

ASSIGNMENT 3

CS5691 Pattern Recognition and Machine Learning

CS5691 Assignment 3

Team Members:

BE17B007 N Sowmya Manojna
PH17B010 Thakkar Riya Anandbhai
PH17B011 Chaithanya Krishna Moorthy

Indian Institute of Technology, Madras



Contents

| | | |
|----------|---------------------------|-----------|
| 1 | Dataset 1A | 2 |
| 1.1 | Perceptron | 2 |
| 1.1.1 | Classes 0 and 1 | 2 |
| 1.1.2 | Classes 0 and 2 | 3 |
| 1.1.3 | Classes 0 and 3 | 4 |
| 1.1.4 | Classes 1 and 2 | 5 |
| 1.1.5 | Classes 1 and 3 | 6 |
| 1.1.6 | Classes 2 and 3 | 7 |
| 1.2 | MLFFNN | 8 |
| 1.2.1 | Classification Accuracies | 8 |
| 1.2.2 | Best Model | 8 |
| 1.2.3 | Decision Region | 9 |
| 1.3 | Linear SVM | 10 |
| 2 | Dataset 1B | 11 |
| 2.1 | MLFFNN | 11 |
| 2.1.1 | Classification Accuracies | 11 |
| 2.1.2 | Best Model | 11 |
| 2.1.3 | Decision Region | 12 |
| 2.1.4 | Surface Plots | 13 |
| 2.1.4.1 | Hidden Layer 1, Node 1 | 14 |
| 2.1.4.2 | Hidden Layer 1, Node 2 | 15 |
| 2.1.4.3 | Hidden Layer 1, Node 3 | 16 |
| 2.1.4.4 | Hidden Layer 1, Node 4 | 17 |
| 2.1.4.5 | Hidden Layer 1, Node 5 | 18 |
| 2.1.4.6 | Hidden Layer 1, Node 6 | 19 |
| 2.1.4.7 | Hidden Layer 1, Node 7 | 20 |
| 2.1.4.8 | Hidden Layer 1, Node 8 | 21 |
| 2.1.4.9 | Hidden Layer 2, Node 1 | 22 |
| 2.1.4.10 | Hidden Layer 2, Node 2 | 23 |
| 2.1.4.11 | Hidden Layer 2, Node 3 | 24 |
| 2.1.4.12 | Hidden Layer 2, Node 4 | 25 |
| 2.1.4.13 | Hidden Layer 2, Node 5 | 26 |
| 2.1.4.14 | Hidden Layer 2, Node 6 | 27 |
| 2.1.4.15 | Hidden Layer 2, Node 7 | 28 |
| 2.1.4.16 | Hidden Layer 2, Node 8 | 29 |
| 2.1.4.17 | Output Layer, Node 1 | 30 |
| 2.1.4.18 | Output Layer, Node 2 | 31 |
| 2.1.4.19 | Output Layer, Node 3 | 32 |
| 2.2 | Non-Linear SVM | 33 |
| 3 | Dataset 2A | 34 |
| 3.1 | MLFFNN | 34 |
| 3.1.1 | Classification Accuracies | 34 |
| 3.1.2 | Best Model | 34 |
| 3.2 | Gaussian-kernel SVM | 35 |

1 Dataset 1A

This dataset contains data for four classes - 0, 1, 2 and 3. The classes are linearly separable and the dimension of the feature space is 2.

1.1 Perceptron

Varying the hyperparameter : Learning Rate (η) for the Perceptron model, the accuracies on the training and validation (30% of the file `dev.csv`) data for all possible pairings of the classes (leading to a total of 6 pairs) were obtained and the best η value based on CV accuracies was chosen as follows:

1.1.1 Classes 0 and 1

| Hyperparameter | Training Accuracy | CV Accuracy |
|----------------|-------------------|-------------|
| 0.001 | 1.0 | 1.0 |
| 0.005 | 1.0 | 1.0 |
| 0.01 | 1.0 | 1.0 |
| 0.05 | 1.0 | 1.0 |
| 0.1 | 1.0 | 1.0 |
| 1.0 | 1.0 | 1.0 |
| 5.0 | 1.0 | 1.0 |
| 10.0 | 1.0 | 1.0 |
| 100.0 | 1.0 | 1.0 |

Table 1: Table of training accuracies and CV accuracies for data of classes 0 and 1 of the 1A data set

The accuracy is 100% for all values of the hyperparameters. Taking the default value of 0.01, the accuracy on the test data is **100%**. The confusion matrices for the training and test data are as in [Figure 1](#).

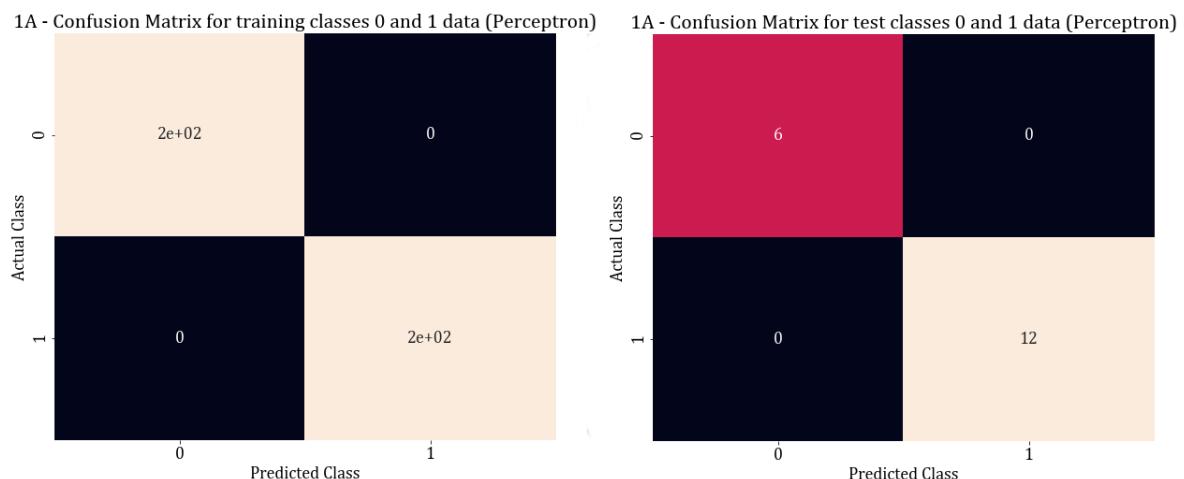


Figure 1: Confusion matrices for training and test data belonging to classes 0 and 1 of data 1A using perceptron classifier

The decision region plot for the perceptron classifier is in [Figure 2](#).

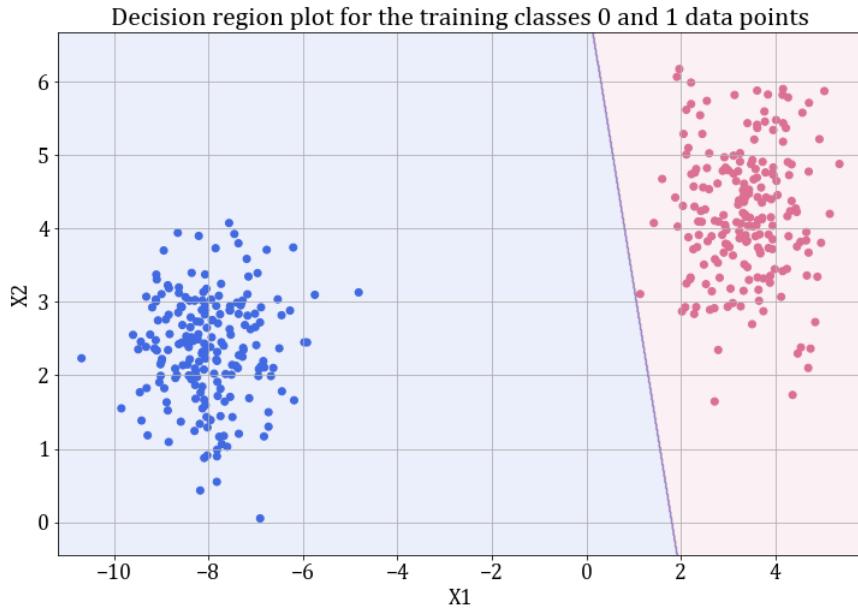


Figure 2: Decision region plot for classes 0 and 1 of data 1A using perceptron classifier

1.1.2 Classes 0 and 2

| Hyperparameter | Training Accuracy | CV Accuracy |
|----------------|-------------------|-------------|
| 0.001 | 1.0 | 1.0 |
| 0.005 | 1.0 | 1.0 |
| 0.01 | 1.0 | 1.0 |
| 0.05 | 1.0 | 1.0 |
| 0.1 | 1.0 | 1.0 |
| 1.0 | 1.0 | 1.0 |
| 5.0 | 1.0 | 1.0 |
| 10.0 | 1.0 | 1.0 |
| 100.0 | 1.0 | 1.0 |

Table 2: Table of training accuracies and CV accuracies for data of classes 0 and 2 of the 1A data set

The accuracy is 100% for all values of the hyperparameters. Taking the default value of 0.01, the accuracy on the test data is **100%**. The confusion matrices for the training and test data are as in [Figure 3](#).

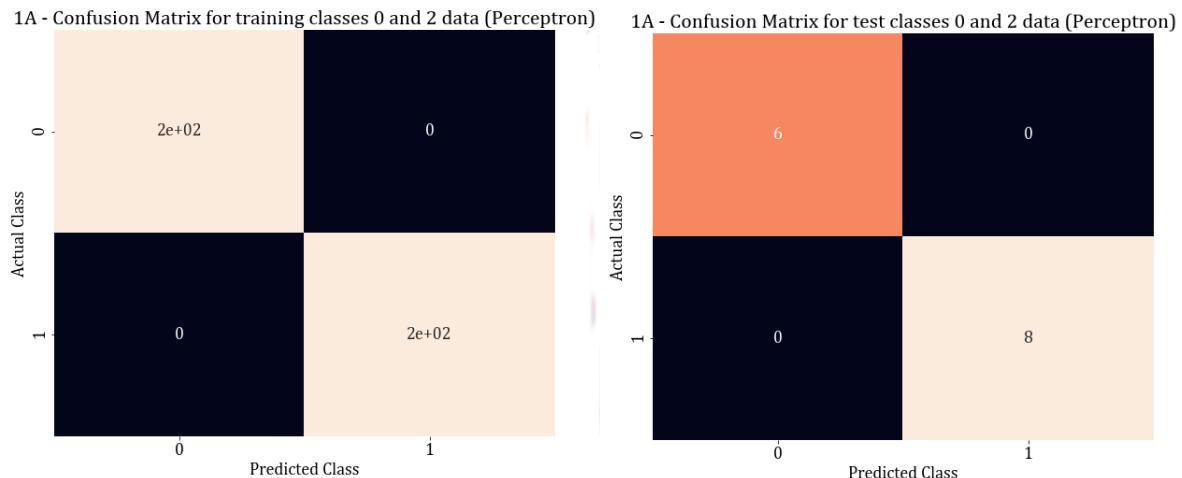


Figure 3: Confusion matrices for training and test data belonging to classes 0 and 2 of data 1A using perceptron classifier

The decision region plot for the perceptron classifier is in [Figure 4](#).



Figure 4: Decision region plot for classes 0 and 2 of data 1A using perceptron classifier

1.1.3 Classes 0 and 3

| Hyperparameter | Training Accuracy | CV Accuracy |
|----------------|-------------------|-------------|
| 0.001 | 1.0 | 1.0 |
| 0.005 | 1.0 | 1.0 |
| 0.01 | 1.0 | 1.0 |
| 0.05 | 1.0 | 1.0 |
| 0.1 | 1.0 | 1.0 |
| 1.0 | 1.0 | 1.0 |
| 5.0 | 1.0 | 1.0 |
| 10.0 | 1.0 | 1.0 |
| 100.0 | 1.0 | 1.0 |

Table 3: Table of training accuracies and CV accuracies for data of classes 0 and 3 of the 1A data set

The accuracy is 100% for all values of the hyperparameters. Taking the default value of 0.01, the accuracy on the test data is **100%**. The confusion matrices for the training and test data are as in [Figure 5](#).

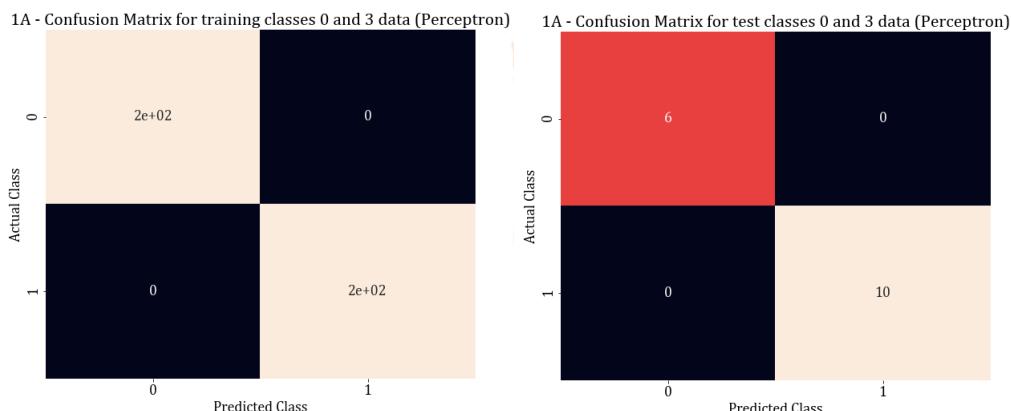


Figure 5: Confusion matrices for training and test data belonging to classes 0 and 3 of data 1A using perceptron classifier

The decision region plot for the perceptron classifier is in [Figure 6](#).



Figure 6: Decision region plot for classes 0 and 3 of data 1A using perceptron classifier

1.1.4 Classes 1 and 2

| Hyperparameter | Training Accuracy | CV Accuracy |
|----------------|-------------------|-------------|
| 0.001 | 1.0 | 0.975 |
| 0.005 | 1.0 | 0.975 |
| 0.01 | 1.0 | 0.975 |
| 0.05 | 1.0 | 1.0 |
| 0.1 | 1.0 | 1.0 |
| 1.0 | 1.0 | 1.0 |
| 5.0 | 1.0 | 1.0 |
| 10.0 | 1.0 | 1.0 |
| 100.0 | 1.0 | 1.0 |

Table 4: Table of training accuracies and CV accuracies for data of classes 1 and 2 of the 1A data set

The accuracy on the CV data is best for $\eta = 0.05$. Using this value for the perceptron model, the accuracy on the test data is **100%**. The confusion matrices for the training and test data are as in [Figure 7](#).

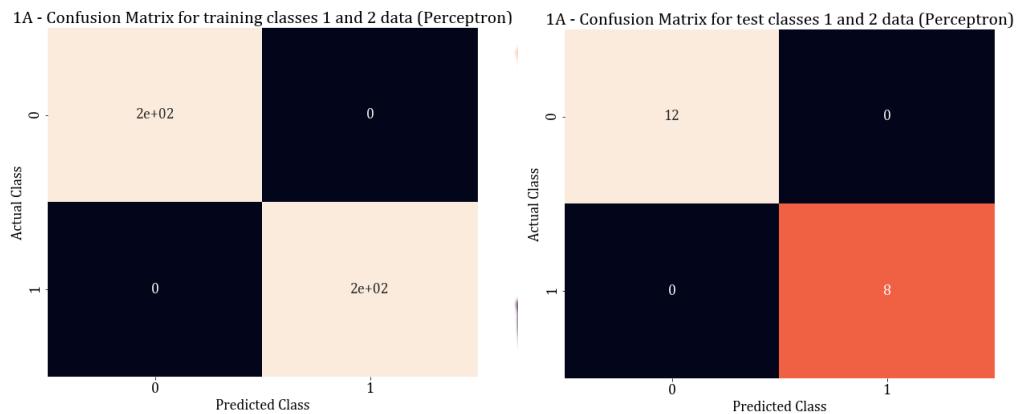


Figure 7: Confusion matrices for training and test data belonging to classes 1 and 2 of data 1A using perceptron classifier

The decision region plot for the perceptron classifier is in [Figure 8](#).



Figure 8: Decision region plot for classes 1 and 2 of data 1A using perceptron classifier

1.1.5 Classes 1 and 3

| Hyperparameter | Training Accuracy | CV Accuracy | Hyperparameter | Training Accuracy | CV Accuracy |
|----------------|-------------------|-------------|----------------|-------------------|-------------|
| 0.001 | 100.0 | 100.0 | 1.0 | 100.0 | 100.0 |
| 0.005 | 100.0 | 100.0 | 5.0 | 100.0 | 100.0 |
| 0.01 | 100.0 | 100.0 | 10.0 | 100.0 | 100.0 |
| 0.05 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 |
| 0.1 | 100.0 | 100.0 | | | |

Table 5: Table of training accuracies and CV accuracies for data of classes 1 and 3 of the 1A data set

The accuracy is 100% for all values of the hyperparameters. Taking the default value of 0.01, the accuracy on the test data is **100%**. The confusion matrices for the training and test data are as in [Figure 9](#).

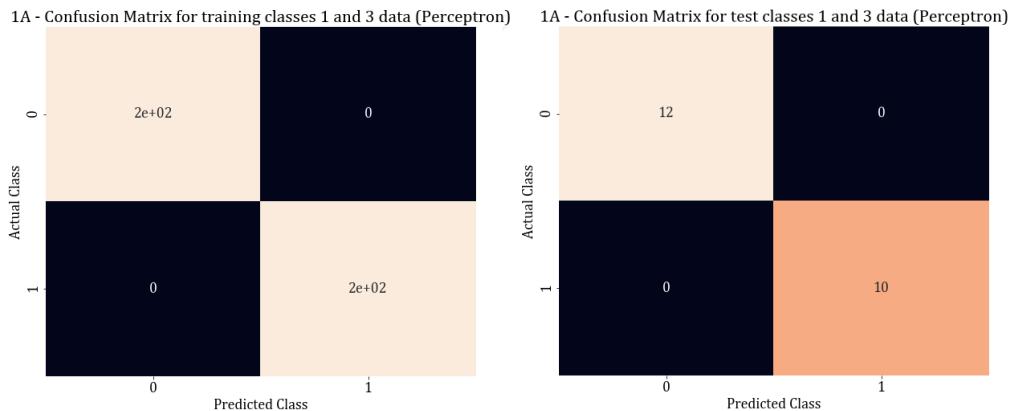


Figure 9: Confusion matrices for training and test data belonging to classes 1 and 3 of data 1A using perceptron classifier

The decision region plot for the perceptron classifier is in [Figure 10](#).



Figure 10: Decision region plot for classes 1 and 3 of data 1A using perceptron classifier

1.1.6 Classes 2 and 3

| Hyperparameter | Training Accuracy | CV Accuracy | Hyperparameter | Training Accuracy | CV Accuracy |
|----------------|-------------------|-------------|----------------|-------------------|-------------|
| 0.001 | 100.0 | 92.86 | 1.0 | 100.0 | 100.0 |
| 0.005 | 100.0 | 100.0 | 5.0 | 100.0 | 97.62 |
| 0.01 | 100.0 | 100.0 | 10.0 | 100.0 | 100.0 |
| 0.05 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 |
| 0.1 | 100.0 | 100.0 | | | |

Table 6: Table of training accuracies and CV accuracies for data of classes 2 and 3 of the 1A data set

The accuracy on the CV data is best for η greater than or equal to 0.005. Using the default value of $\eta = 0.01$ for the perceptron model, the accuracy on the test data is **100%**. The confusion matrices for the training and test data are as in [Figure 11](#).

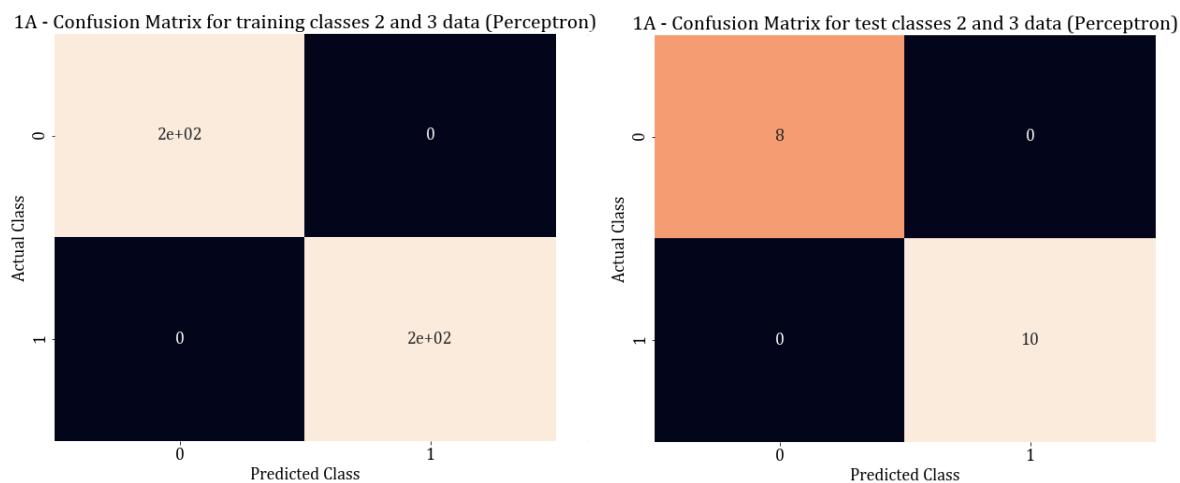


Figure 11: Confusion matrices for training and test data belonging to classes 2 and 3 of data 1A using perceptron classifier

The decision region plot for the perceptron classifier is in [Figure 12](#).



Figure 12: Decision region plot for classes 2 and 3 of data 1A using perceptron classifier

1.2 MLFFNN

The hyperparameters varied and swepted for are - hidden layer size, optimizer, batch size, learning rate, L2 regularization α .

1.2.1 Classification Accuracies

The classification accuracies on the training and validation datasets (30% of the [dev.csv](#)) are as follows:

| # Neurons | Activation | Solver | Batch Size | α | Learning Rate | Accuracy | Validation Accuracy |
|-----------|------------|--------|------------|----------|---------------|----------|---------------------|
| 5 | tanh | lbfgs | 200 | 0.0001 | adaptive | 100.0 | 100.0 |
| 5 | tanh | lbfgs | 200 | 0.0001 | constant | 100.0 | 100.0 |
| 5 | tanh | lbfgs | 200 | 0.0 | invscaling | 100.0 | 100.0 |
| 5 | tanh | lbfgs | 200 | 0.0 | adaptive | 100.0 | 100.0 |
| 5 | tanh | lbfgs | 200 | 0.0 | constant | 100.0 | 100.0 |
| 5 | tanh | lbfgs | 100 | 0.0 | adaptive | 100.0 | 100.0 |
| 5 | tanh | lbfgs | 100 | 0.0001 | invscaling | 100.0 | 100.0 |
| 5 | relu | lbfgs | 200 | 0.0 | constant | 100.0 | 100.0 |
| 5 | relu | lbfgs | 100 | 0.0001 | invscaling | 100.0 | 100.0 |
| 5 | relu | lbfgs | 200 | 0.0 | adaptive | 100.0 | 100.0 |

Table 2: Best 10 Train and Validation Accuracies obtained after performing a [GridSearch](#) on 432 parameter combinations.

1.2.2 Best Model

The parameter combination were additionally sorted based on minimum fitting time (least fitting time - first) and the model that gave the best accuracy the fastest (and potentially the most minimal model that best fits the data), was chosen. Hence the best parameter combination chosen is:

- hidden_layer_sizes: 5
- activation: tanh
- solver: lbfgs
- batch_size: 200

- alpha: 0.0001
- learning_rate: adaptive

The classification accuracy of the best model on the testing data is: 100%. The confusion matrices obtained are as follows:

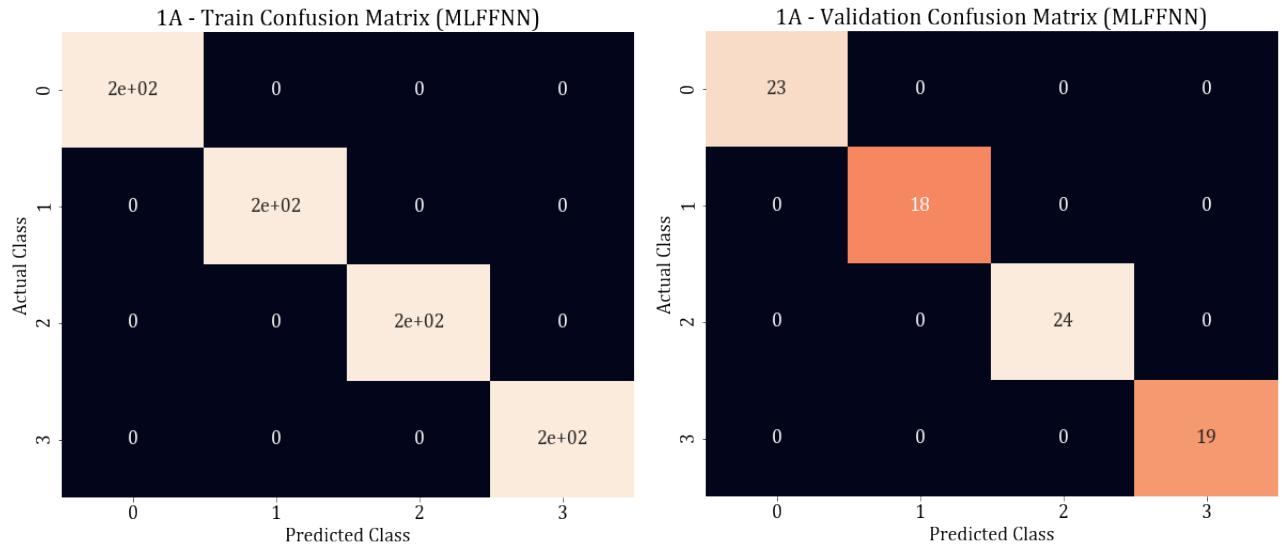


Figure 13: Training and Validation confusion matrices obtained for the best parameter combination, on the left and right respectively.

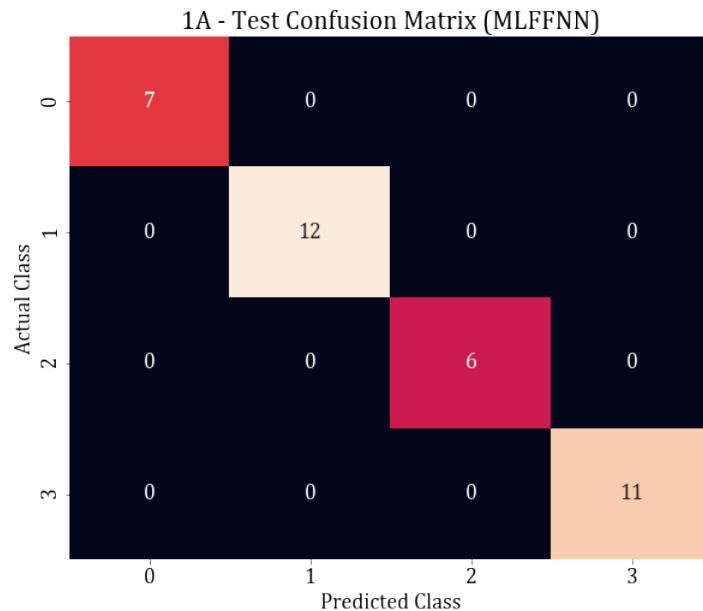


Figure 14: Testing confusion matrices obtained for the best parameter combination.

1.2.3 Decision Region

The decision region plots obtained is as follows:

1A - Decision Region Plot (MLFFNN)

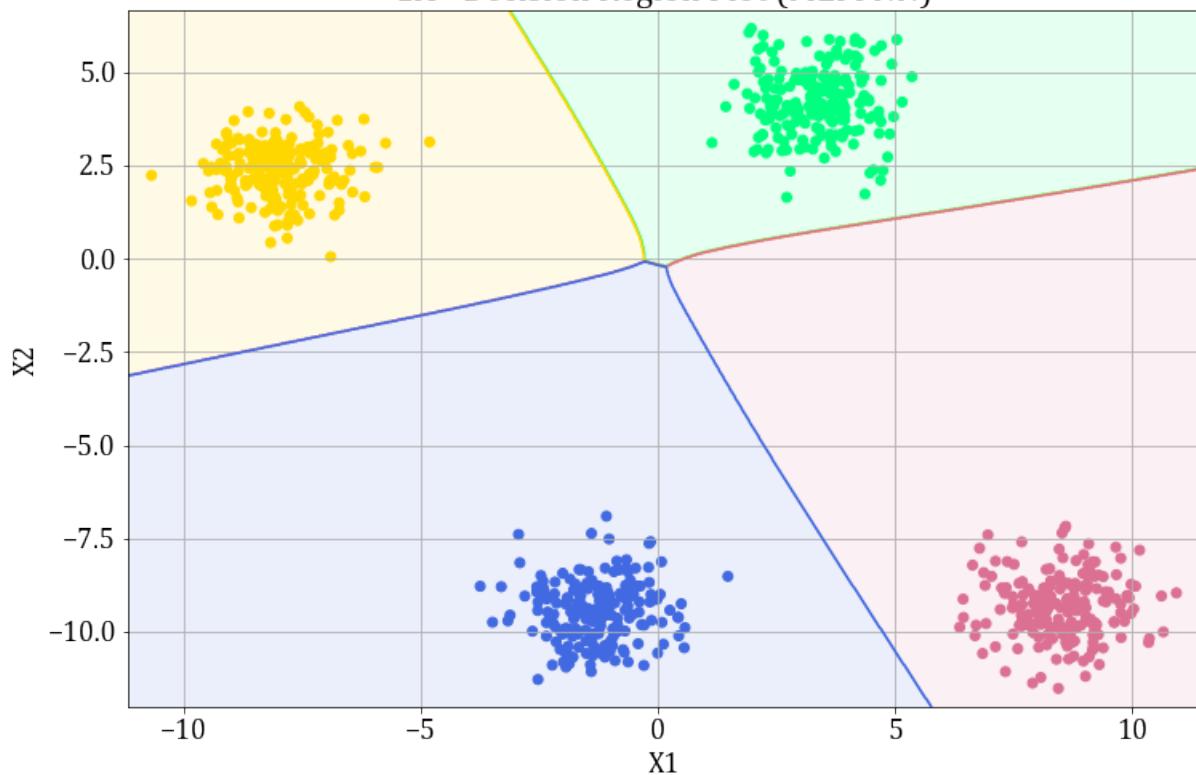


Figure 15: Decision Region Plot obtained for the best parameter combination.

1.3 Linear SVM

2 Dataset 1B

This dataset contains data for three classes - 0, 1 and 2. The classes are non-linearly separable and the dimension of the feature space is 2.

2.1 MLFFNN

The hyperparameters varied and swepted for are - hidden layer size, activation function, batch size, learning rate, L2 regularization α .

2.1.1 Classification Accuracies

The classification accuracies on the training and validation datasets (30% of the [dev.csv](#)) are as follows:

| # Neurons | Activation | Batch Size | Early Stopping | Learning Rate | α | Accuracy | Validation Accuracy |
|-----------|------------|------------|----------------|---------------|----------|----------|---------------------|
| (8, 8) | relu | 50 | False | adaptive | 0.01 | 99.33 | 98.41 |
| (8, 8) | relu | 50 | False | constant | 0.001 | 99.33 | 98.41 |
| (8, 8) | relu | 50 | False | invscaling | 0.01 | 99.33 | 98.41 |
| (8, 8) | relu | 50 | False | adaptive | 0.001 | 99.33 | 98.41 |
| (8, 8) | relu | 50 | False | invscaling | 0.001 | 99.33 | 98.41 |
| (8, 8) | relu | 50 | False | constant | 0.01 | 99.33 | 98.41 |
| (10, 10) | relu | 50 | False | adaptive | 0.01 | 99.0 | 98.41 |
| (10, 10) | relu | 50 | False | constant | 0.01 | 99.0 | 98.41 |
| (10, 10) | relu | 50 | False | invscaling | 0.01 | 99.0 | 98.41 |
| (10, 10) | relu | 50 | False | constant | 0.001 | 99.0 | 96.82 |

Table 3: Best 10 Train and Validation Accuracies obtained after performing a [GridSearch](#) on 432 parameter combinations.

2.1.2 Best Model

The parameter combination were additionally sorted based on minimum fitting time (least fitting time - first) and the model that gave the best accuracy the fastest was chosen. Hence the best parameter combination chosen is:

- hidden_layer_sizes: (8, 8)
- activation: relu
- batch_size: 50
- early_stopping: False
- learning_rate: adaptive
- alpha (L2 regularization): 0.01

The classification accuracy of the best model on the testing data is: 96.296%. The confusion matrices obtained are as follows:

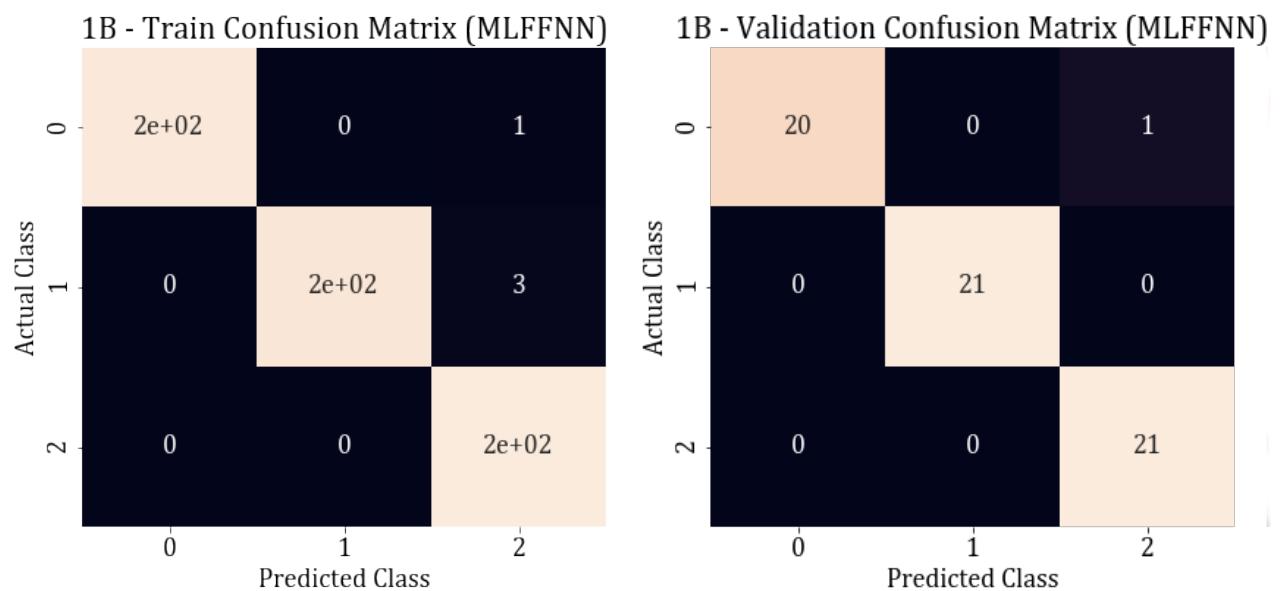


Figure 16: Training and Validation confusion matrices obtained for the best parameter combination, on the left and right respectively.

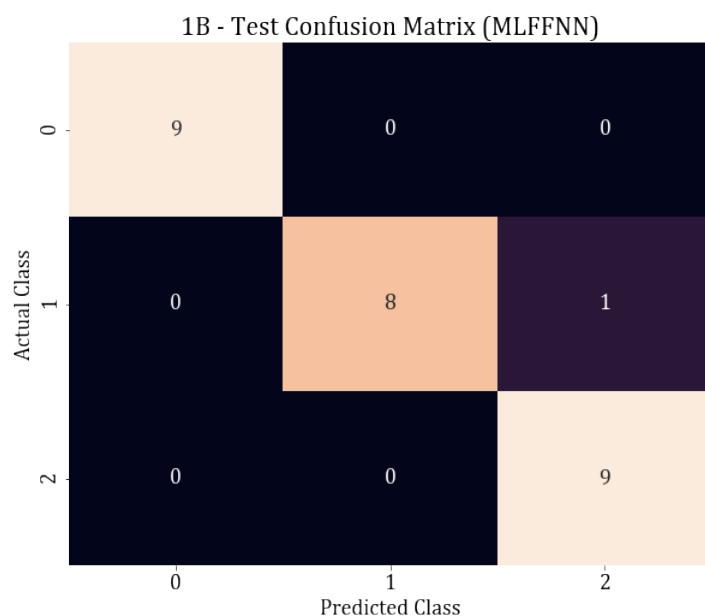


Figure 17: Testing confusion matrices obtained for the best parameter combination.

2.1.3 Decision Region

The decision region plots obtained is as follows:

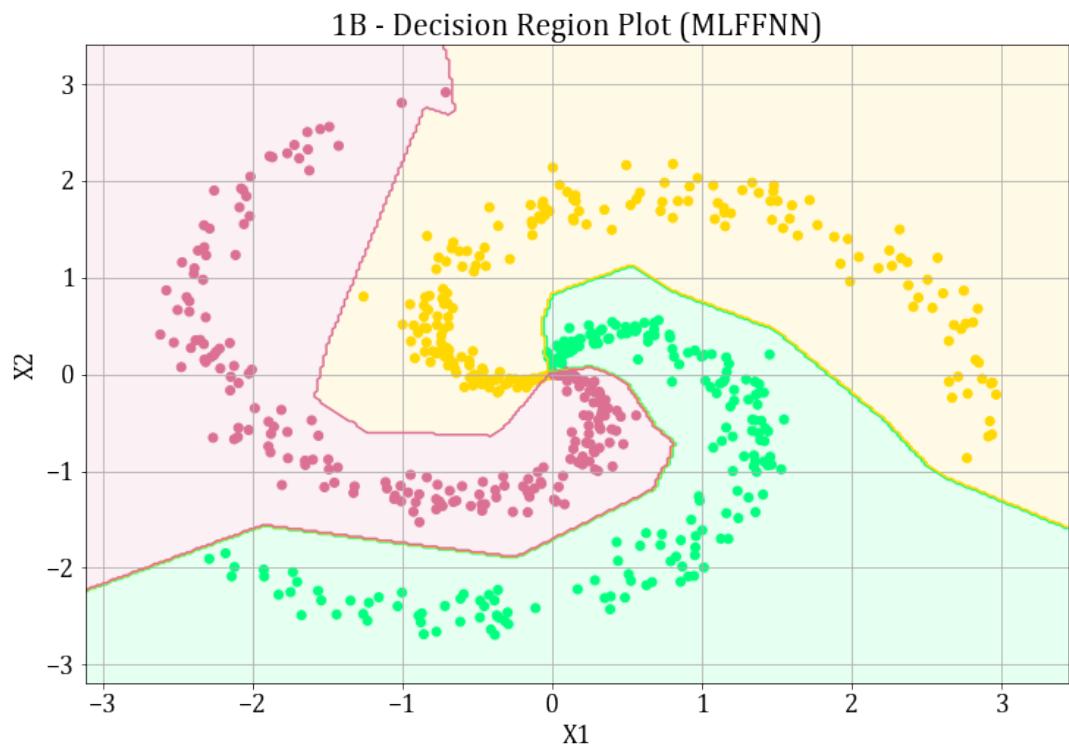


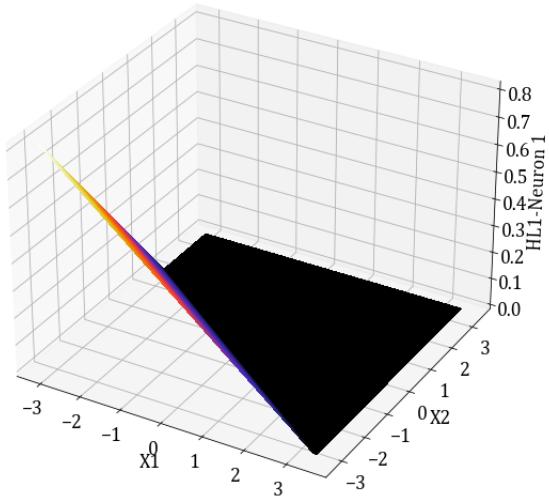
Figure 18: Decision Region Plot obtained for the best parameter combination.

2.1.4 Surface Plots

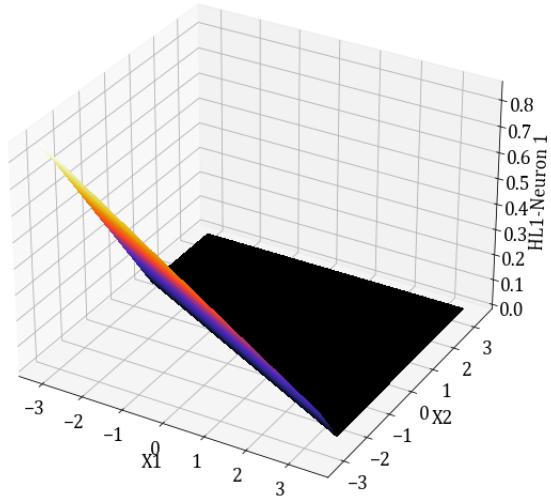
The neuron-wise surface plots obtained for the hidden and output layers is as follows:

2.1.4.1 Hidden Layer 1, Node 1

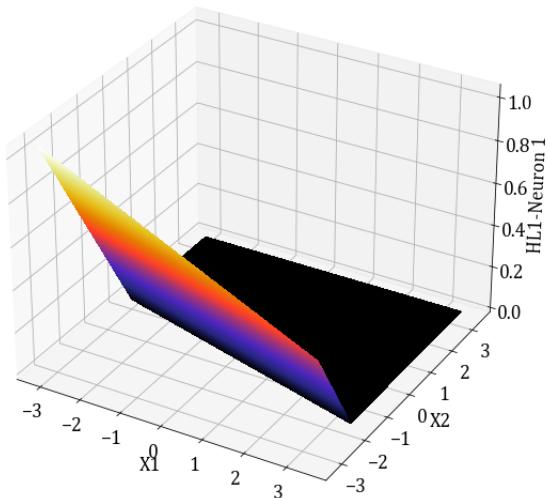
Epoch: 1; Surface for Layer 1, Neuron 1



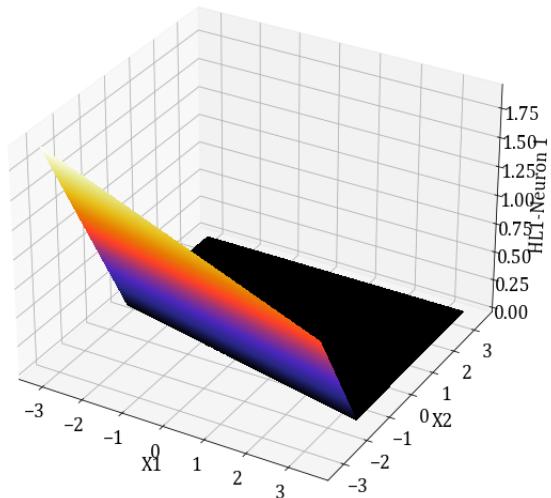
Epoch: 5; Surface for Layer 1, Neuron 1



Epoch: 20; Surface for Layer 1, Neuron 1



Epoch: 100; Surface for Layer 1, Neuron 1



Converged; Surface for Layer 1, Neuron 1

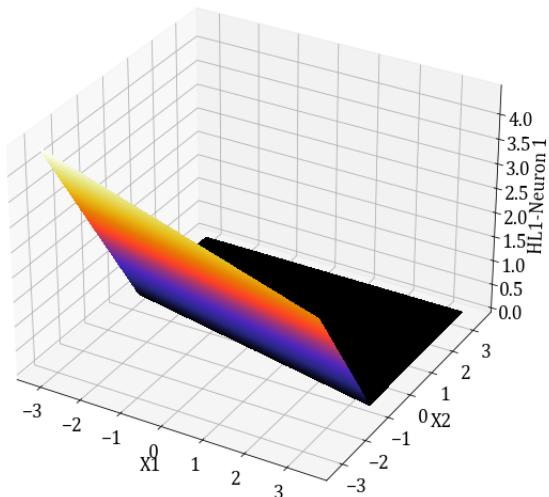
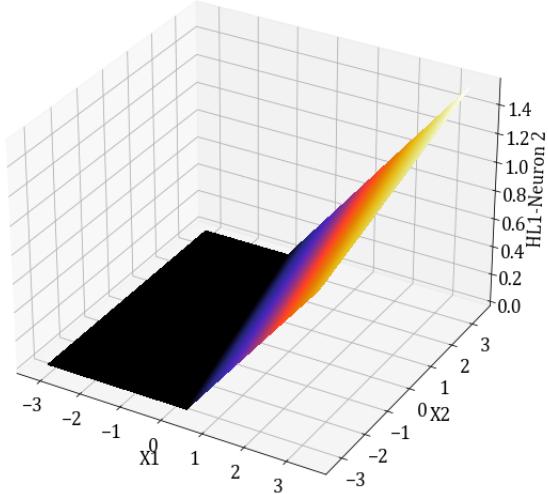


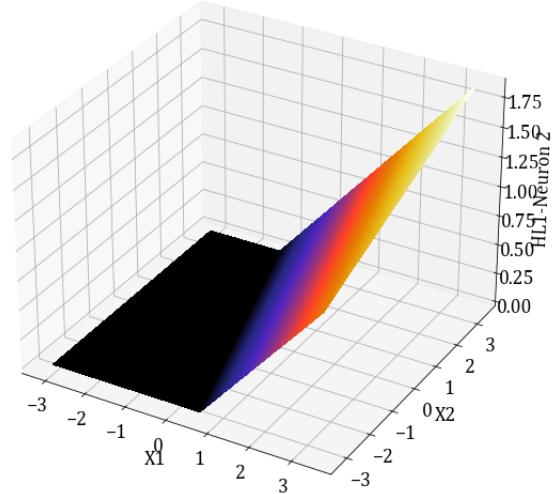
Figure 19: Surface Plots obtained for Hidden Layer 1, Neuron 1, across epochs.

2.1.4.2 Hidden Layer 1, Node 2

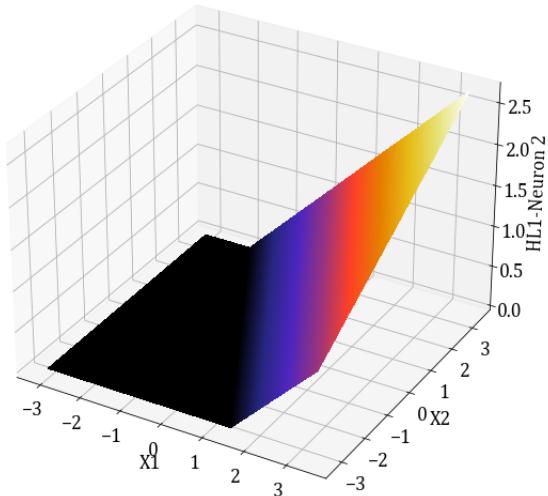
Epoch: 1; Surface for Layer 1, Neuron 2



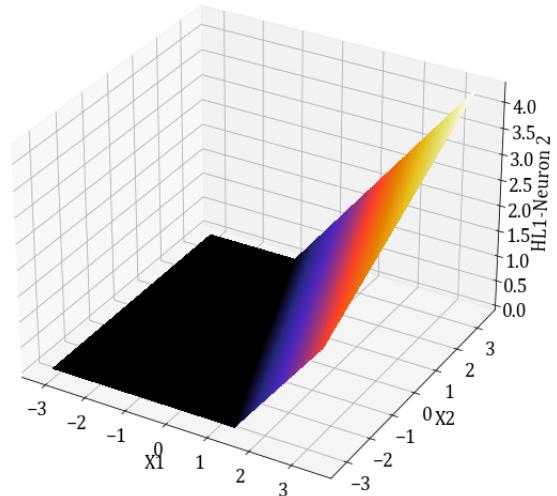
Epoch: 5; Surface for Layer 1, Neuron 2



Epoch: 20; Surface for Layer 1, Neuron 2



Epoch: 100; Surface for Layer 1, Neuron 2



Converged; Surface for Layer 1, Neuron 2

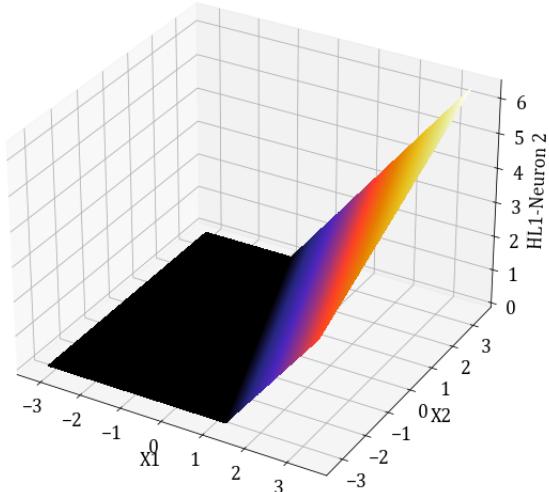
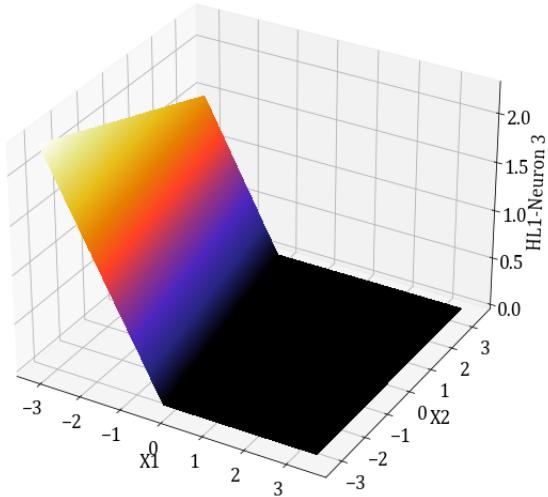


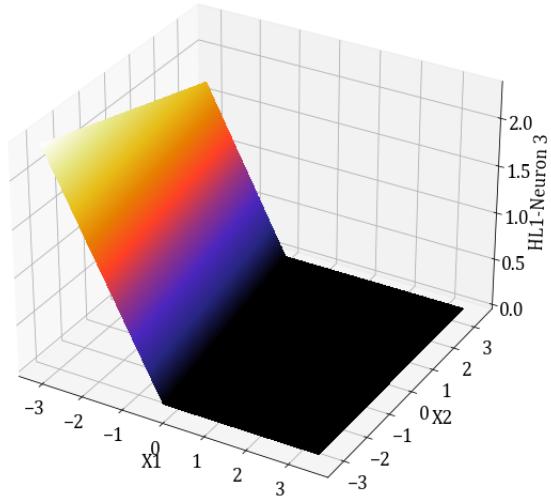
Figure 20: Surface Plots obtained for Hidden Layer 1, Neuron 2, across epochs.

2.1.4.3 Hidden Layer 1, Node 3

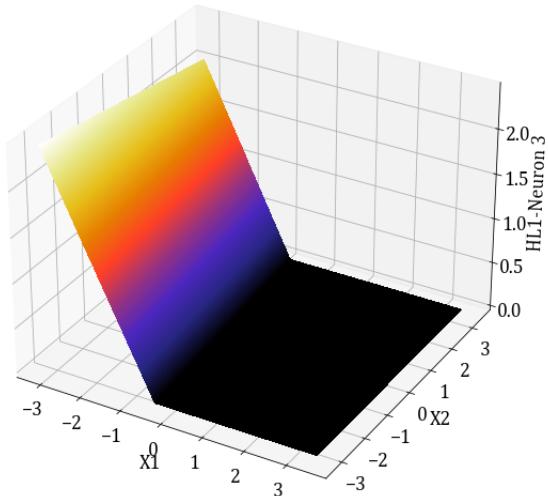
Epoch: 1; Surface for Layer 1, Neuron 3



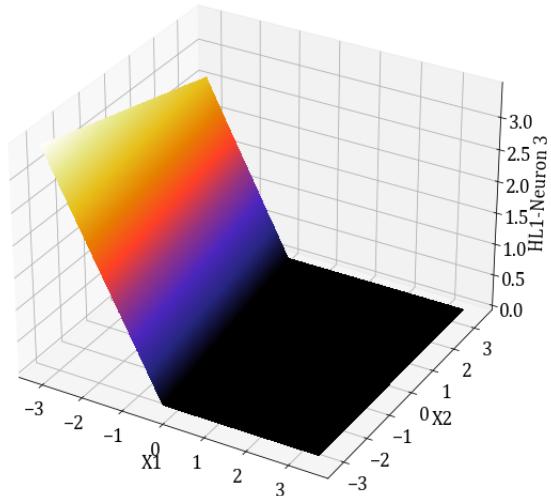
Epoch: 5; Surface for Layer 1, Neuron 3



Epoch: 20; Surface for Layer 1, Neuron 3



Epoch: 100; Surface for Layer 1, Neuron 3



Converged; Surface for Layer 1, Neuron 3

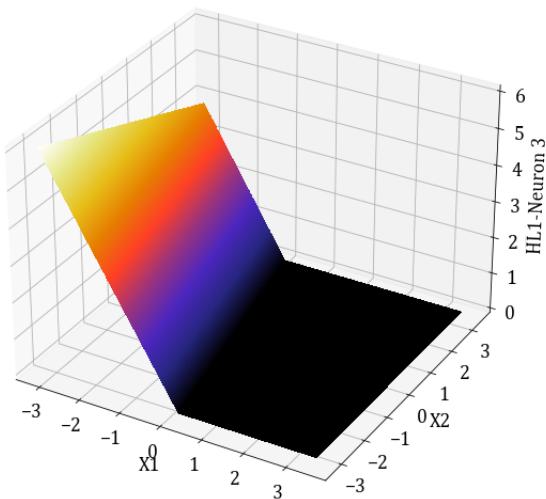


Figure 21: Surface Plots obtained for Hidden Layer 1, Neuron 3, across epochs.

2.1.4.4 Hidden Layer 1, Node 4

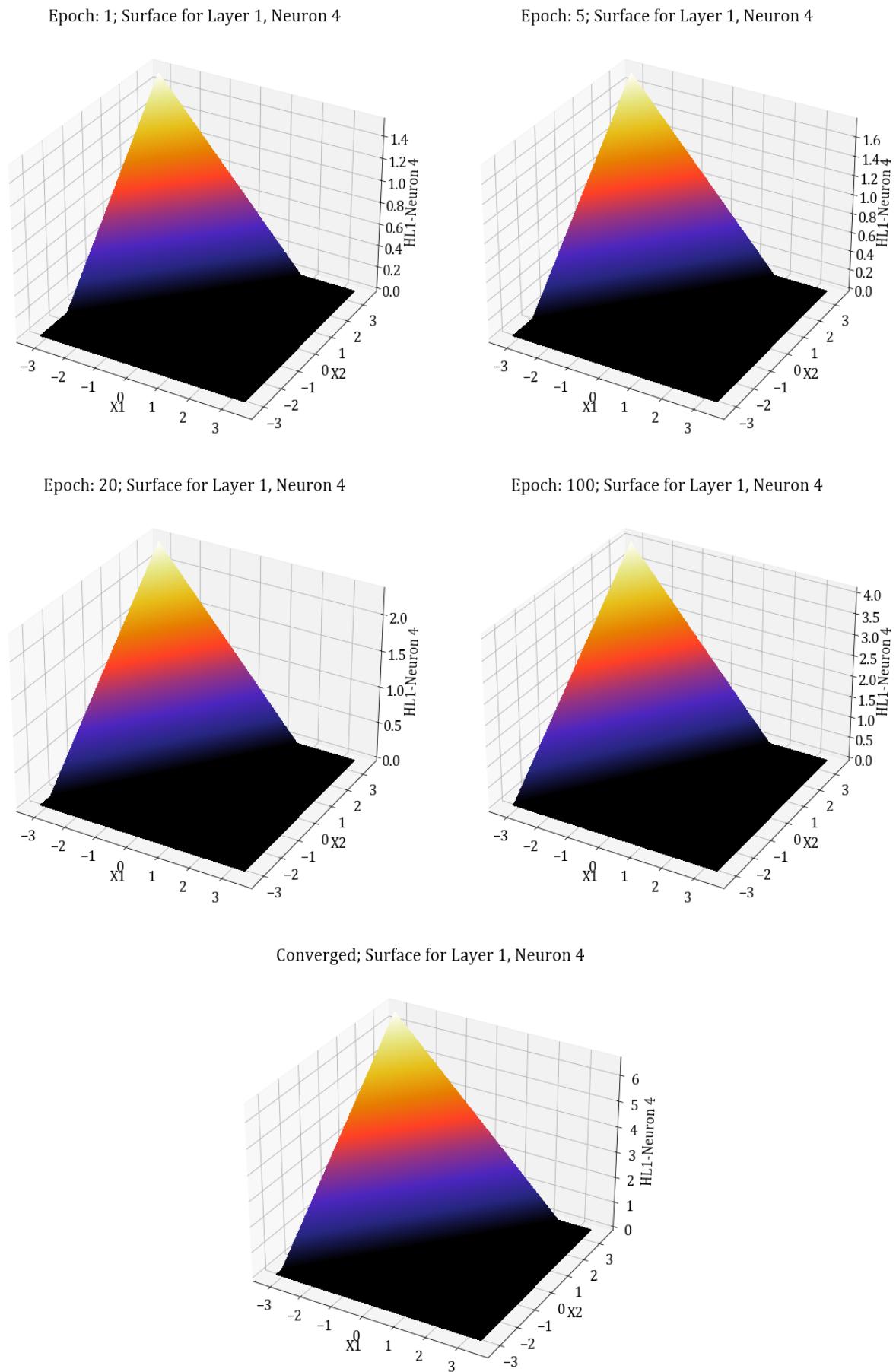
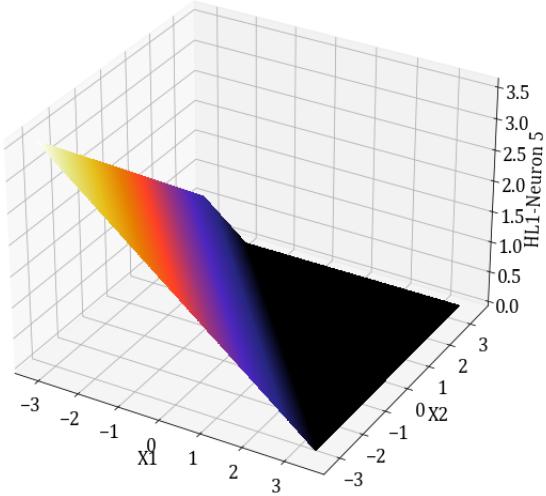


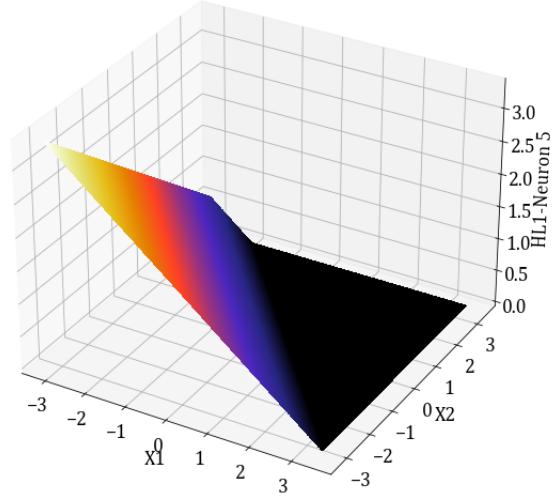
Figure 22: Surface Plots obtained for Hidden Layer 1, Neuron 4, across epochs.

2.1.4.5 Hidden Layer 1, Node 5

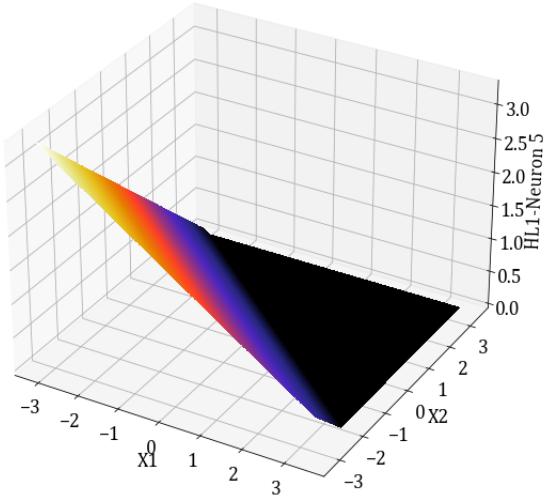
Epoch: 1; Surface for Layer 1, Neuron 5



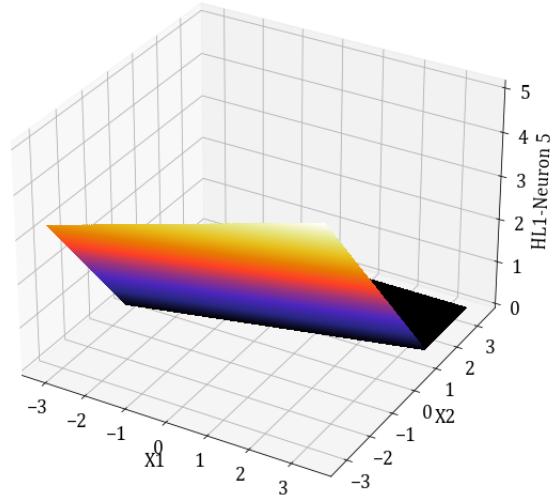
Epoch: 5; Surface for Layer 1, Neuron 5



Epoch: 20; Surface for Layer 1, Neuron 5



Epoch: 100; Surface for Layer 1, Neuron 5



Converged; Surface for Layer 1, Neuron 5

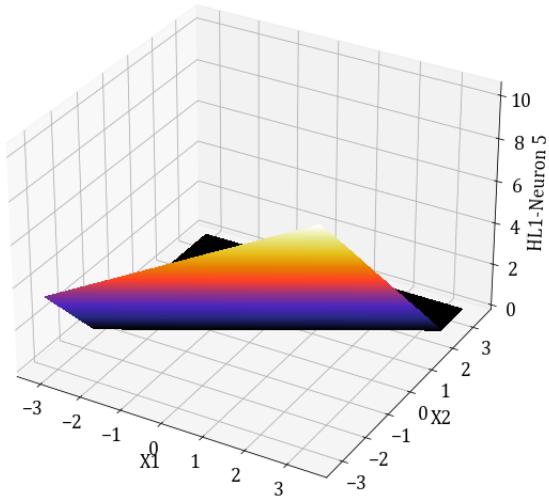


Figure 23: Surface Plots obtained for Hidden Layer 1, Neuron 5, across epochs.

2.1.4.6 Hidden Layer 1, Node 6

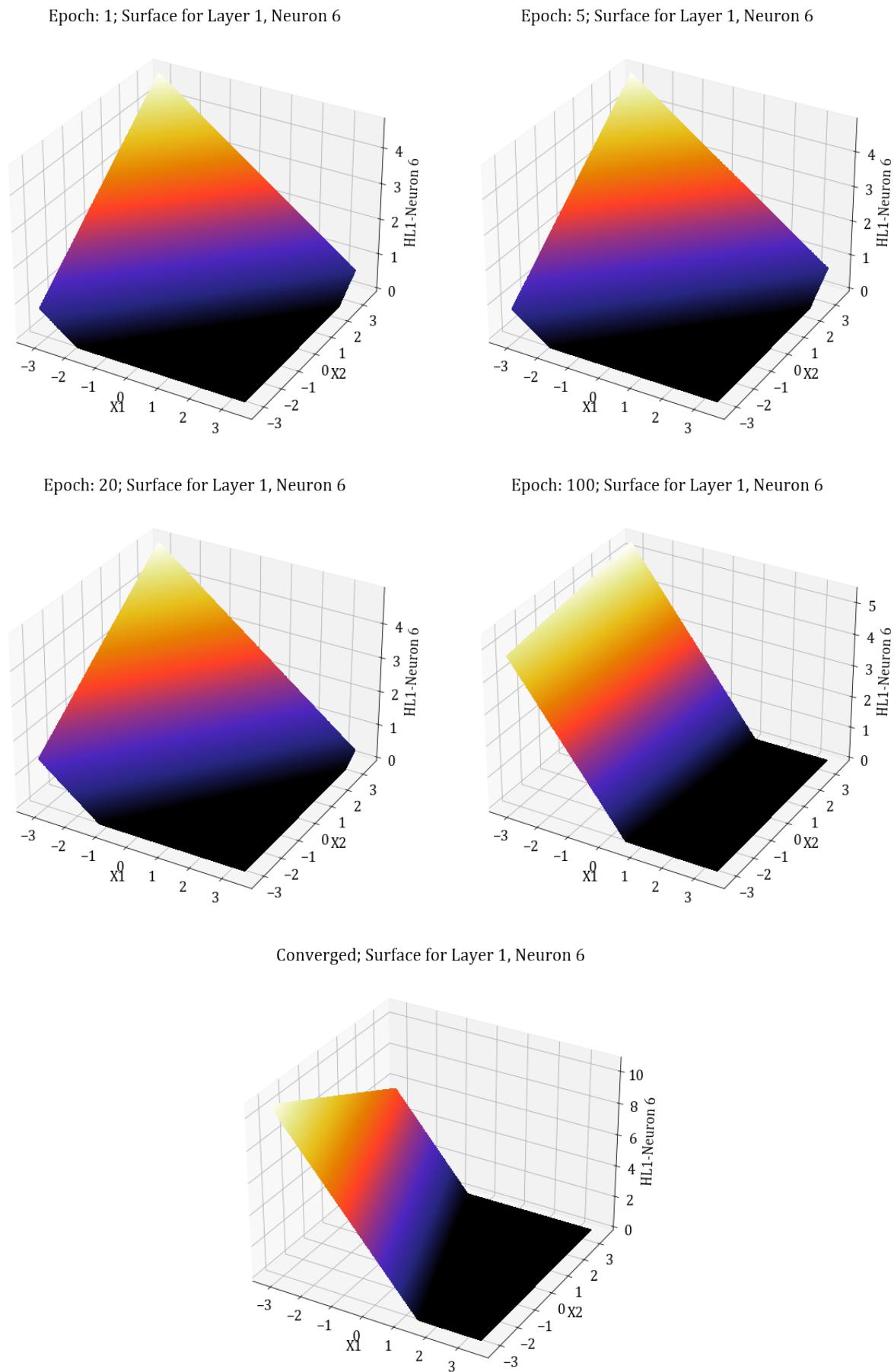
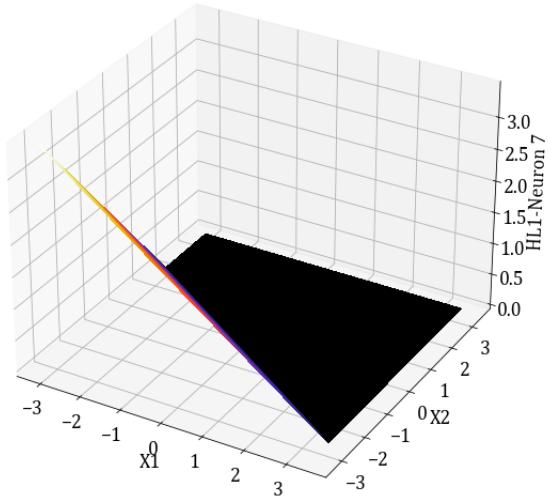


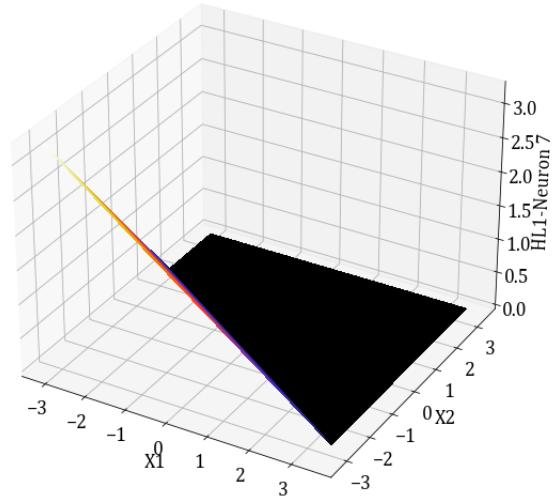
Figure 24: Surface Plots obtained for Hidden Layer 1, Neuron 6, across epochs.

2.1.4.7 Hidden Layer 1, Node 7

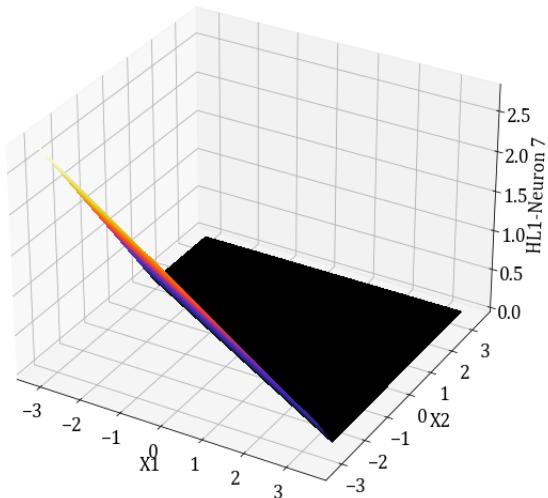
Epoch: 1; Surface for Layer 1, Neuron 7



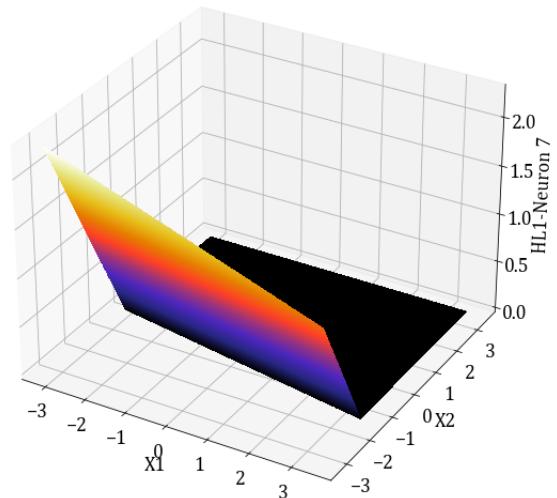
Epoch: 5; Surface for Layer 1, Neuron 7



Epoch: 20; Surface for Layer 1, Neuron 7



Epoch: 100; Surface for Layer 1, Neuron 7



Converged; Surface for Layer 1, Neuron 7

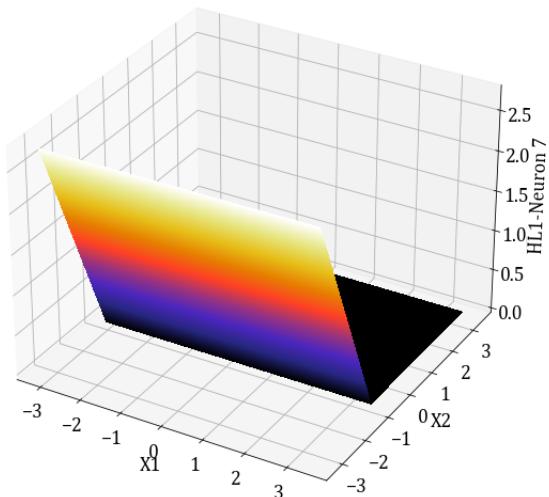


Figure 25: Surface Plots obtained for Hidden Layer 1, Neuron 7, across epochs.

2.1.4.8 Hidden Layer 1, Node 8

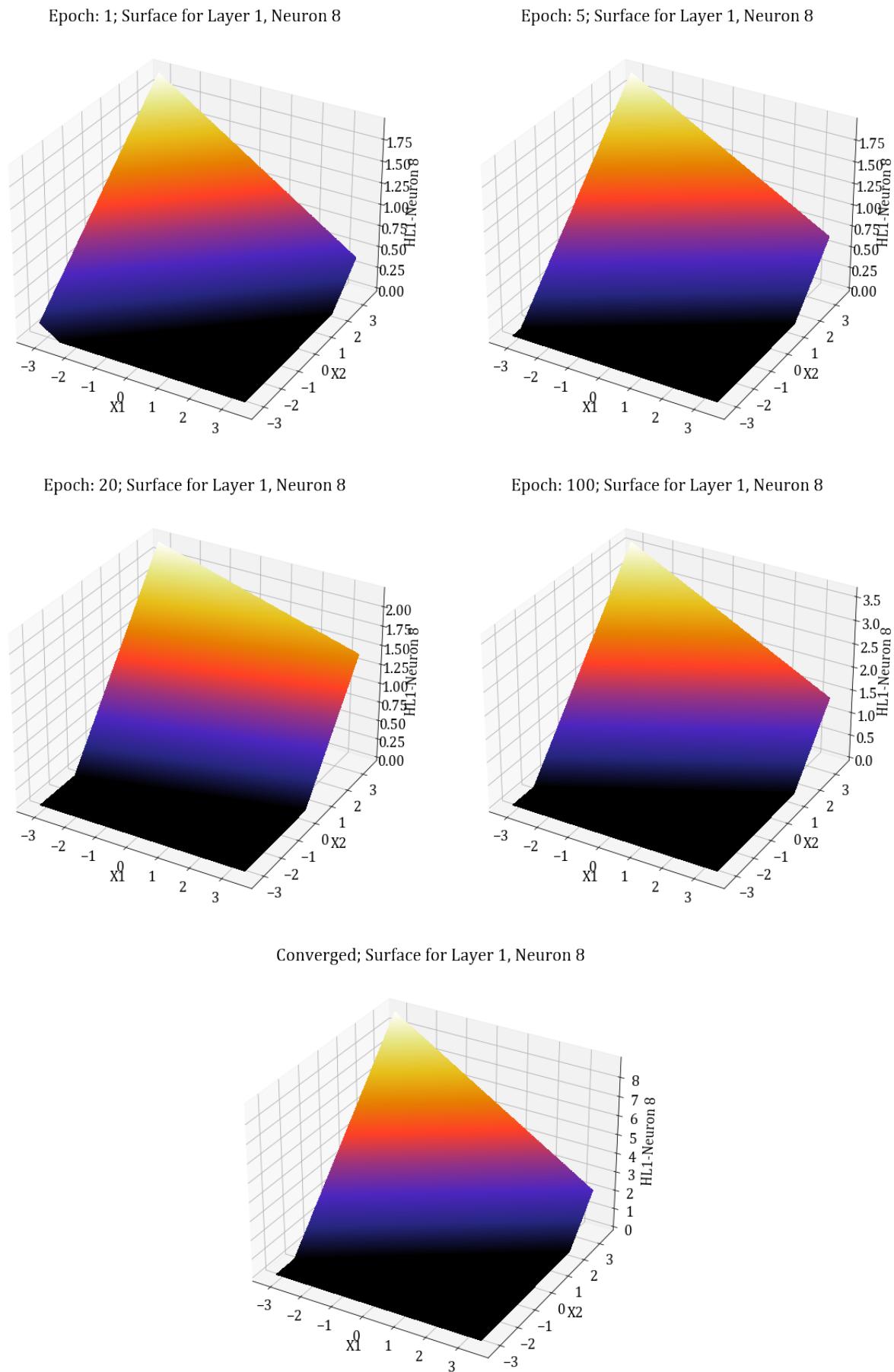
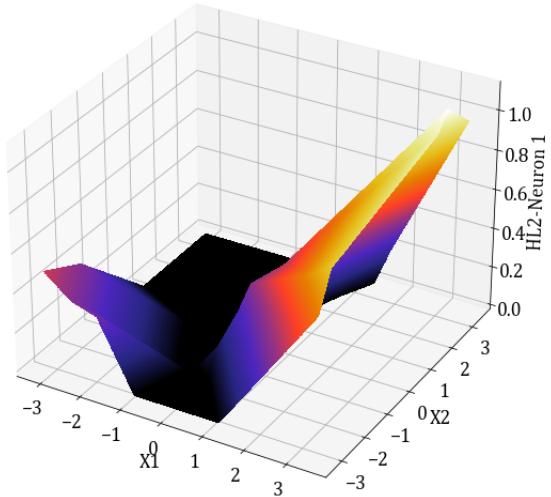


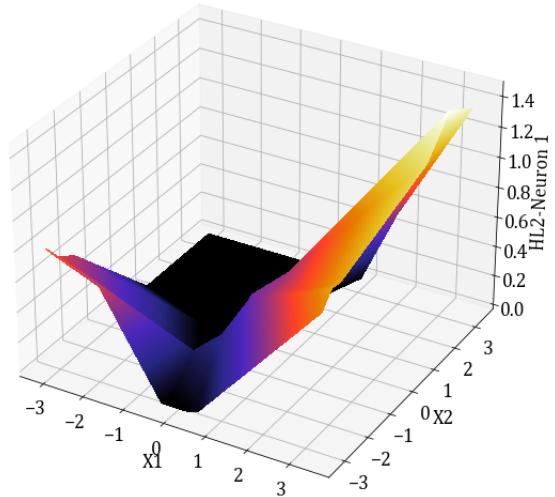
Figure 26: Surface Plots obtained for Hidden Layer 1, Neuron 8, across epochs.

2.1.4.9 Hidden Layer 2, Node 1

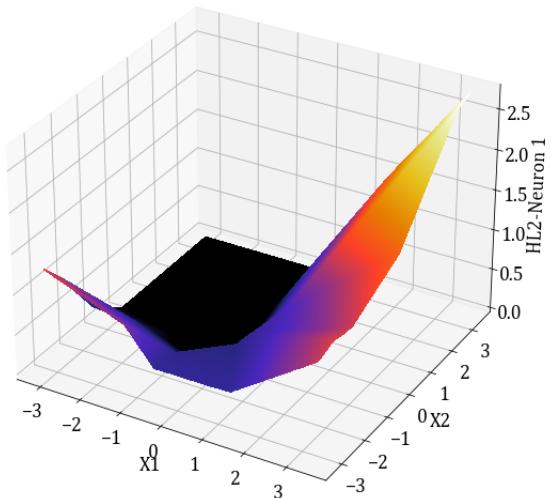
Epoch: 1; Surface for Layer 2, Neuron 1



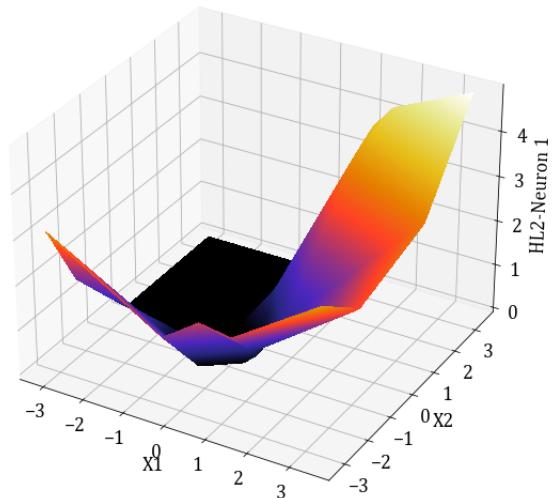
Epoch: 5; Surface for Layer 2, Neuron 1



Epoch: 20; Surface for Layer 2, Neuron 1



Epoch: 100; Surface for Layer 2, Neuron 1



Converged; Surface for Layer 2, Neuron 1

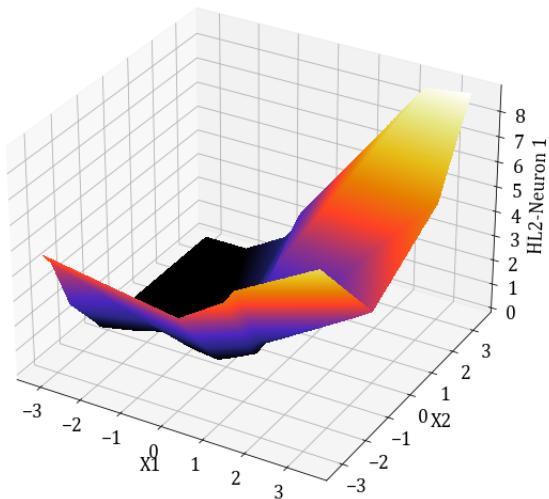
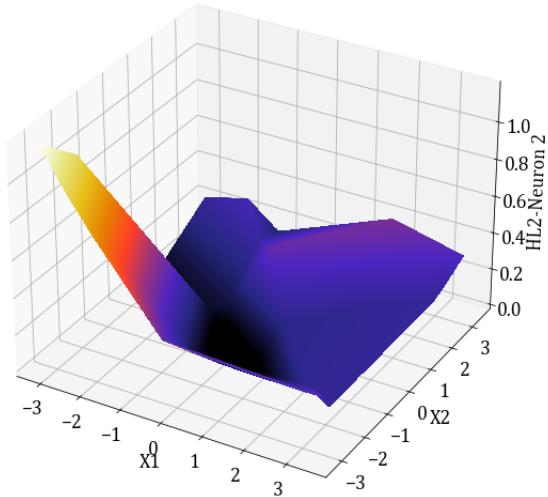


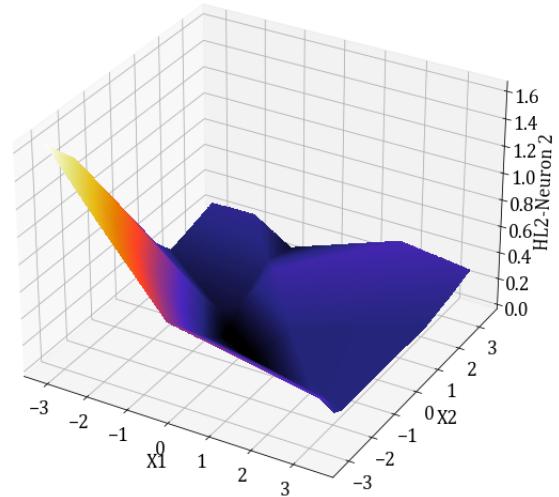
Figure 27: Surface Plots obtained for Hidden Layer 2, Neuron 1, across epochs.

2.1.4.10 Hidden Layer 2, Node 2

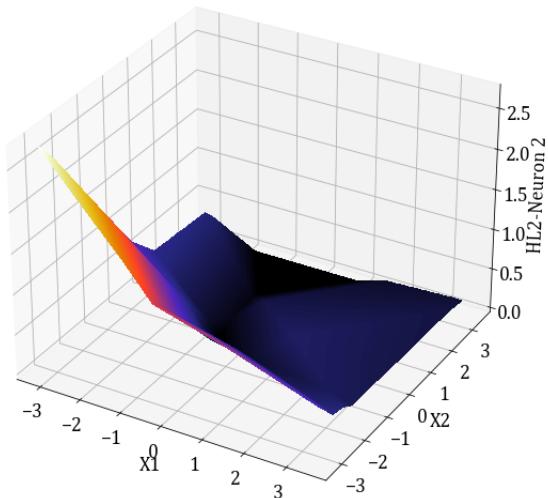
Epoch: 1; Surface for Layer 2, Neuron 2



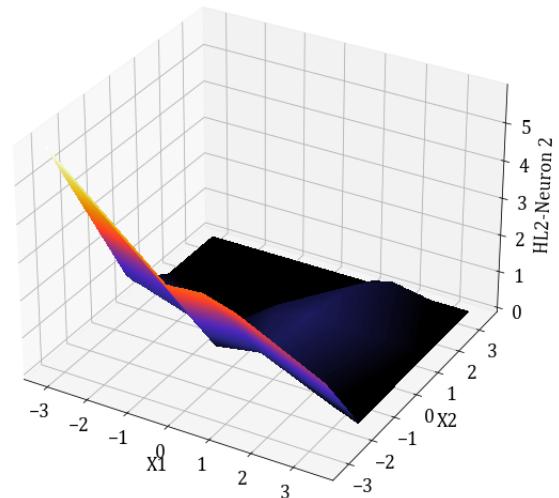
Epoch: 5; Surface for Layer 2, Neuron 2



Epoch: 20; Surface for Layer 2, Neuron 2



Epoch: 100; Surface for Layer 2, Neuron 2



Converged; Surface for Layer 2, Neuron 2

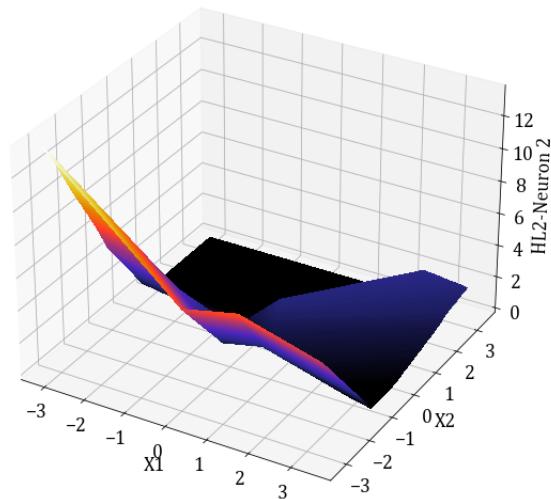
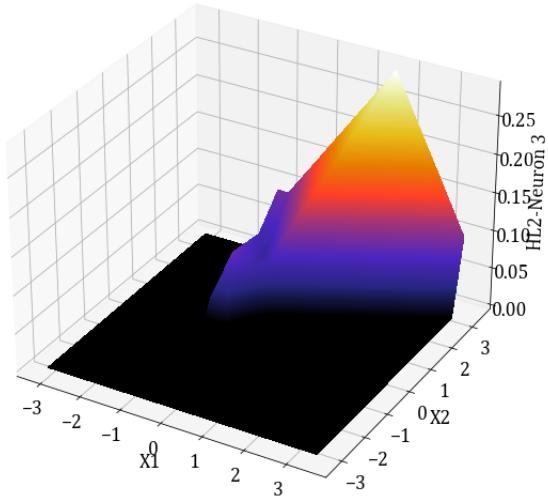


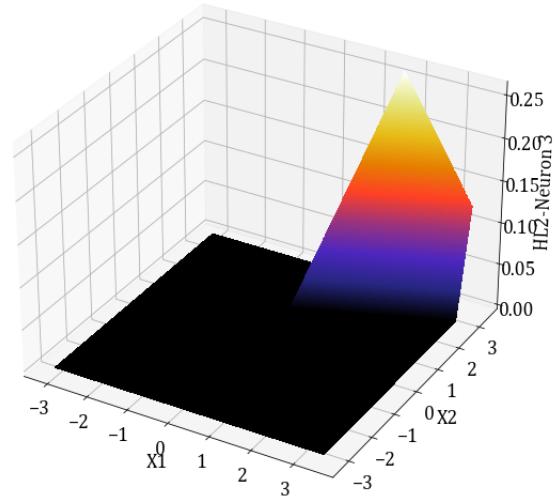
Figure 28: Surface Plots obtained for Hidden Layer 2, Neuron 2, across epochs.

2.1.4.11 Hidden Layer 2, Node 3

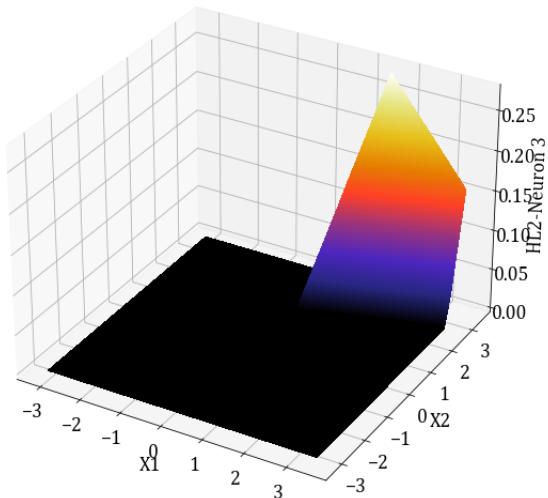
Epoch: 1; Surface for Layer 2, Neuron 3



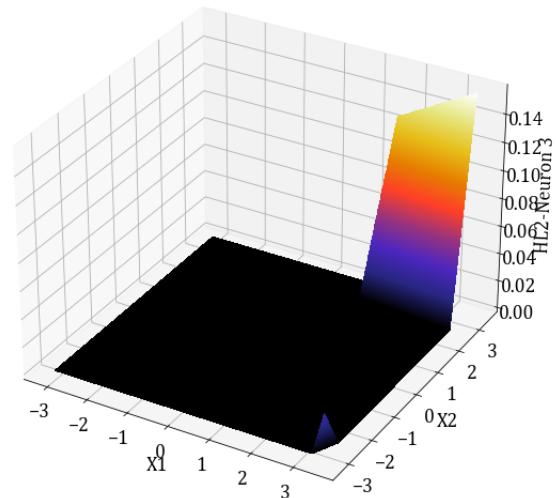
Epoch: 5; Surface for Layer 2, Neuron 3



Epoch: 20; Surface for Layer 2, Neuron 3



Epoch: 100; Surface for Layer 2, Neuron 3



Converged; Surface for Layer 2, Neuron 3

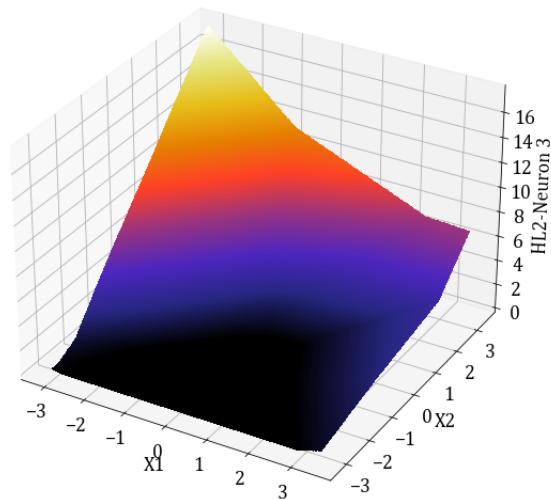
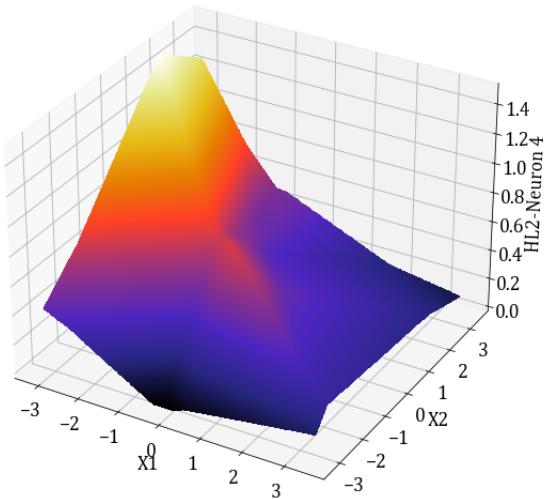


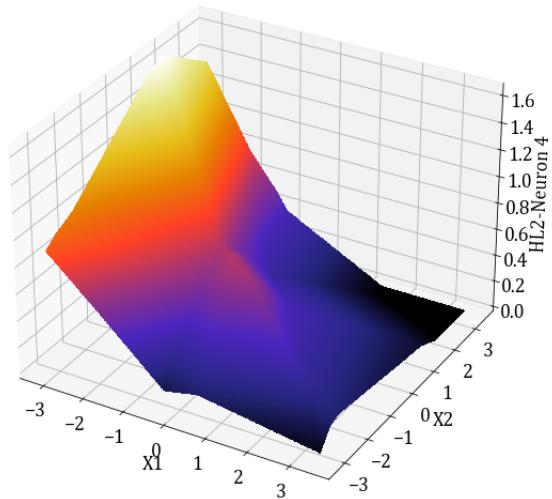
Figure 29: Surface Plots obtained for Hidden Layer 2, Neuron 3, across epochs.

2.1.4.12 Hidden Layer 2, Node 4

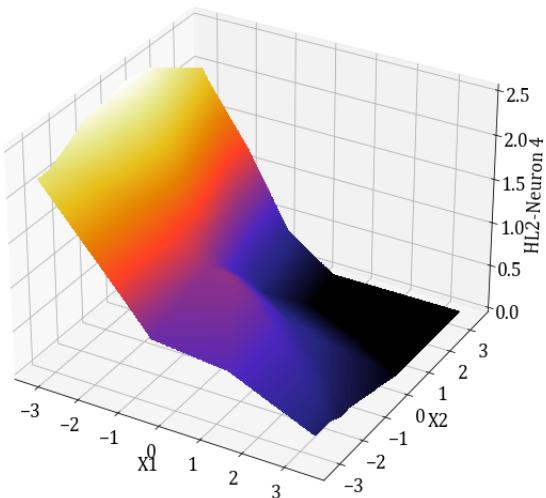
Epoch: 1; Surface for Layer 2, Neuron 4



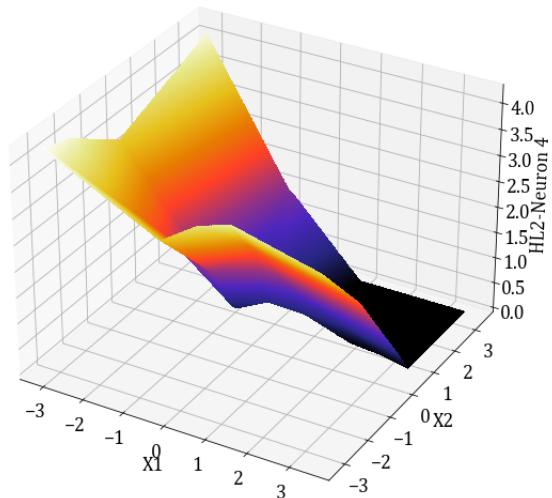
Epoch: 5; Surface for Layer 2, Neuron 4



Epoch: 20; Surface for Layer 2, Neuron 4



Epoch: 100; Surface for Layer 2, Neuron 4



Converged; Surface for Layer 2, Neuron 4

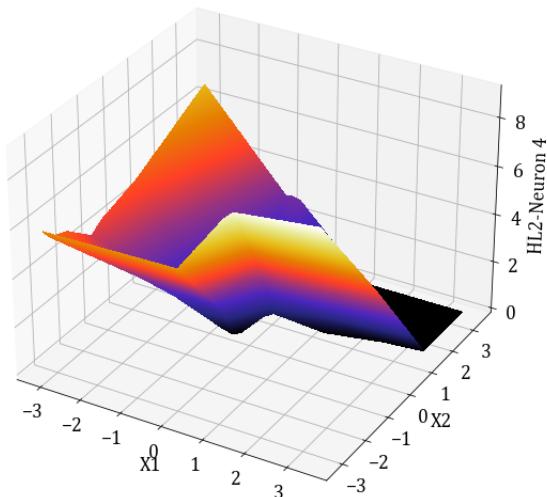
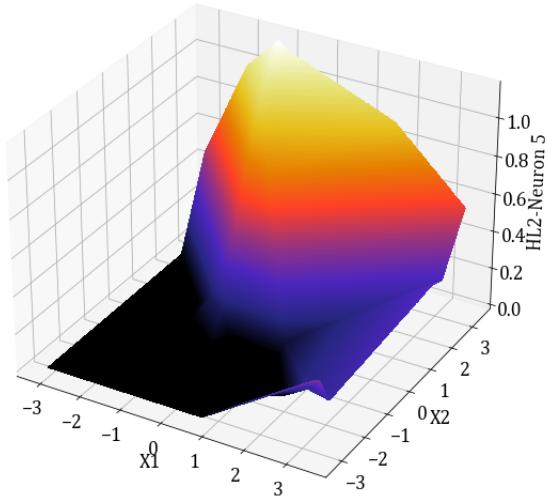


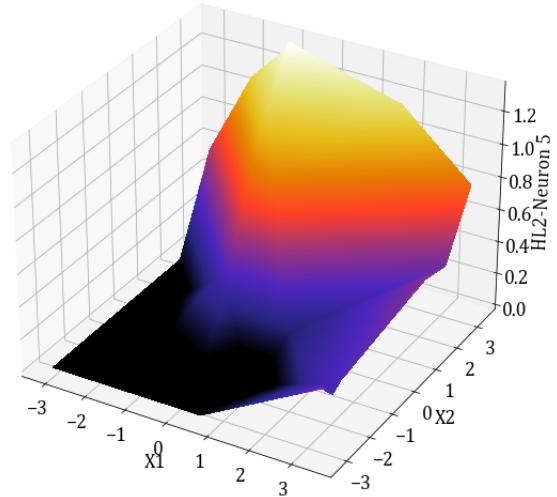
Figure 30: Surface Plots obtained for Hidden Layer 2, Neuron 4, across epochs.

2.1.4.13 Hidden Layer 2, Node 5

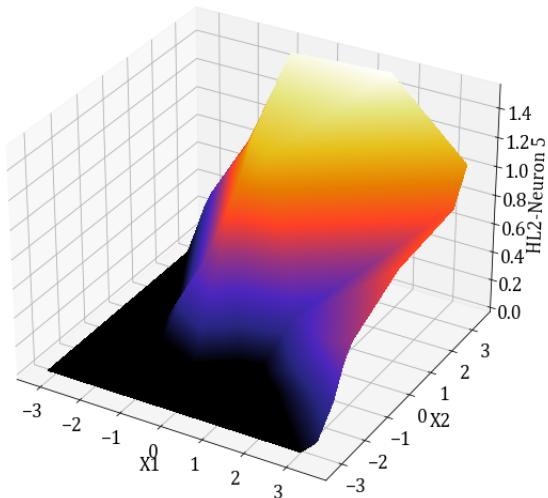
Epoch: 1; Surface for Layer 2, Neuron 5



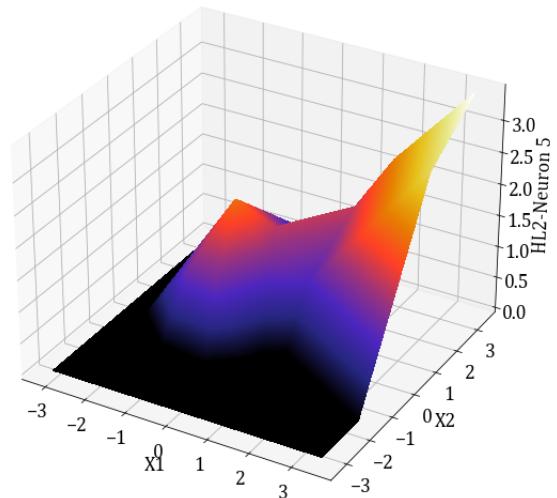
Epoch: 5; Surface for Layer 2, Neuron 5



Epoch: 20; Surface for Layer 2, Neuron 5



Epoch: 100; Surface for Layer 2, Neuron 5



Converged; Surface for Layer 2, Neuron 5

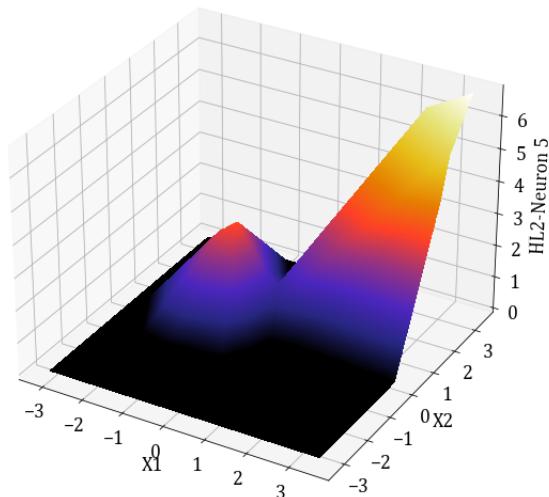
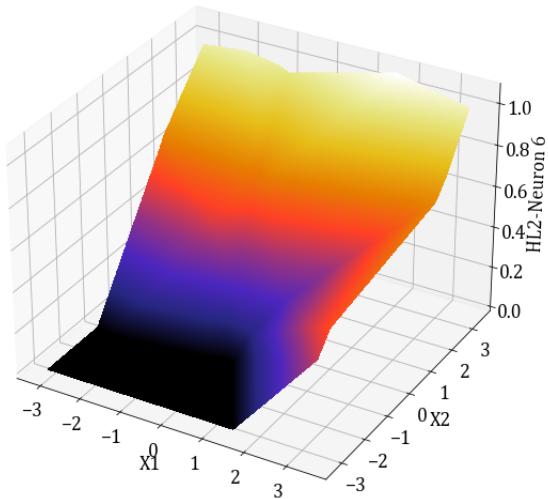


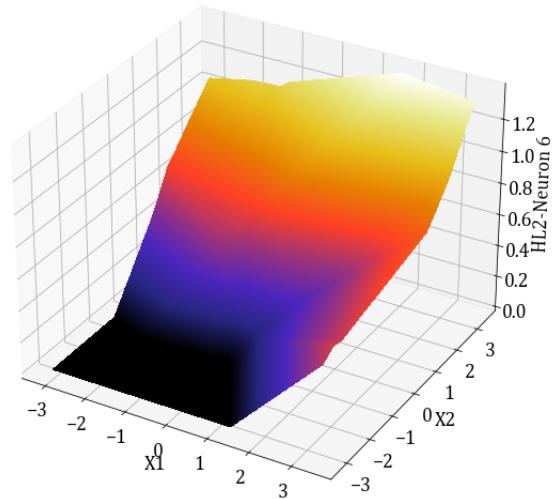
Figure 31: Surface Plots obtained for Hidden Layer 2, Neuron 5, across epochs.

2.1.4.14 Hidden Layer 2, Node 6

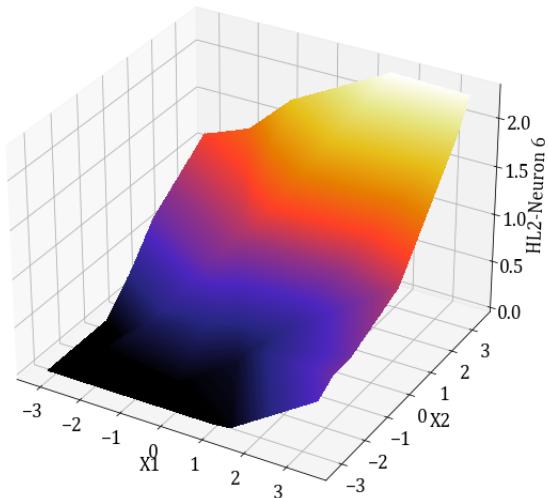
Epoch: 1; Surface for Layer 2, Neuron 6



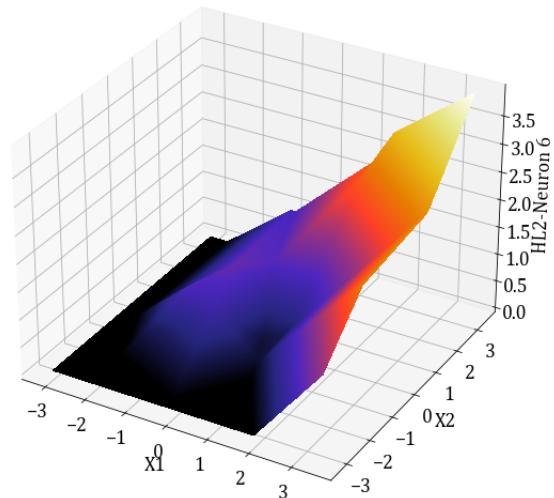
Epoch: 5; Surface for Layer 2, Neuron 6



Epoch: 20; Surface for Layer 2, Neuron 6



Epoch: 100; Surface for Layer 2, Neuron 6



Converged; Surface for Layer 2, Neuron 6

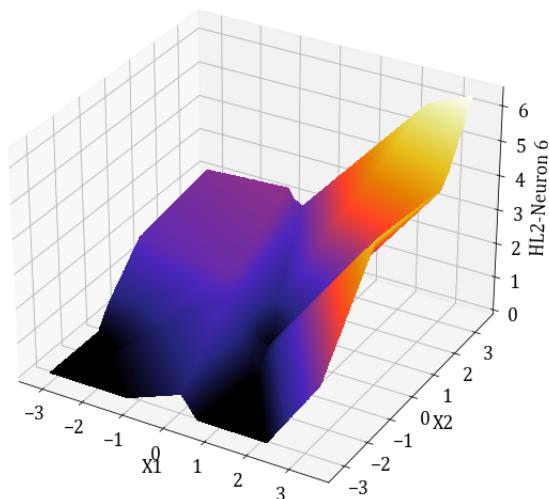
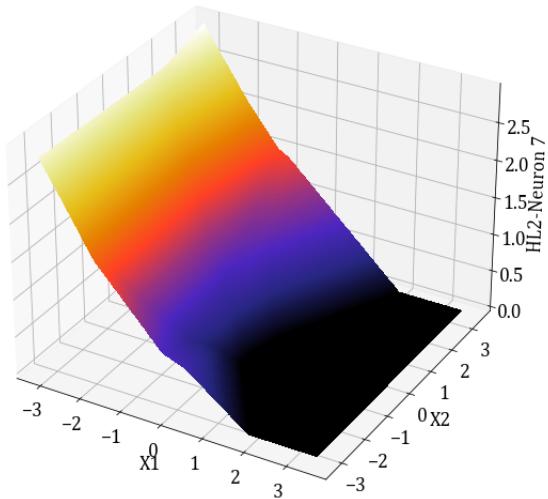


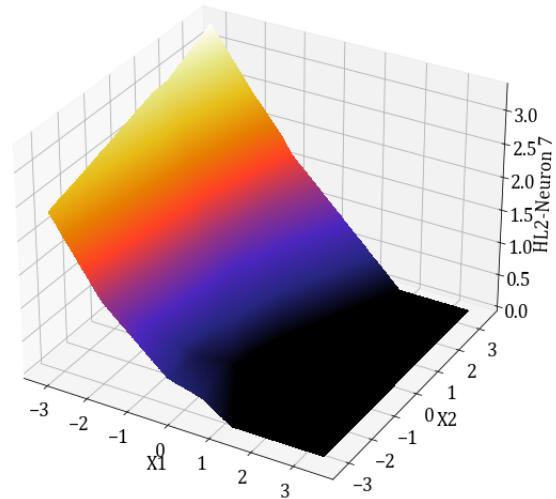
Figure 32: Surface Plots obtained for Hidden Layer 2, Neuron 6, across epochs.

2.1.4.15 Hidden Layer 2, Node 7

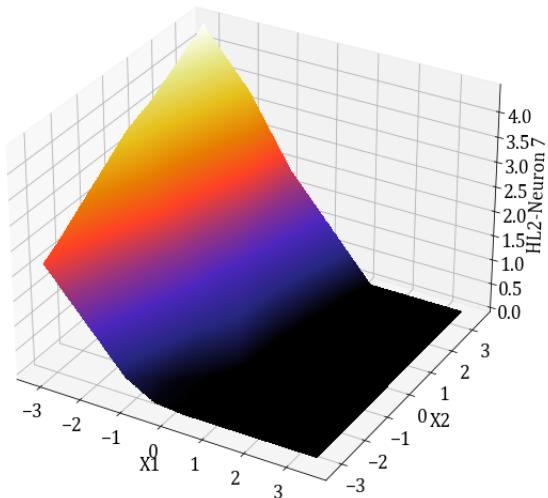
Epoch: 1; Surface for Layer 2, Neuron 7



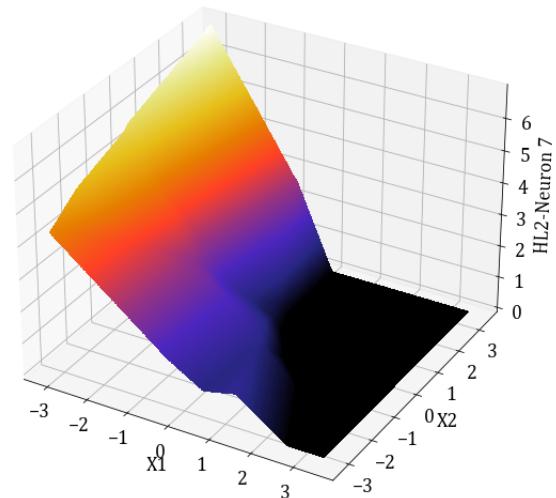
Epoch: 5; Surface for Layer 2, Neuron 7



Epoch: 20; Surface for Layer 2, Neuron 7



Epoch: 100; Surface for Layer 2, Neuron 7



Converged; Surface for Layer 2, Neuron 7

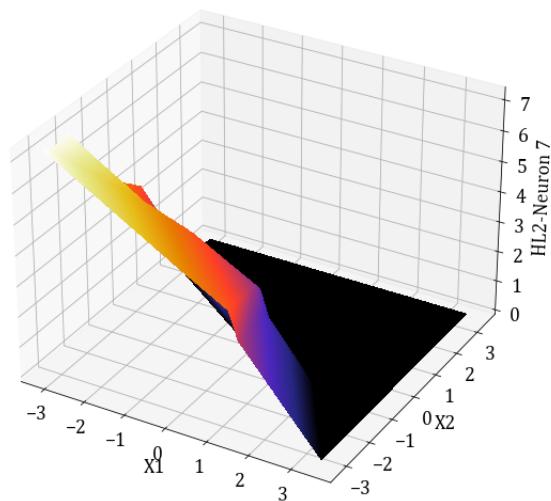
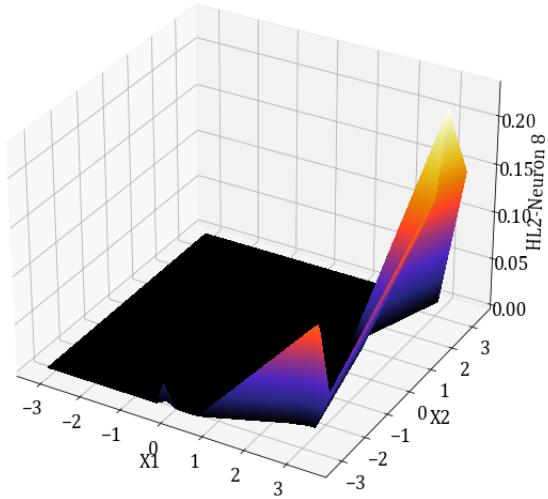


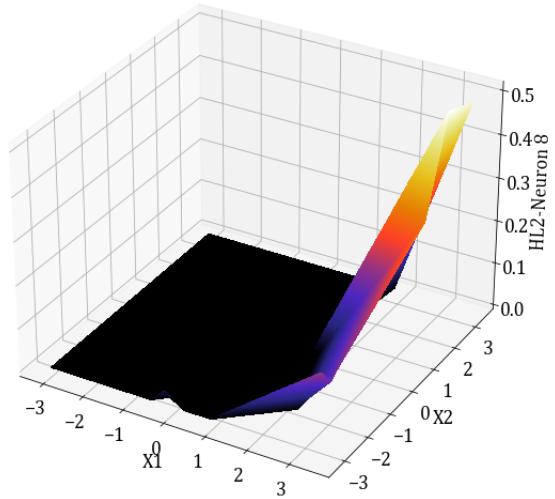
Figure 33: Surface Plots obtained for Hidden Layer 2, Neuron 7, across epochs.

2.1.4.16 Hidden Layer 2, Node 8

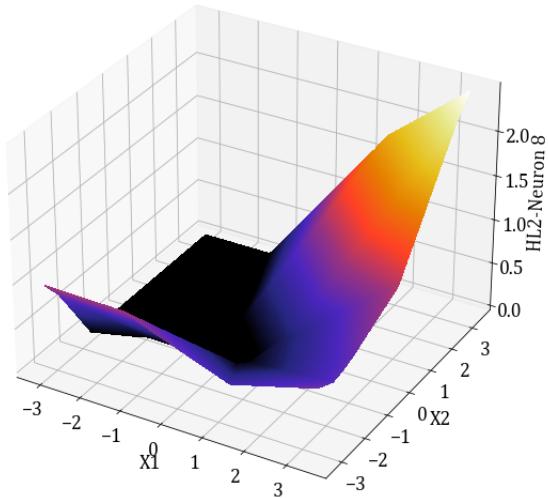
Epoch: 1; Surface for Layer 2, Neuron 8



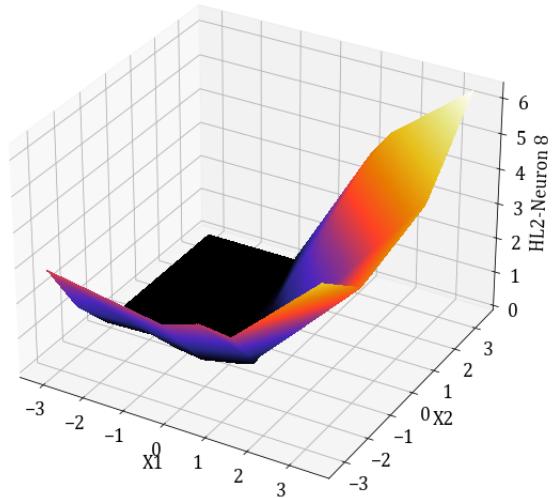
Epoch: 5; Surface for Layer 2, Neuron 8



Epoch: 20; Surface for Layer 2, Neuron 8



Epoch: 100; Surface for Layer 2, Neuron 8



Converged; Surface for Layer 2, Neuron 8

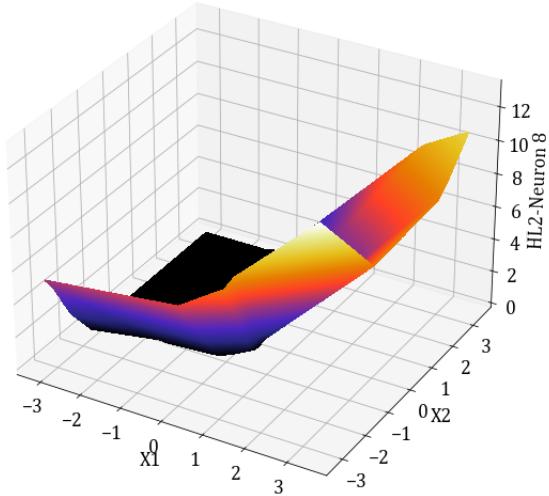
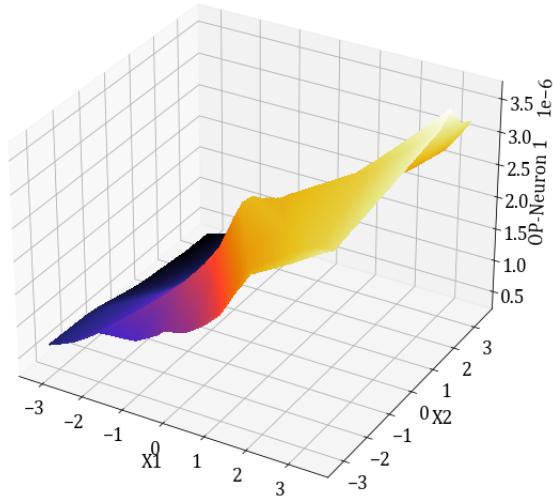


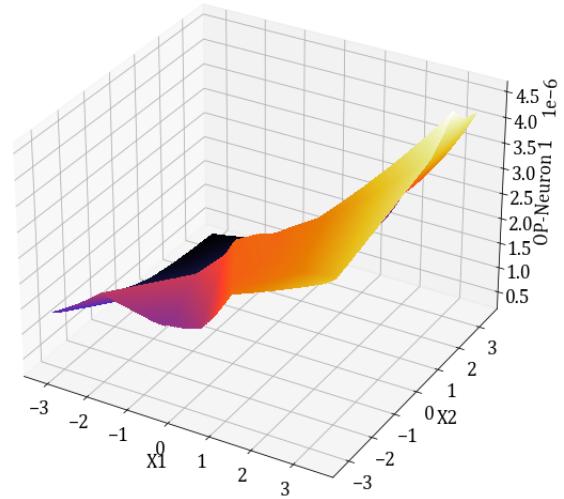
Figure 34: Surface Plots obtained for Hidden Layer 2, Neuron 8, across epochs.

2.1.4.17 Output Layer, Node 1

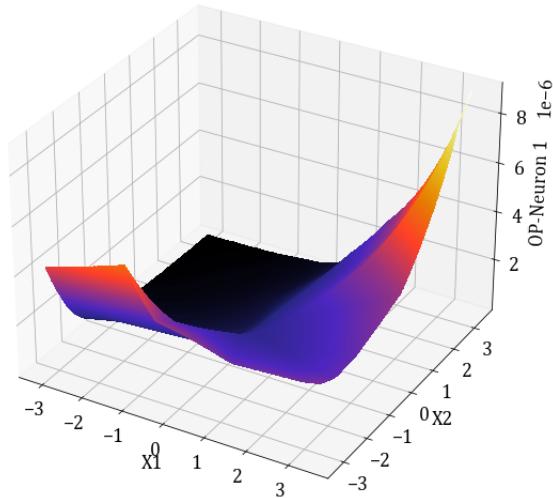
Epoch: 1; Surface for Output Layer, Neuron 1



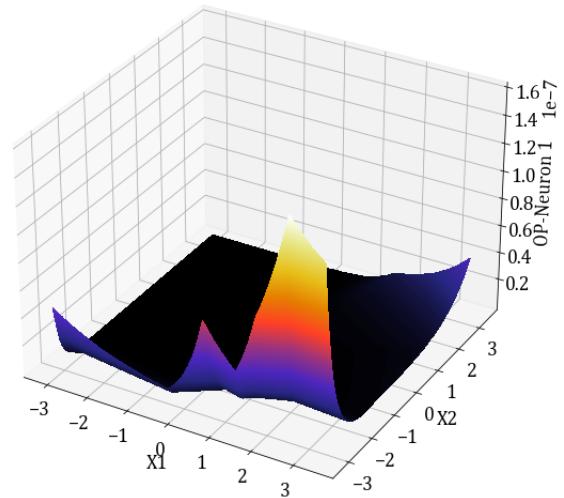
Epoch: 5; Surface for Output Layer, Neuron 1



Epoch: 20; Surface for Output Layer, Neuron 1



Epoch: 100; Surface for Output Layer, Neuron 1



Converged; Surface for Output Layer, Neuron 1

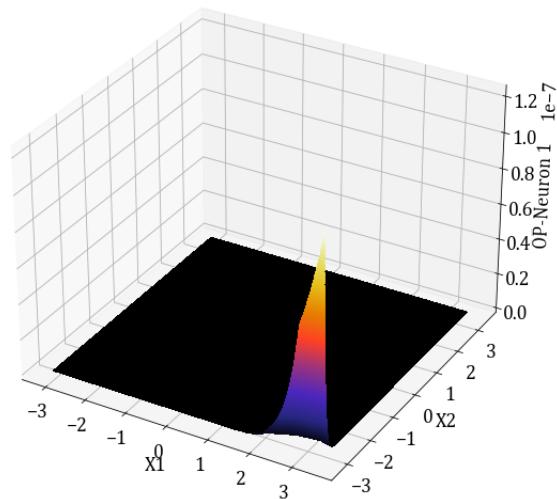
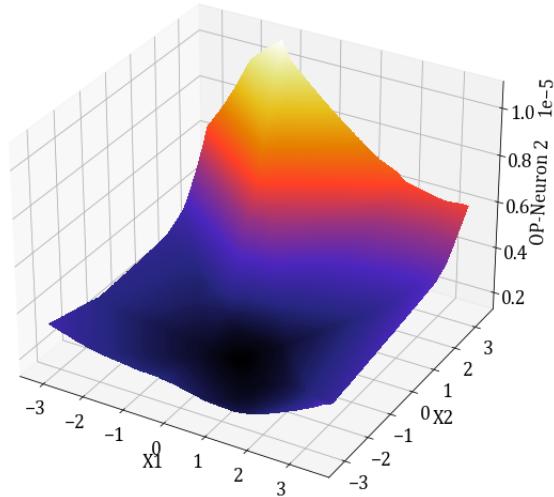


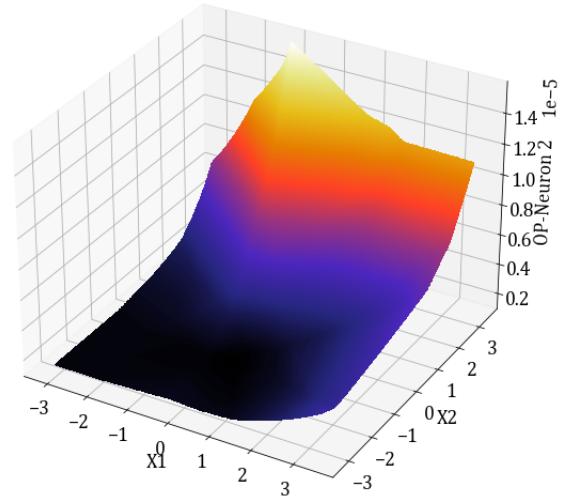
Figure 35: Surface Plots obtained for Output Layer, Neuron 1, across epochs.

2.1.4.18 Output Layer, Node 2

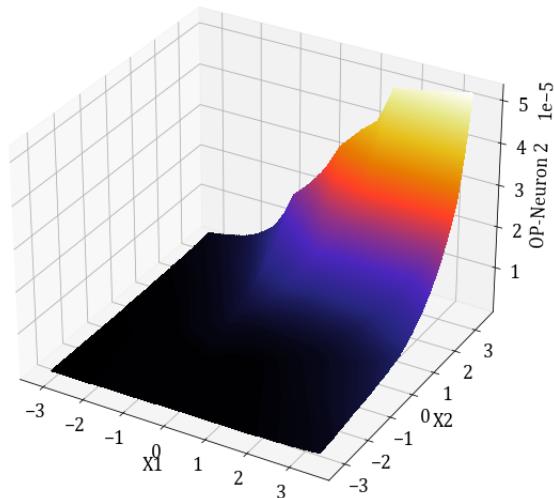
Epoch: 1; Surface for Output Layer, Neuron 2



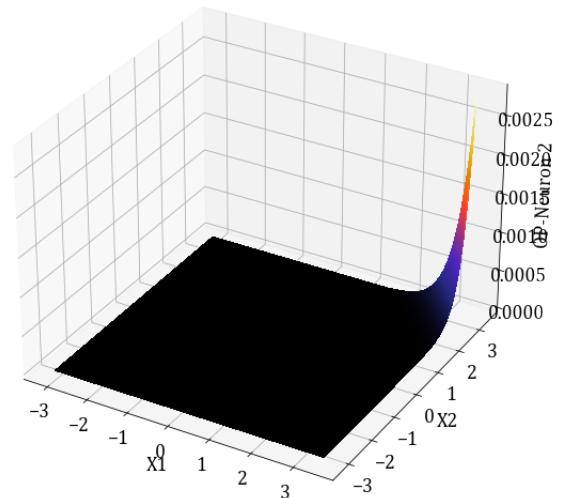
Epoch: 5; Surface for Output Layer, Neuron 2



Epoch: 20; Surface for Output Layer, Neuron 2



Epoch: 100; Surface for Output Layer, Neuron 2



Converged; Surface for Output Layer, Neuron 2

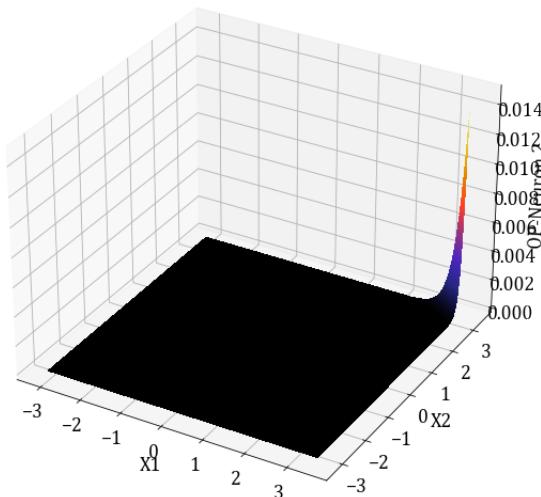
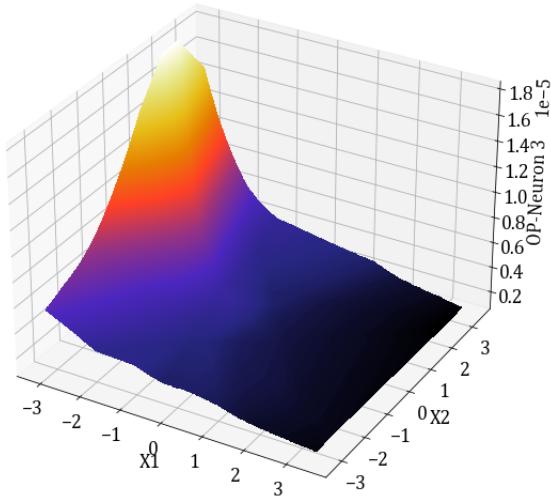


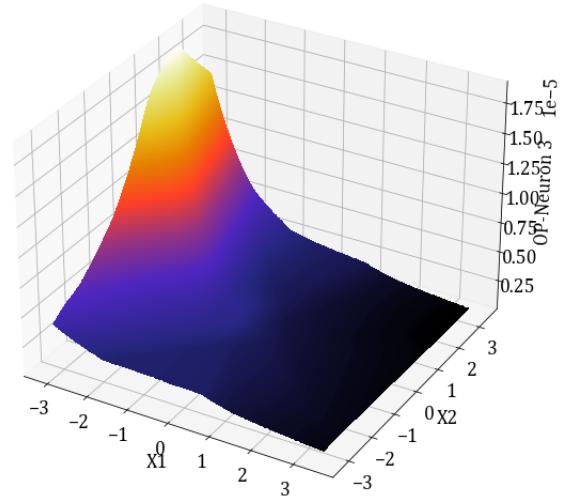
Figure 36: Surface Plots obtained for Output Layer, Neuron 2, across epochs.

2.1.4.19 Output Layer, Node 3

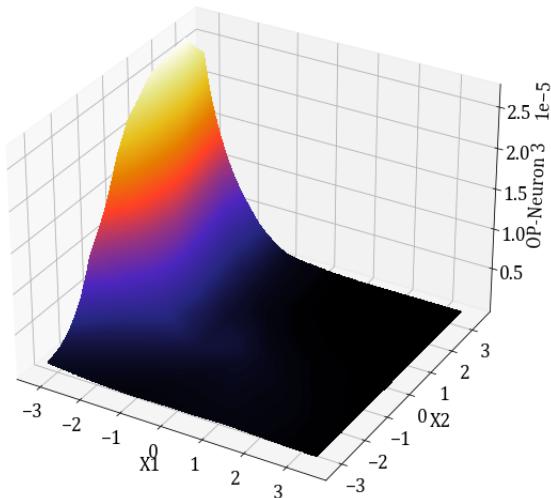
Epoch: 1; Surface for Output Layer, Neuron 3



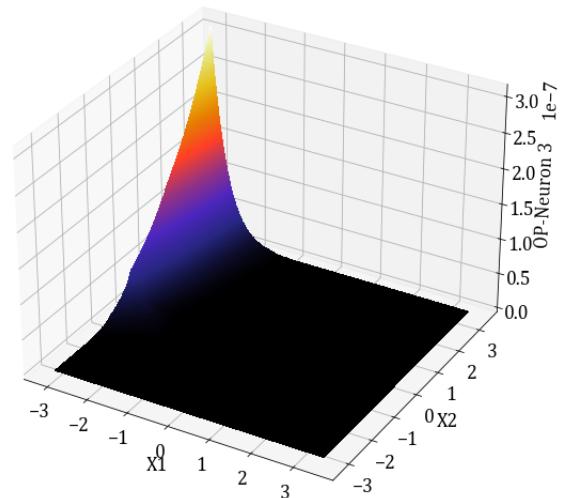
Epoch: 5; Surface for Output Layer, Neuron 3



Epoch: 20; Surface for Output Layer, Neuron 3



Epoch: 100; Surface for Output Layer, Neuron 3



Converged; Surface for Output Layer, Neuron 3

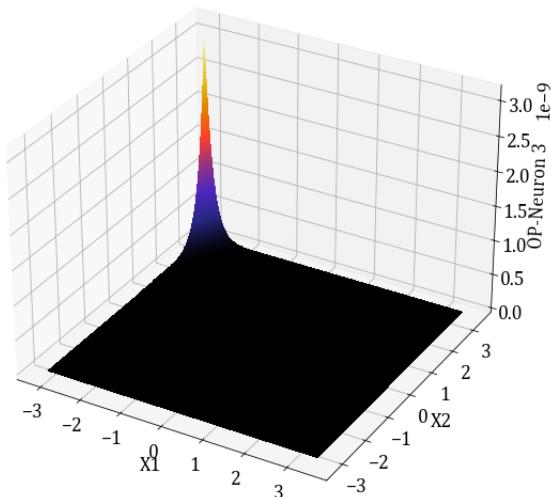


Figure 37: Surface Plots obtained for Output Layer, Neuron 3, across epochs.

From [Figure 19](#)-[Figure 37](#), we observe the following:

- First hidden layer surface plot is non-linear (activation function is [ReLU](#)). However, the surfaces obtained are hyperplanes.
- Responses from the second layer is highly non-linear and the surfaces are very complex.
- The surface plot of the output neurons shows the selection cum localization of different classes in the latent space.

2.2 Non-Linear SVM

3 Dataset 2A

This dataset contains data for five classes. The classes assigned for our team are: `coast`, `highway`, `mountain`, `opencountry` and `tallbuilding`. These classes were assigned the class label 0, 1, 2, 3 and 4 respectively. The classes are non-linearly separable and the dimension of the feature space is 24.

3.1 MLFFNN

An initial parameter search by varying the parameters of the MLFFNN alone was done. However, the accuracy of the model couldn't be improved. In order to increase the accuracy of the model, PCA was used to reduce the dimensionality of the model.

The hyperparameters varied and swepted for are - `PCA # components` `MLFFNN hidden layer size`, `MLFFNN batch size`, `MLFFNN learning rate` and `MLFFNN L2 regularization α` .

3.1.1 Classification Accuracies

The classification accuracies on the training and validation datasets (30% of the `dev.csv`) are as follows:

| # Components | # Neurons | Learning Rate | Batch Size | α | Accuracy | Validation Accuracy |
|--------------|-----------|---------------|------------|----------|----------|---------------------|
| 16 | (50, 50) | invscaling | auto | 0.001 | 79.02 | 52.23 |
| 16 | (50, 50) | adaptive | auto | 0.001 | 79.02 | 52.23 |
| 16 | (50, 50) | constant | auto | 0.001 | 79.02 | 52.23 |
| 20 | (50, 50) | constant | auto | 0.010 | 82.93 | 51.82 |
| 20 | (50, 50) | adaptive | auto | 0.010 | 82.93 | 51.82 |
| 20 | (50, 50) | invscaling | auto | 0.010 | 82.93 | 51.82 |
| 8 | (75, 75) | invscaling | 100 | 0.010 | 75.45 | 51.82 |
| 19 | (25, 25) | adaptive | auto | 0.001 | 68.05 | 51.42 |
| 19 | (25, 25) | constant | auto | 0.001 | 68.05 | 51.42 |
| 19 | (25, 25) | invscaling | auto | 0.001 | 68.05 | 51.42 |

Table 4: Best 10 Train and Validation Accuracies obtained after performing a `GridSearch` on 1728 parameter combinations.

3.1.2 Best Model

The parameter combination were additionally sorted based on minimum fitting time (least fitting time - first) and the model that gave the best accuracy the fastest (and potentially the most minimal model that best fits the data), was chosen. Hence the best parameter combination chosen is:

- `n_components`: 16
- `hidden_layer_sizes`: (50, 50)
- `batch_size`: 200
- `alpha`: 0.001
- `learning_rate`: invscaling

The classification accuracy of the best model on the testing data is: 49.06%.

The confusion matrices obtained are as follows:

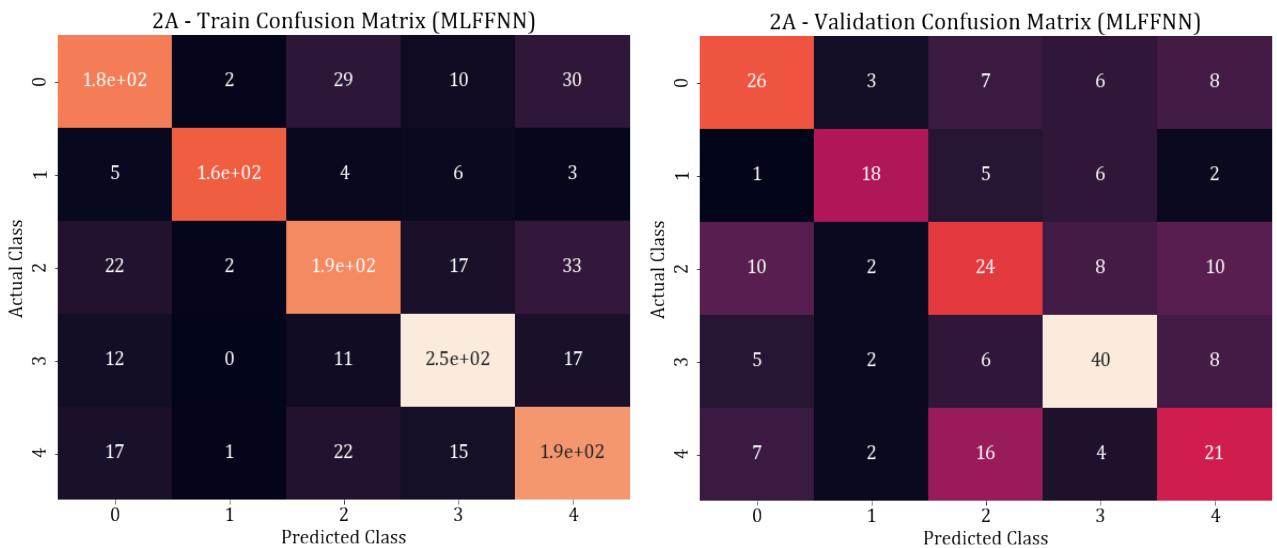


Figure 38: Training and Validation confusion matrices obtained for the best parameter combination, on the left and right respectively.

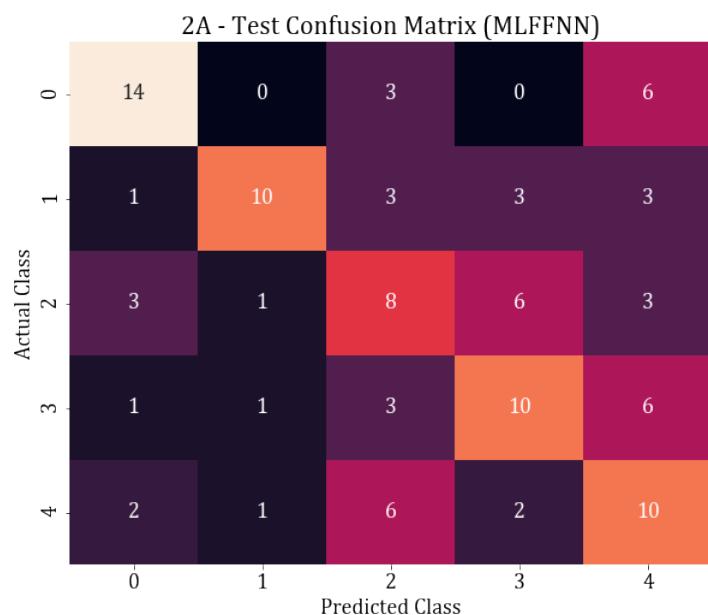


Figure 39: Testing confusion matrices obtained for the best parameter combination.

3.2 Gaussian-kernel SVM