# Deep-Learning实验报告

## （反向传播实现MNIST）实验

日期：2017.6.29

学号：24320142202455

姓名：林金鹏

## 实验目标：

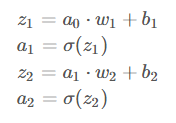
了解反向传播机制

## 实验内容：

利用back propagation 反向传播实现MNIST数据集识别

## 实验相关知识：

* 前向传播：



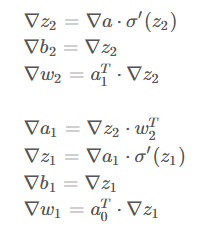
A0 是输入，w1是第一层权值，b1是第一层偏置值，激活函数为sigmoid 函数

W2 是第二层权值，b2 是第二层偏置值，激活函数为 sigmoid 函数

前向传播就是通过将输入加权求和，然后激活得到结果的过程。

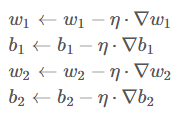
* 反向传播中的反向求导：

反向传播中最关键的是反向求导，根据前向传播的公式进行反向求导。求导的流程如图所示：



目的是要求出 w1,b1,w2,b2 的delta 值，delta值即为变量的偏导数。

* 反向传播中的更新参数：

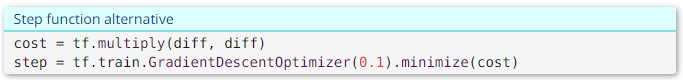


根据反向求导可以求出各个权值和偏置值的delta值，这些值作为参数变化率，乘以步长（学习率）便得到参数将要改变的幅度。

然后根据改变的幅度依次改变参数的值。

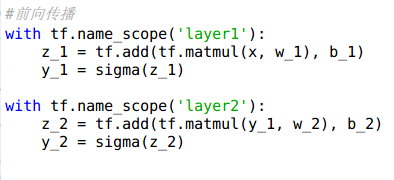
* 反向传播中的更新参数：

在tensorflow 可以方便的使用两个函数实现反向传播，通过计算图，tensorflow可以自动求导，并自动更新权值。

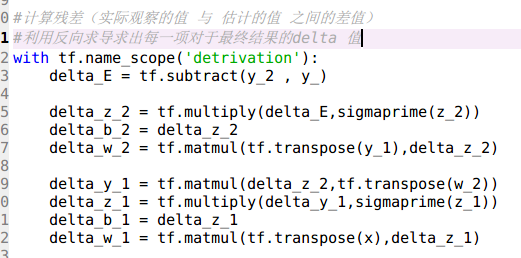


## 关键代码：

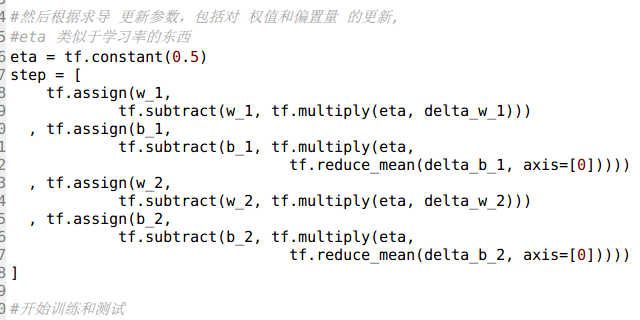
前向传播



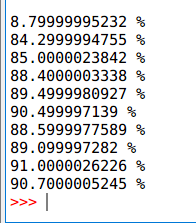
反向求导



更新网络

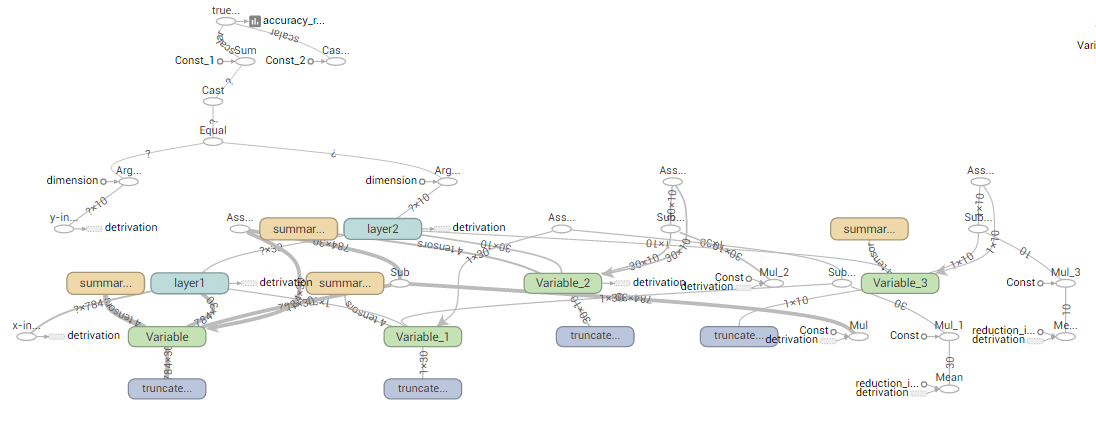


## 实验结果分析：

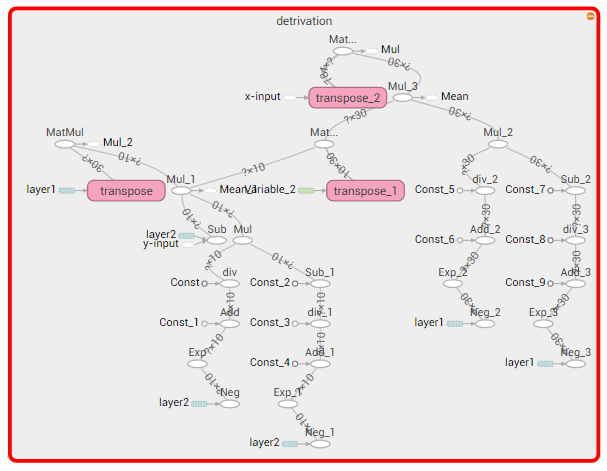


利用手动实现的方向传播可以达到91%的准确率。和使用tensorflow封装的梯度下降函数差不多。

在tensorboard 中的计算图如下：



求导图：



## 实验总结：

反向传播对于实现网络的更新具有重要意义。