**Part1**

**Question1**

Based on the question1, we should use the linear function and log softmax function to design the neural network. After running the source code, we can finally get the accuracy, which is 6966/10000. After doing the calculation, the accuracy is around 70%. Here is the picture of the result.

Furthermore, we can also get the confusion matrix as showing below. Here is the result of confusion matrix.

[[766. 5. 8. 13. 30. 64. 3. 62. 30. 19.]

[ 7. 671. 107. 18. 27. 23. 58. 13. 27. 49.]

[ 5. 58. 691. 27. 28. 21. 47. 37. 46. 40.]

[ 4. 33. 61. 759. 14. 58. 15. 19. 25. 12.]

[ 58. 50. 82. 22. 626. 19. 32. 37. 20. 54.]

[ 8. 28. 128. 17. 20. 722. 26. 8. 33. 10.]

[ 4. 24. 147. 11. 24. 24. 722. 21. 10. 13.]

[ 17. 29. 29. 11. 83. 16. 52. 624. 91. 48.]

[ 9. 35. 95. 41. 8. 31. 46. 7. 707. 21.]

[ 8. 50. 87. 2. 53. 31. 17. 33. 41. 678.]]

**Question2**

Based on the question2, we should using linear function, tanh function and log softmax function to design a neural network with 2 layers. In this case, we can use a function called nn.sequential to wrap these function up. This can make this process more easier. After running the source code, we can easily find the accuracy is 85%, which is 8457/10000. Here is the picture of the result.

Furthermore, we can also get the confusion matrix as showing below. Here is the result of confusion matrix.

[[847. 7. 3. 6. 31. 34. 4. 39. 24. 5.]

[ 4. 824. 27. 2. 17. 15. 59. 7. 17. 28.]

[ 7. 16. 826. 50. 12. 19. 27. 12. 14. 17.]

[ 5. 12. 25. 919. 2. 18. 7. 2. 4. 6.]

[ 35. 34. 21. 4. 818. 10. 27. 18. 19. 14.]

[ 9. 24. 76. 8. 10. 832. 18. 1. 16. 6.]

[ 3. 16. 42. 9. 13. 7. 891. 8. 1. 10.]

[ 22. 13. 17. 3. 16. 9. 30. 841. 22. 27.]

[ 12. 31. 29. 60. 4. 9. 30. 3. 816. 6.]

[ 4. 21. 44. 3. 32. 5. 19. 20. 9. 843.]]

**Question3**