

# Viacvrstvé neurónové siete s učiacimi sa produktovými neurónmi

Slavomír Holenda, Kristína Malinovská a Ľudovít Malinovský

Katedra aplikovanej informatiky, FMFI,  
Univerzita Komenského v Bratislave  
Mlynská dolina, 84248 Bratislava  
Email: {holenda1,malinovska}@fmph.uniba.sk

## Abstrakt

Klasické neurónové siete dosahujú obmedzené výsledky pri probléme parity alebo dvoch špirál pri malom počte neurónov. V tomto článku prezentujeme experimenty s nami navrhnutým konceptom nového neurónu, ktorý používa násobenie namiesto sumácie. Špecifický je tým, že je možné ho učiť a zároveň ho netreba ďalej obmedzovať alebo stabilizovať. Taktiež ho vieme zapojiť do architektúr s klasickými neurónmi. Testovali sme ho na probléme dvoch špirál a porovnávali jeho rôzne zapojenia na probléme parity 7. stupňa. Naše výsledky preukázali efektívnosť použitia produktových vrstiev.

## 1 Úvod

Napriek tomu že je násobenie v neurónových sieťach teoreticky preskúmané, v terajšom state-of-the-art výskume ho nájdeme iba zriedka. Vo väčšine prípadov sa totiž nepodarilo násobenie zovšeobecniť a prekonať problémy implementácie. V predchádzajúcom príspevku (Malinovský, Ľ. and Malinovská, K., 2022) sme práve na konferencii KUŽ prvýkrát predstavili náš model siete s produktovými neurónmi, ktoré majú trénovateľné váhy.

V histórii neurónových sietí sa už v 60-tych rokoch objavili neuróny vyššieho rádu, kde sa vstupný vektor rozšíril o vynásobené a umocnené už existujúce vstupy (Nilsson, 1965). Pri aplikovaní spätného šírenia chyby na takéto neuróny hovoríme o tzv. *sigma-pí* sieťach (Rumelhart a spol., 1986). Náš model vychádza z modelu, ktorý navrhli Ghosh a Shin (1992). Ten najprv vstupy lineárne kombinuje a až následne ich násobí. Nelineárna aktivačná funkcia sa použije až na poslednej vrstve, ktorá sa ako jediná učí pomocou gradientovej metódy. Váhy neurónov predošlých vrstiev sú nastavené na hodnotu 1 a nemenia sa. Tento model vo všeobecnosti prekonáva silu klasického perceptrónu, dokonca je porovnateľný so sieťami s neurónmi vyššieho rádu, avšak neumožňuje trénovanie váh produktových neurónov.

Potenciál násobenia v neurónových sieťach analyzuje Schmitt (2002), ktorý porovnáva rôzne typy sietí s násobením. Aj v oblasti hlbokých neurónových

sietí nájdeme modely, ktoré využívajú násobenie, ako napríklad sum-product siete (Poon a Domingos, 2011; Delalleau a Bengio, 2011) alebo modely, ktoré využívajú násobenie na vybraných miestach (Zhu a spol., 2018; Diba a spol., 2017).

## 2 Náš model

Náš model (Malinovský, Ľ. and Malinovská, K., 2022) sme predstavili ako klasickú pi-sigma sieť (Ghosh a Shin, 1992) s dvoma vrstvami - klasickou sumačnou a následne produktovou. Vrstvu produktových neurónov, tak ako sme ju navrhli my, je však možné zapájať v sieť ľubovoľne na požadované miesta a adaptovať jej váhy pri trénovaní modelu. Ďalšou výhodou je, že náš model netreba pri a po trénovaní stabilizovať alebo obmedzovať.

Pre vstup  $x$ , bez trénovateľného prahu (bias), a váhy  $w$  počítame aktiváciu neurónov ako

$$y_i = \prod_j (1 - \sigma(w_{ij})(1 - x_j)), \quad (1)$$

kde  $\sigma$  je logistická funkcia. Váhy našich produktových neurónov nie sú fixné ako pri pôvodnej pi-sigma sieti. Keďže na ne aplikujeme sigmoidálnu funkciu, pri výpočtoch sa ich hodnoty pohybujú v rozmedzí  $\langle 0, 1 \rangle$ . Následne je použitá analógia produktového neurónu s tým rozdielom, že váhy vstupujú do špeciálnej aktivačnej funkcie, ktorá má skoro totožné vlastnosti s umocnením na čísla z intervalu  $\langle 0, 1 \rangle$ . Narozdiel od umocnenia, je táto funkcia spojitá a spojitou derivovateľná, takže vieme vrstvu takýchto neurónov učiť pomocou gradientových metód. Váhu produktového neurónu si vieme predstaviť ako percento dôležitosti. Určuje v akej miere sa hodnota z presynaptického neurónu použije v násobení.

Pre výstupnú vrstvu môžeme pomocou metódy gradientového zostupu pre strednú kvadratickú chybu odvodiť nasledovné pravidlo pre učenie výstupných váh:

$$\frac{\partial E}{\partial w_{ij}^{\text{out}}} = (d_i - y_i) \left( \prod_{k \neq j} 1 - \sigma(w_{ik}^{\text{out}})(1 - h_k) \right) (h_j - 1) \sigma(w_{ij}^{\text{out}})(1 - \sigma(w_{ij}^{\text{out}})) \quad (2)$$

Ak sa vo výpočtoch nevyskytne delenie nulou, vieme výpočet nahradiť delením nasledovne:

$$\prod_{k \neq j} 1 - \sigma(w_{ik}^{\text{out}})(1 - h_i) = \frac{y_i}{1 - \sigma(w_{ij}^{\text{out}})(1 - h_j)} \quad (3)$$

### 3 Viacvrstvá sieť a experimenty

V tejto časti prezentujeme experimenty a výsledky s naším modelom rozšíreným na viacero skrytých vrstiev. Pre problém parity predstavujeme podrobné preskúmanie možností zapojenia pre dve skryté vrstvy. Na záver ukážeme predbežné výsledky pre problém dvoch špirál s tromi skrytými vrstvami.

#### 3.1 Problém parity

Problém parity je základný nelineárny problém, ktorý sa používa na základné overovanie nových modelov neurónových sietí. Problém spočíva v tom, že na vstupe model dostáva binárny reťazec dĺžky  $n$  a musí určiť, či je počet jednotiek páry (výstup = 0) alebo nepáry (výstup = 1). Úspešnosť modelu v probléme parity sa štandardne vyhodnocuje v zmysle konvergencie, čiže počtu sietí, ktoré skonvergujú k riešeniu. Aby sieť konvergovala musí určitý počet epoch po sebe určiť všetky vstupy na 100% správne. Tento nastaviteľný parameter budeme nazývať okno úspechu (po angl. success window). V predošlej práci (Malinovský, L. and Malinová, K., 2022) sme ukázali, že aj základný model s jednou klasickou skrytou vrstvou a jedným produktovým neurónom na výstupe, dosahuje oveľa lepšie výsledky ako klasický MLP pri parite stupňa 2 (XOR) až 7.

V prezentovanom experimente vyhodnocujeme efektívnosť zapojenia produktovej vrstvy pri parite 7. stupňa. Na testovanie použijeme model s 2 skrytými vrstvami a vyskúšame všetky kombinácie zapojenia. Ak architektúra používa produktovú vrstvu, želané výstupy budú preškálované z  $\langle 0, 1 \rangle$  na  $\langle -1, 1 \rangle$  kvôli väčšej efektívnosti pri násobení. Architektúry sú popísané skratkami aktivačných funkcií daných vrstiev. T značí Hyperbolický tangens, Q je naša pseudo-exponenciálna funkcia (quasi), čiže produktová vrstva, S je Sigmoida.

Pre každú architektúru sme menili počet neurónov na skrytej vrstve. Testované veľkosti skrytých vrstiev (h) boli nasledovné: [2, 2], [3, 5], [5, 3], [5, 15], [10, 15], [15, 10], [15, 15], [15, 10], [15, 5], [10, 5], [20, 20], [25, 25], [30, 30], [40, 20], [20, 40], [40, 40]. V Tab 1 sú popísané nastavenia ostatných hyperparametrov.

Výsledky experimentu môžeme vidieť v Tab 1. Pre MLP s aktivačnou funkciou tangens sa nám nepodarilo nájsť také nastavenie aby sieť konvergovala. Tak isto ani pre architektúru číslo 6. Architektúra číslo 4. je v tabuľke dvakrát. Toto zapojenie veľmi dobre konvergovalo pre viacero veľkostí skrytých vrstiev. Pre veľkosť [10,15] síce konvergovalo všetkých 20 sietí, ale

Názov parametra	Hodnota parametra
Počet inštancií siete	20
Rýchlosť učenia	0.5
Okno úspechu	5
Maximum epoch	400 / 1000

**Tab. 1:** Nastavenie fixných hyperparametrov v experimente parity. Počet 1000 maximálnych epoch bol použitý iba pre klasické MLP siete.

pri veľkosti [5,10] konvergujúce siete potrebovali najmenší počet epoch. Až 2 siete dokázali konvergovať len za 6 epoch, čiže pri odrátaní veľkosti okna úspechu (5), sa tieto inštancie naučili problém parity za 1 epochu.

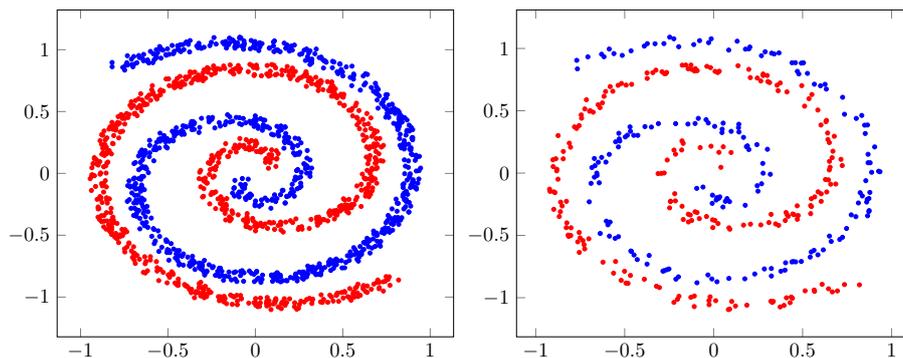
Arch.	Akt. f.	h	konverg.	epochy
1	[T, Q, T]	[15, 5]	18	100.7
2	[T, T, Q]	[20,20]	11	332.5
3	[Q, T, T]	[5,10]	14	148.5
4.1	[Q, Q, T]	[10,15]	20	82.5
4.2	[Q, Q, T]	[5,10]	19	14
5	[Q, T, Q]	[5,5]	20	22.15
6	[T, Q, Q]	[-,-]	0	-
7	[Q, Q, Q]	[2,2]	20	20.55
MLP.1	[S, S, S]	[100,100]	20	710
MLP.2	[T, T, T]	[-,-]	0	-

**Tab. 2:** Tabuľka popisuje koľko z 20 sietí konvergovalo pre danú architektúru, pri najlepšej nami nájdenej veľkosti skrytých vrstiev.

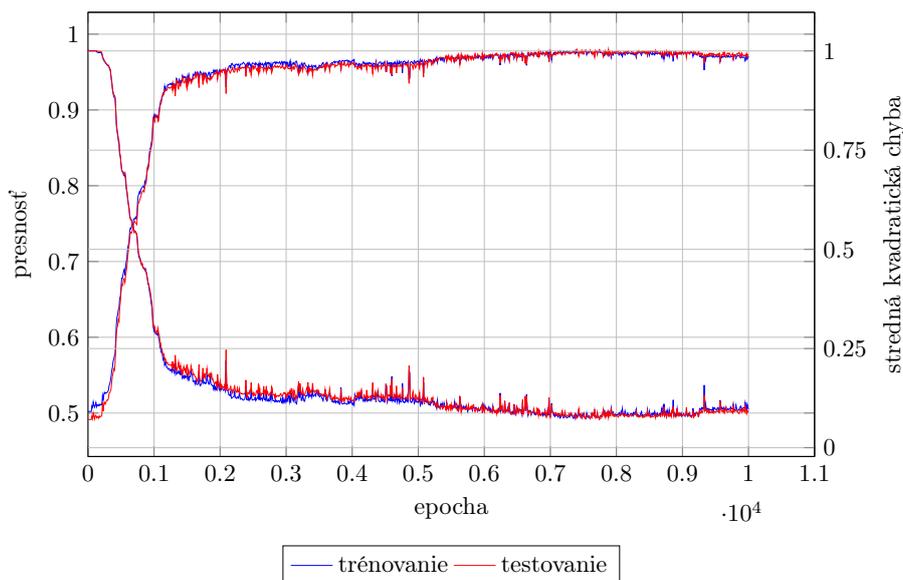
#### 3.2 Problém 2 špirál

Nelineárny problém dvoch do seba zavinených špirál sa taktiež používa na testovanie nových modelov. Na Obr. 1 je zobrazený dataset ktorý pozostáva z 2000 bodov. Želané výstupy boli taktiež preškálované na interval  $\langle -1, 1 \rangle$ .

Pre tento problém sme použili nasledujúcu architektúru: 2 vstupné neuróny, 10 tanh, 80 quasi, 5 tanh, 1 quasi výstupný neurón. Rýchlosť učenia bola nastavená na 0.01, okno úspechu bolo veľké 10 epoch a maximálny počet epoch bol 10000. V experimente sme spustili 10 sietí, z ktorých 4 konvergovali. Priemer dosiahnutej presnosti modelov bol 98.275%.



**Obr. 1:** Zobrazenie datasetu 2 špirál ktorý bol rozdelený v pomere 80:20 - tréning (vľavo) : test (vpravo).



**Obr. 2:** Priemer priebehu trénovania 10 sietí z experimentu 2 špirál.

## 4 Záver

V našej práci sme popísali experimenty s naším novým modelom neurónovej siete s produktovými neurónmi, v ktorých sme skúmali jeho rôzne zapojenia vo viacvrstvovej neurónovej sieti, a to na probléme parity 7. stupňa a dvoch špirál. Výsledky potvrdili, že náš model má značnú výhodu oproti klasickému viacvrstvovému perceptrónu pri problémoch takéhoto typu.

## PodĎakovanie

Tento príspevok vznikol v Centre pre kognitívnu vedu na KAI FMFI UK v Bratislave, s podporou grantu VEGA 1/0373/23 a a KEGA 022UK-4/2023. Za podporu tiež ďakujeme Slovenskej spoločnosti pre kognitívnu vedu SSKV<sup>1</sup>.

<sup>1</sup><https://cogsci.fmph.uniba.sk/sskv/>

## Literatúra

Delalleau, O. a Bengio, Y. (2011). Shallow vs. deep sum-product networks. *Advances in Neural Information Processing Systems*, 24.

Diba, A., Sharma, V. a Van Gool, L. (2017). Deep temporal linear encoding networks. V *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, str. 2329–2338.

Ghosh, J. a Shin, Y. (1992). Efficient higher-order neural networks for classification and function approximation. *International Journal of Neural Systems*, 3(04):323–350.

Malinovský, Ľ. and Malinovská, K. (2022). Neurónová sieť s násobiacou vrstvou. Šejnová, G., Vavrečka a M., Hvorecký, J. (zost.), V *Kognice a umělý život XX*, str. 79–83. České vysoké učení technické v Praze.

Nilsson, N. J. (1965). *Learning machines*. McGrawHill New York.

- Poon, H. a Domingos, P. (2011). Sum-product networks: A new deep architecture. V *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, str. 689–690. IEEE.
- Rumelhart, D. E., Hinton, G. E. a McClelland, J. L. (1986). *A general framework for parallel distributed processing*, vol. 1, str. 26. Cambridge, MA: MIT Press.
- Schmitt, M. (2002). On the complexity of computing and learning with multiplicative neural networks. *Neural Computation*, 14(2):241–301.
- Zhu, J., Zeng, H., Du, Y., Lei, Z., Zheng, L. a Cai, C. (2018). Joint feature and similarity deep learning for vehicle re-identification. *IEEE Access*, 6:43724–43731.