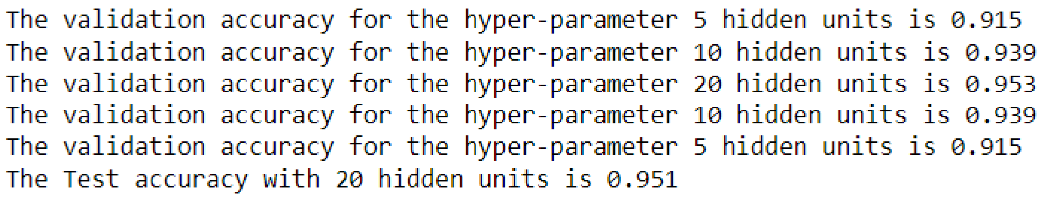
1.(a)

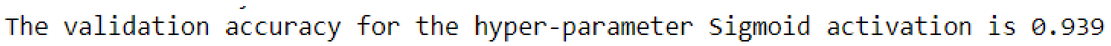


```
The validation accuracy for the hyper-parameter 5 hidden units is 0.915
The validation accuracy for the hyper-parameter 10 hidden units is 0.939
The validation accuracy for the hyper-parameter 20 hidden units is 0.953
The validation accuracy for the hyper-parameter 10 hidden units is 0.939
The validation accuracy for the hyper-parameter 5 hidden units is 0.915
The Test accuracy with 20 hidden units is 0.951
```

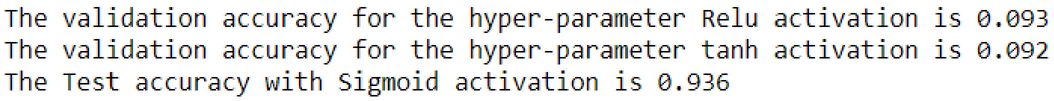
Therefore, the best performance is obtained with 20 units in 1 hidden layer.

The accuracy of the test set with 20 units in 1 hidden layer is 0.951.

1.(b)



```
The validation accuracy for the hyper-parameter Sigmoid activation is 0.939
```



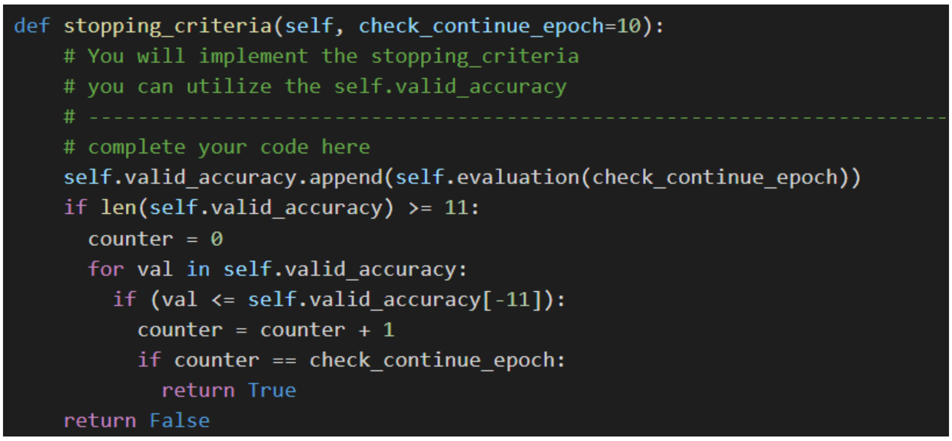
```
The validation accuracy for the hyper-parameter Relu activation is 0.093
The validation accuracy for the hyper-parameter tanh activation is 0.092
The Test accuracy with Sigmoid activation is 0.936
```

Therefore, the best performance is obtained with Sigmoid Activation function.

And the corresponding test accuracy with Sigmoid Activation function is 0.936.

Although, the validation accuracy using Relu, and Tanh activation function should not have been this less. The Relu and Tanh activation functions defined by me was having a problem with the value while calculating exponent as the value was getting very large and it was showing warnings.

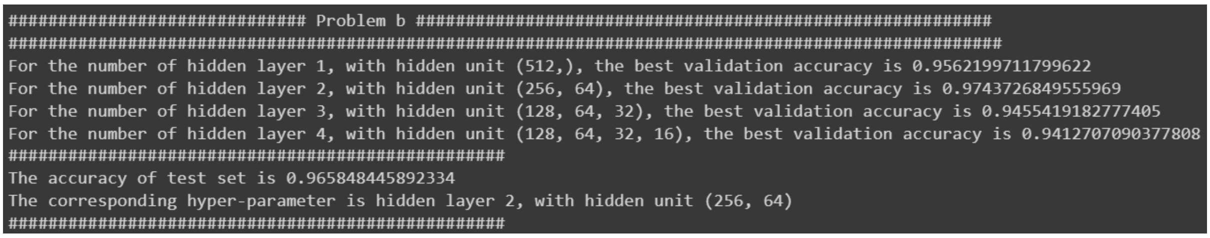
2.(a)



```
def stopping_criteria(self, check_continue_epoch=10):
    # You will implement the stopping_criteria
    # you can utilize the self.valid_accuracy
    # -----------------------------------------------------------------------------
    # complete your code here
    self.valid_accuracy.append(self.evaluation(check_continue_epoch))
    if len(self.valid_accuracy) >= 11:
      counter = 0
      for val in self.valid_accuracy:
        if (val <= self.valid_accuracy[-11]):
          counter = counter + 1
          if counter == check_continue_epoch:
            return True
    return False
```

When the length of the array valid_accuracy is becoming more than 11, we are comparing the value of the 11th element from the last with the last ten elements and increasing the value of the counter variable with 1. If the value of the counter variable equals the check_continue_epoch (=10) which checks that the accuracy of the validation set has not been improved in 10 consecutive epochs, the function returns a True value which ultimately terminates the program from learning further.

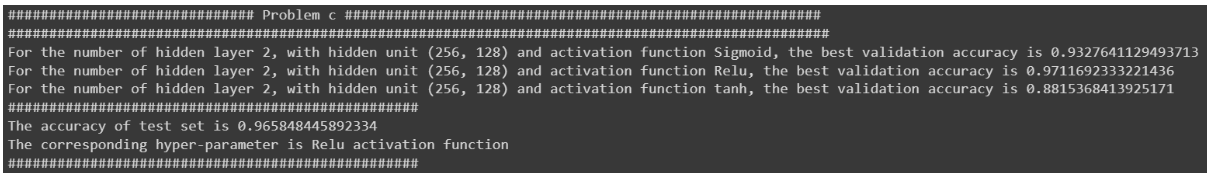
2. (b)



```
############################## Problem b ##########################################################
###################################################################################################
For the number of hidden layer 1, with hidden unit (512,), the best validation accuracy is 0.9562199711799622
For the number of hidden layer 2, with hidden unit (256, 64), the best validation accuracy is 0.9743726849555969
For the number of hidden layer 3, with hidden unit (128, 64, 32), the best validation accuracy is 0.9455419182777405
For the number of hidden layer 4, with hidden unit (128, 64, 32, 16), the best validation accuracy is 0.9412707090377808
##################################################
The accuracy of test set is 0.965848445892334
The corresponding hyper-parameter is hidden layer 2, with hidden unit (256, 64)
##################################################
```

For 1 hidden layer, with hidden unit (512,), the best validation accuracy is 0.956219.
For 2 hidden layers, with hidden unit (256, 64), the best validation accuracy is 0.974373.
For 3 hidden layers, with hidden unit (128, 64, 32), the best validation accuracy is 0.945542.
For 4 hidden layers, with hidden unit (128, 64, 32, 16), the best validation accuracy is 0.941271.
Therefore, the best performance is obtained with 2 hidden layers. And the best accuracy of test set is 0.965848 corresponding to hyper-parameter 2 hidden layers, with hidden unit (256, 64).

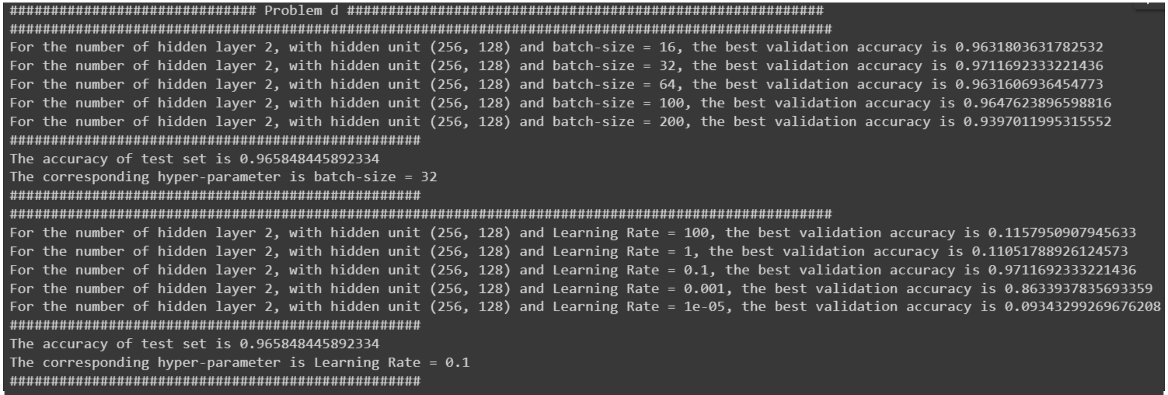
2. (c)



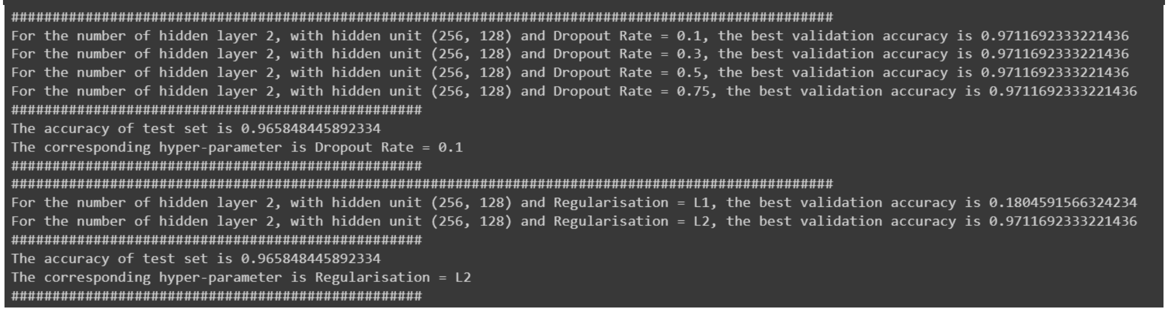
```
############################## Problem c ##########################################################
###################################################################################################
For the number of hidden layer 2, with hidden unit (256, 128) and activation function Sigmoid, the best validation accuracy is 0.9327641129493713
For the number of hidden layer 2, with hidden unit (256, 128) and activation function Relu, the best validation accuracy is 0.9711692333221436
For the number of hidden layer 2, with hidden unit (256, 128) and activation function tanh, the best validation accuracy is 0.8815368413925171
##################################################
The accuracy of test set is 0.965848445892334
The corresponding hyper-parameter is Relu activation function
##################################################
```

For 2 hidden layers, with hidden unit (256, 128) and activation function Sigmoid, the best validation accuracy is 0.932764.
For 2 hidden layers, with hidden unit (256, 128) and activation function Relu, the best validation accuracy is 0.971169.
For 2 hidden layers, with hidden unit (256, 128) and activation function tanh, the best validation accuracy is 0.881537.
Therefore, the best performance is obtained with Relu activation function. And the best accuracy of test set is 0.965848 corresponding hyper-parameter activation function - Relu.

2.(d)



```
############################## Problem d ##########################################################
####################################################################################################
For the number of hidden layer 2, with hidden unit (256, 128) and batch-size = 16, the best validation accuracy is 0.9631803631782532
For the number of hidden layer 2, with hidden unit (256, 128) and batch-size = 32, the best validation accuracy is 0.9711692333221436
For the number of hidden layer 2, with hidden unit (256, 128) and batch-size = 64, the best validation accuracy is 0.963160693645477З
For the number of hidden layer 2, with hidden unit (256, 128) and batch-size = 100, the best validation accuracy is 0.9647623896598816
For the number of hidden layer 2, with hidden unit (256, 128) and batch-size = 200, the best validation accuracy is 0.9397011995315552
##################################################
The accuracy of test set is 0.965848445892334
The corresponding hyper-parameter is batch-size = 32
##################################################
####################################################################################################
For the number of hidden layer 2, with hidden unit (256, 128) and Learning Rate = 100, the best validation accuracy is 0.1157950907945633
For the number of hidden layer 2, with hidden unit (256, 128) and Learning Rate = 1, the best validation accuracy is 0.11051788926124573
For the number of hidden layer 2, with hidden unit (256, 128) and Learning Rate = 0.1, the best validation accuracy is 0.9711692333221436
For the number of hidden layer 2, with hidden unit (256, 128) and Learning Rate = 0.001, the best validation accuracy is 0.8633937835693359
For the number of hidden layer 2, with hidden unit (256, 128) and Learning Rate = 1e-05, the best validation accuracy is 0.09343299269676208
##################################################
The accuracy of test set is 0.965848445892334
The corresponding hyper-parameter is Learning Rate = 0.1
##################################################
```



```
####################################################################################################
For the number of hidden layer 2, with hidden unit (256, 128) and Dropout Rate = 0.1, the best validation accuracy is 0.9711692333221436
For the number of hidden layer 2, with hidden unit (256, 128) and Dropout Rate = 0.3, the best validation accuracy is 0.9711692333221436
For the number of hidden layer 2, with hidden unit (256, 128) and Dropout Rate = 0.5, the best validation accuracy is 0.9711692333221436
For the number of hidden layer 2, with hidden unit (256, 128) and Dropout Rate = 0.75, the best validation accuracy is 0.9711692333221436
##################################################
The accuracy of test set is 0.965848445892334
The corresponding hyper-parameter is Dropout Rate = 0.1
##################################################
####################################################################################################
For the number of hidden layer 2, with hidden unit (256, 128) and Regularisation = L1, the best validation accuracy is 0.1804591566324234
For the number of hidden layer 2, with hidden unit (256, 128) and Regularisation = L2, the best validation accuracy is 0.9711692333221436
##################################################
The accuracy of test set is 0.965848445892334
The corresponding hyper-parameter is Regularisation = L2
##################################################
```

Changing the batch size:

For 2 hidden layers, with hidden unit (256, 128) and batch-size = 16, the best validation accuracy is 0.963180
For 2 hidden layers, with hidden unit (256, 128) and batch-size = 32, the best validation accuracy is 0.971169
For 2 hidden layers, with hidden unit (256, 128) and batch-size = 64, the best validation accuracy is 0.963161
For 2 hidden layers, with hidden unit (256, 128) and batch-size = 100, the best validation accuracy is 0.964762
For 2 hidden layers, with hidden unit (256, 128) and batch-size = 200, the best validation accuracy is 0.939701
Therefore, the accuracy of test set is 0.965848 and the corresponding hyperparameter batch-size best value is 32

Changing the Learning Rate:

For 2 hidden layers, with hidden unit (256, 128) and Learning Rate = 100, the best validation accuracy is 0.115795
For 2 hidden layers, with hidden unit (256, 128) and Learning Rate = 1, the best validation accuracy is 0.110518
For 2 hidden layers, with hidden unit (256, 128) and Learning Rate = 0.1, the best validation accuracy is 0.971169
For 2 hidden layers, with hidden unit (256, 128) and Learning Rate = 0.001, the best validation accuracy is 0.8633938
For 2 hidden layers, with hidden unit (256, 128) and Learning Rate = 1e-05, the best validation accuracy is 0.0934330
Therefore, the accuracy of test set is 0.965848 and the corresponding hyper-parameter learning rate best value is 0.1.

Changing the Dropout:

For 2 hidden layers, with hidden unit (256, 128) and Dropout Rate = 0.1, the best validation accuracy is 0.971169
For 2 hidden layers, with hidden unit (256, 128) and Dropout Rate = 0.3, the best validation accuracy is 0.971169
For 2 hidden layers, with hidden unit (256, 128) and Dropout Rate = 0.5, the best validation accuracy is 0.971169
For 2 hidden layers, with hidden unit (256, 128) and Dropout Rate = 0.75, the best validation accuracy is 0.971169
Therefore, the accuracy of test set is 0.965848 and the corresponding hyper-parameter dropout best value is 0.1.

Changing the L1/L2 Regularisation:

For 2 hidden layers, with hidden unit (256, 128) and Regularisation = L1, the best validation accuracy is 0.180459
For 2 hidden layers, with hidden unit (256, 128) and Regularisation = L2, the best validation accuracy is 0.971169
Therefore, the accuracy of test set is 0.965848 and the corresponding best hyper-parameter regularisation is L2 Regularisation.

Increasing the number of nodes or the number of hidden layers does not necessarily increase the accuracy of the model, while it may increase accuracy in the training data, there might be too much over-fitting. Accuracy with respect to activation functions depend on the type of application. The hyperparameter batch size is maximum at some middle order, lowering or increasing the batch size increases or decreases its accuracy value respectively. Increasing the Learning rate too much, the model might be unstable while decreasing the learning rate might increase the processing time considerably. And lastly, for this particular application L2 Regularisation seem much more feasible.

3.(a) The prediction accuracy with the selected best hyperparameter is 68%.

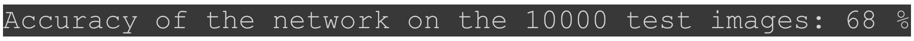
3.(b) Firstly, I changed the kernel size and the number of kernels in both the convolution layer 1 and convolution layer 2. Kernel size of 2x2 would give better results with respect to convolution performed with kernel size 5x5. The number of kernels in first convolution was increased from 6 to 12 and on the second layer from 12 to 24. I tried using three convolution layers, but the loss kept increasing after reaching a certain minimum value.

Secondly, I changed the number of hidden layers and the number of hidden units in all these layers. The output from the convolution had 1176 outputs, and thus I reduced the number of hidden units in respective layers approximately by factors of 2, (1176,512) – (512,256) – (256,128) – (128, 64) – (64, 10) so that there is no drastic reduction in the number of weights from one layer to the next which sometimes makes the model hard to train.

The activation function between each of these layers is the ReLU activation function. While Tanh activation function may not work in convolution neural network due to problem with derivatives at particular points, Sigmoid activation could be used, but in my training ReLU activation function was giving higher accuracy even at lower epochs. So, I have used ReLU activation function within the hidden layers.

Another hyperparameter is the learning rate, which I have kept at 0.001. Learning rate is an important hyperparameter, value of learning rate kept too small will result in a long training process time and even that could get stuck, whereas a value too large may result in unstable training process. I tried using lower learning rate 0.00001 which was taking about 30 mins to converge and for learning rate greater than 0.001 the loss was not decreasing after a certain value.

Lastly, I have increased the epoch to 7, again increasing the epoch not always helps in training the model accurately as increasing the epoch too much may result in overfitting of the training data set and again keeping the too low the model will not be sufficiently trained with very low accuracy.



```
Accuracy of the network on the 10000 test images: 68 %
```