Detec琀椀on of Stroke Disease using Machine Learning

Algorithms

Abstract:

A stroke is a medical condi琀椀on in which poor blood 昀氀ow to the brain results in cell death. It

is now a day a leading cause of death all over the world? Several risk factors believe to be

related to the cause of stroke has been found by inspec琀椀ng the a昀昀ected individuals. Using

these risk factors, a number of works have been carried out for predic琀椀ng and classifying

stroke diseases. Most of the models are based on data mining and machine learning

algorithms. In this work, we have used four machine learning algorithms to detect the type

of stroke that can possibly occur or occurred form a person’s physical state and medical

report data. We have collected a good number of entries from the hospitals and use them to

solve our problem. The classi昀椀ca琀椀on result shows that the result is sa琀椀sfactory and can be

used in real 琀椀me medical report. We believe that machine learning algorithms can help

be琀琀er understanding of diseases and can be a good healthcare companion. Index Terms—

Stroke, machine learning, WEKA, Naive Bayes, J48, k-NN, Random Forest

**Brain Strokes**

**Abstract**

A stroke is a medical condition in which poor blood flow to the brain results in the death of cells. Strokes are now of the leading causes of death particularly in America. There are several risk factors that may correlate with the chances of having a stroke – collecting these risk factors, such as medical history and external situations, allows for machine learning algorithms and implementation which may be able to detect a possible future stroke. The research trains a range of machine learning techniques such as KNN, linear regression, Decision Trees and Random Forest classification. It was found that Random Forest classification was the best performing algorithm for the task of predicting strokes with an accuracy of approximately 96%. The dataset used in this project was the Brain Strokes Dataset found in Kaggle.

**Introduction**

Stroke is a serious life-threatening medical condition that needs to be treated right away to prevent further complications. The creation of a machine learning (ML) model could help with the early diagnosis of stroke and subsequent reduction of its severe repercussions. This study examines the efficacy of various ML algorithms in correctly predicting stroke based on a variety of external factors (residency type, marriage situation) as well as medical history (BMI and glucose levels). In an ideal world, the implementation of a good ML algorithm will help patients obtain early treatment for strokes and rebuild their lives after the event.

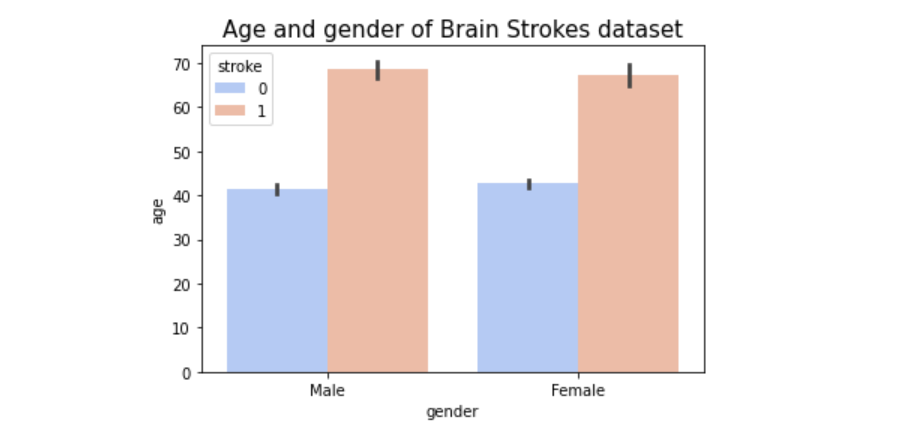
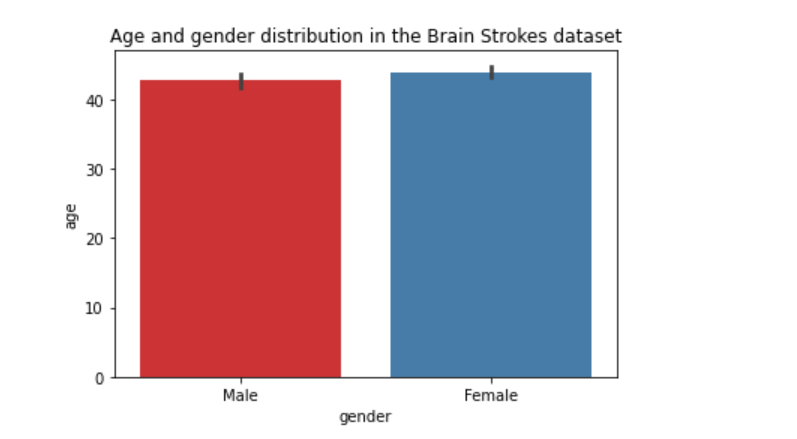
**Methodology**

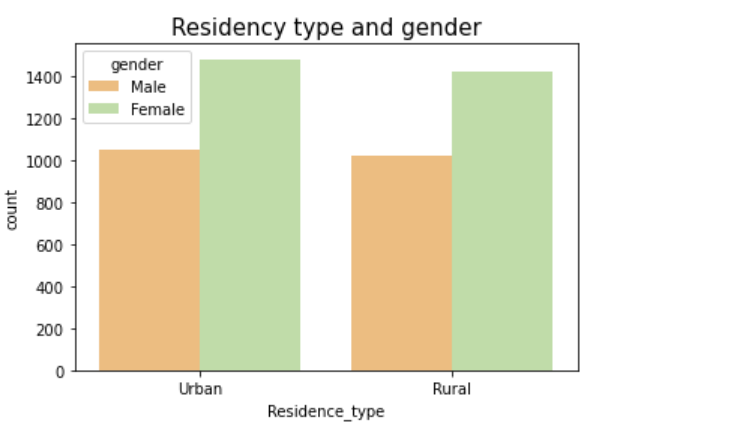
In order to access the data, first the dataset is imported into Jupyter notebook as a csv so that data cleaning can occur using the large variety of Python packages. After viewing the data, a quick summary of the data can be obtained using shape, describe, type, is null functions. Any missing data is replaced with Nan’s then after these nulls are replaced with 0. In order to train and test the data, all categorical inputs should be Label encoded so that the whole data frame has only numeric inputs. Label encoding converts the dataset’s string literals to integer values that the computer can comprehend. As the computer is frequently trained on numbers, the strings must be converted to integers. After the data has been encoded, the data should be scaled – this case uses a Minmax Scalar. If the data is not scaled and hence left unbalanced, the findings from the predictions will be inaccurate, and the forecast will be ineffective.

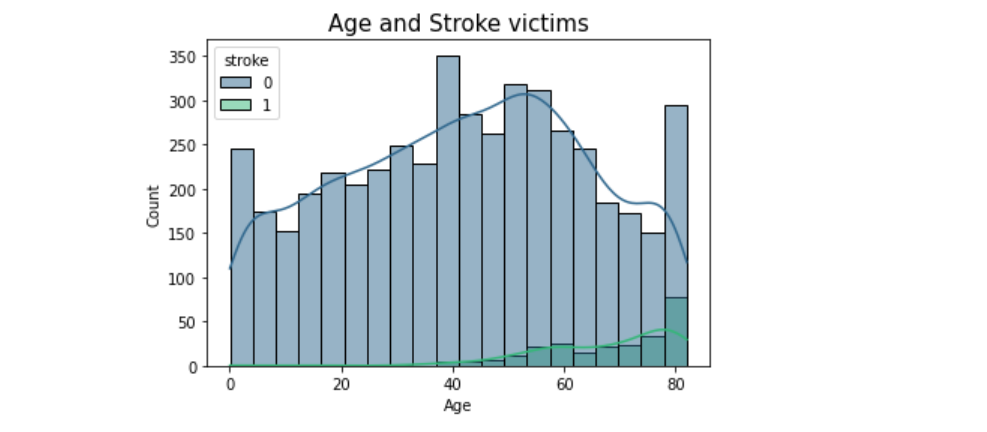
The scaled data can then be split into X and y and thereafter regression and classification models are implemented such as K-Nearest Neighbours, Decision Trees and Random Forest. Accuracy and f1 scores can then be generated for each ML type as well as a classification reports in order to determine the most robust technique. The target variable this report is focusing on is the ‘Stroke’ column.

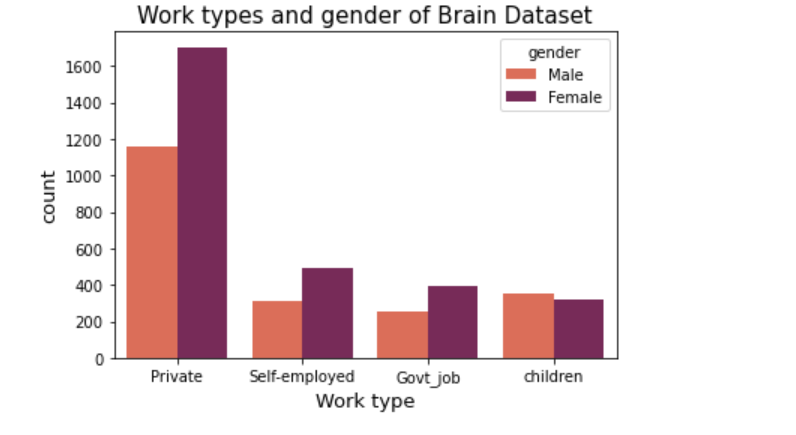
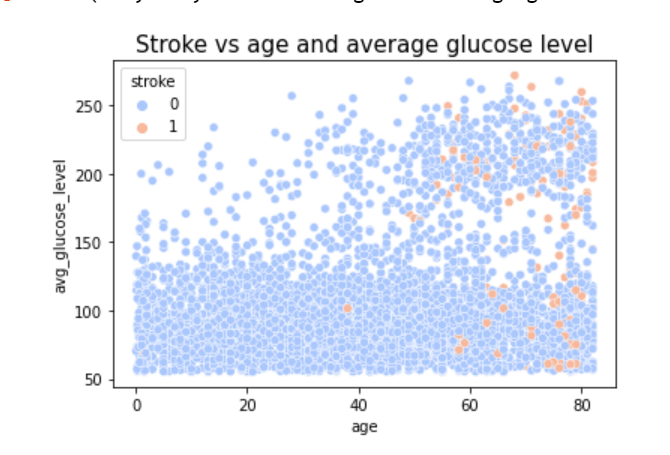
**Brain Stroke Data**

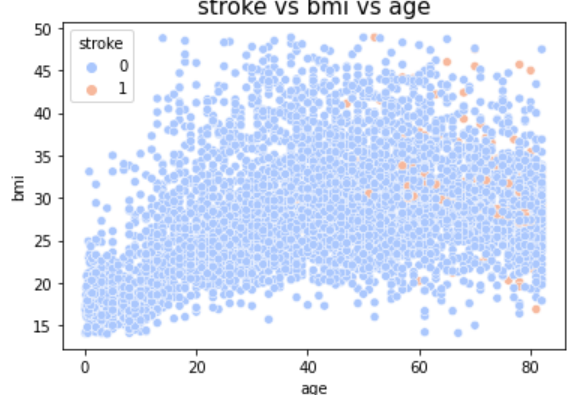
* The dataset used for stroke prediction is very has a total of 4981 rows, with 11 columns.
* The age column had a 104 unique row inputs.
* Minimum age in the dataset was a new-born (between 0 and 6 months).
* Maximum age in the dataset 82 years old.
* Columns include gender, age, hypertension, heart disease, work type, residency type, glucose level, BMI, smoking status and stroke.
* 248 had strokes out of 4981 people, roughly 5% of the users in the Data set.
* 58% Females in the dataset.
* Hypertension, heart disease and stroke have binary outputs with 1 if the patient had a stroke/hypertension/heart disease or 0 if not.

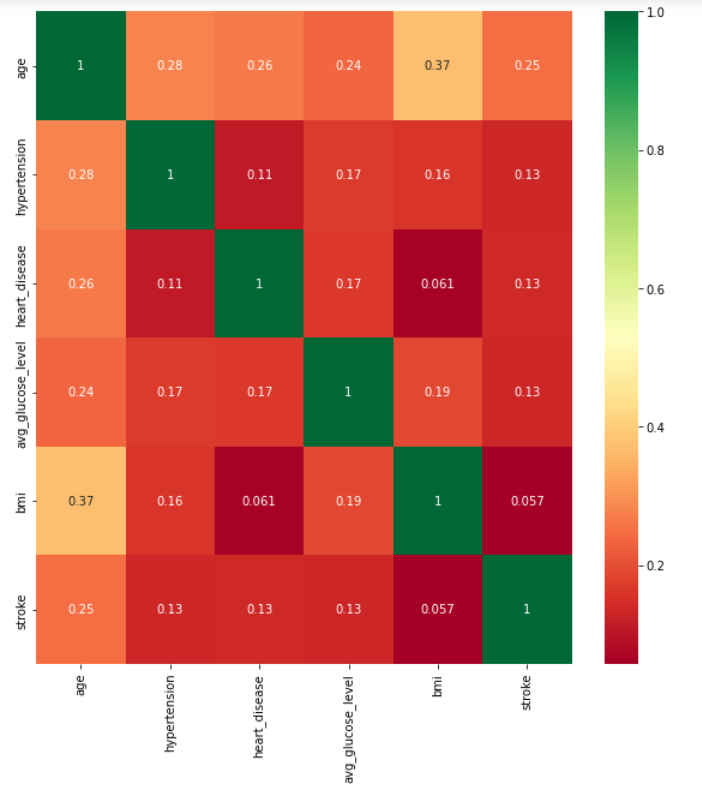
**Insights and Visualisations**











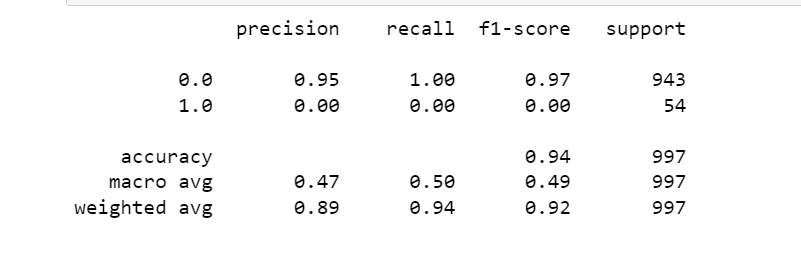
**Insights explained**

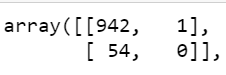
The bar plots indicate that an age>50 there is an increased chance of introducing a stroke, which is backed up by the scatter plot of average glucose level which also suggesting the higher the average glucose level, (150mg/dl or above), the likelihood of having a stroke increase. According to health websites, an average glucose level of 100 to 125 mg/dL indicates you have prediabetes, and 126 mg/dL or higher indicates you have diabetes. Additionally, with a greater BMI (>30) and an age above 50 – there is also an indication that the chances of suffering a stroke are greater. This is also concluded by the correlation heatmap above.

**Train-Test- Split**

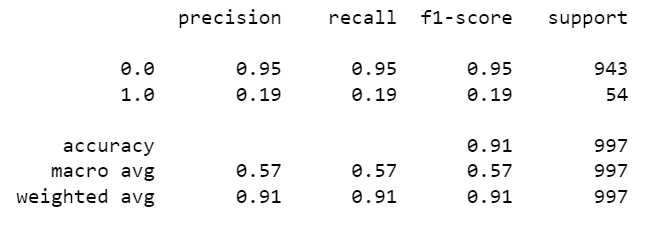
After the data frame has been encoded using label encoder as well as being scaled, the data is ready to be split into X and y. The train-test split procedure is used to estimate the performance of machine learning algorithms when they are used to make predictions on data not used to train the model. It is a quick and simple process to carry out, and the outcomes allows the user to compare the effectiveness of machine learning algorithms for the predictive modelling problem. The split used in this project is 80% training and 20% testing. Then defining and fitting the model on the training dataset to make predictions and evaluate the predictions using the classification accuracy performance metrics on different model testing methods.

**K-Nearest-Neighbours (KNN)**

The KNN algorithm analyses all the data points and then classifies the data, then new cases are then classified based on the already established categories. A KNN object is created with Euclidean distance as the metric (distance between two points) and p value of 2. The p indicated the number of unique inputs in the target variable column. 11 as the n\_neighbours input results in the following classification report, with an accuracy of 94%:

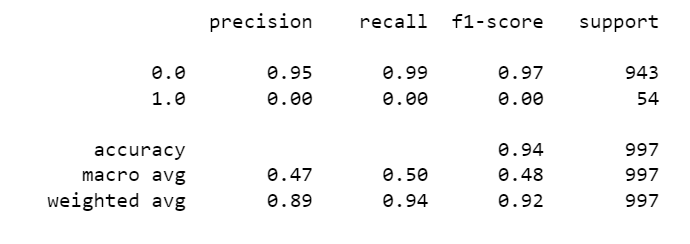
The confusion matrix is a technique for summarizing the performance of a classification algorithm. The KNN matrix was as follows:

**Decision Trees**

Decision tree algorithms are one of the most popular algorithms – it comes under the category of supervised learning and works for both continuous and categorical output variables. First a decision tree object is created and then it is fitted to the model. The classification report for the brain strokes decision tree method resulted in the following, with an accuracy of 91%.

**Random Forest Classifier**

The random forest classifier builds many decision trees on different samples and takes their majority vote for classification and average in case of regression. First make the random forest classifier object and then train and fit the classifier. The success of the random forest classifier can be evaluating by looking into the performance indexes from the classification report:

The random forest classifier returns the highest accuracy and f1 score at 94% and 97% respectively.

**MAE, MSE, RMSE**

The Mean Squared Error, mean absolute error, Root Mean Squared Error metrics are used to evaluate the performance of the model in regression analysis.

* Mean Squared Error represents the average of the squared difference between the original and predicted values in the data set. It measures the variance of the residuals.
* The Mean absolute error represents the average of the absolute difference between the actual and predicted values in the dataset. It measures the average of the residuals in the dataset.
* Root Mean Squared Error is the square root of Mean Squared error. It measures the standard deviation of residuals.

In the brain strokes data set the metrics values returned as follows:

MAE: 0.09864822884724292

MSE: 0.046808981429110344

RMSE: 0.21635383386737186

**Conclusion**

The study suggests that machine learning techniques make it possible to predict the likelihood of stroke in patients with the current features in the dataset. This could lead to significant reductions in the number of strokes induced as well as the level of severity due to early intervention. It is recommended to use Random Forest Classification as it resulted in the highest accuracy 94% as well as f1- score of 97%.