# Deep-Learning实验报告

## （异或）实验

日期：2017.6.26

学号：24320142202455

姓名：林金鹏

## 实验目标：

在tensorflow 的基础上面，使用多层神经网络解决非线性可分的识别的问题。

## 实验内容：

使用多层神经网络实现异或操作的识别

具体如下识别规则如下：

[0,0]🡪1

[0,1]🡪0

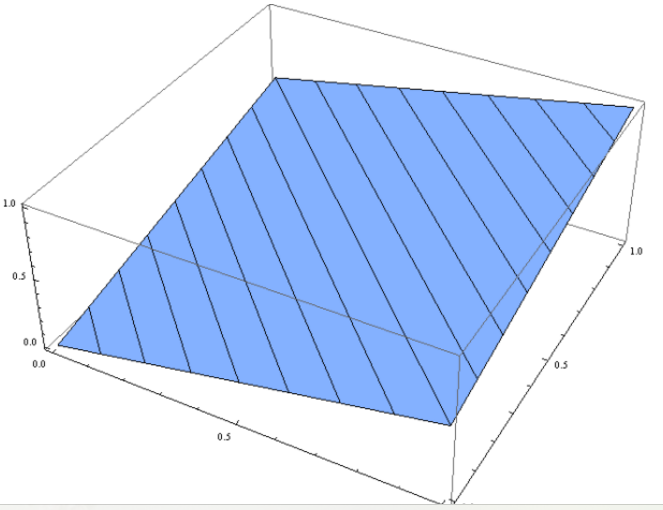
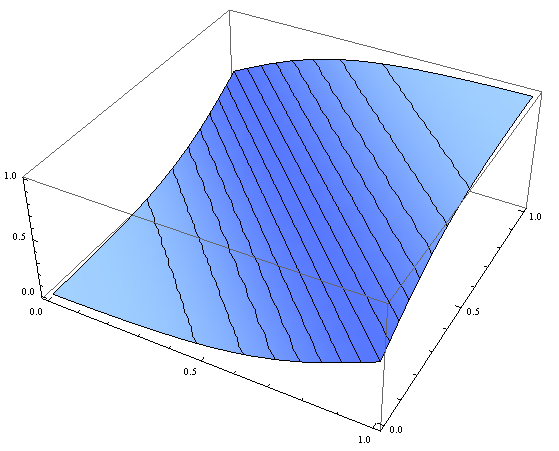
[0,0]🡪0

[1.1]🡪1

## 实验相关知识：

* 非线性可分

对于分类问题，一层神经元相当于一个曲面，可以将解空间的超平面切割成两边，但由于形状问题（没有加入激活函数），无法解决一个线性可分的问题。此时需要加入隐藏层，而隐藏层存在的目的在于多种不同的曲面去叠加拟合给定曲面。而此时如果依然将激活函数取1是不合适的，因为三维空间中任何平面的叠加都是一个平面，此时需要激活函数取将平面弯曲成非平面的形态。

 变成🡪🡪🡪 

* 梯度下降算法

一些梯度下降算法的选用规则:

如果数据是稀疏的，就用自适用方法，即 Adagrad, Adadelta, RMSprop, Adam(在很多情况下的效果是相似的)。

Adam 就是在 RMSprop 的基础上加了 bias-correction 和 momentum，

随着梯度变的稀疏，Adam 比 RMSprop 效果会好。

整体来讲，Adam 是最好的选择。

SGD 虽然能达到极小值，但是比其它算法用的时间长，而且可能会被困在鞍点。

如果需要更快的收敛，或者是训练更深更复杂的神经网络，需要用一种自适应的算法。

经实验发现，本实验使用自适应算法的效果明显好于非自适应算法。

* Tensorboard

简单解释下：TensorBoard是个可视化工具，可以用来查看TensorFlow的图以及过程中的各种值和图像等。

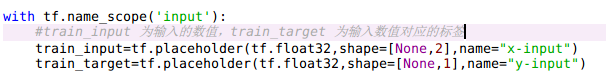
1. 在tensorflow程序中给需要的节点添加“summary operations”，“summary operations”会收集该节点的数据，并标记上第几步、时间戳等标识，写入事件文件。

2.然后 TensorBoard读取事件文件，并可视化Tensorflow的流程。

## 关键代码：

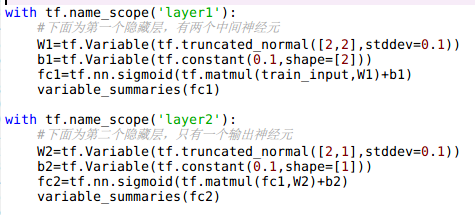
* 首先，进行模型定义。

定义一些初始的变量和占位符:

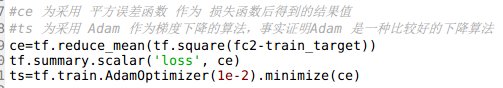


定义两个隐藏层：

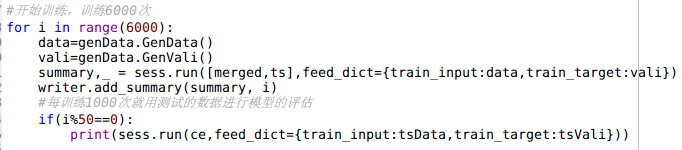




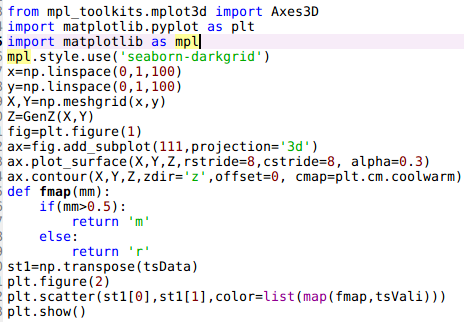
定义损失函数以及以及训练算法：



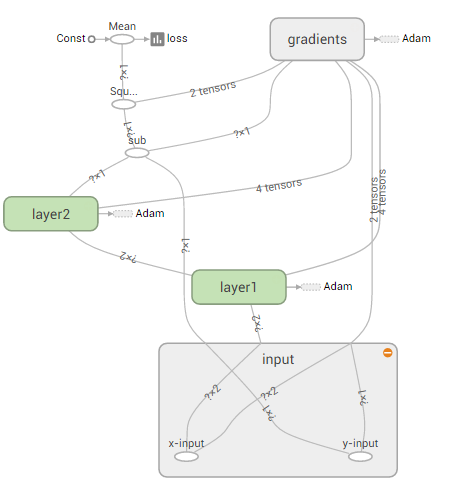
* 上面已经建立了一个有隐藏层的模型，下面开始一个新的对话，然后进行训练。



* 根据实验结果进行图形绘制。

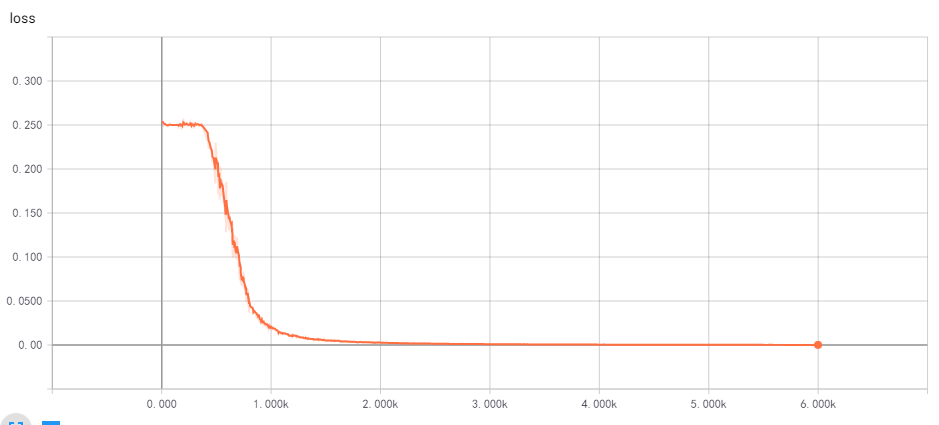


* 通过tensorboard 可以看到整个模型的计算图如下：

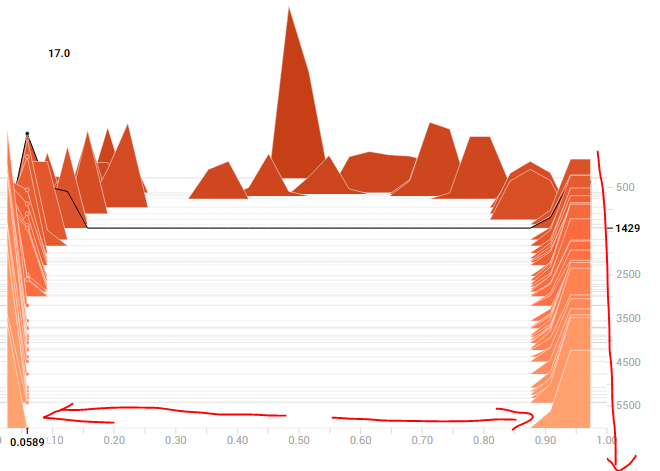


## 实验结果分析：

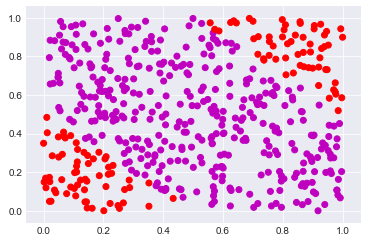
* 损失函数结果值的变化如下，可以看到到1000次时模型基本训练完成。



* 最后一层的sigmoid输出如下，可以看出1000次训练之后的结果已经比较稳定的趋近于0或者趋近于1了：



* 为了看出模型识别的效果，在面积为1的矩形上随机取点，然后通过已经训练好的模型进行预测，可以得到如下一张图（红色的点表示识别为1，蓝色的点表示识别为0）：



通过两条线（y=-x+0.5 和 y = -x+1.5），1x1的正方形被切割成了三部分，可以看出训练出来的模型不仅能识别[1,1],[0,0][0,1][1,0]，还能识别其周围的点。可以是整数的点，也可以是小数的点。

## 实验总结：

利用增加隐藏层并使用恰当的激活函数可以增强神经网络的学习能力，不仅能够进行线性的识别还能进行非线性的识别。

利用tensorboard可以很方便地观看到各个参数变量在训练过程中的变化，以及整个计算模型的整体架构。