

HW3 of Neural Networks

Yang Guang 1140339072

1. Compare the advantages and disadvantages of these three task decomposition methods.

1.1. One vs One

This method constructs $K(K - 1)/2$ classifiers where each one is trained on data from two out of K classes. When testing, we can adopt a voting scheme: for a given testing sample, its class is the one has most votes.

1.2. One vs Rest

This method requires one classifier per category. The i -th SVM will be trained with all of the examples in the i -th class with positive labels, and all other examples with negative labels. When testing, the classifier which has the highest decision value is the class that sample belongs to.

1.3. Part vs Part

This method is based on the one vs one method. Any two-class problem can be further decomposed into a number of two-class sub-problems as small as needed. We can use the min-max structure to ensemble the models of sub-problems.

1.4. Results

Using RBF kernel, the training result shows in Table1, we show the comparing of the 3 methods. We see that when using RBF kernel, one-vs-rest method shows the best performance and achieve an accuracy of 72.1%. One-vs-one performs a little worse, but when each sub-problem becomes small, the training time reduces largely even though there are more classifiers to be trained. Part-vs-part performs the worst. The possible reason of this may be that the training data set is not so big, so decomposing the data into small data set may suffer from the deficiency of training data.

Table 1: Compare of Training Time and Accuracy of 3 Methods

Method	Train time(s)	Accuracy
One vs One	37.3	0.6895
One vs Rest	145.3	0.7212
Part vs Part	86.5	0.6809

2. Try three different kernel functions, namely linear, polynomial and RBF, and make a comparison on training time and generalization performance

In Table 2 and Table 3, we compare the accuracy and traing time when use 3 different kernels. It can be shown that RBF kernel can achieve the best performance while it also costs most training time. Linear and polynomial kernel perform worse here.

Table 2: Training Time and Accuracy of 3 Kernels with One vs One

Kernel	Train time(s)	Accuracy
RBF	37.6	0.72215
Linear	15.2	0.6232
Polynomial	16.8	0.5105

Table 3: Training Time and Accuracy of 3 Kernels with One vs Rest

Kernel	Train time(s)	Accuracy
RBF	142.1	0.7169
Linear	31.2	0.6098
Polynomial	27.9	0.4806