A NOVEL UNSUPERVISED CLASSIFICATION METHOD

In Proc. of the Intl. Conf. on Artificial Neural Networks, Brighton, pages 91-96. IEE, 1993.

J. Schmidhuber and D. Prelinger

Technische Universität München
Germany

Abstract. *Assume we are given a set of pairs of patterns. We know that both patterns of each pair belong to the same class. We do not know in advance, however, anything about the nature of the classes, which features are characteristic for each class, how many classes there are, and which patterns belong to which class. We present a novel unsupervised neural system that learns without a teacher to create distributed representations of classes such that patterns belonging to the same class are represented by the same activation pattern while patterns belonging to different classes are represented by different activation patterns. The approach can be related to the IMAX method of Hinton, Becker and Zemel (1989, 1991). Experiments include a stereo task proposed by Becker and Hinton, which can be solved more readily by our system.*

BASIC IDEA

As an example, consider the following stereo task (Becker and Hinton [2]): There are two binary images called the 'left' image and the 'right' image. Each image consists of two 'strips' – each strip being a binary vector. The right image is purely random. The left image is generated from the right image by choosing, at random, a single global shift to be applied to each strip of the right image. An input pattern is generated by concatenating a strip from the right image with the corresponding strip from the left image. The input can be interpreted as a fronto-parallel surface at an integer depth. The only local property that is invariant across space is the stereoscopic depth or shift (Becker and Hinton, [2]).

With a given pair of different input patterns, the first pattern can tell us something (but not everything) about the second pattern. Likewise, the second pattern can tell us something (but not everything) about the first pattern. Let us assume that with a given pair of different input patterns, an unsupervised learning system is told only that both patterns in some way belong to the same class. It is not told how many different classes there are. It is not told anything about the concept of stereoscopic depth. The system's task is to classify each input pattern such that patterns from the same class (the ones with the same shift – but the system does not know that in advance) are represented by the same activation pattern. This activation pattern should be different from activation patterns representing input patterns with different shifts (belonging to different classes). Thus, after the training phase (after exposure of the unsupervised system to a set of pairs of input patterns), different output patterns should correspond to different shifts (the only non-trivial common properties of both elements of a pair of input patterns). In other words, the system's task is to discover different classes of stereoscopic shift by seeing positive training examples only.

Our basic approach to unsupervised discovery of classifications from positive training examples only is based on two neural networks called T_1 and T_2. Both can be implemented as standard back-prop networks [8]. With a given pair of input patterns, T_1 sees the first pattern, T_2 sees the second pattern. We force each network to convey information about its input – under the constraint that each network has to emit the *same* output in response to the two (in general) different input patterns of each pair. Thus the output of both networks can be regarded as a classification of whatever non-trivial properties are common to both patterns of a pair.

Both networks have q output units. Let $p \in \{1, \ldots, m\}$ index the input patterns. T_1 produces as an output the classification $y^{p,1} \in [0, \ldots, 1]^q$ in response to an input vector $x^{p,1}$. T_2 produces as an output the classification $y^{p,2} \in [0, \ldots, 1]^q$ in response to an input vector $x^{p,2}$. The conflicting goals are: (A) $y^{p,1}$ should convey information about $x^{p,1}$, and $y^{p,2}$ should convey information about $x^{p,2}$. (B) But $y^{p,1}$ and $y^{p,2}$ also should match.

We express the trade-off between (A) and (B) by means of two opposing costs.

(B) is expressed by an error term M (for '*M*atch'):

$$M = \sum_{p=1}^{m} \|y^{p,1} - y^{p,2}\|^2. \tag{1}$$

Here $\|v\|$ denotes the Euclidean norm.

(A) is enforced by additional error terms D_l ($l = \{1, 2\}$) (for '*D*iscrimination'). D_l will be designed to encourage significant Euclidean distance between classifications of different input patterns. As shown by Schmidhuber and Prelinger [7], D_l can be defined in more than one reasonable way. The various alternative definitions of D_l have mutual advantages and disadvantages – in the context of a given problem, the most appropriate definition of D_l can be plugged into equation (2) below. Due to limited space, however, we will limit ourselves to a technique called 'predictability minimization' recently introduced by Schmidhuber [6]. See next section.

Both $T_l, l = 1, 2$ minimize

$$\epsilon M + (1 - \epsilon) D_l. \tag{2}$$

The error functions are minimized by gradient descent. This forces the classifications to be more like each other, while at the same time forcing them not to be too general but to tell something about the current input. The procedure is *unsupervised* in the sense that no teacher is required to tell the classifiers *how* to classify their inputs.

PREDICTABILITY MINIMIZATION FOR DEFINING D_l

Schmidhuber [6] shows how D_l can be defined with the help of *intra-representational* adaptive predictors that try to predict each output unit of some T_l from its remaining output units, while each output unit in turn tries to extract properties of the environment that allow it to *escape* predictability. This was called the *principle of predictability minimization.* This principle encourages the output units to convey maximal information about the input patterns. Furthermore, each output unit of T_l is encouraged to represent environmental properties that are statistically independent from environmental properties represented by the remaining output units. The procedure aims at generating binary 'factorial codes' [1]. Unlike the methods used by Linsker [3], Becker and Hinton [2], and Zemel and Hinton [9]) this method has a potential for removing even non-linear statistical dependencies[1] among the output units of some classifier.

Let us define

$$\bar{D}_l = -\frac{1}{2} \sum_i (s_i^{p,l} - y_i^{p,l})^2, \tag{3}$$

where the $s_i^{p,l}$ are the outputs of S_l^i, the i-th additional so-called *intra-representational* predictor network of T_l (one such additional predictor network is required for each output unit of T_l). The S_l^i are trained to predict the expected value of $y_i^{p,l}$ from $\{y_k^{p,l}, \quad k \neq i\}$ by maximizing $\bar{D}_l$.

To encourage even distributions in output space, we slightly modify $\bar{D}_l$ and obtain

$$D_l = -\frac{1}{2} \sum_i (s_i^{p,l} - y_i^{p,l})^2 + \frac{\lambda}{2} \sum_i (0.5 - \bar{y_i}^l)^2. \tag{4}$$

This is the discriminating error term that goes into equation (2).

PREVIOUS WORK

Becker and Hinton [2] solve the stereo problem by maximizing the mutual information between the outputs of T_1 and T_2. This corresponds to the notion of finding mutually predictable yet informative input transformations. The method was called IMAX.

The nice thing about IMAX is that it expresses the goal of finding mutually predictable yet informative input transformations in a principled way (in terms of a single objective function).

In contrast, our approach involves two separate objective functions that have to be combined using a relative weight factor. An interesting feature of our approach is that it conceptually separates two issues: (A) the desire for information preserving mappings from input to representation, and (B) the desire for mutually predictable representations. There are many different approaches (with mutual advantages and disadvantages) for satisfying (A). As mentioned above, in the context of a given problem, the most appropriate alternative approach can be 'plugged into' the basic architecture.

Another difference between IMAX and our approach is that our approach does not only enforce mutual predictability but also equality of $y^{p,1}$ and $y^{p,2}$. This does not affect the generality of our system, however. In fact, one advantage of our simple approach is that it makes it trivial to decide whether the outputs of both classifier essentially represent the same thing. With IMAX, this is in general more complicated.

Finally, it turns out that certain problems can be solved more easily using our approach instead of IMAX. See next section.

[1] Steve Nowlan has described an alternative non-predictor based approach for finding non-redundant codes [4].

STEREO EXPERIMENT: A COMPARISON WITH IMAX

Schmidhuber and Prelinger [7] describe a number of successful experiments with systems based on the first section. Due to space limitations, this section focuses on an experiment that compares IMAX to our approach.

All networks used below were trained by Werbos' back-propagation algorithm [8]. In all cases we used the activation dynamics of Rumelhart et al. [5], as well as 'on-line' learning: Weight changes took place immediately after each presentation of some randomly chosen input pattern. Approximations of mean values $\bar{y}_i^l$ were updated by the formula

$$\hat{y}_i^l \leftarrow 0.95\hat{y}_i^l + 0.05y_i^l,$$

where $\hat{y}_i^l$ is the approximation of $\bar{y}_i^l$ after observing the current input pattern y^l. $\hat{y}_i^l$ was initially set to 0.5.

Details of the task. There are two binary images called the 'left' image and the 'right' image. Each image consists of 2 'strips' – each strip being a binary input vector with 4 components. There are two classifiers with single output units and non-overlapping inputs: Each classifier has 8 input units and 'sees' an 8-dimensional input vector consisting of a strip from the right image and a corresponding strip from the left image generated as follows: The right image is purely random. The left image is generated from the right image by choosing, at random, a single global shift to be applied to each strip of the right image. The shift can be either one bit to the right or one bit to the left – 'overflow bits' generated by shifting some bit of a strip taken from the right image beyond the strip boundaries reappear on the opposite side of the corresponding 'shifted' strip of the left image ('wraparound'). Ambiguous shifts are excluded. The input may be interpreted as a fronto-parallel surface at an integer depth. Since the right image is random, the only common non-trivial property of both classifier inputs is the stereoscopic depth or shift [2]. The goal is to classify each input pattern such that patterns from the same class (the ones with the same shift – but the system is not told anything about shift) are represented by the same activation pattern. This activation pattern should be different from activation patterns representing patterns from different classes (patterns with different shifts). In other words, the only information about the input of the second classifier that is embedded in the input of the first classifier (and vice versa) is the information about the shift. The goal is to find classifications that can be viewed as an extracted representation of this information.

Since the feature to be extracted is one-dimensional, only one predictor per classifier was necessary to predict the single output unit from a bias unit with constant activation (see the second section). It should be noted that for single output units predictability minimization degenerates to the procedure of maximizing the variance of the unit, which (in the binary case) is equivalent to maximizing the entropy of the unit.

The intra-representational predictors and the classifiers learned simultaneously. Each of the two classifiers T_1 and T_2 had 12 hidden units – the predictors had none. The learning rate of the predictors was 1.0, the classifier's learning rate was 0.5. Parameter settings were $\epsilon = 0.5, \quad \lambda = 1.0$. The task was considered to be solved (the shift was considered to be extracted; the patterns were considered to be classified correctly) if (1) the outputs of both classifiers were always equal (with an error margin of 0.1) and (2) each classifier emitted different binary outputs (again with an error margin of 0.1) in response to input patterns with different shifts. This corresponds to 1 bit of mutual information between the outputs and the shift.

With a first experiment, we employed a separate set of weights for each classifier. With ten test runs involving 100,000 training patterns, the classifiers always learned to extract the shift.

Becker and Hinton report that their system (based on binary probabilistic units) was able to extract the shift only if IMAX was applied in successive layer by layer 'bootstrap' stages. In addition, they heuristically tuned the learning rate during learning. Finally they introduced a maximal weight change for each weight during gradient ascent.

In contrast, our method (based on continuous-valued units) does not rely on successive training stages, bootstrap learning, or learning rate adjustments. Once the learning phase is started, no external mechanism influences the behavior of the system. The performance of our system, however, is comparable to the performance of Becker's and Hinton's bootstrapped system. (It should be noted that Becker and Hinton also devised learning procedures for continuous-valued units and for real-valued shifts. In this paper, however, we do not attempt to apply our technique to the real-valued case.)

With a second experiment, we used only one set of classifier weights shared by both classifiers (this leads to a reduction of free parameters). The result was a significant decrease of learning time – with ten test runs the system needed only between 20,000 and 50,000 training patterns to learn to extract the shift.

No systematic attempt was made to optimize learning speed.

CONCLUDING REMARKS

In contrast to IMAX, our method tends to be simpler. It does not require sequential layer by layer 'bootstrapping' or learning rate adjustments. In the binary case, Becker's and Hinton's stereo task can be solved more readily by our system. The classifications emitted by our networks are easier to analyze.

It remains to be seen how well the method of this paper scales to larger problems.

ACKNOWLEDGEMENTS

Thanks to Mike Mozer for fruitful discussions. Thanks to Sue Becker and Rich Zemel for helpful comments. This research was supported in part by a DFG fellowship to J. Schmidhuber, as well as by NSF award IRI–9058450, grant 90–21 from the James S. McDonnell Foundation.

References

[1] H. B. Barlow, T. P. Kaushal, and G. J. Mitchison. Finding minimum entropy codes. *Neural Computation*, 1(3):412–423, 1989.

[2] S. Becker and G. E. Hinton. Spatial coherence as an internal teacher for a neural network. Technical Report CRG-TR-89-7, Department of Computer Science, University of Toronto, Ontario, 1989.

[3] R. Linsker. Self-organization in a perceptual network. *IEEE Computer*, 21:105–117, 1988.

[4] S. J. Nowlan. Auto-encoding with entropy constraints. In *Proceedings of INNS First Annual Meeting, Boston, MA.*, 1988. Also published in special supplement to Neural Networks.

[5] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by error propagation. In *Parallel Distributed Processing*, volume 1, pages 318–362. MIT Press, 1986.

[6] J. Schmidhuber. Learning factorial codes by predictability minimization. *Neural Computation*, 4(6):863–879, 1992.

[7] J. Schmidhuber and D. Prelinger. Discovering predictable classifications. Technical Report CU-CS-626-92, Dept. of Comp. Sci., University of Colorado at Boulder, November 1992.

[8] P. J. Werbos. *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences.* PhD thesis, Harvard University, 1974.

[9] R. S. Zemel and G. E. Hinton. Discovering viewpoint-invariant relationships that characterize objects. In D. S. Lippman, J. E. Moody, and D. S. Touretzky, editors, *Advances in Neural Information Processing Systems 3*, pages 299–305. San Mateo, CA: Morgan Kaufmann, 1991.