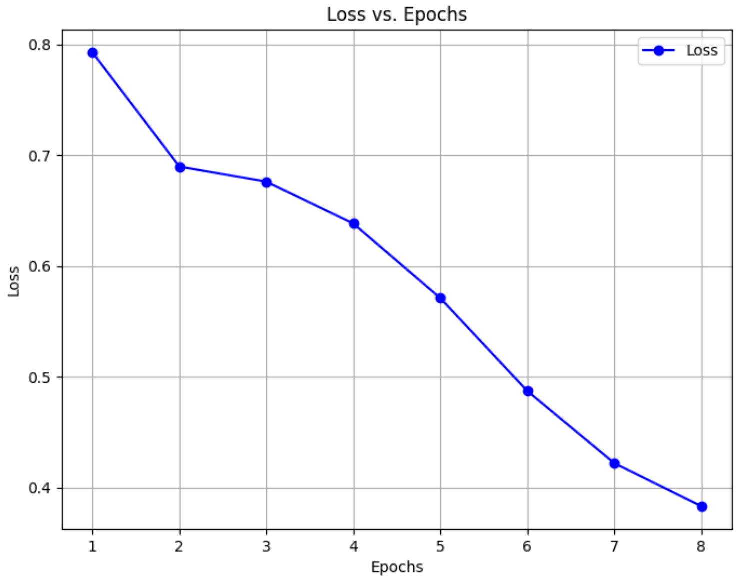
COL774 A2.2 Report

Part A

1. The accuracy of the model on public_test is 80.0%. This suggests that the model was able to generalize well on unseen data, with notable improvement across the epochs.

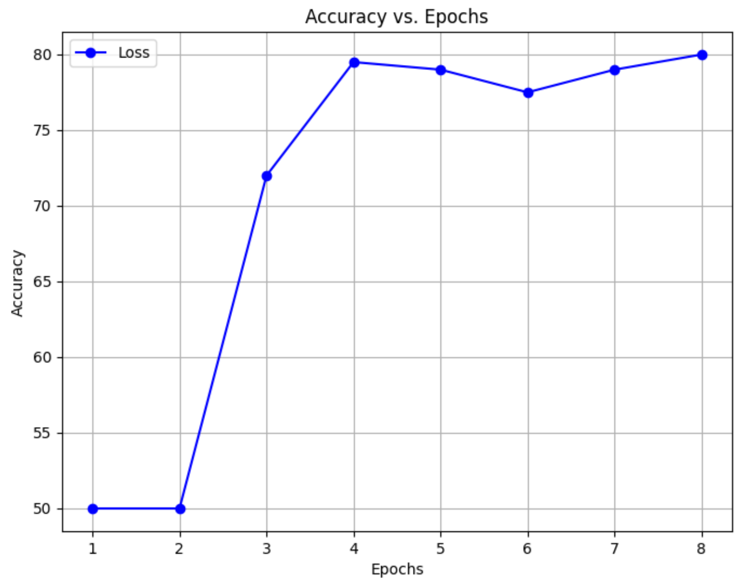
2. The graph below represents the variation of training_loss vs epoch:



The loss starts at
0.8 during the first epoch and gradually decreases to **0.38** by epoch 8.
This consistent drop in loss indicates that the model effectively minimized the error during training as it learned the underlying patterns in the data.

The steep initial decrease in loss signifies that the model made significant progress in the early epochs, followed by a slower decline, which is common in deep learning training.

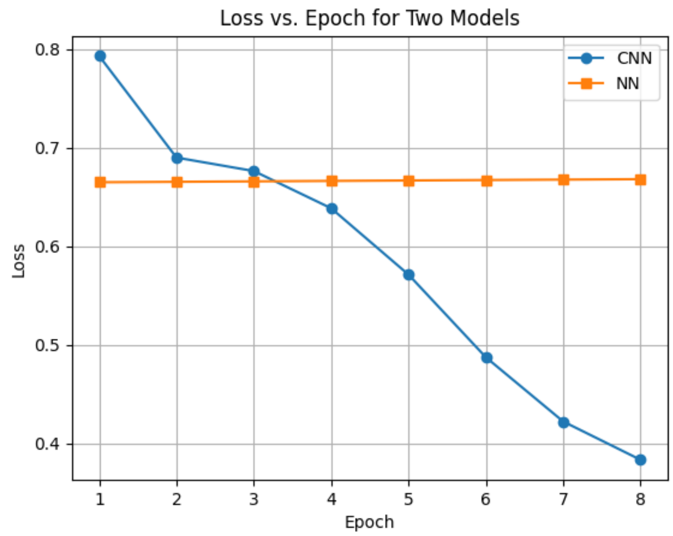
3. The graph below shows the variation of accuracy on public test data vs the epoch number:



The accuracy started at
50% in epoch 1 and increased steadily, reaching **80%** in epoch 8. This substantial improvement suggests that the model effectively captured relevant features in the data with more epochs of training.

The consistent increase in accuracy indicates that the model avoided overfitting and was able to generalize better with each additional epoch.

4. **Comparision:** The graph below compares the loss versus epochs for two models, CNN (Convolutional Neural Network) and NN (Neural Network), on a binary classification dataset.



a. **CNN Loss Curve**:

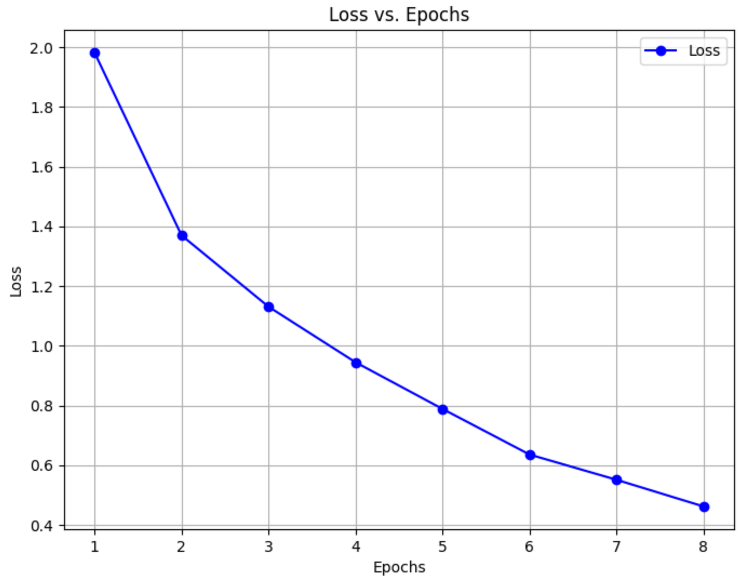
- The CNN model's loss starts at a higher value (around 0.8) in the first epoch but decreases significantly over the epochs.
- It shows a steep decline between the first and third epochs and continues to drop consistently throughout the 8 epochs, reaching approximately 0.35 by the end.
- The steep drop indicates that the CNN model is learning effectively as it reduces the loss with each epoch, showing strong convergence.

b. **NN Loss Curve**:

- The NN model starts with a lower initial loss (around 0.7) compared to CNN but remains almost constant across all epochs.
- The nearly flat curve suggests that the NN model is not learning or improving over time. This could indicate several issues, such as improper learning rate, overfitting, or poor initialization, preventing the model from optimizing the loss further.

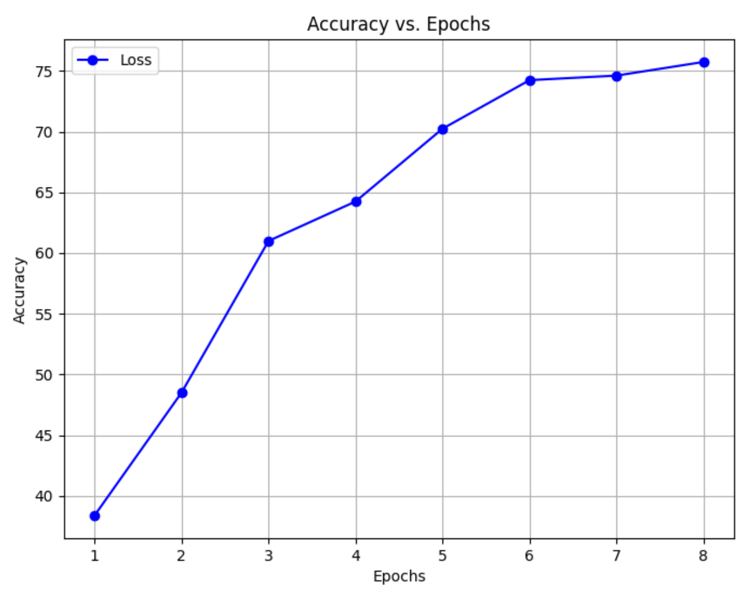
Part B

1. The accuracy of the model on public_test is 75.75%. This result is slightly lower than in Part A, which could be attributed to the complexity of the dataset.
2. The graph below represents the variation of training_loss vs epoch:



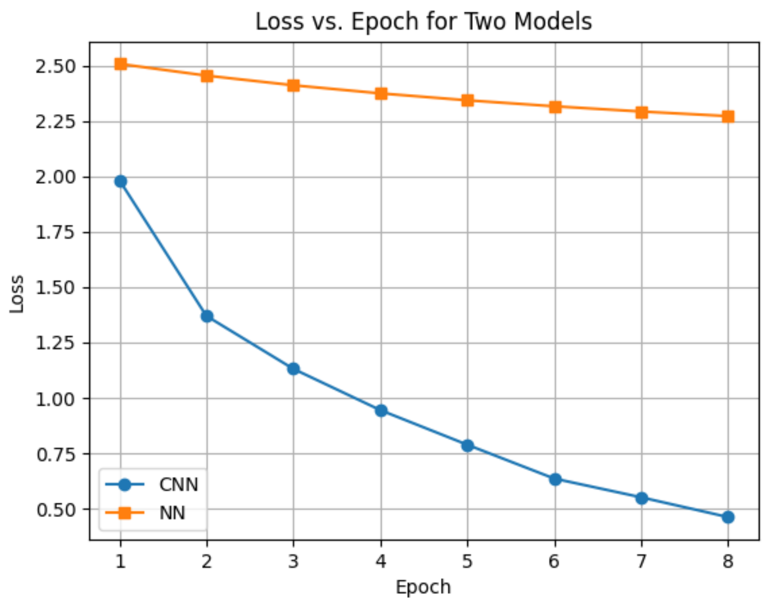
The loss varies from 2.0 in epoch 1 to 0.43 in epoch 8. The sharp drop in loss between the first and second epochs suggests a rapid improvement in early training, after which the loss decreases at a slower but consistent rate.

3. The graph below shows the variation of accuracy on public test data vs the epoch number:



The accuracy varies from 32% in epoch 1 to 75.75% in epoch 8. The initial lower accuracy may be due to more complex patterns or noise in the dataset, but the model's performance still improved consistently.

4. **Comparison:** The graph below compares the loss versus epochs for two models, CNN (Convolutional Neural Network) and NN (Neural Network), on a multi classification dataset.



a. **CNN Loss Curve**:

- The CNN model starts with a loss of about 2.0 and decreases rapidly, especially during the initial epochs. By the 8th epoch, the loss has dropped to about 0.5.
- The sharp decline in loss suggests that the CNN is learning effectively and is able to improve its performance consistently over time.
- The steep reduction in loss for a multi-class dataset highlights the model's ability to capture the underlying structure of the data, possibly leveraging spatial features due to the convolutional layers.

b. **NN Loss Curve**:

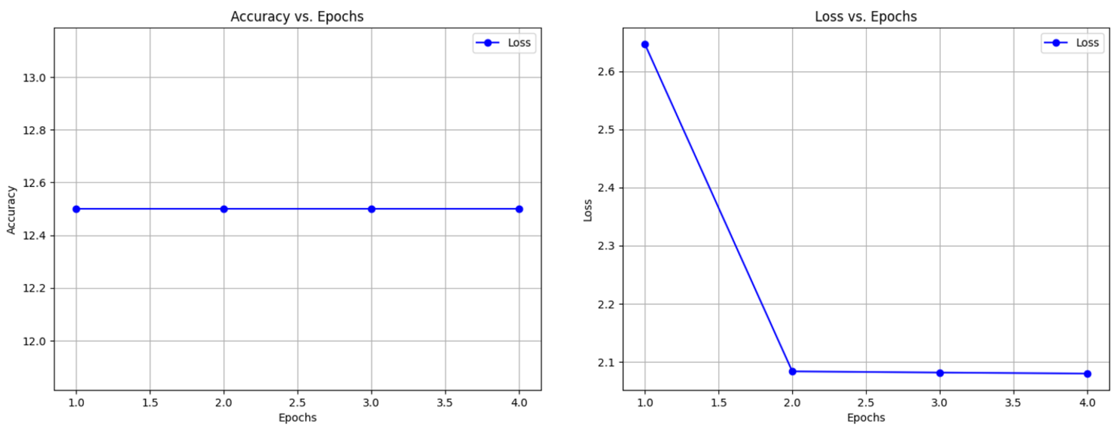
- The NN model starts with a higher initial loss, around 2.5, and only slightly reduces the loss over the course of the 8 epochs. By the 8th epoch, the loss is around 2.2, showing minimal improvement.

- The slower reduction of loss indicates that the NN model is struggling to learn from the data. The curve is relatively flat, which may suggest that the model is not able to effectively capture complex patterns or that the architecture is not well-suited for this multi-class classification task.

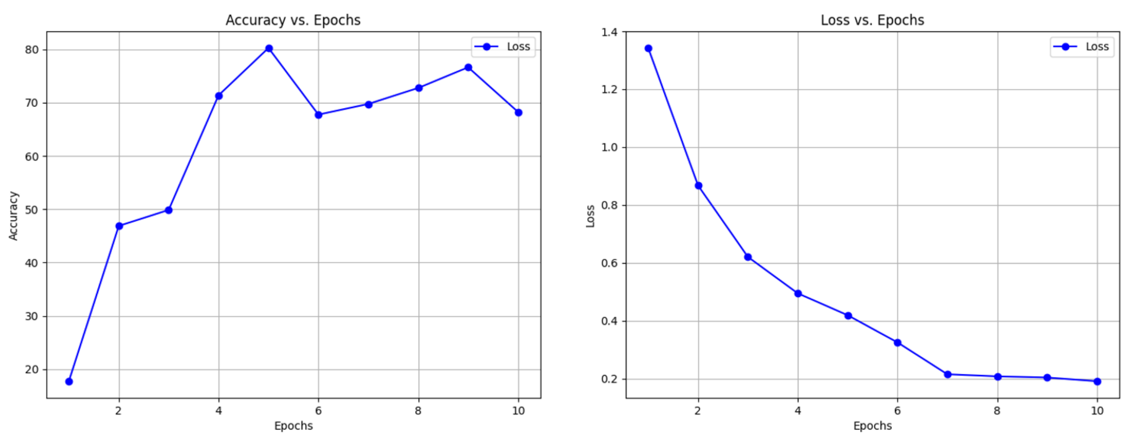
Part C

1. Comparison between AlexNet, ResNet and VGGNet archtitecture model performance:

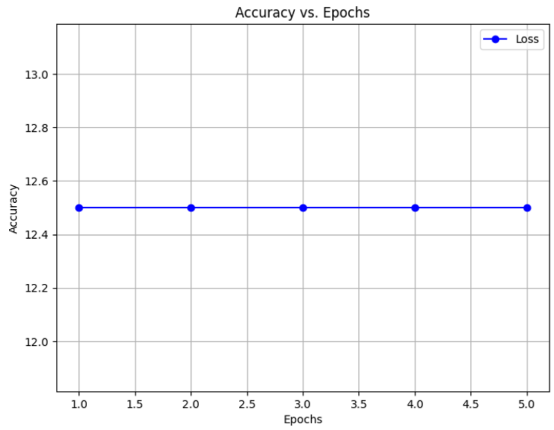
    a. AlexNet

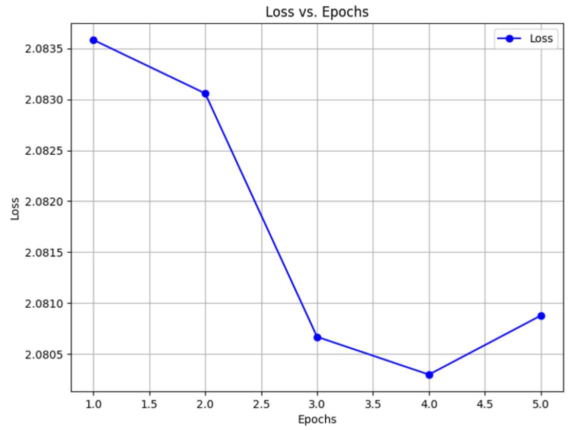


    b. ResNet



    c. VGGNet



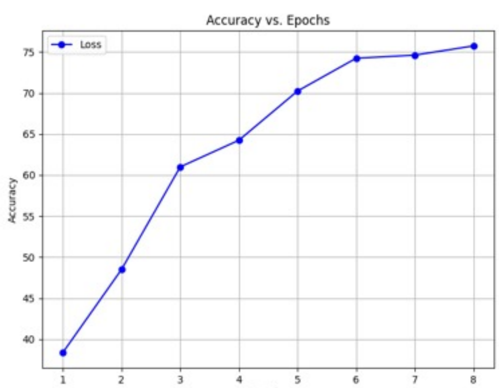


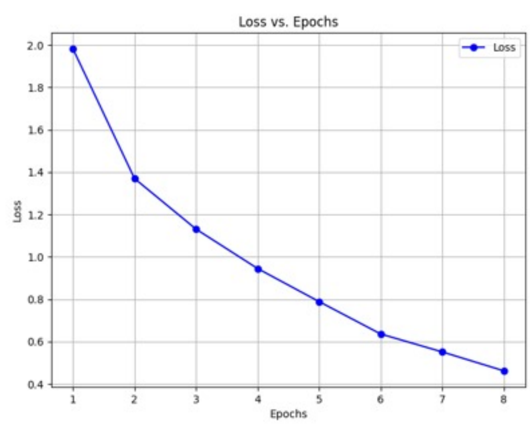
AlexNet and VGGNet: Both architectures exhibit **no improvement in accuracy**, indicating possible data-model incompatibility or training issues, with accuracy stagnating at around 12%.

ResNet: This architecture performs **significantly better**, with accuracy rising to around 80% over 10 epochs, showing ResNet's ability to learn from the data effectively. The fluctuations in accuracy after epoch 5 may require further investigation or regularization techniques to stabilize learning.

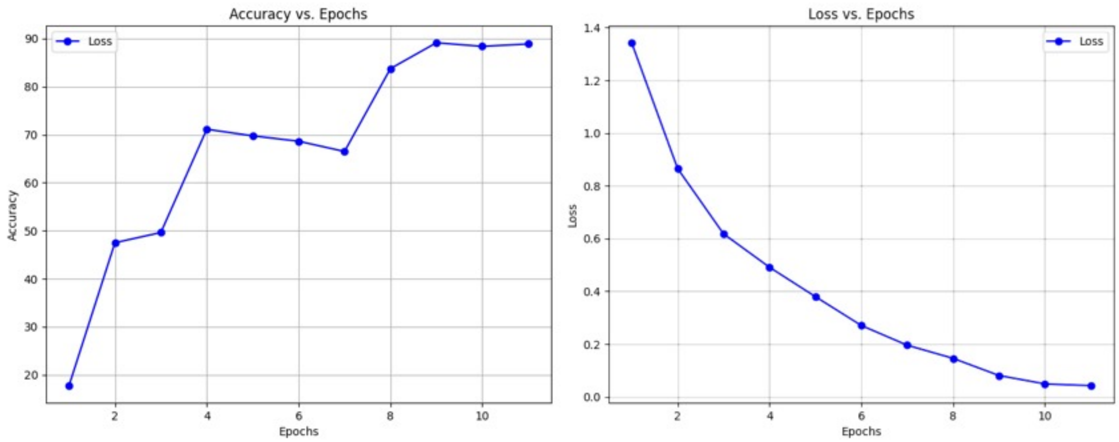
2. Improvements in ResNet Architecture:

    a. AdamW Optimizer for Enhanced Weight Decay

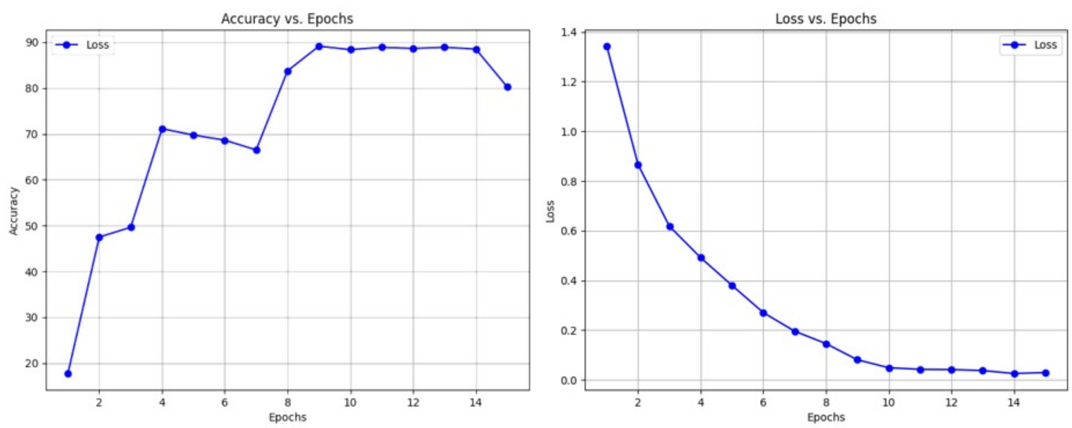




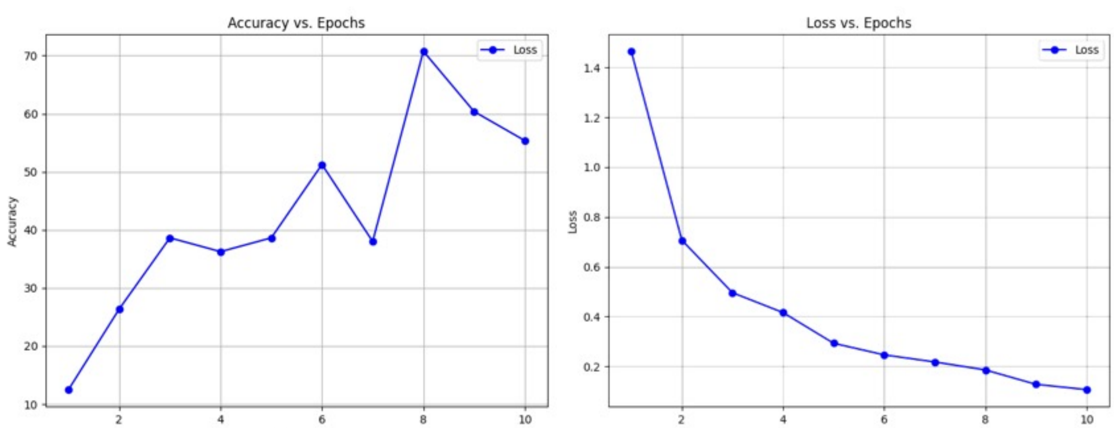
    b. Cosine Annealing Scheduler for Adaptive Learning Rate



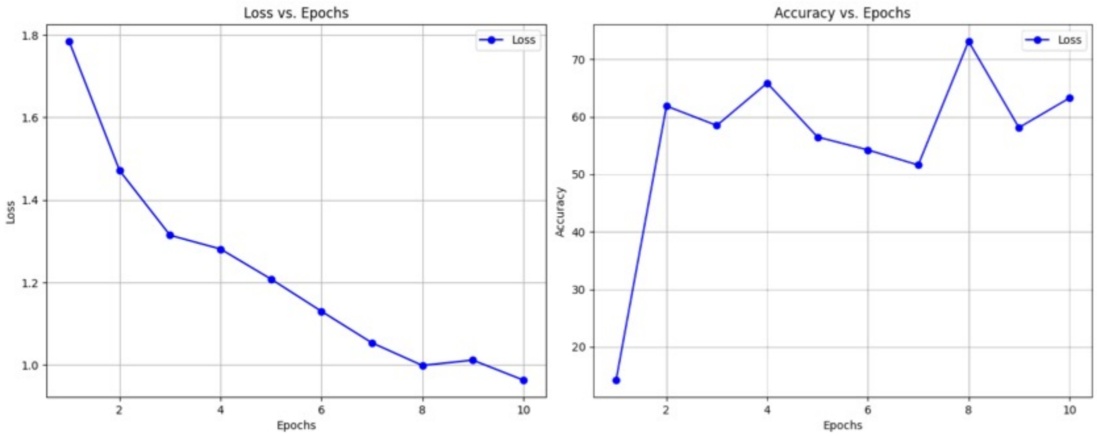
c. Combining AdamW Optimizer and Cosine Annealing for Synergistic Results



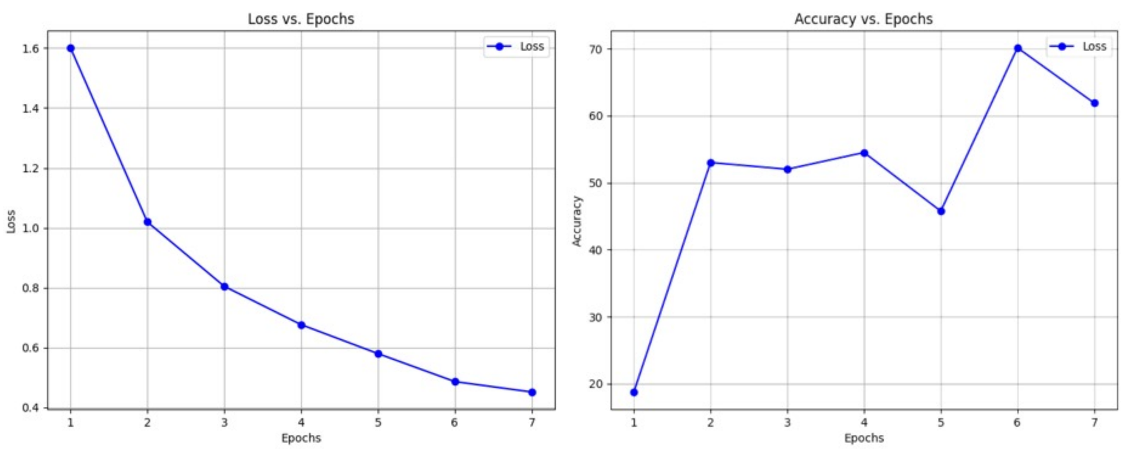
d. Advanced Weight Initialization Techniques for Better Convergence



e. Improving Generalization with Batch Normalization and Dropout



f. Deeper ResNet Architectures for Complex Feature Extraction



Summary

Strategies	Accuracy (in %)
ResNet Baseline	86.15
AdamW Optimizer	91.62
Cosine Annealing Scheduler	91.25
Combined Optimizer and Scheduler	90.56
Weight Initialization	88.72
Batch Normalization & Dropout	78.46

Deep ResNet	77.59

The above analysis shows that AdamW Optimizer and Cosine Annealing Scheduler performs the best compared to all, therefore we have implemented AdamW Optimizer in our final submission.