Available online at www.sciencedirect.com
SCIENCE DIRECT®
Neurocomputing 64 (2005) 211–221
NEUROCOMPUTING
www.elsevier.com/locate/neucom

Online adaptive policies for ensemble classifiers ☆

Christos Dimitrakakis*, Samy Bengio

IDIAP, P.O. Box 592, CH-1920 Martigny, Switzerland

Available online 25 January 2005

Abstract

Ensemble algorithms can improve the performance of a given learning algorithm through the combination of multiple base classifiers into an ensemble. In this paper, we attempt to train and combine the base classifiers using an adaptive policy. This policy is learnt through a *Q*-learning inspired technique. Its effectiveness for an essentially supervised task is demonstrated by experimental results on several UCI benchmark databases.

Keywords: Neural networks; Supervised learning; Reinforcement learning; Ensembles; Mixture of experts; Boosting; Q-learning

1. Introduction

The problem of pattern classification has been addressed in the past using supervised learning methods. In this context, a set of N example patterns $\hat{D} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$ is presented to the learning machine, which adapts its parameter vector so that when input vector x_i is presented to it the machine outputs the corresponding class $y_i \in \{1, 2, \ldots, c\}$, where $c \in \mathbb{N}$ is the number of classes. Let us denote the output of a learning machine for a particular vector x_i as $h(x_i)$. The classification error for that particular example can be designated as $l_i = 1$ if $h(x_i) \neq y_i$ and 0 otherwise. Thus, the classification error for the set of examples $\hat{D}$ can be

☆This work is supported by the Swiss National Science Foundation through the National Centre of Competence in Research on "Interactive Multimodal Information Management" and the European project PASCAL. A shorter version of this paper was presented in ESANN 2004 [7].
*Corresponding author.
E-mail addresses: dimitrak@idiap.ch (C. Dimitrakakis), bengio@idiap.ch (S. Bengio).

0925-2312/$ - see front matter
doi:10.1016/j.neucom.2004.11.031

summarised as the empirical error $\hat{L} = \sum_i l_i/N$, which is simply the zero-one loss. If $\hat{D}$ is a sufficiently large representative sample taken from a distribution D, then $\hat{L}$ should be close to the generalisation error, $L = \int p_D(x)l(x)$. In practice, however, the training set provides limited sampling of the distribution D, leading to problems such as overfitting. Adding the effects of the classifier's inherent bias and variance, we will have $L > \hat{L}$.

Since the generalisation error cannot be directly observed, it has been common to use a part of the training data for validation in order to estimate it. This has led to the development of techniques mainly aimed at reducing the over-fitting caused by limited sampling, such as early stopping and K-fold cross-validation.

Another possible solution is offered by ensemble methods, such as the mixture of experts (MOE) architecture [9], bagging [4] and boosting [8]. The boosting algorithm AdaBoost has been shown to significantly outperform other ensemble techniques. While the good performance of MOE and bagging is related to the independence of experts and the reduction of classifier variance, theoretical results explaining the effectiveness of AdaBoost relate it to the *margin of classification* [13]. See Appendix for a description of margins.

In this work, which is an extended version of a paper presented at ESANN 2004 [7], the possibility of using an adaptive rather than a fixed policy for training and combining base classifiers is investigated. The field of reinforcement learning (RL) [14] provides natural candidates for use in adaptive policies. In particular, the policy is adapted using Q-learning [16], a method that improves a policy through the iterative approximation of an evaluation function Q. Previously Q-learning had been used in a similar mixture model applied to a control task [1]. An Expectation Maximisation based MOE algorithm for supervised learning was presented in [10]. In this paper, we attempt to solve the same task as in the standard MOE model, but through the use of RL rather than expectation maximisation techniques. A description of the similarities between RL and expectation maximisation methods for multi-expert architectures was presented in [15].

The rest of the paper is organised as follows. The framework of RL is introduced in Section 2. Section 2.2 outlines how the RL methods are employed in this work and describes how the system is implemented. Experiments are described in Section 3, followed by conclusions and suggestions for future research.

2. General architecture

The objective in classification tasks is to reduce the expected value of the error, $E\{l\}$. The empirical loss $\hat{L}$ provides an unbiased estimate of this error in the mean-square sense. The suggested classifier ensemble consists of a set of n base classifiers, or experts, $\mathscr{E} = \{e_1, e_2, \ldots, e_n\}$ and a controlling agent that selects the experts to make classification decisions and to train on particular examples. The controlling agent must learn to make decisions so that $E\{l\}$ is minimised. We employ RL for the purpose of finding an appropriate behaviour for the agent.