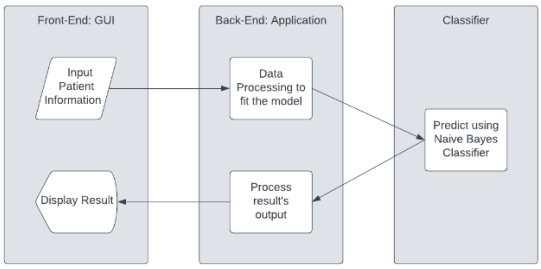

Figure 3: System Architecture for Stroke Predictor Application

SECTION 3.6.3 Graphical User Interface (GUI)

The GUI of Stroke Predictor application can be seen in the figure below.

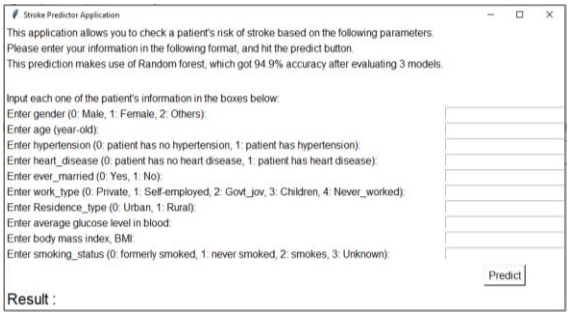

Figure 4: GUI of Stroke Predictor Program

SECTION 3.6.4 User Manual

Once the application is started, the user will be prompted with the GUI as can be seen in Figure 4 above. The user then needs to input the data associated with the patient in each of the boxes in the format specified in the parentheses of each line. An example input can be seen in Figure 5 below.

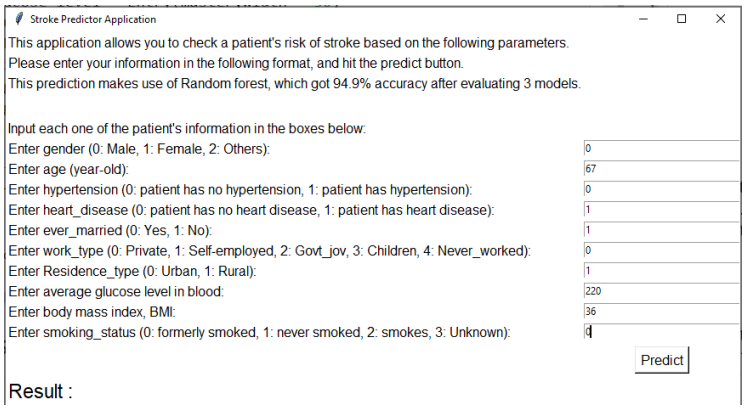

Figure 5: Example of user input

Once the user filled in all the boxes, the user can click on 'Predict' button located below the box to get the result. The result will show up once the 'Predict' button is hit as can be seen in the figure below.

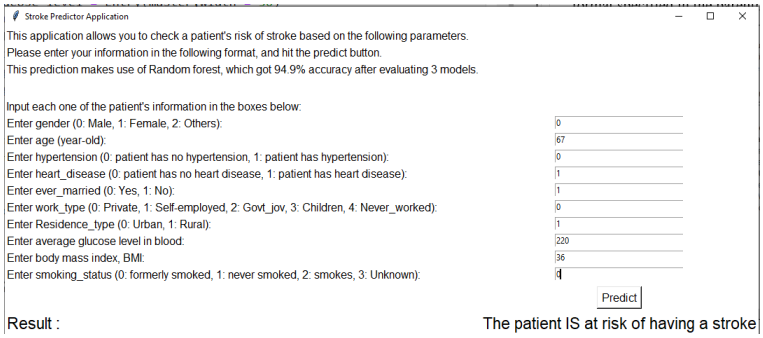

Figure 6: Result of the example input above

Another example of when the patient is not at risk of having a stroke can also be seen below.

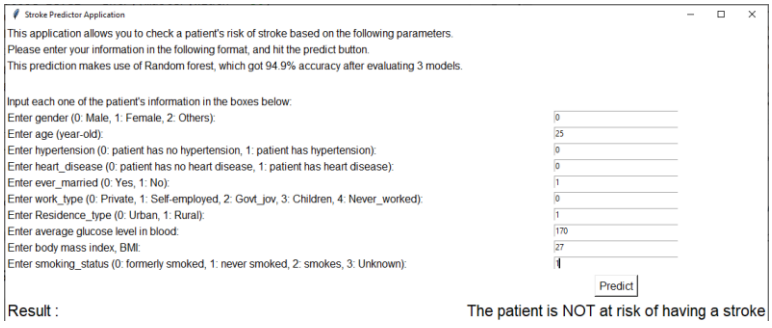

Figure 7: Result when the patient is not at risk of having a stroke

SECTION 4 Conclusion and future work

The Complement Naïve Bayes Classifier performed best overall to predict stroke in a patient as compared to Random Forest and Decision Trees. This is expected as the use of complements ensures that the class imbalance doesn't affect the predictions significantly. Although the random forest classifiers demonstrated that it could achieve a higher accuracy using a smaller proportion of the data, its precision increases as more data are used. However, it suffers from overfitting as most the predictions are very unbalanced, leading to null metrics when a significant portion of the data is used to train the model. The runtime of the model training can also reach up to more than 83 minutes if the whole dataset is used to train the model. Decision Trees faced similar problems, but with much worse metrics as random forest is an extension of this classification model.

The main issue in this paper is the class imbalance in the dataset, where a majority of patients in the data have no stroke and only 4.9% of patients have stroke recorded in the data. The classifiers in this paper suffer from overfitting with imbalanced data. Thus, one main future improvement that can be made is to consider the class imbalance which could greatly improve the performance of the classifiers. Random Forest classifier has the potential to accomplish great performance in predicting stroke if it can solve the overfitting. In other fields, it was suggested that coupling the under-sampling-clustering-oversampling Algorithm (UCO algorithm) with random forest can improve the precision in predicting with imbalanced data. Another suggestion to be made in the future is to improve the predictor to be able to predict the extent of the risk of having stroke. It will be beneficial if the users can predict if the patient is at risk of having a fatal or non-fatal event due to stroke.

References

CDC. Diabetes Tests. Centers for Disease Control and Prevention. 2022. Retrieved Dec 1, 2022, from https://www.cdc.gov/diabetes/basics/getting-tested.html

Easton, J., Stephens, C., & Angelova, M. Risk factors and prediction of very short term versus short/intermediate term post-stroke mortality: A data mining approach. Computers in biology and medicine. 2014. Vol.54, p.199-210. Retrieved September 14, 2022

Fedesoriano, *Stroke Prediction Dataset*. Kaggle. 2020. Retrieved September 14, 2022, from https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset

Govindarajan, P., Soundarapandian, R., Gandomi, A., Patan, R., Jayaraman, P., & Manikandan, R. (2019). *Classification of stroke disease using machine learning algorithms*. Neural computing & applications. 2019. Vol.32 (3), p.817-828. Retrieved September 14, 2022

Hadianfard, Z., Lotfnezhad Afshar, H., Nazarbaghi, S., Rahimi, B., & Timpka, T. *Predicting Mortality in Patients with Stroke Using Data Mining Techniques.* Acta Informatica Pragensia. 2022. Vol.11 (1), p.36-47. Retrieved September 14, 2022.

Kamel H., Abdulah D., Al-Tuwaijari M. *Cancer Classification Using Gaussian Naive Bayes Algorithm*. Journal of clinical epidemiology. 2015. Vol.68 (11), p.1292-1300. Retrieved September 14, 2022, from https://pubmed.ncbi.nlm.nih.gov/25700940/

Yan-Yan S., Yixue S., Ying L. M. *Decision tree methods: applications for classification and prediction.* Biostatistics in psychiatry. 2015, Vol.27 (2), p.130-135. Retrieved September 14, 2022,from https://na02.alma.exlibrisgroup.com/view/action/uresolver.do?operation=resolveService&package_service_id=5226724700002042&institutionId=2042&customerId=2040

Sung, S., Hu, Y., Chen, Y., Chen, C., Lin, H., Kao Yang, Y., & Hsieh, C. *Developing a stroke severity index based on administrative data was feasible using data mining techniques*. Journal of clinical epidemiology. 2015. Vol.68 (11), p.1292-1300. Retrieved September 14, 2022, from https://pubmed.ncbi.nlm.nih.gov/25700940/

Xi S., X., Tan Z., Wang Xi., Yang P., Su Y., Jiang Y., Qin S., Shang L. *Prediction for cardiovascular diseases based on laboratory data: An analysis of random forest model*. Journal of clinical laboratory analysis, 2020, Vol.34 (9), p.e2342 Retrieved December 1, 2022 from https://ieeexplore.ieee.org/document/9349502

Wang M., Yao X., Chen Y., *An Imbalanced-Data Processing Algorithm for the Prediction of Heart Attack in Stroke Patients* , *IEEE access*, 2021, Vol.9, p.25394- 25404, [Online]. Available: https://ieeexplore.ieee.org/document/9349502

WHO. *The top 10 causes of death*. World Health organization. 2022. Retrieved September 14, 2022, from https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death