



```
train-images-idx3-ubyte: training set images
train-labels-idx1-ubyte: training set labels
t10k-images-idx3-ubyte:  test set images
t10k-labels-idx1-ubyte:  test set labels
```

更多 MNIST 信息参考: <http://yann.lecun.com/exdb/mnist/>

## 提供的文件

- ①MNIST\_data: 包含了上文数据格式部分介绍的四个文件;
- ②load\_samples.py: 加载 MNIST\_data 的数据, 以便训练和测试;
- ③NeuralNet.py: BP 网络实现代码;
- ④main.py: 可执行文件, 运行该文件可以直接调用 load\_samples.py 和 NeuralNet.py.

## 作业内容

- ①完善 NeuralNet.py 中的 Sigmoid 函数

提示:  $z$  为函数输入值, return 其 Sigmoid 函数计算结果。

- ②完善 NeuralNet.py 中的 sigmoid 导数函数

提示:  $z$  为函数输入值, return 其导函数计算结果。

- ③完善 NeuralNet.py 中 BP 网络的 feedforward 函数

提示: NeuralNet 类中第一个函数 \_\_init\_\_ 分别产生了两套  $w$  和  $b$ , 存储在 self.w\_ 和 self.b\_ 中。self.w\_ 的两个  $w$  维度分别为  $40 \times 784$ ,  $10 \times 40$ ; self.b\_ 的两个  $b$  维度分别为  $40 \times 1$ ,  $10 \times 1$ 。return 内容为 feedforward 最后一层的计算结果。

前三题的代码请直接在 NeuralNet.py 中填写 (需要补充代码的部分已用两行 # 隔出)

- ④解释 NeuralNet.py 中 BP 网络 backprop 函数的代码 (已在文件中标明)

- ⑤完成①-③后运行 main.py (python main.py), 给出最后的计算结果。

提示: py 脚本按 python3 版本编写, python2 运行可能会出错。

## 部分函数说明

`def update_mini_batch(self, mini_batch, eta):` 更新权重

`def cost_derivative(self, output_activations, y):` 返回 `output_activations` 和 `y` 之差

`def load_samples(dataset="training_data"):` 根据参数加载数据集。并将图像和标签对应起来

`def load_mnist(dataset="training_data", digits=np.arange(10), path="."):` 根据参数从文件中读取相关数据

`def evaluate(self, test_data):` 根据测试数据集对模型进行评估，看有多少识别结果和标签一致

请在 3 月 27 日 10:00 前完成作业并发送到助教邮箱：496674467@qq.com