CogniMem
Neural Network Technology

Decision Space Mapping

Version 1.1

Revised 07/22/2011

CogniMem Technologies, Inc.

Table of Contents

Table of Contents 2
Introduction 3
Building a decision space 4
Example of a 2D decision space 4
Examples of N-dimensional space 4
Construction with an RBF model generator 6
Advantages of the RBF classifier for recognition 7
Three possible classification status 8
Multi-level responses 8
Flexibility of the RBF model generator 9
Non-linear model generator 9
Modulating liberalism by teaching counter examples 10
Modulating liberalism with the Maximum Influence Field 11
Shaping the decision space with a Norm 11
Norm L1 (Manhattan distance) 12
Norm L Sup 12
Norm L2 (Euclidian distance) 12
Building zones of uncertainty 12
Modulating uncertainty zones with the Minimum Influence Field 13
Benefits of Iterative learning 13
Incremental Decision Space mapping 13
Learning curve and feature relevancy 15
KNN (K-Nearest Neighbor) classifier 16
From decision space to knowledge base 17
Bibliography 18
Index 18

Introduction

Pattern recognition starts with the definition of a decision space suitable to discriminate different categories of objects. A decision space can be represented by a graph with N dimensions where N is the number of attributes or measurements considered to represent the objects. The N attributes compose a feature vector or signature which can be plotted in the graph. After significant samples have been entered, the decision space starts taking shape and hopefully revealing clusters of objects belonging to different categories. It can then be used to associate new vectors to these clusters.

The CogniMem chip is a chain of identical neurons which have their own “genetic” material to learn and recall patterns. They can build a non-linear decision space following the adaptive Radial Basis Function (RBF) model generator. They can classify new vectors with this “RBF” model or with a K-Nearest Neighbor classifier.

This manual describes how a decision space can be built with the CogniMem neurons and the powerful features they offer.

Building a decision space

Example of a 2D decision space

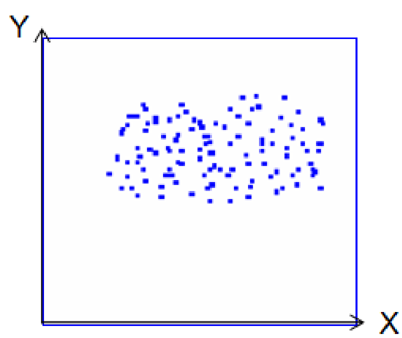
If a population of objects is characterized by two measurements (X,Y), their distribution can be plotted and visualized in a two dimensional graph as seen below. The characterization of population with N measurements instead of 2 just becomes an extension of the illustrations below.

Given a set of (X,Y) collected over many samples, the spatial distribution of these X and Y data can outline significant clusters of points. These clusters can be associated to categories or classes of objects.

"The data depicted in the following graphs is fictitious. Any similarity to any person is merely coincidental. :=)"

Case study 1:

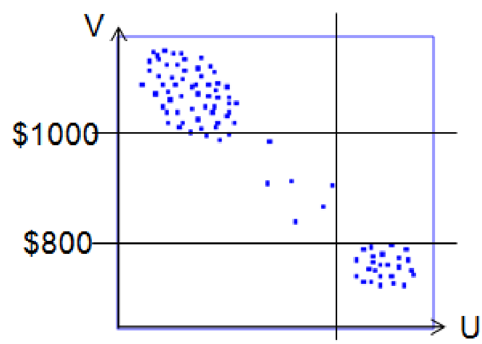
X= Weight
Y= Days of vacation per year



No specific cluster can be outlined. The measurements X and Y are either irrelevant or insufficient to reveal categories of persons within the surveyed population.

Case study 2:

U = Shoe size
V = Annual budget for cosmetics



Two obvious and disjoint clusters appear: (1) persons with shoe size above 10 and spending less than $800/yr in cosmetics; (2) persons with shoe size under 10 and spending over $1000/yr in cosmetics. A few outliers can be observed between the two clusters.

If the intent of the survey is to discriminate males and females, the measurements (U,V) will do a better job than the measurements (X,Y). Other measurements might perform even better.

Examples of N-dimensional space

The selection and availability of the measurements used to model the decision space are critical since they can render a classification obvious, uncertain or undetermined.

In neural network technology, the set of measurements (X,Y) is called the feature vector or signature vector. It can have a dimension n if it is composed of n values (X1, X2, …Xn). The Xi measurements can be unrelated, or they can be linked and represent a progression or variation in space, time, color or another dimension.

Example 1: Vector of unrelated measurements with different dimensions

For the monitoring of prospective customers, a neural network can be trained to classify people as high, medium, low and unlikely buyers. The decision space can be modeled by assembling the following values into a feature vector:

X1= age
X2= gender
X3= ethnicity
X4= average income
X5= single or married
X6-X8= number of children
etc

Example 2: Vector representing a spatial arrangement

For the recognition of fingerprint, the vector is an assembly of minutia points related in the spatial domain.

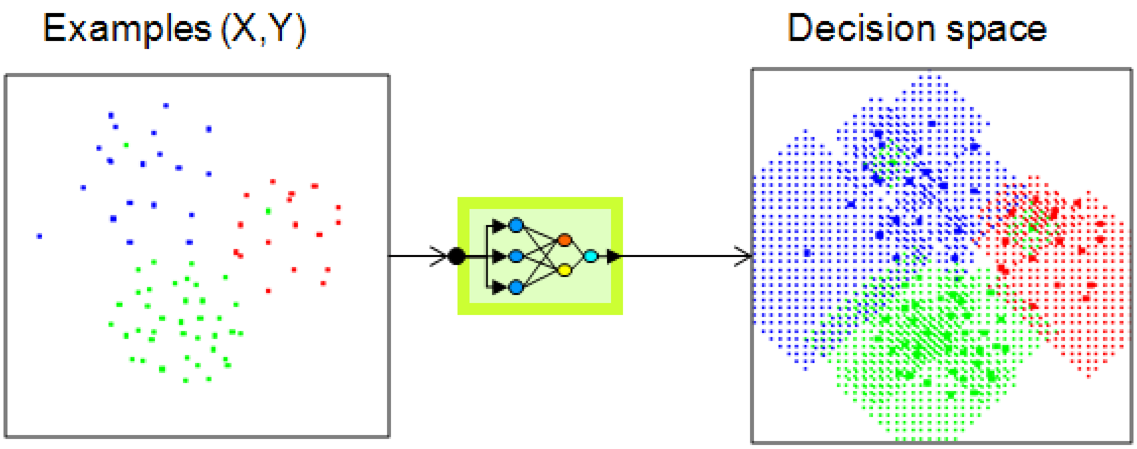
Example 3: Vector representing a time progression

For the monitoring of an EKG, the vector is a repetitive time series of a heart beat amplitude.

Construction with an RBF model generator

Neural networks are trainable and good at modeling decision spaces with some fuzzy logic. They can generalize what they are taught and therefore recognize patterns which they have never seen before.

The decision space is mapped by teaching a series of examples and labeling them with a category.



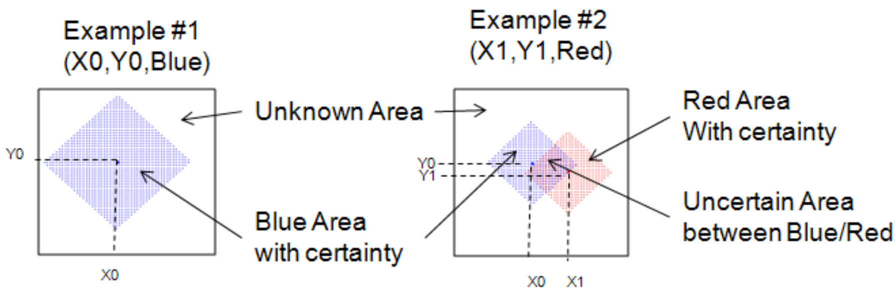
Graph1 Graph2

In Graph1, the example vectors (X,Y) are plotted and their color indicates their category C.

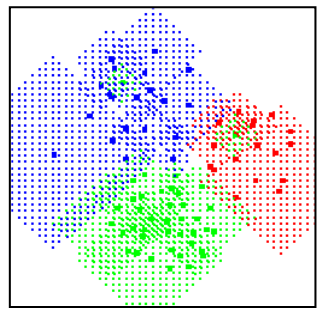
Each time a set (X,Y, C) is presented to the network, the neurons first verify if one of them already recognizes (X,Y) as belonging to category C. If this is the case, no action is taken. If, on the contrary, (X,Y,C) is not recognized by any neuron, a new one is created to store (X,Y) as a reference pattern and C as the category. In addition the new neuron inherits an influence field which defines its area of attraction or similarity domain.

A neuron is represented in the decision space by the point (X,Y) surrounded by a dotted area with a color associated to C. This area represents the influence field of the neuron and its shape is determined by the norm used to calculate the distance between a new point(X,Y) and the reference(Xi, Yi) stored in the neuron. For example, an Euclidian distance produces influence fields with the shape of circle, whereas a Manhattan distance produces influence fields with a diamond shape.

If the next example (X1,Y1) falls in the area of the neuron (X0,Y0), it is considered as belonging to category C0. If a teacher instructs the neural network that (X1,Y1) belongs to a category C1, the neuron storing (X0,Y0,C0) learns its lesson and shrinks its influence field such that it no longer recognizes (X1,Y1) as C0.



Graph 2



As examples are taught, the decision space gets modeled by more smaller neurons. It may feature neurons with portions of their influence fields overlapping with one another. Their intersection represents a zone of uncertainty.

The 77 examples plotted in Graph1 define a decision space which can be modeled by 21 neurons represented in Graph 2.

Advantages of the RBF classifier for recognition

The classification of a vector consists of evaluating if it lies within the influence field of one or more neurons modeling the decision space. The outcome can have three possible classification status: Identified with certainty, Identified with uncertainty, Unknown.

As a result the RBF classifier is very powerful since it allows managing uncertainty for a better, more refined diagnostics. It is also is especially suited for anomaly or novelty detection where the “unknown” classification is the one of importance.

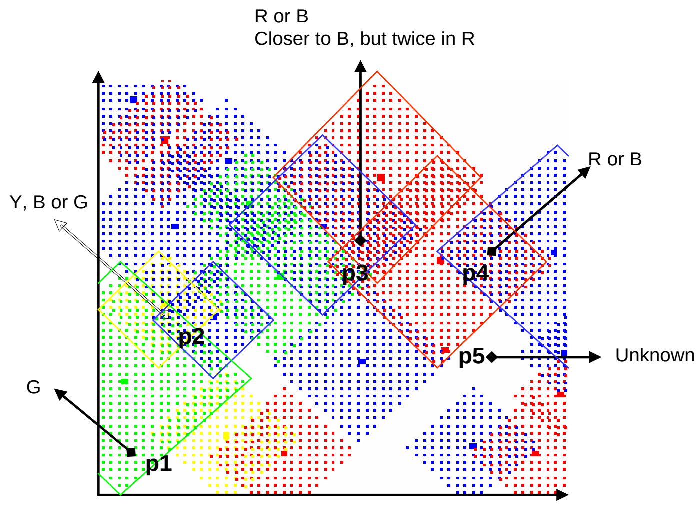
When a vector is broadcasted to the neural network, all the neurons calculate their distance between the input vector and the prototype stored in their memory. If the distance of a neuron is greater than its influence field, the neuron excludes itself from the list of neurons recognizing the vector. Otherwise it “fires” to indicate that it recognizes “somewhat” the vector. The similarity range is expressed with the distance value. Its dimension is a function of the data type stored in the vector and the norm in use to calculate the distance.

Several neurons can recognize the input vector. The one with the smallest distance value has a prototype in memory which is the closest to the input vector. Also, more than one neuron can fire with the same smallest distance. If they have identical categories, it reinforces the confidence level of the recognition. If they do not have the same category, they point a level of uncertainty in the recognition and potential ambiguities between certain categories. This uncertainty can be further considered by reading the categories recognized by the next firing neurons, that is with the next smallest distance value, etc.

The higher the distance, the lesser the similarity between the prototype and the input vector.

If a neuron has a distance equal to 0, it means that the input vector matches exactly its prototype.

In the example to the right, a 2D decision space is divided into four categories labeled with the colors R, G, B, and Y. It has been modeled with 19 neurons: 6 associated to the R category, 3 to the G category, 8 to the B category and 2 to the Y category. The influence fields of the models have variable size and some are overlapping one another.



The classification of the points P1 to P5 is interpreted depending on their positions inside neurons. P1 is identified without uncertainty as G. P2, P3 and P4 are identified with uncertainty. P5 is not recognized.

It is common that a decision space recognizes the majority of a population with a high accuracy after a minimal training and a few neurons, but performs poorly on the remaining population. Obviously the CogniMem non-linear classifier can learn many examples of the corner cases and tune the decision space to recognize them by generating smaller neurons delimiting the edges of the decision space (edges between categories, and edges with the background).

Three possible classification status

The recognition of a vector can have three outcomes or classification status:

Unknown: no neuron recognizes the input vector (ex p5)
Identified: one or several neurons recognize the vector and they agree with its category value (ex: p1)
Uncertain: several neurons recognize the vector and they are not all in agreement with its category value. (ex: p2, p3, p4)

Multi-level responses

When a recognition is uncertain, the user can take different actions depending on his application:

- Read the response of the neuron with the best match (i.e. smallest distance)
- Read the response of the N top firing neurons and apply some statistical or probabilistic rules to generate a single global response
- Discard the response but trigger the use of another network of neurons assigned to a different context and trained with a different feature which might be more relevant to waive the current existing uncertainty.

Best-match classification

The response of the firing neuron with the smallest distance value is equivalent to a best match. It can still be uncertain and in this case a detailed classification can be read out.

If a neuron fires with a distance 0, it means that the vector matches exactly the prototype stored in the neuron. The classification is an **exact match**.

Detailed classification

Examining the distance and category of all the firing neurons can be of interest to reinforce the accuracy of a decision, especially in the cases of uncertainty. Rules have to be established on a "per application" basis depending on the cost of a mistake, the requirements for a minimum throughput, minimum false negative, etc.

Let's take the example of a recognition where a vector is recognized by the following 6 firing neurons:

Neuron	1	2	3	4	5	6
Distance	5	12	13	15	38	39
Category	8	7	7	7	3	5

The best match is a neuron with a category 8 reporting a distance of 5 between its prototype and the input vector. However, the next three firing neurons with a higher distance report a category 7.

If the cost of an inaccurate recognition is low, the response of the 1^{st} neuron with category 8 is the simplest to retrieve (and very fast). On the contrary, if the application cannot afford a false-positive, it might be wiser to involve some statistics and assume that category 7 is the dominant category and should be the one selected for a final decision. More sophisticated rules can be deployed including the analysis of the histogram of the categories, and more. Some applications might even consider the generation of a "response" vector composed of all the "firing" categories (i.e. 8,7,7,7,3,5) and to be classified by another set of neurons taught to classify the "response" vectors. CogniMem can handle up to 127 subsets of neurons trained for different purposes. These subsets are called contexts.

Flexibility of the RBF model generator

- The mapping of the decision space can vary from conservative to moderate by using different values of the Maximum Influence Field. or by teaching counter-examples.
- The decision space can tuned by learning both examples and their counter examples
- The zones of uncertainty can be controlled by using different values of the Minimum Influence Field.

Non-linear model generator

One of the strengths of the CogniMem neurons is that they know when and how to adapt their influence field in order to comply with the teacher.

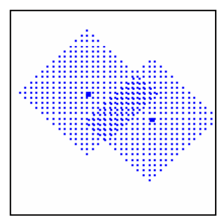
If a neuron containing a prototype of category A recognizes a vector V and V has to be learned as category B, the neuron automatically reduces its influence field to exclude V. All neurons firing with a category different from B do the same and a new neuron is committed to store V as a new prototype with a category B. The influence field of this new neuron is set to the smallest distance of all the firing neurons.

Learning new categories of vectors reduces the influence field of existing neurons and creates new smaller neurons. The category 0 has a special functionality in the sense that it reduces existing neurons, but does not create new ones. Learning a category 0 is equivalent to showing a counter-example, or a background example.

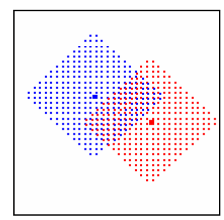
The following illustrations show how neurons adjust their influence field as new examples are taught.

If two examples are significantly different, two neurons are committed to store their (X,Y) vector and C category. Their influence fields are set to the default maximum value. They overlap slightly, thus defining a zone of redundancy for the category C_{blue}, or a zone of uncertainty between the categories C_{red} and C_{blue},

Same category

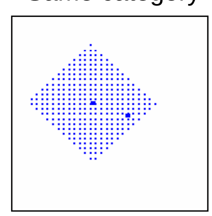


Different category

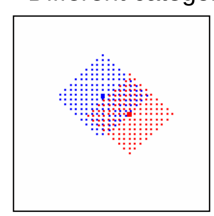


If two examples are close and with the same category, one neuron is sufficient to ensure their recognition. If they have different categories, two neurons are created, but their influence fields becomes equal to their distance minus 1, so the neurons mutually exclude the prototype of the other one.

Same category



Different category

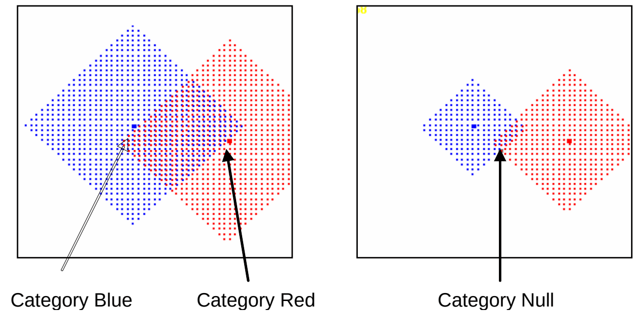


The tendency to create neurons with small or large influence fields can be controlled by teaching counter examples or by changing the Maximum Influence Field global register.

Modulating liberalism by teaching counter examples

The definition of conservatism is to prevent the neurons from over generalizing what they are taught. The easiest way to avoid this behavior is to teach limits or examples which the neurons should not associate to their model. This is the purpose of the category 0 or Null category.

Teaching a category 0 is only intended to force some neurons to reduce their influence field. It does not create any new neuron.

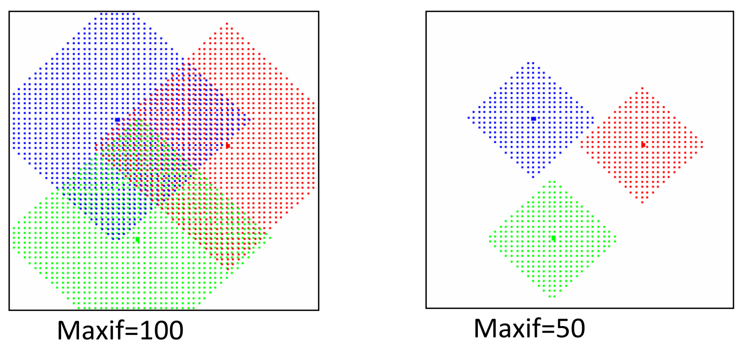


Modulating liberalism with the Maximum Influence Field

The Maximum Influence Field is the default value of a newly committed neuron when no other neuron recognizes the vector to learn

The higher the Maximum Influence Field, the larger the similarity domains of the neurons and the more liberal the recognition engine. The neurons with large influence fields have the tendency to over-generalize and recognize patterns with a high throughput. Liberalism is practical when the cost of an error is not critical and the number of neurons available is limited. If, on the other hand, an application has a very low tolerance to errors, it is best to set a small Maximum Influence Field or teaching many counter-examples. A conservative engine will require more teaching than a liberal engine because it has to understand the diversity and subtle differences between many examples.

The graphs to the right illustrate how the Maximum Influence Field can help fill more or less the decision space using the same set of examples.



Warning:
Changing the Maximum Influence Field once neurons are already committed can affect the consistency of the knowledge under construction.

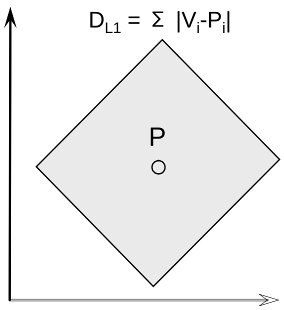
Shaping the decision space with a Norm

The Norm determines how to calculate the distance between the reference pattern or prototype stored in a neuron (P) and an input vector (V). If the calculated distance is less than the influence field of the neuron, the vector V is considered as similar to the prototype P.

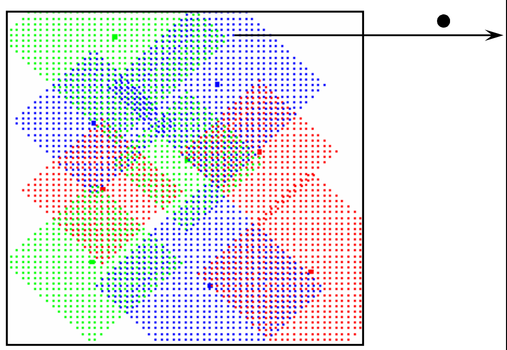
Norm L1 (Manhattan distance)

Distance= Sum |Vi - Pi |

The L1 distance emphasizes the drift of the sum of the all components between V and P.



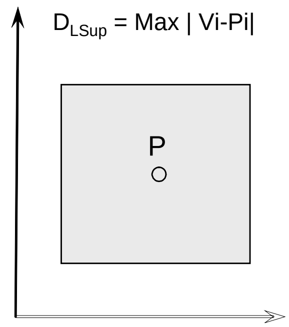
In a two dimentional space, the points located at a distance D_{L1} from P such that D_{L1}<C describe a diamond centered in P and with a side equal to √2 x C.



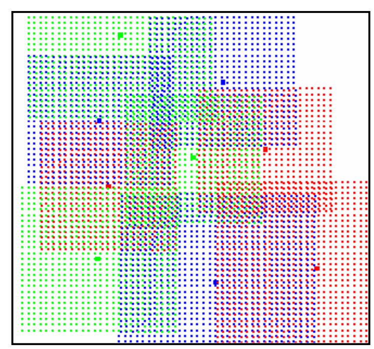
Norm L Sup

Distance = Max.|Vi-Pi |

The Lsup distance emphasizes the largest drift of the same component between V and P.



In a two dimentional space, the points located at a distance D_{Lsup} from P such that D_{Lsup}< C describe a square centered in P and with a side equal to 2xC.



Norm L2 (Euclidian distance)

The L2 Norm calculating Distance= √ Sum (Vi - Pi)^2 would give circular shapes to the neurons' influence field. However, the little precision improvement of the L2 distance compared to the L1 distance did not justify adding this processing-intensive calculation in the ASIC which would have significantly impacted its size.

Building zones of uncertainty

The existence of areas of uncertainty can be important to weigh a decision, especially if a mistake has a high cost. The belonging of a pattern to such an area can trigger the need for a thorough analysis in the current decision space and possibly require the usage of another decision space, if applicable, to give a second opinion.

Modulating uncertainty zones with the Minimum Influence Field

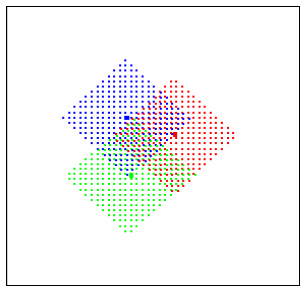
The Minimum Influence Field is the value below which the active influence field of the neurons cannot shrink.

If the influence field of a neuron becomes limited to this minimum value, it means that the prototype stored in the neuron lies close to the boundary of another category, and may be overlapped by another neuron of a different category. The neuron is tagged as degenerated. The higher the Minif, the larger the “Uncertainty” zones in the decision space.

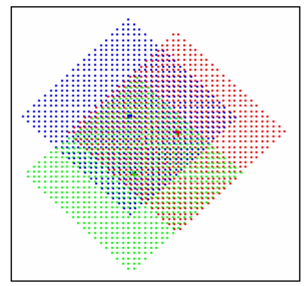
The graphs to the right show how the Minimum Influence Field can enforce zones of uncertainty.

The three neurons created when Minif=50 are all degenerated. They are not excluded from the influence fields of the other two neurons.

A Minimum Influence Field of 1 minimizes the zones of uncertainty.



Minif=10



Minif=50

The higher the Minimum Influence Field, the bigger the extent of “Uncertainty” zones in the decision space.

<u>Warning:</u>
Changing the Minimum Influence Field of the network once neurons are already committed can affect the consistency of the knowledge under construction.

Benefits of Iterative learning

The decision space is modeled as examples are taught and its shape depends on their sequence. This dependency is not desirable to build an accurate knowledge and it is recommended whenever possible to learn the examples repeatedly until the decision space is stable. This condition is established when no new neuron is committed between two iterations. Obviously the ability to execute an iterative learning requires that the examples be archived which is technically possible in most applications.

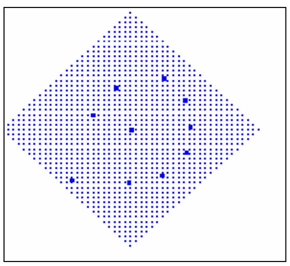
Incremental Decision Space mapping

If an example A is taught as category #1, the CogniMem neural network first determines if it recognizes the example. If only one category is recognized and it is category #1, the network does not take any action and simply discard the example A. If several categories are recognized, including the category #1, the network does not commit a new neuron to store the example, but instructs the neurons responding with a category other than #1 to shrink their influence fields.

With the learning of additional examples, the decision space will continue to change and it is possible that the example A might no longer fall inside the influence field of a neuron recognizing the category #1. If this is the case, teaching this example again will commit a new neuron with the category #1. This illustrates the fact that the order in which the examples are taught has a direct impact on the modeling of the decision space.

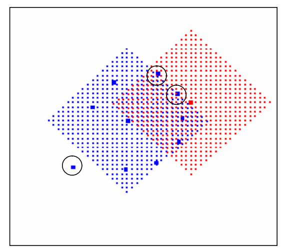
Sequence 1:

A single neuron recognizes the 10 initial examples of the "blue" category.



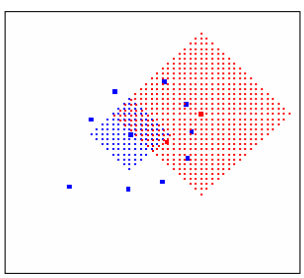
Sequence 2:

The first example of the "red" category causes the single neuron to shrink. Three of the former blue examples are no longer recognized.



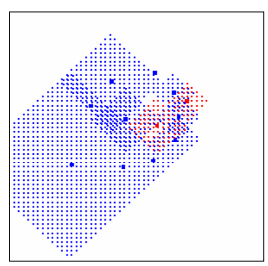
Sequence 3:

The second example of the "red" category causes the first neuron to shrink even more. The 9 original examples are no longer recognized.



Iterative learning:

Re-learning the 9 "excluded" examples will this time create 6 additional "blue" neurons with smaller influence fields.

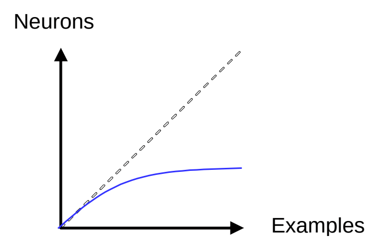


Whenever possible, it is recommended to record the examples so they can be learned in a repetitive loop until the knowledge gets steady and no more neurons are created or shrunken. The decision space is then the most accurate possible since all examples have been taken into consideration.

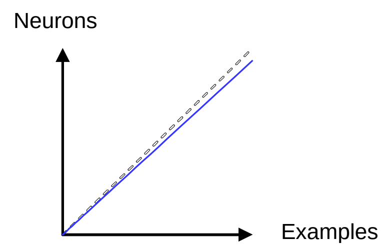
Learning curve and feature relevancy

The learning curve can be represented by the number of neurons created as new examples are taught.

If vectors learned by the neurons are significant for the intended classification, the neurons should generalize quickly and their number should reach a constant following a logarithmic curve



A linear learning curve with a slope close to 1 as shown to the right is a hint that the selected feature vector does not perform well since the neurons do not generalize.



If a feature is not performing as expected, studying the learning curves per categories of examples might be helpful to identify the categories which will be easy or hard to discriminate with the selected feature.

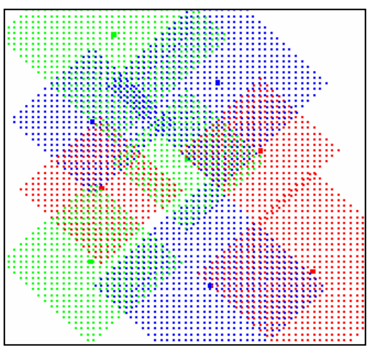
KNN (K-Nearest Neighbor) classifier

In KNN mode the notion of influence field is discarded and the network always returns a response which is the first closest match.

The following plots illustrate how a set of identical neurons can model different decision spaces whether they are used in RBF or KNN recognition mode.

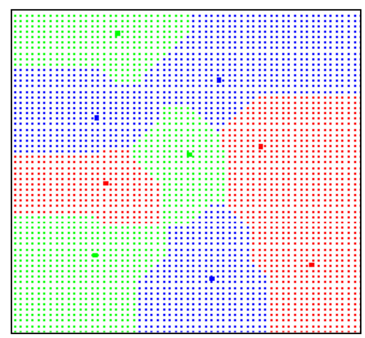
RBF

The space is mapped partially with certain zones being unclassified (i.e. black color). The zones with multiple colors are zones of uncertainty.



KNN

The entire space is mapped and with a single possible category (i.e. color code) per position.



- KNN always gives a response: Closest match (note that the shortest distance value can still be high)
- The zones of uncertainty are non existent

Warning: KNN should be turned off during a teaching phase or the knowledge will be limited to one neuron per category.

From decision space to knowledge base

Once a decision space has been modeled and validated by recognizing many samples, the contents of the neurons can represent a valuable knowledge base and intellectual property. This information includes the prototype stored in the neuron, its context, category and influence field. Also the knowledge file must be associated to the description of the features used to train the neurons for each context in use.

Bibliography

Darpa Neural Network Study October 1987-February 1988 by DARPA Neural Network Study (U.S.) (Author)

How We Learn; How We Remember: Toward an Understanding of Brain and Neural Systems : Selected Papers of Leon N. Cooper (World Scientific Series in 20th Century Physics, Vol 10) by Leon N. Cooper

Practical Approach to Pattern Classification by Bruce Batchelor (Nov 1, 1974)

Index

Best-match. *See* Classification
Category
  Background, 9
  Best match, 9
  Counter-example, 10
  Dominant, 9
  Null, 9, 10
Classification
  Best match, 8
  Detailed, 8
  Exact match, 8
  Status, 8
Confidence level, 7
Counter-example, 10
Decision space
  Conservatism, 10
  Distance Norm, 11
  Learning mechanism, 6
  Recognition mechanism, 7
  Uncertainty areas, 12
Degenerated, 13
Distance, 7
Exact-match. *See* Classification
Feature vector, 5
Firing neuron, 7
Influence Field
  Active, 9
  Maximum, 10
  Minimum, 12
Learning
  Influence fields, 11
  Iterative, 13
  Mechanism, 6
Liberalism, 11
Maximum Influence Field, 9, 10
Minimum Influence Field, 9, 12
Norm, 11
  L Sup, 12
  L1, 11
Recognition
  Classification status. *See* Classification
  Mechanism, 7
Signature vector, 5
Uncertainty, 7