TU-Delft Deep Learning course 2018-2019

02.MLrefresh

20 Feb 2019



Lecturer: Jan van Gemert Several slides credit to Roger Grosse

Questions?

Last time:

- Feed forward networks
- Stochastic gradient descent
- XOR

After this lecture you can:

- Understand maximum likelihood
- Explain the relation between a loss and gradient descent
- Understand binary classification with logistic regression
- Understand multiclass logistic regression

Book chapters: 3.10, 3.13, 5.5, 5.7.1, 6.2

- Training set of m samples $\mathbb{X} = \{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$
- Drawn i.i.d. from the true but unknown distribution $p_{data}(x)$

- Training set of m samples $\mathbb{X} = \{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$
- Drawn i.i.d. from the true but unknown distribution $p_{data}(x)$
- Our model: Parametric family of probability distributions $p_{model}(x;\theta)$
- Maximum likelihood is $\theta_{ML} = \arg \max_{\theta} p_{\mathsf{model}}(\mathbb{X}; \theta)$

- Training set of m samples $\mathbb{X} = \{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$
- Drawn i.i.d. from the true but unknown distribution $p_{\text{data}}(x)$
- Our model: Parametric family of probability distributions $p_{\mathsf{model}}(x;\theta)$
- Maximum likelihood is $\theta_{ML} = \arg \max_{\theta} p_{\mathsf{model}}(\mathbb{X}; \theta)$
- Rewrite in terms of data samples: $\arg\max_{\theta}\prod_{i=1}^{m}p_{\mathsf{model}}(x^{(i)};\theta)$

- Training set of m samples $\mathbb{X} = \{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$
- Drawn i.i.d. from the true but unknown distribution $p_{\text{data}}(x)$
- Our model: Parametric family of probability distributions $p_{\mathsf{model}}(x;\theta)$
- Maximum likelihood is $\theta_{ML} = \arg \max_{\theta} p_{\mathsf{model}}(\mathbb{X}; \theta)$
- Rewrite in terms of data samples: $\arg \max_{\theta} \prod_{i=1}^{m} p_{\mathsf{model}}(x^{(i)}; \theta)$
- Q: What assumption allowed this?

- Training set of m samples $\mathbb{X} = \{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$
- Drawn i.i.d. from the true but unknown distribution $p_{\text{data}}(x)$
- Our model: Parametric family of probability distributions $p_{\mathsf{model}}(x;\theta)$
- Maximum likelihood is $\theta_{ML} = \arg \max_{\theta} p_{\mathsf{model}}(\mathbb{X}; \theta)$
- Rewrite in terms of data samples: $\arg \max_{\theta} \prod_{i=1}^{m} p_{\mathsf{model}}(x^{(i)}; \theta)$
- Q: What assumption allowed this? A: i.i.d.
- Q: What is the problem with multiplying small values?

- Training set of m samples $\mathbb{X} = \{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$
- Drawn i.i.d. from the true but unknown distribution $p_{\text{data}}(x)$
- Our model: Parametric family of probability distributions $p_{\mathsf{model}}(x;\theta)$
- Maximum likelihood is $\theta_{ML} = \arg \max_{\theta} p_{\mathsf{model}}(\mathbb{X}; \theta)$
- Rewrite in terms of data samples: $\arg\max_{\theta}\prod_{i=1}^{m}p_{\mathsf{model}}(x^{(i)};\theta)$
- Q: What assumption allowed this? A: i.i.d.
- Q: What is the problem with multiplying small values? A: goes to 0.
- Log: same maximum, turns multiplications to sums: $\arg\max_{\theta} \sum_{i=1}^m \log p_{\mathsf{model}}(x^{(i)}; \theta)$

- Training set of m samples $\mathbb{X} = \{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$
- Drawn i.i.d. from the true but unknown distribution $p_{\text{data}}(x)$
- Our model: Parametric family of probability distributions $p_{\mathsf{model}}(x;\theta)$
- Maximum likelihood is $\theta_{ML} = \arg \max_{\theta} p_{\mathsf{model}}(\mathbb{X}; \theta)$
- Rewrite in terms of data samples: $\arg\max_{\theta}\prod_{i=1}^{m}p_{\mathsf{model}}(x^{(i)};\theta)$
- Q: What assumption allowed this? A: i.i.d.
- Q: What is the problem with multiplying small values? A: goes to 0.
- Log: same maximum, turns multiplications to sums: $\arg\max_{\theta} \sum_{i=1}^{m} \log p_{\mathsf{model}}(x^{(i)}; \theta)$
- Write as expectation wrt empirical distribution \hat{p}_{data} as $\arg\max_{\theta} \mathbb{E}_{x \sim \hat{p}_{\text{data}}} \log p_{\text{model}}(x; \theta)$,
- Q: Why is this the same?

- Training set of m samples $\mathbb{X} = \{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$
- Drawn i.i.d. from the true but unknown distribution $p_{\text{data}}(x)$
- Our model: Parametric family of probability distributions $p_{\mathsf{model}}(x;\theta)$
- Maximum likelihood is $\theta_{ML} = \arg \max_{\theta} p_{\mathsf{model}}(\mathbb{X}; \theta)$
- Rewrite in terms of data samples: $\arg\max_{\theta}\prod_{i=1}^{m}p_{\mathsf{model}}(x^{(i)};\theta)$
- Q: What assumption allowed this? A: i.i.d.
- Q: What is the problem with multiplying small values? A: goes to 0.
- Log: same maximum, turns multiplications to sums: $\arg\max_{\theta} \sum_{i=1}^{m} \log p_{\mathsf{model}}(x^{(i)}; \theta)$
- Write as expectation wrt empirical distribution \hat{p}_{data} as $\arg\max_{\theta} \mathbb{E}_{x \sim \hat{p}_{\text{data}}} \log p_{\text{model}}(x; \theta)$,
- ullet Q: Why is this the same? A: Dividing by m does not change the maximum

- Max likelihood: $\arg\max_{\theta} \mathbb{E}_{x \sim \hat{p}_{\mathsf{data}}} \log p_{\mathsf{model}}(x; \theta)$
- ullet One interpretation: Minimize dissimilarity between \hat{p}_{data} and p_{model}

Book: 5.5, 3.13

- Max likelihood: $\arg\max_{\theta} \mathbb{E}_{x \sim \hat{p}_{\mathsf{data}}} \log p_{\mathsf{model}}(x; \theta)$
- One interpretation: Minimize dissimilarity between \hat{p}_{data} and p_{model}
- Measure dissimilarity by the KL divergence

$$D(p_{\text{data}}||p_{\text{model}}) = \sum_{i=1}^{m} \hat{p}_{\text{data}}(x^{(i)}) \log \left(\frac{p_{\text{data}}(x^{(i)})}{p_{\text{model}}(x^{(i)})}\right)$$

• Q: rewrite in terms of data samples (expectation) and using logs?

- Max likelihood: $\arg\max_{\theta} \mathbb{E}_{x \sim \hat{p}_{\mathsf{data}}} \log p_{\mathsf{model}}(x; \theta)$
- One interpretation: Minimize dissimilarity between $\hat{p}_{\sf data}$ and $p_{\sf model}$
- Measure dissimilarity by the KL divergence

$$D(p_{\text{data}}||p_{\text{model}}) = \sum_{i=1}^{m} \hat{p}_{\text{data}}(x^{(i)}) \log \left(\frac{p_{\text{data}}(x^{(i)})}{p_{\text{model}}(x^{(i)})}\right)$$

- Q: rewrite in terms of data samples (expectation) and using logs?
- A: $\mathbb{E}_{x \sim \hat{p}_{\mathsf{data}}} \left[\log p_{\mathsf{data}}(x) \log p_{\mathsf{model}}(x) \right]$

- Max likelihood: $\arg\max_{\theta} \mathbb{E}_{x \sim \hat{p}_{\mathsf{data}}} \log p_{\mathsf{model}}(x; \theta)$
- One interpretation: Minimize dissimilarity between \hat{p}_{data} and p_{model}
- Measure dissimilarity by the KL divergence

$$D(p_{\text{data}}||p_{\text{model}}) = \sum_{i=1}^{m} \hat{p}_{\text{data}}(x^{(i)}) \log \left(\frac{p_{\text{data}}(x^{(i)})}{p_{\text{model}}(x^{(i)})}\right)$$

- Q: rewrite in terms of data samples (expectation) and using logs?
- A: $\mathbb{E}_{x \sim \hat{p}_{\mathsf{data}}} \left[\log p_{\mathsf{data}}(x) \log p_{\mathsf{model}}(x) \right]$
- Left/data term $(\log p_{\text{data}}(x))$ does not depend on the model: Remove.

- Max likelihood: $\arg\max_{\theta} \mathbb{E}_{x \sim \hat{p}_{\mathsf{data}}} \log p_{\mathsf{model}}(x; \theta)$
- One interpretation: Minimize dissimilarity between \hat{p}_{data} and p_{model}
- Measure dissimilarity by the KL divergence

$$D(p_{\text{data}}||p_{\text{model}}) = \sum_{i=1}^{m} \hat{p}_{\text{data}}(x^{(i)}) \log \left(\frac{p_{\text{data}}(x^{(i)})}{p_{\text{model}}(x^{(i)})}\right)$$

- Q: rewrite in terms of data samples (expectation) and using logs?
- A: $\mathbb{E}_{x \sim \hat{p}_{\mathsf{data}}} \left[\log p_{\mathsf{data}}(x) \log p_{\mathsf{model}}(x) \right]$
- Left/data term $(\log p_{\text{data}}(x))$ does not depend on the model: Remove.
- So minimize $\mathbb{E}_{x \sim \hat{p}_{\mathsf{data}}}[-\log p_{\mathsf{model}}(x)]$

- Max likelihood: $\arg\max_{\theta} \mathbb{E}_{x \sim \hat{p}_{\mathsf{data}}} \log p_{\mathsf{model}}(x; \theta)$
- One interpretation: Minimize dissimilarity between \hat{p}_{data} and p_{model}
- Measure dissimilarity by the KL divergence

$$D(p_{\text{data}}||p_{\text{model}}) = \sum_{i=1}^{m} \hat{p}_{\text{data}}(x^{(i)}) \log \left(\frac{p_{\text{data}}(x^{(i)})}{p_{\text{model}}(x^{(i)})}\right)$$

- Q: rewrite in terms of data samples (expectation) and using logs?
- A: $\mathbb{E}_{x \sim \hat{p}_{\mathsf{data}}} \left[\log p_{\mathsf{data}}(x) \log p_{\mathsf{model}}(x) \right]$
- Left/data term $(\log p_{\text{data}}(x))$ does not depend on the model: Remove.
- So minimize $\mathbb{E}_{x \sim \hat{p}_{\mathsf{data}}}[-\log p_{\mathsf{model}}(x)]$
- Q: Relate $\arg\min_{\theta} \mathbb{E}_{x \sim \hat{p}_{data}}[\log p_{model}(x)]$ to max likelihood?

- Max likelihood: $\arg\max_{\theta} \mathbb{E}_{x \sim \hat{p}_{\mathsf{data}}} \log p_{\mathsf{model}}(x; \theta)$
- One interpretation: Minimize dissimilarity between \hat{p}_{data} and p_{model}
- Measure dissimilarity by the KL divergence

$$D(p_{\text{data}}||p_{\text{model}}) = \sum_{i=1}^{m} \hat{p}_{\text{data}}(x^{(i)}) \log \left(\frac{p_{\text{data}}(x^{(i)})}{p_{\text{model}}(x^{(i)})}\right)$$

- Q: rewrite in terms of data samples (expectation) and using logs?
- A: $\mathbb{E}_{x \sim \hat{p}_{\mathsf{data}}} \left[\log p_{\mathsf{data}}(x) \log p_{\mathsf{model}}(x) \right]$
- Left/data term $(\log p_{\text{data}}(x))$ does not depend on the model: Remove.
- So minimize $\mathbb{E}_{x \sim \hat{p}_{\mathsf{data}}}[-\log p_{\mathsf{model}}(x)]$
- Q: Relate $\arg\min_{\theta} \mathbb{E}_{x \sim \hat{p}_{\mathsf{data}}}[\log p_{\mathsf{model}}(x)]$ to max likelihood? A: Same.

Book: 3.13

The literature optimizes "cross-entropy".

• KL related to cross-entropy between two distributions $H(p_{\mathsf{data}}, p_{\mathsf{model}}) = H(p_{\mathsf{data}}) + D(p_{\mathsf{data}} || p_{\mathsf{model}})$, where H is entropy.

Book: 3.13

- KL related to cross-entropy between two distributions $H(p_{\text{data}}, p_{\text{model}}) = H(p_{\text{data}}) + D(p_{\text{data}}||p_{\text{model}})$, where H is entropy.
- Q: What is the effect of $H(p_{\text{data}})$ on p_{model} ?

Book: 3.13

- KL related to cross-entropy between two distributions $H(p_{\text{data}}, p_{\text{model}}) = H(p_{\text{data}}) + D(p_{\text{data}}||p_{\text{model}})$, where H is entropy.
- Q: What is the effect of $H(p_{data})$ on p_{model} ?
- A: The model does not depend on $H(p_{data})$, so can be omitted.

Book: 3.13

- KL related to cross-entropy between two distributions $H(p_{\text{data}}, p_{\text{model}}) = H(p_{\text{data}}) + D(p_{\text{data}}||p_{\text{model}})$, where H is entropy.
- Q: What is the effect of $H(p_{data})$ on p_{model} ?
- A: The model does not depend on $H(p_{data})$, so can be omitted.
- The term cross-entropy is generic, not just for classification (Bernoulli or softmax distributions).
- Eg: mean squared error is the cross-entropy between empirical distribution and a Gaussian.

Book: 3.13

- KL related to cross-entropy between two distributions $H(p_{\text{data}}, p_{\text{model}}) = H(p_{\text{data}}) + D(p_{\text{data}}||p_{\text{model}})$, where H is entropy.
- Q: What is the effect of $H(p_{data})$ on p_{model} ?
- A: The model does not depend on $H(p_{data})$, so can be omitted.
- The term cross-entropy is generic, not just for classification (Bernoulli or softmax distributions).
- Eg: mean squared error is the cross-entropy between empirical distribution and a Gaussian.
- Minimize KL divergence
 ⇔ minimize negative log likelihood
 ⇔ minimize cross-entropy
 ⇔ maximize maximum likelihood.

Conditional log-likelihood and output units

Book: 5.5.1 and 6.2.2

Often interested in classification problems

- Conditional log-likelihood: Predict labels Y given data X: $P(Y|X;\theta)$
- $\theta_{ML} = \arg \max_{\theta} P(Y|X;\theta)$
- Assume i.i.d.: $\arg\max_{\theta} \sum_{i=1}^{m} \log P(y^{(i)}|x^{(i)};\theta)$

Conditional log-likelihood and output units

Book: 5.5.1 and 6.2.2

Often interested in classification problems

- Conditional log-likelihood: Predict labels Y given data X: $P(Y|X;\theta)$
- $\theta_{ML} = \arg \max_{\theta} P(Y|X;\theta)$
- Assume i.i.d.: $\arg\max_{\theta} \sum_{i=1}^{m} \log P(y^{(i)}|x^{(i)};\theta)$

What to optimize in a deep net:

- Usually: cross-entropy between data distribution and model distribution
- Output units determine the form of the cross-entropy function

Book: 3.10 and 6.2.2

Bernoulli distribution on P(Y=1|X), single number between [0,1]

Book: 3.10 and 6.2.2

Bernoulli distribution on P(Y = 1|X), single number between [0,1]

• Q: Why is a single number enough for two classes?

Book: 3.10 and 6.2.2

Bernoulli distribution on P(Y = 1|X), single number between [0,1]

- Q: Why is a single number enough for two classes?
- A: P(Y = 0|X) = 1 P(Y = 1|X)

Book: 3.10 and 6.2.2

Bernoulli distribution on P(Y = 1|X), single number between [0,1]

- Q: Why is a single number enough for two classes?
- A: P(Y = 0|X) = 1 P(Y = 1|X)

Lets try linear unit $P(Y=1|X) = \max\left\{0, \min\left\{1, w^\top h + b\right\}\right\} = y$ for features h

Book: 3.10 and 6.2.2

Bernoulli distribution on P(Y = 1|X), single number between [0,1]

- Q: Why is a single number enough for two classes?
- A: P(Y = 0|X) = 1 P(Y = 1|X)

Lets try linear unit $P(Y=1|X) = \max\left\{0, \min\left\{1, w^{\top}h + b\right\}\right\} = y$ for features h

ullet Q: No loss defined, but what happens to the gradients outside interval [0,1]?

Book: 3.10 and 6.2.2

Bernoulli distribution on P(Y = 1|X), single number between [0,1]

- Q: Why is a single number enough for two classes?
- A: P(Y = 0|X) = 1 P(Y = 1|X)

Lets try linear unit $P(Y=1|X) = \max\left\{0, \min\left\{1, w^{\top}h + b\right\}\right\} = y$ for features h

- ullet Q: No loss defined, but what happens to the gradients outside interval [0,1]?
- A: No more gradients: Cannot be optimized by gradient descent

Book: 6.2.2.2

The min/max prevent gradient flow $P(Y=1|X) = \max\left\{0, \min\left\{1, w^{\top}h + b\right\}\right\}$

• Q: How could you make them flow again?

Book: 6.2.2.2

The min/max prevent gradient flow $P(Y=1|X) = \max\left\{0, \min\left\{1, w^\top h + b\right\}\right\}$

- Q: How could you make them flow again?
- A: One answer: remove min/max

Book: 6.2.2.2

The min/max prevent gradient flow $P(Y=1|X) = \max\left\{0, \min\left\{1, w^{\top}h + b\right\}\right\}$

- Q: How could you make them flow again?
- A: One answer: remove min/max
- Q: Continue output; how to binarize to two classes?

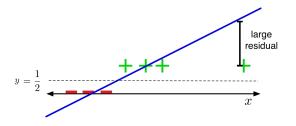
Book: 6.2.2.2

The min/max prevent gradient flow $P(Y=1|X) = \max\left\{0, \min\left\{1, w^{\top}h + b\right\}\right\}$

- Q: How could you make them flow again?
- A: One answer: remove min/max
- Q: Continue output; how to binarize to two classes?
- A: Threshold predictions y at $y = \frac{1}{2}$

loss wants to be exact

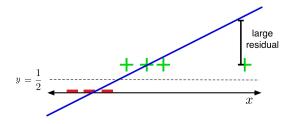
Lets use a square loss, $P(Y=1|X)=\frac{1}{2}((w^{\top}h+b)-t)^2$, t=true label



• Q: What is the problem here?

loss wants to be exact

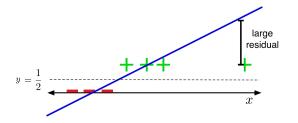
Lets use a square loss, $P(Y=1|X)=\frac{1}{2}((w^{\top}h+b)-t)^2$, t=true label



- Q: What is the problem here?
- A: Large losses for predictions with high confidence.

loss wants to be exact

Lets use a square loss, $P(Y=1|X)=\frac{1}{2}((w^{\top}h+b)-t)^2$, t=true label



- Q: What is the problem here?
- A: Large losses for predictions with high confidence.
- If t = 1, loss is higher for y = 10 then for y = 0.

Book: 5.7.1, 6.2.2.2

 \bullet Q: Other way to limit the output to [0,1] ?

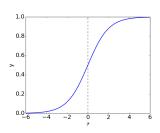
Book: 5.7.1, 6.2.2.2

- Q: Other way to limit the output to [0,1] ?
- A: Squash it between 0 and 1.

Book: 5.7.1, 6.2.2.2

- Q: Other way to limit the output to [0,1] ?
- A: Squash it between 0 and 1.
- The logistic function is sigmoidal (S-shaped)

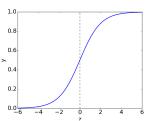
$$\sigma(z) = \frac{1}{1 + e^{-z}}$$



Book: 5.7.1, 6.2.2.2

- Q: Other way to limit the output to [0,1] ?
- A: Squash it between 0 and 1.
- The logistic function is sigmoidal (S-shaped)

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



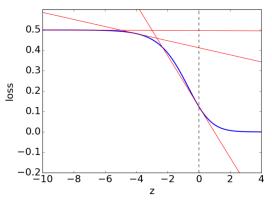
A linear model with a logistic nonlinearity is known as log-linear:

$$z=w^{ op}x+b$$

$$y=\sigma(z)$$
 $\mathcal{L}_{\mathsf{SE}}=rac{1}{2}(y-t)^2$ (SE: Squared Error)

Where σ is the activation function and z is called the logit.

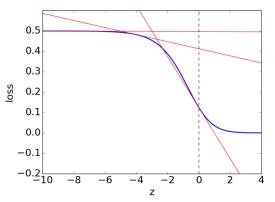
Lets look at the derivatives



Derivatives of squared error loss with logistic nonlinearity for a positive sample.

• Q: What is the problem here?

Lets look at the derivatives



Derivatives of squared error loss with logistic nonlinearity for a positive sample.

- Q: What is the problem here?
- A: Gradient descent: small gradient is a small step.
- Should take large step when the prediction is really wrong

Replace squared loss with Bernoulli cross-entropy loss

• Bernoulli: y if t = 1 and 1 - y if t = 0, t =true label

Q: What is the Bernoulli cross entropy loss?

Replace squared loss with Bernoulli cross-entropy loss

• Bernoulli: y if t = 1 and 1 - y if t = 0, t=true label

Q: What is the Bernoulli cross entropy loss?

$$\mathcal{L}_{CE}(y,t) = \begin{cases} -\log y & \text{if } t = 1\\ -\log(1-y) & \text{if } t = 0 \end{cases}$$
Same as: $-t\log y - (1-t)\log(1-y)$

Replace squared loss with Bernoulli cross-entropy loss

• Bernoulli: y if t = 1 and 1 - y if t = 0, t=true label

Q: What is the Bernoulli cross entropy loss?

$$\mathcal{L}_{\mathsf{CE}}(y,t) = \begin{cases} -\log y & \text{if } t = 1\\ -\log(1-y) & \text{if } t = 0 \end{cases}$$

Same as: $-t\log y - (1-t)\log(1-y)$

Q: Why is this the same?

Replace squared loss with Bernoulli cross-entropy loss

• Bernoulli: y if t = 1 and 1 - y if t = 0, t=true label

Q: What is the Bernoulli cross entropy loss?

$$\mathcal{L}_{\mathsf{CE}}(y,t) = \begin{cases} -\log y & \text{if } t = 1\\ -\log(1-y) & \text{if } t = 0 \end{cases}$$

Same as: $-t\log y - (1-t)\log(1-y)$

Q: Why is this the same?

Q: Draw graphs for t = 1 and t = 0 ?

Replace squared loss with Bernoulli cross-entropy loss

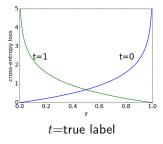
• Bernoulli: y if t = 1 and 1 - y if t = 0, t=true label

Q: What is the Bernoulli cross entropy loss?

$$\mathcal{L}_{\mathsf{CE}}(y,t) = \left\{ \begin{array}{ll} -\log y & \text{if } t = 1 \\ -\log(1-y) & \text{if } t = 0 \end{array} \right.$$
 Same as: $-t\log y - (1-t)\log(1-y)$

Q: Why is this the same?

Q: Draw graphs for t=1 and t=0 ?



Q: What is penalized?

Replace squared loss with Bernoulli cross-entropy loss

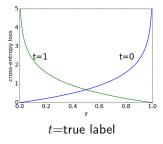
• Bernoulli: y if t = 1 and 1 - y if t = 0, t=true label

Q: What is the Bernoulli cross entropy loss?

$$\begin{split} \mathcal{L}_{\text{CE}}(y,t) &= \left\{ \begin{array}{ll} -\log y & \text{if } t=1 \\ -\log(1-y) & \text{if } t=0 \end{array} \right. \\ \text{Same as: } &-t\log y - (1-t)\log(1-y) \end{split}$$

Q: Why is this the same?

Q: Draw graphs for t=1 and t=0 ?



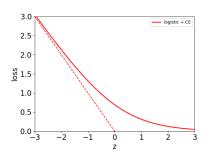
Q: What is penalized?

A: Penalizes small difference of really wrong predictions

$$z = w^{\top}x + b$$

$$y = \sigma(z) = \frac{1}{1 + e^{-z}}$$

$$\mathcal{L}_{\mathsf{CE}} = -t\log y - (1 - t)\log(1 - y)$$

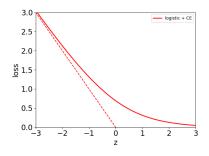


Q: Explain the terms on the left

$$z = w^{\top}x + b$$

$$y = \sigma(z) = \frac{1}{1 + e^{-z}}$$

$$\mathcal{L}_{\mathsf{CE}} = -t \log y - (1 - t) \log(1 - y)$$

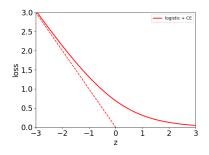


- Q: Explain the terms on the left
- Q: What is shown on this graph?

$$z = w^{\top}x + b$$

$$y = \sigma(z) = \frac{1}{1 + e^{-z}}$$

$$\mathcal{L}_{\mathsf{CE}} = -t \log y - (1 - t) \log(1 - y)$$



Q: Explain the terms on the left

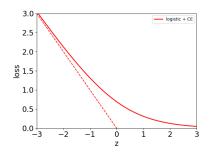
Q: What is shown on this graph?

A: The loss for t = 1

$$z = w^{\top}x + b$$

$$y = \sigma(z) = \frac{1}{1 + e^{-z}}$$

$$\mathcal{L}_{\mathsf{CE}} = -t \log y - (1 - t) \log(1 - y)$$



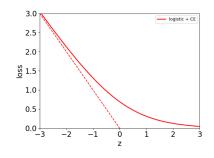
Q: Explain the terms on the left

Q: What is shown on this graph?

A: The loss for t=1

Q: What does it penalize?

$$\begin{split} z &= w^\top x + b \\ y &= \sigma(z) = \frac{1}{1 + e^{-z}} \\ \mathcal{L}_{\mathsf{CE}} &= -t \log y - (1 - t) \log (1 - y) \end{split}$$



Q: Explain the terms on the left

Q: What is shown on this graph?

A: The loss for t=1

Q: What does it penalize?

A: Penalizes very wrong predictions

Multiclass classification

Book: 6.2.2.3

- Targets from a discrete set $\{1,\ldots,K\}$
- Convenient: "One-of-K", or "One-hot" encoding:

$$t = \underbrace{(0,\ldots,0,1,0,\ldots,0)}_{\text{Entry }k \text{ is set to }1}$$

Multiclass classification

Book: 6.2.2.3

- Targets from a discrete set $\{1,\ldots,K\}$
- Convenient: "One-of-K", or "One-hot" encoding:

$$t = \underbrace{(0, \dots, 0, 1, 0, \dots, 0)}_{\text{Entry } k \text{ is set to } 1}$$

• Softmax: generalization of logistic function:

$$Y_k = \operatorname{softmax}(z_1, z_2, \dots, z_K) = \frac{e^{z_k}}{\sum_{k'} e^{z_{k'}}}$$

Multiclass classification

Book: 6.2.2.3

- Targets from a discrete set $\{1, \ldots, K\}$
- Convenient: "One-of-K", or "One-hot" encoding:

$$t = \underbrace{(0,\ldots,0,1,0,\ldots,0)}_{\text{Entry }k \text{ is set to }1}$$

• Softmax: generalization of logistic function:

$$Y_k = \operatorname{softmax}(z_1, z_2, \dots, z_K) = \frac{e^{z_k}}{\sum_{k'} e^{z_{k'}}}$$

Vector of class probabilities, use cross-entropy as loss function

$$\mathcal{L}_{\mathsf{CE}}(y,t) = -\sum_{k=1}^{K} t_k \log y_k = -t^{\top}(\log y)$$

- Outputs positive and sum to 1. (Probabilistic interpretation)
- If one of the z_k is much larger, it approximates the arg max.

Questions?