一、简介

* 神经网络~~瞎扯的~~概述

简要的说，什么是神经网络？其实就是一个很复杂的函数，给一个输入（一般都是一组数）能有一个输出（比如知道是什么类别）。神经网络的魅力在于，它可以自己把这个“函数”学习出来。在“监督学习”中，你可以“喂”一些已经标记好的数据（比如一堆把人脸部分框起来的图片），然后它自己学习明白了，就可以去识别新的图片（从新给的图片中识别出人脸）。

而对于神经网络的基本结构，概要的讲，分为输入层、若干个隐含层、输出层。层与层之间由一些特定参数相联系，其中大多类似于：

X → Y=WTX+b → Z=g(Y)

每一层的输入都通过一个线性的函数（其中有参数w1,w2,w3…和b）和一个激活函数(sigmod, ReLU 等等)，这就是“正向传播”；然后神奇的是，你可以从这个网络的终点出发，利用梯度下降法倒着优化你的参数W和b，以便找到最优的解决方案来适应你给出的训练集（比如前面提到的人脸图片）。

我这么讲你大概不知所云，所以我想给你一些可视化的材料：

深度学习系列视频 3Blue1Brown (Grant Sanderson)

<https://www.bilibili.com/video/av15532370/>

<https://www.bilibili.com/video/av16144388/>

<https://www.bilibili.com/video/av16577449/>

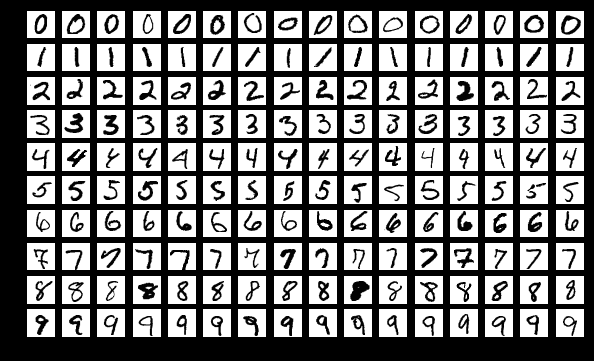
3Blue1Brown的动画做的非常美妙，你可以通过这三个视频直观而清晰的了解什么是神经网络，认真看一定会茅塞顿开。

不过如果你想学的专业一些，请务必去看吴恩达先生的课程，那里会讲的非常详细（最好你的线性代数学的不差……）

吴恩达 Andrew Ng #deeplearniNgAI

<https://mooc.study.163.com/smartSpec/detail/1001319001.htm/?utm_source=weibo.com&utm_medium=timeline&utm_campaign=deepLearning&utm_content=wnd20170831>

* MNIST手写识别数据集和我们HelloWorld的目标



基本上来说，对于神经网络这种需要大量数据的东西，如果在入手的时候刚好有一个经典的数据集，那真是再好不过了。幸运的是，MNIST刚好可以满足初学者的愿望——简单而又实用。这些手写图片都是已经被标记好了的（你可以理解为他们的文件名都是0~9的数字）。一个image对应一个label，images是28\*28像素的文件，可以展开为一个的列向量（在输入层我们一般都这样进行“向量化”，为了方便以后就表示为(784,1)啦）；labels是从0到9的十类，表示为一个(10,1)的向量，每一个都代表成为此类的概率（用于分类的神经网络的输出层一般都这样），而识别的时候，输出的：

就判断为数字8，诸如此类吧。

二、实现

* 基本安装步骤以及算法流程

1. 安装Python环境

这里可以用python 3.5或者3.6都可以，但是我更喜欢用Anaconda，因为里面下载各种库和IDE都比较方便。

1. 安装TensorFlow

这里参考下面给出的网址（也就是官网的安装教程）就可以，启动命令行（Shell或者Anaconda Prompt），输入就可以自动下载安装啦。

不过这玩意是有CPU和GPU版本的区分的，你要是想安装GPU还得进行下面这一步……

1. 安装GPU的各种软件(你得是英伟达NVIDIA的显卡哦)

需要安装的基本是CUDA® Toolkit 9.0.和cuDNN v7.0.，其中第二个还得注册个英伟达的账户，下载网站的话……你可以搜，不过没安这俩执行程序会报错，报错里面就有下载网站。是不是觉得GPU的好麻烦……但是确实比CPU的快百倍啊！

基础Python环境 (Anaconda)

<https://www.continuum.io/downloads>

TensorFlow安装教程 (For Windows)

<https://www.tensorflow.org/install/install_windows>

安装工作大功告成之后，不得不提提我们整个神经网络的架构过程了。不管你是多难或者多简单的，基本都包含在这四步里面了：

1. 定义算法公式（计算图），也就是正向的公式
2. 定义损失函数（loss），选定优化器，并让优化器优化loss
3. 迭代地对数据进行训练（运用For循环）
4. 在测试集或验证集上对准确率进行评测

* 单层实现 (Softmax Regression)

………………代码……………………

* 双层(包含隐含层)实现 (ReLU + Softmax Regression)

………………代码……………………

* 卷积实现 (CNN)

………………代码……………………

代码参考：

《TensorFlow实战》黄文坚、唐源

三、总结

现在这个时代人工智能的奇点即将临近，机遇与挑战并存。各大媒体都在炒热深度学习的时候，我们应该冷静下来，静心思考这玩意到底是什么，这才是更有价值的东西。神经网络并不是什么遥不可及的东西，只要你肯写代码就行。

实现神经网络不止于用TensorFlow，这类都是更新换代很快的，没准下一个半年热点和焦点又会改变；给你推荐Caffe，是很容易上手的深度学习架构；还有你们肯定会接触的MATLAB（在数学实验里），那里面有很多关于深度学习的新玩意可以玩（尤其是R2017b），甚至几行代码就可以体会到神经网络的快乐。

Caffe官网

<http://caffe.berkeleyvision.org/>

试用MATLAB的深度学习组件

<https://cn.mathworks.com/solutions/deep-learning.html>