Contributions of Lattice Computing to Medical Image Processing

Thesis dissertation

Darya Chyzhyk

Advisor: M. Graña
Computational Intelligence Group
University of Basque Country (UPV/EHU)
San Sebastián, Spain

July 26, 2013

Contents

Outline

Contents

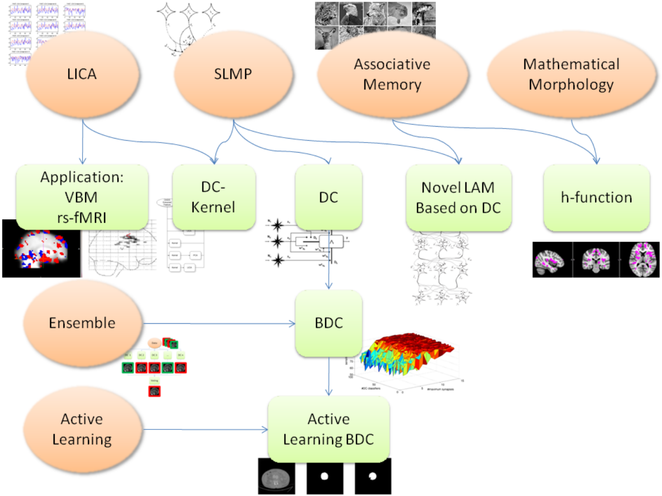
Motivation

This Thesis proceeds along two main lines:

- The exploration of new computational solutions based on the novel paradigm of Lattice Computing.
    - Lattice computing is the class of algorithms that either apply lattice operators inf and sup, involve the use of Lattice Theory results and operators lattice theory.
- The application to medical image data in order to obtain new image processing methods, and computer aided diagnosis systems based on image features that can be used as image biomarkers

Contents

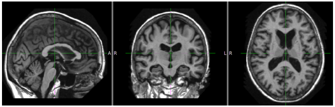
Contents of the Thesis

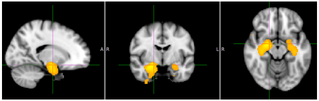


Contents of the Thesis

- Results of Dendritic Computing approaches
  - Single Layer Morphological Perceptron (SLMP) improvement by introducing shrinking hyperboxes, and Kernel LICA preprocessing
  - Ensembles of SLMP Dendritic Classifiers
  - Active learning
  - A novel Auto-Associative Memory
- Results on the application of LICA on three case studies:
  - Voxel Based Morphometry on anatomical MRI
  - Segmentation of fMRI data, i.e. synthetic benchmark data
  - Connectivity in resting state fMRI
- Application of Multivariate Mathematical Morphology to resting state fMRI

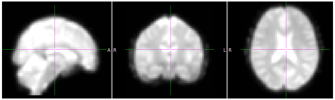
Medical Image Processing

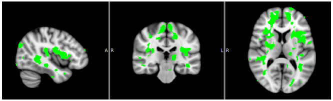




Alzheimer's Disease

- Is a neurodegenerative disorder
- Features extracted from anatomical MRI brain volumes
- Is valuable as a benchmark dataset

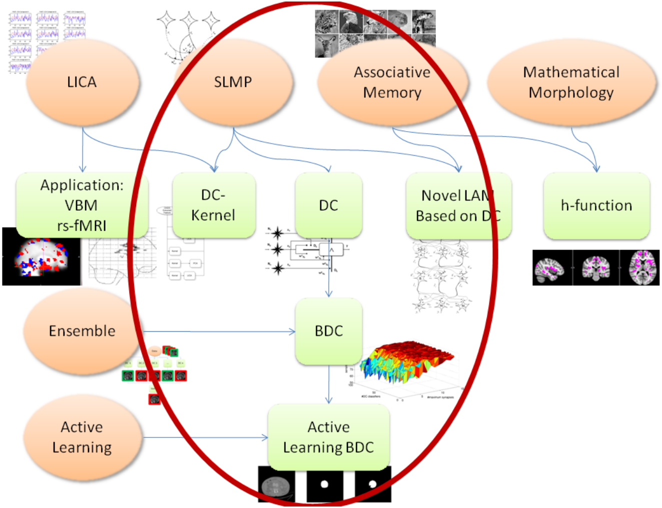




fMRI data

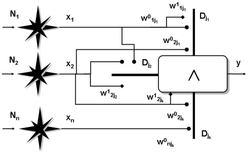
- Exploratory studies of LICA on synthetic benchmark fMRI data
- Connectivity studies on resting state fMRI data of Schizophrenia
  - LICA
  - Multivariate Mathematical Morphology

Dendritic computing



Dendritic computing

Dendritic Computing is based on the concept that dendrites are the basic information processing units of cortical neurons.



Perfect training accuracy: SLMP Dendritic Classifier has been proved to achieve perfect approximation of any data distribution.

However, they generalize badly when tested on conventional k-fold cross-validation schemes.

Dendritic computing

In order to improve generalization of SLMP we have followed various paths:

- Application of hyperbox reduction factor, which relaxes perfect approximation to obtain some improvement in the testing phase
- Performing appropriate combination with data transformations, specifically with the LICA approach and a kernel transformation of the data
- Collection of weak Dendritic Classifiers into ensemble by majority voting, which we call Bootstrapped Dendritic Computing

Dendritic computing

- Multi-layer Dendritic Computing allows to build robust Associative Memories
- Tested on a collection of heavily corrupted images

Lattice matrix products

- Given matrices A and B
- The matrix max -product operator denoted $\boxed{\vee}$ is defined as

$$C = A \boxed{\vee} B = [c_{ij}] \Leftrightarrow c_{ij} = \bigvee_{k=1..n} \left\{ a_{ik} + b_{kj} \right\},$$

- and the dual min-product matrix operator $\boxed{\wedge}$ is defined as

$$C = A \boxed{\wedge} B = [c_{ij}] \Leftrightarrow c_{ij} = \bigwedge_{k=1..n} \left\{ a_{ik} + b_{kj} \right\}.$$

Lattice Associative Memory

- Input/output pairs of patterns

$$(X, Y) = \left\{ \left(\mathbf{x}^{\xi}, \mathbf{y}^{\xi} \right); \xi = 1, .., k \right\}$$

- A linear heteroassociative neural network

$$W = \sum_{\xi} \mathbf{y}^{\xi} \cdot \left(\mathbf{x}^{\xi} \right)'.$$

- Erosive and dilative LAMs, respectively, are constructed as

$$W_{XY} = \bigwedge_{\xi=1}^{k} \left[\mathbf{y}^{\xi} \times \left(-\mathbf{x}^{\xi} \right)' \right] \text{ and } M_{XY} = \bigvee_{\xi=1}^{k} \left[\mathbf{y}^{\xi} \times \left(-\mathbf{x}^{\xi} \right)' \right],$$

where $\times$ is any of the $\boxed{\vee}$ or $\boxed{\wedge}$ operators.

Lattice Auto-Associative Memory

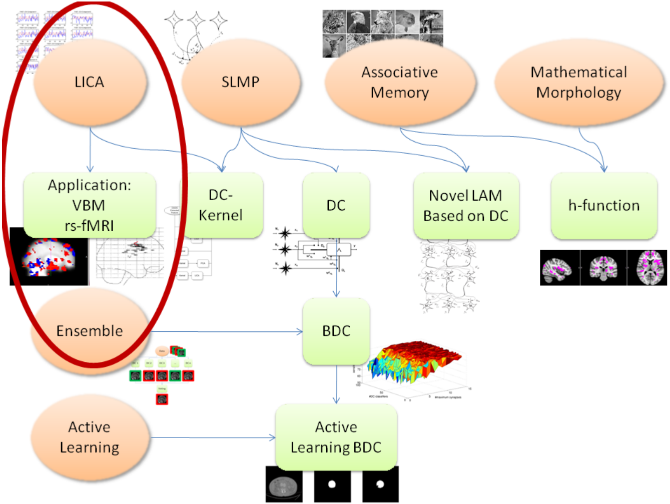
- When $X = Y$ then W_{XX} and M_{XX} are called Lattice Auto-Associative Memories (LAAMs)
- LAAM have perfect recall for an unlimited number of stored patterns

$$W_{XX} \boxed{\vee} X = X = M_{XX} \boxed{\wedge} X$$

and

- Convergence in one step for any input pattern
  - if $W_{XX} \boxed{\vee} \mathbf{z} = \mathbf{v}$ then $W_{XX} \boxed{\vee} \mathbf{v} = \mathbf{v}$
  - if $M_{XX} \boxed{\vee} \mathbf{z} = \mathbf{u}$ then $M_{XX} \boxed{\wedge} \mathbf{u} = \mathbf{u}$

Lattice Independent Component Analysis



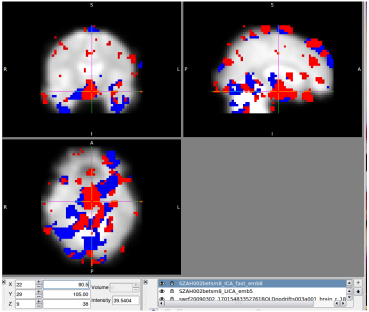
Lattice Independent Component Analysis

LICA assumes that the data is generated as a convex combination of a set of endmembers which are the vertices of a convex polytope covering the input data.

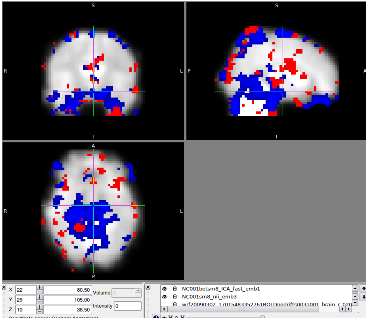
This assumption is similar to the linear mixture assumed by the ICA approach, however LICA does not impose any probabilistic assumption on the data.

Applications: analysis of fMRI data (synthetic datasets), Voxel Based Morphometry of structural MRI, and detecting functional connectivity in resting state fMRI.

Lattice Independent Component Analysis



Patient

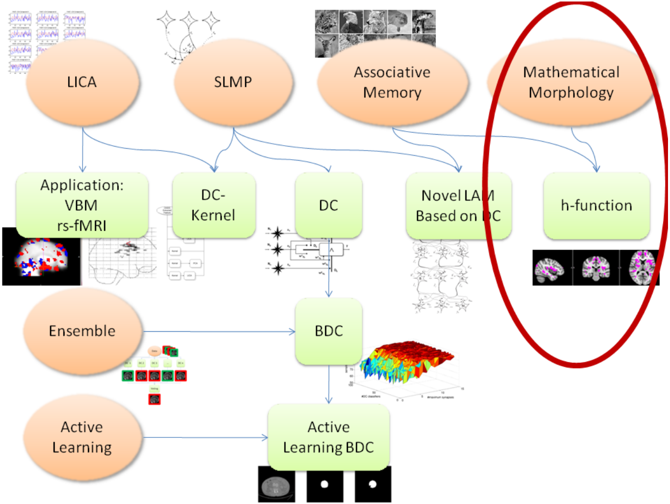


Control

Red corresponds to ICA detection, Blue to LICA detection.

Brain networks detected on Schizophrenia rs-fMRI by ICA and LICA approaches.

Multivariate Mathematical Morphology (MMM)



Multivariate Mathematical Morphology (MMM)

Extension of Mathematical Morphology from gray scale images to high dimensional vector images, i.e. functional Magnetic Resonance Images.

Fundamental issue in MMM is the definition of ordering over the multivariate data space ensuring morphological operators.

Technique consists in using the outputs of two-class classifiers trained on the data to build meaningful reduced orderings.

Classes are defined as foreground and background classes corresponding to target and background features of the data.

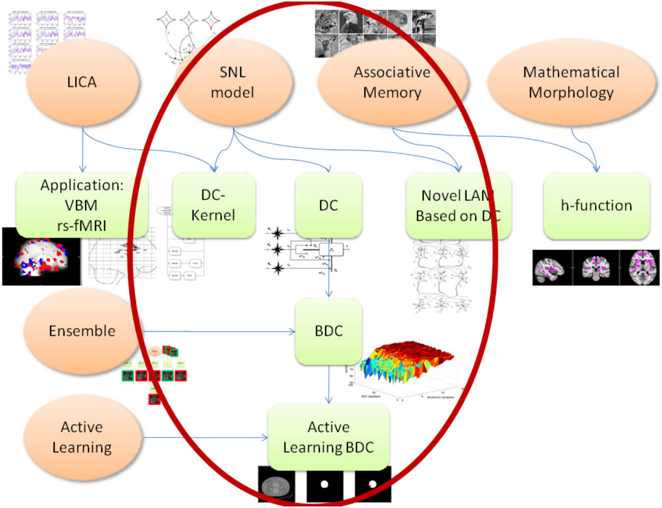
Multivariate Mathematical Morphology

LAAM. We have introduced several approaches to define reduced supervised orderings based on the recall error of the LAAM.

Application - identify functional networks in resting state fMRI data looking for biomarkers of cognitive or neurodegenerative diseases.

Outline

Dendritic computing



Contents

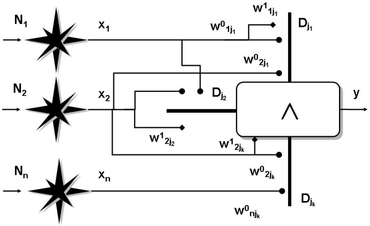
Dendritic Computation was introduced as a simple, fast, efficient biologically inspired method to build up classifiers for binary class problems.

Specifically the SLMP has been proved to perform a perfect approximation to any data distribution.

However SLMP suffers from over-fitting problems. Cross-validation experiments results show very poor performance.

Proposition: to apply a reduction factor on the size of the hyperboxes created by the SLMP learning algorithm. The results show a better balance between sensitivity and specificity, increasing the classifier accuracy.

Illustration of the structure of a single output class SLMP



- $N_1, \ldots, N_n$ - set of presynaptic neurons
- D_j - the dendrites
- $\left(w_{ij}^0, w_{ij}^1\right)$ - inhibitory and excitatory weights

Algorithm 2.1 Dendritic Computing learning algorithm based on elimination.

Training set $T = \left\{\left(\mathbf{x}^{\xi}, c_{\xi}\right) \mathbf{x}^{\xi} \in \mathbb{R}^{n}, c_{\xi} \in \{0,1\}; \xi = 1,\ldots,m\right\}$, $C_1 = \{\xi : c_{\xi} = 1\}$, $C_0 = \{\xi : c_{\xi} = 0\}$

1. Initialize $j = 1, I_j = \{1,\ldots n\}, P_j = \{1,\ldots,m\}, L_{ij} = \{0,1\}$,

$$w_{ij}^{1} = -\bigwedge_{c_{\xi}=1} x_i^{\xi};\ w_{ij}^{0} = -\bigvee_{c_{\xi}=1} x_i^{\xi}, \forall i \in I$$

2. Compute response of the current dendrite D_j, with $p_j = (-1)^{\mathrm{sgn}(j-1)}$:

$$\tau_j\left(\mathbf{x}^{\xi}\right) = p_j \bigwedge_{i \in I_j} \bigwedge_{l \in L_{ij}} (-1)^{1-l}\left(x_i^{\xi} + w_{ij}^{l}\right), \forall \xi \in P_j.$$

3. Compute the total response of the neuron:

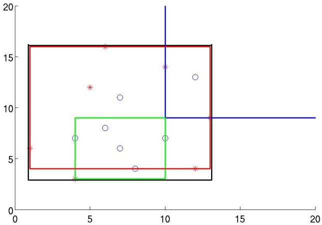
$$\tau\left(\mathbf{x}^{\xi}\right) = \bigwedge_{k=1}^{j} \tau_k\left(\mathbf{x}^{\xi}\right); \xi = 1,\ldots,m.$$

4. If $\forall \xi \left(f\left(\tau\left(\mathbf{x}^{\xi}\right)\right) = c_{\xi}\right)$ the algorithm stops here with perfect classification of the training set.
5. Create a new dendrite $j = j+1, I_j = I' = X = E = H = \emptyset, D = C_1$
6. Select $\mathbf{x}^{\gamma}$ such that $c_{\gamma} = 0$ and $f(\tau(\mathbf{x}^{\gamma})) = 1$.
7. $\mu = \bigwedge_{\xi \neq \gamma}\left\{\bigvee_{i=1}^{n}\left|x_i^{\gamma} - x_i^{\xi}\right| : \xi \in D\right\}$.
8. $I' = \left\{i : \left|x_i^{\gamma} - x_i^{\xi}\right| = \mu, \xi \in D\right\}; X = \left\{\left(i, x_i^{\xi}\right) : \left|x_i^{\gamma} - x_i^{\xi}\right| = \mu, \xi \in D\right\}$.
9. $\forall \left(i, x_i^{\xi}\right) \in X$
   (a) if $x_i^{\gamma} > x_i^{\xi}$ then $w_{ij}^{1} = -(x_i^{\xi} + \mu), E_{ij} = \{1\}$
   (b) if $x_i^{\gamma} < x_i^{\xi}$ then $w_{ij}^{0} = -(x_i^{\xi} - \mu), H_{ij} = \{0\}$
10. $I_j = I_j \bigcup I'; L_{ij} = E_{ij} \bigcup H_{ij}$
11. $D' = \left\{\xi \in D : \forall i \in I_j, -w_{ij}^{1} < x_i^{\xi} < -w_{ij}^{0}\right\}$. If $D' = \emptyset$ then goto step 2, else $D = D'$ goto step 7

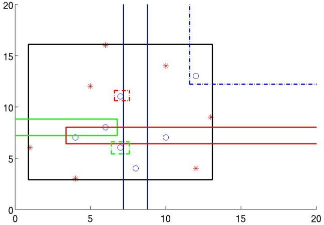
Construct a basic hyperbox

Adding of dendrites to remove misclasified patterns of class 0 that fall inside this hyperbox

Resulting boxes on a synthetic 2D dataset

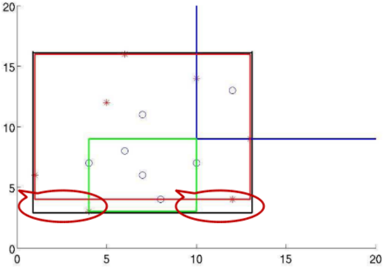


Original SLMP learning

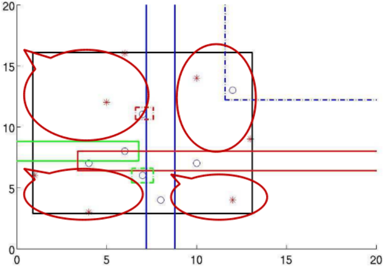


Shrinking factor $\alpha = 0.8$

Resulting boxes on a synthetic 2D dataset



original SLMP learning

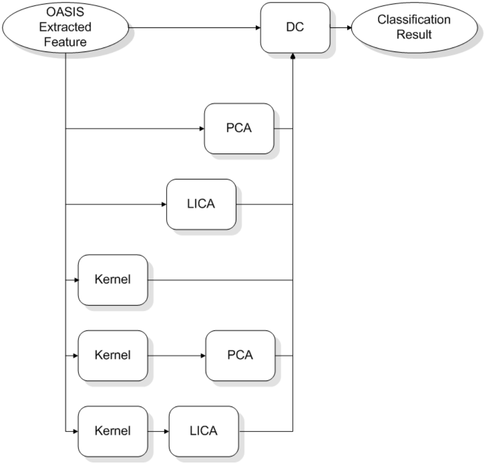


shrinking factor $\alpha = 0.8$

- Better balance of specificity and sensitivity by shrinking the boundaries of the hyperbox.
- Shrink patient class regions to reduce misclassification of control samples.

Contents

Lattice Independent Component Analysis (LICA) and the Kernel transformation of the data as an appropriate feature extraction that improves the generalization of SLNDC



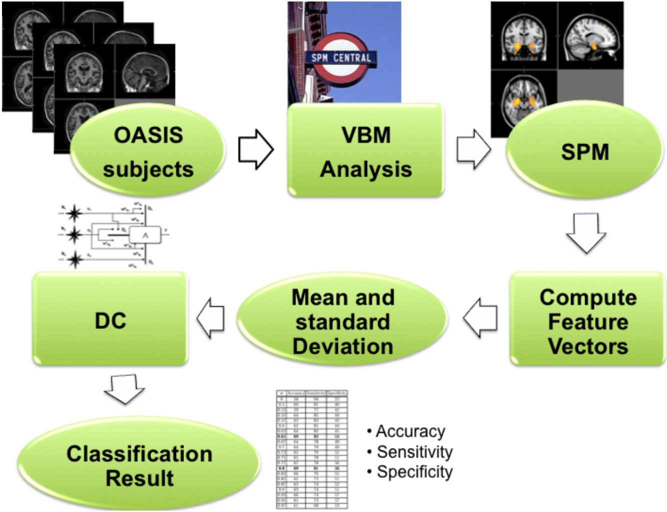


Figure: Pipeline of the process performed, including VBM, feature extraction and classification by DC

Method	NE	α	σ	Accuracy	Sensitivity	Specificity
DC	-	-	-	58	94	23
DC shrinking	-	-	-	**69**	81	56
PCA - DC	1	-	-	68.25	85.5	51
LICA - DC	1	7	-	**72**	88	56
Kernel - DC	-	-	0.2512	55	98	12
Kernel - PCA - DC	8	-	0.0794	66.5	96	37
Kernel - LICA - DC	3	2	0.5012	**74.25**	96	52.5

Table: Summary of best results of validation experiments over AD feature database.

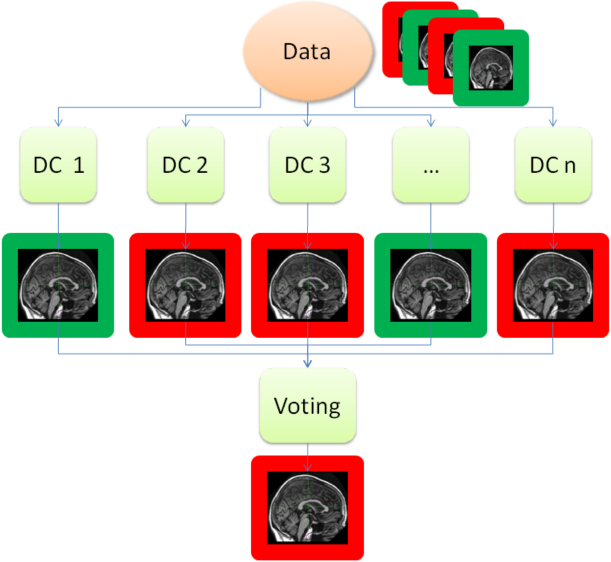
Contents

Bootstrapped Dendritic Classifiers (BDC) is an ensemble of weak Dendritic Classifiers, combining their output by majority voting to obtain improved classification generalization performance.

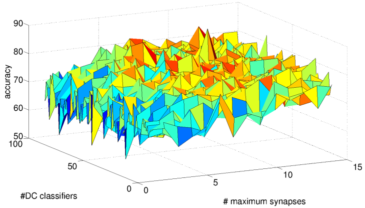
Weak Dendritic Classifiers are trained on bootstrapped samples of the train data setting a limit on the number of dendrites.

There is no additional data preprocessing.

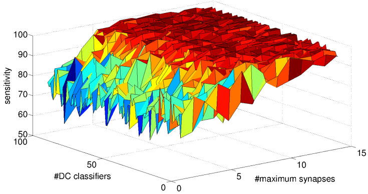
Results performance and the sensitivity to the number of classifiers and the number of dendrites on the classification of Alzheimer's Disease patients, comparing with previous results obtained on the same feature database.


Data
DC 1
DC 2
DC 3
...
DC n
Voting

Average Accuracy and average Sensitivity for varying number of DC classifiers and maximum number of dendritic synapses



Accuracy



Sensitivity

Table: Best results over the OASIS data MSD features for AD detection

Classifiers	Accuracy	Sensitivity	Specificity
rbf SVM	81	89	75
LVQ1	81	90	72
LVQ2	83	92	74
rbf-DAB-SVM	85	92	78
rbfRVM-LVQ1	87	92	73
LICA - DC	72	88	56
Kernel - LICA - DC	74	96	52.5
Bootstrapped DC	**89**	**100**	**80**

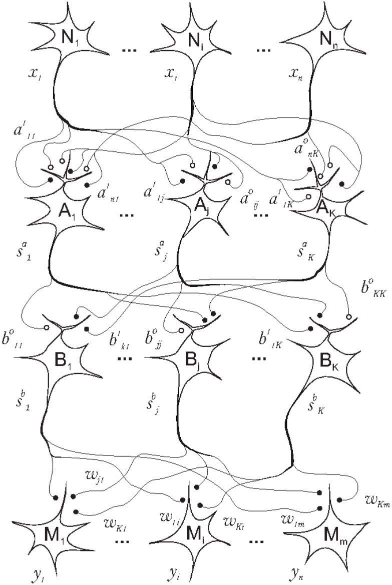
Contents

A Novel Lattice Associative Memory Based on Dendritic Computing

Present a novel hetero-associative memory based on dendritic neural computation.

Proposed multi-layer model

- **allows** to store any finite number of input/output pattern pairs,
- **complexity** grows linearly with number of stored pairs,
- **extremely** robust in the presence of various types of noise and data corruption.



1.- $N_1, \ldots, N_n$ - an input layer

2.- $A_1, \ldots, A_K$ - the first hidden layer
Computes the L_1-distance between the input pattern $\mathbf{x}$ and the jth exemplar pattern $\mathbf{x}^j$

3.- $B_1, \ldots, B_K$ - the second hidden layer
which pattern vector $\mathbf{x}$ is closer to the exemplar pattern $\mathbf{x}^j$, establish the threshold T.

4.- $M_1, \ldots, M_m$ - an output layer

Noise Robustness Experiment

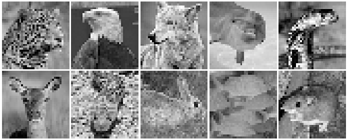


Figure: Set of grayscale image pairs: Upper row, input images (5 Predators). Lower row corresponding output images (5 Preys).

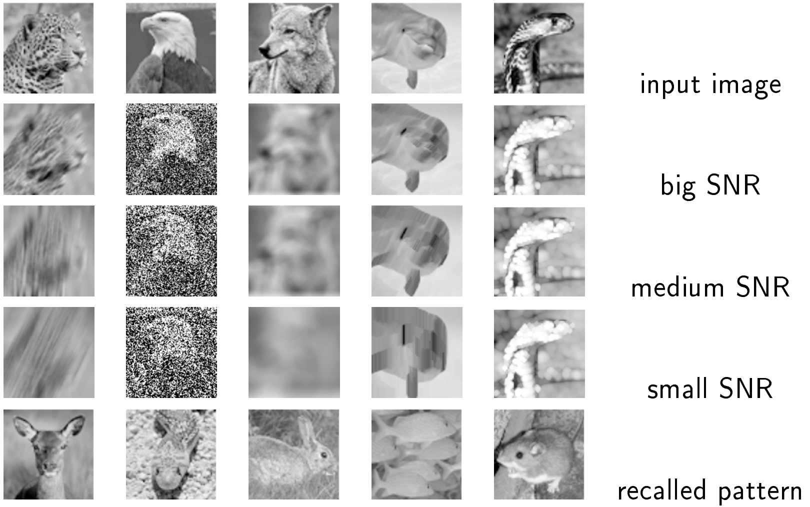
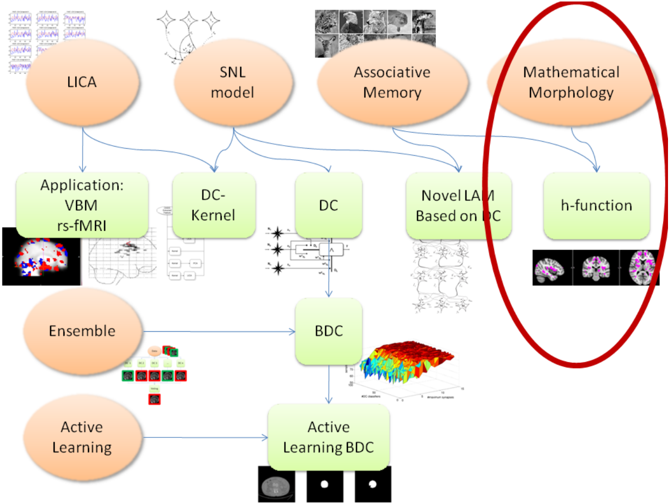


Figure: DLAM recalled patterns for diverse noise scales are identical and perfect

Outline

Multivariate Mathematical Morphology (MMM)



Contents

Introduction

Approach to define Multivariate Mathematical Morphology

based on the definition of a supervised ordering

built on the Lattice Auto-associative Memory recall error

Results on the application on resting state fMRI to discover functional connectivity

General approach consists in the application of Lattice Auto-Associative Memories (LAAMs) to the definition of a *LAAM-supervised ordering*, an specific kind of h-ordering.

Allowing the consistent definition of morphological operators on multivariate data.

All the required calculations are defined using the Lattice algebra operators ($\vee$, $\wedge$ and $+$).

Therefore, LAAM-supervised ordering is faster and imposes less computational burden than the supervised orderings previously proposed.

Contents

Multivariate Mathematical Morphology

The two elementary morphological operators

- erosion: an operation that is distributive with the minimum

$$\varepsilon\left(\bigwedge Y\right) = \bigwedge_{y \in Y} \varepsilon(y)$$

- dilation: distributive with the maximum

$$\delta\left(\bigvee Y\right) = \bigvee_{y \in Y} \delta(y)$$

Image morphological filters:

- morphological gradient

$$g(Y) = \delta(Y) - \varepsilon(Y)$$

- the top-hat

$$t(Y) = Y - \delta(\varepsilon(Y))$$

Multivariate ordering

Morphological operators are well defined for scalar images

However their extension to multivariate images is not straightforward

Properties of the morphological operators not preserve (new colors are generation)

h-supervised ordering is defined

$$\mathbf{x} \leq_h \mathbf{y} \Leftrightarrow h(\mathbf{x}) \leq h(\mathbf{y}); \forall \mathbf{x}, \mathbf{y} \in X.$$

Multivariate morphological operators

- The h-supervised erosion of a multivariate image $\{I(p) \in \mathbb{R}^n\}_{p \in D_I}$, with structural object S, is defined as follows:

$$\varepsilon_{h,S}(I)(p) = I(q) \text{ s.t. } I(q) = \bigwedge_h \{I(s); s \in S_p\}$$

  - where $\bigwedge_h$ is the infimum defined by the reduced ordering $\leq_h$, and
  - S_p is the structural element translated to the pixel position p.
- The h-supervised dilation $\delta_{h,S}(I)(p)$, has dual definition based on supremum $\bigvee_h$.
- h-supervised morphological gradient can be defined as follows:

$$g_{h,S}(I) = h\left(\delta_{h,S}(I)\right) - h\left(\varepsilon_{h,S}(I)\right).$$

Contents

LAAM's h-mapping

- The LAAM h-mapping is defined as the Chebyshev distance between the original pattern vector and the recall obtained from the LAAM.
- Formally, given a sample data vector $\mathbf{x} \in \mathbb{R}^n$ and a non-empty training set $X = \{\mathbf{x}_i\}_{i=1}^{K}$, $\mathbf{x}_i \in \mathbb{R}^n$, the LAAM h-mapping is given by:

$$h_X(\mathbf{c}) = d_C\left(\mathbf{x}^{\#}, \mathbf{x}\right),$$

  - where $\mathbf{x}_M^{\#} = M_{XX} \boxtimes \mathbf{x}$,
  - function $d_C(\mathbf{a}, \mathbf{b})$ denotes the Chebyshev distance:

$$d_C(\mathbf{a}, \mathbf{b}) = \bigvee_{i=1}^{n} |a_i - b_i|.$$

Foreground LAAM h-supervised ordering

- Given a training set X.
- Foreground LAAM h-supervised ordering, denoted by $\leq_X$, is defined on the LAAM h-mapping as follows:

$$\forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n, \ \mathbf{x} \leq_X \mathbf{y} \Longleftrightarrow h_X(\mathbf{x}) \leq h_X(\mathbf{y}).$$

- The Foreground LAAM-supervised ordering generates a complete lattice $\mathbb{L}_X$,
    - bottom element $\perp_X = 0$ corresponds to the set of fixed points of M_{XX} and W_{XX}, i.e. $h(\mathbf{x}) = \perp_X$ for $\mathbf{x} \in \mathcal{F}(X)$
    - top element is $\top_X = +\infty$.

Background/Foreground LAAM h-supervised orderings

- Given disjoint background B and foreground F training sets.
- The Foreground LAAM h-mapping is independently applied to the data using B and F as training sets, obtaining mappings h_B and h_F, respectively.
- We define a Background/Foreground (B/F) LAAM h-mapping $h_r(\mathbf{x})$ as follows:

$$h_r(\mathbf{x}) = h_F(\mathbf{x}) - h_B(\mathbf{x}),$$

which is positive for $\mathbf{x} \in \mathcal{F}(B)$, and negative for $\mathbf{x} \in \mathcal{F}(F)$.

Background/Foreground LAAM h-supervised orderings

- The Background/Foreground (B/F) h-supervised ordering, denoted $\leq_r$, is defined as follows:

$$\forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n, \ \mathbf{x} \leq_r \mathbf{y} \Longleftrightarrow h_r(\mathbf{x}) \leq h_r(\mathbf{y}).$$

- The image of the B/F LAAM h-mapping is a complete lattice $\mathbb{L}_r$
  - bottom and top elements are $\perp = -\infty$ and $\top = +\infty$, respectively.

Contents

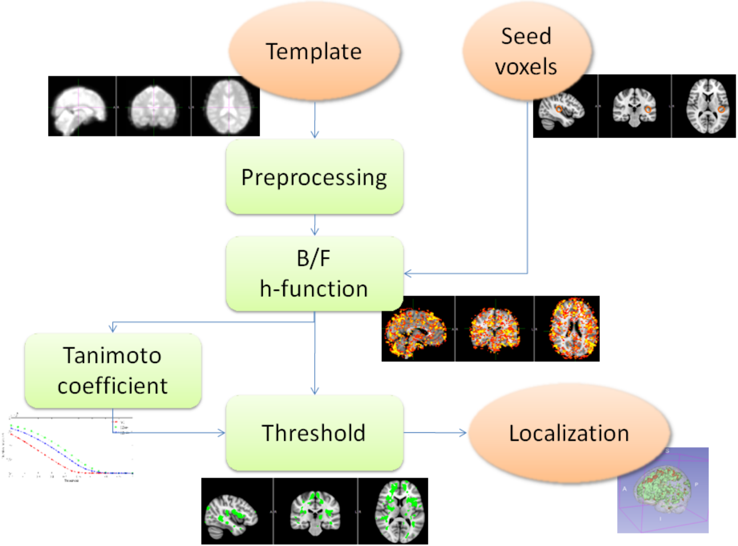
rs-fMRI images

- We consider rs-fMRI data of healthy controls (HC), schizophrenia patients with and without auditory hallucinations (SZAH and SZnAH, respectively).
- A seed voxel BOLD time series is used to build a LAAM, which is then applied to the remaining voxels of the brain fMRI 4D data.
- The *h*-function provides the functional similarity for brain network identification.
- The map obtained from the whole brain volume is thresholded to detect functional connectivity.

Two experiments performed on the rs-fMRI data:

1. Whole brain volume group analysis
   - we build one template for each population (HC, SZAH and SZnAH) by averaging the registered 4D data
   - background/foreground h-function map on each template
   - explore the network induced on each template by an specific localization in the left Heschl's gyrus of the brain
   - optimal threshold value is decided by inspection
     - minimal Tanimoto coefficients between the functional networks of each template.
     - maximum cluster size
   - visual comparison of detected networks
2. Classification results.
   - we build the B/F *h*-function map related to the left Heschl's gyrus on each subject
   - feature selection Pearson correlation coefficient between the *h*-function values and the categorical variable at each voxel site
   - feature vectors are constructed as the *h*-function values at these sites
   - results with the baseline k-NN classifiers

Experiment 1



Experiment 1

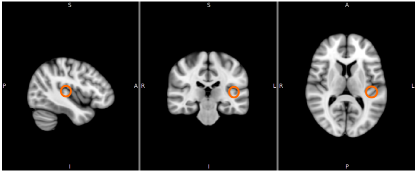


Figure: Foreground voxel seed site from the left Heschl's gyrus (LHG; -42,-26,10).

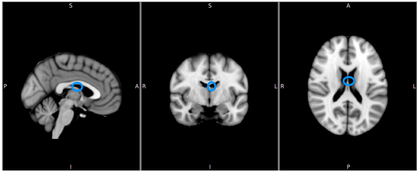
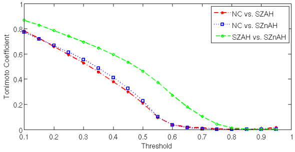
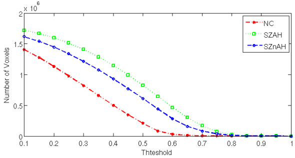


Figure: Background voxel seed site from CSF of the ventricle.

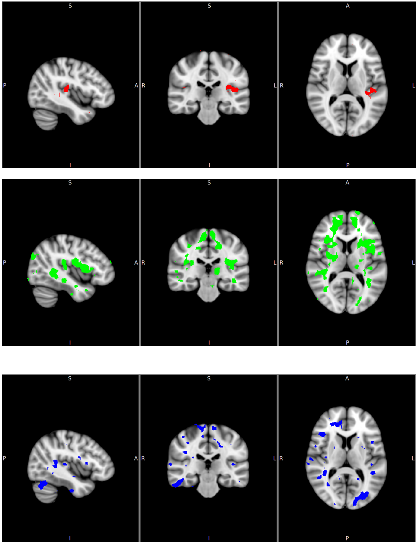


(a)



(b)

Figure: Effect of threshold value on the identified networks on background/foreground *h*-function brain map. (a) Tanimoto Coefficient comparing networks from each pair of population, and (b) size of the detected clusters.



healthy controls

schizophrenics with hallucinations

schizophrenics without hallucinations

Figure: Networks identified by thresholding the Background/Foreground *h*-function induced by the pair of background/foreground seeds

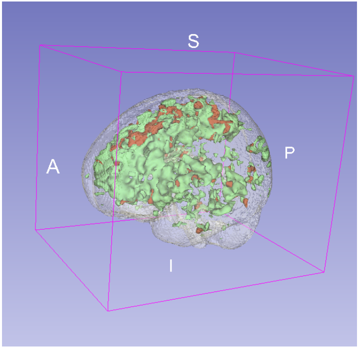


Figure: 3D visualization of the brain networks appearing only in the SZAH population template (green), and the common networks between SZAH and SZnAH populations (brown).

Experiment 2

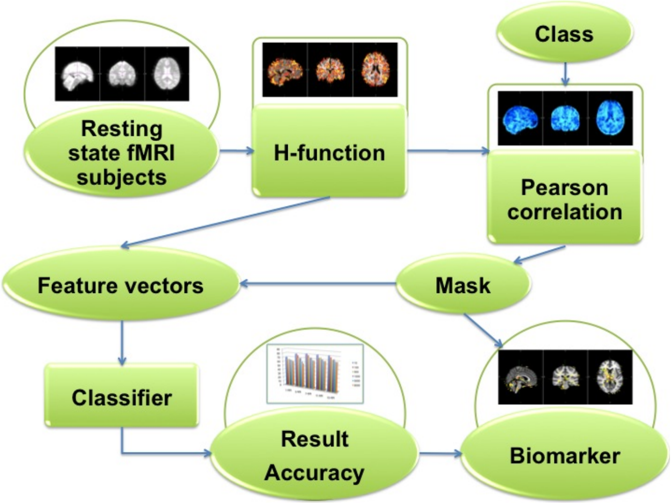
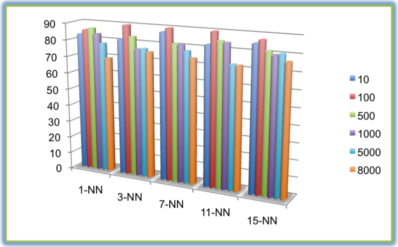
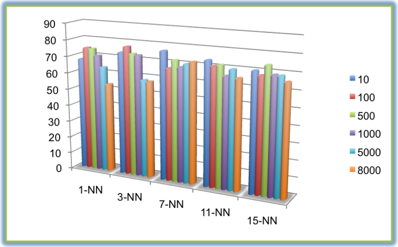


Figure: Pipeline of classification experiment



HC vs. nAH



nAH vs. AH

Figure: Maximum Classifier Accuracy found in 10 repetition of 10-fold cross validation for k-NN classifier $k = 1, 3, 7, 11, 15$. The bar colors represent different number of extracted features.

Outline

1. Introduction
   - Motivation
   - Contents of the Thesis
2. Dendritic Computing
   - Optimal Hyperbox shrinking
   - Hybrid Dendritic Computing with Kernel-LICA
   - Bootstrapped Dendritic Classifiers
   - A Novel Lattice Associative Memory Based on Dendritic Computing
3. Lattice Computing Multivariate Mathematical Morphology
   - Introduction
   - Multivariate Mathematical Morphology
   - LAAM-Supervised Ordering
   - Experimental results on rs-fMRI
4. Conclusions

Conclusions: DC

Problems: over-fitting, degradetion performance when applied cross-validation test.

Relax this over-fitting, found balance between sensitivity and specificity, we introduce a shrinking factor of the hyperboxes.

Preprocessing of the data: linear transformations (PCA), non-linear transformations (Gaussian kernel and LICA), and combinations.

Proposed the Bootstrapped Dendritic Classifier (BDC), which is a collection of weak SLMP trained on independently bootstrapped samples. Classification decision is made by majority voting.

Dendritic Lattice Associative Memory (DLAM) is a multilayer architecture of dendritic computing. DLAM possesses strong robustness against noise corruption.

Conclusions: MMM

Extension
: of Mathematical Morphology to multivariate data, such as fMRI 4D volumes

Definition
: of reduced ordering maps by supervised classification approaches

    Lattice
    : Auto-Associative Memories (LAAM) recall error measured by the Chebyshev

    definition
    : of LAAM based Foreground and Background/Foreground reduced orderings, corresponding to one-class and two-class supervised classification approaches,

    definition
    : of morphological operators and filters

Conclusions: Application

Worked on a set of feature vectors extracted from a subset of structural images from an Alzheimer's Disease public database (OASIS).

Multilayer DLAM was tested on a collection of grayscale images subjected to diverse degrees of noise corruption.

Applying LAAM to resting state fMRI data of Schizophrenia patients versus controls we obtain discriminant brain networks that can be appreciated visually and also serve for feature extraction purposes with good classification performance results using baseline k-NN classifiers.

Contributions of Lattice Computing to Medical Image Processing
Thesis dissertation

Darya Chyzhyk

Advisor: Prof. Dr. Manuel Graña

Thank you very much for your attention!