Machine Learning Assignment3

- For all the following sections, train the model for 50 epochs and plot the curve for loss, training accuracy, and test accuracy evaluated every epoch.

1. Run the tutorial code out of the box and make sure you get reasonable results. You will report these results in Section 4, so no report needed here.

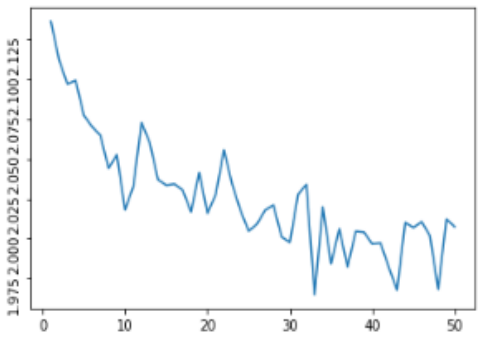
2. Change the code to have only a single fully connected layer. The model will have a single layer that connects the input to the output. What is the number of parameters? In PyTorch, "nn.Linear" can be used for fully connected layer

The model has a single layer that connects the input to the output layer. The code is changed to include a fully connected layer using nn.Linear. The shape(parameters) of this would be 32*32*3 (32 by 32 image and 3 channels). We use x.view(-1,32*32*3) to reshape data which is sent through a single neuron , which has parameters the weight vector and the bias. We get an Accuracy of 31% on test data without ReLu.

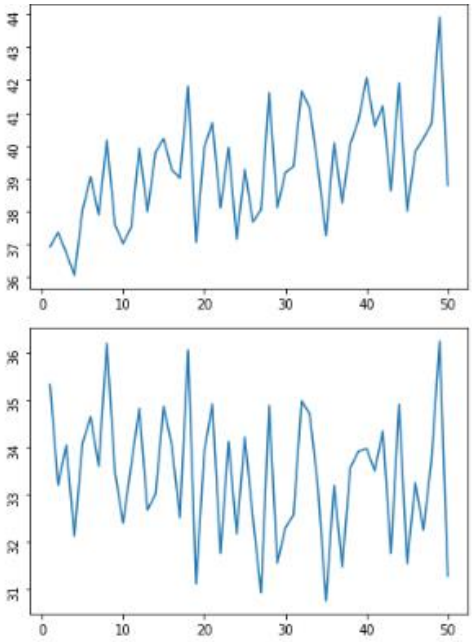
Below is the accuracy obtained at the end of the 50th epoch and also the predicted value against the actual value (GroundTruth) .Also printed is the accuracy of prediction of all the classes.

```
[50,  2000] loss: 1.920
[50,  4000] loss: 1.963
[50,  6000] loss: 1.964
[50,  8000] loss: 1.986
[50, 10000] loss: 2.001
[50, 12000] loss: 2.007
Accuracy on the train images: 38 %
Accuracy on the test images: 31 %
Finished Training
GroundTruth:    cat  ship  ship plane
Predicted:    cat truck plane plane
Accuracy of the network on the 10000 test images: 31 %
Accuracy of plane : 56 %
Accuracy of   car : 43 %
Accuracy of  bird : 24 %
Accuracy of   cat : 14 %
Accuracy of  deer : 35 %
Accuracy of   dog : 27 %
Accuracy of  frog : 17 %
Accuracy of horse : 29 %
Accuracy of  ship : 28 %
Accuracy of truck : 35 %
```

Plot of Loss(Y axis) Vs Epochs(X axis)



Plot of a)Accuracy on Training Set Vs Epochs b) Accuracy on Training Set Vs Epochs



3. Change the code to have multiple fully connected layers. Try having a layer from input to 120 neurons and then a layer to 84 neurons, and finally a layer to 10 neurons, one for each category. What happens if you do not use ReLU? Describe why.

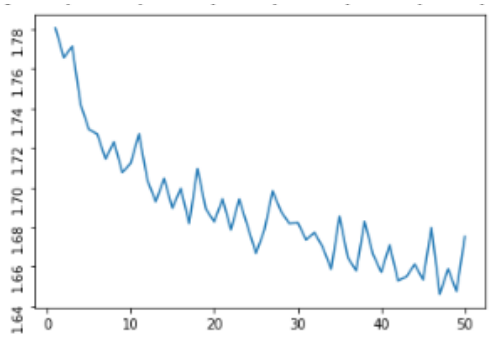
In this section We build 2 classes , one with ReLu and one without ReLu

Without ReLU

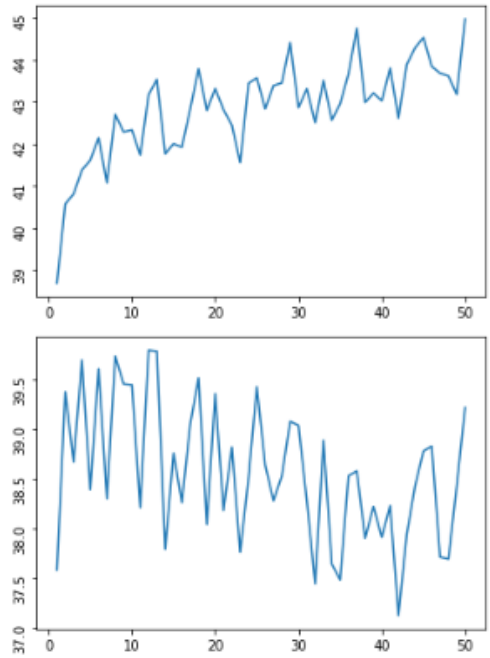
Below is the accuracy obtained at the end of the 50th epoch and also the predicted value against the actual value (GroundTruth) .Also printed is the accuracy of prediction of all the classes.

```
[50,  2000] loss: 1.634
[50,  4000] loss: 1.653
[50,  6000] loss: 1.651
[50,  8000] loss: 1.659
[50, 10000] loss: 1.672
[50, 12000] loss: 1.675
Accuracy on the train images: 44 %
Accuracy on the test images: 39 %
Finished Training
GroundTruth:    cat  ship  ship plane
Predicted:    cat truck  ship  ship
Accuracy of the network on the 10000 test images: 39 %
Accuracy of plane : 46 %
Accuracy of   car : 33 %
Accuracy of  bird : 29 %
Accuracy of   cat : 24 %
Accuracy of  deer : 24 %
Accuracy of   dog : 32 %
Accuracy of  frog : 42 %
Accuracy of horse : 50 %
Accuracy of  ship : 55 %
Accuracy of truck : 52 %
```

Plot of Loss(Y axis) Vs Epochs(X axis)



Plot of a)Accuracy on Training Set Vs Epochs b) Accuracy on Training Set Vs Epochs

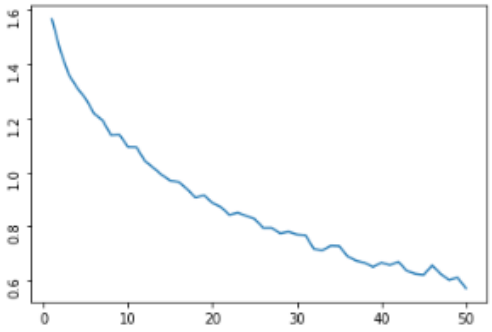


With ReLU

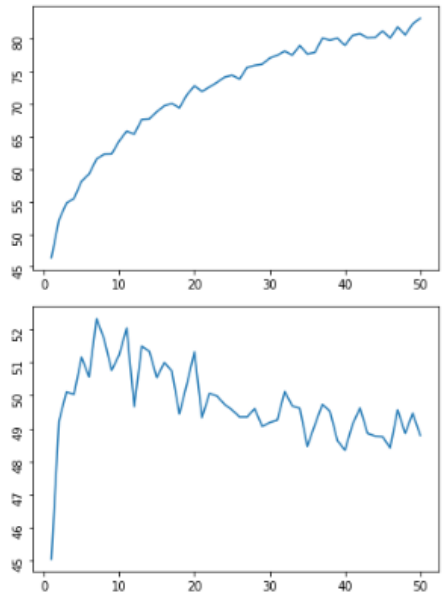
Below is the accuracy obtained at the end of the 50th epoch and also the predicted value against the actual value (GroundTruth) .Also printed is the accuracy of prediction of all the classes.

```
[50,  2000] loss: 0.502
[50,  4000] loss: 0.548
[50,  6000] loss: 0.568
[50,  8000] loss: 0.543
[50, 10000] loss: 0.573
[50, 12000] loss: 0.570
Accuracy on the train images: 83 %
Accuracy on the test images: 48 %
Finished Training
GroundTruth:    cat  ship  ship plane
Predicted:   bird  ship truck  deer
Accuracy of the network on the 10000 test images: 48 %
Accuracy of plane : 58 %
Accuracy of   car : 55 %
Accuracy of  bird : 42 %
Accuracy of   cat : 31 %
Accuracy of  deer : 39 %
Accuracy of   dog : 39 %
Accuracy of  frog : 55 %
Accuracy of horse : 52 %
Accuracy of  ship : 62 %
Accuracy of truck : 51 %
```

Plot of Loss(Y axis) Vs Epochs(X axis)



Plot of a)Accuracy on Training Set Vs Epochs b) Accuracy on Training Set Vs Epochs



We see that Without ReLU we have an Accuracy of 39% and With ReLU an accuracy of 48%. There is a significant increase in accuracy .This is because Neural networks need a non-linearity to be introduced in them, which is done by activation functions like ReLU(By calculating weighted sum and adding bias). Without the non-Linearity it would just be a linear regression model. ReLU does the non-Linear transformation to the input making it capable to learn and perform complex tasks. We update the weights and bias based on the error(Back Propagation) and the activation makes this possible by giving the gradients the error to update the weights and bias. When this is not done the model just becomes less powerful like a linear model.

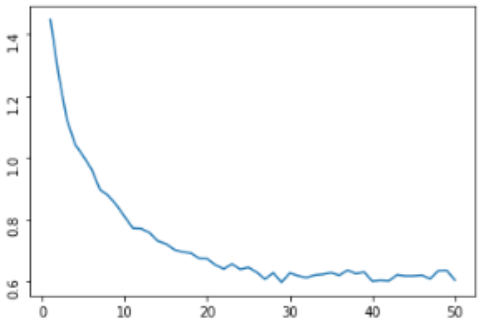
4. Change the code by adding two convolutional layers along with maxpooling layers before the fully connected layers. This will be like the example in the tutorial. Use this model for the following sections.

This model uses 2 convolutional Layers, Maxpooling Layers (fully connected). The loss function used is Cross Entropy with a batch size of 4 and Learning Rate 0.001 with ReLu. An accuracy of 60% on Test Data is Obtained

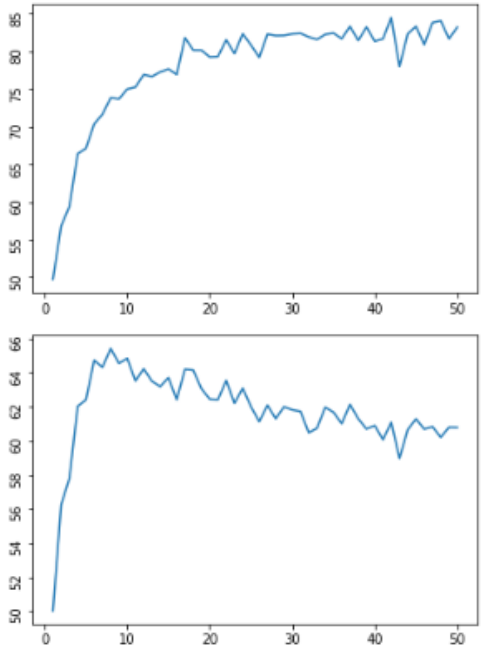
Below is the accuracy obtained at the end of the 50th epoch and also the predicted value against the actual value (GroundTruth) .Also printed is the accuracy of prediction of all the classes.

```
[50,  2000] loss: 0.512
[50,  4000] loss: 0.515
[50,  6000] loss: 0.556
[50,  8000] loss: 0.576
[50, 10000] loss: 0.613
[50, 12000] loss: 0.604
Accuracy on the train images: 83 %
Accuracy on the test images: 60 %
Finished Training
GroundTruth:    cat  ship  ship plane
Predicted:  horse truck   car plane
Accuracy of the network on the 10000 test images: 60 %
Accuracy of plane : 56 %
Accuracy of   car : 77 %
Accuracy of  bird : 51 %
Accuracy of   cat : 38 %
Accuracy of  deer : 57 %
Accuracy of   dog : 43 %
Accuracy of  frog : 67 %
Accuracy of horse : 73 %
Accuracy of  ship : 74 %
Accuracy of truck : 68 %
```

Plot of Loss(Y axis) Vs Epochs(X axis)



Plot of a)Accuracy on Training Set Vs Epochs b) Accuracy on Training Set Vs Epochs



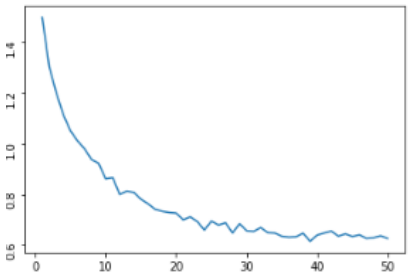
5. Try multiple batch sizes to see the effect and describe the findings. Please use batch size of 1, 4, and 1000. If 1000 does not fit into the memory of your machine, please feel free to reduce it to a largest possible number.

Batch Size 1

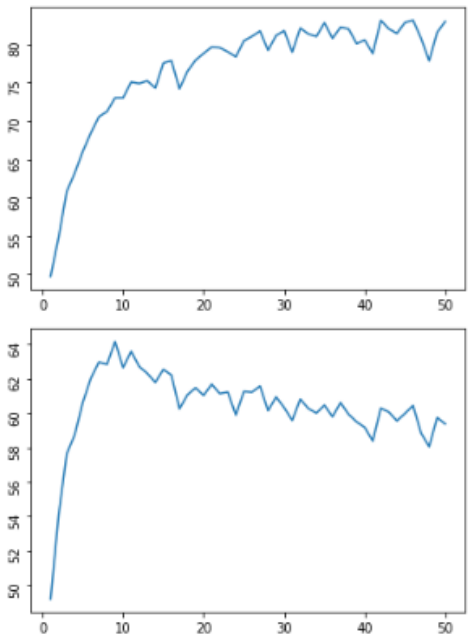
Below is the accuracy obtained at the end of the 50th epoch and also the predicted value against the actual value (GroundTruth) .Also printed is the accuracy of prediction of all the classes.

```
Accuracy on the test images: 59 %
[50,  2000] loss: 0.500
[50,  4000] loss: 0.576
[50,  6000] loss: 0.613
[50,  8000] loss: 0.627
[50, 10000] loss: 0.643
[50, 12000] loss: 0.627
Accuracy on the train images: 82 %
Accuracy on the test images: 59 %
Finished Training
GroundTruth:    cat  ship  ship plane
Predicted:    cat  ship  ship plane
Accuracy of the network on the 10000 test images: 59 %
Accuracy of plane : 64 %
Accuracy of   car : 74 %
Accuracy of  bird : 47 %
Accuracy of   cat : 42 %
Accuracy of  deer : 50 %
Accuracy of   dog : 43 %
Accuracy of  frog : 73 %
Accuracy of horse : 65 %
Accuracy of  ship : 65 %
Accuracy of truck : 67 %
```

Plot of Loss(Y axis) Vs Epochs(X axis)



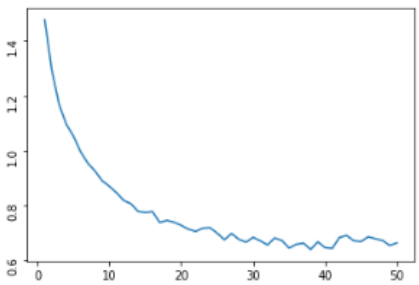
Plot of a)Accuracy on Training Set Vs Epochs b) Accuracy on Training Set Vs Epochs



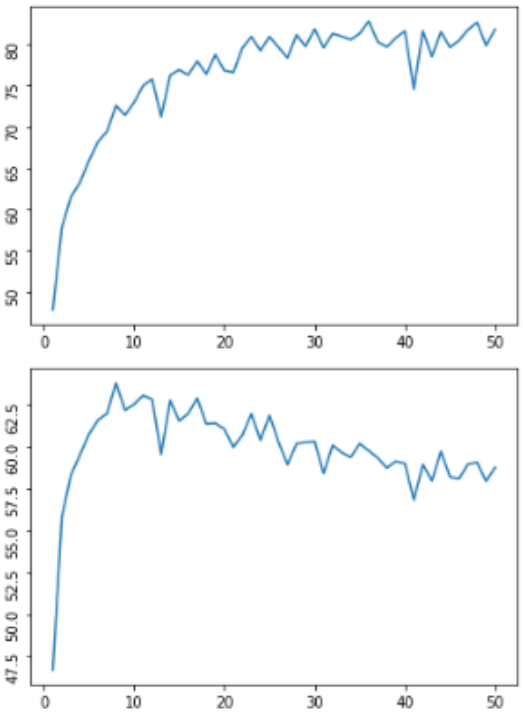
Batch Size 4

```
[50,  2000] loss: 0.552
[50,  4000] loss: 0.626
[50,  6000] loss: 0.631
[50,  8000] loss: 0.680
[50, 10000] loss: 0.650
[50, 12000] loss: 0.664
Accuracy on the train images: 81 %
Accuracy on the test images: 58 %
Finished Training
GroundTruth:    cat  ship  ship plane
Predicted:    cat   car  ship  ship
Accuracy of the network on the 10000 test images: 58 %
Accuracy of plane : 63 %
Accuracy of   car : 77 %
Accuracy of  bird : 44 %
Accuracy of   cat : 44 %
Accuracy of  deer : 56 %
Accuracy of   dog : 43 %
Accuracy of  frog : 61 %
Accuracy of horse : 60 %
Accuracy of  ship : 65 %
Accuracy of truck : 71 %
```

Plot of Loss(Y axis) Vs Epochs(X axis)



Plot of a)Accuracy on Training Set Vs Epochs b) Accuracy on Training Set Vs Epochs

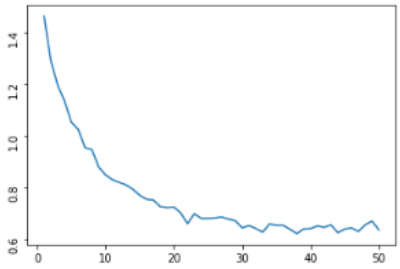


Batch Size 1000

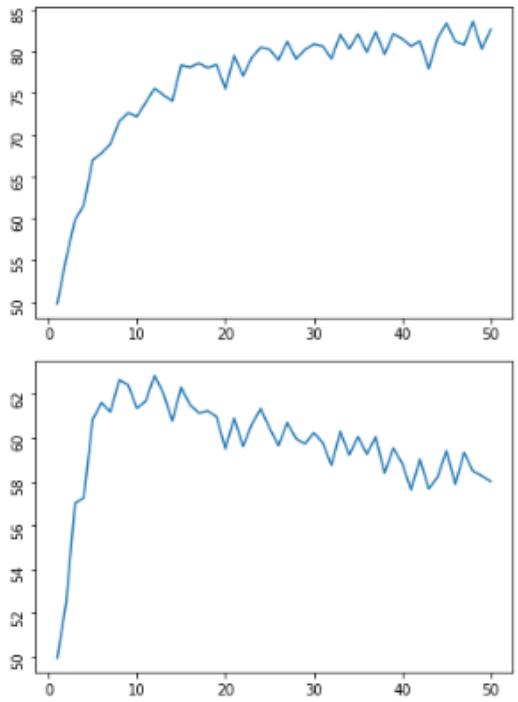
Below is the accuracy obtained at the end of the 50th epoch and the predicted value against the actual value (GroundTruth) .Also printed is the accuracy of prediction of all the classes.

```
[50,  2000] loss: 0.532
[50,  4000] loss: 0.595
[50,  6000] loss: 0.604
[50,  8000] loss: 0.594
[50, 10000] loss: 0.636
[50, 12000] loss: 0.638
Accuracy on the train images: 82 %
Accuracy on the test images: 58 %
Finished Training
GroundTruth:    cat  ship  ship plane
Predicted:  truck plane  ship plane
Accuracy of the network on the 10000 test images: 58 %
Accuracy of plane : 64 %
Accuracy of   car : 72 %
Accuracy of  bird : 42 %
Accuracy of   cat : 37 %
Accuracy of  deer : 50 %
Accuracy of   dog : 57 %
Accuracy of  frog : 65 %
Accuracy of horse : 61 %
Accuracy of  ship : 65 %
Accuracy of truck : 62 %
```

Plot of Loss(Y axis) Vs Epochs(X axis)



Plot of a)Accuracy on Training Set Vs Epochs b) Accuracy on Training Set Vs Epochs



The model is run on different batch sizes. For batch size 1 the complete network including the loss function, optimizer and activation function is run on every data point . Then we used Batch size 4 and then 1000. Both 4 and 1000 almost gave the same accuracy but the training time using 4 was much larger than that using 4. Accuracy using batch size 1 was higher than that with 4 and 1000 but we still would not prefer that because the learning time is very high compared to other batch sizes. The gradient descent optimizer would run faster with higher batch size. With smaller batch size, the updates to weights and bias would be more frequent making the learning algorithm take longer time. Picking the right learning rate and batch size is the key to getting the most accurate results.

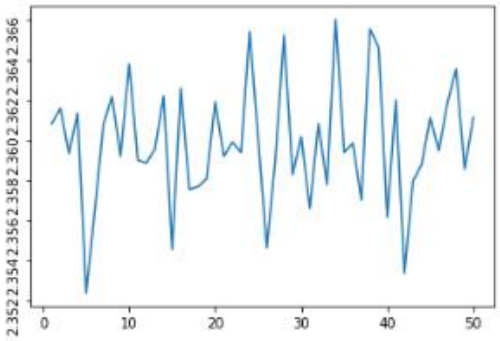
6. Try multiple learning rates to see the effect and describe the findings. Please use learning rates of 10, 0.1, 0.01, and 0.0001.

Below is the accuracy obtained at the end of the 50th epoch and also the predicted value against the actual value (GroundTruth) .Also printed is the accuracy of prediction of all the classes.

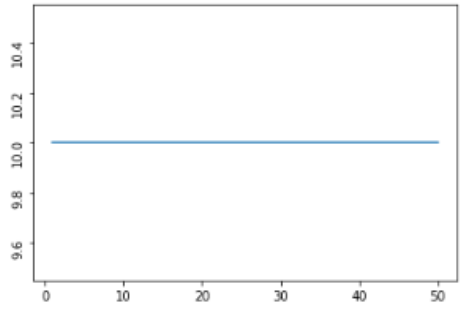
Learning Rate:0.1

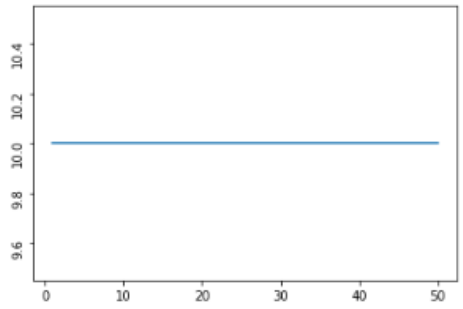
```
[50,  2000] loss: 2.357
[50,  4000] loss: 2.357
[50,  6000] loss: 2.359
[50,  8000] loss: 2.362
[50, 10000] loss: 2.354
[50, 12000] loss: 2.361
Accuracy on the train images: 10 %
Accuracy on the test images: 10 %
Finished Training
GroundTruth:    cat  ship  ship plane
Predicted:  plane plane plane plane
Accuracy of the network on the 10000 test images: 10 %
Accuracy of plane : 100 %
Accuracy of   car :  0 %
Accuracy of  bird :  0 %
Accuracy of   cat :  0 %
Accuracy of  deer :  0 %
Accuracy of   dog :  0 %
Accuracy of  frog :  0 %
Accuracy of horse :  0 %
Accuracy of  ship :  0 %
Accuracy of truck :  0 %
```

Plot of Loss(Y axis) Vs Epochs(X axis)



Plot of a)Accuracy on Training Set Vs Epochs b) Accuracy on Training Set Vs Epochs

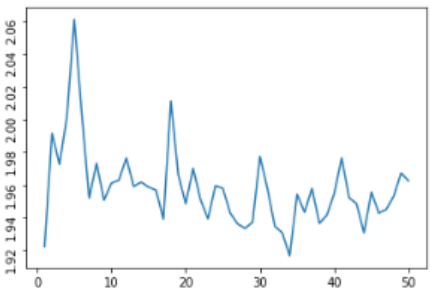




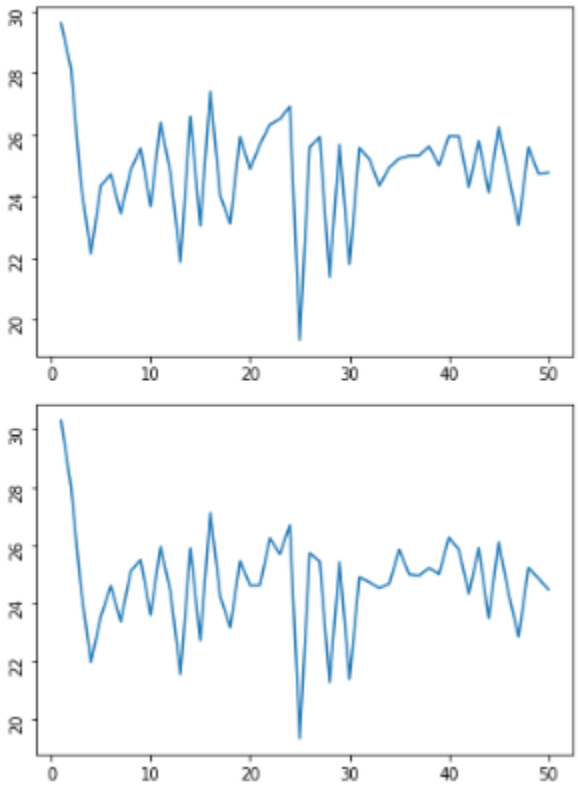
Learning Rate:0.01

```
[50,  2000] loss: 1.963
[50,  4000] loss: 1.959
[50,  6000] loss: 1.946
[50,  8000] loss: 1.953
[50, 10000] loss: 1.988
[50, 12000] loss: 1.962
Accuracy on the train images: 24 %
Accuracy on the test images: 24 %
Finished Training
GroundTruth:    cat  ship  ship plane
Predicted:  plane plane plane plane
Accuracy of the network on the 10000 test images: 24 %
Accuracy of plane : 59 %
Accuracy of   car : 21 %
Accuracy of  bird :  0 %
Accuracy of   cat : 15 %
Accuracy of  deer :  0 %
Accuracy of   dog : 19 %
Accuracy of  frog : 68 %
Accuracy of horse : 28 %
Accuracy of  ship :  1 %
Accuracy of truck : 30 %
```

Plot of Loss(Y axis) Vs Epochs(X axis)



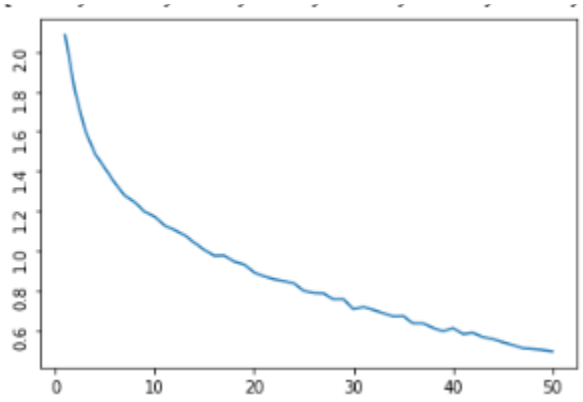
Plot of a)Accuracy on Training Set Vs Epochs b) Accuracy on Training Set Vs Epochs



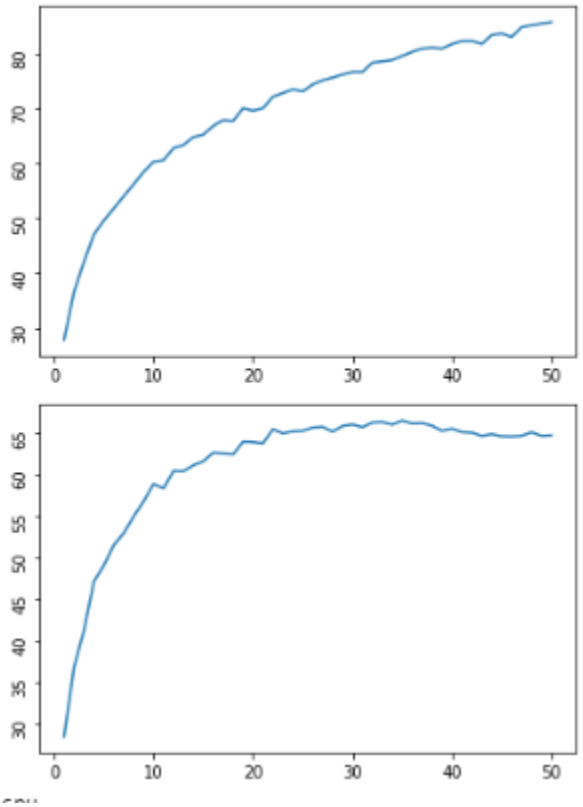
Learning Rate:0.0001

```
[50,  2000] loss: 0.426
[50,  4000] loss: 0.438
[50,  6000] loss: 0.450
[50,  8000] loss: 0.460
[50, 10000] loss: 0.477
[50, 12000] loss: 0.492
Accuracy on the train images: 85 %
Accuracy on the test images: 64 %
Finished Training
GroundTruth:    cat  ship  ship plane
Predicted:    cat  ship  ship plane
Accuracy of the network on the 10000 test images: 64 %
Accuracy of plane : 74 %
Accuracy of   car : 75 %
Accuracy of  bird : 48 %
Accuracy of   cat : 57 %
Accuracy of  deer : 62 %
Accuracy of   dog : 43 %
Accuracy of  frog : 74 %
Accuracy of horse : 69 %
Accuracy of  ship : 71 %
Accuracy of truck : 69 %
```

Plot of Loss(Y axis) Vs Epochs(X axis)



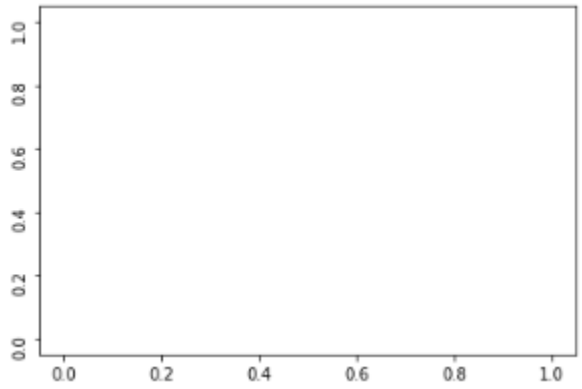
Plot of a)Accuracy on Training Set Vs Epochs b) Accuracy on Training Set Vs Epochs



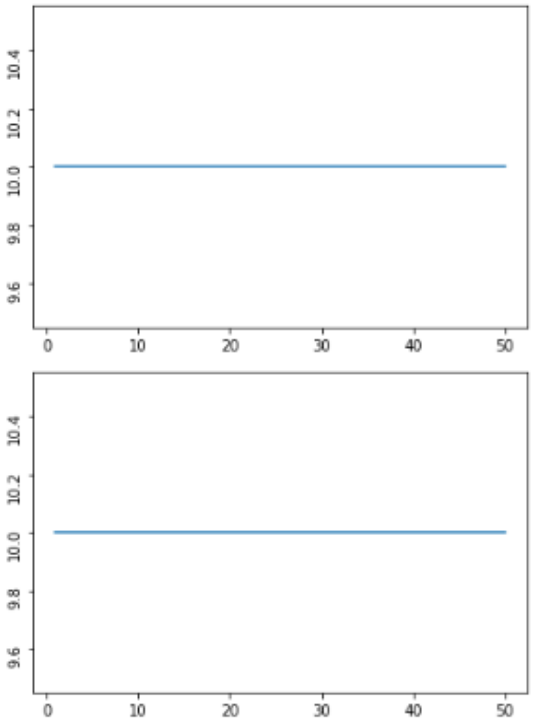
Learning Rate:10

```
[50,  2000] loss: nan
[50,  4000] loss: nan
[50,  6000] loss: nan
[50,  8000] loss: nan
[50, 10000] loss: nan
[50, 12000] loss: nan
Accuracy on the train images: 10 %
Accuracy on the test images: 10 %
Finished Training
GroundTruth:    cat  ship  ship plane
Predicted:  plane plane plane plane
Accuracy of the network on the 10000 test images: 10 %
Accuracy of plane : 100 %
Accuracy of   car :  0 %
Accuracy of  bird :  0 %
Accuracy of   cat :  0 %
Accuracy of  deer :  0 %
Accuracy of   dog :  0 %
Accuracy of  frog :  0 %
Accuracy of horse :  0 %
Accuracy of  ship :  0 %
Accuracy of truck :  0 %
```

Plot of Loss(Y axis) Vs Epochs(X axis)



Plot of a)Accuracy on Training Set Vs Epochs b) Accuracy on Training Set Vs Epochs



We see effect of learning rate on accuracy and loss at batch size 4. We see that the best among these four learning rates is 0.0001 , the learning curve is more stable compared to the rest of the rates, the loss is also reducing gradually unlike fluctuation in other rates It controls the speed at which the model learns. Small rates let to learn maybe a global optimum but at the cost of learning time. At very low learning rate the training is better, but the time taken would be extremely high. On the other hand if the learning rate is very high, the training of the model is faster but there are high chances of jumping from one section to the other without reaching the minima. The model could land up in a situation where it would never converge or diverge like the one above for learning rate 10.

7. Please add some data augmentation to avoid overfitting. Note that you need to do this only for the training and not the testing. You may use line 208 from Imagenet sample code: https://github.com/pytorch/examples/blob/master/imagenet/main.py ”RandomResizedCrop” samples a random patch from the image to train the model on. ”RandomHorizontalFlip” flips randomly chosen images horizontally.

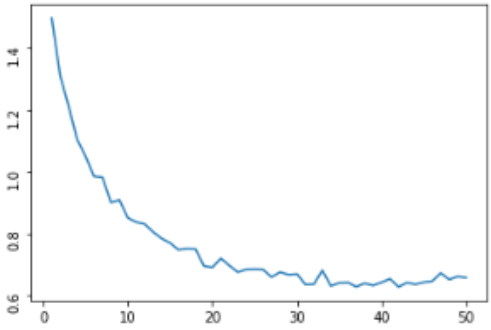
Below is the accuracy obtained at the end of the 50th epoch and also the predicted value against the actual value (GroundTruth) .Also printed is the accuracy of prediction of all the classes.

We added Data Augmentation by resizing the data **”RandomResizedCrop”** and by doing a horizontal flip **”RandomHorizontalFlip”.** We do this to avoid Overfitting. It picks sections from the image and does a

horizontal flip creating a different kind of image to train on. This is to avoid the model from over learning the data set

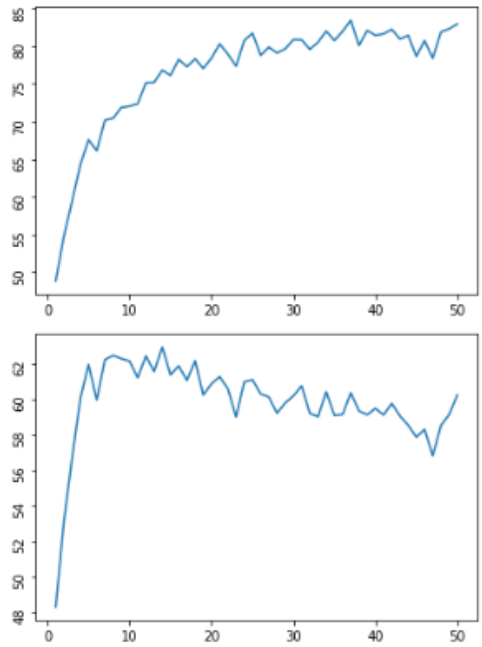
```
[50,  2000] loss: 0.543
[50,  4000] loss: 0.589
[50,  6000] loss: 0.611
[50,  8000] loss: 0.620
[50, 10000] loss: 0.615
[50, 12000] loss: 0.657
Accuracy on the train images: 82 %
Accuracy on the test images: 60 %
Finished Training
GroundTruth:    cat  ship  ship plane
Predicted:   frog plane  ship plane
Accuracy of the network on the 10000 test images: 60 %
Accuracy of plane : 72 %
Accuracy of   car : 77 %
Accuracy of  bird : 47 %
Accuracy of   cat : 38 %
Accuracy of  deer : 52 %
Accuracy of   dog : 43 %
Accuracy of  frog : 72 %
Accuracy of horse : 61 %
Accuracy of  ship : 73 %
Accuracy of truck : 62 %
```

Plot of Loss(Y axis) Vs Epochs(X axis)



Plot of

a)Accuracy on Training Set Vs Epochs b) Accuracy on Training Set Vs Epochs

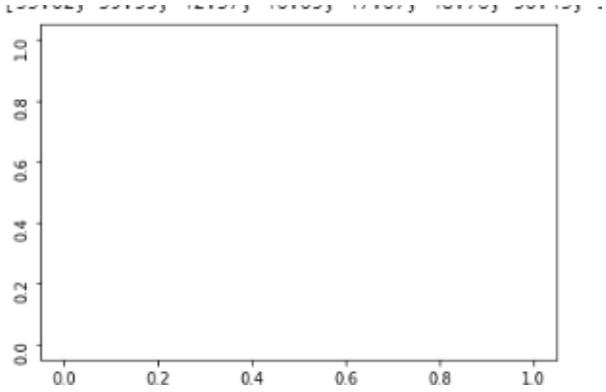


8. Change the loss function from Cross Entropy to Mean Squared Error and report the effect.

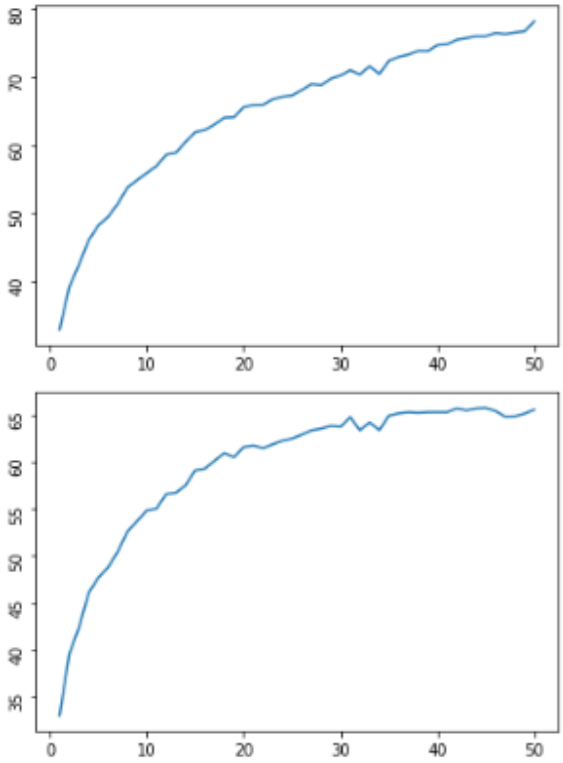
Below is the accuracy obtained at the end of the 50th epoch and also the predicted value against the actual value (GroundTruth) .Also printed is the accuracy of prediction of all the classes.

```
GroundTruth:    cat  ship  ship plane
Predicted:    cat  ship  ship plane
Accuracy of the network on the 10000 test images: 65 %
Accuracy of plane : 67 %
Accuracy of   car : 79 %
Accuracy of  bird : 55 %
Accuracy of   cat : 41 %
Accuracy of  deer : 61 %
Accuracy of   dog : 51 %
Accuracy of  frog : 76 %
Accuracy of horse : 68 %
Accuracy of  ship : 81 %
Accuracy of truck : 73 %
```

Plot of Loss(Y axis) Vs Epochs(X axis)



Plot of a)Accuracy on Training Set Vs Epochs b) Accuracy on Training Set Vs Epochs



We see that MSE works just fine for this neural network but when compared with cross entropy (Question4 graphs), we see that the loss reduces faster in cross entropy in early stage compared to MSE. MSE also took longer to compute compared to cross entropy loss. The model with cross entropy learns faster through gradient descent compared to MSE because of non linear activation function used. In MSE the derivative has the term σ'(Wxi+b). the partial derivative vanishes during initial stages making the learning with gradient descent slow in the beginning stages. Whereas in cross entropy , the derivative does not have the σ' term , the magnitude of the derivative is dependent on magnitude of error. This also means in the early learning stages the derivatives will be large, smaller in the later stages . This is better as when the error is small, we make no large adjustments which could make us move very far away from minima.