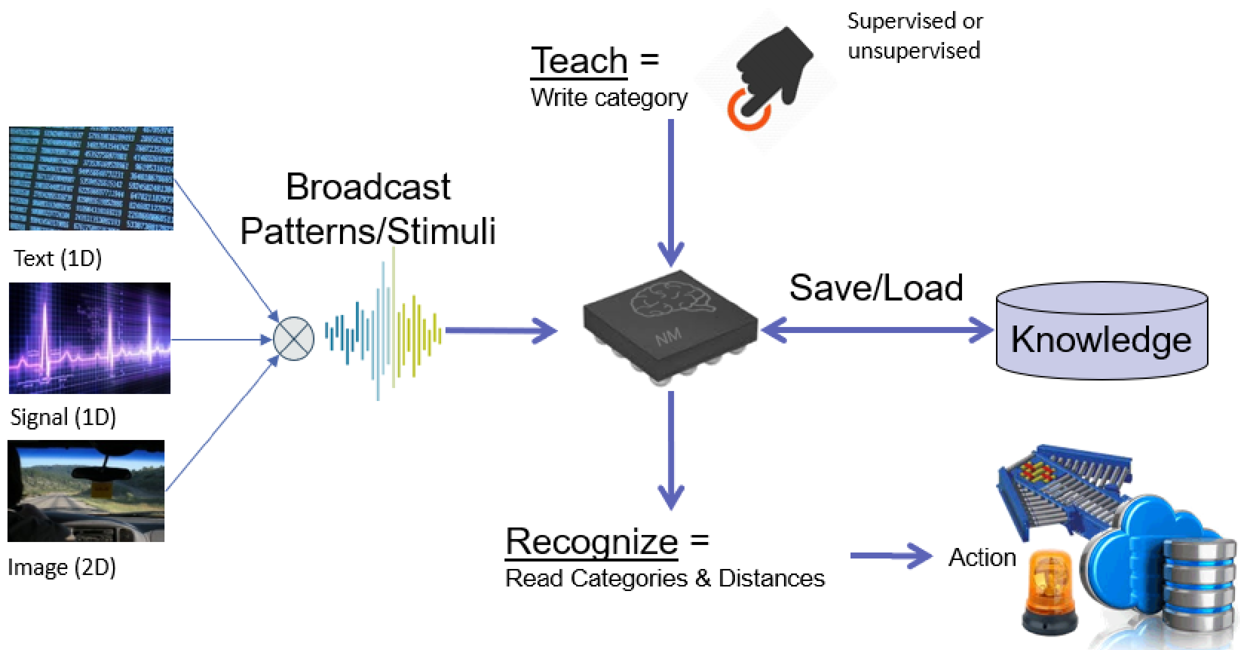
The NeuroMem neural network is a pattern recognition accelerator chip which is also trainable in real-time by learning examples. This technical note describes a simple script illustrating the behavior of the neurons during learning and recognition operations.



The above diagram illustrates how some input data coming from a variety of sources can be converted into a pattern vectors which can then be sent to the neurons for either learning or recognition. In the learning case, the neurons decide autonomously if the input pattern and its associated category (1) represent novelty and should be stored in the next available neuron, and (2) bring conflict to committed neurons which should be corrected. In the recognition case, the neurons decide autonomously which one has the closest match and queue their responses per decreasing level of confidence.

For detailed information about the neurons, their architecture and behavior, refer to the manuals NeuroMem Technology Reference Guide and NeuroMem Decision Space Mapping.

SIMPLE SCRIPT

In this simple example, the vectors are set to simple, short and fixed data set such as [11,11,11,11] and [15,15,15,15]. Their relationship is easy to understand and represent graphically.
In real applications, these vectors are extracted from sensor data or else and can be composed of up to 256 different values ranging between 0 and 256. For example, in the case of images, vectors can represent raw pixel

values, grey-level or color histograms, etc. In the case of audio and voice data, vectors can be cepstral coefficients, etc.

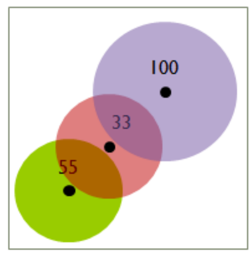
Step 1: Learn

Default vector length= 4 components

Learn vector1 [11,11,11,11] (1) with category 55
Learn vector2 [15,15,15,15] with category 33(2)
Leanr vector3 [20,20,20,20] with category 100

(1) The distance between two "flat" vectors is the difference between their constant value times their number of components. For example the L1 distance between [11,11,11,11] and [15,15,15,15] is equal to 4 * 4.
(2) The category of the second neuron is purposely set to a lesser value than the category of the first neuron to verify that if both neurons fire with a same distance, the category of the neuron on the 2nd chip is still the first the be read out

Fig1 is a representation of the decision space modeled by the 3 neurons if they were limited to 2 components instead of 4.



Committed neurons= 3

NeuronID=1	Components=11, 11, 11, 11,	AIF=16,	CAT=55
NeuronID=2	Components=15, 15, 15, 15,	AIF=16,	CAT=33
NeuronID=3	Components=20, 20, 20, 20,	AIF=20,	CAT=100

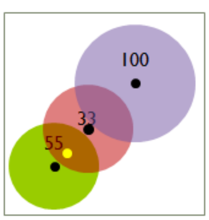
The influence fields of Neuron#0 and Neuron#1 overlap, as well as Neuron#1 and Neuron#2 overlap but differently since their distances from one another are different.

Step2: Recognition

The vectors submitted for recognition are selected purposely to illustrate cases of positive recognition with or without uncertainty, as well as cases of non recognition.

The program reads the responses of all the firing neurons, which is until the distance register returns a value 0xFFFF or 65553.

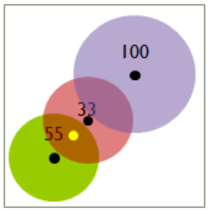
Case of uncertainty, closer to Neuron#1



Vector=12, 12, 12, 12
Neuron 1 and 2 fire. Vector is closer to Neuron1

Response#1	Distance= 4	Category= 55	NeuronID= 1
Response#2	Distance= 12	Category= 33	NeuronID= 2
Response#3	Distance= 65535	Category= 65535	NeuronID= 65535

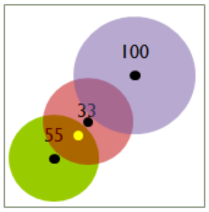
Case of uncertainty, equi-distant to Neuron#1 and Neuron#2



Vector=13, 13, 13, 13,
Neuron 1 and 2 fire. Vector is equi-distant to both neurons

Response#1	Distance= 8	Category= 33	NeuronID= 2
Response#2	Distance= 8	Category= 55	NeuronID= 1
Response#3	Distance= 65535	Category= 65535	NeuronID= 65535

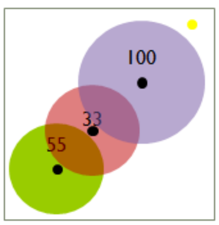
Case of uncertainty, closer to Neuron#2



Vector=14, 14, 14, 14,
Neuron 1 and 2 fire. Vector is closer to Neuron2

Response#1	Distance= 4	Category= 33	NeuronID= 2
Response#2	Distance= 12	Category= 55	NeuronID= 1
Response#3	Distance= 65535	Category= 65535	NeuronID= 65535

Case of unknown



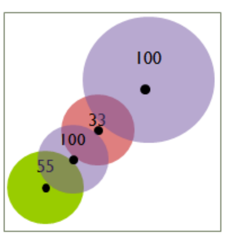
Vector=30, 30, 30, 30,
No neuron fire

Response#1	Distance= 65535	Category= 65535	NeuronID= 65535

Step3: Adaptive learning

Learning a new vector to illustrate the reduction of neurons'AIFs.

Learn vector[13,13,13,13] with category 100. This vector is equidistant to both Neuron1 and Neuron2. Learning it as a new category, will force Neuron1 and 2 to shrink from thei AIF=16 to an AIF=8 in order to make room for a new neuron which will hold the new model and its category.



Committed neurons= 4

NeuronID=1	Components=11, 11, 11, 11,	AIF=8,	CAT=55
NeuronID=2	Components=15, 15, 15, 15,	AIF=8	CAT=33
NeuronID=3	Components=20, 20, 20, 20,	AIF=20	CAT=100
NeuronID=4	Components=13, 13, 13, 13,	AIF=8,	CAT=100

Note that if the vector to learn was [12,12,12,12], Neuron1 would shrink to 4 and Neuron2 to 12.

SIMPLE SCRIPT EXTRA VERIFICATION

If you intend to use the Simple Script to verify proper access to the neurons but also proper communication, you might want to test the use of longer patterns.

Test 2 (V length=256)

By changing Vector Length to 256 which is its maximum possible value, the script should report the following information:

Neurons committed= 3

NCR= 1	COMP= 11, 11, 11, 11, ...11,	AIF= 1024	MINIF= 2	CAT= 33
NCR= 1	COMP= 15, 15, 15, 15, ...15,	AIF= 1024	MINIF= 2	CAT= 55
NCR= 1	COMP= 20, 20, 20, 20, ...20,	AIF= 1280	MINIF= 2	CAT= 100

Recognizing a vector non equi-distant to the neurons: (comp 12)

Dist= 256	Cat= 33	Nid= 1
Dist= 768	Cat= 55	Nid= 2

Recognizing a vector equi-distant to the neurons: (comp 13)

Dist= 512	Cat= 33	Nid= 1
Dist= 512	Cat= 55	Nid= 2

Recognizing a vector equi-distant to the neurons: (comp 14)

Dist= 256	Cat= 55	Nid= 2
Dist= 768	Cat= 33	Nid= 1

Test 3 (V length=N)

By changing Vector Length to N with N<256, the script should report the following information:

Neurons committed= 3

NCR= 1	COMP= 11, 11, 11, 11, ...11,	AIF= 4*N	MINIF= 2	CAT= 33
NCR= 1	COMP= 15, 15, 15, 15, ...15,	AIF= 4*N	MINIF= 2	CAT= 55
NCR= 1	COMP= 20, 20, 20, 20, ...20,	AIF= 4*N	MINIF= 2	CAT= 100

Recognizing a vector non equi-distant to the neurons: (comp 12)

Dist= N	Cat= 33	Nid= 1
Dist= 3*N	Cat= 55	Nid= 2

Recognizing a vector equi-distant to the neurons: (comp 13)

Dist= 2*N	Cat= 33	Nid= 1
Dist= 2*N	Cat= 55	Nid= 2

Recognizing a vector equi-distant to the neurons: (comp 14)

Dist= N	Cat= 55	Nid= 2
Dist= 3*N	Cat= 33	Nid= 1