Geometrical Approaches for Artificial Neural Networks

David Elizondo
Centre for Computational Intelligence
De Montfort University
Leicester, UK
email:elizondo@dmu.ac.uk
http://www.cci.dmu.ac.uk/

Workshop on Principal Manifolds for
Data Cartography and Dimension Reduction
University of Leicester

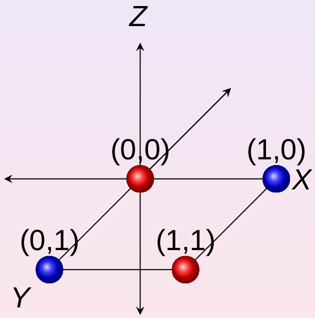
RDP Neural Network

- Multilayer neural network
- Generalisation of single layer perceptron for solving non linearly separable classification problems
- Automatic construction (SVM kernel)
- Convergence guaranteed
- Does not suffer from Catastrophic Interference
- No learning parameters
- Transparent Knowledge Extraction
- Generalisation level comparable to BP, CC, Rulex (Benchmarks, Satellite Images)

RDP Neural Network

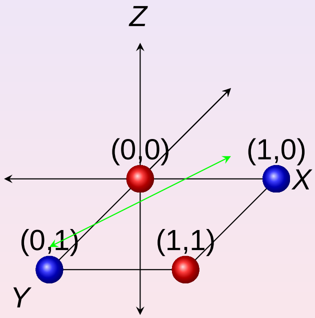
- Any NLS problem has at least one LS point (Vertice of the convex hull)
- Select LS subsets from within NLS sets
- Add dimension to input vector (Input * HP that separates LS subset from rest)
- Mark LS subsets as used
- Converge when either there are no more points left or the NLS problem becomes LS

RDP Neural Network



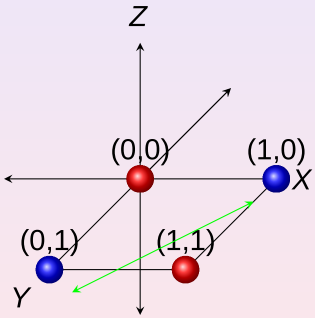
XOR			
I_0	I_1		Out
0	0		0
0	1		1
1	0		1
1	1		0

RDP Neural Network



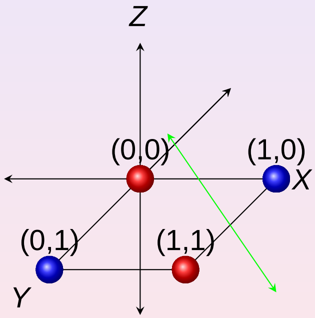
XOR			
I_0	I_1		Out
0	0		0
0	1		1
1	0		1
1	1		0

RDP Neural Network



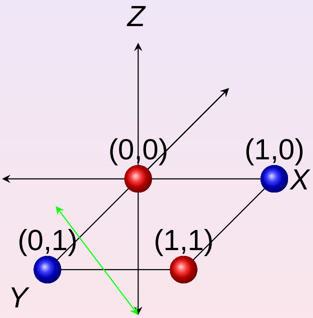
XOR			
I_0	I_1		Out
0	0		0
0	1		1
1	0		1
1	1		0

RDP Neural Network



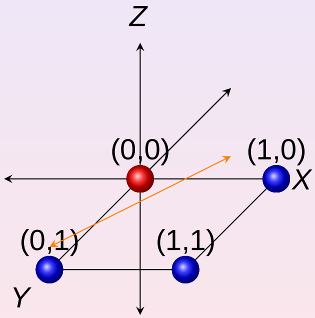
XOR			
I_0	I_1		Out
0	0		0
0	1		1
1	0		1
1	1		0

RDP Neural Network



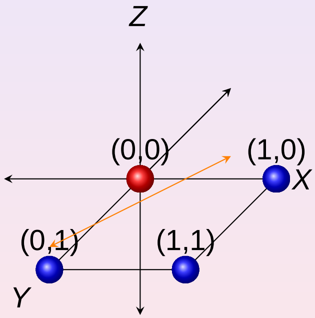
XOR			
I_0	I_1		Out
0	0		0
0	1		1
1	0		1
1	1		0

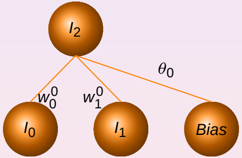
RDP Neural Network



XOR			
I_0	I_1		Out
0	0		0
0	1		1
1	0		1
1	1		0

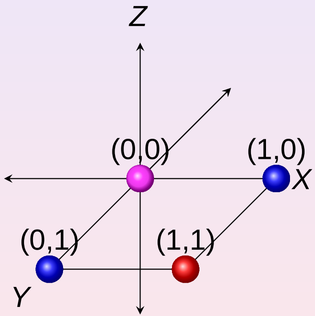
RDP Neural Network



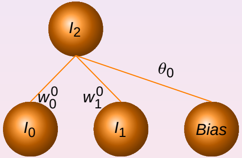


XOR			
I_0	I_1	I_2	Out
0	0	1	0
0	1	0	1
1	0	0	1
1	1	0	0

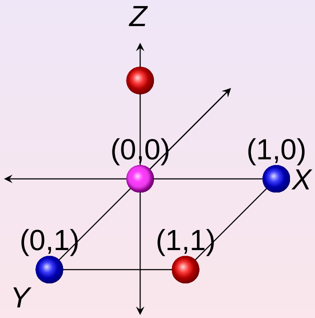
RDP Neural Network



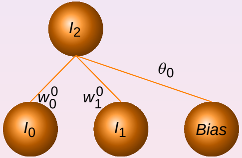
XOR			
I_0	I_1	I_2	Out
0	0	1	0
0	1	0	1
1	0	0	1
1	1	0	0



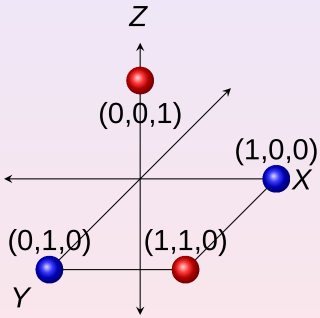
RDP Neural Network



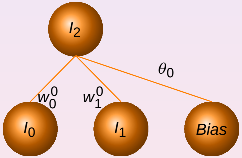
XOR			
I_0	I_1	I_2	Out
0	0	1	0
0	1	0	1
1	0	0	1
1	1	0	0



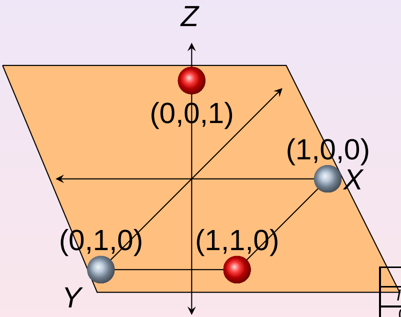
RDP Neural Network



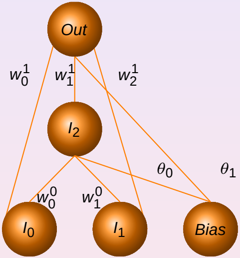
XOR			
I_0	I_1	I_2	Out
0	0	1	0
0	1	0	1
1	0	0	1
1	1	0	0



RDP Neural Network

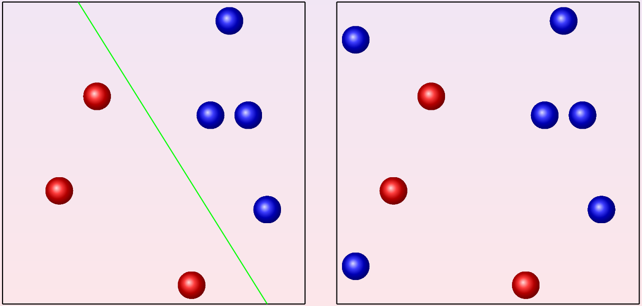


XOR			
I_0	I_1	I_2	Out
0	0	1	0
0	1	0	1
1	0	0	1
1	1	0	0



Linear Separability

Two subsets X and Y of R^d are said to be linearly separable (LS) if there exists a hyperplane P of R^d such that the elements of X and those of Y lie in opposite sides of it.



LS Data set NLS data set

Linear Separability

- **The methods based on solving systems of linear equations.** Fourier-Kuhn elimination, Simplex. Original classification problem represented as a set of constrained linear equations. If the two classes are LS, the two algorithms provide a solution to these equations.
- **The methods based on computational geometry techniques.** Convex hull, class of linear separability method. If two classes are LS, the intersection of the convex hulls of the set of points that represent the two classes is empty. The class of linear separability method consists in characterising the set of points P of R^d by which it passes a hyperplane that linearly separates two sets of points X and Y.

Linear Separability

- **The methods based on neural networks.** Perceptron. If the two classes are LS, the perceptron algorithm is guaranteed to converge, after a finite number of steps, and will find a hyperplane that separates them (Convergence Upper Bound).
- **The methods based on quadratic programming.** SVM. Find a hyperplane that linearly separates two classes by solving a quadratic optimisation problem.
- **The Fisher Linear Discriminant method.** Find a linear combination of input variables, $w \times x$, which maximises the average separation of the projections of the points belonging to the two classes C_1 and C_2 while minimising the within class variance of the projections of those points.

Methods for building RDP Neural Networks

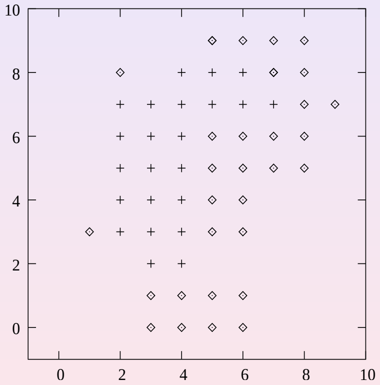
Three methods of construction

- Batch
- Incremental
- Modular
  - Modular/Batch
  - Modular/Incremental

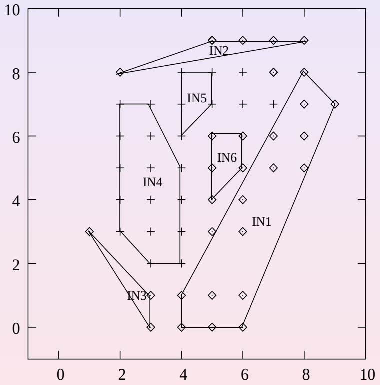
Batch Method

- Selection of LS which belong to the same class and have maximum cardinality
- Addition of dimension to input vector based on HP separating LS subset from rest of samples.
- Complexity NP-Complete

Batch Method

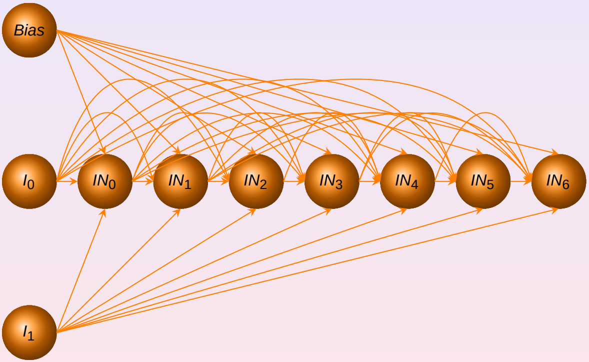


Original NLS Problem



Selected LS subsets

Batch Method

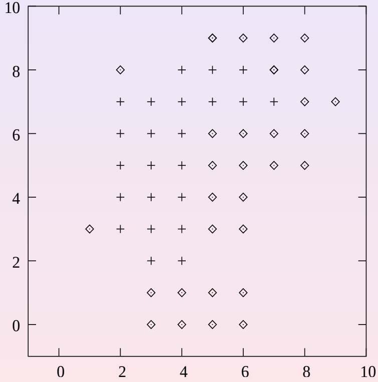


Architecture

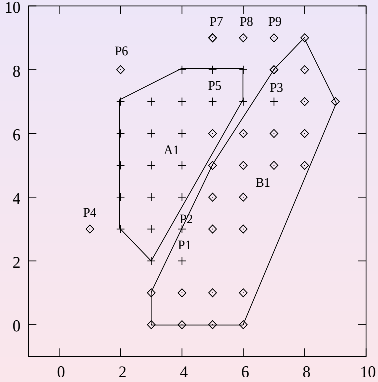
Incremental Method

- Trained using only a subset of the training data set.
- Start usually with all points belonging to class of maximum cardinality
- Add one point at the time from remaining points
- If new point misclassified, add a new HP
- Complexity O(n log n)

Incremental Method

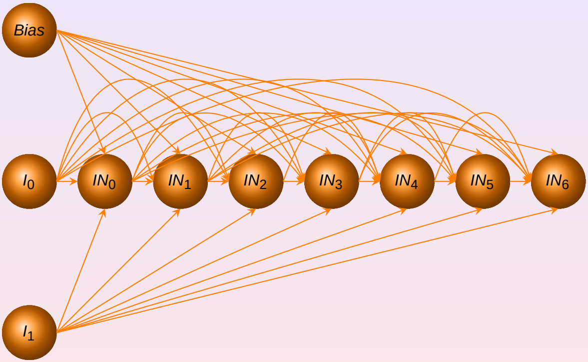


Original NLS Problem



Selected LS subsets

Incremental Method

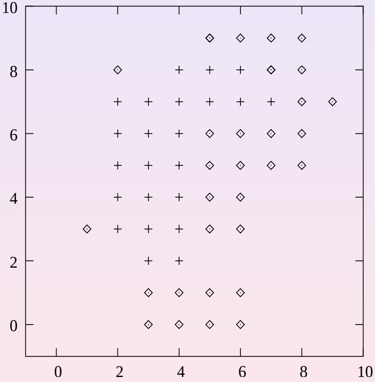


Architecture

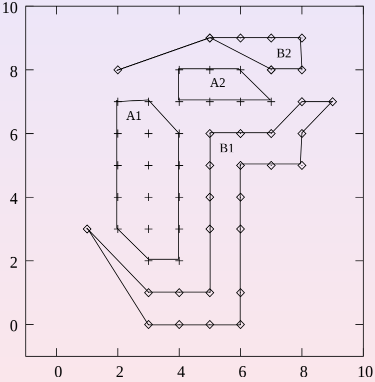
Modular Method

- Divide and conquer approach
- Split original problem into small problems.
- Train independently all sub problems using an RDP (Incremental/Batch)
- Join all sub RDP into a single RDP that can solve original problem

Modular Method

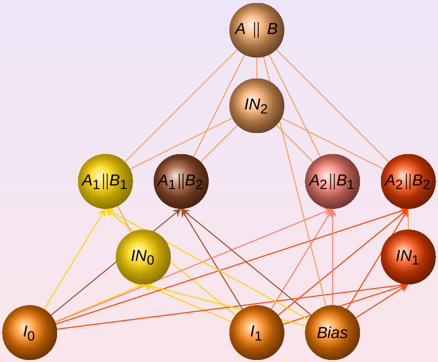


Original NLS Problem



Selected LS subsets

Modular Method



Architecture

Advantages and Limitations of Methods

- Batch method extensively tested with several data sets (result comparable to BP)
- Batch method will produce a small topology
- Batch method suffer from level of complexity
- Incremental method has $O(nlogn)$ complexity (Size of topology?)
- Modular method can be implemented in parallel (Can be combined with both Batch and Incremental methods)

Performance of Methods

- Three criteria to measure levels of performance:
  - Size of Topology
  - Level of Generalisation
  - Time of Convergence
- Three benchmarks used to compare
  - Iris
  - Soybean
  - Wisconsin Breast Cancer

Experimental Setup

- Cross Validation
- 10 train/test data sets
- 60 % train
- 40 % test

Topology Size

	Iris				Soybean				Wisconsin Breast Cancer			
Data Set	Batch	Incr	Mod Batch	Mod Incr	Batch	Incr	Mod Batch	Mod Incr	Batch	Incr	Mod Batch	Mod Incr
1	1	1	1	2	0*	0*	0*	0*	3	9	6	14
2	0*	0*	0*	0*	1	2	0*	0*	3	10	6	14
3	1	1	1	1	0*	0*	0*	0*	4	10	6	13
4	2	1	0*	0*	1	2	0*	0*	3	12	6	16
5	0*	0*	0*	0*	0*	0*	0*	0*	3	10	7	13
6	0*	0*	0*	0*	0*	0*	0*	0*	3	9	6	12
7	0*	0*	0*	0*	1	2	1	1	3	12	8	18
8	2	1	0*	0*	0*	0*	0*	0*	4	12	8	18
9	0*	0*	0*	0*	1	2	0*	0*	3	9	7	14
10	2	1	1	1	1	2	0*	0*	3	11	8	18
Δ	0.8	0.5	0.3	0.4	0.5	1	0.1	0.1	3.2	10.4	6.8	15

Topology Size

- Differences are not very dramatic in small and relatively simple data sets (Iris and Soybean).
- Incremental versions increase the number of intermediate neurons slightly.
- Topology differences seems relevant on larger more difficult data sets (Wisconsin)
- Increase in the topology size for Incremental method is just over 3 times.
- Increase in the topology size for Modular Incremental method is just under 6 times.

Level of Generalisation

	Iris				Soybean				Wisconsin Breast Cancer			
Data Set	Batch	Incr	Mod Batch	Mod Incr	Batch	Incr	Mod Batch	Mod Incr	Batch	Incr	Mod Batch	Mod Incr
1	92.5	92.5	100.0	97.5	52.1*	52.1*	52.1*	48.0*	90.0	93.9	95.4	97.0
2	95.0*	95.0*	87.5*	87.5*	62.5	79.2	79.1*	79.1*	97.0	94.2	94.1	95.6
3	97.5	97.5	95.0	97.5	70.8*	70.8*	72.9*	72.9*	92.7	94.0	97.0	92.5
4	92.5	90.0	87.5*	87.5*	83.3	70.8	83.3*	85.4*	94.2	97.0	98.5	97.0
5	95.0*	95.0*	95.0*	95.0*	64.6*	64.6*	83.3*	81.2*	95.7	94.1	95.6	98.5
6	85.0*	85.0*	90.0*	90.0*	68.7*	68.7*	77.0*	79.0*	90.0	90.0	91.4	94.3
7	95.0*	95.0*	92.5*	90.0*	89.6	79.2	79.1	85.4	97.0	95.5	95.5	94.0
8	92.5	87.5	97.5*	97.5*	68.7*	68.7*	72.9*	72.9*	92.7	94.2	97.1	98.6
9	95.0*	95.0*	92.5*	90.0*	83.3	77.1	83.3*	83.3*	94.2	91.3	94.2	97.1
10	92.5	92.5	97.5	100	77.1	85.4	85.4*	85.4*	95.7	92.9	94.3	93.0
Δ	93.25	92.5	93.5	93.25	72.1	71.6	76.8	77.6	93.9	93.7	95.3	95.7

Level of Generalisation

- Both the Batch and the Incremental methods offer comparable performance.
- The average (Δ) results on the level of generalisation obtained on both methods, using the three benchmarks, only differ by less than 1%.
- In the case of the Modular Batch and Modular Incremental networks, generalisation levels were slightly higher than those obtained with the Batch or Incremental methods. This is perhaps due to the extra degrees of freedom found on the Modular method.

Convergence Time

	Iris				Soybean				Wisconsin Breast Cancer			
Data Set	Batch	Incr	Mod Batch	Mod Incr	Batch	Incr	Mod Batch	Mod Incr	Batch	Incr	Mod Batch	Mod Incr
1	6.7s	0.2s	1.4s	0.2s	0.0*	0.0s*	0.0s*	0.1s*	2.5h	15.5s	1.0h	8.8s
2	0.0s*	0.0s*	0.0s*	0.0s*	15.6s	0.2s	0.0s*	0.1s*	2.5h	17.8s	1.9h	8.8s
3	6.7s	0.1s	1.5s	0.1s	0.0s*	0.0s*	0.0*	0.1s*	2.5h	16.3s	1.2h	7.1s
4	10.1s	0.1s	0.0s*	0.0s*	13.7s	0.2s	0.0s*	0.1s*	2.5h	20.9s	1.6h	9.2s
5	0.0s*	0.0s*	0.0s*	0.0s*	0.0s*	0.0s*	0.0s*	0.1s*	2.5h	15.9s	1.8h	9.0s
6	0.0s*	0.0s*	0.0s*	0.0s*	0.0s*	0.0s*	0.0s*	0.1s*	2.5h	16.6s	1.0h	8.5s
7	0.0s*	0.0s*	0.0s*	0.0s*	14.1s	0.2s	2.7s	0.2s	2.6h	21.0s	1.2h	10.8s
8	10.1s	0.1s	0.0s*	0.0s*	0.0s*	0.0s*	0.0s*	0.1s*	2.5h	20.0s	1.4h	10.7s
9	0.0s*	0.0s*	0.0s*	0.0s*	14.2s	0.2s	0.0s*	0.1s*	2.6h	18.5s	1.4h	10.3s
10	10.1s	0.1s	1.4s	0.1	14.3s	0.2s	0.0*	0.1s*	2.5h	18.8s	1.4h	10.7s
Δ	4.3s	0.1s	0.5s	0.0s	7.2s	0.1s	0.3s	0.1s	2.5h	18.1s	1.4h	9.4s

Convergence Time

- Alternative methods to the Batch produce a dramatic improvement in the construction of the RDP.
- For the Iris data set, the performance relative to the convergence time, the Incremental method executes 50 times faster
- Modular Batch 10 times faster, and the Modular Incremental 30 times.
- As the size of the data set increases, the improvement of the incremental methods becomes more apparent, as in the Soybean data set, the Incremental method executes 65 times faster, the Modular Batch 22 time faster, and the Modular Incremental 60 times.

RDP for m classes ($m > 2$)

- Generalization of the 2-class Recursive Deterministic Perceptron (RDP)
- Allows to always separate, in a deterministic way, m classes.
- Based on a new notion of linear separability
- The sets $X_1, ..., X_m \subset R^d$ are said to be linearly separable relatively to the ascending sequence of real numbers $a_0 <, ..., < a_m$
- M class problem translated into a 2 class problem and then solved as a regular 2 class RDP

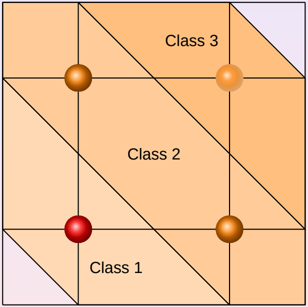
RDP for m classes ($m > 2$)

Let $X_1 = \{(0,0)\}$, $X_2 = \{(0,1),(1,0)\}$ $X_3 = \{(1,1)\}$, and $a_1 = 1; a_2 = 2, a_3 = 3$, and $a_4 = 4$.
X_1, X_2, and X_3 are LS if there exist w_1, w_2, t such that:

$$1 < t < 2 \qquad Class\ 1 \tag{1}$$
$$2 < w_1 + t < 3 \qquad Class\ 2 \tag{2}$$
$$2 < w_2 + t < 3 \qquad Class\ 2 \tag{3}$$
$$3 < w_1 + w_2 + t < 4 \qquad Class\ 3 \tag{4}$$
$$\tag{5}$$

Thus, $w_1 = w_2 = 1$, and $t = 3/2$ is a solution to this problem.

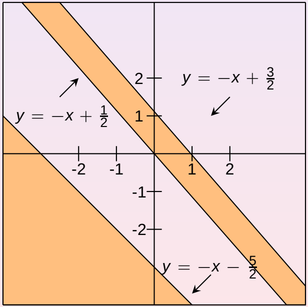
RDP for m classes ($m > 2$)



Hyperplanes that linearly separate the three classes

Knowledge Extraction

We can express, transparently, the knowledge embedded in a RDP neural network as set of definable regions which correspond to a finite union of open polyhedral sets of R^d.



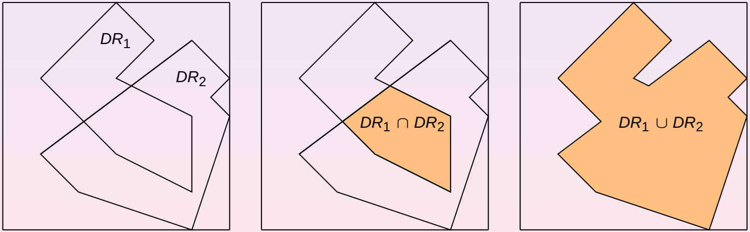
Definable region for the RDP P corresponding to the XOR problem

Boolean operations over the decision regions of an RDP neural network

Given two RDPs P_1 and P_2, we want to compose the following boolean operations by combining the two existing RDPs:

- The intersection of the two polyhedras and can be obtaining by using an RDP which calculates the logical AND function of the two RDPs.
- The union of the two polyhedras and can be obtaining by using an RDP which calculates the logical OR function of the two RDPs.
- The complement of the union of the two polyhedras and can be obtaining by using an RDP which calculates the logical AND function of the two RDPs and multiplying the weight vector and the threshold value by -1.

Boolean operations over the decision regions of an RDP neural network



Summary

- RDP Neural Network
- Linear separability (two novel methods Perceptron Upper bound and Class of linear separability)
- Three Methods for building RDP networks
- RDP for M class classification Problems
- Knowledge Extraction from an RDP

Perspectives

- Practical study of LS methods for best performance
- Use of M class RDP for function approximation
- Knowledge refining
- Partial connectivity on RDPs

References

D. A. Elizondo,
The RDP and some strategies for topology reduction of NN
PhD dissertation, ULP, Strasbourg, France, Jan. 1997.

M. Tajine, D. Elizondo.
The RDP Neural Network.
Neural Networks, 11:1571-1588, 1998.

M. Tajine, D. Elizondo.
Growing Methods for Constructing RDP NN & KE.
Artificial Intelligence, volume 102, pages 295-322, 1998.

D. Elizondo.
The Linear Separability Method Problem: Some Testing Methods.
IEEE Transactions on Neural Networks, 17(2):330-344,

References

- D. A. Elizondo,
  The RDP and some strategies for topology reduction of NN
  PhD dissertation, ULP, Strasbourg, France, Jan. 1997.
- M. Tajine, D. Elizondo.
  The RDP Neural Network.
  Neural Networks, 11:1571-1588, 1998.
- M. Tajine, D. Elizondo.
  Growing Methods for Constructing RDP NN & KE.
  Artificial Intelligence, volume 102, pages 295-322, 1998.
- D. Elizondo.
  The Linear Separability Method Problem: Some Testing Methods.
  IEEE Transactions on Neural Networks, 17(2):330-344,

References

- D. A. Elizondo,
  The RDP and some strategies for topology reduction of NN
  PhD dissertation, ULP, Strasbourg, France, Jan. 1997.
- M. Tajine, D. Elizondo.
  The RDP Neural Network.
  Neural Networks, 11:1571-1588, 1998.
- M. Tajine, D. Elizondo.
  Growing Methods for Constructing RDP NN & KE.
  Artificial Intelligence, volume 102, pages 295-322, 1998.
- D. Elizondo.
  The Linear Separability Method Problem: Some Testing Methods.
  IEEE Transactions on Neural Networks, 17(2):330-344,

References

- D. A. Elizondo,
  The RDP and some strategies for topology reduction of NN
  PhD dissertation, ULP, Strasbourg, France, Jan. 1997.
- M. Tajine, D. Elizondo.
  The RDP Neural Network.
  Neural Networks, 11:1571-1588, 1998.
- M. Tajine, D. Elizondo.
  Growing Methods for Constructing RDP NN & KE.
  Artificial Intelligence, volume 102, pages 295-322, 1998.
- D. Elizondo.
  The Linear Separability Method Problem: Some Testing Methods.
  IEEE Transactions on Neural Networks, 17(2):330-344,