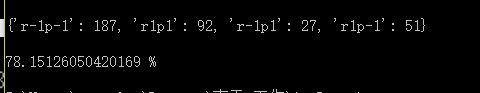
AdaBoost – SVM

陈肖 2016.8.29

本周通过AdaBoost对之前的算法进行了改进，获得了进一步的提高，最高的胜率可以达到78.15%。相对于之前的Bagging – SVM,获得了1.5%的胜率提升。



1. **AdaBoost 算法简介**

AdaBoost, 是英文“Adaptive Boosting”的缩写，于1995年提出。它的自适应性在于分类器分错的样本将会得到加强，加权后的全体样本将再次被用来训练下一个分类器。同时，在每一轮中加入一个新的弱分类器，直到达到某个预定的足够小的错误率或者预先指定的最大迭代次数。

1. **算法流程**

假设样本的类别为：[clip_image002](http://images.cnitblog.com/blog/407700/201410/162249594666427.png)

1、训练部分

a) 为训练集中每个样本分配权重[clip_image004](http://images.cnitblog.com/blog/407700/201410/162250022942658.png)；

b) 训练弱分类器[clip_image006](http://images.cnitblog.com/blog/407700/201410/162250064666632.png)；

c) 计算带权分类误差[clip_image008](http://images.cnitblog.com/blog/407700/201410/162250109985650.png)；

d) 计算弱分类器对应的权重[clip_image010](http://images.cnitblog.com/blog/407700/201410/162250119042794.png)

e) 更新样本权重[clip_image012](http://images.cnitblog.com/blog/407700/201410/162250129047966.png)

最终得到[clip_image014](http://images.cnitblog.com/blog/407700/201410/162250142794953.png)个不同的弱分类器及其对应的权重。 [clip_image016](http://images.cnitblog.com/blog/407700/201410/162250150139811.png)

2、分类部分

a) 对待分类样本[clip_image018](http://images.cnitblog.com/blog/407700/201410/162250160766768.png)计算函数[clip_image020](http://images.cnitblog.com/blog/407700/201410/162250172948240.png)

b) 对该样本最终的分类结果为[clip_image022](http://images.cnitblog.com/blog/407700/201410/162250183579899.png)

1. **具体代码和流程**

见压缩文件夹，内有整个project的代码和详尽的流程。

1. **缺陷**

缺陷主要是计算时间的问题，现在整个包括建模到计算（建模使用1000个数据，预测400个）大概需要15分钟的时间，i5-3210 CPU, 2.5Ghz, 4.00GB内存。

1. **下一步**
2. 自己准备在电脑上通过虚拟机搭建一个Hadoop平台。通过并行计算，节省整个过程的时间。
3. 现在的SVM是未调制参数的，下一步可以调整SVM的参数，看是否能够获得更高的胜率。