《用 Python 玩转数据》之 scikit-learn 机器学习经典入门项目

by Dazhuang@NJU

scikit-learn 是基于 NumPy、SciPy 和 Matplotlib 的著名的 Python 机器学习包，里面包含了大量经典机器学习的数据集和算法实现，请基于经典的鸢尾花数据集 iris 实现简单的分类和聚类功能。

1. 鸢尾花数据集（iris）简介

通过如下语句可获得 iris 数据集（通过 dir(datasets)查看数据集，例如可用 datasets.load_diabetes()获得一个糖尿病病人的数据）

```
>>> from sklearn import datasets
>>> iris = datasets.load_iris()
# 数据存储在.data 属性中
>>> iris.data
# 数据中包含 150 朵鸢尾花的尺寸观测数据，每条包含萼片长度（sepal length (cm)）、萼片宽度（sepal width (cm)）、花瓣长度（petal length (cm)）、花瓣宽度（petal width (cm)）这 4 个特征值（属性）名，打印“iris”可看到相应的特征值名
>>> iris.data.shape
(150, 4)
# 数据所属种类保存在.target 属性中，共有 3 类，分别是山鸢尾（setosa），变色鸢尾（'versicolor），弗吉尼亚鸢尾（virginica），每一类各 50 条记录，打印“iris”可查看
>>> iris.target
array([0, 0, 0, ..., 2, 2, 2])
```

2. **请参考 scikit-learn 官网（http://scikit-learn.org）或本周课程中的代码或其他资源尝试用经典的分类学习算法 KNN 最近邻（k-nearest neighbor，最简单的分类算法，新的观测值的标签由 n 维空间中最靠近它的训练样本标签确定）判断萼片长度和宽度、花瓣长度和宽度分别是 5.0cm, 3.0cm, 5.0cm, 2.0cm 的鸢尾花所属类别。**
3. **请参考 scikit-learn 官网或本周课程中的代码或其他资源尝试用 k-means 聚类算法对原始数据进行聚类（3 类）并观察聚类的正确率（注意，类别用 0，1，2 表示，但并不限定表示某一类）。**

【参考代码见下一页】

【参考代码】

```
# 利用 KNN 分类算法进行分类
from sklearn import neighbors , datasets
iris = datasets.load_iris()
knn = neighbors.KNeighborsClassifier()
knn.fit(iris.data, iris.target)        # 从已有数据中学习
knn.predict([[5.0, 3.0, 5.0, 2.0]])    # 利用分类模型进行未知数据的预测（确定标签）

# 利用 k-means 聚类算法进行聚类
from sklearn import cluster, datasets
iris = datasets.load_iris()
kmeans = cluster.KMeans(n_clusters = 3).fit(iris.data)
pred = kmeans.predict(iris.data)    # 确定数据的类别
# 比较算法正确率
for label in pred:
    print(label, end = ' ')      # 打印预测出的各条数据的标签
print('\n')
for label in iris.target:
    print(label, end = ' ')      # 打印原始标注好的正确标签
```

进一步，还可以利用其他的算法实现类似的功能，例如可以利用常用的 SVM（Support Vector Machine，支持向量机）分类算法对数据进行分类：

```
from sklearn import svm, datasets
iris = datasets.load_iris()
svc = svm.LinearSVC()
svc.fit(iris.data, iris.target) # 学习
svc.predict([[ 5.0, 3.0, 5.0, 2.0]])  # 预测
```

SVM 最简单的是线性支持向量机，它尝试构建一个两个类别（不仅可用于二类分类，也可以用于多类分类）的最大间隔超平面即能将两个类别分割开的直线，但如果数据是线性不可分的，则要用到核函数，将数据映射到高维空间中寻找可区分数据的超平面。如果选择合适的 SVM 参数、核函数和特征，则在模不大的数据集上其表现不错，对机器学习感兴趣的学习者可以进行深入研究，这方面的学习并非能一蹴而就。