General ir

precision recall f1-score support

0 1.00 0.49 0.66 225

1 0.53 1.00 0.69 126

avg / total 0.83 0.68 0.67 351

weighted general ir

precision recall f1-score support

0 1.00 0.74 0.85 225

1 0.68 1.00 0.81 126

avg / total 0.89 0.83 0.84 351

weighted general ir +0.1\*weighted\_overlapping

precision recall f1-score support

0 1.00 0.59 0.74 225

1 0.58 1.00 0.73 126

avg / total 0.85 0.74 0.74 351

增加考虑边界样本的影响

在计算general\_ir时，考虑到集合大小对分类效果并不是决定性的关系，样本分布更加影响分类结果，因此对难以分类的样本，增加其被采样的概率，以获得较好的分类结果。

其次，考虑更一般的情况，在划分边界的分类器中，边界样本对分类效果有比较深远的影响，因此为了增加难以区分的边界样本和噪音类型的非边界样本，我们需要对边界样本的权重进一步调整，希望在这个过程中，能够更加容易地区分出边界难分类样本。

伪代码：

输入：data，label

将数据集划分为positive和negative

对于所有的数据进行循环：

对于该样本，当链条没有收敛时：

compute其不同类别的最近邻和最近距离

compute其最近邻的不同类别的最近邻和最近距离

根据计算结果，对边界样本的index进行确定：

对于某个样本，其距离链中的各个距离进行比较，这里面有个阈值设定：如果前一个距离大于后一个距离的α倍，则认为其不是边界样本，予以删除。

为了降低离群点的采样频率，因为在这种情况下，GIR的值非常高，但是过多的强调这种样本会增加分类器对噪声的敏感程度，因此，我们需要更加精细地对难以分类的样本进行划分：即离群点和边界点；我们对边界样本进行区分，与以往的分类器区分边界点不同，我们采取最近邻的策略对边界点进行判断，在此中引进一种改进的边界点判定方法：

算法：

compute

输入某个样本索引，如果是positive，在negative中寻找其最近邻，保存其最近邻和该最近邻对应的距离；反之亦然。

输入：index，data，label

输出：该index对应最近邻链和距离

tem=-1，link=[]，distance=[]，

while tem not in link：

hey，ha=compute（index，data，label）

link