

计算机视觉 PJ1

姜俊哲 19307130152

April 1, 2023

目录

I 训练	1
i Adam 算法	1
ii 小批量梯度下降	1
iii 正则化	1
II 参数查找	1
III 测试	2
IV 代码地址	2

I. 训练

使用 \tanh 作为激活函数, 采用 $Adam$ 算法使用动量作为参数更新方向并自适应调整学习率, 使用小批量梯度下降算法, 采用 $L2$ 正则化与提前停止策略

i. Adam 算法

记 g_t 为第 t 次迭代计算出的梯度

$$M_t = \beta_1 M_{t-1} + (1 - \beta_1) g_t$$

$$G_t = \beta_2 G_{t-1} + (1 - \beta_2) g_t \odot g_t$$

其中参数取值 $\beta_1 = 0.9, \beta_2 = 0.99$ 。为防

止初始阶段初值为 0 的偏差, 对其进行修正

$$\hat{M}_t = \frac{M_{t-1}}{1 - \beta_1^t}$$

$$\hat{G}_t = \frac{G_{t-1}}{1 - \beta_2^t}$$

对网络参数的更新为

$$\Delta w = -\frac{\alpha}{\sqrt{\hat{G}_t + \epsilon}} \hat{M}_t$$

其中所有乘除均为按元素进行, α 为后面进行参数查找的学习率, ϵ 为防止分母为 0 的小量

ii. 小批量梯度下降

经过一些测试, 我选取了效果不错且计算速度较快的 $mini_batch_size = 8$

iii. 正则化

采用 $L2$ 正则化 $\frac{1}{2}\lambda||w||_2^2$, 正则化强度为后面参数查找的参数之一

按照 9 : 1 将训练集划分为训练集和验证集, 每 5000 次计算一次验证集上的误差, 当 20 次都没有再下降的时候提前停止训练。

II. 参数查找

选取查找范围, 隐藏层大小 $layer \in \{50, 100, 200, 300\}$, 学习率 $\alpha \in \{1e-2, 1e-$

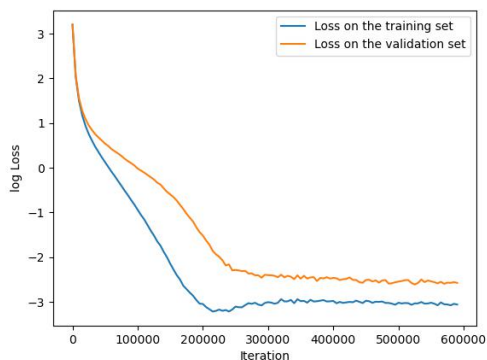
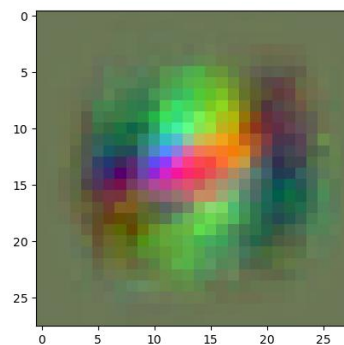
图 1: 训练的 \log -Loss 曲线

图 3: 输入层到隐藏层

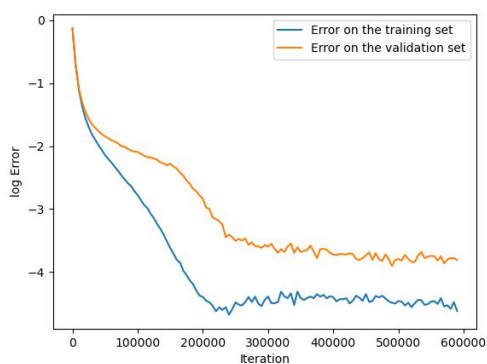
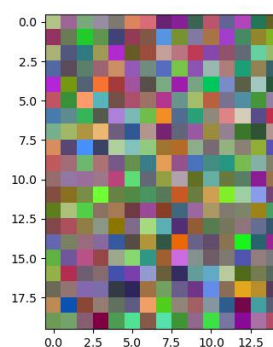
图 2: 训练的 \log -error 曲线

图 4: 隐藏层到输出层

$3, 1e-4, 1e-5\}$, 正则化强度 $\lambda \in \{1e-2, 1e-3, 1e-4, 1e-5\}$ 。

遍历每种组合,并记录在验证集上表现最好的模型超参数,并保存模型。得到最好的超参数为 $layer = 300, \alpha = 1e-4, \lambda = 1e-2$

III. 测试

对参数查找后的模型在测试集上进行测试,准确率为 97.42%

训练的 Loss 曲线 (不含正则化项) 见图1, 训练的 Error 曲线见图2

可以看到由于正则化的存在,在训练集上 loss 和 error 先下降后有轻微上升,同时

使验证集上 loss 和 error 继续下降,也符合我们的要求

对两层的权重用 PCA 降维并可视化,见图3、图4

IV. 代码地址

代码在 [selfspin/Fudan-CV-Assignment-1](#) 上开源。为了方便, MNIST 数据集我将其转化为 npy 文件存在与最终模型和代码同一目录下, 您可以不用额外下载数据直接 git clone 并运行代码; 或者在此网盘地址下载数据文件和模型文件: [百度网盘地址](#), 提取码: **CVPJ**

参考文献

- [1] 邱锡鹏 神经网络与深度学习