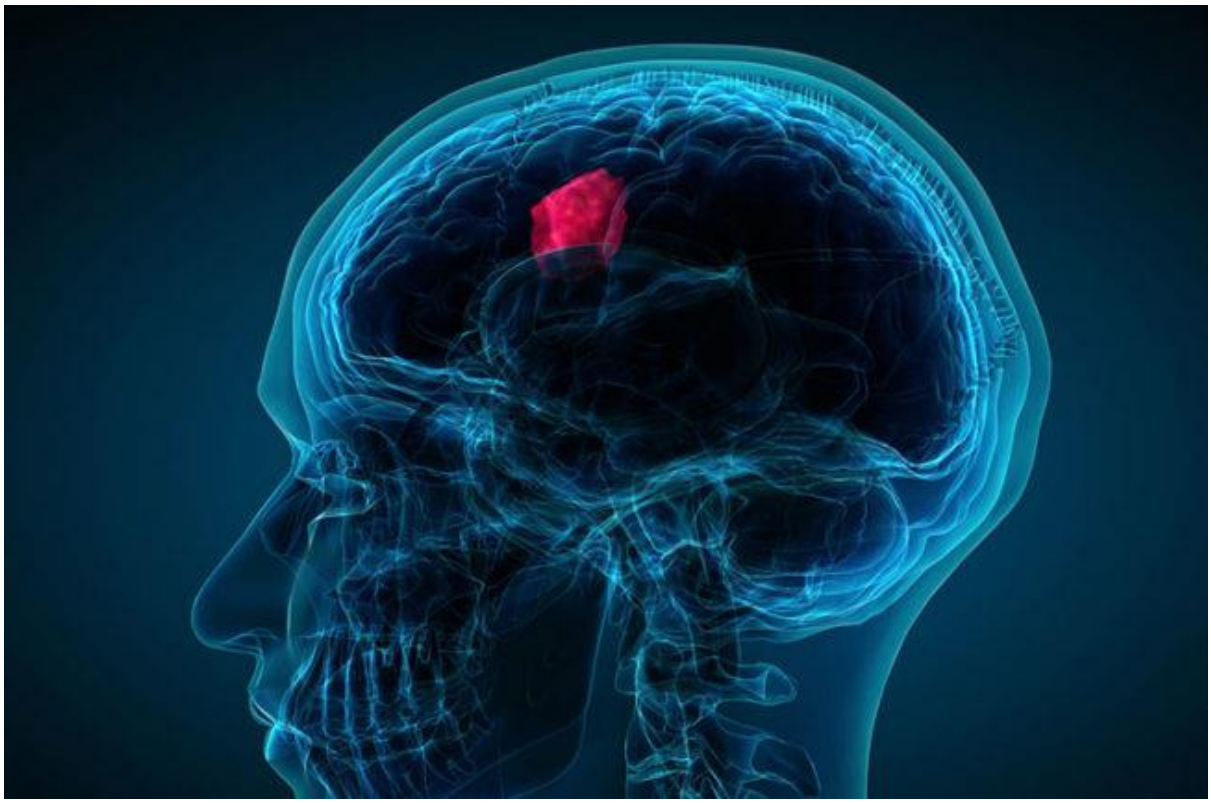
Medical Diagnosis report

CAB320 Artificial Intelligence semester 1 2020

By James Norman



Aim of assessment

The aim of this assessment was to determine whether a tumour was benign or malignant using a set of medical data, which contained information on the tumours such as radius, texture and perimeter. This was done by building a few classifiers that would use this data and try to predict whether the tumour was benign or malignant. These classifiers included, nearest neighbour, decision tree, support vector machine (svm) and neural networks.

classifier overview

In this part of the report we'll take a look at some of the classifiers and how they worked in the code, starting with the nearest neighbour classifier in the code this classifier is fairly simple it just uses the inbuilt sk learn function kneighboursClassifier() and then uses the grid search cv to cross validate the results. But more in depth, in this function The principle behind nearest neighbour methods is to find a predefined number of samples closest in distance to the new point, and predict the label. Next is the decision tree classifier which was once again fairly easily implemented using the decision_tree_classifier() function and then being cross checked by the cv grid. But once again more in depth the goal of the function is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. Next is the svm classifier which again in code uses the simple SVC function and is then cross validated. The svc function which uses a mathematical function to create a set of hyper planes to be used for classification. Lastly the nueral networking classifier was slightly more complex in code using a parameter list and a set of iterations but still using a single mlpclassifier() function and getting cross validated. This uses a multi layered approach to classify the data.

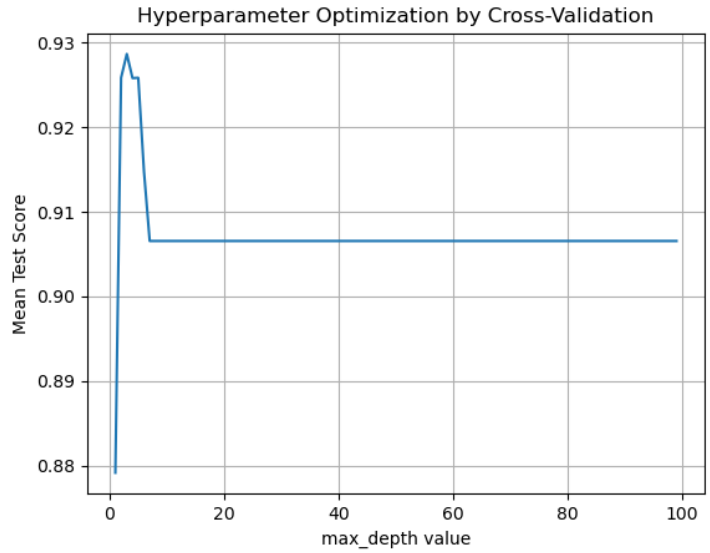
Methodology

The methodology to find the hyperparameter for each classifier was fairly easy using the classified data coming from the sections above the function best_params was used to find the hyperparameter.

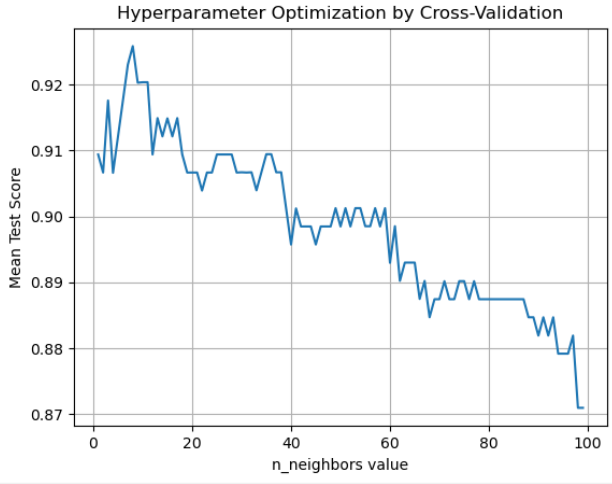
Performance

classifier Type	Average Time Taken over 3 Runs (s)
Decision tree	9.8
Nearest Neighbour	10.7
SVM	5.5
Neural network	8.2

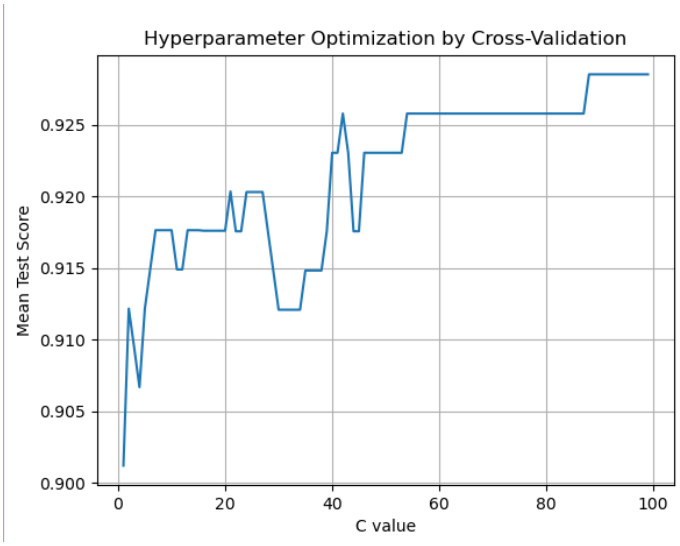
Table 1. Performance comparison between the four classifiers time wise



Graph 1. Hyperparameter optimization graph for Decision tree classifier



Graph 2. Hyperparameter optimization graph for Nearest Neighbour



Graph 3. Hyperparameter optimization graph for SVM

Classification Report for 'Test Data':

	precision	recall	f1-score	support
class B (0)	0.94	0.97	0.96	66
class M (1)	0.96	0.92	0.94	48
accuracy			0.95	114
macro avg	0.95	0.94	0.95	114
weighted avg	0.95	0.95	0.95	114

figure 1. Classification report for the test data of the neural network classifier

Conclusion

To conclude in this assessment four classifier functions were built using functions from the sk learn library to look at a medical data given about a tumour to determine whether is was benign or malignant and give a predication and a predication accuracy, the best hyperparameter was also found for each classifier.