**Machine Learning**

**Report**

Contents

[1.0 The Problem 3](#_Toc140596719)

[1.1 Understanding and Cleaning the Data Cleaning Data: 3](#_Toc140596720)

[2.0 Artificial Neural Networks (ANN) Machine Learning Approach 5](#_Toc140596721)

[3.0 Performance of the Neural Networks 6](#_Toc140596722)

[4.0 Comparison of two Machine Learning Approaches 8](#_Toc140596723)

# 1.0 The Problem

We have chosen to complete our machine learning project on the Wisconsin Breast Cancer dataset which was obtained on Kaggle. Breast cancer is not just specific to women, although it is more common it does affect men as well. Early detection of breast cancer is crucial to successful treatment. In this dataset, we will use real-valued features computed breast masses to classify whether the mass is benign or malignant, which has important implications for whatever clinical practice may be required.

The dataset contains 569 rows and 32 columns of data which need to be computed to present our findings. Each row is an instance, which contains information gathered from analysis of measurements of previous masses. We are hoping to train a model using this dataset to correctly classify the breast masses as either benign or malignant, which is to be our dependent variable.

Before using any machine learning models on the dataset, I will first perform some exploratory data analysis (EDA) to better understand the dataset. Next, I will process the data to make sure it is clean and accurate by deleting any missing values or inconsistencies in the data. This is crucial because faulty or missing values could result in predictions that are incorrect, which might ultimately end in incorrect diagnoses. It is crucial to make sure the data we are working with is as precise as possible because a wrong diagnosis could prevent therapy from being administered or, worse yet, cause someone who does not need it to receive it.

## Understanding and Cleaning the Data Cleaning Data:

When we run the code in the table below we can see that we have an extra column in the data set which is labelled as "Unnamed: 32”, which contains 569 null values. This is a clear indication of inconsistency in our data, which can lead to errors while running any algorithms on the dataset. therefore, it is crucial to remove this column as part of our data-cleaning process. By doing so, we can ensure the accuracy and reliability of our machine learning model, which in turn will help us make informed decisions based on the insights gained from the analysis.

|  |  |
| --- | --- |
| Code: | Results: |
| A picture containing chart  Description automatically generated | Graphical user interface, text, application  Description automatically generated |

We can also clearly see that below that we have a column labelled ID, which is used to identify the people in the dataset, this has no bearing on the machine learning and is unnecessary in predicting if someone is Malignant or Benign, so we have removed this column. Additionally, M and B can easily be converted into a binary classification which will perform much better, this will need altering into 1 or 0.

|  |  |
| --- | --- |
| Results from Info (id and Diagnosis) | ID and Results Values |
|  |  |

In the two graphs below, I have included the code used to remove the two columns that we do not need to compute because they may impair the accuracy of our machine-learning approaches, as well as the choices for substituting M and B with 1 and 0, respectively.

|  |
| --- |
| Dropping the two columns |
|  |

|  |
| --- |
| Replacing the values of M and B (1 and 0) |
| A picture containing text  Description automatically generated |

The graph below shows the distribution of the diagnosis column and takes the variables of 0 and 1 which correspond to benign and malignant respectively. Malignant has 357 and benign has 212, which suggests that the data is slightly imbalanced but not severely enough to cause concern.

|  |
| --- |
| Distribution of Malignant and Benign |
| Chart, histogram  Description automatically generated |

# 2.0 Artificial Neural Networks (ANN) Machine Learning Approach

To tackle the breast cancer classification challenge, I will be using an Artificial Neural Network (ANN), whilst my colleague will be applying a Support Vector Machine (SVM). Neural networks are a family of machine learning algorithms that have shown impressive results in a variety of tasks, including image recognition, natural language processing, and pattern recognition. Through a process known as training, neural networks learn by getting better and better at making predictions. The neural network modifies its parameters (weights and biases) during training to make better predictions. The training procedure consists of three steps: first, the data is processed by the network, which provides an output. Second, the network's parameters are adjusted based on how well it performed. Finally, this cycle is repeated several times until the network's predictions are accurate.

Chart, diagram

Description automatically generated

(Malik, 2019)

Backpropagation is a technique that we employ when training neural networks to determine how to modify the weights in order to lower mistakes. To put it simply, we determine the gradient of the error function for each weight by propagating the mistakes back through the network (Nielsen, M. 2019). Then, we iteratively update the weights using an optimization technique like gradient descent. This aids the network in learning and enhancing its functionality over time. To make sure that the network converges and functions properly, we also modify the learning rate, which determines how frequently the weights are updated.

The following table breaks down the code I have implemented to perform the machine learning on the dataset:

|  |  |
| --- | --- |
| Code Snippet | Explanation |
|  | First, separate the target variables or dependent variables.  Normalising the data using a standard scaler. |
|  | Dive the data into training test sets, with a specific random seed of 50. |
|  | Set up a neural network with one hidden layer and an output node. |
|  | Compile the model using the binary\_crossentropy and accuracy metric. |
|  | Finally, train the data and create the predictions for the test sets. |

When we use neural networks for breast cancer classification, it is important to consider the ethical implications, as biases can occur if the training data is not diverse enough, which may lead to incorrect classification results. We can mitigate this by using more diverse and representative data and testing the algorithm's performance across different populations.

To ensure that the use of machine learning for breast cancer categorization is ethical, keep in mind that machine learning models should not replace, but rather supplement, medical decision-making. To avoid relying too much on the algorithm's output, we should analyse the categorization results with other clinical information and expertise. Furthermore, obtaining informed consent from patients and safeguarding their privacy is critical to ensuring that their data is used ethically and for the intended purposes.

# 3.0 Performance of the Neural Networks

In the first confusion matrix on the left, we have 70 true negatives (TN), 1 false positive (FP), 1 false negative (FN), and 42 true positives (TP). This means that out of the 71 actual negative cases, our model correctly predicted 70 of them (TN), but incorrectly predicted 1 positive case (FP). Similarly, out of the 43 actual positive cases, our model correctly predicted 42 of them (TP) but incorrectly predicted 1 negative case (FN).

In the second confusion matrix, we have 74 TN, 1 false positive (FP), 0 false negatives (FN), and 39 true positives (TP). This means that out of the 75 actual negative cases, our model correctly predicted 74 of them (TN), but incorrectly predicted 1 positive case (FP). Similarly, out of the 39 actual positive cases, our model correctly predicted all of them (TP) and did not make any incorrect negative predictions (FN).

|  |  |
| --- | --- |
| Results from NNL 1 (Random State 42) | Results from NNL 2 (Random State 50) |
| Text  Description automatically generated with medium confidence | A picture containing schematic  Description automatically generated |
|  |  |
| Table  Description automatically generated with medium confidence | A picture containing table  Description automatically generated |

After altering the “random\_state” to 50 instead of 42, the second model has improved rather a lot as it is now showing an accuracy score of 99.12%. The confusion matrix showed that only one sample was misclassified out of 114 samples, with no false negatives and only one false positive. The precision, recall, and F1-score for both classes were 0.98 and 1.00, indicating high precision and recall for both malignant and benign cases.

I attempted to improve the model's performance by increasing the number of epochs, as indicated in the table. The outputs, however, remained consistent with the initial random state of 50. Although I attempted to increase the epochs, it had little effect on the results, and a significant increase could have resulted in overfitting. Overfitting occurs when the model gets overly specialized to the training data and memorizes it instead of learning the broader patterns in the data.

|  |  |
| --- | --- |
| Epochs 500 – Results | |
| A picture containing text, screenshot, font, number  Description automatically generated | A black text on a white background  Description automatically generated with low confidence |

|  |  |
| --- | --- |
| NNL 1 (Random State 42) Training Results | NNL 2 (Random State 50) Training Results |
| Chart  Description automatically generated |  |

Looking at the two tables, we can see that both models have improved in terms of loss and accuracy over time. In comparison to the second model, the first model has a higher binary accuracy score throughout the training process. The second model, on the other hand, has a smaller loss score, indicating that it is better at minimizing the discrepancy between predicted and actual values. This can also be identified by looking at the numeric output as seen below:

|  |
| --- |
| Training History Results in NNL 1 (Random State 42) |
| Graphical user interface, text, table  Description automatically generated |

|  |
| --- |
| Training History Results in NNL 2 (Random State 50) |
| Table  Description automatically generated |

The second model has a higher binary accuracy score nearer the conclusion of the training process, indicating that it is better at predicting outcomes on unseen data. The second model has a lower loss score, which indicates that it is better at generalizing to new data.

To summarize the above, we can see that the two confusion matrices represent the performance of models in classifying both positive and negative cases. The second model with a random state of 50 outperformed the first one, showing higher accuracy, precision, recall, and F1 scores. The training history of both models indicates that the second attempt also had a lower loss score and better generalization ability in predicting outcomes on unseen data. Overall, the second model is a better fit for classifying benign and malignant cases.

# 4.0 Comparison of two Machine Learning Approaches

The best results we received from the Support Vector Machine (SVM) were achieved from the Radial Base Function (RBF). The confusion matrix shows that the SVM algorithm has correctly classified all 71 instances of class 0 and 41 instances of class 1 but has misclassified 2 instances of class 1 as class 0. The ANN algorithm has correctly classified 74 instances of class 0 and 39 instances of class 1, with only 1 instance of class 0 misclassified as class 1. To summarise both models performed really well, however, the SVM misclassified 2 negative cases as positive, and the Neural Network only misclassified 1 negative case as positive.

|  |  |
| --- | --- |
| SVM Confusion Matrix | ANN Confusion Matrix |
|  | A picture containing schematic  Description automatically generated |

Both algorithms gave great results with SVM an accuracy of 98.24% and ANN achieving an accuracy of 99.12% which was only .88% greater.

|  |  |
| --- | --- |
| SVM Accuracy Score | ANN Accuracy Score |
|  |  |

In the classification report, both of our algorithms achieve a high precision and recall value which was very good. The SVM algorithm achieved an average precision of 0.98 and recall of 0.98, whilst the ANN achieved an accuracy of 0.99 and recall of 0.99, so the ANN performed slightly better, however, this suggests that both algorithms are good at correctly identifying between malignant and benign breast cancers.

|  |  |
| --- | --- |
| SVM Accuracy Score | ANN Classification Report |
|  |  |

Overall, both SVM and ANN algorithms have performed well in predicting the diagnosis for the task, with NN performing slightly better overall. This might be because ANNs are more adaptable and capable of handling complex data than SVMs. This is because ANNs can learn several layers of data representation, but SVMs can only learn a single level of representation. To summarise both models performed really well, however, the SVM misclassified 2 negative cases as positive, and the Neural Network only misclassified 1 negative case as positive.