

## Úvod do práce s Neural Network Toolboxem, MATLAB

Podporou pro řešení úloh pomocí umělých neuronových sítí je NN-Toolbox Matlabu. Současné verze umožňují práci s jednoduchými modely, které tvoří základ práce s UNS, ale také s mnoha novými metodami, které byly vyvinuty v posledních letech. Pomocí demonstračních úloh je možné se seznámit s jednotlivými fázemi návrhu sítě, jejího trénování a zpracování výsledků. Toolbox také obsahuje aplikační úlohy. Jeho úspěšné používání předpokládá základní znalosti z programování a z teorie neuronových sítí. V práci uživateli pomáhá Help, který obsahuje všechny potřebné informace. Přesto na tomto místě uvádím alespoň základní pojmy. Podrobnější seznámení s NN-Toolboxem zůstává na uživateli. Pouze experimentování s neuronovými sítěmi umožní porozumět teorii a přispěje k úspěšným aplikacím.

Uvedu zde přehled funkcí, které se v této variantě NN-Toolboxu vyskytují, abych umožnila snazší orientaci při využívání helpu. Na tomto místě uvedu jen názvy hlavních kategorií, pouze u několika z nich přidám více podrobností.

Pro rozdělení funkcí podle typu UNS voláme „**Functions: By Category**“:

Např. „**Training Functions**“

<b>train</b>	trénování neuronové sítě (Train neural network)
<b>trainb</b>	dávkové učení (Batch training with weight and bias learning rules)
<b>traingd</b>	základní gradientní učení BPG (Gradient descent backpropagation)
<b>traingda</b>	učení BPG s adaptivním koeficientem učení (Gradient descent with adaptive learning rate backpropagation)
<b>selforgmap</b>	samoorganizující se mapy (Self-organizing map)

„**Performance Functions**“

<b>mse</b>	průměrná chyba (Mean squared normalized error performance fiction)
<b>sse</b>	suma chyb (Sum squared error performance fiction)

Pro rozdělení funkcí podle tříd volíme „**Data Functions, Distance Functions, Graphical Interface Functions, Layer Initialization Functions, Learning Functions, Line Search Functions, Net Input Functions, Network Initialization Functions, network Use functions, New Network Functions, Performance Functions, Plotting Functions, Processing Functions, Simulink Support Functions, Topology Functions, Training Functions, Transfer Functions, Weight and Bias Initializavtion Functions, Weight Functions, Transfer Function Graphs**“.

### Experimentování:

V Matlabu zvolíme „ Help – Product Help (F1). **V levém okně** se rozbalí nabídka „těla“ Matlabu a Toolboxu. Budeme pracovat s „Neural Network Toolboxem“. Po rozkliknutí se objeví nabídka , ve které jednotlivé položky jsou popisovány textově s příklady – formou manuálu. Jedno z hesel je „Demos“, kde jsou demo-verze. Jejich popis se rozbalí v pravém okně. **V pravém okně** je důležitá nabídka „Functions“ – viz výše.

Rychlá orientace je pod „Help – Function Browser“

Pro snadnější orientaci v NN-Toolboxu je v tabulce D2 uveden rozdíl mezi značením funkcí a proměnných mezi teoretickými úvahami ve skriptu a v monografii (korespondují s velkou většinou literárních pramenů týkajících se umělých neuronových sítí a jejich aplikací ve světě) a značením v toolboxu.

Název - UNS	Značení - UNS	Značení - MATLAB
vstupy do neuronu	$x_i, i = 1, \dots, n$	$p_i, i = 1, \dots, R$
obvodové funkce	$u$	$n$
výstup z neuronu	$y$	$a$
aktivační funkce	$f$	$f$

Tabulka D2: Značení funkcí a proměnných ve skriptu a NN-Toolboxu v MATLABu.

V následujících řádcích si ukážeme, jak pracovat s toolboxem. Pro jednoduchý neuron platí:

- vstup (*input*) je označen ‘ $p$ ’ (obecně je to vektor o  $R$ -elementech)
- váha (*weight*) je označena ‘ $w$ ’ (obecně matice vah  $\mathbf{W}$ ); je-li použita matice vah, představují řádky rozložení neuronů ve vrstvě, sloupce určují připojené vstupy; např.  $w_{1,2}$  určuje, že signál je šířen ze vstupu 2 do neuronu 1)
- aktivační (přenosová) funkce (*transfer function*) je označena ‘ $f$ ’
- výstupní hodnota (*output*) je označen ‘ $a$ ’
- práh (*bias*) je označen ‘ $b$ ’.

Pro vrstevnatou UNS platí, že vstupem je buď vektor  $\mathbf{p}$  s  $R$ -vstupy, resp. matice vstupů s  $Q$ -vektory; pak je dimenze vstupní matice  $(R, Q)$ . Matice vah  $\mathbf{W}^1$  má hodnot  $(S^1, R)$ , kde  $S^1$  je počet neuronů v první skryté vrstvě a  $R$  je počet vstupů.

$$\mathbf{W} = \begin{bmatrix} w_{1,1} & w_{1,2} & \dots & w_{1,R} \\ w_{2,1} & w_{2,2} & \dots & w_{2,R} \\ \dots & \dots & \dots & \dots \\ w_{S,1} & w_{S,2} & \dots & w_{S,R} \end{bmatrix}$$

Matice vah  $\mathbf{W}^2$  má hodnot  $(S^2, S^1)$ , kde  $S^1$  je počet neuronů v první skryté vrstvě a  $S^2$  je počet neuronů ve druhé skryté vrstvě, resp. ve výstupní vrstvě. Z každé vrstvy vystupují vektory  $\mathbf{a}$  a pro jednotlivé neurony vrstvy jsou definovány prahy  $\mathbf{b}$ , které tvoří vektor  $\mathbf{b}$ , oba mají  $S^i$ -elementů. V následujících řádcích ukážeme způsob zápisu vektorů a matic.

$$\begin{aligned} \mathbf{W} &= [1 \ 2] & \text{net.IW}\{1,1\} &= [1,2]; \\ \mathbf{b} &= [0] & \text{net.b}\{1\} &= 0; \end{aligned}$$

Vstupní vektory .

$$p_1 = [1 \ 2]^T \quad p_2 = [2 \ 1]^T \quad p_3 = [2 \ 3]^T \quad p_4 = [3 \ 1]^T$$

$$P = [p_1, p_2, p_3, p_4] = [1 \ 2 \ 2 \ 3; 2 \ 1 \ 3 \ 1];$$

Výstup z neuronové sítě bude zapsán jako

$$\begin{aligned} a &= \text{sim}(\text{net}, P) \\ a &= \\ &5 \ 4 \ 8 \ 5 \end{aligned}$$

Vstupní váhy se v této verzi NN-Toolboxu označují jako *input weights* – např.  $\mathbf{IW}^{1,2}$  (ze vstupu 2 do neuronu1). V této verzi MATLABu platí

$$\mathbf{IW}^{1,2} \longrightarrow \text{net.IW}\{1,2\}$$

Pro váhy ve vrstvě je zavedeno označení *layer weights* –  $\mathbf{LW}$ .

Pro potenciál neuronu lze např. psát:

$$n\{1\} = \text{net.IW}\{1,2\} * p + \text{net.b}\{1\}$$

Příklad zápisu pro 3-vrstvé (podle značení MATLABu) UNS:

- pro výstupy z jednotlivých vrstev

$$\begin{aligned} a^1 &= f^1(\mathbf{IW}^{1,1} p + b^1) \\ a^2 &= f^2(\mathbf{IW}^{2,1} p + b^2) \\ a^3 &= f^3(\mathbf{IW}^{3,1} p + b^3) \end{aligned}$$

- pro výstup z celé sítě

$$a^3 = f^3(\mathbf{IW}^{3,2} f^2(\mathbf{LW}^{2,1} f^1(\mathbf{IW}^{1,1} p + b^1) + b^2) + b^3)$$

Počet vstupů je  $R^1$ , počet neuronů v první (skryté) vrstvě je  $S^1$ , počet neuronů v druhé (skryté) vrstvě je  $S^2$  atd. Tedy např. 3-vrstvá UNS má 1 výstup (layer 3) a 2 skryté vrstvy (layer 1 a layer 2). Vstupy se nepovažují za vrstvu.<sup>1</sup>

Rozlišují se dva základní typy vstupních vektorů, a to vektory vstupující do sítě v jednom čase – *concurrent inputs* (nezáleží na pořadí vektorů) a vektory vstupující do sítě sekvenčně – *sequential inputs* (pořadí vektorů je rozhodující).

### Vykreslení aktivačních funkcí

n = -5: 0.1:5;  
a = purelin(n);  
plot (n, a)

n = -5: 0.1:5;  
a = satlin(n);  
plot (n, a)

n = -5: 0.1:5;  
a = satlins(n);  
plot (n, a)

n = -5: 0.1:5;  
a = tansig(n);  
plot (n, a)

n = -5: 0.1:5;  
a = hardlim(n);  
plot (n, a)

n = -5: 0.1:5;  
a = radbas(n);  
plot (n, a)

n = -5: 0.1:5;  
a = logsig(n);  
plot (n, a)

<sup>1</sup> Rozdílné značení vrstev v MATLABu od skript.