

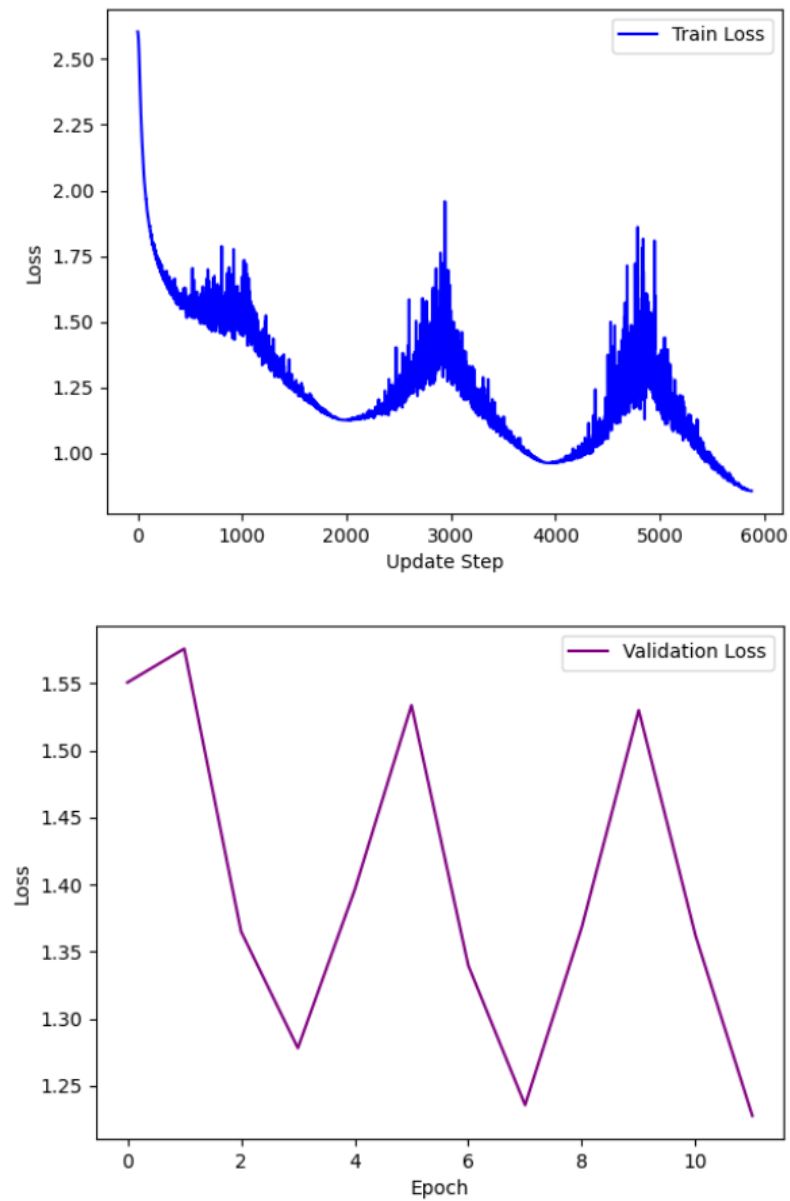
## Assignment2\_BonusPoint

### Exercise 5.1

5.1(a), I set the different hidden node and lambda combo, the highest accuracy combo on test is hidden node = **300**, lambda = **1e-3**. The highest accuracy on test is **0.5609**

	100	200	300
1e-4	0.5359	0.5467	0.5594
1e-3	0.5392	0.5528	<b>0.5609</b>
1e-2	0.5285	0.5306	0.5377

The below chart shows the train and validation loss curve:



(b) In my code, I use the drop from from 51 to 56 at the function Applynetworkrok and from 80 to 84 at Backpass.

```
# Exercise 5.1 (b)
if Train:
    keep_drop = 1 - dropout_p
    dropout_mask = (np.random.rand(*h.shape) > dropout_p).astype(float)
    h *= dropout_mask
    h /= keep_drop
    fp_data['dropout_mask'] = dropout_mask

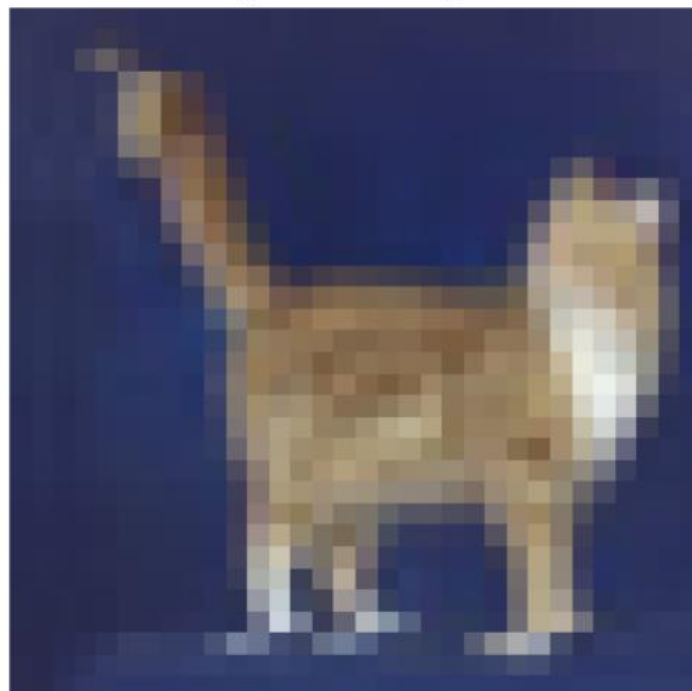
# Exercise 5.1 (b)
if Train:
    keep_prob = 1 - dropout_p
    dropout_mask = fp_data['dropout_mask']
    G_hidded *= dropout_mask
    G_hidded /= keep_prob
```

(c)According the tutorial, I use the data augmentation during the training ( 423 - 503), the below image shows the data before and after augmentation:

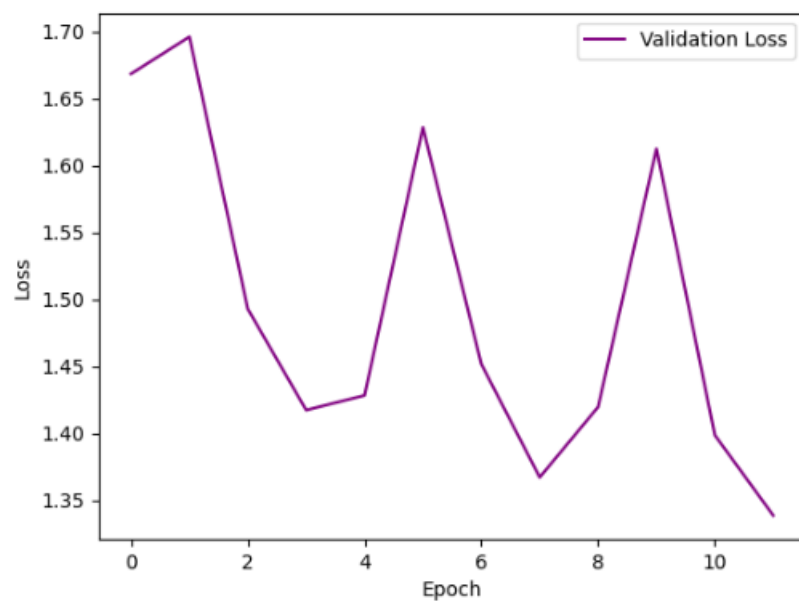
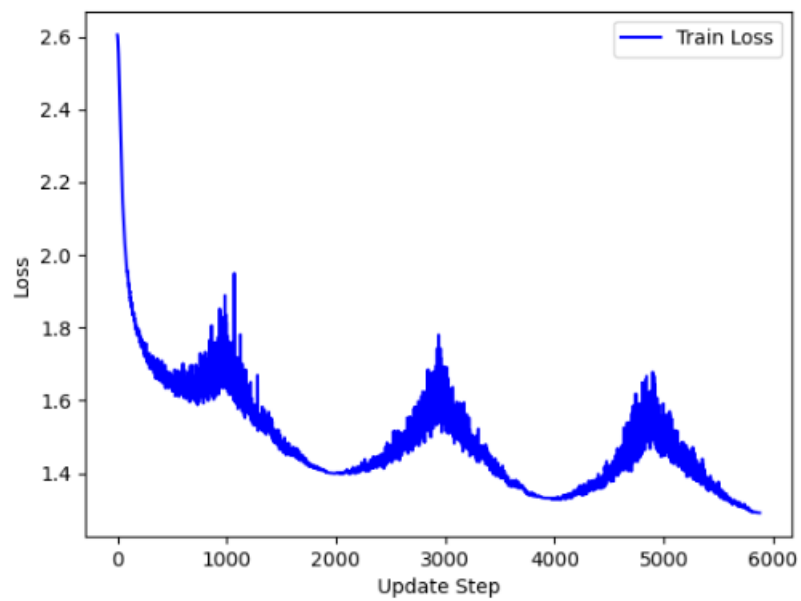
Original Image 0



Augmented Image 0



After these 3 steps, the final accuracy on test is **0.5273** and the train and validation loss curve will be show below:

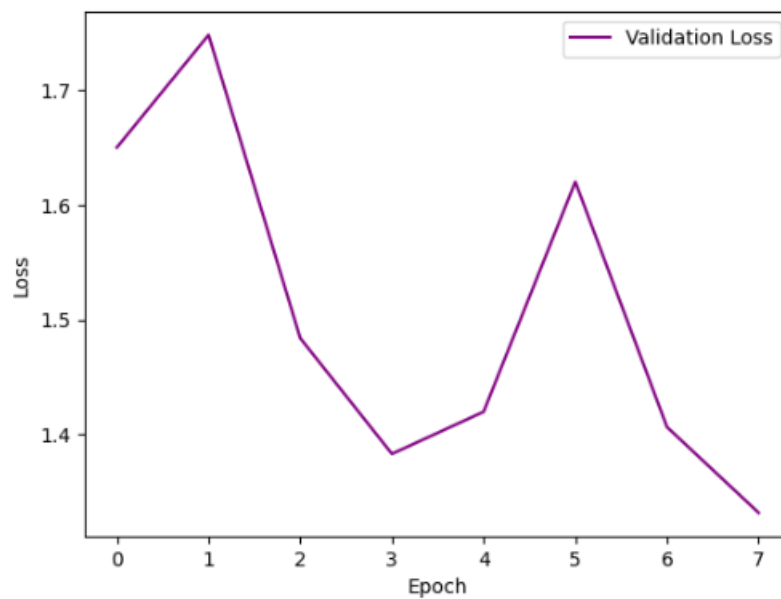
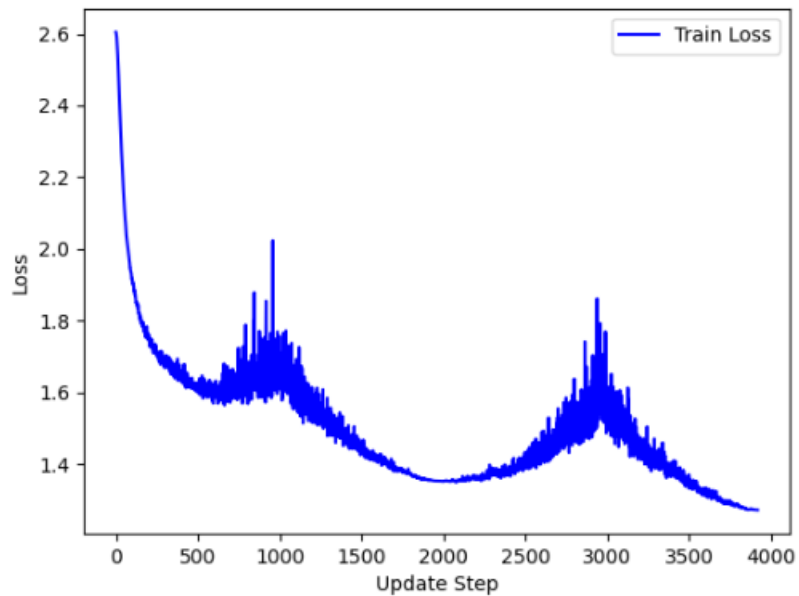


### Exercise5.2:

I set two different semi-extensive test and the below table shows the different accuracy on test accuracy:

Cycles = 2	200	300
1e-4	0.5187	<b>0.5289</b>
1e-3	0.5244	0.527
1e-2	0.4957	0.5011

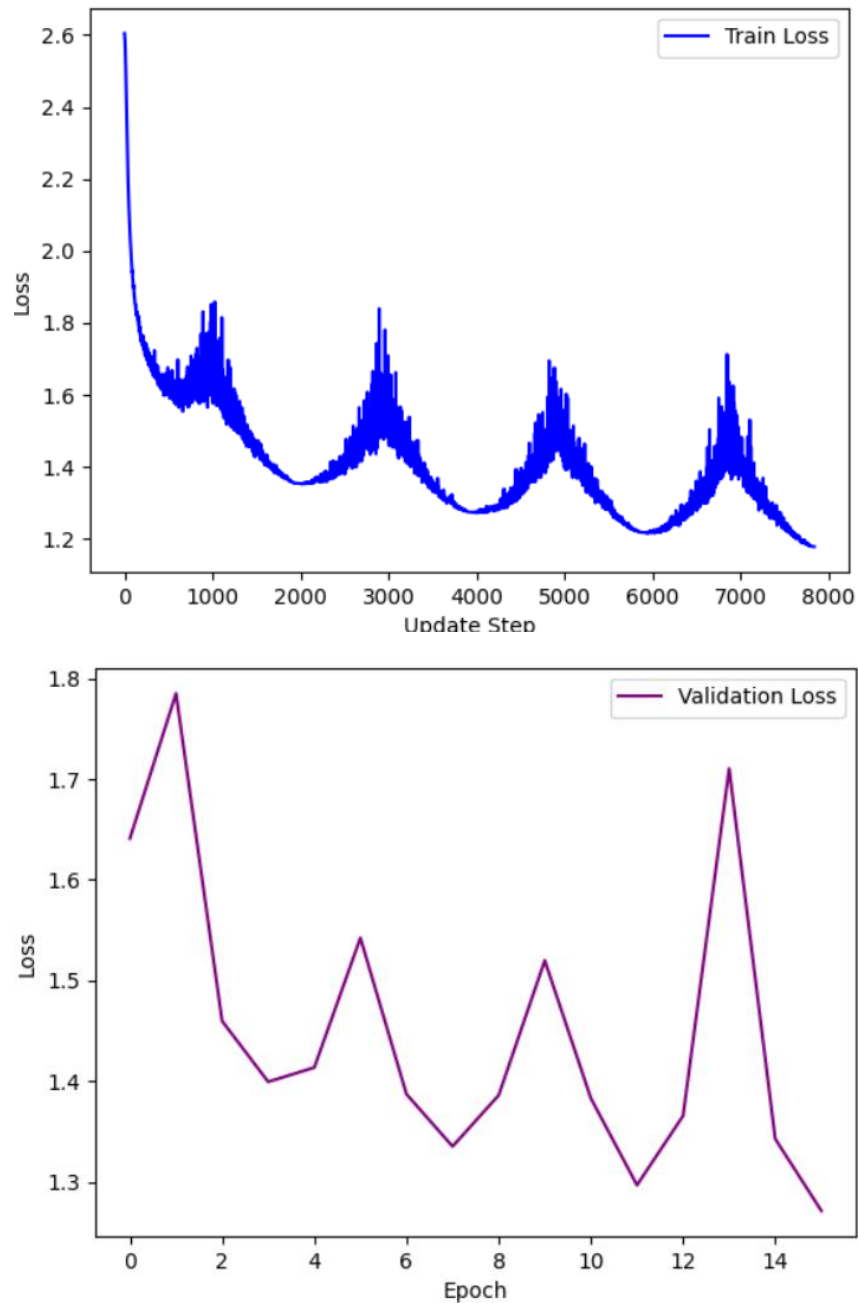
The highest test accuracy is **0.5289** with hidden nodes = **300**, lambda = **1e-4**, the loss on training and validation show:



At this time, I set the **Cycles = 4** and other parameters without any changed:

Cycles = 4	200	300
1e-4	0.5323	<b>0.5445</b>
1e-3	0.5327	0.5391
1e-2	0.4983	0.5035

The highest test accuracy is **0.5445** with hidden nodes = **300**, lambda = **1e-4**, the loss on training and validation show:



### Exercise5.3

According to the tutorial, I used the Adam optimizer to replace cyclical learning rate. As for the Adam, it is an adaptive learning rate optimizing algorithm which includes the momentum and RMSProp, the core function is these:

$$\begin{aligned} m_t &= \beta_1 m_{t-1} + (1 - \beta_1) g_t \\ v_t &= \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \\ m_t &= m_t / (1 - \beta_1^t), v_t = v_t / (1 - \beta_2^t) \\ \theta_t &= \theta_{t-1} - \alpha * m_t / v_t^{1/2} \end{aligned}$$

$m_t$ : The current momentum estimate

$m_{t-1}$ : The last time momentum estimate

$g_t$ : The current gradient

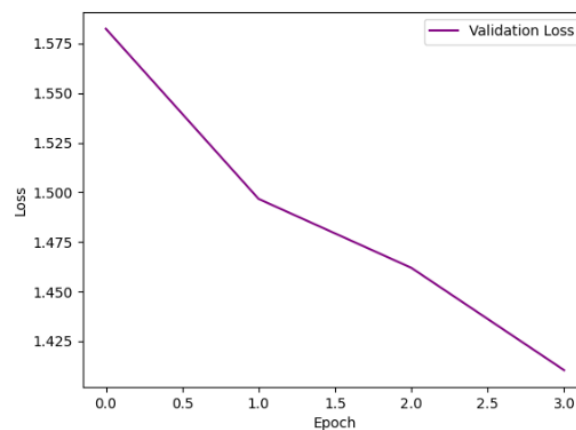
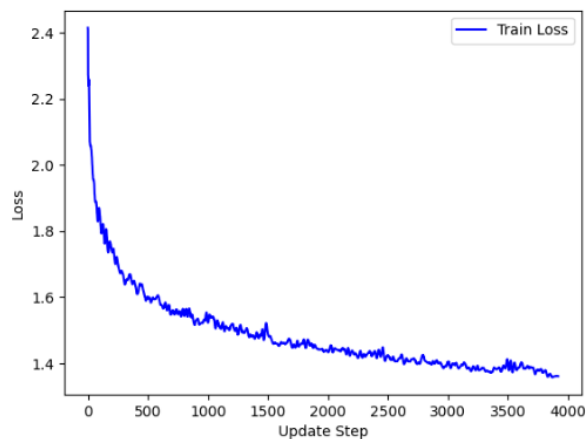
$v_{t-1}$ : The average estimate of the square of the gradient at the previous moment

The first function is momentum update, the second is RMSProp update, the third is correction bias and the last is parameter update.

The grid search of best hyperparameter for Adam:

Beta1,2	LR	1e-3	3e-4	1e-4
0.9, 0.99		0.4419	0.4851	0.4939
0.9, 0.999		0.469	0.4833	0.4864
0.95, 0.99		0.4453	<b>0.4978</b>	0.4865
0.95, 0.999		0.4639	0.4932	0.4892

The highest accuracy on test data is **0.4978**, with  $lr = 3e-4$ ,  $\beta_1 = 0.95$ ,  $\beta_2 = 0.99$



After that, I found the best hyperparameter and increase the epoch from **4 to 15** and the accuracy on test data is **0.5141** with the train and validation Loss curve:

