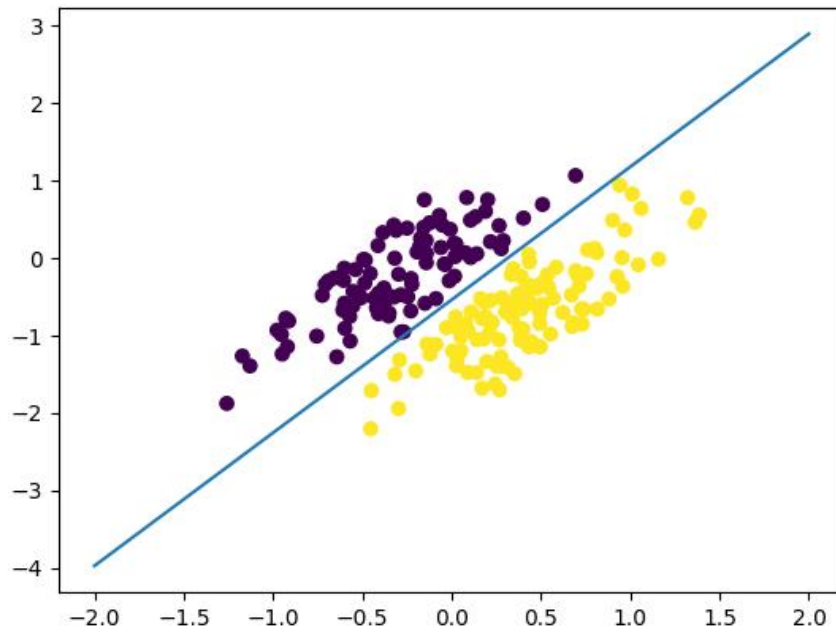


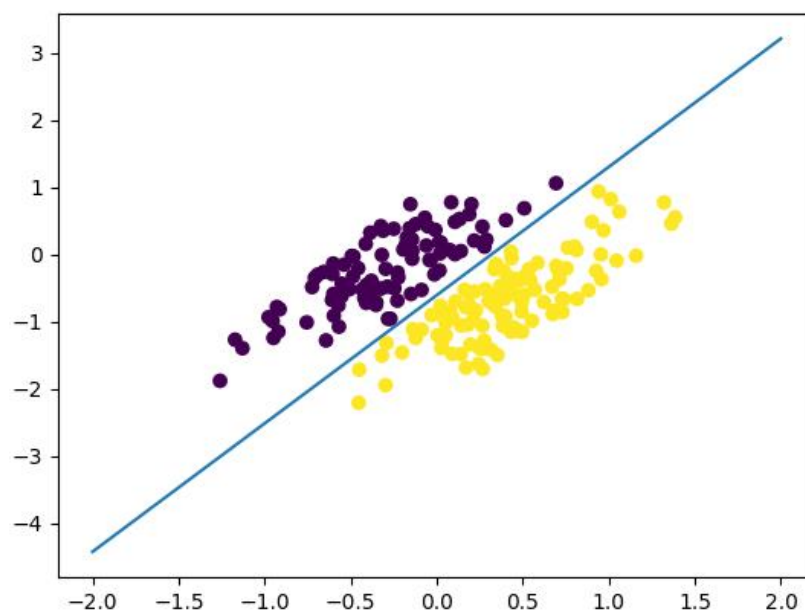
Assigenment2 报告
阳希明 15307130263

Part1:

Least square model 的实验效果如下图:



其准确率为: 100%



Perceptron model 的实验效果如下图:

其准确率为: 100%

Part 2

1.梯度的计算过程如下:

在添加正则项的情况下, 权重 W 与正则项的梯度为 (将 W 与 b 合并为一个矩阵的情况):

$$\frac{\partial(\text{norm})}{\partial W} = 2 * l * (W - b)$$

Softmax 和交叉熵部分, 输入 a 相对于输出 y 的梯度为:

$$\frac{\partial y}{\partial a} = a - y$$

Sigmoid 激活函数部分, 输入 x 相对于输出 a 的梯度为:

$$\frac{\partial a}{\partial x} = \frac{1}{1 + e^x} * \frac{e^x}{1 + e^x}$$

权值乘以输入数据部分, 权值 W 相对于输出 x 的梯度为 (其中 X 为输入):

$$\frac{\partial x}{\partial W} = X$$

所以最终, 在 loss 为 s 的情况下, W 的梯度为

$$\frac{\partial s}{\partial W} = \frac{\partial s}{\partial y} \frac{\partial y}{\partial a} \frac{\partial a}{\partial x} \frac{\partial x}{\partial W} + \frac{\partial(\text{norm})}{\partial W} = s * (a - y) * \left(\frac{1}{1 + e^{xW}} * \frac{e^{xW}}{1 + e^{xW}} \right) * X + 2 * l * (W - b)$$

在实现中, 使用了三个函数来处理梯度的各个部分, 并将结果相乘得到最终梯度。

2.1.2 正则是否需要正则偏置项

由于实际应用中并没有对输入数据正规化, 导致偏置项可能会需要比较大以拟合给出的数据。而正则项是基于参数 W 和 b 在数值上偏大的惩罚, 目的是防止参数项数值过大使得模型复杂, 造成对数据分布估计的过拟合。偏置项数值的变大基本不会影响模型的复杂性, 所以不需要放入正则项中。

3.如何检测梯度计算的正确性

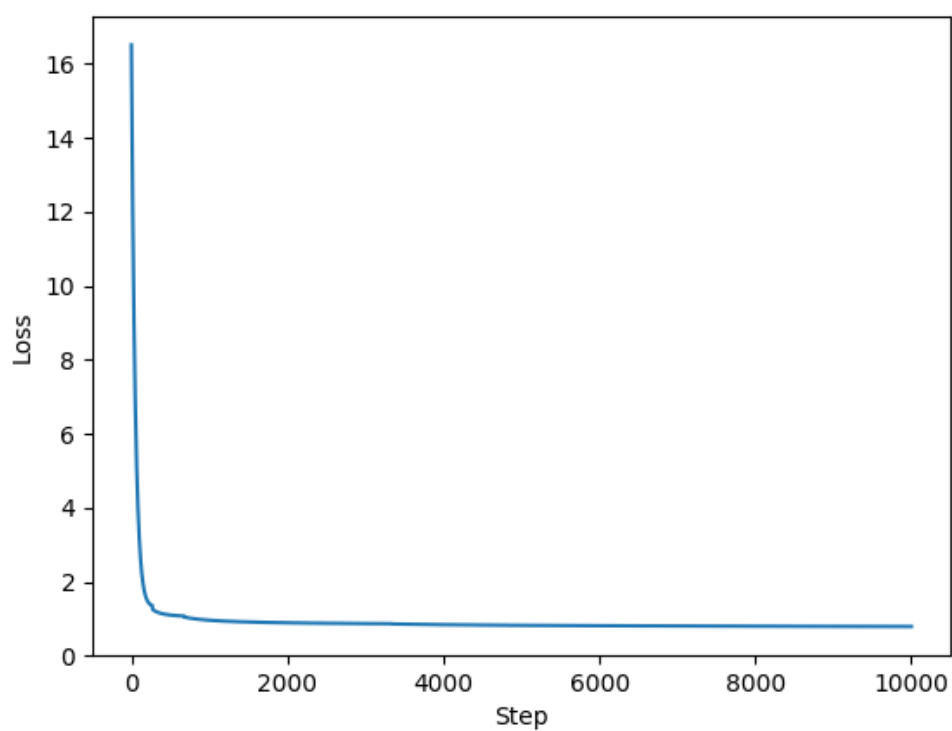
在这次项目中, 我通过手动设置维数极小, 数据简单的输入数据和二分类的简单目标来手动检验梯度计算是否正确。在验证的过程中, 参数初始化为全 1 而非全随机。手动计算每一步梯度输出的预期结果并与程序运行结果相比较, 得到对正确性的判断。

4.如何指定学习率, 如何判断终止训练的时间

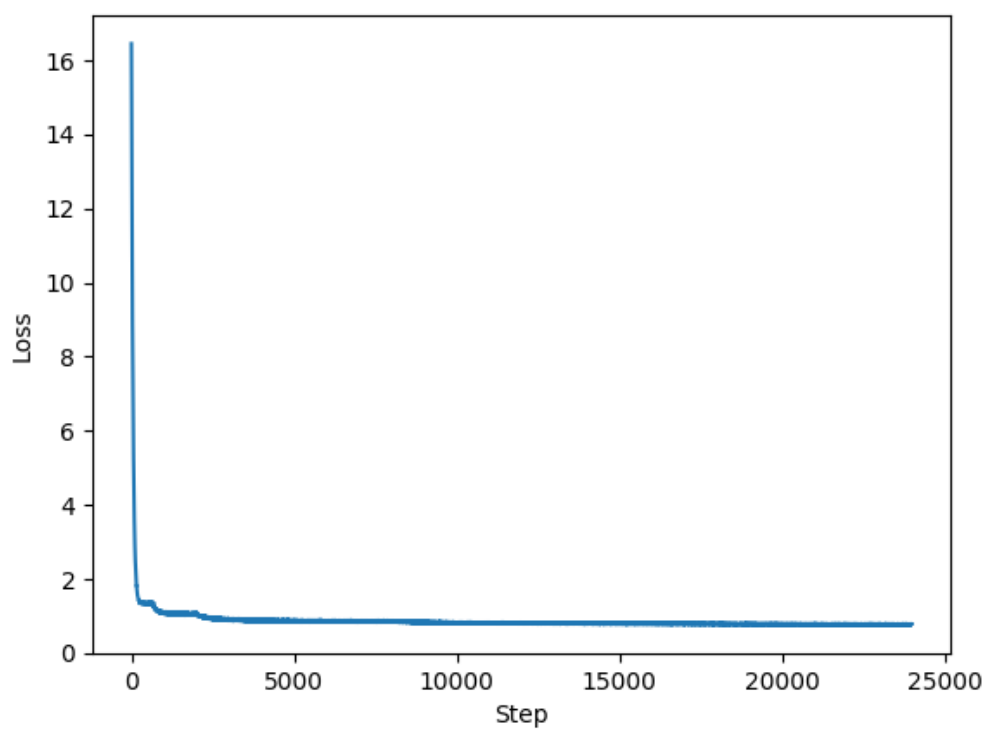
在平常的深度学习过程中, 一般通过给定学习率, 按时间衰减等方式给定学习率。在本次项目中, 我计算了一定窗口的 epoch 内 loss 的平均值, 计算和上一个窗口 loss 的下降幅度, 如果下降幅度小于一个给定的超参数阈值, 则同时按设定比例下调学习率 and 这个阈值。直到学习率低于另一个给定的最低学习率, 下降幅度小于阈值则结束训练。实验中能够很好的加速训练, 保持训练效果, 但由于超参数众多, 不是一个普适的训练方法。实际训练中多以指定最大 epoch 数或手动退出来结束训练

5.调整 batchsize 的研究

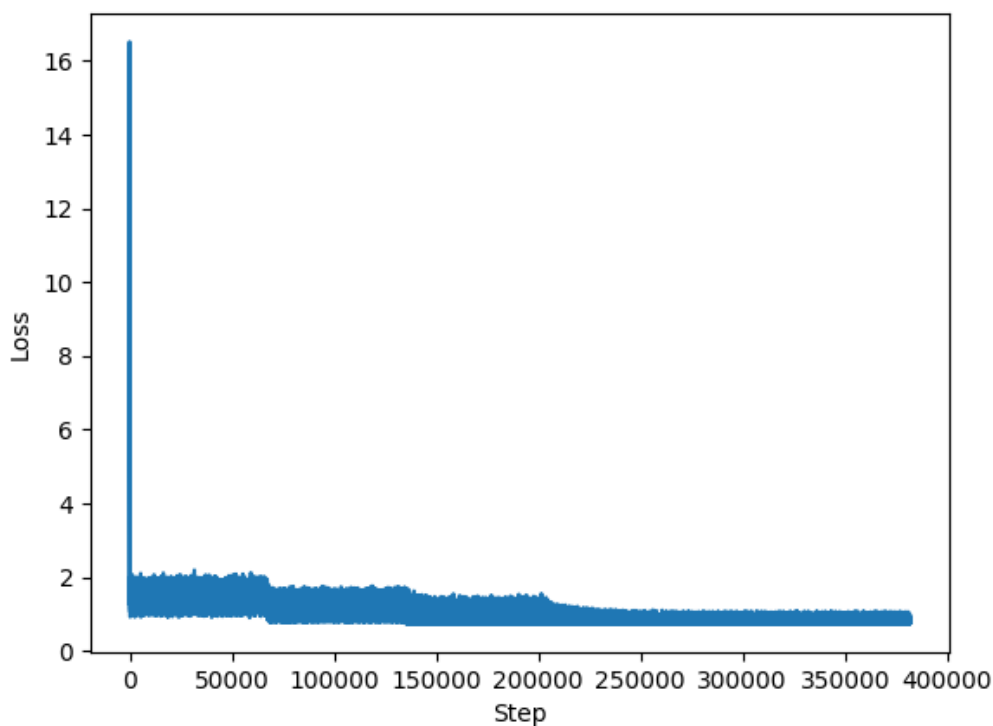
这次项目中共尝试了三种 batchsize, 分别是全数据, 128 和 1. 他们的训练效果如下:



Batch Size = Full



Batch Size = 128



Batch Size = 1

三个模型训练时间，训练集准确率，测试集准确率如下表：

模型 Batch Size	训练集准确率	测试集准确率	训练时间
Full	0.994214	0.780080	1311.720s
128	0.998220	0.770534	59.942s
1	1.0	0.763369	129.224s

可以看出，full batch size 的测试集准确率最高，拟合效果较好，但训练时间过于漫长，也经历了最多的 epoch。Batch size 为 1 的模型出现了较为严重的过拟合现象，loss 的波动较为严重，但训练经历的 epoch 数最少。实现时使用 numpy 进行矩阵运算，batch size 过小导致了单位数据运算开销过大，整体运行时间不如 batch size 为 128 的模型。综合考虑效率与准确率，batch size 适中的模型具有更加优秀的表现。

实际应用中，full batch size 由于内存或显存的瓶颈根本无法实现，batch size 的大小主要由训练设备的内存或显存而决定，对于最终效果的影响并不大。