COMP9417 19T3 Homework 2: Applying and Implementing Machine Learning

Question 1 – Overfitting avoidance

Dealing with noisy data is a key issue in machine learning. Unfortunately, even algorithms that have noise-handling mechanisms built-in, like decision trees, can overfit noisy data, unless their "overfitting avoidance" or *regularization* hyper-parameters are set properly.

You will be using datasets that have had various amounts of "class noise" added by randomly changing the actual class value to a different one for a specified percentage of the training data. Here we will specify three arbitrarily chosen levels of noise: low (25%), medium (50%) and high (75%). The learning algorithm must try to "see through" this noise and learn the best model it can, which is then evaluated on test data *without* added noise to evaluate how well it has avoided fitting the noise.

We will also let the algorithm do a limited *grid search* using cross-validation for the best *over-fitting avoidance* parameter settings on each training set.

Running the classifiers

1(a). [0.5 mark]

Run the code section in the notebook cells below. This will generate a table of results, which you should copy and paste **WITHOUT MODIFICATION** into you report as your answer for "Question 1(a)".

The output of the code section is a table, which represents the percentage accuracy of classification for the decision tree algorithm. The first column contains the result of the "Default" classifier, which is the decision tree algorithm with default parameter settings running on each of the datasets which have had 50% noise added. From the second column on, in each column the results are obtained by running the decision tree algorithm on 0% , 25% , 50% and 75% noise added to each of the datasets, and in the parentheses is shown the result of a grid search (http://en.wikipedia.org/wiki/Hyperparameter_optimization) that has been applied to determine the best value for a basic parameter of the decision tree algorithm, namely max_depth (http://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html) i.e., The maximum depth of the tree.