

Neural Network Learning basics

Richard Yi Da Xu
Yida.Xu@uts.edu.au
Wechat: aubedata

`www-staff.it.uts.edu.au/~ydxu/`

Director of Machine Learning and Data Analytics Lab::GBDTC:: UTS

February 18, 2018

Before we talk Deep Learning ...

Check the **regression** notes on:

- ▶ revise on **sigmoid** and **tanh** function
- ▶ revise on **logistic** and **softmax** regression
- ▶ Then we talk about neural networks and multilayer perceptron

Feedforward Neural Network in a nutshell

We begin Feedforward Neural networks, in a nutshell, comprised of the following steps:

- ▶ Feedforward

$$f(\mathbf{x}) = f_L(\dots f_2(f_1(\mathbf{x}; \theta_1); \theta_2) \dots), \theta_L)$$

- ▶ The objective is to minimize the overall cost:

$$L(\theta) = \frac{1}{n} \sum_{i=1}^n \ell(y_i, f(\mathbf{x}_i; \theta))$$

- ▶ To put it in a gradient descent framework, i.e.,

$$\theta^{t+1} = \theta^t - \eta_t \frac{\partial f}{\partial \theta}(\theta^t)$$

Something about $\frac{\partial f}{\partial \theta}(\theta^t)$

We know all about it already from high school mathematics:

- ▶ **Product rule:**

$$\frac{\partial f}{\partial \theta} (f(\theta) g(\theta)) = f(\theta) \frac{\partial}{\partial \theta} g(\theta) + \frac{\partial}{\partial \theta} f(\theta) g(\theta)$$

- ▶ **Derivative of sums:**

$$\frac{\partial f}{\partial \theta} (f(\theta) + g(\theta)) = \frac{\partial}{\partial \theta} f(\theta) + \frac{\partial}{\partial \theta} g(\theta)$$

- ▶ **Chain rule**

$$\begin{aligned} \frac{\partial f}{\partial \theta_l} &= \frac{\partial}{\partial \theta_l} f_L(\dots f_2(f_1(\mathbf{x}; \theta_1); \theta_2) \dots), \theta_L) \\ &= \frac{\partial f_L}{\partial f_{L-1}^\top} \frac{\partial f_{L-1}}{\partial f_{L-2}^\top} \dots \frac{\partial f_{l+2}}{\partial f_l^\top} \frac{\partial f_l}{\partial \theta_l^\top} \end{aligned}$$

We know all about it already from high school mathematics:

► **Product rule:**

$$\frac{\partial f}{\partial \theta} (f(\theta) g(\theta)) = f(\theta) \frac{\partial}{\partial \theta} g(\theta) + \frac{\partial}{\partial \theta} f(\theta) g(\theta)$$

► **Derivative of sums:**

$$\frac{\partial f}{\partial \theta} (f(\theta) + g(\theta)) = \frac{\partial}{\partial \theta} f(\theta) + \frac{\partial}{\partial \theta} g(\theta)$$

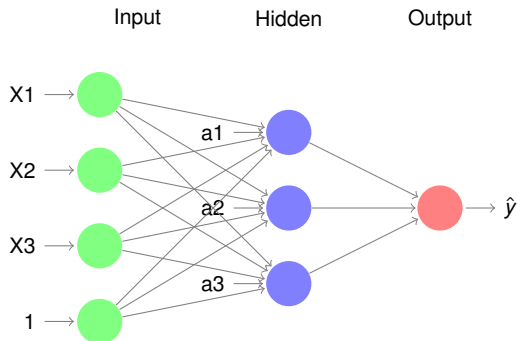
► **Chain rule**

$$\begin{aligned} \frac{\partial f}{\partial \theta_l} &= \frac{\partial}{\partial \theta_l} f_L(\dots f_2(f_1(\mathbf{x}; \theta_1); \theta_2) \dots), \theta_L) \\ &= \frac{\partial f_L}{\partial f_{L-1}^\top} \frac{\partial f_{L-1}}{\partial f_{L-2}^\top} \dots \frac{\partial f_{l+2}}{\partial f_l^\top} \frac{\partial f_l}{\partial \theta_l^\top} \end{aligned}$$

Look at Neural network systematically: Feed Forward (1)

$$\begin{aligned}\hat{y} &= U^T f(Wx + b) \\ &= U^T \underbrace{f(\underbrace{Wx + b}_z)}_a = U^T a\end{aligned}$$

let \hat{y} be a **scalar** score instead of a **softmax** this time.



$$\hat{y} = U^T f \left(\begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \right)$$

Look at Neural network systematically: Feed Forward (2)

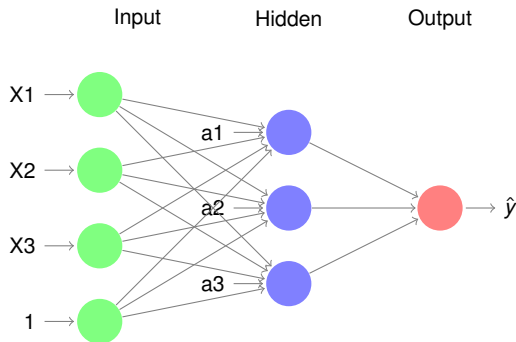
$$z_1 = W_{1,:}^T X + b_1 = \sum_i W_{1,i} X_i + b_1$$

$$z_2 = W_{2,:}^T X + b_2 = \sum_i W_{2,i} X_i + b_2$$

$$z_3 = W_{3,:}^T X + b_3 = \sum_i W_{3,i} X_i + b_3$$

Therefore:

$$W_{(\text{index of } a, \text{index of } X)}$$



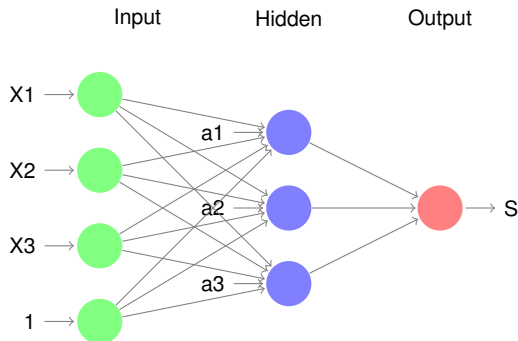
$$\hat{y} = U^T f \left(\begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \right)$$

Neural network: Backpropagation

$$\begin{aligned}\hat{y} &= U^T f(Wx + b) \\ &= U^T \underbrace{f(\underbrace{Wx + b}_z)}_a = U^T a\end{aligned}$$

Careful of their dimensions:

$$\begin{aligned}\frac{\partial \hat{y}}{\partial W} &= \underbrace{\frac{\partial \hat{y}}{\partial a} \frac{\partial a}{\partial z}}_{\text{column vector}} \times \underbrace{\frac{\partial z}{\partial W}}_{\text{row vector}} \\ &= \underbrace{(U \odot f'(z))}_{\text{column vector}} \times \underbrace{x}_{\text{row vector}}\end{aligned}$$



$$\hat{y} = U^T f \left(\begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \right)$$

Backpropagation for $W_{i,j}$

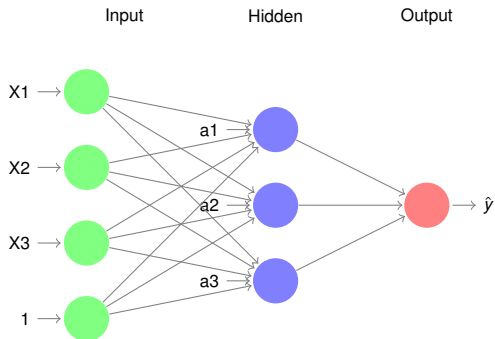
$$\hat{y} = U^T \underbrace{f(\underbrace{WX + b}_z)}_a = U^T a$$

If dimensionality of derivative of W is too hard to see, then we perform derivative one element $W_{i,j}$ at the time:

$$W_{(\text{index of } a, \text{index of } X)}$$

If we were to compute $\frac{\partial S}{\partial W_{i,j}}$:

$$\begin{aligned} \frac{\partial \hat{y}}{\partial W} &= \frac{\partial \hat{y}}{\partial a} \frac{\partial a}{\partial z} \frac{\partial z}{\partial W} \\ \Rightarrow \frac{\partial \hat{y}}{\partial W_{i,j}} &= \frac{\partial \hat{y}}{\partial a_i} \frac{\partial a_i}{\partial z_i} \frac{\partial z_i}{\partial W_{i,j}} \\ &= \underbrace{U_i f'(z_i)}_{\delta_i} X_j \\ &= \underbrace{U_i f'(W_{i,:} X + b_i)}_{\delta_i} X_j \end{aligned}$$



$$\hat{y} = U^T f \left(\begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \right)$$

important to remember:

$$z_i = W_{i,:} X + b_i = (WX + b)_i$$

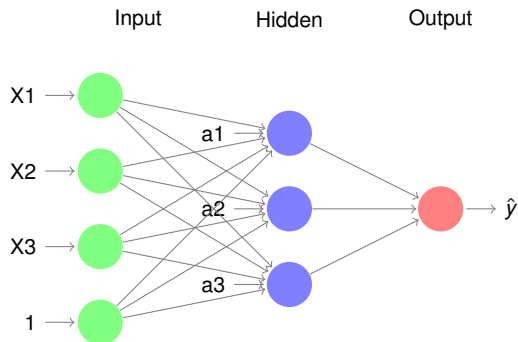
Backpropagation for W

If we were to compute $\frac{\partial \hat{y}}{\partial W_{i,j}}$:

$$\frac{\partial \hat{y}}{\partial W_{i,j}} = \underbrace{U_i f'(W_{i,:}^\top X + b_i)}_{\delta_i} X_j$$

$$\begin{aligned} \delta &= \begin{bmatrix} U_1 f'(W_{1,:}^\top X + b_1) \\ U_2 f'(W_{2,:}^\top X + b_2) \\ U_3 f'(W_{3,:}^\top X + b_3) \end{bmatrix} \\ &= U \odot f'(WX + B) \end{aligned}$$

$$\begin{aligned} \frac{\partial \hat{y}}{\partial W} &= \begin{bmatrix} \delta_1 X_1 & \delta_1 X_2 & \delta_1 X_3 \\ \delta_2 X_1 & \delta_2 X_2 & \delta_2 X_3 \\ \delta_3 X_1 & \delta_3 X_2 & \delta_3 X_3 \end{bmatrix} \\ &= \delta X^\top \end{aligned}$$



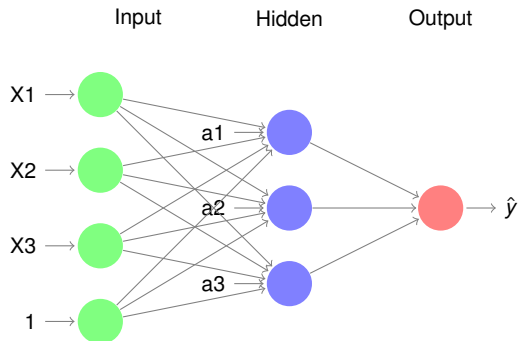
$$\hat{y} = U^\top f \left(\begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \right)$$

Something about δ

$$\delta = \begin{bmatrix} U_1 f'(W_{1,\cdot}^\top X + b_1) \\ U_2 f'(W_{2,\cdot}^\top X + b_2) \\ U_3 f'(W_{3,\cdot}^\top X + b_3) \end{bmatrix}$$
$$= U \odot f'(WX + B)$$

$$\frac{\partial \hat{y}}{\partial W} = \begin{bmatrix} \delta_1 X_1 & \delta_1 X_2 & \delta_1 X_3 \\ \delta_2 X_1 & \delta_2 X_2 & \delta_2 X_3 \\ \delta_3 X_1 & \delta_3 X_2 & \delta_3 X_3 \end{bmatrix}$$
$$= \delta X^\top$$

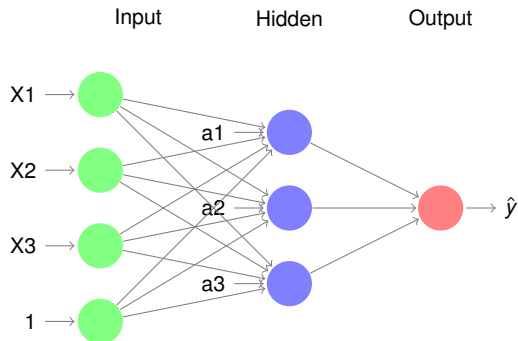
- ▶ δ is the error signal, i.e., $\frac{\partial \hat{y}}{\partial z}$
- ▶ δ involves the derivatives of all the activation function $\{a_i = f(z_i)\}$



$$\hat{y} = U^\top f \left(\begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \right)$$

Backpropagation for b

$$\frac{\partial \hat{y}}{\partial b_i} = \underbrace{U_i f'(W_{i,:}^\top X + b_i)}_{\delta_i} 1$$
$$= \delta_i$$

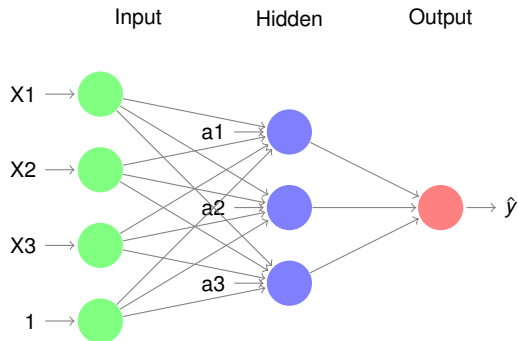


$$\hat{y} = U^\top f \left(\begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \right)$$

Backpropagation for x_j

Note each x_j is contributed by all $\{a_i\}$

$$\begin{aligned}\frac{\partial \hat{y}}{\partial x_j} &= \sum_{i=1}^3 \frac{\partial \hat{y}}{\partial a_i} \frac{\partial a_i}{\partial x_j} \\&= \sum_{i=1}^3 \frac{\partial U^\top a}{\partial a_i} \frac{\partial a_i}{\partial x_j} \\&= \sum_{i=1}^3 U_i \frac{\partial f(W_{i,:}x + b)}{\partial x_j} \\&= \sum_{i=1}^3 U_i \underbrace{f'(W_{i,:}x + b)}_{\delta_i} \frac{\partial W_{i,:}x}{\partial x_j} \\&= \sum_{i=1}^3 \delta_i W_{i,j} = \delta^\top W_{:,j}\end{aligned}$$



$$\hat{y} = U^\top f \left(\begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \right)$$

Backpropagation for two layers with respect to $W^{(2)}$

$$a^{(0)} = x \quad (\text{input})$$

$$z^{(1)} = W^{(1)} a^{(0)} + b^{(1)} \quad (\text{linear})$$

$$a^{(1)} = f(z^{(1)}) \quad (\text{non-linear})$$

$$z^{(2)} = W^{(2)} a^{(1)} + b^{(2)} \quad (\text{linear})$$

$$a^{(2)} = f(z^{(2)}) \quad (\text{non-linear})$$

$$\hat{y} = U^T a^{(2)} \quad (\text{output})$$

$$\hat{y} = U^T f(W^{(2)} f(W^{(1)} x + b^{(1)}) + b^{(2)})$$

$$\frac{\partial \hat{y}}{\partial W_{i,j}} = \underbrace{U_i f'(z_i)}_{\delta_i} X_j \quad (\text{one layer case})$$

$$\Rightarrow \frac{\partial \hat{y}}{\partial W_{i,j}^{(2)}} = \underbrace{U_i f'(z_i^{(2)})}_{\delta_i^{(2)}} a_j^{(2)}$$

$$\Rightarrow \frac{\partial \hat{y}}{\partial W^{(2)}} = \delta^{(2)} a^{(2)T} \quad \text{where } \delta^{(2)} = U \odot f'(z^{(2)})$$

Backpropagation for two layers with respect to $W^{(1)}$

$$a^{(0)} = x \quad (\text{input})$$

$$z^{(1)} = W^{(1)} a^{(0)} + b^{(1)} \quad (\text{linear})$$

$$a^{(1)} = f(z^{(1)}) \quad (\text{non-linear})$$

$$z^{(2)} = W^{(2)} a^{(1)} + b^{(2)} \quad (\text{linear})$$

$$a^{(2)} = f(z^{(2)}) \quad (\text{non-linear})$$

$$\hat{y} = U^T a^{(2)} \quad (\text{output})$$

$$\begin{aligned} \hat{y} &= U^T f(W^{(2)} f(W^{(1)} x + b^{(1)}) + b^{(2)}) \\ &= U^T f\left(W^{(2)} f\left(\underbrace{W^{(1)} x + b^{(1)}}_{z^{(1)}}\right) + b^{(2)}\right) \\ &\quad \underbrace{\hspace{10em}}_{z^{(2)}} \end{aligned}$$

$$\begin{aligned} \frac{\partial \hat{y}}{\partial W^{(1)}} &= \underbrace{U^T \left(W^{(2)} f(W^{(1)} x + b^{(1)}) + b^{(2)} \right)}_{\frac{\partial \hat{y}}{\partial z^{(2)}} \frac{\partial z^{(2)}}{\partial a^{(1)}}} \underbrace{W^{(2)}}_{\frac{\partial z^{(2)}}{\partial a^{(1)}}} \underbrace{f'(W^{(1)} x + b^{(1)})}_{\frac{\partial a^{(1)}}{\partial z^{(1)}}} \underbrace{x}_{\frac{\partial z^{(1)}}{\partial W^{(1)}}} \\ &= \underbrace{\frac{\partial \hat{y}}{\partial a^{(2)}} \frac{\partial a^{(2)}}{\partial z^{(2)}} \frac{\partial z^{(2)}}{\partial a^{(1)}} \frac{\partial a^{(1)}}{\partial z^{(1)}} \frac{\partial z^{(1)}}{\partial W^{(1)}}}_{\delta^{(1)}} \\ &= \underbrace{\delta^{(2)} W^{(2)} f'(z^{(2)})}_{\delta^{(1)}} x \end{aligned}$$

Putting them in the “correct” form of the matrix operations and generalise:

$$\begin{aligned} &= \underbrace{\left(W^{(2)\top} \delta^{(2)} \right) \odot f'(z^{(2)})}_{\delta^{(1)}} X^\top \\ \implies \delta^{(1)} &= \left(W^{(2)\top} \delta^{(2)} \right) \odot f'(z^{(2)}) \\ \implies \delta^{(l)} &= \left(W^{(l)\top} \delta^{(l+1)} \right) \odot f'(z^{(l)}) \end{aligned}$$

Backpropagation in action!

$$a^{(0)} = x \quad (\text{input})$$

$$z^{(1)} = W^{(1)} a^{(0)} + b^{(1)} \quad (\text{linear})$$

$$a^{(1)} = f(z^{(1)}) \quad (\text{non-linear})$$

$$z^{(2)} = W^{(2)} a^{(1)} + b^{(2)} \quad (\text{linear})$$

$$a^{(2)} = f(z^{(2)}) \quad (\text{non-linear})$$

$$\hat{y} = U^T a^{(2)} \quad (\text{output})$$

To generalise this:

$$\delta^{(l)} = \left(W^{(l+1)T} \delta^{(l+1)} \right) \odot f'(z^{(l)})$$

- ▶ step 1: compute δ of the last layer L :

$$\delta^{(L)} = U \odot \underbrace{f'(z^{(L)})}_{\text{from feed-forward}}$$

- ▶ step 2: Generate the whole sequence of $\{\delta^{(l)}\}_1^L$

$$\delta^{(l)} = \left(W^{(l+1)T} \delta^{(l+1)} \right) \odot f'(z^{(l)})$$

- ▶ step 3: compute gradients at each layers $\left\{ \frac{\partial s}{\partial w^{(l)}} \right\}_1^{(L-1)}$:

$$\frac{\partial \hat{y}}{\partial W^{(l)}} = \delta^{(l+1)} a^{(l)T}$$

Note that $\frac{\partial \hat{y}}{\partial w^{(l)}}$ can be obtained as soon as $\delta^{(l+1)}$ becomes available.