Introduction aux réseaux de neurones – exercices

Question 1. Nous considérons un réseau de neurones à convolutions qui reçoit en entrée des images de taille 6×6 ne possédant qu'un seul canal, telles que la matrice X_1 suivante :

$$X_1 = \begin{bmatrix} 0&0&0&0&0&0 \\ 0&1&1&1&1&0 \\ 0&1&0&0&1&0 \\ 0&1&0&0&1&0 \\ 0&1&1&1&1&0 \\ 0&0&0&0&0&0 \end{bmatrix}.$$

Le réseau comprend deux couches de convolutions, chacune suivie d'une fonction d'activation de type ReLU, ainsi qu'une couche pleinement connectée. Il y a un seul neurone de sortie, muni d'une activation linéaire. Aucune couche ne possède de paramètres de biais.

Plus spécifiquement,

1. La première couche de convolution comporte 3 filtres de taille 3×3, exprimés par les matrices suivantes.

$$W_1^{(1)} = \begin{bmatrix} 1&0&-1 \\ 1&0&-1 \\ 1&0&-1 \end{bmatrix}; \quad W_2^{(1)} = \begin{bmatrix} 1&1&1 \\ 0&0&0 \\ -1&-1&-1 \end{bmatrix}; \quad W_3^{(1)} = \begin{bmatrix} 0&1&0 \\ 1&-4&1 \\ 0&1&0 \end{bmatrix}.$$

2. La deuxième couche de convolution comporte un seul filtre de taille $3 \times 3 \times 3$, exprimé par le tenseur suivant :

$$W^{(2)}[i,j,k] = \begin{cases} 1 & \text{si } j = k = 2 \\ 0 & \text{sinon,} \end{cases} \qquad \text{pour } i,j,k \in \{1,2,3\}.$$

3. La troisième couche (la couche de sortie) est une couche pleinement connectée exprimée par un simple vecteur de taille 4 :

$$\mathbf{w}^{(3)} = \begin{bmatrix} 6 \\ 2 \\ 0 \\ -1 \end{bmatrix}.$$

Calculez la valeur de sortie du réseau pour l'image X_1, en suivant les étapes suivantes :

(a) Calculez la représentation associée à la première couche cachée. Plus spécifiquement, pour chacun des trois filtres $W_1^{(1)}$, $W_2^{(1)}$ et $W_3^{(1)}$, calculer le résultat de la convolution suivit de l'activation ReLU :

$$H_i^{(1)} = \text{relu}(X_1 * W_i^{(1)}).$$

SOLUTION:

$$X_1 * W_1^{(1)} = \begin{bmatrix} \text{-1} & 1 & \text{-1} & 1 \\ \text{-1} & 2 & \text{-2} & 1 \\ \text{-1} & 2 & \text{-2} & 1 \\ \text{-1} & 1 & \text{-1} & 1 \end{bmatrix}; \quad H_1^{(1)} = \text{relu}(X_1 * W_1^{(1)}) = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 2 & 0 & 1 \\ 0 & 2 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix},$$

$$X_1 * W_2^{(1)} = \begin{bmatrix} \text{-1} & \text{-1} & \text{-1} & \text{-1} \\ 1 & 2 & 2 & 1 \\ \text{-1} & \text{-2} & \text{-2} & \text{-1} \\ 1 & 1 & 1 & 1 \end{bmatrix}; \quad H_2^{(1)} = \text{relu}(X_2 * W_2^{(1)}) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 2 & 2 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix},$$

$$X * W_3^{(1)} = \begin{bmatrix} \text{-2} & \text{-2} & \text{-2} & \text{-2} \\ \text{-2} & 2 & 2 & \text{-2} \\ \text{-2} & 2 & 2 & \text{-2} \\ \text{-2} & \text{-2} & \text{-2} & \text{-2} \end{bmatrix}; \quad H_3^{(1)} = \text{relu}(X_1 * W_3^{(1)}) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 2 & 2 & 0 \\ 0 & 2 & 2 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

(b) Notons $H_{1:3}^{(1)}$ la juxtaposition des trois matrices calculées en (a) en un seul tenseur de taille $3 \times 3 \times 3$. Calculez la représentation associée à la deuxième couche cachée, c'est-à-dire le résultat de la couche convolution suivante :

$$H^{(2)} = \text{relu}(H_{1:3}^{(1)} * W^{(2)}),$$

avec $H_{1:3}^{(1)}[i,j,k] = H_i^{(1)}[j,k]$ pour $i \in \{1,2,3\}$ et $j,k \in \{1,2,3,4\}$.

SOLUTION:

$$H_{1:3}^{(1)} * W^{(2)} = \begin{bmatrix} 2{+}2{+}2 & 0{+}2{+}2 \\ 2{+}0{+}2 & 0{+}0{+}2 \end{bmatrix} = \begin{bmatrix} 6 & 4 \\ 4 & 2 \end{bmatrix}; \quad H^{(2)} = \text{relu}(H_{1:3}^{(1)} * W^{(2)}) = \begin{bmatrix} 6 & 4 \\ 4 & 2 \end{bmatrix}.$$

(c) Notons $\mathbf{h}^{(2)}$ la *vectorisation* de la matrice $H^{(2)}$ en un vecteur colonne de 4 éléments. Calculez la valeur du neurone de sortie, c'est-à-dire :

$$y = \mathbf{w}^{(3)} \cdot \mathbf{h}^{(2)}.$$

avec $\mathbf{h}^{(2)}[i + 2(j{-}1)] = H^{(2)}[i,j]$ pour $i,j,k \in \{1,2\}$.

SOLUTION:

$$y = \begin{bmatrix} 6 \\ 2 \\ 0 \\ \text{-1} \end{bmatrix} \cdot \begin{bmatrix} 6 \\ 4 \\ 4 \\ 2 \end{bmatrix} = 36 + 8 + 0 - 2 = 42.$$

Question 2. Considérons un réseau de neurones qui reçoit en entrée des images en teintes de gris représentées par des matrices de 20×20 pixels. Chacune de ces images illustre un des fruits suivants : banane, pomme, poire ou ananas. Il s'agit donc d'un problème de classification multi-classe ; le réseau possède quatre neurones de sorties.

(a) Si le réseau possède une seule couche cachée pleinement connectée de n neurones, combien de paramètres devrons-nous optimiser par descente de gradient ?

— Si on n'utilise aucun paramètre de biais, ni pour la couche cachée et ni pour la couche de sortie ?

> <u>SOLUTION:</u>
> Paramètres de la couche cachée : $400n$;
> Paramètres de la couche de sortie : $n4$.
> **Total :** $404n$ paramètres.

— Si on inclut les paramètres de biais pour la couche cachée et pour la couche de sortie ?

> <u>SOLUTION:</u>
> Paramètres de la couche cachée : $(400 + 1)n$;
> Paramètres de la couche de sortie : $(n + 1)4$.
> **Total :** $405n + 4$ paramètres.

(b) Si le réseau possède une première couche de m filtres de convolutions de taille 5×5, suivie d'une couche cachée pleinement connectée de n neurones, et d'aucun paramètre de biais, combien de paramètres devrons-nous optimiser par descente de gradient ?

> <u>SOLUTION:</u>
> Paramètres de la couche de convolution : $5^2m = 25m$;
> Paramètres de la couche pleinement connectée : $(20 - 4)^2mn = 256mn$.
> Paramètres de la couche de sortie : $n4$.
> **Total :** $256mn + 25m + 4n$ paramètres.

(c) Si le réseau possède une première couche de m filtres de convolutions de taille 5×5, suivit d'une couche de *MaxPooling* avec une taille de pas de 4, suivit d'une couche cachée pleinement connectée de n neurones, et d'aucun paramètre de biais, combien de paramètres devrons-nous optimiser par descente de gradient ?

> <u>SOLUTION:</u>
> Paramètres de la couche de convolution : $5^2m = 25m$;
> Paramètres de la couche pleinement connectée : $(\frac{20-4}{4})^2mn = 16mn$.
> Paramètres de la couche de sortie : $n4$.
> **Total :** $16mn + 25m + 4n$ paramètres.

(d) Enfin, considérons l'architecture suivante :

— Entrée du réseau : images de tailles 20×20;
— 1ère couche : m_1 filtres de convolutions de taille 5×5, sans biais.
— *MaxPooling* avec taille de pas 2
— 2e couche : m_2 filtres de convolutions de taille $3 \times 3 \times m_1$, sans biais.
— *MaxPooling* avec taille de pas 2
— 3e couche : n_1 neurones pleinement connectés, sans biais
— 4e couche : n_2 neurones pleinement connectés, sans biais
— 5e couche : 4 neurones de sortie, avec biais.

<u>Solution:</u>
— 1ère couche : $5^2 m_1 = 25m_1$
— 2e couche : $3^2 m_1 m_2 = 9m_1 m_2$
— 3e couche : $\left(\frac{(20-4)/2-2}{2}\right)^2 m_2 n_1 = 9m_2 n_1$
— 4e couche : $n_1 n_2$
— 5e couche : $(n_2 + 1)4$
Total : $25m_1 + 9m_1 m_2 + 9m_2 n_1 + n_1 n_2 + 4n_2 + 4$ paramètres.

Question 3. Reprendre la question 2(a-d) en considérant cette fois-ci que les images sont en couleurs. Le réseau reçoit donc en entrée des matrices de taille $20 \times 20 \times 3$.

<u>Solution:</u>

(a) — Sans paramètres de biais :
Paramètres de la couche cachée : $20 \times 20 \times 3 \times n = 1200n$;
Paramètres de la couche de sortie : $n4$.
Total : $1204n$ paramètres.

— Avec paramètres de biais :
Paramètres de la couche cachée : $(1200 + 1)n$;
Paramètres de la couche de sortie : $(n + 1)4$.
Total : $1205n + 4$ paramètres.

(b) Paramètres de la couche de convolution : $5^2 \times 3m = 75m$;
Paramètres de la couche pleinement connectée : $(20 - 4)^2 mn = 256mn$.
Paramètres de la couche de sortie : $n4$.
Total : $256mn + 75m + 4n$ paramètres.

(c) Paramètres de la couche de convolution : $5^2 \times 3m = 75m$;
Paramètres de la couche pleinement connectée : $(\frac{20-4}{4})^2 mn = 16mn$.
Paramètres de la couche de sortie : $n4$.
Total : $16mn + 75m + 4n$ paramètres.

(d) — 1ère couche : $5^2 \times 3m_1 = 75m_1$
— 2e couche : $3^2 m_1 m_2 = 9m_1 m_2$
— 3e couche : $\left(\frac{(20-4)/2-2}{2}\right)^2 m_2 n_1 = 9m_2 n_1$

— 4e couche : $n_1 n_2$
— 5e couche : $(n_2 + 1)4$

Total : $75m_1 + 9m_1m_2 + 9m_2n_1 + n_1n_2 + 4n_2 + 4$ paramètres.