Predicting Stroke Risk Using Hybrid Deep Transfer Learning Models

Presented To:
Final year Capstone Project Committee
Department of Computer Science & Engineering

Internal Guide:
Mr. Ajay Kumar Badhan
Assistant Professor, CSE

Presented By:

Abburi Naveen Varma	12018007
Morrigadudhula Abhinay	12020232
Mattapalli Veerasai	12009148
Bavanari Anilkumar	12014545
Somishetty Sanjay Varma	12007391
Chinda Harsha Vardhan Raju	12016603

CONTENT

1. Abstract
2. Introduction
3. Objectives
4. Literature Survey
5. Existing System
6. Proposed System
7. Flow Diagram
8. Modules
9. Result
10. System Requirements
11. Conclusion
12. References

Abstract

Our study introduces a novel approach, combining deep neural networks with transfer learning, to predict stroke risk. Utilizing a healthcare dataset, we preprocess the data, encode categorical variables, and train Decision Tree and Random Forest classifiers. Results highlight the hybrid model's effectiveness in accurately predicting stroke risk, showcasing its potential to augment healthcare analytics and provide valuable insights for preventive interventions. This hybrid deep transfer learning framework offers a promising avenue for enhancing stroke risk prediction models, thereby contributing to improved patient care and health outcomes.

INTRODUCTION

Stroke is one of the most prevalent diseases which could lead to death or long-term disability among elderly people all over the world. In a recent report, around 795 000 people experience a new or recurrent stroke each year in the US; one stroke incident occurs in approximately every 40 seconds.

Among the patients who suffered strokes, one in five would die within one year. For the survivals, the cost of treatment and rehabilitation becomes an extremely high burden to their families and the health-care system.

From 2014 to 2015, the direct and indirect cost due to stroke incidents was about 45.5 billion US dollars.

Thus, accurate stroke prediction is highly desirable so that the cost can be reduced with early interventions to delay the onset of and to reduce the risks of stroke.

OBJECTIVES

The main objective of our project is

- To predict or to classify the stroke or non-stroke effectively.
- To implement the feature selection for selecting the best features from our dataset.
- To implement the different machine learning algorithm.
- To enhance the overall performance analysis.

LITERATURE SURVEY

Ref No.	Paper Title	Methodology	Merits	Demerits	Year
1.	**Stroke Risk Prediction With Hybrid Deep Transfer Learning Framework**	Combine the outputs of multiple deep learning models, possibly including both CNNs and RNNs, Assess the performance of the hybrid deep transfer learning framework using appropriate evaluation metrics such as accuracy, precision.	It has the enhanced prediction accuracy, Transfer learning allows the integration of knowledge from diverse domains (e.g., images, text) into stroke risk prediction and reduces the training time	This led to overfitting if the pre-trained models are not appropriately fine-tuned Transfer learning may inadvertently transfer biases present in the pre-trained models	2022
2.	**An Integrated Machine Learning Approach to Stroke Prediction**	We propose a novel automatic feature selection algorithm that selects robust features based on our proposed heuristic: conservative mean. Combined with Support Vector Machines (SVMs)	Machine learning algorithms are capable of identifying features highly related to stroke occurrence efficiently from the huge set of features	Prediction is poor	2020
3.	**Performance Analysis of Machine Learning**	Ten different classifiers have been trained for predicting the stroke.	The results of the base classifiers have been aggregated using the	The prediction is not accurate.	2020

LITERATURE SURVEY

Ref No.	Paper Title	Methodology	Merits	Demerits	Year
1.	**A machine learning-based approach for predicting the outbreak of cardiovascular diseases in patients on dialysis**	We tested different types of algorithm (both linear and non-linear), but the final choice was to use Support Vector Machine. We obtained the best performances using the non-linear SVC with RBF kernel algorithm, optimizing it with Grid Search.	The prediction is accurate.	Time consumption is high and has theoretical limits	2020
2.	**Chronic Heart Failure Detection from Heart Sounds Using a Stack of Machine-Learning Classifiers**	The method consists of filtering, segmentation, feature extraction and machine learning. The method was tested with a leave-one-subject-out evaluation technique on data from 122 subjects, gathered in the study	The decision tree resulted in an average accuracy of 93%, which is higher than the other algorithm	The prediction is not accurate and has Less prediction	2019
3.	**Stroke Prediction Using Machine Learning in a Distributed Environment**	There is a need to design an approach to predict whether a person will be affected by stroke or not. This paper analyse different machine learning	Training time is low.	Error rate is high.	2019

EXISTING SYSTEM

In existing, the system is proposed a novel Hybrid Deep Transfer Learning-based Stroke Risk Prediction (HDTL-SRP) framework which consists of three key components:

(1) Generative Instance Transfer (GIT) for making use of the external stroke data distribution among multiple hospitals while preserving the privacy,

(2) Network Weight Transfer (NWT) for making use of data from highly correlated diseases (i.e., hypertension or diabetes),

(3) Active Instance Transfer (AIT) for balancing the stroke data with the most informative generated instances.

It is found that the proposed HDTL-SRP framework outperforms the state-of-the-art SRP models in both synthetic and real-world scenarios.

DISADVANTAGES

- It doesn’t efficient for large volume of data’s
- Theoretical limits.
- The process is implemented without removing unwanted data.
- The prediction is not accurate.

PROPOSED SYSTEM

In this system, the stroke dataset was taken as input. The input data was taken from the dataset repository. Then, we have to implement the data preprocessing step.

In this step, we have to handle the missing values for avoid wrong prediction. Then, we have to use label encoding, to encode the label for input data. To encode the columns into numeric values.

After that, we can implement the feature selection such as chi square for selecting the best features from pre-processed data.

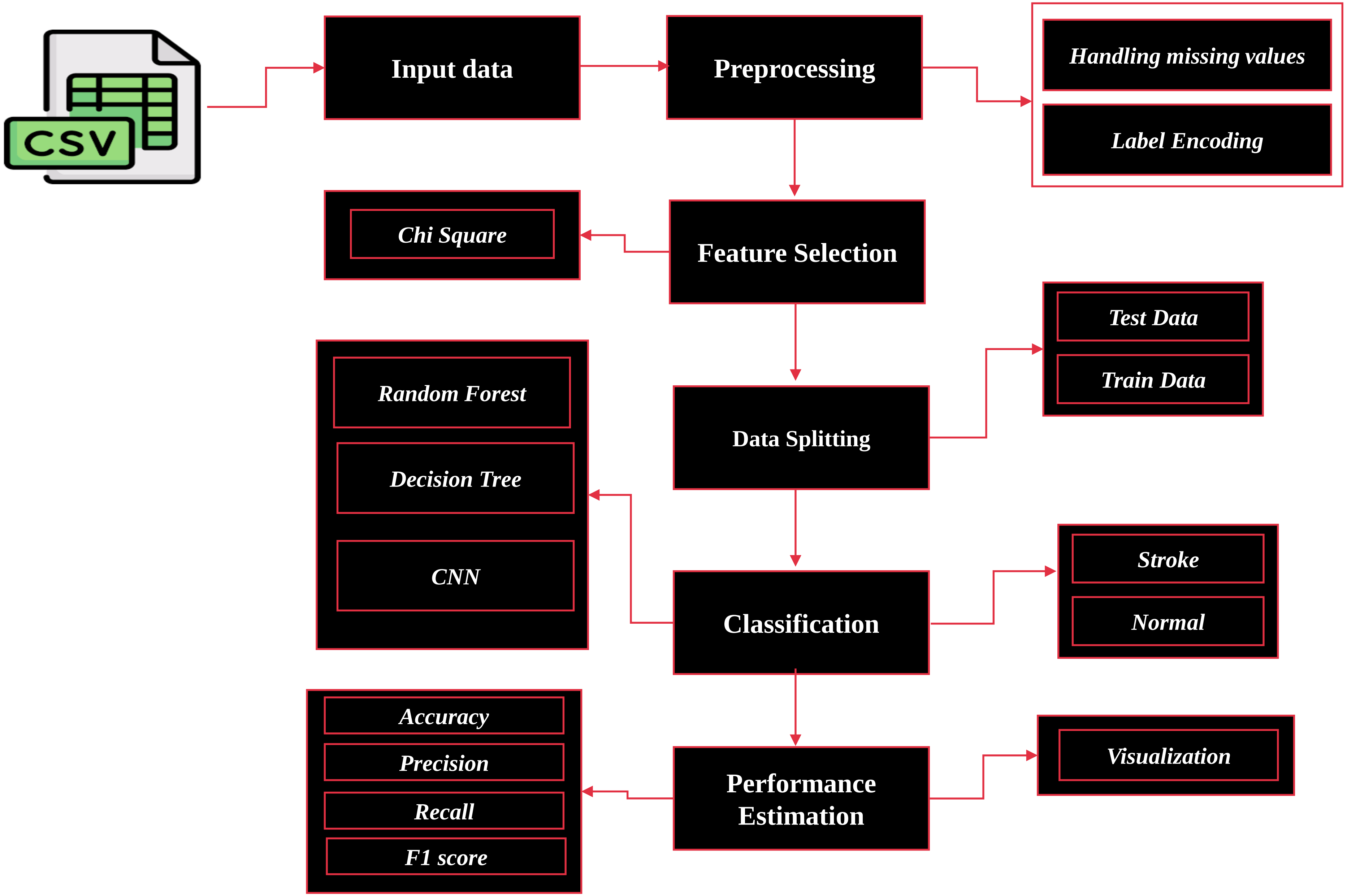
Next, we have to implement the data splitting. In this step, we have to split the data into test and train.

Then, we have to implement the deep and machine learning algorithms such as Convolutional Neural Network (CNN) , Decision Tree, Random Forest (RF). Finally, the experimental results shows that the performance metrics such as accuracy, precision, recall, f1-score and comparison graph.

ADVANTAGES

- It is efficient for large number of datasets.
- To increase the performance metrics results.
- Time consumption is low.
- The process is implemented with removing unwanted data.

Flow Diagram



Input data

Preprocessing

Handling missing values

Label Encoding

Feature Selection

Chi Square

Data Splitting

Test Data

Train Data

Classification

Random Forest

Decision Tree

CNN

Stroke

Normal

Performance Estimation

Accuracy

Precision

Recall

F1 score

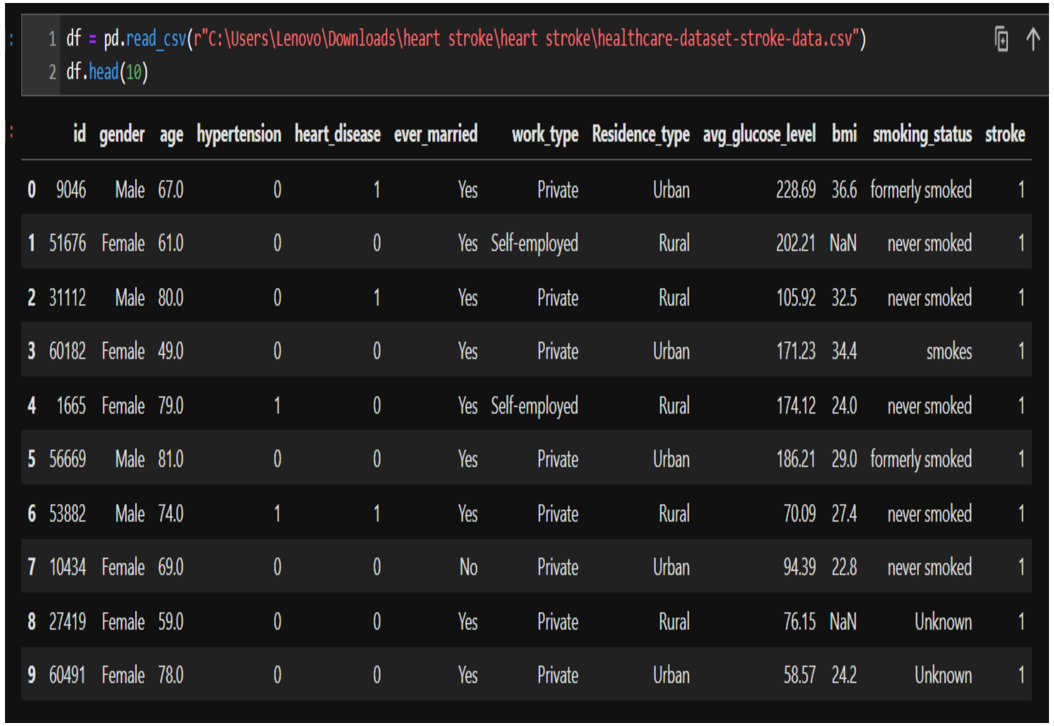
Visualization

MODULES

- Data selection
- Data preprocessing
- Feature Selection
- Data splitting
- Classification
- Prediction
- Performance analysis

DATA SELECTION

- The input data was collected from dataset repository like UCI, GitHub and Kaggle and so on.
- In our process, the stroke dataset is used.
- The dataset contains the information about the patient such as id, age, gender, hypertension, heart disease, ever married, work type, Residence type, average glucose level, Body Mass Index(BMI),smoking status and stroke.
- With the help of panda’s package, we can read our input dataset.
- The dataset is in the format “.csv”.



```
df = pd.read_csv(r"C:\Users\Lenovo\Downloads\heart stroke\heart stroke\healthcare-dataset-stroke-data.csv")
df.head(10)
```

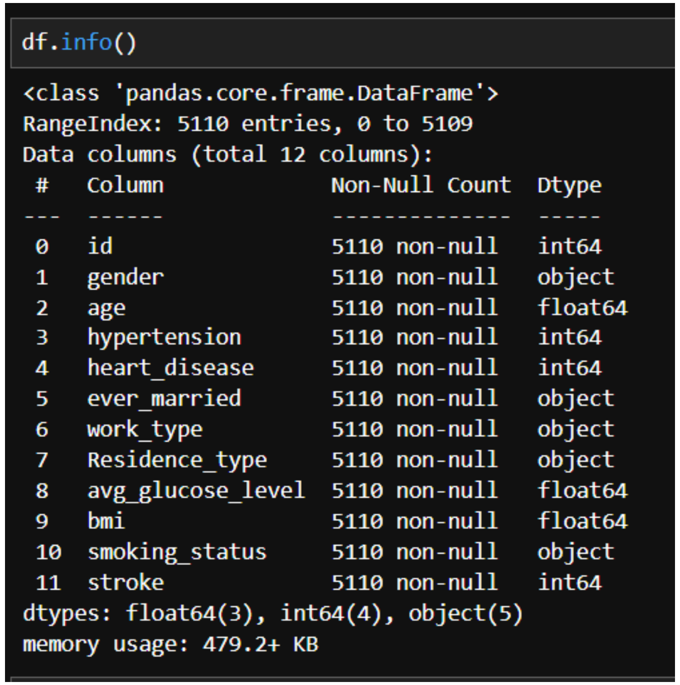
	id	gender	age	hypertension	heart_disease	ever_married	work_type	Residence_type	avg_glucose_level	bmi	smoking_status	stroke
0	9046	Male	67.0	0	1	Yes	Private	Urban	228.69	36.6	formerly smoked	1
1	51676	Female	61.0	0	0	Yes	Self-employed	Rural	202.21	NaN	never smoked	1
2	31112	Male	80.0	0	1	Yes	Private	Rural	105.92	32.5	never smoked	1
3	60182	Female	49.0	0	0	Yes	Private	Urban	171.23	34.4	smokes	1
4	1665	Female	79.0	1	0	Yes	Self-employed	Rural	174.12	24.0	never smoked	1
5	56669	Male	81.0	0	0	Yes	Private	Urban	186.21	29.0	formerly smoked	1
6	53882	Male	74.0	1	1	Yes	Private	Rural	70.09	27.4	never smoked	1
7	10434	Female	69.0	0	0	No	Private	Urban	94.39	22.8	never smoked	1
8	27419	Female	59.0	0	0	Yes	Private	Rural	76.15	NaN	Unknown	1
9	60491	Female	78.0	0	0	Yes	Private	Urban	58.57	24.2	Unknown	1

DATA PREPROCESSING

- Data pre-processing is the process of removing the unwanted data from the dataset.
- Pre-processing data transformation operations are used

  to transform the dataset into a structure suitable for machine learning.
- Missing data removal: In this process, the null values

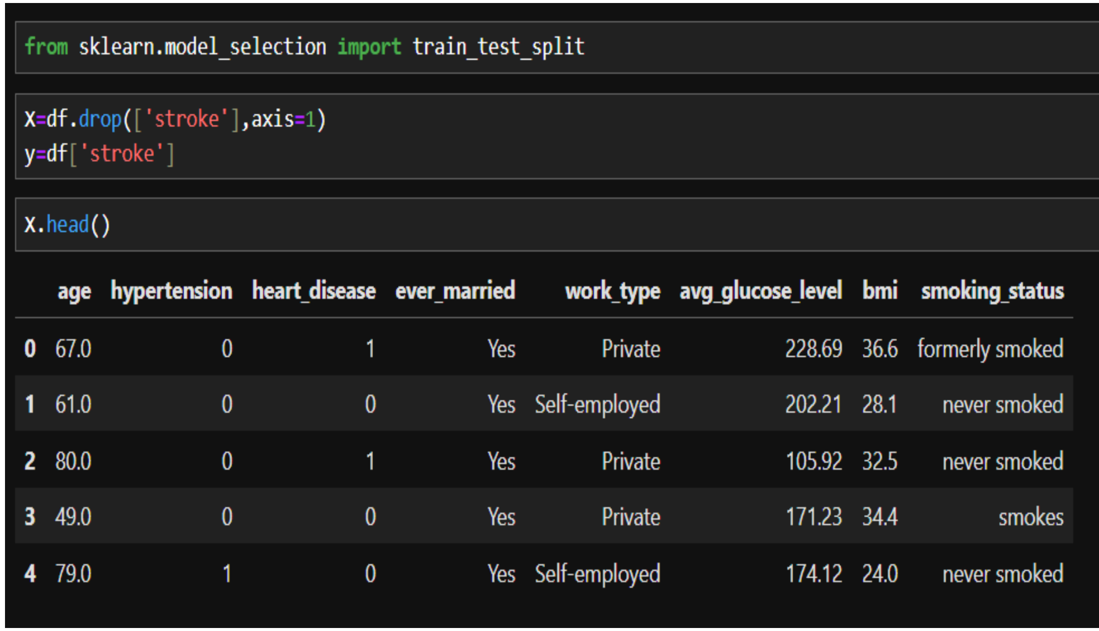
  such as missing values and Nan values are replaced by 0.
- Encoding Categorical data: That categorical data is defined as
- variables with a finite set of label values.
- That most machine learning algorithms require numerical input and output variables.



```
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5110 entries, 0 to 5109
Data columns (total 12 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   id                 5110 non-null   int64  
 1   gender             5110 non-null   object 
 2   age                5110 non-null   float64
 3   hypertension       5110 non-null   int64  
 4   heart_disease      5110 non-null   int64  
 5   ever_married       5110 non-null   object 
 6   work_type          5110 non-null   object 
 7   Residence_type     5110 non-null   object 
 8   avg_glucose_level  5110 non-null   float64
 9   bmi                5110 non-null   float64
 10  smoking_status     5110 non-null   object 
 11  stroke             5110 non-null   int64  
dtypes: float64(3), int64(4), object(5)
memory usage: 479.2+ KB
```

FEATURTE SELECTION

- In our process, we have to implement the feature selection for selecting the best features such as chi square and correlation.
- Then, we have to hybrid the two different feature selection techniques such as chi square and correlation.
- A chi-square test is used in statistics to test the independence of two events. Given the data of two variables, we can get observed count O and expected count E.
- Chi-Square measures how expected count E and observed
- count O deviates each other.



```
from sklearn.model_selection import train_test_split
```

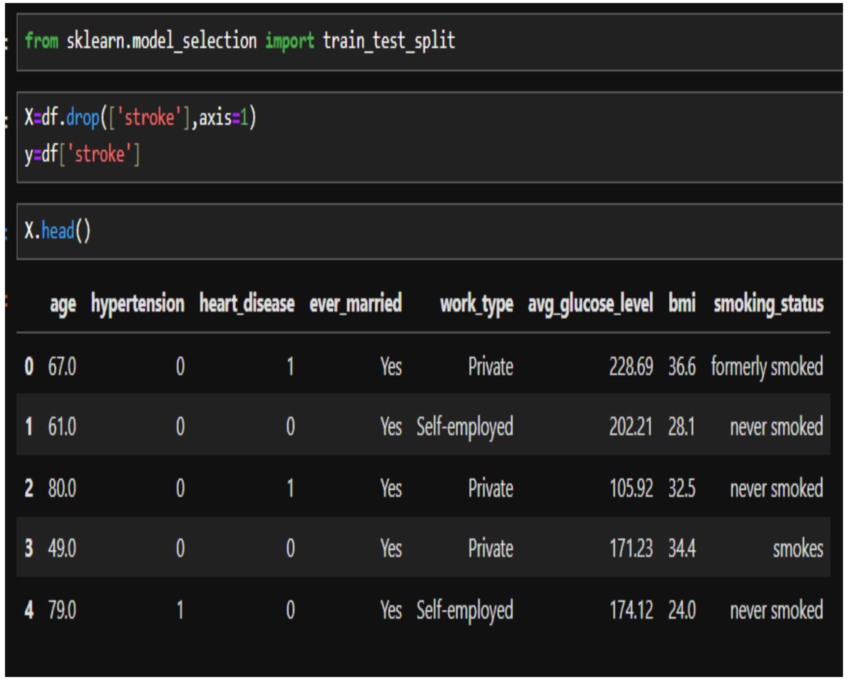
```
X=df.drop(['stroke'],axis=1)
y=df['stroke']
```

```
X.head()
```

	age	hypertension	heart_disease	ever_married	work_type	avg_glucose_level	bmi	smoking_status
0	67.0	0	1	Yes	Private	228.69	36.6	formerly smoked
1	61.0	0	0	Yes	Self-employed	202.21	28.1	never smoked
2	80.0	0	1	Yes	Private	105.92	32.5	never smoked
3	49.0	0	0	Yes	Private	171.23	34.4	smokes
4	79.0	1	0	Yes	Self-employed	174.12	24.0	never smoked

DATA SPLITTING

- During the machine learning process, data are needed so that learning can take place.
- In addition to the data required for training, test data are needed to evaluate the performance of the algorithm in order to see how well it works.
- In our process, we considered 80% of the disease dataset to be the training data and the remaining 20% to be the testing data.
- Data splitting is the act of partitioning available data into two portions, usually for cross-validator purposes.
- One Portion of the data is used to develop a predictive model and the other to evaluate the model's performance.



```
from sklearn.model_selection import train_test_split
```

```
X=df.drop(['stroke'],axis=1)
y=df['stroke']
```

```
X.head()
```

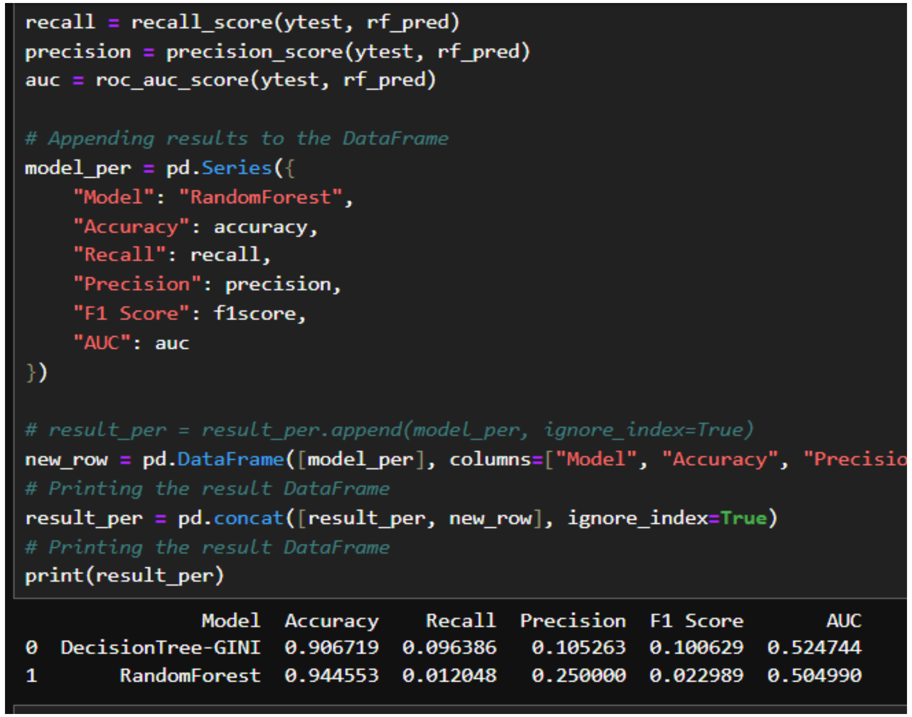
```
    age  hypertension  heart_disease  ever_married      work_type  avg_glucose_level   bmi  smoking_status
0  67.0             0              1           Yes        Private             228.69  36.6 formerly smoked
1  61.0             0              0           Yes  Self-employed             202.21  28.1    never smoked
2  80.0             0              1           Yes        Private             105.92  32.5    never smoked
3  49.0             0              0           Yes        Private             171.23  34.4          smokes
4  79.0             1              0           Yes  Self-employed             174.12  24.0    never smoked
```

CLASSIFICATION

In our process, we can implement the different classification algorithms such as random forest and multi-layer perceptron.

Random forest is a Supervised Machine Learning Algorithm that is used widely in Classification and Regression problems. It builds decision trees on different samples and takes their majority vote for classification and average in case of regression.

Decision Tree a popular machine learning algorithm used for both classification and regression tasks. It is a predictive modeling tool that learns to partition the data space into a series of rectangular regions and predicts the target variable's value based on the input features**.**

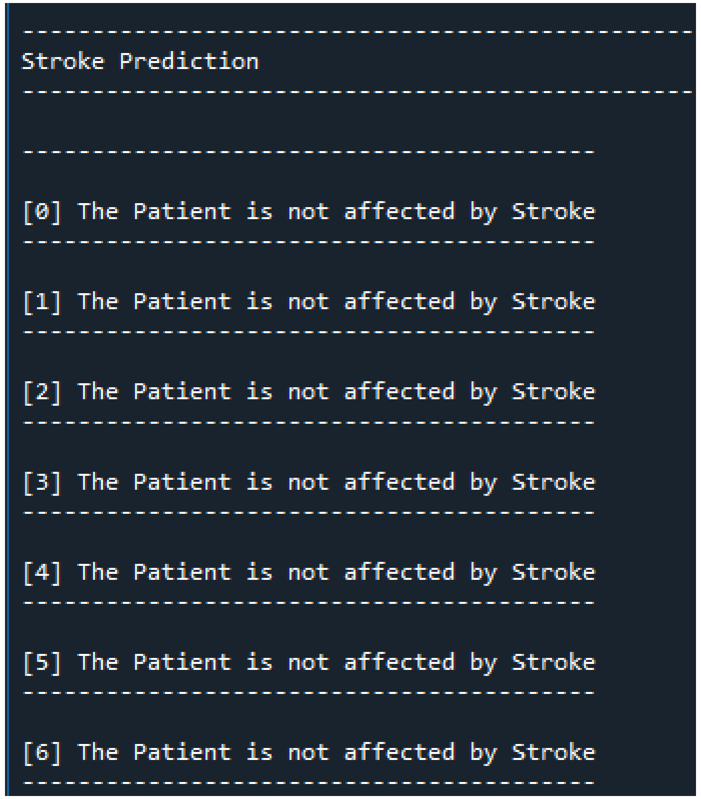


```
recall = recall_score(ytest, rf_pred)
precision = precision_score(ytest, rf_pred)
auc = roc_auc_score(ytest, rf_pred)

# Appending results to the DataFrame
model_per = pd.Series({
    "Model": "RandomForest",
    "Accuracy": accuracy,
    "Recall": recall,
    "Precision": precision,
    "F1 Score": f1score,
    "AUC": auc
})

# result_per = result_per.append(model_per, ignore_index=True)
new_row = pd.DataFrame([model_per], columns=["Model", "Accuracy", "Precisio
# Printing the result DataFrame
result_per = pd.concat([result_per, new_row], ignore_index=True)
# Printing the result DataFrame
print(result_per)
```

```
              Model  Accuracy    Recall  Precision  F1 Score       AUC
0  DecisionTree-GINI  0.906719  0.096386   0.105263  0.100629  0.524744
1      RandomForest  0.944553  0.012048   0.250000  0.022989  0.504990
```

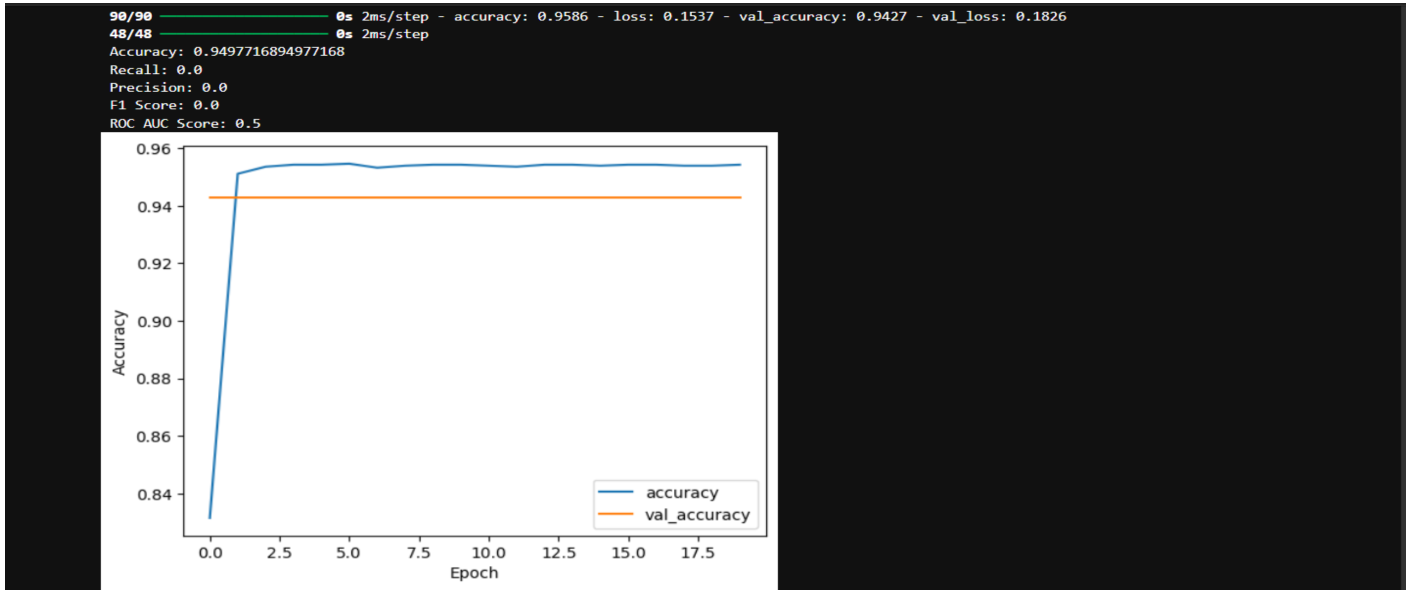
PREDICTION

- In this step, we have to predict the patient is affected by disease or not by using the classification algorithms.



```
------------------------------------------------
Stroke Prediction
------------------------------------------------

-----------------------------------------

[0] The Patient is not affected by Stroke
-----------------------------------------

[1] The Patient is not affected by Stroke
-----------------------------------------

[2] The Patient is not affected by Stroke
-----------------------------------------

[3] The Patient is not affected by Stroke
-----------------------------------------

[4] The Patient is not affected by Stroke
-----------------------------------------

[5] The Patient is not affected by Stroke
-----------------------------------------

[6] The Patient is not affected by Stroke
-----------------------------------------
```

RESULT



```
90/90 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.9586 - loss: 0.1537 - val_accuracy: 0.9427 - val_loss: 0.1826
48/48 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step
Accuracy: 0.9497716894977168
Recall: 0.0
Precision: 0.0
F1 Score: 0.0
ROC AUC Score: 0.5
```

SYSTEM REQUIREMENTS

SOFTWARE REQUIREMENTS:

- O/S : Windows 10.
- Language : Python
- Front End : JUPYTER NOTEBOOK

HARDWARE REQUIREMENTS:

- System : Pentium IV 2.4 GHz
- Hard Disk : 200 GB
- Ram : 4GB

CONCLUSION

We conclude that, the proposed system was implemented or developed the different classification algorithm for predicting or classifying the disease either the patient is affected by stroke or not effectively by using Multi-Layer Perceptron (MLP), Decision Tree and Random Forest (RF).

Then, the system was developed the feature selection technique for selecting the best features from our dataset.

The experimental results shows that some performance metrics such as accuracy, precision, recall and f1-score. Then, we are compared the two algorithms effectively.

REFERENCES

[1] Cui, L., Fan, Z., Yang, Y., Li, R., Wang, D., Feng, Y., … & Fan, Y. (2022). Deep learning in ischemic stroke imaging analysis: a comprehensive review. BioMed Research International, 2022, 1-15.

[2] Wei, Z., Li, M., & Fan, H. (2022). Hybrid deep learning model for the risk prediction of cognitive impairment in stroke patients.

[3] Chantamit-o-pas, P. and Goyal, M. (2017). Prediction of stroke using deep learning model. Neural Information Processing, 774-781.

[4] S. V, R. and R, G. (2023). Hybrid deep transfer learning framework for stroke risk prediction. The International Conference on Scientific Innovations in Science, Technology, and Management.

[5] Rao, B. N., Mohanty, S., Sen, K., Acharya, U. R., Cheong, K. H., & Sabut, S. (2022). Deep transfer learning for automatic prediction of hemorrhagic stroke on ct images. Computational and Mathematical Methods in Medicine, 2022, 1-10.

REFERENCES

[6] Cheon, S., Kim, J., & Lim, J. (2019). The use of deep learning to predict stroke patient mortality. International Journal of Environmental Research and Public Health, 16(11), 1876.

[7] AlArfaj, A. A., Mahmoud, H. A. H., & Hafez, A. M. (2022). A deep learning model for stroke patients' motor function prediction. Applied Bionics and Biomechanics, 2022, 1-9.

[8] Liu, Y., Yu, Y., Ouyang, J., Jiang, B., Yang, G., Ostmeier, S., … & Zaharchuk, G. (2023). Functional outcome prediction in acute ischemic stroke using a fused imaging and clinical deep learning model. Stroke, 54(9), 2316-2327.

[9] Su, S., Li, L., Wang, Y., & Li, Y. (2023). Stroke risk prediction by color Doppler ultrasound of carotid artery-baseddeep learning using Inception V3 and VGG-16. *Frontiers in Neurology, 14*.

[10] Lee, J.H., Kwon, J., Lee, M.S., Cho, Y., Oh, I., Park, J.J., & Jeon, K. (2022). Prediction of atrial fibrillation in patients with embolic stroke with undetermined source using electrocardiogram deep learning algorithm and clinical risk factors. *European Heart Journal*.

Thank You