

【华子】机器学习017-使用GridSearch搜索最佳参数组合 -

(本文所使用的Python库和版本号: Python 3.5, Numpy 1.14, scikit-learn 0.19, matplotlib 2.2)

有时候，一个非常优秀的模型，可能A参数取最优值时，B参数并不一定是最优值，从而使得验证曲线的方式有其自身的弊端。

此处介绍的使用GridSearch来搜索最佳参数组合的方法，可以避免上述弊端，GridSearch可以同时优化多个不同参数的取值。

1. 准备数据集

数据集的准备工作和文章（[\[机器学习014-用SVM构建非线性分类模型\]](#)）中一模一样，不再赘述。

2. 使用GridSearch函数来寻找最优参数

使用GridSearch函数来寻找最优参数，需要首先定义要搜索的参数候选值，然后定义模型的评价指标，以此来评价模型的优劣。GridSearch会自动计算各种参数候选值，从而得到最佳的参数组合，使得评价指标最大化。

```
from sklearn import svm, cross_validation
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import classification_report
# 需要优化的参数及其候选值
parameter_grid = [ {'kernel': ['linear'], 'C': [1, 10, 50, 600]}, {'kernel': ['poly'], 'degree': [2, 3]},
                    {'kernel': ['rbf'], 'gamma': [0.01, 0.001], 'C': [1, 10, 50, 600]},
                    ]
metrics = ['precision', 'recall_weighted'] # 评价指标好坏的标准

for metric in metrics:
    print("Searching optimal hyperparameters for: {}".format(metric))

    classifier = GridSearchCV(svm.SVC(C=1),
                              parameter_grid, cv=5, scoring=metric)
    classifier.fit(train_X, train_y)
    print(classifier.cv_results_) #获取各项参数
    print(classifier.best_params_)#打印最佳的参数组合
    y_pred = classifier.predict(test_X) # 此处自动调用最佳参数
    print(classification_report(test_y, y_pred)) #打印模型报告
```

-----输出-----

Searching optimal hyperparameters for: precision Scores across the parameter grid: {'C': 1, 'kernel': 'linear'}: avg_scores: 0.809

Full performance report: precision recall f1-score support

0 0.75 1.00 0.86 36 1 1.00 0.69 0.82 39

avg / total 0.88 0.84 0.84 75

Searching optimal hyperparameters for: recall_weighted

Scores across the parameter grid: {'C': 1, 'kernel': 'linear'}: avg_scores: 0.653

Highest scoring parameter set: {'C': 600, 'gamma': 0.01, 'kernel': 'rbf'}

Full performance report: precision recall f1-score support

0 1.00 0.92 0.96 36 1 0.93 1.00 0.96 39

avg / total 0.96 0.96 0.96 75

-----完-----

#####小*****结#####

1. 使用GridSearch中的GridSearchCV可以实现最佳参数组合的搜索，但需要指定候选参数和模型的评价指标。
2. 使用classifier.best_params_函数可以直接把最佳的参数组合打印出来，方便以后参数的直接调用
3. classifier.predict函数是自动调用最佳的参数组合来预测，从而得到该模型在测试集或训练集上的预测值。

#####

如果要使用最佳参数来构建SVM模型，可以采用下面的代码来实现：

```
best_classifier=svm.SVC(C=600,gamma=0.01,kernel='rbf') # 模型报告使用的是最佳参数组合
best_classifier.fit(train_X, train_y)
y_pred =best_classifier.predict(test_X)
print("模型报告: {}".format(classification_report(test_y,y_pred)))
```

得到的结果和上面上一模一样。

参考资料:

- 1, Python机器学习经典实例, Prateek Joshi著, 陶俊杰, 陈小莉译