

Abstract A system with some degree of biological plausibility is developed to categorise items from a widely used machine learning benchmark. The system uses fatiguing leaky integrate and fire neurons, a relatively coarse point model that roughly duplicates biological spiking properties; this allows spontaneous firing based on hypo-fatigue so that neurons not directly stimulated by the environment may be included in the circuit. A novel compensatory Hebbian learning algorithm is used that considers the total synaptic weight coming into a neuron. The network is unsupervised and entirely self-organising. This is relatively effective as a machine learning algorithm, categorising with just neurons, and the performance is comparable with a Kohonen map. However the learning algorithm is not stable, and behaviour decays as length of training increases. Variables including learning rate, inhibition and topology are explored leading to stable systems driven by the environment. The model is thus a reasonable next step toward a full neural memory model.

Keywords Compensatory Hebbian Learning; Categorisation; Spontaneous Neural Spiking; Neural Fatigue; Point Neural Model; Self-Organisation

Post and Pre-Compensatory Hebbian Learning for Categorisation

Christian R. Huyck and Ian G. Mitchell
Middlesex University - Dept. of Computer
Science
London - UK
c.huyck@mdx.ac.uk 44-208-411-5412

the date of receipt and acceptance should be inserted later

1 Introduction

Modelling the brain at a neural level can lead to a better understanding of neural and psychological function, but can also lead to better AI systems. In particular, the authors are interested in developing agents and memory systems with good neuro-psychological fidelity. One task that these systems must perform is to learn new categories, and to be able to categorise novel input into one of several categories. Categorisation is also a standard machine learning task.

The scientific community has a growing understanding of brain function, and increasingly accurate models of that function. Nonetheless, current understanding and models of brain function are far from complete. In particular, it would be useful to develop a model of a novel categorisation task with good neuro-biological fidelity.

This paper develops a neural model of an abstract categorisation task that is commonly used as a machine learning problem. It takes into account some constraints from neural behaviour, but is not biologically or psychologically accurate. It is instead a step toward a neuro-psychologically faithful model that is also a reasonable machine learning algorithm.

Biological neuron models that perform psychological tasks have dynamics on two timescales that interact with each other. The first dynamic is the activation dynamic, where activation, typically involving neural firing, spreads from neuron to neuron. The second dynamic is a learning dynamic, where connections between neurons are changed. The activation dynamic is reviewed in section 2.2, while the Fatiguing Leaky Integrate and Fire (FLIF) neural model used in this paper is described in section 3.1. The learning dynamic, which is based on Hebbian learning rules, is reviewed in section 2.3. The post and pre-compensatory Hebbian learning rules used in this paper are described in section 3.2.

Section 4 describes simulations to categorise irises. Two modes are described in this section, one with two subnets and one with three subnets. Spontaneous firing from the neural model enables a second subnet to be used for categorising. A third subnet is added so that the categorisation is done based on a direct measurement of neural firing, and this makes the categorisation more plausible as a neuro-biological model of

Address(es) of author(s) should be given

the task. This system is self-organising and is compared to a Kohonen self-organising map.

Section 5 describes a four subnet system and some problems of learning stability. Solutions to this homeostatic problem are explored, eventually leading to systems that, as far as simulations have explored, continue to improve the longer they learn. Testing the homeostatic system on another categorisation task, yeast, gives poor results. Section 6 concludes considering how this model might be included in a full neuro-psychological memory system.

2 Background

There is a large research community exploring machine learning, and one exploring neuro-psychology. This section provides some context from these communities relating to the task explored later in the paper, namely, learning categories and categorising novel data with some degree of neuro-psychological fidelity.

2.1 Machine Learning

Over the last few decades, machine learning has become industrially viable. For example, Google's search engine takes advantage of machine learning. One common machine learning task is categorisation, which is performed by learning a category from previously categorised data, then, categorising novel data. This has frequently been done using the machine learning repository stored at the University of California at Irvine (Bache and Lichman, 2013). There are hundreds of different tasks, and these tasks are used as benchmarks to compare machine learning algorithms.

There are many machine learning algorithms, for example, multi-layer perceptrons (MLPs) learning via backpropagation, genetic algorithms and statistical methods. New algorithms are being developed, but there is a formal proof that no algorithm is better than others on all data sets (Wolpert and Macready, 1997). Consequently, new algorithms can be useful. Moreover, it is often useful for a data mining expert to explore any particular task to improve overall performance.

While there are many connectionist systems that are called neural networks, for example MLPs, these are not usually faithful models of biological neurons, and are supervised and therefore not self-organising. Since new machine learning algorithms can be useful, any new algorithm is useful, and making use of biological models could relatively easily provide useful machine learning algorithms.

2.2 Neural Models

Modelling the brain at a relatively fine granularity requires a model of neurons. There is an increasing understanding of neural behaviour, with two large classes of models: compartmental models and point models. Compartmental models, almost without exception, more accurately reflect biology than do point models, but they are expensive to simulate. They break the neuron into electrical compartments and then use physical laws to determine the electrical behaviour. The canonical compartmental model is the Hodgkin and Huxley (1952) model, but there are more modern models (see Brette

et al., 2007, for a review). Of course, even with compartmental models, there is the question of what to model. For instance, is neural fatigue (which is considered in section 3.1, see Kohn, 2007, for a review) included in the model?

Point models consider the entire neuron as a single mathematical equation (a point), though synapses are modelled separately. The canonical point model is the integrate and fire model (McCulloch and Pitts, 1943; Abbott, 1999). This model has been extended, and the leaky integrate and fire model is widely used (see Amit, 1989, for an overview). Different biological neurons have widely varying behaviours; for example, given a constant input some spike regularly, and some respond with bursts of spikes. The Izhikevich (2004) neuron exhibits a wide range of these behaviours, is quite efficient, and has become a popular point neural model. The Boltzmann machine (Ackley et al., 1985) is another model, as are a range of continuous output neurons (e.g. O’Reilly, 1996). The FLIF model used in this paper (see section 3.1) is an extension of the leaky integrate and fire model, which like the Boltzmann machine can fire without input.

2.3 Hebbian Learning

It is theorised that a great deal of learning in the brain is based on modification of synaptic strengths. Biologically this can occur by the modification of synapse and dendrite shape, an increase in the capacity to excrete neurotransmitters, and other mechanisms. It is typically modelled as a synaptic weight that increases causing the pre-synaptic neuron to send more activation to the post-synaptic neuron. Modification is done in a Hebbian fashion (Hebb, 1949); if the pre-synaptic neuron tends to cause the post-synaptic neuron to fire, the strength tends to increase. Learning based on neural growth and death, and on synaptic growth and death are also influenced by how often neurons fire and co-fire.

This Hebbian rule is underspecified, and though there is physiological evidence, that evidence is incomplete. Currently, spike-timing dependent plasticity (STDP), having solid biological support (Bi and Poo, 1998), is widely used in simulations. This increases the weight if the pre-synaptic neuron fires before the post-synaptic neuron, but decreases it if the post-synaptic neuron fires first.

STDP requires a relatively tight time granularity for spiking. A longer standing theory is based on firing rates and is thus suitable to continuous output neural models (Bienenstock et al., 1982). There has also been work linking this to STDP (Izhikevich and Desai, 2003; Bush et al., 2010). The neural model described below (section 3.1) is based on single neural firing events and uses a relatively coarse grained spiking behaviour and so is not well suited for a learning rule based on STDP.

There has also been research into mathematical models of neural learning. A particularly good example is Fyfe (2005), which shows how Hebbian-like rules can be used to learn neural systems that can do the complex tasks of principal and independent component analysis. This makes use of anti-Hebbian learning to decorrelate neurons. This means that in some cases neurons that frequently co-fire have their mutual synaptic strengths weakened.

Decorrelation is important because it prevents all of the neurons eventually responding to the same input. The SOM learning algorithm also provides some decorrelation (Kohonen, 1997). It uses the “Mexican hat” function, where elements that are near to the location of the projected input in the SOM space, but not too near, are

pushed away. This is decorrelation. Compensatory learning (see section 3.2) can also provide decorrelation. If two neurons co-fire frequently, but already have a great deal of synaptic strength, the times when they do not co-fire may be sufficient to eventually reduce their mutual synaptic strength.

Clearly there is insufficient space in this paper for a complete review of neural learning models. These models must consider both biological constraints and mathematical constraints. Understanding of these rules is incomplete but a picture is emerging.

3 Model

Broadly speaking, the components of the model used in the simulation are the neural model, the learning algorithm, and the topology (the way the neurons are connected). The system can be found at <http://www.cwa.mdx.ac.uk/chris/hebb/iris/iris.html>.

3.1 FLIF Model

The FLIF neural model is a point model in the family of integrate and fire (McCulloch and Pitts, 1943) models. Each neuron has two dynamic variables, activation A and fatigue F . An integrate and fire model is described by equation 1. The neuron j integrates activity from other neurons i that fired in the last cycle (V_i) weighted by the synaptic strength w_{ij} from the firing neuron to j ; in these simulations strength ranges between 0 and 1. Neuron j fires if activity surpasses a threshold θ . Fatigue is described below.

$$\theta + F_j < A_j = \sum_{i \in V_i} w_{ij} \quad (1)$$

The model is discrete and runs in cycles that roughly correspond to 10 ms. of time. It is leaky as described by equation 2, so if neuron j does not fire in one cycle, it retains some of the activation for the next cycle. Without input, activation decays from step $t - 1$ to step t , being divided by a constant $D > 1$.

$$A_j^t = A_j^{t-1}/D \quad (2)$$

The neuron fatigues each step it fires. Each neuron has a fatigue value that is increased by a constant F_c each step it fires. The neuron's fatigue is added to the threshold, producing a dynamic threshold, so neurons that frequently fire require more activation to fire.

When a neuron does not fire, its fatigue is reduced by a constant F_r in each step that the neuron does not fire. In earlier versions, the fatigue value, F , never went below zero, but the version used in the simulations described in this paper allows fatigue to be negative. When a neuron is hypo-fatigued, it will fire when fatigue is negative enough ($-F > \theta$ as in equation 1 where activity is 0) even if it has no activation. In this model, if the neuron fires and fatigue is less than -.25, the fatigue is halved as described in equation 3. Otherwise, it is increased by F_c as usual.

$$F_j^{t+1} = F_j^t/2 \quad (3)$$

This leaves four parameters to describe the neural model. Threshold θ is 2.2; decay D is 1.12; fatigue increase F_c is 0.45; and fatigue recovery F_r is 0.01. In past simulations, these were free parameters for simulation, but these values have been selected to fit the firing behaviour to biological neurons. The particular neurons modelled were rat somatosensory neurons under a widely varying direct current injection regime. Similarly, the fatigue recovery rule's inclusion, equation 3, led to a closer fit to the biological firing behaviour (Huyck and Parvizi, 2012). Over 90% of simulated spikes mapped to the actual biological spikes with an average difference of less than two cycles (17 ms.).

In addition to an improved fit to neural data, this fatigue model provides a mechanism for neurons not directly stimulated to fire. For neurons to be included in a Hebbian circuit, they need to fire. The synaptic weights to the neurons that are not directly stimulated can increase due to Hebbian learning because these neurons fire. In earlier simulations, neurons have been selected randomly to fire spontaneously. With the current model, spontaneous firing is more principled.

3.2 Compensatory Learning

The learning mechanism is another component of the model. While evidence points to biological modification of synaptic weights being Hebbian, this leaves an infinite range of possible rules. The simulations below use a compensatory learning rule to model the biological behaviour. In addition to the firing behaviour of the two neurons a synapse connects, a compensatory rule takes into account the total weight of the synapses in these neurons, forcing the total weight toward a target total in conjunction with the firing behaviour. The authors have used a compensatory rule based on the total weight from the pre-synaptic neuron's synapses in earlier work to learn hierarchical categories (Huyck, 2007). The simulations described below make use of two variations of the compensatory rule, one based on the sum of synaptic weights to the post-synaptic neuron, and the other prior rule based on the sum of weights from the pre-synaptic neuron; these are termed post-compensatory and pre-compensatory, respectively.

Hebbian rules are typically a combination of a rule for neurons co-firing, and one for when they do not. Equation 4 is used when the neurons co-fire, and equation 5 when the pre-synaptic neuron fires and the post-synaptic neuron does not. When the pre-synaptic neuron does not fire, the weights do not change.

$$\Delta_+ w_{ij} = R * C[(1 - w_{ij}) * 10^{(W_B - W_k)}] \quad (4)$$

$$C[X] = \begin{cases} 1, & \text{if } X > 1 \\ X, & \text{otherwise} \end{cases}$$

$$\Delta_- w_{ij} = -R * C[w_{ij} * 10^{(W_k - W_B)}] \quad (5)$$

In equations 4 and 5, R is the learning rate, which is 0.01 in the simulations below with one explicitly noted exception. W_B is the neuron's target total synaptic weight, its saturation base. W_k is the neuron's current total synaptic weight. C is a clipping function capping its output at 1. Thus a synaptic weight can change by at most the learning rate in a given cycle, and this is often the case in the beginning of a simulation as the initial synaptic weights are low. The rules restrict synaptic weights to values between 0 and 1. W_k is the total synaptic weight leaving neuron i for pre-compensatory

learning and the total synaptic weight entering neuron j for post-compensatory learning; W_B is thus the total target weight leaving i for pre-compensatory, and the total entering j for the post-compensatory rule.

One advantage of the FLIF neural model described above and the compensatory learning mechanism is that they are exploring neural dynamics in the 10ms range. Other neural simulations use a finer time grain. It is thus quite efficient to simulate.

3.3 Categorising Other Domains with this Model

Quite some time ago, a variant of the system described below was used to categorise US Congressional Representatives and for information retrieval (Huyck and Orengo, 2005). In these cases, all of the neurons were directly activated by the environment at one time or other during training. In the current paper, the principled spontaneous activation of neurons allowed the spread of neural circuits beyond the sensory interface, so neurons that were not directly stimulated could be used.

The authors have used the basic mechanism described in section 4.1 to categorise items in two domains. Unlike the simulations used in the Congressional task, these simulations used two subnets, with an internal subnet that was not directly activated by the environment (the input vectors). These domains were the yeast domain (Huyck and Mitchell, 2013), and the car domain (Mitchell and Huyck, 2013), which like the iris domain below, were from the UCI machine learning repository (Bache and Lichman, 2013). All three data sets have input features and categories, and n-fold testing was used on all three. A 2-fold test partitions the data into 2 parts, trains the system on one part, and then tests on the other. Then the system is retrained from scratch on the second data set, and tested on the first. The input features for the car simulations were modelled as discrete categories. The input features for the yeast, like the iris features below, were modelled as continuous variables.

In all three cases, the neural simulations were compared to a self-organising map (Kohonen, 1997) (KSOM) simulation run by the authors (see table 1). Four-fold tests were run on the car data, and 10-fold tests were run on the yeast data for comparison with earlier machine learning systems. A comparison to a multi-layer perceptron learning via back percolation (MLP) is shown in the table. In both cases, the biological neural network performed near to other standard machine learning algorithms.

	Algorithm	Result
Car Data	KSOM	75.75 %
	BioNeural Net	78.92%
Yeast Data	MLP	57 %
	KSOM	56.00 %
	BioNeural Net	52.81%

Table 1 Categorisation results of yeast and car data

4 Iris Categorisation

This paper is using perhaps the most widely tested categorisation benchmark, the iris data set from the UCI repository (Bache and Lichman, 2013), a generally accepted benchmark. The iris data set is based on the classification of three species of iris: *I. setosa*; *I. virginica*; and *I. versicolor*. The data set consists of 50 samples of each species with four parameters: sepal width and length; and petal width and length measured in centimetres. The simulations in this paper use a 2-fold test. The data was partitioned so that both sets had 25 items of each category, but this was randomly selected. All tests use the same data division.

4.1 Categorisation via Pearson Measurements

A straightforward translation from the earlier model, for categorising yeast, is used. There is an Input subnet that is directly stimulated based on the particular training item. There are 110 neurons for each feature, and 20 for each category for a total of 390 neurons. 10 neurons for each input feature value, and 20 neurons for the category (all from the Input subnet) are stimulated during training for 40 cycles. These externally activated neurons are given sufficient activation to fire each cycle, in essence, being clamped on. The input features are normalised so they range from 0 to 1. The neurons around the feature value are stimulated. There are 20 neurons associated with each category input and all are stimulated.

The topology is also similar to the yeast model. All neurons are excitatory, with the input neurons not fatiguing. The network is divided into two subnets, the Input subnet and the SOM subnet. Each neuron in the Input subnet synapses to 20 random neurons in the SOM subnet. Each of the 1000 neurons in the SOM subnet synapses to 10 random neurons in the subnet, with no self-connections. All of these weights are initially low. The neural constants are those derived from biological data: threshold θ is 2.2; decay D is 1.12; fatigue increase F_c is 0.45; and fatigue recovery F_r is 0.01 (see section 3.1).

Learning for neurons in the Input subnet was post-compensatory, and learning for neurons in the SOM subnet was pre-compensatory. Saturation base for the Input subnet was 5, and the SOM subnet was 1. The learning rate was .01.

During training, items were presented in 75 cycle epochs. A test item was presented for 40 cycles, and followed by 35 cycles of no external stimulation. Then the next item was presented. This was done for 20000 cycles. There were 150 training items, and 267 training epochs, so most items were presented twice.

Testing was broken into two phases, and in both learning was turned off. Before each testing epoch in both phases, the network was reset so that each neuron had 0 activation and 0 fatigue, though synaptic weights remained unchanged. In the first phase, each training item was presented for an epoch. During this epoch, the number of times each neuron in the SOM subnet fired was recorded. This firing behaviour was used in the second testing phase to categorise the test items.

In the second testing phase each test item was presented for an epoch, and its firing behaviour was recorded. A Pearson's Product Moment Correlation was calculated between the firing behaviour of each training item and the test item. The test item was categorised as the category associated with the most correlated training item.

Both folds were run on 100 networks with an average performance of 93.67%. The variance was 1.02%, and all nets got between 65 and 73 of the possible 75 answers correct (see table 2).

The best result the authors found in the literature used a form of backpropagation, the weight limit algorithm, with spiking neurons (Wu et al., 2006). The authors duplicated the tests from the simulated biological nets on the same training and test data on a 9*8 node Kohonen SOM with a learning rate decreasing linearly from 0.05 to 0.01 over 50,000 training cycles. The neighbourhood started with 2/3 of node-to-node and the grid was rectangular and non-toroidal. The results for the Kohonen SOM were run ten times and yielded a mean of 92.73% with a variance of 0.07%.

These results merely confirm that the basic simulated biological neural mechanism is relatively successful on another standard benchmark. Unlike the KSOM, the learning rate remains constant during training in all of the biological neural net simulations.

Algorithm	Result	
SOM	92.73%	
Weight Limit	96.6%	
	Pearson	Firing
2 Subnets	93.67%	
3 Subnets	93.50%	84.63%
4 Subnets	93.45%	86.89%
4 Subnets Inhib	93.53%	89.45%

Table 2 Categorisation results of iris data.

4.2 Categorisation via Neural Firing

Categorising using Pearson measurements is not neuro-psychologically viable. That is, when a human categorises an object, they do not measure the firing behaviour of neurons then compare that behaviour with the firing recorded when other known items of that class occurred. What is needed is a mechanism where particular neurons fire for items in a particular category. Fortunately, the above mechanisms are capable of doing this relatively accurately; as shown below, FLIF neurons, learning by pre and post-compensatory learning, can categorise based entirely upon neural firing.

The network is extended to include a third subnet, the Output subnet, as shown in figure 1. The SOM subnet takes its name from its self-organising properties; it is, in a general sense, a self organising map. Neurons in the Output subnet are stimulated during training, and then measured during testing. Initially this fails to categorise well because during training neurons from categories aside from the presented category become active due to firing in the SOM subnet. Running on one subnet for each fold, the firing measurements gets 40.00%, just above chance. The Pearson measurement is still reasonable at 93.33%.

An inhibitory neuron is added, and the net then categorises reasonably well. More explicitly, the Output subnet contains 150 neurons, 50 for each category. 20 of these are randomly selected to be stimulated during training, and no neurons are stimulated in the Input subnet for the category. All of these Output neurons are spontaneously firing FLIF neurons, and they learn using the pre-compensatory learning rule. The

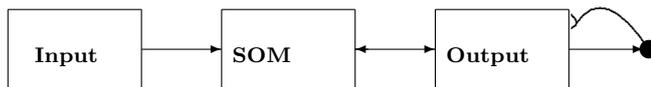


Fig. 1 Gross Topology of categorisation simulations. Boxes represent subnets, the circle the Inhibitory neuron, arrows excitatory connections between subnets, and the spaded arc inhibitory connections.

number of synapses between the Input and SOM subnets, and the number of internal SOM synapses remain the same, 20 and 10. For each neuron in the SOM subnet there are 10 connections to randomly selected neurons in the Output subnet, each Output neuron has 10 synapses with SOM neurons, and each Output neuron has 10 internal connections. The saturation base (see equations 4 and 5) remains 5 for the Input subnet but is increased to 2 for neurons in the SOM subnet, and is 10 for neurons in the Output subnet. This extra synaptic strength facilitates firing in the Output net during testing. The number of connections has been selected by a relatively brief manual search of the space. It is probably not optimal.

The inhibitory neuron merely polls all of the neurons in the Output subnet. For every neuron more than 20 that fires in the Output subnet in any cycle, each neuron receives -0.5 units of activity in the subsequent cycle.

Training and testing proceeds as above, but during testing, the number of Output neurons that fire in each category are recorded. In each test epoch, the category that is selected is the one that has the most firings.

The test was run on 100 nets for both folds, shown above in table 2. The Pearson result based on the SOM subnet were 93.50% with a variance of 0.62%. Categorising using firing behaviour was below this, 84.63%, with a larger variance 4.51%.

During testing, Input neurons are fired externally and transfer activation to the SOM subnet. The SOM and Output subnets have learned associations and use these to transfer firing from the Input subnet to the SOM subnet and onto the Output subnet. This associative transfer is the basis of categorisation.

The SOM subnet still stores significant categorical information as is evident from the ability to categorise using Pearson correlations. Moreover, the overall net is capable of categorising based solely on neural firing. This shows that the combination of pre and post-compensatory learning is able to develop a neural matrix that extracts the information stored in the network that is used to derive the Pearson correlations.

Moreover, categorising using the Pearson measurement requires the network to see each training item once after learning is turned off. Categorising by neural firing does not require this phase, and thus can be quicker.

4.3 Categorisation and Self-Organisation

The iris data set has been categorised by many machine learning techniques, including Kohonen's Self Organising Maps (KSOMs). The authors have re-run the test to ensure the same training and testing samples are presented so that the comparison is as direct as possible. The results are in tables 2 and 3. While the results are comparable, the results from the simulated biological networks are slightly better.

	Units	Mean	Variance	Best
1	72	92.73%	0.07%	72/75
2	64	91.07%	0.15%	72/75

Table 3 KSOM results: the mean and variance is on a two-fold test over 10 runs. First and second row has 9*8 and 8*8 rectangular grids, respectively. The best result is the best of the 10 runs.

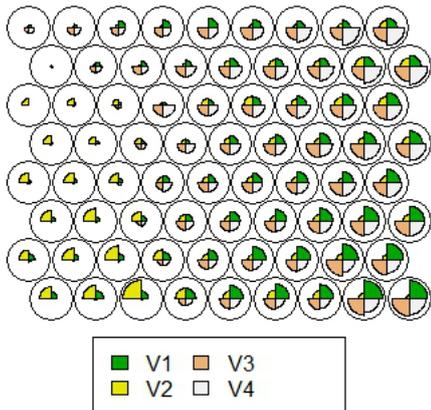


Fig. 2 KSOM for best result in table 3, showing parameters, V_1, V_2, V_3 and V_4 , and corresponding proportional weights inside respective nodes. This shows how the inter-layer weights self-organise and form neighbourhoods e.g. the bottom right neighbourhood weights have high values for parameters V_1 (green) and V_3 (pink) compared with the bottom left neighbourhood that has high values for V_2 (yellow) and low values for V_1 and V_3 . Produced using Wehrens and Buydens (2007).

The different results in table 3 stem from differing numbers of units. The 72 unit system performs better than the 64 unit system. Figure 2 illustrates, after training, how the weights are organised. This figure shows how different units align to different areas of the parameter space. Adjacent neurons tend to be quite similar.

Table 4 contrasts the biological neural network approach with the KSOM approach. The main difference is that in the KSOM exactly one unit wins on any given input, while in the biological neural net a subset of neurons wins and the classification is based on this subset.

Like the KSOM the biological neural nets described in this paper are unsupervised but are they self-organising? Kohonen (1997) defined self-organising, or competitive-learning, as follows:

...the cells, in the simplest structures at least, receive identical input information. By means of lateral interactions, they compete in their activities. Each cell or cell group is sensitized to a different domain of vectorial input signal values, and acts as a decoder of that domain.

Is the simulated biological neural net a self-organising map? From the definition above the proposed net satisfies all the statements, which are broken down from the quote above, as follows:

	BNN	KSOM
Learning	distributed subset of randomly connected nodes organised to represent training data	structured neighbourhood of connected nodes organised to represent training data
	trains a subset of randomly connected nodes to spike by changing connections based on Hebbian rules	trains a neighbourhood of nodes to minimise the error between input vector and connection values
Recall	one node may belong to one or more different classifications	each node belongs to exactly one classification
	a subset of randomly connected nodes represents a classification	neighbourhood of structured connected nodes represents a classification

Table 4 KSOM and Bio-Neural Net(BNN) Differences

- *receive identical input*: while there are random connections between layers these do not change over time. As with KSOM only the values of those connections change, whilst the stimulus input is identical.
- *by means of lateral interactions*: lateral interactions are neural firings and spread of activation.
- *they compete in their activities*: they compete via firing neurons and compensatory learning.
- *cell group is sensitised to a different domain of vectorial input*: section 3.2 shows how the synapses change so that the neurons become sensitised, and the cell group emerges to categorise inputs.
- *cell group acts as a decoder of that domain*: the group (subset) of cells representing the categorisation indicate that the biological neural net is able to decode both by Pearson measurements and by neural firing alone.

5 Categorising with Four Subnets, and Stability

The addition of a fourth subnet, the Hidden subnet, was relatively straight forward. It was placed between the SOM and Output subnets, and was roughly a duplicate of the SOM subnet; it consisted of 1000 neurons, was made of spontaneously firing FLIF neurons, and received no external stimulation. As in the 3 subnet simulations, each neuron in the SOM and Output subnets had synapses with their own subnet, and neurons in the Hidden subnet also had 10 internal synapses. There continued to be 20 synapses from each neuron in the Input subnet to neurons in the SOM subnet. There were 10 synapses between neurons in the Output and Hidden subnets, 15 synapses from SOM neurons to Hidden neurons, and 10 back. The simulation continued to use an inhibitory neuron for the Output subnet.

Saturation base remained 5 in the Input subnet, 2 in the SOM subnet, and 10 in the Output subnet. It is 4 in the Hidden subnet, again supporting more firing.

The results are shown in table 2. Categorisation was correct 93.45% with the Pearson’s measurement and 86.89% with the firing results. Variance was 0.53% and 3.80% respectively.

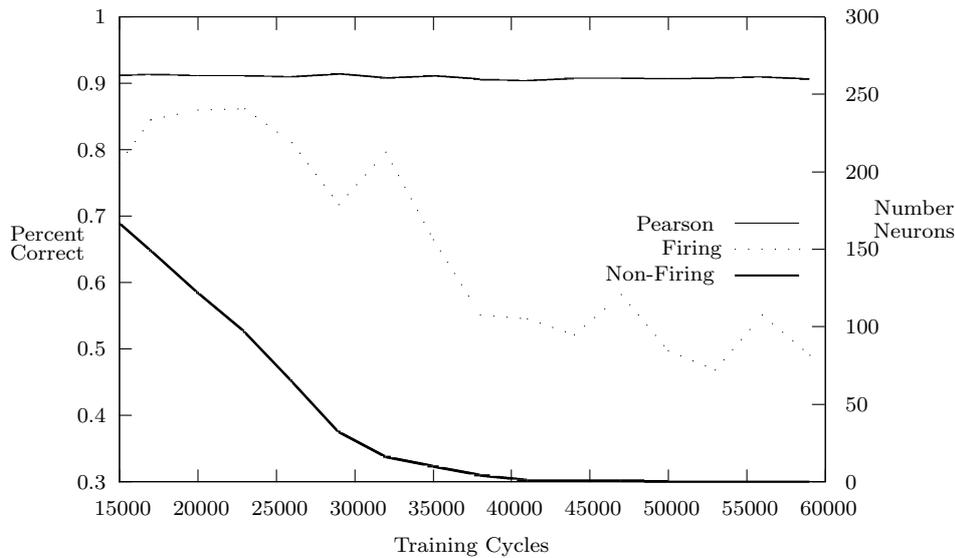


Fig. 3 Results of 10 runs of one fold by training duration. This indicates that for measurement by firing there is gradual improvement to a point, then radical decline. Categorisation by Pearson continues to perform well. The right axis and non-firing line refer to the number of neurons not firing during Pearson categorisation.

5.1 Continued Learning Breaks the Categoriser

While results after training for 20000 cycles are good, they do improve as training increases. However with this type of network, at some point around 30000 cycles, increase in training length leads to a radical fall in the ability to categorise. This is shown in figure 3. This figure represents results of categorisation of one fold run on 10 nets with varying durations. Starting from 14000 cycles of training, there is a datapoint for every 3000 cycles up to 59000 cycles. The results from firing decline from a high just above 86%. This declines to near 40%, though it recovers to above 50% and this continues to at least 3,000,000 cycles (see the *No Inhibition* line in figure 7). Categorisation by the Pearson measurement remains high. Further tests up to 3,000,000 cycles show that this continues for some time, and may remain stable.

Further analysis of the simulations shows that the problem rests with the internal subnets becoming self-absorbed. The synaptic weights grow so that they support self-sustained firing in the internal subnets. At this stage, the associative transfer from Input to Output begins to become disrupted until it eventually fails altogether.

Support for this line of reasoning is provided by a lack of firing of neurons in the SOM subnet during testing shown on the right axis of figure 3. To calculate the Pearson measurement, after learning is turned off, each neuron's firing behaviour is recorded for each of the 75 training samples; the reported result is on one network trained for varying times. The firing behaviour is later used to see which training example has the closest firing behaviour to a given test example. Somewhat surprisingly, some neurons do not fire during any of these testing runs. Since neurons will not fire spontaneously in

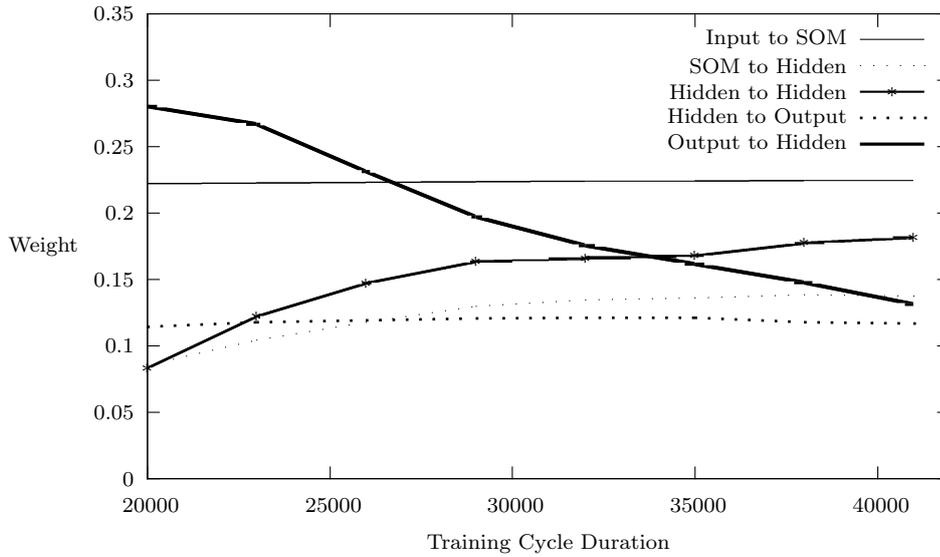


Fig. 4 Average synaptic weight between and within selected subnets.

this phase (because fatigue is reset at the beginning of the epoch), this lack of firing is due to insufficient synaptic input. These neurons are not used for categorisation. The number of such non-firing neurons declines as training length is increased. Eventually all neurons fire during some tests, but this is as categorisation by firing has become ineffective. Neurons still fire sparsely, with most uninvolved in any particular input, and those that are involved typically firing less than 10 times.

The synapses within the internal subnets and to the Output subnet are necessary for categorisation by firing. The Input subnet neurons are fired externally, and they transfer firing directly to the SOM subnet. Initially, all synaptic weights are low and the neurons in the internal subnets fire due to hypo-fatigue. In the first few thousand training cycles, the weights from the Input subnet to the SOM subnet grow, as do the weights from the Output subnet to the Hidden subnet. The weights within and between the SOM and Hidden subnets increase but remain small. While the internal subnet connections also cause firing in the SOM subnet, the continuing good performance of the Pearson measurement implies that the Input to SOM synapses remain relatively stable. This is evident in figure 4. This figure shows the average weight of a synapse from one subnet to another, on one particular net. The measurement is done at the end of training as the training length is increased. The weights from the Input to SOM subnets average .222 at cycle 20000, and increase ever so slightly to .224 by cycle 41000. By cycle 41000, the categorisation performance of this net by firing has rapidly declined to near 55%. It is not the case that individual synaptic weights are saturated; instead they actually decline.

The internal subnet synapses transfer firing to the Output subnet, which is then measured to categorise. The internal synapses store an association between firing in the SOM and Output subnets. As the co-firing of the neurons in the internal subnets increase (see figures 5A and 5B), their own internal dynamics begin to replace the

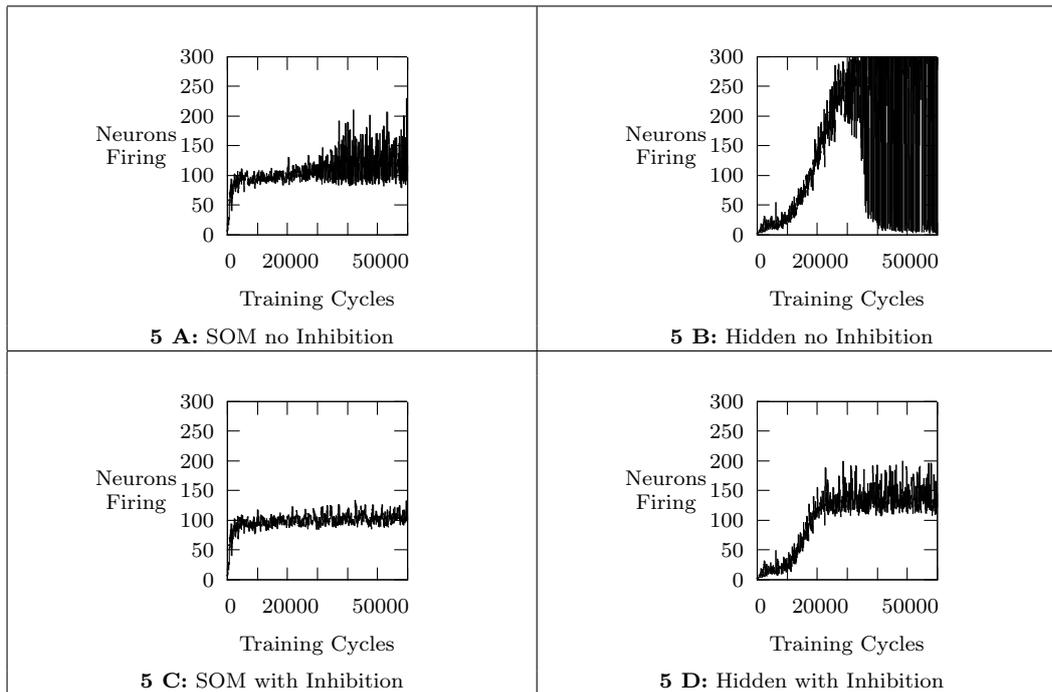


Fig. 5 The Maximum Number of Neurons that fire per epoch in the SOM and Hidden subnets. A and B represent runs on nets with no internal inhibition, with A representing the firing the SOM subnet and B in the Hidden subnet. C and D represent runs with inhibition on the internal subnets (see section 5.2).

dynamics of the associative transfer. These weights increase as the performance begins to decline; for example the average synaptic weight from the SOM subnet to the Hidden subnet increases significantly from .086 to .138, and the average weight within the Hidden subnet more than doubles from .083 to .182. Similarly, synaptic strength is pulled away from neurons in the Output subnet, more than halving from .281 to .131.

This change in weight is both driven and drives an increase in the firing of neurons in the internal subnets. Figure 5 shows the maximum number of neurons firing in a subnet in any given epoch. The firing in the SOM subnet is relatively stable after an initial growth, and until the final decline in categorisation performance. The firing in the Hidden subnet increases until it begins to wildly oscillate during the period of categorisation failure. The small firing rate happens in some epochs after 35,000 cycles because the neurons are fatigued and are thus unable to fire for an entire epoch.¹

The newly increased synaptic weights lead to enough firing in the internal subnets so that these neurons can fire persistently even after the external stimulus ceases. Figure 6 shows the average number of neurons firing in the SOM and Hidden subnets between the 42nd and 75th cycle in an epoch, and between the 50th and 75th. Section 4.1 describes how each epoch consists of 40 cycles of input (external stimulation of input

¹ This elevated firing in the internal subnets contrasts with the firing in those nets when inhibition is added. Figures 5C and 5D reflect the firing behaviour when extra inhibition is added (see section 5.2).

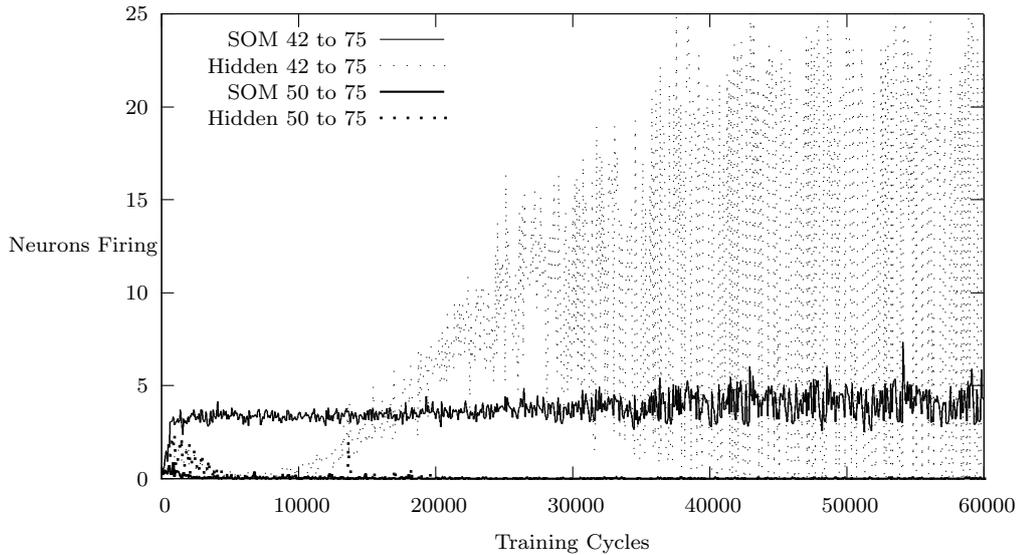


Fig. 6 Average Number of Neurons Firing Late in an Epoch.

neurons), followed by 35 cycles with no external input. Since the internal subnets do not receive any external stimulation, they do not fire in the first cycle except spontaneously. Moreover, any firing after the 42nd cycle is due to internal spread of activation or spontaneous firing due to hypo-fatigue. In the first few epochs, there is firing in both nets at all times, and this is also due to hypo-fatigue. In the SOM subnet, this is rapidly replaced at first by a brief reverberation, with neurons in the network firing driving further neural firing. The reverberation is never sustained until the end of the epoch. However, in the Hidden subnet there is a substantial amount of early reverberation. This is replaced by reverberation oscillating between high and low amounts in epochs as the performance begins to decline.

The use of all SOM neurons during Pearson measurements, the change in the character of the synaptic weights, and the reverberation of the internal neurons all indicate that the net is becoming self-absorbed. When its activation dynamics are driven by internal behaviour and cease to be driven by its inputs, then the associative transfer fails. This is a downward spiral with the network initially driven by the input, but gradually becoming more and more internally absorbed.

5.2 Homeostasis

The eventual self-absorption that the networks show arises from a lack of homeostasis (Hsu et al., 2007). Learning changes the connection weights so that the network does not respond appropriately to the environment. The activation dynamic and the learning dynamic interact in a complex way to make the system ineffective.

As the self-absorption stemmed from too much firing in the internal subnets, simulations explored the use of inhibition in these subnets. This improved behaviour almost resolved the stability problem.

An inhibitory neuron was added for the SOM subnet, and one for the Hidden subnet, in addition to the one already used for the Output subnet. In both of these internal subnets, inhibition began after 100 neurons fired. (So if 100 SOM neurons fired, 100 Hidden neurons fired, and 20 Output neurons fired in a cycle, there was no inhibition, but if an additional SOM neuron fired each SOM neuron received -.5 activation in the subsequent cycle.)

Aside from the new inhibitory neurons, everything remained as the earlier 4 subnet simulations. The system was measured after 50,000 cycles, and performed quite well (see figure 7 as the *Inhibition* line). There is still an increase in firing in the Hidden subnet, as shown in figures 5C and 5D. Thus the performance in the inhibitory 4 subnet simulation also increases and then falls. However, in this case it only falls back to around 78%. The decline begins after 100,000 cycles but instead of going to chance (33%). The simulation continued to be run for increasing times up to 3,000,000 cycles, and it seems to have plateaued at this performance. The Pearson measurement continues to do well at around 91%.

Like the earlier 4 subnet condition, the weights within the internal subnets increase. The overall net becomes self-absorbed and loses some of its ability to transfer firing from the SOM subnet to the Output subnet via synaptic associations. However, the inhibitory neuron reduces the amount of activity in the internal subnets, and consequently, the activity driven by the inputs is sufficient to maintain most stability. Performance does decline but only marginally.

This simple form of inhibition is useful, but does not solve the problem in general. Moreover, there are other mechanisms that can promote stability.

The remainder of this section shows that increased input, fewer internal connections, reduced learning rate, reduced synaptic strength and increased inhibition all promote stability in this type of system. The results over time of several of these systems is shown in figure 7. The *No Inhibition* line refers to the four subnet system described at the beginning of section 5. Like figure 3, figure 7 reflects the average behaviour over 10 networks on one fold of the test. The data points are drawn at roughly 20,000, 60,000, 100,000, 200,000, 400,000, 1,000,000 and 3,000,000 training cycles.

The *No Inhibition* line shows that after initial solid performance, the system's performance declines; it is not stable. The *Inhibition* line in figure 7 refers to the system from the first system in this section. While its performance remains high it does decline.

During early work on this task, systems were developed to categorise based on three inputs instead of four, and with 30 input neurons instead of 40. The *3 Input No Inhibition* line in figure 7 refers to this system and aside from this change of input number, the system is the same as the one from the beginning of section 5. This three input system has a much more rapid, obvious, and complete decline in categorisation by firing performance².

The *3 Input Inhibition* line in figure 7 refers to the system with inhibition added. This system is largely a duplicate of the first system from this section. Aside from the

² It should be noted that the three input task is different, and naturally performance will be lower as there is less input. Performance on the full 2-fold task at 20,000 cycles with 100 nets measuring by firing is 79.65%.

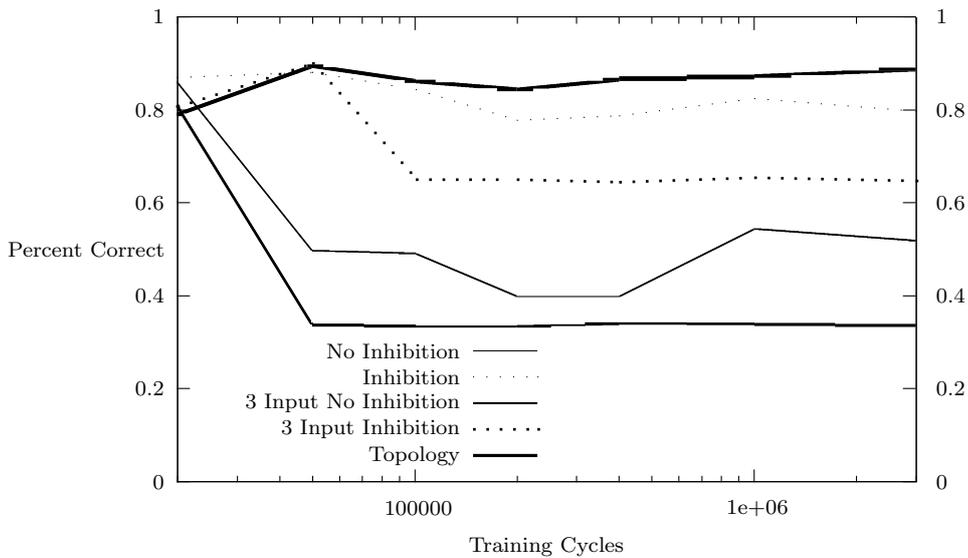


Fig. 7 Categorisation results over 10 nets on one fold at a range of training times on an exponential scale. This shows that both systems with three inputs are not stable. The system with no inhibition and four inputs is also not stable. The system with four inputs and inhibition is stable. The system with fewer internal synapses is also stable.

change of input, there is an increase in saturation base in the SOM subnet from 2 to 4. Performance declines after 50,000 cycles to around 67%, so this system is not stable.

The increase in the number of input neurons has made both the initial system and the inhibitory system appreciably more stable. In the case of no inhibition, the performance declines but is much higher with more input. Similarly, with inhibition, it is almost entirely stable with more inputs. As the overall system has more energy from the inputs, it makes sense that it is less likely to become self-absorbed.

Another way to modify the system is to increase inhibition. In the first simulation from this section, inhibition was activated when more than 100 neurons fired in either the SOM or Hidden subnets. This was modified so that inhibition began when more than 50 neurons fired in either. This minor change led to a stable system. This is the best 4 subnet system found for categorising by neural firing as measured on 100 nets for both folds. The full classification results run for 50,000 cycles are 89.45% (shown in table 2).

Another way to reduce the interactions between the neurons in the internal subnets is to reduce the number of connections. In the above four subnet simulations, neurons in the SOM and Hidden subnets had 10 connections to other neurons in the same subnet; each SOM neuron had 15 connections to the Hidden subnet, and Hidden neurons had 10 connections back to SOM neurons. New simulations are run with all of these reduced; there are 5 connections within both internal subnets, 10 connections from the SOM subnet to the hidden subnet, and 5 connections back. Otherwise, the simulation remains the same as that at the beginning of this section. The results over time are shown in figure 7.

The reduced number of internal synapses led to a stable system. This is shown in the *Topology* line in figure 7. This system is truly stable with performance improving as training duration increases. This system, like many of the systems, has a dip in performance around 100,000 cycles. It is possible that the system starts to become self-absorbed. As it starts to stabilise, it returns to being driven by the inputs and thus continues to improve.

It was also hoped that reducing the learning rate would solve the problem. The idea was that by slowing the change, the system would become less likely to become self-absorbed. Again, the first simulation from this section was modified. That system had a learning rate of 0.01 on the SOM and Hidden subnets. This was modified to 0.005 leading to a stable system performing around 82%.

A second mechanism to reduce activation in the interior networks is to reduce the saturation base constant, thus reducing the overall synaptic strength in the network (see equations 4 and 5). In the earlier simulations, the saturation base on the SOM subnet was 2; that is the target total synaptic weight for synapses coming into a SOM neuron was 4. Reducing this weight to 1 and 3 helped, but did not solve the problem.

It is important to note that the time to simulate was only tested to 3 million cycles. There is not a mathematical proof of stability, so it is plausible that at some point these systems may become unstable and remain unstable. It is also plausible that they are self-correcting and will remain stable.

This particular form of stability is just one rather simple form of homeostasis. The stable systems remain stable under a relatively constrained input regime. This is important, but is not stable in the way a full neural system must remain.

5.3 Learning Variants and Other Tasks

Two reasonable variants to consider are variants of the combination of post and pre-compensatory learning, and performance of this system on other tasks. In the above simulations, neurons in the Input subnet learned using post-compensatory learning, and in the other nets they learned using pre-compensatory learning. What happens when other combinations are used?

If learning is switched so that the first net learns using pre-compensatory learning and the other three use post-compensatory learning, the results are initially reasonably sound. After 20,000 cycles the results using the Pearson measurement is high, and the result via firing is around 67%. However, after 50,000 cycles the system has become self-absorbed; categorisation by both Pearson and firing has gone to chance.

If learning is switched so that all four nets use pre-compensatory learning, the results are again initially reasonable sound. Categorisation results are high using Pearson, and around 67% using firing. By 50,000 cycles the system has already become self-absorbed. While categorisation via Pearson remains high, categorisation by firing has returned to chance.

While the other 13 variants could be simulated, there are an infinite range of variants combining the two mechanisms. A network could learn with post-compensatory learning for a time, then switch to pre-compensatory learning. The two mechanisms could be combined so that the saturation base is the target for all synapses coming into and out of the neuron. The neuron could have one saturation base for incoming synapses and a second for outgoing synapses. Furthermore there could be range of

mechanisms for combining these rules. While these variants are interesting, exploring that space is beyond the scope of this paper.

Another question is how this particular topology and learning mechanism performs on another task. The four subnet topology with pre-compensatory learning on the Input subnet and post-compensatory learning on the other three subnets was employed on the yeast task. The Input subnet was enlarged to accept eight inputs (instead of four). The Output subnet was modified to accept ten categories (instead of three). A ten-fold test was used, duplicating the earlier two subnet simulation described above (and see (Huyck and Mitchell, 2013)). The system was trained for 20,000 cycles, and there was global inhibition on the internal subnets starting with 100 neurons firing. Each fold was run on ten networks.

Categorisation based on the firing mechanism was 30.45%. Categorisation based on Pearson measurements from the SOM subnet was 51.75%. The best prior result that the authors are aware of is 58.3% from a k-nearest neighbor algorithm with boosting (Athitsos and Sclaroff, 2004). Multilayer perceptrons, genetic algorithms, Kohonen maps, neural gas and other mechanisms were near this performance.

6 Discussion and Conclusion

Section 4.1 shows that the model of FLIF neurons learning by a combination of post and pre-compensatory learning is capable of reasonable categorisation of iris data. This is not surprising since a variant of this model has been used to categorise Congressional Representatives, and the same model has been used to categorise yeast and cars; all four tasks are from a commonly used benchmark.

This model uses neurons that spontaneously fire due to hypo-fatigue. Unlike the authors' earlier work, this provides a principled way of moving away from the sensory motor interface. The 3 and 4 subnet simulations have expanded on this. Moreover, the 3 and 4 subnet simulations show categorisation successfully based on neural firing.

The best categorisation results come from the Pearson measurements. Across all tested conditions (2, 3 and 4 subnets; inhibition and no inhibition; time variations; changed learning rate; reduced saturation base; and changed topology), the Pearson measurement on the SOM subnet has been above 90%. This shows that the basic mechanism, using post-compensatory learning on the Input subnet, and pre-compensatory learning on the SOM subnet, successfully stores the information needed to categorise. This is not lost even when the system becomes unstable. All that remains is for neurons to use that information in categorising.

All of the systems are self-organising in a general sense. In a specific sense, they map to Kohonen's maps by representing categories with sets of neurons instead of one unit.

While the simulations and the method are a reasonable machine learning algorithm, they are flawed as a neuro-psychological model. The instability of the system (see section 5) is one flaw. Inhibition alone, inhibition combined with fewer synapses, inhibition combined with reduced target synaptic strength, and to some extent inhibition with reduced learning rate and inhibition with reduced saturation base all lead to stable systems. So, to some extent this flaw has been removed. However, other flaws should be mentioned.

Three neuro-psychological flaws are simplifications to the complete model. They are: a small number of neurons, overly simple presentation, and turning learning off.

The first is that there are a small number of neurons and that there is only one type of neuron; the brain is composed of billions of neurons, and there are several dozen types of neurons. Secondly, the presentation for 400ms of simulated time, followed by 350ms with no presentation that is then repeated is a stretch psychologically. Moreover, neurons based on features are directly stimulated so there are no senses. Furthermore, neurons in the Output subnet orthogonally encode the category, whereas categories are typically included by overlapping sets of neurons in the brain (Freedman et al., 2001). Thirdly, after training, learning is turned off; learning remains on in the brain, though there may be a neuromodulatory mechanism that can be modelled.

These are all reasonable simplifications to begin to explore the problem. However, a psychological concept is held in the brain in a Cell Assembly (Hebb, 1949), which is a reverberating neural circuit. A Cell Assembly reverberates while the concept is in short term memory. The neurons in these simulations do not reverberate for more than 100 ms. while short term memory lasts for seconds.

In explicitly pointing out these flaws, the authors are noting that this is not a good neuro-psychological model. However, it is expected that the neural and learning models can be included in such a neuro-psychological model of memory.

The model described in this paper spreads firing into new neural subsystems. Compensatory learning leads to decorrelation. In the correct configuration, the systems reach homeostasis even while synaptic weights change. Note that in the above simulations subnets are not layers as in multi-layer perceptrons. Aside from the Input subnet, all subnets are interconnected forming one loosely connected recurrent net. These subnets may relate to brain areas or cortical lamina. The combination of spontaneous firing and compensatory learning spreads activation to unused areas. This is evident from the lack of particular neurons firing during testing using the Pearson's measurement. These neurons have yet to be added to the circuit.

This paper has begun an exploration of multiple layers; this may reflect laminar architecture or brain areas, and there are connections in both directions between layers and within layers. Categorising based on neural firing will enable a larger neural system to make use of the ability to categorise. It is hoped that in combination with a better understanding of learning dynamics, a complete memory system can be developed.

Acknowledgements: Thanks to Zhijun Yang and Dan Diaper for comments on this paper.

References

- Abbott, L. (1999). Lapique's introduction of the integrate-and-fire model neuron (1907). *Brain Research*, 50:303–304.
- Ackley, D., Hinton, G., and Sejnowski, T. (1985). A learning algorithm for Boltzmann machines. *Cognitive Science*, 9:147–169.
- Amit, D. (1989). *Modelling Brain Function: The world of attractor neural networks*. Cambridge University Press.
- Athitsos, V. and Sclaroff, S. (2004). Boosting nearest neighbor classifiers for multiclass recognition. Technical report, Boston University.
- Bache, K. and Lichman, M. (2013). UCI machine learning repository.
- Bi, G. and Poo, M. (1998). Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. *Journal of Neuroscience*, 18:24:10464–10472.

- Bienenstock, E., Cooper, L., and Munro, P. (1982). Theory for the development of neuron selectivity: orientation specificity and binocular interaction in the visual cortex. *Journal of Neuroscience*, 2:1:32–48.
- Brette, R., Rudolph, M., Carnevale, T., Hines, M., Beeman, D., Bower, J., Diesmann, M., Morrison, A., Goodman, P., Harris, F., Zirpe, M., Natschalager, T., Pecevski, D., Ermentrout, B., Djurfeldt, M., Lansner, A., Rochel, O., Vieville, T., Muller, E., Dafison, A., ElBoustani, S., and Destexhe, A. (2007). Simulation of networks of spiking neurons: A review of tools and strategies. *Journal of Computational Neuroscience*, 23:349–398.
- Bush, D., Philippides, A., Husbands, P., and O’Shea, M. (2010). Reconciling the stdp and bcm models of synaptic plasticity in a spiking recurrent neural network. *Neural Computation*, 22:2059–2085.
- Freedman, D., Riesenhuber, M., Poggio, T., and Miller, E. (2001). Categorical representation of visual stimuli in the primate prefrontal cortex. *Science*, 291:312–316.
- Fyfe, C. (2005). *Hebbian Learning and Negative Feedback Networks*. Springer.
- Hebb, D. (1949). *The Organization of Behavior*. John Wiley and Sons.
- Hodgkin, A. and Huxley, A. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *J. of Physiology*, 117:500–544.
- Hsu, D., Tan, A., Hsu, M., and Beggs, J. (2007). A simple spontaneously active hebbian learning model: homeostasis of activity and connectivity, and consequences for learning and epileptogenesis. *Physical Review E*, 76:041909.
- Huyck, C. (2007). Creating hierarchical categories using cell assemblies. *Connection Science*, 19:1:1–24.
- Huyck, C. and Mitchell, I. (2013). Compensatory hebbian learning for categorisation in simulated biological neural nets. *Biologically Inspired Cognitive Architectures*.
- Huyck, C. and Orengo, V. (2005). Information retrieval and categorisation using a cell assembly network. *Neural Computing and Applications*, 14:282–289.
- Huyck, C. and Parvizi, A. (2012). Parameter values and fatigue mechanisms for flif neurons. *Journal of Systemics, Cybernetics and Informatics*, 10:4:80–86.
- Izhikevich, E. (2004). Which model to use for cortical spiking neurons? *IEEE Transactions on Neural Networks*, 15:5:1063–1070.
- Izhikevich, E. and Desai, N. (2003). Relating stdp to bcm. *Neural Computation*, 15:1511–1523.
- Kohn, A. (2007). Visual adaptation: Physiology, mechanisms, and functional benefits. *Journal of Neurophysiology*, 97:3155–3164.
- Kohonen, T. (1997). *Self-Organizing Maps*. Springer.
- McCulloch, W. and Pitts, W. (1943). A logical calculus of ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5:115–133.
- Mitchell, I. and Huyck, C. (2013). Self organising maps with a point neuron model. In *17th International Conference on Cognitive and Neural Systems*.
- O’Reilly, R. (1996). *The Leabra Model of Neural Interactions and Learning in the Neocortex*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA.
- Wehrens, R. and Buydens, L. (2007). Self- and super-organizing maps in R: The Kohonen package. *Journal of Statistical Software*, 21(5).
- Wolpert, D. and Macready, W. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1:67–82.
- Wu, Q., Maguire, L., Glackin, B., and Belatreche, A. (2006). Learning under weight constraints in networks of temporal encoding spiking neurons. *Neurocomputing*,

69:1912-1922.