datasets

Guangyao Zhao

2022-11-10

Datasets 模块包含了机器学习领域常用的数据集，另外此模块还有人工数据生成器的功能。

Contents

加载器 1

鸢尾花数据集 1

波士顿房价 2

生成器 3

导入函数包

```
import sklearn.datasets as ds
```

加载器

鸢尾花数据集

多分类型数据集。

```
X, y = ds.load_iris(return_X_y=True, as_frame=True)

samples_num = X.shape[0]  # 样本个数
features_num = X.shape[1]  # 特征个数

print('samples:', samples_num)
```

```
print('features:', features_num)

features_names = list(X.columns)  # 特征种类
iris_names = set(y)  # 鸢尾花种类

print('features_name: ', features_names)
print('iris_names: ', iris_names)

X_head = X.head(3)  # 前三行数据
print('X_head:\n', X_head)
```

```
samples: 150
features: 4
features_name:  ['sepal length (cm)', 'sepal width (cm)', 'petal length (cm)', 'petal width (cm)']
iris_names:  {0, 1, 2}
X_head:
    sepal length (cm)  sepal width (cm)  petal length (cm)  petal width (cm)
0                5.1               3.5                1.4               0.2
1                4.9               3.0                1.4               0.2
2                4.7               3.2                1.3               0.2
```

波士顿房价

回归型数据集

```
X, y = ds.fetch_california_housing(return_X_y=True, as_frame=True)

samples_num = X.shape[0]  # 样本个数
features_num = X.shape[1]  # 特征个数

print('samples:', samples_num)
print('features:', features_num)

features_names = list(X.columns)  # 特征种类
house_min, house_max = y.min(), y.max() # 价格最小值和最大值

```

```
print('features_name: ', features_names)
print('house_min, house_max: ', house_min, house_max)

X_head = X.head(3)  # 前三行数据
print('X_head:\n', X_head)
```

```
samples: 20640
features: 8
features_name:  ['MedInc', 'HouseAge', 'AveRooms', 'AveBedrms', 'Population', 'AveOccup', 'Latitude', 'Long
house_min, house_max:  0.14999 5.00001
X_head:
    MedInc  HouseAge  AveRooms  AveBedrms  Population  AveOccup  Latitude  \
0  8.3252      41.0  6.984127   1.023810       322.0  2.555556     37.88
1  8.3014      21.0  6.238137   0.971880      2401.0  2.109842     37.86
2  7.2574      52.0  8.288136   1.073446       496.0  2.802260     37.85

   Longitude
0    -122.23
1    -122.22
2    -122.24
```

生成器

样本个数为 1000，特征个数为 3，随机加入 10 个异常值。

```
import numpy as np
from sklearn import linear_model

n_samples, n_features = 1000, 3  # 样本个数，特征个数
n_outliers = 10  # 异常值
noise = 0.5  # 高斯噪声标准偏差
bias = 2  # 偏差

X, y, coef = ds.make_regression(n_samples=n_samples,
                                n_features=n_features,
```

```
                                    n_informative=n_features,
                                    coef=True,
                                    bias=bias,
                                    noise=noise)

X[:n_outliers] = 3 + 0.5 * np.random.normal(size=(n_outliers, n_features))
y[:n_outliers] = -2 + 10 * np.random.normal(size=n_outliers)

n_samples, n_features = X.shape[0], X.shape[1]

print('n_samples:', n_samples)
print('n_features:', n_features)
print('coef:', coef)

lr = linear_model.LinearRegression().fit(X, y)

lr_coef = lr.coef_  # 拟合后的参数
lr_intercept = lr.intercept_  # 拟合后的截距
print('lr_coef:', lr_coef)
print('lr_intercept:', lr_intercept)
```

```
n_samples: 1000
n_features: 3
coef: [80.37989479 10.59685683 71.20514016]
lr_coef: [67.21466504 -2.7159688  59.12540121]
lr_intercept: -2.2554901024828964
```