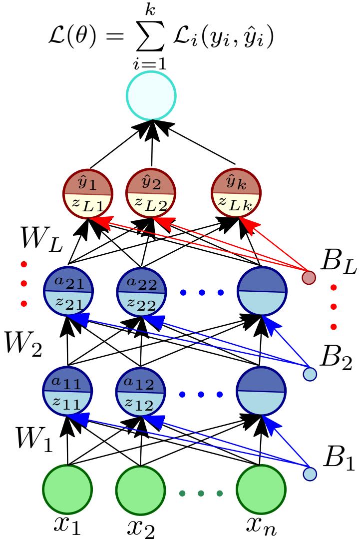
Introduction to Neural Network

Slide Courtesy: Dr. Soumi Chattopadhyay

Indian Institute of Technology Indore

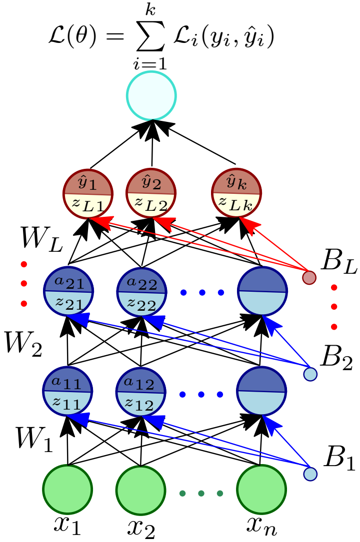
October 26, 2024

Feedforward Neural Network

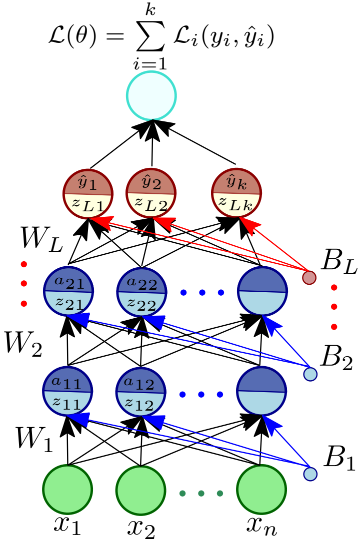


- Input sample: $X_i = [x_1 \ x_2 \ \dots \ x_n] \in \mathbb{R}^n$

Feedforward Neural Network



- Input sample: $X_i = [x_1 \ x_2 \ \ldots \ x_n] \in \mathbb{R}^n$
- Each hidden neuron
  - $z_i = W_i \ . \ a_{i-1} + B_i$

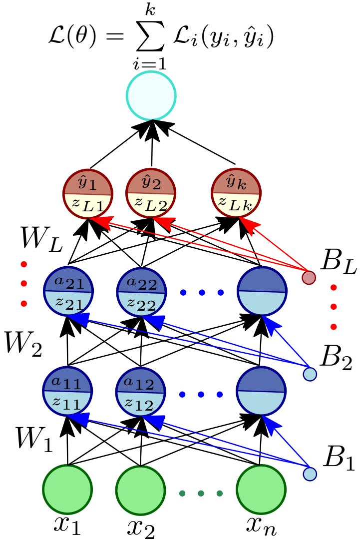


- Input sample: $X_i = [x_1 \ x_2 \ \ldots \ x_n] \in \mathbb{R}^n$
- Each hidden neuron
  - $z_i = W_i \ . \ a_{i-1} + B_i$

Example

$$z_2 = \begin{bmatrix} z_{21} \\ z_{22} \\ z_{23} \end{bmatrix} = \begin{bmatrix} W_{211} & W_{212} & W_{213} \\ W_{221} & W_{222} & W_{223} \\ W_{231} & W_{232} & W_{233} \end{bmatrix} \begin{bmatrix} a_{11} \\ a_{12} \\ a_{13} \end{bmatrix} + \begin{bmatrix} b_{21} \\ b_{22} \\ b_{23} \end{bmatrix}$$

Feedforward Neural Network



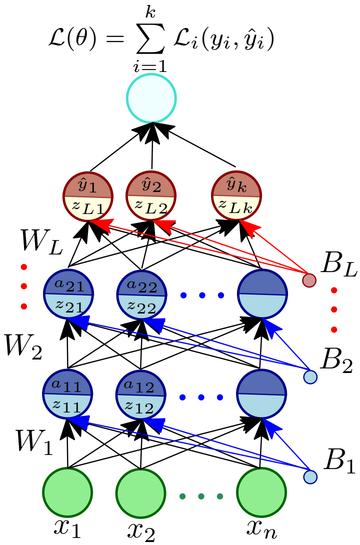
- Input sample: $X_i = [x_1 \ x_2 \ \dots \ x_n] \in \mathbb{R}^n$
- Each hidden neuron
  - $z_i = W_i \ . \ a_{i-1} + B_i$

Example

$$z_2 = \begin{bmatrix} z_{21} \\ z_{22} \\ z_{23} \end{bmatrix} = \begin{bmatrix} W_{211} & W_{212} & W_{213} \\ W_{221} & W_{222} & W_{223} \\ W_{231} & W_{232} & W_{233} \end{bmatrix} \begin{bmatrix} a_{11} \\ a_{12} \\ a_{13} \end{bmatrix} + \begin{bmatrix} b_{21} \\ b_{22} \\ b_{23} \end{bmatrix}$$

  - $a_i = g(z_i)$

Feedforward Neural Network



- Input sample: $X_i = [x_1 \ x_2 \ \dots \ x_n] \in \mathbb{R}^n$
- Each hidden neuron
  - $z_i = W_i \ . \ a_{i-1} + B_i$

Example

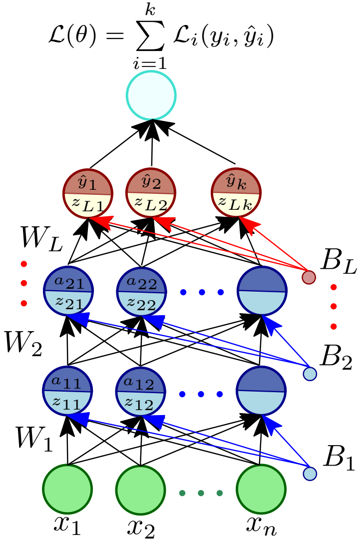
$$z_2 = \begin{bmatrix} z_{21} \\ z_{22} \\ z_{23} \end{bmatrix} = \begin{bmatrix} W_{211} & W_{212} & W_{213} \\ W_{221} & W_{222} & W_{223} \\ W_{231} & W_{232} & W_{233} \end{bmatrix} \begin{bmatrix} a_{11} \\ a_{12} \\ a_{13} \end{bmatrix} + \begin{bmatrix} b_{21} \\ b_{22} \\ b_{23} \end{bmatrix}$$

  - $a_i = g(z_i)$

Example

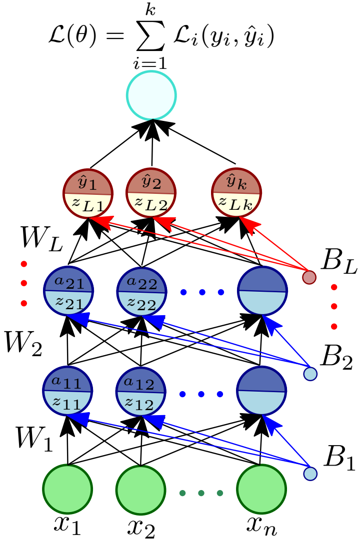
$$a_2 = \begin{bmatrix} a_{21} \\ a_{22} \\ a_{23} \end{bmatrix} = \begin{bmatrix} g(z_{21}) \\ g(z_{22}) \\ g(z_{23}) \end{bmatrix}$$

Feedforward Neural Network



- Each output neuron
  - $z_L = W_L \; . \; a_{L-1} + B_L$
  - $\hat{y} = O(z_L)$

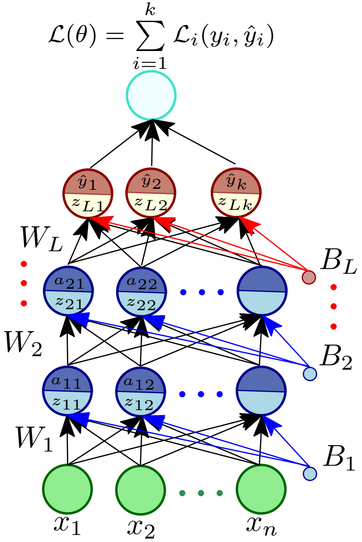
Feedforward Neural Network



- Each output neuron
  - $z_L = W_L \ . \ a_{L-1} + B_L$
  - $\hat{y} = O(z_L)$

Example

$$\hat{y} = \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_k \end{bmatrix} = \begin{bmatrix} O(z_{L1}) \\ O(z_{L2}) \\ \vdots \\ O(z_{Lk}) \end{bmatrix}$$



- Each output neuron
  - $z_L = W_L \, . \, a_{L-1} + B_L$
  - $\hat{y} = O(z_L)$

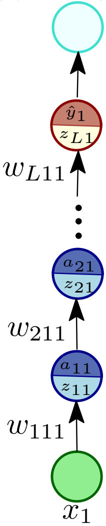
Example

$$\hat{y} = \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_k \end{bmatrix} = \begin{bmatrix} O(z_{L1}) \\ O(z_{L2}) \\ \vdots \\ O(z_{Lk}) \end{bmatrix}$$

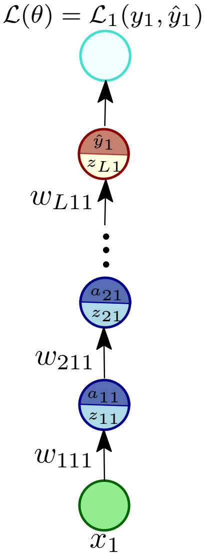
- Compute loss $\mathcal{L}(\theta) = \sum_{i=1}^{k} \mathcal{L}_i(y_i, \hat{y}_i)$

Backpropagation

$\mathcal{L}(\theta) = \mathcal{L}_1(y_1, \hat{y}_1)$

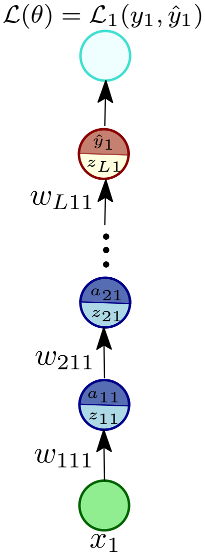


Backpropagation



$$\frac{\partial \mathcal{L}(\theta)}{\partial w_{111}} =$$

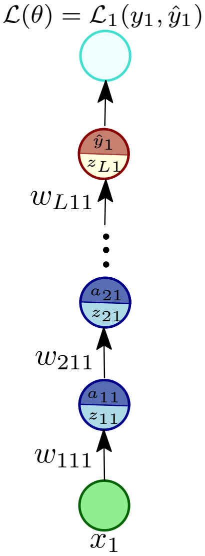
Backpropagation



$$\frac{\partial \mathcal{L}(\theta)}{\partial w_{111}} =$$

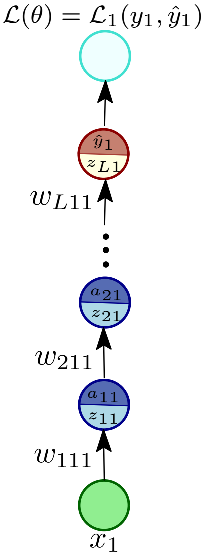
$$\underbrace{\left(\frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_1}\frac{\partial \hat{y}_1}{\partial z_{L1}}\right)}_{\text{PD wrt output neurons}}$$

Backpropagation



$$\frac{\partial \mathcal{L}(\theta)}{\partial w_{111}} =$$

$$\underbrace{\left(\frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_1}\frac{\partial \hat{y}_1}{\partial z_{L1}}\right)}_{\text{PD wrt output neurons}} \quad \underbrace{\left(\frac{\partial z_{L1}}{\partial a_{(L-1)1}}\frac{\partial a_{(L-1)1}}{\partial z_{(L-1)1}}\right)}_{\text{PD wrt hidden neurons}}$$

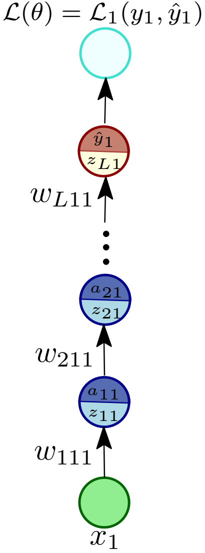


$$\frac{\partial \mathcal{L}(\theta)}{\partial w_{111}} =$$

$$\underbrace{\left(\frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_1} \frac{\partial \hat{y}_1}{\partial z_{L1}}\right)}_{\text{PD wrt output neurons}} \underbrace{\left(\frac{\partial z_{L1}}{\partial a_{(L-1)1}} \frac{\partial a_{(L-1)1}}{\partial z_{(L-1)1}}\right)}_{\text{PD wrt hidden neurons}}$$

$$\ldots \underbrace{\left(\frac{\partial z_{31}}{\partial a_{21}} \frac{\partial a_{21}}{\partial z_{21}}\right)}_{\text{PD wrt hidden neurons}} \underbrace{\left(\frac{\partial z_{21}}{\partial a_{11}} \frac{\partial a_{11}}{\partial z_{11}}\right)}_{\text{PD wrt hidden neurons}}$$

Backpropagation



$$\frac{\partial \mathcal{L}(\theta)}{\partial w_{111}} =$$

$$\underbrace{\left(\frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_1} \frac{\partial \hat{y}_1}{\partial z_{L1}}\right)}_{\text{PD wrt output neurons}} \quad \underbrace{\left(\frac{\partial z_{L1}}{\partial a_{(L-1)1}} \frac{\partial a_{(L-1)1}}{\partial z_{(L-1)1}}\right)}_{\text{PD wrt hidden neurons}}$$

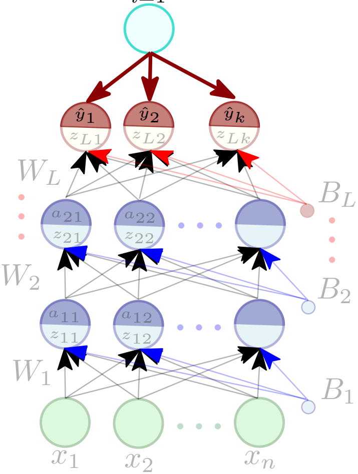
$$\ldots \quad \underbrace{\left(\frac{\partial z_{31}}{\partial a_{21}} \frac{\partial a_{21}}{\partial z_{21}}\right)}_{\text{PD wrt hidden neurons}} \quad \underbrace{\left(\frac{\partial z_{21}}{\partial a_{11}} \frac{\partial a_{11}}{\partial z_{11}}\right)}_{\text{PD wrt hidden neurons}}$$

$$\underbrace{\left(\frac{\partial z_{11}}{\partial w_{111}}\right)}_{\text{PD wrt weight}}$$

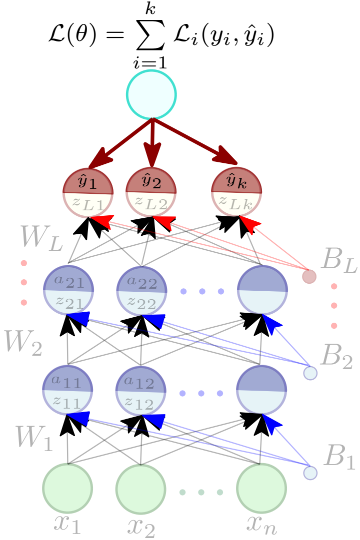
PD: Partial derivative

Backpropagation: Gradient with respect to Output Neurons

$$\mathcal{L}(\theta) = \sum_{i=1}^{k} \mathcal{L}_i(y_i, \hat{y}_i)$$

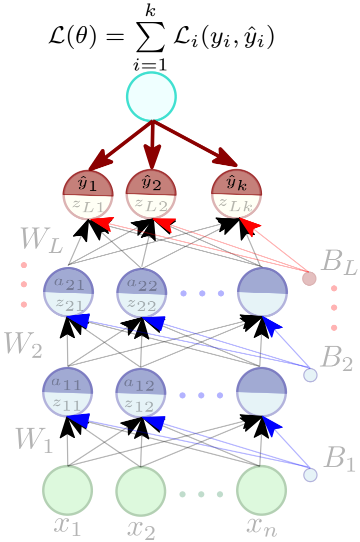


Backpropagation: Gradient with respect to Output Neurons



For classification problem
Output function: Softmax;
Loss function: Cross-entropy

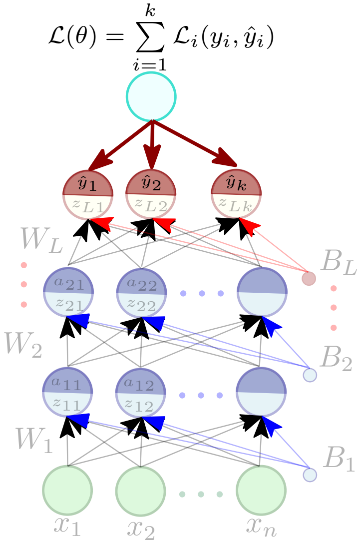
Backpropagation: Gradient with respect to Output Neurons



For classification problem
Output function: Softmax;
Loss function: Cross-entropy

$$\mathcal{L}(\theta) = \sum_{i=1}^{k} y_i(-\log(\hat{y}_i)) =$$

Backpropagation: Gradient with respect to Output Neurons



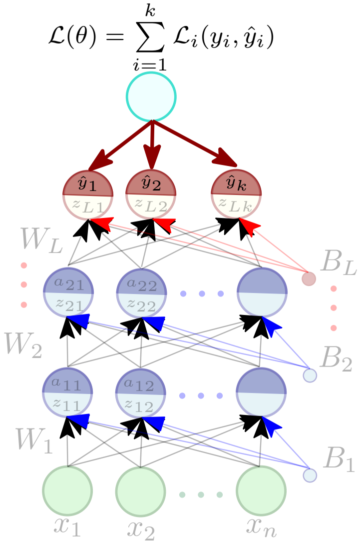
For classification problem
Output function: Softmax;
Loss function: Cross-entropy

$$\mathcal{L}(\theta) = \sum_{i=1}^{k} y_i(-\log(\hat{y}_i)) = -\log(\hat{y}_c)$$

[c is the actual class level of the sample]

$$\frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_i} =$$

Backpropagation: Gradient with respect to Output Neurons



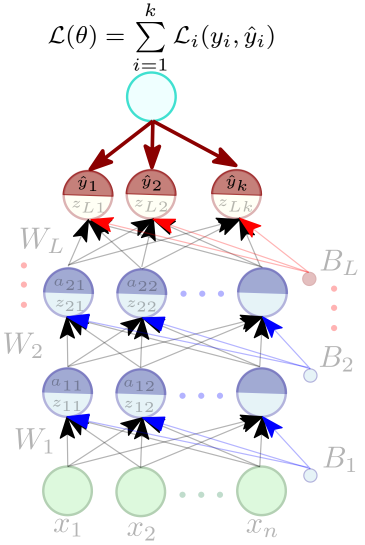
For classification problem
Output function: Softmax;
Loss function: Cross-entropy

$$\mathcal{L}(\theta) = \sum_{i=1}^{k} y_i(-\log(\hat{y}_i)) = -\log(\hat{y}_c)$$

[c is the actual class level of the sample]

$$\frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_i} = \frac{\partial(-\log(\hat{y}_c))}{\partial \hat{y}_i} =$$

Backpropagation: Gradient with respect to Output Neurons



For classification problem
Output function: Softmax;
Loss function: Cross-entropy

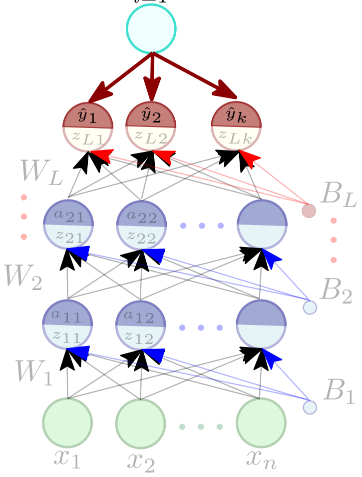
$$\mathcal{L}(\theta) = \sum_{i=1}^{k} y_i(-\log(\hat{y}_i)) = -\log(\hat{y}_c)$$

[c is the actual class level of the sample]

$$\frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_i} = \frac{\partial(-\log(\hat{y}_c))}{\partial \hat{y}_i} = \begin{cases} -\frac{1}{\hat{y}_c} & \textbf{if } i = c \\ 0 & \textbf{otherwise} \end{cases}$$

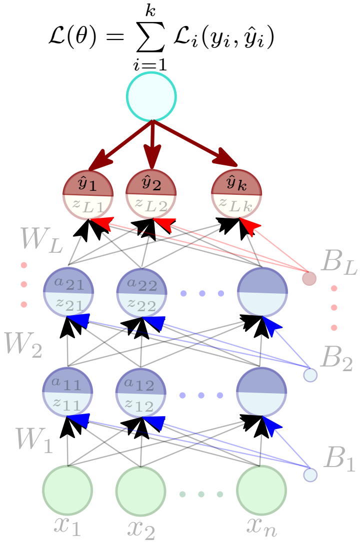
Backpropagation: Gradient with respect to Output Neurons

$$\mathcal{L}(\theta) = \sum_{i=1}^{k} \mathcal{L}_i(y_i, \hat{y}_i)$$



$$\frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_i} = \begin{cases} -\frac{1}{\hat{y}_c} & \textbf{if } i = c \\ 0 & \textbf{otherwise} \end{cases} = -\frac{\mathbb{1}_{c=i}}{\hat{y}_c}$$

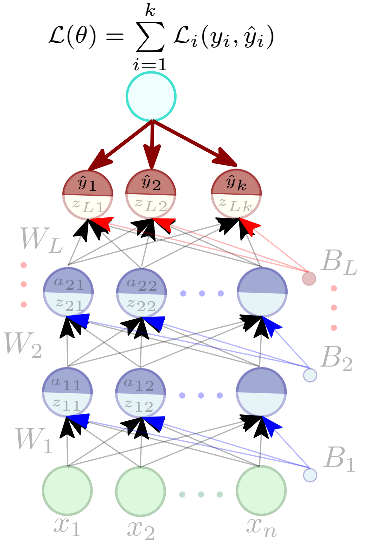
Backpropagation: Gradient with respect to Output Neurons



$$\frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_i} = \begin{cases} -\frac{1}{\hat{y}_c} & \textbf{if } i = c \\ 0 & \textbf{otherwise} \end{cases} = -\frac{\mathbb{1}_{c=i}}{\hat{y}_c}$$

$$\nabla_{\hat{y}} \mathcal{L}(\theta) =$$

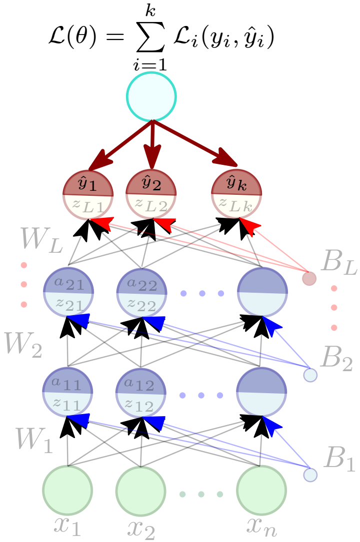
Backpropagation: Gradient with respect to Output Neurons



$$\frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_i} = \begin{cases} -\frac{1}{\hat{y}_c} & \textbf{if } i = c \\ 0 & \textbf{otherwise} \end{cases} = -\frac{\mathbb{1}_{c=i}}{\hat{y}_c}$$

$$\nabla_{\hat{y}} \mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_1} \\ \frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_2} \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_k} \end{bmatrix} =$$

Backpropagation: Gradient with respect to Output Neurons

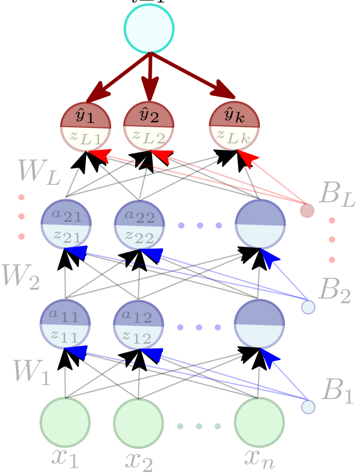


$$\frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_i} = \begin{cases} -\frac{1}{\hat{y}_c} & \textbf{if } i = c \\ 0 & \textbf{otherwise} \end{cases} = -\frac{\mathbb{1}_{c=i}}{\hat{y}_c}$$

$$\nabla_{\hat{y}} \mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_1} \\ \frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_2} \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_k} \end{bmatrix} = -\frac{1}{\hat{y}_c} \begin{bmatrix} \mathbb{1}_{c=1} \\ \mathbb{1}_{c=2} \\ \vdots \\ \mathbb{1}_{c=k} \end{bmatrix} =$$

Backpropagation: Gradient with respect to Output Neurons

$$\mathcal{L}(\theta) = \sum_{i=1}^{k} \mathcal{L}_i(y_i, \hat{y}_i)$$

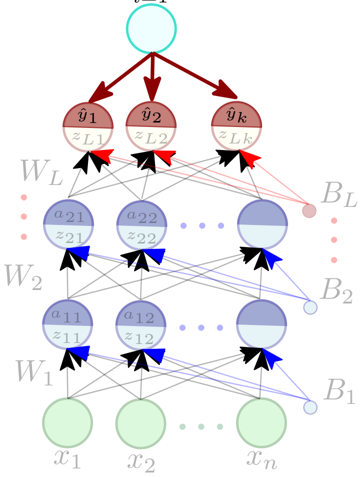


$$\frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_i} = \begin{cases} -\frac{1}{\hat{y}_c} & \textbf{if } i = c \\ 0 & \textbf{otherwise} \end{cases} = -\frac{\mathbb{1}_{c=i}}{\hat{y}_c}$$

$$\nabla_{\hat{y}} \mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_1} \\ \frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_2} \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_k} \end{bmatrix} = -\frac{1}{\hat{y}_c} \begin{bmatrix} \mathbb{1}_{c=1} \\ \mathbb{1}_{c=2} \\ \vdots \\ \mathbb{1}_{c=k} \end{bmatrix} = -\frac{1}{\hat{y}_c} \mathbb{I}(c)$$

Backpropagation: Gradient with respect to Output Neurons

$$\mathcal{L}(\theta) = \sum_{i=1}^{k} \mathcal{L}_i(y_i, \hat{y}_i)$$



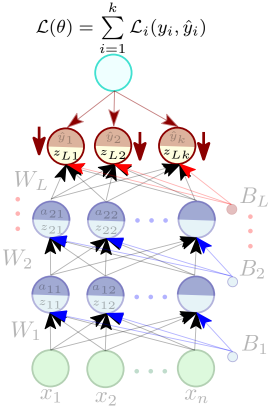
$$\frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_i} = \begin{cases} -\frac{1}{\hat{y}_c} & \textbf{if } i = c \\ 0 & \textbf{otherwise} \end{cases} = -\frac{\mathbb{1}_{c=i}}{\hat{y}_c}$$

$$\nabla_{\hat{y}} \mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_1} \\ \frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_2} \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_k} \end{bmatrix} = -\frac{1}{\hat{y}_c} \begin{bmatrix} \mathbb{1}_{c=1} \\ \mathbb{1}_{c=2} \\ \vdots \\ \mathbb{1}_{c=k} \end{bmatrix} = -\frac{1}{\hat{y}_c} \mathbb{I}(c)$$

$\mathbb{I}$ is a k-dimensional one hot vector with c^{th} entry as 1.

$$\nabla_{\hat{y}} \mathcal{L}(\theta) = -\frac{1}{\hat{y}_c} \mathbb{I}(c)$$

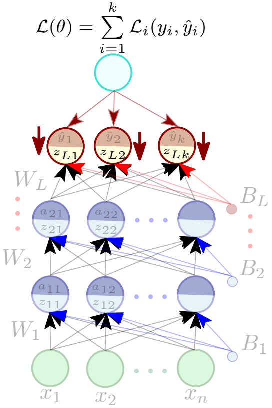
Backpropagation: Gradient with respect to Output Neurons



$$\frac{\partial \mathcal{L}(\theta)}{\partial z_{Li}} = \underbrace{\frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_c}}_{\text{done}}$$

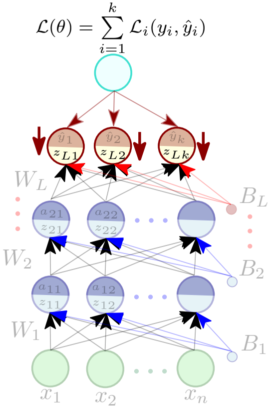
Backpropagation: Gradient with respect to Output Neurons

$$\frac{\partial \mathcal{L}(\theta)}{\partial z_{Li}} = \underbrace{\frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_c}}_{\text{done}} \frac{\partial \hat{y}_c}{\partial z_{Li}}$$



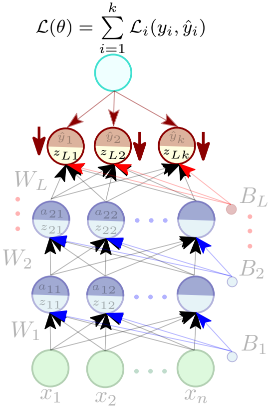
Backpropagation: Gradient with respect to Output Neurons

$$\frac{\partial \mathcal{L}(\theta)}{\partial z_{Li}} = \underbrace{\frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_c}}_{\text{done}} \frac{\partial \hat{y}_c}{\partial z_{Li}}$$



$$\frac{\partial \hat{y}_c}{\partial z_{Li}}$$

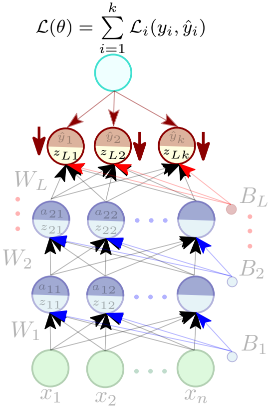
Backpropagation: Gradient with respect to Output Neurons



$$\frac{\partial \mathcal{L}(\theta)}{\partial z_{Li}} = \underbrace{\frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_c}}_{\text{done}} \frac{\partial \hat{y}_c}{\partial z_{Li}}$$

$$\frac{\partial \hat{y}_c}{\partial z_{Li}} = \frac{\partial}{\partial z_{Li}} Softmax(z_{Lc})$$

Backpropagation: Gradient with respect to Output Neurons



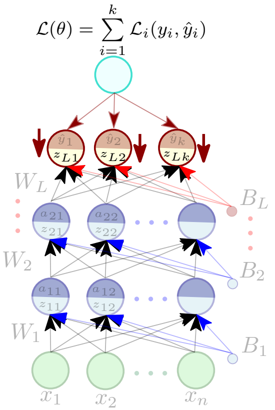
$$\frac{\partial \mathcal{L}(\theta)}{\partial z_{Li}} = \underbrace{\frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_c}}_{\text{done}} \frac{\partial \hat{y}_c}{\partial z_{Li}}$$

$$\frac{\partial \hat{y}_c}{\partial z_{Li}} = \frac{\partial}{\partial z_{Li}} Softmax(z_{Lc})$$

$$= \frac{\partial}{\partial z_{Li}} \left(\frac{exp(z_{Lc})}{\sum_{j=1}^{k} exp(z_{Lj})} \right)$$

Backpropagation: Gradient with respect to Output Neurons

$$\frac{\partial \mathcal{L}(\theta)}{\partial z_{Li}} = \underbrace{\frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_c}}_{\text{done}} \frac{\partial \hat{y}_c}{\partial z_{Li}}$$

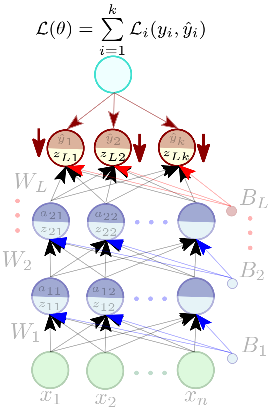


$$\frac{\partial \hat{y}_c}{\partial z_{Li}} = \frac{\partial}{\partial z_{Li}} Softmax(z_{Lc})$$

$$= \frac{\partial}{\partial z_{Li}} \left(\frac{exp(z_{Lc})}{\sum_{j=1}^{k} exp(z_{Lj})} \right)$$

$$= \frac{\left(\sum_{j=1}^{k} exp(z_{Lj})\right) \frac{\partial}{\partial z_{Li}} (exp(z_{Lc})) - exp(z_{Lc}) \frac{\partial}{\partial z_{Li}} \left(\sum_{j=1}^{k} exp(z_{Lj})\right)}{\left(\sum_{j=1}^{k} exp(z_{Lj})\right)^2}$$

Backpropagation: Gradient with respect to Output Neurons



$$\frac{\partial \mathcal{L}(\theta)}{\partial z_{Li}} = \underbrace{\frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_c}}_{\text{done}} \frac{\partial \hat{y}_c}{\partial z_{Li}}$$

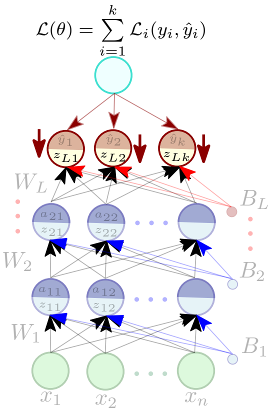
$$\frac{\partial \hat{y}_c}{\partial z_{Li}} = \frac{\partial}{\partial z_{Li}} Softmax(z_{Lc})$$

$$= \frac{\partial}{\partial z_{Li}} \left(\frac{exp(z_{Lc})}{\sum_{j=1}^{k} exp(z_{Lj})} \right)$$

$$= \frac{\left(\sum_{j=1}^{k} exp(z_{Lj})\right) \frac{\partial}{\partial z_{Li}} (exp(z_{Lc})) - exp(z_{Lc}) \frac{\partial}{\partial z_{Li}} \left(\sum_{j=1}^{k} exp(z_{Lj})\right)}{\left(\sum_{j=1}^{k} exp(z_{Lj})\right)^2}$$

$$= \frac{\left(\sum_{j=1}^{k} exp(z_{Lj})\right) \mathbb{1}_{c=i} \; exp(z_{Lc}) - exp(z_{Lc}) exp(z_{Li})}{\left(\sum_{j=1}^{k} exp(z_{Lj})\right)^2}$$

Backpropagation: Gradient with respect to Output Neurons



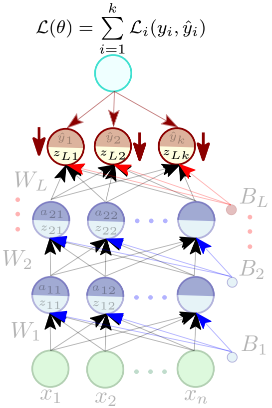
$$\frac{\partial \hat{y}_c}{\partial z_{Li}} = \frac{\partial}{\partial z_{Li}} Softmax(z_{Lc}) = \frac{\partial}{\partial z_{Li}} \left(\frac{exp(z_{Lc})}{\sum_{j=1}^{k} exp(z_{Lj})} \right)$$

$$= \frac{\left(\sum_{j=1}^{k} exp(z_{Lj})\right) \frac{\partial}{\partial z_{Li}} (exp(z_{Lc})) - exp(z_{Lc}) \frac{\partial}{\partial z_{Li}} \left(\sum_{j=1}^{k} exp(z_{Lj})\right)}{\left(\sum_{j=1}^{k} exp(z_{Lj})\right)^2}$$

$$= \frac{\left(\sum_{j=1}^{k} exp(z_{Lj})\right) \mathbb{1}_{c=i} \; exp(z_{Lc}) - exp(z_{Lc}) exp(z_{Li})}{\left(\sum_{j=1}^{k} exp(z_{Lj})\right)^2}$$

$$= \frac{\mathbb{1}_{c=i} \; exp(z_{Lc})}{\left(\sum_{j=1}^{k} exp(z_{Lj})\right)} - \frac{exp(z_{Lc})}{\left(\sum_{j=1}^{k} exp(z_{Lj})\right)} \frac{exp(z_{Li})}{\left(\sum_{j=1}^{k} exp(z_{Lj})\right)}$$

Backpropagation: Gradient with respect to Output Neurons



$$\frac{\partial \hat{y}_c}{\partial z_{Li}} = \frac{\partial}{\partial z_{Li}} Softmax(z_{Lc}) = \frac{\partial}{\partial z_{Li}} \left(\frac{exp(z_{Lc})}{\sum_{j=1}^{k} exp(z_{Lj})} \right)$$

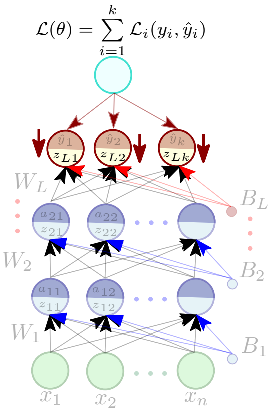
$$= \frac{\left(\sum_{j=1}^{k} exp(z_{Lj})\right) \frac{\partial}{\partial z_{Li}} (exp(z_{Lc})) - exp(z_{Lc}) \frac{\partial}{\partial z_{Li}} \left(\sum_{j=1}^{k} exp(z_{Lj})\right)}{\left(\sum_{j=1}^{k} exp(z_{Lj})\right)^2}$$

$$= \frac{\left(\sum_{j=1}^{k} exp(z_{Lj})\right) \mathbb{1}_{c=i} \; exp(z_{Lc}) - exp(z_{Lc}) exp(z_{Li})}{\left(\sum_{j=1}^{k} exp(z_{Lj})\right)^2}$$

$$= \frac{\mathbb{1}_{c=i} \; exp(z_{Lc})}{\left(\sum_{j=1}^{k} exp(z_{Lj})\right)} - \frac{exp(z_{Lc})}{\left(\sum_{j=1}^{k} exp(z_{Lj})\right)} \frac{exp(z_{Li})}{\left(\sum_{j=1}^{k} exp(z_{Lj})\right)}$$

$$= \mathbb{1}_{c=i} Softmax(z_{Lc}) - Softmax(z_{Lc}) Softmax(z_{Li})$$

Backpropagation: Gradient with respect to Output Neurons



$$\frac{\partial \hat{y}_c}{\partial z_{Li}} = \frac{\partial}{\partial z_{Li}} Softmax(z_{Lc}) = \frac{\partial}{\partial z_{Li}} \left(\frac{exp(z_{Lc})}{\sum_{j=1}^{k} exp(z_{Lj})} \right)$$

$$= \frac{\left(\sum_{j=1}^{k} exp(z_{Lj})\right) \frac{\partial}{\partial z_{Li}} (exp(z_{Lc})) - exp(z_{Lc}) \frac{\partial}{\partial z_{Li}} \left(\sum_{j=1}^{k} exp(z_{Lj})\right)}{\left(\sum_{j=1}^{k} exp(z_{Lj})\right)^2}$$

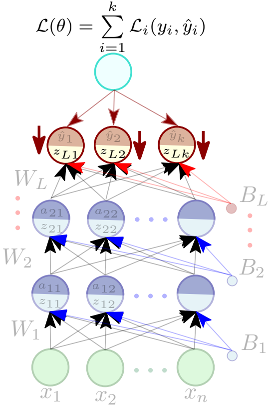
$$= \frac{\left(\sum_{j=1}^{k} exp(z_{Lj})\right) \mathbb{1}_{c=i}\ exp(z_{Lc}) - exp(z_{Lc}) exp(z_{Li})}{\left(\sum_{j=1}^{k} exp(z_{Lj})\right)^2}$$

$$= \frac{\mathbb{1}_{c=i}\ exp(z_{Lc})}{\left(\sum_{j=1}^{k} exp(z_{Lj})\right)} - \frac{exp(z_{Lc})}{\left(\sum_{j=1}^{k} exp(z_{Lj})\right)} \frac{exp(z_{Li})}{\left(\sum_{j=1}^{k} exp(z_{Lj})\right)}$$

$$= \mathbb{1}_{c=i} Softmax(z_{Lc}) - Softmax(z_{Lc}) Softmax(z_{Li})$$

$$= Softmax(z_{Lc}) \left(\mathbb{1}_{c=i} - Softmax(z_{Li})\right)$$

Backpropagation: Gradient with respect to Output Neurons



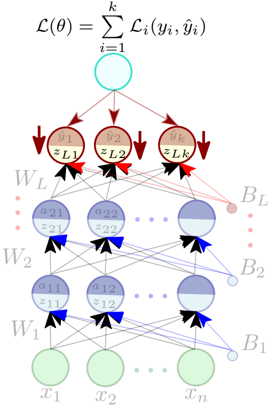
$$\frac{\partial \hat{y}_c}{\partial z_{Li}} = \frac{\partial}{\partial z_{Li}} Softmax(z_{Lc}) = \frac{\partial}{\partial z_{Li}} \left(\frac{exp(z_{Lc})}{\sum_{j=1}^{k} exp(z_{Lj})} \right)$$

$$= \mathbb{1}_{c=i} Softmax(z_{Lc}) - Softmax(z_{Lc}) Softmax(z_{Li})$$

$$= Softmax(z_{Lc}) \left(\mathbb{1}_{c=i} - Softmax(z_{Li}) \right)$$

$$= \hat{y}_c \left(\mathbb{1}_{c=i} - \hat{y}_i \right)$$

Backpropagation: Gradient with respect to Output Neurons



$$\frac{\partial \hat{y}_c}{\partial z_{Li}} = \frac{\partial}{\partial z_{Li}} Softmax(z_{Lc}) = \frac{\partial}{\partial z_{Li}} \left(\frac{exp(z_{Lc})}{\sum_{j=1}^{k} exp(z_{Lj})} \right)$$

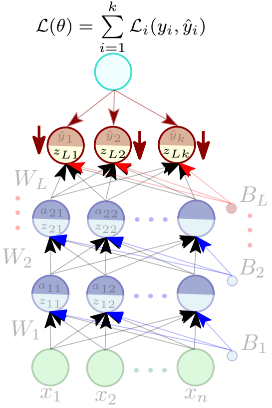
$$= \mathbb{1}_{c=i} Softmax(z_{Lc}) - Softmax(z_{Lc}) Softmax(z_{Li})$$

$$= Softmax(z_{Lc}) \left(\mathbb{1}_{c=i} - Softmax(z_{Li})\right)$$

$$= \hat{y}_c \left(\mathbb{1}_{c=i} - \hat{y}_i\right)$$

$$\frac{\partial \mathcal{L}(\theta)}{\partial z_{Li}} =$$

Backpropagation: Gradient with respect to Output Neurons



$$\frac{\partial \hat{y}_c}{\partial z_{Li}} = \frac{\partial}{\partial z_{Li}} Softmax(z_{Lc}) = \frac{\partial}{\partial z_{Li}} \left(\frac{exp(z_{Lc})}{\sum_{j=1}^{k} exp(z_{Lj})} \right)$$

$$= \mathbb{1}_{c=i} Softmax(z_{Lc}) - Softmax(z_{Lc}) Softmax(z_{Li})$$

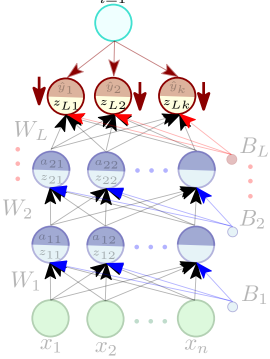
$$= Softmax(z_{Lc}) \left(\mathbb{1}_{c=i} - Softmax(z_{Li})\right)$$

$$= \hat{y}_c \left(\mathbb{1}_{c=i} - \hat{y}_i\right)$$

$$\frac{\partial \mathcal{L}(\theta)}{\partial z_{Li}} = \frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_c} \frac{\partial \hat{y}_c}{\partial z_{Li}} =$$

Backpropagation: Gradient with respect to Output Neurons

$$\mathcal{L}(\theta) = \sum_{i=1}^{k} \mathcal{L}_i(y_i, \hat{y}_i)$$



$$\frac{\partial \hat{y}_c}{\partial z_{Li}} = \frac{\partial}{\partial z_{Li}} Softmax(z_{Lc}) = \frac{\partial}{\partial z_{Li}} \left(\frac{exp(z_{Lc})}{\sum_{j=1}^{k} exp(z_{Lj})} \right)$$

$$= \mathbb{1}_{c=i} Softmax(z_{Lc}) - Softmax(z_{Lc}) Softmax(z_{Li})$$

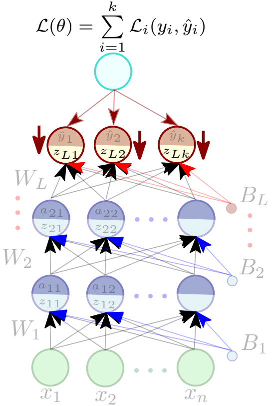
$$= Softmax(z_{Lc}) \left(\mathbb{1}_{c=i} - Softmax(z_{Li}) \right)$$

$$= \hat{y}_c \left(\mathbb{1}_{c=i} - \hat{y}_i \right)$$

$$\frac{\partial \mathcal{L}(\theta)}{\partial z_{Li}} = \frac{\partial \mathcal{L}(\theta)}{\partial \hat{y}_c} \frac{\partial \hat{y}_c}{\partial z_{Li}} = -\frac{1}{\hat{y}_c} \hat{y}_c \left(\mathbb{1}_{c=i} - \hat{y}_i \right)$$

$$\frac{\partial \mathcal{L}(\theta)}{\partial z_{Li}} = -\left(\mathbb{1}_{c=i} - \hat{y}_i \right)$$

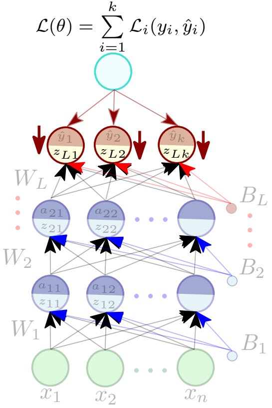
Backpropagation: Gradient with respect to Output Neurons



$$\frac{\partial \mathcal{L}(\theta)}{\partial z_{Li}} = -\left(\mathbb{1}_{c=i} - \hat{y}_i\right)$$

$$\nabla_{z_L} \mathcal{L}(\theta) =$$

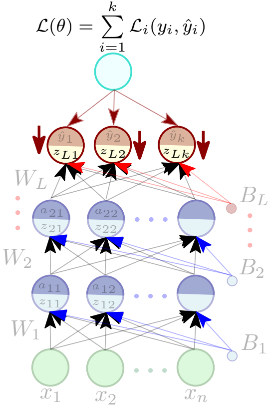
Backpropagation: Gradient with respect to Output Neurons



$$\frac{\partial \mathcal{L}(\theta)}{\partial z_{Li}} = -\left(\mathbb{1}_{c=i} - \hat{y}_i\right)$$

$$\nabla_{z_L} \mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial z_{L1}} \\ \frac{\partial \mathcal{L}(\theta)}{\partial z_{L2}} \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial z_{Lk}} \end{bmatrix} =$$

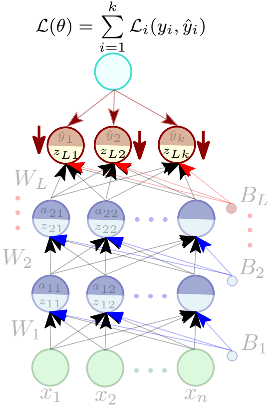
Backpropagation: Gradient with respect to Output Neurons



$$\frac{\partial \mathcal{L}(\theta)}{\partial z_{Li}} = -\left(\mathbb{1}_{c=i} - \hat{y}_i\right)$$

$$\nabla_{z_L}\mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial z_{L1}} \\ \frac{\partial \mathcal{L}(\theta)}{\partial z_{L2}} \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial z_{Lk}} \end{bmatrix} = \begin{bmatrix} -\left(\mathbb{1}_{c=1} - \hat{y}_1\right) \\ -\left(\mathbb{1}_{c=2} - \hat{y}_2\right) \\ \vdots \\ -\left(\mathbb{1}_{c=k} - \hat{y}_k\right) \end{bmatrix} =$$

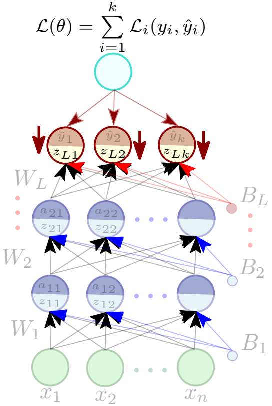
Backpropagation: Gradient with respect to Output Neurons



$$\frac{\partial \mathcal{L}(\theta)}{\partial z_{Li}} = -\left(\mathbb{1}_{c=i} - \hat{y}_i\right)$$

$$\nabla_{z_L}\mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial z_{L1}} \\ \frac{\partial \mathcal{L}(\theta)}{\partial z_{L2}} \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial z_{Lk}} \end{bmatrix} = \begin{bmatrix} -\left(\mathbb{1}_{c=1} - \hat{y}_1\right) \\ -\left(\mathbb{1}_{c=2} - \hat{y}_2\right) \\ \vdots \\ -\left(\mathbb{1}_{c=k} - \hat{y}_k\right) \end{bmatrix} = -(\mathbb{I}(c) - \hat{y})$$

Backpropagation: Gradient with respect to Output Neurons



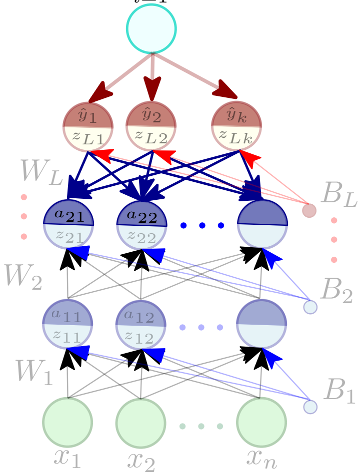
$$\frac{\partial \mathcal{L}(\theta)}{\partial z_{Li}} = -\left(\mathbb{1}_{c=i} - \hat{y}_i\right)$$

$$\nabla_{z_L} \mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial z_{L1}} \\ \frac{\partial \mathcal{L}(\theta)}{\partial z_{L2}} \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial z_{Lk}} \end{bmatrix} = \begin{bmatrix} -\left(\mathbb{1}_{c=1} - \hat{y}_1\right) \\ -\left(\mathbb{1}_{c=2} - \hat{y}_2\right) \\ \vdots \\ -\left(\mathbb{1}_{c=k} - \hat{y}_k\right) \end{bmatrix} = -(\mathbb{I}(c) - \hat{y})$$

$$\nabla_{z_L} \mathcal{L}(\theta) = -\left(\mathbb{I}(c) - \hat{y}\right)$$

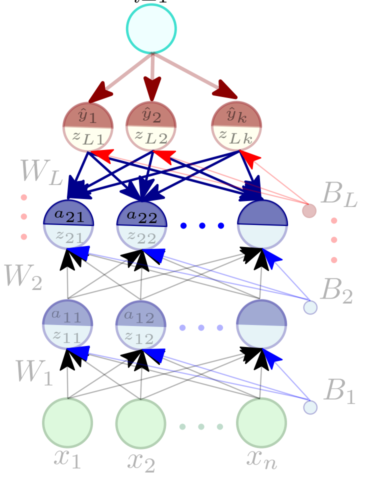
Backpropagation: Gradient with respect to Hidden Neurons

$$\mathcal{L}(\theta) = \sum_{i=1}^{k} \mathcal{L}_i(y_i, \hat{y}_i)$$



Backpropagation: Gradient with respect to Hidden Neurons

$$\mathcal{L}(\theta) = \sum_{i=1}^{k} \mathcal{L}_i(y_i, \hat{y}_i)$$



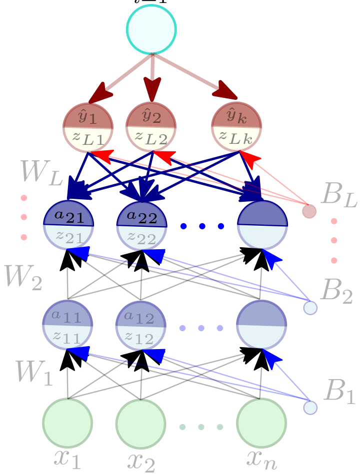
$$\frac{\partial \mathcal{L}(\theta)}{\partial a_{ij}} = \sum_{l=1}^{k} \frac{\partial \mathcal{L}(\theta)}{\partial z_{(i+1)l}} \frac{\partial z_{(i+1)l}}{\partial a_{ij}}$$

$$z_{(i+1)} = W_{(i+1)} a_i + B_{i+1}$$

$$\frac{\partial z_{(i+1)l}}{\partial a_{ij}} = W_{(i+1)lj}$$

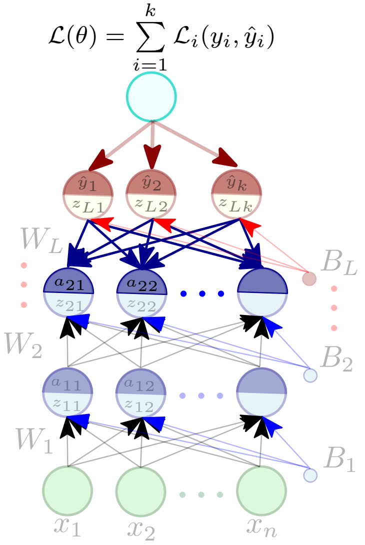
Backpropagation: Gradient with respect to Hidden Neurons

$$\mathcal{L}(\theta) = \sum_{i=1}^{k} \mathcal{L}_i(y_i, \hat{y}_i)$$



$$\frac{\partial z_{(i+1)l}}{\partial a_{ij}} = W_{(i+1)lj}$$

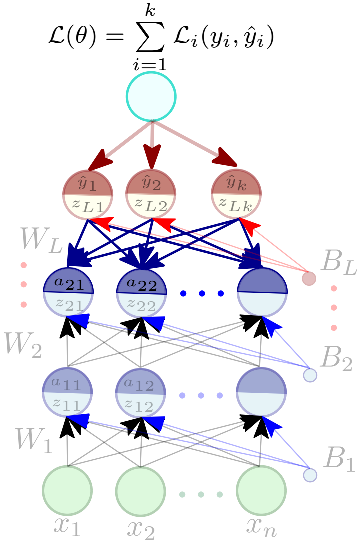
Backpropagation: Gradient with respect to Hidden Neurons



$$\frac{\partial \mathcal{L}(\theta)}{\partial a_{ij}} = \sum_{l=1}^{k} \frac{\partial \mathcal{L}(\theta)}{\partial z_{(i+1)l}} W_{(i+1)lj}$$

$$\nabla_{z_{(i+1)}} \mathcal{L}(\theta) =$$

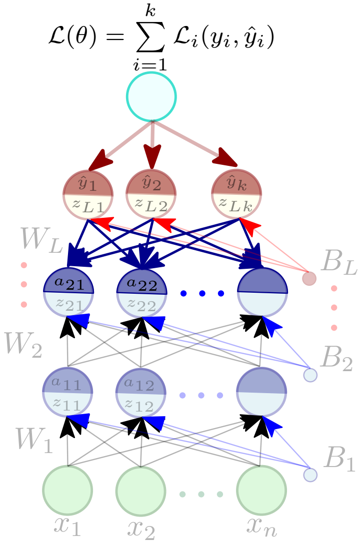
Backpropagation: Gradient with respect to Hidden Neurons



$$\frac{\partial \mathcal{L}(\theta)}{\partial a_{ij}} = \sum_{l=1}^{k} \frac{\partial \mathcal{L}(\theta)}{\partial z_{(i+1)l}} W_{(i+1)lj}$$

$$\nabla_{z_{(i+1)}} \mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial z_{(i+1)1}} \\ \frac{\partial \mathcal{L}(\theta)}{\partial z_{(i+1)2}} \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial z_{(i+1)k}} \end{bmatrix} ; W_{(i+1),j} = \begin{bmatrix} W_{(i+1)1j} \\ W_{(i+1)2j} \\ \vdots \\ W_{(i+1)kj} \end{bmatrix}$$

Backpropagation: Gradient with respect to Hidden Neurons

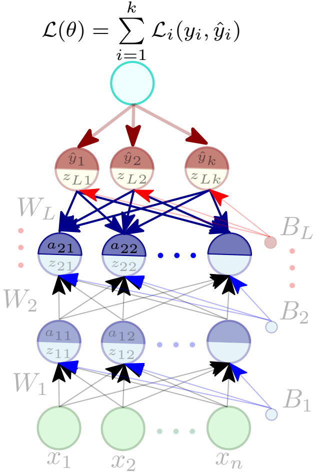


$$\frac{\partial \mathcal{L}(\theta)}{\partial a_{ij}} = \sum_{l=1}^{k} \frac{\partial \mathcal{L}(\theta)}{\partial z_{(i+1)l}} W_{(i+1)lj}$$

$$\nabla_{z_{(i+1)}} \mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial z_{(i+1)1}} \\ \frac{\partial \mathcal{L}(\theta)}{\partial z_{(i+1)2}} \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial z_{(i+1)k}} \end{bmatrix}; W_{(i+1),j} = \begin{bmatrix} W_{(i+1)1j} \\ W_{(i+1)2j} \\ \vdots \\ W_{(i+1)kj} \end{bmatrix}$$

$$(W_{(i+1),j})^T \nabla_{z_{(i+1)}} \mathcal{L}(\theta) = \sum_{l=1}^{k} \frac{\partial \mathcal{L}(\theta)}{\partial z_{(i+1)l}} W_{(i+1)lj}$$

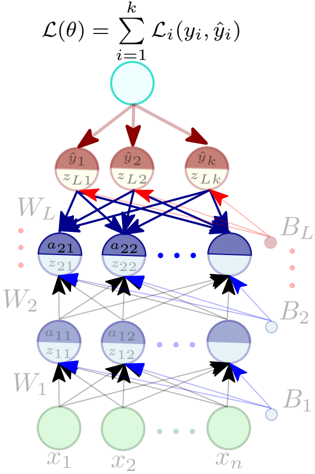
Backpropagation: Gradient with respect to Hidden Neurons



$$\frac{\partial \mathcal{L}(\theta)}{\partial a_{ij}} = (W_{(i+1).,j})^T \nabla_{z_{(i+1)}} \mathcal{L}(\theta)$$

$$\nabla_{a_i} \mathcal{L}(\theta) =$$

Backpropagation: Gradient with respect to Hidden Neurons

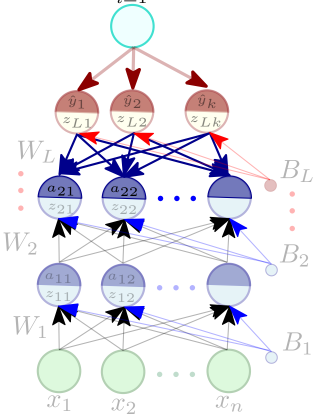


$$\frac{\partial \mathcal{L}(\theta)}{\partial a_{ij}} = (W_{(i+1).j})^T \nabla_{z_{(i+1)}} \mathcal{L}(\theta)$$

$$\nabla_{a_i} \mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial a_{i1}} \\ \frac{\partial \mathcal{L}(\theta)}{\partial a_{i2}} \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial a_{in}} \end{bmatrix} =$$

Backpropagation: Gradient with respect to Hidden Neurons

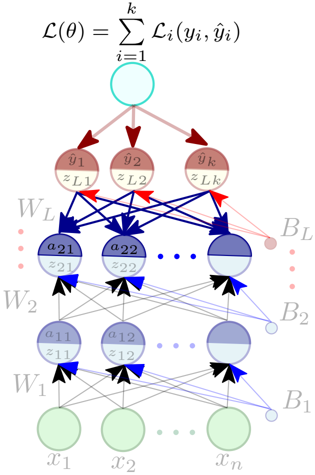
$$\mathcal{L}(\theta) = \sum_{i=1}^{k} \mathcal{L}_i(y_i, \hat{y}_i)$$



$$\frac{\partial \mathcal{L}(\theta)}{\partial a_{ij}} = (W_{(i+1).j})^T \nabla_{z_{(i+1)}} \mathcal{L}(\theta)$$

$$\nabla_{a_i} \mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial a_{i1}} \\ \frac{\partial \mathcal{L}(\theta)}{\partial a_{i2}} \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial a_{in}} \end{bmatrix} = \begin{bmatrix} (W_{(i+1).1})^T \nabla_{z_{(i+1)}} \mathcal{L}(\theta) \\ (W_{(i+1).2})^T \nabla_{z_{(i+1)}} \mathcal{L}(\theta) \\ \vdots \\ (W_{(i+1).n})^T \nabla_{z_{(i+1)}} \mathcal{L}(\theta) \end{bmatrix}$$

Backpropagation: Gradient with respect to Hidden Neurons



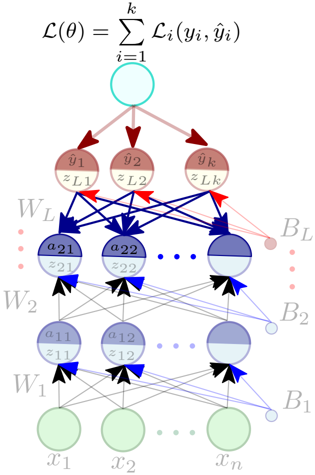
$$\frac{\partial \mathcal{L}(\theta)}{\partial a_{ij}} = (W_{(i+1).j})^T \nabla_{z_{(i+1)}} \mathcal{L}(\theta)$$

$$\nabla_{a_i} \mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial a_{i1}} \\ \frac{\partial \mathcal{L}(\theta)}{\partial a_{i2}} \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial a_{in}} \end{bmatrix} = \begin{bmatrix} (W_{(i+1).1})^T \nabla_{z_{(i+1)}} \mathcal{L}(\theta) \\ (W_{(i+1).2})^T \nabla_{z_{(i+1)}} \mathcal{L}(\theta) \\ \vdots \\ (W_{(i+1).n})^T \nabla_{z_{(i+1)}} \mathcal{L}(\theta) \end{bmatrix}$$

$$\nabla_{a_i} \mathcal{L}(\theta) = (W_{(i+1)})^T \nabla_{z_{(i+1)}} \mathcal{L}(\theta)$$

$$\nabla_{a_i} \mathcal{L}(\theta) = (W_{(i+1)})^T \nabla_{z_{(i+1)}} \mathcal{L}(\theta)$$

Backpropagation: Gradient with respect to Hidden Neurons

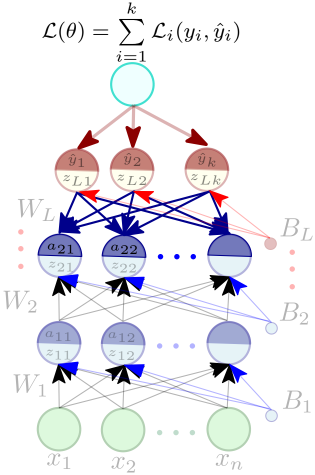


$$\frac{\partial \mathcal{L}(\theta)}{\partial z_{ij}} = \frac{\partial \mathcal{L}(\theta)}{\partial a_{ij}} \frac{\partial a_{ij}}{\partial z_{ij}}$$

$$\frac{\partial a_{ij}}{\partial z_{ij}} = g'(z_{ij})$$

$$\nabla_{z_i} \mathcal{L}(\theta) =$$

Backpropagation: Gradient with respect to Hidden Neurons



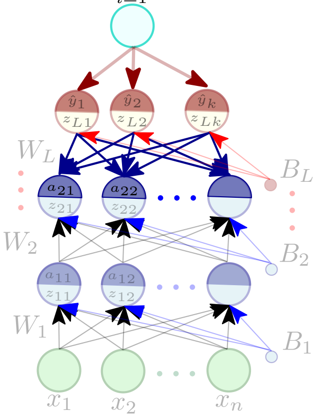
$$\frac{\partial \mathcal{L}(\theta)}{\partial z_{ij}} = \frac{\partial \mathcal{L}(\theta)}{\partial a_{ij}} \frac{\partial a_{ij}}{\partial z_{ij}}$$

$$\frac{\partial a_{ij}}{\partial z_{ij}} = g'(z_{ij})$$

$$\nabla_{z_i} \mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial z_{i1}} \\ \frac{\partial \mathcal{L}(\theta)}{\partial z_{i2}} \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial z_{in}} \end{bmatrix} =$$

Backpropagation: Gradient with respect to Hidden Neurons

$$\mathcal{L}(\theta) = \sum_{i=1}^{k} \mathcal{L}_i(y_i, \hat{y}_i)$$



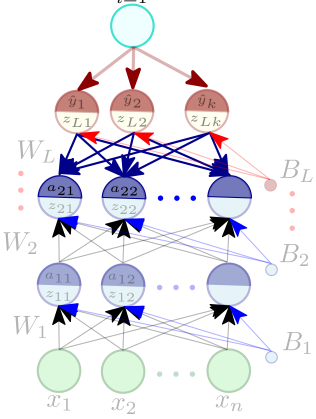
$$\frac{\partial \mathcal{L}(\theta)}{\partial z_{ij}} = \frac{\partial \mathcal{L}(\theta)}{\partial a_{ij}} \frac{\partial a_{ij}}{\partial z_{ij}}$$

$$\frac{\partial a_{ij}}{\partial z_{ij}} = g'(z_{ij})$$

$$\nabla_{z_i} \mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial z_{i1}} \\ \frac{\partial \mathcal{L}(\theta)}{\partial z_{i2}} \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial z_{in}} \end{bmatrix} = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial a_{i1}} g'(z_{i1}) \\ \frac{\partial \mathcal{L}(\theta)}{\partial a_{i2}} g'(z_{i2}) \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial a_{in}} g'(z_{in}) \end{bmatrix}$$

Backpropagation: Gradient with respect to Hidden Neurons

$$\mathcal{L}(\theta) = \sum_{i=1}^{k} \mathcal{L}_i(y_i, \hat{y}_i)$$



$$\frac{\partial \mathcal{L}(\theta)}{\partial z_{ij}} = \frac{\partial \mathcal{L}(\theta)}{\partial a_{ij}} \frac{\partial a_{ij}}{\partial z_{ij}}$$

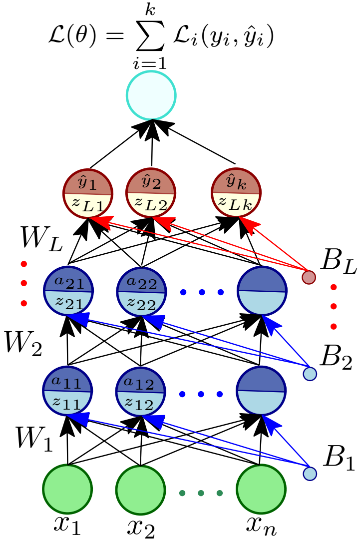
$$\frac{\partial a_{ij}}{\partial z_{ij}} = g'(z_{ij})$$

$$\nabla_{z_i} \mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial z_{i1}} \\ \frac{\partial \mathcal{L}(\theta)}{\partial z_{i2}} \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial z_{in}} \end{bmatrix} = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial a_{i1}} g'(z_{i1}) \\ \frac{\partial \mathcal{L}(\theta)}{\partial a_{i2}} g'(z_{i2}) \\ \vdots \\ \frac{\partial \mathcal{L}(\theta)}{\partial a_{in}} g'(z_{in}) \end{bmatrix}$$

$$\nabla_{z_i} \mathcal{L}(\theta) = \nabla_{a_i} \mathcal{L}(\theta) \odot [g'(z_{i1}) g'(z_{i2}) \dots g'(z_{in})]$$

$$\boxed{\nabla_{z_i} \mathcal{L}(\theta) = \nabla_{a_i} \mathcal{L}(\theta) \odot [g'(z_{i1}) g'(z_{i2}) \dots g'(z_{in})]}$$

Backpropagation: Gradient with respect to Weights



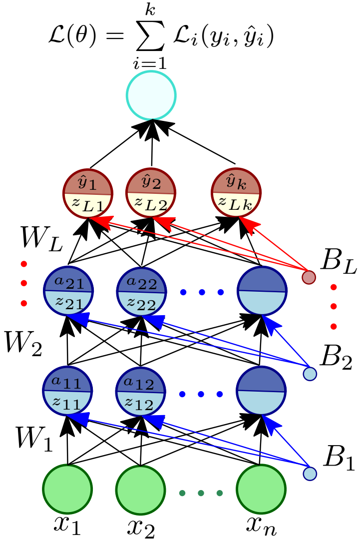
$$\frac{\partial \mathcal{L}(\theta)}{\partial W_{ijl}} = \frac{\partial \mathcal{L}(\theta)}{\partial z_{ij}} \frac{\partial z_{ij}}{\partial W_{ijl}}$$

$$\frac{\partial z_{ij}}{\partial W_{ijl}} = a_{(i-1)l}$$

$$\frac{\partial \mathcal{L}(\theta)}{\partial W_{ijl}} = \frac{\partial \mathcal{L}(\theta)}{\partial z_{ij}} a_{(i-1)l}$$

$$\nabla_{W_i} \mathcal{L}(\theta) =$$

Backpropagation: Gradient with respect to Weights



$$\frac{\partial \mathcal{L}(\theta)}{\partial W_{ijl}} = \frac{\partial \mathcal{L}(\theta)}{\partial z_{ij}} \frac{\partial z_{ij}}{\partial W_{ijl}}$$

$$\frac{\partial z_{ij}}{\partial W_{ijl}} = a_{(i-1)l}$$

$$\frac{\partial \mathcal{L}(\theta)}{\partial W_{ijl}} = \frac{\partial \mathcal{L}(\theta)}{\partial z_{ij}} a_{(i-1)l}$$

$$\nabla_{W_i} \mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial \mathcal{L}(\theta)}{\partial W_{i11}} & \frac{\partial \mathcal{L}(\theta)}{\partial W_{i12}} & \cdots & \frac{\partial \mathcal{L}(\theta)}{\partial W_{i1n}} \\ \frac{\partial \mathcal{L}(\theta)}{\partial W_{i21}} & \frac{\partial \mathcal{L}(\theta)}{\partial W_{i22}} & \cdots & \frac{\partial \mathcal{L}(\theta)}{\partial W_{i2n}} \\ \vdots & & & \\ \frac{\partial \mathcal{L}(\theta)}{\partial W_{in1}} & \frac{\partial \mathcal{L}(\theta)}{\partial W_{in2}} & \cdots & \frac{\partial \mathcal{L}(\theta)}{\partial W_{inn}} \end{bmatrix}$$

Backpropagation: Gradient with respect to Weights

$$\nabla_{W_i}\mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial\mathcal{L}(\theta)}{\partial W_{i11}} & \frac{\partial\mathcal{L}(\theta)}{\partial W_{i12}} & \cdots & \frac{\partial\mathcal{L}(\theta)}{\partial W_{i1n}} \\ \frac{\partial\mathcal{L}(\theta)}{\partial W_{i21}} & \frac{\partial\mathcal{L}(\theta)}{\partial W_{i22}} & \cdots & \frac{\partial\mathcal{L}(\theta)}{\partial W_{i2n}} \\ \vdots \\ \frac{\partial\mathcal{L}(\theta)}{\partial W_{in1}} & \frac{\partial\mathcal{L}(\theta)}{\partial W_{in2}} & \cdots & \frac{\partial\mathcal{L}(\theta)}{\partial W_{inn}} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{\partial\mathcal{L}(\theta)}{\partial z_{i1}} a_{(i-1)1} & \frac{\partial\mathcal{L}(\theta)}{\partial z_{i1}} a_{(i-1)2} & \cdots & \frac{\partial\mathcal{L}(\theta)}{\partial z_{i1}} a_{(i-1)n} \\ \frac{\partial\mathcal{L}(\theta)}{\partial z_{i2}} a_{(i-1)1} & \frac{\partial\mathcal{L}(\theta)}{\partial a_{i2}} a_{(i-1)2} & \cdots & \frac{\partial\mathcal{L}(\theta)}{\partial z_{i2}} a_{(i-1)n} \\ \vdots \\ \frac{\partial\mathcal{L}(\theta)}{\partial z_{in}} a_{(i-1)1} & \frac{\partial\mathcal{L}(\theta)}{\partial z_{in}} a_{(i-1)2} & \cdots & \frac{\partial\mathcal{L}(\theta)}{\partial z_{in}} a_{(i-1)n} \end{bmatrix}$$

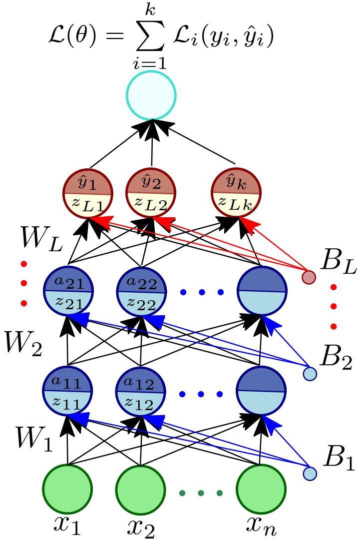
Backpropagation: Gradient with respect to Weights

$$\nabla_{W_i}\mathcal{L}(\theta) = \begin{bmatrix} \frac{\partial\mathcal{L}(\theta)}{\partial z_{i1}} a_{(i-1)1} & \frac{\partial\mathcal{L}(\theta)}{\partial z_{i1}} a_{(i-1)2} & \cdots & \frac{\partial\mathcal{L}(\theta)}{\partial z_{i1}} a_{(i-1)n} \\ \frac{\partial\mathcal{L}(\theta)}{\partial z_{i2}} a_{(i-1)1} & \frac{\partial\mathcal{L}(\theta)}{\partial a_{i2}} a_{(i-1)2} & \cdots & \frac{\partial\mathcal{L}(\theta)}{\partial z_{i2}} a_{(i-1)n} \\ \vdots & & & \\ \frac{\partial\mathcal{L}(\theta)}{\partial z_{in}} a_{(i-1)1} & \frac{\partial\mathcal{L}(\theta)}{\partial z_{in}} a_{(i-1)2} & \cdots & \frac{\partial\mathcal{L}(\theta)}{\partial z_{in}} a_{(i-1)n} \end{bmatrix}$$

$$= \nabla_{z_i}\mathcal{L}(\theta)(a_{(i-1)})^T$$

$$\nabla_{W_i}\mathcal{L}(\theta) = \nabla_{z_i}\mathcal{L}(\theta)(a_{(i-1)})^T$$

Backpropagation: Gradient with respect to Biases



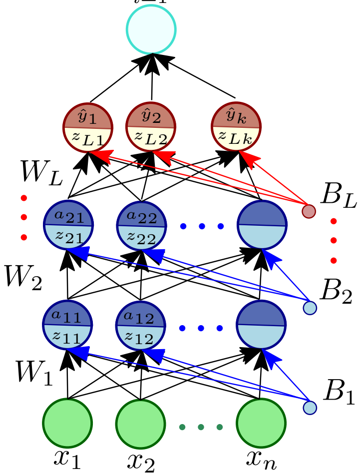
$$\frac{\partial \mathcal{L}(\theta)}{\partial B_{ij}} = \frac{\partial \mathcal{L}(\theta)}{\partial z_{ij}} \frac{\partial z_{ij}}{\partial B_{ij}}$$

$$\frac{\partial z_{ij}}{\partial B_{ij}} = 1$$

$$\frac{\partial \mathcal{L}(\theta)}{\partial B_{ij}} = \frac{\partial \mathcal{L}(\theta)}{\partial z_{ij}}$$

Backpropagation: Gradient with respect to Biases

$$\mathcal{L}(\theta) = \sum_{i=1}^{k} \mathcal{L}_i(y_i, \hat{y}_i)$$



$$\frac{\partial \mathcal{L}(\theta)}{\partial B_{ij}} = \frac{\partial \mathcal{L}(\theta)}{\partial z_{ij}} \frac{\partial z_{ij}}{\partial B_{ij}}$$

$$\frac{\partial z_{ij}}{\partial B_{ij}} = 1$$

$$\frac{\partial \mathcal{L}(\theta)}{\partial B_{ij}} = \frac{\partial \mathcal{L}(\theta)}{\partial z_{ij}}$$

$$\nabla_{B_i} \mathcal{L}(\theta) = \nabla_{z_i} \mathcal{L}(\theta)$$

Algorithm: Feedforward Network with Backpropagation

Algorithm 1 Pseudocode for Feedforward Network with Backpropagation

$t \leftarrow 0$ {Iteration count}

$\theta_0 = (W_1, W_2, \ldots, W_L, B_1, B_2, \ldots, B_L)$; {Initialize learning parameters}

repeat

    $M \leftarrow$ ForwardPropagation(θ_t); {M is the model $(z_i, a_i, \hat{y})$}

    $\Delta_\theta^t \leftarrow$ Backpropagation (M);

    $\theta_{t+1} \leftarrow \theta_t - \eta \Delta_\theta^t$;

    $t+ = 1$;

until Converge

Algorithm: Feedforward Network with Backpropagation

Algorithm 2 Pseudocode for ForwardPropagation

Input: θ_t

Output: $M = (z_i, a_i, \hat{y})$

for $i = 1$ *to* $(L-1)$ **do**

  $z_i = W_i a_{i-1}$[1] $+ B_i$

  $a_i = g(z_i)$ {g: Activation function on i^{th}-layer}

end for

$z_L = W_L a_{L-1} + B_L$

$\hat{y} = O(z_L)$

[1] $a_0 = (x_1 x_2 \dots x_n)$

Algorithm: Feedforward Network with Backpropagation

Algorithm 3 Pseudocode for BackPropagation

Input: $M = (z_i, a_i, \hat{y})$
Output: Δ_θ^t

Compute $\mathcal{L}(\theta)$
$\nabla_{z_L}\mathcal{L}(\theta) = -(\mathbb{I}(c) - \hat{y})$
for $i = L$ *to* 1 **do**
    $\nabla_{W_i}\mathcal{L}(\theta) = \nabla_{z_i}\mathcal{L}(\theta)(a_{(i-1)})^T$
    $\nabla_{B_i}\mathcal{L}(\theta) = \nabla_{z_i}\mathcal{L}(\theta)$
    $\nabla_{a_{i-1}}\mathcal{L}(\theta) = (W_i)^T \nabla_{z_i}\mathcal{L}(\theta)$
    $\nabla_{z_{i-1}}\mathcal{L}(\theta) = \nabla_{a_{i-1}}\mathcal{L}(\theta) \odot [g'(z_{(i-1)1})g'(z_{(i-1)2}) \ldots g'(z_{(i-1)n})]$
end for

Thank You!