

Stack Overflow is a question and answer site for professional and enthusiast programmers. It's 100% free, no registration required.

Take the 2-minute tour x

Neural Networks: Online learning (one-example-at-a-time)

CAREERS 2.0
by stackoverflow



+



Have projects on Codeplex?
Import them easily to your profile

I have a neural network that performs a classification task, and it works fairly well when the training set is large enough.

However, I'm looking for a way to train the NN with [one labelled example at a time](#).

That is, I intercept data, one example at a time, and I need to update my NN weights. I'm not allowed to store the data for future, batch training purposes.

I have a feed-forward NN built(following Stanford's ML course on Coursera) in octave. I can run back-prop on the NN using each new example, but that approach unreliably converges, and takes a significant amount of time.

Are there any other more efficient algorithms for online learning out there in the context of Neural Networks?

I noticed Matlab's [adapt\(\)](#) function seems to do just this. The documentation doesn't specify what it's doing behind the scene. Is it just one iteration of back-prop?

[machine-learning](#) [neural-network](#) [backpropagation](#) [neural-network-tuning](#)

edited Nov 4 '13 at 12:28

asked Oct 28 '13 at 17:38



user8472
49 6

- 1 What exactly didn't work when you ran backprop using one example at a time? That method is called [Stochastic Gradient Descent](#) and generally works well as long as you've set the learning rate, momentum, etc. to "good" values. However, I've seen it work poorly for some problems like XOR. Can you provide more detail in your question? – [Imjohns3](#) Oct 28 '13 at 19:47

Sorry about that, you're right. Stochastic back-prop works fine. It was an implementation bug that got to me. I've modified the question to find other algorithms to do the same. – [user8472](#) Nov 4 '13 at 12:24

[add comment](#)

Know someone who can answer? Share a link to this [question](#) via [email](#), [Google+](#), [Twitter](#), or [Facebook](#).

Browse other questions tagged [machine-learning](#) [neural-network](#) [backpropagation](#) [neural-network-tuning](#) or [ask your own question](#).