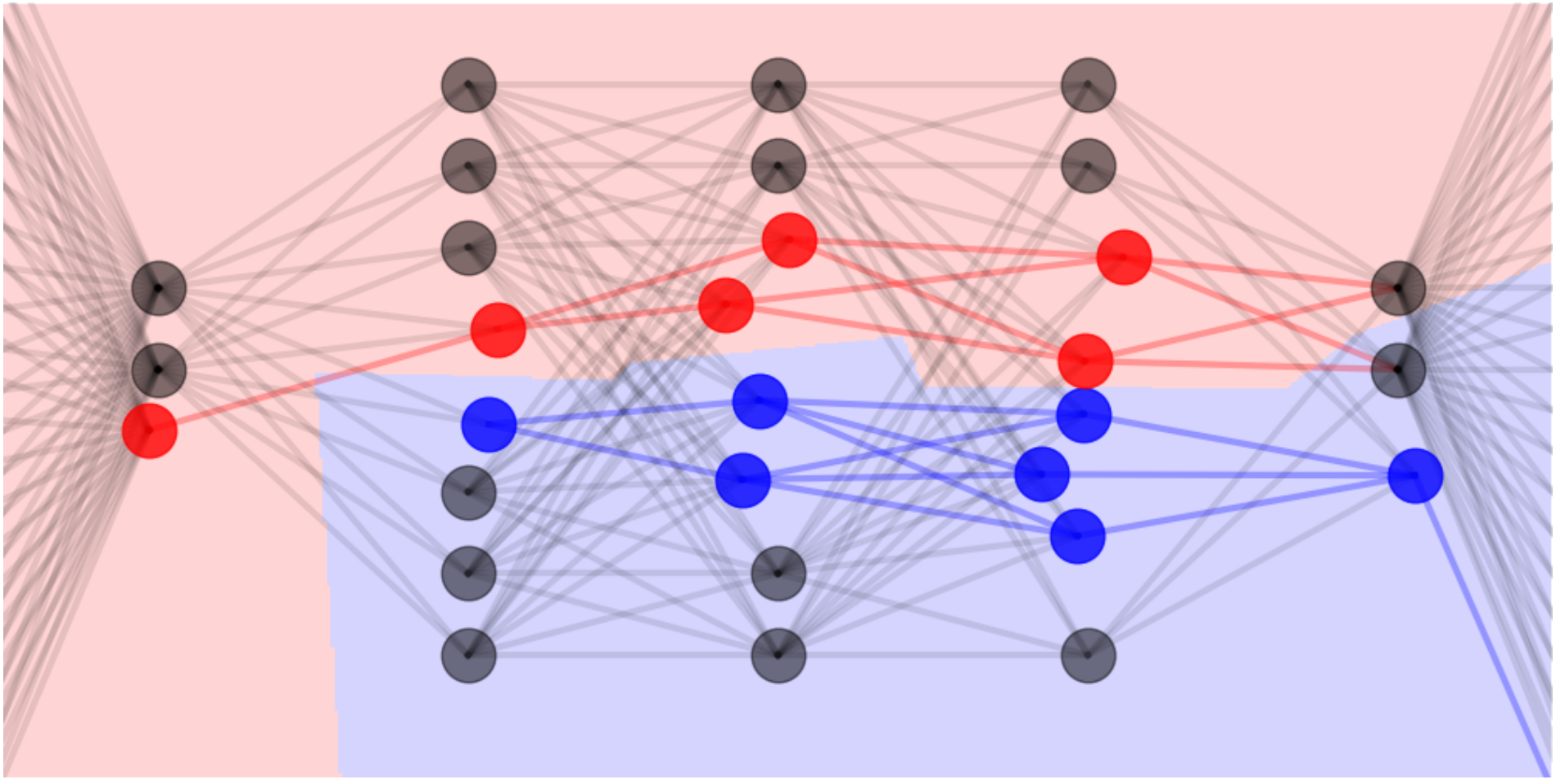
TU-Delft Machine and Deep Learning



Backprop

Lecture 3

Lecturer: Jan van Gemert
Several slides credit to Roger Grosse

Learning goals

After this lecture you can:

- Understand the goal of backpropagation
- Explain the difference between backpropagation and gradient descent
- Explain the forward/backward pass
- Apply the calculus chain rule to compute derivatives
- Apply the backpropagation algorithm to efficiently compute derivatives

DL-book: 4.3, 6.5; UDL-book: 7
See also the extra material by Roger Grosse (on Brightspace)

Training a network

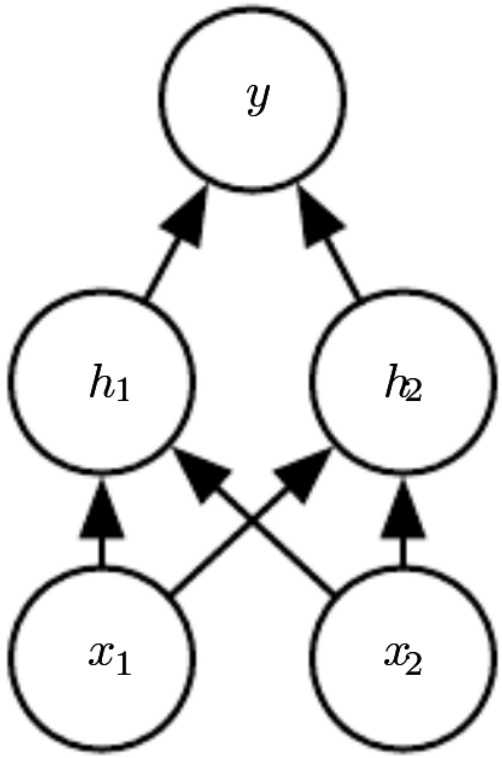
DL-book: 4.3

Training set:

Data		Label
0	1	1
1	1	0
0	0	0
1	0	1

While not converged:

1. Present a training sample
2. Compare the results
3. Update the weights



Training a network

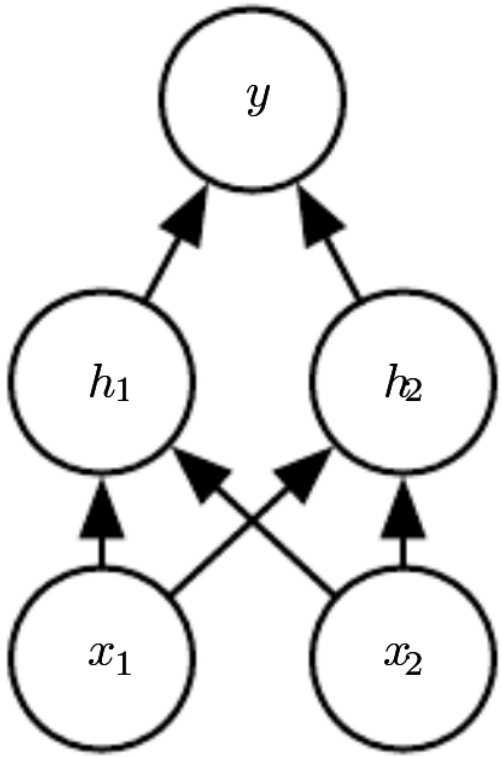
DL-book: 4.3

Training set:

Data		Label
0	1	1
1	1	0
0	0	0
1	0	1

While not converged:

1. Present a training sample $\rightarrow$ Get values
2. Compare the results
3. Update the weights



Training a network

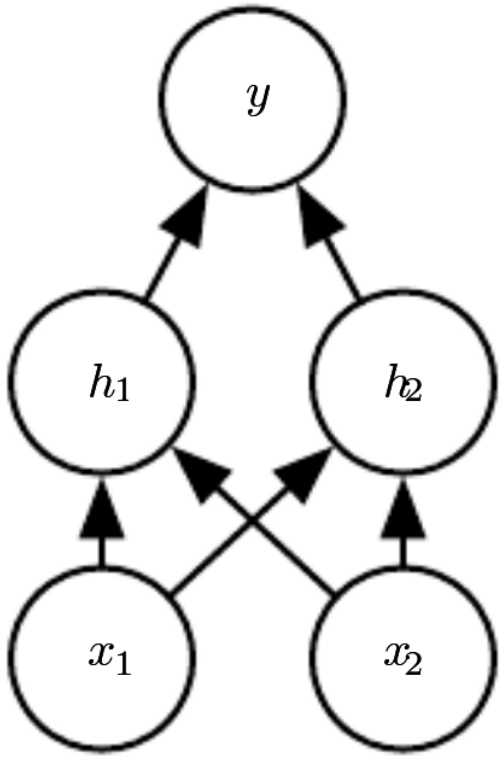
DL-book: 4.3

Training set:

Data		Label
0	1	1
1	1	0
0	0	0
1	0	1

While not converged:

1. Present a training sample $\rightarrow$ Get values
2. Compare the results $\rightarrow$ Loss
3. Update the weights



Training a network

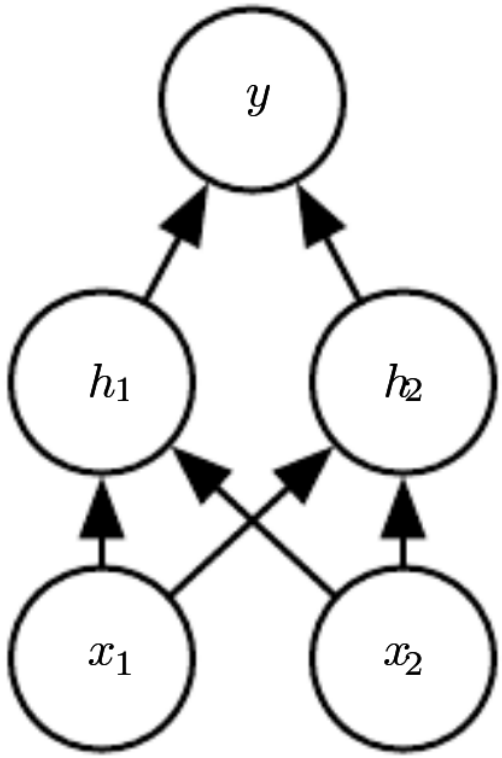
DL-book: 4.3

Training set:

Data		Label
0	1	1
1	1	0
0	0	0
1	0	1

While not converged:

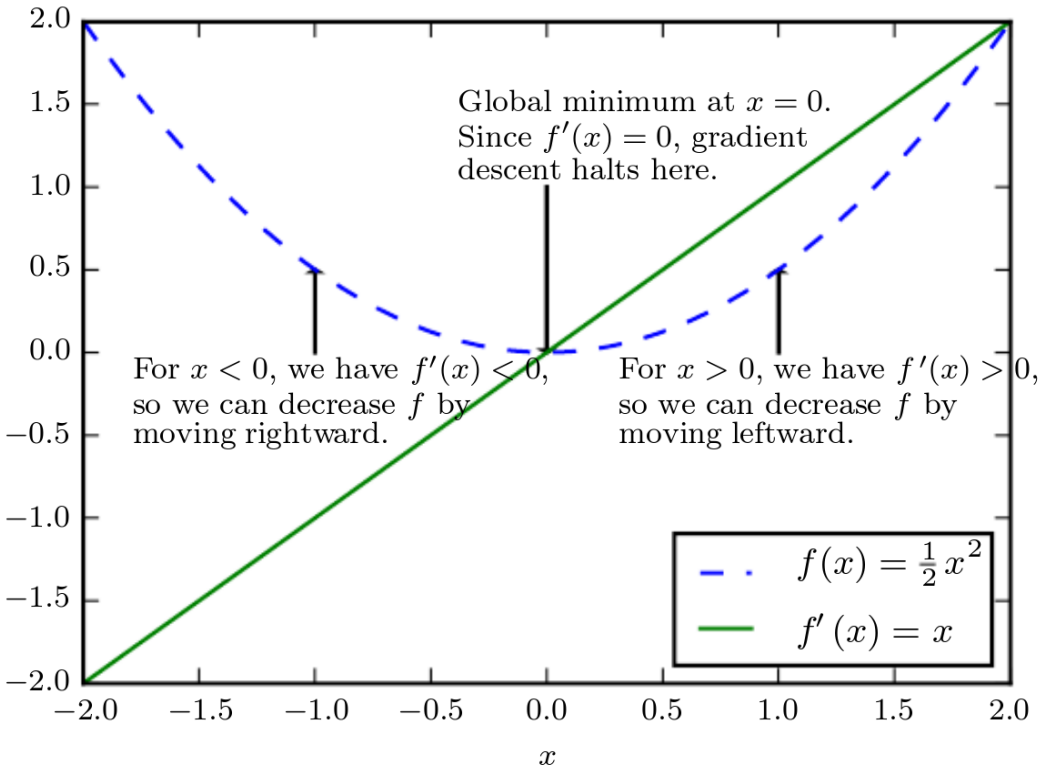
1. Present a training sample $\rightarrow$ Get values
2. Compare the results $\rightarrow$ Loss
3. Update the weights $\rightarrow$ Backprop



Backward pass: Backpropagation

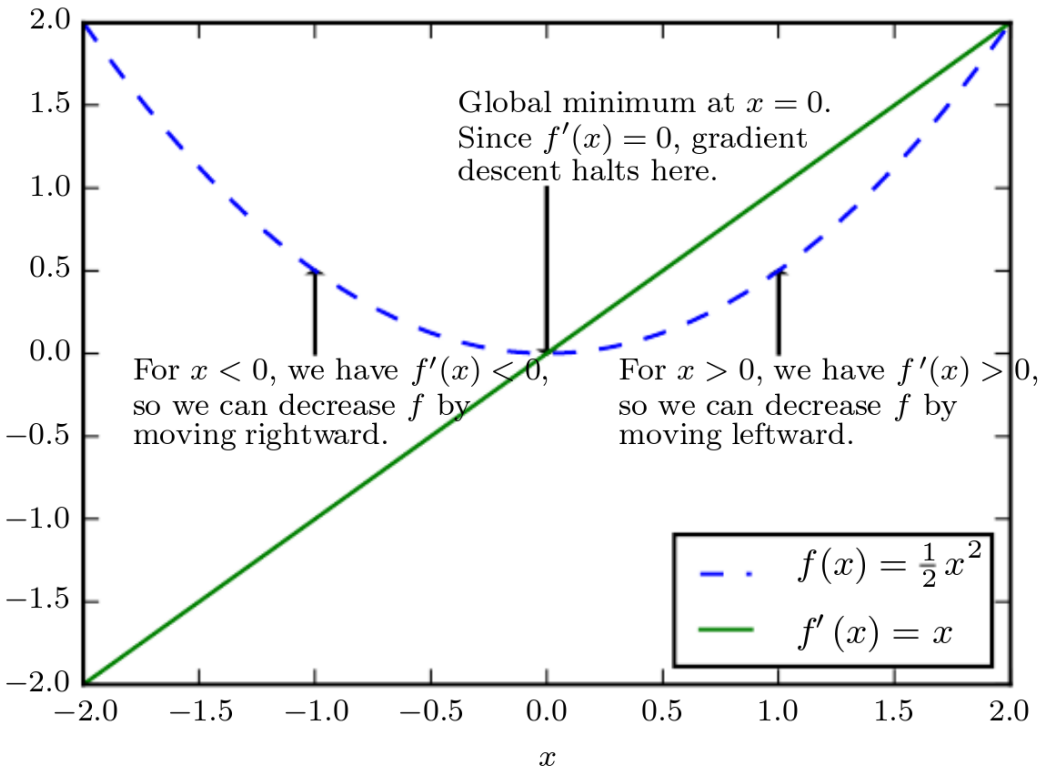
- Backpropagation: Efficient algorithm to compute gradients
- Used to train a huge majority of deep nets
- Engine that powers "End-to-End" optimization and representation learning
- Its "just" a clever application of the chain rule of calculus

Updating the weights: Gradient descent



1-d gradient descent

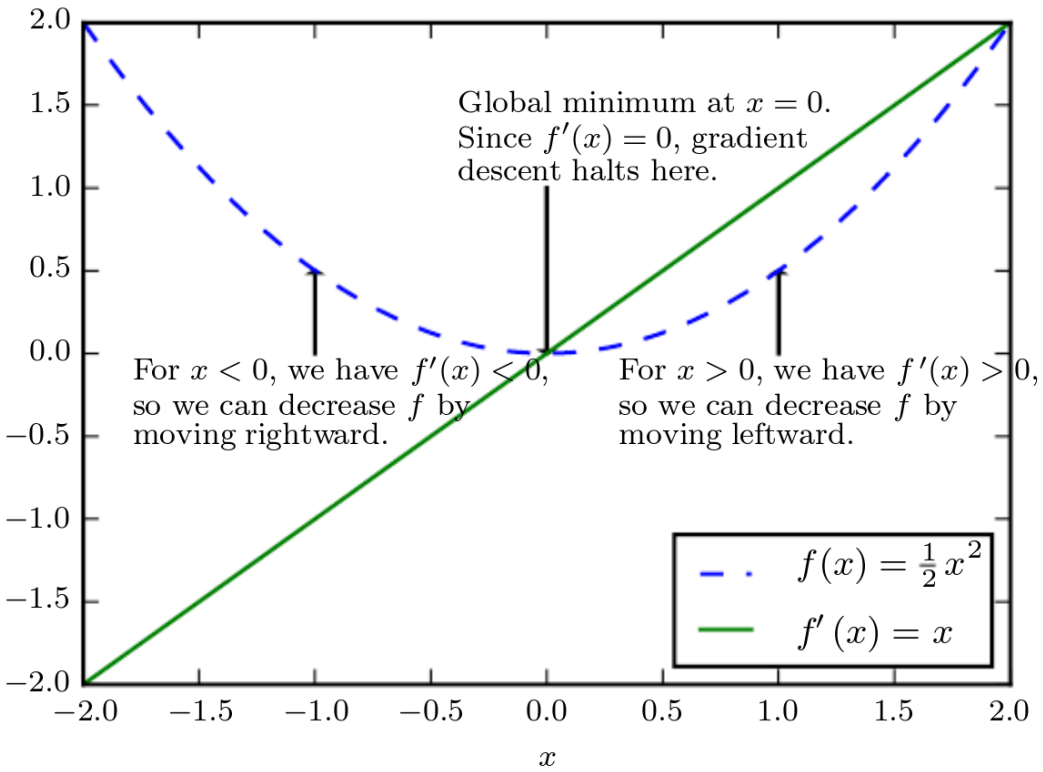
Updating the weights: Gradient descent



1-d gradient descent

- Q: How does "1-d" change for a multi-layer network?

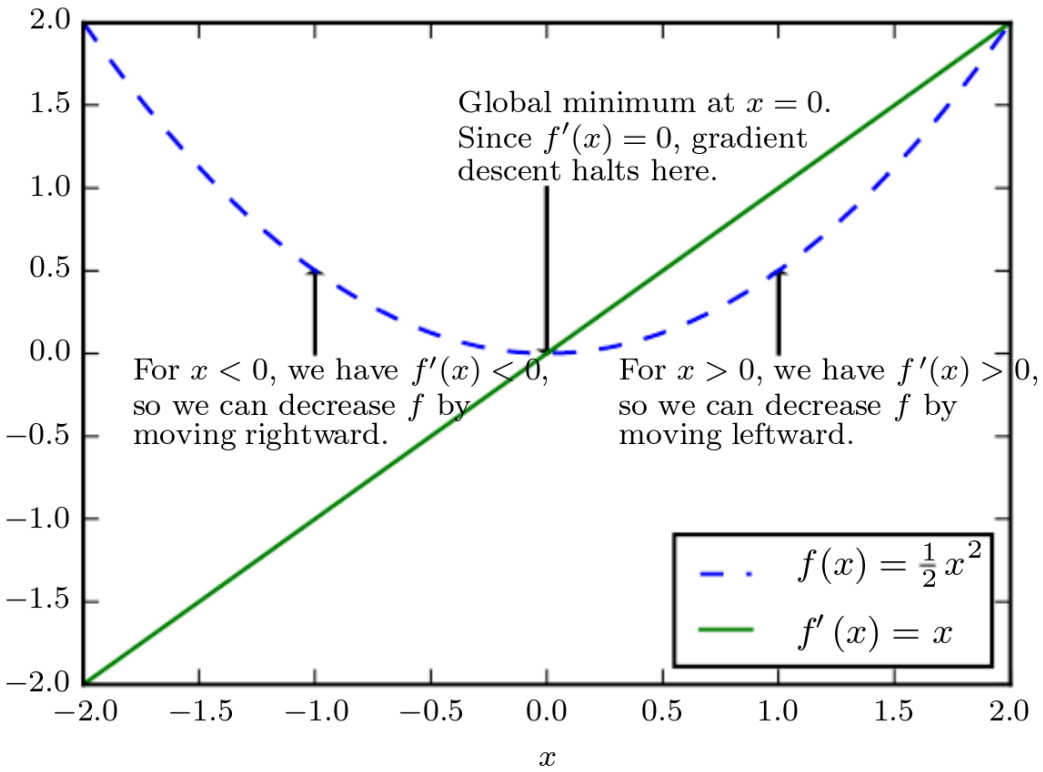
Updating the weights: Gradient descent



1-d gradient descent

- Q: How does "1-d" change for a multi-layer network?
  A: One coordinate for each weight/bias in *all* the layers

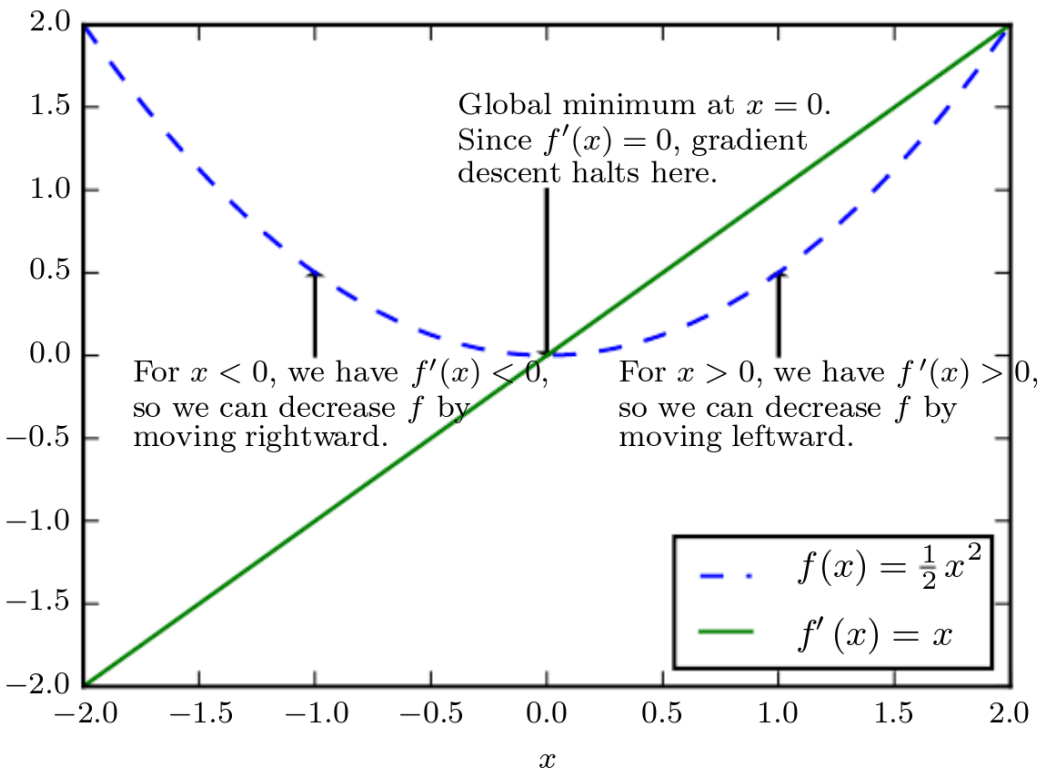
Updating the weights: Gradient descent



1-d gradient descent

- Q: How does "1-d" change for a multi-layer network?
  A: One coordinate for each weight/bias in *all* the layers
- Compute cost gradient: vector of partial derivatives

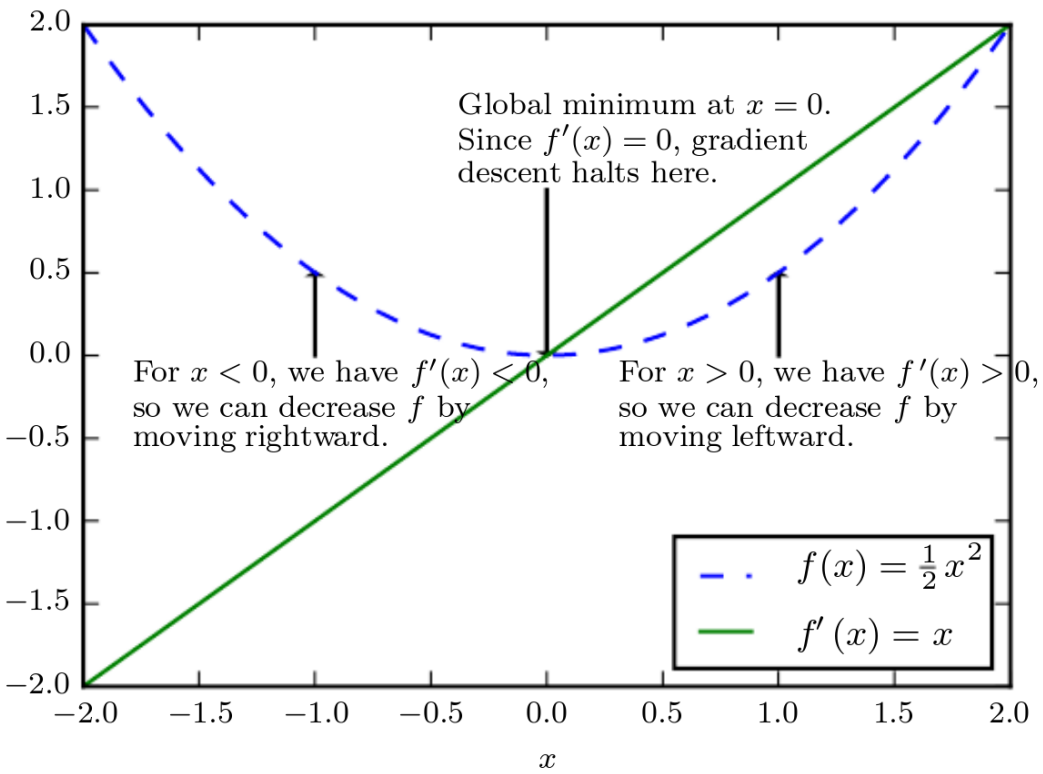
Updating the weights: Gradient descent



1-d gradient descent

- Q: How does "1-d" change for a multi-layer network?
  A: One coordinate for each weight/bias in *all* the layers
- Compute cost gradient: vector of partial derivatives
- Q: Gradient to what again?

Updating the weights: Gradient descent



1-d gradient descent

- Q: How does "1-d" change for a multi-layer network?
  A: One coordinate for each weight/bias in *all* the layers
- Compute cost gradient: vector of partial derivatives
- Q: Gradient to what again?
- A: Trainable parameters $\mathbf{w}$; by computing $\frac{d\mathcal{L}}{d\mathbf{w}}$ for loss $\mathcal{L}$ over training samples.

Chain rule to compute derivatives

Eqs (6), (7) in MDL-BackProp-RGrosse.pdf

- Deep net is a function chain $f(x) = f^{(2)}(f^{(1)}(x))$

  Q: How to compute $f(g(x))'$? (Lagrange notation ' for derivatives)

Chain rule to compute derivatives

Eqs (6), (7) in MDL-BackProp-RGrosse.pdf

- Deep net is a function chain $f(x) = f^{(2)}(f^{(1)}(x))$

  Q: How to compute $f(g(x))'$? (Lagrange notation ' for derivatives)

  A: $f(g(x))' = f'(g(x))g'(x)$

Chain rule to compute derivatives

Eqs (6), (7) in MDL-BackProp-RGrosse.pdf

- Deep net is a function chain $f(x) = f^{(2)}(f^{(1)}(x))$

  Q: How to compute $f(g(x))'$? (Lagrange notation ' for derivatives)
  A: $f(g(x))' = f'(g(x))g'(x)$

- Let: $y = g(x)$, and $z = f(g(x)) = f(y)$.

Chain rule to compute derivatives

Eqs (6), (7) in MDL-BackProp-RGrosse.pdf

- Deep net is a function chain $f(x) = f^{(2)}(f^{(1)}(x))$

  Q: How to compute $f(g(x))'$? (Lagrange notation ' for derivatives)
  A: $f(g(x))' = f'(g(x))g'(x)$

- Let: $y = g(x)$, and $z = f(g(x)) = f(y)$.
- Q: What is the other (Leibniz) notation of $f(g(x))'$? :

Chain rule to compute derivatives

Eqs (6), (7) in MDL-BackProp-RGrosse.pdf

- Deep net is a function chain $f(x) = f^{(2)}(f^{(1)}(x))$

  Q: How to compute $f(g(x))'$? (Lagrange notation ' for derivatives)
  A: $f(g(x))' = f'(g(x))g'(x)$

- Let: $y = g(x)$, and $z = f(g(x)) = f(y)$.
- Q: What is the other (Leibniz) notation of $f(g(x))'$? :

$$\frac{dz}{dx} =$$

Chain rule to compute derivatives

Eqs (6), (7) in MDL-BackProp-RGrosse.pdf

- Deep net is a function chain $f(x) = f^{(2)}(f^{(1)}(x))$

  Q: How to compute $f(g(x))'$? (Lagrange notation ' for derivatives)
  A: $f(g(x))' = f'(g(x))g'(x)$

- Let: $y = g(x)$, and $z = f(g(x)) = f(y)$.
- Q: What is the other (Leibniz) notation of $f(g(x))'$? :

$$\frac{dz}{dx} = \frac{dz}{dy}\frac{dy}{dx}$$

Chain rule to compute derivatives

Eqs (6), (7) in MDL-BackProp-RGrosse.pdf

- Deep net is a function chain $f(x) = f^{(2)}(f^{(1)}(x))$

  Q: How to compute $f(g(x))'$? (Lagrange notation ' for derivatives)
  A: $f(g(x))' = f'(g(x))g'(x)$

- Let: $y = g(x)$, and $z = f(g(x)) = f(y)$.
- Q: What is the other (Leibniz) notation of $f(g(x))'$? :

$$\frac{dz}{dx} = \frac{dz}{dy}\frac{dy}{dx}$$

- Q: Multi-variate chain rule; for $\frac{d}{dt}f(x(t), y(t))$, then

Chain rule to compute derivatives

Eqs (6), (7) in MDL-BackProp-RGrosse.pdf

- Deep net is a function chain $f(x) = f^{(2)}(f^{(1)}(x))$

  Q: How to compute $f(g(x))'$? (Lagrange notation ' for derivatives)
  A: $f(g(x))' = f'(g(x))g'(x)$

- Let: $y = g(x)$, and $z = f(g(x)) = f(y)$.
- Q: What is the other (Leibniz) notation of $f(g(x))'$? :

$$\frac{dz}{dx} = \frac{dz}{dy}\frac{dy}{dx}$$

- Q: Multi-variate chain rule; for $\frac{d}{dt}f(x(t), y(t))$, then

$$\frac{d}{dt}f(x(t), y(t)) = \frac{\partial f}{\partial x}\frac{dx}{dt} + \frac{\partial f}{\partial y}\frac{dy}{dt}$$

Chain rule to compute derivatives

Eqs (6), (7) in MDL-BackProp-RGrosse.pdf

- Deep net is a function chain $f(x) = f^{(2)}(f^{(1)}(x))$

  Q: How to compute $f(g(x))'$? (Lagrange notation ' for derivatives)
  A: $f(g(x))' = f'(g(x))g'(x)$

- Let: $y = g(x)$, and $z = f(g(x)) = f(y)$.
- Q: What is the other (Leibniz) notation of $f(g(x))'$? :

$$\frac{dz}{dx} = \frac{dz}{dy}\frac{dy}{dx}$$

- Q: Multi-variate chain rule; for $\frac{d}{dt}f(x(t), y(t))$, then

$$\frac{d}{dt}f(x(t), y(t)) = \frac{\partial f}{\partial x}\frac{dx}{dt} + \frac{\partial f}{\partial y}\frac{dy}{dt}$$

- Interpretation: How much output changes if parameters slightly change.

Assignment: Compute loss $\mathcal{L}$ derivatives

2.1 in MDL.BackProp-RGrosse.pdf

Model: $\mathcal{L} = \frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2$, where $t =$ true label

Assignment: Compute loss $\mathcal{L}$ derivatives

2.1 in MDL.BackProp-RGrosse.pdf

Model: $\mathcal{L} = \frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2$, where $t =$ true label

Q: a step of gradient descent?

Assignment: Compute loss $\mathcal{L}$ derivatives

2.1 in MDL.BackProp-RGrosse.pdf

Model: $\mathcal{L} = \frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2$, where $t =$ true label

Q: a step of gradient descent? A: $w^* = w - \epsilon \frac{\partial}{\partial w}\mathcal{L}$ and $b^* = b - \epsilon \frac{\partial}{\partial b}\mathcal{L}$

Assignment: Compute loss $\mathcal{L}$ derivatives

2.1 in MDL.BackProp-RGrosse.pdf

Model: $\mathcal{L} = \frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2$, where $t =$ true label

Q: a step of gradient descent? A: $w^* = w - \epsilon \frac{\partial}{\partial w}\mathcal{L}$ and $b^* = b - \epsilon \frac{\partial}{\partial b}\mathcal{L}$

Q: What terms to compute? (leave σ; and/or use σ')

Assignment: Compute loss $\mathcal{L}$ derivatives

2.1 in MDL.BackProp-RGrosse.pdf

Model: $\mathcal{L} = \frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2$, where $t =$ true label

Q: a step of gradient descent? A: $w^* = w - \epsilon \frac{\partial}{\partial w}\mathcal{L}$ and $b^* = b - \epsilon \frac{\partial}{\partial b}\mathcal{L}$

Q: What terms to compute? (leave σ; and/or use σ')

A: Derivative to w

$$\frac{\partial}{\partial w}\mathcal{L} = \frac{\partial}{\partial w}\left[\frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2\right]$$

A: Derivative to b

$$\frac{\partial}{\partial b}\mathcal{L} = \frac{\partial}{\partial b}\left[\frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2\right] =$$

Assignment: Compute loss $\mathcal{L}$ derivatives

2.1 in MDL.BackProp-RGrosse.pdf

Model: $\mathcal{L} = \frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2$, where $t =$ true label

Q: a step of gradient descent? A: $w^* = w - \epsilon \frac{\partial}{\partial w}\mathcal{L}$ and $b^* = b - \epsilon \frac{\partial}{\partial b}\mathcal{L}$

Q: What terms to compute? (leave σ; and/or use σ')

A: Derivative to w

$$\frac{\partial}{\partial w}\mathcal{L} = \frac{\partial}{\partial w}\left[\frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2\right]$$

$$= \left(\sigma(\text{wx+b})\text{-t}\right)\frac{\partial}{\partial w}\left(\sigma(\text{wx+b})\text{-t}\right)$$

$$= \left(\sigma(\text{wx+b})\text{-t}\right)\sigma'(\text{wx+b})\frac{\partial}{\partial w}(\text{wx+b})$$

$$= \left(\sigma(\text{wx+b})\text{-t}\right)\sigma'(\text{wx+b})x$$

A: Derivative to b

$$\frac{\partial}{\partial b}\mathcal{L} = \frac{\partial}{\partial b}\left[\frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2\right] =$$

$$\left(\sigma(\text{wx+b})\text{-t}\right)\frac{\partial}{\partial b}\left(\sigma(\text{wx+b})\text{-t}\right) =$$

$$\left(\sigma(\text{wx+b})\text{-t}\right)\sigma'(\text{wx+b})\frac{\partial}{\partial b}(\text{wx+b}) =$$

$$\left(\sigma(\text{wx+b})\text{-t}\right)\sigma'(\text{wx+b})$$

Assignment: Compute loss $\mathcal{L}$ derivatives

2.1 in MDL.BackProp-RGrosse.pdf

Model: $\mathcal{L} = \frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2$, where $t =$ true label

Q: a step of gradient descent? A: $w^* = w - \epsilon \frac{\partial}{\partial w}\mathcal{L}$ and $b^* = b - \epsilon \frac{\partial}{\partial b}\mathcal{L}$

Q: What terms to compute? (leave σ; and/or use σ')

A: Derivative to w

$$\frac{\partial}{\partial w}\mathcal{L} = \frac{\partial}{\partial w}\left[\frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2\right]$$

$$= \left(\sigma(\text{wx+b})\text{-t}\right)\frac{\partial}{\partial w}\left(\sigma(\text{wx+b})\text{-t}\right)$$

$$= \left(\sigma(\text{wx+b})\text{-t}\right)\sigma'(\text{wx+b})\frac{\partial}{\partial w}(\text{wx+b})$$

$$= \left(\sigma(\text{wx+b})\text{-t}\right)\sigma'(\text{wx+b})x$$

A: Derivative to b

$$\frac{\partial}{\partial b}\mathcal{L} = \frac{\partial}{\partial b}\left[\frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2\right] =$$

$$\left(\sigma(\text{wx+b})\text{-t}\right)\frac{\partial}{\partial b}\left(\sigma(\text{wx+b})\text{-t}\right) =$$

$$\left(\sigma(\text{wx+b})\text{-t}\right)\sigma'(\text{wx+b})\frac{\partial}{\partial b}(\text{wx+b}) =$$

$$\left(\sigma(\text{wx+b})\text{-t}\right)\sigma'(\text{wx+b})$$

Q: What are the disadvantages of this approach?

Assignment: Compute loss $\mathcal{L}$ derivatives

2.1 in MDL.BackProp-RGrosse.pdf

Model: $\mathcal{L} = \frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2$, where $t =$ true label

Q: a step of gradient descent? A: $w^* = w - \epsilon\frac{\partial}{\partial w}\mathcal{L}$ and $b^* = b - \epsilon\frac{\partial}{\partial b}\mathcal{L}$

Q: What terms to compute? (leave σ; and/or use σ')

A: Derivative to w

$$\frac{\partial}{\partial w}\mathcal{L} = \frac{\partial}{\partial w}\left[\frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2\right]$$

$$= \left(\sigma(\text{wx+b})\text{-t}\right)\frac{\partial}{\partial w}\left(\sigma(\text{wx+b})\text{-t}\right)$$

$$= \left(\sigma(\text{wx+b})\text{-t}\right)\sigma'(\text{wx+b})\frac{\partial}{\partial w}(\text{wx+b})$$

$$= \left(\sigma(\text{wx+b})\text{-t}\right)\sigma'(\text{wx+b})x$$

A: Derivative to b

$$\frac{\partial}{\partial b}\mathcal{L} = \frac{\partial}{\partial b}\left[\frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2\right] =$$

$$\left(\sigma(\text{wx+b})\text{-t}\right)\frac{\partial}{\partial b}\left(\sigma(\text{wx+b})\text{-t}\right) =$$

$$\left(\sigma(\text{wx+b})\text{-t}\right)\sigma'(\text{wx+b})\frac{\partial}{\partial b}(\text{wx+b}) =$$

$$\left(\sigma(\text{wx+b})\text{-t}\right)\sigma'(\text{wx+b})$$

Q: What are the disadvantages of this approach? A: Redundant computations.

More efficient computation of loss $\mathcal{L}$ derivatives

Model: $\mathcal{L} = \frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2$

Decompose in: $z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \frac{1}{2}(y - t)^2$

Q: First rewrite using chain rule in Leipniz notation ($\frac{d\mathcal{L}}{dw} = ...; \frac{d\mathcal{L}}{db} = ...$)?

More efficient computation of loss $\mathcal{L}$ derivatives

Model: $\mathcal{L} = \frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2$

Decompose in: $z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \frac{1}{2}(y - t)^2$

Q: First rewrite using chain rule in Leipniz notation ($\frac{d\mathcal{L}}{dw} = ...; \frac{d\mathcal{L}}{db} = ...$)?

A: Derivative to w

$$\frac{d\mathcal{L}}{dw} = \frac{d\mathcal{L}}{dy}\frac{dy}{dz}\frac{dz}{dw}$$

A: Derivative to b

$$\frac{d\mathcal{L}}{db} = \frac{d\mathcal{L}}{dy}\frac{dy}{dz}\frac{dz}{db}$$

More efficient computation of loss $\mathcal{L}$ derivatives

Model: $\mathcal{L} = \frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2$

Decompose in: $z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \frac{1}{2}(y - t)^2$

Q: First rewrite using chain rule in Leipniz notation ($\frac{d\mathcal{L}}{dw} = ...; \frac{d\mathcal{L}}{db} = ...$)?

A: Derivative to w

$$\frac{d\mathcal{L}}{dw} = \frac{d\mathcal{L}}{dy}\frac{dy}{dz}\frac{dz}{dw}$$

A: Derivative to b

$$\frac{d\mathcal{L}}{db} = \frac{d\mathcal{L}}{dy}\frac{dy}{dz}\frac{dz}{db}$$

Q: Nr unique terms to compute both derivatives?

More efficient computation of loss $\mathcal{L}$ derivatives

Model: $\mathcal{L} = \frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2$

Decompose in: $z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \frac{1}{2}(y - t)^2$

Q: First rewrite using chain rule in Leipniz notation ($\frac{d\mathcal{L}}{dw} = ...; \frac{d\mathcal{L}}{db} = ...$)?

A: Derivative to w

$$\frac{d\mathcal{L}}{dw} = \frac{d\mathcal{L}}{dy}\frac{dy}{dz}\frac{dz}{dw}$$

A: Derivative to b

$$\frac{d\mathcal{L}}{db} = \frac{d\mathcal{L}}{dy}\frac{dy}{dz}\frac{dz}{db}$$

Q: Nr unique terms to compute both derivatives? A: 4 (efficient re-use)

More efficient computation of loss $\mathcal{L}$ derivatives

Model: $\mathcal{L} = \frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2$

Decompose in: $z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \frac{1}{2}(y - t)^2$

Q: First rewrite using chain rule in Leipniz notation ($\frac{d\mathcal{L}}{dw} = ...; \frac{d\mathcal{L}}{db} = ...$)?

A: Derivative to w

$$\frac{d\mathcal{L}}{dw} = \frac{d\mathcal{L}}{dy}\frac{dy}{dz}\frac{dz}{dw}$$

A: Derivative to b

$$\frac{d\mathcal{L}}{db} = \frac{d\mathcal{L}}{dy}\frac{dy}{dz}\frac{dz}{db}$$

Q: Nr unique terms to compute both derivatives? A: 4 (efficient re-use)

$$(1): \frac{d\mathcal{L}}{dy} = \qquad (2): \frac{d\mathcal{L}}{dz} = \qquad (3): \frac{d\mathcal{L}}{dw} = \qquad (4): \frac{d\mathcal{L}}{db} =$$

More efficient computation of loss $\mathcal{L}$ derivatives

Model: $\mathcal{L} = \frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2$

Decompose in: $z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \frac{1}{2}(y-t)^2$

Q: First rewrite using chain rule in Leipniz notation ($\frac{d\mathcal{L}}{dw} = ...; \frac{d\mathcal{L}}{db} = ...$)?

A: Derivative to w

$$\frac{d\mathcal{L}}{dw} = \frac{d\mathcal{L}}{dy}\frac{dy}{dz}\frac{dz}{dw}$$

A: Derivative to b

$$\frac{d\mathcal{L}}{db} = \frac{d\mathcal{L}}{dy}\frac{dy}{dz}\frac{dz}{db}$$

Q: Nr unique terms to compute both derivatives? A: 4 (efficient re-use)

$$(1): \frac{d\mathcal{L}}{dy} = y - t, \quad (2): \frac{d\mathcal{L}}{dz} = \frac{d\mathcal{L}}{dy}\sigma'(z), \quad (3): \frac{d\mathcal{L}}{dw} = \frac{d\mathcal{L}}{dz}x, \quad (4): \frac{d\mathcal{L}}{db} = \frac{d\mathcal{L}}{dz}.$$

More efficient computation of loss $\mathcal{L}$ derivatives

Model: $\mathcal{L} = \frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2$

Decompose in: $z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \frac{1}{2}(y - t)^2$

Q: First rewrite using chain rule in Leipniz notation ($\frac{d\mathcal{L}}{dw} = ...; \frac{d\mathcal{L}}{db} = ...$)?

A: Derivative to w

$$\frac{d\mathcal{L}}{dw} = \frac{d\mathcal{L}}{dy}\frac{dy}{dz}\frac{dz}{dw}$$

A: Derivative to b

$$\frac{d\mathcal{L}}{db} = \frac{d\mathcal{L}}{dy}\frac{dy}{dz}\frac{dz}{db}$$

Q: Nr unique terms to compute both derivatives? A: 4 (efficient re-use)

$$(1): \frac{d\mathcal{L}}{dy} = y - t, \quad (2): \frac{d\mathcal{L}}{dz} = \frac{d\mathcal{L}}{dy}\sigma'(z), \quad (3): \frac{d\mathcal{L}}{dw} = \frac{d\mathcal{L}}{dz}x, \quad (4): \frac{d\mathcal{L}}{db} = \frac{d\mathcal{L}}{dz}.$$

Q: Dependency order of computation?

More efficient computation of loss $\mathcal{L}$ derivatives

Model: $\mathcal{L} = \frac{1}{2}\left(\sigma(\text{wx+b})\text{-t}\right)^2$

Decompose in: $z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \frac{1}{2}(y - t)^2$

Q: First rewrite using chain rule in Leipniz notation ($\frac{d\mathcal{L}}{dw} = ...; \frac{d\mathcal{L}}{db} = ...$)?

A: Derivative to w

$$\frac{d\mathcal{L}}{dw} = \frac{d\mathcal{L}}{dy}\frac{dy}{dz}\frac{dz}{dw}$$

A: Derivative to b

$$\frac{d\mathcal{L}}{db} = \frac{d\mathcal{L}}{dy}\frac{dy}{dz}\frac{dz}{db}$$

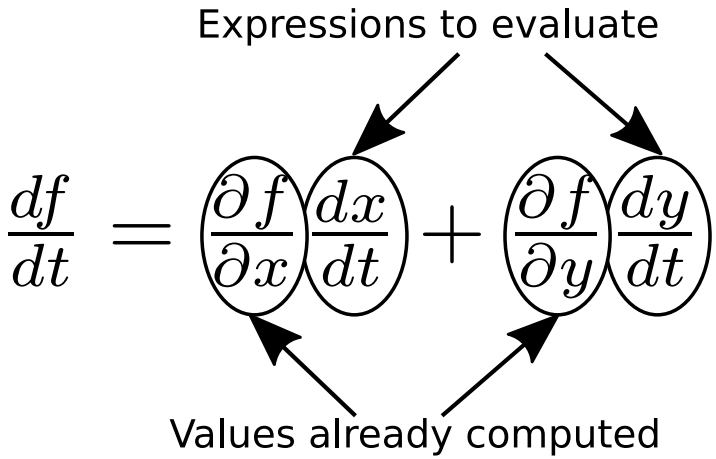
Q: Nr unique terms to compute both derivatives? A: 4 (efficient re-use)

$$(1): \frac{d\mathcal{L}}{dy} = y - t, \quad (2): \frac{d\mathcal{L}}{dz} = \frac{d\mathcal{L}}{dy}\sigma'(z), \quad (3): \frac{d\mathcal{L}}{dw} = \frac{d\mathcal{L}}{dz}x, \quad (4): \frac{d\mathcal{L}}{db} = \frac{d\mathcal{L}}{dz}.$$

Q: Dependency order of computation? A: Starts backwards (ie: with the loss).

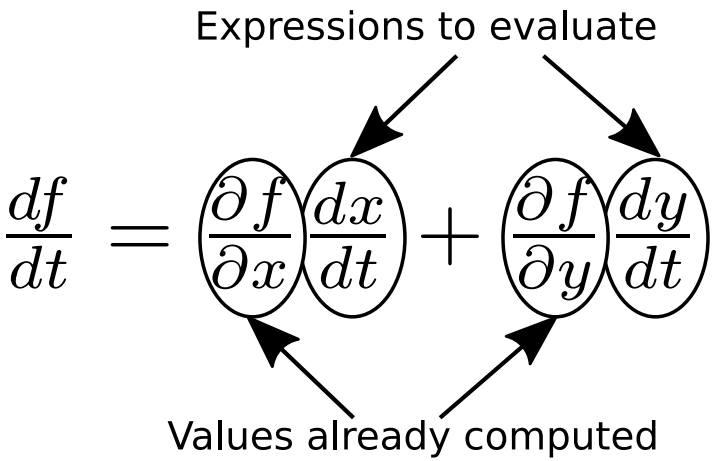
Re-using pre-computed values

2.3 in MDL-BackProp-RGrosse.pdf Bar notation credits to Roger Grosse



Re-using pre-computed values

2.3 in MDL-BackProp-RGrosse.pdf Bar notation credits to Roger Grosse

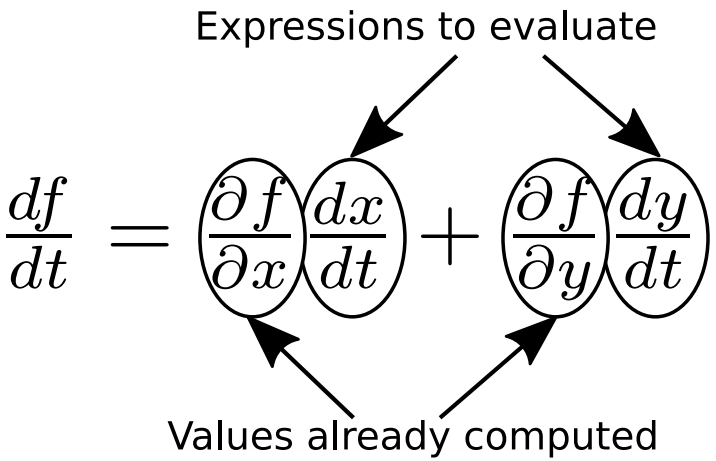


Bar notation:

- Backprop is an algorithm to compute values
- Bar notation[1] for computed values: $\overline{y} = \frac{d\mathcal{E}}{dy}$
- Less cluttered and emphasizes value re-use

Re-using pre-computed values

2.3 in MDL-BackProp-RGrosse.pdf Bar notation credits to Roger Grosse



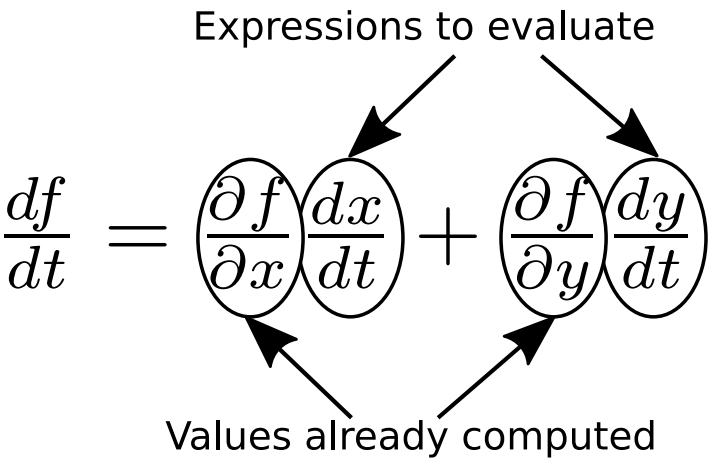
Bar notation:

- Backprop is an algorithm to compute values
- Bar notation[1] for computed values: $\overline{y} = \frac{d\mathcal{E}}{dy}$
- Less cluttered and emphasizes value re-use

For: $z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \frac{1}{2}(y - t)^2$

Re-using pre-computed values

2.3 in MDL-BackProp-RGrosse.pdf Bar notation credits to Roger Grosse



Bar notation:

- Backprop is an algorithm to compute values
- Bar notation[1] for computed values: $\overline{y} = \frac{d\mathcal{E}}{dy}$
- Less cluttered and emphasizes value re-use

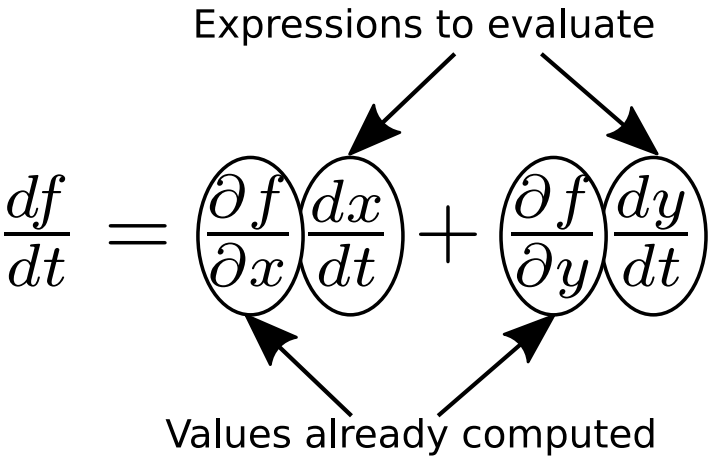
For: $z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \frac{1}{2}(y - t)^2$

Instead of: (from prev. slide)

$$(1) : \frac{d\mathcal{L}}{dy} = y - t, \quad (2) : \frac{d\mathcal{L}}{dz} = \frac{d\mathcal{L}}{dy}\sigma'(z), \quad (3) : \frac{d\mathcal{L}}{dw} = \frac{d\mathcal{L}}{dz}x, \quad (4) : \frac{d\mathcal{L}}{db} = \frac{d\mathcal{L}}{dz}.$$

Re-using pre-computed values

2.3 in MDL-BackProp-RGrosse.pdf Bar notation credits to Roger Grosse



Bar notation:

- Backprop is an algorithm to compute values
- Bar notation[1] for computed values: $\overline{y} = \frac{d\mathcal{E}}{dy}$
- Less cluttered and emphasizes value re-use

For: $z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \frac{1}{2}(y - t)^2$

Instead of: (from prev. slide)

$$(1) : \frac{d\mathcal{L}}{dy} = y - t, \quad (2) : \frac{d\mathcal{L}}{dz} = \frac{d\mathcal{L}}{dy}\sigma'(z), \quad (3) : \frac{d\mathcal{L}}{dw} = \frac{d\mathcal{L}}{dz}x, \quad (4) : \frac{d\mathcal{L}}{db} = \frac{d\mathcal{L}}{dz}.$$

Write:

$$(1) : \overline{y} = y - t, \qquad (2) : \overline{z} = \overline{y}\sigma'(z), \qquad (3) : \overline{w} = \overline{z}x, \qquad (4) : \overline{b} = \overline{z}.$$

Questions?

Questions?

From previous slide:

$$\text{For: } z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \tfrac{1}{2}(y - t)^2$$

$$(1) : \overline{y} = y - t, \qquad (2) : \overline{z} = \overline{y}\sigma'(z), \qquad (3) : \overline{w} = \overline{z}x, \qquad (4) : \overline{b} = \overline{z}.$$

Let: $x = 2$; $w = 3$; $b = 4$; $t = 5$; $\sigma(z) = \max\{0, z\}$, learning rate $= 0.1$

Q: How to compute the new values after a step of gradient descent?

Questions?

From previous slide:

$$\text{For: } z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \tfrac{1}{2}(y - t)^2$$

$$(1) : \overline{y} = y - t, \qquad (2) : \overline{z} = \overline{y}\sigma'(z), \qquad (3) : \overline{w} = \overline{z}x, \qquad (4) : \overline{b} = \overline{z}.$$

Let: $x = 2$; $w = 3$; $b = 4$; $t = 5$; $\sigma(z) = \max\{0, z\}$, learning rate $= 0.1$

Q: How to compute the new values after a step of gradient descent?

Forward pass: (what to compute)?:

Questions?

From previous slide:

$$\text{For: } z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \tfrac{1}{2}(y - t)^2$$

$$(1) : \overline{y} = y - t, \qquad (2) : \overline{z} = \overline{y}\sigma'(z), \qquad (3) : \overline{w} = \overline{z}x, \qquad (4) : \overline{b} = \overline{z}.$$

Let: $x = 2$; $w = 3$; $b = 4$; $t = 5$; $\sigma(z) = \max\{0, z\}$, learning rate $= 0.1$

Q: How to compute the new values after a step of gradient descent?

Forward pass: (what to compute)?: $z, y, \mathcal{L}$;

Questions?

From previous slide:

$$\text{For: } z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \tfrac{1}{2}(y - t)^2$$

$$(1) : \overline{y} = y - t, \qquad (2) : \overline{z} = \overline{y}\sigma'(z), \qquad (3) : \overline{w} = \overline{z}x, \qquad (4) : \overline{b} = \overline{z}.$$

Let: $x = 2$; $w = 3$; $b = 4$; $t = 5$; $\sigma(z) = \max\{0, z\}$, learning rate $= 0.1$

Q: How to compute the new values after a step of gradient descent?

Forward pass: (what to compute)?: $z, y, \mathcal{L}$;
Outcome:

Questions?

From previous slide:

$$\text{For: } z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \tfrac{1}{2}(y - t)^2$$

$$(1) : \overline{y} = y - t, \qquad (2) : \overline{z} = \overline{y}\sigma'(z), \qquad (3) : \overline{w} = \overline{z}x, \qquad (4) : \overline{b} = \overline{z}.$$

Let: $x = 2$; $w = 3$; $b = 4$; $t = 5$; $\sigma(z) = \max\{0, z\}$, learning rate $= 0.1$

Q: How to compute the new values after a step of gradient descent?

Forward pass: (what to compute)?: $z, y, \mathcal{L}$;
Outcome: $z = 10$, $y = 10$; $\mathcal{L} = 12.5$

Questions?

From previous slide:

$$\text{For: } z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \tfrac{1}{2}(y - t)^2$$

$$(1) : \overline{y} = y - t, \qquad (2) : \overline{z} = \overline{y}\sigma'(z), \qquad (3) : \overline{w} = \overline{z}x, \qquad (4) : \overline{b} = \overline{z}.$$

Let: $x = 2$; $w = 3$; $b = 4$; $t = 5$; $\sigma(z) = \max\{0, z\}$, learning rate $= 0.1$

Q: How to compute the new values after a step of gradient descent?

Forward pass: (what to compute)?: $z, y, \mathcal{L}$;
Outcome: $z = 10$, $y = 10$; $\mathcal{L} = 12.5$

Backward pass: (what to compute?)

Questions?

From previous slide:

$$\text{For: } z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \tfrac{1}{2}(y - t)^2$$

$$(1) : \overline{y} = y - t, \qquad (2) : \overline{z} = \overline{y}\sigma'(z), \qquad (3) : \overline{w} = \overline{z}x, \qquad (4) : \overline{b} = \overline{z}.$$

Let: $x = 2$; $w = 3$; $b = 4$; $t = 5$; $\sigma(z) = \max\{0, z\}$, learning rate $= 0.1$

Q: How to compute the new values after a step of gradient descent?

Forward pass: (what to compute)?: $z, y, \mathcal{L}$;
Outcome: $z = 10$, $y = 10$; $\mathcal{L} = 12.5$

Backward pass: (what to compute?) $\overline{y}$, $\overline{z}$, $\overline{w}$, $\overline{b}$;

Questions?

From previous slide:

$$\text{For: } z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \tfrac{1}{2}(y - t)^2$$

$$(1) : \overline{y} = y - t, \qquad (2) : \overline{z} = \overline{y}\sigma'(z), \qquad (3) : \overline{w} = \overline{z}x, \qquad (4) : \overline{b} = \overline{z}.$$

Let: $x = 2$; $w = 3$; $b = 4$; $t = 5$; $\sigma(z) = \max\{0, z\}$, learning rate $= 0.1$

Q: How to compute the new values after a step of gradient descent?

Forward pass: (what to compute)?: $z, y, \mathcal{L}$;
Outcome: $z = 10$, $y = 10$; $\mathcal{L} = 12.5$

Backward pass: (what to compute?) $\overline{y}$, $\overline{z}$, $\overline{w}$, $\overline{b}$;
Outcome:

Questions?

From previous slide:

$$\text{For: } z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \tfrac{1}{2}(y - t)^2$$

$$(1) : \overline{y} = y - t, \qquad (2) : \overline{z} = \overline{y}\sigma'(z), \qquad (3) : \overline{w} = \overline{z}x, \qquad (4) : \overline{b} = \overline{z}.$$

Let: $x = 2$; $w = 3$; $b = 4$; $t = 5$; $\sigma(z) = \max\{0, z\}$, learning rate $= 0.1$

Q: How to compute the new values after a step of gradient descent?

Forward pass: (what to compute)?: $z, y, \mathcal{L}$;
Outcome: $z = 10$, $y = 10$; $\mathcal{L} = 12.5$

Backward pass: (what to compute?) $\overline{y}$, $\overline{z}$, $\overline{w}$, $\overline{b}$;
Outcome: $\overline{y} = 5$, $\overline{z} = 5$, $\overline{w} = 10$, $\overline{b} = 5$

Questions?

From previous slide:

$$\text{For: } z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \tfrac{1}{2}(y - t)^2$$

$$(1) : \overline{y} = y - t, \qquad (2) : \overline{z} = \overline{y}\sigma'(z), \qquad (3) : \overline{w} = \overline{z}x, \qquad (4) : \overline{b} = \overline{z}.$$

Let: $x = 2$; $w = 3$; $b = 4$; $t = 5$; $\sigma(z) = \max\{0, z\}$, learning rate $= 0.1$

Q: How to compute the new values after a step of gradient descent?

Forward pass: (what to compute)?: $z, y, \mathcal{L}$;
Outcome: $z = 10$, $y = 10$; $\mathcal{L} = 12.5$

Backward pass: (what to compute?) $\overline{y}$, $\overline{z}$, $\overline{w}$, $\overline{b}$;
Outcome: $\overline{y} = 5$, $\overline{z} = 5$, $\overline{w} = 10$, $\overline{b} = 5$

Both gradient descent equations:

Questions?

From previous slide:

$$\text{For: } z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \tfrac{1}{2}(y - t)^2$$

$$(1) : \overline{y} = y - t, \qquad (2) : \overline{z} = \overline{y}\sigma'(z), \qquad (3) : \overline{w} = \overline{z}x, \qquad (4) : \overline{b} = \overline{z}.$$

Let: $x = 2$; $w = 3$; $b = 4$; $t = 5$; $\sigma(z) = \max\{0, z\}$, learning rate $= 0.1$

Q: How to compute the new values after a step of gradient descent?

Forward pass: (what to compute)?: $z, y, \mathcal{L}$;
Outcome: $z = 10$, $y = 10$; $\mathcal{L} = 12.5$

Backward pass: (what to compute?) $\overline{y}$, $\overline{z}$, $\overline{w}$, $\overline{b}$;
Outcome: $\overline{y} = 5$, $\overline{z} = 5$, $\overline{w} = 10$, $\overline{b} = 5$

Both gradient descent equations: $w^* = w - \epsilon \frac{\partial}{\partial w}\mathcal{L}$ and $b^* = b - \epsilon \frac{\partial}{\partial b}\mathcal{L}$

Questions?

From previous slide:

$$\text{For: } z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \tfrac{1}{2}(y - t)^2$$

$$(1) : \overline{y} = y - t, \qquad (2) : \overline{z} = \overline{y}\sigma'(z), \qquad (3) : \overline{w} = \overline{z}x, \qquad (4) : \overline{b} = \overline{z}.$$

Let: $x = 2$; $w = 3$; $b = 4$; $t = 5$; $\sigma(z) = \max\{0, z\}$, learning rate $= 0.1$

Q: How to compute the new values after a step of gradient descent?

Forward pass: (what to compute)?: $z, y, \mathcal{L}$;
Outcome: $z = 10$, $y = 10$; $\mathcal{L} = 12.5$

Backward pass: (what to compute?) $\overline{y}$, $\overline{z}$, $\overline{w}$, $\overline{b}$;
Outcome: $\overline{y} = 5$, $\overline{z} = 5$, $\overline{w} = 10$, $\overline{b} = 5$

Both gradient descent equations: $w^* = w - \epsilon \frac{\partial}{\partial w}\mathcal{L}$ and $b^* = b - \epsilon \frac{\partial}{\partial b}\mathcal{L}$
Updated values for w and b:

Questions?

From previous slide:

$$\text{For: } z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \tfrac{1}{2}(y - t)^2$$

$$(1) : \overline{y} = y - t, \qquad (2) : \overline{z} = \overline{y}\sigma'(z), \qquad (3) : \overline{w} = \overline{z}x, \qquad (4) : \overline{b} = \overline{z}.$$

Let: $x = 2$; $w = 3$; $b = 4$; $t = 5$; $\sigma(z) = \max\{0, z\}$, learning rate $= 0.1$

Q: How to compute the new values after a step of gradient descent?

Forward pass: (what to compute)?: $z, y, \mathcal{L}$;
Outcome: $z = 10$, $y = 10$; $\mathcal{L} = 12.5$

Backward pass: (what to compute?) $\overline{y}$, $\overline{z}$, $\overline{w}$, $\overline{b}$;
Outcome: $\overline{y} = 5$, $\overline{z} = 5$, $\overline{w} = 10$, $\overline{b} = 5$

Both gradient descent equations: $w^* = w - \epsilon \frac{\partial}{\partial w}\mathcal{L}$ and $b^* = b - \epsilon \frac{\partial}{\partial b}\mathcal{L}$
Updated values for w and b: $w^* = 3 - 1 = 2$ and $b^* = 4 - 0.5 = 3.5$

Questions?

From previous slide:

$$\text{For: } z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \tfrac{1}{2}(y - t)^2$$

$$(1) : \overline{y} = y - t, \qquad (2) : \overline{z} = \overline{y}\sigma'(z), \qquad (3) : \overline{w} = \overline{z}x, \qquad (4) : \overline{b} = \overline{z}.$$

Let: $x = 2$; $w = 3$; $b = 4$; $t = 5$; $\sigma(z) = \max\{0, z\}$, learning rate $= 0.1$

Q: How to compute the new values after a step of gradient descent?

Forward pass: (what to compute)?: $z, y, \mathcal{L}$;
Outcome: $z = 10$, $y = 10$; $\mathcal{L} = 12.5$

Backward pass: (what to compute?) $\overline{y}$, $\overline{z}$, $\overline{w}$, $\overline{b}$;
Outcome: $\overline{y} = 5$, $\overline{z} = 5$, $\overline{w} = 10$, $\overline{b} = 5$

Both gradient descent equations: $w^* = w - \epsilon \frac{\partial}{\partial w}\mathcal{L}$ and $b^* = b - \epsilon \frac{\partial}{\partial b}\mathcal{L}$
Updated values for w and b: $w^* = 3 - 1 = 2$ and $b^* = 4 - 0.5 = 3.5$

New loss value:

Questions?

From previous slide:

$$\text{For: } z = wx + b, \quad y = \sigma(z), \quad \mathcal{L} = \tfrac{1}{2}(y - t)^2$$

$$(1) : \overline{y} = y - t, \qquad (2) : \overline{z} = \overline{y}\sigma'(z), \qquad (3) : \overline{w} = \overline{z}x, \qquad (4) : \overline{b} = \overline{z}.$$

Let: $x = 2$; $w = 3$; $b = 4$; $t = 5$; $\sigma(z) = \max\{0, z\}$, learning rate $= 0.1$

Q: How to compute the new values after a step of gradient descent?

Forward pass: (what to compute)?: $z, y, \mathcal{L}$;
Outcome: $z = 10$, $y = 10$; $\mathcal{L} = 12.5$

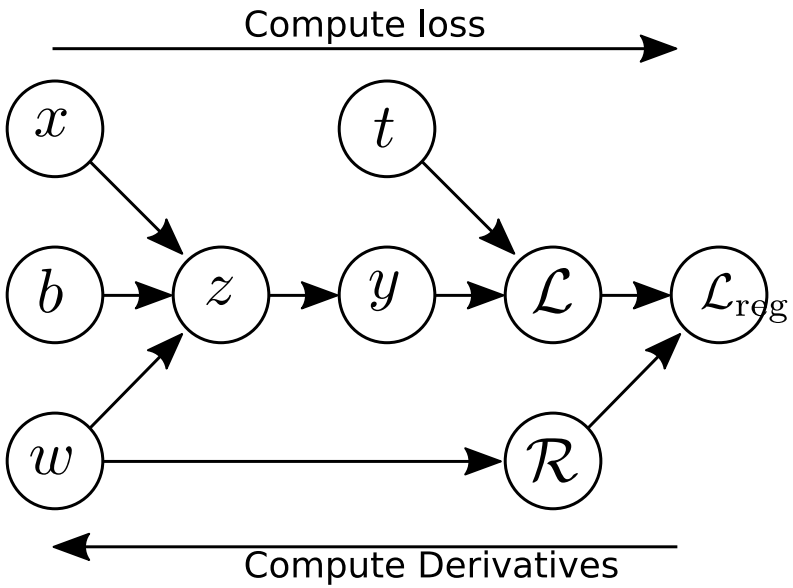
Backward pass: (what to compute?) $\overline{y}$, $\overline{z}$, $\overline{w}$, $\overline{b}$;
Outcome: $\overline{y} = 5$, $\overline{z} = 5$, $\overline{w} = 10$, $\overline{b} = 5$

Both gradient descent equations: $w^* = w - \epsilon \frac{\partial}{\partial w}\mathcal{L}$ and $b^* = b - \epsilon \frac{\partial}{\partial b}\mathcal{L}$
Updated values for w and b: $w^* = 3 - 1 = 2$ and $b^* = 4 - 0.5 = 3.5$

New loss value: $y = 2 \times 2 + 3.5 = 7.5$, $\mathcal{L} = \frac{1}{2}(7.5 - 5)^2 = \frac{1}{2}6.25 = 3.125$

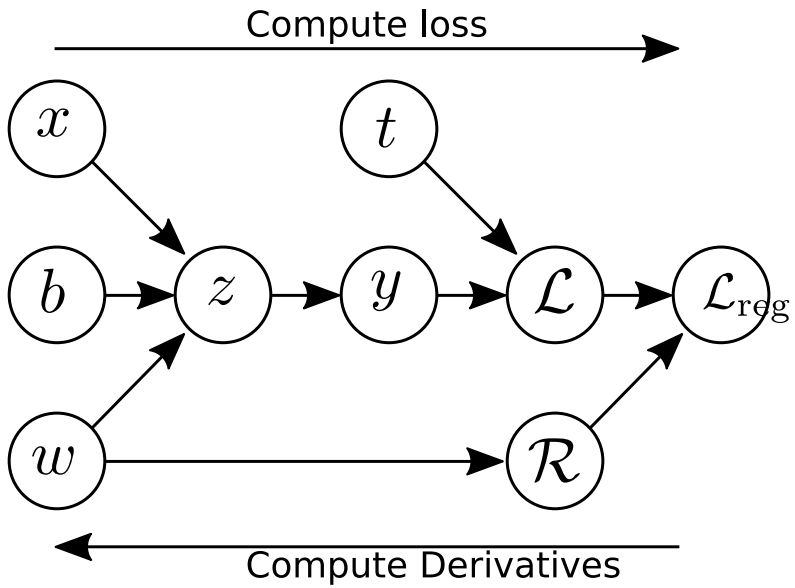
Computational graph

Book: Section 6.5.1 and 2.4 in MDL03.4.BackProp-Background-RGrosse.pdf



Computational graph

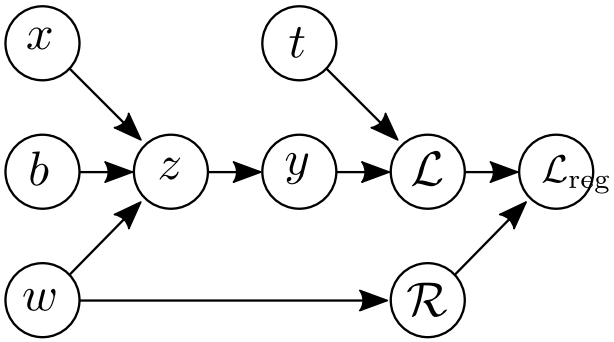
Book: Section 6.5.1 and 2.4 in MDL03.4.BackProp-Background-RGrosse.pdf



- A node is a variable: scalar, vector, matrix, tensor, other node, etc.
- An operation: simple function of 2 variables (often omitted)
- Operation have single output (possibly multiple entries)
- $x \rightarrow y$: y is computed by applying operation to x.

Backpropagation algorithm

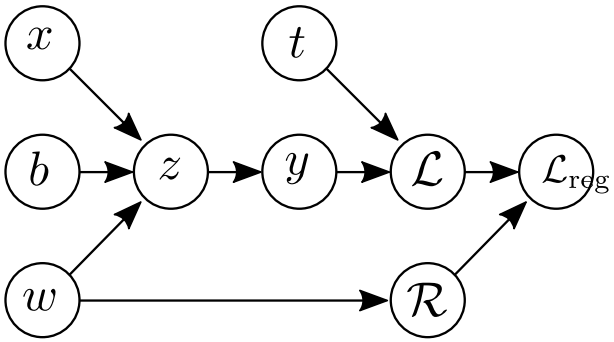
Book: Algorithms 6.1 and 6.2; and 2.4 in MDL03.4.BackProp-Background-RGrosse.pdf



Topological ordering:
Linear ordering of vertices so for every directed edge uv, vertex u comes before v in the ordering.

Backpropagation algorithm

Book: Algorithms 6.1 and 6.2; and 2.4 in MDL03.4.BackProp-Background-RGrosse.pdf

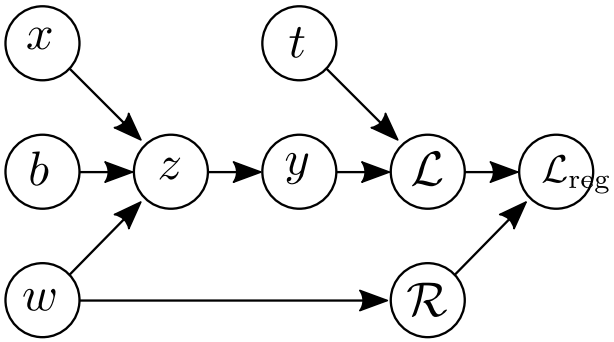


Topological ordering:
Linear ordering of vertices so for every directed edge uv, vertex u comes before v in the ordering.

Q: Topological ordering of graph?

Backpropagation algorithm

Book: Algorithms 6.1 and 6.2; and 2.4 in MDL03.4.BackProp-Background-RGrosse.pdf

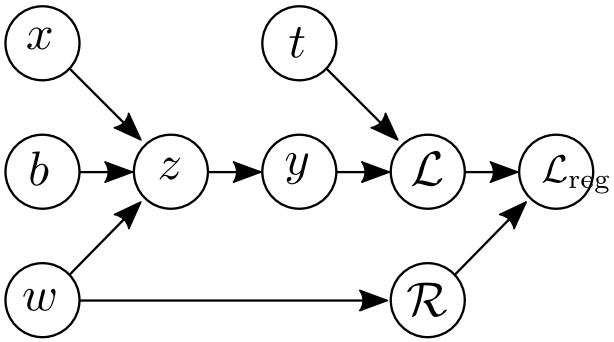


Topological ordering:
Linear ordering of vertices so for every directed edge uv, vertex u comes before v in the ordering.

Q: Topological ordering of graph?
A: $(w, b, x, z, y, t, \mathcal{L}, \mathcal{R}, \mathcal{L}_{\text{reg}})$

Backpropagation algorithm

Book: Algorithms 6.1 and 6.2; and 2.4 in MDL03.4.BackProp-Background-RGrosse.pdf



Topological ordering:
Linear ordering of vertices so for every directed edge uv, vertex u comes before v in the ordering.

Q: Topological ordering of graph?
A: $(w, b, x, z, y, t, \mathcal{L}, \mathcal{R}, \mathcal{L}_{\text{reg}})$

Backpropagation algorithm to compute gradients of node n_N:

1. Create topological ordering of graph

Forward pass $=$

2. For $i = 1, 2, \ldots N$:
   Evaluate n_i using its function $f^{(i)}(n_i)$

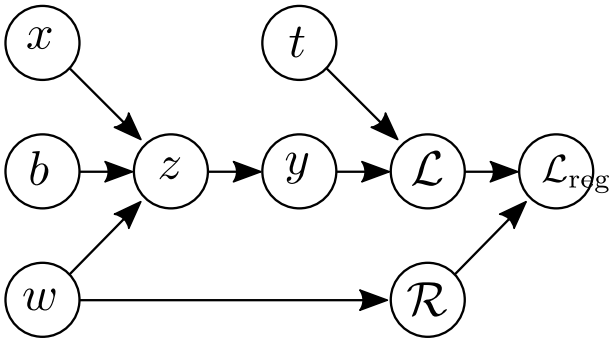
Backward pass $=$

3. $\overline{n_N} = 1$ (gradient of a node wrt itself is 1; $\frac{df}{df} = 1$)
4. For $i = N - 1, \ldots 1$:
   $\overline{n_i} = \sum_{n_j \in \text{Children}(n_i)} \overline{n_j} \frac{\partial n_j}{\partial n_i}$

Note on step 4: node only gets gradients wrt children nodes

Example: Regularized logistic least squares regression

2.4 in MDL03.4.BackProp-Background-RGrosse.pdf



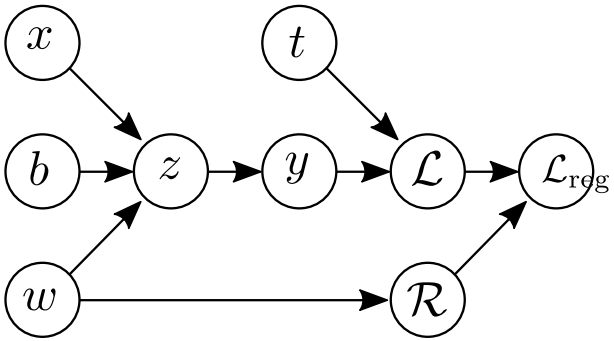
Q: What values to compute? (ie: terms with a bar on top)

Forward pass:

$$z = wx + b$$
$$y = \sigma(z)$$
$$\mathcal{L} = \frac{1}{2}(y - t)^2$$
$$\mathcal{R} = \frac{1}{2}w^2$$
$$\mathcal{L}_{reg} = \mathcal{L} + \lambda\mathcal{R}$$

Example: Regularized logistic least squares regression

2.4 in MDL03.4.BackProp-Background-RGrosse.pdf



Forward pass:

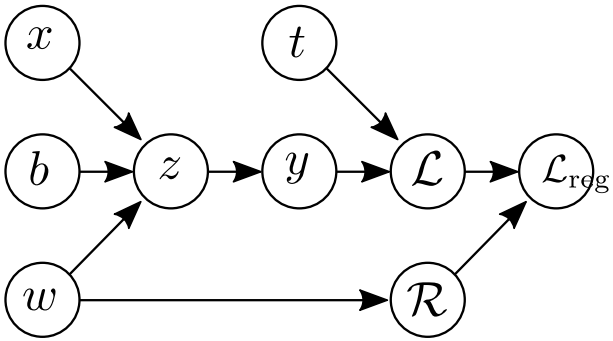
$$z = wx + b$$
$$y = \sigma(z)$$
$$\mathcal{L} = \frac{1}{2}(y - t)^2$$
$$\mathcal{R} = \frac{1}{2}w^2$$
$$\mathcal{L}_{reg} = \mathcal{L} + \lambda\mathcal{R}$$

Q: What values to compute? (ie: terms with a bar on top)

$$\overline{\mathcal{L}_{reg}} =$$
$$\overline{\mathcal{R}} =$$
$$\overline{\mathcal{L}} =$$
$$\overline{y} =$$
$$\overline{z} =$$
$$\overline{w} =$$
$$\overline{b} =$$

Example: Regularized logistic least squares regression

2.4 in MDL03.4.BackProp-Background-RGrosse.pdf



Forward pass:

$$z = wx + b$$
$$y = \sigma(z)$$
$$\mathcal{L} = \frac{1}{2}(y - t)^2$$
$$\mathcal{R} = \frac{1}{2}w^2$$
$$\mathcal{L}_{reg} = \mathcal{L} + \lambda\mathcal{R}$$

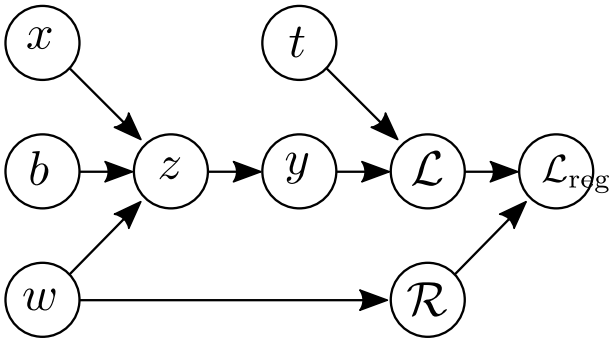
Q: What values to compute? (ie: terms with a bar on top)

Q: Write in computed $\times$ expressions to-evaluate?

$$\overline{\mathcal{L}_{reg}} =$$
$$\overline{\mathcal{R}} =$$
$$\overline{\mathcal{L}} =$$
$$\overline{y} =$$
$$\overline{z} =$$
$$\overline{w} =$$
$$\overline{b} =$$

Example: Regularized logistic least squares regression

2.4 in MDL03.4.BackProp-Background-RGrosse.pdf



Forward pass:

$$z = wx + b$$
$$y = \sigma(z)$$
$$\mathcal{L} = \frac{1}{2}(y - t)^2$$
$$\mathcal{R} = \frac{1}{2}w^2$$
$$\mathcal{L}_{reg} = \mathcal{L} + \lambda\mathcal{R}$$

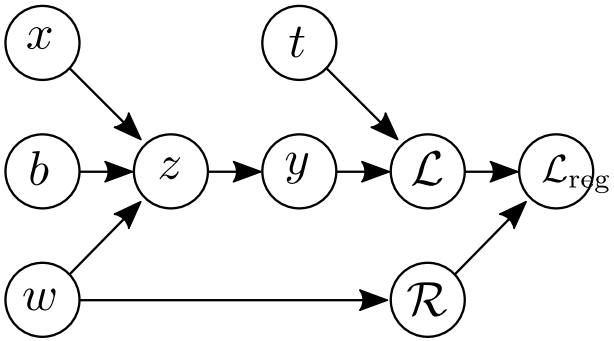
Q: What values to compute? (ie: terms with a bar on top)

Q: Write in computed $\times$ expressions to-evaluate?

$$\overline{\mathcal{L}_{reg}} = 1$$
$$\overline{\mathcal{R}} = \overline{\mathcal{L}_{reg}}\frac{d\mathcal{L}_{reg}}{d\mathcal{R}}$$
$$\overline{\mathcal{L}} = \overline{\mathcal{L}_{reg}}\frac{d\mathcal{L}_{reg}}{d\mathcal{L}}$$
$$\overline{y} = \overline{\mathcal{L}}\frac{d\mathcal{L}}{dy}$$
$$\overline{z} = \overline{y}\frac{dy}{dz}$$
$$\overline{w} = \overline{z}\frac{\partial z}{\partial w} + \overline{\mathcal{R}}\frac{\partial \mathcal{R}}{\partial w}$$
$$\overline{b} = \overline{z}\frac{\partial z}{\partial b}$$

Example: Regularized logistic least squares regression

2.4 in MDL03.4.BackProp-Background-RGrosse.pdf



Forward pass:

$$z = wx + b$$
$$y = \sigma(z)$$
$$\mathcal{L} = \frac{1}{2}(y - t)^2$$
$$\mathcal{R} = \frac{1}{2}w^2$$
$$\mathcal{L}_{reg} = \mathcal{L} + \lambda\mathcal{R}$$

Q: What values to compute? (ie: terms with a bar on top)

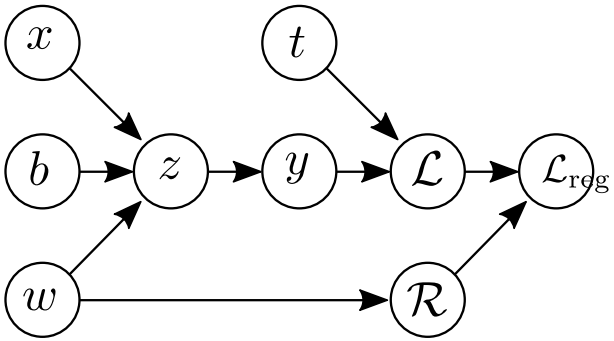
Q: Write in computed $\times$ expressions to-evaluate?

Q: Do the expression evaluation?

$$\overline{\mathcal{L}_{reg}} = 1$$
$$\overline{\mathcal{R}} = \overline{\mathcal{L}_{reg}}\frac{d\mathcal{L}_{reg}}{d\mathcal{R}}$$
$$\overline{\mathcal{L}} = \overline{\mathcal{L}_{reg}}\frac{d\mathcal{L}_{reg}}{d\mathcal{L}}$$
$$\overline{y} = \overline{\mathcal{L}}\frac{d\mathcal{L}}{dy}$$
$$\overline{z} = \overline{y}\frac{dy}{dz}$$
$$\overline{w} = \overline{z}\frac{\partial z}{\partial w} + \overline{\mathcal{R}}\frac{\partial \mathcal{R}}{\partial w}$$
$$\overline{b} = \overline{z}\frac{\partial z}{\partial b}$$

Example: Regularized logistic least squares regression

2.4 in MDL03.4.BackProp-Background-RGrosse.pdf



Forward pass:

$$z = wx + b$$
$$y = \sigma(z)$$
$$\mathcal{L} = \frac{1}{2}(y - t)^2$$
$$\mathcal{R} = \frac{1}{2}w^2$$
$$\mathcal{L}_{reg} = \mathcal{L} + \lambda\mathcal{R}$$

Q: What values to compute? (ie: terms with a bar on top)

Q: Write in computed $\times$ expressions to-evaluate?

Q: Do the expression evaluation?

$$\overline{\mathcal{L}_{reg}} = 1$$
$$\overline{\mathcal{R}} = \overline{\mathcal{L}_{reg}}\frac{d\mathcal{L}_{reg}}{d\mathcal{R}} = \overline{\mathcal{L}_{reg}}\lambda$$
$$\overline{\mathcal{L}} = \overline{\mathcal{L}_{reg}}\frac{d\mathcal{L}_{reg}}{d\mathcal{L}} = \overline{\mathcal{L}_{reg}}$$
$$\overline{y} = \overline{\mathcal{L}}\frac{d\mathcal{L}}{dy} = \overline{\mathcal{L}}(y - t)$$
$$\overline{z} = \overline{y}\frac{dy}{dz} = \overline{y}\sigma'(z)$$
$$\overline{w} = \overline{z}\frac{\partial z}{\partial w} + \overline{\mathcal{R}}\frac{\partial \mathcal{R}}{\partial w} = \overline{z}x + \overline{\mathcal{R}}w$$
$$\overline{b} = \overline{z}\frac{\partial z}{\partial b} = \overline{z}$$

Learning goals

- Explain difference between backpropagation and gradient descent
- Explain about the forward/backward pass
- Apply the calculus chain rule to compute derivatives
- Apply the backpropagation algorithm to efficiently compute derivatives