Do Hidden Units Implement Error-Correcting Codes?

Thomas G. Dietterich
Department of Computer Science
Oregon State University
Corvallis, OR 97331-3202
tgd@turing.cs.orst.edu

Extended Abstract

Consider the problem of learning a k-ary classification function $f : \mathcal{R}^n \mapsto \{1, \ldots, k\}$ from examples of the form $\langle \mathbf{x}, f(\mathbf{x}) \rangle$. The standard neural network approach to this problem is to construct a 3-layer feedforward network with k output units, where each output unit designates one of the k classes. During training, the output units are clamped to 0.0, except for the unit corresponding to the desired class, which is clamped at 1.0. During classification, a new $\mathbf{x}$ value is assigned to the class whose output unit has the highest activation. Let us call this the one-per-class approach.

In Dietterich & Bakiri (1991), we showed that an alternative method, called error-correcting output coding (ECOC) gives superior performance. In this approach, each class i is assigned an m-bit binary string, c_i, called a *codeword*. The strings are chosen (by BCH coding methods) so that the Hamming distance between each pair of strings is guaranteed to be at least d. During training on example $\mathbf{x}$, the m output units of a 3-layer network are clamped to the appropriate binary string $c_{f(\mathbf{x})}$. During classification, the new example $\mathbf{x}$ is assigned to the class i whose codeword c_i is closest (in Hamming distance) to the m-element vector of output activations. The advantage of this approach is that it can recover from any $\lfloor (d-1)/2 \rfloor$ errors in learning the individual output units. Error-correcting codes act as ideal distributed representations.

Table 1 shows the results of training backpropagation (by conjugate-gradient) on the isolated-letter speech recognition task of Cole, Muthusamy & Fanty (1990). In this task, a person speaks the name of one of the 26 letters. The speech signal is captured, and 617 real-valued features are extracted and used for training the network (Cole & Fanty, 1991). (Cross-validation training methods were applied to control overfitting and determine the number of hidden units. Data from 120 speakers was used for training, with a separate set of 30 speakers for testing.) With ECOC, the error rate has been reduced by more than 20%. Similar, but more substantial, performance improvements have been obtained with decision-tree algorithms on the NETtalk task (Dietterich & Bakiri, 1991).

The final step of mapping the output activations to the nearest legal binary string c_i can easily be implemented by a set of ordinary sigmoid output units whose incoming weight vectors encode the c_i values. Each dot product computes a quantity inversely proportional

Table 1: Performance on Isolated Letter Speech Recognition

Configuration	Network Architecture	Percent Correct	Percent Error
one-per-class	617:78:26	95.83	4.17
30-bit ECOC	617:156:30	96.73	3.27

to the Hamming distance, so by selecting the output unit with the highest activation, we are selecting the class with the smallest Hamming distance. This leads to the interesting hypothesis that the hidden units of backpropagation networks are naturally discovering error-correcting distributed representations.

If this hypothesis is true, it would explain (in part) the superior performance of neural networks. It would also explain why it is hard to understand the the meaning of individual hidden units.

To test this hypothesis, let us consider the following two predictions: (P1) a 4-layer network should match or exceed the performance of our 30-bit ECOC network, because it could learn a problem-specific error-correcting output code by itself; (P2) a 3-layer network in which the hidden units are trained to produce an error-correcting code should match or exceed the performance of an ordinary 3-layer network.

To test P1, we applied backpropagation to a fully-connected network with two hidden layers configured to mimic the 30-bit ECOC network (617:156:30:26). We trained on data from 60 speakers while testing generalization performance (every 10 epochs) on data from a disjoint set of 60 speakers.[1] Table 2 shows the peak performance attained by this procedure in four separate runs along with an analogous set of four runs for the 30-bit ECOC. A one-tailed t-test (for the hypothesis that ECOC is better than the 4-layer net) is significant with $p < .01$. Moreover, the performance of the 4-layer network is actually worse than the performance of a 3-layer 617:78:26 network (but not statistically significantly). We conclude that back-propagation is not able to discover error-correcting output codes by itself in this domain.

To test P2, we applied backpropagation to a 2-layer network with a 78-bit ECOC (617:78) and compared its performance to an ordinary one-per-class 3-layer network (617:78:26). (Previous studies determined that the one-per-class 3-layer network attains peak performance with 78 hidden units.) By training a layer of logistic units to implement an error-correcting code, we are effectively dictating the hidden-layer representation of a 3-layer network (since the decoding step of the ECOC method can be implemented by a third layer). At issue is whether this error-correcting hidden layer will perform as well as the hidden layers that backpropagation naturally discovers. We followed the same training procedure as above. Table 3 shows the peak performance attained by each method in four separate runs. A one-tailed t-test (for the hypothesis that one-per-class is better than the 2-layer ECOC net) is significant with $p < .005$. We conclude that the hidden layer discovered by the one-per-class approach is *not* an error-correcting code.

[1]In the full version of this paper, we will report results for training on 120 speakers and testing on 30

Table 2: Peak performance of ECOC versus a four-layer network of the same size.

Method	Run	Percent Correct	Percent Error
4-layer	1	94.58	5.42
	2	94.36	5.64
	3	94.87	5.13
	4	93.23	6.77
30-bit ECOC	1	95.16	4.84
	2	95.57	4.43
	3	95.51	4.49
	4	95.54	4.46

Table 3: Peak performance of the one-per-class approach versus a two-layer network of the same size.

Method	Run	Percent Correct	Percent Error
2-layer ECOC	1	92.14	7.86
	2	92.40	7.60
	3	92.40	7.60
	4	92.40	7.60
one-per-class	1	94.96	5.04
	2	94.74	5.26
	3	95.06	4.94
	4	94.42	5.58

Based on these experiments, we conclude that both predictions are incorrect and that the underlying hypothesis is false. The hidden units of backpropagation networks do not normally discover error-correcting codes themselves. This gives additional evidence that error-correcting output codes provide an additional source of power for improving neural net generalization performance. Along with our previous results, this suggests that error-correcting output codes should be adopted (instead of the one-per-class approach) as the standard method for applying back-propagation to k-ary classification tasks.

References

Cole, R., & Fanty, M. (1991). Spoken letter recognition. In *Advances in Neural Information Processing Systems, 3*, San Mateo, CA: Morgan Kaufmann.

Cole, R. Muthusamy, Y. and Fanty, M. (1990). The ISOLET spoken letter database. Technical Report No. CSE 90-004. Beaverton, OR: Oregon Graduate Institute of Science and Technology, Department of Computer Science and Engineering.

Dietterich, T. G., Bakiri, G. (1991) Error-correcting output codes: A general method for improving multiclass inductive learning programs. *Proceedings of the Ninth National Conference on Artificial Intelligence (AAAI-91)*, Anaheim, CA: AAAI Press.

Sejnowski, T. J., and Rosenberg, C. R. (1987). Parallel networks that learn to pronouce English text. *Complex Systems*, 1, 145–168.

speakers. The data in this summary are the results of our cross-validation runs.