

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

How Sensitive Are FF Neural Networks?

HANDS-ON CUDA[®] TRAINING IN THE CLOUD

OPENACC · CUDA C/C++ · CUDA FORTRAN · CUDA PYTHON · GPU-ACCELERATED LIBRARIES [LEARN MORE >>](#)

CrossPost: <https://stats.stackexchange.com/questions/103960/how-sensitive-are-neural-networks>

I am aware of pruning, and am not sure if it removes the actual neuron or makes its weight zero, but I am asking this question as if a pruning process were not being used.

On variously sized feedforward neural networks on large datasets with lots of noise:

1. Is it possible one (or some trivial amount) extra OR missing hidden neurons OR hidden layers make or break a network? Or will its synapse weights simply degrade to zero if it is not necessary and compensate with the other neurons if it is missing one or two?
2. When experimenting, should input neurons be added one at a time or in groups of X? What is X? Increments of 5?
3. Lastly, should each hidden layer contain the same number of neurons? This is usually what I see in example. If not, how and why would you adjust their sizes if not relying on using pure experimentation?

I would prefer to overdo it and wait longer for a convergence than if larger networks will adapt itself to the solution. I have tried numerous configurations, but it is still difficult to gauge an optimum one.

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

edited Jun 19 at 8:21

asked Jun 19 at 7:53



SilverFox
30 7

[add comment](#)

1 Answer

1) Yes, absolutely. For example, if you have too less neurons in your hidden layer your model will be too simple and have high bias. Similarly, if you have too many neurons your model will overfit and have high variance. Adding more hidden layers allows you to model very complex problems like object recognition but there are a lot of tricks to make adding more hidden layers work; this is known as the field of deep learning.

2) In a single layered neural network its generally a rule of thumb to start with 2 times as many neurons as the number of inputs. You can determine the increment through binary search; i.e. run through a few different architectures and see how the accuracy changes..

3) No, definitely not - each hidden layer can contain as many neurons as you want it to contain. There is no way other can experimentation to determine their sizes; all of what you mention are hyperparameters which you must tune.

Im not sure if you are looking for a simple answer, but maybe you will be interested in a new neural network regularization technique called dropout. Dropout basically randomly "removes" some of the neurons during training forcing each of the neurons to be good feature detectors. It greatly prevents overfitting and you can go ahead and set the number of neurons to be high without worrying too much. Check this paper out for more info: http://www.cs.toronto.edu/~nitish/msc_thesis.pdf

answered Jun 19 at 20:30



rahulm
46 4

I should be using something automated like dropout... But I was really asking because I was varying my neuron by increments of ten, twenty, fifty, when maybe I should have been varying in increments of one? –

[SilverFox](#) Jun 21 at 7:43

[add comment](#)

Not the answer you're looking for? Browse other questions tagged [machine-learning](#) [neural-network](#) or [ask your own question](#).