CS-449 HW1 report

Chen Si
Ding Zhang
Dong Shu
Yuchen Wang

1. (4.0points) Implement and train an MLP network as specified above using binary cross entropy as the loss function. Experiment with each of the four datasets to find the best number of nodes k in the hidden layer. For the best k for each dataset:

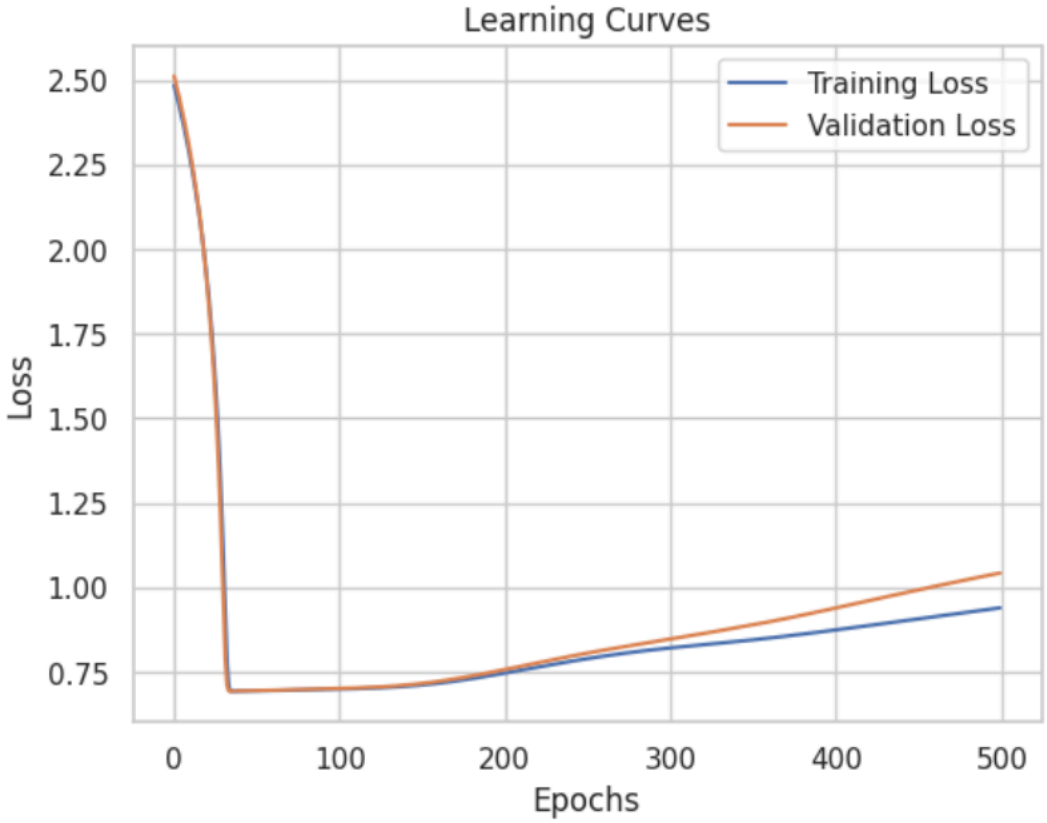
1. `Center_surround dataset:`
    - **a. list hyper-parameters used in the model:**
        - input_size=2
        - hidden_size=12
            - Choose best k (shown below)

```
Hidden Size: 5, Average Accuracy: 0.7057
Hidden Size: 6, Average Accuracy: 0.7063
Hidden Size: 7, Average Accuracy: 0.7015
Hidden Size: 8, Average Accuracy: 0.7038
Hidden Size: 9, Average Accuracy: 0.7012
Hidden Size: 10, Average Accuracy: 0.7020
Hidden Size: 11, Average Accuracy: 0.7063
Hidden Size: 12, Average Accuracy: 0.7077
Hidden Size: 13, Average Accuracy: 0.7052
Hidden Size: 14, Average Accuracy: 0.7020
Hidden Size: 15, Average Accuracy: 0.7032
Hidden Size: 16, Average Accuracy: 0.7055
Hidden Size: 17, Average Accuracy: 0.7042
Hidden Size: 18, Average Accuracy: 0.6985
Hidden Size: 19, Average Accuracy: 0.6602
Hidden Size: 20, Average Accuracy: 0.7002
Hidden Size: 21, Average Accuracy: 0.6590
Hidden Size: 22, Average Accuracy: 0.6440
Hidden Size: 23, Average Accuracy: 0.6273
Hidden Size: 24, Average Accuracy: 0.6053
Hidden Size: 25, Average Accuracy: 0.6235
Hidden Size: 26, Average Accuracy: 0.5713
Hidden Size: 27, Average Accuracy: 0.5247
Hidden Size: 28, Average Accuracy: 0.5595
Hidden Size: 29, Average Accuracy: 0.5367
Best Model - best k: 12, Accuracy: 0.7077
```

        - output_size=1
        - learning_rate=0.01
        - epochs=200

b. Plot the learning curves for training and validation loss as a function of training epochs:

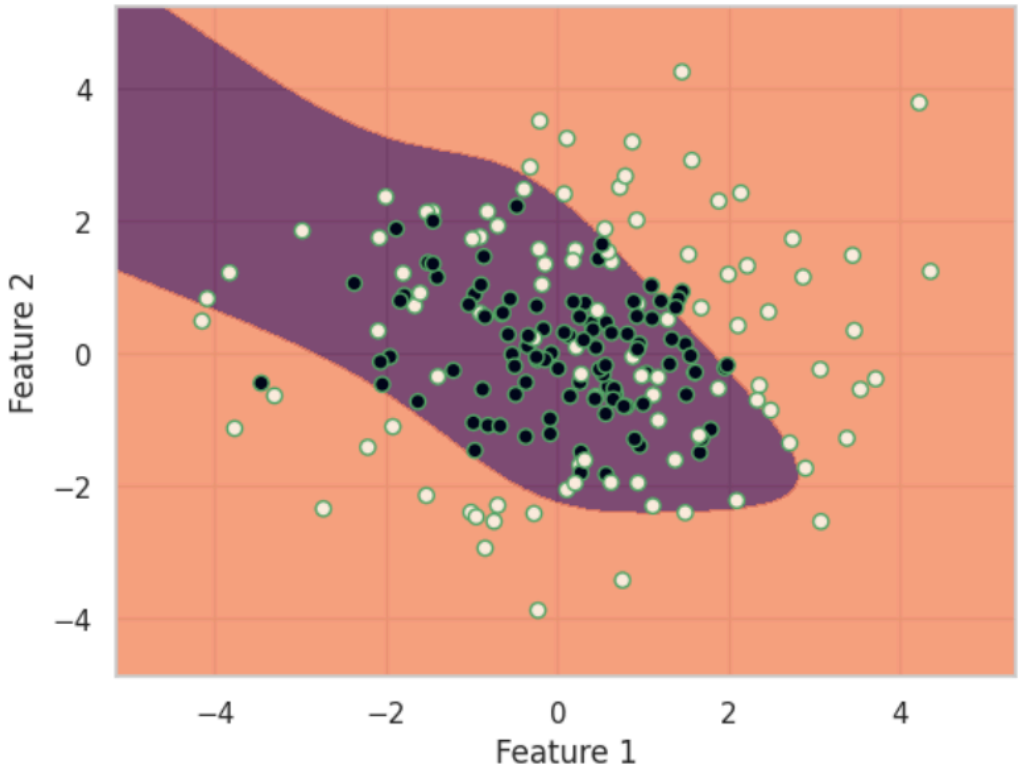
Ans: In the learning curve plot below, the loss increases around the epoch 200, so we choose the epoch number as 200.



c. provide the final text accuracy, defined as the number of correct classifications divided by the total number of examples:

Ans : accuracy: 0.725. This is computed by the function of the implemented mlp classification: compute_accuracy

d. Plot the learned decision surface along with observations from the testset,and:



e. discuss any decisions or observations that you find relevant:
Ans: Setting the number of epochs excessively high typically results in an initial decline in loss, followed by a subsequent increase, indicative of overfitting. However, an interesting observation is that despite the overfitting, the final model accuracy often continues to rise, suggesting an ostensibly high performance. This phenomenon can be attributed to the remarkable similarity between the testing and training datasets. Such a high degree of resemblance implies that, even when the model overfits to the training data, it retains its ability to perform well on the test data. This condition underscores the critical importance of ensuring diversity and representativeness in the datasets to accurately gauge the model's generalization capabilities beyond its training scope.

2. `Spiral dataset:`

a. list hyper-parameters used in the model:

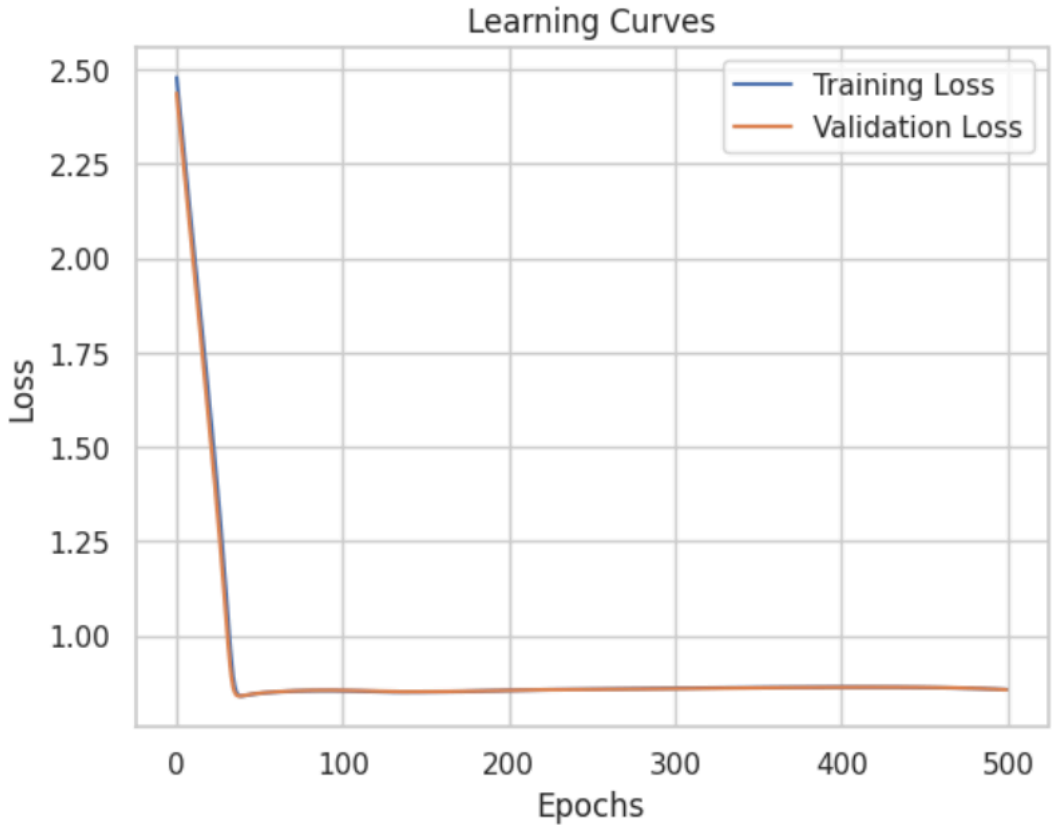
- input_size=2
- hidden_size=15
  - Choose best k (shown below)

```
Hidden Size: 5, Average Accuracy: 0.7965
Hidden Size: 6, Average Accuracy: 0.7978
Hidden Size: 7, Average Accuracy: 0.7965
Hidden Size: 8, Average Accuracy: 0.7967
Hidden Size: 9, Average Accuracy: 0.7998
Hidden Size: 10, Average Accuracy: 0.7990
Hidden Size: 11, Average Accuracy: 0.7985
Hidden Size: 12, Average Accuracy: 0.7977
Hidden Size: 13, Average Accuracy: 0.7973
Hidden Size: 14, Average Accuracy: 0.7970
Hidden Size: 15, Average Accuracy: 0.8005
Hidden Size: 16, Average Accuracy: 0.7980
Hidden Size: 17, Average Accuracy: 0.7975
Hidden Size: 18, Average Accuracy: 0.7967
Hidden Size: 19, Average Accuracy: 0.7983
Hidden Size: 20, Average Accuracy: 0.7978
Hidden Size: 21, Average Accuracy: 0.7988
Hidden Size: 22, Average Accuracy: 0.7947
Hidden Size: 23, Average Accuracy: 0.7993
Hidden Size: 24, Average Accuracy: 0.7963
Hidden Size: 25, Average Accuracy: 0.7972
Hidden Size: 26, Average Accuracy: 0.7992
Hidden Size: 27, Average Accuracy: 0.7998
Hidden Size: 28, Average Accuracy: 0.8003
Hidden Size: 29, Average Accuracy: 0.7950
Best Model - best k: 15, Accuracy: 0.8005
```

- output_size=1
- learning_rate=0.01
- epochs=200

b. Plot the learning curves for training and validation loss as a function of training epochs:

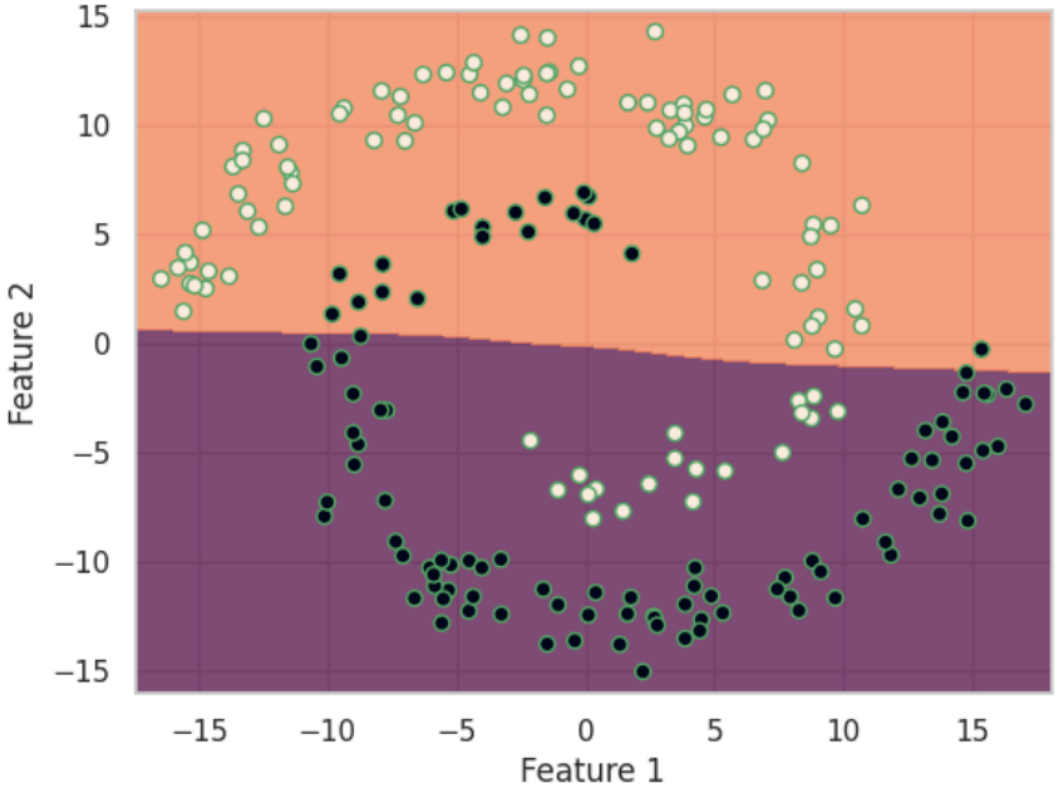
Ans: In the learning curve plot below , the loss becomes stable around the epoch 200, so we choose the epoch number as 200.



c. provide the final text accuracy, defined as the number of correct classifications divided by the total number of examples:

`Ans:` accuracy: 0.805. This is computed by the function of the implemented mlp classification: compute_accuracy

d. Plot the learned decision surface along with observations from the testset,and:



e. discuss any decisions or observations that you find relevant:
Ans: The learning curve indicates that the model's loss stabilizes around 200 epochs, suggesting this as an optimal number for training without further gains from additional epochs, hence avoiding overfitting. The test accuracy is decent at 0.805, but the decision surface plot reveals a linear boundary that doesn't perfectly classify all points, hinting at possible underfitting or that the data may not be linearly separable.

3. **Two Gaussian dataset:**

a. list hyper-parameters used in the model:

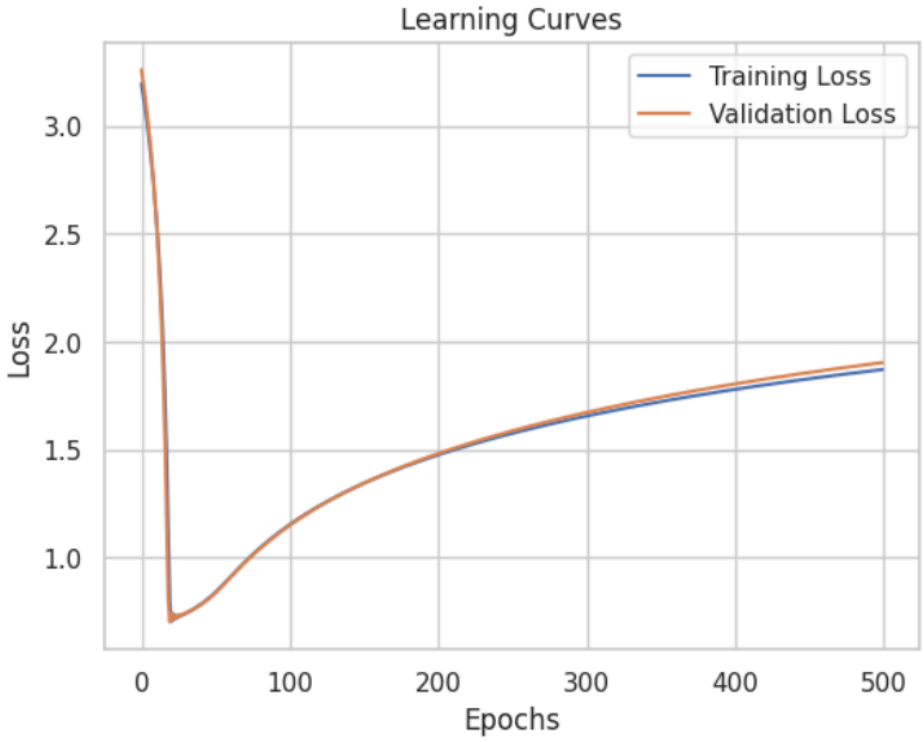
- input_size=2
- hidden_size=20
  - Choose best k (shown below)

```
Hidden Size: 5, Average Accuracy: 0.9130
Hidden Size: 6, Average Accuracy: 0.9142
Hidden Size: 7, Average Accuracy: 0.9135
Hidden Size: 8, Average Accuracy: 0.9128
Hidden Size: 9, Average Accuracy: 0.9122
Hidden Size: 10, Average Accuracy: 0.9137
Hidden Size: 11, Average Accuracy: 0.9132
Hidden Size: 12, Average Accuracy: 0.9130
Hidden Size: 13, Average Accuracy: 0.9132
Hidden Size: 14, Average Accuracy: 0.9122
Hidden Size: 15, Average Accuracy: 0.9135
Hidden Size: 16, Average Accuracy: 0.9133
Hidden Size: 17, Average Accuracy: 0.9125
Hidden Size: 18, Average Accuracy: 0.9102
Hidden Size: 19, Average Accuracy: 0.9127
Hidden Size: 20, Average Accuracy: 0.9148
Hidden Size: 21, Average Accuracy: 0.8932
Hidden Size: 22, Average Accuracy: 0.9113
Hidden Size: 23, Average Accuracy: 0.8758
Hidden Size: 24, Average Accuracy: 0.8713
Hidden Size: 25, Average Accuracy: 0.8827
Hidden Size: 26, Average Accuracy: 0.8545
Hidden Size: 27, Average Accuracy: 0.8245
Hidden Size: 28, Average Accuracy: 0.8558
Hidden Size: 29, Average Accuracy: 0.7303
Best Model - best k: 20, Accuracy: 0.9148
```

- output_size=1
- learning_rate=0.01
- epochs=250

b. Plot the learning curves for training and validation loss as a function of training epochs:

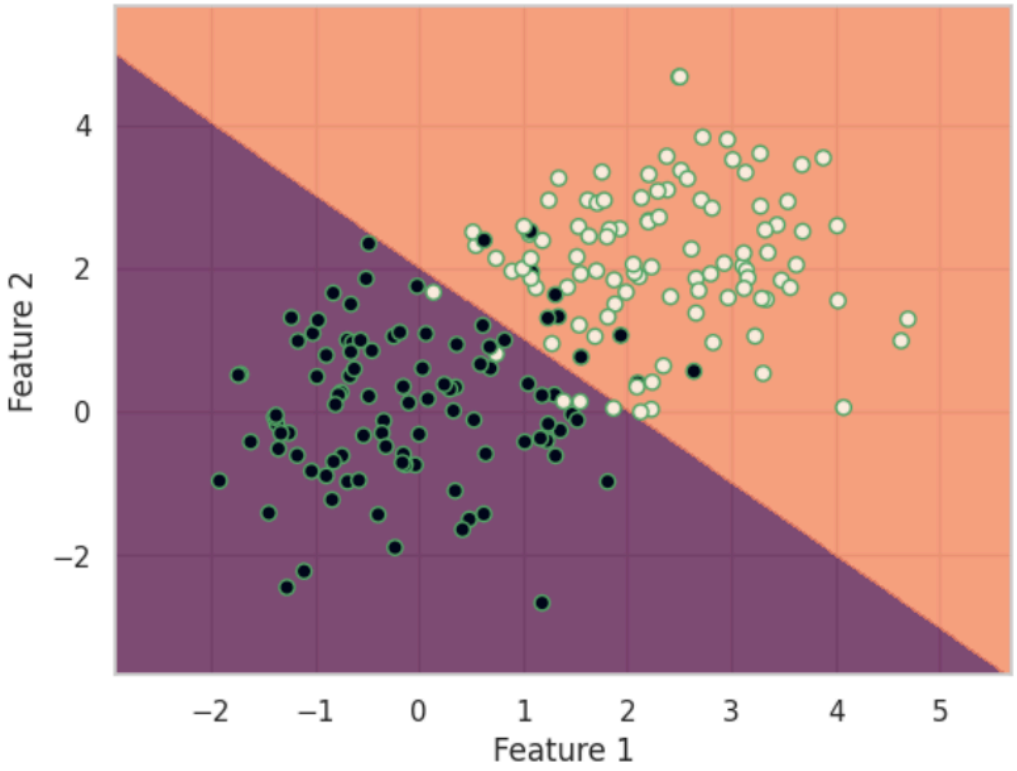
Ans: In the learning curve plot below , the loss increases around the epoch 100, so we choose the epoch number as 100.



c. provide the final text accuracy, defined as the number of correct classifications divided by the total number of examples:

`Ans:` accuracy: 0.925. This is computed by the function of the implemented mlp classification: compute_accuracy

d. Plot the learned decision surface along with observations from the testset,and:



e. discuss any decisions or observations that you find relevant:
Ans: The decision surface plot reveals a non-linear boundary that separates the two classes, with some overlap between them. This overlap might suggest that while the model generally distinguishes between classes well, there are regions where it struggles to clearly differentiate between them, possibly due to the intrinsic data complexity or noise.

4. **Xor dataset:**

a. list hyper-parameters used in the model:

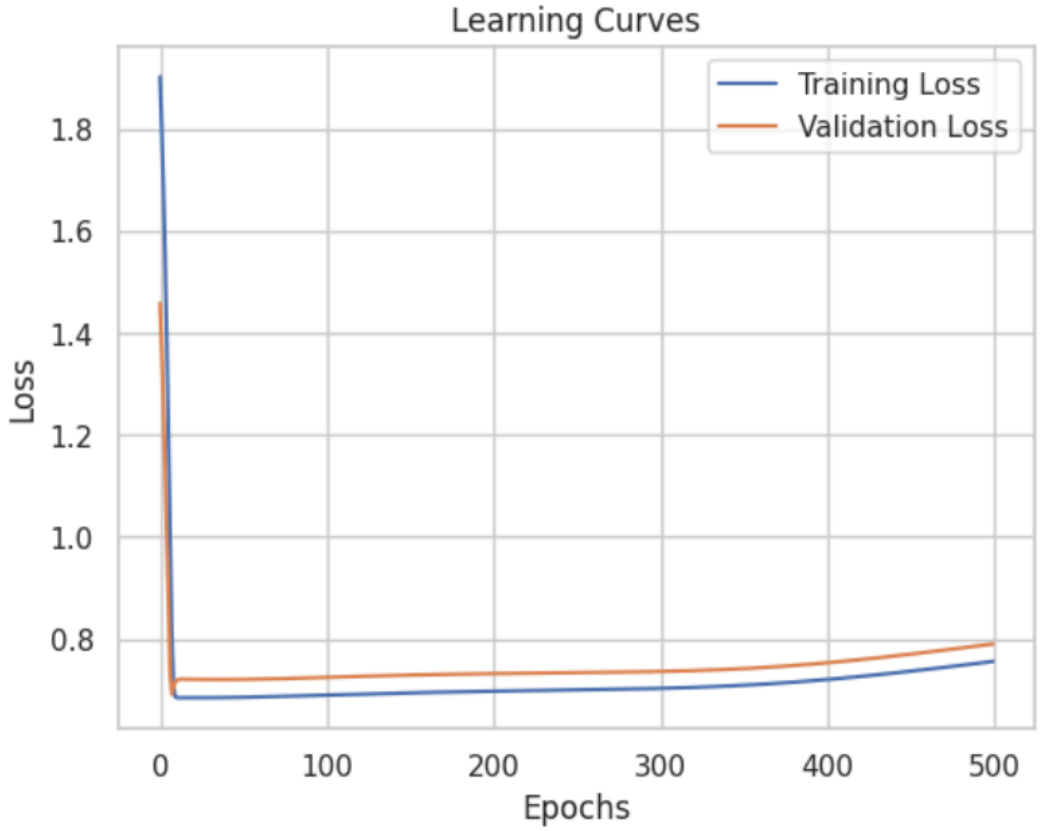
- input_size=2
- hidden_size=12
  - Choose best k (shown below)

```
Hidden Size: 5, Average Accuracy: 0.5905
Hidden Size: 6, Average Accuracy: 0.5910
Hidden Size: 7, Average Accuracy: 0.5967
Hidden Size: 8, Average Accuracy: 0.5983
Hidden Size: 9, Average Accuracy: 0.5950
Hidden Size: 10, Average Accuracy: 0.5987
Hidden Size: 11, Average Accuracy: 0.6008
Hidden Size: 12, Average Accuracy: 0.6017
Hidden Size: 13, Average Accuracy: 0.5988
Hidden Size: 14, Average Accuracy: 0.5980
Hidden Size: 15, Average Accuracy: 0.6002
Hidden Size: 16, Average Accuracy: 0.5992
Hidden Size: 17, Average Accuracy: 0.5900
Hidden Size: 18, Average Accuracy: 0.5990
Hidden Size: 19, Average Accuracy: 0.5825
Hidden Size: 20, Average Accuracy: 0.5950
Hidden Size: 21, Average Accuracy: 0.5810
Hidden Size: 22, Average Accuracy: 0.5603
Hidden Size: 23, Average Accuracy: 0.5408
Hidden Size: 24, Average Accuracy: 0.5178
Hidden Size: 25, Average Accuracy: 0.5235
Hidden Size: 26, Average Accuracy: 0.5268
Hidden Size: 27, Average Accuracy: 0.5110
Hidden Size: 28, Average Accuracy: 0.5172
Hidden Size: 29, Average Accuracy: 0.4998
Best Model - best k: 12, Accuracy: 0.6017
```

- output_size=1
- learning_rate=0.01
- epochs=200

b. Plot the learning curves for training and validation loss as a function of training epochs:

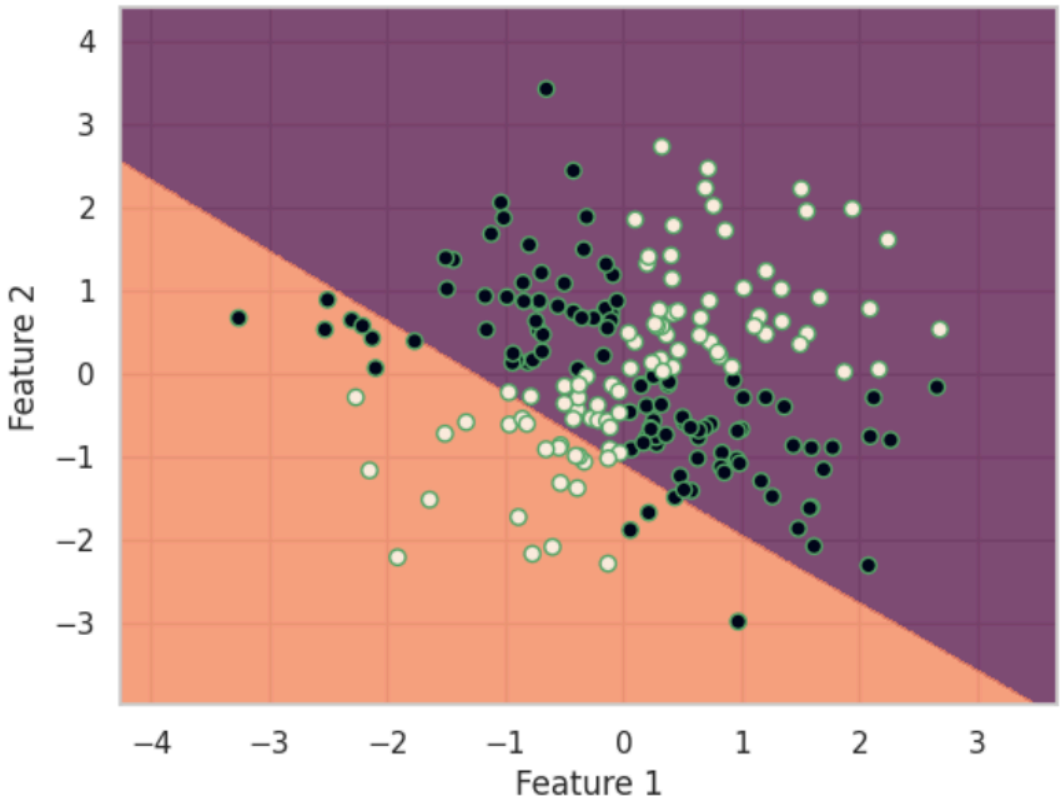
Ans: In the learning curve plot below , the loss becomes stable around the epoch 200, so we choose the epoch number as 200.



c. provide the final text accuracy, defined as the number of correct classifications divided by the total number of examples:

Ans: accuracy: 0.590. This is computed by the function of the implemented mlp classification: compute_accuracy

d. Plot the learned decision surface along with observations from the testset,and:



e. discuss any decisions or observations that you find relevant:
Ans: The final test accuracy is relatively low at 0.590, pointing towards a potential underfitting issue or that the model's current complexity is insufficient to capture the underlying patterns in the data. The decision surface plot shows a boundary that does not cleanly separate all the data points, with a notable number of points being misclassified on either side of the boundary. This reinforces the possibility of underfitting and suggests that the data might not be linear separable and the model may benefit from increasing its complexity or using feature engineering to better capture the distribution of the data.

2. (2.0 points) Repeat Step 1 using mean squared error as the cost function.

1. `Center_surround dataset:`

a. list hyper-parameters used in the model:

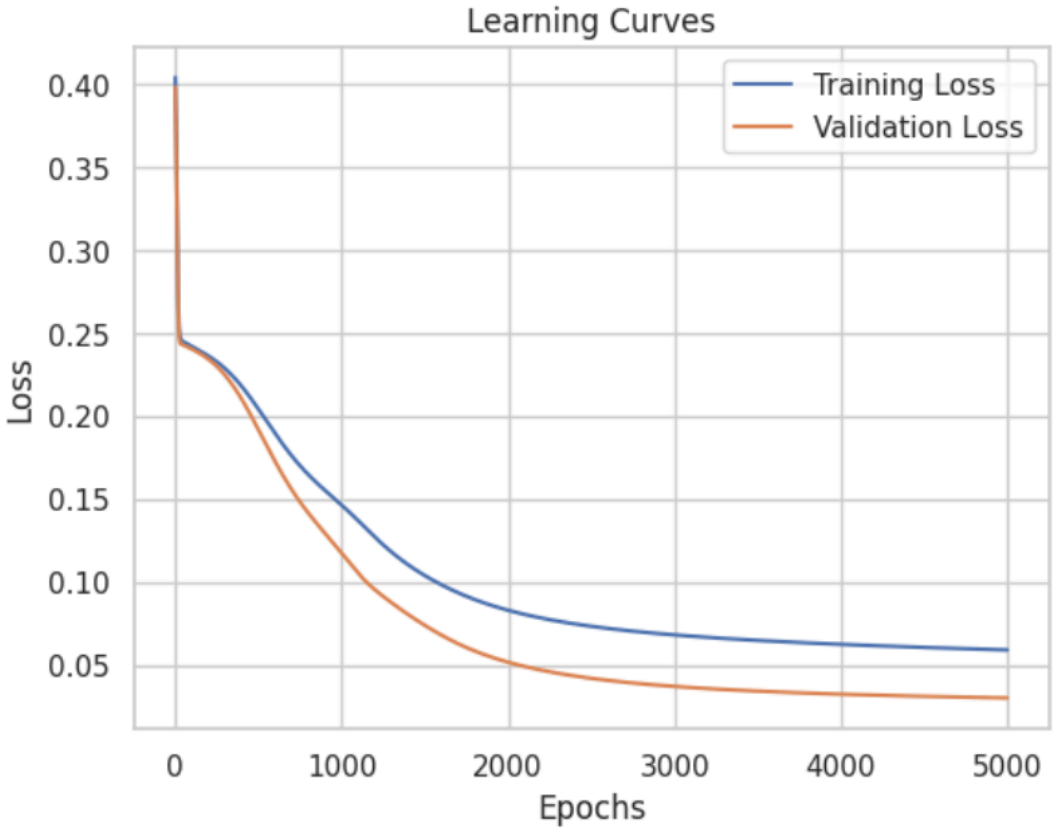
- input_size=2
- hidden_size=8
  - Choose best k (shown below)

```
Hidden Size: 5, Average Accuracy: 0.7595
Hidden Size: 6, Average Accuracy: 0.7642
Hidden Size: 7, Average Accuracy: 0.7650
Hidden Size: 8, Average Accuracy: 0.7668
Hidden Size: 9, Average Accuracy: 0.7585
Hidden Size: 10, Average Accuracy: 0.7602
Hidden Size: 11, Average Accuracy: 0.7432
Hidden Size: 12, Average Accuracy: 0.7450
Hidden Size: 13, Average Accuracy: 0.7225
Hidden Size: 14, Average Accuracy: 0.7252
Hidden Size: 15, Average Accuracy: 0.7140
Hidden Size: 16, Average Accuracy: 0.7115
Hidden Size: 17, Average Accuracy: 0.7112
Hidden Size: 18, Average Accuracy: 0.7065
Hidden Size: 19, Average Accuracy: 0.7055
Hidden Size: 20, Average Accuracy: 0.6970
Hidden Size: 21, Average Accuracy: 0.6762
Hidden Size: 22, Average Accuracy: 0.6672
Hidden Size: 23, Average Accuracy: 0.6355
Hidden Size: 24, Average Accuracy: 0.5710
Hidden Size: 25, Average Accuracy: 0.5505
Hidden Size: 26, Average Accuracy: 0.5600
Hidden Size: 27, Average Accuracy: 0.5352
Hidden Size: 28, Average Accuracy: 0.5000
Hidden Size: 29, Average Accuracy: 0.5157
Best Model - best k: 8, Accuracy: 0.7668
```

- output_size=1
- learning_rate=0.01
- epochs=2000

b. Plot the learning curves for training and validation loss as a function of training epochs:

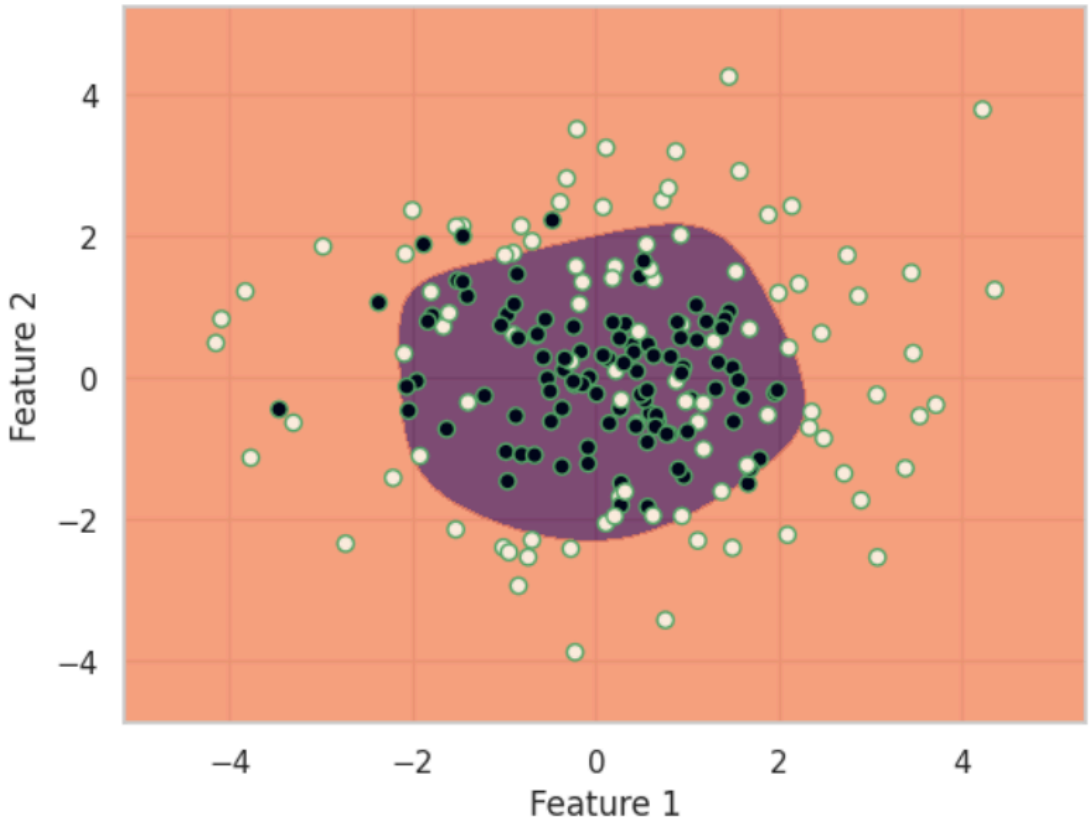
Ans: In the learning curve plot below , the loss becomes stable when the epoch number is around 3000, so we choose the epoch number as 3000.



c. provide the final text accuracy, defined as the number of correct classifications divided by the total number of examples:

`Ans:` accuracy: 0.765. This is computed by the function of the implemented mlp classification: compute_accuracy

d. Plot the learned decision surface along with observations from the testset,and:



e. discuss any decisions or observations that you find relevant:
Ans: In the decision surface plot, a distinct cluster of test data points is not entirely separated by the decision boundary, indicating potential misclassifications. This could imply that the model may benefit from further optimization, such as hyperparameter tuning, exploring more complex model architectures, or employing advanced feature engineering to better capture the nuances in the data.

2. **Spiral dataset:**

a. list hyper-parameters used in the model:

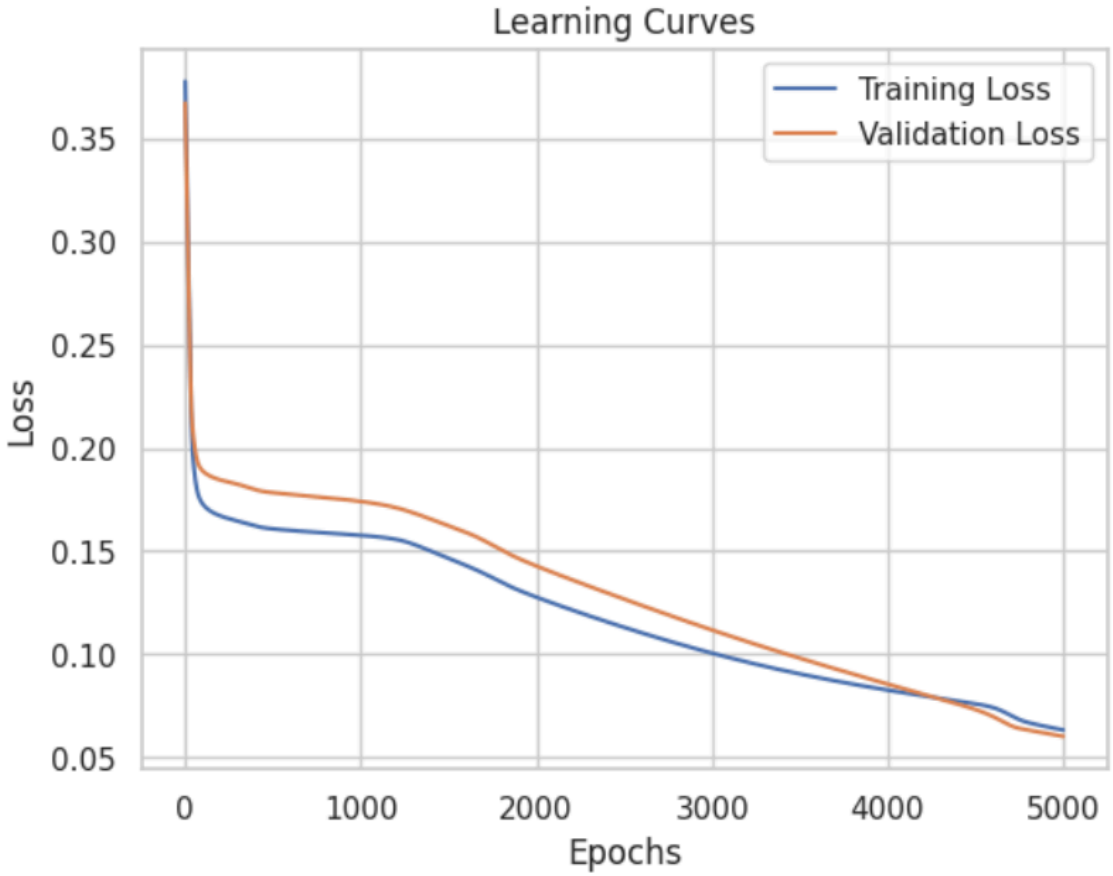
- input_size=2
- hidden_size=8
  - Choose best k (shown below)

```
Hidden Size: 5, Average Accuracy: 0.7987
Hidden Size: 6, Average Accuracy: 0.8083
Hidden Size: 7, Average Accuracy: 0.8075
Hidden Size: 8, Average Accuracy: 0.8183
Hidden Size: 9, Average Accuracy: 0.8063
Hidden Size: 10, Average Accuracy: 0.8095
Hidden Size: 11, Average Accuracy: 0.8017
Hidden Size: 12, Average Accuracy: 0.7980
Hidden Size: 13, Average Accuracy: 0.7988
Hidden Size: 14, Average Accuracy: 0.7975
Hidden Size: 15, Average Accuracy: 0.7952
Hidden Size: 16, Average Accuracy: 0.7983
Hidden Size: 17, Average Accuracy: 0.7953
Hidden Size: 18, Average Accuracy: 0.7958
Hidden Size: 19, Average Accuracy: 0.7953
Hidden Size: 20, Average Accuracy: 0.7952
Hidden Size: 21, Average Accuracy: 0.7953
Hidden Size: 22, Average Accuracy: 0.7950
Hidden Size: 23, Average Accuracy: 0.7955
Hidden Size: 24, Average Accuracy: 0.8013
Hidden Size: 25, Average Accuracy: 0.7973
Hidden Size: 26, Average Accuracy: 0.7950
Hidden Size: 27, Average Accuracy: 0.7985
Hidden Size: 28, Average Accuracy: 0.7993
Hidden Size: 29, Average Accuracy: 0.7937
Best Model - best k: 8, Accuracy: 0.8183
```

- output_size=1
- learning_rate=0.01
- epochs=3000

b. Plot the learning curves for training and validation loss as a function of training epochs:

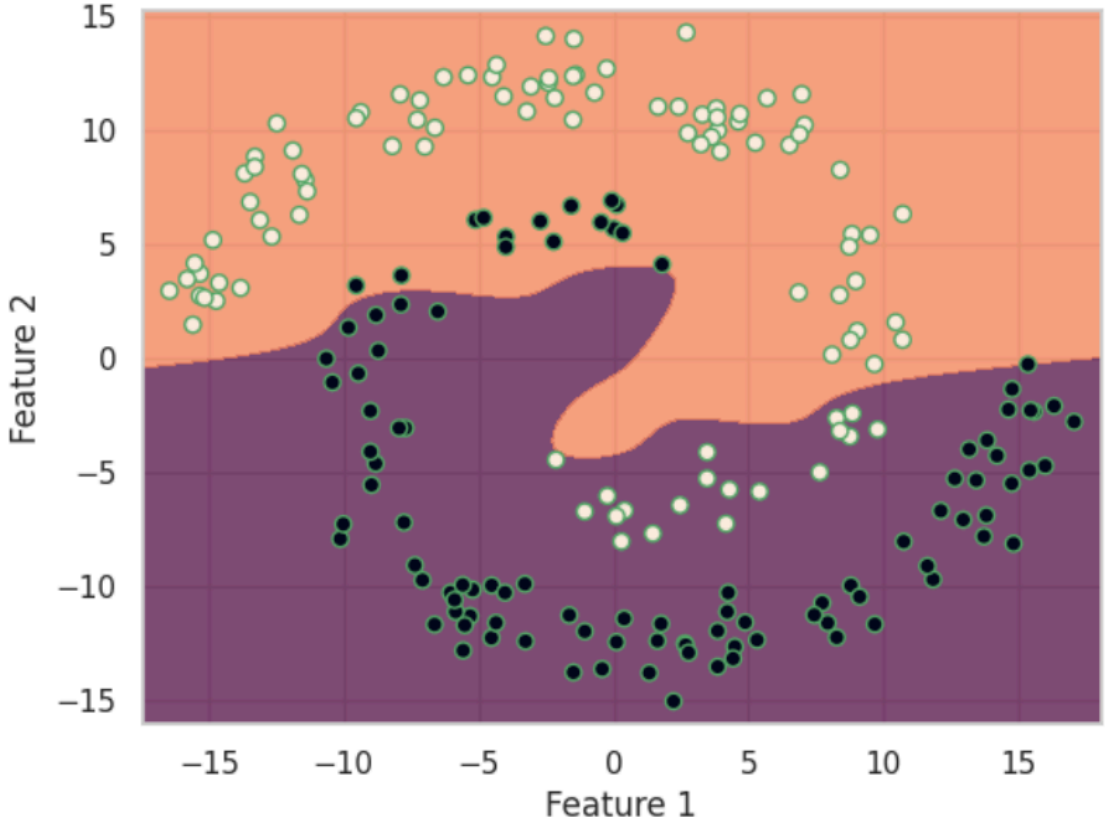
Ans: In the learning curve plot below , the loss is continually decreasing. To avoid overfitting, the epoch number is decided to be 3000.



c. provide the final text accuracy, defined as the number of correct classifications divided by the total number of examples:

Ans: accuracy: 0.825. This is computed by the function of the implemented mlp classification: compute_accuracy

d. Plot the learned decision surface along with observations from the testset,and:



e. discuss any decisions or observations that you find relevant:
Ans: The model achieves a test accuracy of 0.82, demonstrating a solid performance. The decision surface plot, however, shows a non-linear boundary with a significant number of test points from both classes being mixed near the boundary, which could be indicative of some misclassification. This might point to a need for refining the model further, possibly by exploring more complex or different types of models to improve the separation between classes, or by considering additional feature engineering to provide clearer boundaries in the feature space.

3. **Two Gaussian dataset:**

a. list hyper-parameters used in the model:

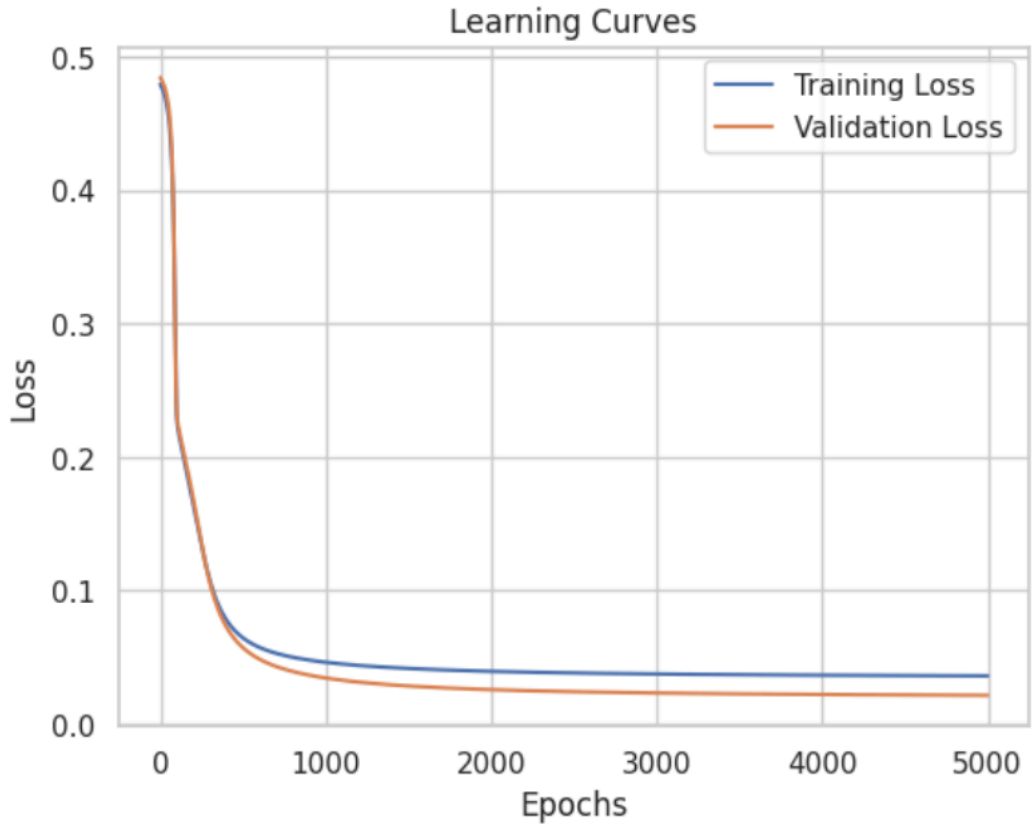
- input_size=2
- hidden_size=18
  - Choose best k (shown below)

```
Hidden Size: 5, Average Accuracy: 0.9147
Hidden Size: 6, Average Accuracy: 0.9145
Hidden Size: 7, Average Accuracy: 0.9130
Hidden Size: 8, Average Accuracy: 0.9117
Hidden Size: 9, Average Accuracy: 0.9115
Hidden Size: 10, Average Accuracy: 0.9137
Hidden Size: 11, Average Accuracy: 0.9113
Hidden Size: 12, Average Accuracy: 0.9155
Hidden Size: 13, Average Accuracy: 0.9153
Hidden Size: 14, Average Accuracy: 0.9185
Hidden Size: 15, Average Accuracy: 0.9205
Hidden Size: 16, Average Accuracy: 0.9193
Hidden Size: 17, Average Accuracy: 0.9223
Hidden Size: 18, Average Accuracy: 0.9238
Hidden Size: 19, Average Accuracy: 0.9015
Hidden Size: 20, Average Accuracy: 0.8715
Hidden Size: 21, Average Accuracy: 0.8680
Hidden Size: 22, Average Accuracy: 0.8265
Hidden Size: 23, Average Accuracy: 0.7998
Hidden Size: 24, Average Accuracy: 0.6025
Hidden Size: 25, Average Accuracy: 0.6628
Hidden Size: 26, Average Accuracy: 0.6075
Hidden Size: 27, Average Accuracy: 0.5578
Hidden Size: 28, Average Accuracy: 0.5825
Hidden Size: 29, Average Accuracy: 0.5188
Best Model - best k: 18, Accuracy: 0.9238
```

- output_size=1
- learning_rate=0.01
- epochs=1000

b. Plot the learning curves for training and validation loss as a function of training epochs:

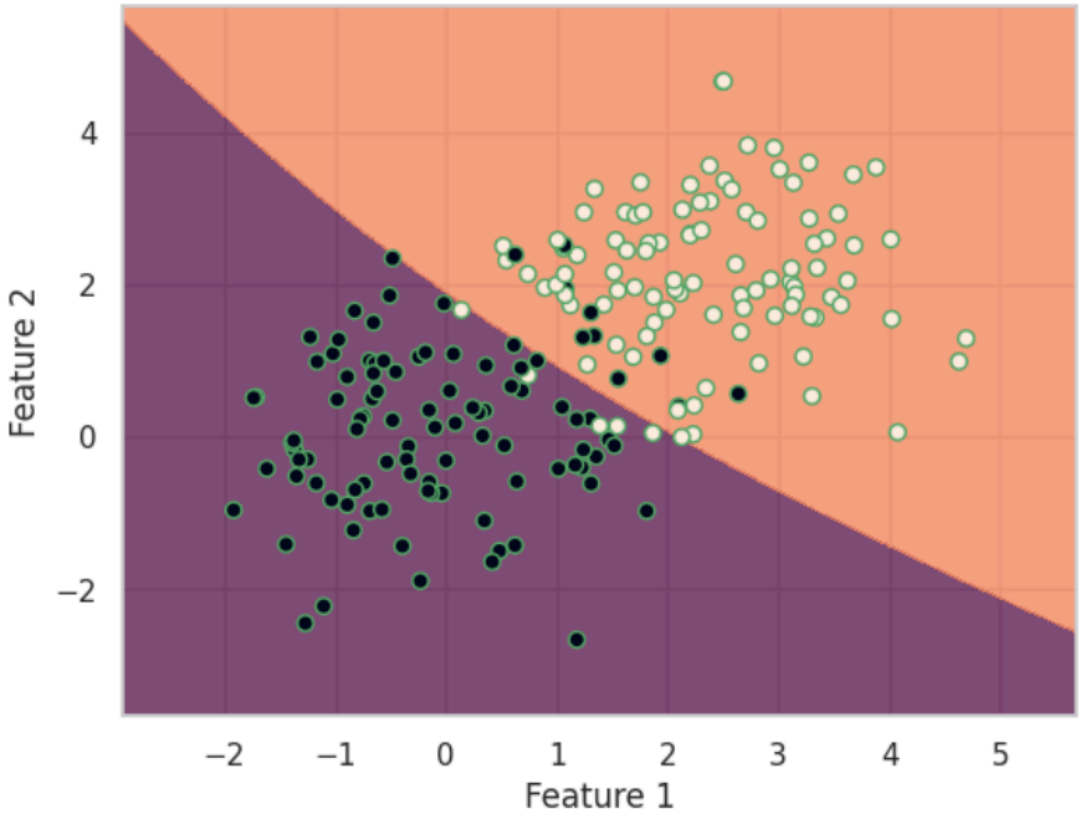
Ans: In the learning curve plot below , the loss is stable when the epoch number is around 1000. So the epoch number is decided to be 1000.



c. provide the final text accuracy, defined as the number of correct classifications divided by the total number of examples:

Ans: accuracy: 0.925. This is computed by the function of the implemented mlp classification: compute_accuracy

d. Plot the learned decision surface along with observations from the testset,and:



e. discuss any decisions or observations that you find relevant:
Ans: The model's high test accuracy of 0.925 suggests that it performs well on unseen data. However, the decision surface plot shows that while there is a clear boundary separating the two classes, there are a number of data points that are close to the boundary, which may be prone to misclassification if the model is challenged with more complex or noisy data. To further improve the model's robustness, one might consider implementing regularization techniques, collecting more training data to better define the decision boundary, or exploring more complex model architectures that could capture the nuances of the data more effectively.

4. **Xor dataset:**

a. list hyper-parameters used in the model:

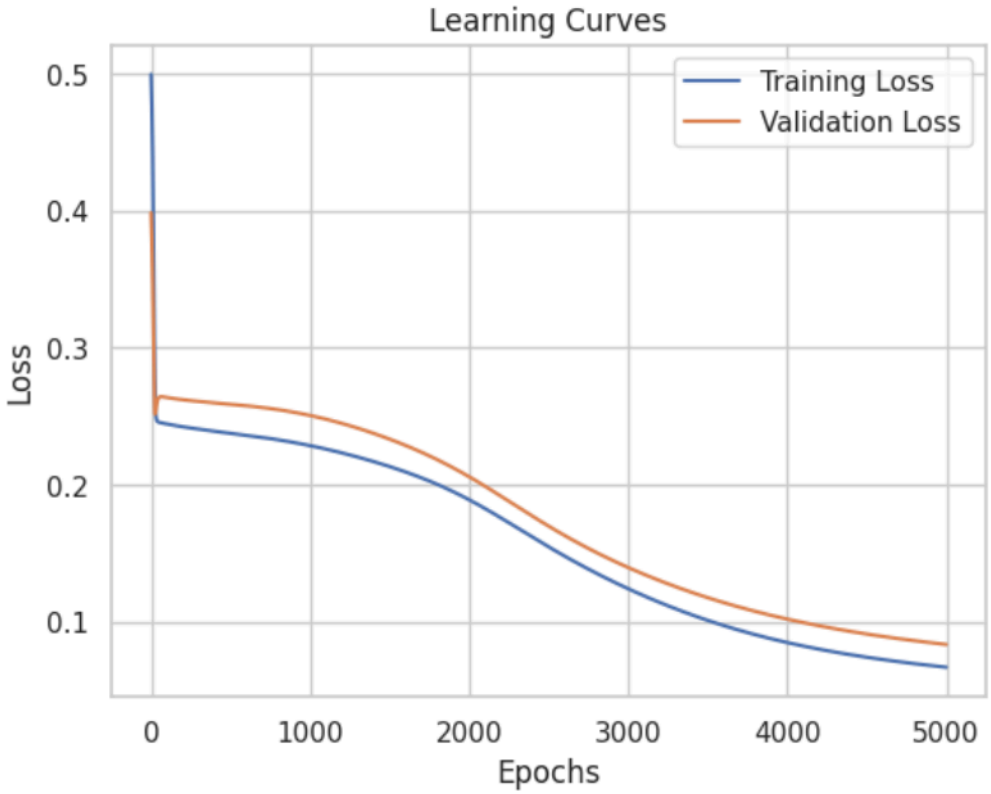
- input_size=2
- hidden_size=7
  - Choose best k (shown below)

```
Hidden Size: 5, Average Accuracy: 0.7787
Hidden Size: 6, Average Accuracy: 0.7938
Hidden Size: 7, Average Accuracy: 0.7960
Hidden Size: 8, Average Accuracy: 0.7933
Hidden Size: 9, Average Accuracy: 0.7565
Hidden Size: 10, Average Accuracy: 0.7787
Hidden Size: 11, Average Accuracy: 0.7485
Hidden Size: 12, Average Accuracy: 0.7245
Hidden Size: 13, Average Accuracy: 0.7248
Hidden Size: 14, Average Accuracy: 0.7055
Hidden Size: 15, Average Accuracy: 0.6925
Hidden Size: 16, Average Accuracy: 0.6705
Hidden Size: 17, Average Accuracy: 0.6537
Hidden Size: 18, Average Accuracy: 0.6365
Hidden Size: 19, Average Accuracy: 0.6253
Hidden Size: 20, Average Accuracy: 0.6287
Hidden Size: 21, Average Accuracy: 0.6092
Hidden Size: 22, Average Accuracy: 0.5873
Hidden Size: 23, Average Accuracy: 0.5687
Hidden Size: 24, Average Accuracy: 0.5573
Hidden Size: 25, Average Accuracy: 0.5440
Hidden Size: 26, Average Accuracy: 0.5370
Hidden Size: 27, Average Accuracy: 0.5033
Hidden Size: 28, Average Accuracy: 0.4845
Hidden Size: 29, Average Accuracy: 0.4838
Best Model - best k: 7, Accuracy: 0.7960
```

- output_size=1
- learning_rate=0.01
- epochs=3000

b. Plot the learning curves for training and validation loss as a function of training epochs:

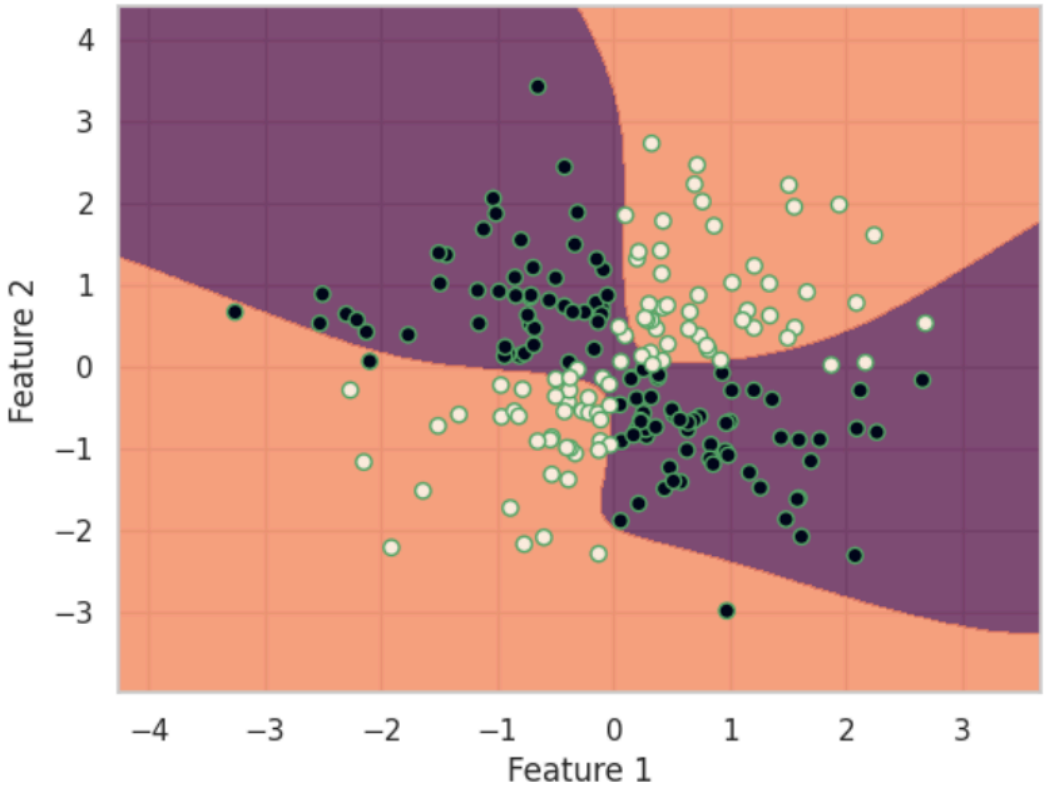
Ans: In the learning curve plot below , the loss is continually decreasing. To avoid overfitting, the epoch number is decided to be 4000.



c. provide the final text accuracy, defined as the number of correct classifications divided by the total number of examples:

`Ans:` accuracy: 0.91. This is computed by the function of the implemented mlp classification: compute_accuracy

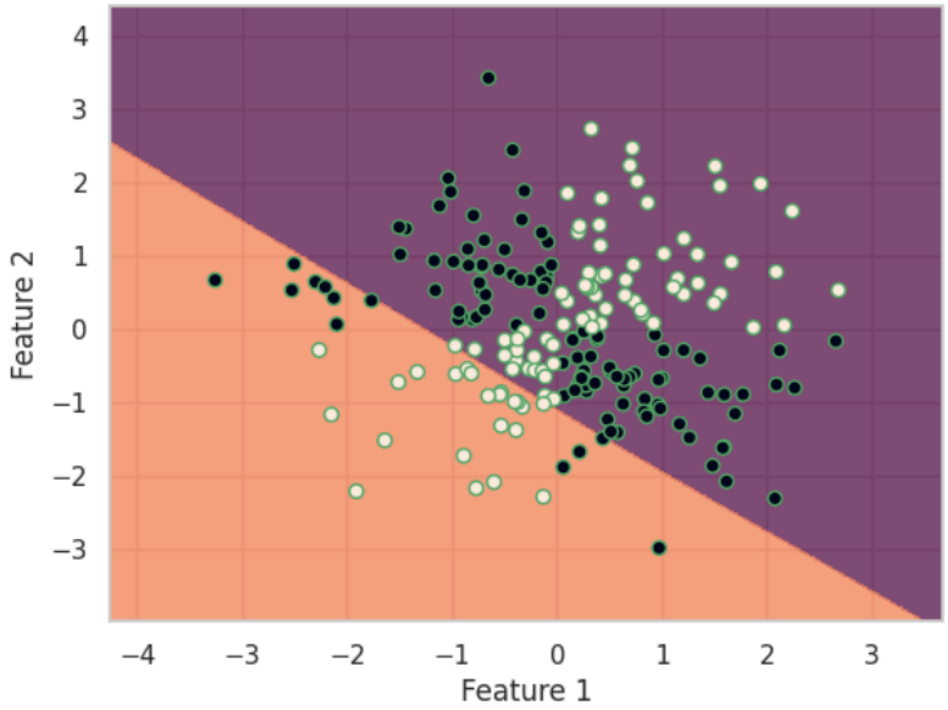
d. Plot the learned decision surface along with observations from the testset,and:



e. discuss any decisions or observations that you find relevant:
Ans: With the final test accuracy reported as 0.91, the model demonstrates high predictive performance on unseen data. However, examining the decision surface plot reveals that while the model distinguishes between classes fairly well, there are areas where the classification boundary is less clear, with several points from each class lying close to the boundary. This could indicate that the model may be overfitting to noise within the training data or that there are inherent overlaps in the feature space of the classes that are difficult to separate.

3. (2.0 points) Select the worst-performing model (dataset, cost function and number of hidden nodes) from the above experiments and plot decision boundaries learned for each node in your hidden layer. Discuss why you selected this instance and speculate about factors contributing to poor performance.

- Based on the above observations, the worst-performing model is the MLP model used on **"XOR Dataset"**, with an accuracy of only 0.59. The hyper-parameters used for training the model are:
    - hidden size is 12
    - cost function is binary cross-entropy.
- The decision boundary plot is below:



- First of all, the XOR dataset is not linearly separable: the output y equals 1 only if exactly one of the inputs is one. There may be several reasons contributing to the poor behavior of our MLP model.
    - The number of neurons is not sufficient. We applied k=12 number of neurons in fitting the XOR dataset. That might be too few and we should consider adding more neurons.
    - Improper selection of the activation function. We are currently using sigmoid for the activation function, and we could potentially try using ReLU for better performance.
    - Overfitting. We could suffer from overfitting if the number of epochs equals 200. We should also apply regularization techniques such as L1 / L2 regularization, dropout, or early stopping.

4. (1.0 points) Discuss how you might encourage your model to learn “feature maps” that could improve the performance of the instance selected for Step 3.

- As discussed in problem 3, there are several ways to potentially increase the performance of the XOR dataset.
    - Increase the number of neurons to the hidden layer.
    - Change the activation function to ReLU.
    - Add regularization techniques to prevent overfitting.

5. (BONUS 2.0 points) Implement the approach specified in Step 4. Discuss and present results that illustrate "why" your approach did or did not enhance performance. Note: You may use PyTorch (or another deep learning package) to implement this part of the assignment.

- Training history:

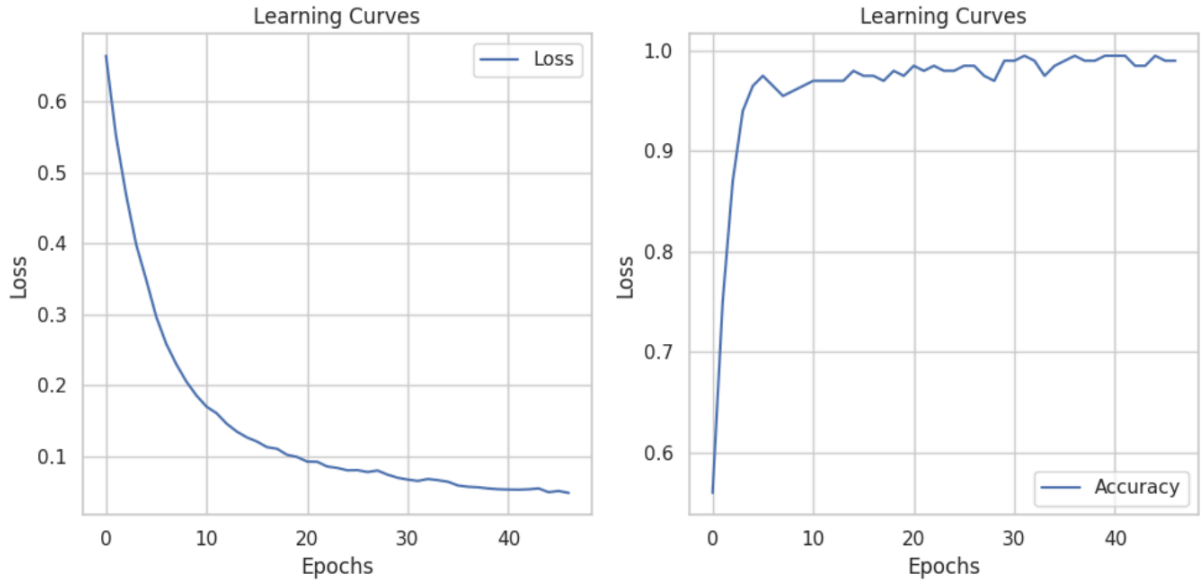
```
Epoch 28/300
7/7 [==============================] - 0s 3ms/step - loss: 0.0795 - accuracy: 0.9750
Epoch 29/300
7/7 [==============================] - 0s 3ms/step - loss: 0.0736 - accuracy: 0.9700
Epoch 30/300
7/7 [==============================] - 0s 3ms/step - loss: 0.0693 - accuracy: 0.9900
Epoch 31/300
7/7 [==============================] - 0s 3ms/step - loss: 0.0667 - accuracy: 0.9900
Epoch 32/300
7/7 [==============================] - 0s 3ms/step - loss: 0.0648 - accuracy: 0.9950
Epoch 33/300
7/7 [==============================] - 0s 3ms/step - loss: 0.0675 - accuracy: 0.9900
Epoch 34/300
7/7 [==============================] - 0s 4ms/step - loss: 0.0659 - accuracy: 0.9750
Epoch 35/300
7/7 [==============================] - 0s 3ms/step - loss: 0.0637 - accuracy: 0.9850
Epoch 36/300
7/7 [==============================] - 0s 3ms/step - loss: 0.0583 - accuracy: 0.9900
Epoch 37/300
7/7 [==============================] - 0s 3ms/step - loss: 0.0566 - accuracy: 0.9950
Epoch 38/300
7/7 [==============================] - 0s 3ms/step - loss: 0.0559 - accuracy: 0.9900
Epoch 39/300
7/7 [==============================] - 0s 4ms/step - loss: 0.0542 - accuracy: 0.9900
Epoch 40/300
7/7 [==============================] - 0s 4ms/step - loss: 0.0531 - accuracy: 0.9950
Epoch 41/300
7/7 [==============================] - 0s 4ms/step - loss: 0.0527 - accuracy: 0.9950
Epoch 42/300
7/7 [==============================] - 0s 4ms/step - loss: 0.0525 - accuracy: 0.9950
Epoch 43/300
7/7 [==============================] - 0s 4ms/step - loss: 0.0530 - accuracy: 0.9850
Epoch 44/300
7/7 [==============================] - 0s 4ms/step - loss: 0.0543 - accuracy: 0.9850
Epoch 45/300
7/7 [==============================] - 0s 4ms/step - loss: 0.0489 - accuracy: 0.9950
Epoch 46/300
7/7 [==============================] - 0s 4ms/step - loss: 0.0507 - accuracy: 0.9900
Epoch 47/300
7/7 [==============================] - 0s 4ms/step - loss: 0.0479 - accuracy: 0.9900
```

- Test Accuracy:

```
Test accuracy: 0.9900000095367432
```

  - We applied keras deep learning framework and created a MLP architecture with the same requirements specified in this homework (two layers: one hidden layer and one output layer). We applied exactly the same strategies discussed in problem 4, namely:
    - Increase the number of neurons in the hidden layer: we changed the number of neurons to be 64 instead of only 12.

    - Activation function: we switched from sigmoid to ReLU activation function.
    - Prevent overfitting: we added an EarlyStopping callback function with patience 15 to prevent the model from memorizing the training dataset. The model ended learning at epoch 47 (could be a different number on different tries).
- The result is significant: the test accuracy improved from 0.59 using our MLP model to 0.99 using Keras' framework.
- Learning curve as shown below:



- Decision boundary shown below: