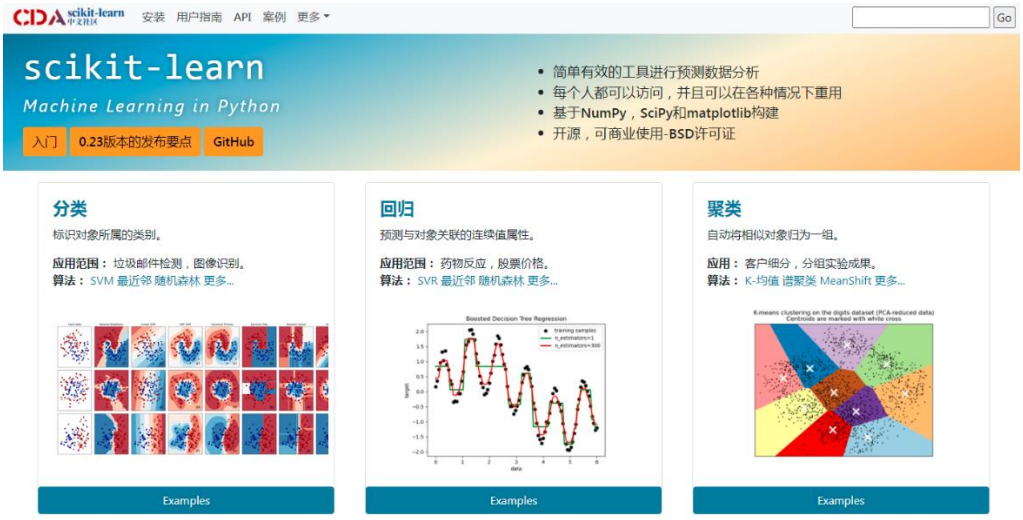
实验三 利用 scikit-learn 实现神经网络分类

准备工作（具体步骤参见实验一）：

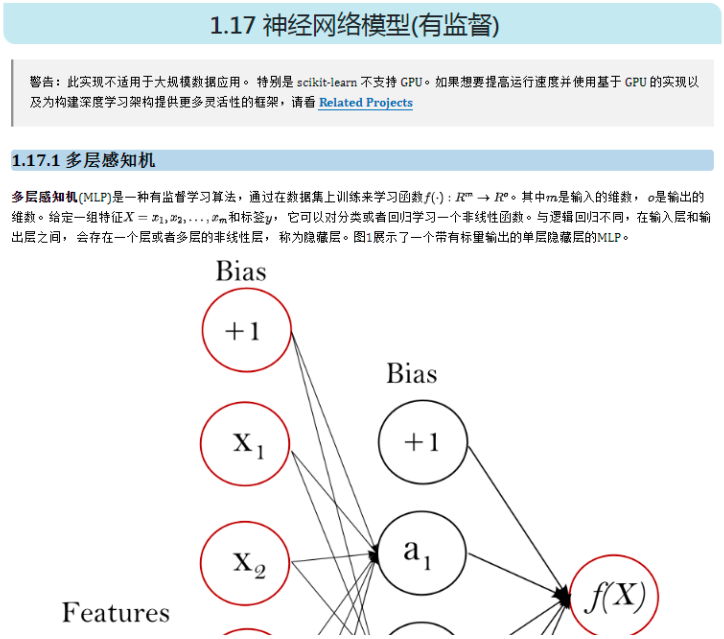
1. 在计算机上安装好 python（推荐 3.x 版本）
2. 安装编译器和编译环境
3. 安装 scikit-learn 工具包

利用 scikit-learn 实现神经网络分类：

1. 了解 scikit-learn 工具包
   英文主页：https://scikit-learn.org/stable/
   中文社区：https://scikit-learn.org.cn/



2. 了解 scikit-learn 中的神经网络分类方法
   https://scikit-learn.org.cn/view/105.html



1.17.2 分类

`MLPClassifier` 类通过使用 **Backpropagation**进行训练实现了多层感知机（MLP）算法。

MLP 在两个数组上进行训练：大小为 (n_samples, n_features) 的数组 X，用来储存表示训练样本的浮点型特征向量；大小为 (n_samples,) 的数组 y，用来储存训练样本的目标值（类别标签）：

```
>>> from sklearn.neural_network import MLPClassifier
>>> X = [[0., 0.], [1., 1.]]
>>> y = [0, 1]
>>> clf = MLPClassifier(solver='lbfgs', alpha=1e-5,
...                     hidden_layer_sizes=(5, 2), random_state=1)
...
>>> clf.fit(X, y)
MLPClassifier(alpha=1e-05, hidden_layer_sizes=(5, 2), random_state=1,
              solver='lbfgs')
```

经过拟合(训练)，该模型可以预测新样品的标签：

```
>>> clf.predict([[2., 2.], [-1., -2.]])
array([1, 0])
```

MLP可以拟合训练数据的非线性模型。 `clf.coefs_` 包含构成模型参数的权重矩阵：

```
>>> [coef.shape for coef in clf.coefs_]
[(2, 5), (5, 2), (2, 1)]
```

3. 学习其中的 sklearn.neural_network.MLPClassifier 方法（输入输出参数的含义、如何使用）
   https://scikit-learn.org.cn/view/713.html

sklearn.neural_network.MLPClassifier

```
class sklearn.neural_network.MLPClassifier(hidden_layer_sizes=(100, ), activation='relu', *, solver='adam', alpha=0.0001, batch_size='au
```

[源码]

多层感知器分类器。

这个模型使用LBFGS或随机梯度下降来优化对数损失函数。

版本0.18中的新功能。

参数	说明
hidden_layer_sizes	**tuple, length = n_layers - 2, default=(100,)** 第i个元素代表第i个隐藏层中的神经元数量。
activation	**{'identity', 'logistic', 'tanh', 'relu'}, default='relu'** 隐藏层的激活函数。 - 'identity'，无操作激活，用于实现线性瓶颈，返回f(x)= x - 'logistic',logistic Sigmoid函数，返回f(x)= 1 / (1 + exp(x))。 - 'tanh',双曲tan函数，返回f(x)= tanh(x)。 - 'relu',整流线性单位函数，返回f(x)= max(0，x)
solver	**{'lbfgs', 'sgd', 'adam'}, default='adam'** 权重优化的求解器。 - " lbfgs"是quasi-Newton方法族的优化程序。 - " sgd"是指随机梯度下降。 - " adam"是指Kingma，Diederik和Jimmy Ba提出的基于随机梯度的优化器

4. 利用 MLPClassifier 方法实现用神经网络对手写数字图像进行分类
   a) 了解手写数字图像数据集

一共包含 1797 个样本，每个样本包括 8*8 像素的图像和一个[0, 9]整数的标签，因此是一个 10 类的分类问题。

b) 在 scikit-learn 中载入此数据集（了解 data 的数据类型、存储方式等）

```
from sklearn import datasets
digits = datasets.load_digits()
```

c) 参考一个例子（使用的是 SVM 分类器）学习如何在此数据集上进行分类任务

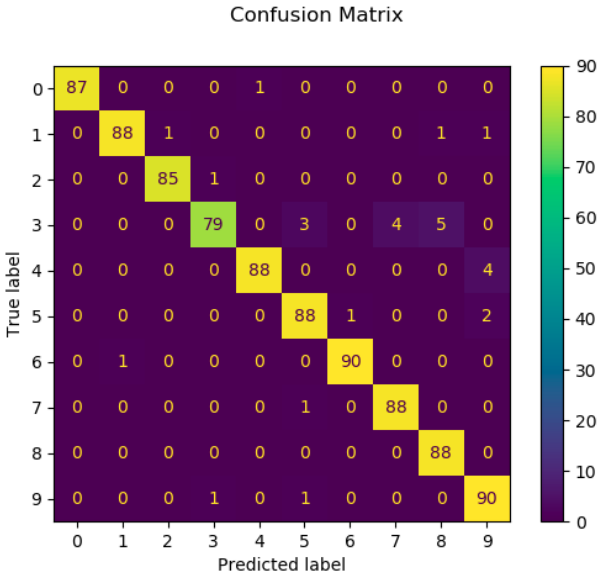
https://scikit-learn.org.cn/view/45.html

d) 使用 MLPClassifier 方法实现神经网络分类器在此数据集上的分类（参考上例输出分类结果报告 Classification report，并画出混淆矩阵 Confusion matrix）

```
Classification report for classifier SVC(gamma=0.001):
              precision    recall  f1-score   support

           0       1.00      0.99      0.99        88
           1       0.99      0.97      0.98        91
           2       0.99      0.99      0.99        86
           3       0.98      0.87      0.92        91
           4       0.99      0.96      0.97        92
           5       0.95      0.97      0.96        91
           6       0.99      0.99      0.99        91
           7       0.96      0.99      0.97        89
           8       0.94      1.00      0.97        88
           9       0.93      0.98      0.95        92

    accuracy                           0.97       899
   macro avg       0.97      0.97      0.97       899
weighted avg       0.97      0.97      0.97       899
```



e) 改变 MLPClassifier 方法中的重要参数（如隐层中的神经元数量、激活函数类型、权重优化的求解器、L2 惩罚（正则项）参数、权重更新的学习速率等），看看对分类结果有何影响？