CS540 Programming Assignment 2 Write up

Saeah Go

January 2025

Q1 a

Learning Rate	Activation Function	Loss Function	Train Loss at 0th Epoch	Train Loss at Last Epoch	Test Loss at 0th Epoch	Test Loss at Last Epoch
0.1	Sigmoid	MSE	0.0902	0.0902	0.0904	0.0903
0.1	Sigmoid	CrossEntropy	2.3073	2.3033	2.3077	2.3026
0.1	Tanh	MSE	0.0893	0.0675	0.0873	0.0683
0.1	Tanh	CrossEntropy	2.0242	0.5306	1.7801	1.5338
0.01	Sigmoid	MSE	0.0943	0.0900	0.0900	0.0900
0.01	Sigmoid	CrossEntropy	2.3049	2.3031	2.3031	2.3027
0.01	Tanh	MSE	0.0924	0.0802	0.0899	0.0800
0.01	Tanh	CrossEntropy	2.2957	1.1692	2.2857	1.2142
0.001	Sigmoid	MSE	0.1138	0.0900	0.0922	0.0900
0.001	Sigmoid	CrossEntropy	2.3396	2.3026	2.3157	2.3026
0.001	Tanh	MSE	0.1072	0.0892	0.1025	0.0892
0.001	Tanh	CrossEntropy	2.3042	2.0127	2.3034	2.0071

Figure 1: Loss Table

Since we have 24 plots for Q1a, we have placed every plot in the Supplement section.

Models with Sigmoid Activation

In Figure 1, we observe that when using the sigmoid activation function, regardless of the loss function (MSE or Cross-Entropy) or learning rate (0.1, 0.01, and 0.001), both the training and test losses show no changes at all (the difference between the initial loss and final loss is 0 or 0.0001). This indicates that the model cannot properly learn features. Sigmoid activation can cause vanishing gradient problems, making training slow and less effective. Moreover, sigmoid is commonly used for binary classification, which could lead to issues in multi-class problems because it does not normalize the outputs across classes like softmax does.

Models with Tanh Activation and MSE Loss

With the Tanh activation function, the model performance is better compared to Sigmoid. With a learning rate of 0.1, Tanh activation, and the MSE loss function, we achieve approximately 50% test accuracy (Figure 8). The loss difference is not significant: its initial value is 0.08, and at the final epoch, the loss is 0.06 (a difference of only 0.02) (Figure 9). At a learning rate of 0.01, we achieve lower accuracy (approximately 35% test accuracy), and the loss difference is only 0.01 (0.09 loss at the 0th epoch and 0.08 loss at the final epoch) (Figure 16, Figure 17). Similarly, at a learning rate of 0.001, we achieve approximately 20% test accuracy, and the loss change is only 0.02 (Figure 24, Figure 25). Tanh activation with the MSE loss function does not perform well in classification tasks because the output of Tanh does not directly represent probabilities, and MSE does not handle probabilities effectively.

Models with Tanh Activation and Cross Entropy Loss

With a learning rate of 0.1, Tanh activation, and the CrossEntropy loss function, we observe overfitting. The training loss decreases monotonically (from 2.02 to 0.53), but the test loss exhibits a slight V-shape (starting at 1.78, reaching a minimum of 1.17, and then increasing again to 1.53). The test loss decreases slightly but increases after the 15th epoch (Figure 10, Figure 11). This behavior suggests that the model memorizes data instead of learning features. With a learning rate of 0.01, the model performs better; we observe both training and test losses decrease (from 2.3 to 1.2), with approximately 60% test accuracy (Figure 18, Figure 19). This makes sense because, at a learning rate of 0.1, the model learns too quickly and eventually memorizes the data. By using a learning rate of 0.01, the model learns features more gradually.
At a learning rate of 0.001, the model achieves approximately 25% test accuracy, and the loss decreases from 2.3 to 2.0 (a difference of approximately 0.3) (Figure 26, Figure 27). This occurs because, at a learning rate of 0.001, the model's weights are updated very slowly compared to 0.01. The model seems to get stuck in local minima, leading to oscillations in both training and test accuracy. Additionally, the loss plot (Figure 27) resembles a decreasing sigmoid (S-shaped curve), which occurs because the loss decreases gradually, leading to vanishing gradients. In conclusion, with a learning rate of 0.001, the test accuracy is lower, and the loss difference is smaller (around 0.3) than in the model with a learning rate of 0.01.
The best parameter set among the 12 combinations is a learning rate of 0.01, Tanh activation function, and Cross Entropy loss. With this combination, we do not observe overfitting, and we achieve the highest test accuracy (approximately 60%).

Q1 b

Since there are so many visuals for this section, all of the images have been placed in the Supplement section.
The visualizations of the feature maps at the C3 layer show that the network is successfully detecting key features such as edges, boundaries, textures, and more abstract patterns crucial for distinguishing CIFAR-10 classes. The dark squares

indicate small or inhibitory weights and the light squares represent large weights. It is a challenging task to interpret each individual feature map in detail, we can observe that the CNN has learned to detect important low-level features like edges, textures, and shapes. The feature maps highlight various aspects of the images, with some focusing on object boundaries and others emphasizing texture patterns. This suggests that the network is effectively learning to extract meaningful features necessary for distinguishing between the different CIFAR-10 categories.

Q2

Since there are so many plots for this section, all of the graphs have been placed in the Supplement section.

Hypothesis

The previous network (Sigmoid & Tanh models) training is slow due to vanishing gradients, and the MSE loss worsens the performance because it treats classification as a regression problem. In contrast, ReLU avoids vanishing gradients and reaches faster convergence (learning better features). Additionally, the ReLU model achieves lower training/test loss values compared to non-ReLU models. Therefore, my hypothesis before the Q2a experiment was that the model with a 0.001 learning rate, ReLU activation, and Cross Entropy loss function would perform better than the best parameter set in Q1 (0.01 learning rate, Tanh activation, and Cross Entropy loss function).

Result & Conclusion

With a 0.001 learning rate, ReLU activation, CrossEntropy loss function, and 40 epochs, I obtained $\approx 16\%$ train and test accuracies, and the loss changed from 2.3 to 2.27 (Figure 39, Figure 38). This result seems worse than the best parameter set model from Q1, but it is still better than the Sigmoid model from Q1. I thought the number of epochs might not have been sufficient to achieve better results, so I tried increasing it to 100 epochs. The model's accuracy improved significantly, reaching 40% accuracy, and the loss changed from 2.3 to 1.69. However, this result is still not better than Q1's best model. I suspect this is because the ReLU model uses 3x3 kernels instead of 5x5, which prevents it from converging quickly. Thus, I decided to try adding momentum to the ReLU model.

Instead of increasing the number of epochs further, I kept it at 40 epochs but added 0.9 momentum to the ReLU model. With this setup, I achieved $\approx 62\%$ test accuracy and a significant improvement in loss (from 2.3 to 1.1) (Figure 41, Figure 40). This performance is better than the best parameter set in Q1 (0.01 learning rate, Tanh activation, and Cross Entropy loss function). The Q1 best model achieved $\approx 58\%$ test accuracy, with the loss decreasing from 2.3 to 1.2. Thus, we can conclude that the ReLU + 0.9 momentum model produces slightly better results. I also tested the model with 0.9 momentum and an increased number of epochs, but I observed overfitting in the ReLU model (Figure 42). Therefore, the best parameter set for ReLU is with 40 epochs and 0.9 momentum.

Q3 a

a

Figure 2: Q3a

The blue color represents train loss, and the orange color represents test loss. The loss chart shows that both train and test losses significantly decrease after a few epochs (after 5 epochs). At the 0th epoch, the train loss is 0.35, and the test loss is 0.23, but at the last epoch, both losses are 0.000002. This indicates that the model learns the sine wave very well, which makes sense as it can predict the sine wave.

Q3 b

Figure 3: Q3b

The blue color represents train loss, and the orange color represents test loss. The loss chart's result shape looks very similar to Q3a, the sine wave RNN model. Both have an elbow shape. The loss chart shows a significant decrease after 2-3 epochs. At the 0th epoch, the train loss is 0.05, and the test loss is 0.0004 (which differs from predicting $\sin(x)$, as the train and test losses are already very small numbers at the 0th epoch with Dow Jones data). At the 100th epoch, the train loss is 0.00008, and the test loss is 0.00005. I was able to verify that the predicted values closely follow the true values.

Supplements

This section includes all of the visuals for this assignment. Regarding line plots, blue represents training dataset, orange represents test dataset.

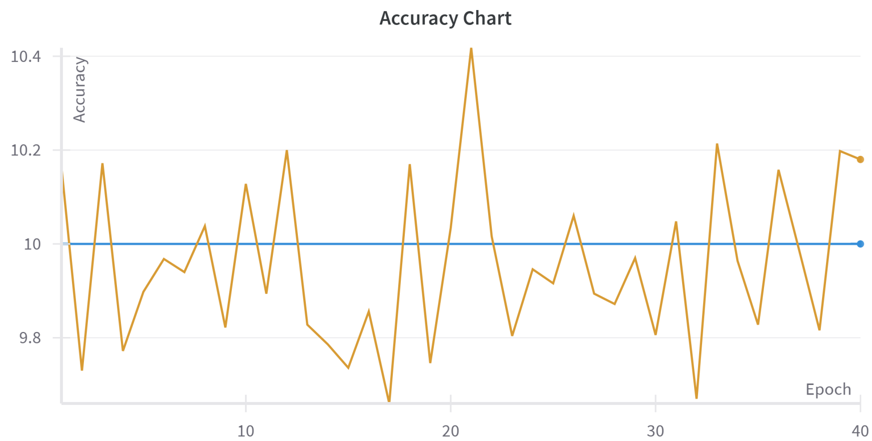

Figure 4: Q1a LR=0.1, Sigmoid, MSE Accuracy Plot

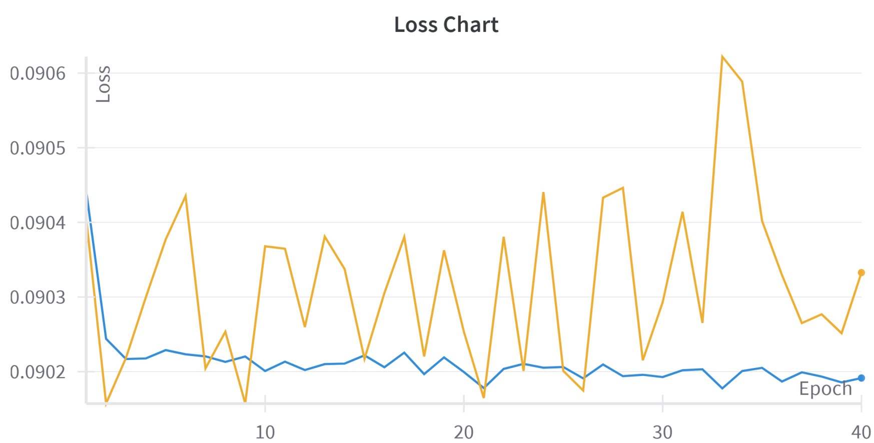

Figure 5: Q1a LR=0.1, Sigmoid, MSE Loss Plot

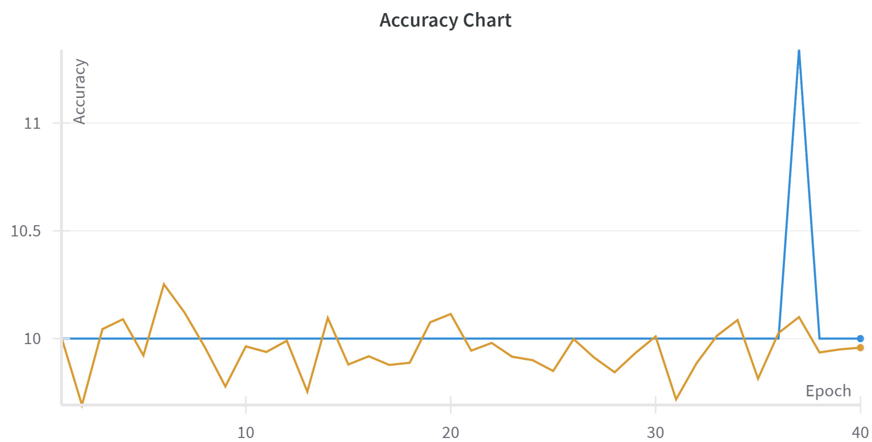

Figure 6: Q1a LR=0.1, Sigmoid, Cross Entropy Accuracy Plot

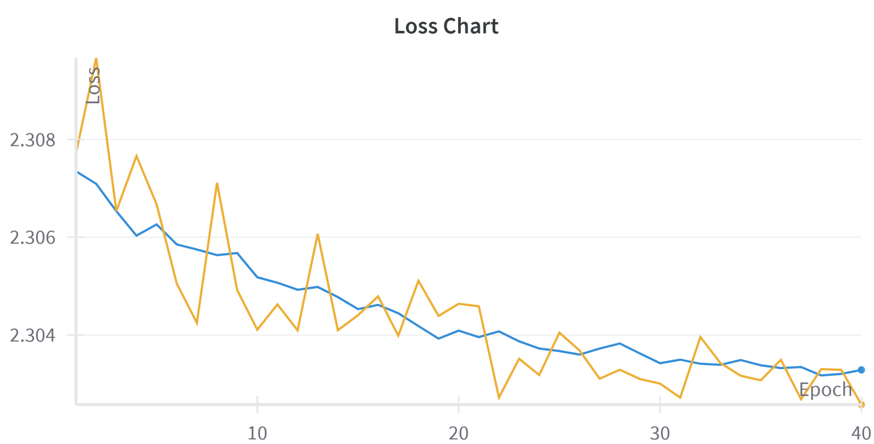

Figure 7: Q1a LR=0.1, Sigmoid, Cross Entropy Loss Plot

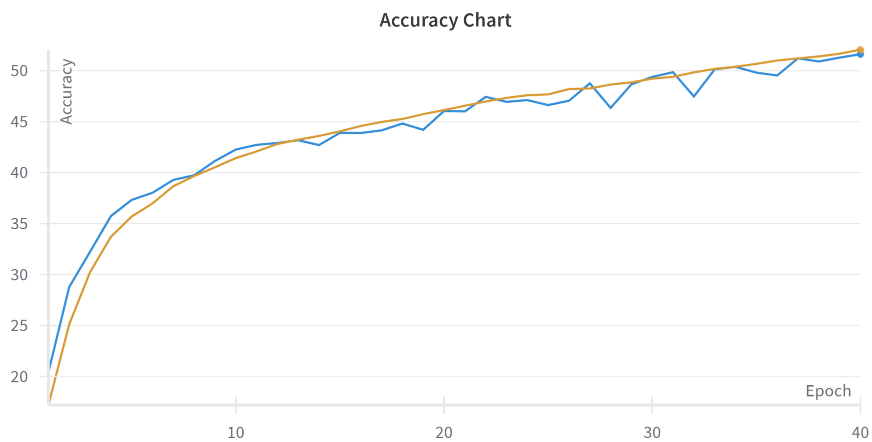

Figure 8: Q1a LR=0.1, Tanh, MSE Accuracy Plot

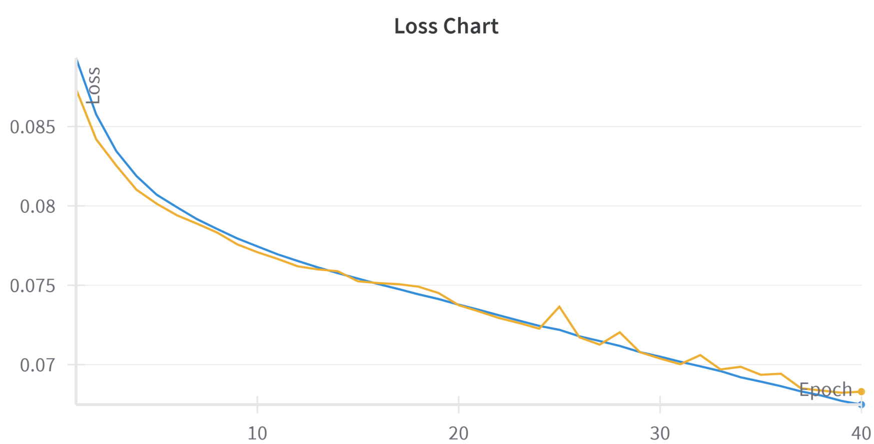

Figure 9: Q1a LR=0.1, Tanh, MSE Loss Plot

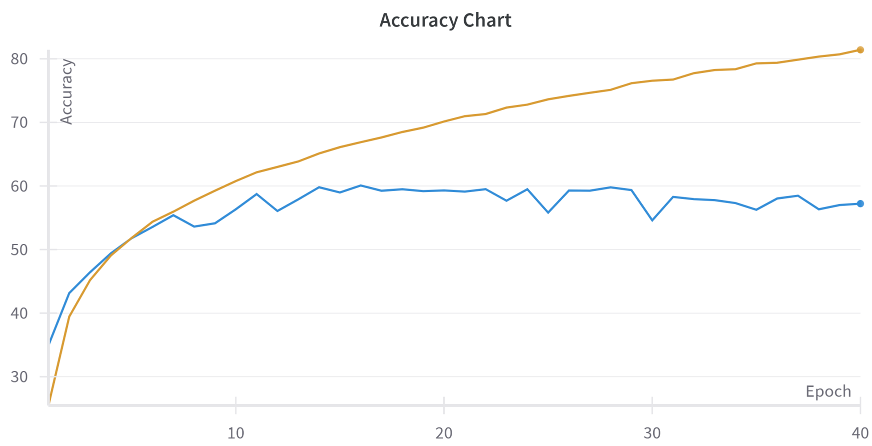

Figure 10: Q1a LR=0.1, Tanh, Cross Entropy Accuracy Plot

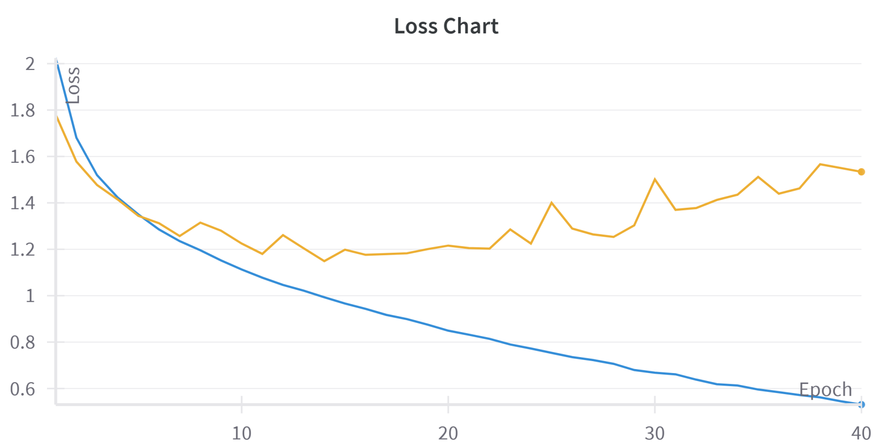

Figure 11: Q1a LR=0.1, Tanh, Cross Entropy Loss Plot

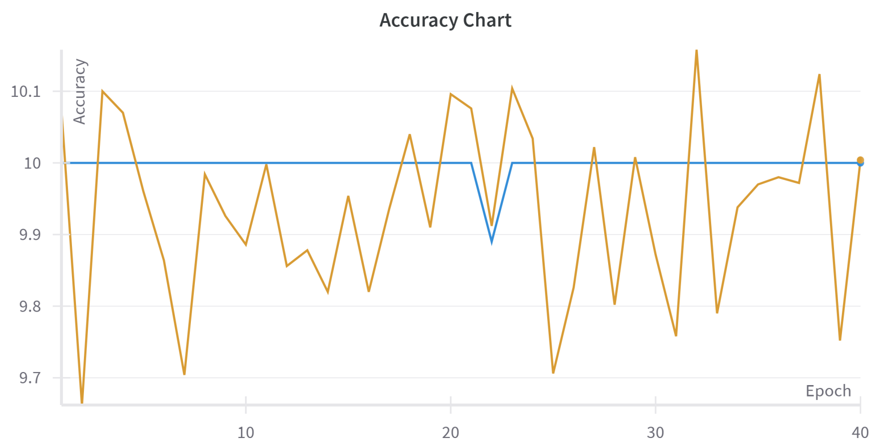

Figure 12: Q1a LR=0.01, Sigmoid, MSE Accuracy Plot

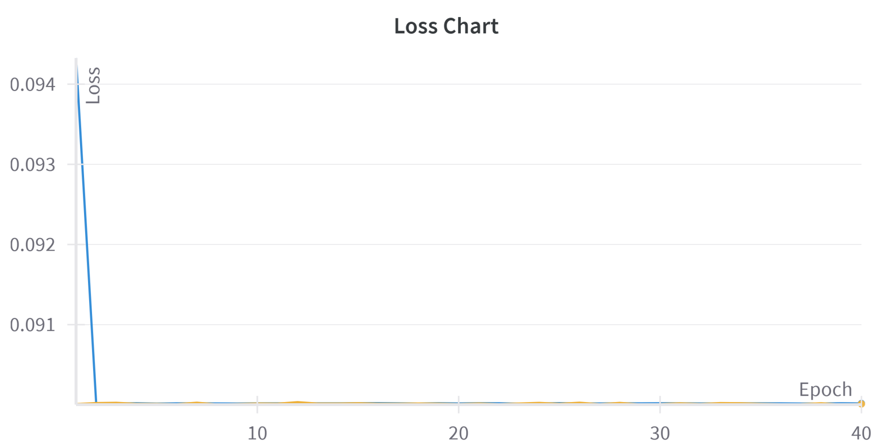

Figure 13: Q1a LR=0.01, Sigmoid, MSE Loss Plot

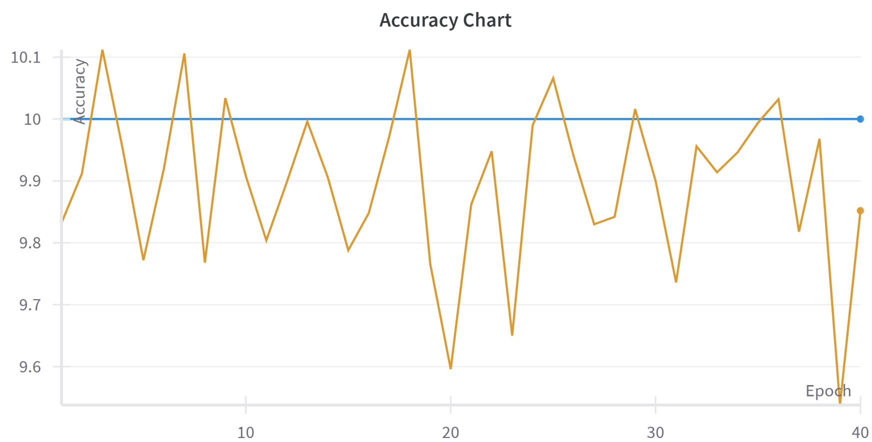

Figure 14: Q1a LR=0.01, Sigmoid, Cross Entropy Accuracy Plot

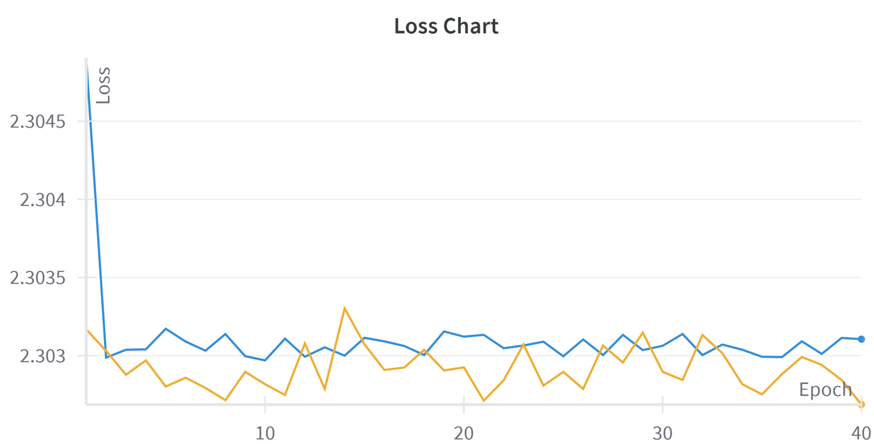

Figure 15: Q1a LR=0.01, Sigmoid, Cross Entropy Loss Plot

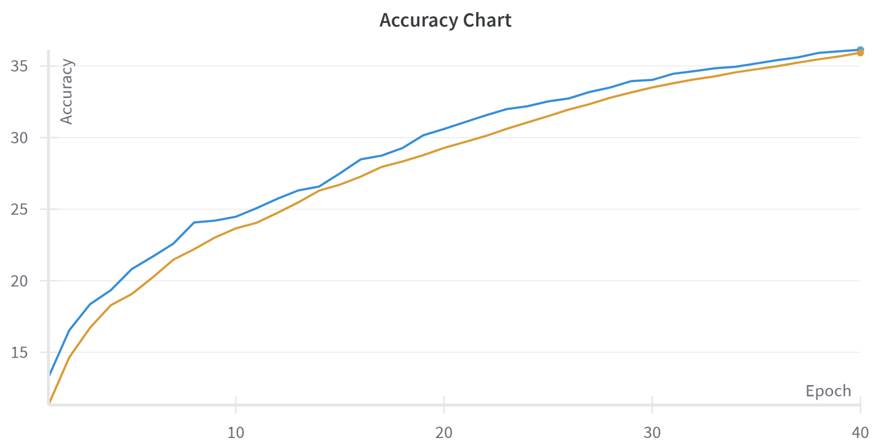

Figure 16: Q1a LR=0.01, Tanh, MSE Accuracy Plot

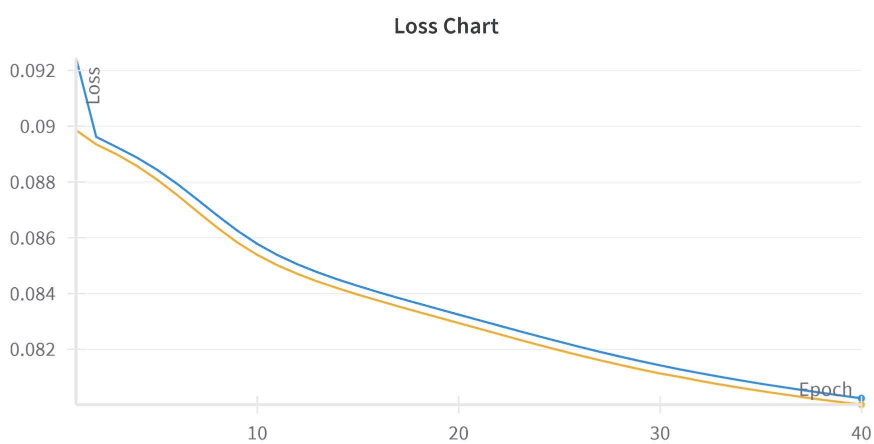

Figure 17: Q1a LR=0.01, Tanh, MSE Loss Plot

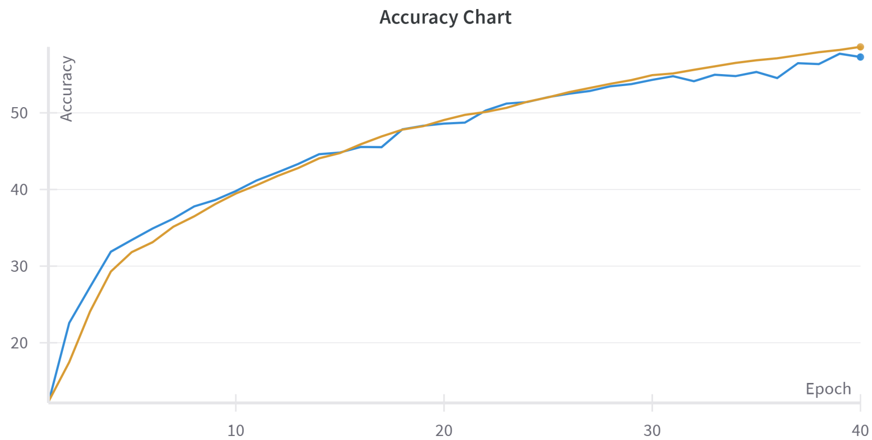

Figure 18: Q1a LR=0.01, Tanh, Cross Entropy Accuracy Plot

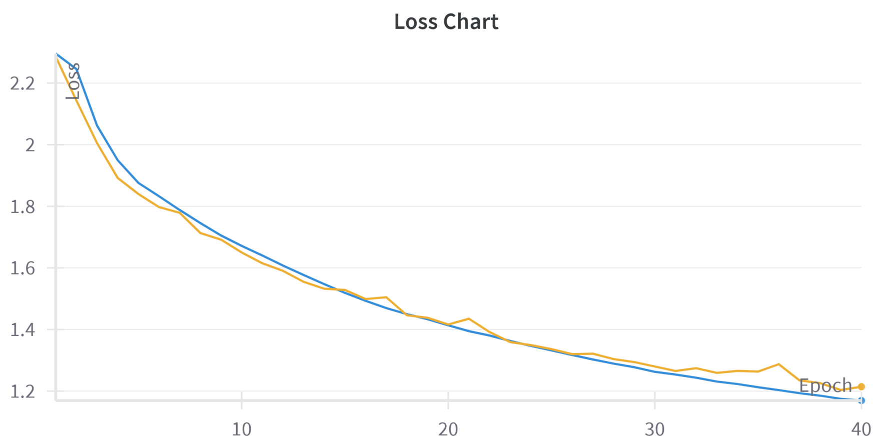

Figure 19: Q1a LR=0.01, Tanh, Cross Entropy Loss Plot

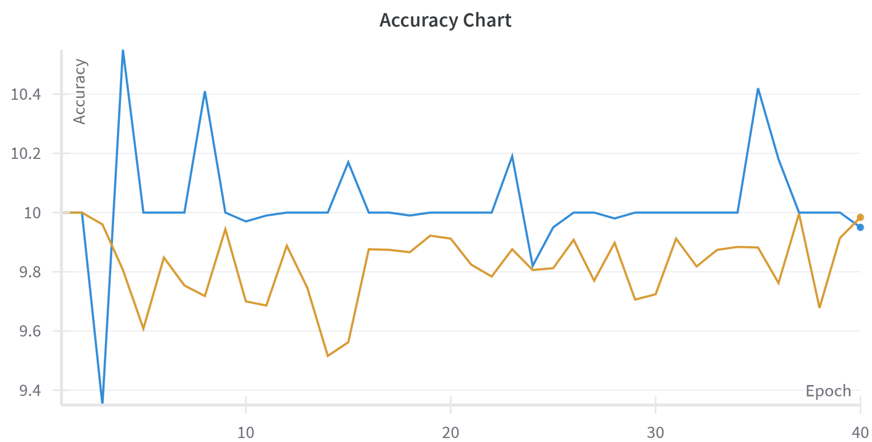

Figure 20: Q1a LR=0.001, Sigmoid, MSE Accuracy Plot

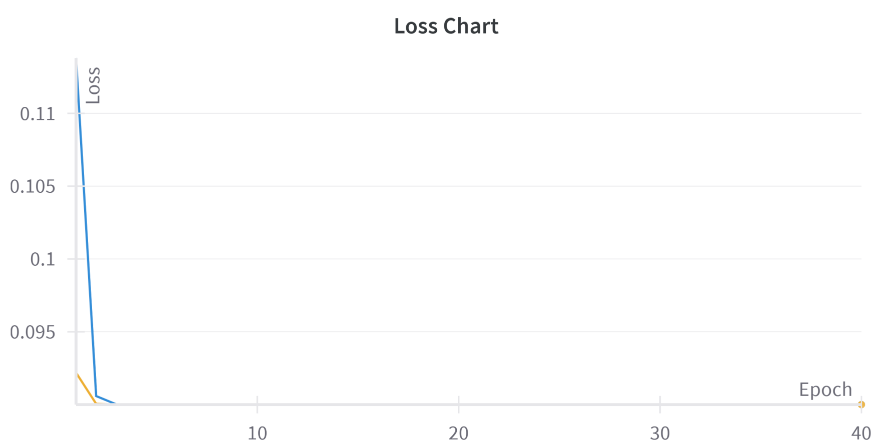

Figure 21: Q1a LR=0.001, Sigmoid, MSE Loss Plot

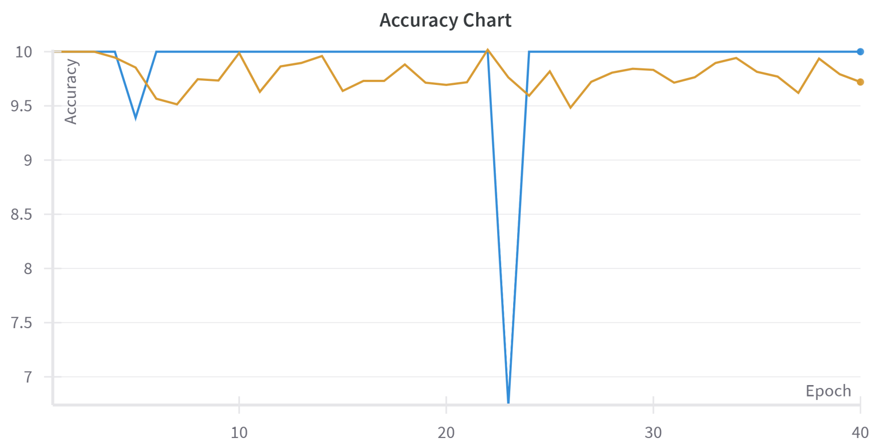

Figure 22: Q1a LR=0.001, Sigmoid, Cross Entropy Accuracy Plot

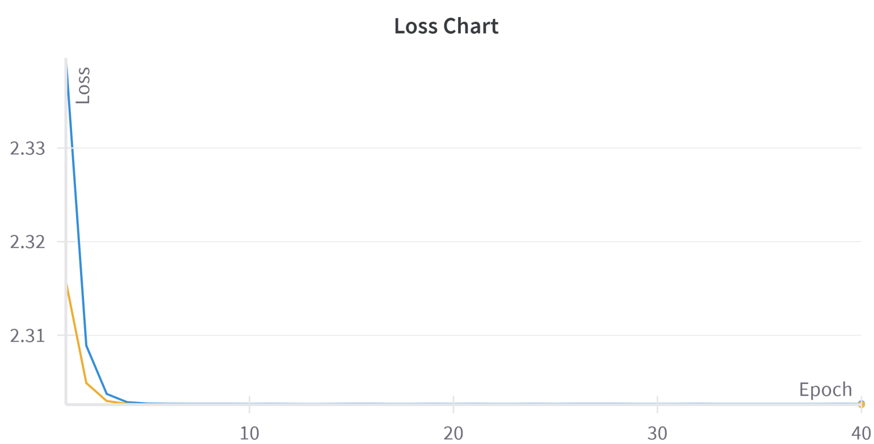

Figure 23: Q1a LR=0.001, Sigmoid, Cross Entropy Loss Plot

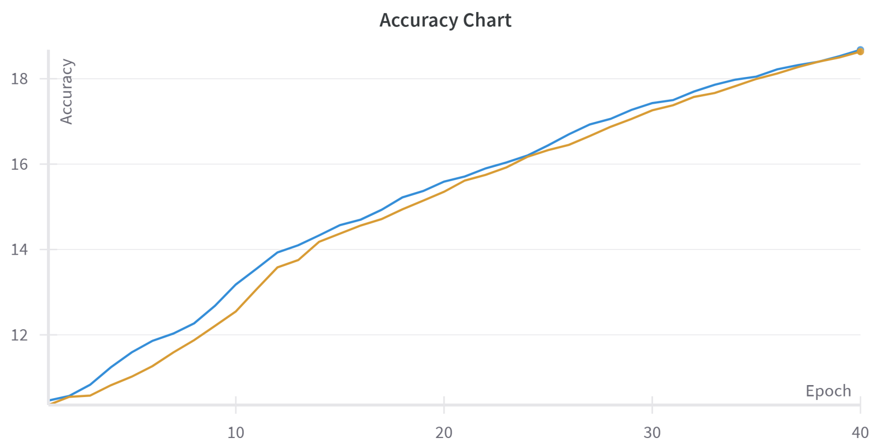

Figure 24: Q1a LR=0.001, Tanh, MSE Accuracy Plot

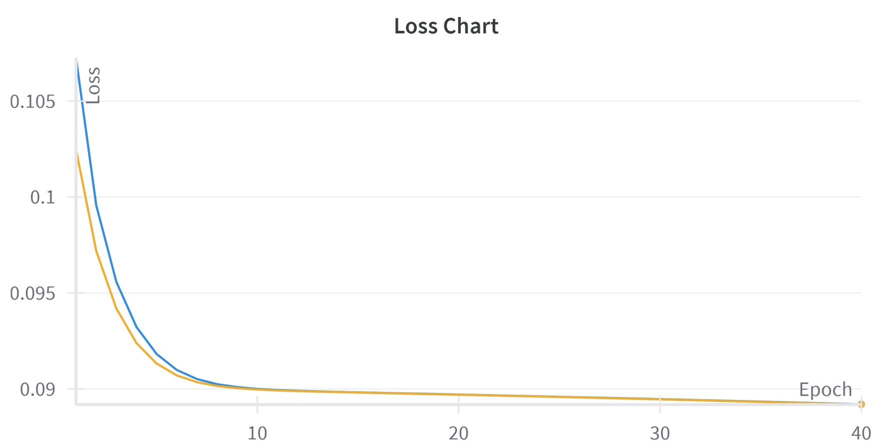

Figure 25: Q1a LR=0.001, Tanh, MSE Loss Plot

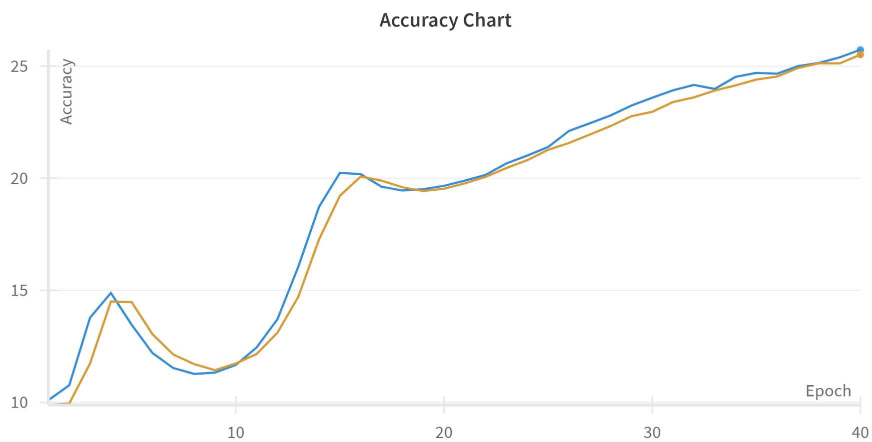

Figure 26: Q1a LR=0.001, Tanh, Cross Entropy Accuracy Plot

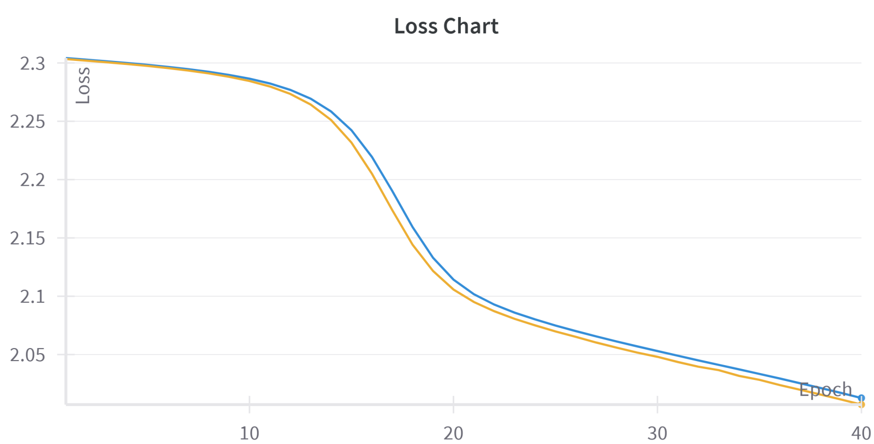

Figure 27: Q1a LR=0.001, Tanh, Cross Entropy Loss Plot

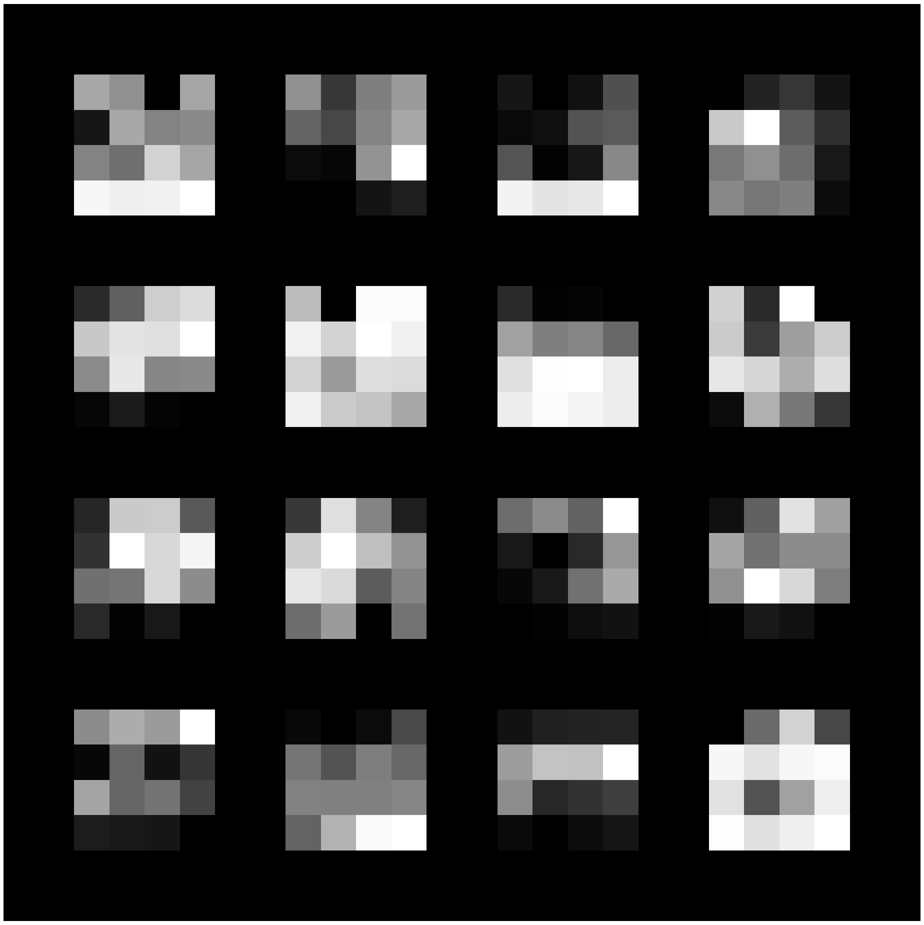

Figure 28: Q1b Feature Map for Image 1

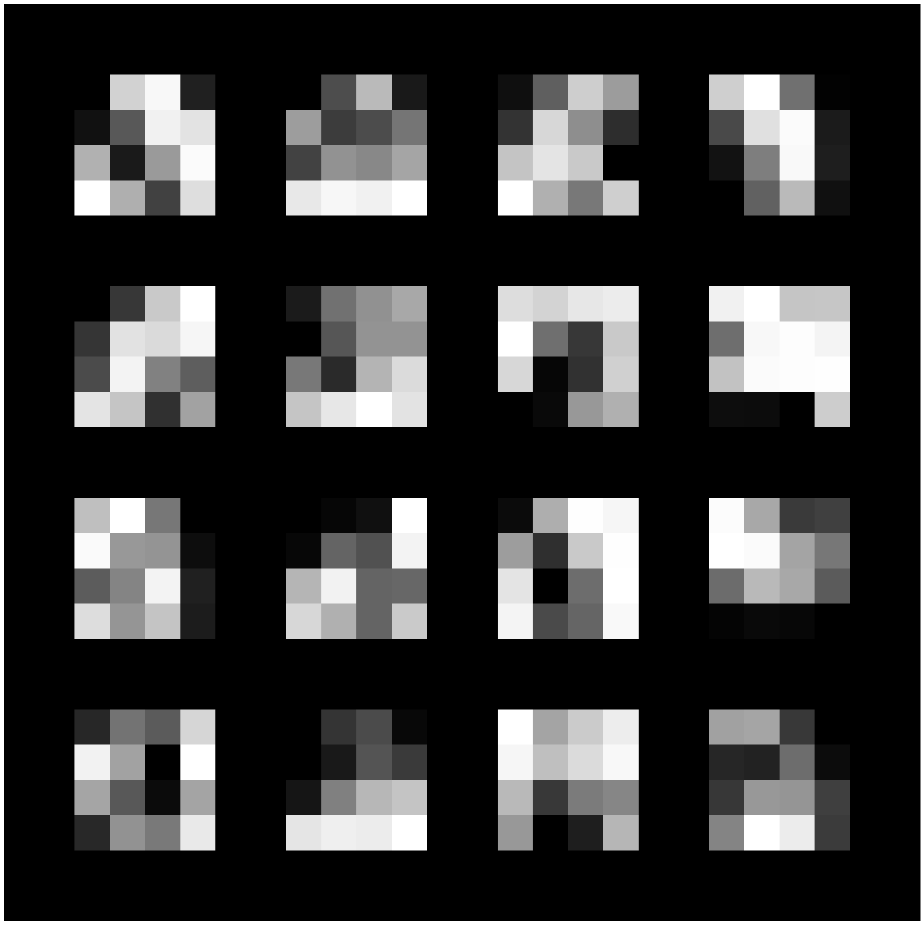

Figure 29: Q1b Feature Map for Image 2

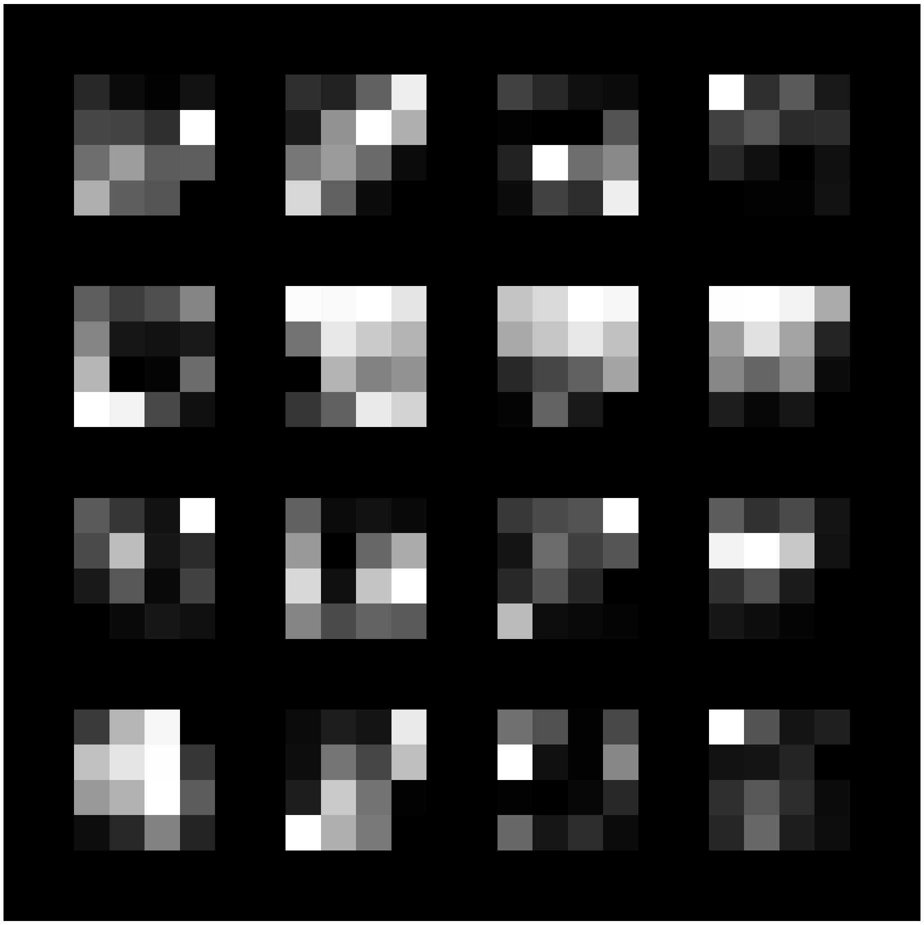

Figure 30: Q1b Feature Map for Image 3

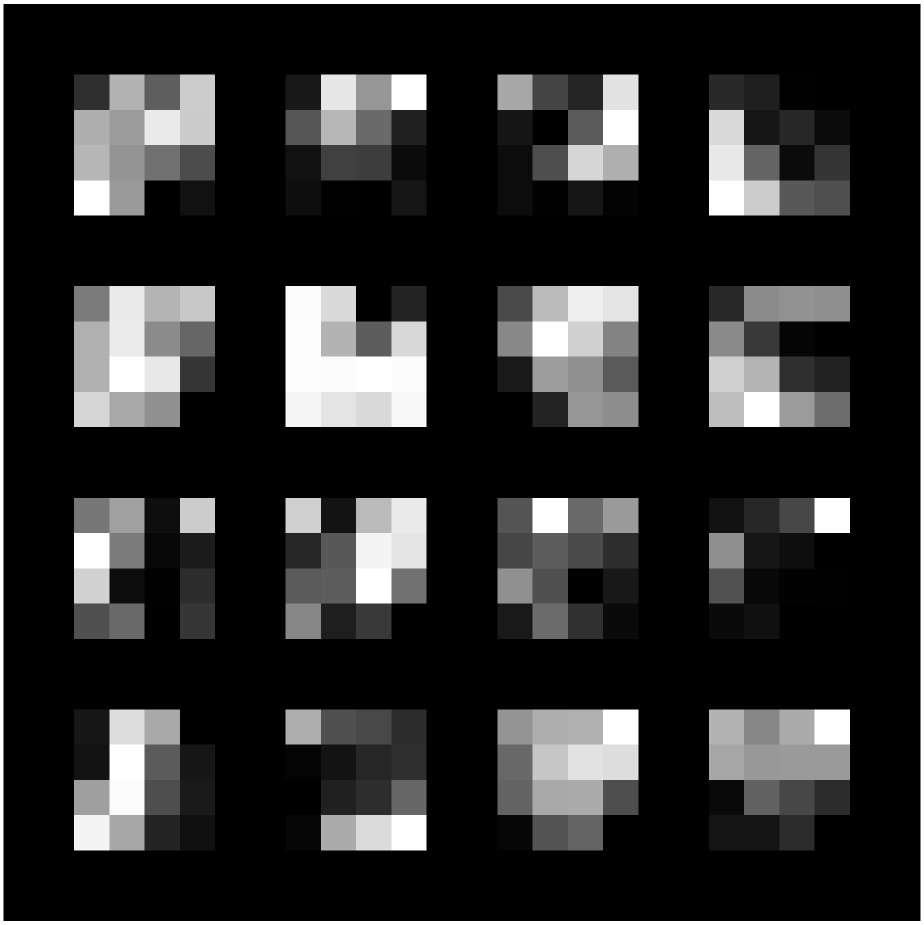

Figure 31: Q1b Feature Map for Image 4

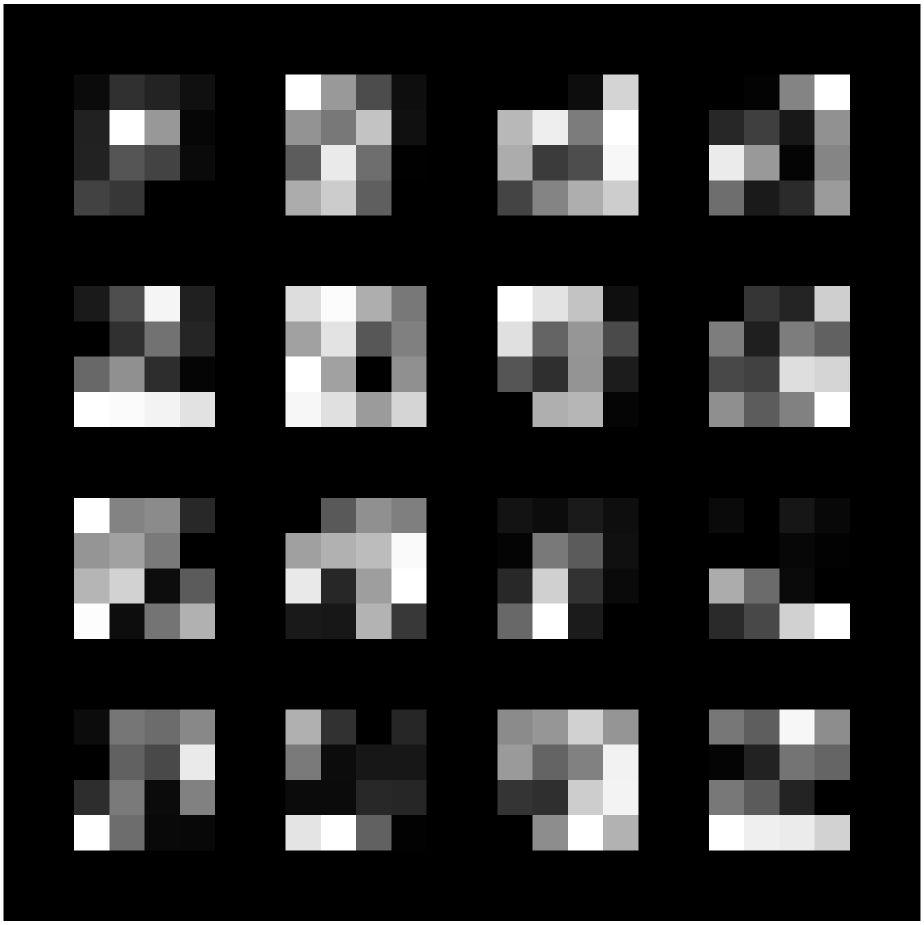

Figure 32: Q1b Feature Map for Image 5

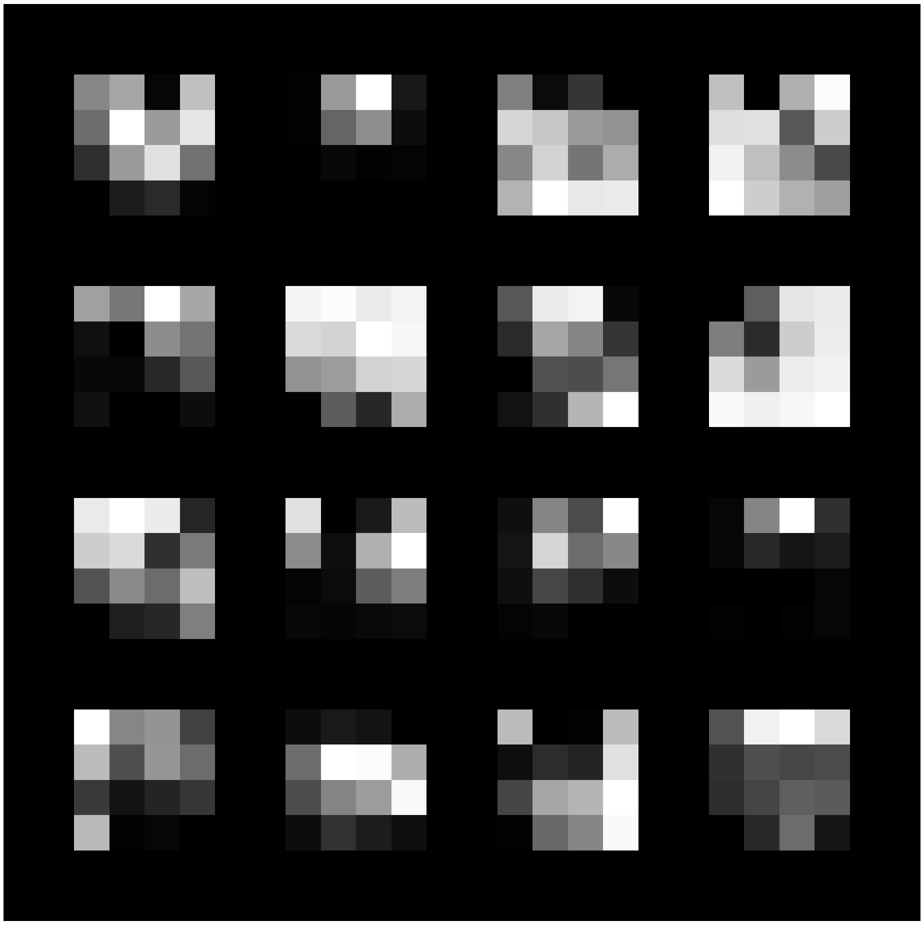

Figure 33: Q1b Feature Map for Image 6

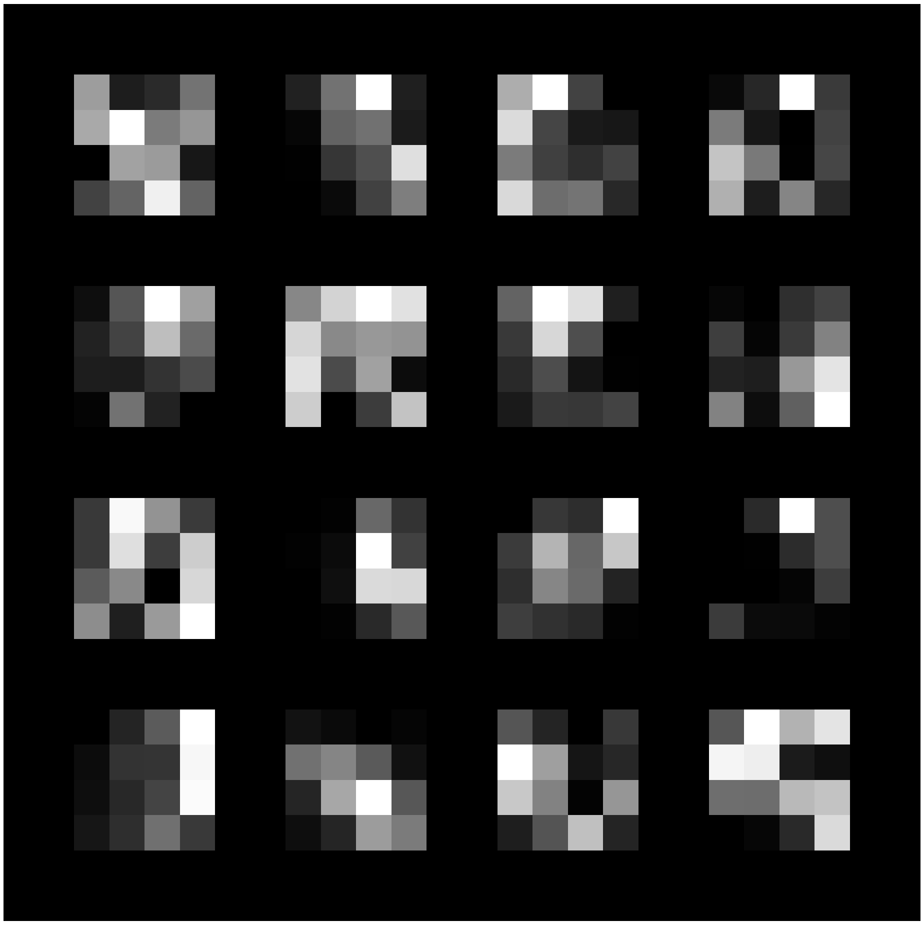

Figure 34: Q1b Feature Map for Image 7

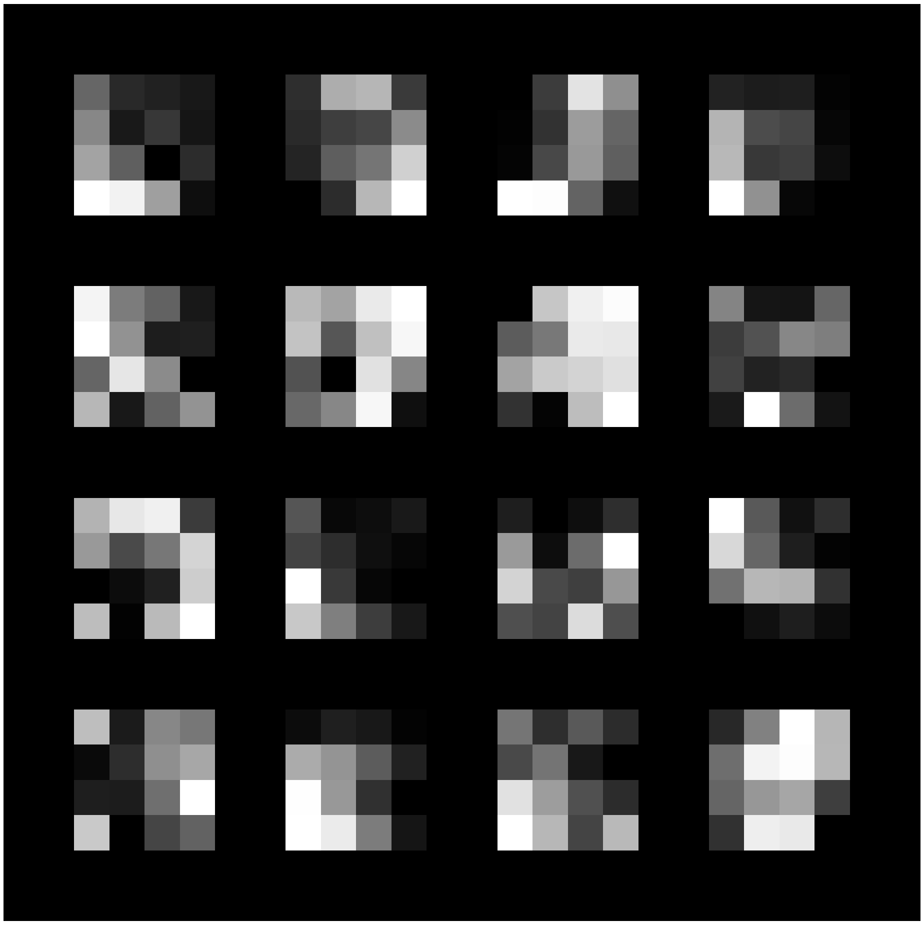

Figure 35: Q1b Feature Map for Image 8

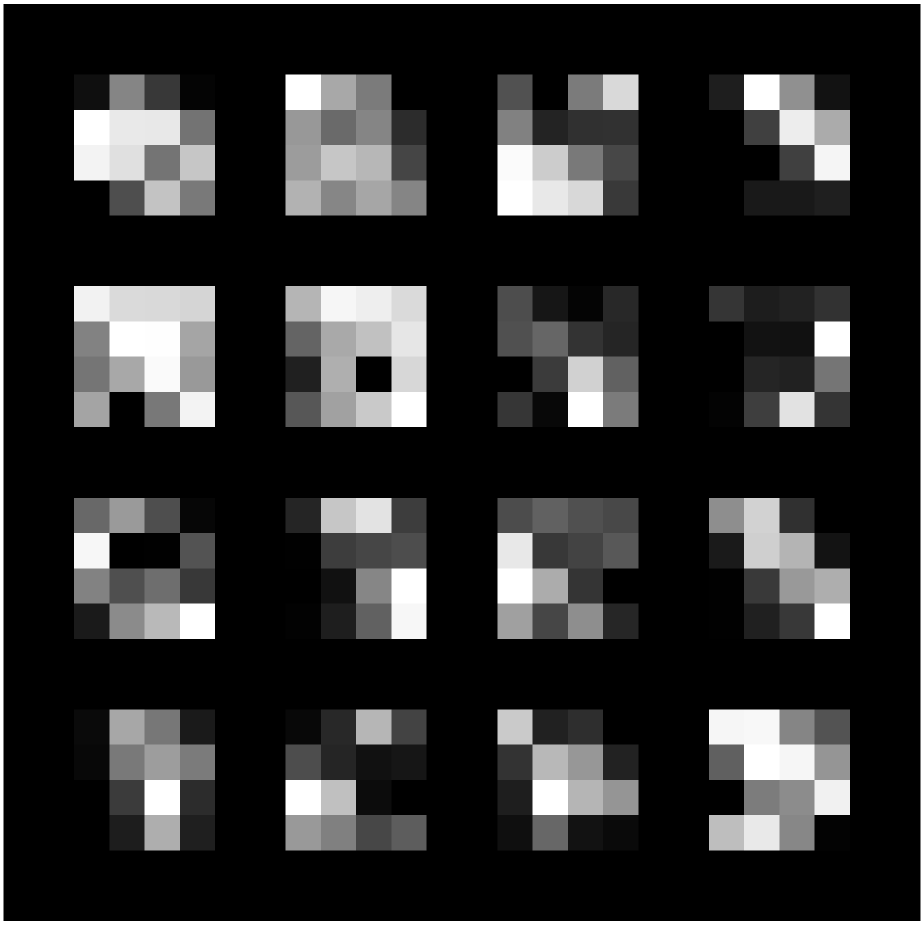

Figure 36: Q1b Feature Map for Image 9

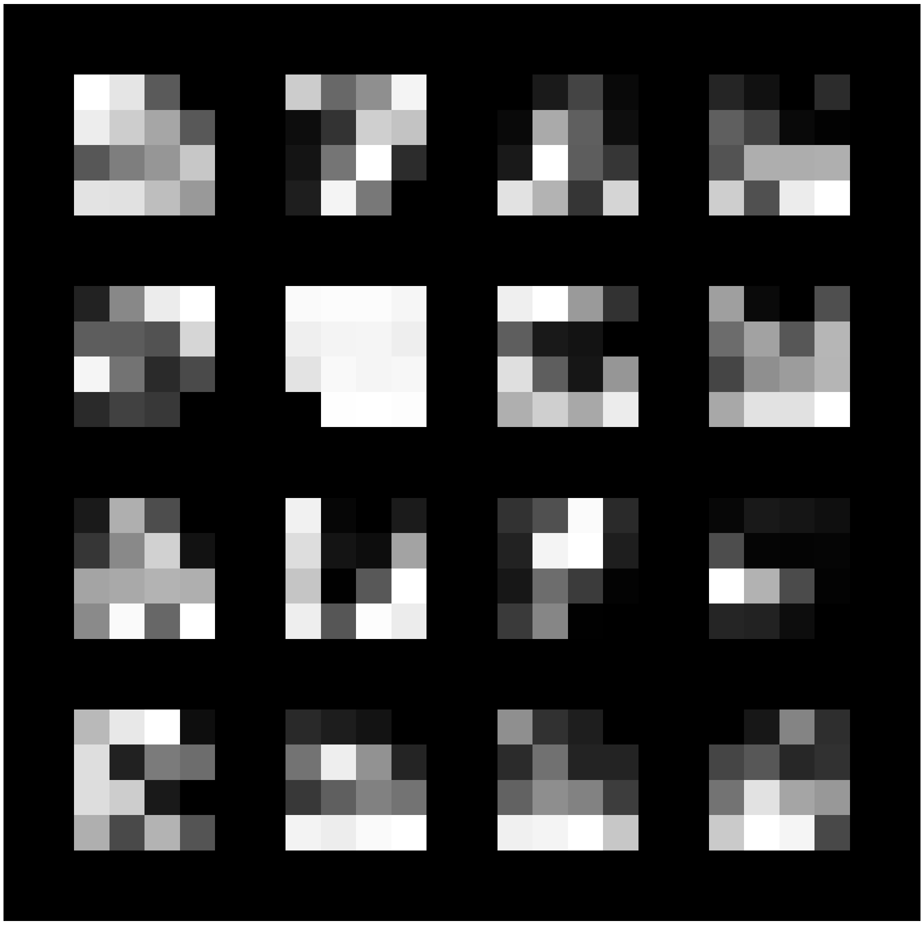

Figure 37: Q1b Feature Map for Image 10

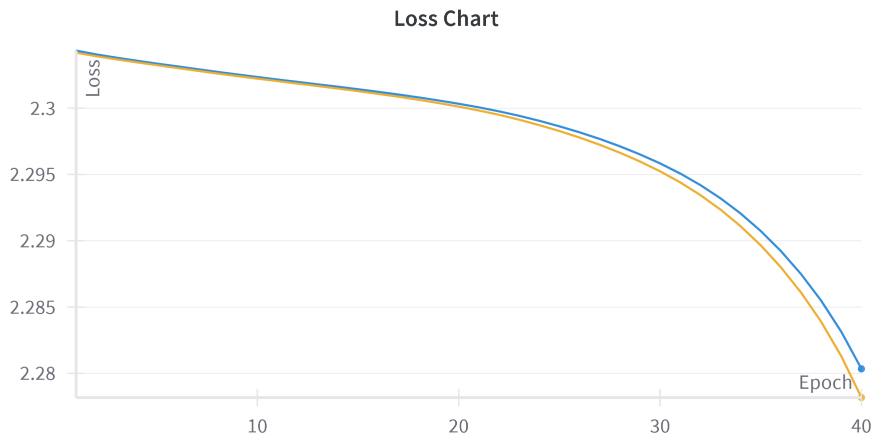

Figure 38: Q2 Loss with 0 Momentum

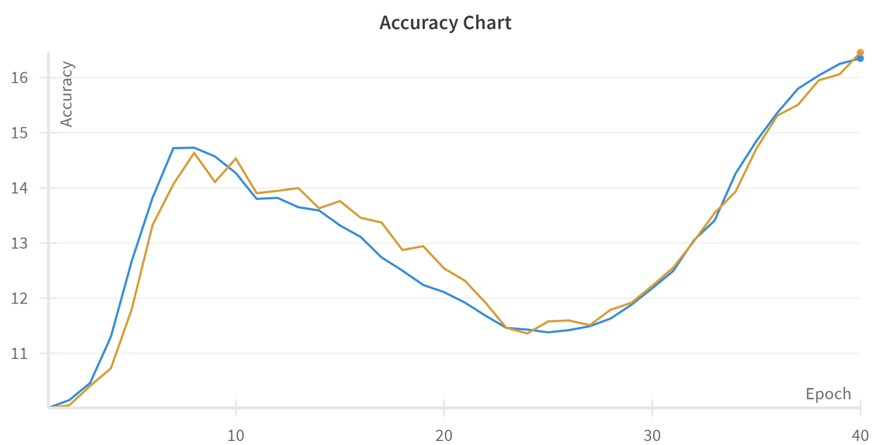

Figure 39: Q2 Accuracy with 0 Momentum

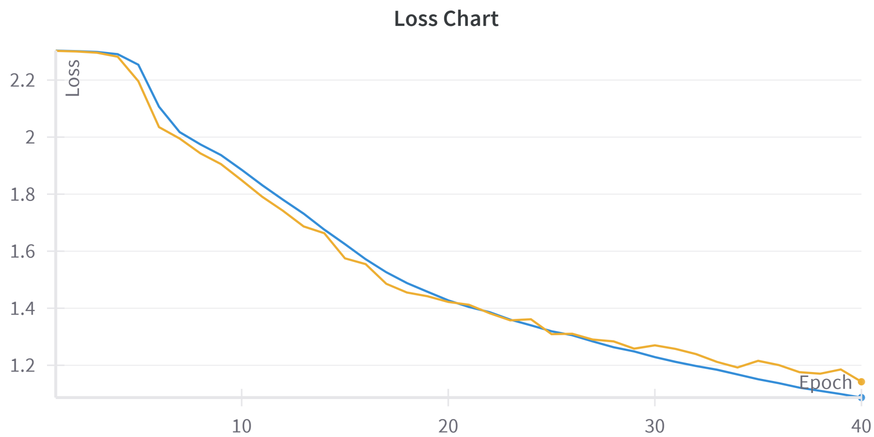

Figure 40: Q2 Loss with 0.9 Momentum

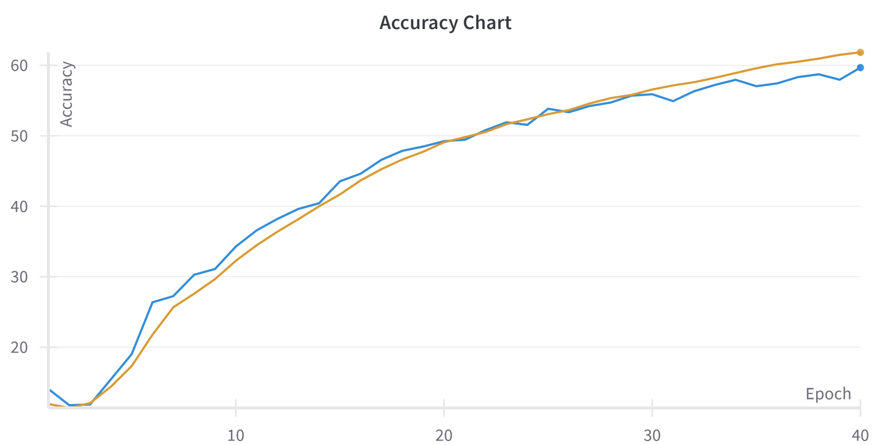

Figure 41: Q2 Accuracy

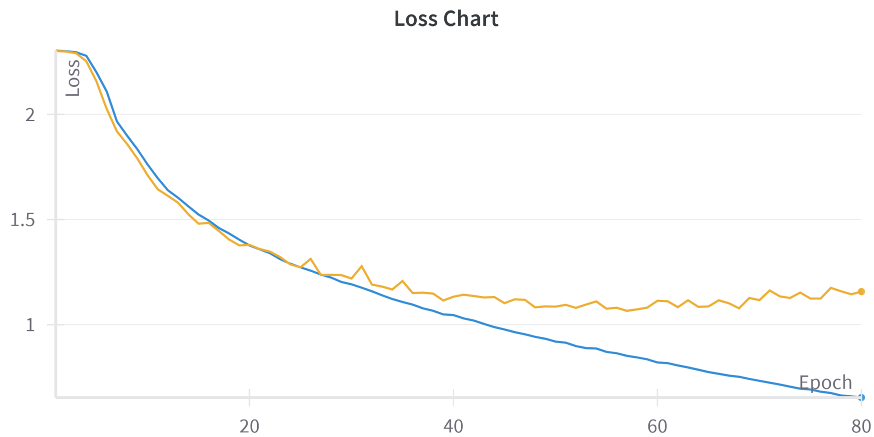

Figure 42: Q2 Loss with 0.9 Momentum, 80 Epoch

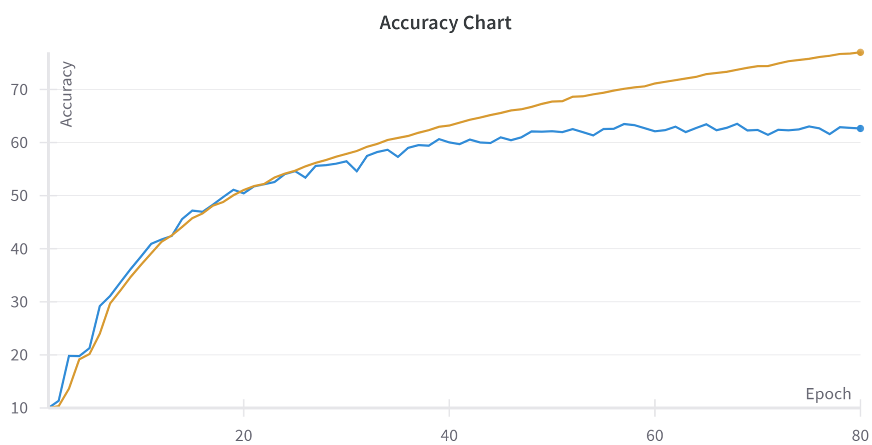

Figure 43: Q2 Accuracy