

姓名：李宇豪

学号：21305412

周数：5

成绩：

程序：

```
import time
import numpy as np
from matplotlib import pyplot as plt
from sklearn.datasets import load_breast_cancer
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.pipeline import Pipeline
from sklearn.model_selection import ShuffleSplit
from sklearn.model_selection import learning_curve
from sklearn.preprocessing import PolynomialFeatures

def polynomial_model(d, c=0.05, **kwarg):
    polynomial_features = PolynomialFeatures(degree=d)
    logistic_regression = LogisticRegression(**kwarg, C=c,
                                              solver='liblinear',
max_iter=300)
    pipeline = Pipeline([("polynomial_features",
polynomial_features),
                        ("logistic_regression",
logistic_regression)])
    return pipeline

def plot_learning_curve(estimator, title, X, y, ylim=None,
                        cv=None, train_sizes=np.linspace(.1, 1.0, 5)):
    plt.title(title)
    if ylim is not None:
        plt.ylim(*ylim)
    plt.xlabel("Training example")
    plt.ylabel("Score")

    train_sizes, train_scores, test_scores = learning_curve(
        estimator, X, y, cv=cv, train_sizes=train_sizes)
    train_scores_mean = np.mean(train_scores, axis=1)
    test_scores_mean = np.mean(test_scores, axis=1)

    plt.grid(ls='--')
    plt.plot(train_sizes, train_scores_mean, 'o-', color="r",
            label="Training score")
```

```

plt.plot(train_sizes, test_scores_mean, 'o-', color="g",
         label="Cross-validation score")
plt.legend(loc="best")
return plt

cancer = load_breast_cancer()
X = cancer.data
y = cancer.target
X_train, X_test, y_train, y_test = \
    train_test_split(X, y, test_size=0.20, random_state=0)
time_all = time.process_time()
for i in [1, 2]:
    for j in ["11", "12"]:
        model = polynomial_model(i, penalty=j)
        start = time.process_time()
        model.fit(X_train, y_train)
        train_score = model.score(X_train, y_train)
        test_score = model.score(X_test, y_test)
        print(i, "order polynomial fitting with", j, "norm:")
        print("elapsed: {0:.4f}; train_score: {1:.4f}; test_score:
              {2:.6f}".format(time.process_time() - start,
                              train_score, test_score))

cv = ShuffleSplit(n_splits=10, test_size=0.2,
                 random_state=0)
title = 'Learning Curve (degree={0}, penalty={1})'
        .format(i, j)

plt.figure(figsize=(6, 4), dpi=200)
plot_learning_curve(polynomial_model(i, penalty=j),
                    title, X, y, cv=cv)

plt.show()
plt.clf()

if i == 2 and j == "11":
    R_train = []
    R_test = []
    for k in [0.01, 0.1, 1, 10, 100]:
        model = polynomial_model(i, k, penalty=j)
        start = time.process_time()
        model.fit(X_train, y_train)
        train_score = model.score(X_train, y_train)
        test_score = model.score(X_test, y_test)
        R_train.append(train_score)
        R_test.append(test_score)
        print("lambda={0}时, 训练集 R 方评分为{1}, 测试集 R 方评
              分为{2}".format(str(1/k), train_score,
                              test_score))

p1, = plt.plot([1, 2, 3, 4, 5], R_train, color="red")
p2, = plt.plot([1, 2, 3, 4, 5], R_test, color="blue")

```

```

plt.grid(ls='--')
legend = plt.legend([p1, p2], ["score of training
                                set", "score of test set"],
                                fontsize=12)
plt.xticks((1, 2, 3, 4, 5),
            ('100', '10', '1', '0.1', '0.01'))
plt.xlabel("lambda")
plt.ylabel("Score")
plt.show()
plt.clf()

print("程序运行总时间: ", time.process_time() - time_all)

```

输出:

```

Python 控制台 × week5 ×

Python 控制台
1 order polynomial fitting with l1 norm:
elapsed: 0.04688; train_score: 0.9319; test_score: 0.938596
1 order polynomial fitting with l2 norm:
elapsed: 0.00000; train_score: 0.9451; test_score: 0.938596
2 order polynomial fitting with l1 norm:
elapsed: 0.17188; train_score: 0.9912; test_score: 0.973684
    lambda=100.0时, 训练集R方评分为0.9802197802197802, 测试集R方评分为0.9473684210526315
    lambda=10.0时, 训练集R方评分为0.9912087912087912, 测试集R方评分为0.9736842105263158
    lambda=1.0时, 训练集R方评分为0.9934065934065934, 测试集R方评分为0.9736842105263158
    lambda=0.1时, 训练集R方评分为1.0, 测试集R方评分为0.9824561403508771
    lambda=0.01时, 训练集R方评分为1.0, 测试集R方评分为0.9736842105263158
2 order polynomial fitting with l2 norm:
elapsed: 0.18750; train_score: 0.9736; test_score: 0.964912
程序运行总时间: 37.71875

```

图:



