# HW4

## 1-Multi-Layer Perceptron

In [1]:

```
# import tensorflow as tf
import tensorflow.compat.v1 as tf
tf.disable_v2_behavior()
import matplotlib.pyplot as plt
from tensorflow.keras.datasets import cifar10
from tensorflow.keras.utils import to_categorical
import numpy as np
```

```
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/tensorflow/pytho
n/compat/v2_compat.py:96: disable_resource_variables (from tensorflow.python.op
s.variable_scope) is deprecated and will be removed in a future version.
Instructions for updating:
non-resource variables are not supported in the long term
```

In [2]:

```
# tf.test.is_gpu_available(
#     cuda_only=False, min_cuda_compute_capability=None
# )
```

In [3]:

```
#Load the data
(x_train, y_train), (x_test, y_test) = cifar10.load_data()
```

In [4]:

```
#define class names
class_names = ['airplane', 'automobile', 'bird', 'cat', 'deer',
               'dog', 'frog', 'horse', 'ship', 'truck']
```

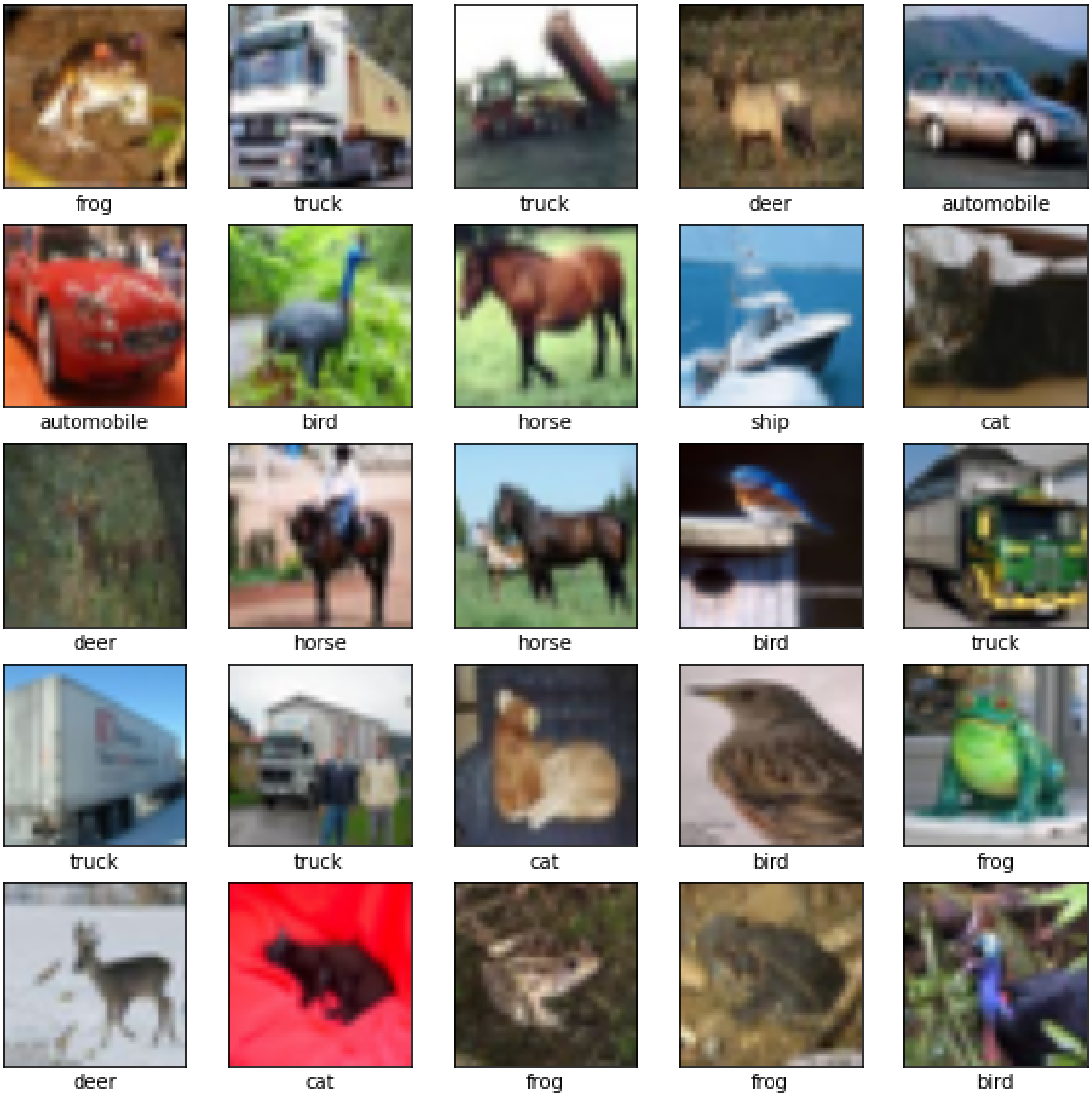
In [5]:

```
print('Train', x_train.shape, y_train.shape)
print('Test', (x_test.shape, y_test.shape))
# normalize pixel values
x_train, x_test = x_train/255, x_test/255
```

```
Train (50000, 32, 32, 3) (50000, 1)
Test ((10000, 32, 32, 3), (10000, 1))
```

In [6]:

```
#Plot some of the images
plt.figure(figsize=(10,10))
for i in range(25):
    plt.subplot(5,5,i+1)
    plt.xticks([])
    plt.yticks([])
    plt.grid(False)
    plt.imshow(x_train[i], cmap=plt.cm.binary)
    # The CIFAR labels happen to be arrays,
    # which is why you need the extra index
    plt.xlabel(class_names[y_train[i][0]])
plt.show()
```



In [7]:

```
#flatting the training data to (50000*3072)
#where 3072 = 32*32*3
# u = x_train.reshape(-1, 3072)
x_train = x_train.reshape(x_train.shape[0], x_train.shape[1]*x_train.shape[2]*x_
x_test = x_test.reshape(x_test.shape[0], x_test.shape[1]*x_test.shape[2]*x_test.
```

In [8]:

```
#One Hot Encode with Keras for the labels
y_train = to_categorical(y_train)
y_test = to_categorical(y_test)
```

In [9]:

```
#Randomize the data
def  shuffle_data(x, y):
    permutation = np.random.permutation(x.shape[0])
    shuffled_x, shuffled_y  = x[permutation], y[permutation]
    return shuffled_x, shuffled_y

def get_next_batch(x, y, start, end):
    x_batch, y_batch = x[start:end], y[start:end]
    return x_batch, y_batch
```

In [10]:

```
#Function for creating layers
def layer(x, num_units, name, use_relu=True):
    """
    Create a fully-connected layer
    :param x: input from previous layer
    :param num_units: number of hidden units in the fully-connected layer
    :param name: layer name
    :param use_relu: boolean to add ReLU non-linearity (or not)
    :return: The output array
    """
    in_dim = x.get_shape()[1]
    shape=[in_dim, num_units]

    W = tf.get_variable('W_' + name,
        dtype=tf.float32,
        shape=shape,
        initializer=tf.truncated_normal_initializer(stddev=0.01))

    b =tf.get_variable('b_' + name,
        dtype=tf.float32,
        initializer=tf.constant(0., shape=[num_units], dtype=tf.float32))

    layer = tf.matmul(x, W)
    layer += b
    if use_relu:
        layer = tf.nn.relu(layer)
    return layer, W
```

In [52]:

```
#Network configuration
h1 = 250 #250                # Number of nodes in the first hidden layer
h2 = 250 #250                   # Number of nodes in the second hidden layer
#input vector size
feature_vector_size = x_train.shape[1]
num_classes = len(class_names)
```

In [53]:

```
# Parameters
learning_rate = 0.001  # The optimization initial learning rate
epochs = 50            # Total number of training epochs
batch_size = 100       # Training batch size
display_freq = 100       # Frequency of displaying the training results
```

In [54]:

```
# Remove previous weights, bias, inputs, etc..
tf.reset_default_graph()
```

In [55]:

```
#creating the network
# Create the graph for the linear model
# Placeholders for inputs (x) and outputs(y)
x = tf.placeholder(tf.float32, shape=[None, feature_vector_size], name='X')
y = tf.placeholder(tf.float32, shape=[None, num_classes], name='Y')
```

In [56]:

```
#Create the network layers
layer_h1, hidden_weights_1 = layer(x, h1, 'h1', use_relu=True)
layer_h2, hidden_weights_2 = layer(layer_h1, h2, 'h2', use_relu=True)
output_logits, hidden_weights_out = layer(layer_h2, num_classes, 'OUT', use_relu
```

In [57]:

```
# Network predictions
cls_prediction = tf.argmax(output_logits, axis=1, name='predictions')

#the loss function, optimizer, accuracy, and predicted class

# Loss function with No Regularization
# loss = tf.reduce_mean(tf.nn.softmax_cross_entropy_with_logits(labels=y, logits

# Loss function using L2 Regularization
loss = (tf.reduce_mean(tf.nn.softmax_cross_entropy_with_logits(labels=y, logits=
        0.001*tf.nn.l2_loss(hidden_weights_1) + \
        0.001*tf.nn.l2_loss(hidden_weights_2) + \
        0.001*tf.nn.l2_loss(hidden_weights_out))


optimizer = tf.train.AdamOptimizer(learning_rate=learning_rate, name='Adam-op').
# optimizer = tf.train.GradientDescentOptimizer(learning_rate = learning_rate).m

correct_prediction = tf.equal(tf.argmax(output_logits, 1), tf.argmax(y, 1), name
accuracy = tf.reduce_mean(tf.cast(correct_prediction, tf.float32), name='accurac
```

In [58]:

```
# Create the op for initializing all variables
init = tf.global_variables_initializer()
```

In [59]:

```
#train
sess = tf.InteractiveSession()
sess.run(init)
global_step = 0
# Number of training iterations in each epoch
num_tr_iter = int(len(y_train) / batch_size)
```

In [61]:

```
for epoch in range(epochs):
    print('Training epoch: {}'.format(epoch + 1))
    x_train, y_train = shuffle_data(x_train, y_train)
    for iteration in range(num_tr_iter):
        global_step += 1
        start = iteration * batch_size
        end = (iteration + 1) * batch_size
        x_batch, y_batch = get_next_batch(x_train, y_train, start, end)

        # Run optimization op (backprop)
        feed_dict_batch = {x: x_batch, y: y_batch}
        sess.run(optimizer, feed_dict=feed_dict_batch)

        if iteration % display_freq == 0:
            # Calculate and display the batch loss and accuracy
            loss_batch, acc_batch = sess.run([loss, accuracy],
                                             feed_dict=feed_dict_batch)

            print("iter {0:3d}:\t Loss={1:.2f},\tTraining Accuracy={2:.01%}".
                  format(iteration, loss_batch, acc_batch))

    # Run validation after every epoch
    feed_dict_valid = {x: x_test[:10000], y: y_test[:10000]}
    loss_valid, acc_valid = sess.run([loss, accuracy], feed_dict=feed_dict_valid
    print('---------------------------------------------------------')
    print("Epoch: {0}, validation loss: {1:.2f}, validation accuracy: {2:.01%}".
          format(epoch + 1, loss_valid, acc_valid))
    print('---------------------------------------------------------')
```

```
Training epoch: 1
iter   0:        Loss=1.47,     Training Accuracy=54.0%
iter 100:        Loss=1.27,     Training Accuracy=64.0%
iter 200:        Loss=1.53,     Training Accuracy=46.0%
iter 300:        Loss=1.30,     Training Accuracy=57.0%
iter 400:        Loss=1.43,     Training Accuracy=49.0%
---------------------------------------------------------
Epoch: 1, validation loss: 1.53, validation accuracy: 49.8%
---------------------------------------------------------
Training epoch: 2
iter   0:        Loss=1.60,     Training Accuracy=46.0%
iter 100:        Loss=1.37,     Training Accuracy=55.0%
iter 200:        Loss=1.46,     Training Accuracy=55.0%
iter 300:        Loss=1.33,     Training Accuracy=59.0%
iter 400:        Loss=1.42,     Training Accuracy=48.0%
---------------------------------------------------------
Epoch: 2, validation loss: 1.54, validation accuracy: 49.7%
---------------------------------------------------------
Training epoch: 3
iter   0:        Loss=1.35,     Training Accuracy=63.0%
iter 100:        Loss=1.32,     Training Accuracy=58.0%
iter 200:        Loss=1.15,     Training Accuracy=63.0%
iter 300:        Loss=1.38,     Training Accuracy=59.0%
iter 400:        Loss=1.39,     Training Accuracy=60.0%
---------------------------------------------------------
Epoch: 3, validation loss: 1.58, validation accuracy: 49.6%
---------------------------------------------------------
Training epoch: 4
iter   0:        Loss=1.26,     Training Accuracy=61.0%
iter 100:        Loss=1.29,     Training Accuracy=67.0%
iter 200:        Loss=1.22,     Training Accuracy=66.0%
iter 300:        Loss=1.38,     Training Accuracy=50.0%
iter 400:        Loss=1.34,     Training Accuracy=54.0%
---------------------------------------------------------
Epoch: 4, validation loss: 1.53, validation accuracy: 51.3%
---------------------------------------------------------
Training epoch: 5
iter   0:        Loss=1.27,     Training Accuracy=59.0%
iter 100:        Loss=1.28,     Training Accuracy=58.0%
iter 200:        Loss=1.36,     Training Accuracy=56.0%
iter 300:        Loss=1.21,     Training Accuracy=55.0%
iter 400:        Loss=1.50,     Training Accuracy=53.0%
---------------------------------------------------------
Epoch: 5, validation loss: 1.53, validation accuracy: 49.8%
---------------------------------------------------------
Training epoch: 6
iter   0:        Loss=1.58,     Training Accuracy=50.0%
iter 100:        Loss=1.56,     Training Accuracy=52.0%
iter 200:        Loss=1.26,     Training Accuracy=60.0%
iter 300:        Loss=1.52,     Training Accuracy=46.0%
iter 400:        Loss=1.67,     Training Accuracy=47.0%
---------------------------------------------------------
Epoch: 6, validation loss: 1.57, validation accuracy: 49.0%
---------------------------------------------------------
Training epoch: 7
iter   0:        Loss=1.38,     Training Accuracy=56.0%
iter 100:        Loss=1.34,     Training Accuracy=58.0%
iter 200:        Loss=1.19,     Training Accuracy=60.0%
iter 300:        Loss=1.28,     Training Accuracy=58.0%
iter 400:        Loss=1.62,     Training Accuracy=38.0%
---------------------------------------------------------
Epoch: 7, validation loss: 1.54, validation accuracy: 50.2%
---------------------------------------------------------
Training epoch: 8
iter   0:        Loss=1.44,     Training Accuracy=48.0%
iter 100:        Loss=1.19,     Training Accuracy=59.0%
iter 200:        Loss=1.35,     Training Accuracy=56.0%
iter 300:        Loss=1.36,     Training Accuracy=54.0%
iter 400:        Loss=1.50,     Training Accuracy=50.0%
---------------------------------------------------------
Epoch: 8, validation loss: 1.50, validation accuracy: 51.5%
---------------------------------------------------------
Training epoch: 9
iter   0:        Loss=1.34,     Training Accuracy=53.0%
iter 100:        Loss=1.34,     Training Accuracy=54.0%
iter 200:        Loss=1.16,     Training Accuracy=65.0%
iter 300:        Loss=1.54,     Training Accuracy=51.0%
iter 400:        Loss=1.29,     Training Accuracy=61.0%
---------------------------------------------------------
Epoch: 9, validation loss: 1.50, validation accuracy: 51.9%
---------------------------------------------------------
Training epoch: 10
iter   0:        Loss=1.25,     Training Accuracy=59.0%
iter 100:        Loss=1.28,     Training Accuracy=52.0%
iter 200:        Loss=1.54,     Training Accuracy=53.0%
iter 300:        Loss=1.46,     Training Accuracy=51.0%
iter 400:        Loss=1.31,     Training Accuracy=56.0%
---------------------------------------------------------
Epoch: 10, validation loss: 1.50, validation accuracy: 51.2%
---------------------------------------------------------
Training epoch: 11
iter   0:        Loss=1.03,     Training Accuracy=74.0%
iter 100:        Loss=1.31,     Training Accuracy=62.0%
iter 200:        Loss=1.50,     Training Accuracy=49.0%
iter 300:        Loss=1.24,     Training Accuracy=60.0%
iter 400:        Loss=1.39,     Training Accuracy=50.0%
---------------------------------------------------------
Epoch: 11, validation loss: 1.56, validation accuracy: 49.9%
---------------------------------------------------------
Training epoch: 12
iter   0:        Loss=1.53,     Training Accuracy=51.0%
iter 100:        Loss=1.32,     Training Accuracy=60.0%
iter 200:        Loss=1.52,     Training Accuracy=49.0%
iter 300:        Loss=1.22,     Training Accuracy=58.0%
iter 400:        Loss=1.39,     Training Accuracy=52.0%
---------------------------------------------------------
Epoch: 12, validation loss: 1.54, validation accuracy: 50.5%
---------------------------------------------------------
Training epoch: 13
iter   0:        Loss=1.58,     Training Accuracy=49.0%
iter 100:        Loss=1.25,     Training Accuracy=59.0%
iter 200:        Loss=1.22,     Training Accuracy=61.0%
iter 300:        Loss=1.38,     Training Accuracy=57.0%
iter 400:        Loss=1.50,     Training Accuracy=48.0%
---------------------------------------------------------
Epoch: 13, validation loss: 1.53, validation accuracy: 50.9%
---------------------------------------------------------
Training epoch: 14
iter   0:        Loss=1.54,     Training Accuracy=52.0%
iter 100:        Loss=1.24,     Training Accuracy=60.0%
iter 200:        Loss=1.40,     Training Accuracy=53.0%
iter 300:        Loss=1.48,     Training Accuracy=56.0%
iter 400:        Loss=1.26,     Training Accuracy=62.0%
```

```
---------------------------------------------------------
Epoch: 14, validation loss: 1.57, validation accuracy: 48.9%
---------------------------------------------------------
Training epoch: 15
iter   0:        Loss=1.35,     Training Accuracy=56.0%
iter 100:        Loss=1.45,     Training Accuracy=53.0%
iter 200:        Loss=1.40,     Training Accuracy=54.0%
iter 300:        Loss=1.38,     Training Accuracy=56.0%
iter 400:        Loss=1.23,     Training Accuracy=64.0%
---------------------------------------------------------
Epoch: 15, validation loss: 1.48, validation accuracy: 52.7%
---------------------------------------------------------
Training epoch: 16
iter   0:        Loss=1.44,     Training Accuracy=49.0%
iter 100:        Loss=1.37,     Training Accuracy=52.0%
iter 200:        Loss=1.38,     Training Accuracy=56.0%
iter 300:        Loss=1.41,     Training Accuracy=61.0%
iter 400:        Loss=1.31,     Training Accuracy=61.0%
---------------------------------------------------------
Epoch: 16, validation loss: 1.56, validation accuracy: 49.4%
---------------------------------------------------------
Training epoch: 17
iter   0:        Loss=1.31,     Training Accuracy=55.0%
iter 100:        Loss=1.29,     Training Accuracy=58.0%
iter 200:        Loss=1.36,     Training Accuracy=54.0%
iter 300:        Loss=1.32,     Training Accuracy=53.0%
iter 400:        Loss=1.33,     Training Accuracy=60.0%
---------------------------------------------------------
Epoch: 17, validation loss: 1.54, validation accuracy: 49.8%
---------------------------------------------------------
Training epoch: 18
iter   0:        Loss=1.39,     Training Accuracy=57.0%
iter 100:        Loss=1.34,     Training Accuracy=58.0%
iter 200:        Loss=1.28,     Training Accuracy=66.0%
iter 300:        Loss=1.51,     Training Accuracy=46.0%
iter 400:        Loss=1.44,     Training Accuracy=53.0%
---------------------------------------------------------
Epoch: 18, validation loss: 1.50, validation accuracy: 51.5%
---------------------------------------------------------
Training epoch: 19
iter   0:        Loss=1.42,     Training Accuracy=59.0%
iter 100:        Loss=1.25,     Training Accuracy=61.0%
iter 200:        Loss=1.43,     Training Accuracy=55.0%
iter 300:        Loss=1.24,     Training Accuracy=59.0%
iter 400:        Loss=1.37,     Training Accuracy=56.0%
---------------------------------------------------------
Epoch: 19, validation loss: 1.50, validation accuracy: 52.6%
---------------------------------------------------------
Training epoch: 20
iter   0:        Loss=1.31,     Training Accuracy=59.0%
iter 100:        Loss=1.34,     Training Accuracy=68.0%
iter 200:        Loss=1.40,     Training Accuracy=49.0%
iter 300:        Loss=1.21,     Training Accuracy=57.0%
iter 400:        Loss=1.38,     Training Accuracy=56.0%
---------------------------------------------------------
Epoch: 20, validation loss: 1.49, validation accuracy: 52.1%
---------------------------------------------------------
Training epoch: 21
iter   0:        Loss=1.34,     Training Accuracy=57.0%
iter 100:        Loss=1.26,     Training Accuracy=64.0%
iter 200:        Loss=1.30,     Training Accuracy=60.0%
iter 300:        Loss=1.40,     Training Accuracy=62.0%
iter 400:        Loss=1.53,     Training Accuracy=49.0%
---------------------------------------------------------
Epoch: 21, validation loss: 1.50, validation accuracy: 51.7%
```

```
---------------------------------------------------------
Training epoch: 22
iter   0:        Loss=1.33,     Training Accuracy=62.0%
iter 100:        Loss=1.41,     Training Accuracy=57.0%
iter 200:        Loss=1.49,     Training Accuracy=53.0%
iter 300:        Loss=1.46,     Training Accuracy=51.0%
iter 400:        Loss=1.43,     Training Accuracy=50.0%
---------------------------------------------------------
Epoch: 22, validation loss: 1.51, validation accuracy: 51.2%
---------------------------------------------------------
Training epoch: 23
iter   0:        Loss=1.19,     Training Accuracy=66.0%
iter 100:        Loss=1.36,     Training Accuracy=54.0%
iter 200:        Loss=1.40,     Training Accuracy=59.0%
iter 300:        Loss=1.31,     Training Accuracy=57.0%
iter 400:        Loss=1.38,     Training Accuracy=55.0%
---------------------------------------------------------
Epoch: 23, validation loss: 1.58, validation accuracy: 48.8%
---------------------------------------------------------
Training epoch: 24
iter   0:        Loss=1.32,     Training Accuracy=59.0%
iter 100:        Loss=1.26,     Training Accuracy=60.0%
iter 200:        Loss=1.36,     Training Accuracy=58.0%
iter 300:        Loss=1.30,     Training Accuracy=58.0%
iter 400:        Loss=1.42,     Training Accuracy=56.0%
---------------------------------------------------------
Epoch: 24, validation loss: 1.52, validation accuracy: 51.1%
---------------------------------------------------------
Training epoch: 25
iter   0:        Loss=1.29,     Training Accuracy=59.0%
iter 100:        Loss=1.35,     Training Accuracy=60.0%
iter 200:        Loss=1.30,     Training Accuracy=57.0%
iter 300:        Loss=1.43,     Training Accuracy=56.0%
iter 400:        Loss=1.26,     Training Accuracy=59.0%
---------------------------------------------------------
Epoch: 25, validation loss: 1.53, validation accuracy: 50.3%
---------------------------------------------------------
Training epoch: 26
iter   0:        Loss=1.38,     Training Accuracy=52.0%
iter 100:        Loss=1.34,     Training Accuracy=57.0%
iter 200:        Loss=1.45,     Training Accuracy=54.0%
iter 300:        Loss=1.47,     Training Accuracy=50.0%
iter 400:        Loss=1.42,     Training Accuracy=58.0%
---------------------------------------------------------
Epoch: 26, validation loss: 1.53, validation accuracy: 50.7%
---------------------------------------------------------
Training epoch: 27
iter   0:        Loss=1.30,     Training Accuracy=52.0%
iter 100:        Loss=1.41,     Training Accuracy=57.0%
iter 200:        Loss=1.21,     Training Accuracy=62.0%
iter 300:        Loss=1.19,     Training Accuracy=60.0%
iter 400:        Loss=1.22,     Training Accuracy=64.0%
---------------------------------------------------------
Epoch: 27, validation loss: 1.53, validation accuracy: 50.9%
---------------------------------------------------------
Training epoch: 28
iter   0:        Loss=1.36,     Training Accuracy=58.0%
iter 100:        Loss=1.35,     Training Accuracy=53.0%
iter 200:        Loss=1.42,     Training Accuracy=53.0%
iter 300:        Loss=1.34,     Training Accuracy=61.0%
iter 400:        Loss=1.31,     Training Accuracy=58.0%
---------------------------------------------------------
Epoch: 28, validation loss: 1.54, validation accuracy: 50.6%
---------------------------------------------------------
Training epoch: 29
```

```
iter   0:        Loss=1.43,     Training Accuracy=58.0%
iter 100:        Loss=1.21,     Training Accuracy=57.0%
iter 200:        Loss=1.53,     Training Accuracy=56.0%
iter 300:        Loss=1.33,     Training Accuracy=58.0%
iter 400:        Loss=1.17,     Training Accuracy=63.0%
---------------------------------------------------------
Epoch: 29, validation loss: 1.54, validation accuracy: 50.2%
---------------------------------------------------------
Training epoch: 30
iter   0:        Loss=1.24,     Training Accuracy=59.0%
iter 100:        Loss=1.23,     Training Accuracy=66.0%
iter 200:        Loss=1.09,     Training Accuracy=69.0%
iter 300:        Loss=1.20,     Training Accuracy=67.0%
iter 400:        Loss=1.27,     Training Accuracy=56.0%
---------------------------------------------------------
Epoch: 30, validation loss: 1.57, validation accuracy: 49.8%
---------------------------------------------------------
Training epoch: 31
iter   0:        Loss=1.42,     Training Accuracy=58.0%
iter 100:        Loss=1.37,     Training Accuracy=53.0%
iter 200:        Loss=1.21,     Training Accuracy=64.0%
iter 300:        Loss=1.37,     Training Accuracy=48.0%
iter 400:        Loss=1.42,     Training Accuracy=57.0%
---------------------------------------------------------
Epoch: 31, validation loss: 1.51, validation accuracy: 51.8%
---------------------------------------------------------
Training epoch: 32
iter   0:        Loss=1.32,     Training Accuracy=57.0%
iter 100:        Loss=1.33,     Training Accuracy=57.0%
iter 200:        Loss=1.48,     Training Accuracy=59.0%
iter 300:        Loss=1.41,     Training Accuracy=54.0%
iter 400:        Loss=1.51,     Training Accuracy=61.0%
---------------------------------------------------------
Epoch: 32, validation loss: 1.51, validation accuracy: 52.2%
---------------------------------------------------------
Training epoch: 33
iter   0:        Loss=1.34,     Training Accuracy=55.0%
iter 100:        Loss=1.43,     Training Accuracy=51.0%
iter 200:        Loss=1.45,     Training Accuracy=51.0%
iter 300:        Loss=1.47,     Training Accuracy=51.0%
iter 400:        Loss=1.34,     Training Accuracy=55.0%
---------------------------------------------------------
Epoch: 33, validation loss: 1.50, validation accuracy: 51.5%
---------------------------------------------------------
Training epoch: 34
iter   0:        Loss=1.22,     Training Accuracy=61.0%
iter 100:        Loss=1.26,     Training Accuracy=61.0%
iter 200:        Loss=1.31,     Training Accuracy=57.0%
iter 300:        Loss=1.21,     Training Accuracy=66.0%
iter 400:        Loss=1.26,     Training Accuracy=61.0%
---------------------------------------------------------
Epoch: 34, validation loss: 1.53, validation accuracy: 50.9%
---------------------------------------------------------
Training epoch: 35
iter   0:        Loss=1.33,     Training Accuracy=61.0%
iter 100:        Loss=1.21,     Training Accuracy=65.0%
iter 200:        Loss=1.25,     Training Accuracy=58.0%
iter 300:        Loss=1.42,     Training Accuracy=57.0%
iter 400:        Loss=1.18,     Training Accuracy=59.0%
---------------------------------------------------------
Epoch: 35, validation loss: 1.51, validation accuracy: 51.6%
---------------------------------------------------------
Training epoch: 36
iter   0:        Loss=1.29,     Training Accuracy=67.0%
iter 100:        Loss=1.13,     Training Accuracy=65.0%
iter 200:        Loss=1.32,     Training Accuracy=56.0%
iter 300:        Loss=1.31,     Training Accuracy=56.0%
iter 400:        Loss=1.35,     Training Accuracy=61.0%
---------------------------------------------------------
Epoch: 36, validation loss: 1.56, validation accuracy: 50.1%
---------------------------------------------------------
Training epoch: 37
iter   0:        Loss=1.49,     Training Accuracy=55.0%
iter 100:        Loss=1.25,     Training Accuracy=58.0%
iter 200:        Loss=1.14,     Training Accuracy=60.0%
iter 300:        Loss=1.38,     Training Accuracy=55.0%
iter 400:        Loss=1.26,     Training Accuracy=50.0%
---------------------------------------------------------
Epoch: 37, validation loss: 1.54, validation accuracy: 50.3%
---------------------------------------------------------
Training epoch: 38
iter   0:        Loss=1.43,     Training Accuracy=45.0%
iter 100:        Loss=1.52,     Training Accuracy=48.0%
iter 200:        Loss=1.20,     Training Accuracy=62.0%
iter 300:        Loss=1.28,     Training Accuracy=61.0%
iter 400:        Loss=1.38,     Training Accuracy=51.0%
---------------------------------------------------------
Epoch: 38, validation loss: 1.50, validation accuracy: 52.6%
---------------------------------------------------------
Training epoch: 39
iter   0:        Loss=1.26,     Training Accuracy=61.0%
iter 100:        Loss=1.18,     Training Accuracy=64.0%
iter 200:        Loss=1.25,     Training Accuracy=59.0%
iter 300:        Loss=1.65,     Training Accuracy=44.0%
iter 400:        Loss=1.54,     Training Accuracy=52.0%
---------------------------------------------------------
Epoch: 39, validation loss: 1.48, validation accuracy: 52.6%
---------------------------------------------------------
Training epoch: 40
iter   0:        Loss=1.52,     Training Accuracy=51.0%
iter 100:        Loss=1.39,     Training Accuracy=51.0%
iter 200:        Loss=1.35,     Training Accuracy=60.0%
iter 300:        Loss=1.46,     Training Accuracy=52.0%
iter 400:        Loss=1.35,     Training Accuracy=57.0%
---------------------------------------------------------
Epoch: 40, validation loss: 1.51, validation accuracy: 51.7%
---------------------------------------------------------
Training epoch: 41
iter   0:        Loss=1.32,     Training Accuracy=57.0%
iter 100:        Loss=1.25,     Training Accuracy=62.0%
iter 200:        Loss=1.37,     Training Accuracy=51.0%
iter 300:        Loss=1.29,     Training Accuracy=61.0%
iter 400:        Loss=1.37,     Training Accuracy=54.0%
---------------------------------------------------------
Epoch: 41, validation loss: 1.50, validation accuracy: 52.2%
---------------------------------------------------------
Training epoch: 42
iter   0:        Loss=1.39,     Training Accuracy=51.0%
iter 100:        Loss=1.39,     Training Accuracy=60.0%
iter 200:        Loss=1.23,     Training Accuracy=56.0%
iter 300:        Loss=1.42,     Training Accuracy=57.0%
iter 400:        Loss=1.19,     Training Accuracy=64.0%
---------------------------------------------------------
Epoch: 42, validation loss: 1.52, validation accuracy: 51.6%
---------------------------------------------------------
Training epoch: 43
iter   0:        Loss=1.36,     Training Accuracy=56.0%
iter 100:        Loss=1.38,     Training Accuracy=54.0%
iter 200:        Loss=1.21,     Training Accuracy=59.0%
iter 300:        Loss=1.41,     Training Accuracy=57.0%
iter 400:        Loss=1.39,     Training Accuracy=50.0%
---------------------------------------------------------
Epoch: 43, validation loss: 1.56, validation accuracy: 49.6%
---------------------------------------------------------
Training epoch: 44
iter   0:        Loss=1.30,     Training Accuracy=54.0%
iter 100:        Loss=1.11,     Training Accuracy=68.0%
iter 200:        Loss=1.31,     Training Accuracy=64.0%
iter 300:        Loss=1.40,     Training Accuracy=52.0%
iter 400:        Loss=1.47,     Training Accuracy=53.0%
---------------------------------------------------------
Epoch: 44, validation loss: 1.54, validation accuracy: 50.3%
---------------------------------------------------------
Training epoch: 45
iter   0:        Loss=1.12,     Training Accuracy=65.0%
iter 100:        Loss=1.64,     Training Accuracy=44.0%
iter 200:        Loss=1.33,     Training Accuracy=60.0%
iter 300:        Loss=1.28,     Training Accuracy=63.0%
iter 400:        Loss=1.41,     Training Accuracy=51.0%
---------------------------------------------------------
Epoch: 45, validation loss: 1.54, validation accuracy: 51.3%
---------------------------------------------------------
Training epoch: 46
iter   0:        Loss=1.33,     Training Accuracy=55.0%
iter 100:        Loss=1.29,     Training Accuracy=63.0%
iter 200:        Loss=1.43,     Training Accuracy=52.0%
iter 300:        Loss=1.34,     Training Accuracy=54.0%
iter 400:        Loss=1.31,     Training Accuracy=57.0%
---------------------------------------------------------
Epoch: 46, validation loss: 1.55, validation accuracy: 50.4%
---------------------------------------------------------
Training epoch: 47
iter   0:        Loss=1.45,     Training Accuracy=53.0%
iter 100:        Loss=1.46,     Training Accuracy=51.0%
iter 200:        Loss=1.45,     Training Accuracy=51.0%
iter 300:        Loss=1.33,     Training Accuracy=60.0%
iter 400:        Loss=1.47,     Training Accuracy=51.0%
---------------------------------------------------------
Epoch: 47, validation loss: 1.52, validation accuracy: 51.2%
---------------------------------------------------------
Training epoch: 48
iter   0:        Loss=1.46,     Training Accuracy=54.0%
iter 100:        Loss=1.19,     Training Accuracy=62.0%
iter 200:        Loss=1.33,     Training Accuracy=51.0%
iter 300:        Loss=1.18,     Training Accuracy=63.0%
iter 400:        Loss=1.30,     Training Accuracy=57.0%
---------------------------------------------------------
Epoch: 48, validation loss: 1.51, validation accuracy: 51.6%
---------------------------------------------------------
Training epoch: 49
iter   0:        Loss=1.39,     Training Accuracy=56.0%
iter 100:        Loss=1.58,     Training Accuracy=51.0%
iter 200:        Loss=1.30,     Training Accuracy=57.0%
iter 300:        Loss=1.39,     Training Accuracy=55.0%
iter 400:        Loss=1.44,     Training Accuracy=48.0%
---------------------------------------------------------
Epoch: 49, validation loss: 1.55, validation accuracy: 49.9%
---------------------------------------------------------
Training epoch: 50
iter   0:        Loss=1.39,     Training Accuracy=51.0%
iter 100:        Loss=1.24,     Training Accuracy=57.0%
iter 200:        Loss=1.45,     Training Accuracy=53.0%
iter 300:        Loss=1.54,     Training Accuracy=48.0%
iter 400:        Loss=1.36,     Training Accuracy=50.0%
---------------------------------------------------------
```

```
Epoch: 50, validation loss: 1.53, validation accuracy: 50.8%
----------------------------------------------------------
```

# Results

| Config. | Classification Error | | |
|---|---|---|---|
| *** | Training | Testing | Accuracy |
| 50HLN+no regularization | 1.33 | 1.42 | 49.5 |
| 50HLN+L2 regularization | 1.24 | 1.54 | 48.8 |
| 250HLN+no regularization | 0.80 | 1.50 | 52.0 |
| 250HLN+L2 regularization | 1.36 | 1.53 | 50.8 |

From the above results, we can conclude that MLP does not do a good job on CIFAR-10 dataset. The next step is to use CNN and compare the results.

In [ ]: