## Adaboost的原理深入思考

Adaboost本质形式是一个非线性函数（弱分类器）的线性组合，单独从这个非常一般性的描述可以得到一些有用的属性分析。

Adaboost实际使用中，经常出现特征数目远远小于弱分类器数目的情况，此时某些特征被重复选择利用，但是构造出来的弱分类器fm(x)的参数不一样:

上式本质上是一个分段函数。

### 1维特征空间

以输入空间1维{x1}为例，使用Adaboost能够拟合几乎任意形状的分类函数吗？因为输入特征只有一个，因此最终加性判别函数实际上是在组合一大堆的分段函数，这种组合是否能够给出任意分类函数的分段拟合？

答案是肯定的，假设目标分类函数F(x)的形式也是分段函数，取值范围为{-1,1}，分类突变分界点集合为{th1,th2,th3,…thN}，则我们可以在数轴上从左到右处理这些分界点集合对应的弱分类器fm(x)的选取：我们这样选择：

… …

这样的弱分类器选取就可以拟合任意分段F(x)分类函数了。并且如果把任意形状分类函数使用分段函数进行“量化拟合”的话，可以得出结论：1维特征空间中，只要弱分类器数目足够多，就可以拟合任意形状的分类函数。

### 2维特征空间

2维特征空间可以这样来看，在最终的弱分类器线性组合中，所有跟x1有关的项拟合出了一个任意函数H(x1)，所有跟x2有关的项拟合出另一个任意函数G(x2)，则最终分界面函数为：

我们的问题跟之前一样，这个函数能够拟合任意分界曲面F(x1,x2)吗？

这次的答案是否定的，因为容易举反例：就无法分解为(2.1)的形式。

可以发现这种F(x1,x2)正是“异或”运算。跟神经网络问题稍微不同的是，由于这里的弱分类器都是针对一个特征进行构造的，因此无法解决复杂分界曲面问题。如果限定目标函数形式是多项式形式的话，我们可以给出(2.1)能够拟合的形式的普遍形式为：

由于幂可以取负值，因此(2.2)式对应的分界曲面包含双曲线族在第一象限的全部。虽然已经很强了（可以处理圆形，椭圆形，抛物型分界曲面），但是正如分析指出的，由于不包含交叉项，因此简单的旋转椭圆，旋转抛物线等分界曲面就无法处理了。

另外二维空间中的分段函数，也就是像切西瓜一样多次使用线性分类器切割得到的凸多边形，可以使用(2.1)描述吗？如果每个弱分类器都是同时使用x1,x2两个特征值的线性分类器，则可以：例如一个由一组直线的正分类区间围起来的多边形可以这样描述：

… …

则因为是凸多边形，因此最终判别函数F(x,y)只有在所有直线的正区间内时才会为正数，在凸多边形外部的话，只要落在一条边所代表的直线外部就会导致判别函数为负数了。显然凹多边形不可以这样。

### 如何克服这些局限

从另一个角度来看，这些局限并非Adaboost的，乃是由于弱分类器选取形式的问题，如果我们再构造一个弱分类器，其形式为K(x1\*x2)，则最终分界函数为：

则这样的形式就可以几乎囊括所有的多项式形式了。这个解决方法有点类似于SVM的输入空间扩展形式。

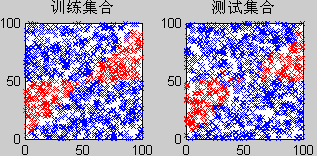
使用matlab简单模拟可以验证上面的结论：

|  |  |  |  |
| --- | --- | --- | --- |
| 线性分界  X1-20 > X2  可以很好处理 |  | 圆形分布  (X1-50)^2 + (X2-50)^2 > 20^2  可以很好处理 |  |
| 双曲  (X1-50)>(300/(x2-50)) |  | (100/(x1-50))+ (100/(x2-50))>1  也可以很好的处理 |  |
| 可以得到结论：  **只要能够很好的分解为(2.1)形式的分界曲面，不论如何复杂，都可以很好的被Adaboost处理。** | | | |
|  |  |  | |
| 以(50,50)为中心的异或:  (X1-50)\* (X2-50)  无法处理 |  | 增加一维特征，就是x1\*x2之后，由于可以被分解为(2.2)形式，因此仍然可以很好的处理 |  |

### 一个很有启发性的例子

下面考虑形式更为怪一点的异或：(X1-50)/ (X2-50)>1，该形式的分界曲面应该如何处理呢？简单变形一下分类函数有：

因此增加特征x1/(x2-50)，简单训练发现效果仍然很差，仔细检查此特征发现，当x2接近50时，此除法产生的值可能是一个非常大的数，而使用matlab的gentleBoost实现训练时，为了简化尝试的阈值个数，使用了某维特征的最大值和最小值范围内均匀划分N个阈值点的方式，这种方式对于分布非常不均匀的特征，或者存在噪声干扰的样本会失效：例如99%的特征都分布在[-1,1]之间，1%的特征分布在[-1000,1000]之间，此时在-1000到1000之间均匀划分100份也无法很好的处理那99%的特征。因此修改代码，逐个尝试阈值为特征的全部集合，虽然运算量大了不少，但是执行结果比较满意：



另外一种变形从分类角度进行等效变换：

上面的不等式从分类角度来看，等效于：

因此展开来看，只要增加x1\*x2项即可，实测为：

|  |  |
| --- | --- |
| 原始特征 | 增加x1\*x2特征之后，存在较大改善 |
|  |  |

虽然通过增加一维组合特征能解决一些新的问题，但是这说明了一个非常重要的问题，那就是，在某种意义上adaboost不会比组成它的弱分类器能力更强，如果说，某些时候adaboost说明了“三个臭皮匠，顶个诸葛亮”，那么另外一些事实说明，有时候再多的臭皮匠都无法顶上一个诸葛亮。因为很多更为复杂的分类器结构是不能使用简单的线性叠加来表达的，此时必须使用类似SVM的方法来扩展特征空间维度。

### 不增加新的特征，而是扩展Adaboost的组合形式，用决策树构造弱分类器

决策树跟Adaboost的组合很早以前就被机器学习领域的专家评价为目前可实用化的最好的分类器组合，二者的组合可以轻松解决异或问题甚至多类聚类的问题：

|  |  |  |  |
| --- | --- | --- | --- |
| 以(50,50)为中心的异或:  (X1-50)\* (X2-50)>0  无法处理 |  | 仍然使用2维特征X1，X2，但是使用深度2的CART决策树作为弱分类器，只需要10级弱分类器即可得到很好的结果 |  |

|  |  |
| --- | --- |
| 多类问题，  分别是位于3个地点的半径10的圆  100级的CARTD2决策树仍然可以很好的处理。原因是决策树可以更好的切割目标空间。 |  |

### Viola Jones文章中的对角线Haar特征是否多余？

根据上面的启发，我们发现，一个对角线Haar特征弱分类器似乎可以使用两个水平或者垂直Haar特征弱分类器线性组合得到：

|  |  |
| --- | --- |
|  | 对角线Haar弱分类器为：  而两个水平Haar弱分类器为：  现在的问题是，可以使用（可能多个不同的）**后两种**弱分类器线性组合成为**前一种**弱分类器吗？令Da=A1-A2, Db=B1-B2有hd(x)是两个特征的线性组合：  其分解面是(Da,Db)坐标系中的一个斜线。  而任意的线性组合只能表达矩形区域：  除非对组合的输出再加上一个非线性映射，不然很难达到我们的目的。 |