**Full Summary Table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Classifier | precision | accuracy | F1 score | recall | Training time |
| Decision Tree: Gini, Depth: 5 | 0.681 | 0.696 | 0.688 | 0.696 | 0:00:01.944 |
| Decision Tree: Gini, Depth: 10 | 0.778 | 0.77 | 0.775 | 0.773 | 0:00:04.024 |
| Decision Tree: Entropy, Depth: 5 | 0.732 | 0.702 | 0.721 | 0.710 | 0:00:03.440 |
| Decision Tree: Entropy, Depth: 10 | 0.797 | 0.788 | 0.794 | 0.792 | 0:00:07.816 |
| KNN | 0.836 | 0.827 | 0.832 | 0.828 | 0:00:00.628 |
| SVM | 0.010 | 0.105 | 0.019 | 0.1 | 0:03:48.457 |
| Random Forest | 0.837 | 0.832 | 0.836 | 0.835 | 0:00:01.522 |
| Multilayer perception, Neurons: 50 | 0.707 | 0.689 | 0.703 | 0.699 | 0:00:11.924 |
| Multilayer perception, Neurons: 100 | 0.826 | 0.81 | 0.820 | 0.814 | 0:00:14.089 |
| Multilayer perception, Neurons: 100, 10 | 0.010 | 0.097 | 0.018 | 0.1 | 0:00:12.158 |
| Multilayer perception, Neurons: 50, 20 | 0.732 | 0.71 | 0.724 | 0.717 | 0:00:08.708 |
| KNN – Data Split | 0.826 | 0.825 | 0.824 | 0.823 | 0:00:00.464 |
| SVM – Data Split | 0.009 | 0.09 | 0.017 | 0.1 | 0:03:51.497 |

**Decision Tree**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Classifier | precision | accuracy | F1 score | recall | Training time |
| Decision Tree: Gini, Depth: 5 | 0.681 | 0.696 | 0.688 | 0.696 | 0:00:01.944 |
| Decision Tree: Gini, Depth: 10 | 0.778 | 0.77 | 0.775 | 0.773 | 0:00:04.024 |
| Decision Tree: Entropy, Depth: 5 | 0.732 | 0.702 | 0.721 | 0.710 | 0:00:03.440 |
| Decision Tree: Entropy, Depth: 10 | 0.797 | 0.788 | 0.794 | 0.792 | 0:00:07.816 |

A greater max tree depth increased both the training time required and performance in all areas regardless of the criterion used. For our data, given the same max tree depth, Entropy performed better than Gini. Entropy likely performed better as it is better suited to attributes that occur in classes whereas Gini is more suited to continuous attributes.

**KNN, SVM, Random Forest**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Classifier | precision | accuracy | F1 score | recall | Training time |
| KNN | 0.836 | 0.827 | 0.832 | 0.828 | 0:00:00.628 |
| SVM | 0.010 | 0.105 | 0.019 | 0.1 | 0:03:48.457 |
| Random Forest | 0.837 | 0.832 | 0.836 | 0.835 | 0:00:01.522 |

SVM performed extremely poorly compared to the KNN and Random forest classifiers. Not only is the training time required several magnitudes greater, its scores are all significantly worse. Random forest appears to do marginally better than the KNN classifier but has a longer training time. SVM does extremely poorly as it is a linear classifier.

**Multilayer Perceptron**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Classifier | precision | accuracy | F1 score | recall | Training time |
| Multilayer perception, Neurons: 50 | 0.707 | 0.689 | 0.703 | 0.699 | 0:00:11.924 |
| Multilayer perception, Neurons: 100 | 0.826 | 0.81 | 0.820 | 0.814 | 0:00:14.089 |
| Multilayer perception, Neurons: 100, 10 | 0.010 | 0.097 | 0.018 | 0.1 | 0:00:12.158 |
| Multilayer perception, Neurons: 50, 20 | 0.732 | 0.71 | 0.724 | 0.717 | 0:00:08.708 |

Assuming an equal number of hidden layers, increasing the number of neurons increased the performance while also increasing the training time. More neurons allow the classifier to more accurately represent the training data, which likely lead to the improvement in performance. However if there is a second hidden layer with a significantly lower amount of neurons, when compared to the first layer, the classifier actually performs worse (see Neurons: 100 vs [100, 10]). I assume this phenomenon is due to information being lost in the second hidden layer due to an insufficient number of neurons. The 10 neurons are not able to represent all the information output from the previous layer. Otherwise, adding another layer (see Neurons: 50 vs [50, 20]) results in a significant increase in performance. This is likely due to the complexity the classifier gained with the additional layer and weights.

**Data Split**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Classifier | precision | accuracy | F1 score | recall | Training time |
| KNN | 0.836 | 0.827 | 0.832 | 0.828 | 0:00:00.628 |
| SVM | 0.010 | 0.105 | 0.019 | 0.1 | 0:03:48.457 |
| KNN – Data Split | 0.826 | 0.825 | 0.824 | 0.823 | 0:00:00.464 |
| SVM – Data Split | 0.009 | 0.09 | 0.017 | 0.1 | 0:03:51.497 |

For both KNN and SVM, there is a slight decrease in performance across the board. However this decrease does not appear to be very significant and could easily be within the margin of error. This decrease in performance could be explained by the new training data being easier to over fit or the test data included more outliers.