Improved Uniformity Enforcement in Stochastic Discrimination

Matthew Prior and Terry Windeatt

Centre for Vision Speech and Signal Processing

University of Surrey, Guildford, Surrey, GU2 7XH, UK

{*m.prior,t.windeatt*}*@surrey.ac.uk*

Abstract

There are a variety of methods for inducing predictive systems from observed data. Many of these methods fall into the field of study of machine learning. Some of the most effective algorithms in this domain succeed by combining a number of distinct predictive elements to form what can be described as a type of committee. Well known examples of such algorithms are AdaBoost, bagging and random forests. Stochastic discrimination is a committee-forming algorithm that attempts to combine a large number of relatively simple predictive elements in an effort to achieve a high degree of accuracy. A key element of the success of this technique is that its coverage of the observed feature space should be uniform in nature. We introduce a new uniformity enforcement method, which on benchmark datasets, leads to greater predictive efficiency than the currently published method.

1 Introduction

There are many techniques available for inducing predictive algorithms from observed data. Those methods that use a combination of classifiers, called a committee or ensemble, such as AdaBoost[5], bagging[2] and random forests[7] have demonstrated very good performance on real-world problems. Stochastic discrimination is an alternative method of constructing committees of classifiers. It has a sound theoretical basis and is robust to sources of over-fitting, other than those attributable to small sample size effects[4, 8, 10]. It intrinsically deals with two class problems but can be extended to multi-class problems by the use of such techniques as one-versus-all and error correcting output coding[11] decompositions.

One of the principle differences between conventional ensemble methods, such as bagging, and stochastic discrimination is that in conventional ensembles

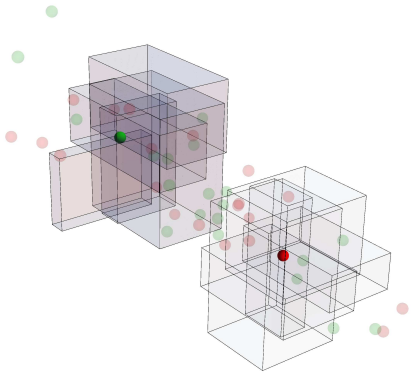

Figure 1: A collection of 10 rectangular parallelepiped thick models centred around two data instances from the training set. Additional instances from the two classes are shown and they are all embedded in a three dimensional feature space.

each individual classifier is normally expert, to some degree, on the whole data space. In a stochastic discrimination ensemble this is not the case[9]. The set of weak classifiers in a stochastic discrimination ensemble may view the data space in a uniform fashion but individual classifiers may not, and in general will not, do this. More specifically, they will only consider a limited subspace of the feature space, with each dimension in the feature space having a degree of coverage selected at random.

The extent to which a stochastic discrimination ensemble views the feature space without unduly favouring one region over another is known as its uniformity. The method of uniformity enforcement is one of the key elements of an implementation of stochastic discrimination. Uniformity ensures that the ensemble as a whole can generalise effectively over the full extent of the feature space.

The implementation of stochastic discrimination described in detail in [9, 12] uses a method of uniformity enforcement that is based on measurements relating to the average coverage for elements of the ensemble predicting a specific class. We propose an alternative uniformity enforcement scheme based on the minimum instance coverage.

2 Stochastic Discrimination

Typically classifier combination methods such as AdaBoost, bagging and random forests seek to merge base classifiers that have knowledge of the full extent of the feature space via the training set. Stochastic discrimination differs in its approach to combining weak base classifiers, which it refers to as thick models, in that it seeks to assemble elements that cover subsets of the training data. These are drawn from embedded subspaces of a finite n dimensional space, $F \in \mathbb{R}^n$. These subspaces are constructed from a geometric model centered on an instance of the training set and which form an n dimensional rectangular parallelepiped.

The coverage of the parallelepiped in each feature dimension is a random proportion of the feature extent. This sub-sampling of the feature space is one of the methods responsible for ensuring diversity in the produced population of thick models and additionally acts as a regularisation mechanism to alleviate the potential for over-fitting.

A stream of thick models is generated by randomly selecting an instance from the training set and generating a geometric model around it. This stream is then thinned according to each model's ability to discriminate between instances of classes within its embedded subspace of the feature domain. Suitably discriminating models undergo further selection based on the existing coverage of the feature domain. The underlying theory of stochastic discrimination[9] requires that coverage should be uniform to ensure that there is no bias towards particular areas of the feature space. This implies that the number of thick models capturing each instance in the training set should be equal.

In the context of a two class classification problem in which feature vectors, q, are drawn from two classes, $\{1, 2\}$, embedded in feature space F, instances from the available dataset are randomly partitioned into a training and test set, $\{TR, TE\}$. TR is further partitioned into instances from class 1 and class 2, $\{TR_1, TR_2\}$. Stochastic discrimination creates a stream of models by randomly sampling instances from TR and builds a space-enveloping thick model around them. The thick model, m, is constructed from random proportions of the feature extent in each of the n dimensions of the feature space, as depicted in Figure 1.

It is worthwhile observing that the generalisation ability of the stochastic discrimination algorithm is a function sensitive to a number of factors

$$\begin{aligned} SD_{Acc} &= f(Model_{Number}, Enrichment_{Degree}, \\ &\quad Coverage_{Uniformity}, Model_{Size}). \end{aligned}$$

It is resistant to overtraining and in general the more models in the ensemble the higher the accuracy. Additionally there is a strong relationship between the number and distribution of instances in TR and the number of thick models required to adequately capture their distribution. Stochastic discrimination also relies on the assumption, which is a requirement for other machine learning

Algorithm 1 Stochastic discrimination thick model stream production algorithm, P.

```
Do
 Generate a Thick Model from a random instance in TR_1
 If ( Enriched( Thick Model ) )
     If( ImprovesUniformity( Thick Model ) )
           Accept( Thick Model )
Until ( Enough( Thick Models ) )
```

algorithms, that there is a projectability between the distribution of samples in the training set and the test set.

2.1 Enrichment

For the stream of stochastically generated thick models to be useful for the task of classification it is necessary that they possess some discriminative power to separate the classes. To this end the thick models are selectively filtered based on their enrichment. Enrichment is calculated from the proportion of instances for each of the classes in the training set which are captured by the thick model, m. If the proportion of instances of class 1 captured from TR_1 is greater than those of class 2 that are captured from TR_2, then the model is considered enriched with respect to class 1.

$$\frac{\mid m \subset TR_1 \mid}{\mid TR_1 \mid} > \frac{\mid m \subset TR_2 \mid}{\mid TR_2 \mid}. \tag{1}$$

2.2 Uniformity Enforcement Strategies

In the standard version of stochastic discrimination[9] each thick model subset, m, that satisfies the enrichment criteria is further subjected to a uniformity forcing step as indicated in Algorithm1 and referred as algorithm P. Uniformity is enforced via the measurement of coverage. The coverage of a data instance, c_q, is defined as the number of thick models that include the data instance, q, within their volume, divided by the size of the set of thick models, M, produced so far, $\mid M \mid$. Thus,

$$c_q = \frac{\mid \{m \in M : q \in m\} \mid}{\mid M \mid}. \tag{2}$$

A thick model, m, is considered an acceptable candidate for the ensemble of thick models if it is enriched and its average coverage for points captured from TR_1, c_{TR_1}, is below the average cover for all points in TR_1, $\bar{C}_{TR_1}$.

$$(\frac{1}{|\{q \in TR_1 : q \in m\}|} \sum_{\{q \in TR_1 : q \in m\}} c_q) < \bar{C}_{TR_1}. \tag{3}$$

Algorithm 2 Stochastic discrimination thick model stream production algorithm, L.

```
Do
 Generate a Thick Model using least covered instance in TR_1
 If ( Enriched( Thick Model ) )
           Accept( Thick Model )
Until ( Enough( Thick Models ) )
```

In effect this filters models from the stream that favour instances which are under-represented in the current thick model working set, M.

We propose an alternative strategy to achieve uniformity. This entails selecting the least covered instance in TR_1. By choosing the instance that has minimum coverage as the basis for the new thick model, we aggressively focus on the area in the feature space that instantaneously exhibits the least coverage and ensure that it is increased. If there are ties for the least covered instance then these can be broken randomly or, as an enhancement, an instance from particularly ill represented region can be searched for.

$$\{q : \arg \min_{q \in TR_1} c_q\}. \tag{4}$$

Our modified algorithm, described in Algorithm 2 and referred to as L, seeks to improve the mean coverage and improve the variance of the coverage array by decreasing the contribution from the largest reducible component. If the minimum value in the coverage array, C, is not unique then one of the corresponding instances is selected at random.

To quantify the degree of uniformity, D, present in the thick model streams we calculate the standard deviation of the instance wise coverage.

$$D = \sqrt{\frac{1}{|TR|} \sum_{q \in TR} (c_q - \bar{C})^2}, \tag{5}$$

where $\overline{C}$ is the mean value of the coverage set. The minimum value for D is zero and this indicates that all instances in the training set have equivalent coverage, higher values of D represent increasingly poor levels of uniformity.

2.3 Discriminant

Once a thick model set of the desired size has been formed, a discriminant function can be used to classify instances. The discriminant function uses the difference in probabilities of capture by the thick models in M to assign class membership. In the case where the models have been enriched for TR_1, an unknown instance, x, will be captured by a larger number of thick models if it is of class 1 than were it of class 2. A suitable threshold can be chosen to optimise the classification accuracy of the discriminant function.

3 Theoretical Exploration

Under the assumption that variance is a valid measurement of uniformity we examine the behaviour of the classic uniformity algorithm P as defined in section 2.2 and Algorithm 1. and L Algorithm 2. These algorithms have simplified to highlight the essential differences between the two approaches. For more detailed implementation information see [9].

The coverage values, C, form a set of positive integers in the range from 0 to the size of the thick model set , $\mid M \mid$. Considering the limiting case in which a thick model captures only one point from TR, the addition of this thick model to M will result in a unity increment of a single value within C. The maximum reduction to the variance of C will be achieved if that point is the one that is the furthest below the mean value of the coverage set, $\overline{C}$, known as c_{min}. Ties in the value of c_{min} should be broken randomly.

By considering the contribution made to the change in variance by incrementing either c_{min} or another arbitrary member of C with a value larger than c_{min} we can show that

$$\frac{((k+1)-\overline{C})^2+(l-\overline{C})^2}{\mid M \mid} \leq \frac{((l+1)-\overline{C})^2+(k-\overline{C})^2}{\mid M \mid}, \tag{6}$$

where k, l are positive real integers representing values within C and with $k \leq l$. It follows that the reduction in variance , and hence increase in the uniformity, will be greatest if k is the minimum value in C. The change to the mean value of C is constant in this limiting case. Thus for the special case where the thick model covers only a single instance in TR, algorithm L should always improve uniformity by at least as much as algorithm P.

When the thick model covers more than one point, the analysis of performance is more complicated. The degree of improvement of uniformity and the mean value of cover,$\overline{C}$, will depend on the specific sample of instances captured by the thick model, m. At the limit, where all points in TR_1 are captured by m, there will no difference in the change in uniformity and $\overline{C}$ between algorithms P and L. Where m only captures a percentage of TR_1 and if points are chosen at random, the expectation will be that the reduction in variance from algorithm L will always exceed or equal that from algorithm P. But this ignores the contribution from the average cover related enforcement strategy employed by algorithm P, which will undoubtedly improve the situation over a purely random selection. Furthermore, the performance will be dependent on the exact distribution of the dataset under consideration.

However, each new thick model under algorithm L will always contain the most beneficial point, c_{min}, whilst under algorithm P, m will only have some probability of capturing c_{min}. This probability will be dependent on the size of TR_1, the amount of the feature space that m captures and the distribution of instances in the feature space. Our experimental results suggest that on average algorithm L is more effective.

4 Experiment Details

Experiments were performed on twenty datasets, eighteen datasets from the UCI Machine Learning Repository [1] and 2 synthetic ones from [3]. These contained a mixture of binary and multi-class problems. Multi-class problems are handled using a one-versus-all decomposition strategy. To estimate the generalisation error of the induced classifiers, ten repetitions of ten-fold cross validation were performed for each dataset within a WEKA[6] framework. Identical trials were performed for the standard uniformity enforcement algorithm, described in [9], based on mean class coverages and our method of uniformity enforcement using the least covered point. The number of thick models used for each classifier was fixed at 3001. The minimum allowable thick model size was adjusted between 0.01 and 0.5 percent of the feature space. The following datasets were used. Balance[BAL], credit-a[CRA], diabetes[DIA], ecoli[ECO], glass[GLA], heart[HRT], hepatitis[HEP], ionosphere[ION], iris[IRS], labor[LAB], lymph[LYM], parkinsons[PAR], satellite[SAT], segment[SEG], sonar[SON], vehicle[VEH], vowel[VOW], Wisconsin breast cancer[WIS], twonorm[2NM] and threenorm[3NM].

5 Experimental Results

We present individual experimental results for a selection of the datasets in Figure 2. These show the test error rates and normalised standard deviation of the coverage, D, plotted against minimum model size for algorithms L and P. From Figure 2. it is not easy to determine a direct relationship between test accuracy and D, the trend, except in the case of WIS, is that lower coverage deviation leads to lower test error. The averaged values Figure 3. support this view. The minimum error rates in Table 1. confirm that neither algorithm is superior on all datasets. Where L performs worse than P then the difference is generally small, as is the case with datasets DIA, HEP and SAT. Where algorithm L exceeds the performance of P the difference can be significant, as is the case with datasets GLA, IRS, LYM, SEG, VEH, VOW and the exceptions being PAR, 3NM.

Figure 3. contains averaged results over all datasets for test error and average cover on the left and normalised thick model retry rates and the normalised standard deviation of the coverage on the right. The averaged graphs give a clearer indication of the relative performance of the two algorithms. L consistently outperforms P in terms of test accuracy across all minimum thick model sizes and also for absolute coverage values. This implies that L is building larger thick models that capture more points and will tend to generalise better.

The right of Figure 3. shows that the averaged normalised standard deviation, D, is consistently better for L across all model sizes. It also shows that the stream production efficiency for L, measured by the number of retries required to find a suitable model, can be as little as half the value of P.

Table 1. shows the minimum value of the test error for each of the twenty

	BAL	CRA	DIA	ECO	GLA	HRT	HEP	ION	IRS	LAB
L	.10	.03	.24	.14	.29	.17	.23	.09	.04	.19
P	.10	.04	.23	.16	.42	.17	.21	.10	.11	.19

	LYM	PAR	SAT	SEG	SON	VEH	VOW	WIS	2NM	3NM
L	.35	0.17	.12	.09	.12	.29	.14	.03	.03	.16
P	.47	0.13	.11	.16	.13	.32	.20	.04	.03	.14

Mean test error L	0.151
Mean test error P	0.173

Table 1: Minimum test error rates for algorithms L and P and mean values for L and P.

datasets for uniformity enforcement algorithms L and P. Averaging over all datasets, algorithm P has a minimum test error that is 15 percent worse than L. Subjecting these results to a paired T test rejects the null hypothesis at a significance level of 0.05.

6 Conclusion

The strategy of uniform coverage enforcement is an important element of the stochastic discrimination method. Our experiments indicate that simply selecting the least covered instance in the training set is an effective alternative to the standard method of choosing a random instance and then checking for its effect on coverage. Though it is not certain for any particular dataset which strategy will be most effective, over a range of datasets, algorithm L achieves better accuracy, more uniform coverage, larger thick models and a lower retry rate than algorithm P. Finally, we would like to thank the reviewers for their helpful comments.

References

[1] A. Asuncion and D.J. Newman. UCI machine learning repository, 2007.

[2] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

[3] L. Breiman. Bias, variance, and arcing classifiers, 1996.

[4] D. Chen, P. Huang, and X. Cheng. A concrete statistical realization of kleinberg's stochastic discrimination for pattern recognition, part i. two-class classification. *Annals of Statistics*, 31(5):1393–1412, 2003.

[5] Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. In *International Conference on Machine Learning*, pages 148–156, 1996.

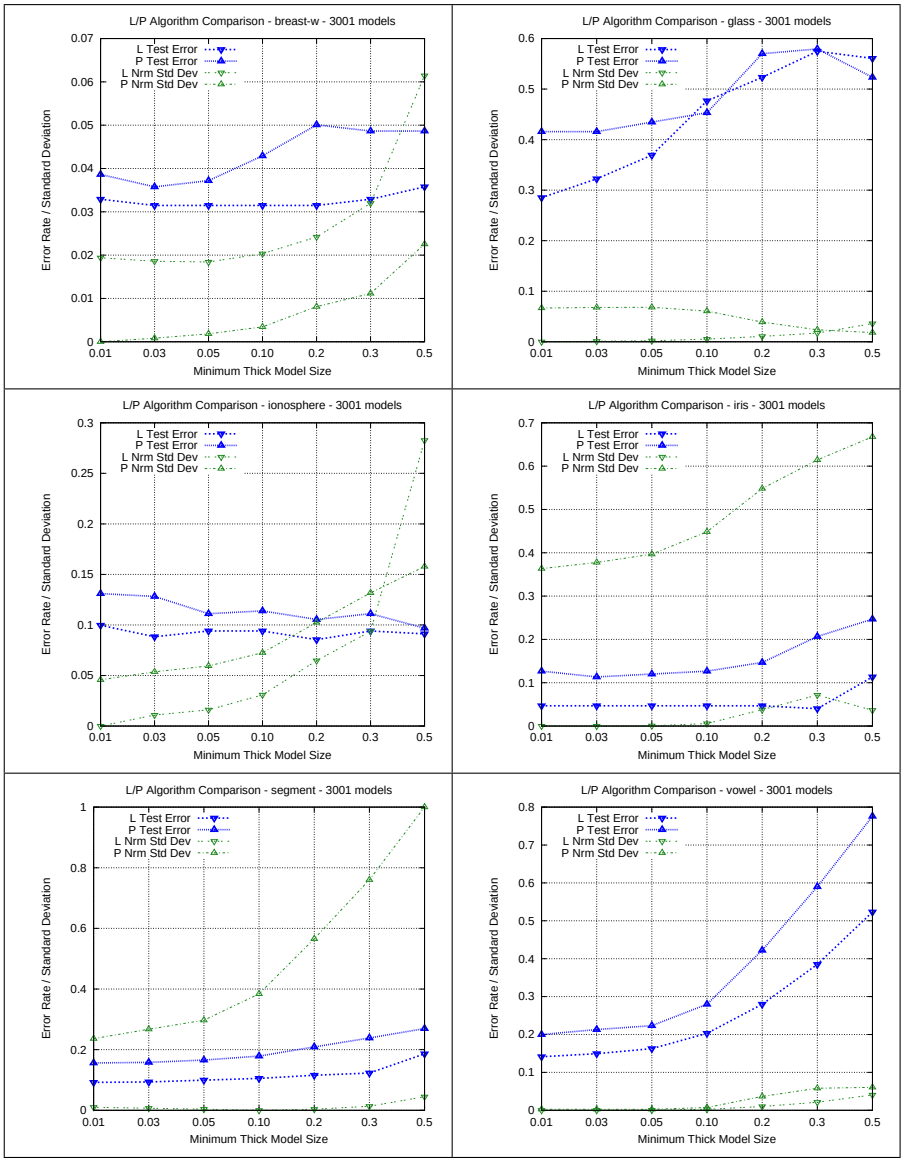

Figure 2: Test error versus minimum thick model size and normalised standard deviation of coverage for uniformity enforcement algorithms L and P.

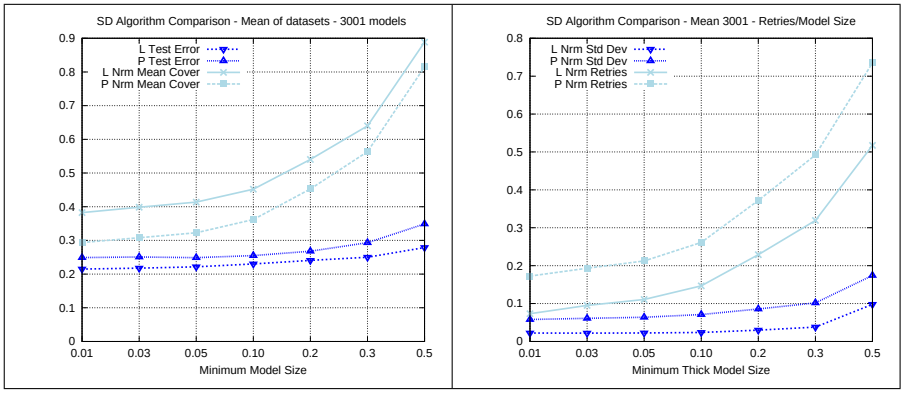

Figure 3: Mean test error rate and normalised coverage over all datasets (left) and mean normalised retries and normalised standard deviation of coverage, D, (right).

[6] Stephen R. Garner. Weka: The waikato environment for knowledge analysis. In *In Proc. of the New Zealand Computer Science Research Students Conference*, pages 57–64, 1995.

[7] T.K. Ho. The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):832–844, 1998.

[8] E. M. Kleinberg. Stochastic discrimination. *Annals of Mathematics and Artificial Intelligence*, 1, 1990.

[9] E. M. Kleinberg. On the algorithmic implementation of stochastic discrimination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(5):473–490, 2000.

[10] E. M. Kleinberg and T. K. Ho. Pattern recognition by stochastic modeling. In *Proceedings of the Third International Workshop on Frontiers in Handwriting Recognition*, pages 175–183. Partners Press, 1993.

[11] M. Prior and T. Windeatt. Over-fitting in ensembles of neural network classifiers within ecoc frameworks. In *Proc. Int. Workshop on Multiple Classifier Systems (LNCS 3541)*, pages 286–295, 2005.

[12] M. Prior and T. Windeatt. Parameter tuning using the out-of-bootstrap generalisation error estimate for stochastic discrimination and random forests. In *International Conference on Pattern Recognition*, pages 498–501, 2006.