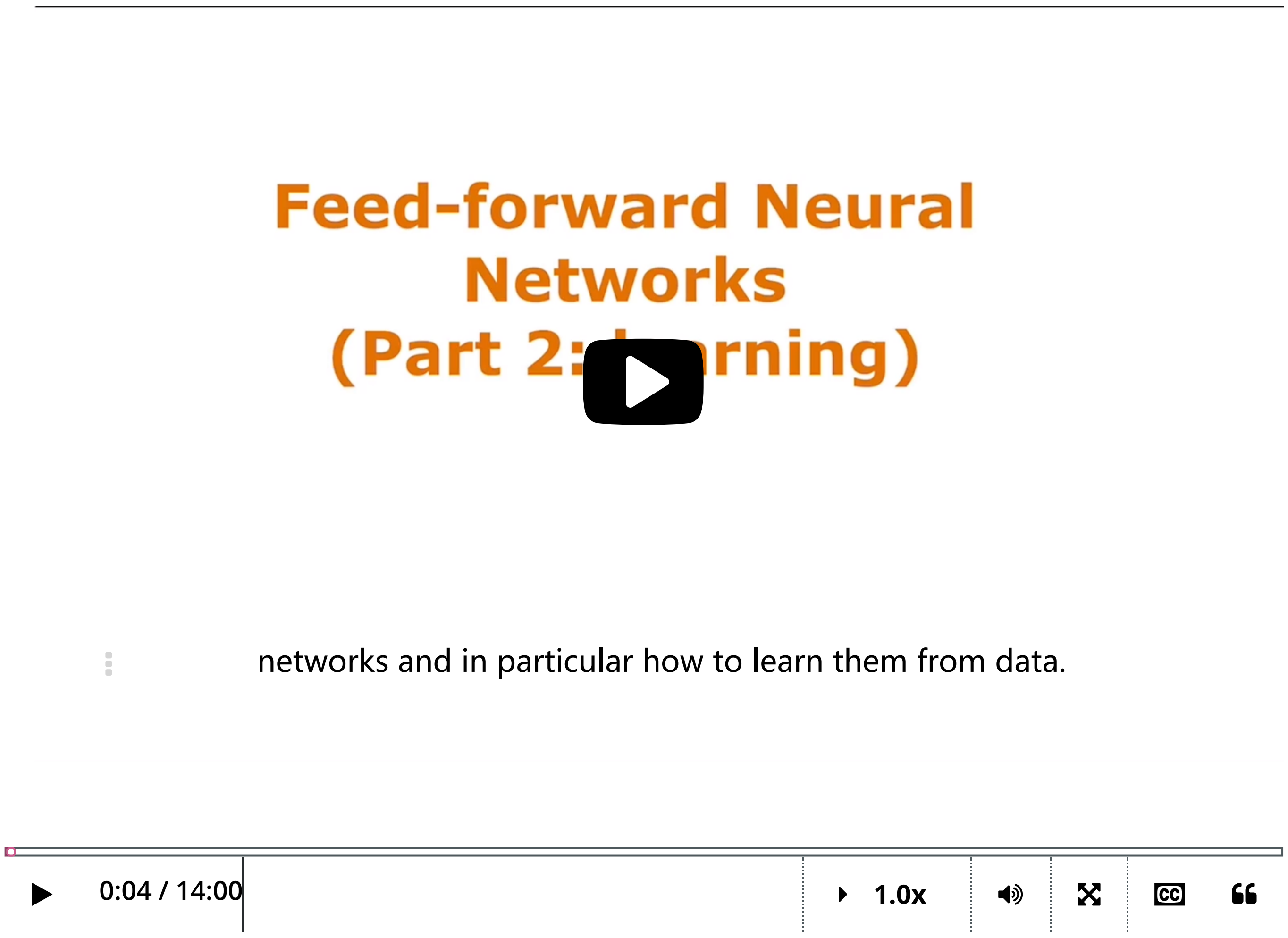
Course › Unit 3 Neural networks (2.5 weeks) › Lecture 9. Feedforward Neural Networks, Back Propagation, and Stochastic Gradient Descent (SGD) › 2. Back-propagation Algorithm

2. Back-propagation Algorithm

Back-propagation Algorithm


Feed-forward Neural Networks (Part 2: Learning)
networks and in particular how to learn them from data.
0:04 / 14:00
1.0x

Welcome back.

Today, we're going to be talking about feed-forward neural

networks and in particular how to learn them from data.

If you recall, feed-forward neural networks

with multiple hidden layers mediating the calculation

from the input to the output are complicated models

that are trying to capture the representation of the examples

towards the output unit in such a way

as to facilitate the actual prediction task.

It is this representation learning part--

Video

Download video file

Transcripts

Download SubRip (.srt) file

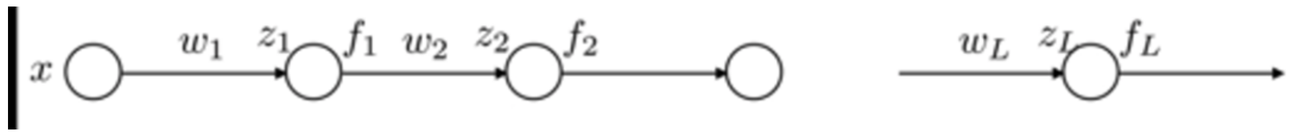
Download Text (.txt) file

Once we set up the architecture of our (feedforward) neural network, our goal will be to find weight parameters that minimize our loss function. We will use the **stochastic gradient descent algorithm** (which you learned in Lecture 4 and revisited in lecture 5) to carry out the optimization.

This involves computing the gradient of the loss function with respect to the weight parameters.

Since the loss function is a long chain of compositions of activation functions with the weight parameters entering at different stages, we will break down the computation of the gradient into different pieces via the chain rule; this way of computing the gradient is called the back-propagation algorithm.

In the following problems, we will explore the main step in the stochastic gradient descent algorithm for training the following simple neural network from the video:



This simple neural network is made up of Lhidden layers, but each layer consists of only one unit, and each unit has activation function f. As usual, x is the input, z_i is the weighted combination of the inputs to the i^{th} hidden layer. In this one-dimensional case, weighted combination reduces to products:

$$\begin{aligned} z_1 &= xw_1 \\ \text{for } i=2\ldots L: \quad z_i &= f_{i-1}w_i \quad \text{where } f_{i-1} = f(z_{i-1}). \end{aligned}$$

We will use the following loss function:

$$\mathcal{L}(y, f_L) = (y - f_L)^2$$

where y is the true value, and f_L is the output of the neural network.

Gradient Descent Update

1/1 point (graded)

Let η be the learning rate for the stochastic gradient descent algorithm.

Recall that our goal is to tune the parameters of the neural network so as to minimize the loss function. Which of the following is the appropriate update rule for the paramter w_1 in the stochastic gradient descent algorithm?

☑ $w_1 \leftarrow w_1 - \eta \cdot \nabla_{w_1} \mathcal{L}(y, f_L)$ ✓

□ $w_1 \leftarrow w_1 + \eta \cdot \nabla_{w_1} \mathcal{L}(y, f_L)$

□ $w_1 \leftarrow \eta \cdot \nabla_{w_1} \mathcal{L}(y, f_L)$

□ $w_1 \leftarrow -\eta \cdot \nabla_{w_1} \mathcal{L}(y, f_L)$

✓

Solution:

The value of a function is non-decreasing in the direction of its gradient from any given point in its domain. Since our goal is to tune the parameters of the neural network so as to minimize the loss, we update the weights in the direction opposite to that of the gradient from any point.

The learning rate η controls the magnitude of the update made during gradient descent.

The final update for any parameter θ with gradient $\nabla_\theta \mathcal{L}$ would be given as follows:

$$\theta \leftarrow \theta - \eta \cdot \nabla_\theta \mathcal{L}.$$

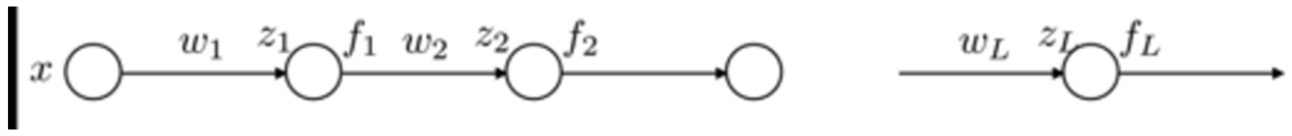
Submit

You have used 1 of 2 attempts

Answers are displayed within the problem

Recursive Expression - Part I

1/1 point (graded)



As above, let $\mathcal{L}(y, f_L)$ denote the loss function as a function of the predictions f_L and the true label y. Recall

$$z_1 \;=\; x w_1 \quad \text{for } i = 2 \ldots L: \quad z_i \;=\; f_{i-1} w_i \quad \text{where } f_{i-1} \;=\; f(z_{i-1}).$$

Let $\delta_i = \dfrac{\partial \mathcal{L}}{\partial z_i}$.

The first step to updating any weight w is to calculate $\frac{\partial \mathcal{L}}{\partial w}$.

Which of the following option(s) is/are correct expression(s) for $\frac{\partial \mathcal{L}}{\partial w_1}$?

(Choose all that apply.)

☐ $\frac{\partial \mathcal{L}}{\partial w_1} = \frac{\partial w_1}{\partial z_1} \cdot \frac{\partial \mathcal{L}}{\partial z_1}$

☑ $\frac{\partial \mathcal{L}}{\partial w_1} = \frac{\partial z_1}{\partial w_1} \cdot \frac{\partial \mathcal{L}}{\partial z_1}$ ✓

☑ $\frac{\partial \mathcal{L}}{\partial w_1} = x \cdot \delta_1$ ✓

☐ $\frac{\partial \mathcal{L}}{\partial w_1} = x + \delta_1$

✓

Solution:

From the chain rule we have:

$$\frac{\partial \mathcal{L}}{\partial w_1} = \frac{\partial z_1}{\partial w_1} \frac{\partial \mathcal{L}}{\partial z_1}.$$

Since $z_1 = w_1 x$, we get

$$\frac{\partial z_1}{\partial w_1} = \frac{\partial \left(w_1 . x\right)}{\partial w_1} = x.$$

Therefore, equivalently,

$$\frac{\partial \mathcal{L}}{\partial w_1} = x\delta_1.$$

Submit You have used 2 of 2 attempts

ℹ Answers are displayed within the problem

Recursive Expression - Part II

1/1 point (graded)

As above, let $\mathcal{L}\left(y, f_L\right)$ denote the loss function as a function of the predictions f_L and the true label y. Let $\delta_i = \frac{\partial \mathcal{L}}{\partial z_i}$.

In this problem, we derive a recurrence relation between δ_i and δ_{i+1}

Assume that f is the hyperbolic tangent function:

$$\begin{aligned} f\left(x\right) &= \tanh\left(x\right) \\ f'\left(x\right) &= \left(1 - \tanh\left(x\right)^2\right). \end{aligned}$$

Which of the following option is the correct expression for δ_1 in terms of δ_2?

● $\delta_1 = (1 - f_1^2) \cdot w_2 \cdot \delta_2$ ✓

○ $\delta_1 = (1 - f_1^2) \cdot w_1 \cdot \delta_2$

○ $\delta_1 = (1 - f_2^2) \cdot w_2 \cdot \delta_2$

○ $\delta_2 = (1 - f_1^2) \cdot w_2 \cdot \delta_1$

Solution:

The chain rule gives

$$\delta_1 = \frac{\partial f_1}{\partial z_1} \cdot \frac{\partial z_2}{\partial f_1} \cdot \frac{\partial \mathcal{L}}{\partial z_2}.$$

Let us examine the first two factors above:

- Since $f_1 = tanh\left(z_1\right)$, we have

$$\frac{\partial f_1}{\partial z_1} = \left(1 - f_1^2\right).$$

- Since $z_2 = w_2 \cdot f_1$, we have

$$\frac{\partial z_2}{\partial f_1} = w_2.$$

Substituting the values of $\frac{\partial f_1}{\partial z_1}, \frac{\partial z_2}{\partial f_1}$ into the main expression for δ_1 we get:

$$\delta_1 = \left(1 - f_1^2\right) \cdot w_2 \cdot \frac{\partial \mathcal{L}}{\partial z_2} = \left(1 - f_1^2\right) \cdot w_2 \cdot \delta_2.$$

Submit You have used 2 of 2 attempts

Answers are displayed within the problem

Final Expression of the Gradient

1/1 point (graded)

As above, let $\mathcal{L}\left(y, f_L\right)$ denote the loss function as a function of the predictions f_L and the true label y. Let $\delta_i = \dfrac{\partial \mathcal{L}}{\partial z_i}$.

In this problem, we unroll the recurrence expression for $\frac{\partial \mathcal{L}}{\partial w_1}$. We will use the loss function

$$\mathcal{L}\left(y, f_L\right) = \left(y - f_L\right)^2.$$

Compute $\dfrac{\partial \mathcal{L}}{\partial w_1}$ and select the correct option from below.

○ $\frac{\partial \mathcal{L}}{\partial w_1} = x\left(1 - f_2^2\right) \cdots \left(1 - f_L^2\right) w_2 w_3 \cdots w_L . \left(2\left(f_L - y\right)\right)$

○ $\frac{\partial \mathcal{L}}{\partial w_1} = x\left(1 - f_1^2\right)\left(1 - f_2^2\right) \cdots \left(1 - f_L^2\right) w_1 w_2 w_3 \cdots w_L \left(2\left(f_L - y\right)\right)$

● $\frac{\partial \mathcal{L}}{\partial w_1} = x\left(1 - f_1^2\right)\left(1 - f_2^2\right) \cdots \left(1 - f_L^2\right) w_2 w_3 \cdots w_L \left(2\left(f_L - y\right)\right)$ ✓

○ $\frac{\partial \mathcal{L}}{\partial w_1} = x w_2 w_3 \cdots w_L \left(2\left(f_L - y\right)\right)$

Solution:

From the previous problem, we know the following:

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial w_1} &= x \cdot \delta_1 \\ \delta_1 &= (1 - f_1^2) \cdot w_2 \cdot \delta_2.\end{aligned}$$

Similarly, $\delta_2, \delta_3 \ldots \delta_L$ can be given as follows:

$$\begin{aligned}\delta_2 &= (1 - f_2^2) \cdot w_3 \cdot \delta_3 \\ \delta_3 &= (1 - f_3^2) \cdot w_4 \cdot \delta_4 \\ &\vdots \\ \delta_{L-1} &= (1 - f_{L-1}^2) \cdot w_L \cdot \delta_L \\ \delta_L &= \frac{\partial \mathcal{L}}{\partial z_L} \\ \delta_L &= \frac{\partial \mathcal{L}}{\partial f_L} \cdot \frac{\partial f_L}{\partial z_L} \\ \delta_L &= \frac{\partial (f_L - y)^2}{\partial f_L} \frac{\partial f_L}{\partial z_L} \\ \delta_L &= 2\,(f_L - y)\,\frac{\partial f_L}{\partial z_L} \\ \delta_L &= 2\,(f_L - y)\,(1 - f_L^2)\,.\end{aligned}$$

Plugging the above equations into the expression for $\frac{\partial \mathcal{L}}{\partial w_1}$ we get:

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial w_1} &= x \cdot \delta_1 \\ \frac{\partial \mathcal{L}}{\partial w_1} &= x \cdot (1 - f_1^2) \cdot w_2 \cdot \delta_2 \\ \frac{\partial \mathcal{L}}{\partial w_1} &= x \cdot (1 - f_1^2) \cdot w_2 \cdot (1 - f_2^2) \cdot w_3 \cdot \delta_3 \\ &\vdots \\ \frac{\partial \mathcal{L}}{\partial w_1} &= x\,(1 - f_1^2)\,(1 - f_2^2) \cdots (1 - f_L^2)\,w_2 w_3 \cdots w_L\,(2\,(f_L - y))\,.\end{aligned}$$

Submit   You have used 2 of 2 attempts

Answers are displayed within the problem

Discussion

Show Discussion

Topic: Unit 3 Neural networks (2.5 weeks):Lecture 9. Feedforward Neural Networks, Back Propagation, and Stochastic Gradient Descent (SGD) / 2. Back-propagation Algorithm