Scikit Learn

Tiền xử lý dữ liệu và mô hình hồi quy

HK1, Năm học 2022 - 2023

Chuẩn hóa dữ liệu

Chuẩn hóa dữ liệu là một bước quan trọng trong bài toán máy học.

Trung tâm hóa dữ liệu (Centering data): Đưa các điểm dữ liệu trong tập dữ liệu về xoay quanh giá trị 0 thay vì xoay quanh giá trị trung bình của tập dữ liệu

Co giãn dữ liệu (Scaling data): chuẩn hóa phạm vi của các đặc trưng dữ liệu

- Chuẩn hóa min-max (rescaling)

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

- Co giãn trung bình (mean normalization)

$$x' = \frac{x - \text{average}(x)}{\max(x) - \min(x)}$$

Chuẩn hóa dữ liệu

Co giãn dữ liệu (Scaling data): chuẩn hóa phạm vi của các đặc trưng dữ liệu

- Chính quy hóa (standardization): giúp giá trị của mỗi đặc trưng có trung bình bằng 0 và phương sai bằng 1

$$x' = \frac{x - \bar{x}}{\sigma}$$

- Vector đơn vị: biến đổi vector đặc trưng sau cho khi biến đổi có độ dài bằng 1

Chuẩn hóa dữ liệu

Mã hóa đặc trưng nhóm: thuộc tính của dữ liệu có thể lưu trữ dưới dạng chữ. *Ví dụ:* blue, green; small, medium, large... Có nhiều phương pháp mã hóa đặc trưng dạng nhóm, phổ biến:

- Mã hóa số (Numeric Encoding): gán mỗi giá trị của đặc trưng dạng nhóm thành số theo từng đôi một $\Leftrightarrow$ còn gọi là *mã hóa nhãn (label encoding)* hay *mã hóa số nguyên (integer encoding)*
- Mã hóa one-hot (One-hot encoding): biến mỗi giá trị của thuộc tính thành một biến nhị phân.

  Ví dụ: Thuộc tính màu sắc có 03 giá trị red, green, blue. Khi áp dụng mã hóa thì các giá trị lần lượt được biểu diễn bằng vector nhị phân là: 100, 010, 001

Chuẩn hóa min-max

```
from sklearn import datasets
from sklearn import preprocessing
from sklearn.model_selection import train_test_split

iris = datasets.load_iris()
X = iris.data
y = iris.target
X_train, X_test, y_train, y_test = train_test_split(X
    , y, test_size = 0.4, random_state = 123)

scaler = preprocessing.MinMaxScaler()
X_train_preprocess = scaler.fit_transform(X_train)
X_test_preprocess = scaler.transform(X_test)
```

Chính quy hóa

```
from sklearn import datasets
from sklearn import preprocessing
from sklearn.model_selection import train_test_split

iris = datasets.load_iris()
X = iris.data
y = iris.target
X_train, X_test, y_train, y_test = train_test_split(X
    , y, test_size = 0.4, random_state = 123)

scaler = preprocessing.StandardScaler().fit(X_train)
X_train_preprocess = scaler.transform(X_train)
X_test_preprocess = scaler.transform(X_test)
```

Mã hóa số

```
enc = preprocessing.OrdinalEncoder()
X = [['male', 'from US', 'uses Safari'], ['female', 'from Europe', 'uses Firefox']]
enc.fit(X)
enc.transform([['female', 'from US', 'uses Safari']])
```

Mã hóa one-hot

```
enc = preprocessing.OneHotEncoder()
X = [['male', 'from US', 'uses Safari'], ['female', 'from Europe', 'uses Firefox']]
enc.fit(X)
enc.transform([['female', 'from US', 'uses Safari'], ['male', 'from Europe', 'uses Safari']]).toarray()
```

Bài 1 - Chuan hoa cho IRIS

```
from sklearn import preprocessing
# import thu vien can thiet khac

# load du lieu iris

# chia du lieu iris theo ti le 8:2
X_train, X_test, y_train, y_test =

# thuc hien chuan hoa min-max
scaler = preprocessing.MixMaxScaler()
scaler.fit(X_train)
X_train1 = scaler.transform(X_train)
X_test1 = scaler.transform(X_test)

# thiet lap mo hinh va huan luyen mo hinh tren X_train1
model =
model.fit(X_train1, y_train)
# thuc hien du doan tren X_test1
y_pred1 = model1.predict(X_test1)
# danh gia tren y_pred1
score1 = accuracy_score(y_test, y_pred1)
```

Bài 2 - Chuan hoa cho WINE

- Sử dụng dữ liệu `wine`
- Chia tập dữ liệu theo tỷ lệ 8: 2
- Thực hiện chính quy hóa dữ liệu
- Cài đặt mô hình KNN và huấn luyện trên tập dữ liệu chưa chuẩn hóa
- Cài đặt mô hình KNN khác và huấn luyện trên tập dữ liệu đã chuẩn hóa
- Cài đặt mô hình SVM cho phân lớp và huấn luyện trên tập dữ liệu đã chuẩn hóa

```
1 from sklearn.svm import SVC
2 model = SVC()
```

Bài 3 - Dataset Haberman

```
import pandas as pd
# import package can thiet khac
# ten cua cac cot trong dataset
names = ['Age', 'Year operation', 'Axillary nodes detected', '
    Survival status']
# duong dan den file du lieu
url = "haberman.csv"
# load file csv
dataset = pd.read_csv(url, names=names)
# in mot so mo ta cho datasest
print(dataset.describe())
print(dataset.head(10))
print(dataset.shape)
# chia ra feature matrix X va target y
array  = dataset.values
X = array[:,:3]
y = array[:, 3]
# chia tap du lieu train, test 7:3

# Cai dat mo hinh SVM danh cho phan lop

# Danh gia viec huan luyen mo hinh
```

Bài 4 - Dataset Diabetes

```
from sklearn.linear_model import LinearRegression
from sklearn.neighbors import KNeighborsRegressor
from sklearn.metrics import mean_squared_error, r2_score
# import...
# load diabetes tu sklearn
diabetes =
# Hien thi so dong, so cot cua diabetes

# Tach X, y

# cho biet target thuoc mien gia tri vo han hay gioi han

# Chia du lieu theo ty le 8:2

# Thiet lap 02 mo hinh LinearRegression va KNeighborsRegressor

# Huan luyen

# Danh gia bang mean_squared_error va r2_score

# Danh gia cac mo hinh biet rang: MSE cang nho cang tot va R2
    cang gan 1 cang tot
```