Introduction aux réseaux de neurones – exercices

Question 1. Considérons un problème de régression où l'ensemble d'apprentissage $S \in \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ contient des couples $(\mathbf{x}_i, y_i) \in \mathbb{R}^d \times \mathbb{R}$. Nous exprimons ainsi la fonction objectif à minimiser pour résoudre les moindres carrés régularisés (aussi nommé la *Régression de Ridge*) :

$$F(\mathbf{w}, b) = \frac{1}{n} \sum_{i=1}^{n} (f_{\mathbf{w},b}(\mathbf{x}_i) - y_i)^2 + \frac{\rho}{2} \|\mathbf{w}\|^2 ,$$

où le prédicteur linéaire $f_{\mathbf{w},b}(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b$ est caractérisé par un vecteur de poids $\mathbf{w} \in \mathbb{R}^d$ et un biais $b \in \mathbb{R}$.

(a) Pour $k \in \{1, \dots, d\}$, calculer $\dfrac{\partial F(\mathbf{w}, b)}{\partial w_k}$, c'est-à-dire le gradient de la fonction objectif selon le $k$ième paramètre du vecteur de poids.

Solution:

$$\begin{aligned}
\frac{\partial F(\mathbf{w}, b)}{\partial w_k} &= \frac{1}{n} \sum_{i=1}^{n} \frac{\partial}{\partial w_k} (\mathbf{w} \cdot \mathbf{x}_i + b - y_i)^2 + \frac{\rho}{2} \sum_{j=1}^{d} \frac{\partial}{\partial w_k} w_j^2 \\
&= \frac{1}{n} \sum_{i=1}^{n} 2\,(\mathbf{w} \cdot \mathbf{x}_i + b - y_i) \frac{\partial}{\partial w_k} (\mathbf{w} \cdot \mathbf{x}_i + b - y_i) + \frac{\rho}{2}\, 2\, w_k \\
&= \frac{1}{n} \sum_{i=1}^{n} 2\,(\mathbf{w} \cdot \mathbf{x}_i + b - y_i) \frac{\partial}{\partial w_k} (w_k\, x_{i,k}) + \rho\, w_k \\
&= \frac{2}{n} \sum_{i=1}^{n} (\mathbf{w} \cdot \mathbf{x}_i + b - y_i)\, x_{i,k} + \rho\, w_k .
\end{aligned}$$

(b) Déduire de la réponse précédente l'expression du gradient $\nabla_{\mathbf{w}} F(\mathbf{w}, b)$.

Solution:

$$\nabla_{\mathbf{w}} F(\mathbf{w}, b) = \frac{2}{n} \sum_{i=1}^{n} (\mathbf{w} \cdot \mathbf{x}_i + b - y_i)\, \mathbf{x}_i + \rho\, \mathbf{w} .$$

(c) Calculez $\dfrac{\partial F(\mathbf{w}, b)}{\partial b}$, c'est-à-dire le gradient du biais.

SOLUTION:

$$\begin{aligned}\frac{\partial F(\mathbf{w},b)}{\partial b} &= \frac{1}{n}\sum_{i=1}^{n}\frac{\partial}{\partial b}(\mathbf{w}\cdot\mathbf{x}_i+b-y_i)^2+\frac{\rho}{2}\sum_{j=1}^{d}\frac{\partial}{\partial b}w_j^2\\ &= \frac{1}{n}\sum_{i=1}^{n}2\,(\mathbf{w}\cdot\mathbf{x}_i+b-y_i)\frac{\partial}{\partial b}(\mathbf{w}\cdot\mathbf{x}_i+b-y_i)\\ &= \frac{2}{n}\sum_{i=1}^{n}(\mathbf{w}\cdot\mathbf{x}_i+b-y_i)\,.\end{aligned}$$

Question 2. Considérons un problème de classification binaire où l'ensemble d'apprentissage $S \in \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ contient des couples $(\mathbf{x}_i, y_i) \in \mathbb{R}^d \times \{0,1\}$. Nous exprimons ainsi la fonction objectif à minimiser pour résoudre la régression logistique :

$$F(\mathbf{w},b) = \frac{1}{n}\sum_{i=1}^{n} F_i(\mathbf{w},b)\,,$$

avec

$$\begin{aligned}F_i(\mathbf{w},b) &= -y_i(\mathbf{w}\cdot\mathbf{x}_i+b)+\log(1+e^{\mathbf{w}\cdot\mathbf{x}_i+b})+\frac{\rho}{2}\|\mathbf{w}\|^2\\ &= L_{\text{nlv}}\big(\sigma(\mathbf{w}\cdot\mathbf{x}_i+b),y_i\big)+\frac{\rho}{2}\|\mathbf{w}\|^2\end{aligned}$$

(a) Calculer $\dfrac{\partial L_{\text{nlv}}(\hat{y},y)}{\partial \hat{y}}$, où $L_{\text{nlv}}(\hat{y},y) = -y\log(\hat{y}) - (1-y)\log(1-\hat{y})\,.$

SOLUTION:

$$\begin{aligned}\frac{\partial L_{\text{nlv}}(\hat{y},y)}{\partial \hat{y}} &= \frac{\partial}{\partial \hat{y}}\left[-y\log(\hat{y})-(1-y)\log(1-\hat{y})\right]\\ &= -y\frac{\partial}{\partial \hat{y}}\log(\hat{y})-(1-y)\frac{\partial}{\partial \hat{y}}\log(1-\hat{y})\\ &= -y\frac{1}{\hat{y}}-(1-y)\frac{-1}{1-\hat{y}}\\ &= -\frac{y}{\hat{y}}+\frac{1-y}{1-\hat{y}}\,.\end{aligned}$$

(b) Calculer $\dfrac{\partial \sigma(a)}{\partial a}$ où $\sigma(a) = \dfrac{1}{1+e^{-a}}\,.$

SOLUTION:

$$\begin{aligned}
\frac{\partial \sigma(a)}{\partial a} &= \frac{\partial}{\partial a}(1+e^{-a})^{-1} \\
&= -(1+e^{-a})^{-2}\,\frac{\partial}{\partial a}(1+e^{-a}) \\
&= -\frac{1}{(1+e^{-a})^2}\,\left[\frac{\partial}{\partial a}e^{-a}\right] \\
&= -\frac{1}{(1+e^{-a})^2}\,\left[-e^{-a}\right] \\
&= \frac{e^{-a}}{(1+e^{-a})^2}.
\end{aligned}$$

Notez qu'il est commun de simplifier l'expression ainsi :

$$\begin{aligned}
\frac{e^{-a}}{(1+e^{-a})^2} = \frac{1+e^{-a}}{(1+e^{-a})^2} - \frac{1}{(1+e^{-a})^2} &= \frac{1}{1+e^{-a}} - \left(\frac{1}{1+e^{-a}}\right)^2 \\
&= \sigma(a) - \Big(\sigma(a)\Big)^2 \\
&= \sigma(a)\,\Big(1-\sigma(a)\Big)\,.
\end{aligned}$$

(c) Pour $k \in \{1,\ldots,d\}$, calculer $\dfrac{\partial(\mathbf{w}\cdot\mathbf{x}+b)}{\partial w_k}$ où $\mathbf{w} = \begin{bmatrix} w_1 \\ \vdots \\ w_d \end{bmatrix}$, $\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix}$.

SOLUTION:

$$\begin{aligned}
\frac{\partial(\mathbf{w}\cdot\mathbf{x}+b)}{\partial w_k} &= \frac{\partial}{\partial w_k}\left[\sum_{i=1}^{d} w_i x_i + b\right] \\
&= \frac{\partial}{\partial w_k} w_k x_k \\
&= x_k\,.
\end{aligned}$$

(d) Pour $k \in \{1,\ldots,d\}$, calculer $\dfrac{\partial\|\mathbf{w}\|^2}{\partial w_k}$.

Rappel : $\|\mathbf{w}\|^2 = \mathbf{w}\cdot\mathbf{w}$

SOLUTION:

$$\begin{aligned}
\frac{\partial\,\mathbf{w}\cdot\mathbf{w}}{\partial w_k} &= \frac{\partial}{\partial w_k}\left[\sum_{i=1}^{d} w_i^2\right] \\
&= \frac{\partial}{\partial w_k} w_k^2 \\
&= 2\,w_k\,.
\end{aligned}$$

(e) À partir des réponses aux questions précédentes, calculer $\dfrac{\partial F_i(\mathbf{w},b)}{\partial w_k}$.

Rappel de la règle de la dérivation en chaîne : $\frac{\partial f(h(x))}{\partial x} = \Big[\frac{\partial f(a)}{\partial a}\Big]_{a=h(x)} \frac{\partial h(x)}{\partial x}$.

SOLUTION:

Grâce à la réponse (d), nous avons $\dfrac{\partial}{\partial w_k}\dfrac{\rho}{2}\|\mathbf{w}\|^2 = \dfrac{\rho}{2}\big[2\,w_k\big] = \rho\,w_k$. En utilisant la règle de dérivation en chaîne avec les réponses obtenues en (a), (b) et (c), nous obtenons :

$$\begin{aligned}
\frac{\partial}{\partial w_k} L_{\text{nlv}}\big(\sigma(\mathbf{w}\cdot\mathbf{x}_i+b), y\big) &= \left[\frac{\partial L_{\text{nlv}}(\hat{y}, y_i)}{\partial \hat{y}}\right]_{\hat{y}=\sigma(\mathbf{w}\cdot\mathbf{x}_i+b)} \frac{\partial \sigma(\mathbf{w}\cdot\mathbf{x}_i+b)}{\partial w_k} \\
&= \left[\frac{\partial L_{\text{nlv}}(\hat{y}, y_i)}{\partial \hat{y}}\right]_{\hat{y}=\sigma(\mathbf{w}\cdot\mathbf{x}_i+b)} \left[\frac{\partial \sigma(a)}{\partial a}\right]_{a=\mathbf{w}\cdot\mathbf{x}_i+b} \frac{\partial \mathbf{w}\cdot\mathbf{x}_i+b}{\partial w_k} \\
&= \left[-\frac{y_i}{\hat{y}_i} + \frac{1-y_i}{1-\hat{y}_i}\right]\Big[\hat{y}_i\Big(1-\hat{y}_i\Big)\Big] x_k \\
&= \Big[-y_i\,(1-\hat{y}_i) + (1-y_i)\,\hat{y}_i\Big] x_k \quad \text{avec } \hat{y}_i = \sigma(\mathbf{w}\cdot\mathbf{x}_i+b).
\end{aligned}$$

Ainsi :

$$\begin{aligned}
\frac{\partial F_i(\mathbf{w},b)}{\partial w_k} &= \frac{\partial}{\partial w_k} L_{\text{nlv}}\big(\sigma(\mathbf{w}\cdot\mathbf{x}_i+b), y_i\big) + \frac{\partial}{\partial w_k}\frac{\rho}{2}\|\mathbf{w}\|^2 \\
&= \Big[-y_i\,(1-\sigma(\mathbf{w}\cdot\mathbf{x}_i+b)) + (1-y_i)\,\sigma(\mathbf{w}\cdot\mathbf{x}_i+b)\Big] x_k + \rho\,w_k\,.
\end{aligned}$$

(f) Déduire de la réponse précédente l'expression du gradient $\nabla_{\mathbf{w}} F_i(\mathbf{w}, b)$.

SOLUTION:

$$\nabla_{\mathbf{w}} F_i(\mathbf{w},b) = \Big[-y_i\,(1-\sigma(\mathbf{w}\cdot\mathbf{x}_i+b)) + (1-y_i)\,\sigma(\mathbf{w}\cdot\mathbf{x}_i+b)\Big]\mathbf{x}_i + \rho\,\mathbf{w}\,.$$

(g) Calculez $\dfrac{\partial F_i(\mathbf{w}, b)}{\partial b}$, c'est-à-dire le gradient du biais.

SOLUTION:

$$\begin{aligned}
\frac{\partial F_i(\mathbf{w},b)}{\partial b} &= \frac{\partial}{\partial b} L_{\text{nlv}}\big(\sigma(\mathbf{w}\cdot\mathbf{x}_i+b), y_i\big) + \frac{\partial}{\partial b}\frac{\rho}{2}\|\mathbf{w}\|^2 \\
&= \frac{\partial}{\partial b} L_{\text{nlv}}\big(\sigma(\mathbf{w}\cdot\mathbf{x}_i+b), y_i\big) + 0 \\
&= \left[\frac{\partial L_{\text{nlv}}(\hat{y}, y)}{\partial \hat{y}}\right]_{\hat{y}=\sigma(\mathbf{w}\cdot\mathbf{x}_i+b)} \frac{\partial \sigma(\mathbf{w}\cdot\mathbf{x}_i+b)}{\partial b} \\
&= \left[\frac{\partial L_{\text{nlv}}(\hat{y}, y)}{\partial \hat{y}}\right]_{\hat{y}=\sigma(\mathbf{w}\cdot\mathbf{x}_i+b)} \left[\frac{\partial \sigma(a)}{\partial a}\right]_{a=\mathbf{w}\cdot\mathbf{x}_i+b} \frac{\partial \mathbf{w}\cdot\mathbf{x}_i+b}{\partial b} \\
&= \left[-\frac{y_i}{\hat{y}_i} + \frac{1-y_i}{1-\hat{y}_i}\right]\Big[\hat{y}_i\Big(1-\hat{y}_i\Big)\Big] \times 1 \\
&= \Big[-y_i\,(1-\hat{y}_i) + (1-y_i)\,\hat{y}_i\Big] \quad \text{avec } \hat{y}_i = \sigma(\mathbf{w}\cdot\mathbf{x}_i+b).
\end{aligned}$$

Question 3. Imaginons que nous optimisons la descente en gradient en initialisant tous les paramètres à zéro.

(a) Est-ce que cela peut poser problème dans le cas de la régression logistique ? Rappelons que la régression logistique peut être exprimée comme un réseau n'ayant aucune couche cachée.

> SOLUTION: Non, la régression logistique est un problème convexe et la descente de gradient convergera vers un minimum global peu importe l'initialisation.

(b) Est-ce que cela peut poser problème dans le cas d'un réseau de neurones à une couche cachée ?

> SOLUTION: Oui, car les symétries dans le problème d'optimisation feront en sorte que les gradients associés au poids de chaque neurone de la couche cachée seront les mêmes.
>
> Par exemple, considérez un réseau très simple, avec un neurone d'entrée x, une couche cachée de deux neurones avec fonction d'activation f (notons les paramètres de la couche cachée w_1 et w_2), et un neurone de sortie avec une fonction d'activation linéaire notons les paramètres de la couche de sortie v_1 et v_2. On a donc
>
> $$R(x) = v_1 f(w_1 x) + v_2 f(w_2 x)$$
>
> Pour un couple (x, y), le gradient la fonction de perte $L : \mathbb{R}^2 \to \mathbb{R}$ associé à w_1 est
>
> $$\begin{aligned}\frac{\partial L(R(x), y)}{\partial w_1} &= L'(R(x), y) \times \frac{\partial}{\partial w_1} \left[v_1 f(w_1 x) + v_2 f(w_2 x)\right] \\ &= L'(R(x), y) \times \left[v_1 \frac{\partial}{\partial w_1} f(w_1 x)\right] \\ &= L'(R(x), y)\, v_1\, f'(w_1 x)\, \frac{\partial w_1 x}{\partial w_1} \\ &= L'(R(x), y)\, v_1\, f'(w_1 x)\, x\end{aligned}$$
>
> Par des calculs très similaires, nous obtenons le gradient associé à w_2 est
>
> $$\frac{\partial L(R(x), y)}{\partial w_2} = L'(R(x), y)\, v_2\, f'(w_2 x)\, x\,.$$
>
> Nous pouvons donc constater que si $w_1 = w_2$ et $v_1 = v_2$, les gradients selon w_1 et w_2 seront égaux. En particulier, si tous les paramètres d'initialisation sont nuls, on remarque que, pour cet exemple, les gradients selon w_1 et w_2 seront nuls et la valeur de ces paramètres ne seront pas modifiés à la première itération ! Cependant, selon la fonction d'activation f, il est possible que les paramètres v_1 et v_2 soient modifiés, auquel cas w_1 et w_2 le seront aussi dès la deuxième itération de la descente de gradient.

(c) Dans le cas d'un réseau de neurones à une couche cachée, est-ce qu'utiliser le «dropout» sur la couche cachée peut résoudre un éventuel problème dû à l'initialisation des paramètres.

SOLUTION: Oui, car les éventuelles symétries seront brisées au fil des itérations ; le «dropout» fera en sorte que certains poids ne seront pas mis à jour lors de la rétropropagation des erreurs.

Question 4. Les fonctions d'activations utilisées pour les couches cachées d'un réseau de neurones sont rarement des fonctions linéaires $f(a) = a$. Illustrez la raison à l'aide d'un exemple.

SOLUTION: Un réseau de neurones dont les couches cachées possèdent que des fonctions d'activations linéaires peut s'exprimer en un réseau n'ayant aucune couche cachée.

Pour illustrer ce phénomène, considérons un réseau à une couche cachée exprimée par la matrice de poids $\mathbf{W} \in \mathbb{R}^{D \times d}$ et une couche de sortie $\mathbf{V} \in \mathbb{R}^{s \times D}$. Ce réseau possède donc d neurones d'entrées, une couche cachée de D neurones, et s neurones de sortie.

Étant donné un exemple $\mathbf{x} \in \mathbb{R}^d$, exprimons la sortie de ce réseau comme $R(\mathbf{x}) = g(\mathbf{V}f(\mathbf{W}\mathbf{x}))$, où $f(a) = a$ est une fonction d'activation linéaire et g est la fonction d'activation de la couche de sortie.

$$\begin{aligned} R(\mathbf{x}) &= g(\mathbf{V}f(\mathbf{W}\mathbf{x})) \\ &= g(\mathbf{V}\mathbf{W}\mathbf{x}) \\ &= g(\mathbf{U}\mathbf{x}) \qquad\qquad \text{avec } \mathbf{U} = \mathbf{V}\mathbf{W} \in \mathbb{R}^{s \times d}. \end{aligned}$$

Le réseau à deux couches (dont une couche cachée) R peut donc être exprimé comme un réseau à une seule couche $R'(\mathbf{x}) = g(\mathbf{U}\mathbf{x})$.

Question 5. Considérons une architecture de réseau de neurones dédiée à un problème de classification à C classes. On représente chaque classe par un entier $y \in \{1, \ldots, C\}$, et la couche de sortie du réseau possède C neurones. Pour un exemple d'apprentissage $(\mathbf{x}, y)$, on converti y sous la forme d'un vecteur «one-hot» $\mathbf{y} \in \mathbb{R}^C$, possédant la valeur 1 à l'index correspondant à y, et les valeurs 0 autrement :

$$y = 1 \mapsto \mathbf{y} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \qquad y = 2 \mapsto \mathbf{y} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Les valeurs de couche de sortie $\hat{\mathbf{y}} \in \mathbb{R}^C$ sont obtenues en appliquant la fonction d'activation «softmax» aux valeurs $\mathbf{a} \in \mathbb{R}^C$ propagées par les couches précédentes :

$$\hat{\mathbf{y}} = \begin{bmatrix} \hat{y}_1 \\ \vdots \\ \hat{y}_C \end{bmatrix} \quad \text{avec } \hat{y}_i = \text{softmax}(a_i) = \frac{e^{a_i}}{\sum_{j=1}^{C} e^{a_j}}$$

On interprète le vecteur $\mathbf{y}$ comme une distribution de probabilité sur les classes. Lors de l'optimisation du réseau, on minimise la perte du négatif log-vraisemblance :

$$L_{\text{nlv}}(\hat{\mathbf{y}}, y) = -\ln(\hat{y}_y),$$

où $\hat{y}_y$ corresponds à la sortie d'index y du réseau (la probabilité associée à la classe de l'exemple).

Cet exercice consiste à calculer $\nabla_{\mathbf{a}} L_{\text{nlv}}(\hat{y}_y, y)$, c'est-à-dire le gradient qui modifiera les poids associés à la couche de sortie. Pour ce faire, procédons en trois étapes.

(a) Calculer la dérivée partielle associée au neurone de sortie correspondant à la bonne classe ($y = i$) :

$$\frac{\partial}{\partial a_y} L_{\text{nlv}}(\hat{y}_y, y) = \frac{\partial}{\partial a_y} \ln \left[\frac{1}{\text{softmax}(a_y)} \right].$$

Solution:

$$\begin{aligned}
\frac{\partial}{\partial a_y} L_{\text{nlv}}(\hat{y}_y, y) &= \frac{\partial}{\partial a_y} \ln \left[\frac{1}{\text{softmax}(a_y)} \right] \\
&= \frac{\partial}{\partial a_y} \ln \left[\frac{\sum_{j=1}^{C} e^{a_j}}{e^{a_y}} \right] \\
&= \frac{1}{\sum_{j=1}^{C} e^{a_j}} \times \frac{\partial}{\partial a_y} \left[\sum_{j=1}^{C} e^{a_j} \right] - \frac{\partial}{\partial a_y} a_y \\
&= \frac{1}{\sum_{j=1}^{C} e^{a_j}} \times a_y - 1 \\
&= \text{softmax}(a_y) - 1 \\
&= \hat{y}_y - 1
\end{aligned}$$

(b) Calculer la dérivée partielle associée au neurone de sortie correspondant à la bonne classe ($y \neq i$) :

$$\frac{\partial}{\partial a_i} L_{\text{nlv}}(\hat{y}_y, y) = \frac{\partial}{\partial a_i} \ln \left[\frac{1}{\text{softmax}(a_y)} \right], \text{ avec } a_i \neq a_y \,.$$

SOLUTION:

$$\begin{aligned}
\frac{\partial}{\partial a_i} L_{\text{nlv}}(\hat{y}_y, y) &= \frac{\partial}{\partial a_i} \ln \left[\frac{1}{\text{softmax}(a_y)} \right] \\
&= \frac{\partial}{\partial a_i} \ln \left[\frac{\sum_{j=1}^{C} e^{a_j}}{e^{a_y}} \right] \\
&= \frac{1}{\sum_{j=1}^{C} e^{a_j}} \times \frac{\partial}{\partial a_i} \left[\sum_{j=1}^{C} e^{a_j} \right] - \frac{\partial}{\partial a_i} a_y \\
&= \frac{1}{\sum_{j=1}^{C} e^{a_j}} \times a_i \\
&= \text{softmax}(a_i) \\
&= \hat{y}_i
\end{aligned}$$

(c) À partir des réponses (a) et (b), déduire le gradient $\nabla_{\mathbf{a}} L_{\text{nlv}}(\hat{y}_y, y)$.

SOLUTION:

$$\nabla_{\mathbf{a}} L_{\text{nlv}}(\hat{y}_y, y) = \hat{\mathbf{y}} - \mathbf{y} \,.$$