Available online at www.joac.info

ISSN: 2278-1862

Journal of Applicable Chemistry

2014, 3 (5): 1807-1893

(International Peer Reviewed Journal)

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

Mathematical Neural Network (MaNN) Models
Part V: Radial basis function (RBF) neural networks (NNs) in Chemometrics, Envirometrics and Medicinometrics (ChEM)

K RamaKrishna[1], Ch. V. Kameswara Rao[1,3], V. Anantha Ramam[2] and R. Sambasiva Rao[2*]

1. Dept. of Chemistry, Gitam Institute of Science, Gitam University, Visakhapatnam, 530 017, **INDIA**
2. School of Chemistry, Andhra University, Visakhapatnam 530 003, **INDIA**
3. Department of Chemistry, GMRIT, Rajam 532 127, **INDIA**

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

Accepted on 20th September2014

(Dedicated with reverence to Sri S Somayajulu, former Head of chemistry Dept., S R R & C V R Govt. College, Vijayawada on his birth centenary celebrations)

ABSTRACT
Radial basis function (RBF)-neural network (NN) has a single hidden (radial) layerof neurons with Gaussian kernelas transfer function (TF). The names of many RBF_NNs correspond to the type TF viz. raised cosine-, generalised-binary, q-, Hunt- etc. The product type functions are Lowe, thin-plate and AuPar. *The individual neurons in hidden layer of NN operated by TF (RBF, sigmoid) perform non-linear operation and layer as a whole maps input space into higher dimensions.A two phase training involving determination of centers of RBFs by clustering procedures followed by estimation of WRLO (weight vector of connections between neurons of radial and output layers) is the simplest protocol adapted. Although pseudo-inverse and orthogonal procedures are sought after optimization methods in weights refinement, Bayesian Ying-Yang (BYY-), incremental-, reinforced-, rival-penalized-continuous- (RPCL) and life-long-learning are used with success. Universal function approximation theorem, convergence proofs and error boundsimparted a strong theoretical support.*

The evolution in architecture leads to recurrent-, self-organizing-, growing- and shrinking categories. Clifford/complex-RBF_NNs accept imaginary values for input unlike other RBF category. The trained network is pruned by temporary dynamic decay algorithm. The resource-allocating- (RA-), minimum-RA-, dynamic-decay-adjustment- belong to growing architecture category. The off-spring of RBF_NN are generalized regression- and probabilistic- NNswith statistical flavor. The power of RBF_NN increases with evolution of structure of network and weights of connections. The addition/deletion of RB neurons, connection making/breaking are implementable through mutation operators. Novelty detection, popularized by Grossberg in ART type NNs, is implemented in RBF_NN.Binary hybridization of RBF_NN with wavelets, support vector machines (SVM), self-organizing maps(SOM), logistic regression etc. increased the functional value of this NN. Nature inspired algorithms viz. evolutionary strategy, genetic algorithm, particle swarm optimization(PSO), mimetic approach, honey bee algorithm are instrumental in arriving at viable solution of intractable hard task of simultaneous optimization of number of clusters, their centers/ widths and weights. SOM-Generalised_RBF emulates finite automata.The imbibing capability of RBF_NN, novelty detection, robustcharacter brought it to the forefront in modeling phase of *interdisciplinary research tasks. The ensembles, voting methods and Pareto-front brought renaissance to multiple alternate decisions instead of single best one of yesteryears. The applications of RBF_NN and its clones encompass science, engineering, industry, commerce and forex. The noteworthy results in chemometrics, envirometrics, piscimetrics, pharmacometrics and medicinometrics are briefly discussed pertaining to multi-variate-multi-response calibration, function approximation, interpolation, classification with non-linear boundaries, time-series data, pattern recognition and parameterization.*

Keywords: Radial basis function, Neural network, Chemometrics, Medicine, Pharmacometrics, Environment, Interpolation, Function approximation, Classification.

Contents: Radial basis function_NN in chemical sciences

INTRODUCTION

Simplism in modeling of many real life phenomena in nineteenth century was not by choice but limited by the tools. Enquiry in twentieth century set firm ground for fundamental laws of quantum physics/chemistry, molecular biology, particle physics, computer technology, information theory and computational science. Now, with advances in twenty first century, experimental handling of nanomaterials, single cell/ molecule, hyphenated_multisensor instruments, peta-scale computing power, (mobile) cloud-computing facilities, claytronics (virtual reality) 3D-visuals, humanoid-robots with E-eye/E-nose/E_tongue, nature-mimicking algorithms/software are routine and models at micro level details are a child play.

Neuroscience crossed teething problems, attained maturity and it is right time for a change to higher intellectual level as otherwise only breadthwise expansion of pseudo replica will dominate. The neurological activity of recognition/ remembering of one's own grandmother's face and an anonymous face is distributed throughout the whole cortex of human brain(Fig. 1). Sharks are most electrically sensitive animals. They respond to even as low as 5 nV/cm DC fields and sensing an electric dipole alone is sufficient to invade a prey. Always, a more sophisticated/advanced theory is better than its predecessor. The newer one gives a good description of more expanded domains of science and/or better/accurate description of the same domain. Further, the relation between the newer theory and its predecessor gives one the power to recover the older theory with ease and modeled by straight forward (recent domains) mathematics. In a nutshell, interdisciplinary scientific developments are fast emerging /evolving necessitating holistic approach through binary/ ternary/ quaternary hybridization. The sequential, loose-coupled, conditional, tight bound and integrated forms of combining algorithms /disciplines/ domains/ instruments are in vogue in advanced projects.

The primary activity in NN modeling is optimizing the architecture and weights (Ws) of connections between neurons (processing units) for real life task /simulated/ designed training datasets.

The preliminary assessment of (interpolation/extrapolation) practicability is with simulated/toy/real data sets of standard category/bench mark/in house type test cases. The results are compared either with non-NN models or earlier reported NNs. The reported results progressively pave way to the breadth wise acceptance of the technology and directs towards a study in depth. The gap between algorithms of research outcome and availability in software packages continues. The improvement of current algorithms by a few groups through personnel communication is the important step in the stabilization and micro modification before its wide spread use. The second phase, many a time, is expensive pruning/reanalysis of the trained NN, looking into confidence contours, generation (production) of ensembles/ forests and their analysis for better generalizability and robustness. For mega-tasks using NNs in F16 bombers, humanoid robots, decisions in the government/space exploration/ocean technology and rational drug design, the cycle (vide supra) is a pre requisite. Further, reinvestigation of a decade old models are useful as starters in this brain storming modeling.

The expert system driven numerical computations for complex equilibria, chemical kinetics and solute-solvent interaction were reported from this laboratory [1-12]. In 1990, generation of computer software from basic modules of optimization, convergence and initialization under one cover was proposed for complex equilibria programs from 1960s onwards. The applications of NNs in hydrogen ion effect on rate constants in solution phase kinetics, bauxite ore benefaction and multi-component-multivariate calibration were studied. Recently, typical NN architectures viz. SOM, ART/ARTMAP, recurrent nets and E-man and swarm intelligence are reviewed [1] with applications in chemical-/medicinal-/environmental-/pharmaceutical- sciences, chemical-technology/ engineering and commerce. In this communication we report evolution in architecture, training methods of radial basis function neural network (RBF_NN) (a sibling of single_hidden_layer NN of feed_forward- category) and a wide range of its typical applications [13-243]. The futuristic researchers with these interfaces probe into micro- to yocto- (septillionth or 10^{-24} gm) level details of processes and routes to their roots.

2.Radial Basis Function (RBF) Neural Network (RBF_NN)

2.1 Biological inspiration

In several segments of the nervous system[132], a short-range response of a biological neuron is frequent and dominant. Also, auditory system (selective to a small band of frequencies of sound and cells in visual cortex (sensitive to a limited region in the visible spectrum) are typical sense organswith locally activated neurons. These trivial biological phenomena for a common man of times immemorial but of incredible and not easily replicable technological products are the inspiration for the scientist to probe deep into RBF_NNs. The first publication in radial basis function (RBF) dates back to 1960s in the context of classification [33]of objects.According to Cover’s theorem[76] a pattern classification task in a high (compared to that in low) dimensional space is more likely linearly separable. This promoted research in support vector machines.

The association of words and their figurative representations, terminology and concepts, hierarchical explanations and relevance, procedures (numerical, logical, conceptual) and corresponding do’s /don’ts etc. are learnt, memorized (short term, long term, need-based), recapitulated in toto /perturbed versions, integrated (intentional while awake, during rest/sleep). New alternate viabilities, rare sparkles (known/unknown), discoveries, and inventions are all just miraculous consequences of millions of processes over time in the human brain of an individual.Horzyk[156] modeled a select set of sequences of processes in biological associative neural systems paving way to develop, dialate/epitomize knowledge in a human-like fashion. Here, the trained artificial associative neural network architecture, corresponding weights and also previous states of neurons are utilized to trigger new paths. This leads to generalization, robustness and (may be small) creativity. The results show not only routine training and classification of static objects but also explore new sequences with the notable outcome. It is another root of exploiting knowledge based searches, which are better than heuristic, numerical rules extracted from data or simple meta-knowledge structures underwent several phases of development over the last half a century.

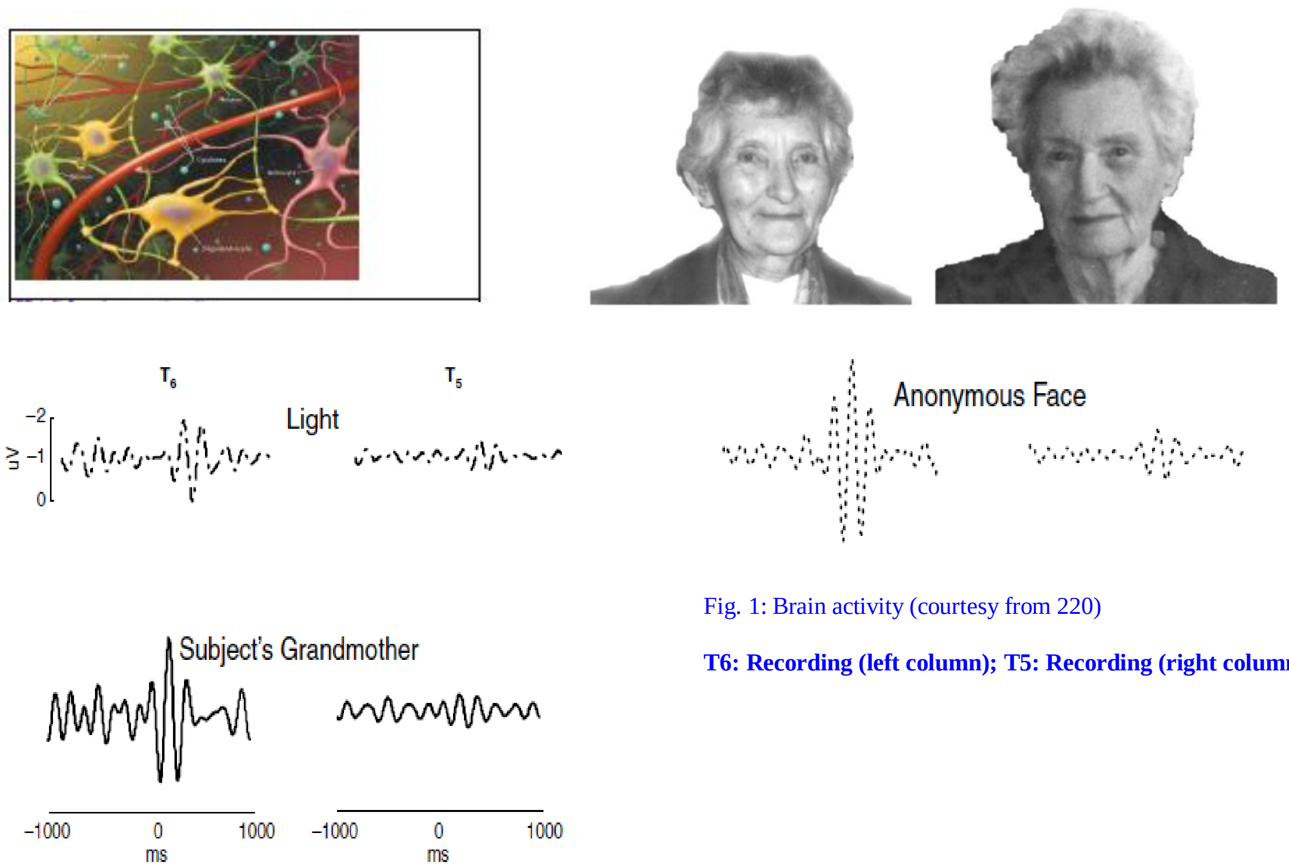


Fig. 1: Brain activity (courtesy from 220)

T6: Recording (left column); T5: Recording (right column);

2.2 Need for alternate approach in numerical interpolation

Interpolation (unlike extrapolation) of data was safe, respected and coveted for missing data, experimentally not available at a desired value (KB.1). The average of two or mean of multiple points was sought after for imputation over decades. Numerical interpolation techniques viz. Newton's forward formula, Newton's backward formula, Bessel, Lagrange, Atkin schemes based on the position of interpolated argument in the data sequence won laurels in1960s. We reported [1] calculation of stability constants of biomolecules using Lagrange interpolation with knowledge base (KB) in first order logic implemented in Fortran-IV. Rational polynomials were applied in calculating dielectric constant at a required % composition of co-solvent in aquo-organic mixtures for interpretation of solute-solvent interactions of protonic equilibria of neuro-transmitters. Cubic spline interpolation method was another well nurtured procedure. Broomhead,interested in interpolation of complicated functions, proposed RBF_NN [48] in 1988 for interpolation in multi-dimensional functions. Rao et al. [59] made use of RBF_NN in the calibration (regression) task of unicomponent-uniresponse-first order instrumental data for labetalol hydrochloride. Rao et al. [1]reportedmulti_component–multi_response (MComp.MResp.) calibration with RBF_ /ARTMAP_ NN models for non_Nerstian ion-selective electrode data.RBF_NN was a milestone in multi-dimensional function interpolation techniques.

KB. 1: RBF_NN for interpolation

If	Data is with noise
Then	Approximate solution viablethan exact solution
If	Solution for exact interpolation is found & No noise
Then	profile passes through every data point
If	Solution for exact interpolation is found & Noise is present
Then	Function oscillates between given data points

A search on science direct resulted in about 2700 publications on RBF(Chart 1). The abstracts of 400 preliminarily scrutinized list is run with are keywords generator matlab function files [1] developed in this lab. The short listed set of papers was sorted keyword wise. The two hundred and odd references cited in this review in the journal format are outputted by m-programs. Typical tables and figures are prepared from around two hundred full papers consulted since 1991. The multi-keyword set for SOM, RecNN, ART-ARTMAP, RBF, MLP without losing the flavor of full titles reflecting method_base and developments [1b] will be published separately.

Chart 1: Search strategy (Science Direct)

pub-date > 1989 &

TITLE-ABSTR-KEY (radial basis function networks) or

TITLE-ABSTR-KEY(RBF)

Search results : **2,704**

Year	#	Year	#
2014	206	2012	241
2013	264	2011	232
2010	180		
…..		.....	

Searched on 10th July, 2014

2.3 Architecture, Neuron and Transfer Function (Ant)

Architecture

The architecture is how a group of neurons are (inter-/intra-) connected. Changes of connection with different magnitudes of weights (-1 to +1 through 1) produce activation/inhibition of varying (zero to 100%) synaptic strengths in the language of biological neural networks. The transfer functions, in fact all most all known mathematical (polynomial/ exponential/ transcendental/ wavelet/ ridgelet/ statistical/ fuzzy/ chaotic/) coordinate-free system ofClifford or geometric algebra, quaternion functions alone and/or their hybrids produce profiles of wide spectrum of output of NNs. The real-, complex-, andquaternion-valued NNs are special cases of the geometric_algebra_m-D-NNs. The support multi_vectormachines (SmVMs)render the task simpler in generating RBFs for neuro-computing (NNs) and automatic finding of optimal parameters. The very type [binary, floating point, complex, quaternion, character, strings, images (pixels, voxels)] of input data, connection_weights produce different types of information. The confluence operator for the input and weight to a neuron opened vistas in different frames (Boolean, algebraic, fuzzy etc.). The distance measures (Euclidean, Mahalanobis, Manhattan, city_block etc) are crucial in looking at similarity/dissimilarity of data**.**

Neuron

An artificial neuron (processing unit) transforms the data.

Transfer Function

The heuristic or mathematical function used in this transformation is called a transfer function (TF) in mathematical parlance. This is also called an activation function (AF) from neurobiological stand point.Point or axis transformations in Euclidean or polar space brings forth not only interesting but also useful output. Eigen, principal component, varimax-factor spaces transform correlated x-variables into orthogonal ones. Many popular mathematical functions have been used as transfer functions (TFs) in artificial neurons of artificial-NNs. In the simplest case, the input/output of a TF is scalar. It may be as simple as multiplication by unity (do[ing] nothing) and is used in the neurons of input layer of SLP, ART etc. The other TFs are linear, polynomial (quadratic, cubic etc.), hyperbolic (tanh, sigmoid), kernel (RBF), wavelet, higher order (tensorial) algebraic, Clifford geometric or fuzzy-membership functions. The classification of TFs is from different points of view viz. local/global, static/adaptive or dynamic, zero-/first-/higher-order, real /imaginary [complex, quaternions or Clifford], linear/non-linear, polynomial/exponential/transcendental etc.Kernel function (Gaussian) belongs to the local categoryand are also good candidates as TFs in NNs for classification task trained with BP algorithm. The non-local basis functions are sub divided into positive definite (Z log Z) where Z= $|| x-\mu ||$ and non-positive definite (SPLINE).In recent times, combination (product) of transfer functions is also in practice.

RBF-NN

NN	TF in H1	TF in H2	Output	NN-$$
SLP	RBF	---	Linear	RBF_lin
	RBF	---	Sigmoid	RBF_sig
	RBF	Sigmoid	Linear	RBF-sigmoid_lin

Architecture. RBF_NNRBF-NN has a three-layered feed forward fully connected architecture i.e. it is SLP-NN with radial basis function as TF in the hidden layer, called radial layer. An object oriented representation of RBF-NN was discussed earlier [1e]. But a four layer RBF is recently proposed with radial basis transfer function in the first hidden layer and sigmoid in the second hidden layer. The input and output of RBF_NN is real (integer, floating point), Boolean (binary) and complex (quaternion, Clifford algebra) variables.

Input layer. RBF_NN:The number of neurons in the input layer corresponds to the explanatory/ causative /independent variables.

Radial Neuron A radial neuron (chart 2) uses radial basis function defined by the center, radius and (type of) distance measure as a transfer/activation function. The radial units are used in the hidden (second) layer of Kohonen-, Prob- and generalized regression (Gen.Reg-) NNs. A neuron with binary RBF_TF is called binary-RBF neuron. It is at least as powerful as linear threshold neuron [196]. Depending upon distance used, binary-generalized-RBF-neuron and binary-Euclidean-RBF-neuron are proposed.

	Chart 2. Neuron_RBF			
input	Confluence operator	TF	out	output
	$dist\;\; inp, center, width$	kernel	scalar	

Radial_layer.RBF_NN

The radial layer contains radial neurons. In essence, it differs from SLP-NN in that sigmoid TF is used in hidden layer.The individual neurons perform non-linear transformation and layer as a whole maps input into higher/lower dimensional space.

Output of radial_neuron (Processing Element, PE) in radial_layer

The output of a processing element (PE) or neuron in radial layer of RBF_NN is equal to the radial distance between each input (pattern) vector and center of the Kernel function.

Output_ layer.RBF_NN

Each of response /dependent variables forms a processing unit in output layer. Generally, the output neurons perform linear combination of outputs of hidden neurons, as a linear TF is default. But, with sigmoid TF, another non-linear transformation of S shape is obtained. This is advantageously employed in binary classification by round-off operation of output of sigmoid TF.

2.4 Distance measure in RBF:The distance measure for pair wise similarity and/or dissimilarity is widely used in machine learning algorithms Although, Euclidian distance is generally employed, other measures like Mahalanobis and city-block found place with added advantages for real life data processing. Recently, non-linear extensions of linear metric learning methods are proposed, but with limited forms of distance metrics and similarity constraints. Baghshah [179] reported a non-linear metric learning method in the context of synthetic digit/letter data analysis. This method learns a completely flexible distance metric via a non-parametric kernel metrics using both similarity and dissimilarity constraints.

Chart 3: Learning through NNs

2.5 Functioning of RBF: The activation function of neurons in input layer and WIR (weight tensor connecting neurons in input to RB layer) are identity matrices resulting in do nothing type of operation. In other words, the input layer transmits the pattern to the radial layer as it is. The output of the radial basis layer after modulating is passed to the output layer.

```
Learning till satisfied
    Iterate until converged
        If convergence criteria not met
            Modify/correct/rectify/repair
        Else converged
        Endif
    endIterate
satisfied
```

2.6 Training RBF**:**The learning inRBF_NN is semi-parametric, since both unsupervised and supervised approaches are combined. The determination of centers of Gaussian units is self-organizing as there is no specific target value for each training vector. The density modeling motivates this unsupervised clustering. Training weights of connection between radial and output layer neurons (WRO) is an optimization problem (Chart 3). The estimation of WRO (Appendix-A2) is a supervised parametric task as the response is available to compute residual (sometimes referred as error) function.Pseudo inverse, an unambiguous method, is used in training WBO of RBF_NN.Here, two-phase and three-phase training procedures (vide infra) are popular.

- Two-phase learning

In two phase learning procedure, centers and width of RBFs and WROs are learnt sequentially (chart 4).

Chart 4: Sequential multiphase learning/training in RBF_NN

Input

Primary Data

Input of data (X, y)

User Chosen

Number of radial layer neurons

Training of RBF_NN

Phase I	Calculate centers and deviation of RBFs
Phase II	Estimate WRO
Phase III	Simultaneous refinement of centers, deviations and WROs

$$y\_RL = \left\{ TF\_RBL\left[\quad scal \quad x^T \quad , centers, width \right] \right\} \quad \text{Eqn. 01}$$

$$y\_OL = TF\_\mathrm{OL}\left[\quad y\_RL^{\;T} * WRO \right] \quad \text{Eqn. 02}$$

$$y\_\mathrm{RBF} = [unscal(y\_OL)] \quad \text{Eqn. 03}$$

Alg. RBF-01a : Training-Phase-I

Input

Phase I

Step	:	1	Calculation of centers of RL neurons
Step	:	2	Calculation of widths of RL neurons
Step	:	3	Calculate output of RL neurons

Output

Center, Width vectors for RBFs

Alg 01b: Two phase learning

Input

Phase I

Phase II

- Calculate Ws between RL and OL by [batch algorithm : pseudo inverse, online : BP, CG etc.]

Output

- yRBF, [WRO]

Alg. 01c: Phase-III learning – RBF_NN	
Input :	[Centers, width], [WRO], from Phase II
▶ Simultaneous training/optimization of [Centers, width], [WRO]	
Output:	Refined values of [Centers, width], [WRO], yRBF

Phase I

Input is the unsupervised form of data. Further, ifsome of measured values do not have response, they can also be used in this phase (Alg. 1a).

Center of RBF:During the first phase, centers of radial units of Kernel function are calculatedby unsupervised clustering viz. sample or k-means procedures(Fig.2, Appendix-A3).

Width (or deviation) of RBF:The deviation is prefixed, user chosen or determined by isotropic, explicit or k-nearest neighbor algorithms(Fig.2).

Alg.2: Optimum RL neurons-RBF_NN

Input **(Fig.2)**
NRLneurons = NP

Do until convergence
For i= 1 : NRLneurons
Two phase training **(Alg.01b)**
If ESS(y) < Tol, converged,
else not-converged
endif
endfor
endDO

Phase II: In the simple case, the output is calculated using linear postsynaptic processing (PSP) with the already refined centers and σ^2. The optimal values of WRO matrix, (connection weights between radial layer and output layer) are calculated by the supervised regression procedures with a chosen object (minimization of error) function (Alg. 1b).

Traditional fast optimization techniques (like pseudo inverse, BFGS) are in wide spread use. The output model consists y_RBF (y calculated with RBF_NN), refined WRO, co-ordinates of centers, one width for each of RBL neurons, explicit form of RBF and distance used. Support Vector (SV) learning in RBF is a special type of one phase model and leads to complex network structure.

▶ Three-phase learning

In three phase learning mode (Alg. 1c), the results of two-phase learning viz., centers, deviations, WRO are employed as approximate/initial model parameters. B P like procedure is used to refine the whole set parameters. With this approach, there is a substantial improvement in the performance of classification task. Schwenker [132]studied two-, three- and support vector learning in RBF for classification tasks. Montazer[182] proposed a three-phase learning algorithm which optimizes the functionality of the optimum steepest descent method and applied to RBF training.

But, a straight forward simultaneous refinement task of weights, number radial layer neurons (PEs), centers, deviation and distance function is hard problem and results in non-differentiable optimization function. Natural computational techniques (vide infra) have been recently proposed[125]with notable success.

- Software for RBF_NN

Default and user chosen parameters
Except in complete automatic design of RBF architecture, number of radial layer neurons, centers, deviation (spread) and distance measure are user chosen or default in RBF_NN.A simple algorithm for selecting minimum number of neurons is given Alg. 2.

Intelligent Problem Solver (IPS) of Trajan (Fig.2): It builds a set of models using heuristics in varying the number of centers, deviation and the methods of calculation. The best architecture along with the entire set is outputted. The positive features and limitations of RBF_NN in real life tasks are described in chart 5.

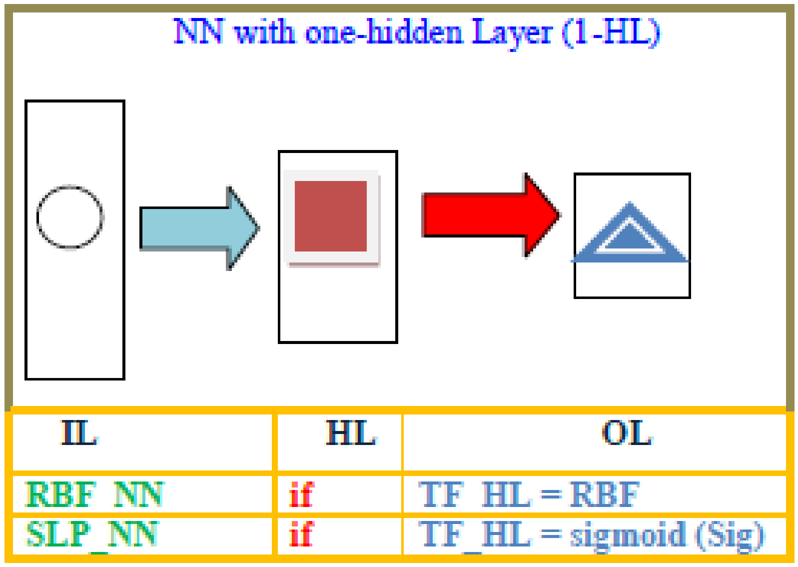
Chart 5: Positive features of RBF_NN

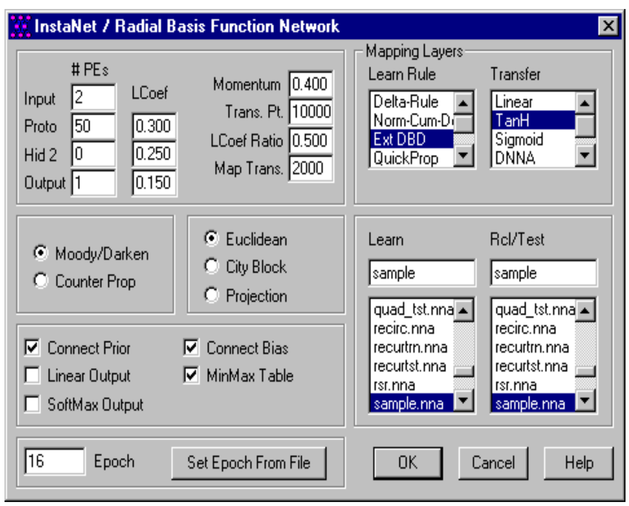
+ Simple topological structure
+ Applicability of universal function approximation theorem
+ Trend leaning of data patterns faster than MLP_NN
+ Numbers of radial neurons do not scale up with number of variables/patterns

Limitations of RBF in applications to real life tasks

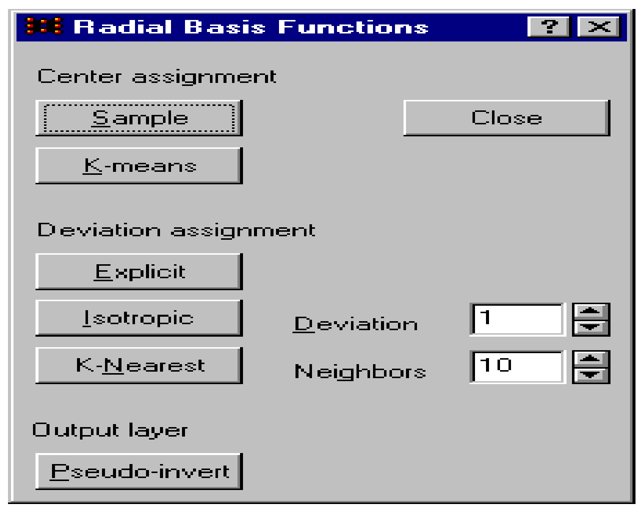
- If curve representing training patterns is nearly constant in a specific interval, RBF is inefficient, unless width tends to infinity.
- RBF is not appropriate, ifpairwise distances between patterns are nearly equal in high-dimensional space. In other words distances to nearest and furthest neighbors are same..

⇨ **Remedy:**RBF with flat profile

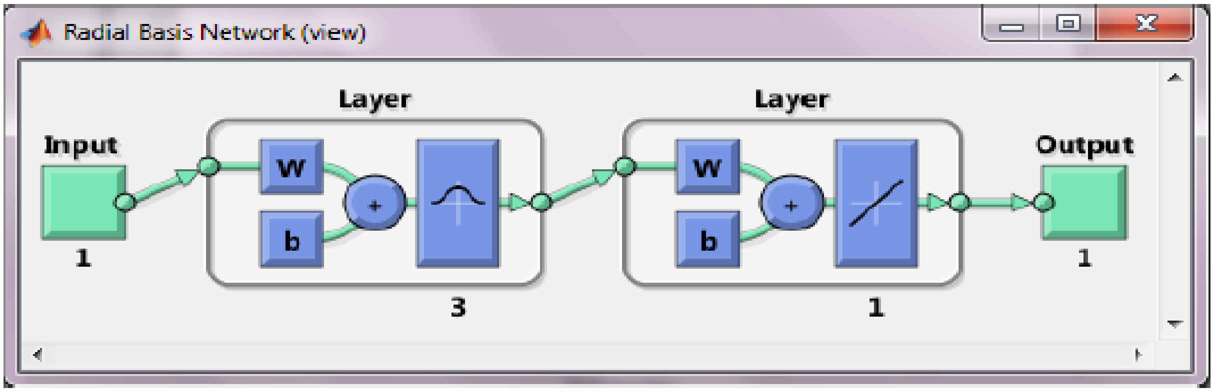




Professional II



Trajan



MATLAB_NN toolbox

Fig. 2: Graphics user interface (GUI) for Input and method choice in commercial software

2.7 Illustration

➪ Dataset.Classification. XOR.RBF_NN

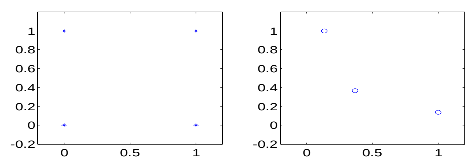
RBF with architecture 2-2-1 models XOR gate. It is a fact that the outputs of XOR are not linearly separable (Table 1). The outputs of the two radial neurons plotted (Table 1, Fig b) are linearly separable.

Table 1: output of Radial-basis-layer of RBF_NN with two RB-neurons for XOR gate

input		*output*	
		RB1	*RB2*
0	*0*	*0.1353*	*1.0000*
1	*0*	*0.3679*	*0.3679*
0	*1*	*0.3679*	*0.3679*
1	*1*	*1.0000*	*0.1353*

Architecture : RBF (2-2-1)

centers RB neurons : [1,1]
[0,0]



(a) *(b)*

3. Applications of RBF_NNs

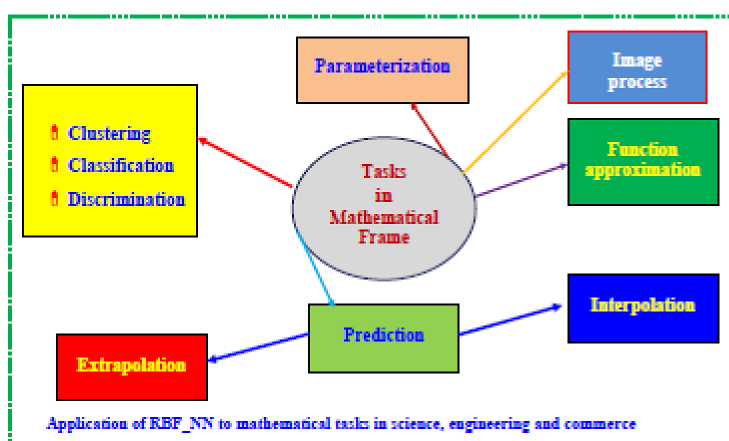


Fig 3.

The planet earth is a smart system with surrounding atmosphere for sustenance of life --micro-organism to human beings / animal kingdom. The evolution over a period of time, paved way for healthy living with safety/security/comforts and wisdom. From mid twentieth century, smarter devices includingcomputers weremanufactured and the attention was directedto smartmaterials, which adapt themselves to changing scenarios. IBM recently committed for high performance systems, storage and cognitive computing at a higher level resorting for alternatives to silicon technology. Materials consist of chemical elements and/or their compounds in different (gaseous, liquid, solution, solid, and plasma) states/ phases. The functional characteristics arise due to chemical structure, isomerism, number of moieties in a unit/unit_volume (nano- to macro-molecular assemblies), variation of temperature, pressure, electromagnetic radiation, charge, gravity and so on. An in depth study of these variationsare means for understanding sub micro- to mega- systems. In the materialistic world, nature already achieved eco-balance over longer period of time. Further, life-adaption promoted the bio-synthesis of compounds in animal/plant kingdom or man prepared them first at laboratory and then adaptable/upgradable industrial scaleas per demand. The natural instinct is to have materials of desired characteristics and combat with undesired ones. On a unified scale, it is all transfer of energy and discipline wise nomenclature goes as chemical, physical, nuclear, biological, geological, wind and sub-particle (inter-/intra-) interactions.

It is not an easy task for applied scientist to run a mile to adapt a new mathematical and/or nature mimicking algorithm as soon as it is proposed. Yet, if one goes the extra mile with available software, renaissance in not only accuracy but also robustness of computational procedures in applied disciplines is not a million mile from here.

RBF_NN has been popular in supervised classification, regression, function approximation [19] and prediction of dynamic systems including time series data (Fig. 3). The chemical applications include multi component analysis (MCA) of spectral data, modeling NMR chemical shifts [40]and process control. However it has an edge for classification tasks as the radial layer essentially functions like Kohonen mapping and the other layer is in a supervised mode. This network found astounding success even in function approximation of complex multidimensional ones. RBF_NN [185] is used to estimate the unknown continuous function. A non-linear system with input with delayed feedback from hidden layer is converted to a system without delayed feedback. Interpolation is possible after mapping on to a 2D space.

A synopsis of applications of RBF_NN in Medicino _metrics, Bio _metrics, Chemo _metrics, Pisci _metrics, Econo _metrics, Techno _metrics, Informato_metrics, software _metrics, Biblio _metrics, performance _metrics and Metrics _metrics under the cover of omni_metrics follow.

3.1 Chemometrics (Chemistry + o + me+tri+cs)

Omnimetrics [1b] is a hybrid paradigm of matured application based disciplines and measurement science associated with software/hardware implementation of mathematical/ statistical/ possibility/ information_based and nature_inspired algorithms. The applications using RBF_NN in Enviro _metrics, Pharmaco_metrics(Pharmacy + o + me+trics), Specio _metrics , Kineto _metrics, Dieteto _metrics , Nano _metrics, Quali _metrics are focus of chemometrics. Welczek and Massart [40] reported that RBF_NN performs function approximation, multi component analysis, time series and supervised classification to the desired degree of precision and accuracy in chemistry.In chemical sciences the latest trend is consideration of (thousands of) molecular descriptors in understanding (modeling) and predicting the not easily measurable response characteristics of virtual to real life systems. In the experimental front, self-organizing processes viz., crystallization, dewetting and phase segregation result in defective molecular films. But these processes have tremendous technological applications. The outcome of stamp-assisted deposition to control size, density, positions of material assemblies results in new chemical/physical functionalities.

Table 2: Best QSAR model with Q2 statistic

NN	Method-II	Q2
RBF	RM	**0.9766**
RBF	K-means	0.7965
RBF	Cluster Analysis	0.7084

Table 3: Struct. Prop.Rel

Method	RMSE
MLR	12.58
PLSR	12.58
RBF	**8.21**

➪ **SXR (structure X relationships)**

Malek-Khatabi et al. [39]applied RBF_NN along with replacement method (RM) to predict activities of a series of 1-[2-hydroxyethoxy-methyl]-6-(phenylthio) thymine] (HEPT) derivatives, as non-nucleoside reverse transcriptase inhibitors (NNRTIs). This QSAR model (Table 2) chooses the informative centers reducing volume of data. It increases the prediction ability without pruning descriptor space for HEPT, Selwood and GABA data sets. Xing [38] developed a QSAR model for antimicrobial agents against Candida albicans using PSO induced classification tree. The modified_PSO was used to induce a globally optimal classification tree (CT) via simultaneously searching the optimal splitting variables, their values and structure of the tree. Of the hundred molecular descriptors from Material Studio 4.0, the descriptors viz.HOMO eigenvalue, Dipole_Y, S_ssCH2, S_dsCH, S_dO, AlogP98, Molecular flexibility, Wiener index, Kappa-2, Subgraph counts (2): path, Subgraph counts (3): cluster, Chi (0), Bond information content, Principal moment of inertia X, Ellipsoidal volume, Shadow_YZ, Shadow_ZX, total molecular mass, and atom count are influential explanatory variables. The results from CT, RBF_NN, PLS_Disc.Anal and mod.PSO are compared. It isreported that classification of antimicrobial compounds with classification tree configured by mod_PSO using distance-edge vector index as a molecular descriptorwas developed for polybrominated diphenyl ethers.

Chart 6: Contrast between ellipsoidal and spheroidal RBFs

RBF		
RBF parameters	**Ellipsoidal**	**Spheroidal**
Width matrix	+ Varying widths for different input variables	- Identical width for all input variables.
Spread of Different input variables	+ Takes care of variability in the spread of different input variables, + Ellipsoidal data clusters modeled with very few ellipsoidal RBF neurons	- Ellipsoidal data clusters split into multiple spherical ones → - Modeled with large number of spheroidal_ RBF_ neurons - More complicated model - Increased risk of over fitting

⇨ Struct. Prop.Rel.

Jiao [37] formulated quantitative relation between molecular distance-edge vector (MDEV) index and relative retention time (RRT) of 126 polybrominated diphenyl ethers using MLR, PLSR and RBF_NN (Table 3).

ॐ Physico-chemical parameters

Tetteh et al.[41] developed RBF_NN model for simultaneous estimation of flash point (Tf) and boiling point (Tb) of 400 compounds with nonhomogeneous functional groups. Twenty fivemolecular functional groups and their first-order molecular connectivity index are used as structure representing variables.Zhou et al. [174] reported PSO forsimultaneous optimization ofarchitecture, centers,widths and Ws of RBF_NN with ellipsoidal Gaussian transfer function (Chart 6, Alg. 3). Continuous-PSO was used for refining Ws and modified discrete PSO for fine tuning network topology. The particle in PSO is a real number string corresponding to centers, widths and Ws of RB layer neurons. In modified discrete PSO, binary stringwith values zero or one refer to number of Ws, centers and widths. If WRO for jth neuron is set to zero, it means that it is pruned. The comparative study with other methods (table 4) shows that this hybrid algorithm is better than component procedures.

Alg. 3: Hybrid PSO_RBF_NN

Initiation of parameters

- PSO
- RBF_NN

Iterate Until convergence | maxIter

Cal Hadamard product of PAR and STR

Cal fitFnValue

Refine PAR ofpopulation

Update bits in STR

bits_center ←1

bits_widths ←1

End iterate

Table 4: Efficiency of different models for QSAR study

Statistics	PLS	PSO+ RBF_NN	k-means + regularization	[Disc Conti] PSO +RBF_NN
ESS	0.58	0.43	0.42	0.35
R	0.66	0.84	0.85	0.89

If AccCompl larger
Then Network simple
- local minima
- Divergence

If AccCompl small
Then Fast convergence
- Over fitting

AccCompl : Trade-off-between complexity and accuracy

default_value : 0.1

$$objFn = RMSE * \left[1 + AccCompl * \frac{no\_Wts > 0}{total\_no\_Wts}\right]$$

3.2 Dietetometrics (Dietetics + o + metrics)

The pivotal importance in sensory evaluation of beverages, fruit juices, food products etc. has a niche in the annals of analysis since times immemorial. The era of panel experts (tea tasters, wine tasters) with strict life style and rigorous training has given way to advanced soft_chemometric_methods, commercial products like E-tongue and E-nose. Yet, indispensable human role still persists to improve the high quality finished products, counter checking information from machine learning and take advantages of experts' individual innovative/unknown sparkles from miraculous integration in the brain.The concerted effort of fundamental researchers and governmental agencies including FDA (USA) is a beacon light in imposing stringent stipulations on food quality and safety to mitigate spread of epidemics. More than 3.2 million litres of vinegar is consumed [229] every day in China. The state-of-the-art-techniques and commercial promotion is one side of the coin. Dietetometrics (Chart 7a), a hybrid cross-discipline, is another reflection of well nurtured metrics on foods and beverages amenable for human consumption during healthy and post-surgical period.It avoids (in a trust worthy mode) the wreckage of ship of suggestions in the ocean of data/information/knowledge with eye catching multi-true-color multi-dimensional visuals and attention drawing verbal schizophrenia.

Chart 7(a): Food and health

- **Dietetics**
  - Study of nutrition related to health
  - **Sitology**
    - Food preparation and intake
- **Macrobiotics**
  - Theory of promoting health and longevity
    - Whole beans and grains
      - Regulation of food intake and preparation

Chen et al. [234]reviewed applications of imaging (hyperspectral-, magnetic resonance- , soft X-ray- , ultrasound-, thermal- , fluorescence- , odor-)techniques in detection of food quality and safety by non-destructive approaches. The cutting edge advantages of the present trends in ensuring the production of high safe quality foods with enhanced taste and flavor are discussed. The added dimension is ensuring natural aroma of food product retained deep in the memory of consumer passed on through genotype and phenotype information. The technical challenges of today are briefly summarized guiding future scenario projections.

The geographical genotype classification of Arabica coffee [36] and meat spoilage [101] is studied with RBF_NN using FT_IR spectral data. RBF_SVR [116] model detected of apple disease was reported.

Calibration_ditetometrics

ॐ NIR

➡ Grain protein. MVC_

Lin et al.[62] applied RBF_NN in MVC of grain protein content in barley using NIR. The first order derivative of spectrum is computed followed by normalization of response. It not only enhanced signal-to-noise ratio but also reduced background /baseline shifts. Linear (Lin-) Least squares (LS) support vector regression (SVR) [Lin_LSSVR] and RBF_LSSVR models exhibited higher accuracy compared to PLS and RBF_NN(Chart 7b). The results are applicable in quality control in feed processing, breeding selection, and malting.

Chart 7b: RBF_NN for NIR data

Step 1:
- The best spectral window (4100–7750 cm−1) was obtained with PLS.
- This NIR spectral region was standardized to Z-transform.
- The first derivative spectrum was calculated

Step 2
- PLSR and RBF_NN models were developed

➡ NIR-wheat

Mao et al. [108] reported RBF_NN estimates of the protein content in wheat from NIR spectra. PSO algorithm is used to minimize number of RB neurons and also to avoid premature convergence. The low RMSEP (0.265) is indicator of adequacy of this procedure and is better than conventional semimicro-Kjeldahl method in practice for testing in agricultural fields.

➡ NIR spectra

Chen et al. [230] studied quantification of total polysaccharides and triterpenoidsin Ganoderma lucidum and Ganoderma atrumoriginating from a variety of geological areas by soft-models and NNs from NIR spectral measurements. NIR with chemometric computational models are adaptable for routine monitoring of chemical composition in G. lucidum and G. atrum.

➡ UV-Visible spectra

Ni et al. [231] studied the quantitative estimation of four food additives with overlapping UV_Vis profiles, although their individual profiles are well defined. Pre-separationis indispensable by classical calibration procedures. The chemometric procedures permit multi-component mixture calibration followed by prediction in real life samples.Here, the outcome with synthetic solutions of these four compounds(chart 8) established the feasibility of the procedures. The results show RBF_NN successfully models multi_component-multi_response-first_order-calibration datasets. Typical chemometric mathematical techniques viz. PC_RBF_NN, PLS and PCR are used in the kinetic spectrophotometric simultaneous estimation of maltol and ethyl maltol based on the formation of a pink colored complex (having absorption maximum at 524 nm) with iron(III) and o-phenanthroline in sulfuric acid medium. The full spectrum in the range 370-900nm for 0-30 sec adhered to Beer's law in the range of 4.0–76.0 mg L^{-1}. The results of these chemometric approach yielded equivalent results to HPLC_UV method.

Chart 8: Multi_component multi_response- first_order-calibration models

Compound		LOD (mg l^{-1})
Maltol	:	0.39
ethyl maltol	:	0.56
vanillin	:	0.49 and
ethyl vanillin	:	0.38

UV Spectra : 200–350 nm

RBF_NN >
[PLS, PCR, DPLS DPCR] >>
CLS and DCLS

Information bits (IBs)
- CLS and DCLS perform poorer for mutli-component – multi_response (MIR) data
- No advantage of using first-/ second derivatives of spectra

Abbasi-Tarighat et al. [232] employed RBF and MLP NNs in simultaneous estimation of Fe(III) and Mn(II) in foods, vegetable and water samples. The spectrophotometric data of the complexes (ML, L_2) of these metal ions with 4,4′[(4-cholorophenyl)methylene] bis(3-methyl-1-phenyl-1H-pyrazol-5-ol) at 6.7 pH showed a calibration range of 0.20–7.5 for Fe(III) and 0.30–9.0 for Mn(II) mg l^{-1}. This chemometric calibration tool is applied to the simultaneous determination of metal ions in different water, tablet, rice, tea leaves, tomato, cabbage and lettuce samples.

Classification_ditetometrics

Wan and Harrington [106] studied classification of poly(chlorobiphenyl) (PCBs) and Italian olive oil by RBF_NN. The mass spectra of PCBs containing one to nine chlorine atoms with 161 training patterns are used. The source regions of olive oil are classified using quantities of eight fatty acids (Chart 9). The centroids and radii of RBFs were calculated with linear-averaging algorithm with better performance compared to k-means. Central composite design was used to study the effects of number of RBL neurons and threshold magnitude.

Chart 9: Classification of PCBs and Italian olive oils

W estimation	QuickProp SVD
System	**% prediction accuracy**
PCB	>94
Italian olive oil	100

Food_toxicology

Dataset.Microbiology.q_RBF_NN:The food borne pathogen (Staphylococcus aureus)causes several ill effects on human health. The categorical classification dataset (Chart 10) is analyzed with q-RBF_NN [94] using synthetic minority over-sampling technique (SMOTE, Alg. 4) algorithm. SMOTE is designed for tasks where minority class is of vital interest. It is clear that SMOTE_qRBF_NN excelled standard SVM, logistic regression, classical RBF etc. This is a method of long self-life and deserves further research and knowledge driven mode of operation.

Chart 10: Categorical classification (categorical) of Staphylococcus aureus growth

Influencing Variables (X)	Output y
Temperature [8, 10, 13, 16, 19°C] pH [4.5 to 7.5 (0.5 intervals)] Water activity 19 points [0.856 to 0.999]	Prob(pertaining to one class)

If Prob(growth) =1
Then Growth

If Prob(growth) =0
Then No_Growth

If Prob(growth) ~= 0 & Prob(growth) ~=1
Then Growth_transition

Model	**% Correct classification rate**
MLogistic (standard model)	76.60
SMOTE+MLogistic (standard model)	71.63
RBFN	75.18
SVM	80.98
MQRBF	80.31 ±3.34
SmoteOverSampling	75.60 ±4.03
SMQRBF	**82.77 ±1.90**

Alg. 4: Smote Memetic alg. for QRBF

Data set
Pre-processing
 Over sampling procedure
Generate initial population
Evaluate candidate solutions
While not converged
 % operate Memetic Alg.
 Cross over
 Mutation
 Structural
 parameter
 Evaluate population
 If optimized, Create clusters
 IfIrprop+
 over representative
 Then Continue iteration
End while
Best **Smote Memetic** _QRBF

If input data exhibit elliptical & normalized
Then normalized input follows q-Gaussian probability law

A. Valero, F. Prez-Rodríguez, E. Carrasco, J.M. García-Alventosa, R.M. García-Gimeno, G. Zurera, Modelling the growth boundaries of Staphylococcus aureus: effect of temperature, pH and water activity, Inter. J. Food Microbiology**2009**, 133 (1–2), 186–194.

3.3 Envirometrics (Environment + o + m+ e+ trics)

Still, combustion of fossil fuels viz. coal, petroleum, and natural gas is in wide spread use in domestic as well as industrial sectors. The unmanaged CO_2 escaping into local atmosphere resulted in global ill effects on the health of environment. CO_2 capture and storage is to reduce pollution of atmosphere and save environment from global warming and decreasing UV-radiation reaching the surface of earth. One of the recommendations for mitigation was capturing CO_2 from fossil fuel-fired power plants. H_2 production and storage is an alternate source for fuel and energy. Methane nodules under ocean also play a key role in these pursuits.

Rain fall:Wu et al. [147] reported combination of elitist/GA/ PSOstrategies and RBF_NN in the prediction of rainfall with superior results compared to any of the individual approaches. The highlight of this hybridization is automatic selection of number of radial layer neurons, centers and widths of RBFs. In every generation, the members of population are sorted in increasing fitness criteria. The upper 50% of best performing solutions are put in elite category and lower half of worst performance is called swarm. The selection, crossover and mutation are operated on elites which are the prospering carriers of information in the next generation.

- Global exploration ability
- Avoids premature convergence

Wind speed: Wang et al. [87] introduced a hybrid model consisting exponential smoothing (ESM), seasonal adjustment (SAM) and RBF_NN to predict from hourly data at two meteorological stations in the Hexi Corridor of China. It is superior to Holt–Winters model (HWM), MLP_NN, ESM, RBF_NN, and binary hybrid models viz. SAM+ESM, SAM+RBF_NN and ESM+RBFN.

Thermal conductivity (TCs): The effective Thermal conductivity (TCs) of dry and oil saturated sandstone at a wide range of environmental conditions was modeled with MLP, RBF, Gen.Reg. and Cascade_FF-NN. Temperature, pressure, porosity, bulk density of rock, fluid density and oil saturation are employed as independent variables. The absolute average relative prediction accuracies for saturated and dry sandstone are 2.73% and 3.81% with MLP architecture of I#-[7-15]-O#.

Porosity carbonate fields: Ansari et al. [95] developed a multiphase intelligent computational tool to predict porosity of Iranian carbonate fields in the Persian Gulf. Seismic colored inversion (SCI) technique was used to calculate sample based attributes from 3D seismic volume. The prediction of porosity log is done from seismic attributes using Prob_NN, MLP_NN, RBF_NN, e-SVR and ANFIS models. These results are combined in a power law committee machine (with empirical competitive algorithm) which outputs a single but improved solution.

CO_2 absorption:Fu et al. [228] used RBF_NN and MLP_NN in prediction of the mass-transfer performance of CO_2 absorption into aqueous mono-ethanolamine (MEA) in different packed columns. The inputs and outputs of the system are in chart 11. Based on average absolute deviation (AAD),root-mean-square of percentage error (RMSPE), the performance of RBF_NN was found to be than MLP_NN.

Chart 11: Prediction of CO_2 absorption by RBF_NN

Input	Packed Columns	Output –Targeted_output mass transfer variables
▶ Inert gas flow rate ▶ Liquid flow rate ▶ Solution concentration ▶ Liquid CO_2 loading ▶ CO_2 mole fraction ▶ Temperature ▶ Total packing area etc.	▶ Berl saddles ▶ Pall rings ▶ IMTP random packing ▶ 4A gempack ▶ Sulzer DX structured packing	▶ Volumetric mass flux ▶ CO_2 mole fraction ▶ Temperature profiles along height of packed column

RBF_NN + GA: Ge et al. [47] proposed hybridization of RBF_NN with GA in developing optimal design of dividing wall column (DWC) in chemical engineering tasks. The simulation data is used in the optimum architecture of RBF_NN and GA in arriving at minimum stage numbers of DWC.

Petroleum Engineering

Bagheripour [97] made use of committee neural network to arrive at more reliable prediction of rock permeability of major carbonate reservoir rocks in Kangan and Dalan, South Pars Gas Field-Iran (chart 12). The outcome of committee of networks is superior to those of individual NNs.

Chart 12: Petroleum rock permeability in South Pars Gas Field-Iran

Alg.	Input	output	Remark
PCA	Petroleum well log data	PCs	Orthogonalization of variables
MLP RBF Gen. reg.NN	PCs	Predicted rock permeability	PCs extracted using MLP
Committee_of_ NNs	Output_of_NNs (350 outputs)	Consencious output	Test set : 245
GA			Assigns weight factor to each ANN

Electro-slag smelting

It produces high quality steel with uniform crystal density. The re-melting is a complex process for control with multi-variate distributed parameters and correlated influencing variables (chart 13). RBF_NN was trained by artificial fish swarm optimization algorithm (fig.4). Fish.Swarm.Alg performs cluster analysis of errors. The swarms of fish are distributed into every clusters of different error type and optimization is carried out. The centers and learning rate are refined based on these results followed by up gradation of WRO values. The multiple convergence criteria are SSE, RMSE, maximum positive error and maximum negative error.

Chart 13: Input and output of electro-slag smelting

Input factors

- Remelting current
- Remelting voltage
- Depth of the slag pool
- Flow rate
  - Cooling water
  - Outlet water
- Temperature
  - Cooling water
- Cone height of Consumable electrode

$$_RBF	Melting speed	Pure Coef.
ASFA	0.0028	3.34
Grad_	0.0150	7.23
OLS_	0.0104	5.25
GA_	0.0055	4.56
PSO_	0.0068	4.17
MLPNN	0.0163	6.16

Output

- Melting speed
- Cone purification coefficient (Pure. Coef) of consumable electrodes

Visual	:	Sensor distance of fish
Step	:	Max(moving step of fish)
Delta	:	Congestion degree

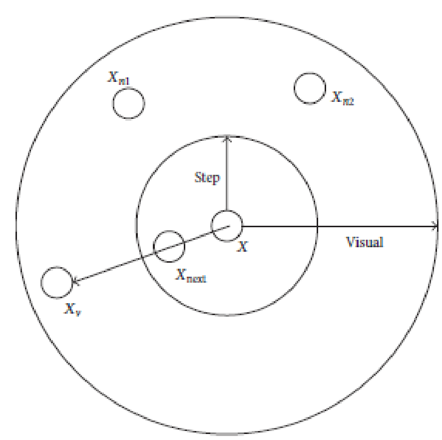


Fig 4: Fish moving step and its visual scope

Artificial fish swarm algorithm (Fish.Swarm.Alg) is another nature-inpired optimization approach mimicking foraging, chasing, random movement, leaping, and swarm behavior fish schools. The visual sensor behavior of artificial fish is adapted.

Max_positive_err = $\max \quad ycal - y \;,0$

Max_negative_err = $\min \quad ycal - y \;,0$

The collective behavior and swarm intelligence in nature evident in foraging ants/honeybees, birdsflying to unknown places, hunting and escape in prey-predator cycle etc

Fish.Swarm.Alg is hybridized with Langrangian, PSO, information theory, density function based fuzzy-c-means etc.

3.4 Medicinometrics (Medicine + o + metrics)

Themorbidity and mortality rate of patients of carcinoma, HIV and cardiac diseases draws continual attention of medical professionals and Governments all over the world [237]. Mostly, women with estrogen receptor-positive metabolism suffer from breast cancer. During post-menopausal period of women, the major source of estrogen production is through catalytic activity of aromatase (an enzyme) in the biosynthesis pathway. The aromatization of C19 androgens under the influence of aromatase produces C18 estrogens [239].The tumor progression is associated with high level of estrogen. Thus, inhibition of aromatase reduces the estrogen level and lowering the breast cancer instances. FDA_US approved aromatase inhibitors (AIs) as a first-line treatment for estrogen receptor-positive post-menopausal women and also in relapse cases of tamoxifen [240,241]. Since, it a promising target for breast cancer [238,240, 241], a large number of QSAR studies are published. The steroidal drugs snugly bind in the binding pocket, where as non-steroidal type binds the azole nitrogen. These drugs coordinate to the iron containing heme prosthetic group [242].

Chart 14: Electromyogram of Infrahyoid & suprahyoid muscle groups

	% accuracy	Method
VS	78.05	Neck muscle EMG
	65.24	Respiratory trace
F0	RMSE : 2.81 semitones	

▶ Medical diagnosis

Medical diagnosis comprises of classifying symptomatic observations and clinical tests to a category of diseases. The confirmative evidence is from non-invasive ultra-sound and MRIimages, clinical markertests and /or invasive procedures.

Mao et al. [223] proposed a hybrid RBF-NN for classification and applied to medical datasets viz. thyroid, breast cancer, EEG_eye state and diabetes (Table 5). The non-symmetric fuzzy means methodis used for automatic selection of number of clusters and their co-ordinates. Evolutionary simulating algorithm refines co-ordinates of centers in each generation. Then, RBF_NN performs classification task calculating weights by inversion procedure. This cycle is repeated until convergence criteria are satisfied or pre-fixed number of generations is completed.

Chart 15: Heart beat discrimination with RBF_NN+Honeybee alg.

Step 1: Stationary wavelet transform reduces noise in ECG data

Step 2 : The higher order statistics in ECG signals are extracted
Three timing interval features are also obtained.

Step3: Artificial Bess algorithm + RBF_NN discriminates five types of heart beats from ECG

⇨ EMG (electromyogram)

De Armas et al. [226] recorded EMG signal images from the infrahyoid and suprahyoid muscle groups(chart 14) and respiratory trace of 10 able-bodied adult males (18–60 years old) for three kinds of vocal tasks viz. tones, legatos and phrases. The results of predictionof fundamental frequency (F0) and voicing state (VS) using neck muscle EMG are of immense use in post-surgical period. The majority of laryngectomees use the electrolarynx as their primary mode of verbal communication after total laryngectomy surgery. However, the archetypal electrolarynx suffers from a monotonous tone and the inconvenience of requiring manual control.

⇨ ECG. [RBF + ABC]

Ebrahimzadeh et al. [225] reported combination of RBF_NN and honey bee algorithm to discriminate normal and abnormal heartbeats(Chart 15). The abnormal category consists of right bundle branch block, leftbundle branch block, premature ventricular contractions and atrial prematurecontractions. The system is tested with discrimination of 95.79% accurate for the 4000 beats in first dataset of MIT/BIH arrhythmia database. The overall performance for eight files is 95.18%.

Lin [78] used RBF to train signals and non-stationary patterns, step signals contaminated with Gaussian/symmetrically distributed non-Gaussian noise. The third order cumulants of the signals are given as the input to RBF. The higher order cumulants are the coefficients are the coefficients of Taylor series expansion of the joint characteristic function. The Ws are refined with a higher order spectral (HOS-) based leaning algorithm, which is superior to LMS under different noise levels. HOS-Based input results in superior performance of RBF and NLMS.

Table 5: Comparison of models for Diabetes and Cardiotocography data sets

Method		Accuracy % validation	
		Pima Indians Diabetes	Cardiotocography NSP
ESA-NSFM	:	82.8	95.1
SFM	:	80.2	94.0
SVM	:	78.1	92.9

NSP : Fetal state [Normal, Suspect, Pathologic]

SFM : Symmetric fuzzy-C-means

RBF : Thin plate spline function

⇨ Diagnosis_Cancer

Data. Gen.RBF.Evol-Navarro:The gene microarray datasets for six types of cancer are classified with Generalized-RBF_NN_Navarro (Alg. 4) approach[224]. It is a hybridization of evolutionary and gradient methods for arriving at architecture, node topology and weights for classification datasets. Thousands of genes in microarray data directly contribute to class membership of each pattern**.** Three filters viz. Fast Correlation-Based, Best Incremental Ranked Subset and Best Agglomerative RankedSubset are applied to pick up salient expression genes. The selected gene subsets are now used as input variables for classification. The results establish that this hybrid neural (RBF) classifier increased performance compared to other popular methods (Table 6,7).

Alg 4. Hybrid Gen.RBF.Evol-Navarro algorithm

Initiation
Random population of size NP

Repeat Until maxGen

While Not converged
Cal fitness fn
Sort in ascending order based on FitFnValue

10% of best members are taken Best10
Replicate them → RepBest10
Replace 10% worst members with RepBesst10

Operate mutation of parameters for pm% members
Operate structural mutation for (100-pm%) members
End while

Improved_Rprop_plus (Resilent BP +)
algorithm to the best solution obtained by EA
End repeatGen

Table 6: Number of genes and samples in diagnosis of cancer

Disease	# genes	# samples	Characteristics	RB_neurons	
				Min	Max
Breast cancer	24,481	97	46 : patients labeled as relapse 51 : non-relapse	1	3
CNS	7129	60	21 : Survivors 39 : failures	1	3
Colon	6500		40 : tumor 22 : normal colon tissue samples		
Leukaemia malignant neoplasms of haematopoietic stem cell	7129	72	47 : acute lymphobastic leukaemia 25 : acute myeloid leukaemia	1	3
Lung	12,600	203	139 : lung adenocarcinomas 21 : squamous cell carcinoma 20 : pulmonary carcinoid tumors 6 : small cell lung cancer cases 17 : normal lung samples	5	8
GCM	16,063	190	14: tumor varieties [11,10,11,11,22, 11,10,10,30,11, 11,11,11,20]	25	28

Table 7: Comparision of Gen.RBF_NN_ with standard classification techniques

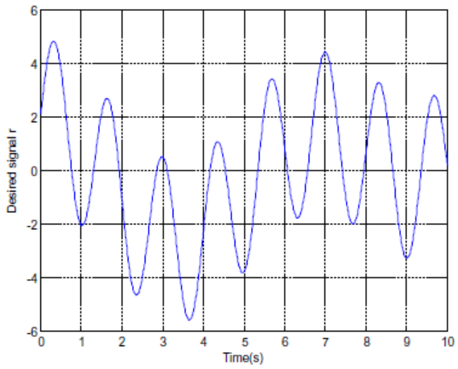
	Gen_RBF_	RBF_NN	SVM (RBF_kernel)	Multi_ Logistic	Logistic	Logistic_Tree	C4.5	Naïve Bayes
Breast	80.0	83.06	80.0	80.0	72.0	72	72.0	52.0
Leukaemia	100.0	94.44	66.67	100.00	88.89	88.89	83.33	100.00
CNS	76.88	80.00	66.66	73.33	66.66	66.66	66.66	66.66

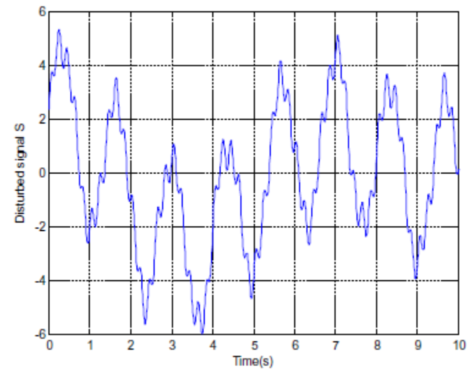
- Robotic surgery

Liu et al. [223]eliminated the effect of physiological hand tremor in robotic assisted microsurgery using multi-resolution analysis of 3D-fuzzy_wavelet_ transforms(Mexican Hat- and Morlet-), fuzzy_PSO and NNs. The simultaneous extraction of time-frequency characteristics of tremor signals, calculations with incomplete information of tremor signals are performed with 3D-Fuzzy_wavelet filter (Fig.5). Fuzzy PSO optimizes the adaptive parameters of the network (chart 16).The results are superior to RBF_NN. The very low error, better estimation accuracy and compensation performance for perturbations endorse this method for incorporation in commercial robotic surgical systems.

chart 16: Fuzzy_PSO

- Fuzzy set theory
    - Gaussian membership
        - Position and velocity
- No gradient information
- Uncertaintiestaken care
- Robust algorithm
- No premature convergence
- Exploration
    - Convergence
- Exploitation
    - Global search





[courtesy of Ref 223]

Desired operation signal r(t)	Signal corrupted with tremor of low frequency and small amplitude n(t)

$$r(t) = 3\sin(15\pi t) + 2\cos(3\pi t) + 5t^4 - 0.8t^3 - 2t$$
$$n(t) = 0.3\sin(0.04\pi t) + 0.1\sin(0.0312\pi t) + 0.6\sin(0.1\pi t)$$
$$s(t) = r(t) + n(t)$$

Tremor suppression with small amplitude and low frequency

Method	Mean_ RMSE
RBF	**0.305**
3DFWN	0.2692
3DFWN-FPSO	**0.1826**

Fig.5: Simulation and extraction of signals

3.5 Biometrics (Biology + om +etrics)

Discrimination of infested crops

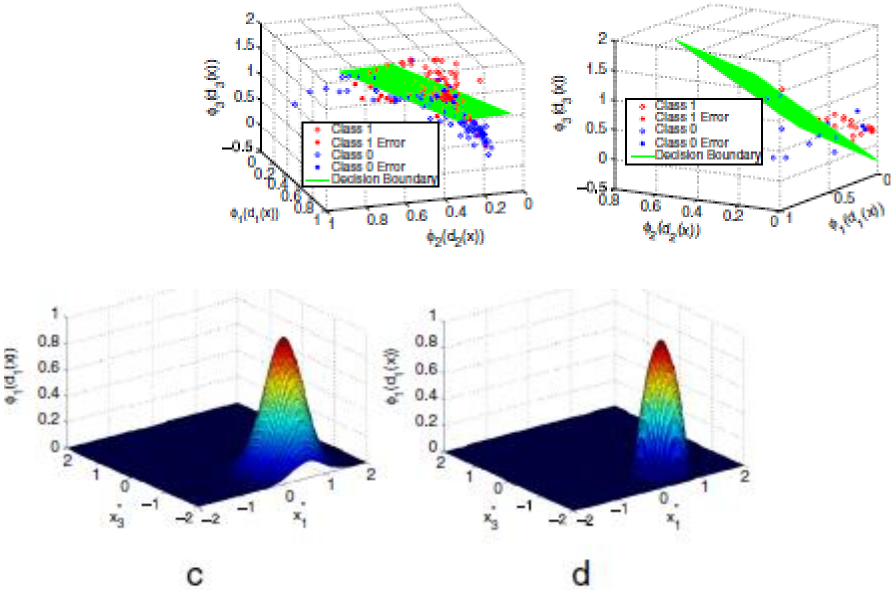
Fernandez-Navarro et al. [170] applied q-Gaussian NN for classification tasks from UCI repository and discrimination of R. segetum patches in the sunflower crops in two naturally infested fields. The study is in Andalusia, southern Spain, called Matabueyes and SantaCruz. The input (chart 17) is from remote-sensing aerial imagery of the areas in conventional-color and color infrared regions recorded during middle periods of May, June and July. The images were digitalized followed by re-sampling corresponding to 40cm x 40cm ground area. The outputs of the neural networks (table 8) are interpreted from the point of view of probability through the use of the softmax activation function.

Table 8a: Comparative study of q-RBF_NN with other RBF_NNs

	RBF				
Dataset	**OS_ELM**	**Evol_ELM**	**Classical**	**Cauchy**	**q-Gaussian**
Sonar	0.78	**0.88**	0.88	0.83	0.86
Hepatitis	0.80	0.81	**0.86**	0.81	0.82
Breast	0.68	0.69	0.67	0.66	**0.71**
Liver	**0.72**	0.69	0.70	0.65	0.71

UCI Datasets

- Labor
- Promoters
- Vote
- Card
- German
- Heart
- Breast-cancer
- Liver
- Hepatitis
- Sonar



q-Gaussian RBF_NN model for Liver data set φ1(d1(x)) using the variables x∗1 and x∗3 (c) q → 1 (Standard_RBF) (d) optimized_q (= 0.106); [courtesy of 170]

Table 8b: Comparative study of q-RBF_NN with other methods			
Dataset	**Best CG(%)**		
	Evol_PU_NN	**LR+EPU**	**q-Gaussian_ RBF-NN**
Mtb-May	70.6	70.6	**72.50**
Mtb-June	98.7	99.2	**99.38**
Mtb-july	79.8	79.0	**80.00**
Sc-May	78.4	77.5	**79.77**
Sc-June	98.4	**98.7**	98.42
Sc-july	83.1	84.3	**85.30**

ELM	:	Extreme Learning Machine (Extr. Lrn. Machin.)
OS	:	Online Sequential (Online Seq.)
Evol_PU	:	Product-Unit
LR	:	Logistic Regression (Logist.Reg.)

Chart 17: Aerial imagery data		
Input	:	digital values of image in all bands
conventional-color	:	blue (B), green (G) and red (R) broad bands
color infrared	:	green, red and near infra-red bands
Pixels		**Corresponded to**
800		R. segetum
400		bare soil class

$$softMax(x) = \frac{e^{f_{RBF\_NN}(x)}}{1+e^{f_{RBF\_NN}(x)}}$$ **Eqn. 4**

softMax(x)	:	probability that a pattern belongs to class A
1- softMax(x)	:	probability of a pattern x not belonging to class A

Error_function : negative log-likelihood

$$LogErrFn = \frac{1}{NP} * \sum_{i=1}^{i=NP} \left[-y_i * fn_{RBF\_NN}\ x_i + \log\exp\ fn_{RBF\_NN}\ x_i \right]$$

ObjFn: min (error_function) …..**Eqn.5**

If	pattern x_i belongs to class A
Then	y = 1, Else y = 0;

➪ Differential equations (DEs)

Reformulated RBF_NN of Ciocoiu[178]has more generalization accuracy compared to MLP-NN with BP. This NN efficiently solves differential equations.

Commerce

Shen et al. [113] reported forecast of Shanghai stock index with RBF_NN with more acceptable results compared to GA, PSO, ARIMA, MLP_BP and SVM. Here, k-means clustering method optimized by artificial fish swarm algorithm is used to train processes of RBF_NN.

3.6 Engineering

➪ Multi-sensor instrumental data

In data processing of a multi(m-) sensor instrument [137], oneRBF_NN is used for each sensor to detect influential features which are the input to fuzzy NN. This hybrid RBF-FIS is self-organizing and self-adjusting NN. It is applied for metal cutting process with increased accuracy.

➪ Fault diagnosis

Zhang [171] reported multiple feature selection algorithm in fault diagnosis of stator winding short-circuit failure of rolling element bearings in rotatory machines. These datasets contain 256 and 200 features respectively. The hybrid feature selection procedure consists of fitter models and wrapper models. The features are pre-ranked with eight filler models. RBF_NN is used to re-rank features based on weighted voting scheme. Now a binary search and SBS (sequential backward search) models are used to increase the accuracy by selecting relevant features. This multiphase hybrid model employing RBF_NN resulted in a reduction of number of features from 256 to 5 and 200 to 2. In both cases RBF classifier error was zero.

➪ Electrical power systems

Kavousifard [172]used Fuzzy logic to reduce uncertainty in assessment of electric power systems. RBF NN models the nonlinear behavior of training set process with fuzzy logic. The results endorse the model on comparison with a standard reliability test systems (RTS-96).

➪ Servo drive

El-Sousy [29] reported Rec_RBF_NN based on Self-evolving-fuzzy_NN to control torque for permanent-magnet synchronous motor (PMSM) servo drive. This hybrid system learns structure and parameters simultaneously. The approximation errors in neglecting higher order terms of Taylor's series are accounted. Lyapunov stability analysis guaranteed stability of solution. The results established precise and robust control irrespective of load disturbances and PMSM uncertainties.

➯ Communication systems

Chen [77] used symmetric RBF classifier in multiple-antenna-aided communication systems. Orthogonal forward selection procedure in applied to construct a sparse representation of symmetric RBF. The object function is based on Fisher ratio of class separability and leave-one-out misclassification rate. This NN gave a high SNR gain and is robust to the choice of the width of RBF. It is applied to four users with two receiving antennas or seven users with four antenna channels. When a priori knowledge is available, it is apt to incorporate it in the model which is a black box or grey.

➯ Laminar and turbulent flows

Mohammed et al. [25] applied 1D-integrated_RBF-network for dispersion of laminar and turbulent flows in an open channel with a smooth bottom. In general advection–diffusion equation (with higher-order derivatives) is used for the depth-averaged concentration of substances in channels. The results of 2D-equation with analytical derivatives and 1D-numerical approach are comparable showing negligible effect of longitudinal diffusion.

➯ PSO + adaptive mutation (Uniform/Gaussian) +RBF_NN

Yao et al [53]put forward an adaptive optimization strategy using mutation operator with uniform and Gaussian distribution for PSO. It is hybridized with RBF_NN and applied to a target localization task. A comparison with GA/ PSO based RBF_NNs (Alg.5) showed adaptive mutation based RBF_ PSO_NN is faster with good performance characteristics.

Alg. 5: PSO + adaptive mutation (Uniform/Gaussian) +RBF_NN

```
Initialization PSO
Cal fitness value of each particle

Repeat    until iter <iter_max
          global extreme point ← particle with min(FitFnvalue)

          DO for all particles
             individual extreme point ← current particle
          End DO

          Update (Chart 18)
          →  position of particle
          →  Velocity of particle
          →  Apply adaptive mutation operator
          →  update particle position

End repeatTrn
```

Chart 18: PSO + adaptive mutation (Uniform/Gaussian) +RBF_NN

	Eqn.
$V_i = \omega \cdot V_i + c_1 r_1 (P_i - X_i) + c_2 r_2 (P_g - X_i)$	6
$X_{i+1} = X_i + V_i$	7
$\begin{cases} x_{id} = x_{\min} + rand \times (x_{\max} - x_{\min}) & rand \le p_m \\ x_{id} = x_{id} \times (1 + 0.5 * Gaussian(\sigma)) & rand > p_m \end{cases}$	8

Initiation

GA

population size	20
inertia weights_max	0.9
inertia weights_min	0.2
Generations_max	100
prob_mutation_max	0.1
prob_mutation_max	0.01

$p_m = \begin{cases} p_{max} - \dfrac{(p_{max} - p_{min})(f_{max} - f)}{f_{max} - f_{avg}}, f \geq f_{avg} \\ p_{max}, f < f_{avg} \end{cases}$	9		**PSO_particles** initial velocity range [-1, +1] Coef_acceleration [2.0;2.0]
$\begin{cases} new_{TDOA} = old_{TDOA} \times 10^5 \\ new_{coordinate} = old_{coordinate} \times 10^{-5} \end{cases}$	10		

Evol.MARS + Honeybee alg.: Cheng et al. [27]employedhybrid paradigm with EMARS and honeybee algorithm to predict energy performance (Chart 19) of civil constructions for example buildings. A tenfold cross validation study revealed that the cooling and heating loads are reduced by 65% and 45% of other methods.

Chart 19: Energy performance	
Input : 6 variables	
Output : Heating load, cooling load	
Methods compared : SVM, CART,MLP_BP	
+ Autonomous functioning + No human ntervention + No domain knowledge	
MARS	Multivariate Adaptive Regression Splines
Evol. MARS	Evolutionary MARS

Robot

Le and Kang et al. [150] reported an adaptive tracking controller for parallel robotic manipulators with RBF_NN to account for modeling error and frictional/external disturbances. The simulated data for parallel manipulator with two degree of freedom (DOF) and Lyapunov stability ensures the applicability of the method.Zhao et al. [166] applied RBF_NN inconsensus control for multiple robotic manipulators. The observers are developed to calculate uncertainty of the system followed by arriving at control torque of the leader manipulators. An adaptive consensus control is developed to refine Ws of RBF_NN using Lyapunov stability theory. The result is RBF_NN enhances accuracy and the consensus error is stabilized.

3.7 Alpha numeric character and Text categorization

Analogue circuits for RBF_NN

Alg. 6: RBF_NN with resource allocating and EKF procedures

Trn_preliminary_over = No

Phase 1

Initializes preliminary structure (RB neurons, initial parameters, Ws) with RAN
- Reduces complexity of NN
- Preliminary optimization

Repeat until training_ preliminary_over

Improved k-means and average-linkage algorithm for initial centers and width cluster
Input training data is input to RAN_NN
Number of RB neurons adjusted with improved novelty criteria

If Input sample does not satisfy the novelty
Then Add this sample as a new neuron
Else RAN adjusts Ws by LMS
Endif

Carvajal, and Figueroa [142] derived analog models for the circuits for implementing Nearest Neighbor (NN) method and RBF_NN. The analog classifiers for standard face and hand-writing digits databases are emulated.

Text categorization

Song et al. [28] used RBF_NN with resource allocating network (RAN) learning algorithm and EKF in a phase wise manner (Alg.6, chart 20) for improved precision, accuracy of classification in text categorization.

> **If** no RB neuron added for three consecutive iterations
> **Then** Trn_preliminary_over = **yes**
> **End repeatTrn**
>
> Phase II
>
> - Refines parameters of NN by EKF
>   - Improves the classification accuracy
>   - Reduces complexity of NN and computations
>   - Higher level refinement of structure, Ws

Dataset. Reuter-21578:1500 documents containing 10 categories (coffee, crude, earn, grain, interest, money-fx, acq, ship, sugar,and trade) are selected. After removing stop words, weight of each feature is calculated. The results with RBF and MLP_NNs hybridized are given in table 9.

Dataset.20-newsgroup corpus set: 1200 items dealing with 10 topics (alt.atheism, comp.windows.x, sci.crypt, rec.motorcycles, rec.sporthockey, misc.forsale, talk.politics.guns, talk.politics.mideast, sci.space and sci.med) are chosen (table 9) .

Chart 20: shortcomings and remedial measures of algorithms

- Initialization -- noise data
  - Deteriorates novelty criteria
  - Increases network training time
  - Leads to reduction of employment effect
- RAN Uses LMS
  - Lower convergence rate
  - Remedy : Extended KF
    - Increases network complexity
    - Increase in CPU run time
    - Remedy : Cal initial centers of RBF clusters by mean clustering method
      - Overcomes local optimal of clustering algs.
    - Apply root mean square (RMS) sliding window
      - Reduce effect of noise
      - Improves novelty criteria of RAN

Table 9: Comparision of RBF and MLP hybrid methods for text classification

Hybrid alogrithm		DataSet 1		DataSet 2	
Alg.1	**Alg.2**	**MAE**	**Dim**	**MAE**	**Dim**
Clustering RBF	VSM	9.3e-05	1000	4.3e-04	1000
	SFS	42.e-05	300	9.3e-04	200
MLP_BP	SFS	7.8e-05	300	7.2e-04	200
Improved RAN	SFS	4.0e-05	300	6.5e-04	200
		Features total	7856	13,642	
		Features selected	1,000	1,200	
	SFS:	Semantic feature selection		VSM:	Vector space model

3.8 Mathematical/statistical tasks

Classification

Classification can also be conceived as a decision-making task [132]. Here, RBF transfers continuous input space R^{dc} into finite set of classes. During the training phase, weight vectors of network are determined for a finite set of training data set of feature vector (x) labeled with class membership. Testing phase consists of presentation of unlabeled $x_{un} \in \mathrm{R}^{dc}$ to the trained network and the output is the estimate of the membership of $y_{un} \in y^{d}$. Instead of global scaling parameter $\sigma \in R$ for all basis functions, each basis function can have its own width parameter $\sigma_j \in R$. Euclidean, Mahalanobis or city block distances are used. The output of ideal transfer function for a classification task should be a constant within the class interval and should rapidly decay at the interclass boundaries. Thus RBF employing Kernel function with adoptive R (nearer to zero) and σ (a small value, 0.5) is superior to simple Gaussian functions.

Hard classification boundaries

RBF_NN is used with success to classify many bench mark (Wisconsin, statlog) datasets viz. breast cancer, heart, iris, sun spot, Mackey-Glass, DNA, character recognition, and image analysis. Teixeira [175] used RBF kernel in classification models. The classification of datasets with 2 to 60 features and data points of 140 to 1010 are tested with the kernel subspace model.
q-Gaussian distribution_NN was used as wave function to probe into properties of high density Bose–Einstein condensates and DNA molecules.

Spiral problem:In nineteen sixties, Minsky and Peppert proposed spiral task which is connected to Ulman's inside and outside problem. The types of spirals and their solution are wide spread in literature.

Function approximation

Unknown non-linear function: Zhu et al.[186] proposed robust adaptive neural tracking control for switched affine non-linear system using RBF_NN, which approximates unknown non-linear function.

Hybrid RBF

Pedro and Takahashi [157] used RBF network to develop a function representing decision-maker's utility steps with ordinal information generated from queries. Interactive non-dominated sorting algorithm (NSGA-II) with a preference model finds sampling regions. A dynamic crowding distance (DCD) density control method is employed.A detailed sampling of the DM's preferred regions are used for non-uniform sampling of the Pareto-optimal front while non-preferred regionsfor a coarse sampling. This ensures fine-tuning control of Pareto optimal front density of samples.

3.9 Standard data sets

Modeling datasets with missing values

Ravi and Krishna [149] tested mean imputation methods for missing values in classification, regression, bankruptcy and credit score datasets. The hybrid model consists of auto associative (Auto.assoc.,AA) neural network and RBF/PSO/GR for three types modes of training viz. online, off line and semi-online respectively. The results of Wilcoxon signed rank test show Generalized Regression_AA_NN performs better than PSO_AA_NN and RBF_AA_NN.

Parameter calculation

Gan [26] used RBF_NN to calculate autoregressive coefficients in a quasi-linear_AR model. The hybrid GA_ steepest descent algorithm selects the influential input variables and simultaneous optimization of model parameters. The results for monthly time series data are of high quality.

Dataset.[**SVR + Annealing robust learning algorithm+ RBF_NN**] **[Alg.7]:** A simulated data setis generated with function (Fn.1). Three artificial outliers are merged with data and training of RBF_NN is performed (Fig.6). From the figure, it is eye-catching that outliers remain as they are without influencing the model outcome.It is applied to soft sensors batch polymerization reactor and statistical process monitoring.

Alg. 7:SVR + Annealing robust learning algorithm+ RBF_NN]

Phase 1: RBF_NN is used to extract latent structures through linear algebra methods

- No need of nonlinear optimization

Phase 2: The predictor and response variables in original (measured) frame are reconstructed with two FF_NNs

- Multiple latent structures can be extracted in sequential or parallel mode
- Nonlinear relationship between the latent structures and the original predictor variables is the output
- Fast convergence speed

Experimental design-robust parameters

Hitherto full quadratic models are popular over decadesfor experimental design of influencing factors inChemometrics and engineering. Robust parameter design optimization deserves attention with latest non-parametricmethods. Elsayed and Lacor [23] found from a rigorous study That Kriging, RBF and RBFNN (with MSE) are accurate in reproducing non-linear input-out data without a priori known implicit functional relationship. Further, Pareto front (Fig. 7)will open options for alternative solutions ready for an in-depth inquiry.

Fn.1 : Simulated function with outliers

$$u_1(\mathrm{k}) = \sin\left(\frac{2 * pi * k}{25}\right) \quad \text{….. Eqn. 11}$$

$$u_2(\mathrm{k}) = \cos\left(\frac{2 * pi * k}{25}\right) \quad \text{….. Eqn. 12}$$

$$\begin{bmatrix} y_1(k+1) \\ y_2(k+1) \end{bmatrix} = \begin{bmatrix} \dfrac{y_1(k)}{1+y_2^2(k)} \\ \dfrac{y_1(k)y_2(k)}{1+y_2^2(k)} \end{bmatrix} + \begin{bmatrix} u_1(k) \\ u_2(k) \end{bmatrix}.$$

….. Eqn. 13

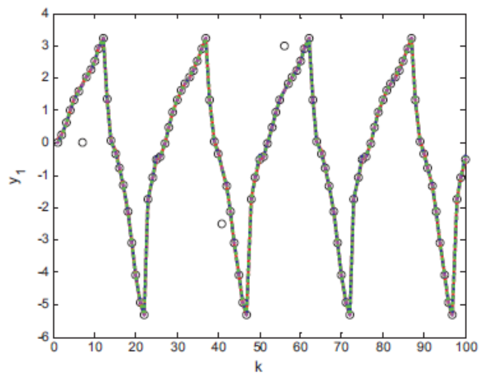


Fig.6:Profile of y1 contaminated with outliers; Epochs: 500

Rules_extract ion-classification model

Pereira [159] introduced genetic program to extract rules from classification. The input datasets of different categories viz. numerical, logical, textual, geographical [points, lines, polygons] are analysed. The bi-objective functions comprise of maximization of accuracy and minimization of complexity of classifier. The results of wine, hepatitis, power transformers incipient faults and level of development of cities datasets are compared with C4.5, RBF and SVM. This multi-objective-GP is suitable for transparent classification, although the CPU time is high.Wang et al. [88] proposed RBF_NN foracquisition of knowledge, storing, retrieval and its reuse in an uncertain non-linear-SISO system (chart 21, Fig. 8). It also considers dynamics of the unknown system and predefined behavior of tracking error.

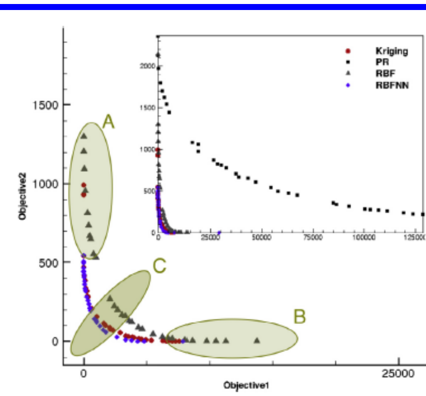


Fig. 7: Pareto front for multi-objective optimization

Obj1 : $\mu - T^2$ **; obj2 :** σ^2 **;**

Methods : Kriging, PR, RBF_NN (Courtesy of Ref: 23)

Chart 21: Components of Hybrid RBF-NN

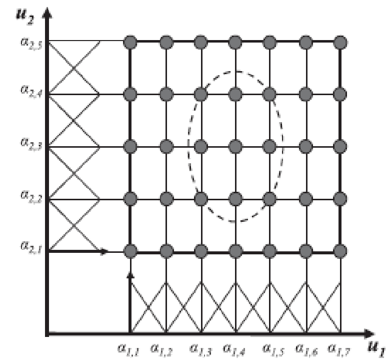
k -means

- A priori knowledge of number of clusters
  - Remedy: fuzzy C- means
- **Non-symmetric (triangular) fuzzy c- means**
  - Faster than k-means
  - Automatic number of clusters
  - Increased number of parameters
    Remedy:**Evolutionary SA (ESA)**

Simulated annealing (Sim.Anneal, SA)

- Single solution
- Harder to escape from local minima
  During the last stages of the algorithm
  - Remedy:**Evolutionary SA (Evol.SA)**

Fig. 8: 2D- Elliptical mf; Non-symmetric



[courtesy Ref 88]

If NP is small

Then exhaustive search

Else Evolutionary SA (Evol.SA)

4. Architecture Evolution in RBF_NN

If NN is a goal, sub goals include choice of architecture, parameters of TF, training methods etc. But, NN itself is a sub goal in control of F16 safe landing, weather forecasting, diagnosis/ treatment of a disease, chemical plant operation or pollution abatement and sustained environmental quality. The limitations of first generation RBF_NNs are summed up in chart 22. Keeping the targets in memory, the significant advances in RBF_NNs during last two decades include the application of second-generation AI tools (Genetic/ evolutionary algorithms, fuzzy logic, particle swarm/ ant colony optimization and so on). The knowledge transfer between the two disciplines emerged in new tools with robust and predictive capabilities.The discussion follows with modification or improvement in architecture, kernel function, calculating optimum number of centers /their co-ordinates, initialization of weights, training algorithms for estimation of W, cell structure, automation of parameters of TF/W, inversion of trained net, rule generation/knowledge extraction and appropriate use of mathematical/statistical algorithms for sub goals. It is all to equip RBF_NNs with robust technical features for takeoff into a new computational world to solve real time mega tasks unlike yesteryear's toy problems.

Chart 22: Limitations and remedial measures of RBF_NN

- Basis functions are fixed
  - Remedy: Flexible or adoptive Basis Functions like those in WAVELETS
- Complexity of RBF_NN increases with the shape and discrete patterns with oblique boundaries (clusters)
  - Remedy: centroid MLP
- RBF does not give good prediction outside the range of training data set because of the local nature of TF
- Optimization of adjustable/free/unbounded parameters is mixed integer hard problem
  - Number of RL neurons
  - centers, widths, type of distance
  - WRO
  - Remedy: Natural computational procedures
- For function approximation of simple linear or quadratic trends in one or two dimensions, classical NNs in general areinferior tools

Newer noteworthy architectures proposed are with two hidden layers, recurrent inter-layer-connections, growing and shrinking number of radial neurons. A few typical publications deal with minimum resource allocating NN[84], dynamic decay adjustment [201], temporary dynamic decay adjustment[201], growing RBF [133] /cell growing structure (RBF-CGS) [117], extended GRBF[134]with reverse connections from output to RB layer, recurrent RBF[202], SOM-GRBF[134], finite recurrent discrete time RBF, and complex weight RBF_NN[81].

KB. 2: Number of neurons of RBFs

Strict interpolated RBF

If #neurons_RBLayer = NP
Then Strict interpolation
+ good curve fitting procedure
- poor generalizability

+ It is a statistically sound procedure

If noisy data sets
Then
- large size of network
- over fitting
- numerical problems

If number of radial units small
Then poor representation of the surface

If System is non-linear & dynamic
Then Simple RBF is not adequate

4.1 Number of Radial basis layer neurons

The minimum RBF_NN architecture is 1-1-1 for SISO system. But in practice,a few neurons in the radial layer are sufficient for classification and function approximationtasks. But, in presence of high noise and outliers and overlapping patches the optimum number of neurons in an adequate NN-model are high. Sometimes they exceeds even number of patterns in the training set [201]. In strict interpolation, the number of radial neurons is equal to the number of patterns (KB. 2)in the training dataset. Regularized strict interpolation RBF_NN [13] is an improvement over strict interpolated RBF_NN and outperformedmany earlier statistical methods. Further, the results are superior to standard RBF in classification, non-linear time series, speech recognition and solutionof stochastic differential equations. The sole concept of wisely choosing numbers of RBFs is for higher generalizability, prediction of unseen data, novelty detection, robust to outliers/ noise, enough plasticity to learn new patterns_unseen in training set, high stability not to create (inflate) unnecessary RBFs for slightly different trends. At the same time, the architecture should be as simple as possible with knowledge extractability in the form of If-then-else rules.Recent reports focused attention on these aspects and there is noteworthy progress in incorporation some new features and surmounting hurdles compared to the standard RBF_NN [201].

Walczak[40] reportedorthogonal LS and kernel function to select optimum number of neurons in radial basis layer. Tetteh et al.[41] used biharmonic spline interpolation for optimization of neurons. Billings[125] selected minimum number of neurons by GA using single/multi-object functions. Lacerda [93] also used GAs to optimize the size of RBF_NN, which is yet a parsimonious model with good generalizability. But, this NN takes longer training time. Andrieu [120]employed global optimization techniques like reversible jump Markov-chain-Monte-Carlo (MCMC) and simulated annealing algorithm (SAA) to arrive at the number of RBL neurons, regularization/noise parameters and spread of RBF in the joint parameter space. Here, a hierarchical full Bayesian model is used.

4.2 Direction of Connections

❄ Recurrent RBF_NN

The neurons are connected into an oriented graph with each edge (i,j) starting from ith to jth processing element (PE) or neuron with a real parameter c(j,i)∈ R. A deterministic finite automaton is achieved with a maximum norm[202]. It is at least as powerful as finite automata. Second order neural networks are related to Recurrent-RBF_NNs.

❄ Self-organizing GRBF (SO-GRBF)

It is RBF_NN with reverse connections from output to radial layer (RL) and functions like Bayesian classifier with an additional feature of novelty detection. The hierarchy of deterministic annealing method which implements a multi-scale approach surmounts the convergence problem of conventional expectation maximization (EM) algorithm. The threshold values for misclassification rate, bias in output neurons and spread constant are the critical factors to detect novelty and invoking a new class [207]. SO-GRBF is applied to speech recognition task.

4.3 Number of hidden layers

The RBF_NN with two hidden layers include classification RBF wherein competition layer[138]performs classification task more efficiently (fig. 9a). That way Generalized regression_RBF_NN also has summation layer in addition to radial layer. RBF_NN with one radial and one sigmoid layer model more complicated profiles than those tackled by either of these transfer functions alone.

⇨ Classification RBF_NN (Class-RBF_NN)

It is a four layer RBF_NN consisting of competitive layer in addition to radial, input and output layers. The m-D pattern space is divided into a set of maximum size hyper ellipsoid subspaces based on the statistical distribution of training samples. Zhu[138]used global learning of parameters employing maximum likelihood classification. It results in a trained final architecture. An evolutionary learning algorithm is used with global distribution of training samples. There is no need of a priori setting of parameters and a unique solution of Class-RBF_NN is obtained for a given set of data.

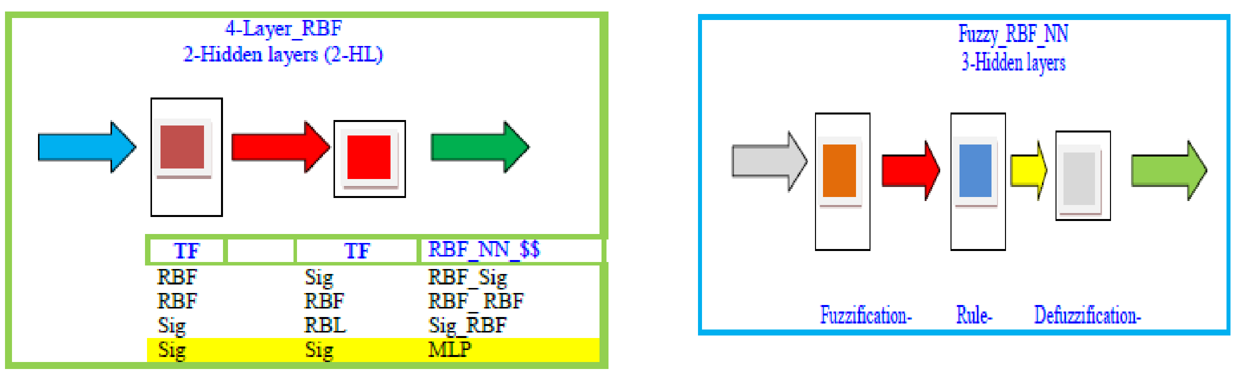


Fig 9(a) RBFs with multiple hidden layers

4.4 Growing RBF_NN

A recent approach of growing nets is twofold. The first is retaining the layer structure and the other random growth with no unique topology. Resource allocating network (RAN), Min_RAN, Dyn.decay.adjust. (DAA), Temp-DAA etc. implement different strategies to land on the targeted goal.

Alg. 8: Addition of neuron to RBF_NN[84]	
Algorithm	**Equation**
A neuron is added if ➢ input is far away from all centers& difference between yi and ynni is significant	If $\|X_n - c_{n,r}\| > Threshold_n$ & $Threshold_n = \|y_n - f(X_n)\| > Threshold_{min}$ Then new hidden node

Growing RBF_NN [133]

Dynamic Decay Adjustment (DDA) RBF (Fig. 9)

Chart 23: Dynamic Decay Adjustment RBF_NN

- + Surmounts plasticity-stability dilemma
- − Neuron explosion results training difficulties

Bertheold[210]proposed the growing radial neurons to explain new patterns without increasing the instability. The prime objective is to surmount the notorious stability-plasticity dilemma[133]. The remedy for stability/plasticity issue is similarity based vigilance criteria. Grossberg introduced a new node when there is decrease in local error for a specific range of input (Alg.8). The number of neurons of radial layer is grown to adapt to the changing patterns. In lifelong learning, the learning rate is altered dynamically at each neuron. In RBFNN-DDA, a neuron is inserted for each outlier (Chart 23). Similarly new neurons are introduced for overlapping and noisy regions. In the modified algorithm deleting the neurons is also implemented as an online process.

Temporary dynamic decay adjustment-RBF_NN [RBF_NN-DDA (T)]

The neuron labeled as temporary is stored in the permanent list of architecture when it covers sufficient number of data[201]. Other superfluous neurons are pruned after an epoch. The centers of these RBFNNs are not used any more as training data in subsequent learning epochs. This approach of deleting the temporary ones reduced the range from 33 to 80 % of neurons required for DDA-RBF_NN in the case for standard data sets (Table 10, Fig. 9b) viz. IRIS, cancer, heart etc. However, there is no significant change in correct classification rate, althoughnumber of neurons is reduced. Thus, it is a milestone in NN training procedures.

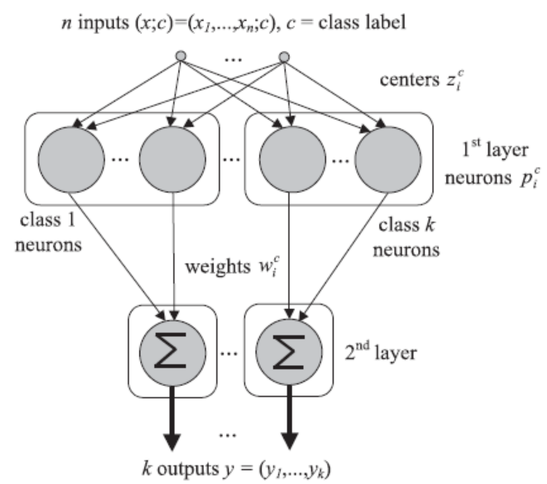


Fig. 9b: topology of DDA-RBF_NN with separate set of neurons for each class (courtesy of Ref 201)

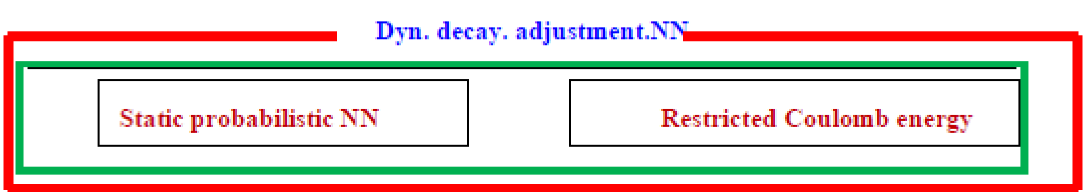


Fig 9c: Component algorithms of Dyn.Decay.Adjustment-NN

Table 10 : Reduction in number of RL neurons with DDA(T) algorithm

	IRIS	Diabetes	Horse	Cancer	Heart	Australian	Vehicle	Segment	Stat Image
DDA	15	288	163	71	84	179	321	256	1533
DDA(T)	14	66	19	38	33	84.3	86	171	510
	Shuttle	DNA	Letter						
DDA	291	1453	4171						
DDA(T)	260	89	2361						

Restricted coulomb energy + Probabilistic NN

Paetz [201] combined the philosophies of restricted coulombenergy and probabilisticNN principles. Here, the network growsdynamically using an online learning technique ofPaetz. Thus, it has lowernumber (<NP) of neurons in the hidden layer unlike Probabilistic-NN, but leads to reliable classification(Chart 24). In this type of NN, neurons are inserted during training[34]. Hitherto the strength of connection of a neuron to the other is considered.It is the first of its kind where the basic concept shrinking neuron radius in the radius of neuron is envisaged. The pruning of neurons is affected (Alg. 9) after each epoch rather than at every training cycle.

Resource allocating RBF

Resource allocating RBF consists of input, radial basis layer and output layer with sigmoid neurons [127]. It was proposed [34]for function interpolation and the sequential learning methodhas the advantage of its suitability for on-line modeling of non-stationary processes. Wallace employed gradient descent method for training using min(ESS) as convergence criteria.

Chart 24: Shortcomings and remedial measures of Restricted coulomb energy + Probabilistic NN

+ High classification performance and fast training

- More number of neurons

Remedy: Pruning process adopted in Neuro-fuzzy NN

- Involves a new complete training procedure

Remedy: Paetz [201] adopted pruning of neurons after each epoch rather than the training cycle.

+ Implemented in hardware

- Greedy neuron insertion algorithm

Remedy: On-line pruning strategy

Functioning:Initially, it does not contain any hidden neuron. The parameters of the network are refined using gradient descent method employing ESS[127]. Ifboth (prediction error and novelty detection) criteria are satisfied, a new hidden node is added to the Resource_Allocat_RBF_NN.

The advantage is unsupervised clustering offers a procedure to initialize RAN architecture, which contain radial basis neurons. The clusters can be represented through their mean vector. However, the limitation is outliers create unnecessary nodes, when NN is started with no hidden neurons. It leads to increasing learning effort and convergence time. Further it deteriorates generalization performance.

4.5 Shrinking RBF_NN

The pruning strategy for the neurons involves comparison of normalized output for all neurons with a threshold (Alg.9). This learning approach is tested for classification of 4- and 8-clusters, function approximation of a sine function and time series prediction of sunspot data [135].

Alg.9: Pruning of neurons

A neuron is pruned if its'

- Contribution is a little (< threshold) to the overall output for the whole data set
- Activation region is small
  - Eg. Neurons over trained i.e. A neuron j has low values in different dimensions of input space
- Transfer function is similar to other neurons

4.6 Growing & shrinking (resizing) RBF_NN

In pseudo-Gaussian-RBF, a provision is available for addition of a new neuron, detection and removal of less active neuron[90].The architecture of RBF in the framework of network/graph theory is proposed[194].

Minimum resource allocation learning algorithm

It is a dynamic RBF incorporating growing and pruning of RL neurons. In 1998, Yingwei[84] proposed this algorithm (Alg. 10) consisting of two phases to arrive at minimal topology-RBF. The network growth is performed with resource allocating strategy. In the second phase, radial layer neurons are deleted if contribution is relatively small to the overall network output. This strategy and extended Kalman filter (EKF) are combined in RAN resulting in minimal RAN. MRA-RBF_NN is used for neuro controller implemented using F8 fighter aircraft longitudinal model.It is better than MLP trained with R-PROP and Moody's dependence identification algorithm for classification and function approximation. It is robust and tolerated more serious false conditions compared to FF-NNs [119].

Alg. 10: Minimum Resource Allocation network [119]
Online Algorithm

Step 1 **For** every new input calculate
- ⇨ residual
- ⇨ distance

end

Step 2 **% Criteria to create a new neuron**

If error exceeds a minimum threshold &
mean square error of NN for the series of past data is greater than threshold &
new center is sufficiently far away from existing center
Then create a new neuron

Step 3 **% Criteria to prune current neuron**

For all hidden units
- ⇨ cal. hidden unit output
- ⇨ find largest absolute hidden unit output
- ⇨ normalize output

If normalized output of kth unit < threshold for
M consecutive observations
Then remove kth unit

Step 4 If there is no need for a new neuron
Then new_neuron = .no.

If new_neuron = .no.
Then refine parameters [W, C, width] with Kalman filter

Step 5 If normalized contribution to output of a center is below
threshold for consecutive inputs
Then neuron is pruned

Step 6 If two centers are close to each other
Then combine two neurons to one

5. Learning Evolution in RBF_NN

5.1 Learning in Biological species

The biological inspiration of numerical learning algorithms in vogue is from a deep level understanding of how a child learns need based skills/tactics or undergoes training from mother,on a prenatal chassis (skeleton of brain, CNS- and endocrine- system developed on genetic code during pre- (or anti-)/ peri-/ post-/neo-natal periods) and continued later through surroundings, teachersandmedia. Based on this analogy, the term learning was used in the historical NN terminology to adapt weights of connections between neurons of NNs with changing input patterns. The human brain understands and distinguishes the numbers, letters, pictures, images and scenes even when corrupted with noise in 2- or 3-dimensional space or in different orientations/sizes/colors/intensities. Each of these categories can be considered as a matrix of binary signals indicating the presence or absence of a dot and each row is a pattern. Since, this activity is perception with eyes, the artificial neural network with a hidden layer is known as a perceptron model. The basic components (neurons) are named perceptrons. The knowledge and skills are imparted through training in daily life inspired neural network scientists to describe the procedure as training. Just like assimilation and integration of concepts over time, the training is repeated again and again several times. It later included lifelong learning, forgetting schedules, reward-punishment impacts and grand-ma effect. Typical object functions in SVM and wavelets and NN are described in Chart 25.

5.2 Types of learning

Many modes of learning viz. incremental/adaptive/life-long, on-line/offline and task dependent/independent learning have been tried with success.In competitive learning mode, procedures like rival penalized[135], Gaussian masking, maximum likelihood classification [138], SVM, minimum resource allocating NN[84]and reinforced FS are introduced. Relearning, BYY/ Bayesian learning [12], forgetting schedules, unlearning are in the fore front of learning methodologies including novelty detection.

Chart 25: Typical object functions

ObjFn_NNs	Min(error on the training data) Usual Empirical Risk Minimization principle
ObjFn_ SVM	Minimizes an upper bound on the anticipated risk through Structural Risk Minimization principle + Statistical learning increases generalization
Wavelets	Discrete Continuous
	- Generates large data - More computational time resources **Remedy :** high speed computers + Efficiency Mother wavelet db2 < db4 < db8

- **Online-**

Online learning is adaptive [136]as and when new data is acquired through automated instrumentation or even by manual procedures. A master equation, which describes the dynamics of weight space probability density, is used to analyze online learning in RBF_NN. If the number of input nodes (I#) is less than the number of radial layer neurons (RL#), the dynamics is not affected by the problem of symmetry and breaking of symmetry. This is a continuation of earlier reports for I# > RL#.

- **Batch (Offline)-**

Offline learning corresponds to modeling after complete acquisition of data and batch mode is choice of the learning. The weights are accumulated and used to upgrade only after the presentation of all the pairs of patterns.

- slow learning process

- Lifelong-

On-line learning is limited to processes operated over a definite time period. And, lifelong learning affects need based modification of the model for a system in operation under a dynamic environment. It learns new patterns without forgetting earlier preservable/critical/core ones. It preserves the previous knowledge, which is not contradictory to the information generated with the current data. Ideally, there should be separate cells to retain the contradictory information with illustrations. The patterns endorsing or refuting the knowledge will lead to relative weighing of the contradiction. Further, reinforced learning and search for similar patterns operate in science and human brain. But, such provisions are not yet envisaged in NN paradigm. In catastrophic inference, sudden and complete erasing of all earlier learned patterns occurs. Catastrophic interference occurs in a distributed representation with insufficient number of neurons. Although lifelong learning is a solace, what to learn, what to forget and when to resort to newer paradigm and stop pruning worn out concepts is yet a stigma.

- Adaptive learning [133]

The minimization of bias even in on line learning is desirable. Robbins and Monro[20] proposed a stochastic gradient descent method with decreasing learning rate to zero is reported. In dynamic systems, it does not allow a fast change of weights resulting in a conflict. Amari [85] employed an adoptive learning factor of learning

$$\Delta w = \eta(t) * f(w,x)$$

Thus, Δw depends upon the weight in the previous iteration and the value of pattern in the current epoch. Recently, a global scale factor for adjustment of all weights is proposed. If the error is large, a higher value of the learning rate is adopted. When the error decreases to zero, the learning rate also decreases to zero.

- Forgetting schedules

The decay parameter with time in a learning schedule is a simple way of inducing forgetting process. Some of the typical procedures used are selective, catastrophic forgetting in NNs.

5.3 Learning algorithms

Some of the typical popular methods employed in learning of Ws of RBF_NN (KB. 3) include wake-sleep, robust full Bayesian learning [120], Bayesian Kullback Ying-Yang dependence reduction theory [206], abstraction Bayesian ensemble model [165], dynamic [88], fast, fuzzy_NN [118], generalized [188], generalized multiscale [124], genetic_evol orthogonal niches [70], global, improving generalization [135], lifelong [133], minimal _NN [84], multi-phase [132] and orthogonal least squares [67,73]

- ❊ Bayesian Ying-Yang (BYY) learning

It is an adaptive learning algorithm introduced by Xu [89]to decide the number of RL neurons, experts or basis functions. It corresponds to extended k-means algorithm for clustering using Mahalanobis distance. A subset of it is rival penalized learning. Harmony learning results in expectation maximization (EM) method for maximum likelihood learning.

- ❊ Rival Penalized competitive learning (Riv.penal.compet.learn)

Rival Penalized competitive learning [135]selects number of centers and adjusts their co-ordinates. Regularized LS (RegLS) estimates W and constructs parsimonious NNs. RBF with Riv.penal.compet.learn and RegLS performs better even in presence of severe noise and exhibits good generalizability. It avoids ill conditioning of solution of learning problem, as it does not involve matrix computations. The sun spot data, simulated trigonometric functions, and (4- and 8-) clusters are tested.

6. Training algorithmsEvolution in RBF_NNs

In NN models, weight values are learnt or the dataset is trained for a fixed network configuration. In statistical and mathematical literature, it is referred iterative refinement of model parameters. Maximumlikelihood classification, a global learningtool [138] and expectation maximization[131] are also used in training of RBF_NNs. Gaussian masking algorithm, a robust and computationally efficient LP model[144] is employed to train data for classification by RBF_NN. A classification function modifying Euclidean distance [139] accounts for the correlated input in time series data. A variational RBF approximation extending to a recent Bayesian algorithmconverges to original variational algorithm. It is applied to univariate stochastic double well multi-variate, Lorenz-3D and Lorenz-40D systems. It recovers the system and noise estimates. Training of WRO (weights of RBL to output layer) with EKF, SAA, wavelet, GA and regularized orthogonal LS have several advantages over simple pseudo-inverse, BP and quasi Newton algorithms. These adaptations not only overcome the dilemma of plasticity vs stability, short comings of overtraining, but also results in parsimonious models. Sequential learning is used to adapt to the structure of new architecture in pseudo Gaussian RBF [90]. Regularization theory comes to rescue where J(acobian matrix) is ill posed (ill conditioned) or inverse does not exist and the RBF_NN reported is named as Hyper RBF.In RBF_NNs plateau phenomenon occurs when two or more component basis functions are identical or the magnitude of one component becomes null. It results in singularities in parameter space. The stability of the Hessian and the existence of plateaus in batch and on-line learning were confirmed. Whenever new data becomes available, the training is repeated retaining the old prototype patters. Generally, the magnitudes of parameters (W) change as in the case of segmented algorithm for mean. The initial architecture is a two layer (I/O) NN. During, training the hidden layers and neuron are added on need basis. The repetitive process is stopped at an optimum value of ESS and generalizability. The performance of the algorithm is low. The limitation is that old patterns are not retained in the memory.Genetic orthogonal least squares [189], expectation maximization [131] and selective back-propagation [198] were also used in training Ws of RBF_NNs.

KB.3: Learning weights[133]

IF	learning rate is high
THEN	already learnt W is corrupted
IF	learning rate is low
THEN	change in W with environmental factors is not modeled
IF	learning rate is constant
THEN	never converges to a global optimum

7. Activation (Trasfer) functionsEvolution in RBF_NN

During these two decades, inRBF_NNs raised cosine[79], radial wavelet [143], mixed functions, cross product terms [81], pseudo Gaussian[90, 139] and generalized binary equations are used as transfer function in radial layer(Appendix-A1). Shape adaptive RBF, orthogonalization of kernel function and complex weight RBF have notable characteristics. The mixed RBFs was used as a TF and trained with GA with multiple-object functions including Akaike information criteria (AIC). They allowed points to be centers other than training vectors. Dual orthogonal RBF_NN replaced Euclidean distance and thus, overcomes ill effects of correlation of input vectors in models. The cross product term in the transfer function results for a better performance in classification task. . Clifford algebra as well as SVM is used to develop RBF_NN. WebbandShannon[83]reported an optimal choice procedure for TFs in RBF_NN. Hitherto, no choice is available in packages to arrive at an optimum TF, although many can be tested by user's choice or by heuristics in intelligent/expert mode.

♈ Normalized RBF

The output of neurons in the RBL layer is divided by the total input activity of radial layer neurons. Now, it is radially symmetric although the data may be of any distribution. In Generalized RBF_NN, normalization is done after weighting scheme [90].

.

Normalized RBF

- + No curse of dimensionality
- + Better performance compared to non-normalized RBF

♈ Complex- RBF_NN (Cmpl-RBF_NN)

Complex-RBF_NNs are those wherein any or all variables viz. output/input of RL-neuronsand/or parameters (center of RBF and Ws) haveimaginary component. In many signal processing applications, one comes across complex valued signals and data. But, the variance is always a real valued scalar. Igelnik [81] proved that complex-weight (CW)-RBF_NN(where only Ws are complex) approximates any continuous function in the standard unit hyper cube. The superiority increases with increase in complicated nature of task. Nitta[222] reported complex valued RBF with application for signal detection, communication channel equalization and multiple antennas aided wireless systems. But, for toy problems (XOR), simple RBF is more accurate and efficient [81].

♈ Radial Wavelet NN (Rad. wavelet.NN)

Holmes[143]testedthe performance of Rad.wavelet.NN with radial, harmonic, addition, and complicated and simple functions. Continuous wavelet NNs is viable alternatives to non-parametric regression. An a priori belief of plausibility of set of models is incorporated in the computations within the frame wok of Bayesian approach. Automatic Occam factor accepts probability of Markov chain Monte Carlo model.

8. Simultaneous refinement of architecture, centers, widths and Ws

In yesteryears, the variation of one-variable-at-a-time was in practice due to computational constraints associated with norms of mathematical/statistical methods regarding signal and noise structure in a system. Now, with advances in multi-objective-multivariable-multi-response soft/distribution free global and Pareto-front approaches, there is a noteworthy progress in simultaneous refinement of architecture and connection weights of RBF_NNs.

RBF parameters
⟲ Center
⟲ Width
⟲ Power
⟲ Distance: [Eucl; Manha; Mahala;]   ⟲ Function : [exp, log, poly ..]
Ws for refinement
⟲ # neurons
⟲ Layers, TFs, DataType
⟲ WROs

8.1 Recent advances in co-ordinates of centers of RBF:Broomhead and Lowe[189] suggested that the coordinates of the subset of points chosen at random can be taken as centers. This remained as one of the popular startup RBF_NN modeling activity. However, this choice is sensitive to how well the selected points represent the overall population. An information-based approach is to cluster the input vectors and allocating the cluster centers to the centers of Gaussian kernels. In recent times, the centers of clusters in RBF are determined by learning vector quantization (LVQ), classification trees [132], probability density function (pdf)/class membership[106], stochastic method[92]and rival penalized competitive learning [135]. If OLS is used for locating the centers, the responses of RL neurons are decorrelated. But, the method leads to poor generalization with inferior global optimization properties. Bishop [123] employed expectation maximization (EM) algorithm in optimizing the centers of the clusters. This is a sub-optimal approximation[68]. But, it performs a maximum likelihood learning, large class separation and selects parsimoniousnetwork architecture. The performance of RBF with the components of SVM as centersexcelsthat of simple RBF.

$$\exp\ -x^2 \qquad \text{Eqn. 14}$$

$$\exp\ -x'*x \qquad \text{Eqn. 14b}$$

Shi et al.[128]reported critical vectors for the selection of centers. The sensitivity is the expectation of the square of output deviation caused by perturbation RBF centers. The sensitivity analysis results in critical vectors used to obtain the centers of the RBFs. This approach exhibits better performance for classification tasks (Iris, glass data, Sat-image or letter) compared to conventional RBF and C4.5 algorithm. GA is extensively used in configuration of architecture of RBF_NN [125].The hybridization of RBF_NN with SVR uses the ε-insensitive loss function to determine the number of hidden nodes, the initial parameters of the kernel, and the initial weights of the network with fast convergence speed than BP or simple RBFNN.A memory mechanism is used into the adaptive growing learning algorithm to determine the number of hidden nodes.Recently, subtractive clustering [203], recursive orthogonal least squares [74], regularization [122] and Fisher ratio class severability measure [122] have been used in determination of centers of RBF neurons

8.2 Recent advantages in deviation in RBF_NN

In the statistical literature, mean and standard deviation are the free parameters calculable from data adhering to normal distribution. For a standard normal distribution, mean is taken as zero and variance as unity and the equation looks simpler. In quantum chemistry and physical chemistry, the equation is referred as Gaussian function. Thus, popular terms Gaussian type of orbitals (GTOs), distribution of molecular velocities are in vogue. Further, chromatographic/spectrophotometric profiles are sums of Gaussians. The word Kernel is a familiar in SVM meaning core. However, the kernel in operating system is not a mathematical function, but the core of software suit. Finally Radial basis function is sought after in feed forward neural networks where it is used as an activation/transfer function. Whatever is the name, it has a pivotal role in modeling and number of papers in theory and applications occupies major area.

KB 4a: Spread parameter in RBF	
If	Many data vectors are approximately significant to the same extent
Then	Single width parameter is sufficient
If	input data is variance scaled
Then	Deviation = 1
If	σ^2 tend to zero in RBF
Then	initial response function becomes delta distributed & normalized activation function tends to be tessilation function

Deviation (also called smoothing factor, variance, or slope of the kernel) is a scale parameter for distance ||x-c|| and reflects the spread of input data. It indicates the spiky nature of the profile(KB. 4).Ideally, the deviation should be such that Gaussians overlap with a few nearby centers and was calculated based on heuristics [121]. Different algorithms like explicit, isotropic and K-nearest neighbors (Appendix-A3) are employed[134]. Biharmonic spline function is used in calculating spread of the neuron.

Since, a single value of deviation is inadequate in complicated multi-dimensional data sets, variable values are employed for different kernels. But, generalized RBF has different widths and power terms in different neurons.A 1_D algorithm was employed to choose optimum deviation width among the user chosen and exhaustive search on all RBFs. Wang[199]used a guided random search based on boosting optimizationto calculate individual diagonal covariance matrix. In multi-scale RBF (which is similar to wavelet decomposition), very small and very large width values are chosen for inducing both local and global characteristics in RBF_activation. It is tested with a second order NL-I/O data set and terrestrial magnetosphere dynamic responses. It is more robust than LM procedure. The shortcoming, however, is large memory requirement to store the large number basis functions. Zhou [174] introduced Bayesian information criterion (BIC) to calculate fitnessfunction value of OLS by employing meta-heuristic, differential evolution. The widths of Gaussian function are optimized improving the generalization. The new automatic-RBF model is efficient to model nonlinear dynamic systems with known and unknown noise.

KB. 4b:	**Choice of width (scale) of RBF**
If	Non-linear dynamic systems
Then	RBF_NN with uniform width inadequate
If	Scale is small
Then	RBF_NN captures local characteristics of signal
	- Low general accuracy
	- Dips appear in interpolation
If	Scale is large
Then	NN captures global characteristics of signal
	+ Interpolation is smooth
	+ Good general accuracy
	- Ill conditioned training
	Remedy : multi-scale RBF_NN

8.3 Automation of RBF-architecture and parameters

The recent attempts of partial automation of RBF_NN follow. A linear averaging of class membership and probability distribution of training data is used [106] instead of popular k-means algorithm to select centroids and the radii of radial layer neurons. Here, SVD learning is found superior to quick propagation for Italian olive oil and PCB data sets. Cooperative-competitive genetic_evolution has been used for simultaneous refinement of centers and widths. Du[176]proposed a multi-ouput-fast-recursive algorithm in RBF to select centers and simultaneously optimize Ws.

It is computationally less complex. The method is based on reduction in the trace of error covariance of x. In Bayesian RBF_NN (Bayes-RBF_NN)[120], the optimization procedure is reversible jump Markow chain Monte Carlo (MCMC)-SAA, which performs a globalsearch in the joint space of parameters and number of parameters. Andrieu [120]applied it for number of neurons, regularization parameters, RBF parameters considering them as unknownrandom variables. The geometric convergence for full Bayesian model and SAA is established. The outcome is a more parsimonious model compared to RBF_NN and better approximation of errors. Acomparison of results between Bayesian model andpenalized likelihood model selection is made using Akaike informationand minimum description length criterion.

Alg. 11: SVR + Annealing robust learning algorithm+ RBF_NN	
▶ Initialize structure of ARRBFNs with SVR **While** not converged **For** all Tr data Compute ycal, residual **End** **For e**ach epoch Determine values of annealing schedule Update synaptic weights centers width of Gaussian Fn Detect influence of the outliers Discriminate outliers Compute robust objFn **While end**	**β** : deterministic annealing schedule; acts like cut-off points $Resid_i = yNN_i - yobs_i$ **Eqn.15** $logistic\_loss\_function_i(j_i) = \frac{\beta(epoch)}{2} * \ln\left[1 + \frac{Resid_i^2}{\beta(epoch)}\right]$ **Eqn. 16** $J = \frac{1}{NP}\sum_{i=1}^{NP} j_i$ **Eqn. 17**

SVR + Annealing robust learning algorithm+ RBF_NN

Fu et al. [227]proposed a Robust_RBF_NN using SVR for RBF parametersand annealing learning. Here, SVR regression optimizes number of hidden nodes, the initial parameters of the kernel and approximate weights of the network. SVR uses a quadratic programming optimization approach. A new objFn is introduced to detect outliers in the modeling of non-linear dynamic system. An annealing robust learning algorithm (Anneal.robust. Lrn) uses robust BP (Alg.11).

Simultaneous optimization of RL neurons and number of Tr points

Kahrizi and Hashemi [96]introduced 'neuron curve', a measure of learning curve to simultaneously find optimum number of radial layer neurons and minimum size of training dataset. The seismic refraction data is analyzed for first break picks using these two curves. The effect of noise on architecture of MLP and RBF also studied for the first break picking.

Simultaneous optimization of RL neurons and spread

Tetteh et al. [41]used biharmonic spline interpolation for optimization of neurons and spread parameter. It is applied to auto-ignition temperature of 232 organic compounds with six descriptors. .

⇨ Role of E-man in RBF_NN automation

Genetic algorithm is one of the widely used E-man (Evolutionary Mimics of Algorithms of Nature) [1] in all phases of automatic design and training of RBF_NN. A binary bit corresponds to the presence or absence of the connection between two processing elements (PEs) or neurons. During GA progress, the solution is evolved. The limitations are increase in the search space for large networks and getting stuck into illegal architectures. An indirect encoding method with heuristics is binary codification of grammar surmounting the problem of generation of unstable NNs. To start with, NNs are trained to surmount the problem of initiation and its tendency in getting trapped in a local optimal solution. But, near the optimum it becomes painfully slow and gradient based algorithms, regularized orthogonal LS are used to overcome the speed dependent disadvantages. However, the point of switching between the local searching and natural computational algorithm is still a stigma. Task dependent heuristics and meta rules based on numerical and simulation literature improves the functioning of hybrid algorithms. GA algorithm was invoked to optimize the centers and widths of RL neurons. Here, the center and width of the neurons are encoded in the chromosome which consists of a list of genes associated with a hidden neuron. A modified two point cross over is operated. It exchanges hyper volumes of input space instead of chunks of chromosome structure. This approach alleviates the functional equivalence problems. GA was used to locate the centers of RBF in the forecasting of time series data. They employed k-nearest neighbors for calculating width and singular value decomposition (SVD) for estimation of Ws. The whole set constitutes RBF_NN and during evolution the centers and width are refined. GA optimizes the number and approximate centers of the clusters, while k-means algorithm refines the location of centers. Here, GA is a support for another algorithm.Single- and multiple- object functions are used to select optimal subset of centers from training data.

Architecture_RBF: The modified discrete PSO was used to refine the architecture of RBFNN. Each particle was encoded as a string of binary bits associated with the centers width and weights. The particle of continuous PSO was encoded as a real string. It consists of real number denoting centers, widths and weights of RBFNN.

9.Offspring of RBF_NN

The networks derived from RBF_NN are probabilistic-, generalized regression- and competitive-, NNs (Fig.10). The first two mimic statistical framework, showing the imbibing character of NN paradigm.

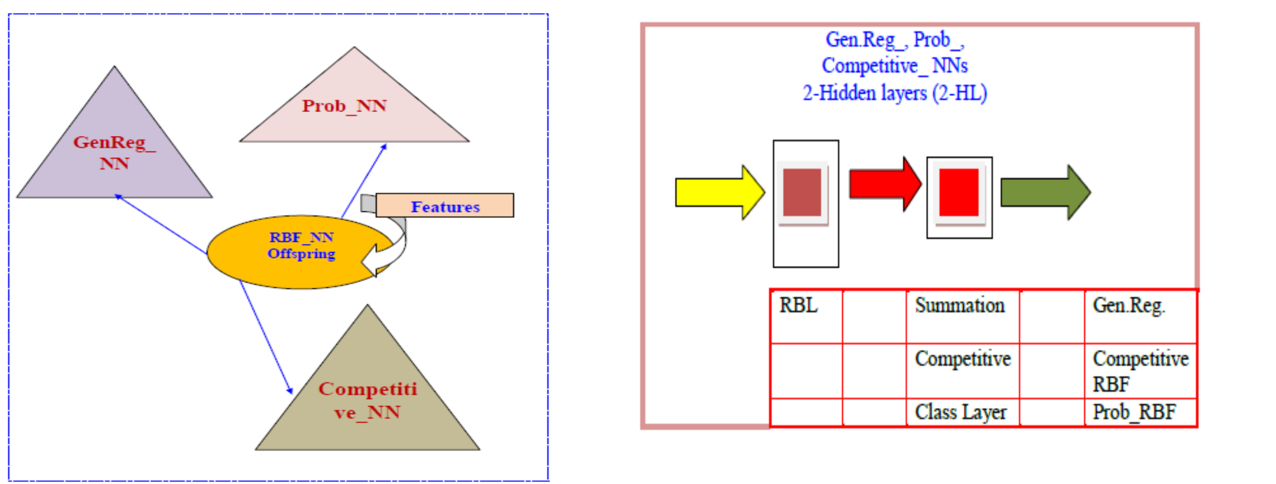
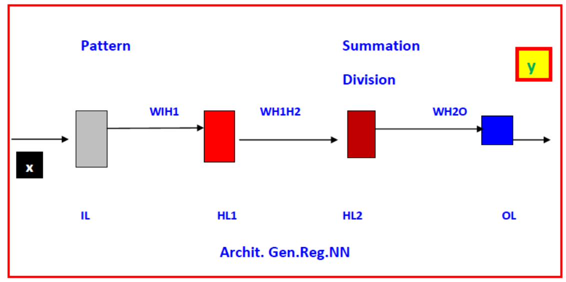


Fig. 10 Off spring of RBF_NN

9.1 Generalized Regression NN (Gen.Reg.NN, GRNN)

Gen.Reg.NNemulates statistical regression as it was introduced within the spirit of statistical frame. The architecture, input to commercial software packages (Trajan, Profession II,MatLab) and general numerical/ graphical outputs are described in Chart 26. Gheyeas and Smith [177] proposed a non-parametric GRNN-ensemble for single as well as multiple imputations in incomplete dataset during preprocessing step. The method is compared with 25 popular missing data imputation algorithms on 98 real world and synthetic datasets with various percentage of missing values. The results of GRNN, MLP and logistic regression are compared for a three_class problem.

Name	Gen.Reg.NN
Basis	RBF
Data	multi x, multi-y
Architecture	I# → RL→ Summation layer→output



Output of PII for GRNN

- Weight matrix
- Confusion matrix for each PE-output up to a maximum of 4
- Weight distribution

- RMS-Graph
  - Strip chart
- B-Coefficient
  - Bar graph

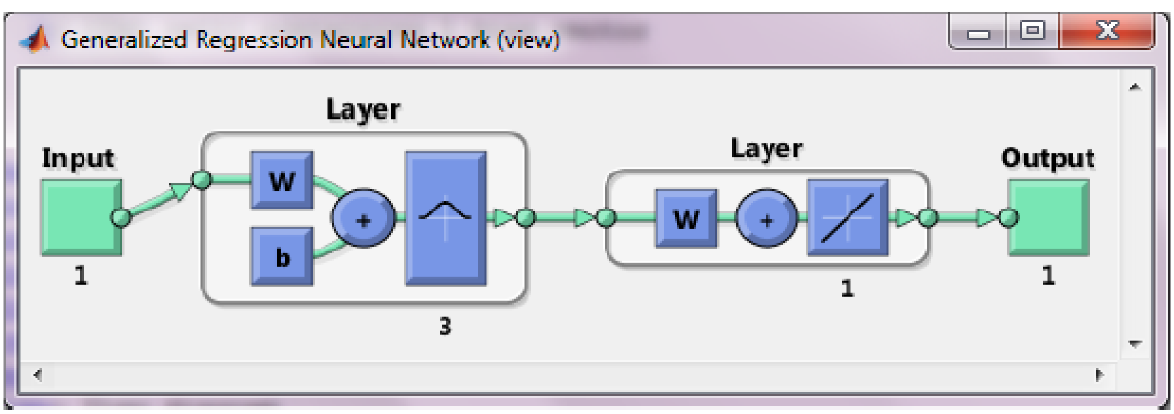
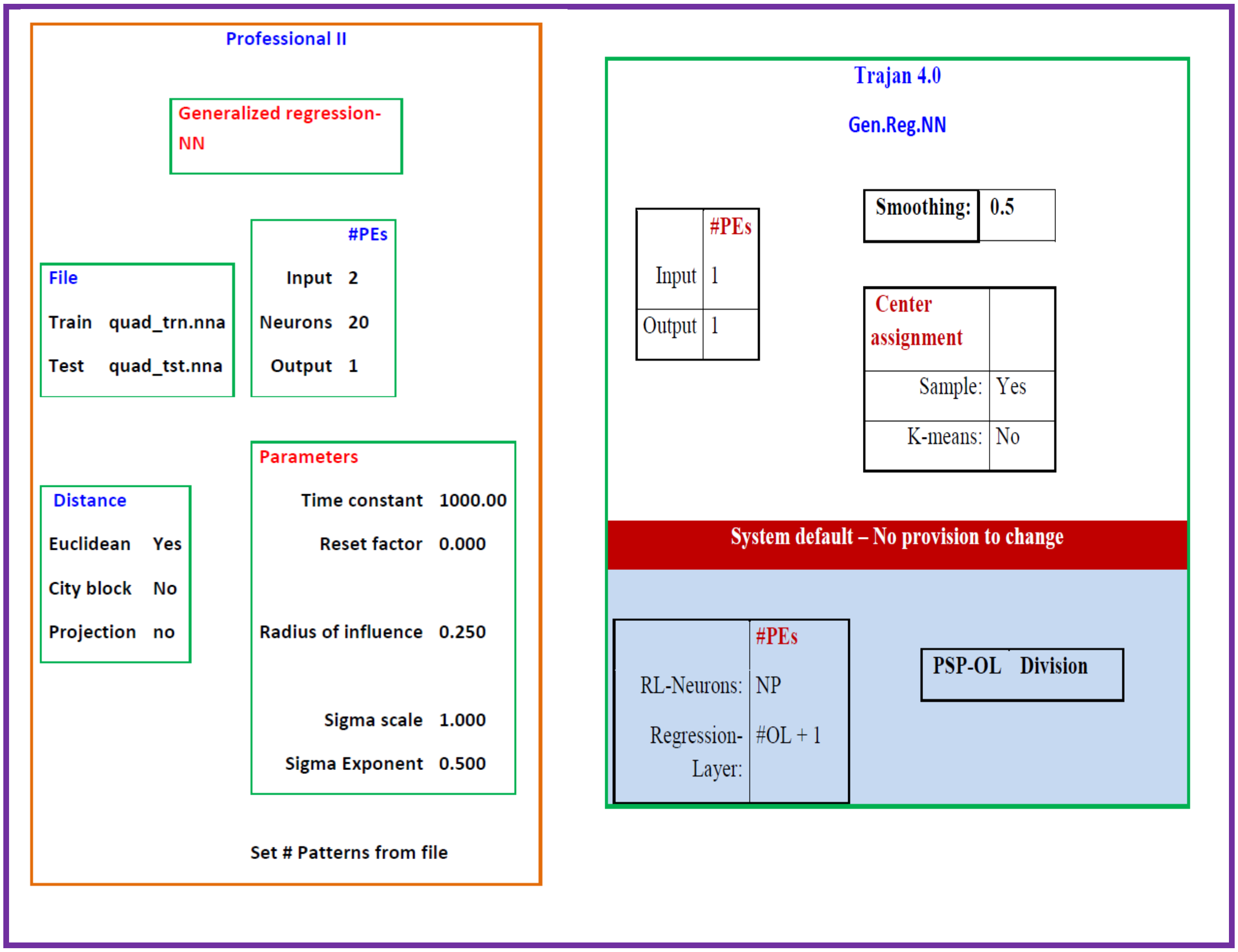


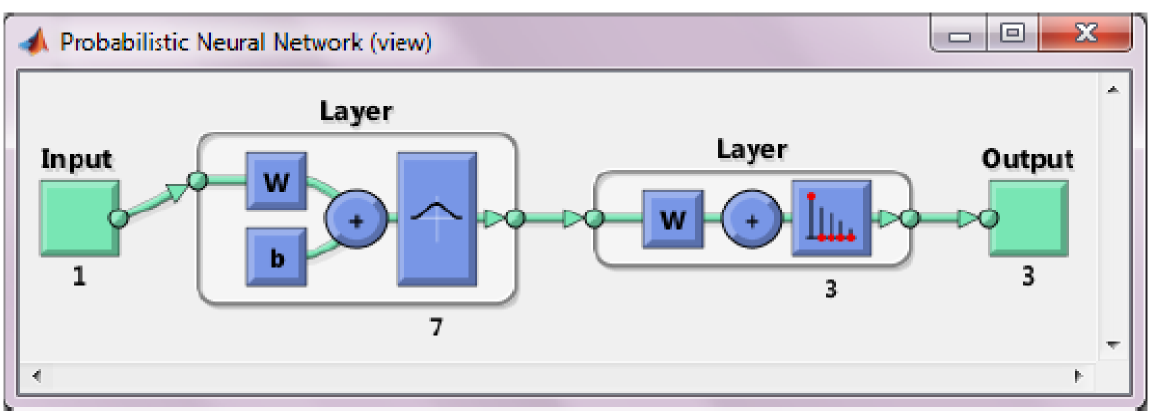
Chart 26: Gen.Reg.NN--Architecture, input to Trajan, Profession II, MatLaband general numerical/ graphical output

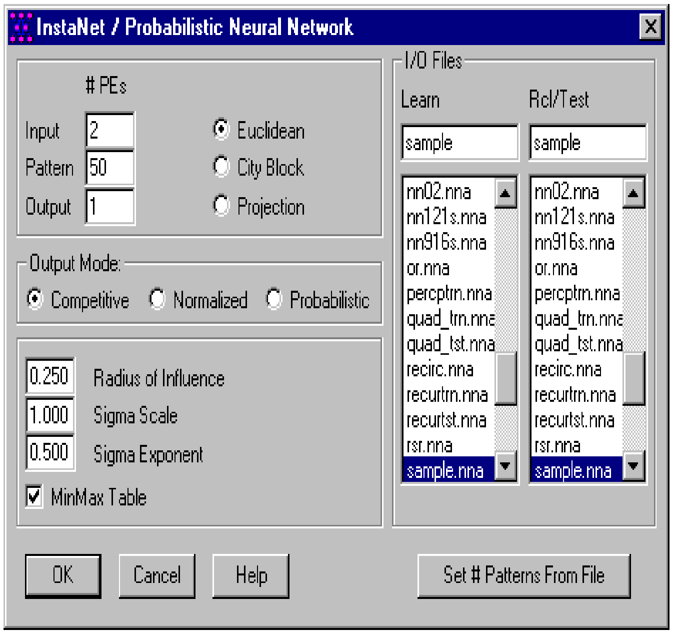


9.2 Probabilistic NN(Prob.NN, PNN)

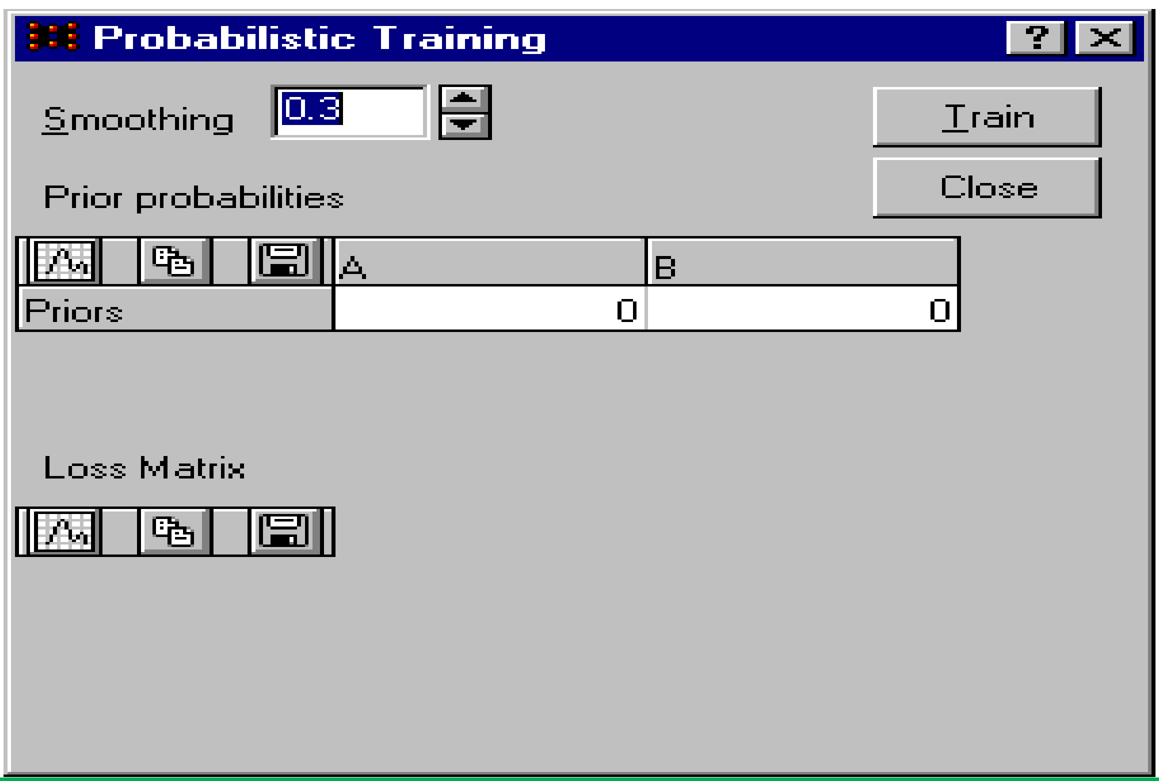
Probabilistic neural networkwas proposed for classification in the statistical distribution domain.Chart 27 displays characteristics and input to Trajan and MATLAB software.

Chart 27: Input of Probabilistic_NN in Trajan- and MatLab software





Name	probabilistic-NN
Basis	RBF, Each PE is a parzan kernel
Data	Numerical
Architecture	I# - RL- output
Convergence	Bayes optimal classification
Application	Sparse data



10. Add-ons from other domains

10.1 Novelty Detection

The novelty detection feature resident in ART_ARTMAP NNs of Grossberg and Carpenter is incorporated in self-organizing GRBF and resource allocating RBF (vide supra) with a facelift of functioning of RBF_NNs even in noisy environment.

Outlier and novelty

In classical statistics, a distinctly deviant experimental observation (beyond 3-σ_limits) is known as an outlier. Theseextreme values are of paramount importance in time series prediction and also the basis of emergence of extreme statistics. A set of consecutive points exhibiting a different trend (or set of outliersin one frame but corresponding to another process) is modeled separately. A popular instance in classical physical chemistry is ortho-effect of equilibrium constants and rate constants of substituted benzoic acid. Respecting Beer-Lambert's law of linear response of UV-Vis response of colored products in calibration, drift from linearity was deemed to be an outlying trend. But, Beer's law is applicable for a definite concentration range and non-linear calibration is the state-of-art-concept. A simplistic view is to consider two more intersecting straight lines of different slopes in a wide range of concentration. In cluster analysis, an outlier is called a singleton. Though, it appears to be rejectable, it may be the beginning of detection of novelty and a new real trend/ process. A few more patterns around it render the set of points to grow into another full-fledged cluster [134]. The concepts of outlier and novelty are two sides of the coin or the two eyes. With either of them, it is like seeing the visible world and ignoring the invisible altogether. The third eye is the perspective (paradigm), purpose and era of concern for complete knowledge.

Alg.12a: Novelty detection in RBF

Initiation
Invoke one RL neuron

Novelty detection

If	**en = abs (yn-yrbf) > eps & maxi < tow**
Then	**novelty**
If	**minj(dist(xi-ej)) > tol**
Then	**novelty**
	Limitation: It overlooks σ

Novelty detection in Neural networks

In ART-MAP and FUZZY_ARTMAP algorithms, Grossberg introduced a feature called novelty detection. Recently Chuang et al. [197]implemented it in robust RBF. Xu et.al. [89,108, 140,206] found upper bounds for the convergence rates of approximation error.Albrecht and Tavan[134]proposed a parameter α (>1) to detect a novel pattern. When the sum of α*TF for all neurons in the radial layer is smaller than the cut-off parameter, the response at the output layer becomes insignificant. But this pattern belongs to a new category or novelty (Alg. 12). The novelty detection parameter (for vigilance and cut-off) prevents the unknown pattern (outlier/novel observation) to be categorized in one of the existing classes.

Alg.12b: Addition of a neuron

If novelty detected
Then add a neuron

	Removal of neurons
If	3* (TF(x) for ith neuron < tol1
Then	ith neuron contributes less than tol for the data set
If	sum < tol2
Then	ith neuron has very small activation region i.e. it represents an over trained learning
If	ψ_i * ψ_j > tol3 (=1) ω j ≠ i
Then	Delete one of the two neurons **Reason:** both ψ_i and ψ_j have similar activation level

10.2 Robust RBF_NN

In classical RBF NNs of 1990s, the number of radial basis layer neurons grew (in an optimal model) withincrease of outliers.Chuang et al [197] proposedannealing robust RBFNN for function approximation task with outliers. Simulation studies with periodic, exponential and power functions establish the robust nature (Fig. 11) of algorithm to outliers (chart 29b, Alg. 13). The output of m-file [1k] for display of references from a suit of procedures developed in this laboratory is given for typical literature reports on robust RBF_NN.

Chart 29:Annealing Robust RBF training

Phase I
- ॐ **SVR with Gaussian kernel function**
  - ▶ Invoke initial structure (RB neurons Parameters of Gaussian fn, WRO)

Phase II
- ॐ **Annealing robust learning algorithms**
  - ▶ Cal Gaussian function parameters
  - ▶ Refine WROs
- + Fast convergence, speed
- + Robust against outliers
- + Surmounts initialization problems
- + Circumvents ill-effects of cut-off points in robust learning algorithm

Alg. 13: Annealing Robust RBF training [197]

Repeat Until convergence or termination criteria satisfied

Compute estimated result
Comp error for all training data

Update synaptic weights, centers, width of Gaussian function

Detect sequence of outliers
Discriminate sets of outliers

For each epoch, determine values of annealing schedule
Compute robust cost function

End repeat

Chart 29b: Output of om_ref_JAVATYP.m
Robust NNs in function approximation and classification tasks

D.S. Chen, R.C. Jain ▶ A robust back-propagation learning algorithm for function approximation	IEEE Trans. Neural Networks 5 (3) (1994) 467–479
C.C. Chuang, S.F. Su, C.C. Hsiao ▶ The annealing robust back propagation (BP) learning algorithm	IEEE Trans. Neural Networks 11 (5) (2000) 1067–1077
L. Huang, B.L. Zhang, Q. Huang, ▶ Robust interval regression analysis using neural network	Fuzzy Sets Syst. 3 (3) (1998) 337–347
C.C. Lee, P.C. Chung, J.R. Tsai, C.I. Chang ▶ Robust radial basis function neural networks	IEEE Trans.Syst. Man Cyber. 29 (6) (1999) 674–685
K. Liano ▶ Robust error measure for supervised neural network learning with outliers	IEEE Trans. Neural Networks 7 (1) (1996) 246–250
V.D. A. Sanchez ▶ Robustization of learning method for RBF networks	Neurocomputing 9 (1995) 85–94

om_ref_JAVATYP.m : **object module (om_) reference (ref_)**
Journal Author Volume And (JAVATYP) Title Year Pages
matlab function(.m)

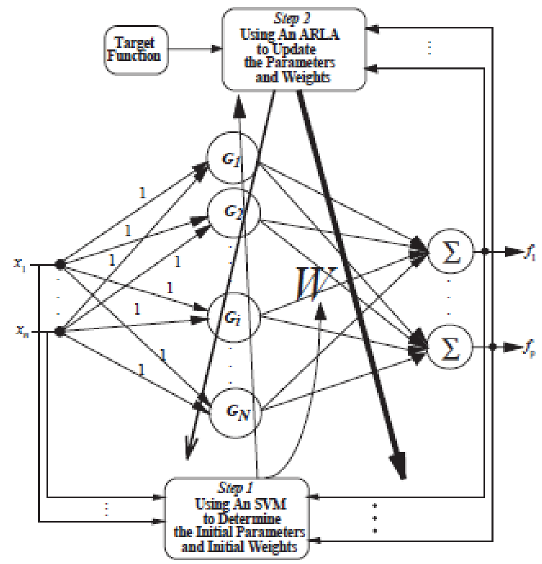
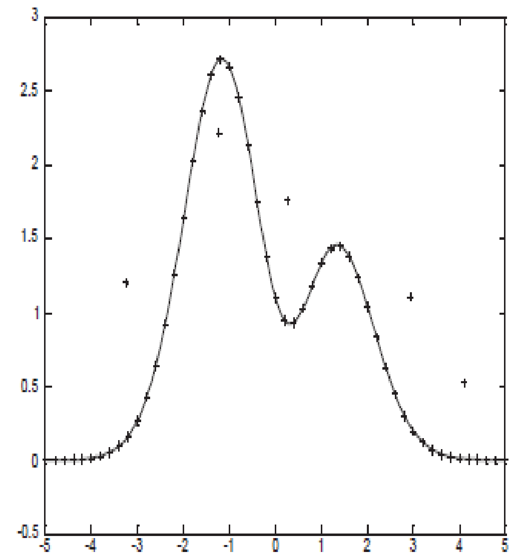


Fig. 11: Robust nature of Annealing Robust RBF to outliers in function approximation [Courtesy of Ref 197]



$y = 1.1 \cdot (1 - x + 2x^2) \cdot e^{-x^2/2}$ with $x \in [-5, 5]$.

NP = 51; Five outliers; epochs : 400

10.3Hybrid methods

Wavelet + RBF

Shoaib et al. [102] extensively studied prospects of sequential hybridization of FF_NNs with discrete/continuous wavelets(Chart 30) for rainfall runoff data. The coefficients of Daubechies(DB8) mother wavelet of continuous category are inputted to either RBF_NNor SLP with sigmoid TF. It resulted in superior models compared to component SLP and RBF NNs [235].

- Wavelet transformation simultaneously models both spectral (periodic) and the temporal information imbedded within rainfall–runoff time series data.

Chart 30: Mother wavelets with subclasses

Continuous	
Mother	**subclass**
Daubechies	[2, 3, 4, 5, 6, 7, 8, 9, 10]
Coiflets	[1, 2, 3, 4, 5]
Symlets	[2, 3, 4, 5, 6,7]
Haar	
Discrete	
Meyer	

EA + qRBF

Navarro [170]used q-Gaussian-RBF_NNs for binary classification tasks and eleven UCI classification datasets (sonar, breast cancer, liver, etc). A Hybrid algorithm consisting of EA and q-RBF is proposed to increase the discriminative power of RBF_NNs. The results are compared with Cauchy, inverse-multi-quadratic Gaussian RBFs and online-sequential extreme_learning_machine-RBF, SIG-ELM, RBF-ELM and Evol-ELM-RBF.

10.4 Ensemble

Yu et al. [184] reported multi-stage RBF_NN-ensemble to predict foreign exchange rates. A large number of single RBF_NNs are trained. The ensemble members are chosen by conditional generalized variance (CGV) method. Another RBF_NN is used for prediction.

11. Functional equivalence & Imbibing character of RBF_NN

The study of mathematical equivalence, functional identity/similarity of two procedures based on different (mathematical/ statistical/ fuzzy, data driven/ model driven etc.) philosophies/ paradigms is of utmost importance to probe into subtle differences to invoke a generalized procedure resulting in innovative tools. In recent times, there is an increasing activity of

⊗ proving equivalence of NNs with statistical/mathematical algorithms and
⊗ demonstrating the imbibing capabilities of NNs

ॐ Degenerates to Kernel regression method

ॐ RBF_NN imbibes @@@

RBF_NN imbibes Kernel regression [140] and in fact both are based on similar philosophies. In other words Kernel regression is a special case of RBF with hyper spherically shaped reception fields[140].

ॐ Norm_ RBF_NN equivalent to @@@

Self-organizing GRBF [134] is equivalent to Bayesian classifier. Normalized RBF is proved to be equivalent to Nadaraya Watson regression estimate (NWRE), a nonparametric kernel method. The empirical errors by KRE are equal to those from RBF. The equivalence between fuzzy expert systems (Fuzzy inference systems) and RBF was proposed. But, there is no strong theoretical proof.

ॐ Non-Normalized RBF

It is interpreted in terms of Tikhonov regularization theory[89].

ॐ RBF_NNgenerated from Markov chains

The numbers of states in the counter calculate the free parameters viz. center and deviation of RBF. The output pulse probabilities are related to input through a function, which is closely related to a Gaussian TF. The means and variances of Gaussians are controlled by the output of combinatorial logic function of the binary counter variables. The stochastic based on popular statistical mechanic models are used instead of k-means algorithm for centers and deviation. Thus, stochastic counters based on non-deterministic finite state machine (Markov chains) [92] approximate RBF_NN.

12. Emulation of standard mathematical techniques by RBF.NNs

ॐ Generation of Boolean functions

Generalized binary RBF_NN (Gen.Bin.RBF.NN) [196] computes every Boolean function that is possible with linear threshold neurons. It is introduced to construct Boolean functions that cannot be computed by a weighted (other than Euclidean) norm and to count the number of Boolean functions computable with each type of norm. The generalized binary RBF neuron is always as powerful as a linear neuron, although it excels in some cases.

ॐ Emulation of NAND gate

Sorel et al. [202] reported that recurrent RBF networks mimic analog computational devices and are efficient and reliable. They are comparable to perceptron NNs. Further, they are as powerful as finite automata (Fig. 12).

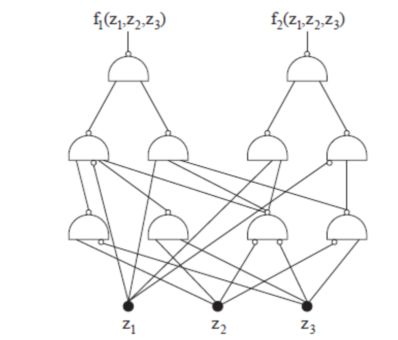


Fig 12 (a) NAND circuit with 10 units

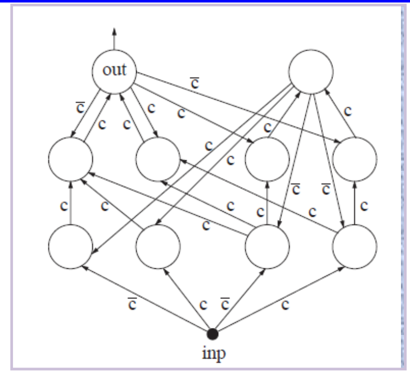


Fig 12 (b) RBF neuromaton (Courtesy of Ref 202)

ॐ Pseudo inverse of matrix

Hyper-RBF_NN[200] computes an approximation of pseudo inverse of Jacobian (J), even when J does not exist. It is thus, a theoretical tool to compute inverse of Jacobian (Fig. 13). Hyper-RBF_NN learns to compute a locally linear approximation of the Jacobian pseudoinverse at each value of parameter.

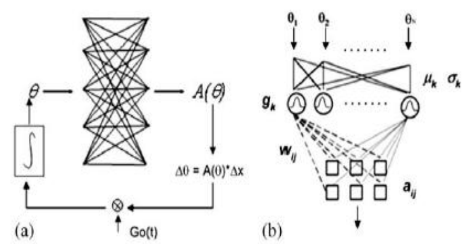


Fig 13: Calculation of Pseudo inverse by Hyper_RBF_NN (a) architecture (b) Cal of elements of estimated J^+
(Courtesy of Ref 200)

ॐ Generation of finite difference type formulas

Polynomials generate finite difference (FD) and compact FD stencils for 1-D profiles. Now, RBFs are shown to generate the formulae for m-D scattered node layouts[100]. Further, RBF-FD, RBF_HFD formulae become equivalent to standard FD and HFD when ε> 0. RBF-FDs do not require global meshes, which are indispensable for finite element methods. The computational time and algorithmic complexity does not explode with dimensionality and geometric complexity of the methods. Elliptical equations dramatically improve the accuracy of RBF-FD formulae and preserve the diagonal dominance. An in-depth research will throw light on solution of many other classes of partial differential equations (PDEs) in RBF-HFD paradigm.

ॐ Emulation of PLS

RBF_NN emulates PLS and the procedure is briefed in chart 31. It is applied to simulated batch polymerization reactor (chart 32) data. The rival penalized competitive learning method arrived at six RB neurons.

Chart 31: Emulation of PLS and application to polymerization reactor

- Phase 1:RBF_NNextracts the latent structures of augmented X and Y matrices [X,Y]
- Phase 2:Two FF_NNs calculateoriginal (inputted) predictor and response variables
  - Architecture: Neurons 8,6 ; TF_hidden layer : sigmoid; TF_outputLayer : linear

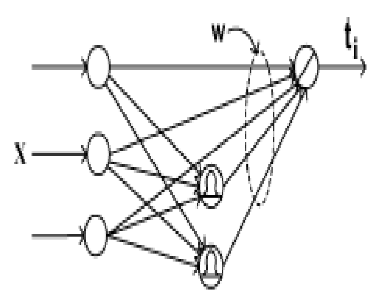


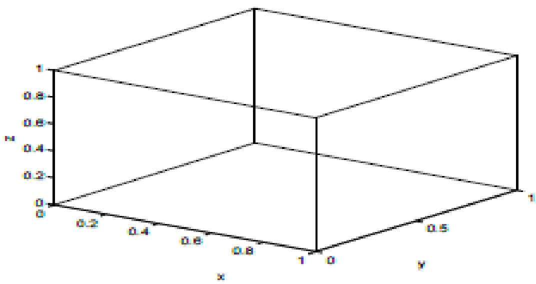
Chart 32: Dataset.**Simul. Batch Polymerization Reactor**

Dataset.simul
time= uniform[-1:1];
noise = randu([-0.1,0.1])
Tr(aining) : 100;
Va(lidation) : 100;
Te(sting) : 100;

$$y = x_1^2 + x_1 * x_2 + 3 * \cos\ x_3 + noise_4$$

$$x_1 = t^2 - t + 1 + noise_1$$
$$x_1 = \sin(t) + noise_2$$
$$x_1 = t^2 + t + noise_3$$

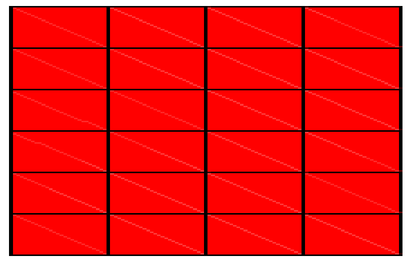
[**Third order tensor (or 3-way data)]**



X

Batches x process_variables x Time points[30 x 4 x 5]

[Second order tensor (or matrix or 2-way data)]



Y

Batches x Response[30 x3]

13. Theoretical proof characteristics of RBF_NN

Xu [140]proved constructively the existence and consistent estimator for RBF_NN. He derived the upper bounds for convergence rates of approximation error as a function of radial layer neurons. The upper bounds for L2 convergence rates of best consistent estimator are given in the case of number of patters (NP), but radial layer (RL) neurons tend to infinity. Further, a theoretical investigation of selection of width (size of receptive field) and influencing factors is conducted. The future research will be around the study of type I and type II errors resulted when LS is used in estimation of W in RBF_NN.

Computational power and complexity: The computational properties of MLP are well understood. But, for recurrent-RBF_NNs, the partial results available are from the study of hybrid RBF and perceptron layers. Sorel and Sima [202] proved that recurrent-RBF_NNs are at least as powerful as finite automata. Thus, recurrent-RBF-neurons can be called RBF-neuromata.

Error propagation in RBF_NN: Derks et al. [43]and Faber et al. [42]reported that error propagation in NNs is a delicate issue especially in presence of large noise and strong violation of error propagation constraints. An in depth investigation focusing torch on earlier literature studies dealing with error propagation in PCR and PLSR brings forth a clarity in the confidence intervals rather than point estimates. The latter (confidence contours) kept mathematical/statistical models in a high pace for several decades. Derks [43] studied the effect of random errors in input to RBF_NN and MLP-NN models for the linear and quadratic simulated data sets and experimental data of mechanical properties of PET yarn. The range of inputs, their connection structure and configurations of network architecture all influence the sensitivity in a complicated manner. The noises of different distributions at various levels through Monte Carlo simulations are added to the input and robustness of RBF was studied. Faber and Kowalski[42] reported that RBF is less sensitive to random errors in the input.

14. Universal function approximation theorem for RBF_NN

The approximation power of NNs using Gaussian activation function was extensively investigated [125]. The seminal studies of Poggio and Girosi [211] on regularization theory for approximating the RBF_NN for a variety of functions and plausible explanation in biological systems brought RBF_NN to the forefront of supervised learning NNs. The consequence in the application front is realization of RBF_NN modeling of multi-dimensional non-linear hyperspaces. Arteaga and Marrero [109] replaced translational operator of RBFs with Delsarte , of course with different smoothing factors in different kernel nodes. This imparts universal function approximation characteristic for images in weighted image-spaces.

15. Inversion of RBF_NN

In the first instance, 2-D XOR task was trained with RBF architecture of 2I# -2RL# -1O#. The inversion of RBF_NN was studied the by transforming the task into NLP followed by inversion with minimal and maximal single element method (Imin, Imax). It is reported that either minimization or maximization of object function, solution x1_invRBF, x2_invRBF obtained from inversion of RBF are in conformity with the expected values viz. 0,1 or 1,0. This is the simplest case but conceptually a breakthrough in focusing attention to tackle complex real life inverse tasks. Behera et al. [221] used extended Kalman filter (EKF) to obtain inversion of trained RBF_NN model of trajectory tracking control of a two-link manipulator.

16. Current state of RBF_NN in research mode (2014)

Clifford algebra multi-dimensional NNs (CAMD-NNs)[200]are co-ordinate free systems. It emulates real (RBF-,MLP-), complex- and quaternion- valued NNs. The application of SVMs within the framework of geometric algebra promotes multi-dimensional learning. And, support vector machines also generate RBFs and optimal parameters in geometric algebra. RBF_NN is a special case of alternative maximum expectation (Max.Expect.) model with constraints.

The functional details of typical neural network packages are discussed earlier [1j]. The results of each category of popular NN_modelswith synthetic data for chemometric tasks along with method/knowledge base will be published [59]. Radial basis function belonging to kernel group occupied a niche in mathematical and applied sciences. A few in the long list of RBFsplaying key role are UV-VIS/ IR-spectra, chromatographic profiles, additive Gaussians in multicomponent systems,product Gaussians to model orbitals in quantum chemistry (GTOs), noise in many natural processes etc. In, neural networks discipline, RBF_NN was proposed just after SLPs and its importance (Appendix-A4) is noteworthy. Even a bird's eye view is Herculean task andhere a method-base format (Chart 33) is chosen to focus current state-of-RBF_NNs awaiting research mode software.

Chart 33State-of-art-of- RBF_NNin research mode

Hidden Layers	Hidden Layer. Neurons	TF_RBL
Default : [1]	Default : [1]	Default : normal
2	dimX	
	rand_int([1 to NP]	Method_base.TF
	custom[User_chosen]	

Learning/Training

Phase-1	Phase 2	Phase 3
RBF_par	Connection weights	Simultaneous
Center		
Width	WHO	Center

Width
WHO

INPUT

Datatype
Numerical
Categorical
Image
Geographic

Datatype
Numerical
Binary
Multivalued integer
Floating point
Real
Complex

Datatype Numerical
Complex
Imaginary (a + i* b)
Quaternary
(a + i* b +j*c + k *d)

Datatype
Image
Pixels
Voxel

Datatype
Geographic
Points,
Lines
Polygons

Architecture

Pruning Addition/growing

Hybrid RBF_NN

(Binary) Hybrid RBF_NN
None
Fully_Organized-SO- ART

MethodBase

Methodbase_Center

Methodbase_width

[Center, width]
MB :

[Center, width, WBO]
MB :

Methodbase_ Transfer Fns

Rules/Knowledge/Intelligence

Intelligent_Swarm (NNs) Intelligence (IntSwarmInt)

Ensemble

Choice
Default
Custom (User_Chosen)
Expert

Knowledge-Information (intelligence)-Data(KID)_base of Datasets

Datasets
Experimental
Simulation

Data
Training
Testing
CV

Datasets. A priori Information
Research Results
Methods used
Knowledge bits
Intelligence [process_specific, swarm]

Task base

Tasks
Interpolation
Function approximation
Parameterisation
Classification
Time Series

Time series
Modeling
Prediction

Prediction
Hindcast
Nowcast
Forecast

Forecast
One-step ahead
Multi-step aheda
Forecast

17. Future scope

The track record shows a genius comes out with a new paradigm steeringscience in more precise and right direction approaching dynamic nature. Such mile-stone happenings of new scientific covers are real and evolves into merging with truth viz. atoms are divisible against their indivisibility, parallel computation vs. sequential execution, nature-inspired/swarm- intelligence vs. algorithmic programming, artificial neural networks vs. biological NNs in brain, LOTUS-1-2-3 vs. component techniques, airplane mimicking bird fly, and so on.The support, finances, goals (but with no assured/desired outcome) and leadership result in miraculousinnovative discoveries

- Experimental evidence of boson at six sigma level confidence in CERN after 50 years of its theoretical proposition by Higgs
- Complete mapping of Human genome
- Robotic tele surgery
- Experimental studies of a single cell, single molecule, single atom

The boson research in the coming years rewrites the twentieth century laws of physics and opens new vistas in unveiling the unsolved riddles and moving ahead to be nearer than before to the truth of truth.User chosen, machine arrived strategies (fast/current-research/machine-research-background-automatic recovery and continuation-after-switch-off-and-switch-on) are awaited.

Future explorations in RBF_NN include use of general norms including Euclidean and finding the upper bounds of computational power of analogue-recurrent-RBF_NNs. Similar results of simulating Turing machines are known for analogue-perceptron-NNs. Plug and play mode algorithms under a transparent platform with data/function/noise simulators have multifold utility in hypermedia augmented instruction

(HAI), exploratory research, modifications and simulation of user desired sequence of operations and cropping up of newer robust/efficient algorithms.

Acknowledgements

RSR thanks Sri S. S. Somayajulu, B A, M Sc., B Ed., for his inspiring and knowledgeable lectures in under graduate study in physical chemistry. His time consciousness, speed, virility, attention drawing punctuality and provoking instinct for discipline in the lectures is evident with his sparkling epilogue for completion of a lecture that 'we have five more minutes to complete this period, let us solve two more problems'.

RSR appreciates Prof A Satyanarayana for his contributions for over a decade during the early phase of his (ASN's) research pursuits. He deserves a forte for his innate flair of penetrating unfathomable arena of chemical logic, computer code and even bringing life and leaving a marked signature to the solution of a research pursuit.

We acknowledge the patience of R C S Rama Sastry and P D Sravana Kumar, post-graduate students of chemistry (during 2006-2009) to take down part of preliminary draft of the manuscript.

Appendix A1- Evolution of Radial basis function (Evol. RBF)

The changes in 2D- and 3D-profiles of different transfer functions with variation of their parameters in a wide data range produce a galaxy of shapes (hyper surfaces) including patches. Briefly, this is the output of a single neuron. But, NN consists of several neurons interconnected with different magnitudes of weights (Ws). Thus, their accumulation and passing out of output layer through hidden layers results in signals of any shape tilt and breaks. Their visualization in 1D-, 2D-, 3D- (orthogonal, non-orthogonal) axes, scaling [{[0 to 1], multi-dimensional}], transformation ([PCA, PLS]), [FT, Hilbert etc.]), rotation ([perspective, along a chosen/ chemical/ physical axis….]) and so on is a sought after activity for human comprehension. Also, it is more crucial in pattern recognition/ matching, fault detection and sensory perception in automated systems/autonomous robots. The computations involving translating into wavelets, quaternion, fuzzy, Fourier domains and bringing back the final results into measurement space is another intelligent approach called inverse transformation/modeling.

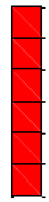
The non-linear transfer function in hidden layer(s) of NNs projects input profiles into non-linear output streams. The inherent characteristics of TFs render transfer of high dimensional space into lower ones, correlated variables into (partially/completely) independent. Also, depending upon the architecture, a low dimensional input is projected onto higher dimensional space. They not only mimic classical statistical/mathematical (linear) procedures but also go the extra mile. For instance, with RBF, the output of a radial layer neuron is traditional Gaussian. But, with modified RBFs, the profiles are flat at peak, broad/non-asymptotic tails etc. For, example with sigmoid function, the output is S-shape, hard limiter, straight lines parallel to x-axis, y-axis, diagonal etc. The linear discriminant analysis (Lin_DA or LDA) and quadratic DA (Quad_DA or QDA) are mimicked in the first fewiterations, but afterwards classify clusters/classes of data points with more non-linear discriminating boundaries. In regression, it imbibes linear least squares (Lin_LS or LLS), quadratic LS (Quad_LS or QLS), but mimics several other non-linear procedures. It is a consequence of universal function approximation theorem for sigmoid, radial basis functions in Euclidean space, of course with a few constraints. Thus, NNs and/or E-man modules are miles better than traditional mathematical techniques.

Data structure.RBF:For an input matrix of explanatory/feature variables (X) and vector/tensor of response (y), the parameters of general radial basis functions are given in chart A1-1.

Chart A1-1: Data structure and architectures of RBF_NN

Variable	Column Vector	Abbreviations and full form

$$y \quad : \begin{bmatrix} y_1 \\ y_2 \\ y_i \\ y_{np} \end{bmatrix}$$



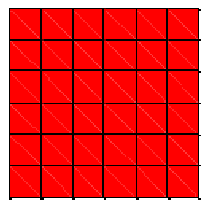
y(np,1)

np	:	Number of points(patterns)
y_i	:	Response of ith point
x_{id}	:	Value of i^{th} variable in d^{th} dimension

y=fn([X;par])

$$X: \begin{bmatrix} x_{1,1} & x_{1,2} & x_{1,d} \\ x_{2,1} & x_{2,2} & x_{2,d} \\ x_{i,1} & x_{i,2} & x_{i,d} \\ x_{np,1} & x_{np,2} & x_{np,d} \end{bmatrix}$$

MATRIX



X(np,dimX)

Data

$$\begin{array}{c} \\ NP \end{array} \begin{bmatrix} 11 & 12 & 13 \\ 21 & \mathbf{XX} & 23 \\ i^{th} & 32 & 33 \\ 41 & 42 & 43 \end{bmatrix}^{\dim X}$$

$$\mathbf{x} = \mathbf{XX}(i,:)^{T}$$

$$center: \quad {}_{nclusters}\begin{bmatrix} c_{1,1} & c_{1,2} & c_{1,3} \\ c_{2,1} & c_{2,2} & c_{2,3} \end{bmatrix}^{\dim X}$$

$\text{Center}_j = [Cz(j,:)]^T$

$$\text{covx } or \ width:$$

$$_{\dim X}\begin{bmatrix} \text{var}_{1,1} & \text{cov}_{1,2} & \ldots\text{cov}_{1,d} \\ \text{cov}_{2,1} & \text{var}_{2,2} & \ldots\text{cov}_{2,d} \\ \text{cov}_{i,1} & x_{i,2} & \ldots.\text{cov}_{i,d} \\ \text{cov}_{d,1} & \text{cov}_{d,2} & \ldots.\text{cov}_{d,d} \end{bmatrix}^{\text{dimX}}$$

$$diagonal\_matrix(DM):$$

$$_{\dim X}\begin{bmatrix} \text{var}_{1,1} & 0 & \ldots 0 \\ 0 & \text{var}_{2,2} & \ldots 0 \\ o_{i,1} & 0_{i,2} & \ldots.0_{i,d} \\ 0_{d,1} & 0_{d,2} & \ldots.\text{var}_{d,d} \end{bmatrix}^{\text{dimX}}$$

If variances are all equal

Then $\text{var}_{i,i} = \text{constant}$

$$_{\dim X}\begin{bmatrix} \text{varc} & 0 & \ldots 0 \\ 0 & \text{varc} & \ldots 0 \\ 0_{i,1} & 0_{i,2} & \ldots.0_{i,d} \\ 0_{d,1} & 0_{d,2} & \ldots.\text{varc} \end{bmatrix}^{\text{dimX}}$$

If	Width parameter is same in all dimensions
Then	Hyper spherical ball, radially symmetric, constant width used in interpolation and SVM
If	Width parameter changes in different dimensions
Then	Elliptical ball, not radially symmetric, axes of hyper ellipsoid parallel to feature space
If	Non-diagonal matrix
Then	Shape and size of ellipsoids/patches are arbitrary in feature space

$\underset{x}{\begin{bmatrix} x \end{bmatrix}} - \underset{center}{\begin{bmatrix} c \end{bmatrix}} = \underset{dev}{\begin{bmatrix} dev \end{bmatrix}}$	**Eqn. A1-1**
$z_i = \left[\dfrac{x_i - c_k}{width_k} \right]$ **Standardized input**	**Eqn. A1-2**

norm	
$\lVert dev \rVert = norm\ \ dev\ = L2norm\ \ dev$	**Eqn. A1-3**
$= \sqrt{\left[\ dev\ ^T *\ dev\ \right]}$	
$= \sqrt{\left[\ x - center\ ^T *\ x - center\ \right]}$	

Euclidean distance	
$dist.Euclidean = \sqrt{\ x - c\ ^T *\ x - c\ }$ $= norm\ \ x - c$	**Eqn. A1-4**

Kernel function

A kernel function is a non-linear local/global weighting function and used to control the influence of distant points. These are typically positive functions that integrate to one and have compact support. In other words they are only positive for "local" cases. A kernel function with two input variables exhibit mountains (hills) or valleys (craters) in a 3D- surface plot. Three parameters viz. top flatness, (R) steepness of the hill (S) and shape of the base of the hill (A) govern the profile.

Distance functions: In two or higher dimensional space, the distance between a point and another point (or cluster center) is calculated by a metric, generally Euclidean distance.The other popular distances are Mahalanobis, cityBlock, etc. The generalized distance is $\ x - c\ ^T * \mathrm{cov}\, x^{-1} *\ x - c\ $. In statistical parlance, it is covariance matrix which is obviously the positive definite and also symmetric.

Admissible radial basis functions

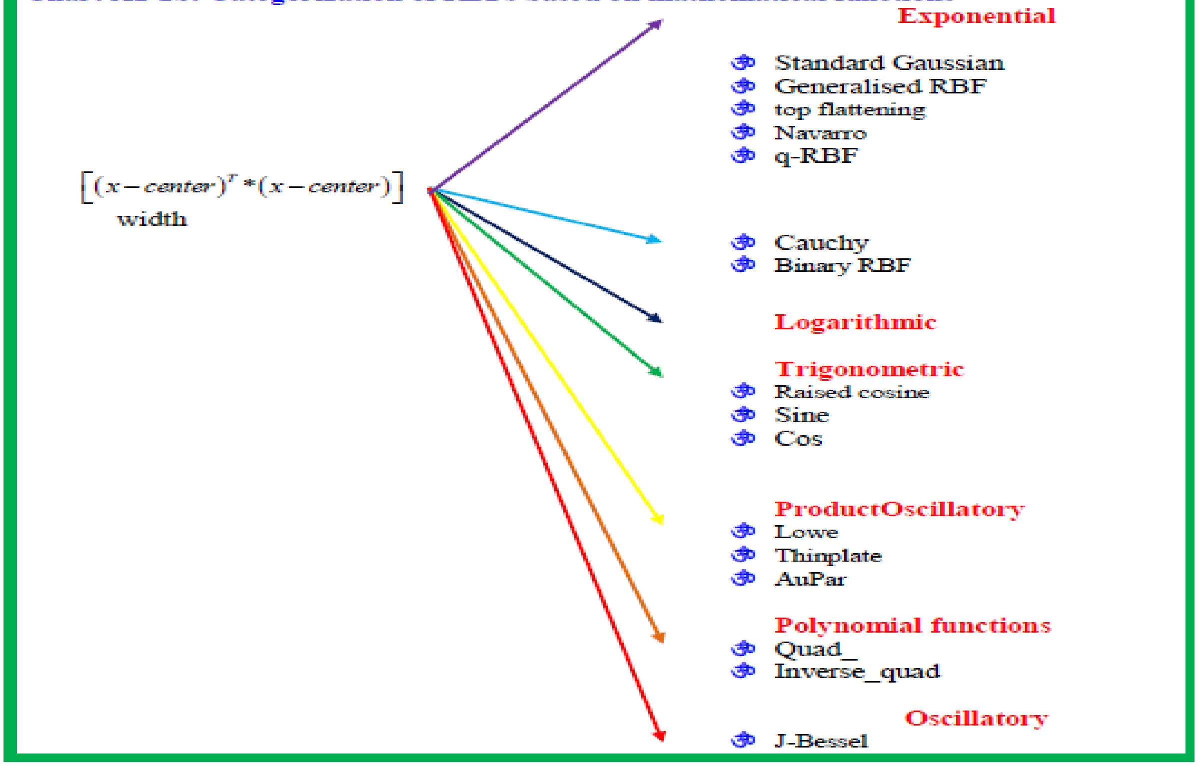
A radial basis function has local field with maximum response at one point and fast decaying away from it. The axioms needed for a function to be RBF are given in Chart A1-2a. The first three are mild mathematical restrictions and fourth and fifth together will render it to be admissible RBF in the strict sense (KB. A1-1). The RBFs in vogue based on mathematical terminology are collected in chart A1-2b.

KB. A1-1: Mild and strong conditions for admissible RBF

If		Five conditions (chart A1-2a)are satisfied
Then		Function is admissible in strict sense as RBF
	Elseif	Only first three axioms are satisfied
	Then	Admissible as RBF in the mild sense, but not in the strict sense
	end	
Else		Function is not at all suitable as RBF
end		

Chart A1- 2(a) : Axioms required for RBF	
	(a) Mild conditions for a function to be RBF
Axiom	
1	RBF is a continuous function in x-space in the range of $0\,to\,\infty$ such that $out.RBneuron(i,j) = h(i,j) > 0\ \forall\, x\ and\ \forall\ cluster\ centers$
2	It is a monotonously decreasing function of z in the range $0\,to\,\infty$ i.e $\frac{\partial h(i,j)}{\partial x(i)} = gradient\ \ h(i,j)\ < 0$ $If\ \ \left\Vert x_i - c_j \right\Vert^2 < \left\Vert x_k - c_j \right\Vert^2$ $then\ \ h(i,j) > h(k,j)$
3	Gradient of function is monotonously increasing in the range of $x \in\ 0\,to\,\infty$ and $\frac{\partial^2 h(i,j)}{\partial x(i)^2} = Hessian\,(z) < 0$ $If\ \ \frac{\frac{\partial h(i,j)}{\partial x(i)}}{\left\Vert x_i - c_j \right\Vert^2} > \frac{\frac{\partial h(k,j)}{\partial x(k)}}{\left\Vert x_k - c_j \right\Vert^2}$ $Then\ \ function\,is\,RBF\,in\,the\,mild\,sense$
	$If\ \ \left\Vert x_i - c_j \right\Vert^2 > \left\Vert x_k - c_j \right\Vert^2\ \&$ $\frac{\frac{\partial h(i,j)}{\partial x(i)}}{\left\Vert x_i - c_j \right\Vert^2} > \frac{\frac{\partial h(k,j)}{\partial x(k)}}{\left\Vert x_k - c_j \right\Vert^2}$ $then\ \ \ function\,is\,RBF\,in\,the\,strict\,sense$
	(b) strong conditions for a function to be RBF
	Conditions in (a) and
4	$Lim_{z \to 0}\ RBF(z) = L$, where L is a finite number
5	$\frac{\partial z(i,j)}{\partial x(i)}\ \ + 2 * z * h(z) > 0,\ z \in [0\,to\,\infty]$

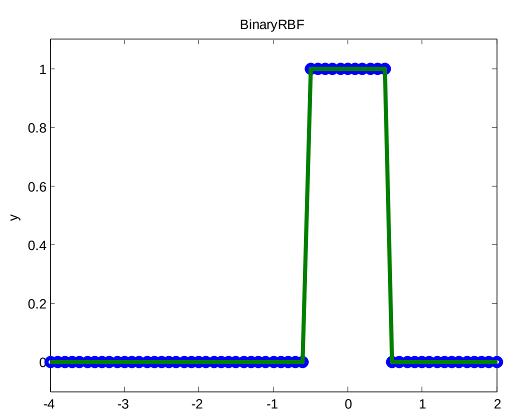
Chart A1-2b: Categorization of RBFs based on mathematical functions



▶ Binary RBF

A RBF (vide supra) operated on a floating point input results a real value. But, it can be rendered binary by comparing with a threshold (chart A1-3) and found applications in classification, discrete signal processing and many other disciplines.

Chart A1-3: Binary RBF



$$RBF\_binary(x) = \begin{cases} 1 & if \ \|x - center\|_A \le width \\ 0 & else \end{cases}$$

Eqn. A1-5

```
%
%  om_binrbf.m   18/10/03   9:06
%      F. Friedrichs, M. Schmitt,
%  Neurocomput., 2005, 63, 483-498
function [y] =
om_binrbf(x,center,width,fig)
[r,c] = size(x);
one = ones(r,1);
dev = (x-center*one);

for i = 1: length(dev)
if norm(dev(i,1)) <= width
       yRBFbinary(i,1) = 1;
else
       yRBFbinary(i,1) = 0;
end
end
```

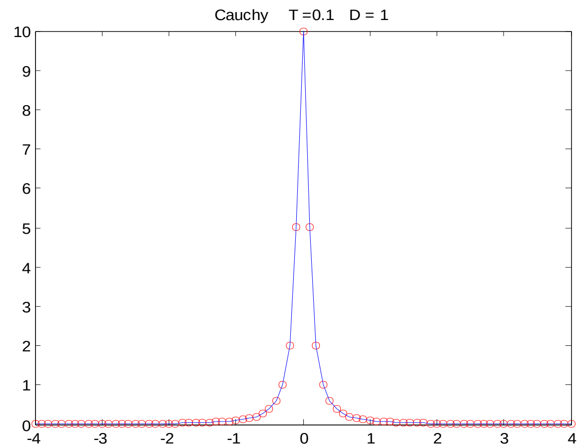
▶ Cauchy

It is used as a TF in RBF_NNs. T is free empirical parameter and D is a power dictating the shapes of profiles (chart A1-4). This function is derivable from q-distribution (chart A1-12).The Cauchy_RBF _NN has been successfully applied to image retrieval and Computerized Tomography.

Chart A1-4: Cauchy Fn

$$yCauchy = \frac{T}{\left[T^2 + x^2 {}^{\left(\frac{D+1}{2}\right)} \right]} \quad \textbf{Eqn. A1-6}$$

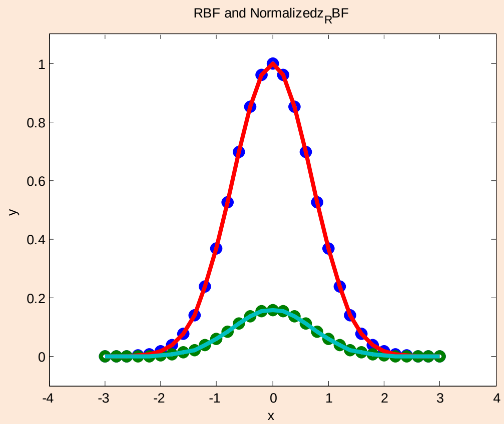
```
    T = 0.1; D = 1;
    lt=4;
    xi= -lt;xinc=0.1;xf=lt;
    x = [xi:xinc:xf]';
[ycauchy]=om_cauchy(x,T,D);
```




```
%
%   om_cauchy.m    (R S Rao 13/10/03)
%
function [ycauchy] = om_cauchy(x,T,D)
[r,c]=size(x); one = [ones(r,c)];
ycauchy  = ([one*T]./[(one*T).^2+x.^2].^[(D+1)/2]);
```

▶ Radial basis functions (RBF)

RBF is an example of a kernel group of functions. It is a non-linear function of a distance measure (chart. A1-1). In other words, if interpolation probability of



RBF (blue) and normalized RBF (red)

$$RBF(x):$$
$$NLf\left[\ dist(x, center\ \ , width\right]$$

RBF in matrix notation

$$y\_RBF = \exp\left[-\ \ x - c\ ^{T} * \operatorname{cov} x^{-1} *\ \ x - c\ \right]$$

if Euclidean distance

Ec

Eqns. A1-(A): Univariate and multi-variate RBFs

Univariate RBF

$$yRBF = \left[\exp\left\{-\left(\frac{x-c}{sd}\right)^2\right\}\right]$$ **Eqn. A1-7**

$$term2 = \frac{1}{\sqrt{2 * \Pi * width}}$$

$$yRBF\_norm(x) = term2 * yRBF$$ **Eqn. A1-8**

$$yRBF\_norm(x) = \frac{1}{\sqrt{2 * \Pi * sd}} * \left[\exp\left\{-\left(\frac{x-c}{sd}\right)^2\right\}\right]$$

$$RBF(x) = \begin{cases} 1 & if\ \ x = center \\ 0 & if\ |x - center| > 3 * SD \\ 0 < RBF(x) < 1 & if\ |x - center| < 3 * SD \end{cases}$$

Eqn. A1-10

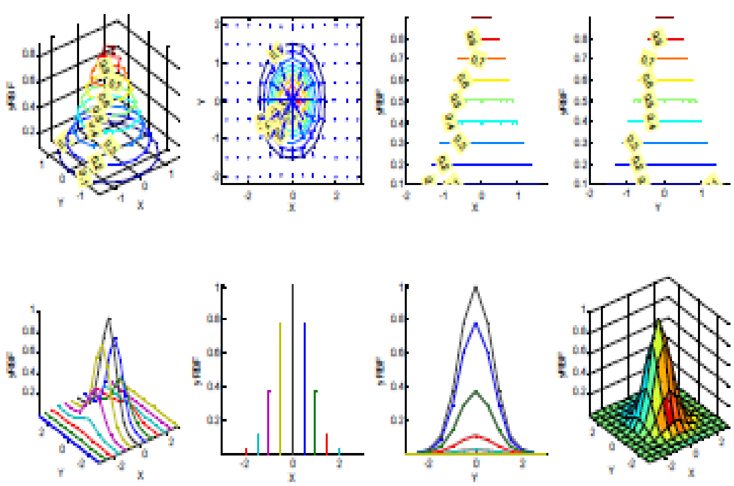
a function has unique solution for any closed point, then it is called RBF.In mathematical analysis, this popular function was used in strict interpolation of data in multi-dimensional space [139]. If the interaction between the elements of domain occurs only through inner products, then RBF (kernel) methods are good alternatives.

Broomhead [48,189] employed RBF for the design of NNs in the context of interpolation tasks. Gaussian function exhibits maximum response at the center and decays fast on either side of it.

It is integrable and meets Park and Sandberg approximation and also referred as Gaussian function. But, exp[(norm(x-c))/s] is not integrable, although it adheres the relaxed conditions. The variation of the center and standard deviation (sd) of the cluster on the changes in the shape of RBF are depicted in chart.A1-5for 1D- and 2D- X-space. The standardization of this function is done by multiplication of yRBF with $\frac{1}{\sqrt{2 * \Pi * \text{width}}}$ (**Eqn. A1-8**).

Generalized multivariate RBF	
$distFn = \left[norm\ x-c_j\ .\wedge\ 0.5*power\right]$ $xcwFn = distFn^T * \left[DM\ \ a_k .\wedge power\right] * distFn$ $RBF\_Gen = \exp(-xcwFn)$	**Eqn. A1-11**
$term1 = \dfrac{1}{2*\Pi^{power/2} * \left[\det\ \operatorname{cov} X\right]^{power/2}}$	**Eqn. A1-12**
$RBF\_Gen =$ $\exp\ -\left[norm\ \ x-c_j\ .\wedge\ 0.5*power\right]^T *$ $\left[DM\ \ a_k .\wedge power\right] *$ $\left[norm\ \ x-c_j\ .\wedge\ 0.5*power\right]$	**Eqn. A1-13**

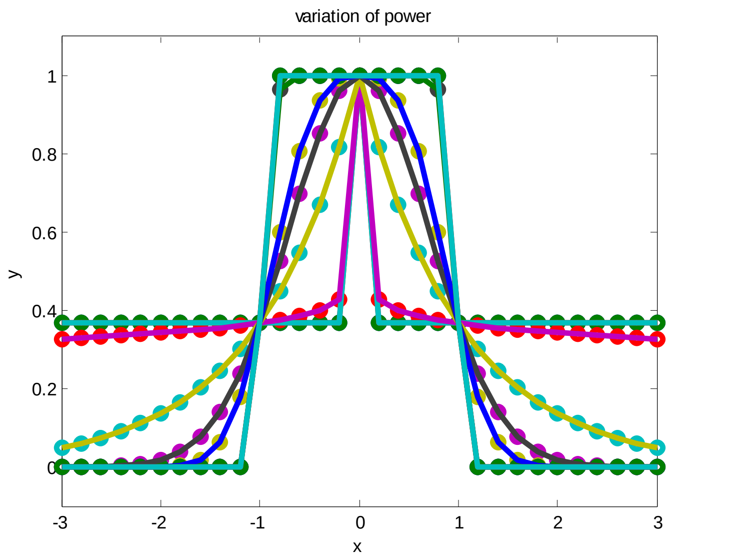
RBF in two dimensions



▶ RBF_Generalized

Instead of a square of the standardized input $z_i = \left[\dfrac{x_i - c_k}{sd_k}\right]$, a variable power is introduced (Eqn.A1-13, Chart A1-5(b)). It results in more non-linear shape compared to Euclidean distance measure. Further, introduction of variable dispersion (variance) in different dimensions of the data space results non-linear patches (Chart A1-5). A change of magnitude of center shifts RBF profile to the right or left of zero. If power is two, Hunt's formula results. Further, assuming same variation in all directions, the popular RBF with Euclidean distance is obtained. It is interesting to monitor visual display of surfaces and contours with variation in A matrix (even with non-symmetric off-diagonal elements. The functions discussed (vide supra) are all used as TFs in RBF-NNs and 4-layer RBF-SG-NN. Each software package employs only one and at the most two types of functions. The research mode software contains all functions available at the first instance. Later the best suitable selection procedure will enhance the capability of NNs.

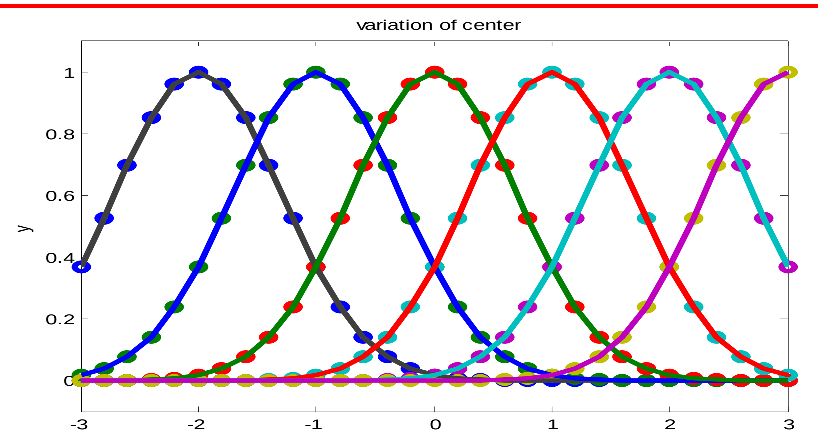
Chart A1-5(b): Variation of power in RBF-generalised



If	$power \neq 2$
Then	patches
If	power =2 widths are different
Then	Hyperellipsoid
If	power =2 same width for all clusters
Then	hyper spheroid

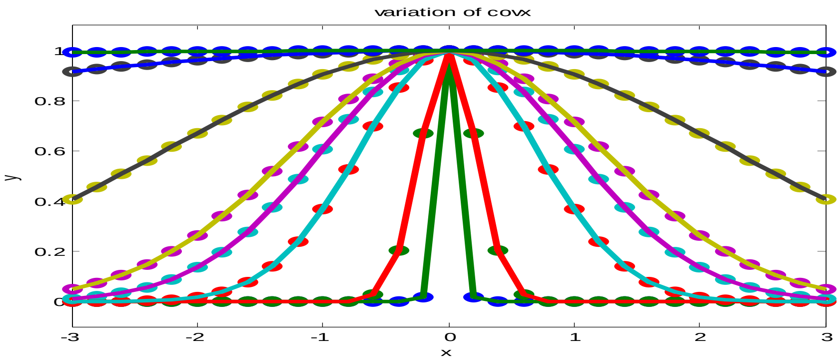
```
%
% variation of power
%
      powerz = [1e-100 0.001,0.1,1,2,3,15,114 1e100]';
      center = 0; covx = [ 1 0 ; 0 1];
%
for i = 1:length(powerz)
          power = powerz(i,1);
          yp(:,i) = om_rbfg1x(xx,center,covx,power,0);
end
```

Variation of center



```
%
% variation of center
%
      centerz = [-2,-1,0,1,2,3]';
      power = 2; covx = [1 0 ; 0 1];
for i = 1:length(centerz)
          center = centerz(i,1);
          yc(:,i) = om_rbfg1x(xx,center,covx,power,0);
end
```

Variation of Covariance




```
%
% variation of cov
%
      center = 0;power = 2; covx = [1 0 ; 0 1];
      varz = [0.01 0.1 1 2 3 10 101 1100 ]';
for i = 1:length(varz)
          covx(1,1)  = varz(i,1);
          sd = varz(i,1);
          ysd(:,i) = om_rbfg1x(xx,center,covx,power,0);
end
```

Shape adaptive RBF

A shape parameter is an empirical constant enhancing the accuracy of model with minimum basis functions. But, unfortunately, it varies from task to task. In RBF literature zero to three shape parameters have been in use (Chart A1-6). There are diverse research findings regarding the efficiency and depends upon the task and mode of solving it. Inverse_multiquadratic_RBF (KB. A1-2)was used for real-time signal-processing in scientific and engineering applications.Webb [83] reported an optimal choice of TFs in RBF_NN. The shape adaptive RBF function is not sensitive to prediction error, but depends upon initial conditions. It results in lower errors compared to standard RBF_NN for a given number of neurons. An optimal scaling method alternating between linear and non-linear methods is used which can be extendable to MLP and other NNs

Billings [124,125] proposed a generalized-multi-scale-RBF_NN (Gen.MultiScale-RBF_NN), which models non-linear

Chart A1-6: Typical TFs and associated shape parameters

#Shape Par.	Name_RBF
0	Thin Plate Splines Cubic Splines
1	Conical Multiquadric Inverse Multiquadric Gaussian Complex Fourier J-Bessel
3	Conical + Spline
4	Fourier_RBF

dynamic systems. The k-means clustering algorithm is used to find out centers of the clusters and each of selected centers has multiple-kernel-widths. Since the TF has multiple scale parameters, it is referred as Gen.MultiScale-RBF_NN. In this study, orthogonal least squares (Ortho_LS) is used to refine WROs.

KB. A1-2: Comparative tail behavior of RBFs		
Longer tails than Standard RBF [i.e., their activations for patterns distant from centroid of RBF are bigger than activation of SRBF for those patterns] For sufficiently large distance norms, the decay very slow.	**If**	Cauchy Inverse multiquadratic
Do not fall asymptotically to zero.	**If**	Cauchy, inverse multiquadratic, Standard rbf

Generalized-RBF_Navarro

The introduction of a shape parameter to standard Gaussian distribution results in generalized Gaussian distribution (Gen.Gauss.Dist) (Eqn.A1-18). Fernández-Navarro [224]ignored the probability constraint and independently introduced generalized radial basis function (RBF_Navarro) for input space of xdim (chart A1- 7) with a shape parameter used as power.

Eqns. A1-(B): Generalized Gaussian distribution

tau; z; normal_standard_deviation | **x; center; shape : shape_index**

$$width_j = \sigma_j * \sqrt{\frac{gama\left(\frac{1}{tau_j}\right)}{gama\left(\frac{3}{tau_j}\right)}}$$

Eqn. A1-14

$$GGFn = \exp\left(-\frac{norm\left(x - center_j\right)^{tau_j}}{width_j^{\,tau_j}}\right)$$

Eqn. A1-15

$$gama = \int_0^{\infty} tau^{z-1} * \exp(z) * dz$$

Eqn. A1-16

$$const = \left[\frac{shape_j}{2 * width_j * Gama * \left(\frac{1}{tau_j}\right)}\right]$$

Eqn. A1-17

$$pdf\_GGD(j) = const * GGFn$$ **Eqn. A1-18**

Chart A1-7RBF_Navarro

$$RBF\_Navarro_j = \exp\left(-\frac{norm\left(x - center_j\right)^{power_j}}{width_j^{\,power_j}}\right)$$

Eqn. A1-19

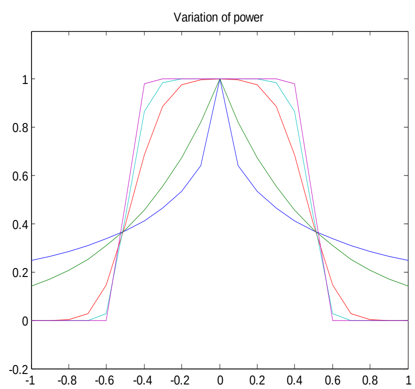
power : relaxes or contracts Fn profile

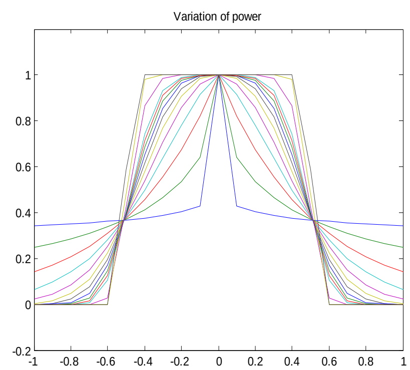
- It reproduces continuously and smoothly different RBFs by varying magnitude of floating point free parameter

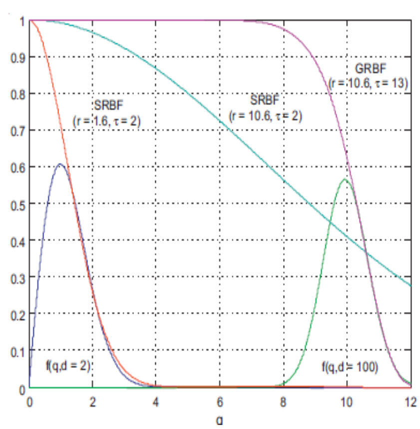
If	power → 0
Then	Close to uniform distribution
If	power = 1
Then	Laplacian distribution
If	power = 2
Then	Gaussian distribution
If	Power > high

- Approximates constant functions at specific intervals
- Appropriate kernel in quantifying similarity even in high dimensional spaces(>90)

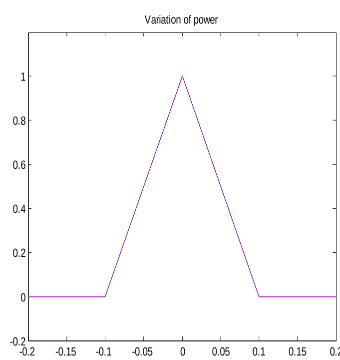
Then Approximately rectangular







PDF at different distances (courtesy of Ref 224)



sd = .000013

```
 %
%  rbfNavarro   19/07/14 (0:58 hrs)
%
function [yrbf,yrbf_norm] = om_rbfNavarro(xx,center,width,tow,fig)

% default input
if nargin <5
          clean, fig = 1;
          [xx]= [-3:0.1:3]';
          [r,c]=size(xx);
           sd = .51
          center = [0];
%tow = [1/10, 1/2,1,3/2,2,5/2,3,7/2,4,9/2,5,8,16,32 ]';
          tow = [  1/2,1,  4, 8,16]';
end
%
```

```
%
%%
     % F. Fernández-Navarroa, C. Hervás-Martíneza,
% R. Ruizc, J. C. Riquelmeb,
%           Appl. Soft Comp., 2012,  12,  1787-1800
%

for jj = 1:size(tow)
    power = tow(jj,1)
    [r,c] = size(xx);
%
for i = 1:r
x(:,1) = xx(i,:);
       d = norm(x-center)
       yrbf(i,jj) = exp(-d.^power/sd.^power)
end
end


% Visual output
if fig
       xlim = 2;
       xlim = 1; plot (xx,yrbf), axis([-xlim,xlim, -0.2 1.2])
      title(['   Variation of power    '])
end
```

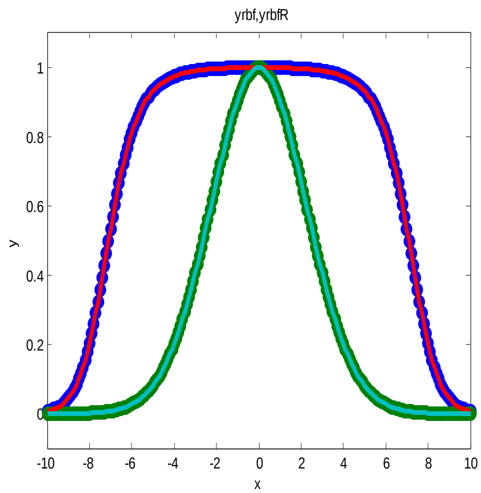
RBF with top flattening

An equation with wide scope of sharp to flattening and variable smoothness of the hill (max(y_RBF))is realizable with Eqn.A1.20, a modified form of RBF (KB. A1-3). A few typical profiles simulated are in chart A1-8).

Chart A1-8: TopFlat_RBF

$\text{yRBF\_topFlat} = \left[\dfrac{1+R}{R+\exp\left(-s*(x-c)^T*A*(x-c)\right)} \right]$ Eqn.A1.20	A : Normalization matrix or Eccentricity of Kernel; Positive definite matrix R : Flattering parameter around center TOP flatness S : Slope of Kernel or Steepness of smoothness hill factor

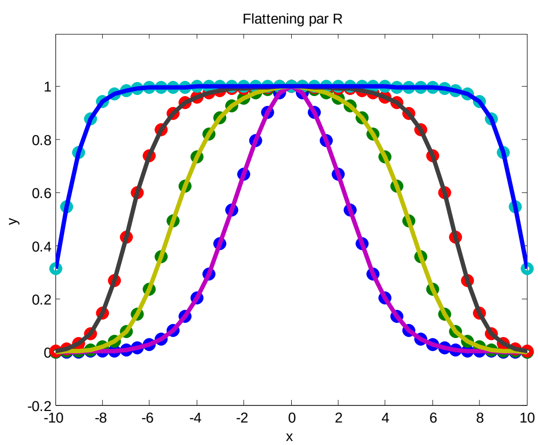
$$\text{yRBF\_topFlat} = NLf\left[\begin{array}{c} dist(x, center\ , EccentMat, \\ R, Slope \end{array}\right]$$




```
%
%  om_rbfR.m   31/08/10 (12:00 hrs)
%
function [yrbfR] =
om_rbfR(x,center,steepness,A,R,fig)
if nargin < 5, clean,fig = 1;inc= 0.1;
    [x,one] =  xgrid(-10,10,inc) ;
    steepness = 1; A = [1 0 ;0 1];center
= 0;A=.1;R = 152;
end
       [r,c] = size(x);
for i = 1:r
     z = x(i)-center;
     f(i,1)  = exp(steepness * z'*A*z);
     yrbf(i,1) = 1./f(i,1);
     yrbfR(i,1) = (1+R)/(R + f(i,1));
end

if nargin <5 |fig==1
      plot2d(x ,yrbfR),
      title(['   yrbf,yrbfR '])
end
```

Variation of Flattening parameter (R)



KB. A1-3: Effect of lamda on smoothing

If A = I(dentity) * lamda
&
landa is small
Then weak smoothing

If landa is large
Then significant smoothing

```
%
%  dem_rbfR.m   31/08/10 (12:00 hrs)
%

  clean,inc= 0.5;
    [x,one] =  xgrid(-10,10,inc) ;
    [r,c] = size(x);
%
%  variation of R
%
    steepness = 1; A = [1 0 ;0 1];center = 0;
    A=.1;
    Rz = [0, 10,100,1e4,1e10]';
    [rR,cR] = size(Rz);

for i = 1:rR
     R = Rz(i,1);
```

```
     yrbfR(i,:) = om_rbfR(x,center,steepness,A,R,0);
y(i,:) = [yrbfR(i,:)];
end
%
   plot2d(x ,[[y(1,:)]',[y(2,:)]',[y(3,:)]',[y(4,:)]' ]),axis([-10,10,-0.2,1.2])
   title(' Flattening par R')
```

The contents ofKB.A1-4 represent one way of climbing down a ladder of a more general function from top to base level i.e. simpler forms of function for limiting values of R, A and s. The other part of trivial exercise, but crucial in auto-discovery systems, is to develop generalized form of a simpler model up the ladder by introducing one by one parameter like climbing up step by step.

KB. A1-4: Simpler Fns from TopFlat-RBF

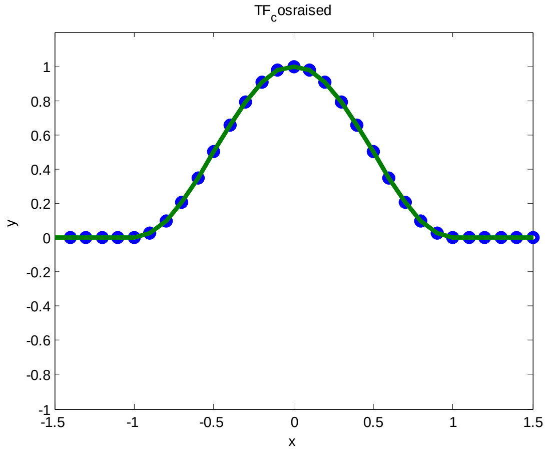
	Antecedent	Consequent yRBF=
If	R>0	$\left[\dfrac{1+R}{R+\exp\left(-s*(x-c)^T*A*(x-c)\right)}\right]$
If	R= 0	$\exp\left(-s*(x-c)^T*A*(x-c)\right)$
If	R = 0 & s= 1	$\exp\left(-(x-c)^T*A*(x-c)\right)$
If	R = 0 & s= 1 & A = I	$\exp\left(-(x-c)^T*(x-c)\right)$
If	R = 0 & s= 1 & A = I & c = 0	$\exp\left(-x^T*x\right)$ i.e. $=\exp\left(-x^2\right)$
If	s= 0	$\left[\dfrac{1+R}{R}\right]$: constant
If	s= 0 & R = 1	2

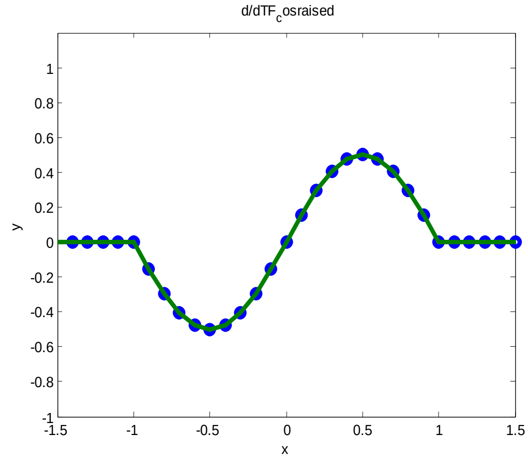
▶ **Raised cosine function** (RBF/22)

If a target function varies in some regions and relatively flat in other regions, raised cosine [195] TF is more appropriate to simple RBF. It is smooth, yet compact. The output of this TF and its first derivative are continuous functions (**Chart A1-9**). The results of NNs with raised cosine TFs coincide with exact solution at higher grid density. Although not exact, it is analogous to piece wise linear regression and piece wise quadratic interpolation.

.

Chart A1-9: Raised cosine_Fn

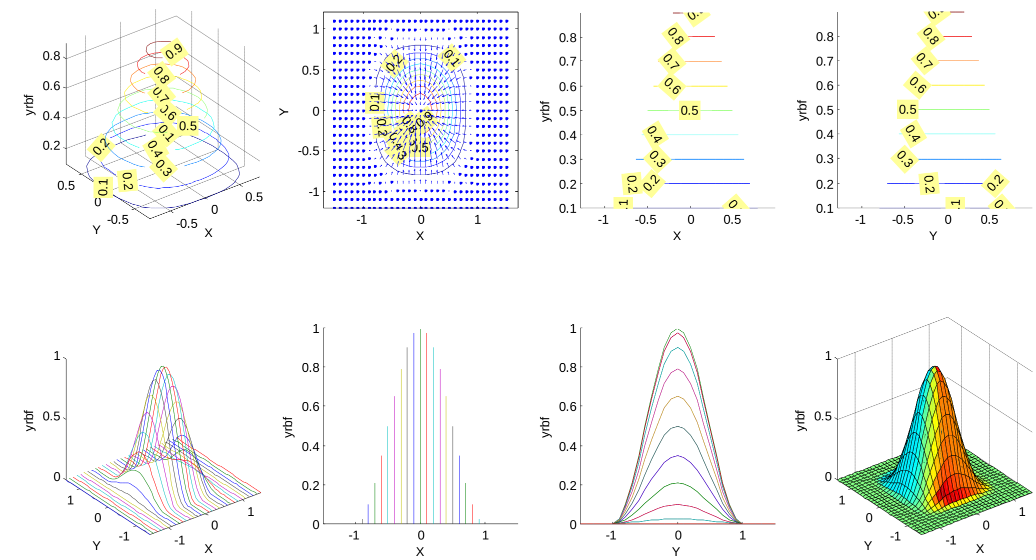





```
%
%   om_cos1raised1x.m   (R S Rao 30/9/03)
%
function [tfcosr,tfdcosr] = om_cosraised1x(x,fig)
%
if nargin <2, fig = 1; end
if nargin <1,
   clean,[x] = [-2:0.1:2]';
   fig = 1;
end
%
[r,c]= size(x);one = ones(r,1);

tfcos  = [one + cos(pi*x)]/2;
tfdcos = [      sin(pi*x)]/2;
%
for i = 1:r
if abs(x(i,1)) > 1
      tfcosr(i,1) = 0;
      tfdcosr(i,1)= 0;
%
else
      tfcosr(i,1)  = tfcos(i,1);
      tfdcosr(i,1) = tfdcos(i,1);
end
%
end
if fig, plot2d(x,tfcosr),axis([-1.5,1.5,-1.,1.2]),
    title('TF_cosraised')
    plot2d(x,tfdcosr),axis([-1.5,1.5,-1.,1.2]),
    title(' d/dTF_cosraised')
end
```

Raised cosine_2D_Fn



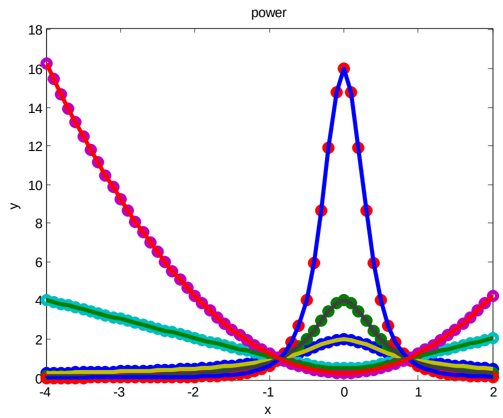
```
%
%  om_cosraised2x.m     (R S Rao 30/9/03)
%
function [z] = om_cosraised2x(xx,yy,one,fig)
%
if nargin <2,
   clean,[xx,yy,one] = xygrid(-1.5,1.5,0.1,-1.5,1.5,0.1);
   [r,c]=size(xx);fig = 1;
end
%
if nargin <4,fig =0;end
%
%
[r,c]=size(xx);one = ones(r,c);
      f1 = [one + cos(pi*xx)]/2;
      f2 = [one + cos(pi*yy)]/2;
      z = f1.*f2;
%
for i =  1:r
for j = 1:c
if  abs(xx(i,j))>1.|abs(yy(i,j))>1.
z(i,j) = 0.;
end%if
end%for j
end% for i

if nargin<3|fig ==1
    xyz_plot(z,xx,yy,'yrbf')
end
```

Product functions

The exponential function is a popular for a dumb bell way of fading out of response with distance from center.The other forms with kernel characteristics are piecewise linear functions, product functions viz. quadratic inverse, multi-quadratic, cubic, logarithmic,Lowe, thin plate spline, AUPARfunctions (Chart A1-10). The product of two basis functions may become local even though the component functions are non-local. Lowe suggesteda product function (non-positive and non-local) instead of a simple Gaussian. The graphical display of the function reveals a large scope of non-linear_basis_ patches to model multi-dimensional functions and discriminating hyper surfaces in classification.

Chart A1-10 : Typical Basis Functions used in RBF-NNs from general Nonlinear functions $y = \left(z^2 + width^2\right)^{power}$ with width> 0 and $x \in R$



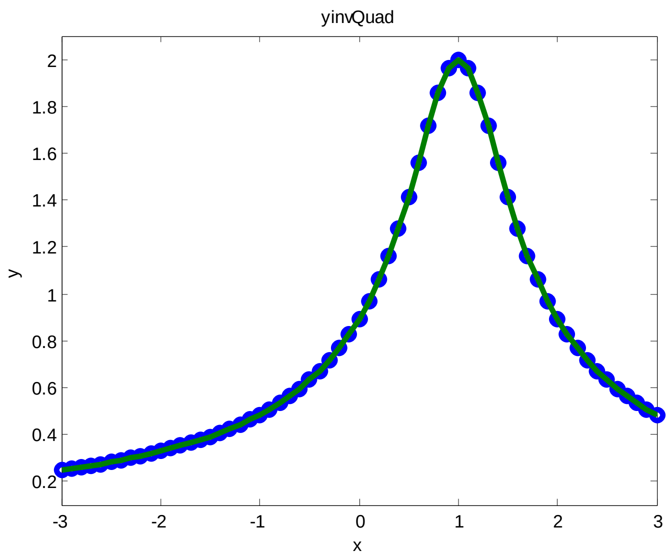
$y\_Quad = Fn\left(\text{distFn, x, center}\right), \text{width, power}$

```
%
%   power
%
powerz = [-0.5,-1,-2,0.5,1,2]';
%
for i = 1:length(powerz)
    power = powerz(i,1);
    y2 = (dev.^2 + (width*width)*one).^power;
    ykernel(:,i) = y2;y(:,i) = y2;
end
%
plot2d(dev,y(:,1),y(:,2),y(:,3),y(:,4),y(:,5))
title('power')
plot2d(dev,[y(:,1),y(:,2),y(:,3),y(:,4),y(:,5)])
title('power')
```

$$y\_Quad = \left(z^2 + width^2\right)^{power}$$

Eqn. A1-21

$y\_Quad =$	If p is
$\dfrac{1}{\left(z^2 + width^2\right)^2}$	-2
$\dfrac{1}{z^2 + width^2}$	-1
z^2	1 & If width is 0 or 1e-3

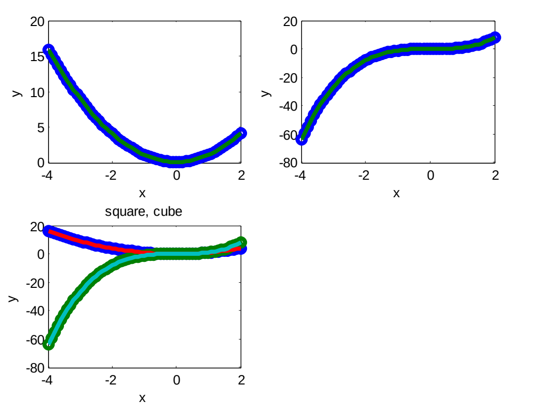


$$y\_Quad = \left(dev^2 + width^2\right)^{power}$$ **Eqn. A1-21**

$$\frac{1}{\sqrt{dev^2 + width^2}}$$ **Eqn. A1-22**	Inverse quadratic, if p = -0.5

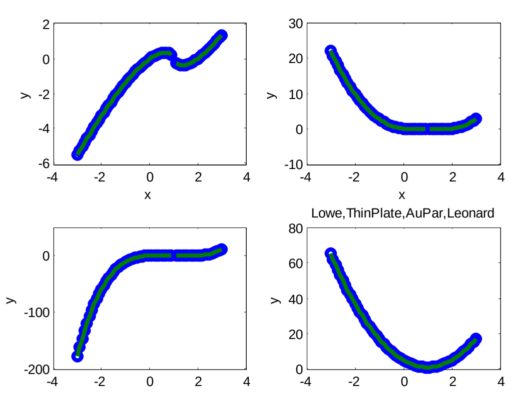
```
%
%   Inverse quadratic
%
yinvQuad = (dev.^2 +
(width*width)*one).^(-0.5);

plot2d(x,yinvQuad )
title('yinvQuad')
```



Quadratic	$z^2 + width * z + threshold$ **Eqn. A1-23**
Cubic	z^3 **Eqn. A1-24**

```
%
%   square, cubic
ySquare = dev.^2;
yCubic = dev.^3;
%
plot2d(dev,ySquare,yCubic,[ySquare,yCubic])
title('square, cube')
```

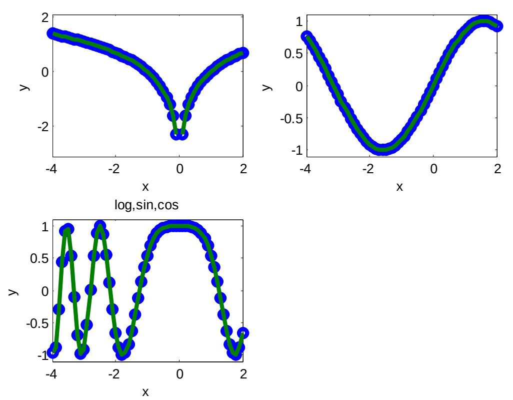


(c) Product functions	
	$z^{power1} * \log\ z^{power2}$ **Eqn. A1-25**
Power2 = 1	
Lowe	$z * \log\ z$
Thinplate	$z^2 * \log\ z$
AuPar	$z^3 * \log\ z$

```
%
%
```

```
%   Lowe, ThinPlate,  Leonard AuPar
%
dev =  (x-center);
yLowe = dev.*log(dev);
yThinPlate = (dev.^2).*log(dev);
yAuPar = (dev.^3).*log(dev.^2);
yLeonard = ((dev.^2) + (width.^2*one))./width.^2;

plot2d(x,yLowe,yThinPlate,yAuPar,yLeonard)
title('Lowe,ThinPlate,AuPar,Leonard')
```



Logarithmic and trigonometric Fns		
	$y =$	
Logarithmic	$\log \ z^{power}$	**Eqn. A1-26**
Trigonometric	$\sin \ z^{power}$	**Eqn. A1-27**
	$\cos \ z^{power}$	**Eqn. A1-28**
Exponential	exp(-z)	**Eqn. A1-29**

If z=0
Then $\log \ z^{power}$ undefined

If z= 1
Then $\log \ z^{power}$ is zero

```
%
%   sine, log
%
p = 1; ylogdev1 = log(dev.^p);
p = 1; ysindev1 =  sin(dev.^p);
p = 2; ysindev2 =  cos(dev.^p);

plot2d(dev,ylogdev1,ysindev1,ysindev2)
title('log,sin,cos')
```

- Radial cubic B-spline:

The performance is similar to standard_RBF, as both these functions have similar convergence properties.

If Kernel is singular
Then Smoothing trick resolves singularity

- **Complex Fourier RBF**

There has a continuous debate to seek an unequivocal crisp answer to the query ‘which type of function is the most suitable as RBF?’ Till date, scientists respect the verdict that there is no universal method (like general problem solver contemplated by AI community in mid nineteenth century) for all tasks although more adequate, robust and high impact algorithms are evolving. Javaran et al. [24] proposed real Fourier form as RBF and applied to 2D transient analysis of elasto-dynamics. But, the progress in complex number analysis, quaternion algebra found a niche in this decade in applied sciences and engineering. The mapping of real space tasks into complex space promised reduction of algebraic manipulations for

Chart A1-11: Complex_Fourier_RBF
+ More accurate solutions
+ Less number of degree of freedom
+ Processes both Gaussian & real Fourier RBF simultaneously
→ More robustness and potency

solving system of equations. Respecting this thesis, Javaran et al. [24] recently used complex form of Fourier series in functional space in place of real Fourier form in choosing RBF(chart A1-11).

q-Gaussian-RBF

The principle of maximum Boltzmann–Gibbs–Shannon (BGS) entropy with constraints produces Gaussian, exponential and Laplace distributions. The q-logarithm and q-exponential functions formed the mathematical basis of Tsallis statistics. The chart. A1-12contains q-exponential function reducing to exp(x) in the limit as q→1. By replacing exponential function by q-exponential function and maximizing entropy under constraints results in q-Gaussian distribution. The pdf is in the range –inf to 3. Fernández-Navarro et al. [170,224] proposed q-Gaussian as a transfer function in RBF_NN and applied to critical classification tasks with better results. The change in numerical magnitude of a real valued q-parameter of q-RBF alters the shapes of profiles with same architecture (i.e. number of neurons and connections) of q-RBF_NN.
(KB.A1-5).

Chart A1.12a: q-Gaussian-RBF

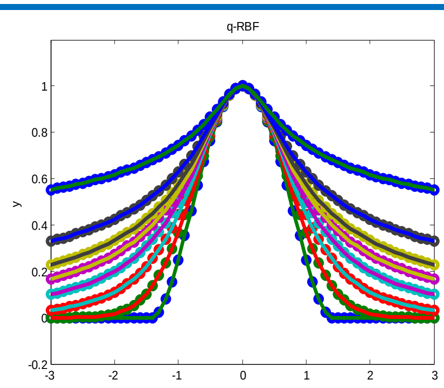
$$q\,\mathrm{exp}x = 1+ 1-q \;*x^{\frac{1}{1-q}}$$ **Eqn. A1-30**

$$q\,\mathrm{exp}x = \frac{1}{1- \;q-1 \;*x^{\frac{1}{q-1}}}$$

$$\mathrm{Bq} = \left[\;3-q \;*\sigma_q^2\right]^{-1}$$
$$q \in (-\infty, 3)$$ **Eqn. A1-31**

$$\mathrm{qGaussRBF(i)} = qwxp(j)^{-dev(j)}$$ **Eqn. A1-32**

$$\mathrm{qGaussRBF(i)} = \begin{cases} 1- \;1-q \;*dev \;\; x^{\frac{1}{1-q}} \\ \quad if \;\; 1- \;1-q \;*dev \;\; x \;\; \geq 0 \\ 0 \;\; otherwise \end{cases}$$ **Eqn. A1-33**




```
%
%  om_qrbf.m   17/07/14 (23:55
hrs)
%
function [yrbf,yrbf_norm] =
om_qrbf2(xx,center,covx,power,fig
)

if nargin <5
        clean, fig = 1;
        [xx]= [-3:0.1:3]';
        [r,c]=size(xx);
        power = 0.2;power = 2
        center = [0];
        covx = [1];sd = 1;
```

KB. A1-5:Q-Gaussian–RBF

If	q is close to 2.0
Then	q-Gaussian is Cauchy_ RBF
If	q=3
Then	inverse multi quadratic _ RBF
If	q → 1.0
Then	q-Gaussian converges to standard RBF
If	$q \neq 1$
Then	Indicatesdeparture from Gaussian statistics
If	$q \geq 1$
Then	Eqn. is not normalizable
If	q <5/3
Then	usual variance (second-order moment) is finite
If	−∞ <q <3
Then	q-variance remains finite for the full range equal to unity for the standard q-Gaussian distribution
If	5/3 ≤ q <3
Then	q-Gaussian distribution diverges

```
end
```

```
       [r,c] = size(xx);

       q = 1-eps;q= 2.;
       qz = [0.5,0.99,1.5,2.0,2.5, 3,4,8]'
for jj = 1: length(qz)
     q = qz(jj,1);
for i = 1:r
x(:,1) = xx(i,:);

       dev = [norm(x-center)/sd].^2;
if [1 - (1-q) * dev] >= 0
       yqrbf (i,1) = [1 - (1-q) * dev].^(1/(1-q));
else
           yqrbf (i,1) = 0;
end
end
       yq(:,jj) = [yqrbf(:,1)];
end

plot2d(xx,[yq(:,1),yq(:,2),yq(:,3),yq(:,4),yq(:,5),yq(:,6),yq(:,7),yq(:,8) ])     axis([-3,3,-0.2,1.2])
title(' q-RBF')
```

J-Bessel _RBF

The RBFs used earlier in NN literature as TFs were either local (compact supported) or global (canonical). None of them have oscillatory component and thus not ideal for dynamic systems. Javaran et al. [24] proposed J-Bessel_ RBF which has an oscillatory behavior (KB. A1.6). It is applied to rectangular domain in plane strain condition, Cantilever plate excitation and Perforated strip subjected to tension stress tasks. The results of J-Bessel_RBF_NN with analytical solutions and with other RBF_NNS for numerical tasks are more accurate with smaller degrees of freedom.

KB.A1-6: J-Bessel (JB) Fns as class of oscillatory RBFs

$$JBesselFn = \frac{J_{\frac{d}{2}-1}(\varepsilon * r)}{(\varepsilon * r)^{\frac{d}{2}-1}}$$

Eqn. A1-34

$\varepsilon > 0$
d: [1,2,…]
$r^m = \left| x - y^m \right|$: distance between x and y^m

If D = 5

Then

$$JBesselFn5\ \ r = \frac{J_{1.5}(\varepsilon * r)}{(\varepsilon * r)^{1.5}}$$

Eqn. A1-35

General set of JB RBFs

$$JBesselFnm\ \ r = \frac{J_{1.5}(\varepsilon * r^m)}{(\varepsilon * r^m)^{1.5}}$$

Eqn. A1-36

S. Hamzeh Javaran, N. Khaji, A. Noorzad, ⇨ *Acta Mech*., **2011**,218, 247–258.

Appendix A2: Gradient of RBF with respect to center, width and exponent

A function of error (residual) sum of squares (ESS)is coveted single object function in arriving at best RBF_NN. Later Akaike information [124,125] has been used with success.Lacerdaetal.[93] put forward a multiple object function criterion on training as well as validation set using GA to choose centers and spread of the clusters. Billings [125] minimized Akaike information in training and test data sets (with multiple optimization criteria) and found the centers in RBF training.The first and second derivatives of RBF with its parameters throw light on trends, breaks and flat regions of error surface of optimization function. This information is instrumental in not only the choice of solution methods but also on improvements in the function with desired characteristics.

If distance measure is Euclidian norm, $RBF = \exp\left\{-\frac{\left\|x_i - c_j\right\|^2}{2*\sigma_j^{\,2}}\right\} = \exp(-z)$, and derivatives are

$$\frac{\partial RBF_i}{\partial par_j} = \frac{\partial \exp(-z)}{\partial z} * \frac{\partial z}{\partial par_j}, \text{ where } par_j = \left[c_j ; \sigma_j\right]$$ (Table A2-1).

Table A2-1-: Derivatives of RBF with center and sigma

RBF	$\frac{\partial RBF_i}{\partial c(entre)}$	$\frac{\partial^2 (RBF_i)}{\partial \sigma^2}$	
$\exp\left\{-\frac{\left\|x_i - c_j\right\|^2}{2*\sigma_j^{\,2}}\right\}$ **Eqn. A2-1**	$2*(x_i - c_j)*\exp\left\{-\frac{\left\|x_i - c_j\right\|^2}{2*\sigma_j^{\,2}}\right\}$ **Eqn. A2-2**	$\frac{\partial \exp(-zz)}{\partial zz} * \frac{\left\|x_i - c_j\right\|^2}{2} * \frac{1}{\sigma_j^{\,4}}$ **Eqn. A2-3**	

Appendix A3: Numerical methods for assignment of center of RBF (KB. 3-1)

k-Nearest neighbor deviation (TRAJAN)

In k-nearest neighbors' method, the mean distance of the centers to its k-NN is calculated. It is used as the deviation for k^{th} radial distance. If the deviations are smaller (or in a dense cluster) the details are preserved. The fine details are obtained by interpolation when the deviations are in a sparse area. Bishop [123] used RMS distance from k-closest to each unit as the standard deviation. Here, the coincident points are not considered. In the algorithm each unit has a different deviation based on the density of points closer to it. The algorithm is fine-tuned by multiplying the RMS value with the coefficient given in the deviation field. If the non-coincident neighbors are less than that the user specified value, the algorithm uses the available neighbors.

Isotropic deviation (TRAJAN)

Haykin [13] proposed a heuristic to calculate the deviation of the radial PE based on the spread of the centers of the clusters σ^2 = (#centers/$dist^2$) where dist is the distance between the most distant centers. The deviation is selected heuristically based on number of centers and volume of space (spread of the training data).

KB. 3-1: KB for centers of RBF_NN

If NP is small
Then Co-ordinates of points can be taken as centers of kernels

If NP is large
Then Unsupervised clustering method to select a prefixed number of centers

Explicit method

Using the user chosen value, the deviation is calculated. It is related to the standard deviation as dev= [1/(2* σ^2)]. There is a provision of assigning different deviations for different RBL neurons in TRAJAN software. The deviation here is not the standard deviation of the Gaussian but equal to the product of the distance of the weight vector from the input vector and the user chosen value. Thesethree methods are implemented in TRAJAN, a commercial software package for NNs.

Appendix A4: Radial basis function in SVM classification/regression

Chart A4.1: Vinegar quality control with NIR and SVM (RBF kernel)

- Step 0:95 vinegar samples were collected from 11 provinces in China, belonging to 14 different origins

In recent times, radial basis function in various forms has been a competing kernel function in SVM to model multi-class category tasks. The standard SVM was proposed for a two class problem initially. Later, it was extended to multiple classes and number of publications has exponentially grown over time.

- Step 1: Five classes are mature-/ aromatic-/ rice-/ fruit-/ white vinegar
- Step 2: One PC of 2D-PCA of NIR separated Fruit- and white vinegar
- Step 3: The first two PCs with maximum score values are selected as input to LS-SVM
- Step 4: LS-SVM identifies mature-/ aromatic-/ rice vinegar

Classification with NIR spectra:
Ji-yong et al. [229] proposed a chemometric tool for rapid classificationof different types of vinegar (chart A4-1). The outcome of the cited procedure [LS_SVM(RBF)] is superior to MLP_NN with BP algorithm for quantitative measurement of total acid content. This promotes that NIR with LS_SVM (with RBF kernel) as an adaptable quality monitoring/control tool for vinegar.

Structure_sensory response_relationships (struct.SensResp.Rel):Dong et al. [64] reported the results ofstructure_sensory response_relationships (Struct. SensResp. Rel) with PLS, GA_MLP(BP), SVM(RBF) of higher alcohol and ester(flavor compounds) and sensory response to ensure reproducible quality of beer.

Table A4-1: Quality of Beer with chemometric techniques

Method	Kernel Fn.	Pred %
GA-BP		52.1
PLS		31.7
SVM	polynomial	32.9
SVM	**RBF**	**96.2**

EEG for seizure/nonseizure discrimination:Generally, epileptic seizures are unpredictable and irregular. Thus, automatic EEG analysis for seizures is pivotal in detecting epileptic episodes. Fu et al. [35]classified seizures and nonseizures from EEG signals in the theta band. The receiver operating characteristics (ROC) curve measure shows 99.125% accuracy for theta rhythm of EEG signals.

- Hilbert–Huang transform (HHT) is used to represent EEG as a time-frequency image.
- From the frequency-bands of rhythms of EEG signals, TFI is segmented.
- The mean, variance, skewness and kurtosis of pixel intensity of histogram of grayscale sub-images are calculated.
- Radial basis based SVM classifies seizures.

SVM_RBFs have also been applied to surface EMG signalsfor fundamental frequency (F0) voicing state (VS), estimation of PDF in two class discrimination[145], parameters of Weibull distribution, 3D- planar redundant manipulator with six-DOF [160] and detection of black spots on apple leaves [116].

REFERENCES

[1] K RamaKrishna, V. Anantha Ramam, R. Sambasiva Rao, *J. Applicable Chem.*,**2014**, 3 (4), 1337-1422; (b**) 2014,** 3 (3):919-989; (c) **2014**, 3 (2), 449-492;(d) **2014**, 3 (2), 834-884; (e) **2014**, 3 (1), 006-029 ; (f) **2013**, 2 (6), 1413-1458 ; (g) **2013**, 2 (5), 1007-1034 ; (h) **2013**, 2 (4): 698-713 ; (i) **2012**,1, 109-124. (j) *Fisheries Res*., **2008**, 92,115-139; (k) Ch.V. Kameswara Rao, M. Phil. (Thesis), 2012, Achrya Nagarjuna University, Visakhapatnam.

[2] R. Sambasiva Rao and G. Nageswara Rao, Computer applications in Chemistry, *Himalaya Publisher*, New Delhi (India), **2005**.

[3] A. Braibanti, R. Sambasiva Rao, G. Priyabrinda and G. Nageswara Rao, *Ann. Chim.*,**1999**, 89, 193-209.

[4] G Priyabrinda, G. Nageswara Rao and R. Sambasiva Rao, *J T R Chem.*,**1997**, 4(2), 29-42.

[5] V. Ananta Ramam, D. Subrahmanyam, S.V.V. Satyanarayna Rao and R. Sambasiva Rao, *J. Teaching and Research in Chem.*, **1997**, 4, 20-28.

[6] A. Braibanti, R. Sambasiva Rao, A. Ravindra Babu and G. Nageswara Rao, *Ann. Chim. Rome*, **1995**, 85, 17-29.

[7] G. Nageswara Rao and R. Sambasiva Rao, *J. Teaching & Research in Chem.*,**1995**, 2, 15-28.

[8] G. Nageswara Rao, S.V.V. Satyanarayana and R. Sambasiva Rao, *Ind. J. Chem. Sci.*,**1990**, 4, 73-112.

[9] S.V.V.Satyanarayana Rao, A. Ravindra babu, G.Nageswara Rao, A. Satynarayana and R. Sambasiva Rao, *Ind. J. Chem.*,**1989**, 28A, 292-297.

[10] G. Nageswara Rao, S.V.V. Satyanarayna Rao, A.Ravindra Babu and R. Sambasiva Rao, *J. Ind. Council of Chemists.*,**1998**, 4, 77-92.

[11] G.Nageswara Rao, S. V. V. Satyanarayna Rao, A.Ravindra Babu and R. Sambasiva Rao, *J. Chem. Science*., **1988**, 4, 77-92.

[12] S.V.V. Satyanarayana Rao, G. Nageswara Rao, A. Ravindra Babu and R. Sambasiva Rao, *J. Chem. Science*., **1987**, 13, 1-31.

[13] P.V. Yee, S. Haykin, Regularized radial basis function networks, Theory and applications, **2001**, John Wiley & sons, Inc., New York.

[14] C.T. Leondes (Editor), Neural network systems, Techniques and applications, **1998**, Academic Press, San Diego, USA. (a) 1. Algorithms and architecutres; 2. Optimisation techniques; 3. Implementation techniques, 4. Industrila and manufacturing systems; 5. Image processing and pattern recognition 6. Fuzzy logic and expert systems; 7. Control and dynamic systems.

[15] J.A.S. Freeman, M. J. L. Orr, D. Saad, Statistical theories of learning in radial basis function networks, **1998**, Neural network systems, Techniques and applications, Vol 1: Algorithms and architectures, 1-59.

[16] L.M.C.Buydens, W.J. Melsses (editors), *Chemometrics-Exploring and exploiting chemical information*, **1994**, Katholick Univ., Nijmegan, Netherlands.

[17] P. H. Fidencio, R. J. Poppi and J. carlos de rade, *Anal. Chim. Acta*, **2002,** 453, 125-134.

[18] A. Pulido, I. Ruisanchez, F.X. Rius, *Anal. Chim. Acta*.,**1999,** 388, 273-281.

[19] Y.L. Loukas, *Anal. Chim. Acta.*,**2000,** 417, 221-229.

[20] H. Robbins, S. Monro,*Annual Math.Stat.*,**1951,** 22,400-407.

[21] L. Kang, X. Zhao, J. Ma, *Appl. Energy*, **2014,** 121, 15, 20-27.

[22] N. Kumar, V. Panwar, J. Borm, J. Chai, *Appl. Math. Comput.*,**2014,** 231, 15, 320-328.

[23] K. Elsayed, C. Lacor, *Appl. Math. Comput.*,**2014,** 236, 1, 325-344.

[24] S. H. Javaran, N. Khaji, *Appl. Math. Mod.*, **2014,** 38, (14-15), 3641-3651; (b) *Eng. Anal. Bound. Elem*., **2011**, 35, 85–95; (c) *Acta Mech*., **2011**, 218, 247–258.

[25] F.J. Mohammed, D. Ngo-Cong, D.V. Strunin, N. Mai-Duy, T. Tran-Cong, *Appl. Math. Mod.*, **2014,** 38(14-15), 3672-3691.

[26] M. Gan, Yu Cheng, Kai Liu, Gang-lin Zhang, *Appl. Soft Comput.*, **2014** (in press).

[27] M. Cheng, M. Cao, *Appl. Soft Comput.*,**2014,** 22, 178-188.

[28] W. Song, J. Z. Liang, X. L. He, P. Chen, *Appl. Soft Comput.*,**2014,** 21, 210-220.

[29] Fayez F.M. El-Sousy, *Appl. Soft Comput.*,**2014,** 21, 509-532.

[30] Lu Xi, H. Muzhou, M. Ho Lee, J. Li, D. Wei, H. Hai, Y. Wu, *Appl. Soft Comput.*, **2014,** 15, 57-66.

[31] X. Hong, S. Chen, A. Qatawneh, K. Daqrouq, M. Sheikh, Ali Morfeq, *Appl. Soft Comput.*,**2014**,23, 9-18.

[32] M. Y. Cheng, M.T. Cao, D. H. Tran, Automat. *Construct.* **2014**,41, 60-69.
[33] O.A. Bashkirov, E.M. Braverman, I. B. Muchnik, Automat. *Remote Cont.***, 1964,** 25, 629–631.
[34] D. L. Reilly, L. N. Cooper, C Elbaum, *Biol Cybern*, **1982**,45,35–41.
[35] K. Fu, J. Qu, Y. Chai, Y. Dong, *Biomed. Signal Process. Control*, **2014**,13, 15-22.
[36] J. V. Link, re L. G. Lemes, I. Marquetti, M. B. dos Santos Scholz, E. Bona, *Chemomet. Intel. Lab. Syst.*, **2014**,135, 15, 150-156.
[37] L. Jiao, Z. Xue, G. Wang, X. Wang, H. Li, *Chemomet. Intel. Lab. Syst.*, **2014**(in press).
[38] J. J. Xing, Y. F. Liu, Y. Q. Li, H. Gong, Y. P. Zhou, *Chemomet. Intel. Lab. Syst.,* **2014**, (in press).
[39] Atefe Malek-Khatabi, Mohsen Kompany-Zareh, Soeh Gholami, Saeed Bagheri, *Chemomet. Intel. Lab. Syst.*, **2014**,135, 15, 157-165.
[40] B. Walczak, D.L. Massart, *Chemomet. Intell. Lab. Syst.*, **2000**, 50, 179–198.
[41] J. Tetteh, S. Howells, Ed Metcalfe, T. Suzuki, *Chemomet. Intell. Lab. Syst.*, **1998**,41, 17-29.
[42] K. Faber, B. R.Kowalski, *Chemomet. Intell. Lab. Syst.*, **1996**,34, 293-297.
[43] E. P. P. A. Derks, M. S. S. Pastor, L. M. C. Buydens, *Chemomet. Intell. Lab. Syst.*, **1995**,28, 49-60.
[44] C. Li, S. C. Yang, Q. S. Guo, K. Y. Zheng, Y.F. Shi, X. F. Xiao, G. Q. Long, *Chemomet. Intel. Lab. Syst.*, **2014**,136,115-120.
[45] Y. Wang, B. Li, Y. Ni, S. Kokot, *Chemomet. Intel. Lab. Syst.*, **2014**,130, 15, 159-165.
[46] A. Sharifahmadian, R. R. Simons, *Coastal Eng.*,**2014**,83,190-204.
[47] X. Ge, X. Yuan, C. Ao, K. K. Yu, *Comp.& Chem. Eng.*,**2014,** 68,**38-46.**
[48] D. S. Broomhead, D. Lowe, *Complex Syst.*,**1988**,2,321–355.
[49] J. Luo, W. Lu, *Comput. Geosci.*,**2014**,67, 110-116.
[50] A. D. Mehr, E. Kahya, C. Yerdelen, *Comput. Geosci.*,**2014**,70, 63-72.
[51] Jiani Du, Zhitao Liu, Youyi Wang, *Control Eng. Practice*, **2014**,26, 11-19.
[52] H. Joutsijoki, K. Meissner, M. Gabbouj, S. Kiranyaz, J. Raitoharju, J. Arje, S. Karkkainen, V. Tirronen, T. Turpeinen, M. Juhola, *Ecol. Informat.*,**2014**,20,1-12.
[53] J. Yao, J. Yang, L. Wang, Y. Han, J. Pan, AASRI Procedia, **2012**, 1,183 – 188.
[54] K. Seetharaman, S. Sathiamoorthy, *Egyptian Informat. J.*, **2014,** 15, 1, 59-68.
[55] G. K. Ziyatdinova, A. A. Saveliev, G.A. Evtugyn, H.C. Budnikov, *Electrochim. Acta*, **2014**,137, 10, 114-120.
[56] T. C. Ou, C. M. Hong, *Energy*, **2014**,66, 1, 314-323.
[57] T. Tanbay, B. Ozgener, *Eng. Anal. Boundary Elements*, **2014**,46, 30-40.
[58] W. Chai, J. Qiao,*Eng. Appl. Art. Intel.*,**2014,** 28, 1-12
[59] *R.*Sambasiva Rao, (unpublished work).
[60] S. fu Li, X. L. Wang, J. Z. Xiao, Z. J. Yin, *Expert Syst. Appl.*, **2014,** 41, 4, Part 1, 1262-1270.
[61] G. S. Babu, S. Suresh, B. S. Mahan, *Expert Syst. Appl.*, **2014,** 41(2), 478-488.
[62] C. Lin, X. Chen, L. Jian, C. Shi, Xiaoli Jin, Guoping Zhang, *Food Chem.*, **2014**,162, 1, 10-15.
[63] J. J. Dong, Q. L. Li, H. Yin, C. Zhong, J. G. Hao, P. F. Yang, Y. H. Tian, S. R. Jia, *Food Chem.*, **2014**,161, 15, 376-382.
[64] J. J. Dong, Q. L. Li, H. Yin, C. Zhong, J.G. Hao, P. F. Yang, Y. H. Tian, S. R. Jia, *Food Chem.*, **2014,** 161,376-382**.**
[65] C. Riverol, V. Pilipovik, *Food Control*, **2014,** 35, 1, 18-21.
[66] S. K. Oh, W. D. Kim, W. Pedrycz, K. Seo, *Fuzzy Sets Syst.*, **2014**,237, 16, 96-117.
[67] A Sherstinsky, RW Picard, IEEE *Trans Neural Networks*, **1996**,7 (1), 195–200.
[68] Mao, K Z, *IEEE Trans Neural Netw,***2002,** 13(5),1211–1217.

[69] A.R. Webb S. Shannon, *IEEE Trans Neural Netw,***1998,** 9 (6), 1155-1166.
[70] BA Whitehead, *IEEE Trans Neural Netw,* **1996**,7(6), 1525–1528.
[71] BA Whitehead, TD Choate, *IEEE Trans Neural Netw,* **1995,** 7(4), 869–880.
[72] BA Whitehead, TD Choate, *IEEE Trans Neural Netw,* **1993,** 5(1), 15–23.
[73] S. Chen, C.F.N. Cowan, P.M. Grant, IEEE *Trans Neural Netw,* **1991,** 2(2), 302–309.
[74] J. B. Gomm, D. L. Yu, *IEEE Trans Neural Netw.*,**2000**,112.
[75] Y. Tan, J. Wang, J. M. Zurada, *IEEE Trans. Neural Netw.*, **2001**, 12, 124-134.
[76] T. M. Cover, *IEEE Trans. Electron. Comput. EC-14***,** **1965**,326–334.
[77] S. Chen, A. Wolfgang, C. J. Harris, L. Hanzo, *IEEE Trans. Neural Netw.*, **2008,** 19,737-745.
[78] B. S. Lin, B. S. Lin, F. C. Chong, F. Lai, *IEEE Trans. Neural Netw.*,**2007**,18.
[79] R. J. Schilling, J. J. Carroll, Jr., A. F. Al-Ajlouni, *IEEE Trans. Neural Netw.*, **2001,** 12,1-15.
[80] K. Chen, D. Wang, *IEEE Trans. Neural Netw.*,**2001**,12, 1084-1102.
[81] B. Igelnik, M. T. Azar, S. R. LeClair, *IEEE Trans. Neural Netw.*,**2001**,12, 236-248.
[82] M. F. Azeem, M. Hanmlu, N. Ahmad, *IEEE Trans. Neural Netw.*,**2000**,11,1332-1346.
[83] R. Webb, S. Shannon, *IEEE Trans. Neural Netw.*,**1998**, 9, 1155-1165.
[84] L. Yingwei, N. Sundararajan, P. Saratchran, *IEEE Trans. Neural Netw.*, **1998**,9, 308-318.
[85] S. Amari,*IEEE. Trans.Electronic Comput,* **1967**,16,299-307.
[86] J. Kilian, H. T. Siegelmann, *Inform Comput***,** **1996**,128(1), 48–56.
[87] J. Wang, W. Zhang, J. Wang, T. Han, L. Kong, *Information Sci.*, **2014**,273, 20 , 304-318.
[88] M. Wang, C. Wang, X. Liu, *Information Sci.*, **2014,** 279, 20, 874-888.
[89] L. Xu, *Int. J. Neural syst.*, **2001**,11, 43-69.
[90] I. Rojas, J. Gonzalez, A. Canas, A. F. Diaz, F. J. Rojas, M. Rodriguez, *Inter. J neural syst.*, **2000**,10, 353-364.
[91] C. M. Hong, C. H. Chen, *Inter. J. Electrical Power & Energy Syst.*, **2014**,55, 554-561.
[92] H. C. Card, *Inter. J. neural syst.*, **2001**,11, 353-364.
[93] E. Lacerda, A. De Carvalho, T. Ludermir, *Inter. J. neural syst.*, **2001**,11, 43-69.
[94] A. Deihimi, A. Solat, *International J. Electrical Power & Energy Syst.*, **2014**,54, 408-424.
[95] H. R. Ansari, *J. Appl. Geophys.*, **2014** (in press).
[96] A. Kahrizi, H. Hashemi, *J. Appl. Geophys.*, **2014** (in press).
[97] P. Bagheripour, *J. Appl. Geophys.*, **2014**,104, 142-148.
[98] Alexridis, E. C. Alex, *J. Biomed. Informatics,* **2014**, 49, 61-72.
[99] A. A. Jamshidi, C.W. Gear, I.G. Kevrekidis, *J. Comput. Appl. Math.*, **2014**,266, 15, 61-72.
[100] G. B.Wright, B. Fornberg, *J. Comput. phys.*, **2006**, 99-123.
[101] V. S. Kodogiannis, E. Kontogianni, J. N. Lygouras, *J. Food Eng.*, **2014**,142, 118-131.
[102] M. Shoaib, A. Y. Shamseldin, B. W. Melville, *J. Hydrology*, **2014,** 515, 16, 47-58.
[103] R. Dash, B. Majhi, *J. Light Electron Optics,* **2014,** 125, 5, 1634-1640.
[104] Y. Xiong, X. Xiao, X. Yang, D. Yan, C. Zhang, H. Zou, H. Lin, L. Peng, X. Xiao, Y. Yan, *J. Pharmacol. Biomed. Anal.,* **2014,** 91,68-72.
[105] H. G. Han, H. H. Qian, J. F. Qiao, *J. Process Control.*,**2014,** 24, 3,47-59.
[106] C. Wan, P. de B. Harrington, *J.Chem.Comput.Sci.*,**1999,** 39, 1049-1056.
[107] S. Li, J. P. Boyd, *J. Comput. Physics.*,**2014,** 258, 1, 931-947.
[108] X. Mao, L. Sun, G. Hui, L. Xu, *J. Food Drug Analysis.*, **2014,** 22, 2, 230-235.
[109] C. Arteaga, I. Marrero, *J. Math. Analysis Applicat.*,**2014,** 414, 1, 450-460.
[110] H. R. Hosseinkhani, F. Esmaeilzadeh, D. Mowla, *J. Natural Gas Science Eng.*, **2014,** 18, 296-303.

[111] T. Li, J. Hong, J. Zhang, F. Guo, *J. Neuroscience Methods.*,**2014,** 224, 15, 26-38.

[112] B. Vaferi, V. Gitifar, P. Darvishi, D. Mowla, *J. Petroleum Science Eng.*, **2014,** 119, 69-78.

[113] W. Shen, X. Guo, C. Wu, D. Wu, *Knowledge-Based Syst*., **2011**, 24 (3), 378–385.

[114] L. T. Danh, K. K .Ahn, *J. Sound Vibration.*,**2014,** 333, 5, 1245-1268.

[115] A.A. Masrur Ahmed, *King Saud University - Eng. Sci.*, **2014** (in press).

[116] E. Omrani, B. Khoshnevisan, S. Shamshirb, H. Saboohi, N. B. Anuar, M. H. Nizam Md Nasir, *Measurement.*,**2014,** 55, 512-519.

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

[118] S. M. Jayanta Basak, *Neural Comput & App.*, **2001,** 10**,** 244–252.

[119] Y. Li, N. Sundararajan, P. Saratchran, *Neural Comput & Applic.*, **2001,** 10, 172-183.

[120] C. Andrieu, No De Freitas, A. Doucet, *Neural Comput.*,**2001**,13, 2359-2407.

[121] J. E. Moody, C. J. Darken, *Neural Comput.*,**1989,** 1, 2, 281-294.

[122] M. J. L. Orr. , *Neural Comput.*,**1995**,7, 3, 606-623.

[123] C. M. Bishop, *Neural Comput.*,**1991**,3, 4, 579-581.

[124] S. A. Billings, H. L. Wei, M. A. Balikhin., *Neural Netw,* **2007**,20, 1081-1094.

[125] S. A. Billings, G. L. Zheng, *Neural Netw,* **1995**,8, 877-890.

[126] D. F. Specht, *Neural Netw,* **1990**,3, 109-118.

[127] M. Wallace, N. Tsapatsoulis, S. Kollias, *Neural Netw.* **2005**,18212, 117-122.

[128] D. Shi, D. S. Yeung, J. Gao, *Neural Netw.* **2005**,18, 951-957.

[129] H. N. Mhaskar, *Neural Netw.* **2004**,17, 989-1001.

[130] Yi Liao, S. C. Fang, H. L. W. Nuttle, *Neural Netw.* **2003**,16, 1019-1028.

[131] M. Lazaro, I. Santamaria, C. Pantaleon, *Neural Netw.*,**2003**,16, 69–77.

[132] F. Schwenker, H. A. Kestler, G. Palm, *Neural Netw.*,**2001**,14, 439–458.

[133] F. H. Hamker, *Neural Netw.* **2001**, 14, 551–573.

[134] S. Albrecht, J. Busch, M. Kloppenburg, F. Metze, P. Tavan, *Neural Netw.*,**2000,** 13, 1075–1093.

[135] Z. ou Wang, T. Zhu, *Neural Netw.*, **2000**,13, 545–553.

[136] M. Marinaro, S. Scarpetta, *Neural Netw.*,**2000,** 13, 719-729.

[137] R. J. Kuo, P. H. Cohen, *Neural Netw.*,**1999**,12, 355-370.

[138] Q. Zhu, Y. Cai, L. Liu, *Neural Netw.*,**1999**,12, 527-540.

[139] S. A. Billings, X. Hong, *Neural Netw.*,**1998**,11, 479-493.

[140] L. Xu, A. Krzyzak, A. Yuille, *Neural Netw.*,**1994**,7, 609-628.

[141] J. Gajewski, T. Sadowski, *Neural Netw. Finite Element Method Comput. Materials Sci.*, **2014**, 82, 1, 114-117.

[142] G. Carvajal, M. Figueroa, *Neural Netw.*,**2014,** 55, 72-82.

[143] C. C. Holmes, B. K. Mallick, IEEE Trans, *Neural Netw.*,**2000,** 11,27-35.

[144] A. Roy, S. G. R. Mira, *Neural Networks,* **1995,** 8**,** 179–201.

[145] M. Gao, X. Hong, S. Chen, C. J. Harris, E. Khalaf, *NeuroComput.*, **2014,** 138, 22, 248-259.

[146] J. M. Wu, C. C. Wu, C. W. Huang, *NeuroComput.* , **2014,** 136, 20, 56-70.

[147] J. Wu, J. Long, M. Liu, *NeuroComput.*,**2014** (in press).

[148] K. Arbateni, A. Bennia, *NeuroComput.*,**2014** (in press).

[149] V. Ravi, M. Krishna, *NeuroComput.*,**2014**,138, 22, 106-113.

[150] T. D. Le, H. J. Kang, *NeuroComput.*,**2014**,137, 5, 12-23.

[151] H. G. Han, L. D. Wang, J. F. Qiao, *NeuroComput.*,**2014**,128, 27, 128-135.

[152] D. Zhao, Q. Zhu, N. Li, S. Li, *NeuroComput.*,**2014**,124, 26, 149-161.

[153] W. Zeng, C. Wang, *NeuroComput.*,**2014**,125, 11, 172-182.
[154] V. Latorre, D. Y. Gao, *NeuroComput.*,**2014**,134, 25, 189-197.
[155] J. Li, Xi. Guo, Z. Li, W. Chen, *NeuroComput.*,**2014,** 142, 22, 190-200.
[156] A. Horzyk, *NeuroComput.*,**2014**,144, 20, 238-257.
[157] L. R. Pedro, R. H.C. Takahashi, Information Sci., **2014**, 268(1), 202-219.
[158] L. Zhao, W. J. Cai, Z. H. Man, *NeuroComput.*, 128, 27, 242-248.
[159] M. de A. Pereira, C. o D. Júnior, E. G. Carrano, J. A. de Vasconcelos, *NeuroComput.*, **2014**,133(10), 342-357.
[160] S. Yahya, M. Moghavvemi, H. A.F. Mohamed, *NeuroComput.*,**2014**,137, 5, 34-46.
[161] Z. Zhang, T. Wang, X. Liu, *NeuroComput.*,**2014**,131, 5, 368-376.
[162] E. P. Gutierrez, V. M. Rivas, M. G. Arenas, M. J. del Jesus, *NeuroComput.*,**2014**,128, 27, 433-446.
[163] S. A. Etemad, A. Arya, *NeuroComput.*,**2014**,129, 10, 585-595.
[164] J. Qiao, Z. Zhang, Y. Bo, *NeuroComput.*,**2014**,125, 11, 7-16.
[165] M. P. Naeini, B. Moshiri, B. N. Araabi, M. Sadeghi, *NeuroComput.*,**2014**,130, 23, 73-82.
[166] D. Zhao, W. Ni, Q. Zhu, *NeuroComput.*,**2014**,140, 22, 8-18.
[167] Z. Liu, Q. Zhang, Z. Han, G. Chen, *NeuroComput.*,**2014**,125, 11, 95-101.
[168] Gh. A. Montazer, R. Sabzevari, F. Ghorbani, *Neurocomput.*,**2013**,111, 2, 169-176.
[169] X. Nie, J. Cao, S. Fei, *Neurocomput.*,**2013**,119, 7, 281-291.
[170] F. F. Navarro, C. H. Martínez, P. A. Gutiérrez, J. M. P. Barragán, F. L. Granados, *NeuroComput.*,**2012**,75-1, 123-134; Neural netw., 2011, 24, 779–784.
[171] K. Zhang, Y. Li, P. Scarf, A. Ball, *NeuroComput.*,**2011,** 74-17, 2941-2952.
[172] A. Kavousifard, H. Samet, *NeuroComput.***2011**,74, 17, 3420-3427.
[173] M. Gao, X. Hong, S. Chen, C. J. Harris, *NeuroComput.*, **2011**,74-17,3456-3466.
[174] P. Zhou, D. Li, H. Wu, F. Cheng, *NeuroComput.*,**2011**, 74-17, 3628-3637.
[175] A. R. Teixeira, A. M. Tomé, E.W. Lang, *NeuroComput.*,**2011**,74, 5, 820-830.
[176] D. Du, K. Li, M. Fe, *NeuroComput.*,**2010**,73(10-12)**,** 2196-2202.
[177] I. A. Gheyas, L. S. Smith, *NeuroComput.*,**2010**,73(16-18)**,** 3039-3065.
[178] I. B. Ciocoiu, *NeuroComput.*, **2010**,73(1-3)**,** 119-126.
[179] M. S. Baghshah, S. B. Shouraki, *NeuroComput.*,**2009**,73(7-9)**,** 1352-1361.]
[180] J. Tikka, *Neurocomput.*,**2009**,72, 10–12, 2649-2658.
[181] F. Long, S. Fei, *Neurocomput.*,**2008**,71, (7–9), 1741-1747.
[182] V. Fathi, G. A. Montazer, NeuroComput.,**2008**,71(07-09)**,** 1797-1802.
[183] K. Y. Huang, K. J. Chen, J. D. You, An-Ching Tun, *NeuroComput.*,**2008**,71(16-18),3264-3274.
[184] L. Yu, K. K. Lai, S. Wang, *NeuroComput.*,**2008**,71(16-18)**,** 3295-3302.
[185] Qing Zhu, Shumin Fei, Tianping Zhang, Tao L, *NeuroComput.*,**2008**,71(16-18)**,** 3617-3624.
[186] Q. Zhu, S. Fei, T. Zhang, T. Li, *NeuroComput.*, **2008,** 71, 16–18,3617-3624.
[187] Gh.A. Montazer, R. Sabzevari, H. Gh. Khatir, *Neurocomput.*,**2008**,71, 1-3,167-173.
[188] A. S. Zayed, A. Hussain, R. A. Abdullah, *Neurocomput.*,**2006**,69, 1868–1881.
[189] A. M. S. Barreto, H. J. C. Barbosa, N. F. F. Ebecken, *Neurocomput.*,**2006**,69, 2041–2064.
[190] C. Harphama, C.W. Dawson, *Neurocomput.*,**2006**, 69, 2161–2170.
[191] D. Manrique, J. Rios, A. R. Paton, *Neurocomput.*,**2006**, 69, 2268–2283.
[192] O. Falca, T. Langlois, A. Wichert, *Neurocomput.*,**2006**,69, 2356–2359.
[193] H. M. Feng, *Neurocomput.*,**2006**,70, 241–251.
[194] Ji. X. Du, De. S. Huang, G. J. Zhang, Z. Fu. Wang, *Neurocomput.*,**2006**,70, 592–596.

[195] F. Rossi, N. Delannay, B. C. Guez, M. Verleysen, *Neurocomput.,***2005**,64, 183-210.
[196] F. Friedrichs, M. Schmitt, *Neurocomput.*,**2005**,63, 483–498.
[197] C. C. Chuang, J.T. Jeng, P. T. Lin, *Neurocomput.,***2004**,56, 123-139.
[198] M. T. V. Baghmisheh, N. P. G. Baffi, E. B. Martin, *Neurocomput.*, **2004**,62, 39-64.
[199] X. X. Wang, S. Chen, D. J. Brown, *Neurocomput.,***2004**, 62, 441-457.
[200] J. M. Vilaplana, J. L. Pedren o-Molina, J. Lo pez-Coronado, *Neurocomput.*,**2004**,61, 495–501.
[201] J. Paetz, *Neurocomput.*,**2004**, 62, 79-91.
[202] M. Sorel, J. Sima, *Neurocomput.,***2004**, 62, 93-110.
[203] H. Sarimveis, A. Alexridis, G. Bafas, *Neurocomput.*,**2003**,51, 501-505.
[204] A. Saastamoinen, T. Pietila, A. Varri, M. Lehtokangas, J. Saarinen, *Neurocomput.,***1998**,20, 1-3, 1-13.
[205] M. R. Berthold, J. Diamond, *NeuroComput.*, **1998**,19,167-183.
[206] L. Xu,*Neurocomput.*,**1998**,22, 81-112.
[207] Y. Li, M. J. Pont, N. B. Jones, *Pattern recog. Let.*,**2002**,23, 569-577.
[208] W. Zeng, C. Wang, F. Yang, *Pattern Recognition*, **2014** (in press).
[209] R. Liu, Y. Chen, L. Jiao, Y. Li, *Pattern Recognition,***2014**,47, 6, 2143-2152.
[210] M. R. Berthold, J. Diamond, *Proceedings of the Advances in Neural Information Processing Systems,***1995**,7,521–528.
[211] T. Poggio, F. Girosi,*Proceedings of the IEEE,* **1990**,78, 9, 1481–1497.
[212] M. Bagheri, S.A. Mirbagheri, M. Ehteshami, Z. Bagheri, *Process Safety Environmental Protection*, **2014** (in press).
[213] J.W. Kay and D.M. Titterington (Ed), Statistics and Neural networks, Advances at the interface **1999**, Oxford University Press, New York, 65-95.
[214] H. Toshani, M. Farrokhi, *Robotics Autonomous Systems.*,**2014**,62, 6, 766-781.
[215] G. E. Hinton, P. Dayan, B. J. Frey R. M. Neal, *Science.,***1995**,268, 1158-1161.
[216] M. B. Gholiv, Ali R. Jalalv, H. C. Goicoechea, T. Skov, *Talanta.,***2014***,119*, 553-563.
[217] A. J. Choobbasti, H. Tavakoli, S. S. Kutanaei, *Tunnelling Underground Space Technology.*,**2014**,40, 192-202.
[218] R. E. Shaffer, S. L.Rose-Pehrsson, *Anal. Chem.*, **1999**,71,4263-4271.
[219] P. D. Mosier, P. C. Jurs,*J Chem Inf Comput Sci.*,**2002**,42,1460-1470.
[220] E. Basçar, Memory and brain Dynamics Cognition, **2004**, *CRC Press, LLC New York.*
[221] L. Behera, M.Gopal, S. Chaudhury, *IEEE.Trans.Neural Netw.*,**1996**,7,1401-1414.
[222] T. Nitta, Complex-Valued Neural Networks: Utilizing High-Dimensional Parameters,**2009**, Information science reference, New York.
[223] Z. Liu, C. Mao, J. Luo, Y. Zhang, C.L. P. Chen, *Knowledge-Based Systems*, **2014**, 66, 13-27.
[224] F. Fernández-Navarroa, C. Hervás-Martíneza, R. Ruizc, J. C. Riquelmeb, *Appl. Soft Comp.*,**2012,** 12, 1787–1800.
[225] A. Ebrahimzadeh, B. Shakiba, A. Khazaee, *Appl. Soft Comp.*,**2014**, 22, 108-117.
[226] W. De Armas, K. A. Mamun, T. Chau, *Speech Commun.*,**2014**, 63–64, 15-26.
[227] Y. Fua, C. Wub, C. Koa, J. Jengc,*Applied Soft Comp.*, **2011**, 11, 3083–3092.
[228] K. Fu,G. Chen,Z. Liang, T. Sema,R. Idem,P. Tontiwachwuthiku, *Ind. Eng. Chem. Res.*, **2014**, 53, 4413−4423.
[229] S. Ji-yong, Z. Xiao-bo, H. Xiao-wei, Z. Jie-wen, L. Yanxiao, H. Limin, Z. Jianchun, *Food Chem.*,**2013**, 138(1), 192-199.

[230] Y. Chen, M. Xie, H. Zhang, Y. Wang, S. Nie, C. Li , *Food Chem.*,**2012**, 135(1), 268-275.
[231] Y. Ni, G. Zhang, S. Kokot, *Food Chem.*,**2005**, 89(3), 465-473.
[232] M. Abbasi-Tarighat, E. Shahbazi, K. Niknam, *Food Chem.*, **2013**, 138 (2–3) 991-997.
[233] Y. Ni, Y. Wang, S. Kokot ,*Food Chem.*,**2008**, 109 (2) 431-438.
[234] Q. Chen, C. Zhang, J. Zhao, Q. Ouyang, *TrAC Trends in Anal. Chem.*, **2013** (52), 261-274.
[235] M. Shoaib, A. Y. Shamseldin, B W. Melville, *J. of Hydrology*, **2014**, 515 (2014) 47–58.
[236] Rupenaguntla Naga Satish Chandra, **2003**, Master's thesis, University of Miami, USA.

Cancer research

[237] A. Jemal, F. Bray, M. M. Center, J. Ferlay, E. Ward, D. Forman, CA Cancer J. Clin. **2011**, 61, 69-90.
[238] A. D. Favia, O. Nicolotti, A. Stefanachi, F. Leonetti, A. Carotti, *Expert Opin. Drug Discov*., **2013**, 8, 395-409.
[239] N. Suvannang, C. Nantasenamat, C. Isarankura-Na-Ayudhya, V. Prachayasittikul, *Molecules*, **2011**, 16, 3597–3617.
[240] J. Narashimamurthy, A.R. Rao, G.N. Sastry, Curr. Med. Chem. Anticancer Agents., **2004**, 4, 523–534.
[241] J. K. Litton, B. K. Arun, P.H. Brown, G. N. Hortobagyi, *Expert. Opin. Pharmacother.*, **2012**, 13 325–331.
[242] C. Nantasenamat, H. Li, P. Mandi, A. Worachartcheewan, T. Monnor, C. Isarankura-Na-Ayudhya, V. Prachayasittikul, *Mol. Divers.*, **2013**, 17, 661–677.
[243] T.Hodgkin, *Med.-Chir. Trans.***1832,** 17, 68-114.