

PV021: Neuronové sítě

Tomáš Brázdil

Organizační poznámky

Zdroje:

- ▶ **hlavní zdroj:** přednáška
- ▶ Jiří Šíma a Jan Neruda, Teoretické otázky neuronových sítí
<http://www2.cs.cas.cz/~sima/kniha.pdf>
(nás se týká především Část I.)
- ▶ (přednáška je založena na částech dalších knih, např. Christopher M. Bishop, Pattern Recognition and Machine Learning)

Organizační poznámky

Zdroje:

- ▶ **hlavní zdroj:** přednáška
- ▶ Jiří Šíma a Jan Neruda, Teoretické otázky neuronových sítí
<http://www2.cs.cas.cz/~sima/kniha.pdf>
(nás se týká především Část I.)
- ▶ (přednáška je založena na částech dalších knih, např. Christopher M. Bishop, Pattern Recognition and Machine Learning)

Závěrečné hodnocení:

- ▶ projekt
 - ▶ dvoučlenné týmy; implementace vybraného modelu + analýza vlastních dat
 - ▶ implementace v jazycích C++ nebo Java bez využití knihoven pro strojové učení a data mining
 - ▶ data pokud možno reálná, která je nutno připravit pro analýzu
- ▶ ústní zkouška (cokoliv z přednášky *včetně důkazů, které budu psát na tabuli*)

Strojové učení obecně

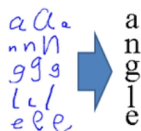
- ▶ Strojové učení = konstrukce systémů, které se mohou učit z dat (a tudíž je není třeba explicitně programovat)

Strojové učení obecně

- ▶ Strojové učení = konstrukce systémů, které se mohou učit z dat (a tudíž je není třeba explicitně programovat)
 - ▶ spam filter
 - ▶ učí se na databázi “označkových” emailů
 - ▶ poté samostatně rozpozná nový spam

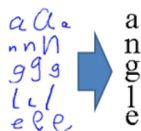
Strojové učení obecně

- ▶ Strojové učení = konstrukce systémů, které se mohou učit z dat (a tudíž je není třeba explicitně programovat)
 - ▶ spam filter
 - ▶ učí se na databázi “označkových” emailů
 - ▶ poté samostatně rozpozná nový spam
 - ▶ čtečka rukou psaných písmen
 - ▶ učí se na databázi rukou psaných písmen s označením správné odpovědi
 - ▶ poté samostatně rozpoznává písmena



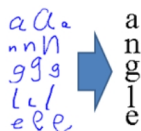
Strojové učení obecně

- ▶ Strojové učení = konstrukce systémů, které se mohou učit z dat (a tudíž je není třeba explicitně programovat)
 - ▶ spam filter
 - ▶ učí se na databázi “označkových” emailů
 - ▶ poté samostatně rozpozná nový spam
 - ▶ čtečka rukou psaných písmen
 - ▶ učí se na databázi rukou psaných písmen s označením správné odpovědi
 - ▶ poté samostatně rozpoznává písmena
 - ▶ ...
 - ▶ a další obrovské množství aplikací ...



Strojové učení obecně

- ▶ Strojové učení = konstrukce systémů, které se mohou učit z dat (a tudíž je není třeba explicitně programovat)
 - ▶ spam filter
 - ▶ učí se na databázi “označkových” emailů
 - ▶ poté samostatně rozpozná nový spam
 - ▶ čtečka rukou psaných písmen
 - ▶ učí se na databázi rukou psaných písmen s označením správné odpovědi
 - ▶ poté samostatně rozpoznává písmena
 - ▶ ...
 - ▶ a další obrovské množství aplikací ...
- ▶ Základní požadavky na daný učící algoritmus:
 - ▶ **reprezentace**: schopnost zachytit strukturu učících dat
 - ▶ **generalizace**: schopnost dobře fungovat na nových datech



Učení (obvykle) spočívá v konstrukci matematického modelu dat a jeho následné aplikaci na nová data.

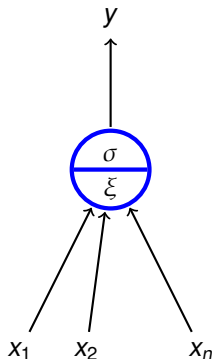
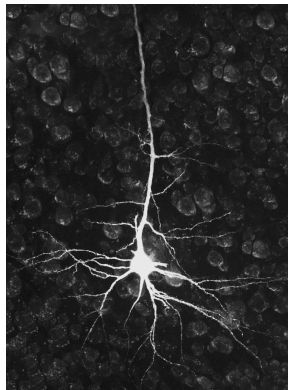
Existuje mnoho druhů modelů:

- ▶ rozhodovací stromy
- ▶ induktivní logické programování
- ▶ Bayesovské sítě (a tzv. grafické modely)
- ▶ **neuronové sítě**
- ▶ support vector machines
- ▶ skryté Markovovy modely
- ▶ ...

Neuronové sítě, vycházející z modelů lidského mozku, jsou přirozeným základem pro učící algoritmy!

Co jsou umělé neuronové sítě

- ▶ **Umělý neuron** je *hrubou matematickou aproximací* biologického neuronu.
- ▶ **(Umělá) neuronová síť** se skládá ze vzájemně propojených (umělých) neuronů. "Schopnosti" sítě jsou zakódovány v síle spojů mezi neurony.



Modelování biologických neuronových sítí (computational neuroscience).

Modelování biologických neuronových sítí (computational neuroscience).

- ▶ zjednodušený matematický model pomáhá identifikovat důležité mechanismy

Modelování biologických neuronových sítí (computational neuroscience).

- ▶ zjednodušený matematický model pomáhá identifikovat důležité mechanismy
 - ▶ Jak mozek přijímá a zpracovává informace?

Modelování biologických neuronových sítí (computational neuroscience).

- ▶ zjednodušený matematický model pomáhá identifikovat důležité mechanismy
 - ▶ Jak mozek přijímá a zpracovává informace?
 - ▶ Jak uchovává informace?

Modelování biologických neuronových sítí (computational neuroscience).

- ▶ zjednodušený matematický model pomáhá identifikovat důležité mechanismy
 - ▶ Jak mozek přijímá a zpracovává informace?
 - ▶ Jak uchovává informace?
 - ▶ Jak se mozek vyvíjí?
 - ▶ ...

Modelování biologických neuronových sítí (computational neuroscience).

- ▶ zjednodušený matematický model pomáhá identifikovat důležité mechanismy
 - ▶ Jak mozek přijímá a zpracovává informace?
 - ▶ Jak uchovává informace?
 - ▶ Jak se mozek vyvíjí?
 - ▶ ...
- ▶ neurovědy jsou silně multidisciplinární - precizní (matematický) popis usnadňuje komunikaci mezi odborníky na jednotlivé podoblasti

Této oblasti se nebudu příliš věnovat!

Proč umělé neuronové sítě ...

Neuronové sítě ve strojovém učení.

- ▶ Většinou mnohem “primitivnější” modely, které se postupně odchyľují od biologické motivace.

Neuronové sítě ve strojovém učení.

- ▶ Většinou mnohem “primitivnější” modely, které se postupně odchyľují od biologické motivace.
- ▶ Silně zaměřené na aplikace v různých oblastech, např.
 - ▶ řízení a rozhodování - autonomní řízení vozidel, řízení výrobních procesů, zpráva přírodních zdrojů
 - ▶ hry - vrhcáby, poker
 - ▶ finančnictví - vývoj cen akcií a měnových kurzů, analýza rizik, dělení klientů do specifických skupin
 - ▶ medicína - diagnostika, zpracování signálu (EKG, EEG, ...), analýza obrazu (MRI, rentgen, ...)

Neuronové sítě ve strojovém učení.

- ▶ Většinou mnohem “primitivnější” modely, které se postupně odchyľují od biologické motivace.
- ▶ Silně zaměřené na aplikace v různých oblastech, např.
 - ▶ řízení a rozhodování - autonomní řízení vozidel, řízení výrobních procesů, zpráva přírodních zdrojů
 - ▶ hry - vrhcáby, poker
 - ▶ finančnictví - vývoj cen akcií a měnových kurzů, analýza rizik, dělení klientů do specifických skupin
 - ▶ medicína - diagnostika, zpracování signálu (EKG, EEG, ...), analýza obrazu (MRI, rentgen, ...)

 - ▶ další rozpoznávání vzorů - rozpoznávání řeči, textu, obrazu, radarových a sonarových signálů
 - ▶ zpracování signálu - filtrace, odstranění šumu
 - ▶ ...

Této oblasti se budu věnovat!

Významné vlastnosti neuronových sítí

- ▶ Masivní paralelismus
 - ▶ mnoho pomalých výpočetních prvků zpracovává informace paralelně na mnoha úrovních

Významné vlastnosti neuronových sítí

- ▶ Masivní paralelismus
 - ▶ mnoho pomalých výpočetních prvků zpracovává informace paralelně na mnoha úrovních
- ▶ Adaptace a učení
 - ▶ dítě se naučí poznat králíka poté, co se mu ukáže několik králíků

Významné vlastnosti neuronových sítí

- ▶ Masivní paralelismus
 - ▶ mnoho pomalých výpočetních prvků zpracovává informace paralelně na mnoha úrovních
- ▶ Adaptace a učení
 - ▶ dítě se naučí poznat králíka poté, co se mu ukáže několik králíků
- ▶ Schopnost generalizace
 - ▶ po shlédnutí několika králíků je dítě schopno poznat další (jiné) králíky

Významné vlastnosti neuronových sítí

- ▶ Masivní paralelismus
 - ▶ mnoho pomalých výpočetních prvků zpracovává informace paralelně na mnoha úrovních
- ▶ Adaptace a učení
 - ▶ dítě se naučí poznat králíka poté, co se mu ukáže několik králíků
- ▶ Schopnost generalizace
 - ▶ po shlédnutí několika králíků je dítě schopno poznat další (jiné) králíky
- ▶ Odolnost vůči nepřesnosti vstupu
 - ▶ rozmazaná fotka králíka může být stále klasifikována správně jako obraz králíka

Významné vlastnosti neuronových sítí

- ▶ Masivní paralelismus
 - ▶ mnoho pomalých výpočetních prvků zpracovává informace paralelně na mnoha úrovních
- ▶ Adaptace a učení
 - ▶ dítě se naučí poznat králíka poté, co se mu ukáže několik králíků
- ▶ Schopnost generalizace
 - ▶ po shlédnutí několika králíků je dítě schopno poznat další (jiné) králíky
- ▶ Odolnost vůči nepřesnosti vstupu
 - ▶ rozmazaná fotka králíka může být stále klasifikována správně jako obraz králíka
- ▶ Odolnost vůči poškození
 - ▶ mnoho experimentů prokázalo, že i poškozená neuronová síť je stále schopna uspokojivě fungovat
 - ▶ poškozená síť se může přeadaptovat, stávající neurony mohou převzít funkci poškozených

Cíl předmětu

- ▶ Na co se zaměříme
 - ▶ Základní techniky a principy neuronových sítí (NS)

- ▶ Na co se zaměříme
 - ▶ Základní techniky a principy neuronových sítí (NS)
 - ▶ Přehled základních modelů NS a jejich použití

- ▶ Na co se zaměříme
 - ▶ Základní techniky a principy neuronových sítí (NS)
 - ▶ Přehled základních modelů NS a jejich použití
- ▶ Co si (doufám) odnesete
 - ▶ Znalost základních modelů NS
(perceptron, vícevrstvá síť, hluboké sítě, Hopfieldova síť, Boltzmannův stroj, Kohonenova mapa)

- ▶ Na co se zaměříme
 - ▶ Základní techniky a principy neuronových sítí (NS)
 - ▶ Přehled základních modelů NS a jejich použití
- ▶ Co si (doufám) odnesete
 - ▶ Znalost základních modelů NS
(perceptron, vícevrstvá síť, hluboké sítě, Hopfieldova síť, Boltzmannův stroj, Kohonenova mapa)
 - ▶ Znalost jednoduchých aplikací těchto modelů

- ▶ Na co se zaměříme
 - ▶ Základní techniky a principy neuronových sítí (NS)
 - ▶ Přehled základních modelů NS a jejich použití
- ▶ Co si (doufám) odnesete
 - ▶ Znalost základních modelů NS
(perceptron, vícevrstvá síť, hluboké sítě, Hopfieldova síť, Boltzmannův stroj, Kohonenova mapa)
 - ▶ Znalost jednoduchých aplikací těchto modelů
 - ▶ Znalost základních principů učení NS

- ▶ Na co se zaměříme
 - ▶ Základní techniky a principy neuronových sítí (NS)
 - ▶ Přehled základních modelů NS a jejich použití
- ▶ Co si (doufám) odnesete
 - ▶ Znalost základních modelů NS
(perceptron, vícevrstvá síť, hluboké sítě, Hopfieldova síť, Boltzmannův stroj, Kohonenova mapa)
 - ▶ Znalost jednoduchých aplikací těchto modelů
 - ▶ Znalost základních principů učení NS
 - ▶ Přehled elementárních "implementačních" technik pro učení a aplikaci NS

- ▶ Na co se zaměříme
 - ▶ Základní techniky a principy neuronových sítí (NS)
 - ▶ Přehled základních modelů NS a jejich použití
- ▶ Co si (doufám) odnesete
 - ▶ Znalost základních modelů NS
(perceptron, vícevrstvá síť, hluboké sítě, Hopfieldova síť, Boltzmannův stroj, Kohonenova mapa)
 - ▶ Znalost jednoduchých aplikací těchto modelů
 - ▶ Znalost základních principů učení NS
 - ▶ Přehled elementárních "implementačních" technik pro učení a aplikaci NS
 - ▶ přehled některých souvisejících modelů a technik
(redukce dimenze dat, PCA)

Základy neuronových sítí

- ▶ Biologický neuron a neuronová síť
- ▶ Formální neuron
- ▶ Neuronové sítě a jejich dynamika

Biologická neuronová síť

- ▶ Nervová soustava člověka se skládá z přibližně 10^{11} neuronů (centimetr krychlový lidského mozku obsahuje až 50 milionů nervových buněk)
- ▶ Každý neuron je spojen s přibližně 10^4 neurony
- ▶ Neurony jsou velmi komplexní systémy

Biologická neuronová síť

- ▶ Nervová soustava člověka se skládá z přibližně 10^{11} neuronů (centimetr krychlový lidského mozku obsahuje až 50 milionů nervových buněk)
- ▶ Každý neuron je spojen s přibližně 10^4 neurony
- ▶ Neurony jsou velmi komplexní systémy

Velmi hrubý popis funkce nervové soustavy:

- ▶ Vnější podněty jsou přijímány *receptory* (např. buňky oka).

Biologická neuronová síť

- ▶ Nervová soustava člověka se skládá z přibližně 10^{11} neuronů (centimetr krychlový lidského mozku obsahuje až 50 milionů nervových buněk)
- ▶ Každý neuron je spojen s přibližně 10^4 neurony
- ▶ Neurony jsou velmi komplexní systémy

Velmi hrubý popis funkce nervové soustavy:

- ▶ Vnější podněty jsou přijímány *receptory* (např. buňky oka).
- ▶ Informace jsou dále přenášeny pomocí periferní nervové soustavy (PNS) do centrální nervové soustavy (CNS - mícha, mozek) kde jsou zpracovávány (integrovány)

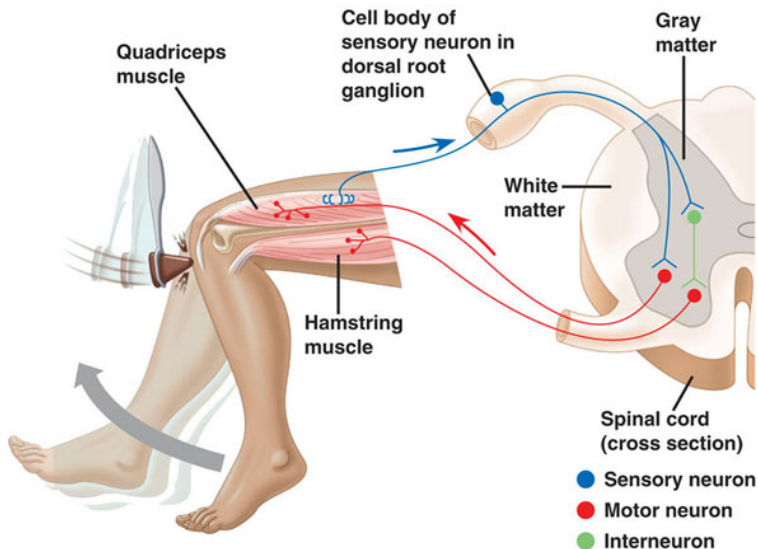
Biologická neuronová síť

- ▶ Nervová soustava člověka se skládá z přibližně 10^{11} neuronů (centimetr krychlový lidského mozku obsahuje až 50 milionů nervových buněk)
- ▶ Každý neuron je spojen s přibližně 10^4 neurony
- ▶ Neurony jsou velmi komplexní systémy

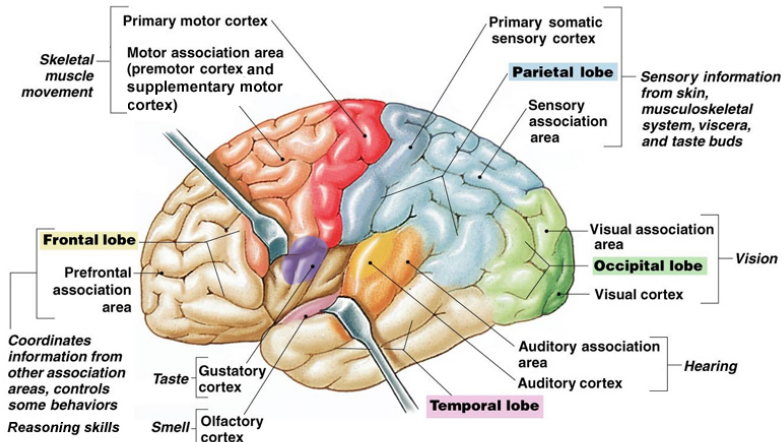
Velmi hrubý popis funkce nervové soustavy:

- ▶ Vnější podněty jsou přijímány *receptory* (např. buňky oka).
- ▶ Informace jsou dále přenášeny pomocí periferní nervové soustavy (PNS) do centrální nervové soustavy (CNS - mícha, mozek) kde jsou zpracovávány (integrovány)
- ▶ Po zpracování informace jsou (pomocí PNS) případně aktivovány *efektory* (např. svalové buňky)

Biologická neuronová síť - příklad



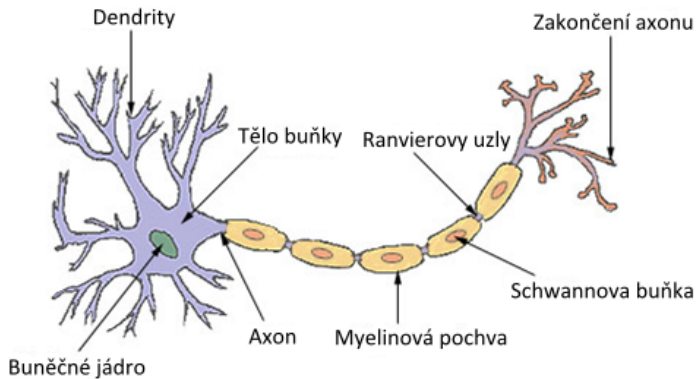
Mozková kůra



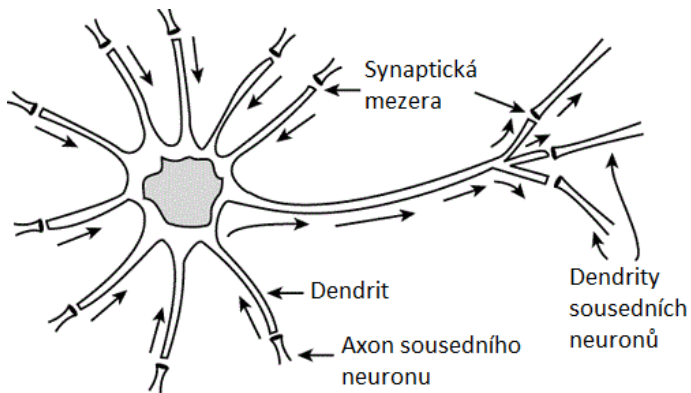
Copyright © 2007 Pearson Education, Inc., publishing as Benjamin Cummings.

Fig. 9-15

Biologický neuron

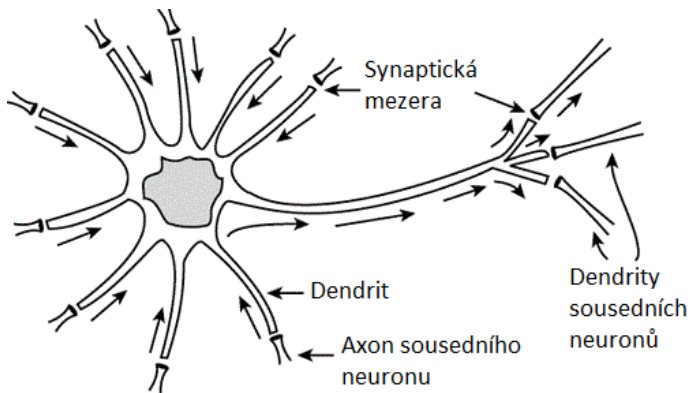


Sumace a akční potenciál



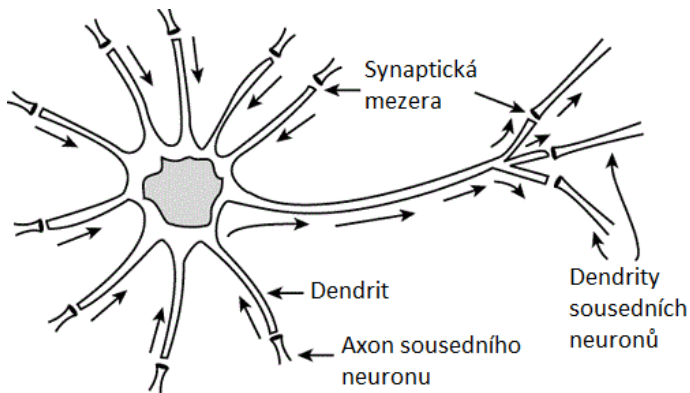
- ▶ Klidový potenciál v těle neuronu ≈ -70 mV

Sumace a akční potenciál



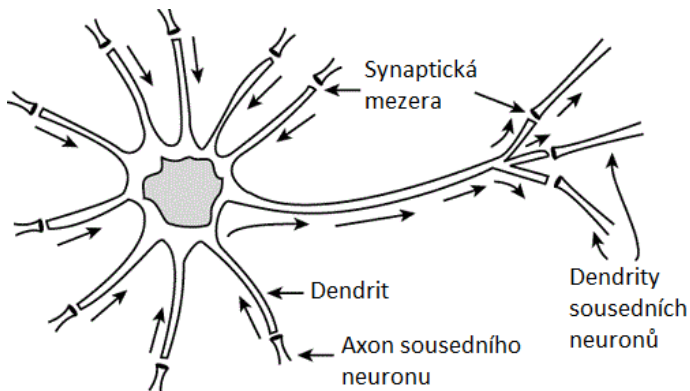
- ▶ Klidový potenciál v těle neuronu ≈ -70 mV
- ▶ Vnější podněty mění potenciál v axonovém hrbolku

Sumace a akční potenciál



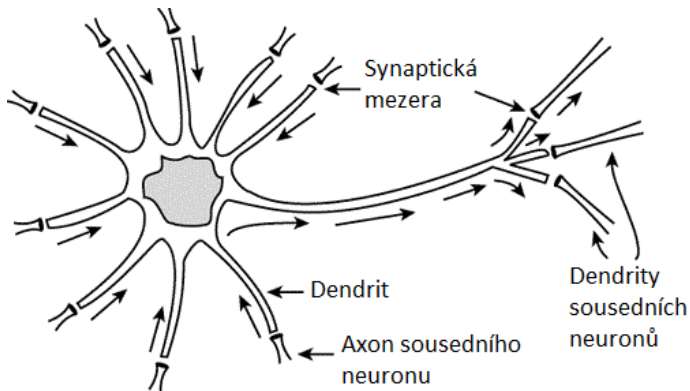
- ▶ Klidový potenciál v těle neuronu ≈ -70 mV
- ▶ Vnější podněty mění potenciál v axonovém hrbolku
- ▶ Po překročení prahu ≈ -50 mV je generován *akční potenciál* ≈ 40 mV

Sumace a akční potenciál



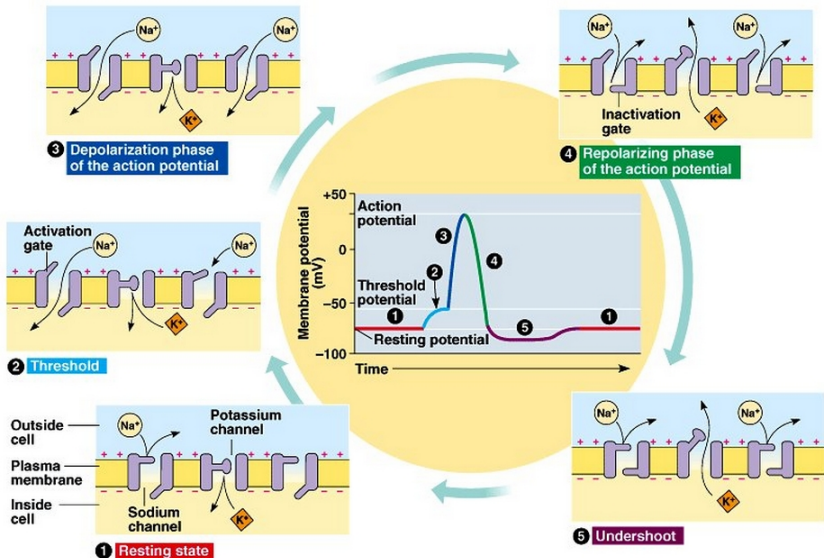
- ▶ Klidový potenciál v těle neuronu ≈ -70 mV
- ▶ Vnější podněty mění potenciál v axonovém hrbolku
- ▶ Po překročení prahu ≈ -50 mV je generován *akční potenciál* ≈ 40 mV
- ▶ Poté nastane krátká *refrakce* ≈ -80 mV

Sumace a akční potenciál



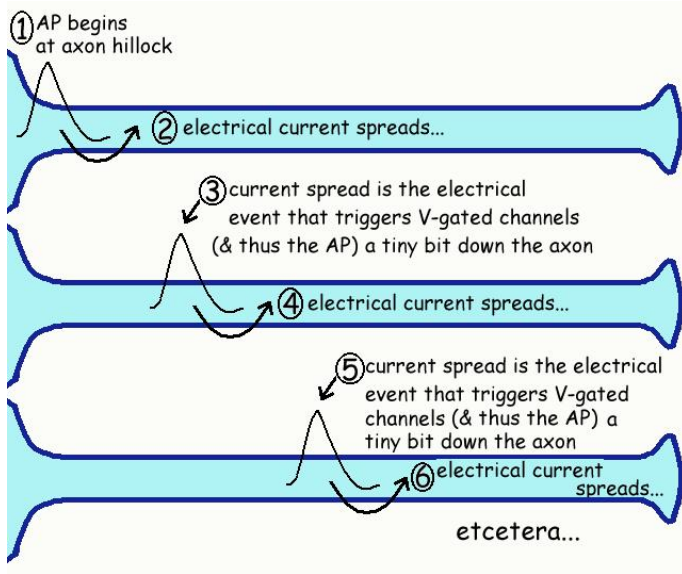
- ▶ Klidový potenciál v těle neuronu ≈ -70 mV
- ▶ Vnější podněty mění potenciál v axonovém hrbolku
- ▶ Po překročení prahu ≈ -50 mV je generován *akční potenciál* ≈ 40 mV
- ▶ Poté nastane krátká *refrakce* ≈ -80 mV
- ▶ akční potenciál se šíří axonem, v axonovém zakončení vyvolá chemický proces, který změní potenciál v sousedním neuronu

Akční potenciál - podrobněji

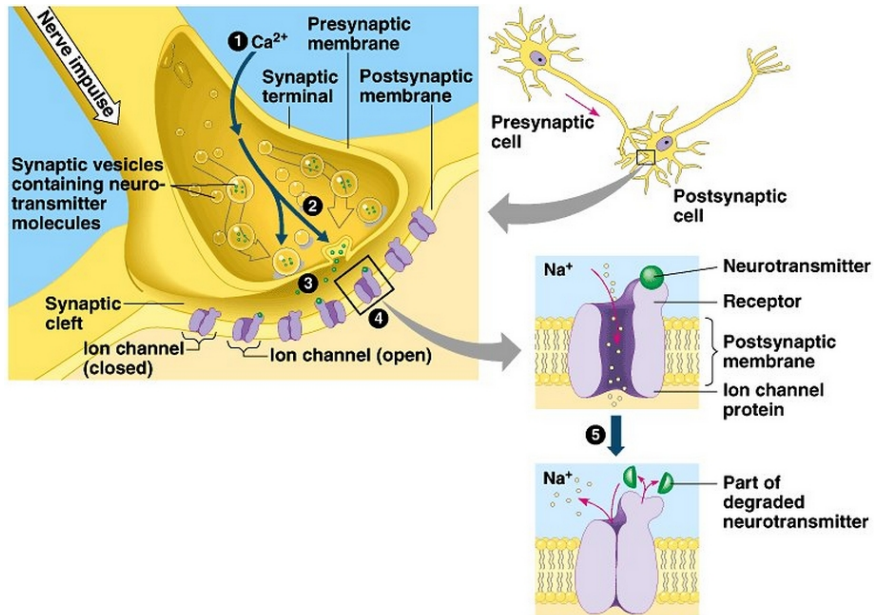


Copyright © Pearson Education, Inc., publishing as Benjamin Cummings.

Šíření akčního potenciálu axonem



Synaptický přenos



Sumace

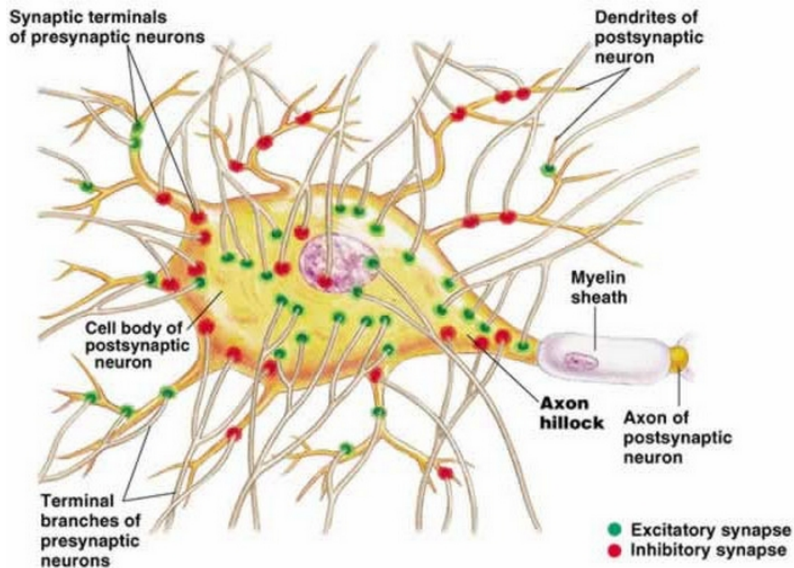
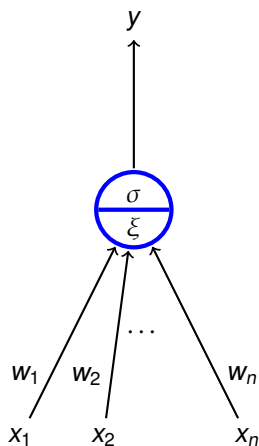


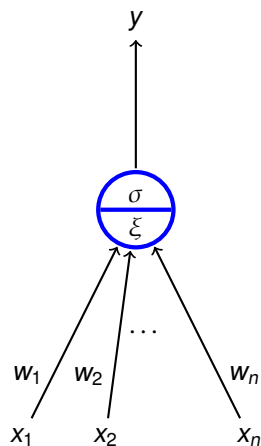
Figure 48.11(a), page 972, Campbell's *Biology*, 5th Edition

Formální neuron bez biasu

- ▶ x_1, \dots, x_n jsou reálné **vstupy** („dendrity“)

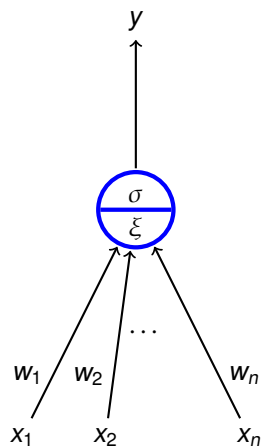


Formální neuron bez biasu



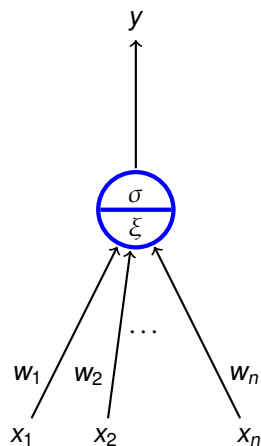
- ▶ x_1, \dots, x_n jsou reálné **vstupy** („dendrity“)
- ▶ w_1, \dots, w_n jsou reálné **váhy** („propustnost synapsí“)

Formální neuron bez biasu



- ▶ x_1, \dots, x_n jsou reálné **vstupy** („dendrity“)
- ▶ w_1, \dots, w_n jsou reálné **váhy** („propustnost synapsí“)
- ▶ ξ je **vnitřní potenciál**;
většinou $\xi = \sum_{i=1}^n w_i x_i$

Formální neuron bez biasu



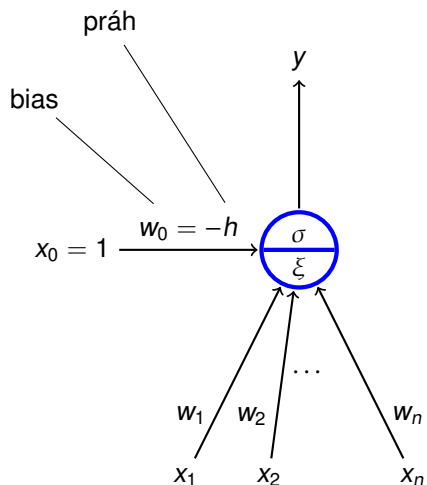
- ▶ x_1, \dots, x_n jsou reálné **vstupy** („dendrity“)
- ▶ w_1, \dots, w_n jsou reálné **váhy** („propustnost synapsí“)
- ▶ ξ je **vnitřní potenciál**; většinou $\xi = \sum_{i=1}^n w_i x_i$
- ▶ y je **výstup** daný $y = \sigma(\xi)$ kde σ je **aktivační funkce**;
např. *ostrá nelinearita*

$$\sigma(\xi) = \begin{cases} 1 & \xi \geq h; \\ 0 & \xi < h. \end{cases}$$

kde h je reálný *práh*.

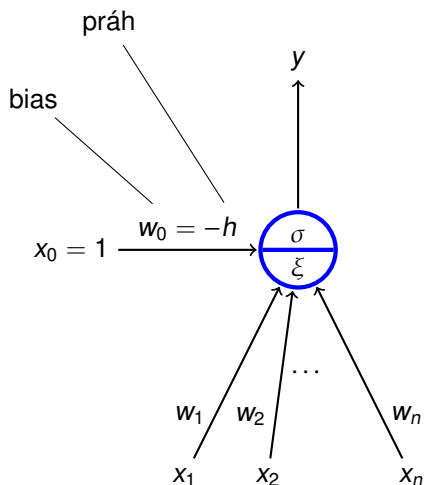
Formální neuron (s biasem)

- ▶ x_1, \dots, x_n jsou reálné **vstupy**

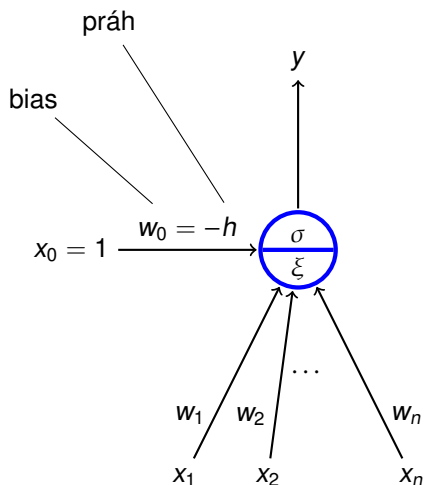


Formální neuron (s biasem)

- ▶ x_1, \dots, x_n jsou reálné **vstupy**
- ▶ x_0 je speciální vstup, který má vždy hodnotu 1

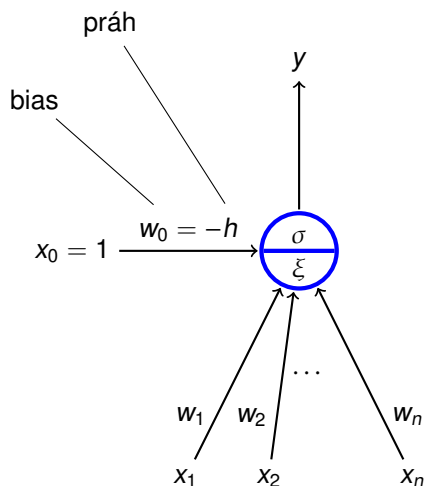


Formální neuron (s biasem)



- ▶ x_1, \dots, x_n jsou reálné **vstupy**
- ▶ x_0 je speciální vstup, který má vždy hodnotu 1
- ▶ w_0, w_1, \dots, w_n jsou reálné **váhy**

Formální neuron (s biasem)

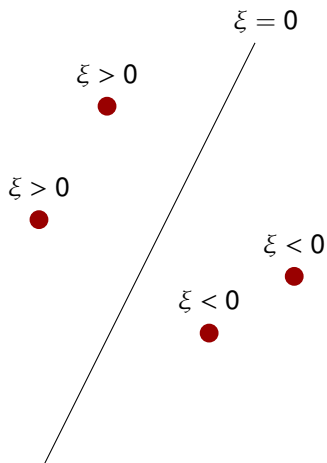


- ▶ x_1, \dots, x_n jsou reálné **vstupy**
- ▶ x_0 je speciální vstup, který má vždy hodnotu 1
- ▶ w_0, w_1, \dots, w_n jsou reálné **váhy**
- ▶ ξ je **vnitřní potenciál**;
většinou $\xi = w_0 + \sum_{i=1}^n w_i x_i$
- ▶ y je **výstup** daný $y = \sigma(\xi)$
kde σ je **aktivační funkce**; např.
ostrá nelinearita

$$\sigma(\xi) = \begin{cases} 1 & \xi \geq 0; \\ 0 & \xi < 0. \end{cases}$$

(práh aktivační funkce σ je roven 0;
reálný práh byl nahrazen vstupem
 $x_0 = 1$ a váhou $w_0 = -h$)

Neuron a lineární separace



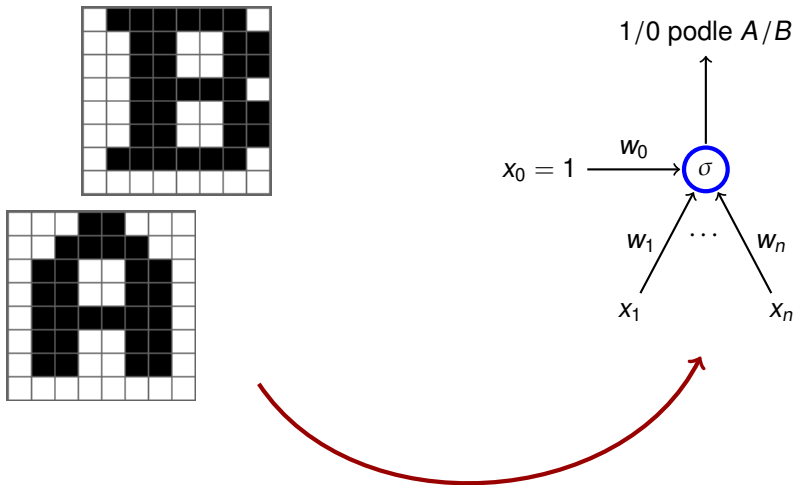
- ▶ vnitřní potenciál

$$\xi = w_0 + \sum_{i=1}^n w_i x_i$$

zadává separační nadrovinu v n -rozměrném **vstupním prostoru**

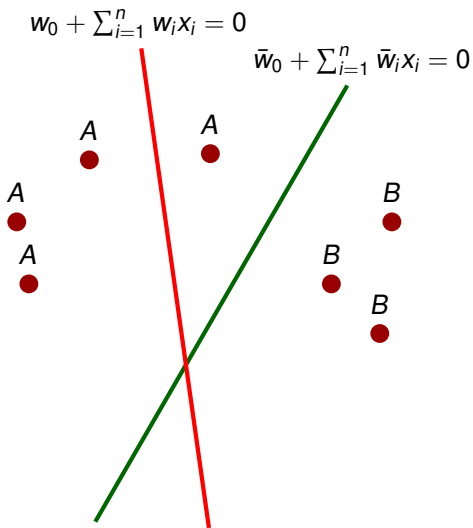
- ▶ v 2d přímka
- ▶ v 3d plocha
- ▶ ...

Neuron a lineární separace



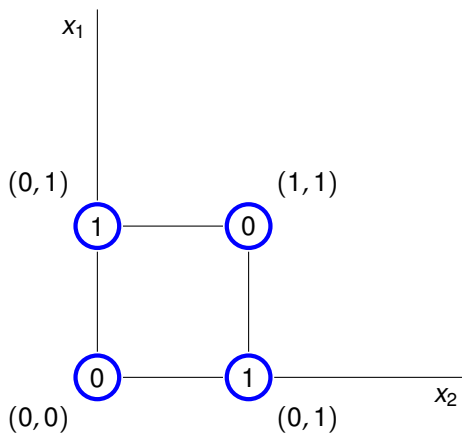
Zde $n = 8 \cdot 8$ tedy počet pixelů v obrázcích. Vstupy jsou binární vektory dimenze n (tmavý bod ≈ 1 , světlý bod ≈ 0).

Neuron a lineární separace



- ▶ Červená přímka nesprávně klasifikuje
- ▶ Zelená klasifikuje správně (může být výsledkem korekce učícího algoritmu)

Neuron a lineární separace



- ▶ Neexistuje přímka, která by oddělila body 1 od bodů 0.

- ▶ **Neuronová síť** se skládá z formálních neuronů, které jsou vzájemně propojeny tak, že výstup jednoho neuronu je vstupem obecně více neuronů.

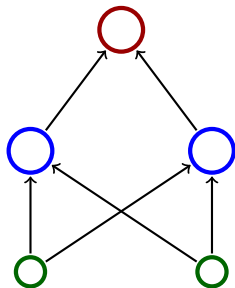
- ▶ **Neuronová síť** se skládá z formálních neuronů, které jsou vzájemně propojeny tak, že výstup jednoho neuronu je vstupem obecně více neuronů.
- ▶ **Architektura sítě** je určena počtem a vzájemným propojením neuronů.

- ▶ **Neuronová síť** se skládá z formálních neuronů, které jsou vzájemně propojeny tak, že výstup jednoho neuronu je vstupem obecně více neuronů.
- ▶ **Architektura sítě** je určena počtem a vzájemným propojením neuronů.
- ▶ **Stav** sítě je vektor hodnot všech neuronů.
(Stavy sítě s n neurony jsou prvky \mathbb{R}^n)
- ▶ **Stavový prostor** sítě je množina všech stavů.

- ▶ **Neuronová síť** se skládá z formálních neuronů, které jsou vzájemně propojeny tak, že výstup jednoho neuronu je vstupem obecně více neuronů.
- ▶ **Architektura sítě** je určena počtem a vzájemným propojením neuronů.
- ▶ **Stav** sítě je vektor hodnot všech neuronů.
(Stavy sítě s n neurony jsou prvky \mathbb{R}^n)
- ▶ **Stavový prostor** sítě je množina všech stavů.
- ▶ **Konfigurace** sítě je vektor hodnot všech vah v síti.
(Konfigurace sítě s m spoji jsou prvky \mathbb{R}^m)
- ▶ **Váhový prostor** sítě je množina všech konfigurací.

Neurony rozdělujeme na

- ▶ Výstupní
- ▶ Skryté
- ▶ Vstupní



Dynamiku sítě dělíme do tří *režimů*

Dynamiku sítě dělíme do tří *režimů*

- ▶ **Organizační**

- ▶ architektura sítě a její případná změna

Dynamiku sítě dělíme do tří režimů

- ▶ **Organizační**

- ▶ architektura sítě a její případná změna

- ▶ **Aktivní**

- ▶ počáteční stav sítě (hodnoty neuronů) a jeho změny v čase (při pevné architektuře a konfiguraci)
(mimo jiné určuje způsob výpočtu vnitřních potenciálů ξ a aktivační funkce σ všech neuronů)

Dynamiku sítě dělíme do tří režimů

▶ Organizační

- ▶ architektura sítě a její případná změna

▶ Aktivní

- ▶ počáteční stav sítě (hodnoty neuronů) a jeho změny v čase (při pevné architektuře a konfiguraci)
(mimo jiné určuje způsob výpočtu vnitřních potenciálů ξ a aktivační funkce σ všech neuronů)

▶ Adaptivní

- ▶ počáteční konfigurace sítě (hodnoty vah) a její změna v čase (učení)

Organizační dynamika

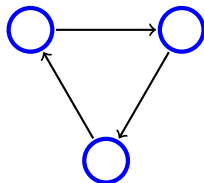
Organizační dynamika určuje strukturu sítě.

Organizační dynamika

Organizační dynamika určuje strukturu sítě.

Rozlišujeme dva typy architektury:

- ▶ Cyklická (resp. rekurentní), pokud obsahuje orientovaný cyklus.

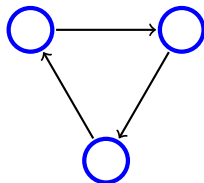


Organizační dynamika

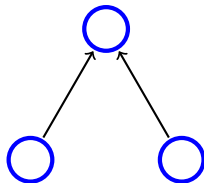
Organizační dynamika určuje strukturu sítě.

Rozlišujeme dva typy architektury:

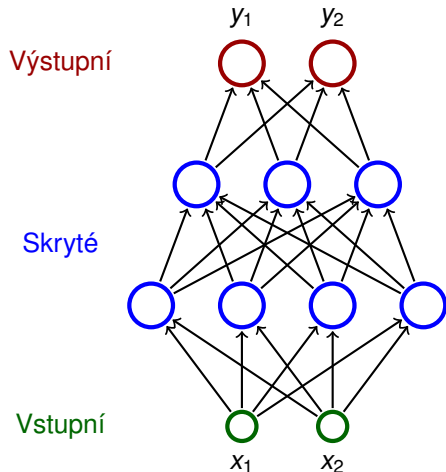
- ▶ Cyklická (resp. rekurentní), pokud obsahuje orientovaný cyklus.



- ▶ Acyklická (resp. dopředná)

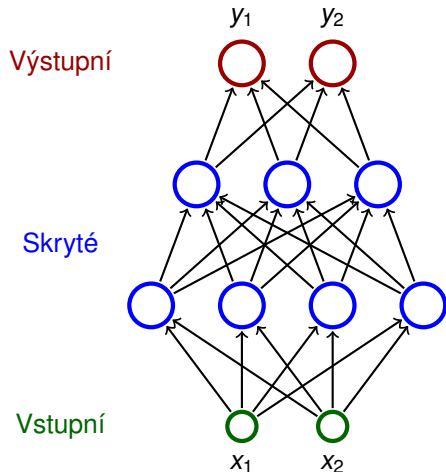


Organizační dynamika - vícevrstvé sítě



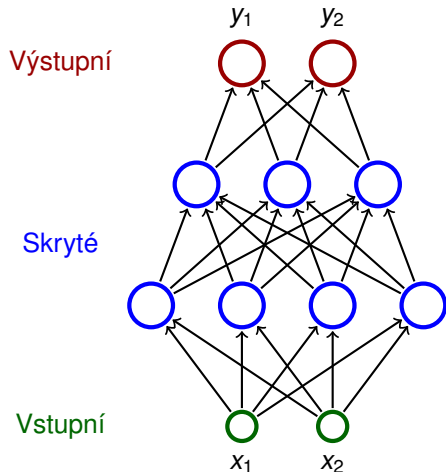
- ▶ Neurony jsou rozděleny do **vrstev** (vstupní a výstupní vrstva, obecně několik skrytých vrstev)

Organizační dynamika - vícevrstvé sítě



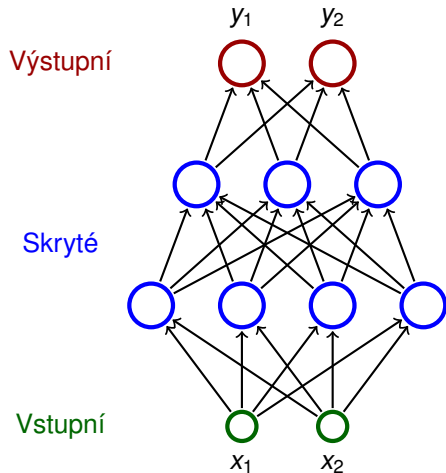
- ▶ Neurony jsou rozděleny do **vrstev** (vstupní a výstupní vrstva, obecně několik skrytých vrstev)
- ▶ Vrstvy číslujeme od 0; vstupní vrstva je nultá
 - ▶ Např. třívrstvá síť se skládá z jedné vstupní, dvou skrytých a jedné výstupní vrstvy.

Organizační dynamika - vícevrstvé sítě



- ▶ Neurony jsou rozděleny do **vrstev** (vstupní a výstupní vrstva, obecně několik skrytých vrstev)
- ▶ Vrstvy číslujeme od 0; vstupní vrstva je nultá
 - ▶ Např. třívrstvá síť se skládá z jedné vstupní, dvou skrytých a jedné výstupní vrstvy.
- ▶ Neurony v i -té vrstvě jsou spojeny se všemi neurony ve vrstvě $i + 1$.

Organizační dynamika - vícevrstvé sítě



- ▶ Neurony jsou rozděleny do **vrstev** (vstupní a výstupní vrstva, obecně několik skrytých vrstev)
- ▶ Vrstvy číslujeme od 0; vstupní vrstva je nultá
 - ▶ Např. třívrstvá síť se skládá z jedné vstupní, dvou skrytých a jedné výstupní vrstvy.
- ▶ Neurony v i -té vrstvě jsou spojeny se všemi neurony ve vrstvě $i + 1$.
- ▶ Vícevrstvou síť lze zadat počty neuronů v jednotlivých vrstvách (např. 2-4-3-2)

Aktivní dynamika určuje, jakým způsobem síť počítá.

Aktivní dynamika určuje, jakým způsobem síť počítá.

Mějme síť s n neurony z nichž k je vstupních a ℓ výstupních.

Aktivní dynamika určuje, jakým způsobem síť počítá.

Mějme síť s n neurony z nichž k je vstupních a ℓ výstupních.

- ▶ **Vstup sítě** je vektor k reálných čísel, tedy prvek \mathbb{R}^k .
(někdy se omezíme pouze na jistou podmnožinu \mathbb{R}^k)
- ▶ **Vstupní prostor** sítě je množina všech vstupů.

Aktivní dynamika určuje, jakým způsobem síť počítá.

Mějme síť s n neurony z nichž k je vstupních a ℓ výstupních.

- ▶ **Vstup sítě** je vektor k reálných čísel, tedy prvek \mathbb{R}^k .
(někdy se omezíme pouze na jistou podmnožinu \mathbb{R}^k)
- ▶ **Vstupní prostor** sítě je množina všech vstupů.
- ▶ **Počáteční stav**
Vstupní neurony jsou nastaveny na hodnoty ze vstupu sítě
(každá složka vstupu má přiřazen příslušný vstupní neuron)
Ostatní neurony jsou iniciálně nastaveny na 0.

Aktivní dynamika - výpočet

- ▶ **Výpočet** (obvykle) probíhá v diskrétních krocích.

- ▶ **Výpočet** (obvykle) probíhá v diskrétních krocích. V každém kroku se provede následující:

Aktivní dynamika - výpočet

- ▶ **Výpočet** (obvykle) probíhá v diskrétních krocích. V každém kroku se provede následující:
 1. Podle **pravidla aktivní dynamiky** je vybrán jeden neuron (sekvenční výpočet) nebo více neuronů (paralelní výpočet).
 2. Vybraný neuron změní svůj stav v závislosti na hodnotách svých vstupů.
(Hodnota neuronu, který nemá vstupy, zůstává konstantní.)

Výpočet je **konečný** pokud se od jistého kroku dál nemění stav sítě.

Aktivní dynamika - výpočet

- ▶ **Výpočet** (obvykle) probíhá v diskrétních krocích. V každém kroku se provede následující:
 1. Podle **pravidla aktivní dynamiky** je vybrán jeden neuron (sekvenční výpočet) nebo více neuronů (paralelní výpočet).
 2. Vybraný neuron změní svůj stav v závislosti na hodnotách svých vstupů.
(Hodnota neuronu, který nemá vstupy, zůstává konstantní.)

Výpočet je **konečný** pokud se od jistého kroku dál nemění stav sítě.

- ▶ **Výstup sítě** je vektor hodnot všech výstupních neuronů (tedy prvek \mathbb{R}^{ℓ}). Výstup se mění v průběhu výpočtu!

Aktivní dynamika - výpočet

- ▶ **Výpočet** (obvykle) probíhá v diskrétních krocích. V každém kroku se provede následující:
 1. Podle **pravidla aktivní dynamiky** je vybrán jeden neuron (sekvenční výpočet) nebo více neuronů (paralelní výpočet).
 2. Vybraný neuron změní svůj stav v závislosti na hodnotách svých vstupů.
(Hodnota neuronu, který nemá vstupy, zůstává konstantní.)

Výpočet je **konečný** pokud se od jistého kroku dál nemění stav sítě.

- ▶ **Výstup sítě** je vektor hodnot všech výstupních neuronů (tedy prvek \mathbb{R}^{ℓ}). Výstup se mění v průběhu výpočtu!

Pro *vícevrstvé* sítě používáme následující pravidlo aktivní dynamiky:

V *i*-tém kroku vyhodnot' právě všechny neurony v *i*-té vrstvě.

Definice

Mějme neuronovou síť s n neurony z nichž k je vstupních a ℓ výstupních. Nechť $A \subseteq \mathbb{R}^k$ a $B \subseteq \mathbb{R}^\ell$. Předpokládejme, že výpočet této sítě skončí pro každý vstup z A .

Řekneme, že tato síť **počítá funkci** $F : A \rightarrow B$ pokud pro každý vstup $\vec{v} \in A$ je $F(\vec{v})$ výstupem sítě po skončení výpočtu.

Definice

Mějme neuronovou síť s n neurony z nichž k je vstupních a ℓ výstupních. Nechť $A \subseteq \mathbb{R}^k$ a $B \subseteq \mathbb{R}^\ell$. Předpokládejme, že výpočet této sítě skončí pro každý vstup z A .

Řekneme, že tato síť **počítá funkci** $F : A \rightarrow B$ pokud pro každý vstup $\vec{v} \in A$ je $F(\vec{v})$ výstupem sítě po skončení výpočtu.

Podle toho, zda je funkce sítě diskrétní nebo spojitá rozlišujeme **diskrétní** a **analogové** neuronové sítě.

Definice

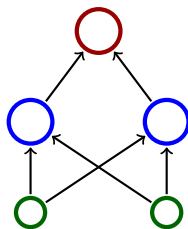
Mějme neuronovou síť s n neurony z nichž k je vstupních a ℓ výstupních. Nechť $A \subseteq \mathbb{R}^k$ a $B \subseteq \mathbb{R}^\ell$. Předpokládejme, že výpočet této sítě skončí pro každý vstup z A .

Řekneme, že tato síť **počítá funkci** $F : A \rightarrow B$ pokud pro každý vstup $\vec{v} \in A$ je $F(\vec{v})$ výstupem sítě po skončení výpočtu.

Podle toho, zda je funkce sítě diskrétní nebo spojitá rozlišujeme **diskrétní** a **analogové** neuronové sítě.

Příklad 1

Tato síť počítá funkci z \mathbb{R}^2 do \mathbb{R} .



Aktivní dynamika - aktivační funkce

Aktivní dynamika určuje aktivační funkci σ pro každý neuron.

- ▶ Ostrá nelinearita

$$\sigma(\xi) = \begin{cases} 1 & \xi \geq 0; \\ 0 & \xi < 0. \end{cases}$$

Aktivní dynamika - aktivační funkce

Aktivní dynamika určuje aktivační funkci σ pro každý neuron.

- ▶ Ostrá nelinearita

$$\sigma(\xi) = \begin{cases} 1 & \xi \geq 0; \\ 0 & \xi < 0. \end{cases}$$

- ▶ Logistická sigmoida

$$\sigma(\xi) = \frac{1}{1 + e^{-\lambda \cdot \xi}} \quad \text{kde } \lambda \in \mathbb{R} \text{ je parametr } \textit{strmosti}.$$

- ▶ Hyperbolický tangens

$$\sigma(\xi) = \frac{1 - e^{-\xi}}{1 + e^{-\xi}}$$

Aktivní dynamika - vnitřní potenciál

Aktivní dynamika určuje způsob výpočtu vnitřního potenciálu ξ každého neuronu.

Aktivní dynamika - vnitřní potenciál

Aktivní dynamika určuje způsob výpočtu vnitřního potenciálu ξ každého neuronu.

Pokud nebude uvedeno jinak, předpokládáme, že

$$\xi = w_0 + \sum_{i=1}^n w_i \cdot x_i$$

Aktivní dynamika - vnitřní potenciál

Aktivní dynamika určuje způsob výpočtu vnitřního potenciálu ξ každého neuronu.

Pokud nebude uvedeno jinak, předpokládáme, že

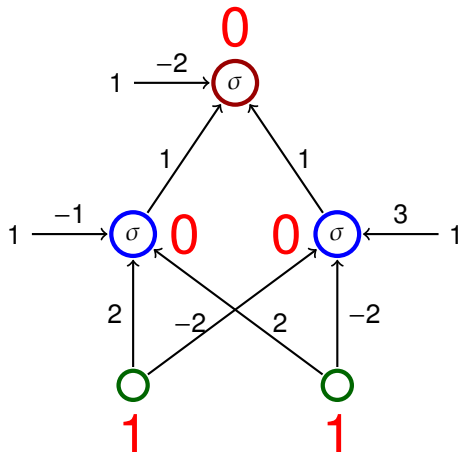
$$\xi = w_0 + \sum_{i=1}^n w_i \cdot x_i$$

Později využijeme další možnosti, např.

$$\xi = \|\vec{x} - \vec{w}\|$$

kde $\|\cdot\|$ je daná vektorová norma (nejčastěji Euklidovská), $\vec{x} = (x_1, \dots, x_n)$ jsou vstupy neuronu a $\vec{w} = (w_1, \dots, w_n)$ jsou váhy.

Aktivní dynamika - XOR



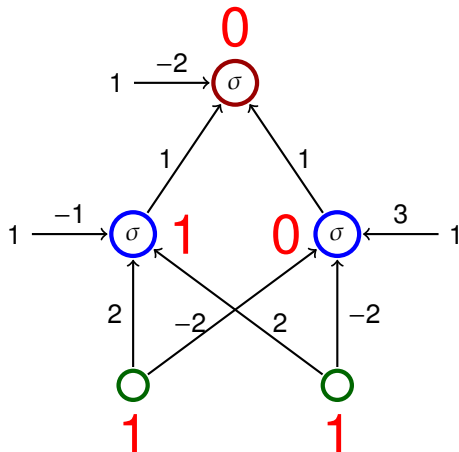
- ▶ Aktivační funkce je ostrá nelinearita

$$\sigma(\xi) = \begin{cases} 1 & \xi \geq 0; \\ 0 & \xi < 0. \end{cases}$$

- ▶ Síť počítá funkci $XOR(x_1, x_2)$

x_1	x_2	y
1	1	0
1	0	1
0	1	1
0	0	0

Aktivní dynamika - XOR



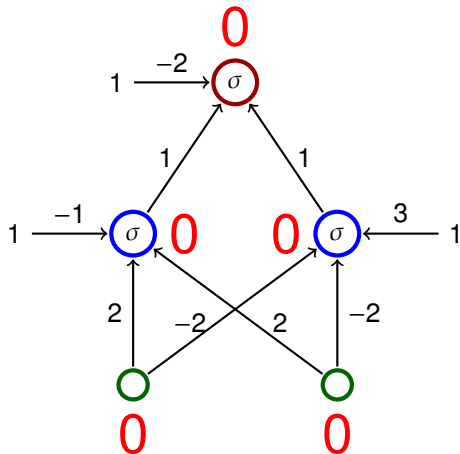
- ▶ Aktivační funkce je ostrá nelinearita

$$\sigma(\xi) = \begin{cases} 1 & \xi \geq 0; \\ 0 & \xi < 0. \end{cases}$$

- ▶ Síť počítá funkci $XOR(x_1, x_2)$

x_1	x_2	y
1	1	0
1	0	1
0	1	1
0	0	0

Aktivní dynamika - XOR



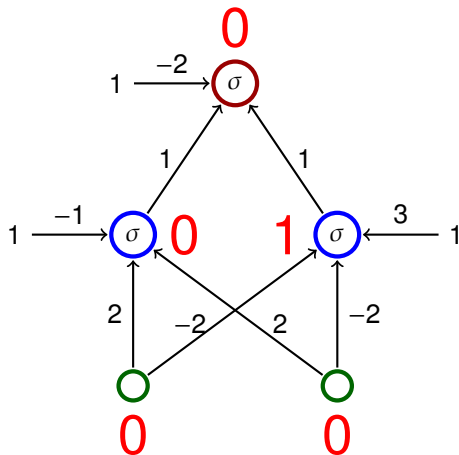
- ▶ Aktivační funkce je ostrá nelinearita

$$\sigma(\xi) = \begin{cases} 1 & \xi \geq 0; \\ 0 & \xi < 0. \end{cases}$$

- ▶ Síť počítá funkci $XOR(x_1, x_2)$

x_1	x_2	y
1	1	0
1	0	1
0	1	1
0	0	0

Aktivní dynamika - XOR



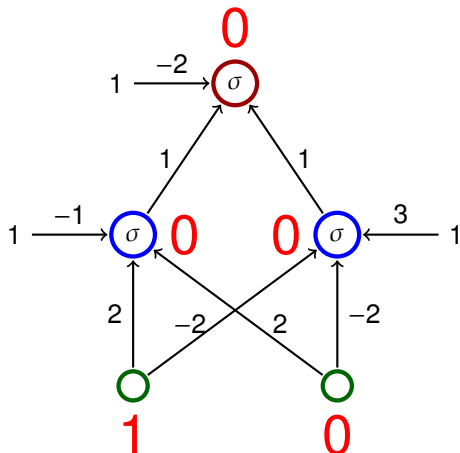
- ▶ Aktivační funkce je ostrá nelinearita

$$\sigma(\xi) = \begin{cases} 1 & \xi \geq 0; \\ 0 & \xi < 0. \end{cases}$$

- ▶ Síť počítá funkci $XOR(x_1, x_2)$

x_1	x_2	y
1	1	0
1	0	1
0	1	1
0	0	0

Aktivní dynamika - XOR



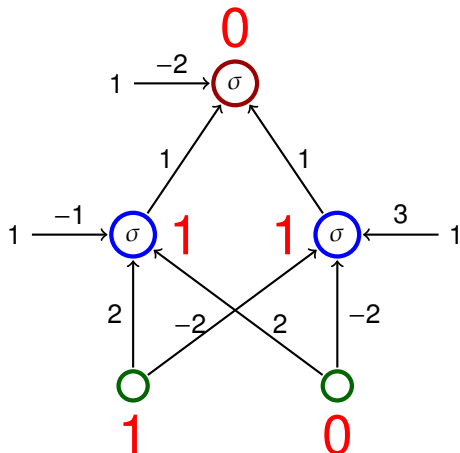
- ▶ Aktivační funkce je ostrá nelinearita

$$\sigma(\xi) = \begin{cases} 1 & \xi \geq 0; \\ 0 & \xi < 0. \end{cases}$$

- ▶ Síť počítá funkci $XOR(x_1, x_2)$

x_1	x_2	y
1	1	0
1	0	1
0	1	1
0	0	0

Aktivní dynamika - XOR



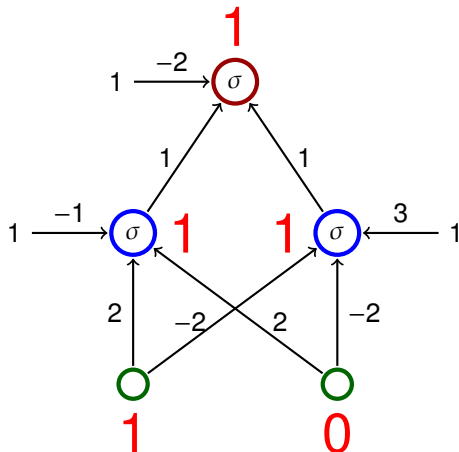
- ▶ Aktivační funkce je ostrá nelinearita

$$\sigma(\xi) = \begin{cases} 1 & \xi \geq 0; \\ 0 & \xi < 0. \end{cases}$$

- ▶ Síť počítá funkci $XOR(x_1, x_2)$

x_1	x_2	y
1	1	0
1	0	1
0	1	1
0	0	0

Aktivní dynamika - XOR



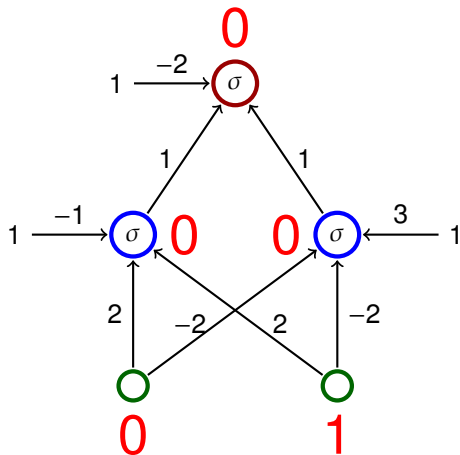
- ▶ Aktivační funkce je ostrá nelinearita

$$\sigma(\xi) = \begin{cases} 1 & \xi \geq 0; \\ 0 & \xi < 0. \end{cases}$$

- ▶ Síť počítá funkci $XOR(x_1, x_2)$

x_1	x_2	y
1	1	0
1	0	1
0	1	1
0	0	0

Aktivní dynamika - XOR



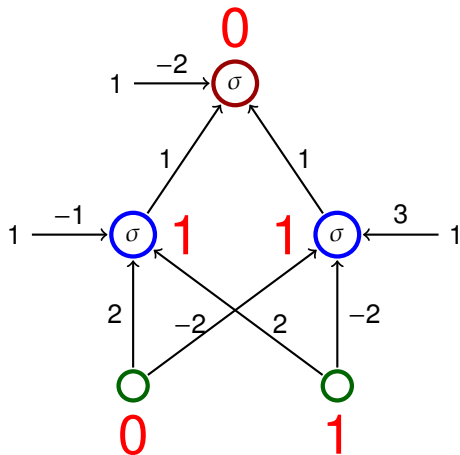
- ▶ Aktivační funkce je ostrá nelinearita

$$\sigma(\xi) = \begin{cases} 1 & \xi \geq 0; \\ 0 & \xi < 0. \end{cases}$$

- ▶ Síť počítá funkci $XOR(x_1, x_2)$

x_1	x_2	y
1	1	0
1	0	1
0	1	1
0	0	0

Aktivní dynamika - XOR



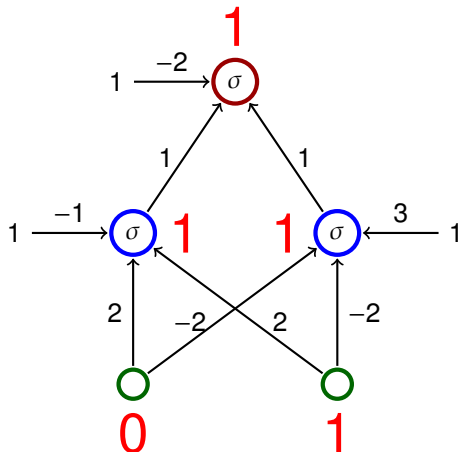
- ▶ Aktivační funkce je ostrá nelinearita

$$\sigma(\xi) = \begin{cases} 1 & \xi \geq 0; \\ 0 & \xi < 0. \end{cases}$$

- ▶ Síť počítá funkci $XOR(x_1, x_2)$

x_1	x_2	y
1	1	0
1	0	1
0	1	1
0	0	0

Aktivní dynamika - XOR



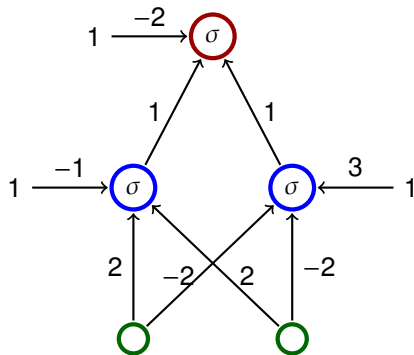
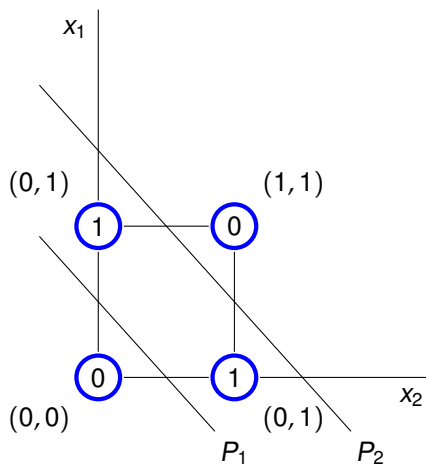
- ▶ Aktivační funkce je ostrá nelinearita

$$\sigma(\xi) = \begin{cases} 1 & \xi \geq 0; \\ 0 & \xi < 0. \end{cases}$$

- ▶ Síť počítá funkci $XOR(x_1, x_2)$

x_1	x_2	y
1	1	0
1	0	1
0	1	1
0	0	0

Neuron a lineární separace



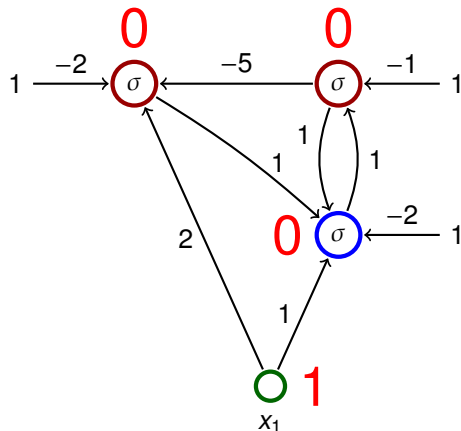
- ▶ Přímka P_1 má rovnici $-1 + 2x_1 + 2x_2 = 0$
- ▶ Přímka P_2 má rovnici $3 - 2x_1 - 2x_2 = 0$

Aktivní dynamika - příklad

Aktivační funkce ostrá
nelinearita

$$\sigma(\xi) = \begin{cases} 1 & \xi \geq 0; \\ 0 & \xi < 0. \end{cases}$$

vstup je roven 1

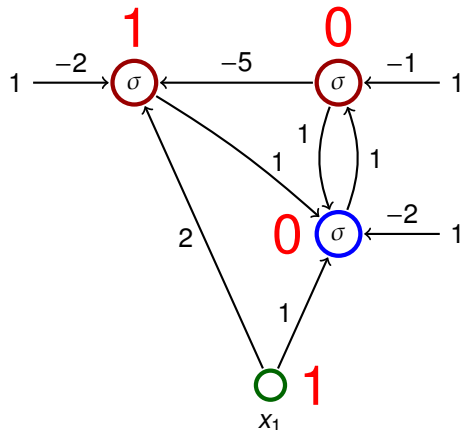


Aktivní dynamika - příklad

Aktivační funkce ostrá
nelinearita

$$\sigma(\xi) = \begin{cases} 1 & \xi \geq 0; \\ 0 & \xi < 0. \end{cases}$$

vstup je roven 1

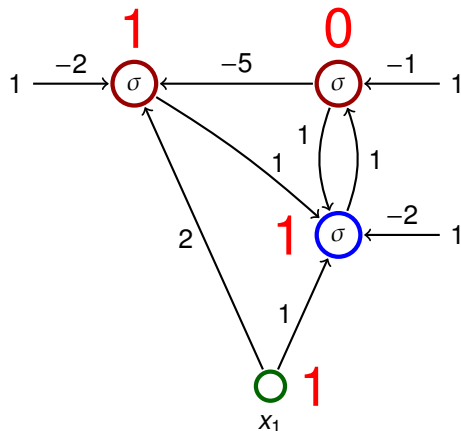


Aktivní dynamika - příklad

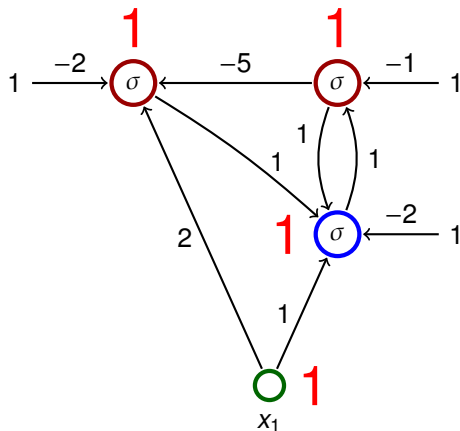
Aktivační funkce ostrá
nelinearita

$$\sigma(\xi) = \begin{cases} 1 & \xi \geq 0; \\ 0 & \xi < 0. \end{cases}$$

vstup je roven 1



Aktivní dynamika - příklad



Aktivační funkce ostrá
nelinearita

$$\sigma(\xi) = \begin{cases} 1 & \xi \geq 0; \\ 0 & \xi < 0. \end{cases}$$

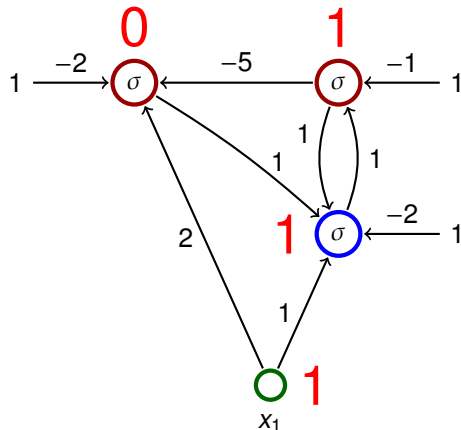
vstup je roven 1

Aktivní dynamika - příklad

Aktivační funkce ostrá
nelinearita

$$\sigma(\xi) = \begin{cases} 1 & \xi \geq 0; \\ 0 & \xi < 0. \end{cases}$$

vstup je roven 1



Adaptivní dynamika určuje, jakým způsobem se síť učí.

Adaptivní dynamika určuje, jakým způsobem se síť učí.

- ▶ **počáteční konfigurace**

váhy mohou být nastaveny buď náhodně nebo na základě
předběžné znalosti vstupů sítě

Adaptivní dynamika určuje, jakým způsobem se síť učí.

- ▶ **počáteční konfigurace**

váhy mohou být nastaveny buď náhodně nebo na základě předběžné znalosti vstupů sítě

- ▶ **učící pravidlo** pro (postupnou) adaptaci vah

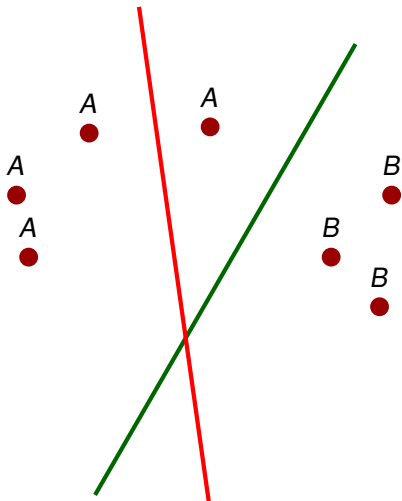
cílem je adaptovat váhy tak, aby síť počítala danou funkci

- ▶ učení s učitelem
 - ▶ Požadovaná funkce je zadána množinou *tréninkových vzorů* což jsou dvojice tvaru (vstup, výstup).
 - ▶ Při učení se hledá konfigurace sítě, která nejlépe odpovídá daným vzorům (vzhledem k danému kvalitativnímu kritériu).

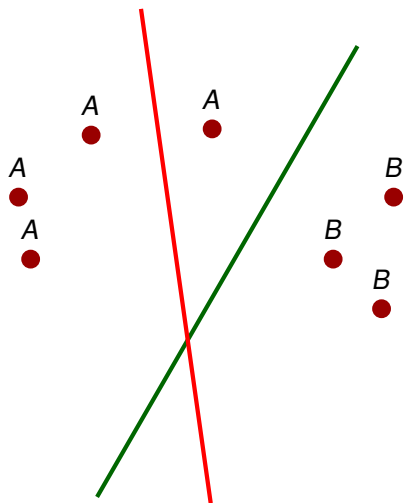
- ▶ učení s učitelem
 - ▶ Požadovaná funkce je zadána množinou *tréninkových vzorů* což jsou dvojice tvaru (vstup, výstup).
 - ▶ Při učení se hledá konfigurace sítě, která nejlépe odpovídá daným vzorům (vzhledem k danému kvalitativnímu kritériu).
- ▶ učení bez učitele
 - ▶ Tréninková množina obsahuje pouze vstupy sítě.
 - ▶ Cílem je odhalit strukturu v množině vstupů (shlukování, samoorganizace)

Učení s učitelem - ilustrace

- ▶ klasifikace v rovině pomocí jednoho neuronu

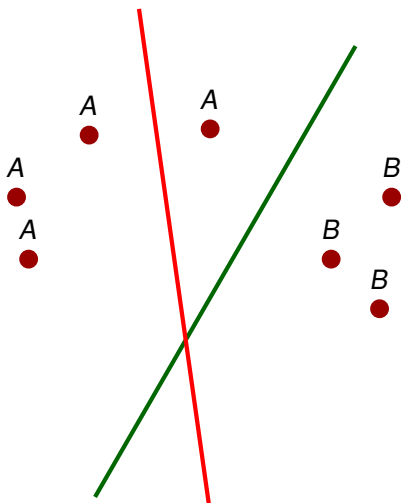


Učení s učitelem - ilustrace



- ▶ klasifikace v rovině pomocí jednoho neuronu
- ▶ tréninkové vzory jsou tvaru (bod, hodnota) kde hodnota je buď 1 nebo 0 podle toho zda je bod ze skupiny A nebo B

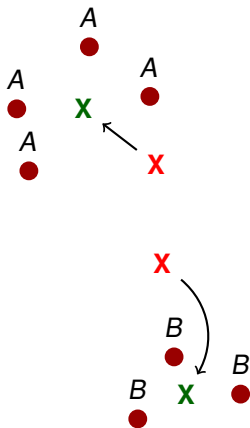
Učení s učitelem - ilustrace



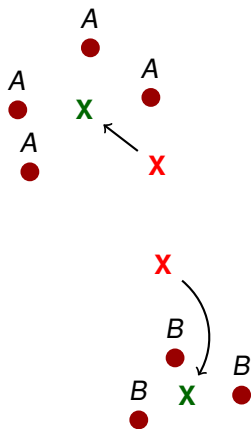
- ▶ klasifikace v rovině pomocí jednoho neuronu
- ▶ tréninkové vzory jsou tvaru (bod, hodnota) kde hodnota je buď 1 nebo 0 podle toho zda je bod ze skupiny A nebo B
- ▶ po předložení nesprávně klasifikovaného vzoru skupiny A (červená přímka), učící algoritmus pootočí přímku ve směru nesprávně klasifikovaného bodu (zelená přímka).

Učení bez učitele - ilustrace

- ▶ hledáme dva reprezentanty „shluků“

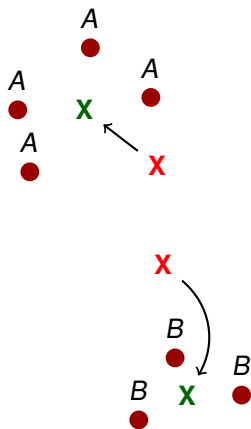


Učení bez učitele - ilustrace



- ▶ hledáme dva reprezentanty „shluků“
- ▶ červené křížky odpovídají reprezentantům před aplikací učícího algoritmu, zelené po aplikaci

Učení bez učitele - ilustrace



- ▶ hledáme dva reprezentanty „shluků“
- ▶ červené křížky odpovídají reprezentantům před aplikací učícího algoritmu, zelené po aplikaci
- ▶ učící algoritmus může např. napočítat množinu bodů, které jsou nejbližší danému reprezentantovi a potom reprezentanta posunout do těžiště této množiny bodů

Výhody umělých neuronových sítí

- ▶ Masivní paralelismus
 - ▶ neurony mohou být vyhodnocovány současně

Výhody umělých neuronových sítí

- ▶ Masivní paralelismus
 - ▶ neurony mohou být vyhodnocovány současně
- ▶ Adaptace a učení
 - ▶ existuje mnoho učících algoritmů, které „programují“ neuronové sítě na základě příkladů požadovaného chování

Výhody umělých neuronových sítí

- ▶ Masivní paralelismus
 - ▶ neurony mohou být vyhodnocovány současně
- ▶ Adaptace a učení
 - ▶ existuje mnoho učících algoritmů, které „programují“ neuronové sítě na základě příkladů požadovaného chování
- ▶ Schopnost generalizace a odolnost vůči nepřesnosti vstupu
 - ▶ informace jsou v síti kódovány *přibližně* pomocí vah mnoha neuronů
 - ▶ na vstup podobný vstupu tréninkového vzoru, reaguje naučená síť podobným výstupem
 - ▶ takto je schopna extrahovat charakteristické vlastnosti dat

Výhody umělých neuronových sítí

- ▶ Masivní paralelismus
 - ▶ neurony mohou být vyhodnocovány současně
- ▶ Adaptace a učení
 - ▶ existuje mnoho učících algoritmů, které „programují“ neuronové sítě na základě příkladů požadovaného chování
- ▶ Schopnost generalizace a odolnost vůči nepřesnosti vstupu
 - ▶ informace jsou v síti kódovány *přibližně* pomocí vah mnoha neuronů
 - ▶ na vstup podobný vstupu tréninkového vzoru, reaguje naučená síť podobným výstupem
 - ▶ takto je schopna extrahovat charakteristické vlastnosti dat
- ▶ Odolnost vůči poškození
 - ▶ poškození se obvykle projevuje postupnou ztrátou přesnosti výsledků