COMPLEX DATA CLUSTERING: FROM NEURAL NETWORK ARCHITECTURE TO THEORY AND APPLICATIONS OF NONLINEAR DYNAMICS OF PATTERN RECOGNITION *

GUOJUN GAN

Research and Development, Global Variable Annuity Hedging,
Manulife Financial, Canada
Email: Guojun_Gan@manulife.com

JIALUN YIN

National University of Defense Technology
Changsha, Hunan, China
E-mail: genial@126.com

YULIA WANG AND JIANHONG WU†

Laboratory for Industrial and Applied Mathematics, York University,
Toronto, Ontario, M3J 1P3, Canada
Centre for Disease Modelling, York Institute of Health Research,
Toronto, Ontario, M3J 1P3, Canada
E-mail: wujh@mathstat.yorku.ca, terry_312@hotmail.com

*This work at the Laboratory for Industrial and Applied Mathematics on the theoretical foundation and applications of projected clustering of high dimensional and big data has been supported by a number of programs and funding agencies including the Canada Research Chairs program, the Natural Sciences and Engineering Research Council of Canada (discovery grant, collaborative research development program), the Mitacs globalink program, the Mitacs accelerate program, the Fields-Mitacs summer research program, the Canada Foundation for Innovation and the Ontario Innovation Trust. A few industrial partners have also been involved, these partners include the Generation 5 Mathematical Technologies Inc. and the InferSystems Corporation.
†Corresponding author.

We present some progress in high dimensional data clustering, made at the Laboratory for Industrial and Applied Mathematics over the last ten years. The focus is on the role of information processing delay as an adaptive mechanism for pattern recognition in subspaces of high dimensional data. Our objective is to develop both mathematical foundation and effective techniques/tools for projective clustering. We also present some applications to gene filtering, cancer diagnosis, neural spike trains pattern recognition, text mining, stock associations, and online social network news aggregation.

1. Introduction

The purpose of this survey is to organize a few PhD theses and MSc dissertations, research publications, and projects conducted at the York University's Laboratory for Industrial and Applied Mathematics (LIAM) in a coherent framework about information processing delay, high dimension data clustering, and nonlinear neural dynamics.

The objective of this decade long effort at LIAM is to develop both mathematical foundation and effective techniques/tools for pattern recognition in high dimensional data. We refer to the monograph[21] for our collection of existing clustering algorithms, and the survey paper[44] for a heuristic description of our philosophy that the nonlinear dynamic systems theory may provide some theoretical foundation and principles based on recent biological evidences for novel neural network inspired clustering architectures.

In the papers[9,10] and in the thesis by Cao[8], we developed a novel neural network architecture and algorithm to detect low dimensional patterns in a high dimensional data set. These patterns are associated with the projective clusters introduced by Aggarwal and his co-workers from the IBM Watson Centre[2]. The developed projective adaptive resonance theory (PART) has received much attention by data clustering researcher community and industry, and formed the core of a Collaborative Research Development project funded by the Natural Science and Engineering Research Council of Canada (NSERC) in collaboration with Generation 5 Mathematical Technologies Inc. The PART algorithm has since been used in a number of applications. For example, it was used to develop a powerful gene filtering and cancer diagnosis method in [39], which shows that PART was superior for gene screening. As will be documented in later sections, the PART was also used for clustering neural spiking trains, ontology construction, stock associations, and online social network news aggregation. The PART algorithm was further extended to deal with categorical data in the thesis [20],

and with supervised clustering in the dissertations [36,28].

The PART architecture is based on the well known ART developed by Carpenter and Grossberg, with a selective output signalling (SOS) mechanism to deal with the inherent sparsity in the full space of the data points in order to focus on dimensions where information can be found. The key feature of the PART network is a hidden layer of neurons which incorporates SOS to calculate the dissimilarity between the output of a given input neuron with the corresponding component of the template (statistical mean) of a candidate cluster neuron and to allow the signal to be transmitted to the cluster neuron only when the similarity measure is sufficiently large. Recently discovered physiological properties of the nervous system, the adaptability of transmission time delays and the signal losses that necessarily arises in the presence of transmission delay, enabled us to interpret SOS as a plausible mechanism from the self-organized adaptation of transmission delays driven by the aforementioned dissimilarity. The result is a novel clustering network, termed PART-D, with physiological evidence from living neural network and rigorous mathematical proof of exceptional computational performance. This clustering network was developed in [45].

Such an adaptation can be regarded as a consequence of the Hebbian learning law, and the dynamic adaptation can be modelled by a nonlinear differential equation. As a result, we obtained a new class of multi-scale systems of delay differential equations with adaptive delay. A key issue then is how to analytically formulate the delay adaptation. This links to another PhD thesis by Beamish[3], which proposed an alternative neural network formulation of the Fitts' law for the speed-accuracy trade-off of information processing. A number of publications have been resulted from this thesis work, including [5,6,4,7]. It remains an open problem how to use this alternative neurodynamical formulation to obtain a precise delay adaption rule of the PART-D neural network architecture for projective clustering.

When the delay adaption rates are in certain ranges, we anticipate nonlinear oscillatory behaviors of the PART-D neural network as the signal processing delay has been recognized as a major mechanism for nonlinear oscillation in the form of Hopf bifurcations, and this oscillation slows down the convergence of the clustering algorithm. How to detect the birth and to describe the global persistence of these nonlinear oscillations is the central subject of the thesis by Hu[29] and the subsequent publications[32,31,30].

In summary, there have been increasing physiological evidences to support the idea of projective clustering using neural networks with delay adap-

tion, there has been some preliminary theoretical analysis to show why such a network architecture works well for high dimensional data, and there have been sufficient applications to illustrate our PART network based clustering algorithm is efficient. An interdisciplinary approach for high dimensional data clustering clearly shows the potential to develop a dynamical system framework for pattern recognition in high dimensional data.

2. Clustering and clustering neural networks

Data clustering, a common cognitive task effectively performed by our central nervous system routinely, becomes increasingly important and challenge in today's "big data" reality. It aims to finding certain homogeneous patterns in data sets containing many heterogeneous structures. The goal of data clustering is to reorganize subsets of data points into groups, called clusters, so that the data points within the same group share some common features while points in different clusters are distinguished by some of these common features. In unsupervised clustering, these features have to be identified during the process of clustering.

The approach taken by the Laboratory for Industrial and Applied Mathematics (LIAM) towards data clustering problem is to consider the clustering process as an *inverse process* of *pattern formation* of complex dynamical systems. In this approach, the goal for clustering a data is to *construct a dynamical system* to *automatically and adaptively* identify patterns hidden in the given data set. Namely, for a given data set D in R^m, we try to construct a dynamical system with data-specified local attractors (such as equilibria or periodic orbits) $CR_1, \cdots, CR_n$ so that each CR_i represents a cluster (for example, the centre of a cluster) and its domains of attraction gives the cluster criterion that distinguishes this cluster from others.

There have been a few effective projected clustering algorithms developed such as CLIQUE and PROCLUS, see [44,21] and references therein. Here we describe a neural dynamics inspired architecture with which a dynamical system is constructed from adaptively processing a high dimension data. A key issue is what constitutes of the minimal size and structure of a nonlinear dynamical system required to identify the clusters hidden in arbitrary unknown subspaces of the given data set.

We consider a given data set D of n points in the m dimensional Euclidean space. If we try to mimic the clustering functioning of our central nervous system to construct a network of neurons to identify the hidden patterns of the data set, we will need:

- a layer of m neurons to process inputs (the input layer);
- a layer of neurons to represent clusters (the clustering layer), with the number of clustering neurons unspecified;
- synaptic connections (bottom-up weights) between the input layer and the clustering layer to weight appropriately the output (activation) of each input neuron so that every clustering neuron can calculate the sum of weighted activations generated by a given input vector for the purpose of selecting a candidate cluster;
- synaptic connections among clustering neurons so that this layer of neurons can automatically select a winner as the candidate cluster;
- a mechanism to update the feature (statistical mean) of the selected cluster neuron and to store the updated feature at the synaptic connections (top-down weights, or templates);
- a mechanism and learning rule with which the top-down weights and bottom-up weights are updated to learn the experience.

Specific principles for the connection topology (competitive network) of the clustering layer and for the learning rules to update top-down and bottom-up weights have led to the renowned ART (Adaptive Resonance Theory) neural networks, which have been shown to be very effective in self-organized clustering in *full dimensional spaces*. ART was first introduced by Grossberg in 1976[26,27] in order to analyze how brain networks can learn in real time about a changing world in a rapid but stable fashion, based on which Capenter and Grossberg[14,13,16] developed two classes of ART neural network architectures ART1 and ART2, whose computational performance (dynamics) is described by systems of differential equations. ART1 self-organizes recognition categories for arbitrary sequences of binary input patterns, while ART2 does the same for either binary or continuous inputs. Some other classes of ART neural network architectures such as Fuzzy ART[12], ARTMAP[11], Fuzzy ARTMAP[17], and Gaussian ARTMAP[43] were then developed with increasingly powerful learning and patten recognition capabilities in either an unsupervised or a supervised mode.

Examples have been provided in Cao-Wu [9] to show that the ART neural network needs additional structure in order to perform the task of subspace clustering in high dimensional data sets since ART focuses on similarity of patterns in the *full* dimensional space. The first paper [9] of a series of studies introduces a new mechanism to deal with the identification of subspaces where clusters are formed, this is the so-called *selective output signaling* (SOS in short) and the corresponding ART is termed PART. This SOS

mechanism selectively selects the signal from an input neuron only when the signal is similar to the top-down weight (template) between the input neuron and the targeted clustering neuron, hence PART focuses on only those dimensions where information is relevant for a particular cluster. We refer to [9] for a schematic illustration of the PART architecture.

3. Projected ART with Adaptive Delay

Cao and Wu [9,10] implemented PART and demonstrated that PART networks outperform ART networks for pattern recognition in high dimensional spaces. The key feature of a PART network is a hidden layer which incorporates the SOS mechanism to calculate the similarity between the output of a given input neuron with the corresponding component of the template of a candidate cluster neuron and allows the signal to be selectively transmitted to the cluster neuron only when the similarity measure is sufficiently large. So, in PART the output signal of an input neuron will be completely prohibited to be transmitted to its target cluster neuron if the similarity measure is small although in practice, this output signal may still play a (relatively minor) role in the final clustering result. This issue has been successfully addressed by the novel clustering network, termed PART–D, which interprets the SOS mechanism in terms of two recently emphasized properties of the nervous system, namely the adaptability of transmission time delays and the signal losses that necessarily arises in the presence of transmission delay. In PART-D, the SOS mechanism is shown to arise because the self-organized adaptation of transmission delays is driven by the dissimilarity between the input pattern and the stored pattern (represented by the template of a cluster neuron). Such an adaptation can be regarded as a consequence of the Hebbian learning law, and the dynamic adaptation can be modeled by a nonlinear differential equation. As a result, we obtain a new class of systems of delay differential equations with adaptive delay as follows:

$$\epsilon_p \frac{dx_i(t)}{dt} = -x_i(t) + I_i(t), \tag{1}$$

$$\epsilon_c \frac{dy_j(t)}{dt} = -y_j(t) + [1 - Ay_j(t)][f_c(y_j(t)) + T_j(t)] \tag{2}$$

$$- [B + Cy_j(t)] \sum_{k \neq j, k \in \Lambda_2} f_c(y_k(t)), \tag{3}$$

$$T_j(t) = D \sum_{1 \leq i \leq m} z_{ij}(t) f_p(x_i(t - \tau_{ij}(t)) e^{-\alpha \tau_{ij}(t)}, \tag{4}$$

$$\beta \frac{d\tau_{ij}(t)}{dt} = -\tau_{ij}(t) + E[1 - h_{ij}(t)], \tag{5}$$

$$h_{ij}(t) = h_\sigma(d(f_p(x_i(t)), w_{ji}(t)))l_\theta(z_{ij}(t)), \tag{6}$$

$$\delta \frac{dz_{ij}(t)}{dt} = f_c(y_j(t))[(1 - z_{ij}(t))Lf_p(x_i(t - \tau_{ij}(t))e^{-\alpha\tau_{ij}(t)}, \tag{7}$$

$$- z_{ij}(t) \sum_{k \neq i, k \in \Lambda_1} f_p(x_k(t - \tau_{kj}(t)e^{-\alpha\tau_{kj}(t)}], \tag{8}$$

$$\gamma \frac{dw_{ji}(t)}{dt} = f_c(y_j(t))[-w_{ji}(t) + f_p(x_i(t - \tau_{ij}(t))e^{-\alpha\tau_{ij}(t)})]. \tag{9}$$

In the above model, the activation of the i-th inout neuron is denoted by x_i, the activation of the j-th clustering neuron is denoted by y_j; the bottom-up weight between the i-th input neuron and the j-th clustering neuron is denoted by z_{ij}, while the top-down weight is denoted by w_{ji}.

In the Short Term Memory trace equations for input neurons, $0 < \epsilon_p \ll 1$, I_i is the constant input imposed on the i-th neuron. This is based on the assumption that for an isolated neuron, the dynamics is the balance of the internal decay and the external input excitation. For the change of the Short Term Memory trace equations for clustering neurons, we assume that the activation of the cluster neuron depends on the internal decay, the excitation from self-feedback, the inhibition from other cluster neurons and the excitation by the bottom-up filter inputs from input neurons. In the equations, $0 < \epsilon_c \ll 1$, $f_c : R \to R$ is a signal function, A, B, and C are non-negative constants. In the the bottom-up filter input T_j calculation, D is a scaling constant, and $f_p : R \to R$ is the signal function of the input layer. It is assumed the signal transmissions between two layers are not instantaneous and the signal decays exponentially at a rate $1/\alpha > 0$.

The term τ_{ij} is the signal transmission delay between the ith input neuron and the j-th clustering neuron. We assume this delay is driven by the dissimilarity in the sense that the signal processing from the input neuron to the cluster neuron is faster when the output is similar to the corresponding component of w_{ji} of the feature vector $w_j = (w_{ji})_{1 \leq i \leq m}$ of the cluster neuron. In the equation for the delay adaptation, $\beta > 0$, $E \in (0, 1)$ are constants and $h_{ij}(t) = S(d(f_p(x_i(t)), w_{ji}(t)), z_{ij}(t))$ is the similarity measure between the output signal $f_p(x_i(t))$ and the corresponding component $w_{ji}(t)$ of the feature vector of the cluster neuron, with respect to the significance factor of the bottom-up synaptic weight $z_{ij}(t)$, here d is the usual distance function in the one dimensional Euclidean space and $S : R^+ \times [0, 1] \to [0, 1]$ is a given function, non-increasing with respect to

the first argument and non-decreasing with respect to the second argument. Moreover, $S(0,1) = 1$ (The similarity measure is 1 with complete similarity and maximal synaptic bottom-up weight) and $S(+\infty, z) = S(x, 0) = 0$ for all $z \in [0,1]$ and $x \in R^+ := [0, \infty)$ (The similarity measure is 0 with complete dissimilarity or minimal bottom-up synaptic weight). In the above formulation, we used some special function of S where $h_{ij}(t)$ is determined by the distance between the output signal $f_p(x_i(t))$ and the corresponding component $w_{ji}(t)$ of the feature vector of the cluster neuron, multiplied by the significance factor of the bottom-up synaptic weight $z_{ij}(t)$, with a threshold parameter $\theta > 0$.

The equation governing the change of the weights follows from the synaptic conservation rule of [41] and only connections to activated neurons are modified. The top-down weights are modified so that the template will point to the direction of the delayed and exponentially decayed outputs from the input layer (with the exponential decay rate $\gamma > 0$). The bottom-up weights are changed according to the competitive learning law and Weber Law Rule that says that LTM (Long Term Memory) size should vary inversely with input pattern scale to present a clustering neuron that has learned a particular pattern from also coding every superset pattern (see [15]). In the equation, $0 < \delta \ll \gamma = O(1)$ and $L > 0$ is a given constant.

We refer to [9] for the equations of the LTM equations for non-committed candidate neurons and the discussion of a reset mechanism. In particular, a candidate (active) node will be reset if at any given time $t \geq 0$, the degree of match is less than a prescribed vigilance. Namely, reset occurs if and only if $\sum_{1 \leq i \leq m} h_{ij}(t) < \rho$, here $\rho \in \{1, 2, \ldots, m\}$ is a vigilance parameter.

The following theorem describes the computational dynamics during a trial.

Theorem 3.1. *We can choose small ϵ_p, ϵ_c, and δ so that:*

(i) *(Inhibition of Non-Candidate Neurons): For $j \neq J$ and $t \geq 0$, $y_j(t) < \eta_c$ and $f_c(y_j(t)) = 0$;*
(ii) *(Sustained Excitation of the Candidate Neuron): There exists $\Gamma > 0$ such that $y_J(t) < \eta_c$ and $f_c(y_J(t)) = 0$ when $t < \Gamma$, and $y_J(t) \geq \eta_c$ and $f_c(y_J(t)) = 1$ when $t \geq \Gamma$;*
(iii) *(Invariance of Similarity): For any $t \geq 0$, $h_{ij}(t) = h_{ij}(0)$;*
(iv) *(Learning at Infinity): For any $j \in \Lambda_c$ with $j \neq J$, $z_{ij}(t)$ and $w_{ji}(t)$ remain unchanged for all $t \geq 0$. But $\lim_{t \to \infty} w_{Ji}(t) = f_p(I_i) e^{-\alpha \tau^*_{iJ}}$*

and

$$\lim_{t\to\infty} z_{iJ}(t) = \begin{cases} 0 & \text{if } h_{iJ}(0) = 0, \\ \frac{L}{L+l_i} & \text{if } h_{iJ}(0) = 1, \end{cases}$$

where $l_i = \#\{k \neq i; h_{kJ}(0) = 1\}$.

Learning may take place in a much faster pace, the following theorem describes the transit computation performance and it also shows an amazing choice of a discrete Lyapunov function that was used to prove both theorems.

Theorem 3.2. *We can choose small* ϵ_p, ϵ_c, *and* δ *so that:*

(v) (Fast Excitation): $\Gamma \in (0, 1)$*;*

(vi) (Fast Learning): Write $z_{ij}^{\epsilon_p,\epsilon_c,\delta}$ *and* $w_{ji}^{\epsilon_p,\epsilon_c,\delta}$ *to indicate explicitly the dependence on* $(\epsilon_p, \epsilon_c, \delta)$*. Then we have (with* $q = 1 - e^{-1/\gamma}$*)*

$$\lim_{\delta\to 0} z_{iJ}^{\epsilon_p,\epsilon_c,\delta}(1) = \begin{cases} 0 & \text{if } h_{iJ}(0) = 0, \\ \frac{L}{L+l_i} & \text{if } h_{iJ}(0) = 1, \end{cases}$$

$$\lim_{\epsilon_p\to 0,\beta\to 0} w_{Ji}^{\epsilon_p,\epsilon_c,\delta}(1) = (1-q)w_{Ji}(0) + qf_p(I_i)e^{-\alpha\tau_{iJ}^*};$$

(vii) (Convergence of Projective Subspace) Let $D_j(t) = \{i; l_\theta(z_{ij}(t)) = 1\}$*. Then, as* $\epsilon_p, \epsilon_c, \delta \to 0$*, we have*

$$D_j(t) = D_j(0) \text{ for any } j \neq J;$$
$$D_J(t_2) \subseteq D_J(t_1) \text{ if } t_2 \geq t_1 \geq 0;$$
$$D_J(t) = D_J(1) \text{ for all } t \geq 1.$$

These theorems describe the computational performance of PART during a trial, either in terms of long-term behaviours or transit behaviours. It confirms that the network does provide the winner-take-all paradigm: all clustering neurons with $j \neq J$ are always inactive, but the J-th clustering neuron will be activated after some finite time. This activated clustering neuron receives the the largest bottom-up filter input (T_J), and (v) shows that the identification of a clustering neuron can take place very fast. The above theorems also give the learning formulae (either long-term or fast learning). Note also that $D_j(t)$ is the set of dimensions of projected subspace associated with cluster representing by the j-th neuron at time t, and above results indicate that the set of dimensions is non-increasing during the learning. This non-increasing property of dimensions contributes to stabilizing learning in response to arbitrary sequences of input patterns.

4. Algorithms and applications

An effective algorithm based on the above results, specially the fast learning rules, has been developed in [9,10] (for PART) and then in [45] for PART-D. These algorithms consist of the following major steps:

- Input Processing and Select Output Signals from Input Layer;
- Activation, Inhibition, and Identification of a Potential Cluster;
- Confirmation, Vigilance and Reset;
- Fast Learning;
- Identification of Subspaces;
- Outliers collection.

The time cost of these algorithms is $O(mnNM)$, where m is the number of dimensions of data space, n is the number of clustering neurons, m is the number of all data points and M is the number of iterations.

4.1. *Experiments on synthetic data*

Extensive simulations on high dimensional synthetic data showed that the clustering layer becomes stable after only a few iterations. Here we describe one example on a high dimensional synthetic data generated via the method introduced by Aggarwal et al [1] in 1999. The input data has 20,000 data points in a 100-dimensional space, which has 6 clusters generated in 20, 24, 17, 13, 16 and 28-dimensional subspaces respectively. The data points are presented in random order, and the clustering results can be reported as number of clusters found, dimensions found, centers of clusters found, and the contingency table of input clusters (original clusters) and output clusters (clusters found).

Output\Input	1	2	3	4	5	6	Sums
1	5144	0	0	0	0	0	5144
2	0	1878	0	0	0	0	1878
3	0	0	4412	0	0	0	4412
4	0	0	0	2716	0	0	2716
5	0	0	0	0	2608	0	2608
6	0	0	1	0	0	1185	1186
Outliers	106	66	239	290	68	287	2056
Sums	5250	1944	4652	3006	2676	1472	20000

The above table shows the simulation results with $\rho = 10$. Note that in the reported results we have treated as outliers the data points in the clustering neurons with very small sizes (less than 1.2% of total data points). The simulation results show that the PART algorithm succeeds in finding the exact number of original clusters and in finding almost exact centers of all original clusters. Note that the dimensions found in different clusters are different, for example, cluster 1 is formed with respect to dimensions $10, 12, 17, 37, 46, 58, 61, 79, 81, 99$, while cluster 2 is formed with respect to dimensions $5, 8, 13, 15, 16, 18, 30, 70, 85, 92$. Also note that the dimensions found are not identical to those of the original clusters (for example, the dimensions of the original cluster 1 include $1, 6, 10, 12, 15, 17, 31, 36, 37, 38, 45, 46, 52, 54, 58, 61, 67, 79, 81, 99$), but these found dimensions are contained as subsets of the associated dimensions of original clusters. These subsets are sufficiently large so that, after a further reassignment procedure, we are able to reproduce the original clusters from the found cluster centers, the found number of clusters and the found dimensions.

4.2. *Application to neural spiking trains clustering*

In [33], PART was used as an effective tool for clustering neural spiking trains via transient behaviors. It was noted that "the detection of nonstationarities in neural spike trains recorded from chronically implanted multielectrode grids, such as transient synchronizations in a neural subpopulation, becomes increasingly difficult as the number of electrodes increases". This calls for unsupervised learning algorithms that can be used to "group, or cluster, spike trains based on the presence of local, shared features". The feature of PART that allows comparisons be made between inputs and learned patterns using a subset of the total number of spikes available enables the network to learn the characteristics that defines each cluster making as few assumptions about the statistical properties of the spike trains as possible.

> "The result is an extremely powerful tool for clustering neural spike trains that is computationally inexpensive. The fact that projective clustering dramatically increases the ability of an artificial neural network to discover patterns in its sensory inputs raises the question of whether analogous mechanisms operate in the nervous system. Thus we anticipate that PART neural networks will not only have increasing applications for data analysis, but also have the

potential to provide insights into the computational activities of the nervous system."

Spike train inputs for the PART neural network have the general form

$$\left(\overbrace{s_{11}, s_{12}, \cdots, s_{1p}}^{bin_1}, \overbrace{s_{21}, s_{22}, \cdots, s_{2p}}^{bin_2}, \cdots, \overbrace{s_{k1}, s_{k2}, \cdots, s_{kp}}^{bin_k} \right),$$

where the dimension, m, is equal to the number of bins, bin_k, of size Δt times the number of statistical features of interest, and the notation s_{kp} denotes the p–th statistical feature evaluated for the k–th bin. The number of input neurons is m and the number of clustering neurons is much greater than the expected number of clusters. At onset all of the clustering neurons are non–committed. The few round of trails generates a committed neuron to represent a cluster. Once the committed clustering neuron has been determined, the next spike train is presented. All spike trains that belong to the same committed neuron belong to the same cluster.

The number of input patterns that can be learned by a PART neural network is limited only by the finiteness of the number and length of spike trains that can be presented to it. There are a number of consequences for the practical application of PART neural networks:

- it is better to cluster data sets with respect to a few, e.g. one, statistical features at a time;
- the order of presentation of spike trains may have an influence of the clustering results;
- the number of clustering neurons must be larger than the number of suspected clusters;
- there will always be a small number of spike trains which do not cluster well: following [9] we placed all such data into an *outlier node*.

The PART clustering algorithm was validated on populations of neural spike trains constructed using two types of model neurons: 1) the leaky integrate and fire (LIF) mode, and 2) a reduced Hodgkin–Huxley model. The goal in constructing these data sets was to pose a difficult clustering problem consistent with the known physiological responses of neurons. Validation using this procedure is facilitated by the fact that the natures and numbers of the true clusters are known.

4.3. *Experiments on online social network news aggregation*

We have recently considered [19,46] the issue of news aggregation in online social networks with a pilot project on the social news website Digg.com, a content discovery and sharing application launched in 2004. According to the traffic statistics by Alexa.com in 2010, Digg is the 117th most popular website globally and 52nd in the USA[40]. Digg allowed people to vote web content up or down, called digging and burying, respectively. Users in Digg can share the content with other users who are connected to them by voting for or against the news. We have used the dataset from K. German (http://www.isi.edu/lerman/downloads/digg2009.html), who collected the information of stories on the Digg's front page over a period of a month in 2009. 3553 popular stories are voted for 3,018,197 times by 139,409 distinct users and on average, each story received about 850 votes.

All of the stories were provided the voter ID as well as the exact time of when voted. We are able to obtain the time series curve of each story. Apparently, large amount of superficial information can be found, such as how the popularity was, when the curve started and how the voting rate was going. When thousands of news gather together, they show some similarities to the tendency of the curves. We formatted the votes data over a period of 50 hours, as is used in [42]. Most of the stories were almost fixed and experienced little change on the vote number at the end of 50 hours. For each story, at the end of each hour, we obtained the cumulative number of voted users and used the value as a measurement of voting density. Here we can get a 3553×50 matrix indicating the increasing vote trend for all the stories.

Obviously, clustering these data in the 50 dimensional spaces is meaningless since every news distinguished itself from all others. Indeed, when we try to cluster these news in the full space, we found a large number of clusters with every cluster containing very small number of pieces of news. Projective clustering in relatively lower dimensional subspaces does generate some meaningful clusters, for example, Figure 1 gives a cluster, when we specify the PART algorithm to find clusters in subspaces of at least 30 dimension. Effectively, these pieces of news are grouped in one single cluster as they all reached the equilibrium states (final size of the news outbreak) after 20 hours from the source, and the final accumulated numbers of votes are close to each other. A better way is to look at the number of new votes each hour, and this give a curve of "influence votes" for each news, very much similar to the typical epidemic curve of an outbreaking

infectious disease. Then we can define certain features for each news, such as the initial time t_i when the total number of votes reach a pre-assigned number (say 50 in Figure 2), the beginning and ending times (t_b and t_e) of the so-called "viral period"–when the "epidemics" starts and ends, the turning point t_{tu} when the growth rate of the number of influenced users changes from being positive to negative. In this way, each time series of a given news is characterized by these features. Figure 2 gives a projective cluster of news with respect to the subspaces (t_i, t_b, t_{tu}, t_e).

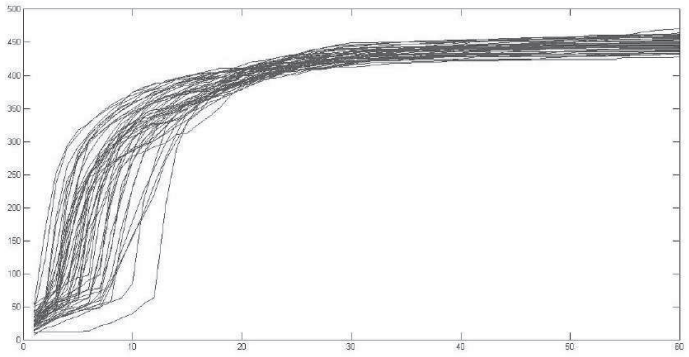

Figure 1. An example of projective clustering of the time series for the accumulated votes in the Digg networks. The cluster is formed based on the final size, and for those news reaching the equilibrium state within 20 ours since their release from the sources.

4.4. *Application to gene filtering and cancer diagnosis*

In [37], PART was used as a gene filtering method for the construction of robust prognostic predictors. Subspace clustering is essential for establishing prognostic predictors of various diseases using DNA microarray analysis technology, since it is desired to selectively find significant genes for constructing the prognostic model and also necessary to eliminate nonspecific genes or genes with error before constructing the model. According to the authors,

> "Genes selected by PART were subjected to our FNN-SWEEP modeling method for construction of a cancer class prediction model. The model performance was evaluated through comparison with a conventional screening signal-to-noise (S2N) method

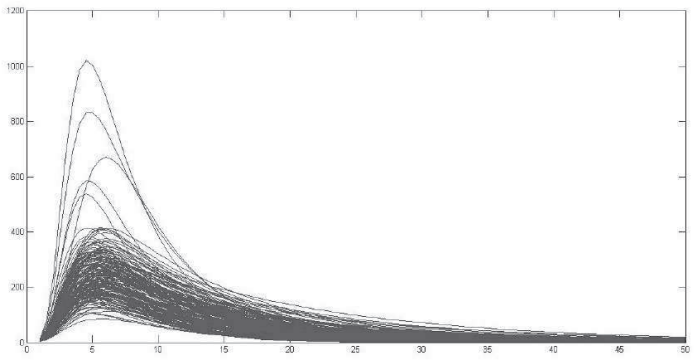

Figure 2. An example of projective clustering of the news in the Digg network, using features that are selected to reflect some "epidemic" nature of the number of new "influence votes."

> or nearest shrunken centroids (NSC) method. The FNN-SWEEP predictor with PART screening could discriminate classes of acute leukemia in blinded data with 97.1% accuracy and classes of lung cancer with 90.0% accuracy, while the predictor with S2N was only 85.3% and 70.0% or the predictor with NSC was 88.2% and 90.0%, respectively. The results have proven that PART was superior for gene screening."

The PART network was also used in [39] as a gene filtering method for cancer diagnosis marker extraction for soft tissue sarcomas. The authors noted that

> "In a previous study, we developed the PART filtering method by modifying PART, and reported that PART exhibited a higher performance than conventional methods, such as S2N and NSC. The combination method of PART and BFCS (PART-BFCS) was developed and applied to gene expression data, such as lymphoma and esophageal cancer. In the present study, we applied the various filtering methods to the gene expression profile data for the STS subtypes and constructed SVM models using the filtered genes. The results showed that the accuracy of the model based on the genes filtered by PART was the highest."

Further, the same group of authors combined PART with boosted fuzzy classifier and the SWEEP operator method effectively for gene selection. They concluded that [38]

> "In the present study, we investigated combinations of various filter and wrapper approaches, and found that combination method of PART and BFCS (a kind of boosting) is significantly superior to other methods with regard to high prediction accuracy for construction of class predictor from gene expression data. This method could select some marker genes related to cancer outcome. In addition, we proposed improved RIBFCS of PART-BFCS. Based on this new index, the discriminated group with over 90% prediction accuracy was separated from the others. It is necessary that there are about 90% or more prediction accuracy in the practical diagnosis application. These results suggest that the PART-BFCS method has a high potential to function as a new method of marker gene selection for the diagnosis of patients, using high dimensional data such as DNA microarray, mass spectrometry (MS), and two-dimensional polyacrylamide gel electrophoresis (2D-PAGE)."

4.5. *Application to text mining*

In [34], PART was used as an effective tool for reducing multidimensional text document space and also the text document clustering. It was shown that PART overcomes some lacks of computational complexity in traditional clustering algorithms in multidimensional space. They noted that with appropriate parameter settings of distance, the PART neural network achieved very good results on the clustering of multidimensional text documents and sorts precisely selected documents to corresponding supposed clusters. In addition to the correct classification of the text documents, PART was able to distinguish the projective dimension centers in each cluster and group noisy documents included in the outlier cluster. They stated that

> "Clustering algorithm via PART can correctly collect input documents to corresponding clusters, when the number of dimensions in a text document dataset increases, distance measure does not become increasingly meaningless and effort of system do not go down. The PART with appropriate input parameters enables to find the correct number of clusters, the correct centers for each

cluster and sufficiently large subset of dimensions in which clusters are formed. Results of our approach show, that application of PART for clustering of text documents can easy discover intrinsic clusters and also discover noisy patterns in datasets of text documents. Thanks to using average similarity degree function and exact settings of input parameters has this modification of PART very good computational efficiency in process of C."

4.6. *Application to stock associations*

In [35], the PART algorithm was improved with buffer management known as BPART to overcome the disadvantage of PART depending on accurate parameters and orders of input data sets. The authors noted that although "Projective Adaptive Resonance Theory, based on the ART and PROCLUS, is very good at recognizing self-organizing patterns in arbitrary sequences," the clustering accuracy may be degraded if an incorrect value ρ is chosen. They proposed an improvement-buffer management, which can neglect the noise data sets and achieve a parameter-free algorithm.

"We find that there are 4 out of 100 stocks which have concurrence associations. In detail, 00001 Cheung Kong, 00004 Wharf (Hldgs), 00012 Henderson Land and 00293 Cathay Pac Air are related (or concurrence) in 90 days out of 481 transaction days, and partly related in 105 days out of 481. Compared with PART, our algorithm initializes the important parameter ρ to 2, which is easily estimated and applied. And from this result above, any value more than 4 fails to find the concurrence of four stocks. Therefore, our algorithm over PART can obtain the good result without an accurate parameter ρ."

4.7. *Application to ontology construction*

In [18], the PART algorithm was used along with Bayesian network probability theory to construct an ontology in the system. In details, it was an efficacious tool for clustering the web pages based on the frequency of the term. It was shown that the PART tree can provide critical information about the hierarchical relation of the projective clusters. The PART neural network does a good job because it not only considers the data points but also the dimensions. What's more, it can deal with the lack of flexibility in the cluster.

> "PART shows better results when the quantity of data is large. In this experiment, we attempt to demonstrate that PART is better than ART in web page clustering. We used the ART neural network to cluster all the web pages (1523 web pages) and compared the result with the results of clustering by PART. In order to emphasize the equity of comparison, the parameter settings of ART were identical to those of PART .... Afterward, we used the method described in Chen et al. (2008) to generate the pattern for ART. It is clear that the concept precision (C_P) and concept location precision (C_L_P) of PART are both better than those of ART."

4.8. *Comments on challenges and future directions*

Seeking a neural-network inspired dynamical system architecture that automatically identifies projected clusters in high dimensional data leads us very naturally to the extension of the celebrated Adaptive Resonance Theory by incorporating the delay adaptation in neural computation. This extension generated an effective Projective Adaptive Resonance Theory neural network, whose global dynamical behaviour is governed by a large scale system of delay differential equations with adaptive delay. This adaptive delay, a special case of the so-called state-dependent delay, has been the focus of recent and intensive study in the field of functional differential equations and infinite dimensional dynamical systems. Consequently, we hope further development, in terms of neural physiological evidence, the qualitative and numerical theory, and applications, of this neural network architecture should provide inspiration for the development of a comprehensive theory for systems of state-dependent delay differential equations. Specifically, we have mentioned the PhD thesis of Qingwen Hu and the subsequent publications about the nonlinear oscillations in the form of Hopf bifurcation and global continuation. There is some evidence in the paper of [45] that PART-D may exhibit some oscillatory behaviours in certain parameter ranges, and how the general theory can be applied to exclude or confirm this oscillatory behaviours of a clustering algorithm remains to be a subject for future study.

We have shown that delay in neural networks may play a very useful role in regulating the speed with which different set of information is processed in order to identify hidden patterns in subspaces. This is based on the assumption that delay decays naturally without learning and external

stimuli, and that adaptive delay is increased if the input pattern is different from the stored pattern of a potential cluster. Our PART-D model formulates this as a simple linear equation with forcing and we had some very preliminary justification of this delay decay and adaptation law using the cable equation in the appendix of [45]. We would like to derive this delay decay and adaptation law from some first principles in neural dynamics. It would be interesting to see how this is linked to the thesis of Beamish[3] and its subsequent publications [5,6,4,7]. In particular, we would like to know how to use this alternative neurodynamical formulation to obtain a precise delay adaption rule of the PART-D neural network architecture for projective clustering.

PART has been successfully extended to deal with category data and fuzzy clustering in the thesis of Gan [20] and in its subsequent publications [22,24,23,25] . How to extend PART-D along this direction remains open. There have been two MSc theses [36,28] at LIAM dedicated to extending PART for supervised clustering to deal with the useful annotation information of some data sets, further development would need good indices to evaluate the effectiveness of a good clustering algorithm and clustering results, as functions of algorithm-relevant parameters.

Finally, we note that high-dimensional data clustering poses significant challenges for traditional clustering algorithms when correlations among features appear as a result of increasing number of dimensions. These local arbitrarily oriented correlations are the interesting hidden patterns in many applications. In [47], we developed a new correlation clustering algorithm by designing an ART-type neural network architecture. Our new iterative clustering algorithm PART-D-MCA incorporates minor component analysis to a delay-driven winner-take-all architecture. The resulting method shows very promising properties, and demonstrates the potential of extending this theory for clustering data sets in nonlinear submanifolds.

References

1. C.C. Aggarwal, J.L. Wolf, P.S. Yu, C. Procopiuc, and J.S.Park. Fast algorithms for projected clustering. In *Proceedings of the 1999 ACM SIGMOD international conference on Management of data*, pages 61–72. ACM Press, 1999.
2. R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. In *SIGMOD Record ACM Special Interest Group on Management of Data*, pages 94–105, New York, NY, USA, 1998. ACM Press.

3. D. Beamish. *50 years later: a neurodynamic explanation of Fitts' law*. PhD thesis, Department of Mathematics and Statistics, York University, 2004.
4. D. Beamish, M. Bhatti, S. MacKenzie, and J. Wu. Fifty years later: a neurodynamic explanation of fitts' law. *Journal of the Royal Society Interface*, 3(10):649–654, 2006.
5. D. Beamish, S.A. Bhatti, C.S. Chubbs, I.S. MacKenzie, J. Wu, and Z. Jing. Estimation of psychomotor delay from the fitts' law coefficients. *Biological Cybernetics*, 101(4):279–296, 2009.
6. D. Beamish, S.A. Bhatti, J. Wu, and Z. Jing. Performance limitation from delay in human and mechanical motor control. *Biological Cybernetics*, 99(1):43–61, 2008.
7. D. Beamish, S. MacKenize, and J. Wu. Speed-accuracy trade-off in planned arm movements with delayed feedback. *Neural Networks*, 19(5):582–599, 2006.
8. Y. Cao. *Neural Networks for Clustering: Theory, Architecture, Algorithms and NeuralDynamics*. PhD thesis, Department of Mathematics and Statistics, York University, Toronto, ON, Canada, October 2002.
9. Y. Cao and J. Wu. Projective ART for clustering data sets in high dimensional spaces. *Neural Networks*, 15(1):105–120, January 2002.
10. Y. Cao and J. Wu. Dynamics of projective adaptive resonance theory model: the foundation of PART algorithm. *IEEE Transactions on Neural Networks*, 15(2):245–260, 2004.
11. G. A. Carpenter, S. Grossberg, and J. H. Reynolds. ARTMAP: Supervised real-time learning and classification of nonstationary data by a self-organizing neural network. *Neural Networks*, 4:565–588, 1991.
12. G. A. Carpenter, S. Grossberg, and D. B. Rosen. Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance system. *Neural Networks*, 4:759–771, 1991.
13. G.A. Carpenter and S. Grossberg. ART2: Self-organization of stable category recognition codes for analog input patterns. *Applied Optics*, 26:4919–4930, 1987.
14. G.A. Carpenter and S. Grossberg. A massively parallel architecture for a self-organizing neural pattern recognition machine. *Computer Vision, Graphics and Image Processing*, 37:54–115, 1987.
15. G.A. Carpenter and S. Grossberg. Neural dynamics of category learning and recognition: Attention, memory consolidation, and amnesia. In S. Grossberg, editor, *The Adaptive Brain I Cognition, Learning, Reinforcement, and Rhythm*, volume 42 of *Advances in Psychology*, pages 239 – 286. North-Holland, 1987.
16. G.A. Carpenter and S. Grossberg. ART3: Hierarchical search using chemical transmitters in self-organizingpattern recognition architectures. *Neural Networks*, 3:129–152, 1990.
17. G.A. Carpenter, S. Grossberg, N. Markuzon, J.H. Reynolds, and D.B. Rosen. Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps. *IEEE Transactions on Neural Networks*, 3:698–713, 1992.

18. R.-C. Chen and C.-H. Chuang. Automating construction of a domain ontology using a projective adaptive resonance theory neural network and bayesian network. *Expert Systems*, 25(4):414–430, 2008.
19. M. Freeman, J. McVittie, I. Sivak, and J. Wu. An epidemiological approach to information propagation in the digg online social network. *submited*, 2013.
20. G. Gan. Subspace clustering for high dimensional categorical data. Master's thesis, Department of Mathematics and Statistics, York University, Toronto, Canada, October 2003.
21. G. Gan, C. Ma, and J. Wu. *Data Clustering: Theory, Algorithms, and Applications*, volume 20 of *ASA-SIAM Series on Statistics and Applied Probability*. SIAM Press, SIAM, Philadelphia, ASA, Alexandria, VA, USA, 2007.
22. G. Gan and J. Wu. A convergence theorem for the fuzzy subspace clustering (fsc) algorithm. *Pattern Recognition*, 41(6):1939–1947, 2008.
23. G. Gan, J. Wu, and Z. Yang. A fuzzy subspace algorithm for clustering high dimensional data. In X. Li, O.R. Zaiane, and Z. Li, editors, *Lecture Notes in Artificial Intelligence*, volume 4093, pages 271–278. Springer, August 2006.
24. G. Gan, J. Wu, and Z. Yang. A genetic fuzzy k-modes algorithm for clustering categorical data. *Expert Systems with Applications*, 36(2):1615–1620, 2009.
25. G. Gan, Z. Yang, and J. Wu. A genetic k-modes algorithm for clustering categorical data. In X. Li, S. Wang, and Z.Y. Dong, editors, *Proceedings on Advanced Data Mining and Applications: First InternationalConference, ADMA 2005, Wuhan, China*, volume 3584 of *Lecture Notes in Artificial Intelligence*, pages 195–202. Springer-Verlag GmbH, July 2005.
26. S. Grossberg. Adaptive pattern classification and universal recoding, i: parallel development and coding of neural feature detectors. *Biological Cybernetics*, 23:121–134, 1976.
27. S. Grossberg. Adaptive pattern classification and universal recoding, ii: feedback, expectation, olfaction, and illusions. *Biological Cybernetics*, 23:187–202, 1976.
28. H. Habibkhani. Supervised projective adaptive resonance theory with top scoring pair (part-tsp). *MSc Dieertation, York University*, 2012.
29. Q. Hu. *Differential equations with state-dependent delay: global Hopf bifurcation and smoothness dependence on parameters*. PhD thesis, Department of Mathematics and Statistics, York University, 2008.
30. Q. Hu and J. Wu. Global continua of rapidly oscillating periodic solutions of state-dependent delay differential equations. *Journal of Dynamics and Differential Equations*, 22(2):253–284, 2010.
31. Q. Hu and J. Wu. Global hopf bifurcation for differential equations with state-dependent delay. *Journal of Differential Equations*, 248(12):2801–2840, 2010.
32. Q. Hu, J. Wu, and X. Zou. Estimates of periods and global continua of periodic solutions for state-dependent delay equations. *SIAM Journal on Mathematical Analysis*, 44(4):2401–2427, 2012.
33. J. Hunter, J. Wu, and J. Milton. Clustering neural spike trains with transient responses. In *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, 2008.

34. R. Krakovsky, S. Ruzomberok, and I. Mokris. Clustering of text documents by projective dimension of subspaces using PART neural network. In *Applied Computational Intelligence and Informatics (SACI), 2012 7th IEEE International Symposium on*, pages 203–207, 2012.
35. L. Liu, L. Huang, M. Lai, and C. Ma. Projective ART with buffers for the high dimensional space clustering and an application to discover stock associations. *Neurocomputing*, 72(4-6):1283–1295, 2009.
36. W. Liu. Supervised projective adaptive resonance theory. *MSc Dieertation, York University*, 2007.
37. H. Takahashi, T. Kobayashi, and H. Honda. Construction of robust prognostic predictors by using projective adaptive resonance theory as a gene filtering method. *Bioinformatics*, 21:179–186, 2005.
38. H. Takahashi, Y. Murase, T. Kobayashi, and H. Honda. New cancer diagnosis modeling using boosting and projective adaptive resonance theory with improved reliable index. *Biochemical Engineering Journal*, 33:100–109, 2007.
39. H. Takahashi, T. Nemoto, T. Yoshida, H. Honda, and T. Hasegawa. Cancer diagnosis marker extraction for soft tissue sarcomas based on gene expression profiling data by using projective adaptive resonance theory (PART) filtering method. *BMC Bioinformatics*, 7:1–11, 2006.
40. S. Tang, , N. Blenn, C. Doerr, and P. Van Mieghem. Digging in the digg social news website. *IEEE Transactions on Multimedia*, 13(5):1163–1175, 2011.
41. C. von der Malsburg. Self-organization of orientation sensitive cells in the striate cortex. *Kybernetik*, 14:85–100, 1973.
42. F. Wang, H. Wang, K. Xu, J. Wu, and X. Jia. Characterizing information diffusion in online social networks with linear diffusive model. In *2013 IEEE 33rd International Conference on Distributed Computing Systems*, 2013.
43. J.R. Williamson. Gaussian ARTMAP: a neural network for fast incremental learning of noisy multidimensional maps. *Neural Networks*, 9:881–897, 1996.
44. J. Wu. High dimensional data clustering from a dynamical systems point of view. In W. Nagata and N. Sri. Namachchivaya, editors, *Fields Institute Communications*, volume 49 of *Bifurcation Theory and Spatio-Temporal pattern Formation*, pages 117–150, American Mathematical Society 2005.
45. J. Wu, H. ZivariPiran, J. Hunter, and J. Milton. Projective clustering using neural networks with adaptive delay and signal transmission loss. *Neural Computation*, 23(6):1568–1604, 2011.
46. J. Yin, J. McVittie, and J. Wu. Dynamic modelling assisted feature selection for online social network news aggregation. *submitted*, 2013.
47. Hossein ZivariPiran and J. Wu. Similarity-driven delay adaptation for clustering skewed high-dimensional data. *submitted*, 2013.