Feature Selection and Pruning in Network Reduction for Classification of Breast Cancer

Mehika Manocha

Research School of Computer Science, Australian National University
u5607484@anu.edu.au

Abstract. A Neural Network(NN) is used to perform classification on the Breast Cancer Wisconsin (Diagnostic) dataset. The network is trained using backpropagation and implements distinctiveness, a network reduction technique to reduce the number of hidden neurons by finding redundant or insignificant neurons. The pruned network is then optimised by feature selection which finds a subset of the informative features from the input features using the genetic algorithm. The aim of the paper is to explore different techniques to reduce the network and, therefore, the performance of the different techniques used in the model are compared to the various similar approaches implemented in existing research papers. The final model with the application of genetic algorithm as well as pruning has an accuracy of 95.29%

Keywords: Neural Networks, Backpropagation, Classification, Breast Cancer Wisconsin Data Set, Pruning, Distinctiveness, Feature Selection, Genetic Algorithm.

1 Introduction

The paper studies the performance of a Neural Network also referred to as a network, model, or NN for the rest of the paper on the Wisconsin Breast Cancer Diagnosis Dataset. The overall aim of the paper is to outline different methods to reduce the complexity of the model by applying feature selection and reducing the number of hidden neurons by pruning for a dataset where accuracy plays a critical role. Network reduction can refer to removing irrelevant features or hidden neurons. This paper aims to implement one of the pruning methods known as distinctiveness, which is discussed in detail in the first sub-section of the Methods section. The second sub-section examines the application of feature selection using evolutionary algorithms, specifically genetic algorithms. The NN was coded in Python using PyTorch and DEAP libraries.

In the Results and Discussion section, these techniques are evaluated using different metrics of performance. The paper with which the results of pruning are compared is "Approximate Distance Classification" (ADC) and is of a similar age to that of the dataset [1]. The accuracy of the pruned model and ADC are compared to draw conclusions about the implementation. Further, the accuracy of the model with genetic algorithms is compared to a modern paper of "Feature Selection Using Genetic Algorithm for Breast Cancer Diagnosis" [2] which is referred as the GA comparison paper for the rest of the report. The last section of Conclusion and Future Work concludes the report by listing some limitations and the steps that can be taken in the future.

2 Background

The dataset created by the University of Wisconsin which can be obtained from UCI Machine-Learning repository contains 569 instances and 32 features with 30 usable for classification [3]. The dataset contains 357 instances of Benign and 212 instances of Malignant making it a reasonably complex dataset. The 30 real-valued features are used to classify the diagnosis of a tumour as either Malignant (0) which is cancerous or Benign (1) which is not cancerous, making it a binary classification problem. The dataset is chosen primarily because it shows a real-world application of neural networks. Furthermore, the dataset is easy to understand, it is not noisy, and contains no missing values. The 30 features are made up of the mean, standard error, and maximum (worst) of each of the 10 values, for instance, the radius of the cell nucleus. Figure 1 gives more insights into the data by depicting the scatter plot of the mean, standard error, and the worst value of the radii of the 569 instances.

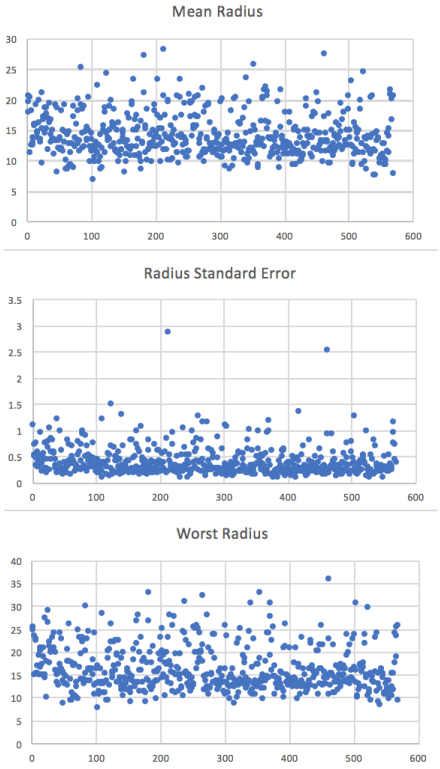

Fig. 1. The diagram shows the measurements recorded in the data for the radius - the mean, standard error, and worst.

The neural network applies the methods of pruning that are discussed in the paper of "Network Reduction Techniques" [4], which is referred as the technique paper throughout the report. The paper was chosen because pruning is known to increase the efficiency of neural networks as well as reduce the space needed. The aim of pruning is to remove neurons which do not provide additional information to the network to maximise its utilisation. Furthermore, there are several disadvantages of using backpropagation with one of them being the difficulty in deciding how many hidden neurons are needed to train the network. The problem lies in being able to find the minimal size of the network to get the optimal results and the technique paper discusses different means of determining the neurons that should be removed. The technique paper aims to improve the slow and inefficient method of backpropagation by discussing distinctiveness which is one of the five different pruning methods listed in the technique paper that can be applied to neural networks. Distinctiveness involves starting with excessive hidden neurons, more than needed, and then removing the ones that are either similar or complementary to get an optimal minimal network [4].

In addition to pruning the hidden neurons, it is also essential to perform feature selection to select the most relevant features as using the entire set of features can result in a huge network which takes a long computation time and has a poor performance especially for large datasets. Feature selection is a process of selecting the input features that are neither redundant nor do have an effect or decrease the accuracy of the model. This is significant as reducing the number of inputs to the model will help improve the accuracy, performance as well as the generalisation of the model [5]. To implement feature selection, genetic algorithms, referred also as GA for the rest of the paper is a type of an evolutionary algorithm that is used. The genetic algorithm is an optimisation technique that is known to be inspired by the natural selection process. It is a stochastic method which evaluates the features in the subset using an objective function. This randomised approach, given the 30 input features in the dataset, is used to select the ones that represent the complete information needed to classify the type of cancer.

GA follows the steps of crossover, mutation, and selection to search the space of features to find the desired features with the methods being highly correlated with the natural section process. Crossover resembles the process of reproduction while mutation like in real-world aims to introduce diversity to the population. The principle of the "survival of the best" is applied for the selection process of the most relevant features. The objective function also known as the fitness function of the algorithm is used to determine the most suitable features. The fitness value is evaluated for a 'chromosome' - a set of features in the 'population' – the entire set of features. This value refers to the performance of the model using this set of features. The technique of genetic algorithm is chosen for feature selection because the algorithm is biologically inspired and seem like the right fit for the application of breast cancer diagnosis.

3 Methodology

3.1 Dataset Manipulation

The first stage involved pre-processing the data by keeping in mind the dataset and the classification problem to be solved. The dataset was modified by removing the first attribute of the dataset – the ID number of the patient. The identifier leads to the grouping of data points and is not relevant for training as it plays no role in the classification of the tumour type. As seen in Figure 1, the values of the attributes differ greatly, thus, normalisation needed to be applied to ensure that all the attributes have the same range of values. String target values of 'M and 'B' were changed to numeric target values of 0 and 1 respectively as the algorithm is performed on numerical values.

The original dataset has no column names, therefore, for analysing the results of feature selection, column names are added to the dataset to clearly see which features are being selected at the end. The names of the features are taken from the UCI Machine Learning Repository [3].

3.2 Implementation of Neural Network

After pre-processing, as per the technique paper about pruning, a feed-forward network with three layers was implemented for the dataset. The network has been designed following the guidelines of the paper – the network uses the sigmoid logistic activation function with backpropagation. Even though the paper states that following these does not generalise the techniques used in the paper, it was nevertheless followed.

The inputs to the network are the 30 features, the hidden layer consisting of 100 neurons at the start, and the output being either 0 or 1 representing the tumour type. The cross-entropy loss function is then used to calculate how far the output is from the target. In addition, the stochastic gradient descent optimizer is used with a momentum of 0.8. The number of epochs used is 3000 considering that if more epochs are run, then the generalization of the dataset will be affected. The value of the hyperparameters like the number of epochs was chosen by trial and error and was increased until it no longer improved the accuracy.

Even if the data is shuffled, splitting the data into testing and training does not ensure that each set contains the same number of classes and this might result in overfitting. A technique to overcome this problem is k-fold cross validation which involves using k-1 samples for training and 1 sample for testing. However, for classification tasks, stratified k-fold should be used instead of k-fold to ensure that each split has roughly the same proportion of target classes. Therefore, the neural network uses stratified 10-fold validation to make better use of the dataset instead of using the traditional split between test and training data [7].

3.3 Pruning using Distinctiveness

After the neural network was set up with all the above components, distinctiveness was implemented which refers to removing neurons which are either similar or complementary. A vector containing the output activation over all the instances is constructed for each hidden neuron. The angle between each pair of vectors is found, and depending on the normalized angle between 0 and 180 degrees, it is decided whether the neurons are to be removed. If the angular separation is less than 15 degrees then the pair of neurons are similar, therefore, any one of them could be removed and its weight could be added to the other. On the other hand, if the angle between them is greater than 165 degrees then both the neurons could be removed as they are considered complementary. To be able to compute this, the cosine similarity is calculated and if that is greater than 0.96 (less than 15 degrees) then the vectors are merged, otherwise, if it is less than 0.96 (greater than 165) then the vectors are removed [4]. The network reports the training accuracy, confusion matrix for training, testing accuracy before pruning, testing accuracy after pruning, and the corresponding confusion matrices.

3.4 Feature Selection using Genetic Algorithms

After implementing a neural network which applies pruning, feature selection was incorporated in the NN using the genetic algorithm. Binary encoding is used to encode the chromosomes with the value of 1 representing a selected feature while the value of 0 corresponds to a feature which is not selected by feature selection. To produce off-springs, the one-point crossover method is used which involves using a random point where the two parents are swapped to produce off-springs. A crossover rate of 0.80 is used which means that 80% of the subset of the features in the next generations are produced by crossover. To be able to introduce variations in the population of features, the bit flip mutation is used which flips each bit and allows more exploration in the population. As there should not be an excess of variation but enough for the algorithm to explore the space of features, a low mutation rate of 0.05 was chosen.

The stopping condition for the algorithm is reaching a fixed number of generations. Having an adequate number of generations is needed for the algorithm to converge, on the other hand, a huge number could lead to the algorithm

having a very long computation time. Therefore, a middle ground between finding a more accurate solution and finding it in shorter time had to be decided, therefore, 30 was selected as the number of generations. While the size of each population in a generation is 30 with the same argument of finding a balance between computation time and search capability and it also corresponds to the number of features of the dataset. The decision of these parameters was made using trial-and-error to find the parameters that provide the best results in a reasonable amount of time.

DEAP (Distributed Evolutionary Algorithms in Python) is the library used to implement genetic algorithms [8]. It provides a toolbox which allows to set the required values and change the different tools in the toolbox to get the appropriate results with ease. Using the toolbox assisted in experimenting with different parameters to get the optimal results. To evaluate the performance of GA, the classification error rate and the number of selected features are reported in the Results section.

4 Results and Discussion

4.1 Pruning

All the experiments were performed ten times to take into the account the difference in random weights assigned at the start and the average of the results are reported. The confusion matrix for testing is depicted in Table 1. Let Malignant be the positive label while Benign the negative label. Looking at the problem at hand, it is important that all false negatives are captured, as missing something that should be classified as cancerous is dangerous, making recall an appropriate measure. So that no cases are missed when the type is cancerous – all the true positives need to be recorded. Precision would be considered less useful in this case as false negatives are more costly than false positives. To summarise these two results, F1 score, the weighted average of precision and recall was also calculated. The formulas for these are: Recall = TP/(TP+FN), Precision = TP/(TP+FP) & F1 score = 2 * (Precision * Recall / Precision + Recall) [9]. In addition to these measurements, the proportion of correct positive (same as precision) and negative results are also reported. These results are summarised in Table 2. From the results, it can be concluded that recall is better than precision which makes the model more complete than exact and that is what was expected. Furthermore, comparing the proportion of positive and negative results that were correct, it is evident that the proportion of correct negative results is greater. This needs to be improved as the aim should be to get as many correct positives as possible as well. To further depict the performance of the neural network, the loss function can be seen in Figure 2. All the measures refer to the model with distinctiveness implemented.

Table 1. The Confusion Matrix for the best accuracy with distinctiveness

Confusion Matrix		**Predicted**	
		M	B
Actual	M	33 (TP)	2 (FN)
	B	4 (FP)	65 (TN)

Table 2. The Confusion Matrix for the best accuracy with distinctiveness

	Backpropagation + Pruning
Precision	0.89
Recall	0.94
F1 Score	1.6732
Proportion of correct positive results	0.94 (TP/(TP+FP)
Proportion of correct negative results	0.97 (TN/(FN+TN)

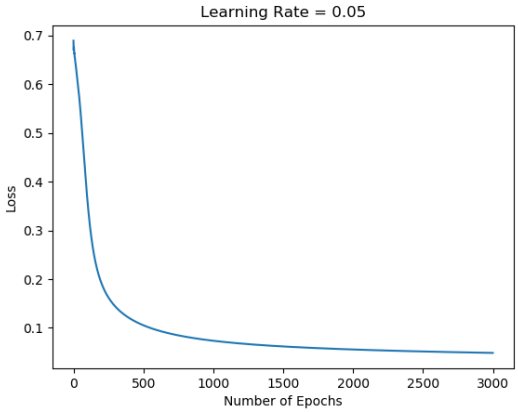

Fig. 2. The loss function with the learning rate of 0.05

.

Furthermore, the purpose of pruning is to get a better structure of the network which leads to a better performance that is not dependent on the dataset and for this accuracy is not the appropriate measure. However, for classifying breast cancer, it is not significant how much time it takes to train the network or how small it is as long as it makes the predictions accurately. Therefore, even though more appropriate measures to record the performance of pruning would be the network size and training time, they are not as significant for the dataset in hand. In all, a weakness of the model is that it reduces the accuracy of the prediction which is much more important for the dataset than having less computation time and smaller size.

4.2 Genetic Algorithms

After analysing the implementation of pruning, the model is evaluated for the application of GA. The "Without Genetic Algorithm" row in the tables refers to the model with only distinctiveness implemented. Table 3 depicts the first measure being evaluated - the number of features selected as the primary aim of GA was to reduce the features in the dataset. As the genetic algorithm is a randomised approach, the features selected each time the program is run is different and, thus, the average number is reported. For one run the selected features were: [‘m_radius', 's_radius', 'm_texture', 'm_perimeter', 'w_perimeter', 's_area', 'w_area', 'm_smoothness', 'm_compactness', 's_compactness', 's_concavep', 'm_symmetry', 's_symmetry', 'w_symmetry', 'm_fractal']. The ‘m’ refers to the mean, ‘s’ refers to standard error, and ‘w’ refers to the worst value of the attribute. Even though different features are selected every time, it was observed that for each run that at least ten features were irrelevant and were disregarded by the algorithm. This shows the power of the algorithm as even in a small dataset of 30 features if only 18 are selected, then for larger datasets, the number would be reduced even further.

Table 3. Comparison of Features using Genetic Algorithm

	Number of Features	Number of Instances
Without Genetic Algorithm	30	569
With Genetic Algorithm	18	569

To further analyse the model with GA, the classification error rate of prior to its application is compared to the value after its application. The classification error rate is used as a measure of performance as the aim of feature selection is to reduce the error of classification by using a smaller subset of features. The difference in error rates are depicted in Table 4 and clearly shows that genetic algorithms decrease the error rate from 7.26% to 5.69% which verifies the claim that application of feature selection reduces the error in classification.

Table 4. Evaluation of Genetic Algorithm using the metric of error rate

	Error Rate (%)	Description
Without Genetic Algorithm	7.26	Neural Network with Pruning
With Genetic Algorithm	5.69	Neural Network with Pruning + Feature Selection

A major drawback of the algorithm was the computation time taken, initially the number of generations was selected as 50, however, that took extremely long to finish. Then with 40 generations, the algorithm took almost two minutes to

converge, therefore, the number was decreased to 30 with only a slight reduction in accuracy and the computation time of about a little more than a minute. The value was not reduced further to ensure that the algorithm has enough number of generations to find the irrelevant features in the dataset before it is stopped.

4.3 Comparison of Techniques

After analysing the neural network in isolation, to get a better understanding, the results of the model were compared to different research papers on the accuracy metric and are tabulated in Table 5. By just using back propagation the average accuracy was 94.23%, as compared to the accuracy of 92.60% of the network in conjunction with the algorithm of distinctiveness. In comparison, the ADC paper reports their best accuracy as 96.60% using the quadratic discriminant function. From the results, it is evident that pruning does not improve the results as the aim of pruning is not to improve the accuracy but instead to make the neural network more efficient and take up less space which is particularly important when the dataset is huge. Furthermore, evaluating the performance of genetic algorithms, it can be concluded from Table 5 that applying genetic algorithms increases the accuracy to 95.29% as expected. The accuracy increases due to the removal of features that did not contribute well to the model, with the possibility of redundant features also decreasing the classification accuracy of the model. Comparing this result to the GA comparison paper which reports their accuracy for the neural network as 97.3%, it shows that there is still some scope for improvement in the model implemented for this paper. In conclusion, it can be clearly seen that optimisation using genetic algorithms increases the accuracy of the model while pruning results in a decrease in accuracy. Comparing against other techniques indicates that are possibilities of improving the current model. Overall, the final model with pruning and feature selection is more accurate than the baseline model with backpropagation.

Table 5. Comparison of Techniques using the metric of accuracy

	Testing Accuracy (%)	Description
Back Propagation (Baseline Model)	94.23	Basic Neural Network
Back Propagation + Pruning	92.60	Pruning of Hidden Neurons using Distinctiveness
ADC	96.60	Quadratic Discriminant Function
Feature Selection + Pruning	95.29	Distinctiveness Genetic Algorithm
GA Comparison Paper	97.30	3-layer NN + Feature Selection using Genetic Algorithm

5 Conclusion and Future Work

This report discusses implementing a network reduction technique – distinctiveness for a feed-forward network with backpropagation and optimising using a feature selection technique – genetic algorithms. Comparing the results shows that application of feature selection is more useful for this dataset than using the technique of pruning as it reinforces that the model for such an application should be more accurate than efficient.

It was reported that the accuracy of the network with pruning was comparable to the ADC paper. However, as the ADC paper does not implement pruning, a drawback is the lack of comparison with a network that also applies pruning on the same dataset. A limitation of the reported results is not evaluating the space measurements and efficiency between the network with backpropagation and the network with pruning and this could be an area of focus in the future. The major limitation of implementing pruning is the reduction in accuracy as compared to the network with backpropagation – a trade-off that is made between accuracy and efficiency.

On the other hand, using genetic algorithms increases the accuracy of the model. GA uses different operators to find the optimal subset of features, like using a different crossovers rate like Shuffle Crossover [10] which can be explored for better optimisation results. There was an attempt to find the best results in a reasonable amount of time if not efficiently, in the future, comparing the parameters which result in the most efficient model versus parameters which result in the most accurate model for the given dataset can be explored. Even though implementing using a built-in library of DEAP maximises the use of existing resources and is a good means to test the applicability of the concept, implementing evolutionary algorithms from scratch will be attempted to enhance the understanding of the algorithm.

Moving forward, this model which uses genetic algorithms for feature selection can also be compared to feature selection using correlation methods. Applying evolutionary algorithms, in general, has a limitation that there is no upper bound on how much time the algorithm would take. GA usually require a lot of computation time to converge which may not seem so much for a relatively small dataset like this but can take very long for large and complex datasets. Furthermore, performing feature selection might lead to loss of data that was captured by the removed features,

therefore, in the future, the methods of feature selection can be contrasted with the application of the feature extraction technique which ensures that the information of the features is preserved.

Moreover, attributes of the dataset used are of type real, in the future, the aim would be to apply the model to categorical attribute types. Furthermore, the task at hand was classification, so the possibility of transforming the model to perform a regression task or a classification and regression task together could be explored. The technique paper discusses four other network reduction techniques that can be also be implemented and the results could be compared with distinctiveness. Moving forward, attempts could be made to improve the efficiency and accuracy of the model.

Lastly, GA does not make any assumptions about the underlying dataset, and because the neural network was only tested with the breast cancer dataset, to be able to validate it further, the techniques could be compared over multiple datasets of various sizes and applications. This will allow to verify the robustness of the NN and, thereby, draw better results about the model. A future direction to improve the model would be to focus on increasing the accuracy as well as making the model more efficient.

In conclusion, the model was not only a means to get a better understanding of Python, PyTorch, and DEAP but also demonstrated the applicability of neural networks, especially distinctiveness and genetic algorithms in a crucial real-world application of determining the tumour type for breast cancer. The future directions that the neural network can take need to be explored and continuous attempts made to improve it.

6 References

1. Adam H. Cannon, Lenore J. Cowen and Carey E. Priebe. Approximate Distance Classification. Department of Mathematical Sciences The Johns Hopkins University.
2. Aalaei Shokoufeh, Shahraki Hadi, Rowhanimanesh Alireza and Eslami Saeid. (2016). Feature Selection Using Genetic Algorithm For Breast Cancer Diagnosis: Experiment on three different datasets. Iranian Journal of Basic Medical Sciences. 19. 476-482.
3. UCI Machine Learning Repository: Breast Cancer Wisconsin (Diagnostic) Data Set", 2018. [Online]. Available: http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29
4. T.D. Gedeon and D. Harris. (1991) "Network Reduction Techniques," Proceedings International Conference on Neural Networks Methodologies and Applications, AMSE, San Diego, vol. 1: 119-126
5. Samina Khalid, Tehmina Khalil and Shamila Nasreen. (2014) "A Survey of Feature Selection and Feature Extraction Techniques in Machine Learning," Science and Information Conference, London, 2014, pp. 372-378. DOI: 10.1109/SAI.2014.
6. Fernando Gómez and Alberto Quesada, "Genetic algorithms for feature selection in Data Analytics | Neural Designer", 2018. [Online]. Available: https://www.neuraldesigner.com/blog/genetic_algorithms_for_feature_selection.
7. Cross-validation: evaluating estimator performance - scikit-learn 0.19.1 documentation", 2018. [Online]. Available: http://scikit-learn.org/stable/modules/cross_validation.html
8. Félix-Antoine Fortin, François-Michel De Rainville, Marc-André Gardner, Marc Parizeau and Christian Gagné. (2012) “DEAP: Evolutionary Algorithms Made Easy”, Journal of Machine Learning Research, pp. 2171-2175, no 13.
9. Towards Data Science – Beyond Accuracy: Precision and Recall, https://towardsdatascience.com/beyond-accuracy-precision-and-recall-3da06bea9f6c
10. A.J. Umbarkar and P.D. Sheth. (2015) “Crossover Operators In Genetic Algorithms: A Review,” ICTACT Journal on Soft Computing, vol. 06, no. 01, pp. 1083–1092.