清华大学电子工程系

媒体与认知 课堂 2

2021-2022 学年春季学期

作业 2

郭中贺 2022 年 4 月 3 日

理论部分

1 单选题（15 分）

1.1 D

1.2 C

1.3 B

1.4 C

1.5 D

2 计算题（15 分）

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

$$\mathbf{y}=\tanh(\mathbf{z})=\frac{e^{\mathbf{z}}-e^{-\mathbf{z}}}{e^{\mathbf{z}}+e^{-\mathbf{z}}}$$

若训练过程中的目标函数为 L，且已知 L 对 y 的导数 $\frac{\partial L}{\partial \mathbf{y}}=[\frac{\partial L}{\partial y_1},\frac{\partial L}{\partial y_2},...,\frac{\partial L}{\partial y_n}]$ 和 $\mathbf{y}=[y_1,y_2,...,y_n]$ 的值。

2.1.1 请使用 y 表示出 $\frac{\partial \mathbf{y}}{\partial \mathbf{z}}$

经过计算可得到

$$\frac{\partial y_i}{\partial z_j}=\begin{cases}0, & i\neq j\\ \frac{4}{\left(e^{z_j}+e^{-z_j}\right)^2}, & i=j\end{cases}$$

另一方面

$$y_i^2=1-\frac{4}{\left(e^{z_i}+e^{-z_i}\right)^2}$$

因此，可求得 $\frac{\partial y_i}{\partial z_i}=1-y_i^2$

2.1.2 请使用 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 x,W,b 具有相同维度。

$$\frac{\partial L}{\partial \mathbf{x}} = \sum_{i=1}^{n} \frac{\partial L}{\partial z_i} * \frac{\partial z_i}{\partial \mathbf{x}} = \frac{\partial L}{\partial \mathbf{z}} \cdot \mathbf{W}$$

其中

$$\frac{\partial L}{\partial z_i} = \frac{\partial L}{\partial y_i} * \frac{\partial y_i}{\partial z_i} = \frac{\partial L}{\partial y_i} * (1 - y_i^2)$$

因此

$$\frac{\partial L}{\partial \mathbf{z}} = \frac{\partial L}{\partial \mathbf{y}} * (1 - \mathbf{y}^2)$$

$$\frac{\partial L}{\partial \mathbf{x}} = \left(\frac{\partial L}{\partial \mathbf{y}} * (1 - \mathbf{y}^2) \right) \cdot \mathbf{W}$$

上式中的 * 和平方均代表向量中的对应元素运算，· 表示矩阵（向量）乘法. 同理可得:

$$\frac{\partial L}{\partial \mathbf{W}} = \sum_{i=1}^{n} \frac{\partial L}{\partial z_i} * \frac{\partial z_i}{\partial \mathbf{W}} = \left(\frac{\partial L}{\partial \mathbf{z}} \right)^T \cdot \mathbf{x} = \left(\frac{\partial L}{\partial \mathbf{y}} * (1 - \mathbf{y}^2) \right)^T \cdot \mathbf{x}$$

$$\frac{\partial L}{\partial \mathbf{b}} = \sum_{i=1}^{n} \frac{\partial L}{\partial z_i} * \frac{\partial z_i}{\partial \mathbf{b}} = \frac{\partial L}{\partial \mathbf{z}} = \frac{\partial L}{\partial \mathbf{y}} * (1 - \mathbf{y}^2)$$

编程部分

3 编程作业报告

3.1 训练模型

3.1.1 SGD 优化器默认参数

运行 python recognition.py –mode train 训练模型后，控制台输出结果与 loss 变化曲线分别如图 1、2 所示。

结果分析：由图 3 曲线可知，一开始训练的 loss 较高，约为 14。随着训练轮数的增加，loss 逐渐下降，最终稳定到 0.015 左右。另一方面，由图 2 输出，可知模型在验证集上的准确率为 70% 左右，并且随着训练轮数增加准确率有小幅度上升。但是由于数据集数量过少，因此最终训练后准确率并不太高。

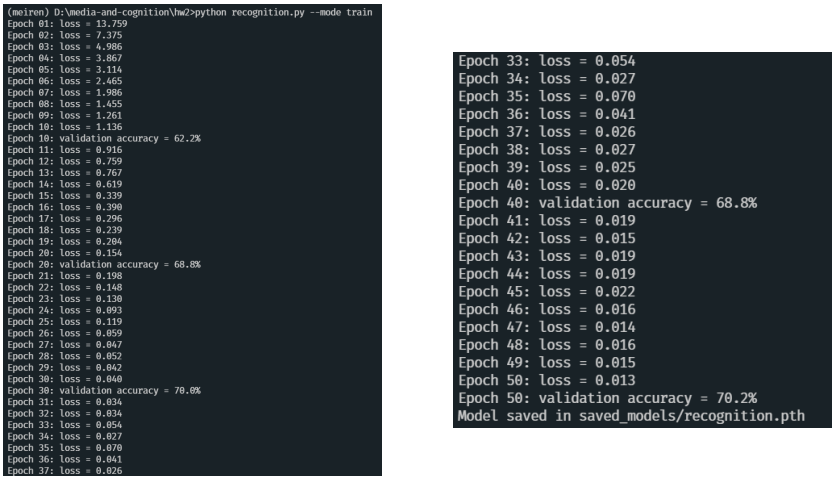

图 1: cmd 输出结果

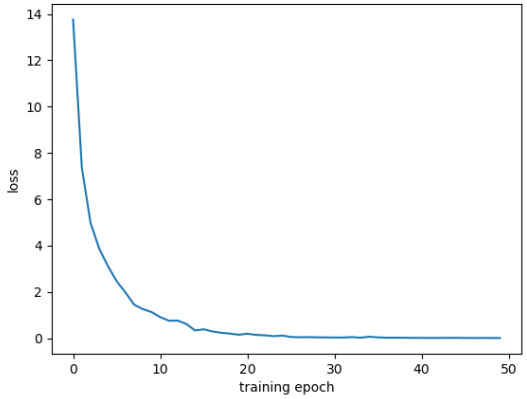

图 2: loss 曲线

3.1.2 Adam 优化器并修改参数

运行 python recognition.py –mode train –hsize 64 –lr 2e-3 –optim_type adam –momentum 0 –weight_decay 0.1 训练模型后，控制台输出结果与 loss 变化曲线分别如图 3、4 所示。

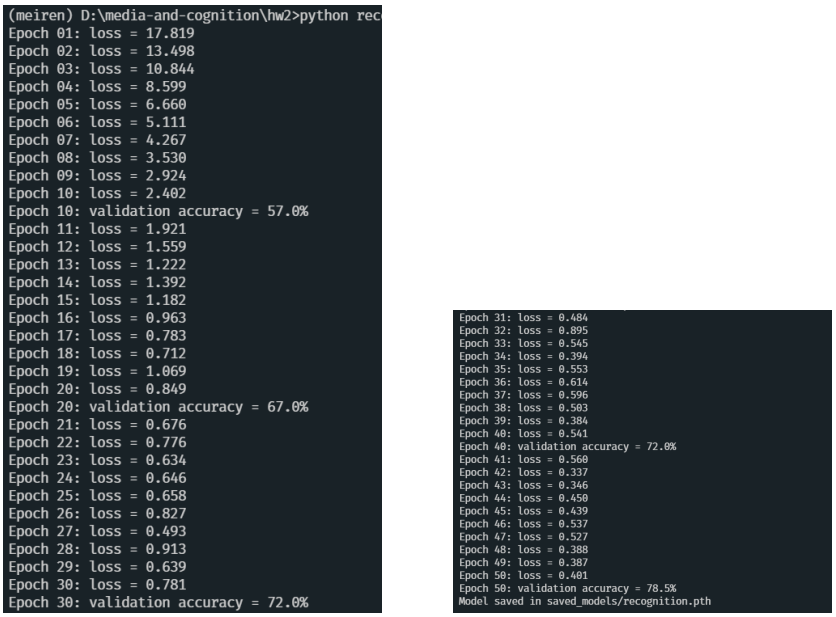

图 3: cmd 输出结果

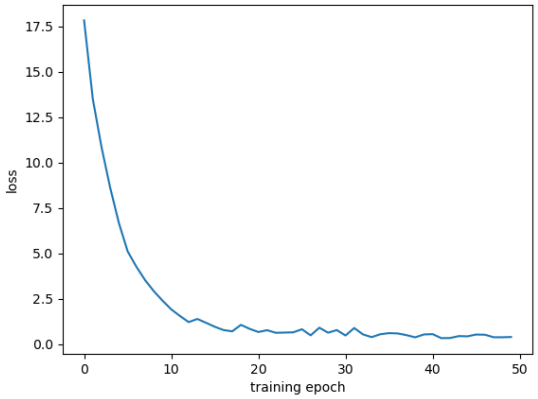

图 4: loss 曲线

通过对比可以看出，使用 adam 优化器后，训练的整体 loss 略有上升，但是训练的最终准确率有一定幅度的提高，提高到了 78% 左右，因此，最终决定使用 adam 优化器和相应参数。

3.2 测试模型及可视化

运行 python recognition.py –mode test 验证模型，控制台输出结果如图 5 所示。

```
(meiren) D:\media-and-cognition\hw2>python recognition.py --mode test
[Info] Load model from saved_models/recognition.pth
C:\Users\siwen\anaconda3\envs\meiren\lib\site-packages\sklearn\manifo
.0 to 'auto' in 1.2.
  FutureWarning,
C:\Users\siwen\anaconda3\envs\meiren\lib\site-packages\sklearn\manifo
 standard deviation of PC1 equal to 1e-4 in 1.2. This will ensure bet
  FutureWarning,
[Info] Test accuracy = 81.0%
```

图 5: cmd 输出结果

可见使用训练 50 轮之后的模型，在测试集上进行测试，其准确率更高达到了 81.0%。可见，该训练模型并不只能识别用于训练和验证的数据，具有一定的泛用性。
可视化分类结果如图 6 所示。

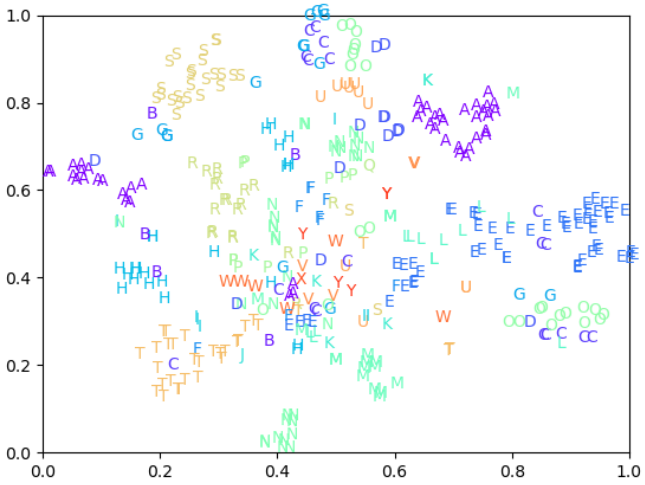

图 6: 可视化结果

通过分析比较，我们可以看出：部分字母可以较好地分辨出来，比如左上角的 S、R，右侧的 E、O，左下角的 T 等。但是也有很多字母分辨性能较差，如 C 和 G 容易混淆。其主要原因，还是训练的数据集样本过少，不能很好地训练出各个字母的特征，模型也就无法分辨出不同字母。

3.3 预测图像类别

此步骤使用 adam 优化器训练后的模型 recognition.pth，输入样本为提供的 predict01.png（字母 A）和 predict02.png（字母 B）。预测结果分别如下。图 7 为 predict01.png 预测结果，图 8 为 predict02.png 预测结果。

```
(meiren) D:\media-and-cognition\hw2>python recognition.py --mode predict --im_path data/character_classification/new_images/predict01.png
[Info] Load model from saved_models/recognition.pth
Prediction: A
```

图 7: predict01.png 预测结果

```
(meiren) D:\media-and-cognition\hw2>python recognition.py --mode predict --im_path data/character_classification/new_images/predict02.png
[Info] Load model from saved_models/recognition.pth
Prediction: B
```

图 8: predict02.png 预测结果

可见这两张图片的预测结果十分准确。

3.4 本次作业遇到的问题及解决方法

无。

3.5 对本次作业的意见及建议

本次作业也较为简单，通过代码注释中的一步一步教程，我们能很快了解下一步该做什么，该如何实现。我的建议是也可以试着让同学们自己编写一个自定义激活函数以达到练习的目的。