preprocessing

Guangyao Zhao

2022-10-17

数据预处理在机器学习里是一个非常重要的步骤，指的是在清洗过后的数据的基础上，对数据消除量纲的步骤。常用的有标准化（Standardization）和归一化（Normalization）。因为后者易受异常值影响，所以一般在实际使用中使用前者。另外要注意，如果特征量纲一样，则不需要消除量纲的操作，因为这反而会损失部分信息。

Contents


```python
from sklearn import preprocessing
import numpy as np

np.random.seed(0)
X_train = np.random.randint(low=1, high=10, size=(3,5))
X_test = np.random.randint(low=1, high=10, size=(3,5))

print('X_train:\n', X_train)
print('X_test:\n', X_test)
```

```
X_train:
 [[6 1 4 4 8]
 [4 6 3 5 8]
 [7 9 9 2 7]]
X_test:
 [[8 8 9 2 6]
```

```
 [9 5 4 1 4]
 [6 1 3 4 9]]
```

StandardScaler

$$x = \frac{x - \mu}{\sigma} \tag{1}$$

根据 Eq. 1 进行标准化：

```python
scaler = preprocessing.StandardScaler().fit(X_train)

X_train_Satndard = scaler.transform(X_train) # 标准化训练集
X_test_Satndard = scaler.transform(X_test) # 标准化测试集
print('X_train_Satndard:\n', X_train_Satndard)
print('X_test_Satndard:\n', X_test_Satndard)
```

```
X_train_Satndard:
 [[ 0.26726124 -1.31319831 -0.50800051  0.26726124  0.70710678]
 [-1.33630621  0.20203051 -0.88900089  1.06904497  0.70710678]
 [ 1.06904497  1.1111678   1.3970014  -1.33630621 -1.41421356]]
X_test_Satndard:
 [[ 1.87082869  0.80812204  1.3970014  -1.33630621 -3.53553391]
 [ 2.67261242 -0.10101525 -0.50800051 -2.13808994 -7.77817459]
 [ 0.26726124 -1.31319831 -0.88900089  0.26726124  2.82842712]]
```

查看标准化后的平均值和标准差：

```python
X_mu = X_train_Satndard.mean(axis=0) # 平均值
X_sigma = X_train_Satndard.std(axis=0) # 标准差

print('X_mu:\n', X_mu)
print('X_sigma:\n', X_sigma)
```

```
X_mu:
 [-2.22044605e-16  7.40148683e-17  7.40148683e-17  1.48029737e-16
 -5.92118946e-16]
X_sigma:
 [1. 1. 1. 1. 1.]
```

MinMaxScaler

$$x = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{2}$$

根据 Eq. 2 将特征值缩小到 [0,1] 内

```
scaler = preprocessing.MinMaxScaler().fit(X_train)
X_train_MinMax = scaler.transform(X_train)
X_test_MinMax = scaler.transform(X_test)

print('X_train_MinMax:\n', X_train_MinMax)
print('X_test_MinMax:\n', X_test_MinMax)
```

```
X_train_MinMax:
 [[0.66666667 0.         0.16666667 0.66666667 1.        ]
 [0.         0.625      0.         1.         1.        ]
 [1.         1.         1.         0.         0.        ]]
X_test_MinMax:
 [[ 1.33333333  0.875       1.          0.         -1.        ]
 [ 1.66666667  0.5         0.16666667 -0.33333333 -3.        ]
 [ 0.66666667  0.          0.          0.66666667  2.        ]]
```

查看标准化后的最大值和最小值:

```
X_max = X_train_MinMax.max(axis=0) # 平均值
X_min = X_train_MinMax.min(axis=0) # 标准差

print('X_max:\n', X_max)
print('X_min:\n', X_min)
```

```
X_max:
 [1. 1. 1. 1. 1.]
X_min:
 [0. 0. 0. 0. 0.]
```