Frequency Coding: An Effective Method for Combining Dichotomizers

Srinivas Andra[a], George Nagy[a] and Cheng-Lin Liu[b]

[a]DocLab, ECSE Department, Rensselaer Polytechnic Institute, Troy NY 12180, USA
[b] NLPR, Institute of Automation, Chinese Academy of Sciences, Beijing 100080, P.R. China
andras@rpi.edu nagy@ecse.rpi.edu liucl@nlpr.ia.ac.cn

ABSTRACT

Binary classifiers (dichotomizers) are combined for multi-class classification. Each region formed by the pairwise decision boundaries is assigned to the class with the highest frequency of training samples in that region. With more samples and classifiers, the frequencies converge to increasingly accurate non-parametric estimates of the posterior class probabilities in the vicinity of the decision boundaries. The method is applicable to non-parametric discrete or continuous class distributions dichotomized by either linear or non-linear classifiers (like support vector machines). We present a formal description of the method and place it in context with related methods. We present experimental results on machine-printed digits that demonstrate the viability of frequency coding in a classification task.

Keywords: Frequency coding, dichotomizers, nonparametric classification

1. INTRODUCTION

Discriminative classifiers like support vector machines[1] and boosting algorithms[2] are most readily formulated for two-class problems. Any multiclass problem with N_c classes can be decomposed in many ways, including N_c one-class-versus-the-rest binary classifiers, and $N_c(N_c-1)/2$ dichotomies. We concentrate here on dichotomizers, which are generally simpler and easier to train. For instance, even if every pair of classes is linearly separable, one or more of the classes may not be linearly separable from all of the others.

The conventional methods of combining maximum-margin classifiers (e.g., SVM and Boosting) include majority voting, and computing the posterior class-conditional probabilities according to the distance of the unknown pattern from the separating boundaries.[3–5] The former suffers from the occurrence of tied votes, and the latter from the difficulty of estimating the tails of probability distributions near the decision boundaries. *Frequency coding* is an alternative method for combining dichotomizers that significantly reduces both problems by means of direct estimates of the class probabilities corresponding to every possible output of the dichotomizers.

This classifier is based on labeling the regions of feature space formed by the decision boundaries of the dichotomizers. The maximum number M of convex regions that can be formed by K hyperplanes grows super-linearly with n, the number of features (i.e., the dimensionality of the feature space), and with K, the number of hyperplanes. With nonlinear boundaries, the maximum number of regions grows much faster. For the linear case (hyperplanes), it can be shown, using Zaslavsky's Theorem,[6] that

$$M = \binom{K}{0} + \binom{K}{1} + \binom{K}{2} + \ldots + \binom{K}{n}. \tag{1}$$

Please address correspondence to Srinivas Andra.

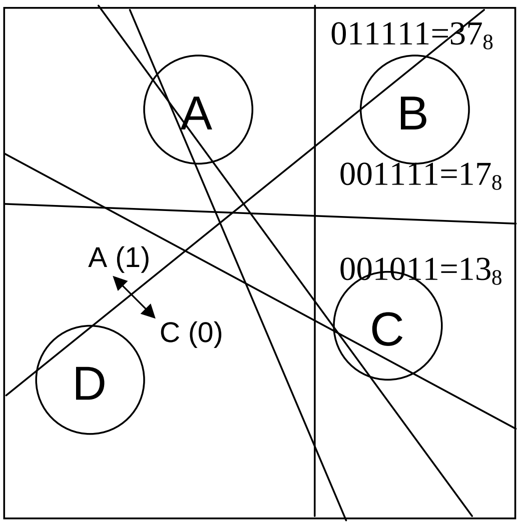

Figure 1. Some region indices for 4 classes.

2. FREQUENCY CODING

The training set has N_t patterns. Each training pattern is labeled with one of N_c class labels $\{C_1, C_2, \ldots, C_{N_c}\}$, and one of N_s style labels $\{S_1, S_2, \ldots, S_{N_s}\}$.* The training patterns are classified by K dichotomizers, where the output of dichotomizer Y_k on training pattern $\mathbf{x}_i$ is $Y_k(\mathbf{x}_i) = 1$ or $Y_k(\mathbf{x}_i) = 0 : \mathbf{Y} = \mathbf{Y}(\mathbf{x}_i) = (Y_1(\mathbf{x}_i), Y_2(\mathbf{x}_i), \ldots, Y_K(\mathbf{x}_i))$.

The dichotomizers together assign pattern $\mathbf{x}_i$ to a region in the feature space. A region can be described either by its K-element binary region vector $\mathbf{Y}$ or by its scalar region index m, $m = 1, \ldots, M, M = 2^K$. The index m can be the value of the binary integer formed by concatenating the elements Y_k of $\mathbf{Y}$ in any fixed order. The result of assigning all the training patterns $\mathbf{x}_i$ is a set of region assignment matrices $\{\mathbf{B}_m, m = 1, 2, \ldots, M\}$ corresponding to the M possible regions. Each matrix $\mathbf{B}_m$ has elements b_{ij}^m, where b_{ij}^m is the number of training patterns of class C_i and style S_j assigned to region m by the dichotomizers. For the case of linear dichotomizers, we can see from (1) that there are far fewer than 2^K regions. In Fig. 1 we show a possible configuration for 4 classes and 2 features. The 6 dichotomizers, determined by pairwise training on a training set, are A/B, A/C, A/D, B/C, B/D, C/D. There are at most 22 rather than $2^6 = 64$ regions, but the region index is shown for only 3 of the regions. Under the assumption that the joint class-and-style posterior probabilities (in case of style-unlabeled training patterns, class posterior probabilities) are uniform in each region, these probabilities are estimated from the *frequencies* of the training samples contained in the region assignment matrices $\{\mathbf{B}_m,\ m = 1, 2, \ldots, M\}$ as follows:

$$P(\mathsf{C} = C_i, \mathsf{S} = S_j|\mathbf{x}) = P(\mathsf{C} = C_i, \mathsf{S} = S_j|m) = \frac{b_{ij}^m}{\sum\limits_{i=1}^{N_c}\sum\limits_{j=1}^{N_s} b_{ij}^m}, \tag{2}$$

where the joint class-and-style posterior probability at $\mathbf{x}$ is equal to the joint class-and-style posterior probability in the region m assigned to $\mathbf{x}$. For singlet classification (i.e., classification of individual patterns under the assumption of class-conditional independence), we need only class posterior probability:

$$P(\mathsf{C} = C_i|\mathbf{x}) = P(\mathsf{C} = C_i|m) = \frac{\sum\limits_{j=1}^{N_s} b_{ij}^m}{\sum\limits_{i=1}^{N_c}\sum\limits_{j=1}^{N_s} b_{ij}^m} \propto \sum_{j=1}^{N_s} b_{ij}^m. \tag{3}$$

*In multifont or multi-writer classification, the class-conditional feature distributions are mixture densities. The *style label* of each training pattern identifies its style (component density, font or writer). Although we consider the general case of class-and-style-labeled training patterns, frequency coding may be used without style labels.

Each class is thus assigned to the corresponding estimate of its posterior probability in the whole region, and the region is then assigned the class label of the dominant class. The posterior probabilities can also be used to compute a confidence measure.

Ties can be broken according to the frequencies of the tied classes in the adjacent regions. The number of tied region frequencies resulting in ambiguous regions decreases with increasing training set size, in contrast to conventional voting, where the number of ambiguous regions remains constant. A special case of ties occurs with empty regions. If a test pattern falls into an empty region m, i.e., if no training patterns were assigned to it, then its b_{ij}^{m} are set to $b_{ij}^{m'}$, corresponding to the frequencies of the nearest non-empty region m' as measured by the Hamming distance between the region vectors $\mathbf{Y}$. Ties between equally-near regions are broken first by dominant class among all these regions, second by testing the next-nearest regions. Alternatively, regions without a dominant class may be assigned to a REJECT category.

New samples are classified by every dichotomizer. Then the class label assigned to the resulting region index is retrieved from the region table, i.e., the set of region assignment matrices.

3. RELATION TO OTHER CLASSIFIERS

Frequency coding is related to stacking[7] and, in particular, stacking pairwise classifiers or simply *pairwise stacking*.[8] Like the latter method, frequency coding uses the binary outputs of the pairwise classifiers as an intermediate step in the final classification decision. Pairwise stacking classifies patterns in the space of dichotomizer binary outputs (level-1 space - binary feature space) using any arbitrary classifier. However, frequency coding classifies patterns in the input space (level-0 space) by estimating the candidate class posterior probabilities. Each dichotomizer's 0/1 output defines a (linear or nonlinear) half-space in level-0 space. Their intersections form regions that constitute the frequency coding interpretation. The alternative interpretation, in terms of pairwise stacking, is that the dichotmizer outputs together form a binary feature vector in level-1 space.

The confidence scores of dichotomizers may be used instead of their binary outputs. For instance, the confidence scores may be converted into calibrated (pairwise) posterior probabilities using the Platt's method,[3] and combined for multiclass classification by pairwise coupling and related schemes.[4, 5] In this case, the similarity to frequency coding disappears entirely. An alternative stacking scheme could use the pairwise probabilities as level-1 features for classification by a level-1 classifier, instead of combining them by pairwise coupling.

Pairwise stacking with a nearest neighbor classifier as the level-1 classifier and frequency coding produce identical results when a test pattern falls in a region already populated by the training patterns. When the region is empty, however, frequency coding may assign a different label than pairwise stacking, depending on the populations of the adjacent regions.

Most importantly, the probabilistic formulation of frequency coding enables style-constrained classification.[9, 10] While some stacking methods attempt to minimize the impact of irrelevant dichotomizers (dichotomizers classifying test patterns belonging to *untrained* classes), style-constrained classification benefits from the view that the output of such a dichotomizer may be very relevant. For instance, it is likely that a 5/9 digit dichotomizer for hand-printed digits will almost always opt for 5 on 6's, and for 9 on 8's, even though it was never trained on either 6's or 8's.

The region class frequencies yield a piecewise uniform approximation to the posterior class probabilities. Unlike Parzen Windows,[11] the "windows" do not shrink with increasing sample size. As in k-Nearest Neighbors,[12] only nearby samples within a convex region participate in each decision, but the number of participating neighbors varies from region to region instead of being set in advance. The windows tend to cluster around the decision boundaries, providing finer resolution of the posterior class probabilities just where it matters.

Both the frequency coding scheme and the error-correcting output coding (ECOC) scheme[13, 14] combine margin-based dichotomizers for multi-class classification. The ECOC scheme combines binary classifiers (dichotomizers) trained on subsets of classes for multiclass classification. The decisions of K binary classifiers are combined into codewords to uniquely represent each of N_c classes. An $N_c \times K$ coding matrix $\Xi \in \{0,1\}^{N_c \times K}$ defines the relation between N_c classes and K dichotomizers: each row gives the codeword for a class. On a test pattern, the outputs of the dichotomizers (either binary decisions or margin values) form a prediction vector. The test pattern is then assigned the class whose codeword is closest to the prediction vector in some sense (Hamming distance or loss-based distance). Alternatively, the margins are transformed into two-class probabilities, which are then combined into multi-class probabilities. The error rate of ECOC scheme depends on the chosen distance metric or the probability estimation technique.

4. EXPERIMENTAL RESULTS

4.1. Machine-printed Digits

The data consisted of 24,000 6-pt printed digits scanned at 200 dpi. The printed digits were evenly distributed among five different fonts viz., Avant Garde (A), Bookman Old Style (B), Helvetica (H), Times New Roman (T), and Verdana (V). The five fonts were divided into two styles: *Style 1* (A, B, H) and *Style 2* (T, V). This division maximized the inter-style confusion between *Style 1* and *Style 2* as observed from the experiments. A few representative samples are shown in Figure 2.

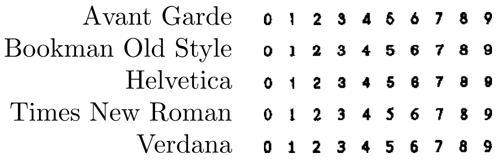

Figure 2. Samples of the machine-printed digits, reproduced at approximately actual size.

We used the top 5 principal components of the 64-dimensional directional edge features as classification features. The features were normalized using max-min normalization scheme. Each feature (attribute) in the training set is normalized as follows:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{4}$$

where x_{max} and x_{min} are the maximum and minimum values of feature x across all the training samples. This results in each feature value in the training set being in the range $[0, 1]$. The test set is normalized using the x_{max} and x_{min} values obtained from the training set.

We constructed two region-frequency classifiers based on frequency coding: (i) *class-pair region-frequency classifier* – uses 45 class-pair dichotomizers, and (ii) *class-and-style-pair region-frequency classifier* – uses both class-pair and class-and-style-pair dichotomizers, a total of 135 dichotomizers (45 + 2 × 45) including the 45 class-and-style-pair dichotmomizers from each style. All the 135 dichotomizers were SVMs with a linear kernel and a capacity constant of $C = 1$. The error rates of the two region-frequency classifiers are compared with class-pair majority vote, unimodal Gaussian (one Gaussian per class) and bimodal Gaussian (one Gaussian per class-and-style) classifiers.

We present some statistics for the class-pair region-frequency classifier. Detailed statistics for both region-frequency classifiers may be found in .[15] The training set of 12,000 digits populated 1926 regions. (A larger training set could conceivably populate all of the maximum number of 1,385,980 regions formed in 5-D by 45 hyperplanes in general positions.) The test set, also of 12,000 digits, fell into 1906 regions, with 601 of these regions corresponding to empty regions in the training set. The fraction of errors was almost twice as high in the regions to which no training sample was assigned, but the number of such regions decreases with the size of the training set. Of the 1926 regions populated by the training set, there are 66 regions of disagreement between the class-pair region-frequency classifier and the class-pair majority vote classifier. Most of these regions are nearly empty with only one training sample falling in 41 of them. Table 1 shows a few rows of the 1926-row region table, including the region labels assigned by the class-pair region frequency classifier and the class-pair majority vote classifier.

The error rates of the five classifiers are shown in Table 2. The region-frequency classifiers compare favorably to the other classifiers. Both region-frequency classifiers yield lower error rates than the conventional vote classifier. The class-and-style-pair region frequency classifier yields the best accuracy of all classifiers due to the

Table 1. Part of the region table for the class-pair region-frequency classifier (45 dichotomizers) in a five-dimensional feature space.

Classes Octal Codes	0	1	2	3	4	5	6	7	8	9	Region frequency label	Majority vote label
000001777760000	0	0	34	0	0	0	0	0	0	0	2	2
000367732160026	0	0	0	0	0	0	0	84	0	0	7	7
404010657570441	0	0	0	8	0	0	0	0	0	0	**3**	**8**
444010647711441	0	0	0	11	0	0	0	0	3	2	3	3
444110747711440	0	0	0	4	0	0	0	0	0	8	**9**	**3**
464010647410441	0	0	0	3	0	0	0	0	34	4	8	8
644010044510771	0	0	0	0	0	13	38	0	0	0	6	6
744010004510771	0	0	0	0	0	11	6	0	0	0	**5**	**6**
777010646410440	345	0	0	0	0	0	0	0	0	0	0	0

fine division of the feature space into a large number of regions (the training patterns fell into 6616 regions). The experiments were repeated with interchanged training and test sets (as shown in the right sub-columns of Table 2) without significant difference. The difference between the training and test error rates indicates that the region-frequency classifiers would benefit from more training samples. Unequal class priors, which are not taken into account in conventional voting, would favor the proposed method even more.

Table 2. Comparison of five classifiers on 12,000 digits

Classifier	Original training and test sets		Interchanged training and test sets	
	Training error %	Test error %	Training error %	Test error %
Class-pair majority vote classifier	3.2	3.3	3.2	3.2
Class-pair region-frequency classifier	2.4	3.0	2.2	2.8
Class-and-style-pair region-frequency classifier	0.7	2.2	0.9	2.3
Unimodal Gaussian classifier	2.9	2.9	2.8	2.9
Bimodal Gaussian classifier	2.3	2.4	2.3	2.4

4.2. Handwritten digits

We conducted experiments on handwritten digits to assess the suitability of frequency coding in real-world applications. The dataset SD7, which is contained in the NIST special database SD19, was used for these experiments.[16] The writers of SD7 were high school students. The dataset is considered difficult to recognize. A few samples are shown in Figure 3.

The SD7 dataset was divided into training and test sets with 100 writers in each set. The training set comprised the writers (2100–2199), whereas the test set the writers (2200–2299). Due to the limitations of our current implementation of frequency coding scheme, experiments were conducted only on a subset of five digits $(3, 5, 6, 8, 9)$. The training and test sets consisted of 5689 and 5722 samples respectively. The top 50 principal components of the 100-dimensional directional edge features were used as classification features. The

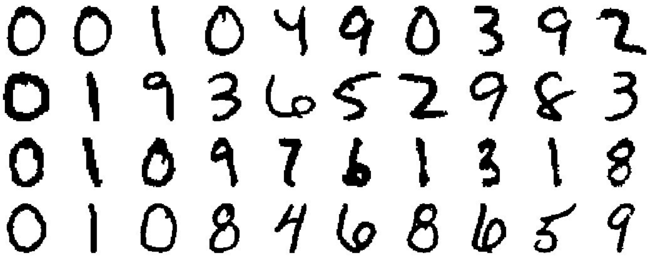

Figure 3. Handwritten samples from NIST SD7 dataset.[16]

features were normalized using max-min normalization scheme as described in Section 4.1. The large number of writers in the training or test set precluded experiments with class-and-style-pair region-frequency classifier and multimodal Gaussian classifier (for the case of machine-printed digits with two broad styles, bimodal Gaussian classifier was a natural choice).

Table 3. Comparison of five classifiers on 12,000 digits

Classifier	Original training and test sets		Interchanged training and test sets	
	Training error %	Test error %	Training error %	Test error %
Class-pair majority vote classifier	0.0	1.2	0.0	1.6
Class-pair region-frequency classifier	0.0	1.3	0.0	1.6
Unimodal Gaussian classifier	1.1	2.5	1.2	2.7

The SVM dichotomizers in both region-frequency and majority vote classifiers used an RBF kernel with parameters $(C, \gamma) = (5, 2)$ obtained by 2-fold cross-validation. The error rates of the three classifiers are shown in Table 3. The region-frequency classifier yields similar error rates as the majority vote classifier, but the unimodal Gaussian classifier yields relatively higher error rates. The training set populated 136 regions. It is noteworthy that perfect accuracy could be achieved on the training set with both majority vote and region-frequency classifiers. The difference between the training and test error rates of all classifiers points towards the poor generalization accuracies with the SD7 dataset.

We encountered the following problems in our experiments with the full SD7 dataset with all the ten digits. The training and test sets of the full SD7 dataset have 11585 and 11660 samples respectively. We observed that the training set populated nearly as many regions as the number of samples. Since these regions are nonempty, most of the regions therefore have only one sample in them. Two problems arise as a consequence. On the experimental front, assigning test patterns to feature space regions requires computation of a distance matrix with more than 100 million entries. In our current implementation, this causes MATLAB to throw the *out of memory* error. This problem however can be resolved with an efficient implementation.

The second problem concerns with the nature of posterior probability assignment to the regions. This is a major problem because, with only one sample per region, there is hardly any gradation in the posterior probability assignment. In this case, frequency coding is similar to a nearest neighbor classifier, albeit with different decision boundaries. Our future work will focus on developing algorithms and strategies to address this problem.

5. EXTENSIONS

The proposed decision scheme can be applied to combine any arbitrary set of dichotomizers. This is similar to ECOC scheme, but the method of combination is different. As is the case with ECOC, the design of a set of effective dichotomizers would be an issue. The dichotomizers may be chosen so as to achieve consistent classification of irrelevant classes, as mentioned in Section 3. Use of nonlinear SVMs (i.e., SVMs with nonlinear kernels) as dichotomizers allows us to consider more complex regions in the feature space, which we have investigated.[15]

Binary style dichotomizers can also be used for field classification.[15, 17] For field recognition, we concatenate the feature vectors of all the patterns in the field, and treat the entire field as a single pattern. It is not, however, necessary to train the dichotomizers on pattern fields (this would require an inordinate number of patterns because the number of field classes increases exponentially with field length). Instead, the region indices of singlet patterns are concatenated to form region indices for virtual field patterns. Hence training does not take any longer for field classification than for singlet classification.

During classification, the field label of the nearest field region index is assigned to an unknown pattern field. The field region indices need not be stored because the required field indices can be quickly computed at classification time from the singlet region indices. The computation ensures that each field index is based only on same-style patterns. As with other style-constrained classification methods,[9, 10] increase in field length reduces inter-style confusions, possibly at the cost of some additional intra-style confusions.[18]

Acknowledgements

We are indebted to Dr. S. Veeramachaneni, IRST, Trento, Italy for advice and valuable discussions on this topic.

REFERENCES

1. V. Vapnik, *Statistical Learning Theory*, Wiley, New York, 1998.
2. Y. Freund, "Boosting a weak learning algorithm by majority," *Inf. Comput.* **121**(2), pp. 256–285, 1995.
3. J. Platt, "Probabilistic outputs for support vector machines and comparison to regularized likelihood methods," in *Advances in Large Margin Classifiers*, A. Smola, P. Bartlett, B. Schölkopf, and D. Schuurmans, eds., MIT Press, (Cambridge, MA), 2000.
4. T. Hastie and R. Tibshirani, "Classification by pairwise coupling," *The Annals of Statistics* **26**(1), pp. 451–471, 1998.
5. T.-F. Wu, C.-J. Lin, and R. C. Weng, "Probability estimates for multi-class classification by pairwise coupling," *Journal of Machine Learning Research* **5**, pp. 975–1005, 2004.
6. T. Zaslavsky, "Facing up to arrangements: Face-count formulas for partitions of space by hyperplanes," *AMS Memoirs* **1**(154), 1975.
7. D. H. Wolpert, "Stacked generalization," *Neural Networks* **5**, pp. 241–259, 1992.
8. P. Savicky and J. Fürnkranz, "Combining pairwise classifiers with stacking," in *Proceedings of the 5th International Symposium on Intelligent Data Analysis (IDA-03)*, pp. 219–229, Springer, 2003.
9. P. Sarkar and G. Nagy, "Style consistent classification of isogenous patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **27**, pp. 88–98, January 2005.
10. S. Veeramachaneni and G. Nagy, "Style context with second-order statistics," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **27**, pp. 14–22, January 2005.
11. E. Parzen, "On estimation of a probability density function and mode," *Annals of Mathematical Statistics* **33**(3), pp. 1065–1076, 1962.
12. T. M. Cover and P. E. Hart, "Nearest neighbor pattern classifications," *IEEE Transaction on Information Theory* **13**(1), pp. 21–27, 1967.
13. T. G. Dietterich and G. Bakiri, "Solving multiclass learning problems via error-correcting output codes," *J. Artif. Intell. Res. (JAIR)* **2**, pp. 263–286, 1995.
14. E. L. Allwein, R. E. Schapire, and Y. Singer, "Reducing multiclass to binary: A unifying approach for margin classifiers," *Journal of Machine Learning Research* **1**, pp. 113–141, 2000.

15. S. Andra, *Nonparametric approaches to style consistent classification.* PhD thesis, Rensselaer Polytechnic Institute, Troy, NY, 2006.

16. P. Grother, "Handprinted forms and character database, NIST special database 19," March 1995. Technical Report and CDROM.

17. S. Andra and G. Nagy, "Combining dichotomizers for MAP field classification," in *Proceedings of the 18th International Conference on Pattern Recognition*, (Hong Kong), 2006.

18. X. Zhang and S. Andra, "Towards quantifying the amount of style in a dataset," in *IS&T/SPIE International Symposium on Electronic Imaging 2006*, *Proceedings of the SPIE* **6067**, SPIE, 2006. Document Recognition & Retrieval XIII.