

✓

1 / 1 points

1.

Suppose w is the weight on some connection in a neural network. The network is trained using gradient descent until the learning

converges. However, the dataset consists of two mini-batches, which differ from each other somewhat. As usual, we alternate between the mini-batches for our gradient calculations, and that has implications for what happens

after convergence. We plot the change of w as training progresses. Which of the following scenarios shows that convergence has occurred? **Noticethat we're plotting the change in w , as opposed to w itself.**

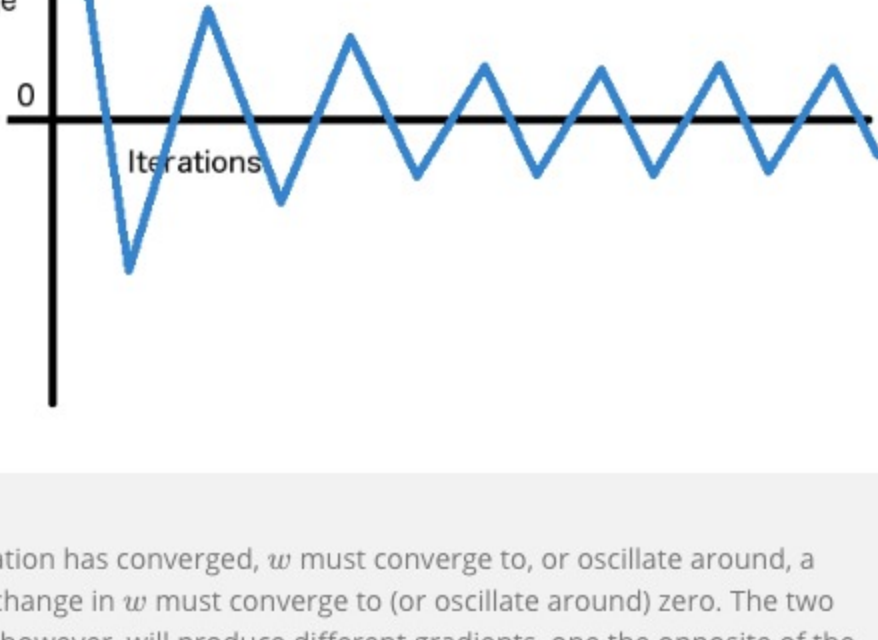
Note that in the plots below, each *iteration* refers to a single *step* of steepest descent on a *single minibatch*.

1 / 1 points

Change in w

0

Iterations




Correct

If the optimization has converged, w must converge to, or oscillate around, a point. So the change in w must converge to (or oscillate around) zero. The two mini-batches, however, will produce different gradients, one the opposite of the other. That causes w to go back and forth.

Change in w

0


Iterations



Change in w

0

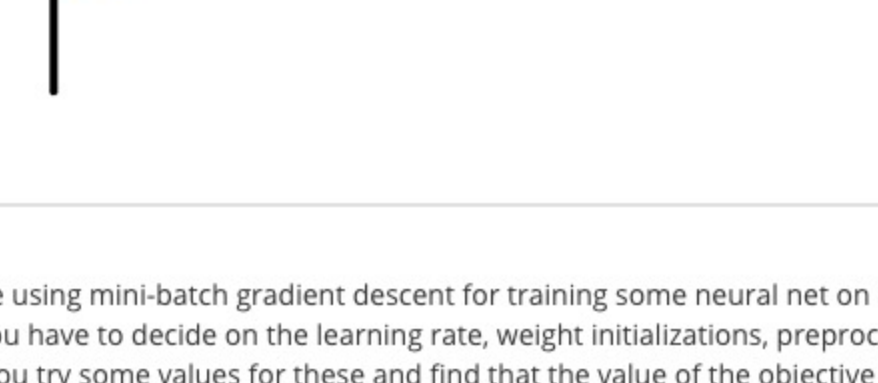
Iterations



Change in w

0

Iterations



✓

1 / 1 points

2.

Suppose you are using mini-batch gradient descent for training some neural net on a large dataset. You have to decide on the learning rate, weight initializations, preprocess the inputs etc. You try some values for these and find that the value of the objective function on the training set decreases smoothly but very slowly. What could be causing this? Check all that apply.

The dataset is too small.

Un-selected is correct

The learning rate may be too small.

Correct

A small learning rate leads to small changes in the parameters, and to slow convergence.

The weights might have been initialized to very large values (hint: think of what this would do to the logistic hidden units).

Correct

Large values of weights may saturate the hidden units. Their derivatives would become small (be on a "plateau") and learning would get slowed down.

The inputs might have a very large scale (hint: think of what this would do to the logistic hidden units).

Correct

Large values of inputs may saturate the hidden units. Their derivatives would become small (be on a "plateau") and learning would get slowed down.

✓

1 / 1 points

3.

Full-batch gradient descent can be used to minimize an objective function if the dataset is not too large. Which statement regarding full-batch gradient descent is **false**?

For some setting of the learning rate, it is possible that the objective function increases in some iteration.

There is always a learning rate schedule such that the objective function monotonically decreases.

For every **fixed** learning rate, the objective function will monotonically decrease.

Correct

If the learning rate is high, the weights get updated too much in the direction of steepest descent and "climb up the ravine".

Adaptive learning rate methods perform well for full-batch (or large mini-batch) gradient descent.

✓

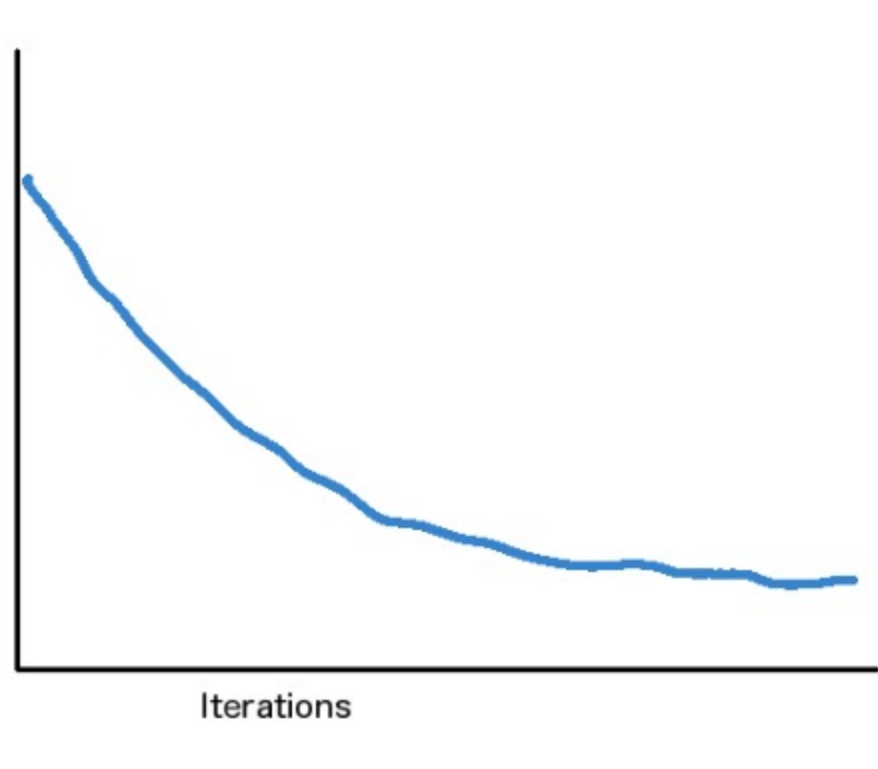
1 / 1 points

4.

Claire is training a neural net using mini-batch gradient descent. She chose a particular learning rate and found that the training error decreased as more iterations of training were performed, as shown here in blue:

Training Error

Iterations

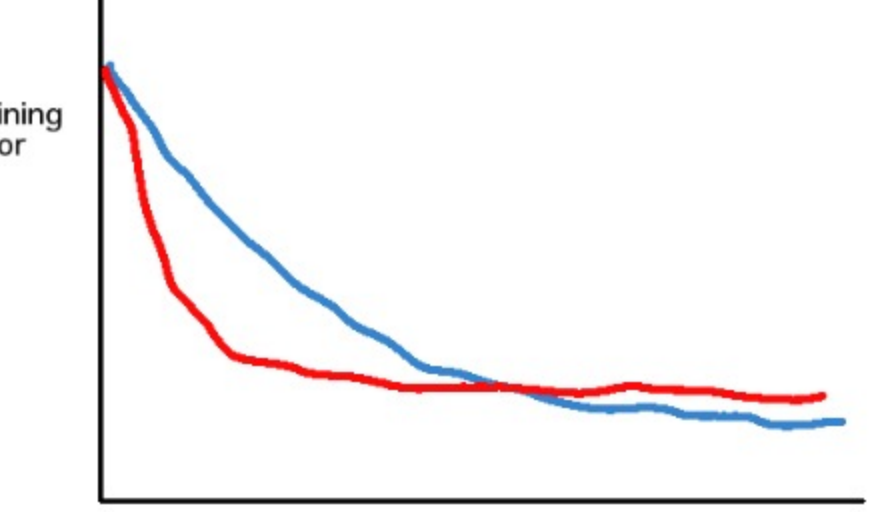


She was not sure if this was the best she could do. So she tried a **bigger** learning rate. Which of the following error curves (shown in red) might she observe now? Select the two most likely plots.

Note that in the plots below, each *iteration* refers to a single *step* of steepest descent on a *single minibatch*.

Training Error

Iterations

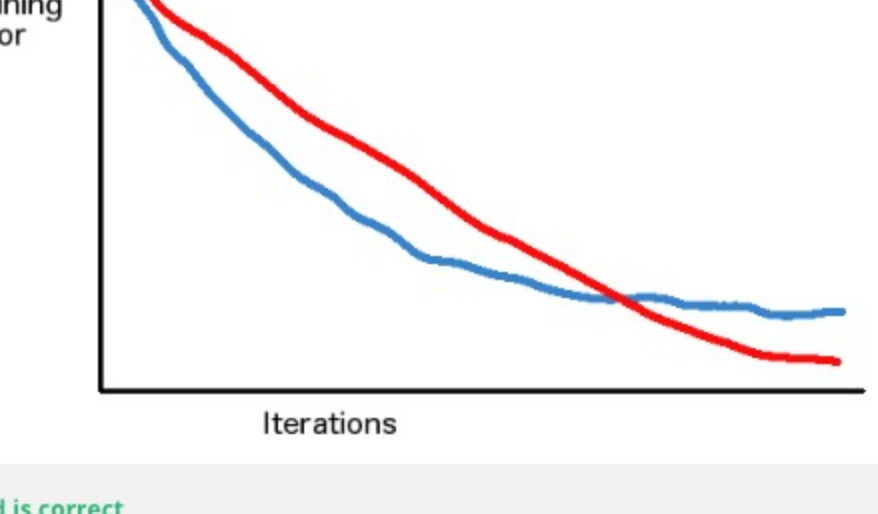


Correct

Large learning rate may show fast progress initially but lead to worse final performance.

Training Error


Iterations



Un-selected is correct

Training Error

Iterations

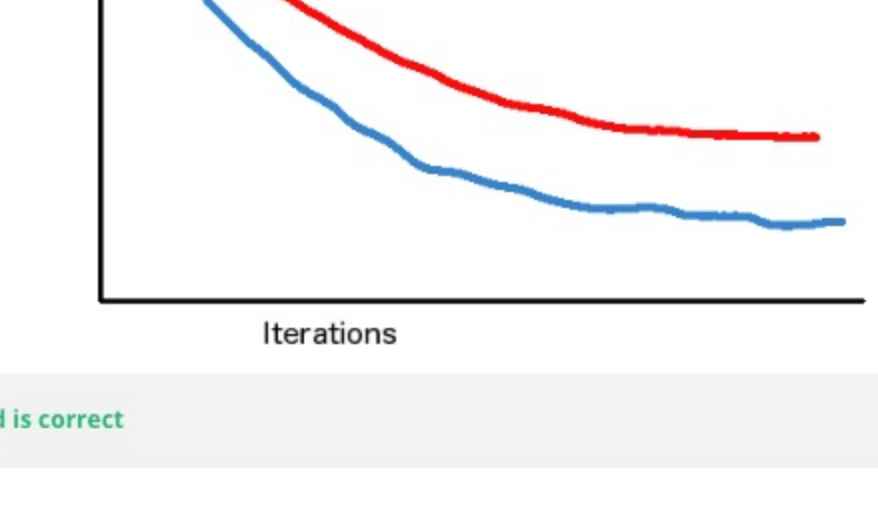


Correct

Large learning rate may show better final performance.

Training Error

Iterations



Un-selected is correct

✓

1 / 1 points

5.

In the lectures, we discussed two kinds of gradient descent algorithms: mini-batch and full-batch. For which of the following problems is mini-batch gradient descent likely to be **a lot better** than full-batch gradient descent?

Sentiment Analysis: Decide whether a given movie review says that the movie is 'good' or 'bad'. The input consists of the word count in the review, for each of 50,000 words. The training set consists of 100 movie reviews written by experts for a newspaper.

Un-selected is correct

Sentiment Analysis: Decide whether a given movie review says that the movie is 'good' or 'bad'. The input consists of the word count in the review, for each of 50,000 words. The training set consists of 1,000,000 movie reviews found on the internet.

Correct

Predict if an experiment at the Large Hadron Collider is going to yield positive results. The input consists of 25 experiment parameters (energy level, types of particles, etc). The training set consists of the 200 experiments that have already been completed (some of those yielded positive results; some yielded only negative results).

Un-selected is correct

Object detection: Identify which of 1000 categories an object image belongs to, given 10 million 256 X 256 pixel images.

Correct