Available online at www.joac.info

ISSN: 2278-1862

Journal of Applicable Chemistry

2014, 3 (2): 834-884

(International Peer Reviewed Journal)

State-of-Art-Review (SAR-Invited)

Mathematical Neural Network (MaNN) Models
Part II: Self Organizing Maps (SOMs) in chemical sciences

K RamaKrishna[1], V. Anantha Ramam[2], R. Sambasiva Rao[2*]

1. Department of Chemistry, Gitam Institute of Science, Gitam University, Visakhapatnam, 530 017, **INDIA**
2. School of Chemistry, Andhra University, Visakhapatnam 530003, **INDIA**

Email: karipeddirk@gmail.com, rsr.chem@gmail.com

Accepted on 12th March 2014

(Dedicated to Dr. K V Ramana, former professor of bioinorganic chemistry, Andhra University, on completion of seventy five years of life on the lap of Mother Nature)

ABSTRACT

Vector quantization (VQ) determines representative set of vectors, each of them called a quantizer/code vector/template/centroid for unsupervised multi-dimensional data sets (i.e. without teaching signal or response). The limitation is that it does not have the concept of neighborhood and topology. The geometric proximity of pre-synaptic biological neurons in the brain was the source of inspiration for Kohonen-self-organizing-map (Kohonen-SOM) with a grid of 1D-, 2D- or 3D- frame of a vector-, matrix- and tensor- of equi-distant neurons which are not connected to each other. The shapes of the neighborhood structures which are in wide use are diamond, square and hexagonal. Winner takes all (WTA) and winner takes most (WTM) mechanisms are used to determine winning neurons or quantizers. It belongs to a class of unsupervised-NN model for numeric data employing competitive learning with neighborhood lateral interaction. The end result is arriving at a topological structure hidden in the data set. In the visual display of Kohonen map, clusters of different classes are clearly distinguished and two patterns close in input space are nearer in output space. SOM is equivalent as a special case to the popular multi-dimensional-scaling (MDS) and regularized mixture models. U-, U-, P-, U*F procedures are used in the display of average distances of winning neurons from neighbors. ViSOM, generative topographic mapping, consensus tree etc., are recent visualization methods. The noteworthy advances in architectures are evident in tree-, evolving-tree, self-evolving-tree-, hierarchical-, hybrid-hierarchical-, grey-, spherical-, geo-, parallel-, kernel-, granular-, greedy-granular-, median- and self-organizing-relationship- SOMs. The scope of chemical science in this century is broad encompassing not only bio-, environmental-, geo-, marine-, drug-/ material-, clinical-, dietary- pharmaceutical- tasks but also atomic to macro-molecular systems at very-high-/very-low temperatures/pressures/sizes. The future thrust area of fundamental prime research is around chemical transformations to the present day universe since the formation of hydrogen, helium and lighter chemical-elements with the knowledge of particle physics mind blowing research. The references are sorted journal wise for ease of down loading from print/ online-or-offline electronic resources.*

Keywords: Neural network models, Self-organising-map, Unsupervised multi dimensional data, 2D-/3D-visualization, Chemical-engineering-medical applications, Hybrid SOM, Hierarchical-growing-SOM..

Contents.SOM

INTRODUCTION

Nature comprises of life, matter, energy and hidden nature-of-nature. The origin of universe dates back to 13.7 billion years. The ants have been there on earth since hundred million years. Human being on earth is 200,000 years young. Science, even if counted from Aristotle started two thousand years ago. The experimental and theoretical foundations started just before two to three centuries. Biologists precisely describe life with three primary characteristics viz. digestion, locomotion and reproduction. Modern chemistry, quantum physics, theoretical biology, brain chemistry etc. did not even complete hundred years of practice. Information science, hardware and software systems, artificial intelligence have their origin around nineteen fifties. The man (Homo-sapeon) amazed at nature, then appreciated, admired and even worshipped. Slowly he grew to understand and mimic it. The efforts are directed towards even to control surrounding nature for what he thought to be beneficial to the then existing/future mankind or animal kingdom. A tiny attempt is in the direction of artificial life to simulate/emulate part of nature and with a far off goal of creating life in Toto to achieve eradication of dreaded diseases, enhancing the human life span to 150 years, clean environment maintaining eco-balance and diversity at the same time. Around 1890s, William James [1], a renowned psychologist mentioned in his two volumes set entitled 'Principles of Psychology' that discrimination and association are two indispensable components for orderly progress of scientific psychology. The analogy is that one of the two legs of a walking man is always behind the other criticized as a pessimists' dogma, while one leg is ahead of the other is an optimist's hope. The fact is both are true, with the exceptional rarity being that both are at the same dot spot. Apart from biological neural nets, central themes of psychiatry, the role of brain in voluntary and involuntary functions of a living species are amazing. The core activity in mathematical neural network (MaNN) research is around improving the function of (artificial) mathematical neuron (or processing unit) and architecture. The latter comprises of direction of connection between neurons, transfer functions (TFs) and accumulation operators. The training algorithms and basis/object functions that are available in mathematical sciences are borrowed here. In a few instances they are modified to suit the context.

Biological neuron: Biological neuron is the basic unit of brain and nervous system. Neuroscience probes into functioning of sense organs, memory/thought/consciousness, voluntary and an involuntary activity as a result of the electrical spikes generated and transmitted in neurons. The cumulative effect of confluence of input signals and their synaptic strength, activation function to fire output for a bundle (10,000) of neurons produce miraculous outcomes.

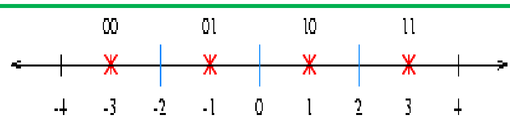
Neuron model or artificial neuron: McCulloch and Pitts (MP) proposed [2] a simple model of neuron in 1949 with fixed weights between neurons and binary inputs. It explained Boolean 'not' gate and MP-NN mimicked 'AND', 'OR' binary truth table. The enhanced power of artificial neuron (now popular as processing unit in computational intelligence) to transform input into output is through a variety of transfer functions (TFs) viz. sigmoid, atanh, radial basis function (RBF), wavelet, ridgelet, support vectors (SVs), complex/Geometric/algebraic equations, fuzzy formula. Some of the neurons derive their name from TFs employed as activation functions. Different confluence operators gave birth to sigma, pi, mu and fuzzy neurons. McCulloch and Pitts, Rossenblaut, Hodgkin-Huxlay neurons are named after the scientists. All these neurons come under the category of static type. The feedback with and without time delay and distribution brought revolution in NN research to model dynamic and time series data. IIR, FIR, NARMAX, recurrent, higher order tensors are like encapsulated modules bringing down the physical size of neural networks. Quantum neuron is a hope of the future quantum computer. Tensor notation for connection and pictorial representation is used to introduce artificial neurons, the heart of NNs. An integrated circuit like neuron from software and hardware perspective is awaited for computational intelligence/bio-mimicking devices with the ultimate target of a human brain followed by super/hyper gadget.

Biological neural network (NN.Biol): The evolution of architecture of biological neural networks underwent phenomenal changes from species to species over long time/generations. In human brain itself, there are more than 10^{11} neurons and as many as 10^4 connections exist with different synoptic strengths for each neuron. In the artificial NN front, the architecture did not even cross a primary stage from this perspective. Yet, the astounding results excelling in accuracy for real time dynamic multiple processes over the two century old mathematical models are the impetus for open minded research. The fixed architecture with input and output layers was proposed by McCulloch and Pitts [2]. The fixed weight stigma was surmounted by Rosenblatt by training weights with input patterns. The failure of simple architecture to explain non-linearly separable tasks (XOR) was a death blow to progress of NNs for over 25 years. The hidden layer with non-linear TF was a breakthrough and successfully modeled XOR. The back-propagation (BP) algorithm in training weights connecting neurons (Ws) confirmed a berth for NN research. During the dark period of former NN paradigm, independent schools of thought due to Grossberg [3], Hopfield [4] and Anderson [5] worked with alternate architectures, firing criteria and weight up-gradation schemes. The progress in feed forward layered fixed architectures was in invoking different TFs, number of layers and accumulation operators. Two hidden layers in MLP could model difficult non-linear transformation. In addition to layer wise connections, backward connections are involved in recurrent NN architecture. Elman and Jordan NNs belong to partial recurrence connections. Hopfield NN has acyclic and cyclic architectures. Fully recurrent with self feedback are the order of day, of course, with difficulties at training phase. Recirculation architectures are a special type in this category.

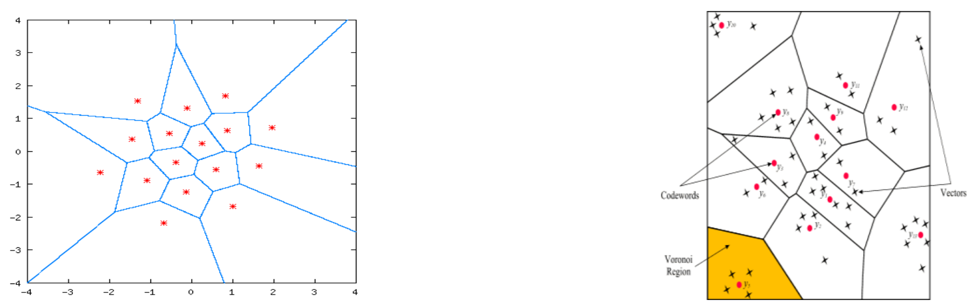
Mathematical-/Artificial- neural network (NN.Math, NN.Artfis): For clustering/classification tasks with unsupervised data containing only explanatory variables (X) without response (y), Grossberg proposed ART type architecture with feedback from category to feature layer. Kohonen architecture has a grid of 2D- or 3D- set of neurons connected from input layer using WTA heuristic. Neocognitron, LVQ and ARTMAP are supervised NNs corresponding to the unsupervised counterparts SOM and ART. The progress in ARTMAP and SOM type NNs is both extensive and intensive during the last two decades. Time delay NN architecture includes delay period, distribution and transmission of the delayed output of the hidden layer. Growing architectures both in layered and unlayered type received attention to arrive at optimum architecture depending upon the nature of task. Combination of two NNs or more than two gave birth to sequential and hierarchical structures.

In yester years, the change in architecture is mostly manual as per the choice of user. The software TRAJAN has a provision to change number of layers/neurons in them/TFs in feed forward (FF-) NNs based on built-in heuristics in its intelligent problem solver (IPS) mode. Professional II, in one of its forms, completely automates the architecture and training process. Predict from Neural ware is a healthy combination of NNs and statistics in right proportion for twenty first century tasks just like a multi-drug therapy and intervention procedures for multiple organ treatment. MATLAB in its tool box is a white box approach with open source code. In recent times, genetic algorithm (GA) and evolutionary programming (EP) are used in automating architecture as well as training of Ws.

1.1 Vector Quantization: In 1980s, an unsupervised vector quantization method to represent (m-D) real data by a finite number of vectors called quantizes (Fig. 1) was proposed which is also referred as hard-VQ (or hard-c-means) in fuzzy literature. It divides unsupervised data patterns into true (natural) groups. VQ is applicable when no teaching signal (y) is available. The objective is to determine a small but representative set of vectors (coordinates of centers). It is applicable for conceptualization, creating new categories/concepts, compression, dimension reduction and clustering from examples (of images/speech or signals from instruments [6, 7].The synonyms of quantizer are centroid, code vector or template. The number of quantizers is always less than the number of samples. The quantizers (vectors) are determined by minimizing the difference between expected Euclidean distance between all data vectors and their corresponding quantizers, or minimum loss of information in the model. This method projects R^d data space into a subspace exploiting the internal structure of input space. However, the number of quantizers and criteria for quality of clustering are user chosen quantities. The noise in data which perturbs VQ method, is taken care of in channel optimized VQ.



x < (-n) is approximated as (–n-1) Eg. Values less than -2 are represented by -3



Red star (*): code vector representing points falling in region covered

full blue lines (___); codebook is set of all code vectors

Fig. 1 Vector quantization (VQ)

Recent advances.VQ: VQ is similar to clustering methods like k-means or LBG algorithm. Recently, fuzzy-VQ, annealed-VQ and information-VQ are proposed. The objective is to arrive at minimum quantization error [8, 9], but not to achieve good generalization error. The optimization criterion for annealed VQ is equal to maximum likelihood employed for mixture of Gaussians. In the information theoretical approach of VQ, neighborhood learning is not a matter of concern.

VQ
- No neighborhood and no topology
  - SOM
- Local minima
  - Neural Gas-NN

Hybrid VQ-SVM: A hybrid VQ-SVM frame work was proposed to incorporate prior domain knowledge in NN. It is a hierarchical semi-parametric machine learning method applied to imbalanced datasets.

Elastic nets: Here topology is introduced by adding penalty term to annealed VQ error. This method is less suitable for visualization of high dimensional space [10].

2 Self Organizing Map (SOM): In feed forward NNs (MLP, RBF, Fuzzy-NN), the input is transformed into output by supervised learning. Willshaw and Van der Malsburg proposed in 1976 a self organizing unsupervised model based on geometric proximity of pre-synaptic neurons, which are coded as correlation in electrical activity. In this NN, threshold learning is used. The limitation is that dimension of output is equal to input resulting large number of connections.

Kohonen SOM: Kohonen [11-13] proposed and improvised [14-16] self-organizing map (SOM.Kohonen). The non-parametric unsupervised NN-SOM is a non-statistical data driven exploratory clustering method. The modifications, advances of SOM are mind blowing and applications over the years are extensive [17-263]. Even a bibliographic citation is beyond the scope of this review. In its naive form, it is also called Crisp-SOM to distinguish it from fuzzy-SOM reported by Kaburlasos [193]. It finds out rapidly the features and trends of clusters. To start with, the objective of SOM was to model the human brain, but, till to date, it is not successful in its entirety. However, it is one of the best data mining tools and excelled many statistical and mathematical procedures. The primary target is to approximate high dimensional data to a low dimensional one. Crisp-SOM computes n-D reference vectors using convex combination in n-dimensional Euclidian space (R^n). Thus it captures locally the first order statistics in the training data. SOMs function better than classical clustering and principal axes (PCA, correspondence) techniques.

Kohonen SOM is a variant of VQ with additional lateral interactions i.e. neighborhood effect. Here, topological property is the main perspective and generalized distortion is minimized. SOM organizes itself to learn on its own and categorizes inputs into groups of similar patterns. SOM itself is the end product in unsupervised classification task and is used for prediction. NL-projection presents the m-D data in human perceivable (2D or 3D) form based on the similarities among the inputs [57].

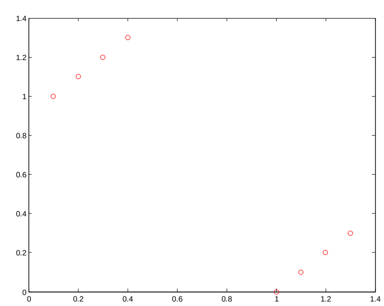
2.1 Biological inspiration.SOM.Kohonen : The inputs of different sensory (visual, tactile, acoustic) organs are mapped on to corresponding areas of cerebral cortex [195] in an ordered manner. The cerebral cortex envelops the brain and obscures other parts. A biological neuron might have finite resource necessary to maintain the incoming synapses. This might keep an upper limit on the total summed size of the incoming synapses. The artificial counterparts of somatotropic and visual maps belonging to cortical area are either erroneous or defective. Kohonen [194] reported a remedy for this task.

2.2 Architecture.Kohonen-SOM: The neurons in the co coordinating/ competing/ clustering/classification layer are in a fixed frame of 1D-, 2D- or 3D- structure containing a vector, matrix or tensors of neurons. The neurons are equidistant, but not interconnected with each other (Fig. 2).

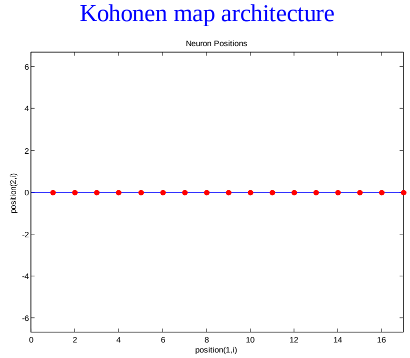
2D- Input Data – Two non overlapping linearly separable linear clusters

X1	1	1.1	1.2	1.3	0.1	0.2	0.3	0.4
X2	0	0.1	0.2	0.3	1	1.1	1.2	1.3

Scatter diagram of data

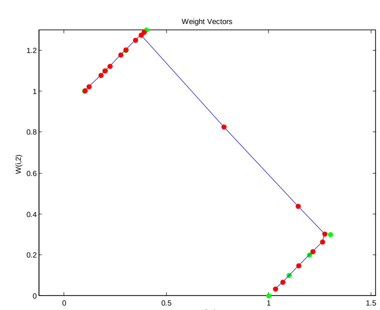


Kohonen map architecture

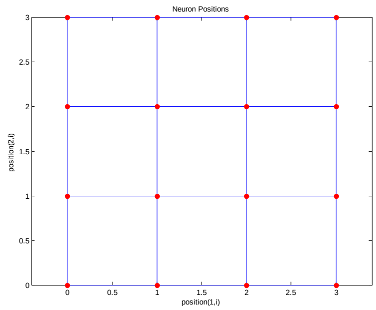


1D- 18 neurons

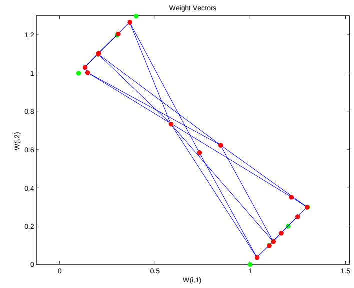
SOM display

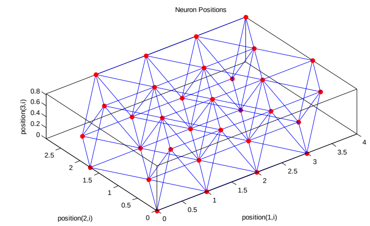


840



2D- 4 x 4 neurons; gridtop





3D 4 x4 x2 hexatop

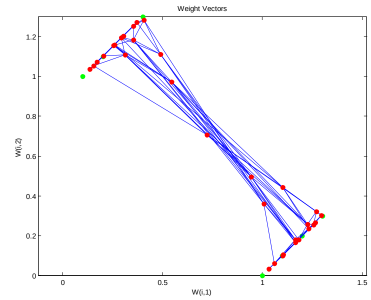
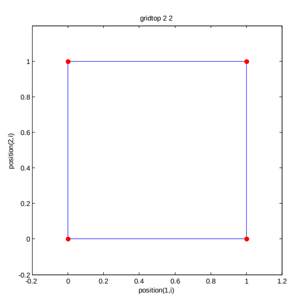


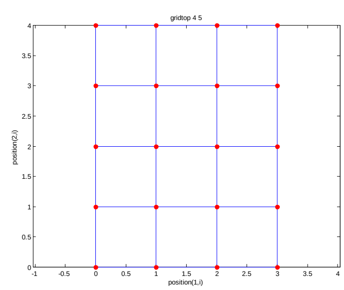
Fig. 2 SOM model of two non-overlapping linearly separable clusters with different architectures

The input layer is fully connected in the forward direction to each of the neurons in the Kohonen layer. The number of neurons in the input layer is equal to the number of variables in the data matrix. Each neuron in Kohonen layer has a single weight vector with dimension equal to input vector [13].

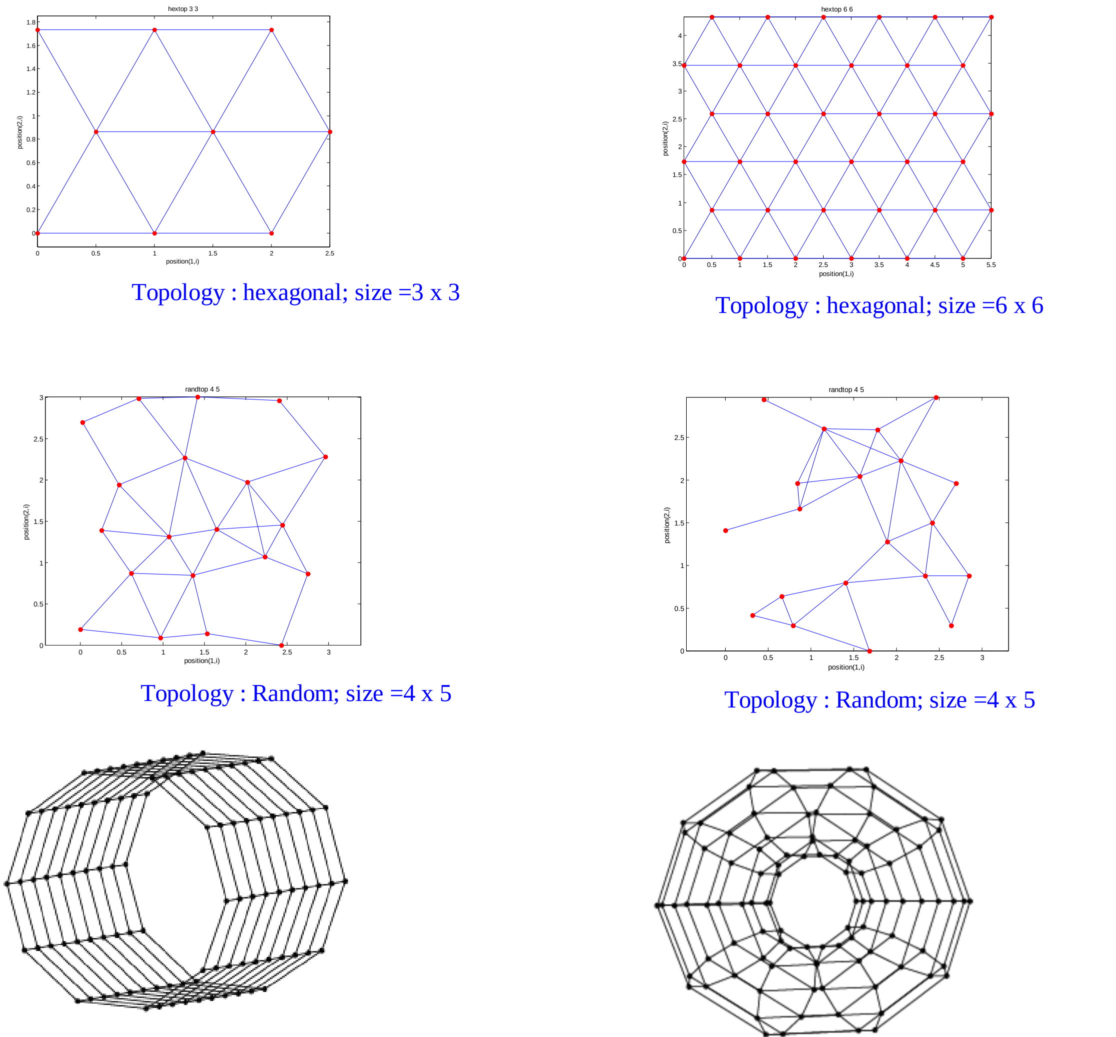
2.3 *Neighborhood architecture in SOM :* The influence of neighboring neurons on the winning one during competition is the heart of SOM philosophy. Different types of topologies viz. diamond, square, hexagonal and/or alternating among them are in vogue for neighborhood structure (Fig 3). SOM ensures realistic VQs only if topology of output grid and topology of input data are same.



Topology : square; size =2 x 2



Topology : square; size =4 x 5



Topology : hexagonal; size =3 x 3

Topology : hexagonal; size =6 x 6

Topology : Random; size =4 x 5

Topology : Random; size =4 x 5

cylinder

Toroid

Fig 3: Topologies prevalent in SOM-NN

SOM on planar triangle surface: A new SOM on planar triangle surface was recently proposed. It is derived from conformal SOM. The mapping of the model (curved seamless) surface and the sphere surface is one-to-one.

Border effect in Kohonen-SOM: The grid points at the boundary have less number of neighbors compared to the units inside the map. This inherent less neighborhood of neurons results in less number of chances for up gradation. It is referred as border effect [190] as it occurs along the border line of SOM map.

Border effect

- The net result is the W vectors of these units collapse to the center of the input space
  - Remedy: Mathematical solution – Heuristic weighting rule Local linear smoothing
    - global search for the best unit consumes a large CPU time
      - Uniform hierarchical structure of hyperbolic grid
        - accelerates the processing of the time consuming step

2.4 Data structure.SOM: The input data for today's SOM ranges from real values (binary/ floating point), images (pixels, voxels), non-numeric data (categorical, symbolic) and conceptual/ contextual sentences (abstracts, text of technical notes).

2.5 Input.SOM: At the first level, the input is 1D- to m-D matrix of real values with NP patterns/ responses/ feature values. It does not require a priori knowledge of distribution of data, a great relief to overcome the strict non-adherence of data sets to the stipulations of statistics. In Neural ware professional II software package, the dimensionality (1-D, 2-D, 3-D), shape (square, diamond, hexagonal, triangular), number of neurons in each dimension are all user chosen (Fig. 3) and fixed for a configuration.

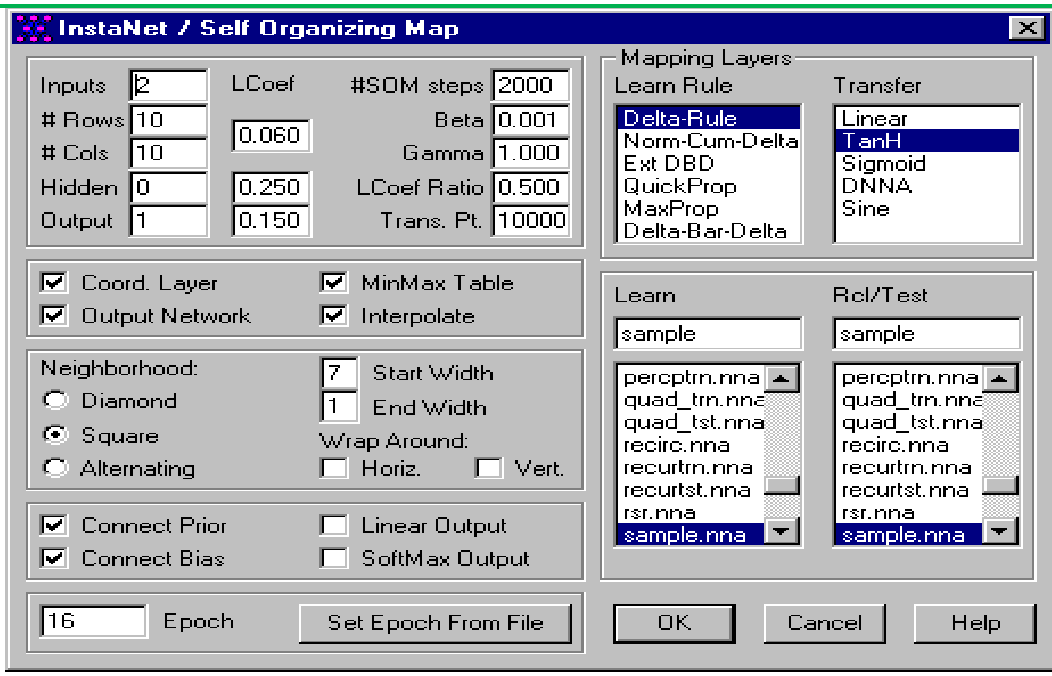


Fig. 3 : GUI frame of input for SOM in professional II neural network package

2.6 Winner takes all (WTA) :

The Euclidian distance of the Ws of PE in Kohonen layer to the incoming input is calculated. The PE with minimum distance is called a winning neuron and the mechanism winner takes all (WTA) (Alg. 1). WTA is as close as possible to the input (tensor) value and in an idealistic situation represents the output value itself. Each one of the neurons represents a cluster widely separated. In other cases, more than one neuron is necessary for each cluster. The winning neuron may be considered as a quantizer.

Alg. 1: WTA

Step	:	-1	Input X
Step	:	0	Initialisation of W
Step	:	1	Cal Euclidian distance for all PEs (D)
Step	:	2	Find the minimum of D
Step	:	3	Winning neuron ← PE with minimum D
Step	:	4	output of WTA ←1.0 output of all other neurons ← 0.0

WTA

- Adaptive learning restricted to winner takes all
- Under utilization or dead nodes hurdle
- Some neurons will never become winners due to random initialization

Remedy:
- Winner takes most (WTM)
- neural gas
- fuzzy competitive learning

2.7 Winner takes most (WTM): Recently, more than one neuron is used. The concept of next best is like in simplex optimization and the output is positive. At convergence, topological ordering in input space i.e. neurons adjacent to lattice have similar synaptic weights.

WTM

+ Result is independent of initialization of locations of prototypes
− Side effect
  Remedy : Rival penalized competitive learning

2.8 Output.SOM: The outcome is a topographic mapping of multi-dimensional data into a low (1D-,2D-,3D-) space. For instance, uni dimensional topology (1D) topology is similar to a bar. The information of a cluster is stored in Kohonen SOM as a group of nodes with short distances for patterns in a cluster and long distances for patterns in different clusters.

2.9 Functioning of SOM: It is an unsupervised NN, mostly practiced as 2-D visualization tool showing the clusters. A multi-dimensional exploratory variable (feature) data is transformed in SOM into 2-D or 3-D-space with graph invariant properties. SOM implements VQ with a fixed size of the grid and a predefined neighborhood structure around winning neuron. It employs internode's distances in a fixed output lattice. Topology preservation is the correspondence between positioning of patterns in m-D input and 2-D cluster space. It creates classes based on their distances on a plane and thus similar data elements are placed close together. Groups of neurons with short distance represent clusters. Noticeably, SOM deals with topological relationships (e.g., adjacency) among output nodes without employing any explicit model of internodes (lateral) connectivity. 1D- SOM is a simple as possible (SAP) to start with and 2D-SOM is in routine use. But, 3-D SOM finds a significant improvement. SOM maps input such that similar signals excite neurons that are close together. Neurons along with its neighbors compete to reproduce the input pattern. The process is repeated several times for all patterns to arrive at a stable system. It divides the input space into discernable categories and dynamically adjusts the size with respect to the distance to the origin [196].

2.10 Learning & training of SOM : In SOM Hebbian learning with and without forgetting schedules is used in training WIH (SOM) [195]. After each iteration of learning, all the Ws converging on to a neuron are divided by the sum of the incoming Ws (or square root of the sum of the squared Ws). W spreads over the structure of the data. It decreases with neighborhood size. W adaptation will have smaller field of influence with increase of iterations.

Hebbian learning in SOM

+ Strengthens the association between the input (stream line) and winning neurons
− Leads to unconstrained weight growth
  Remedy : W normalization

In SOM, the training is competitive, cooperative and adaptive. The quality of classification of Kohonen SOM is measured by distortion. During training not only the winning neuron but also a few neighborhood neurons learn. The neurons other than the winning are dictated by the topology and predefined radius. The learning rate decreases within the cardinality distance of a neighbor neuron from the winning neuron. The change of Ws is in tune with preserving the topological distance (information) of the input data (Alg. 2).

Alg. 2: Training algorithm of SOM
Learning rate (user chosen)
Initialization of W (code book vectors)
Repeat until maximum iterations or SOM is stabilized
    select input vector randomly
    WTA
    up gradation of W
        winner unit
        neighboring neuron
    Reduce learning rate
End repeat

If number of non-winning PEs < average frequency of neurons
Then Alter distances [→ increase in non-winning PEs]

If average frequency of number of non-winning PEs > average frequency
Then Alter distances [→ decrease in non-winning PEs]

Alg. 2b: **Conscience mechanism to find the winning set of neurons**
Input : select average frequency

If number of non-winning PEs < average frequency of neurons
Then Alter distances [→ increase in non-winning PEs]

If average frequency of number of non-winning PEs > average frequency
Then Alter distances [→ decrease in non-winning PEs]

Adjusted distance formula
+ It results in uniform data representation in SOM layer

2.11 Visual display of SOM results : There is more familiarity right from childhood to see/observe/analyze/inspect/generate 2-D color/grey scale visual world in geographic/population/political maps. Thus, the first and foremost simple desire of an end user of a soft or hard unsupervised modeling is to visualize the data clusters. Definitely, not the clustering of nodes (neurons), weight profiles or even how well the method modeled the data. The later, no doubt, are more important for the data analyst, neuro-compuational scientist, software personnel and researchers.

Dataset. Market_basket_data-*SOM:* The dataset of super market contains 199 products groups [Fig.4] with 193 639 transactions. The SOM with 60 x 40 nodes is used for the data matrix of 199 x 1999 with the entries of relative frequencies. The training algorithm is expectation maximization (EM) using a value of 1 to 3 for acceleration. Here, the number of nodes (2400) is much higher than the number of points (199) clustered and thus it is an instance of emergent-SOM.

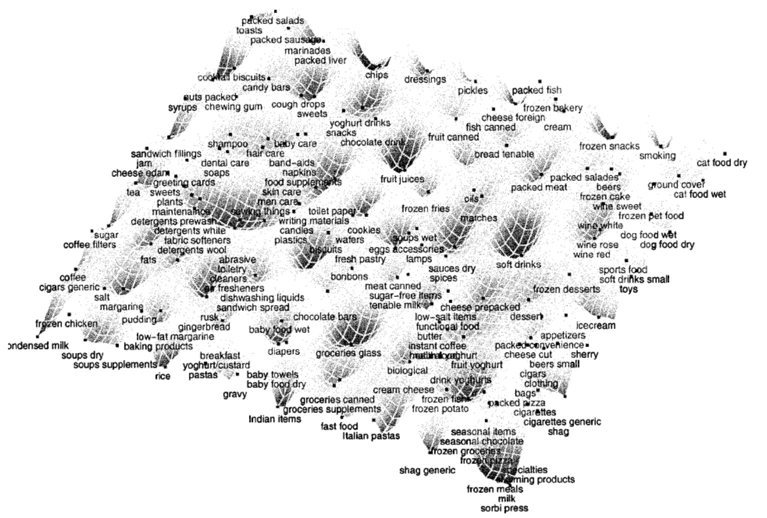
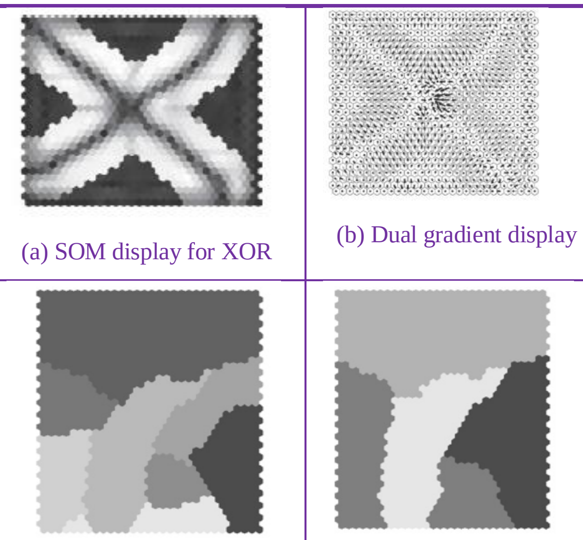


Fig. 4 SOM trained with EM for market basket data [courtesy of Ref 10]
Height: visualization of marginal probabilities of the nodes; Markers: winning nodes

SOM models are popular even among non-mathematical application practitioners due to the multi-color/grey/marker visual display of hidden correlated relationships in data of feature/multi-response spaces [205]. Code book vectors and distribution of data samples are two basic approaches in developing visuals of the results of SOM. The visual output of SOM with rectangular or orthogonal grids has exemplary legibility. The grid of SOM is non-linear and can be considered as a compromise between a high dimensional set of clusters and the 2D-plane [Fig.5] generated by any set of principal axes [202].



(a) SOM display for XOR
(b) Dual gradient display
(c) 8-clusters with k-means
(d) 4-clusters with k-means

Fig. 5: Display of artificial data sets for XOR and multiple-clusters [courtesy of Ref 202]

For each node, the visualization framework [196] allows the display of graphical attributes like 3D-graph type, colour, size, texture or text labels. For visualization of SOM output, it is desirable that all neurons receive equal geometric treatment. Some of the post-processing techniques in visual display of

SOM output are Cluster Connections, P-matrix or U-matrix and their modifications. They incorporate the distance information in the visual display by using coloring schemes.

Tree representation of SOM results: The root is placed at the threshold representing a single cluster containing the entire SOM [192]. The process of SOM gives a series of nested clusters which can be represented in a tree format. The leaves are attached at the lowest threshold, where each neuron forms a cluster of its own. The tips show the individual elements found in the corresponding cluster. Branch length is calculated as the difference between the thresholds corresponding to the ends of the branch. The lower threshold marks the point where the corresponding cluster is split. Thus, the sum of all branch lengths on the path from the loop to the last node is the same for each path. It is equal to the difference between maximal and minimal threshold values. A branch with black shade represents that no majority is found. Phylogentic trees can also be depicted in a similar manner. The corners of phylogentic trees are squared while those of SOM are round. The display of protein sequences is compared. When phylogentic tree is placed orthogonal to the SOM surface, the visual understanding is superior [206].

phylogenic tree + SOM
It is manual and requires aggregation
Remedy : Tree representation

Unified distance matrix (U-matrix): The individual neurons of the SOM are represented with the cells on a colored/grey/black/white with shading based on the average distance from this neuron to its neighbors. The black color reflects largest while white with zero distances (Fig. 6). The difference between zero and maximum distance is represented by a continuous fading black and white or the spectrum of visual colors.

Unified distance matrix (U-matrix)
Not suitable for large space SOMs
Remedy : Gradient filed technique
Dots tend to obscure the shading in large SOM maps
Indistinguishability of neurons from their borders.
Remedy: Somsonova et al [206]

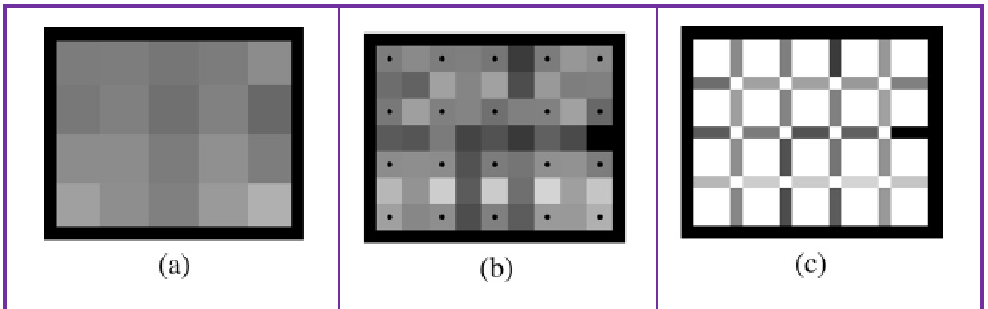


Fig. 6 Visualisation of SOM with U-matrix (a)shading as per average distance of neuron to its neighbors (b) Grid distance in (a) is doubled and dot is a mark (c) distances between the neighboring neurons indicated by shading borders [courtesy of Ref 206]

Samsonova [206] used the largest distance between any two adjacent neurons. Here, the light areas contain similar neurons. The dark areas function as boarders between the clusters. The grey areas are interpreted in two ways. The first one is that the distances between neurons are medium sized. The other possibility is that the neurons are very similar to their neighbors on one side, while very far off on the other side. This ambiguity is cleared by doubling the original grid density. The advantage is increased visual clarity where in half the cells represent neurons and the remaining their distances from the neighboring ones.

The local cluster boundaries are visually presented in U-matrix method. It is now a popular visualization technique to pin point clusters in the output of SOM. The local cluster boundaries are visually presented [205] from pair wise distances of neighboring prototype vectors. It is called unified distance matrix or U-matrix.

U*-matrix (Ultsch 2003B): It is applicable to large sized SOMs. The U-matrix value is multiplied by a scaling factor induced by the local density of the data points around the corresponding prototype vector.

The elements of U-matrix are positive sum of the distances of each node to its direct neighbors. If the data density is low the distances to the neighboring areas is high and vice versa.

SOM visualization-Cluster detection (discovery): A variant of U-matrix method is U*F approach. The cluster borders are generally depicted as black. The cluster areas are shaded based on a convention, for example, the average distance between the nodes in the cluster rather than the distance between the neighboring nodes.

U*F method
— The maps are poorly readable
∇ Reason : Shading bordered clusters areas in multiple colors
☏ Remedy : A larger size of the SOM display

P-matrix method: It displays the number of samples that are within a sphere with a certain radius around prototype vectors. The radius is a quantile of pair wise distances of the data.

Gradient filed technique: Polzlbauer [205] proposed gradient field technique. It smoothens over a broader neighborhood. This method applied is altogether a different style of representation.

Display methods disregarding topology of SOM: Prototype method belongs to this category and identifies homogeneous regions. Kaski and Kohonen [16] reported display based on gradients.

Hit histograms: A plot of names and categories mapped on to a unit shows the distribution of data. Smoothed histograms show the connections of map nodes that are close in feature space. Here, each data sample is mapped to a number of map units.

Another extension of SOM display is DIPOLSOM. It computes a distance preserving projection. The nodes are moved in an additional projection layer by employing a heuristic online adaptation rule. Map lattice is used as a platform in which the different shades of colors or markers of different size depict the quantitative information. The advances in displays in geography paved way to the improvements in SOM visualization. The projections of the single dimension of the code vectors are called component planes. The plot of component planes in all dimensions reveals all information about the prototype vectors. But, it is not easy to infer the cluster structure from these maps.

Adaptive Coordinates [105, 185, 235, 239] and Double SOM [100] allow visualizing the original structure of the data in a low-dimensional output space. They use a heuristic updating rule to move and group the output nodes in a continuous output space.

+ Post-processing techniques are not used
— Do not preserve intra-cluster and inter-cluster distances
— distances between codebook vectors are not directly represented in the map
— *(shift to visualization) U-matrix:* The clusters are visualized and it shows the relative distances between maps nodes on the whole map. The distance between W vectors of map units and their neighbors is calculated. The two individual patterns neighboring classes are close in the input space.

Kaski et al. [55] reported that a projection method necessarily makes a tradeoff between trustworthiness and continuity. The trustworthiness guarantees that at least a portion of the similarities will be perceived correctly [55]. Other measures are topology preservation [110, 112], and rank order. SOM and CCA methods have a high trustworthiness, while Isomap and Local Linear Embedding are inferior in this respect. The performance measure was defined for SOM with rectangular lattices and extension is proposed to other general lattices.

If	Visualized proximities hold in original space
Then	Trustworthy
If	All proximities of original data are Visualized
Then	Continuous

Calibration: Calibration is mapping of data on a trained SOM [206], where in each pattern is assigned to the node that is most similar to it. The result is that some nodes may get many data elements, while others none at all. The nodes with no data are

Calibration
+ The tight and loose clusters are clear from the shades

crossed out and are not used in cluster analysis. Here, the clusters are shaded according to the average distance between the data elements rather than nodes with in the corresponding cluster. Here, also white color indicates identical elements, while the black represents the largest distance between two elements in the data set. A cluster with single element is called singleton and is represented by the number encircled for singletons. The color is obviously white due to the fact the distance of any point to itself is zero.

Geometric topographic mapping [GTM]: It is proposed as an alternative to SOM, where in the output space is continuous. GTM models the probability distribution in feature space. The magnification factors **[171]** describe local stretching of the map as ellipsoids in a discrete number of lattice space centers.

Kigiwig method: Here, there is a progressive darkening of the edges indicating the stronger differences between the concerned cells [202]. The joining of centroids of the non-empty cells is called the minimum spanning tree. It is also drawn on the output map. The distances between the clusters are reflected in the visual display. The correct number of units and the stability of neighborhood relation with bootstrap procedure is used.

Toroidal: The area associated with each neuron varies significantly (larger around the outer circle and compressed near the inner circle) on the surface of a torous. Thus, it fails to offer any intuitive readable visible map.

Spherical SOM: It is visually more effective than toroidal one.

Generative topographic mapping: Samsonova [193] proposed GTM and is an extension of Kohonen SOM based on mixture models. It is based on constrained mixture Gaussians which assumes (a priori) parametric (Gaussian) pdf. GTM defines logarithm of likelihood as an object function. The centers in the data space are non-linear functions of the position of the nodes on the topological maps. The parameters are set of weights. GTM with hard wired structure is better if topology preservation is prime criteria. The nonlinear function mapping of the positions of neurons in the data space is user chosen. The parameters are optimized by maximum likelihood method like EM which guarantees convergence to a local minimum. It automatically trains many SOMs, generated by different random seed numbers. A tree representation allows calculating confidence of clusters based on consensus tree building methods [167].

Generative topographic mapping

+ Overcomes limitations of Kohonen SOM
+ Allows non-linear transformation

Consensus tree**:** It represents an average of a set of trees with frequencies of occurrence of its branches compared to the set of all trees representing reliable clusters as sub-trees.

Consensus tree

+ It provides a cluster hierarchy
+ The map reveals spatial ordering of clusters
+ Enables one to view the clusters from different perspectives

Visualization induced-SOM (Vi-SOM): Yin [107] proposed Visualization induced-SOM which is an extension of SOM. It keeps the position of proto-vectors approximately equidistant. The advantage is that it captures the characteristics of the data set, but avoids post processing. ViSOM constrains the lateral contraction force between neurons in the SOM. It allows preserving the inter-point distances on the input data on the map, along with the topology

Visualization induced-SOM

+ Preserves the inter-neuron distances in the map
− Fixed grid structure of neurons
− Uniform distribution of the codebook vectors in the input space
− Requires a large number of codebooks to get an adequate quantization error
− Heavy computational load
  - Remedy : Local linear projection procedures

Curvilinear component analysis

- Computational complexity of CCA is O(N), while MDS and NLM are O(N^2)
- Cost function of CCA allows unfolding even strong nonlinear or closed structures.
- Output is a continuous space that is able to take the shape of the data manifold
- Topology is not a fixed grid

Curvilinear component analysis (CCA): CCA [111] performs vector quantisation of the data in input space using SOM. It makes a nonlinear projection of the quantizing vectors. The cost function is minimization of inter-point distances. The projection module is similar to multidimensional scaling (MDS) or Sammon's mapping (NLM) [127]. Lee proposed enhanced version of CCA incorporating curvilinear distances instead of Euclidean distances in the input space [232].

Dittenbach [105,185, 205, 235] attempted to bring out cluster structures as a part of the topology of SOM. These efforts resulted in many flexible topologies.

Tree view SOM : Freeman and Yin [225] proposed tree-view SOM in this decade surpassing the visual earlier display procedures for large (text) databases. . A set of independently spanned, growing 1D-SOMs are automatically organized in a dynamic hierarchy during training to categorize and organize documents. The depth and coverage of root-SOM/subsequent levels are fully adaptive and dynamic. The limitations of earlier popular procedures are surmounted in this new display algorithm. The other hierarchical structures or more simply tree-view structures are Tewey decimal classification (TDC), file explorer, web-portals or web-directories.

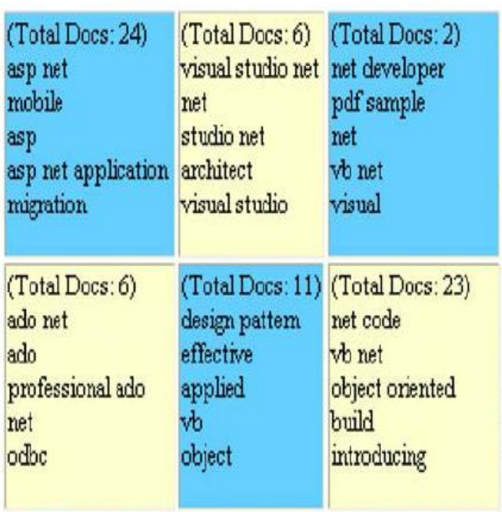
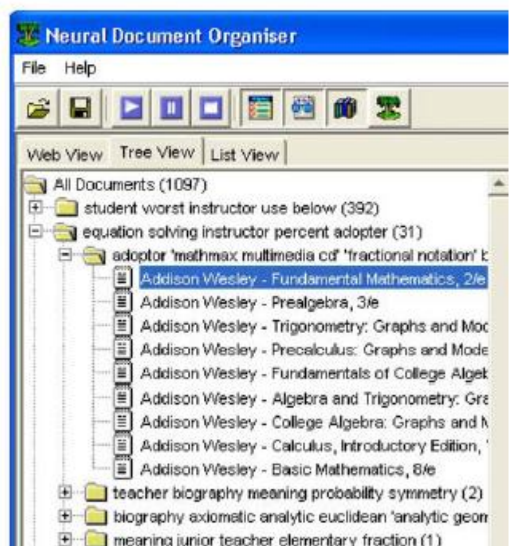


Fig. 7(a) Exert of a 2D (6 x 6) -SOM display of classification of documents ; Number of documents clustered around each node are in parentheses



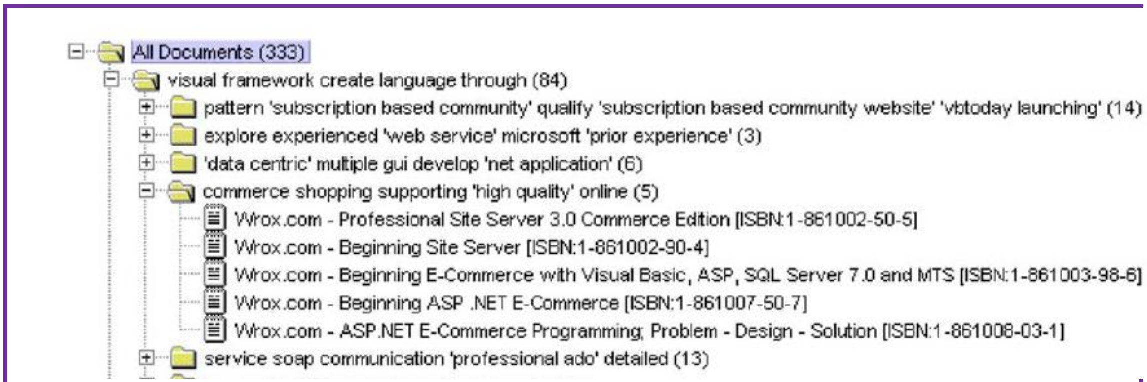


Fig. 7 (b) Exert of SOM created taxonomy display of documents ; Number of documents clustered around each node [courtesy of Ref 225]

Architecture.Tree view SOM: Tree view SOM consists of a set of growing and independently generated 1D-SOMs. They are organized in hierarchical manner. The training is similar to SOM. The input document vector is mapped to 1D-topology of neurons. The number of unique terms in the collection plays a role in the dimension of W. The output of a tree-view SOM is a list of topics in a hierarchical structure. They are presented similar to the most of computer like file management systems in an intuitive way. The topics which are judged to be similar are located closely at each level in the hierarchy.
Dataset.books.Tree view SOM: The dataset containing 333 documents (8Mb) deals with technical programming books including web client-side. The output generated by 2D-SOM (6 x 6) topology divides the books. The number of documents is given at the top in parentheses.
Dataset.accounting.Tree-view-SOM: The first dataset contains 618 documents (20Mb) pertaining to accounting, computing, sociology, business and engineering. It is successfully analyzed with tree-view-SOM. In this dataset [Fig. 7] the vocabulary is diverse for each topic resulting in a sparse matrix representation for document versus words. Further, the topics are overlapping and implicit structure of documents is hierarchical. It is not possible to cluster them with yesteryears' algorithms. The second data set is about technical programming books including web client-side (JavaScript and Dynamic html), web server-side (ASP, .NET and CGI) and programming languages (C^{++}, C] and Java). The description is from a paragraph about the essence of the book to complete details. These datasets have many overlapping topics leading to sparse vocabulary. The hierarchical and partitioning algorithms do not function efficiently.

$$\arg\min_{i\in A}\left(\left\|Bal.fac.-w_i\right\|-Bal.fac*\left(\frac{1}{N}-vig.par\right)\right)$$

Limitations of typical cluster display-methods

- hierarchical clustering algorithms
  - manual decision where to cut dendrogram
- k-means
  - a priori knowledge of number of clusters
- GCS
  - pre-processing to visualize the tree
    - Coloring schemes
    - Complicated grid neighborhood relations
- SOM

–	If	size of the map is small
	Then	dis-similar objects (i.e. objects belonging to different classes) are forced together
	If	size of the map is large
	Then	groups are separated on the map by empty node
	If	size of the map is too large
	Then	objects of the groups are divided over different nodes

Treeview-SOM

- + Documents belonging to multiple clusters can easily be identified
- + The cluster growth process is automated i.e. no decision is needed on where to cut the dendrogram
- + Efficient initialization of W of child maps with inherited values from parent nodes
- + Display layout is user friendly like in most file managers/web directories
- + More effective in information retrieval and visualization compared to Kohonen SOM
- + Organizes documents in 1D-space, providing clearer and insightful taxonomies
- + Relationships are retained efficiently
- + Retains nonlinear trends and preserves topology
- + No post processing or further identification of clusters for visual display
- + Improved navigation and visualization

3. Applications of SOM : The fields of applications are all in science, engineering, commerce, social sciences, industrial activities and progress is both need based and advances in tools of mathematical/computer science.

3.1 Cortical development model: SOM is instrumental for a model of cortical development. Choe [47] showed the importance of lateral connections in contour integration and segmentation. Sirosh [172] reported simultaneous development of receptive field properties and lateral interactions in a realistic model of primary visual cortex.

3.2 Visual exploratory data analysis (VEDA): SOM is a method of choice for visualisation of multi-dimensional unsupervised data in 2D- and 3D- dimensions. In the application domain, the popularity grew as it does not require much pre-processing, transformation or projection into other spaces. It is a competing approach for PCA and other unsupervised techniques. Mostly, SOM is applied in off line learning. It is used as a front-end module in counter-propagation NNs. The centers in RBF are also calculated with SOM.

3.3 Chemical Science: SOM is applied to divide aqueous solubility of 1293 Compounds [146] into training and testing datasets, classification of photochemical reactions/physicochemical properties of the bonds [155], proteins [152] , Petroleum distillate (MPD) products [24], discrimination of toxic-nontoxic compounds [151], aromaticity [49], green chemistry [17], and solid state NMR for 72 siloxane-based phosphine hybrid polymers [148].

Food science: In Classification of dry-cured hams NIR [186], Classification of available food base/diet of 52 small perch and 38 ruffe specimens [72], adulteration of extra virgin olive oil (EVOO) [142], 3-way data analysis of 50 PAHs from crude/fuel oils spilled under controlled experimental conditions over a period of four months [26], strawberry aroma [143] using HS-SPME-GC-MS data. heavy oil intelligent processing [74] and knowledge extraction from plant input-output data [136] SOM was employed.

Process chemistry: In process engineering [28] of nonlinear processes [135], online automated monitoring tool in the plant's distributed control system [131], process route selection early in development of amino acid sequences of 41 proteins in Industrial operation [89], Chemical composition assessment of produced water in oil wells [93], multi-model fusion strategy in brewing industry with NIR/MIR instrumental data [261], discrimination of sandal wood oil grown in different conditions and extraction methods using NIR spectra [90] the modeling is performed with Kohonen SOM.

3.4 Biological processes: Mining biological data/rule extraction of protein sequences [51], metabolic diversity in Type 1 diabetes [164], metabolic profiling with NMR multiclass SOM discrimination index (SOMDI) from 96 samples of human saliva [59], gene-expression levels microarray experimental data [19], kinase inhibitors [245] based 3D-spacial descriptors, conformational analysis of lipids [76], molecular mechanism of hormetic effects of selective serotonin reuptake inhibitors (SSRIs) in Daphnia magna reproduction [78], prediction of cellular uptake of 109 magneto fluorescent nanoparticles (NPs) in pancreatic cancer cells [130], inhibition of *β*-amyloid aggregation by 62 *N*-phenylanthranilic acids [79], screening of 82 5-aryl-2-thio-1,3,4-oxadiazole derivatives for anti-mycobacterial activities against *Mycobacterium tuberculosis* H37Rv using electronic-topological descriptors [54], quality control index of continuous pharmaceutical process using online HPLC [25], pharmaceuticals [56], relationship between chemotypes and screened agents from NCI antitumor drug screening data [160], Clustering Biological data [83], contamination of the breast milk with PAH [262], phylogenetic diversity of gene sequences [220] and changes in gene expression from microarrays comprising of 18,000 human gene/EST sequences [58] employed SOM in the modeling study. SOTA is applied to study familial binding profiles (Sandelin 2007). FBPs are used to classify a novel motif and to restrict motif finders for finding a specific class of motifs. SOM-biological regulatory element (SOM-BREO) [192] (BP-SOM) characterizes a complete set of motifs and simultaneously separates weak motif signals.

Bioinformatics: In bioinformatics, the identification of short DNA sequence motifs is a critical issue at the moment. Statistical unsupervised learning methods were in practice in the discovery of motifs. The scaling of difficulties for large genomic data bases have to be addressed from a different frame like artificial intelligence-2 (AI2). Mahony et al. [192] proposed Kohonen SOM, viewing the motif identification as a clustering task. The sequence databases are considered as a set of short overlapping substrings. Based on the similarity of the sequences clusters are developed which can be put into different bins.

3.5 Environmental science: The unsupervised self organizing technique, SOM played a key role in clustering 25 micro watersheds in Rajasthan into homogeneous groups [159], Surface water quality assessment [60] and to reduce irrelevant information in Water quality assessment by Hasse diagram [31], cloud classification [237], and crop evapotranspiration [76].

Waste management: Waste water treatment plant processes are dynamic and involve temporal variability of inflow and concentrations of components like Municipal activated-Sludge [265]. Each of the micro-processes are complex and many a time poorly (particularly) known viz. interaction among different unit processes –hydro dynamic phenomenon, adoptive responses of living micro organisms. Further, the cause and effect relationship between the process variables is strongly non-linear. Added to it, limitations exist in measurement of dynamic operation (performance) of waste water treatment plant (WTP) by direct means. Evolutionary self organizing model for dynamic behavior of WTP [Hong 2003] not only predicted the process behavior accurately but paved way to probe into the dynamic behavior of partially known WTP.

3.6 Structure X Relationships (SXR): The QSAR studies of inhibitory activity of 117 Aurora-A kinase inhibitors [53], dihydrofolate reductase (DHFR) inhibition compounds [77, 150, 154], QSBioactR in 404 Acetylcholinesterase [52], acute toxicity for over 300 benzo-triazoles ((B)TAZs [158], structure –

biodegradability relationships in PCBs [169] and prediction of decaying concentration profiles of BPA, (*p*-boronophenylalanine) in blood during BNCT therapy [33] involved SOM in the process of modeling

3.7 Medical diagnosis: SOM found a niche in adolescent idiopathic scoliosis detection among 1,776 surgically treated patients [263], classification of abnormal brain image [246], C93 identification in blader cancer patients [81], classification of perfusion abnormalities using computed tomography perfusion (CTP) maps analysis [243], molecular subtyping of cancer [82], classification of 186 chemicals and 117 drugs causing rhadomyolysis [264] and in lower body coordination with different types of foot orthoses [98].

SOM is used in cortical motor map training [176] and in recognizing psychographic and cognitive factors on organ donation in Egypt [86]. The target plan of UK's National Health Service (NHS) is to sequence the genomes of up to 100,000 patients in the anonymized mode not to reveal the identity of individuals. It probes into DNA information to unlock the stumble blocks hurdling today's promotion of better/ sure-drugs. The outcome of this shrewd venture is a centralized database of whole genome sequencing for high quality diagnostic tools making for probable access to genomic tests. This mega projects to a tune of 100 million UKP trusts to provide high quality prospective health care in the next decade.

3.8 Training of FFNN: Nasr and Chtourou [39] proposed the learning of weights of NN with a hybrid algorithm. The first phase is a structure learning process by the addition of hidden neurons followed by optimization of the network parameters. The weights between input and hidden neurons are refined by SOM with a fuzzy neighborhood. Gradient method is used for optimizing weights of connections from hidden to output neurons. This hybrid learning scheme is superior to yesteryears' procedures for a simulated test set.

3.9 Classification/Discrimination/clustering

Feature selection methods: SOM for structured (numerical/attribute) data excelled many classical clustering procedures. The extension to graph structured information is of recent interest and it is extended for cyclic and directed graphs. The clusters are formed in the state space of SOM to represent the strengths of activation of neighboring vertices. In the previous ventures the state-space of the surrounding vertices is used to represent the strengths of activations. Conan-Guez [203] used dis-similarity-Kohonen-SOM to protein clustering, string clustering, and spectrometric data [203]. SOM is used for feature selection in the prediction of properties (including density, viscosity, methanol content, and water concentration) of biodiesel fuel [32], classification of Felder-Silverman learning styles, automatic determination of the number of clusters and detecting clusters of complex shapes [247], discrete data clustering [38], automatic classification method [255], automatic-cluster detection [115], noise removal in clustering [69] and Web 2.0 tool for creating intelligent adaptive tutoring systems for mobile learning environments [88]. Using self-organizing-incremental-NNs, adjusted-SO-inc-NN classifier is proposed. It automatically learns the number of prototypes required to determine the decision boundary. It learns new information without destroying old learned information, robust to noisy data and fast.

Fault detection: The fault detection in induction-machine-stator-winding, determination of centers of fuzzy cluster [36], extracting fuzzy rules from Kohonen Self-Organizing Map for transformer failure diagnosis [141], random early detection (RED) at a router output link during congestion [75], sensor fault detection/isolation [16] in desalination plant operation with reverse osmosis (RO) [1] had a new phase with SOM compared to PCA and Eigen vector analysis.

Internet and Web: SOM is extensively applied in analysis of web usage data [203] and in web document mining.

Dynamic systems: The variants of dissimilarity SOM is applied for time-series data and in internal parameter changes in a stationary, non-linear SISO dynamic system [22]. Three recursive SOMs (viz. SOMSD, MSOM, Recur-SOM) perform modeling data with general structures like sequences and trees. The efficiency of the model is based on unit's memory depth, differentiation among trees, statistics of label's distribution and spacio-temporal information encoded in the map. The datasets used are binary syntactic tree, ternary linguistic proposition and 5-ary graphical data.

3.10 Electrical engineering: One-day-ahead forecast of Spanish electricity market load using weather data [73] and electricity demand assessment by predicting the daily peak load for the next month [40] applied SOM network.

Communication system: In communication systems, equalizers are used in high speed modems while echo-cancelar for long distance telephone (Widrow 1990). An equalization task involves the recovery of information at the receiver. During transmission from source, the signal is subjected to noise, inter-system interference, co-channel/adjacent channel interference, non-linear distortion and fading and many of them are varying with time. Barreto [198] used SOM for nonlinear channel equalization and inverse mapping identification. Kohonen SOM played a significant role in "intelligent computation" and "adaptation" capability for wireless sensor networks [254].

3.11 Travelling Sales problem (TSP): It searches for the shortest closed tour [57] with the constraint to visit each city only once. It is a NP-hard complete (1) task. Hopfield used NNs for the first time to solve TSP using the minimization of energy function. At the point of convergence the local minimum corresponds to good solution. Bai [57] used twelve test problems for TSP with different SOM procedures, although many others solved TSP using SOMs. He used an efficient initialization method.

TSP with Hopfield NN

– It does not ensure feasibility of the tour. In other words, the paths at the minima of the energy function do not result in feasible path ways to traverse for the travelling salesman.

3.12 Commerce: The forecast of financial failure scenario [24] and forecasting horizon of a financial failure model [70] are investigated with SOM. CRI scheme of Zedah is mapped on to generate SOM fuzzy NN to synthesize gen-SOM-fuzzy-NN-CRI (S) NN. It is applied for classification and prediction of failures of banks. It results in positive as well as negative rules and consistently performs better than COX model. MLP of course has superior performance but the architecture is a black box. Modified cerebellar model articulation controller (MCMAC) (ref in abstract) is also better than gen.SOM fuzzy NN CRI.

3.13 Economics: SOM successfully evaluated poverty, welfare and development indicators [245] in social development scenario.

SOM was applied to image data compression [124], perpetual pattern recognition [213], curved trajectory prediction [138] and forecasting [231]. Further, Kalman filtering [175], adaptive filting [198], structured data unsupervised processing [230] and PCA [217] were implemented using SOM.

4. Advantages and limitations of Kohonen-SOM : In the original SOM, the dimensionality (1-D, 2-D, 3-D), shape (square, diamond, hexagonal, triangular), number of neurons in each dimension are all user chosen and fixed for a configuration and thus one can concentrate on problem on hand. However, this fixed structure of SOM limits the adaptability in complex tasks. Automatic selection procedures prevalent in MLP, RBF etc also apply and results n popular intelligent software. In fact, a set of heuristics implemented in traditional programming languages does the job. Several researchers contributed to the development of self-

Advantages.SOM

+ No need of a priori knowledge of distribution of input data
+ Training preserves the topology of input space
+ Reduction of dimension of input space
+ For each neuron a potential function is used
+ SOM is superior to PCA, PLSA, MDS and orthogonalizing approaches
+ SOM performs better than classical SCL
+ Preferable to VQs even where topological preservation is not of interest
+ Batch procedures are faster especially in high dimensional space

creating/self-growing/self-pruning/self-adaptive software and self-reconfigurable hardware NNs. In the context of SOM, growing neural gas and growing cell structures are noteworthy categories. Further, Kohonen-SOM and crisp clustering algorithms cannot cope up with ambiguity in applications. Tsao [252] reported that lack of sound optimization and convergence criteria add to the limitations.

Limitations-remedial measures.SOM(classical)

Architecture

- User chosen 1D-, 2D- or 3-D fixed structure of Kohonen layer
- User chosen neighborhood (shape and size) structure
  - **Remedy:** Growing cell structures-SOM, Tree-view-SOM
- suboptimal as data topology depends on the task

Object Function

- No object, cost, or energy function [175]
  - **Remedy:** Neural gas-NN

Learning

- Topological mismatches are more in batch mode compared to the online SOM
- WTA is most time consuming step
  - **Remedy:** H^2 SOM
- selection of learning rate and decreasing function
  - **Remedy:** RPSOM (rival penalized SOM)

Input

- It does not deal with symbolic data
  - **Remedy:** Symbolic SOM
- Toplogy of input data is not known in advance
  - **Remedy:** Greedy-Granular-SOM
- SOM does not reflect the input space (as it is uniformly distributed in the output space)
- Hierarchical relationship cannot be detected in a single SOM
  - **Remedy:** Hierarchical-SOM
- Noisy data/outliers affect output accuracy

Oder of Presentation of input patterns to SOM

- Order of presentation and initialization process results in different clusters
  - **Remedy:** ensemble of SOM-NNs with varying random seeds
    - Lengthy procedure
    - Not automated easily
    - intractable by manual analysis for large dataset
- Linear TF in SOM produces multitude of simultaneous responses to a mixture of superimposed stimuli
- Termination is not based on optimizing any model of the process or its data
  - **Remedy:** Greedy-Granular-SOM
- **Output**
- Several interpretations of SOM output
  - **Remedy:** increasing stability of neighborhood structure
    - prunes number of possible interpretations by
- **CPU time**
- Large CPU time for global search
  - **Remedy:** Uniform hierarchical structure of hyperbolic grid
  - Growing hierarchical SOM
- Crisp-SOM captures local-first order statistics in data
  - **Remedy:** Greedy-Granular-SOM
- It is a heuristic approach

SOM is equivalent to: SOM is equivalent to regularized mixture models with additional regularization. SOM learning is equivalent to EM. If β → inf, then EM is called Batch-map algorithm [Kohonen 1998, Cheng 1997]. Here, there is no neighborhood averaging in E-step.

If	$NP \to \infty$
Then	Learning dynamics cannot be described by a gradient descent distortion measure

SOM is similar to MDS, popular in statistics. A batch SOM algorithm, on the other hand, is similar to the Forgy BVQ algorithm. SOM based adaptive filter can be viewed as a network of local experts. The competitive nature on SOM based filters can be reduced to modular networks. SOM is comparable to elastic net approach. It is a special class of NNs based on denominated competitive NNs. Each neuron competes with others to get activated. At any given moment the outcome is that only one output neuron is activated. SOM is proven to be approximation of gradient of distortion measure. Kohonen map is proven that it converges some times on equilibrium points.

SOM reduces to: SOM without lateral interaction reduces to standard VQ. With no neighborhood (i.e. number of neighbors = 0), SOM becomes SCL (simple competitive learning) algorithm, in its classical stochastic form. That is why, SCL is also called 0-neighbor Kohonen algorithm. Two SOMs are linked via the method of winning neuron. The winner is selected and centers ($ws1_i$ *and* $ws2_i$) of first and second space are upgraded. The winner is redefined in order to surmount the failure condition.

4. Advances in SOM research : The new research pursuits since by Kohonen proposed SOM two decades ago were in the multiple directions; extending to all types of data (numeric, symbolic, abstracts, technical-notes etc), novel structures in architecture, learning algorithms, neighborhood patterns, decreasing CPU time, preservation of topology in the raw datasets on a strict measure, increasing in visualisation of output for knowledge extraction etc. The recent efforts are around growing structures, increasing function of a neuron, hybridisation with other tools, and trying to reach ultimate self-adaptive, self-corrective, self-repairing, self evolving SOMs for multivariate multidimensional data. A synopsis of major improvements in learning, architectural breakthroughs, impact of fuzzy theory and extension to mega databases follow.

Training : The error minimization is the top priority of hitherto available statistical/mathematical procedures. A concept named 'enhancement learning' based on information-theoretic approach is used to train SOM model. The information from several network configurations is combined through extraction of features common to all configurations and also specific to some configurations. The relative information results in attention to a more valid network. The results of this method on IRIS-flowers and cancer datasets showed reliable determination of number of clusters. Rousset [201] reported an increase in reliability of SOM with Homeo-static synaptic scaling [195]. It leads to proper organized SOM map (compared to standard W normalization), better representation of input probability distribution (in comparison with normalization of weights) and drives the network to a state of increasing information transfer. Seo [183] employed deterministic annealing in SOM modeling. Furao [191] used incremental learning in SOM.

Neighborhood structure

Robust-MAP: The Robust-map (Alg. 3) is a selected structure which minimizes the distance D-of the different solutions of SOM [201]. In other words, it is one closest to the aggregation of individual measures and corresponds to the most common interpretation of data structure. Robust-map sheds light on the classification as well as the neighborhood structure between classes. It is applied to classification of daily electrical consumption profiles and financial classification. Its ability to adjust to the data structure indicates the relevance of chosen NN model.

Alg. 3: R-MAP-SOM
Divide input into several groups
For each group
Train with SOM
End group
RobustMap ← map (MinDist)

Robust-MAP-SOM

+ Neighborhood structure is most robust compared to any randomly selected map
+ Preserves the global topology

+	More stable to the choice of
○	Sampling method
○	Learning algorithm of SOM
▪	Initialization
▪	Order of presentation of presentation
—	It is local (at individual level) rather than global

Distance, similarity, dissimilarity measures: For a long time, the popular index of similarity metric is Euclidian distance. The disadvantage is that clusters resulted tend to be isotropic form. Further they cannot account for local distractions or correlation of data. Recently, a local PCA-SOM which implicitly uses Mahalanobis distance and reconstruction error is proposed. It uses a covariance metrics which contains local data distribution and does not require knowledge of number of principal components. A general metric is also used with an advantage of Ellipsoidal clusters. This is tested with Gaussian clusters of spirals, checkerboard, UCI classification and image compression datasets.

Architecture

NN for a neuron in SOM architecture: In a conventional SOM, the neurons are arranged in 1D-, 2D- or 3D-grid. In modular-SOM-NN-of-NNs, there is a NN instead of a neuron in the SOM topology. In general, any trainable NN can be used. The system learns a set (group) of functional relationships (or systems in parallel). The output generates a feature map of these input-output relationships. This NN has a function space rather than vector space. The real time meteorological dynamics map and simulated cubic functions are tested with success. **A** SOM on planar triangle surface and another rectangular SOM architecture are proposed with a prospective outcome.

Symbolic SOM: In contrast to numerical values, attributes, multi-labels and text belonging to symbolic data [96, 250] also prevail in real time applications. SOM was modified to suit to model categorical (qualitative) data. Here, instead of distance measure among feature variables, a probabilistic framework without any assumption of distribution of data is employed. Each unit in SOM is upgraded based on approximation of a discrete distribution. This SOM is trained with a learning rule based on stochastic approximation theory. The applications include inducing descriptive decision making knowledge from classification data, large vocabulary continuous speech recognition systems (LVCSR), Speech recognition from non-fluent and fluent utterance records [64] and Polish language processing [242]. Symbolic data analysis provides suitable tools for managing aggregated data described by partitioning interval data [244]. Kohonen-SOM is modified for non-vectorial data [193]. Yang et al [96] proposed symbolic-SOM wherein a cluster center is a structure and contains events and associated memberships. The structure of this symbolic neuron can be refined during training phase. The fuzzy c-means method expands the largest membership degree while suppressing those of others. It is used as a learning rule for these neurons. The limitation of this SOM is that feature map display like conventional SOM is not possible since input data and neurons are a symbolic type.

*Dataset.classification.*Symb-SOM: The fat oil data set consists of eight types of oils with four physico-chemical variables with interval values and one qualitative characteristic. Three symbolic neurons are adequate in the cluster analysis.

Oil	Gravity	Freezing point	io.vlaue	sa.value	m.f.acids
Linseed	0.930 to 0.935	-27 to -8	170 to 204	118 to 196	L,Ln,O,P,M
Perilla	0.930 to 0.937	-5 to -4	192 to 208	188 to 197	L,Ln,O,P,S
Cotton-seed	0.916 to 0.918	-6 to -1	99 to 113	189 to 198	L,O,P,M,S
…….					

The other data sets analyzed with symb.SOM are classification of 37 cities in the world with interval temperature data over a year, simulated four cluster data with varying covariance and centers and a four cluster soybean data set with 47 sets objects and 35 qualitative features. The results for these real symbolic data sets crossed the test for feasibility and deserve deeper study.

Faster versions of SOM

Tree-SOM: Samsonova [206] proposed tree-SOM which divides SOM into nested clusters at different threshold values. The software in C^{++} is available as an open source. A factor of 5.5 times fastness compared to Kohonen-SOM was achieved by reducing a number of time intensive steps. The typical ones are replacing linked lists/arrays and computing full distances only if necessary. The outcome is segregation of data as well as clusters in hierarchical manner. This method functions well even for data with missing values. The datasets (abalone, protein localization sites and voting behavior of different countries during the yearly EuroVision Song Contests) are analysed with success.

Novel parallel clustering algorithms based on the Kohonen's SOM: The heuristics proposed [95] maximize the speed and at the same minimizing the topological error. In the first two algorithms each node executes an on-line SOM. The third algorithm executes as a quasi-batch SOM. The weights computed by the slave nodes are recombined by the master nodes. Then, the next epoch of SOM continues until convergence. It outperforms the currently available methods for parallelizing the SOM. A case study from bioinformatics revealed meaningful clusters are arrived in massive data mining rapidly from CPU time point of view. The data is divided among the nodes.

Experimental design (ED): SOMs have several adaptable/tunable parameters and the selection of appropriate network architectures is required in order to make accurate predictions. The Effects of network size, training epochs and learning rate are optimization influencing factors [30]. Hitherto, this is performed manually in a custom mode varying one factor at a time. Recently, statistical experimental design which brought renaissance in chemistry, pharmacy, food science entered clustering. A set of five variables (viz., type of SOMs, training algorithm, topology, boundary condition and weights initialization) at two level factorial design (FD) is used to maximize performance of classification. The samples are divided into 80%training and 20% testing maintaining number of samples ratio. The procedure was repeated 30 times to estimate statistical error. A noteworthy inference from ANOVA is that the effect of architecture (CP-, XY-fusion, supervised SOM) has profound influence on classification.

Parallel SOM: Classical SOMs process patterns (i=1 to NP) one by one and refines the NN model. In parallel SOM, the whole input is processed in parallel [204] and the patterns are learnt. Or in other words, values of W and neighborhood structure are refined. The advantage in this case is a priori knowledge of input space can be utilized to reorganize the parts of the patterns.

Supervised SOM: Kohonen introduced LVQ (learning vector quantization), a supervised version of SOM in 1987. During this quarter century, advances in LVQ include generalized-,Yizhak-, Cline-, information-theory-based-, generalized-relevance-, fuzzy-, ordered-weighted-LVQs and hybridisation with simulated annealing algorithm (SAA) and fuzzy system. In the case of supervised neural-gas newer methods viz. supervised relevance NG, Median-, winner-relaxing-, growing, robust-growing-NG algorithms are proposed. The details and applications of these supervised NNs and counter-propagation will be detailed elsewhere.

Counter propagation NN
- Train Kohonen layer
- Pad output Kohonen layer into a hidden layer
- Use BP to train hidden and output layer
- categorical layer contains predictive values

Evolution + SOM

4.1 Self evolving SOM NN : Wu [120] proposed self_organizing-self_evoling neural network. It is superior to a single SAA in optimization and CPU time. SOSENS is population based optimization algorithms using multiple SAs with self evolving and self organizing capabilities. Tabu search can be used instead of SA, but it is a local search method and cannot guarantee the global optimum. The weight of a winner neuron representing best solution at a time is the input. The set of candidate solutions generally used in GA/PSO (population based algorithms) are the weights connecting the input neuron to the neurons in Kohonen layer.

Architecture.self evolving NN: 2D-rectangular or hexagonal grid of self-evolving neurons forms SOM-layer. Each neuron in the layer performs simulated annealing (SA) optimization (Alg.4). A 3 x 3 grid is chosen with solid circles as the initial position of the neuron. The solid lines are the connection between neighboring neurons. When all the SAs of SOSENs evolved and reached equilibrium temperature, solid circles are the positions of the neurons and dotted lines represent the connections between neighboring neurons. Neuron-3 is the winner, since its position is the nearest candidate to the global minimum (around -0.4). The new positions of other neurons are self organized around the winner neurons. All neurons evolve in their respective local optima by SA. After self evolving, all co-ordinates are self organized towards the neuron with the best optimum value at a time.

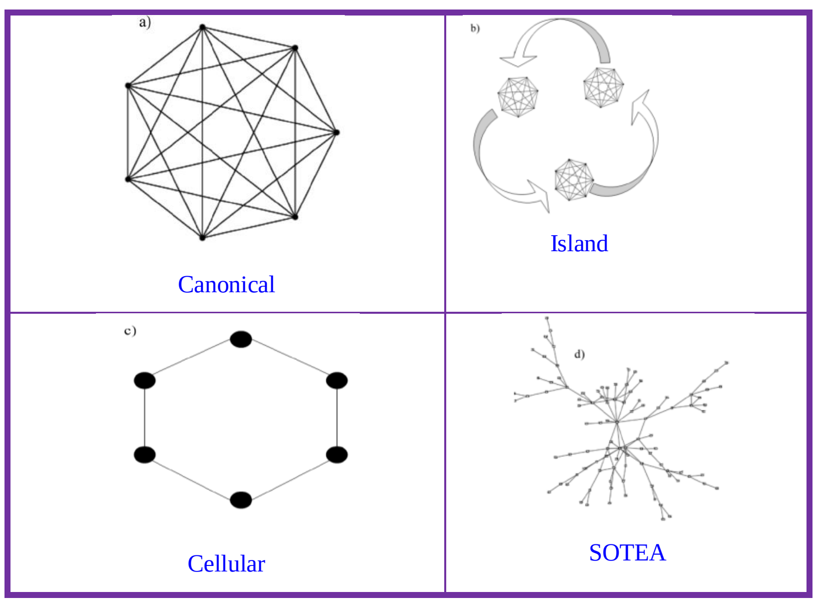
Alg. 4: SO-Self Evol-NN
Initialisation
Random Ws
Initial temperature (T0) for SAA (user option)
DO until convergence
    Each neuron evolves by SAA in parallel
    Repeat until T0Equil
        Find winner neuron among grid
        Upgrade weights
    End repeat
    Decrease temperature
End do

If SO-Self Evol-NN have one candidate & There is no self organizing behavior
Then SO-Self Evol-NN is equal to SA

*Dataset.optimization.*SO-Self Evol-NN*:* Sixteen test optimization functions, each with 100 variables are optimized with SOSENs (6 x 6). Each target function is run 100 times with PSO, DE, SOMA (SO-migrating algorithm) and SA. The neighborhood radius is 6 and population size is between 20 to 60. The difference between best value in the current and previous iterations less than 10^{-5} is criterion for convergence.

*Dataset.TSP.*SO-SelfEvol-NN*:* Discrete TSP is non-deterministic polynomial time (NP) hard task. It is typical that can be extended to vehicle routing/scheduling, PCB design etc. Lin-Kennigham (LK) algorithm is used to choose the neighborhood of the neuron in SOSEN. The results are tabulated. The single SAA for TSP is equivalent to SOSEN-NN with only one neuron and without up gradation of winner neuron step. The number of cities range from 318 to 4461.

SO-Self-Evol-NN
+ Chance of reaching global optimum increases compared SAA
    Reason : multiple SAs run in parallel in each epoch



4.2 Self organizing topology EA (SOTopolEA): Self organization of local neuron structure and interaction epistasis is introduced in EA.

Motivated by complex biological systems in development of structures relevant to the behaviour, self organization of interaction networks are proposed. The fitness value around the neighborhood of an individual is called epistatitc fitness inspired by genetic epistatics. Whitacre [101] introduced SOTopolEA (self organizing topology EA) in 2008 (Alg. 5) and typical network structures are in Fig.8.

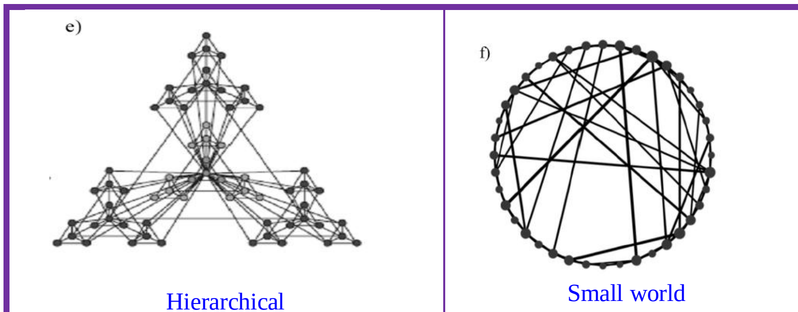


Fig. 8. Interaction networks. Number of connections/nodes (= neighborhood) decrease form panmietic to cellar through island models [courtesy from Ref. 101]

Growing Cell structures

SOM (Kohonen) has a fixed number (user chosen) of neurons in 1D-, 2D- or 3D- architectures. The neighborhood (diamond, hexagonal etc) shape is also user chosen. But, incremental NNs grow as they learn. Fritzke [20-21, 42, 190, 219] introduced growing cell structures with varying topologies. Some of the recent reported categories in GCS belong to internal, external and both mechanisms show internally growing architectures by inserting a node within the existing topology. As a result, the shape and size of the structure is increased [170]. The patterns with higher pdf in the output space are represented by more elements in the GCS output space.

Architecture.GCS: A simplex of k-dimensions (straight line for k =1, triangle for k = 2, tetrahedron for k = 3, hyper tetrahedron for k > 3) space is used as the initial topological structure of GCS.

As learning (self organization) by competitive delta rule [190] proceeds, new cells are added to take into account of novel/new trends. The midpoint of the edge connecting maximum resource vertex and the most distant node in the topological neighborhood is calculated. The superfluous/redundant neurons are deleted. After each modification, the network consists of k-D simplex. Each neuron has an n-dimensional vector denoting the position of the cell in the input space. The refinement of Ws is same as that in Kohonen NN.

Alg. 05: Self-organizing topology evolutionary algorithm (SOTopolEA) or cellular GA	**Reproduction.rule.SOTEA and cellar GA**

Initialization : population
individuals connected in a ring structure
 	
DO Until max.generations or convergence
  For i=1 to M
    Random selection of an individual i
    Offspring generation through mutation
    Application of reproduction rule
  End for
 	
  For i=1 to M
    Random selection of ith individual
    worst neighbor selected
    worse of i eliminated
    Links of loser to winner assigned
  End for
End DO
	Addition of a new (offspring) neuron to the network
	Offspring and parent are linked
	IF SOTEA
	  Offspring inherits parent links with 0.1 probability
	  Parent looses links with 0.1 probability
	EndIf
	
	IF cellular GA
	  Offspring inherits one of parent's links
	  Parent looses inherited link
	EndIf
	
	Competition. rule
	Individual selected randomly from parent and offspring populations
	Selected individual compared with it's least epistic fit neighbor
	Better individual inherits all links from worse

	individual Worse individual is removed from population (corresponding neuron is removed from the network)

Typical extensions reported to Growing cell structures (GCS) are Growing neural gas (CNG), dynamic cell structure (DCS), hierarchical growing SOMs (GH.SOMS), probabilistic growing cell structure [222], growing RBF-NNs, growing multi-dimensional SOM , tree Growing cell structures (Tree-GCS) [219], recursive SOM (Recurs.SOM) [207,230] and hyperbolic SOM (Hyperbolic.SOM).

4.3 Externally cell growing structure: The visualization of high dimensional structure in incremental grid growing NNs is the basis for externally growing cell structure. If maximum resource/maximum error vertex is a boundary node, then a new cell is grown externally. The algorithm is tested with classification tasks viz. two spirals, mines versus rocks, chemical sensors, brands of coffee and mixtures of organic compounds like toluene, octane and propanol.

Learning of growing SOM: Generally, a pattern with missing label or feature is to be deleted from the dataset. A semi-supervised learning method for growing self-organizing-map (grow-SOM), the advantage being that it trained with up to 60% missing class labels and 25% of feature data. The unique feature is that prediction accuracy is over 90% even two spirals, IRIS and breast cancer datasets. It is compared with semi-supervised k-means algorithm and its variants. It affords fast visualization of classes on 2D-feature map.

Dataset.classification.EGCS: The metal oxide chemical gas sensors are used in the analysis of sonar-mine/rock separation task. Externally GCS was found better than supervised-GCS and MLP (Table. 1).

Table 1 : Comparison of performance of Externally GCS with other NNs for chemical gas sensor data

Algorithm	Training		Test
	MSE	% CR	% CR
KNN	--	--	82.7
MLP-BP	--	--	90.4
Supervised GCS	0.224	93.3	90.4
Externally GCS	0.044	100.0	93.3

CR : Classification rate

Dataset.classification.EGCS: Seven coffee brands available in German market are analysed with 16 sensors. The data consists of 16 inputs, 7 outputs and 42 samples. Externally GCS performs better than supervised GCS (table 2).

Dataset.classification.two spiral.EGCS: Two spirals coil three times around origin and one another. Using 184 training data lying on the spirals, the performance follows the order
EGCS2(85) >EGCS1 (104) >[DCS-GCS (177) = SGCS(180)] >> [QuickProp (7900), BP(1100)], where the number within parentheses correspond to the number of epochs for training.

Table 2: Comparison of performance of Externally GCS with other NNs for classification of coffee brands

	SGCS	EGCS
CPU time (sec)	0.31	0.25
Training SSE	2.85	2.20
Training CR%	100	100
Testing	82.52	86.75
Growing cells	17	24

Modified growing SOM: It is successfully applied to travelling sales man problem (TSP) with 442 cities. The limitation is that a node is invoked even when one or two points with high error are in the training set. The remedy for this catastrophic allocation of new node is a modification of cell structure. It balances stability and plasticity dynamically. If the local error (for even more than two consecutive points) exceeds a preset threshold, the points are not considered in the model, but will be shown as outliers.

GCS

No a priori user defined network topology

optimum network structure is automatically generated

No need to define a decay schedule, which is essential in neighborhood learning

All parameters of the model are constant

There is no decay learning schedule as in SOM

4.4 Evolving-Tree-SOM: Pakkanen [196] proposed Evolving-Tree-SOM-NN in 2004. It is a freely growing network. The shortest path between two nodes in a tree is used as the neighborhood function for self organization process.

Evolving-Tree-SOM
- Visualization remains a demanding task
- C P U intensive for large SOM training of voluminous data sets

Remedy : Tree structured SOM

4.5 Growing Hierarchical SOM (Grow.Hierarch.SOM): The limitations of SOM and growing SOM paved way to the development of Growing Hierarchical SOM. There are two types of NNs under this category. The first one uses growing grid for map growth and hierarchical SOM for hierarchical growth. It is used in the analysis of CIA-world FACT book, legal documents or news articles. It uses label-SOM to assign topical descriptors to each of the neurons and efficient W initialization method. The other type is called Tree-growing cell structures. It starts with GCS structure to grow, but weights until 90% of the maximum number of nodes permitted is reached. Then the neurons are deleted to improve the stability of resulting dendrogram. The tree is created from the formation of sub-structure as cells are deleted from the structure. This method is expensive ($O(n^3)$ complexity) and apply only on very small datasets.

Growing SOM
- It is difficult to visualize all the data on a single map
- Training is very long

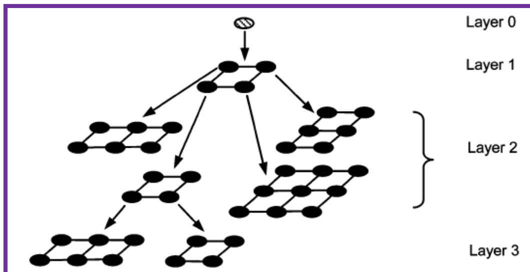


Fig. 9(a). Architecture of trained GHSOM

Alg 6b: Growth in depth of Grow.Hierarch.SOM

Train the data at zero level of GHSOM
While depth < max depth level
    For i=1:no of neurons
        If QEi > τ_2 * QE0, then expansion = true
    End
    If expansion, add a new SOM in the next level
    Train input
End while

Alg. 6: Grow.Hierarch.SOM

Initialization
W with random values
The error for each neuron is set to zero

Repeat until convergence
    Training with Standard SOM
    For each input vector
        Calculate quantization error of corresponding winner
        Update winner's error variable by adding qe to E_i
    End for

    Identify error neuron with highest e_i
    Growth
    Insert a row or column between error unit and its most dissimilar neighboring unit in terms of input space
    If measured qe < threshold, then converged
End repeat

Growing hierarchical SOM
- Better topology with best match to data
- clusters document items in hierarchical manner
- Combines virtues of SOM and hyperbolic space for adaptive data visualization

Architecture.Growing Hierarchical SOM: At layer zero, a single unit SOM serves as a representation of complete data set. In the first layer, down in the hierarchy, a single unit (2 x 2) SOM represents the complete dataset represented at layer zero (Fig. 9 a). It means details of the dataset are self organized into four sub regions. In the second layer, for every unit of the first layer, a separate SOM with increasing size is developed. The growth in depth is done by increasing the levels of hierarchy (Alg. 6). In the case of growth of width, the number of neurons is increased stepwise. This help in each neuron not representing too many patterns. In the case of growth of depth the philosophy is to form a new map in the subsequent layer for units representing a set of very diverse set of input vectors. The data flow in GHSOM is shown in Fig 9b.

- Exponential growth of hyperbolic lattice
- A more complex map dependent on the data
- scaling problem due to very large number of nodes
  - Remedy : Hierarchically growing hyperbolic SOM

Yen [114] analysed textual abstracts concerned with animals, anthrax, and SOMs with growing hierarchical SOM, after transforming document space into multi-dimensional vector space. The trained NN results are projected with ranked centroid projection method whereby the input vectors are projected to a hierarchy of 2D-output maps.

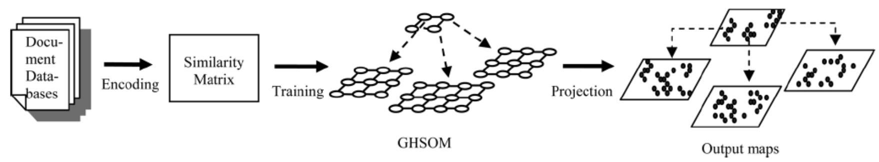


Fig. 9(b). Data flow in GHSOM [courtesy from Ref. 114]

Dataset.zoo.Grow.Hierarch.SOM: It is a simulated zoo data comprising of 100 patterns of animals with 16 features. The number of classes is seven. The clusters of standard-SOM (9 x 9) are somewhat identifiable, they are not well separated. The output of first layer of Grow.Hierarch.SOM (2 x 2) results in a clearer distinction of clusters. PCA and Shannon's mapping failed to capture present cluster structure.

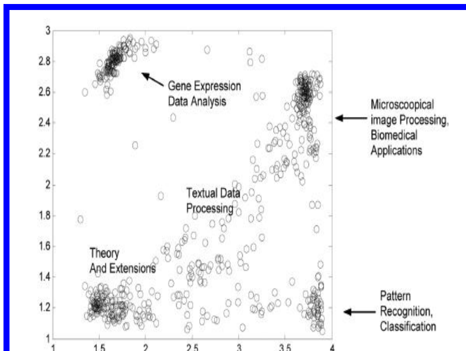


Fig 10(a): SOM display of papers published in Journals; circle: document

Dataset.clusters.Hierarch.SOM: Three Gaussian clusters (centers [0,0,0],[3,3,3] and [9,0,0]) each of 300 data points are simulated with variance one. Hierarchical.SOM clearly distinguished the clusters.

Dataset.literature.Grow.Hierarch.SOM: The published literature in ISI (Institutional science indicators) using the key word (SOM-s) [114] during the period 1990 to early 2005 resulted in 1349 documents. After eliminating irrelevant papers 638 remained. The first layer map consists of 3 x 4 neurons. The position of all documents on the first layer map is given in Fig. 10. Using citation count, it is found that the largely cited papers by 'Tonoren 1999' and 'Tamayo 1999' appear in Fig. 7-70--16. A more detailed map (Fig. 7-71---17) shows Kohonen contributions.

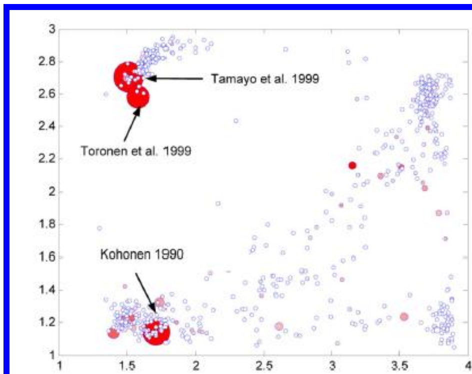


Fig 10(b): Browsing a section of Fig 10(a): size of circle : Number of times a document was referred

Dataset.anthrox.Grow.Hierarch.SOM: Yen [114] used 987 papers on anthrox covered by ISI web of science during the year 1981 to end of 2001.

A three layer Grow.Hierarch.SOM (Fig. 11) has generated with a threshold values of τ = 0.78 and τ= 0.004. The first layer consists of 3 x 4 neurons and the papers are distributed into broad topics using the number of citations. More details are obtained like seminal contributions. The display corresponding to one node in the second layer describes 192 documents. This hierarchical view is in consistent with probing from general to more specific.

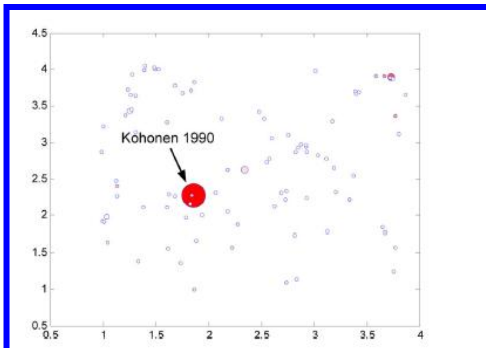


Fig. 10(c): Magnifying a section of Fig 10(b): Red circle : Kohonen seminal contribution [courtesy of Ref 114]

4.6 Hierarchically growing hyperbolic SOM (H2 SOM) : H2 SOM It is introduced by Ontrup [196] and is a good combination of several features: hierarchical data organization, adaptive growing to a required granularity, good scaling behavior, smooth trend and map based browsing. It embeds a complete hierarchy within a continuous browsable space. It is an extension of hyperbolic SOM. A hyperbolic lattice structure is built incrementally. Another critical feature is to search only a small fraction of all existing nodes to identify a close-to-optimal match. It is a alternative computational tool to standard SOM and hierarchical SOM. It allows more flexible growing of nodes and thus is similar to coding tree in classification. It does not form regular SOM layers as the tree search SOM. Hedge (2004) applied an information theoretical approach to VQ, of course with a neighborhood learning [197].

Architecture.H^2 SOM: It has the same lattice structure as that of Hyperblic-SOM. The root neuron of the hierarchy is placed at the horizon of H^2. Starting with two neurons in the first sub-hierarchy, the neurons are placed at the vertices of three equilateral triangles Fig.12. These nodes must cover the full circle in H^2.

Growing step in H^2-SOM: Each node in the periphery is expanded with nb=3 children neurons (Fig 12a). It is affected mathematically by applying Mobius transformation. The expanded node now resides in the center (Alg. 7). The neuron 7 is expanded Fig 12b. It has already one parent neuron and two siblings and thus there are five additional neurons. Fig 12c shows the expansion of NN for other neurons in the first sub-hierarchy.

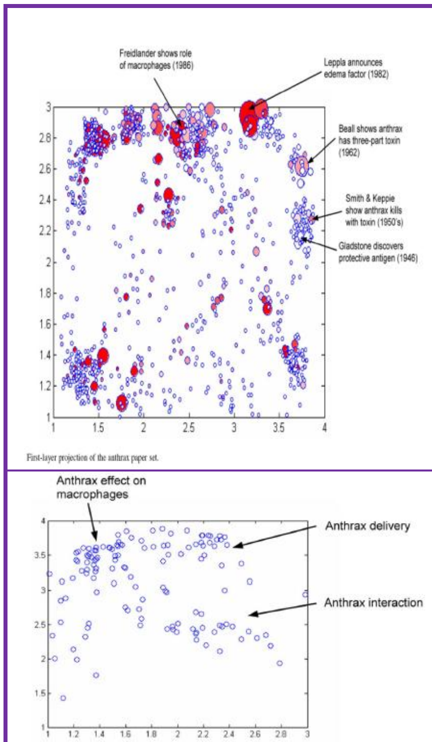


Fig. 11 Research papers on anthrax [courtesy of Ref 114]

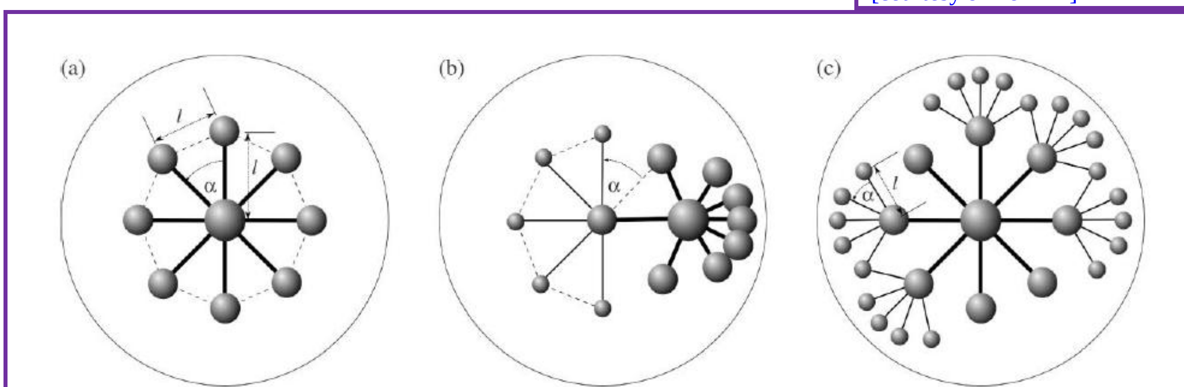


Fig. 12(a) Topology of the HH-SOM : (a) Eight nodes ; (b) A node expanded with 5 (= nodes-3) children (c) Grown up size of HH-SOM after iterative expansion [courtesy of Ref 196]

Fig. 12(d) shows a "wrinkled" structure resembling a saddle at every point of the surface. In nature, some corals require maximum contact area for their survival with the surrounding water which carries vital nutrients. It is spectacular that the growth behavior in these coral is like a hyperbolic surface. The human cerebral cortex is of 2-4 mm thin and nature optimized it into a corrugated structure of minimum area in commensurate with skull. But, its area is 2500 cm^2 if stretched flat. During browsing, a discrete jump results in loss of context in the surrounding topics. H^2SOM is superior to Tree-Structured SOM (TS-SOM), the Hierarchical SOM [105], the Self-Organizing Tree Algorithm (SOTA) [43], the Adaptive Topological Tree Structure (ATTS) [180] or the Evolving Tree by Pakkanen et al. [221] (2004) and provided a continuous smooth browsable space. A framework using the open source visualization library VTK1 is developed which displays a 3D scene with user interaction.

Alg. 7: H^2-SOM

Input data
z_a : 2D-position of neuron in the complex Poincare Disk
Initialize center node with center of mass of training data
Weight of the node (W) is projected into the data space 1
[It is not refined in the entire training cycle training cycle]
Initialize first hierarchy neurons with small deviation from W of center

```
For i=1:max_hierarchy
    Do until it = maxit
        Train neurons in first sub-hierarchy
        Upgrade w_a
        Cal hyperbolic neuron distance from their position
        Decrease width and learning step size
    End do
    Calculate quantization error
    If error > threshold, there is a need for growth
    If growth, then expand the architecture
    If   all nodes satisfy the growth criteria
    Then fix weight vectors from previous hierarchy
        Adjust Ws of the neurons at the new hierarchy level
    End if
End for i
```

***Fig. 12(d) Display of local embedding of H^2 in R^3. Similar to natural coral and* human cerebral cortex**
[courtesy of Ref 196]

The data sets of handwritten digits and news-wise (Reuters-21578) articles are used to quantify the efficiency of the method and to affect the classification and visualization. These, two data sets are of high dimensional ones. This method achieves better topology preservation and lower quantization error compared to other similar sized SOMs. The computational complexity is O(log N).

Dataset.digits.H^2-SOM: MNIST contains 600,000 handwritten digits (for training) written by 250 writers and 10,000 text samples from a disjoint set of 250 other writers. The original 784-dimensional datasets, resembles 28 x28 pixels of grey level images of the digit. H^2-SOM (Fig. 12c) with a branching factor of 8 and 2,3,4,5 or 6 rings with maximal 41, 161, 609, 2281, 8521 neurons are trained each with six lakhs of steps. The average of 10 runs except for 6 rings is superior to standard SOMs of sizes 7 x7, 13 x 13, 25 x 25, 48 x 48 with 49, 169, 625, 2304 and 8521 neurons respectively. The termination criterion is combination of maximum depth and quantisation error. H^2-SOM with 2281 neurons is 180 times faster than SOM with 2304 neurons.

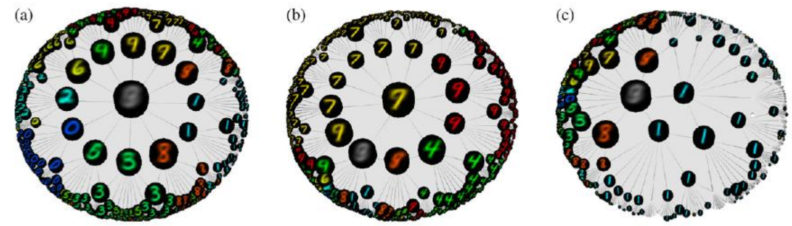


Fig. 12(c). MNIST database (a) coarse structure (b) focus point of 7-node from 1 o'clock position of (a); (c) perspective view covering '1'

Dataset.Reuters 21578.H^2-SOM: It contains neurowire articles from 1984 onwards. It is a benchmark in text mining applications. The training set contains 9603 items (Fig. 12d). The text data set has 3299 documents. The number of distinct words is 5093 after preprocessing (word stemming) and deletion of stop words. H^2-SOM with a maximum depth of five rings is superior to standard SOM of 48 x 48 topology and is approximately 60 times faster.

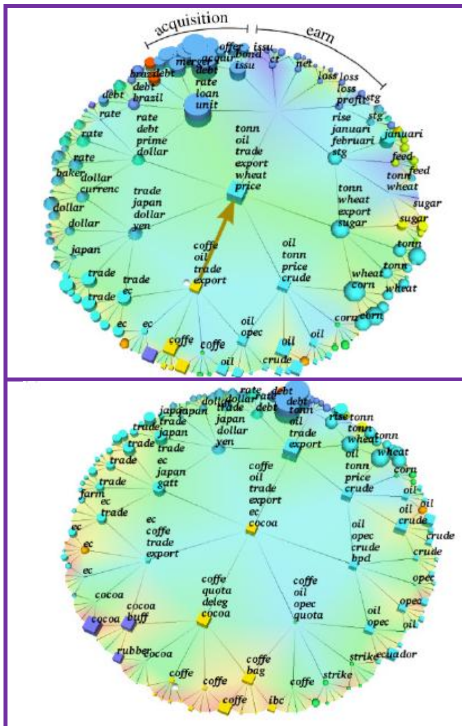


Fig 12(d) : Reuters-21,578 collection model using H^2SOM [courtesy of Ref 196]

4.7 Spherical SOM [204]: *Tesselation*

Each triangular phase of the polyhedron is sub divided into several smaller triangles by lines running parallel to the original edges of the triangle.

Icosahedron is most similar to a sphere. It is clear that variance in edge length is smallest after tessellation. Most of the vertices have six immediate neighbors. On the other hand, the original twelve vertices of icoshedron have 5 immediate neighbors. The number of vertices (N) after tesselation are $N = f^2 *10 + 2$. Thus, icosahedron based geodesic domes are more suitable for spherical SOMs.

The frequency (f) means the number of parts into which the original edges are divided. In the case of polyhedra the faces are not triangles. For a cube and docecahedron their faces are to be triangulated first. A hexagonal lattice has better geometric environment compared to a rectangular one in 2D-space. Every grid unit has the same number of immediate neighbors. Further, the distances between a unit and its immediate are the same. In the case of a sphere, this type of uniformity is achievable only for five platonic polyhedrons viz. tetrahedron, cube, octahedron, icosahedron and dodecahedron. Figure 13 depicts tessellation of a triangle and a icosahedron with 1 to 4 frequencies. In the case of a triangle, the number of triangles is equal to

Spherical-SOM

+ It removes the border effect of 2D-SOM and thus reduces data distortion
+ Gives more information about high-dimensional data

— Neighborhood searching with existing data structures
  - not space efficient
  - time consuming

Remedy : GEO-SOM

square of number of frequencies. These polyhedra can be further tessellated into different frequencies of the geodesic domes. Many types of spherical SOMs are developed and applied to different types of datasets. The tesselaed platonic polyhedron was proposed as the lattice. Sangole [68] used 3D-immersive vertical reality environments for interactive data analysis. The spherical SOM was used in 3D-object modelling.

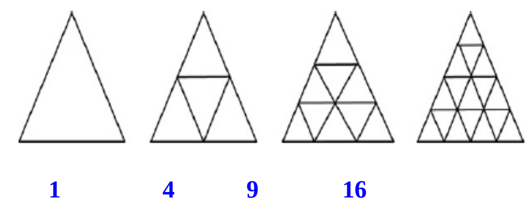


Fig 13(a). Tessellation; 1- to 4-frequencies in a triangle

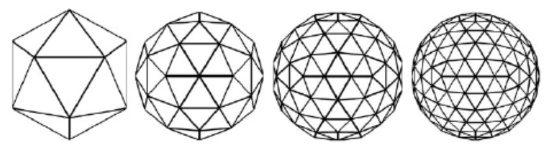
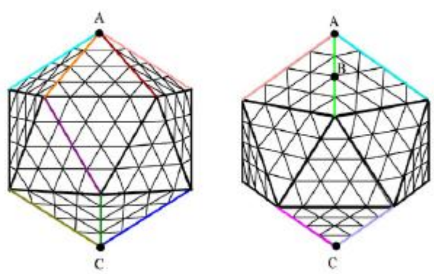
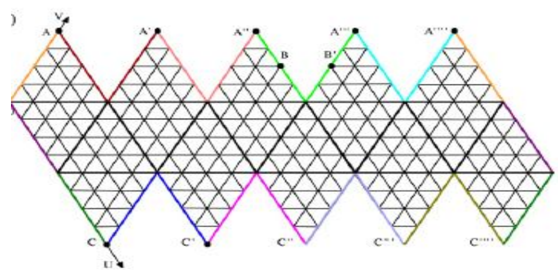


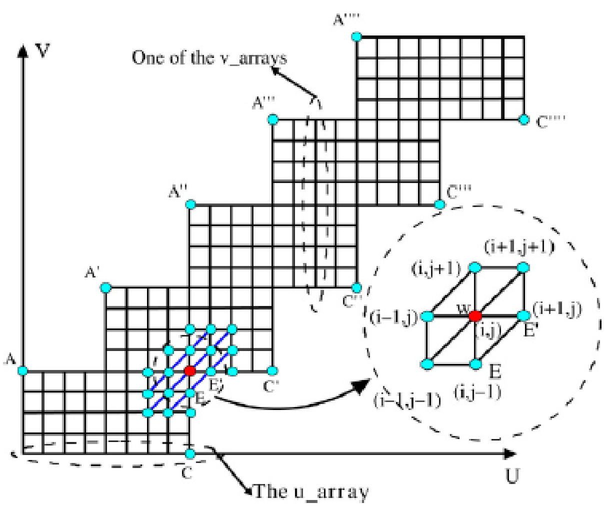
Fig 13(b). Tessellation; 1- to 4-frequencies in a Icosahedron



(c)



(d)



(e)

Fig. 13(c) Front and back views of icosahedron. The dome is cut open along the colored edges
Fig. 13(d) Four frequency geodesic dome when cut opened
Fig. 13(e) Data structure in two dimensions for geodesic domes based on icosahedrons [courtesy of Ref 204]

Geo-SOM

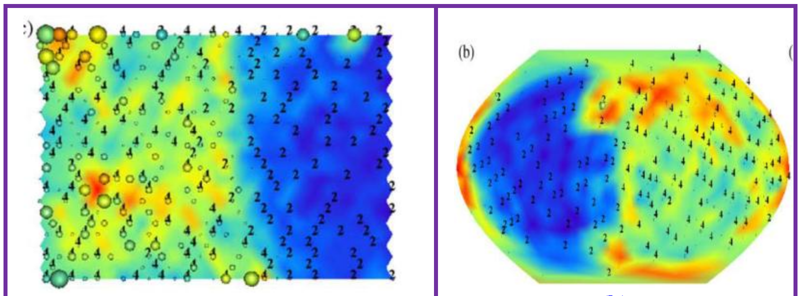
- Reduction in overheads in spherical SOM
- Efficient method to find immediate neighbor of a

vertex (neuron)

+ Fast dome tessellation i.e. increasing the number of neurons

4.8 GEO-SOM: Wu and Takatsuka reported GEO-SOM [204], an improved version of spherical SOM. It is a spherical SOM using 2D- data structure. The border effect standard 2D-SOM is over come in Spherical SOM. But, existing data structures of geodesic domes are not space efficient or time consuming when searching neighborhood. Wu introduced 2D-rectangular grid data structures to store the icosahedron based geodesic dome. The number of neurons can be efficiently increased.

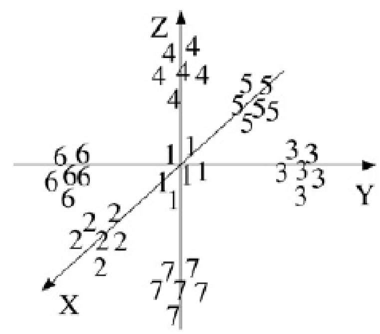
Dataset.breast cancer.Geo-SOM [204]: The benign samples are labelled as 2 and cancerous ones as 4. Geo-SOM used 8-frequency geodesic dome (642 neurons) and RDSOM 28 x 23 hexagonal grids (644 neurons) with a initial update radius of 11. After 150 epochs the sizes are distortion spheres are more uniform and smaller for Geo-SOM compared to 2D-SOM (Fig. 14).



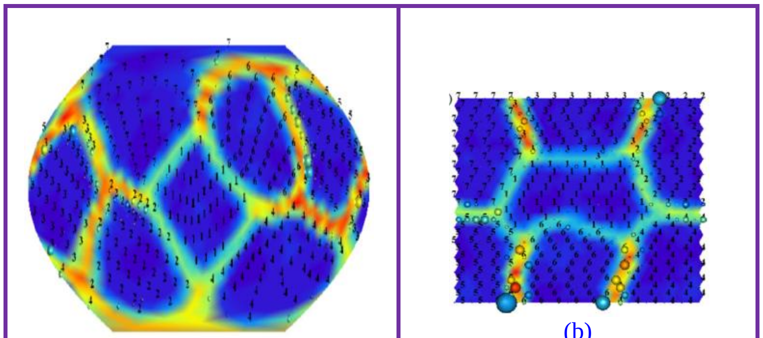
(a) (b)

Fig. 14 Discrimination of benign ('2') versus cancerous ('4') breast biopsy samples
(a) Display of trained ordinary-2D-SOM (b) Projection of trained Geo-SOM on to 2D-plane

Dataset.sevenCluster.simulation.Geo-SOM: Seven clusters each with 500 data points in 3D-space are simulated and analyzed with Geo-SOM (Fig. 15) with 9-frequency geodesic dome (812 neurons) and 2D-SOM 29 x 28 hexagonal grid (812 neurons). The initial update radius is 14. In the input space cluster 5 is close to 4, 3, 1 and next level 6, 2, 7. In 2D-SOM 5 is closer to 7, 1. 6 and 3, 2, 4 are in the next level. In Geo-SOM 6 and 4 are in the first level, 7 and 1 in the second level and 3, 2 in the first level. Distortion around each neuron is larger along the boundaries of clusters in both the methods.



Cluster centers :
(0, 0, 0), (10, 0, 0), (0, 10, 0), (0, 0, 10), (−10, 0, 0), (0,−10, 0), (0, 0,−10)
SD = 1.0 in each dim



(a) (b)

Fig. 15 visual separation of clusters in (a)Nine-frequency-Geo_SOM (812 neurons) (b) 29 x 28 (= 812) neuron-2D-hexagonal-SOM for a simulated data set [[courtesy of Ref 204]**]**

Misceleneous

4.9 Rival-model penalized self organizing map (RPSOM)

The rival penalized competitive learning (RPCL) and rival penalization controlled competitive learning (RPCCL) methods have been used in cluster analysis. RPSOM [121] is based on these postulates and the algorithm is brief in Alg. 8.

Alg. 8: RPSOM [121]
Initialize W for m x n Kohonen map
The winning frequency of each neuron set to 1
Repeat until convergence of SOM
For i = 1:NP

Input x(i) pattern
Find k-nearest neurons (k =1, 2) in the adopted neighborhood topology
Find Best matching unit (BMU) is found

Increment winning frequency neuron by 1
Identify Rival neurons belonging to first k-NN and not 1-neighborhood neuron
Update Ws of BMU and its neighbors
Redefine Neighborhood function of BMU
Penalize rivals in W vector
End for
End repeat

Datasets: RPSOM is tested with two synthetic data and IRIS.

4.10 Gray-SOM: Yeh and Chang [228] proposed Gray-SOM (Alg. 9) considering the gray relation between the input data and each adjustable output node in the learning rule. It considers the input training data and all adjustable weights as n-tuple sequences, and not as ‘‘n-dimensional patterns’’.

Dataset.TSP.Gray-SOM:

The distance covered in TSP task using Gray-SOM-NN is nearer to optimal length compared to G-SOM and SOM (Table 3).

Table 3: Comparison of performance of Gray-SOM with GSOM and SOM for TSP

Datasets	# cities	Optimal length	GraySOM	GSOM	SOM
EIL51	51	426	436.90	437.71	443.90
ST70	70	675	683.12	629.06	692.80
RD100	100	7910	8119.00	8143.72	8137.90
EIL101	101	629	653.78	658.04	688.70
BIER127	127	11828	120790.00	121181.30	122211.70

Alg. 9: Gray-SOM
Initialization
W
parameters
Do until connection weight vectors converge
For i=1:NP
ith training pattern is inputted to the NN
Calculate Euclidean distance between W(i) and x(
Determine the winning neuron
Refine W
Determine neighboring nodes around winning neuron
Select output nodes which are highly related to output
Update corresponding Ws
End for
Increase the threshold,
shrink learning rate & t size
End do

If $\left\| w_i \quad t+1 \; - w_i \quad t \; \right\| < tol$
Then converged

4.11 Concept- SOM: ConSOM (Alg.10) is more sensitive to semantics and the quality of clusters and is superior to SOM and 'SOM plus VSM'. Table 4 summarizes documents of different categories each containing 160 records analysed.

The architecture of concept-SOM is similar to Kohen SOM except that each neuron in Kohonen layer has two vectors corresponding to concept and feature. Each input sample also has tow vectors. Liu [224] proposed conceptual (concept-)

Table 4: Datasets analysed with ConSOM

	Description of Data set	#FV	#CV	Source
A	Wheat, grain, ship, trade	494	723	Reuters 21578
B	Corn, wheat, grain, ship	441	652	Reuters 21578
C	Space, Auto, guns, medicine	612	716	20 newsgroups
D	Space, baseball, Christian, medicine, education	568	629	20 newsgroups
E	martial, traffic, computer, politics	575	791	http://news.sina.com
F	economics, culture, martial	742	915	http://news.sina.com
#FV	Feature vector dimension			
#CV	Concept vector dimension			

SOM for clustering of text documents. The documents are represented in the feature space and neurons in an extended concept space. The similarities are calculated in both the spaces and are used to update the weights. The frequency of occurrence of a word plays a role.

This model has the benefits of the knowledge of relevance of concept of SOM. A common sense data/knowledge base named 'Hownet' with concepts/words relating the words with defense, soldiers and doctors are used. For example, the word doctor does not distinguish a civilian or defense professional. But, soldier unambiguously means a defense personal that too belongs to infantry and not to navy/air force. Only the combination of both the words viz. doctor and defense correspond to a medical practitioner in the warfront (may be in infantry/navy/air force). Still the ambiguity lies whether he is in the war field with defense-operation or in defense-hospital amidst civilian habitat.

Alg. 10: Concept extension SOM
[Liu 2008]
Input : Document
Parse the document
Delete stop/grammatical words
Count the frequency of each word
Pick up the most frequent words into a vector S
While S is not empty
Pop a word
Find the word in the knowledgebase (HOWNET) and get sense word S(wi)
For every sense record
    Find all words relevant to concept
End for
Find the common words (intersection) between the document and sense words
End while
Count the word frequency
Output : concept word vector

4.12 Self organizing relationships (SO.Relation)-NN: Koga [199] proposed SO.relationships-NN to approximate I/O relations extending the domain of Kohonen-SOM and learning is through a critic.

Architecture.SOR-NN: The input consists of x and y vectors. The SO layer is same as that in Kohonen-NN. The functioning involves two stages-learning and testing (execution). SOR learns the data relationship in the first phase.A reference vector is a paired real values representing the weight of x to Kohonen neuron and Wy to the same Kohonen neuron and aht algorithm is in Alg.11.

Data structure: The paired data vectors of explanatory variables and response are input to 2D-SOM layer. The evaluation value Ei is the user chosen or intuition based. The learning is attractive or repulsive depending upon whether Ei is positive or negative.

Alg. 11: SOR. learning

X and Wy are initialized by random numbers
Cal similarity measure between given input vector and all reference vectors in the input space
Repeat until convergence
For i=1:NP
Cal best matching unit for ith learning vector
Cal Gaussian neighborhood function
Refine values of each reference vector
Cal parameters
End for
End repeat

Cal output of network, which is the weighted average of Wyi and zi
Cal Nth element of y

Learning: Self organizing relationship SOM learns from undesirable behavior leading to undesirable I/O relationships. Repulsive learning is similar to reinforced learning prevalent in animal kingdom. The objective is to realize an approximation of a desirable I/O relation and mainly used in on-line learning. Undesirable I/O relationships are obtained by trial and error. They are actively used in repulsive learning, which is similar to reinforced learning. Reinforced learning is a search based algorithm, but requires a large number of trials.

Dataset.trailer_truck.SOrelationships: The trailer truck control system has three inputs and one output with non-linear relationship. In the experimental system, the motion is captured with two CCD cameras in the form co-ordinates of three markers attached to the trailer truck. The front wheel angle is calculated from two angles and the distance. The training set consists of 6561 learning vectors and 25 x 25 Kohonen layer is used. Starting with any position, the truck reaches the target with SOR-NN.

SOM with higher order neurons: In this type of NN, higher neurons are used instead of conventional neurons. The detection of chromosomes in the human cell is modelled with four third SOM.

Clustering discrete group of data : Ghaseminezhad and Karami [38] proposed a modified SOM for automatic clustering of discrete groups of data. It starts with a "second winner" algorithm where neurons in the competitive layer find their initial location in the network space. It is followed by batch learning to train SOM. Now, the wrong links between neurons are removed. The method is effective for real and synthetic datasets.

SOM with supposed maximum information: Kamimura [249] introduced a new SOM with maximum information content and tested with animal data, SPECT heart diseases, voting attitude tasks. The limitations and possibilities are discussed

Hybrid SOM-NNs

Hybridisation SOM with another neural network, statistical procedure, fuzzy method, evolutionary algorithm continues to be an active area to enhance the beneficial features, diminish limitations and increasing application scope in inter-/intra- disciplinary tasks. A brief description of binary hybridisation of SOM with RBF, immune algorithm, statistical concepts and SCL widened the scope this novel self organizing (2D-, 3D-) visualization platform from high dimensional feature space. The hybrid algorithms are far superior to simple SOM for very large text documents in categorization.

Fuzzy theory + SOM

Fuzzy Kohonen Clustering Network combines fuzzy membership jargon with values for learning rates. It processes. It processes data sets or images with ambiguity and/or uncertainty.

4.13 FuzzyNN + [GA, PSO] + SOM: A self-organizing-Fuzzy-NN based on GA and PSO was reported. In the first phase, fuzzy structure is identified using Takagi-Sugano (TS) fuzzy model tuning. Optimal number of clusters is obtained from fuzzy-cluster validity index. The second phase involves fine-tuning of parameter set of the fuzzy-model from first phase with GA and PSO. Static function approximation and non-linear dynamic system identification data sets are trained with SOM-Fuzzy-(GA-SOM)-NN.

4.14 Self organizing-adaptive-fuzzy-NN: Hsu **[117]** reported self organizing, adaptive fuzzy neural control for online estimation of controlled system dynamics of electro-chaotic circuit. It consists of computational and supervisory controllers. The structure and W learning phases of fuzzy NN are used in computation control.

The optimum structure learning includes on-line generation [223] and elimination of fuzzy rules (Alg. 13). This method automates structure and parameter optimisation simultaneously based on input and target values. The first phase is SOM operation in arriving at network structure. It is followed by a supervised approach and applied to a simulated data of function approximation. L2 norm with a desired attenuation level is the objective to be achieved for good performance. Lyapunov function is the basis of W learning ensuring system stability.

4.15 Granular SOM: Kaburlasos [193] proposed a distribution of fuzzy interval numbers for the data in his Granular SOM. Lattice theory is the basis for rigorous mathematical analysis of Granular SOM. It aims at fuzzy rule induction for linguistic classification data. Visualization is not the objective here. Fuzzy interval numbers (FINs) represent a local non-parametric PDFs and/or a fuzzy set. The parametric mass functions are to introduce tunable non-linearities. There is one-to-one

Alg.13: Self Organizing fuzzy-NN algorithm [223]

```
Lo samples are randomly picked up whose coordinates are set to cluster centers
it= 1
While it <maxit
        For i= 1 : NP
            k= Random number in the range [1:NP]
            z= x(k)
            Cal distance matrix
        end
        Winning (win) and rival (rival) neuron calculation
```

$$d(z,cwin)=\min_{k} d(z,ck)$$

$$d(z,crival)=\min_{k\neq win} d(z,ck)$$

```
        Up gradation of Ws

        it= it+1
Endwhile
For each sample
    find the nearest cluster center ck
    near_cluster_center = (k)
endfor
Compute the ratios between the number of
samples in each cluster and the number of total samples
If    ratio of some cluster is smaller than the threshold x,
Then delete the corresponding cluster.
Nclust = +1
If nclust == 2, then stop, otherwise continue
```

correspondence between FINs and PDFs. The interpretation of FINs is that they are the antecedents or IF part of the fuzzy rule. The category (classification) label is THEN (consequent) part of the rule. This model seeks optimization looking for a different mass function in a data dimension. GA is used [193] to compute optimal mass functions for tuning a metric distance between non-parametric fuzzy inference numbers. This algorithm uses calculated FINs and Minkowski metric in F^n. It extracts descriptive decision making knowledge from training data. It reads the Euclidian space R^n as the Cartesian product of N totally ordered lattices R. Thus, it adheres to linguistic semantics. In order words, difficult quantities (weight, speed) are involved in different dimensions. Gran-SOM requires batch process to refine W belonging to F^n.

median-SOM

Tackles classification where Euclidean distance is not available

protein structure, text documents, biological signals

Future venture: An incremental Gran.SOM using convex combinations of FINs is contemplated. But, it may leave part of the training data outside all fuzzy rule interval support. It is interesting to compare the function and behavior of Gran-SOM with probabilistic mixture models.

Greedy Granular SOM

optimization of well defined object function

Guarantees full coverage training data domain

It retains linguistic interpretation.

Captures locally all order statistics in the training data

Handling of missing data based on the theory of probability

Does not consider alternate divergence (distance) function

Cannot cope up with linguistic data

4.16 Greedy granular SOM: The term greedy refers to an increase in the number of components in the mixture models. Greedy-Gran-SOM [193] calculates a distribution of FINs. It induces non-parametric FINs for PR data leading to fuzzy data clusters.

4.17 Fuzzy ART-NN + growing cell SOM: A hybrid Fuzzy ART-NN with growing cell structure, resulting in growing-Fuzzy-Topology-ART-NN. The growing cell structure results in growing NN. In the present model a restriction on topology preserving is achieved. The training algorithm used is called push-pull learning method. The model is tested with synthetic and real time data sets. The categorization of pedestrian and car is obtained real traffic roads (KNU and MIT-CBCL databases). The five different objects in COIL-DB are successfully discriminated.

Growing-Cell-Structure-RBF

A categorization property of Fuzzy ART enhances the class dependent clustering representation of GCS

The proliferation of growing nodes in F2 layer is reduced . It is achieved by replacing each of F2 nodes with GCS

Push – pull training increases the discriminating power of clusters and partially improves, the forgetting problem

Auto resonance theory (ART) has niche as unsupervised paradigm for binary data with distinct learning process. ARTMAP is ART in the supervised mode using both X (explanatory/causative) and Y (response/effect) datasets. Fuzzy theory enables to deal with floating point data. The state-of-the-art-of this brainchild of Grossberg and Carpenter will be detailed elsewhere. The present model is combination of fuzzy-ARTMAP with Kohonen-SOM enveloping growing architectural advantages.

Mathematical space +SOM

4.18 Kohonen-SOM-Riemannian space: Peltonen [182] extended Kohonen-SOM to Riemannian (non-Euclidian) spaces (matrices) (Alg. 12). It is an FIS extension and processes linguistic fuzzy data using simplified 3D-vector representation of linguistic data.

Alg. 12: Kohonen-SOM-Riemannian space [193]

Step	:	1	Learning of centers of fuzzy sets by crisp-SOM
Step	:	2	Fuzzy sets with triangular mf is inserted followed by fine tuning
Step	:	3	Continuous valued output ← weighted average of output of activated rule

4.19 Turing unorganized machines + SOM: Turing unorganized machines consist of self organized connections as opposed to self organizing neurons in Kohonen SOM. Beaton et al [248] proposed a hybrid SOM with Turing unorganized machines with both self organizing neurons and connections through a connection learning rate, connection reorganization, and a neuron responsibility radius. Hybrid model envisaged both self organizing neurons and connections through a connection learning rate, connection reorganization, and a neuron responsibility radius. It is implemented in a 1-dimensional network (with. chain of neurons) and theoretical implications are demonstrated. It is superior to the classical SOM algorithm in speed until convergence and produces independent clusters and tangle-free networks.

Statistics + SOM

4.20 Dis-similarity SOM or median-SOM: Kohonen proposed median SOM, where the mean value of the batch SOM is substituted by generalized median. Median-SOM uses k-means algorithm. It is very slow compared to standard SOM. It's time complexity is quadratic while that for standard SOM is linear. But, there is improved computational efficiency [203] over earlier DSOM. Cottrell et. al. [23, 46] derived a batch version for modified SOM, NG and k-means. The proof of convergence is derived and batch-NG is related to an optimization by Newton method [203].

Kohonen-SOM-Riemannian space

- Only triangular mfs used
- Accepts crisp but not Fuzzy inputs
- Constant mass function used implicitly and thus do not have any statistical interpretation
- importance of structure identification is not recognized

4.21 SO-mixture (density) network: Yin **[196]** formulated self organizing mixture NN wherein each node characterizes a conditional probability distribution. The joint probability density of data (or NN) is described by a mixture distribution. The proposed complimentary method [202] adds on statistical perspective to the non-statistical SOM. It helps in deeper analysis and interpretation. It is an instance of hybridizing information from paradigms of different philosophies.

If	SO-mixture NN and equal variance and equal priors for all nodes and number of nodes is large
Then	SOM approximates to a Kernel method i.e. SOM is a special case of Kernel method
If	Kernel SOM and prototype conditional density is used as kernel function
Then	Kernel SOM ≈ mixture density model
If	Data density is smooth and number of neurons
Then	SOM and Kernel SOM have similar performance for classification

The original Kohonen-SOM model was extended to incorporate an underlying probability distribution. Lopez proposed SOM based on mixture of multi-variate student-t components. The earlier popular Gaussian mixtures of PDFs are used. It is robust to outliers.

Architecture: A tree structure is proposed for SOM in 1990 [Ontrup 2006] and adaptive feature is added [196]. Later, it uses an evolving strategy. The growing hierarchical hyperbolic SOM is a hybrid product of growing hierarchical-SOM and hyperbolic SOM with tremendous applications. In twinned self organizing maps [231], two SOMs are linked via the method of winning neuron. The concept of granular approach resulted in granular and greedy granular SOM [193].

k-means + SOM: A hybrid SOM with k-means and modified leader clustering algorithms is tested on Reuters-21758v1.0 and 20 new screw collections. SOM with k-means is better than stand alone SOM, or its modification with leader algorithm.

Overlapping SOM: Cleuziou [250] proposed overlapping SOM, a hybrid algorithm with overlapping-variant-of-k_means and Heskes-variant-of-Kohonen SOM. It is superior to conventional SOM. The theoretical aspects of associated energy function and complexity of the algorithm are discussed. Ambroise [237] formulated probabilistic SOM.

Kernel SOM: The k-means clustering algorithm kernalised and a neighborhood learning is added [197]. The input is transformed into a feature space followed by application of non-linear kernel function. It resulted in the improved classification. Graepel et al. [240] transformed the input space into a high dimensional space using kernel function. Here, the distance metric is transformed into non-linear form which adds flexibility in VQ to capture the data structure. Yin [109] and Van Hulle [187] employed Gaussian or other kernel neurons. This approach is approximately equivalent to a mixture of Gaussian/kernel-distributions of the data. Here, Kullaback-Leibler divergence between the neural model and the data is minimized. Based on these results, Yin [197] established a formal link between Kernel

If	Conditional density function is kernel type or
	Kernel function is of density type and
	Both are isotropic or symmetric
Then	The two methods are equivalent

SOMs and SO-mixture networks [109]. SOM implicitly approximates the kernel methods. There is a connection between kernel approach and probabilistic model. It shows the superiority of function of kernel SOMs over standard SOMs. Kernel SOMs model data density better and thus improved classification. The data points and neuron weights (defined in input space) are mapped to a feature space. It is followed by the application of SOM in the mapped dot product space. It is termed as type II kernel SOM. Kernel SOM is entropy optimized mixture density learner. The core advantage is improved classification.

NNs + SOM

***4.22 RBF + GCSSOM*:** Fritzke [190] put forward a supervised hybrid GCS-RBF-NN. It is the start of a new paradigm crossing the boundaries of the layered structure entering into the realm of reality (brain).

Architecture.GCS-RBF: The hidden layer of SLP consists of Kohonen topology with hyper-tetrahedron neighborhood structure (Fig. 7.74). The activation function for the neurons of hidden layer is RBF. The output is the weighted sum of the output of neurons on the hidden layer (chart zzz.). For a classification task, the largest activation indicates the classification label. The insertion of the neurons is based on error/signal criteria. For example, the classification error at the current moment can be used to find the position of insertion of a new neuron. SOM-RBF is another novel NN over the long nurtured center detection algorithms of clusters. Hecht-Nielson [635] [1987] reported counter-propagation SOM [199], another supervised SOM-NN. It approximates a desired I/O relation of a target system.

Growing-Cell-Structure-RBF

+ Automatic determination of number of RB neurons, their width and center (position) in the growth process itself
+ parallel processing of position of RB (hidden) neurons and refinement of W
+ Good generalization
+ Size (or number of neurons) is relatively small compared to general RBF which requires a larger number of neurons

Nature Inspired alg +SOM

4.23 ImmuneAlg + SOM: A tree-structured artificial-immune network along with SOM was recently proposed. This hybrid SOM-immune-NN strictly generates topological structure as a tree. This permits the analysis of data hierarchically. The novel antibody interaction inspired from immune system and SOM maintains consistency between shape, space metric and topological metric. It is an important concept in high-dimensional data analysis. SOM-IA-NN is applied for IRIS and synthetic datasets with low VQ errors and promising data visualization.

4.24 EA + SOM : A memetic-NN is used for TSP using Euclidean distance. SOM is hybridized with EA. The evolutionary dynamics consists of intervening SOM execution with a mapping operator. Fitness evaluation and selection operators are also used. SOM and mapping operators have a similar structure based on closest point finding. Simple moves are performed in the plane. TSP up to 85,900 cities is solved. The performance for 91 datasets is publicly available. The approach is superior to other NNs. Yi proposed an extended elastic-NN to solve TSP by introducing time-dependent parameters. Here, neurons move quickly near to the cities during the first few epochs.

4.25 Ensembles of SOMs: SOMs, in general, provide visual output sacrificing as little as possible topology of the data. But, the limitation is artifacts of single training. The ensemble approach for SOMs corrects small defects arising as a result of single training. This method retains smoother representation of the inner structure of the datasets. However, it does not supersede in lowering classification/distortion of errors of single models. Yet, it fabricates the model with more truthful and organized representation of the data and trained SOM-ensembles outperform other learning methods. The inter relation between diversity and sub-local accuracy inside SOMs is possible due to transparency of these models. For visual summarization of the results of an ensemble of SOMs, a weighted voting super position fusion algorithm was recently applied. It performs a weighted voting process between the units of SOMs in the ensemble. The added advantage is the preservation of topology of the map. The results of analysis of IRIS, Echo-

Cardiogram and wine datasets are compared with other two algorithms viz. fusion_ED and fusion_Voronoi polygon similarity.

5. Research mode SOM : SOM, proposed by Kohonen in 1990s, as an unsupervised exploratory tool for 2D- visual display of multi-dimensional data without an apriori knowledge of data structure, probability distribution etc, arose interest in the development of newer procedures and extensive applications in diverse disciplines for numerical to symbolic data. The exhaustive comparison of all the components for a task is a formidable job and availability of the algorithms in software implementable mode with a white box approach of code is the need of the hour for research and pedagogic purposes. The state-of-the-art-of-SOM in the method-base mode is described in Chart 24.

Chart 24. State-of-art-of- Kohonen_SOM in research mode

SOM
Unsupervised SOM
Supervised SOM

Supervised SOM
Counter propagation
XY-fusion
Supervised Kohonen
LVQ
Supervised Neural gas

Training mode-SOM
Sequential
Batch
Parallel

Software packages
Matlab
Professional II
Trajan
….

Topology_SOM
Square
Hexagonal

Boundary condition_SOM
Normal
Toroidal

Weight initialisation
Random
fn(Eigen vectors)

Experimental Design
None
Factorial

Training algorithms
Hebbian
Conscience

Method Base_ SOM

Growing cell structures (GCS)
None
Externally CGS
Evolving-Tree-SOM
Growing Hierarchical SOM
Hierarchically growing hyperbolic SOM (H^2 SOM)
Spherical SOM
GEO-SOM

Evolution + SOM
None
Self evolving SOM
SO self evolving NN
SOM-EA
SO-topology evolution

Neurons
None
Higher order
Symbolic
MLP

Hybrid_ SOM

Mathematical space + SOM
Euclidian
Riemannian space

Statistics + SOM
None
Median

Nature inspired + SOM
None
Immune Alg

Mixture density

EA

Fuzzy theory + SOM
None
SO-adaptive-Fuzzy
Granular SOM
Greedy granular SOM
Fuzzy ART-NN + growing cell SOM
FuzzyNN + [GA, PSO] + SOM

Ensembles
None
Majority volte

Miscelleneous-SOM
None
Rival-model penalized self organizing map (RPSOM)
Self organizing relationships (SO.Relation)-NN
Gray-SOM
Concept-SOM
Rival-model penalized self organizing map (RPSOM)
Self organizing relationships (SO.Relation)-NN

Scientific vocabulary	Definition
MLP	Multi-layer perceptron
RBF	Radial basis function
Fuzzy-NN	Fuzzy-
NN	Neural network
VQ	Vector quantisation
SVM	Support vector machines
SOM	Self organizing method
LVQ	Learning VQ

Scientific vocabulary	Definition
VEDA	Visual exploratory data analysis
SXR	Structure X Relationships
X	[activity property Biodegradability]
QSXR	Quantitative SXR
ARMA	Autoregressive moving average
IIR	Infinite input Response
FIR	Finite input Response
XOR	Exclusive (Boolean) OR

6. Future scope : The future direction in architecture should be in emulating hitherto existing best types and even random (heuristic) intelligent combination of them with the choice of adequate (simple to complex) neurons depending upon the task. The chaotic to stable state concept can be the basis of the venture. The application end-user looks for tidbits in the results of problem on hand within the established frame, although he/she does not grasp or browse into the details. The software designed to display the status of results in the expert mode/critical analysis mode along with necessary conditions/limitations/remedial measures of method, data, error profiles, computational time/costs etc. In computational quantum chemistry, HF, post-HF, DFT etc reached a status of reliability and at least partial alternatives to experiments. John Pople, Nobel laureate in chemistry and a core mathematician proposed smart (called Gaussian [G1, G2, G3]) frames from 1990 onwards. These Gn (including recent G4, a continuation of saga by Curtis et al.) tools (each being a bunch of models intelligently interwoven/executed) are phase wise refinement in moving up the ladder with high level models for accurate (electronic) energy calculations. SOM with a niche in unsupervised paradigm, a new approach with sequential, parallel and hierarchical intelligent knowledge based numerical expert system front-/back-end and imbedded/infused heuristic modules is awaited. The combined results with other methods of choice viz. SVM, possibilistic procedures, information content and transformed mathematical spaces enhance the Xmetric-eye-vision (chemo-, software-, method-). The simulated-data-generators from simple as possible (SAP) to mega size of X and response with explicit functional relations are good training tools as well as a roadmap for further exploration/exploitation in the future frame. Experimental design and numerical expert systems of new generation to explain/control/repair/advise the way outs for stumble blocks real life problem solving are welcome features. The knowledge base for extraction of information/knowledge of the visual display of model and/or experimental results is a board for takeoff into future computational paradigm complimenting and supplementing human brain rational ventures.

REFERENCES

[1] W. James, *Principles of psychology*, **1890**, chapter XVI, 253-279.
[2] W.S. McCulloch, W. Pitts, *Bull. Math. Biophy.*, **1943,** 5, 115–133.
[3] S. Grossberg, *Biol. Cybernet.*, **1976**, 23, 121–134.
[4] J.J. Hopfield, *Proc. Nat. Acad. Sci.*, **1982**, 79, 2554-2558.
[5] J.A. Anderson, *Mathematical Biosciences*, **1972**, 14, 197-220; **1970**, 8, 137.
[6] S. Gadrari, K. Rose, *IEEE Trans. Commun.*, **1999**, 47, 1113–1116.
[7] A. Gersho, R.M. Gray, Vector Quantization and Signal Compression, Kluwer, Amsterdam, The Netherlands, **1992**.
[8] W.J. Hwang, F.J. Lin, C.T. Lin, *IEEE Commun. Lett.*, **2000**, 4, 408–410.
[9] W.J. Hwang, S.L. Hong, *Opt. Eng.* **1999**, 233–239.
[10] T. Heskes, *IEEE Trans. Neural Netw.*, **2001,** 12(6),1299-1305.
[11] T. Kohonen, *Proc. IEEE*, **1990,** 78, 1464-1480.
[12] T. Kohonen, *Alied optics* **1987,** 26, 1.
[13] T. Kohonen, *Self-Organizing Maps, 2nd edn., Springer-Verlag, Berlin* **1997,**
[14] T. Kohonen, *Neurocomput.*, **1998,** 21, 1–6.
[15] S. Kaski, J. Kangas, T. Kohonen, Bibliography of Self-Organizing Map SOM) papers: 1981–1997, *Neural Computing Surveys*, **1998,** *1, 102–350*.
[16] T. Kohonen, S. Kaski, K. Lagus, J. Salojarvi, J. Honkela, V. Paatero, A. Saarela, *IEEE Trans. Neural Netw.*, **2000,** 11, 574–585.
[17] M. Tobiszewski, S. Tsakovski, V. Simeonov, J. Namieśnik, *Green Chem.*, **2013**, 15, 1615-1623.
[18] C.G. Prato, V. Gitelman, S. Bekhor, *Accident Analysis & Prevention*, **2012,** 44 (1), *56-62*.
[19] C.S. Kong, J. Yu, F.C. Minion, K. Rajan, *ACS Comb. Sci.*, **2011**, 13 (5), 562–571.
[20] B. Fritzke, *Adv. Neural Inf. Process. Systems*, **1995** 7, 625–632.
[21] B. Fritzke, *Adv. Neural Inf. Process. Systems*, **1993** 5, 123–130.
[22] E. Swiercz, *Advan. Eng. Software*, **2012,** *45 (1), 28-41*.
[23] M. Cottrell, P. Rousset, *Advances in self-organising maps, Springer* **1997,** 861–871.
[24] W.N.S Mat-Desa, D. Ismail, N. NicDaeid, *Anal. Chem.*, **2011**, 83 (20), 7745–7754.
[25] S. Kittiwachana, D.L.S. Ferreira, L.A. Fido, D.R. Thompson, R.E.A. Escott, R.G. Brereton, *Anal. Chem.*, **2010,** 82 (14), 5972–5982.
[26] R. Fernández-Varela, M. P. Gómez-Carracedo, D. Ballabio, J. M. Andrade, V. Consonni, R. Todeschini, *Anal. Chem.*, ***2010**, 82 (10), 4264–4271*.
[27] K. Wongravee, G.R. Lloyd, C.J. Silwood, M. Grootveld, R.G. Brereton, *Anal. Chem.*, **2010**, 82 (2), 628–638.
[28] A.C. Jørgensen, J. Rantanen, P. Luukkonen, S. Laine, J. Yliruusi, *Anal. Chem.*, **2004**, 76 (18), 5331–5338.
[29] N. Minovski, Š. Župerl, V. Drgan, M. Novic, *Anal. Chim. Acta*, **2013,** *759, 28-42*.
[30] D. Ballabio, M. Vasighi, P. Filzmoser, *Anal. Chim. Acta*, **2013,** 765, 45-53.
[31] T. Voyslavov, S. Tsakovski, V. Simeonov, *Anal. Chim. Acta*, **2013,** 770, 29-35.
[32] Roman M. Balabin, Sergey V. Smirnov, *Anal. Chim. Acta*, **2011,** *692, (1–2), 63-72*.
[33] A. García-Reiriz, J. Magallanes, J. Zupan, S. Líberman , *Appl. Rad. Isotopes*, **2011,** 69 (12), *1793-1795*.
[34] A. Fathi, A. Mozaffari , *Appl. Soft Comput.*, **2014,** 14, Part B, 229-251.

[35] M.S. Hamdi , *Appl. Soft Comput.*, **2011,** 11 (1), 1310-1321.
[36] M. F.S.V. D'Angelo, R.M. Palhares, R.H.C. Takahashi, R.H. Loschi, L.M.R. Baccarini, W.M. Caminhas , *Appl. Soft Comput.*, **2011,** *11 (1), 179-192*.
[37] V.F. López, J.M. Corchado, J.F. De Paz, S. Rodríguez, J. Bajo , *Appl. Soft Comput.*, **2011,** 11 (2), *2925-2933*.
[38] M.H. Ghaseminezhad, A. Karami, *Appl. Soft Comput.*, **2011,** 11 (4), 3771-3778.
[39] M.B. Nasr, M. Chtourou, *Appl. Soft Comput.*, **2011,** 11 (8), 4458-4464.
[40] J. Nagi, K. Siah Yap, F. Nagi, S.K. Tiong, S.K. Ahmed , *Appl. Soft Comput.*, **2011,** 11 (8), 4773-4788.
[41] M. Piliougine, D. Elizondo, L. Mora-López, M. Sidrach-de-Cardona, *Applied Energy* **2013,** 112, 610-617.
[42] B. Fritzke, *Artificial neural networks II, North-Holland, Amersterdam* **1992,** 1051-1056.
[43] J. Herrero, A.Valencia, J. Dopazo, *Bioinformatics,* **2001,** 172, 126–136.
[44] F. Koeth, H.G. Marques, T. Delbruck, *Biol. Insp. Cognitive Architectures*, **2013,** 6, *8-11*.
[45] Christian R. Huyck, Ian G. Mitchell, *Biol. Insp. Cognitive Architectures*, **2013,** 6, *3-7*.
[46] M. Cottrell, E. de Bodt, M. Verleysen, *Biological and artificial computation: From neuroscience to technology, 7–14 Springer* **2001.**
[47] Y. Choe, R. Miikkulainen, *Biological Cybernetics*, **2004,** 90, 75–88.
[48] Li, X., Gasteiger, J. Aupan, J., *Biological Cybernetics,* **1993**, 70, 189–198.
[49] M. Alonso, C. Miranda, N. Martín, B. Herradón, *Phys. Chem. Chem. Phys.*, 2011,**13**, 20564-20574.
[50] S. Xuan, Y. Wu, X. Chen, J. Liu, A. Yan, *Bioorg. & Med. Chem. Let.*, **2013,** 23 (6), 1648-1655.
[51] F. Bonachera, G. Marcou, N. Kireeva, A. Varnek, D. Horvath, *Bioorg. & Med. Chem. Let.,* **2012,** 20 (18), 5396-5409.
[52] A. Yan, K. Wang, *Bioorg. & Med. Chem. Let.*, **2012,** 22 (9), 3336-3342.
[53] A. Yan, Y. Chong, L. Wang, X. Hu, K. Wang , *Bioorg. & Med. Chem. Let.*, **2011,** 21 (8), *2238-2243*.
[54] F. Macaev, Z. Ribkovskaia, S. Pogrebnoi, V. Boldescu, G. Rusu, N. Shvets, A. Dimoglo, A. Geronikaki, R. Reynolds , *Bioorg. & Med. Chem.*, **2011,** 19 (22), *6792-6807*.
[55] S. Kaski, J.Nikkil , M. Oja, J. Venna, P. Toronen, E. Castren, *BMC Bioinformatics*, **2003,** 4, 48.
[56] S. Kikuchi, Y. Onuki, A. Yasuda, Y. Hayashi, K. Takayama, J. Pharmaceutical Sci., **2011**, 100 (3), 964–975.
[57] Y. Bai, W. Zhang, Z. Jim , *Chaos, Solutions and fractals* **2006,** 28, 1082-1089.
[58] W. Luo, W. Fan, H. Xie, L. Jing, E. Ricicki, P. Vouros, L.P. Zhao, H. Zarbl , *Chem. Res. Toxicol.*, **2005**, 18 (4), 619–629.
[59] D. Ballabio, M. Vasighi , *Chemomet. Intel. Lab. Syst.*, **2012,** 118, 24-32.
[60] T. Voyslavov, S. Tsakovski, V. Simeonov , *Chemomet. Intel. Lab. Syst.*, **2012,** 118, 280-286.
[61] D. Adandedjan, S.A. Montcho, A. Chikou, P. Laleye, G. Gourene , *Comp. Rendus Biologies,***2013,** 336 (5–6), 244-248.
[62] Franck Assous, Joël Chaskalovic, *Comptes Rendus Mécanique* **2013,** *341 (3) 304-313*.
[63] Z. Wang, B. Zineddin, J. Liang, N. Zeng, Y. Li, M. Du, J. Cao, X. Liu , *Comput. Methods and Programs in Biomedicine,* **2013,** *111 (1), 189-198*.
[64] I. Świetlicka, W. Kuniszyk-Jóźkowiak, E. Smołka, *Comput. Speech & Language,* **2013,** *27 (1*), *228-242*.
[65] A.Sangole, G. K. Knopf, *Computers & Graphics,* **2003,** 276), 963–976.
[66] R. Sambasiva Rao et.al. (unpublished)
[67] K. Chakraborty, A. De, A. Chakrabarti, *Computers & Elect. Eng.*, **2012,** 38 (4), *819-826*.
[68] A. Sangole, G.K. Knopf, *Computers & Graphics,* **2003,** 276, 963–976.
[69] L. Calabrese, G. Campanella, E. Proverbio, *Construction and Building Materials,* **2012,** 34, *362-371*.
[70] P.du Jardin, E. Séverin, *Decision Support Systems,* **2011,** 51 (3), *701-711*.

[71] N.L. Beebe, J.G. Clark, G.B. Dietrich, M.S. Ko, D. Ko, *Decision Support Systems,* **2011,** 51 (4), *732-744*.

[72] M. Dukowska, M. Grzybkowska, A. Kruk, E. Szczerkowska-Majchrzak , *Ecolog. Model.,* **2013,** 265, 221-229.

[73] M. López, S. Valero, C. Senabre, J. Aparicio, A. Gabaldon , *Electric Power Syst. Res.,* **2012,** 91, *18-27*.

[74] Y. Zhao, C. Xu, S. Zhao, Q. Shi, *Energy Fuels*, **2012**, 26 (12), 7251–7256.

[75] E. Lochin, B. Talavera, *Eng. App. Art. Intel.,* **2011,** 24 (1), 77-86.

[76] A. J. Adeloye, R. Rustum, I. D.Kariyama, *Environ. Model. & Software* **2012,** 29 (1), 61-73.

[77] R. Rallo, B. France, R. Liu, S. Nair, S. George, R. Damoiseaux, F. Giralt, A. Nel, K. Bradley, Y. Cohen, *Environ. Sci. Technol* **2011**, 45 (4), 1695–1702.

[78] B. Campos, N. Garcia-Reyero, C. Rivetti, L. Escalon, T. Habib, R. Tauler, S. Tsakovski, B. Piña, C. Barata, *Environ. Sci. Technol.,* **2013**, 47 (16), 9434–9443.

[79] A. Afantitis, G. Melagraki, P.A. Koutentis, H. Sarimveis, G. Kollias, *European J. Med. Chem.,* **2011,** 46 (2), 497-508.

[80] Philippe du Jardin, Eric Séverin, *European J. Oper. Res.,* **2012,** 221(2), 378-396.

[81] E.M. Borkowska, A. Kruk, A. Jedrzejczyk, Z. Jablonowski, M. Constantionou, M. Traczyk, M. Pietrusinski, M. Banaszkiewicz, P. Marks, M. Rozniecki, M. Sosnowski, B. Kaluzewski , *European Urology Supplements* **2012,** 11 (4), *104*.

[82] E.M. Borkowska, A. Kruk, A. Jedrzejczyk, M. Rozniecki, Z. Jablonowski, M. Traczyk, M. Constantinou, M. Banaszkiewicz, M. Pietrusinski, M. Sosnowski, F.C. Hamdy, S. Peter, J.W.F. Catto, B. Kaluzewski , *European Urology Supplements*, **2013,** 12 (4), e1213, C105.

[83] D.H. Milone, G.Stegmayer, L. Kamenetzky, M. López, F. Carrari, *Expert Syst. Appl.,* **2013,** 40 (9), *3841-3845*.

[84] K.L. Chung, Y. Huang, J. Wang, M. Cheng, *Expert Syst. Appl.,* **2012,** 39 (3), 2427-2432.

[85] J. Rasti, A. Monadjemi, A. Vafaei, *Expert Syst. Appl.,* **2011,** 38 (10), 13188-13197.

[86] M.M. Mostafa, *Expert Syst. Appl.,* **2011,** 38 (6), 6906-6915.

[87] Mohamed M. Mostafa, *Expert Syst. Appl.,* **2011,** 38 (7), *8782-8803*.

[88] R.Z. Cabada, M.L. Barrón, Estrada, Carlos Alberto Reyes García, *Expert Syst. Appl.,* **2011,** 38 (8), 9522-9529.

[89] C. J. O'Malley, G.A. Montague, E.B. Martin, J. M. Liddell, B. Kara, N.J. Titchener-Hooker , *Food and Bioproducts Processing* **2012,** 90 (4), *755-761*.

[90] Saji Kuriakose, Hubert Joe, *Food Chem.,* **2012,** 135 (1), *213-218*.

[91] X. Yang, Y. Chong, A. Yan, J. Chen, *Food Chem.,* **2011,** 128 (3), *653-658*.

[92] Jean Daniel Coïsson, Marco Arlorio, Monica Locatelli, Cristiano Garino, Donatella Resta, Elena Sirtori, Anna Arnoldi, Giovanna Boschin, *Food Chem.,* **2011,** 129 (4), *1806-1812*.

[93] F. A.L. Ribeiro, F.F. Rosário, M. C.M. Bezerra, R.C.C. Wagner, A.L.M. Bastos, V.L.A. Melo, R.J. Poppi, *Fuel,* **2014,** 117, 381-390.

[94] Jolanta J. Adamczyk, Andrzej Kruk, Tadeusz Penczak, David Minter, *Fungal Biol.,* **2012,** 116 (9), *995-1002*.

[95] Alberto Faro, Daniela Giordano, Francesco Maiorana, *Future Generation Computer Syst.* **2011,** 27 (6), *711-724*.

[96] M.Yang, W.Hung, De-Hua Chen, *Fuzzy Sets and Systems,* **2012,** 203, 49-73.

[97] Vuorimaa P, *Fuzzy Sets and Systems,* **1994,** 662), 223–231.

[98] P. F. Lamb, A. Mündermann, R.M. Bartlett, A. Robins , *Gait & Posture*, **2011,** 34 (4), *485-489*.

[99] M. Pfeiffer, A. Hohmann , *Human Movement Sci.,* **2012,** 31 (2), *344-359*.

[100] M. C. Su, H. T. Chang, , *IEEE Trans. Neural Netw.,* **2001** 12, 153–158.

[101] J.M. Whitacre, R.A. Sarker, Q. T. Pham, *IEEE Trans. Evol. Comput.,* **2008,** 12,220-230.

[102] F.L. Chung, T.C. Fu, V. Ng, R.W.P. Luk, *IEEE Trans. Evolut. Comput.,* **2004,** 8.

[103] J.M. Whitacre, R.A. Sarker, Q.T. Pham, *IEEE Trans. Evolutionary Comput* **2008,** 12.

[104] V.J. Hodge, J. Austin, *IEEE Trans. Knowl. Data Eng.,* **2001,** 13, 207–218.

[105] A. Rauber, D. Merkl, M. Dittenbach, *IEEE Trans. Neural Netw.* **2002,** 13, 1331–1341.
[106] H. Yin, *IEEE Trans. Neural Netw.* **2002,** 13, 237–24.
[107] H. Yin, *IEEE Trans. Neural Netw.* **2002,** 131, 237–243.
[108] M. C.Su, H. T.Chang, *IEEE Trans. Neural Netw.* **2001,** 12, 153–158.
[109] H. Yin, N. Allinson, *IEEE Trans. Neural Netw.* **2001,** 12, 405–411.
[110] A. Konig, *IEEE Trans. Neural Netw.* **2000,** 113, 615–624.
[111] P. Demartines, J. Herault, *IEEE Trans. Neural Netw.* **1997,** 81, 148–154.
[112] T.Villmann, R. Der, M.Herrmann, T.M.Martinetz, *IEEE Trans. Neural Netw.* **1997,** 82, 256–266.
[113] A.H. Tan, N. Lu, D. Xiao, *IEEE Trans. Neural Netw.*, **2008,** 19, 230-244.
[114] G. G. Yen, Z. Wu, *IEEE Trans. Neural Netw.*, **2008,** 19, 245-259.
[115] D. Brugger, M. Bogdan, W. Rosenstiel, *IEEE Trans. Neural Netw.*, **2008,** 19, 442-459.
[116] S. D. Teddy, C. Quek, E. M.K. Lai, *IEEE Trans. Neural Netw.*, **2008,** 19, 689-712.
[117] C.F. Hsu, *IEEE Trans. Neural Netw.*, **2007,** 18,1232-1241.
[118] Y. Zhao, J.A. Farrell, *IEEE Trans. Neural Netw.*, **2007,** 18, 1220-1231.
[119] D. Vidaurre J. Muruzábal, *IEEE Trans. Neural Netw.*, **2007,** 18, 1524-1528.
[120] S. Wu, T.W.S. Chow, *IEEE Trans. Neural Netw.*, **2007,** 18, 385-396.
[121] Y. Cheung, L. Law, *IEEE Trans. Neural Netw.*, **2007,** 18, 289-295.
[122] S. Pal, A. Datta, N.R. Pal, *IEEE Trans. Neural Netw.*, **2001,** 12.
[123] J. B. Lont, W. Guggenbuhl, *IEEE Trans. Neural Netw.*, **1992,** 3 457-.
[124] W.C. Fang, O.T.C Chen, *IEEE Trans. Neural Netw.*, **1992,** 3 506.
[125] S. Mitra, S.K. Pal, *IEEE Trans. Systems, Man and Cybernetics,* **1994,** 243, 385–399.
[126] S. Mitra, S. K. Pal, *IEEE Trans.Systems, Man and Cybernetics.*, **1996,** 265, 608–620.
[127] J. W. Sammon, *IEEE Transactions on Computers,* **1969,** C18, 401–409.
[128] J. Vesanto, E. Alhoniemi, *IEEE Transactions on Neural Networks,* **2000,** 113, 586–600.
[129] X. Pascual, H. Gu, A. Bartman, A. Zhu, A. Rahardianto, J. Giralt, R. Rallo, P. D. Christofides, Y. Cohen, *Ind. Eng. Chem. Res.*, **2014,** 44, (000).
[130] M. Ghorbanzadeh, M.H. Fatemi, M. Karimpour, *Ind. Eng. Chem. Res.,* **2012**, 51 (32), 10712–10718.
[131] F. Corona, M. Mulas, R. Baratti, J. A. Romagnoli, *Ind. Eng. Chem. Res.,* **2012**, 51 (42), 13732–13742.
[132] B. Bhushan, J.A. Romagnoli, A. Gordon, M. Cain, *Ind. Eng. Chem. Res.*, **2008**, 47 (12), 4209–4219.
[133] Y.S. Ng, R. Srinivasan, *Ind. Eng. Chem. Res.,* **2008**, 47 (20), 7744–775.
[134] Y.S. Ng, R. Srinivasan, *Ind. Eng. Chem. Res.,* **2008**, 47 (20), 7758–777.
[135] H. Zhuang, W. Ang, M. Ohshima, M. Chiu, *Ind. Eng. Chem. Res., **2002**, 41 (12), 2941–2947.*
[136] M. Ge, M. Chiu, Q. Wang, *Ind. Eng. Chem. Res.*, **2000**, 39 (10), 3778–3788.
[137] C.A. Astudillo, B.J. Oommen, *Information Sci.*, **2011,** 181 (18), 3798-3815.
[138] J.A. Marshall, V. Srikanthy, *Int. J Neural Syst.*, **2000,** 10, 59-70.
[139] Andras P, *Int. J. Neural Systems,* **2002,** 12, 117–135.
[140] Vesanto J, *Intelligent Data Analysis,* **1999,** 32, 111–126.
[141] A.C. M. da Silva, A.R.G. Castro, V. Miranda , *Inter. J. Electrical Power & Energy Syst.,* **2012,** 43 (1), 1034-1042.
[142] J.S. Torrecilla, E. Rojo, M. Oliet, J.C. Domínguez, F. Rodríguez , *J. Agric. Food Chem.,* **2009**, 57 (7), 2763–2769.
[143] L. Urruty, J. Giraudel, S. Lek, P. Roudeillac, M. Montury , *J. Agric. Food Chem.,* **2002,** 50 (11), 3129–3136.
[144] M.T. Hyvönen, Y. Hiltunen, W. El-Deredy, T. Ojala, J. Vaara, P.T. Kovanen, M. Ala-Korpela, *J. Am. Chem. Soc.,* **2001**, 123 (5), 810–816.
[145] B.K. Lavine, C.E. Davidson, D.J. Westover, *J. Chem. Inf. Comput. Sci.,* **2004**, 44 (3), 1056–1064.
[146] A. Yan, J. Gasteiger, *J. Chem. Inf. Comput. Sci.,* **2003**, 43 (2), 429–434.
[147] Z.R. Yang, K. Chou, *J. Chem. Inf. Comput. Sci.,* **2003**, 43 (6), 1748–1753.

[148] F. Hoehn, E. Lindner, H.A. Mayer, T. Hermle, W. Rosenstiel , *J. Chem. Inf. Comput. Sci.,* **2002**, 42 (1), 36–45.

[149] R.D. Beger, D.A. Buzatu, J.G. Wilkes, O. Lay, Jr, *J. Chem. Inf. Comput. Sci*., **2002**, 42 (5), 1123–1131.

[150] Y. Xiao, R. Harris, E. Bayram, P. Santago II, J. D. Schmitt , *J. Chem. Inf. Model* **2006**, 46 (1), 137–144.

[151] M. von Korff, T. Sander, *J. Chem. Inf. Model., **2006**, 46 (2), 536–544*.

[152] J.M. Otaki, A. Mori, Y. Itoh, T. Nakayama, H. Yamamoto , *J. Chem. Inf. Model., **2006**, 46 (3), 1479–1490*.

[153] A. Yan, *J. Chem. Inf. Model*., **2006**, 46 (6), 2299–2304.

[154] Y. Xiao, A. Clauset, R. Harris, E. Bayram, P. Santago, J.D. Schmitt , *J. Chem. Inf. Model*., **2005**, 45 (6), 1749–1758.

[155] Q. Zhang, J. Aires-de-Sousa, *J. Chem. Inf. Model*., **2005**, 45 (6), 1775–1783.

[156] M. Fernández, A. Tundidor-Camba, J. Caballero, *J. Chem. Inf. Model*., **2005**, 45 (6), 1884–1895.

[157] M. Lee, G. Schneider, *J. Comb. Chem*., **2001**, 3 (3), 284–289.

[158] S. Cassani, S. Kovarich, E. Papa, P.P. Roy, L. van der Wal, P. Gramatica , *J. Hazardous Materials,* **2013,** 258–259, 50-60.

[159] K. Srinivasa Raju, D. Nagesh Kumar, *J. Hydro-environ. Res*., **2011,** *5 (2), 101-109*.

[160] A.A. Rabow, R.H. Shoemaker, E.A. Sausville, D.G. Covell, *J. Med. Chem*., **2002**, 45 (4), 818–840.

[161] I.V. Tetko, V.V. Kovalishyn, D.J. Livingstone, *J. Med. Chem*., ***2001***, 44 (15), 2411–2420.

[162] J. Dopazo, J. M. Carazo, *J. Molecular Evolution,* **1997,** 44, 226–233.

[163] I. Cavero, J. Guillon, *J. Pharmacological and Toxicological Methods,* **2013.** *In Press, Uncorrected Proof*.

[164] V. Mäkinen, T. Tynkkynen, P. Soininen, T. Peltola, A.J. Kangas, C. Forsblom, L.M. Thorn, K. Kaski, R. Laatikainen, M. Ala-Korpela, P.H. Groop , *J. Proteome Res*., **2012**, 11 (3), 1782–1790.

[165] José Valero Galván, Luis Valledor, Rafael Mª. Navarro Cerrillo, Eustaquio Gil Pelegrín, Jesus V. Jorrín-Novo, *J. Proteomics,* **2011,** 74 (8), *1244-1255.*

[166] Vlassis N A, *Lecture notes in computer science* **1997,** P649, 1337.

[167] Margush, T, McMorris, F. R, *Mathematical Biology,* **1981,** 43, 239–244.

[168] M. Prevolnik, D. Andronikov, B. Žlender, M. Font-i-Furnols, M. Novič, D. Škorjanc, M. Čandek-Potokar , *Meat Sci*., **2014,** *96 (1), 14-20.*

[169] H.M. Cartwright, *Neural Comput & Alic.,* **2002,** 11, 30-36.

[170] G. Cheng, A. Zell, *Neural Comput & Alic.,* **2001,,** 10,89–97.

[171] Bishop, C. Svensen, M. Williams, *Neural comput*., **1998** 101, 215–234.

[172] J. Sirosh, R. Miikkulainen, *Neural comput.,* **1997,** 93, 577–594.

[173] T. Graepel, K. Obermayer, *Neural comput.,* **1999,** 11, 139-155.

[174] Y. Cheng, *Neural Comput.,* **1997,** 9, 1667–1676.

[175] K. Haese, *Neural comput.,* 1999 111, 211-1233.

[176] S.V. Adams, T. Wennekers, S. Denham, P.F. Culverhouse, *Neural Netw.,* **2013,** 44, *6-21.*

[177] Marta Kolasa, Rafał Długosz, Witold Pedrycz, Michał Szulc, *Neural Netw.,* **2012,** 25, 146-160.

[178] F. Shen, O.Hasegawa, *Neural Netw.,* **2006,** 19, 90–106.

[179] F. Shen, O. Hasegawa, *Neural Netw.,* **2006,** 19, 694–704.

[180] R.T. Freeman, H. Yin, *Neural Netw.,* **2004,** 1255–1271.

[181] M. Cottrell, S. Ibbou, P. Letremy, *Neural Netw.,* **2004,** 17, 1149–1168.

[182] J. Peltonen, A. Klami, S. Kaski, *Neural Netw.,* **2004,** 178–9, 1087–1100.

[183] S. Seo, K. Obermayer, *Neural Netw.,* **2004,** 178–9, 1211–1229.

[184] P. J. Somervuo, *Neural Netw.,* **2004,** 178–9, 1231–1239.

[185] A. Rauber, D. Merkl, M. Dittenbach, *Neural Netw.,* **2002,** 136, 1331–1341.

[186] H. Yin, *Neural Netw.,* **2002,** 15, 1005-1016.

[187] M. van Hulle, *Neural Netw.,* **2002,** 15, 1029–1039.

[188] S. Mitra, Y. Hayashi, *Neural Netw.,* **2000,** 113, 748–768.

[189] T. Villmann, R. Der, M. Herrmann, T.M.Martinetz, *Neural Netw.*, **1997,** 82, 256–266.
[190] B. Fritzke, *Neural Netw.*, **1994,** 7,1441-1460.
[191] S. Furao, T. Ogura, O. Hasegawa, *Neural Netw.*, **2007,** 20, 893–903.
[192] S. Mahony, P.V. Benos, T.J. Smith, A. Golden, *Neural Netw.*, **2006,** 19, 950–962.
[193] V.G. Kaburlasos, S.E. Papadakis, *Neural Netw.*, **2006,** 19, 623–643.
[194] T. Kohonen, *Neural Netw.*, **2006,** 19, 723–733.
[195] T.J. Sullivan, V.R. de Sa, *Neural Netw.*, **2006,** 19, 734–743.
[196] J. Ontrup, H. Ritter, *Neural Netw.*, **2006,** 19, 751–761.
[197] H. Yin, *Neural Netw.*, **2006,** 19, 780–784.
[198] G.A. Barreto, L.G.M. Souza, *Neural Netw.*, **2006,** 19, 785–798.
[199] T. Koga, K. Horio, T. Yamakawa , *Neural Netw.*, **2006,** 19, 799–811.
[200] J. Rynkiewicz, *Neural Netw.*, **2006,** 19, 830–837.
[201] P. Rousset, C. Guinot, B. Maillet, *Neural Netw.*, **2006,** 19, 838–846.
[202] L. Lebart, *Neural Netw.*, **2006,** 19, 847–854.
[203] B. Conan-Guez, F. Rossi, A. El Golli, *Neural Netw.*, **2006,** 19, 855–863.
[204] Y. Wu, M. Takatsuka , *Neural Netw.*, **2006,** 19, 900–910.
[205] G. Polzlbauer, M. Dittenbach, A. Rauber, *Neural Netw.*, **2006,** 19, 911–922.
[206] E.V. Samsonova, J.N. Kok, A.P. IJzerman, *Neural Netw.*, **2006,** 19, 935–949.
[207] B. Hammer, A. Micheli, A. Sperduti, M. Strickert, *Neural Netw.*, **2004,** 17, 1061–1085.
[208] M. Cottrell, S. Ibbou, P. Letremy, *Neural Netw.*, **2004,** 17, 1149–1167.
[209] S. Seo, K. Obermayer, *Neural Netw.*, **2004,** 17, 1211–1229.
[210] J.A. Flanagan, *Neural Netw.*, **2001,** 14, 1405–1417.
[211] M. M. Campos, G.A. Carpenter, *Neural Netw.*, **2001,** 14, 505-525.
[212] J. Kuroiwa, S.Inawashiro, S. Miyake, H. Aso, *Neural Netw.*, **2000,** 13, 31-40.
[213] J.A. Marshall, *Neural Netw.*, **1995,** 8, 335-362.
[214] F. M. Mulier, V.S. Cherkassky, *Neural Netw.*, **1995,** 8, 717-727.
[215] D. Martinez, M.M. Van Hulle, *Neural Netw.*, **1995,** 8, 891-900.
[216] P. Thiran, M. Hasler, *Neural Netw.*, **1994,** 7, 1427-1439.
[217] K. Matsuoka, M. Kawamoto, *Neural Netw.*, **1994,** 7, 753-765.
[218] S. Jockusch, H. Ritter, *Neural Netw.*, **1994,** 71229-1239.
[219] B. Fritzke, *Neural Process. Lett.* **1995,** 2, 9–13.
[220] M. Mitsumori, S. Nakagawa, H. Matsui, T. Shinkai, A. Takenaka, J. Appl. Microbiology, 2010, 109(3), 763–770.
[221] J. Pakkanen, J. Iivarinen, E. Oja, *Neural Processing Letters*, **2004,** 203, 199–211.
[222] J.J. Verbeek, N. Vlassis, B.J.A. Krose, *Neurocomput.*, **2005,** 63, 99–123.
[223] J. Qiao, H. Wang, *Neurocomput.*, **2008,** 71, 564–569.
[224] Y. Liu, X. Wang, C. Wu, *Neurocomput.*, **2008,** 71, 857–862.
[225] R.T. Freeman, H. Yin, *Neurocomput.*, **2005,** 63, 415–446.
[226] M.M. Merino, A. Munoz, *Neurocomput.*, **2005,** 63,171–192.
[227] Y. Matsuda, K. Yamaguch, *Neurocomput.*, **2005,** 64, 285–299.
[228] M.F. Yeh, K.C. Chang, *Neurocomput.*, **2005,** 67, 281–287.
[229] I. Valova, D. Szer, N. Gueorguieva, A. Buer, *Neurocomput.*, **2005,** 68, 177–195.
[230] B. Hammer, A. Micheli, A. Sperduti, M. Strickert, *Neurocomput.*, **2004,** 57,3 – 35.
[231] Y. Han, E. Corchado, C. Fyfe, *Neurocomput.*, **2004,** 57, 37 – 47.
[232] J. A. Lee, A. Lendasse, M. Verleysen, *Neurocomput.*, **2004,** 57, 49–76.
[233] K.I. Amemori, S. Ishii, *Neurocomput.*, **2004,** 61, 291 – 316.
[234] K.S. Leung, H.D. Jin, Z.B. Xu, *Neurocomput.*, **2004,** 62, 267 – 292.
[235] M. Dittenbach, A. Rauber, D. Merkl, *Neurocomput.*, **2002,** 48, 199–216.
[236] D. Kim, S. Ahn, D.S. Kang, *Neurocomput.*, **2000,** 30 249-272.
[237] C. Ambroise , G. Seze, F. Badran, S. Thiria , *Neurocomput.*, **2000,** 30, 47-52.
[238] M. Cottrell, J.C. Fort, G. Pages, *Neurocomput.*, **1999,** 21, 119–138.

[239] D. Merkl, *Neurocomput.,* **1998,** 21 61–77.
[240] R.J.T. Graepel, M. Burger, K. Obermayer, *Neurocomput,* **1998,** 21, 173–190.
[241] S. Delgado, C. Gonzalo, E. Martinez, A. Arquero, *Neurocomputing* **2011,** 74 (16), 2624-2632.
[242] L. Gajecki, *Neurocomputing,* **2014.** In Press,.
[243] T. Hachaj, M.R. Ogiela, *Neurocomputing,* **2013,** 122, *33-42*.
[244] C.W.D. de Almeida, R.M.C.R. de Souza, A. L.B. Candeias, *Neurocomputing,* **2013,** 99, 65-75.
[245] P. Sarlin, *Neurocomputing,* **2013,** 99, 496-508.
[246] D.J. Hemanth, C.K.S. Vijila, A.I. Selvakumar, J. Anitha , *Neurocomputing,* **2013,** (in press).
[247] N. Ilc, A. Dobnikar, *Neurocomputing,* **2012,** 96, 47-56.
[248] D. Beaton, I. Valova, D. MacLean, *Neurocomputing,* **2011,** *74 (17), 3125-3141*.
[249] R. Kamimura, *Neurocomputing,* **2011,** 74 (7), *1116-1134*.
[250] G. Cleuziou, *Pattern Recog. Let.,* **2013,** 34 (3), *239-246*.
[251] A. Majumder, L. Behera, V.K. Subramanian, *Pattern Recog.,* **2014,** 47 (3), 1282-1293.
[252] Tsao, E. C.-K., Bezdek, J. C., Pal, N. R., *Pattern Recog.,* **1994,** 275, 757–764.
[253] I. Valova, G. Georgiev, N. Gueorguieva, J.Olson , *Procedia Computer Sci.,* **2013,** 20, 52-57.
[254] G. Serpen, J. Li, L. Liu , *Procedia Computer Sci.,* **2013,** 20, *406-413*.
[255] Z.B. Mustapha, S. Alvain, C. Jamet, H. Loisel, D. Dessailly, *Remote Sensing of Environ.,* **2013,** (in press).
[256] S. Grebby, J. Naden, D. Cunningham, K. Tansey, *Remote Sensing of Environ.,* **2011,** 115 (1), *214-226*.
[257] M.L. Carreño, O.D. Cardona, A.H. Barbat , *Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería,* **2011,** *27 (4*), *278-293*.
[258] F. Palamara, F. Piglione, N. Piccinini, *Safety Sci.,* **2011,** 49 (8–9), 1215-1230.
[259] R. Carafa, L. Faggiano, M. Real, A. Munné, A. Ginebreda, H. Guasch, M. Flo, L. Tirapu, P.C. von der Ohe, *Sci. of The Total Environ.* **2011,** 409 (20), *4269-4279*.
[260] P. Klement, V. Snášel , *Simul. Model. Practice and Theory,* **2011,** 19 (1), *98-109*.
[261] C. Tan, H. Chen, C. Wang, W. Zhu, T. Wu, Y. Diao, *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy,* **2013,** 105, *1-7*.
[262] C. H. Kowalski, G.A. da Silva, H.T. Godoy, R.J. Poppi, F. Augusto , *Talanta,* **2013,** 116, 315-321.
[263] P. Phan, N. Mezghani, E.K. Wai, J. Guise, H. Labelle , *The Spine Journal,* **2013,** 13 (11), 1527-1533.
[264] X. Hu, A. Yan , *Toxicology in Vitro,* **2011,** 25 (8), 2017-2024.
[265] Y.S. Hong, R. Bhamidimarri, *Water Res.,* **2003**, 37, 1199–1212.