AMERICAN INTERNATIONAL UNIVERSITY-BANGLADESH

Stroke Disease Prediction Using K-NN (K-Nearest Neighbor)

A Report By:

Chakraborty, Kowshik

ID: 18-36200-1

Subject: Data Mining

Section: C

Semester: Summer (2020-21)

Submitted To:

Dr. Md. Mahbub Chowdhury Mishu

Assistant Professor and Head in Charge (UG)

Dept. of Computer Science

Faculty of Science and Technology

American International University-Bangladesh

Abstract: Stroke disease is the most common and life-taking disease in the world now. That is why prediction of stroke has become one of the major concern today. There are different kinds of medical sectors trying to implement system that can predict stroke correctly. A solution is needed to build the system and data mining can be useful in that case because data mining helps to collect large data and stroke can be predicted easily using those data.

In this report, stroke disease is predicted using three data mining techniques K-NN (K- Nearest Neighbor), Naive Bayes and Decision Tree. Among these three classification techniques, K-NN is chosen and some information about this classification technique, reason of choosing this technique and the stroke dataset, prediction accuracy are also narrated here.

Introduction: Stroke is the most dangerous disease and cause of many people's death around the world. 5.5 million people are dying and 49 million people are being disabled every single year because of this disease[1]. There are several kinds of reason for which stroke happen. Smoking, hypertension, obesity, high cholesterol levels, diabetes, alcohol are some important causes of this disease. If a mechanism can be built for predicting stroke perfectly, then severe situations can be stopped and people's live can be rescued.

Data mining has become very popular and playing a key role in medical sectors because with the help of data mining, data can be turned into useful information. Stroke can be predicted precisely using those information. There are various kinds of techniques used (K-NN, Naive Bayes, Decision Tree, Logistic Regression etc.) for gathering useful information from dataset. An important thing to consider while choosing dataset that it should be proper. Dataset containing unnecessary data and bad features will give an inaccurate result and diminish the performance level of data mining technique. So, valid data and important features should be present in dataset.

<u>Dataset and Reason of Choosing This Dataset:</u> The stroke prediction dataset is collected from Kaggle containing 5110 instances with 12 attributes. The type of this dataset is .csv (Comma Separated Value). No unsatisfied data are present in this dataset and all the features (attributes) are good enough for predicting stroke disease.

No.	1: id Numeric	2: gender Nominal	3: age Numeric	4: hypertension Numeric	5: heart_disease Numeric	6: ever_married Nominal	7: work_type Nominal	8: Residence_type Nominal	9: avg_glucose_level Numeric	10: bmi Numeric	11: smoking_status Nominal	12: **stroke** Numeric
1	9046.0	Male	67.0	0.0	1.0	Yes	Private	Urban	228.69	36.6	formerly smoked	1.0
2	5167...	Female	61.0	0.0	0.0	Yes	Self-emplo...	Rural	202.21	28.8...	never smoked	1.0
3	3111...	Male	80.0	0.0	1.0	Yes	Private	Rural	105.92	32.5	never smoked	1.0
4	6018...	Female	49.0	0.0	0.0	Yes	Private	Urban	171.23	34.4	smokes	1.0
5	1665.0	Female	79.0	1.0	0.0	Yes	Self-emplo...	Rural	174.12	24.0	never smoked	1.0
6	5666...	Male	81.0	0.0	0.0	Yes	Private	Urban	186.21	29.0	formerly smoked	1.0
7	5388...	Male	74.0	1.0	1.0	Yes	Private	Rural	70.09	27.4	never smoked	1.0
8	1043...	Female	69.0	0.0	0.0	No	Private	Urban	94.39	22.8	never smoked	1.0
9	2741...	Female	59.0	0.0	0.0	Yes	Private	Rural	76.15	28.8...	Unknown	1.0
10	6049...	Female	78.0	0.0	0.0	Yes	Private	Urban	58.57	24.2	Unknown	1.0
11	1210...	Female	81.0	1.0	0.0	Yes	Private	Rural	80.43	29.7	never smoked	1.0
12	1209...	Female	61.0	0.0	1.0	Yes	Govt_job	Rural	120.46	36.8	smokes	1.0
13	1217...	Female	54.0	0.0	0.0	Yes	Private	Urban	104.51	27.3	smokes	1.0
14	8213.0	Male	78.0	0.0	1.0	Yes	Private	Urban	219.84	28.8...	Unknown	1.0
15	5317.0	Female	79.0	0.0	1.0	Yes	Private	Urban	214.09	28.2	never smoked	1.0
16	5820...	Female	50.0	1.0	0.0	Yes	Self-emplo...	Rural	167.41	30.9	never smoked	1.0
17	5611...	Male	64.0	0.0	1.0	Yes	Private	Urban	191.61	37.5	smokes	1.0
18	3412...	Male	75.0	1.0	0.0	Yes	Private	Urban	221.29	25.8	smokes	1.0
19	2745...	Female	60.0	0.0	0.0	No	Private	Urban	89.22	37.8	never smoked	1.0
20	2522...	Male	57.0	0.0	1.0	No	Govt_job	Urban	217.08	28.8...	Unknown	1.0
21	7063...	Female	71.0	0.0	0.0	Yes	Govt_job	Rural	193.94	22.4	smokes	1.0
22	1386...	Female	52.0	1.0	0.0	Yes	Self-emplo...	Urban	233.29	48.9	never smoked	1.0
23	6879...	Female	79.0	0.0	0.0	Yes	Self-emplo...	Urban	228.7	26.6	never smoked	1.0
24	6477...	Male	82.0	0.0	1.0	Yes	Private	Rural	208.3	32.5	Unknown	1.0
25	4219.0	Male	71.0	0.0	0.0	Yes	Private	Urban	102.87	27.2	formerly smoked	1.0
26	7082...	Male	80.0	0.0	0.0	Yes	Self-emplo...	Rural	104.12	23.5	never smoked	1.0
27	3804...	Female	65.0	0.0	0.0	Yes	Private	Rural	100.98	28.2	formerly smoked	1.0
28	6184...	Male	58.0	0.0	0.0	Yes	Private	Rural	189.84	28.8...	Unknown	1.0
29	5482...	Male	69.0	0.0	1.0	Yes	Self-emplo...	Urban	195.23	28.3	smokes	1.0
30	6916...	Male	59.0	0.0	0.0	Yes	Private	Rural	211.78	28.8...	formerly smoked	1.0
31	4371...	Male	57.0	1.0	0.0	Yes	Private	Urban	212.08	44.2	smokes	1.0
32	3387...	Male	42.0	0.0	0.0	Yes	Private	Rural	83.41	25.4	Unknown	1.0
33	3937...	Female	82.0	1.0	0.0	Yes	Self-emplo...	Urban	196.92	22.2	never smoked	1.0
34	5440...	Male	80.0	0.0	1.0	Yes	Self-emplo...	Urban	252.72	30.5	formerly smoked	1.0
35	1424...	Male	48.0	0.0	0.0	No	Govt_job	Urban	84.2	29.7	never smoked	1.0
36	712.0	Female	82.0	1.0	1.0	No	Private	Rural	84.03	26.5	formerly smoked	1.0
37	4726...	Male	74.0	0.0	0.0	Yes	Private	Rural	219.72	33.7	formerly smoked	1.0
38	2497...	Female	72.0	1.0	0.0	Yes	Private	Rural	74.63	23.1	formerly smoked	1.0
39	4730...	Male	58.0	0.0	0.0	No	Private	Rural	92.62	32.0	Unknown	1.0
40	6260...	Female	49.0	0.0	0.0	Yes	Private	Urban	60.91	29.9	never smoked	1.0

<u>Table 01: Example of Dataset.</u>

An overall description of dataset is given below,

Serial	Features	Feature Code	Description
1	ID	id	ID of patients
2	Gender	gender	Male, Female, Others
3	Age	age	Age in years
4	Hypertension	hypertension	0 = No hypertension 1 = Having hypertension
5	Heart disease	heart_disease	0 = No heart disease 1 = Having heart disease
6	Ever Married	ever_married	Marital Status (Yes, No)
7	Work Type	work_type	Types of works (Private, Self employed, Government job, Children, Never worked.)
8	Residence Type	Residence_type	Types of residences (Urban, Rural)
9	Average Glucose Level	avg_glucose_level	Level of Glucose (in average)
10	BMI	bmi	Body Mass Index
11	Smoking Status	smoking_status	Formerly smoked, Never smoked, Smokes, Unknown
12	Stroke	stroke	0 = No stroke 1 = Presence of stroke

Table 02: Description of Dataset.

As mentioned in introduction part, mortality rate is increasing each year because of stroke disease. So, prediction of stroke disease is very important to take steps against this disease before happening something serious. Moreover, the dataset is valid and quality of features are good and well understood. Besides, there are enough instances (5110) and attributes (12) for implementing stroke prediction system. That is why this kind of dataset is chosen.

Reason of Choosing K-NN (K-Nearest Neighbor): Three techniques used for predicting stroke disease and these techniques are:

K-NN (K-Nearest Neighbor): K-NN is a simple classification algorithm and it is widely used. To predict the classification of a new simple point, it uses database and data are separated into different classes in that database.

Naive Bayes: This classifier works based on probabilities of events and prior knowledge of the condition related to the event. Naive Bayes can be applied to get reverse probabilities if the conditional probability is known.

Decision Tree: Decision Tree uses supervised learning and this technique is used for classification and regression. It learns decision rule from the dataset and then generates a model that can predict the value of a target variable. CART and ID3 (Iterative Dichotomiser 3) are the two types of Decision Tree technique where CART uses gini index and ID 3 uses information gain.

Accuracy rate of these three techniques are given below:

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        4857               95.0489 %
Incorrectly Classified Instances       253                4.9511 %
Kappa statistic                          0.0653
Mean absolute error                      0.0832
Root mean squared error                  0.2246
Relative absolute error                 89.5808 %
Root relative squared error            104.3397 %
Total Number of Instances             5110

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.997    0.960    0.953      0.997    0.975      0.117    0.617     0.962     0
                 0.040    0.003    0.417      0.040    0.073      0.117    0.617     0.090     1
Weighted Avg.    0.950    0.913    0.927      0.950    0.931      0.117    0.617     0.920

=== Confusion Matrix ===

    a    b   <-- classified as
 4847   14 |    a = 0
  239   10 |    b = 1
```

Figure 01: K-NN Accuracy Test.

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        4532               88.6888 %
Incorrectly Classified Instances       578               11.3112 %
Kappa statistic                          0.1693
Mean absolute error                      0.1329
Root mean squared error                  0.2881
Relative absolute error                143.0946 %
Root relative squared error            133.8352 %
Total Number of Instances             5110

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.915    0.667    0.964      0.915    0.939      0.181    0.820     0.988     0
                 0.333    0.085    0.168      0.333    0.223      0.181    0.820     0.168     1
Weighted Avg.    0.887    0.638    0.925      0.887    0.904      0.181    0.820     0.948

=== Confusion Matrix ===

    a    b   <-- classified as
 4449  412 |    a = 0
  166   83 |    b = 1
```

Figure 02: Naive Bayes Accuracy Test.

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        4837               94.6575 %
Incorrectly Classified Instances       273                5.3425 %
Kappa statistic                          0.0176
Mean absolute error                      0.0905
Root mean squared error                  0.2213
Relative absolute error                 97.3832 %
Root relative squared error            102.8083 %
Total Number of Instances             5110

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.994    0.984    0.952      0.994    0.973      0.028    0.668     0.966     0
                 0.016    0.006    0.125      0.016    0.028      0.028    0.668     0.097     1
Weighted Avg.    0.947    0.936    0.911      0.947    0.927      0.028    0.668     0.924

=== Confusion Matrix ===

    a    b   <-- classified as
 4833   28 |    a = 0
  245    4 |    b = 1
```

Figure 03: Decision Tree Accuracy Test.

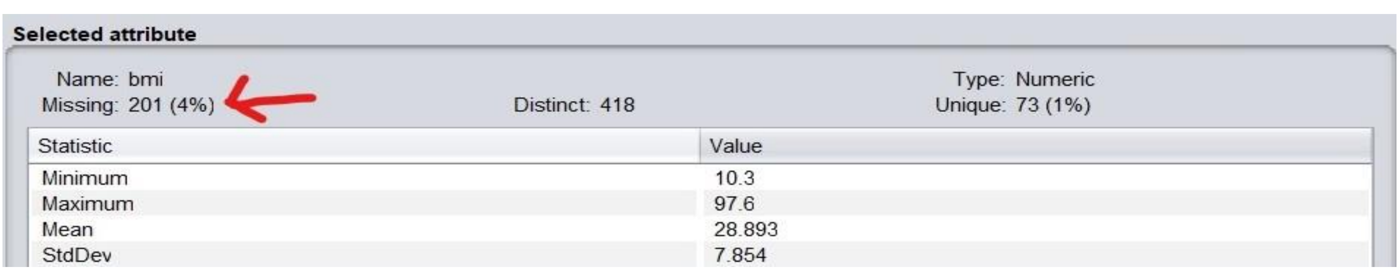
From the above three figures, it is clearly seen that K-NN has the highest accuracy rate in predicting stroke disease (95.05%) than other two techniques Naive Bayes (88.69%) and Decision Tree (94.66%). That is why K-NN is chosen for stroke prediction. Moreover, there are other reasons of choosing K-NN and these are:

1. K-NN is known as Lazy Learner. That means it does not need learning in training period and for this reason K-NN works much faster than other classification techniques.
2. As it does not require training before prediction, new data can be included and the accuracy of this algorithm will not be affected.
3. It is very easy to construct K-NN because it has only two parameters the value of K and the distance function (Euclidean, Manhattan etc.)
4. By using K-NN, it is very simple to work on multi-class problem without additional effort. On the contrary, it is a bit difficult to work on multiple-class using other techniques.
5. K-NN can be used in classification as well as regression problems and that is probably one of the reasons that K-NN is popular and widely used technique.

<u>Methodology:</u> All the tasks are done using Weka Tool and before prediction, there were some missing values in the dataset which are replaced using Weka Tool and after that, normalization is performed on the whole dataset. In this way data is pre-processed.

Missing Values

In dataset, about 201 missing values were found in the attribute named ‘bmi’. Then all the missing values are replaced using Weka Tool.



<u>Figure 04: Presence of Missing Values in Dataset.</u>

Selected attribute

Name: bmi
Missing: 0 (0%)
Distinct: 419
Type: Numeric
Unique: 73 (1%)

Statistic	Value
Minimum	10.3
Maximum	97.6
Mean	28.893
StdDev	7.698

Figure 05: After Replacing Missing Values.

Data Normalization

All the attributes in the dataset are normalized using ‘Normalize’ option in Weka Tool. Normalization helps the data converting the values ranging from 0 to 1.

No.	1: id Numeric	2: gender Nominal	3: age Numeric	4: hypertension Numeric	5: heart_disease Numeric	6: ever_married Nominal	7: work_type Nominal	8: Residence_type Nominal	9: avg_glucose_level Numeric	10: bmi Numeric	11: smoking_status Nominal	12: **stroke** Nominal
1	0.12...	Male	0.81...	0.0	1.0	Yes	Private	Urban	0.801264887821992...	0.30...	formerly smoked	1
2	0.70...	Female	0.74...	0.0	0.0	Yes	Self-emplo...	Rural	0.679023174222140...	0.21...	never smoked	1
3	0.42...	Male	0.97...	0.0	1.0	Yes	Private	Rural	0.234512048748961...	0.25...	never smoked	1
4	0.82...	Female	0.59...	0.0	0.0	Yes	Private	Urban	0.536007755516572...	0.27...	smokes	1
5	0.02...	Female	0.96...	1.0	0.0	Yes	Self-emplo...	Rural	0.549349090573354...	0.15...	never smoked	1
6	0.77...	Male	0.98...	0.0	0.0	Yes	Private	Urban	0.605161111624042...	0.21...	formerly smoked	1
7	0.73...	Male	0.90...	1.0	1.0	Yes	Private	Rural	0.069107192318345...	0.19...	never smoked	1
8	0.14...	Female	0.84...	0.0	0.0	No	Private	Urban	0.181285199889206...	0.14...	never smoked	1
9	0.37...	Female	0.71...	0.0	0.0	Yes	Private	Rural	0.097082448527375...	0.21...	Unknown	1
10	0.82...	Female	0.95...	0.0	0.0	Yes	Private	Urban	0.015926507247714...	0.15...	Unknown	1
11	0.16...	Female	0.98...	1.0	0.0	Yes	Private	Rural	0.116840550272366...	0.22...	never smoked	1
12	0.16...	Female	0.74...	0.0	1.0	Yes	Govt_job	Rural	0.301634198134982...	0.30...	smokes	1
13	0.16...	Female	0.65...	0.0	0.0	Yes	Private	Urban	0.228002954482503...	0.19...	smokes	1
14	0.11...	Male	0.95...	0.0	1.0	Yes	Private	Urban	0.760409934447419...	0.21...	Unknown	1
15	0.07...	Female	0.96...	0.0	1.0	Yes	Private	Urban	0.733865755701227...	0.20...	never smoked	1
16	0.79...	Female	0.60...	1.0	0.0	Yes	Self-emplo...	Rural	0.518373188071276...	0.23...	never smoked	1
17	0.76...	Male	0.78...	0.0	1.0	Yes	Private	Urban	0.630089557750900...	0.31...	smokes	1
18	0.46...	Male	0.91...	1.0	0.0	Yes	Private	Urban	0.767103683870372	0.17...	smokes	1
19	0.37...	Female	0.73...	0.0	0.0	No	Private	Urban	0.157418520912196...	0.31...	never smoked	1
20	0.34...	Male	0.69...	0.0	1.0	No	Govt_job	Urban	0.747668728649247...	0.21...	Unknown	1
21	0.96...	Female	0.86...	0.0	0.0	Yes	Govt_job	Rural	0.640845720616748...	0.13...	smokes	1
22	0.18...	Female	0.63...	1.0	0.0	Yes	Self-emplo...	Urban	0.822500230818945...	0.44...	never smoked	1
23	0.94...	Female	0.96...	0.0	0.0	Yes	Self-emplo...	Urban	0.801311051611116...	0.18...	never smoked	1
24	0.88...	Male	1.0	0.0	1.0	Yes	Private	Rural	0.707136921798541...	0.25...	Unknown	1
25	0.05...	Male	0.86...	0.0	0.0	Yes	Private	Urban	0.220432093066198...	0.19...	formerly smoked	1
26	0.97...	Male	0.97...	0.0	0.0	Yes	Self-emplo...	Rural	0.226202566706675...	0.15...	never smoked	1
27	0.52...	Female	0.79...	0.0	0.0	Yes	Private	Rural	0.211707136921798...	0.20...	formerly smoked	1
28	0.84...	Male	0.70...	0.0	0.0	Yes	Private	Rural	0.621918567075985...	0.21...	Unknown	1
29	0.75...	Male	0.84...	0.0	1.0	Yes	Self-emplo...	Urban	0.646800849413719...	0.20...	smokes	1
30	0.94...	Male	0.71...	0.0	0.0	Yes	Private	Rural	0.723201920413627...	0.21...	formerly smoked	1
31	0.59...	Male	0.69...	1.0	0.0	Yes	Private	Urban	0.724586834087341...	0.38...	smokes	1
32	0.46...	Male	0.51...	0.0	0.0	Yes	Private	Rural	0.130597359431262...	0.17...	Unknown	1
33	0.53...	Female	1.0	1.0	0.0	Yes	Self-emplo...	Urban	0.654602529775643...	0.13...	never smoked	1
34	0.74...	Male	0.97...	0.0	1.0	Yes	Self-emplo...	Urban	0.912196473086510...	0.23...	formerly smoked	1
35	0.19...	Male	0.58...	0.0	0.0	No	Govt_job	Urban	0.134244298772043...	0.22...	never smoked	1
36	0.00...	Female	1.0	1.0	1.0	No	Private	Rural	0.133459514356938...	0.18...	formerly smoked	1
37	0.64...	Male	0.90...	0.0	0.0	Yes	Private	Rural	0.759855968977933...	0.26...	formerly smoked	1
38	0.34...	Female	0.87...	1.0	0.0	Yes	Private	Rural	0.090065552580555...	0.14...	formerly smoked	1
39	0.64...	Male	0.70...	0.0	0.0	No	Private	Rural	0.173114209214292...	0.24...	Unknown	1
40	0.85...	Female	0.59...	0.0	0.0	Yes	Private	Urban	0.026728833902686...	0.22...	never smoked	1

Table 03: Example of Normalized Dataset.

After data pre-processing, prediction is done and from the prediction, confusion matrix is found. Confusion matrix provides the prediction result. By using this confusion matrix, accuracy, sensitivity and specificity can be calculated manually. Confusion matrix looks like this:

		Predicated Class	
Actual Class		NO	YES
	NO	TN	FP
	YES	FN	TP

Here,

1. P (Positive): Positive observation.
2. N (Negative): Negative observation.
3. TP (True Positive): Positive Observation and prediction is positive.
4. TN (True Negative): Negative Observation and prediction is negative.
5. FP (False Positive): Negative Observation but prediction is positive.
6. FN (False Negative): Positive Observation but prediction is negative.

Accuracy

Performance of a technique can be found by seeing accuracy rate. By seeing accuracy rate, whether a model is trained correctly or not can be found. The formula to calculate accuracy is:

$$\text{Accuracy} = (TP + TN) / (TP + FP + FN + TN)$$

Sensitivity

It gives the True Positive rate. Percentage of target value (stroke diseases in this case) can be identified correctly from the sensitivity. Formula to calculate sensitivity is:

$$\text{Sensitivity} = TP / (TP + FN)$$

Specificity

Opposite of sensitivity. It gives us True Negative Rate and from the specificity, percentage of target value which is normal can be identified correctly. That means in this case, patients without stroke disease can be identified correctly. Formula is:

Specificity: TN / (TN + FP)

Predicted Result: After doing prediction in Weka, K-NN gives the highest accuracy (95.05%) for predicting stroke disease. In Weka Tool, IBk is selected for predicting stroke disease because K-NN is known as IBk in Weka. 10-fold cross validation is used for the prediction so that a better accuracy can be achieved.

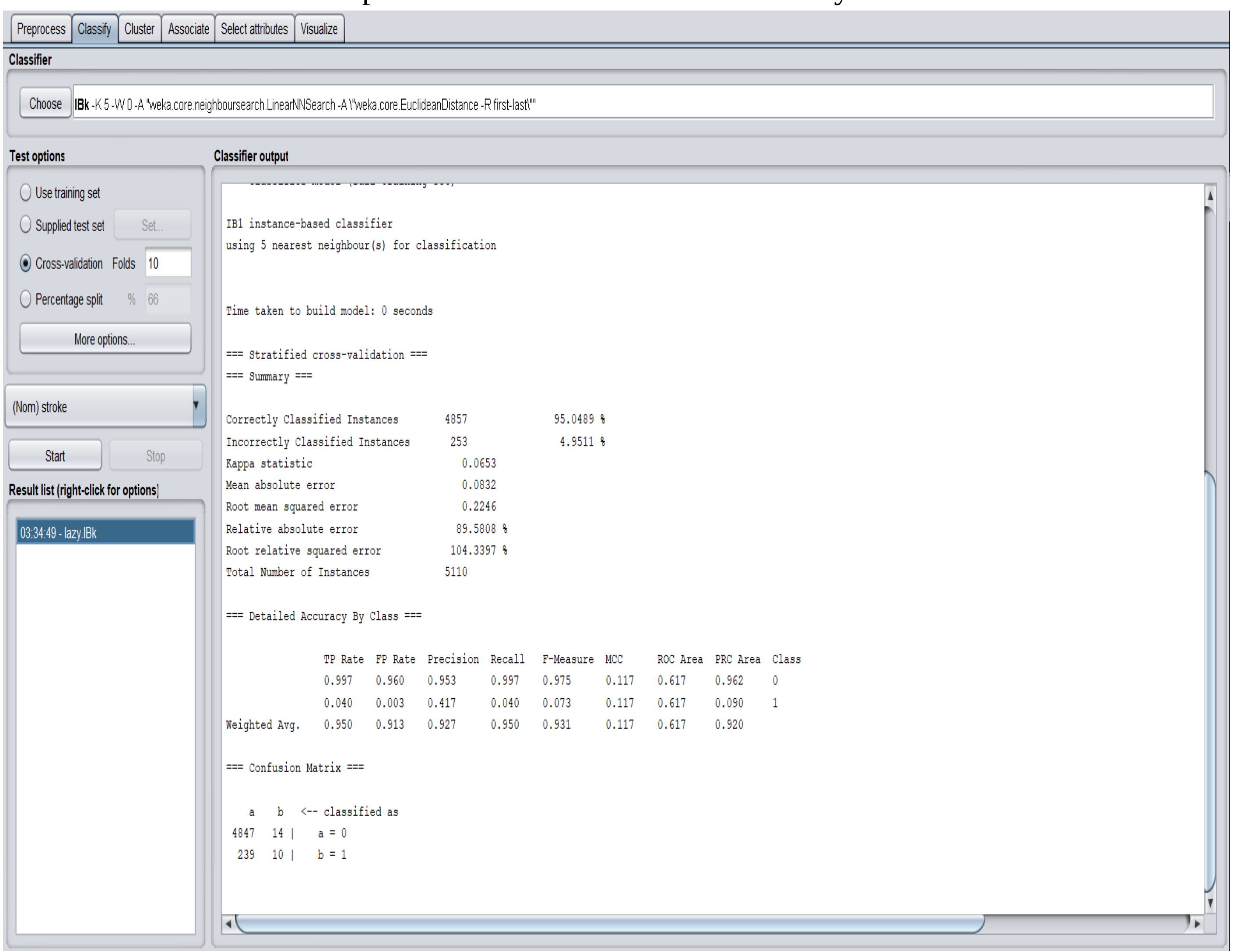


Figure 06: K-NN Technique for Predicting Stroke Disease.

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        4857               95.0489 %
Incorrectly Classified Instances       253                4.9511 %
Kappa statistic                          0.0653
Mean absolute error                      0.0832
Root mean squared error                  0.2246
Relative absolute error                 89.5808 %
Root relative squared error            104.3397 %
Total Number of Instances             5110
```

Figure 07: Stratified Cross-Validation.

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.997	0.960	0.953	0.997	0.975	0.117	0.617	0.962	0
	0.040	0.003	0.417	0.040	0.073	0.117	0.617	0.090	1
Weighted Avg.	0.950	0.913	0.927	0.950	0.931	0.117	0.617	0.920	

Figure 08: Detailed Accuracy.

```
=== Confusion Matrix ===

    a    b   <-- classified as
 4847   14 |    a = 0
  239   10 |    b = 1
```

Figure 09: Confusion Matrix.

From the above confusion matrix,

TP = 10, TN = 4847, FN = 239, FP = 14, a = No stroke diseases, b = Presence of stroke disease.

Accuracy: (TP + TN) / Total = (10 + 4847) / 5510 = 0.950489 = 95.05 %

Sensitivity: TP / (TP + FN) = 10 / (10 + 239) = 0.952075 = 95.21 %

Specificity: TN / (TN + FP) = 4847 / (4847 + 14) = 0.997119 = 99.72 %

Conclusion: Stroke is now the second main cause of people's death after heart disease stated by WHO (World Health Organization). Not only that, it is also responsible for about 11% of total deaths[2]. For this reason, accurate prediction of stroke is needed and data mining technique K-NN can solve this issue because by using this algorithm, approximately 95.05% accuracy found and 95.21% case of stroke disease are correctly predicted and 99.72% are predicted accurately for those who do not have stroke. Some errors (4.95%) found in the prediction result and future improvement can be done by taking dataset containing more valid and good features so that it can predict stroke diseases more accurately.

References: 1. Cox AM, McKevitt C, Rudd AG, Wolfe CD (2006) Socioeconomic status and stroke. Lancet Neurol5(2):181–188.

2. https://www.kaggle.com/fedesoriano/stroke-prediction-dataset.