Combining Methods for Dynamic Multiple Classifier Systems

Amber Tomas

The University of Oxford, Department of Statistics
1 South Parks Road, Oxford OX2 3TG, United Kingdom

Abstract. Most of what we know about multiple classifier systems is based on empirical findings, rather than theoretical results. Although there exist some theoretical results for simple and weighted averaging, it is difficult to gain an intuitive feel for classifier combination. In this paper we derive a bound on the region of the feature space in which the decision boundary can lie, for several methods of classifier combination using non-negative weights. This includes simple and weighted averaging of classifier outputs, and allows for a more intuitive understanding of the influence of the classifiers combined. We then apply this result to the design of a multiple logistic model for classifier combination in dynamic scenarios, and discuss its relevance to the concept of diversity amongst a set of classifiers. We consider the use of pairs of classifiers trained on label-swapped data, and deduce that although non-negative weights may be beneficial in stationary classification scenarios, for dynamic problems it is often necessary to use unconstrained weights for the combination.

Keywords: Dynamic Classification, Multiple Classifier Systems, Classifier Diversity.

1 Introduction

In this paper we are concerned with methods of combining classifiers in multiple classifier systems. Because the performance of multiple classifier systems depends both on the component classifiers chosen and the method of combining, we consider both of these issues together. The methods of combining most commonly studied have been simple and weighted averaging of classifier outputs, in the latter case with the weights constrained to be non-negative. Tumer and Ghosh [8] laid the framework for theoretical analysis of simple averaging of component classifiers, and this was later extended to weighted averages by Fumera and Roli [2]. More recently, Fumera and Roli [3] have investigated the properties of component classifiers needed for weighted averaging to be a significant improvement on simple averaging. Although this work answers many questions about combining classifier outputs, it does not provide a framework which lends itself to an intuitive understanding of the problem.

The work presented here we hope goes some way to remedying this situation. We present a simple yet powerful result which can be used to recommend

L. Prevost, S. Marinai, and F. Schwenker (Eds.): ANNPR 2008, LNAI 5064, pp. 180–192, 2008.

a particular method of combination for a given problem and set of component classifiers. We then apply this result to dynamic classification problems. For the purposes of this paper, we define a *dynamic classification problem* as a classification problem where the process generating the observations is changing over time. Multiple classifier systems have been used on dynamic classification by many researchers. A summary of the approaches is given by Kuncheva [5].

The structure of this paper is as follows: in section 2 we present the model for classifier combination that we will be using. We then present our main result in section 3, and discuss its relevance to dynamic classification and classifier diversity. In section 4 we explore the use of component classifier pairs which disagree over the whole feature space, and then in section 5 demonstrate our results on an artificial example.

2 The Model

Because we are interested in dynamic problems, the model we use is time dependent. Elements which are time dependent are denoted by the use of a subscript t. We assume that the population of interest consists of K classes, labelled $1, 2, \ldots, K$. At some time t, an observation $\boldsymbol{x}_t$ and label y_t are generated according to the joint probability distribution $P_t(\boldsymbol{X}_t, Y_t)$. Given an observation $\boldsymbol{x}_t$, we denote the estimate output by the ith component classifier of $\text{Prob}\{Y_t = k|\boldsymbol{x}_t\}$ by $\hat{p}_i(k|\boldsymbol{x}_t)$, for $k = 1, 2, \ldots, K$ and $i = 1, 2, \ldots, M$.

Our final estimate of $P_t(Y_t|\boldsymbol{x}_t)$ is obtained by combining the component classifier outputs according to the multiple logistic model

$$\hat{p}_t(k|\boldsymbol{x}_t) = \frac{\exp(\boldsymbol{\beta}_t^T \boldsymbol{\eta}_k(\boldsymbol{x}_t))}{\sum_{i=1}^{M} \exp(\boldsymbol{\beta}_t^T \boldsymbol{\eta}_i(\boldsymbol{x}_t))}, \; k = 1, 2, \ldots, K, \tag{1}$$

where $\boldsymbol{\beta}_t = (\beta_{t1}, \beta_{t2}, \ldots, \beta_{tM})$ is a vector of parameters, the ith component of $\boldsymbol{\eta}_k(\boldsymbol{x}_t), \eta_{ki}(\boldsymbol{x}_t)$, is a function of $\hat{p}_i(k|\boldsymbol{x}_t)$, and $\boldsymbol{\eta}_1(\boldsymbol{x}_t) = 0$ for all $\boldsymbol{x}_t$. In this model we use the same set of component classifiers for all t. Changes in the population over time are modelled by changes in the parameters of the model, $\boldsymbol{\beta}_t$.

Before we can apply (1) to a classification problem, we must specify the component classifiers as well as the form of the functions $\boldsymbol{\eta}_k(\boldsymbol{x}_t), k = 1, 2, \ldots, K$. Specifying the model in terms of the $\boldsymbol{\eta}_k(\boldsymbol{x}_t)$ allows flexibility for the form of the combining rule. In this paper we consider two options:

$$1. \;\; \eta_{ki}(\boldsymbol{x}_t) = \hat{p}_i(k|\boldsymbol{x}_t) - \hat{p}_i(1|\boldsymbol{x}_t), \text{ and} \tag{2}$$

$$2. \;\; \eta_{ki}(\boldsymbol{x}_t) = \log\left(\frac{\hat{p}_i(k|\boldsymbol{x}_t)}{\hat{p}_i(1|\boldsymbol{x}_t)}\right) . \tag{3}$$

Both options allow $\eta_{ki}(\boldsymbol{x}_t)$ to take either positive or negative values. Note that when using option (3), the model (1) can be written as a linear combination of classifier outputs

$$\log\left(\frac{\hat{p}(k|\boldsymbol{x}_t)}{\hat{p}(1|\boldsymbol{x}_t)}\right) = \sum_{i=1}^{M} \beta_{ti} \log\left(\frac{\hat{p}_i(k|\boldsymbol{x}_t)}{\hat{p}_i(1|\boldsymbol{x}_t)}\right) . \tag{4}$$

3 Bounding the Decision Boundary

In this section we present our main result. We consider how the decision boundary of a classifier based on (1) is related to the decision boundaries of the component classifiers. The following theorem holds only in the case of 0–1 loss, i.e. when the penalty incurred for classifying an observation from class j as an observation from class j' is defined by

$$L(j, j') = \begin{cases} 1 & \text{if } j \neq j' \\ 0 & \text{if } j = j' \end{cases} . \tag{5}$$

In this case, minimising the loss is equivalent to minimising the error rate of the classifier. At time t, we classify $\boldsymbol{x}_t$ to the class with label $\hat{y}_t$, where

$$\hat{y}_t = \text{argmax}_k \ \hat{p}_t(k|\boldsymbol{x}_t), \tag{6}$$

and $\hat{p}_t(k|\boldsymbol{x}_t)$ is given by (1).

Theorem 1. *When using a 0–1 loss function and non-negative parameter values $\boldsymbol{\beta}_t$, the decision boundary of the classifier (6) must lie in regions of the feature space where the component classifiers "disagree".*

Proof. Assuming 0–1 loss, the decision boundary of the ith component classifier between the jth and j'th classes is a subset of the set

$$\{\boldsymbol{x} : \hat{p}_i(j|\boldsymbol{x}) = \hat{p}_i(j'|\boldsymbol{x})\}. \tag{7}$$

Define $\mathcal{R}_j^i$ as the region of the feature space in which the ith component classifier would classify an observation as class j. That is,

$$\mathcal{R}_j^i = \{\boldsymbol{x} : j = \text{argmax}_c \ \hat{p}_i(c|\boldsymbol{x})\}, \ j = 1, 2, \ldots, K. \tag{8}$$

Hence for all $\boldsymbol{x} \in \mathcal{R}_j^i$,

$$\hat{p}_i(j|\boldsymbol{x}) > \hat{p}_i(j'|\boldsymbol{x}), \text{ for } j \neq j'. \tag{9}$$

Define

$$\mathcal{R}_j^* = \cap_i \ \mathcal{R}_j^i. \tag{10}$$

Then for all i, for all $\boldsymbol{x} \in \mathcal{R}_j^*$,

$$\hat{p}_i(j|\boldsymbol{x}) > \hat{p}_i(j'|\boldsymbol{x}). \tag{11}$$

From (11), for $\beta_{ti} \geq 0, i = 1, 2, \ldots, M$, it follows that for all $\boldsymbol{x} \in \mathcal{R}_j^*$,

$$\sum_{i=1}^{M} \beta_{ti}\{\hat{p}_i(j|\boldsymbol{x}) - \hat{p}_i(1|\boldsymbol{x})\} > \sum_{i=1}^{M} \beta_{ti}\{\hat{p}_i(j'|\boldsymbol{x}) - \hat{p}_i(1|\boldsymbol{x})\}. \tag{12}$$

Similarly, from (11), we can show that for $\beta_{ti} \geq 0, i = 1, 2, \ldots, M$, for $\boldsymbol{x} \in \mathcal{R}_j^*$,

$$\sum_{i=1}^{M} \beta_{ti} \log \left(\frac{\hat{p}_i(j|\boldsymbol{x})}{\hat{p}_i(1|\boldsymbol{x})} \right) > \sum_{i=1}^{M} \beta_{ti} \log \left(\frac{\hat{p}_i(j'|\boldsymbol{x})}{\hat{p}_i(1|\boldsymbol{x})} \right). \tag{13}$$

For the classification model (6), the decision boundary between the jth and j'th classes can be written as

$$\{\boldsymbol{x} : \boldsymbol{\beta}_t^T \boldsymbol{\eta}_j(\boldsymbol{x}_t) = \boldsymbol{\beta}_t^T \boldsymbol{\eta}_{j'}(\boldsymbol{x}_t)\}. \tag{14}$$

Therefore, for the definitions of $\boldsymbol{\eta}_j(\boldsymbol{x}_t)$ considered in section 2, we can see from (12) and (13) that $\mathcal{R}_j^*$ does not intersect with the set (14). That is, there is no point on the decision boundary of our final classifier that lies in the region where all component classifiers agree.

Note that from (11) it is easy to show that this result also holds for the combining rules used in [8] and [2], in which case the result can be extended to any loss function. For example, figure 1 shows the decision boundaries of three component classifiers in a two-class problem with two-dimensional feature space. The shaded areas represent the regions of the feature space where all component classifiers agree, and therefore the decision boundary of the classifier must lie outside of these shaded regions.

This result helps us to gain an intuitive understanding of classifier combination in simple cases. If the Bayes boundary does not lie in the region of disagreement of the component classifiers, then the classifier is unlikely to do well. If the

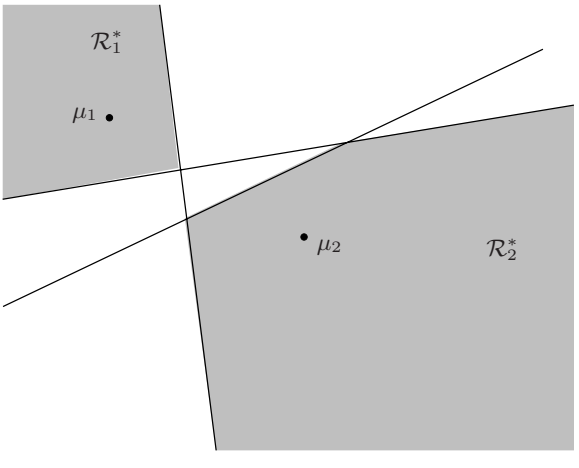


Fig. 1. The decision boundaries of the component classifiers are shown in black, and the regions in which they all agree are shaded grey. $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ denote the means of classes one and two respectively. When using non-negative parameter values, the decision boundary of the classifier (6) must lie outside of the shaded regions.

region of disagreement does contain the Bayes boundary (at least in the region of highest probability density), then the smaller this region the closer the decision boundary of the classifier must be to the optimal boundary. However, clearly in practice we do not know the location of the Bayes boundary. If the component classifiers are unbiased, then they should "straddle" the Bayes boundary. If the component classifiers are biased, then the Bayes boundary may lie outside the region of disagreement, and so it is possible that one of the component classifiers will have a lower error rate than a simple average of the classifier outputs. In this case, using a weighted average should result in improved performance over the simple average combining rule. This corresponds to the conclusions of Fumera and Roli [3].

3.1 Relevance to Dynamic Scenarios

If the population of interest is dynamic, then in general so is the Bayes boundary and hence optimal classifier [4]. However, because our model uses the same set of component classifiers for all time points, the region of disagreement is fixed. Therefore, even if the Bayes boundary is initially contained within the region of disagreement, after some time this may cease to be the case. If the Bayes boundary moves outside the region of disagreement, then it is likely the performance of the classifier will deteriorate. Therefore, if using non-negative weights, it is important to ensure the region of disagreement is as large as possible when selecting the component classifiers for a dynamic problem.

3.2 On the Definition of Diversity

Consider defining the diversity of a set of classifiers as the volume of the feature space on which at least two of the component classifiers disagree, i.e. the "region of disagreement" discussed above. Initially this may seem like a reasonable definition. However, it is easy to construct a counter example to its appropriateness. Consider two classifiers c_1 and c_2 on a two class problem which are such that whenever one of the classifiers predicts class one, the other classifier will predict class two. Then according to the definition suggested above, the set of component classifiers $\{c_1, c_2\}$ is maximally diverse. This set is also maximally diverse according to the *difficulty* measure introduced by Kuncheva and Whitaker [6]. However, using the combining rule (1), for all values of the parameters β_{t1} and β_{t2}, the final classifier will be equivalent to either c_1 or c_2 (this is proved in section 4). Thus although the region of disagreement is maximised, there is very little flexibility in the decision boundary of the classifier as $\boldsymbol{\beta}_t$ varies.

The problem with considering the volume of the region of disagreement as a diversity measure is that this is a bound on the flexibility of the combined classifier. Ideally, a measure of diversity would reflect the actual variation in decision boundaries that it is possible to obtain with a particular set of classifiers and combining rule. However, the region of disagreement is still a useful concept for the design of dynamic classifiers. For a classifier to perform well on a dynamic scenario it is necessary that the region of disagreement is maximised as well as

the flexibility of the decision boundary within that region. One way to improve the flexibility of the decision boundary whilst maximising the region of disagreement (and maintaining an optimal level of "difficulty" amongst the component classifiers) is now discussed.

4 Label-Swapped Component Classifiers

Consider again the pair of component classifiers discussed in section 3.2 which when given the same input on a two-class problem will always output different labels. One way in which to produce such a pair of classifiers is to train both classifiers on the same data, except that the labels of the observations are reversed for the second classifier. We refer to a pair of classifiers trained in this way as a *label-swapped pair*.

In this section we consider combining several pairs of label-swapped classifiers on a two-class problem. The region of disagreement is maximised (as each pair disagrees over the entire feature space), and we increase the flexibility of the decision boundary within the feature space by combining several such pairs.

Suppose we combine M pairs of label-swapped classifiers using model (1), so that we have $2M$ component classifiers in total. An observation $\boldsymbol{x}_t$ is classified as being from class 1 if $\hat{p}_t(1|\boldsymbol{x}_t) > \hat{p}_t(2|\boldsymbol{x}_t)$, i.e.

$$\sum_{i=1}^{2M} \beta_{ti}\eta_{2i}(\boldsymbol{x}_t) < 0. \tag{15}$$

Theorem 2. *Suppose* $\eta_{2i}(\boldsymbol{x}_t) > 0$ *if and only if* $\hat{p}_i(2|\boldsymbol{x}_t) > \hat{p}_i(1|\boldsymbol{x}_t)$*, and that*

$$\eta_{22}(\boldsymbol{x}_t) = -\eta_{21}(\boldsymbol{x}_t). \tag{16}$$

Then the classifier obtained by using (6) with two label-swapped classifiers c_1 *and* c_2 *and parameters* β_{t1} *and* β_{t2} *is equivalent to the classifier* c_i*, where* $i = argmax_j \beta_{tj}$*.*

Proof. From (15), with $M = 1$, we see that $\hat{p}_t(1|\boldsymbol{x}_t) > \hat{p}_t(2|\boldsymbol{x}_t)$ whenever

$$(\beta_{t1} - \beta_{t2})\eta_{21}(\boldsymbol{x}_t) < 0. \tag{17}$$

Therefore, $\hat{p}_t(1|\boldsymbol{x}_t) > \hat{p}_t(2|\boldsymbol{x}_t)$ when either

$$\begin{aligned} &\beta_{t1} < \beta_{t2} \text{ and } \eta_{21}(\boldsymbol{x}_t) > 0, \\ \text{or} \quad &\beta_{t1} > \beta_{t2} \text{ and } \eta_{21}(\boldsymbol{x}_t) < 0, \end{aligned}$$

i.e. when

$$\begin{aligned} &\beta_{t1} < \beta_{t2} \text{ and } \hat{p}_2(1|\boldsymbol{x}_t) > \hat{p}_2(2|\boldsymbol{x}_t), \\ \text{or} \quad &\beta_{t1} > \beta_{t2} \text{ and } \hat{p}_1(1|\boldsymbol{x}_t) > \hat{p}_1(2|\boldsymbol{x}_t). \end{aligned}$$

So if $\beta_{t1} > \beta_{t2}$, the combined classifier is equivalent to using only c_1, and if $\beta_{t2} > \beta_{t1}$ the combined classifier is equivalent to using only c_2.

Note that the conditions required by theorem 2 hold for the two definitions of $\eta_2(\boldsymbol{x}_t)$ recommended in section 2, namely

$$\eta_{2i}(\boldsymbol{x}_t) = \log\left(\frac{\hat{p}_i(2|\boldsymbol{x}_t)}{\hat{p}_i(1|\boldsymbol{x}_t)}\right), \text{ and}$$
$$\eta_{2i}(\boldsymbol{x}_t) = \hat{p}_i(2|\boldsymbol{x}_t) - \hat{p}_i(1|\boldsymbol{x}_t).$$

Corollary 1. *For all β_{t1} and β_{t2}, when using label-swapped component classifiers c_1 and c_2, the decision boundary of the combined classifier (6) is the same as the decision boundary of c_1 (and c_2).*

This follows directly from theorem 2.

Now suppose we combine M pairs of label-swapped classifiers and label them such that c_{2i} is the label-swapped partner of c_{2i-1}, for $i = 1, 2, \ldots, M$.

Theorem 3. *Using M pairs of label-swapped classifiers with parameters β_{t1}, $\beta_{t2}, \ldots, \beta_{t,2M}$ is equivalent to the model which uses only classifiers c_1, $c_3, \ldots, c_{2M-1}$ with parameters $\beta^*_{t1}, \beta^*_{t2}, \ldots, \beta^*_{tM}$, where*

$$\beta^*_{ti} \stackrel{\triangle}{=} \beta_{t,2i-1} - \beta_{t,2i}. \tag{18}$$

Proof. From (15),

$$\hat{p}_t(1|\boldsymbol{x}_t) > \hat{p}_t(2|\boldsymbol{x}_t) \tag{19}$$

when

$$\sum_{i=1}^{2M} \beta_{ti}\eta_{2i}(\boldsymbol{x}_t) < 0, \tag{20}$$

i.e. when

$$(\beta_{t1} - \beta_{t2})\eta_{21}(\boldsymbol{x}_t) + (\beta_{t3} - \beta_{t4})\eta_{23}(\boldsymbol{x}_t) + \ldots + (\beta_{t,2M-1} - \beta_{t,2M})\eta_{2(2M-1)}(\boldsymbol{x}_t) < 0$$

i.e. when

$$\sum_{i=1}^{M} \beta^*_{ti}\eta_{2(2i-1)}(\boldsymbol{x}_t) < 0, \tag{21}$$

where

$$\beta^*_{ti} = \beta_{t,2i-1} - \beta_{t,2i}. \tag{22}$$

Comparing (21) with (15), we can see this is equivalent to the classifier which combines $c_1, c_3, \ldots, c_{2M-1}$ with parameters $\beta^*_{t1}, \beta^*_{t2}, \ldots, \beta^*_{tM}$.

Importantly, although the β_{tj} may be restricted to taking non-negative values, in general the β^*_{ti} *can* take negative values. Hence we have shown that using label-swapped component classifiers and non-negative parameter estimates is equivalent to a classifier with unconstrained parameter estimates and which does not use label-swapped pairs. However, because in practice we must estimate the parameter values for a given set of component classifiers, using label-swapped classifiers with a non-negativity constraint will not necessarily give the same classification performance as using non-label-swapped classifiers with unconstrained parameter estimates. For example, LeBlanc and Tibshirani [7] and

Breiman [1] reported improved classification performance when constraining the parameters of the weighted average of classifier outputs to be non-negative. The benefit of the label-swapped approach is that it combines the flexibility of unconstrained parameters required for dynamic problems with the potential improved accuracy of parameter estimation obtained when imposing a non-negativity constraint. Clearly then the benefit of using label-swapped classifiers (if any) will be dependent on the algorithm used for parameter estimation.

It is important to note that using label-swapped classifiers leads us to requiring twice as many component classifiers and hence parameters as the corresponding model with unconstrained parameters. In addition, the additional computational effort involved in enforcing the non-negativity constraint means that the label-swapped approach is significantly more computationally intensive than using standard unconstrained estimates.

5 Example

In this section we demonstrate some of our results on an artificial dynamic classification problem. There exist two classes, class 1 and class 2, and observations from each class are distributed normally with a common covariance matrix. The probability that an observation is generated from class 1 is 0.7. At time $t = 0$ the mean of class 1 is $\mu_1 = (1, 1)$, and the mean of class 2 is $\mu_2 = (-1, -1)$. The mean of population 1 changes in equal increments from $(1, 1)$ to $(1, -4)$ over 1000 time steps, so that at time t, $\boldsymbol{\mu}_1 = (1, 1 - 0.005t)$. It is assumed that observations arrive independently without delay, and that after each classification is made the true label of the observation is revealed before the following observation arrives.

Before we can apply the classification model (1) to this problem, we need to decide on how many component classifiers to use, train the component classifiers, decide on the form of $\boldsymbol{\eta}_k(\boldsymbol{x}_t)$ and decide on an algorithm to estimate the parameter values $\boldsymbol{\beta}_t$ for every t. Clearly each one of these tasks requires careful thought in order to maximise the performance of the classifier. However, because this is not the subject of this paper, we omit the details behind our choices. For the following simulations we used three component classifiers, each of which was trained using linear discriminant analysis on an independent random sample of 10 observations from the population at time $t = 0$. Figure 2 shows the Bayes boundary at times $t = 0$, $t = 500$ and time $t = 1000$, along with the decision boundaries of the three component classifiers. We choose to use $\boldsymbol{\eta}_k(\boldsymbol{x}_t)$ defined by (3) (so for this example the decision boundary of the classifier is also linear), and used a particle filter approximation to the posterior distribution of $\boldsymbol{\beta}_t$ at each time step. The model for parameter evolution used was

$$\boldsymbol{\beta}_{t+1} = \boldsymbol{\beta}_t + \boldsymbol{\omega}_t, \tag{23}$$

where $\boldsymbol{\omega}_t$ has a normal distribution with mean $\mathbf{0}$ and covariance matrix equal to 0.005 times the identity matrix. 300 particles were used for the approximation. An observation $\boldsymbol{x}_t$ was classified as belonging to class k if

$$k = \text{argmax}_j \ \hat{E}_{\boldsymbol{\beta}_t}[\hat{p}_t(j|\boldsymbol{x}_t)]. \tag{24}$$

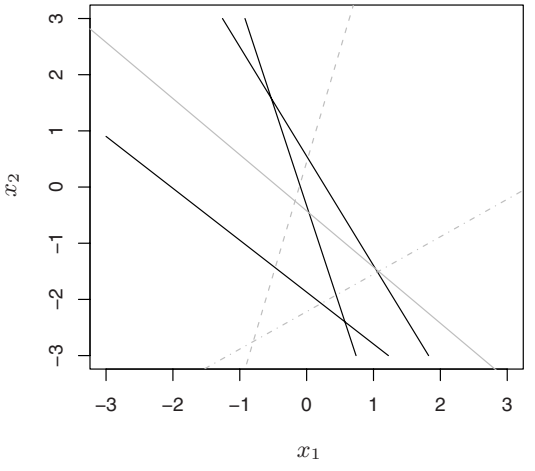


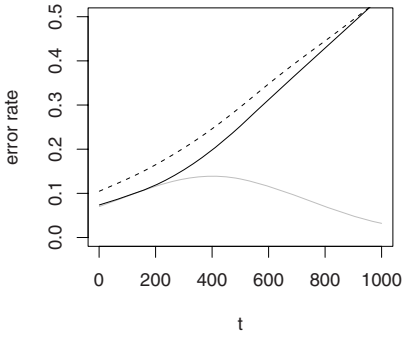
Fig. 2. The decision boundaries of the component classifiers (black) and the Bayes boundary (grey) at times $t = 0$ (solid), $t = 500$ (dashed) and $t = 1000$ (dot-dashed)

Denote by Err_{-i} the error rate of the ith component classifier on the training data of the other component classifiers, for $i = 1, 2, 3$. The value of β_{0i} was chosen to be proportional to $1 - \mathrm{Err}_{-i}$ for $i = 1, 2, 3$. Each simulation involved repeating the data generation, classification and updating procedure 100 times, and the errors of each run were averaged to produce an estimate of the error rate of the classifier at every time t.

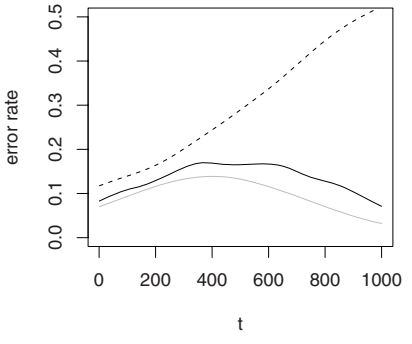
We repeated the simulation three times. In the first case, we constrained the parameters $\boldsymbol{\beta}_t$ to be non-negative. A smooth of the estimated average error rate is shown in figure 3(a), along with the Bayes error (grey line) and the error of the the component classifier corresponding to the smallest value of Err_{-i} (included to demonstrate the deterioration in performance of the "best" component classifier at time $t = 0$, dashed line). The error rate of the classifier is reasonably close to the Bayes error for the first 200 updates, but then the performance deteriorates. After $t = 200$, the Bayes boundary has moved enough that it can no longer be well approximated by a linear decision boundary lying in the region of disagreement.

In the second case, we use the same three classifiers as above, but include their label-swapped pairs. We hence have six component classifiers in total. A smooth of the estimated error rate is shown in figure 3(b), and it is clear that this classifier does not succumb to the same level of decreased performance as seen in figure 3(a).

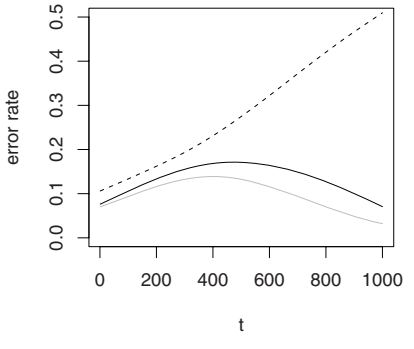
Thirdly, we used the same set of three component classifiers but without constraining the parameter values to be non-negative. The resulting error rate, shown in figure 3(c), is very similar to that using label-swapped classifiers.



(a) Non-negative parameter values

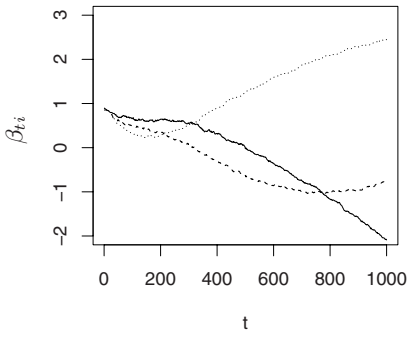


(b) Label-swapped classifiers

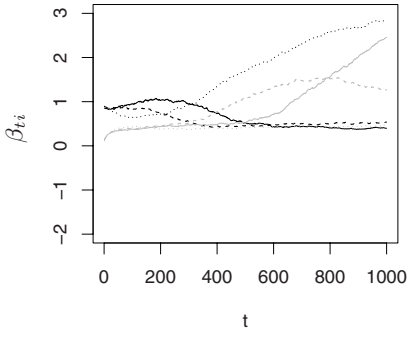


(c) Unconstrained parameter values

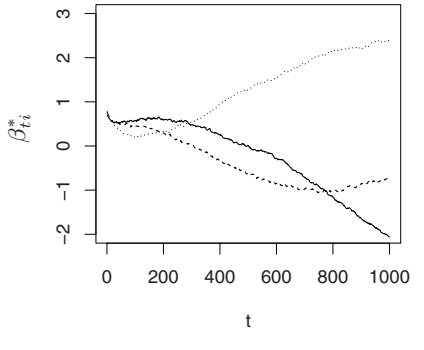
Fig. 3. Smoothed average error rates for the three scenarios described (solid black line) along with the Bayes error (grey) and error rate of the component classifier with the lowest estimated error rate at time $t = 0$ (dashed black line)



(a) Unconstrained parameter values



(b) Label-swapped classifiers: $\boldsymbol{\beta}_t$



(c) Label-swapped classifiers: $\boldsymbol{\beta}^*_t$

Fig. 4. Average expected parameter values β_i, for $i = 1$ (solid line), $i = 2$ (dashed) and $i = 3$ (dotted). In figure 4(b) the grey and black lines of each type correspond to a label-swapped pair.

In figure 4, we show the average expected parameter values returned by the updating algorithm in cases two and three. Clearly the values of $\boldsymbol{\beta}_{ti}^{*}$ shown in figure 4(c) are very similar to the unconstrained parameter values in figure 4(a), which explains the similarity of classification performance between the label-swapped and unconstrained cases. Furthermore, we can see from figure 4 that negative parameter values become necessary after about 200 updates, again explaining the behaviour seen in figure 3(a).

This example shows that it is important to consider the region of disagreement in dynamic classification problems. Furthermore, we found no clear difference in performance between the classifier using label-swapped component classifiers with non-negative parameter values, and the classifier using unconstrained parameter estimates.

6 Conclusions

When using a combining model of the form (1) or a linear combiner with non-negative parameter values, it can be useful to consider the region of disagreement of the component classifiers. This becomes of even greater relevance when the population is believed to be dynamic, as the region of disagreement is a bound on the region in which the decision boundary of the classifier can lie. If the Bayes boundary lies outside the region of disagreement, then it is unlikely that the classifier will perform well. In stationary problems it may be beneficial to constrain the region of disagreement. However, in dynamic scenarios when the Bayes boundary is subject to possibly large movement, it seems most sensible to maximise this region. This can be done for a two-class problem by using label-swapped classifiers with non-negative parameter estimates, or more simply and efficiently by allowing negative parameter values. Which of these approaches results in better classification performance is likely to depend on the parameter estimation algorithm, and should be further investigated.

References

1. Breiman, L.: Stacked Regressions. Machine Learning 24, 49–64 (1996)
2. Fumera, G., Roli, F.: Performance Analysis and Comparison of Linear Combiners for Classifier Fusion. In: Caelli, T.M., Amin, A., Duin, R.P.W., Kamel, M.S., de Ridder, D. (eds.) SPR 2002 and SSPR 2002. LNCS, vol. 2396, pp. 424–432. Springer, Heidelberg (2002)
3. Fumera, G., Roli, F.: A Theoretical and Experimental Analysis of Linear Combiners for Multiple Classifier Systems. IEEE Trans. Pattern Anal. Mach. Intell. 27(6), 942–956 (2005)
4. Kelly, M., Hand, D., Adams, N.: The Impact of Changing Populations on Classifier Performance. In: KDD 1999: Proc. 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, California, United States, pp. 367–371. ACM, New York (1999)
5. Kuncheva, L.I.: Classifier Ensembles for Changing Environments. In: Roli, F., Kittler, J., Windeatt, T. (eds.) MCS 2004. LNCS, vol. 3077, pp. 1–15. Springer, Heidelberg (2004)

6. Kuncheva, L.I., Whitaker, C.J.: Measures of Diversity in Classifier Ensembles and Their Relationship with the Ensemble Accuracy. Machine Learning 51, 181–207 (2003)
7. Le Blanc, M., Tibshirani, R.: Combining Estimates in Regression and Classification. Technical Report 9318, Dept. of Statistics, Univ. of Toronto (1993)
8. Tumer, K., Ghosh, J.: Analysis of Decision Boundaries in Linearly Combined Neural Classifiers. Pattern Recognition 29, 341–348 (1996)