

Teoretické základy AI

Neuronové sítě a strojové učení

Co je to umělá inteligence

Funkční emulace lidského myšlení

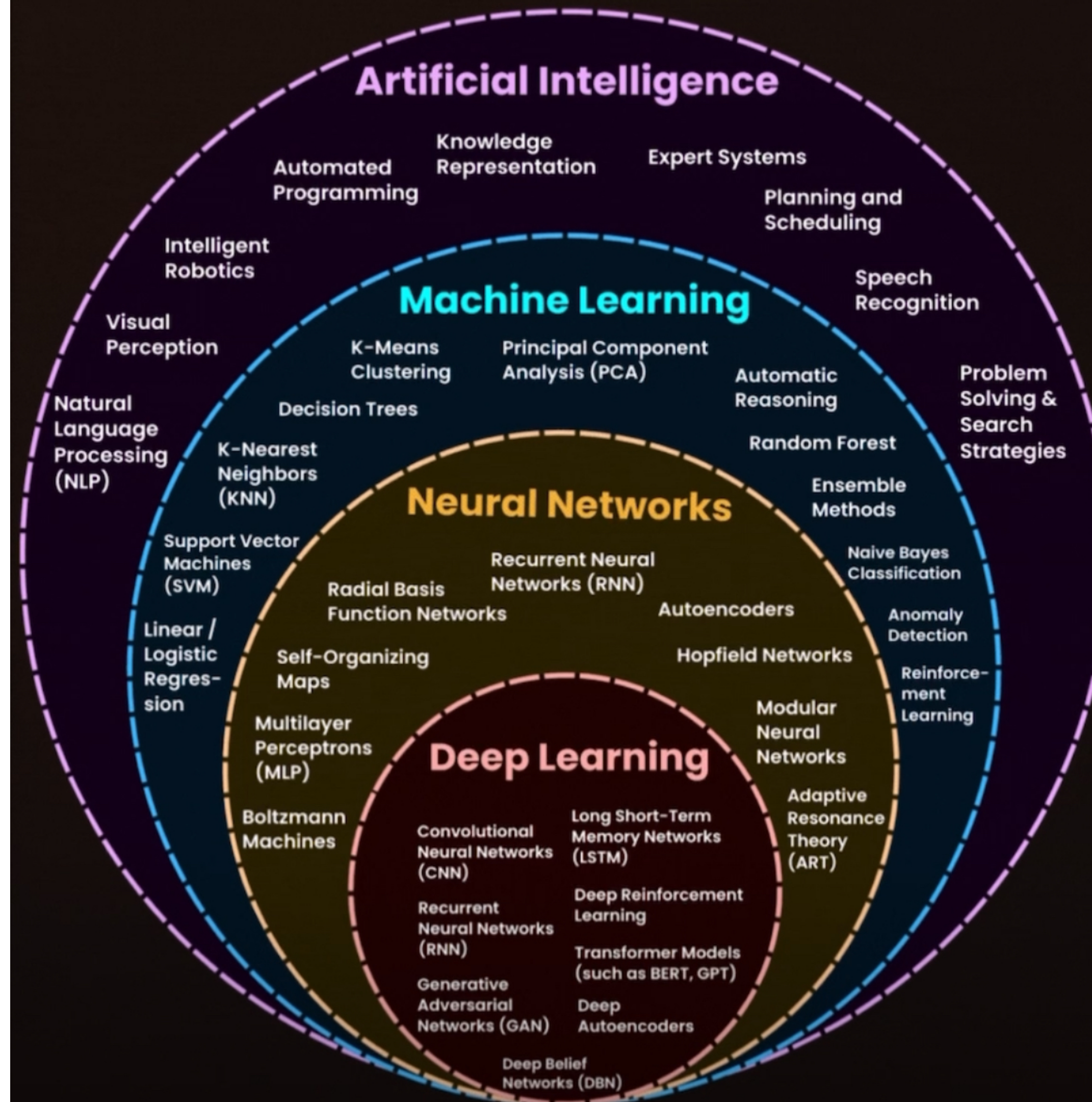
- Umělá inteligence je obor informatiky zabývající se tvorbou systémů řešících komplexní úlohy jako je **rozpoznávání či klasifikace**, např. v oblastech **zpracování obrazu** či zpracování **psaného textu či mluveného jazyka**, nebo **plánování či řízení** na základě **zpracování velkých objemů dat**.
- **Učení a adaptace:** AI systémy mohou adaptovat a zdokonalovat své schopnosti na základě získaných zkušeností a dat.
- ...
- Alternativně definice srovnáním: **Umělá inteligence je horší než přirozená blbost.**

Co je to umělá inteligence

Funkční emulace lidského myšlení

- Umělá inteligence je obor informatiky zabývající se tvorbou systémů řešících komplexní úlohy jako je **rozpoznávání či klasifikace**, např. v oblastech **zpracování obrazu** či zpracování **psaného textu či mluveného jazyka**, nebo **plánování či řízení** na základě **zpracování velkých objemů dat**.
- **Učení a adaptace:** AI systémy mohou adaptovat a zdokonalovat své schopnosti na základě získaných zkušeností a dat.
- ...
- ~~Alternativně definice srovnáním: Umělá inteligence je horší než přirozená blbost.~~

AI EXPLAINED



Strojové učení

Učit se, znamená se zlepšovat. Platí to i v tomto případě?

- **Učení s učitelem:** učení probíhá na páru vstupních dat a odpovídajících výstupních dat. Cílem je naučit model (obvykle neuronovou síť), jak mapovat vstupy na výstupy.
- **Učení bez učitele:** modely analyzují data bez známých výstupních hodnot. Cílem je najít vzory, seskupení nebo struktury v datech.
- **Nebezpečí manipulativního učení!**

Ekonomické hledisko

Kdo pozná listonoše nejen lépe,
ale i levněji?

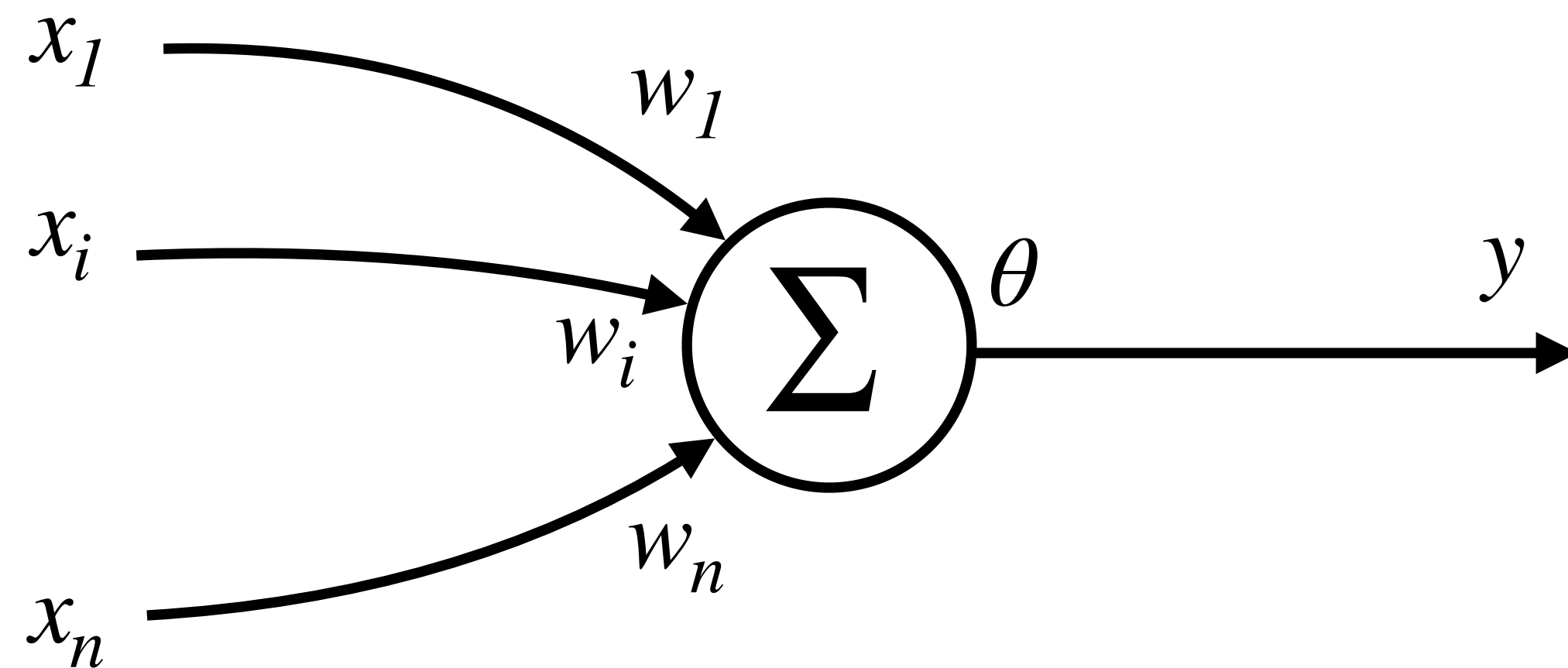
- Oswald, za miskou granulí ...
- IT4I za cenu řady MWh
- **AI bude generovat vysoké energetické nároky na výpočetní systémy!**



Trocha teorie (nikoho nezabije 🤗)

Perceptron

Nejjednodušší neuronová síť (1957: Frank Rosenblatt)



Potenciál neuronu

$$z = \sum_{i=1}^n w_i x_i$$

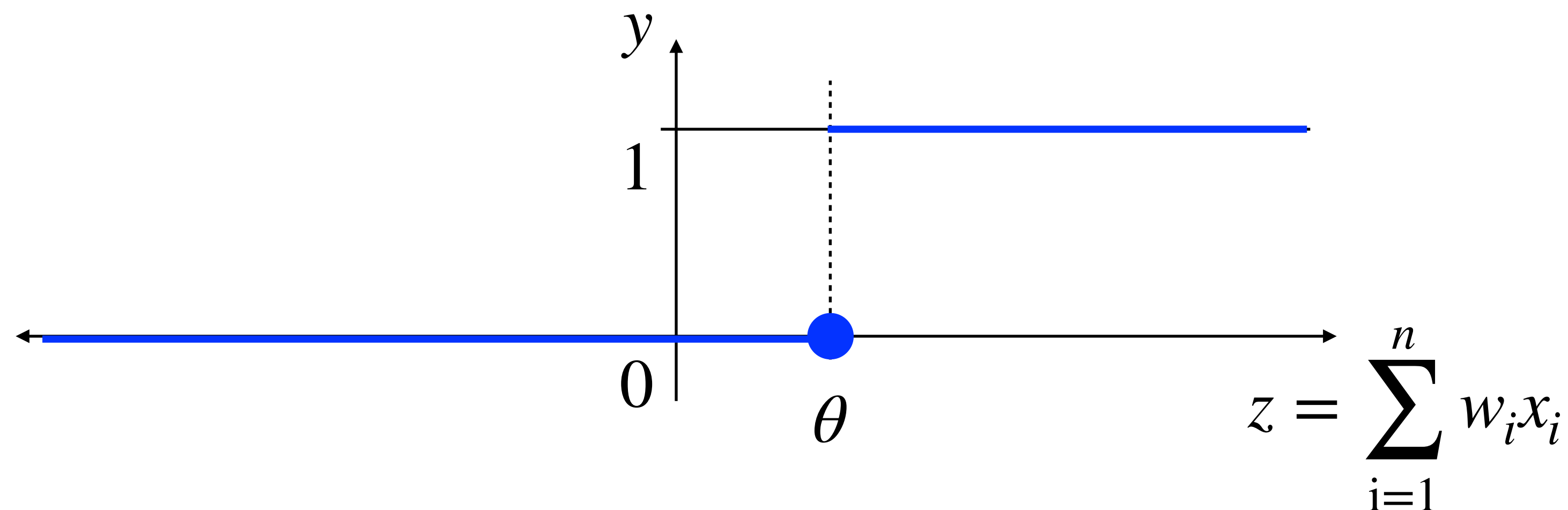
Excitace neuronu

$$y = \begin{cases} 1 & \text{if } z > \theta \\ 0 & \text{if } z \leq \theta \end{cases}$$

$$x_i \in [0, 1]$$

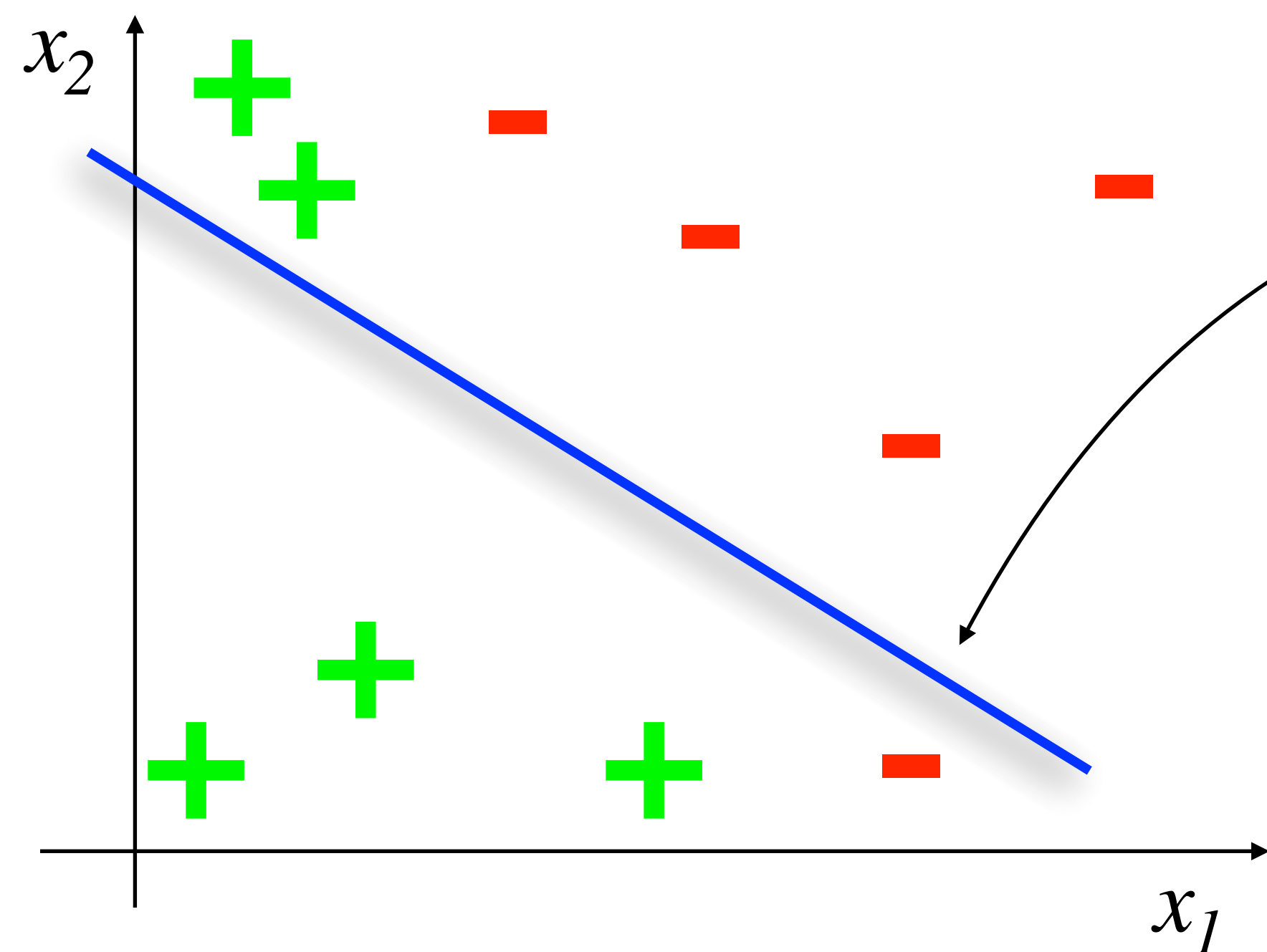
$$w_i \in \mathbb{R}$$

$$y \in \{0, 1\}$$



Učení neuronové sítě

Geometrická reprezentace problému



Rovnice přímky

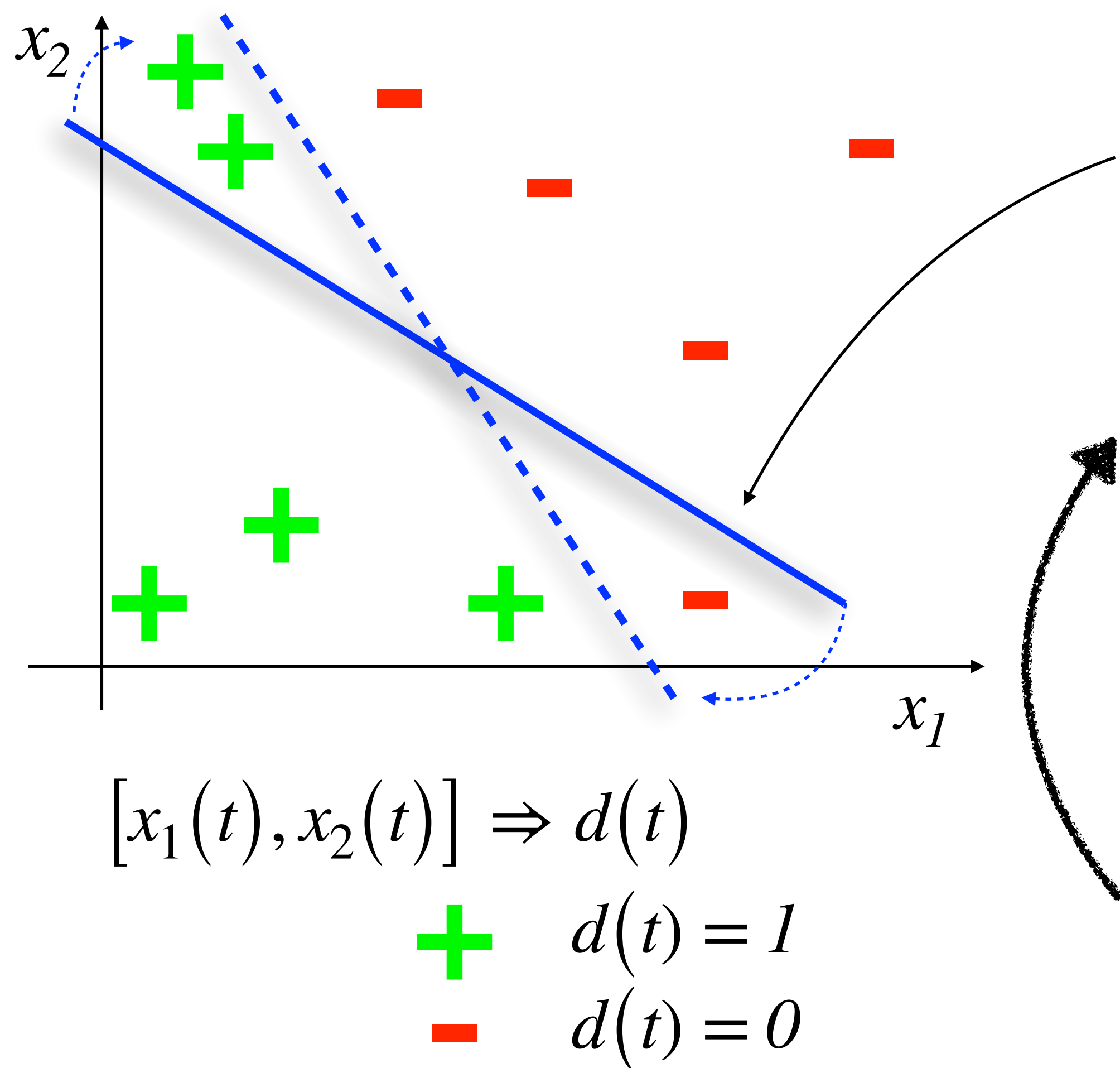
$$\sum_{i=0}^n w_i x_i = 0$$

$$[x_1(t), x_2(t)] \Rightarrow d(t)$$

$$\begin{array}{ll} + & d(t) = 1 \\ - & d(t) = 0 \end{array}$$

Učení neuronové sítě

Geometrická reprezentace problému

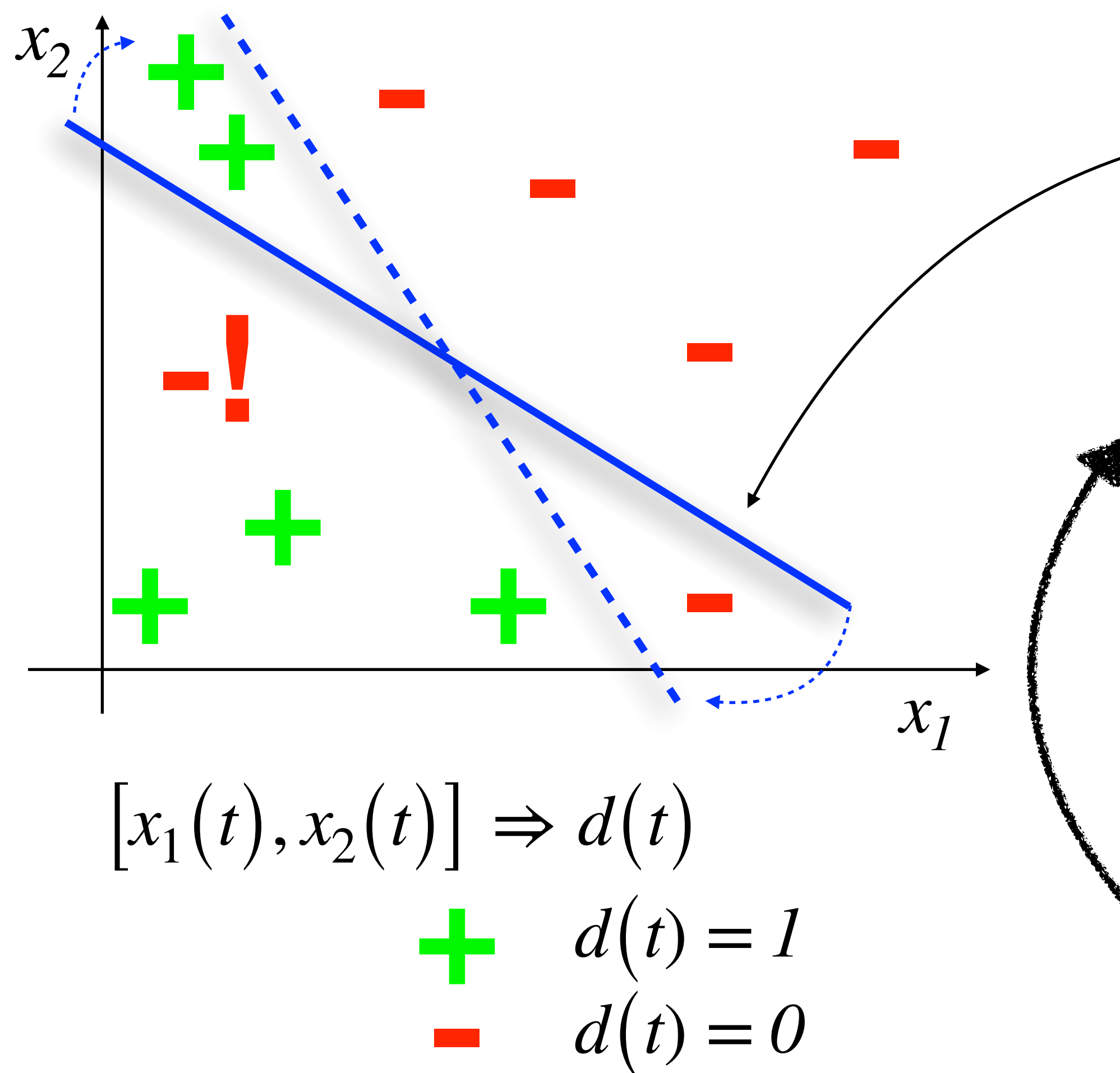


Rovnice přímky $\sum_{i=0}^n w_i x_i = 0$

1. Inicializace vah w_i na čísla blízké 0
2. Předložení prvku trénovací množiny
 $[x_1(t), x_2(t)] \Rightarrow d(t)$
3. Porovnání požad. a skutečné odezvy
 $\Delta(t) = d(t) - y(t)$
4. Adaptace vah neuronu na novou hodnotu
 $w_i(t+1) = w_i(t) + \eta \cdot \Delta(t) \cdot x_i(t)$

Učení neuronové sítě

Geometrická reprezentace problému



Rovnice přímky $\sum_{i=0}^n w_i x_i = 0$

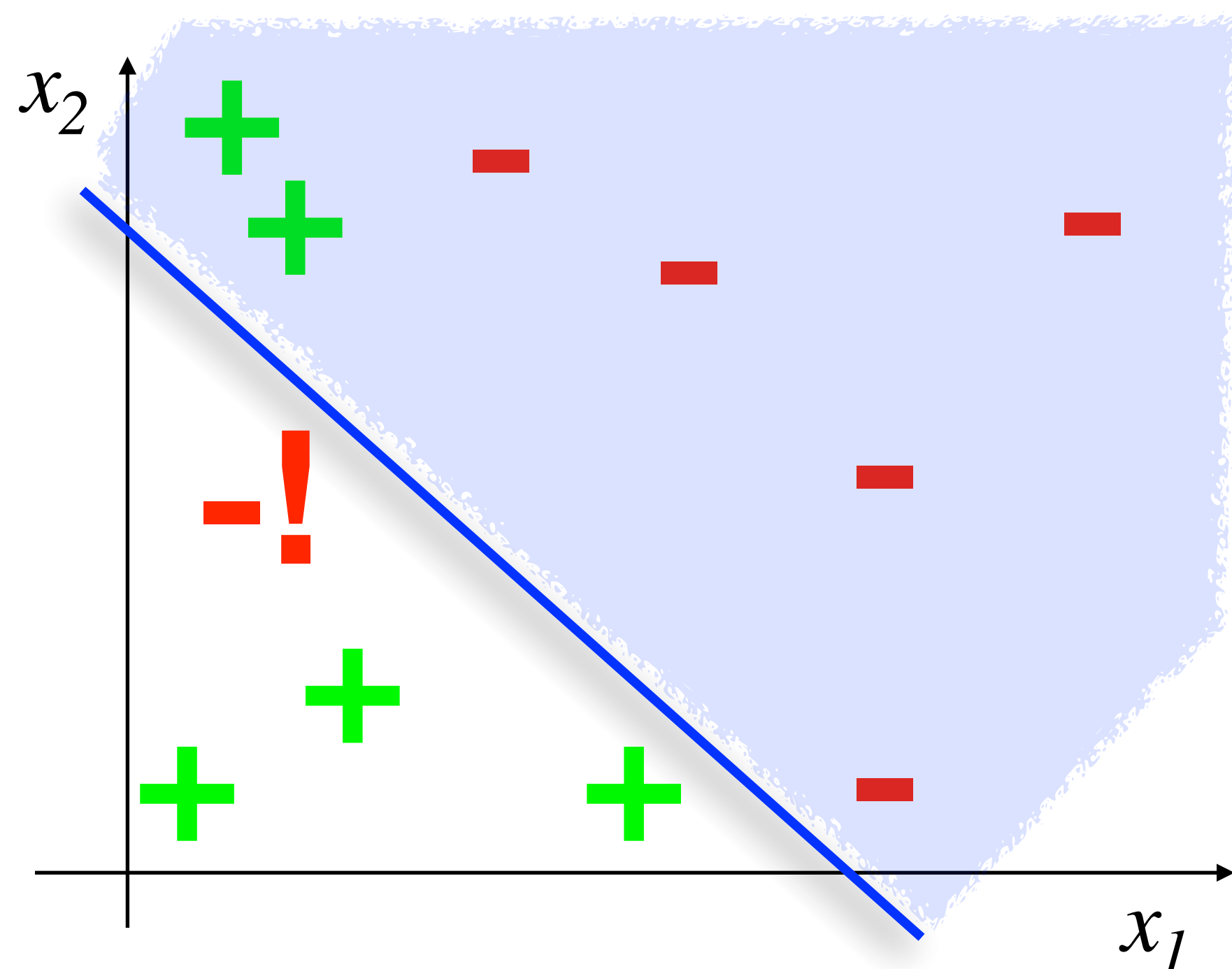
1. Inicializace vah w_i na čísla blízké 0
2. Předložení prvku trénovací množiny
 $[x_1(t), x_2(t)] \Rightarrow d(t)$
3. Porovnání požad. a skutečné odezvy
 $\Delta(t) = d(t) - y(t)$
4. Adaptace vah neuronu na novou hodnotu
 $w_i(t+1) = w_i(t) + \eta \cdot \Delta(t) \cdot x_i(t)$

Příklad: Perceptron

<https://dev.ivondrak.cz/notebooks/PerceptronNets/perceptron.ipynb>

Vícevrstvé neuronové sítě

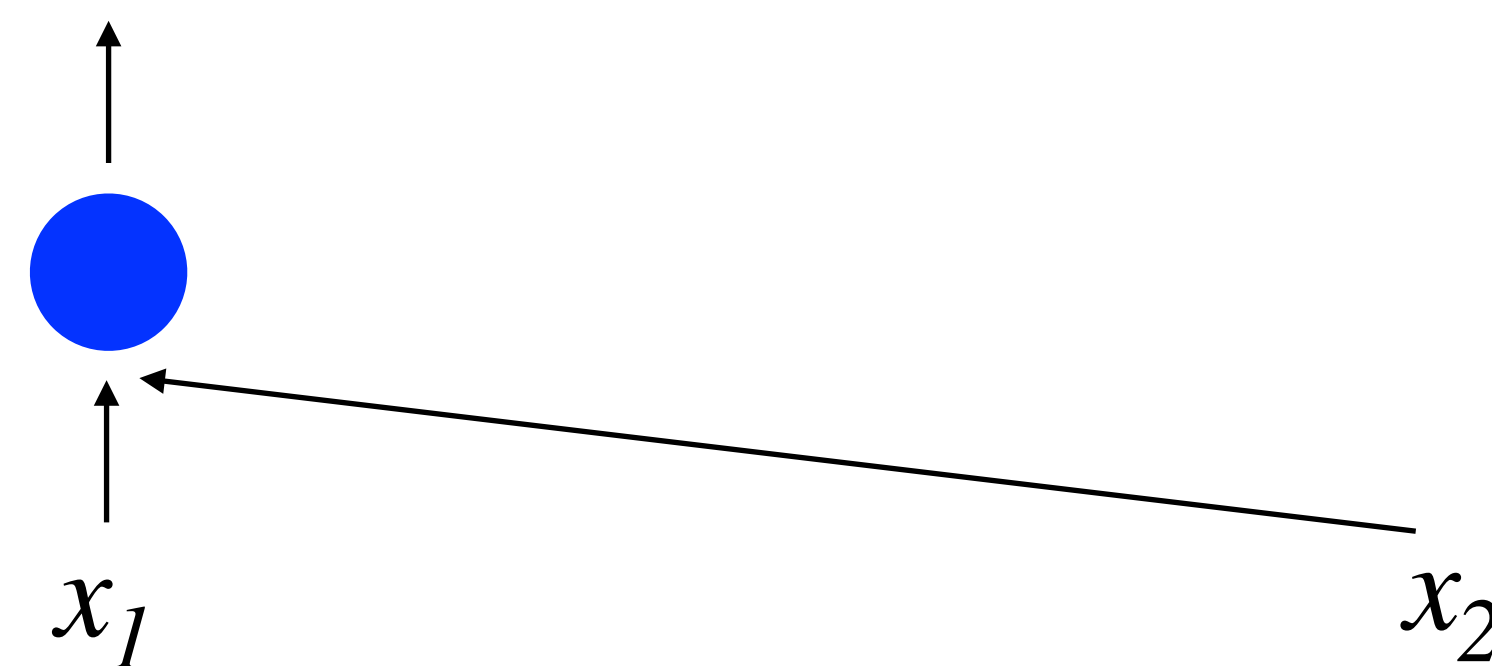
Svět není lineární 🙄



$$[x_1(t), x_2(t)] \Rightarrow d(t)$$

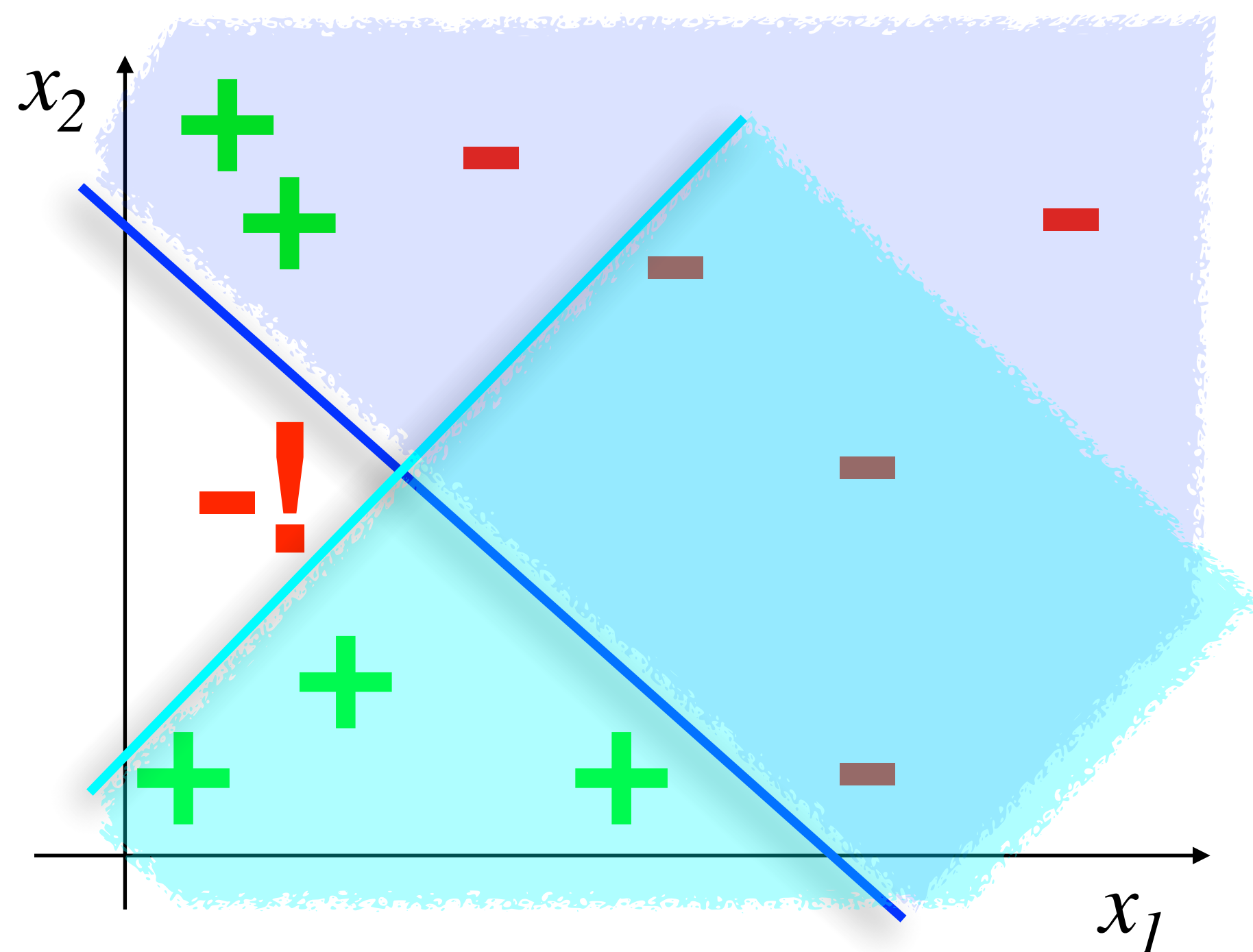
$$\begin{array}{ll} + & d(t) = 1 \\ - & d(t) = 0 \end{array}$$

Perceptron rozpoznává pouze **lineárně separabilní oblasti!**



Vícevrstvé neuronové sítě

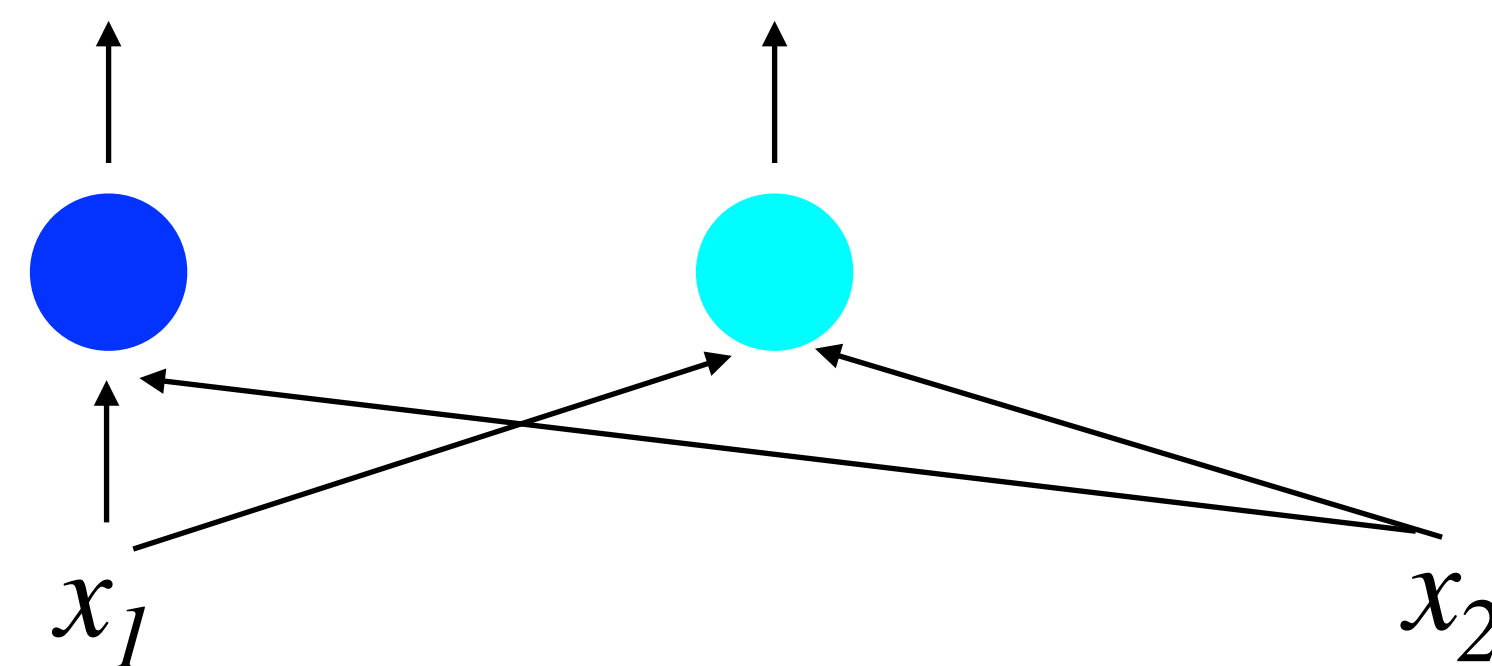
Svět není lineární 🙄



$$[x_1(t), x_2(t)] \Rightarrow d(t)$$

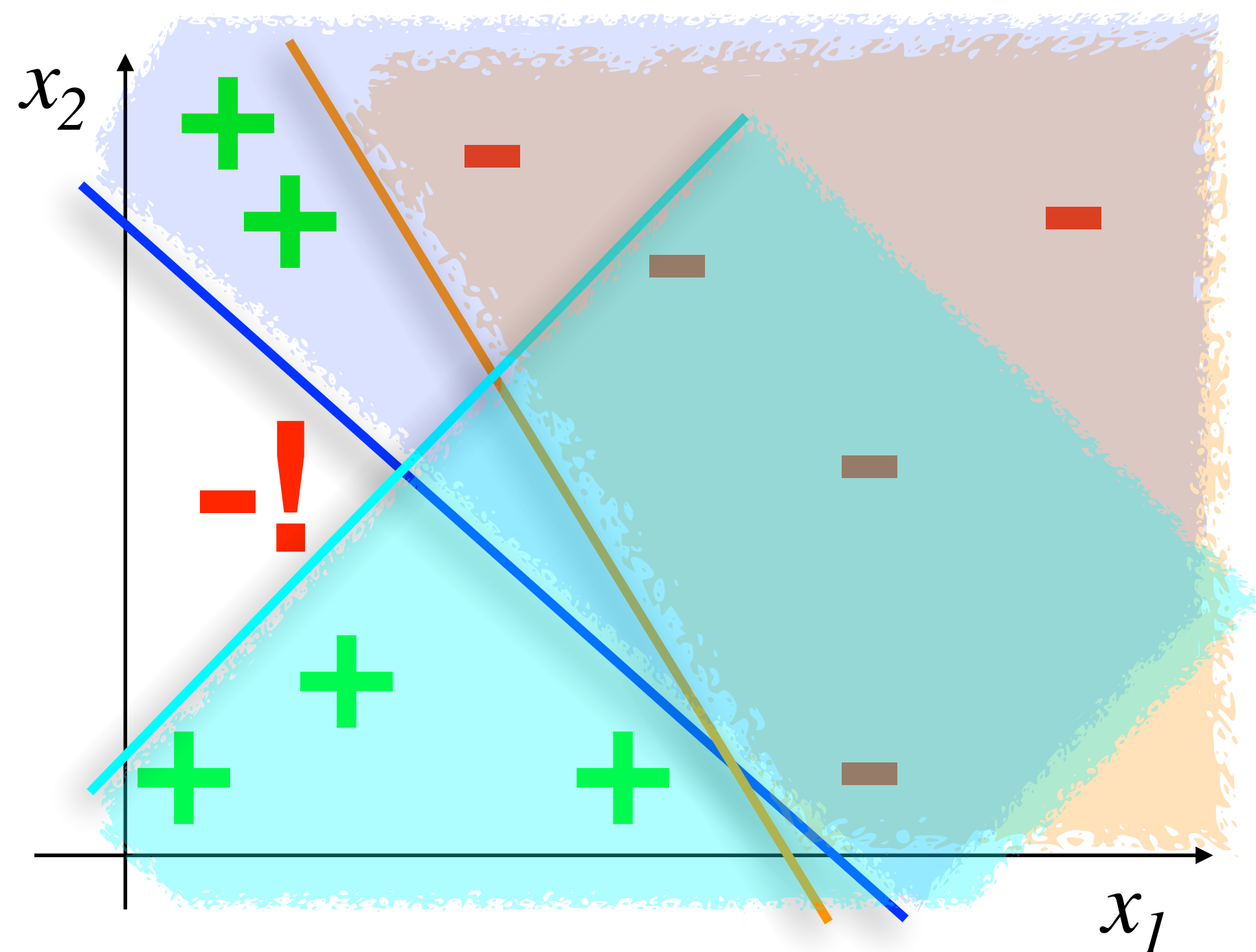
$$\begin{aligned} + & d(t) = 1 \\ - & d(t) = 0 \end{aligned}$$

Přidáním perceptronů dosáhneme
rozpoznat další takové oblasti.



Vícevrstvé neuronové sítě

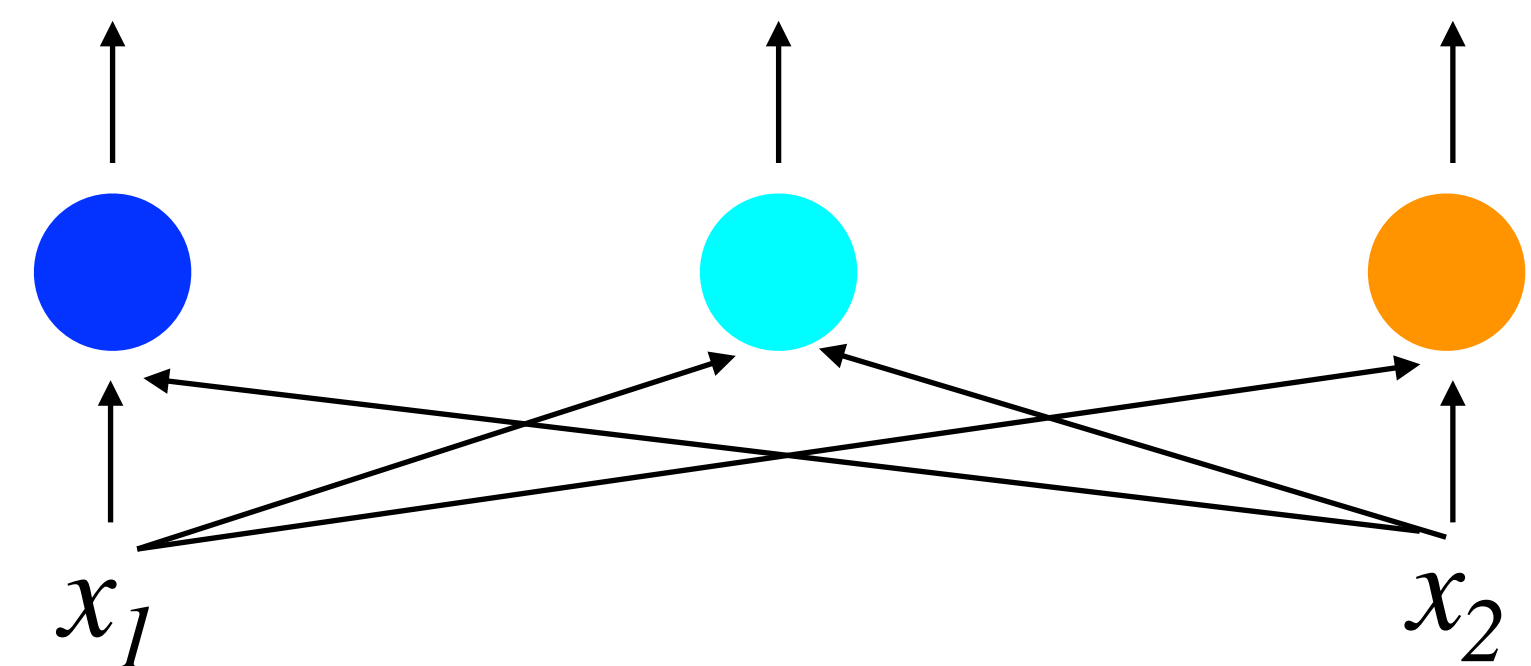
Svět není lineární 🙄



$$[x_1(t), x_2(t)] \Rightarrow d(t)$$

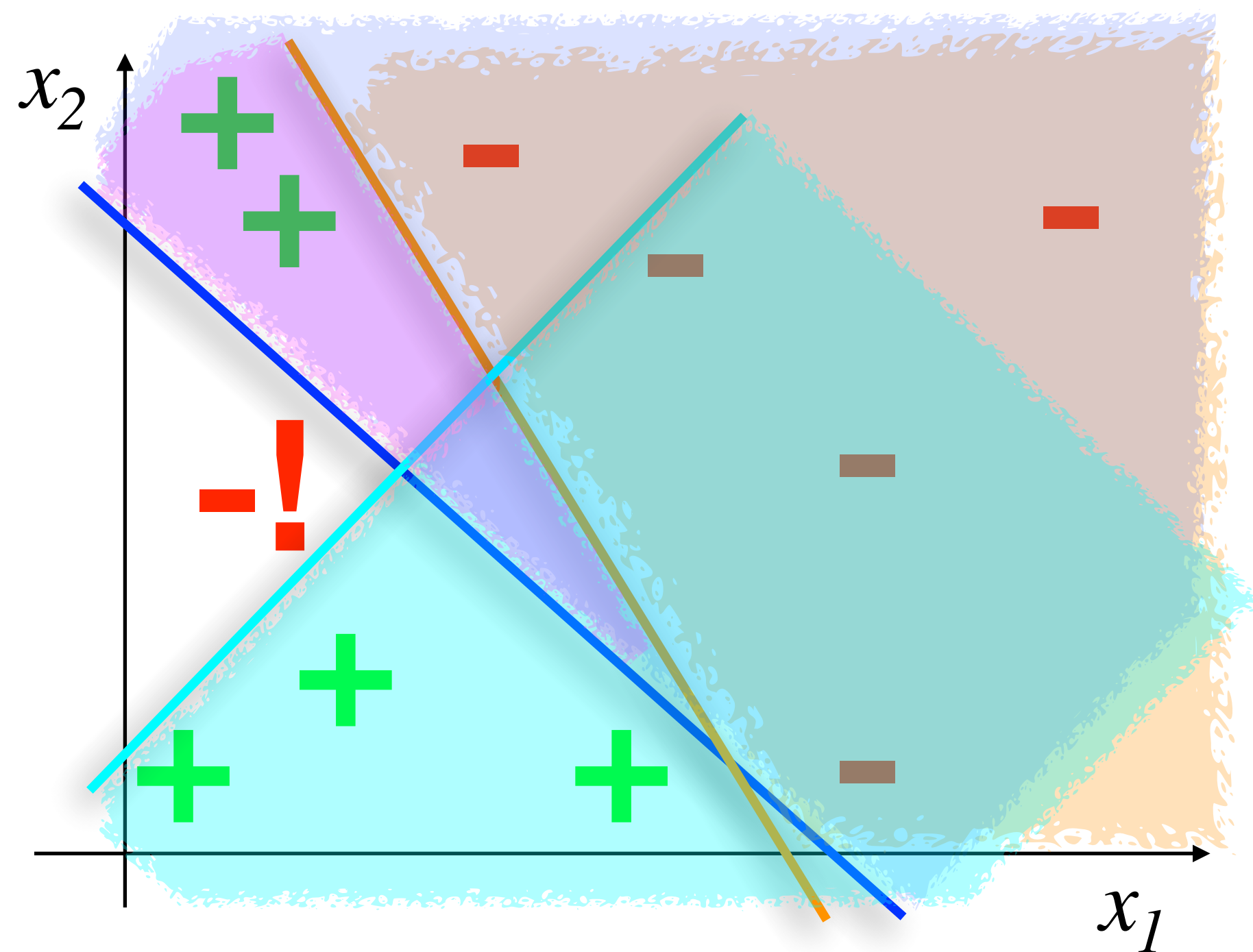
$$\begin{aligned} + & d(t) = 1 \\ - & d(t) = 0 \end{aligned}$$

Přidáním perceptronů dosáhneme
rozpoznat další takové oblasti.



Vícevrstvé neuronové sítě

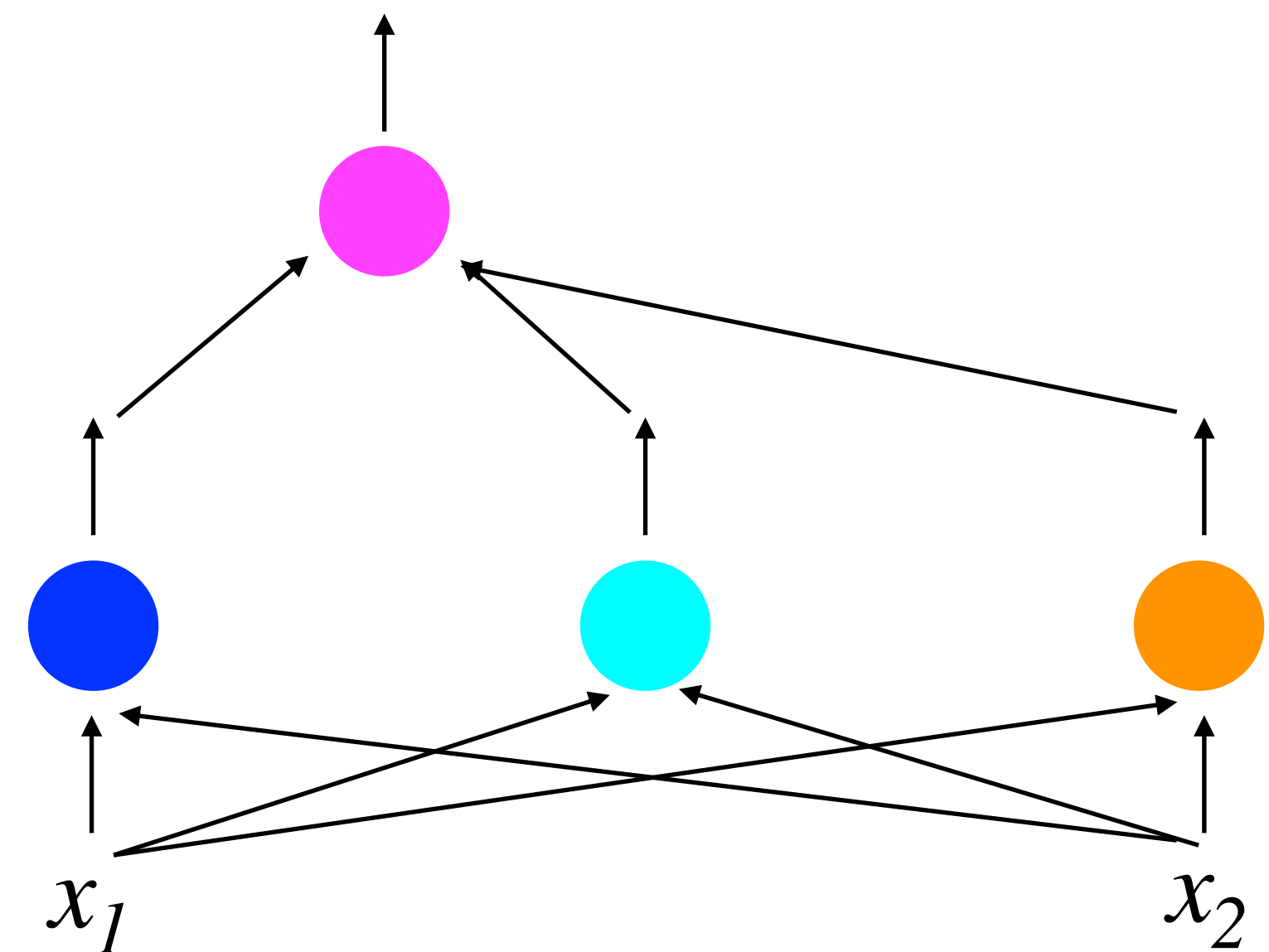
Svět není lineární 🙄



$$[x_1(t), x_2(t)] \Rightarrow d(t)$$

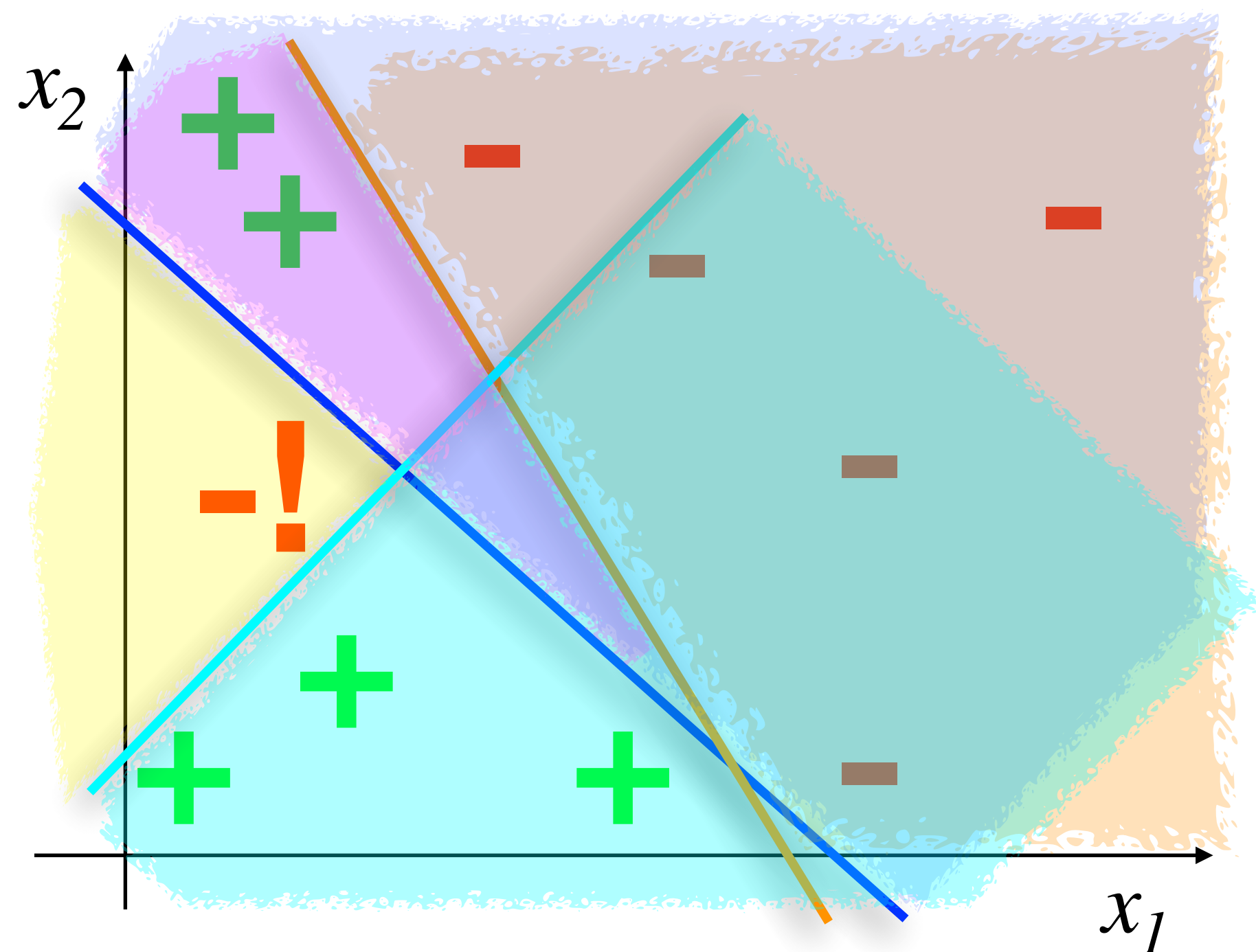
$$\begin{aligned} + & d(t) = 1 \\ - & d(t) = 0 \end{aligned}$$

Přidáním další vrstvy neuronů zajistíme rozpoznání **konvexních oblastí**.



Vícevrstvé neuronové sítě

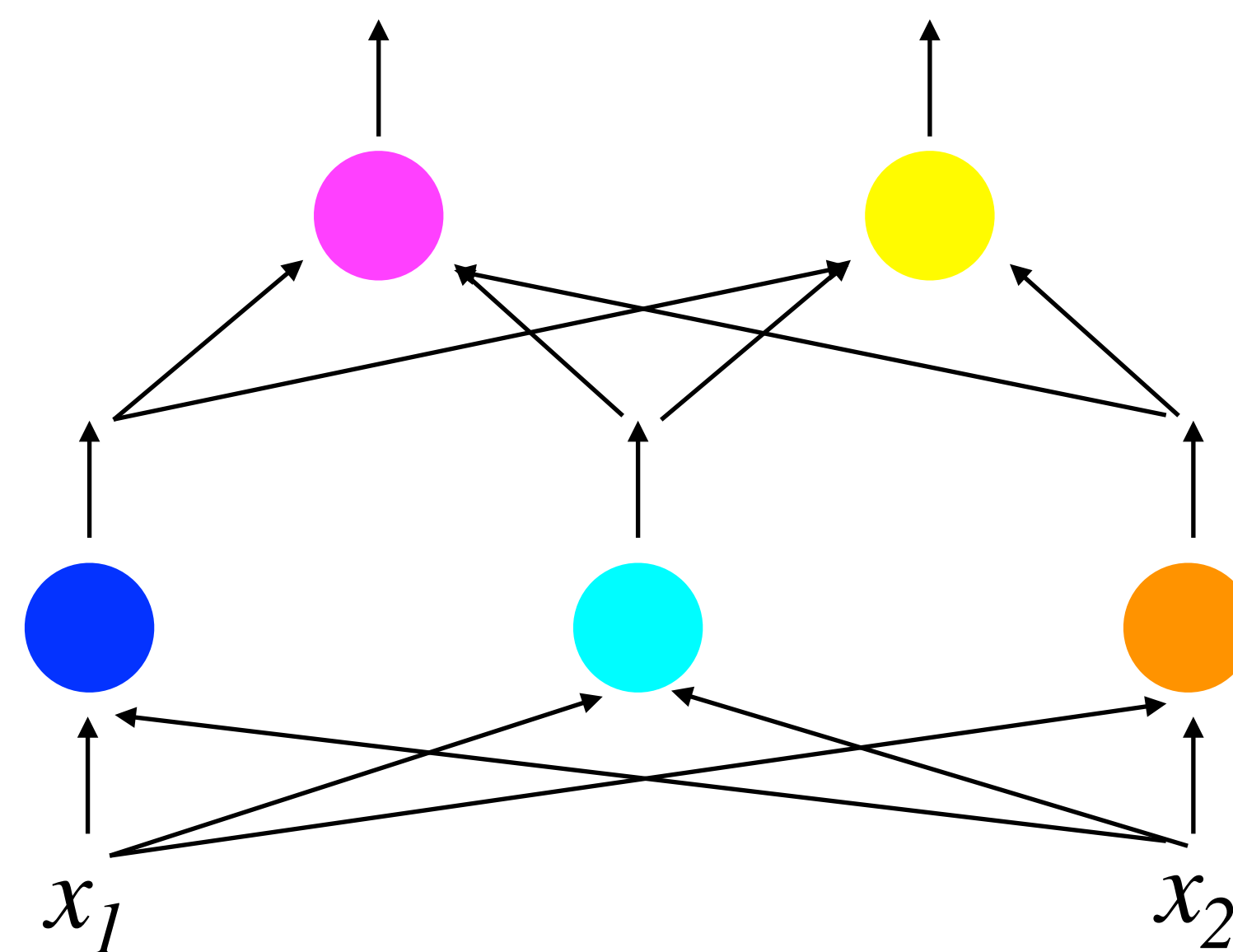
Svět není lineární 🙄



$$[x_1(t), x_2(t)] \Rightarrow d(t)$$

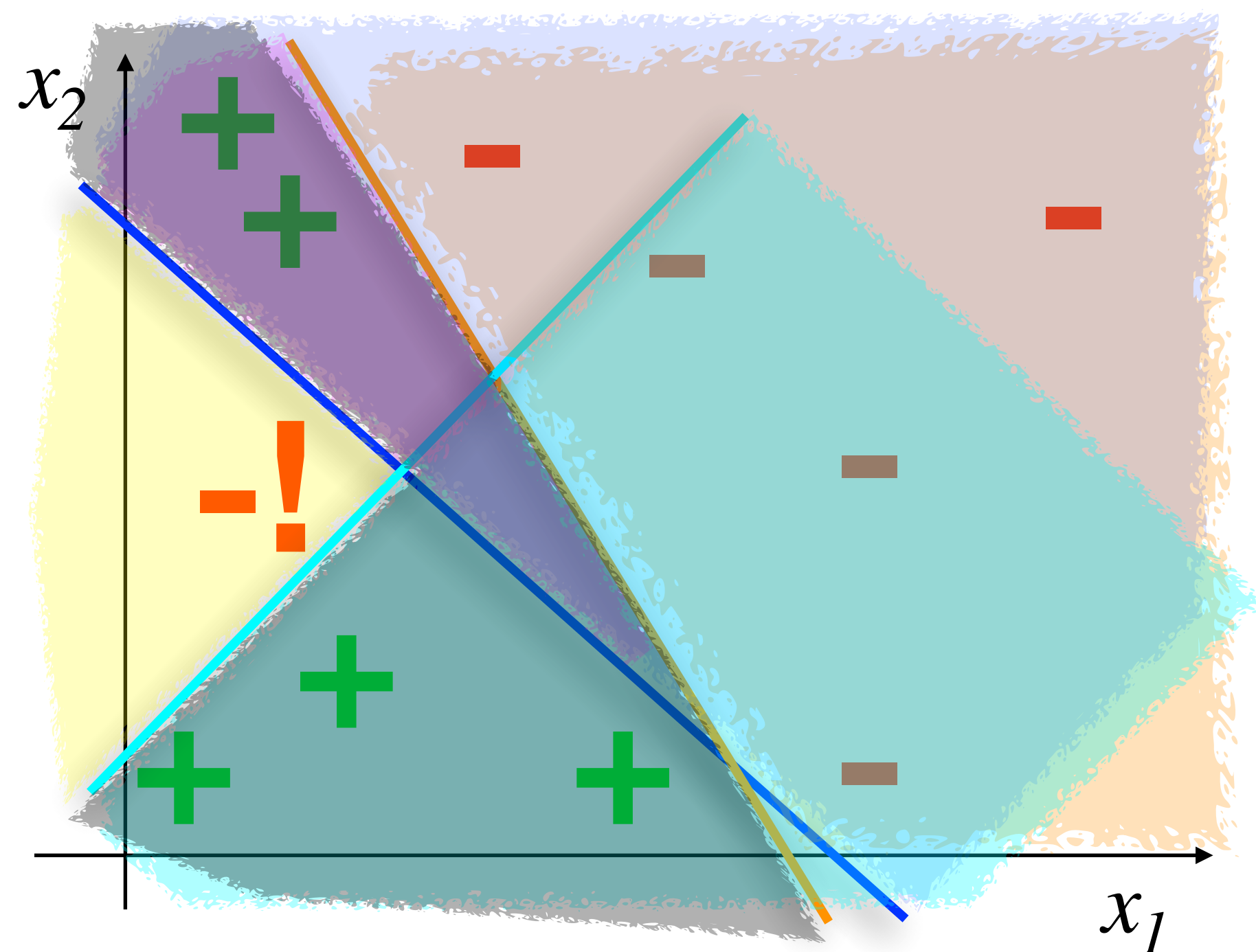
+ $d(t) = 1$
 - $d(t) = 0$

Přidáním další vrstvy neuronů zajistíme rozpoznání **konvexních oblastí**.



Vícevrstvé neuronové sítě

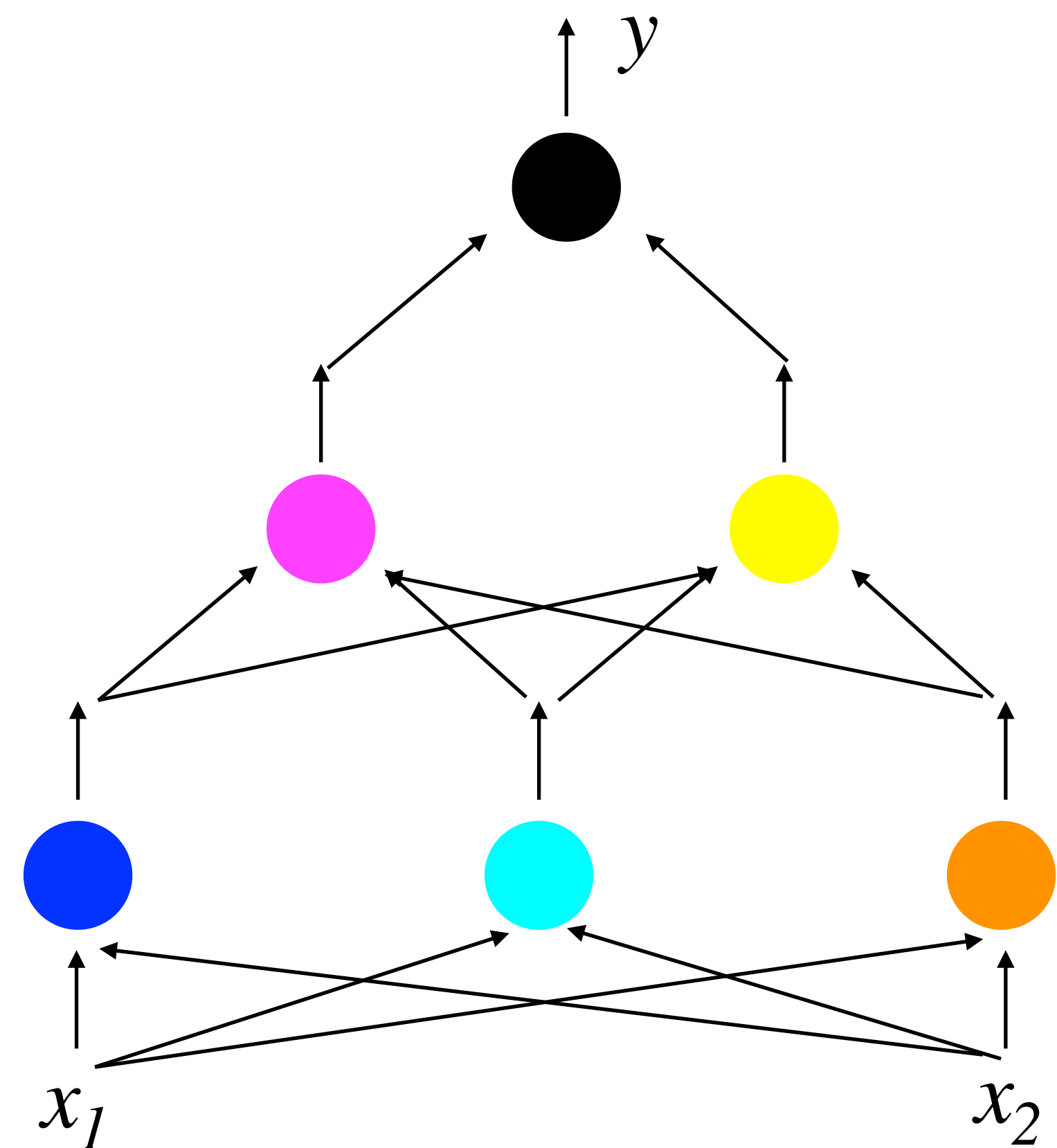
Svět není lineární 🙄



$$[x_1(t), x_2(t)] \Rightarrow d(t)$$

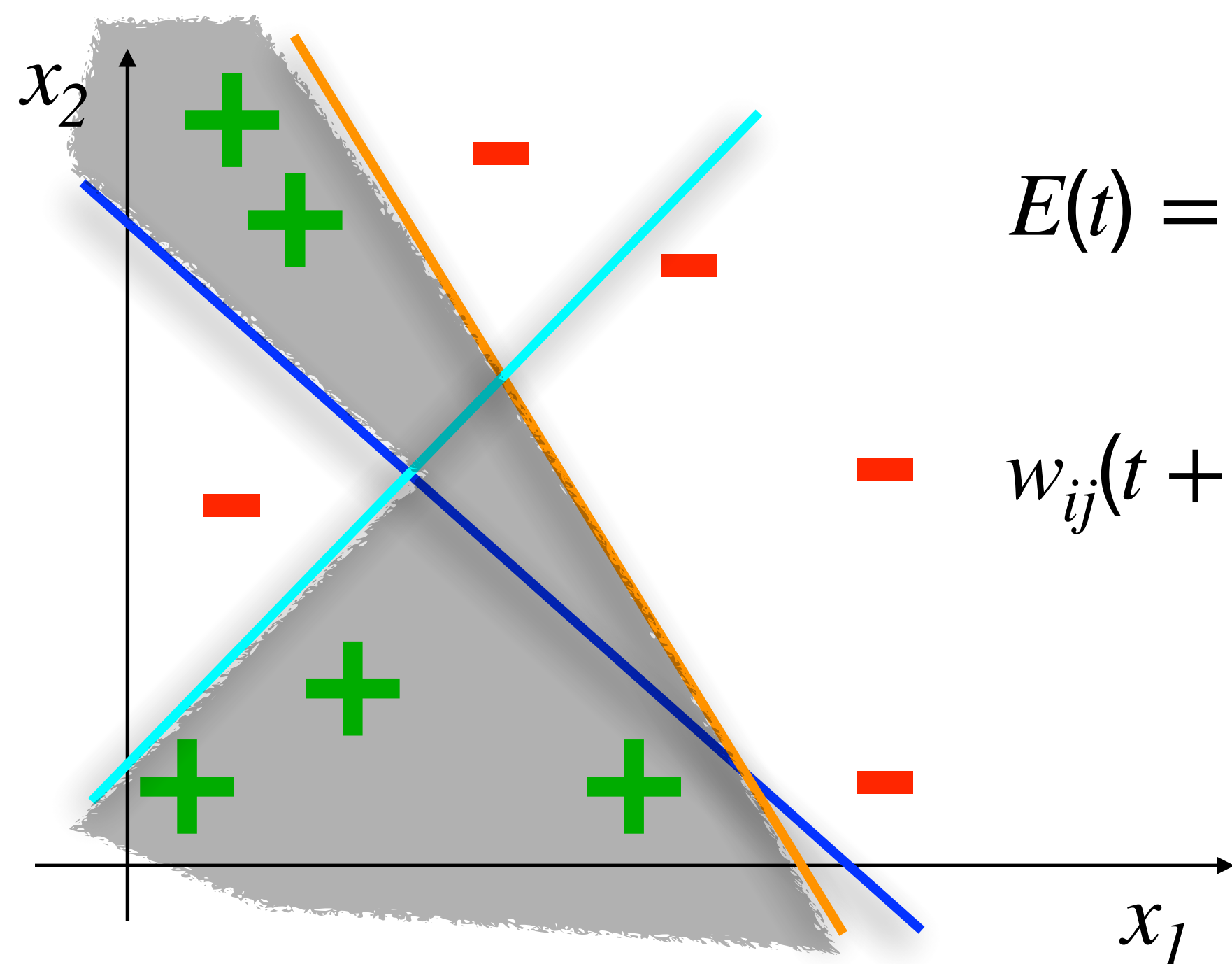
+ $d(t) = 1$
 - $d(t) = 0$

Další vrstva rozpoznává **obecně definované oblasti**.



Učení vícevrstvé neuronové sítě

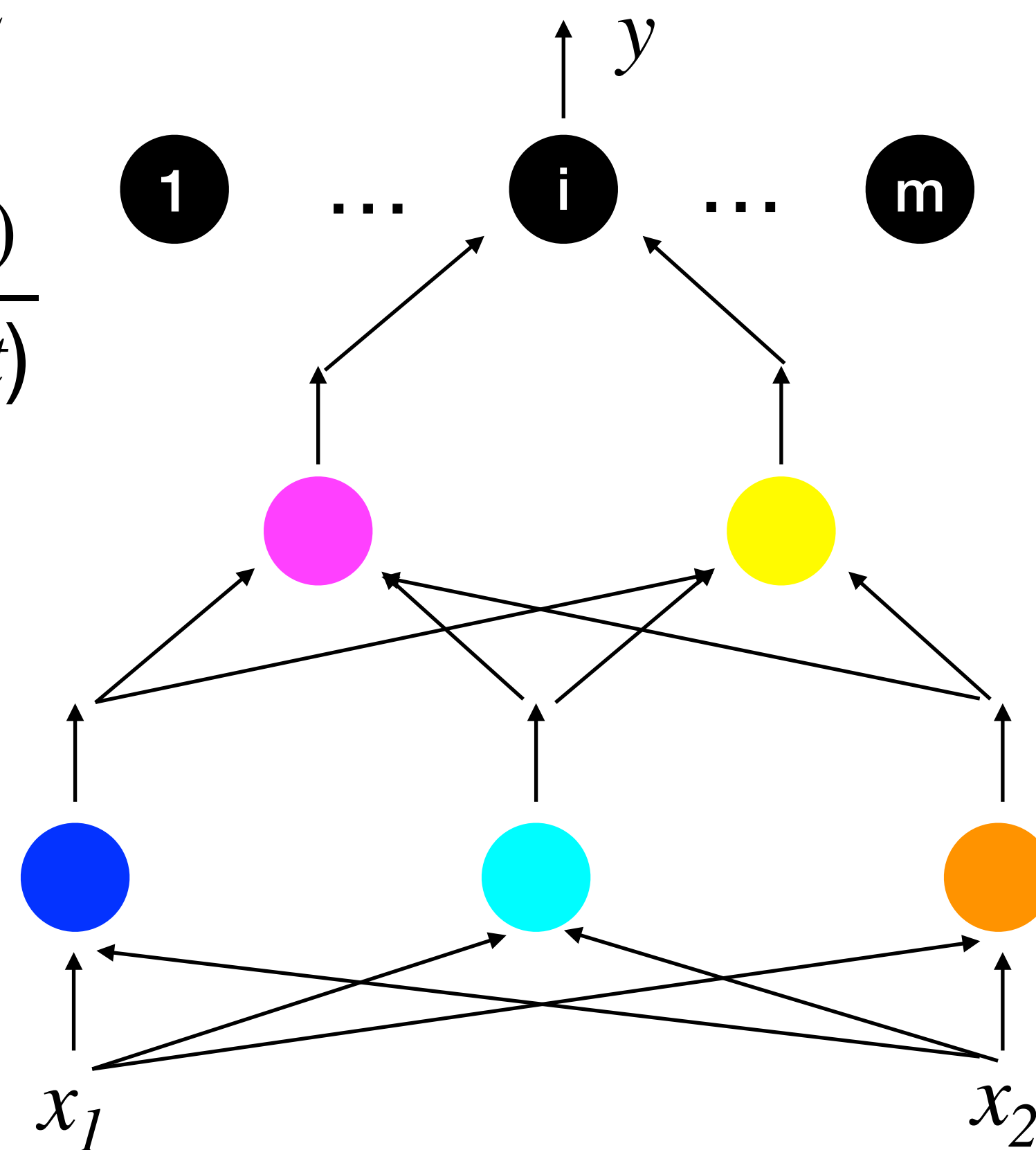
Backpropagation - zpětné šíření chyby a adaptace vah



$$[x_1(t), x_2(t)] \Rightarrow d(t)$$

$$\begin{aligned} + & d(t) = 1 \\ - & d(t) = 0 \end{aligned}$$

$$E(t) = \frac{1}{2} \sum_{i=1}^m (d_i(t) - y_i(t))^2$$
$$w_{ij}(t+1) = w_{ij}(t) - \eta \cdot \frac{\partial E(t)}{\partial w_{ij}(t)}$$



Příklad: Backpropagation

<https://dev.ivondrak.cz/notebooks/MultilayeredNeuralNets/backpropagation.ipynb>

www.ivondrak.cz

Děkuji za pozornost