# coding: utf-8

import sys, os

sys.path.append(os.pardir) # 用于导入父目录中的文件的设置import numpy as np

import matplotlib.pyplot as plt

from dataset.mnist import load\_mnist

#from simple\_convnet import SimpleConvNet

from common.trainer import Trainer

#导入数据

(x\_train, t\_train), (x\_test, t\_test) = load\_mnist(flatten=False)

# 在处理速度慢的情况下减少数据

#x\_train, t\_train = x\_train[:5000], t\_train[:5000]

#x\_test, t\_test = x\_test[:1000], t\_test[:1000]

max\_epochs = 20

network = SimpleConvNet(input\_dim=(1,28,28),

conv\_param = {'filter\_num': 30, 'filter\_size': 5, 'pad': 0, 'stride': 1},

hidden\_size=100, output\_size=10, weight\_init\_std=0.01)

trainer = Trainer(network, x\_train, t\_train, x\_test, t\_test,

epochs=max\_epochs, mini\_batch\_size=100,

optimizer='Adam', optimizer\_param={'lr': 0.001},

evaluate\_sample\_num\_per\_epoch=1000)

trainer.train()

# 保存参数

network.save\_params("params.pkl")

print("Saved Network Parameters!")

# 绘制图表

markers = {'train': 'o', 'test': 's'}

x = np.arange(max\_epochs)

plt.plot(x, trainer.train\_acc\_list, marker='o', label='train', markevery=2)

plt.plot(x, trainer.test\_acc\_list, marker='s', label='test', markevery=2)

plt.xlabel("epochs")

plt.ylabel("accuracy")

plt.ylim(0, 1.0)

plt.legend(loc='lower right')

plt.show()