Sklearn svm解决multi-class、multi-label分类问题

2019年1月14日 16:30

本文将介绍使用sklearn实现基本的multi-class、multi-label分类问题，包含内容如下：

1.multi-class与multi-label的区别

2.SVM二分类示例

3.SVM+multi-class

4.SVM+multi-label

5.multi-class+multi-label

1.multi-class与multi-label的区别

Multi-class多类别问题，指的是待分类的数据只有一个标签（label），但这个标签取值可以是多个，标签的全部取值是个有限集。例如，视频的点击量，视频特征作为输入数据，点击量作为唯一的标签，它的取值可以是100、200、...。

Multi-label多标签问题，指的是待分类的数据有多个标签，但每个标签取值都是二值的，即0/1。比如视频可以分为喜剧、爱情、恐怖等，每一个标签下，若值为1，表示属于该类，0表示不属于该类，一个视频可以同时属于多类。

实际上，需要分类的任务不止这么简单，可能是multi-class和multi-label的结合。即数据具有多个标签，同时每个标签可以取多个值。针对每种任务的分类方法，会在下文一一介绍。

2.SVM二分类示例

本文以SVM模型处理分类问题为示例。设想最简单的二分类问题：输入数据特征矩阵，预测每条数据属于0类/1类。

Toy example:

```
From sklearn.svm import SVC
From sklearn.metrics import precision_score
#模型定义
model=SVC(kernel='linear')
#数据准备
X=[[0.1,0.2,0.3],
[0.1,0.2,0.2],
[0.1,0.3,0.3]
]
labels=[0,1,1]
#训练
model.fit(X,labels)
```

```
#使用训练好的模型预测标签
pre_labels=model.predict(X)
#输出precision
print(precision_score(pre_labels,labels))
```

实际任务中，输入数据量非常庞大，当特征矩阵和标签矩阵非常为稀疏矩阵时，可以使用压缩矩阵存储以节省空间，如csr_matrix类型。model.fit()的输入可以接受压缩矩阵。

3.SVM+multi-class

在处理multi-class任务时，有两种方式学习分类器，OneVsRestClassifier(OVR)和OneVsOneClassifier（OVO），具体解释参考sklearn文档。使用方式如下：

```
fromsklearn.multiclassimportOneVsRestClassifier
model=OneVsRestClassifier(SVC(kernel='linear'))
```

multi-class 的labels形如labels=[1, 2, 3]可以直接输入到模型中。

4.SVM+multi-label

multi-label的标签形如labels=[[0 1 1],[1 0 0],[1 1 1]]可以直接输入到模型中，需注意此时labels应为array类型， 模型同样需套上OneVsRestClassifier（）或者OneVsOneClassifier（）。

5.multi-class+multi-label

此时raw labels 形如labels=[[2], [3, 5], [3, 4]]，标签的长度不一。此时首先将labels转化为二值array，转换后的labels=[[0, 0, 1, 0, 0, 0], [0, 0, 0, 1, 0, 1], [0, 0, 0, 1，1, 0]]，即为raw labels的one-hot编码， 可以作为模型的输入。

当raw labels为大的稀疏矩阵时，可以将其转换为csr_matrix，np.array和csr_matrix互相转换的方式如下：

```
fromnumpyimportarray
fromscipy.sparseimportcsr_matrix
A=array([[0,0,1,0,0,0],[0,0,0,1,0,1],[0,0,0,1,1,0]])
A_csr=csr_matrix(A)
A_arr=A_csr.todense()
```