Intro to Deep Learning with Keras

Nathan Danneman

January 18, 2020

Acknowledgements

Big thanks to:

- Data Machines (www.datamachines.io)

Outline

1. Just enough theory/background
2. Feed forward neural networks
3. Image classification with convolutional neural networks
4. Text classification with sequence models

About This Class

Goals:

- Give you nodding familiarity with theory and practice
- Give you ample pointers to dive deeper on your own
- Leave you a starter code-base

This class lightly assumes:

- you have some familiarity with Python
- knowledge of some general data science concepts

Models Are Function Approximators

Image here with various functions to approximate a...function
Most estimation is loss minimization.

Neurons

Little sigmoid functions... Equation Image

The Feed Forward Neural Network

Input is a vector...ex. Every element goes into a neuron...
And the outputs of those neurons go to every neuron in the next layer
And then at the end you have a (possibly special) output layer that maps the outputs of the preceding layer to your target values.
Image of layers...

Estimation

Stochastic gradient descent

Closed form

What is Keras?

Light wrapper over TensorFlow (or the other one).
(website) APIs allow simplicity and still power.
Great for *using* neural networks, even very complex ones.
Terrible for research *on* neural networks.

Setup

pip install keras – it brings along (stuff you'll likely need) (that was easy)
Setup with/for GPU-based estimation: How? Why?

Code Example: Iris Classification

The iris data: table... See if we can classify type.

Hyperparameters and Overfitting

Hyperparameters: Number of layers Number of neurons in each layer

Strategies to avoid overfitting: Premature halting Dropout Regularization

When Would You Use This?

Um...

Image Classification with Neural Networks

Images are arrays (Height x Width x Depth) You could unwrap one, and then use a fully connected feed forward network... But: Tons of parameters Portions of images might not be in sample place

CNNs

Idea: Estimation: Find some little 3x3xDepth mini-pictures that express the ideas in all your images Use the same set of mini-pictures everywhere (parameter sharing) Inference: Figure out which mini-picture best approximates the neighborhood of each pixel. Capture that information, and send it to a dense layer.

CNNs – a bit further

Idea: Estimation: Find some little 3x3xDepth mini-pictures that express the ideas in all your images Use the same set of mini-pictures everywhere (parameter sharing)
At every OTHER pixel, record the maximum activation in the neighborhood. Drop the rest. Then, find ANOTHER set of mini-pictures (now each one represents a 6x6 area) ... repeat. Hierarchy! (strained analogy to visual cortex – useless).

CNNs: Visualized

pointer to some website

Are Your Images Special?

Chances are, your images share low-level features with others. So, get a pre-trained model for everyday objects, cut some layers off, and train your own. This is called ”fine tuning.” Unless your

images, at a very low level, are really atypical.

Let's Code One Together!

Dog image vs Cat image

Language Modeling as Sequence Modeling

Most language is treated as sequences of tokens. The tokens are typically lowercase, lemmatized words. Though some models include much more (e.g. Bert).

What's a Sequence Model?

Try to predict the next token from the previous x tokens (i.e. Markov Model). Try to come up with an embedding that represents a sequence or its type (i.e. hidden Markov Model).

Long Short-term Memory Models (LSTM)

how these work.
some details...er...

A Fun Example:

Who said it: XX or YY.

Questions?

nathandanneman [at] datamachines [dot] io