ESANN'2005 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 27-29 April 2005, d-side publi., ISBN 2-930307-05-6.

Averaging on Riemannian Manifolds and Unsupervised Learning using Neural Associative Memory

Dimitri Nowicki[1,2] and Oleksiy Dekhtyarenko[1]*

1 – Institute of Mathematical Machines and Systems,
42, Glushkov ave., Kyiv 03187, Ukraine
2 – MIP. Département de Mathématique,
Université Paul Sabatier, 31062 Toulouse cedex 04, France,

Abstract: This paper is dedicated to the new algorithm for unsupervised learning and clustering. This algorithm is based on Hopfield-type pseudoinverse associative memory. We propose to represent synaptic matrices of this type of neural network as points on the Grassmann manifold. Then we establish the procedure of generalized averaging on this manifold. This procedure enables us to endow the associative memory with ability of data generalization. In the paper we provide experimental testing for the algorithm using simulated random data. After the synthesis of associative memory containing generalized data. Cluster centers are retrieved using procedure of associative recall with random starts.

1. Introduction

The aim of this paper is to propose a new neural algorithm for unsupervised learning and clustering.

Our algorithm is based on pseudoinverse associative memory [1]. Such a memory like other Hopfield-type networks is able to some kind of "unsupervised learning": it can memorize unlabeled data. But such networks could not be used for clustering because they cannot generalize: training patterns are memorized "as is". So, the network cannot retrieve cluster centroids from large amount of data patterns.

This problem is partially solved in [2] and [3]. Authors propose the algorithm of adaptive filtering. This algorithm possesses some properties of data generalization but weight matrix of the network is not projective. So, the network deteriorates as number of memorized data is augmented. Since certain number of training patterns ability of associative recall is completely lost.

Unlike [2], [3] our method always produces projective matrices. Using techniques of generalized averaging over Riemannian manifold we construct the synaptic matrix of our network. Associative memory with such a matrix contains vectors generalizing training data. So, these vectors might be used as centroids of the clusters.

Since our method is based on non-iterative neural paradigm it has a good speed; only small number of epochs is needed even for large data sets. This feature makes

* This research was partially supported by INTAS grants POLL 01-257 and YSF 03-55-795

associative-memory algorithm competitive in comparison with self-organizing maps (SOM) of Kohonen [4], the most known neural paradigm used for the purpose of clustering. Unfortunately training of SOMs is often very slow; millions of epochs are required for training of sufficiently large network.

We provide experimental evidence for the associative-memory clustering. This method was tested using sufficiently large simulated data sets.

2. The Algorithm

2.1 Problem statement

Let us have a training sample containing K patterns $\mathbf{x}_1 \ldots \mathbf{x}_K \in \mathsf{R}^n$. Associative memory with generalized patterns is constructed as follows:

At first we divide the training sample into N groups; each group contains m vectors. The number $m<n$ should not exceed n; it is more or equal to desired quantity of clusters. Then we make N matrices of pseudoinverse associative memory: $\mathbf{C}_k$, $k=1 \ldots N$. To join all these instances of associative memory in one matrix we should use the procedure of generalized averaging.

2.2 Generalized averaging on the manifold

Consider a metric space M with metric $\rho(x,y)$ a finite set $\{x_i\}_{i=1}^N \subset M$. The element

$$\bar{x} = \min_{x \in M} \sum_{i=1}^{N} (\rho(x, x_i))^2 \tag{1}$$

is called the *generalized average* of points of this set. Similarly, the point

$$x_m = \min_{x \in M} \sum_{i=1}^{N} \rho(x, x_i) \tag{2}$$

is *a generalized median* of the same set. If M is an Euclidian space generalized average and median are usual average and median respectively. Generalized averaging is considered in [4], problem of generalized averaging on homogenous manifolds might be found in [5].

2.3 Computing generalized average on the Grassmann manifold

Here we use representation of points of the Grassmann manifold $G_{n,m}$ as $n \times n$ (symmetric) projective matrices of rank m; the metric is induced by the Frobenius norm. Hence the problem of generalized average is equivalent to the following minimization problem:

$$\min \varphi(\mathbf{X}) = \sum_{k=1}^{N} \left\| \mathbf{X} - \mathbf{C}_k \right\|^2 \quad \text{s.t.} \quad \mathbf{X}^2 = \mathbf{X}; \operatorname{rank} \mathbf{X} = m \tag{3}$$

After some transformations of the objective function we get:

$$\varphi(\mathbf{X}) = N \left\| \mathbf{X} - \frac{1}{N} \sum_{k=1}^{N} \mathbf{C}_k \right\|^2 + \text{const} = N \left\| \mathbf{X} - \overline{\mathbf{C}} \right\|^2 + \text{const}$$

Thus the problem (3) has been reduced to finding projective matrix of rank m closest to the simple average $\overline{\mathbf{C}}$ of the matrices $\mathbf{C}_k$.

Such a problem might be solved using Newton or conjugated-gradient methods on Grassmann manifold described in [4] but for high-dimensional vectors this became computationally hard. In this paper we use a simplified approach.

Note that the Frobenius norm is invariant with respect to changing orthonormal basis. So, we can choose the basis of eigenvectors of $\overline{\mathbf{C}}$. Let them be ranged by way of decreasing of corresponding eigenvalues. In this basis $\overline{\mathbf{C}}$ is diagonal. We choose $\mathbf{X}$ equal to

$$\mathrm{diag}((\delta_k)_{k=1}^{m}) \tag{4}$$

in this basis; where δ_k=1; k=1…m and δ_k=0 otherwise. Such a matrix is the closest to $\overline{\mathbf{C}}$ amongst projective matrices of rank m. Indeed, making non-diagonal elements non-zero just increases $\left\|\mathbf{X}-\overline{\mathbf{C}}\right\|^2$. Since $\mathbf{X}$ is diagonal (4) is the optimal solution. Thus $\mathbf{X}$ is a matrix of projection to the linear hull of m first eigenvectors of $\overline{\mathbf{C}}$.

3. Experimental Technique

The goal of these series of experiments is to demonstrate network's ability to deal with data having predefined "clustered" structure. Training data could be divided into subsets grouping around the known centers. We are able to tell when the algorithm is able to retrieve these centers.

3.1. The Data

All experiments were carried out using 256-dimensional data vectors with bipolar component values {+1,-1}. The training set was generated as follows:

At first p cluster centers were produced; they were random bipolar vectors with equal probability of values. Then, data vectors themselves were constructed by adding a bipolar noise to center. More precisely, to make a data vector we took h randomly selected components of a center and changed their signs. Noise intensity h was random uniformly distributed number from 1 to H. We shall say that H is a *cluster radius*. For each cluster we generated equal number N of data points. We took K=1000 for all tests. Before entering to the network data were shuffled.

3.2. The Network

At first, N instances of associative memory were trained using pseudoinverse learning rule. Each network memorized m randomly picked data vectors. Synaptic matrices of these networks were averaged using the algorithm described above; and the resulting projective matrix $\mathbf{X}$ was obtained. The network with this matrix was used for simulations in order to retrieve cluster centers.

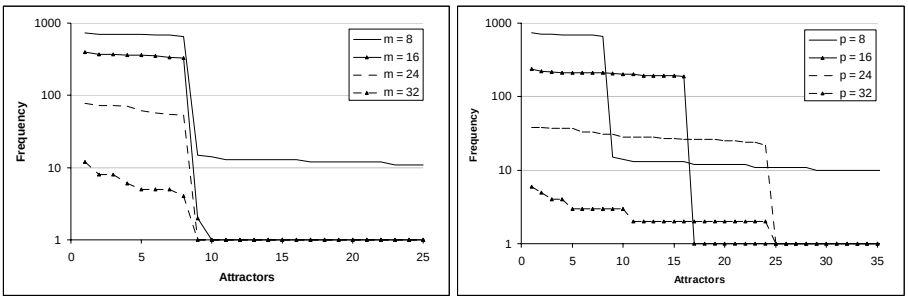
3.3. Finding Attractors

In order to find attractors we performed examination procedure with activation function $f(x)=\text{sign}(x)$. Initial point was taken randomly; iterations were continued until a fixed point was reached.

Recall procedure ran T=10000 times; all attractors found were stored. Then the attractors were sorted by frequency or distinction coefficient.

4. Experimental results

In order to investigate network's behavior we performed experiments described above for different values of parameters. We used a network of 256 neurons and clusters with radius H=64. The matrix of the resulting network was computed by generalized averaging of N=1000 projective matrices. In these experiments all cluster centers were found by convergence from random starts. This was verified by comparing attractors found with centers; first p attractors were identical to centers.



a) **b**)

Fig. 1. Frequencies of attractors of associative clustering network for: different m, p=8 (a); different p, and $m=p$ *(b)*

Figure 1.a) corresponds to the case of constant number of clusters p=8; we varied invariant subspace dimension m. This parameter also means a number of data patterns stored in each instance of pseudoinverse associative memory. We can see that the algorithm works for large range of $m>p$. However, if m is large probability of convergence to a center decreases and number of spurious attractors grows. For m=32 these probabilities have the same order; further increasing of m makes them identical; and centers will be lost.

The second series of experiments is related to the case of $m=p$. In the Figure 1.b) attractors are sorted by frequency; difference between centers and spurious equilibria decreases as number of clusters grows. For $m=p$=32 the network was not able to solve its task; only 24 centers of 32 were found.

Figure 2 demonstrates another way of selecting attractors; here they are sorted by distinction coefficient $r(\mathbf{x},\mathbf{X})$ with network's synaptic matrix. Results of this experiments show that difference of this measure between centers and spurious attractors is much stronger than for frequencies. This ratio is almost the same for different network configurations. So, the distinction coefficient might be used to

reveal centers efficiently. Unfortunately, usage of this criterion combining with random starts cannot guarantee that number of network runs was sufficient to retrieve all centers. This can be seen from the results in Figure. 2 for $p = 32$ – only 28 out of 32 centers were found using the value of distinction coefficient.

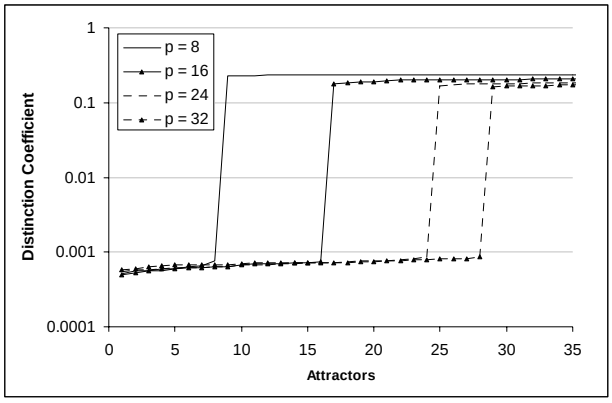


Fig. 2. Distinction coefficients of attractors of associative clustering network for different p, and $m=p$

Note also that usage of successive iterations is necessary to find interesting attractors. If convergence to a stable state is not performed than the probabilities of finding cluster center or any spurious state are practically equal (especially for larger m).

5. Conclusion

Experimental results described above show that proposed associative memories are able to generalize patterns. This makes them a good tool for clustering. Non-iterative nature of neural associative memories makes them quite attractive in comparison with many other neural algorithms of unsupervised learning.

Unfortunately, setting the value of parameter m in associative-memory clustering algorithms requires some a priori knowledge about data to be clustered. This value must be greater or equal than the number of clusters p, but, in the same time, must not exceed this number considerably. Moreover, m is bounded by the well known limitation on memory capacity of Hopfield-type NNs (which is of order n, preferably $m<0.3n$). This limitation might be eliminated by changing type of the manifold and/or metric used in of the main algorithm.

Note that this approach is based on optimization on Riemannian manifolds. This is a powerful technique that could be applied for some other tasks of learning and neural networks. In this paper we used specific manifolds (Grassmann). For this manifold we selected only one type of distance (based on the Frobenius norm) and averaging. Moreover, the solution of corresponding optimization task was not exact. We expect that usage of different metric combining with exact geometric optimization may yield better performance of the associative-memory clustering. Development of

appropriate techniques of high-dimensional optimization is a subject of the future work.

The proposed method may also be generalized for wider class of manifolds. In this case we should use geometric computation that works for arbitrary manifold (e.g. described in [6]). This extension of associative-clustering technique will enable to solve wider class of tasks.

References

[1] L. Personnaz, I. Guyon, G. Dreyfus, "Collective computational properties of neural networks: New learning mechanisms," *Phys. Rev. A*, Vol.34 (5), pp. 4217-4228, 1986

[2] Reznik A.M Non-Iterative Learning for Neural Networks. *Proceedings of the International Joint Conference on Neural Networks.* (Washington DC), July 10-16, 1999

[3] A.S. Sitchov. Methods of Improvement of Neural Associative Memory and its Application to Hybrid Modular Neural Networks. Ph. D. thesis, IMMSP of NAS of Ukraine, Kyiv, Ukraine 2003 (in Ukrainian)

[4] Kohonen, Teuvo *Self-organizing maps.* Third edition. Springer Series in Information Sciences, 30. Springer-Verlag Berlin, 2001.

[5] P.-A. Absil, R. Mahony, R. Sepulchre. Riemannian geometry of Grassmann manifolds with a view on algorithmic computation, *Acta Applicandae Mathematicae*, Vol. 80, No 2, pp. 199–220, Jan. 2004

[6] J.-P. Dedieu, D. Nowicki, Symplectic Methods for the Approximation of the Exponential Map and the Newton Iteration on Riemannian Submanifolds. Submitted to the *Journal of Complexity* (2004)