**cikit-learn 中 Boston Housing 数据集问题解决方案**

在部分旧教程或教材中是 sklearn，现在【2023】已经变更为 scikit-learn

* 作用：开源机器学习库，支持有监督和无监督学习。它还提供了用于模型拟合、数据预处理、模型选择、模型评估和许多其他实用程序的各种工具。
* 安装 pip install scikit-learn

**Boston Housing 数据集**

此数据集原本应该在 sklearn 中是自带数据集之一，但在 scikit-learn 1.2 版本由于某些特殊原因被移除，所以无法使用 load\_boston() 获取

解决办法：既然自带的数据集没有 Boston Housing，那就想办法在网上找到开放式公共数据集，下载后加载到程序中。这也是网上常见的解决方案，大多借助 pandas, scipy, numpy 等方法下载，然后标准化加载数据，供 scikit-learn 使用。

我将表述一下我所有使用的方法：通过从 openml.org 存储库下载数据集，我直接使用 fetch\_openml()

|  |  |
| --- | --- |
|  | from sklearn.datasets import fetch\_openml |
|  |  |
|  | data\_x, data\_y = fetch\_openml(name="boston", version=1, as\_frame=True, return\_X\_y=True, parser="pandas") |

* 其中 name 是数据集在 openml.org 上的名称
* version 是版本号，根据 openml.org 上的描述，使用 1 版本是原始数据集，所以我选择 1 版本，具体根据对应数据集的描述选择
* as\_frame=True 表示返回 pandas 的 DataFrame 格式，这样可以直接使用 pandas 的方法进行数据处理
* return\_X\_y 表示分别返回特征和标签，如果为 False 则返回一个字典【包含特征和标签】，如果你想要的是字典格式，可以设置为 False，而且默认也是 False
* parser 表示用于加载 ARFF 文件的解析器，默认的是 liac-arff
* 更复杂的参考官方文档：<https://scikit-learn.org/stable/modules/generated/sklearn.datasets.fetch_openml.html#sklearn.datasets.fetch_openml>

对 as\_frame 分不分，看下面的内容你应该会有熟悉感觉，一般在分配训练数据和测试数据时都是下面步骤，我实验需求决定，所以我直接使用 as\_frame=True 获取我想要的数据，如果你需要完整的，可以不使用 as\_frame=True

|  |  |
| --- | --- |
|  | from sklearn.model\_selection import train\_test\_split |
|  |  |
|  | train\_x, test\_x, train\_y, test\_y = train\_test\_split(data\_x, data\_y, test\_size=0.3, random\_state=1001) |

**其他问题**

使用上面可能会遇见一些问题【TypeError: can't multiply sequence by non-int of type 'float'】，一般是数据集格式问题，我在使用中是使用 numpy 进行调整的

|  |  |
| --- | --- |
|  | import numpy as np |
|  | from sklearn import linear\_model |
|  |  |
|  | model = linear\_model.LinearRegression() |
|  | model.fit(train\_x, train\_y) |
|  | pred\_y = model.predict(test\_x.astype(np.float64)) |

* 像是 predict 运算时，需要将 test\_x 转换为 np.float64 类型，反正报错时会提醒你使用什么格式的数据，根据情况进行转换就可以了

上面加载数据集时我使用 parser="pandas" 也是为了避免，sklearn 中有时对 pandas 数据格式的需求

**总结**

想办法获取远程或离线的数据集，通过 scikit-learn 自带工具或其他工具【pandas, scipy, numpy 等】加载即可使用，在使用时注意不同情况下使用的数据格式并做出对应调整。

scikit-learn 适用于存储为 numpy 数组或 scipy 稀疏矩阵的任何数字数据，因为 scikit-learn 开发中也使用这些工具。比如在上面的报错中有部分内部代码涉及 np，所以使用 numpy 转化格式就解决了报错问题。

|  |  |
| --- | --- |
|  | File /opt/conda/envs/education/lib/python3.8/site-packages/sklearn/utils/extmath.py:189, in safe\_sparse\_dot(a, b, dense\_output) |
|  | 187 ret = np.dot(a, b) |
|  | 188 else: |
|  | --> 189 ret = a @ b |