**Reference**

1. Stanford class notes (CS231n Convolutional Neural Networks for Visual Recognition)

<https://cs231n.github.io/neural-networks-2/#datapre>

1. Hochreiter, S. (1998). The vanishing gradient problem during learning recurrent neural nets and problem solutions. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 6(2), 107-116.

https://doi.org/10.1142/S0218488598000094

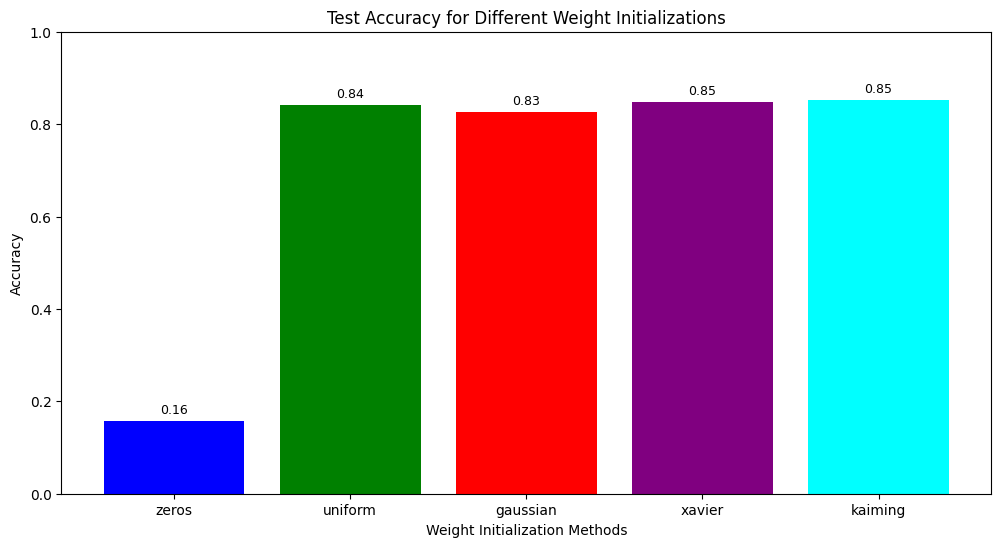
1. Lu, L., Shin, Y., Su, Y., & Karniadakis, G. E. (2020). Dying ReLU and Initialization: Theory and Numerical Examples. arXiv preprint arXiv:1903.06733. https://arxiv.org/abs/1903.06733

**Appendix**

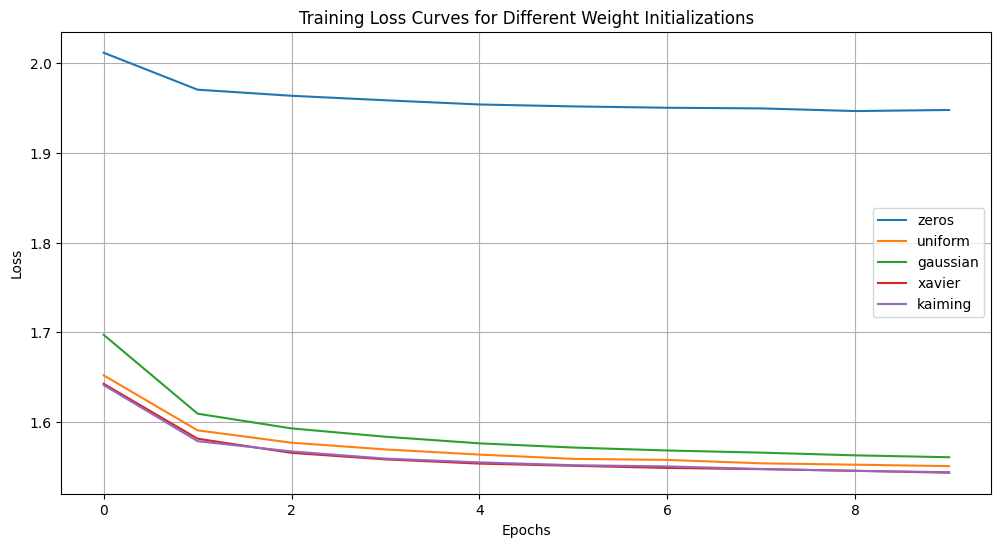
1. **Multilayer Perceptron (MLP)**
2. **Different Weight Initialization For Each Activation**

The report has shown the performance of different weight initializations in ReLu Activation. For a more comprehensive analysis, we have also analyzed the influence of the initialization methods in Sigmoid and Softplus respectively.

Sigmoid Activations: Except for Zeros, all the weight initializations perform well.

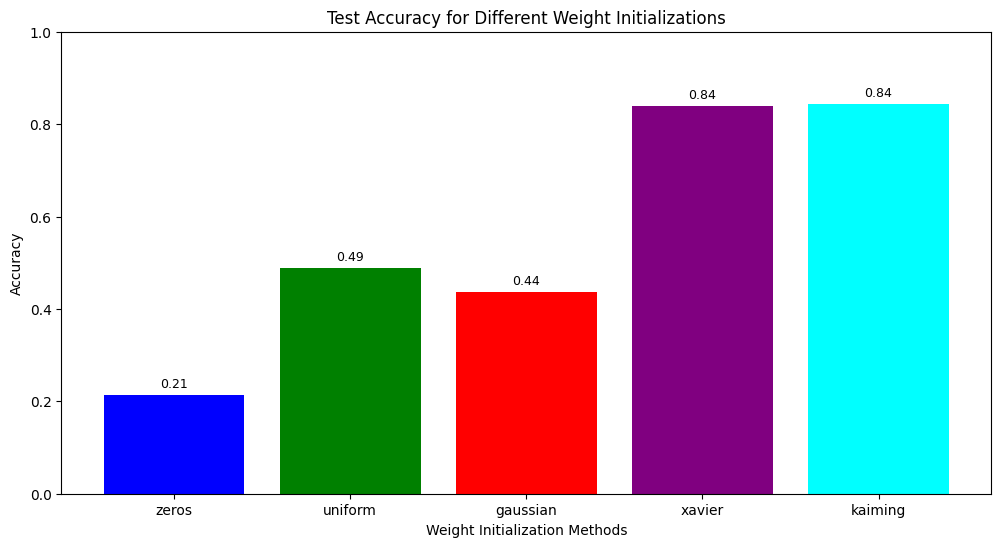


(Figure A1.1 The test accuracy in Sigmoid with different weight methods)

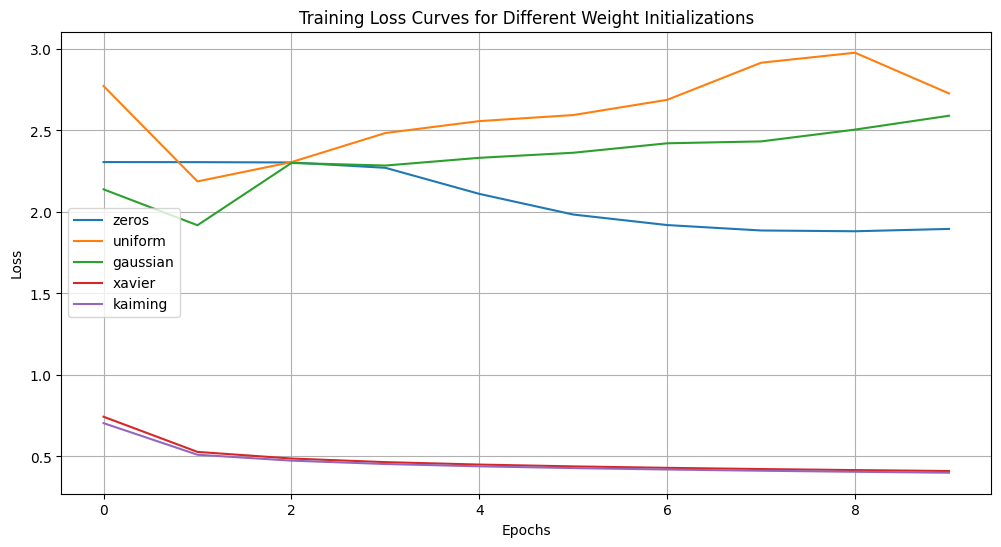


(Figure A1.2 The loss curves in Sigmoid with different weight methods)

SoftPlus Activations: The zeros, uniform, and Gaussian methods show inconsistent performance, while the Xavier and Kaiming are presenting pretty well and consistent results in test accuracy.



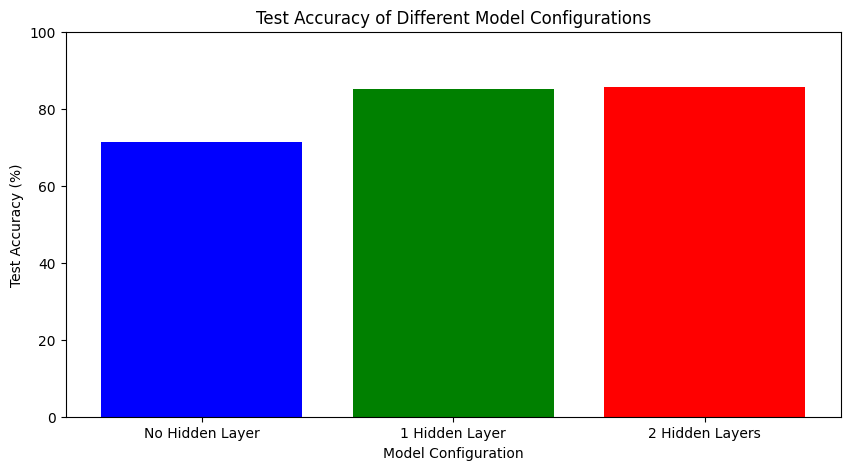
(Figure A1.3 The Test Accuracy in SoftPlus with different weight methods)



(Figure A1.4 The Loss Curves in SoftPlus with different weight methods)

1. **Difference of Different Layer Structures in ReLu**

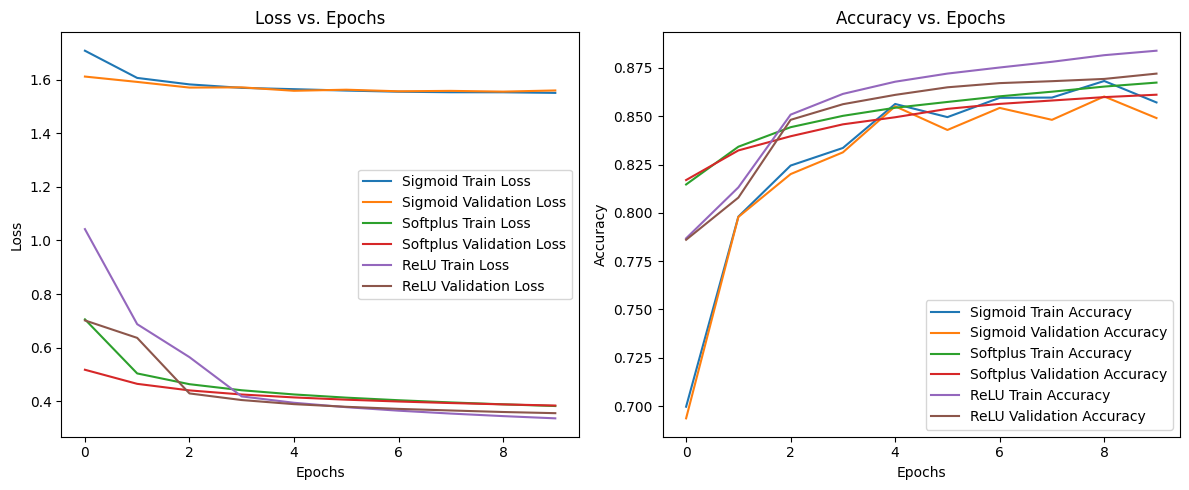
Additional information showing the test accuracy of different layer settings. The No-Hidden Layer is worse than the Having-Hidden Layers. However, it seems the network depth did not play an importance here.



(Figure A2. The Loss Curves in SoftPlus with different weight methods)

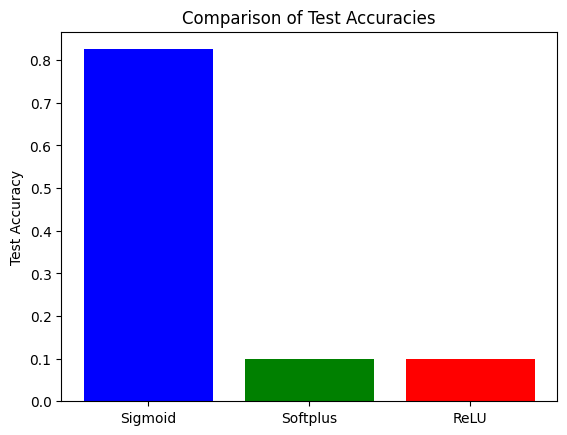
1. **Performance of Different Activation Functions**

Additional informations are shown here. The following cases show the loss curve and accuracy performance when sigmoid is using relatively high learning rate which is 0.01, while the relu and softplus are using relatively low learning rate which is 0.0001. The outputs are reasonable showing the expected trend.

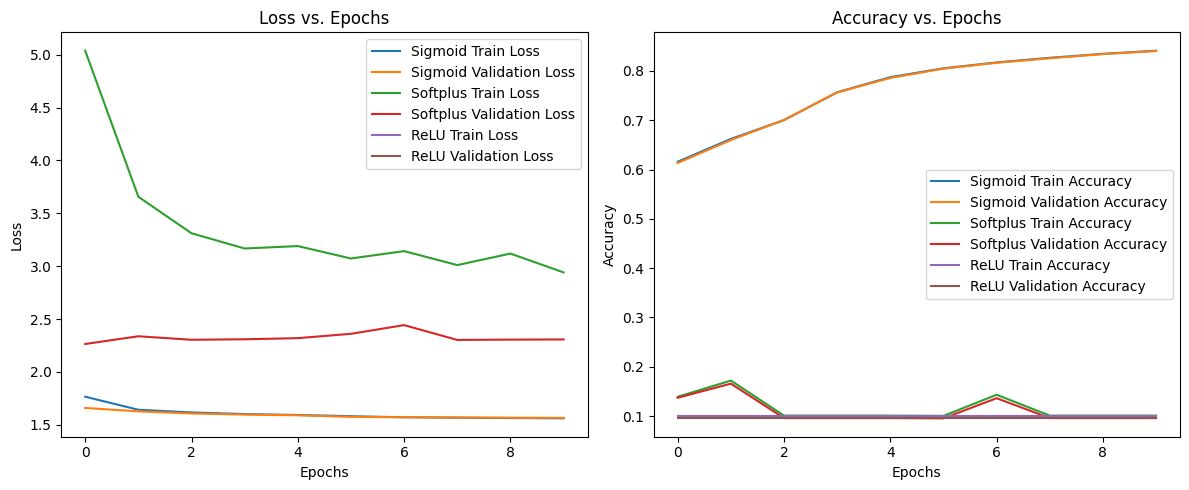
****

(Figure A3.1 The Loss and Learning Curves of different activations with valid LR)

The below two graphs show when the learning rates are inappropriately selected, the performance of all activations is hugely different. Because Softplus and ReLU are sensitive to learning rate. Using the same learning rate of 0.01 for the three models shows the worse performance like below.

****

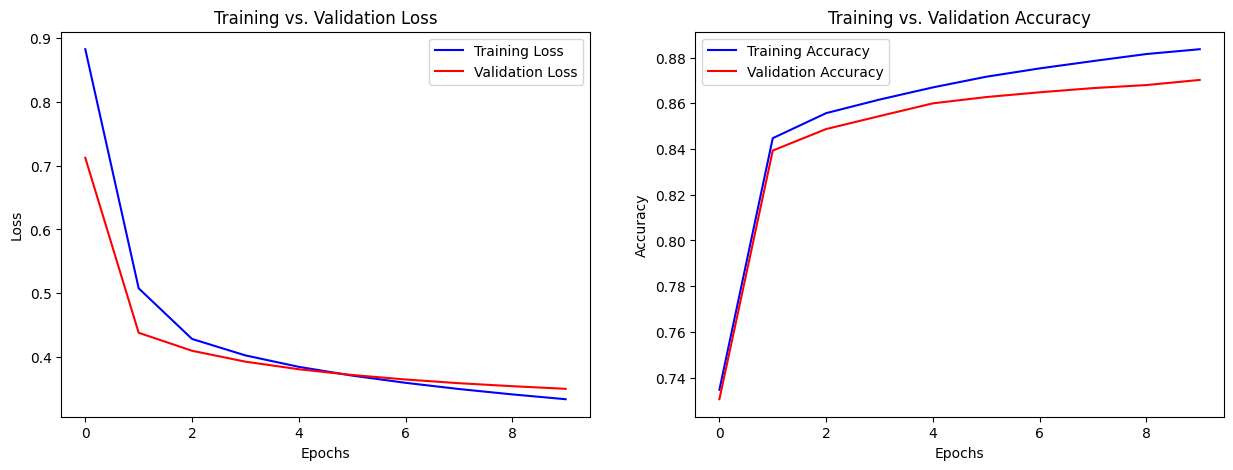
(Figure A3.2 The Test Accuracy in different activations with inappropriate LR)

****

(Figure A3.3 The Loss and Learning Curves of different activations with inappropriate LR)

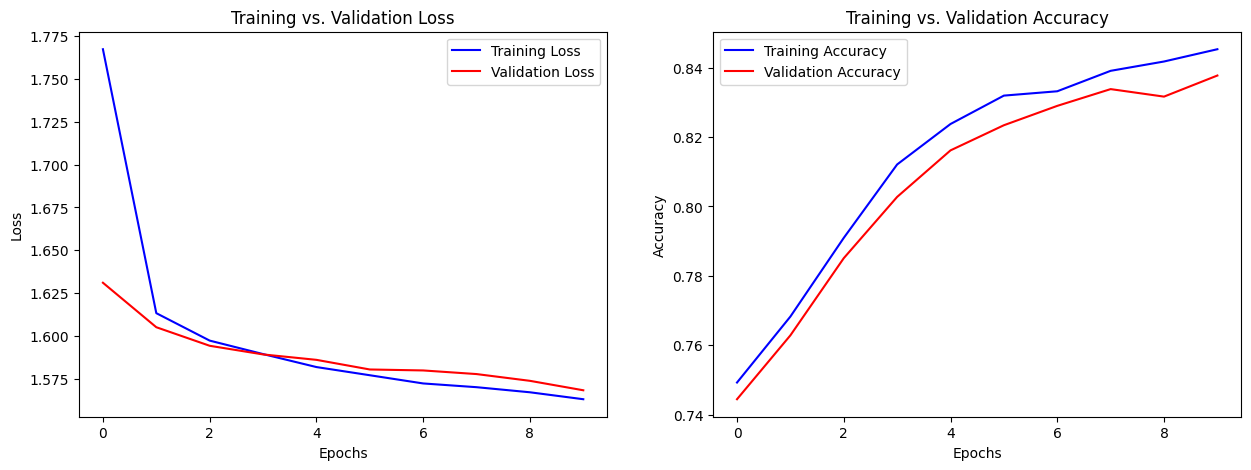
1. **Best Learning Curves of Each Activation**

**Relu**

****

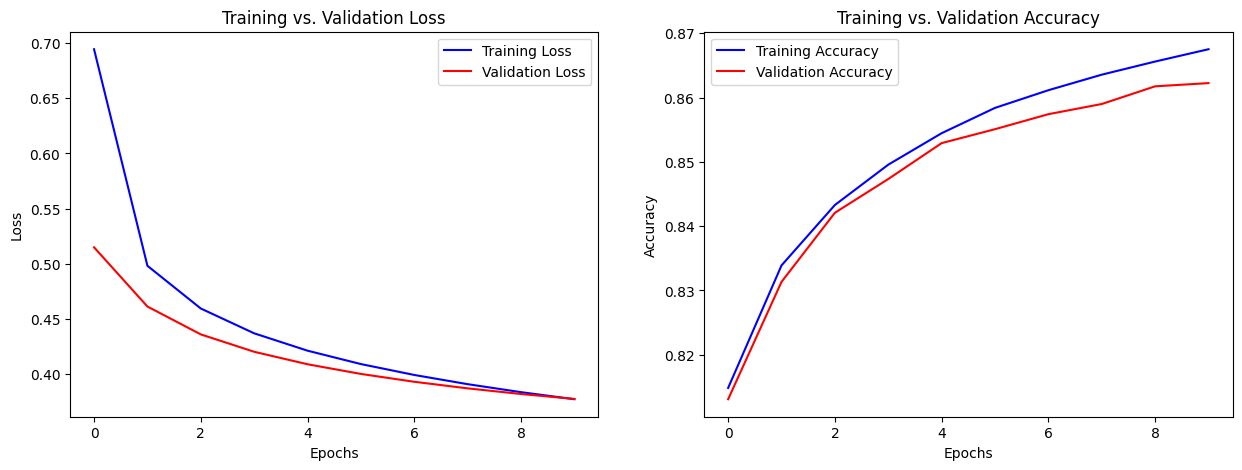
(Figure A4.1 The loss and learning curve of Relu using best parameters)

**Sigmoid**

****

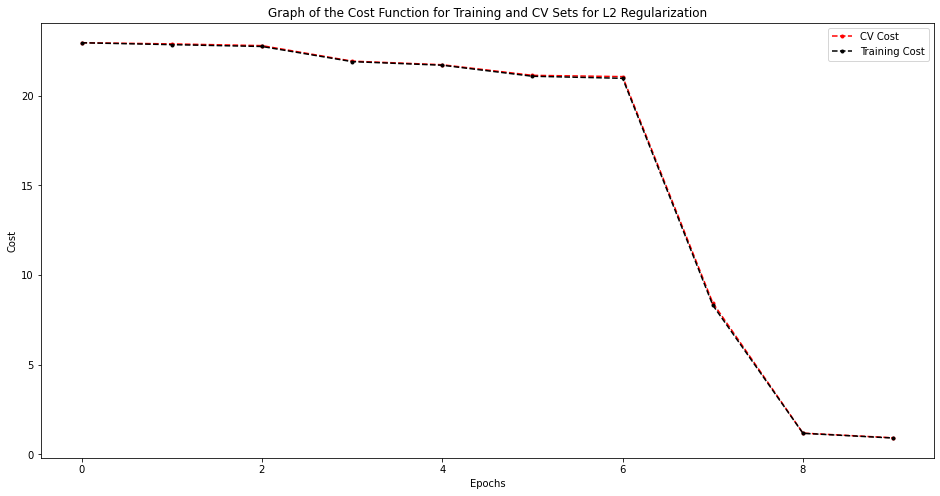
(Figure A4.2 The loss and learning curve of Sigmoid using best parameters)

**SoftPlus**

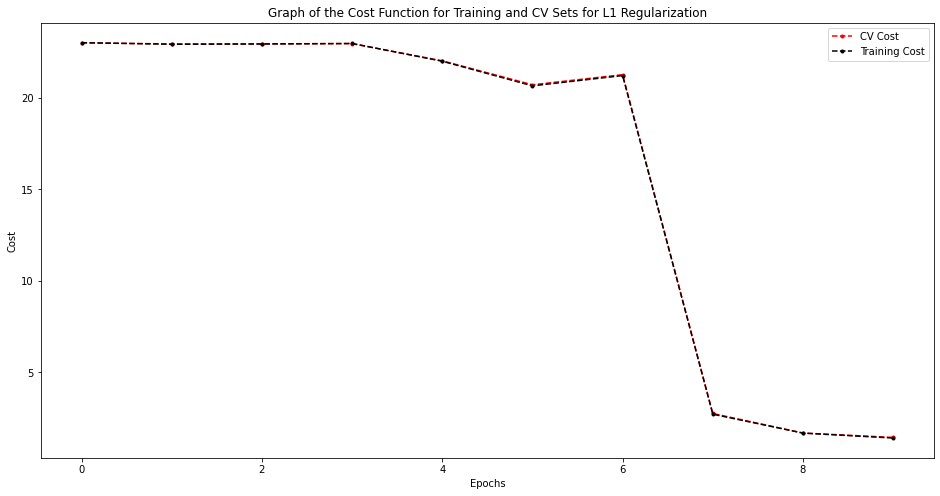
****

(Figure A4.3 The loss and learning curve of SoftPlus using best parameters)

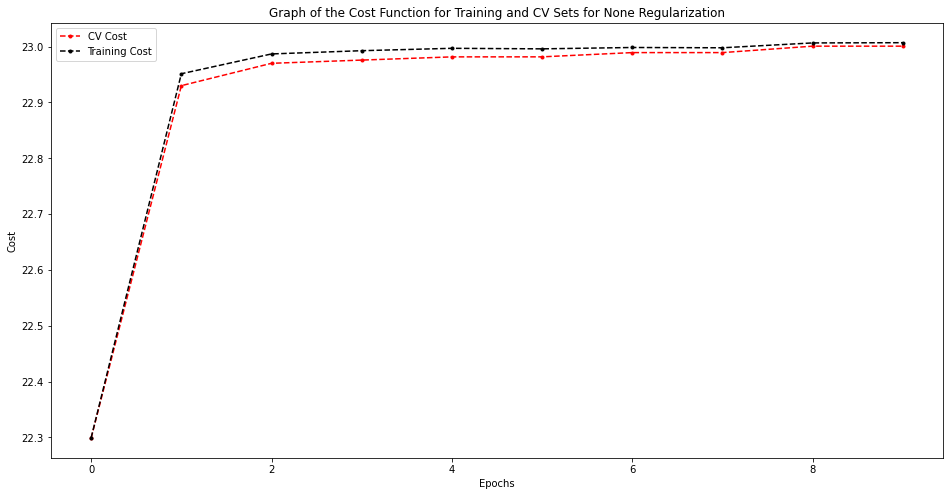
4. Effects of regularization



(Figure A4. 1 Mean loss curves for L2 regularization. 0.806 Accuracy. ReLU activation)

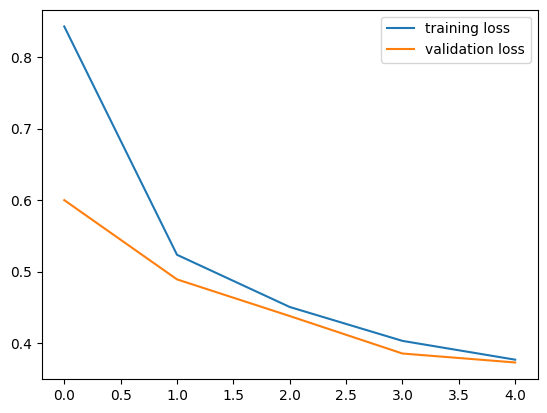
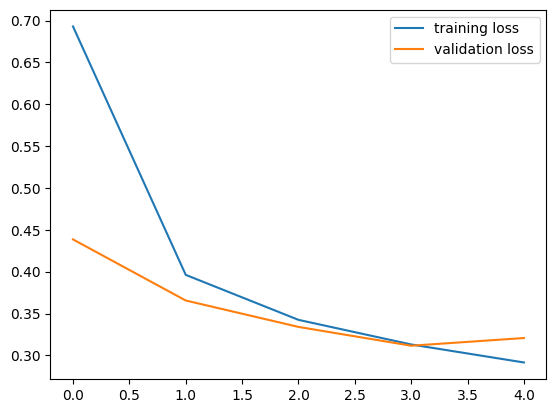


(Figure A4. 2 Mean loss curves for L1 regularization. 0.717 Accuracy. ReLU activation)

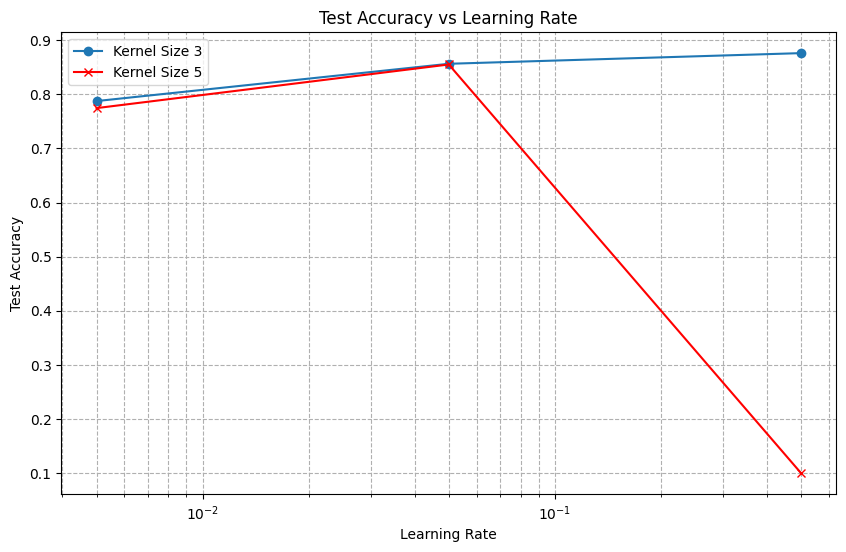


(Figure A4. 3 Mean loss curves for no regularization. 0.101 Accuracy. ReLU activation)

1. **CNN**

****

(Figure B1.1 Loss Curves for CNN trained on Fashion MNIST for differing learning rates. LR = 0.5 (left) and LR = 0.05 (right))

****

(Figure B1.2 Testing different hyperparameters against test accuracy. We choose the most appropriate learning rate and kernel size from this)

**B2. CNN vs MLP on CIFAR-10**

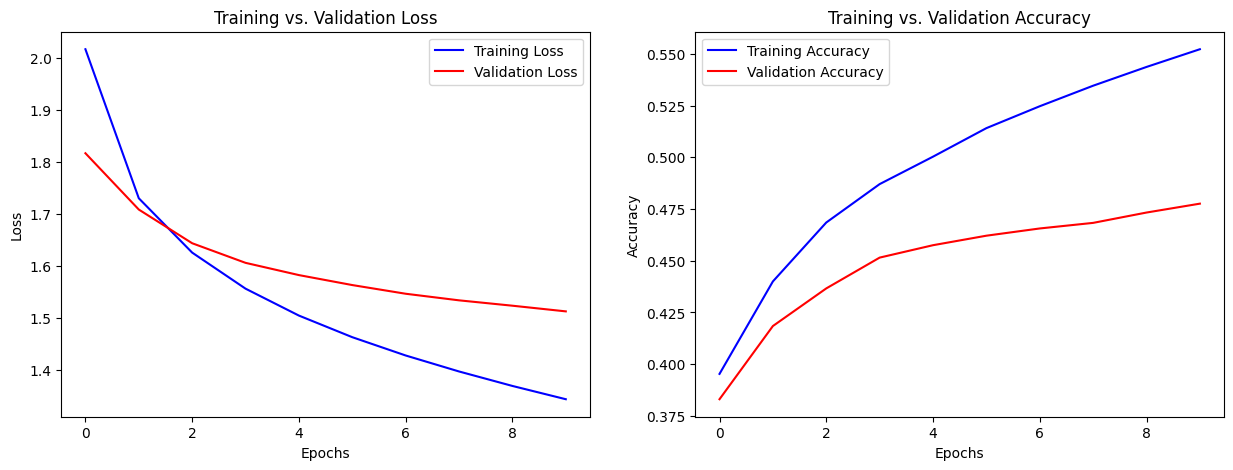


Figure B2.1 Loss curves and training + CV accuracies graphs of MLP

**B3. SGD vs Adam Optimizations**

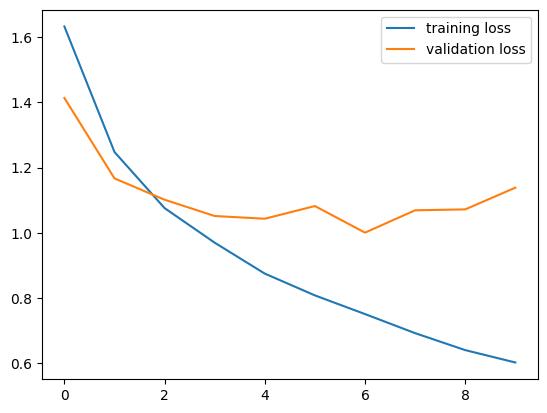


Figure B3.1 Loss curves for training and CV graph for SGD optimization with momentum = 0.3

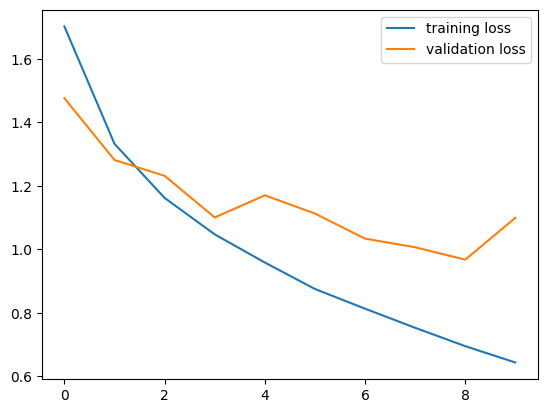


Figure B3.2 Loss curves for training and CV graph for SGD optimization with momentum = 0.6

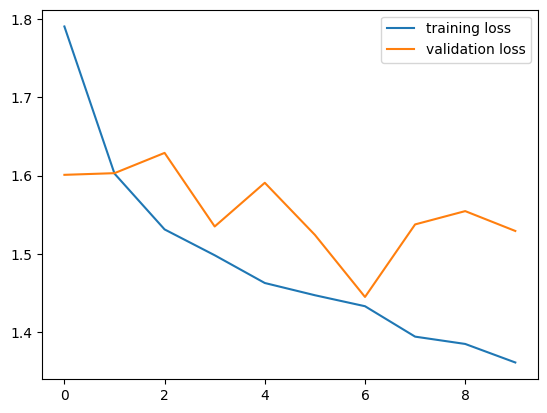


Figure B3.3 Loss curves for training and CV graph for SGD optimization with momentum = 0.9

**Note:** For all 3 graphs, the Y-axis represents the loss value and X-axis represents the epochs

**B4. Extras: Pre-Trained Models on CIFAR-10 ( ResNet18)**

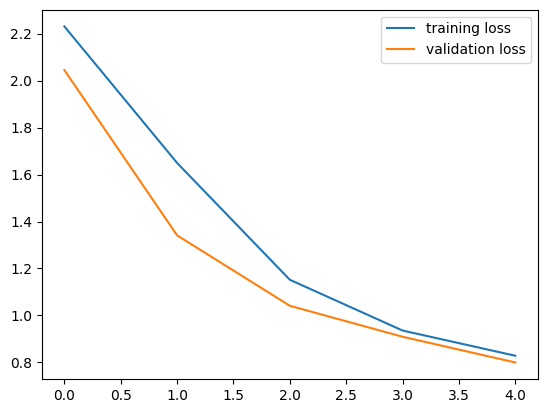
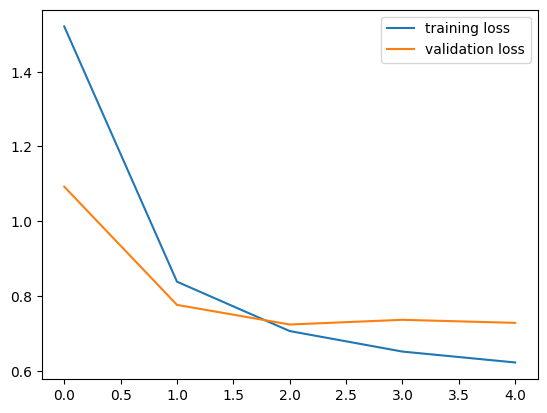
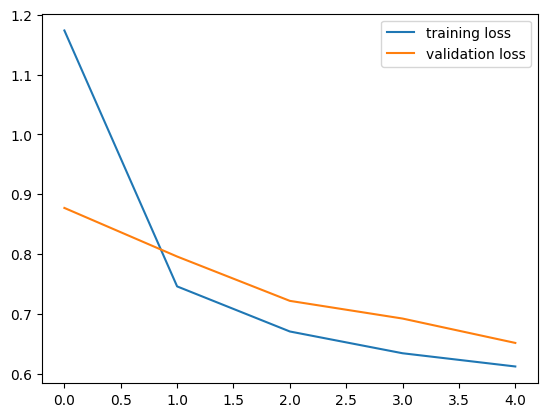


Figure B4.1 Graphs of the loss curves for ResNet18 with zero, one, and three hidden layers respectively (counting from left to right and up down). The Y-axis represents the loss value, and the X-axis the number of epochs