

KONVOLUCIONA NEURALNA MREŽA. Ovdje ćemo prikazati postupak računanja na jednostavnom primjeru dvoslojne neuralne mreže (jedan konvolucioni sloj i jedan potpuno povezani sloj). Ulazni signal sa $N = 8$ uzoraka može sadržavati varijantu signala trouglastog oblika, $feature_1 = [-0.5, 1, -0.5] + v(n)$, ($\mathbf{t} = [0, 1]^T$) ili može sadržavati varijantu ravnog signala od tri uzorka $feature_2 = [1, 1, 1] + v(n)$, $\mathbf{t} = [1, 0]^T$, gdje je $v(n)$ slučajni uniformni šum od 0 do 0.3, koja uvodi odstupanja u oblicima. Signal je zatim umetnut u aditivni slučajni Gausov šum sa standardnom devijacijom od 0.05, i potom normalizovan na jediničnu energiju. Koriste se konvolucionni filteri dužine $M = 3$ uzorka, kako bi se proizveli $K = 3$ kanala. Na izlazu se koristi Softmax funkcija, sa dvije vrijednosti koje bi, nakon procesa treniranja, trebalo da odgovaraju ciljanoj signalu, $\mathbf{t}$. Mreža se trenira sa 200 realizacija signala koje se koriste 10 puta (u 10 epoha). Nakon treniranja mreže, testira se na 100 novih slučajnih realizacija signala.

Propagacija unaprijed: Od ulaznog do izlaznog signala
•Ulaz signal, $\mathbf{x}$, dužine $N = 8$. Ciljni signal $\mathbf{t} = [1 \ 0]^T$, jer je $feature_2$ prisutan. $\mathbf{x} = [-0.18 \ -0.28 \ -0.23 \ -0.32 \ 0.45 \ 0.45 \ 0.45 \ -0.35]^T$
•Inicijalne težine, slučajno $w_k^1(m) \sim \mathcal{N}(0,1)\sqrt{2/3}$, $M = 3$ za $K = 3$ kanala: $\mathbf{w}_1^1 = [-0.07 \ -0.01 \ -1.47]^T$, $\mathbf{w}_2^1 = [0.44 \ 0.14 \ -0.30]^T$, $\mathbf{w}_3^1 = [1.15 \ -1.01 \ -1.83]^T$
•Konvolucija, $\mathbf{y}_k = \mathbf{w}_k^1 * \mathbf{x} + b_k$, $k = 1, 2, 3$ sa inicijalnim biasom $b_1 = 0$, $b_2 = 0$, i $b_3 = 0$. $\mathbf{y}_1 = [0.35 \ 0.49 \ -0.65 \ -0.65 \ -0.69 \ 0.48]^T$ $\mathbf{y}_2 = [-0.05 \ -0.06 \ -0.28 \ -0.21 \ 0.13 \ 0.37]^T$ $\mathbf{y}_3 = [0.48 \ 0.50 \ -0.77 \ -1.66 \ -0.76 \ 0.71]^T$
•ReLU aktivaciona funkcija, $\mathbf{F}_k = f(\mathbf{y}_k^1) = \max\{0, \mathbf{y}_k^1\}$. $f(\mathbf{y}_1) = [0.35 \ 0.49 \ \mathbf{0.00} \ \mathbf{0.00} \ \mathbf{0.00} \ 0.48]^T$ $f(\mathbf{y}_2) = [\mathbf{0.00} \ \mathbf{0.00} \ \mathbf{0.00} \ \mathbf{0.00} \ 0.13 \ 0.37]^T$ $f(\mathbf{y}_3) = [0.48 \ 0.50 \ \mathbf{0.00} \ \mathbf{0.00} \ \mathbf{0.00} \ 0.71]^T$
•Max-pooling, $o_k^1(m) = \max\{F_k(mP), \dots, F_k(mP + P - 1)\}$, sa $P = 3$, $\mathbf{o}^1 = \begin{bmatrix} \max\{0.35 & 0.49 & \mathbf{0.00}\} & \max\{\mathbf{0.00} & \mathbf{0.00} & \mathbf{0.00}\} \\ \max\{\mathbf{0.00} & \mathbf{0.00} & \mathbf{0.00}\} & \max\{\mathbf{0.00} & 0.13 & 0.37\} \\ \max\{0.48 & 0.50 & \mathbf{0.00}\} & \max\{\mathbf{0.00} & \mathbf{0.00} & 0.71\} \end{bmatrix}^T = \begin{bmatrix} 0.49 & 0.48 \\ \mathbf{0.00} & 0.37 \\ 0.50 & 0.71 \end{bmatrix}^T$, Indikator izabranih vrijednosti od ReLU, $\mathbf{M}^{ReLU}$, i max-pooling, $\mathbf{M}^{MP}$, $\mathbf{M}^{ReLU} = \begin{bmatrix} \mathbf{1} & \mathbf{1} & 0 & 0 & 0 & \mathbf{1} \\ \mathbf{0} & 0 & 0 & 0 & \mathbf{1} & \mathbf{1} \\ \mathbf{1} & \mathbf{1} & 0 & 0 & 0 & \mathbf{1} \end{bmatrix}^T$ and $\mathbf{M}^{MP} = \begin{bmatrix} 0 & \mathbf{1} & 0 & 0 & 0 & \mathbf{1} \\ \mathbf{1} & 0 & 0 & 0 & 0 & \mathbf{1} \\ 0 & \mathbf{1} & 0 & 0 & 0 & \mathbf{1} \end{bmatrix}^T$, koji će biti iskorištena u propagaciji unazad.
•Flattening, $N_F = (N - M + 1)P = 2$, $o_F^1((k - 1)N_F + m) = o_k^1(m)$, $k = 1, 2, 3$, $m = 0, 1$. $\mathbf{o}_F^1 = [0.49 \ 0.48 \ 0.00 \ 0.37 \ 0.50 \ 0.71]^T$,
•Inicijalne težine za FC sloj, slučajno $w_k^2(n) \sim \mathcal{N}(0,1)\sqrt{2/6}$ $\mathbf{w}^2 = \begin{bmatrix} -0.47 & -0.06 & -0.05 & -0.81 & 0.62 & -0.18 \\ 0.02 & 0.12 & -0.15 & -0.07 & -0.87 & -0.53 \end{bmatrix}^T$,
•Izlaz iz FC sloja, $\mathbf{y}_k^2 = \sum_{n=0}^5 o_F^1(n) w_k^2(n)$. $\mathbf{y}^2 = (\mathbf{w}^2)^T \mathbf{o}_F^1 = [-0.38 \ -0.77]^T$.
•Softmax, sa $S = 2$, $P_k = e^{y_k^2} / (e^{y_1^2} + e^{y_2^2})$, $k = 1, 2$ $\mathbf{P} = [0.60 \ 0.40]^T$, $\Delta^2 = \mathbf{P} - \mathbf{t} = [-0.40 \ 0.40]^T$, $\Delta_k^2 = P_k - t_k$

Propagacija unazad: Delta error, gradijent, ažuriranje težina
•Gradijent u FC sloju, $g_k^2(m) = \Delta_k^2 o_F^1(m)$, $\mathbf{g}^2 = \mathbf{o}_F^1 (\Delta^2)^T = \begin{bmatrix} -0.12 & -0.12 & -0.00 & -0.09 & -0.12 & -0.17 \\ 0.12 & 0.12 & 0.00 & 0.09 & 0.12 & 0.17 \end{bmatrix}^T,$
•Ažuriranje težina u FC sloju, $w_k^2(m) \leftarrow w_k^2(m) - 0.1 g_k^2(m)$. $\mathbf{w}^2 = \mathbf{w}^2 - 0.1 \mathbf{g}^2 = \begin{bmatrix} -0.45 & -0.04 & -0.05 & -0.79 & 0.64 & -0.15 \\ 0.00 & 0.10 & -0.15 & -0.08 & -0.89 & -0.56 \end{bmatrix}^T,$
•Delta error propagacija unazad u konvolucionom sloju, $\Delta_k^1(m) = \sum_p \Delta_p^2 w_p^2(m)$, $(\Delta^2)^T \mathbf{w}^2 = [0.18, 0.06, \mathbf{-0.04}, 0.29, -0.62, -0.16]$, Izmjena pozicija: $\{(\Delta^2)^T \mathbf{w}^2\} = \begin{bmatrix} 0.18 & 0.06 \\ -0.04 & 0.29 \\ -0.62 & -0.16 \end{bmatrix}$, prateći $\mathbf{o}_F^1 \rightarrow \mathbf{o}^1$
•Izmjena pozicija elemenata $\Delta_k^1 = [(\Delta^2)^T \mathbf{w}^2]$ na pozicijama definisanim sa $\mathbf{M}_k^{MP}] \odot \mathbf{M}_k^{ReLU}$ $\Delta_1^1 = [0 \ 0.18 \ 0 \ 0 \ 0 \ 0.06]^T$, $\Delta_2^1 = [\mathbf{0} \ 0 \ 0 \ 0 \ 0 \ 0.29]^T$, $\Delta_3^1 = [0 \ -0.62 \ 0 \ 0 \ 0 \ -0.16]^T$ $\odot$ je Hadamardov element-po-element proizvod.
•Gradijent u konvolucionom sloju, $g_k^1(m) = \Delta_k^1(m) *_c x(m)$, $\mathbf{g}_k^1 = \mathbf{x} *_c \Delta_k^1$ $\mathbf{g}_1^1 = [-0.03 \ 0.03 \ 0.06]^T,$ $\mathbf{g}_2^1 = [0.01 \ 0.20 \ 0.12]^T,$ $\mathbf{g}_3^1 = [0.06 \ -0.02 \ 0.06]^T$
•Ažuriranje težina u konvolucionom sloju $w_k^1(m) \leftarrow w_k^1(m) - 0.1 g_k^1(m)$ $\mathbf{w}_1^1 = [-0.06 \ -0.01 \ -1.47]^T$ $\mathbf{w}_2^1 = [0.45 \ 0.13 \ -0.31]^T,$ $\mathbf{w}_3^1 = [1.12 \ -1.01 \ -1.84]^T$
•Ažuriranje biasa, $b_k \leftarrow b_k - 0.05 \sum_n \Delta_k^1(n)$ $\mathbf{b} \leftarrow \mathbf{b} - 0.05 ([1 \ 1 \ 1 \ 1 \ 1 \ 1] \Delta^1)^T = \mathbf{0} - 0.05 [0.24 \ 0.29 \ -0.78]^T.$
•Nova iteracija sa novim signalom, $\mathbf{t} = [0, \ 1]$, $\mathbf{x} = [-0.10 \ -0.12 \ -0.05 \ -0.24 \ 0.89 \ -0.35 \ -0.02 \ -0.00]^T,$ Vraćanje na prvi korak sa novim (ažuriranim) težinama, $\mathbf{w}^1$ i $\mathbf{w}^2$, i bias $\mathbf{b}$.