Final Report

Introduction

Strokes are a leading cause of death and serious disabilities for many Americans today. According to the CDC, about 795,000 people have a stroke each year in the United States. The data to be analyzed and used is from Kaggle.com by the user "fedesoriano" and is a stroke prediction dataset. In this dataset, there are 5110 observations with 12 attributes. The target variable will be a binary variable, stroke. Using machine learning methods of logistic regression and random forests and analyzing the performance of each with 10-fold cross validation, this dataset will be used to predict the likelihood of a patient having a stroke based on given attributes.

Exploratory Analysis

Before doing any exploratory data analysis, the dataset must be processed and cleaned. This dataset has a total of 12 variables, but for our analysis, the variables, "ever_married", "work_type", and "Residence_type", will not be factored in. That is because they are external factors, and we want to focus on the internal factors that would influence the likelihood of a patient having a stroke. In addition, when we did a logistic regression, the predicting variables provided the most significant ones, and we eliminated the variables that were not significant to combat overfitting the model. We eliminated the variables, "bmi", "gender" and "smoking status" because we found it to be too complicated for the model and led to high variance in the training data, which would ultimately lead to poor test results which would be further exaggerated by our unbalanced data set. Continuing, we cleaned up the four predictive variables by making them each a factor of level 2 and the average glucose level variable stays numerical.

By exploring the data, we can see that the proportion of strokes to no strokes is 95:1 which is highly unbalanced. Which can be seen in Figure 3. We balanced this data by oversampling the minority class by 50% which will be used in both mining methods later. Further exploring the data provides information about distribution and density. In Figure 1, putting counts or stroke against age shows that as age increases, the counts of strokes also increase. In Figure 4, the density of the BMI averages around 20-30, which is the normal weight BMI and the graph skews right. In Figure 2, the proportion between gender and having a stroke is 0.037% of females and 0.053% of males having a stroke.

Mining Methods

The mining methods used for this data set are logistic regression and random forests. We chose these two because they work with classification problems.

Firstly, before training the data with either methods, we used an over-sampling technique called ROSE ('Random Over-Sampling Examples') provided by the ROSE package that provides artificial datapoints that would smooth or balance out our data. By doing this technique, we can achieve a fifty-fifty ratio of strokes to no strokes, which would help with first, over fitting the model too well, and secondly, providing more predictions for having a stroke. Otherwise, the model would eventually majority predict for having no strokes, which is not optimal.

Our first method, logistic regression looks at the characteristics of our variables and help determine the probability of a patient having a stroke or no stroke. As previously mentioned, a logistic regression was done to eliminate any unnecessary variables that would unbalance the data more. In Figure 5, we can see that the variables, gender, bmi, ever married, and residence type are all insignificant and are therefore removed for the final training of the model. In our final model, shown in Figure 6, we can see that the final four variables are still significant even with oversampling of the data. To evaluate the performance of logistic regression model, we created a confusion matrix and calculated its accuracy, precision, and recall. In Figures 7 and 8 the graphs depict the confusion matrix for the training set and the testing set, respectively. Figure 9 is of the training and testing confusion matrix values. Accuracy measures the total number of predictions that the model gets correct. Accuracy decreased from 0.762 to 0.727 from training to testing dataset. Precision measures the how accurate and precise the model predicts positives. Our precision increased from 0.799 to 0.811. Recall is the percentage of actual positives the model had identified. The recall decreased extensively from 0.741 to 0.129.

The second method is random forests. We chose this method because it is great for classification and is great for identifying factors that are impacting our target variable. In Figure 11, the random forest model was trained using our ROSE oversampled dataset with 500 trees. Additionally, there is a confusion matrix shown with a class error of 0.2396 and 0.2306 respectively. After training the model, we can see the mean decrease Gini, which is the average gain of purity of each variable, shown in Figure 12. Age has the highest mean decrease Gini, followed by average glucose level, then heart disease, and hypertension. To evaluate the performance of the model, we create a confusion matrix (shown in Figures 13 and 14) from both the training and testing set. Again, we calculated (Figure 9) the accuracy, precision, and recall. The accuracy increased from 0.765 to 0.768. The precision decreased from 0.77 to 0.743 and the recall decreased dramatically from 0.761 to 0.141.

The models may have been overfitted and worked too well with the training data, that is why the calculations, especially the recall value have been decreased extensively.

When it comes to predicting stroke, it is important that the predictions are not deadly. Meaning, it is more ideal to obtain a false positive than a false negative as a false negative would mean the patient has a stroke, but was not predicted to be a stroke, which could be deadly. While a false positive would mean extra costs and unnecessary treatment. As per the precision-recall tradeoff, the values have an indirect relationship. This is indicated by both models' evaluation performances between their training and testing performance. As their precision increases, the recall drops significantly. It is more ideal in this situation to have high precision when detecting stroke than higher recall because it is better to have more false positives than false negatives. Even though both are very costly errors, it would be more ideal to leave a patient in the hospital for more extensive checkups, than to discharge a patient who has half their body paralyzed walking without any treatment.

Model Evaluation

Using 10-fold cross validation on both models, we can compare both models to evaluate which one is the better model to predict with. Cross validation will be used on the training data set. In Figure 10, we can see the values of the cross validation (10-fold) resampling results and its accuracy. Additionally, in the same figure, we can see the confusion matrix along with the statistics that will be used to compare to the values of the random forest model. The accuracy of the cross validated matrix is 0.761. Sensitivity aka the recall value, or the ability to correctly identify the patients with a stroke, is 0.724. Specificity is the proportion of true negatives out of those who do not have a stroke and the value is 0.799. The positive predictive value (PPV) is the probability that the person who is predicted to have a stroke will truly have a stroke. The negative predictive value (NPV) is the same, but with results of no stroke. The values are 0.785 and 0.741 respectively.

For random forest, we did the same cross validation on the training set to achieve a confusion matrix and calculations which can be seen in Figure 15. The accuracy, sensitivity, specificity, PPV, and NPV are: 0.865, 0.835, 0.890, 0.885, 0.842, respectively.

To compare the results and make a meaningful conclusion of these values, we can say that the values for the random forest are slightly better than the logistic regression model. All the values are larger, indicating better detection rates. This is important when it comes to detecting and predicting weather or not a paitient has a stroke. This means that the logistic regression model will be a better model when it comes to unseen data points and for future data points.

Conclusion

To conclude, the better model of the two is the random forest. After oversampling, splitting, and training each of the data, we can decisively say that the random forest model is better at predicting future data points, although not by a landslide. It is important to indicate that this dataset was extremely unbalanced and even with oversampling the minority, the model had overfitted the datapoints a bit too well, indicating that the model may have trouble with future datapoints. Overall, comparing the two models, random forest is the better model to predict stroke.

From viewing the two methods, we can conclude that your age, and glucose levels have a big impact on predicting strokes for a patient, while having a heart disease or hypertension are important, they do not have as great of an impact as the other two variables.

Bibliography

1. Khintibidze, Lasha. “Visualize Confusion Matrix Using Caret Package in R.” *Delft Stack*, 16 May 2021, https://www.delftstack.com/howto/r/visualize-confusion-matrix-in-r/.

2. “Precision-Recall.” *Scikit*, https://scikit-learn.org/stable/auto_examples/model_selection/plot_precision_recall.html.

3. “Stroke.” *Centers for Disease Control and Prevention*, Centers for Disease Control and Prevention, 2 Aug. 2021, https://www.cdc.gov/stroke/index.htm.

Tables and Graphs

EDA

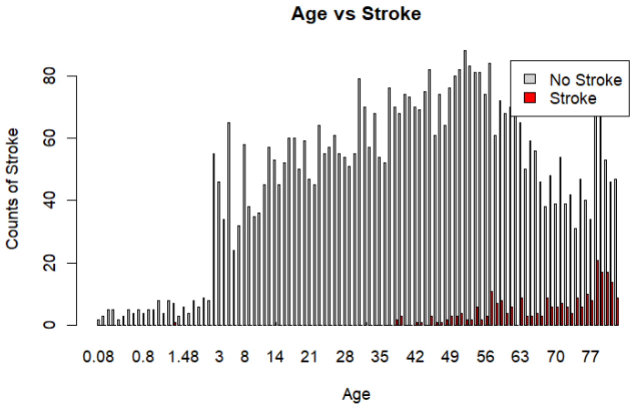


Figure 1

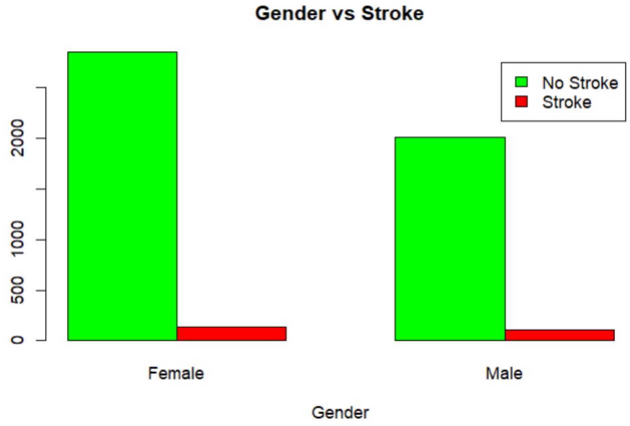


Figure 2

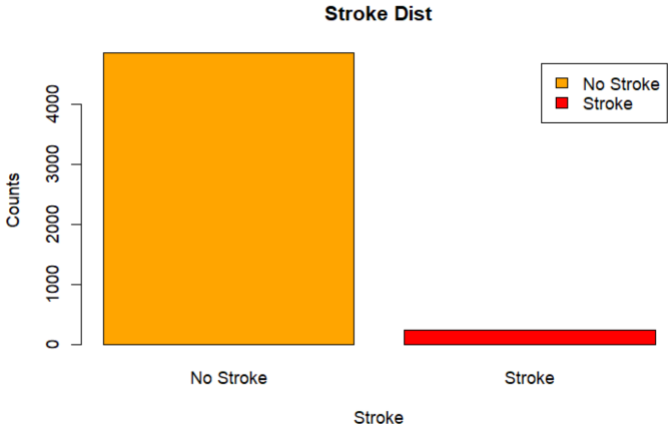


Figure 3

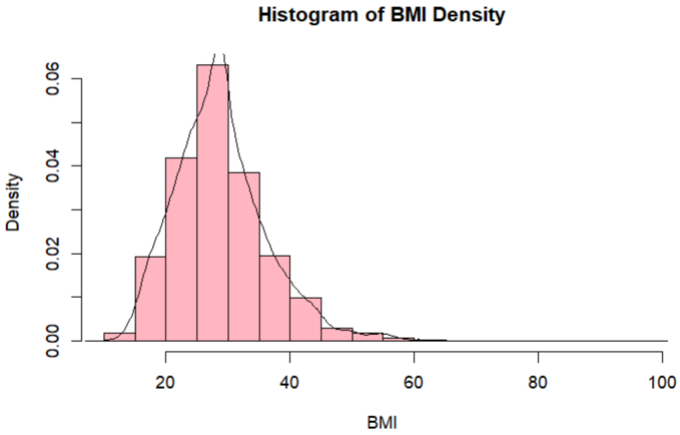
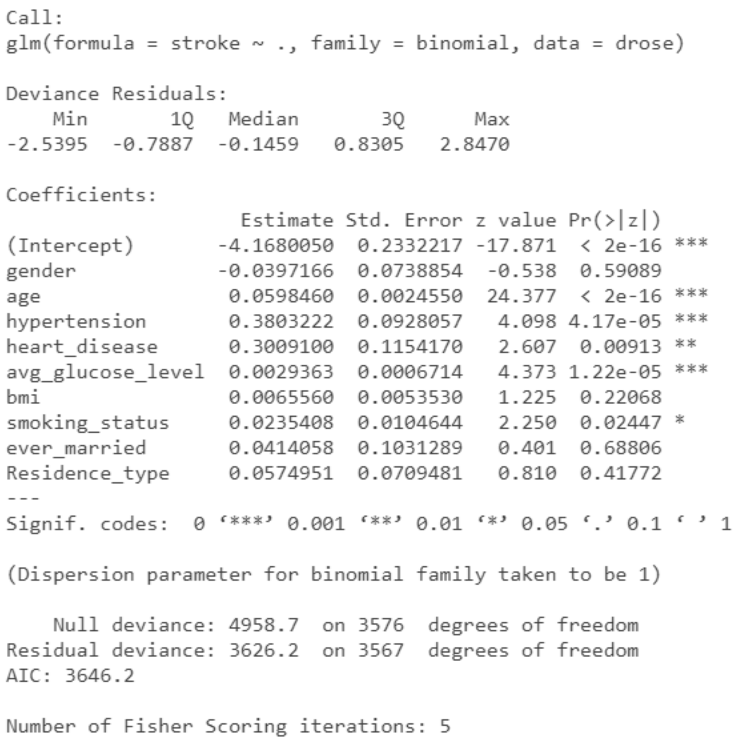


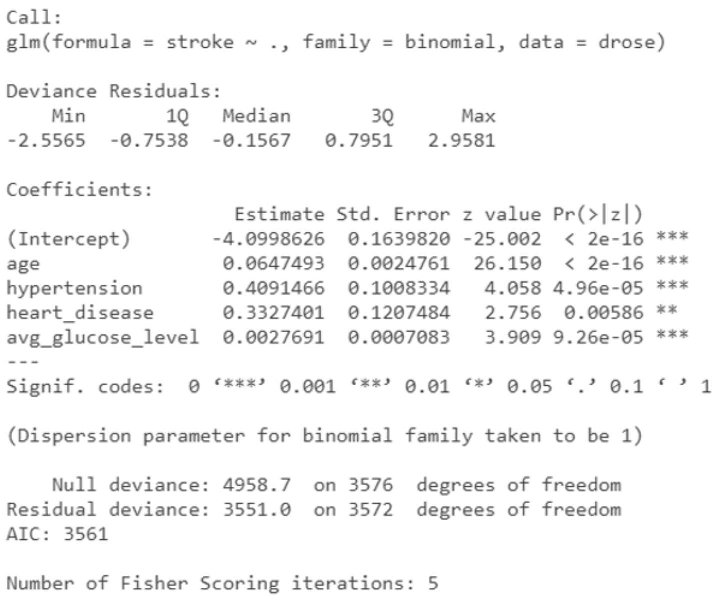
Figure 4

Logistic Regression



```
Call:
glm(formula = stroke ~ ., family = binomial, data = drose)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.5395  -0.7887  -0.1459   0.8305   2.8470

Coefficients:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)       -4.1680050  0.2332217 -17.871  < 2e-16 ***
gender            -0.0397166  0.0738854  -0.538  0.59089
age                0.0598460  0.0024550  24.377  < 2e-16 ***
hypertension       0.3803222  0.0928057   4.098 4.17e-05 ***
heart_disease      0.3009100  0.1154170   2.607  0.00913 **
avg_glucose_level  0.0029363  0.0006714   4.373 1.22e-05 ***
bmi                0.0065560  0.0053530   1.225  0.22068
smoking_status     0.0235408  0.0104644   2.250  0.02447 *
ever_married       0.0414058  0.1031289   0.401  0.68806
Residence_type     0.0574951  0.0709481   0.810  0.41772
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 4958.7  on 3576  degrees of freedom
Residual deviance: 3626.2  on 3567  degrees of freedom
AIC: 3646.2

Number of Fisher Scoring iterations: 5
```

Figure 5



```
Call:
glm(formula = stroke ~ ., family = binomial, data = drose)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.5565  -0.7538  -0.1567   0.7951   2.9581

Coefficients:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)       -4.0998626  0.1639820 -25.002  < 2e-16 ***
age                0.0647493  0.0024761  26.150  < 2e-16 ***
hypertension       0.4091466  0.1008334   4.058 4.96e-05 ***
heart_disease      0.3327401  0.1207484   2.756  0.00586 **
avg_glucose_level  0.0027691  0.0007083   3.909 9.26e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 4958.7  on 3576  degrees of freedom
Residual deviance: 3551.0  on 3572  degrees of freedom
AIC: 3561

Number of Fisher Scoring iterations: 5
```

Figure 6

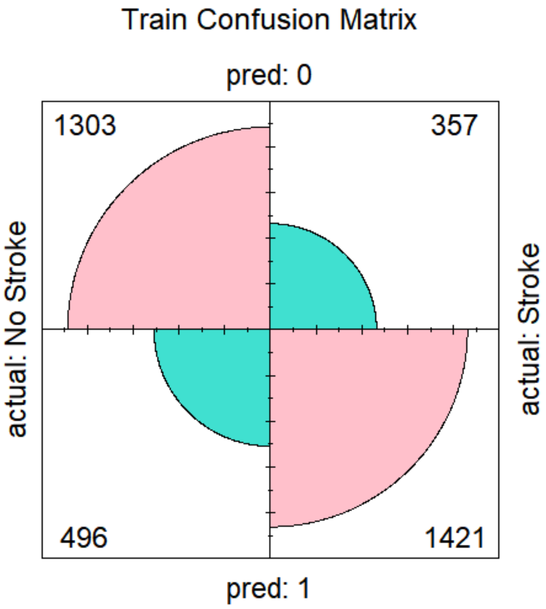


Figure 7

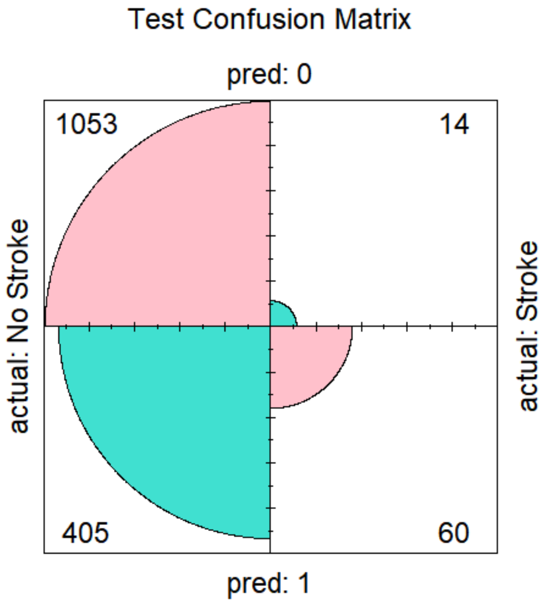
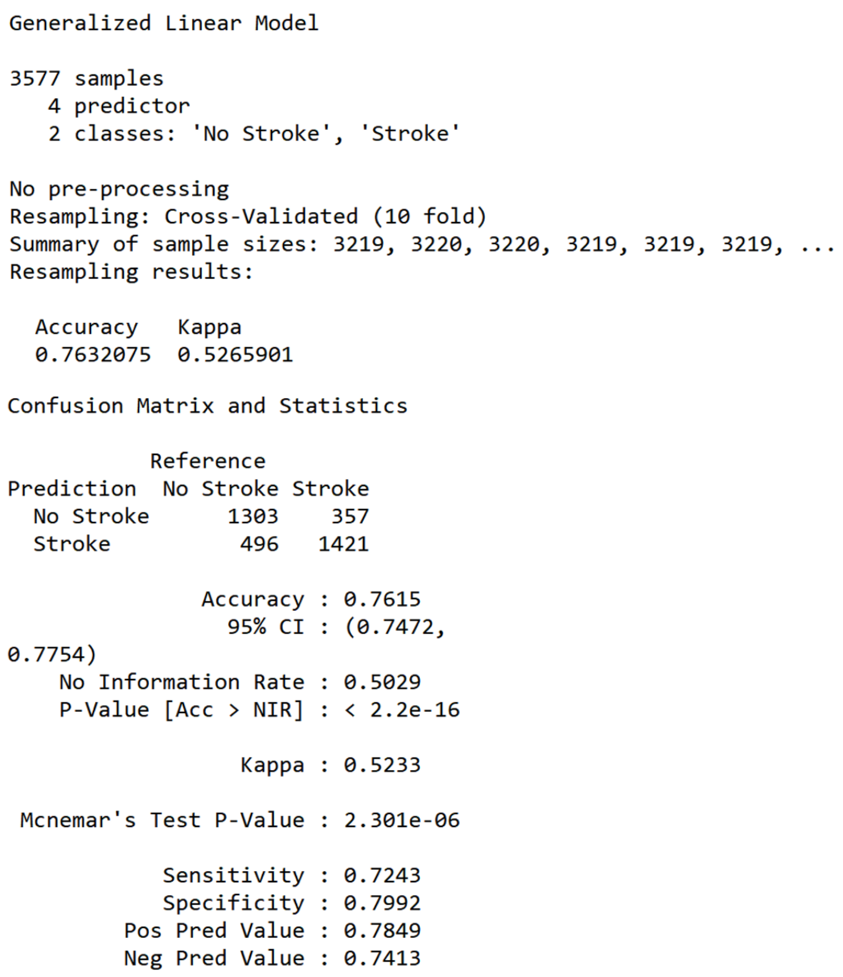


Figure 8

	Logistic Regression		**Random Forest**	
Data Set	Train	Test	Train	Test
Accuracy	0.762	0.727	0.765	0.768
Precision	0.799	0.811	0.769	0.743
Recall	0.741	0.129	0.760	0.141

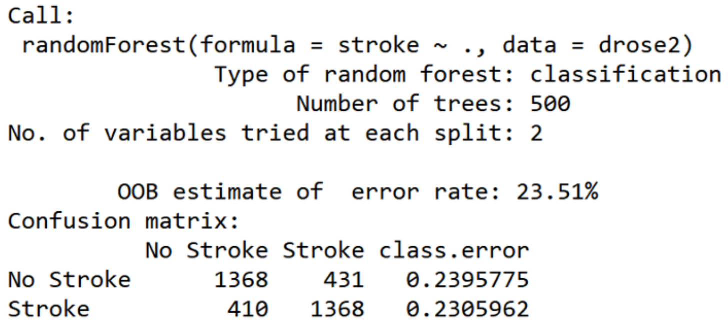
Figure 9



```
Generalized Linear Model

3577 samples
   4 predictor
   2 classes: 'No Stroke', 'Stroke'

No pre-processing
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 3219, 3220, 3220, 3219, 3219, 3219, ...
Resampling results:

  Accuracy   Kappa
  0.7632075  0.5265901

Confusion Matrix and Statistics

           Reference
Prediction  No Stroke Stroke
  No Stroke      1303    357
  Stroke          496   1421

               Accuracy : 0.7615
                 95% CI : (0.7472,
0.7754)
    No Information Rate : 0.5029
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.5233

 Mcnemar's Test P-Value : 2.301e-06

            Sensitivity : 0.7243
            Specificity : 0.7992
         Pos Pred Value : 0.7849
         Neg Pred Value : 0.7413
```

Figure 10

Random Forest



```
Call:
 randomForest(formula = stroke ~ ., data = drose2)
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 2

        OOB estimate of  error rate: 23.51%
Confusion matrix:
          No Stroke Stroke class.error
No Stroke      1368    431   0.2395775
Stroke          410   1368   0.2305962
```

Figure 11

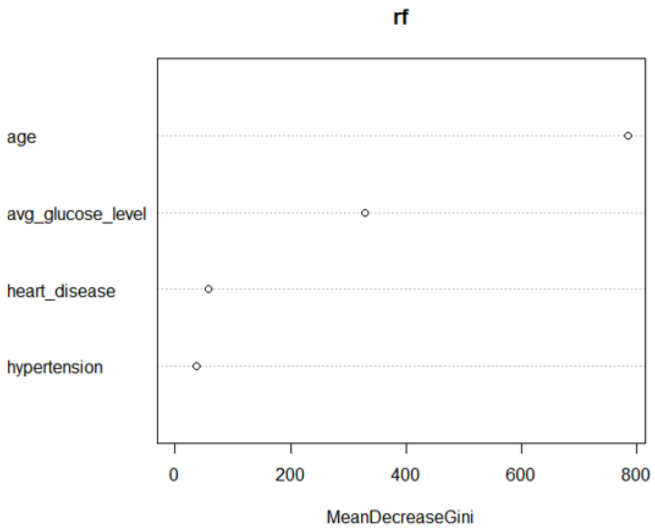
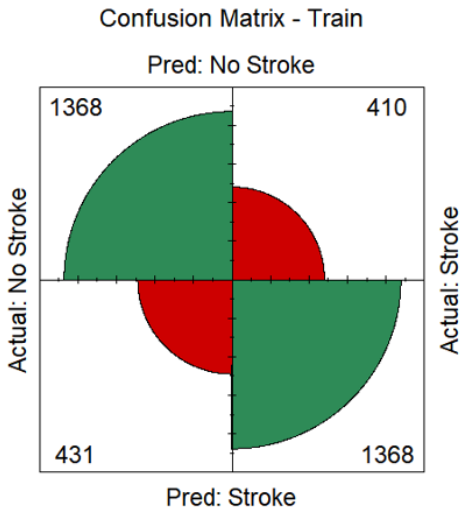


Figure 12



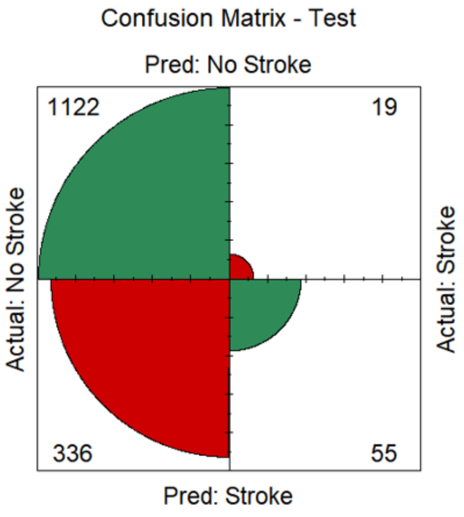
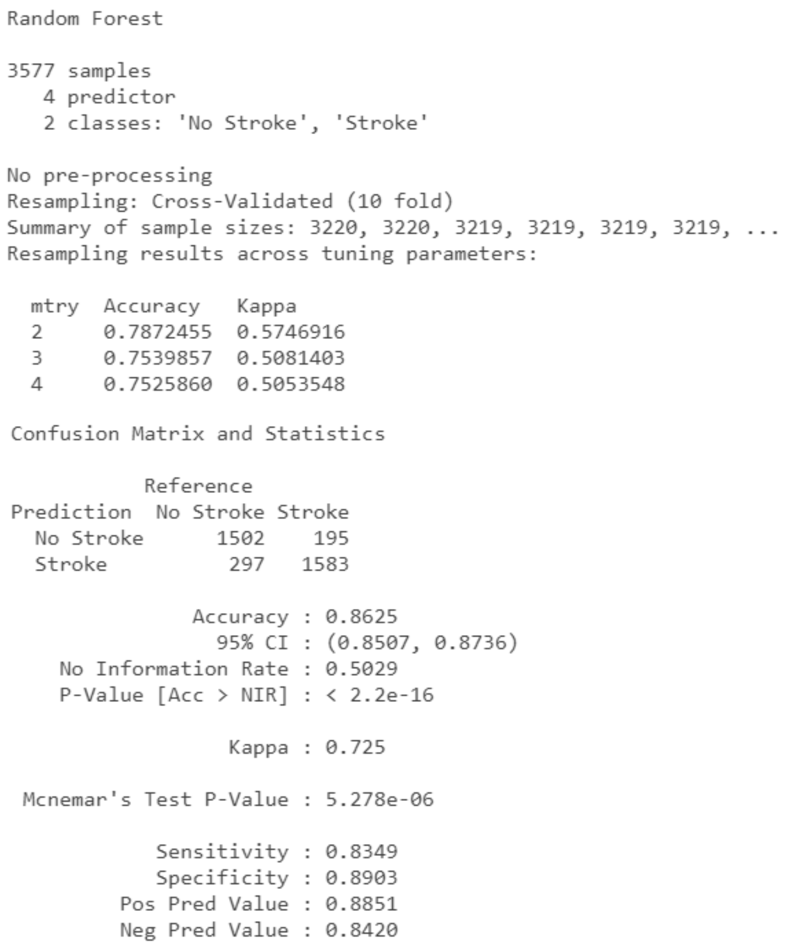


Figure 13

Figure 14



```
Random Forest

3577 samples
   4 predictor
   2 classes: 'No Stroke', 'Stroke'

No pre-processing
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 3220, 3220, 3219, 3219, 3219, 3219, ...
Resampling results across tuning parameters:

  mtry  Accuracy   Kappa
  2     0.7872455  0.5746916
  3     0.7539857  0.5081403
  4     0.7525860  0.5053548

Confusion Matrix and Statistics

           Reference
Prediction  No Stroke Stroke
  No Stroke      1502    195
  Stroke          297   1583

               Accuracy : 0.8625
                 95% CI : (0.8507, 0.8736)
    No Information Rate : 0.5029
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.725

 Mcnemar's Test P-Value : 5.278e-06

            Sensitivity : 0.8349
            Specificity : 0.8903
         Pos Pred Value : 0.8851
         Neg Pred Value : 0.8420
```

Figure 15