

# MLP-MINST实验报告

---

姓名：常亮

学号：2016211207

## 概述

---

本次实验是使用MLP(Multi Layer Perceptron)在MNIST数据集上进行手写数字的识别，由于之前使用Tensorflow, Pytorch等框架实现过，但经过上课学习后，发现自己关于模型的还有许多细节不够了解，又由于之后自己计划从事自然语言处理方向。本次不使用机器学习框架，考虑到计算的速度问题，仅使用Python中的Numpy库进行编写，使用python-mnist进行数据集的读取。

在实现中遇到许多困难，但模型所有代码均为自己原创。

本次实验完成：

在训练集上训练参数，在测试集上测试准确率

使用了softmax分类器

使用了全连接神经网络(MLP)分类器

实现了MSE loss, 交叉熵loss, L2正则化

实现了batch GD, mini-batch GD, SGD优化算法

实现了学习率下降算法

## 任务定义

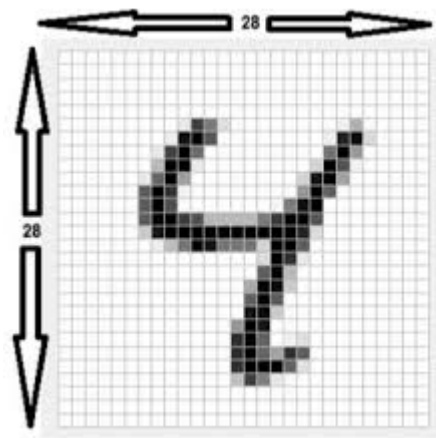
---

在邮件分发的过程中，有大量邮件需要根据邮政编码决定发往何处，这一过程是非常枯燥的，如果能够使用计算机进行自动化，必然极大的提高处理效率。

因此，我们的任务是对一组写有阿拉伯数字的黑白图片进行识别

## 输入

28\*28的像素方阵，共784个数字，每一个数字范围为0~255，表示单个像素的灰度值。



## 输出

输出一个阿拉伯数字作为结果(0~9)

## 方法描述

本次实验使用多层感知机模型，输入层为784个灰度值，经过若干个隐层，输出为该图片对于10个数字的概率值，取概率最大的一个作为预测结果。

## 结果分析

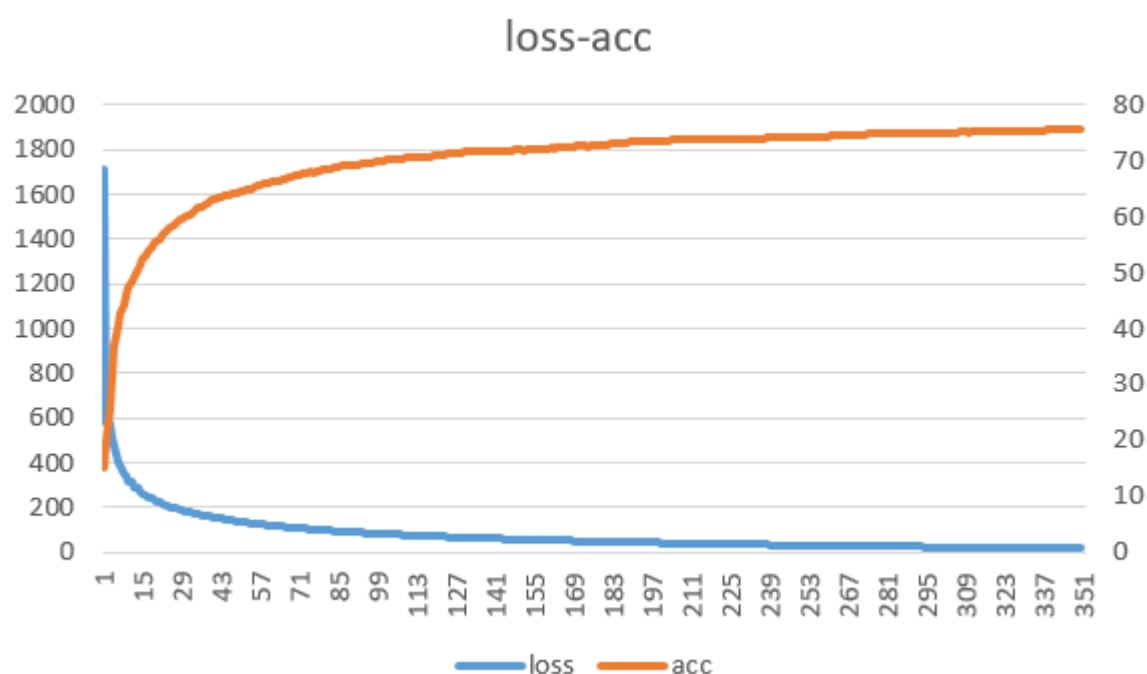
本次实验是迭代式的，在模型上增加一个个trick，观察trick的作用

每次实验将随机种子固定，将不同实验结果的随机性减少到最小。

### 基本实验

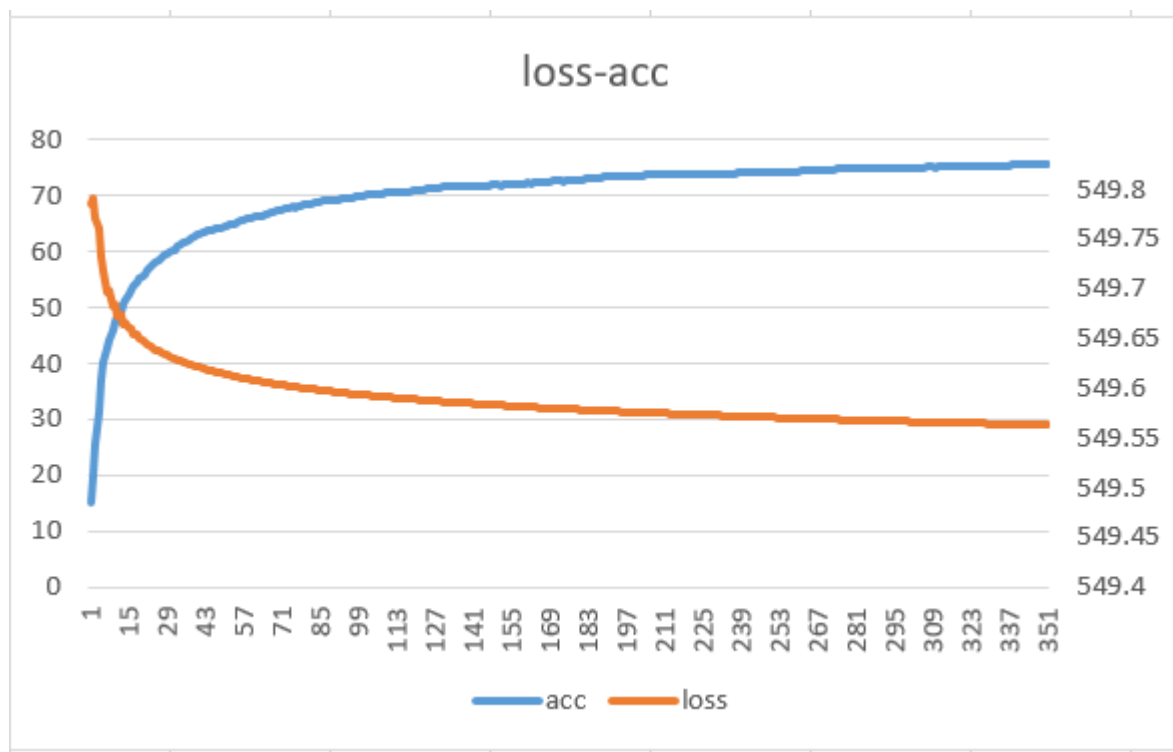
sigmoid+softmax+交叉熵+mini-batch

batch\_size为550



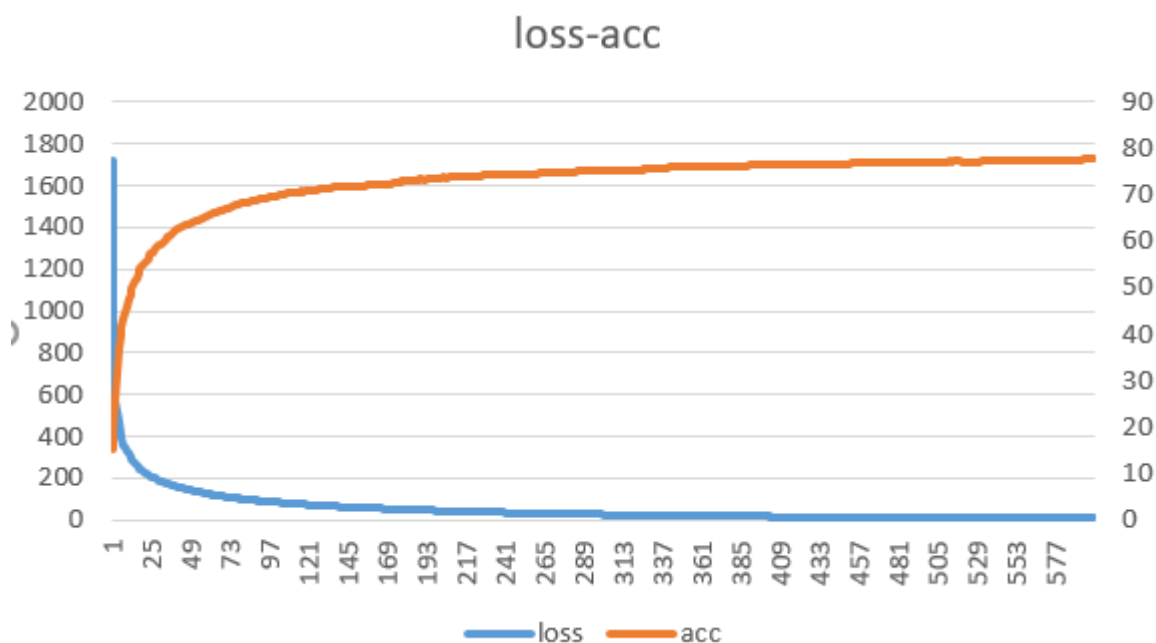
最终测试集准确率为75.77

## 加入L2正则化



最终测试集准确率为76.30 结论：L2正则化可以提高模型表现

## sigmoid+softmax+交叉熵+学习率下降算法



采用学习率下降算法，在350个epoch时，准确率为76.02，但发现准确率未收敛，在600个epoch的实验后，最终准确率为77.68。结论：使用学习率下降算法可以提高模型表现，但是收敛速度变慢

## sigmoid+softmax+交叉熵+学习率下降算法+训练集乱序

## 实验记录

实验设置	隐层大小	模型效果
L2 loss+minibatch	300	62
L2 loss+minibatch	300,100	71.05
交叉熵loss+minibatch	300	75
交叉熵loss+minibatch	300,100	81.05
交叉熵loss+minibatch	300,100,100	79.4
交叉熵loss+minibatch	300,200	81.6
交叉熵loss+minibatch	800,100	<b>83</b>
交叉熵loss+softmax+minibatch	300	75.77
交叉熵loss+softmax+minibatch+L2正则化	300	76.3
交叉熵loss+softmax+minibatch+L2正则化+学习率下降	300	77.68
交叉熵loss+softmax+minibatch+L2正则化+训练集乱序	300	76.2
交叉熵loss+softmax+batch+L2正则化+训练集乱序	300	74.03
交叉熵loss+softmax+SGD+L2正则化+训练集乱序	300	8.92

## 源码运行环境

python3

numpy

python-minst(<https://github.com/sorki/python-mnist>)