COMP 5212: Machine Learning

Zhen Li 12226876

Project Set 2

1. Convolutional Neural Network

1.1 Experiment Setting

We set the learning rate as 0.0015, 0.001, and 0.0003, and apply the cross-validation method to find the optimal one of them. Also, based on the previous experience we choose the batch size as 64, and the momentum as 0.9.

1.2 Mini-batch stochastic optimization algorithm

We adopt the mini-batch stochastic optimization algorithm as follows:

```
permutation = np.random.permutation(y.shape[0])
x = x[permutation]
y = y[permutation]
for n_pass in range(num_iterations):
  for i in train_set:
      batch_x = x[i * BATCH_SIZE : (i + 1) * BATCH_SIZE]
      batch_y = y[i * BATCH_SIZE : (i + 1) * BATCH_SIZE]
      feed_dict = { placeholder_x: batch_x, placeholder_y: batch_y }
      _, acc, loss_ = sess.run([train_op, accuracy, loss], feed_dict = feed_dict)
```

Shuffling the original data and re-grouping them, then applying multiple stochastic gradient descent on them in every loop.

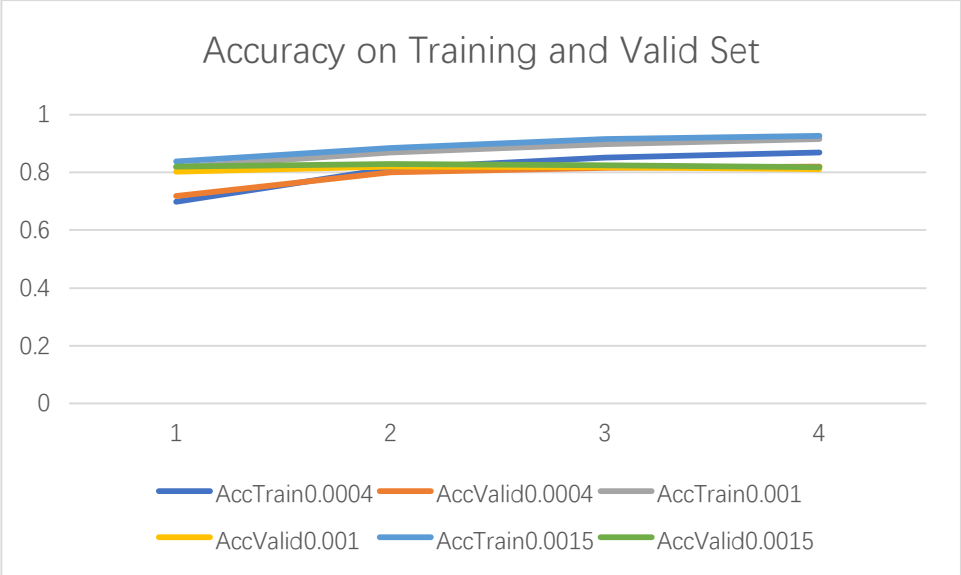
The momentum update algorithm:

```
def gradient_decent(loss, params, learning_rate=0.001, momentum=0.9):
    train_op_1 = []
    train_op_2 = []
```

```
for i in range(len(params)):
    delta = tf.Variable(tf.zeros(params[i].shape), dtype=tf.float32)
    gradient = tf.gradients(loss, params[i])[0]
    cu_delta = momentum * delta - learning_rate * gradient
    train_op_1.append(tf.assign_add(params[i], cu_delta))
    train_op_2.append(tf.assign(delta, cu_delta))

train_op = train_op_1 + train_op_2
return train_op
```

1.3 performance over time for different parameter settings



Learning Rate = 0.0004, Max Epoch = 20,

Epoch #5 , Acc_train=0.699 Loss_train=1.011, Acc_valid=0.717, Loss_valid=0.964

Epoch #10 , Acc_train=0.813, Loss_train=0.588, Acc_valid=0.799 Loss_valid=0.674

Epoch #15 , Acc_train=0.850, Loss_train=0.459, Acc_valid=0.815, Loss_valid=0.615

Epoch #19 , Acc_train=0.868, Loss_train=0.398, Acc_valid=0.819, Loss_valid=0.605

Learning Rate = 0.0010, Max Epoch = 20,

Epoch #5 , Acc_train=0.817, Loss_train=0.570, Acc_valid=0.802, Loss_valid=0.633

Epoch #10 , Acc_train=0.869, Loss_train=0.387, Acc_valid=0.820, Loss_valid=0.591

Epoch #15 , Acc_train=0.897, Loss_train=0.301, Acc_valid=0.817, Loss_valid=0.629

Epoch #19 , Acc_train=0.915, Loss_train=0.255, Acc_valid=0.810, Loss_valid=0.674

Learning Rate = 0.0015, Max Epoch = 20,

Epoch #5 , Acc_train=0.837, Loss_train=0.495, Acc_valid=0.820, Loss_valid=0.568

Epoch #10 , Acc_train=0.885, Loss_train=0.336, Acc_valid=0.828, Loss_valid=0.557

Epoch #15 , Acc_train=0.914, Loss_train=0.253, Acc_valid=0.824, Loss_valid=0.617

Epoch #19 , Acc_train=0.926, Loss_train=0.214, Acc_valid=0.818, Loss_valid=0.682

2.4 Experiment Result

cnn_0.0015_0.9_64

Result , Acc_test=0.8145161271095276, Loss_test=0.6541479825973511

cnn_0.001_0.9_64

Result , Acc_test=0.8079637289047241, Loss_test=0.6273872256278992

cnn_0.0004_0.9_64

Result , Acc_test=0.8160282373428345, Loss_test=0.602705180644989

2. Convolutional Autoencoder

2.1 Experiment Setting

We set the learning rate as 0.0005, 0.0008, and 0.0010, and there is error occurs when the rate set as 0.0010. Also, based on the previous experience we choose the batch size as 64, and the momentum as 0.9.

2.2 Experiment Result

Epoch #5, Loss_train=0.03861572965979576

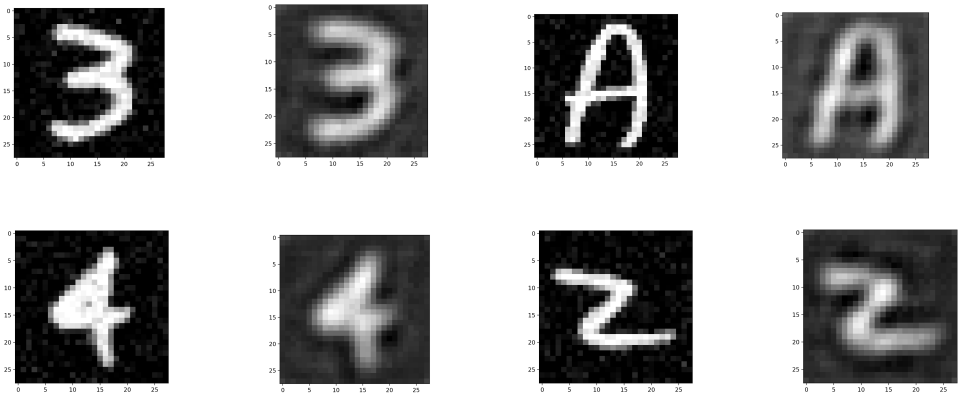
Epoch #10, Loss_train=0.026166768744587898

Epoch #15, Loss_train=0.020421285182237625

Epoch #19, Loss_train=0.017822369933128357

Loss on testset = 0.017654667

2.3 A few feature maps and reconstructed images produced by the CAE model



Example of feature map:

[[[0.87184197 0.7096853] [0.7473166 0.33004403] [0.95246553 0.7535666] [1.2819445 0.8504615] [1.4999938 0.6342538] [1.2007645 0.46327776] [1.2637042 0.6132579]]

[[0.84579736 0.22648048] [0.87583816 -0.12212849] [0.81118566 0.18797743] [2.783831 2.438582] [0.09575433 0.32365388] [1.4704258 -0.13659787] [1.1297866 0.30737597]]

[[0.8403415 0.08197916] [0.71428365 -0.15619898] [2.0802326 1.7169778] [2.8018744 1.8436596] [2.5963993 2.0568476] [3.0723317 0.9826435] [1.3730428 0.14496732]]

[[0.8501065 0.23363179] [0.9964335 0.48251688] [2.9486856 2.9676967] [1.3993677 1.4864876] [0.9823092 1.2224166] [0.99050087 1.0155503] [1.434549 0.3227402]]

[[0.53531337 0.5467569] [2.770674 2.0200105] [0.8370061 0.8190327] [1.217453 -0.63463354] [1.9189775 -0.4485718] [1.6556928 -0.3448727] [1.6023064 0.23025733]]

[[0.9165551 1.6440792] [1.4010097 1.4378352] [1.184716 -0.3284917] [1.4876192 0.03785104] [1.1331784 0.15350902] [1.1937983 0.13428706] [1.3868511 0.35576314]]

[[1.026262 0.4108966] [0.88165545 -0.08568025] [1.2166476 0.2104975] [0.8687876 0.35843182] [0.82536733 0.3770985] [0.94269156 0.4088667] [1.1389284 0.5832718]]]

The detail data of feature maps (which corresponding to the output of conv3) shows in “ae_0.0008_64.txt” .

3. Model Training Time

Model	Learning Rate	Batch Size	Time
CNN	0.0015	64	6m34.769s
CNN	0.0010	64	6m30.205s
CNN	0.0004	64	6m32.397s
CAE	0.0005	64	7m40.345s