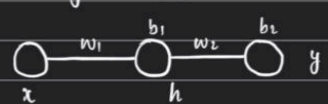
ML Assignment 3

Monsoon 2024
Dr. Jainendra Shukla
Rahul Oberoi
2021555

Section A:

1a) $x=1$ $\quad w_1 = 0.5$ $\quad b_1 = 0.5$ $\quad \eta = 0.01$

$y=3$ $\quad w_2 = 0.5$ $\quad b_2 = 0.5$



$h = ReLU\,(1(0.5) + 0.5) = 1$

$\hat{y} = ReLU\,(1(0.5) + 0.5) = 1$

$Loss = \frac{1}{2}(\hat{y} - y)^2 = \frac{(1-3)^2}{2} = 2$

① $\frac{\partial loss}{\partial \hat{y}} = \hat{y} - y = 1-3 = -2$

② $\frac{\partial loss}{\partial w_2} = \frac{\partial loss}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_2} = -2 \cdot h = -2$

③ $\frac{\partial loss}{\partial b_2} = \frac{\partial loss}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial b_2} = -2 \cdot 1 = -2$

④ $\frac{\partial loss}{\partial h} = \frac{\partial loss}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial h} = -2.0\, w_2 = -2 \cdot 0.5 = -1$

⑤ $\frac{\partial loss}{w_1} = \frac{\partial loss}{\partial h} \cdot \frac{\partial h}{\partial w_1} = -1 \cdot x = -1(1) = -1$

⑥ $\frac{\partial loss}{b_1} = \frac{\partial loss}{\partial h} \frac{\partial h}{\partial b_1} = -1 \cdot (1) = -1$

$w_1 = 0.5 - 0.01(-1.0) = 0.5 + 0.01 = 0.51$

$b_1 = 0.5 - 0.01(-1.0) = 0.5 + 0.01 = 0.51$

$w_2 = 0.5 - 0.01(-2.0) = 0.5 + 0.02 = 0.52$

$b_2 = 0.5 - 0.01(-2.0) = 0.5 + 0.02 = 0.52$

$h = ReLU(0.51(2) + 0.51) = ReLU(1.53) = 1.53$

$\hat{y} = 0.52(1.53) + 0.52 = 0.7956 + 0.52 = 1.31$

$Loss = \frac{1}{2}(1.31 - 4)^2 = \frac{1}{2}(7.23) = 3.6$

Gradients

① $= 1.31 - 4 = 2.69$ $\qquad w_1 = 0.51 + 0.02 = 0.53$

② $= -2.69(1.53) = -4.11$ $\qquad b_1 = 0.51 + 0.14 = 0.524$

③ $= -2.69(1) = -2.69$ $\qquad w_2 = 0.52 + 0.04 = 0.56$

④ $= -2.69(0.52) = -1.4$ $\qquad b_2 = 0.52 + 0.02 = 0.54$

⑤ $= -1.4(2) = -2.8$

⑥ $= -1.4(1) = -1.4$

$h = ReLU(0.53(3) + 0.52) = 2.13$

$\hat{y} = ReLU(0.56(2.13) + 0.54) = 1.7$

$Loss = \frac{1}{2}(1.7 - 5)^2 = 5.4$

Gradients

① $= 1.7 - 5 = -3.3$ $\qquad w_1 = 0.59$

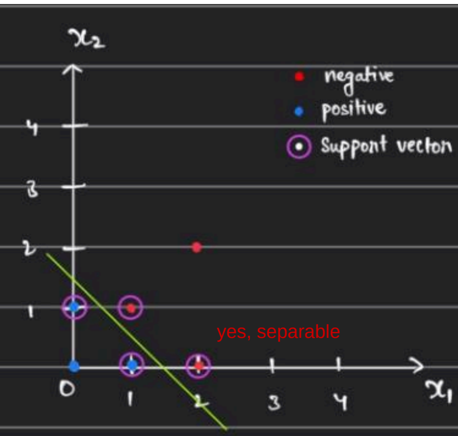
② $= -3.3(2.13) = -6.9$ $\qquad b_1 = 0.54$

③ $= -3.3(1) = -3.3$ $\qquad w_2 = 0.63$

④ $= -3.3(0.56) = -1.82$ $\qquad b_2 = 0.57$

⑤ $= -1.82(3) = -5.46$

⑥ $= -1.82(1) = -1.82$



(a)

For +ve class

$$w x_i + b \geq +1$$

Support vectors

$$w x_i + b = \pm 1$$

For -ve class

$$w x_i + b \leq -1$$

(1,0)	(1,1)	(2,0)
$w_1 + b = +1$	$w_1 + w_2 + b = -1$	$2w_1 + b = -1$

$$w_1 + \cancel{b} = 1$$
$$-2w_1 \pm \cancel{b} = \mp 1$$
$$-w_1 = 2 \qquad w_1 = -2$$

(b)

$$-2 + b = 1 \qquad -2 + w_2 + 3 = -1$$
$$b = 3 \qquad w_2 = -2$$

Verifying for support vectors:

$-2 + 3 = 1$ ✓ (1,0)

$-2 - 2 + 3 = -1$ ✓ (1,1)

$-4 + 3 = -1$ ✓ (2,0)

$\boxed{-2 + 3 = 1 \ (0,1) \text{ is also Support Vector}}$

c)

$$\text{Margin} = \gamma = \frac{1}{\|w\|}$$

$$\Rightarrow \frac{1}{\sqrt{(-2)^2+(0)^2}} = \frac{1}{\sqrt{4}} = \frac{1}{2} = 0.5$$ a

Support Vectors are data points which lie on boundaries

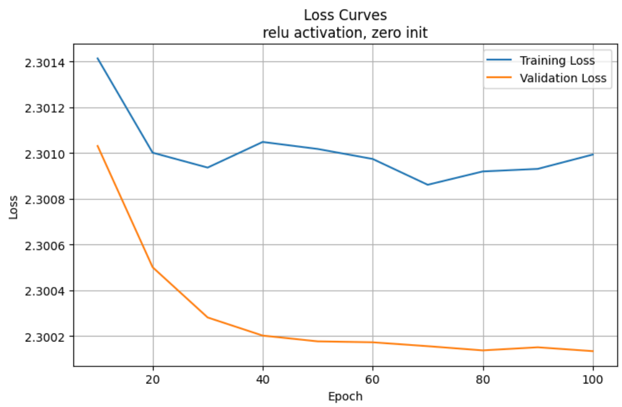
$$\Rightarrow -2x_1 + 0x_2 + 5 = 0$$
$$-2x_1 + 5 = 0$$

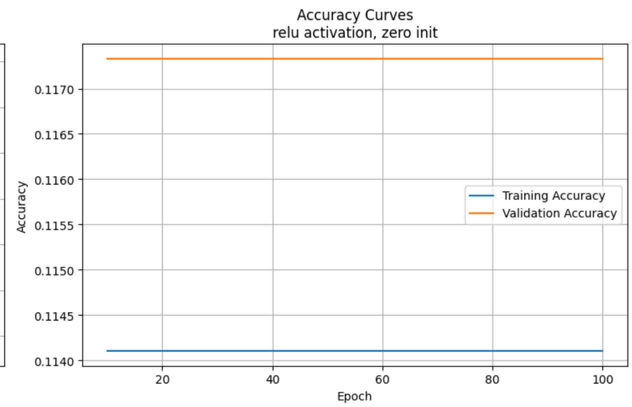
(1,2)	(2,3)	(3,3)	(4,1)
$-2(1)+5$	$-2(2)+5$	$-2(3)+5$	$-2(4)+5$
$\Rightarrow 3$	$\Rightarrow 1$	$\Rightarrow -1$	$\Rightarrow -3$
	Support vector b	Support Vector	

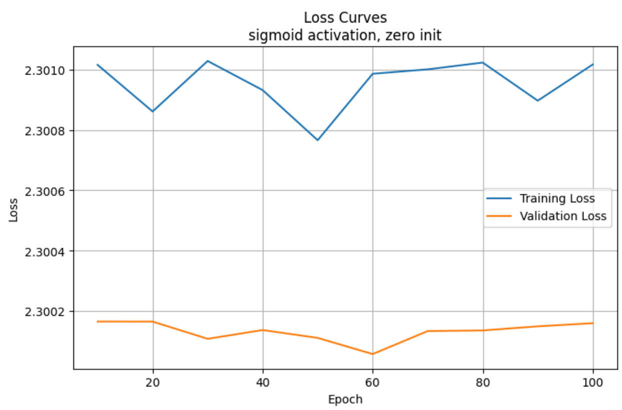
Class of $(1,3) \Rightarrow -2(1)+0(3)+5$
$\Rightarrow 3$
$\boxed{\Rightarrow +ve}$ c

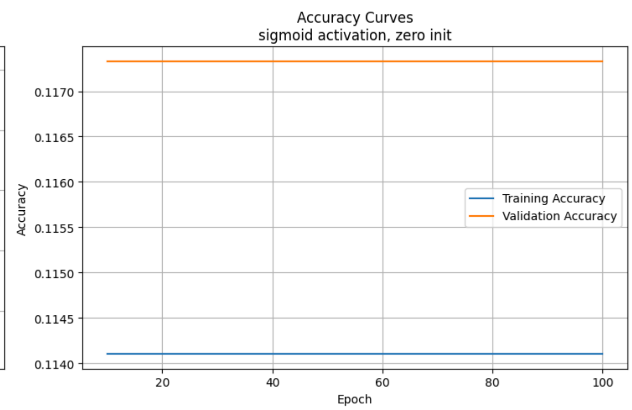
Section B:

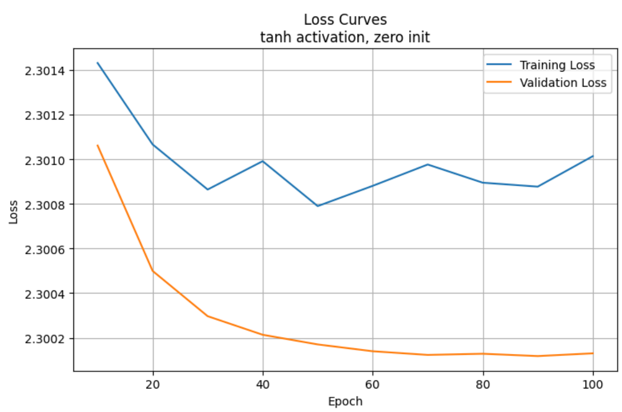
Plots:

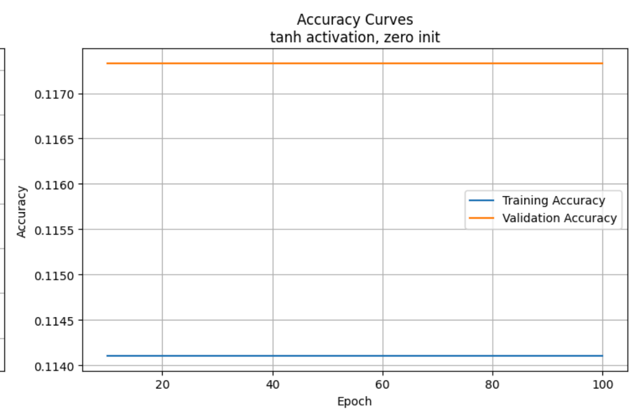


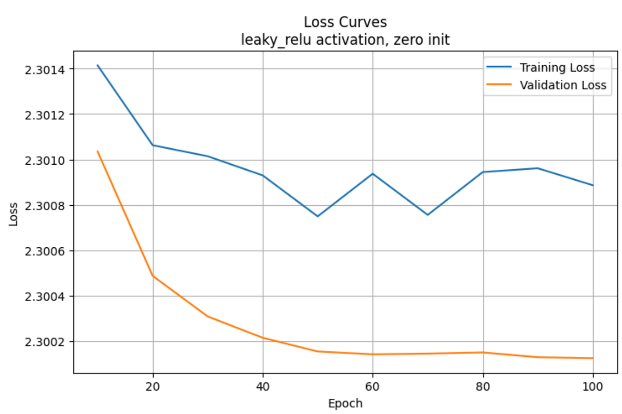


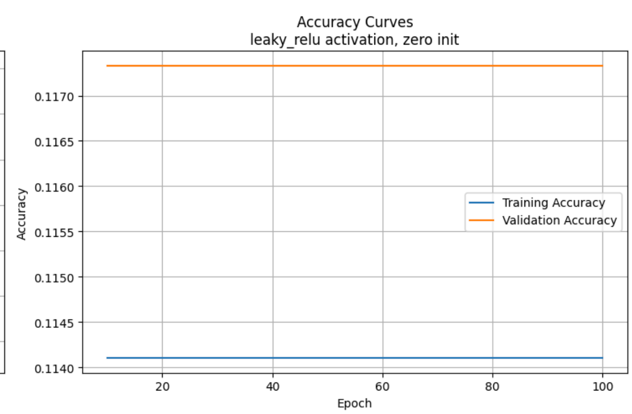


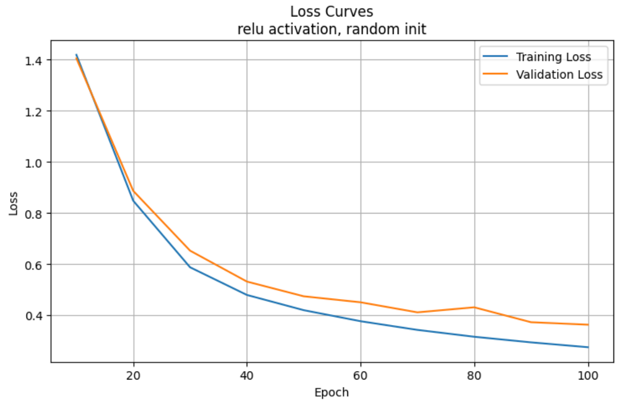


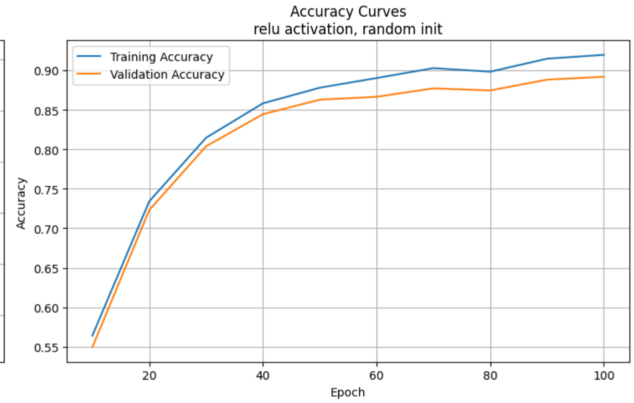


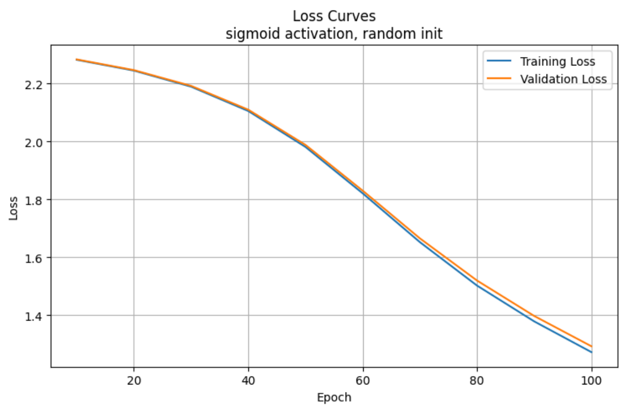


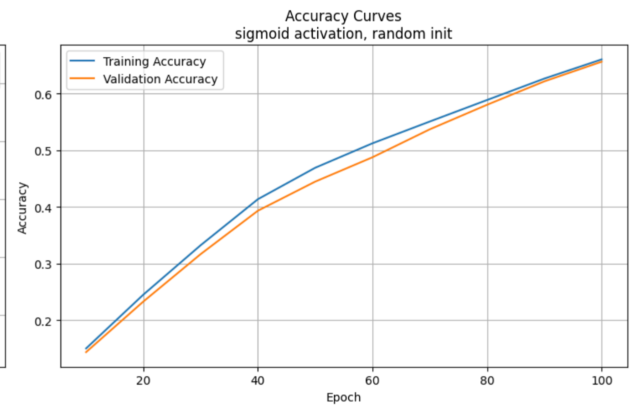


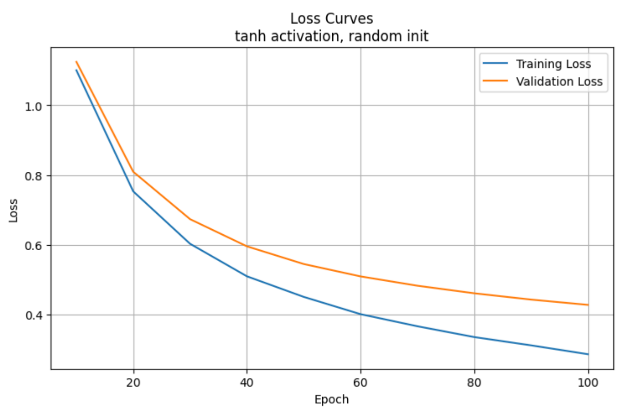


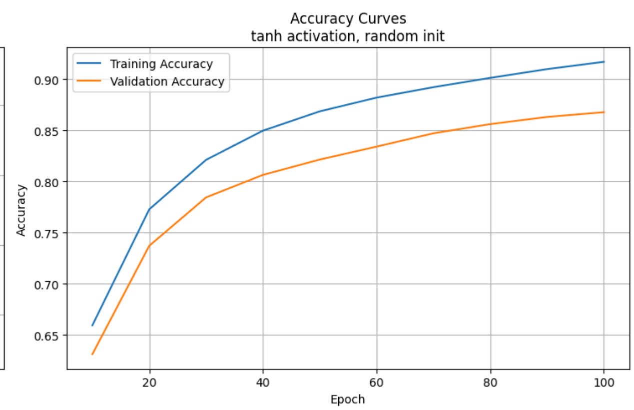

Loss Curves
relu activation, random init
Training Loss
Validation Loss
Loss
Epoch

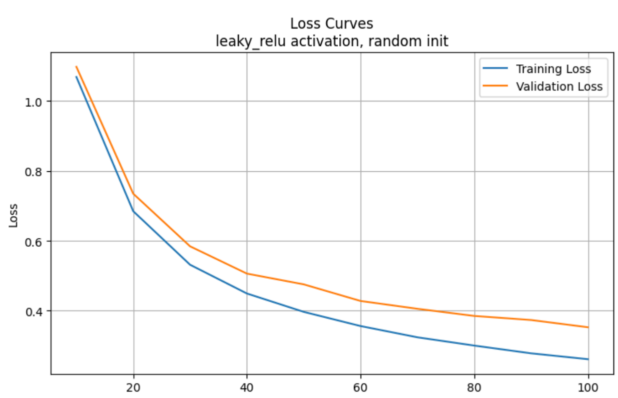

Accuracy Curves
relu activation, random init
Training Accuracy
Validation Accuracy
Accuracy
Epoch

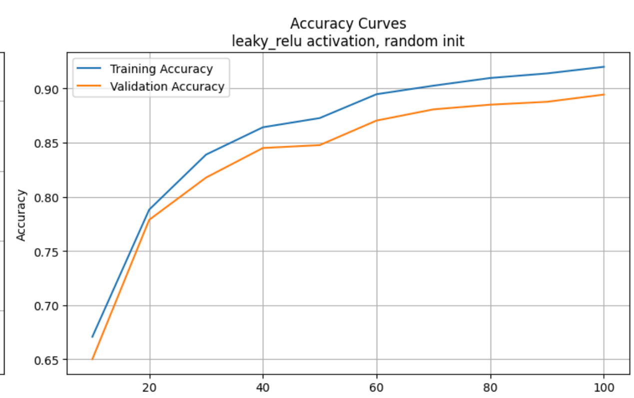

Loss Curves
sigmoid activation, random init
Training Loss
Validation Loss
Loss
Epoch

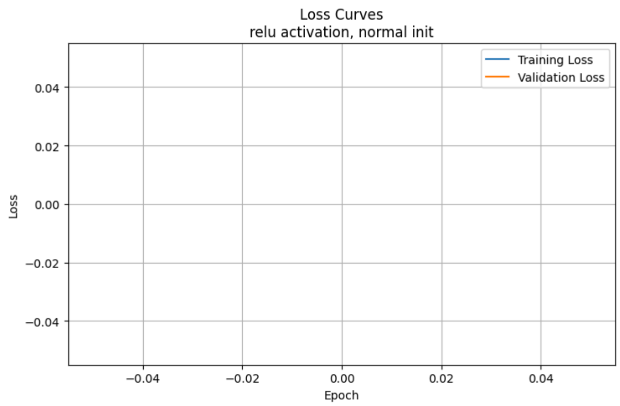

Accuracy Curves
sigmoid activation, random init
Training Accuracy
Validation Accuracy
Accuracy
Epoch

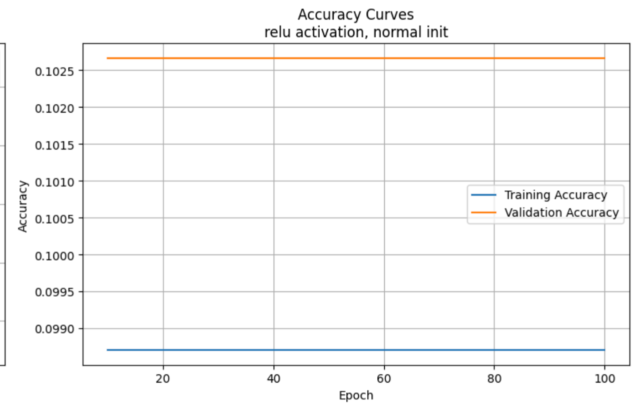

Loss Curves
tanh activation, random init
Training Loss
Validation Loss
Loss
Epoch

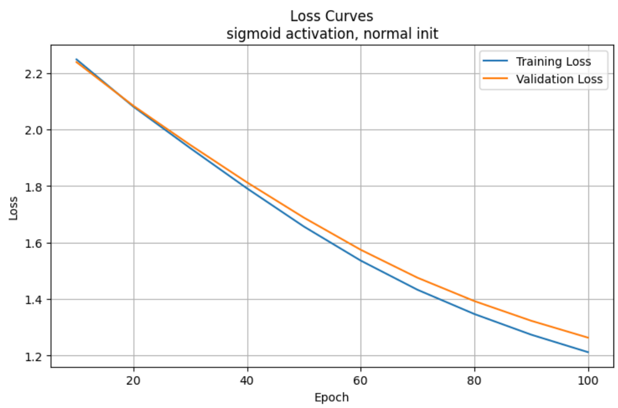

Accuracy Curves
tanh activation, random init
Training Accuracy
Validation Accuracy
Accuracy
Epoch

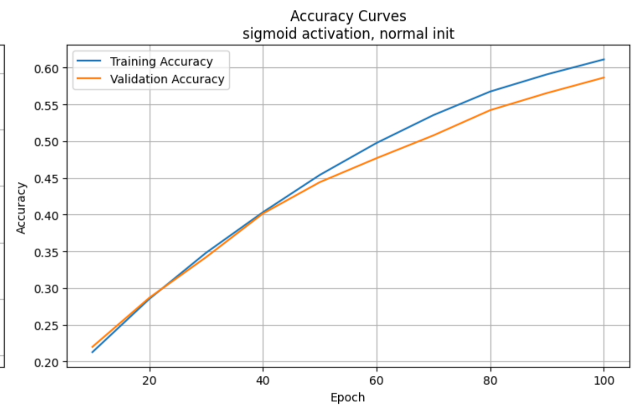

Loss Curves
leaky_relu activation, random init
Training Loss
Validation Loss
Loss
Epoch

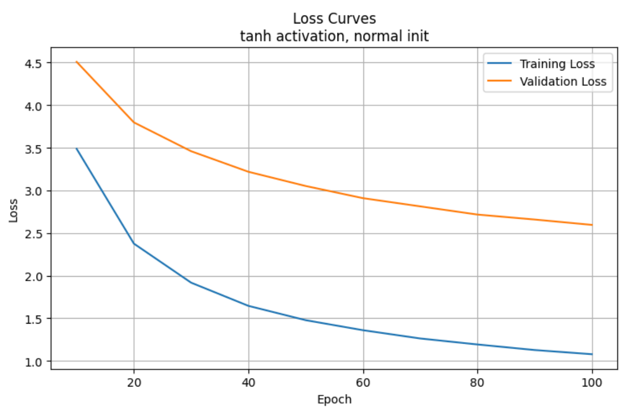

Accuracy Curves
leaky_relu activation, random init
Training Accuracy
Validation Accuracy
Accuracy
Epoch

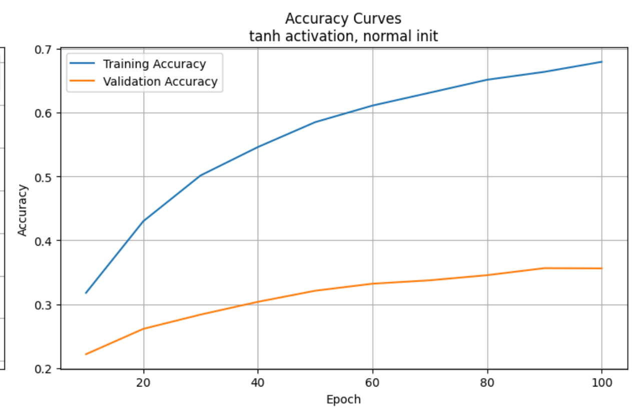

Loss Curves
relu activation, normal init
Training Loss
Validation Loss
Loss
Epoch

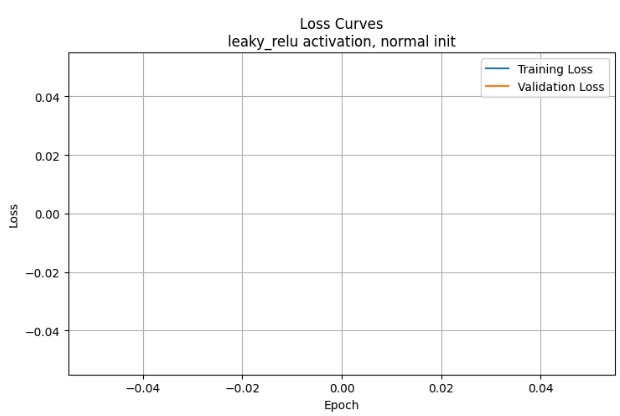

Accuracy Curves
relu activation, normal init
Training Accuracy
Validation Accuracy
Accuracy
Epoch

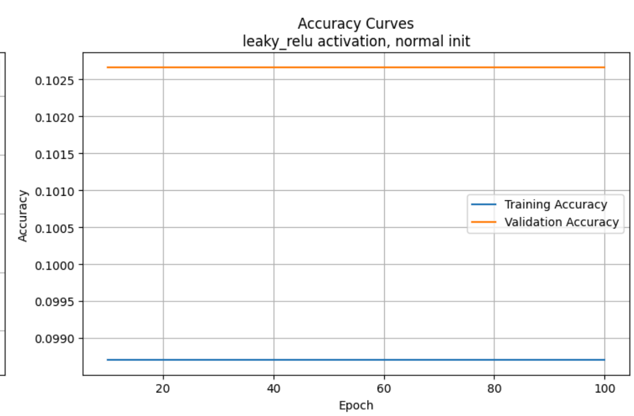

Loss Curves
sigmoid activation, normal init
Training Loss
Validation Loss
Loss
Epoch


Accuracy Curves
sigmoid activation, normal init
Training Accuracy
Validation Accuracy
Accuracy
Epoch


Loss Curves
tanh activation, normal init
Training Loss
Validation Loss
Loss
Epoch


Accuracy Curves
tanh activation, normal init
Training Accuracy
Validation Accuracy
Accuracy
Epoch


Loss Curves
leaky_relu activation, normal init
Training Loss
Validation Loss
Loss
Epoch


Accuracy Curves
leaky_relu activation, normal init
Training Accuracy
Validation Accuracy
Accuracy
Epoch

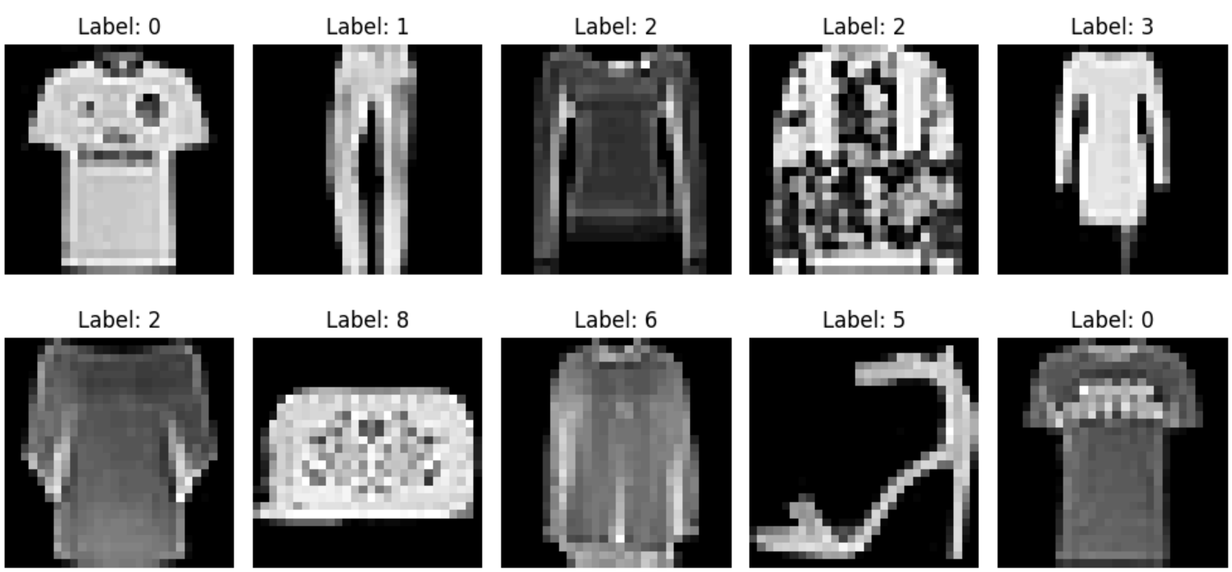
Results on learning rate = 2e-2:

Weight Init	**Activation**	**Train Loss**	**Train Acc.**	**Val Loss**	**Val Acc.**	**Test Acc.**
Zero	Relu	2.3009	0.1141	2.3001	0.1173	0.11
Zero	Sigmoid	2.3010	0.1141	2.3001	0.1173	0.11
Zero	Tanh	2.3009	0.1141	2.3001	0.1173	0.11
Zero	Leaky Relu	2.3008	0.1141	2.3001	0.1173	0.11
Random	**Relu**	**0.2604**	**0.9132**	**0.3261**	**0.897**	**0.895**
Random	Sigmoid	1.1552	0.7175	1.1806	0.7040	0.706
Random	Tanh	0.3076	0.9102	0.484	0.8547	0.844
Random	**Leaky Relu**	**0.2544**	**0.9250**	**0.3256**	**0.8987**	**0.893**
Normal	Relu	NaN	0.0987	NaN	0.1027	0.08
Normal	Sigmoid	1.1214	0.6291	1.1857	0.6093	0.604
Normal	Tanh	1.1635	0.6511	2.673	0.3373	0.332
Normal	Leaky Relu	NaN	0.0987	NaN	0.1027	0.08

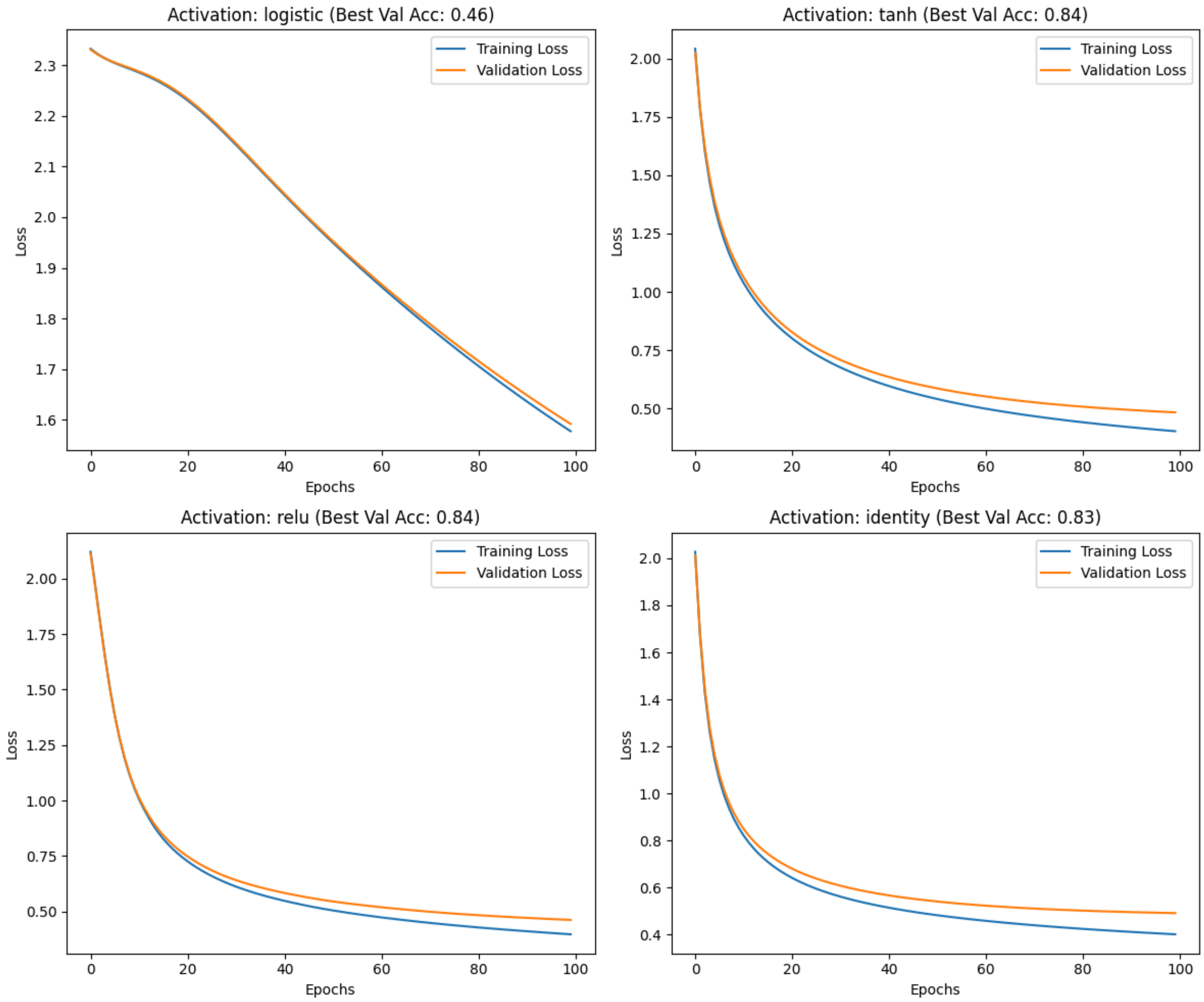
The reason why I am getting NaN values for weight init = normal and activation = relu/leaky relu is likely due to exploding gradients. I experimented with multiple values but I only got NaN values. The accuracies are probably calculated as the last predictions (before exploding gradients) are obtained.

Section C:

Part 1. Data has been normalized and first 10 samples of test set are:



Part 2. The best performance was received on "ReLU" activation function and the second best was "tanh". The metrics and plots for each of the activation function are attached below:



Activation Function	Val Loss (Val Set)	Macro F1 (Test Set)
Logistic	1.5920	0.37
Tanh	0.4834	0.84
ReLU	0.4620	0.84
Identity	0.4913	0.83

Rest of the metrics (Precision, Recall and Accuracy) are available in the code.

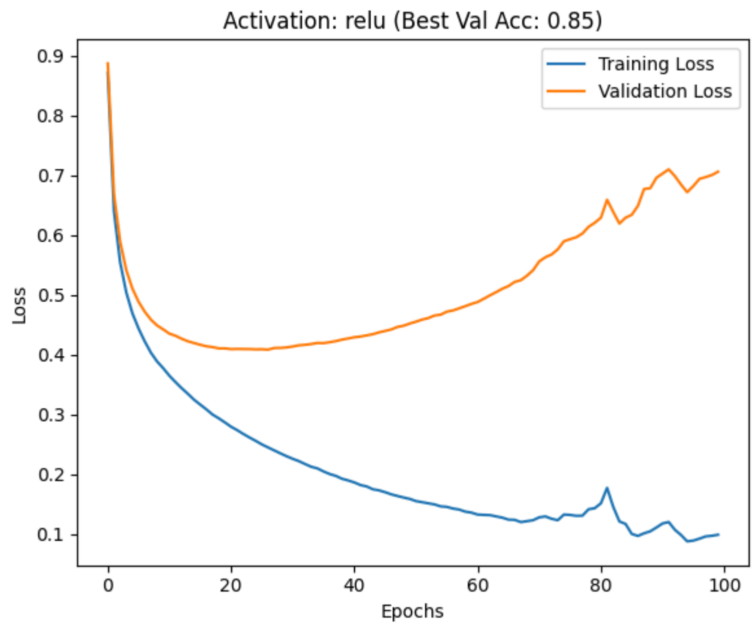
Part 3. Parameters tested on 3 folds using ReLU activation as it was the best activation function obtained in the previous part:

Parameters	Values
Solver	"adam", "sgd"

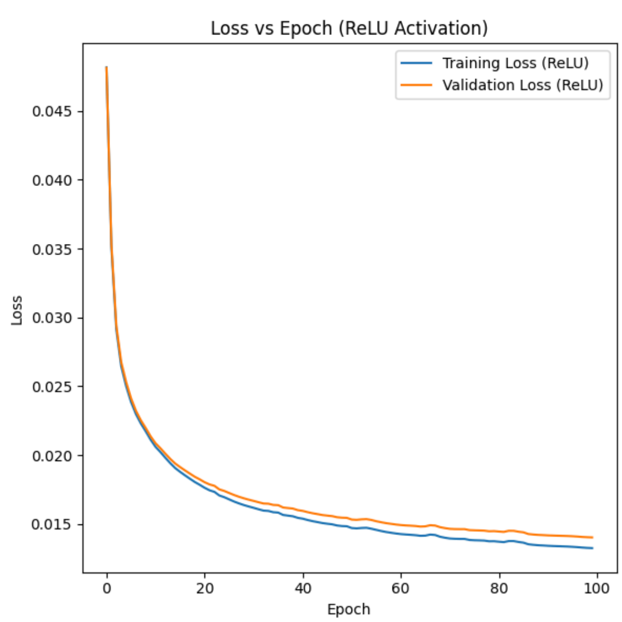
Batch Size	128, 64, 32
Learning Rate	2e-4, 2e-5, 2e-6

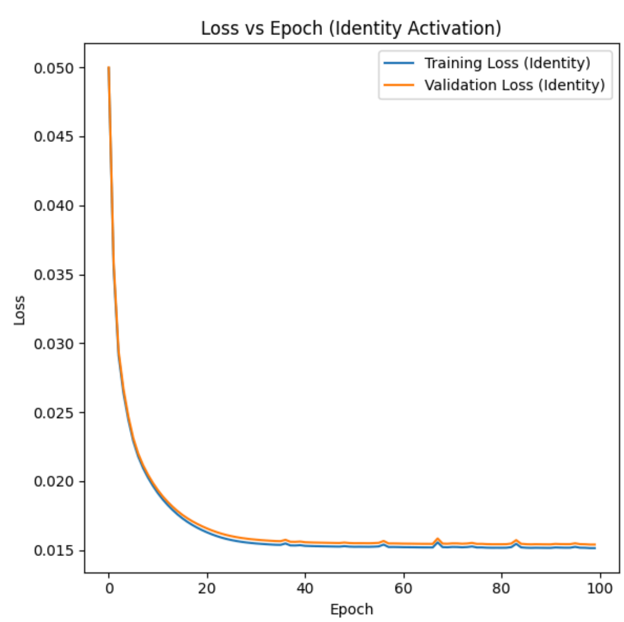
Best parameters (Macro F1 = 0.85, Val Loss = 0.4080):

Solver	“adam”
Batch Size	64
Learning Rate	2e-4

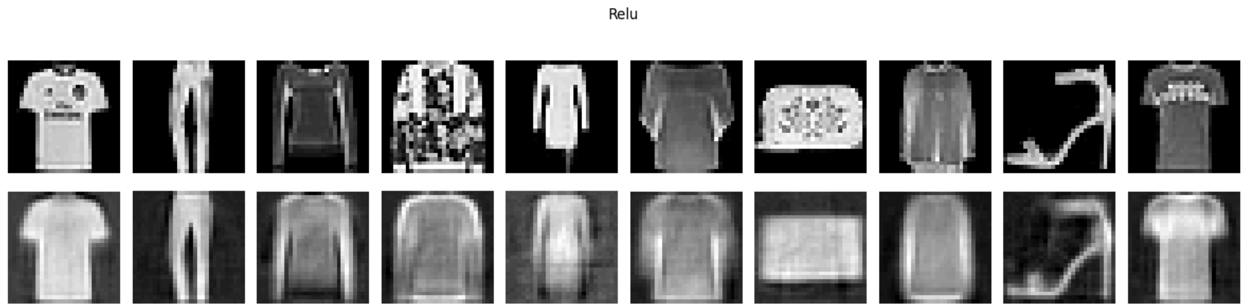


Part 4: Plots obtained:

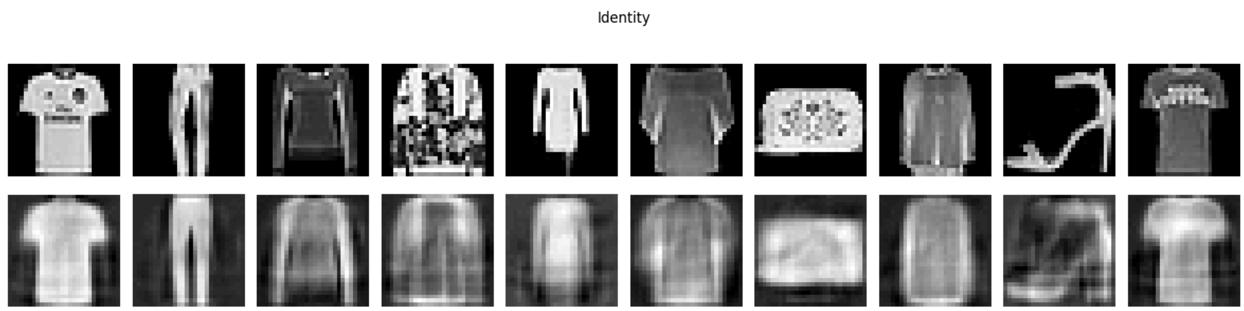




Regenerated Images using ReLU (Images above are original):



Regenerated Images using Identity (Images above are original):



Part 5:
Accuracy of classifier trained on ReLU features: 0.712
Accuracy of classifier trained on Identity features: 0.5915

The reason for obtaining decent performance here could be due to the fact that there were 2 layers before this “a” layer (32 neurons) and the information at this layer is high-quality which enables the smaller MLP to perform decently. It reduces the dimensions while maintaining a good amount of information. Since part 4 was an autoencoder reconstruction task, the main goal was to minimize the mean squared error (minimizing the gap between the predicted and true pixels) whereas in part 2 was a classification task and minimized the log loss. Hence, there is some loss in information in the extracted 32-sized vector but it still performs decently.