WARSAW UNIVERSITY OF TECHNOLOGY
FACULTY OF MATHEMATICS
AND INFORMATION SCIENCE

Neural Networks

Lecture 8

Model ART

Adaptive Resonance Theory

The problem of stability is very important in a self-organizing systems. Most of neural networks models faces the problem known as the *stability-plasticity dilemma*.
Stability means that already learned templates will not be affected by the new data.

Adaptive Resonance Theory

The network unable to learn new information on the top of old is the great problem.
In a multilayer perceptron trying to add a new template to an already trained network may destroy all previous learning by interfering with weight values.

Adaptive Resonance Theory

Decreasing the learning coefficient (i.e. influence of successive input signals) the already existing classes can be locked up. But such a mechanism yields lack of plasticity – ability to react for the new data.

Adaptive Resonance Theory

Stability **-** vs- **plasticity** it was Grossbergs' dilemma, how both can be achieved simultaneously?

Stability problem is connected with the number of output elements (the number of classes). If this number is fixed there is no possibility to create the new classes for new templates.

Adaptive Resonance Theory

When the number of classes is not limited it yields to the very fine resolution of the space of input signals up to the limit

1 template = 1 class

One of possible solution is the possibility to add new output elements to create the new stable class (category).

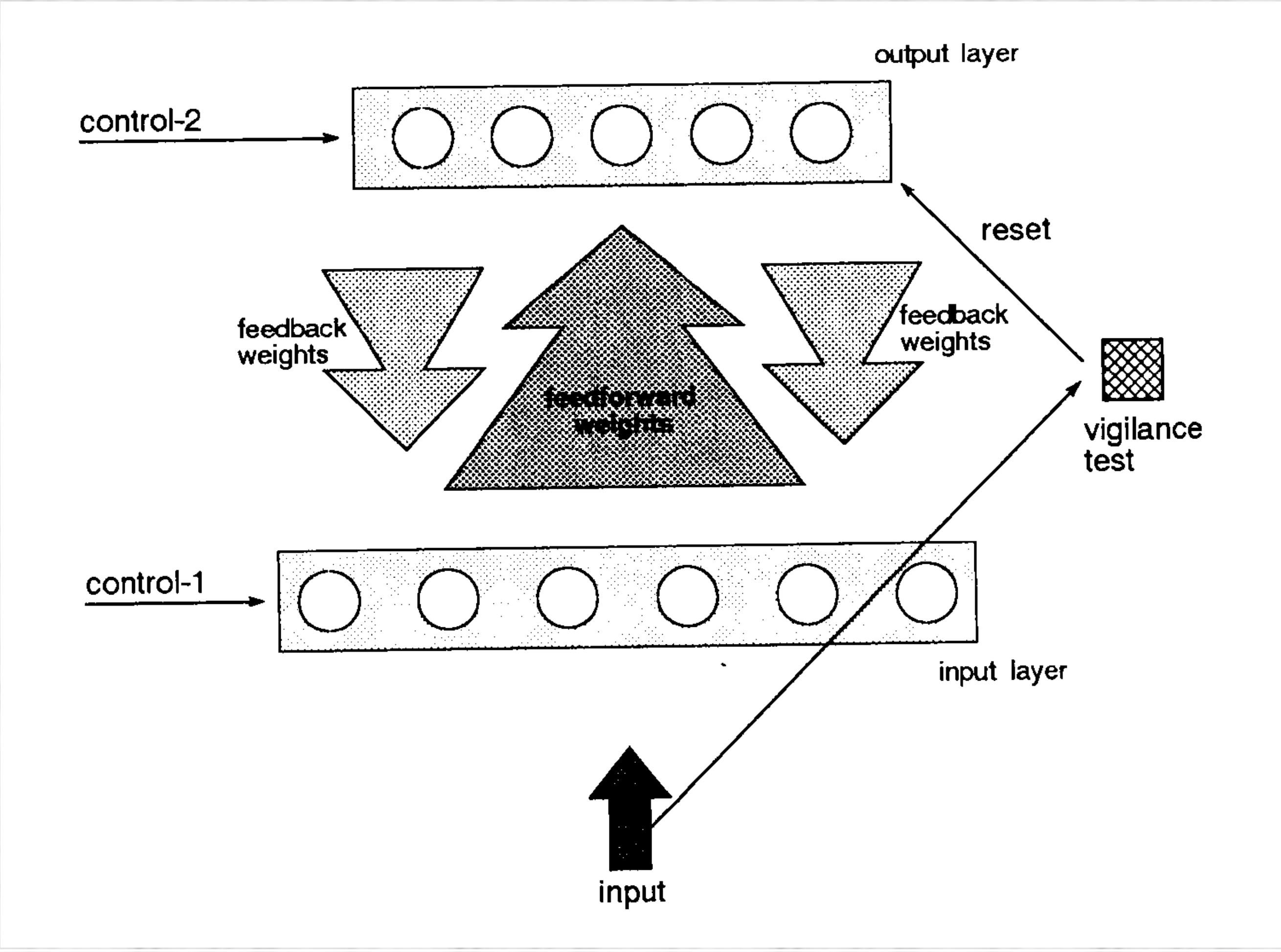
Adaptive Resonance Theory

The major feature of ART model, proposed by Gail Carpenter and Stephen Grossberg from MIT is the ability to switch modes between plastic (the learning state where the internal parameters can be modified) and stable (a fixed classification set), without detriment to any previous learning.

Model A R T – Authors

- Stephen Grossberg – professor of mathematics, psychology and biomedical engineering
- Gail Carpenter – professor of mathematics

Model A R T – network description



Adaptive Resonance Theory

The concept of the network **ART** –

Adaptive **R**esonance **T**heory –

The model main idea: *two signals - input and memorized template are in a resonance if they are sufficiently similar.*

If the input signal is not similar to any of memorized templates (it can not be included into existing classes) – the new class, new category is formed – and its template is this signal. The new output element is added.

Adaptive Resonance Theory

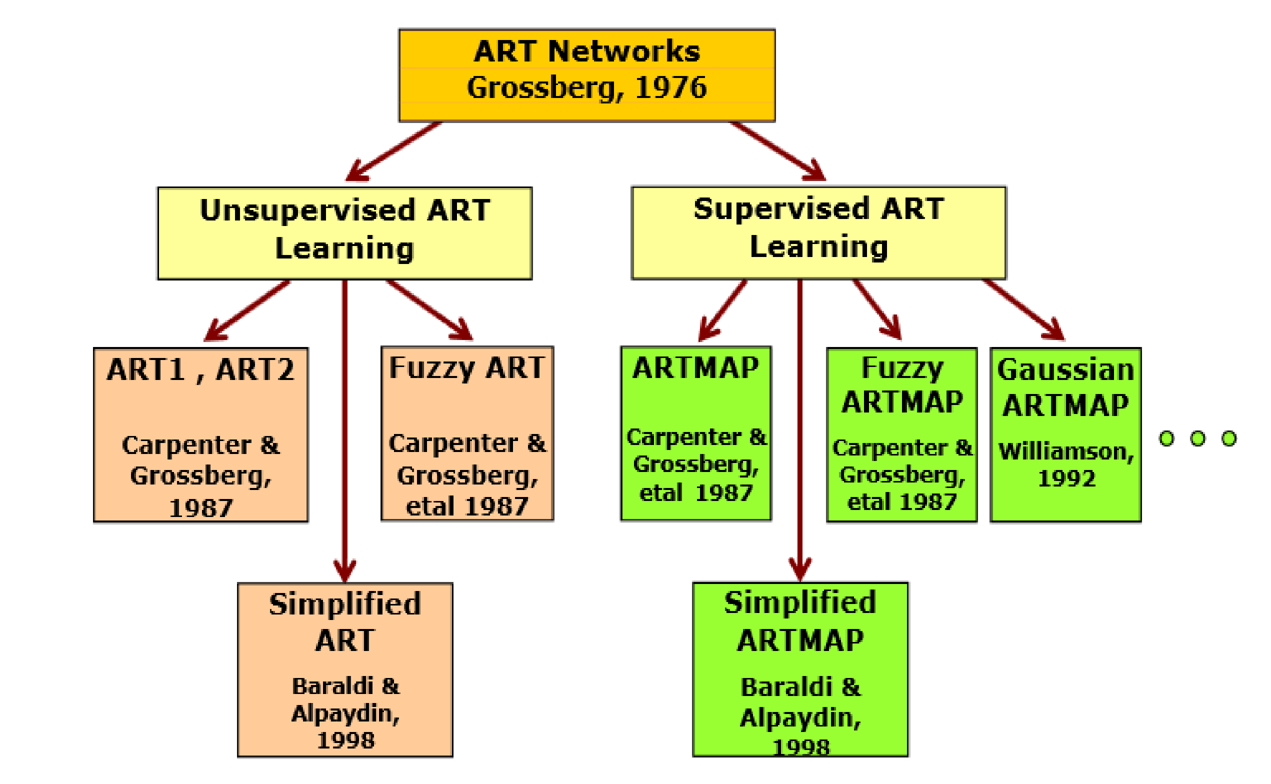
When an input learning vector is presented:

- If the network has learned it previously, a resonant state is achieved
- quickly.
- Otherwise, it searches to the stored patterns for a match.
- If no match is found, a new pattern is stored.
- The previously learned patterns remains without change

Adaptive Resonance Theory

The similarity depends of the *vigilance threshold* $\rho \in$ <0;1>.
This parameter controls the resolution of the classification process. A low threshold (<0.4) will produce a low resolution creating fewer class types. A high vigilance threshold (tending to 1) will produce a very fine resolution classification, meaning that even slight variations between input patterns will force a new class to be made.

Important ART Networks



Model A R T – network description

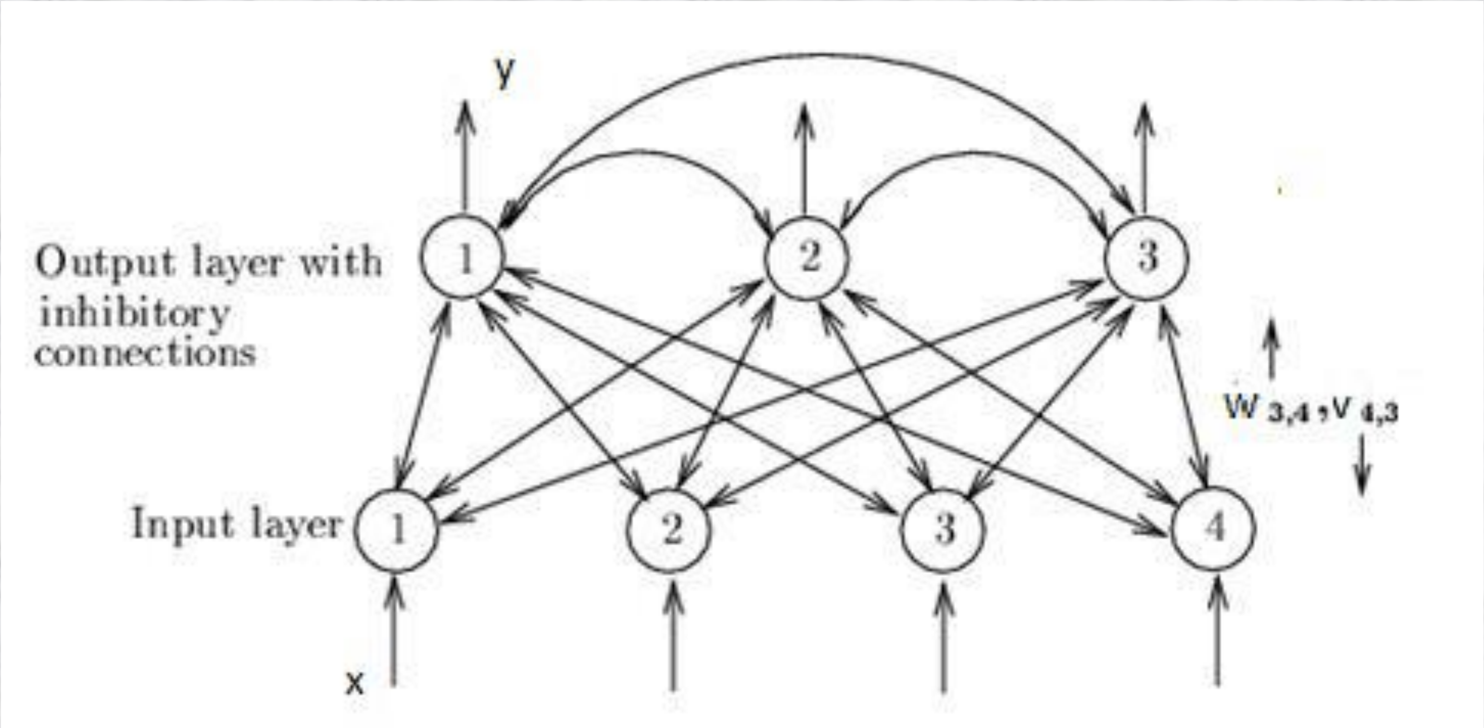
The ART network has two layers:

The first is the *input–comparison* layer,

The second is the *output–recognition (category)* layer.

These layers are connected together with *feedforward connections* from input layer to the output layer, *feedback connections* – from the output layer to the input layer. There are also connections between the nodes of the output layer as *lateral inhibition*.

Model A R T – network description



Input

Model A R T – network description

The ART network has ***feedforward*** weight matrix from the input layer to the output layer and ***feedback*** weight matrix from the output to the input layer. These paths will be marked ***W*** and ***B*** respectively.

For each layer there are also logic control signals that control the data flow through layers at each stage of network operation: ***control-1*** and ***control-2***.

Model A R T – network description

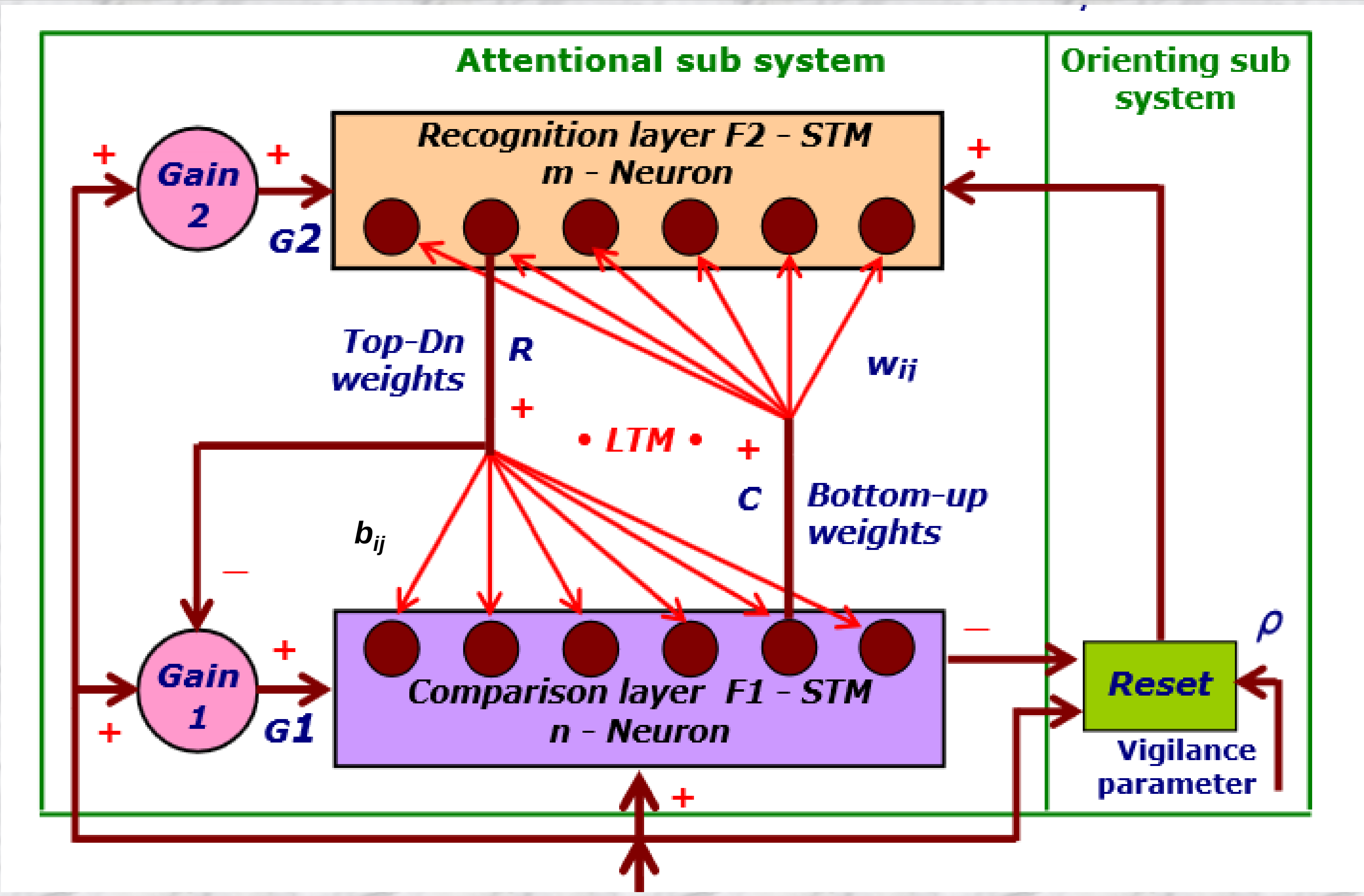
Control-1 determines the course of data flow for the input layer – its' binary value toggles the first layer of nodes between two modes: *input* and *comparison*.

The state of control-1 is *one* whenever a valid input is presented to the network but is forced to *zero* if any node in the recognition (output) layer is active.

Model A R T – network description

The binary value of ***Control-2*** enables or disables the nodes in the recognition (output) layer. It is *one* for any valid input pattern but *zero* after a failed vigilance test (this disables the recognition layer nodes and resets their activation levels to zero).

Model A R T – network description



Model A R T – network description

Between the input and output layers there is also a ***reset circuit***. It performs a reset function for output nodes – it is actually responsible for comparing the input signals to a vigilance threshold that determines whether a new class pattern should be created for an input pattern.

ART1 – the phases

There are several phases to learning or classification in an ART network.

The main paradigm is that continually modified input vector is passed forward and backward between the layers in a cyclic process.

The network action and activity of separate layers can be described.

The phases: ***an initialization phase, a recognition phase, a comparison phase*** and ***a search phase.***

ART1

1. The inicjalization phase.

The weight vectors ***W*** and ***B*** must be initialized.

The feedback links (***B***) are all set to one, so every node in the output layer is initially connected to every node in the input layer.

The feedforward links (***W***) are set to constant value, for example

$$w_i = \frac{1}{1+n}$$

where n is the number of input nodes.

The vigilance threshold is set in a range $0 < \rho < 1$.

ART1

2. The recognition phase.

In this phase the input vector is passed through the network and its value is matched against the classification represented at each node in the output layer. The neuron with largest input has weights that best match the input vector. It wins the competition.

ART1

Each weight vector ***W*** at each recognition node (output) is a „*stored template*”, or exemplar class.

The input vector is compared to the exemplar at each node and the best match is found. The best match comparison is done by the dot product of the input vector and a node’s weight vector. The node with closest weight vector will yield the largest result.

(of course several nodes in the output layer may have high level of activation).

ART1

The lateral inhibition between the nodes will turn off each node except the maximum response node – only this node remain active in the output layer. This node passes its stored pattern ***B*** (the class exemplar) back to the comparison (input) layer.

ART1

3. The comparison phase

Two vectors are present at the input layer where each node has three inputs. A comparison between the input vector ***X*** and the ***B*** is done and if degree of similarity is less then vigilance parameter the network causes reset. The effect of the reset is to force the output of firing neuron in the recognition layer to zero, disabling it for the duration of current classification.

ART1

4. The search phase

S<ρ means that the node was the best match, but the classification was wrong.

If there is no reset signal generated, the match is adequate and the classification is finished. Otherwise, the other stored pattern must be research to seek a correct match. This process repeats, until one of the two events occurs.

If no such node is found the network declare the input vector an unknown class and allocate it to a unused node in the output layer.

ART1 Algorithm Description

1 Step. Initialize

$b_{ij}(0) = 1$ $\qquad$ $w_{ij}(0) = 1/(1+n)$

$0 \le i \le n-1,\ 0 \le j \le M-1$

Set ρ where $0 \le \rho \le 1$

where $t_{ij}(\tau)$ is the top-down and $w_{ij}(\tau)$ is the bottom-up connection weight between node i and node j at time τ. ρ is the vigilance threshold which determines how close an input has to be to correctly match a stored exemplar; M number of output nodes, n number of input nodes.

all output elements are enable.

ART1 Algorithm Description

Step 2. Apply new input

Step 3. Computing matching

$$\mu_j = \sum_{i=1}^{N} w_{ij}(\tau)\, x_i$$

i = 0,1,2,...M-1, j = 0,1,2,..., N-1

μ_j is the output node j and x_i is element of the input which can be either 0 or 1.

Step 4. Select best matching element

$$\mu_{j*} = \max_j[\mu_j]$$

ART1 Algorithm Description

Step 5. Test

$$\|X\| = \sum_i x_i$$

$$\|\mathbf{BX}\| = \sum_{\mathbf{i}} \mathbf{b}_{\mathbf{k^*i}}(\tau)\mathbf{x}_{\mathbf{i}}$$

i = 0,1,2,...,N-1

$$\frac{\|\mathbf{BX}\|}{\|\mathbf{X}\|} > \rho \qquad \begin{cases} \textbf{yes} \rightarrow \textbf{go to Step 7} \\ \textbf{no} \rightarrow \textbf{go to Step 6} \end{cases}$$

ART1 Algorithm Description

Step 6.

Disable best match

Set output of best match node to 0. Go to Step 3

$$\mathbf{b}_{\mathbf{j^*i}}(\tau+\mathbf{1}) = \mathbf{b}_{\mathbf{j^*i}}(\tau)\mathbf{x}_{\mathbf{i}}$$

Step 7.

Adapt best match

$$\mathbf{w}_{\mathbf{ij^*}} = \mathbf{b}_{\mathbf{j^*i}}(\tau)/(\mathbf{0.5} + \sum_{\mathbf{i}} \mathbf{b}_{\mathbf{j^*i}}(\tau)\mathbf{x}_{\mathbf{i}})$$

Step 8. Repeat. Enable and disabled nodes, them go to Step **2**

Model A R T

The procedure can be stopped if:

1.The stored exemplar correctly match with the input signal **X,** then the signal **X** is included into this class and its exemplar (described by the vector **W**) is modified (if necessary)

2. There is no possibility to include input signal into existing class, then the 1st unused element creates a new class with **W = X**;.

Model A R T

3. If there is no possibility to include input signal into existing class and there is no more unused elements the input signal is not taken into account (omitted).

Model A R T

This algorithm solves both problems of **stability** and **plasticity**. It kept plasticity as long so all reserve nodes are not used, and stability – because the procedure of changes in the net structure (changes of weights $\mathbf{W}^k$ values) can be performed only a limited number of times defined by the set of input signals.

It is the result of the algorithm where in the Step 7, we can only remove bits (i.e. put some weights to zero), but the bits can not be added. The stored exemplars can not be reconstructed in a cyclic way.

Model A R T

The loop Step 3 → ... → Step 6 → Step 3 is searching through the set of stored exemplars:

• computing matching

• selecting the best matching, next second in line etc. according to the criterion

maximum value of **$B^k * X$** ***until the condition***

$$\frac{\|\mathbf{BX}\|}{\|\mathbf{X}\|} > \rho$$

is fulfilled

Model A R T – network description

The network is operating automatically. It does not need external signals and is able to cope even with the infinite stream of input data.

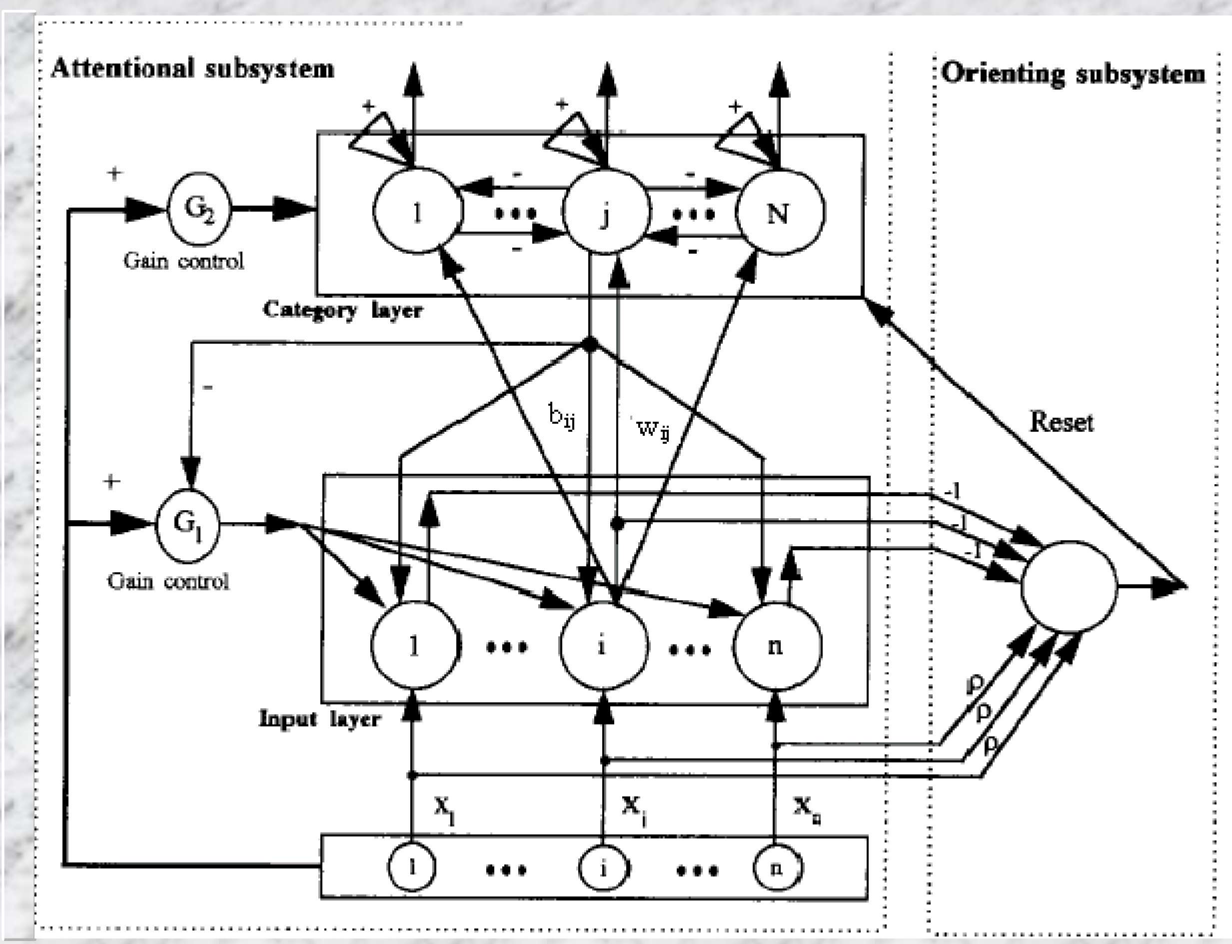
Basic features:

- fast access to classes (categories),
- possibility to create a new classes if necessary
- rejection of unknown inputs after exceeding the network capacity

Model A R T – network description

- can be used to memorize the binary patterns
- the self-organizing and self stabilizing network
- learn with the assumed accuracy
- can learn 2^n different patterns, where n – pattern dimension

Model A R T – network description



where:

n – dimension of an input

N – number of category

w_{ji} – weights bottom – up

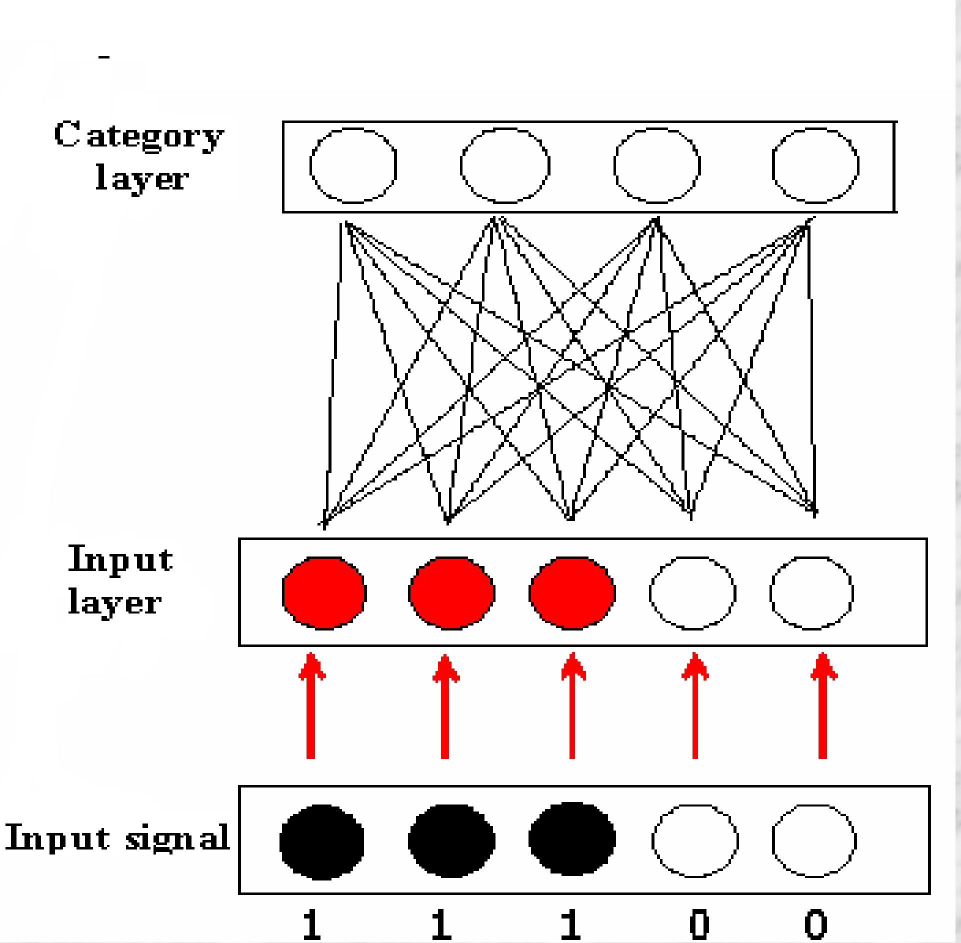
b_{ji} - weights top-down

ρ - vigilance threshold

Model A R T – network operation

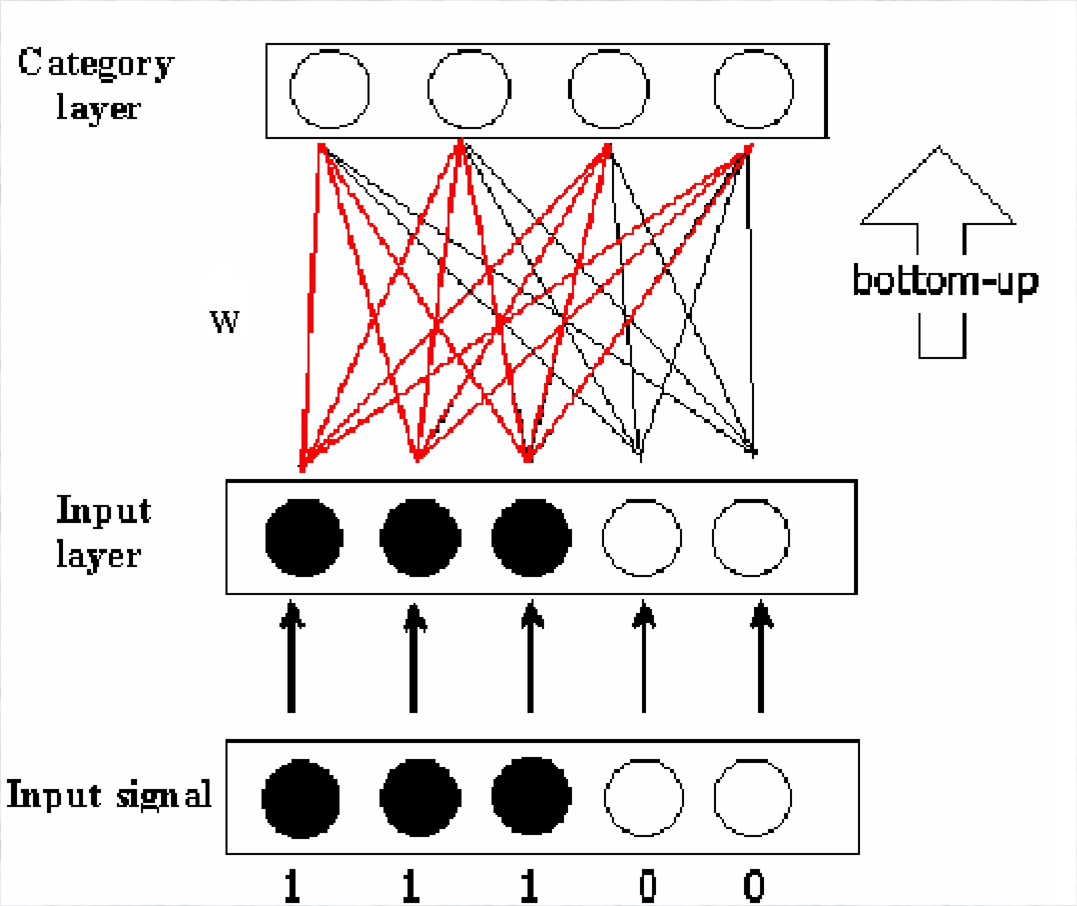
- Input signal

  X=[1,1,1,0,0]



Model A R T – network operation

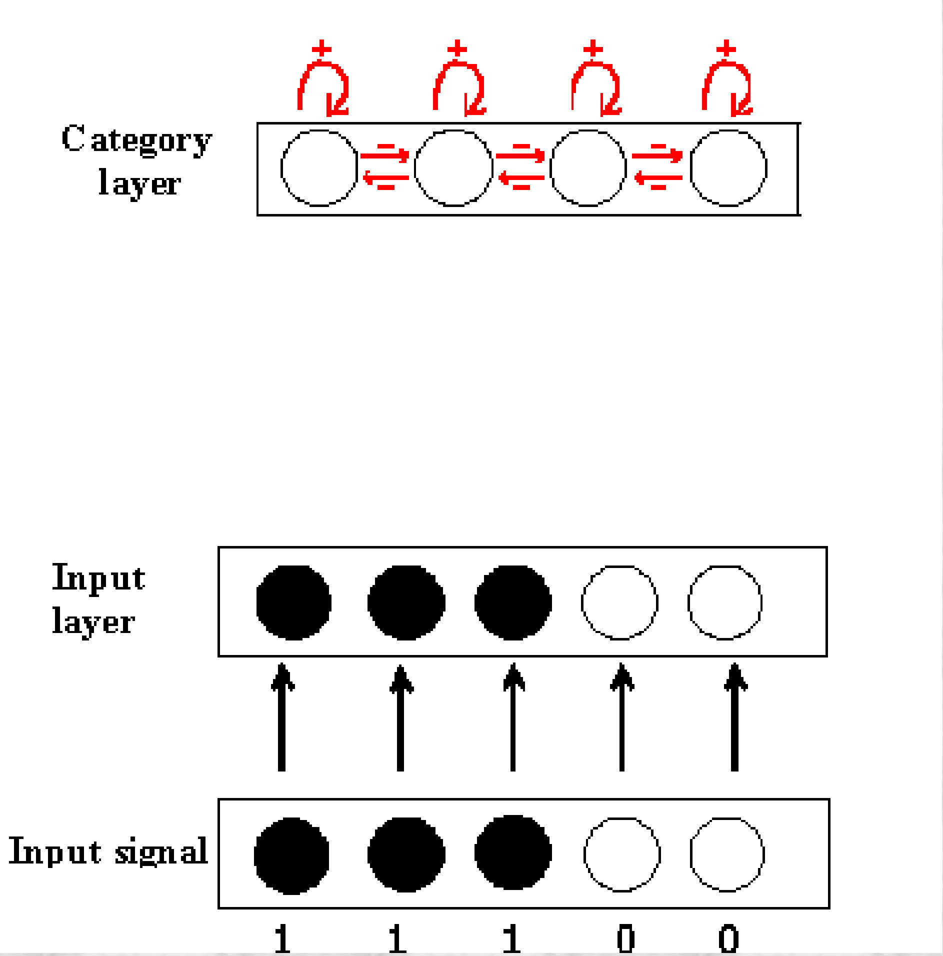
- Signal goes through weight matrix W



Model A R T – network operation

- Selection of best matching category

$$J = \arg_j \max_{j=1\ldots N} u_j$$



Model A R T – network operation

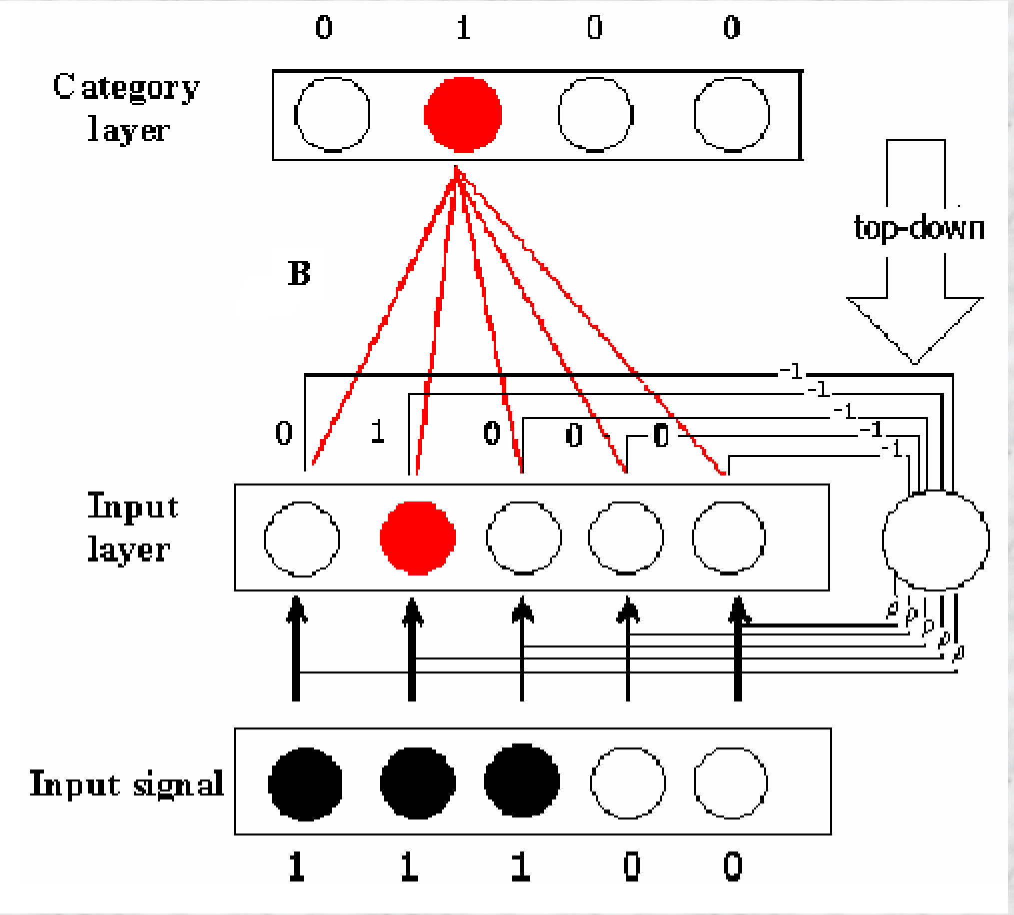
- Comparison of input signal and category template, if

$$\frac{|B_J \cap X|}{|X|} = \frac{\sum_{i=1}^{n} b_{Ji} x_i}{\sum_{i=1}^{n} x_i} \le \rho$$

reset category

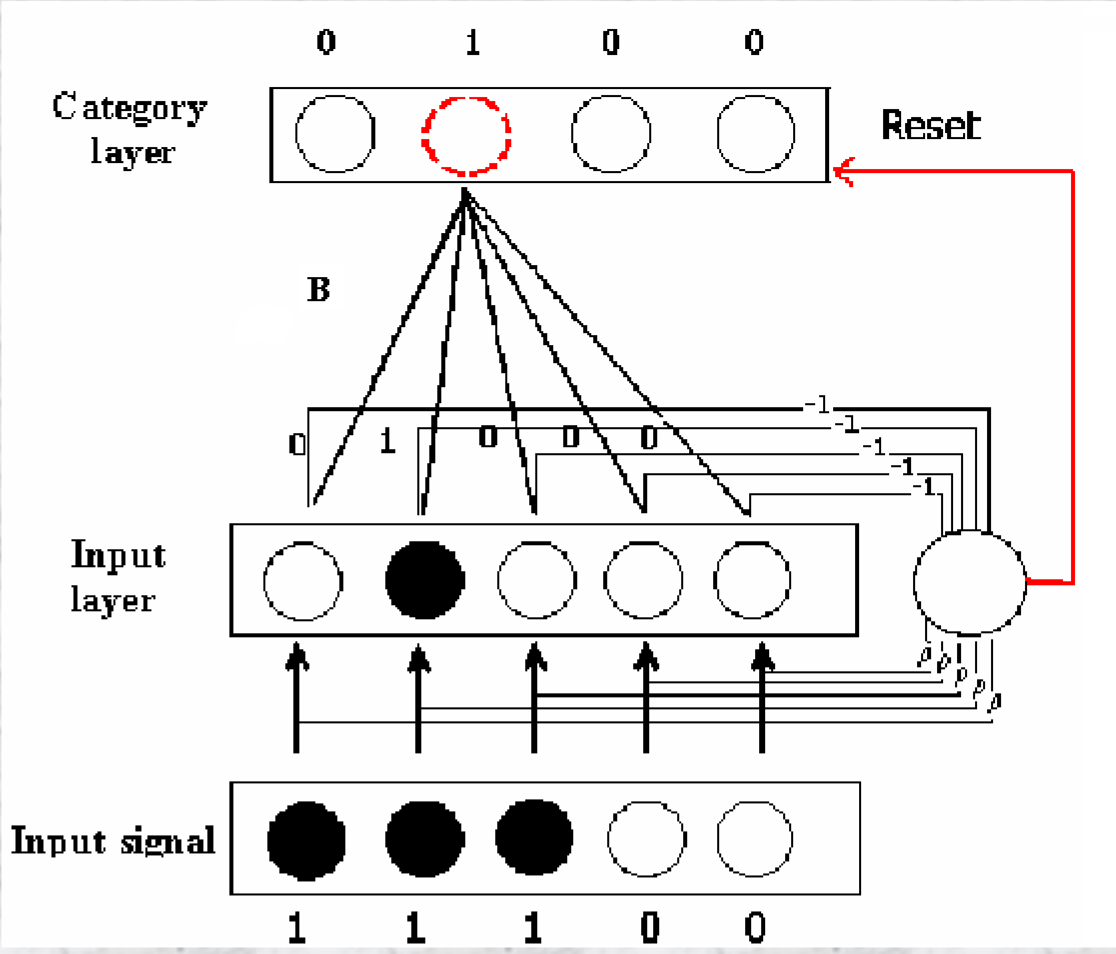
otherwise category was correctly recognize

(resonance), weight correction



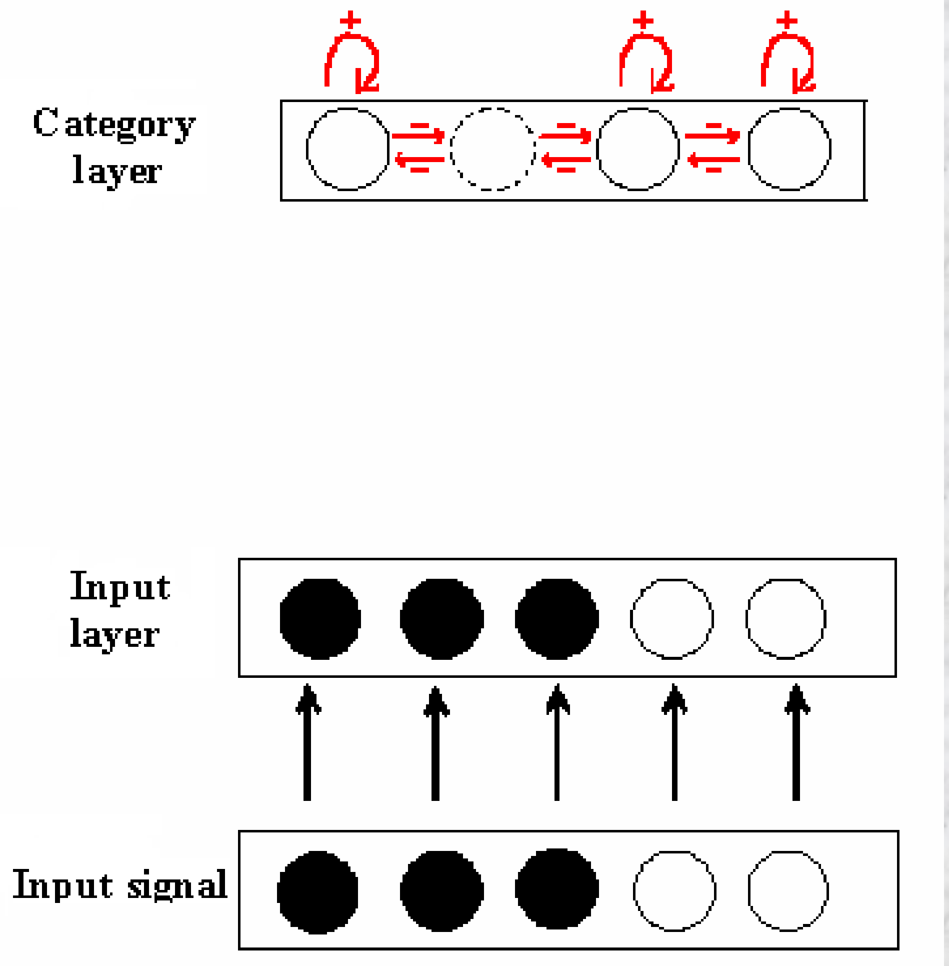
Model A R T – network operation

Reset category



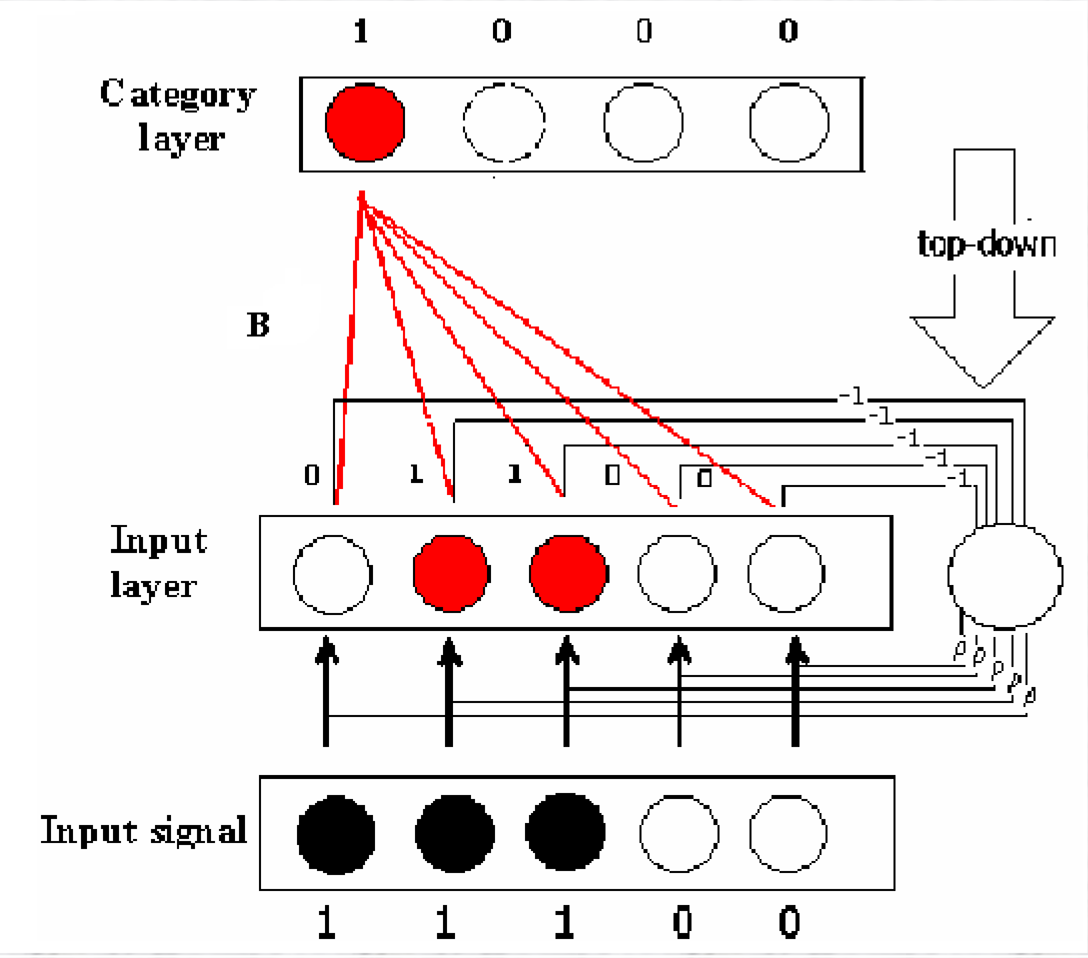
Model A R T – network operation

Selection of best category



Model A R T – network operation

Comparison of input signal and category template



Connection weights

Correction of the weight values, matrices ***B*** and ***W***

$$b_{ji}^{(new)} = \begin{cases} \dfrac{w_{ji}^{(old)} x_i}{\alpha + \sum_{i=1}^{n} w_{ji}^{(old)} x_i} & j = J \\ b_{ji}^{(old)} & j \neq J \end{cases}$$

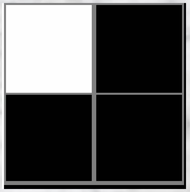
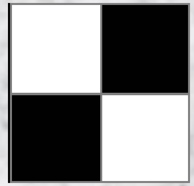
$$w_{ji}^{(new)} = \begin{cases} w_{ji}^{(old)} x_i & j = J \\ w_{ji}^{(old)} & j \neq J \end{cases}$$

Learning precision

- Learning accuracy depends of the parameter ρ; $0 \leq \rho \leq 1$
- Quality of categories depends if ρ
  - wide categories for small ρ values
  - precise (narrow) categories for ρ close to 1

ART1 modifications

Input signals x_1 = [0,1,1,1], x_2 = [0,1,1,0]

if shown in this sequence will be classified into one category, because at the beginning B_o=[0,1,1,1], and next

$$\frac{|B_0 \cap X_2|}{|X_2|} = \frac{\sum_{i=1}^{n} b_{0i} x_{2i}}{\sum_{i=1}^{n} x_{2i}} = \frac{2}{2} = 1$$

we get the exemplar B_o=[0,1,1,0],

ART1 modifications

To avoid such problems the inputs can be extended adding the complementary vectors

$x_1 = [0,1,1,1,|\ 1,0,0,0\ ]$, $x_2 = [0,1,1,0,|\ 1,0,0,1]$,

then

$$B_o=[0,1,1,1,1,0,0,0]$$

$$|B_0 \cap X_2| = 3$$

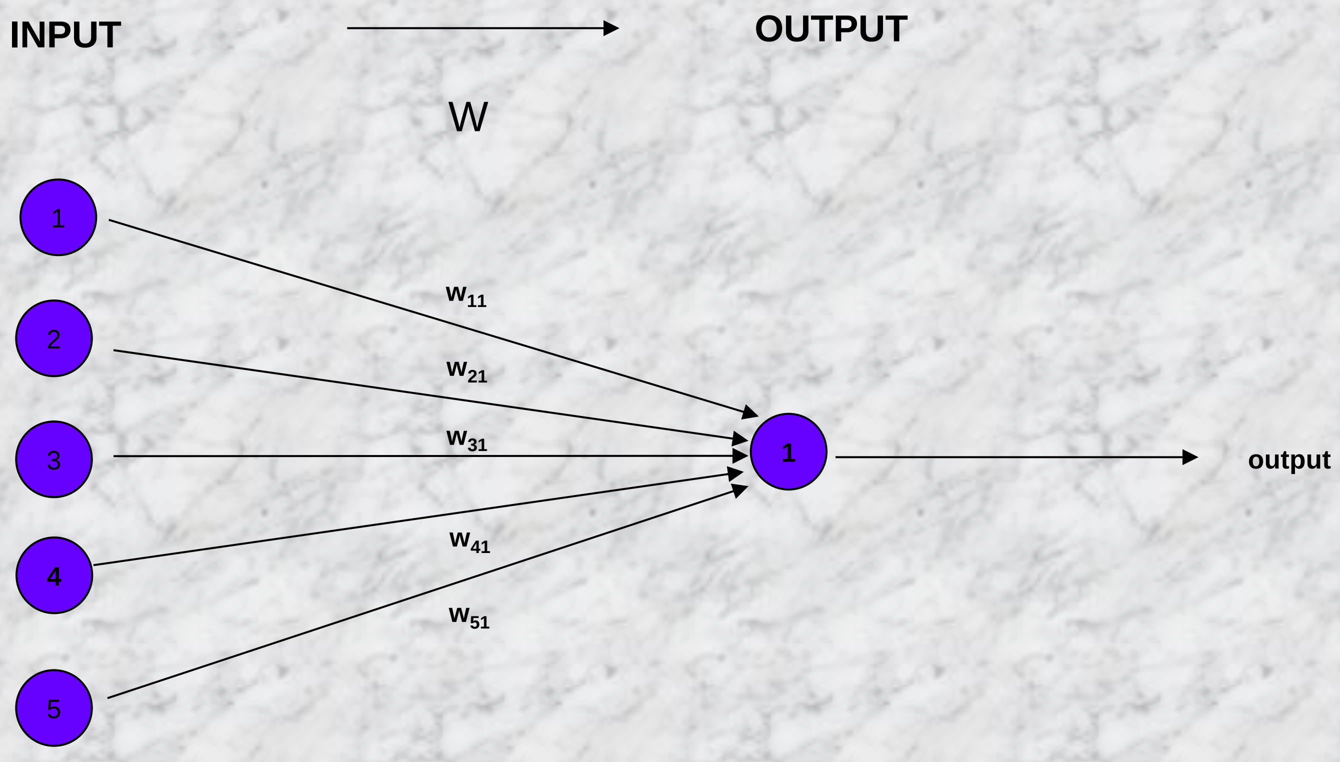
and next

hence

$$\frac{|B_0 \cap X_2|}{|X_2|} = \frac{\sum_{i=1}^{n} b_{0i} x_{2i}}{\sum_{i=1}^{n} x_{2i}} = \frac{3}{4}$$

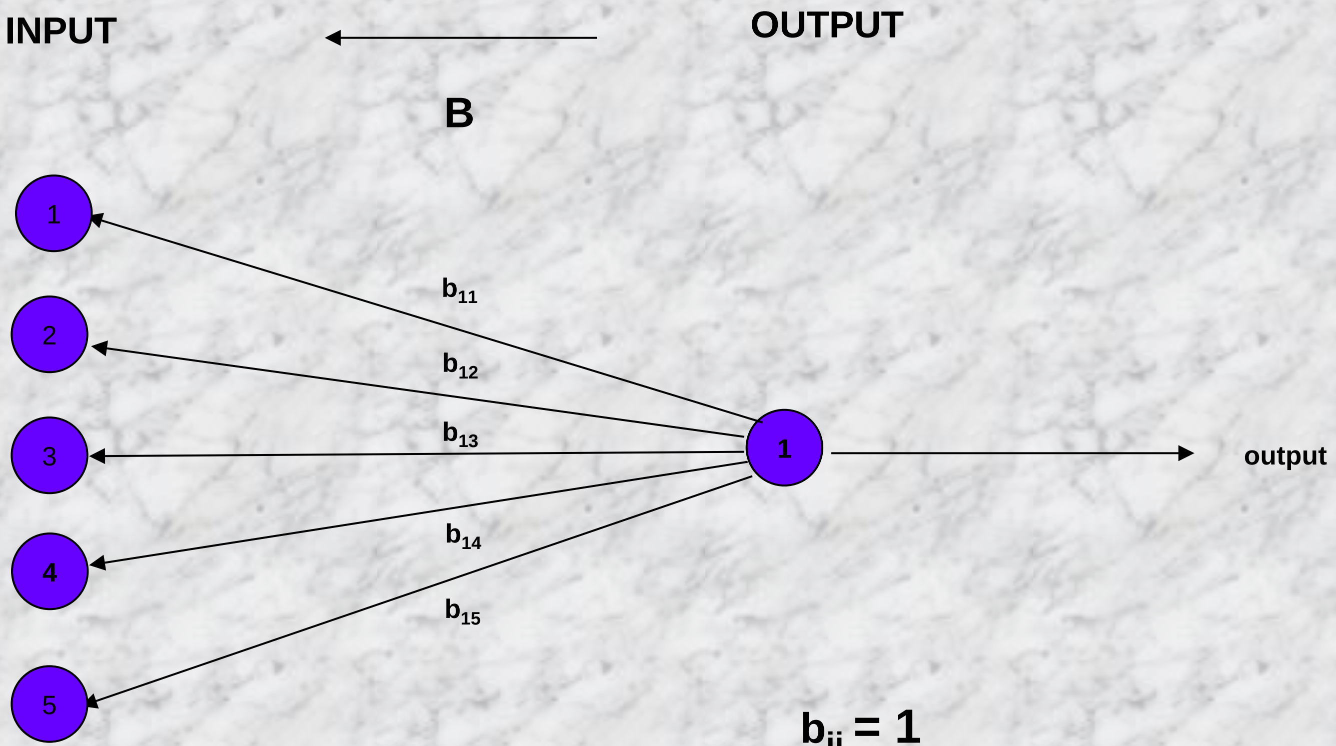
ART Example

Initial structure



ART Example

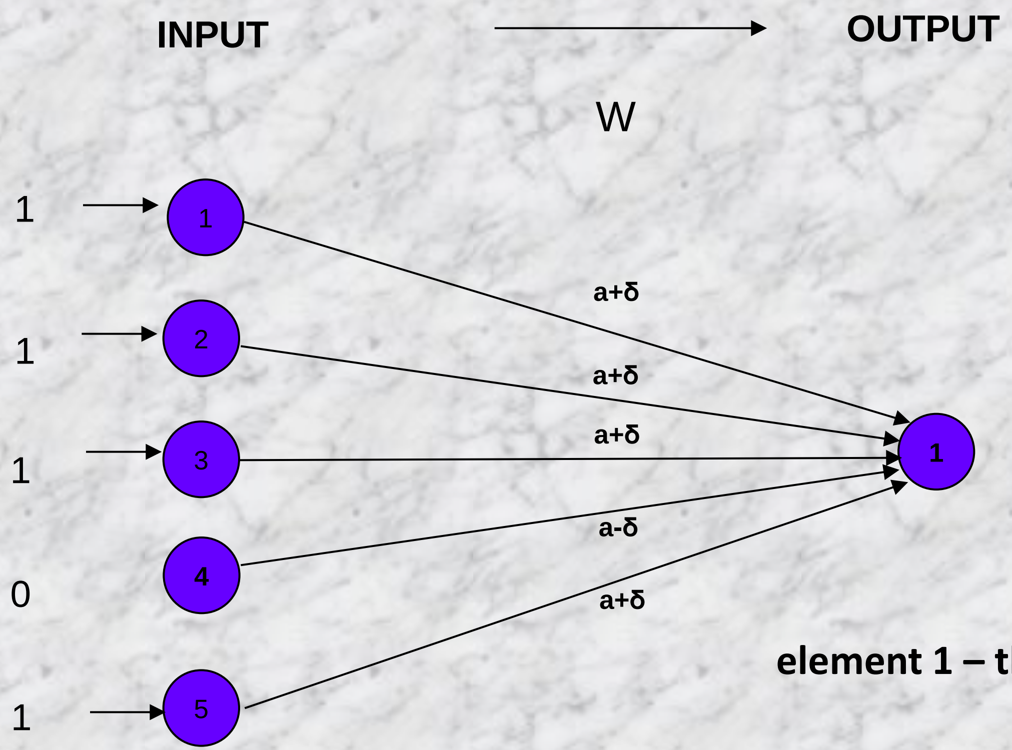
Initial structure



$b_{ij} = 1$

ART Example

1^{st} input signal X_1=[1 1 1 0 1]

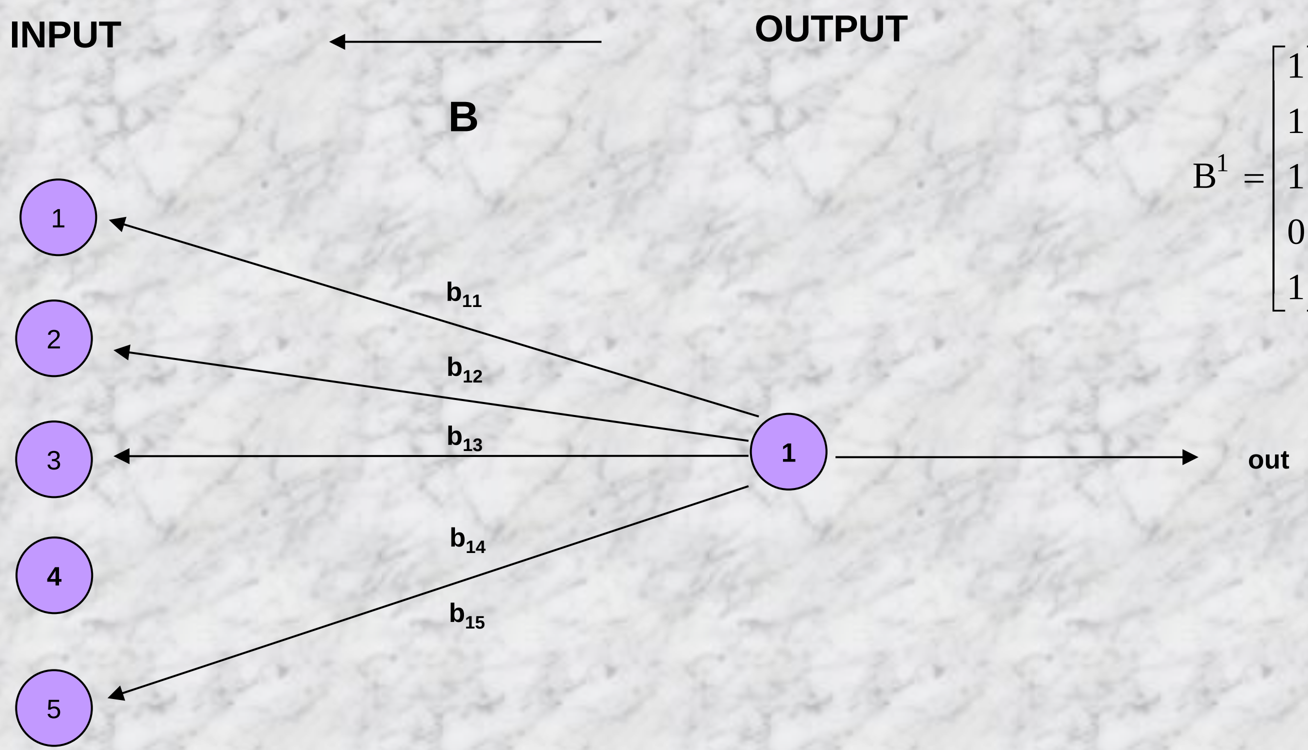


$$W^1 = \begin{bmatrix} a+\delta \\ a+\delta \\ a+\delta \\ a-\delta \\ a+\delta \end{bmatrix}$$

element 1 – the winner

ART Example

1st input signal

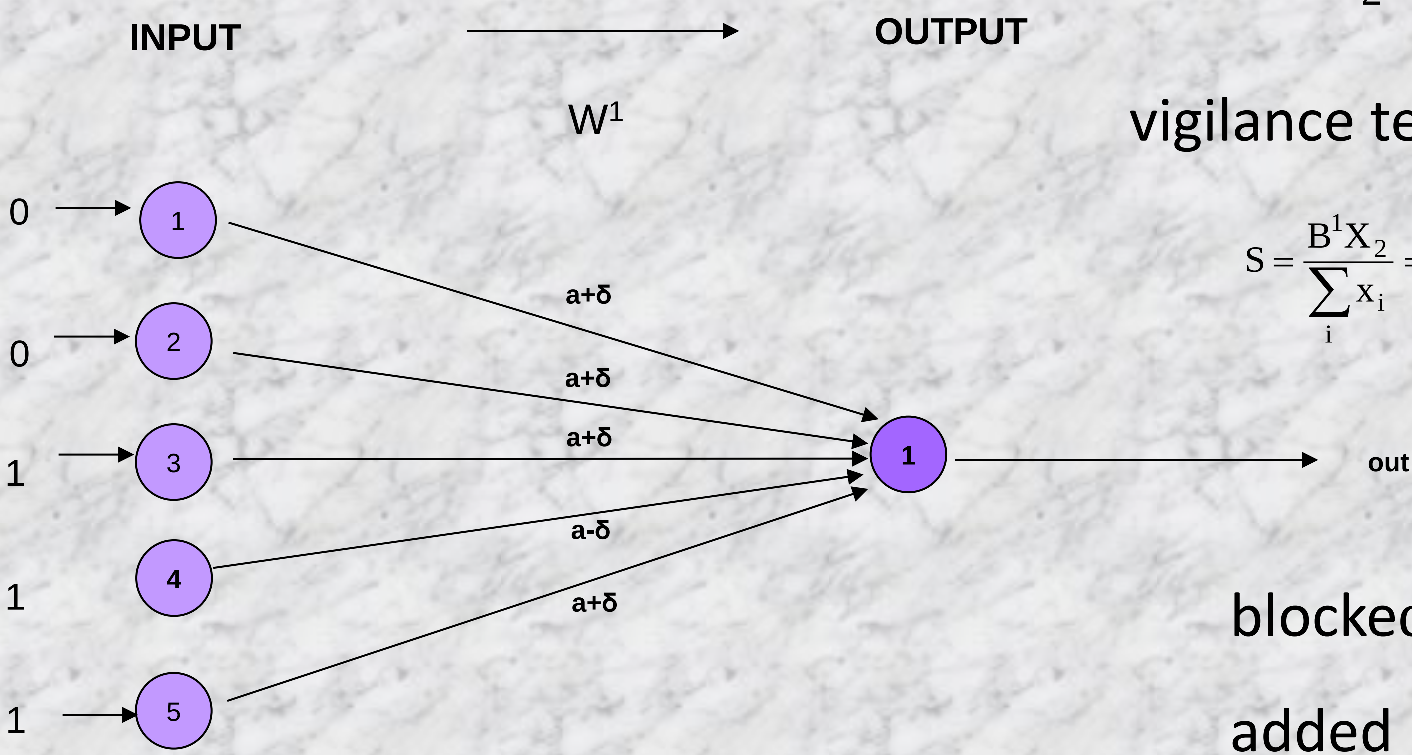


$$B^1 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 1 \end{bmatrix}$$

ART Example

2^{nd} input signal X_2=[0 0 1 1 1]

$W^1X_2 = 3a+\delta$



vigilance test

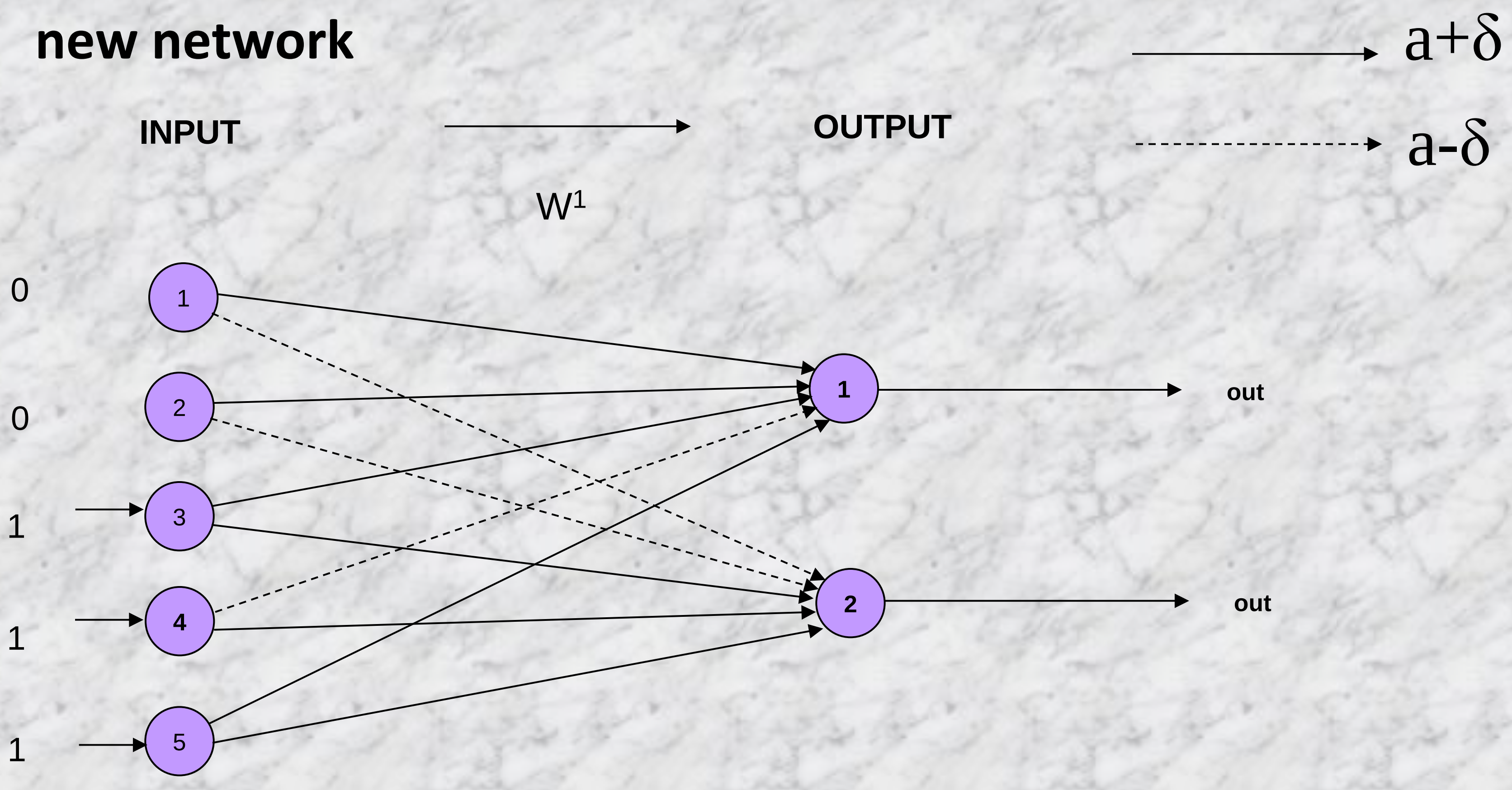
$$S = \frac{B^1 X_2}{\sum_i x_i} = \frac{2}{3} < \rho = 0.9$$

blocked 1

added 2

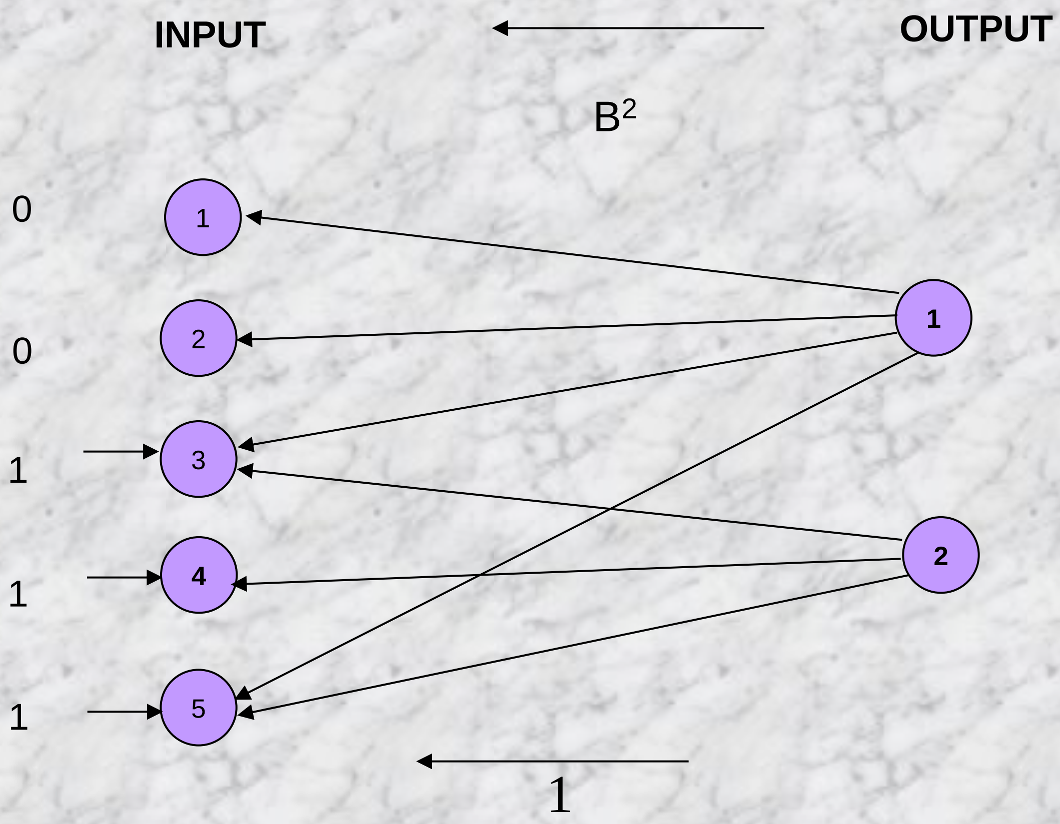
ART Example

new network



ART Example

new network

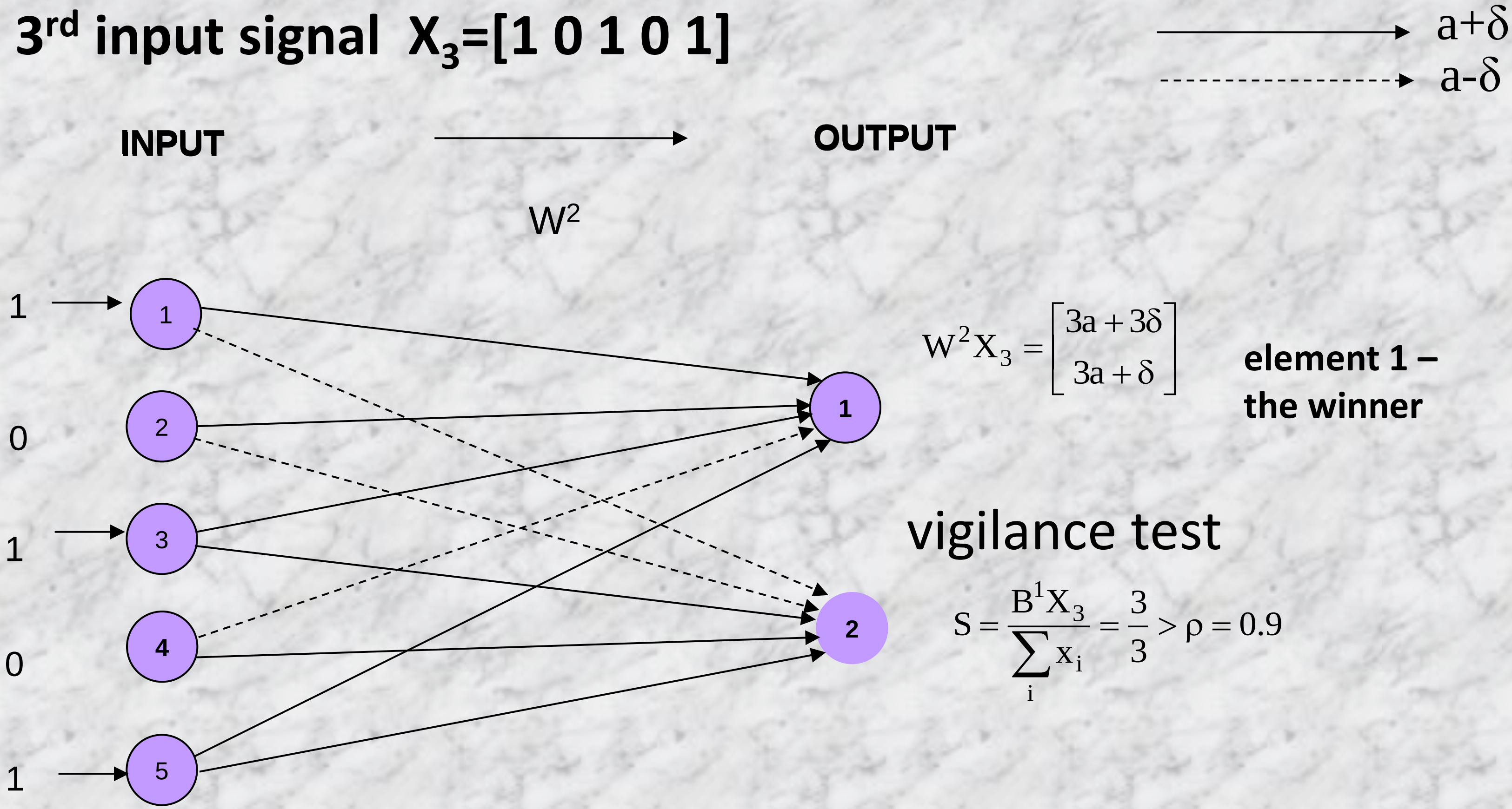


$$W^2 = \begin{bmatrix} a+\delta & a-\delta \\ a+\delta & a-\delta \\ a+\delta & a+\delta \\ a-\delta & a+\delta \\ a+\delta & a-\delta \end{bmatrix}$$

$$B^2 = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 1 \\ 0 & 1 \\ 1 & 1 \end{bmatrix}$$

ART Example

3rd input signal X_3=[1 0 1 0 1]



$$W^2 X_3 = \begin{bmatrix} 3a+3\delta \\ 3a+\delta \end{bmatrix}$$

element 1 – the winner

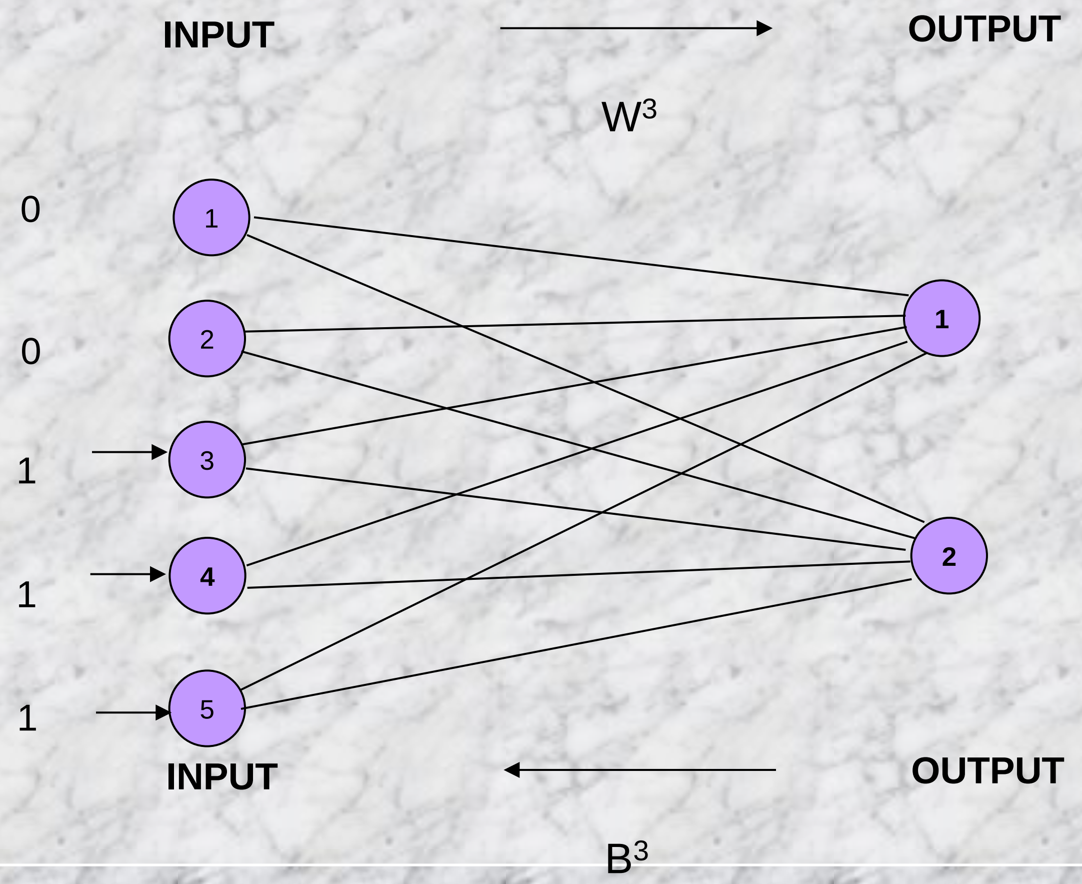
vigilance test

$$S = \frac{B^1 X_3}{\sum_i x_i} = \frac{3}{3} > \rho = 0.9$$

input X_3 is added to the class X_1

ART Example

new network

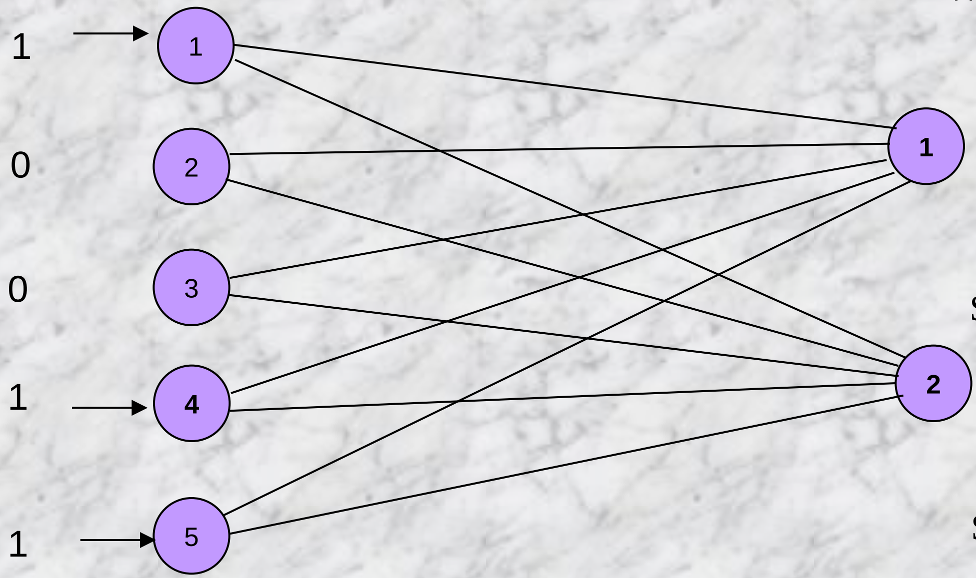


$$W^3 = \begin{bmatrix} a+2\delta & a-\delta \\ a & a-\delta \\ a+2\delta & a+\delta \\ a-2\delta & a+\delta \\ a+2\delta & a-\delta \end{bmatrix}$$

$$B^3 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 1 & 1 \\ 0 & 1 \\ 1 & 1 \end{bmatrix}$$

ART Example

^{4th} **input signal X_4=[1 0 0 1 1]**



$$W^3 X_4 = \begin{bmatrix} 3a + 2\delta \\ 3a + \delta \end{bmatrix}$$

element 1 – the winner

vigilance test

$$S = \frac{B_1^3 X_4}{\sum_i x_i} = \frac{2}{3} < \rho = 0.9$$

element 1 blocked

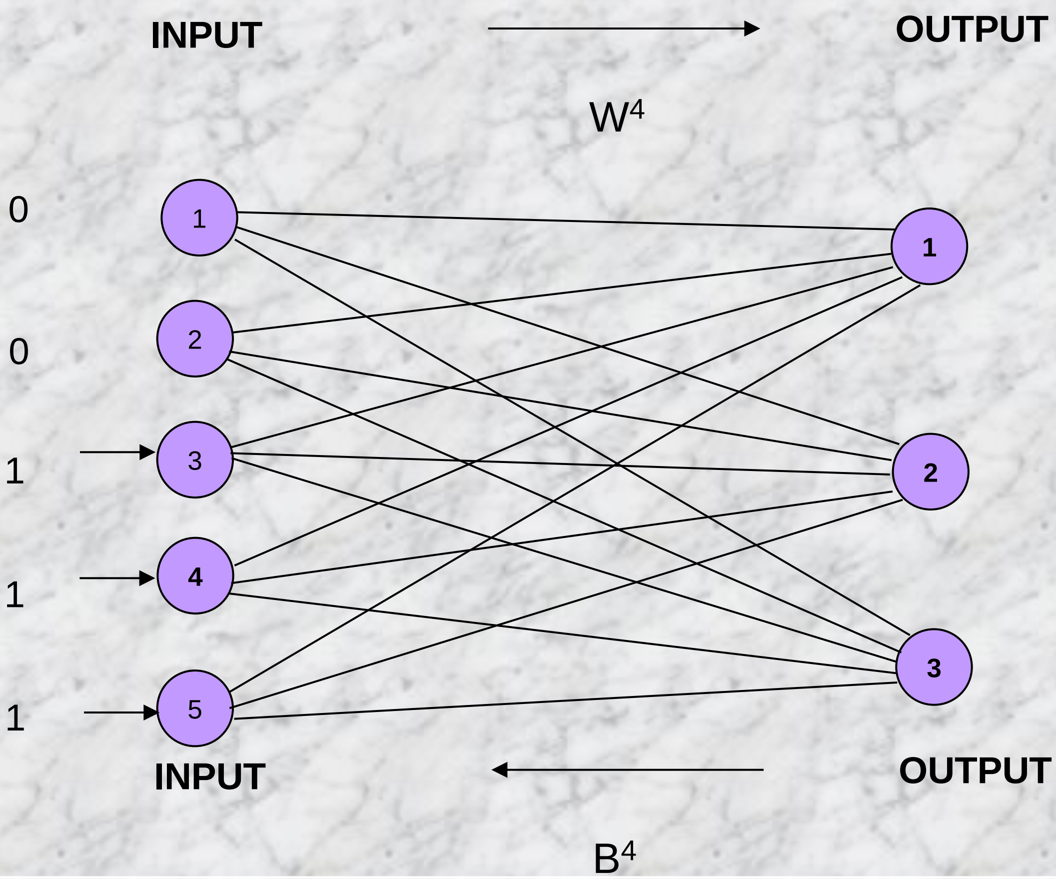
$$S = \frac{B_2^3 X_4}{\sum_i x_i} = \frac{2}{3} < \rho = 0.9$$

element 2 blocked

new element 3 is added

ART Example

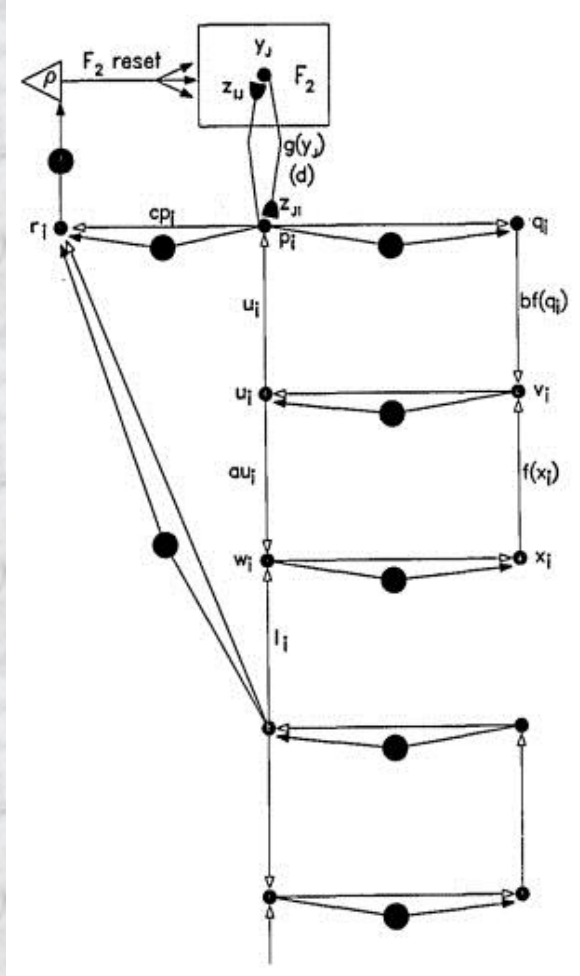
new network



$$W^4 = \begin{bmatrix} a+2\delta & a-\delta & a+\delta \\ a & a-\delta & a-\delta \\ a+2\delta & a+\delta & a-\delta \\ a-2\delta & a+\delta & a+\delta \\ a+2\delta & a+\delta & a+\delta \end{bmatrix}$$

$$B^4 = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

ART2



F2

F1

F0

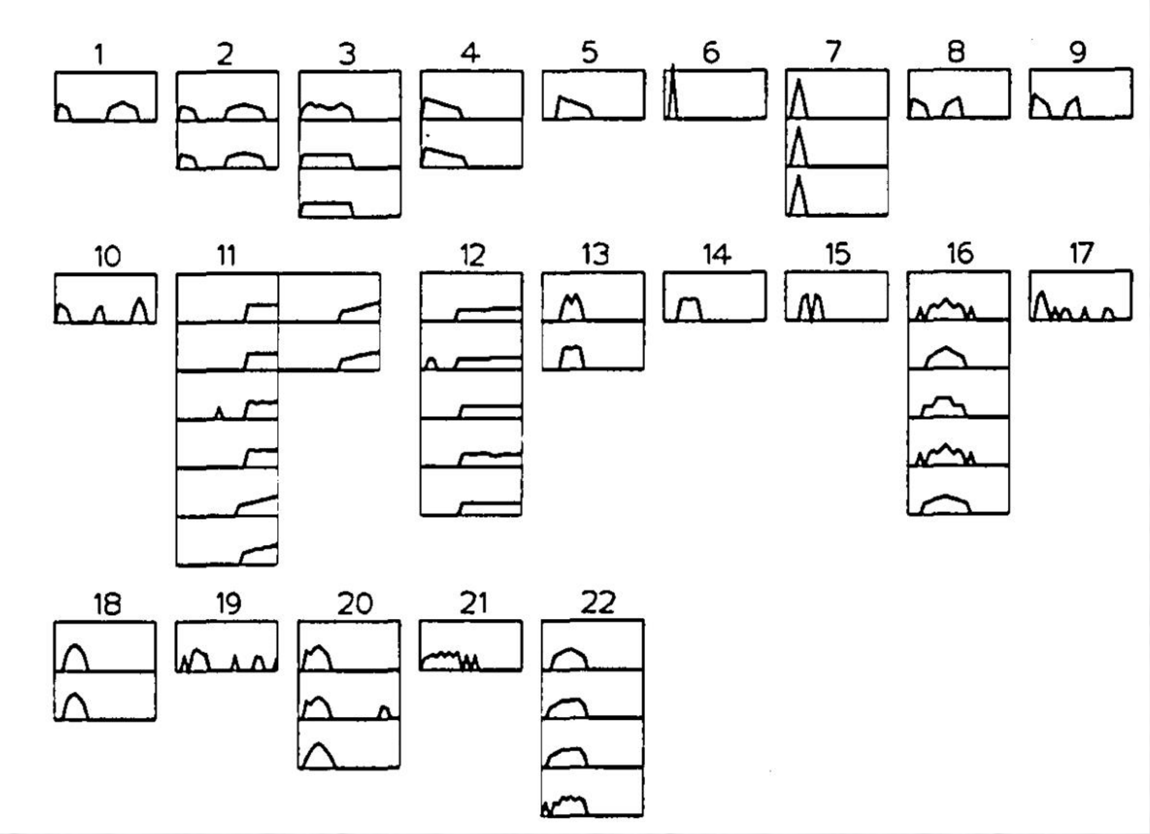
F2 (category field) is similar to the recognition layer in ART1, which is responsible for competitive matching to current input modes representation.

F1 (feature representation field) is similar to the comparison layer in ART1, which includes some calculating tiers and gains controller.

F0 (preprocessing layer) perform normalization, contrast enhancement, noise reduction

source: G.A.Carpenter, S.Grossberg *ART 2: self-organization of stable category recognition codes for analog input patterns,* Applied Optics **26**/23, 4919-4930 (1987)

ART2

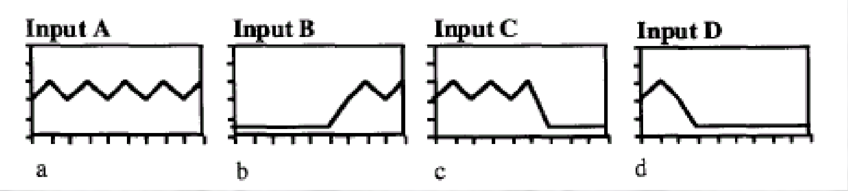


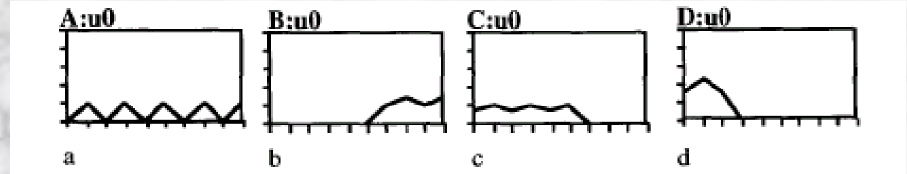
Recognition category summary for the ART 2 system.

source: G.A.Carpenter, S.Grossberg *ART 2: self-organization of stable category recognition codes for analog input patterns,* Applied Optics **26**/23, 4919-4930 (1987)

ART2

- Can classify the analog pattern (gray scale)
- Preprocessing in the layer F0: normalization, contrast enhancement, noise reduction
- Comparison with input in the layer F1
- Patterns are similar if are proportional



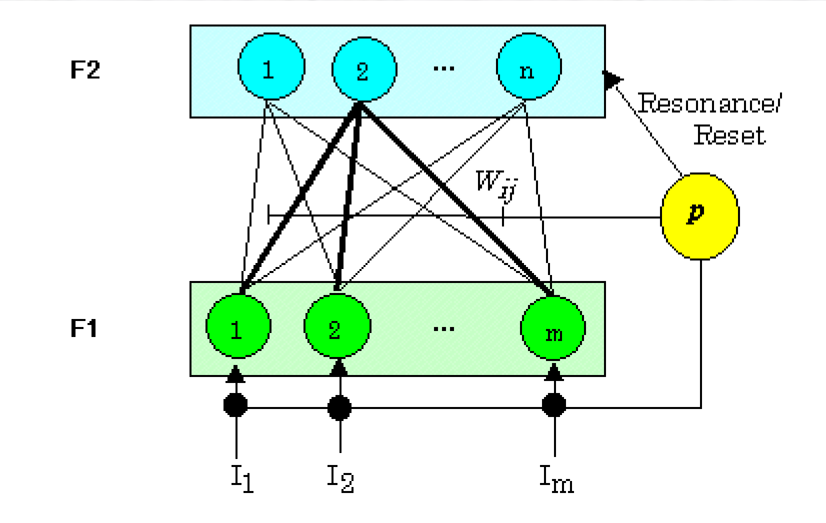


source: G.A.Carpenter, S.Grossberg *ART 2: self-organization of stable category recognition codes for analog input patterns,* Applied Optics **26**/23, 4919-4930 (1987)

Fuzzy ART

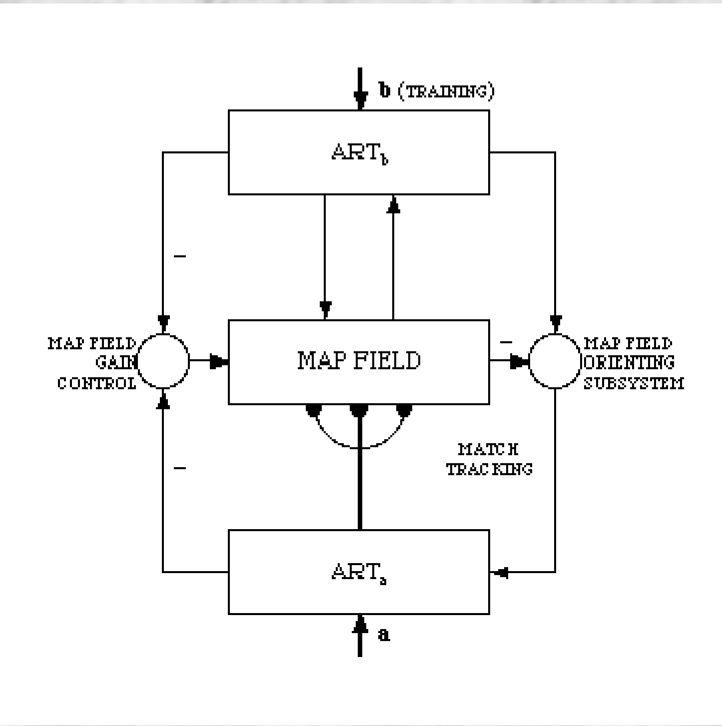
A Fuzzy Adaptive Resonance Theory (ART) model capable of rapid stable learning of recognition categories in response to arbitrary sequences of analog or binary input patterns is described. Fuzzy ART incorporates computations from fuzzy set theory into the ART 1 neural network.

The generalization to learning both analog and binary input patterns is achieved by replacing appearances of the **AND** operator in ART1 by the **MIN** operator of fuzzy set theory.



source: G.A.Carpenter, S.Grossberg, D.Rosen *Fuzzy ART: Fast Stable Learning and Categorization of Analog Patterns by an Adaptive Resonance System,* Neural Networks,**4**,759-771 (1991)

ARTMAP



ARTMAP incorporates one or more of the unsupervised ART modules. In the ARTMAP architecture, (see figure) two ART modules, ART$_a$ and ART$_b$, are linked together via an inter-ART module, called the map field. ARTMAP can be used for mapping multidimensional vectors.

source: D.Weenink *Category ART: a variation on adaptive resonance theory neural networks*, University of Amsterdam, Proceedings 21, 117-129 (1997)

ARTMAP

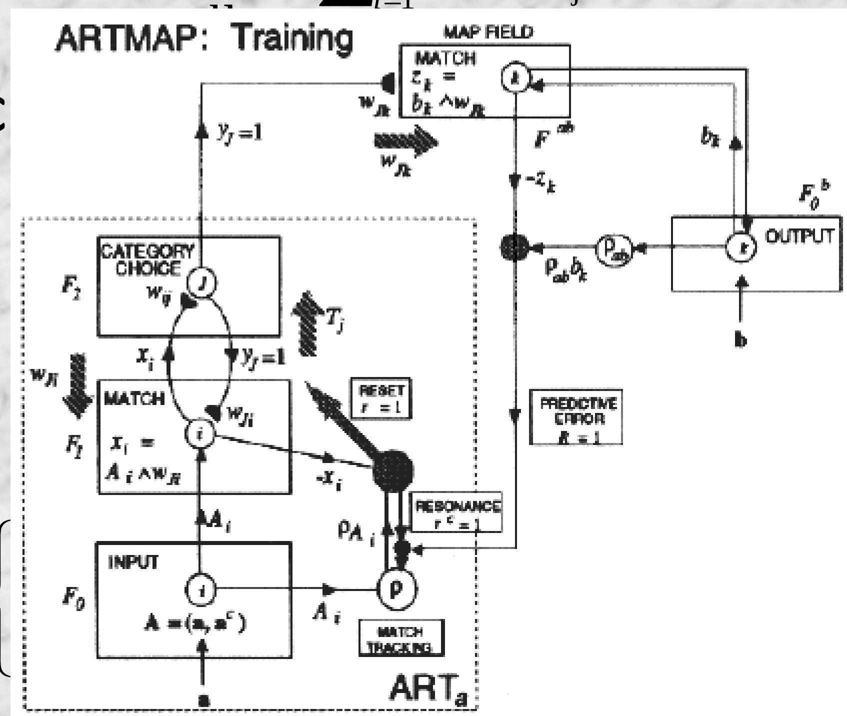
- Calculation of input in the category (output) layer

$$\sum_{i=1}^{N} \min(w_{ji}, x_i)$$

- Vigilanc

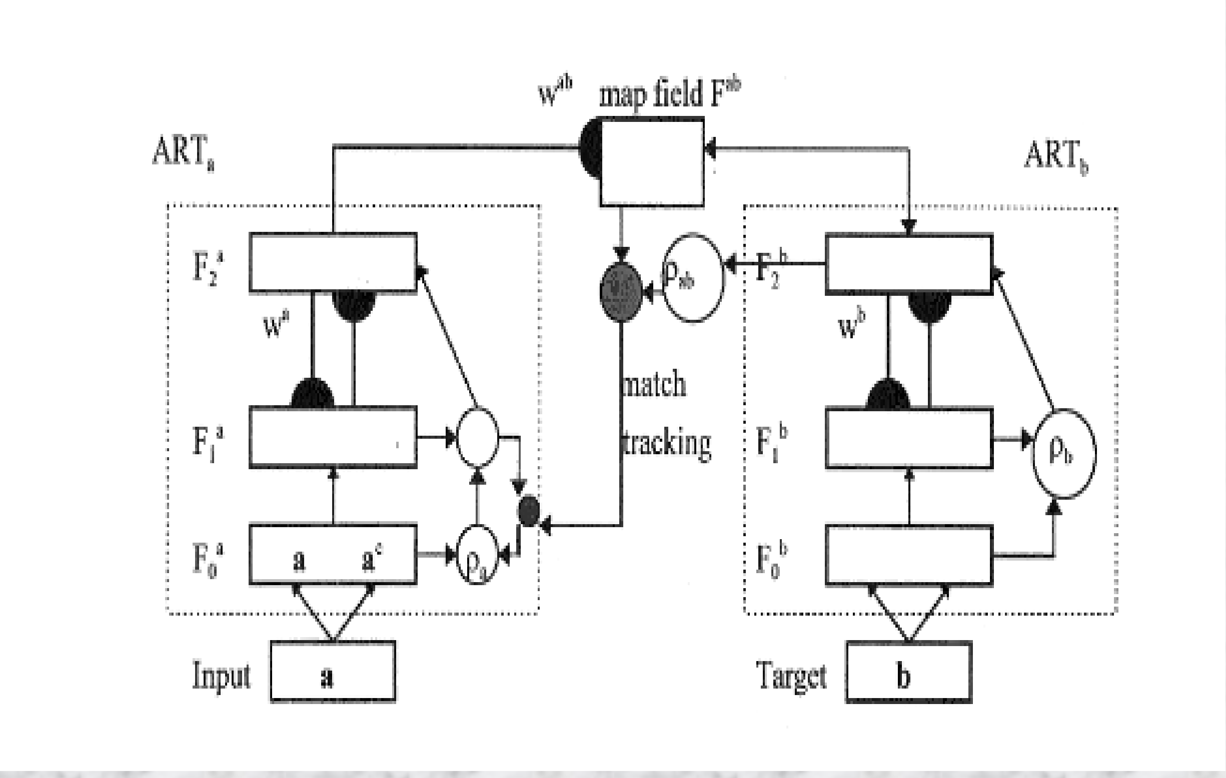
- Change

$$w_{Ji}^{(new)} = \begin{cases} \quad \text{r existing category} \\ \quad \text{for new category} \end{cases}$$



$0 \leq \beta \leq 1$ learning factor (β=1 fast learning)

Fuzzy ARTMAP



In a Fuzzy ARTMAP, categories formed by two Fuzzy ART units ART_a and ART_b, are associated via a MAP field as category and class respectively.

source: G.Carpenter at all. *Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps.* IEEE Trans on Neural Networks,**3**,5,(1992)

Fuzzy ART and Fuzzy ART1

- Calculation of input in the category (output) layer

$$u_i = \frac{\sum_{i=1}^{N} \min(w_{ji}, x_i)}{\alpha + \sum_{i=1}^{N} w_{ji}}$$

- Vigilance test

$$\frac{\sum_{i=1}^{N} \min(w_{Ji}, x_i)}{\sum_{i=1}^{N} w_{Ji}} < \rho$$

- Change weight values

$$w_{Ji}^{(new)} = \begin{cases} w_{Ji}^{(old)} + \bar{\beta}(\min(w_{Ji}^{(old)}, x_i) - w_{Ji}^{(old)}) & \text{for existing category} \\ x_i & \text{for new category} \end{cases}$$

$0 \leq \beta \leq 1$ learning factor (β=1 fast learning)

Applications

- Recognition within the pictures
- Recognition objects from the radar
- Speech recognition
- ....