



DD2437 – Artificial Neural Networks and Deep Architectures (annda)

Lecture 5: Radial basis function NN and introduction to competitive learning

Pawel Herman

Computational Science and Technology (CST)

KTH Royal Institute of Technology

- Interpolation problem and RBFs
- RBF networks – hybrid learning
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

Lecture overview

- Interpolation problem and radial-basis functions (RBFs)
- RBF networks – hybrid learning
- Weight interpretation
- Competitive mechanisms for unsupervised learning

- Interpolation problem and RBFs
- RBF networks – hybrid learning
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

The separability of patterns

Cover's theorem

“A complex pattern classification problem, projected nonlinearly to a high-dimensional space, is more likely to be linearly separable than in low dimensional space, especially if it is not populated too densely.”

Cover, 1965

- Interpolation problem and RBFs
- RBF networks – hybrid learning
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

The separability of patterns

Cover's theorem

“A complex pattern classification problem, projected nonlinearly to a high-dimensional space, is more likely to be linearly separable than in low dimensional space, especially if it is not populated too densely.”

Cover, 1965

So, we need to have:

- 1) Many nonlinear mappings $\varphi_i(\mathbf{x} \in \mathbb{R}^M): \mathbb{R}^M \rightarrow \mathbb{R}^1$, where $i = 1, \dots, N$ (large)
- 2) Linear function for separability in high N -dimensional space

- **Interpolation problem and RBFs**
- RBF networks – hybrid learning
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

The radial-basis-function (RBF) technique

Nonlinear mapping with the use of *radial-basis-functions* (RBFs):

$$\varphi_i \left(\left\| \mathbf{x} - \mathbf{x}_i \right\| \right)$$

\mathbf{x}_i – RBF centre

$\|\cdot\|$ – vector norm, often Euclidean

$\varphi_i(r)$ – kernel function, often Gaussian

- Interpolation problem and RBFs
- RBF networks – hybrid learning
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

The radial-basis-function (RBF) technique

Nonlinear mapping with the use of *radial-basis-functions* (RBFs):

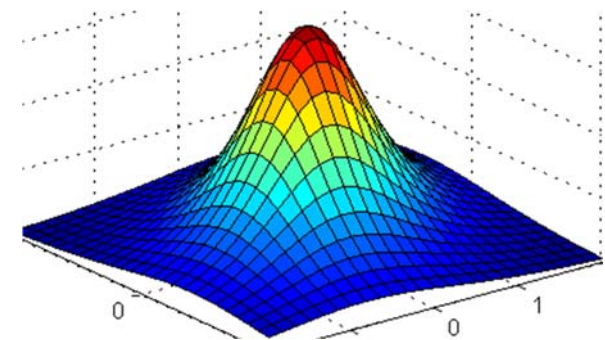
$$\varphi_i \left(\left\| \mathbf{x} - \mathbf{x}_i \right\| \right)$$

\mathbf{x}_i – RBF centre

$\|\cdot\|$ – vector norm, often Euclidean

$\varphi_i(r)$ – kernel function, often Gaussian

$$\varphi_i \left(r = \left\| \mathbf{x} - \mathbf{x}_i \right\| \right) = \exp \left(-\frac{r^2}{2\sigma^2} \right)$$



- **Interpolation problem and RBFs**
- RBF networks – hybrid learning
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

The radial-basis-function (RBF) technique

Nonlinear mapping with the use of *radial-basis-functions* (RBFs):

$$\varphi_i \left(\left\| \mathbf{x} - \mathbf{x}_i \right\| \right)$$

\mathbf{x}_i – RBF centre

$\|\cdot\|$ – vector norm, often Euclidean

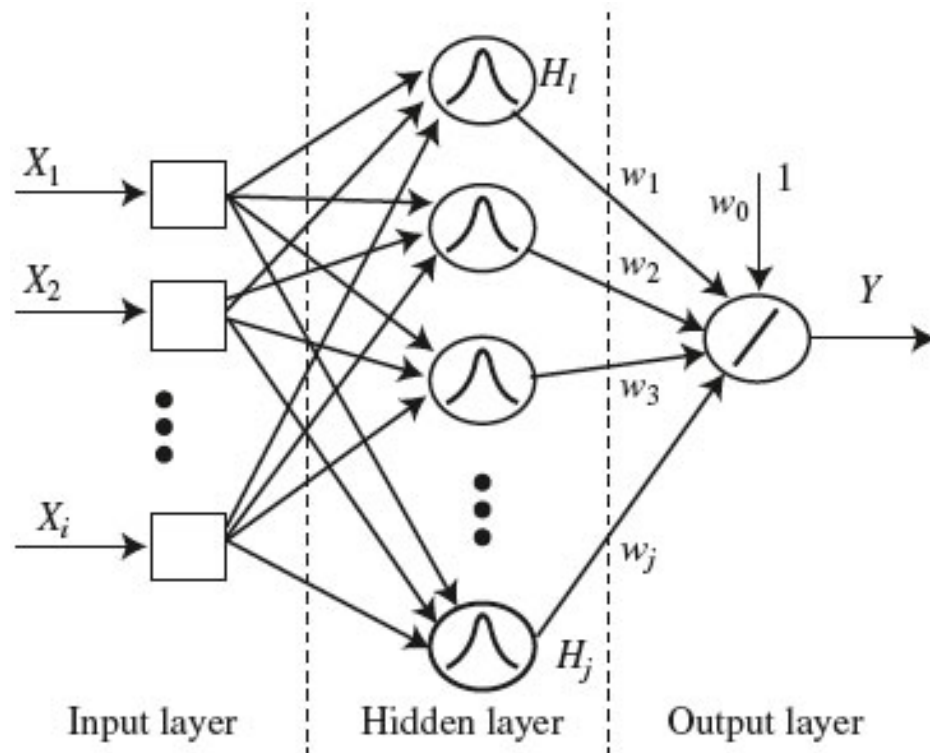
$\varphi_i(r)$ – kernel function, often Gaussian

Linear operation in N -dimensional space:

$$F(\mathbf{x}) = \sum_{i=1}^N w_i \varphi_i \left(\left\| \mathbf{x} - \mathbf{x}_i \right\| \right)$$

- Interpolation problem and RBFs
- **RBF networks – hybrid learning**
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

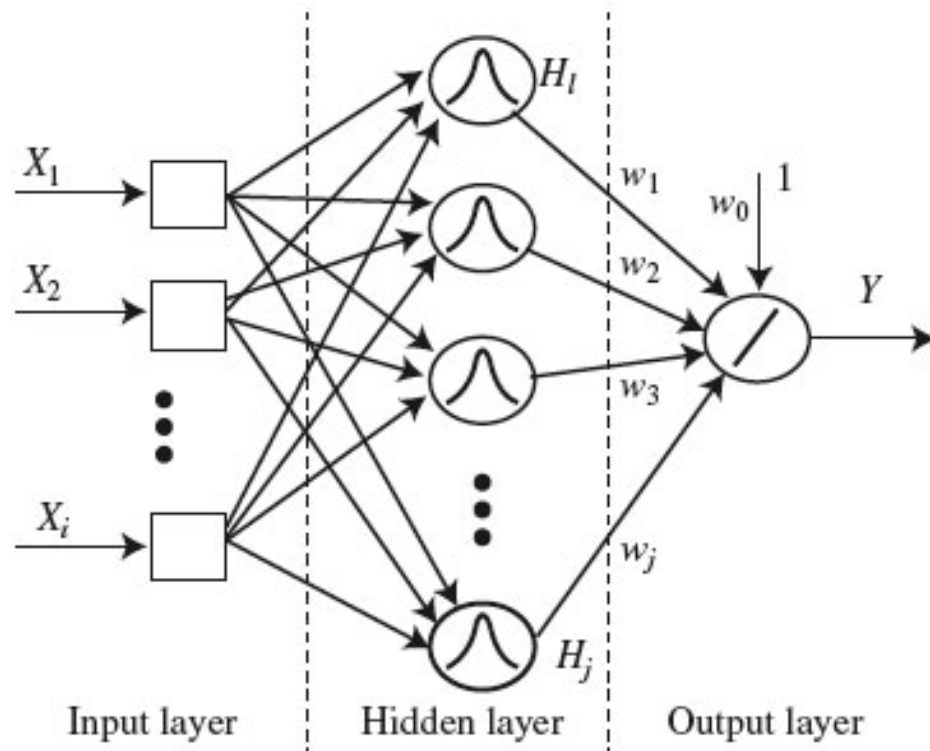
The RBF neural network concept



In the *exact interpolation*, the size of the hidden layer, N , is equal to the number of samples n

- Interpolation problem and RBFs
- **RBF networks – hybrid learning**
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

The RBF neural network concept

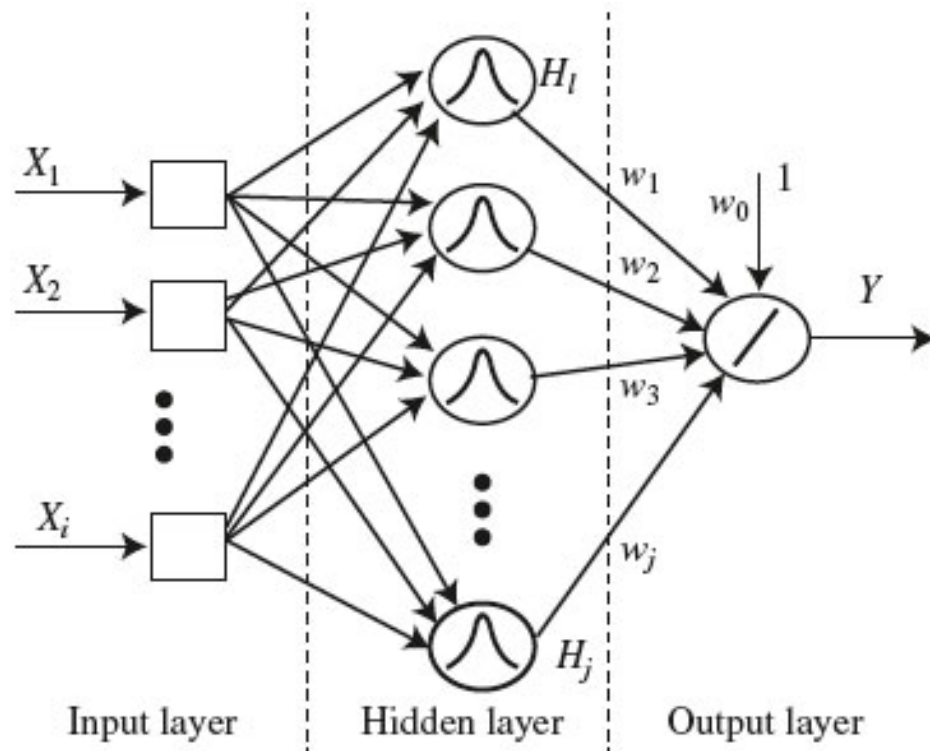


In the *exact interpolation*, the size of the hidden layer, N , is equal to the number of samples n

BUT this is not robust especially if a lot of samples are corrupted with noise!

- Interpolation problem and RBFs
- **RBF networks – hybrid learning**
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

The RBF neural network concept



In the *exact interpolation*, the size of the hidden layer, N , is equal to the number of samples n

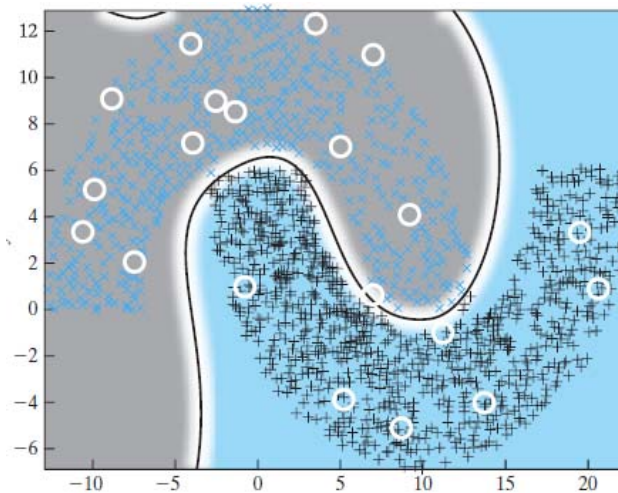
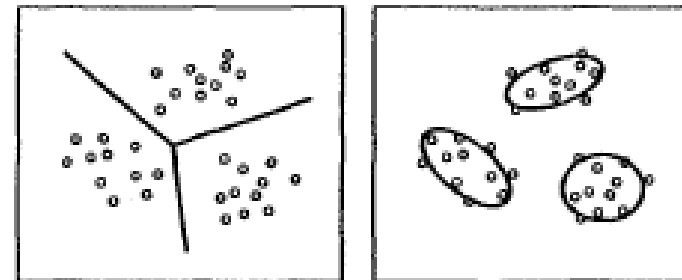
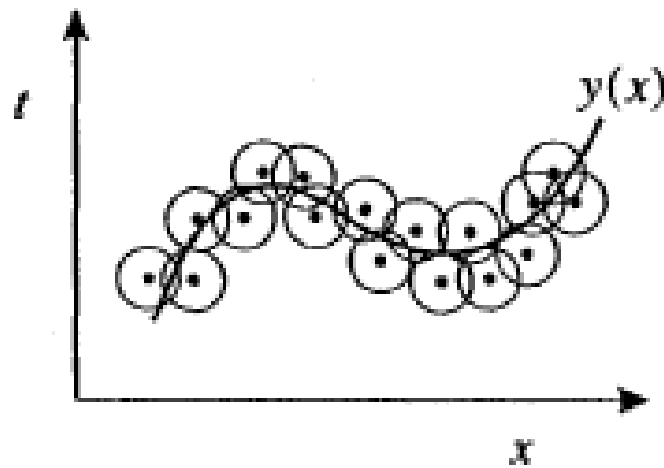
BUT this is not robust especially if a lot of samples are corrupted with noise!

We need modifications:

- 1) $N < n$
- 2) centres \mathbf{x}_i different from samples
- 3) widths, σ , also differ across RBF nodes
- 4) it is possible to include biases

- Interpolation problem and RBFs
- **RBF networks – hybrid learning**
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

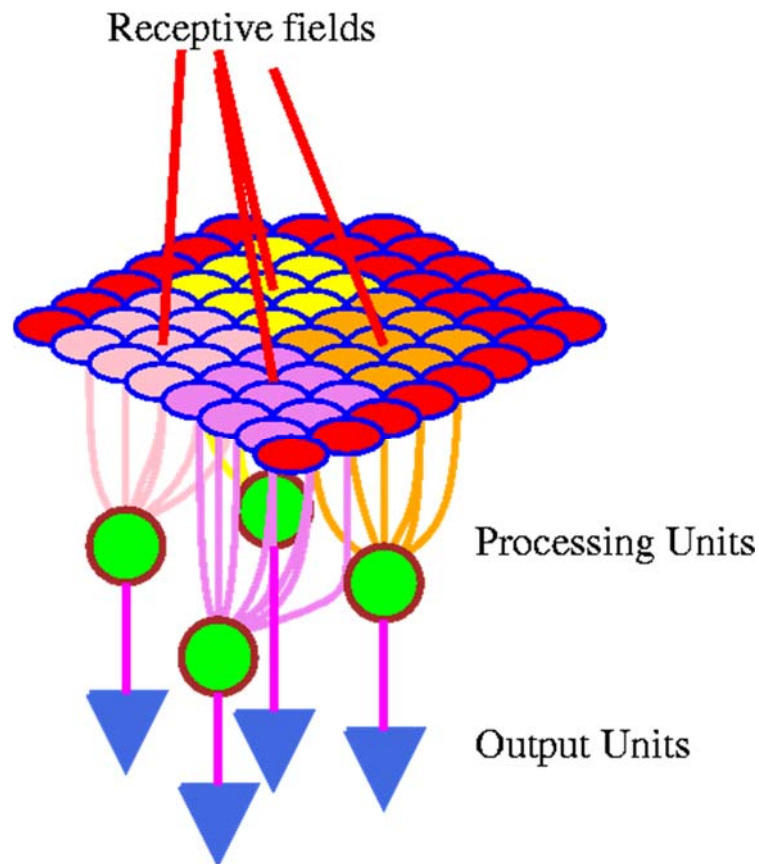
Examples



See *Bishop (1995)* for Bayesian interpretation of classification with RBF networks

- Interpolation problem and RBFs
- **RBF networks – hybrid learning**
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

Analogy with receptive fields

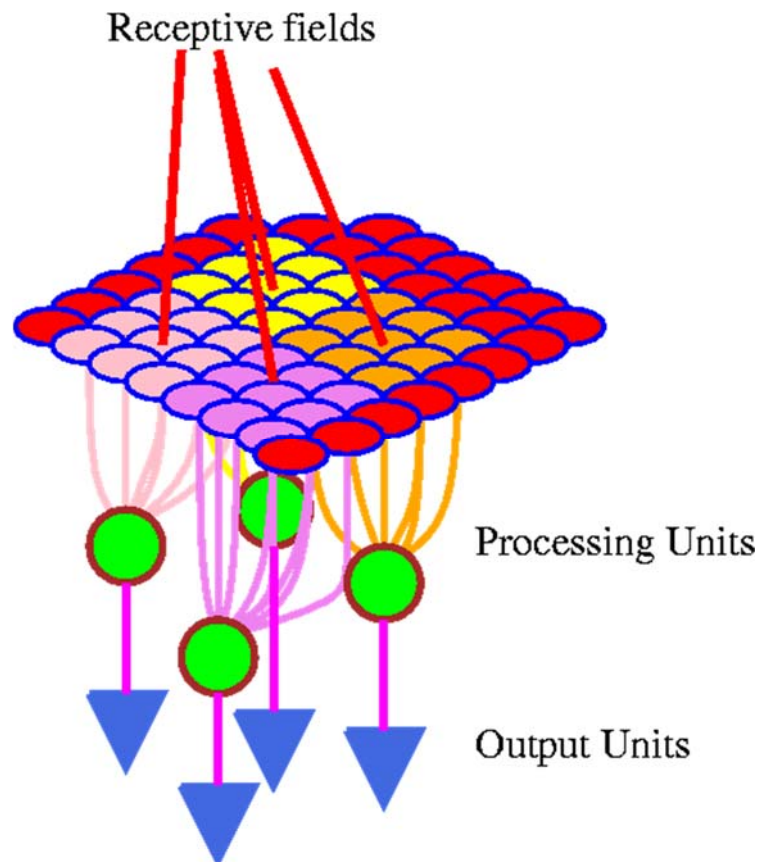


The **receptive field** of a unit is the region of the input space from which a stimulus pattern evokes a response.

Haykin, 2009

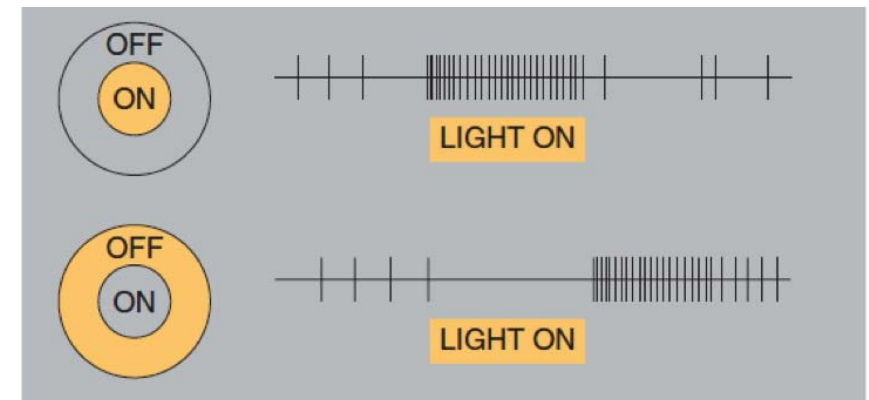
- Interpolation problem and RBFs
- **RBF networks – hybrid learning**
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

Analogy with receptive fields



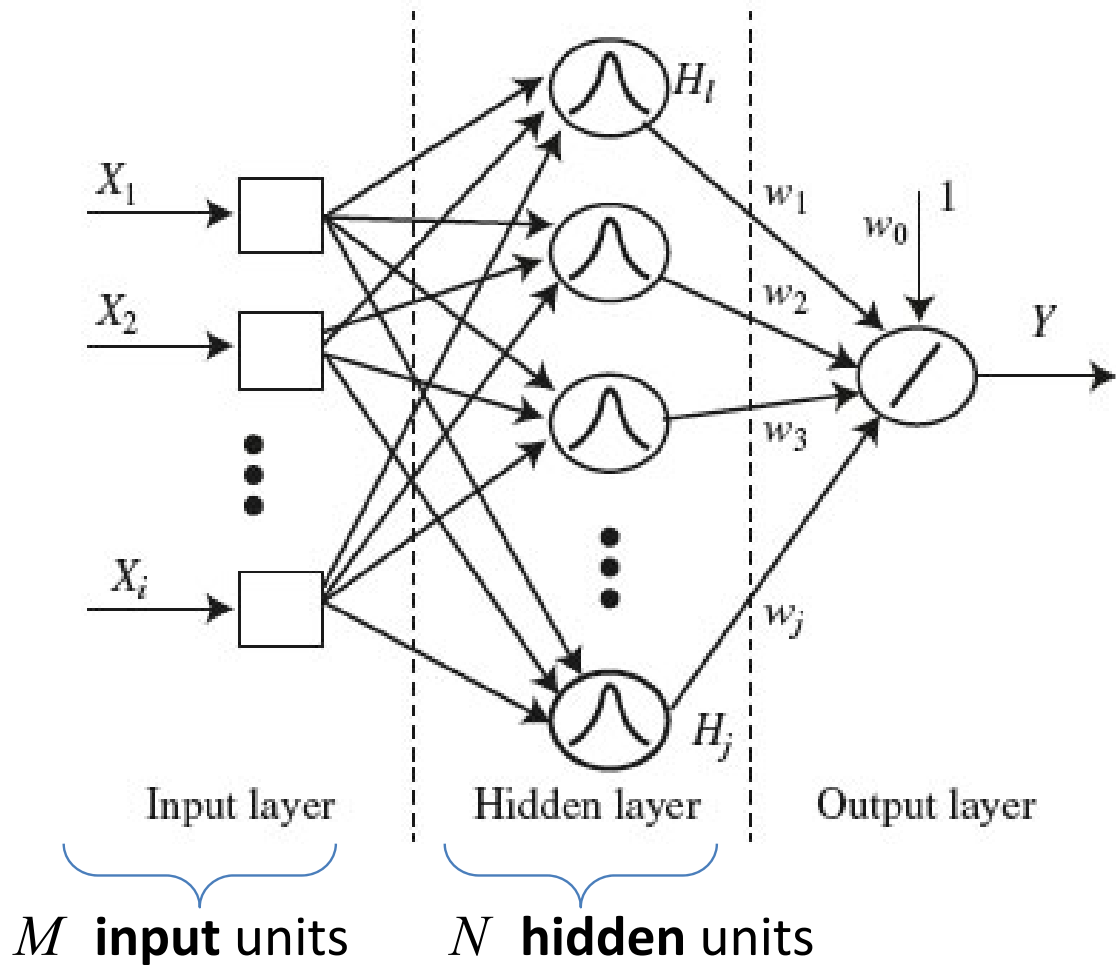
Example:

Retinal ON-centre and OFF-centre cells



- Interpolation problem and RBFs
- **RBF networks – hybrid learning**
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

Hybrid learning algorithm

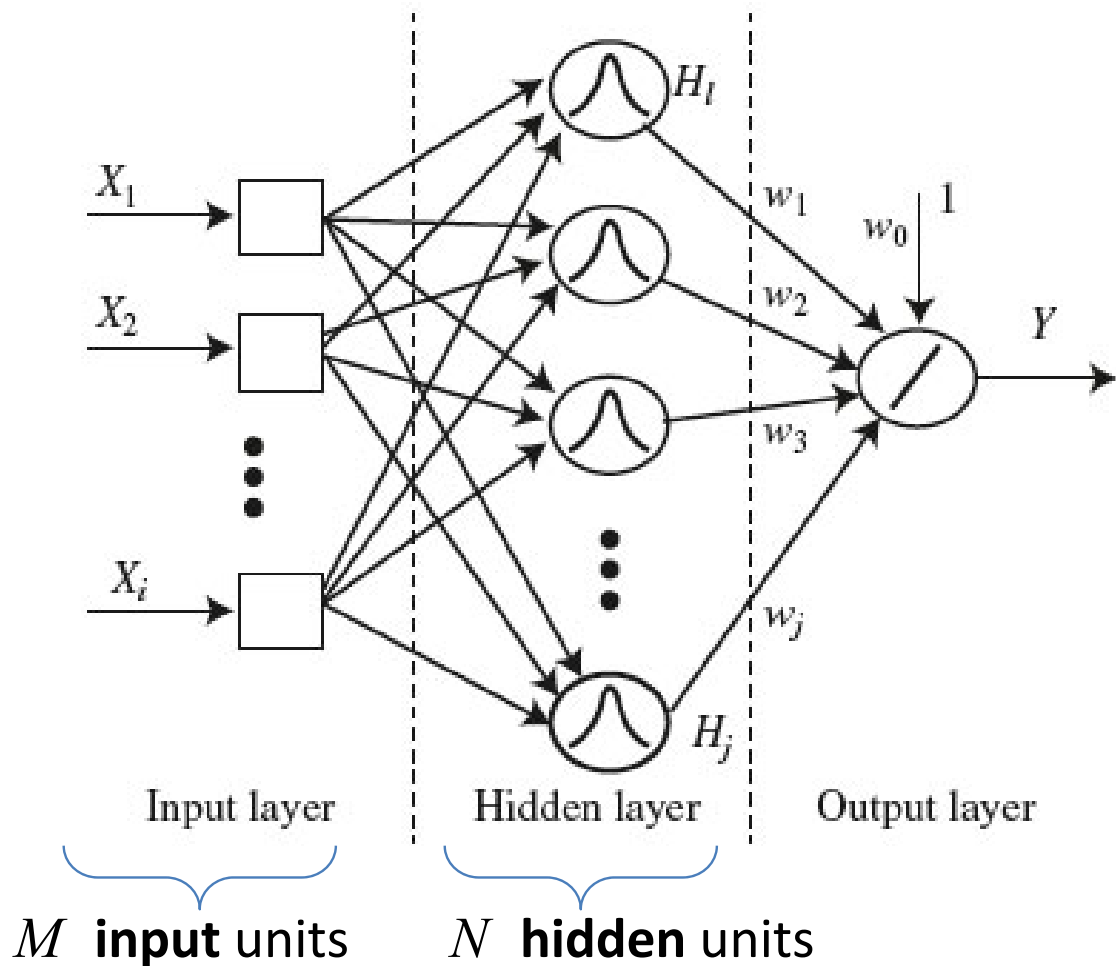


n training samples

mapping: $\mathbb{R}^M \rightarrow \mathbb{R}^N \rightarrow \mathbb{R}^1$

- Interpolation problem and RBFs
- **RBF networks – hybrid learning**
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

Hybrid learning algorithm



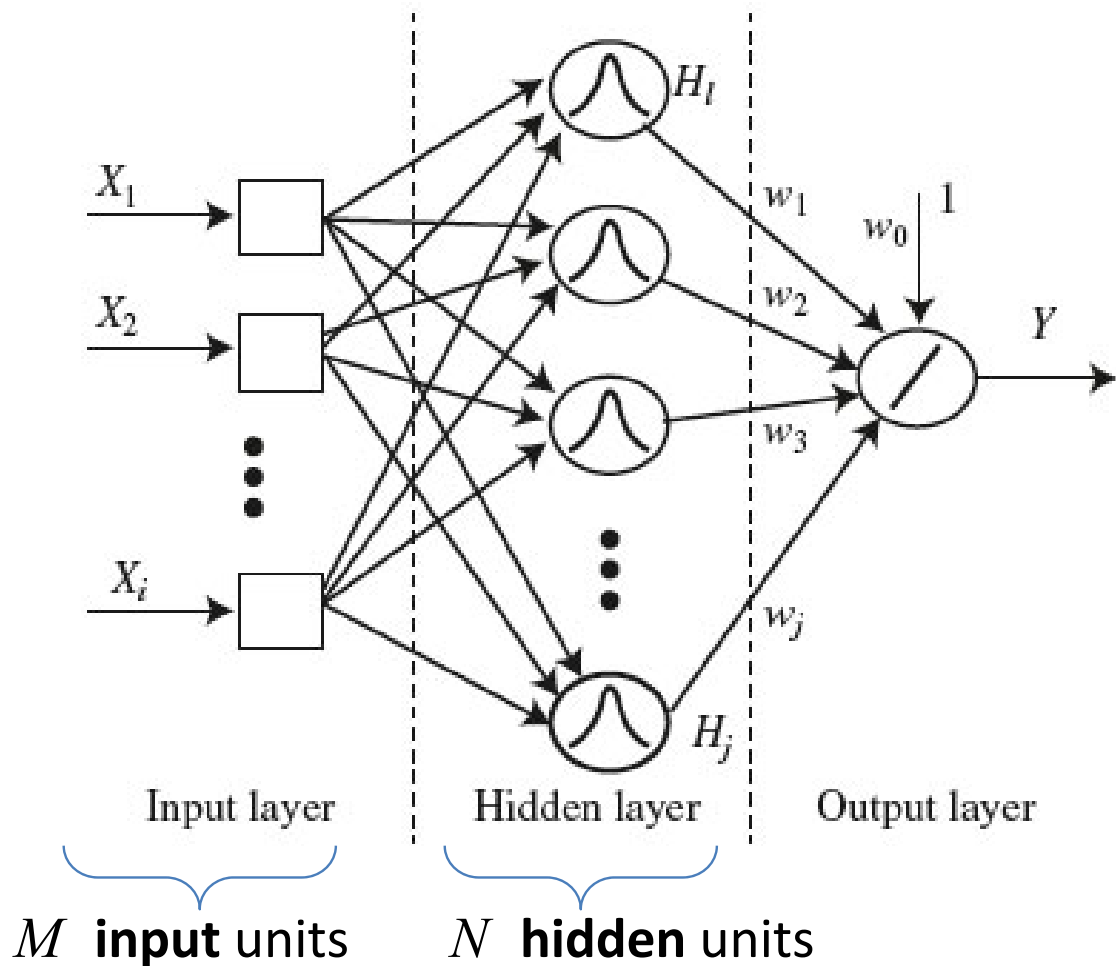
Size of the **input layer** determined by the dimensionality of the input.

n training samples

mapping: $\mathbb{R}^M \rightarrow \mathbb{R}^N \rightarrow \mathbb{R}^1$

- Interpolation problem and RBFs
- **RBF networks – hybrid learning**
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

Hybrid learning algorithm



n training samples

mapping: $\mathbb{R}^M \rightarrow \mathbb{R}^N \rightarrow \mathbb{R}^1$

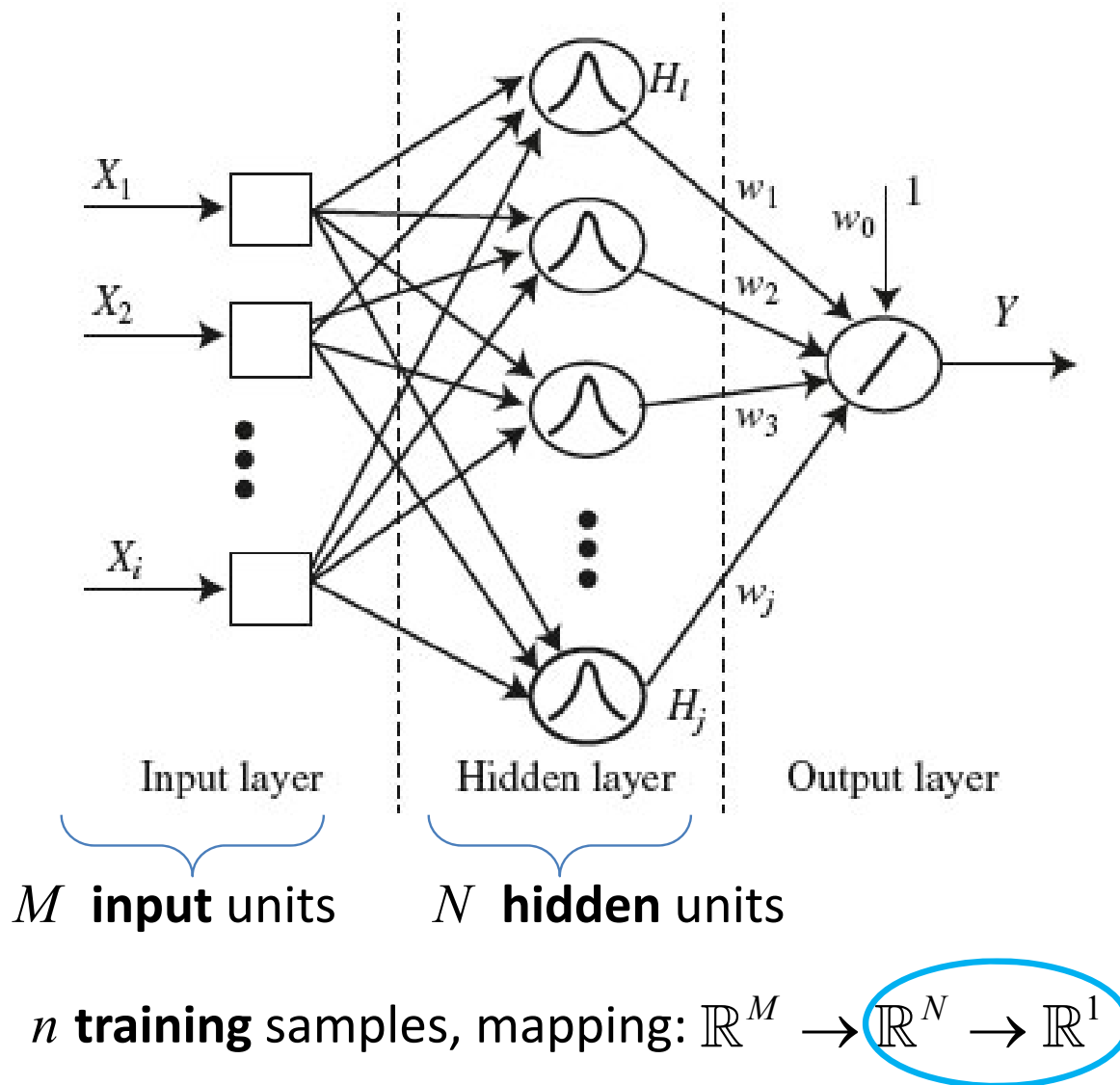
Size of the **input layer** determined by the dimensionality of the input.

Hidden layer

- N has to be decided
- centres and widths of hidden units have to be identified

- Interpolation problem and RBFs
- **RBF networks – hybrid learning**
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

Hybrid learning algorithm



Size of the **input layer** determined by the dimensionality of the input.

Hidden layer

- N has to be decided
- centres and widths of hidden units have to be identified

Output layer performs a linear mapping – e.g., training with least square methods

$$\begin{bmatrix} \varphi_{11} & \varphi_{12} & \dots & \varphi_{1N} \\ \varphi_{21} & \varphi_{22} & \dots & \varphi_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_{n1} & \varphi_{n2} & \dots & \varphi_{nN} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{bmatrix}$$

- Interpolation problem and RBFs
- **RBF networks – hybrid learning**
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

Hybrid learning – RBF layer

- Clustering algorithms
 - k-means clustering
 - clustering with Kohonen feature maps (SOMs), vector quantization (VQ)
 - estimate of the cluster width (variance problem)
- Gaussian mixture models: expectation-maximization (EM) alg.

- Interpolation problem and RBFs
- **RBF networks – hybrid learning**
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

Hybrid learning – RBF layer

- Clustering algorithms
 - k-means clustering
 - clustering with Kohonen feature maps (SOMs), vector quantization (VQ)
 - estimate of the cluster width (variance problem)
- Gaussian mixture models: expectation-maximization (EM) alg.
- Supervised tuning of the RBF parameters with gradient descent

$$\frac{\partial E}{\partial \mu_{ji}} = \sum_n \sum_k \{y_k(\mathbf{x}^n) - t_k^n\} w_{kj} \exp\left(-\frac{\|\mathbf{x}^n - \mu_j\|^2}{2\sigma_j^2}\right) \frac{(x_i^n - \mu_{ji})}{\sigma_j^2}$$

$$\frac{\partial E}{\partial \sigma_j} = \sum_n \sum_k \{y_k(\mathbf{x}^n) - t_k^n\} w_{kj} \exp\left(-\frac{\|\mathbf{x}^n - \mu_j\|^2}{2\sigma_j^2}\right) \frac{\|\mathbf{x}^n - \mu_j\|^2}{\sigma_j^3}$$

Bishop, 1995

- Interpolation problem and RBFs
- **RBF networks – hybrid learning**
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

Hybrid learning – RBF layer

- Clustering algorithms
 - k-means clustering
 - clustering with Kohonen feature maps (SOMs), vector quantization (VQ)
 - estimate of the cluster width (variance problem)
- Gaussian mixture models: expectation-maximization (EM) alg.
- Supervised tuning of the RBF parameters with gradient descent

$$\frac{\partial E}{\partial \mu_{ji}} = \sum_n \sum_k \{y_k(\mathbf{x}^n) - t_k^n\} w_{kj} \exp\left(-\frac{\|\mathbf{x}^n - \mu_j\|^2}{2\sigma_j^2}\right) \frac{(x_i^n - \mu_{ji})}{\sigma_j^2}$$

$$\frac{\partial E}{\partial \sigma_j} = \sum_n \sum_k \{y_k(\mathbf{x}^n) - t_k^n\} w_{kj} \exp\left(-\frac{\|\mathbf{x}^n - \mu_j\|^2}{2\sigma_j^2}\right) \frac{\|\mathbf{x}^n - \mu_j\|^2}{\sigma_j^3}$$

Bishop, 1995

- Subset selection with orthogonal least squares

- Interpolation problem and RBFs
- **RBF networks – hybrid learning**
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

Hybrid learning – output layer

- Least-squares fitting – batch approach

$$\begin{bmatrix} \varphi_{11} & \varphi_{12} & \dots & \varphi_{1N} \\ \varphi_{21} & \varphi_{22} & \dots & \varphi_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_{n1} & \varphi_{n2} & \dots & \varphi_{nN} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{bmatrix}$$

- Interpolation problem and RBFs
- **RBF networks – hybrid learning**
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

Hybrid learning – output layer

- Least-squares fitting – batch approach

$$\begin{bmatrix} \varphi_{11} & \varphi_{12} & \dots & \varphi_{1N} \\ \varphi_{21} & \varphi_{22} & \dots & \varphi_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_{n1} & \varphi_{n2} & \dots & \varphi_{nN} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{bmatrix}$$

- Recursive least-squares estimation (see *Haykin*, RLS algorithm)

delta rule:

$$\begin{aligned} \Delta \mathbf{w} &= -\eta \nabla_{\mathbf{w}} \hat{\xi} \\ &= -\eta \frac{1}{2} \nabla_{\mathbf{w}} (f(x_k) - \Phi(x_k)^\top \mathbf{w})^2 \\ &= \eta (f(x_k) - \Phi(x_k)^\top \mathbf{w}) \Phi(x_k) \\ &= \eta e \Phi(x_k) \end{aligned} \quad \Phi(x_k) = \begin{pmatrix} \phi_1(x_k) \\ \phi_2(x_k) \\ \vdots \\ \phi_n(x_k) \end{pmatrix}$$

- Interpolation problem and RBFs
- **RBF networks – hybrid learning**
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

RBF NN vs MLP

- Hidden units in MLP rely on weighted linear summations of inputs
(a matter of interpretation: *it could be seen as the correlation or scalar prod*)
 - RBFs rely on distance to prototype vectors

- Interpolation problem and RBFs
- **RBF networks – hybrid learning**
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

RBF NN vs MLP

- Hidden units in MLP rely on weighted linear summations of inputs
 - RBFs rely on distance to prototype vectors
- In MLPs, function approximation is defined as by a nested sum of weighted summations
 - In RBFs, the approximation is defined by a single weighted sum

- Interpolation problem and RBFs
- **RBF networks – hybrid learning**
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

RBF NN vs MLP

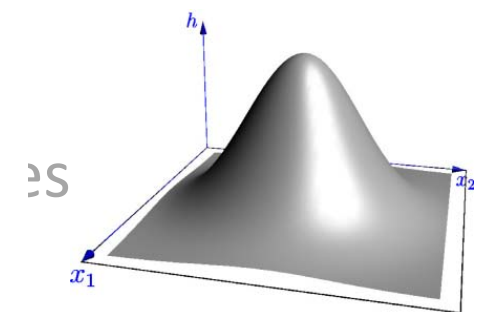
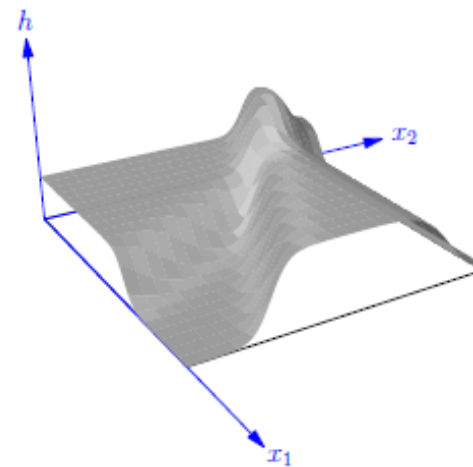
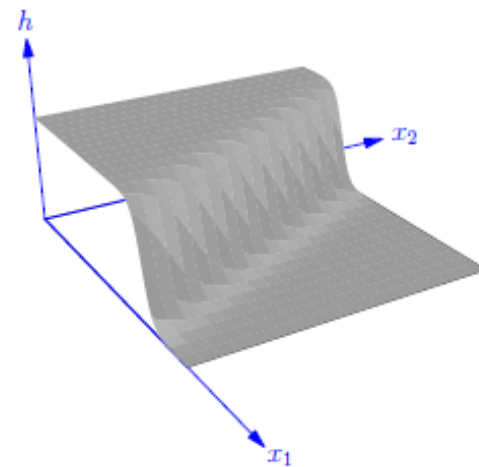
- Hidden units in MLP rely on weighted linear summations of inputs

Weighted sum of Base Functions

➤ RBFs rely

- In MLPs, fun
weighted su

➤ In RBFs, t



weighted sum

- MLPs form distributed activations (many hidden units contribute to the output for a given input, which partly leads to local minima etc.)
 - In RBFs, very few local basis functions (wrt. input) are activated for a given input

- Interpolation problem and RBFs
- **RBF networks – hybrid learning**
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

RBF NN vs MLP

- In MLPs, usually all parameters/weights are trained at the same time
 - In RBFs, hybrid two-stage training is used

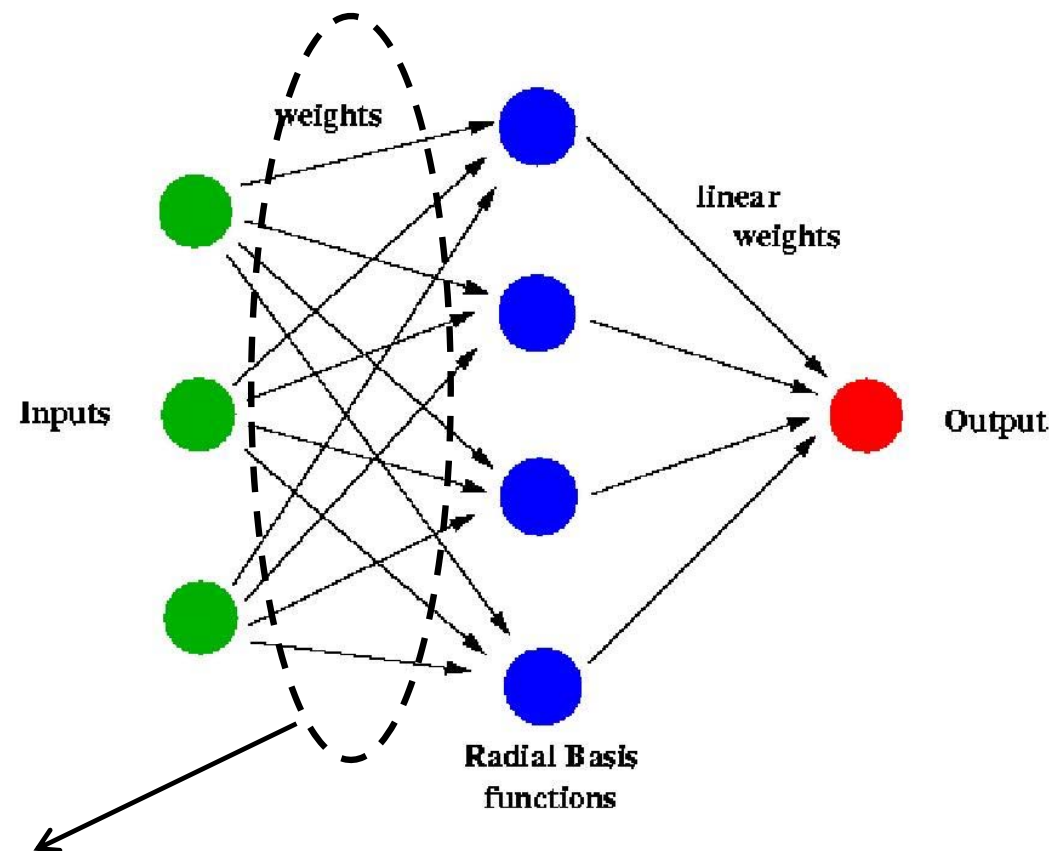
- Interpolation problem and RBFs
- **RBF networks – hybrid learning**
- Weight interpretation in the input space
- Competitive mechanisms for unsupervised learning

RBF NN vs MLP

- In MLPs, usually all parameters/weights are trained at the same time
 - In RBFs, hybrid two-stage training is used
- MLPs rely on complex multi-layer architecture
 - RBF NNs have simple one-hidden-layer architecture

- Interpolation problem and RBFs
- RBF networks – hybrid learning
- **Weight interpretation in the input space**
- Competitive mechanisms for unsupervised learning

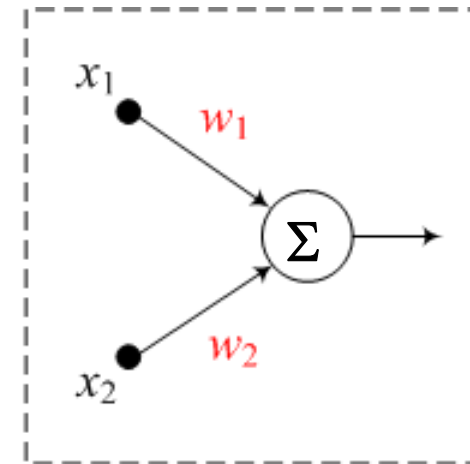
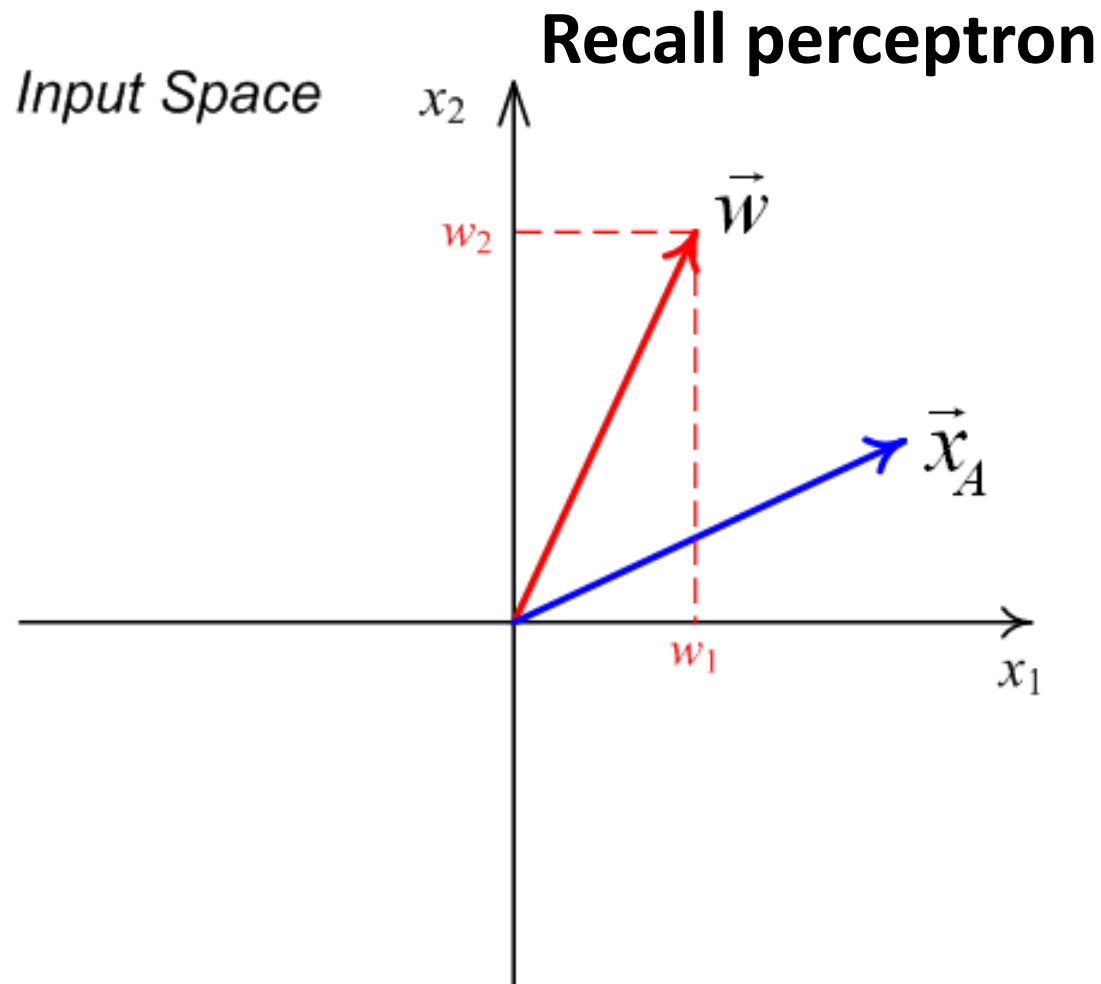
Interpretation of the weights to hidden layer



Do these connections have weights?

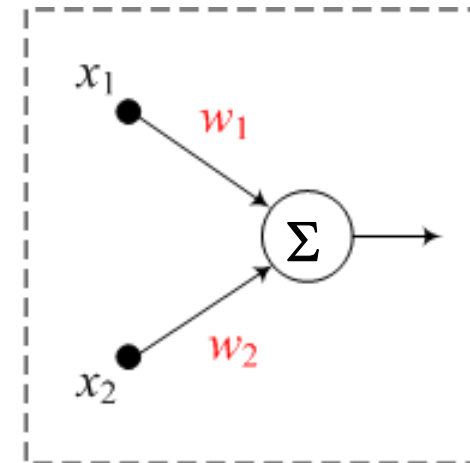
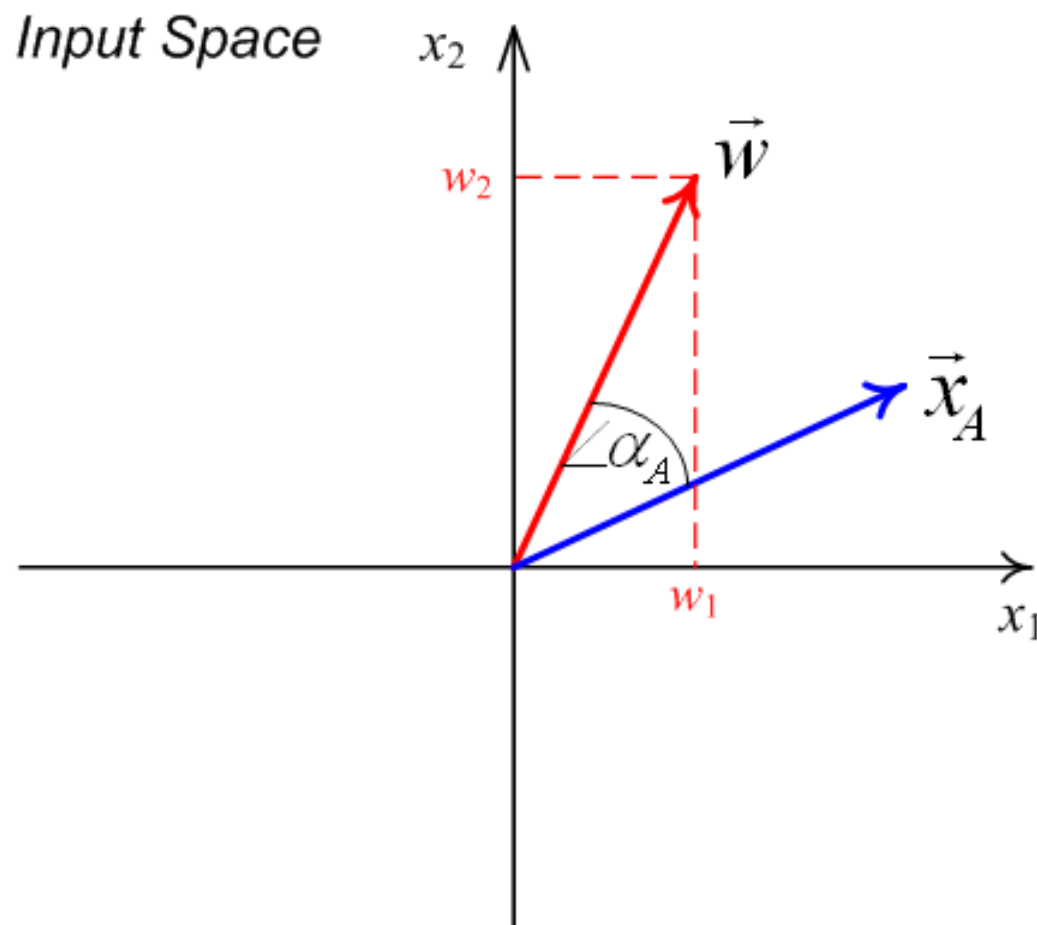
- Interpolation problem and RBFs
- RBF networks – hybrid learning
- **Weight interpretation in the input space**
- Competitive mechanisms for unsupervised learning

Weights in the input space



- Interpolation problem and RBFs
- RBF networks – hybrid learning
- **Weight interpretation in the input space**
- Competitive mechanisms for unsupervised learning

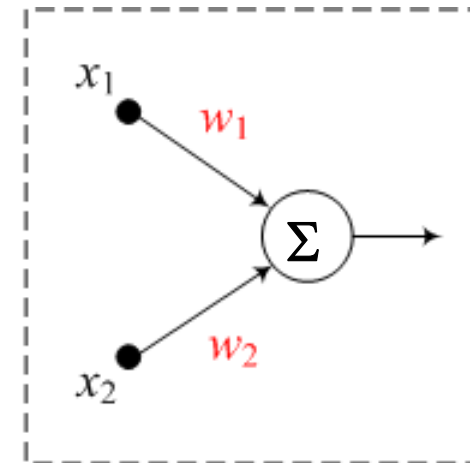
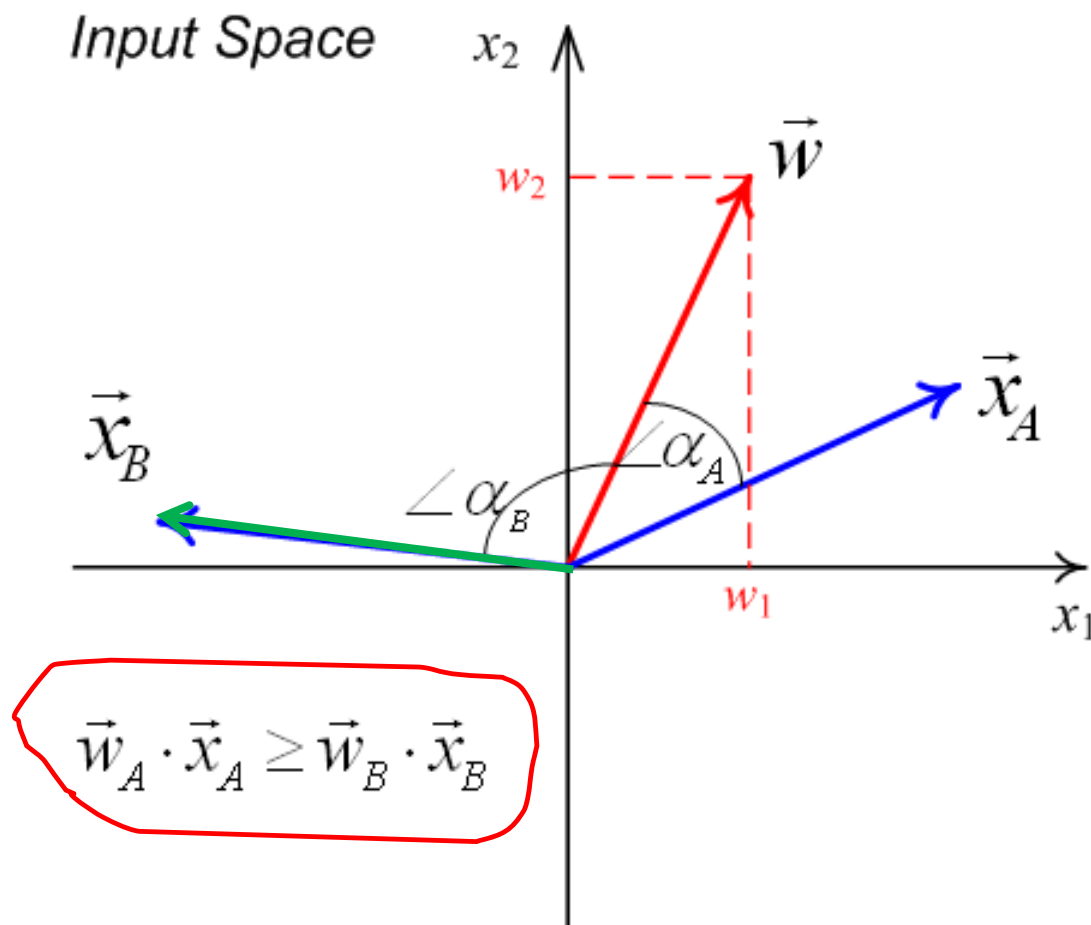
Weights in the input space – scalar product



$$\vec{w} \cdot \vec{x} = \sum_{i=1}^2 w_i x_i = w_1 x_1 + w_2 x_2 = \|\vec{w}\| \|\vec{x}\| \cos(\angle \alpha)$$

- Interpolation problem and RBFs
- RBF networks – hybrid learning
- **Weight interpretation in the input space**
- Competitive mechanisms for unsupervised learning

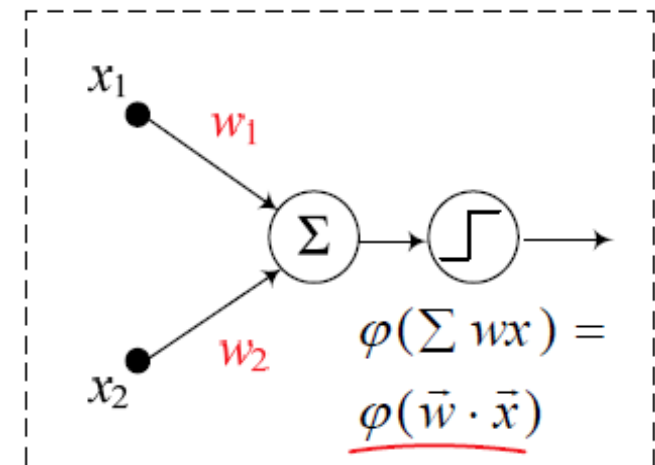
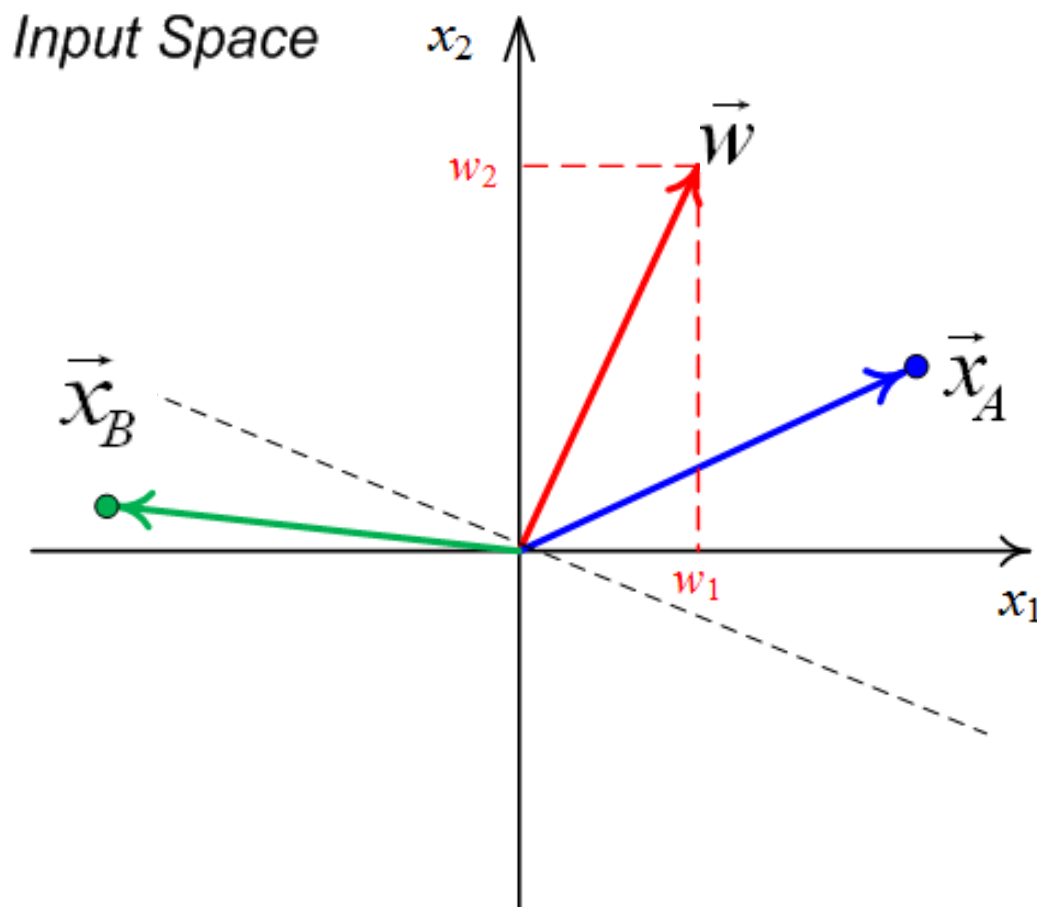
Weights in the input space – scalar product



$$\vec{w} \cdot \vec{x} = \sum_{i=1}^2 w_i x_i = w_1 x_1 + w_2 x_2 = \|\vec{w}\| \|\vec{x}\| \cos(\angle \alpha)$$

- Interpolation problem and RBFs
- RBF networks – hybrid learning
- **Weight interpretation in the input space**
- Competitive mechanisms for unsupervised learning

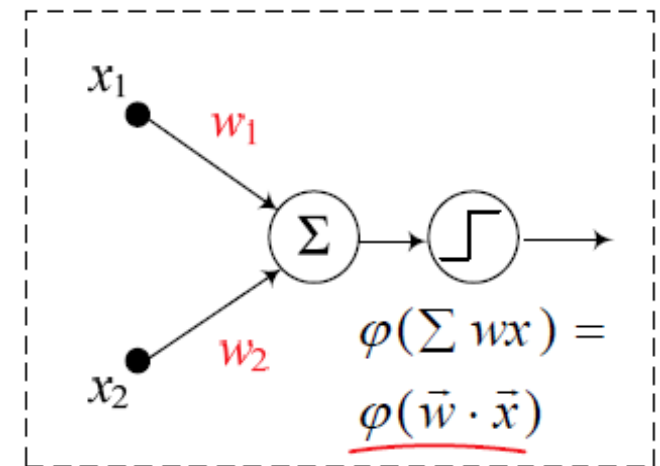
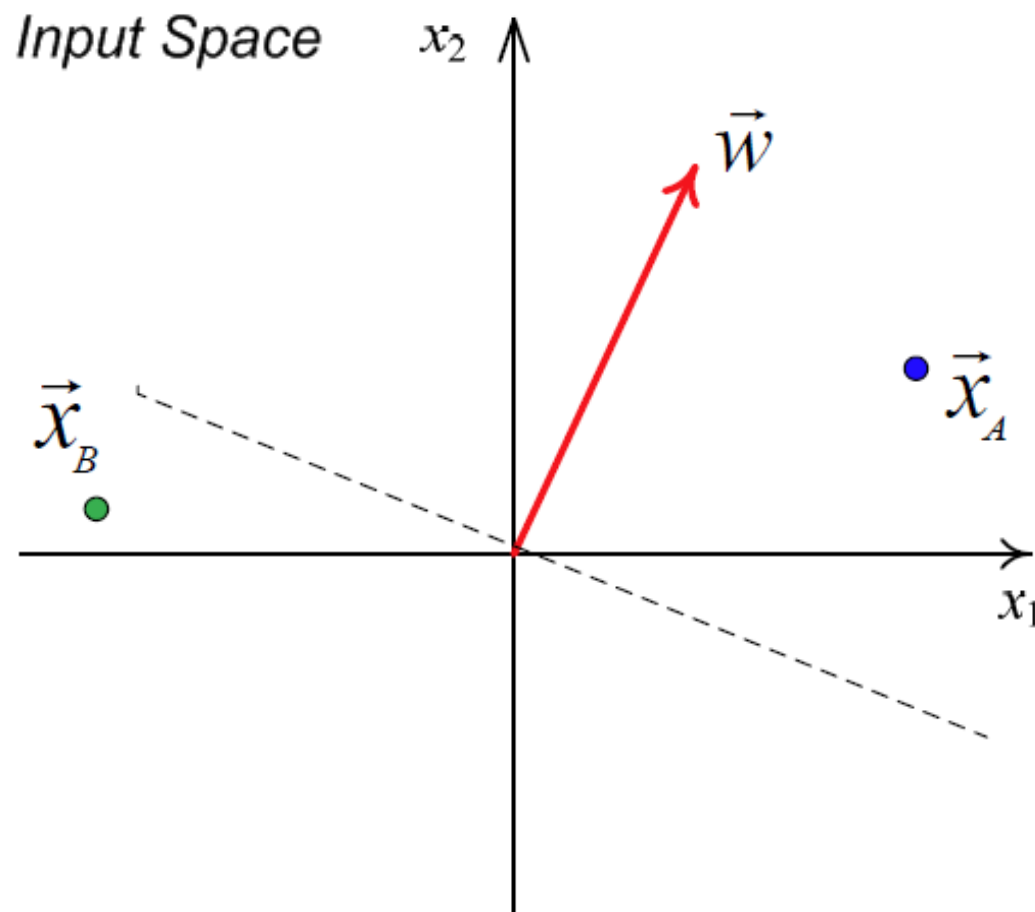
Weights in the input space – perceptron classifier



$$\bar{w} \cdot \bar{x} = \sum_{i=1}^2 w_i x_i = w_1 x_1 + w_2 x_2 = \|\bar{w}\| \|\bar{x}\| \cos(\angle \alpha)$$

- Interpolation problem and RBFs
- RBF networks – hybrid learning
- **Weight interpretation in the input space**
- Competitive mechanisms for unsupervised learning

Weights in the input space – perceptron classifier

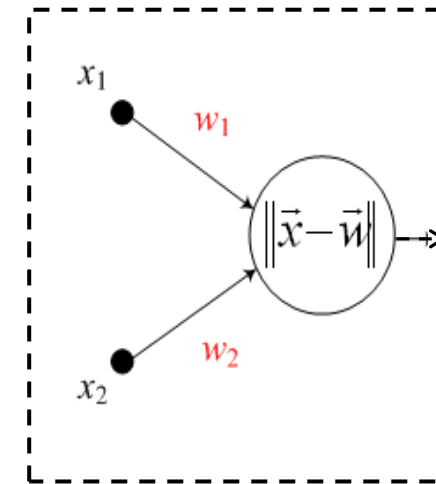
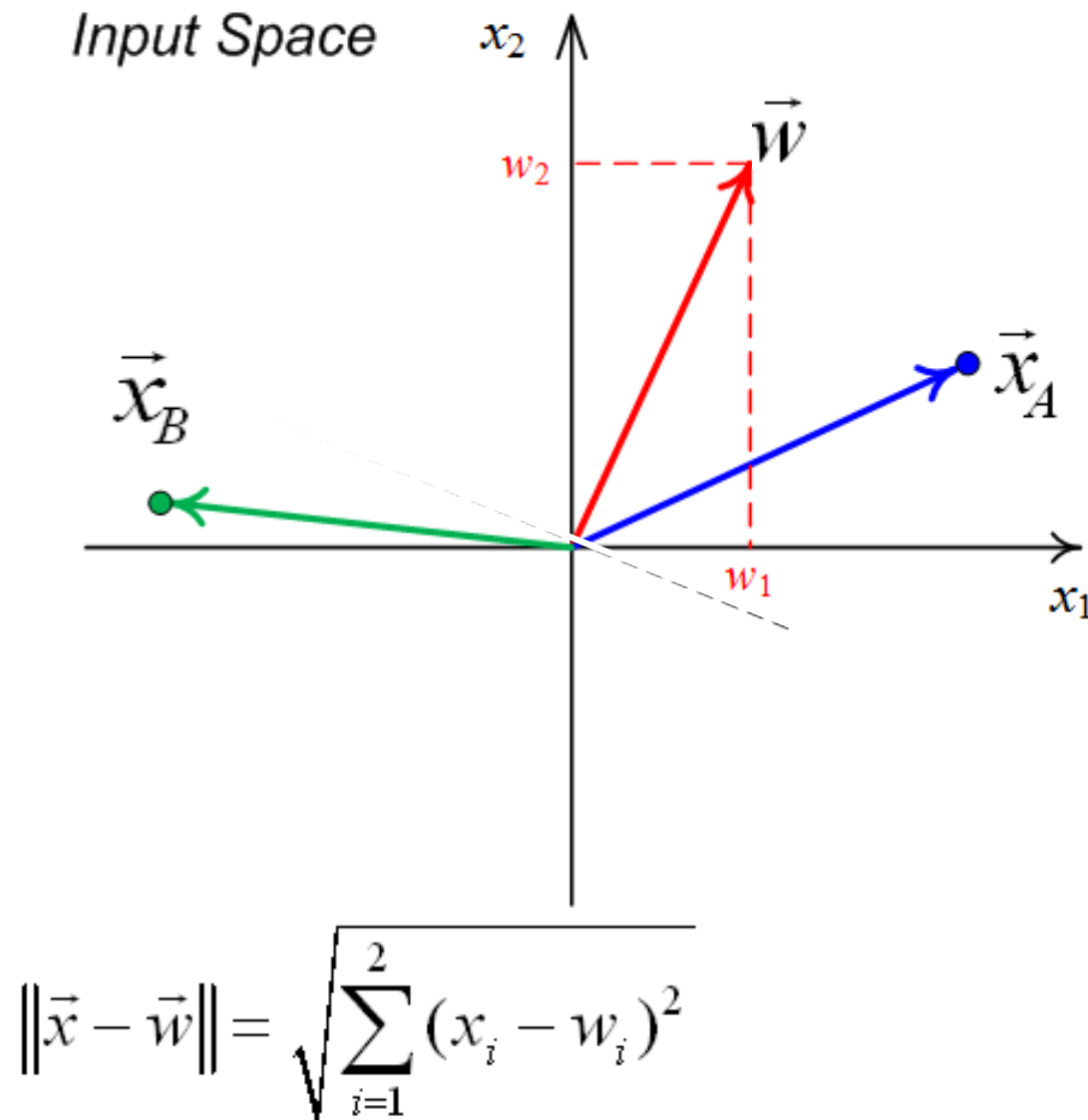


based on scalar product

$$\vec{w} \cdot \vec{x} = \sum_{i=1}^2 w_i x_i = w_1 x_1 + w_2 x_2 = \|\vec{w}\| \|\vec{x}\| \cos(\angle \alpha)$$

- Interpolation problem and RBFs
- RBF networks – hybrid learning
- **Weight interpretation in the input space**
- Competitive mechanisms for unsupervised learning

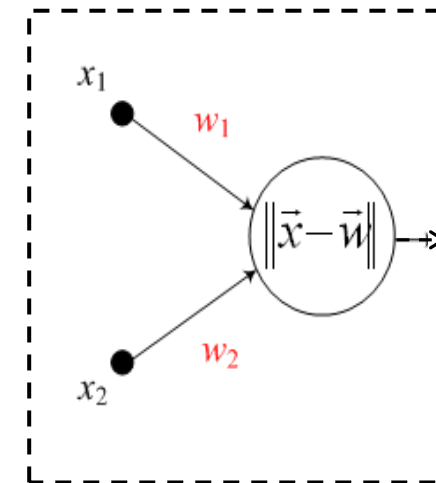
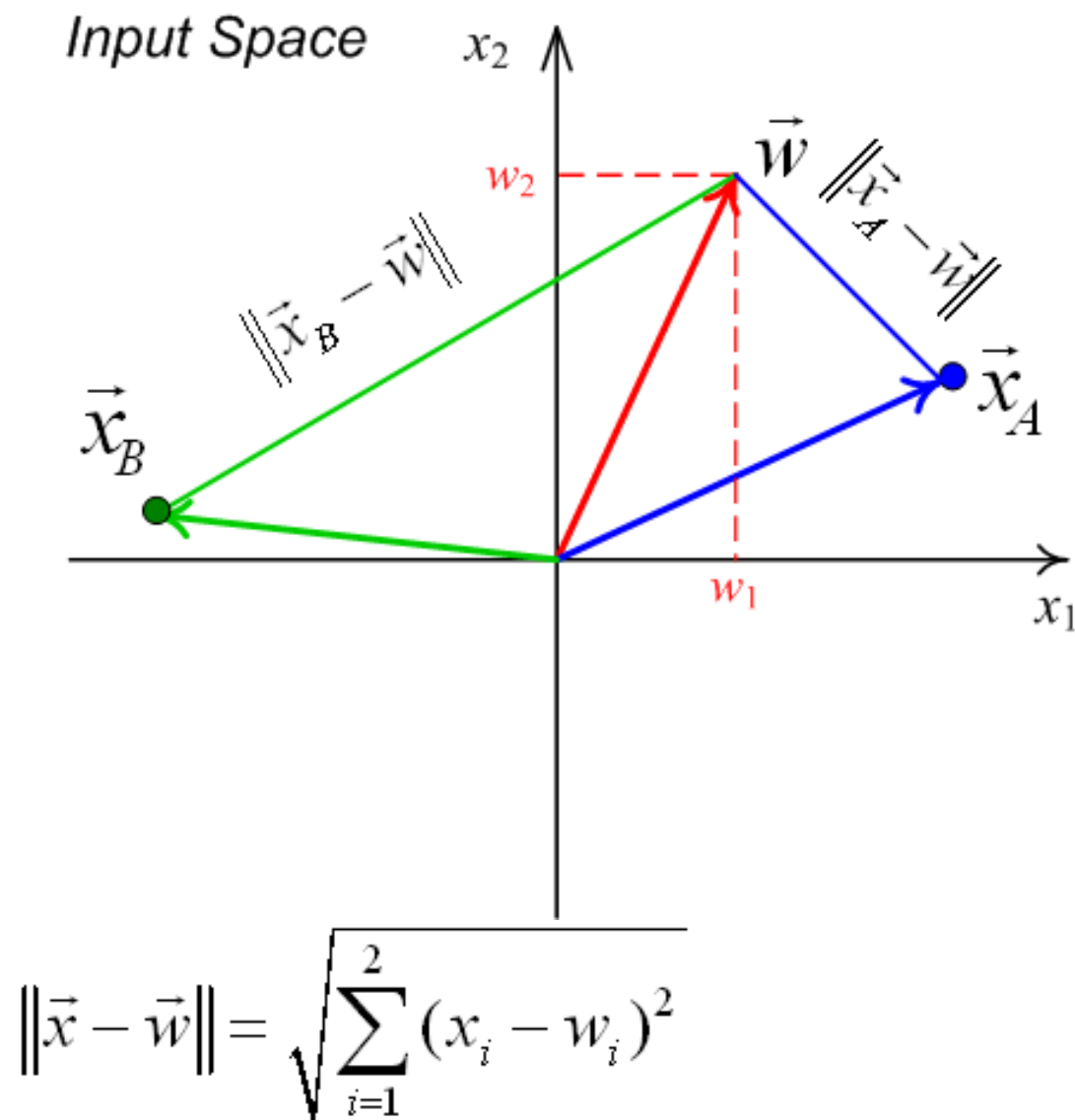
Weights in the input space – Euclidean distance



Euclidean distance
measure: $\|\vec{x} - \vec{w}\|$

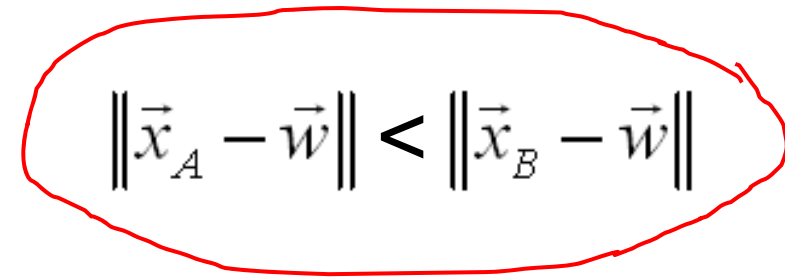
- Interpolation problem and RBFs
- RBF networks – hybrid learning
- **Weight interpretation in the input space**
- Competitive mechanisms for unsupervised learning

Weights in the input space – Euclidean distance



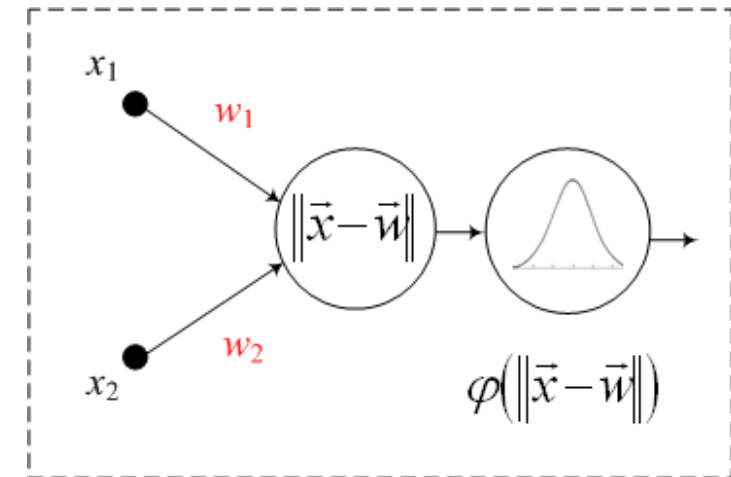
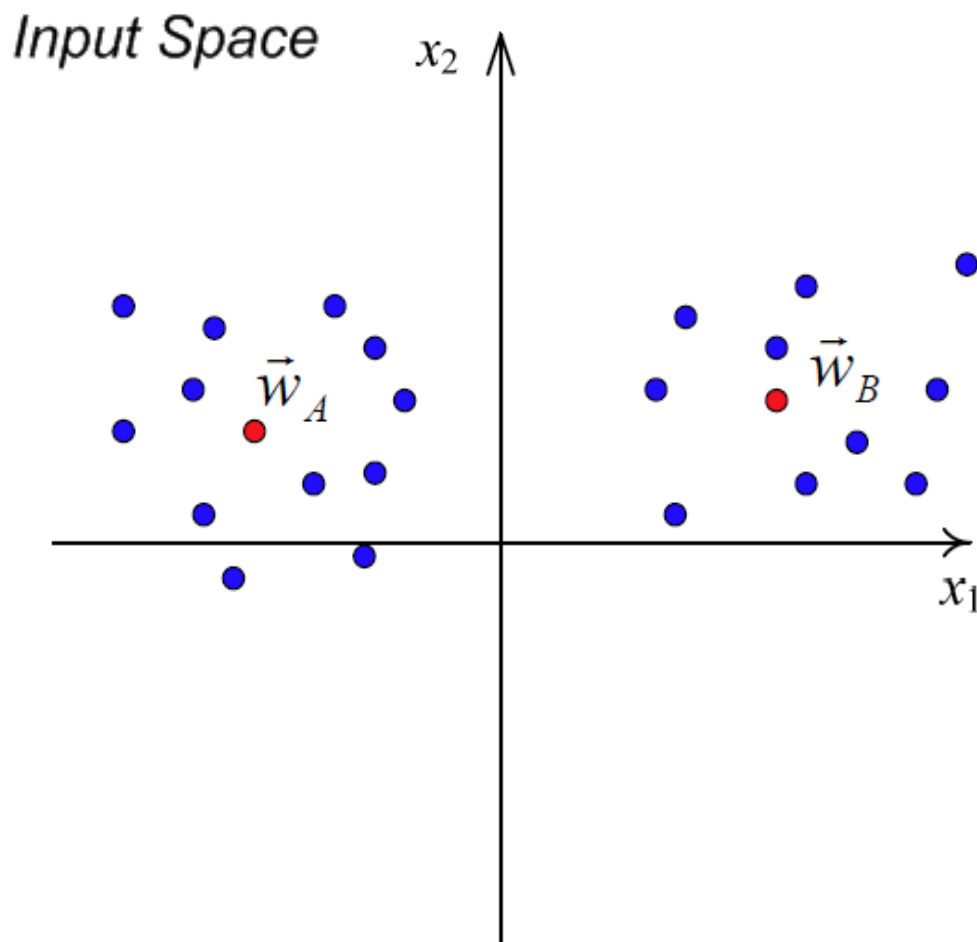
Euclidean distance
measure: $\|\vec{x} - \vec{w}\|$

- # Weights in the input space – Euclidean distance



- Interpolation problem and RBFs
- RBF networks – hybrid learning
- **Weight interpretation in the input space**
- Competitive mechanisms for unsupervised learning

Weights in the input space – proximity measure

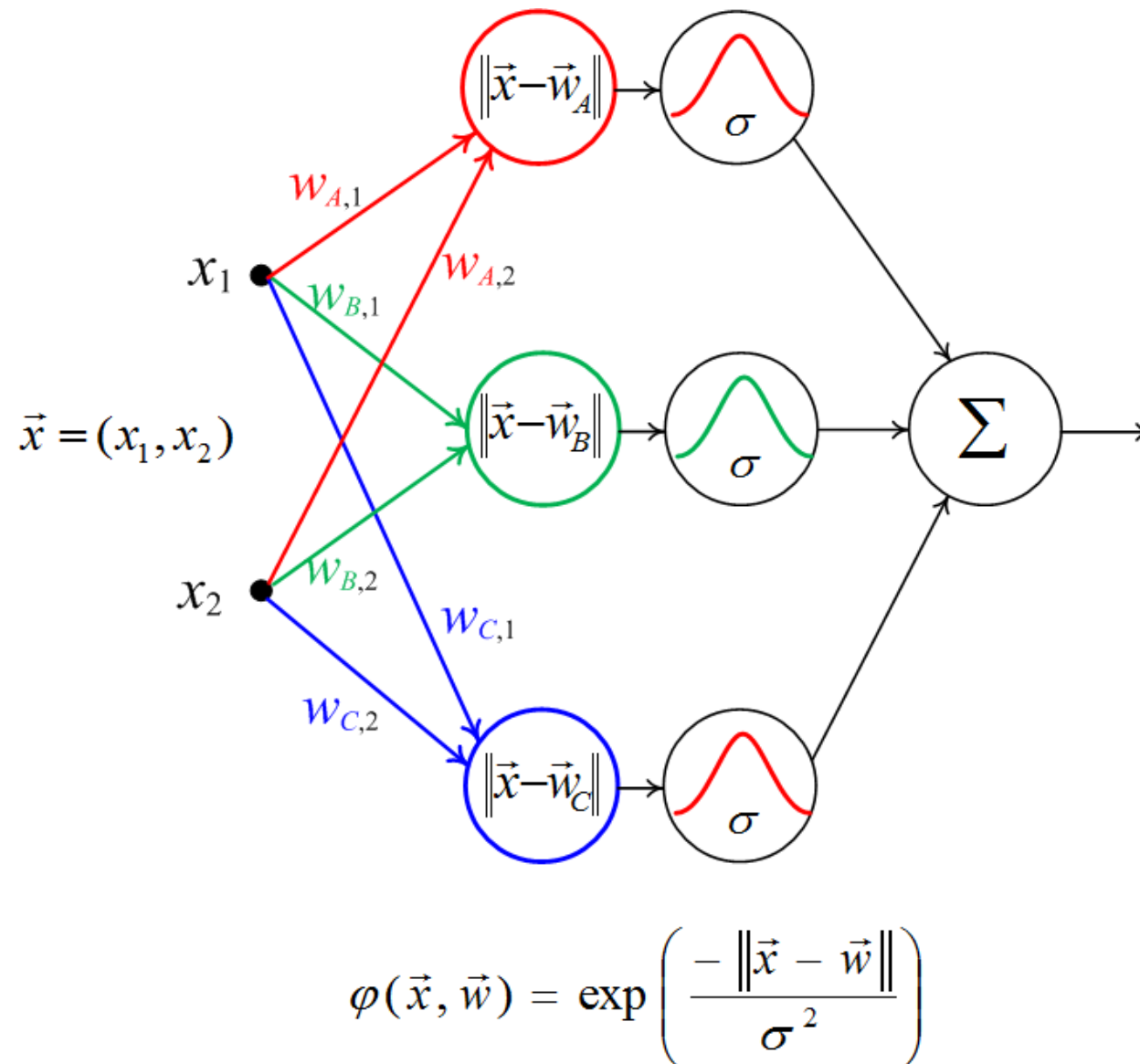


based on Euclidean distance

$$\varphi(\|\vec{x} - \vec{w}\|) = \exp\left(-\frac{\|\vec{x} - \vec{w}\|^2}{\sigma^2}\right)$$

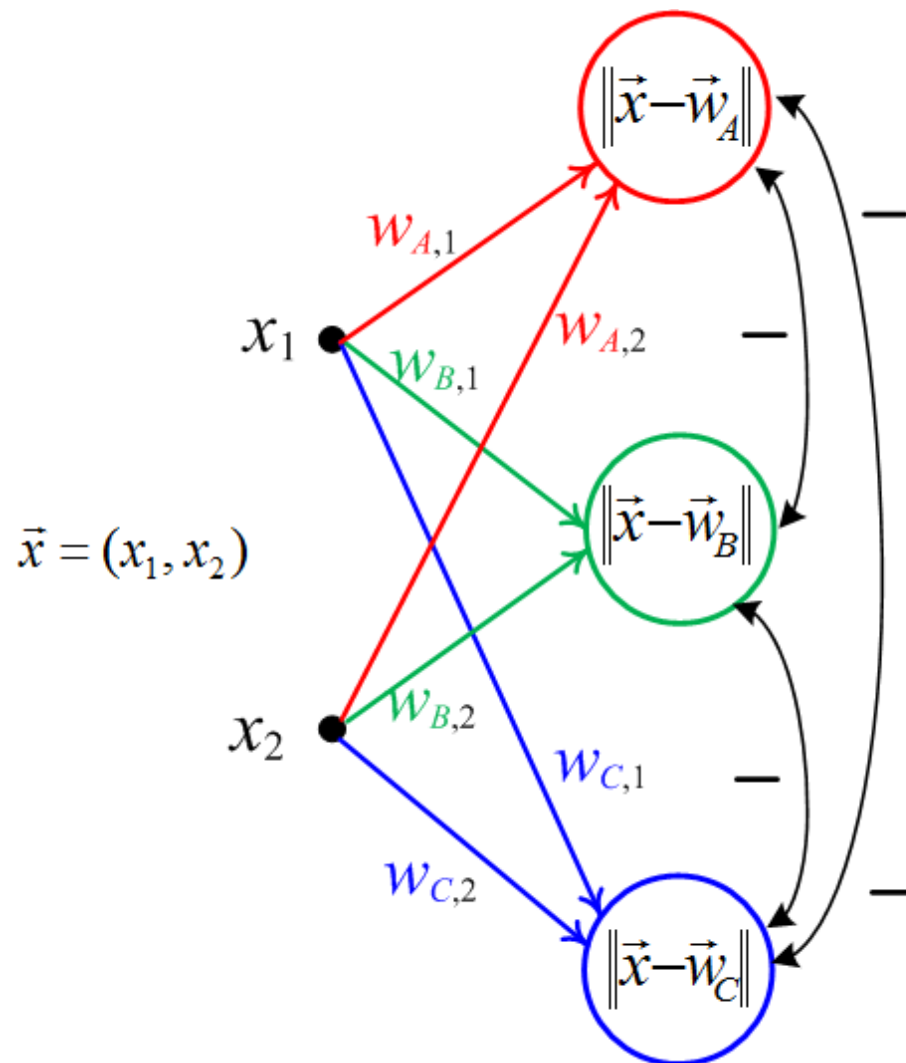
- Interpolation problem and RBFs
- RBF networks – hybrid learning
- **Weight interpretation in the input space**
- Competitive mechanisms for unsupervised learning

RBF network – interpretation of the input weights



- Interpolation problem and RBFs
- RBF networks – hybrid learning
- Weight interpretation in the input space
- **Competitive mechanisms for unsupervised learning**

Competition and winner-take-all mechanism



Competition
between nodes
– Winner
Takes All
(WTA)

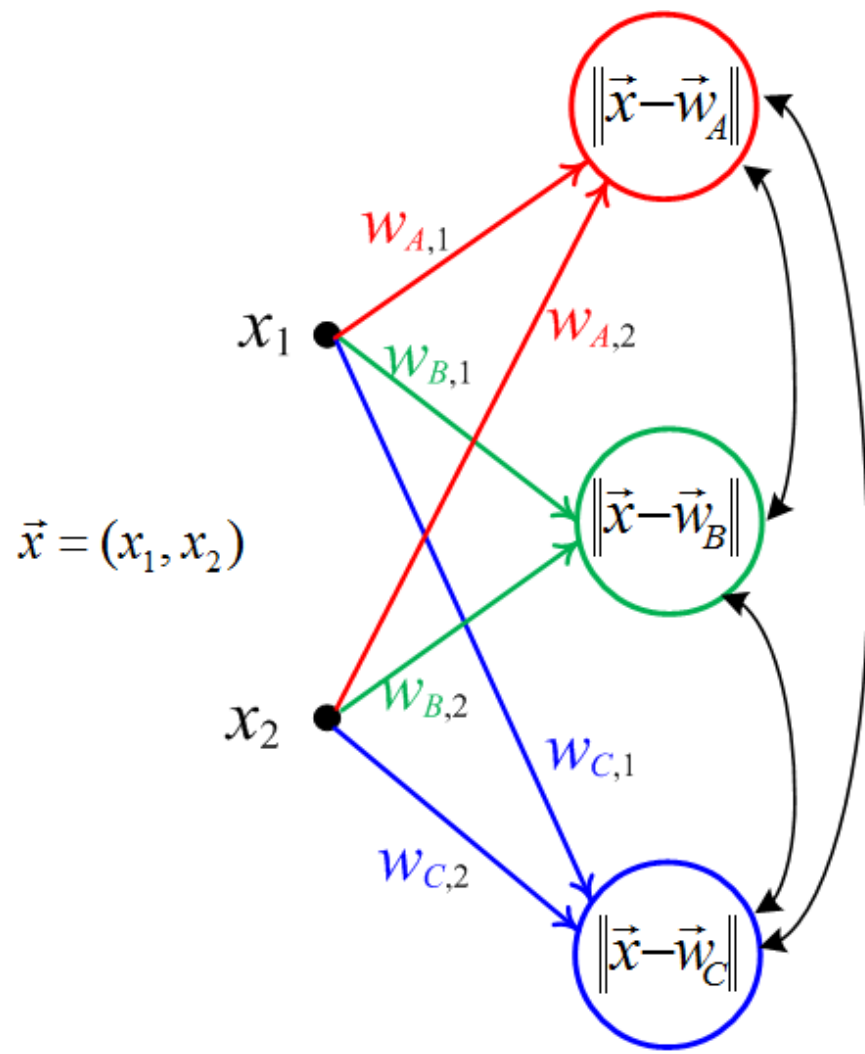
$$\vec{w}_A = (w_{A,1}, w_{A,2})$$

$$\vec{w}_B = (w_{B,1}, w_{B,2})$$

$$\vec{w}_C = (w_{C,1}, w_{C,2})$$

- Interpolation problem and RBFs
- RBF networks – hybrid learning
- Weight interpretation in the input space
- **Competitive mechanisms for unsupervised learning**

Competitive mechanisms



If the **red** node \vec{w}_A wins, then:

$$\Delta \vec{w}_A = \eta \vec{x}$$

OR

$$\Delta \vec{w}_A = \eta (\vec{x} - \vec{w}_A)$$

$$\vec{w}_A = (w_{A,1}, w_{A,2})$$

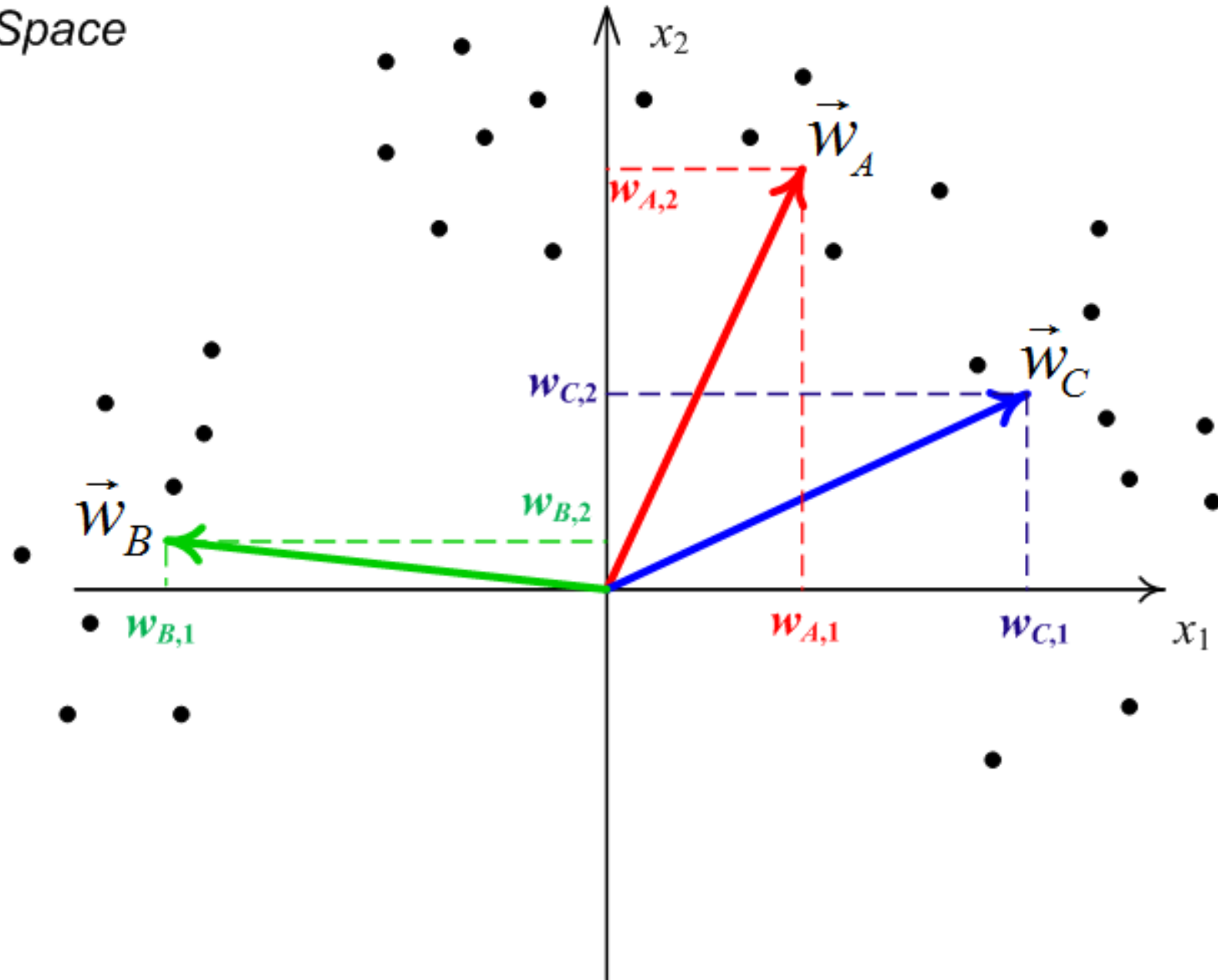
$$\vec{w}_B = (w_{B,1}, w_{B,2})$$

$$\vec{w}_C = (w_{C,1}, w_{C,2})$$

- Interpolation problem and RBFs
- RBF networks – hybrid learning
- Weight interpretation in the input space
- **Competitive mechanisms for unsupervised learning**

Competition – update of the input weights, example

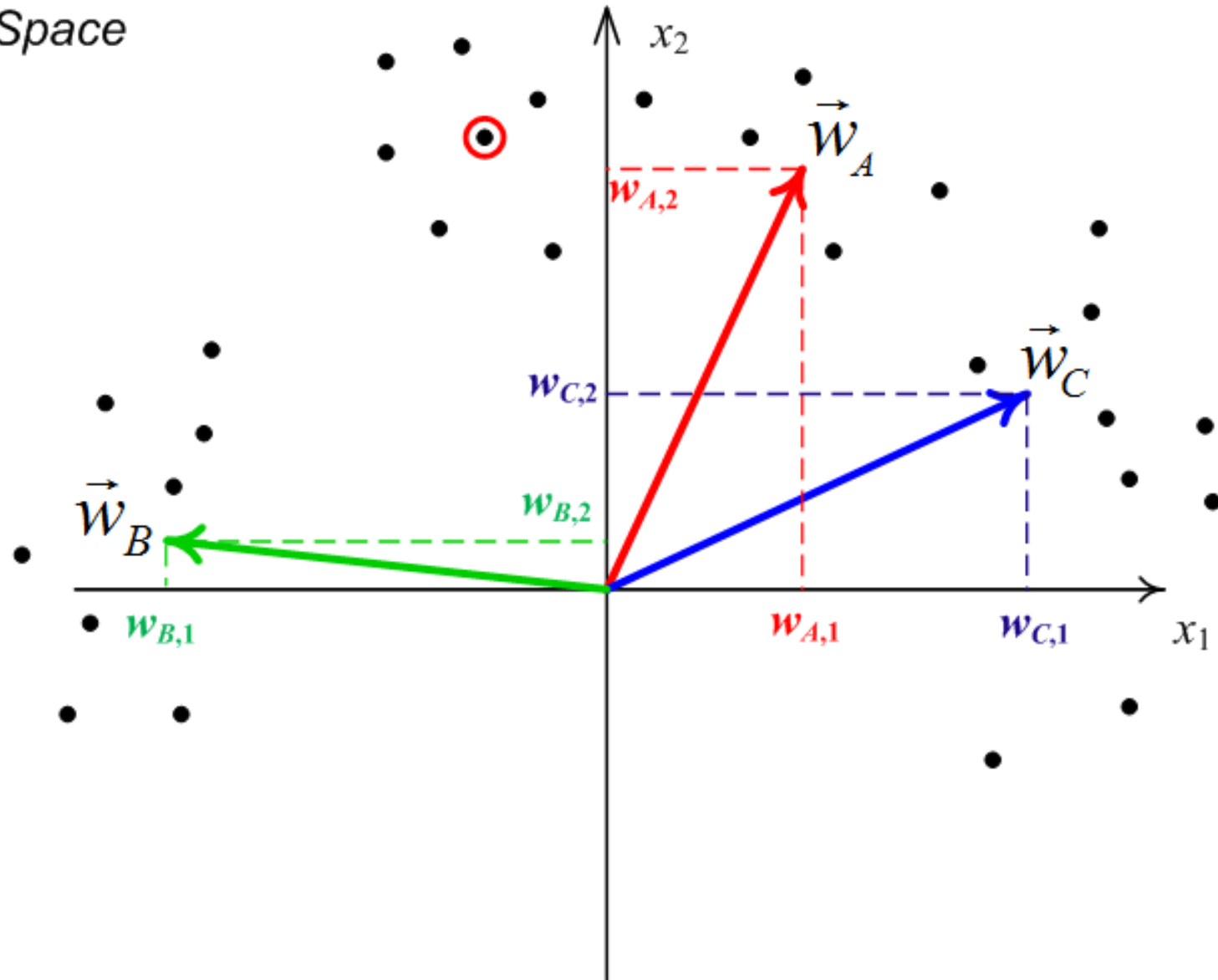
Input Space



- Interpolation problem and RBFs
- RBF networks – hybrid learning
- Weight interpretation in the input space
- **Competitive mechanisms for unsupervised learning**

Competition – update of the input weights

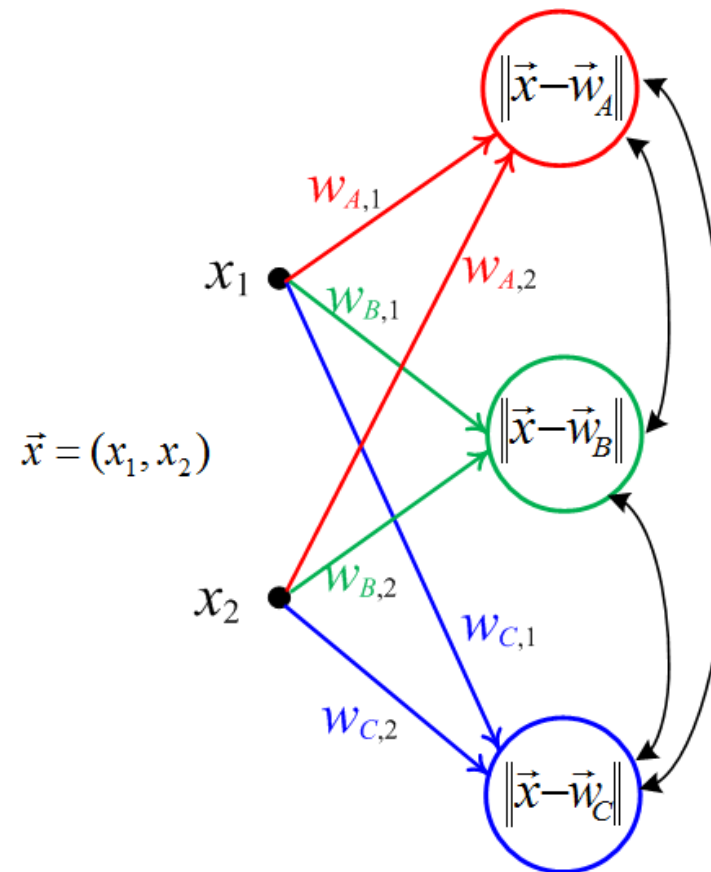
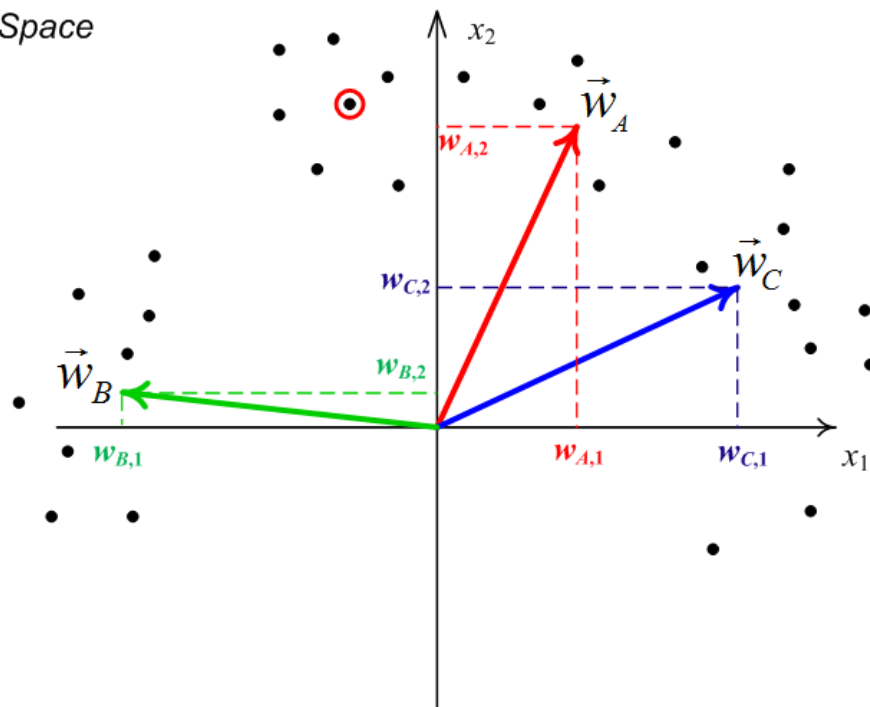
Input Space



- Interpolation problem and RBFs
- RBF networks – hybrid learning
- Weight interpretation in the input space
- **Competitive mechanisms for unsupervised learning**

Competition – update of the input weights

Input Space



If the **red** node w_A wins, then:

$$\Delta \vec{w}_A = \eta \vec{x}$$

OR

$$\Delta \vec{w}_A = \eta (\vec{x} - \vec{w}_A)$$

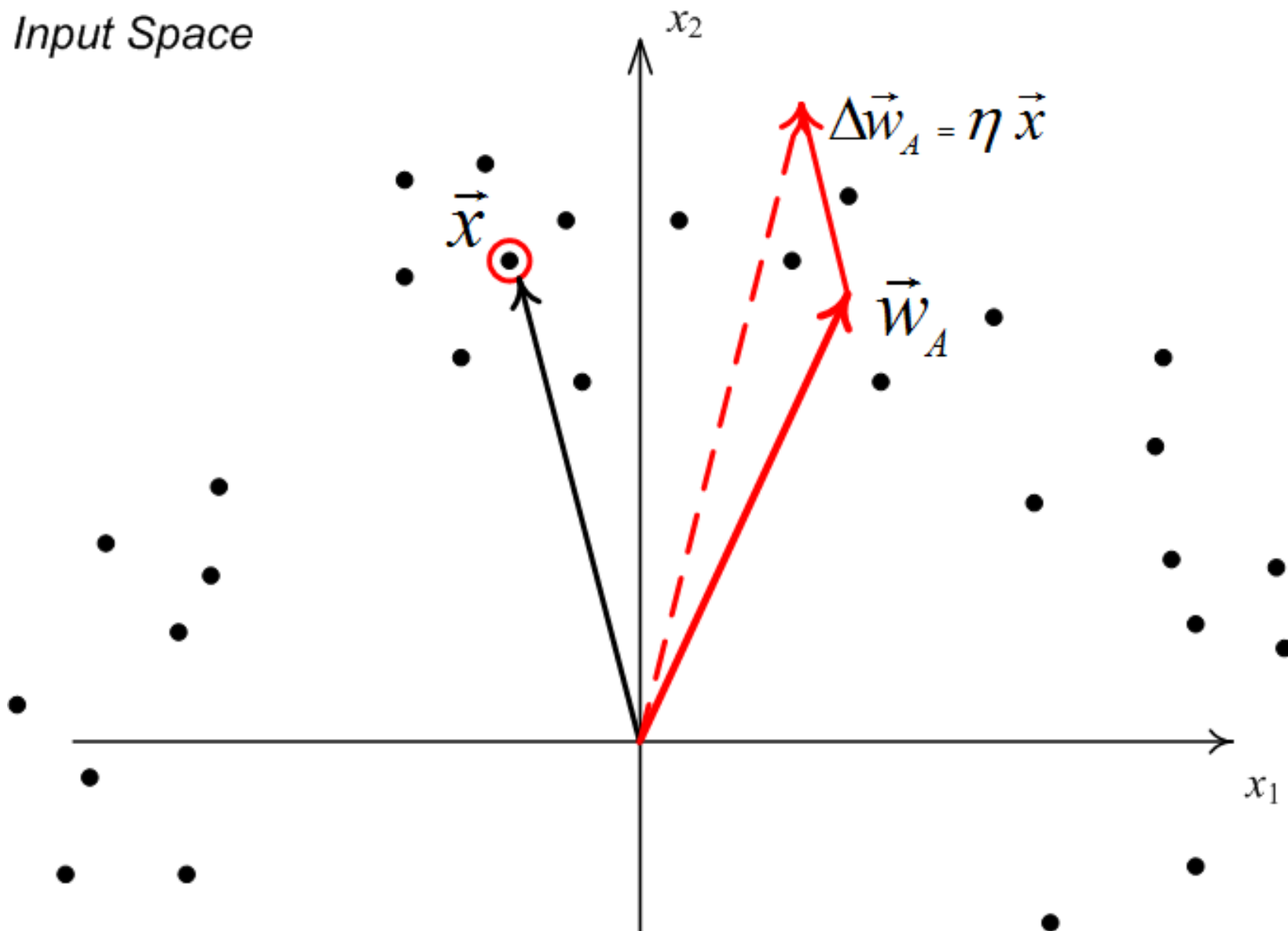
$$\vec{w}_A = (w_{A,1}, w_{A,2})$$

$$\vec{w}_B = (w_{B,1}, w_{B,2})$$

$$\vec{w}_C = (w_{C,1}, w_{C,2})$$

- Interpolation problem and RBFs
- RBF networks – hybrid learning
- Weight interpretation in the input space
- **Competitive mechanisms for unsupervised learning**

Competition – update of the input weights, ver. 1



- Interpolation problem and RBFs
- RBF networks – hybrid learning
- Weight interpretation in the input space
- **Competitive mechanisms for unsupervised learning**

Competition – update of the input weights, ver. 2

