COMP9444 Neural Networks and Deep Learning

Assignment 1

Jiayang Li

z5337892

Part 1. Japanese Character Recognition

Question 1.

Final Confusion matrix =

[[765. 7. 7. 4. 59. 8. 5. 16. 11. 8.]

[5. 671. 65. 37. 51. 26. 23. 29. 37. 52.]

[8. 109. 689. 61. 84. 125. 150. 26. 94. 85.]

[12. 18. 26. 759. 21. 17. 10. 12. 42. 3.]

[31. 27. 27. 14. 623. 20. 27. 89. 6. 51.]

[65. 22. 21. 56. 20. 726. 25. 19. 32. 34.]

[2. 58. 45. 14. 33. 27. 719. 53. 45. 19.]

[61. 14. 37. 17. 34. 8. 21. 620. 6. 31.]

[32. 25. 44. 26. 19. 32. 9. 89. 704. 39.]

[19. 49. 39. 12. 56. 11. 11. 47. 23. 678.]]

Final → Test set: Average loss: 1.0107, **Accuracy:** 6954/10000 (70%)

Question 2.

Num_id = 500 **accuracy** = 85%

Num_id = 1000 **accuracy** = 85%

Num_id = 1500 **accuracy** = 84%

Num_id = 2000 **accuracy** = 84%

Num_id = 3000 **accuracy** = 84%

Final Confusion matrix =

[[850. 6. 7. 2. 37. 11. 3. 17. 11. 4.]

[5. 814. 13. 8. 31. 17. 14. 11. 28. 19.]

[2. 37. 831. 24. 16. 84. 40. 20. 29. 49.]

[6. 2. 46. 927. 7. 6. 10. 6. 55. 4.]

[26. 20. 10. 3. 823. 11. 16. 16. 4. 28.]

[36. 10. 19. 15. 6. 832. 6. 8. 8. 4.]

[2. 58. 26. 5. 28. 17. 894. 33. 28. 21.]

[42. 5. 13. 2. 14. 1. 7. 836. 2. 17.]

[28. 17. 20. 6. 23. 15. 2. 23. 830. 9.]

[3. 31. 15. 8. 15. 6. 8. 30. 5. 845.]]

Final -> **Test set: Average loss:** 0.4935, **Accuracy:** 8482/10000 (85%)

Question 3.

Training No	**Accuracy (%)**
1	93
2	93
3	93
4	93
5	93

Final Confusion Matrix =

[[939. 3. 11. 3. 19. 6. 5. 5. 3. 6.]

[7. 926. 11. 1. 14. 19. 10. 12. 22. 12.]

[3. 4. 894. 18. 4. 50. 21. 10. 15. 11.]

[1. 0. 36. 957. 3. 9. 3. 2. 11. 2.]

[28. 3. 2. 2. 912. 3. 7. 8. 5. 5.]

[3. 2. 10. 6. 4. 887. 3. 1. 5. 0.]

[2. 46. 19. 3. 17. 14. 946. 16. 10. 5.]

[11. 3. 3. 5. 10. 6. 2. 926. 4. 3.]

[2. 2. 7. 1. 9. 2. 1. 2. 922. 2.]

[4. 11. 7. 4. 8. 4. 2. 18. 3. 954.]]

Test set: Average loss: 0.2747, **Accuracy:** 9263/10000 (93%)

Question 4.

A.

The most accurate results are from Question 3 using convolutional neural network, as it is the best when it comes to classifying images because the convolutional layers extract more features from the images which then can be learned in fully connected layers in part of the network.

Conclusion: CNN network hits 93% accuracy consistently.

B.

Comparing between the three confusion matrixes, the diagonal values are increasing as we move from NetLin towards NetConv meaning that there is an increase of the performance in the model.

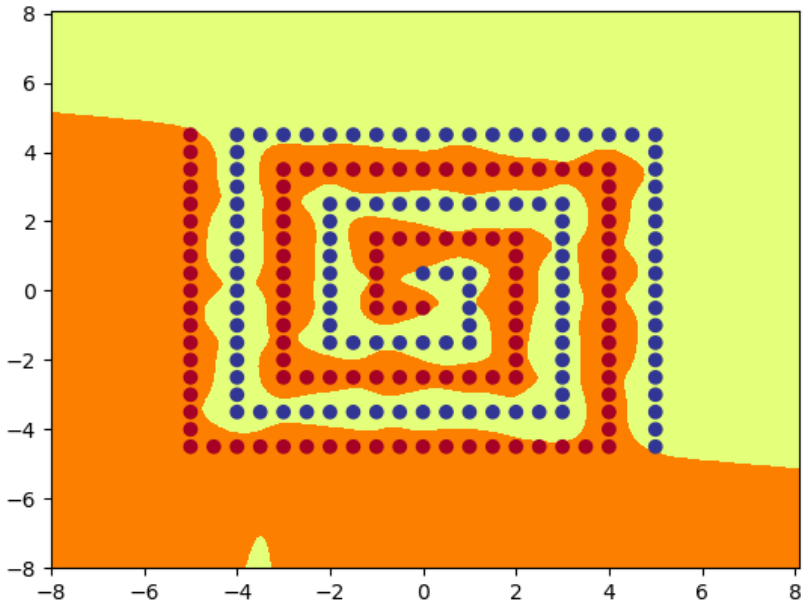
For the NetLin confusion matrix, the matrix has high values when it makes a prediction of 2 for target values of 1,3,4,5,6,8,9 characters so it is possible that the model will most likely make a error when making prediction for these characters.

Lastly for the confusion matrix for NetFull model, the diagonal values have increased and most likely the character of this model might classify wrong is 5 when the model is predicting 2.

Part 2. Rectangular Spirals Task

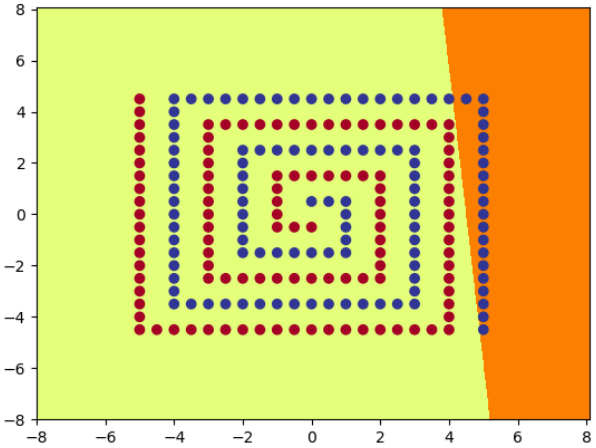
Question 3.

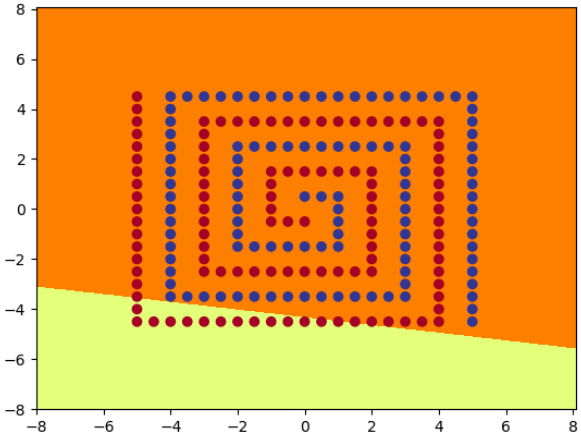
For this part, I have determined that I needed 19 hidden nodes to reach 100 accuracies. Averaging 4200 Epochs.

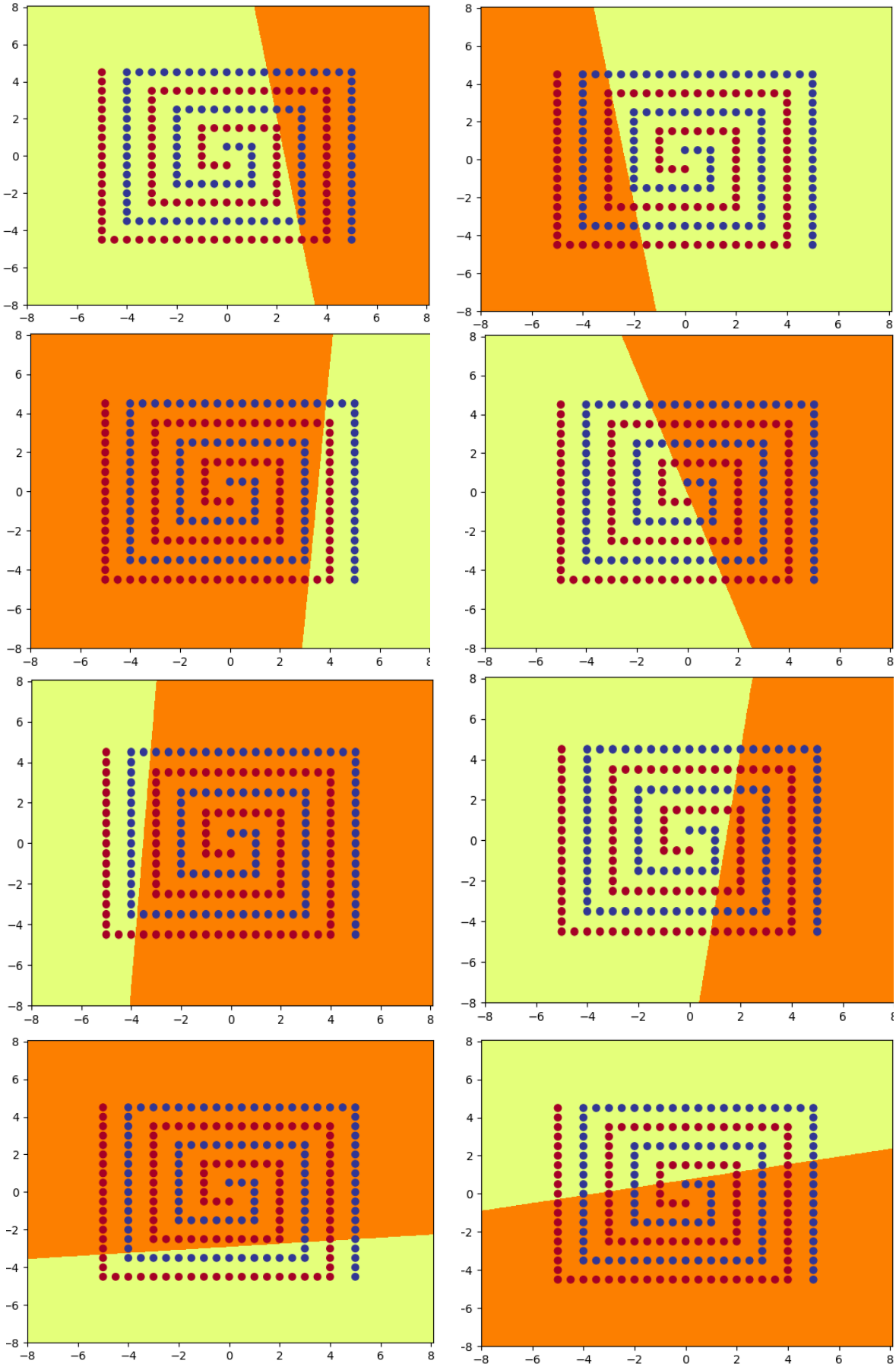


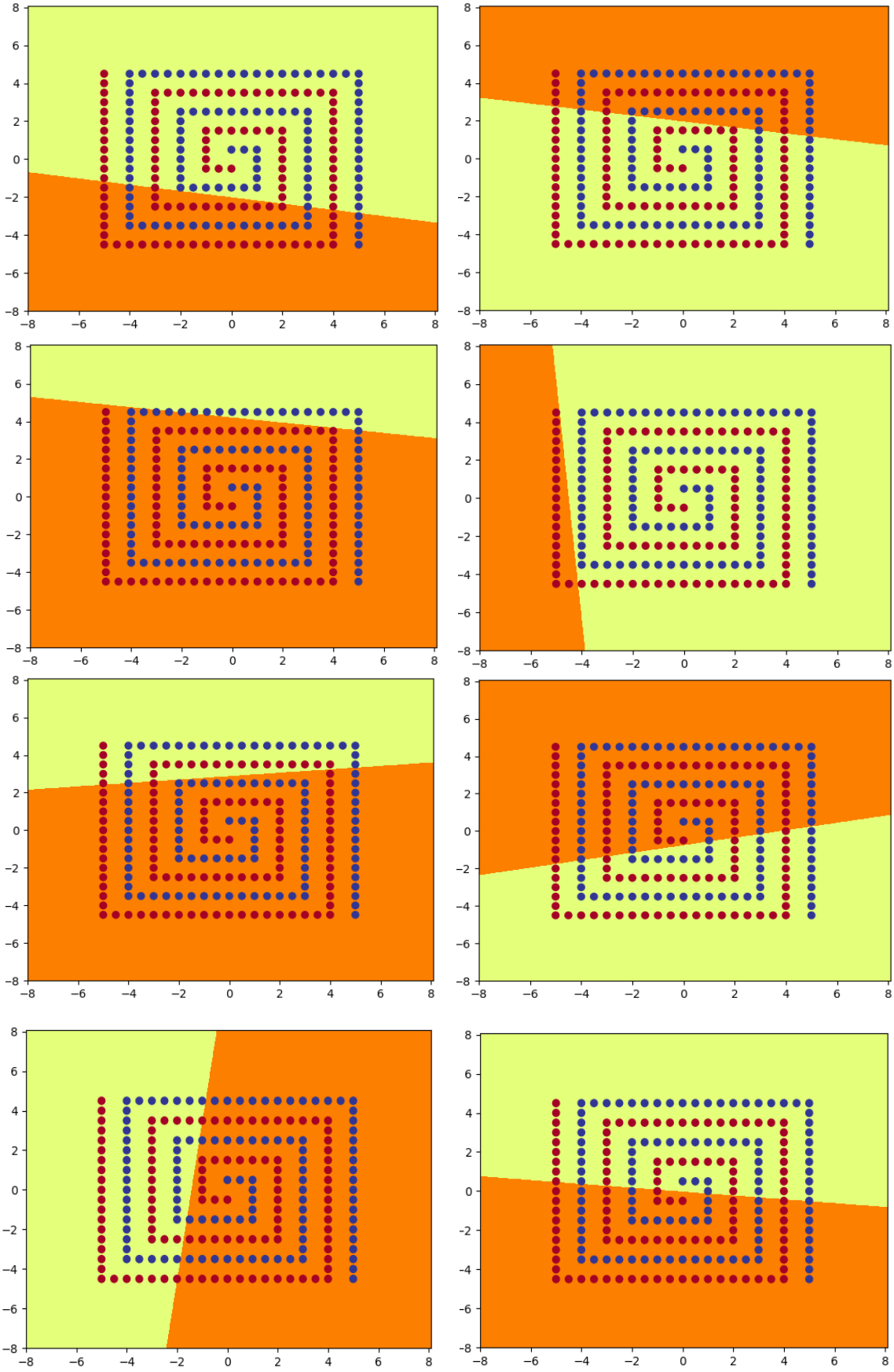
Output 1_19.png

All plots of hidden nodes – hid1_19_1.png to hid1_19_18.png



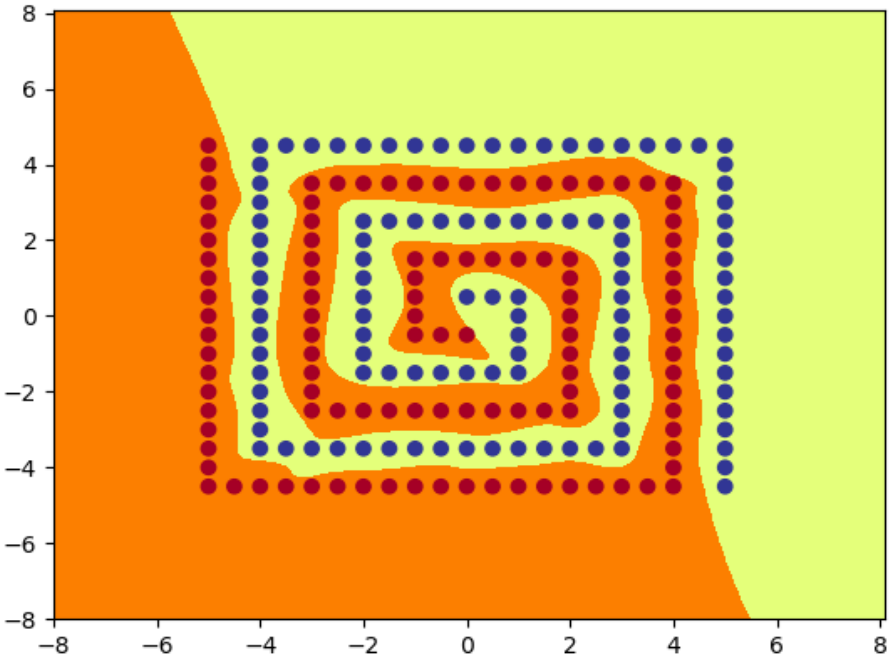






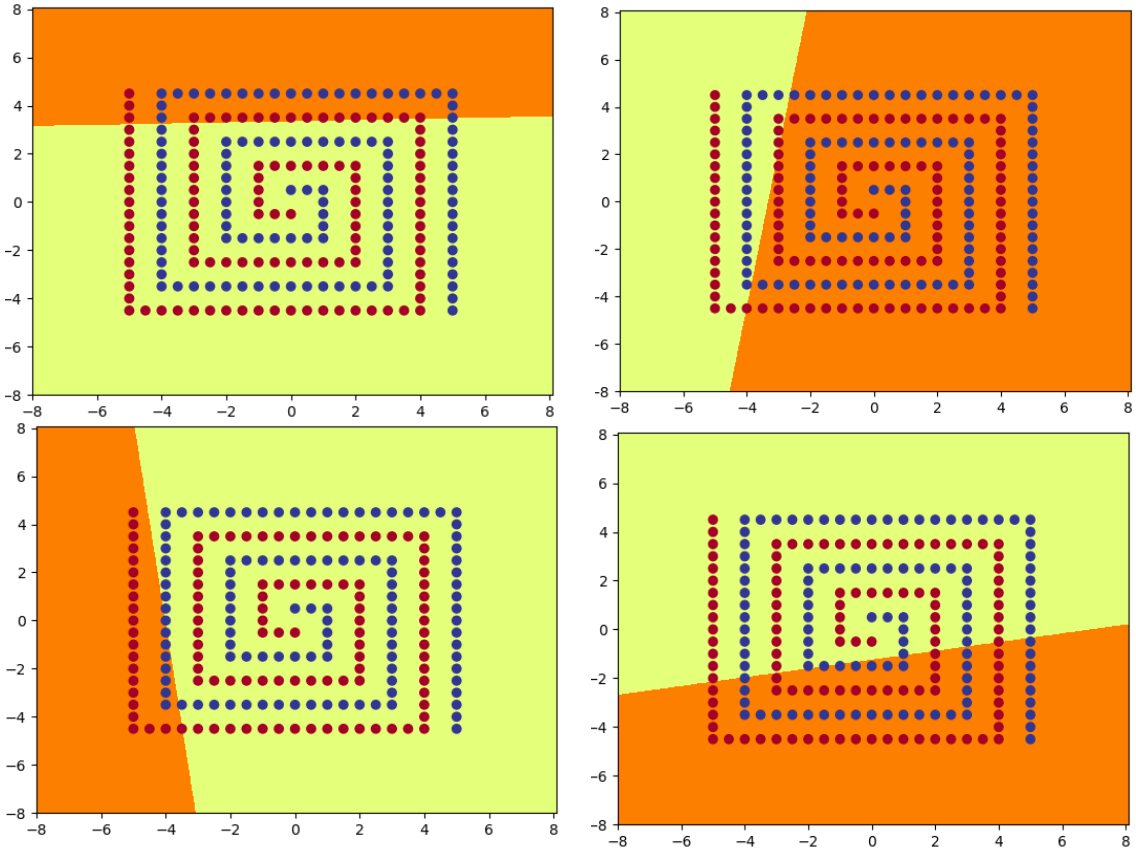
Question 4.

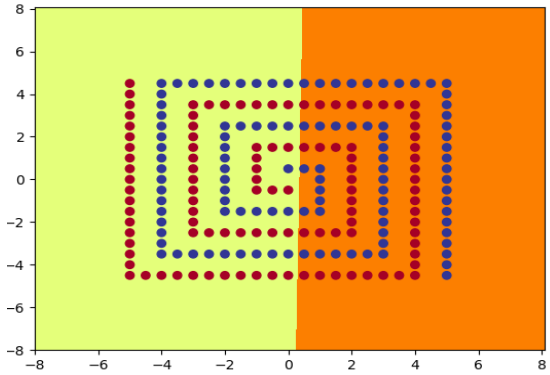
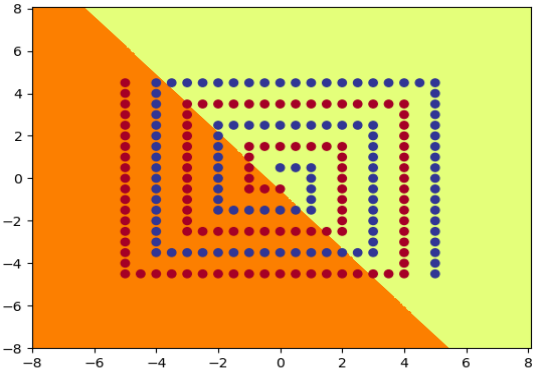
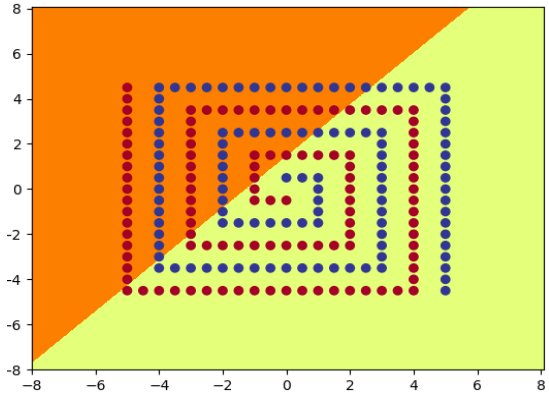
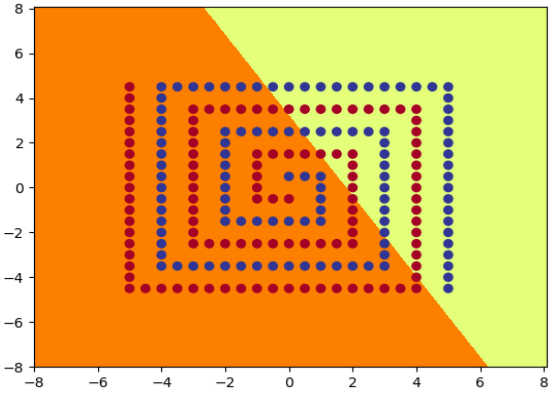
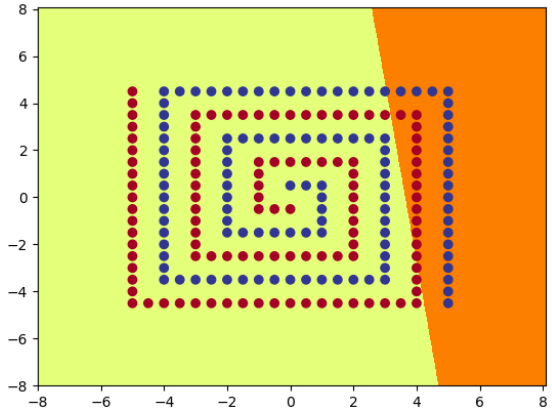
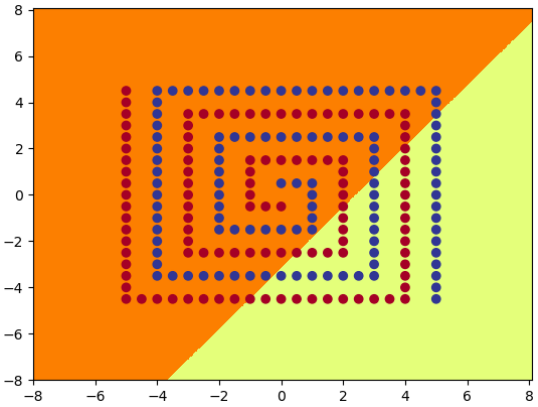
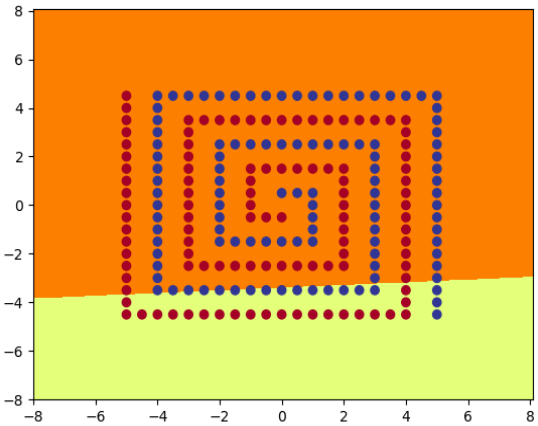
For this part, I have determined that I needed 12 hidden nodes to reach 100 accuracies. Averaging 6800 Epochs.



Out2_12.png

All plots of hidden nodes – hid2_12 between 1 – 11 png



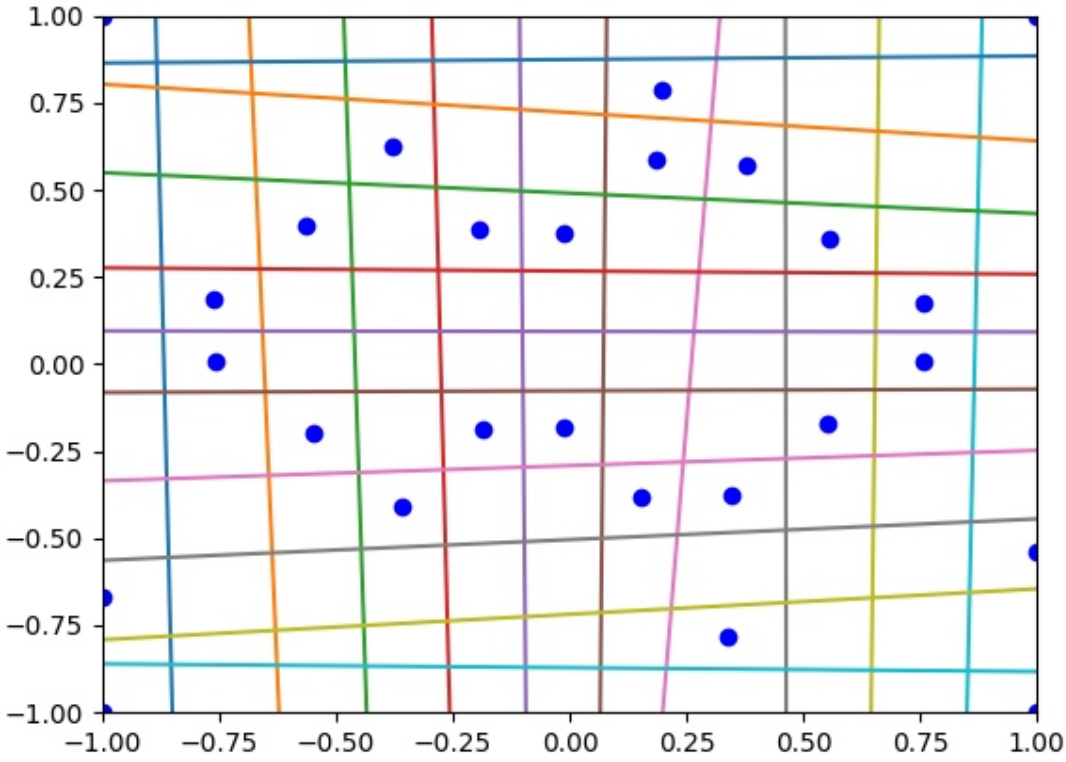
Question 5.

A. For the network module, the module is converting the coordinates and then sending the input to the network learning the spiral problem. The network is trying to learn different parts of the spiral patterns by learning the coordinates as the we examine the images of the hidden layers in the learning parts of the model for the spiral. These patterns then combined in the output layer to product the model.

B. As we see from the results that, layer 2 uses less hidden nodes than layer 1. As layer 1 requires 19 hidden nodes while layer 2 only needed 12 hidden nodes.

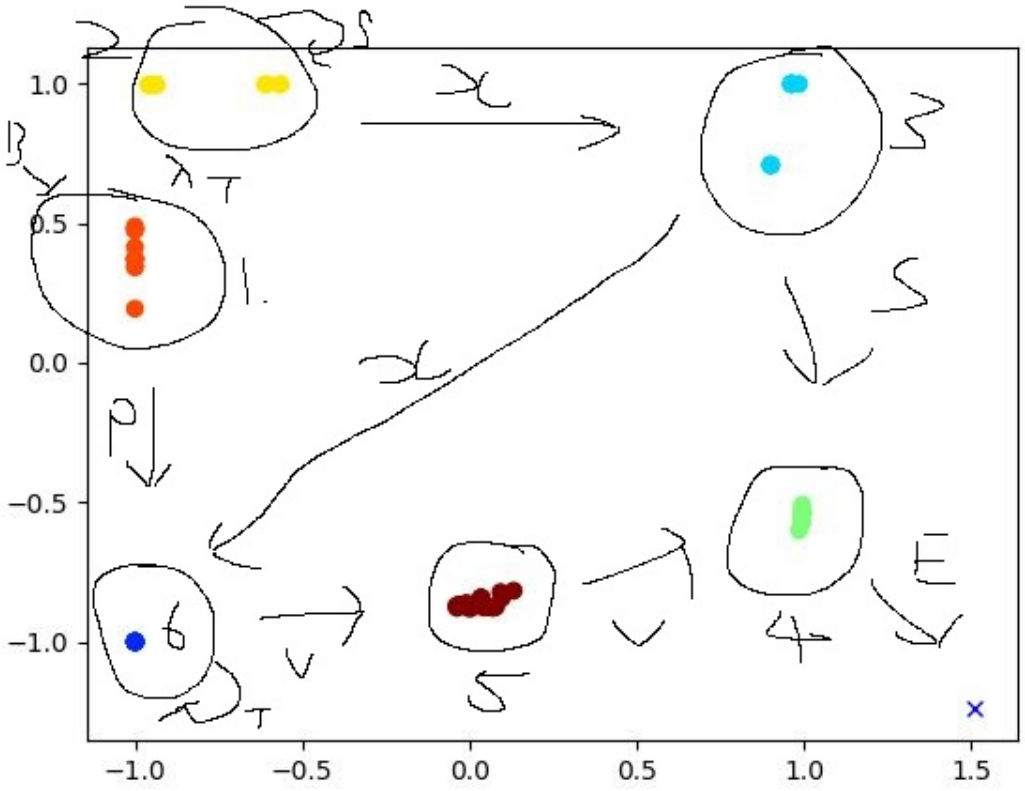
Part 3. Encoder Networks

Final Image

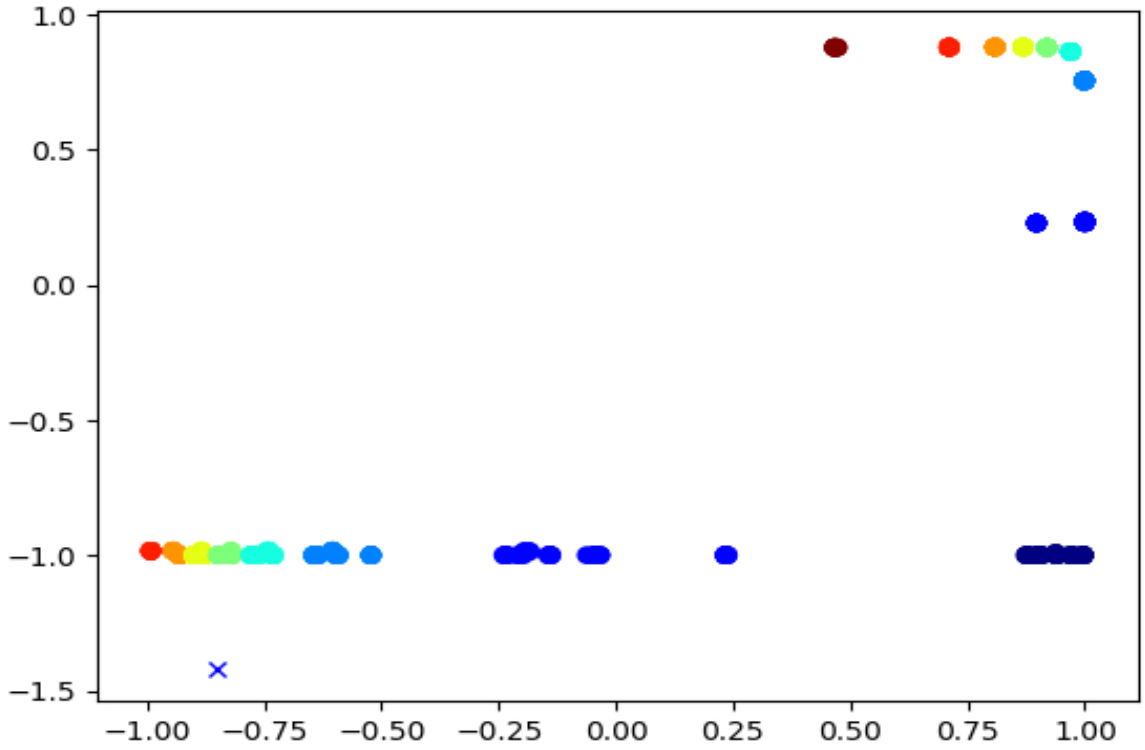


Part 4. Hidden Unit Dynamics for Recurrent Networks

Question 1.



Question 2.

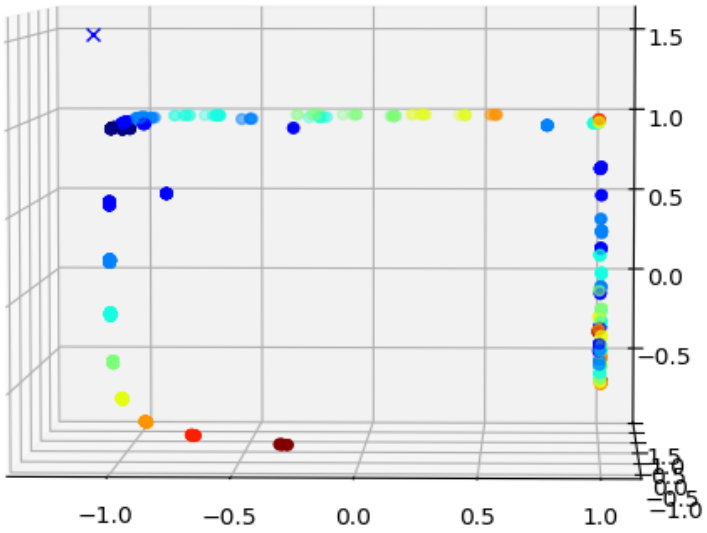


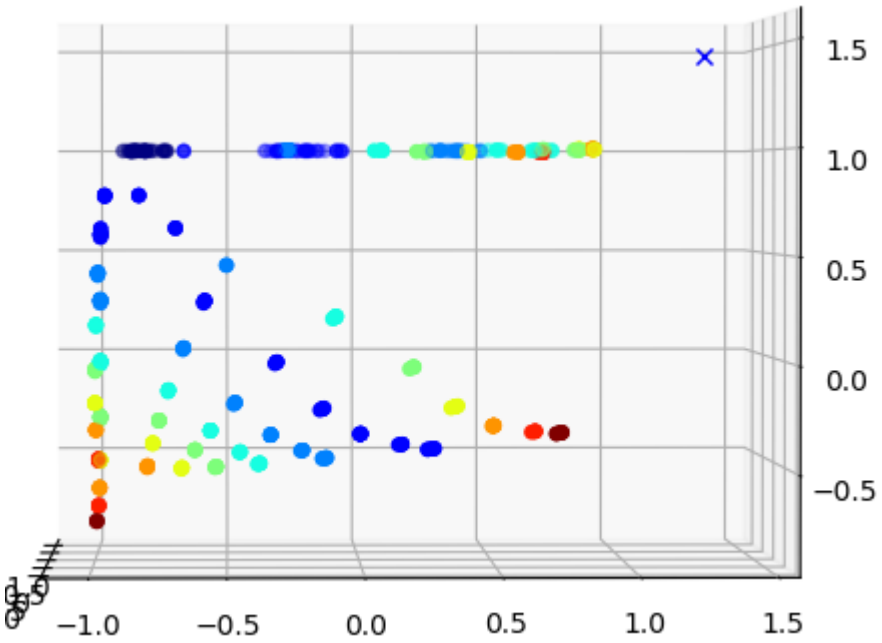
Question 3.

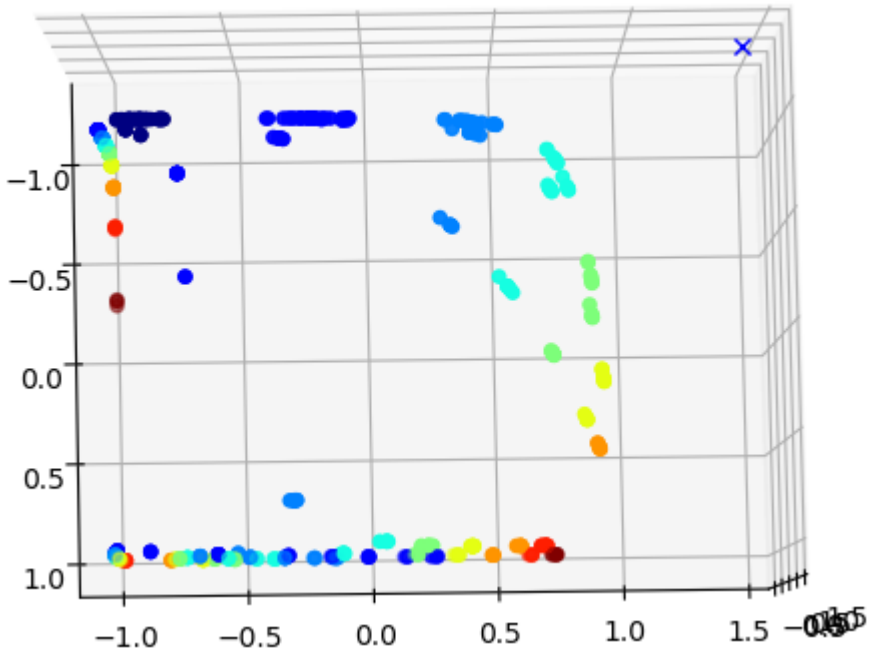
As every sequence of consecutive A's must be followed by an equal number of B's.

In this case, the network scans a sequence of the input characters one at a time and try to take each step to predict the next character in the sequence. So referencing Question 2, the prediction is possible as for anbn task the first b is not predictable but the subsequent B's are predictable.

Question 4.







Question 5.

This model produces a three-dimensional view as this SRN has three inputs, outputs, and hidden units. The SRN counts simultaneously counts down while another is counting up. This is similar to the principal of anbn. As A crosses to B there are one direction that is attracting via a fixed point and the other direction is repelling. Which is why the model is similar shaped to a star.

Question 6.

As the LSTM trains, there are some patterns being analyised.

As the numbers of epochs continues to train, the numbers for hidden activations and output probabilities moves closer towards 0. As postive numbers decrease to move closer to 0 while negative numbers increase to move closer to 0.

As the hidden nodes increase, there is a decrease in error rate and for final it eventually reaches to 0 as the epochs continues to be trained.