Comparative Performances of Stochastic Competitive Evolutionary Neural Tree (SCENT) with Neural Classifiers

W. Pensuwon[1,2], R. G. Adams[1] and N. Davey[1]

[1]Department of Computer Sciences
University of Hertfordshire
Hatfield, Herts, AL10 9AB, United Kingdom.

[2]Department of Electrical&Electronics Engineering
Ubonratchathani University
Ubonratchathani, 34190, Thailand.

W.Pensuwon@herts.ac.uk, R.G.Adams@herts.ac.uk and N.Davey@herts.ac.uk

Abstract

A stochastic competitive evolutionary neural tree (SCENT) is described and evaluated against the best neural classifiers with equivalent functionality, using a collection of data sets chosen to provide a variety of clustering scenarios. SCENT is firstly shown to produce flat classifications at least as well as the other two neural classifiers used. Moreover its variability in performance over the data sets is shown to be small. In addition SCENT also produces a tree that can show any hierarchical structure contained in the data. For two real world data sets the tree captures hierarchical features of the data.

1 Introduction

Unsupervised learning algorithms are used widely for the detection of implicit structure in unlabeled data, or, on a more modest scale, to assist human users in finding hidden structure in data interactively. Dynamic Neural Tree Networks (DNTNs) are unsupervised neural networks, that create nodes as needed in response to the data, placing them in a tree, so that any hierarchical structure in the data is represented. A recent DNTN is the Competitive Evolutionary Neural Tree (CENT), [1,2]. This model is comparatively robust, with respect to its parameter settings when compared with other DNTNs [3]. A stochastic version of CENT has been recently developed, Stochastic CENT (SCENT) [10].

This paper compares the performance of SCENT to two other high quality clusterers over a variety of data sets and runs. The most popular and well known neural clusterer is Kohonen's self organising map (SOM). Martinez's NeuralGas is considered to be one of the best performing neural classifiers [8]. Both of these clusters produce flat (non-hierarchical) classifications, so the comparisons reported here only evaluate SCENTs capabilities as a flat clusterer.

Sections 2 and 3 introduce CENT and SCENT respectively. Section 4 briefly discusses the other classifiers used for comparison. Section 5 sets out the details of the experiments and Section 6 gives the results. Finally we conclude with a summary.

2 Competitive Evolutionary Neural Tree (CENT)

In CENT, the tree structure is created dynamically in response to structure in the data set. The neural tree starts with a root node with its *tolerance* (the radius of its classificatory hypersphere) set to the standard deviation of input vectors and its position is set to the mean of input vectors. It has 2 randomly positioned children. Each node has two counters, called *inner* and *outer*, which count the number of occasions that a classified input vector is within or outside *tolerance*, respectively. These counters are used to determine whether the tree should grow children or siblings once it has been determined that growth is to be allowed.

2.1 Top-Level Algorithm

At each input presentation, a recursive search through the tree is made for a winning branch of the tree. Each node on this branch is moved towards the input using the standard competitive neural network update rule.

Any winning node is allowed to grow if it satisfies 2 conditions. It should be mature (have existed for an epoch), and the number of times it has won compared to the number of times its parent has won needs to exceed a threshold. A finite limit is put on the number of times a node attempts growth.

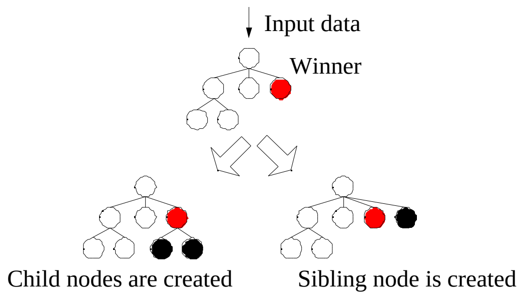


Figure 1. In CENT the winning leaf node can grow. Both downgrowth, in which children are created (on the left) and sidegrowth in which siblings are created (on the right) are shown.

When a node is allowed to grow, if it represents a dense cluster, then its inner counter will be greater than its outer counter and it creates two children. Otherwise, it produces a sibling node. The process of growth is illustrated in Figure 1.

To improve the tree two pruning algorithms, short and long term, are applied to delete the insufficiently useful nodes. The short-term pruning procedure deletes nodes early in their life, if their existence does not improve the classificatory error. The long-term pruning procedure removes a leaf when its activity is not greater than a threshold. See Figure 2 for the pruning process.

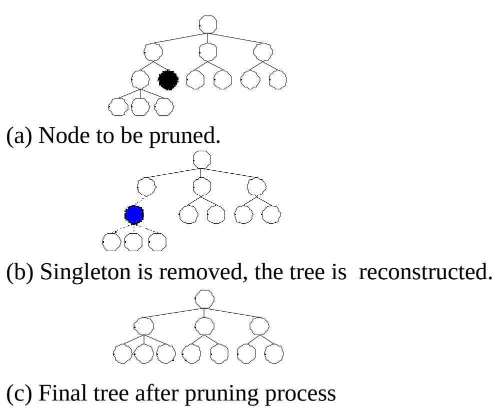


Figure 2. In CENT nodes that are ineffective can be pruned (a). If the resulting tree contains a node with no siblings (b), it is also removed (c).

3 Stochastic Competitive Evolutionary Neural Tree (SCENT)

Deterministic methods of search will perform poorly if they find local minima of the cost function. Stochastic methods are used to overcome some of these difficulties. The reason for adding stochasticity to CENT is to allow it to more fully explore the space of possible trees. In order to do this it may be useful to create more tentative new growth and for the pruning process to be more common. The addition of stochasticity can either soften decisions or add noise to generated values. Both of these 2 essentially different modes of stochasticity are used in SCENT and are described next.

The first mode is called Decision Based Stochasticity where the sharp change of decision, depending on some input, is made softer by the addition of some randomness. Decision based stochasticity is essentially the same technique as the addition of stochasticity to neural networks such as the Hopfield Network forming the Boltzmann machine [6]. The addition of stochasticity was implemented by altering the strong decision making in 3 key procedures, into softer ones using the logistic function, as depicted in Figure 3. The 3 procedures decide whether growth is allowed for a node, what type of growth (down or across), and whether pruning should occur.

In the deterministic version the decision is made at a precise value of the decision variable plotted on the horizontal axis. However, in the stochastic version the value obtained by the logistic function is compared to a random number between 0 and 1, and if larger, the decision is accepted. In this way values of the decision variable less than the original threshold can lead to positive decisions and values greater than the precise one can lead to negative decisions.

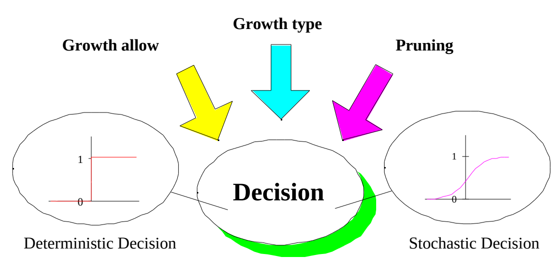


Figure 3. Decision Based Stochasticity. The probability of accepting a decision produced in the left ellipse is crisp whereas the probability of accepting a decision in the right ellipse is fuzzy.

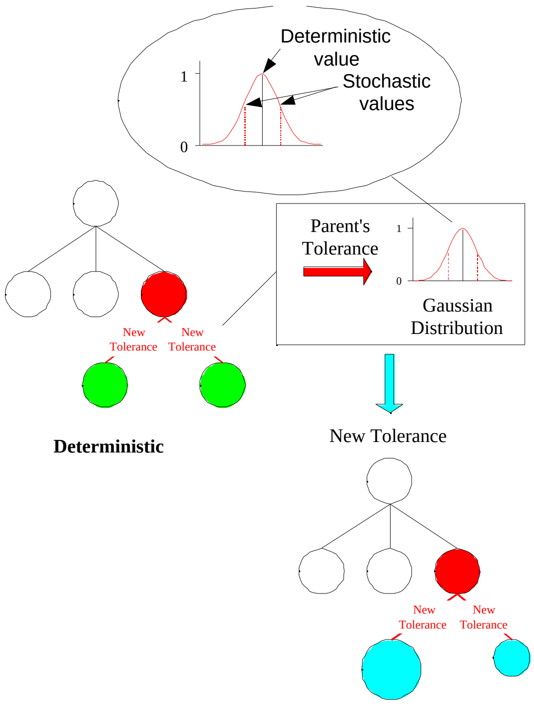
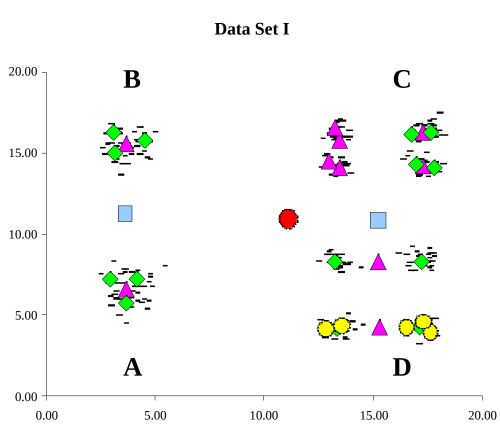


Figure 4. Generative Stochasticity. The rightmost of the level two nodes is producing 2 children. In deterministic CENT both children have the same value of tolerance, inherited from the performance of the parent (left hand tree). In SCENT a Gaussian is superimposed on the deterministic value to generate 2 different child tolerances (right hand tree).

The second mode is *Generative Stochasticity*, which adds some randomness to value generation. This technique shown in Figure 4 is similar to the Soft Competition Scheme [11].

Together these two modes add "soft" decisions to each of the major decision points in the code and add a degree of randomness to each major place where new values are generated.

Figure 5 illustrates an example of a tree structure produced by SCENT with nodes and leaf positions represented by different shapes according to their level in the tree.



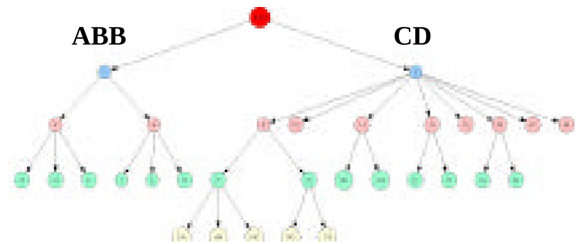


Figure 5. The position of the nodes in a SCENT tree is shown superimposed upon a 2 dimensional data set (top). SCENT splits the data into 2 halves at level 1 (squares) and then finds subclusters. For example on the left the two subclusters (triangles) are found at level 2. The hierarchical structure is shown with a dark circle at root level, and subsequent levels represented by squares, triangles diamonds and open circles.

4 Other Neural Classifiers

The most common and well-known neural classifiers are Neural gas [8] and the Self-Organising Map [7]. These two models were chosen for comparison with SCENT.

Neural gas was proposed by Martinez et al. [8]. The algorithm uses soft competition, in which many nodes move at each data presentation, where the degree of movement is based on the rank order of error. A temperature factor is also used to control the softness of the competition over time. Neural gas can converge quickly to a low distortion error, however, it may take a comparatively long time for this to occur.

SOMs have been used extensively for many applications since first being proposed by Kohonen.

5 Experiments

To compare the three neural classifiers, SCENT, SOM and NGAS were used to produce clusterings of a variety of unlabelled data sets. Each model was run over each data set for 30 complete runs.

Both the SOM and NGAS require the number of classifying nodes to be preset. So that the final clusterings are directly comparable both models are initialised with the same number of nodes as produced by SCENT on each data set.

5.1 Data Sets

The data sets used here vary in: size, shape of clusters, number of cluster present, balanced and unbalanced hierarchy structure, degree of overlap between clusters and dimensionality. These data sets have been generated to test the networks over a wide range of different performance areas and sensitivity of the networks to alterations in the dimensionality of the data. 7 artificial data sets (such as the one shown in Figure 5) and 9 real world data sets (such as the well know IRIS data set) are used.

5.2 Cluster Measures

The general goal in many clustering applications is to arrive at clusters of objects that show small within-cluster variation relative to the between-cluster variation [4,5,9]. There are two types of clustering measures, ones that grade the flat clustering performance of the leaf nodes and ones that grade the hierarchical structure. Here only the quality of the flat clustering is considered. The *Gamma* measure was selected as the best of the proposed measures [10]. It is defined by:

$$\Gamma = \frac{s(+) - s(-)}{s(+) + s(-)}$$

where: s(+) is the number of times when two points not clustered together are further apart than two points which are in the same cluster and s(-) is the number of times when two point not clustered together are closer than two points which are in the cluster.

This gives a value between -1 and +1, where +1 is optimal. In the results here the values are rescaled between 0 and 1.

6 Comparative Results

In this section, we present comparative results of SCENT with neural gas and the SOM. SCENT produces tree structures, but it can be evaluated as a flat clusterer by only considering the leaf nodes, and therefore the *gamma* value is calculated using only these leaf nodes.

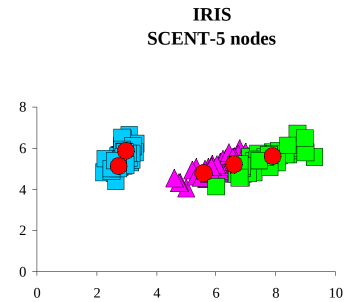
Table 1 shows the average and standard deviation of the *gamma measure*, over all 16 data sets and for all three networks.

Table 1. Average *gamma* measures of the classifications produced by the 3 neural network models tested. 16 data sets are used. The higher values of *gamma* are better.

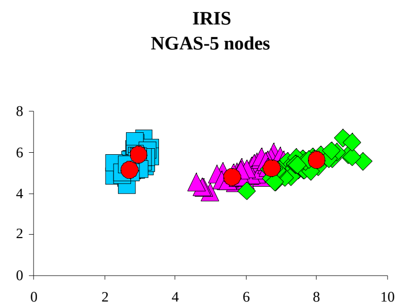
Neural classifiers	Average	Standard Deviation
SCENT	**0.761**	**0.193**
Neural gas	0.733	0.267
SOM	0.686	0.310

Table 1 indicates that SCENT produced the best mean *gamma* value and also had the least variation in performance over the data sets. This is a significantly strong result as Neural Gas normally produces very good classifications. Furthermore SCENT gives more information about the data since it produces a hierarchical classification as well as a flat classification.

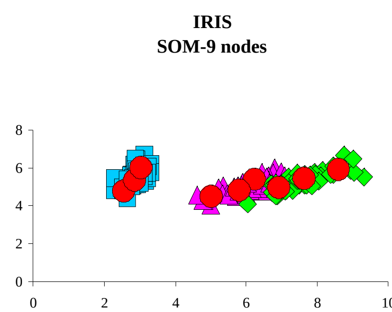
Figures 7 and 8 show the position of the leaf nodes for all three networks on two real world data sets: the well known IRIS data set (150 4-ary vectors) and a data set representing the chemical composition of wines from Italy (178 13-ary vectors). Since the data is more than 2-dimensional it is shown in a PCA projection. As can be seen all three models place the leaves in reasonable positions. It is notable that SCENT places a larger number of nodes above the left hand cluster in the WINE data, due to the density of this part of the data.



(a) SCENT with 5 leaf nodes



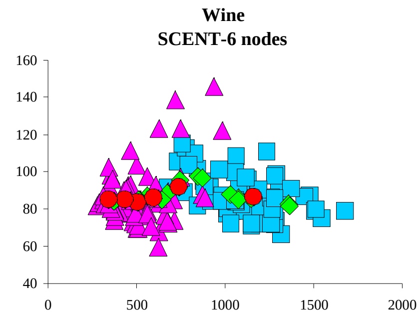
(b) Neural gas with 5 nodes



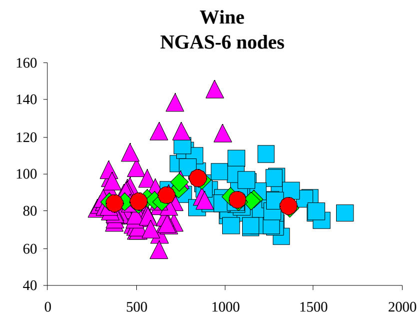
(c) SOM with 9 nodes (3×3 grid)

Figure 7. Leaf positions of three different networks of IRIS which consists of 3 main classes using the non-constrained parameter settings. Circles represent nodes produced by a particular mode, different shapes represent different classes in the

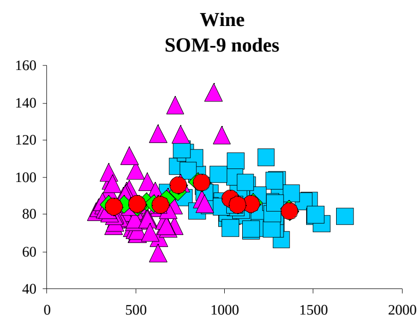
data.



(a) SCENT with 6 leaf nodes



(b) Neural gas with 6 leaf nodes



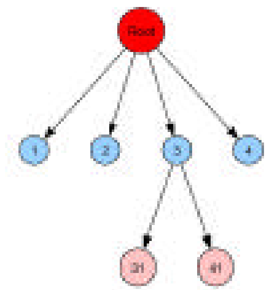
(c) SOM with 9 nodes (3×3 grid)

Figure 8. Leaf positions of three different networks of WINE which consists of 3 main classes using the constrained parameter settings.

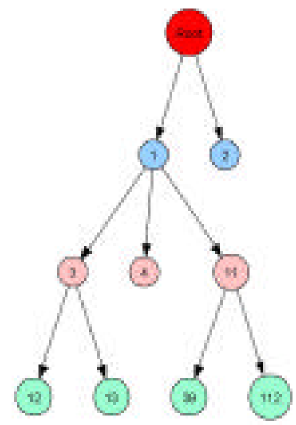
In order to illustrate SCENT's abilities as a hierarchical clusterer we examine the full trees produced for the previous two data sets. Figure 9 shows two representative trees.

For the IRIS data set SCENT has produced 4 level one nodes, two in each of the disjoint clusters. The right hand cluster has an area where two types of iris overlap and here SCENT has generated two level 3 nodes to represent this.

SCENT has represented the WINE data with an unbalanced tree, capturing the differential densities of this data.



(a) IRIS



(b) WINE

Figure 9. The tree structure produced by SCENT for the IRIS and WINE data sets.

7 Discussion and Conclusion

SCENT is a dynamic neural tree classifier. It is a stochastic version of its deterministic precursor, in which both decisions and generated values are made non-deterministic. Extensive testing has shown that the addition of stochasticity gives a performance benefit, particularly for data with unusual structure [10]. This paper has concentrated on its flat clustering ability in order to compare its performance with other well respected clusterers.

The most important result presented here is that of Table 1. This shows that SCENT performs as well, if not better, than both Neural Gas and the SOM, in terms of the *gamma* measure over the varied data sets used here. Not only does SCENT have a slightly better mean *gamma*, but the variability over the data sets is significantly smaller. Since SCENT is dynamic, and therefore does not need the topology of the network to be specified (neither the number of nodes nor the tree shape is prespecified), and produces reliably good classifications it can be recommended as a data exploration tool.

In addition to this it also produces a tree that can show any hierarchical structure contained in the data. Overlapping clusters can be disambiguated by subtrees and this is illustrated with the IRIS data set. High density regions of a data set can be decomposed by lower levels of the tree as shown for the WINE data set.

Acknowledgments

Part of this research work is financially supported by the Royal Thai Government.

.

References

[1] R. G. Adams. K. Butchart. And N. Davey, “Classification with a Competitive Evolutionary Neural Tree,” *Neural Networks*, Vol. 12, pp. 541-551, 1999.

[2] K. Butchart, *Hierarchical Clustering Using Dynamic Self Organising Neural Network*, Ph.D. Thesis. University of Hertfordshire, 1996.

[3] N. Davey, R.G. Adams. and S. G. George, “The Architecture and Performance of a Stochastic Competitive Evolutionary Neural Tree,” *Applied Intelligence*, Vol. 12, No. 1/2, pp. 75-93, 2000.

[4] A. D. Gordon, *Classification*, Chapman & Hall, London, 1999.

[5] J. A. Hartigan, *Clustering Algorithms*, John Wiley & Sons, USA, 1975.

[6] J. Hertz, A. Krogh. and R. G. Palmer, *An Introduction to the Theory of Neural Computation*, Addision Wesley, USA, 1991.

[7] T. Kohonen, *Self-Organizing Maps*, 2 edn. Berlin: Springer-Verlag, 1997.

[8] T.M Martinetz,, S.G. Berkovich and Schulten, K.J. “Neural Gas Network for Vector Quantization and its Application to Time-Series Prediction,” *IEEE Transactions on Neural Networks* **4**, 558-569, 1993.

[9] G. W. Milligan. and M. C. Cooper, “An Examination of Procedures for Detemining the Number of Clusters in a Data Set,” *Psychometrika*, Vol. 50, No. 2, pp 159-179, 1985.

[10] W. Pensuwon, *Stochastic Hierarchical Dynamic Neural Networks*, Ph.D. Thesis. University of Hertfordshire, 2001.

[11] E. Yair, K. Zeger and A. Gersho, “Competitive Learning and Soft Competition for Vector Quantiser Design,” *IEEE Transactions on Signal Processing* **40**, 294-308, 1992.