

Vážení zákazníci,

dovolujeme si Vás upozornit, že na tuto ukázkou knihy se vztahují autorská práva, tzv. copyright.

To znamená, že ukáзка má sloužit výhradně pro osobní potřebu potenciálního kupujícího (aby čtenář viděl, jakým způsobem je titul zpracován a mohl se také podle tohoto, jako jednoho z parametrů, rozhodnout, zda titul koupí či ne).

Z toho vyplývá, že není dovoleno tuto ukázkou jakýmkoliv způsobem dále šířit, veřejně či neveřejně např. umístováním na datová média, na jiné internetové stránky (ani prostřednictvím odkazů) apod.

redakce nakladatelství BEN – technická literatura
redakce@ben.cz



Po agregaci následuje operace nelineárního zobrazení s prahováním. Obecně lze tyto operace souhrnně popsat rovnicí:

$$y(t) = F[u(t), w_0], \tag{3.1.7}$$

kde F značí nelineární přenosovou (aktivační) funkci neuronu (kap. 3.1.2) a w_0 prahovou hodnotu. Je-li hodnota veličiny $u(t)$ nižší než hodnota prahu w_0 , je na výstupu signál, odpovídající pasivnímu stavu neuronu. Při překročení prahové hodnoty pak s rostoucí velikostí $u(t)$ výstupní signál monotónně roste až do určité saturované hodnoty. Zavedením další pomocné veličiny $v(t)$ můžeme rovnici (3.1.7) přepsat na tvar:

$$y(t) = F[v(t)], \quad v(t) = u(t) - w_0. \tag{3.1.8}$$

Z uvedeného tvaru a předcházejícího kvalitativního popisu nelineární funkce vyplývají požadavky, které by měla tato funkce splňovat. Pro nejpoužívanější funkci sigmoidu (kap. 3.1.2) platí:

$$y[v(t)] = X[v(t)] = 1 / \left(1 + e^{-v(t)} \right). \tag{3.1.9}$$

Vlastnosti základního modelu neuronu lze shrnout v *tab. 3.1.1*.

Tab. 3.1.1 Vlastnosti základního modelu neuronu

SYNAPTICKÉ OPERACE	Konfluence	$z_i(t) = x_i(t) \cdot w_i(t)$
SOMATICKÉ OPERACE	Agregace	$u(t) = \sum_{i=1}^n z_i(t)$
	Prahování	$v(t) = u(t) - w_0$
	Nelineární zobrazení (sigmoida)	$y(t) = 1 / (1 + e^{-v(t)})$

3.1.2 Neuronová síť a její dynamika

Každá neuronová síť je složena z formálních neuronů, které jsou vzájemně propojeny tak, že výstup jednoho neuronu je vstupem do dalších neuronů. Je tomu podobně, jako jsou terminály axonu biologického neuronu přes synaptické vazby spojeny s dendrity jiných neuronů. Počet neuronů a jejich vzájemné propojení v síti určuje **topologii (architekturu)** neuronové sítě. Z hlediska využití rozlišujeme v síti **vstupní, skryté a výstupní neurony**. Šíření a zpracování informace v síti je umožněno změnou stavů neuronů ležících na cestě mezi vstupními a výstupními neurony. Stavů všech neuronů v síti určují **stav** neuronové sítě a **synaptické váhy** všech spojů představují **konfiguraci** sítě. Neuronová síť se v čase vyvíjí, mění se stav neuronů a adaptují se váhy. V souvislosti se změnou těchto charakte-

ristik v čase lze rozdělit celkovou **dynamiku** neuronové sítě do tří dynamik a uvažovat pak **tři režimy práce** neuronové sítě:

- **organizační,**
- **aktivní,**
- **adaptivní.**

Organizační režim práce sítě spočívá ve změně topologie sítě. **Aktivní režim** práce sítě je spojen se změnou stavu sítě. Při **adaptivním režimu** práce sítě se mění konfigurace sítě.

Výše uvedené dynamiky neuronové sítě jsou obvykle zadány počátečním stavem sítě a matematickou rovnicí nebo pravidlem, které určuje vývoj příslušné charakteristiky sítě v čase. Vývojem příslušné charakteristiky se myslí topologie, stav nebo konfigurace sítě. Konkretizací jednotlivých dynamik obdržíme různé modely neuronových sítí, které jsou vhodné pro řešení různých skupin úloh. V následujícím jsou proto dynamiky blíže popsány.

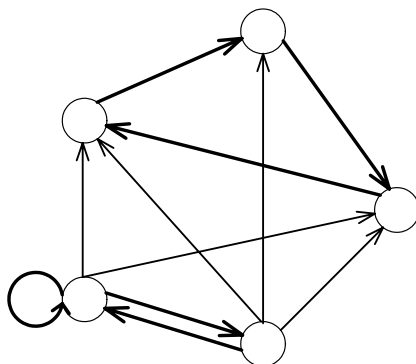
Organizační dynamika

Organizační dynamika specifikuje **topologii** neuronové sítě. Většinou v technické praxi pracujeme s **pevnou topologií** sítě, jejíž architektura se v čase nemění. Tomuto stavu sítě však předchází její důkladný rozbor a až potom stanovení definitivní a pevné topologie sítě, tedy pevná architektura sítě je až následný krok po adaptivním režimu práce sítě, kdy je síť v případě konkrétní praktické potřeby rozšířena o další neurony, je určen jejich stav a proveden rozbor konfigurace sítě.

U **pevné topologie** rozlišujeme **dva typy** sítí:

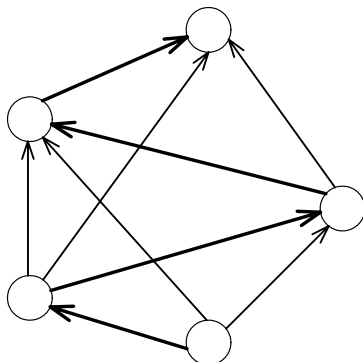
- **cyklické (rekurentní),**
- **acyklické (dopředné).**

V případě **cyklické** topologie existuje v síti skupina neuronů, která je spojena v kruhu (tzv. cyklus). To znamená, že v této skupině neuronů je výstup prvního neuronu vstupem druhého neuronu, jehož výstup je opět vstupem třetího neuronu, atd., až výstup posledního neuronu v této skupině je vstupem prvního neuronu. Nejjednodušším příkladem cyklu je **zpětná vazba** neuronu, jehož výstup je zároveň jeho vstupem. Nejvíce cyklů je v **úplné** topologii cyklické neuronové sítě, kde výstup libovolného neuronu je vstupem každého neuronu. Příklad cyklické neuronové sítě je uveden na *obr. 3.1.3*, kde lze sledovat všechny možné cykly.



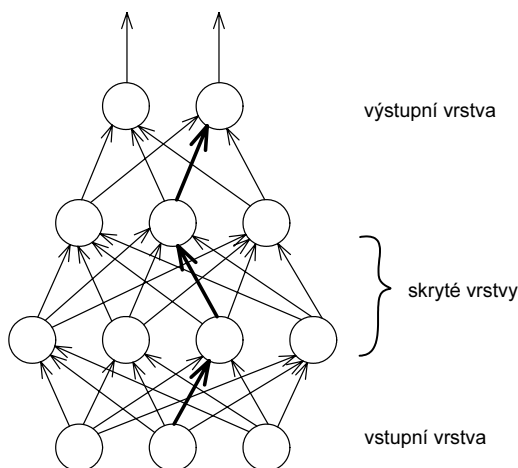
Obr. 3.1.3 Příklad cyklické neuronové sítě

V **acyklických** sítích neexistuje cyklus a všechny cesty vedou jedním směrem. Příklad acyklické sítě je na *obr. 3.1.4*, kde je silnějšími šipkami vyznačena nejdelší cesta.



Obr. 3.1.4 Příklad acyklické neuronové sítě

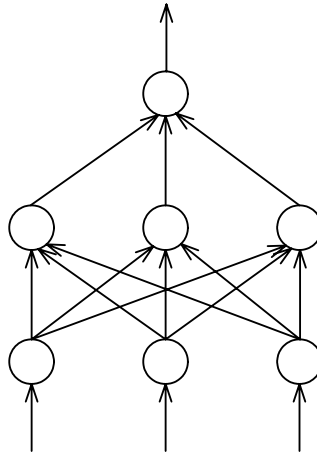
U acyklické sítě lze neurony vždy rozdělit do **vrstev**, které jsou uspořádány tak, že spoje mezi neurony vedou jen z nižších vrstev do vrstev vyšších. Speciálním případem takové topologie je **vícevrstvá neuronová síť**. Vícevrstvé neuronové sítě nacházejí v technické praxi značné uplatnění. Tyto sítě se skládají ze **vstupní vrstvy**, **výstupní vrstvy** a **skryté vrstvy** (jedné nebo častěji více), kde v jednotlivých vrstvách je určitý počet příslušných neuronů (vstupní neurony, výstupní neurony, skryté neurony). Počet jednotlivých neuronů je specifikován číslem v označení neuronové sítě. Příklad architektury vícevrstvé neuronové sítě s označením 3-4-3-2 je na *obr. 3.1.5*.



Obr. 3.1.5 Příklad vícevrstvé neuronové sítě

Jedná se o neuronovou síť, kde je vstupní vrstva se 3 neurony, výstupní vrstva s 2 neurony a dále dvě skryté vrstvy po 4 a 3 neuronech.

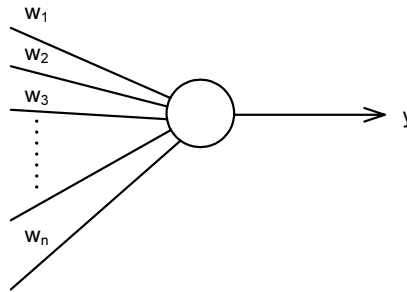
K nejstarším vícevrstvým sítím patří především síť typu **perceptron**. Perceptron je nej-jednodušší třívrstvá neuronová síť dle *obr. 3.1.6*, která byla schopna učení a byla podrobně popsána v roce 1957 Frankem Rosenblattem.



Obr. 3.1.6 Schéma neuronové sítě typu perceptron

První vrstva v této síti nemá měnitelné váhy. Tato vrstva provádí konstantní transformaci ze vstupu. Druhá vrstva je vrstva rovněž s pevnými váhami a vrstva třetí (výstupní) je nejdůležitější, neboť obsahuje perceptrony – rozpoznávače vzorů (*pattern recognizers neboli perceptrons*). Váhy na vstupech vrstvy výstupní při trénování nastavujeme.

Perceptronem bývá tedy často nazýván **samostatný neuron** dle obr. 3.1.7, který má nastavitelné váhy u spojů ($w_1, w_2, \dots, w_n$), které vstupují do výstupní vrstvy. Perceptron je tedy z hlediska měnitelných vah jednovrstvou sítí, ale i přes to bývá zařazován mezi vícevrstvé sítě.

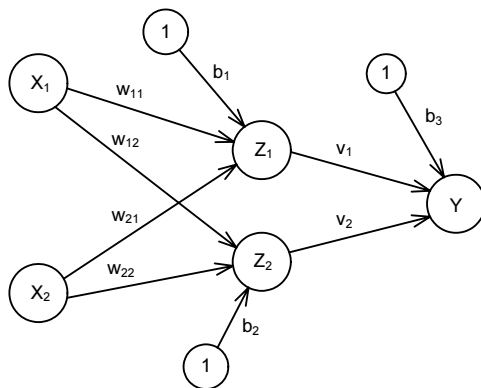


Obr. 3.1.7 Samostatný neuron

Perceptron byl Rosenblattem vyvinut na základě existence dvou poznatků, a to na základě McCullochova-Pittsova modelu neuronu (tento měl pevně nastavené váhy) a faktu prezentovaného D. O. Hebbem, že učení u biologických neuronů je realizováno díky změnám v synapsích mezi neurony. Pokud bychom napojili vstupní vektor přímo na výstupní vrstvu, pak bychom dostali tzv. **jednovrstvý perceptron**.

Na závěr této kapitoly bude ještě stručně popsána neuronová síť s označením **madaline** (**M**any **A**Daptive **L**inear **N**euron). Základním prvkem v tomto modelu je neuron adaline, který je značně podobný perceptronu, který byl popsán výše.

Jednoduchá architektura neuronové sítě madaline je zobrazena na obr. 3.1.8.



Obr. 3.1.8 Neuronová síť s označením madaline

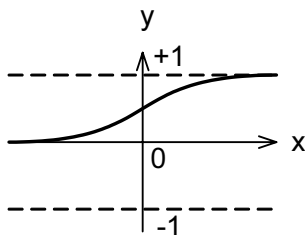
Zde jsou dva skryté neurony typu adaline (Z_1, Z_2) a jeden výstupní neuron rovněž typu adaline (Y). Tato neuronová síť se z hlediska adaptačního algoritmu značně vyvíjela, původní adaptační algoritmus (označení MRI) z roku 1960 adaptuje pouze váhové hodnoty příslušející oběma skrytým neuronům, zatímco váhové hodnoty příslušející výstupnímu neuronu jsou fixní. Váhové hodnoty v_1 a v_2 a bais b_3 jsou určeny tak, že výstupní signál z Y je roven 1, pokud je alespoň jedna hodnota signálu vycházejícího ze skrytých neuronů rovna jedné. Pokud jsou oba signály vysílané ze Z_1 a Z_2 rovny -1 , má výstupní signál z Y hodnotu -1 . Adaptační algoritmus (označení MRII) z roku 1987 naproti algoritmu dřívějšímu upravuje všechny váhové hodnoty.

Aktivní dynamika

Aktivní dynamika specifikuje počáteční stav sítě a způsob jeho změny v čase při pevné topologii a konfiguraci. V aktivním režimu se na začátku nastaví stavy vstupních neuronů na vstup sítě a zbylé neurony jsou v uvedeném počátečním stavu. Po inicializaci stavu sítě probíhá vlastní výpočet. Obecně se předpokládá spojitý vývoj stavu neuronové sítě v čase a hovoří se o **spojitém modelu**, kdy stav sítě je spojitou funkcí času, která je obvykle v aktivní dynamice zadána diferenciálními rovnicí. Většinou se však předpokládá **diskrétní čas** (tj. na začátku se síť nachází v čase 0 a stav sítě se mění jen v čase 1, 2, 3, ...). V každém takovém časovém kroku je podle daného pravidla aktivní dynamiky vybrán jeden neuron nebo více neuronů, které mění (aktualizují) svůj stav na základě svých vstupů, tj. stavů sousedních neuronů, jejichž výstupy jsou vstupy aktualizovaných neuronů. Je-li vybrán jeden neuron, hovoříme o tzv. **sekvenčním výpočtu**, v případě výběru více neuronů se jedná o tzv. **paralelní výpočet**. Podle toho, zda neurony mění svůj stav nezávisle na sobě nebo je jejich aktualizace řízena centrálně, rozlišujeme synchronní a asynchronní modely neuronových sítí. Stav výstupních neuronů, který se obecně mění v čase, je výsledkem výpočtu a výstupem neuronové sítě. Obvykle se však uvažuje taková aktivní dynamika, že výstup sítě je po nějakém čase konstantní a neuronová síť tak v aktivním režimu realizuje nějakou **přenosovou funkci**, tj. ke každému vstupu sítě vypočítá právě jeden výstup. Je zřejmé, že v aktivním režimu se neuronová síť využívá k vlastním výpočtům.

V praxi se můžeme setkat s různými **přenosovými funkcemi**. Dále jsou nejčastěji používané přenosové funkce uvedeny, a to nejprve jejich grafický průběh, označení, zkratka a nakonec matematické vyjádření.

Přenosová funkce – sigmoida (sigmoidální) funkce



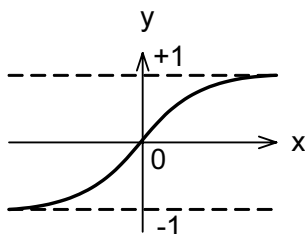
Obr. 3.1.9 Přenosová funkce – sigmoida (sigmoidální funkce)

Přenosová funkce – sigmoida bývá v zahraniční literatuře označována také názvem: *Log-Sigmoid Transfer Function* a zkratkou: $y = \text{logsig}(x)$. Matematické vyjádření je dáno vztahem 3.1.10.

$$y = \frac{1}{1 + e^{-\lambda x}}, \quad (3.1.10)$$

kde λ je strmost, $x \rightarrow \infty$: úplná excitace vnitřního potenciálu neuronu, $x \rightarrow -\infty$: úplná inhibice vnitřního potenciálu neuronu.

Přenosová funkce – hyperbolická tangenta

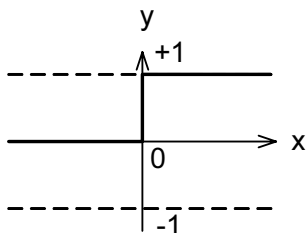


Obr. 3.1.10 Přenosová funkce – hyperbolická tangenta

Přenosová funkce – hyperbolická tangenta bývá v zahraniční literatuře označována také názvem: *Tan-Sigmoid Transfer Function* a zkratkou: $y = \text{tansig}(x)$. Matematické vyjádření je dáno vztahem 3.1.11.

$$y = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (3.1.11)$$

Přenosová funkce – skoková funkce unipolární

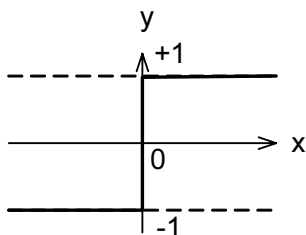


Obr. 3.1.11 Přenosová funkce – skoková funkce unipolární

Přenosová funkce – skoková funkce unipolární bývá v zahraniční literatuře označována také názvem: *Hard Limited Transfer Function* a zkratkou: $y = \text{hardlim}(x)$. Matematické vyjádření je dáno vztahem 3.1.12.

$$\begin{aligned} y &= 0 & x < 0 \\ y &= +1 & x \geq 0 \end{aligned} \tag{3.1.12}$$

Přenosová funkce – skoková funkce bipolární

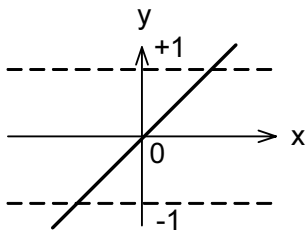


Obr. 3.1.12 Přenosová funkce – skoková funkce bipolární

Přenosová funkce – skoková funkce bipolární bývá v zahraniční literatuře označována názvem: *Symmetric Hard Limited Transfer Function* a zkratkou: $y = \text{hardlims}(x)$. Matematické vyjádření je dáno vztahem 3.1.13.

$$\begin{aligned} y &= -1 & x < 0 \\ y &= +1 & x \geq 0 \end{aligned} \tag{3.1.13}$$

Přenosová funkce – lineární funkce

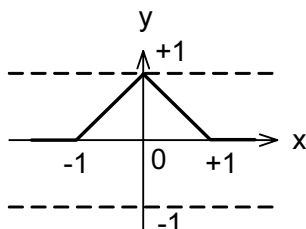


Obr. 3.1.13 Přenosová funkce – lineární funkce

Přenosová funkce – lineární funkce bývá v zahraniční literatuře označována názvem: *Linear Transfer Function*. Matematické vyjádření je dáno vztahem 3.1.14.

$$y = k \cdot x \quad (3.1.14)$$

Přenosová funkce – trojúhelníková funkce

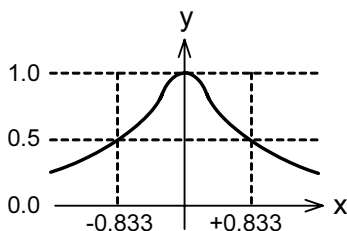


Obr. 3.1.14 Přenosová funkce – trojúhelníková funkce

Přenosová funkce – trojúhelníková funkce bývá v zahraniční literatuře označována názvem: *Triangular Basic Function* a zkratkou: $y = \text{tribas}(x)$. Matematické vyjádření je dáno vztahem 3.1.15.

$$y = \text{tribas}(x) \quad \begin{cases} y = 0 & x \leq -1 \\ y = k \cdot x & -1 < x < 0 \\ y = -k \cdot x & 0 < x < +1 \\ y = 0 & x \geq +1 \end{cases} \quad (3.1.15)$$

Přenosová funkce – lichoběžníková funkce

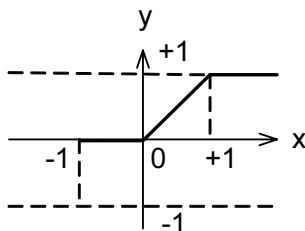


Obr. 3.1.15 Přenosová funkce – lichoběžníková funkce

Přenosová funkce – lichoběžníková funkce bývá v zahraniční literatuře označována názvem: *Radial Basis Transfer Function* a zkratkou: $y = \text{radbas}(x)$. Matematické vyjádření je dáno vztahem 3.1.16.

$$y = e^{\left(-\frac{(x-c)^2}{r^2} \right)}, \quad (3.1.16)$$

kde c je střed a r je poloměr.



Obr. 3.1.16 Přenosová funkce – lineární prahová funkce

Tato funkce bývá v zahraniční literatuře označována názvem: *Satlin Transfer Function* a zkratkou: $y = \text{satlin}(x)$. Matematické vyjádření je dáno vztahem 3.1.17.

$$\begin{aligned} y &= 0 & x &\leq 0 \\ y &= k \cdot x & 0 < x < +1 \\ y &= +1 & x &\geq +1 \end{aligned} \quad (3.1.17)$$

Adaptivní dynamika

Adaptivní dynamika neuronové sítě specifikuje počáteční konfiguraci sítě a způsob, jakým se mění váhové hodnoty na spojeních mezi jednotlivými neurony v čase. Všechny možné konfigurace sítě tvoří **váhový prostor** neuronové sítě. V adaptivním režimu se tedy na začátku nastaví váhy všech spojů v síti na počáteční konfiguraci. V praxi se často na počátku nastaví váhy všech spojů náhodně. Po inicializaci **konfigurace** sítě probíhá vlastní **adaptace**. Podobně jako v aktivní dynamice, obecně se uvažuje spojitý model se spojitým vývojem konfigurace neuronové sítě v čase, kdy váhy sítě jsou spojitou funkcí času. Častěji se však předpokládá diskrétní čas adaptace. Jak již bylo konstatováno, vlastní funkce sítě v aktivním režimu závisí na konfiguraci.

Cílem adaptace je nelézt takovou konfiguraci sítě ve váhovém prostoru, která by v aktivním režimu realizovala předepsanou funkci. Jestliže se aktivní režim sítě využívá k vlastnímu výpočtu funkce sítě pro daný vstup, potom adaptivní režim slouží k **učení** této funkce. Způsob adaptace, kdy požadované chování sítě modeluje učitel, se nazývá **učení s učitelem** (*supervised learning*).

Blokové schéma učení neuronové sítě s učitelem je na obr. 3.1.17.

Neuronové síti při učení s učitelem [16] jsou postupně předkládány dvojice vstup – požadovaný výstup z trénovací množiny. Požadovaný výstup představuje informaci učitele, jak má síť na příslušný vstup zareagovat. Po předložení vstupu je vypočítán aktuální výstup sítě, který se porovnává s požadovaným výstupem. Učící algoritmus pak upraví hodnoty vah tak, aby rozdíl mezi skutečným a požadovaným výstupem byl minimální. Tento rozdíl je poměřován tzv. **chybovou funkcí**, která je znázorněna graficky a také popsána později.

Vhodný průběh učení se pozná podle hladké chybové funkce, která vykazuje na počátku strmý pokles. To indikuje vhodně zvolenou velikost trénovací množiny a dobře zvolené parametry algoritmu učení.