智能计算体系结构课程lab2实验报告

作者：北京航空航天大学计算机学院 陈胤佳

学号：19373383

一、实验目的与要求：

本次实验采用Python语言、结合集成机器学习框架Tensorflow，进行了简单的卷积神经网络（CNN）LeNet的搭建；该网络是一个较为简单、小型、用于识别手写数字的CNN模型，但包含了全部深度学习的基本模块。搭建完成后，使用给定的训练集和测试集，Mnist官方数据集对训练模型进行训练与测试。内部细节采用了业界最为经典的几种算法与函数，旨在Lab1对基础DNN结构有了详尽掌握的基础上，对CNN结构进行进一步熟悉，并掌握常用机器学习框架的使用方法，以服务于课程后期智能计算体系结构的搭建。

二、实验环境：

IDE：PyCharm 2021.2.2 (Professional Edition)；

Environment：Python 3.7、Tensorflow；

OS：Windows 10 Professional

三、实验内容与步骤

1. 分析思路：

本次实验的重点在于理解CNN的结构与工作模式。以LeNet为例，CNN网络模型共包括7层，分别为卷积层1、池化层1、卷积层2、池化层2、与三层全连接层。

卷积层部分的参数包括每个特征平面各一套的一个卷积核、与一个偏移量；卷积核用于提取特征，每一次提取对应一个神经元；对同一个特征平面采用权值共享模式，大大减少了权值参数的数量，在一定程度上能够避免过拟合。

池化层将特征平面进行压缩，进一步降低数据维度，有利于减少后面几层的计算量。压缩过程采用对一个选择框进行求和取平均值再乘上一个权值、加上一个偏移量的方式。

其中卷积层包含了激活函数，而池化层则不包括。

进入全连接层前，需将第二层池化层输出的高维向量按一定规律压缩为一维向量；全连接层部分与Lab1的DNN原理一致，共有三层。

在LeNet中，输入图片尺寸为32\*32，经6个不同的5\*5的卷积核卷积运算后，会生成6个28\*28的特征平面；特征平面经2\*2的池化层选择框进行压缩，会生成6个对应的14\*14的压缩后特征平面。经过特定的转换后，池化层的输出会转化为16个特征平面，再重复上述过程，经过卷积层+池化层，压缩后最终生成784\*1的全连接层输入。

据此，前向传播算法结束，其中的诸多细节操作，如卷积、池化均可以通过Tensorflow框架提供的方法完成。进一步，采用传统算法反向更新各层参数，并重复此过程，直至损失函数的结果降低至一定范围内，即误差被认定为正确，则该数据集训练完毕。由于更新权值、偏移量部分的工作本次实验并没有做出要求，因此在此不做详述。

2. 设计方案

在课程组提供模板项目（TF-Mnist-Template）的基础上进行修改以及补全。

模板结构：

Input\_data.py：包含了读取数据集文件的过程，用于加载mnist数据集。

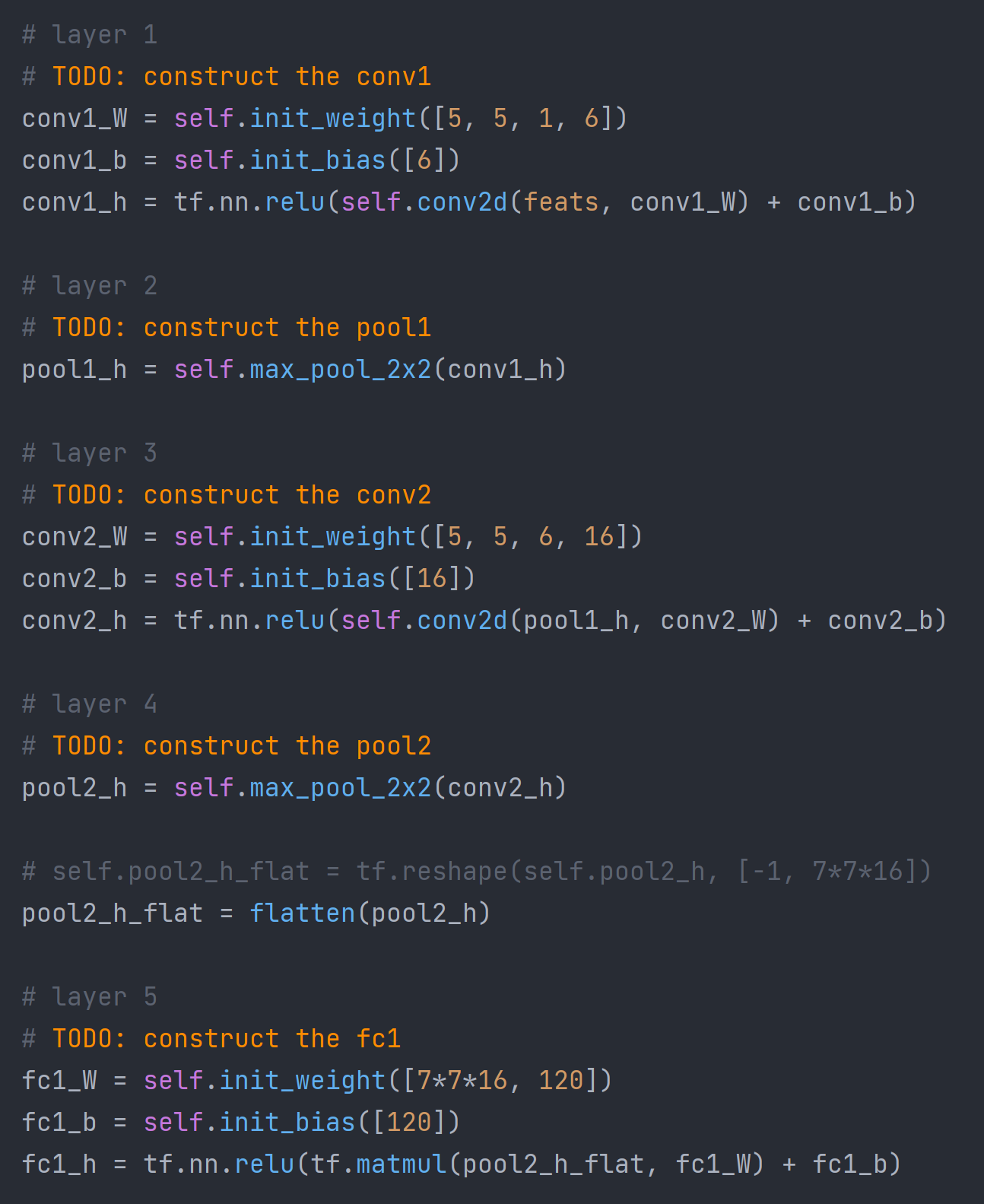
tf\_minist.py：主文件，包含了训练、测试流程、以及模型的保存过程。

data\_object.py：数据类文件、包含处理数据的实现。

tf\_model.py：测试固定参数后，所训练的模型的准确率。

tf\_network.py：包含了LeNet类，其中所包含的构造器方法、net()方法、forward()方法是要填充核心代码的部分。外部会调用forward()方法、后者会调用net()方法，前向传播计算网络输出，需要填写的部分主要位于net方法中。

net()方法主要管理LeNet的前向传播，包括在同时对于各层变量进行初始化，代码如下：



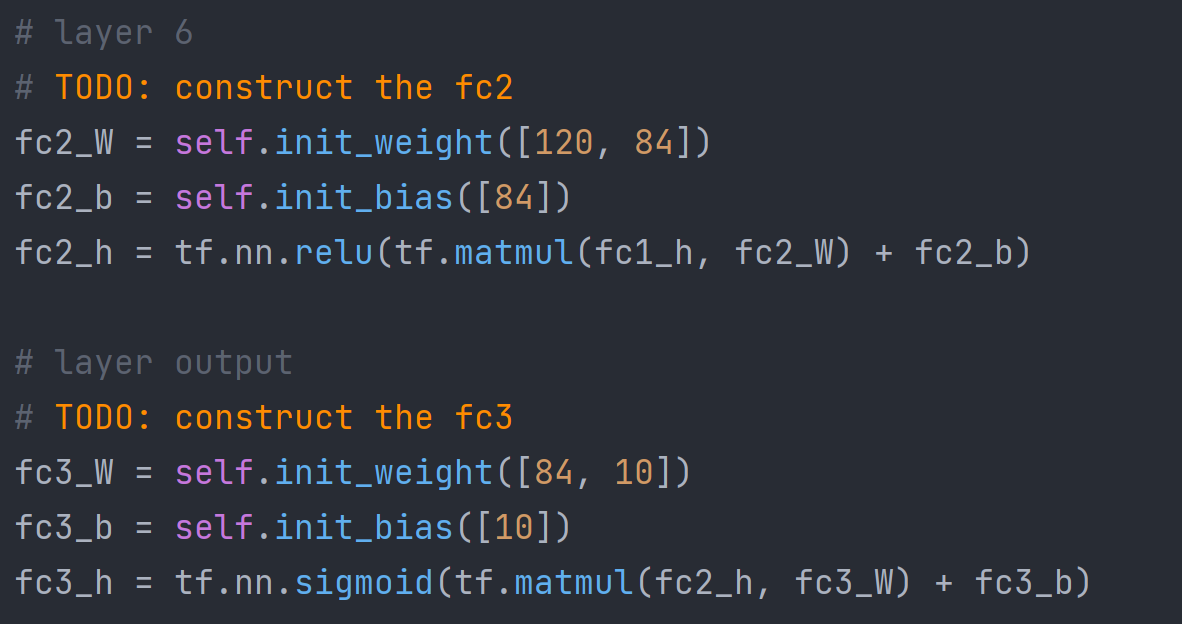


图1

它会返回网络的运算结果。

根据设计方案在模板中实现Python代码。

3. 遇到问题

主要问题在于Tensorflow新版本某些方法包含库做出了调整，导致模板中部分函数不能使用。

解决方式：将解释器改为Python3.7，下载tensorflow 1.13.1、覆盖新版本后可以使用。

四、实验结果

训练过程输出结果如图2：

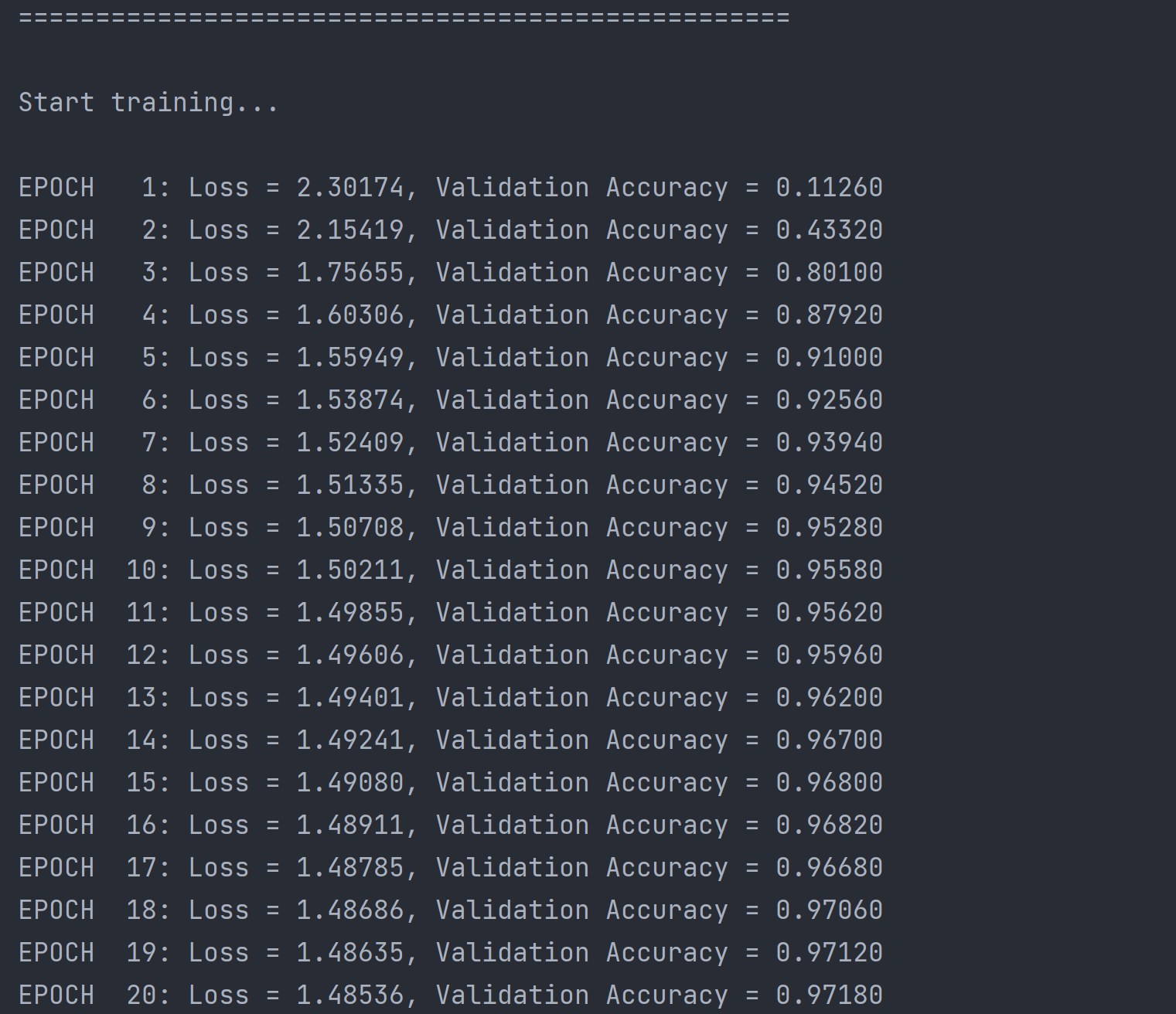


图2

可以观察到在20遍训练过程中，损失由最开始的2.30174收敛到了最终的1.48536，准确率由11.26%收敛到了最终的97.18%，训练效果较为良好。

测试过程输出结果如图3：

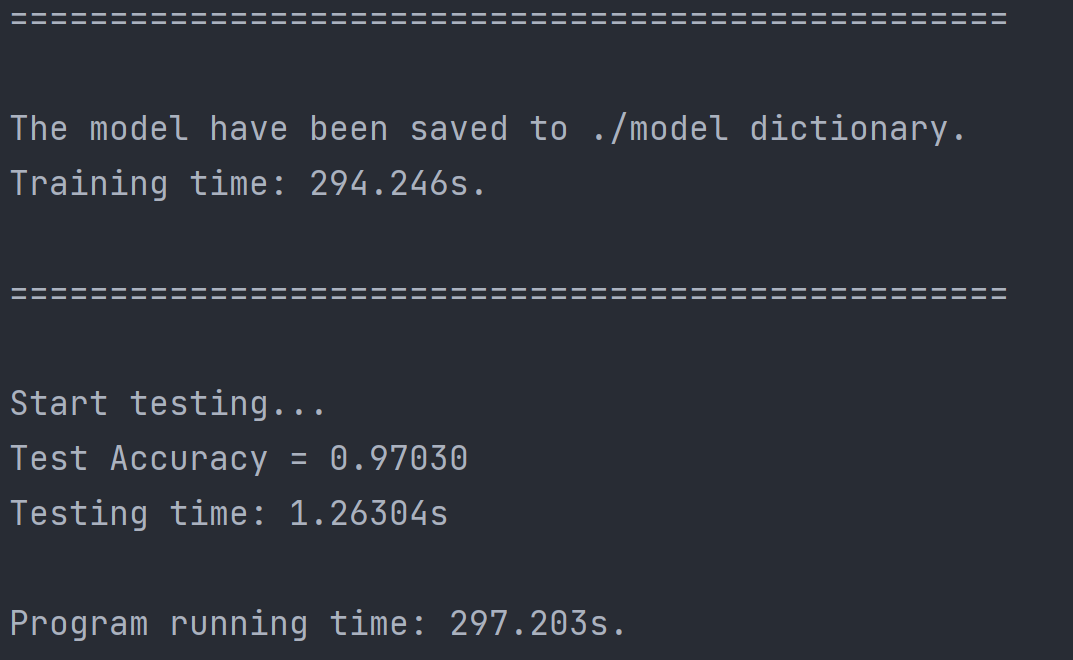


图3

可以观察到测试结果为准确度0.97030，准确度较高，训练效果较好。

Tensorboard结果可视化（图4为多次训练的准确度收敛情况、图5为多次训练的损失收敛情况）：

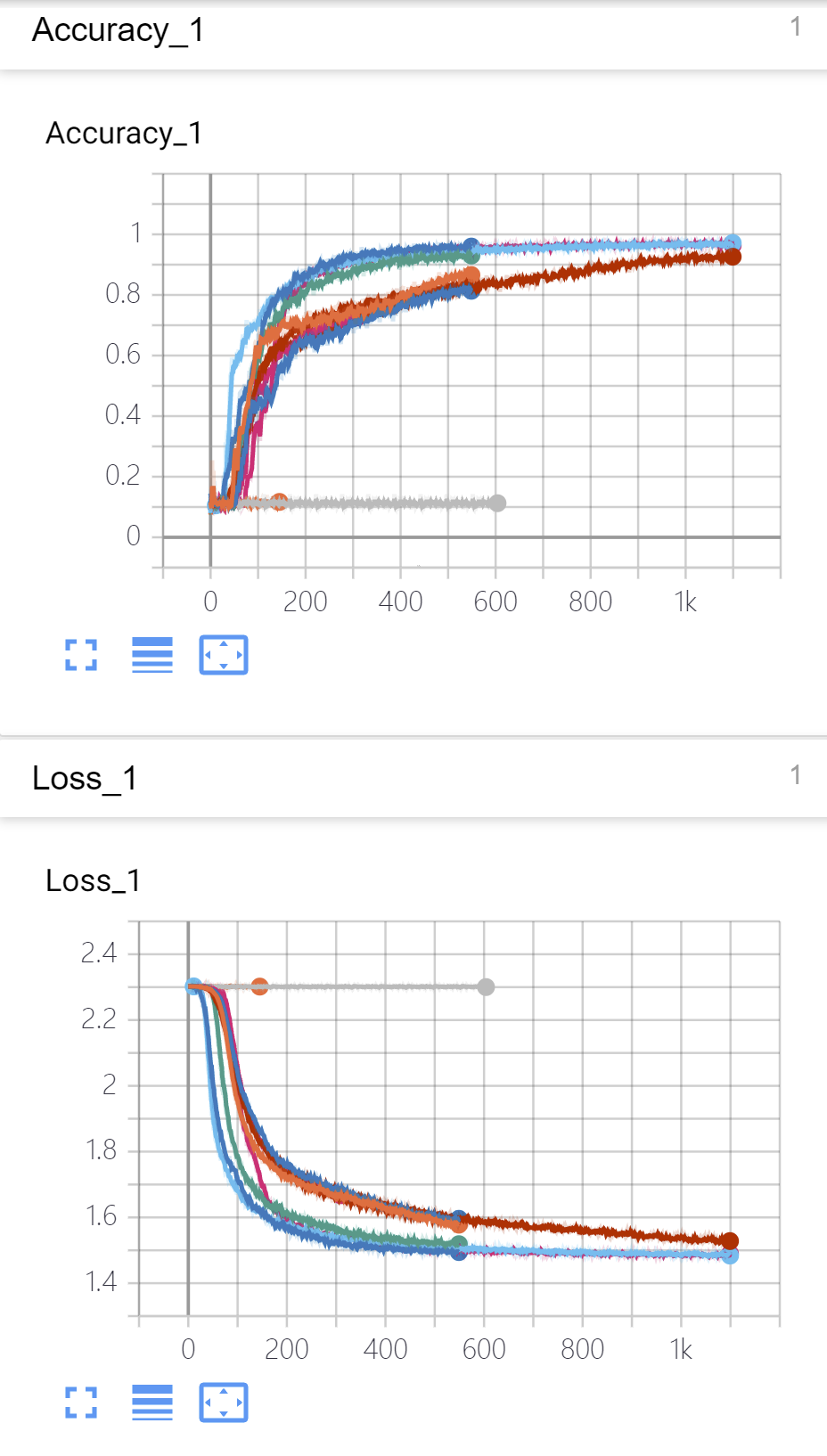


图4

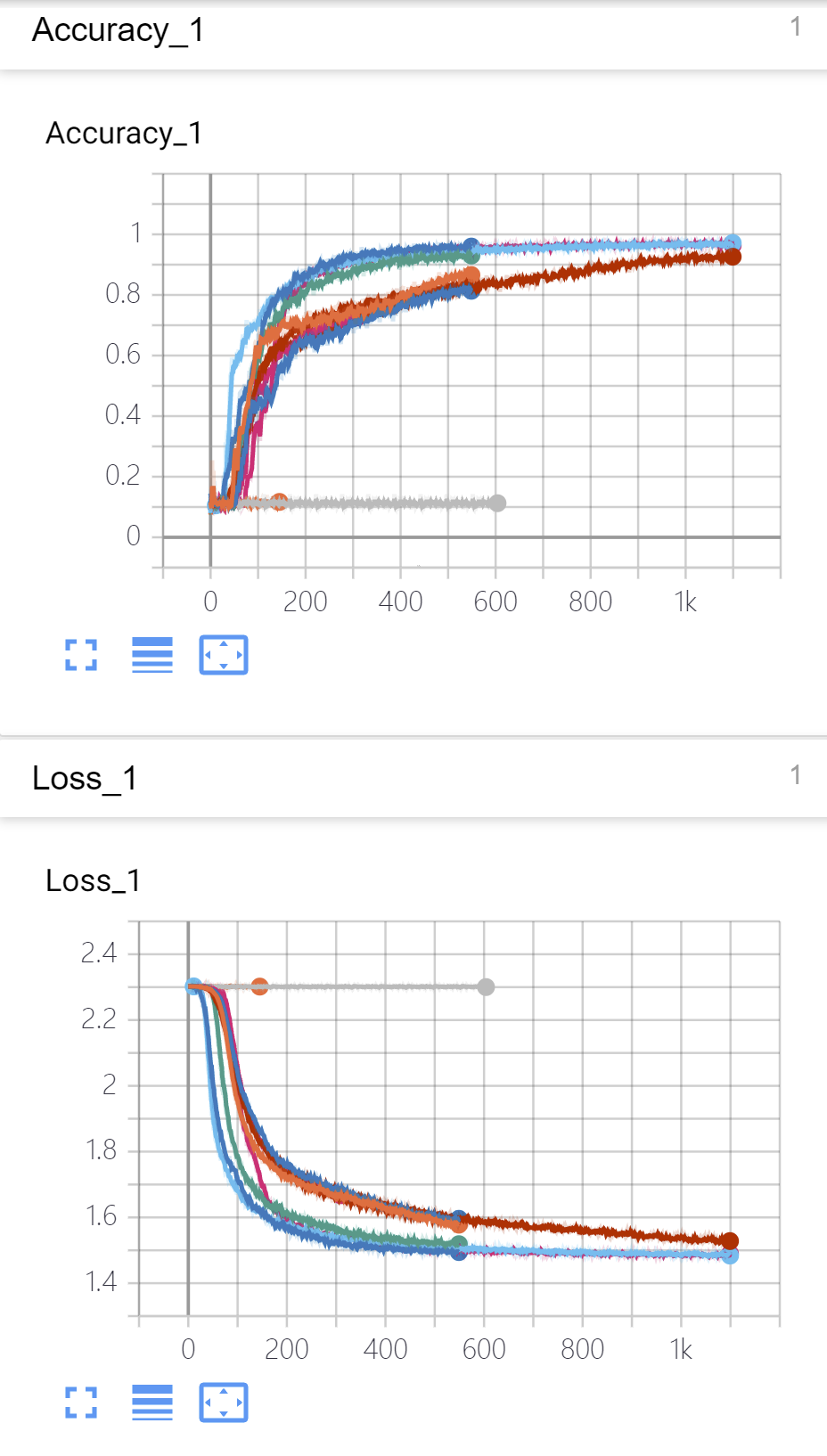


图5

其中，浅蓝色线下覆盖住的粉色线为上文示例中的训练样例。

五、分析与讨论

本次实验利用TensorFlow框架，根据已有算法实现了一个简单的卷积神经网络，并利用所提供的数据集对该网络进行了训练与测试。本次lab旨在对卷积神经网络的结构与运算过程进行理解与掌握、并对常见深度学习框架进行初步运用。实际上只要理解了CNN的结构以及LeNet的特点，建立模型的过程实际上并不困难。希望本次实验能够为日后课程的lab打下坚实的基础。