

**数据挖掘**

**互评作业四**

**题　　目： 离群点分析与异常检测**

**学 院： 计算机学院**

**专业名称： 计算机科学与技术**

**学 号： 3120191033**

**姓 名： 彭成**

**任课教师： 汤世平老师**

2020年6月22日星期一

目录

[Video Game Sales 电子游戏销售分析 3](#_Toc42499603)

[数据集介绍、探索方向及提交内容 3](#_Toc42499604)

[数据集预处理 3](#_Toc42499605)

[电子游戏市场分析 5](#_Toc42499606)

[受欢迎的平台 5](#_Toc42499607)

[受欢迎的游戏 5](#_Toc42499608)

[受欢迎的游戏类型 6](#_Toc42499609)

[受欢迎的游戏发行商 7](#_Toc42499610)

[预测每年的电子游戏销售额 8](#_Toc42499611)

[每年的销售金额分析 8](#_Toc42499612)

[通过北美洲的销售额来预测全球的销售额 9](#_Toc42499613)

[通过年份来预测某几年的销售额 10](#_Toc42499614)

[可视化应用 11](#_Toc42499615)

# 问题描述

本次作业将从Anomaly Detection Meta-Analysis Benchmarks提供的benchmark数据集中任选两个进行分析。本次实验选取的是wine\_benchmarks数据集和abalone\_benchmarks数据集。

## wine\_benchmarks数据集

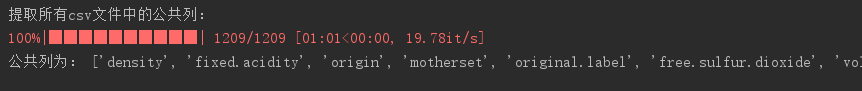
### 数据集预处理

#### 找出公共列

首先，由于有数千个csv文件，且每个csv文件有许多不同的列，我没定义一个方法来提取所有文件中相同的列的名称。



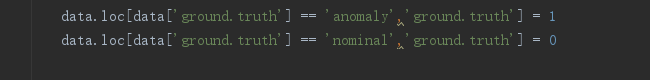
提取的结果是：



公共列为： ['density', 'fixed.acidity', 'origin', 'motherset', 'original.label', 'free.sulfur.dioxide', 'volatile.acidity', 'chlorides', 'total.sulfur.dioxide', 'pH', 'alcohol', 'sulphates', 'citric.acid', 'ground.truth', 'diff.score', 'residual.sugar']

其中[‘motherset’, ‘origin’]不是数值属性，我们不对其进行考虑。

而[‘ground.truth’]，将其转化为0,1标签。

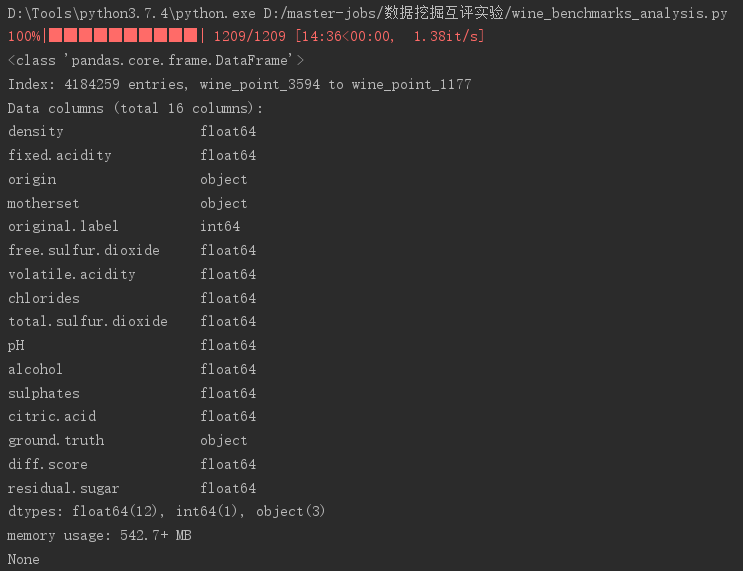


#### 数据读取

为了节约时间，在上一步找出公共列之后，直接将结果写入list中，将那部分代码注释。

将数据集按照7:3的比例划分为训练集与测试集，我们将在训练集上对各种检测离群点的方法进行训练，再在测试集上查看各种方法的效果，并进行对比。

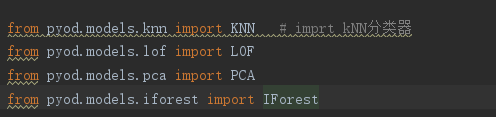
可以看到数据的形式如下：



### 模型选择与异常检测

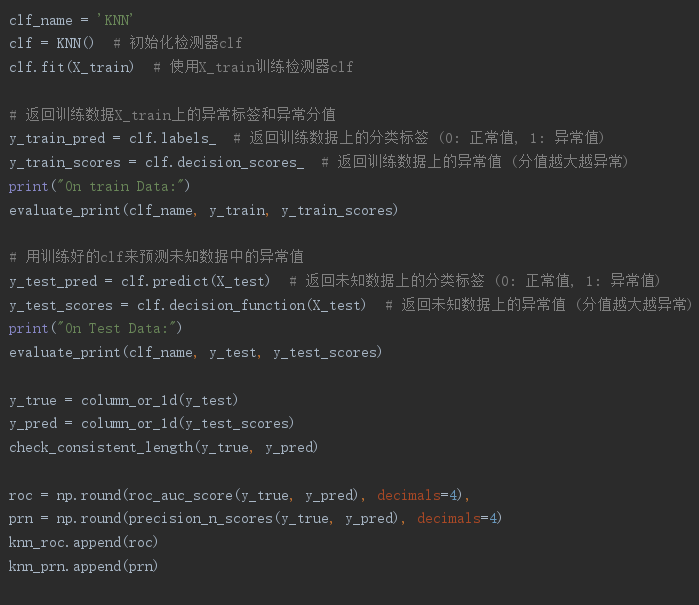
#### 模型选择

本次实验选取KNN、LOF、PCA、IForest四种算法来检测异常，并计算每种算法在各个csv文件中的ROC值和PRN值。

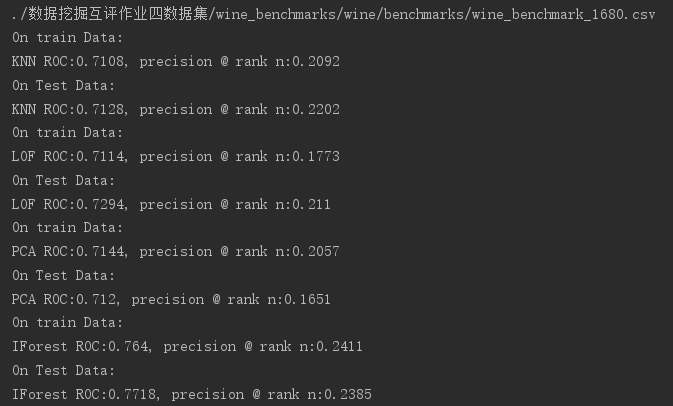


#### 计算

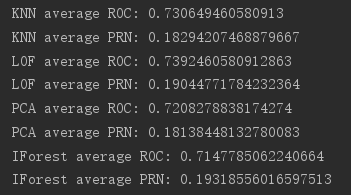
对每个csv文件，分别将其换分为训练集和测试集，在训练集和测试集上分别计算ROC和PRN值。代码为：



每个csv文件对每种算法输出对应的ROC值和PRN值。



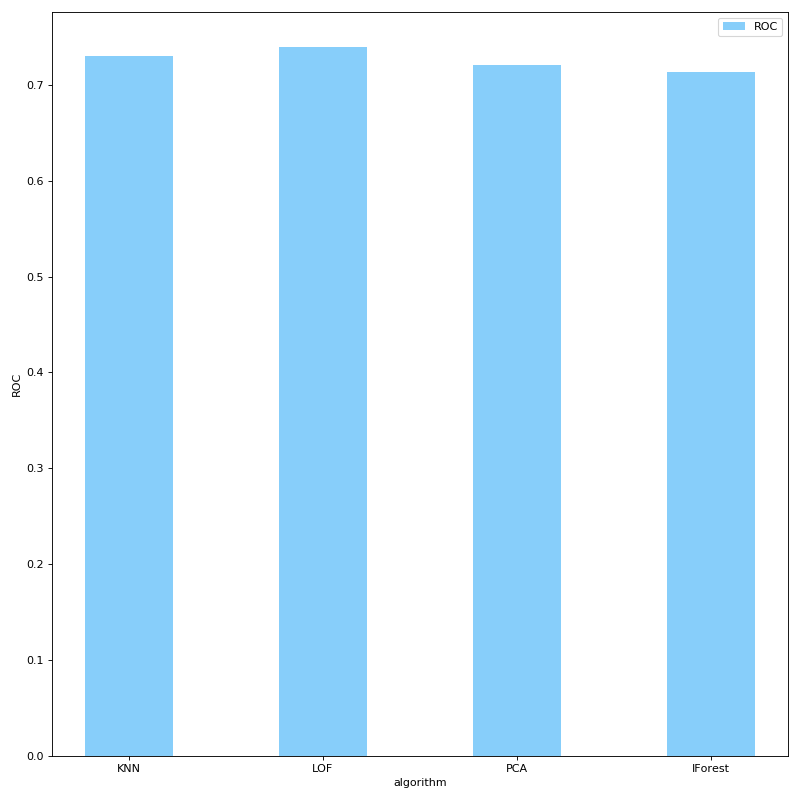
在运行完所有cssv文件之后，计算其平均值，得到以下结果。

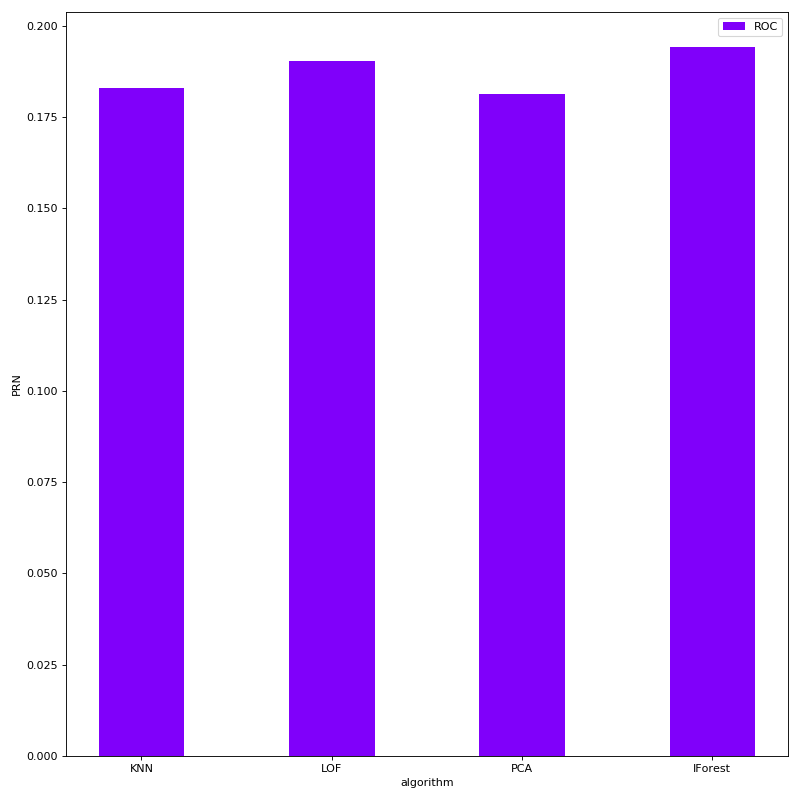


### 算法运行结果比较与结果分析

我们可以看到，通过比较ROC可以发现，在数据集wine上，LOF的效果比较好，而通过PRN的比较可以发现，在数据集wine上，IForest的效果比较好。

对比图如下：





代码如下：



## abalone\_benchmarks数据集

### 数据集预处理

#### 找出公共列

首先，由于有数千个csv文件，且每个csv文件有许多不同的列，我没定义一个方法来提取所有文件中相同的列的名称。

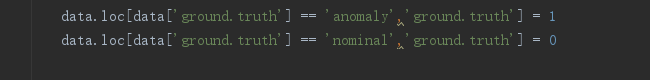


提取的结果是：

公共列为： ['ground.truth', 'V2', 'diff.score', 'original.label', 'motherset', 'V1', 'origin', 'V4', 'V3', 'V7', 'V5', 'V6']

其中[‘motherset’, ‘origin’]不是数值属性，我们不对其进行考虑。

而[‘ground.truth’]，将其转化为0,1标签。



#### 数据读取

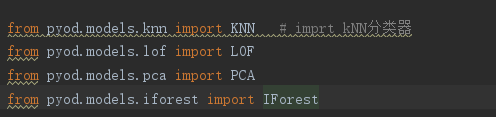
为了节约时间，在上一步找出公共列之后，直接将结果写入list中，将那部分代码注释。

将数据集按照7:3的比例划分为训练集与测试集，我们将在训练集上对各种检测离群点的方法进行训练，再在测试集上查看各种方法的效果，并进行对比。

### 模型选择与异常检测

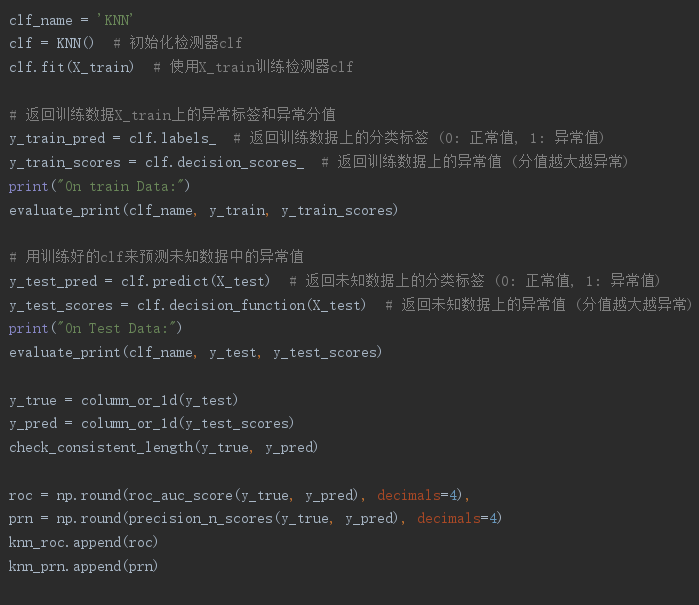
#### 模型选择

本次实验也是选取的KNN、LOF、PCA、IForest四种算法来检测异常，并计算每种算法在各个csv文件中的ROC值和PRN值。

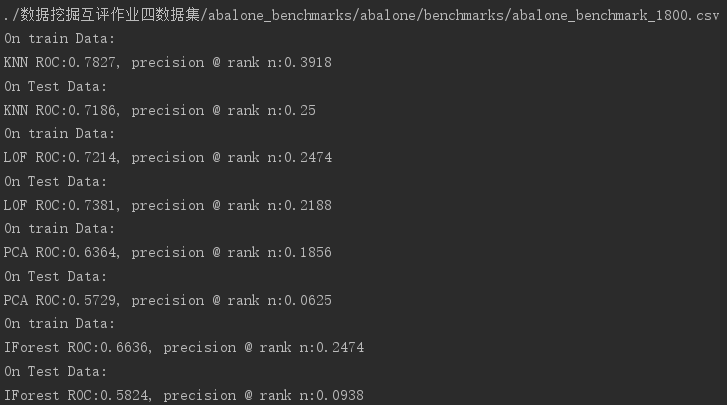


#### 计算

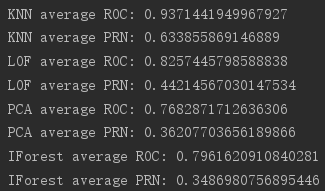
对每个csv文件，分别将其换分为训练集和测试集，在训练集和测试集上分别计算ROC和PRN值。代码为：



每个csv文件对每种算法输出对应的ROC值和PRN值。



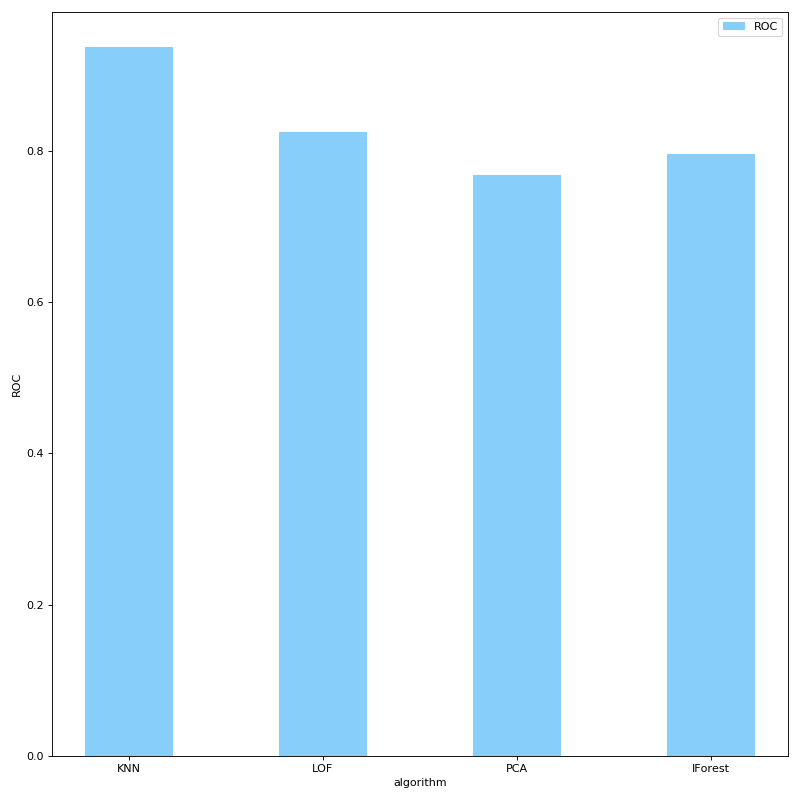
在运行完所有cssv文件之后，计算其平均值，得到以下结果。

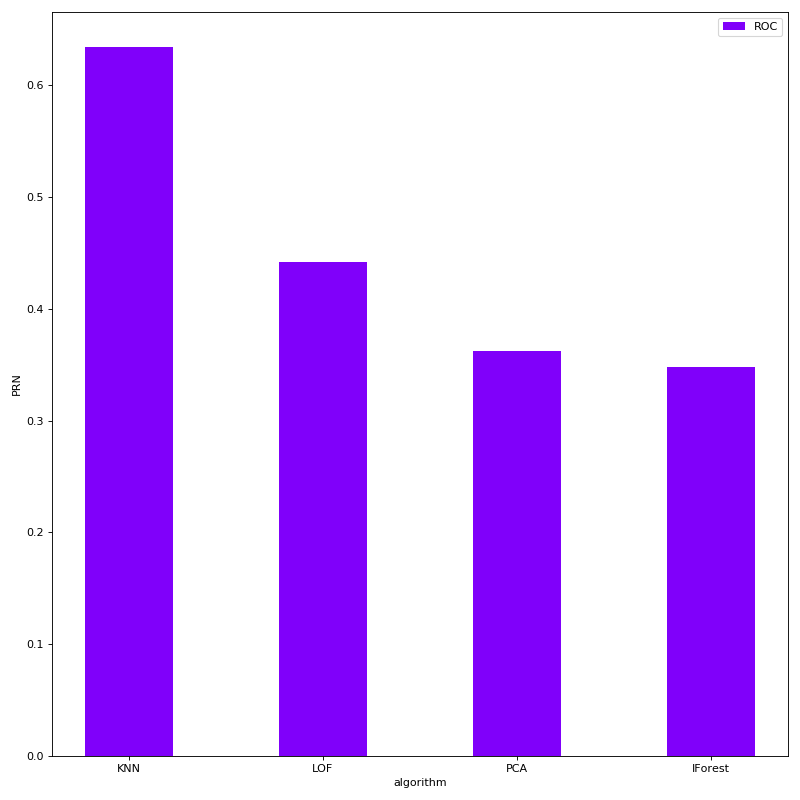


### 算法运行结果比较与结果分析

我们可以看到，通过比较ROC可以发现，在数据集abalone上，KNN的效果比较好，而通过PRN的比较可以发现，在数据集abalone上，还是KNN的效果比较好。

对比图如下：





代码如下：

