

计算机科学与技术学院神经网络与深度学习课程实验报告

实验题目: Homework 1		学号: 201605130098
日期: 2019. 4. 12	班级: 16 智能	姓名: 李云开
Email: liyunkai@foxmail.com		
<p>实验目的:</p> <p>In this assignment you will practice putting together a simple image classification pipeline, based on the softmax classifier. The goals of this assignment are as follows:</p> <ul style="list-style-type: none">• understand the basic Image Classification pipeline and the data-driven approach;• implement and apply a softmax classifier;• implement and apply a three-layer neural network classifier. <p>You will be given a set of *.py files which include tools to check your code. You just need to upload all your code and report and do not upload datasets.</p>		
<p>实验软件和硬件环境:</p> <p>Python</p>		
<p>实验原理和方法:</p> <p>链式法则</p>		
<p>实验步骤: (不要求罗列完整源代码)</p> <p>嗯, 我感觉这个实验的主要任务在于推导。</p> <p>理解 softmax_train</p> $\text{loss} = -\frac{1}{N} \sum_i y_i \log p_i + \frac{1}{2N} \text{reg} \sum_L \ W_L\ _2^2$ <p>主要是</p>		

我更喜欢这么写的

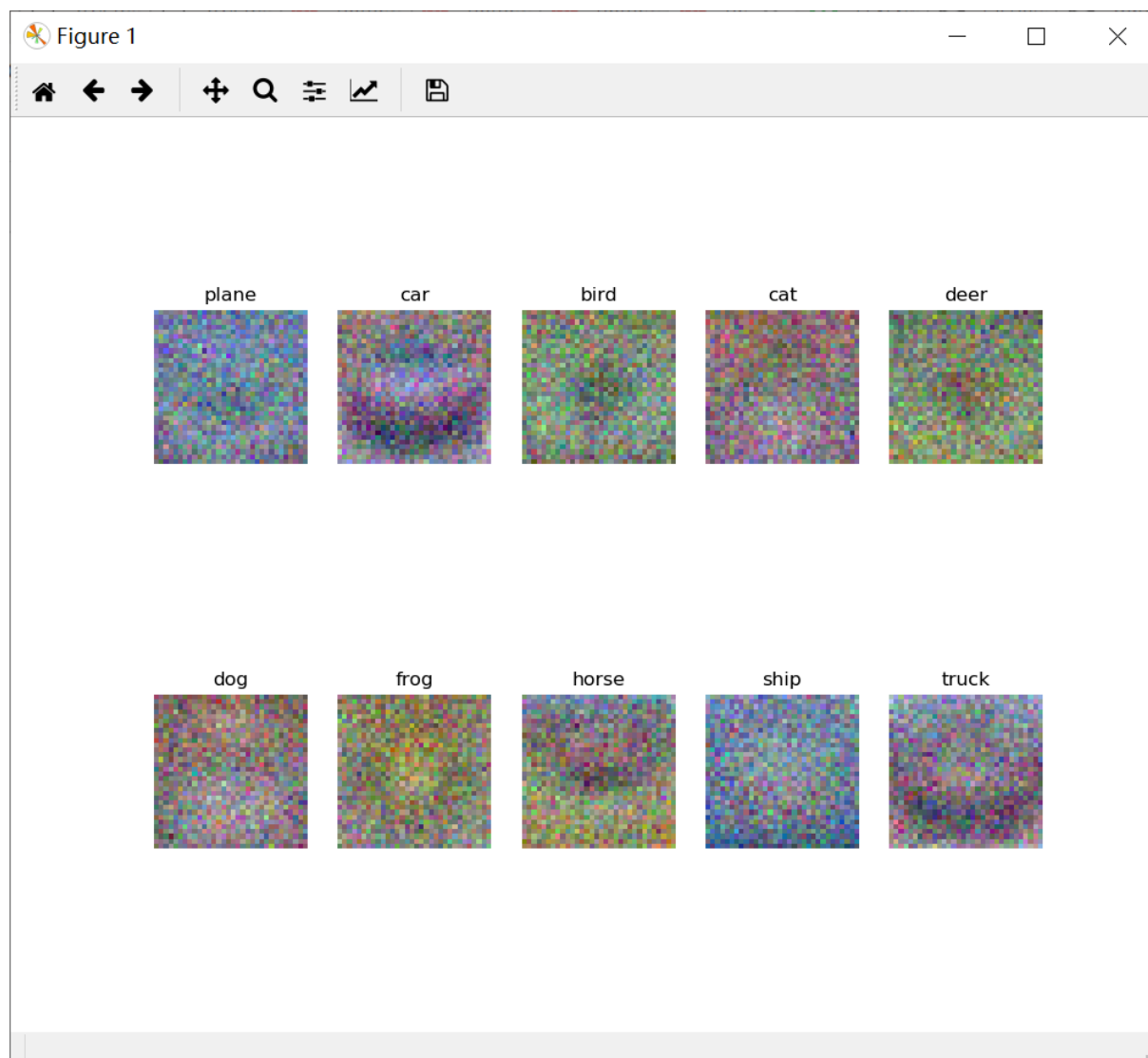
The loss can be described as:

$$\text{loss}(x, \text{class}) = -\log \left(\frac{\exp(x[\text{class}])}{\sum_j \exp(x[j])} \right) = -x[\text{class}] + \log \left(\sum_j \exp(x[j]) \right)$$

(来自 [pytorch.org torch.nn.CrossEntropyLoss](https://pytorch.org/torch.nn.CrossEntropyLoss))

这个式子的求导比较简单，就是 softmax 的结果减去一部分，然后用链式法则做就好。

下图来自 softmax_train.py



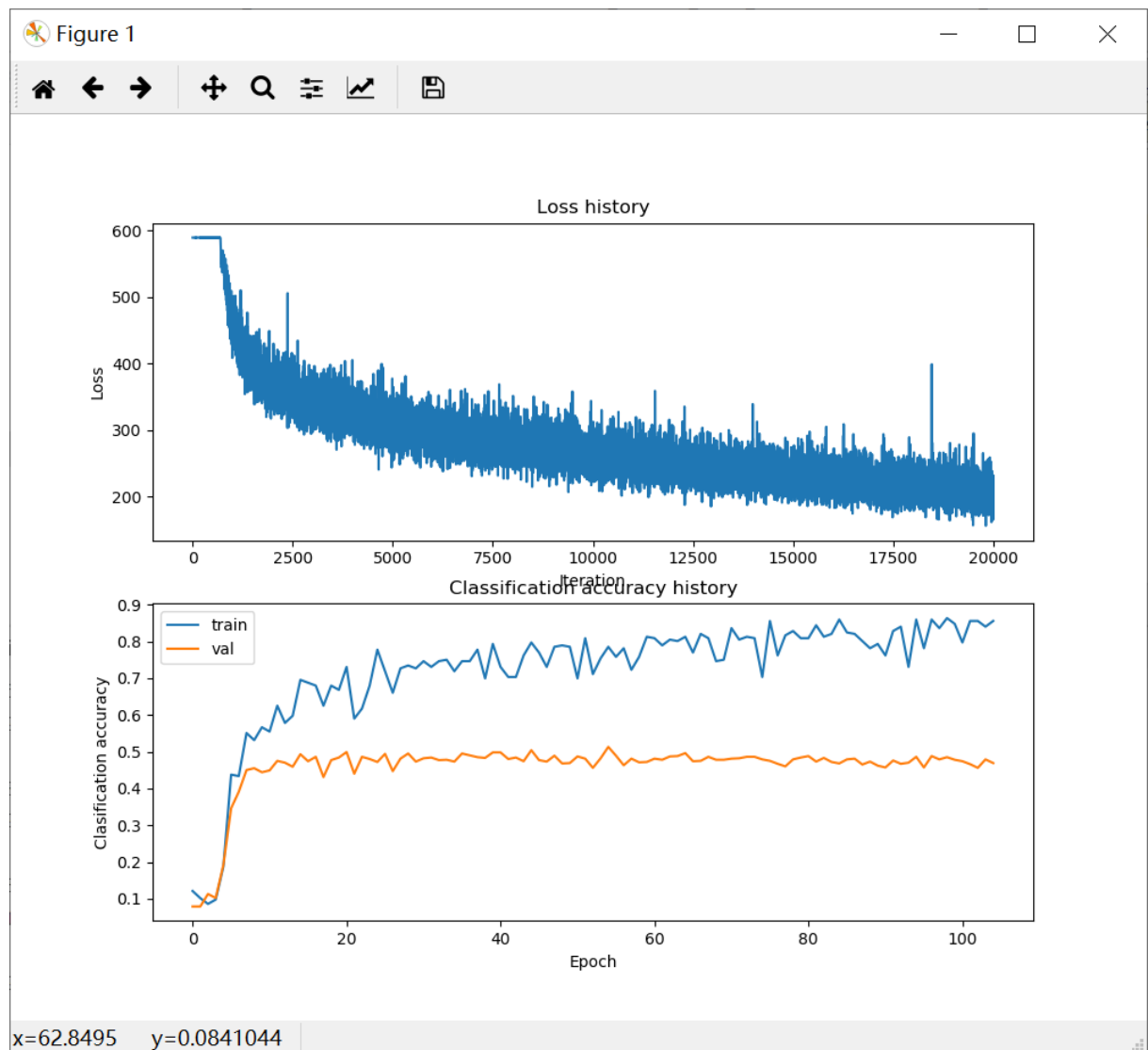
Inline Question:

loss: 2.392328

sanity check: 2.302585

这里 W 使用的是随机初始化的矩阵，输入 X_dev, Y_dev，第一次输出的十类分类结果概率是均等的，每类均为 0.1，. 经过 NLLoss，就是 $-\log(0.1)$ 。这里我加了 reg_loss，. 所以比 2.30 大

这是调参时的一个图， hidden-size= 50, batch-size=256, 后期能明显看得出过拟合



结论分析与体会：

Softmax_train.py : 36.1%

Three_layer_net.py: 56.8% hiddenlayer = 256

就实验过程中遇到和出现的问题，你是如何解决和处理的，自拟 1—3 道问答题：

Relu 与 leakyrelu，如何取舍：

学长说 relu 坑，网上的代码都是 relu 的，二层神经网络改三层神经网络很容易，交叉熵的推导卡了我半天，leakyrelu 只需要改一改两个地方就 ok。