SLĒPTO NEIRONU LOMA TIEŠĀS IZPLATĪBAS TĪKLOS

Pēteris Grabusts
Rēzeknes Augstskola
90 Atbrīvošanas aleja, Rēzekne LV-4600, Latvija
Tālrunis: 4623798 Fax: 4623709
E-mail: peter@cs.ru.lv

Anotācija

Parādīta slēpto neironu loma tiešās izplatības mākslīgajos neironu tīklos. Slēpto neironu skaita izteiksme parasti tiek noteikta katrā atsevišķā gadījumā empīriski. Aprakstītas metodikas slēpto neironu skaita noteikšanai.

1. Ievads

Literatūrā par mākslīgajiem neironu tīkliem bieži sastopams jēdziens - "slēptais" (***hidden)*** neirons vai arī slēptais slānis. Parasti tiek teikts, ka tīkla apmācībā ar sarežģītiem ieejas datiem nepieciešami slēptie neironi. Patiešām, slēpto neironu loma lielākajai daļai interesentu nav skaidri saprotama. Arī speciālajā literatūrā to loma un funkcijas aprakstītas minimāli. Tiek dotas ,piem., šāda tipa definīcijas : "Starp šiem slāņiem ir viens vai vairāki starpslāņi (slēptie slāņi), kuri pilda zināšanu uzkrājēja funkcijas " [1].

Ja formāli neironu tīklu traktējam kā sastāvošu no ieejas slāņa un izejas slāņa, tad slēptajos slāņos faktiski tiek realizēta konkrēto apmācības algoritmu darbība.

Rakstā dots mēģinājums paskaidrot slēpto neironu lomu, izskaidrot situāciju, kad rodas nepieciešamība pēc slēptajiem neironiem un aprakstītas metodes, kas ļauj izvēlēties optimālo slēpto neironu skaitu, kā arī ļauj noskaidrot, vai strādājot ar konkrētajiem ieejas datiem - apmācības procesā būs nepieciešami slēptie neironi vai nē.

2. Priekšvēsture

2.1. Perceptrons

1950. - to gadu beigās Frenks Rozenblats Kornvelas Aeronautikas laboratorijā izstrādāja acs tīklenes darbības modeli, ko nosauca par **perceptronu**. Tas tika radīts, lai izskaidrotu un modelētu vizuālo tēlu pazīšanas mehānismu un iespējas t.i. tēlu atpazīšanai. Perceptrons, tāpat kā **Adaline**, tiek pamatoti uzskatīti par mākslīgo neironu tīklu pamatelementiem. Pats būtiskākais bija tas, ka perceptronu var apmācīt, lai tas ieejas tēlus varētu sadalīt divās klasēs. Turpmāk izmantosim sekojošus apzīmējumus :

Perceptrona elementam (skat. Zīm. 1.) ir sekojoša struktūra :

1) Ir **n** ieejas kanāli un viens izejas kanāls;

2) Katram ieejas kanālam $\mathbf{x_i}$ atbilstībā tiek piešķirts skaitlis $\mathbf{w_i}$, ko sauc par šī kanāla svaru. Svaru vektors ir $\mathbf{w=\{w_1, w_2, \ldots\ldots w_n\}}$.
3) Visu ieeju svērtie signāli tiek summēti :

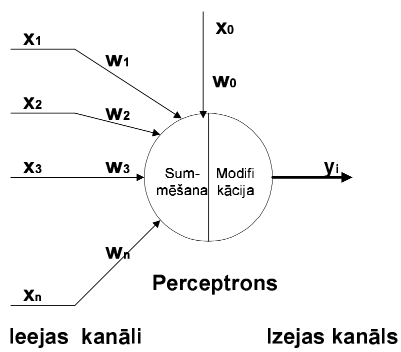
$$a=\sum_i w_i x_i$$

4) Summārais signāls tiek modificēts ar pārvades jeb aktivācijas funkciju un padots uz izejas kanālu.

Kā pārvades funkcijas var tikt izmantotas :

sliekšņa funkcija $y(a)=\theta(a)=\begin{cases} +1 & a>0 \\ -1 & a\le 0 \end{cases}$

sigmoidālā(loģistiskā) $y(a)=\dfrac{1}{1+e^{-a}}$ $y\in(0,1)$



Zīm. 1. Perceptrona elements

Perceptrona apmācība balstās uz svaru korekciju. Apmācības algoritms ir samērā vienkāršs un to var atrast, piemēram, [1],[2].

2.2. XOR problēma

Kā viena prasībām, kas parasti tiek pieprasīta no neironu tīklu elementiem, ir to spēja strādāt ar loģiskajām funkcijām **AND, OR, XOR.**

AND

x_1	x_2	t
0	0	0
0	1	0
1	0	0
1	1	1

OR

x_1	x_2	t
0	0	0
0	1	1
1	0	1
1	1	1

XOR

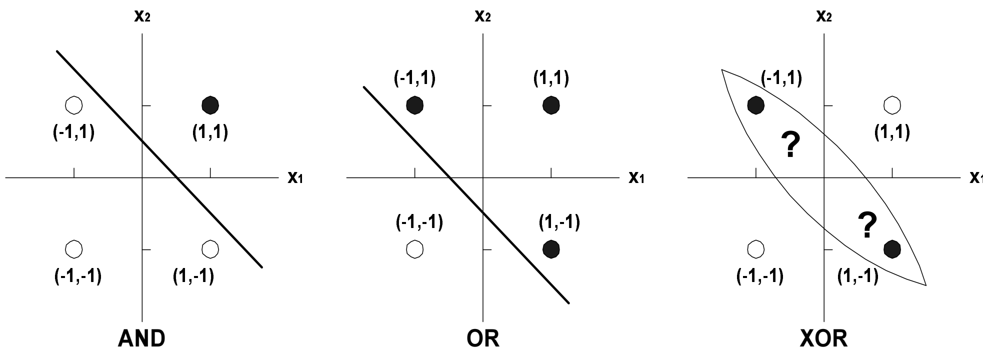
x_1	x_2	t
0	0	0
0	1	1
1	0	1
1	1	0

Rozenblats perceptrona pētījumos parādīja, ka perceptrons spēj pareizi klasificēt divas lineāri atdalāmas klases. Lineārā atdalāmība nozīmē to, ka divu klašu objekti var tikt atdalīti ar taisni. Priekš perceptrona (vienkāršākajā gadījumā - ar divām ieejām un vienu izeju) - robeža starp klasēm tiek noteikta ar taisnes vienādojumu :

$$\mathbf{w_1x_1 + w_2x_2 + w_0 = 0}$$

Klases, kuras šī taisne atdala, vienkāršības dēļ apzīmēsim ar **+1** un **-1**. No Zīm.2. redzams, ka loģiskajām funkcijām **AND** un **OR** pastāv taisne, kas atdala klases **+1** un

-1, bet priekð **XOR** tāda taisne neeksistē,t.i. perceptrons var realizēt loģiskās funkcijas AND un OR, bet nevar XOR.

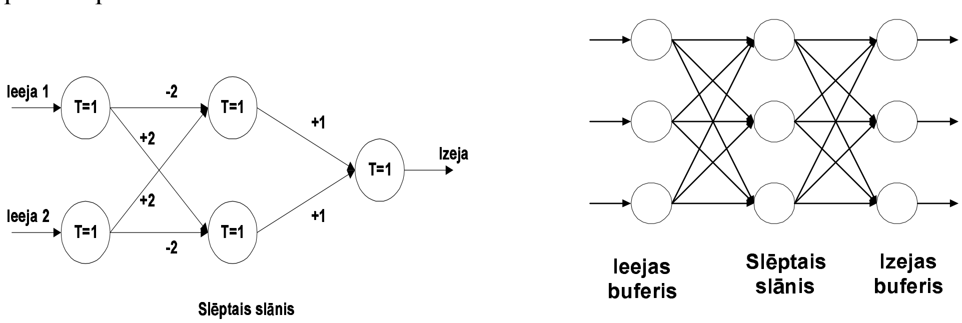


Zīm. 2. Loģisko funkciju klašu atdalāmība

2.3. Daudzslāņu perceptrons

Izrādās, ka XOR problēmu var atrisināt, ievedot papildus neironus tā sauktajā slēptajā slānī, ar to uzsverot, ka šādi neironi tieši neiespaido ieejas un izejas datu izskatu. Šādus neironus arī nosauca par neredzamajiem jeb slēptajiem neironiem.

XOR atrisinājums parādīts Zīm.3a, kur **T** - sliekšņa vērtība. Šādu neironu tīklu nosauca par d*audzslāņu perceptronu,* kas ir viens no populārākajiem tīkliem lietotāju vidū. Katrs neirons aktivizējas ar sigmoidālās vai sliekšņa funkcijas palīdzību. Apmācībā var izmantot kļūdas atgriezeniskās izplatības algoritmu, Adaline vai Madaline algoritmus. Veicamie tīkla uzdevumi - tēlu klasifikācija, funkciju aproksimācija, prognozēšana, vadība. Vispārīgā veidā daudzslāņu perceptrons parādīts zīm. 3b.



a) XOR atrisinājuma piemērs b) daudzslāņu perceptrona shēma

Zīm. 3. Daudzslāņu perceptrons

Tādējādi - daudzslāņu perceptrona tīklā var būt vairāki slēptie slāņi ar slēptajiem neironiem. Lietojot šādus tīklus, piemēram, klasifikācijas uzdevumos, parādās sekojoša problēma - **cik daudz šādu slēpto elementu vajag katrā slānī ?**

Patlaban tiešs atrisinājums nav zināms. Praktiski to dara ar kļūdu un mēģinājumu metodi. Literatūrā ir atrodamas dažas izteiksmes, kas ļauj novērtēt slēpto neironu skaita augšējo robežu.

3. Cik slēpto neironu vajag ?

3.1. Teorētiskie novērtējumi

Dažādi autori ir devuši dažādus slēpto neironu skaita novērtējumus. Visizplatītākais novērtējums ir šāds : Ja ir **N** ieejas elementi, tad kopējais ieejas vektoru skaits ir $\mathbf{2^N}$. Tad maksimālais slēpto neironu skaits ir $\mathbf{2^{N-1}}$.

Darba [3] autori piedāvā šādu novērtējumu (zināms kā Bauma-Hausslera likums) :

$$N_{hidden} \le \frac{N_{train} E_{tolerance}}{N_{pts} + N_{output}}, \text{ kur}$$

N_{hidden} - slēpto neironu skaits;

N_{train} - apmācības paraugu skaits;
N_{pts} - elementu daudzums apmācības piemērā;
N_{output} - izejas neironu skaits;
$E_{tolerance}$ - kļūdas pielaide.

3.2. Praktiskas rekomendācijas

Vairāki autori, to starpā pazīstamās monogrāfijas [4] autors, dod šādas, šķietami vienkāršas, praktiskas rekomendācijas tiešās izplatības tīklu realizācijai :

a) Izmantojiet vienu slēpto slāni;
b) Izmantojiet pēc iespējas mazāk slēpto neironu;

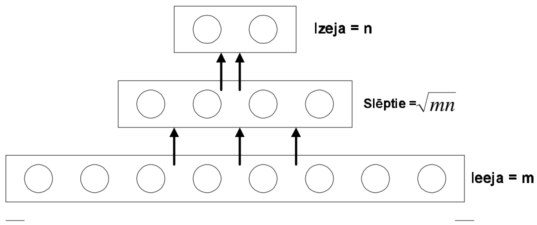
<u>Cik daudz slēpto slāņu ?</u>

Nav iemesla lietot vairāk nekā 1 slēpto slāni -jo citādi tīkla apmācība var noritēt ļoti lēni. Protams, vienmēr pastāv iespēja, ka vairāki slāņi lieliski atrisina komplicētāku problēmu. Tāpēc tiešās izplatības tīklu praktiskajā realizācijā pirmajā piegājienā tiek ieteikts lietot vienu slēpto slāni. Ja liels skaits slēpto neironu pirmajā slānī nedod pietiekoši labu problēmas risinājumu, ir vērts pamēģināt izmantot otru slēpto slāni, samazinot kopējo slēpto neironu kopskaitu.

<u>Cik daudz slēpto neironu ?</u>

Nepieciešamība izvēlēties pareizo slēpto neironu skaitu ir ļoti būtiska.Lietojot pārāk maz- tīklam pietrūks resursu apmācības algoritma realizācijai. Lietojot pārāk daudz- palielināsies apmācības laiks, varbūt pat neiespējami būs to adekvāti apmācīt saprātīgā laika periodā. Tātad, jācenšas izmantot absolūti minimālu slēpto neironu skaitu, kas atrisina problēmu.

Lielākajai daļai uzdevumu ir pielietojams *ģeometriskās piramīdas likums* . Tas nosaka, ka neironu skaits jāsamazina no ieejas uz izeju, tā- kā tas redzams Zīm.4.



Zīm. 4. Tipisks trīsslāņu tīkls 8-4-2

Līdzīgi varētu uzdot četrslāņu tīklu, piem. 8-4-2-1. Slēpto neironu skaits slāņos HID_1 un HID_2 jāizvēlas pēc sekojošām sakarībām:

$$\mathbf{HID_1 = nr^2}$$

$$\mathbf{HID_2 = nr}\text{ , kur }\quad \mathbf{r}= \sqrt[3]{\frac{m}{n}}$$

Augstāk uzrakstītās formulas dod tikai aptuvenu novērtējumu. Praksē katram konkrētam uzdevumam nepieciešamais slēpto neironu skaits tiek noteikts eksperimentāli.

3.3. Empīriska slēpto neironu noteikšana

Rakstā [5] autors Kinser J. apraksta metodiku, kas ir interesanta ar to, ka piedāvā noteikt nepieciešamību pēc slēptajiem neironiem *pirms apmācības* t.i. tikai analizējot ieejas datus. Tā nosaukta par Slēptā neirona teoriju un apraksta procedūru, kas iegūst no ieejas datien visu informāciju , kas nepieciešama tīkla arhitektūras noteikšanai pirms apmācības procesa. Var tikt noteikts iespējamo slēpto neironu skaits, to specifiskā loma un to stāvoklis visiem apmācāmās kopas datu pāriem. Metode ļauj atrast tīkla arhitektūru ar minimālu slēpto neironu skaitu.

Idejas būtība ir sekojoša. Slēpto neironu skaits ir tieši atkarīgs no apmācošo datu iekšējās struktūras un to būtība ir atrisināt **konfliktus** ieejas datos - ja ieejas dati satur konfliktus, tad neironu tīklam vajadzīgi slēptie neironi.

Paskaidrosim to uz XOR piemēra. XOR problēmā ir 4 apmācības elementi. Kā aktivācijas funkcija tiek izmantota sliekšņa funkcija :

$$y(a)=\begin{cases} 1 & ja \quad a > \gamma \\ 0 & citadi \end{cases} \qquad \text{, kur } 0 < \gamma < 0$$

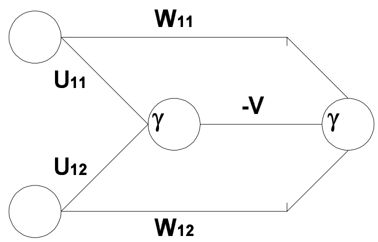
Apmācības datu pāriem **(x:y)** ir spēkā šādas nevienādības :

$$\sum_i w_{ji} x_i \begin{cases} > \gamma & , ja \;\; y = 1 \\ < \gamma & , ja \;\; y = 0 \end{cases}$$

Tādējādi XOR problēma pārrakstās ar nevienādību virkni :

$$\begin{array}{lll} 0 & + \; 0 & < \gamma \\ 0 & + W_{12} & \geq \gamma \\ W_{11} & + \; 0 & \geq \gamma \\ W_{11} & + W_{12} & < \gamma \end{array}$$

Viegli redzēt, ka gadījumā, ja izpildās $W_{12} \geq \gamma$ un $W_{11} \geq \gamma$, tad nevienādība $W_{11} + W_{12} < \gamma$ nevar izpildīties. Saka, ka nevienādības ir konfliktā. Šāda konflikta atrisinājums arī ir slēpto neironu ievešana, kā redzams Zīm. 3a. un 5.



Zīm. 5. XOR problēmas atrisinājuma piemērs

Attiecīgās nevienādību sistēmas būs sekojošas :

$$
\begin{array}{lllll}
0 + 0 < \gamma & & 0 + 0 + 0 < \gamma \\
0 + U_{12} < \gamma & \Rightarrow & 0 + W_{12} + 0 \geq \gamma \\
U_{11} + 0 < \gamma & & W_{11} + 0 + 0 \geq \gamma \\
U_{11} + U_{12} \geq \gamma & & W_{11} + W_{12} + V < \gamma
\end{array}
$$

Tas nozīmē, ka konflikts ir atrisināts, ieviešot papildus neironu.

Meklējot konfliktus lielās apmācību kopās šīs metodes vājā vieta, diemžēl, ir nevienādību sistēmu rēķināšanā.

Nobeigums

Strādājot ar tiešās izplatības tīklu arhitektūras modeļiem un lietojot atgriezeniskās izplatības apmācības algoritmu, nākas saskarties ar slēpto neironu skaita noteikðanas problēmu. Rakstā dotas dažas slēpto neironu skaita izteiksmes un praktiski ieteikumi. Pēdējā nodaļā ieskicēta metodika, kas analizē neironu tīkla ieejas datus un pārbauda situācijas, kad ir nepieciešami slēptie neironi. Tas varētu dot iespēju apzināt slēpto neironu skaitu pirms tīkla apmācības, kas ļautu veidot optimālu tīklu. Metode der tikai tiešas izplatības neironu tīkla modelim.

Ieteicamā literatūra

[1] Mākslīgie neironu tīkli: arhitektūra, algoritmi un pielietojumi.(1998).*Mācību līdzeklis. Rīga.*

[2] Alexander, I., Morton, H. (1991) An Introduction to Neural Computing. *Chapman & Hall, London.*

[3] Baum, E.B., Haussler, D. (1988) What size net gives valid generalization. *Neural Computation, 1, pp. 151-160.*

[4] Masters, T. (1993) Practical Neural Network recipes in C++.*Academic Press.*

[5] Kinser J.M.,(1996) The determination of Hidden Neurons. *Optical Memories and Neural Networks, 5(4), 245-262.*