Lab11　sklearn 中的 SVM 应用实践

实验简介：sklearn 库的 SVM 分类器的实现。sklearn 官网：http://scikit-learn.org/stable/，包含 sklearn 资源，模块下载，文档、例程等。利用 scikit-learn 中自带的 iris 数据集，学习数据规范化、数据集切分、分类、预测，以及分类器 SVC 的测试评估等。

1、准备数据

从 sklearn.datasets 中导入 iris 数据集

```
from sklearn.datasets import load_iris
iris = load_iris().data
iris_target = load_iris().target
# 数据预处理
from sklearn.preprocessing import MinMaxScaler
MinMax = MinMaxScaler()
MinMax.fit(iris)
iris_transf = MinMax.transform(iris)

# 数据集分割

from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(iris_transf, iris_target, random_state=14)
```

2、构造 SVM 分类器

利用 sklearn 中的 SVC 估计器，用默认参数训练

```
from sklearn.svm import SVC
svc = SVC()
svc.fit(X_train, y_train)
```

运行结果：（1）

3、模型评估

```
print("test_svc.score: {:.3f}".format(svc.score(X_test, y_test)))
print("train_svc.score: {:.3f}".format(svc.score(X_train, y_train)))
```

运行结果：（2）

4、模型预测

```
predict_labels = svc.predict([[0.13888889, 0.58333333, 0.15254237, 0.04166667]])
print(predict_labels)
```

运行结果：（3）

5、利用 Grid_searchCV 寻找最佳参数

```
from sklearn.model_selection import GridSearchCV
from sklearn.svm import SVC
svc = SVC()
# 候选参数
tuned_parameters = [{'kernel': ['rbf'], 'gamma': [1e-3, 1e-4],
                     'C': [1, 10, 100, 1000]},
                    {'kernel': ['linear'], 'C': [1, 10, 100, 1000]}]
clf = GridSearchCV(svc, tuned_parameters)
clf.fit(X_train, y_train)

# 输出最优参数
print("Best parameters: ",clf.best_params_)
```

运行结果：（4）

WangBianqin, Public Laboratory & Teaching Center, Guangzhou East Campus, Sun Yat-sen University

6、利用获得最优参数预测

```
from sklearn.svm import SVC
svc = SVC(C=1, kernel='linear')
svc.fit(X_train, y_train)
print("test_svc.score: {:.3f}".format(svc.score(X_test, y_test)))
print("train_svc.score: {:.3f}".format(svc.score(X_train, y_train)))
```

运行结果：（5）

7、利用 matplotlib 可视化不同参数模型的准确率变化

```
from sklearn.svm import SVC
Cs = []
score = []
for c in range(10, 100, 10):
    svc = SVC(C=c, gamma=0.001, kernel= 'rbf')
    svc.fit(X_train, y_train)
    accuracy = svc.score(X_train, y_train)
    print("the accuracy of svc model with C = {0} is {1}".format(c, accuracy))
    Cs.append(c)
    score.append(accuracy)

import matplotlib.pyplot as plt
%matplotlib inline

plt.title("SVC model: argument: C")
plt.xlabel("C")
plt.ylabel("Accuracy")
plt.plot(Cs, score, 'b')
```

运行结果：（6）

8、作业习题

（1）参照 PPT，可视化基于 iris 数据集的 SVM 分类器的决策边界，并操作说明 decision_function（）方法得的功能和用法，以及 matplotlib.pyplot.contour()的功能和用法。

（2）按照上述步骤，对手写体数字集识别，并利用 matplotlib 可视化测试集中前 10 个数字的灰度图像。

注：sklearn.datasets 中包含 digits，可用如下方式导入：

```
from sklearn.datasets import load_digits
digits = datasets.load_digits()
```

深入了解该数据集，请参考官网 https://scikit-learn.org/dev/index.html

（3）利用 SVR 预测患有疝病的马的死亡率，并分别利用网格搜索和随机搜索寻找最佳组合参数。（数据集见 Lab9）

WangBianqin, Public Laboratory & Teaching Center, Guangzhou East Campus, Sun Yat-sen University