

# Inteligentă artificială - laborator

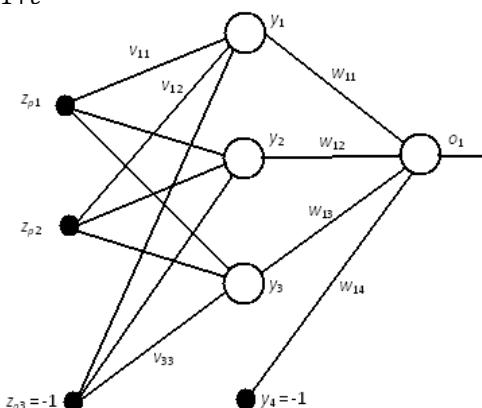
## Algoritmul “Backpropagation of error”

Algoritmul de instruire *backpropagation of error* implementează *regula delta generalizată* care este o metodă de instruire supervizată a unei rețele multistrat de perceptri de tip continuu. Spre deosebire de regula delta care funcționează pentru pattern-uri liniar separabile, regula delta generalizată se poate aplica și pattern-urilor neseparabile liniar. Algoritmul *backpropagation of error* a fost propus de D.E. Rumelhart, G.E. Hinton și R.J. Williams într-un articol apărut în 8 octombrie 1986 în revista *Nature*. O copie a articolului se găsește la <http://www.cs.toronto.edu/~hinton/absps/naturebp.pdf>.

În figura de mai jos avem o rețea care folosește:

- La intrare pattern-uri cu două componente la care se adaugă o intrare suplimentară cu valoarea constantă -1. Notăm cu  $z_{p1}$ ,  $z_{p2}$  și  $z_{p3}$  intrările rețelei neurale, unde  $p$  este indicele pattern-ului curent de instruire extras din setul de  $n$  pattern-uri de instruire;
- Un strat ascuns cu 3 perceptri de tip continuu ale căror ieșiri devin intrările neuronului de pe stratul de ieșire. Ieșirile neuronilor ascunși sunt notate cu  $y_1$ ,  $y_2$  și  $y_3$ . Legăturile dintre intrări și neuronii ascunși sunt notate cu  $v_{11}$ ,  $v_{12}$  ...  $v_{33}$ . În exemplul nostru sunt 3x3 astfel de conexiuni.
- Un strat de ieșire format dintr-un singur perceptron continuu. Intrările sale sunt  $y_1$ ,  $y_2$ ,  $y_3$  și intrarea suplimentară  $y_4=-1$ . Legăturile dintre neuronii ascunși și neuronul de ieșire sunt notate cu  $w_{11}$ ,  $w_{12}$  și  $w_{13}$ , la care se adaugă  $w_{14}$ . Ieșirea rețelei este notată cu  $o_1$ .

Funcția de activare a perceptriilor este continuă. Vom folosi varianta continuă bipolară care are formula  $f(\text{net}) = \frac{2}{1+e^{-\text{net}}} - 1$ .



Algoritmul *backpropagation of error* se poate implementa astfel:

1. Se citesc pattern-urile de instruire  $\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_n$  și ieșirile dorite  $\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_n$ . Pattern-urile sunt extinse cu o componentă suplimentară cu valoarea -1. Se stabilește numărul de intrări și numărul de perceptri. Se inițializează aleator ponderile  $\mathbf{v}_{ji}$  și  $\mathbf{w}_{kj}$ . Se inițializează constanta de instruire  $c$ . Pentru a accelera instruirea puteți să inițializați constanta de instruire  $c$  cu o valoare mare, de exemplu  $c=50$ , și să o scădeți treptat. Se stabilește eroarea maximă  $E_{max}$ . Se inițializează  $E=0$ .
2. Având un pattern  $\mathbf{z}_p$  dintre cele  $n$ , se calculează ieșirile fiecărui perceptron:

$$y_j = f(v_{j1}z_{p1} + v_{j2}z_{p2} + \dots + v_{j3}z_{p3}), \quad j=1 \dots J,$$

# Inteligentă artificială - laborator

$$o_k = f(w_{k1}y_1 + w_{k2}y_2 + \dots + w_{kj}y_j), \quad k=1 \dots K,$$

unde  $I$  este numărul de intrări,  $J$  este numărul de perceptri ascunși,  $K$  este numărul de perceptri de ieșire, iar  $f$  este funcția de activare bipolară continuă.

3. Se calculează semnalele de eroare:

$$\delta_j = 0.5(1-y_j^2)(\delta_{o1}w_{1j} + \dots + \delta_{ok}w_{kj})$$

$$\delta_{ok} = 0.5(d_j - o_j)(1 - o_j^2)$$

4. Se actualizează toate ponderile rețelei conform regulii:

$$v_{ji}^{(t+1)} = v_{ji}^{(t)} + c\delta_{ij}z_{pi}$$

$$w_{kj}^{(t+1)} = w_{kj}^{(t)} + c\delta_{ok}y_j$$

5. Se calculează eroarea cumulată:

$$E^{(t+1)} = E^{(t)} + 0.5(d_1 - o_1)^2 + 0.5(d_2 - o_2)^2 + \dots + 0.5(d_K - o_K)^2.$$

6. Se revine la pasul 2 cu un nou vector de instruire, până la epuizarea setului de  $n$  pattern-uri.

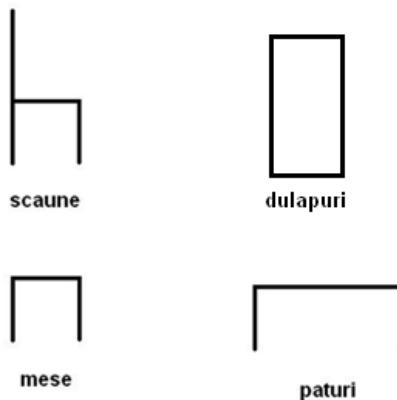
7. Dacă  $E > E_{max}$ , se revine la pasul 2 pentru o nouă epocă de instruire în care se reiau toate pattern-urile de instruire și se reinicializează  $E=0$ .

Implementați *regula delta generalizată* pentru o rețea cu două straturi de perceptri, ca în figura de mai sus. Rețeaua folosește pattern-uri bidimensionale cărora li se adaugă a treia intrare -1,  $\mathbf{z}_1, \mathbf{z}_2 \dots \mathbf{z}_{12}$  cu ieșirile dorite  $\mathbf{d}_1, \mathbf{d}_2 \dots \mathbf{d}_{12}$ :

45	85	1	48	40	-1
50	43	-1	195	41	1
40	80	1	43	87	1
187	107	-1	192	105	-1
55	42	-1	190	40	1
200	43	1	188	100	-1

Acste pattern-uri reprezintă lățimea și înălțimea unor obiecte de mobilier asemănătoare celor din imaginea de mai jos. Obiecte sunt depozitate în două incinte separate, scaunele și paturile într-o, iar mesele și dulapurile în cealaltă. Obiectele trebuie clasificate automat astfel:

- Scaunele și dulapurile în clasa 1;
- Mesele și paturile în clasa -1.



La final afișați pattern-urile de instruire și răspunsul rețelei pentru fiecare pattern.