Pattern Recognition by Stochastic Modeling

Eugene M. Kleinberg

Department of Mathematics
State University of New York at Buffalo
Buffalo, NY 14214, USA

Tin Kam Ho

AT&T Bell Laboratories [1]
600 Mountain Avenue, Murray Hill, NJ 07974

Abstract

We describe a method first proposed in [4] for constructing a classifier by combining a large number of weak classifiers. Each weak classifier is identified with its decision regions that are point sets in a feature space. Such point sets can be generated by a stochastic process, and selected according to criteria that guarantee some minimum discrimination performance and uniform coverage of the space. It is shown that the combined power of a large collection of such weak classifiers can be improved continuously. In an application to handwritten digit recognition, the method achieved an accuracy comparable to that of expensive nearest-neighbor matching.

1 Introduction

We present a method first proposed in [4] for constructing a generic classifier for an arbitrary multi-class pattern recognition problem. The mathematics underlying this method is given in [4] and [5]. The analysis suggests that by combining a large number of stochastically generated discriminators, a classifier can be made arbitrarily close to perfection. It also provides an approach to study the effects of the representativeness of training samples, the discriminating power of features, and correlations of features on classification accuracy. In this paper, we describe the method as a computational procedure, and present our experimental results in an application to the recognition of handwritten digits.

From a geometrical point of view, a classifier is a partition of a given feature space F into a set of non-overlapping *decision regions* each associated with a class. A given point in F is said to be in class c if it falls within any one of the regions associated with class c. A classifier is said to be perfect if all samples of class c lie in the regions of class c that it computes. Depending on the nature of a given problem, the true decision regions may take an arbitrary shape. We make no assumptions on the shapes of the true decision regions.

This approach is different from many classical methods where a classifier is represented by a set of *decision surfaces*, i.e., boundaries of the decision regions, which are

[1] This work was completed while Tin Kam Ho was with CEDAR, State University of New York at Buffalo.

approximated by polynomials obtained by discriminant analysis or sequential learning procedures. Our method focuses on approximating the interiors of the decision regions instead.

Since each classifier can be represented as a partition of the feature space, the number of all possible classifiers equals the number of all possible partitions of the space. Our method is based on sampling such a population of classifiers. We consider two classifiers to be equivalent as long as their decision regions are identical. Note that here we study the classifiers solely by their decisions, regardless of how these classifiers are obtained.

Given an algorithm which would enumerate all the possible partitions of a feature space, it is clear that a classifier with perfect performance can be obtained eventually. Obviously, given the large number of possible partitions, the probability of obtaining a perfect one in a small number of trials is extremely low. However, by an analysis of the expected performance of an arbitrary classifier in such an enumeration, it can be shown that within a small number of trials, a classifier of some minimum performance can be obtained. The essence of our method is in accumulating a large but feasible number of such nearly trivial classifiers, so that the accuracy of their combined decisions can be made close to perfection [4].

We assume that an n-class pattern recognition problem is given in a real feature space F of l dimensions. We also assume that a set of *training samples* is available. The samples are vectors in F and are labeled with their true classes. The performance of potential solutions is evaluated with a set of *testing samples*, which are also vectors in F.

We call a decision region corresponding to a random partition of the feature space a *weak model*. The discriminating power of each weak model is evaluated against the set of *training samples*. Only those models that satisfy certain conditions are retained. One condition is that the model should classify the training samples of a certain class to a minimum degree of correctness. Such models are said to be *enriched*. We will need sufficient weak models to cover every part of the feature space where training samples are present, so that there is always a decision for each input. Moreover, the models should be sufficiently different from one another for their combination to be effective. These conditions are summarized in a property that we will refer to as *uniformity*.

Our method consists of three major procedures:

1. Generation of the weak models.
2. Selection of the weak models according to enrichment and uniformity criteria.
3. Combination of the selected weak models.

We will discuss the details of each procedure in the following sections. Section 2 discusses issues involved in the generation of the weak models. Section 3 describes the essential criteria for selecting the weak models. Section 4 discusses the combination of the selected weak models and the convergence of their combined performance. Section 5 presents some experimental results in an application to handwritten digit recognition.

2 Sampling the Population of Weak Models

The difficulties present in typical pattern recognition problems are often two-fold: (1) drawing inferences from limited training samples to unseen ones; and (2) obtaining accurate discrimination functions which are efficient in both space and time.

We construct a classifier by collecting a set of weak models. A weak model is a set of points in the feature space. In a discrete space of l dimensions each having a distinct levels, there are a^l-many points in the space, and 2^{a^l}-many subsets of the space. In a continuous space, there are uncountably many possible sets of points. Each of these sets is a candidate of a weak model. An exhaustive enumeration is usually infeasible, and samples from such a population are used instead. In order that the above mentioned difficulties are addressed, two guidelines are observed in drawing the samples:

(1) *projectability*: A weak model should be able to capture more than one point so that the solution can be projectable to points not included in the training set. Geometrically, this means that a useful model must be of certain minimum size, and it should be able to capture points that are considered as *neighbors* of each other.

(2) *simplicity of representation*: A weak model should have a simple representation. That means, the characteristic function of the model must be cheaply computable. To illustrate this, consider representing a model as a listing of all the points it contains. This is unacceptable, since the resultant solution could be as expensive as an exhaustive template matching using all the points in the feature space. An example of a model with a simple representation is a half-plane in a two-dimensional feature space.

Note that for a solution to be projectable to unseen testing patterns, one needs an assumption that the training patterns are *representative* of the unseen ones. Geometrically it means that there must be a training sample in a *neighborhood* of every unseen sample. Here the definition of the neighborhood can be arbitrary, as long as there is a way to construct a weak model that contains all the points within such a neighborhood. Since our only knowledge of the configuration of the true decision regions is from the given training set, there is no basis for any inference on the part of the feature space where no representative training samples are given.

Conditions (1) and (2) restrict the type of weak models yet by no means reduce the number of candidates to any tangible limit. To obtain an unbiased collection of the candidates with minimum effort, random sampling without replacement is considered useful. The training of the method thus relies on a stochastic process which, at each iteration, generates a weak model that satisfies the above conditions.

3 Enrichment and Uniformity

Consider an arbitrary partition of points into n disjoint sets in a space F. If this partition is created without access to the training set, it is hardly likely that the partition will match the training set exactly. On the other hand, it is also very unlikely that each of the n sets will capture the same number of points of each class. The most common case

will be that some of the sets capture points of some classes more than those of other classes. Our method takes advantage of such commonly seen unevenness in a random partition.

We can now define the concept of enrichment: if a set captures points of a particular class more than points of other classes, we call that set an *enriched* weak model with respect to that class. Given any two classes i and j, the degree of enrichment of a model m with respect to class i versus class j is described by a rating

$$d_{ij}(m) = r_i(m) - r_j(m),$$

where

$$r_i(m) = \frac{|m \cap TR_i|}{|TR_i|},$$

$$r_j(m) = \frac{|m \cap TR_j|}{|TR_j|},$$

TR_i and TR_j are the training sets for classes i and j, respectively, and $|A|$ denotes the cardinality of a set A.

At a minimum, we require that $|d_{i,j}| > 0$, that is, the models should be *nontrivial* in the sense that they are enriched with respect to either class i or class j. A threshold on the rating d_{ij} can be used to select the weak models from the random stream so that they are of some minimum quality. There is a tradeoff between the number of models tried and the required minimum rating, since models of higher ratings are less likely to appear. Using a higher threshold, one will be able to produce a smaller collection of models that yield a classifier of the same level of accuracy. However, more trials are needed to obtain such a collection.

The rating r_i measures how well a model is able to capture points of class i, but it tells nothing about which points are being captured. If all the models are capturing the same set of points again and again, even though we have a collection of models of high ratings, little has been achieved for the points that are not covered by any models. Therefore in addition to the enrichment criterion, another condition that is referred to as *uniformity* has to be satisfied by the collection of models. Intuitively, we require that the selected models must cover all parts of the feature space where training samples are given. Moreover, we require that the set of selected models should be *unbiased* for any particular point, in other words, the number of models of a particular rating covering each point of a given class in the training set should be the same. This uniformity condition is critical in the proof of convergence of the combined accuracy [5]. Given the need for projectability of the models, strict uniformity is often difficult to obtain. However, empirical evidence shows that satisfactory results can be obtained with approximate uniformity, a fact which is also supported by theoretical study [2].

4 Combining the Weak Models

The weak models with their r_i ratings are combined to give a set of discriminant functions. Let q be a point in the feature space F, and m be a weak model from a given collection

M. Given any two classes i and j, a random variable is defined as follows:

$$X_{ij}(q,m) = \begin{cases} 2\,(\frac{C_m(q)-r_j(m)}{r_i(m)-r_j(m)}) - 1 & \text{if } r_i(m) \neq r_j(m), \\ 0 & \text{if } r_i(m) = r_j(m). \end{cases}$$

where $C_m(q)$ is the characteristic function of m evaluated at q. ($C_m(q)$ is 1 if $q \in m$, and 0 if $q \notin m$.) The variable $X_{ij}(q,m)$ has a value for each point q and each model m. It can be shown [5] that, as long as the enrichment and uniformity conditions are satisfied, for each point q, the expectation of X_{ij} is 1 if q is a member of class i, and is -1 if q is a member of class j; furthermore, if for each point q we let

$$Y_{ij}(q) = \frac{1}{p_{ij}} \sum_{k=1,\ldots,p_{ij}} X_{ij}(q,m_k),$$

where p_{ij} is the number of models accumulated for the class pair i and j, then the expectation of Y_{ij} is 1 if q is a member of class i, is -1 if q is a member of class j, and the variance of Y_{ij} approaches 0 as the number of models p_{ij} increases.

An n-class ($n > 2$) problem is solved by decomposing the problem into $n(n-1)$ pairwise discrimination problems. For each point q, and each class i, let

$$Z_i(q) = \sum_{j=1,\ldots,n,j\neq i} Y_{ij}(q).$$

$Z_i(q)$ can then be used as a discriminant function for class i. A point q is assigned to class i for which $Z_i(q)$ is the maximum.

The complexity of the resultant classifier is determined by the number of weak models used. Given the ability to improve performance by accumulating additional weak models, the classifier's power can be continuously enhanced.

5 Experiments on Handwritten Digits

The method was applied to a set of handwritten digit images from the NIST (National Institute of Standards and Technology) Special Database 3. The database contains digit and character images extracted from a set of forms designed to collect samples of handwriting. The forms were scanned at a resolution of 300 pixels per inch and the images were binarized.

We derived three feature vectors to represent each digit. Each digit is first size-normalized to 16x16 pixels. In the process of size-normalization, the number of black pixels is counted for each of the windows in the original image that correspond to one pixel in the normalized image. The ratio of the count versus the window size is scaled to the range [0, 255] and is taken as a pseudo-gray level value for the destination pixel. These pseudo-gray level values are the 256 numeric fields in the first type of vector we used to represent the digit. The pseudo-gray values are then converted to binary values by comparison against a threshold. These binary pixel values are the 256 numeric fields

in the second type of vector we used to represent the digit. Finally, the binary image is convolved with 32 structural feature masks defined on 7x7 windows [1]. The convolution responses are thresholded and counted in each of the 16 4x4 nonoverlapping regions in the binary image. These counts then comprise the 512 numeric fields in the third type of vector we used to represent the digit.

A classifier is derived using each of these three feature vector representations. Each classifier consists of a set of weak models generated from the feature space defined by the corresponding vectors. Each weak model in this implementation is a subset of the feature space composed from intersections of small numbers of half-spaces. Each classifier produces a ranking of the 10 classes (0-9) for each digit image as output.

The database is organized by the serial number of the forms from which the digit images were extracted. The first sample of each of the ten digits in each form was used. Samples from all the 2,000 forms were divided sequentially into four data sets. Set 1 was used as the training set for stochastic modeling. Each obtained classifier was tested on sets 2, 3 and 4. To determine the correlation of the classification results using different features, the rankings from the three classifiers are combined by a logistic regression model [3]. Classification results on set 2 were used to estimate the regression parameters. The regression model was then applied to sets 3 and 4. The size and use of each data set are summarized in Table 1.

Table 1: Sizes and Uses of Data Sets

Data Set	Form Serial No.	Number of Samples	Used for
Set 1	0000-0499	4997	evaluating the weak models
Set 2	0500-0999	4975	testing the classifiers and estimating regression parameters
Set 3	1000-1499	4966	testing the classifiers and the regression model
Set 4	1500-1999	4876	testing the classifiers and the regression model

The accuracy of the classifiers improves as training continues and the collection of models grows. Taking the classifier using pseudo-gray values as an example, the increases in correct rate on both the training (set 1) and testing (set 2) sets as the models accumulate are shown in Figure 1. Note that there is little tradeoff between the accuracy on the training set and the generalization power[2].

Table 2 shows the results of classification using each of the three feature sets. For each feature set, the training of the classifier was stopped first at 250 and then at 750 enriched weak models for each class pair. The classifiers obtained at these two levels of training were tested on the data sets separately. The results were compared to those of a nearest-neighbor classifier (1NN) and a k-nearest-neighbor (k=3) classifier (3NN) that use Euclidean distance and data set 1 as prototypes. It can be seen from Table 2 that, stochastic modeling (SM) achieves almost the same level performance as 1NN and 3NN, even though 1NN and 3NN are much more expensive procedures[3]. In fact, SM

[2]The degree to which there exists any tradeoff here is based on the degree to which the training data is flawed – in theoretical contexts, there is no tradeoff.

[3]A run on 4975 images (set 2) using the pseudo-gray classifier trained on 4997 images (set 1) to 250

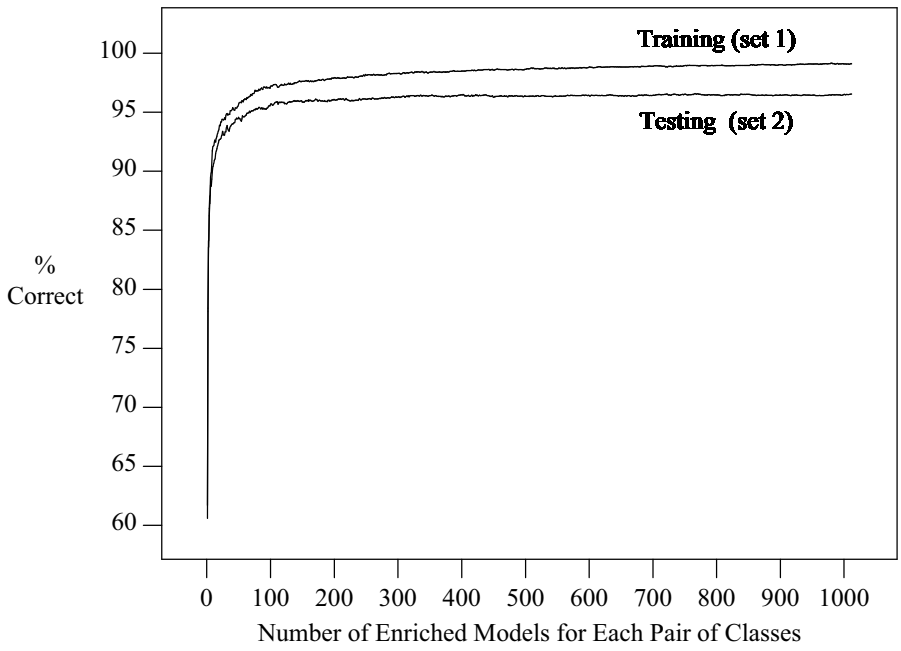


Figure 1: Growth of correct rate for training (set 1) and testing sets (set 2) as the number of enriched models increases (pseudo-gray data).

outperformed both 1NN and 3NN in some of tests with 250 models, and more with 750 models.

Table 2: Summary of Classification Results

Classifier	*% Correct at Top Choice* Set 1	Set 2	Set 3	Set 4
SM using pseudo-gray values (250 models)	98.14	96.10	96.25	96.76
SM using binary pixel values (250 models)	97.40	95.20	95.25	95.96
SM using convolution responses (250 models)	98.04	95.42	95.73	96.16
SM using pseudo-gray values (750 models)	98.96	96.54	96.58	97.15
SM using binary pixel values (750 models)	98.48	95.84	95.85	96.39
SM using convolution responses (750 models)	98.76	95.86	95.93	96.68
1NN using pseudo-gray values	-	97.09	97.38	97.64
1NN using binary pixel values	-	95.22	95.63	95.73
1NN using convolution responses	-	95.50	95.43	96.25
3NN using pseudo-gray values	-	97.13	97.18	97.72
3NN using binary pixel values	-	95.14	95.35	95.45
3NN using convolution responses	-	96.02	95.71	96.23

Using the results of stochastic modeling on set 2, a logistic regression model was estimated for the combination of the rankings using three feature sets. The estimated coefficients for the rankings using pseudo-gray, binary, and convolution response data with 250 models are 2.3019, 1.3813, and 1.8869, and with 750 models are 2.4079, 1.4571, and 1.9707 respectively. It can be seen from Table 3 that, the classifiers derived using different feature sets are sufficiently independent, so that further improvement on performance is possible by their combination.

Table 3: Results of Classifier Combination

Classifier	*% Correct at Top Choice* Set 1	Set 2	Set 3	Set 4
Combination of 3 SM classifiers (250 models)	-	96.56	96.58	97.17
Combination of 3 SM classifiers (750 models)	-	96.98	96.80	97.50

6 Conclusions

We have presented a method for automatic classifier construction for an arbitrary pattern recognition problem. The method is unique in that the power of the classifier improves as training continues, and, modulo the quality of a given training set, there is little tradeoff between accuracy on the training set and the projectability to an unseen test set. Experimental results show that the method is effective on handwritten digit recognition using real-world images, and it achieves a level of accuracy that is close to expensive nearest-neighbor matching procedures.

models took 1192 seconds on a SUN SPARC 2, whereas a run of 1NN with the same data took 5635 seconds on the same machine. A run with SM using 750 models took 3609 seconds.

The method is at an early stage of investigation, and development in many directions is possible. Challenging open problems include the relationship between the model types and their discriminating powers, convergence of classification performance under approximate uniformity, and alternative methods for model combination. Moreover, the study of the method reveals some fundamental problems in pattern recognition that are important to many classification methodologies. These include the meaning of representativeness of a training set, measures of mutual dependencies among features, characterizations of the complexity of a given recognition problem, as well as analytical studies of classifier performance versus computational complexity.

Acknowledgements

The authors would like to thank Evelyn Kleinberg and Hing-Kai Hung for many useful discussions, and Jonathan Hull for assistance in obtaining the NIST database.

References

[1] H.S. Baird, H.P. Graf, L.D. Jackel, W.E. Hubbard, A VLSI Architecture For Binary Image Classification, in J.C. Simon, ed., *From Pixels to Features* , North Holland, 1989, 275-286.

[2] R. Berlind, Almost-uniformity in Stochastic Modeling, to appear.

[3] T.K. Ho, *A Theory of Multiple Classifier Systems And Its Application to Visual Word Recognition*, Doctoral Dissertation, Department of Computer Science, State University of New York at Buffalo, 1992.

[4] E.M. Kleinberg, Stochastic Discrimination, *Annals of Mathematics and Artificial Intelligence*, **1**, 1990, 207-239.

[5] E.M. Kleinberg, The Theory of Stochastic Modeling in Pattern Recognition, to appear.