

Name: Le Minh Bang – 黎明朋

ID: 313540015

HOMEWORK 1: DEEP LEARNING

I. Regression

1. Data Pre-processing

The dataset consists of **768 entries** (rows) and **10 features** (columns), which pertain to the energy efficiency of buildings. Each row corresponds to a different building configuration, and the columns contain measurements of structural attributes and the resulting energy loads (**Heating Load and Cooling Load**). The goal of analyzing this dataset is likely to understand the relationship between these structural features and the energy efficiency of buildings.

- Preprocessing Step Explained:

- o Load data
- o Shuffle data: The dataset is randomly shuffled to ensure that the order of data doesn't impact the training and testing processes.
- o One-Hot encoding: The categorical features **Orientation and Glazing Area Distribution** are transformed into multiple binary columns (one-hot encoding). This allows the machine learning model to treat these categorical variables as numerical data.
- o Scaling: Certain continuous features (Relative Compactness, Surface Area, etc.) are scaled to a range between 0 and 1, which ensures that all features have a similar range.
- o Splitting into Training and Testing Data: The dataset is split into training (75%) and testing (25%) sets, ensuring that the model is trained on a portion of the data and evaluated on unseen data to measure its performance.

2. Network Architecture

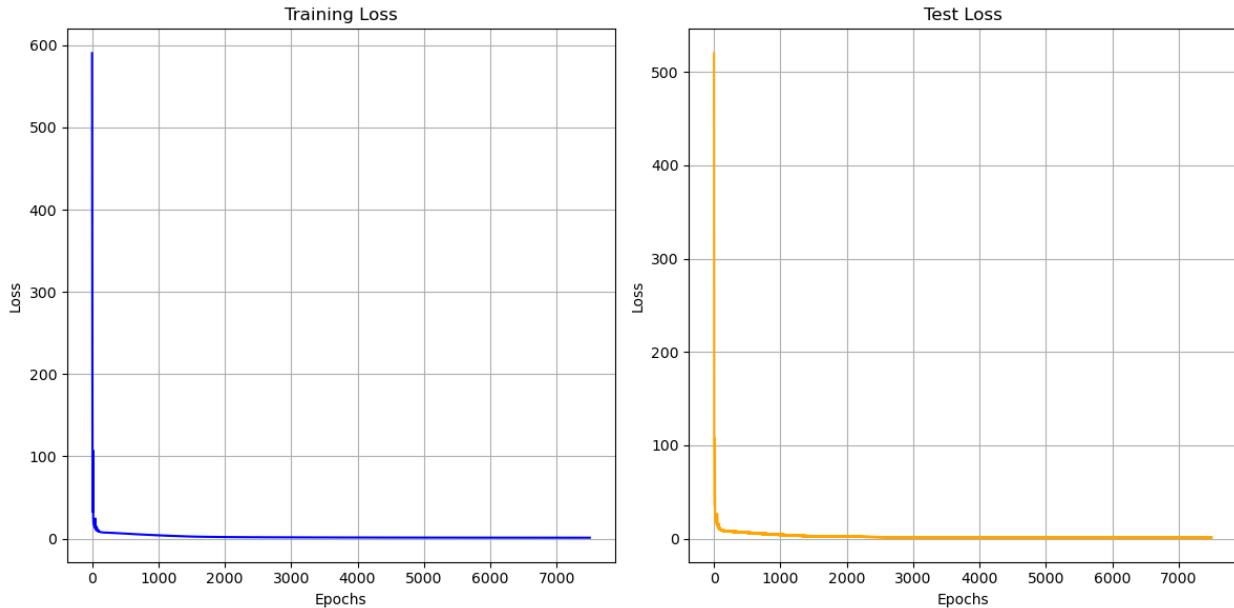
- The number of layers: 3
- Input layer: the same number of features 16 (One-Hot)
- Hidden layer: 8
- Output layer: 1 (Heating Load)
- Epoches: 7500
- Learning rate: 0.05

3. Learning Curve

- Learning Curve from both training and testing:

Name: Le Minh Bang – 黎明朋

ID: 313540015



In the early stages, both the training and test losses drop sharply, indicating that the model is learning quickly. This fast reduction in loss is typical when the model begins to capture the underlying patterns in the data. Both curves follow a similar trajectory, suggesting that the model is not overfitting, as there's no significant divergence between the training and test losses. Typically, overfitting is indicated when the training loss continues to decrease, but the test loss plateaus or increases. Here, both the training and test losses decrease together, suggesting that the model is generalizing well to unseen data. Additionally, if you were to plot both curves on the same graph, they would likely overlap or stay very close to one another due to their similarity, further reinforcing the lack of overfitting. This means that, up to this point, the model is performing well on both the training and test datasets without any signs of excessive variance or memorization of the training data.

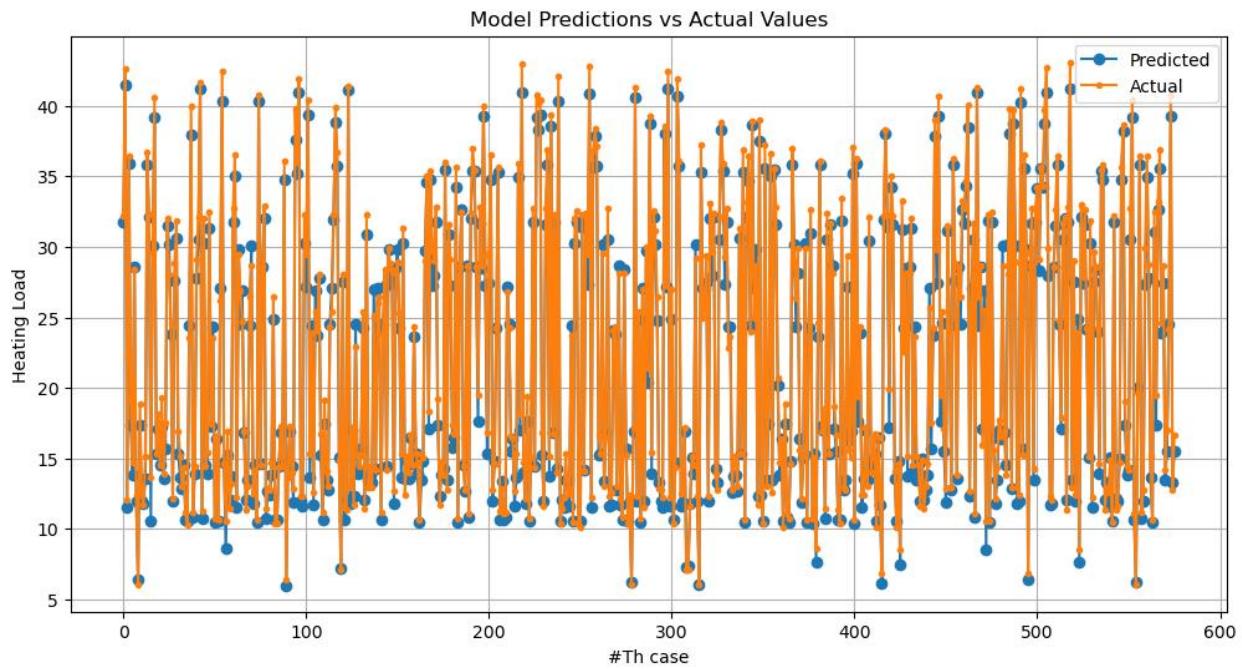
4. Error Rate & Comparison

- Training MSE and RMSE loss: **1.064215656850869** and **1.0316082865365463**
- Testing MSE and RMSE loss: **1.2806711043093506** and **1.131667400038258**

Name: Le Minh Bang – 黎明朋

ID: 313540015

5. Regression Result with Training Labels



- Training Results Comparision – First 10 rows

Training Results Comparison:

Predicted: 31.766032715692173 - Actual: 33.94

Predicted: 41.44409142398669 - Actual: 39.07

Predicted: 11.48848348661901 - Actual: 15.57

Predicted: 35.87410092443042 - Actual: 36.81

Predicted: 17.392350659907247 - Actual: 21.54

Predicted: 13.856916807600253 - Actual: 16.44

Predicted: 28.61149075192992 - Actual: 29.49

Predicted: 11.956564540640553 - Actual: 15.35

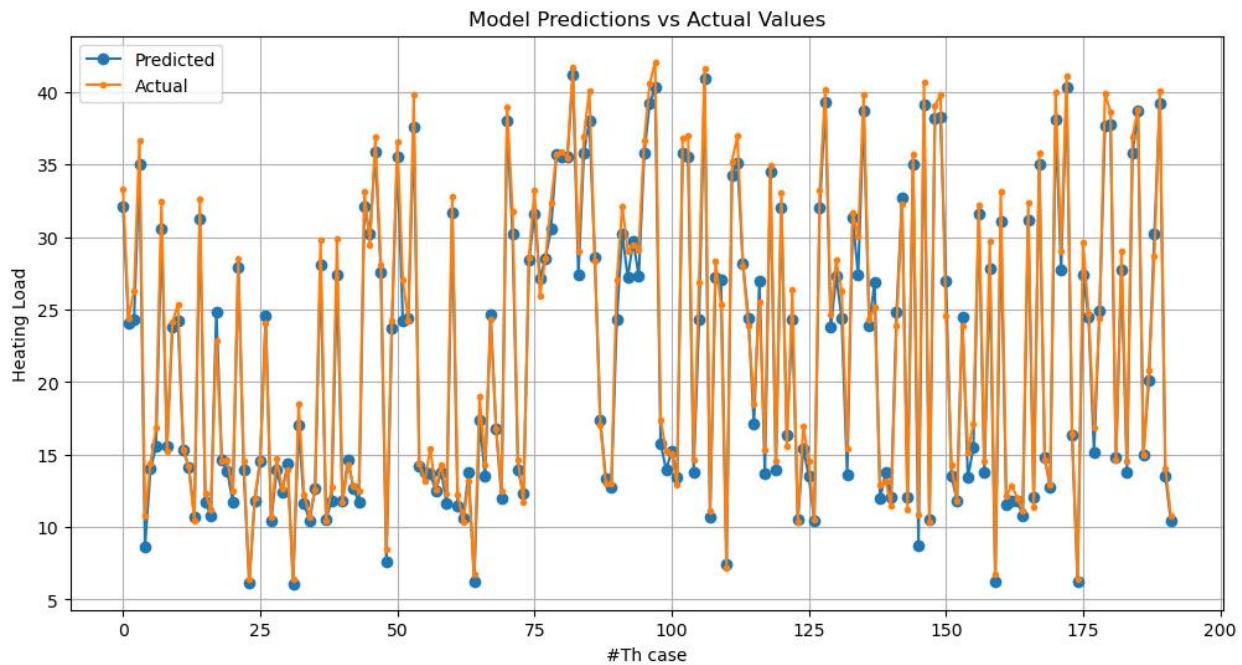
Predicted: 6.375833900979271 - Actual: 10.9

Predicted: 17.38892087533687 - Actual: 22.09

Name: Le Minh Bang – 黎明朋

ID: 313540015

6. Regression Result with Test Labels



- Testing Results Comparision – First 10 rows

Test Results Comparison:

Predicted: 32.093234619521596 - Actual: 34.11

Predicted: 24.020650539111074 - Actual: 26.02

Predicted: 24.292850780881047 - Actual: 31.06

Predicted: 35.00515386276186 - Actual: 34.29

Predicted: 8.654419241865615 - Actual: 16.75

Predicted: 14.000102974634025 - Actual: 17.2

Predicted: 15.54486836524579 - Actual: 20.56

Predicted: 30.570981369321352 - Actual: 35.48

Predicted: 15.614709808183507 - Actual: 18.14

Predicted: 23.823582895613853 - Actual: 26.13

7. Important features

Which input features influence the energy load significantly?

To deal with this question, I desire follow these steps:

1. Summing the Absolute Weights: Summing all weight values at the first layer weights (Why absolute? Because some value can be negative)
2. Selecting Important Features: Based on weight sums I can find which are important features in the dataset.

Name: Le Minh Bang – 黎明朋

ID: 313540015

3. Creating a DataFrame to Display Feature Importance: Just to show clearly the pair of features and weights.

```
weight_sums = np.sum(np.abs(W1), axis=1)

important_features = X_train.columns[weight_sums > 0]

df_important_features = pd.DataFrame({
    "Feature": X_train.columns,
    "Weight Sum": weight_sums
}).sort_values(by="Weight Sum", ascending=False)
df_important_features.reset_index(drop=True, inplace=True)
print("Important Features Order: \n", df_important_features)
```

Feature	Weight Sum
Wall Area	19.941050
Roof Area	13.714497
# Relative Compactness	8.739245
Glazing Area	5.841303
Overall Height	5.231094
GlazingAreaDist_0.0	1.737106
Surface Area	1.300620
GlazingAreaDist_5.0	0.237150
GlazingAreaDist_3.0	0.201560
Orientation_2.0	0.071692
Orientation_5.0	0.071308
GlazingAreaDist_1.0	0.068754
Orientation_3.0	0.066371
Orientation_4.0	0.050539
GlazingAreaDist_4.0	0.037924
GlazingAreaDist_2.0	0.024974

Name: Le Minh Bang – 黎明朋

ID: 313540015

II. Classification

1. Data Pre-processing

This dataset, **2024 Ionosphere Data**, consists of **35** columns, where the first **34** columns represent various radar return features, and the **35th** column is the target variable. The target is binary, with values 1 (represented as 'g' in the original data, indicating good radar returns) and 0 ('b' for bad returns). The data aims to classify radar returns as good or bad, based on signal attributes. The dataset is shuffled and split into **80%** for training and **20%** for testing, enabling the evaluation of machine learning models on radar return classification.

2. Network Architecture

The neural network design includes:

- **Input Layer:** Comprising 34 neurons representing the 34 attributes in each row of the provided dataset.
- **Hidden Layer:** Initially with 3 neurons, which will later be adjusted from 3 to 8 neurons to observe the training results.
- **Output Layer:** Comprising 1 neuron, designed for the classification task between 'g' (good) and 'b' (bad) or between 1 and 0.

Additional settings:

- Learning_rate = 0.05
- Epoches = 10001
- Step = 5000, and 50000 (to see more change when visualizing latent space)
- Lambda = 0.045 (L2)
- Batch size = 32 (for visualize latent space) and 128 (for training with batching)

3. Workflows

First, I built a neural network (**Model 1**) without applying any advanced techniques to check the baseline model's accuracy. Next, I applied L2 regularization to minimize the overfitting during training (**Model 2**). Third, I implemented batching to improve the model's efficiency. This model was used for further experimentation (**Model 3**). Finally, I explored the changes in the model when adjusting the number of neurons in the hidden layer to track and compare the results.

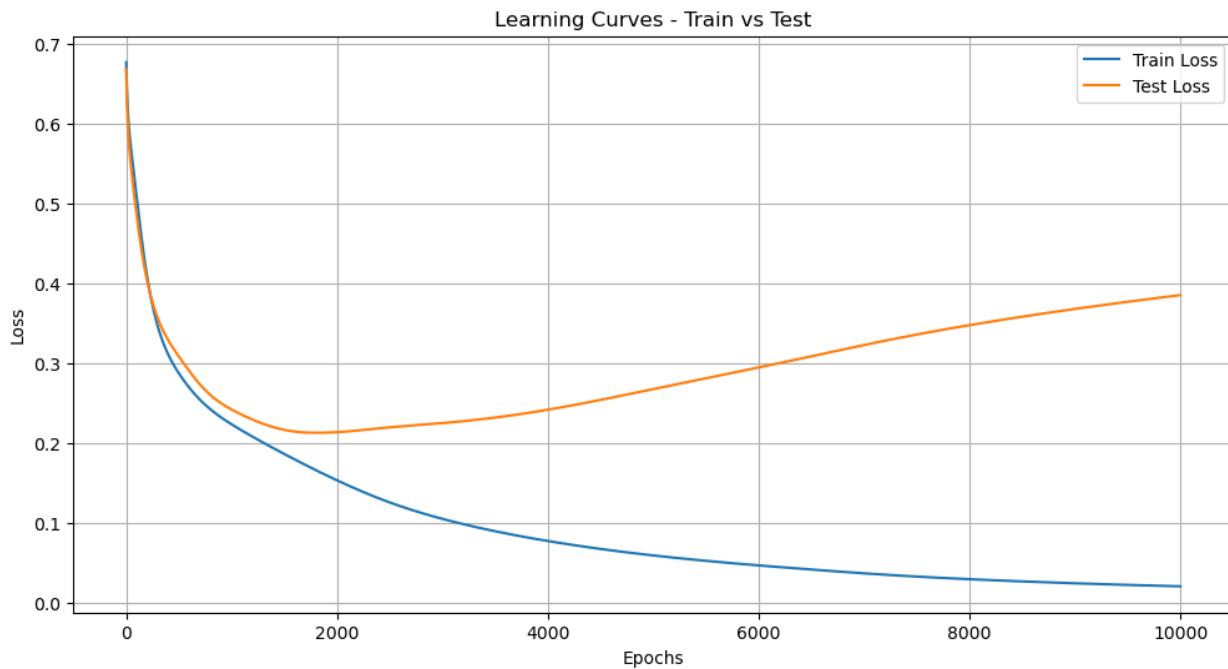
4. Learning Curve & Comparision

- **Model 1:** The model shows a clear sign of overfitting. While the training loss steadily decreases, the test loss begins to rise after an initial decline, indicating that the model fits the training data well but struggles to generalize to unseen test data.

Name: Le Minh Bang – 黎明朋

ID: 313540015

Epoch 0/10001 - Loss of Train: 0.6763807048903826 & Test: 0.6678373837579957
Epoch 1000/10001 - Loss of Train: 0.22340289423748064 & Test: 0.2419381053214526
Epoch 2000/10001 - Loss of Train: 0.15334563323433323 & Test: 0.21365317546601556
Epoch 3000/10001 - Loss of Train: 0.10515531158770293 & Test: 0.22498072345209152
Epoch 4000/10001 - Loss of Train: 0.07745725622820758 & Test: 0.24172837934966582
Epoch 5000/10001 - Loss of Train: 0.05941463537643273 & Test: 0.26752661021408386
Epoch 6000/10001 - Loss of Train: 0.04694014030672532 & Test: 0.2946345662046159
Epoch 7000/10001 - Loss of Train: 0.037006789278224615 & Test: 0.3226725115033572
Epoch 8000/10001 - Loss of Train: 0.029677430552104513 & Test: 0.34752056264409875
Epoch 9000/10001 - Loss of Train: 0.024571973092368294 & Test: 0.36780515434146155
Epoch 10000/10001 - Loss of Train: 0.020788221167689285 & Test: 0.385126518897679

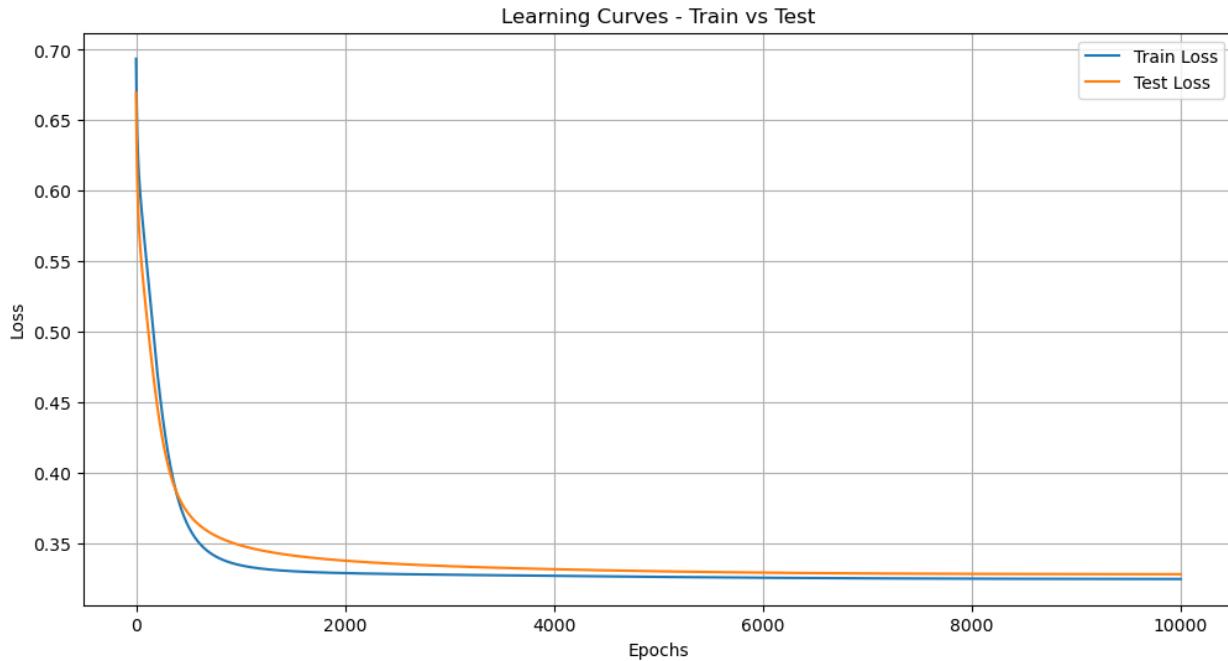


Name: Le Minh Bang – 黎明朋

ID: 313540015

- **Model 2:** This model demonstrates better generalization with both the training and test losses converging closely over time. The losses remain steady and parallel, suggesting that the model has minimized overfitting and can generalize well to new, unseen data, making it more reliable for predictions.

Epoch 0/10001 - Loss of Train: 0.6936325007559299 & Test: 0.6694725068849623
Epoch 1000/10001 - Loss of Train: 0.334212784487698 & Test: 0.34834572088896637
Epoch 2000/10001 - Loss of Train: 0.3286327408328779 & Test: 0.33746255650597345
Epoch 3000/10001 - Loss of Train: 0.327481796721 & Test: 0.33355384308045405
Epoch 4000/10001 - Loss of Train: 0.32670020948114686 & Test: 0.33139640082291427
Epoch 5000/10001 - Loss of Train: 0.3259671764924408 & Test: 0.32994563144167877
Epoch 6000/10001 - Loss of Train: 0.32535643250770657 & Test: 0.3289738105403448
Epoch 7000/10001 - Loss of Train: 0.3249384012423062 & Test: 0.32839828778850216
Epoch 8000/10001 - Loss of Train: 0.32468562117766187 & Test: 0.32809318386980074
Epoch 9000/10001 - Loss of Train: 0.3245424813639947 & Test: 0.3279444731511425
Epoch 10000/10001 - Loss of Train: 0.32446469181012544 & Test: 0.3278784283542565



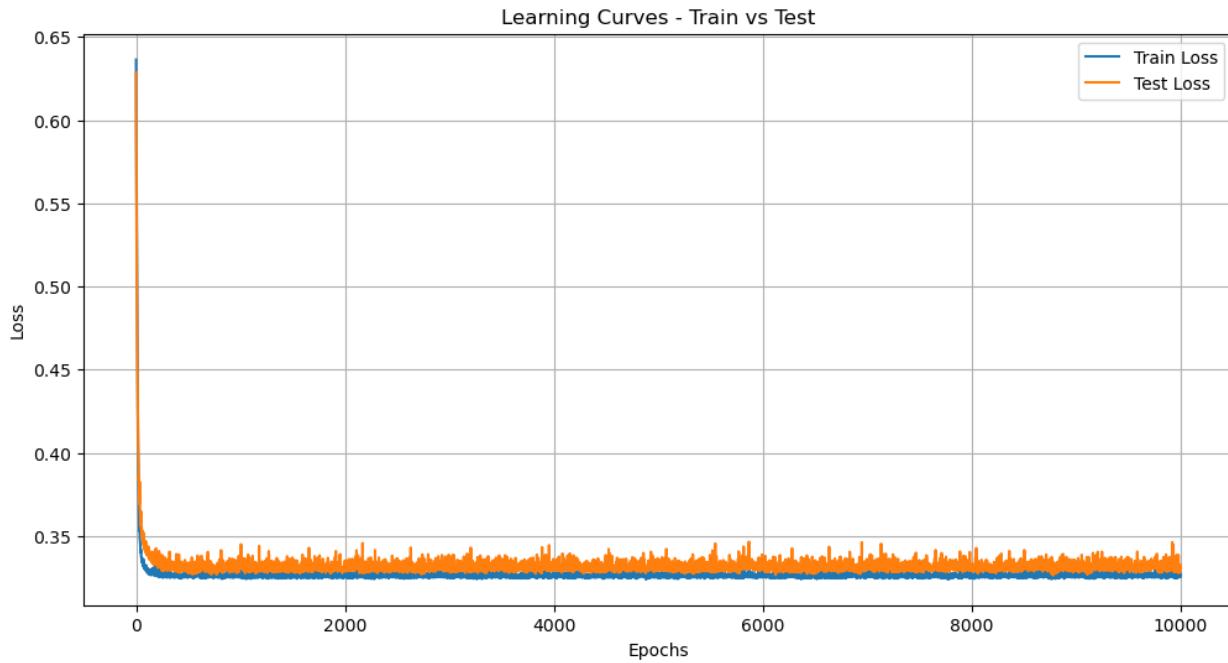
- **Model 3:** The model trained using batching shows a more fluctuating pattern in both the training and test losses compared to models without batching. This is expected, as batching introduces variance in gradient updates due to smaller subsets of data being used in each update. Despite the fluctuations, the training and test losses remain relatively close, with no significant divergence, indicating that the model generalizes well. The final losses are stable and low, reflecting

Name: Le Minh Bang – 黎明朋

ID: 313540015

that batching has not led to overfitting and the model maintains a strong generalization ability.

Epoch 0/10001 - Loss of Train: 0.6629991345232179 & Test: 0.6498625086234295
Epoch 1000/10001 - Loss of Train: 0.3235702392069424 & Test: 0.32863714617567563
Epoch 2000/10001 - Loss of Train: 0.3229029712870683 & Test: 0.32635484981526774
Epoch 3000/10001 - Loss of Train: 0.3228062497497416 & Test: 0.32610154620235093
Epoch 4000/10001 - Loss of Train: 0.32583545801863417 & Test: 0.3277303754662038
Epoch 5000/10001 - Loss of Train: 0.32486600125217474 & Test: 0.3275040500242268
Epoch 6000/10001 - Loss of Train: 0.32506769388103324 & Test: 0.33122952040288434
Epoch 7000/10001 - Loss of Train: 0.3248472662200649 & Test: 0.3264850283336299
Epoch 8000/10001 - Loss of Train: 0.32573550712451466 & Test: 0.3304503124612579
Epoch 9000/10001 - Loss of Train: 0.3260785902289491 & Test: 0.32557297930250456
Epoch 10000/10001 - Loss of Train: 0.32449204858972086 & Test: 0.33037823741511424

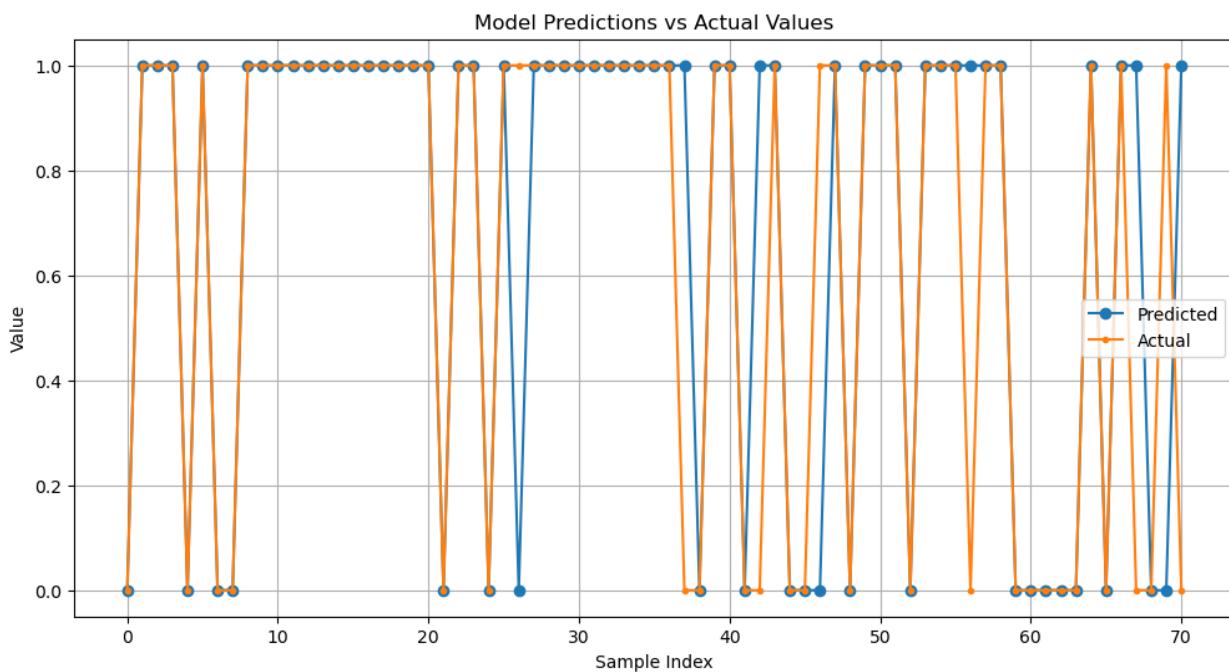
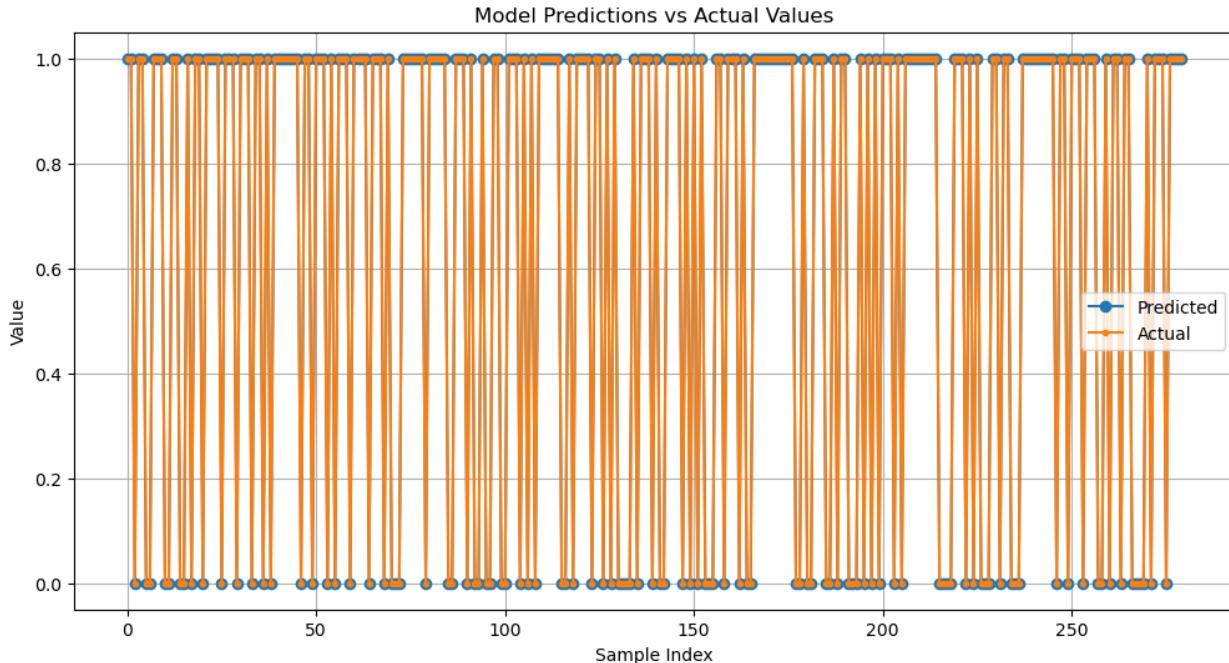


5. Error Rate & Comparision

- Model 1:
 - Training Binary Cross Entropy Loss: **0.020784937090155928**, Testing Binary Cross Entropy Loss: **0.385126518897679**
 - Comparision: Training and Testing

Name: Le Minh Bang – 黎明朋

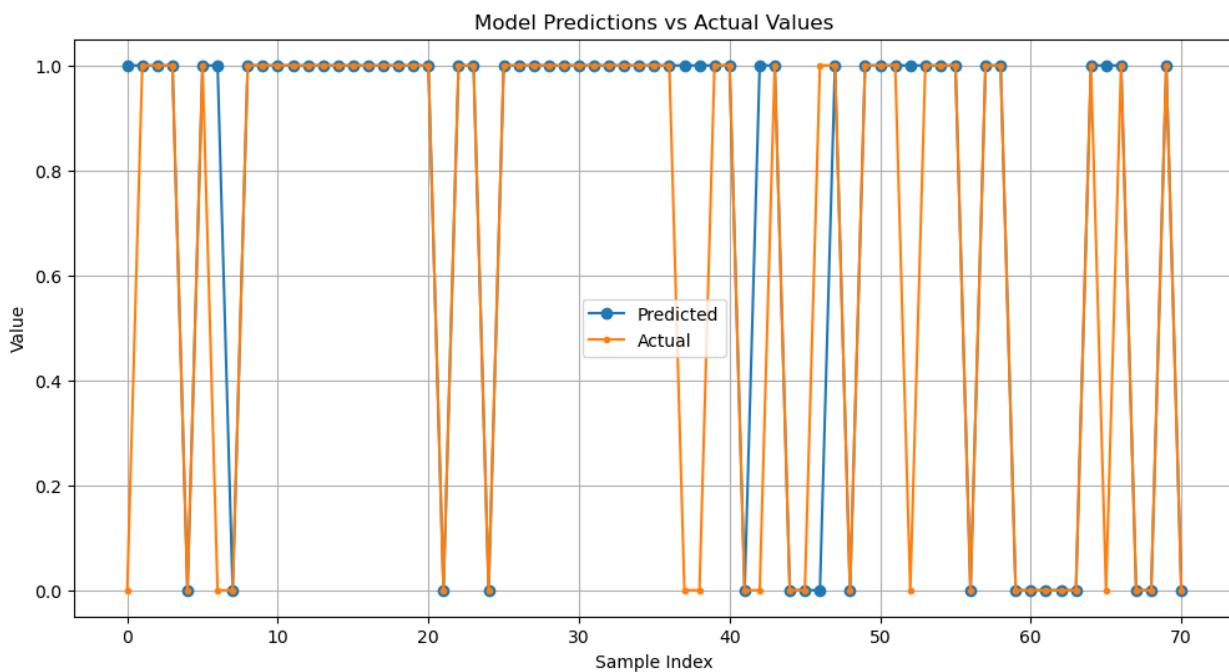
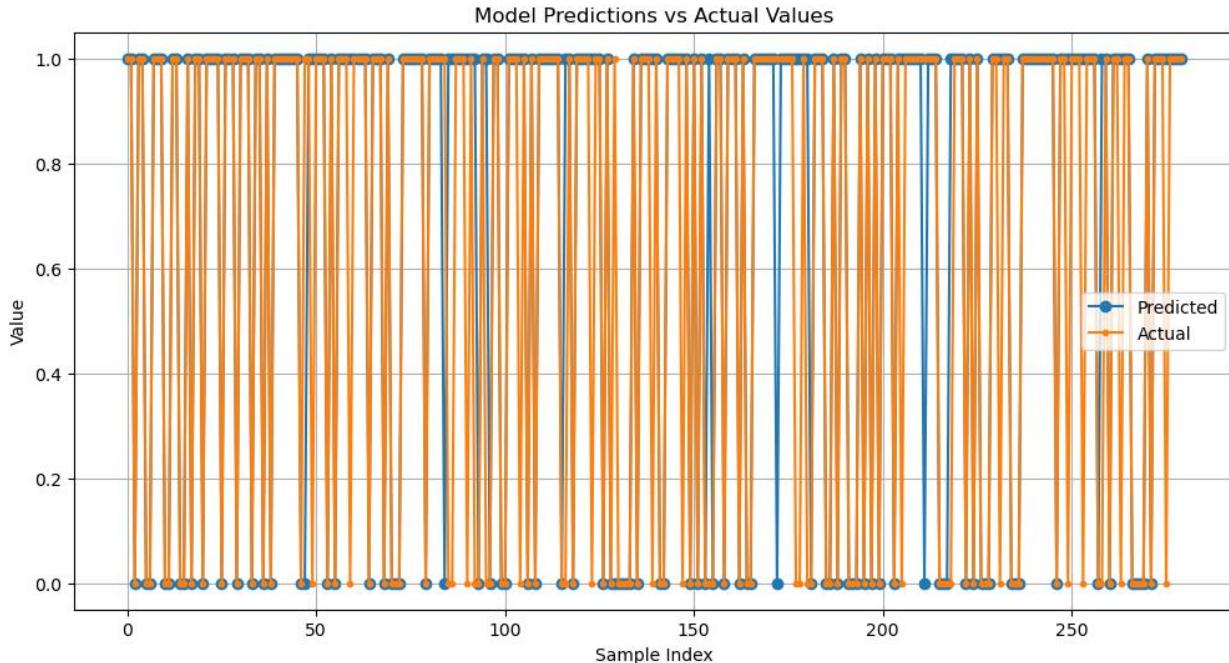
ID: 313540015



- Model 2:
 - Training Binary Cross Entropy Loss: **0.3244646361259648**, Testing Binary Cross Entropy Loss: **0.3278784283542565**
 - Comparison: Training and Testing

Name: Le Minh Bang – 黎明朋

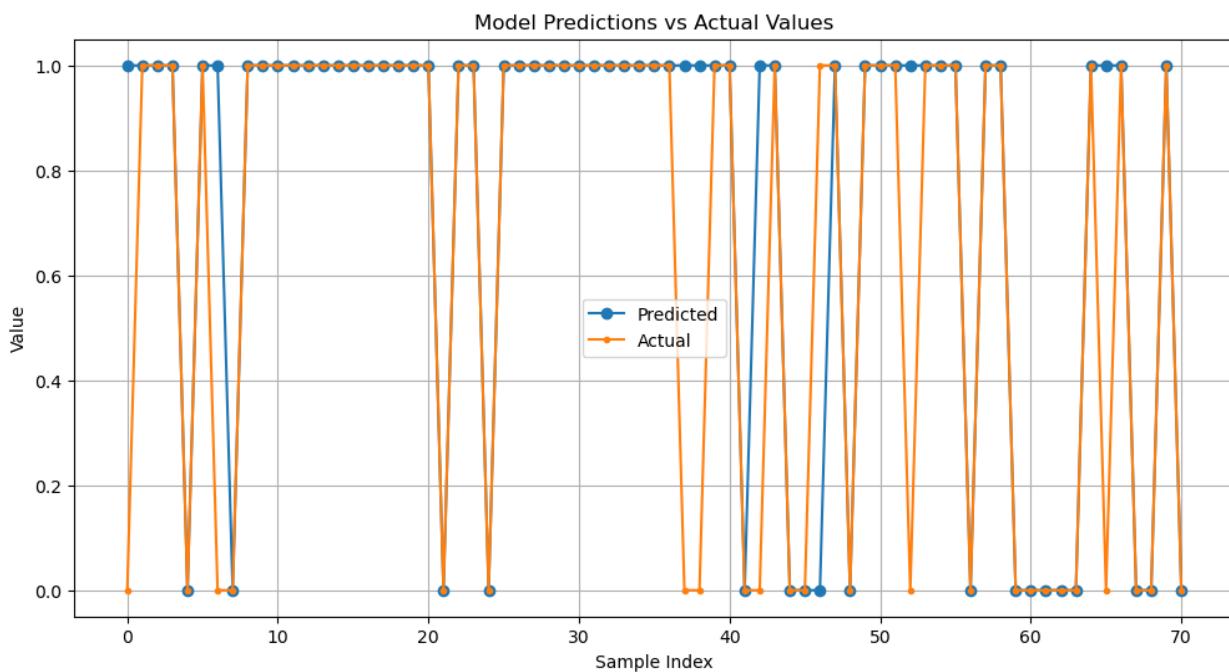
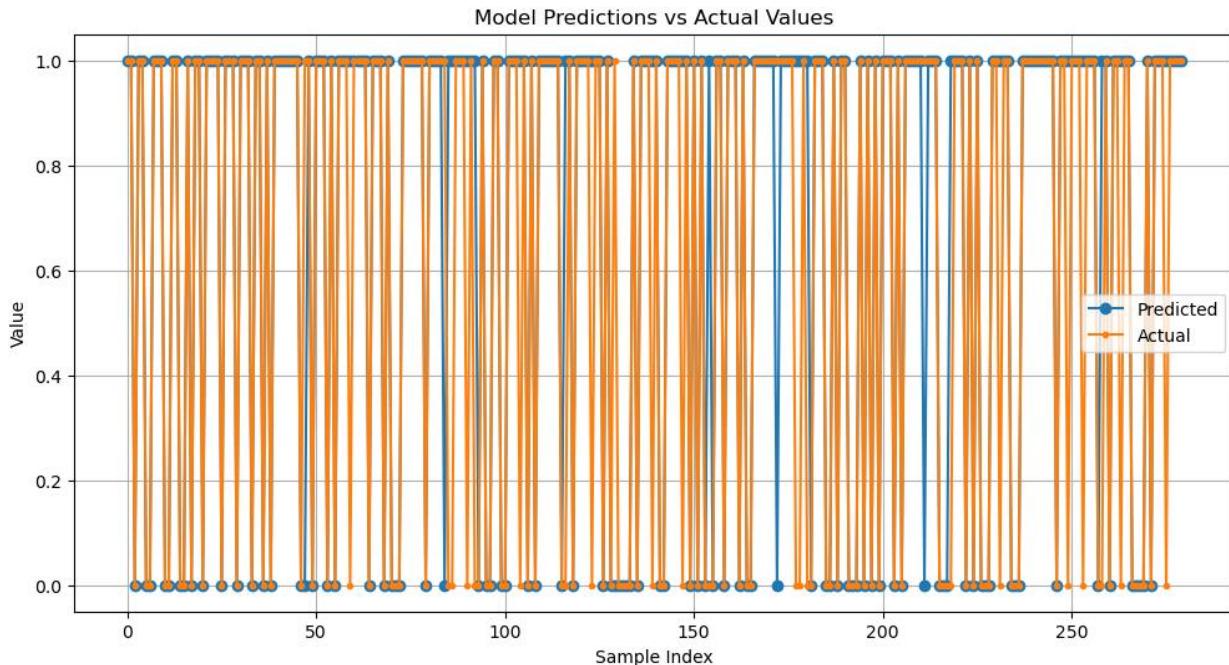
ID: 313540015



- Model 3:
 - Training Binary Cross Entropy Loss: **0.32583654485079716**, Testing Binary Cross Entropy Loss: **0.3299724590556967**
 - Comparison: Training and Testing

Name: Le Minh Bang – 黎明朋

ID: 313540015



Model 1 (No Regularization) shows clear signs of overfitting, with a training loss of 0.0207 and a significantly higher test loss of 0.3851. This indicates that the model performs well on the training data but fails to generalize to unseen data. **Model 2 (L2 Regularization)** demonstrates much better generalization, with both the training loss (0.3244) and the test loss (0.3278) being close. This suggests that the L2 regularization effectively mitigates overfitting, resulting in a more stable model. **Model 3 (Batching)** maintains similarly low losses but introduces fluctuations due to batch gradient descent.

Name: Le Minh Bang – 黎明朋

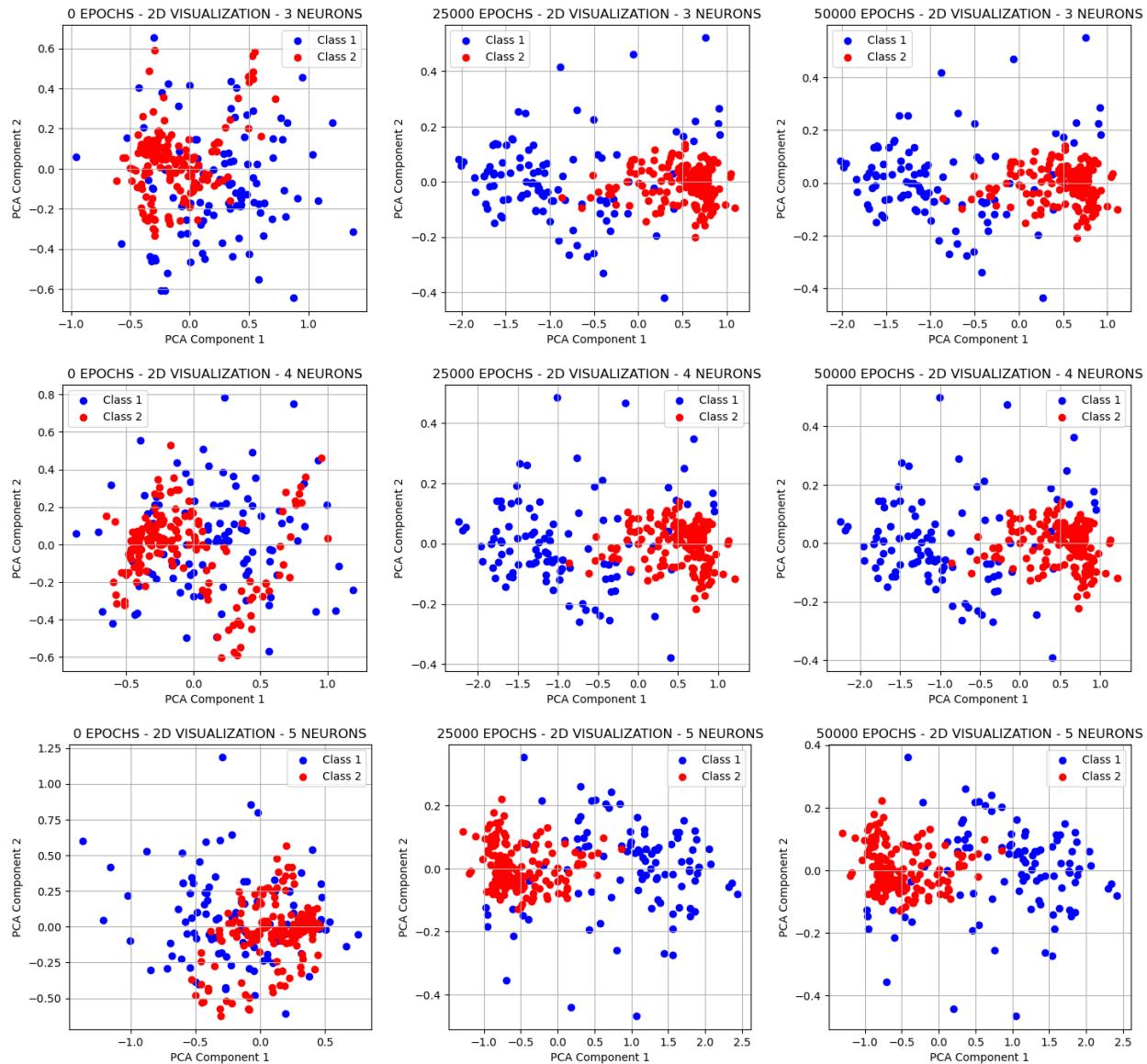
ID: 313540015

Despite this, the model's final train loss (0.3258) and test loss (0.3303) indicate that batching helps maintain generalization without significant overfitting.

In conclusion, **Model 2** shows the most stable performance with minimal overfitting, while **Model 3** exhibits similar performance with some fluctuation due to batching.

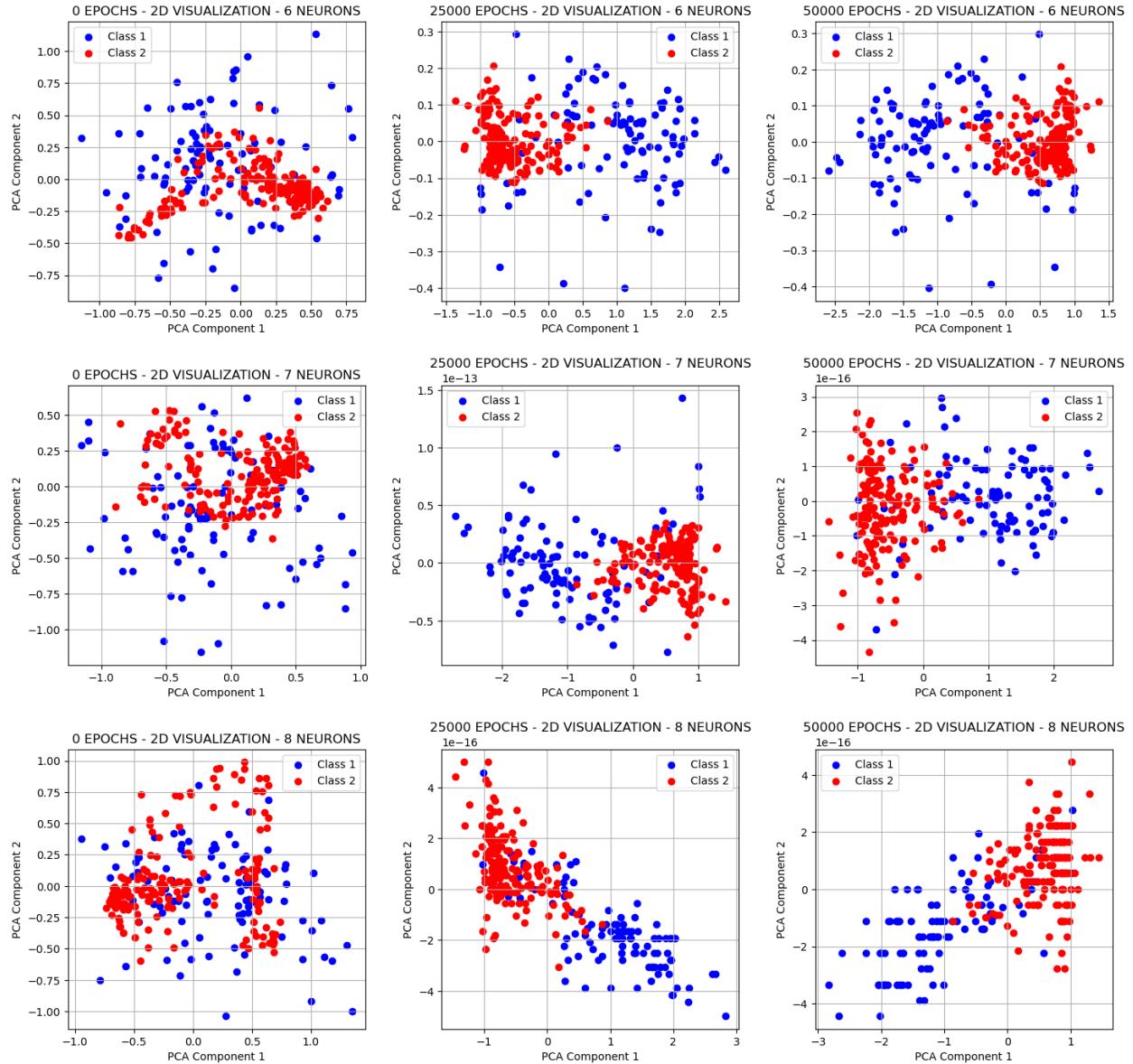
6. Latent Features

In this section, we successively trained the model with 3 to 8 neurons in the hidden layer to observe changes in the model 2-dimensional space. The results are presented as follows:



Name: Le Minh Bang – 黎明朋

ID: 313540015



As observed, the model efficiently separates class 1 and class 2 data points under different neuron configurations. Initially, when the epoch count is 0, the data points are not well-separated. However, as the training progresses, especially in the middle of the training period, the separation becomes more distinct across all neuron configurations. Notably, the model performs best with 8 neurons during this stage. By the final epoch (50,000), there isn't much difference between the configurations, but the 8-neuron case clearly shows the data points more discretely separated compared to earlier stages.

Thank you 😊