A1 Report - Kevin Armbruster

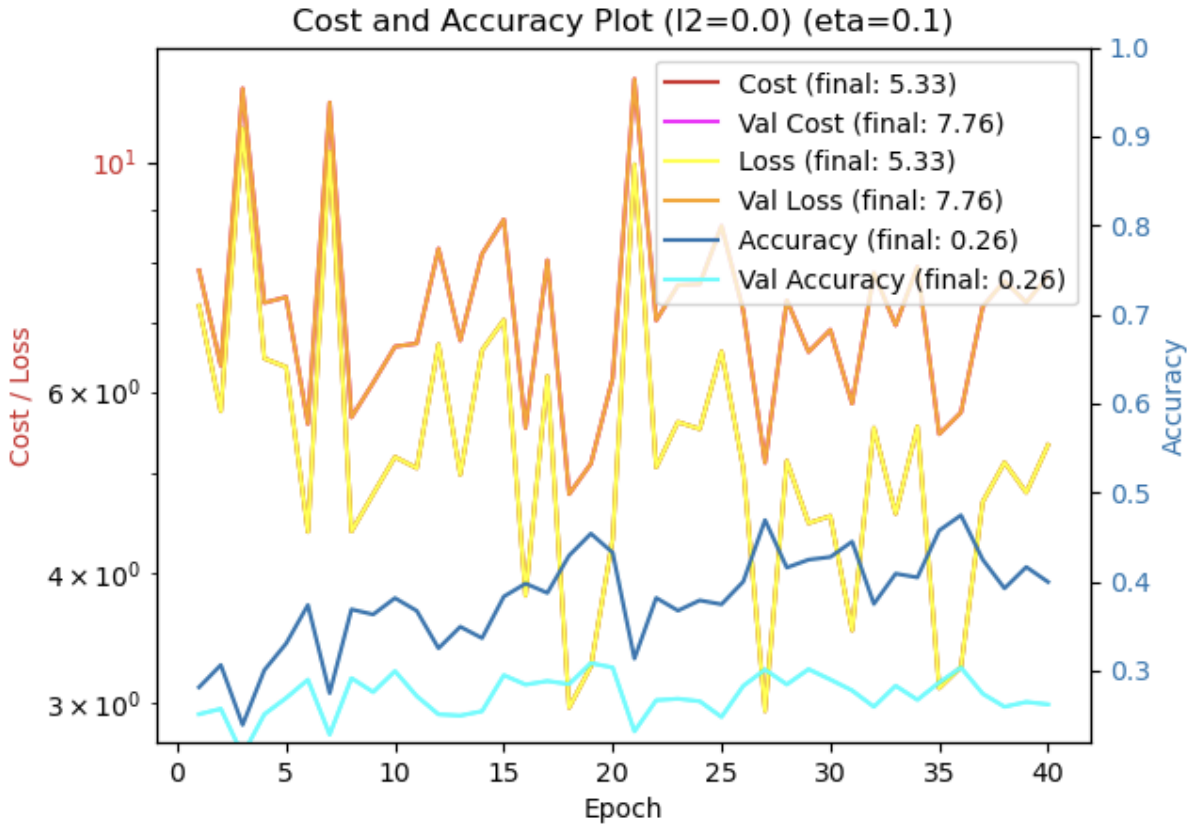
I **successfully implemented** the functions as was required. I compared the numerical and analytical solutions with np.allclose and set the absolute tolerance to be 1e-4, although mostly 1e-7 worked, but not always. I observed bigger divergences with the regularization turned on.

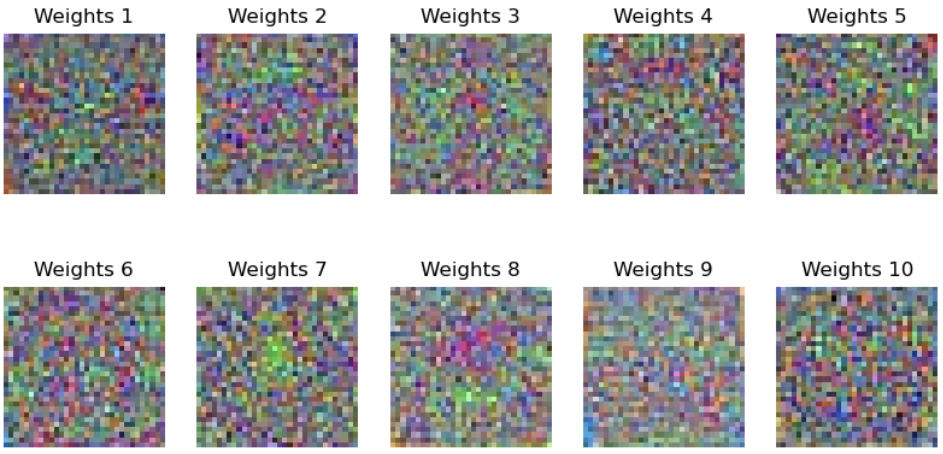
The **importance of learning rate** can be easily seen if one compares plot 1 and 2. In 1 we see a very unstable jigsaw pattern, although in general it seems to improve. Hence, it seems not to be completely unstable, but it's likely we don't converge (to the local minima). While in comparison plot 2 exhibits a smooth learning behavior and could converge to a local minima without "jumping" out of it / around it.

Regularization reduces the effects of overfitting. Since we used a very small model we are underfitting and hence observed no immediate effects comparing the validation and test scores between plots 2 and 3.
Between plots 3 and 4 we observe that too strong regularization diminishes the network performance by about 0.02 in validation and test accuracy. Additionally we observe the cost curves to increase because as the network learns the parameters got bigger and it increased the value.

1. lambda=0, n epochs=40, n batch=100, eta=.1

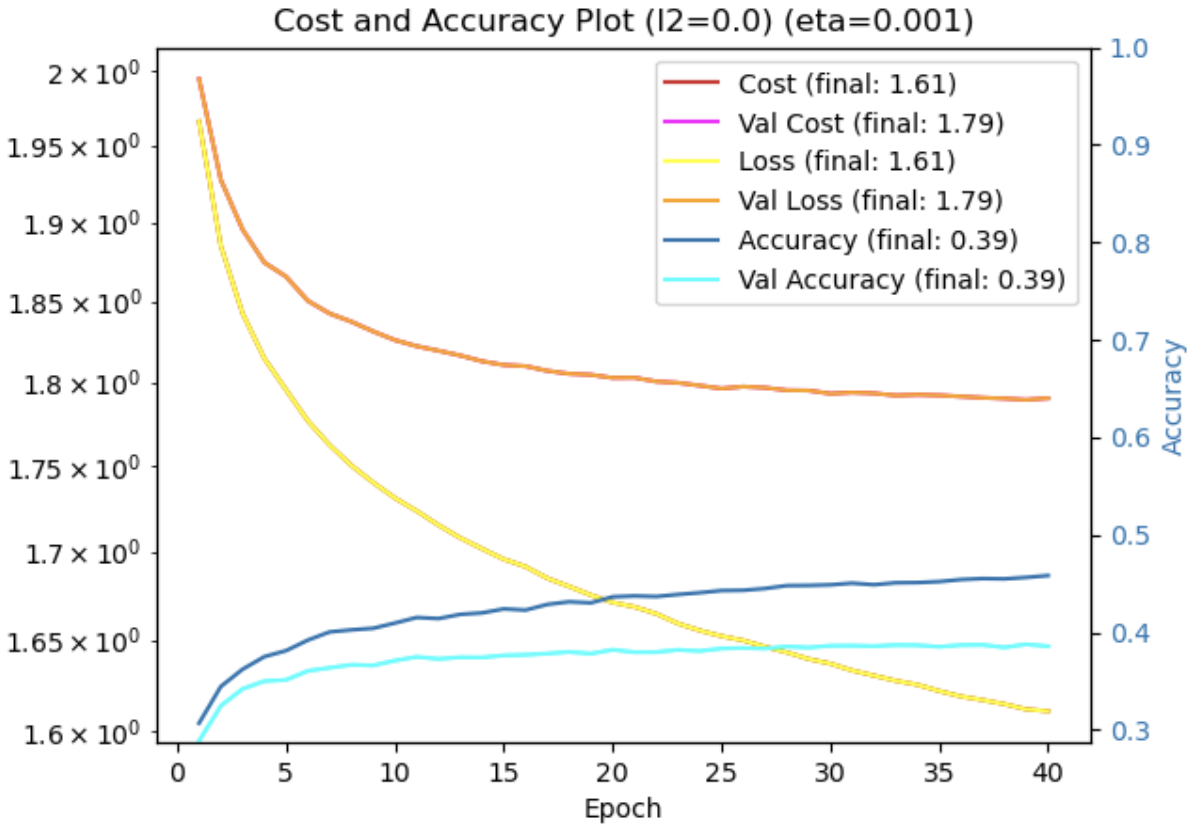
Final test accuracy 0.277

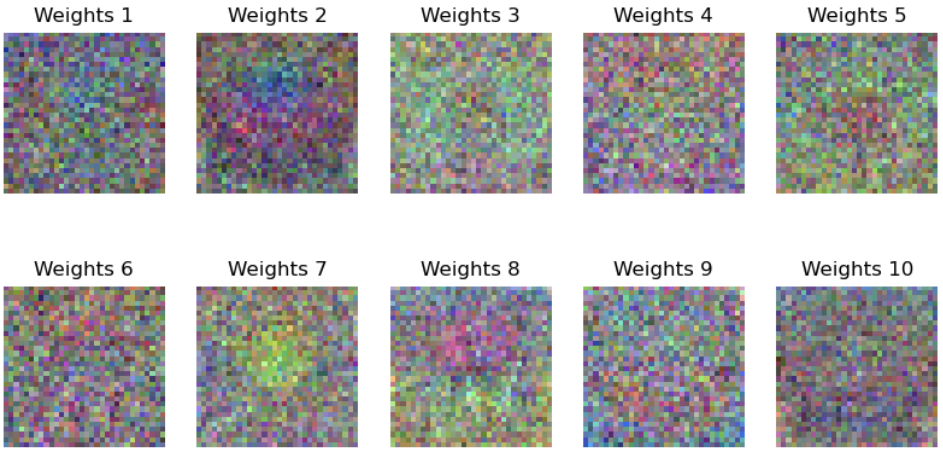




2. lambda=0, n epochs=40, n batch=100, eta=.001

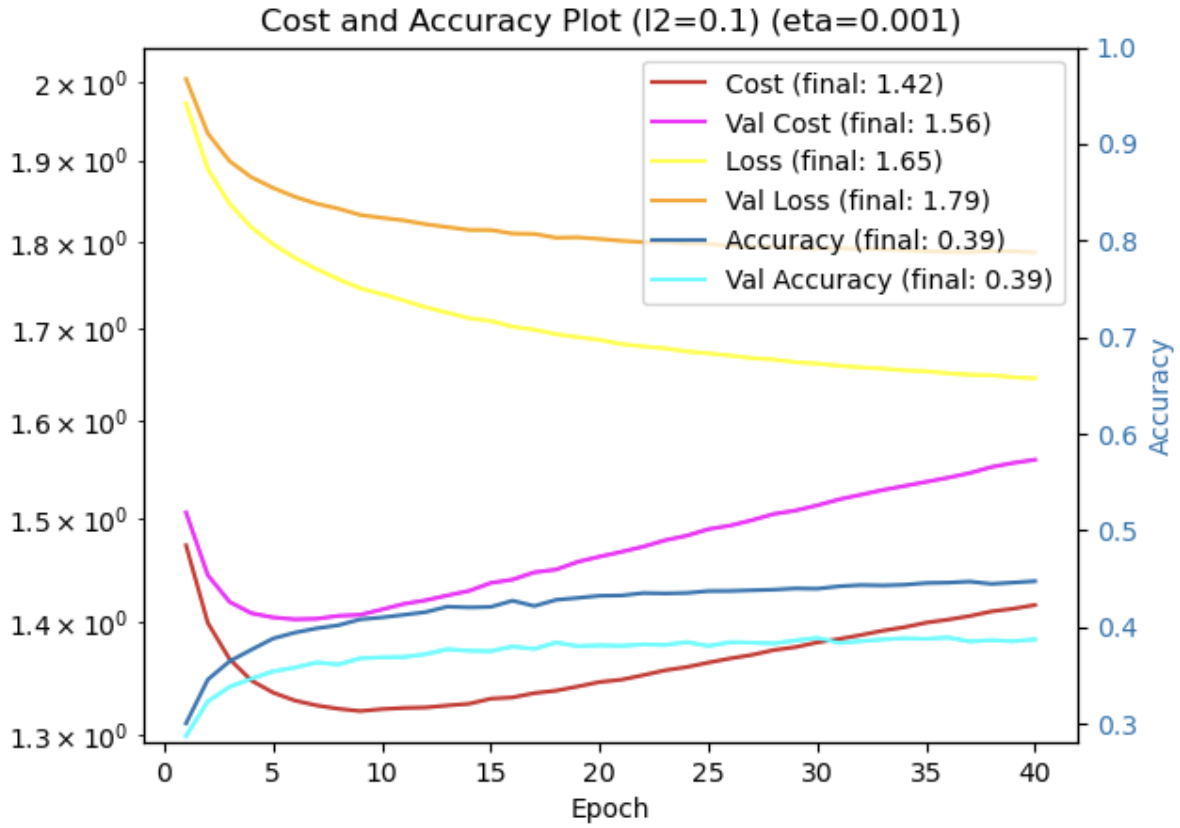
Final test accuracy 0.3936

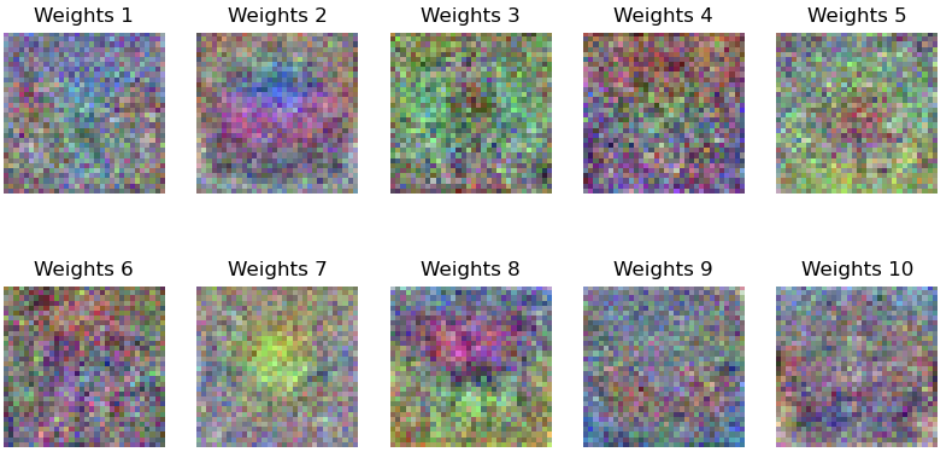




3. lambda=.1, n epochs=40, n batch=100, eta=.001

Final test accuracy 0.3911





4. lambda=1, n epochs=40, n batch=100, eta=.001

Final test accuracy 0.3773