# MNIST digit recognition Neural Net

Ahmed Gado

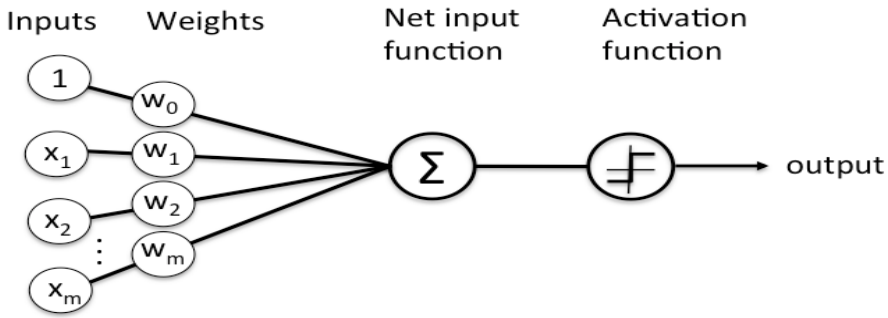
May 11, 2017

**Abstract**:
Neural nets are the most recent breakthrough in Artificial Intelligence. By mimicking the basic structure of an animal nervous system, artificial neural nets can finally ”learn” and achieve many great feats. This project implements and trains a neural net using 60,000 images from MNIST data set. Using Stochastic Gradient Descent Learning and calculus approaches, the final neural net managed to recognize handwritten digits by an accuracy of 82%

Tufts

# Background

- NNs can work with a variety of inputs by having many (and many) layers
- Each layer applies a calculation to the preceding layer.
- The goal of learning is to “tweak” this calculation in order to improve the output




Tufts

# Technical Approach – Keep Trying

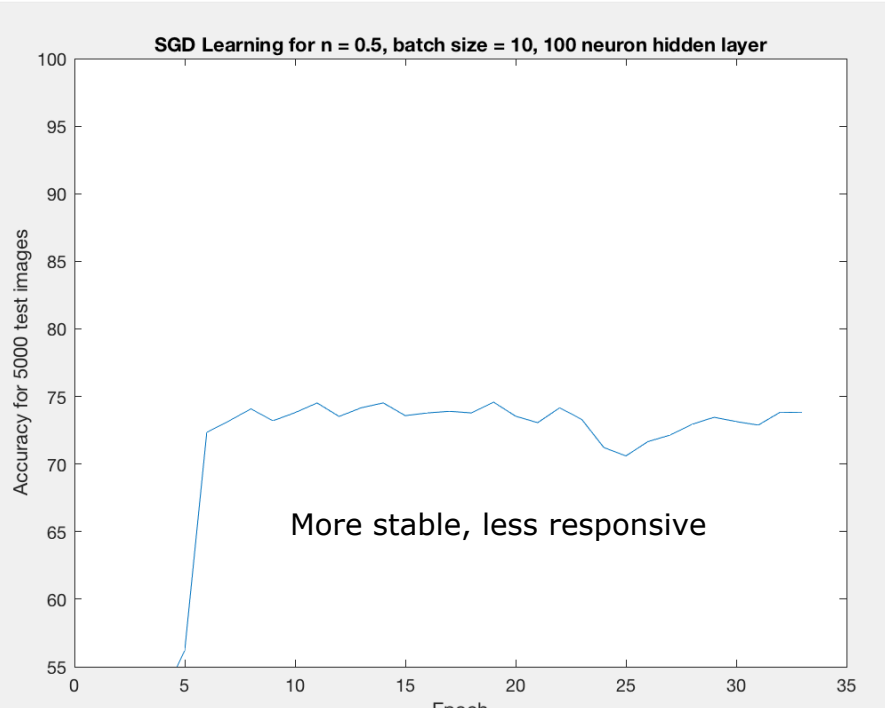
- Tweak every single parameter independently and see if it improves the output. If it does, make the change permanent. If not, try again.
- It worked! But it was too slow. A better way had to be found.
- Very dumb. Took 10 seconds to run through thousands of parameters for one single image.

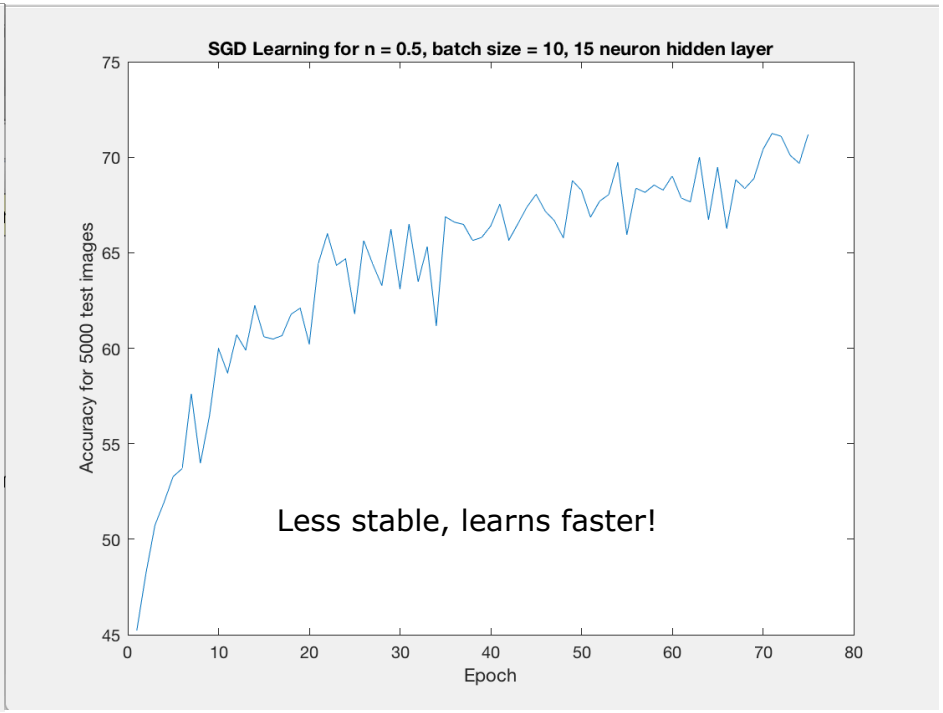
# Technical Approach – Stochastic Gradient Descent

- Uses Calculus (partial derivatives) to implement a more informed, more efficient learning method. (called backpropagation)
- It learns very fast! 2 seconds to run through 60,000 images.

# Results | Stochastic Gradient Descent

- Learning curve has different behavior depending on the NN configuration (hyperparameres)




Tufts

# Conclusions and Next Steps

- By pure experimentation, the best configuration I reached is:
  - 30 neuron hidden layer, 10 images batch size
  - Learning rate = 0.5
- Reaches 82% accuracy! On 5000 test images
- Next steps: implement better parameter initializing methods, more regularization techniques (ex: dropout), more complex NNs (convolutional neural nets)

# References

These Links were very helpful for understanding how NNs work. However, I strictly did not look at any line of code (from any programming language).


1. http://neuralnetworksanddeeplearning.com/
2. https://mattmazur.com/2015/03/17/a-step-by-step-backpropagation-example/