



University  
of Cyprus

**MAI612 - MACHINE LEARNING  
Assignment 3 – Neural Networks**

**Chrysis Andreou  
(UC1366020)**

## B1

### Part B1: Training and Testing the Neural Network

Epoch 200/1400, Training Loss: 0.07979872601048439, Test Loss:

0.08942652216914486

Epoch 400/1400, Training Loss: 0.06477387889186248, Test Loss:

0.0764771819566046

Epoch 600/1400, Training Loss: 0.05756420394151484, Test Loss:

0.07308519393822113

Epoch 800/1400, Training Loss: 0.05344158576453639, Test Loss:

0.06846196254830014

Epoch 1000/1400, Training Loss: 0.05202881655907902, Test Loss:

0.06567367512156046

Epoch 1200/1400, Training Loss: 0.050672801479359486, Test Loss:

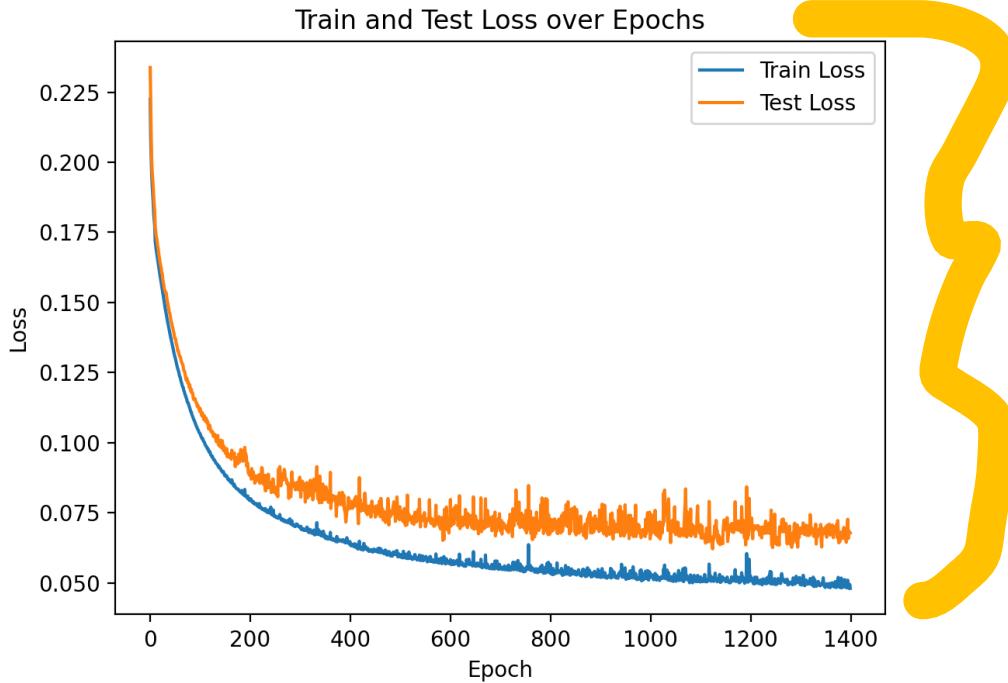
0.06928753241683387

Epoch 1400/1400, Training Loss: 0.04804998388632924, Test Loss:

0.06780105813050315

### Part B1: Analysis and Explanation:

1. Training loss is decreasing consistently, but test loss shows significant fluctuations after early epochs.
2. The model might be overfitting, as indicated by the unstable test loss.
3. Use Early Stopping to prevent overfitting by halting training when the test loss stops improving.
4. Apply regularization (e.g., L2 regularization or dropout) to enhance generalization.
5. Experimenting with different learning rates or increasing the momentum might also help stabilize the training process.
6. Use a learning rate scheduler to reduce the learning rate gradually, helping stabilize the training process.



## B2.1

### Part B2.1: Learning Rate Experiments

Learning Rate 0.0001: Final Training Loss: 0.14279708800525165, Final Test Loss: 0.153399027112199

Learning Rate 0.0001: Train Accuracy: 0.90625, Test Accuracy: 0.9122807017543859

Learning Rate 0.001: Final Training Loss: 0.12743397858488337, Final Test Loss: 0.14082828407433678

Learning Rate 0.001: Train Accuracy: 0.884765625, Test Accuracy: 0.8771929824561403

Learning Rate 0.01: Final Training Loss: 0.23339252030540766, Final Test Loss: 0.23731736404797799

Learning Rate 0.01: Train Accuracy: 0.62890625, Test Accuracy: 0.6140350877192983

Learning Rate 0.1: Final Training Loss: 0.25123371020406915, Final Test Loss: 0.25904143249888134

Learning Rate 0.1: Train Accuracy: 0.62890625, Test Accuracy: 0.6140350877192983

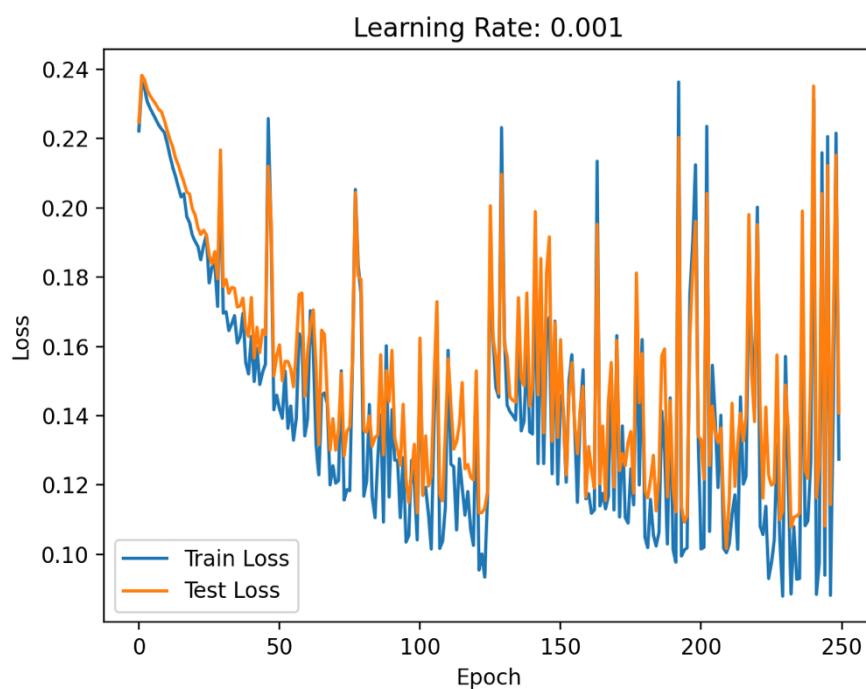
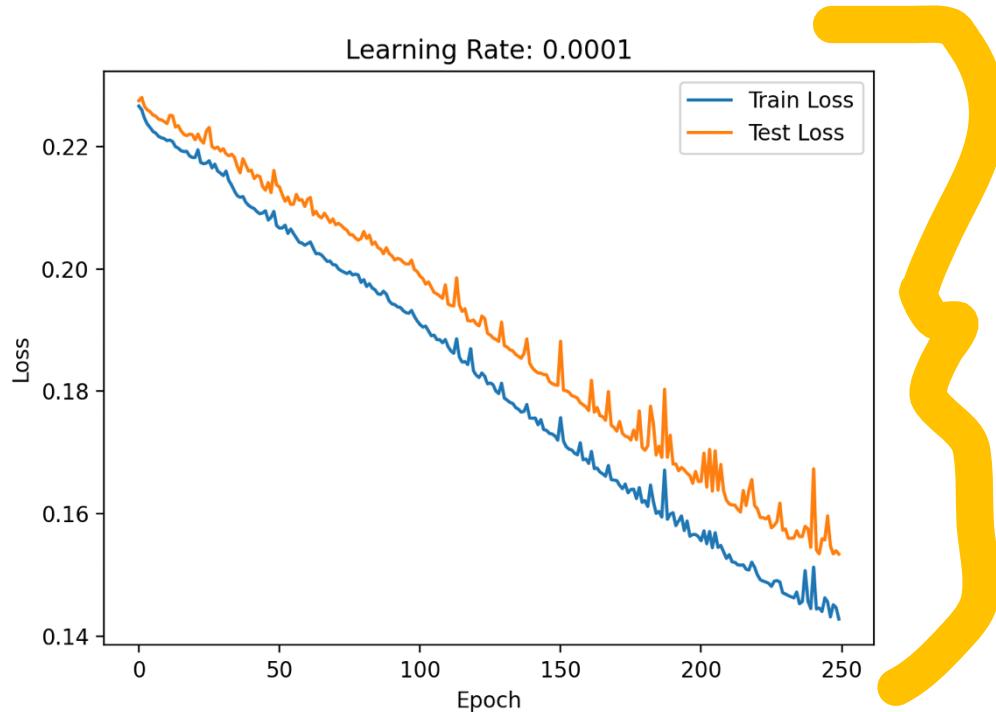
Learning Rate 1: Final Training Loss: 0.2868987808012704, Final Test Loss: 0.29761317519369274

Learning Rate 1: Train Accuracy: 0.62890625, Test Accuracy: 0.6140350877192983

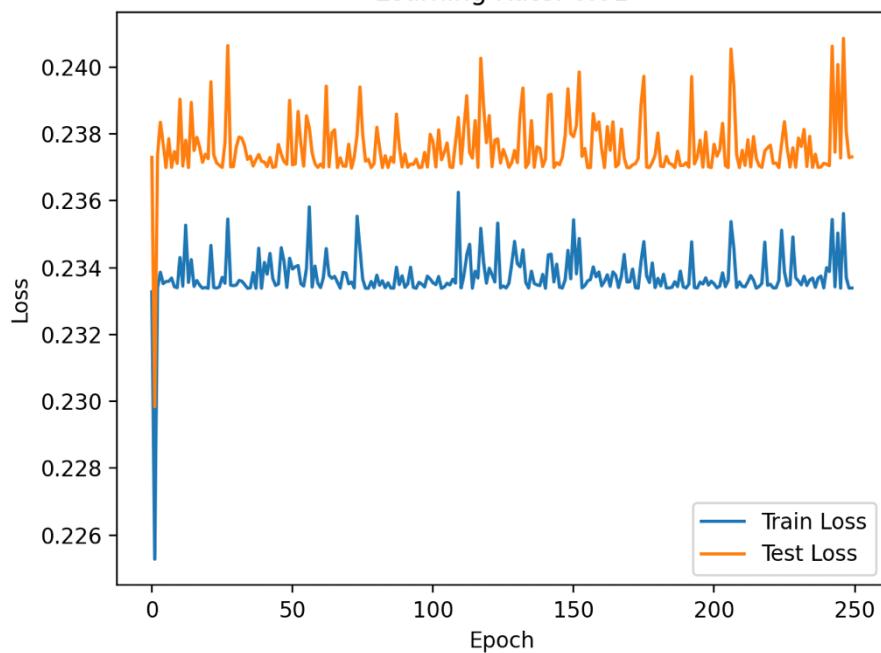
### Part B2.1 Comments on results:

1. Learning Rate 0.0001: The model shows stable and consistent learning with slow but steady convergence.
2. Learning Rate 0.001: The model converges faster but exhibits instability due to slight overshooting.
3. Learning Rate 0.01: The model fails to converge, showing flat loss curves and poor learning due to an excessively high learning rate.
4. Learning Rate 0.1: The model diverges with high fluctuations in loss, indicating severe instability and ineffective learning.

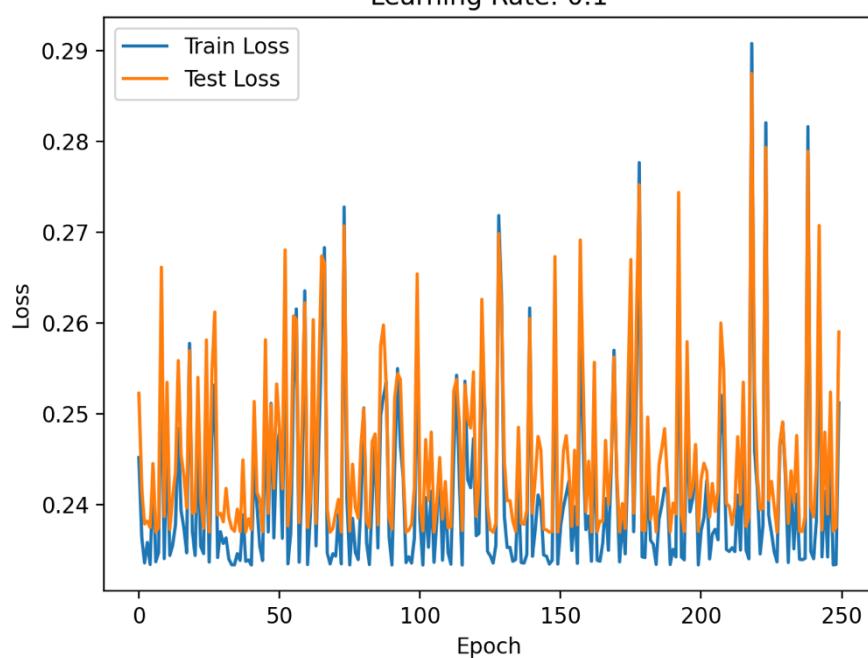
5. Learning Rate 1: The model exhibits extreme divergence with no signs of convergence, resulting in very poor performance.
6. Final Train and Test Loss vs Learning Rate: Increasing the learning rate beyond 0.001 leads to a sharp increase in loss, highlighting instability and poor convergence at higher rates.
7. Train and Test Accuracy vs Learning Rate: Accuracy drops drastically at learning rates above 0.001, indicating significant underfitting and divergence.

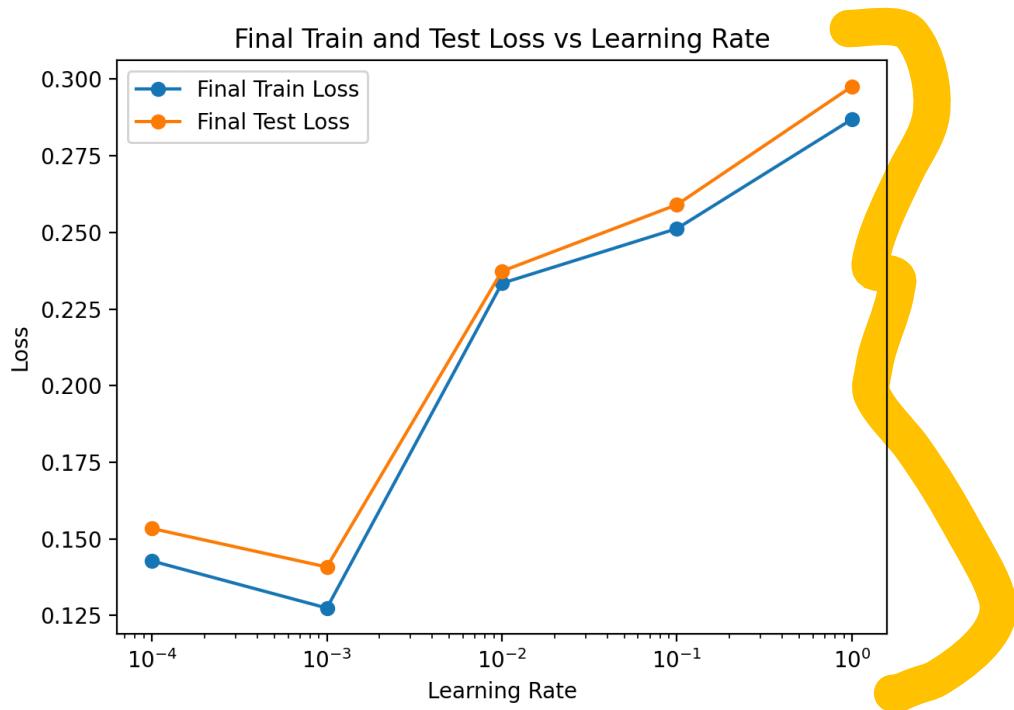
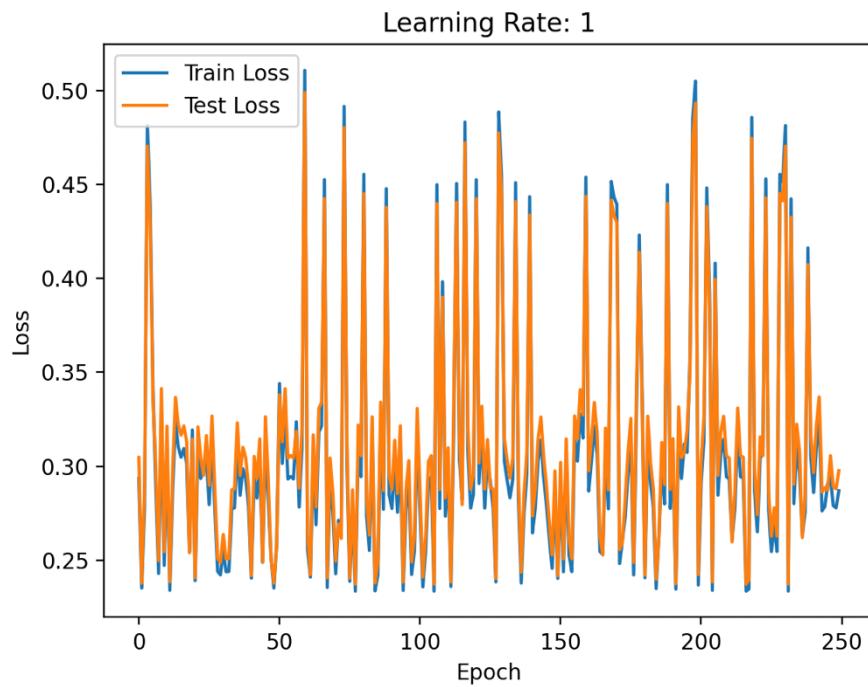


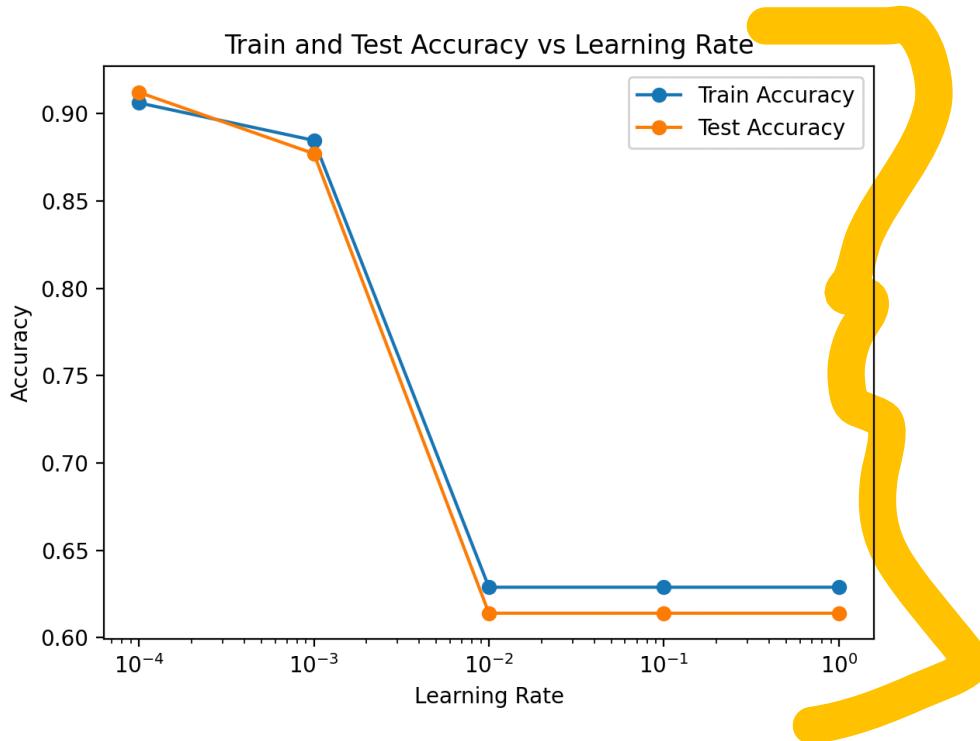
Learning Rate: 0.01



Learning Rate: 0.1







## B2.2

### Part B2.2: Hidden Layer Nodes Experiments

Final Training Loss for hidden nodes 8: 0.2334053522003992

Accuracy for hidden nodes 8: 0.6140350877192983

Final Training Loss for hidden nodes 16: 0.1687102039109753

Accuracy for hidden nodes 16: 0.8947368421052632

Final Training Loss for hidden nodes 32: 0.14279708800525165

Accuracy for hidden nodes 32: 0.9122807017543859

Final Training Loss for hidden nodes 64: 0.11931563219286795

Accuracy for hidden nodes 64: 0.8947368421052632

Final Training Loss for hidden nodes 128: 0.09172692232312457

Accuracy for hidden nodes 128: 0.8596491228070176

Final Training Loss for hidden nodes 256: 0.07399337798287386

Accuracy for hidden nodes 256: 0.9122807017543859

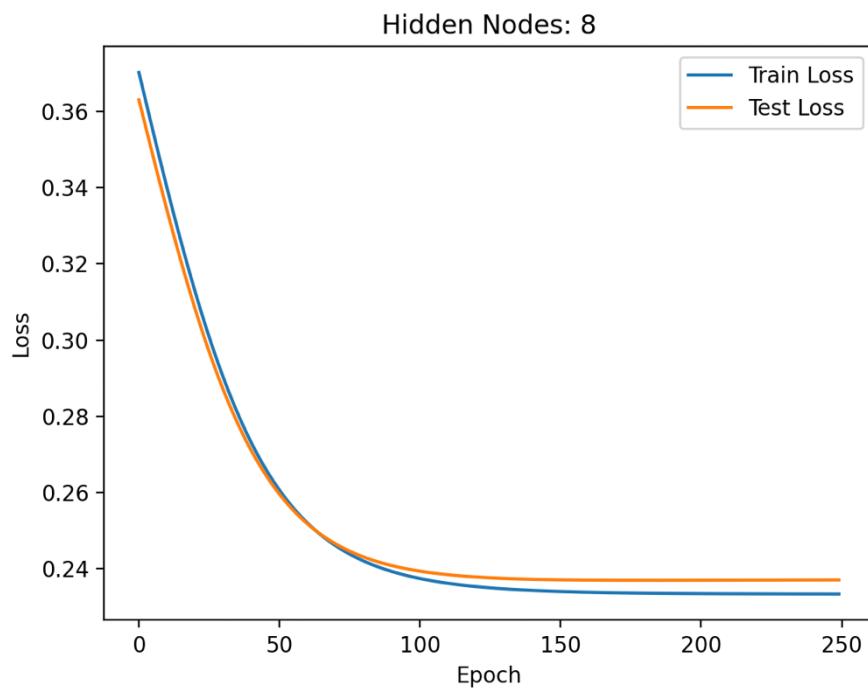
Final Training Loss for hidden nodes 512: 0.06217029535958994

Accuracy for hidden nodes 512: 0.9122807017543859

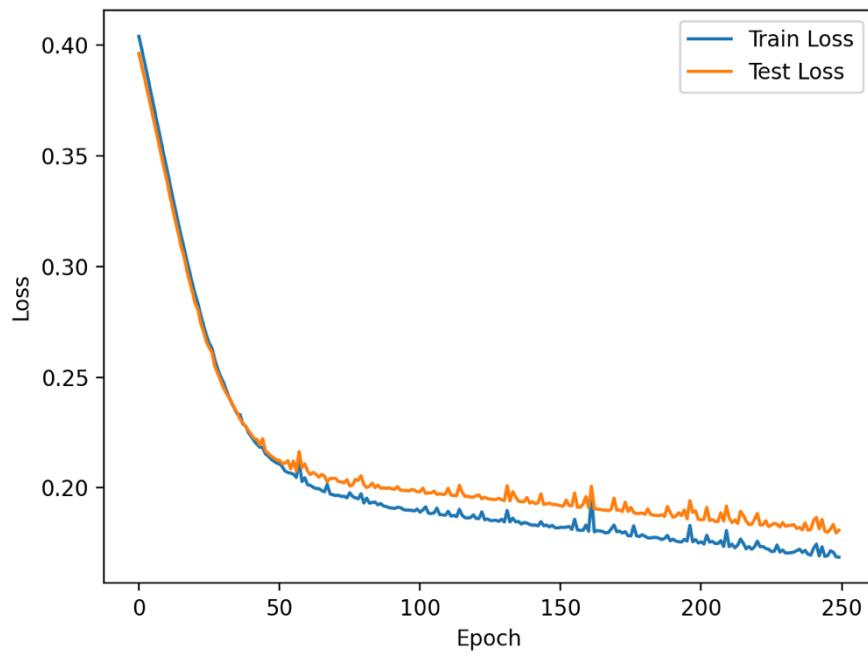
### Part B2.2 Comments on results:

1. Hidden Nodes: 8 - The model with 8 hidden nodes shows limited learning capacity, resulting in high loss and low accuracy.
2. Hidden Nodes: 16 - Increasing to 16 hidden nodes improves accuracy significantly, but slight overfitting begins to appear.
3. Hidden Nodes: 32 - With 32 hidden nodes, the model achieves high accuracy with minimal overfitting, indicating an optimal balance of capacity.
4. Hidden Nodes: 64 - The model continues to reduce training loss but shows a widening gap between training and test loss, suggesting increasing overfitting.
5. Hidden Nodes: 128 - The model shows clear signs of overfitting despite further reductions in training loss, with a slight decrease in accuracy.

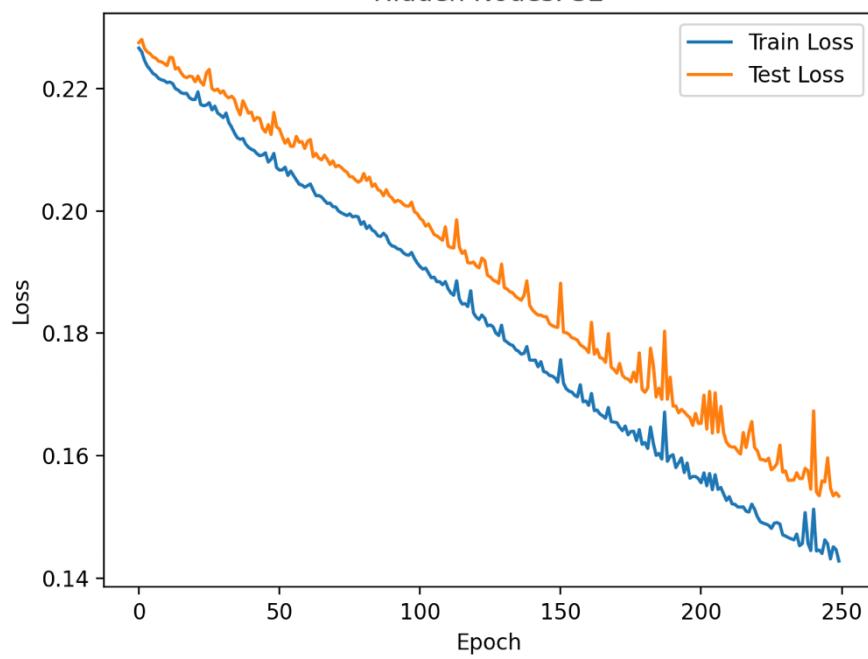
6. Hidden Nodes: 256 - Increasing to 256 nodes reduces both training and test loss effectively, maintaining high accuracy with manageable overfitting.
7. Hidden Nodes: 512 - The model with 512 hidden nodes achieves the lowest losses but shows persistent overfitting without further gains in accuracy.
8. Training Loss vs Hidden Nodes - Training loss consistently decreases as the number of hidden nodes increases, but with diminishing returns beyond a certain point.
9. Accuracy vs Hidden Nodes - Accuracy increases sharply initially and stabilizes around 32 nodes, with little to no improvement from further increasing the hidden nodes.



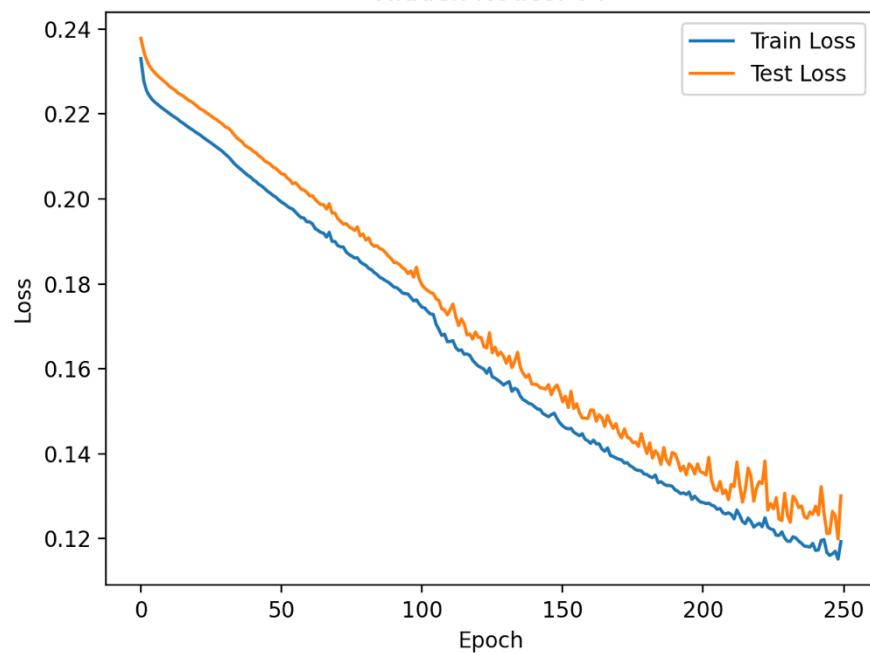
Hidden Nodes: 16



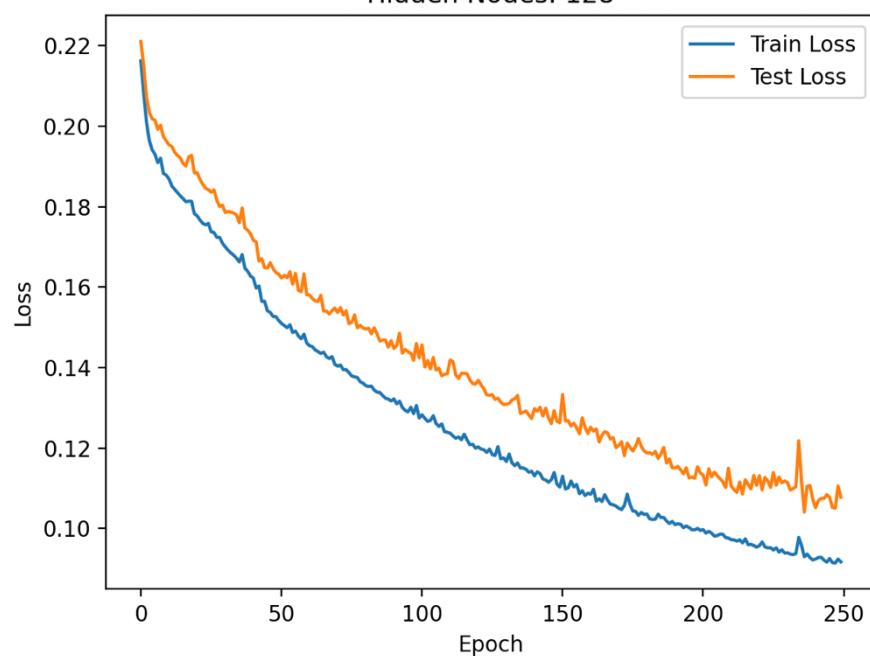
Hidden Nodes: 32



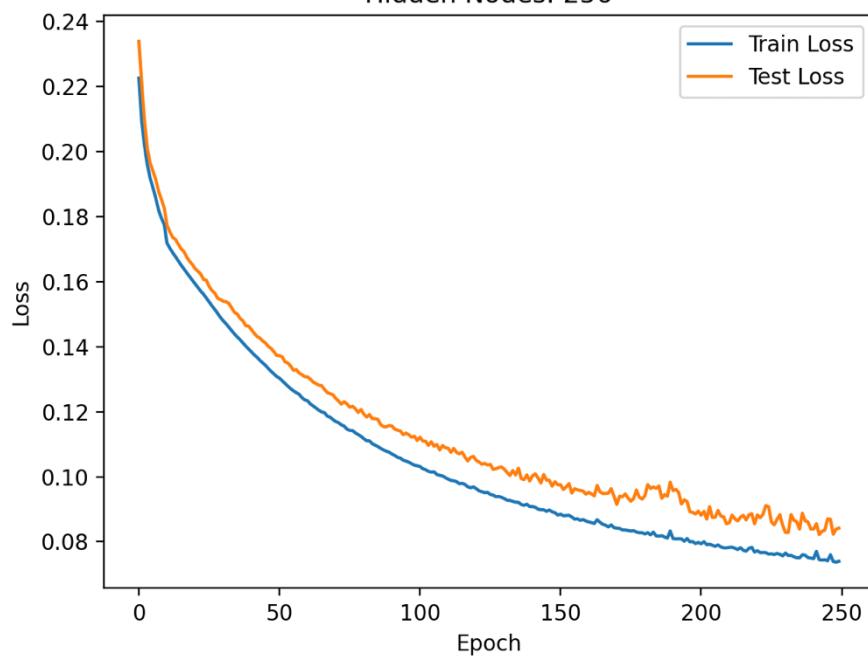
Hidden Nodes: 64



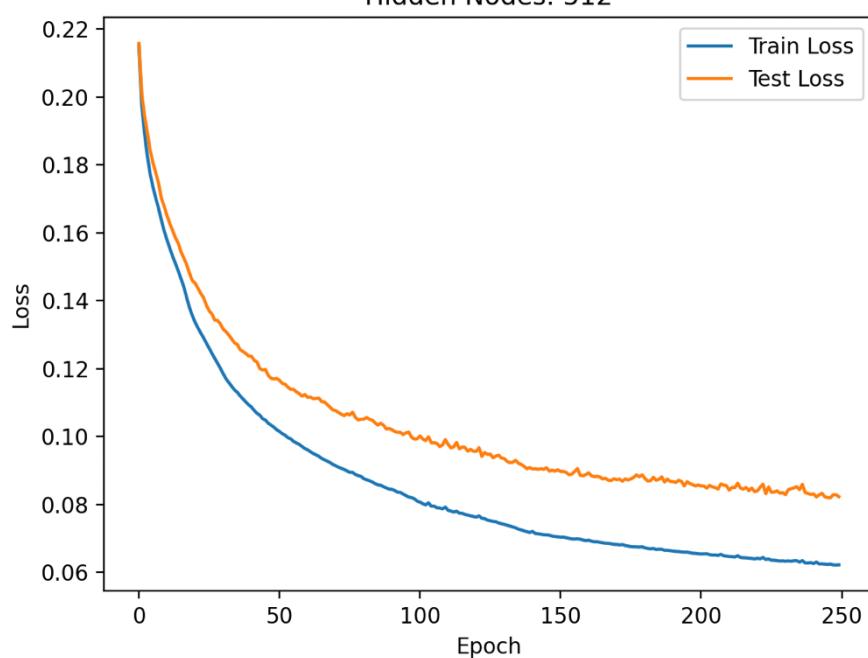
Hidden Nodes: 128

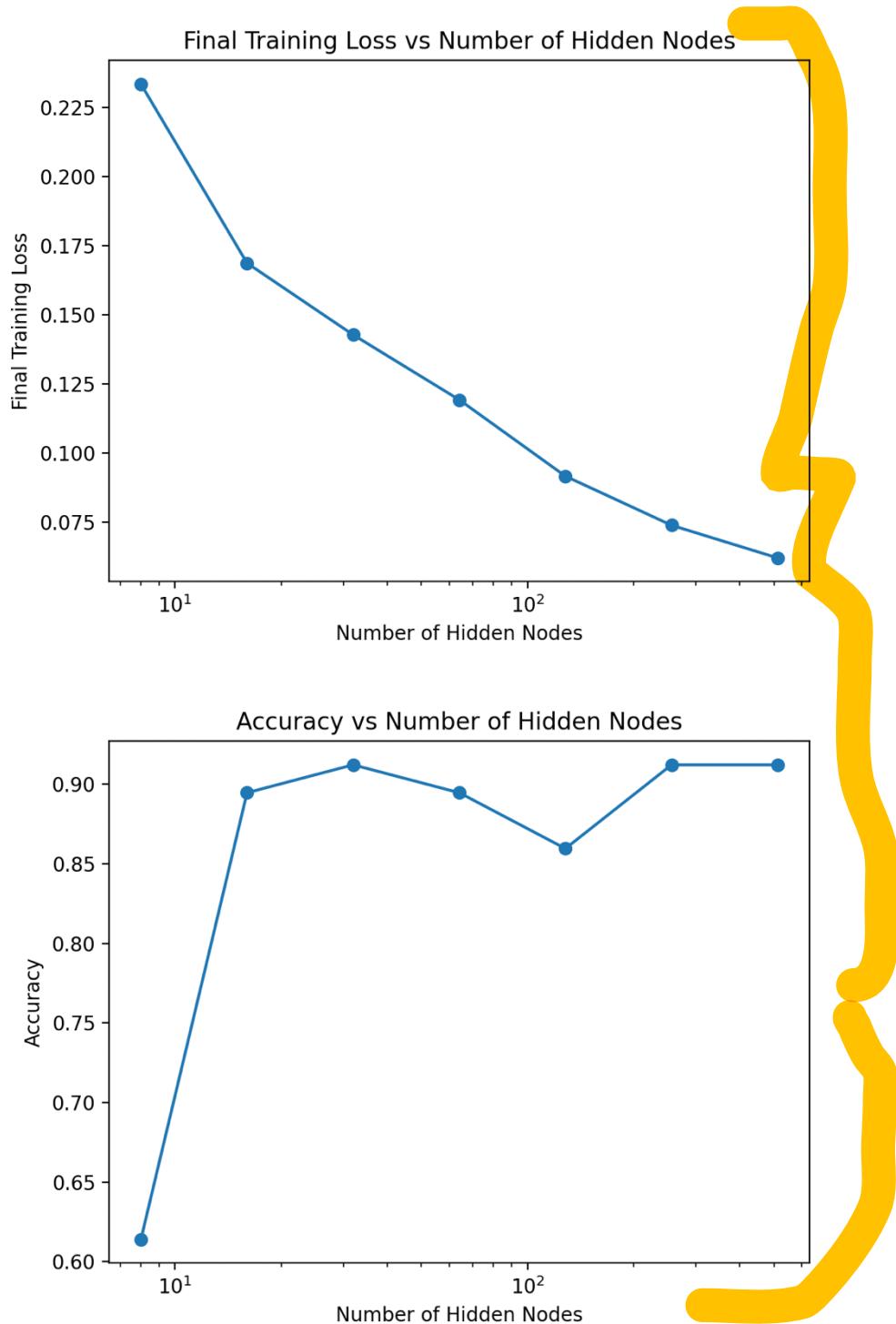


Hidden Nodes: 256



Hidden Nodes: 512





### B2.3

#### Part B2.3: Momentum Experiments

Final Training Loss for momentum 0: 0.12743397858488337

Final Test Loss for momentum 0: 0.14082828407433678

Accuracy for momentum 0: 0.8771929824561403

Final Training Loss for momentum 0.1: 0.15593047096789692

Final Test Loss for momentum 0.1: 0.1499506205132958

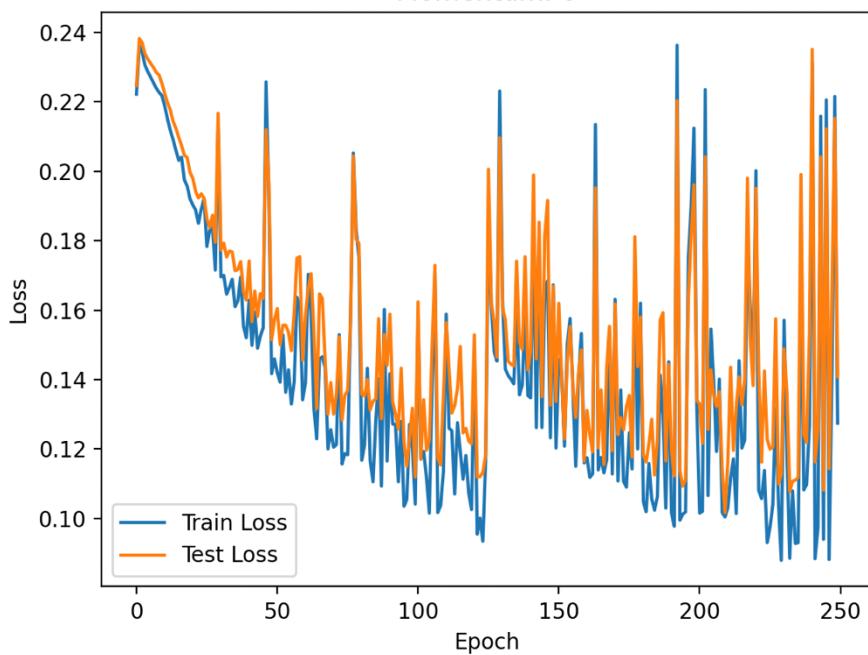
Accuracy for momentum 0.1: 0.8596491228070176

Final Training Loss for momentum 0.25: 0.2334591943659199  
Final Test Loss for momentum 0.25: 0.23702368444999464  
Accuracy for momentum 0.25: 0.6140350877192983  
Final Training Loss for momentum 0.5: 0.2333833860090173  
Final Test Loss for momentum 0.5: 0.2372319641293278  
Accuracy for momentum 0.5: 0.6140350877192983  
Final Training Loss for momentum 1: 0.37109375  
Final Test Loss for momentum 1: 0.38596491228070173  
Accuracy for momentum 1: 0.6140350877192983

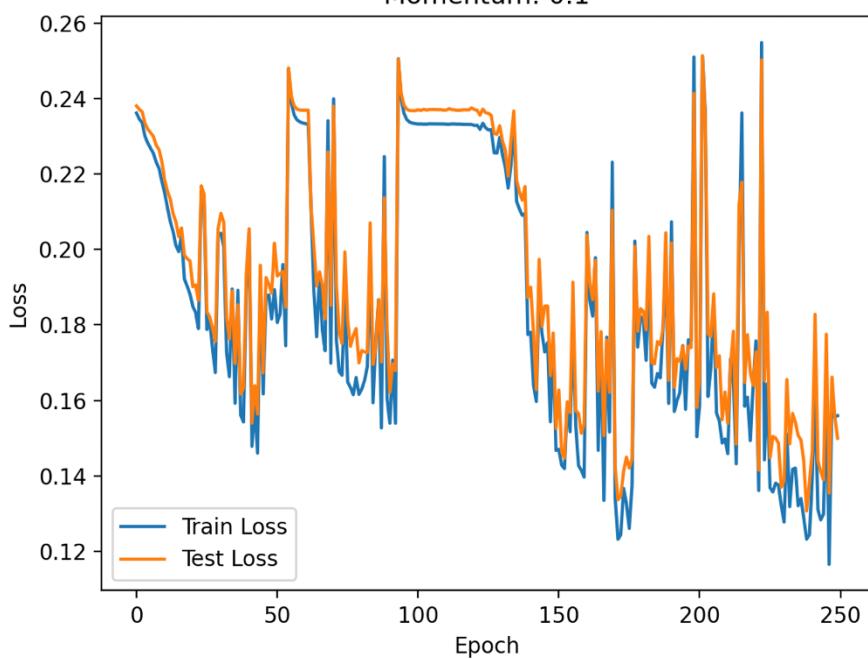
#### Part B2.3 Comments on results:

1. Graph 1 (Momentum = 0): Without momentum, the model shows decent performance but experiences noticeable fluctuations in loss values.
  2. Graph 2 (Momentum = 0.1): A small momentum helps reduce fluctuations slightly but leads to a minor increase in loss and a drop in accuracy.
  3. Graph 3 (Momentum = 0.25): Increasing the momentum further results in unstable training, higher losses, and a significant drop in accuracy.
  4. Graph 4 (Momentum = 0.5): The model exhibits plateauing behavior, indicating poor convergence and minimal learning progress.
  5. Graph 5 (Momentum = 1): The model fails completely with a momentum of 1, as both training and test losses remain high and constant throughout.
  6. Graph 6 (Final Loss vs. Momentum): Higher momentum values lead to consistently increasing losses, showing a negative impact on model performance.
  7. Graph 7 (Accuracy vs. Momentum): The accuracy decreases significantly when the momentum exceeds 0.1, indicating that higher momentum values lead to instability and poor convergence. This underscores the importance of carefully tuning the momentum, along with other hyperparameters, to maintain stable learning and achieve good generalization performance.
- Momentum hyperparameter is not functioning as expected, it is not able to improve the performance of the model, I suspect this is due to the fact that the other hyperparameters are not tuned properly, as we will see in the next part momentum of 0.8 gives the best results

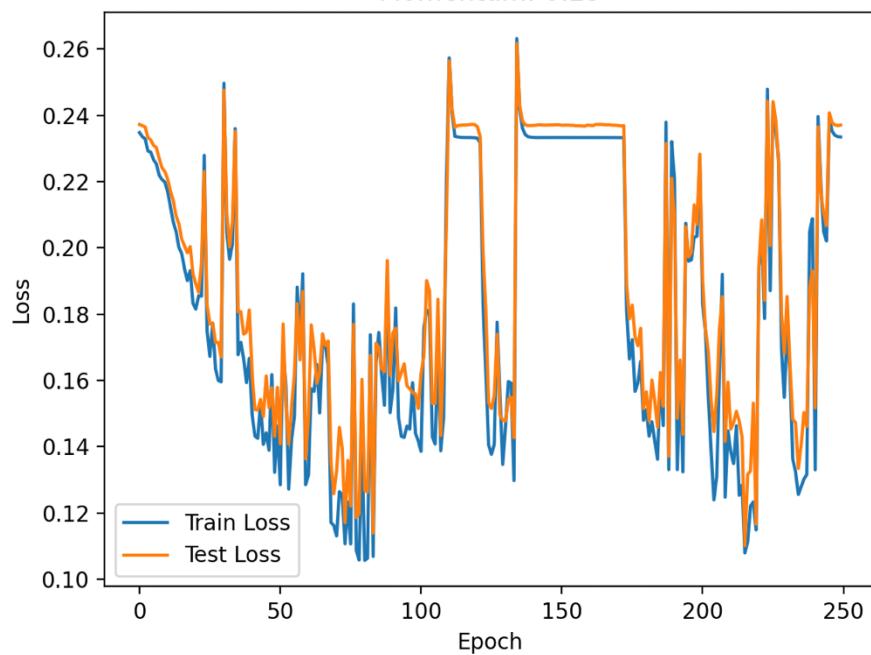
Momentum: 0



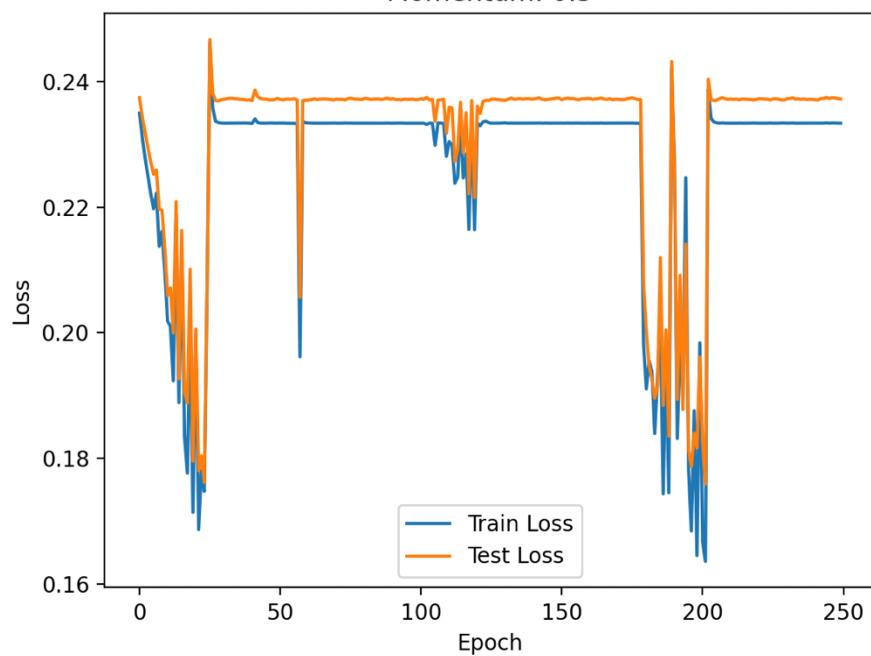
Momentum: 0.1

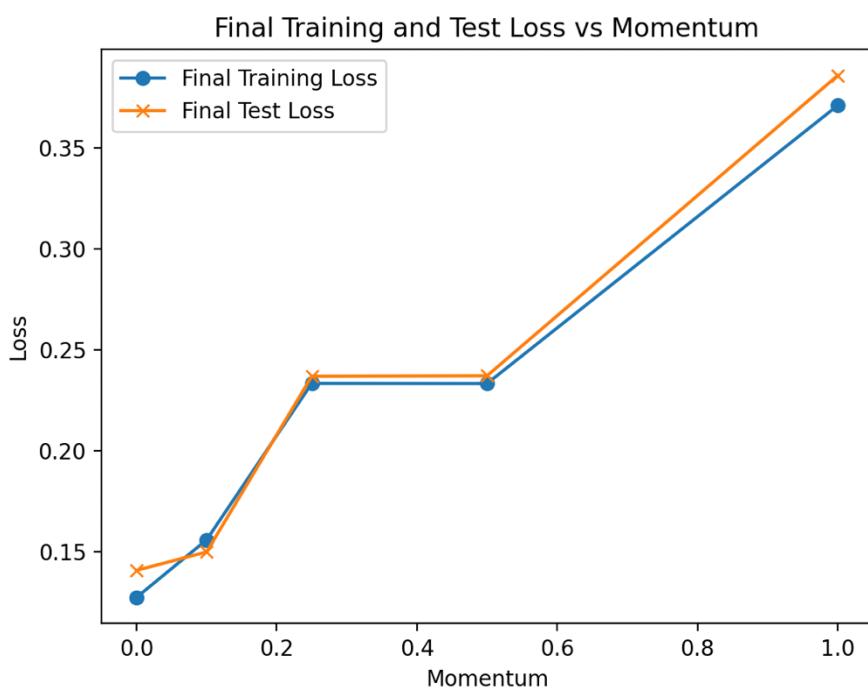
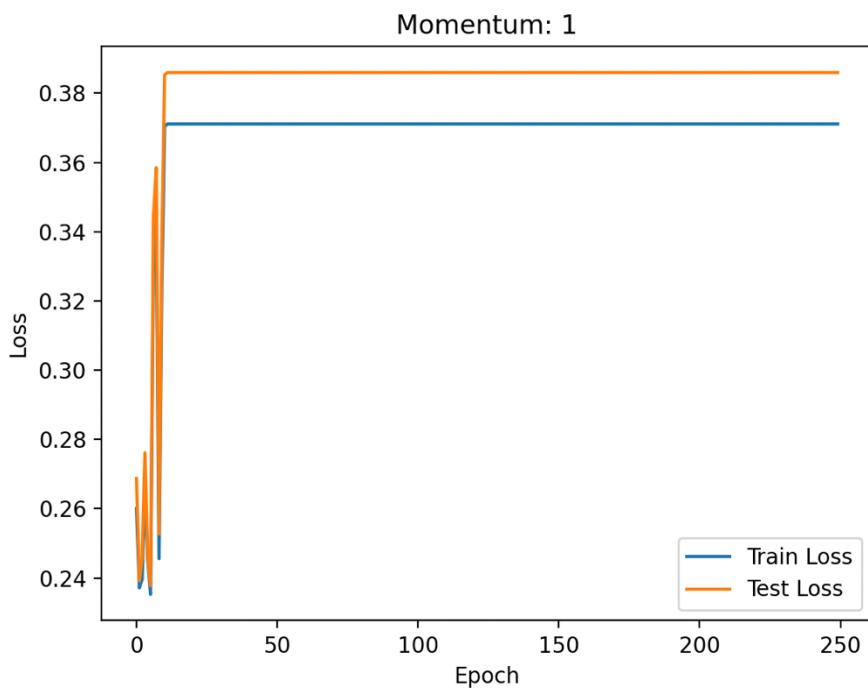


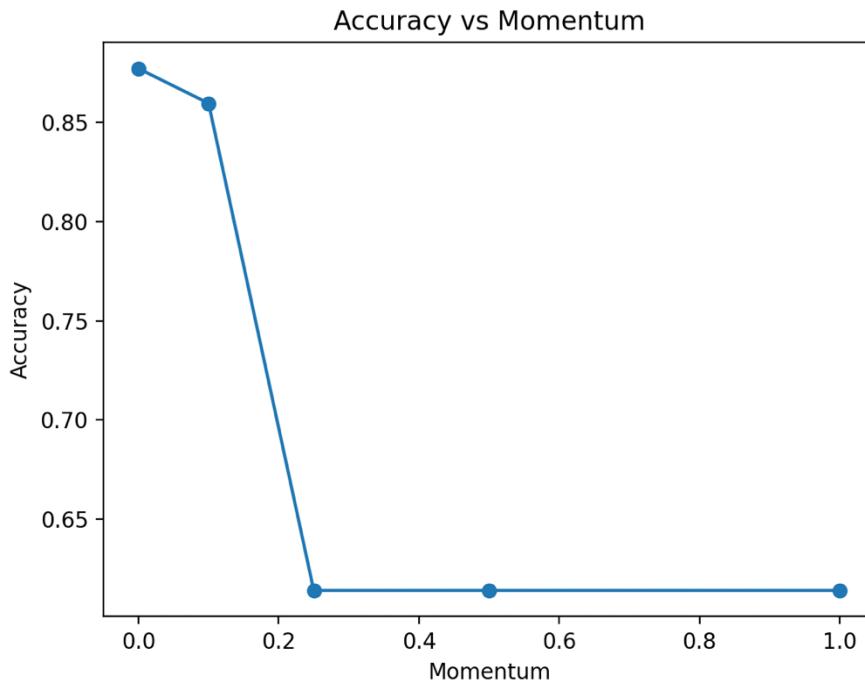
Momentum: 0.25



Momentum: 0.5







## B2.4

### Part B2.4: Final Neural Network – Hyperparameter finetuning

Training with learning\_rate=5e-05, hidden\_nodes=256, momentum=0.8, batch\_size=32

Validation Accuracy: 0.9824561403508771

Training with learning\_rate=5e-05, hidden\_nodes=256, momentum=0.8, batch\_size=64

Validation Accuracy: 0.9824561403508771

Training with learning\_rate=5e-05, hidden\_nodes=256, momentum=0.9, batch\_size=32

Validation Accuracy: 0.9824561403508771

Training with learning\_rate=5e-05, hidden\_nodes=256, momentum=0.9, batch\_size=64

Validation Accuracy: 0.9649122807017544

Training with learning\_rate=5e-05, hidden\_nodes=512, momentum=0.8, batch\_size=32

Validation Accuracy: 0.9649122807017544

Training with learning\_rate=5e-05, hidden\_nodes=512, momentum=0.8, batch\_size=64

Validation Accuracy: 0.9649122807017544

Training with learning\_rate=5e-05, hidden\_nodes=512, momentum=0.9, batch\_size=32

Validation Accuracy: 0.9649122807017544

Training with learning\_rate=5e-05, hidden\_nodes=512, momentum=0.9, batch\_size=64

Validation Accuracy: 0.9649122807017544

Training with learning\_rate=0.0001, hidden\_nodes=256, momentum=0.8, batch\_size=32

Validation Accuracy: 0.9824561403508771

Training with learning\_rate=0.0001, hidden\_nodes=256, momentum=0.8, batch\_size=64

Validation Accuracy: 0.9824561403508771

Training with learning\_rate=0.0001, hidden\_nodes=256, momentum=0.9, batch\_size=32

Validation Accuracy: 0.9473684210526315

Training with learning\_rate=0.0001, hidden\_nodes=256, momentum=0.9, batch\_size=64

Validation Accuracy: 0.9298245614035088

Training with learning\_rate=0.0001, hidden\_nodes=512, momentum=0.8, batch\_size=32

Validation Accuracy: 0.9649122807017544

Training with learning\_rate=0.0001, hidden\_nodes=512, momentum=0.8, batch\_size=64  
 Validation Accuracy: 0.9649122807017544  
 Training with learning\_rate=0.0001, hidden\_nodes=512, momentum=0.9, batch\_size=32  
 Validation Accuracy: 0.9473684210526315  
 Training with learning\_rate=0.0001, hidden\_nodes=512, momentum=0.9, batch\_size=64  
 Validation Accuracy: 0.9298245614035088

Best hyperparameter combination based on validation accuracy:

learning_rate	0.00005
hidden_nodes	256
momentum	0.8
batch_size	32
<b>val_accuracy</b>	<b>0.982456</b>
combination	lr=5e-05, hn=256.0, m=0.8, bs=32.0
Name:	0, dtype: object

Final Training Loss: 0.06677854787119471

Test Accuracy: 0.9649122807017544

