Clustering with Spiking Neurons

Irit Opher* David Horn†
School of Physics and Astronomy
Raymond and Beverly Sackler Faculty of Exact Sciences
Tel Aviv University, Tel Aviv 69978, Israel

Brigitte Quenet‡
Laboratoire d'Electronique
E.S.P.C.I., 75005 Paris, France

Abstract

We present a novel neural method for data clustering using temporal segmentation of spiking neurons. Our clustering algorithm relies only on distances between data points. Each point is associated with a neuron, and the distances are used to determine the synaptic weights between those neurons. The dynamical development of this system leads to synchronous firing of neurons that belong to the same cluster, while different clusters fire at different times. Such dynamic behavior is called temporal segmentation. It is achieved via two mechanisms - intra cluster synchrony, induced by excitatory connections within each cluster, and desynchronization between clusters induced by inhibitory competition. We test our clustering method on the iris data set. For problems that do not have a unique clustering solution, we construct a pair-correlation matrix on the basis of multiple clustering solutions. By performing a second clustering algorithm, this time on the pair-correlation matrix, we are able to define second order clusters of the original distance matrix. This method is demonstrated on a biological data set.

1 Introduction

Analyzing large sets of data has become an important task in many scientific research areas. This inevitably involves clustering [4], which becomes a difficult task when the data space is high dimensional, i.e. when the number of features that characterize each point is very large. In such situations, it might be useful to rely solely on the distances between the data points. This is the basis of many clustering methods that use these distances as a dissimilarity measure. Under the assumption that points within a cluster are closer to one another than to points in other clusters, clustering becomes equivalent to searching for groups of points where the distances within each group (cluster) are much smaller than the distances between points in different clusters.

In this paper we present a non parametric approach to data clustering, based on an analogy between data points and a neural system. We associate the data points with spiking neurons, and use the distances to determine the synaptic interactions between these neurons. We show how temporal properties of our neural system, namely segmentation and binding, can be used to perform clustering. Synchrony is believed to be a powerful binding mechanism in animal brains. This inspires us to employ a temporal tagging device as the means of clustering. We demonstrate how a system of integrate-and-fire ($I\&F$) neurons can be used to perform a clustering task, relying on the fact that coupled $I\&F$ neural systems can exhibit staggered oscillations of neuronal cell assemblies. These assemblies are defined through the synchrony of their neurons, and we use them to represent clusters. As a result, neurons that belong to the same cluster fire together, while neurons that belong to different clusters are temporally segmented. Thus, some of the

*irito@post.tau.ac.il
†horn@neuron.tau.ac.il
‡Brigitte.Quenet@espci.fr

topology of the data space is observed. We apply our clustering procedure to the well known iris data[5], and to a biological data set[2].

Whereas for the first data-base our method leads to unique clustering solutions, this is not the case for the second data base. In the latter we find many possible clustering solutions, hence we have to employ a second clustering algorithm in order to obtain a clear cut classification of the data set. To do that, we consider the pair-wise correlations of neurons in many different clustering solutions to the same problem. These correlations can serve as a measure of the similarity between data points. Using these correlations we define a distance matrix that forms the basis for the second order clustering problem. The resulting clusters reveal new structure in data space.

2 Interacting $I\&F$ Neurons

Our analysis is performed by a set of continuous integrate-and-fire (CIF) neurons for which we employ a description [6] that uses two dynamic variables. When driven by strong external input $I = I^{\text{ext}}$, our CIF neuron is a nonlinear oscillator. The details of the single neuron dynamics are specified in [6].

We consider an array of such neurons interacting with each other via constant synaptic couplings W_{ij}. This means that in addition to the external input, each neuron receives internal input that results from the firings of other neurons:

$$I_i(t) = I_i^{\text{ext}} + \Sigma_j W_{ij} f_j(t) \tag{1}$$

where f_j denotes the spike of neuron j. Note that these pulse-coupled interactions do not undergo any delay. Excitatory interactions in a system like that lead to synchrony among the firing neurons, whereas inhibition leads to desynchrony. This is the property that we rely on when we perform data clustering [7].

3 The Clustering Method

Our clustering method[7] is based on distances between the points of the data set. If the feature space is not Euclidean, we have to deal with the problem of defining the relative scaling of the various axes and the suitable metric to be used for definition of the distance. One may use the city-block metric, a Euclidean metric or a generalized Minkowski distance [12]. For measuring distances between clouds of points with different variance, it is sometimes helpful to use the Mahalanobis distance[8]. The calculation of the latter is done after performing a whitening similarity transformation, leading to a trivial correlation matrix in the new basis.

Another difficulty arises when the various features differ by orders of magnitude (e.g. meters and nanometers). This can be overcome by normalization of the data features. In our applications we saw that the specific choice of normalization had almost no effect on our clustering results. Therefore, unless otherwise noted, we restrict ourselves to simply calculating the Euclidean distance between data points whose coordinates lie between 0 and 1.

From now on, we will assume that the data we analyze are characterized by a distance matrix D_{ij} between data points i and j. We will associate each data point with one of our neurons, and choose the coupling W_{ij} to be a decreasing function of D_{ij}, e.g.

$$W_{ij} = \frac{a - D_{ij}}{b}, \quad W_{ii} = 0 \tag{2}$$

For example, one may take a and b as the average value and standard deviation of the distance matrix, respectively. This standard choice will assign positive synaptic weights between neurons whose distance is less than the mean value of the distance matrix, and negative weights otherwise, while maintaining the values of the weights within certain limits.

Under these conditions, neurons that are close to one another tend to fire synchronously, thus defining a cluster. For better separation, we induce competition between the clusters using global inhibition proportional to the total activity:

$$I_i(t) = I_i^{\text{ext}} + \Sigma_j W_{ij} f_j(t) - \gamma \Sigma_j f_j(t) \tag{3}$$

Taking the external input I_i^{ext} to be the same for all neurons, the system generally converges after a short while onto a periodic solution, where the different clusters fire at different times, thus achieving temporal segmentation.

3.1 The Iris data

As first demonstration of our technique we apply it to the iris data set[5], that consists of 150 data points belonging to 3 groups of equal size. The groups are 3 species of iris, and each of the data points is described by 4 valued features: petal-length, petal-width, sepal-length, and sepal-width.

Using synaptic weights of the form of eq. 2 and running the dynamics of eq. 3 we were able to classify correctly 85% of the data points. Better results can be obtained by a slight modification of the clustering procedure. We impose a normalization condition, rescaling the synaptic weights so that $\Sigma_j W_{ij}$ is constant. Moreover, we impose such normalization conditions separately on the excitatory and on the inhibitory connections. In addition, we limit each neuron to one spike per cycle, where a cycle is defined by the overall period of the system. This requires disregarding the spike of a neuron unless all others have fired since its last spike. Under these conditions, our neural simulation turns into an algorithm that classifies correctly 90% of the iris data points. The results are shown in Fig. 1.

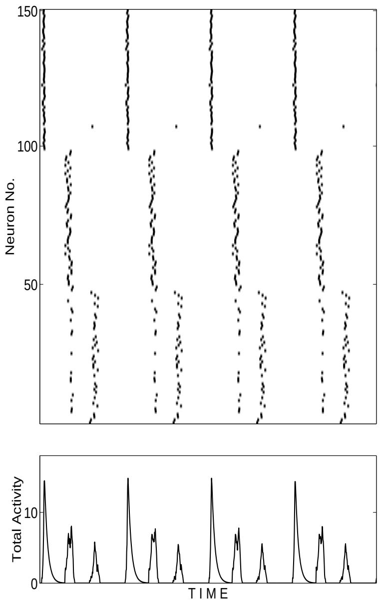

Figure 1. Results of our clustering algorithm for the iris data set. The sum of weights that each neuron receives is the same, and a neuron is not allowed to fire twice in the same cycle. The bottom frame displays the total spiking activity of the system while the upper frame shows the activity of all 150 neurons. The neurons are ordered according to their species, so that neurons 1-50 belong to one species, etc.

In the lower frame we see that the periodic total activity of the system has three different peaks, the phenomenon we refer to as temporal segmentation. The groups of neurons belonging to the three peaks define the three clusters of the data. The segregated activity of the three clusters is shown in the upper frame as a raster plot. Note that each peak in the total activity corresponds to the activation of one of the clusters.

It is interesting to note that our results are comparable with other powerful clustering algorithms. Our success rate of 90% is an improvement over the 83% of [3] (who use an analogy of data points with magnetic spins) but falls short of the 98% success rate of the geometric neurons of Lipson and Siegelmann[10].

4 Second Order Clustering

The answer to a clustering problem is not always unique. This often happens when one studies biological or psychological data where the features of the data points are not embedded easily into a continuous metric space. Moreover, the topology of the data space may not reduce to a hierarchical order of clusters, where one can divide the data into a few large subsets that are composed of a few subsets themselves, and so on. Rather, the data space may be composed of many overlapping clusters. Such a situation is manifested in our system by the existence of many cluster candidates to the same problem. These solutions may even be contradictory, i.e. a particular pair ij of two elements may sometimes appear in the same cluster and sometimes in two different clusters. The number of clusters may change as well. An example is shown in Table 1, where the 16 most frequent cluster solutions of the bee data set[2], to be explained below, are presented.

In general, multiplicity of cluster solutions does not mean that there is no structure in the data space. The individual solutions need to be examined in order to extract some information regarding that structure. We suggest the following analysis that results in clear cut classification of the data set.

Our method relies on first generating many clustering candidates by varying over the parameters a, b and γ in Eqs. 2 and 3. We then construct a correlation matrix

$$C_{ij} = \frac{1}{M} \sum_{m=1}^{M} X_{ij}^{m} \qquad (4)$$

where the factor X_{ij}^m is 1, if i and j appear in the same cluster in the solution m, and is zero otherwise. M is the number of cluster solutions that were produced. We can regard the elements C_{ij} as a similarity measure between neurons i and j. Since our clustering procedure is applied on a dissimilarity matrix, it would be better to define a distance matrix, based on the correlations in C. This can be done by defining $\delta_{ij}^m = 1 - X_{ij}^m$. Note that the triangular inequality is satisfied by the δ's. Next we define distance elements by averaging over all solutions:

$$d_{ij} = \frac{1}{M} \sum_{m=1}^{M} \delta_{ij}^m. \tag{5}$$

Clearly, the elements of d, that are the sums of δ's obey the triangular inequality:

$$d_{ij} \leq d_{ik} + d_{kj} \;\; \forall i, j, k. \tag{6}$$

Therefore, d can be regarded as a new distance matrix which may be used for performing a second clustering procedure. The clusters of the correlation-based distance matrix are then to be regarded as second order clusters of the original distance matrix D.

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	1	1	1	2	2	2	3	3	1	3	3	2	1
2	1	2	1	1	1	3	3	3	2	3	2	1	1
3	1	2	1	3	3	3	2	2	1	2	2	1	1
4	1	1	1	2	2	2	3	3	3	3	3	2	1
5	1	1	1	2	2	2	3	3	1	3	3	3	1
6	1	1	1	2	2	2	3	3	3	1	3	2	1
7	1	2	1	1	1	3	2	2	3	3	3	3	1
8	1	2	1	1	1	2	3	3	3	2	3	2	1
9	1	2	1	3	3	2	2	2	2	2	3	3	1
10	1	2	1	3	3	3	2	1	1	2	2	2	1
11	1	1	1	2	2	2	3	3	3	3	3	1	1
12	1	2	1	3	3	2	2	2	2	2	2	3	1
13	1	2	1	1	1	2	3	3	3	2	3	1	1
14	1	2	1	3	3	3	2	2	1	2	1	3	1
15	1	2	1	1	1	3	2	2	3	3	3	2	1
16	1	2	1	1	1	2	2	3	3	2	3	3	1

Table 1: Each line describes one clustering candidate solution of the bee data set. There are 13 elements in this problem, and each is assigned one of three clusters in every one of the 16 candidate solutions shown here. The cluster to which the first element belongs is always assigned the number 1.

It is interesting to note that the leading eigenvectors of C can serve as a geometrical representation of some of the structure of the data set. In the next section, where we discuss the bee data set, it becomes obvious that whereas no apparent structure of D is observed by a multi-dimensional scaling technique, some structure of C comes out from an analogous two-dimensional display while a clear-cut classification is obtained by the second clustering algorithm performed on the new distance matrix d.

5 Analysis of the Bee Data Set

We apply our second-order clustering method to a data set resulting from the analysis of the cuticular hydrocarbon profiles of honeybees extracted from the same colony and belonging to 13 subfamilies. The cuticular hydrocarbon profiles have been shown to be genetically based: in a given colony, two workers that are full sisters (they are said to belong to the same 'subfamily') have significantly closer profiles than half sisters[1]. Each subfamily can be labeled by a profile averaged over its members and represented by a normalized vector in the 22 dimensional space of the chemical compounds of a profile. The following question arises: can the subfamilies of a given colony be gathered into different classes according to their distances in profile space?[2]

There exist 13 subfamilies in this problem. They were identified by using the Mahalanobis distance with a metric governed by the mean intra-subfamily covariance matrix weighted by the number of cases in each subfamily (see for instance [11]). This calculation led to a 13×13 matrix of distances D_{ij} between the 13 subfamilies which are the elements of our clustering problem. To envisage the structure that such a matrix implies one can apply a conventional multi-dimensional scaling technique such as classical scaling[8] (also known as metric scaling). This method is based on the assumption that the distances can be derived from an embedding of the N data points in an N-1 dimensional Euclidean space, i.e. all distances obey the triangular inequality. One may then define a positive semi-definite matrix F by performing the following transformations on the distance matrix D:

$$e_{ij} = -0.5(D_{ij})^2 \tag{7}$$

$$F_{ij} = e_{ij} - \langle e_{il} \rangle_l - \langle e_{lj} \rangle_l + \langle e_{lk} \rangle_{lk} \tag{8}$$

where $\langle \rangle_l$ means average on l. The eigenvectors of F can serve as principal coordinates. In our case, there

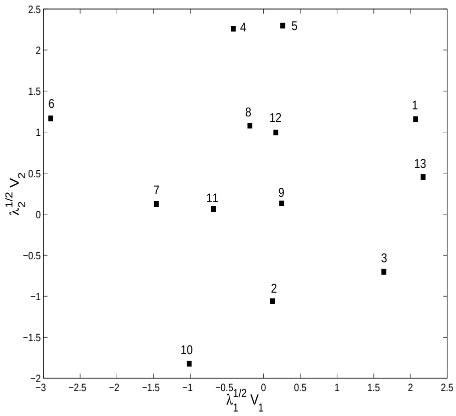

Figure 2. Results of classical scaling performed on the bee distance matrix. The 13 elements are spread throughout the 2-dimensional space.

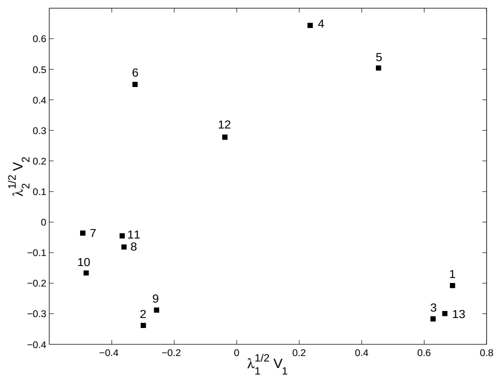

Figure 3. A two dimensional projection of the correlation matrix of the 13 elements of the bee data set reveals some clustering structure.

are two large eigenvalues (whose sum is half the sum of all eigenvalues). Taking $\sqrt{\lambda_j}V_j$ as the jth axis, for $j = 1, 2$ results in a diffuse configuration of the 13 elements, with no evident structure, as can be seen in Fig. 2. This rather homogeneous distribution of data points must be the reason for the many possible clustering solutions of the type shown in Table 1. We proceed now to show how more structure can be extracted from these data. Using our clustering algorithm we obtained many possible classifications. Table 1 shows the 16 most frequent ones, all of which correspond to 3 cluster classifications. 4 and 5 cluster classifications occur as well but they are relatively rare. We have used a large set of such solutions (860 different clustering candidates in 1781 solutions) to construct the 13×13 correlation matrix of Eq. 4. This matrix turns out to have three large eigenvalues. The eigenvector with the highest eigenvalue, $\lambda_0 = 4.47$, has roughly identical contributions from all 13 elements, since it points to the mean of the distribution of pairs. Hence it is not helpful in finding the structure we are after. However, the next two eigenvectors, λ_1 and λ_2, whose sum is approximately equal to the sum of all remaining 10 eigenvalues, can serve such a purpose. These eigenvectors are identical with the two highest eigenvectors of the covariance matrix, in which the mean is eliminated. They span the two-dimensional space of Fig. 3.

Fig. 3 displays quite clearly some of the clustering structure of the correlation matrix. It is important to note that this structure could not have been extracted directly from the original distance matrix. It took the clustering procedure, and the correlation matrix based on the clustering outcome, to generate this result. We may identify in Fig. 3 some structure that is also evident in Table 1. Thus note that the points 1, 3 and 13 are very close to one another, reflecting the fact that they belong to the same cluster. In Table 1 these elements always appear together. The slight separation between them in Fig. 3 results from the existence of other solutions that we take into account in constructing the matrix C, where these three elements are not always grouped into the same cluster.

At this stage we perform the second clustering algorithm on the matrix d itself. The result, which is robust under changes of a and b, resembles the partial representation of Fig. 3. We find the following clusters: A=[1,3,4,5,13], B=[2,7,8,9,10,11] and C=[6,12]. One may wonder to what extent this method really describes structure existing in the data. To demonstrate that it does, we employ a distance criterion that can serve as the means for defining a cluster partitioning: The average distance between two clusters (defined by averaging over all distances between points in the two clusters) has to be larger than the average distance between the elements in each cluster[1]. The partitioning of the data into groups A B and C obeys this rule. Hence we conclude that it describes true structure that is not evident unless second-order clustering is performed.

[1] Minimization of this criterion can serve as the basis for a clustering algorithm. However, when the number of data points is large, this calculation becomes impractical.

6 Summary

We have shown that a simple model of CIF neurons can serve as a tool for data clustering. Clustering relies on dynamical synchronization properties of spiking neurons: Intra cluster synchrony can be induced by instantaneous excitatory connections, while inter cluster desynchronization is induced by global instantaneous inhibition. This method can be applied to data points that are embedded in high dimensional spaces, since it uses only the distances between the points. Specifying guidelines for determining the weights, our method does not require problem specific preprocessing. Fast convergence of systems of spiking neurons allows us to use this clustering method for problems of arbitrary size. Imposing a constraint so that each neuron is allowed to fire only once in a cycle, and using normalization constraints on the weights, we considerably improved the results of our clustering procedure.

This neural clustering method can be generalized to perform image analysis. For this purpose we need to embed our CIF neurons in a two-dimensional surface, so that each neuron corresponds to one pixel in the image. Identifying objects in the image with clusters, the task of segmenting an image into its components may be identified with clustering. The relevant dimensions in this problem are defined by the two-dimensions of the surface and the grey scale of the pixels. In other words, pixels that lie close to one another and have very similar grey scales are to be associated with the same object, or cluster. In [7] we have demonstrated how image analysis can be performed by our neural system. Other algorithms for image segmentation that rely on neuronal synchrony exist in the literature. They include LEGION [13] and the PCNN method [9].

When the clustering procedure does not lead to a unique solution we propose a method that allows us to uncover some structure in the data and define second order clusters. This method can be described as performing two consecutive clustering algorithms. The first one, clustering of the distance matrix D, leads to many (contradictory) clustering solutions. They serve as the basis for defining a pairwise correlation matrix C, out of which a new distance matrix d is calculated. Then, the second clustering procedure may be performed. Using this second procedure we were able to uncover structure that was not observable otherwise.

Acknowledgment

We thank G. Arnold for providing the bee data. This research was partially supported by the Israel Science Foundation.

References

[1] Arnold, G., Quenet, B., Cornuet J. M., Masson, C., De Schepper, B., Estoup, A. & Gasqui, P., 1996. Kin recognition in honeybees.*Nature* 379, 498.

[2] Arnold, G., Quenet B., de Shepper B., Masson C., 1998. Role of the hive and social environment in the ontogeny (evolution) of the subfamily cuticular hydrocarbon profiles in the honeybee. Preprint.

[3] Blatt, M., Wiseman, S. and Domany, E., 1997. Data clustering using a model granular magnet. *Neural Computation* 9, 1805–1842.

[4] Duda, R. O. and Hart, P. E., 1973. *Pattern classification and scene analysis.* New York, Wiley-Interscience.

[5] Fisher, R. A., 1936. The use of multiple measurements in taxonomic problems. *Ann. Eugen.* 7, 179–184.

[6] Horn, D. and Opher, I., 1997. Solitary waves of integrate and fire neural fields *Neural Comp.* 9, 1805–1818.

[7] Horn, D. and Opher, I., 1999. Collective Excitation Phenomena and their Applications. In: *Pulsed Neural Networks*, Eds: W. Maass and C. B. Bishop, MIT Press, 297–316.

[8] Krzanowski, W.J., 1988. *Principles of Multivariate Analysis* Clarendon Press, Oxford.

[9] Lindblad, Th. and Kinser, J. M. *Image Processing using Pulse-Coupled Neural Networks* Springer, 1998.

[10] Lipson, H. and Siegelmann, H.T., 1998. Clustering irregular shapes using high-order neurons. submitted.

[11] Manly, B. F. J., 1986. *Multivariate statistical methods* Chapman & Hall.

[12] Schiffman, S.S., Reynolds, M.L. and Young, F.W. 1981. *Introduction to Multidimensional Scaling*, Academic press.

[13] Wang, D. and Terman, D. (1997). Image Segmentation Based on Oscillatory Correlation. *Neural Comp.* 9. 805–836; Err: 9, 1623–1626.