

作业 1

毕嘉仪

2024 年 4 月 9 日

理论部分

1 单选题 (15 分)

1.1 B

1.2 A

1.3 B

1.4 A

1.5 B

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。

解.

$$\frac{\partial \mathbf{y}^T}{\partial \mathbf{z}}_{n \times n} = \begin{bmatrix} \frac{\partial y_1}{\partial z_1} & \cdots & \frac{\partial y_n}{\partial z_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial y_1}{\partial z_n} & \cdots & \frac{\partial y_n}{\partial z_n} \end{bmatrix}$$

当 $i \neq j$, 易知 $\frac{\partial y_i}{\partial z_j} = 0$

当 $i = j$,

$$\tanh' z_i = 1 - \tanh^2 z_i$$

$$\begin{aligned}
 z_i &= \operatorname{arctanh} y_i \\
 \therefore \tanh' z_i &= 1 - y_i^2 \\
 \therefore \frac{\partial \mathbf{y}^T}{\partial \mathbf{z}}_{n \times n} &= \operatorname{diag}(1 - y_i^2) = \begin{bmatrix} 1 - y_1^2 & & & \\ & 1 - y_2^2 & & \\ & & \ddots & \\ & & & 1 - y_n^2 \end{bmatrix}
 \end{aligned}$$

□

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。

解. 对 $\frac{\partial L}{\partial \mathbf{x}}$, 由链式法则:

$$\frac{\partial L}{\partial \mathbf{x}}_{m \times 1} = \frac{\partial \mathbf{z}^T}{\partial \mathbf{x}}_{m \times n} \frac{\partial \mathbf{y}^T}{\partial \mathbf{z}}_{n \times n} \frac{\partial L}{\partial \mathbf{y}}_{n \times 1} = \mathbf{W} \operatorname{diag}(1 - y_i^2) \frac{\partial L}{\partial \mathbf{y}}$$

对于 $\frac{\partial L}{\partial \mathbf{W}}$, 先计算

$$\begin{aligned}
 \frac{\partial \mathbf{z}_i}{\partial \mathbf{W}}_{m \times n} &= \begin{bmatrix} \frac{\partial z_i}{\partial W_{11}} & \cdots & \frac{\partial z_i}{\partial W_{1n}} \\ \vdots & \ddots & \vdots \\ \frac{\partial z_i}{\partial W_{m1}} & \cdots & \frac{\partial z_i}{\partial W_{mn}} \end{bmatrix} = \begin{bmatrix} 0 & \cdots & \frac{\partial z_i}{\partial W_{1i}} & \cdots & 0 \\ \vdots & & \vdots & & \vdots \\ 0 & \cdots & \frac{\partial z_i}{\partial W_{mi}} & \cdots & 0 \end{bmatrix} \\
 &= \begin{bmatrix} x_1 \\ \vdots \\ x_m \end{bmatrix} = \begin{bmatrix} 0 & \cdots & \mathbf{x} & \cdots & 0 \end{bmatrix}
 \end{aligned}$$

$$\begin{aligned}
 \therefore \frac{\partial L}{\partial \mathbf{W}}_{m \times n} &= \left(\frac{\partial z_1}{\partial \mathbf{W}}_{m \times n} \frac{\partial \mathbf{y}^T}{\partial \mathbf{z}}_{n \times n} \frac{\partial L}{\partial \mathbf{y}}_{n \times 1}, \frac{\partial z_2}{\partial \mathbf{W}}_{m \times n} \frac{\partial \mathbf{y}^T}{\partial \mathbf{z}}_{n \times n} \frac{\partial L}{\partial \mathbf{y}}_{n \times 1}, \dots, \frac{\partial z_n}{\partial \mathbf{W}}_{m \times n} \frac{\partial \mathbf{y}^T}{\partial \mathbf{z}}_{n \times n} \frac{\partial L}{\partial \mathbf{y}}_{n \times 1} \right)_{m \times n} \\
 &= \left(\mathbf{x} \operatorname{diag}(1 - y_i^2) \frac{\partial L}{\partial \mathbf{y}}, \dots, \mathbf{x} \operatorname{diag}(1 - y_i^2) \frac{\partial L}{\partial \mathbf{y}} \right)
 \end{aligned}$$

对于 $\frac{\partial L}{\partial \mathbf{b}}$, 先计算

$$\begin{aligned}
 \frac{\partial \mathbf{z}^T}{\partial \mathbf{b}}_{n \times n} &= \begin{bmatrix} \frac{\partial z_1}{\partial b_1} & \cdots & \frac{\partial z_n}{\partial b_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial z_1}{\partial b_n} & \cdots & \frac{\partial z_n}{\partial b_n} \end{bmatrix} = \mathbf{I}_n \\
 \therefore \frac{\partial L}{\partial \mathbf{b}}_{n \times 1} &= \frac{\partial \mathbf{z}^T}{\partial \mathbf{b}}_{n \times n} \frac{\partial \mathbf{y}^T}{\partial \mathbf{z}}_{n \times n} \frac{\partial L}{\partial \mathbf{y}}_{n \times 1} = \operatorname{diag}(1 - y_i^2) \frac{\partial L}{\partial \mathbf{y}}
 \end{aligned}$$

□

编程部分

3 编程作业报告

(1) 使用默认配置对模型进行训练和测试。

1) 训练模型：

input:

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

output:

```
1 Epoch 01: loss = inf
2 Epoch 02: loss = inf
3 Epoch 03: loss = inf
4 Epoch 04: loss = 4.573
5 Epoch 05: loss = 3.583
6 Epoch 06: loss = 2.447
7 Epoch 07: loss = 1.830
8 Epoch 08: loss = 1.818
9 Epoch 09: loss = 1.374
10 Epoch 10: loss = 1.055
11 Epoch 10: validation accuracy = 61.3%
12 Epoch 11: loss = 0.835
13 Epoch 12: loss = 0.653
14 Epoch 13: loss = 0.586
15 Epoch 14: loss = 0.396
16 Epoch 15: loss = 0.254
17 Epoch 16: loss = 0.240
18 Epoch 17: loss = 0.187
19 Epoch 18: loss = 0.157
20 Epoch 19: loss = 0.112
21 Epoch 20: loss = 0.132
22 Epoch 20: validation accuracy = 62.7%
23 Epoch 21: loss = 0.096
24 Epoch 22: loss = 0.061
25 Epoch 23: loss = 0.060
26 Epoch 24: loss = 0.056
27 Epoch 25: loss = 0.048
28 Epoch 26: loss = 0.054
29 Epoch 27: loss = 0.038
```

```
30 Epoch 28: loss = 0.039
31 Epoch 29: loss = 0.050
32 Epoch 30: loss = 0.052
33 Epoch 30: validation accuracy = 64.0%
34 Epoch 31: loss = 0.036
35 Epoch 32: loss = 0.032
36 Epoch 33: loss = 0.028
37 Epoch 34: loss = 0.026
38 Epoch 35: loss = 0.025
39 Epoch 36: loss = 0.026
40 Epoch 37: loss = 0.024
41 Epoch 38: loss = 0.024
42 Epoch 39: loss = 0.021
43 Epoch 40: loss = 0.027
44 Epoch 40: validation accuracy = 65.2%
45 Epoch 41: loss = 0.021
46 Epoch 42: loss = 0.020
47 Epoch 43: loss = 0.017
48 Epoch 44: loss = 0.019
49 Epoch 45: loss = 0.016
50 Epoch 46: loss = 0.015
51 Epoch 47: loss = 0.016
52 Epoch 48: loss = 0.015
53 Epoch 49: loss = 0.014
54 Epoch 50: loss = 0.015
55 Epoch 50: validation accuracy = 65.5%
56 Model saved in saved_models/recognition.pth
```

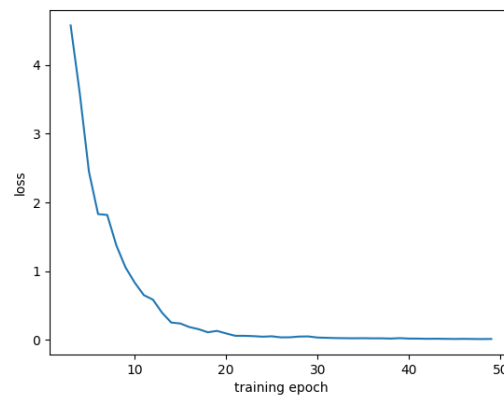


图 1

2) 测试模型:

input:

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

output:

```
1 [Info] Load model from saved_models/recognition.pth
2 [Info] Test accuracy = 67.0%
```

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

1) 训练模型

input:

```
1 python recognition.py --mode train --hsize 64 --lr 2e
  -3 --optim_type adam --momentum 0 --weight_decay
  0.1 --model_path .\saved_models\ADAM_opt.pth
```

output:

```
1 Epoch 01: loss = inf
2 Epoch 02: loss = inf
3 Epoch 03: loss = inf
4 Epoch 04: loss = inf
5 Epoch 05: loss = inf
6 Epoch 06: loss = inf
7 Epoch 07: loss = inf
8 Epoch 08: loss = inf
9 Epoch 09: loss = inf
10 Epoch 10: loss = 3.594
11 Epoch 10: validation accuracy = 59.5%
12 Epoch 11: loss = 2.529
13 Epoch 12: loss = 2.012
14 Epoch 13: loss = 1.733
15 Epoch 14: loss = 1.475
16 Epoch 15: loss = 1.215
17 Epoch 16: loss = 1.047
18 Epoch 17: loss = 1.050
19 Epoch 18: loss = 1.053
20 Epoch 19: loss = 0.776
21 Epoch 20: loss = 0.761
22 Epoch 20: validation accuracy = 65.0%
23 Epoch 21: loss = 0.768
```

```
24 Epoch 22: loss = 0.737
25 Epoch 23: loss = 0.683
26 Epoch 24: loss = 1.086
27 Epoch 25: loss = 0.613
28 Epoch 26: loss = 0.630
29 Epoch 27: loss = 0.670
30 Epoch 28: loss = 0.562
31 Epoch 29: loss = 0.683
32 Epoch 30: loss = 0.488
33 Epoch 30: validation accuracy = 72.0%
34 Epoch 31: loss = 0.597
35 Epoch 32: loss = 0.858
36 Epoch 33: loss = 0.559
37 Epoch 34: loss = 0.635
38 Epoch 35: loss = 0.462
39 Epoch 36: loss = 0.551
40 Epoch 37: loss = 0.470
41 Epoch 38: loss = 0.403
42 Epoch 39: loss = 0.412
43 Epoch 40: loss = 0.524
44 Epoch 40: validation accuracy = 75.5%
45 Epoch 41: loss = 0.544
46 Epoch 42: loss = 0.359
47 Epoch 43: loss = 0.611
48 Epoch 44: loss = 0.331
49 Epoch 45: loss = 0.334
50 Epoch 46: loss = 0.487
51 Epoch 47: loss = 0.448
52 Epoch 48: loss = 0.334
53 Epoch 49: loss = 0.313
54 Epoch 50: loss = 0.464
55 Epoch 50: validation accuracy = 78.2%
56 Model saved in .\saved_models\ADAM_opt.pth
```

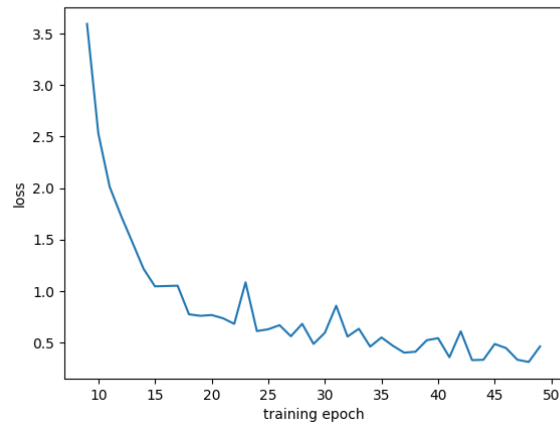


图 2

2) 测试模型

input:

```
1 python recognition.py --mode test --hsize 64 --
  model_path .\saved_models\ADAM_opt.pth
```

output:

```
1 [Info] Load model from .\saved_models\ADAM_opt.pth
2 [Info] Test accuracy = 80.0%
```

(3) 选择效果最好的模型，对新的图像进行识别。

1) 选择模型:

经过一些尝试，以下模型预测准确值为 85.5%，遂选取该模型进行预测：

input:

```
1 python recognition.py --mode train --hsize 64 --lr 1e
  -3 --optim_type adam --momentum 15 --weight_decay
  0.1 --epoch 100 --model_path .\saved_models\
  ADAM_opt_trial_lr1e-3_momentum15_epoch100.pth
```

output:

```
1 Epoch 01: loss = inf
2 Epoch 02: loss = inf
3 Epoch 03: loss = inf
```

```
4 Epoch 04: loss = inf
5 Epoch 05: loss = inf
6 Epoch 06: loss = inf
7 Epoch 07: loss = inf
8 Epoch 08: loss = inf
9 Epoch 09: loss = inf
10 Epoch 10: loss = inf
11 Epoch 10: validation accuracy = 41.0%
12 Epoch 11: loss = inf
13 Epoch 12: loss = inf
14 Epoch 13: loss = inf
15 Epoch 14: loss = inf
16 Epoch 15: loss = inf
17 Epoch 16: loss = inf
18 Epoch 17: loss = 3.093
19 Epoch 18: loss = 2.537
20 Epoch 19: loss = 2.197
21 Epoch 20: loss = 1.908
22 Epoch 20: validation accuracy = 56.8%
23 Epoch 21: loss = 1.571
24 Epoch 22: loss = 1.438
25 Epoch 23: loss = 1.177
26 Epoch 24: loss = 1.031
27 Epoch 25: loss = 0.942
28 Epoch 26: loss = 0.913
29 Epoch 27: loss = 0.707
30 Epoch 28: loss = 0.613
31 Epoch 29: loss = 0.622
32 Epoch 30: loss = 0.481
33 Epoch 30: validation accuracy = 61.3%
34 Epoch 31: loss = 0.441
35 Epoch 32: loss = 0.407
36 Epoch 33: loss = 0.492
37 Epoch 34: loss = 0.408
38 Epoch 35: loss = 0.369
39 Epoch 36: loss = 0.385
40 Epoch 37: loss = 0.316
41 Epoch 38: loss = 0.472
42 Epoch 39: loss = 0.345
43 Epoch 40: loss = 0.351
44 Epoch 40: validation accuracy = 65.5%
45 Epoch 41: loss = 0.318
```



```
46 Epoch 42: loss = 0.313
47 Epoch 43: loss = 0.302
48 Epoch 44: loss = 0.284
49 Epoch 45: loss = 0.196
50 Epoch 46: loss = 0.380
51 Epoch 47: loss = 0.268
52 Epoch 48: loss = 0.158
53 Epoch 49: loss = 0.239
54 Epoch 50: loss = 0.315
55 Epoch 50: validation accuracy = 70.2%
56 Epoch 51: loss = 0.338
57 Epoch 52: loss = 0.300
58 Epoch 53: loss = 0.257
59 Epoch 54: loss = 0.227
60 Epoch 55: loss = 0.216
61 Epoch 56: loss = 0.219
62 Epoch 57: loss = 0.282
63 Epoch 58: loss = 0.276
64 Epoch 59: loss = 0.260
65 Epoch 60: loss = 0.270
66 Epoch 60: validation accuracy = 77.8%
67 Epoch 61: loss = 0.301
68 Epoch 62: loss = 0.218
69 Epoch 63: loss = 0.241
70 Epoch 64: loss = 0.280
71 Epoch 65: loss = 0.300
72 Epoch 66: loss = 0.299
73 Epoch 67: loss = 0.283
74 Epoch 68: loss = 0.304
75 Epoch 69: loss = 0.313
76 Epoch 70: loss = 0.276
77 Epoch 70: validation accuracy = 81.0%
78 Epoch 71: loss = 0.334
79 Epoch 72: loss = 0.300
80 Epoch 73: loss = 0.302
81 Epoch 74: loss = 0.287
82 Epoch 75: loss = 0.303
83 Epoch 76: loss = 0.301
84 Epoch 77: loss = 0.314
85 Epoch 78: loss = 0.320
86 Epoch 79: loss = 0.319
87 Epoch 80: loss = 0.327
```

```
88 Epoch 80: validation accuracy = 76.8%
89 Epoch 81: loss = 0.327
90 Epoch 82: loss = 0.341
91 Epoch 83: loss = 0.336
92 Epoch 84: loss = 0.324
93 Epoch 85: loss = 0.324
94 Epoch 86: loss = 0.357
95 Epoch 87: loss = 0.370
96 Epoch 88: loss = 0.352
97 Epoch 89: loss = 0.371
98 Epoch 90: loss = 0.343
99 Epoch 90: validation accuracy = 83.5%
100 Epoch 91: loss = 0.327
101 Epoch 92: loss = 0.343
102 Epoch 93: loss = 0.353
103 Epoch 94: loss = 0.347
104 Epoch 95: loss = 0.361
105 Epoch 96: loss = 0.355
106 Epoch 97: loss = 0.343
107 Epoch 98: loss = 0.357
108 Epoch 99: loss = 0.404
109 Epoch 100: loss = 0.352
110 Epoch 100: validation accuracy = 82.8%
111 Model saved in .\saved_models\ADAM_opt_trial_lr1e-3
    _momentum15_epoch100.pth
```

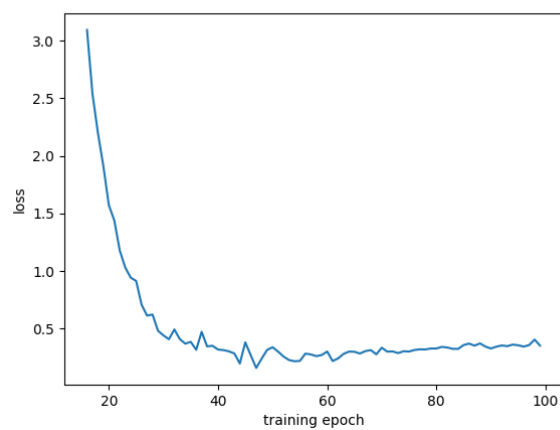


图 3

input:

```
1 python recognition.py --mode test --hsize 64 --  
   model_path .\saved_models\ADAM_opt_trial_lr1e-3_  
   _momentum15_epoch100.pth
```

output:

```
1 [Info] Load model from .\saved_models\  
   ADAM_opt_trial_lr1e-3_momentum15_epoch100.pth  
2 [Info] Test accuracy = 85.5%
```

2) 进行预测:

input respectively:

```
1 python recognition.py --mode predict --im_path data/  
   character_classification/new_images/1_W.jpg --  
   model_path .\saved_models\ADAM_opt_trial_lr1e-3_  
   _momentum15_epoch100.pth --hsize 64  
2 python recognition.py --mode predict --im_path data/  
   character_classification/new_images/2_I.jpg --  
   model_path .\saved_models\ADAM_opt_trial_lr1e-3_  
   _momentum15_epoch100.pth --hsize 64  
3 python recognition.py --mode predict --im_path data/  
   character_classification/new_images/3_S.jpg --  
   model_path .\saved_models\ADAM_opt_trial_lr1e-3_  
   _momentum15_epoch100.pth --hsize 64  
4 python recognition.py --mode predict --im_path data/  
   character_classification/new_images/4_H.jpg --  
   model_path .\saved_models\ADAM_opt_trial_lr1e-3_  
   _momentum15_epoch100.pth --hsize 64  
5 python recognition.py --mode predict --im_path data/  
   character_classification/new_images/5_Y.jpg --  
   model_path .\saved_models\ADAM_opt_trial_lr1e-3_  
   _momentum15_epoch100.pth --hsize 64  
6 python recognition.py --mode predict --im_path data/  
   character_classification/new_images/6_O.jpg --  
   model_path .\saved_models\ADAM_opt_trial_lr1e-3_  
   _momentum15_epoch100.pth --hsize 64  
7 python recognition.py --mode predict --im_path data/  
   character_classification/new_images/7_U.jpg --  
   model_path .\saved_models\ADAM_opt_trial_lr1e-3_  
   _momentum15_epoch100.pth --hsize 64  
8 python recognition.py --mode predict --im_path data/  
   character_classification/new_images/8_A.jpg --
```

```
model_path .\saved_models\ADAM_opt_trial_lr1e-3
momentum15_epoch100.pth --hsize 64
9 python recognition.py --mode predict --im_path data/
character_classification/new_images/9_N.jpg --
model_path .\saved_models\ADAM_opt_trial_lr1e-3
momentum15_epoch100.pth --hsize 64
10 python recognition.py --mode predict --im_path data/
character_classification/new_images/10_I.jpg --
model_path .\saved_models\ADAM_opt_trial_lr1e-3
momentum15_epoch100.pth --hsize 64
11 python recognition.py --mode predict --im_path data/
character_classification/new_images/11_C.jpg --
model_path .\saved_models\ADAM_opt_trial_lr1e-3
momentum15_epoch100.pth --hsize 64
12 python recognition.py --mode predict --im_path data/
character_classification/new_images/12_E.jpg --
model_path .\saved_models\ADAM_opt_trial_lr1e-3
momentum15_epoch100.pth --hsize 64
13 python recognition.py --mode predict --im_path data/
character_classification/new_images/13_D.jpg --
model_path .\saved_models\ADAM_opt_trial_lr1e-3
momentum15_epoch100.pth --hsize 64
14 python recognition.py --mode predict --im_path data/
character_classification/new_images/14_A.jpg --
model_path .\saved_models\ADAM_opt_trial_lr1e-3
momentum15_epoch100.pth --hsize 64
15 python recognition.py --mode predict --im_path data/
character_classification/new_images/15_Y.jpg --
model_path .\saved_models\ADAM_opt_trial_lr1e-3
momentum15_epoch100.pth --hsize 64
```

prediction results:



(4) 遇到的问题及解决方法。

- 1) 在将样本图片转化为一维向量而使用 `tensor.flatten(input, start_dim=0)` 函数时，没有注意到 `tensor` 变量 `imgs` 的第 0 维是 `batch_size`，导致后续计算 `tensor` 维数错误。因为一开始就对 `tensor.flatten(input, start_dim=0)` 用法不确定，多留意了一下，侥幸发现了这个问题。
- 2) 使用 `tensor.sum()` 时总是搞不清楚叠加的维度究竟是哪一维。借助浏览器的力量解决了这一问题。
- 3) 没有注意到 `opt` 的 `hsize` 参数 `default=32`，而模型训练的时候用的是 64；进行预测时又没有指定 `hsize`，导致 `tensor` 维度不对应。在重新查看了 `run command` 之后发现了这个小错。

(5) 建议：也许可以多介绍一下目录里各个类型的文件都是用来做什么的？第一次搭模型，很多文件类型都没见过，不知道它们有什么用。