# Self Organized Learning Applied To Global Positioning System (GPS) Data

AHMAD R. NSOUR, M.S MOHAMED A. ZOHDY, PhD

Department of Electrical & Computer Engineering
Oakland University
Rochester, MI, 48309
USA
Phone: (248) 370-2234
Fax: (248) 370-4633

*Abstract*: – In this paper, we applied self organized unsupervised neural learning to process global navigation systems with multiple input and multiple output signals, we considered versions of topological one-dimensional feature map layer. Several learning methods were examined in order to determine their effect on minimization of the clustering distance errors. New visualization of the multivariable network performance is introduced and applied to sample global positioning system (GPS) data. Although this study is significant, extensions to two-dimensional and higher dimensional layers feature map are currently being developed.

*Keywords:* - Self Organizing Feature Map, Global Positioning System, Learning rates, Cluster Distance Error, One dimensional Network.

## 1. Introduction

Neurons form the basic building blocks of a nervous system, and they successfully communicate information and perform rather complex sequential pattern processing and recognition, characterized by intensive connections, inherent parallelism, self adaptation, and organization.

The growing scientific field of artificial neural networks uses mathematical modeling and computer simulation to achieve robust learning and pattern information processing analogous to the nervous system by interconnecting simple yet nonlinear computational elements.

Several potential applications areas have been recently considered, ranging from control system to speech synthesis and image processing. The preliminary results are promising and have opened up new research alternatives to conventional computer serial programming paradigms and even artificial intelligence systems [12].

Artificial neural networks are broadly classified by the type of their connective structure, input-output transfer function, and learning algorithm, which describes how the connective weights are adjusted to adapt the network's dynamic performance in order to achieve application goals [4].

Unsupervised learning received much attention because it can extract information without prior constraints [13].

The so called self-transforming and feature mapping in unsupervised neural networks is typically associated with the special adaptive behavior of connective weights in a training phase, intended to selectively extract salient input space features under either a deterministic or a stochastic environment. A self-informing learning style capitalizes on the competition among output neurons and their surrounding neighborhoods in order to code input distribution and consistently improves with training [5].

The network weights asymptotically approach exemplars of input clusters, which is reminiscent of simulated annealing and similar approaches in the related field of global optimization. [3], [11].

The network's output neurons are usually conveniently arranged in either one-dimensional or two-dimensional layers. Full connectivity to the inputs is tacitly assumed. A specific output neuron can reach "on state" and thus updates its weights and the weights of all its surrounding neighborhoods. Normalization of all weights, in addition to adaptively controlling the size of neighborhoods,

usually improves the performance by equalizing the relative changes in respective weight connections.
Self transforming feature maps have been used for multidimensional data reduction. Its advantages is in its parallelism and distribution and robust to faults [2], [10].

## 2. Problem Formulation

### 2.1 Feature Map Neural Net Learning

The neuron net learning can be represented by a set of discrete nonlinear mathematical equations. [1], [6], [12].
The strength of inter- connective weights are expressed in a $\nu \times \mu$ weight matrix $W(\kappa)$, $\nu$ is the total number of output layer neurons, $\mu$ is the total number of input neurons to be learned.
$Y(\kappa) \in R^n$ is the neuron outputs at the $\kappa$th discrete iteration step. $X(\kappa) \in R^m$ and $U(\kappa) \in R^n$ are the input stimuli vector and the net weighted sum.
Finally, consider a nonlinear activation function designated by $\phi: R^n \rightarrow R^n$
Then the output neuron activity is modeled by:
(1) $Y(\kappa+1) = \phi[V(\kappa)]$
(2) $V(\kappa) = U(\kappa) + \beta C(\kappa) Y(\kappa)$
(3) $U(\kappa) = W(\kappa) X(\kappa)$
And β reflects relaxation that increase or decrease the effect of any lateral feedback connections

#### 2.1.1 Neural Learning algorithm

Self transforming feature mapping is a transformation from the input signal space to a topologically ordered but reduced dimensional output neuron activity pattern $X \in R^{mn}$ is the input vector;
$W_i(\kappa) \in R^m$ is the associated connection weight.
The dot vector product $W_i X$ is generally a scalar measure of the geometric projection of the input vector on a subspace spanned by the weights.

(4) $(W_i, X) = \| W_i - X \|_2 = \sum_{j=1}^{m} (W_{ij} - X_j)^2$

$i(x)$ = number of neuron widths $(W_i, X)$

(5) $W_i(\kappa+1) = W_i(\kappa) + \alpha(\kappa)[X - W_i(\kappa)]$

Where $W_i(\kappa) \in R^m$ is the weight vector at the $\kappa$th iteration, $X \in R^m$ is the input vector, and $\alpha(\kappa)$ is a scalar learning rate. [7], [8].

Several possible learning rate expressions are considered as follows:

(6) $\alpha(\kappa) = \alpha_c$ **(constant)**

(7) $\alpha(\kappa) = \alpha_c \dfrac{1}{k}$ **(reciprocal)**

(8) $\alpha(\kappa) = \alpha_\alpha \exp(\dfrac{-k}{J_\alpha})$ **(exponential)**

(9) $\alpha(\kappa) = \alpha_\alpha \exp(\dfrac{-k^2}{J_\alpha})$ **(double expenon.)**

(10) $\alpha(\kappa) = \alpha_\alpha \dfrac{1}{\ln(1+k)}$ **( logarithmic)**

Where $J_\alpha$ the moment of inertia and k is is the number of iterations.

## 3. Problem Solution

### 3.1 An Application to GPS Data

Global positioning system (GPS) is a modern satellite navigation and communication system, mostly used for determining precise relative spatial distribution and also providing a highly accurate time reference almost anywhere on Earth or Earth orbit. It uses intermediate circular orbit (ICO) satellite constellations of at least 24 satellites.
The Global positioning system was designed by and is controlled by the United States Department of Defense but can be used by anyone, free of charge. A GPS system is usually divided into three segments: space, control and user. The space segment comprises the GPS satellite constellation. The control segment comprises computerized ground stations around the world that are responsible for monitoring the flight paths of the GPS satellites, synchronizing the satellites' onboard atomic clocks, and uploading data for transmission by the satellites. The user segment consists of GPS receivers used for both military and civilian applications. A GPS receiver aggregate time signal transmissions from multiple satellites and calculates its position by mathmatical trilateration.
The data applied in this paper form a sample of global positioning system (GPS) and is shown in Table 1.it contains data corresponding to single “snapshot” of GPS measurement; they represent pseudo range, defined as the distance between the GPS satellite at some transmit time and the receiver at some receiver time in (meters), number of carrier cycles, the fraction carrier

cycles (1/2048 cycles) and signal to noise ratio (dB-Hz) [9].

### 3.2 Simulation Results and Discussion

Three output decision neurons and four input sensory neurons pseudo range, carrier, fraction of carrier cycles and signal to noise ratio were considered in this paper. A weight matrix was initialized with small uniform random numbers. The Euclidean distance was measured between the weights and each of the inputs, representing the error distance, shown in Figure (1) through Figure (5) for different forms and values of the learning rate with $J_\alpha = 3.1$.

For simulating one dimensional neural network based on the learning algorithm discussed in equations (4) and (5). A new method for improving the convergence of the Learning algorithm for one dimensional unsupervised network is used. As mentioned previously we started with three competitive output neurons. Each output neuron was connected into four input neurons, each resembling a different weight value. The output neurons weights were mapped according to their respective inputs.

| **Pseudo range** | **Carrier cycles** | **fraction of carrier cycles** | **signal to noise ratio** |
|---|---|---|---|
| 24284349.612 | 12032286 | 125 | 45 |
| 24286332.459 | 12616964 | 1438 | 45 |
| 26745896.433 | 10501718 | 613 | 33 |
| 23326709.812 | 10891224 | 49 | 45 |
| 24187299.188 | 12270594 | 799 | 43 |
| 23856016.416 | 10464010 | 1774 | 44 |
| 24290150.030 | 12062767 | 187 | 45 |
| 24292443.277 | 12649077 | 549 | 45 |
| 26750829.611 | 10527646 | 936 | 34 |
| 23331920.306 | 10918604 | 2009 | 45 |
| 24193210.579 | 12301658 | 435 | 43 |
| 23861078.866 | 10490617 | 469 | 44 |
| 24295949.201 | 12093244 | 396 | 45 |
| 24298553.027 | 12681186 | 201 | 45 |
| 26755764.616 | 10553570 | 634 | 34 |
| 23337130.473 | 10945982 | 688 | 45 |

**Table 1 - GPS Data**

The output neurons are shown in Figure (6), each separate curve represents an output neuron performance which helps visualize the results of simulations. The curves are drawn to chart the change of the weights in relation to their inputs. After training the neurons the algorithm converged and the lines were almost harmonious as shown in Figure (7). Similar procedure was applied to learning with $\alpha_c = 0.01$, we notice the better convergence pattern of the weights as shown in Figure (8). A new test with small uniform random weights and learning rate as shown in Figure (9).and the weights after being trained as shown in Figure (10) when convergence was slow. So the $\alpha_\alpha$ was changed to 0.01 and that improved the convergence as shown in Figure (11).

## 4. Conclusion

In this paper, we have shown that a neural network feature map with self organized learning methods have the capability to cluster multiple GPS inputs and outputs and introduced output layer learning performance visualization. The generalized clustering distance errors were found for various learning rates that depended on parameters including the number of iterations, neighborhood size and the learning coefficients

*References:*

[1] Bishop C., *Neural Networks for Pattern Recognition*. Oxford University Press, 1st edition, 1996.

[2] De Wolf, T. and Holvoet, T. Emergence Versus Self-Organisation: Different Concepts but Promising When Combined. *In Engineering Self Organising Systems: Methodologies and Applications, Lecture Notes in Computer Science,* Vol *.3464*, 2005, pp.115

[3] Knudsen, T. & DuLac, S. and Easterly S. Computational maps in the brain. *Annu. Rev Neurosci*. Vol.*10*, 1987, pp.41-65.

[4] Kohonen, T., *Self-Organization and Associative Memory*, New York Springer-Verlag, 3rd edition, 1989.

[5] Kohonen, T., *Self- Organizing Maps*. New York: Springer- Verlag, 3rd edition, 2001

[6] Kaski, S. Data exploration using self Organizing maps. *Acta- Polytechnica Scandinavia Mathematics Computing and*

*management in Engineering series the Finnish Academy of Technology*, 1997, 57pp.

[7] Lapedes, A. and Farber, R. A self optimizing non symmetrical neural net for content addressable memory and pattern recognition. *Proceedings of the Eighth Annual Conference of the Cognitive Science Society,Physica* Vol.*22D*, 1986, pp.247-259.

[8]Luttrell, S., Organization: a derivation from first principles of a class of learning algorithm.*Proc89.2:1989*, 1989, pp495- 498.

[9]Misra P.&Enge, P., *Global Positioning System, Signals, Measurements, and Performance*, Ganga-Jamuna Press, 1st edition, 2001.

[10] Prehofer, C. & Bettstetter, C. Organization in Communication Networks: Principle and Design Paradigms, *IEEE Communications Magazine*, Vol.*43 (7),* 2005.

[11] Von der Malsburg, C. Outline of a theory for ontogenesis iso-orientation domains in visual cortex. *Springer Berlin / Heidelberg, Biological, cybern*, Vol.*45 (1)*, 1982, pp.49- 56.

[12] Zohdy, H. & Zohdy, M. Self-organizing Feature Maps. *The Wiley Encyclopedia for Electrical Engineering,* 1997, pp.*767* -772.

[13] Zohdy, M. and Bucak I., Reinforcement Learning Control of Nonlinear Multi-Link, System. *Engineering Application of Artificial Intelligence*, 2001, pp.563-573.

## 5. List of Figures

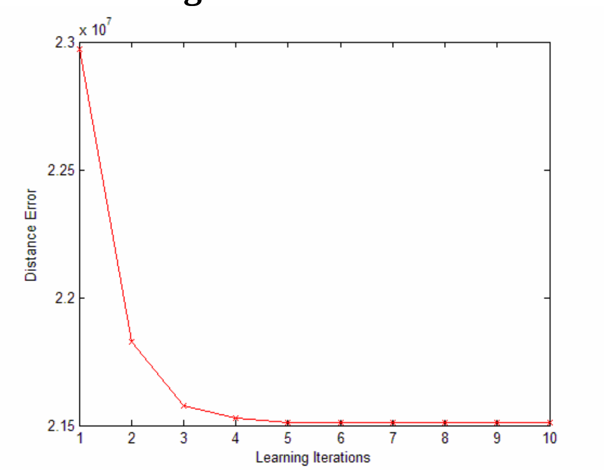


Fig.1 - learning error of $\alpha(\kappa) = \alpha_{\alpha} \; exp \; (\frac{-k^2}{J_{\alpha}})$

Given $\alpha_{\alpha}$ =0.01 , $k$=10

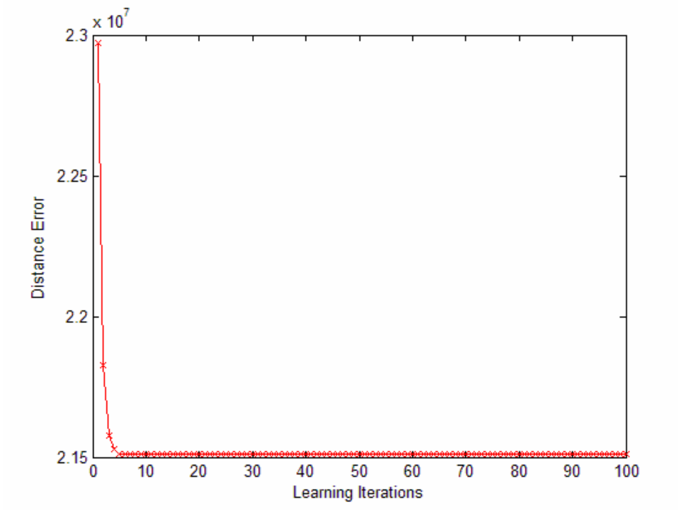


Fig.2 - learning error *of* $\alpha(\kappa) = \alpha_{\alpha} \; exp \; (\frac{-k^2}{J_{\alpha}})$

Given $\alpha_{\alpha}$ =0.01 , $k$=100 (the number of iterations)

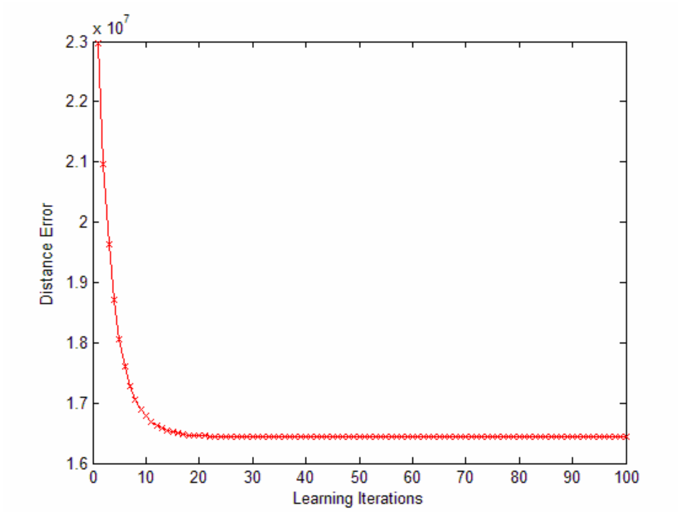


Fig.3 –learning error of $\alpha(\kappa) = \alpha_{\alpha} \; exp \; (\frac{-k}{J_{\alpha}})$,

given $\alpha_{\alpha}$ =0.01 , $k$=100 (the number of iterations)

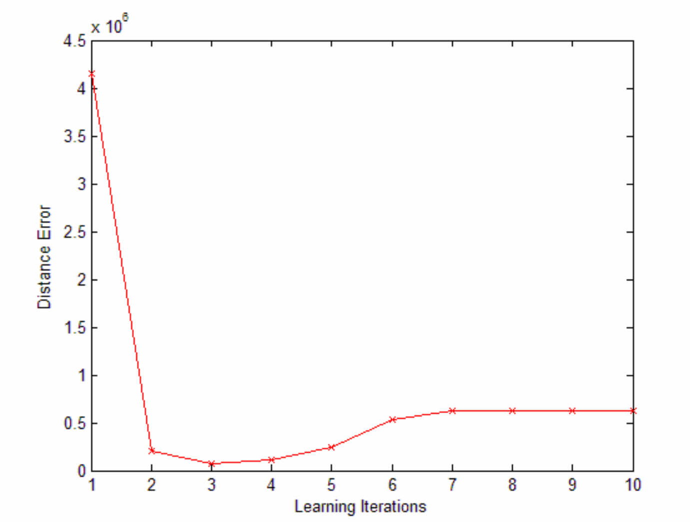


Fig.4 –learning error *of* $\alpha(\kappa) = \alpha_\alpha \frac{1}{\ln(1+k)}$, given $\alpha_\alpha$ =0.1 $k$=10 (the number of iterations)

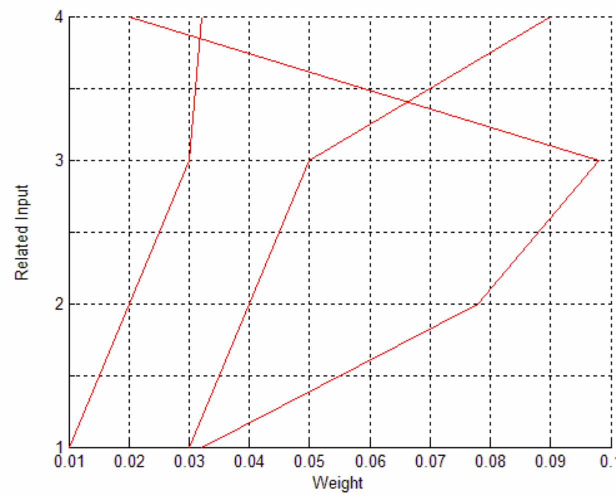


Fig.6 - Initial weights before training.

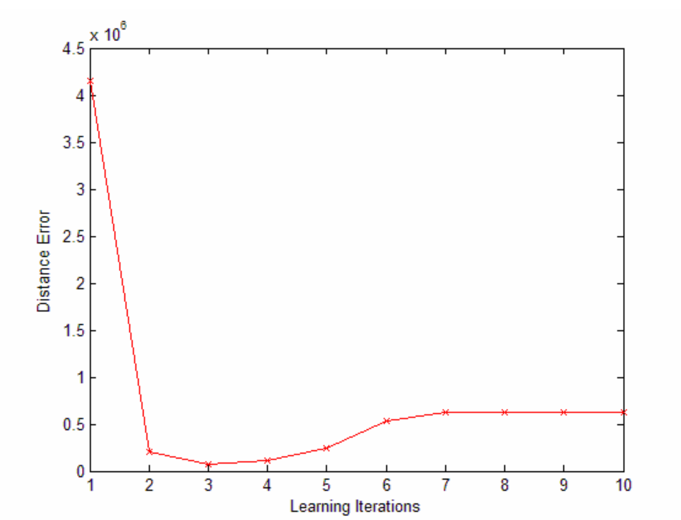


Fig.5 – learning error for $\alpha(\kappa) = \alpha_\alpha$, where $\alpha_\alpha$ =0.1

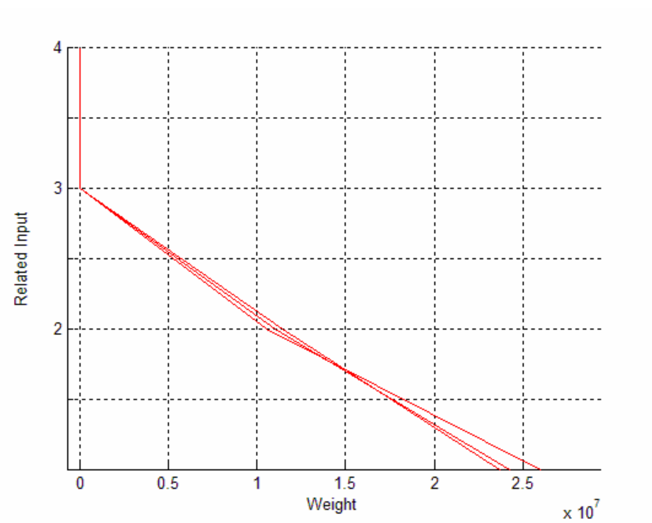


Fig.7 - weights after training, for learning rate of $\alpha(\kappa) = \alpha_c \, exp\,(\frac{-k^2}{J_\alpha})$ **,** where $\alpha_c$**=0.9,** k= 100

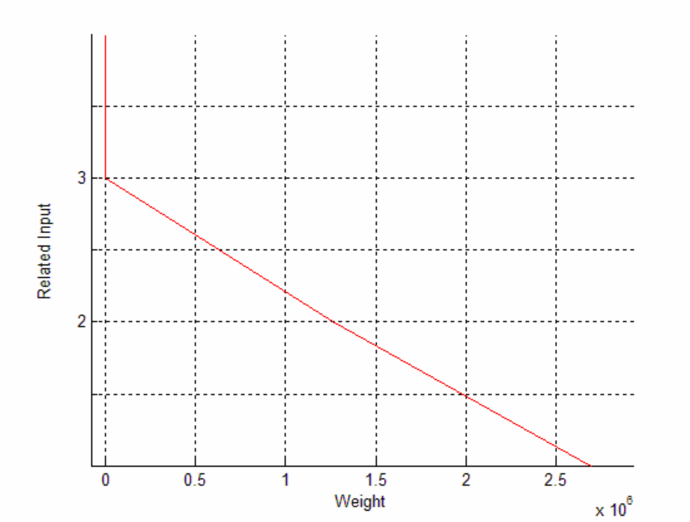


Fig.8 - weights after training, for learning rate of $\alpha(\kappa) = \alpha_c \, exp\,(\frac{-k^2}{J_\alpha})$, where $\alpha_c$=0.01 and k=100

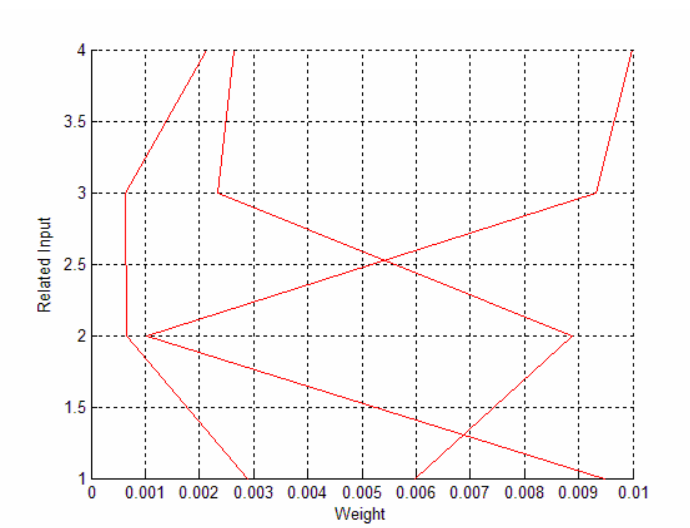


Fig.9 - Initial weights before training.

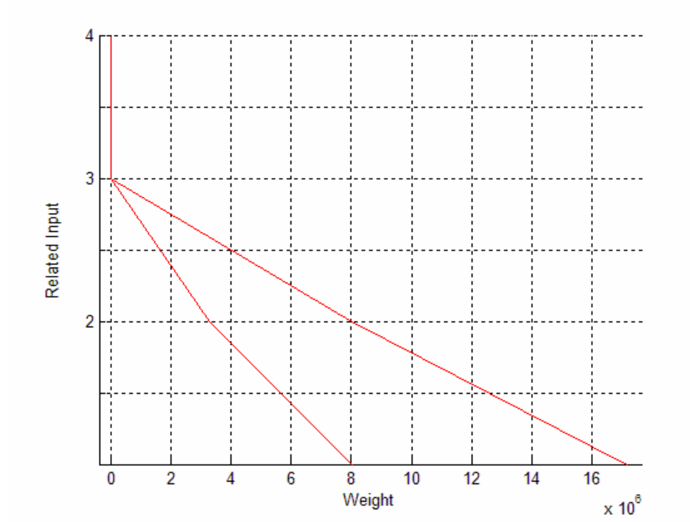


Fig.10 - weights after training, for learning rate *of* $\alpha(\kappa) = \alpha_\alpha \; exp\,(\frac{-k}{J_\alpha})$ where $\alpha_c$=0.1.

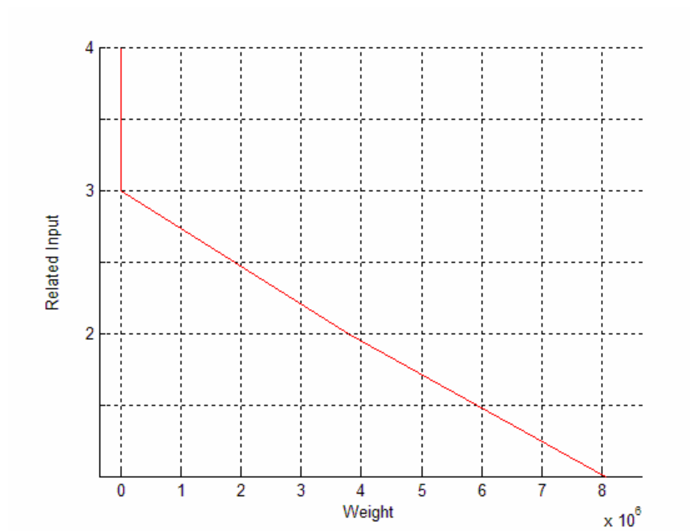


Fig.11 - weights after training, for learning rate of $\alpha(\kappa) = \alpha_\alpha \; \exp\,(\frac{-k}{J_\alpha})$ where $\alpha_c$=0.01.