Deep Learning

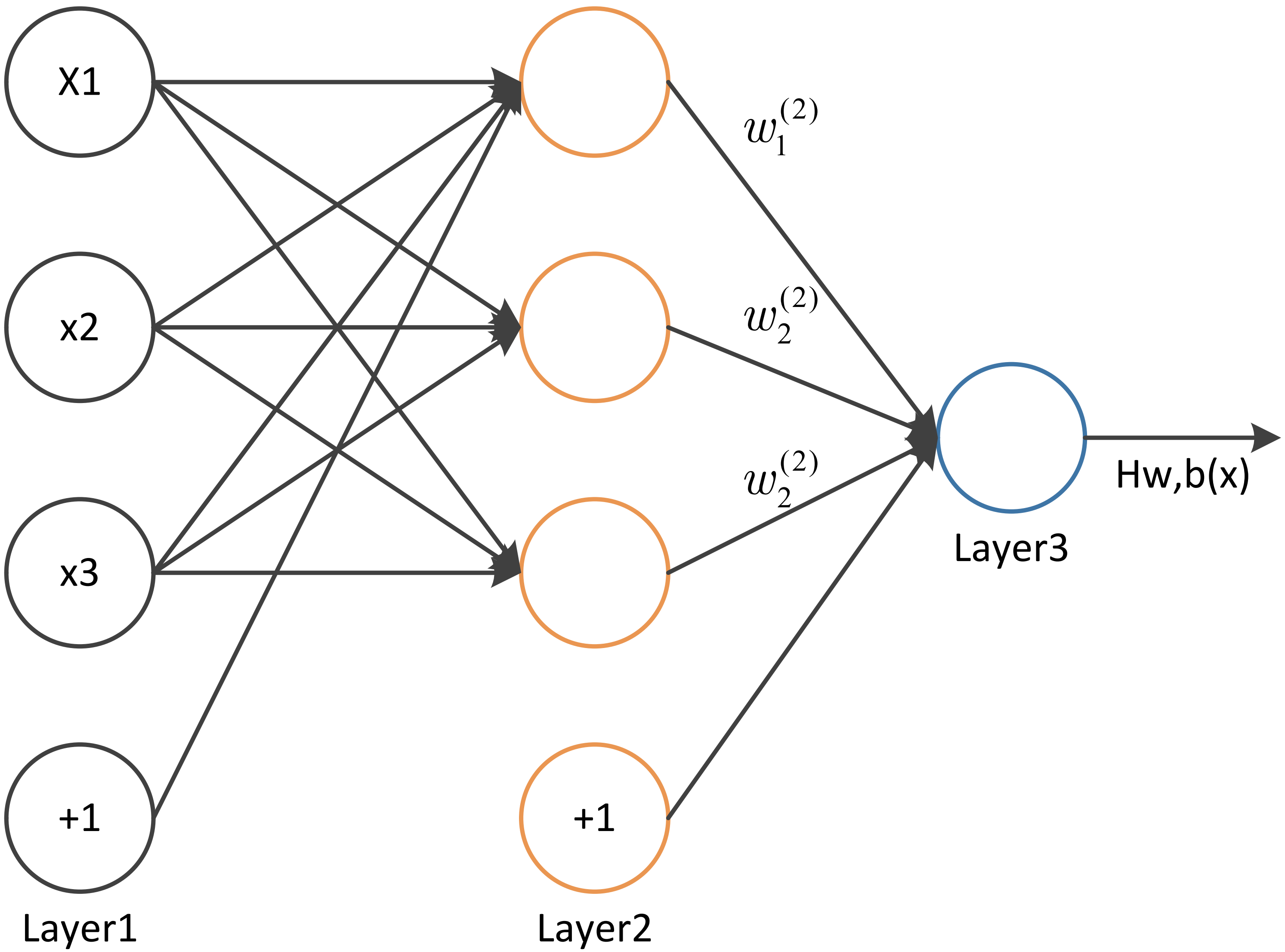
2017年9月13日

陈加忠-华中科技大学计算机科学与技术学院

jzchen@hust.edu.cn

QQ: 645044204

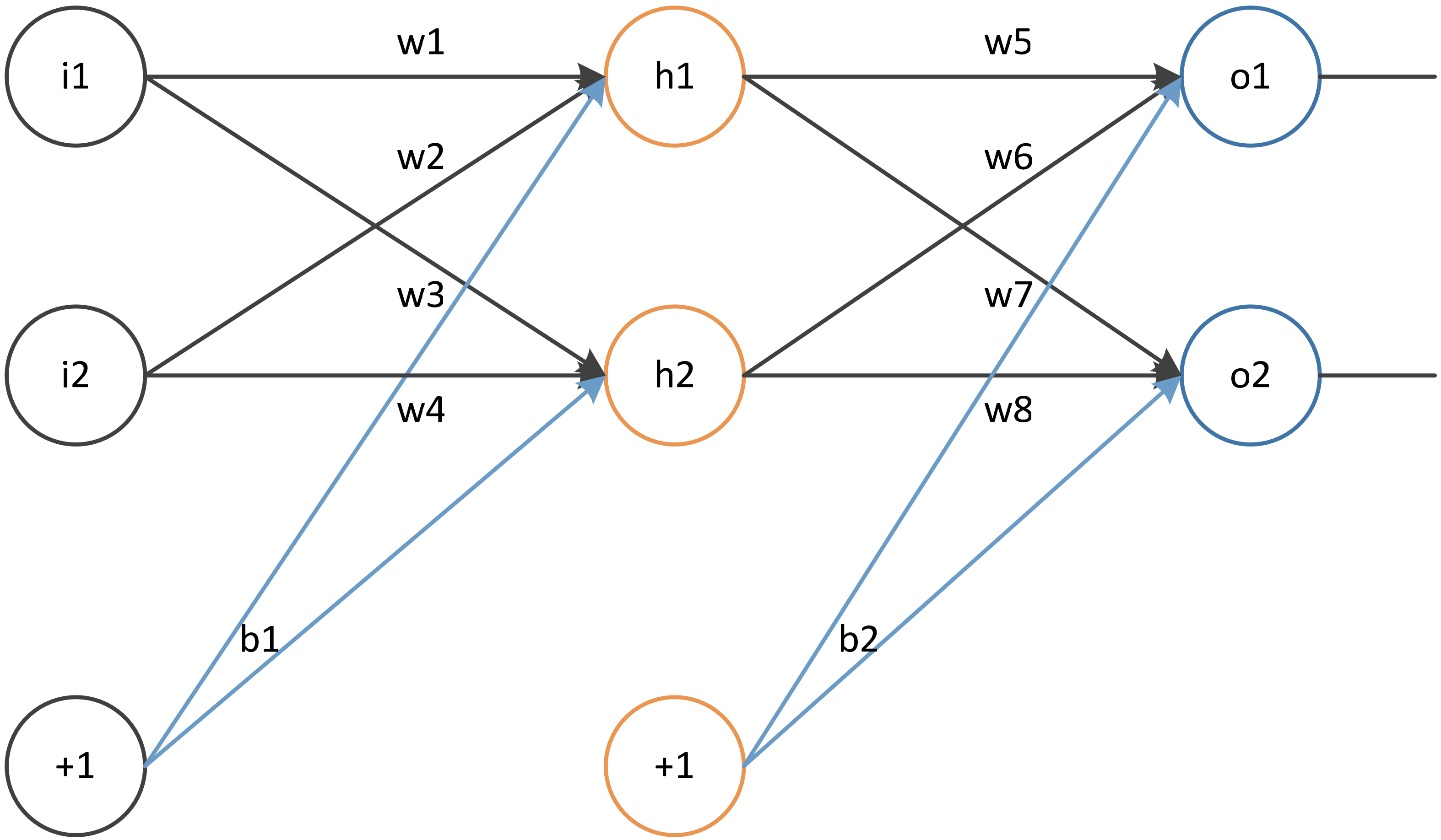
Radical Basis Function 径向基函数



典型的三层神经网络

- 如果希望输出和原始输入一样, 那么就是最常见的自编码模型（Auto-Encoder）.
- 为什么要输入输出都一样呢? 图像识别, 文本分类,……
- 如果输出和原始输入不一样, 那么就是很常见的人工神经网络了, 相当于让原始数据通过一个映射来得到我们想要的输出数据.

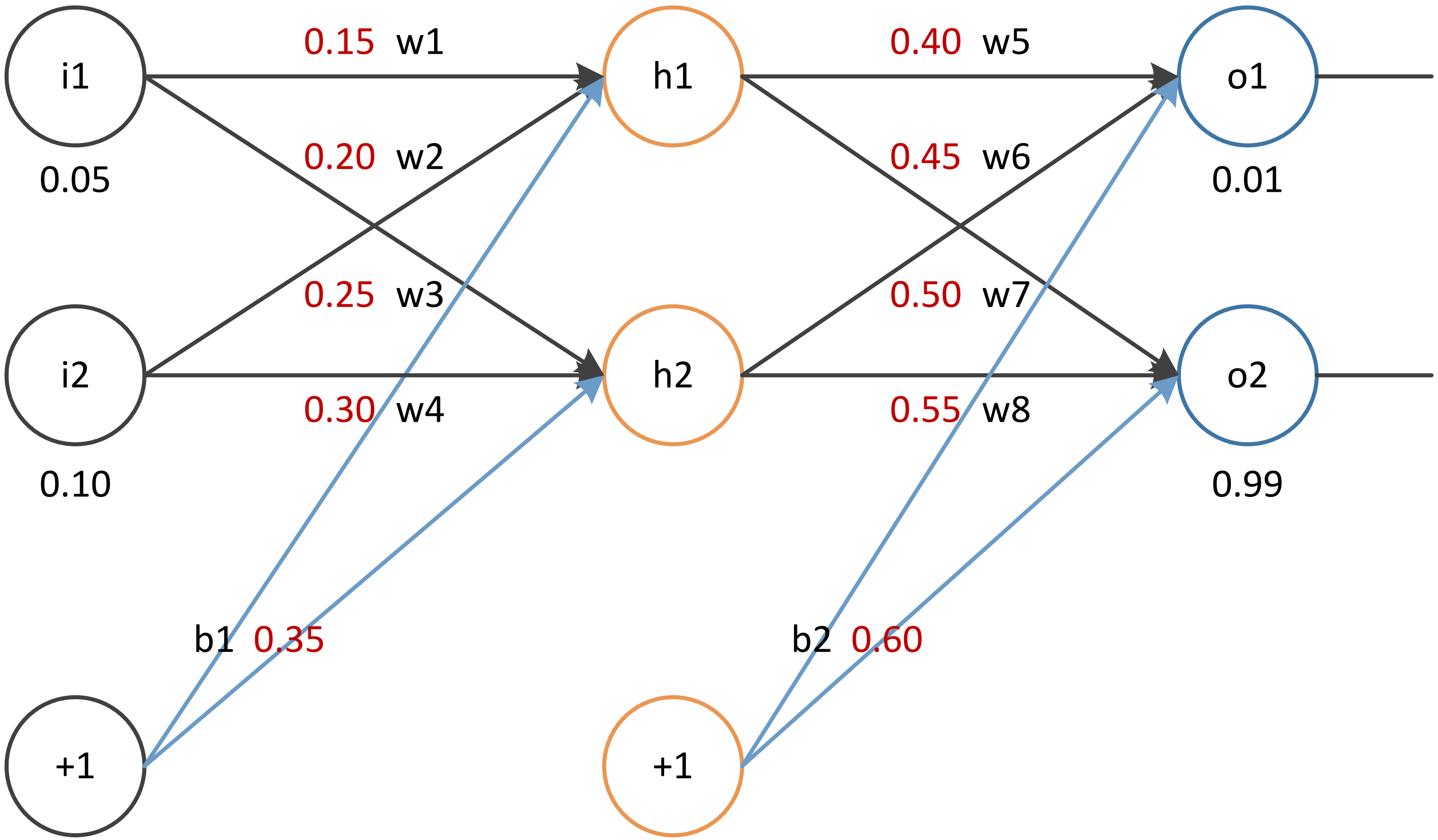
Inference



网络举例

- 第一层是输入层, 包含两个神经元i1, i2和截距项b1
- 第二层是隐含层, 包含两个神经元h1, h2和截距项b2
- 第三层是输出o1, o2
- 每条线上标的w_i是层与层之间连接的权重
- 激活函数默认为sigmoid函数

Inference



i1
0.05
i2
0.10
+1
h1
h2
+1
o1
0.01
o2
0.99
0.15 w1
0.20 w2
0.25 w3
0.30 w4
0.40 w5
0.45 w6
0.50 w7
0.55 w8
b1 0.35
b2 0.60

参数

- 其中，输入数据 i1=0.05, i2=0.10
- 输出数据 o1=0.01, o2=0.99
- 初始权重 w1=0.15, w2=0.20, w3=0.25, w4=0.30
- w5=0.40, w6=0.45, w7=0.50, w8=0.55
- 目标: 给出输入数据i1, i2(0.05和0.10), 使输出尽可能与原始输出o1,o2(0.01和0.99)接近

前向传播(Inference)

- 计算神经元h1的输入加权和

$$net_{h1} = w_1 \times i_1 + w_2 \times i_2 + b_1 \times 1$$

$$net_{h1} = 0.15 \times 0.05 + 0.2 \times 0.1 + 0.35 \times 1 = 0.3775$$

前向传播(Inference)

- 神经元h1的输出o1(假设激活函数为sigmoid函数)

$$out_{h1} = \frac{1}{1+e^{-net_{h1}}} = \frac{1}{1+e^{-0.3775}} = 0.593269992$$

$$out_{h2} = 0.596884378$$

前向传播(Inference)

- 计算输出层神经元o1和o2的值

$$net_{o1} = w_5 \times out_{h1} + w_6 \times out_{h2} + b_2 \times 1$$

$$net_{o1} = 0.4 \times 0.593269992 + 0.45 \times 0.596884378$$
$$+0.6 \times 1 = 1.0105905967$$

$$out_{o1} = \frac{1}{1+e^{-net_{o1}}} = \frac{1}{1+e^{-1.105905967}} = 0.75136507(0.01)$$

$$out_{o2} = 0.772928465(0.99)$$

反向传播(Learning)

- 计算总误差 (Square error, Cross entropy)

$$E_{total} = \sum \frac{1}{2}(target - output)^2$$

$$E_{01} = \frac{1}{2}(target_{o1} - out_{o1})^2 = \frac{1}{2}(0.01 - 0.7513657)^2$$

$$= 0.274811083$$

$$E_{o2} = 0.023560026$$

$$E_{total} = E_{o1} + E_{o2} = 0.274811083 + 0.023560026$$

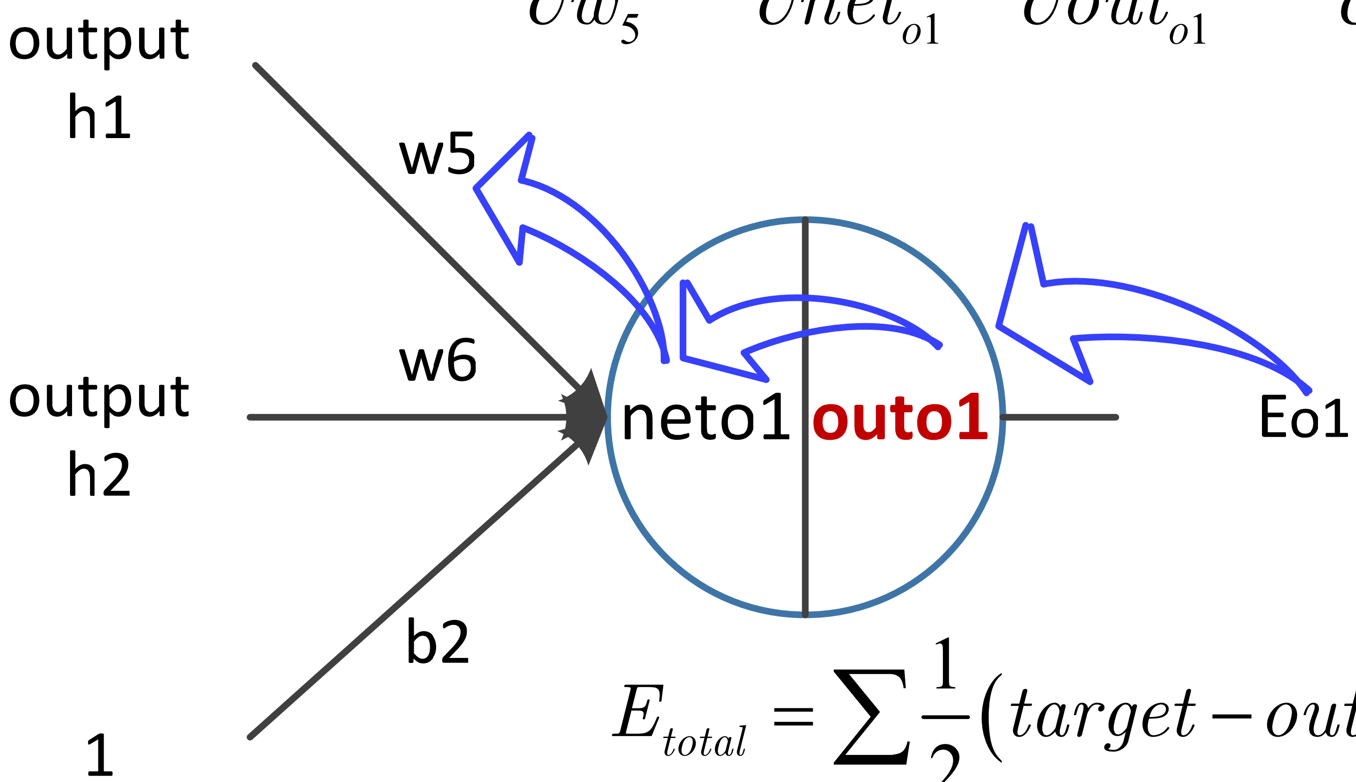
隐层到输出层的权值更新

- 以w_5为例, 求偏导(Chain rule)

$$\frac{\partial E_{total}}{\partial w_5} = \frac{\partial E_{total}}{\partial out_{o1}} \times \frac{\partial out_{o1}}{\partial net_{o1}} \times \frac{\partial net_{o1}}{\partial w_5}$$

Learning

$$\frac{\partial net_{o1}}{\partial w_5} \times \frac{\partial out_{o1}}{\partial net_{o1}} \times \frac{\partial E_{total}}{\partial out_{o1}} = \frac{\partial E_{total}}{\partial w_5}$$



$$E_{total} = \sum \frac{1}{2}(target - output)^2$$

$$E_{total} = E_{o1} + E_{o2}$$

Learning计算偏导

$$E_{total} = \frac{1}{2}\left(target_{o1} - out_{o1}\right)^2 + \frac{1}{2}\left(target_{o2} - out_{o2}\right)^2$$

$$\frac{\partial E_{total}}{\partial out_{o1}} = \boxed{\frac{\partial E_{o1}}{\partial out_{o1}}} = -\left(target_{o1} - out_{o1}\right) = -\left(0.01 - 0.75136507\right)$$

$$= 0.74136507$$

Learning计算偏导

$$out_{o1} = \frac{1}{1+e^{-net_{o1}}}$$

$$\boxed{\frac{\partial out_{o1}}{\partial net_{o1}}} = out_{o1}\left(1 - out_{o1}\right)$$

$$= 0.75136507\left(1 - 0.75136507\right) = 0.186815602$$

Learning计算偏导

$$net_{o1} = w_5 \times out_{h1} + w_6 \times out_{h2} + b_2 \times 1$$

$$\frac{\partial net_{o1}}{\partial w_5} = out_{h1} = 0.593269992$$

Learning计算偏导

$$\frac{\partial E_{total}}{\partial w_5}=\frac{\partial E_{total}}{\partial out_{o1}}\times\frac{\partial out_{o1}}{\partial net_{o1}}\times\frac{\partial net_{o1}}{\partial w_5}$$

$$\frac{\partial E_{total}}{\partial w_5}=0.74136507\times 0.186815602\times 0.593269992$$

$$=0.082167041$$

权值更新 Gradient descent

- 学习率

$$w_5^+ = w_5 - \eta \times \frac{\partial E_{total}}{\partial w_5} = 0.4 - 0.5 \times 0.082167041$$

$$= 0.35891648$$

$$w_6^+ = 0.408666186$$

$$w_7^+ = 0.511501270$$

$$w_8^+ = 0.561370121$$

为迭代做准备

$$\delta_{o1}^{2} = \frac{\partial E_{total}}{\partial net_{o1}} = \frac{\partial E_{total}}{\partial out_{o1}} \times \frac{\partial out_{o1}}{\partial net_{o1}}$$

$$\delta_{o1}^{2} = -\left(target_{o1} - out_{o1}\right) \times out_{o1} \times \left(1 - out_{o1}\right)$$

$$\frac{\partial E_{total}}{\partial w_5} = \delta_{o1}^{2} \times \frac{\partial net_{o1}}{\partial w_5} = \delta_{o1}^{2} \times out_{h1}$$

隐层到隐层的权值更新

- 在计算总误差对w5的偏导时，是从out(o1)→net(o1)→w_5
- 但在隐含层之间的权值更新时，是out(h1)→net(h1)→w_1
- 而out(h1)会接受E(o1)和E(o2)两个地方传来的误差
- 所以这两个误差都要卷入计算

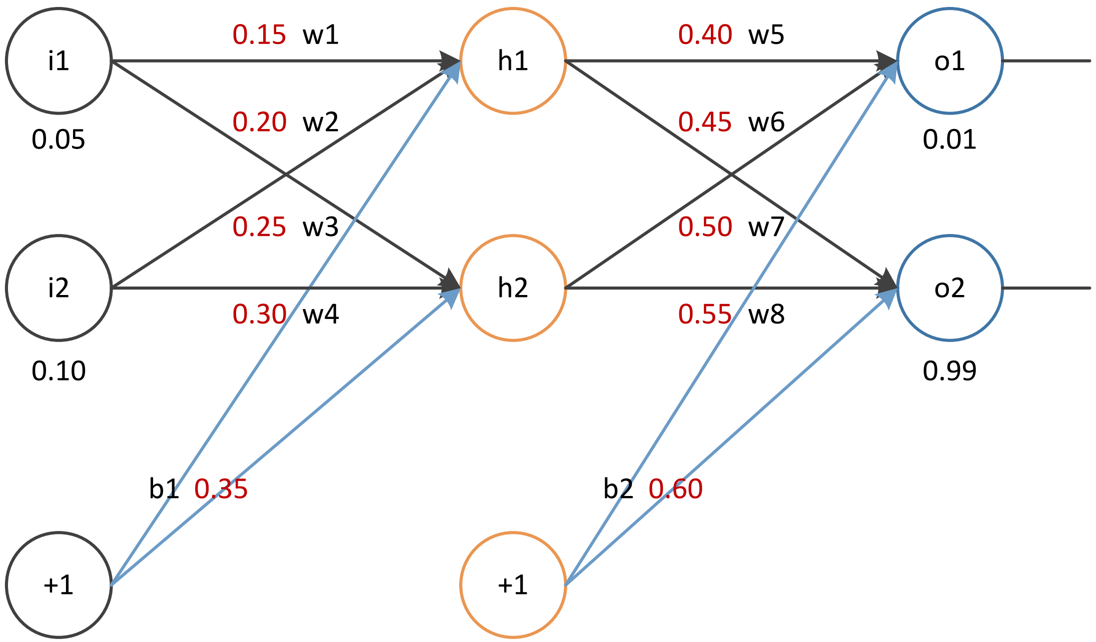
Learning

$$\frac{\partial E_{total}}{\partial w_1} = \frac{\partial E_{total}}{\partial out_{h1}} \times \frac{\partial out_{h1}}{\partial net_{h1}} \times \frac{\partial net_{h1}}{\partial w_1}$$

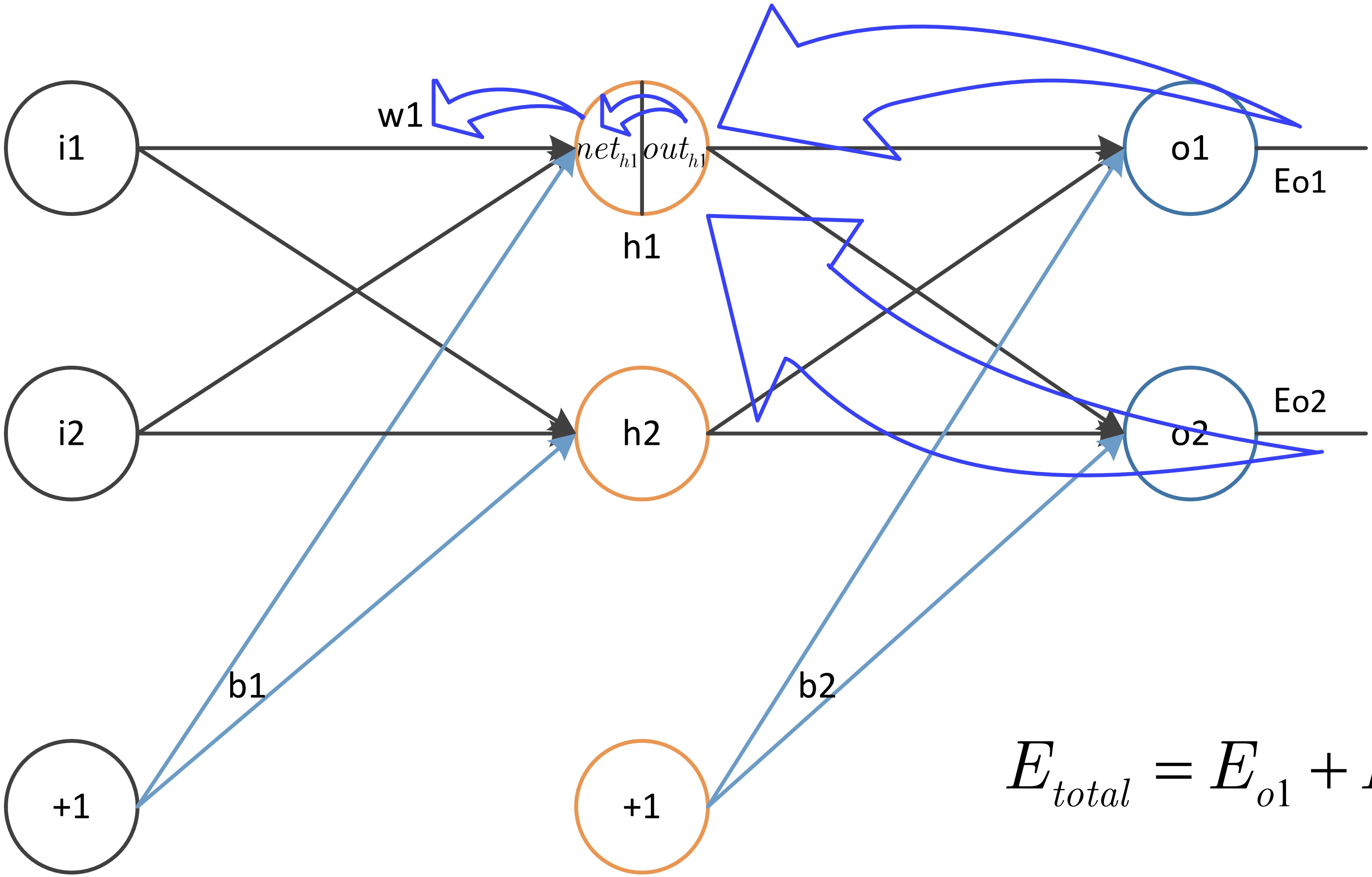
$$\frac{\partial E_{total}}{\partial w_2} = \frac{\partial E_{total}}{\partial out_{h1}} \times \frac{\partial out_{h1}}{\partial net_{h1}} \times \frac{\partial net_{h1}}{\partial w_2}$$

$$\frac{\partial E_{total}}{\partial w_3} = \frac{\partial E_{total}}{\partial out_{h2}} \times \frac{\partial out_{h2}}{\partial net_{h2}} \times \frac{\partial net_{h2}}{\partial w_3}$$

$$\frac{\partial E_{total}}{\partial w_4} = \frac{\partial E_{total}}{\partial out_{h2}} \times \frac{\partial out_{h2}}{\partial net_{h2}} \times \frac{\partial net_{h2}}{\partial w_4}$$



Learning



$$E_{total} = E_{o1} + E_{o2}$$

Learning

$$\frac{\partial E_{total}}{\partial out_{h1}} = \frac{\partial E_{o1}}{\partial out_{h1}} + \frac{\partial E_{o2}}{\partial out_{h1}}$$

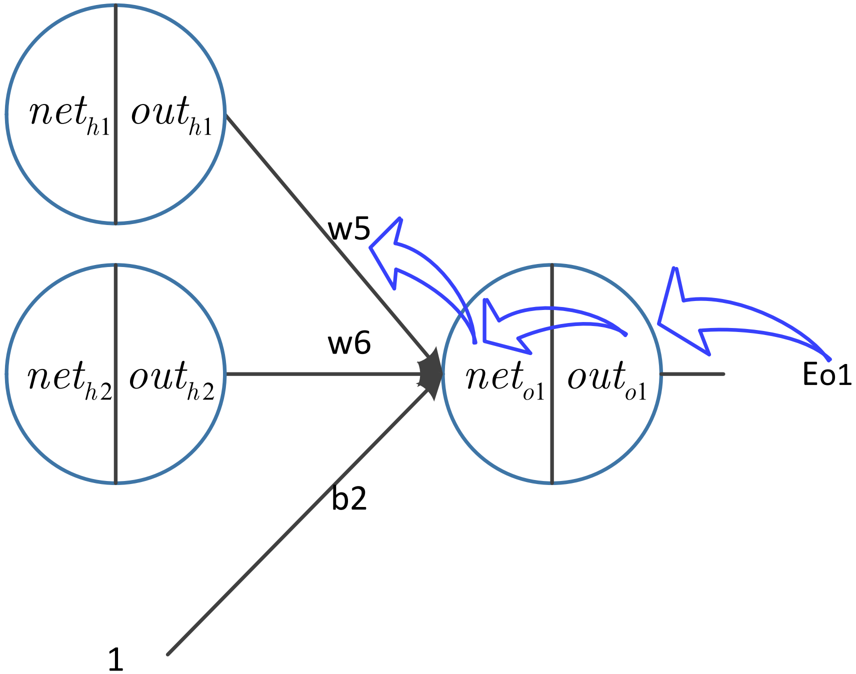
$$\frac{\partial E_{o1}}{\partial out_{h1}} = \boxed{\frac{\partial E_{o1}}{\partial net_{o1}}} \times \frac{\partial net_{o1}}{\partial out_{h1}}$$

$$\frac{\partial E_{o1}}{\partial net_{o1}} = \underline{\frac{\partial E_{o1}}{\partial out_{o1}}} \times \underline{\frac{\partial out_{o1}}{\partial net_{o1}}}$$

$$= 0.74136507 \times 0.186815602 = 0.138498562$$

Learning

$$net_{o1} = w_5 \times out_{h1} + w_6 \times out_{h2} + b_2 \times 1$$



$$\frac{\partial net_{o1}}{\partial out_{h1}} = w_5 = 0.4$$

Learning

$$\frac{\partial E_{o1}}{\partial out_{h1}} = \frac{\partial E_{o1}}{\partial net_{o1}} \times \frac{\partial net_{o1}}{\partial out_{h1}}$$

$$= 0.138498562 \times 0.4 = 0.055399425$$

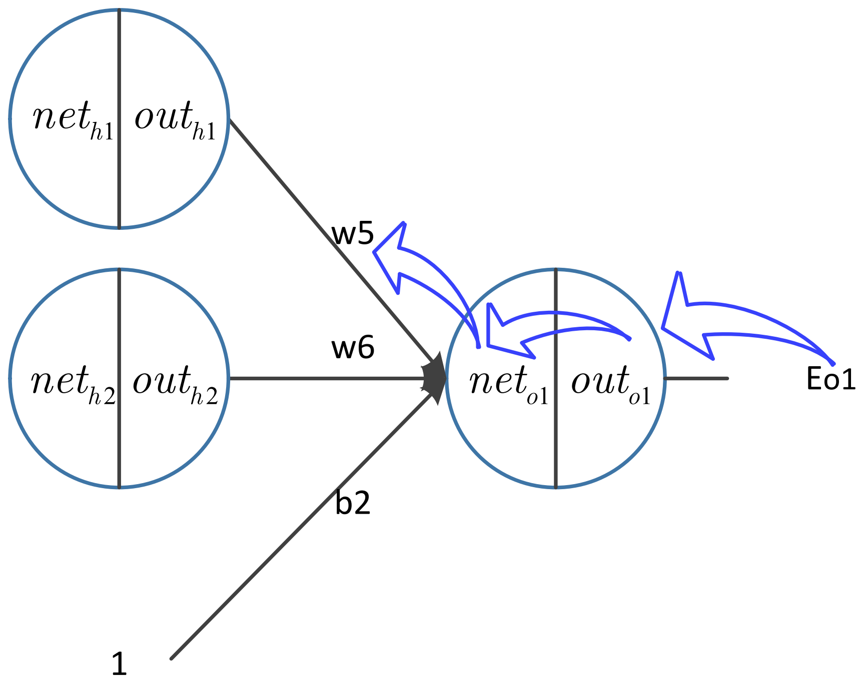
$$\frac{\partial E_{o2}}{\partial out_{h1}} = -0.019049119$$

$$\boxed{\frac{\partial E_{total}}{\partial out_{h1}}} = \frac{\partial E_{o1}}{\partial out_{h1}} + \frac{\partial E_{o2}}{\partial out_{h1}}$$

$$= 0.055399425 - 0.019049119 = 0.036350306$$

Learning

$$out_{h1} = \frac{1}{1+e^{-net_{h1}}}$$



$$\boxed{\frac{\partial out_{h1}}{\partial net_{h1}}} = out_{h1}\left(1-out_{h1}\right)$$

$$= 0.59326999\left(1-0.59326999\right) = 0.241300709$$

Learning

$$net_{h1} = w_1 \times i_1 + w_2 \times i_2 + b_1 \times 1$$

$$\frac{\partial net_{h1}}{\partial w_1} = i_1 = 0.05$$

$$\frac{\partial E_{total}}{\partial w_1} = \frac{\partial E_{total}}{\partial out_{h1}} \times \frac{\partial out_{h1}}{\partial net_{h1}} \times \frac{\partial net_{h1}}{\partial w_1}$$

$$= 0.036350306 \times 0.241300709 \times 0.05 = 0.00438568$$

Updating with learning rate

$$w_1^+ = w_1 - \eta \times \frac{\partial E_{total}}{\partial w_1} = 0.15 - 0.5 \times 0.000438568$$

$$= 0.149780716$$

$$w_2^+ = 0.19956143$$

$$w_3^+ = 0.24975114$$

$$w_4^+ = 0.29950299$$

E_{02}展开

$$\frac{\partial E_{o2}}{\partial out_{h1}} = \frac{\partial E_{o2}}{\partial net_{o2}} \times \frac{\partial net_{o2}}{\partial out_{h1}}$$

$$\frac{\partial E_{total}}{\partial out_{o2}} = \frac{\partial E_{o2}}{\partial out_{o2}} = -\left(target_{o2} - out_{o2}\right) = -\left(0.99 - 0.772928465\right)$$

$$= -0.21707154$$

$$\frac{\partial out_{o2}}{\partial net_{o2}} = out_{o2}\left(1 - out_{o2}\right)$$

$$= 0.772928465\left(1 - 0.772928465\right) = 0.17551006$$

$$\frac{\partial E_{o2}}{\partial net_{o2}} = \frac{\partial E_{o2}}{\partial out_{o2}} \times \frac{\partial out_{o2}}{\partial net_{o2}}$$

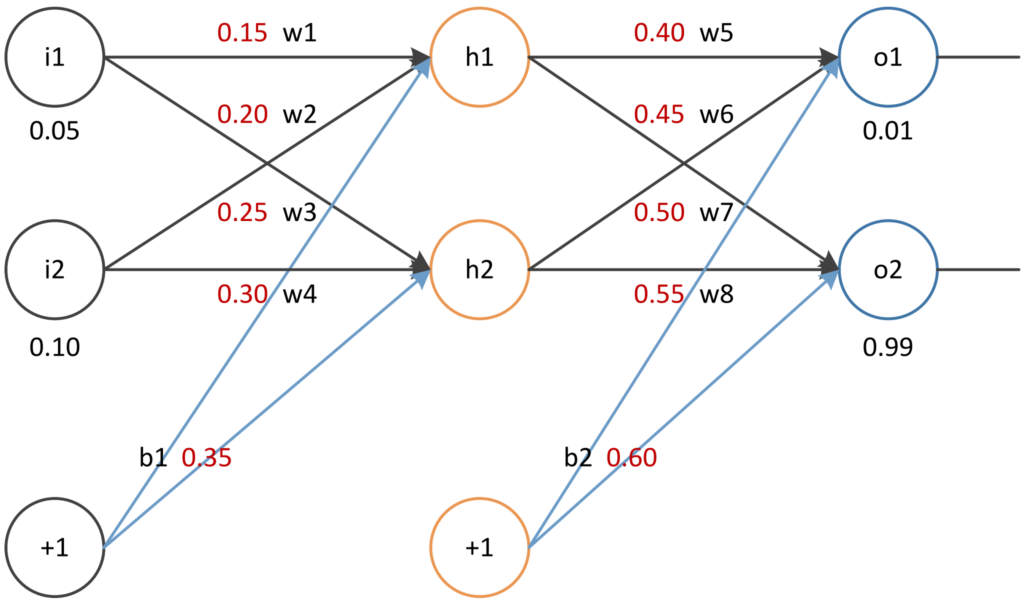
$$= -0.21707154 \times 0.17551006 = -0.03809824$$

E_{02}展开

$$net_{o2} = w_7 \times out_{h1} + w_8 \times out_{h2} + b_2 \times 1$$

$$\frac{\partial net_{o2}}{\partial out_{h1}} = w_7 = 0.5$$

$$\frac{\partial E_{o2}}{\partial out_{h1}} = \frac{\partial E_{o2}}{\partial net_{o2}} \times \frac{\partial net_{o2}}{\partial out_{h1}} = -0.03809824 \times 0.5 = 0.019049119$$



为迭代做准备

$$\frac{\partial E_{total}}{\partial out_{h1}}=\frac{\partial E_{o1}}{\partial out_{h1}}+\frac{\partial E_{o2}}{\partial out_{h1}} \qquad \frac{\partial E_{o1}}{\partial out_{h1}}=\frac{\partial E_{o1}}{\partial net_{o1}}\times\frac{\partial net_{o1}}{\partial out_{h1}} \qquad \frac{\partial E_{o2}}{\partial out_{h1}}=\frac{\partial E_{o2}}{\partial net_{o2}}\times\frac{\partial net_{o2}}{\partial out_{h1}}$$

$$\frac{\partial E_{total}}{\partial w_1}=\left(\frac{\partial E_{o1}}{\partial net_{o1}}\times\frac{\partial net_{o1}}{\partial out_{h1}}+\frac{\partial E_{o2}}{\partial net_{o2}}\times\frac{\partial net_{o2}}{\partial out_{h1}}\right)\times\frac{\partial out_{h1}}{\partial net_{h1}}\times\frac{\partial net_{h1}}{\partial w_1}$$

$$\frac{\partial E_{total}}{\partial w_1}=\left(\frac{\partial E_{o1}}{\partial out_{o1}}\times\frac{\partial out_{o1}}{\partial net_{o1}}\times\frac{\partial net_{o1}}{\partial out_{h1}}+\frac{\partial E_{o2}}{\partial out_{o2}}\times\frac{\partial out_{o2}}{\partial net_{o2}}\times\frac{\partial net_{o2}}{\partial out_{h1}}\right)\times\frac{\partial out_{h1}}{\partial net_{h1}}\times\frac{\partial net_{h1}}{\partial w_1}$$

为迭代做准备

$$\frac{\partial E_{total}}{\partial w_1}=\left(\frac{\partial E_{o1}}{\partial out_{o1}}\times\frac{\partial out_{o1}}{\partial net_{o1}}\times\frac{\partial net_{o1}}{\partial out_{h1}}+\frac{\partial E_{o2}}{\partial out_{o2}}\times\frac{\partial out_{o2}}{\partial net_{o2}}\times\frac{\partial net_{o2}}{\partial out_{h1}}\right)\times\frac{\partial out_{h1}}{\partial net_{h1}}\times\frac{\partial net_{h1}}{\partial w_1}$$

$$\frac{\partial E_{total}}{\partial w_1}=\left(\sum_i\frac{\partial E_{\boxed{oi}}}{\partial out_{oi}}\times\frac{\partial out_{oi}}{\partial net_{oi}}\times\frac{\partial net_{oi}}{\partial out_{h1}}\right)\times\frac{\partial out_{h1}}{\partial net_{h1}}\times\frac{\partial net_{h1}}{\partial w_1}$$

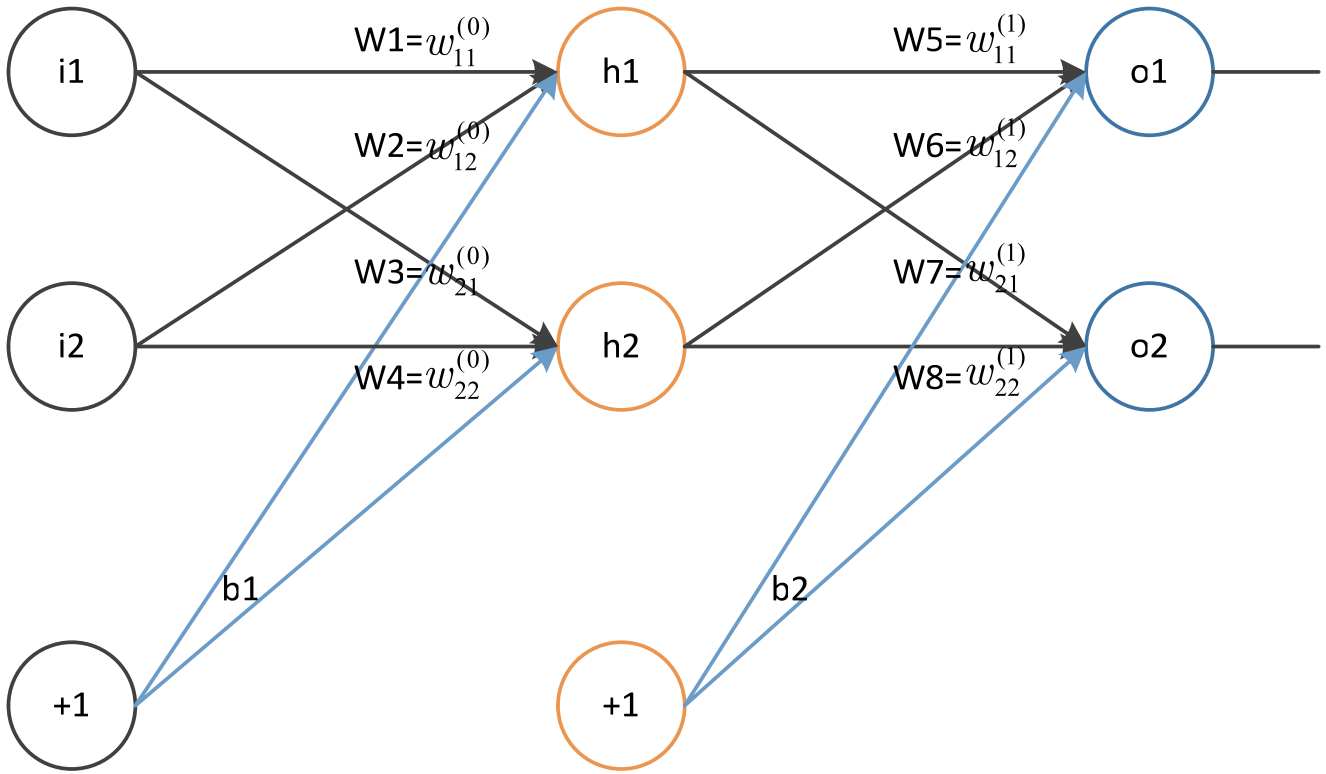
$$=\left(\sum_i\frac{\partial E_{\boxed{total}}}{\partial out_{oi}}\times\frac{\partial out_{oi}}{\partial net_{oi}}\times\frac{\partial net_{oi}}{\partial out_{h1}}\right)\times\frac{\partial out_{h1}}{\partial net_{h1}}\times\frac{\partial net_{h1}}{\partial w_1}$$

$$=\left(\sum_i\delta_{oi}^2\times\frac{\partial net_{oi}}{\partial out_{h1}}\right)\times\frac{\partial out_{h1}}{\partial net_{h1}}\times\frac{\partial net_{h1}}{\partial w_1}$$

$$\delta_{o1}^2=\frac{\partial E_{total}}{\partial net_{o1}}=\frac{\partial E_{total}}{\partial out_{o1}}\times\frac{\partial out_{o1}}{\partial net_{o1}}$$

$$\boxed{\frac{\partial net_{o1}}{\partial out_{h1}}=w_5,\quad\frac{\partial net_{o2}}{\partial out_{h1}}}=w_7$$

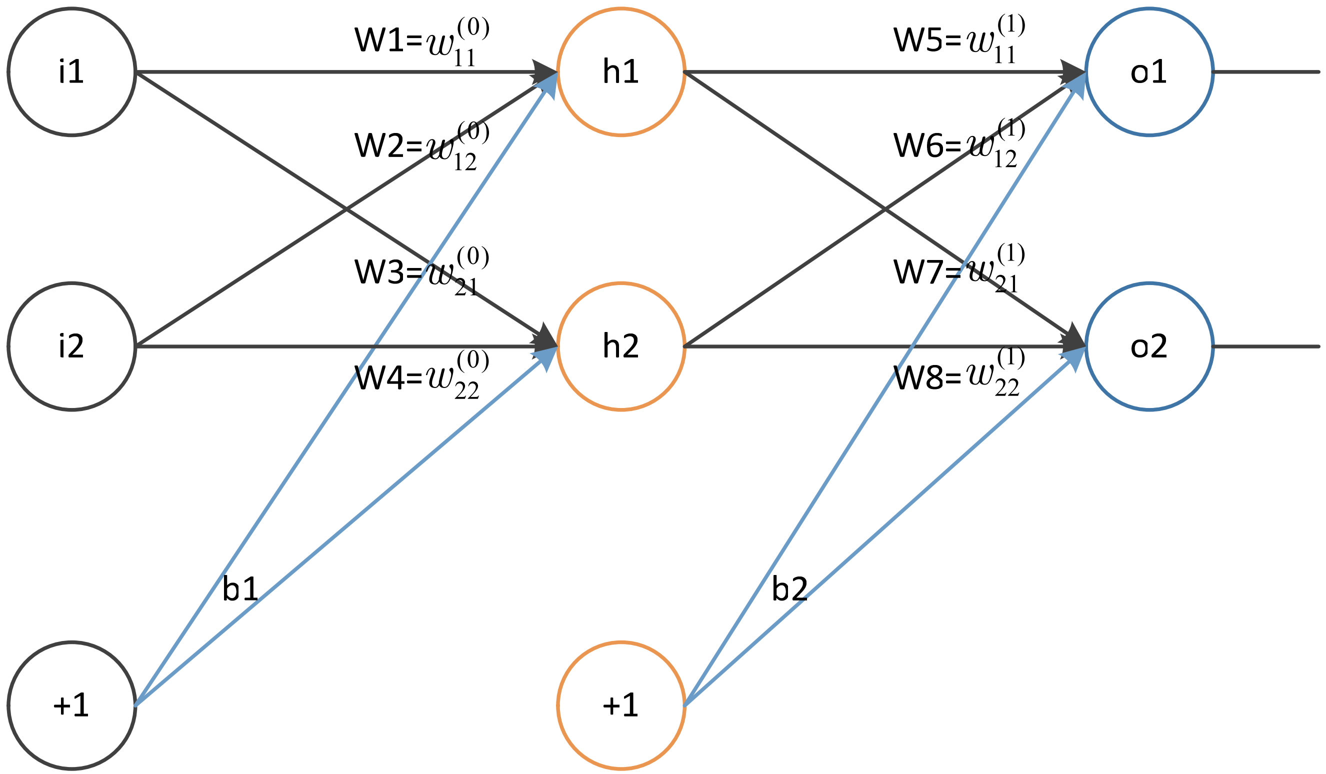
一般情况下的迭代



$$\frac{\partial E_{total}}{\partial w_{11}^{(0)}} = \left(\sum_i \delta_{oi}^2 \times \frac{\partial net_{oi}}{\partial out_{h1}}\right) \times \frac{\partial out_{h1}}{\partial net_{h1}} \times \frac{\partial net_{h1}}{\partial w_{11}^{(0)}} = \left(\sum_i \delta_{oi}^2 \times w_{i1}^{(1)}\right) \times \frac{\partial out_{h1}}{\partial net_{h1}} \times \frac{\partial net_{h1}}{\partial w_{11}^{(0)}}$$

$$\frac{\partial E_{total}}{\partial w_{\boxed{1k}}^{(0)}} = \left(\sum_i \delta_{\boxed{oi}}^2 \times \frac{\partial net_{oi}}{\partial out_{h1}}\right) \times \frac{\partial out_{h1}}{\partial net_{h1}} \times \frac{\partial net_{h1}}{\partial w_{1k}^{(0)}} = \left(\sum_i \delta_{oi}^2 \times w_{i1}^{(1)}\right) \times \frac{\partial out_{h1}}{\partial net_{h1}} \times \frac{\partial net_{h1}}{\partial w_{1k}^{(0)}}$$

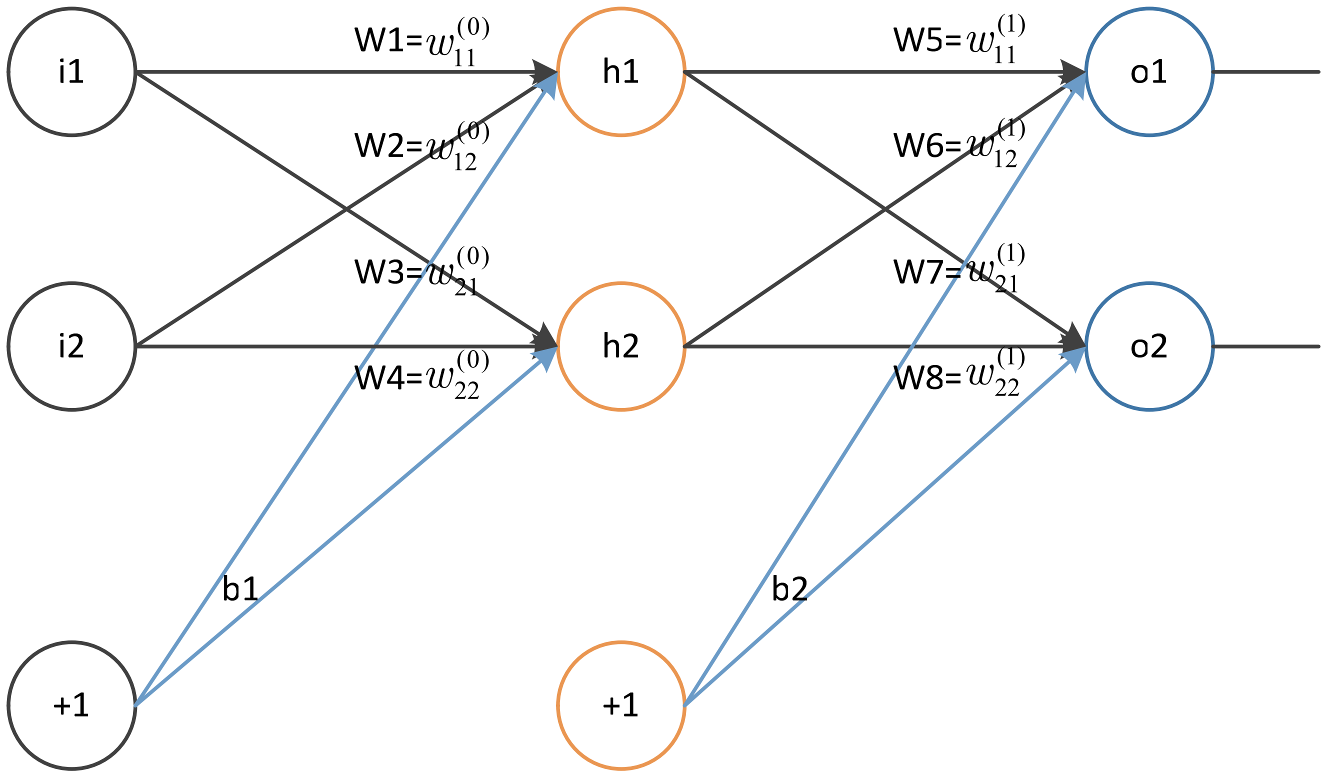
一般情况下的迭代



$$\frac{\partial E_{total}}{\partial w_{1k}^{(0)}} = \left(\sum_i \delta_{oi}^2 \times \frac{\partial net_{oi}}{\partial out_{h1}} \right) \times \frac{\partial out_{h1}}{\partial net_{h1}} \times \frac{\partial net_{h1}}{\partial w_{1k}^{(0)}} = \left(\sum_i \delta_{oi}^2 \times w_{1i}^{(1)} \right) \times \frac{\partial out_{h1}}{\partial net_{h1}} \times \frac{\partial net_{h1}}{\partial w_{1k}^{(0)}}$$

$$\frac{\partial E_{total}}{\partial w_{jk}^{(0)}} = \left(\sum_i \delta_{oi}^2 \times \frac{\partial net_{oi}}{\partial out_{hj}} \right) \times \frac{\partial out_{hj}}{\partial net_{hj}} \times \frac{\partial net_{hj}}{\partial w_{jk}^{(0)}} = \left(\sum_i \delta_{oi}^2 \times w_{ji}^{(1)} \right) \times \frac{\partial out_{hj}}{\partial net_{hj}} \times \frac{\partial net_{hj}}{\partial w_{jk}^{(0)}}$$

一般情况下的迭代

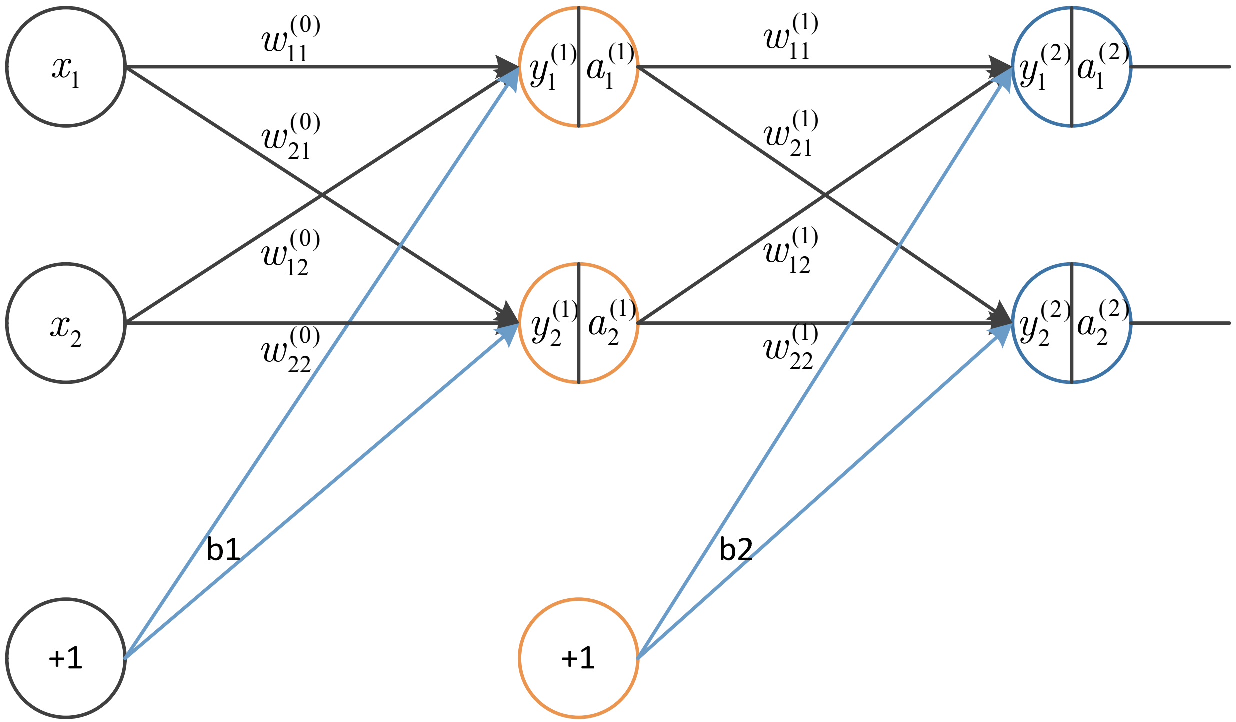


$$\frac{\partial E_{total}}{\partial w_{jk}^{(0)}}=\left(\sum_i \delta_{oi}^2\times w_{ji}^{(1)}\right)\times\frac{\partial out_{hj}}{\partial net_{hj}}\times\frac{\partial net_{hj}}{\partial w_{jk}^{(0)}}=\delta_{oi}^1\times\frac{\partial net_{hj}}{\partial w_{jk}^{(0)}}$$

$$\delta_{oi}^1=\left(\sum_i \delta_{oi}^2 w_{ji}^{(1)}\right)\times\frac{\partial out_{hj}^{(1)}}{\partial net_{hj}^{(1)}}\rightarrow\delta_{oi}^{l-1}=\left(\sum_i \delta_{oi}^l w_{ji}^{(l-1)}\right)\times\frac{\partial out_{hj}^{(l-1)}}{\partial net_{hj}^{(l-1)}}$$

$$\frac{\partial E_{total}}{\partial w_{jk}^{(l-1)}}=\delta_{oi}^{l-1}\times\frac{\partial net_{hj}^{(l-1)}}{\partial w_{jk}^{(l-1)}}$$

一般情况下的迭代



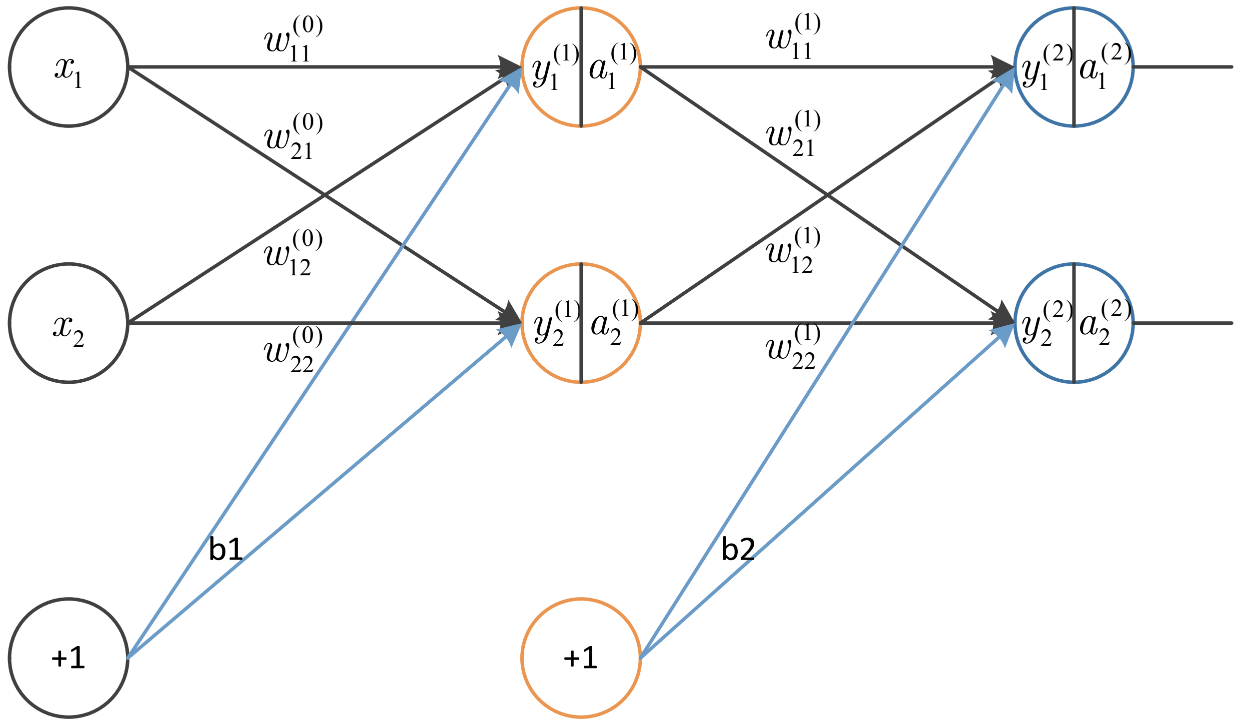
$$\frac{\partial E_{total}}{\partial w_{jk}^{(0)}} = \left(\sum_i \delta_{oi}^2 \times w_{ji}^{(1)}\right) \times \frac{\partial a_j^{(1)}}{\partial y_j^{(1)}} \times \frac{\partial y_j^{(1)}}{\partial w_{jk}^{(0)}} = \delta_{oi}^1 \times \frac{\partial y_j^{(1)}}{\partial w_{jk}^{(0)}}$$

$$\delta_{oi}^1 = \left(\sum_i \delta_{oi}^2 w_{ji}^{(1)}\right) \times \frac{\partial a_j^{(1)}}{\partial y_j^{(1)}} \rightarrow \delta_{oi}^{L-1} = \left(\sum_i \delta_{oi}^L w_{ji}^{(L-1)}\right) \times \frac{\partial a_j^{(L-1)}}{\partial y_j^{(L-1)}}$$

$$\delta_{oi}^L = \frac{\partial E_{total}}{\partial y_i^{(L)}} = \frac{\partial E_{total}}{\partial a_i^{(L)}} \times \frac{\partial a_i^{(L)}}{\partial y_i^{(L)}}$$

$$\frac{\partial E_{total}}{\partial w_{jk}^{(L-2)}} = \delta_{oi}^{L-1} \times \frac{\partial y_j^{(L-1)}}{\partial w_{jk}^{(L-2)}}, \quad L = 2$$

一般情况下的迭代



$$\delta_{oi}^{1}=\left(\sum_i \delta_{oi}^{2} w_{ji}^{(1)}\right)\times\frac{\partial a_j^{(1)}}{\partial y_j^{(1)}} \to \delta_{oi}^{L-1}=\left(\sum_i \delta_{oi}^{L} w_{ji}^{(L-1)}\right)\times\frac{\partial a_j^{(L-1)}}{\partial y_j^{(L-1)}}$$

$$\delta_{oi}^{l}=\left(\sum_i \delta_{oi}^{l+1} w_{ji}^{(l)}\right)\times\frac{\partial a_j^{(l)}}{\partial y_j^{(l)}}$$

$$\frac{\partial E_{total}}{\partial w_{jk}^{(l-1)}}=\delta_{oi}^{l}\times\frac{\partial y_j^{(l)}}{\partial w_{jk}^{(l-1)}},\ l=1,2,\ldots,L-1$$

Cross entropy

$$E_{total} = \frac{1}{2}\left(target_{o1} - out_{o1}\right)^2 + \frac{1}{2}\left(target_{o2} - out_{o2}\right)^2$$

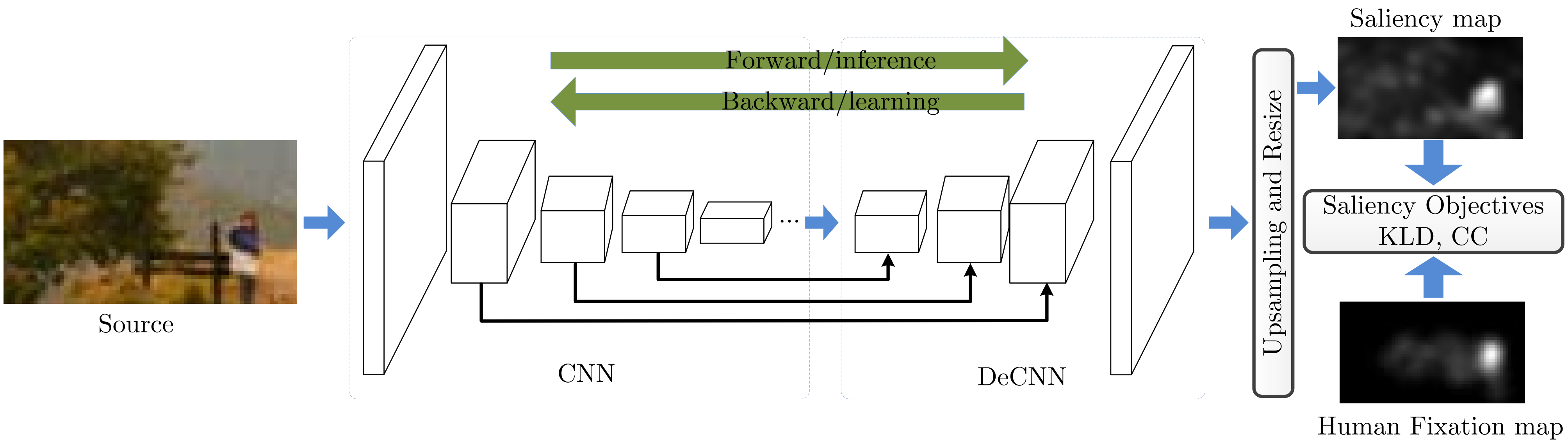
$$\delta_{oi}^{2} = -\left(target_{oi} - out_{oi}\right) \times \frac{\partial out_{oi}}{\partial net_{oi}}$$

$$E_{entrop} = target_{o1} \log out_{o1} + target_{o2} \log out_{o2}$$

$$\delta_{oi}^{2} = \frac{\partial E_{entropy}}{\partial y_i^{(2)}} = \frac{\partial E_{entropy}}{\partial a_i^{(2)}} \times \frac{\partial a_i^{(2)}}{\partial y_i^{(2)}}$$

$$\frac{\partial E_{entropy}}{\partial out_{oi}} \times \frac{\partial out_{oi}}{\partial net_{oi}} = \frac{target_{oi}}{out_{oi}} \times \frac{\partial out_{oi}}{\partial net_{oi}}$$

CNN


Saliency map
Forward/inference
Backward/learning
Source
CNN
DeCNN
Upsampling and Resize
Saliency Objectives
KLD, CC
Human Fixation map

CNN

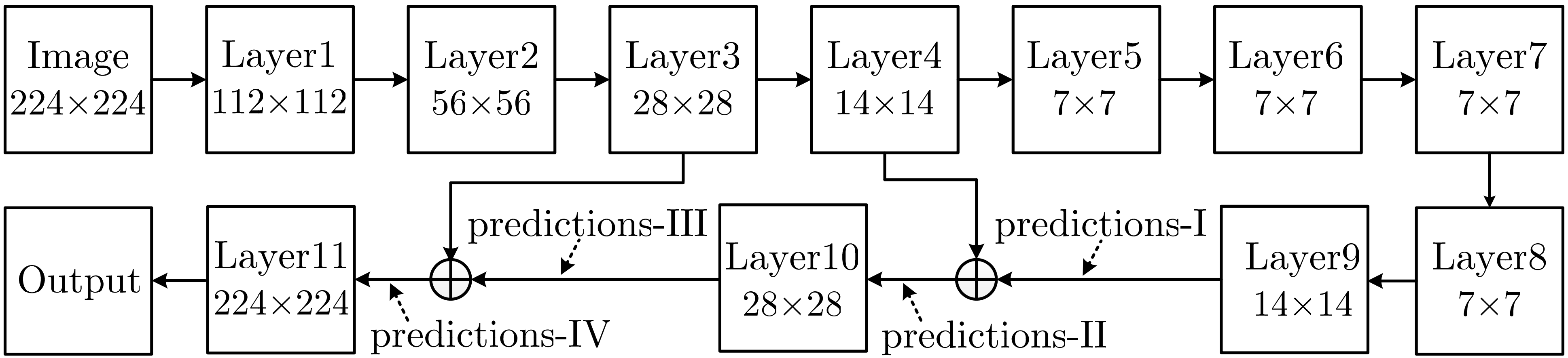
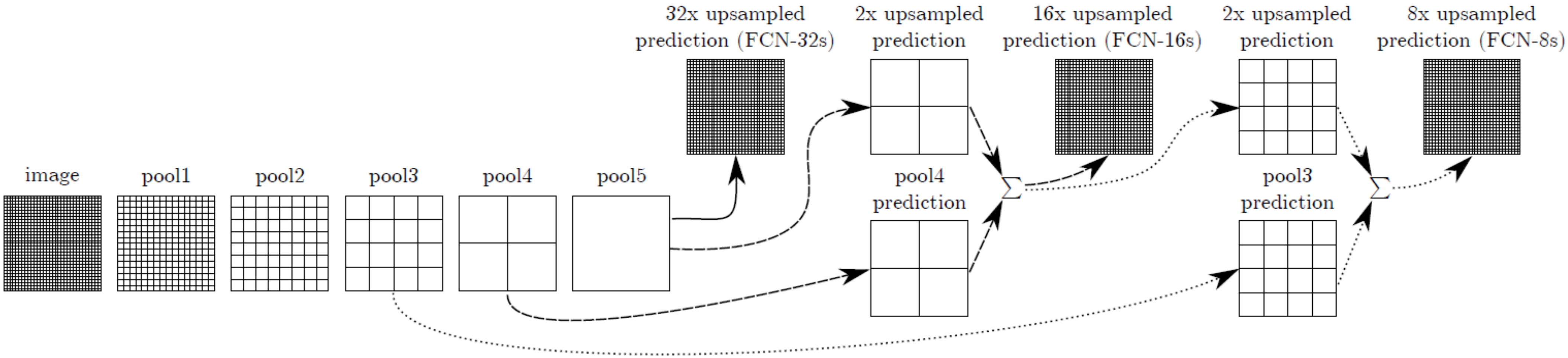


Image
224×224
Layer1
112×112
Layer2
56×56
Layer3
28×28
Layer4
14×14
Layer5
7×7
Layer6
7×7
Layer7
7×7
Output
Layer11
224×224
predictions-III
predictions-IV
Layer10
28×28
predictions-I
predictions-II
Layer9
14×14
Layer8
7×7

Receptive field



其他

- 总训练次数
- Batch大小: 一次训练的图像张数
- Epoch: 整个数据集训练的次数
- 10,000张图像, 每个batch为4张图像, 总训练100,000次, 请问, 共训练了几个epoch?
- 10,000图像都训练一遍需要2,500个batch, 即2,500次, 100,000/2,500=40个epoch

CNN

Layer	1	2	3	4	5	6	7	8	9	10	11
Type	conv	conv	conv	conv	conv	conv	conv	conv	deconv	deconv	deconv
Conv. kernel size	3×3	3×3	3×3	3×3	3×3	7×7	1×1	1×1	4×4	4×4	16×16
#kernel input channels	3	64	128	256	512	512	4096	4096	256	512	256
#kernel output channels	64	128	256	512	512	4096	4096	256	512	256	256
Conv. stride	1	1	1	1	1	1	1	1	2	2	8
Pooling size	2×2	2×2	2×2	2×2	2×2	-	-	-	-	-	-
Pooling stride	2	2	2	2	2	2	2	-	-	-	-
Zero-padding size	1	1	1	1	1	0	0	0	1	1	4
Spatial input size	224×224	112×112	56×56	28×28	14×14	7×7	7×7	7×7	7×7	14×14	28×28
Spatial output size	112×112	56×56	28×28	14×14	7×7	7×7	7×7	7×7	14×14	28×28	224×224

CNN for Saliency

Model	Toronto				Kootstra				MIT				Cerf				ImgSal				Avg.			
	KLD	CC	NSS	SIM	KLD	CC	NSS	SIM	KLD	CC	NSS	SIM	KLD	CC	NSS	SIM	KLD	CC	NSS	SIM	KLD	CC	NSS	SIM
Ours	**0.730**	**0.677**	**1.863**	**0.522**	**0.330**	**0.562**	**0.824**	**0.687**	**0.862**	**0.599**	**1.750**	0.478	**0.839**	**0.617**	1.863	**0.488**	**0.501**	**0.747**	**1.811**	**0.613**	**0.652**	**0.640**	**1.622**	**0.557**
BMS	0.983	0.447	1.519	0.447	0.383	0.445	0.675	0.637	1.141	0.403	1.247	0.409	1.141	0.390	1.230	0.411	0.698	0.615	1.545	0.547	0.870	0.460	1.243	0.490
QDCT	1.041	0.476	1.362	0.424	0.419	0.389	0.567	0.651	1.209	0.364	1.009	0.384	1.168	0.393	1.201	0.398	0.766	0.554	1.374	0.515	0.920	0.435	1.119	0.476
SigSal	1.009	0.497	1.378	0.435	0.406	0.397	0.561	0.657	1.184	0.374	1.099	0.393	1.153	0.398	1.191	0.405	0.739	0.560	1.358	0.526	0.898	0.445	1.117	0.483
Judd	1.127	0.411	1.151	0.402	0.442	0.343	0.507	0.641	1.253	0.342	1.017	0.370	1.254	0.343	1.033	0.369	0.831	0.510	1.259	0.491	0.981	0.390	0.993	0.455
AWS	1.089	0.432	1.240	0.421	0.555	0.349	0.562	0.614	1.225	0.361	1.142	0.396	1.328	0.311	1.001	0.379	0.787	0.553	1.398	0.522	0.997	0.401	1.069	0.466
LG	1.213	0.350	1.019	0.384	0.480	0.308	0.498	0.626	1.309	0.301	0.943	0.360	1.413	0.233	0.745	0.341	0.888	0.450	1.117	0.477	1.061	0.328	0.864	0.438
HFT	1.062	0.431	1.204	0.429	0.500	0.372	0.532	0.624	1.218	0.350	1.040	0.394	1.194	0.373	1.111	0.401	0.708	0.589	1.439	0.543	0.936	0.423	1.065	0.476
ITTI	0.969	0.478	1.297	0.448	0.460	0.373	0.525	0.639	1.135	0.387	1.103	0.407	1.137	0.384	1.104	0.410	0.655	0.579	1.377	0.556	0.871	0.440	1.081	0.492
AER	1.031	0.483	1.335	0.424	0.398	0.433	0.643	0.657	1.170	0.395	1.164	0.394	1.161	0.394	1.175	0.398	0.756	0.569	1.370	0.515	0.903	0.455	1.137	0.478
Shallow	0.804	0.621	1.116	0.517	0.715	0.518	0.717	0.662	0.875	0.590	1.731	**0.490**	0.867	0.597	**1.887**	0.486	0.611	0.680	1.632	0.606	0.774	0.601	1.417	0.552
eDN	1.117	0.498	1.248	0.405	–	–	–	–	1.176	0.493	1.297	0.386	–	–	–	–	–	–	–	–	–	–	–	–

Visual Comparison with Junting Model