A detailed comparison of SVM and MLP algorithms applied to the problem of breast cancer diagnosis

Brief description and motivation of the problem

Breast cancer is the second most common cancer worldwide and the most common to be diagnosed in women [5]. Algorithms including neural networks are increasingly used in biomedical research to diagnose illnesses such as cancer which are complex and difficult to detect, and are recognised as having greater accuracy than humans in some domains. Mammograms are a noninvasive and relatively painless diagnostic method: by increasing the accuracy of diagnosis based on this data, it can reduce the costs and subject less women to painful further tests. Using algorithms in this way can aid early interventions, reduce treatment costs, and improve patient outcomes.

This study will compare and contrast the standard and modified performance of Support Vector Machines (SVM) and a Multilayer Perceptron (MLP) in a binomial classification problem using the Wisconsin breast cancer diagnostic dataset available from the UCI Machine Learning repository. This study will refer to a 2018 paper by Agarap, which compared several different algorithms on this dataset [1]. SVM and ANN are both supervised learning techniques, so can be compared fairly as long as experimental integrity is kept (eg using comparable assessment methods, no data snooping etc).

Description of the dataset including data types

I will use the popular Wisconsin diagnostic breast cancer dataset, as even though it is now over 25 years old and contains only 579 samples, the same principles of algorithm design can be applied to newer and larger datasets. This is a binary classification problem: the two labels are ‘M’ for malignant, and ‘B’ for benign. There are 31 numeric continuous features, which can be divided into three categories: for each of ten features, the mean, standard error, and worst measurements are provided, as well as the ID column. There is a small class imbalance, as the mean diagnosis is 0.37, but this is not significant enough to warrant SMOTE or similar balancing measures (however it should be noted that any result below 63% is no better than random chance.) There are no missing values, and each value is recorded to four significant digits. The only non-numeric variable was the target class, which I encoded with 1 and 0 replacing ‘M’ and ‘B’. Agarap standardised his data, but I did not as I was curious as to the effect of scaling on my results.

After drawing histograms of all variables, I found that some were right-skewed, in particular the standard error variables, so I did log transformations to create a normal distribution. I also did a pairwise spearman correlation and found strong correlation between radius, perimeter and area variables. I later tested the effect of PCA on my models to see if it improved them.

A detailed comparison of SVM and MLP algorithms applied to the problem of breast cancer diagnosis

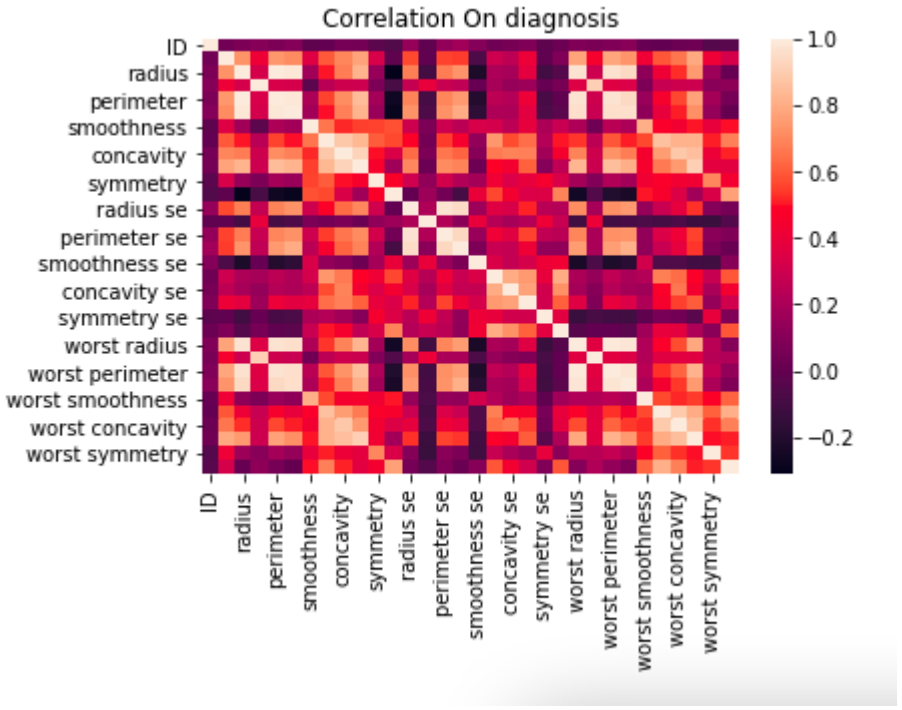

Fig. 1 Initial covariance matrix

Brief summary of the two neural networks with their pros and cons

Support Vector Machines (SVM)

First developed by Vapnik [6], SVM is a supervised learning model which creates a hyperplane in n-dimensional space and can be used for both regression and classification tasks. It works by using the 'kernel trick' which has the effect of mapping data into high dimensional spaces, so that one or more hyperplanes can be built in the feature space. A larger margin denotes a clearly defined class. Points on the boundary line between classes are referred to as support vectors and are used to draw the hyperplane/s. SVM is commonly used in facial detection and text/image classification.

Advantages

SVM is a popular model as it does not need much feature engineering to give a good performance and can work with sparse data. It works particularly well when there is a clear separation of data, and is also likely to have a short training time. Finally, it performs well on high dimensional data as it uses the dot product, so can theoretically have infinite dimensions.

Disadvantages

It does not perform well on large datasets, has a tendency to overfit, and is not probabilistic.

Multilayer Perceptron (MLP)

A Multilayer Perceptron, also known as an Artificial Neural Network, is designed based on the architecture of the brain's neurons, pioneered by McCulloch and Pitts (1943) [7]. ANN consists of an input layer, one or more hidden layers which extract useful features, and an output layer with output classes. The input and hidden layers may be modified using weights and biases to improve performance. In the first forward pass, the weights and biases are randomly initialised, which creates an output that can be compared with the target output,

creating an error signal which is backpropagated to the input layer and the weights are updated.

Advantages

ANNs are fault tolerant, can work with incomplete data, can be parallelised which reduces computational load, and consequently they work well with large datasets.

Disadvantages

The biggest disadvantage of ANNs is that they are known to be a 'black box' model, i.e. it is difficult to know how decisions have been reached, and thus poses a problem for researchers interested in accountable algorithms. They also take longer to train than SVMs due to their consideration of all data. Finally, they have a problem with falling into local minima when optimising using Stochastic Gradient Descent (SGD), which needs to be mitigated.

Hypothesis statement (5%)

Based on the findings of Agarap (2018), I expect that the MLP model will achieve higher accuracy, as Agarap achieved a 99.04% accuracy, compared to 96.09% for SVM. However, I expect MLP to have a longer training time. I expect feature engineering to have little impact on performance, but that hyperparameter tuning will improve both, and parameter tuning will significantly affect the performance of MLP.

Description of choice of training and evaluation methodology

Firstly, the data was shuffled and split into 70% training/validation, 30% test, in line with Agarap's methodology, in order to compare models for each algorithm. Once I identified the two best-performing algorithms, I retrained them on the whole 70% training/validation set and used the final 30% as holdout data.

The training methodology entailed a grid search to optimize hyperparameters for both SVM and MLP. I used a fivefold stratified cross-validation which used the same amount of samples from each target (benign and malignant). This should increase sensitivity and specificity according to [3.

For the final algorithm comparison, the best performing two models were retrained on the 70% data, and then assessed on the testing data. I evaluated using a combination of accuracy, ROC, precision and recall, in order to get a fuller understanding of the performance of each model. Accuracy is often prized, but an accurate model may nonetheless have high variance or bias, which needs to be considered in evaluation.

Choice of parameters and experimental results (5%)

SVM

The hyperparameters I tuned for SVM were kernel, C and gamma functions. The most important is the kernel as it determines the function used to map points into a feature space. C is the misclassification/error penalty, to control the tradeoff between decision boundary and misclassification term. A small value of c creates a small margin hyperplane. Gamma affects the fit: a lower value of gamma will loosely tested linear, polynomial, sigmoid and RBF kernels, C values of [0.1, 1, 10, 100], and gamma values of [1, 0.1, 0.01, 0.001, 0.0001]. I also tested gamma functions scale and auto. Aside from these hyperparameters,

A detailed comparison of SVM and MLP algorithms applied to the problem of breast cancer diagnosis

SVM can be improved through preprocessing and feature engineering, which I did try in order to improve the results.

MLP

Standard MLP architecture has three layers: one input, one hidden and one output, with randomised weights. For the hyperparameter tuning, I tested activation function, alpha value, hidden layer sizes, learning rate, and solver.

I also tested the effects of momentum, weight decay, and early stopping, and decided to implement early stopping to save computation power.

A “Bayesian Regulation backpropagation” training function is used to update the neurons weights. A softmax output function, which is commonly used for classification problems, was chosen, giving a probabilistic decision for the mutual exclusive outputs of the classification process; consequently, a cross entropy performance function was applied [8]. I chose a softmax function as the final layer, as this should reduce overfitting. As part of the initial neural network process, it usually begins with random weight values to avoid networks being stuck in the same local minimum each time they are trained. The random weights in place will cause network to be initialized differently in each training process, hence causing the final results and accuracies to be different. I set the number of epochs to 30, to avoid overfitting. In addition, the samples used for training, validating and testing will also affect the accuracy. This effect can be minimized by doing a k-fold cross-validation for the validating 4 stage, but not for the testing one (new data).

Analysis and critical evaluation of results (20%)

SVM

Starting accuracy of 94% was already high, but in the medical field it is important to get it really high. Performed PCA to see if it would help which revealed a strong correlation between radius, perimeter and area, so eliminated 6 variables (3 each for perimeter and area). I was not expecting this to improve results, as SVM generally performs well without feature engineering. However, it did improve by 2 percentage points. I then ran a grid search hyperparameter tuning, which returned linear kernel, C of 100 and gamma score, returning a high accuracy of 0.97, and a precision and recall of 0.98. The fact that precision and recall were the same is likely due to overfitting, which may have been exacerbated by the use of stratified train and test.

SVM had a very good performance even without training, which reflects both the low difficulty and size of the dataset and the performance of SVM generally. In medicine it is important to minimise the false negatives, as this could have severe consequences for the lives of breast cancer patients. Therefore accuracy is the most important metric.

MLP

In the initial implementation, the model got exponentially worse over successive epochs, so by epoch 11, the training loss was infinite. I managed to stop this by increasing the learning rate, however I still found that a learning rate of 0.001 was best for this model, reducing the training loss by a factor of 6. A learning rate of 0.01 also greatly improved the model, so this could also be considered.

I experimented with the size and number of hidden layers, and found that the loss could be decreased with a single hidden layer of size 568. Since most of the literature used a simple three-layer architecture for MLP, I decided that this was most appropriate. I did experiment with a four-layer architecture, similar to a probabilistic neural network (PNN) but this did not increase accuracy significantly. Initially, I found early stopping useful, but once I decreased the learning rate, I found my neural network was taking longer so it did not improve performance; weight decay and momentum did not affect the network,

According to the sklearn MLP classifier, the best parameters were:
{'activation': 'tanh', 'alpha': 0.05, 'hidden_layer_sizes': (50, 50, 50), 'learning_rate': 'constant', 'solver': 'adam'}

My results for MLP are significantly worse than those achieved by other researchers, including Agarap: my most accurate model achieved an accuracy of 0.63%, which is the same as classifying all instances as benign. As he did not specify his parameters, I do not know the exact reason. It is possible that scaling the data would improve this, however I did not find that in my results. I managed to significantly reduce the training loss but the accuracy stayed the same. I believe this is because my data is too sparse: according to [8], sparse data can negatively affect the performance of neural networks.

The MLP model was quicker to train than expected, and hyperparameter grid search actually took less time than SVM due to the parallelisation. However, it very quickly fell into local minima due to the use of stochastic gradient descent, after which it was difficult to improve the model. This performance is likely due to the sparseness of the data, indicating that MLP would not be a good choice here, as maybe bootstrapping would be better.
MLP is sometimes considered to be self-validating, so cross validation may have been unnecessary, but I implemented it in order to ensure fair comparison. I do not think that cross validation had much effect on MLP but I think it did improve the SVM model, reducing bias and generalisation error.

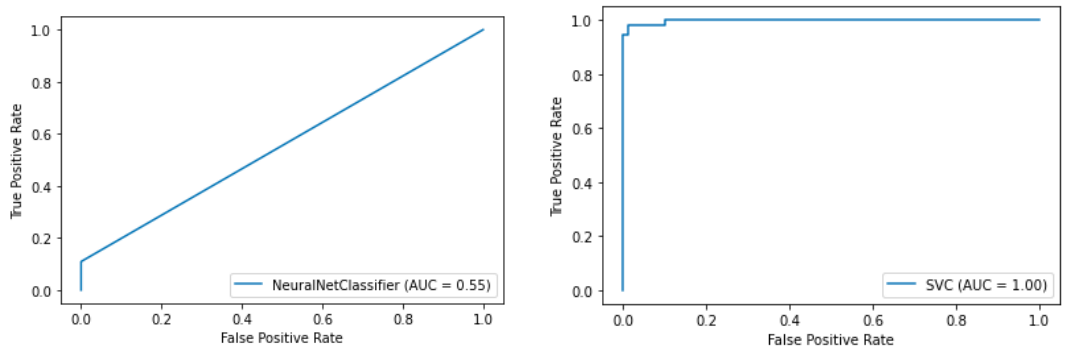

Fig. 2 comparison of AUC of neural network and SVM.

The AUC shows that unfortunately according to my results, the neural network is no better than chance, as classifying all instances as 0 would be more accurate. My AUC varied with the model between 0.55 and 0.46. Although Agarap achieved a better result with an ANN, most other researchers have achieved better results with a SVM, so I am not really surprised by this result [4].

Conclusions, lessons learned and future work

A detailed comparison of SVM and MLP algorithms applied to the problem of breast cancer diagnosis

I would conclude that for this dataset SVM is the most appropriate choice, based not only on my findings but also the literature available.
In future, I hope that more work is done to make more recent similar datasets available, as the field of breast cancer diagnosis has moved on significantly, and the quality of data available has a huge impact on the performance of algorithms which are so crucial in diagnosis. In more complex or larger datasets, it is likely that MLP will perform better, which is reflected in the growing importance of neural networks in the medical field.
I would also be interested in trying SMOTE or similar techniques to resample data, since stratifying my test and train made a significant difference to the performance of both models.
Hyperparameter tuning can improve both SVM and MLP algorithms, especially activation function for MLP and kernel function for SVM.
The fact that precision and accuracy were the same shows the dataset was too easy.
The data could potentially be binned as an alternate problem, with the number of bins as a hyperparameter.
Combined models are also popular, so a SVM-ANN model would theoretically give a very high accuracy, as has been seen in some literature.
Finally, it would be interesting to compare both these algorithms with others, as has been done in other literature, with various parameters and hyperparameters.

References

[1] A. F. M. Agarap, 'On breast cancer detection: an application of machine learning algorithms on the wisconsin diagnostic dataset', in *Proceedings of the 2nd International Conference on Machine Learning and Soft Computing*, New York, NY, USA, Feb. 2018, pp. 5–9, doi: 10.1145/3184066.3184080.

[2] F. Burden and D. Winkler, 'Bayesian Regularization of Neural Networks', in *Artificial Neural Networks: Methods and Applications*, D. J. Livingstone, Ed. Totowa, NJ: Humana Press, 2009, pp. 23–42.

[3] V. R. Elgin Christo, H. Khanna Nehemiah, B. Minu, and A. Kannan, 'Correlation-Based Ensemble Feature Selection Using Bioinspired Algorithms and Classification Using Backpropagation Neural Network', *Computational and Mathematical Methods in Medicine*, Sep. 23, 2019. https://www.hindawi.com/journals/cmmm/2019/7398307/ (accessed Mar. 04, 2021).

[4] S. Ghosh, S. Mondal, and B. Ghosh, 'A comparative study of breast cancer detection based on SVM and MLP BPN classifier', in *2014 First International Conference on Automation, Control, Energy and Systems (ACES)*, Feb. 2014, pp. 1–4, doi: 10.1109/ACES.2014.6808002.

[5] J. Hernández Santiago, J. Cervantes, A. López-Chau, and F. G. Lamont, 'Enhancing the Performance of SVM on Skewed Data Sets by Exciting Support Vectors', in *Advances in Artificial Intelligence – IBERAMIA 2012*, Berlin, Heidelberg, 2012, pp. 101–110, doi: 10.1007/978-3-642-34654-5_11.

[6] V. Vapnik, 'Knowledge transfer in SVM and neural networks | SpringerLink'. https://link.springer.com/article/10.1007/s10472-017-9538-x (accessed Apr. 12, 2021).

[7] L. Vig, 'Comparative Analysis of Different Classifiers for the Wisconsin Breast Cancer Dataset', *Open Access Library Journal*, vol. 1, no. 6, Art. no. 6, Sep. 2014, doi: 10.4236/oalib.1100660.

[8]

A detailed comparison of SVM and MLP algorithms applied to the problem of breast cancer diagnosis

W. Yue, Z. Wang, H. Chen, A. Payne, and X. Liu, 'Machine Learning with Applications in Breast Cancer Diagnosis and Prognosis', *Designs*, vol. 2, no. 2, Art. no. 2, Jun. 2018, doi: 10.3390/designs2020013.