**CNN 卷积神经网络 2**

作者: **年拾柒**编辑: **莫烦 2016-11-03**

学习资料:

* [相关代码](https://github.com/MorvanZhou/tutorials/tree/master/tensorflowTUT/tf18_CNN2)
* 为 TF 2017 打造的[新版可视化教学代码](https://github.com/MorvanZhou/Tensorflow-Tutorial)
* 机器学习-简介系列 [什么是 CNN](https://morvanzhou.github.io/tutorials/machine-learning/ML-intro/2-2-CNN/)

这一次我们会说道 CNN 代码中怎么定义 Convolutional 的层和怎样进行 pooling.

基于上一次卷积神经网络的介绍，我们在代码中实现一个基于MNIST数据集的例子

[**定义卷积层的 weight bias**](https://morvanzhou.github.io/tutorials/machine-learning/tensorflow/5-04-CNN2/#%E5%AE%9A%E4%B9%89%E5%8D%B7%E7%A7%AF%E5%B1%82%E7%9A%84-weight-bias)

首先我们导入

**import** tensorflow as tf

采用的数据集依然是tensorflow里面的mnist数据集

我们需要先导入它

python **from** tensorflow.examples.tutorials.mnist **import** input\_data

本次课程代码用到的数据集就是来自于它

mnist=input\_data.read\_data\_sets('MNIST\_data',one\_hot=true)

接着呢，我们定义Weight变量，输入shape，返回变量的参数。其中我们使用tf.truncted\_normal产生随机变量来进行初始化:

def weight\_variable(shape):

inital=tf.truncted\_normal(shape,stddev=0.1)

return tf.Variable(initial)

同样的定义biase变量，输入shape ,返回变量的一些参数。其中我们使用tf.constant常量函数来进行初始化:

def bias\_variable(shape):

initial=tf.constant(0.1,shape=shape)

return tf.Variable(initial)

定义卷积，tf.nn.conv2d函数是tensoflow里面的二维的卷积函数，x是图片的所有参数，W是此卷积层的权重，然后定义步长strides=[1,1,1,1]值，strides[0]和strides[3]的两个1是默认值，中间两个1代表padding时在x方向运动一步，y方向运动一步，padding采用的方式是SAME。

def conv2d(x,W):

return tf.nn.conv2d(x,W,strides=[1,1,1,1]，padding='SAME')

[**定义 pooling**](https://morvanzhou.github.io/tutorials/machine-learning/tensorflow/5-04-CNN2/#%E5%AE%9A%E4%B9%89-pooling)

接着定义池化pooling，为了得到更多的图片信息，padding时我们选的是一次一步，也就是strides[1]=strides[2]=1，这样得到的图片尺寸没有变化，而我们希望压缩一下图片也就是参数能少一些从而减小系统的复杂度，因此我们采用pooling来稀疏化参数，也就是卷积神经网络中所谓的下采样层。pooling 有两种，一种是最大值池化，一种是平均值池化，本例采用的是最大值池化tf.max\_pool()。池化的核函数大小为2x2，因此ksize=[1,2,2,1]，步长为2，因此strides=[1,2,2,1]:

def max\_poo\_2x2(x):

return tf.nn.max\_pool(x,ksize=[1,2,2,1],strides=[1,2,2,1])

好啦，如果你对本节课内容已经了解，下一次课我们将构建卷积神经网络的架构~