**Comparing Decision Tree, Support Vector Machine and Neural Networks for Solving Classiﬁcation Problem brought by Breast Cancer Dataset**

Jigyasa Kohli (jk423), Mahesh Muraleedharan Nair (mmn9), Venkata Jayasimha Hari Chidiri (vc279)

Group 3, MGMT 635 851

Information Systems Department,

New Jersey Institute of Technology, Newark.

**Abstract**

*In this paper, an approach to solving classiﬁcation problems by using decision trees, Support Vector Machine, neural networks system is elaborated. Field of Information theory is used to select a set of important attributes that can be used to classify tuples. Data mining topics will be discussed and a well dataset of Breast Cancer will be used to create a neural network and churn this system’s performance.*

**On this Dataset the various Techniques and Approaches which are being used are:**

**1. Decision Tree**

**2. Support Vector Machine**

**3. Neural Network**

We used Python, scikit-learn Learn for implementing machine learning algorithms. The libraries which are available for the machine learning algorithms, that is, Decision Tree, SVM and Neural Network and used in the project are as follows :

1. **DecisionTreeClassifier**

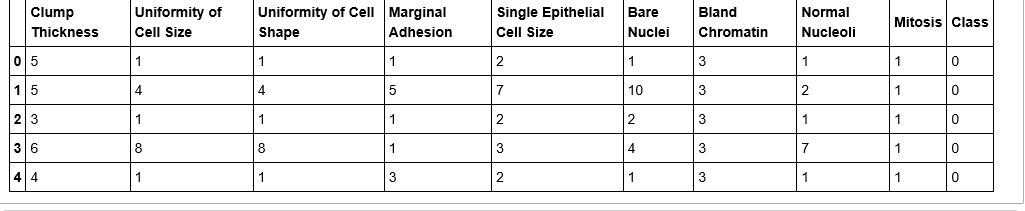
*from sklearn.tree import DecisionTreeClassifier*

1. **SVC (RBF)**

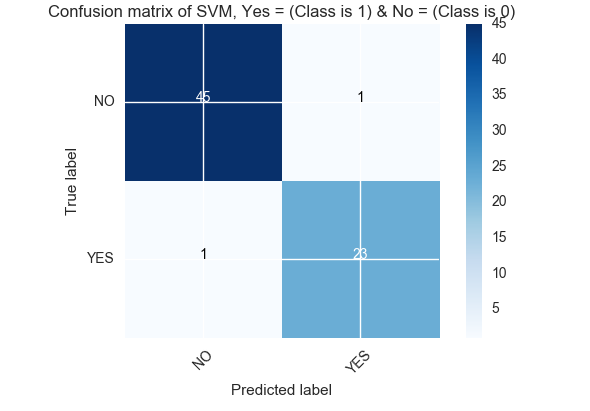
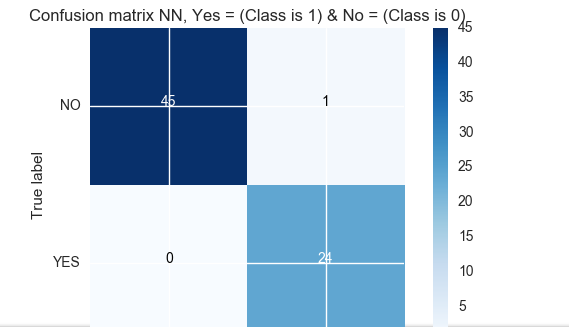
*from sklearn.svm import SVC*

1. **MLPClassifier**

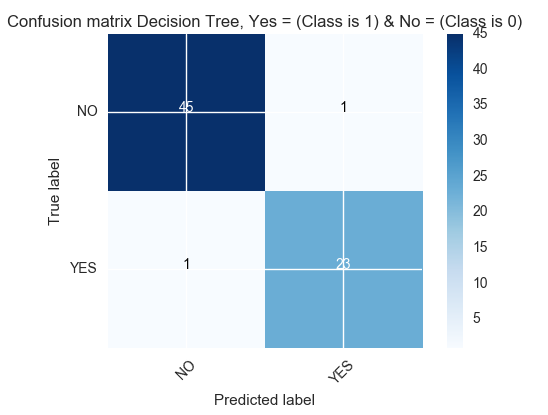
*from sklearn.neural\_network import MLPClassifier*

 **Data would look like the above table for most part of the modeling.**

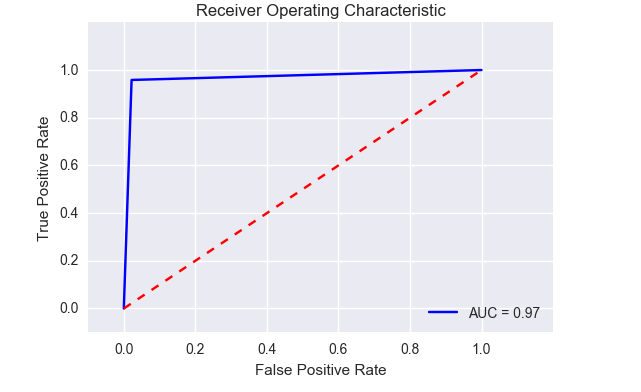
On running the modeling tool like decision trees using scikit on the prepared dataset to create one or more models, a decision tree figure was generated. The max depth was kept at 3, even 4 was found favorable anything else made the tree complicated. ROC was drawn for Neural Network classifier. SVM plot was drawn for the SVM classifier. Finally Confusion Matrixs were drawn for all 3.Hidden layers were kept at 8 to compare with 10 tuples (10 attributes) and so as not to complicate the model and to improve the accuracy.



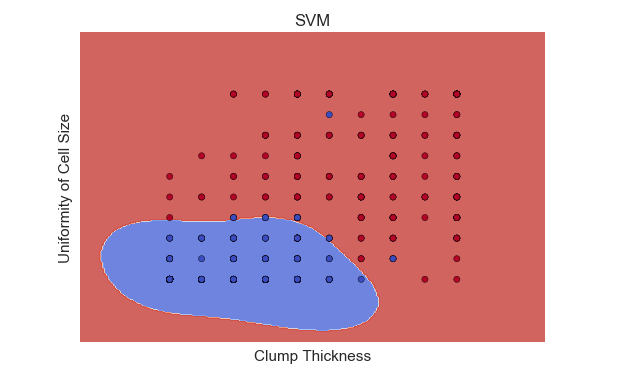
*Fig a1 .Confusion Matrix – NN Fig a2. Confusion Matrix - SVM*



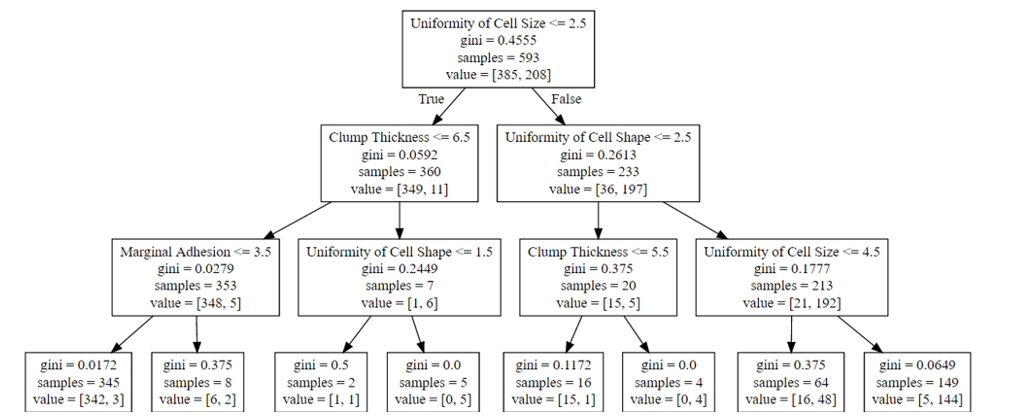
*Fig a3. Confusion Matrix - DT*



*Fig b. ROC*



*Fig c. SVM*



*Fig d. Decision tree*

**DEPLOYMENT:**

**Accuracy of Decision Tree=** 0.971428571429

**Accuracy of MLP Neural network =**0.985714285714  
**Accuracy of SVM Classifier=** 0.971428571429

The results show highly positive results for Neural Network having close to 98.6% accuracy score. Hence highly reliable to predict whether patient is likely to have breast cancer or not, making only inaccurate prediction in the set of 80 test rows . And SVM with accuracy of 97%, produced few outliers as seen in Fig C. Decision Tree classifier too has 97% accuracy, with Uniformity of Cell Size being the prominent node in decision making. Creating a 65:35 rift between the samples.

The accuracy is comfortably good for all three classifiers, Neural Network being very encouraging for this dataset.

**RESULTS AND DESCRIPTION OF RESULTS**

The last column on the classification is whether the patient has breast cancer or not .Thus the prediction is to be done to classify each pattern good or bad.

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Attribute** | **G** | **G’** |
| 1. | Clump Thickness | 0.291058428 | 0.293003247 |
| 2. | Uniformity of Cell  Size | 0.465734095 | 0.568964388 |
| 3. | Uniformity of Cell Shape | 0.460728396 | 0.549866904 |
| 4. | Marginal Adhesion | 0.289549500 | 0.309126018 |
| 5. | Single Epithelial Cell Size | 0.290890823 | 0.393458928 |
| 6. | Bare Nuclei | 0.460469111 | 0.534899357 |
| 7. | Bland Chromatin | 0.387834562 | 0.476396605 |
| 8 | Normal Nucleoli | 0.314596460 | 0.408016262 |
| 9. | Mitosis | 0.0654743423 | 0.207833299 |
| **Average** | | **0.3435344511** | **0.438820341** |

**Table 1.** 10 Attribute gains of the Breast cancer data set.

Accuracy of 99% achieved for Neural Network is very encouraging; this classifier would be a great fit for the Breast Cancer dataset.

**References:-**

<https://biesnecker.com/2015/03/27/visualizing-a-scikit-learn-decision-tree/>

<http://scikit-learn.org/stable/auto_examples/model_selection/plot_confusion_matrix.html#sphx-glr-auto-examples-model-selection-plot-confusion-matrix-py>