# Lattice Algebra: Theory and Applications

Prof. Gerhard Ritter

CISE Department, University of Florida

# Overview

**Part I:** Theory

- Pertinent algebraic structures
- Lattice algebra with focus on $\ell$-vector Spaces
- Concluding remarks and questions

**Part II:** Applications

- LNNs
- Matrix based LAMs
- Dendritic LAMs
- Concluding remarks and questions

## History

- Lattice theory in image processing and AI
- Image algebra, mathematical morphology, and HPC

**A pertinent question:**

Why is $(-1) \cdot (-1) = 1$?

# Algebraic Structures

Some basic backgound

Let $G$ be a set with binary operation $\circ$. Then

1. $(G, \circ)$ is a *groupoid*
2. if $x \circ (y \circ z) = (x \circ y) \circ z$, then $(G, \circ)$ is a *semigroup*
3. if $G$ is a semigroup and and $G$ has an identity element, then $G$ is a *monoid*
4. if $G$ is a monoid and every element of $G$ has an inverse, then $G$ is a *group*
5. if $G$ is a group and $x \circ y = y \circ x \; \forall \, x, y \in G$, then $G$ is an *abelian* group.

# Algebraic Structures

Why are groups important?

**Theorem.** If $(X, \cdot)$ is a group and $a, b \in X$, then the linear equations $a \cdot x = b$ and $y \cdot a = b$ have unique solutions in $X$.

**Remark:** Note that the solutions $x = a^{-1} \cdot b$ and $y = b \cdot a^{-1}$ need not be the same unless $X$ is abelian.

# Algebraic Structures

Sets with Multiple Operations

Suppose that $X$ is a set with two binary operations $\star$ and $\circ$. The operation $\circ$ is said to be *left distributive* with respect to $\star$ if

$$x \circ (y \star z) = (x \circ y) \star (x \circ z) \; \forall \, x, \, y, \, z \in X \qquad (1)$$

and *right distributive* if

$$(y \star z) \circ x = (y \circ x) \star (z \circ x) \; \forall \, x, \, y, \, z \in X. \qquad (2)$$

Division on $\mathbb{R}^+$ is not left distributive over addition;

$(y + z)/x = (y/x) + (z/x)$ but

$x/(y + z) \neq (x/y) + (x/z)$.

When both equations hold, we simply say that $\circ$ is *distributive* with respect to $\star$.

# Algebraic Structures

**Definition:** A *ring* $(R, +, \cdot)$ is a set $R$ together with two binary operations $+$ and $\cdot$ of addition and multiplication, respectively, defined on $R$ such that the following axioms are satisfied:

1. $(R, +)$ is an abelian group.
2. $(R, \cdot)$ is a semigroup.
3. $\forall\, a, b, c \in R$, $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$ and $(a + b) \cdot c = (a \cdot c) + (b \cdot c)$.

If axiom 1 in this definition is weakened to $(R, +)$ is a commutative semigroup, then $R$ is called a *semiring*.

# Algebraic Structures

If $(R, +, \cdot)$ is a ring, we let $0$ denote the additive identity and $1$ the multiplicative identity (if it exists). If $R$ satisfies the property

- For every nonzero $a \in R$ there is an element in $R$, denoted by $a^{-1}$, such that $a \cdot a^{-1} = a^{-1} \cdot a = 1$ (i.e. $(R \setminus \{0\}, \cdot)$ is a group),

then $R$ is called *division* ring. A commutative division ring is called a *field*
You should now be able to prove that $(-1) \cdot (-1) = 1$.

# Partially Ordered Sets

**Definition:** A relation $\preccurlyeq$ on a set $X$ is called a *partial order* on $X$ if and only if for every $x,\ y,\ z \in X$ the following three conditions are satisfied:

1. $x \preccurlyeq x$ (reflexive)
2. $x \preccurlyeq y$ and $y \preccurlyeq x \Rightarrow x = y$ (antisymmetric)
3. $x \preccurlyeq y$ and $y \preccurlyeq z \Rightarrow x \preccurlyeq z$ (transitive)

The *inverse* relation of $\preccurlyeq$, denoted by $\succcurlyeq$, is also a partial order on $X$.

**Definition:** The *dual* of a partially ordered set $X$ is that partially ordered set $X^*$ defined by the inverse partial order relation on the same elements.

Since $(X^*)^* = X$, this terminology is legitimate.

# Lattices

**Definition:** A *lattice* is a partially ordered set $L$ such that for any two elements $x, y \in L$, $glb\{x, y\}$ and $lub\{x, y\}$ exist. If $L$ is a lattice, then we define $x \wedge y = glb\{x, y\}$ and $x \vee y = lub\{x, y\}$.

- A *sublattice* of a lattice $L$ is a subset $X$ of $L$ such that for each pair $x, y \in X$, we have that $x \wedge y \in X$ and $x \vee y \in X$.
- A lattice $L$ is said to be *complete* if and only if for each of its subsets $X$, $infX$ and $supX$ exist. We define the symbols $\bigwedge X = infX$ and $\bigvee X = supX$.

## $s\ell$-Semigroups and $\ell$-Groups

Suppose $(R, \circ)$ is a semigroup or group and $R$ is a lattice $(R, \vee, \wedge)$ or semilattice $(R, \vee)$.

**Definition:** A *group translation* $\psi$ is a function $\psi : R \to R$ of form

$$\psi(x) = a \circ x \circ b,$$

where $a$, $b$ are constants.

The translation $\psi$ is said to be *isotone* if and only if

$$x \preccurlyeq y \;\Rightarrow\; \psi(x) \preccurlyeq \psi(y)$$

Note that a group translation is a unary operation.

# $s\ell$-Semigroups and $\ell$-Groups

**Definition:** A $\ell$-*group* ($\ell$-*semigroup*) is of form $(R, \vee, \wedge, +)$, where $(R, +)$ is a group (semigroup) and $(R, \vee, \wedge)$ is a lattice, *and* every group translation is isotone.

If $R$ is just a semilattice - i.e., $(R, \vee)$ or $(R, \wedge)$ - in the definition, then $(R, \vee, +)$ (or $(R, \wedge, +)$) an $s\ell-$*group* if $(R, +)$ is a group and an $s\ell$-*semigroup* if $(R, +)$ is a semigroup.

# $s\ell$-Vector Spaces and $\ell$-Vector Spaces

**Definition:** A $s\ell-$*vector space* $\mathbb{V}$ over the $s\ell$-group (or $s\ell$-monoid) $(R, \vee, +)$, denoted by $\mathbb{V}(R)$, is a semilattice $(\mathbb{V}, \vee)$ together with an operation called *scalar addition* of each element of $\mathbb{V}$ by an element of $R$ on the left, such that $\forall\, \alpha,\ \beta \in R$ and $\mathbf{v},\ \mathbf{w} \in \mathbb{V}$, the following conditions are satisfied:

1. $\alpha + \mathbf{v} \in \mathbb{V}$
2. $\alpha + (\beta + \mathbf{v}) = (\alpha + \beta) + \mathbf{v}$
3. $(\alpha \vee \beta) + \mathbf{v} = (\alpha + \mathbf{v}) \vee (\beta + \mathbf{v})$
4. $\alpha + (\mathbf{v} \vee \mathbf{w}) = (\alpha + \mathbf{v}) \vee (\alpha + \mathbf{w})$
5. $0 + \mathbf{v} = \mathbf{v}$

# $s\ell$-Vector Spaces and $\ell$-Vector Spaces

The $s\ell$-vector space is also called a *max* vector space, denoted by $\vee$-vector space. Using the duals $(R, \wedge, +)$ and $(\mathbb{V}, \vee)$, and replacing conditions (3.) and (4.) by

3'. $(\alpha \wedge \beta) + \mathbf{v} = (\alpha + \mathbf{v}) \wedge (\beta + \mathbf{v})$

4'. $\alpha + (\mathbf{v} \wedge \mathbf{w}) = (\alpha + \mathbf{v}) \wedge (\alpha + \mathbf{w})$,

we obtain the *min* vector space denoted by $\wedge$-vector space.

Note also that replacing $\vee$ (or $\wedge$) by $+$ and $+$ by $\cdot$, we obtain the usual axioms defining a vector space.

# $s\ell$-Vector Spaces and $\ell$-Vector Spaces

**Definition:** If we replace the semilattice $\mathbb{V}$ by a lattice $(\mathbb{V}, \vee, \wedge)$, the $s\ell$-group (or $s\ell$-semigroup) $R$ by an $\ell$-group (or $\ell$-semigroup) $(R, \vee, \wedge, +)$, and conditions 1 through 5 and 3' and 4' are all satisfied, then $\mathbb{V}(R)$ is called an $\ell$-*vector space*.

**Remark.** The lattice vector space definitions given above are drastically different from *vector lattices* as postulated by Birkhoff and others! A vector lattice is simply a partially ordered real vector space satisfying the isotone property.

# Lattice Algebra and Linear Algebra

The theory of $\ell$-groups, $s\ell$-groups, $s\ell$-semigroups, $\ell$-vector spaces, etc. provides an extremely rich setting in which many concepts from linear algebra and abstract algebra can be transferred to the lattice domain via analogies. $\ell$-vector spaces are a good example of such an analogy. The next slides will present further examples of such analogies.

# Lattice Algebra and Linear Algebra

**Ring:** $(\mathbb{R}, +, \cdot)$

- $a \cdot 0 = 0 \cdot a = 0$
- $a + 0 = 0 + a = a$
- $a \cdot 1 = 1 \cdot a = a$
- $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$

**Semi-Ring or $s\ell$-Group:** $(\mathbb{R}_{-\infty}, \vee, +)$

- $a + (-\infty) = (-\infty) + a = -\infty$
- $a \vee (-\infty) = (-\infty) \vee a = a$
- $a + 0 = 0 + a = a$
- $a + (b \vee c) = (a + b) \vee (a + c)$

# Lattice Algebra and Linear Algebra

- Since $(\mathbb{R}_{-\infty}, \vee, +)^* = (\mathbb{R}_{\infty}, \wedge, +^*)$, $(\mathbb{R}_{\infty}, \wedge, +^*)$ is also an $s\ell-$semigroup (with $+^* = +$) isomorphic to $(\mathbb{R}_{-\infty}, \vee, +)$
- Defining $a +^* b = a + b \, \forall \, a, b \in \mathbb{R}_{-\infty}$ and

$$
\begin{aligned}
-\infty + \infty = \infty + -\infty = -\infty \\
-\infty +^* \infty = \infty +^* -\infty = \infty,
\end{aligned}
$$

we can combine $(\mathbb{R}_{-\infty}, \vee, +)$ and $(\mathbb{R}_{\infty}, \wedge, +)$ into one well defined algebraic structure $(\mathbb{R}_{\pm\infty}, \vee, \wedge, +, +^*)$.

# Lattice Algebra and Linear Algebra

- The structure $(\mathbb{R}, \vee, \wedge, +)$ is an $\ell$-group.
- The structures $(\mathbb{R}_{-\infty}, \vee, \wedge, +)$ and $(\mathbb{R}_{\infty}, \vee, \wedge, +)$ are $\ell$-semigroups.
- The structure $(\mathbb{R}_{\pm\infty}, \vee, \wedge)$ is a bounded distributive lattice.
- The structure $(\mathbb{R}_{\pm\infty}, \vee, \wedge, +, +^*)$ is called a *bounded lattice ordered group* or *blog*, since the underlying structure $(\mathbb{R}, +)$ is a group.

# Matrix Addition and Multiplication

Suppose $A = (a_{ij})_{m\times n}$ and $B = (b_{ij})_{m\times n}$ with entries in $\mathbb{R}_{\pm\infty}$. Then

- $C = A \vee B$ is defined by setting $c_{ij} = a_{ij} \vee b_{ij}$, and
- $C = A \wedge B$ is defined by setting $c_{ij} = a_{ij} \wedge b_{ij}$.

If $A = (a_{ij})_{m\times p}$ and $B = (b_{ij})_{p\times n}$, then

- $C = A \boxdot\!\!\!\vee\, B$ is defined by setting $c_{ij} = \bigvee_{k=1}^{p}(a_{ik} + b_{kj})$, and
- $C = A \boxdot\!\!\!\wedge\, B$ is defined by setting $c_{ij} = \bigwedge_{k=1}^{p}(a_{ik} +^{*} b_{kj})$.
- $\boxdot\!\!\!\vee$ and $\boxdot\!\!\!\wedge$ are called the *max* and *min* products, respectively.

# Zero and Identity Matrices

For the semiring $(M_{n\times n}(\mathbb{R}_{-\infty}), \vee, \boxed{\vee})$, the *null* matrix is

$$\Phi = \begin{pmatrix} -\infty & \cdot & \cdot & \cdot & -\infty \\ \cdot & -\infty & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ -\infty & \cdot & \cdot & \cdot & -\infty \end{pmatrix}$$

# Zero and Identity Matrices

For the semiring $(M_{n\times n}(\mathbb{R}_{-\infty}), \vee, \boxed{\vee})$, the *identity* matrix is

$$I = \begin{pmatrix} 0 & -\infty & \cdot & \cdot & -\infty \\ -\infty & 0 & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & -\infty \\ -\infty & \cdot & \cdot & -\infty & 0 \end{pmatrix}$$

# Matrix Properties

We have $\forall\, A,\ B,\ C \in M_{n\times n}(\mathbb{R}_{-\infty})$

$$\begin{aligned}
A \boxtimes (B \vee C) &= (A \boxtimes B) \vee (A \boxtimes C) \\
I \boxtimes A &= A \boxtimes I = A \\
A \vee \Phi &= \Phi \vee A = A \\
A \boxtimes \Phi &= \Phi \boxtimes A = \Phi
\end{aligned}$$

Analogous laws hold for the semiring $(M_{n\times n}(\mathbb{R}_{\infty}), \wedge,\ \boxtimes\ )$,

## Conjugation

If $r \in \mathbb{R}_{\pm\infty}$, then the *additive conjugate* of $r$ is the unique element $r^*$ defined by

$$r^* = \begin{cases} -r & \text{if } r \in \mathbb{R} \\ -\infty & \text{if } r = \infty. \\ \infty & \text{if } r = -\infty \end{cases}$$

- $(r^*)^* = r$ and $r \wedge s = (r^* \vee s^*)^*$
- It follows that $r \wedge s = -(-r \vee -s)$ and
- $A \wedge B = (A^* \vee B^*)^*$ and $A \boxtimes B = (B^* \boxtimes A^*)^*$,

where $A = (a_{ij})$ and $A^* = (a^*_{ji})$.

## $s\ell$-Sums

**Definition:** If $X = \{\mathbf{x}^1, \ldots, \mathbf{x}^k\} \subset \mathbb{R}^n_{-\infty}$ (or $X \subset \mathbb{R}^n_{\infty}$), then $\mathbf{x} \in \mathbb{R}^n_{-\infty}$ (or $\mathbf{x} \in \mathbb{R}^n_{\infty}$) is said to be a *linear max (min) combination* of $X$ if $\mathbf{x}$ can be written as

$$\mathbf{x} = \bigvee_{\xi=1}^{k} (\alpha_\xi + \mathbf{x}^\xi) \quad (\textit{or } \mathbf{x} = \bigwedge_{\xi=1}^{k} (\alpha_\xi + \mathbf{x}^\xi)),$$

where $\alpha \in \mathbb{R}_{-\infty}$ (or $\alpha \in \mathbb{R}_{\infty}$) and $\mathbf{x}^\xi \in \mathbb{R}^n_{-\infty}$ (or $\mathbf{x}^\xi \in \mathbb{R}^n_{\infty}$.

The expressions $\bigvee_{\xi=1}^{k}(\alpha_\xi + \mathbf{x}^\xi)$ and $\bigwedge_{\xi=1}^{k}(\alpha_\xi + \mathbf{x}^\xi)$ are called a *linear max sum* and a *linear min sum*, respectively.

# $s\ell$-Independence

**Definition:** Given the $s\ell$-vector space $(\mathbb{R}^n_{-\infty}, \vee)$ over $(\mathbb{R}_{-\infty}, \vee, +)$, $X = \{\mathbf{x}^1, \ldots, \mathbf{x}^k\} \subset \mathbb{R}^n_{-\infty}$, and $\mathbf{x} \in \mathbb{R}^n_{\infty}$, then $\mathbf{x}$ is said to be *max dependent* or *$s\ell$-dependent* on $X \Leftrightarrow \mathbf{x} = \bigvee_{\xi=1}^{k}(\alpha_\xi + \mathbf{x}^\xi)$ for some linear max sum of vectors from $X$. If $\mathbf{x}$ is not max dependent on $X$, then $\mathbf{x}$ is said to be *max independent* of $X$.

The set $X$ is *$s\ell$-independent* or *max independent* $\Leftrightarrow$ $\forall\, \xi \in \{1, \ldots, k\}$, $\mathbf{x}^\xi$ is $s\ell$-independent of $X \setminus \{\mathbf{x}^\xi\}$.

# $s\ell$-Subspaces and Spans

**Definition:** If $X \subset \mathbb{R}^n_{-\infty}$, then $(X, \vee)$ is an $s\ell$-subspace of $(\mathbb{R}^n_{-\infty}, \vee) \Leftrightarrow$ the following are satisfied:

1. if $\mathbf{x}$, $\mathbf{y} \in X$, then $\mathbf{x} \vee \mathbf{y} \in X$
2. $\alpha + \mathbf{x} \in X \; \forall \, \alpha \in \mathbb{R}_{-\infty}$ and $\mathbf{x} \in X$.

**Definition:** If $X \subset \mathbb{R}^n_{-\infty}$, then the $s\ell$-*span* of $X$ is the set

$$S(X) = \{\mathbf{x} \in \mathbb{R}^n_{-\infty};\ \mathbf{x} \text{ is max dependent on } X\}.$$

## $s\ell$-Spans and Bases

**Remark:** If $\mathbf{x} \in S(X)$, then $\alpha + \mathbf{x} \in S(X)$ and $\mathbf{x} \vee \mathbf{y} \in S(X)\ \forall\, \mathbf{x}, \mathbf{y} \in S(X)$. Thus $S(X)$ is an $s\ell$-vector subspace of $\mathbb{R}^n_{-\infty}$.

If $S(X) = \mathbb{R}^n_{-\infty}$, then we say that $X$ *spans* $\mathbb{R}^n_{-\infty}$ and $X$ is called a set of *generators* for $\mathbb{R}^n_{-\infty}$.

**Definition:** A basis for an $s\ell$-vector space $(\mathbb{V}, \vee)$ (or $(\mathbb{V}, \wedge)$) is a set of $s\ell$-independent vectors which spans $\mathbb{V}$.

# $s\ell$-independence

**Example.** The set $X = \{(0, -\infty), (-\infty, 0)\}$ spans $\mathbb{R}^2_{-\infty}$ and is $s\ell$-independent. Thus $X$ is a basis for $\mathbb{R}^2_{-\infty}$

**Question:** What is a basis for $\mathbb{R}^2$?

**Question:** If $a \in \mathbb{R}$, what is the span of $X = \{(0, a), (-\infty, 0)\}$ in $\mathbb{R}^2_{-\infty}$?

**Question:** What is the span of $X = \{(1, 0), (0, 1)\}$ in $\mathbb{R}^2_{-\infty}$?

# $\ell$-Vector Spaces

Most of what we have said for $s\ell$-vector spaces also holds for $\ell$-vector spaces with the appropriate changes. Thus, for $(\mathbb{R}^n_{\pm\infty}, \vee, \wedge)$ we have:

- If $\{\mathbf{x}^1, \ldots, \mathbf{x}^k\} \subset \mathbb{R}^n_{\pm\infty}$, then a *linear minimax combination* of vectors from the set $\{\mathbf{x}^1, \ldots, \mathbf{x}^k\}$ is any vector $\mathbf{x} \in \mathbb{R}^n_{\pm\infty}$ of form

$$\mathbf{x} = \mathfrak{S}(\mathbf{x}^1, \ldots, \mathbf{x}^k) = \bigvee_{j \in J} \bigwedge_{\xi=1}^{k} (a_{\xi j} + \mathbf{x}^\xi), \quad (3)$$

where $J$ is a finite set of indices and $a_{\xi j} \in \mathbb{R}_{\pm\infty}$ $\forall j \in J$ and $\forall \xi = 1, \ldots, k$.

# $\ell$-Vector Spaces

- The expression $\mathfrak{S}(\mathbf{x}^1, \ldots, \mathbf{x}^k) = \bigvee_{j \in J} \bigwedge_{\xi=1}^{k} (a_{\xi j} + \mathbf{x}^{\xi})$ is called a *linear minimax sum* or an $\ell$-*sum*.
- Similarly we can combine the structures $(M(\mathbb{R}^n_{\pm\infty})_{n \times n}, \vee, \boxed{\vee})$ and $(M(\mathbb{R}^n_{\pm\infty})_{n \times n}, \wedge, \boxed{\wedge})$ to obtain the blog $(M(\mathbb{R}^n_{\pm\infty})_{n \times n}, \vee, \wedge, \boxed{\vee}, \boxed{\wedge})$ in order to obtain a coherent minimax theory for matrices.
- Many of the concepts found in the corresponding linear domains can then be realized in these lattice structures via appropriate analogies.

## $\ell$-Transforms

**Definition:** A *linear max transform* or $s\ell$-*transform* of an $s\ell$-vector space $\mathbb{V}(R)$ into an $s\ell$-vector space $\mathbb{W}(R)$ is a function $L : \mathbb{V} \to \mathbb{W}$ which satisfies the condition

$$L((\alpha + \mathbf{v}) \vee (\beta + \mathbf{u})) = (\alpha + L(\mathbf{v})) \vee (\beta + L(\mathbf{u}))$$

for all scalars $\alpha, \beta \in R$ and all $\mathbf{v}, \mathbf{u} \in \mathbb{V}$.

A linear *min* transform obeys

$$L((\alpha + \mathbf{v}) \wedge (\beta + \mathbf{u})) = (\alpha + L(\mathbf{v})) \wedge (\beta + L(\mathbf{u}))$$

and a linear *minimax* transform of an $\ell$-vector space $\mathbb{V}(R)$ into an $\ell$-vector space $\mathbb{W}(R)$ obeys both of the equations.

# $s\ell$-transforms and polynomials

- Just as in linear algebra, it is easy to prove that any $m \times n$ matrix $M$ with entries from $\mathbb{R}^m_{-\infty}$ (or $\mathbb{R}^m_{\infty}$) corresponds to a linear max (or min) transform from $\mathbb{R}^m_{-\infty}$ into $\mathbb{R}^n_{-\infty}$ (or $\mathbb{R}^m_{\infty}$ into $\mathbb{R}^n_{\infty}$). Simply define

$$L_M(\mathbf{x}) = M \boxtimes \mathbf{x} \quad \forall\, \mathbf{x} \in \mathbb{R}^m_{-\infty}$$

- The subject of $\ell$- and $s\ell$-polynomials also bears many resemblances to the theory of polynomials and waits for further exploration.

# $s\ell$-Polynomials

**Definition:** max polynomial of degree $n$ with coefficients in the appropriate semiring $R$ in the indeterminate $x$ is of form

$$p(x) = \bigvee_{i=0}^{\infty} (a_i + ix),$$

where $a_i = -\infty$ for all but a finite number of $i$.

- If for some $i > 0$ $a_i \neq -\infty$, then the largest such $i$ is called the *degree* of $p(x)$ If no such $i > 0$ exists, then the degree of $p(x)$ is zero.
- For min polynomials simply replace $\bigvee$ by $bigwedge$. Combining the two notions will result in minimax polynomials.

# Discussion and Questions

1. Many items have not been discussed; e.g., eigenvalues and eigenvectors.
2. Applications have not been discussed. We will discuss some in the second talk.
3. Questions?

**Thank you!**

# Associative Memories (AMs)

Suppose $X = \{\mathbf{x}^1, \ldots, \mathbf{x}^k\} \subset \mathbb{R}^n$ and $Y = \{\mathbf{y}^1, \ldots, \mathbf{y}^k\} \subset \mathbb{R}^m$.

- A function $M : \mathbb{R}^n \to \mathbb{R}^m$ with the property that $M(\mathbf{x}^\xi) = \mathbf{y}^\xi \; \forall \xi = 1, \ldots, k$ is called an *associative memory* that identifies $X$ with $Y$.
- If $X = Y$, then $M$ is called an *auto-associative memory* and if $X \neq Y$, then $M$ is called a *hetero-associative memory*.
- $M$ is said to be *robust in the presence of noise* if $M(\tilde{\mathbf{x}}^\xi) = \mathbf{y}^\xi$, for every corrupted version $\tilde{\mathbf{x}}^\xi$ of the prototype input patterns $\mathbf{x}^\xi$.

# Robustness in the Presence of Noise

- We say that $M$ is *robust in the presence of noise* bounded by $\mathbf{n} = (n_1, n_2, \ldots, n_n)'$ if and only if whenever $\mathbf{x}$ represents a distorted version of $\mathbf{x}^\xi$ with the property that $\left|\mathbf{x} - \mathbf{x}^\xi\right| \leq \mathbf{n}$, then $M(\mathbf{x}) = \mathbf{y}^\xi$.

**Remark**: In this theory, it may be possible to have $n_i = \infty$ for some $i$ if that is desirable.

- The concept of the noise bound can be generalized to be bounded by the set $\left\{\mathbf{n}^1, \mathbf{n}^2, \ldots, \mathbf{n}^k\right\}$, with $\mathbf{n}$ being replaced by $\mathbf{n}^\xi$ in the above inequality so that $\left|\mathbf{x} - \mathbf{x}^\xi\right| \leq \mathbf{n}^\xi$.

# Matrix Bases AMs

- The Steinbuch Lernmatrix (1961), auto- and hetero-associative memories.
- The classical Hopfield net is an example of an auto-associative memory.
- The Kohonen correlation matrix memory is an example of a hetero-associative memory.
- The lattice based correlation matrix memories $W_{XY}$ and $M_{XY}$.

# Lattice-based Associative Memories

- For a pair $(X, Y)$ of pattern associations, the two canonical lattice memories $W_{XY}$ and $M_{XY}$ are defined by:

$$w_{ij} = \bigwedge_{\xi=1}^{k} \left( y_i^{\xi} - x_j^{\xi} \right) \text{ and } m_{ij} = \bigvee_{\xi=1}^{k} \left( y_i^{\xi} - x_j^{\xi} \right).$$

- **Fact**. If $X = Y$, then

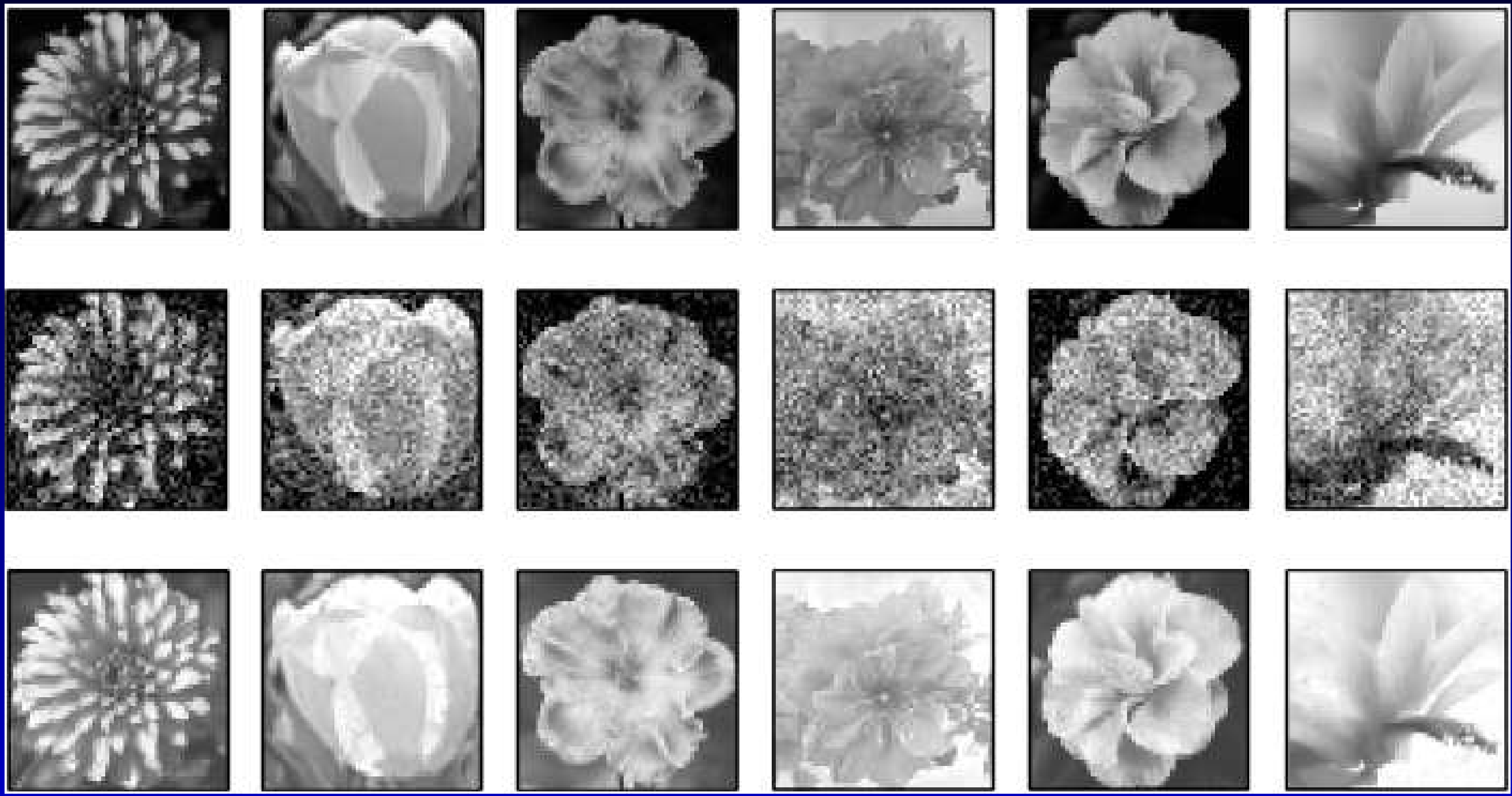
$$W_{XX} \boxtimes \mathbf{x}^{\xi} = \mathbf{x}^{\xi} = M_{XX} \boxtimes \mathbf{x}^{\xi} \;\; \forall \xi = 1, \ldots, k\,.$$

# Lattice-based Associative Memories

We have

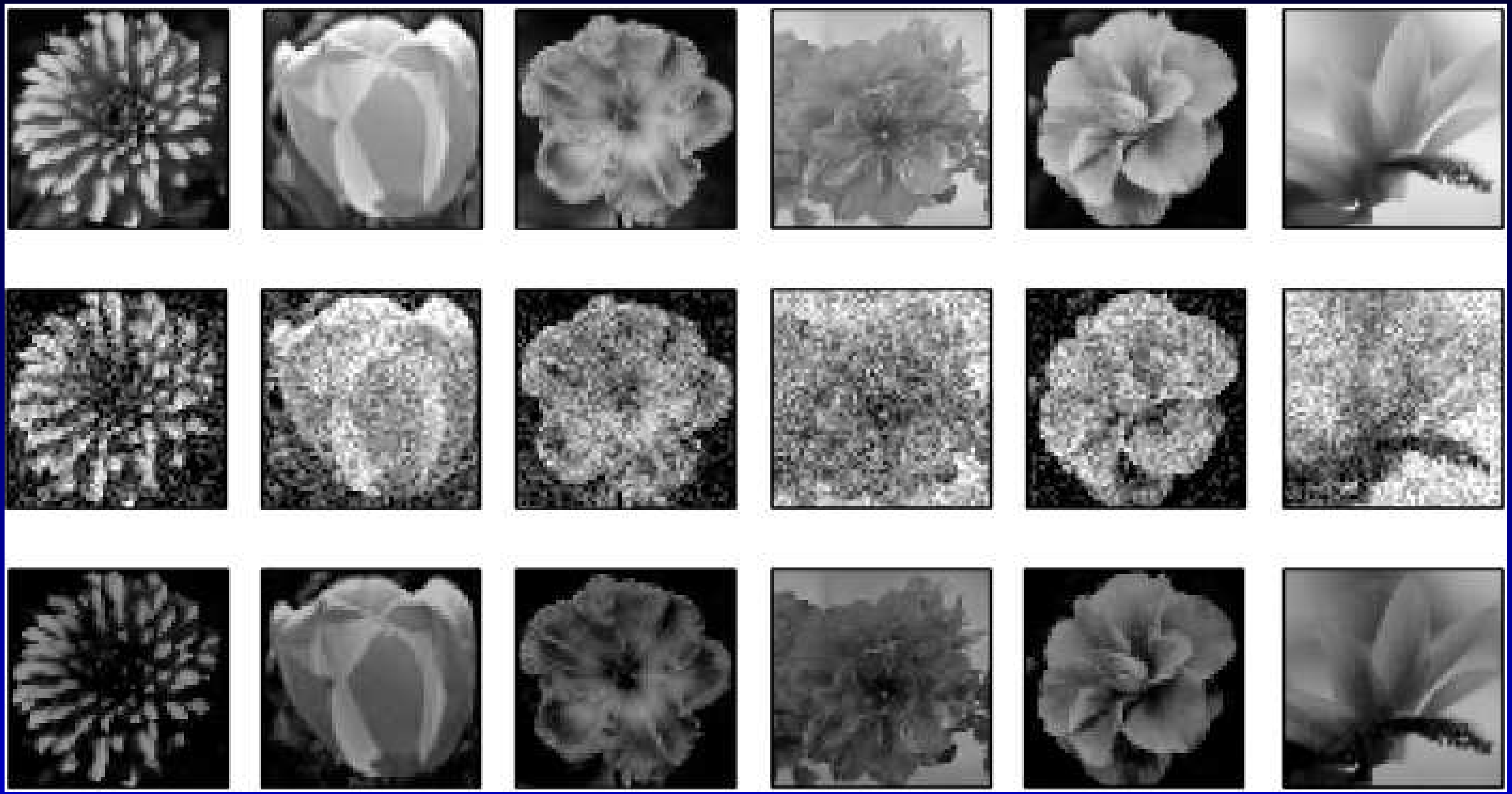
1. $W_{XY} = Y \boxed{\wedge} X^*$ and $M_{XY} = Y \boxed{\vee} X^*$.
2. $W_{XY} = (X \boxed{\vee} Y^*)^* = M^*_{YX}$ and
   $M_{YX} = (X \boxed{\wedge} Y^*)^* = W^*_{YX}$.
3. $\mathbf{x}^\xi \;\rightarrow\; \{W_{XY} \,|\, M_{XY}\} \;\rightarrow\; \mathbf{y}^\xi \;\rightarrow$
   $\{M_{YX} \,|\, W_{YX}\} \;\rightarrow \mathbf{x}^\xi$.
4. This provides for a biassociative memory (LBAM).

# Behavior of $W_{XX}$ in Presence of Random Noise



Top row to bottom row patterns: Original; Noisy; Recalled. The output of $W_{XX}$ appears shifted towards white pixel values.

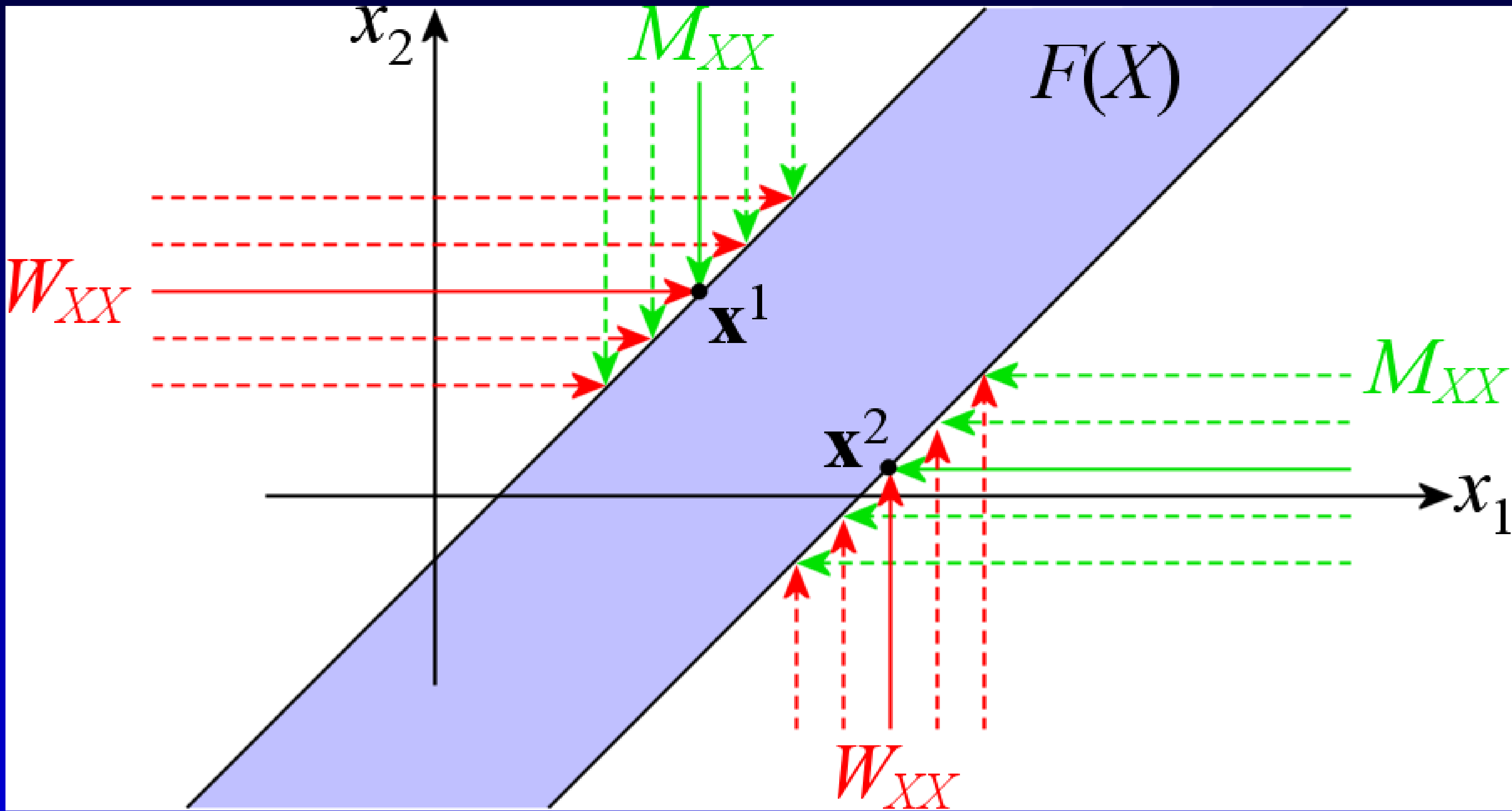
# Behavior of $M_{XX}$ in Presence of Random Noise



Top row to bottom row patterns: Original; Noisy; Recalled. The output of $M_{XX}$ appears shifted towards black pixel values.

# Behavior of $W_{XX}$ and $M_{XX}$ in $\mathbb{R}^2$

The orbits of $W_{XX}$ and $M_{XX}$ for $X = \{\mathbf{x}^1, \mathbf{x}^2\} \subset \mathbb{R}^2$:




$F(X) =$ set of fixed points of $W_{XX}$.

# The data polyhedron $\mathfrak{B}(\mathbf{v}, \mathbf{u}) \cap F(X)$

- Let $\mathbf{v}^\ell = \mathbf{W}^\ell_{XX}$ and $\mathbf{u}^\ell = \mathbf{M}^\ell_{XX}$.
- Set $\mathbf{u} = \bigvee_{\xi=1}^k \mathbf{x}^\xi$ and $\mathbf{v} = \bigwedge_{\xi=1}^k \mathbf{x}^\xi$.
- Set $\mathbf{w}^j = u_j + \mathbf{v}^j$ and $\mathbf{m}^j = v_j + \mathbf{u}^j$.
- Define $W = \{\mathbf{w}^1, \ldots, \mathbf{w}^n\}$ and $M = \{\mathbf{m}^1, \ldots, \mathbf{m}^n\}$
- $W$ is affinely independent whenver $\mathbf{w}^\ell \neq \mathbf{w}^j$ $\forall \ell \neq j$. Similarly for $M$.

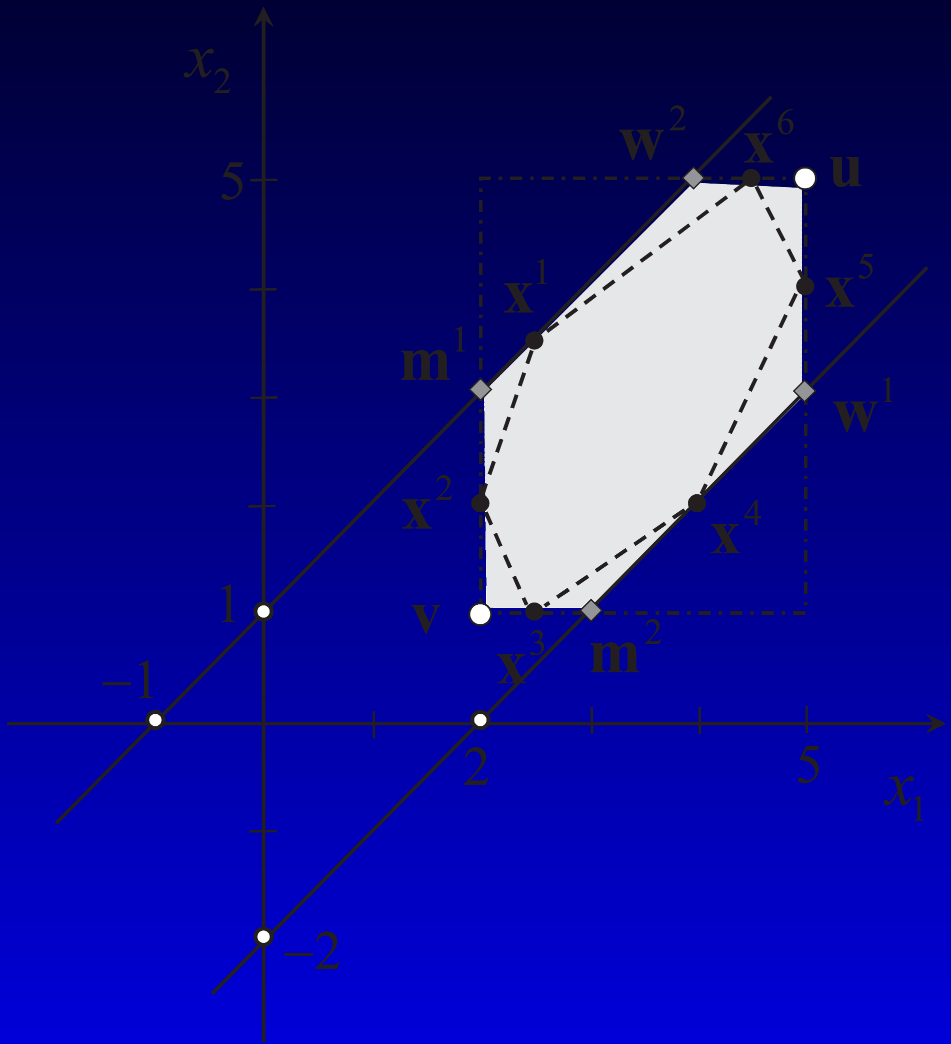
# The data polyhedron $\mathfrak{B}(\mathbf{v}, \mathbf{u}) \cap F(X)$

- Let $\mathfrak{B}(\mathbf{v}, \mathbf{u})$ denote the hyperbox determined by $\{\mathbf{v}, \mathbf{u}\}$.
- We obtain $X \subset C(X) \subset \mathfrak{B}(\mathbf{v}, \mathbf{u}) \cap F(X)$.
- The vertices of the polyhedron

$$\mathfrak{B}(\mathbf{v}, \mathbf{u}) \cap F(X)$$

are the elements of $W \cup M \cup \{\mathbf{v}, \mathbf{u}\}$

# The data polyhedron $\mathfrak{B}(\mathbf{v}, \mathbf{u}) \cap F(X)$




The fixed point set $F(X)$ is the infinite strip bounded by the two lines of slope 1.

# Rationale for Dendritic Computing

- The number of synapses on a *single* neuron in the cerebral cortex ranges between 500 and 200,000.
- A neuron in the cortex typically sends messages to approximately $10^4$ other neurons.
- Dendrites make up the *largest component* in both surface area and volume of the brain.
- Dendrites of cortical neurons make up $> 50\%$ of the neuron's membrane.

# Rationale for Dendritic Computing

- Recent research results demonstrate that the dynamic interaction of inputs in dendrites containing voltage-sensitive ion channels make them capable of realizing nonlinear interactions, logical operations, and possibly other local domain computation (Poggio, Koch, Shepherd, Rall, Segev, Perkel, et.al.)
- Based on their experimentations, these researchers make the case that it is the *dendrites* and not the neural cell bodies *are the basic computational units of the brain.*

# Our LNNs Are Based On Biological Neurons

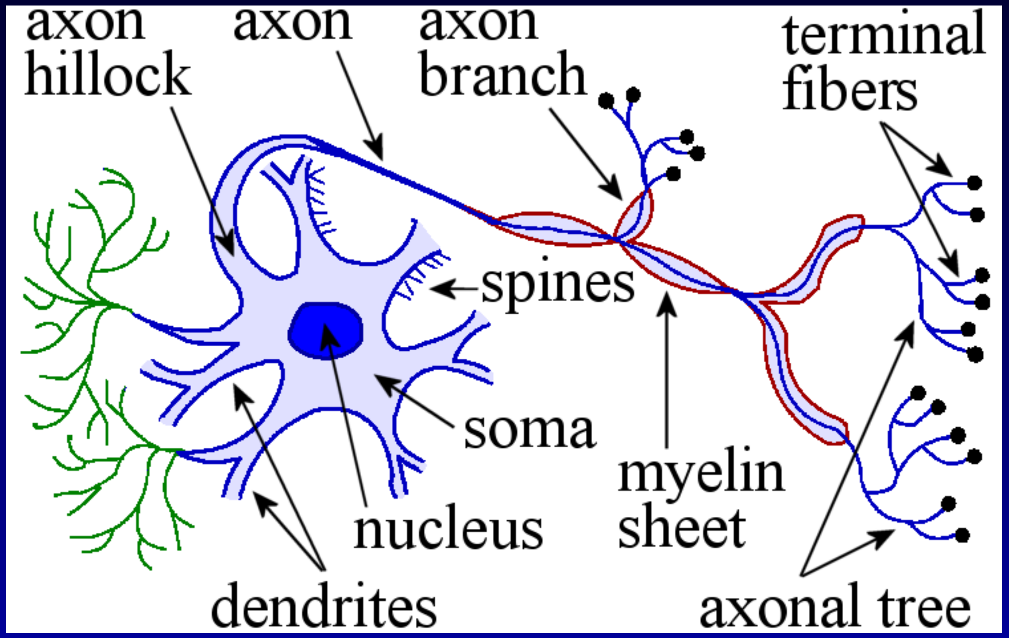
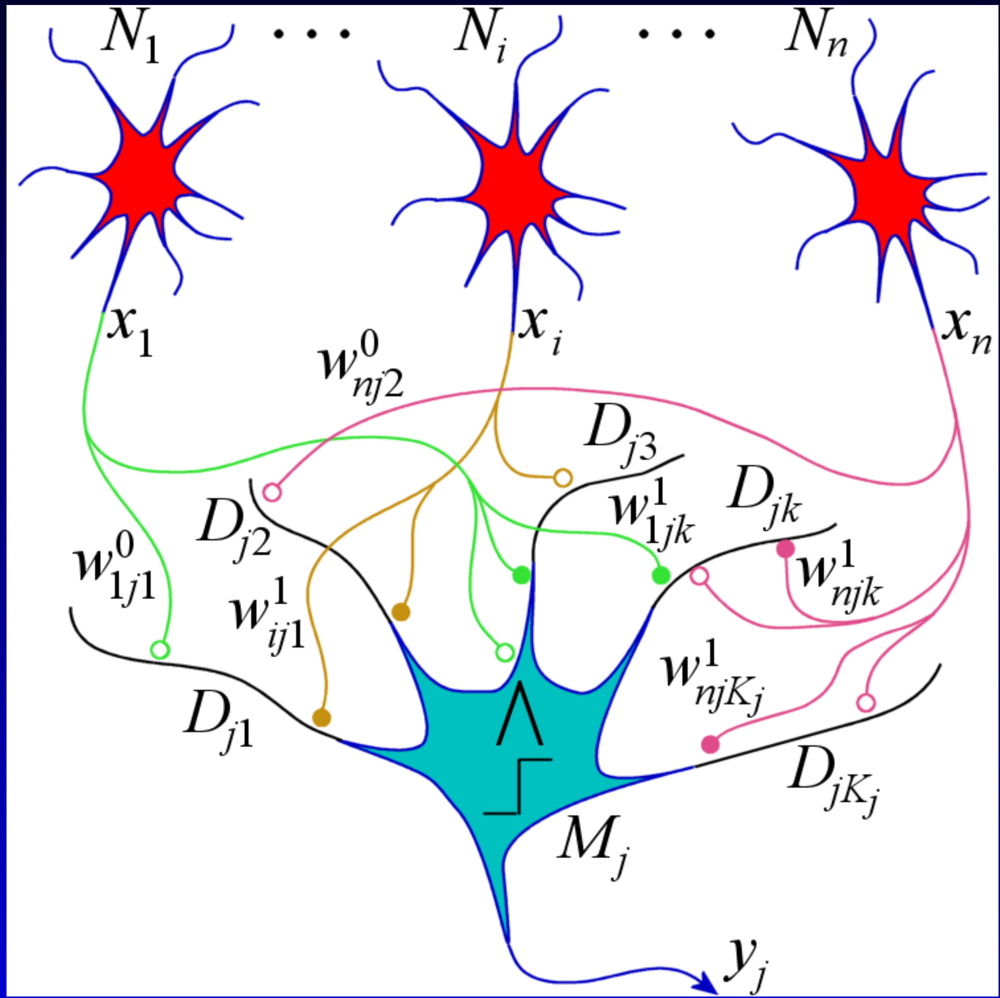


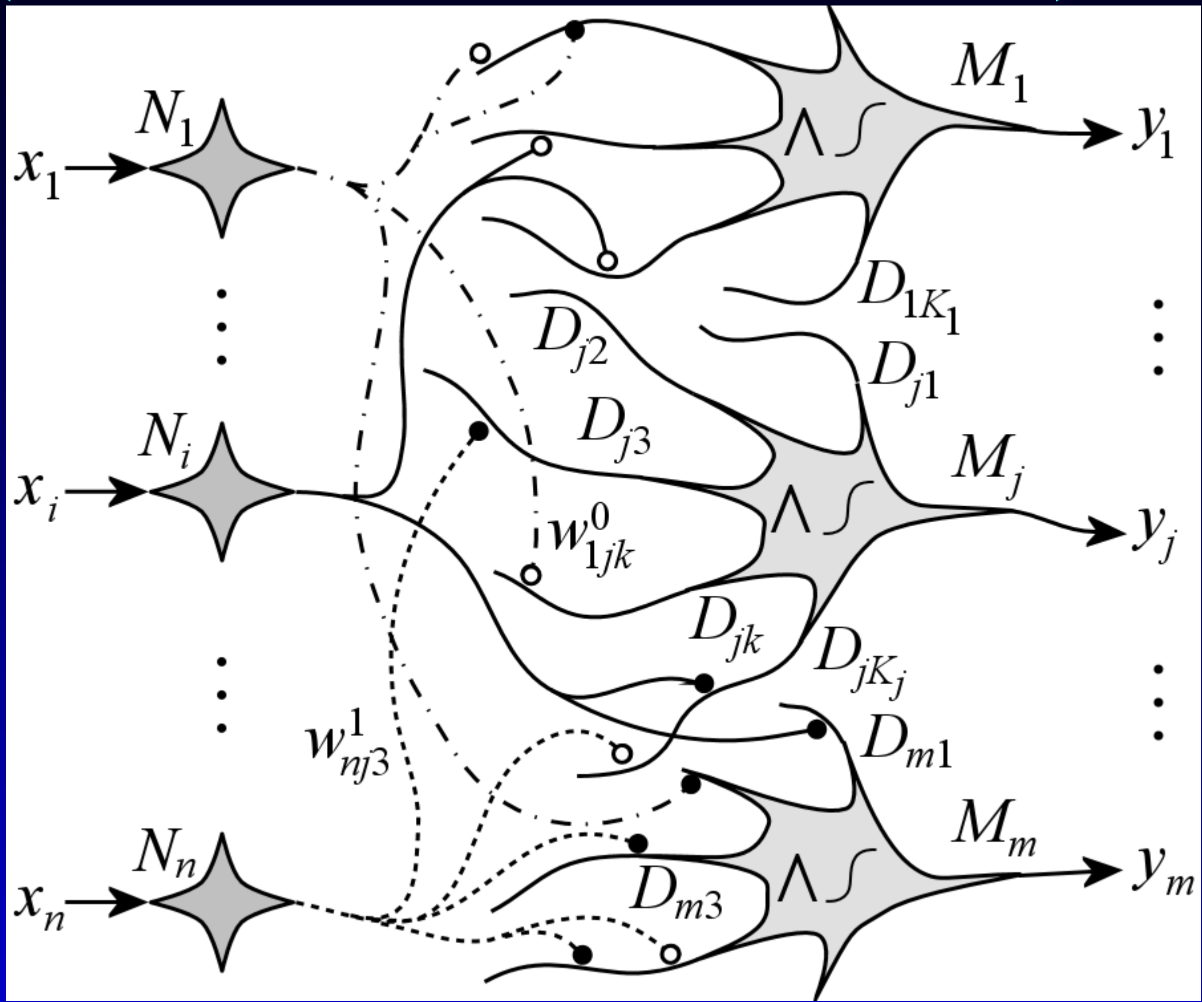

Figure 1: Simplified sketch of the processes of a biological neuron.

# Dendritic Computation: Graphical Model




$w^{\ell}_{ijk}$ = synaptic weight from the $N_i$ to the $k$th dendrite of $M_j$; $\ell = 0$ for inhibition and $\ell = 1$ for excitation.

# SLLP (with Dendritic Structures)




Graphical representation of a single-layer lattice based perceptron with dendritic structure.

# Dendritic LNN model

- In the dendritic ANN model, a neuron $M_j$ has $K_j$ dendrites. A given dendrite $D_{jk}$ $(k \in \{1, \ldots, K_j\})$ of $M_j$ receives inputs from axonal fibers of neurons $N_1, \ldots, N_n$ and computes a value $\tau_k^j$.
- The neuron $M_j$ computes a value $\tau^j$ which will correspond to the maximum (or minimum) of the values $\tau_1^j, \ldots, \tau_{K_j}^j$ received from its dendrites.

# Dendritic Computation: Mathematical Model

The computation performed by the $k$th dendrite for input $\mathbf{x} = (x_1, \ldots, x_n)' \in \mathbb{R}^n$ is given by

$$\tau_k^j(\mathbf{x}) = p_{jk} \bigwedge_{i \in I(k)} \bigwedge_{\ell \in L(i)} (-1)^{1-\ell} \left( x_i + w_{ijk}^{\ell} \right) \ ,$$

where

- $x_i$ – value of neuron $N_i$;
- $I(k) \subseteq \{1, \ldots, n\}$ – set of all input neurons with terminal fibers that synapse on dendrite $D_{jk}$;
- $L(i) \subseteq \{0, 1\}$ – set of terminal fibers of $N_i$ that synapse on dendrite $D_{jk}$;
- $p_{jk} \in \{-1, 1\}$ – IPSC/EPSC of $D_{jk}$.

# Dendritic Computation: Mathematical Model

- The value $\tau_k^j(\mathbf{x})$ is passed to the cell body and the state of $M_j$ is a function of the input received from all its dendritic postsynaptic results. The total value received by $M_j$ is given by

$$\tau^j(\mathbf{x}) = p_j \bigwedge_{k=1}^{K_j} \tau_k^j(\mathbf{x}).$$

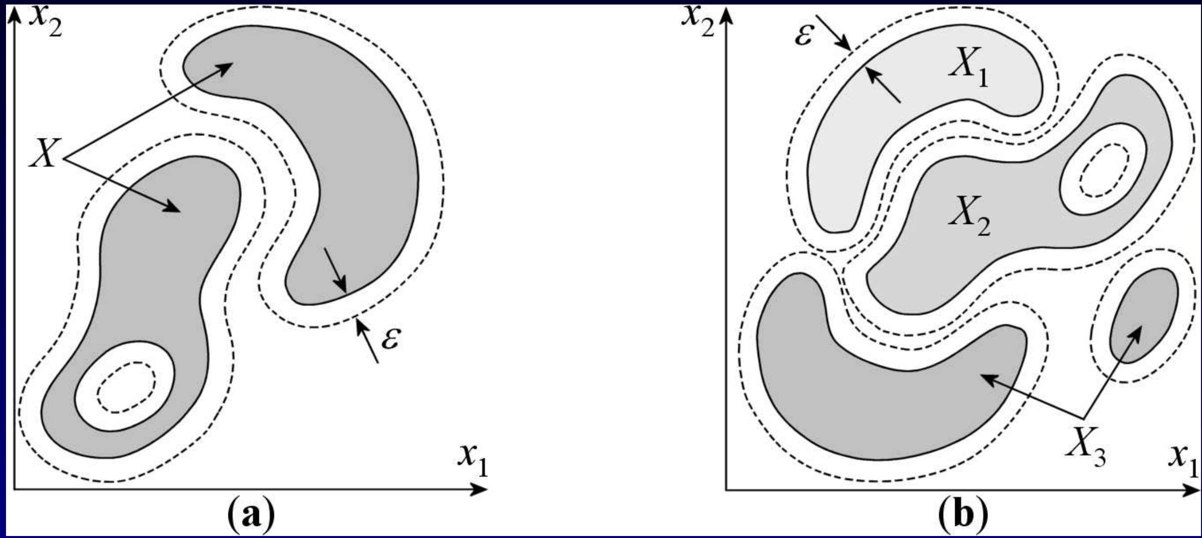
# The Capabilities of an SLLP

- An SLLP can distiguish between any given number of pattern classes to within any desired degree of $\varepsilon > 0$.
- More precisely, suppose $X_1$, $X_2$, $\ldots$, $X_m$ denotes a collection of disjoint compact subsets of $\mathbb{R}^n$.
- For each $p \in \{1, \ldots, m\}$, define
  $Y_p = \bigcup_{j=1, j \neq p}^{m} X_j$
  $\varepsilon_p = \mathrm{d}(X_p, Y_p) > 0$
  $\varepsilon_0 = \frac{1}{2} \min\{\varepsilon_1, \ldots, \varepsilon_p\}$.
- As the following theorem shows, a given pattern $\mathbf{x} \in \mathbb{R}^n$ will be recognised correctly as belonging to class $C_p$ whenever $\mathbf{x} \in X_p$

# The Capabilities of an SLLP

- **Theorem.** *If* $\{X_1, X_2, \ldots, X_m\}$ *is a collection of disjoint compact subsets of* $\mathbb{R}^n$ *and* $\varepsilon$ *a positive number with* $\varepsilon < \varepsilon_0$*, then there exists a single layer lattice based perceptron that assigns each point* $\mathbf{x} \in \mathbb{R}^n$ *to class* $C_j$ *whenever* $\mathbf{x} \in X_j$ *and* $j \in \{1, \ldots, m\}$*, and to class* $C_0 = \neg \bigcup_{j=1}^m C_j$ *whenever* $\mathrm{d}(\mathbf{x}, X_i) > \varepsilon$*,* $\forall i = 1, \ldots, m$*. Furthermore, no point* $\mathbf{x} \in \mathbb{R}^n$ *is assigned to more than one class.*

# Illustration of the Theorem in $\mathbb{R}^2$

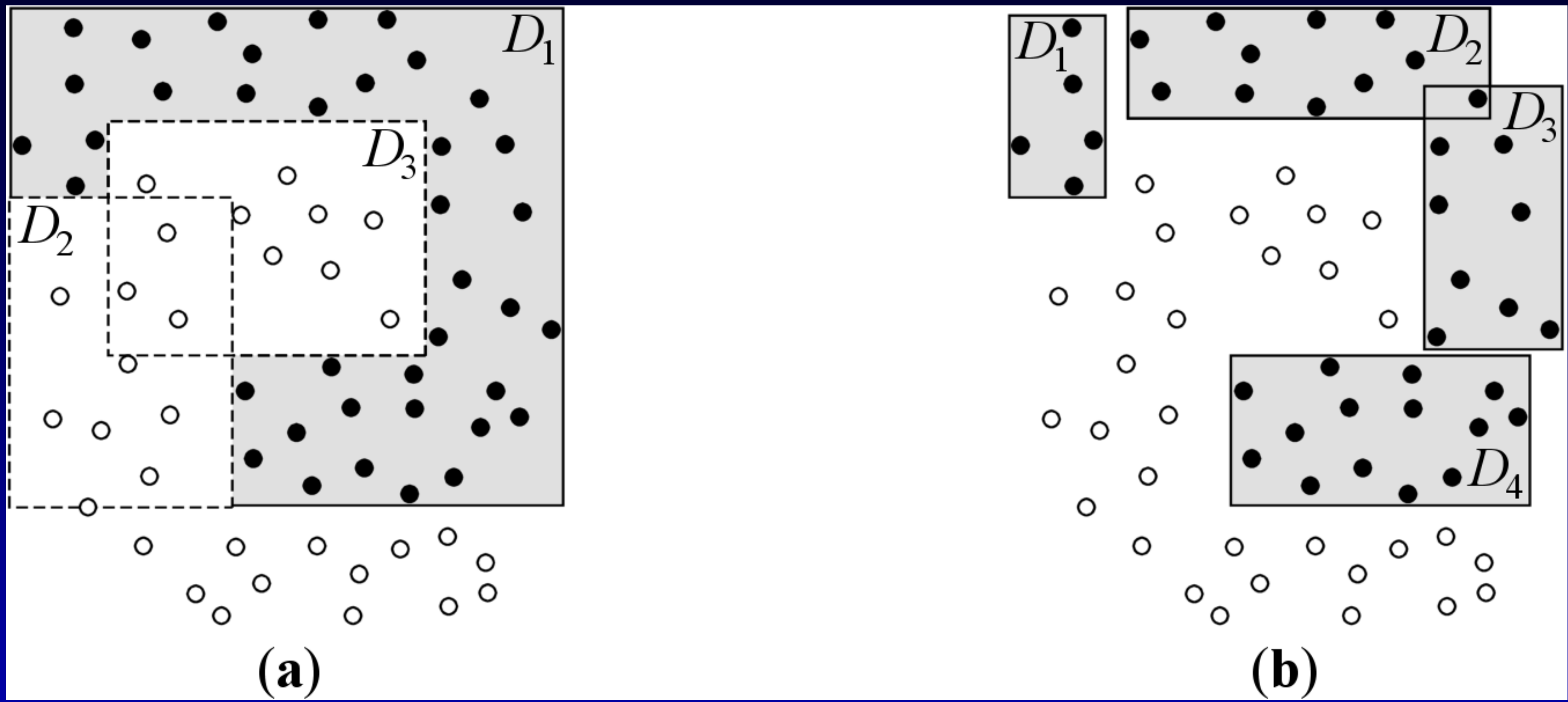



Any point in the set $X_j$ is identified with class $C_j$; points within the $\epsilon$-band may or may not be classified as belonging to $C_j$, points outside the $\epsilon$-bands will not be associated with a class $C_j$ $\forall j$.

# Learning in LNNs

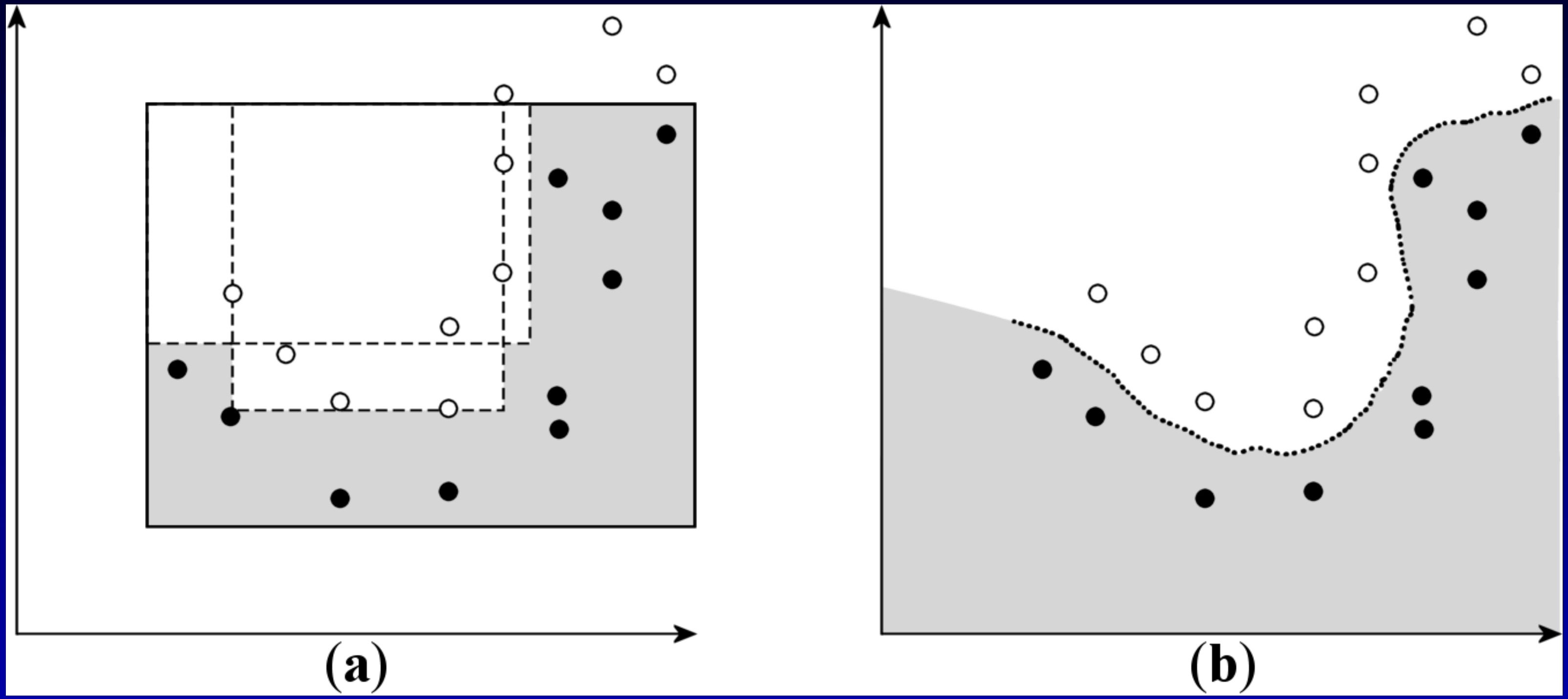
- Early training methods were based on the proofs of the preceding Theorems.
- All training algorithms involve the growth of axonal branches, computation of branch weights, creation of dendrites, and synapses.
- The first training algorithm developed was based on elimination of foreign patterns from a given training set (min or intersection).
- The second training algorithm was based on small region merging (max or union).

# Example of the two methods in $\mathbb{R}^2$



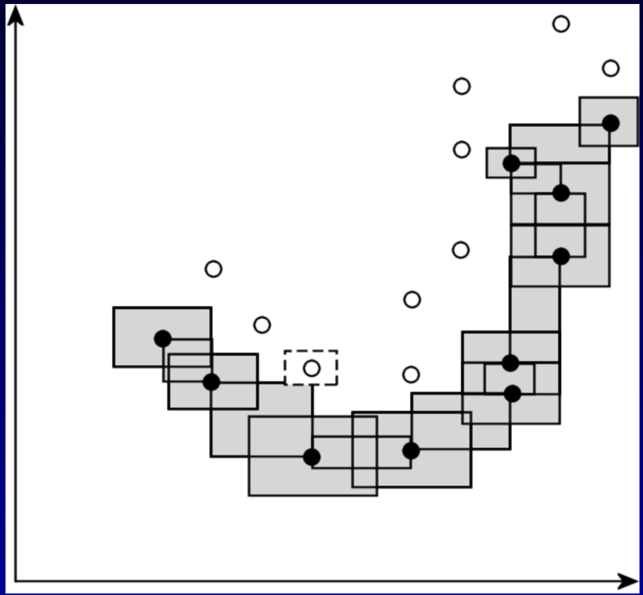

The two methods partition the pattern space $\mathbb{R}^2$ in terms of intersection (**a**) and union (**b**), respectively.

# SLLP Using Elemination VS MLP



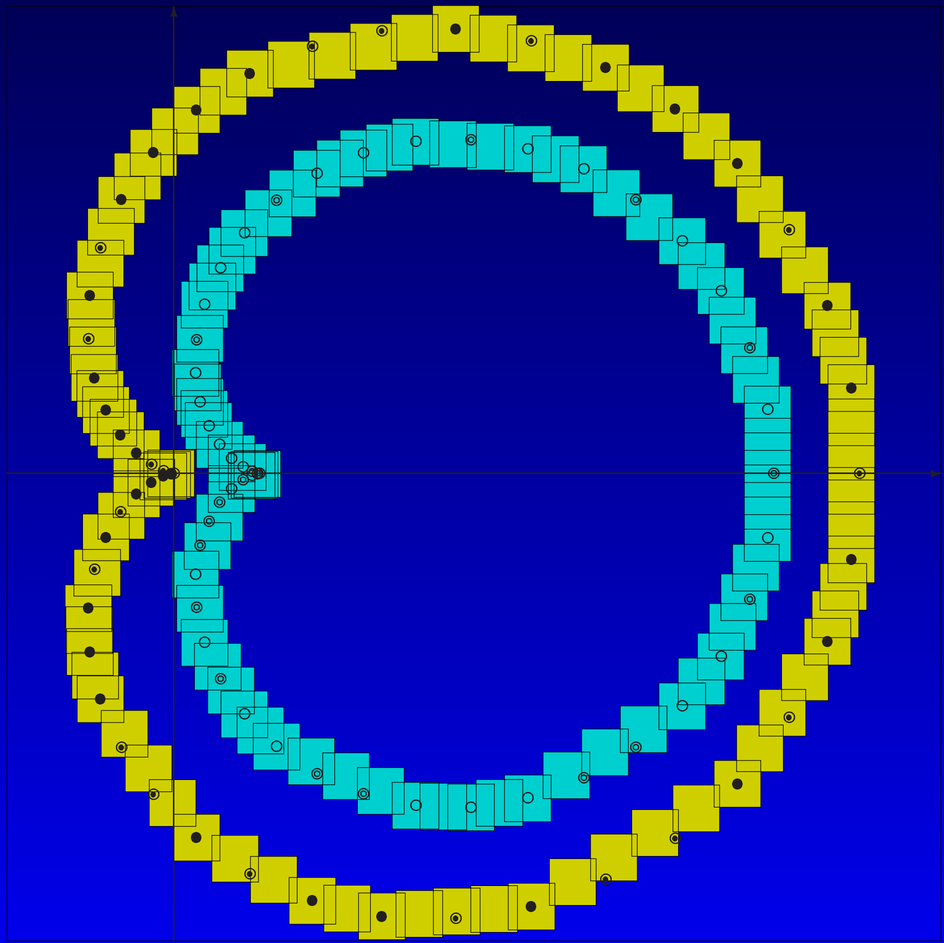

(**a**) SLLP: 3 dendrites, 9 axonal branches. (**b**) MLP 13 hidden neorons and 2000 epochs.

# SLLP Using Merging



During training, the SLLP grows 20 dendrites, 19 excitatory and 1 inhibitory (*dashed*).

# Another Merging Example



# Learning in LNNs

- L. Iancu developed a hybrid method using both Merging and Elimination. The method is reminiscent of the Expansion-Contraction method for hyperboxes developed by P.K. Simson for training Mini-Max Neural Networks, but it is distinctly different.
- L. Iancu also extended this learning to Ritter's Fuzzy SLLP

# Fuzzy LNNs

$\mathrm{The Problem}:$

- Classify all points in the interval $[a, b] \subset \mathbb{R}$ as belonging to class $C_1$, and every point outside the interval $[a - \alpha, b + \alpha]$ as having no relation to class $C_1$, where $\alpha > 0$ is a specified fuzzy boundary parameter.
- For a point $x \in [a - \alpha, a]$ or $x \in [b, b + \alpha]$ we would like $y(x)$ to be close to $1$ when $x$ is close to $a$ or $b$, and $y(x)$ close to $0$ whenever $x$ is close to $a - \alpha$ or $b + \alpha$.

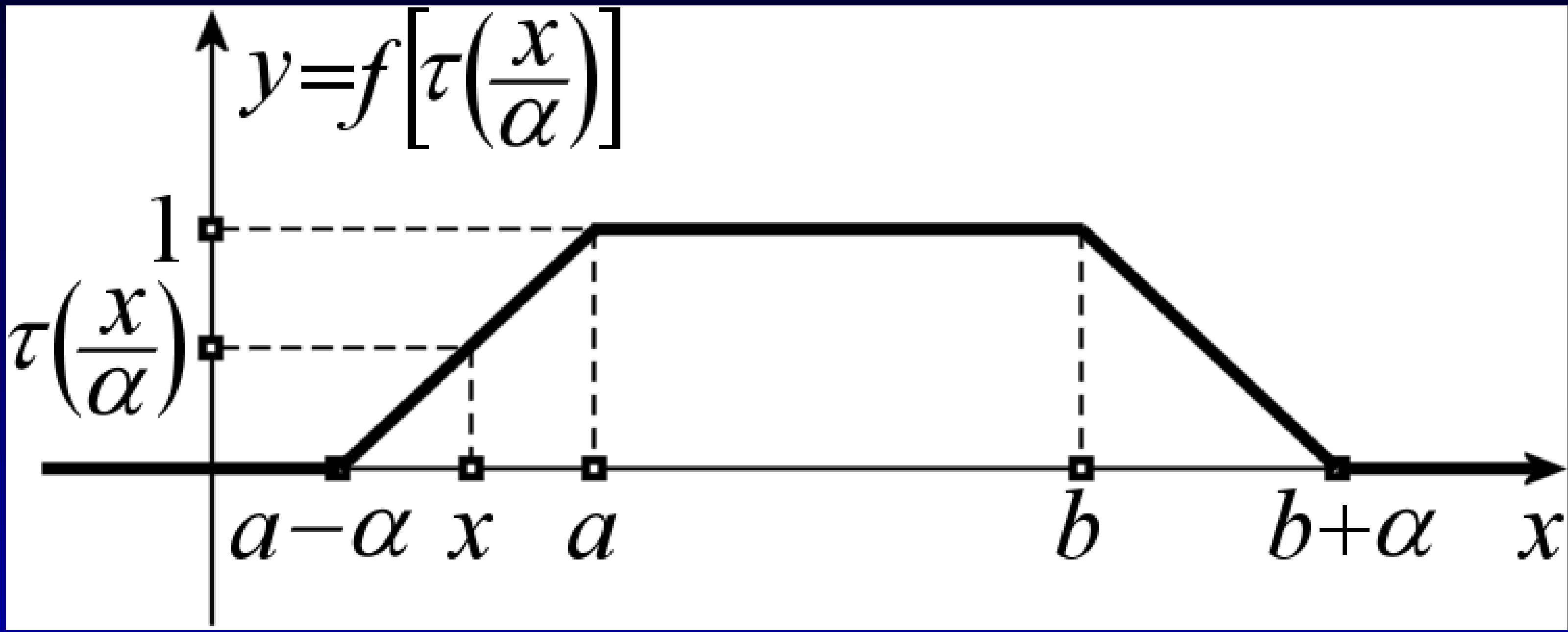
# Fuzzy LNNs

**Solution** :

- Change the weights $w_1^0 = -b$ and $w_1^1 = -a$ found by one of the previous algorithms to $v_1^0 = -\frac{w_1^0}{\alpha} - 1$ and $v_1^1 = -\frac{w_1^1}{\alpha} + 1$, and use the input $\frac{x}{\alpha}$ instead of $x$.
- Use the activation function

$$f(z) = \begin{cases} 1 & \text{if } z \geq 1 \\ z & \text{if } 0 \leq z \leq 1 \\ 0 & \text{if } z \leq 0 \end{cases} .$$

# Fuzzy LNNs




Computing fuzzy output values with an SLLP using the ramp activation function.

# Learning in LNNs

| Classifier | Recognition |
|---|---|
| SLLP (elimination) | 98.0% |
| Backpropagation | 96% |
| Resilient Backpropagation | 96.2% |
| Bayesian Classifier | 96.8% |
| Fuzzy LNN | 100% |

UC Irvine Ionosphere data set (2-class problem in $\mathbb{R}^{34}$ with training set = 65% of data set)

# Learning in LNNs

| Classifier | Recognition |
|---|---|
| Fuzzy SLLP (merge/elimination) | 98.7% |
| Backpropagation | 95.2% |
| Fuzzy Min-Max NN | 97.3% |
| Bayesian Classifier | 97.3% |
| Fisher Ratios Discimination | 96.0% |
| Ho-Kashyap | 97.3% |

Fisher's Iris Data Set. A 3-class problem in $\mathbb{R}^4$ with training set = 50% of data set.

# Learning in LNNs

- A. Barmpoutis extended the elimination method to arbitrary orthonormal basis settings.
- A dynamic Backpropagation Method is currently under development.

# Learning in LNNs

In Barmpoutis's approach, the equation

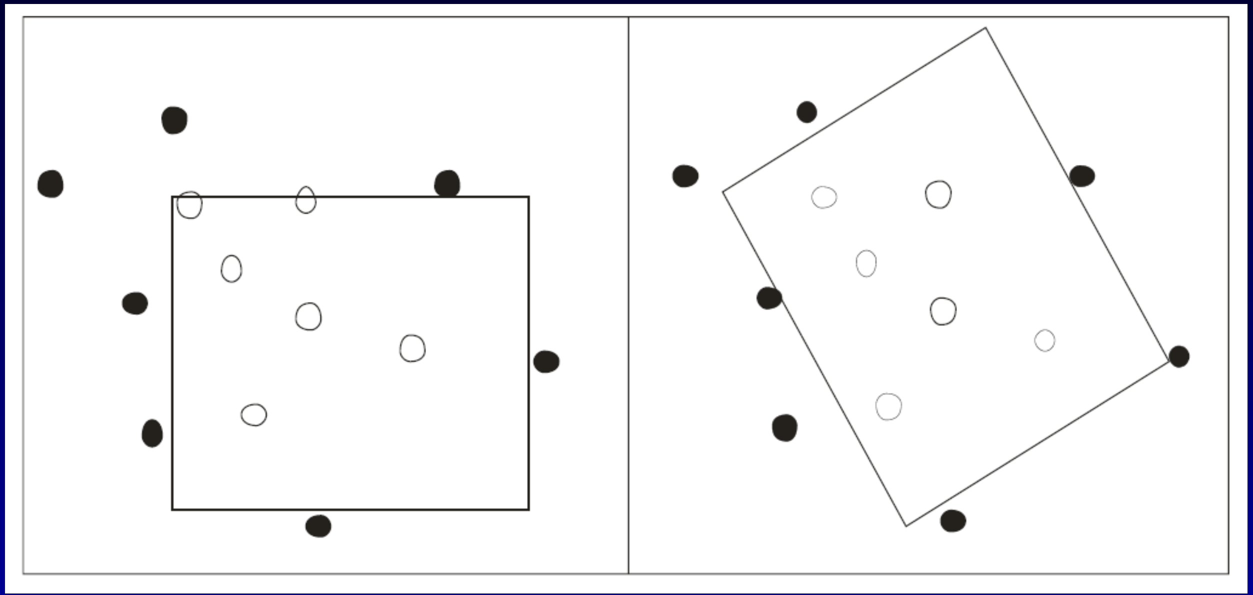
$$\tau_k^j(\mathbf{x}) = p_{jk} \bigwedge_{i \in I(k)} \bigwedge_{\ell \in L(i)} (-1)^{1-\ell} \left( x_i + w_{ijk}^{\ell} \right) \ ,$$

is replaced by

$$\tau_k^j(\mathbf{x}) = p_{jk} \bigwedge_{i \in I(k)} \bigwedge_{\ell \in L(i)} (-1)^{1-\ell} \left( R(\mathbf{x})_i + w_{ijk}^{\ell} \right) \ ,$$

where $R$ is a rotation matrix obtained in the learning process.

# LNNs employing Orthonormal Basis

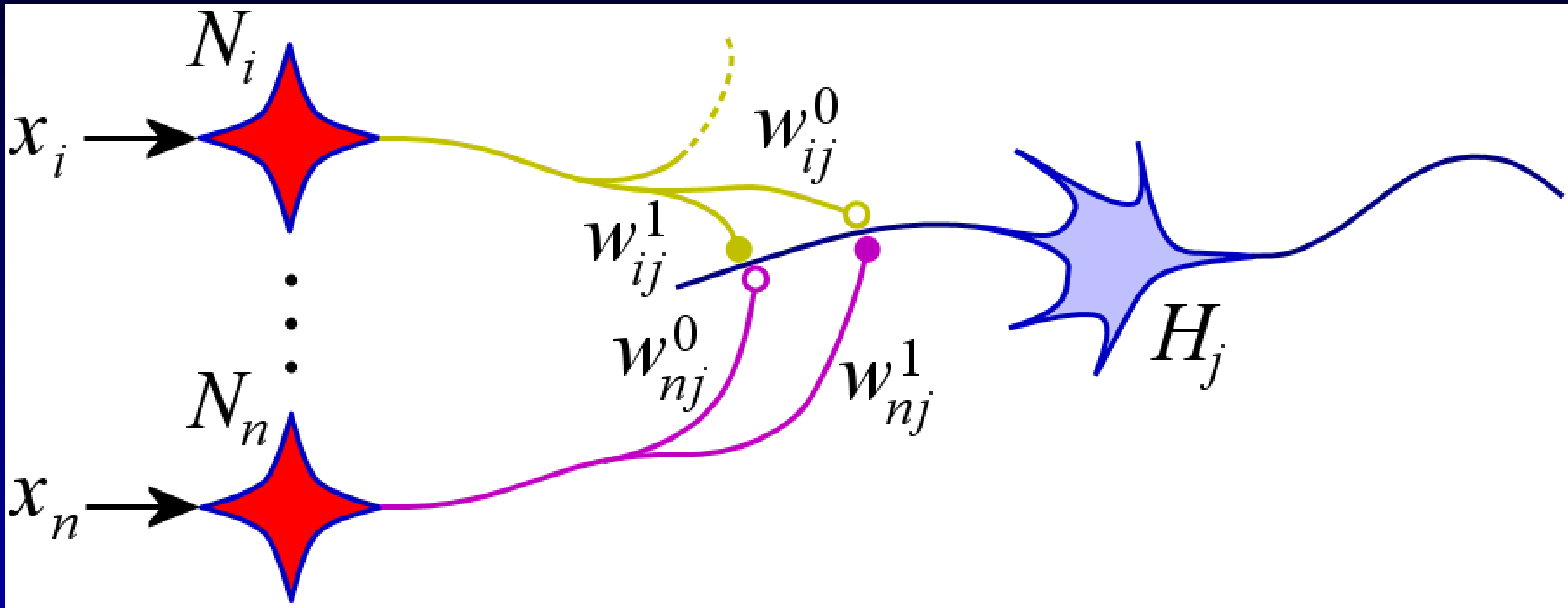


Left: Maximal hyperbox for elimination in the standard basis for $\mathbb{R}^n$.
Right: Maximal hyperbox for elimination in another orthonormal basis for $\mathbb{R}^n$.

# Dendritic Model of an Associative Memory

- $X = \{\mathbf{x}^1, \dots, \mathbf{x}^k\} \subset \mathbb{R}^n$.
- $n$ input neurons $N_1, \dots, N_n$ accepting input $\mathbf{x} = (x_1, \dots, x_n)' \in \mathbb{R}^n$, where $x_i \to N_i$.
- One hidden layer containing $k$ neurons $H_1, \dots, H_k$.
- Each neuron $H_j$ has exactly one dendrite which contains the synaptic sites of exactly two terminal axonal fibers of $N_i$ for $i = 1, \dots, n$.
- The weights of the two terminal fibers of $N_i$ making contact with the dendrite of $H_j$ are denoted by $w_{ij}^{\ell}$, with $\ell = 0, 1$.

# Input and Hidden Layer Neural Connection




Every input neuron connects to the dendrite of each hidden neuron with two axonal fibers, one excitatory and the other inhibitory.

# Computation at the Hidden Layer

- For input $\mathbf{x} \in \mathbb{R}^n$, the dendrite of $H_j$ computes

$$\tau^j(\mathbf{x}) = \bigwedge_{i=1}^{n} \bigwedge_{\ell=0}^{1} (-1)^{1-\ell} \left(x_i + w_{ij}^{\ell}\right).$$

- The state of neuron $H_j$ is determined by the hard-limiter activation function

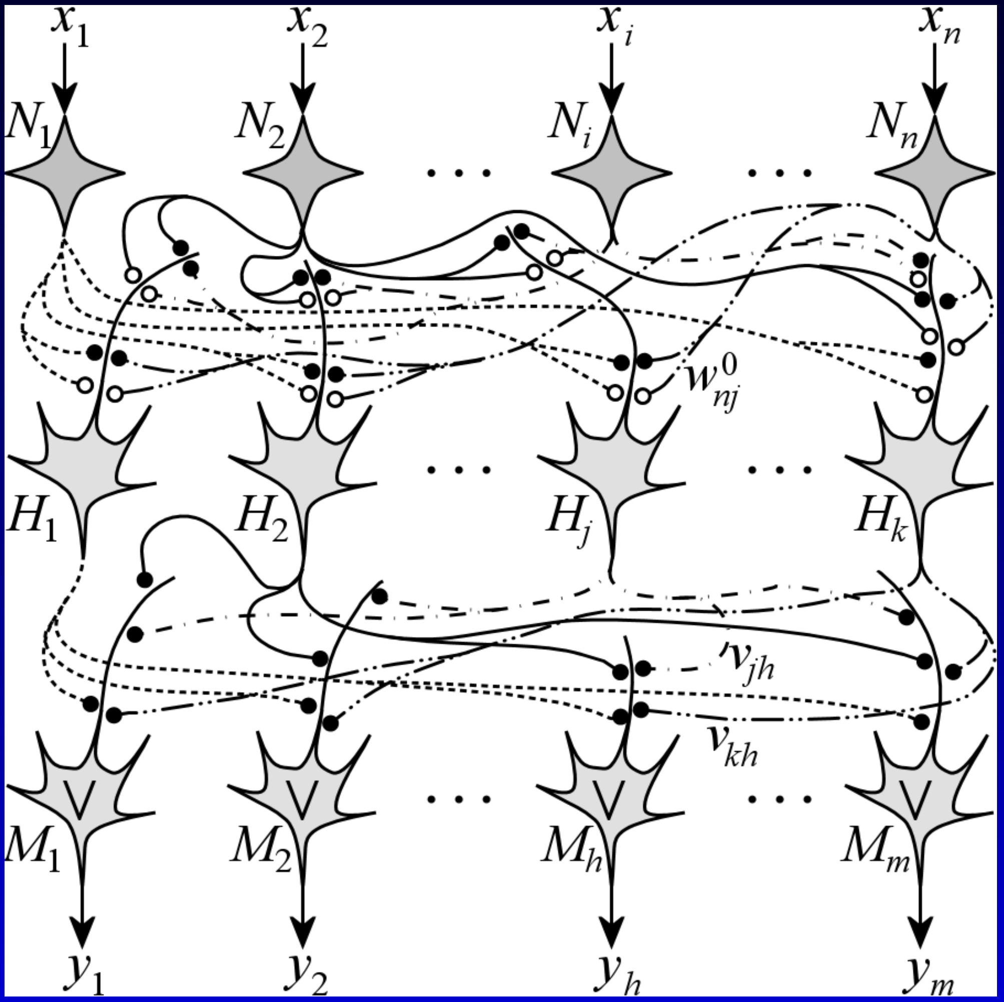
$$f(z) = \begin{cases} 0 & \text{if } z \geq 0 \\ -\infty & \text{if } z < 0 \end{cases}.$$

- The output of $H_j$ is $f\left[\tau^j\left(\mathbf{x}\right)\right]$.
- The output flows along the axon of $H_j$ and its axonal fibers to $m$ output neurons $M_1, \ldots, M_m$.

# Computation at the Output Layer

- Each output neuron $M_h$, $h = 1, \ldots, m$, has exactly one dendrite.
- Each hidden neuron $H_j$ ($j = 1, \ldots, k$) has exactly one excitatory axonal fiber terminating on the dendrite of $M_h$.
- The synaptic weight of the excitatory axonal fiber of $H_j$ terminating on the dendrite of $M_h$ is preset as $v_{jh} = y_h^j$ for $j = 1, \ldots, k$; $h = 1, \ldots, m$.
- The computation performed by $M_h$ is $\tau^h(\mathbf{q}) = \bigvee_{j=1}^{k} \left( q_j + v_{jh} \right)$, where $q_j = f\left[\tau^j(\mathbf{x})\right]$ denotes the output of $H_j$.
- The activation function for each output neuron $M_h$ is the identity function $g(z) = z$.

# Dendritic Model of an Associative Memory

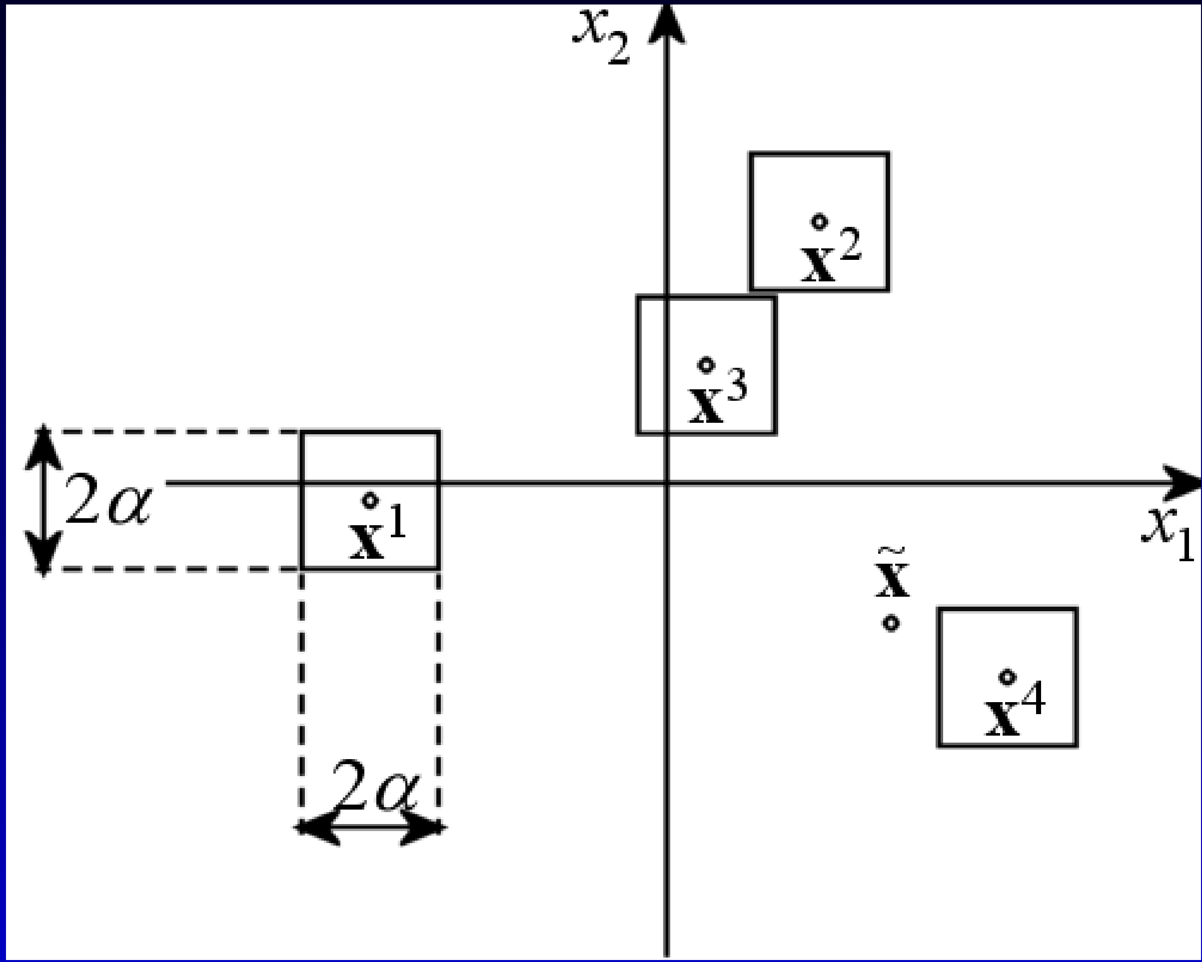



Topology of the dendritic associative memory based on the dendritic model. The network is fully connected.

# Computation of the Weights $w_{ij}^{\ell}$

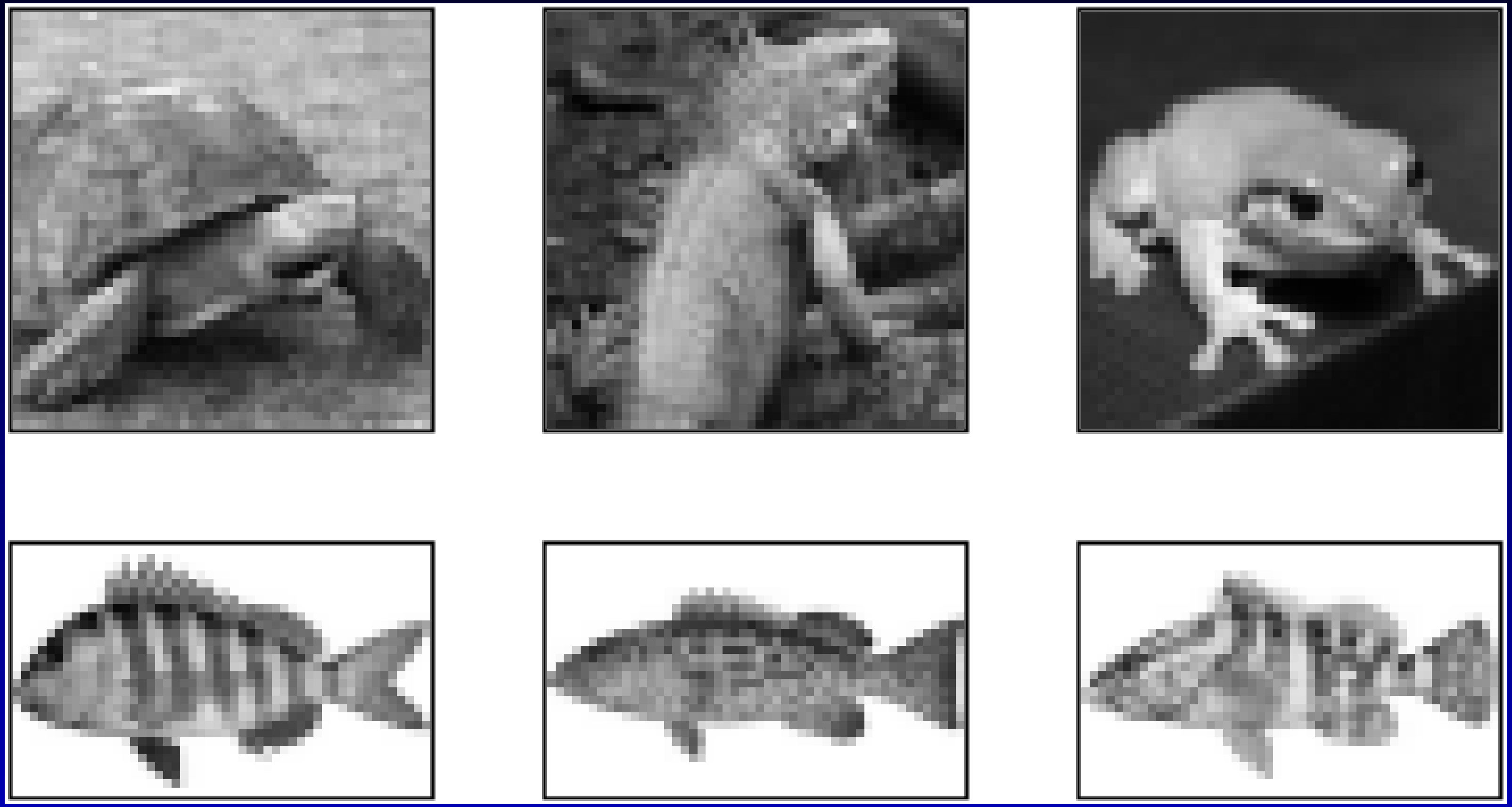
- Compute
  $d(\mathbf{x}^{\xi}, \mathbf{x}^{\gamma}) = \max\left\{\left|x_i^{\xi} - x_i^{\gamma}\right| : i = 1, \ldots, n\right\}$.
- Choose a noise parameter $\alpha > 0$ such that $\alpha < \frac{1}{2}\min\{ud(\mathbf{x}^{\xi}, \mathbf{x}^{\gamma}) : \xi < \gamma,\ \xi, \gamma \in \{1, \ldots, k\}\}$.
- Set $w_{ij}^{\ell} = \begin{cases} -\left(x_i^j - \alpha\right) & \text{if } \ell = 1 \\ -\left(x_i^j + \alpha\right) & \text{if } \ell = 0 \end{cases}$.
- Under these conditions, given input $\mathbf{x} \in \mathbb{R}^n$, the output $\mathbf{y} = (y_1, \ldots, y_m)'$ from the output neurons will be $\mathbf{y} = \left(y_1^j, \ldots, y_m^j\right)' = \mathbf{y}^j \iff \mathbf{x} \in B^j$, where $B^j = \left\{ (x_1, \ldots, x_n)' \in \mathbb{R}^n : x_i^j - \alpha \leq x_i \leq x_i^j + \alpha,\ i = 1, \ldots, n\right\}$.

# Patterns that will be correctly associated



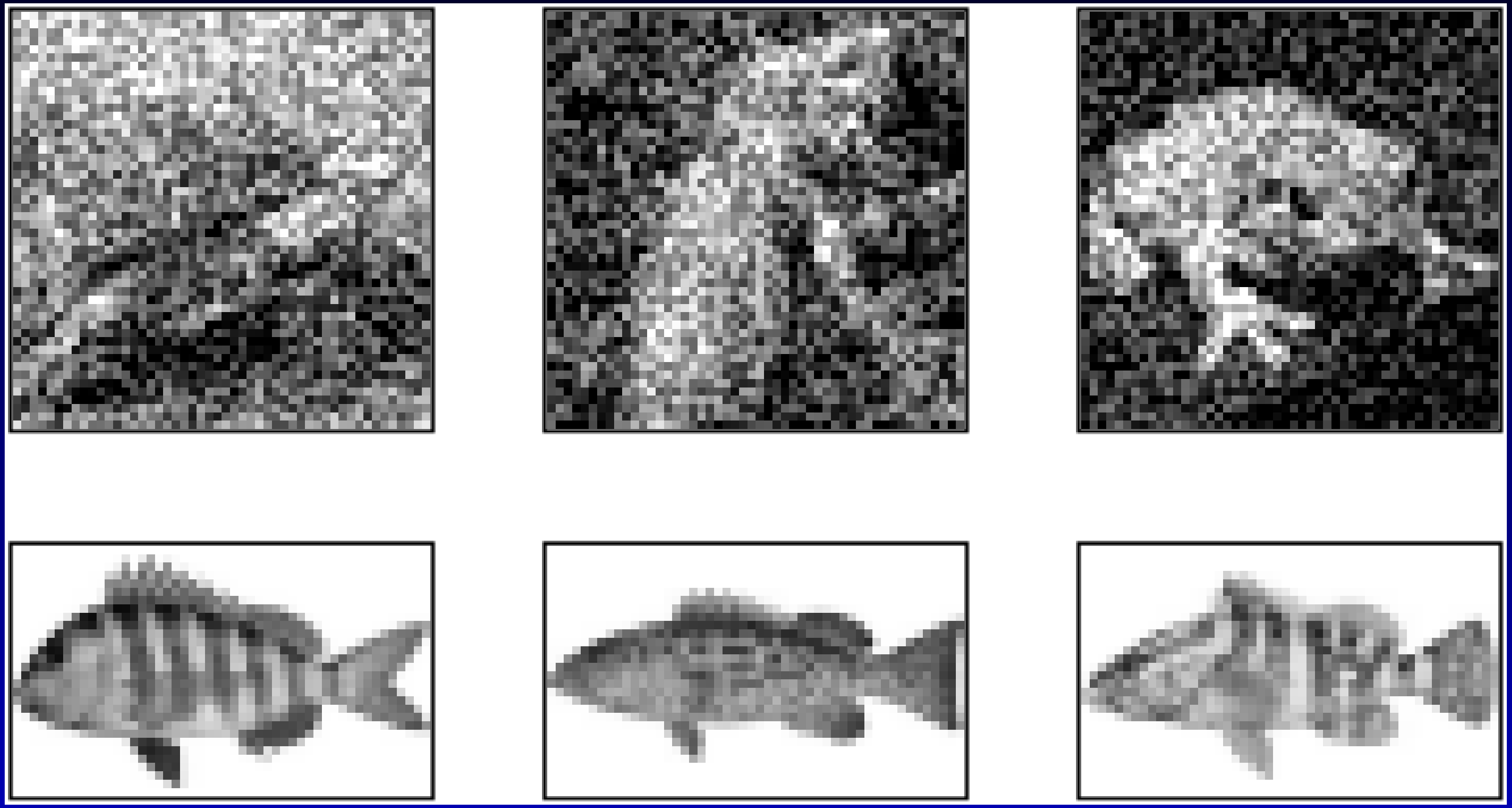

Any patter residing in the box with center $\mathbf{x}^\xi$ will be idntified as pattern $\mathbf{x}^\xi$. The pattern $\tilde{\mathbf{x}}$ will not be associated with any prototype pattern.

# Patterns to Store



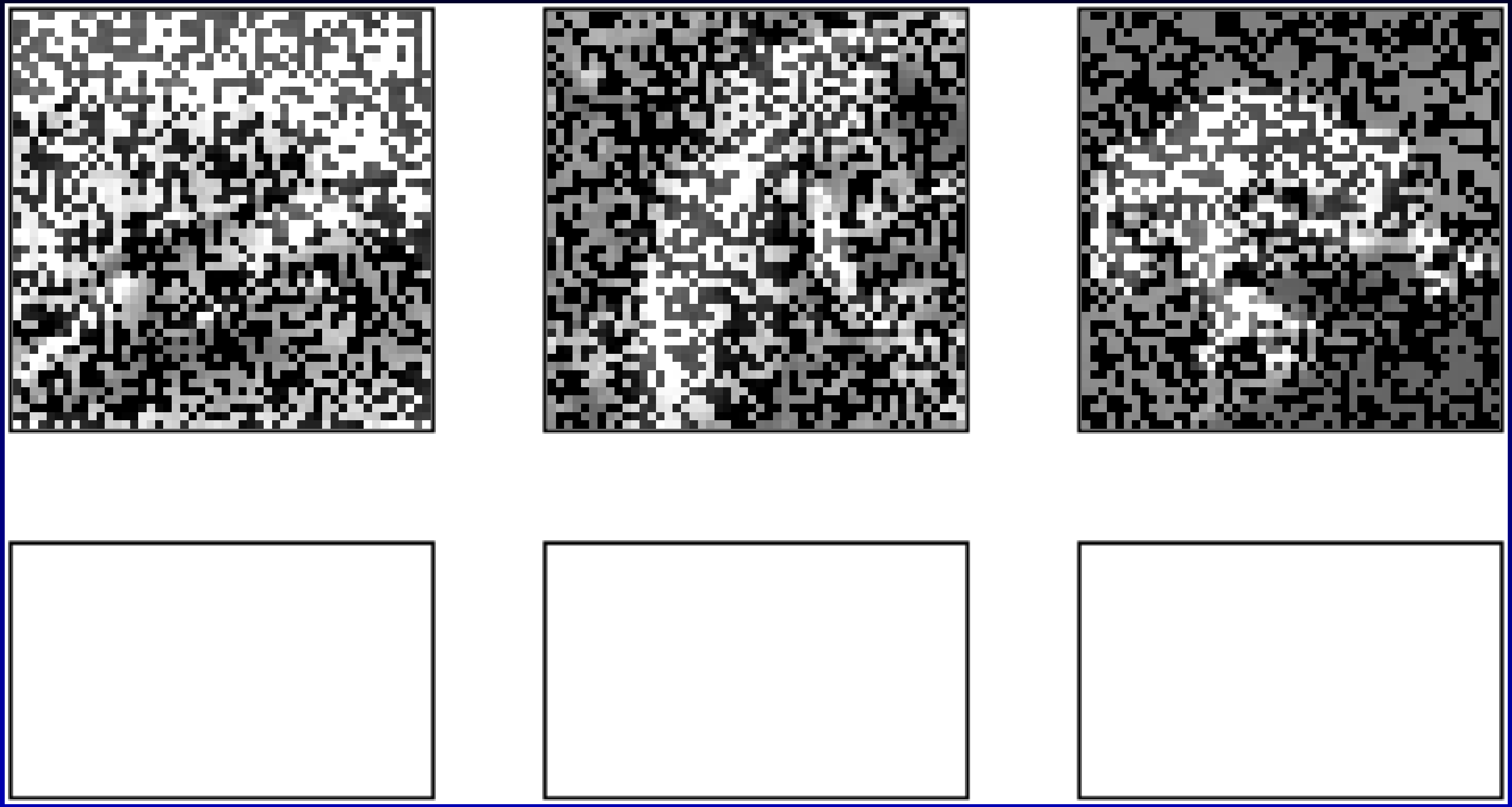
Top row represents the patterns $\mathbf{x}^1, \mathbf{x}^2$, and $\mathbf{x}^3$, while the bottom row depicts the associated patterns $\mathbf{y}^1, \mathbf{y}^2$, and $\mathbf{y}^3$. Here $n = 2500$ and $m = 1500$.

# Recall of Corrupted Patterns



Distorting every vector components of $\mathbf{x}^j$ with random noise within the range $[-\alpha, \alpha]$, with $\alpha = 75.2$ results in perfect recall association.

# Recall Failure when Noise Exceeds $\alpha$



The memory rejects the patterns if they are corrupted with random noise exceeding $\alpha = 75.2$.

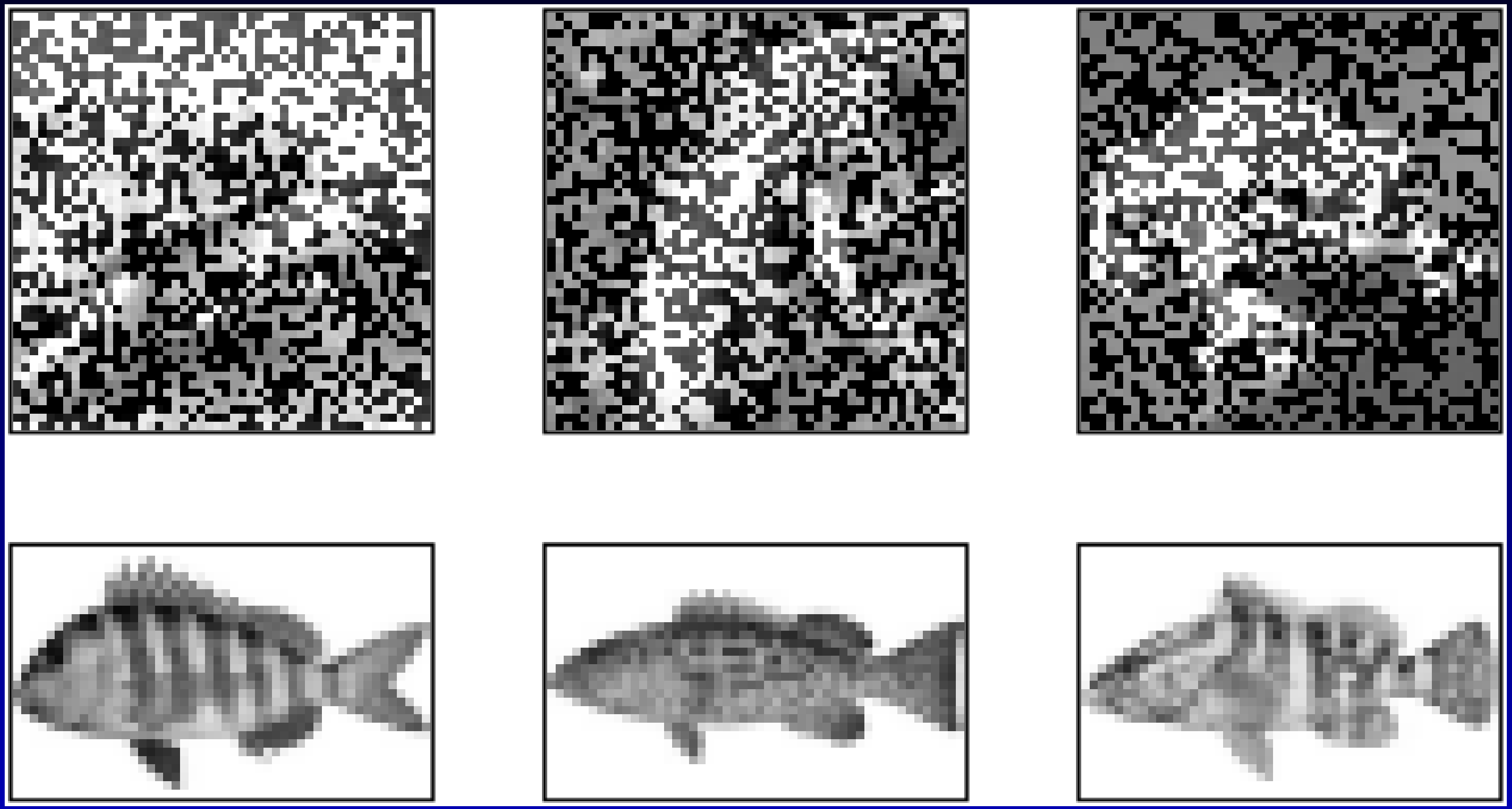
# Increasing the Noise Tolerance

- For each $\xi = 1, \ldots, k$ compute an allowable noise parameter $\alpha_\xi$ by setting $\alpha_\xi < \frac{1}{2} \min\{d(\mathbf{x}^\xi, \mathbf{x}^\gamma) : \gamma \in K(\xi)\}$, where $K(\xi) = \{1, \ldots, k\} \setminus \{\xi\}$.
- Reset the weights by

$$w_{ij}^{\ell} = \begin{cases} -\left(x_i^j - \alpha_j\right) & \text{if } \ \ell = 1\,, \\ -\left(x_i^j + \alpha_j\right) & \text{if } \ \ell = 0\,, \end{cases}$$

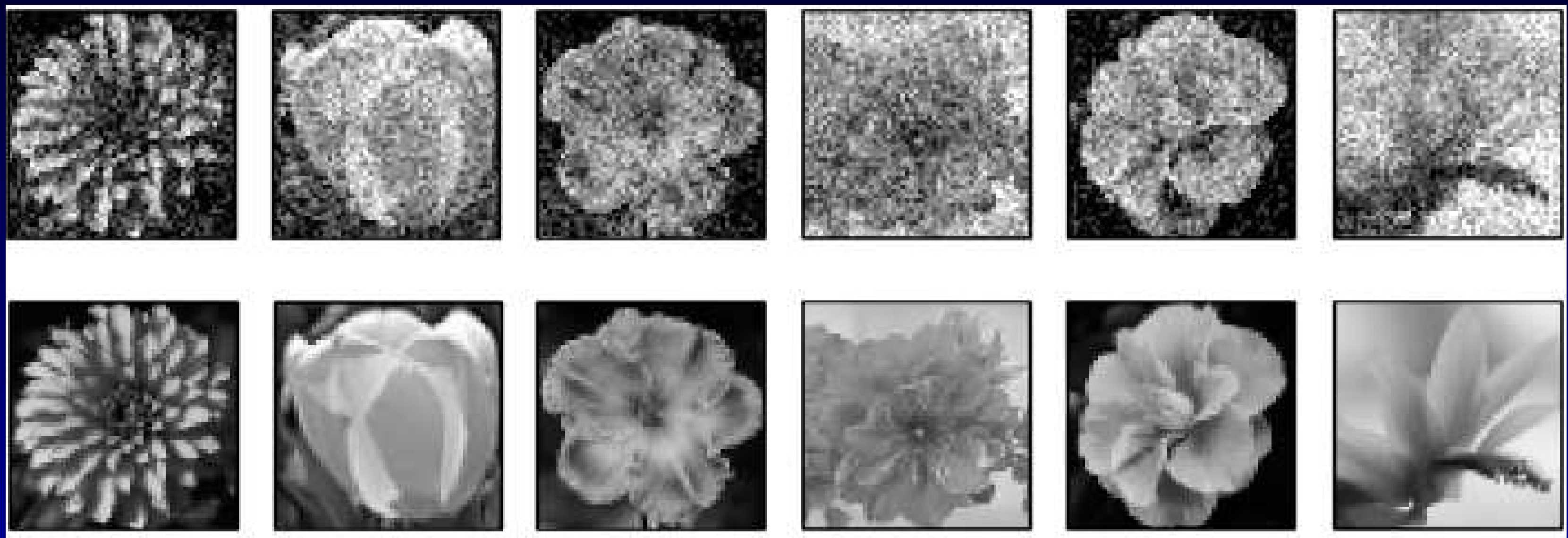
- Each output neuron $H_j$ will have a value $q_j = 0$ if and only if $\mathbf{x}$ is an element of the hypercube $B^j = \left\{\mathbf{x} \in \mathbb{R}^n : x_i^j - \alpha_j \leq x_i \leq x_i^j + \alpha_j\right\}$ and $q_j = -\infty$ whenever $\mathbf{x} \in \mathbb{R}^n \setminus B^j$.

# Successful Recall of the Refined Model



The top row shows the same input patterns as in the last figure. This time recall association is perfect.

# Recall of AAM based on the Dendritic Model



- Top row: patterns distorted with random noise within noise parameter $\alpha$.
- Bottom row: perfect recall of the auto-associative memory based on the dendritic model.

# A New LNN Model

- In this model the synapses on spines of dendrites are used. The presynaptic neuron is either excitatory or inhibitory, but not both.
- $N = \{N_i : i = 1, \ldots, n\}$ denotes the set of presynaptic (input) neurons.
- $\sigma(j, k) = j$th spine on on dendrite $D_k$
- $N(j, k) =$ set of presynaptic neurons with synapses on $\sigma(j, k)$. Thus, $N(j, k) \subset N$.
- $j_k =$ number of spines on $D_k$

# A New LNN Model

- The $k$th dendrite $D_k$ now computes

$$\tau_k = p_k \bigwedge_{j=1}^{j_k} [w_{kj} + \sum_{i \in N(j,k)} (-1)^{1-\ell(i)} s_i x_i],$$

where $\ell(i) = 0$ if $N_i$ is inhibitory and $\ell(i) = 1$ if $N_i$ is exitatory.

- $s_i =$ number of spikes in spike train produced by $N_i$ in an interval $[s - t, t]$
- Note that $\bigcup_{j=1}^{k} N(j,k)$ corresponds to the set of input neurons with terminal axonal fibers on $D_k$.

# Questions and Comments

- Thank you for your attention.
- Any questions or comments?