

作业 1

2024 年 3 月 13 日

本次作业通过多层感知机实现非线性分类任务，帮助同学们掌握使用 Pytorch 训练神经网络。作业内容分为理论部分、编程部分以及作业报告。其中理论部分包含第 1, 2 题，所有同学均需完成，答案附在作业报告中；编程部分包含第 3、4 题，作业报告为第 5 题，已申报自选课题的同学需完成第 6 题。

1. 单选题 (15 分)
2. 计算题 (15 分)
3. 完成非线性分类器的程序代码 (30 分)
4. 训练/测试/可视化 (30 分)
5. 撰写作业报告 (10 分)
6. 汇报自选课题进度 (70 分) *

理论部分

1 单选题 (15 分)

1.1 在 Pytorch 中，如果想要定义一个新的神经网络层，应该继承下列哪个类？

- (A) torch.autograd.Layer
- (B) torch.nn.Module
- (C) torch.layers.Base

1.2 以下代码在执行顺序上有什么逻辑错误？

```
1 optimizer.zero_grad()
2 loss = loss_fn(model(input), target)
3 optimizer.step()
4 loss.backward()
```

- (A) `optimizer.step()` 应该在调用 `loss.backward()` 后调用
- (B) 应该在计算损失 `loss` 之前调用 `optimizer.zero_grad()` 清空优化器中保存的模型参数梯度
- (C) `optimizer.step()` 应该在 `optimizer.zero_grad()` 之后立即调用

1.3 ReLU (Rectified Linear Unit) 激活函数定义如下：

$$y = \max(0, x) \quad (1)$$

以下关于编程实现 ReLU 的说法正确的是：

- (A) ReLU 函数在 x 为正数时可能会导致梯度爆炸
- (B) ReLU 函数在 x 为负数时的导数为 0，可能导致部分神经元“死亡”
- (C) ReLU 函数在所有范围内的输出值都可能大于 1

1.4 考虑使用复合的函数来作为激活函数，如下：

$$y = f(g(z)) + t(z) \quad (2)$$

在其误差反向传播过程如下，其中正确的是：

- (A) $\frac{\partial L}{\partial z} = \frac{\partial L}{\partial y} * (f'(g(z))g'(z) + t'(z))$
- (B) $\frac{\partial L}{\partial z} = \frac{\partial L}{\partial y} * (f'(z)g'(z) + t'(z))$
- (C) $\frac{\partial L}{\partial z} = \frac{\partial L}{\partial y} * (f'(y)g'(y) + t'(y))$
- (D) $\frac{\partial L}{\partial z} = \frac{\partial L}{\partial y} * (f'(y)g'(z) + t'(z))$

1.5 设标量 $z = \mathbf{x}^T B \mathbf{y}$, 其中 $\mathbf{x} \in \mathbb{R}^{m \times 1}$, $B \in \mathbb{R}^{m \times p}$, $\mathbf{y} \in \mathbb{R}^{p \times 1}$, 现求标量 z 对向量 \mathbf{y} 的偏导数 $\frac{\partial z}{\partial \mathbf{y}}$ 为:

(A) $\mathbf{x}^T B$

(B) $B^T \mathbf{x}$

(C) $B \mathbf{x}$

(D) $\mathbf{x} B^T$

2 计算题 (15 分)

2.1 设隐含层为 $\mathbf{z} = \mathbf{W}^T \mathbf{x} + \mathbf{b}$, 其中 $\mathbf{x} \in R^{(m \times 1)}$, $\mathbf{z} \in R^{(n \times 1)}$, $\mathbf{W} \in R^{(m \times n)}$, $\mathbf{b} \in R^{(n \times 1)}$ 均为已知, 其激活函数如下:

$$\mathbf{y} = \delta(\mathbf{z}) = \tanh(\mathbf{z})$$

\tanh 表示双曲正切函数。若训练过程中的目标函数为 L , 且已知 L 对 \mathbf{y} 的导数 $\frac{\partial L}{\partial \mathbf{y}} = [\frac{\partial L}{\partial y_1}, \frac{\partial L}{\partial y_2}, \dots, \frac{\partial L}{\partial y_n}]^T$ 和 $\mathbf{y} = [y_1, y_2, \dots, y_n]^T$ 的值。

2.1.1 请使用 \mathbf{y} 表示出 $\frac{\partial \mathbf{y}^T}{\partial \mathbf{z}}$, 这里的 \mathbf{y}^T 为行向量。

2.1.2 请使用 \mathbf{y} 和 $\frac{\partial L}{\partial \mathbf{y}}$ 表示 $\frac{\partial L}{\partial \mathbf{x}}$, $\frac{\partial L}{\partial \mathbf{W}}$, $\frac{\partial L}{\partial \mathbf{b}}$ 。

提示: $\frac{\partial L}{\partial \mathbf{x}}$, $\frac{\partial L}{\partial \mathbf{W}}$, $\frac{\partial L}{\partial \mathbf{b}}$ 与 $\mathbf{x}, \mathbf{W}, \mathbf{b}$ 具有相同维度。

编程部分

编程部分包括第 3, 4 题。

3 完成非线性分类器的程序代码 (30 分)

使用多层感知机完成识别英文字符图像 (非线性分类) 任务, 在本任务中, 我们将使用 MLP 识别英文字符图像, 即输入一张图像, 模型输出识别结果。请注意图像样本包括大小写英文字母, 识别时不区分大小写形式, 即大写字母 “A” 和小写字母 “a” 都对应于一个类别, 输出时转为大写字母 “A” 显示。

程序清单如下:

文件或目录	说明	注意事项
hw1.zip	作业 1 程序压缩包	解压可以得到下列文件
\data	存放本次作业所用数据集	请勿修改
\saved_models	存放本次作业保存模型	请勿修改
activations.py	激活函数定义	需要完成代码
losses.py	损失函数定义	需要完成代码
network.py	神经网络结构定义	需要完成代码
recognition.py	非线性分类程序	需要完成代码

需要完成的代码清单如下：每处需要完成的地方都有代码提示和步骤提示。

activations.py 文件中待完成内容：

序号	行号	内容	说明
TODO 1	67-97	完成 Sigmoid 激活函数	请参照 activations.py 中实现的 tanh 激活函数完成 Sigmoid 激活函数的前向过程和反向传播过程
TODO 2	99-135	完成 ReLU 激活函数	请参照 activations.py 中实现的 tanh 激活函数完成 ReLU 激活函数的前向过程和反向传播过程

losses.py 文件中待完成内容：

序号	行号	内容	说明
TODO 1	56-文件末	完成交叉熵损失函数	请参照 losses.py 中实现的 MSE 损失函数完成交叉熵损失函数的前向过程和反向传播过程，注意在前向过程中要使用改进版的 softmax 计算方式

network.py 文件中待完成内容：

序号	行号	内容	说明
TODO 1	103	初始化模型的父类	注意一个网络模型要在定义模型结构之前先对父类 nn.Module 进行初始化
TODO 2	109-137	完成 MLP 网络结构	注意要根据层数的不同分两种情况讨论
TODO 3	147	完成模型的前向过程	对定义好的网络结构进行合理的调用

recognition.py 文件中待完成内容：

序号	行号	内容	说明
TODO 1	76	完成 List-Dataset	根据注释完成内容
TODO 2	145-170	计算 loss 和训练网络	按照 step 1-6 完成相关内容
TODO 3	209-226	测试模型得到前向计算结果	按照 step 1-5 完成相关内容
TODO 4	243-246	完成对图像数据的预处理	需要与 ListDataset 类中的预处理一致
TODO 5	325-341	完成对 MLP 模型的创建	根据输入数据的维度确定合适的网络结构参数，并根据需要将模型放置在恰当的设备上

4 训练/测试/可视化 (30 分)

(1) 使用默认配置对模型进行训练和测试。其中训练和测试的运行命令如下：

1) 训练模型，使用默认配置的命令如下：

```
python recognition.py --mode train
```

2) 测试模型，使用默认配置的命令如下：

```
python recognition.py --mode test
```

注意：请打开命令行终端，在终端中输入上述命令。

(2) 调整优化器为 Adam 优化器，修改其他参数如下，观察网络训练、验证和测试的性能。

```
python recognition.py --mode train --hsize 64 --lr 2e-3 --optim_type adam
--momentum 0 --weight_decay 0.1
```

注意：由于该分类的任务训练数据比较小，在增加隐含层层数时可能会导致性能下降，同时在该任务中 ReLU 为较合适的激活函数，在选择其他激活函数时可能会有严重的性能下降。

(3) 使用训练好的模型预测图像类别：请选择你认为效果最好的一个模型，对新的图像进行识别（设置 mode 为 predict）。可以使用我们提供的 predict01.png 和 predict02.png，也可以自行收集更多图像进行测试。请把

使用模型的情况、输入样本的预测结果写入作业报告。

使用训练好的模型预测输入样本的类别，使用命令例如：

```
python recognition.py --mode predict --im_path  
data/character_classification/new_images/predict02.png
```

当程序运行训练 (train) 和测试过程可视化显示 loss 曲线图片时，在图片显示窗口工具栏有保存图片的按钮，可手动保存图片。手动关闭图片窗口后，程序可以继续运行至结束。可视化截图需要附在作业报告中。

5 撰写作业报告 (10 分)

将 hw1 目录和作业报告打包为一个文件 (例如 *.zip) 提交到网络学堂。作业报告中包括选择题答案，计算题的解题步骤及答案、任务 3、4 运行结果及分析，本次作业遇到的问题及解决方法，对本次作业的意见及建议。如果大家使用大模型来完成作业，请在作业中说明，并指出使用大模型过程中出现的问题。推荐同学们使用随作业发布的 LaTeX 模板 HW1-template.zip 完成作业报告，LaTeX 模板可以使用 <https://www.overleaf.com> 打开。

6 自选课题进度汇报 (70 分) *

请已申报自选课题的同学，完成简短的自选课题工作进度汇报，例如，文献阅读、或者研究方案设计、或者原型系统搭建及实验结果等内容。

关于作业迟交的说明：由于平时作业计入总评成绩，希望同学们能按时提交作业。若有特殊原因不能按时提交，请在提交截止时间之前给本次作业责任助教发 Email 说明情况并给出预计提交作业的时间。对于未能按时说明原因的迟交作业，将酌情扣分。

本次作业责任助教为刘芳甫 (Email: liuff23@mails.tsinghua.edu.cn)。