Artificial Intelligence

Neural Networks

# Lesson 10: Radial Basis Function Networks

**Vincenzo Piuri**

Università degli Studi di Milano

# Contents

# **Radial Basis Function Networks** (1)

- A **Radial Basis Function Network (RBF)** is a feed-forward 3-layered neural network with radial basis functions as activation functions in the hidden layer

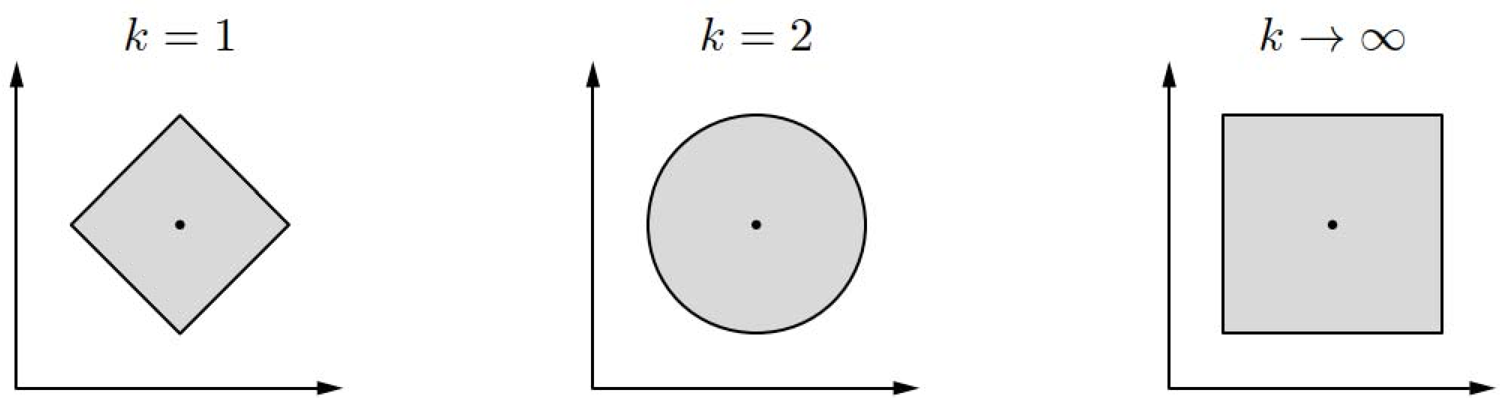
# Radial Basis Function Networks (1)

- The network input function of the output neurons is the weighted sum of their inputs
- The network input function of each hidden neuron is a distance function of the input vector and the weight vector
  - $\forall u \in U_{hidden}: f_{net}^{(u)}\left(\overrightarrow{w_u}, \overrightarrow{in_u}\right) = d(\overrightarrow{w_u}, \overrightarrow{in_u})$
  - $d(\vec{x}, \vec{y}) = 0 \iff \vec{x} = \vec{y}$
  - $d(\vec{x}, \vec{y}) = d(\vec{y}, \vec{x})$ (Symmetry)
  - $d(\vec{x}, \vec{z}) \leq d(\vec{x}, \vec{y}) + d(\vec{y}, \vec{z})$ (Triangle inequality)

© Vincenzo Piuri, Università degli Studi di Milano, Italy

# Radial Basis Function Networks (2)

- Distance functions: **Minkowski Family**
  - $d_k(\vec{x}, \vec{y}) = \left(\sum_{i=1}^{n} |x_i - y_i|^k\right)^{\frac{1}{k}}$
  - $k = 1$: Manhattan or city block distance
  - $k = 2$: Euclidean distance
  - …
  - $k \to \infty$: Maximum distance



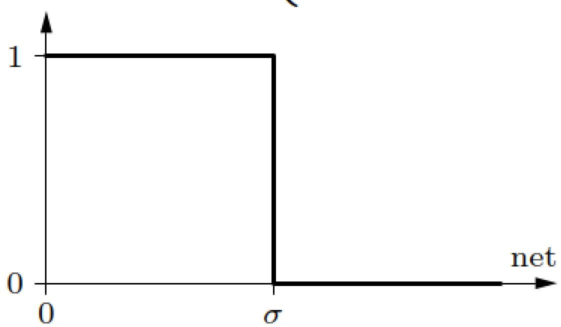
# Radial Basis Function Networks (3)

- The activation function of each output neuron is a **linear function**
- The activation function of each hidden neuron is a **radial function**
  - Monotonically decreasing function
    - $f: \mathbb{R}_0^+ \rightarrow [0,1]$ with $f(0) = 1$ and $\lim_{x \to \infty} f(x) = 0$
  - Size of the catchment region defined by the **reference radius** σ

# Radial Activation Functions

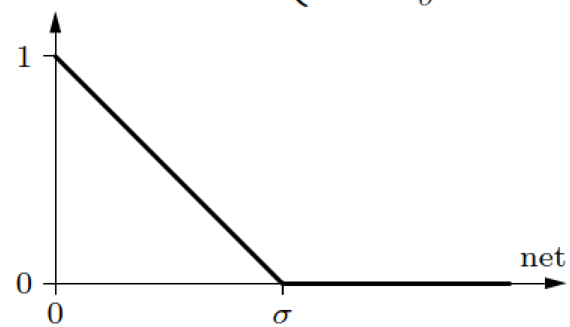
rectangle function:

$$f_{\text{act}}(\text{net}, \sigma) = \begin{cases} 0, & \text{if net} > \sigma, \\ 1, & \text{otherwise.} \end{cases}$$



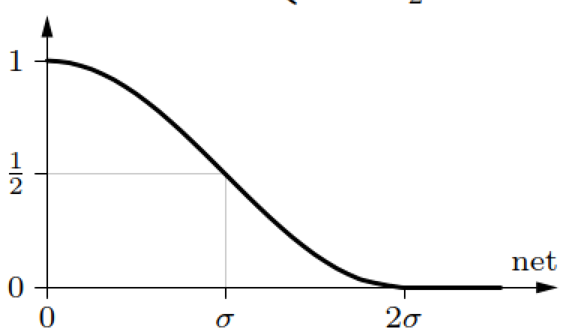

triangle function:

$$f_{\text{act}}(\text{net}, \sigma) = \begin{cases} 0, & \text{if net} > \sigma, \\ 1 - \frac{\text{net}}{\sigma}, & \text{otherwise.} \end{cases}$$



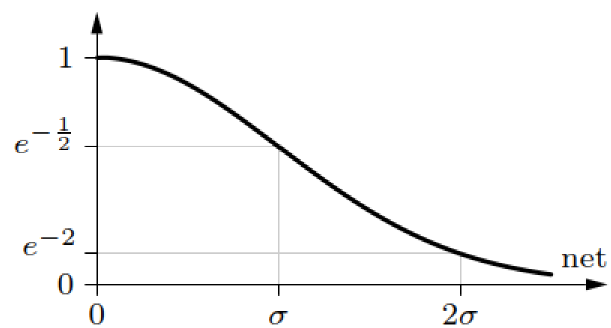

cosine until zero:

$$f_{\text{act}}(\text{net}, \sigma) = \begin{cases} 0, & \text{if net} > 2\sigma, \\ \frac{\cos\left(\frac{\pi}{2\sigma}\text{net}\right)+1}{2}, & \text{otherwise.} \end{cases}$$



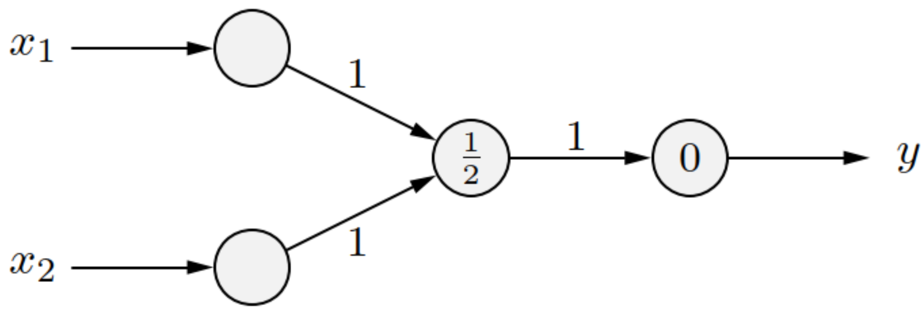

Gaussian function:

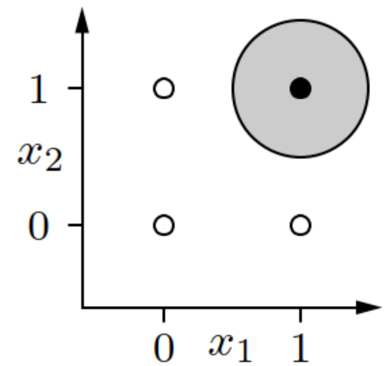
$$f_{\text{act}}(\text{net}, \sigma) = e^{-\frac{\text{net}^2}{2\sigma^2}}$$

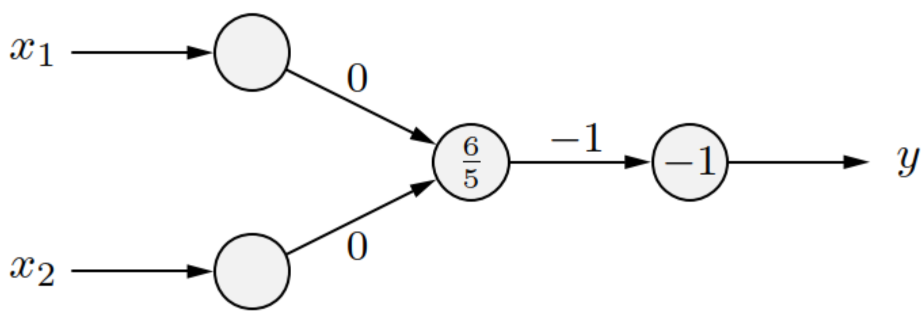


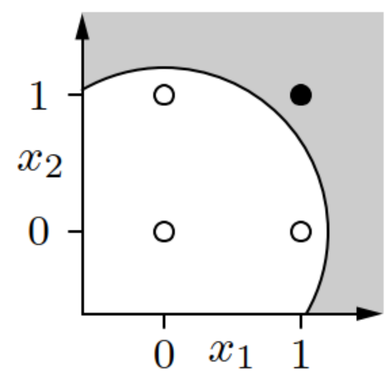
# Examples of RBFNs (1)

- Radial basis function networks for the conjunction $x_1 \wedge x_2$



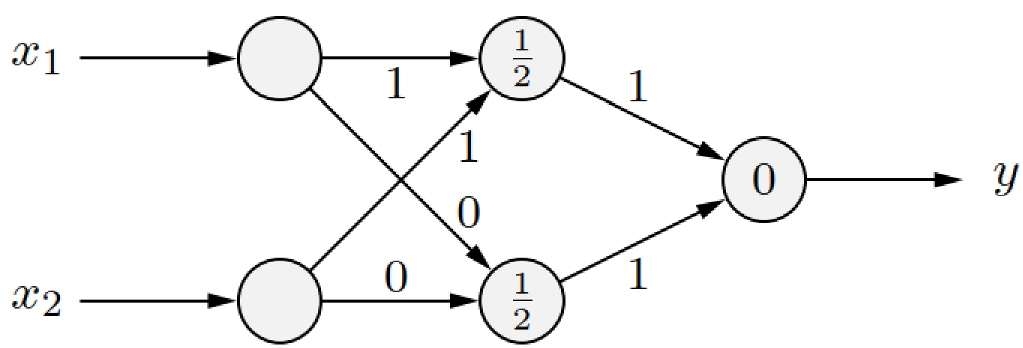



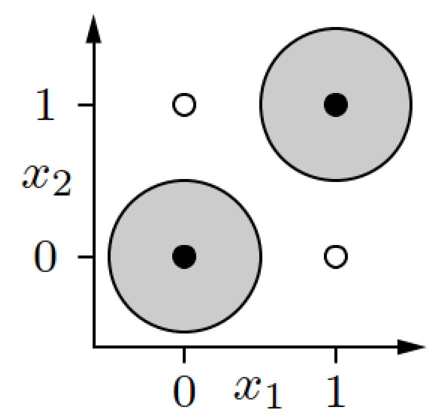

© Vincenzo Piuri, Università degli Studi di Milano, Italy

# Examples of RBFNs (2)

- Radial basis function networks for the biimplication $x_1 \leftrightarrow x_2$
  - Logical decomposition: $x_1 \leftrightarrow x_2 \equiv (x_1 \wedge x_2) \vee \neg(x_1 \vee x_2)$

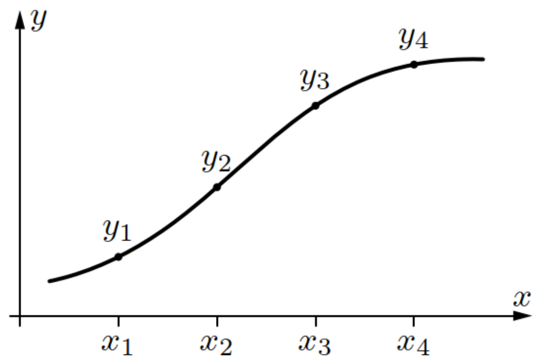
$$x_1 \leftrightarrow x_2 \quad \equiv \quad (x_1 \wedge x_2) \vee \neg(x_1 \vee x_2)$$

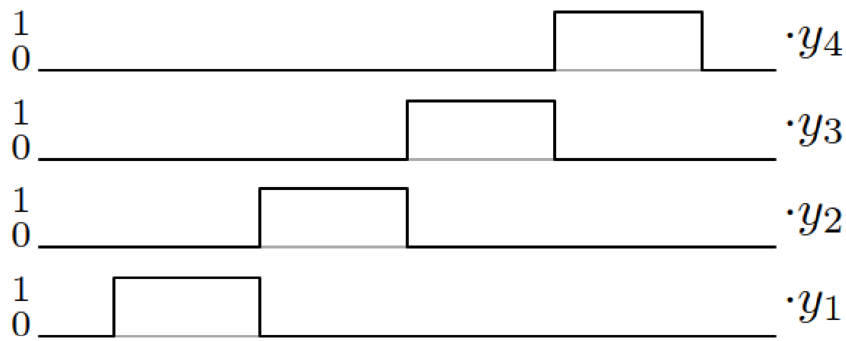


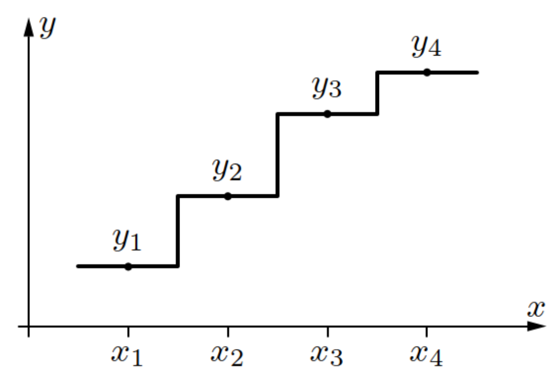



# Function Approximation (1)

- Approximation of a function
  by rectangular pulses



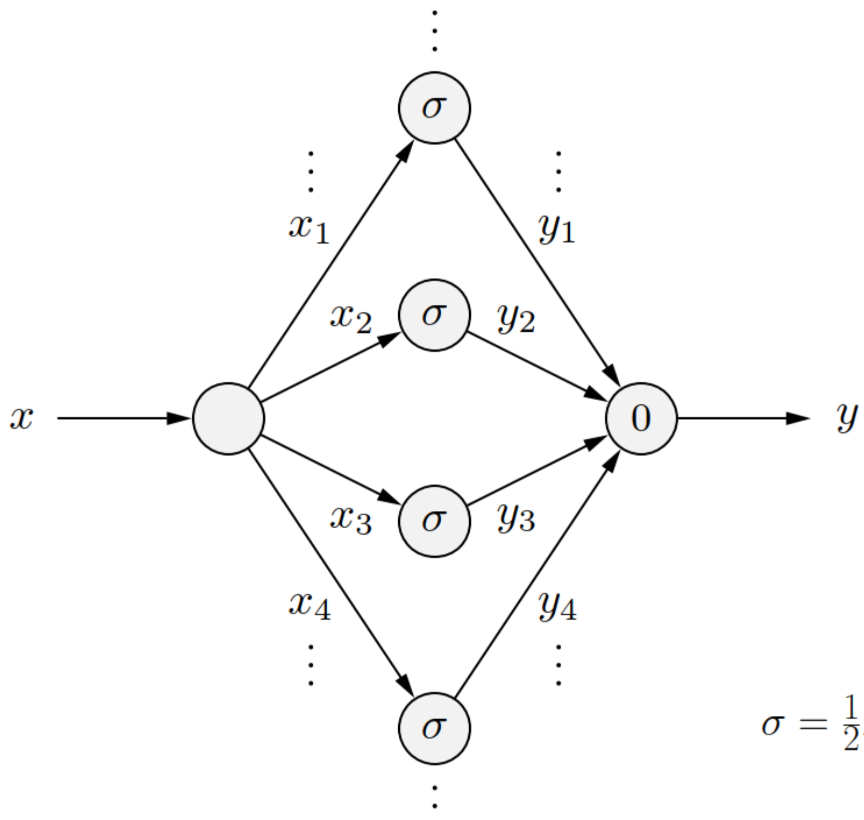




© Vincenzo Piuri, Università degli Studi di Milano, Italy

# **Function Approximation** (2)

- Each pulse can be represented by a neuron of a radial basis function network

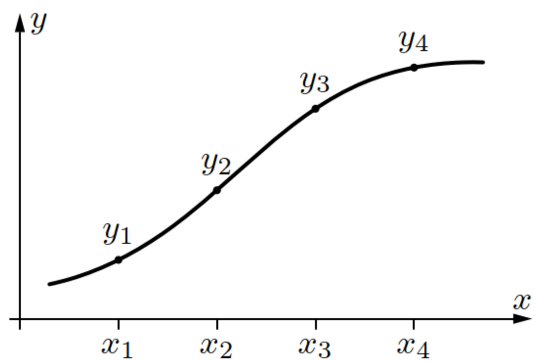


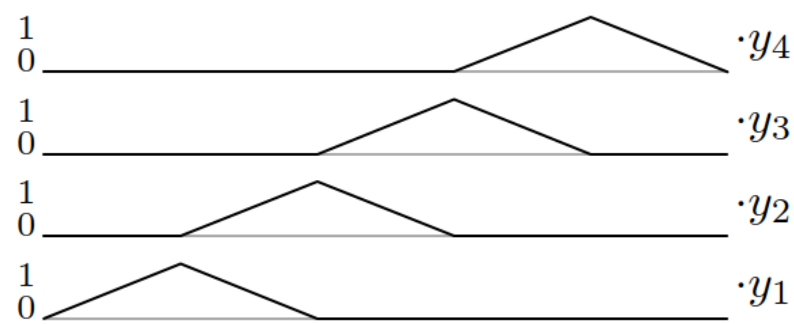

$$\sigma = \frac{1}{2}\Delta x = \frac{1}{2}(x_{i+1} - x_i)$$

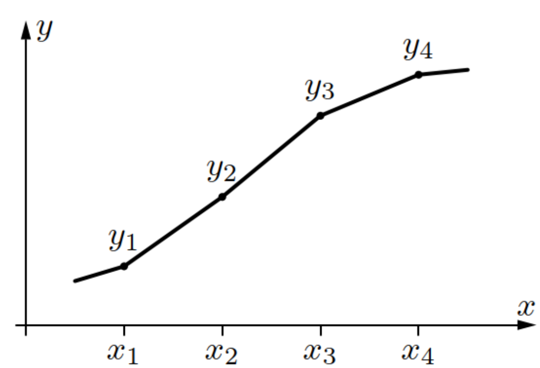
© Vincenzo Piuri, Università degli Studi di Milano, Italy

# **Function Approximation** (3)

- Approximation of a function
  by triangular pulses

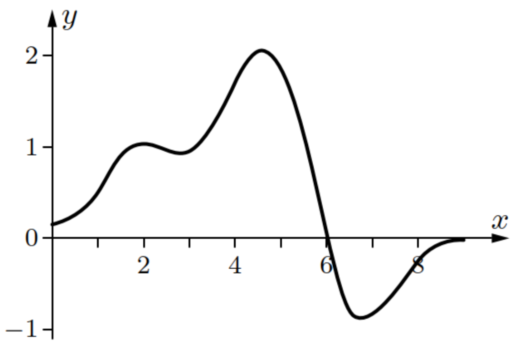


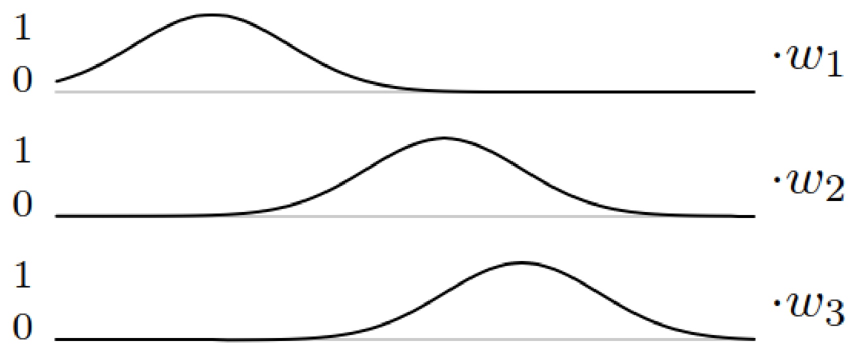



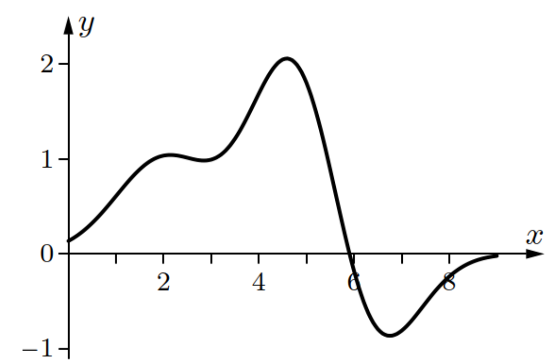



# **Gaussian functions** (1)

- Approximation of a function by Gaussian functions

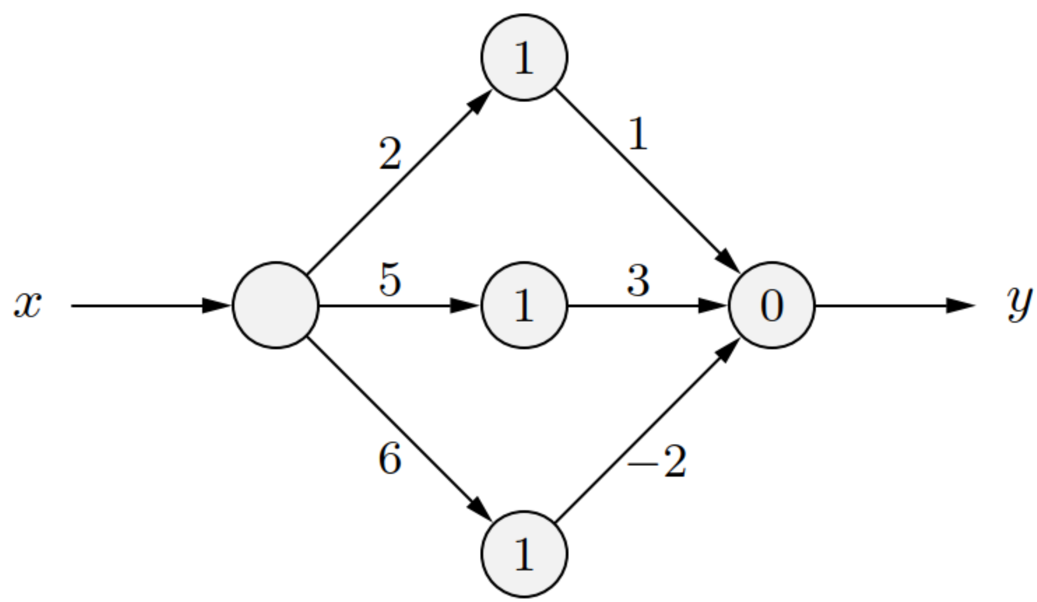


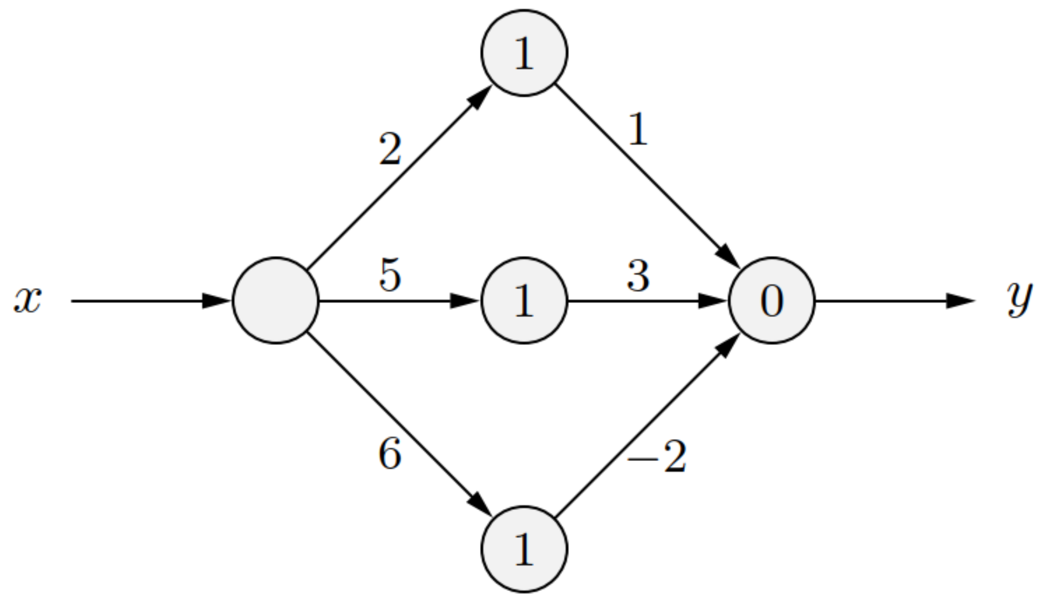
# **Gaussian functions** (2)

- Radial basis function network for a sum of three Gaussian functions



# **Gaussian functions** (3)

- The weights of the connections from the input neuron to the hidden neurons determine the locations of the Gaussian functions.



## **Gaussian functions** (4)

- The weights of the connections from the hidden neurons to the output neuron determine the height/direction (upward or downward) of the Gaussian functions.