Medical Diagnosis

CAB320 Assignment 2

Liam Percy | N9959807 | 5/6/2020

1 Introduction

A record of tumour measurements taken from cancer patients has been collected and stored in a filesystem. Each individual record contains features pertaining to the tumour such as texture, symmetry, and area. More importantly however, each record also contains an entry specifying whether the tumour is malignant or benign. Given this information, we can utilise machine learning algorithms to build classifiers that are able to predict whether a new tumour record will be either malignant or benign.

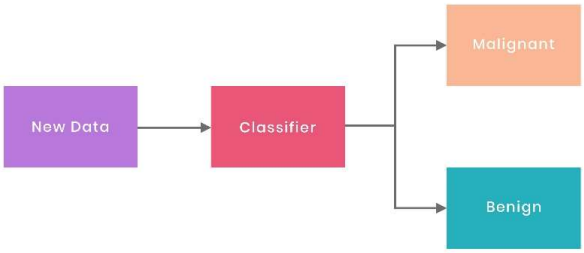

Figure 1: Basic Flow Diagram of Classification Process

Four different types of classifiers were developed in this project, including a decision tree, nearest neighbour, support vector machine and a neural network. This report aims to outline the methodology behind the development of these classifiers, and further, evaluates and compares the performance of each.

2 Methodology

2.1 Overview

To evaluate the performance of each classifier, it is important for us to first separate our records into a collection of training data and test data. Training data is used to improve a classifier's ability to group, or *classify,* new information. Essentially, it is informing the classifier on how it should make decisions.

Each classifier will have a set of hyperparameters that we can manipulate, to improve its performance. To fine-tune our selected hyperparameters for each classifier, cross validated grid search was used. This method is part of the sklearn python library for machine learning, and performs an exhaustive search over specified parameter values for a given classifier. We can then return the optimal hyperparameter for any given metric, such as accuracy, precision, or recall. Once we have selected our hyperparameters, fine-tuned them using cross validation and built up our classifier, we can use our test data set to evaluate its performance.

The ratio of test data to training data will affect the performance of our classifiers. If we have too little training data, then our classifiers will return less accurate results, however if we have too little test data, we will not be able to effectively evaluate our classifiers. By selecting the ratio that maximised the average accuracy across the 4 classifiers, we can ensure we will return the best results possible. As we can observe in table 1, the optimal split for test data is 25% of the original dataset.

Table 1: Average Accuracy Across Classifiers for Different Ratios of Test/Training Data

Test Split (%)	**20**	**25**	**30**	**35**	**40**
Classifier	**Accuracy (%)**				
Decision Tree	*93*	*90*	*88*	*85*	*86*
Nearest Neighbour	92	92	92	92	*90*
Support Vector Machine	93	94	94	94	*93*
Neural Network	90	95	93	92	*65*
Average Accuracy (%)	92	92.75	91.75	90.75	*83.5*

2.2 Parameter Selection

As stated above, sklearn's cross validated grid search is used to find the optimal hyperparameter value for each classifier. This method allows us to specify a metric we would like to optimise, based on our hyperparameter value. For the purposes of these experiments, the accuracy metric was chosen. This metric represents the percentage of predicted labels for a set that match their corresponding set of labels in our test data set.

2.3 Results

2.2.1 Decision Tree Classifier

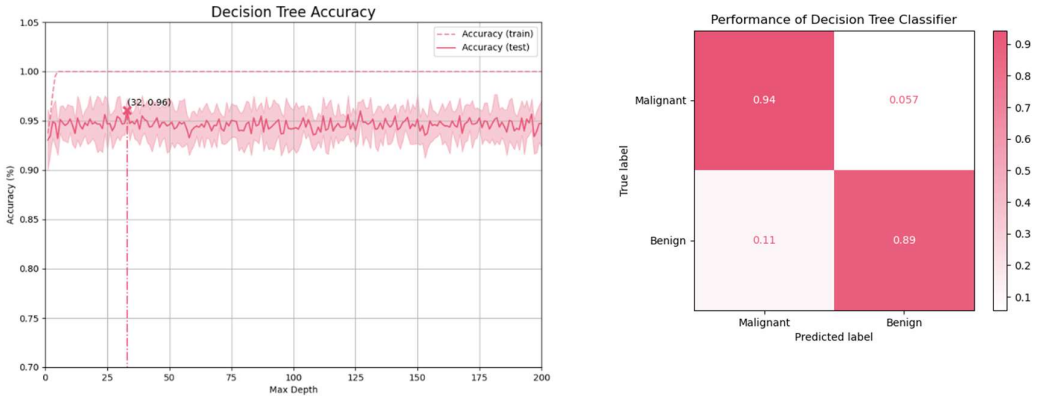

Figure 2: Accuracy of Testing and Training Data Using Decision Tree Classifier and Max Depth Hyperparameter

2.2.2 Nearest Neighbour Classifier

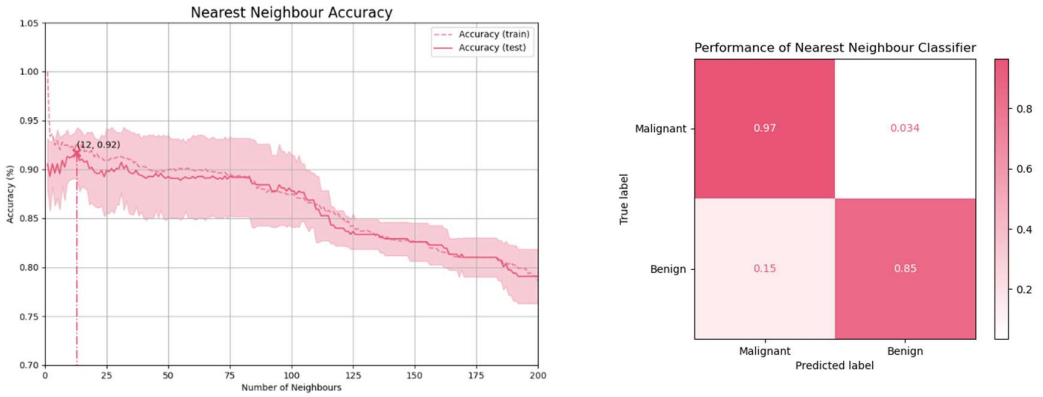

Figure 3: Accuracy of Testing and Training Data Using Nearest Neighbour Classifier and Number of Neighbours Hyperparameter

2.2.3 Support Vector Machine Classifier

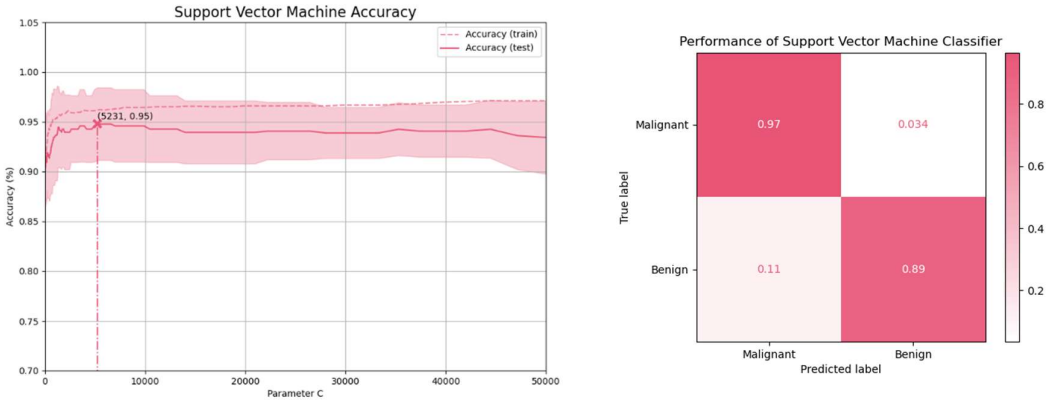

Figure 3: Accuracy of Testing and Training Data Using Support Vector Machine Classifier and Hyperparameter C

2.2.4 Neural Network Classifier

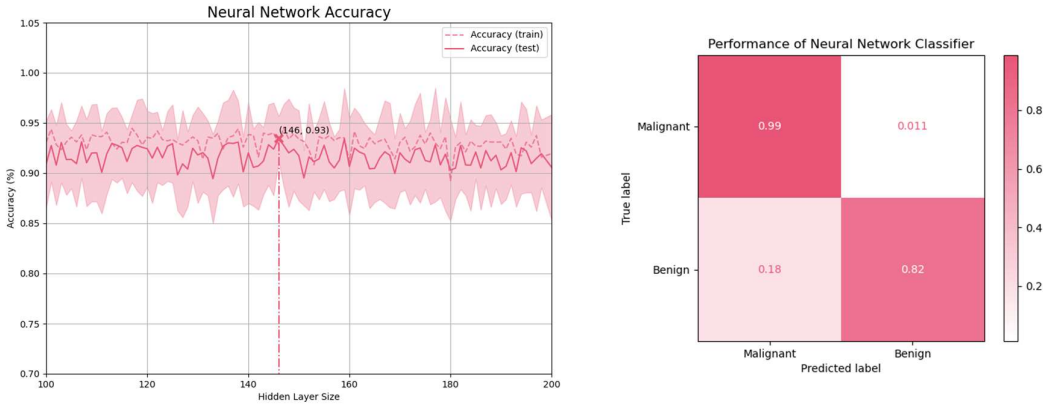

Figure 4: Accuracy of Testing and Training Data Using Neural Network Classifier and Neurons in Hidden Layers Hyperparameter

2.2.5 Overall

Table 2: Classifier Accuracy and Optimal Hyperparameter Value

Classifier	Hyperparameter	Optimal Value	Accuracy
Decision Tree	Max Depth	32	0.96
Nearest Neighbour	Number of Neighbours	12	0.92
Support Vector Machine	Parameter C	5231	0.95
Neural Network	Neurons in Hidden Layers	146	0.93

As we can observe from the table above, the most accurate classifier is the decision tree classifier. Using a max depth of 32, it has an accuracy of approximately 96%.

Looking at the figures individually, we can see that both the decision tree and neural network have a very sporadic accuracy depending on the hyperparameter value selected. It can be noted that the decision tree classifier performs particularly well when classifying test data, once the maximum depth becomes greater than approximately 10.

The nearest neighbour and support vector classifiers both follow a similar trend, reaching a maximum point before trailing off as the hyperparameter value increases. The nearest neighbour accuracy drops of significantly after the optimal hyperparameter is found.

Finally, each figure also has an associated confusion matrix. This matrix displays the normalised spread of correctly and incorrectly predicted labels. As we can observe, each classifier is particularly astute at classifying malignant tumours, however they all struggle to classify benign labels. Misclassification of a benign tumour could potentially cause significant stress, both emotionally and financially, to patients.

3 Performance

3.1 Overview

While the accuracy of a classifier is important in determining its performance, there are also other metrics that we must consider. One of the most important metrics to consider is a classifiers F1-score. This represents a weighted average of the precision and recall. Precision defines the ability of the classifier not to label as positive a sample that is negative, whereas recall defines the ability of the classifier to find all the positive samples.

Additionally, it is important for us to consider the time cost of training each classifier. If one classifier produces roughly the same performance as another, but takes orders of magnitude longer to train, then it cannot be rated to be as effective.

3.2 Results

To find these metrics, sklearn's classification report function was used. This function takes a set of correct values i.e. our test set, and a set of estimated targets, i.e. a prediction set returned by the classifier, and then builds a text report displaying the performance metrics.

Decision Tree

Class	Precision	Recall	F1-Score
Malignant	0.93	0.94	0.94
Benign	0.91	0.89	0.9
Time Cost		6.21s	

Nearest Neighbour

Class	Precision	Recall	F1-Score
Malignant	0.93	0.97	0.95
Benign	0.95	0.87	0.91
Time Cost		36.39s	

Support Vector Machine

Class	Precision	Recall	F1-Score
Malignant	0.93	0.97	0.95
Benign	0.94	0.89	0.92
Time Cost		21.31s	

Neural Network

Class	Precision	Recall	F1-Score
Malignant	0.91	0.98	0.95
Benign	0.96	0.85	0.9
Time Cost		326.56s	

Figure 5: Classification Report for Classifier Testing Data

As we can observe from the figure above, the support vector machine has the greatest performance for identifying and correctly labelling tumours, whereas the decision tree has the worst performance in these areas. As identified in section 2.2.5, we can also observe lower precision and recall scores for each classifier when predicting benign labels.

From a time-cost perspective, we can see that the decision tree returns the optimal hyperparameter faster than the other classifiers, closely followed by the support vector machine and the nearest neighbour classifiers. The neural network takes by far the longest time to identify its optimal parameters, and once they have been identified, the classifier still does not perform as well as the support vector machine, or the nearest neighbour classifier. This is an example of how extended periods of training do not always yield improved performance.

4 Conclusion

Given all the performance data above, we can deduce the ideal classifier. The performance of each classifier is the mean of the combination of accuracy and F1 score.

Table 3: Time Cost, Accuracy, F1 Score and Performance of Each Classifier

Class	Time Cost	Accuracy	Avg. F1-Score	Performance
Decision Tree	6.21s	0.96	0.92	0.94
Nearest Neighbour	36.39s	0.92	0.93	0.925
Support Vector Machine	21.31s	0.95	0.935	0.9425
Neural Network	326.56s	0.93	0.925	0.9125

As we can observe, the support vector machine has the greatest performance when compared to all other classifiers, and thus the argument could be made for its selection. With that said however, these results vary slightly when changing the particular random seed value for NumPy, which is used to split the test and training data. Further testing should be completed to analyse the performance across additional seeds.

Furthermore, it is clear that there are some classification issues with mislabelling benign tumours. It is important that we continue to collect tumour data from patients, so we can more effectively classify our labels.