In Proc. of the IEEE SBRN (Brazilian Symposium on Artificial Neural Networks), pp. 84-89, Rio de Janeiro, 22-25 November, 2000.

An Evolutionary Immune Network for Data Clustering

Leandro Nunes de Castro
lnunes@dca.fee.unicamp.br

Fernando J. Von Zuben
vonzuben@dca.fee.unicamp.br

Department of Computer Engineering and Industrial Automation – DCA
School of Electrical and Computer Engineering – FEEC
State University of Campinas – UNICAMP

Abstract

This paper explores basic aspects of the immune system and proposes a novel immune network model with the main goals of clustering and filtering unlabeled numerical data sets. It is not our concern to reproduce with confidence any immune phenomenon, but to show that immune concepts can be used to develop powerful computational tools for data processing. As important results of our model, the network evolved will be capable of reducing redundancy, describing data structure, including the shape of the clusters. The network will be implemented in association with a statistical inference technique, and its performance will be illustrated using two benchmark problems. The paper is concluded with a trade-off between the proposed network and artificial neural networks used to perform unsupervised learning.

1. Introduction

The *vertebrate immune system* has several useful mechanisms from the viewpoint of information processing. Among the many immunological models, we can stress the *immune network theory* [6] and the *clonal selection and affinity maturation algorithms* [1,2]. In this work, we will briefly review these theories and show that many of their concepts and ideas can be used to develop an *artificial network structure* capable of solving similar *pattern recognition* tasks as the natural immune system. This paper formally derives the network model, discusses its applications and how to interpret the results. The link between our model and *artificial neural networks* for unsupervised learning is also discussed.

A network will be constructed to give answers to the following questions: (1) Is there a great amount of redundancy in the data and, if yes, how can we reduce it? (2) Is there any group or subgroup intrinsic to the data? (3) How many groups are there within the data set? (4) What is the structure of these data (groups)? (5) How can we generate decision rules to classify novel samples?

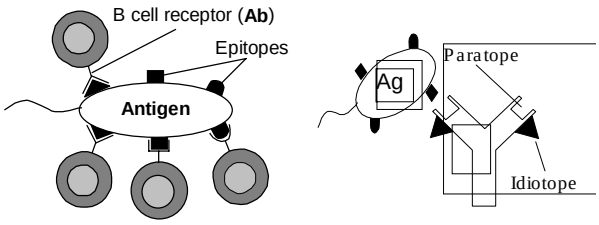
1.1 A Brief Description of the Immune System

To begin, we shall sketch a few aspects of the human *adaptive immune system*. A number of concepts and technical terms will be introduced to make the reader familiar with the terminology. Master details about the *immune network theory* and the *clonal selection principle* will be given in dedicated sections. An interested reader shall refer to Janeway Jr. & Travers [5] for an introductory text in immunology and de Castro & Von Zuben [3] for immunology under the AI perspective.

The *immune system* is a complex of cells, molecules and organs with the primary role of limiting damage to the host organism by *pathogens* (called *antigens*, Ag), which elicit an *immune response*. One type of response is the secretion of *antibody* molecules by B *cells* (or B-*lymphocytes*). *Antibodies* (Ab) are Y-shaped receptor molecules bound on the surface of a B cell with the primary role of *recognizing* and *binding*, through a *complementary match*, with an antigen.

The Ab *recognizes* a portion of the Ag called its *epitope*. An *idiotype* is defined as the set of epitopes displayed by the variable regions of a set of Ab, and an *idiotope* is each single idiotypic epitope. While each B cell is known to have a single type of Ab, thus being called *monospecific*, an Ag typically has several different types of epitopes, and can be recognized by several different antibodies. The antibody portion responsible for matching (recognizing) an antigen is called *paratope*, also known as V-region, for *variable region*. It is variable because it can *alter its shape* to achieve a better match (complementarily) with a given antigen. The strength and specificity of the Ag-Ab interaction is measured by the *affinity* of their match. Figure 1 illustrates an Ag with its many epitopes and an Ab with its paratope and idiotope.

In order to be protective, the immune system must learn to distinguish between our own (*self*) cells and malefic external (*nonself*) invaders. This process is called *self/nonself discrimination*: those cells recognized as self do not promote an immune response, while the unrecognized ones provoke a reaction.



(a) (b)

Figure 1: B cell, Ag, Ab, epitopes, paratopes and idiotopes. (a) Monospecific B cell with its receptor(Ab). (b) Antibody combining site (V-region or paratope), and its idiotope.

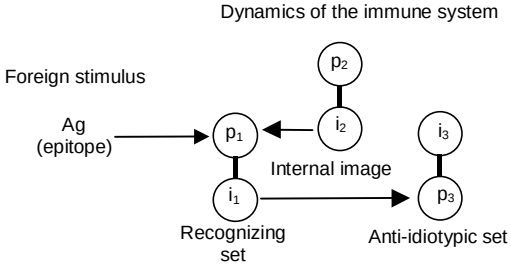


Figure 2: Idiotypic network representation. (a) An antigen stimulates the Ab production of class 1, who stimulates class 2, and so on (see [6] for description).

2. Immune Network Theory

The immune network theory, as originally proposed by Jerne [6], hypothesized a novel viewpoint of lymphocyte activities, natural Ab production, pre-immune repertoire selection, tolerance, self/nonself discrimination, memory and the evolution of the immune system. The immune system was formally defined as an enormous and complex network of paratopes, that recognize sets of idiotopes, and of idiotopes, that are recognized by sets of paratopes. The relevant events in the immune system are not only the molecules, but also their interactions. The immune cells can respond either positively or negatively to the recognition signal. A positive response would result in cell *proliferation*, *activation* and antibody secretion, while a negative response would lead to *tolerance* and *suppression*. Figure 2 depicts the immune network idea.

In the model proposed by Varela & Coutinho [12], we can stress three characteristics of the immune networks: 1) its *structure*, that describes the types of interaction among the network components, represented by matrices of connectivity; 2) its *dynamics*, that accounts for the variation in time of the concentrations and affinities of its cells; and 3) its *metadynamics*, a property addressed to the continuous production of novel antibodies and death of non-stimulated or self-reactive cells. The central characteristic of the immune network theory is the definition of the individual's molecular identity (*internal images*), which emerges from a network organization followed by the learning of the molecular composition of the environment where the system develops.

The network approach is particularly interesting for the development of computer tools because it potentially provides a precise account of emergent properties such as learning and memory, self-tolerance, size control and diversity of cell populations. In general terms, the structure of most network models can be represented as

$$RPV = \begin{matrix}\text{influx}\\ \text{of new}\\ \text{cells}\end{matrix} - \begin{matrix}\text{death of}\\ \text{unstimulated}\\ \text{cells}\end{matrix} + \begin{matrix}\text{reproduction}\\ \text{of stimulated}\\ \text{cells}\end{matrix} \quad (1)$$

where RPV is the rate of population variation, and the last term includes Ab-Ab recognition and Ag-Ab stimulation.

3. Clonal Selection and Maturation of Immune Responses

When stimulated, a B cell *proliferates* and *secretes* its receptor molecules as free Ab. Antibodies thus can either be free or receptors attached to cells. Secretion requires that B cells become activated, undergo *proliferation* (*cloning*) and finally differentiate into *plasma* (large Ab secretors) and *memory cells* (high affinity, long living cells), as illustrated in Figure 3. A *clone* is a cell, or set of cells, which are the progeny of a single cell. Those cells that recognize antigens grow in *concentration* and *affinity* (*affinity maturation*), while those that do not, die out, a phenomenon usually addressed as the *maturation of the immune response* (the *immune learning mechanism*). This basic process of *pattern recognition and selection* is known as *clonal selection* [2] and is similar to *natural selection*, except that it occurs on a rapid time scale, on the order of days, within our bodies (*microevolution*).

Repeatedly exposure to a given antigen considerably enhances the effectiveness of the immune response through the storage of high affinity memory cells from the early infections. This scheme is similar to a *reinforcement learning strategy* [11], where the system continuously improves its capability to perform its task (in this case, recognition of antigens).

The maturation of the immune response requires that the Ab, in the matured response, be structurally different from those present in the primary response. Random changes (mutations) are introduced into the V-region genes and occasionally one such change will lead to an increase in the antibody affinity. It is these high-affinity variants that are selected to enter the pool of memory cells. Those cells with low affinity receptors, or the self-reactive cells, must be efficiently eliminated (or become anergic).

The selection mechanism provides a means by which the regulation of the hypermutation process is made dependent on receptor affinity. Cells with low affinity receptors may be further mutated and eliminated if their antigenic affinity remain small. In cells with high-affinity receptors, hypermutation may be inactivated [8].

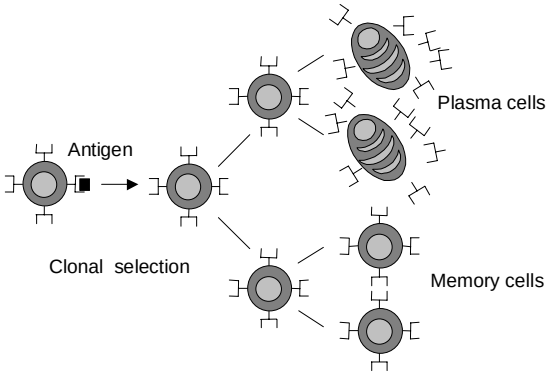


Figure 3: Ag-Ab interactions. A minority of cells will recognize the antigen, and be activated by clonal selection.

4. aiNet: An Evolutionary Artificial Immune Network

Let a *shape-space* S be a multi-dimensional metric space where each axis stands for a physico-chemical measure characterizing a molecular shape [10]. We will assume a set of unlabeled patterns $\boldsymbol{X} = \{\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_{Np}\}$, where each pattern $\mathbf{x}_i$, $i = 1,...N_p$, is described by p variables, to characterize a molecular configuration as a point $s \in S$. Hence, a point in S^p specifies the set of features necessary to determine the Ab-Ab and Ag-Ab interactions that can be mathematically represented as a p-dimensional vector. The possible interactions within the system will be represented in the form of a connectivity graph. Our network model can be formally defined as:

Definition 1: The proposed artificial immune network, named aiNet, can be defined as an *edge-weighted graph*, not necessarily fully connected, composed of a set of nodes, called *cells*, and sets of node pairs called *edges*. Each connected edge has a number assigned, called *weight* or *connection strength*.

The aiNet is said to be *evolutionary* because evolution strategies (based on genetic variation and selection within a population of individuals) will be used to control the network *dynamics* and *plasticity*. It is also *connectionist*, once a matrix of connection strengths is defined to measure affinities among the network cells.

The clusters in the network will serve as *internal images*, responsible for mapping existing clusters in the data set into network clusters. As an illustration, consider the data set of Figure 4(a). A hypothetical network structure, generated by the aiNet, is shown in Figure 4(b). The cell labels and connection strengths are presented. The dashed lines suggest connections to be pruned, in order to detect clusters and define the final network structure. Notice that the number of network cells is much smaller than the number of data samples, characterizing an architecture suitable for data compression. Note that the network size is automatically defined.

Like the models proposed by Farmer *et al*. and Jerne [4,6,7], we make no distinction between the B cell and the Ab. The Ag-Ab *affinity* is measured by a distance metric (*dissimilarity*) between them. Oppositely, the Ab-Ab affinity is defined by a *similarity* metric between them.

As proposed in the original immune network theory (Section 2), the existing cells will compete for antigenic recognition and those successful will lead to network activation and cell proliferation, clonal selection, (Section 3), while those who fail will be eliminated. In addition, Ab-Ab recognition will result in network suppression. In our model, suppression is performed by eliminating the self-recognizing cells, given a suppression threshold σ_s. Every pair Ag-Ab will relate to each other within the shape-space S through the affinity d_{ij} of their interactions (dissimilarity), which reflects the probability of starting a clonal response. Similarly, an affinity s_{ij} will be assigned to each pair Ab-Ab, reflecting their interactions (similarity).

The following notation will be adopted:
$\boldsymbol{X}$: data set composed of N_p vectors ($\boldsymbol{X} \in \Re^p$);
$\boldsymbol{C}$: matrix containing all the N_t network cells ($\boldsymbol{C} \in \Re^{Ntxp}$);
$\mathbf{M}$: matrix of the N memory cells, ($\mathbf{M} \subseteq \boldsymbol{C}$);
N_c: number of clones generated by each stimulated cell;
$\mathbf{D}$: dissimilarity matrix with elements d_{ij} (Ag-Ab);
$\mathbf{S}$: similarity matrix with elements s_{ij} (Ab-Ab);
n: n highest affinity cells selected to clone and mutate;
ζ: percentage of the matured cells to be selected; and
$\sigma_{d,s}$: natural death and suppression threshold, respectively.

The learning algorithm aims at building a memory set that recognizes and represents the data structural organization. The more specific the cells, the less parsimonious the network (low compression rate), whilst the more generalist the cells, the more parsimonious the network with relation to the number of cells (improved compression). The suppression threshold (σ_s) controls the specificity level of the cells, the clustering accuracy and network plasticity. As a suggestion, the user must first set a small value for σ_s (e.g., $\sigma_s \leq 10^{-3}$) and continuously fine-tune the network performance.

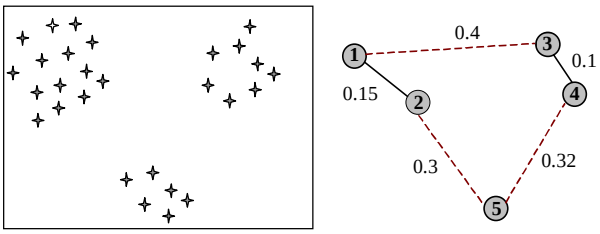


Figure 4: aiNet illustration. (a) Data set with clusters of high data density. (b) Network of labeled cells with weights assigned to the links. Dashed lines indicate connections to be pruned to generate disconnected sub-graphs (clusters).

The aiNet learning algorithm works as follows:

1. At each iteration step, do:
   - 1.1 For each antigen *i*, do:
     - 1.1.1 Determine its affinity, d_{ij}, to all the network cells according to a distance metric;
     - 1.1.2 Select the *n* highest affinity network cells;
     - 1.1.3 Reproduce (clone) these *n* selected cells. The higher the cell affinity, the larger N_c;
     - 1.1.4 Apply Equation (2) to these N_c cells;
     - 1.1.5 Determine **D** for these improved cells;
     - 1.1.6 Re-select ζ% of the highest affinity cells and create a partial $\mathbf{M}_p$ memory cell matrix;
     - 1.1.7 Eliminate those cells whose affinity is inferior to threshold σ_d, yielding a reduction in the size of the $\mathbf{M}_p$ matrix;
     - 1.1.8 Calculate the network Ab-Ab affinity, s_{ij};
     - 1.1.9 Eliminate $s_{ij} < \sigma_s$ (*clonal suppression*);
     - 1.1.10 Concatenate $\boldsymbol{C}$ and $\mathbf{M}_p$, ($\boldsymbol{C} \leftarrow [\boldsymbol{C};\mathbf{M}_p]$);
   - 1.2 Determine **S**, and eliminate those cells whose $s_{ij} < \sigma_s$ (*network suppression*);
   - 1.3 Replace r% of the worst individuals;
2. Test the stopping criterion.

In steps 1.1.1, 1.1.5 and 1.1.8 we adopted the Euclidean distance as a metric of similarity and dissimilarity. Steps 1.1.1 to 1.1.7 describe the clonal selection and affinity maturation processes. Steps 1.1.8 to 1.1.10 and 1.2 to 1.3 simulate the immune network activity. The affinity of the cells with the given antigen *i* can be improved by the following expression (directed mutation):

$$\boldsymbol{C} = \boldsymbol{C} - \alpha(\boldsymbol{C} - \boldsymbol{X}), \qquad (2)$$

where, $\boldsymbol{C}$ is the matrix of network cells, $\boldsymbol{X}$ the matrix of antigens and α is the *learning rate*, or *mutation rate*. The α value is set according to the Ag-Ab affinity, the higher the affinity the smaller the α. Equation (2) proposes a biased search, where the Ag-Ab complementarity is increased proportionally to α. By doing so, we guide our search to locally optimize the network cells (*greedy search*) in order to improve their antigenic recognition capability along the iterations.

As can be seen from this algorithm, a clonal immune response is elicited to each presented antigenic pattern. Notice also the existence of two suppressive steps in this algorithm (1.1.9 and 1.2): the *clonal suppression* is responsible for eliminating intra-clonal self-recognizing cells, while the *network suppression* searches for similarities between different sets of network clones. After the learning phase, the network cells represent internal images of the antigens (or groups of antigens) presented to it. As a complement to the general network structure presented in Equation (1), our model suppresses self-recognizing cells (steps 1.1.9 and 1.2).

The network outputs can be taken to be the matrix of memory cells' coordinates (**M**) and the matrix of inter-cell affinities (**S**). While matrix **M** represents the network internal images of groups of antigens, matrix **S** is responsible for determining which cells are connected to each other, describing the general network structure. To achieve a problem specific network structure, we will analyze the *minimal spanning tree* of the resulting net, to be described in the next section.

To evaluate the network convergence we propose several different criteria:

1. Stop iterating after a pre-defined number of steps (used in all experiments performed).
2. Stop the iterative process when the network reaches a pre-defined number of cells.
3. Evaluate the error between the antigens and **M**.
4. The network is supposed to have converged if its average error rises after *k* consecutive iterations.

4.1 Knowledge Extraction and Structure of the Trained aiNet

The network structure could simply be determined by fully connecting all the network cells according to matrix **S**, but it would make the network interpretation and knowledge extraction difficult tasks, mainly for $p > 3$. One way to alleviate the complexity of analysis and to detect clusters is to suppress all those connections whose strength extrapolates a pre-defined threshold. This idea, though simple, will not be adopted here because it does not account for any correlation within the network (indirectly in the data set) and might lead to erroneous interpretations. It is our main purpose here, to supply the user with a formal and sophisticate network interpretation strategy. Explicitly speaking, our goals are to determine (1) the number of *clusters*, and (2) the network cells belonging to each of the identified clusters, given insights into the shapes of each cluster. To do so, we use the network output, which is composed of matrix **M** and the upper triangular matrix **D**, along with a principle from *cluster analysis*. The problem is stated as follows:

Given a network with N memory cells ($\mathbf{M} \in \Re^{N \times p}$) *and their interconnections* (**S**), *devise a clustering scheme to detect inherent separations between clusters of* **M** *in a metric space governed by a distance measure d*(*x*,*y*).

The *minimal spanning tree* (MST) of a graph is a powerful mechanism to search for a locally adaptive interconnecting strategy for the network cells [9,13], and will serve as an aid to detect and describe the structure of our network clusters.

Definition 2: A tree is a *spanning tree* of a graph if it is a sub-graph containing all the vertices of the graph. A *minimal spanning tree* of a graph is a spanning tree with minimum weight, where the weight of a tree is the sum of the weights of its constituent edges.

The visualization of the MST is only feasible for $p \leq 3$, but we can draw a bar graph representing the distances between neighboring cells. Notice that the concept of neighborhood makes sense only after the generation of the MST. It is necessary to define a procedure for deleting edges from an MST so that the resulting connected subtrees correspond to the network clusters. The following criterion is used:

An MST edge (*i*,*j*) *whose weight* s_{ij} *is significantly larger than the average of nearby edge weights on both sides of the edge* (*i*,*j*) *should be deleted.* This edge is called *inconsistent*.

There are two natural ways to measure the significance referred to. One is to see how many sample standard deviations separate s_{ij} from the average edge weights on each side. The other is to calculate the *factor* (f) or ratio between s_{ij} and the respective averages [13].

5. aiNet Evaluation

In this section, we are going to apply the aiNet algorithm to two artificial benchmark problems in order to illustrate and discuss some characteristics of our model.

Figure 5(a) depicts the simple problem of classifying 50 samples in $\Re^2$ into five distinct classes, each of which contains 10 samples. In Figure 5(b), we have the well-known two-donut problem.

The parameters used for training each aiNet and the results obtained are presented in Figures 6 and 7. *SC* is the stopping criterion (fixed number of generations), f the factor discussed in Section 4.1, N the final number of cells, and the other parameters were studied in Section 4.

6. Discussion

In the first problem tested, the network produced a 66% compression rate and, in the second case, it could reduce the data set size in 92.6%. In addition, a correct classification was achieved in both cases, in the sense of detecting the number and shapes of the clusters.

In our network, clonal selection controls the amount and locations of the network cells (its dynamics and metadynamics), and the minimal spanning tree is used to define the final network structure. The learning algorithm is generic, but the resultant networks are problem dependent, i.e., the set of patterns (Ag) to be recognized will guide the search for the net structure and shape of cells. As its main drawbacks, we can mention its high number of user-defined parameters, its computational cost per iteration $O(p^3)$ with relation to the length p of the input vectors, and the network sensitivity to the suppression threshold (σ_s). Like most clustering neural networks (e.g., the Kohonen SOM), the MST criterion would have difficulties to determine the correct number of network clusters in cases characterized by intersections among the different clusters in the data set.

6.1 Artificial Neural Networks and aiNet

As final remarks, we can list the most striking similarities and differences between neural networks and the introduced aiNet model:

1. *Differences*:

Neural Networks

- Learning by altering connection strengths
- "Knowledge" stored in the connections
- Excitatory and inhibitory connections

aiNet

- Learning through the variation in concentration and affinity of network cells
- "Recognition" performed by cell receptors
- Activation and suppressive interactions

2. *Similarities*:

- Great diversity and amount of highly specific cells;
- Noise tolerance, generalization capability, auto-associative memory; and
- Sub-symbolic, parallel and distributed processing based upon a competitive learning scheme.

Acknowledgements

Leandro Nunes de Castro thanks FAPESP (Proc. n. 98/11333-9) and Fernando Von Zuben thanks FAPESP (Proc. n. 98/09939-6) and CNPq (Proc. n. 300910/96-7) for their financial support.

References

[1] Ada, G. L. & Nossal, G. (1987), "The Clonal Selection Theory", *Scientific American*, 257(2), pp. 50-57.

[2] Burnet, F. M. (1978), "Clonal Selection and After", In *Theoretical Immunology*, (Eds.) G. I. Bell, A. S. Perelson & G. H. Pimbley Jr., Marcel Dekker Inc., pp. 63-85.

[3] de Castro, L. N. & Von Zuben, F. J. (1999), "Artificial Immune Systems: Part I – Basic Theory and Applications", *Technical Report – RT DCA 01/99*, p. 90, URL: http://www.dca.fee.unicamp.br/~lnunes

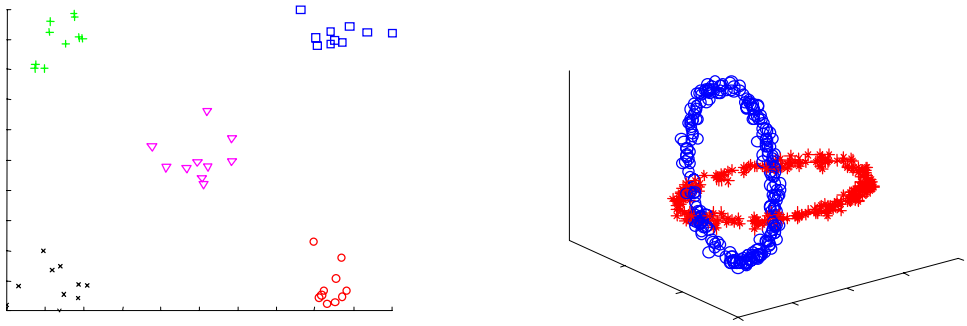
[4] Farmer, J. D., Packard, N. H. & Perelson, A. S. (1986), "The Immune System, Adaptation, and Machine Learning", *Physica 22D*, pp. 187-204.

[5] Janeway Jr, C. A. & P. Travers (1997), "Immunobiology The Immune System in Health and Disease", Artes Médicas (in Portuguese), 2nd Ed.

[6] Jerne, N. K. (1974a), "Towards a Network Theory of the Immune System", *Ann. Immunol.* (Inst. Pasteur) 125C, pp. 373-389.

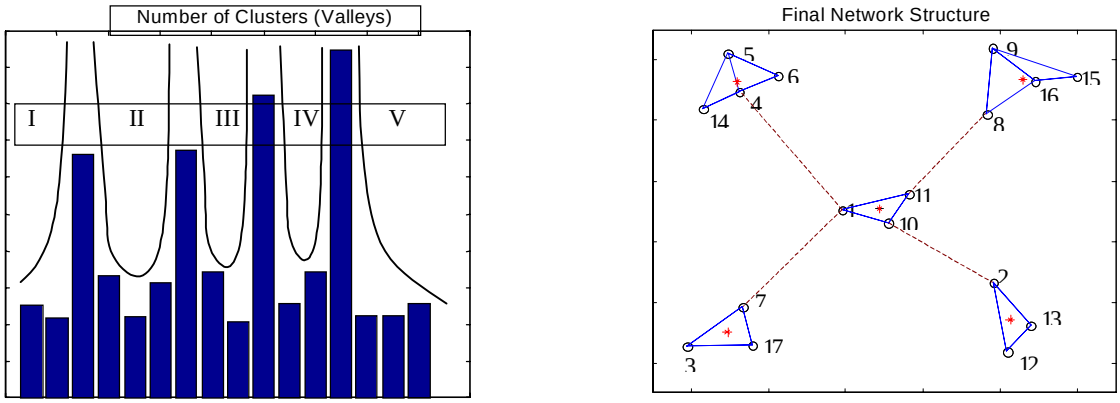
[7] Jerne, N. K. (1974b), "Clonal Selection in a Lymphocyte Network". In *Cellular Selection and Regulation in the Immune Response*, ed. G. M. Edelman, Raven Press, p. 39.

[8] Kepler, T. B. & Perelson, A. S. (1993), "Somatic Hypermutation in B Cells: An Optimal Control Treatment", *J. theor. Biol.*, 164, pp. 37-64.
[9] Leclerc, B. (1995), "Minimum Spanning Trees for Tree Metrics: Abridgements and Adjustments", *Journal of Classification*, 12, pp. 207-241.
[10] Perelsen, A. S. & Oster, G. F. (1979), "Theoretical Studies of Clonal Selection: Minimal Antibody Repertoire Size and Reliability of Self-Nonself Discrimination", *J. theor.Biol.*, 81, pp. 645-670.
[11] Sutton, R. S. & Barto, A. G. (1998), "Reinforcement Learning an Introduction", *A Bradford Book*.
[12] Varela, F. J. & Coutinho, A. (1991), "Second Generation Immune Networks", *Imm. Today*, 12(5), pp. 159-166.
[13] Zahn, C. T. (1971), "Graph-Theoretical Methods for Detecting and Describing Gestalt Clusters", *IEEE Trans. on Computers*, C-20(1), pp. 68-86.



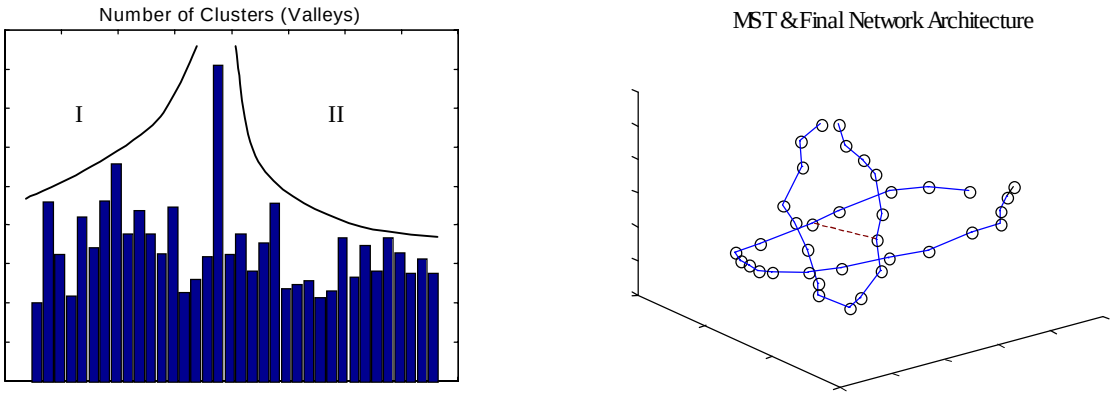
(a) (b)

Figure 5: Test problems for the aiNet learning algorithm. (a) Five linearly separable classes. (b) Two-donut problem.



(a) (b)

Figure 6: Results for the five classes problem, N_p = 50, ζ = 20%, σ_s = 0.1, *SC* = 10 gen. (a) Bar graph generated from the MST. (b) Resultant network architecture (N = 17). The edges pruned by the MST are the dashed ones.



(a) (b)

Figure 7: Results for the two-donut problem N_p = 50, ζ = 20%, σ_s = 0.1, *SC* = 10 gen. (a) Histogram generated from the MST. (b) MST and resultant network architecture (N = 37).