Basic Mathematical Processes in Deep Neural Networks

For researchers interested in studying Earth science with deep learning.

All resources in lectures are available at https://github.com/MrXiaoXiao/DLiES

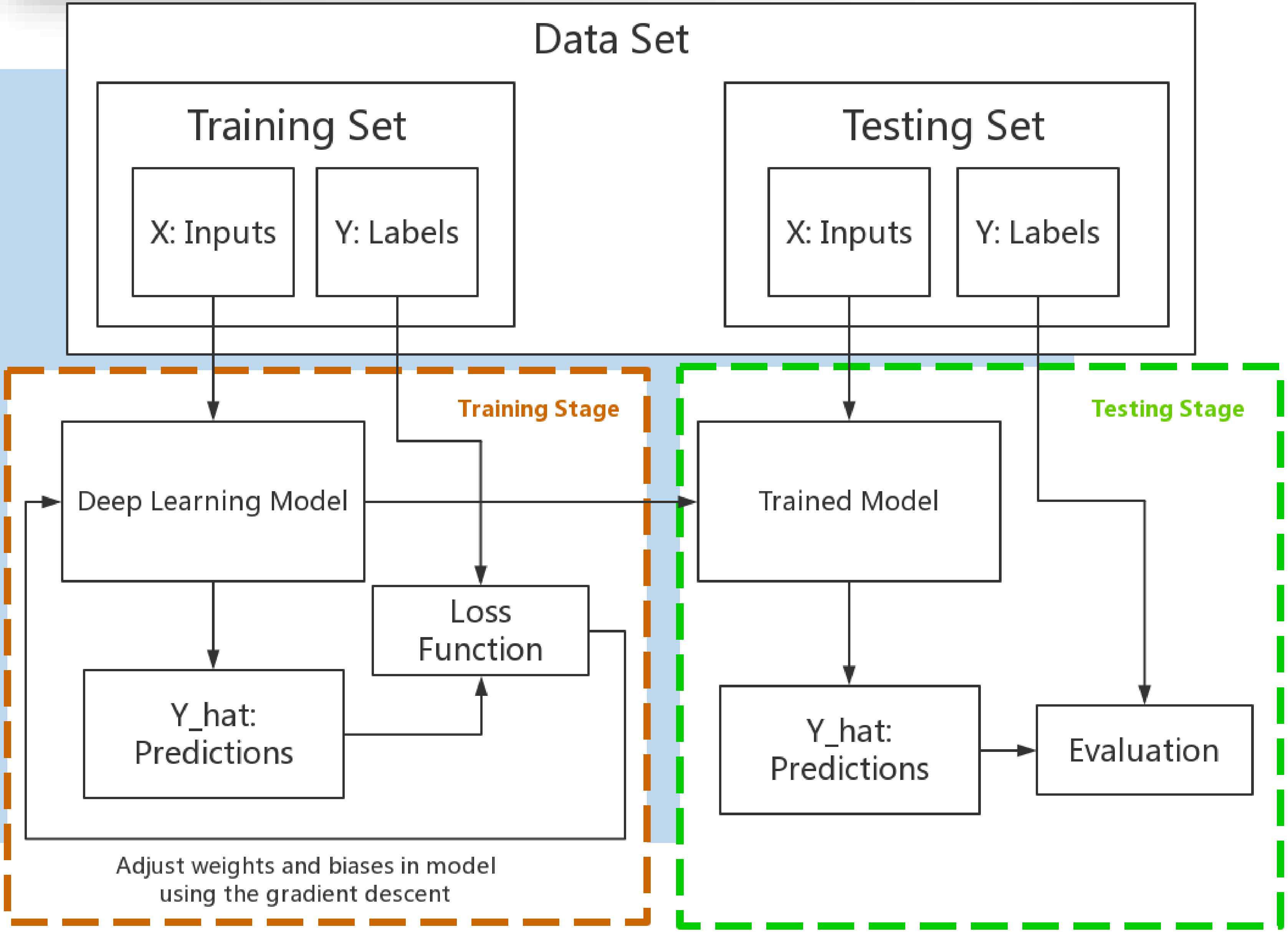
Deep Learning in Earth Science
Lecture 4
By Xiao Zhuowei

OUTLINES

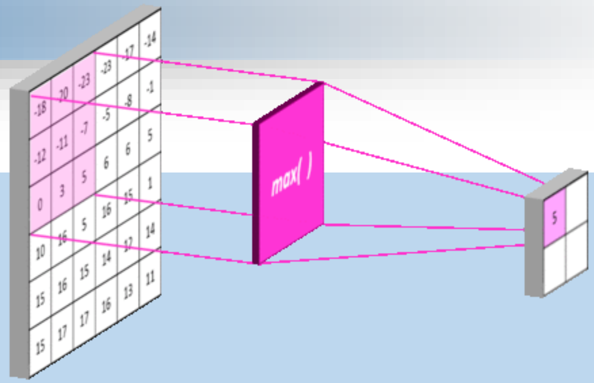
1. Cross Entropy
2. Back Propagation
3. Paper Reading
4. Discussions

Cross Entropy

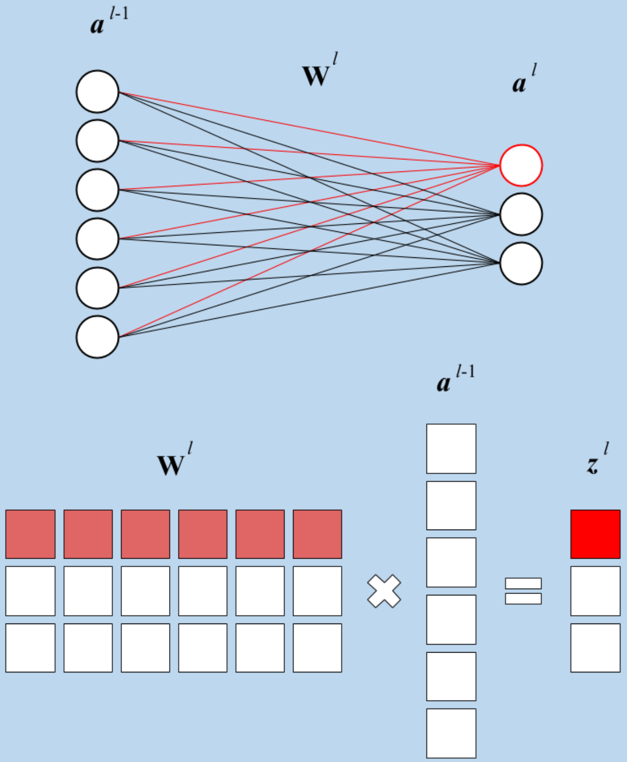
Quick Recap

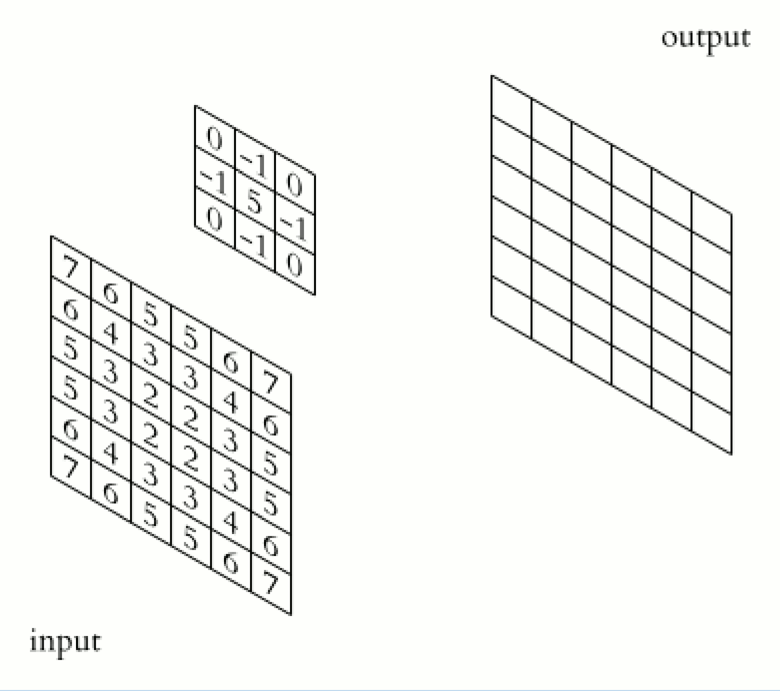


Cross Entropy



Quick Recap





Cross Entropy

Cross Entropy (CE)

A commonly used loss function in classification tasks.

What is cross-entropy and how to calculate it?

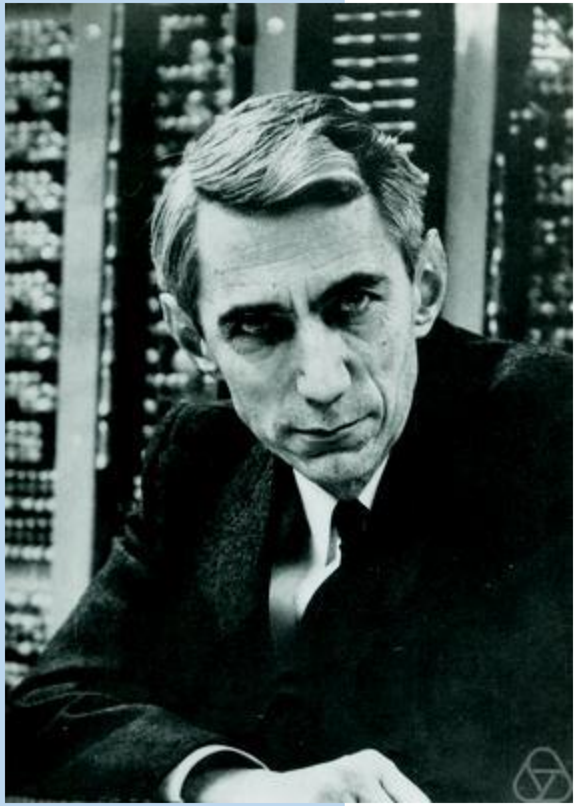
$$H(\mathrm{P};Q) = -\sum_{x\in X} p(x)\log q(x)$$

Entropy

***Information Entropy* (信息熵)**

How to measure the amount of useful information?

$$H(P) = -\sum_{x \in X} p(x)\ log_2 p(x)$$



Claude Elwood Shannon (1916–2001)

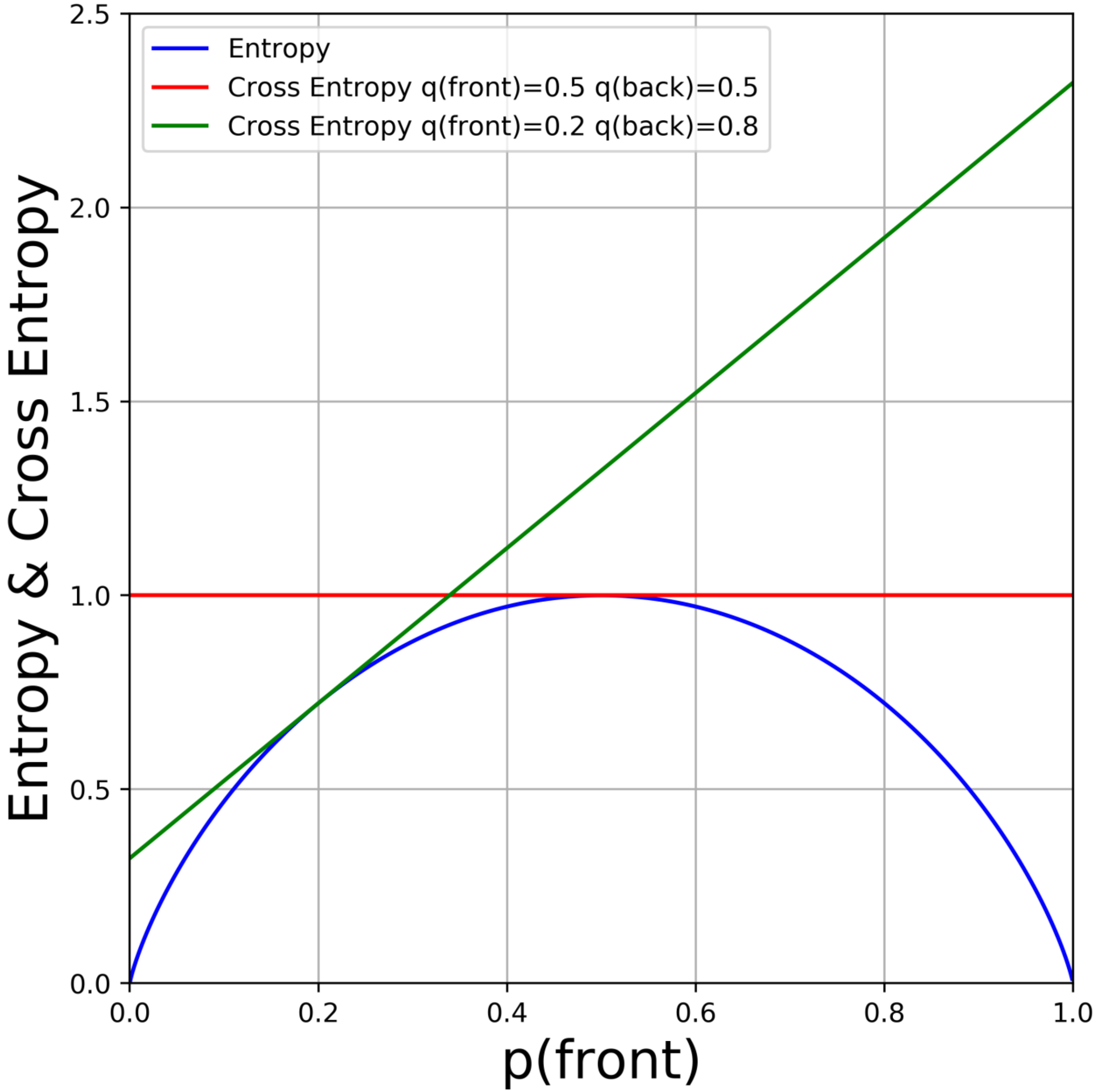
https://en.wikipedia.org/wiki/Claude_Shannon

Entropy

Entropy of flipping a coin

$$H(P) = -\sum_{x \in X} p(x)\ log_2\, p(x)$$

$$H(\mathrm{P};Q) = -\sum_{x \in X} p(x)\, log_2\, q(x)$$

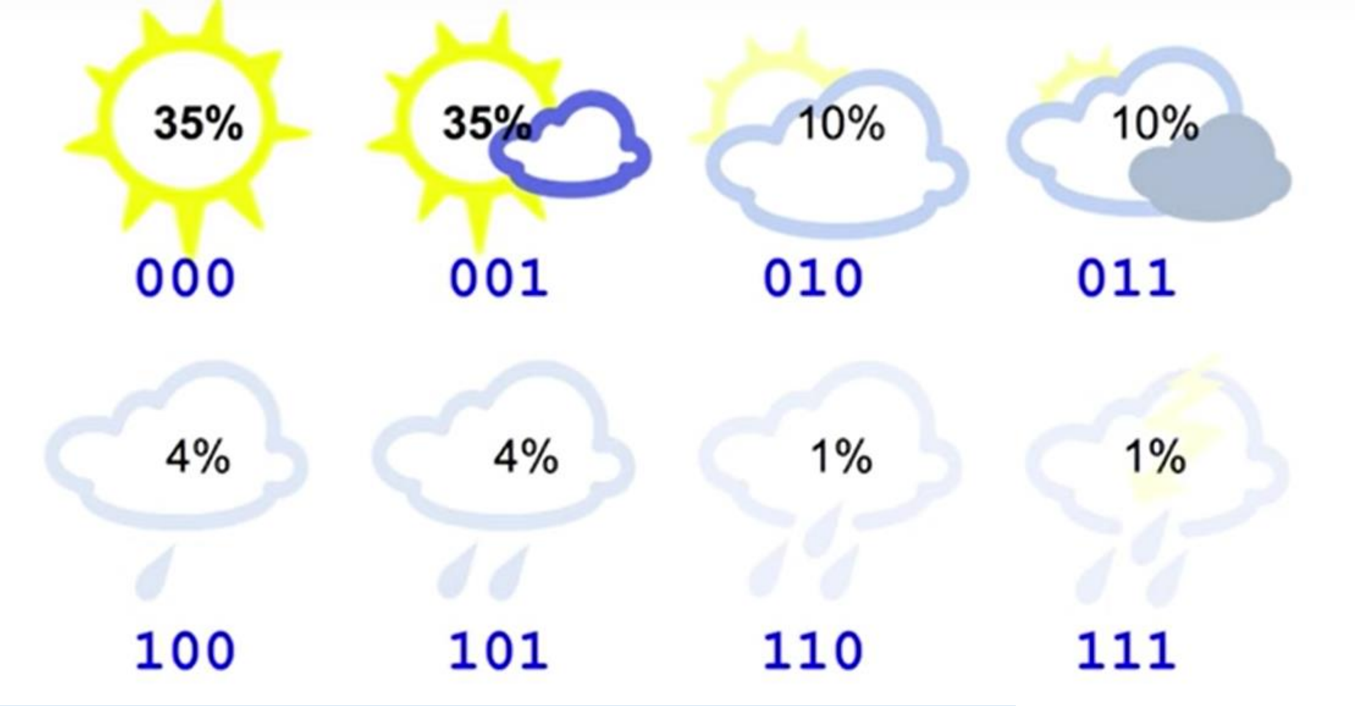


Cross Entropy

Cross Entropy: Average message length.

https://www.youtube.com/watch?v=ErfnhcEV1O8

Weather Example



$$H(\mathrm{P};Q) = -\sum_{x\in X} p(x)\, log_2\, q(x)$$

$$= \sum_{x\in X} p(x) \; * log_2 8 = 1 * 3.0 = 3.0\ bits$$

$$H(P) = -\sum_{x\in X} p(x)\; log_2\, p(x)$$

$$= -0.35 * log_2(0.35) + (-0.35) * log_2(0.35) + (-0.1) * log_2(0.1) + (-0.1) * log_2(0.1)$$
$$+(-0.04) * log_2(0.04) + (-0.04) * log_2(0.04) + (-0.01) * log_2(0.01) + (-0.01) * log_2(0.01)$$
$$= 2.228972\ bits$$

Cross Entropy

Cross Entropy: Average message length.

Weather Example

$$H(\mathrm{P};Q) = -\sum_{x \in X} p(x)\, log_2\, q(x)$$
$$= 0.35 * 2 + 0.35 * 2 + 0.1 * 3 + 0.1 * 3$$
$$+0.04*4 + 0.04*4 + 0.01*5 + 0.01*5$$
$$= 2.42\ bits$$

https://www.youtube.com/watch?v=ErfnhcEV1O8

Cross Entropy

For one-hot vector label in neural networks:

Class Name	Labels (P)	Predictions (Q)
cat	0	0.05
dog	0	0.40
deer	1	0.20
frog	0	0.80

$$H(\mathrm{P};Q) = -\sum_{x\in X} p(x)\, log_2\, q(x)$$

$$= 0 * \log(0.05) + 0 * \log(0.40)$$
$$+(-1.0) * \log(0.20) + 0 * \log(0.80)$$
$$= 1.6094$$

Other Loss Functions

Kullback–Leibler divergence

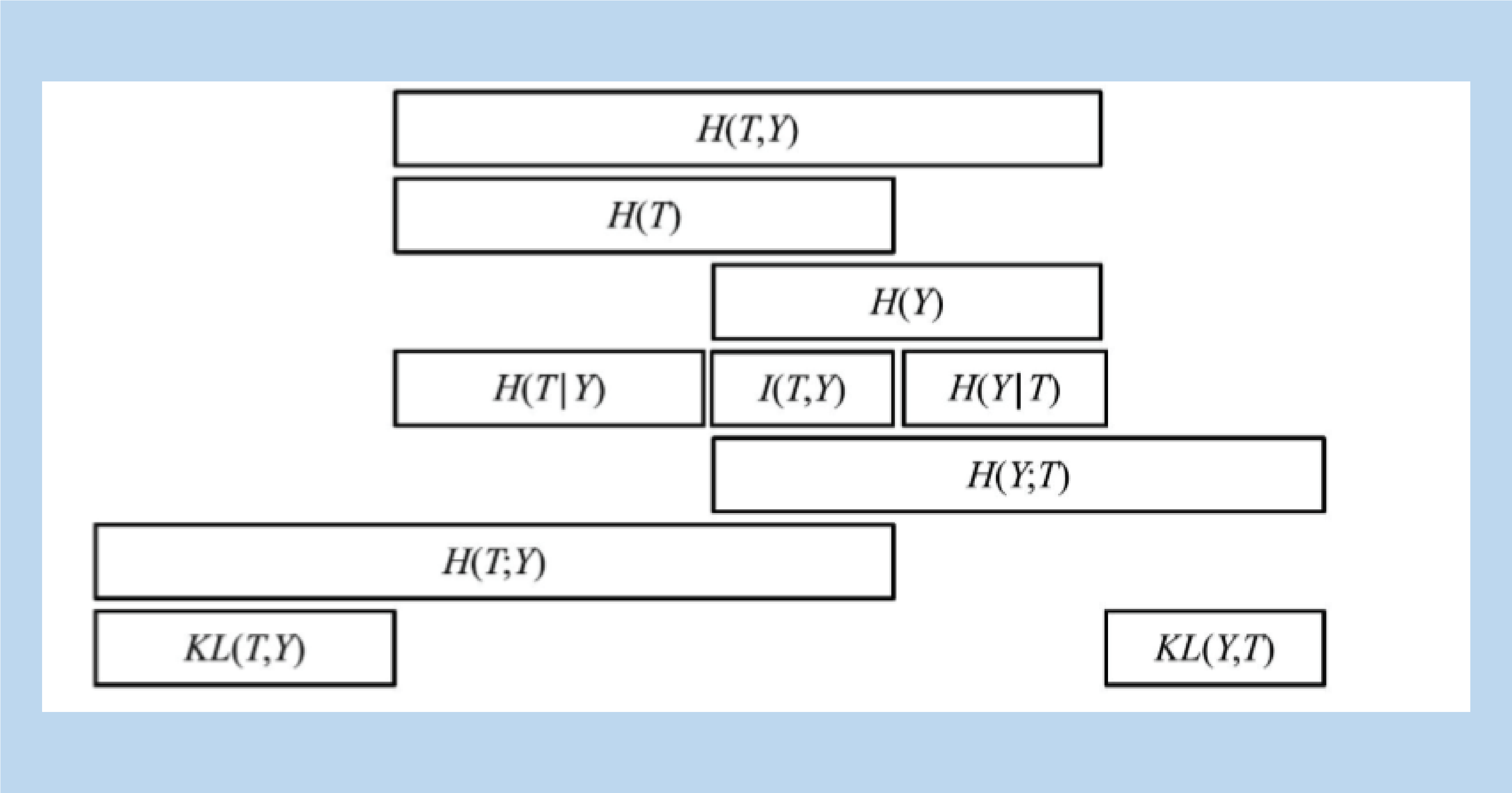
$$D_{KL}(P||Q) = -\sum_{x \in X} p(x)\, log_2\, q(x) + \sum_{x \in X} p(x)\, log_2\, p(x)$$

$$= H(\mathrm{P}; Q) - H(P)$$

Gibbs' inequality

$$-\sum_{i=1}^{n} p_i \log_2 p_i \le -\sum_{i=1}^{n} p_i \log_2 q_i$$

Cross Entropy



(Mackay, 2003)

Other Loss Functions

Mean Absolute Error (MAE) L1 loss

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n} = \frac{\sum_{i=1}^{n} |e_i|}{n}.$$

Mean Square Error (MSE) L2 loss

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2.$$

See for more on loss functions:
Picking Loss Functions - A comparison between MSE, Cross Entropy, and Hinge Loss
http://rohanvarma.me/Loss-Functions/

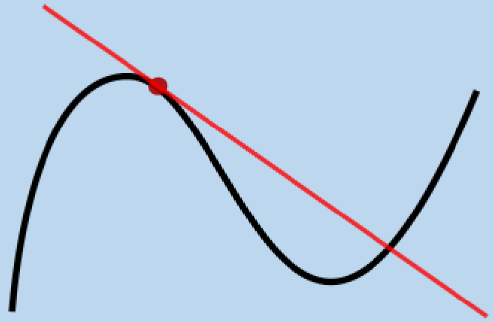
OUTLINES

1. Cross Entropy
2. Back Propagation
3. Paper Reading
4. Discussions

Back Propagation

How to we adjust weights and biases based on loss function?

Derivative and Chain rule



$$\frac{dy}{dx} = \frac{dy}{du} \cdot \frac{du}{dx}.$$

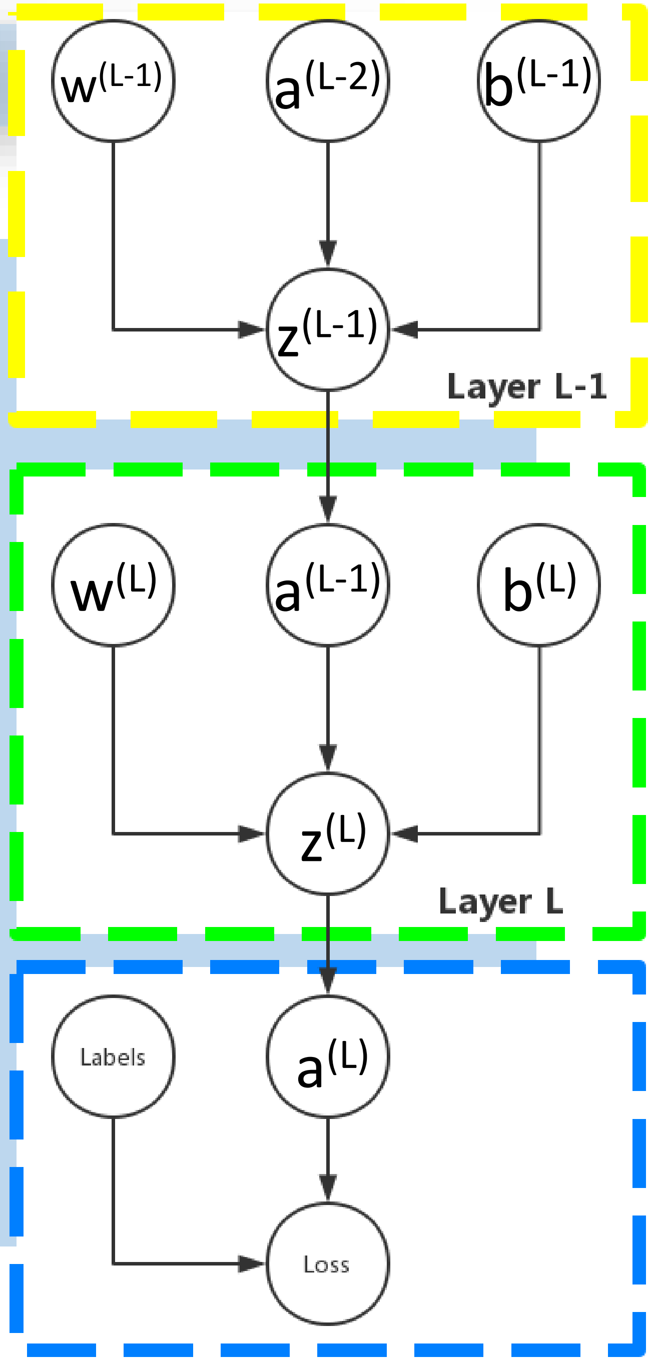
Back Propagation

MSE loss with sigmoid activation

$$Loss = (y - a^{(L)})^2$$

$$a^{(L)} = \frac{1}{1 + e^{-z^{(L)}}}$$

$$z^{(L)} = w^{(L)} a^{(L-1)} + b^{(L)}$$



Back Propagation

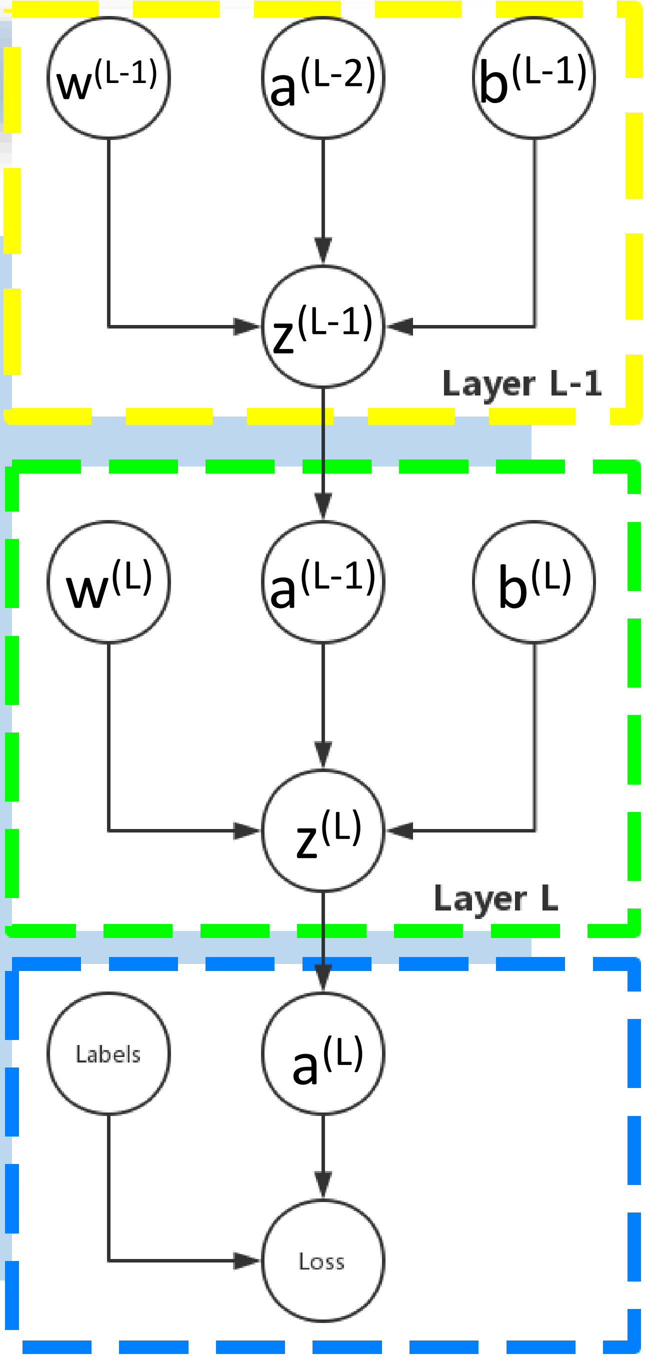
MSE loss with sigmoid activation

$$\frac{\partial Loss}{\partial w^{(L)}} = \frac{\partial Loss}{\partial a^{(L)}} * \frac{\partial a^{(L)}}{\partial z^{(L)}} * \frac{\partial z^{(L)}}{\partial w^{(L)}}$$

$$\frac{\partial Loss}{\partial a^{(L)}} = -2 * (y - a^{(L)})$$

$$\frac{\partial a^{(L)}}{\partial z^{(L)}} = \frac{1}{1+e^{-z^{(L)}}} * \left(1 - \frac{1}{1+e^{-z^{(L)}}}\right)$$

$$\frac{\partial z^{(L)}}{\partial w^{(L)}} = a^{(L-1)}$$

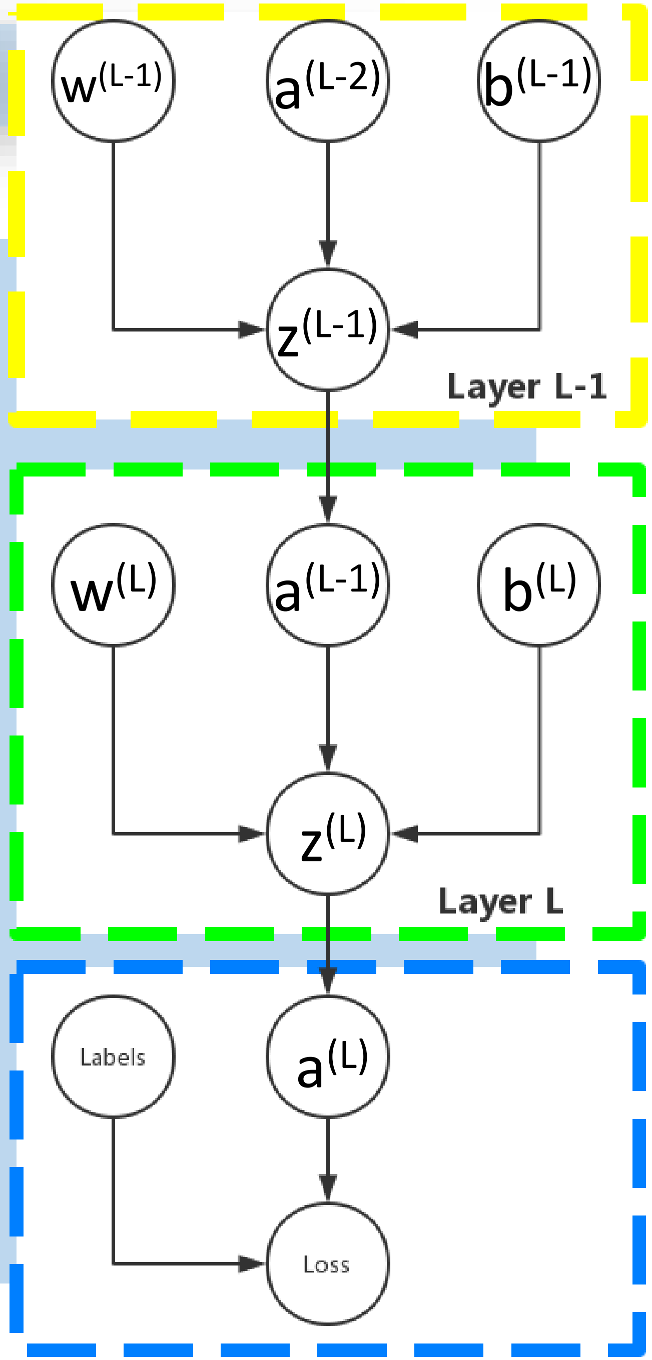


Back Propagation

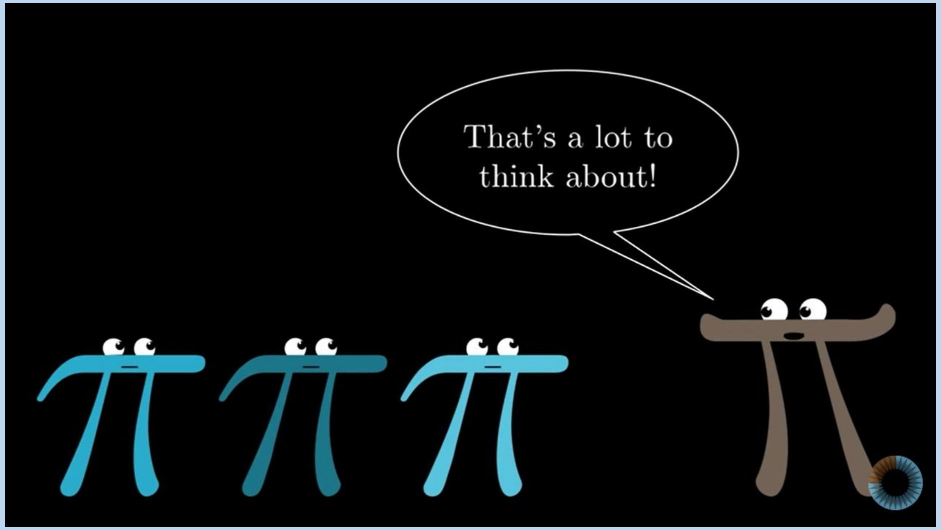
MSE loss with sigmoid activation

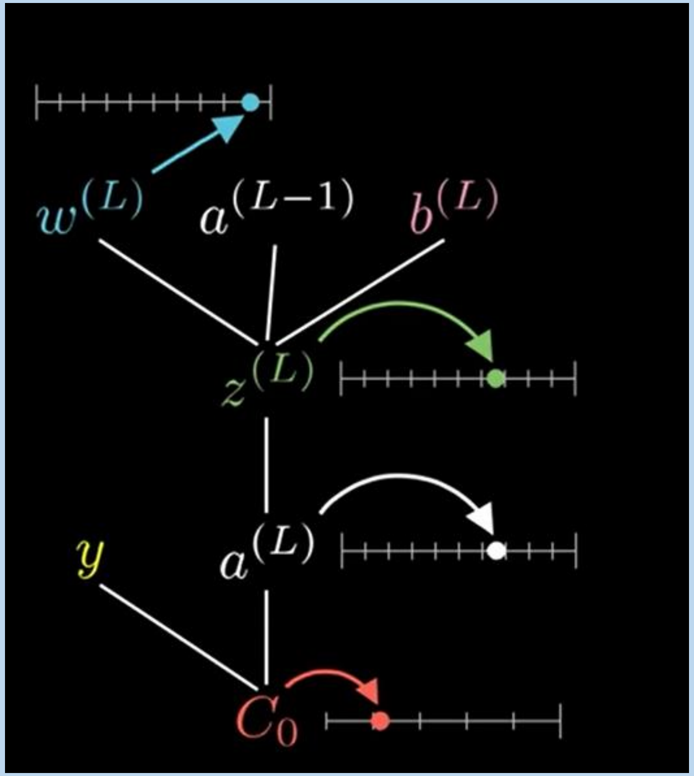
$$\frac{\partial Loss}{\partial w^{(L-1)}} = \frac{\partial Loss}{\partial a^{(L)}} * \frac{\partial a^{(L)}}{\partial z^{(L)}} * \frac{\partial z^{(L)}}{\partial a^{(L-1)}} * \frac{\partial a^{(L-1)}}{\partial z^{(L-1)}} * \frac{\partial z^{(L-1)}}{\partial w^{(L-1)}}$$

$$\boxed{\frac{\partial z^{(L)}}{\partial a^{(L-1)}} = w^{(L)}}$$



Back Propagation





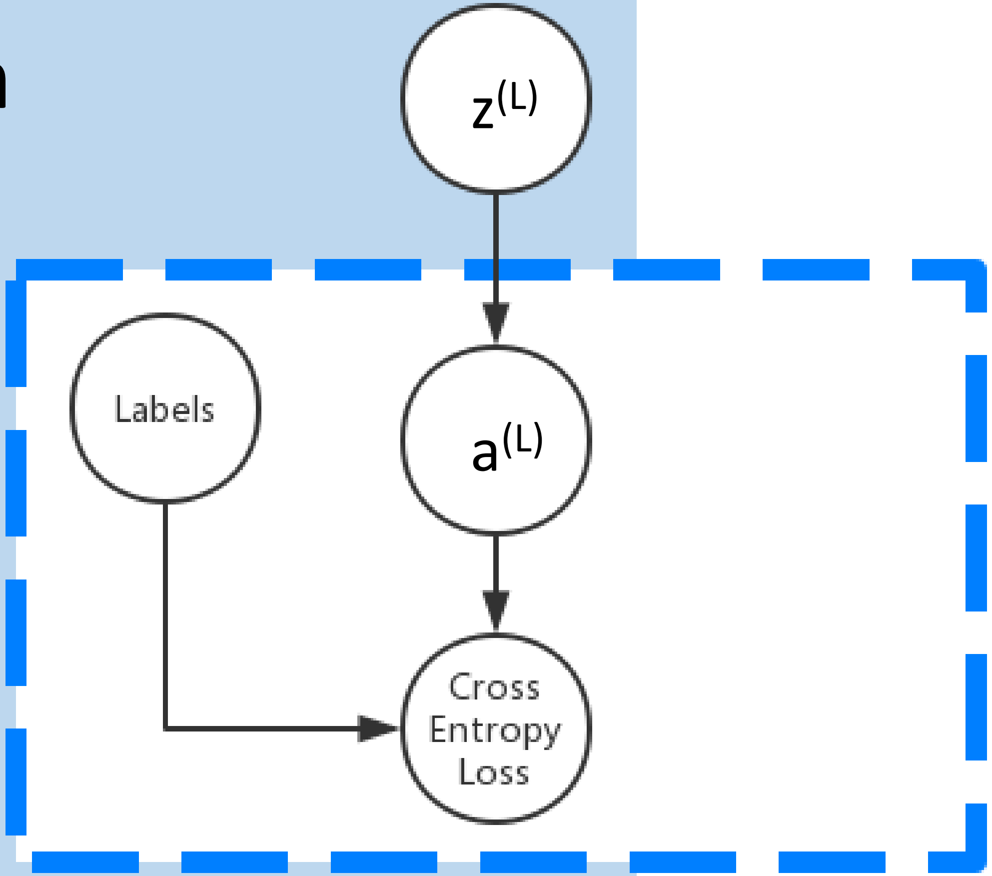
Recommend Video
https://www.youtube.com/watch?v=tIeHLnjs5U8

Back Propagation

CE loss with softmax activation

$$Loss = -\sum_i y_i \log a_i^{(L)}$$

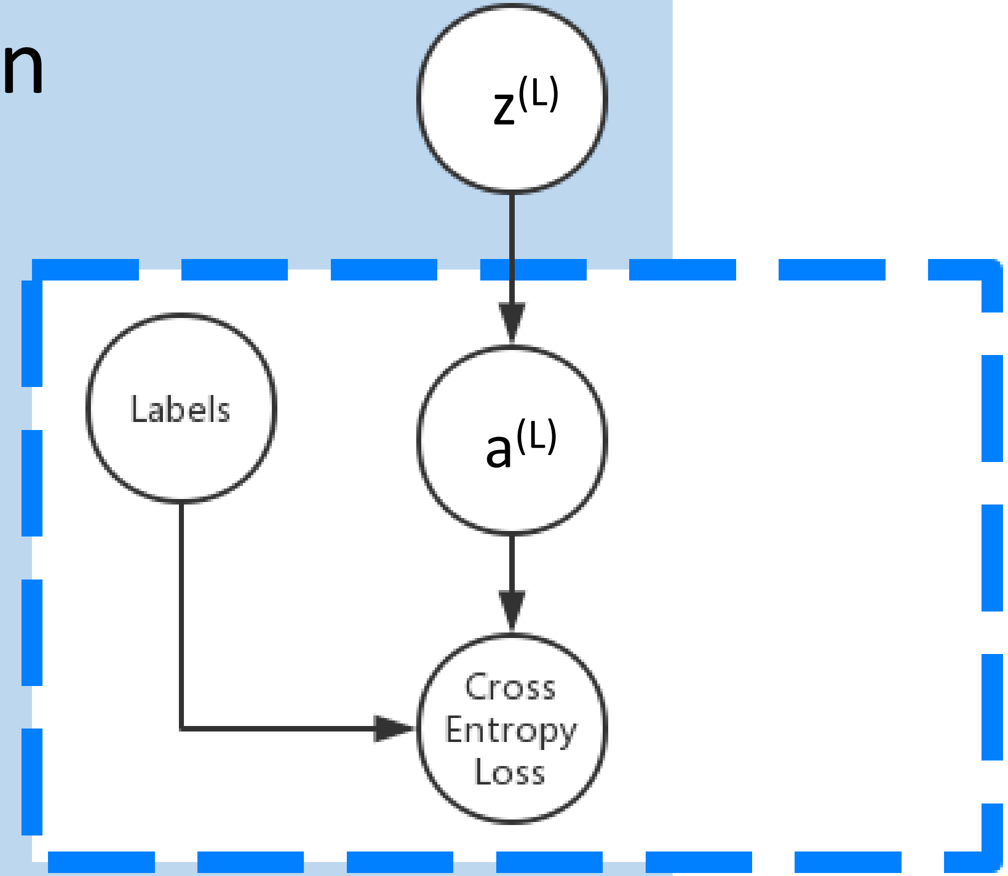
$$a_i^{(L)} = \frac{e^{z_i^{(L)}}}{\sum_{k=1}^{N} e^{z_k^{(L)}}} = \frac{e^{z_i^{(L)}+log(C)}}{\sum_{k=1}^{N} e^{z_k^{(L)}+log(C)}}$$



Back Propagation

CE loss with softmax activation

$$\frac{\partial Loss}{\partial a_i^{(L)}} = -\sum y_i * \frac{1}{a_i^{(L)}}$$

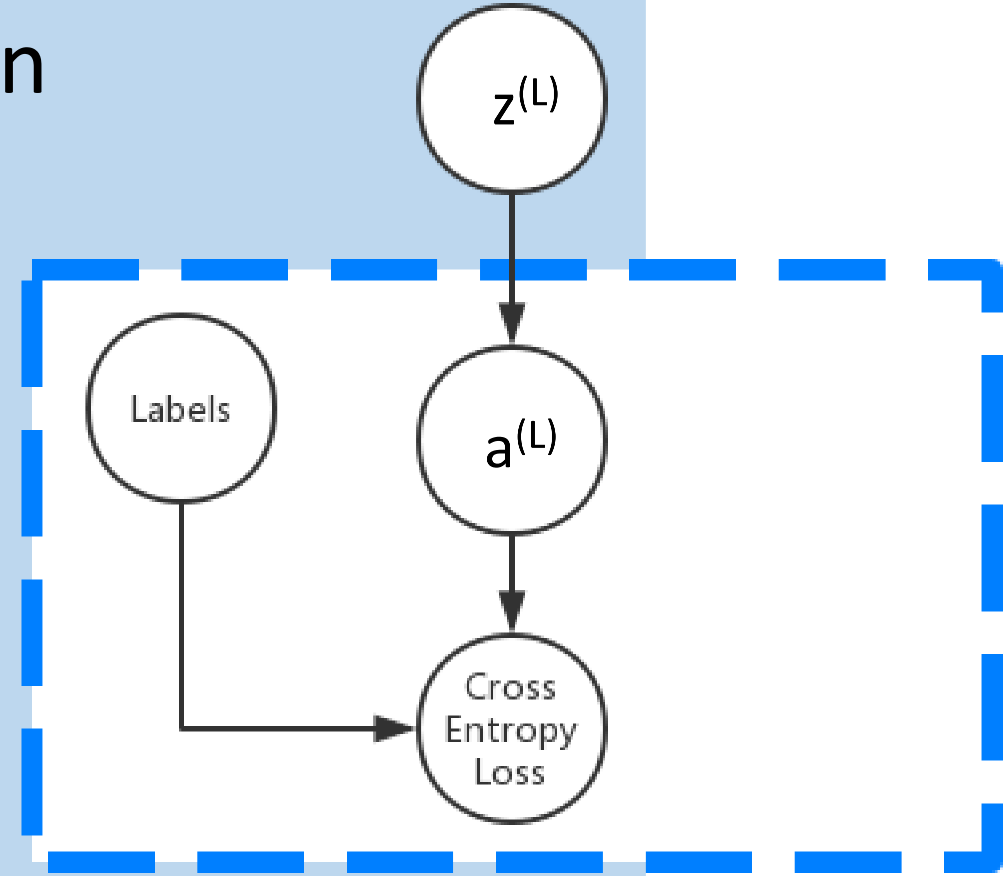


Back Propagation

CE loss with softmax activation

$$\frac{\partial a_i^{(L)}}{\partial z_j^{(L)}} = \frac{\partial \frac{e^{z_i^{(L)}}}{\sum_{k=1}^{N} e^{z_k^{(L)}}}}{\partial z_j^{(L)}}$$

$$f(x) = \frac{g(x)}{h(x)} \qquad f'(x) = \frac{g'(x)h(x) - h'(x)g(x)}{h(x)^2}$$



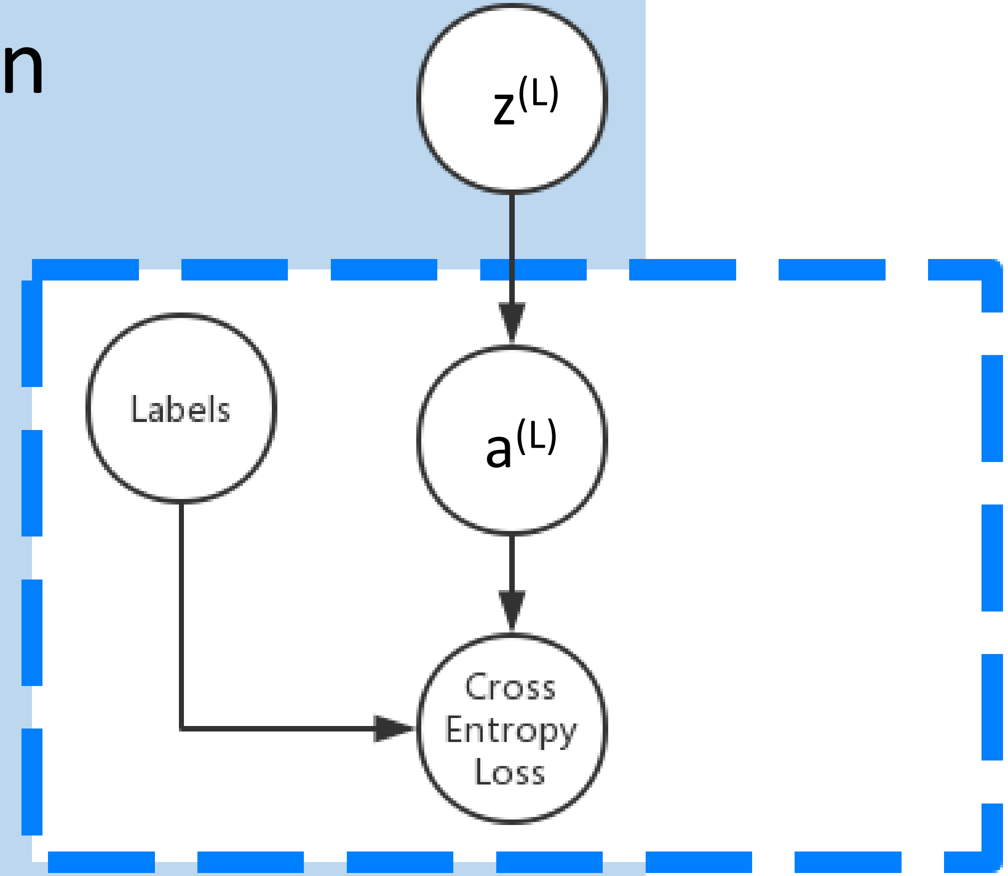
Back Propagation

CE loss with softmax activation

$$g\left(z_i^{(L)}\right) = e^{z_i^{(L)}} \qquad if\ i = j\text{:}\ \frac{\partial g\left(z_i^{(L)}\right)}{\partial z_j^{(L)}} = e^{z_i^{(L)}}$$

$$if\ i \neq j\text{:}\ \frac{\partial g\left(z_i^{(L)}\right)}{\partial z_j^{(L)}} = 0$$

$$\mathrm{h}\left(z_i^{(L)}\right) = \sum_{k=1}^{N} e^{z_k^{(L)}} \quad \frac{\partial h\left(z_i^{(L)}\right)}{\partial z_j^{(L)}} = e^{z_j^{(L)}}$$



Back Propagation

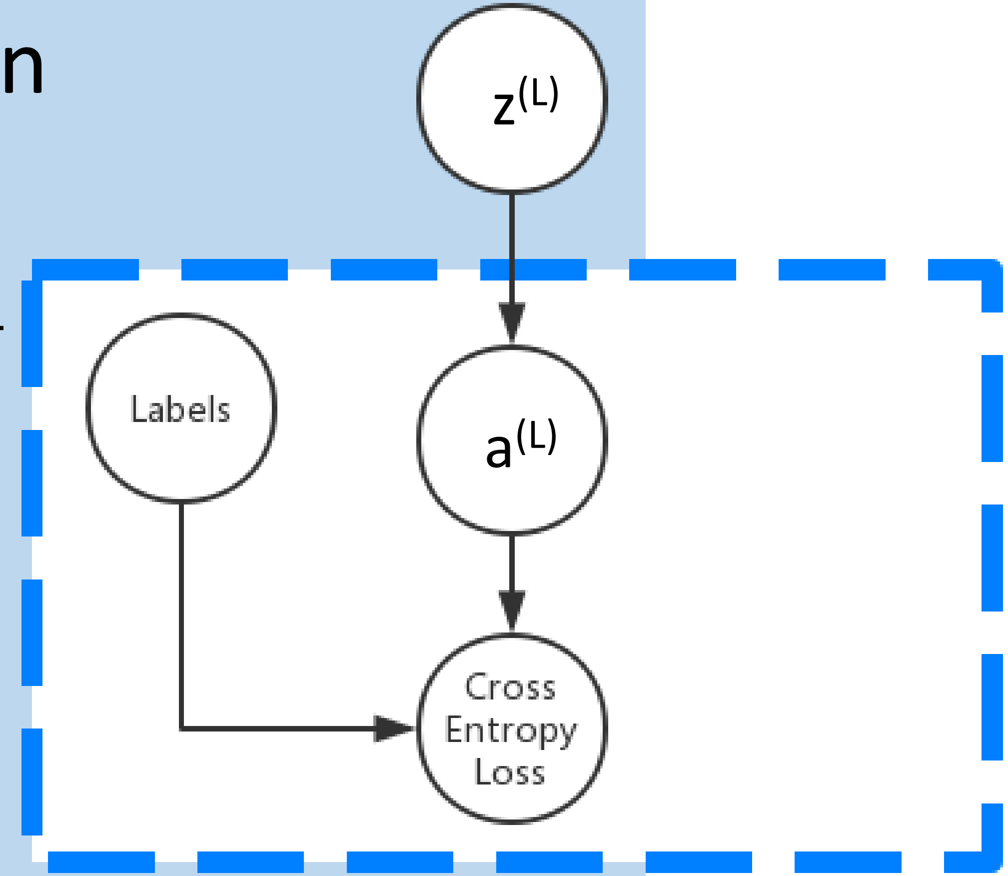
CE loss with softmax activation

$$\text{if } i = j: \frac{\partial a_i^{(L)}}{\partial z_j^{(L)}} = \frac{e^{z_i^{(L)}} \sum_{k=1}^{N} e^{z_k^{(L)}} - e^{z_j^{(L)}} e^{z_i^{(L)}}}{\left(\sum_{k=1}^{N} e^{z_k^{(L)}}\right)^2}$$

$$= \frac{e^{z_i^{(L)}}}{\sum_{k=1}^{N} e^{z_k^{(L)}}} \times \frac{\sum_{k=1}^{N} e^{z_k^{(L)}} - e^{z_j^{(L)}}}{\sum_{k=1}^{N} e^{z_k^{(L)}}}$$

$$= a_i^{(L)} \left(1 - a_j^{(L)}\right)$$

$$= a_i^{(L)} - a_i^{(L)} a_j^{(L)}$$



Back Propagation

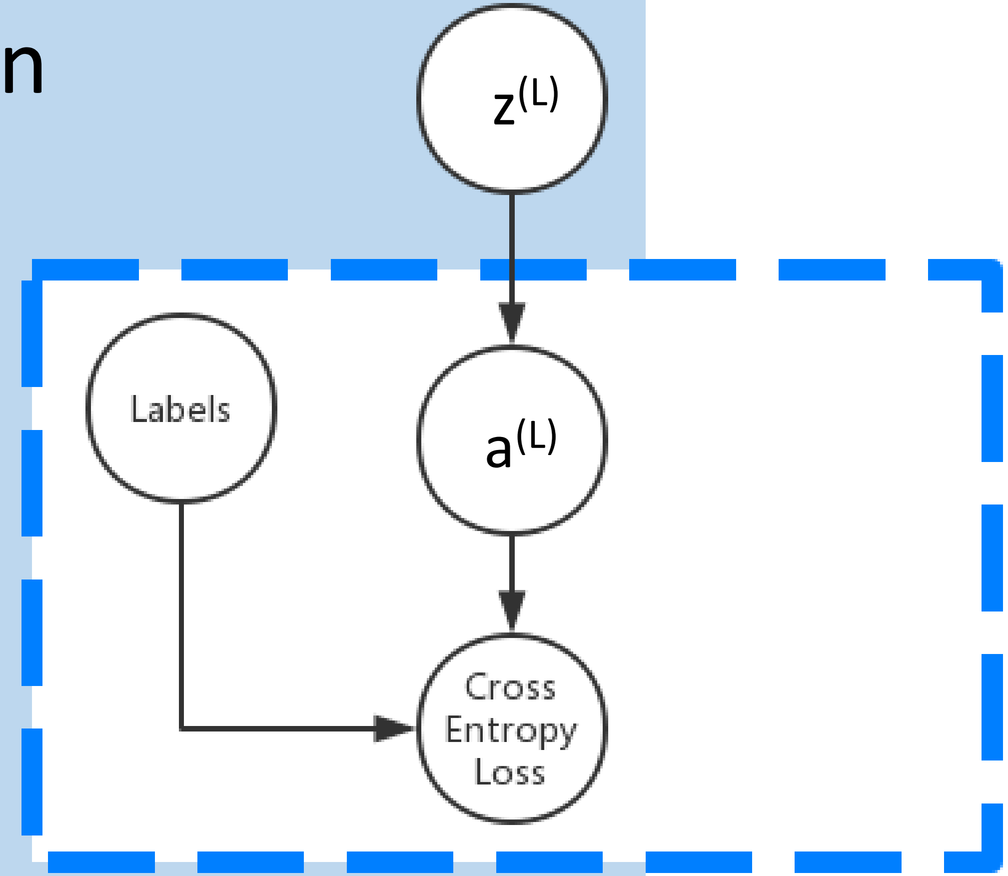
CE loss with softmax activation

$$if\ i \neq j:\ \frac{\partial a_i^{(L)}}{\partial z_j^{(L)}} = \frac{0 - e^{z_j^{(L)}} e^{z_i^{(L)}}}{\left(\sum_{k=1}^{N} e^{z_k^{(L)}}\right)^2}$$

$$= \frac{e^{z_i^{(L)}}}{\sum_{k=1}^{N} e^{z_k^{(L)}}} \times \frac{-e^{z_j^{(L)}}}{\sum_{k=1}^{N} e^{z_k^{(L)}}}$$

$$= -a_i^{(L)} a_j^{(L)}$$

$$\frac{\partial a_i^{(L)}}{\partial z_j^{(L)}} = \delta_{ij} a_i^{(L)} - a_i^{(L)}\ a_j^{(L)}$$



Back Propagation

CE loss with softmax activation

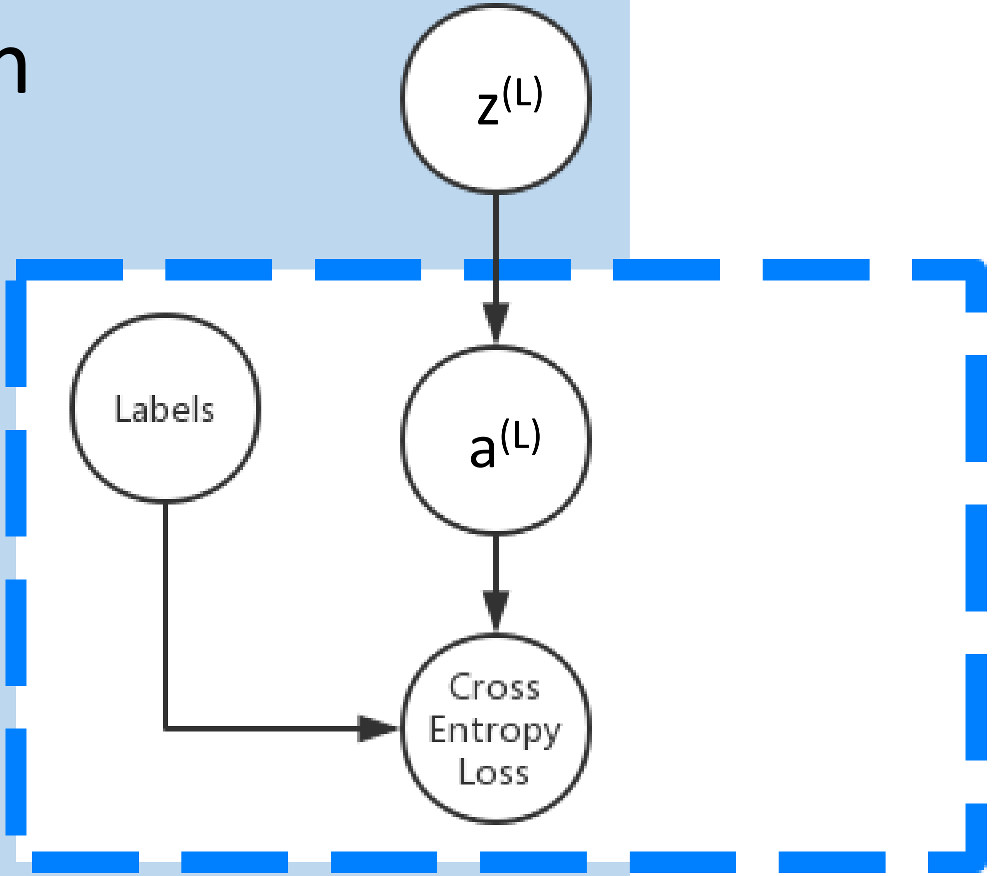
$$\frac{\partial Loss}{\partial z_i^{(L)}} = \sum_{k=1}^{N} \frac{\partial Loss}{\partial a_k^{(L)}} * \frac{\partial a_k^{(L)}}{\partial z_i^{(L)}}$$

$$= -\sum_{k=1}^{N} y_i * \frac{1}{a_k^{(L)}} * (\delta_{ki}\, a_k^{(L)} - a_k^{(L)}\, a_i^{(L)})$$

$$= -y_i + y_i a_i^{(L)} + \sum_{k=1,k\neq i}^{N} y_k a_i^{(L)}$$

$$= -y_i + \sum_{k=1}^{N} y_k a_i^{(L)}$$

$$= -y_i + a_i^{(L)}$$

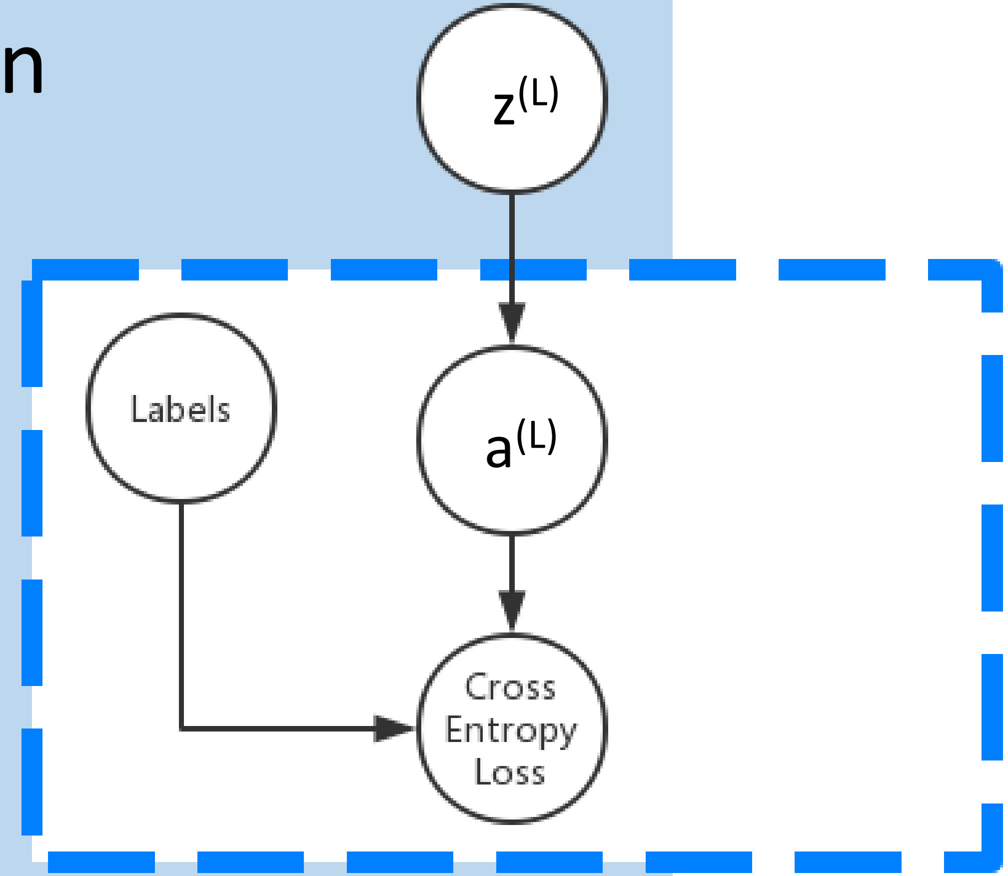


Back Propagation

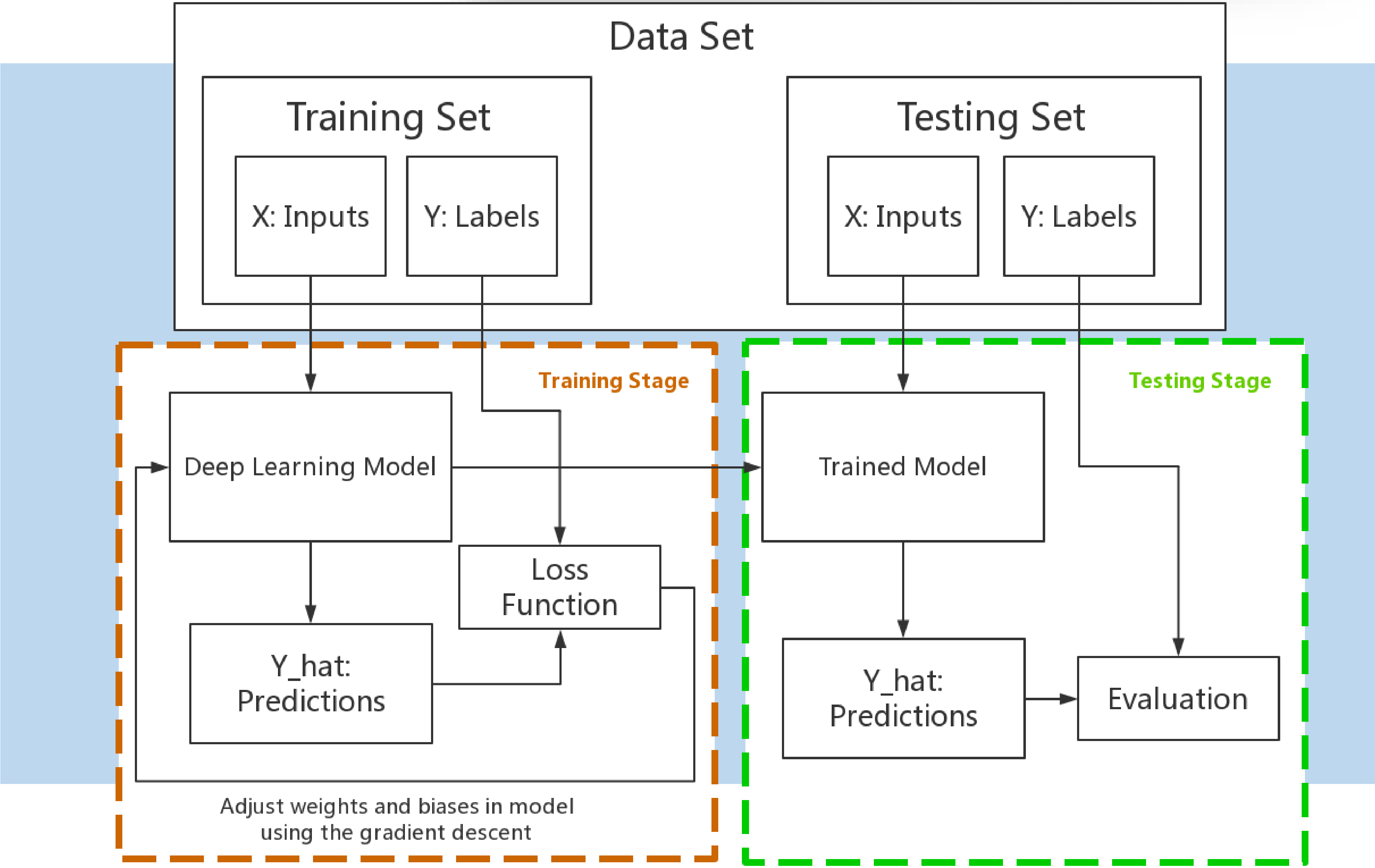
CE loss with softmax activation

See Derivation of Backpropagation in Convolutional Neural Network (CNN) for more details.

http://web.eecs.utk.edu/~zzhang61/docs/reports/2016.10%20-%20Derivation%20of%20Backpropagation%20in%20Convolutional%20Neural%20Network%20(CNN).pdf



Back Propagation



OUTLINES

1 Cross Entropy

2 Back Propagation

3 Paper Reading

4 Discussions

Paper Reading

submitted to *Geophys. J. Int.*

PhaseNet: A Deep-Neural-Network-Based Seismic Arrival Time Picking Method

Weiqiang Zhu* and Gregory C. Beroza

Department of Geophysics, Stanford University

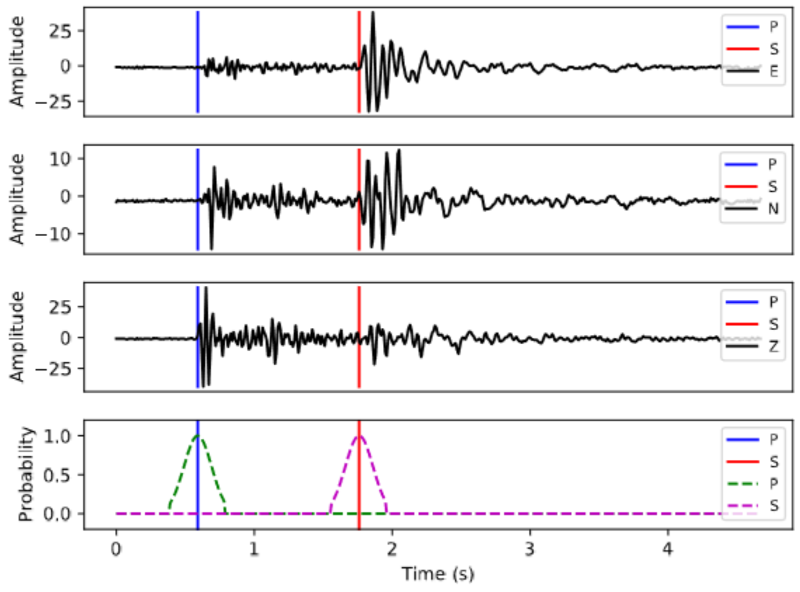
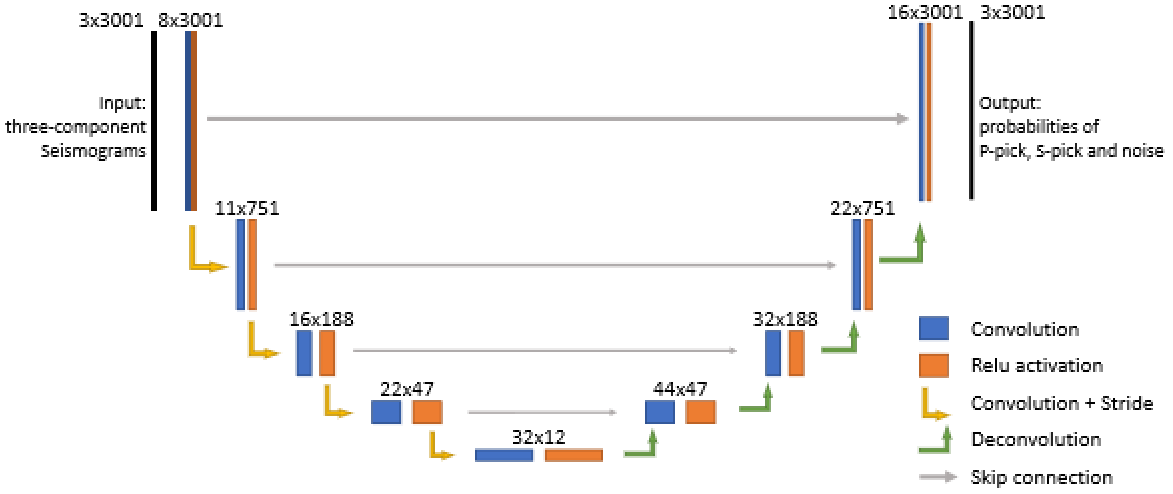


Figure 4. A sample from the dataset. (a) - (c) Seismograms of the "ENZ" (East, North, Vertical) components. The blue and red vertical lines are the manually picked P and S arrival times. (d) The converted probability distribution for P and S pickers. The shape is a truncated Gaussian distribution with mean ($\mu = 0s$) and standard deviation ($\sigma = 0.1s$).

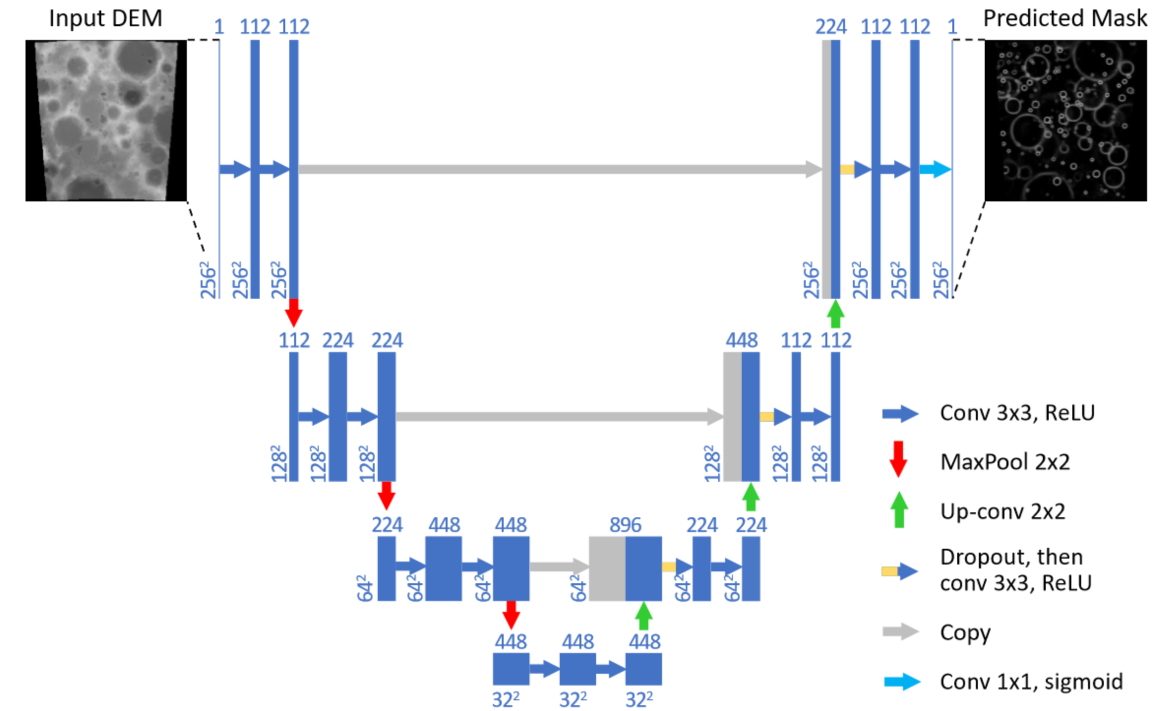


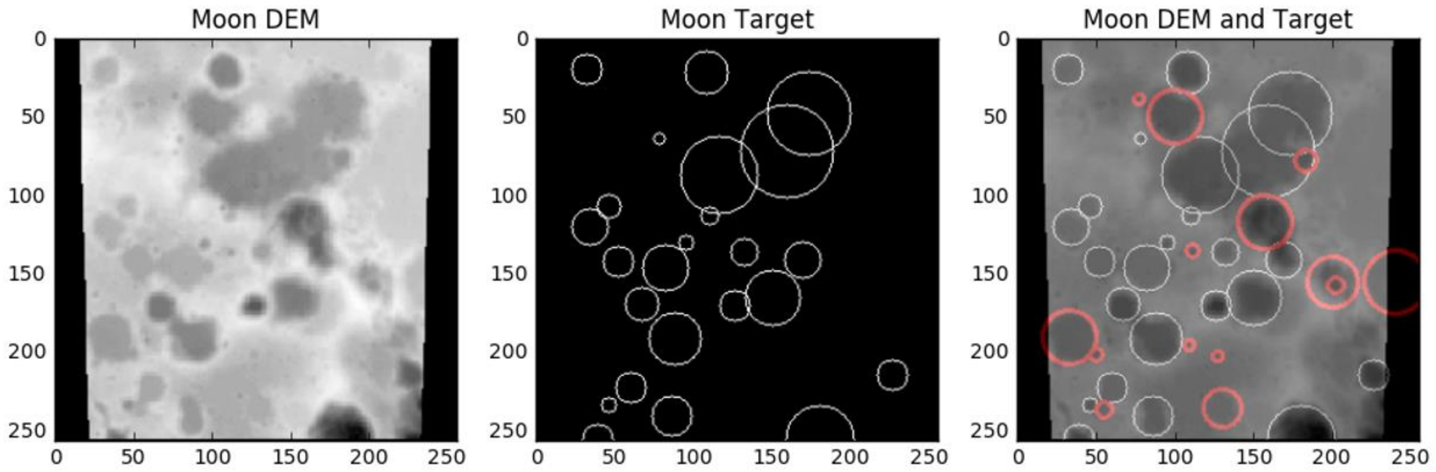
Paper Reading

Lunar Crater Identification via Deep Learning

Ari Silburt[a,b,c,f], Mohamad Ali-Dib[a,d,f], Chenchong Zhu[b,d], Alan Jackson[a,b,e], Diana Valencia[a,b], Yevgeni Kissin[b], Daniel Tamayo[a,d], Kristen Menou[a,b]

[a]*Centre for Planetary Sciences, Department of Physical & Environmental Sciences, University of Toronto Scarborough, Toronto, Ontario M1C 1A4, Canada*
[b]*Department of Astronomy & Astrophysics, University of Toronto, Toronto, Ontario M5S 3H4, Canada*
[c]*Department of Astronomy & Astrophysics, Penn State University, Eberly College of Science, State College, PA 16801, USA*
[d]*Canadian Institute for Theoretical Astrophysics, 60 St. George St, University of Toronto, Toronto, ON M5S 3H8, Canada*
[e]*School of Earth and Space Exploration, Arizona State University, 781 E Terrace Mall, Tempe, AZ 85287-6004, USA*
[f]*These authors contributed equally to this work.*





OUTLINES

1. Cross Entropy
2. Back Propagation
3. Paper Reading
4. Discussions

Discussions

References

Silburt, A., Ali-Dib, M., Zhu, C., Jackson, A., Valencia, D., Kissin, Y., Tamayo, D., Menou, K., 2019. Lunar Crater Identification via Deep Learning. Icarus 317, 27–38. https://doi.org/10.1016/j.icarus.2018.06.022

Zhu, W., Beroza, G.C., 2018. PhaseNet: A Deep-Neural-Network-Based Seismic Arrival Time Picking Method. Geophysical Journal International. https://doi.org/10.1093/gji/ggy423

Mackay, D.J.C., Information Theory, Inference, and Learning Algorithms, Cambridge University Press, 2003.