**杭州电子科技大学计算机学院**

**数据仓库与数据挖掘**

**实验3：分类和预测**

时间：2019年11月11日，学号：17061833 姓名：於文卓

**注意：**

**1）实验开始前，填写学号和姓名。**

**2）将文件名由“分类和预测”更改为“学号+姓名+分类和预测”。**

**3）作业做完后，验收之后通过作业提交系统提交。提交网址是：**

https://www.wjx.cn/jq/46478225.aspx

# 一、实验目的

1、理解分类的一般过程和基本原理；

2、巩固分类算法的算法思想，能够进行分类操作；

3、学会分类预测问题中的性能评估方法。

# 二、实验原理

1、常用的预测模型：决策树、朴素贝叶斯分类器和支持向量机（SVM）等。

2、评价预测结果，常用的性能评价指标：

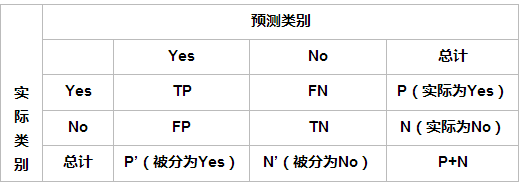
混淆矩阵

True Positive(真正，TP)：将正类预测为正类数

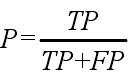
True Negative(真负，TN)：将负类预测为负类数

False Positive(假正，FP)：将负类预测为正类数(误报)

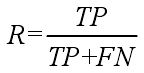
False Negative(假负，FN)：将正类预测为负类数 (漏报)



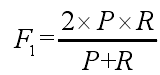
（1）**精确率**P(Precision)是指被正确预测为正例数与所有被预测为正例数的比率，反映了预测模型的准确程度，也称为查准率。



（2）**召回率** R(Recall)是指被正确预测为正例数与实际正例数的比率，反映了一个有缺陷模块被正确预测出的概率，也称为查全率。



（3）**F-Measure** 是信息检索领域的一个评价指标，常用的是F1 度量，即为精确率与召回率的调和平均数。

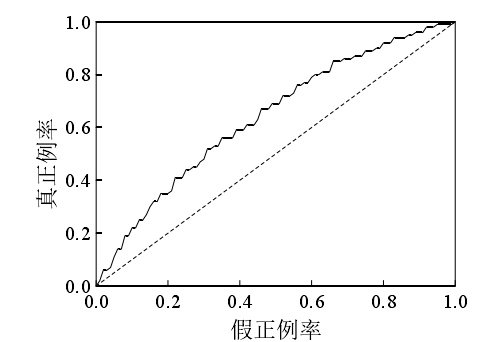


（4）**AUC**

真正例率 TPR(True Positive Rate)与召回率相同，也是指被正确预测为正例数与实际正例数的比率。

假正例率 FPR(False Positive Rate)是指被错误预测为正例数与实际负例数的比率。

接受者操作特征(Receiver Operating Characteristic，ROC)曲线是描述分类模型真正例率 TPR 和假正例率 FPR 之间关系的一种图形化方法，如下图所示。



横坐标表示假正例率，纵坐标表示真正例率。对于一个特定的预测模型和训练数据集，其预测结果对应于 ROC 曲线上的一个点，通过调整该模型的阈值即可得到一条经过(0, 0)和(1, 1)的曲线，曲线下方的面积即为AUC(Area Under the Curve)的值。其中，AUC的取值范围为0~1，当 AUC 为 0.5 时，表示随机猜测模型的性能，如上图中的虚线所示。AUC 值越大，说明该模型的性能越好。因此，好的预测模型应尽可能地靠近坐标系的左上角。

# 三、实验内容：

选取12个 NASA 数据集作为实验对象，采用不同的分类方法分别对这12个数据集进行软件缺陷预测分析。（数据见data文件夹）

## 1. 特征

(1) 原始特征集；

(2) **选择前 特征作为特征子集**：采用上机1 中数据预处理方法，获得特征排序列表。在特征排序列表的基础上，按照顺序依次进行特征子集选择。选择前个特征，d表示原始特征数；

**(3) 选出与类别相关度较高的特征子集**：采用上机1 中数据预处理方法，获得特征排序列表。在特征排序列表的基础上，按照顺序依次进行特征子集选择。例如，得到的特征排序列表为，d表示特征个数，排序越靠前，说明该特征与类别的相关程度越高。由此，根据特征排序顺序可以得到d个特征子集，分别为，最后评价所有特征子集的分类性能，并据此选取与类别相关度较高的特征子集，并进一步构建分类预测模型。 （**此特征子集选做**）

## 2. 预测模型方法

分别根据(1.特征) 部分的三种特征选取方法得到相应的特征，对于特征分别构建分类器预测模型，可以在决策树、Naive Bayes (NB) 、SVM分类器中**选择2个**或者**3个**分类器构建预测模型。

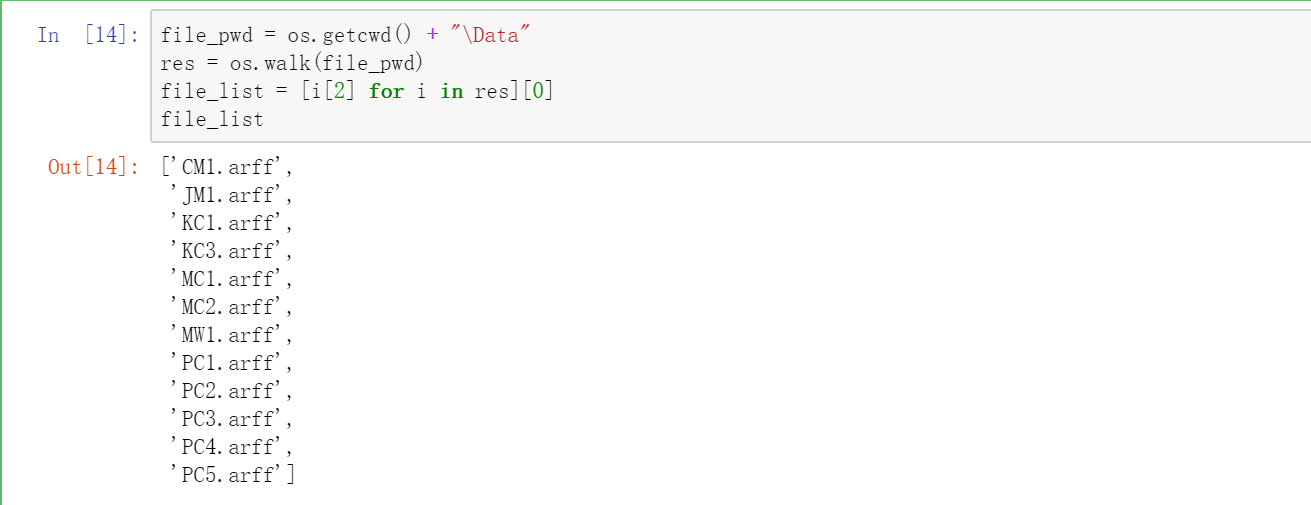
## 3. 性能评价

选取**F-Measure**或**AUC**作为性能评价指标；

采用 10 次 10 折交叉验证，保证实验结果的准确性和可靠性。

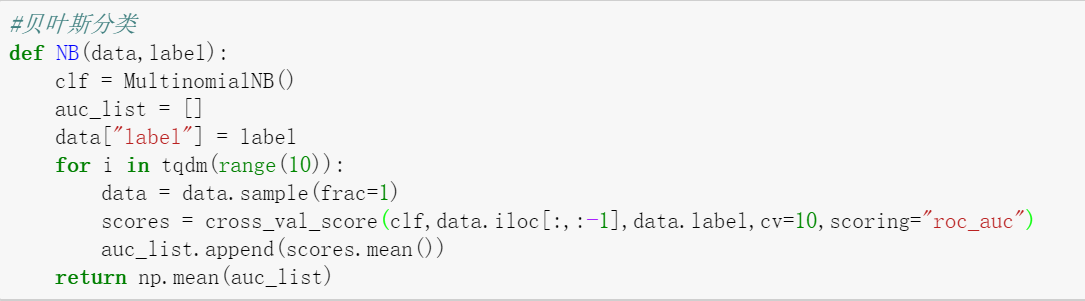
# 四、实验步骤

## 1. 数据读入



## 2. 定义不同的分类器

定义贝叶斯、决策树和SVM分类器，并使用十次试折交叉验证的方法进行模型评估。评估指标使用AUC







## 3. 使用数据集的全部特征进行训练



## 4. 使用数据集的Log2D的特征进行训练

首先使用实验一的算法，基于相似性度量将数据集特征排序，排序后取前Log2D个特征进行训练，其中D为数据集特征总数。代码中的prepross即为预处理函数。



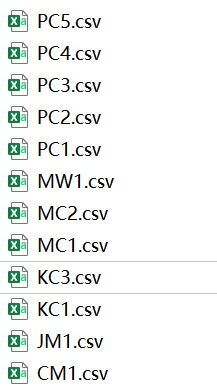
## 5. 选出与类别相关度较高的特征子集

对每一个样本，将特征排序。将每一个特征子集传入模型训练，从而找出最优的特征子集

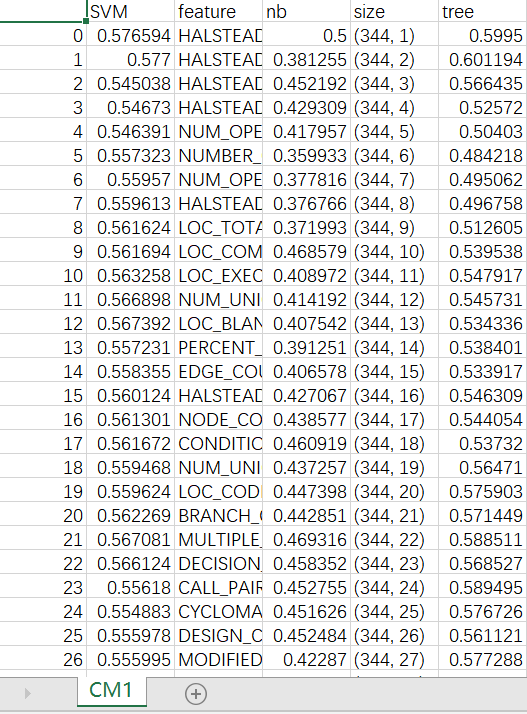


# 五、实验结果

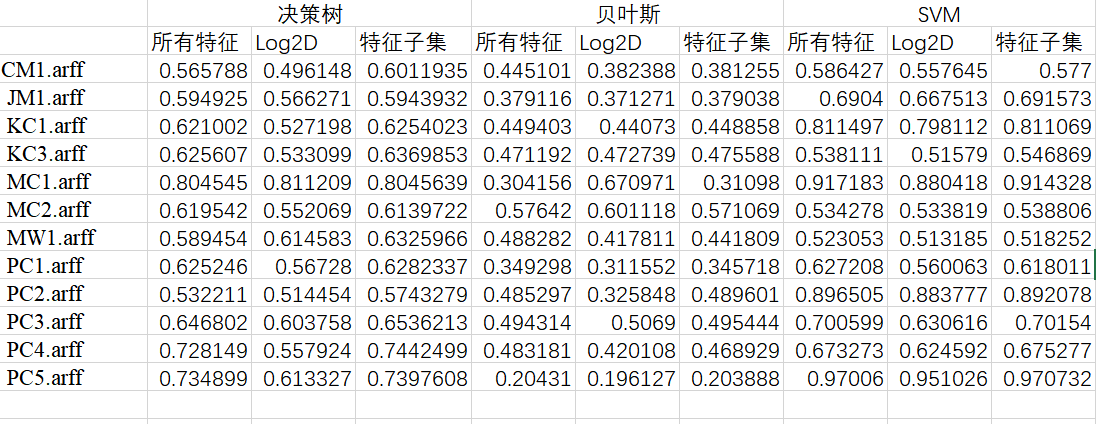
在选出最优特征子集时，得到了12个文件



每一个文件的内容如下，是不同特征子集在不同分类器下的AUC值，取SVM和Tree的平均值，最大值对应的样本作为最优子集。



最终结果如下



可以看到最优特征子集的结果和所有特征的结果很近似。从结果来看，最优子集所得到的的结果和所有数据拿去训练得到的结果类似，似乎说明将特征排序后取Log2D组特征在这些样本中并不是一个最好的办法。

有一些特征子集的结果小于所有特征的结果，这是因为每次交叉验证的数据都是随机的，是由于训练数据不同而产生的误差。

# 六、心得体会

1.最优子集实在是太难跑了，自己的电脑跑的太慢，通过搜索找到了一个不错的平台提供免费CPU和GPU，最终用时5小时左右。



2. 数据预处理没有唯一的答案，就像这些样本，使用特征排序+Log2D的效果不一定好于所有特征一起训练的效果。所以在自己做实验时，需要多尝试。