个人资料

一只鸟的天空

访问： 179832次
积分： 3166
等级： BLOG 5
排名： 第6122名

原创： 114篇 转载： 22篇
译文： 2篇 评论： 49条

文章搜索

文章分类

C、C++开发 (87)
C#开发 (5)
JAVA开发 (2)
个人情感 (0)
电脑知识 (2)
VS应用 (2)
C++ XML (1)
图形图像 (1)
算法设计与分析 (6)
微信开放平台 (2)
Hadoop (10)
数据挖掘与机器学习 (23)
HBase学习 (5)
mahout (0)
神经网络 (0)

文章存档

2015年10月 (4)
2015年09月 (4)
2015年08月 (4)
2015年02月 (2)
2014年09月 (2)

展开

[置顶] 在分类中如何处理训练集中不平衡问题

标签： 分类 数据不平衡 类别不均衡 imbalance 机器学习

2015-10-25 23:09 853人阅读 评论(0) 收藏 举报

分类： 数据挖掘与机器学习（22）

原文地址：一只鸟的天空，http://blog.csdn.net/heyongluoyao8/article/details/49408131

在分类中如何处理训练集中不平衡问题

在很多机器学习任务中，训练集中可能会存在某个或某些类别下的样本数远大于另一些类别下的样本数目。即类别不平衡，为了使得学习达到更好的效果，因此需要解决该类别不平衡问题。

Jason Brownlee的回答：

原文标题：8 Tactics to Combat Imbalanced Classes in Your Machine Learning Dataset

当你在对一个类别不均衡的数据集进行分类时得到了90%的准确度（Accuracy）。当你进一步分析发现，数据集的90%的样本是属于同一个类，并且分类器将所有的样本都分类为该类。在这种情况下，显然该分类器是无效的。并且这种无效是由于训练集中类别不均衡而导致的。

首先举几个所收到的邮件中关于类别不均衡的例子：

- 在一个二分类问题中，训练集中class 1的样本数比class 2的样本数是60:1。使用逻辑回归进行分类，最后结果是其忽略了class 2，即其将所有的训练样本都分类为class 1。
- 在分类任务的数据集中，有三个类别，分别为A，B，C。在训练集中，A类的样本占70%，B类的样本占25%，C类的样本占5%。最后我的分类器对类A的样本过拟合了，而对其它两个类别的样本欠拟合。

什么是类别不均衡问题

类别数据不均衡是分类任务中一个典型的存在的问题。简而言之，即数据集中，每个类别下的样本数目相差很大。例如，在一个二分类问题中，共有100个样本（100行数据，每一行数据为一个样本的表征），其中80个样本属于class 1，其余的20个样本属于class 2，class 1:class2=80:20=4:1，这便属于类别不均衡。当然，类别不均衡问同样会发生在多分类任务中。它们的解决方法是一样的。因此，为了便于讨论与理解，我们从二分类任务入手进行讲解。

类别不均衡问题是现实中很常见的问题

大部分分类任务中，各类别下的数据个数基本上不可能完全相等，但是一点点差异是不会产生任何影响与问题的。

在现实中有很多类别不均衡问题，它是常见的，并且也是合理的，符合人们期望的。如，在欺诈交易识别中，属于欺诈交易的应该是很少部分，即绝大部分交易是正常的，只有极少部分的交易属于欺诈交易。这就是一个正常的类别不均衡问题。又如，在客户流失的数据集中，绝大部分的客户是会继续享受其服务的（非流失对象），只有极少数部分的客户不会再继续享受其服务（流失对象）。一般而已，如果类别不平衡比例超过4:1，那么其分类器会大大地因为数据不平衡性而无法满足分类要求的。因此在构建分类模型之前，需要对分类不均衡

阅读排行

Hadoop与HBase中遇到| (8553)
MapReduce生成HFile文 (6684)
Windows下使用Word2ve (6088)
C++string知识大全 (5067)
循环神经网络(RNN, Rec (4096)
C++产生m到n之间的随机 (3883)
常用的机器学习&数据挖掘 (3197)
区间图着色问题（贪心算 (3060)
判断一个图是否连通 (2683)
微信开发平台教程(1) (2573)

评论排行

循环神经网络(RNN, Rec (7)
MapReduce生成HFile文 (5)
Hadoop与HBase中遇到| (4)
Windows下使用Word2ve (3)
常见的机器学习&数据挖掘 (3)
C++中int型与string型互相 (3)
当今世界最NB的25位大数 (2)
流水线调度最优问题（装 (2)
C++string知识大全 (2)
二叉查找树（二叉排序树 (2)

推荐文章

*机器学习与数据挖掘网上资源搜罗——良心推荐
*架构设计：系统间通信（17）——服务治理与Dubbo 中篇（分析）
*数据库性能优化之SQL语句优化
*Android应用开发allowBackup敏感信息泄露的一点反思
*Linux多线程实践（四 ）线程的特定数据
*Android点击Button水波纹效果

最新评论

Windows下使用Word2vec继续训
p_sunrise: 您好，请问您现在还有训练好的词向量吗？
常见的机器学习&数据挖掘知识点
淇元: 博主在做一件伟大的事情。
当今世界最NB的25位大数据科学
happy勇敢的心: 多谢博主！ 雪中送碳！
循环神经网络(RNN, Recurrent N
一只鸟的天空: @FLiszt:谢谢，说得对，应该为循环神经网络
循环神经网络(RNN, Recurrent N
一只鸟的天空: 我的意思是值会不停的优化，只是使用同一个矩阵，而多隐藏层之间的矩阵是不同的。
循环神经网络(RNN, Recurrent N
一只鸟的天空: @jjlqiuzhizhe:我的意思是值会不停的优化，只是使用同一个矩阵，而多隐藏层之间的矩阵是不同...
循环神经网络(RNN, Recurrent N
Flying-J: 文章中"如果这是一个多层的传统神经网络，那么xt到st之间的U矩阵与xt+1到st+1之间的U是不同...
循环神经网络(RNN, Recurrent N
一只鸟的天空: @FLiszt:是一样的

性问题进行处理。

在前面，我们使用准确度这个指标来评价分类质量，可以看出，在类别不均衡时，准确度这个评价指标并不能work。因为分类器将所有的样本都分类到大类下面时，该指标值仍然会很高。即，该分类器偏向了大类这个类别的数据。

八大解决方法

- 可以扩大数据集吗？

  当遇到类别不均衡问题时，首先应该想到，是否可能再增加数据（一定要有小类样本数据），更多的数据往往战胜更好的算法。因为机器学习是使用现有的数据多整个数据的分布进行估计，因此更多的数据往往能够得到更多的分布信息，以及更好分布估计。即使再增加小类样本数据时，又增加了大类样本数据，也可以使用放弃一部分大类数据（即对大类数据进行欠采样）来解决。

- 尝试其它评价指标

  从前面的分析可以看出，准确度这个评价指标在类别不均衡的分类任务中并不能work，甚至进行误导（分类器不work，但是从这个指标来看，该分类器有着很好的评价指标得分）。因此在类别不均衡分类任务中，需要使用更有说服力的评价指标来对分类器进行评价。如何对不同的问题选择有效的评价指标参见这里。

  上面的超链接中的文章，讲述了如何对乳腺癌患者复发类别不均衡数据进行分类。在文中，推荐了几个比传统的准确度更有效的评价指标：

  - 混淆矩阵(Confusion Matrix)：使用一个表格对分类器所预测的类别与其真实的类别的样本统计，分别为：TP、FN、FP与TN。
  - 精确度(Precision)
  - 召回率(Recall)
  - F1得分(F1 Score)：精确度与找召回率的加权平均。

  特别是：

  - Kappa (Cohen kappa)
  - ROC曲线(ROC Curves)：见Assessing and Comparing Classifier Performance with ROC Curves

- 对数据集进行重采样

  可以使用一些策略该减轻数据的不平衡程度。该策略便是采样(sampling)，主要有两种采样方法来降低数据的不平衡性。

  - 对小类的数据样本进行采样来增加小类的数据样本个数，即过采样（over-sampling ，采样的个数大于该类样本的个数）。
  - 对大类的数据样本进行采样来减少该类数据样本的个数，即欠采样（under-sampling，采样的次数少于该类样本的个素）。

  采样算法往往很容易实现，并且其运行速度快，并且效果也不错。更详细的内容参见这里。

  一些经验法则：

  - 考虑对大类下的样本（超过1万、十万甚至更多）进行欠采样，即删除部分样本；
  - 考虑对小类下的样本（不足1为甚至更少）进行过采样，即添加部分样本的副本；
  - 考虑尝试随机采样与非随机采样两种采样方法；
  - 考虑对各类别尝试不同的采样比例，比一定是1:1，有时候1:1反而不好，因为与现实情况相差甚远；
  - 考虑同时使用过采样与欠采样。

- 尝试产生人工数据样本

  一种简单的人工样本数据产生的方法便是，对该类下的所有样本每个属性特征的取值空间中随机选取一个组成新的样本，即属性值随机采样。你可以使用基于经验对属性值进行随机采样而构造新的人工样本，或者使用类似朴素贝叶斯方法假设各属性之间互相独立进行采样，这样便可得到更多的数据，但是无法保证属性之前的线性关系（如果本身是存在的）。

  有一个系统的构造人工数据样本的方法SMOTE(Synthetic Minority Over-sampling Technique)。SMOTE是一种过采样算法，它构造新的小类样本而不是产生小类中已有的样本的副本，即该算法构造的数据是新样本，原数据集中不存在的。该基于距离度量选择小类别下两个或者更多的相似样本，然后选择其中一个样本，并随机选择一定数量的邻居样本对选择的那个样本的一个属性增加噪声，每次处理一个属性。这样就构造了更多的新生数据。具体可以参见原始论文。

意思

当今世界最NB的25位大数据科学springXu: MARK

常见的机器学习&数据挖掘知识点ttangqichao: 博主归纳的很强大，佩服

这里有SMOTE算法的多个不同语言的实现版本：

- Python: UnbalancedDataset模块提供了SMOTE算法的多种不同实现版本，以及多种重采样算法。
- R: DMwR package。
- Weka: SMOTE supervised filter。

- 尝试不同的分类算法

  强烈建议不要对待每一个分类都使用自己喜欢而熟悉的分类算法。应该使用不同的算法对其进行比较，因为不同的算法使用于不同的任务与数据。具体可以参见“Why you should be Spot-Checking Algorithms on your Machine Learning Problems”。

  决策树往往在类别不均衡数据上表现不错。它使用基于类变量的划分规则去创建分类树，因此可以强制地将不同类别的样本分开。目前流行的决策树算法有：C4.5、C5.0、CART和Random Forest等。基于R编写的决策树参见这里。基于Python的Scikit-learn的CART使用参见这里。

- 尝试对模型进行惩罚

  你可以使用相同的分类算法，但是使用一个不同的角度，比如你的分类任务是识别那些小类，那么可以对分类器的小类样本数据增加权值，降低大类样本的权值（这种方法其实是产生了新的数据分布，即产生了新的数据集，译者注），从而使得分类器将重点集中在小类样本身上。一个具体做法就是，在训练分类器时，若分类器将小类样本分错时额外增加分类器一个小类样本分错代价，这个额外的代价可以使得分类器更加“关心”小类样本。如penalized-SVM和penalized-LDA算法。

  Weka中有一个惩罚模型的通用框架CostSensitiveClassifier，它能够对任何分类器进行封装，并且使用一个自定义的惩罚矩阵对分错的样本进行惩罚。

  如果你锁定一个具体的算法时，并且无法通过使用重采样来解决不均衡性问题而得到较差的分类结果。这样你便可以使用惩罚模型来解决不平衡性问题。但是，设置惩罚矩阵是一个复杂的事，因此你需要根据你的任务尝试不同的惩罚矩阵，并选取一个较好的惩罚矩阵。

- 尝试一个新的角度理解问题

  我们可以从不同于分类的角度去解决数据不均衡性问题，我们可以把那些小类的样本作为异常点(outliers)，因此该问题便转化为异常点检测(anomaly detection)与变化趋势检测问题(change detection)。

  异常点检测即是对那些罕见事件进行识别。如通过机器的部件的振动识别机器故障，又如通过系统调用序列识别恶意程序。这些事件相对于正常情况是很少见的。

  变化趋势检测类似于异常点检测，不同在于其通过检测不寻常的变化趋势来识别。如通过观察用户模式或银行交易来检测用户行为的不寻常改变。

  将小类样本作为异常点这种思维的转变，可以帮助考虑新的方法去分离或分类样本。这两种方法从不同的角度去思考，让你尝试新的方法去解决问题。

- 尝试创新

  仔细对你的问题进行分析与挖掘，是否可以将你的问题划分成多个更小的问题，而这些小问题更容易解决。你可以从这篇文章In classification, how do you handle an unbalanced training set?中得到灵感。例如：

  - 将你的大类压缩成小类；
  - 使用One Class分类器（将小类作为异常点）；
  - 使用集成方式，训练多个分类器，然后联合这些分类器进行分类；
  - ....

这些想法只是冰山一角，你可以想到更多的有趣的和有创意的想法去解决问题。更多的想法参加Reddit的文章http://www.quora.com/In-classification-how-do-you-handle-an-unbalanced-training-set。

选择某一种方法并使用它

你不必成为一个精通所有算法的算法奇才或者一个建立准确而可靠的处理数据不平衡的模型的统计学家，你只需要根据你的问题的实际情况从上述算法或方法中去选择一种或两种方法去使用。希望上述的某些方法能够解决你的问题。例如使用其它评价指标或重采样算法速度快并且有效。

总结

记住，其实并不知道哪种方法最适合你的任务与数据，你可以使用一些启发式规则或经验去选择某一个较优算法。当然最好的方法测试每一种算法，然后选择最好的方法。最重要的是，从点滴开始做起，根据自己现有的知识，并不断学习去一步步完善。

Further Reading...

这里有一些我认为有价值的可供参考的相关资料，让你进一步去认识与研究数据不平衡问题：

- 相关书籍
  - Imbalanced Learning: Foundations, Algorithms, and Applications
- 相关论文
  - Data Mining for Imbalanced Datasets: An Overview
  - Learning from Imbalanced Data
  - Addressing the Curse of Imbalanced Training Sets: One-Sided Selection (PDF)
  - A Study of the Behavior of Several Methods for Balancing Machine Learning Training Data

Sergey Feldman的回答：

- 设超大类中样本的个数是极小类中样本个数的L倍，那么在随机梯度下降（SGD，stochastic gradient descent）算法中，每次遇到一个极小类中样本进行训练时，训练L次。
- 将大类中样本划分到L个聚类中，然后训练L个分类器，每个分类器使用大类中的一个簇与所有的小类样本进行训练得到。最后对这L个分类器采取少数服从多数对未知类别数据进行分类，如果是连续值（预测），那么采用平均值。
- 设小类中有N个样本。将大类聚类成N个簇，然后使用每个簇的中心组成大类中的N个样本，加上小类中所有的样本进行训练。
- 无论你使用前面的何种方法，都对某个或某些类进行了损害。为了不进行损害，那么可以使用全部的训练集采用多种分类方法分别建立分类器而得到多个分类器，采用投票的方式对未知类别的数据进行分类，如果是连续值（预测），那么采用平均值。
- 在最近的ICML论文中，表明增加数据量使得已知分布的训练集的误差增加了，即破坏了原有训练集的分布，从而可以提高分类器的性能。这篇论文与类别不平衡问题不相关，因为它隐式地使用数学方式增加数据而使得数据集大小不变。但是，我认为破坏原有的分布是有益的。
- More details than you need: imho, the most interesting of the corrupting distributions is the blankout distribution, where you just zero out a random subset of features. Why is it interesting? Because you are helping your classifier be sturdier/hardier by giving it variations of your data that have essentially missing features. So it has to learn to classify correctly even in adverse conditions. 一个相关的想法是，在神经网络中，随机选择部分隐藏层单元来继续训练（即，随机去掉一部分隐藏层单元，(zeroed-out)）。具体见http://web.stanford.edu/~sidaw/cgi-bin/home/lib/exe/fetch.php?media=papers:fastdropout.pdf

Kripa Chettiar的回答：

- 增加新数据，可以使用SMOTE或SMOTEBoost产生人造数据。
- 将大类压缩。压缩比例需要具体情况具体分析，取决于你所拥有的数据。例如，A类中有30个样本，B类中有4000个样本，那么你可以将B类压缩成1000（进行采样）。
- 可以结合1与2
- 对于那种极小类是异常点的分类任务，因此分类器需要学习到大类的决策分界面，即分类器是一个单个类分类器（One Class Classifier）。Weka中有相关的库。
- 获得更多的数据。

Roar Nybø的回答：

- 对小类进行过采样。并且使用集成模式会获得更好的效果。

Dan Levin的回答：

- 一个很好的方法去处理非平衡数据问题，并且在理论上证明了。这个方法便是由Robert E. Schapire于1990年在Machine Learning提出的" The strength of weak learnability" ，该方法是一个boosting算法，它递归地训练三个弱学习器，然后将这三个弱学习器结合起形成一个强的学习器。我们可以使用这个算法的第一步去解决数据不平衡问题。

  首先使用原始数据集训练第一个学习器L1。

  然后使用50%在L1学习正确和50%学习错误的的那些样本训练得到学习器L2，即从L1中学习错误的样本集与学习正确的样本集中，循环一边采样一个。

  接着，使用L1与L2不一致的那些样本去训练得到学习器L3。

最后，使用投票方式作为最后输出。

那么如何使用该算法来解决类别不平衡问题呢？

假设是一个二分类问题，大部分的样本都是true类。让L1输出始终为true。使用50%在L1分类正确的与50%分类错误的样本训练得到L2，即从L1中学习错误的样本集与学习正确的样本集中，循环一边采样一个。因此，L2的训练样本是平衡的。L使用L1与L2分类不一致的那些样本训练得到L3，即在L2中分类为false的那些样本。最后，结合这三个分类器，采用投票的方式来决定分类结果，因此只有当L2与L3都分类为false时，最终结果才为false，否则true。

自己已经在实践中使用过很多次，并且效果都不错。

Kaushik Kasi的回答：

- 对小类中的样本进行复制以增加该类中的样本数，但是可能会增加bias。
- 对小类中的样本通过调整特征值来人工生成样本，而使得该类中样本个数增多。如在图像中，对一幅图像进行扭曲得到另一幅图像，即改变了原图像的某些特征值。但是该方法可能会产生现实中并存在的样本。

Quora User的回答：

- 简单快速的方法：对大类欠采样或者对小类过采样。
- 更有效的方法：使用代价函数学习得到每个类的权值，大类的权值小，小类的权值大。刚开始，可以设置每个类别的权值与样本个数比例的倒数，然后可以使用过采样进行调优。

Dayvid Victor的回答：

在类别不平衡中，以下几个点需要注意：

- 常规的分类评价指标可能会失效，比如将所有的样本都分类成大类，那么准确率、精确率等都会很高。这种情况下，AUC 标。
- 你能够使用原型选择技术去降低不平衡水平。选择那些重要的样本。One-Sided Selection (OSS) 是一个预处理技术（模 ），能够处理类别不平衡问题。
- 从另一个角度， 样本个数，可以使用过采样与原型生成技术（prototype-generation techniques）。
- 在K-Fold 校验中，每一份数据集中原则上应该保持类别样本比例一样或者近似，如果每份数据集中小类样本数目过少，那么应该降低K的值，知道小类样本的个数足够。

一般来说，如果事前不对不平衡问题进行处理，那么对于小类别的样本则会错误率很高，即大部分甚至全部小类样本都会分错。

Muktabh Mayank的回答：

- 这里有一个类似SVM的方法来处理不平衡问题。具体参见这里。

Sandeep Subramanian的回答：

- 使用SMOTE（Synthetic Minority Oversampling TEchnique）方法人工生成小类数据。其类似于最近邻算法。

Quora User的回答：

- 赋予小类样本更高的训练权值
- 对小类进行过采样
- 某些时候，高不平衡性下仍然可以得到效果较好的训练结果。我认为对于某些评价指标是有意义的，如AUC。

Sumit Soman 的回答：

- 如果你使用SVM分类器进行分类，那么可以使用Twin SVM（Twin Support Vector Machines for Pattern Classification），其能够应付类别不平衡问题。

Abhishek Ghose的回答：

参见：Abhishek Ghose’s answer to What’s the most efficient classification algorithm for unbalanced data sets? And what pre-processing could be done to optimize the score?

原文：https://www.quora.com/In-classification-how-do-you-handle-an-unbalanced-training-set

顶 踩
2 0

上一篇 循环神经网络(RNN, Recurrent Neural Networks)介绍
下一篇 机器学习中防止过拟合的处理方法

我的同类文章

数据挖掘与机器学习（22）

- 推荐算法知识
- 主要的推荐算法简介
- 数据挖掘与机器学习十八大经典算法
- Python机器学习库
- Windows下使用Word2vec继续词向量训练
- 大数据之"用户行为分析"
- 个性化推荐技术的十大挑战
- 机器学习之开源库大总结
- 最优化之无约束优化
- NLTK的词性

更多

主题推荐 class

猜你在找

有趣的算法（数据结构）
数据结构基础系列(1)：数据结构和算法
数据结构和算法
Spark 1.x大数据平台
《C语言/C++学习指南》加密解密篇（安全相关算法）
分类算法简介
各种分类算法比较
各常用分类算法的优缺点总结
数据挖掘--分类算法的优缺点
各种分类算法比较

查看评论

暂无评论

您还没有登录,请[登录]或[注册]

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

核心技术类目

全部主题 Hadoop AWS 移动游戏 Java Android iOS Swift 智能硬件 Docker
OpenStack VPN Spark ERP IE10 Eclipse CRM JavaScript 数据库 Ubuntu NFC
WAP jQuery BI HTML5 Spring Apache .NET API HTML SDK IIS Fedora XML
LBS Unity Splashtop UML components Windows Mobile Rails QEMU KDE Cassandra
CloudStack FTC coremail OPhone CouchBase 云计算 iOS6 Rackspace Web App
SpringSide Maemo Compuware 大数据 aptech Perl Tornado Ruby Hibernate ThinkPHP
HBase Pure Solr Angular Cloud Foundry Redis Scala Django Bootstrap