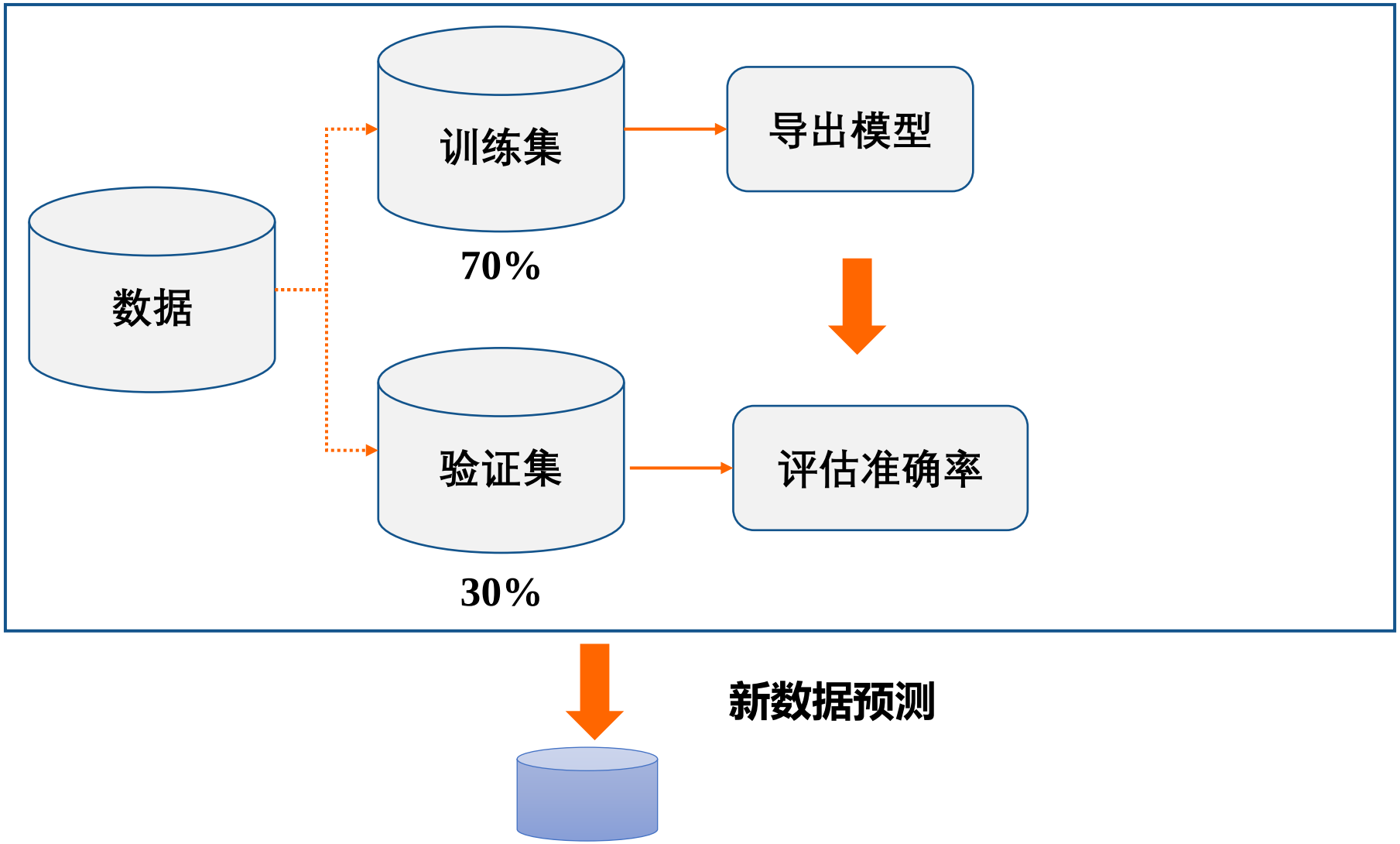
数据挖掘

模型的评价

2024 年 12 月

基本思路

分类问题： 数据预处理→模型训练→模型调整→对新数据分类→模型评价



评价目的

- 对于解决同一问题的不同模型，通过比较模型指标来比较模型之间的优劣，选取最优模型

- 对于同一模型，通过比较模型指标来调整模型参数

南京大学
NANJING UNIVERSITY

目录

准确率评价

混淆矩阵

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

准确率（Accuracy）$= \frac{a+d}{a+b+c+d} = \frac{TP+TN}{TP+TN+FP+FN}$

例子

- 考虑一个二分类问题
  - 0类的实例数 = 9990
  - 1类的实例数 = 10

- 如果模型预测每个实例为0类, 则准确率为 **99.9%**
  - 准确率是误导
  - 模型不能正确预测任何1类实例
  - 而在**疾病检测、地震预报**等问题中，1类更需要被关心

其它度量

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

- 真阳率TP，真阳性（True positive rate, TPR）或灵敏度（sensitivity）、查全率（recall）
  - TPR = TP/(TP + FN)
- 真阴率TN，真阴性（True negative rate, TNR）或特指度（specificity）
  - TNR = TN/(TN + FP)
- 假阳率FP，假阳性（False positive rate, FPR）或 误报率
  - FPR = FP/(TN + FP)
- 假阴率FN，假阴性（False negative rate, FNR）漏报率（与查全率此消彼长）
  - FNR = FN/(TP + FN)

其它度量

- 两个广泛使用的度量
  - 召回率（查全率，recall）和精确率（查准率，precision）

$$recall = \frac{TP}{TP + FN}$$

$$precision = \frac{TP}{TP + FP}$$

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

例子

- 假设我们手上有60个正样本，40个负样本，我们要找出所有的正样本，系统查找出50个，其中只有40个是真正的正样本，计算上述各指标。

  - TP: 将正类预测为正类数：40
  - FN: 将正类预测为负类数：20（60-40，剩余没正确分类的正样本）
  - FP: 将负类预测为正类数：10
  - TN: 将负类预测为负类数：30

  - 准确率(accuracy) = 预测对的/所有 = (TP+TN)/(TP+FN+FP+TN) = 70%
  - 精确率(precision) = TP/(TP+FP) = 80%
  - 召回率(recall) = TP/(TP+FN) = 66.7%

查全率vs.查准率

下面是两个场景：

- **地震的预测:** 对于地震的预测，我们希望的是recall非常高，也就是说每次地震我们都希望预测出来。这个时候我们可以牺牲precision。情愿发出1000次警报，把10次地震都预测正确了，也不要预测100次，对了8次，漏了2次。
- **嫌疑人定罪:** 基于不错怪一个好人的原则（无罪推定原则，presumption of innocence），对于嫌疑人的定罪我们希望是非常准确的（precision高），及时有时候放过了一些罪犯（recall低），但也是值得的。

$$F_1 = \frac{2rp}{r+p} = \frac{2 \times TP}{2 \times TP + FP + FN}$$

ROC曲线

- 前面分类器性能评价的局限性：分类器预测结果为离散的1或者0
- 朴素贝叶斯输出？ $p(x|y)$=?
- 其他分类器输出？Softmax,…

A\P	C	¬C	
C	TP	FN	P
¬C	FP	TN	N
	P’	N’	All

ROC曲线

- 前面分类器性能评价的局限性：分类器预测结果为离散的1或者0
- 朴素贝叶斯输出？ $p(y|x)$=?
- 其他分类器输出？ Softmax,…

输出是一个连续的概率值

A\P	C	¬C	
C	TP	FN	P
¬C	FP	TN	N
	P’	N’	All

Instance	P(+|A)
1	0.95
2	0.93
3	0.87
4	0.85
5	0.85
6	0.85
7	0.76
8	0.53
9	0.43
10	0.25

ROC曲线

- 前面分类器性能评价的局限性：分类器预测结果为离散的1或者0
- 朴素贝叶斯输出？ $p(x|y)$=?
- 其他分类器输出？Softmax,…
- 解决方法：连续的值离散化
- 导致的问题：离散阈值难以确定

输出是一个连续的概率值

A\P	C	¬C	
C	TP	FN	P
¬C	FP	TN	N
	P’	N’	All

Instance	P(+|A)
1	0.95
2	0.93
3	0.87
4	0.85
5	0.85
6	0.85
7	0.76
8	0.53
9	0.43
10	0.25

ROC曲线

- 接收者操作特征曲线（Receiver Operating Characteristic Curve，或者叫ROC曲线）是一种坐标图式的分析工具，用于
  - 选择最佳的分类模型、舍弃次佳的模型。
  - 在同一模型中设定最佳阈值。
- 给定一个二元分类模型和它的阈值，就能从所有样本的(阳性／阴性)真实值和预测值计算出一个 (X=FPR, Y=TPR) 坐标点。

A\P	C	¬C	
C	TP	FN	P
¬C	FP	TN	N
	P’	N’	All

$$TPR = TP/(TP + FN)$$

$$FPR = FP/(TN + FP)$$

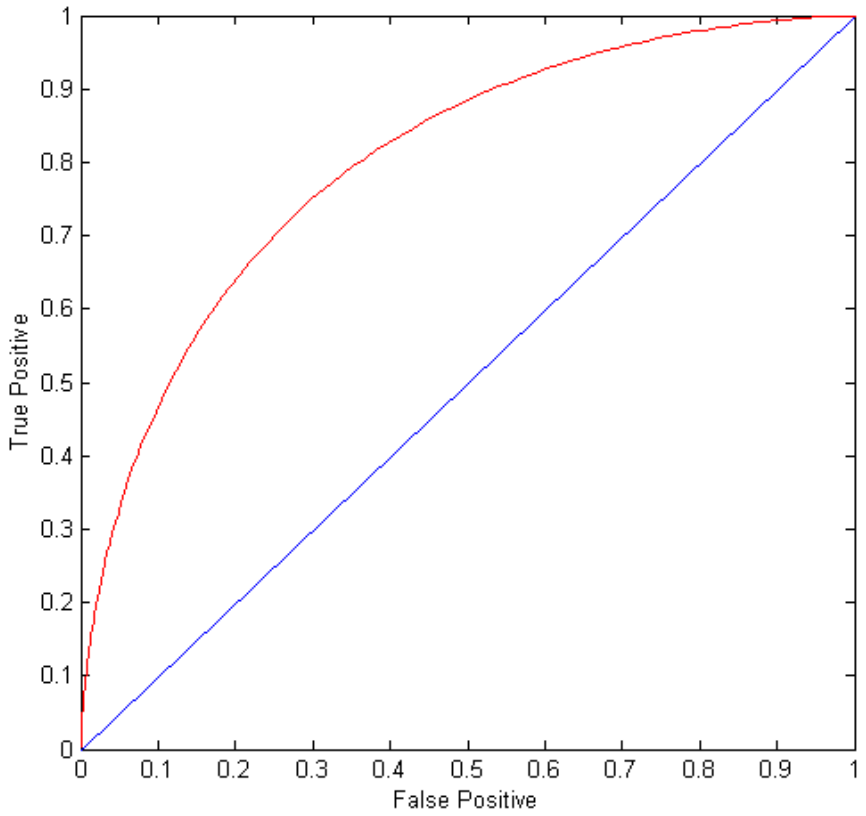
$$TNR = TN/(TN + FP)$$

$$FNR = FN/(TP + FN)$$

ROC曲线属性

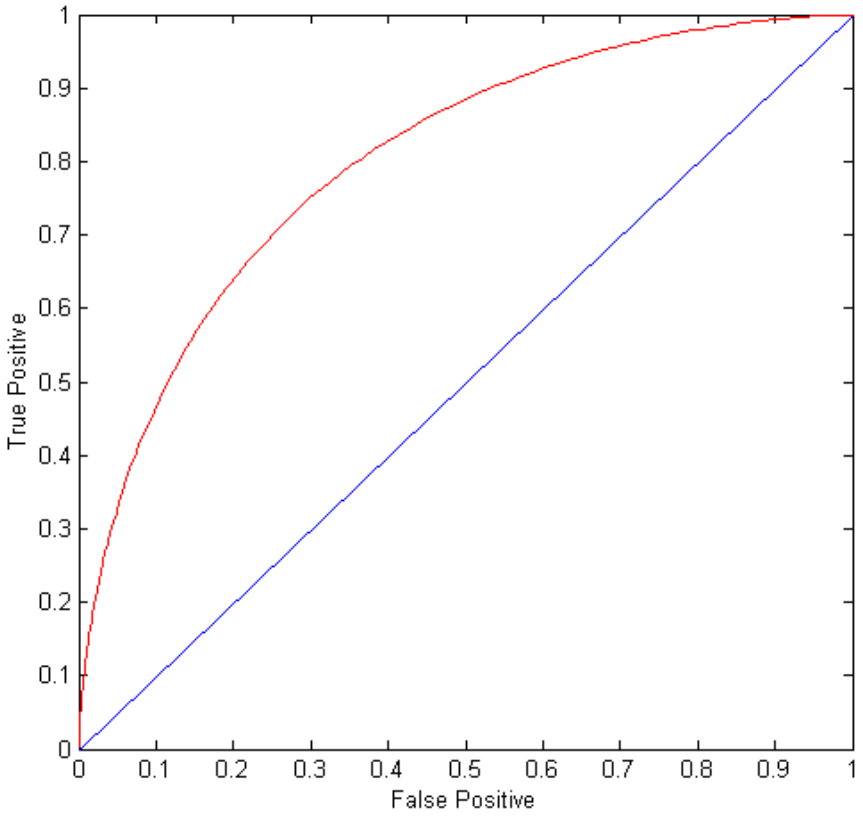
(FPR, TPR):

- (0,0): 任何分类都是阴性
- (1,1): 任何分类都是阳性
- (0,1): 理想分类
- 对角线:
  - 随机猜测结果
  - 对角线以下：预测结果与真实结果相反



ROC曲线下方面积：AUC

- ROC曲线下方的区域称为AUC，Area Under the ROC curve
  - Ideal:
    - Area = 1
  - Random guess:
    - Area = 0.5



$$TPR = TP/(TP + FN)$$

$$FPR = FP/(TN + FP)$$

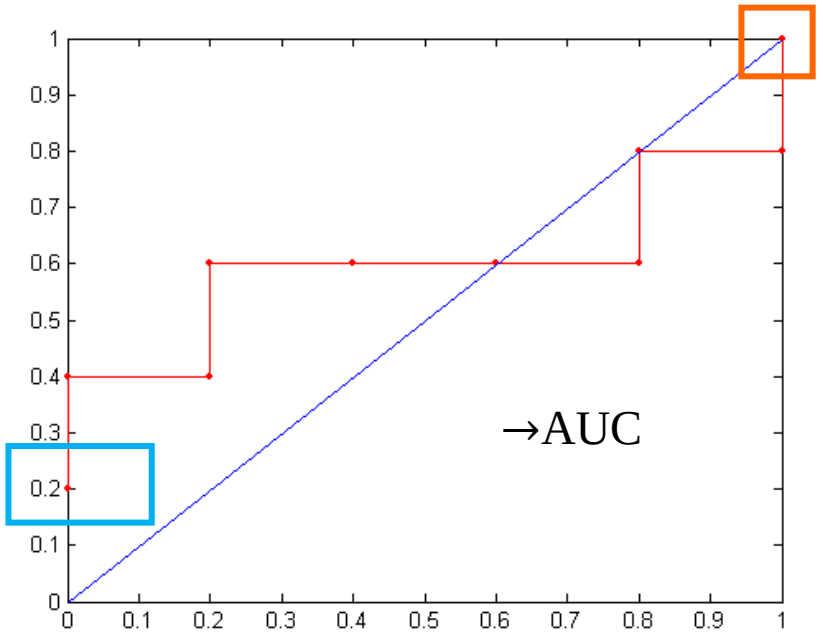
如何构建ROC曲线

- 首先利用分类器计算每个数据记录的后验概率P(+|A)
- 将这些数据记录对应的P(+|A)从高到低排列（如右表）：
  - 由低到高，对于每个P(+|A)值（threshold，阈值），把对应的记录以及那些值高于或等于阈值指派为阳性类positive，把那些值低于阈值指派为阴性类negative
  - 统计 TP, FP, TN, FN
  - 计算TPR = TP/(TP+FN)和FPR = FP/(FP + TN)
- 绘出诸点(FPR, TPR)并连接它们

Instance	P(+|A)	True Class
1	0.95	+
2	0.93	+
3	0.87	–
4	0.85	–
5	0.85	–
6	0.85	+
7	0.76	–
8	0.53	+
9	0.43	–
10	0.25	+

例子

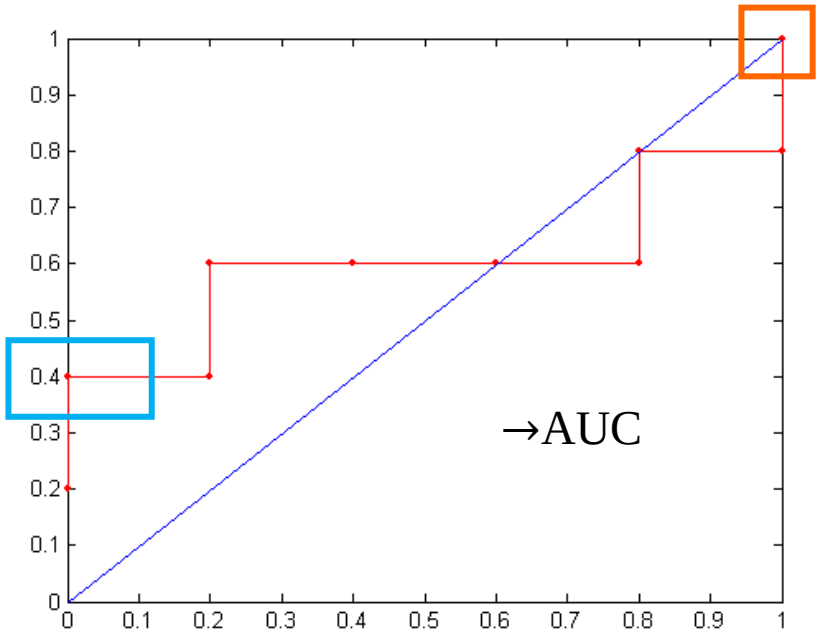
	Class	+	-	+	-	-	-	+	-	+	+	
	Threshold >=	0.25	0.43	0.53	0.76	0.85	0.85	0.85	0.87	0.93	0.95	0.95
	TP	5	4	4	3	3	3	3	2	2	1	0
	FP	5	5	4	4	3	2	1	1	0	0	0
	TN	0	0	1	1	2	3	4	4	5	5	5
	FN	0	1	1	2	2	2	2	3	3	4	5
→	TPR	1	0.8	0.8	0.6	0.6	0.6	0.6	0.4	0.4	0.2	0
→	FPR	1	1	0.8	0.8	0.6	0.4	0.2	0.2	0	0	0



Instance	P(+|A)	True Class
1	0.95	+
2	0.93	+
3	0.87	–
4	0.85	–
5	0.85	–
6	0.85	+
7	0.76	–
8	0.53	+
9	0.43	–
10	0.25	+

例子

	Class	+	-	+	-	-	-	+	-	+	+	
Threshold >=		0.25	0.43	0.53	0.76	0.85	0.85	0.85	0.87	0.93	0.95	0.95
	TP	5	4	4	3	3	3	3	2	2	1	0
	FP	5	5	4	4	3	2	1	1	0	0	0
	TN	0	0	1	1	2	3	4	4	5	5	5
	FN	0	1	1	2	2	2	2	3	3	4	5
→	TPR	1	0.8	0.8	0.6	0.6	0.6	0.6	0.4	0.4	0.2	0
→	FPR	1	1	0.8	0.8	0.6	0.4	0.2	0.2	0	0	0



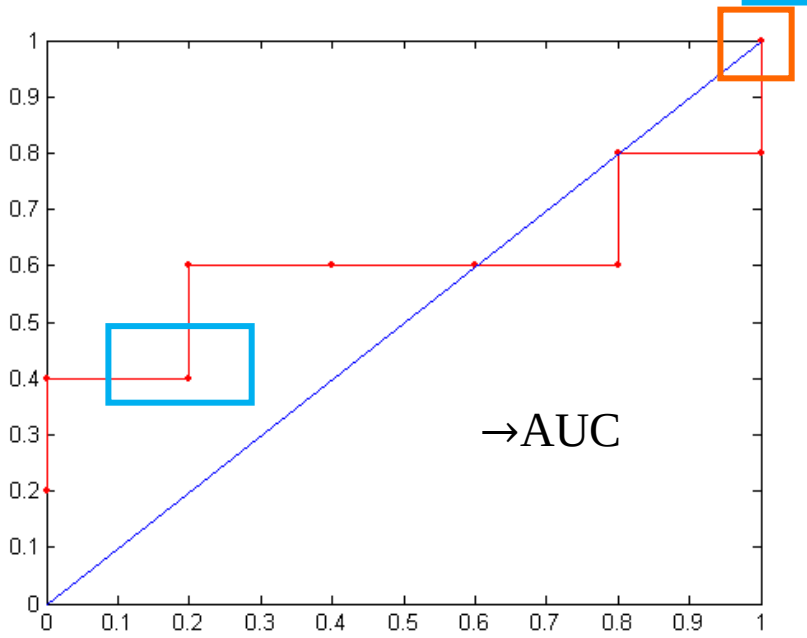
Instance	P(+|A)	True Class
1	0.95	+
2	0.93	+
3	0.87	–
4	0.85	–
5	0.85	–
6	0.85	+
7	0.76	–
8	0.53	+
9	0.43	–
10	0.25	+

例子

	Class	+	-	+	-	-	-	+	-	+	+	
Threshold >=		0.25	0.43	0.53	0.76	0.85	0.85	0.85	0.87	0.93	0.95	0.95
	TP	5	4	4	3	3	3	3	2	2	1	0
	FP	5	5	4	4	3	2	1	1	0	0	0
	TN	0	0	1	1	2	3	4	4	5	5	5
	FN	0	1	1	2	2	2	2	3	3	4	5
→	TPR	1	0.8	0.8	0.6	0.6	0.6	0.6	0.4	0.4	0.2	0
→	FPR	1	1	0.8	0.8	0.6	0.4	0.2	0.2	0	0	0

A= [填空1]

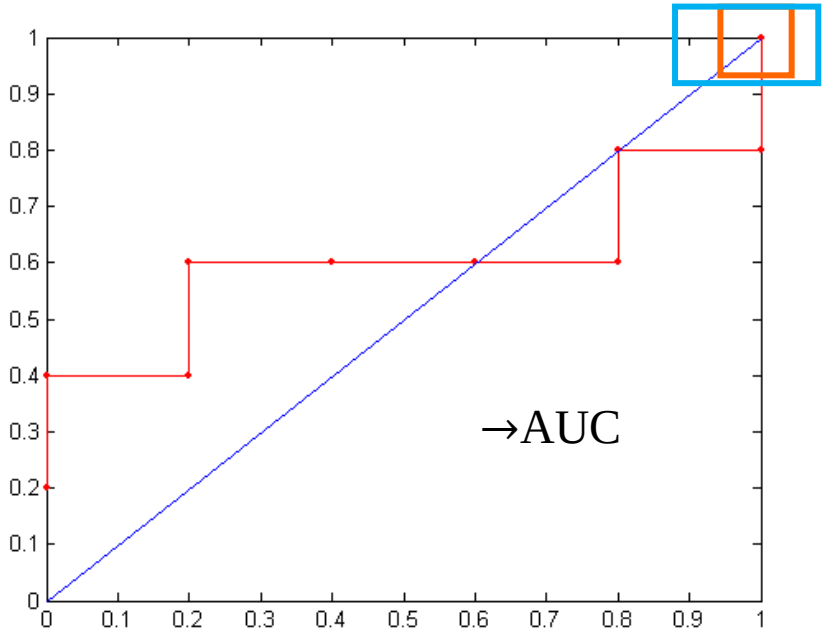
B= [填空2]



Instance	P(+|A)	True Class
1	0.95	+
2	0.93	+
3	0.87	–
4	0.85	–
5	0.85	–
6	0.85	+
7	0.76	–
8	0.53	+
9	0.43	–
10	0.25	+

如何构建ROC曲线

	Class	+	-	+	-	-	-	+	-	+	+	
Threshold >=		0.25	0.43	0.53	0.76	0.85	0.85	0.85	0.87	0.93	0.95	0.95
	TP	5	4	4	3	3	3	3	2	2	1	0
	FP	5	5	4	4	3	2	1	1	0	0	0
	TN	0	0	1	1	2	3	4	4	5	5	5
	FN	0	1	1	2	2	2	2	3	3	4	5
→	TPR	1	0.8	0.8	0.6	0.6	0.6	0.6	0.4	0.4	0.2	0
→	FPR	1	1	0.8	0.8	0.6	0.4	0.2	0.2	0	0	0

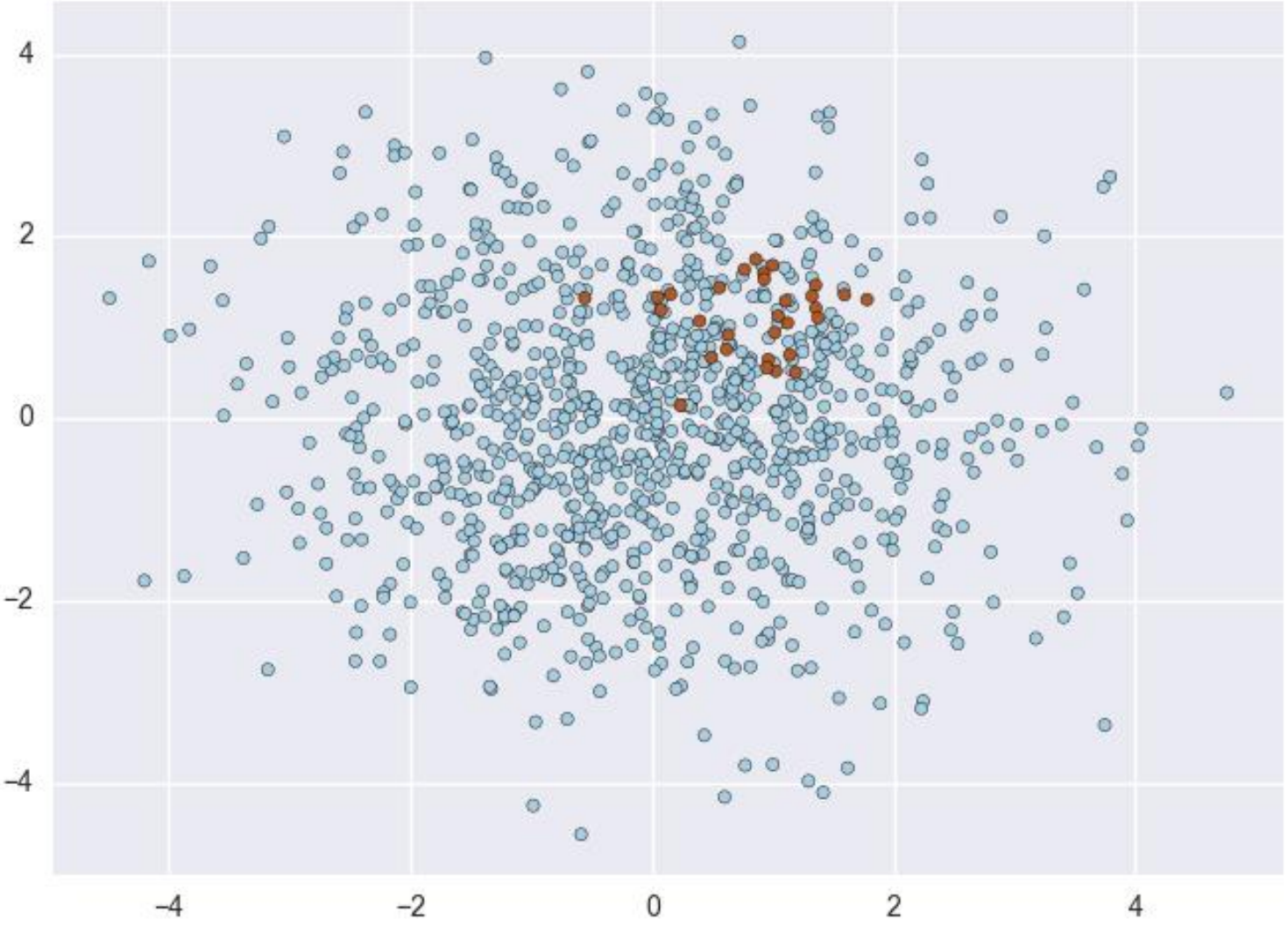


Instance	P(+|A)	True Class
1	0.95	+
2	0.93	+
3	0.87	–
4	0.85	–
5	0.85	–
6	0.85	+
7	0.76	–
8	0.53	+
9	0.43	–
10	0.25	+

南京大学
NANJING UNIVERSITY

目录

数据不平衡问题

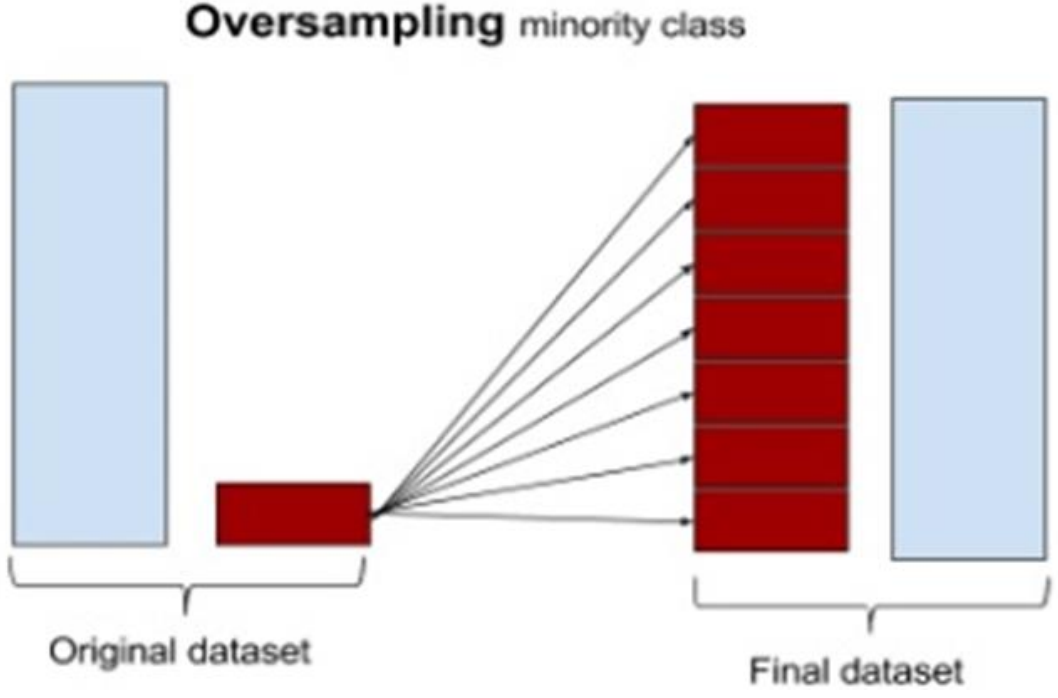


基于抽样的方法

- 基于抽样的方法
  - 考虑一个包含100个正样本和1000个负样本的数据集
  - Oversampling 过采样
    - 复制正样本, 直到训练集中正样本和负样本一样多
    - 可能导致模型过分拟合，因为一些噪声样本也可能被复制多次
  - Undersampling欠采样
    - 随机抽取100个负样本, 与所有的正样本一起形成训练集
    - 问题: 一些有用的负样本可能没有选出来用于训练，因此导致一个不太优的模型
    - 解决问题的方法：多次执行不充分抽样, 并归纳类似于集成学习方法的多分类器
  - Oversampling + Undersampling

基于抽样的方法

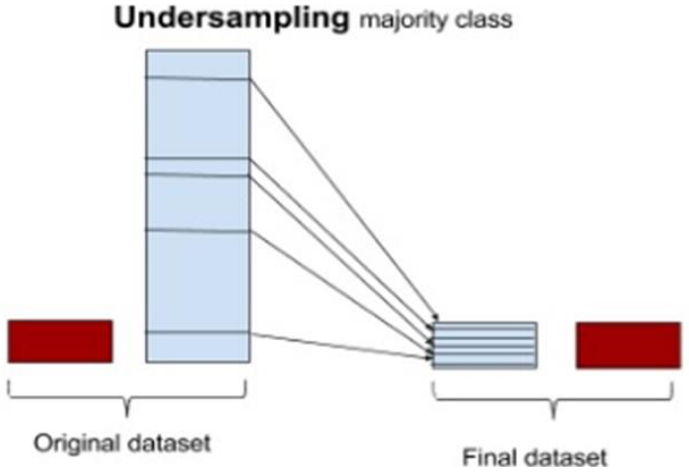
- 基于抽样的方法
  - Oversampling 过采样
    - 复制正样本, 直到训练集中正样本和负样本一样多
    - 可能导致模型过分拟合，因为一些噪声样本也可能被复制多次



噪声样本也可能被复制多次

基于抽样的方法

- 基于抽样的方法
  - Undersampling欠采样
    - 随机抽取100个负样本, 与所有的正样本一起形成训练集
    - 问题: 一些有用的负样本可能没有选出来用于训练，因此导致一个不太优的模型
    - 解决问题的方法：多次执行不充分抽样, 并归纳类似于组合学习方法的多分类器



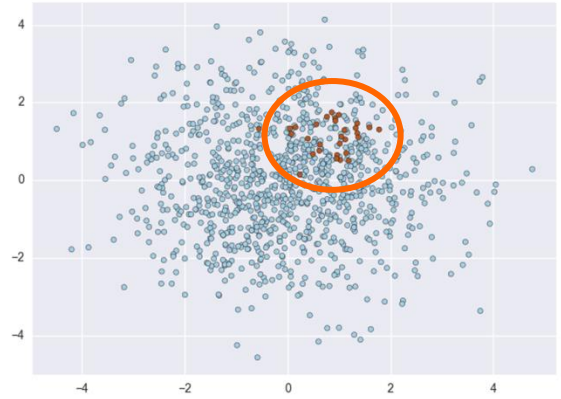
有用的负样本可能没有选出来用于训练

基于抽样的方法

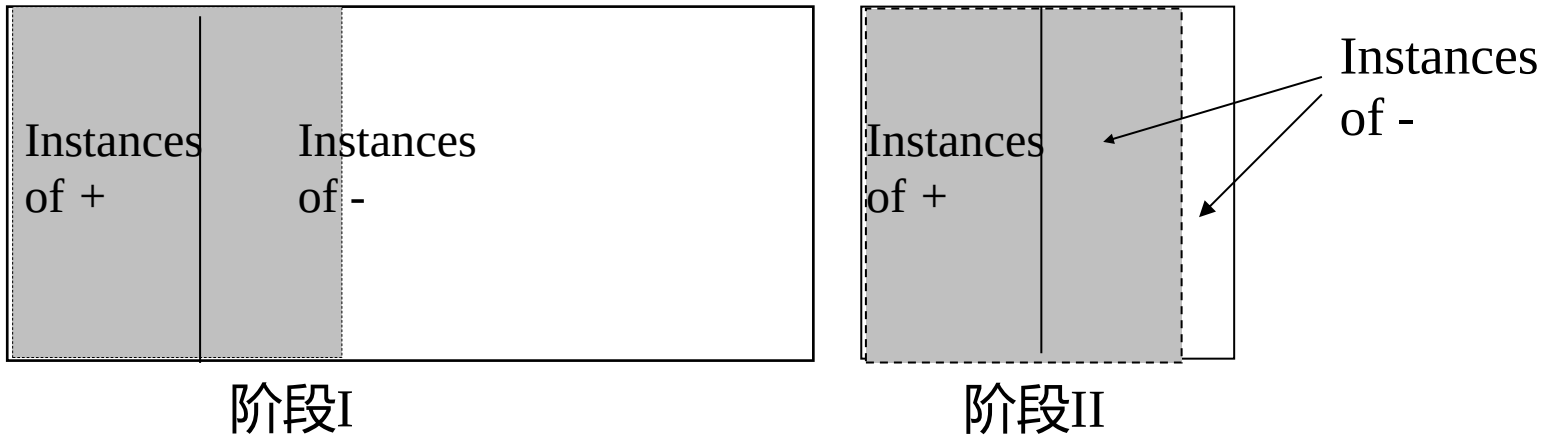
- 基于抽样的方法
  - 考虑一个包含100个正样本和1000个负样本的数据集
  - Oversampling 过采样
    - 复制正样本, 直到训练集中正样本和负样本一样多
    - 可能导致模型过分拟合，因为一些噪声样本也可能被复制多次
  - Undersampling欠采样
    - 随机抽取100个负样本, 与所有的正样本一起形成训练集
    - 问题: 一些有用的负样本可能没有选出来用于训练，因此导致一个不太优的模型
    - 解决问题的方法：多次执行不充分抽样, 并归纳类似于组合学习方法的多分类器
  - Oversampling + Undersampling

两阶段学习

- 两阶段学习：PN-Rules
  - 是基于规则的分类
  - 学习分两个阶段，每个阶段学习一组规则

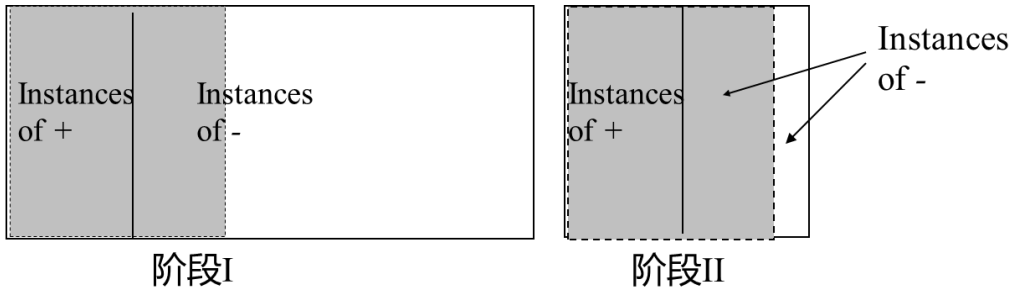


- 训练
  - 阶段I：学习一组规则，尽可能覆盖正类（少的那一类）
  - 阶段II：使用阶段I覆盖的正类和负类样本+部分其它负类样本，学习一组规则



两阶段学习

- 分类
  - 用第一组规则对x分类，如果分到负类，则x属于负类
  - 否则，用第二组规则确定x所属的类
- R. Agarwal, and M. V. Joshi. PNrule: A New Framework for Learning Classifier Models in Data Mining (A Case-Study in Network Intrusion Detection). In Proc. of the First SIAM Conference on Data Mining. Chicago, USA, April 2001



南京大学
NANJING UNIVERSITY

目录

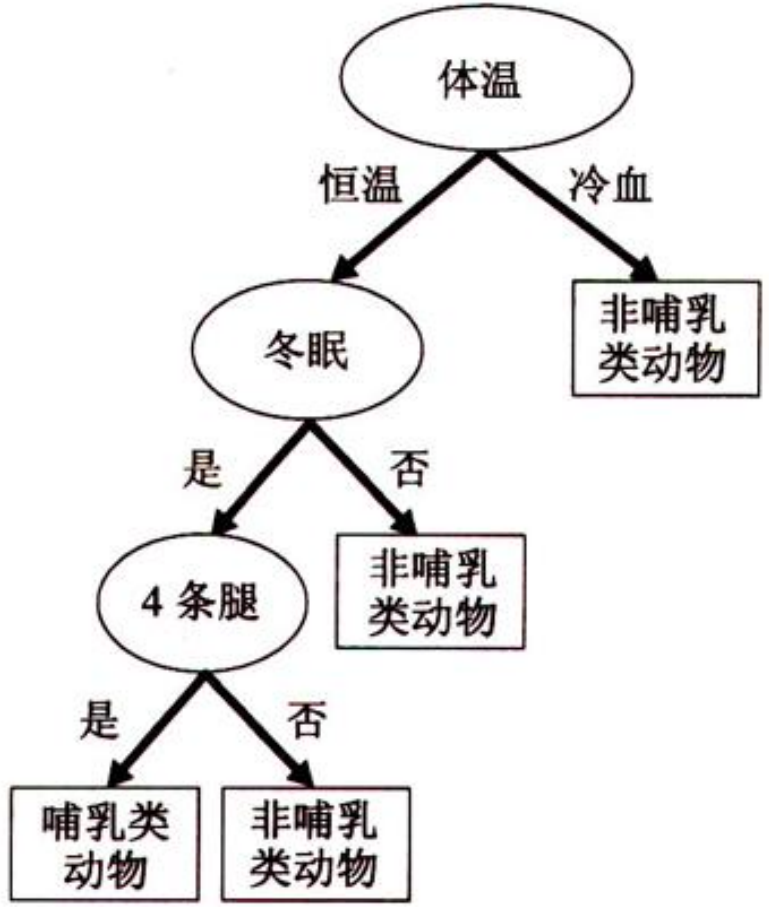
模型过分拟合和拟合不足

- 分类模型的误差大致分为两种：
  - 训练误差：是在训练记录上误分类样本比例
  - 泛化误差：是模型在未知记录上的期望误差
- 一个好的分类模型不仅要能够很好的拟合训练数据，而且对未知样本也要能准确分类。
- 换句话说，一个好的分类模型必须具有低训练误差和低泛化误差。
- 当训练数据拟合太好的模型（较低训练误差），其泛化误差可能比具有较高训练误差的模型高，这种情况成为模型过分拟合。

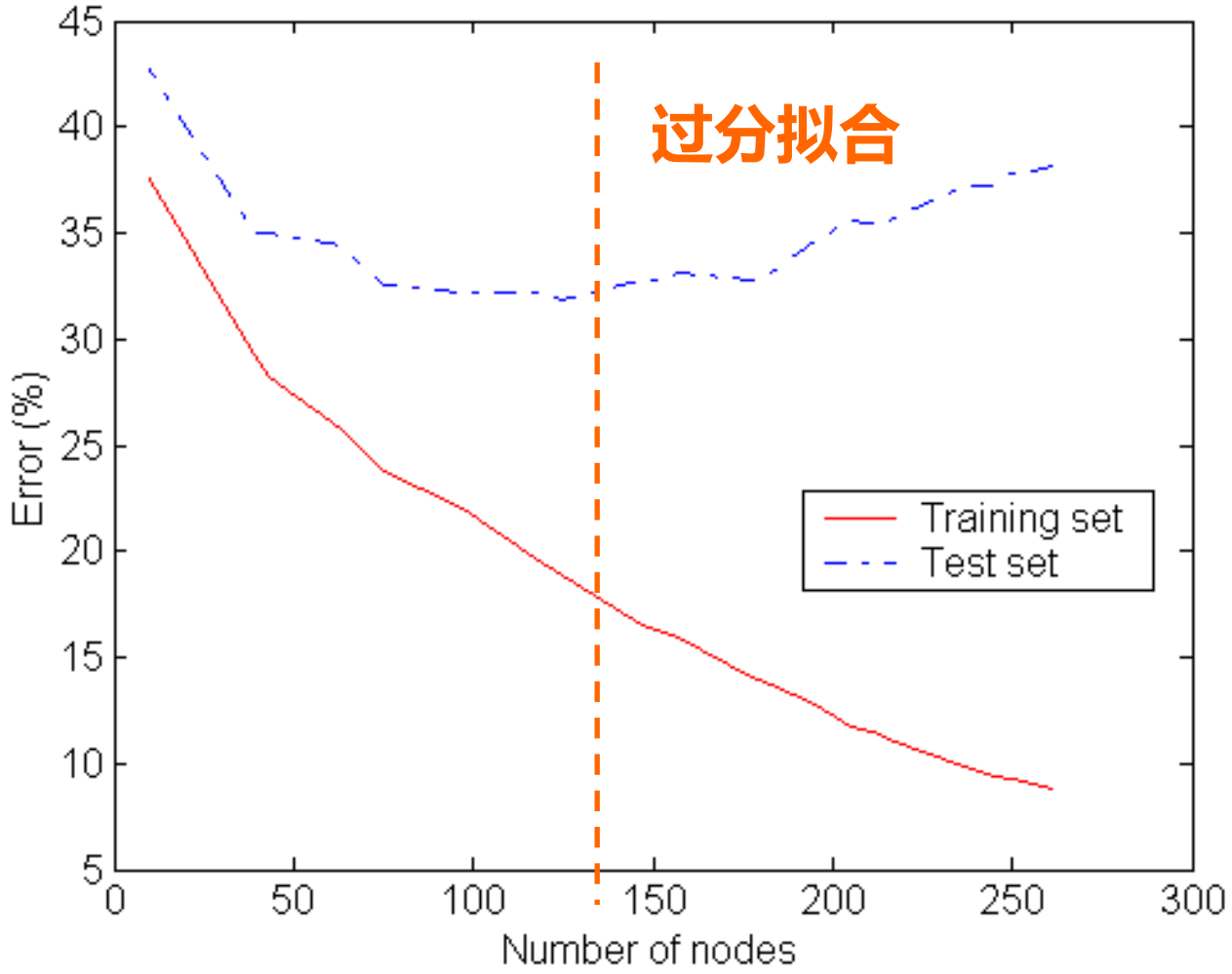
数据预处理→模型训练→模型调整→对新数据分类→模型评价

模型过分拟合和拟合不足

- 以决策树算法为例
  - 当决策树很小时，训练和检验误差都很大，这种情况称为模型拟合不足。出现拟合不足的原因是模型尚未学习到数据的真实结构。
  - 随着决策树中结点数的增加，模型的训练误差和泛化误差都会随之下降。
  - 当树的规模变得太大时，即使训练误差还在继续降低，但泛化误差开始增大，导致模型过分拟合。



模型过分拟合和拟合不足



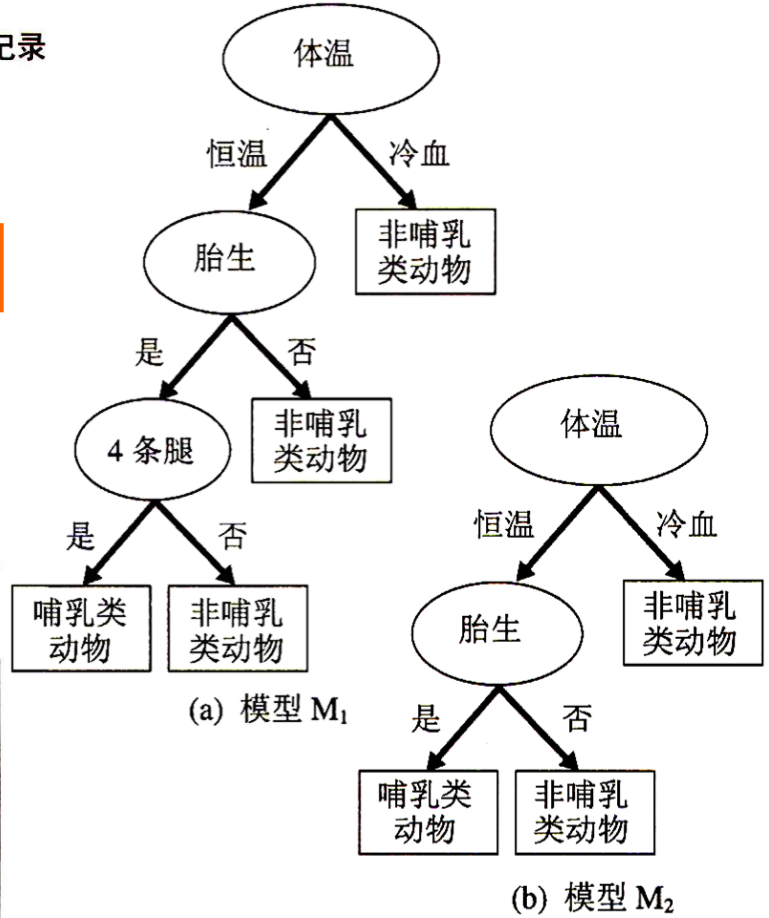
例子

表 4-3　哺乳类动物分类的训练数据集样本。打星号的类标号代表错误标记的记录

名称	体温	胎生	4 条腿	冬眠	类标号
豪猪	恒温	是	是	是	是
猫	恒温	是	是	否	是
蝙蝠	恒温	是	否	是	否*
鲸	恒温	是	否	否	否*
蝰蟒	冷血	否	是	是	否
科莫多巨蜥	冷血	否	是	否	否
蟒蛇	冷血	否	否	是	否
鲑鱼	冷血	否	否	否	否
鹰	恒温	否	否	否	否
虹鳉	冷血	是	否	否	否

表 4-4　哺乳类动物分类的检验数据集样本

名称	体温	胎生	4 条腿	冬眠	类标号
人	恒温	是	否	否	是
鸽子	恒温	否	否	否	否
象	恒温	是	是	否	是
豹纹鲨	冷血	是	否	否	否
海龟	冷血	否	是	否	否
企鹅	冷血	否	否	否	否
鳗	冷血	否	否	否	否
海豚	恒温	是	否	否	是
针鼹	恒温	否	是	是	是
希拉毒蜥	冷血	否	是	是	否



决策树M1的训练误差为 0，
但它在检验数据上的误差达 30%

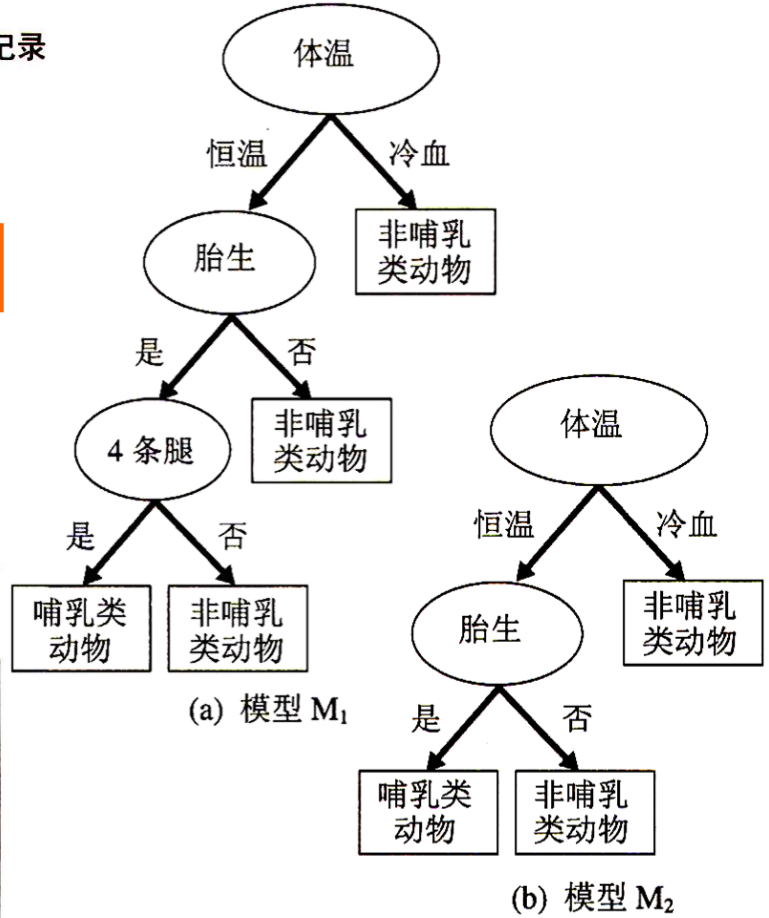
例子

表 4-3　哺乳类动物分类的训练数据集样本。打星号的类标号代表错误标记的记录

名称	体温	胎生	4 条腿	冬眠	类标号
豪猪	恒温	是	是	是	是
猫	恒温	是	是	否	是
蝙蝠	恒温	是	否	是	否*
鲸	恒温	是	否	否	否*
蝾螈	冷血	否	是	是	否
科莫多巨蜥	冷血	否	是	否	否
蟒蛇	冷血	否	否	是	否
鲑鱼	冷血	否	否	否	否
鹰	恒温	否	否	否	否
虹鳉	冷血	是	否	否	否

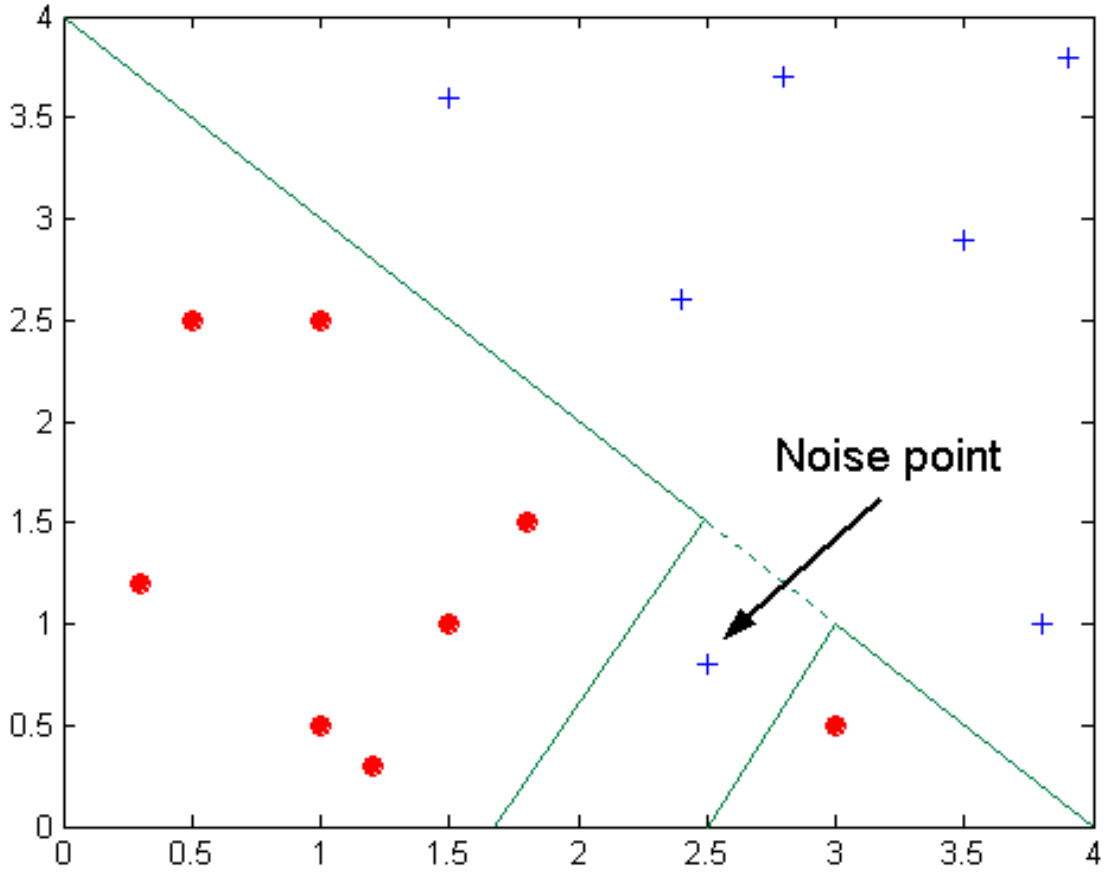
表 4-4　哺乳类动物分类的检验数据集样本

名称	体温	胎生	4 条腿	冬眠	类标号
人	恒温	是	否	否	是
鸽子	恒温	否	否	否	否
象	恒温	是	是	否	是
豹纹鲨	冷血	是	否	否	否
海龟	冷血	否	是	否	否
企鹅	冷血	否	否	否	否
鳗	冷血	否	否	否	否
海豚	恒温	是	否	否	是
针鼹	恒温	否	是	是	是
希拉毒蜥	冷血	否	是	是	否



决策树M2的训练误差为 20%，但它在检验数据上的误差达 10%

噪声导致的过分拟合



噪声导致决策边界的改变

缺乏代表性样本导致的过分拟合

- 根据**少量训练记录**做出分类决策的模型也容易受过分拟合的影响。
- 由于训练数据缺乏具有代表性的样本，在没有多少训练记录的情况下，**学习算法仍然细化模型**就会产生过分拟合。

表 4-5　哺乳动物分类的训练集样本

名称	体温	胎生	4 条腿	冬眠	类标号
蝾螈	冷血	否	是	是	否
虹鳉	冷血	是	否	否	否
鹰	恒温	否	否	否	否
弱夜鹰	恒温	否	否	是	否
鸭嘴兽	恒温	否	是	是	是

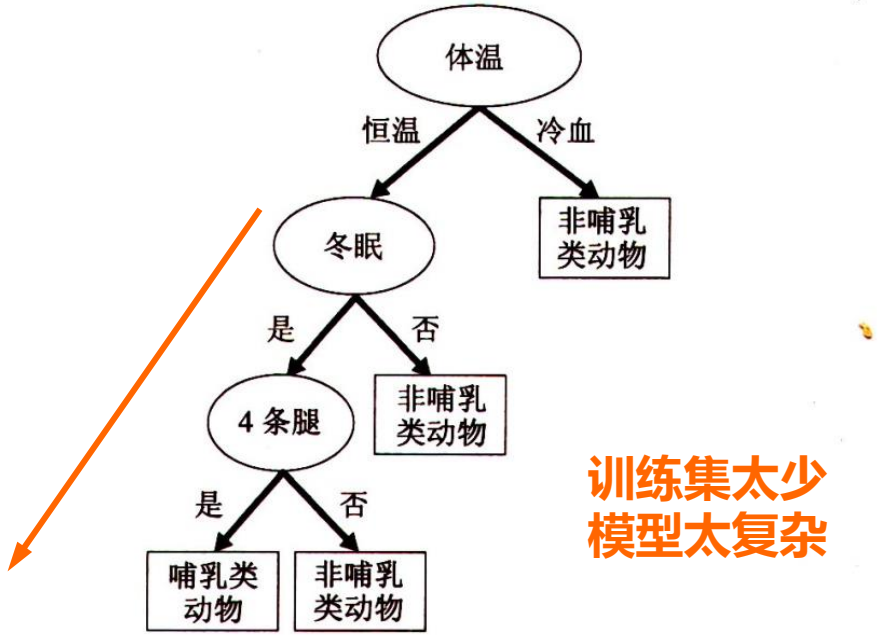
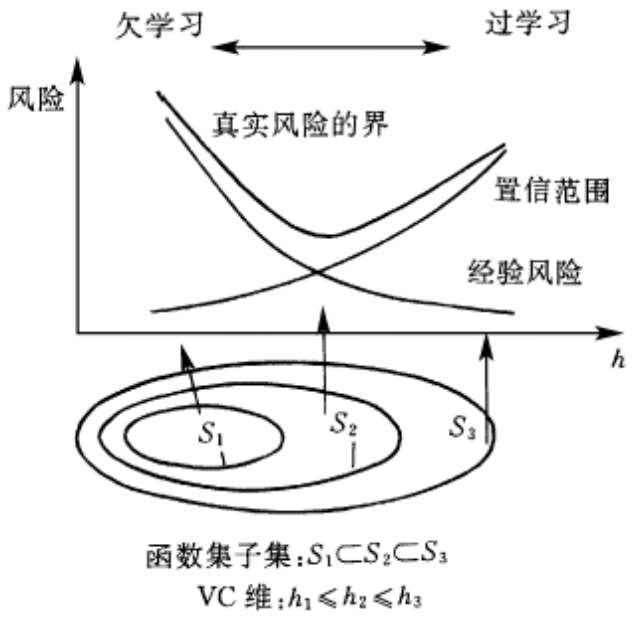


图 4-26　根据表 4-5 中的数据集建立的决策树

减少泛化误差

- 过分拟合的主要原因一直是个争辩的话题，但数据挖掘研究界普遍认为**模型的复杂度**对模型的过分拟合有影响。
- 如何确定正确的模型复杂度？理想的复杂度是能产生最低泛化误差的模型的复杂度。
- **奥卡姆剃刀定律**



函数集子集：$S_1 \subset S_2 \subset S_3$

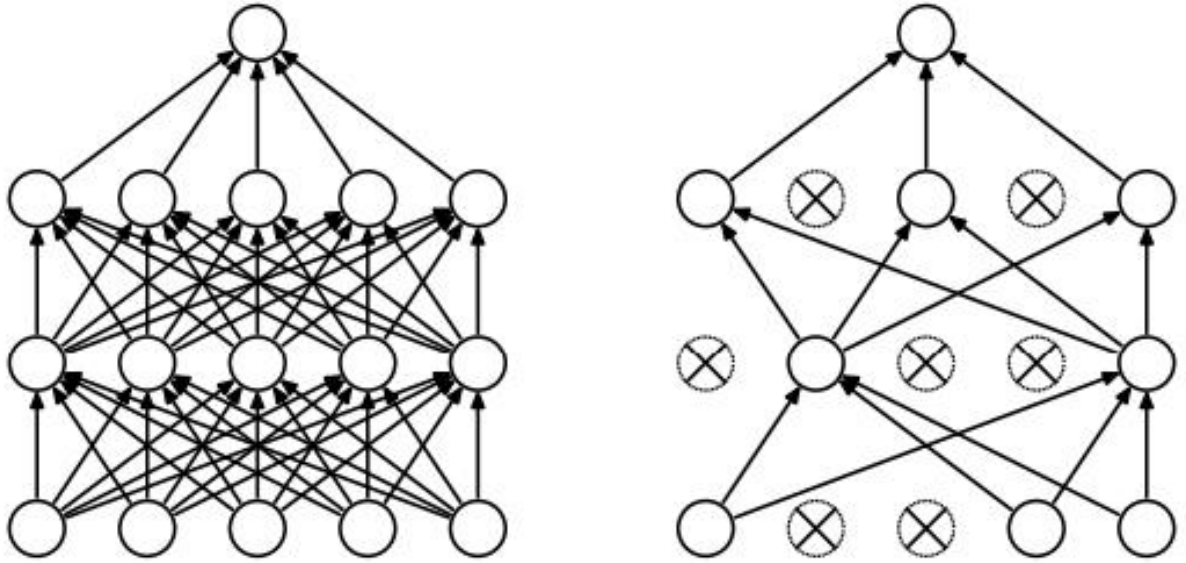
VC 维：$h_1 \leqslant h_2 \leqslant h_3$

奥卡姆剃刀(Occam's Razor)

- 奥卡姆剃刀（Occam‘s Razor），拉丁文为lex parsimoniae，意思是简约之法则。
- 是由14世纪逻辑学家、圣方济各会修士威廉奥卡姆William of Occam （约1287年至1347年）提出的一个解决问题的法则。
- 他在《箴言书注》第2卷15章说“切勿浪费较多东西，去做：用较少的东西，同样可以做好的事情”。
- 奥卡姆剃刀定律被广泛运用在多个学科的逻辑定律，它的简单表述：
  - 如无必要，勿增实体
  - Entities should not be multiplied unnecessarily

减少泛化误差

- 根据奥卡姆剃刀原则
  - 引入惩罚项，使较简单的模型比复杂的模型更可取
    - 引入正则项
    - 神经网络中，引入dropout机制



(a) Standard Neural Net

(b) After applying dropout.

减少泛化误差

- **使用验证集**
  - 该方法中，不是用训练集估计泛化误差，而是把原始的训练数据集分为两个较小的子集，一个子集用于训练，而另一个称为验证集，用于估计泛化误差。
  - 该方法为评估模型在未知样本上的性能提供了较好办法。