



```
In [1]: 1 from keras.datasets import mnist
2 from keras import models
3 from keras import layers
4 (train_images, train_labels), (test_images, test_labels) = mnist.load_data()
5 print("[Info] train data={:7,}").format(len(train_images)))
6 print("[Info] test data={:7,}").format(len(test_images)))
```

```
D:\Anaconda3\lib\site-packages\h5py\_init__.py:36: FutureWarning: Conversion of the second argument of issubdtype from `float` to `np.floating` is deprecated. In future, it will be treated as `np.float64 == np.dtype(float).type`.
    from ._conv import register_converters as _register_converters
Using TensorFlow backend.
```

```
[Info] train data= 60,000
[Info] test data= 10,000
```

```
In [2]: 1 train_images = train_images.reshape((60000, 28 * 28))
2 train_images = train_images.astype('float32') / 255
3
4 test_images = test_images.reshape((10000, 28 * 28))
5 test_images = test_images.astype('float32') / 255
6
7 from keras.utils import to_categorical
8
9 train_labels = to_categorical(train_labels)
10 test_labels = to_categorical(test_labels)
```

```
In [3]: 1 import matplotlib.pyplot as plt
2 %matplotlib inline
```

```
In [7]: 1 def show_train_history(train_history, train, validation, epoch):
2     plt.plot(train_history.history[train], linewidth=3)
3     plt.plot(train_history.history[validation], linewidth=3)
4     plt.title('Train History')
5     plt.ylabel(train)
6     plt.xlabel('Epoch')
7     plt.legend(['Train', 'Validation'], loc='best')
8     plt.grid(True)
9     if train == 'acc':
10         plt.savefig("epochs_acc_" + str(epoch) + ".jpg")
11     if train == 'loss':
12         plt.savefig("epochs_loss_" + str(epoch) + ".jpg")
13     plt.show()
```

```
In [5]: 1 #epochs=5
2 network = models.Sequential()
3 network.add(layers.Dense(512, activation='relu', input_shape=(28 * 28,)))
4 network.add(layers.Dense(10, activation='softmax'))
5 network.compile(optimizer='rmsprop', loss='categorical_crossentropy', metrics=['accuracy'])
6 train_history = network.fit(train_images, train_labels, validation_split=0.2)
7 test_loss, test_acc = network.evaluate(test_images, test_labels)
8 print('test_loss:', test_loss)
9 print('test_acc:', test_acc)
10 #print(train_history.history)
```

```
WARNING:tensorflow:From D:\Anaconda3\lib\site-packages\tensorflow\python\framework\op_def_library.py:263: colocate_with (from tensorflow.python.framework.ops) is deprecated and will be removed in a future version.
```

```
Instructions for updating:
```

```
Colocations handled automatically by placer.
```

```
WARNING:tensorflow:From D:\Anaconda3\lib\site-packages\tensorflow\python\ops\math_ops.py:3066: to_int32 (from tensorflow.python.ops.math_ops) is deprecated and will be removed in a future version.
```

```
Instructions for updating:
```

```
Use tf.cast instead.
```

```
Train on 48000 samples, validate on 12000 samples
```

```
Epoch 1/5
```

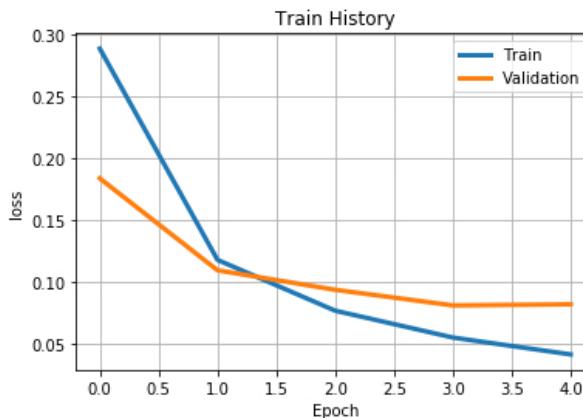
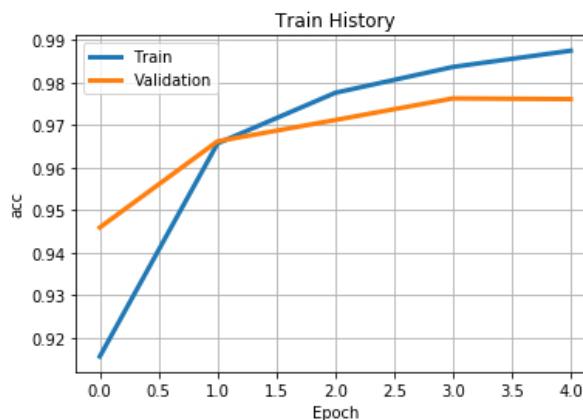
```
48000/48000 [=====] - 9s 192us/step - loss: 0.2885 - acc: 0.9156 - val_loss: 0.1838 - val_acc: 0.9459
```

```
Epoch 2/5
```

```
48000/48000 [=====] - 5s 98us/step - loss: 0.1180 - acc: 0.9657 - val_loss: 0.1096 - val_acc: 0.9662
Epoch 3/5
48000/48000 [=====] - 5s 103us/step - loss: 0.0770 - acc: 0.9776 - val_loss: 0.0939 - val_acc: 0.9712
Epoch 4/5
48000/48000 [=====] - 4s 73us/step - loss: 0.0553 - acc: 0.9836 - val_loss: 0.0813 - val_acc: 0.9763
Epoch 5/5
48000/48000 [=====] - 4s 93us/step - loss: 0.0418 - acc: 0.9875 - val_loss: 0.0822 - val_acc: 0.9761
10000/10000 [=====] - 1s 136us/step
test_loss: 0.07455049938930897
test_acc: 0.9775
```

In [8]:

```
1 epoch = 5
2 show_train_history(train_history, 'acc', 'val_acc', 5)
3 show_train_history(train_history, 'loss', 'val_loss', 5)
```



In [9]:

```
1 #評估模型準確率
2 scores_5 = network.evaluate(train_images, train_labels)
3 print()
4 print("[Info] Accuracy of testing data = {:.1f}%".format(scores_5[1]*100.0))
```

```
60000/60000 [=====] - 7s 123us/step
```

```
[Info] Accuracy of testing data = 98.9%
```

In [10]:

```
1 #epochs=20
2 network = models.Sequential()
3 network.add(layers.Dense(512, activation='relu', input_shape=(28 * 28,)))
4 network.add(layers.Dense(10, activation='softmax'))
5 network.compile(optimizer='rmsprop', loss='categorical_crossentropy', metrics=['accuracy'])
6 train_history = network.fit(train_images, train_labels, validation_split=0.2)
7 test_loss, test_acc = network.evaluate(test_images, test_labels)
8 print('test_loss:', test_loss)
9 print('test_acc:', test_acc)
10 #print(train_history.history)
```

```
Train on 48000 samples, validate on 12000 samples
```

```
Epoch 1/20
```

```
48000/48000 [=====] - 5s 104us/step - loss: 0.2858 - acc: 0.9165 - val_loss: 0.1490 - val_acc: 0.9569
```

```
Epoch 2/20
```

```

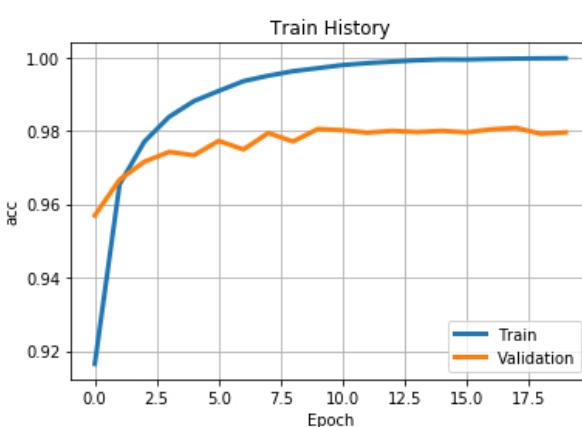
48000/48000 [=====] - 5s 108us/step - loss: 0.1173 - acc: 0.9654 - val_loss: 0.1089 - val_acc: 0.9667
Epoch 3/20
48000/48000 [=====] - 5s 107us/step - loss: 0.0764 - acc: 0.9771 - val_loss: 0.0922 - val_acc: 0.9716
Epoch 4/20
48000/48000 [=====] - 5s 111us/step - loss: 0.0548 - acc: 0.9838 - val_loss: 0.0827 - val_acc: 0.9743
Epoch 5/20
48000/48000 [=====] - 5s 108us/step - loss: 0.0404 - acc: 0.9881 - val_loss: 0.0925 - val_acc: 0.9733
Epoch 6/20
48000/48000 [=====] - 5s 106us/step - loss: 0.0313 - acc: 0.9909 - val_loss: 0.0775 - val_acc: 0.9773
Epoch 7/20
48000/48000 [=====] - 4s 87us/step - loss: 0.0227 - accuracy: 0.9935 - val_loss: 0.0918 - val_acc: 0.9749
Epoch 8/20
48000/48000 [=====] - 5s 114us/step - loss: 0.0177 - acc: 0.9950 - val_loss: 0.0796 - val_acc: 0.9794
Epoch 9/20
48000/48000 [=====] - 5s 108us/step - loss: 0.0128 - acc: 0.9962 - val_loss: 0.0846 - val_acc: 0.9771
Epoch 10/20
48000/48000 [=====] - 5s 114us/step - loss: 0.0101 - acc: 0.9971 - val_loss: 0.0815 - val_acc: 0.9805
Epoch 11/20
48000/48000 [=====] - 4s 75us/step - loss: 0.0071 - accuracy: 0.9979 - val_loss: 0.0901 - val_acc: 0.9802
Epoch 12/20
48000/48000 [=====] - 4s 88us/step - loss: 0.0058 - accuracy: 0.9985 - val_loss: 0.0862 - val_acc: 0.9795
Epoch 13/20
48000/48000 [=====] - 5s 110us/step - loss: 0.0042 - acc: 0.9989 - val_loss: 0.0912 - val_acc: 0.9800
Epoch 14/20
48000/48000 [=====] - 5s 109us/step - loss: 0.0035 - acc: 0.9992 - val_loss: 0.0947 - val_acc: 0.9797
Epoch 15/20
48000/48000 [=====] - 6s 118us/step - loss: 0.0024 - acc: 0.9995 - val_loss: 0.1005 - val_acc: 0.9800
Epoch 16/20
48000/48000 [=====] - 5s 94us/step - loss: 0.0020 - accuracy: 0.9994 - val_loss: 0.0978 - val_acc: 0.9796
Epoch 17/20
48000/48000 [=====] - 4s 91us/step - loss: 0.0016 - accuracy: 0.9996 - val_loss: 0.0997 - val_acc: 0.9804
Epoch 18/20
48000/48000 [=====] - 4s 77us/step - loss: 0.0012 - accuracy: 0.9997 - val_loss: 0.1009 - val_acc: 0.9808
Epoch 19/20
48000/48000 [=====] - 5s 103us/step - loss: 8.4412e-04 - accuracy: 0.9998 - val_loss: 0.1145 - val_acc: 0.9792
Epoch 20/20
48000/48000 [=====] - 5s 108us/step - loss: 8.0652e-04 - accuracy: 0.9998 - val_loss: 0.1158 - val_acc: 0.9796
10000/10000 [=====] - 1s 86us/step
test_loss: 0.09473391755018697
test_acc: 0.9813

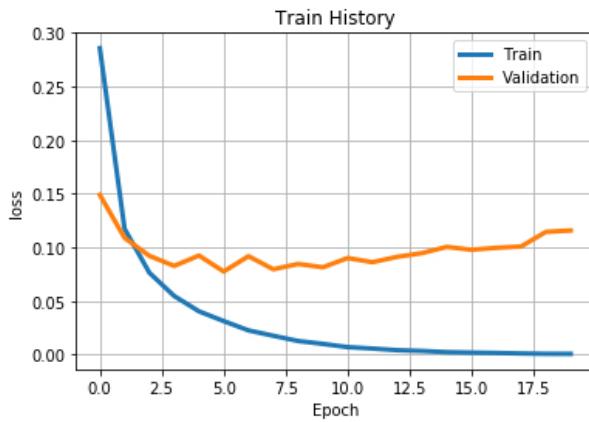
```

```

In [11]: 1 epoch = 20
2 show_train_history(train_history, 'acc', 'val_acc', 20)
3 show_train_history(train_history, 'loss', 'val_loss', 20)

```





```
In [12]: 1 #評估模型準確率
2 scores_20 = network.evaluate(train_images, train_labels)
3 print()
4 print("[Info] Accuracy of testing data = {:.2f}%".format(scores_20[1]*100.0))
5
6 60000/60000 [=====] - 4s 73us/step
7 [Info] Accuracy of testing data = 99.5%
```

```
In [13]: 1 #epochs=50
2 network = models.Sequential()
3 network.add(layers.Dense(512, activation='relu', input_shape=(28 * 28,)))
4 network.add(layers.Dense(10, activation='softmax'))
5 network.compile(optimizer='rmsprop', loss='categorical_crossentropy', metrics=['accuracy'])
6 train_history = network.fit(train_images, train_labels, validation_split=0.2,
7 test_loss, test_acc = network.evaluate(test_images, test_labels)
8 print('test_loss:', test_loss)
9 print('test_acc:', test_acc)
10 #print(train_history.history)

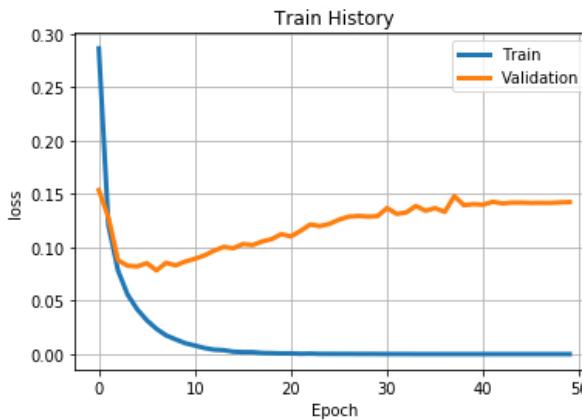
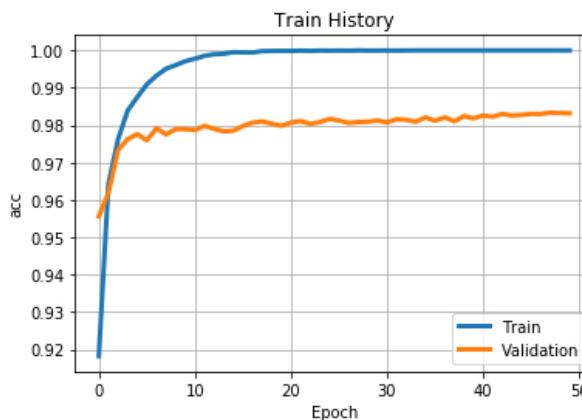
Train on 48000 samples, validate on 12000 samples
Epoch 1/50
48000/48000 [=====] - 5s 105us/step - loss: 0.2863 - acc: 0.9182 - val_loss: 0.1536 - val_acc: 0.9556
Epoch 2/50
48000/48000 [=====] - 5s 100us/step - loss: 0.1204 - acc: 0.9642 - val_loss: 0.1281 - val_acc: 0.9618
Epoch 3/50
48000/48000 [=====] - 5s 114us/step - loss: 0.0781 - acc: 0.9762 - val_loss: 0.0879 - val_acc: 0.9733
Epoch 4/50
48000/48000 [=====] - 4s 91us/step - loss: 0.0556 - acc: 0.9839 - val_loss: 0.0828 - val_acc: 0.9762
Epoch 5/50
48000/48000 [=====] - 4s 76us/step - loss: 0.0419 - acc: 0.9875 - val_loss: 0.0821 - val_acc: 0.9776
Epoch 6/50
48000/48000 [=====] - 4s 76us/step - loss: 0.0316 - acc: 0.9909 - val_loss: 0.0851 - val_acc: 0.9759
Epoch 7/50
48000/48000 [=====] - 3s 71us/step - loss: 0.0236 - acc: 0.9932 - val_loss: 0.0783 - val_acc: 0.9792
Epoch 8/50
48000/48000 [=====] - 5s 108us/step - loss: 0.0176 - acc: 0.9951 - val_loss: 0.0855 - val_acc: 0.9775
Epoch 9/50
48000/48000 [=====] - 4s 91us/step - loss: 0.0138 - acc: 0.9961 - val_loss: 0.0831 - val_acc: 0.9789
Epoch 10/50
48000/48000 [=====] - 5s 101us/step - loss: 0.0102 - acc: 0.9971 - val_loss: 0.0867 - val_acc: 0.9789
Epoch 11/50
48000/48000 [=====] - 5s 108us/step - loss: 0.0080 - acc: 0.9978 - val_loss: 0.0893 - val_acc: 0.9787
Epoch 12/50
48000/48000 [=====] - 5s 103us/step - loss: 0.0057 - acc: 0.9985 - val_loss: 0.0924 - val_acc: 0.9798
Epoch 13/50
48000/48000 [=====] - 5s 109us/step - loss: 0.0041 - acc: 0.9990 - val_loss: 0.0969 - val_acc: 0.9790
Epoch 14/50
48000/48000 [=====] - 5s 105us/step - loss: 0.0037 - acc:
```

```
cc: 0.9991 - val_loss: 0.1004 - val_acc: 0.9783
Epoch 15/50
48000/48000 [=====] - 5s 110us/step - loss: 0.0023 - a
cc: 0.9995 - val_loss: 0.0991 - val_acc: 0.9785
Epoch 16/50
48000/48000 [=====] - 5s 106us/step - loss: 0.0019 - a
cc: 0.9995 - val_loss: 0.1030 - val_acc: 0.9797
Epoch 17/50
48000/48000 [=====] - 5s 107us/step - loss: 0.0019 - a
cc: 0.9994 - val_loss: 0.1021 - val_acc: 0.9807
Epoch 18/50
48000/48000 [=====] - 5s 105us/step - loss: 0.0011 - a
cc: 0.9998 - val_loss: 0.1054 - val_acc: 0.9810
Epoch 19/50
48000/48000 [=====] - 5s 102us/step - loss: 8.8951e-04
- acc: 0.9999 - val_loss: 0.1077 - val_acc: 0.9804
Epoch 20/50
48000/48000 [=====] - 5s 102us/step - loss: 6.6543e-04
- acc: 0.9999 - val_loss: 0.1123 - val_acc: 0.9798
Epoch 21/50
48000/48000 [=====] - 5s 104us/step - loss: 6.7314e-04
- acc: 0.9999 - val_loss: 0.1102 - val_acc: 0.9807
Epoch 22/50
48000/48000 [=====] - 5s 108us/step - loss: 3.1067e-04
- acc: 0.9999 - val_loss: 0.1154 - val_acc: 0.9811
Epoch 23/50
48000/48000 [=====] - 5s 109us/step - loss: 5.0515e-04
- acc: 0.9999 - val_loss: 0.1213 - val_acc: 0.9803
Epoch 24/50
48000/48000 [=====] - 5s 107us/step - loss: 2.2175e-04
- acc: 1.0000 - val_loss: 0.1199 - val_acc: 0.9808
Epoch 25/50
48000/48000 [=====] - 5s 104us/step - loss: 2.2427e-04
- acc: 0.9999 - val_loss: 0.1219 - val_acc: 0.9817
Epoch 26/50
48000/48000 [=====] - 5s 104us/step - loss: 1.8803e-04
- acc: 1.0000 - val_loss: 0.1259 - val_acc: 0.9812
Epoch 27/50
48000/48000 [=====] - 5s 110us/step - loss: 2.1945e-04
- acc: 0.9999 - val_loss: 0.1286 - val_acc: 0.9806
Epoch 28/50
48000/48000 [=====] - 5s 102us/step - loss: 1.2211e-04
- acc: 1.0000 - val_loss: 0.1294 - val_acc: 0.9808
Epoch 29/50
48000/48000 [=====] - 5s 107us/step - loss: 1.5941e-04
- acc: 1.0000 - val_loss: 0.1287 - val_acc: 0.9809
Epoch 30/50
48000/48000 [=====] - 5s 107us/step - loss: 1.3631e-04
- acc: 1.0000 - val_loss: 0.1292 - val_acc: 0.9812
Epoch 31/50
48000/48000 [=====] - 5s 97us/step - loss: 6.0368e-05
- acc: 1.0000 - val_loss: 0.1368 - val_acc: 0.9808
Epoch 32/50
48000/48000 [=====] - 5s 103us/step - loss: 8.3929e-05
- acc: 1.0000 - val_loss: 0.1313 - val_acc: 0.9816
Epoch 33/50
48000/48000 [=====] - 5s 99us/step - loss: 5.5324e-05
- acc: 1.0000 - val_loss: 0.1327 - val_acc: 0.9814
Epoch 34/50
48000/48000 [=====] - 5s 105us/step - loss: 1.3720e-05
- acc: 1.0000 - val_loss: 0.1387 - val_acc: 0.9809
Epoch 35/50
48000/48000 [=====] - 5s 108us/step - loss: 1.4043e-05
- acc: 1.0000 - val_loss: 0.1342 - val_acc: 0.9821
Epoch 36/50
48000/48000 [=====] - 5s 104us/step - loss: 9.5863e-06
- acc: 1.0000 - val_loss: 0.1367 - val_acc: 0.9812
Epoch 37/50
48000/48000 [=====] - 5s 108us/step - loss: 8.9915e-06
- acc: 1.0000 - val_loss: 0.1332 - val_acc: 0.9821
Epoch 38/50
48000/48000 [=====] - 5s 98us/step - loss: 3.7345e-06
- acc: 1.0000 - val_loss: 0.1480 - val_acc: 0.9810
Epoch 39/50
48000/48000 [=====] - 5s 109us/step - loss: 4.8787e-06
- acc: 1.0000 - val_loss: 0.1394 - val_acc: 0.9824
Epoch 40/50
48000/48000 [=====] - 5s 104us/step - loss: 5.5385e-07
- acc: 1.0000 - val_loss: 0.1404 - val_acc: 0.9818
Epoch 41/50
48000/48000 [=====] - 5s 110us/step - loss: 2.3466e-07
```

```
- acc: 1.0000 - val_loss: 0.1399 - val_acc: 0.9826
Epoch 42/50
48000/48000 [=====] - 5s 105us/step - loss: 3.7684e-07
- acc: 1.0000 - val_loss: 0.1426 - val_acc: 0.9822
Epoch 43/50
48000/48000 [=====] - 5s 100us/step - loss: 5.0378e-07
- acc: 1.0000 - val_loss: 0.1411 - val_acc: 0.9830
Epoch 44/50
48000/48000 [=====] - 5s 104us/step - loss: 1.4684e-07
- acc: 1.0000 - val_loss: 0.1418 - val_acc: 0.9826
Epoch 45/50
48000/48000 [=====] - 5s 106us/step - loss: 1.4005e-07
- acc: 1.0000 - val_loss: 0.1418 - val_acc: 0.9827
Epoch 46/50
48000/48000 [=====] - 5s 106us/step - loss: 1.3646e-07
- acc: 1.0000 - val_loss: 0.1415 - val_acc: 0.9830
Epoch 47/50
48000/48000 [=====] - 5s 106us/step - loss: 1.3416e-07
- acc: 1.0000 - val_loss: 0.1416 - val_acc: 0.9830
Epoch 48/50
48000/48000 [=====] - 5s 107us/step - loss: 1.3249e-07
- acc: 1.0000 - val_loss: 0.1415 - val_acc: 0.9833
Epoch 49/50
48000/48000 [=====] - 4s 93us/step - loss: 1.3157e-07
- acc: 1.0000 - val_loss: 0.1420 - val_acc: 0.9832
Epoch 50/50
48000/48000 [=====] - 5s 95us/step - loss: 1.3007e-07
- acc: 1.0000 - val_loss: 0.1424 - val_acc: 0.9832
10000/10000 [=====] - 1s 123us/step
test_loss: 0.13545972370512
test_acc: 0.9823
```

In [14]:

```
1 epoch = 50
2 show_train_history(train_history, 'acc', 'val_acc', 50)
3 show_train_history(train_history, 'loss', 'val_loss', 50)
```



In [15]:

```
1 #評估模型準確率
2 scores_50 = network.evaluate(train_images, train_labels)
3 print()
4 print("[Info] Accuracy of testing data = {:.2f}%".format(scores_50[1]*100.0))
```

```
60000/60000 [=====] - 6s 96us/step
```

```
[Info] Accuracy of testing data = 99.7%
```

```
In [16]: 1 #epochs=100
2 network = models.Sequential()
3 network.add(layers.Dense(512, activation='relu', input_shape=(28 * 28,)))
4 network.add(layers.Dense(10, activation='softmax'))
5 network.compile(optimizer='rmsprop', loss='categorical_crossentropy', metrics=['accuracy'])
6 train_history = network.fit(train_images, train_labels, validation_split=0.2,
7 test_loss, test_acc = network.evaluate(test_images, test_labels)
8 print('test_loss:', test_loss)
9 print('test_acc:', test_acc)
10 #print(train_history.history)
```

Train on 48000 samples, validate on 12000 samples

Epoch 1/100
48000/48000 [=====] - 5s 114us/step - loss: 0.2826 - acc: 0.9177 - val_loss: 0.1495 - val_acc: 0.9557

Epoch 2/100
48000/48000 [=====] - 5s 113us/step - loss: 0.1169 - acc: 0.9651 - val_loss: 0.1009 - val_acc: 0.9699

Epoch 3/100
48000/48000 [=====] - 5s 101us/step - loss: 0.0771 - acc: 0.9775 - val_loss: 0.0914 - val_acc: 0.9732

Epoch 4/100
48000/48000 [=====] - 5s 113us/step - loss: 0.0554 - acc: 0.9838 - val_loss: 0.0821 - val_acc: 0.9763

Epoch 5/100
48000/48000 [=====] - 5s 104us/step - loss: 0.0407 - acc: 0.9879 - val_loss: 0.0776 - val_acc: 0.9775

Epoch 6/100
48000/48000 [=====] - 5s 102us/step - loss: 0.0304 - acc: 0.9910 - val_loss: 0.0803 - val_acc: 0.9779

Epoch 7/100
48000/48000 [=====] - 5s 112us/step - loss: 0.0228 - acc: 0.9936 - val_loss: 0.0829 - val_acc: 0.9779

Epoch 8/100
48000/48000 [=====] - 6s 115us/step - loss: 0.0180 - acc: 0.9947 - val_loss: 0.0889 - val_acc: 0.9757

Epoch 9/100
48000/48000 [=====] - 5s 98us/step - loss: 0.0133 - acc: 0.9961 - val_loss: 0.0819 - val_acc: 0.9783

Epoch 10/100
48000/48000 [=====] - 5s 107us/step - loss: 0.0105 - acc: 0.9970 - val_loss: 0.0830 - val_acc: 0.9792

Epoch 11/100
48000/48000 [=====] - 6s 117us/step - loss: 0.0075 - acc: 0.9981 - val_loss: 0.0882 - val_acc: 0.9793

Epoch 12/100
48000/48000 [=====] - 6s 125us/step - loss: 0.0057 - acc: 0.9984 - val_loss: 0.0921 - val_acc: 0.9799

Epoch 13/100
48000/48000 [=====] - 6s 118us/step - loss: 0.0044 - acc: 0.9989 - val_loss: 0.0919 - val_acc: 0.9800

Epoch 14/100
48000/48000 [=====] - 6s 117us/step - loss: 0.0036 - acc: 0.9990 - val_loss: 0.0966 - val_acc: 0.9793

Epoch 15/100
48000/48000 [=====] - 5s 110us/step - loss: 0.0025 - acc: 0.9994 - val_loss: 0.0984 - val_acc: 0.9810

Epoch 16/100
48000/48000 [=====] - 5s 108us/step - loss: 0.0020 - acc: 0.9995 - val_loss: 0.1118 - val_acc: 0.9793

Epoch 17/100
48000/48000 [=====] - 5s 107us/step - loss: 0.0015 - acc: 0.9996 - val_loss: 0.1125 - val_acc: 0.9783

Epoch 18/100
48000/48000 [=====] - 5s 106us/step - loss: 0.0011 - acc: 0.9998 - val_loss: 0.1083 - val_acc: 0.9793

Epoch 19/100
48000/48000 [=====] - 5s 105us/step - loss: 0.0011 - acc: 0.9996 - val_loss: 0.1080 - val_acc: 0.9811

Epoch 20/100
48000/48000 [=====] - 5s 103us/step - loss: 9.7771e-04 - acc: 0.9996 - val_loss: 0.1141 - val_acc: 0.9796

Epoch 21/100
48000/48000 [=====] - 5s 109us/step - loss: 6.9523e-04 - acc: 0.9998 - val_loss: 0.1149 - val_acc: 0.9810

Epoch 22/100
48000/48000 [=====] - 5s 104us/step - loss: 3.6526e-04 - acc: 0.9999 - val_loss: 0.1225 - val_acc: 0.9796

Epoch 23/100
48000/48000 [=====] - 5s 110us/step - loss: 3.6239e-04 - acc: 0.9999 - val_loss: 0.1197 - val_acc: 0.9802

```
Epoch 24/100
48000/48000 [=====] - 5s 110us/step - loss: 3.1926e-04
- acc: 0.9999 - val_loss: 0.1210 - val_acc: 0.9803
Epoch 25/100
48000/48000 [=====] - 5s 104us/step - loss: 4.0485e-04
- acc: 0.9999 - val_loss: 0.1200 - val_acc: 0.9811
Epoch 26/100
48000/48000 [=====] - 4s 82us/step - loss: 3.1143e-04
- acc: 0.9999 - val_loss: 0.1260 - val_acc: 0.9812
Epoch 27/100
48000/48000 [=====] - 4s 75us/step - loss: 2.7454e-04
- acc: 0.9999 - val_loss: 0.1227 - val_acc: 0.9818
Epoch 28/100
48000/48000 [=====] - 4s 85us/step - loss: 1.5232e-04
- acc: 1.0000 - val_loss: 0.1283 - val_acc: 0.9806
Epoch 29/100
48000/48000 [=====] - 5s 95us/step - loss: 6.3561e-05
- acc: 1.0000 - val_loss: 0.1334 - val_acc: 0.9800
Epoch 30/100
48000/48000 [=====] - 5s 102us/step - loss: 5.8451e-05
- acc: 1.0000 - val_loss: 0.1262 - val_acc: 0.9812
Epoch 31/100
48000/48000 [=====] - 5s 108us/step - loss: 1.4983e-04
- acc: 0.9999 - val_loss: 0.1316 - val_acc: 0.9812
Epoch 32/100
48000/48000 [=====] - 5s 108us/step - loss: 2.4620e-05
- acc: 1.0000 - val_loss: 0.1375 - val_acc: 0.9800
Epoch 33/100
48000/48000 [=====] - 5s 107us/step - loss: 2.3495e-05
- acc: 1.0000 - val_loss: 0.1310 - val_acc: 0.9819
Epoch 34/100
48000/48000 [=====] - 5s 108us/step - loss: 1.2521e-05
- acc: 1.0000 - val_loss: 0.1368 - val_acc: 0.9815
Epoch 35/100
48000/48000 [=====] - 5s 108us/step - loss: 1.5495e-05
- acc: 1.0000 - val_loss: 0.1341 - val_acc: 0.9819
Epoch 36/100
48000/48000 [=====] - 5s 110us/step - loss: 1.1720e-05
- acc: 1.0000 - val_loss: 0.1414 - val_acc: 0.9802
Epoch 37/100
48000/48000 [=====] - 5s 109us/step - loss: 1.0568e-05
- acc: 1.0000 - val_loss: 0.1330 - val_acc: 0.9819
Epoch 38/100
48000/48000 [=====] - 5s 101us/step - loss: 1.2263e-05
- acc: 1.0000 - val_loss: 0.1364 - val_acc: 0.9822
Epoch 39/100
48000/48000 [=====] - 5s 109us/step - loss: 1.2390e-05
- acc: 1.0000 - val_loss: 0.1401 - val_acc: 0.9807
Epoch 40/100
48000/48000 [=====] - 5s 99us/step - loss: 2.1018e-06
- acc: 1.0000 - val_loss: 0.1380 - val_acc: 0.9818
Epoch 41/100
48000/48000 [=====] - 4s 91us/step - loss: 4.2000e-07
- acc: 1.0000 - val_loss: 0.1390 - val_acc: 0.9819
Epoch 42/100
48000/48000 [=====] - 5s 100us/step - loss: 1.7150e-07
- acc: 1.0000 - val_loss: 0.1380 - val_acc: 0.9823
Epoch 43/100
48000/48000 [=====] - 5s 105us/step - loss: 1.4930e-07
- acc: 1.0000 - val_loss: 0.1375 - val_acc: 0.9823
Epoch 44/100
48000/48000 [=====] - 5s 107us/step - loss: 1.4208e-07
- acc: 1.0000 - val_loss: 0.1378 - val_acc: 0.9827
Epoch 45/100
48000/48000 [=====] - 5s 102us/step - loss: 1.3824e-07
- acc: 1.0000 - val_loss: 0.1382 - val_acc: 0.9827
Epoch 46/100
48000/48000 [=====] - 5s 106us/step - loss: 1.3568e-07
- acc: 1.0000 - val_loss: 0.1385 - val_acc: 0.9825
Epoch 47/100
48000/48000 [=====] - 5s 107us/step - loss: 1.3384e-07
- acc: 1.0000 - val_loss: 0.1388 - val_acc: 0.9824
Epoch 48/100
48000/48000 [=====] - 5s 104us/step - loss: 1.3225e-07
- acc: 1.0000 - val_loss: 0.1388 - val_acc: 0.9827
Epoch 49/100
48000/48000 [=====] - 5s 107us/step - loss: 1.3110e-07
- acc: 1.0000 - val_loss: 0.1386 - val_acc: 0.9826
Epoch 50/100
48000/48000 [=====] - 5s 107us/step - loss: 1.2986e-07
- acc: 1.0000 - val_loss: 0.1393 - val_acc: 0.9824
```

```
Epoch 51/100
48000/48000 [=====] - 5s 106us/step - loss: 1.2909e-07
- acc: 1.0000 - val_loss: 0.1392 - val_acc: 0.9826
Epoch 52/100
48000/48000 [=====] - 5s 101us/step - loss: 1.2828e-07
- acc: 1.0000 - val_loss: 0.1392 - val_acc: 0.9826
Epoch 53/100
48000/48000 [=====] - 5s 107us/step - loss: 1.2752e-07
- acc: 1.0000 - val_loss: 0.1393 - val_acc: 0.9825
Epoch 54/100
48000/48000 [=====] - 5s 109us/step - loss: 1.2682e-07
- acc: 1.0000 - val_loss: 0.1392 - val_acc: 0.9827
Epoch 55/100
48000/48000 [=====] - 5s 102us/step - loss: 1.2640e-07
- acc: 1.0000 - val_loss: 0.1391 - val_acc: 0.9828
Epoch 56/100
48000/48000 [=====] - 5s 109us/step - loss: 1.2587e-07
- acc: 1.0000 - val_loss: 0.1395 - val_acc: 0.9828
Epoch 57/100
48000/48000 [=====] - 5s 109us/step - loss: 1.2543e-07
- acc: 1.0000 - val_loss: 0.1399 - val_acc: 0.9825
Epoch 58/100
48000/48000 [=====] - 5s 107us/step - loss: 1.2501e-07
- acc: 1.0000 - val_loss: 0.1399 - val_acc: 0.9826
Epoch 59/100
48000/48000 [=====] - 5s 107us/step - loss: 1.2464e-07
- acc: 1.0000 - val_loss: 0.1398 - val_acc: 0.9827
Epoch 60/100
48000/48000 [=====] - 5s 110us/step - loss: 1.2436e-07
- acc: 1.0000 - val_loss: 0.1401 - val_acc: 0.9824
Epoch 61/100
48000/48000 [=====] - 5s 107us/step - loss: 1.2405e-07
- acc: 1.0000 - val_loss: 0.1401 - val_acc: 0.9828
Epoch 62/100
48000/48000 [=====] - 5s 107us/step - loss: 1.2372e-07
- acc: 1.0000 - val_loss: 0.1403 - val_acc: 0.9826
Epoch 63/100
48000/48000 [=====] - 5s 101us/step - loss: 1.2351e-07
- acc: 1.0000 - val_loss: 0.1402 - val_acc: 0.9826
Epoch 64/100
48000/48000 [=====] - 5s 114us/step - loss: 1.2323e-07
- acc: 1.0000 - val_loss: 0.1403 - val_acc: 0.9827
Epoch 65/100
48000/48000 [=====] - 5s 98us/step - loss: 1.2302e-07
- acc: 1.0000 - val_loss: 0.1404 - val_acc: 0.9828
Epoch 66/100
48000/48000 [=====] - 5s 107us/step - loss: 1.2283e-07
- acc: 1.0000 - val_loss: 0.1406 - val_acc: 0.9826
Epoch 67/100
48000/48000 [=====] - 5s 102us/step - loss: 1.2260e-07
- acc: 1.0000 - val_loss: 0.1406 - val_acc: 0.9826
Epoch 68/100
48000/48000 [=====] - 5s 105us/step - loss: 1.2245e-07
- acc: 1.0000 - val_loss: 0.1406 - val_acc: 0.9826
Epoch 69/100
48000/48000 [=====] - 4s 86us/step - loss: 1.2224e-07
- acc: 1.0000 - val_loss: 0.1406 - val_acc: 0.9827
Epoch 70/100
48000/48000 [=====] - 12s 242us/step - loss: 1.2208e-07
- acc: 1.0000 - val_loss: 0.1407 - val_acc: 0.9828
Epoch 71/100
48000/48000 [=====] - 21s 433us/step - loss: 1.2192e-07
- acc: 1.0000 - val_loss: 0.1407 - val_acc: 0.9828
Epoch 72/100
48000/48000 [=====] - 6s 127us/step - loss: 1.2179e-07
- acc: 1.0000 - val_loss: 0.1408 - val_acc: 0.9827
Epoch 73/100
48000/48000 [=====] - 4s 87us/step - loss: 1.2165e-07
- acc: 1.0000 - val_loss: 0.1408 - val_acc: 0.9827
Epoch 74/100
48000/48000 [=====] - 4s 78us/step - loss: 1.2157e-07
- acc: 1.0000 - val_loss: 0.1410 - val_acc: 0.9828
Epoch 75/100
48000/48000 [=====] - 4s 78us/step - loss: 1.2142e-07
- acc: 1.0000 - val_loss: 0.1410 - val_acc: 0.9826
Epoch 76/100
48000/48000 [=====] - 4s 82us/step - loss: 1.2133e-07
- acc: 1.0000 - val_loss: 0.1412 - val_acc: 0.9827
Epoch 77/100
48000/48000 [=====] - 5s 104us/step - loss: 1.2123e-07
- acc: 1.0000 - val_loss: 0.1411 - val_acc: 0.9827
```

```

Epoch 78/100
48000/48000 [=====] - 5s 109us/step - loss: 1.2113e-07
- acc: 1.0000 - val_loss: 0.1411 - val_acc: 0.9828
Epoch 79/100
48000/48000 [=====] - 5s 111us/step - loss: 1.2105e-07
- acc: 1.0000 - val_loss: 0.1412 - val_acc: 0.9826
Epoch 80/100
48000/48000 [=====] - 6s 117us/step - loss: 1.2096e-07
- acc: 1.0000 - val_loss: 0.1414 - val_acc: 0.9827
Epoch 81/100
48000/48000 [=====] - 5s 106us/step - loss: 1.2089e-07
- acc: 1.0000 - val_loss: 0.1413 - val_acc: 0.9828
Epoch 82/100
48000/48000 [=====] - 5s 105us/step - loss: 1.2080e-07
- acc: 1.0000 - val_loss: 0.1414 - val_acc: 0.9828
Epoch 83/100
48000/48000 [=====] - 4s 84us/step - loss: 1.2072e-07
- acc: 1.0000 - val_loss: 0.1414 - val_acc: 0.9828
Epoch 84/100
48000/48000 [=====] - 4s 94us/step - loss: 1.2066e-07
- acc: 1.0000 - val_loss: 0.1415 - val_acc: 0.9827
Epoch 85/100
48000/48000 [=====] - 5s 99us/step - loss: 1.2059e-07
- acc: 1.0000 - val_loss: 0.1414 - val_acc: 0.9827
Epoch 86/100
48000/48000 [=====] - 5s 109us/step - loss: 1.2054e-07
- acc: 1.0000 - val_loss: 0.1413 - val_acc: 0.9827
Epoch 87/100
48000/48000 [=====] - 5s 112us/step - loss: 1.2048e-07
- acc: 1.0000 - val_loss: 0.1415 - val_acc: 0.9827
Epoch 88/100
48000/48000 [=====] - 6s 125us/step - loss: 1.2041e-07
- acc: 1.0000 - val_loss: 0.1416 - val_acc: 0.9827
Epoch 89/100
48000/48000 [=====] - 5s 102us/step - loss: 1.2035e-07
- acc: 1.0000 - val_loss: 0.1416 - val_acc: 0.9826
Epoch 90/100
48000/48000 [=====] - 5s 102us/step - loss: 1.2032e-07
- acc: 1.0000 - val_loss: 0.1416 - val_acc: 0.9827
Epoch 91/100
48000/48000 [=====] - 5s 111us/step - loss: 1.2025e-07
- acc: 1.0000 - val_loss: 0.1417 - val_acc: 0.9828
Epoch 92/100
48000/48000 [=====] - 5s 99us/step - loss: 1.2024e-07
- acc: 1.0000 - val_loss: 0.1417 - val_acc: 0.9827
Epoch 93/100
48000/48000 [=====] - 4s 89us/step - loss: 1.2017e-07
- acc: 1.0000 - val_loss: 0.1417 - val_acc: 0.9828
Epoch 94/100
48000/48000 [=====] - 5s 105us/step - loss: 1.2012e-07
- acc: 1.0000 - val_loss: 0.1417 - val_acc: 0.9827
Epoch 95/100
48000/48000 [=====] - 5s 97us/step - loss: 1.2009e-07
- acc: 1.0000 - val_loss: 0.1418 - val_acc: 0.9826
Epoch 96/100
48000/48000 [=====] - 5s 108us/step - loss: 1.2005e-07
- acc: 1.0000 - val_loss: 0.1418 - val_acc: 0.9828
Epoch 97/100
48000/48000 [=====] - 3s 73us/step - loss: 1.2003e-07
- acc: 1.0000 - val_loss: 0.1419 - val_acc: 0.9826
Epoch 98/100
48000/48000 [=====] - 4s 91us/step - loss: 1.2000e-07
- acc: 1.0000 - val_loss: 0.1419 - val_acc: 0.9828
Epoch 99/100
48000/48000 [=====] - 5s 94us/step - loss: 1.1997e-07
- acc: 1.0000 - val_loss: 0.1420 - val_acc: 0.9826
Epoch 100/100
48000/48000 [=====] - 6s 115us/step - loss: 1.1992e-07
- acc: 1.0000 - val_loss: 0.1419 - val_acc: 0.9828
10000/10000 [=====] - 1s 81us/step
test_loss: 0.13257047040182926
test_acc: 0.9834

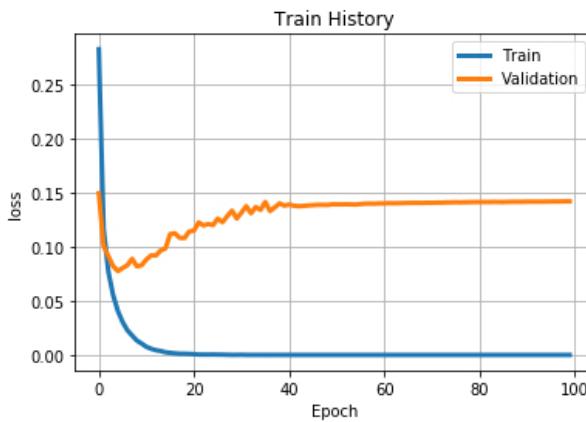
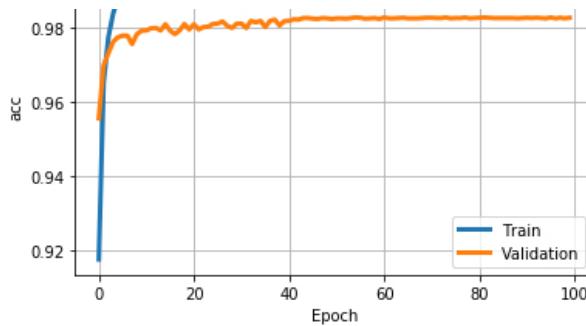
```

```

In [17]: 1 epoch = 100
2 show_train_history(train_history, 'acc', 'val_acc', 100)
3 show_train_history(train_history, 'loss', 'val_loss', 100)

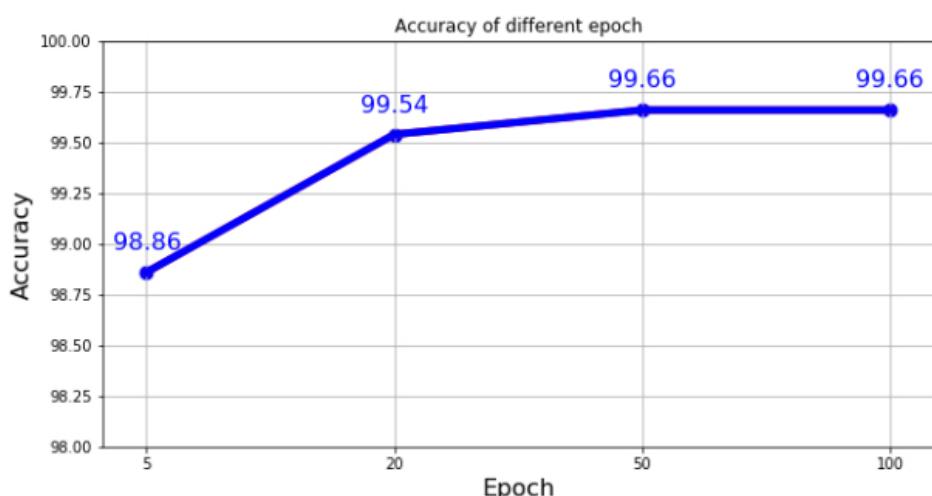
```





```
In [18]: 1 #評估模型準確率
2 scores_100 = network.evaluate(train_images, train_labels)
3 print()
4 print("[Info] Accuracy of testing data = {:.2f}%".format(scores_100[1]*100.0))
5
6 60000/60000 [=====] - 6s 92us/step
7
8 [Info] Accuracy of testing data = 99.7%
```

```
In [24]: 1 x = ['5','20','50','100']
2 accuracy = [round(scores_5[1]*100.0,2),round(scores_20[1]*100.0,2),round(scores_50[1]*100.0,2),round(scores_100[1]*100.0,2)]
3 plt.figure(figsize=(10,5))
4 plt.plot(x,accuracy,'r',lw=5)
5 plt.title('Accuracy of different epoch')
6 plt.ylabel('Accuracy',fontsize=16)
7 plt.xlabel('Epoch',fontsize=16)
8 plt.ylim(98,100)
9 plt.plot(x,accuracy,'blue',lw=5)
10 plt.scatter(x, accuracy, s = 75,color='b')
11 plt.xticks(range(4),['5','20','50','100'])
12 plt.grid(True)
13 for x,y in enumerate(accuracy):
14     plt.text(x,y+0.11,'%s %s' % (x,y), color='b', fontsize=16, ha='center')
15 plt.savefig('Accuracy of different epoch.jpg',dpi=300)
```



```
In [ ]:
```

```
1
```

