

# Polytech - Deep Learning - TP 3-4

## Introduction aux réseaux de neurones

### Notes additionnelles

Les batch seront stockés dans des matrices avec 1 exemple par ligne. Par exemple pour l'entrée on a une matrice  $\mathbf{X}$  de taille  $N \times n_x$  pour  $N$  exemples chacun de dimension  $n_x$ . Chaque exemple  $\mathbf{x}$  sera donc un vecteur ligne (dimension  $1 \times n_x$ ). On fera de même pour les résultats intermédiaires. Ainsi, la matrice de poids  $\mathbf{W}^h$  sera par exemple de taille  $n_x \times n_h$ . Le vecteur de bias  $\mathbf{b}^h$  sera un vecteur ligne (dimension  $1 \times n_h$ ).

$\odot$  designe le produit terme-à-terme  $\text{sum}_{\text{line}}(\mathbf{X})$  avec  $X$  de taille  $N \times n_x$  fait une somme par ligne et retourne un vecteur colonne de taille  $N$ .  $\text{repmat}_{N \text{ lines}}(\mathbf{b})$  répète le vecteur-ligne  $\mathbf{b}$  (dimension  $1 \times p$ )  $N$  fois en ligne pour produire une matrice de taille  $N \times p$ .

### Forward

Elementwise

Vectoriel

Vectoriel par batch

$$\left\{ \begin{array}{l} \tilde{h}_i = \sum_{j=1}^{n_x} W_{j,i}^h x_j + b_i^h \\ h_i = \tanh(\tilde{h}_i) \\ \tilde{y}_i = \sum_{j=1}^{n_h} W_{j,i}^y h_j + b_i^y \\ \hat{y}_i = \text{SoftMax}(\tilde{y}_i) = \frac{e^{\tilde{y}_i}}{\sum_j e^{\tilde{y}_j}} \end{array} \right. \quad \left\{ \begin{array}{l} \tilde{\mathbf{h}} = \mathbf{x} \mathbf{W}^h + \mathbf{b}^h \\ \mathbf{h} = \tanh(\tilde{\mathbf{h}}) \\ \tilde{\mathbf{y}} = \mathbf{W}^y \mathbf{h} + \mathbf{b}^y \\ \hat{\mathbf{y}} = \text{SoftMax}(\tilde{\mathbf{y}}) \end{array} \right. \quad \left\{ \begin{array}{l} \tilde{\mathbf{H}} = \mathbf{X} \mathbf{W}^h + \text{repmat}_{N \text{ lines}}(\mathbf{b}_h) \\ \mathbf{H} = \tanh(\tilde{\mathbf{H}}) \\ \tilde{\mathbf{Y}} = \mathbf{H} \mathbf{W}^y + \text{repmat}_{N \text{ lines}}(\mathbf{b}_y) \\ \hat{\mathbf{Y}} = \text{SoftMax}_{\text{line}}(\tilde{\mathbf{Y}}) \end{array} \right.$$

### Loss

$$\left\{ \begin{array}{l} \ell(\mathbf{y}, \tilde{\mathbf{y}}) = - \sum_{i=1}^{n_y} y_i \log \hat{y}_i = - \sum_{i=1}^{n_y} y_i \tilde{y}_i + \log \sum_{j=1}^{n_y} e^{\tilde{y}_j} \\ \mathcal{L}(\mathbf{Y}, \hat{\mathbf{Y}}) = - \frac{1}{N} \sum_{k=1}^N \sum_{i=1}^{n_y} Y_{k,i} \log \hat{Y}_{k,i} = - \text{mean}_{\text{col}}(\text{sum}_{\text{line}}(\mathbf{Y} \odot \log \hat{\mathbf{Y}})) \end{array} \right.$$

### Backward

Elementwise

Vectoriel

Vectoriel par batch

$$\left\{ \begin{array}{l} \delta_i^y = \frac{\partial \ell}{\partial \tilde{y}_i} = \hat{y}_i - y_i \\ \frac{\partial \ell}{\partial W_{i,j}^y} = \delta_{y,i} h_j \\ \frac{\partial \ell}{\partial b_i^y} = \delta_i^y \\ \delta_i^h = \frac{\partial \ell}{\partial \tilde{h}_i} = (1 - h_i^2) \sum_{j=1}^{n_y} \delta_j^y W_{j,i}^y \\ \frac{\partial \ell}{\partial W_{h,i,j}} = \delta_i^h x_j \\ \frac{\partial \ell}{\partial b_{h,i}} = \delta_i^h \end{array} \right. \quad \left\{ \begin{array}{l} \nabla_{\tilde{\mathbf{y}}} = \hat{\mathbf{y}} - \mathbf{y} \\ \nabla_{\mathbf{W}^y} = \mathbf{h}^\top \nabla_{\tilde{\mathbf{y}}} \\ \nabla_{\mathbf{b}^y} = \nabla_{\tilde{\mathbf{y}}} \\ \nabla_{\tilde{\mathbf{h}}} = (\nabla_{\tilde{\mathbf{y}}} \mathbf{W}^{y\top}) \odot (1 - \mathbf{h}^2) \\ \nabla_{\mathbf{W}^h} = \mathbf{x}^\top \nabla_{\tilde{\mathbf{h}}} \\ \nabla_{\mathbf{b}^h} = \nabla_{\tilde{\mathbf{h}}} \end{array} \right. \quad \left\{ \begin{array}{l} \nabla_{\tilde{\mathbf{Y}}} = \hat{\mathbf{Y}} - \mathbf{Y} \\ \nabla_{\mathbf{W}^y} = \mathbf{H}^\top \nabla_{\tilde{\mathbf{Y}}} \\ \nabla_{\mathbf{b}^y} = \text{sum}_{\text{ligne}}(\nabla_{\tilde{\mathbf{Y}}}) \\ \nabla_{\tilde{\mathbf{H}}} = (\nabla_{\tilde{\mathbf{Y}}} \mathbf{W}^{y\top}) \odot (1 - \mathbf{H}^2) \\ \nabla_{\mathbf{W}^h} = \mathbf{X}^\top \nabla_{\tilde{\mathbf{H}}} \\ \nabla_{\mathbf{b}^h} = \text{sum}_{\text{col}}(\nabla_{\tilde{\mathbf{H}}})^\top \end{array} \right.$$