ABSTRACT

Breast cancer is the second most prevalent form of cancer worldwide. In this research paper we have proposed a computer aided model based on supervised multi-layered artificial neural network that can assist medical professionals in determination of breast cancer from results of fine needle aspirate (FNA) test on breast mass of the patient. The proposed approach was evaluated on a dataset made available by the University of Wisconsin and our model achieved significant accuracy on the same.

INTRODUCTION

Breast cancer is the second most prevalent form of cancer worldwide. In 2018 alone, there were more than 2 million new cases of breast cancer reported worldwide (as per data from World Cancer Research Fund). One of the most critical test in determination of breast cancer in a patient is a fine needle aspirate (FNA) of the patient's breast mass. A hollow needle attached to a syringe is used to withdraw the required amount of tissue from the area of suspicion. Features are subsequently computed from a digitized image of the sample. These features describes various characteristics of the cell nuclei present in the image in a 3-dimensional space. In this research paper we have focused on developing a computer-aided model based on multi-layered artificial neural network that is capable of conducting analysis of the prior mentioned features and predict malignancy with high degree of accuracy. This will assist medical professionals and minimize decisional errors that can be critical to human life.

DATASET

The dataset used for this research was obtained from [UCI Machine Learning Repository](https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29). The dataset comprises of numerous parameters calculated from digitized image obtained after conducting fine needle aspirate (FNA) test on 569 patients. Each of the record in turn is labelled as either 'B' or 'M' standing for ‘Benign’ and ‘Malignant’ respectively.

LITERATURE REVIEW

A sizeable amount of research has been done till date on machine learning aided breast cancer detection and the associated literature consists of a number of astonishing works. **Guo, H. and Nandi, A.K.** (2006) in their paper had proposed a **Multilayer Perceptron (MLP)** model with retro propagation of error algorithm which attained an accuracy of **96.21%**. **Christobel, A. and Sivaprakasam, Y.** (2011) compared the performance several classification algorithms and was able to achieve **96.99%** predictive accuracy with the **SVM (Support Vector Machine)** model. **Karabatak, M. and Ince, M.C. 2009** developed a diagnosis system based on Neural Networks (NNs) and Association Rules (AR) which was able to achieve classification accuracy of **97.4%**. **Shahnaz, C., Hossain, J., Fattah, S.A., Ghosh, S. and Khan, A.I. 2017** was able to obtain an astonishing predictive accuracy of **98.06%** with their proposed Convolutional Neural Network model. **In our research, we were able to attain a median accuracy of 97.54%** (on a training and test split of 50% each) **with a neural network featured with adaptive learning capabilities.**

EVALUATION METRIC

The various model that has been developed as part of this project will be evaluated based on the area under their individual [Receiver Operating Characteristic (ROC)](https://en.wikipedia.org/wiki/Receiver_operating_characteristic) curve.

DATA ANALYSIS

We found that there are 357 records with an outcome of ‘B’ (stands for ‘Benign’) whereas 212 records have ‘M’ (stands for ‘Malignant’) as outcome.



Figure 1 : Distribution of output class

Thus, the distribution of the output data are slightly skewed in favour of ‘’Benign outcomes.

We subsequently analysed the distribution of the mean of each independent variable in the dataset and it was clear that there were a couple of features that has much higher range of values compared to the others.



Figure 2 : Mean of individual features

This necessitated a statistical normalization to be conducted on the set of independent variables.

Equation 1 : Statistical Normalization

For which we used **min-max normalization** technique as our preferred method.

FEATURE SELECTION

**Principle Component Analysis** : After the normalization of the independent features was completed, we conducted a principle component analysis (PCA) on the feature set to understand the variance explained by each of them.

The test was conducted with increasing test set concentration (starting from 10% to 50% with a step of 5% in between) and in every single instance we found that around 99% of the variance in the data was explained by 15 of the 30 features in feature set.

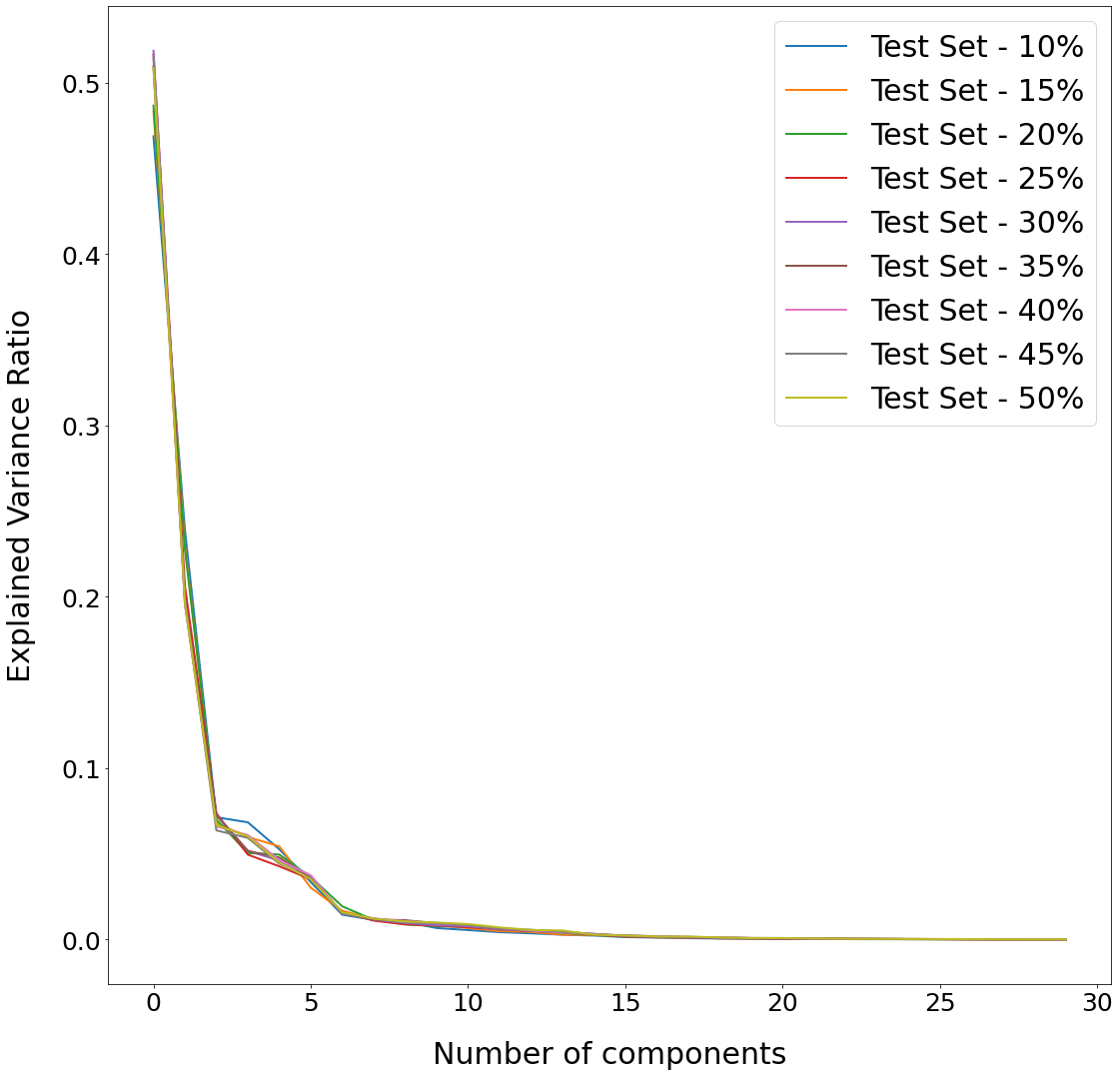


Figure 3 : Result of Principle Component Analysis

Despite the observation we made in our PCA test, we decided to execute an analysis of variance (ANOVA) test on the feature set.

This decision was taken as the principle component analysis test only measures the extent of variance explained by the independent variables. It however does not consider the interaction between the independent variables and the output variable.

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

Figure 4 : Result of ANOVA F-Statistics Test

We conducted this test with increasing test set concentration (starting from 10% to 50% with a step of 5% in between) as well.

We observed that the ANOVA F-Statistics test concluded around 25 of the 30 features to be useful. Though the extent of their usefulness was largely different as made clear by the range of scores obtained by the different variables.

As the main idea behind our feature selection process was to identify the features that had almost negligible impact on the outcome, the result we obtained from the ANOVA F-Statistics Test made it possible. Hence, we decided to eliminate the 5 independent variables from the dataset and use the remaining 25 for our models in the subsequent section.

**MODELS :**

In this section, we have first performed a comparative analysis of the prediction accuracy achieved on the Wisconsin Breast Cancer Diagnosis dataset by the various classification models. Followed by that we introduced a deep learning model and recorded the prediction accuracy it achieved on the same dataset.

**Model Training** : The model training in all scenarios were done by steadily increasing the test set concentration (starting from 10% to 50% with a step of 5% in between).

**Classification Models** : The classification models we used for this analysis are Naïve Bayes (NB), Logistic Regression (LR), K-Nearest Neighbors (KNN), Support Vector Machine (SVM) and Random Forest (RF). We used the default set hyperparameters for each of these models.

We also created an ensemble model (denoted by EN) consisting of the classification models mentioned above and recorded the prediction accuracy obtained by it.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Train | Test | NB | LR | KNN | SVM | RF | EN |
| 90% | 10% | 89.47% | 98.25% | 96.49% | 100.00% | 94.74% | 98.25% |
| 85% | 15% | 91.86% | 97.67% | 97.67% | 100.00% | 96.51% | 98.84% |
| 80% | 20% | 91.23% | 96.49% | 97.37% | 97.37% | 96.49% | 96.49% |
| 75% | 25% | 91.61% | 96.50% | 95.80% | 96.50% | 96.50% | 95.80% |
| 70% | 30% | 90.64% | 95.32% | 96.49% | 97.66% | 97.66% | 96.49% |
| 65% | 35% | 91.00% | 95.00% | 96.50% | 97.50% | 95.50% | 97.00% |
| 60% | 40% | 91.23% | 94.30% | 96.05% | 96.93% | 94.30% | 96.93% |
| 55% | 45% | 92.22% | 94.16% | 95.33% | 96.50% | 94.16% | 97.28% |
| 50% | 50% | 93.33% | 95.09% | 95.79% | 97.19% | 95.09% | 96.84% |

Table 1 : Comparative Analysis of Classification Models

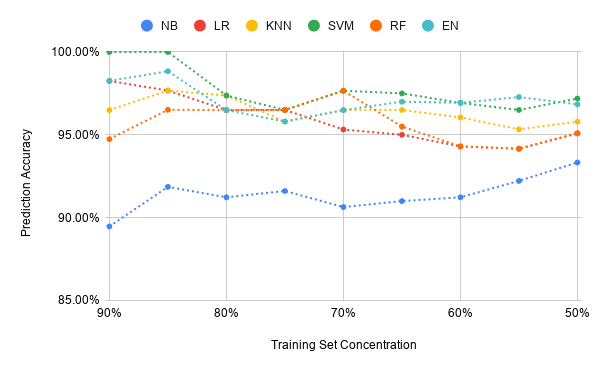


Figure 5 : Predictive Accuracy vs Training Set Concentration

**Deep Learning Models** : In this section, we have used a number of different models based on artificial neural network and have tested their performance on the Wisconsin Breast Cancer Diagnosis dataset.

First we kept the learning rate of the model constant but used two different optimizers and recorded the performance of the model on varying training set concentrations and batch sizes.

Both of the model used the set of hyperparameters enlisted in Table - 2.

|  |  |  |
| --- | --- | --- |
| Parameter | Description | |
| Kernel Initializer | Uniform | |
| Loss | Binary Crossentropy | |
| Activation Functions | 1st Hidden Layer | ReLU |
| Output Layer | Sigmoid |

Table 2 : List of Hyperparameters

In Table – 3 below, we have recorded the accuracy obtained by the model when we used RMSProp (Root Mean Square Propagation) as our optimizer of choice.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Train | Test | BS = 32 | BS = 16 | BS = 8 | BS = 4 | BS = 2 | BS = 1 |
| 90% | 10% | 100.00% | 100.00% | 100.00% | 100.00% | 100.00% | 100.00% |
| 85% | 15% | 100.00% | 100.00% | 100.00% | 100.00% | 100.00% | 99.42% |
| 80% | 20% | 97.37% | 97.37% | 97.37% | 97.37% | 97.37% | 97.37% |
| 75% | 25% | 97.20% | 97.20% | 97.20% | 97.20% | 97.90% | 97.90% |
| 70% | 30% | 97.08% | 97.08% | 97.08% | 97.08% | 96.49% | 97.08% |
| 65% | 35% | 97.50% | 97.50% | 97.50% | 97.50% | 97.50% | 97.50% |
| 60% | 40% | 97.81% | 97.81% | 97.81% | 97.81% | 97.37% | 96.93% |
| 55% | 45% | 97.67% | 97.67% | 97.67% | 97.67% | 97.28% | 96.89% |
| 50% | 50% | 97.19% | 97.19% | 97.19% | 97.19% | 96.84% | 96.49% |

Table 3 : Model with RMSProp Optimizer

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Train | Test | BS = 32 | BS = 16 | BS = 8 | BS = 4 | BS = 2 | BS = 1 |
| 90% | 10% | 160 | 125 | 102 | 92 | 81 | 80 |
| 85% | 15% | 172 | 134 | 110 | 99 | 91 | 83 |
| 80% | 20% | 154 | 108 | 80 | 64 | 52 | 43 |
| 75% | 25% | 131 | 92 | 70 | 56 | 48 | 43 |
| 70% | 30% | 180 | 127 | 96 | 77 | 64 | 55 |
| 65% | 35% | 220 | 164 | 121 | 98 | 86 | 71 |
| 60% | 40% | 258 | 189 | 143 | 113 | 94 | 73 |
| 55% | 45% | 282 | 203 | 151 | 121 | 100 | 71 |
| 50% | 50% | 270 | 211 | 164 | 132 | 95 | 67 |

Table 4 : Epoch Count RMSProp Optimizer

In Table – 5, we have recorded the accuracy obtained by the model when we used Adam (Adaptive Moment Estimation) as our optimizer of choice.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Train | Test | BS = 32 | BS = 16 | BS = 8 | BS = 4 | BS = 2 | BS = 1 |
| 90% | 10% | 100.00% | 100.00% | 100.00% | 100.00% | 100.00% | 100.00% |
| 85% | 15% | 100.00% | 100.00% | 100.00% | 100.00% | 100.00% | 100.00% |
| 80% | 20% | 97.37% | 97.37% | 97.37% | 97.37% | 97.37% | 97.37% |
| 75% | 25% | 97.20% | 97.20% | 97.20% | 97.20% | 97.20% | 97.20% |
| 70% | 30% | 97.08% | 97.08% | 97.08% | 97.08% | 97.08% | 97.08% |
| 65% | 35% | 97.50% | 97.50% | 97.50% | 97.50% | 98.00% | 98.00% |
| 60% | 40% | 97.81% | 97.81% | 97.81% | 97.81% | 98.25% | 97.81% |
| 55% | 45% | 97.67% | 97.67% | 97.67% | 97.67% | 97.28% | 97.28% |
| 50% | 50% | 97.19% | 97.19% | 97.19% | 97.19% | 97.19% | 96.84% |

Table 5 : Model with Adam Optimizer

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Train | Test | BS = 32 | BS = 16 | BS = 8 | BS = 4 | BS = 2 | BS = 1 |
| 90% | 10% | 124 | 105 | 87 | 82 | 74 | 76 |
| 85% | 15% | 120 | 111 | 90 | 83 | 78 | 75 |
| 80% | 20% | 102 | 81 | 67 | 56 | 47 | 39 |
| 75% | 25% | 76 | 60 | 49 | 42 | 37 | 32 |
| 70% | 30% | 118 | 93 | 75 | 61 | 52 | 45 |
| 65% | 35% | 145 | 140 | 94 | 77 | 61 | 53 |
| 60% | 40% | 178 | 142 | 112 | 91 | 72 | 61 |
| 55% | 45% | 190 | 151 | 119 | 98 | 81 | 65 |
| 50% | 50% | 185 | 149 | 124 | 108 | 86 | 72 |

Table 6 : Epoch Count Adam Optimizer

Now, unlike the previous two models, we devised a model that utilized adaptive learning capabilities during its training process. Wherein, it will begin its training with a relatively higher learning rate and in turn will make comparatively larger adjustment to its weights in the early phases. But as the training progresses, its learning rate will attenuate in a step-wise manner and thus the magnitude at which the weights are adjusted will become much smaller later on in its training process. The benefit of this approach is that the model will learn the optimal weights early in its training and then will keep on fine tuning them.

This model also utilized the same set of hyperparameters described in Table - 2 above. However, we chose Stochastic Gradient Descent (SGD) as our choice of optimizer for it.

As SGD can often produce volatile results, for each combination of training set concentration and batch size, we chose to record the median accuracy obtained by this model over 100 iterations. So that, the effect of volatility is minimized.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Train | Test | BS = 32 | BS = 16 | BS = 8 | BS = 4 | BS = 2 | BS = 1 |
| 90% | 10% | 94.74% | 98.25% | 100.00% | 100.00% | 100.00% | 100.00% |
| 85% | 15% | 95.35% | 97.67% | 100.00% | 100.00% | 100.00% | 100.00% |
| 80% | 20% | 94.74% | 96.49% | 96.49% | 96.49% | 97.37% | 97.37% |
| 75% | 25% | 94.41% | 96.50% | 97.20% | 97.20% | 97.20% | 96.50% |
| 70% | 30% | 92.98% | 94.15% | 97.08% | 97.66% | 97.66% | 97.66% |
| 65% | 35% | 92.50% | 93.50% | 96.50% | 97.00% | 98.00% | 98.00% |
| 60% | 40% | 92.54% | 92.54% | 95.61% | 97.37% | 97.81% | 97.81% |
| 55% | 45% | 93.00% | 93.00% | 94.94% | 96.50% | 97.67% | 97.67% |
| 50% | 50% | 94.04% | 93.33% | 95.44% | 96.84% | 97.19% | 97.54% |

Table 7 : Step Decay Model

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Train | Test | BS = 32 | BS = 16 | BS = 8 | BS = 4 | BS = 2 | BS = 1 |
| 90% | 10% | 203 | 209 | 179 | 47 | 41 | 39 |
| 85% | 15% | 205 | 211 | 179 | 49 | 44 | 44 |
| 80% | 20% | 203 | 208 | 189 | 169 | 36 | 27 |
| 75% | 25% | 205 | 200 | 154 | 52 | 27 | 22 |
| 70% | 30% | 206 | 203 | 180 | 59 | 35 | 26 |
| 65% | 35% | 206 | 208 | 192 | 170 | 65 | 31 |
| 60% | 40% | 208 | 209 | 215 | 220 | 180 | 50 |
| 55% | 45% | 208 | 208 | 215 | 224 | 228 | 68 |
| 50% | 50% | 209 | 206 | 198 | 178 | 66 | 57 |

Table 8 : Epoch Count Step Decay Model

**RESULTS**

We observed that the model based on Support Vector Machine consistently outperformed the other classification models in terms of prediction accuracy. However, its predictive prowess was still bettered by each of the three deep learning models we proposed in this research work with a peak predictive accuracy of **97.54%** for a training and test set concentration of 50% each.

**CONCLUSION**

I observed that the time taken by the deep learning model to converge was consistently greater in comparison to those taken by the traditional classifiers for the same training set concentration. In addition, the convergence time of the deep learning model increased steadily with decrease in training set concentration. Thus the use of a deep learning model instead of traditional classifiers came with a trade-off between accuracy and time. This can be problem in several real life applications. Hence, there is visible scope for future improvements in this area with development of deep learning models with better learning rate and faster convergence time.

REFERENCES

Efficient Approaches for Accuracy Improvement of Breast Cancer Classification Using Wisconsin Database

Deep Learning for Automatic Pneumonia Detection