北风网
IBEIFENG.COM

人工智能之机器学习

数据清洗和特征工程

上海育创网络科技股份有限公司

主讲人：刘老师(GerryLiu)

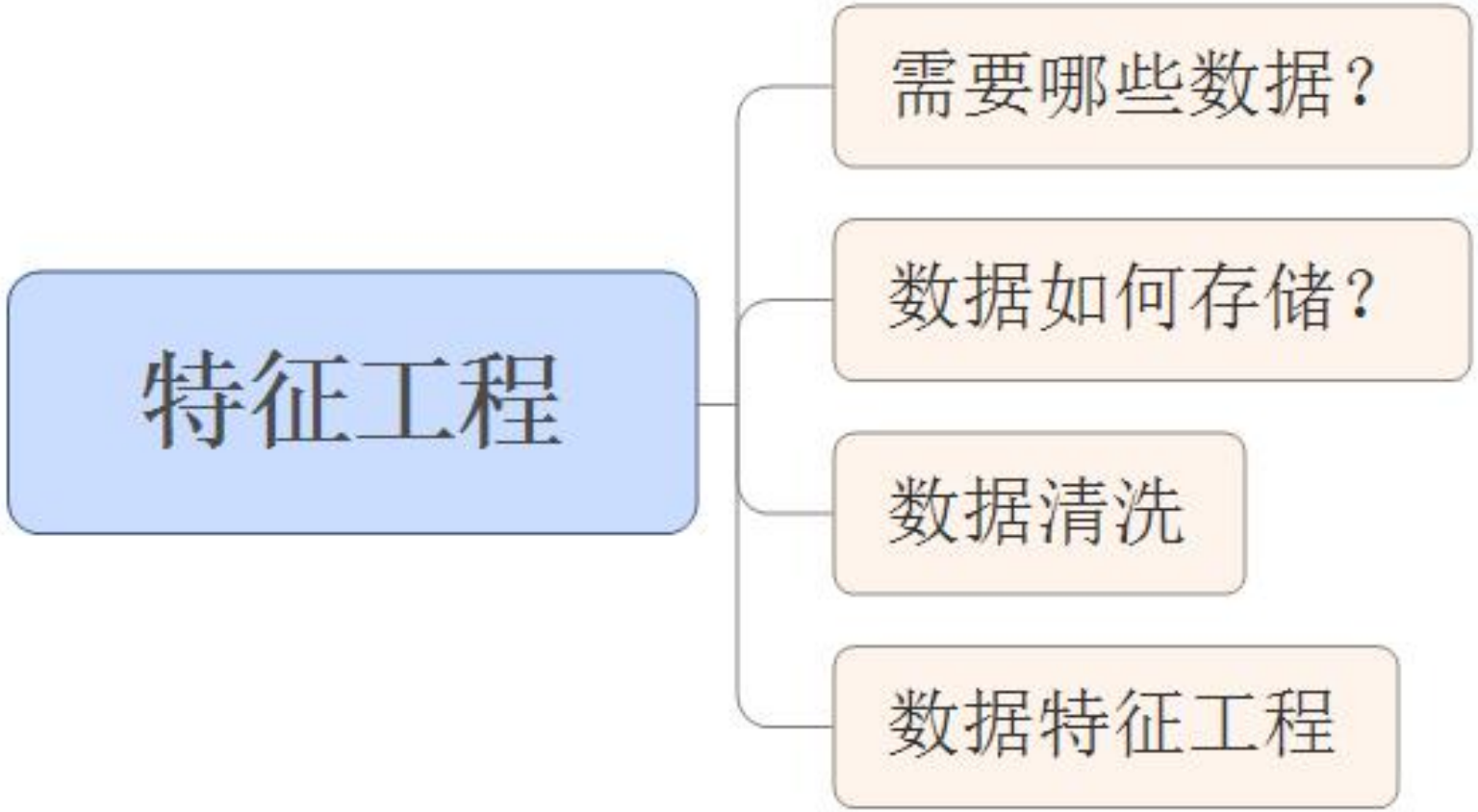
课程要求

- 课上课下“九字”真言
  - 认真听，**善摘录，勤思考**
  - **多温故，乐实践**，再发散
- 四不原则
  - 不懒散惰性，不迟到早退
  - 不请假旷课，不拖延作业

特征工程做一个总结

- 所有一切为了**让模型效果变的更好**的数据处理方式都可以认为属于特征工程这个范畴中的一个操作；
- 至于需求做不做这个特征工程，需要我们在开发过程中不但的进行尝试。
- 常规的特征工程需要处理的内容：
  - 异常数据的处理
  - 数据不平衡处理
  - 文本处理：词袋法、TF-IDF
  - 多项式扩展、哑编码、标准化、归一化、区间缩放法、PCA、特征选择.....
  - 将均值、方差、协方差等信息作为特征属性，对特征属性进行对数转换、指数转换.....
  - 结合业务衍生出一些新的特征属性....

知识要点



需要哪些数据?

- 在进行机器学习之前，收集数据的过程中，我们主要按照以下规则找出我们所需要的数据：
  - 1. 业务的实现需要哪些数据?
    - 基于对业务规则的理解，尽可能多的找出对因变量有影响的所有自变量数据。
  - 2. 数据可用性评估
    - 在获取数据的过程中，首先需要考虑的是这个数据获取的成本；
    - 获取得到的数据，在使用之前，需要考虑一下这个数据是否覆盖了所有情况以及这个数据的可信度情况。

需要哪些数据?

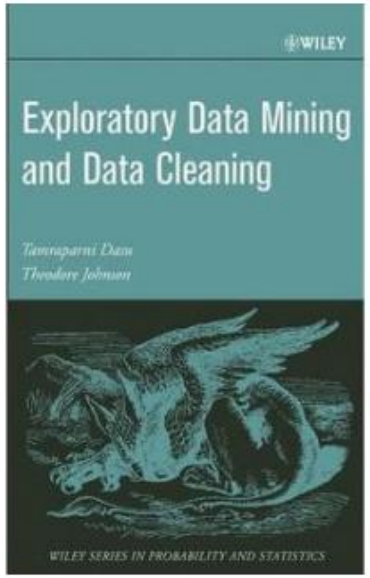
- 一般公司内部做机器学习的数据源:
  - 用户行为日志数据：记录的用户在系统上所有操作所留下来的日志行为数据
  - 业务数据：商品/物品的信息、用户/会员的信息……
  - 第三方数据：爬虫数据、购买的数据、合作方的数据....

数据如何存储?

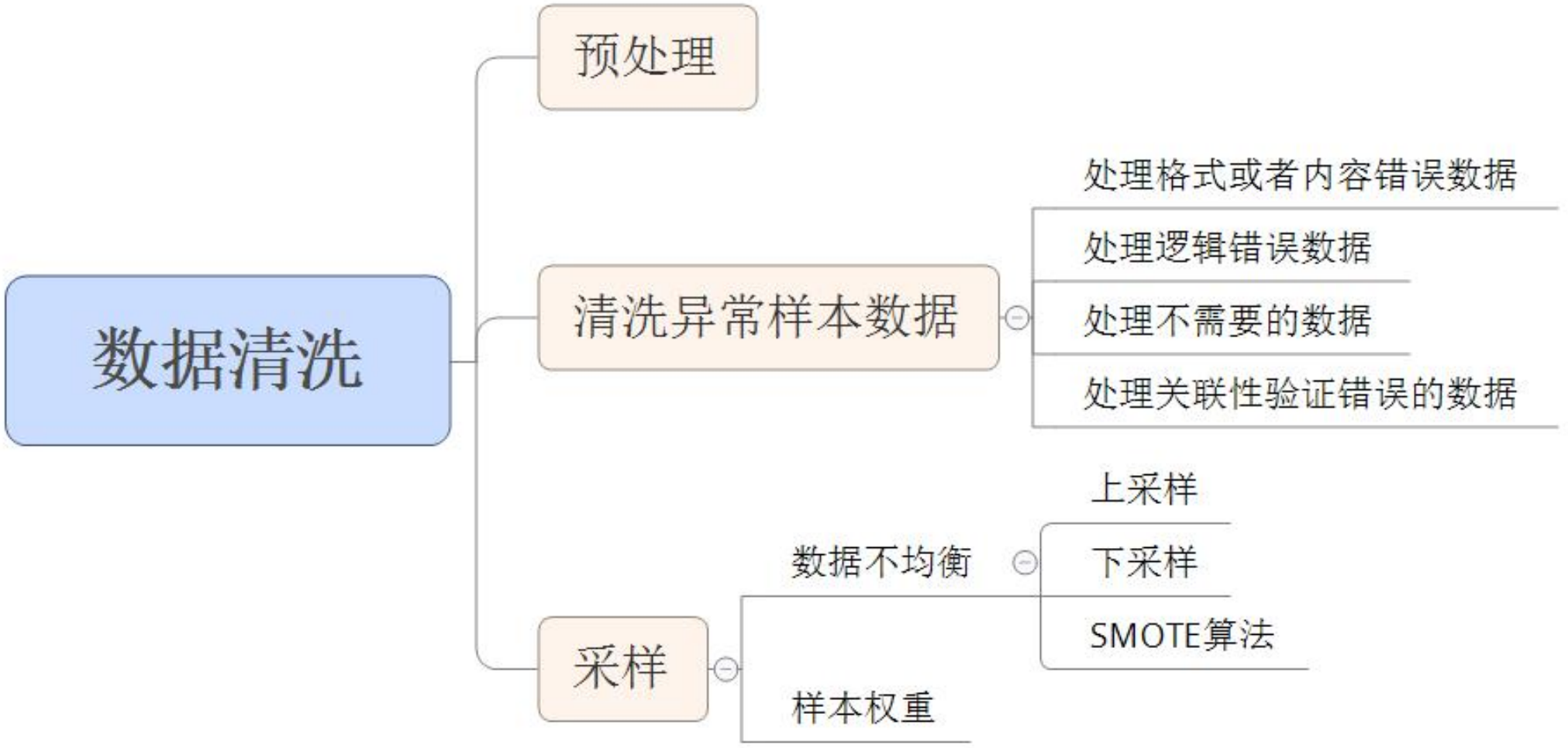
- 一般情况下，用于后期模型创建的数据都是存在在本地磁盘、关系型数据库或者一些相关的分布式数据存储平台的。
  - 本地磁盘
  - MySQL
  - Oracle
  - HBase
  - HDFS
  - Hive

数据清洗

- 数据清洗(data cleaning)是在机器学习过程中一个不可缺少的环节，其数据的清洗结果直接关系到模型效果以及最终的结论。在实际的工作中，数据清洗通常占开发过程的30%-50%左右的时间。



数据清洗过程


数据清洗
预处理
清洗异常样本数据
处理格式或者内容错误数据
处理逻辑错误数据
处理不需要的数据
处理关联性验证错误的数据
采样
数据不均衡
上采样
下采样
SMOTE算法
样本权重

数据清洗--预处理

- 在数据预处理过程主要考虑两个方面，如下：
  - 选择数据处理工具：关系型数据库或者Python
  - 查看数据的元数据以及数据特征：一是查看元数据，包括字段解释、数据来源等一切可以描述数据的信息；另外是抽取一部分数据，通过人工查看的方式，对数据本身做一个比较直观的了解，并且初步发现一些问题，为之后的数据处理做准备。

数据清洗--格式内容错误数据清洗

- 一般情况下，数据是由用户/访客产生的，也就有很大的可能性存在格式和内容上不一致的情况，所以在进行模型构建之前需要先进行数据的格式内容清洗操作。格式内容问题主要有以下几类：
  - 时间、日期、数值、半全角等显示格式不一致：直接将数据转换为一类格式即可，该问题一般出现在多个数据源整合的情况下。
  - 内容中有不该存在的字符：最典型的就是在头部、中间、尾部的空格等问题，这种情况下，需要以半自动校验加半人工方式来找出问题，并去除不需要的字符。
  - 内容与该字段应有的内容不符：比如姓名写成了性别、身份证号写成手机号等问题。

数据清洗--逻辑错误清洗

- 主要是通过简单的逻辑推理发现数据中的问题数据，防止分析结果走偏，主要包含以下几个步骤：
  - 数据去重
  - 去除/替换不合理的值
  - 去除/重构不可靠的字段值(修改矛盾的内容)

数据清洗--去除不需要的数据

- 一般情况下，我们会尽可能多的收集数据，但是不是所有的字段数据都是可以应用到模型构建过程的，也不是说将所有的字段属性都放到构建模型中，最终模型的效果就一定会好，实际上来讲，字段属性越多，模型的构建就会越慢，所以有时候可以考虑将不要的字段进行删除操作。在进行该过程的时候，要注意备份原始数据。

数据清洗--关联性验证

- 如果数据有多个来源，那么有必要进行关联性验证，该过程常应用到多数据源合并的过程中，通过验证数据之间的关联性来选择比较正确的特征属性，比如：汽车的线下购买信息和电话客服问卷信息，两者之间可以通过姓名和手机号进行关联操作，匹配两者之间的车辆信息是否是同一辆，如果不是，那么就需要进行数据调整。

数据不平衡

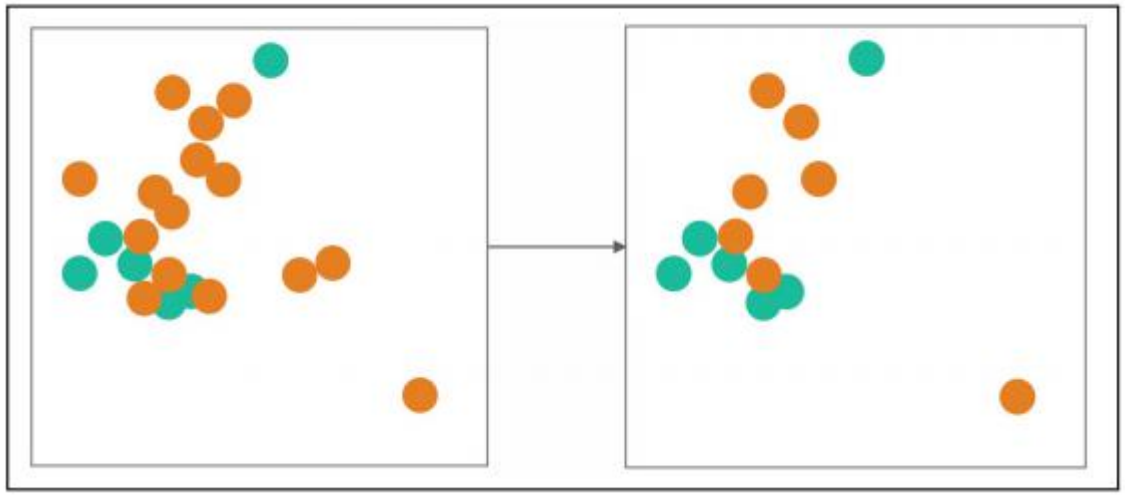
- 在实际应用中，数据往往分布得非常不均匀，也就是会出现“长尾现象”，即绝大多数的数据在一个范围/属于一个类别，而在另外一个范围或者另外一个类别中，只有很少的一部分数据。那么这个时候直接使用机器学习可能效果会不太少，所以这个时候需要我们进行一系列的转换操作。

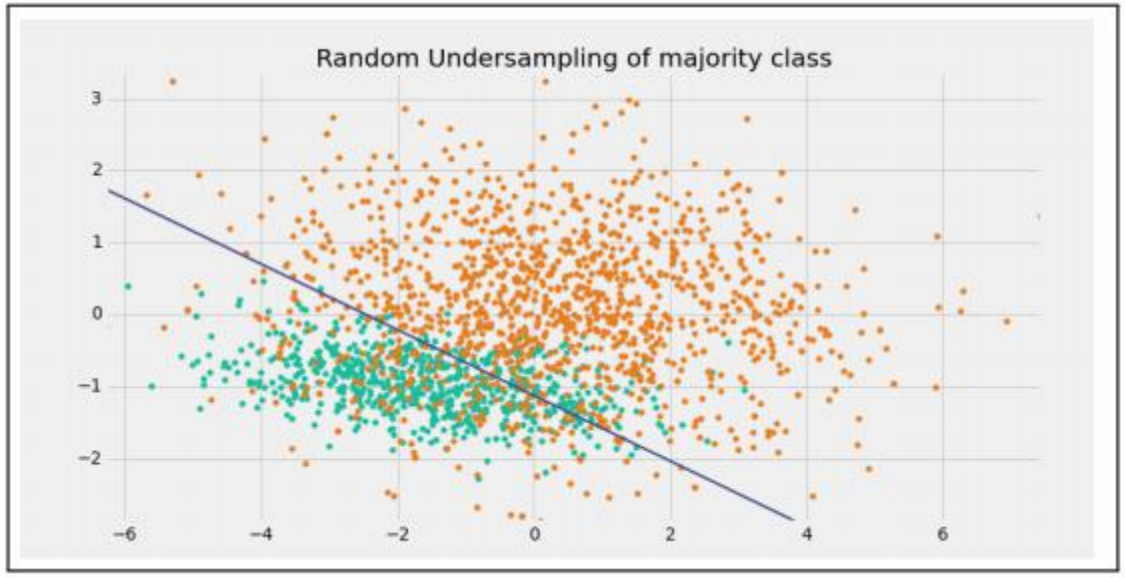
数据不平衡解决方案一

- 设置损失函数的权重，使得少数类别数据判断错误的损失大于多数类别数据判断错误的损失，即当我们的少数类别数据预测错误的时候，会产生一个比较大的损失值，从而导致模型参数往让少数类别数据预测准确的方向偏。可以通过scikit-learn中的class_weight参数来设置权重。

数据不平衡解决方案二

- 下采样/欠采样(under sampling)：从多数类中随机抽取样本从而减少多数类别样本数据，使数据达到平衡的方式。



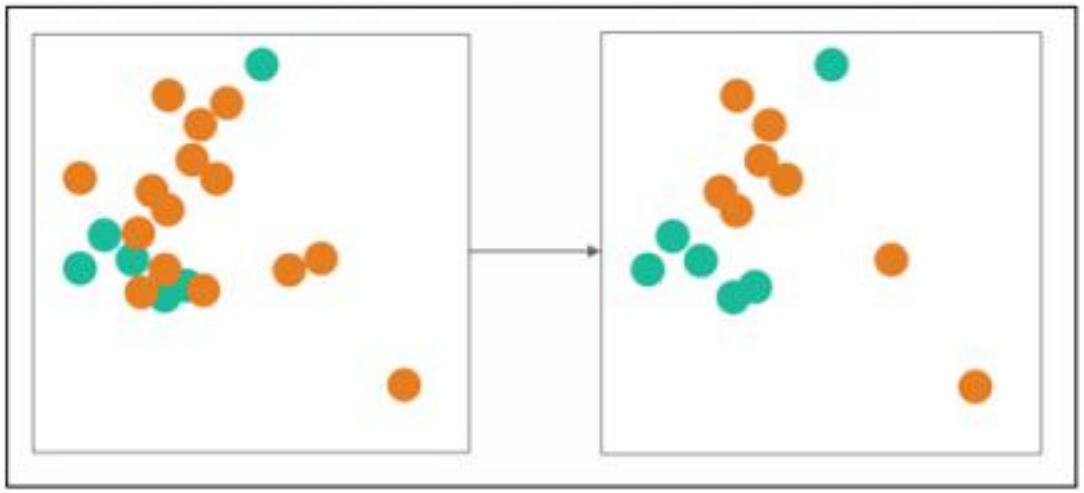


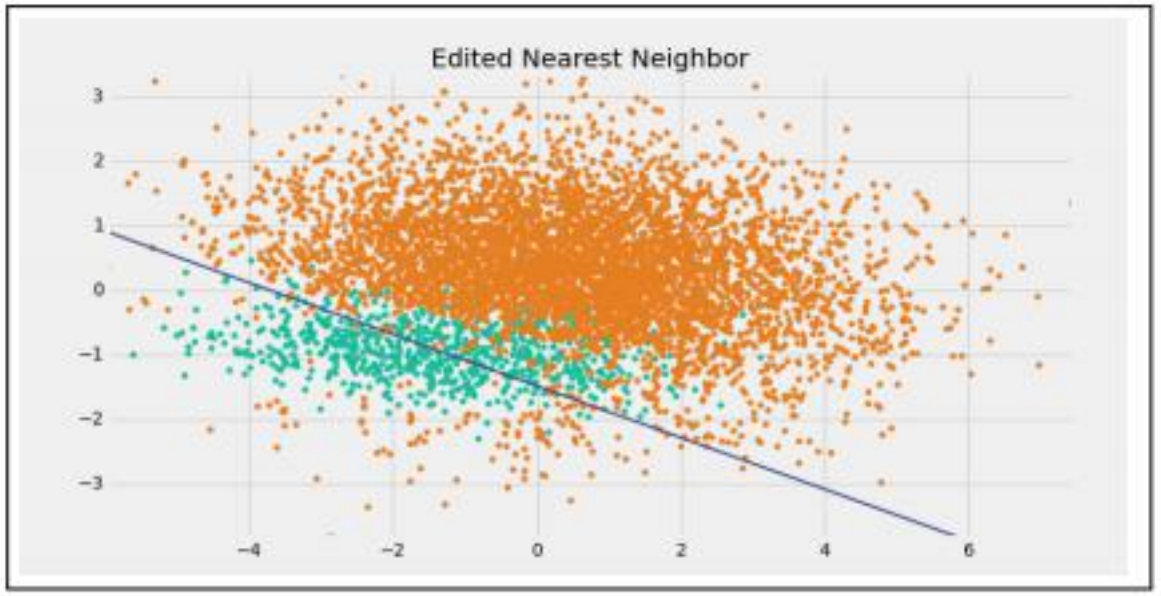
数据不平衡解决方案二

- 集成下采样/欠采样：采用普通的下采样方式会导致信息丢失，所以一般采用集成学习和下采样结合的方式来解决这个问题；主要有两种方式：
  - EasyEnsemble
    - 采用不放回的数据抽取方式抽取多数类别样本数据，然后将抽取出来的数据和少数类别数据组合训练一个模型；多次进行这样的操作，从而构建多个模型，然后使用多个模型共同决策/预测。
  - BalanceCascade
    - 利用Boosting这种增量思想来训练模型；先通过下采样产生训练集，然后使用Adaboost算法训练一个分类器；然后使用该分类器多对所有的大众样本数据进行预测，并将预测正确的样本从大众样本数据中删除；重复迭代上述两个操作，直到大众样本数据量等于小众样本数据量。

数据不平衡解决方案三

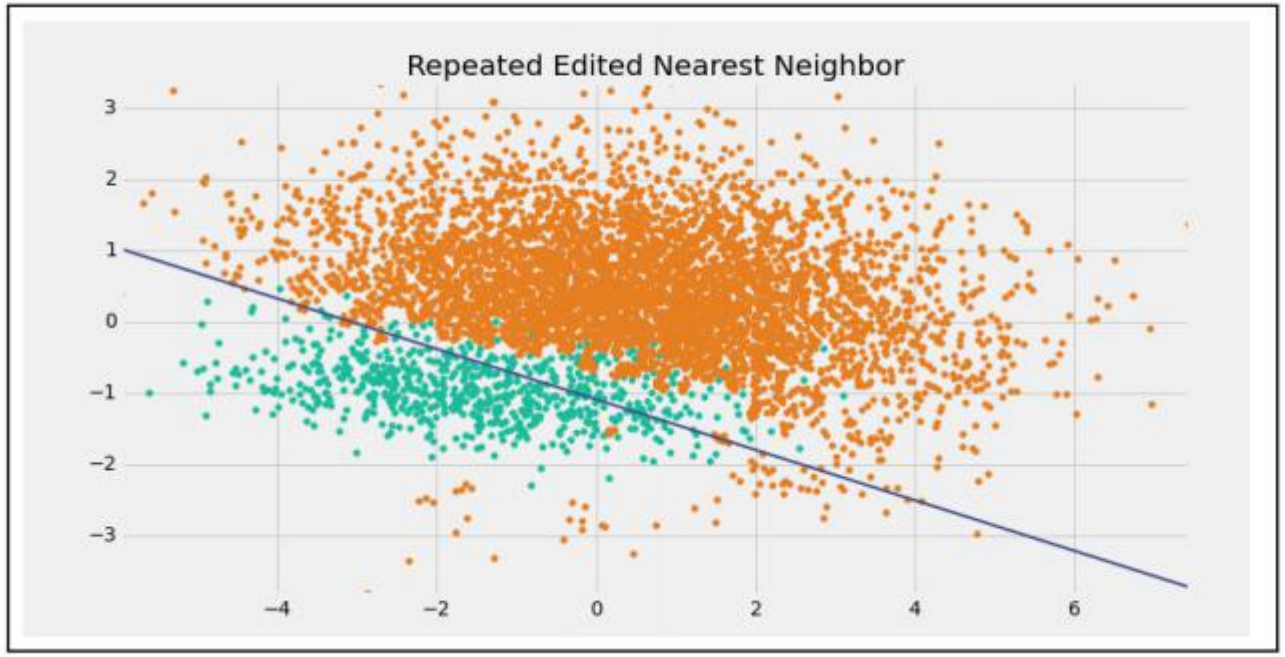
- Edited Nearest Neighbor(ENN)：对于多数类别样本数据而言，如果这个样本的大部分k近邻样本都和自身类别不一样，那我们就将其删除，然后使用删除后的数据训练模型。





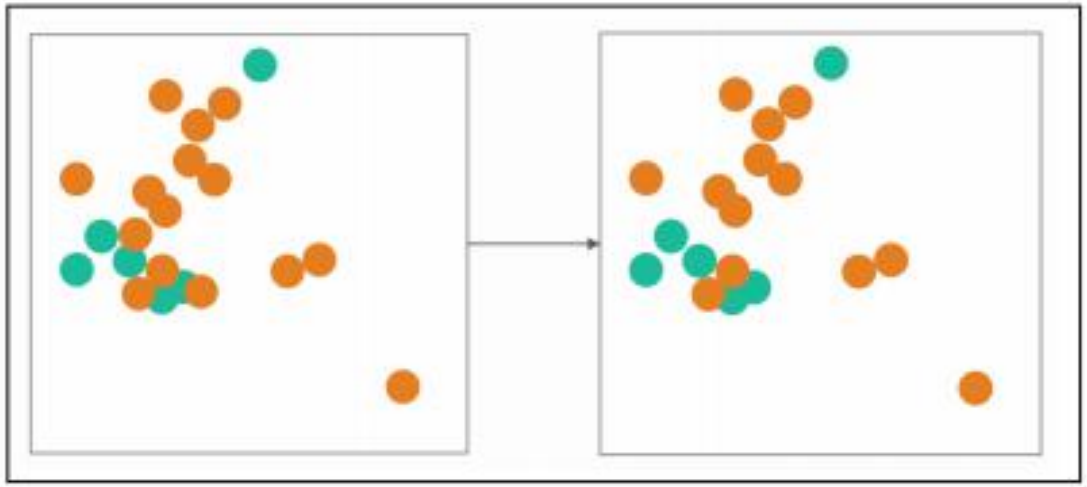
数据不平衡解决方案四

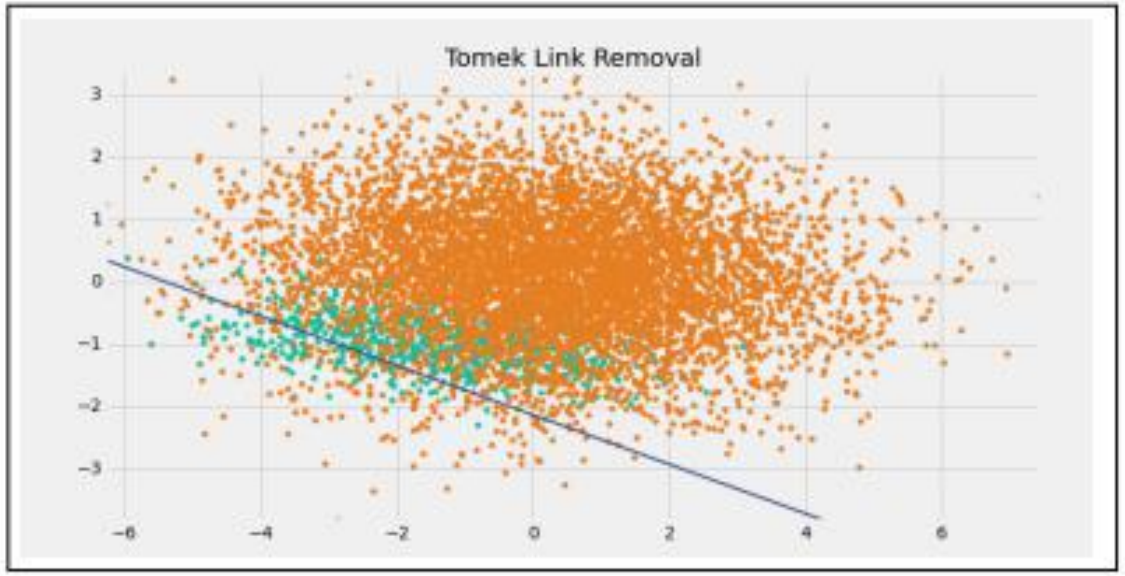
- Repeated Edited Nearest Neighbor(RENN)：对于多数类别样本数据而言，如果这个样本的大部分k近邻样本都和自身类别不一样，那我们就将其删除；重复性的进行上述的删除操作，直到数据集无法再被删除后，使用此时的数据集据训练模型。



数据不平衡解决方案五

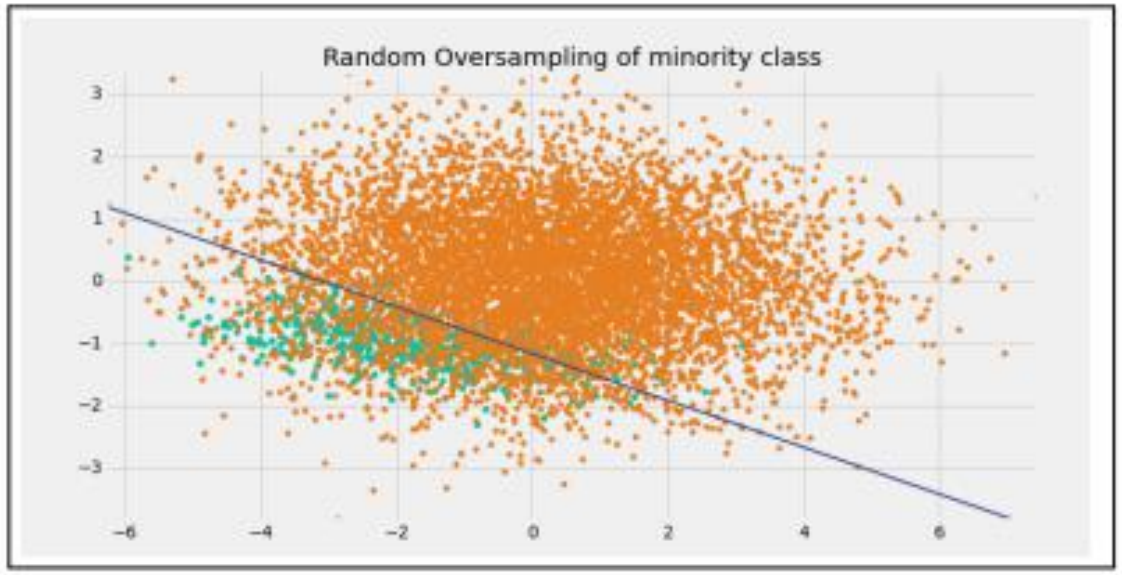
- Tomek Link Removal：如果两个不同类别的样本，它们的最近邻都是对方，也就是A的最近邻是B，B的最近邻也是A，那么A、B就是Tomek Link。将所有Tomek Link中多数类别的样本删除。然后使用删除后的样本来训练模型。





数据不平衡解决方案六

- 过采样/上采样(Over Sampling)：和欠采样采用同样的原理，通过抽样来增加少数样本的数目，从而达到数据平衡的目的。一种简单的方式就是通过有放回抽样，不断的从少数类别样本数据中抽取样本，然后使用抽取样本+原始数据组成训练数据集来训练模型；不过该方式比较容易导致过拟合，一般抽样样本不要超过50%。

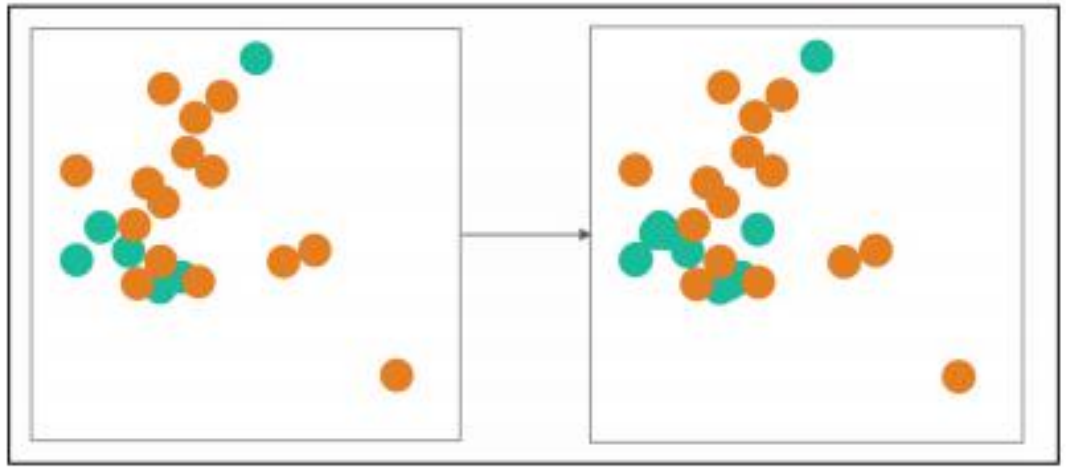


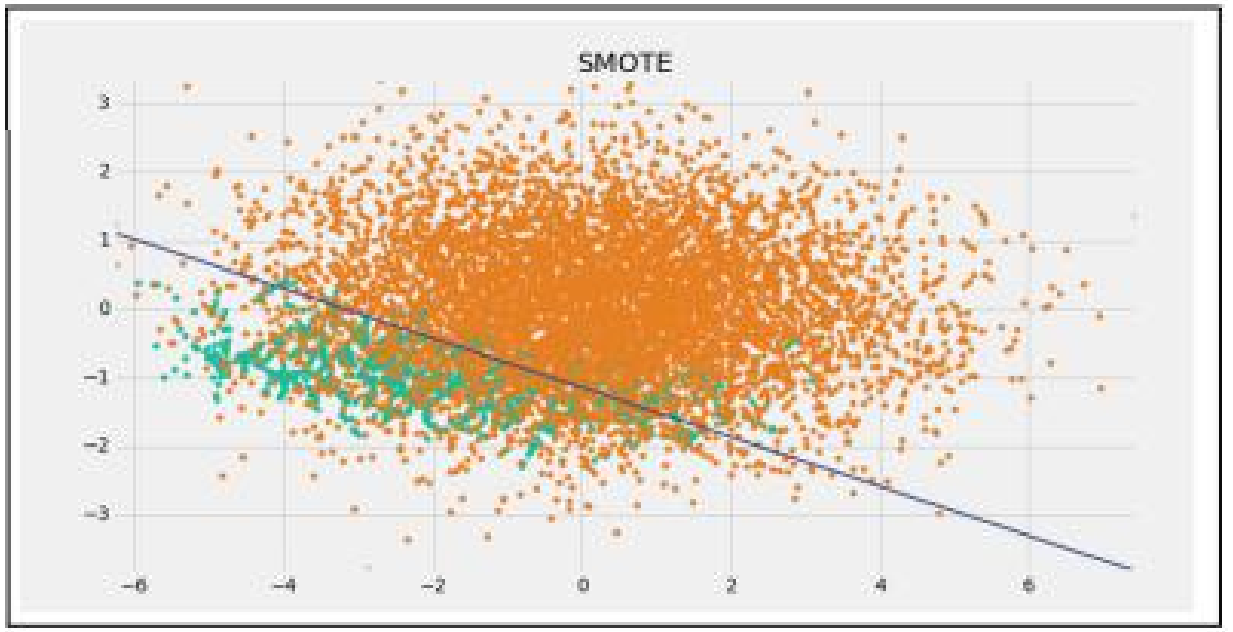
数据不平衡解决方案六

- 过采样/上采样(Over Sampling)：因为在上采样过程中，是进行是随机有放回的抽样，所以最终模型中，数据其实是相当于存在一定的重复数据，为了防止这个重复数据导致的问题，我们可以加入一定的随机性，也就是说：在抽取数据后，对数据的各个维度可以进行随机的小范围变动，eg: (1,2,3) --> (1.01, 1.99, 3)；通过该方式可以相对比较容易的降低上采样导致的过拟合问题。

数据不平衡解决方案七

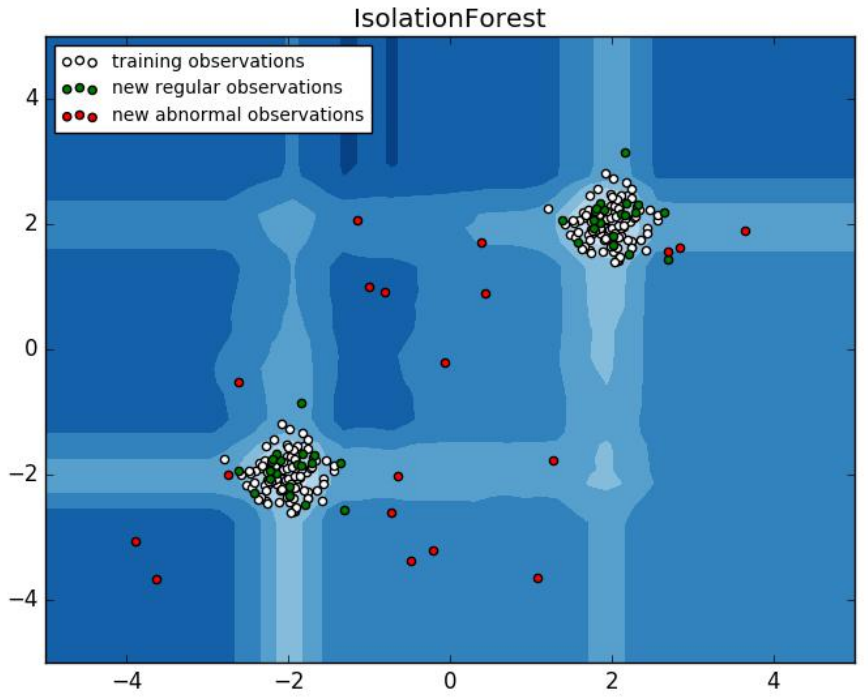
- 采用数据合成的方式生成更多的样本，该方式在小数据集场景下具有比较成功的案例。常见算法是SMOTE算法，该算法利用小众样本在特征空间的相似性来生成新样本。





数据不平衡解决方案八

- 对于正负样本极不平衡的情况下，其实可以换一种思路/角度来看待这个问题：可以将其看成一分类(One Class Learning)或者异常检测(Novelty Detection)问题，在这类算法应用中主要就是对于其中一个类别进行建模，然后对所有不属于这个类别特征的数据就认为是异常数据，经典算法包括：One Class SVM、IsolationForest等。




北风网
IBEIFENG.COM