

**COMP9444 Neural Networks and Deep Learning**

**Assignment 1**

**Term 1, 2025**

Submitted by

zID: xxxxxxxx

Name: xxxxxxxxxx

I declare that:

This assessment item is entirely my own original work, except where I have acknowledged use of source material [such as books, journal articles, other published material, the Internet, and the work of other student/s or any other person/s.

This assessment item has not been submitted for assessment for academic credit in this, or any other course, at UNSW or elsewhere.

I understand that:

 The assessor of this assessment item may, for the purpose of assessing this item, reproduce this assessment item and provide a copy to another member of the University.

The assessor may communicate a copy of this assessment item to a plagiarism checking service (which may then retain a copy of the assessment item on its database for the purpose of future plagiarism checking).

I certify that I have read and understood the University Rules in respect of Student Academic Misconduct.

Name, student id, and date

**Part 1: Japanese Character Recognition**

1. Answer question 1

Train Epoch: 10 [57600/60000 (96%)] Loss: 0.671839

<class 'numpy.ndarray'>

[[763. 5. 8. 12. 29. 65. 2. 64. 30. 22.]

[ 7. 670. 110. 17. 28. 23. 57. 14. 24. 50.]

[ 9. 61. 692. 28. 25. 19. 47. 35. 46. 38.]

[ 3. 35. 57. 762. 15. 55. 14. 18. 28. 13.]

[ 62. 53. 82. 20. 621. 19. 31. 34. 20. 58.]

[ 8. 29. 124. 17. 19. 722. 29. 9. 33. 10.]

[ 5. 25. 147. 10. 25. 26. 720. 20. 8. 14.]

[ 16. 30. 27. 12. 85. 15. 54. 623. 90. 48.]

[ 11. 39. 95. 41. 6. 32. 44. 6. 703. 23.]

[ 8. 52. 88. 3. 51. 30. 19. 28. 39. 682.]]

Test set: Average loss: 1.0092, Accuracy: 6958/10000 (70%)

1. Answer question 2

Train Epoch: 10 [0/60000 (0%)] Loss: 0.361453

Train Epoch: 10 [6400/60000 (11%)] Loss: 0.220010

Train Epoch: 10 [12800/60000 (21%)] Loss: 0.227565

Train Epoch: 10 [19200/60000 (32%)] Loss: 0.211923

Train Epoch: 10 [25600/60000 (43%)] Loss: 0.126040

Train Epoch: 10 [32000/60000 (53%)] Loss: 0.238420

Train Epoch: 10 [38400/60000 (64%)] Loss: 0.196619

Train Epoch: 10 [44800/60000 (75%)] Loss: 0.339821

Train Epoch: 10 [51200/60000 (85%)] Loss: 0.118984

Train Epoch: 10 [57600/60000 (96%)] Loss: 0.246220

<class 'numpy.ndarray'>

[[851. 3. 1. 6. 30. 31. 5. 39. 29. 5.]

[ 6. 809. 36. 4. 15. 10. 65. 4. 20. 31.]

[ 7. 10. 842. 37. 9. 21. 25. 10. 21. 18.]

[ 3. 7. 27. 921. 2. 14. 5. 4. 7. 10.]

[ 40. 31. 22. 5. 813. 5. 23. 19. 21. 21.]

[ 11. 10. 95. 10. 10. 822. 18. 3. 14. 7.]

[ 3. 12. 45. 10. 11. 4. 895. 9. 2. 9.]

[ 20. 11. 24. 6. 22. 8. 32. 822. 25. 30.]

[ 7. 25. 27. 43. 4. 8. 33. 2. 842. 9.]

[ 4. 15. 47. 5. 28. 6. 27. 11. 12. 845.]]

Test set: Average loss: 0.5072, Accuracy: 8462/10000 (85%)

(28×28×200)+200+(200×10)+10 =(784×200)+200+(2000)+10= 159010 parameters

1. Answer question 3

Train Epoch: 10 [0/60000 (0%)] Loss: 0.024336

Train Epoch: 10 [6400/60000 (11%)] Loss: 0.025435

Train Epoch: 10 [12800/60000 (21%)] Loss: 0.041802

Train Epoch: 10 [19200/60000 (32%)] Loss: 0.025455

Train Epoch: 10 [25600/60000 (43%)] Loss: 0.032418

Train Epoch: 10 [32000/60000 (53%)] Loss: 0.057490

Train Epoch: 10 [38400/60000 (64%)] Loss: 0.012455

Train Epoch: 10 [44800/60000 (75%)] Loss: 0.151626

Train Epoch: 10 [51200/60000 (85%)] Loss: 0.006380

Train Epoch: 10 [57600/60000 (96%)] Loss: 0.027990

<class 'numpy.ndarray'>

[[956. 2. 1. 0. 18. 2. 0. 13. 3. 5.]

[ 1. 921. 2. 0. 9. 2. 32. 7. 4. 22.]

[ 9. 11. 880. 30. 6. 13. 16. 10. 6. 19.]

[ 1. 1. 10. 957. 4. 7. 6. 2. 1. 11.]

[ 16. 10. 1. 5. 936. 1. 9. 5. 11. 6.]

[ 2. 10. 42. 4. 5. 893. 22. 8. 3. 11.]

[ 4. 4. 11. 2. 7. 2. 963. 4. 0. 3.]

[ 2. 3. 0. 0. 2. 0. 4. 967. 6. 16.]

[ 3. 12. 6. 1. 4. 1. 4. 5. 958. 6.]

[ 9. 2. 11. 2. 6. 0. 5. 9. 11. 945.]]

Test set: Average loss: 0.2466, Accuracy: 9376/10000 (94%)

Conv1 Parameters=(3×3×1×32)+32=288+32=320

Conv2 Parameters=(3×3×32×64)+64=18,432+64=18,496

FC1 Parameters=(64×7×7×128)+128=401,408+128=401,536

FC2 Parameters=(128×10)+10=1,280+10=1,290

Total Parameters=320+18,496+401,536+1,290=421,642

1. Answer question 4

**Part 2: Multi-Layer Perceptron**

1. Answer question 1

图表, 散点图

AI 生成的内容可能不正确。图表, 散点图

AI 生成的内容可能不正确。图表, 散点图

AI 生成的内容可能不正确。

图表, 散点图

AI 生成的内容可能不正确。

图表, 散点图

AI 生成的内容可能不正确。图表, 散点图

AI 生成的内容可能不正确。图表, 散点图

AI 生成的内容可能不正确。

Above is hid7\_0 – hid7\_6

图表, 漏斗图

AI 生成的内容可能不正确。

This is out\_7

Overall accuracy is 100%:

文本

AI 生成的内容可能不正确。

1. Answer question 2

H(w1x1+w2x2+b)

w1x1+w2x2+b=0

h1: −x−y+1=0 -x - y + 1 = 0 −x−y+1=0 → y=−x+1 y = -x + 1 y=−x+1

h2: x+y−1=0 x + y - 1 = 0 x+y−1=0 → y=−x+1 y = -x + 1 y=−x+1

h3: x−y−1=0 x - y - 1 = 0 x−y−1=0 → y=x−1 y = x - 1 y=x−1

h4: −x+y−1=0 -x + y - 1 = 0 −x+y−1=0 → y=x+1 y = x + 1 y=x+1

**Hidden nodes:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| X | Y | H1 | H2 | H3 | H4 | output | target |
| -1 | -1 | 1 | 0 | 0 | 0 | 1.5 | 1 |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

图示

AI 生成的内容可能不正确。

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| X1 | Hidden 1 | 1 |  |
| X2 | Hidden 1 | 1 |  |
| X1 | Hidden 2 | 1 |  |
| X2 | Hidden 2 | -1 |  |
| X1 | Hidden 3 | -1 |  |
| X2 | Hidden 3 | 1 |  |
| X1 | Hidden 4 | -1 |  |
| X2 | Hidden 4 | -1 |  |

1. Answer question 3

**Part 3: Hidden Unit Dynamics for Recurrent Networks**

1. Answer question 1
2. Answer question 2
3. Answer question 3
4. Answer question 4