Rozważamy sieć neuronową, która ma L gęstych warstw, po $n^{[l]}$ neuronów w warstwie $l = 1, \dots L$, i która przyjmuje wektor z $\mathbf{R}^{n^{[0]}}$. Dla uproszczenia notacji zakładamy, że we wszyskich warstwach używamy takiej samej funkcji aktywacji $\varphi : \mathbf{R} \to \mathbf{R}$.

Niech $W^{[l]}$ będzie macierzą wag w warstwie l, tj.

$$W^{[l]} = [w^{[l]}_{k,j}]_{k=1,\dots,n^{[l]};j=0,\dots,n^{[l-1]}}, \quad l = 1, 2, \dots, L.$$

Powiedzmy, że argumentem sieci jest wektor

$$a^{[0]} = \begin{bmatrix} a^{[0]}_1 \\ a^{[0]}_2 \\ \dots \\ a^{[0]}_{n^{[0]}} \end{bmatrix} \in \mathbf{R}^{n^{[0]}}.$$

Propagacja w przód (obliczanie wartości), krok indukcyjny. Powiedzmy, że $l \in \{0, 1, \dots, L-1\}$ oraz że mamy dany wektor

$$a^{[l]} = \begin{bmatrix} a^{[l]}_1 \\ a^{[l]}_2 \\ \dots \\ a^{[l]}_{n^{[l]}} \end{bmatrix}.$$

z $\mathbf{R}^{n^{[l]}}$. Kładziemy $x^{[l]}_0 = 0$ oraz $x^{[l]}_j = a^{[l]}_j$ dla $j \geq 1$, czyli

$$x^{[l]} = \begin{bmatrix} x^{[l]}_0 \\ x^{[l]}_1 \\ x^{[l]}_2 \\ \dots \\ x^{[l]}_{n^{[l]}} \end{bmatrix} = \begin{bmatrix} 1 \\ a^{[l]}_1 \\ a^{[l]}_2 \\ \dots \\ a^{[l]}_{n^{[l]}} \end{bmatrix}.$$

Wektor $x^{[l]}$ jest więc wektorem $a^{[l]}$ z dołączoną jedynką na początku. Obliczamy

$$net^{[l+1]} = W^{[l+1]} x^{[l]},$$

oraz

$$a^{[l+1]} = \varphi(net^{[l+1]}) := \left[\varphi(\sum_{j=0}^{n^{[l]}} w^{[l+1]}_{k,j} x^{[l]}_j) \right]_{k=1,\dots,n^{[l+1]}}.$$

Propagacja w przód (obliczanie wartości), wyjście sieci. Mając dany wektor $a^{[0]}$, możemy stosować L-krotnie powyższy krok indukcyjny, aby obliczyć kolejno $a^{[1]}, a^{[2]}, \dots, a^{[L]}$. Wyjściem sieci jest ostatni wektor $a^{[L]}$. Innymi słowy, rozważana sieć neuronowa jest funkcją następującej postaci

$$\mathbf{R}^{n^{[0]}} \ni a^{[0]} \mapsto a^{[L]} \in \mathbf{R}^{n^{[L]}}.$$

Funkcja kosztu. Powiedzmy, że dla $x^{[0]}$ obliczyliśmy $a^{[L]}$ jak wyżej, jednak spodziewaliśmy się otrzymać inny wektor, $y \in \mathbf{R}^{n^{[L]}}$. Zmodyfikujemy wagi za pomocą algorytmu *gradient descent*, licząc gradient funkcji kosztu względem wag.

Przyjmijmy, że nasza funkcja kosztu ma postać

$$\mathbf{L}(y, a^{[L]}) = \frac{1}{2} \sum_{k=1}^{n^{[L]}} (y_k - a^{[L]}_k)^2.$$

Najpierw obliczymy

$$\frac{\partial \mathbf{L}}{\partial a_k^{[L]}} = a_k^{[L]} - y_k, \qquad k = 1, \ldots, n^{[L]}.$$

Powyższe równości będziemy zapisywać w skrócie tak:

$$\frac{\partial \mathbf{L}}{\partial a^{[L]}} = \left[a_k^{[L]} - y_k \right]_{k=1,\ldots,n^{[L]}}, \tag{0.1}$$

po obu stronach mamy wektor kolumnowy.

Propagacja wstecz, krok indukcyjny. Powiedzmy, że $l \in \{1, \ldots, L\}$ oraz że mamy dany wektor

$$\frac{\partial \mathbf{L}}{\partial a^{[l]}}.$$

Przypomnijmy, że zachodzi wzór

$$a^{[l]} = \left[\varphi\Big(\sum_{j=0}^{n^{[l-1]}} w_{k,j}^{[l]} x_j^{[l-1]} \Big) \right]_{k=1,\ldots,n^{[l]}}.$$

Możemy więc obliczyć $\frac{\partial \mathbf{L}}{\partial w_{k_0,j_0}^{[l]}}$ używając reguły łańcucha

$$\begin{aligned} \frac{\partial \mathbf{L}}{\partial w_{k_0,j_0}^{[l]}} &= \sum_{k=1,\ldots,n^{[l]}} \frac{\partial \mathbf{L}}{\partial a_k^{[l]}} \cdot \frac{\partial a_k^{[l]}}{\partial w_{k_0,j_0}^{[l]}} \\ &= \frac{\partial \mathbf{L}}{\partial a_{k_0}^{[l]}} \, \varphi' \left(\sum_{j=0}^{n^{[l-1]}} w_{k_0,j}^{[l]} x_j^{[l-1]} \right) \, x_{j_0}^{[l-1]} =: \delta_{k_0}^{[l]} x_{j_0}^{[l-1]}, \end{aligned}$$

gdzie wektor kolumnowy $\delta^{[l]}$ jest określony następująco

$$\delta^{[l]} = \left[\frac{\partial \mathbf{L}}{\partial a_{k_0}^{[l]}} \, \varphi' \left(\sum_{j=0}^{n^{[l-1]}} w_{k_0,j}^{[l]} x_j^{[l-1]} \right) \right]_{k=1,\ldots,n^{[l]}}.$$

Taka notacja pozwala zapisać w skrócie otrzymane wyżej wzory na pochodne,

$$\frac{\partial \mathbf{L}}{\partial w^{[l]}} = \delta^{[l]} \cdot (x^{[l-1]})^T,$$

gdzie po obu stronach mamy macierze o $n^{[l]}$ wierszach i $(n^{[l-1]} + 1)$ kolumnach.

Podobnie możemy też obliczyć $\frac{\partial \mathbf{L}}{\partial x_{j_0}^{[l-1]}}$ używając reguły łańcucha

$$\begin{aligned} \frac{\partial \mathbf{L}}{\partial x_{j_0}^{[l-1]}} &= \sum_{k=1}^{n^{[l]}} \frac{\partial \mathbf{L}}{\partial a_k^{[l]}} \cdot \frac{\partial a_k^{[l]}}{\partial x_{j_0}^{[l-1]}} \\ &= \sum_{k=1}^{n^{[l]}} \frac{\partial \mathbf{L}}{\partial a_k^{[l]}} \varphi' \left(\sum_{j=0}^{n^{[l-1]}} w_{k,j}^{[l]} x_j^{[l-1]} \right) \, w_{k,j_0}^{[l]} = \sum_{k=1}^{n^{[l]}} w_{k,j_0}^{[l]} \delta_k^{[l]}. \end{aligned}$$

W notacji macierzowej:

$$\frac{\partial \mathbf{L}}{\partial x^{[l-1]}} = (W^{[l]})^T \cdot \delta^{[l]}.$$

Przypomnijmy, że $x_j^{[l-1]} = a_j^{[l-1]}$ dla $j \geq 1$, a więc pomijając pierwszy element powyższego wektora otrzymujemy

$$\frac{\partial \mathbf{L}}{\partial a^{[l-1]}}.$$

Propagacja wstecz, podsumowanie. Korzystając ze wzoru (0.1), możemy zastosować krok indukcyjny dla $l = L$, otrzymamy pochodne funkcji kosztu po wagach z ostatniej warstwy oraz $\frac{\partial \mathbf{L}}{\partial a^{[L-1]}}$. To pozwala zastosować krok indukcyjny dalej, kolejno dla $l = L-1, \ldots, 1$. W ten sposób otrzymamy pochodne $\mathbf{L}$ po wszystkich wagach, co pozwala zastosować algorytm *gradient descent*.

Funkcja softmax i „categorical cross entropy". W problemach kategotyzacji często używa się w *ostatniej* warstwie funkcji aktywującej *softmax*. Ma ona taką zaletę, że wówczas wyjście sieci ma współrzędne nieujemne sumujące się do 1, można więc je interpretować jako rozkład prawdopodobieństwa. Niestety, funkcja softmax nie wpisuje się w ogólny schemat pokazany do tej pory, ponieważ zależy ona od całego wektora $net^{[L]}$, konkretnie

$$a^{[L]} = \psi(net^{[L]}) := \left[\frac{\exp(net_k^{[L]})}{\sum_{j=1}^{n^{[L]}} \exp(net_j^{[L]})} \right]_{k=1,\ldots,n^{[L]}}.$$

W związku z tym propagacja wsteczna dla ostatniej warstwy będzie miała inną postać, którą teraz znajdziemy. Załóżymy tutaj, że używamy funkcji kosztu *categorical cross entropy*, określonej następująco

$$\mathbf{L}(y, a^{[L]}) = -\sum_{k=1}^{n^{[L]}} y_k \log(a_k^{[L]}) = -\sum_{k=1}^{n^{[L]}} y_k \left(net_k^{[L]} - \log(\sum_{j=1}^{n^{[L]}} \exp(net_j^{[L]})) \right).$$

Zwróćmy uwagę, że $a_k^{[L]} > 0$, a więc funkcja ta jest dobrze określona. Przypomnijmy, że

$$net_k^{[L]} = \sum_{j=0}^{n^{[L-1]}} w_{k,j}^{[L]} x_j^{[L-1]}, \qquad k = 1, \ldots, n^{[L]}.$$

Obliczamy

$$\begin{aligned}
\frac{\partial \mathbf{L}}{\partial w_{k_0,j_0}^{[L]}} &= -y_{k_0} x_{j_0}^{[L-1]} + \sum_{k=1}^{n^{[L]}} y_k \frac{1}{\sum_{j=1}^{n^{[L]}} \exp(net_j^{[L]})} \cdot \left(\frac{\partial}{\partial w_{k_0,j_0}^{[L]}} \exp(net_{k_0}^{[L]}) \right) \\
&= -y_{k_0} x_{j_0}^{[L-1]} + \sum_{k=1}^{n^{[L]}} y_k \frac{\exp(net_{k_0}^{[L]})}{\sum_{j=1}^{n^{[L]}} \exp(net_j^{[L]})} x_{j_0}^{[L-1]} \\
&= \left(\left(\sum_{k=1}^{n^{[L]}} y_k \right) \cdot a_{k_0}^{[L]} - y_{k_0} \right) x_{j_0}^{[L-1]}.
\end{aligned}$$

Kładąc

$$\delta^{[L]} = \left[\left(\sum_{j=1}^{n^{[L]}} y_j \right) \cdot a_k^{[L]} - y_k \right]_{k=1,\ldots,n^{[l]}}, \tag{0.2}$$

otrzymujemy taki sam wzór, jak poprzednio:

$$\frac{\partial \mathbf{L}}{\partial w^{[L]}} = \delta^{[L]} \cdot (x^{[L-1]})^T.$$

Obliczmy jeszcze podobnie

$$\begin{aligned}
\frac{\partial \mathbf{L}}{\partial x_{j_0}^{[L-1]}} &= -\sum_{k=1}^{n^{[L]}} y_k w_{k,j_0}^{[L]} + \sum_{k=1}^{n^{[L]}} y_k \frac{1}{\sum_{j=1}^{n^{[L]}} \exp(net_j^{[L]})} \cdot \left(\frac{\partial}{\partial x_{j_0}^{[L-1]}} \sum_{j=1}^{n^{[L]}} \exp(net_j^{[L]}) \right) \\
&= -\sum_{k=1}^{n^{[L]}} y_k w_{k,j_0}^{[L]} + \sum_{k=1}^{n^{[L]}} y_k \sum_{p=1}^{n^{[L]}} \frac{\exp(net_p^{[L]}) w_{p,j_0}^{[L]}}{\sum_{j=1}^{n^{[L]}} \exp(net_j^{[L]})} \\
&= -\sum_{p=1}^{n^{[L]}} y_p w_{p,j_0}^{[L]} + \sum_{k=1}^{n^{[L]}} y_k \sum_{p=1}^{n^{[L]}} a_p^{[L]} w_{p,j_0}^{[L]} \\
&= \sum_{p=1}^{n^{[L]}} \left(\Big(\sum_{k=1}^{n^{[L]}} y_k \Big) a_p^{[L]} - y_p \right) w_{p,j_0}^{[L]} \\
&= \sum_{p=1}^{n^{[L]}} \delta_p^{[L]} w_{p,j_0}^{[L]}.
\end{aligned}$$

Otrzymaliśmy znowu taki sam wzór jak poprzednio, w notacji macierzowej:

$$\frac{\partial \mathbf{L}}{\partial x^{[L-1]}} = (W^{[L]})^T \cdot \delta^{[L]}.$$

Podsumowując, zachodzą takie same wzory jak poprzednio, jeśli tylko zmodyfikujemy definicję $\delta^{[L]}$ (tylko dla ostatniej warstwy), przyjmując (0.2).

Zobaczmy jeszcze, że wzór na $\delta^{[L]}$ upraszcza się, jeśli założymy, że $\sum_{k=1}^{n^{[L]}} y_k = 1$ (tak typowo jest w zagadnieniach klasyfikacyjnych). Wówczas

$$\delta^{[L]} = \left[a_k^{[L]} - y_k \right]_{k=1,\ldots,n^{[l]}}.$$