CS6910- Assignment 1

Creation of a neural network from scratch using numpy only in python and apply it in Fashion-MNIST Dataset and see how it perform under various hyper parameters and also after applying various optimizers to see how they vary from each other on different aspects

Shuvrajeet Das

Question 1:

The Fashion-MNIST Dataset's sample images are shown

examples

Question 2:-

The Neural Network code is implemented in the following project.

Question 3:-

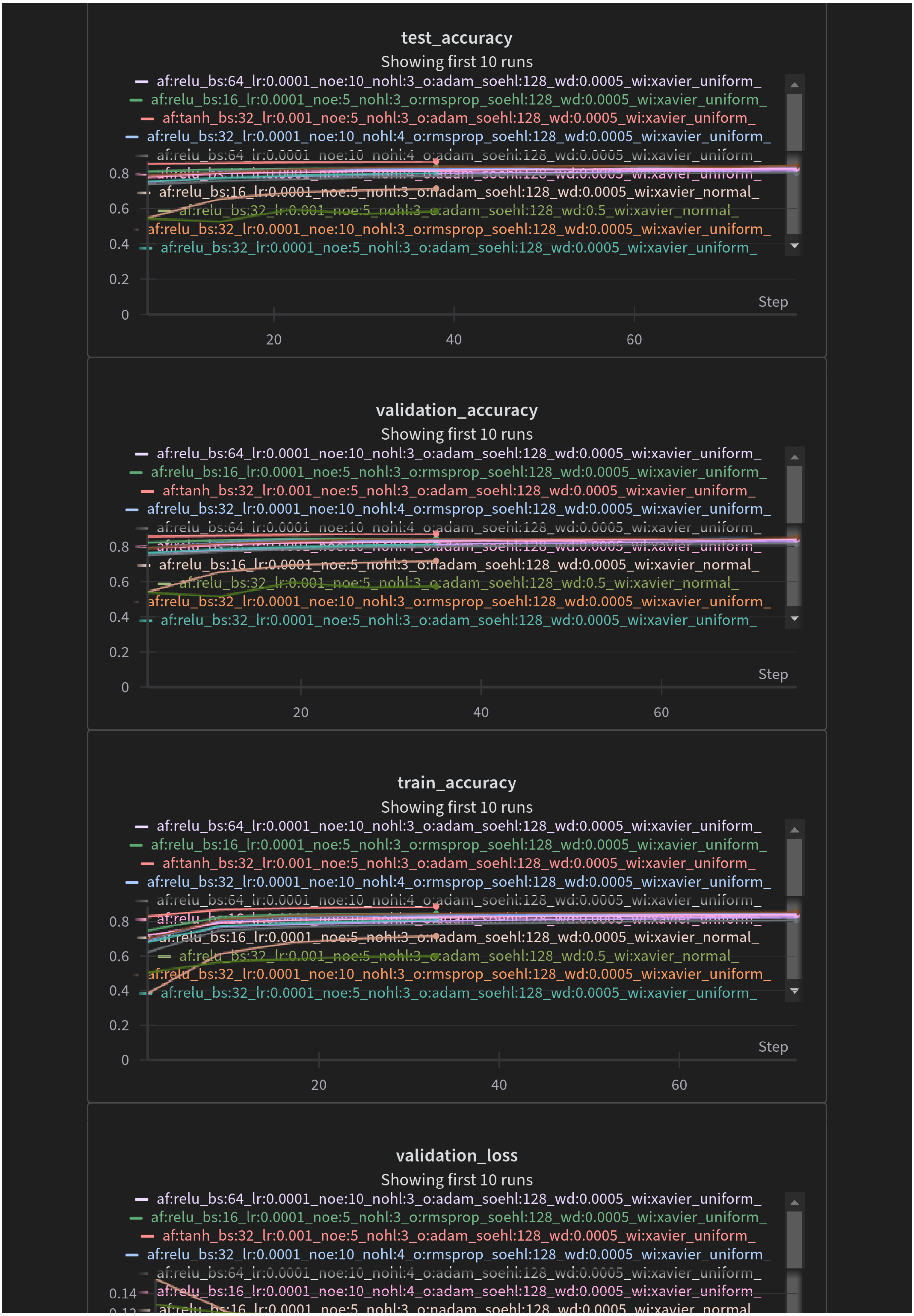
Implementation of the following optimizers are done:-

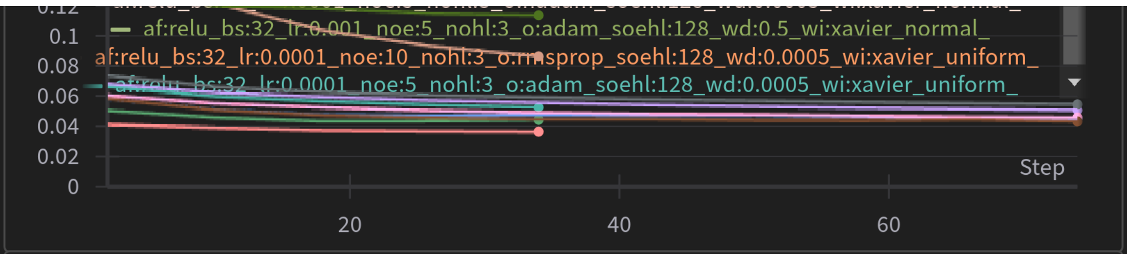
- **sgd**
- **momentum based gradient descent**
- **nesterov accelerated gradient descent**
- **rmsprop**
- **adam**
- **nadam**

Question 4:-

Hyper parameter setup:-

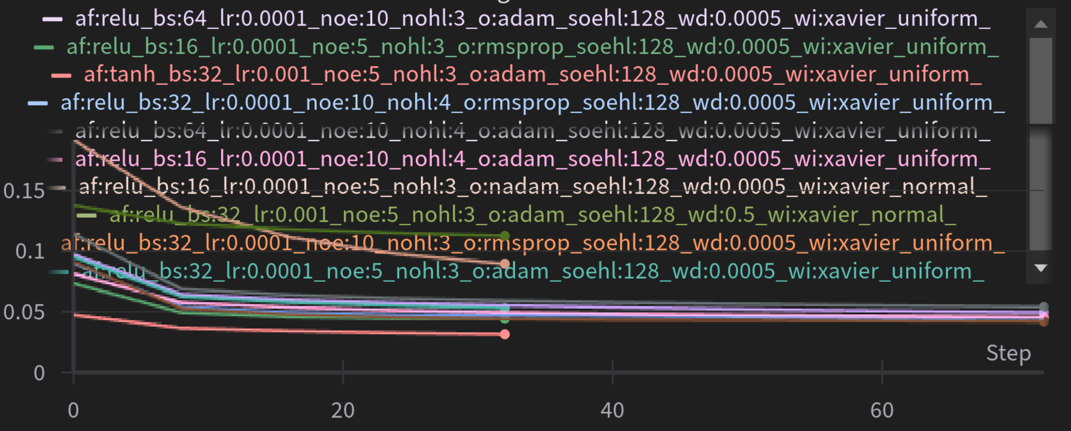
- number of epochs: 5, 10
- number of hidden layers: 3, 4, 5
- size of every hidden layer: 32, 64, 128
- weight decay (L2 regularisation): 0, 0.0005, 0.5
- learning rate: 1e-3, 1 e-4
- optimizer: sgd, momentum, nesterov, rmsprop, adam, nadam
- batch size: 16, 32, 64
- weight initialisation: random, Xavier
- activation functions: sigmoid, tanh, ReLU


test_accuracy
Showing first 10 runs
af:relu_bs:64_lr:0.0001_noe:10_nohl:3_o:adam_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:16_lr:0.0001_noe:5_nohl:3_o:rmsprop_soehl:128_wd:0.0005_wi:xavier_uniform_
af:tanh_bs:32_lr:0.001_noe:5_nohl:3_o:adam_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:32_lr:0.0001_noe:10_nohl:4_o:rmsprop_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:64_lr:0.0001_noe:10_nohl:4_o:adam_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:16_lr:0.0001_noe:10_nohl:4_o:adam_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:16_lr:0.0001_noe:5_nohl:3_o:nadam_soehl:128_wd:0.0005_wi:xavier_normal_
af:relu_bs:32_lr:0.001_noe:5_nohl:3_o:adam_soehl:128_wd:0.5_wi:xavier_normal_
af:relu_bs:32_lr:0.0001_noe:10_nohl:3_o:rmsprop_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:32_lr:0.0001_noe:5_nohl:3_o:adam_soehl:128_wd:0.0005_wi:xavier_uniform_
0.8
0.6
0.4
0.2
0
Step
20
40
60
validation_accuracy
Showing first 10 runs
af:relu_bs:64_lr:0.0001_noe:10_nohl:3_o:adam_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:16_lr:0.0001_noe:5_nohl:3_o:rmsprop_soehl:128_wd:0.0005_wi:xavier_uniform_
af:tanh_bs:32_lr:0.001_noe:5_nohl:3_o:adam_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:32_lr:0.0001_noe:10_nohl:4_o:rmsprop_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:64_lr:0.0001_noe:10_nohl:4_o:adam_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:16_lr:0.0001_noe:10_nohl:4_o:adam_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:16_lr:0.0001_noe:5_nohl:3_o:nadam_soehl:128_wd:0.0005_wi:xavier_normal_
af:relu_bs:32_lr:0.001_noe:5_nohl:3_o:adam_soehl:128_wd:0.5_wi:xavier_normal_
af:relu_bs:32_lr:0.0001_noe:10_nohl:3_o:rmsprop_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:32_lr:0.0001_noe:5_nohl:3_o:adam_soehl:128_wd:0.0005_wi:xavier_uniform_
0.8
0.6
0.4
0.2
0
Step
20
40
60
train_accuracy
Showing first 10 runs
af:relu_bs:64_lr:0.0001_noe:10_nohl:3_o:adam_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:16_lr:0.0001_noe:5_nohl:3_o:rmsprop_soehl:128_wd:0.0005_wi:xavier_uniform_
af:tanh_bs:32_lr:0.001_noe:5_nohl:3_o:adam_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:32_lr:0.0001_noe:10_nohl:4_o:rmsprop_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:64_lr:0.0001_noe:10_nohl:4_o:adam_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:16_lr:0.0001_noe:10_nohl:4_o:adam_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:16_lr:0.0001_noe:5_nohl:3_o:nadam_soehl:128_wd:0.0005_wi:xavier_normal_
af:relu_bs:32_lr:0.001_noe:5_nohl:3_o:adam_soehl:128_wd:0.5_wi:xavier_normal_
af:relu_bs:32_lr:0.0001_noe:10_nohl:3_o:rmsprop_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:32_lr:0.0001_noe:5_nohl:3_o:adam_soehl:128_wd:0.0005_wi:xavier_uniform_
0.8
0.6
0.4
0.2
0
Step
20
40
60
validation_loss
Showing first 10 runs
af:relu_bs:64_lr:0.0001_noe:10_nohl:3_o:adam_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:16_lr:0.0001_noe:5_nohl:3_o:rmsprop_soehl:128_wd:0.0005_wi:xavier_uniform_
af:tanh_bs:32_lr:0.001_noe:5_nohl:3_o:adam_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:32_lr:0.0001_noe:10_nohl:4_o:rmsprop_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:64_lr:0.0001_noe:10_nohl:4_o:adam_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:16_lr:0.0001_noe:10_nohl:4_o:adam_soehl:128_wd:0.0005_wi:xavier_uniform_
af:relu_bs:16_lr:0.0001_noe:5_nohl:3_o:nadam_soehl:128_wd:0.0005_wi:xavier_normal_
0.14
0.12



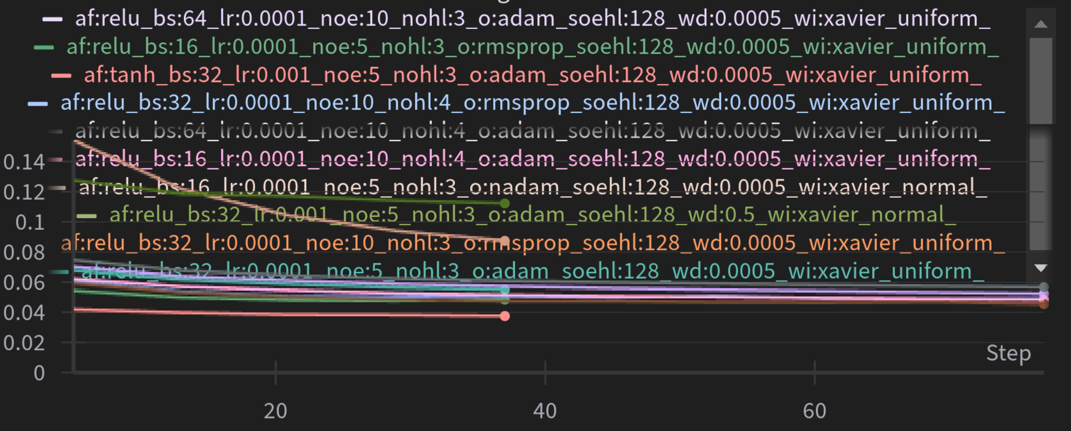
train_loss

Showing first 10 runs



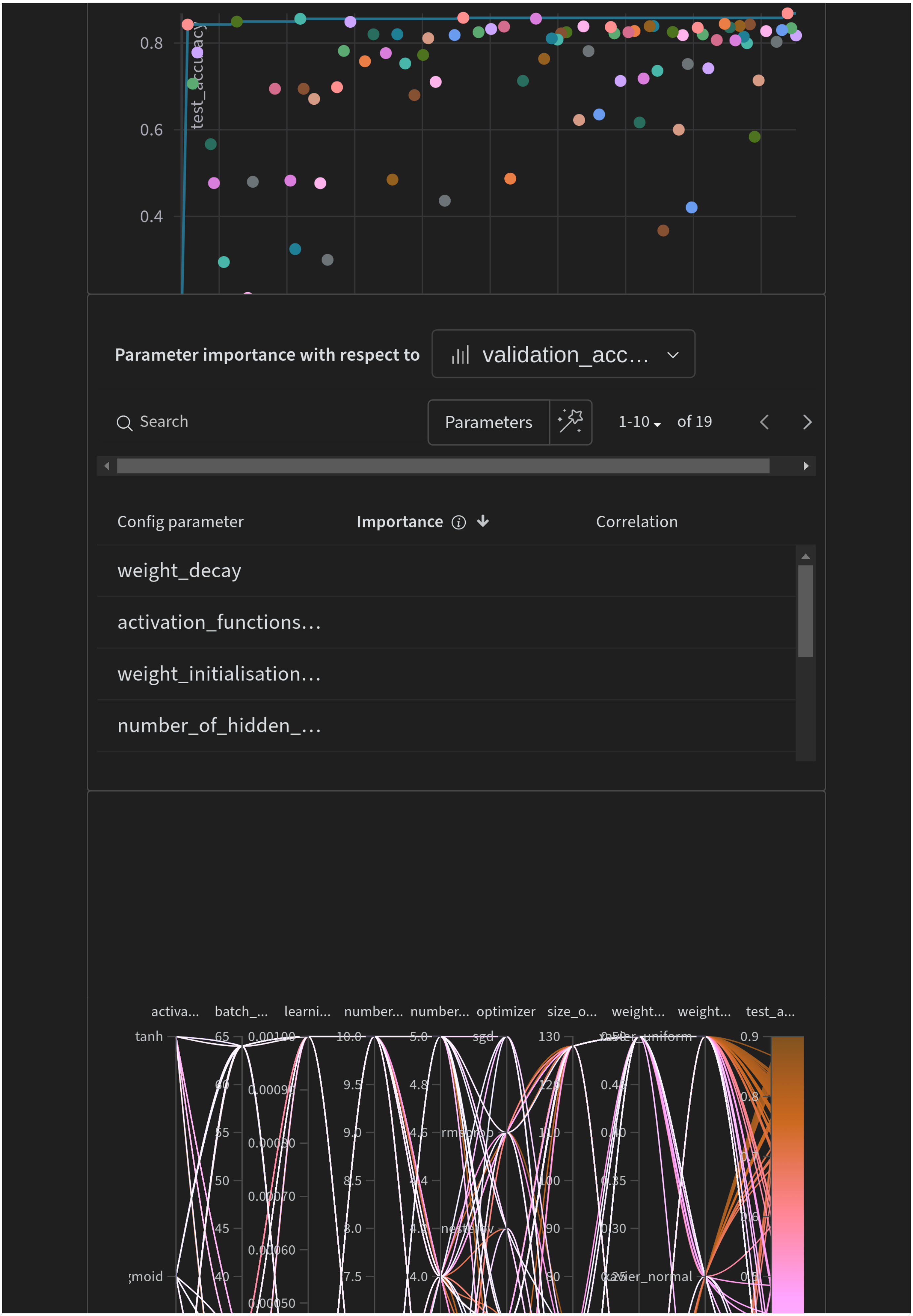
test_loss

Showing first 10 runs



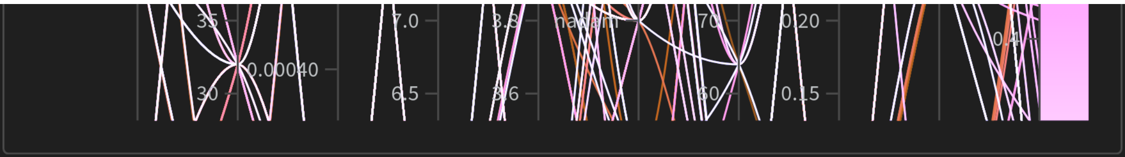
Question 5 and 6:-

test_accuracy v. created



Parameter importance with respect to validation_acc…

Config parameter	Importance	Correlation
weight_decay		
activation_functions…		
weight_initialisation…		
number_of_hidden_…		



The Best results achieved so far is with the hyper parameters settings as:-

- Activation function: Tanh
- Batch size: 32
- Learning rate: 0.001
- Number of epochs: 5
- Number of hidden layers: 3
- Activation function: Adam
- Size of every hidden layer: 128
- Weight decay: 0.0005
- Weight initialisation: Xavier uniform

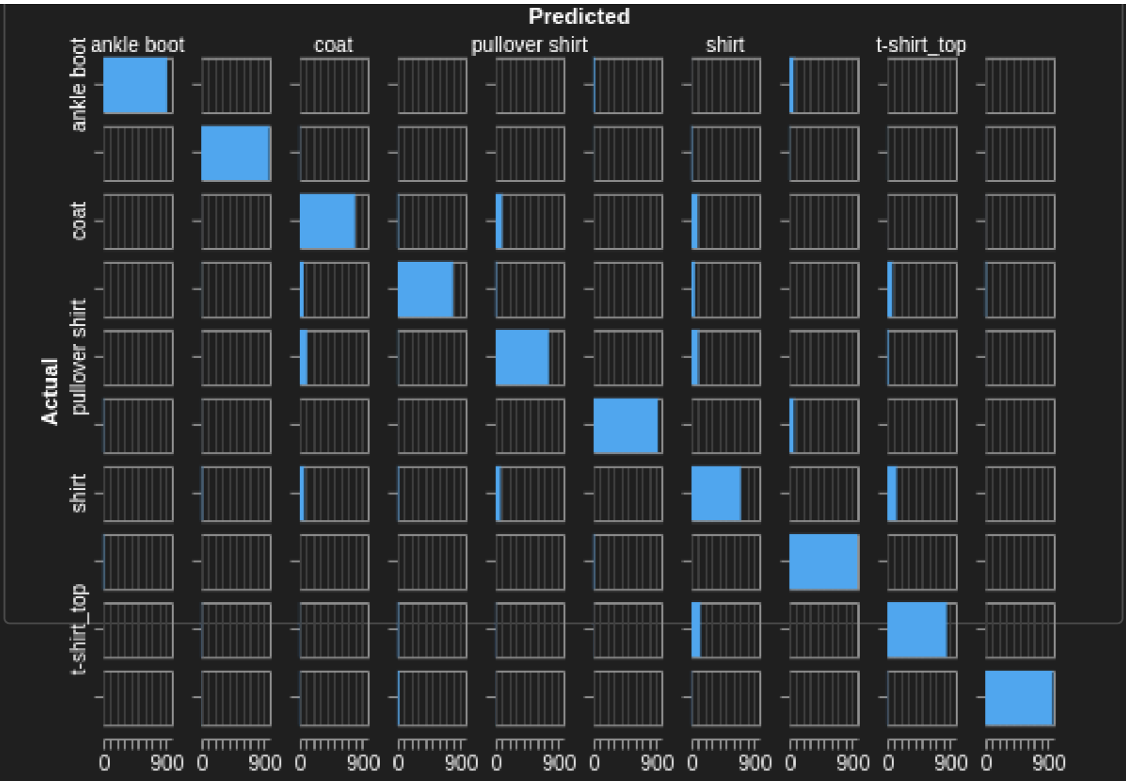
Making it reach up to an accuracy of 86.85%.

The observations done:

- The longer the runtime the more better the result gets
- Adam optimizer is highly suitable for this as it is highly correlated
- The smaller the batch size the better result result that is tries to produce
- Lesser number of hidden layers produces better results
- Weight decay is highly discarded since it is opting for smaller complexity models on its own.
- Mostly tanh as an activation is performing better than the rest activation.

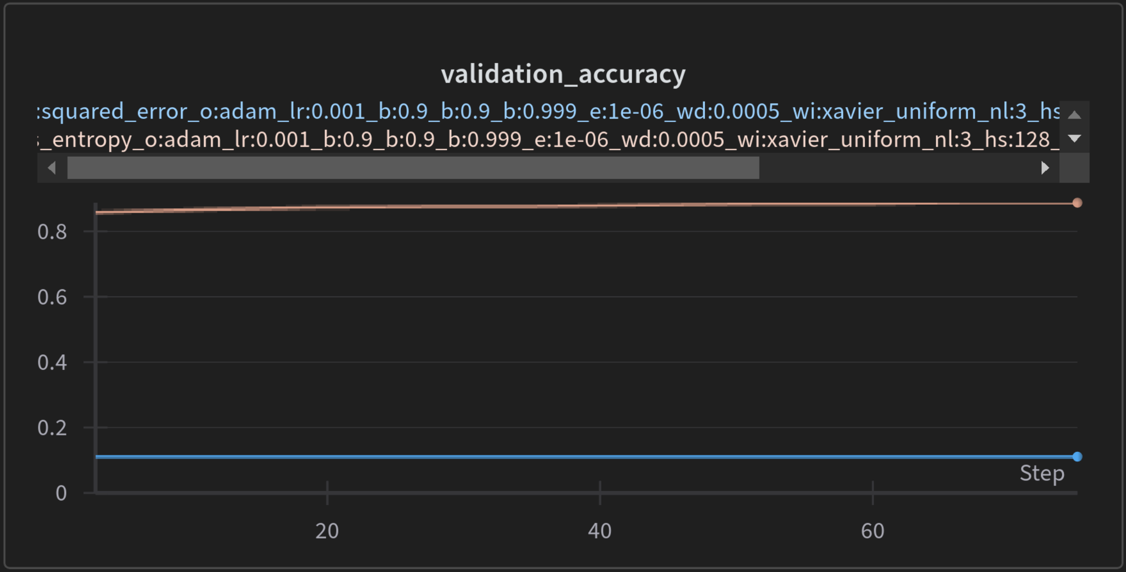
Question 7:-

The confusion matrix of the best model.



Question 8:-

The validation accuracy plot of Categorical Cross Entropy loss vs Mean Squared Error loss:



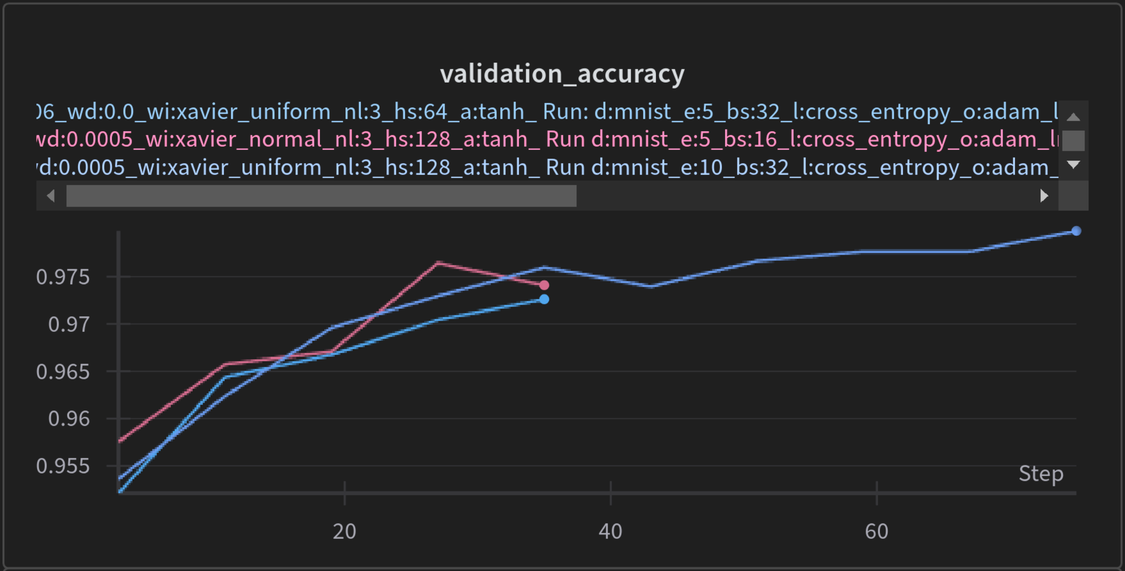
Thus we can see that the neural network didn't perform very well in the case of Mean Squared Error loss function.

Question 9:-

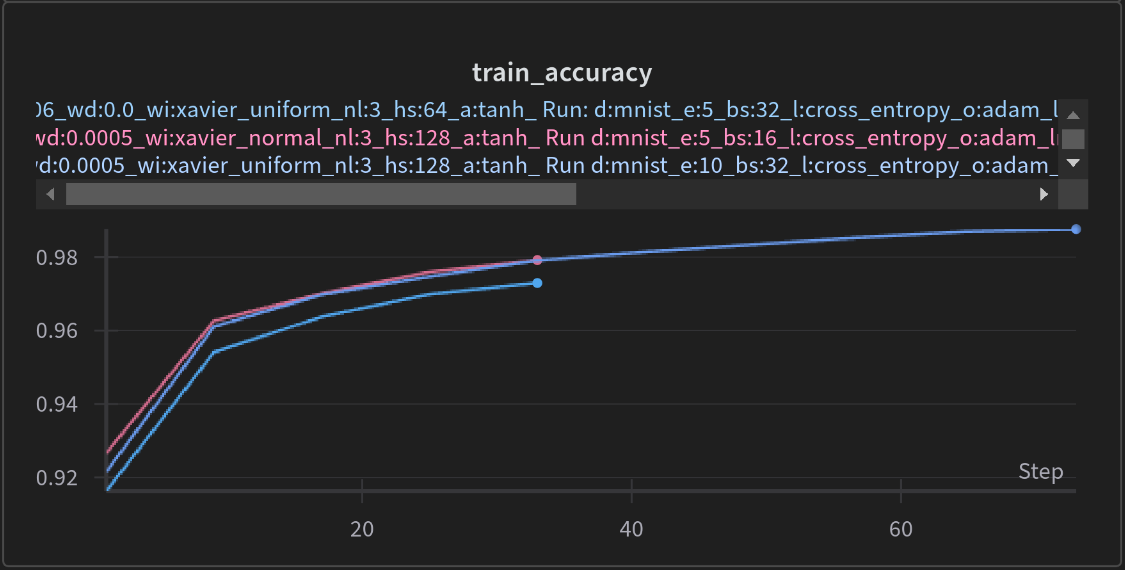
The github link of the assignment is: https://github.com/TheUnsolvedDev/cs6910_assignment1

Question 10:-

The Choice of the Hyper parameters for the MNIST Dataset:-



validation_accuracy



train_accuracy

Below contains the table for the hyper parameters chosen for this experiment.

MNIST	5	Cross Entropy	Adam	0.001	0	Xavier Uniform	3	64	Tanh
MNIST	5	Cross Entropy	Adam	0.001	0.0005	Xavier Normal	3	128	Tanh
MNIST	10	Cross Entropy	Adam	0.001	0.0005	Xavier Normal	3	64	Tanh

Created with ❤ on Weights & Biases.

https://wandb.ai/shuvrajeet/CS23E001_DL_1/reports/CS6910-Assignment-1--Vmlldzo3MDY0OTU0