不平衡数据处理方法

不平衡数据集是数据集中的特殊情况,针对不平衡数据集分类问题,目前不平衡分类方法可以划分为三类：

1. 数据层：在训练基分类器前，采取抽样技术重构数据集，降低不平衡度，包括过采样和欠采样。欠取样随机删除大类实例来平衡类分布，过采样通过随机增加少数类实例的方法来降低不平衡度。Chawla等人提出的SMOTE(synthetic minority over-sampling technique)方法，Huaxiang Zhang提出的RWO 抽样方法，都是通过采用合成的新实例来增加少数类实例数量。
2. 算法层：根据不平衡数据特点构造新算法或改造现有算法。比如代价敏感学习(Cost-sensitive learning)，在评价中引入代价，把少数类误判的代价要高于多数类误判，此时分类目标不是为了降低错误率，而是为了降低错误分类而带来的代价。Britta Anker Bak在文献中提出一种高维数据不平衡分类器。
3. 混合方法：将数据层方法和算法结合起来，如将数据抽样技巧与集成学习方法结合在一块。在基分类器训练前，先对训练数据抽样，降低其不平衡度，或者构建不同平衡度下的基分类器，通过投票提高最终预测效果。Díez-Pastor J F提出的Random balance & AdaBoost方法；Kuncheva L I提到的RUSBoost、SMOTEBoost等方法都是基于混合方法角度。李雄飞等提出的不平衡数据分类算法PC-Boost，算法以信息增益率为分裂准则构建决策树，作为弱分类器，在每次迭代初始，利用数据合成方法添加合成的少数类样例，平衡训练信息；在子分类器形成后，修正“扰动”，删除未被正确分类的合成样例。

在不平衡数据中，使用传统分类算法的准确率或错误率作为分类器性能评价指标可能会出现偏差。目前不平衡数据分类算法的评价方法包括：AC、TPR、Precision、 Recall 、F-measure、Gmean、ROC曲线等指标。

随机采样

***记S***为训练集;，***Smin***是少数类样本, ***Smaj***为多数类样本集合,***E***为随机选择的样本。

在随机过采样中，E是从少数类验本中直接采样的一个集合，此时：

虽然随着E的增加，使得少数类样本个数增加，也即在某种程度上改变了数据的分布，似的分类器更容易学习，但是由于E中的样本直接来源于部分少数类样本的直接拷贝，这样容易使得分类器过拟合。

随机欠采样，从原始数据的多数类样本中直接选择一部分样本，并从训练集中移除，此时：

显然，直接移除一部分的多数类样本会使得原始数据出现信息丢失。

Informed Undersample

EsayEnsemble

EsayEnsemble算法每次从多数类样本中抽取一个子集，然后与少数类合并构成一个新的训练集，然后训练一个学习器。这样通过多以独立抽样，就能训练多个分类器，最后训练得到的多个学习器进行集成即可得到最终的结果。

算法描述：

输入：少数类样本集P,多数类样本集N，从N中进行抽样的子集大小T,训练Adaboost ensemble的迭代次数

1. ;
2. repeat
3. i=i+1
4. 随机从N中选择样本子集使得;
5. 利用和P训练,是Adaboost ensemble,共有个弱分类器，相应的权重为, 阈值为,也即：
6. Until ;
7. 输出：

BalanceCascade

BalanceCascade算法通过迭代的构造平衡数据集，并切移除多数类中的冗余样本来学习分类器。具体的，当被训练完成之后，如果样本被正确分类,就可以推断在给定的条件下，可能是N中的冗余样本，因此也就可以从N中移除一些被正确分类的样本。

算法描述

输入：少数类样本集P,多数类样本集N，从N中进行抽样的子集大小T训练Adaboost ensemble的迭代次数

1. ，,f是需要满足的false positive rate。
2. repeat
3. i=i+1
4. 随机从N中选择样本子集使得;
5. 利用和P训练,是Adaboost ensemble,共有个弱分类器，相应的权重为, 阈值为,也即：
6. 调整，使得的false positive rate是f.
7. 从N中移除所有被正确分类的样本
8. Until ;
9. 输出：

OOS(One Side Selection)

OOS算法的基本思想也是从多数类中选择一个具有代表性的子集，然后将该子集和少数类集合合并构成一个新的训练集，进而对新的训练集施加数据清洗的一些操，从而得到最终的训练集。

OOS算法描述：

1. 记S为原始训练集；
2. 初始化C，是的C包含所有的少数类样本以及一部分随机选择的多数类样本；
3. 基于C中的样本，通过1-NN准则对S中的样本进行分类，将S中未正确分类的样本移入C中；
4. 从C中移除所有满足Tomek links的多数类样本，保留少数类样本，从而得到新的数据集T，其中移除的多数类样本被认为是边界点或者噪声。

(注：对于两个不同类别的样本x,y，记表示二者之间的距离。如果不存在样本z使得或者,则称数据对(x,y)为Tomek links.)

**Synthetic Sampling with Data Generation**

SMOTE算法

SMOTE算法的主要思想就是利用特征空间的相似性生成新的少数类合成样本。针对少数类样本，首先计算其k个近邻，然后随机选择一个近邻和原始的样本生成新的样本。



算法描述

输入：少数类样本个数T；上采样倍率N%；近邻个数k;

（\*如果N<100%,就从小类样本随机选择一定比例的样本进行上采样\*）

;

从少数类样本中随机选择T个样本；

N=100;

EndIf

N=(int)(N/100);

K=最近领个数；

Numattrs=数据集中属性个数；

Sample[][]:原始少数类样本组；

newindex:记录生成的样本个数，初始值为0；

Systhetic[][]:合成样本数组；

For i=1 to T

计算少数类样本i的k个近邻，并将索引保存在nnarray中；

Populate(N,i,nnarray)

EndFor

Populate(N,i,nnarray)(用于生成合成样本)

While N！=0

生成1-k之间的一个随机数，记为nn,随机选择呢样本i的k个近邻中的一个；

For attr=1 to numattrs

计算：dif=Sample[nnarray[nn]][attr]-Sample[i][attr];

生成一个0-1之间的随机数gap;

Systhetic[newindex][attr]=Sample[i][attr]+gap\*dif;

EndFor

newindex++;

N=N-1

EndWhile

输出：(N/100)\*T个合成的小类样本

Border-line-SMOTE

为了获得更好的预测精度，大多数分类器试图在训练过程中，尽可能学习准确的分类边界。于是在边界上或者边界附近的样本更容易被错误分类，也即对分类器更加重要。Border-line-SMOTE首先找出边界少数类样本，然后根据找到的样本点由SMOTE方法生成新的合成样本。



算法描述

1. 针对每一个确定其m个最近邻，记为;
2. 确定近邻中多数类验本的个数;
3. 选择满足吧：
4. 根据得到的，采用SMOTE方法进行样本生成，

ADASYN

ADASYN的主要思想是通过一个密度函数来决定每个少数类样本需要生成的样本个数。通过密度函数来强制学习方法重点关注难以学习的样本。

1. 计算生成样本的个数，其中是一个参数，用来指定期望的平衡水平。
2. 对于每一个，找到其K个最近邻,计算作为分布函数，表示样本的近邻中属于多数类的个数，Z为归一化因子。
3. 确定需要生成的样本个数;
4. 采用SMOTE方法生成:

**Sampling with Data Cleaning**

Tomek links

1. 给定样本对(),其中，记表示二者之间的距离.
2. 如果不存在样本,使得或者,则称数据对为Tomek links.)

Tomek links 常用其他方法集合一起使用，例如OOS，SMOTE+Tomek.



Random Balance

该算法的基本思想是随机选择多数类和少数类样本的个数，进而SMOTE和Random Undersampling用来进一步的增加或者减少样本使其数目达到期望值大小。

算法描述

输入：原始数据集S,SMOTE方法中的紧邻个数k;

输出：通过Random Balance方法构建的新的数据集.

1. 记;
2. 在2到totalSize-2中生成一个随机整数；
3. \
4. if
6. 从中随机抽取大小为个样本，并加入
7. 基于，采用SMOTE方法生成个样本，并加入
8. else
10. 从中随机抽取大小为个样本，并加入
11. 基于，采用SMOTE方法生成个样本，并加入;
12. End if
13. Return

RB-Boost算法

算法描述：

输入：原始数据集S,,弱学习器weakLearn,迭代次数T,SMOTE方法中的紧邻个数k;

输出：构建得到的RB-Boost

1. 初始化分布;
2. ;
3. 如果 ，否则,对;
4. 采用和权重训练weakLearn ;
5. 计算的伪损失
7. 更新
9. 归一化：记
10. End for
11. 返回