Train SVM on features

################################################################################

# TODO:                                                                        #

# Use the validation set to set the learning rate and regularization strength. #

# This should be identical to the validation that you did for the SVM; save    #

# the best trained classifer in best\_svm. You might also want to play          #

# with different numbers of bins in the color histogram. If you are careful    #

# you should be able to get accuracy of near 0.44 on the validation set.       #

################################################################################

Train Neural Network on image features

################################################################################

# TODO: Train a two-layer neural network on image features. You may want to    #

# cross-validate various parameters as in previous sections. Store your best   #

# model in the best\_net variable.                                              #

################################################################################

# \*\*\*\*\*START OF YOUR CODE (DO NOT DELETE/MODIFY THIS LINE)\*\*\*\*\*

best\_val = -1

learning\_rates=[1e-3,3e-3,1e-2,3e-2,1e-1,3e-1]

regularization\_strengths=[1e-3,3e-3,1e-2,3e-3,1e-1]

results={}

for learning\_rate in learning\_rates:

  for regularization\_strength in regularization\_strengths:

    net=TwoLayerNet(input\_dim,hidden\_dim,num\_classes)

    loss\_hist = net.train(X\_train\_feats, y\_train, X\_val\_feats, y\_val,

            num\_iters=1000, batch\_size=200,

            learning\_rate=learning\_rate, learning\_rate\_decay=0.95,

            reg=regularization\_strength, verbose=False)

    y\_train\_pred=net.predict(X\_train\_feats)

    training\_accuracy = np.mean(y\_train == y\_train\_pred)

    y\_val\_pred=net.predict(X\_val\_feats)

    validation\_accuracy=np.mean(y\_val\_pred==y\_val)

    results[(learning\_rate, regularization\_strength)] = (training\_accuracy, validation\_accuracy)

    if best\_val<validation\_accuracy:

      best\_val=validation\_accuracy

      best\_net=net

for lr,rs in sorted(results):

    val\_acc = results[(lr,rs)]

    print('lr:',lr,"reg:",rs,"accuracy:",val\_acc)

print('best validation accuracy achieved during cross-validation:%f' %best\_val)