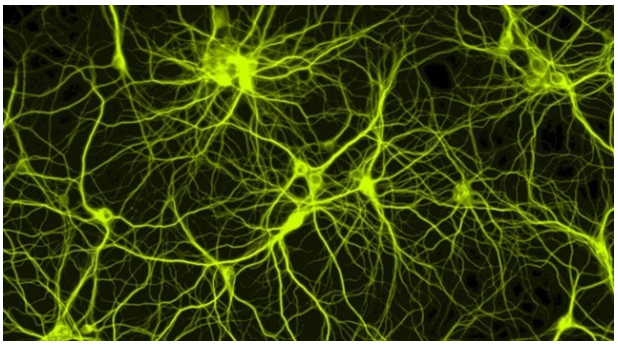
How to build a simple neural network in 9 lines of Python code

Milo Spencer-Harper
Jul 21, 2015 · 6 min read



As part of my quest to learn about AI, I set myself the goal of building a simple neural network in Python. To ensure I truly understand it, I had to build it from scratch without using a neural network library. Thanks to an excellent blog post by Andrew Trask I achieved my goal. Here it is in just 9 lines of code:

```
from numpy import exp, array, random, dot
training_set_inputs = array([[0, 0, 1], [1, 1, 1], [1, 0, 1], [0, 1, 1]])
training_set_outputs = array([[0, 1, 1, 0]]).T
random.seed(1)
synaptic_weights = 2 * random.random((3, 1)) - 1
for iteration in xrange(10000):
    output = 1 / (1 + exp(-(dot(training_set_inputs, synaptic_weights))))
    synaptic_weights += dot(training_set_inputs.T, (training_set_outputs - output) * output * (1
print 1 / (1 + exp(-(dot(array([1, 0, 0]), synaptic_weights))))
```

short_version.py hosted with ♡ by GitHub view raw

In this blog post, I'll explain how I did it, so you can build your own. I'll also provide a longer, but more beautiful version of the source code. And I've created a video version of this blog post as well.

But first, what is a neural network? The human brain consists of 100 billion cells called neurons, connected together by synapses. If sufficient synaptic inputs to a neuron fire, that neuron will also fire. We call this process "thinking".