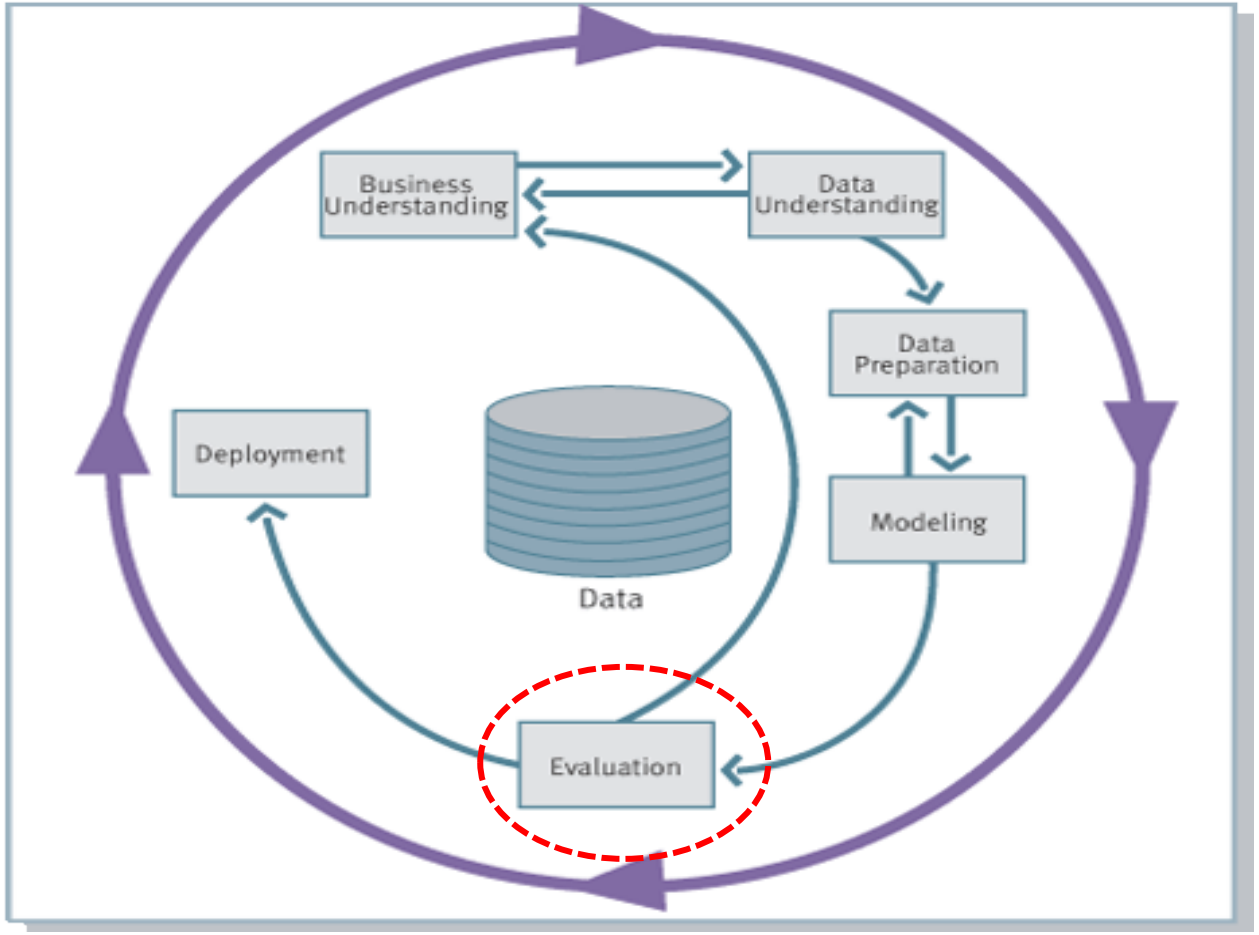
# MM5425 商业分析

WEEK 6 LECTURE – MODEL EVALUATION (DECISION TREE)

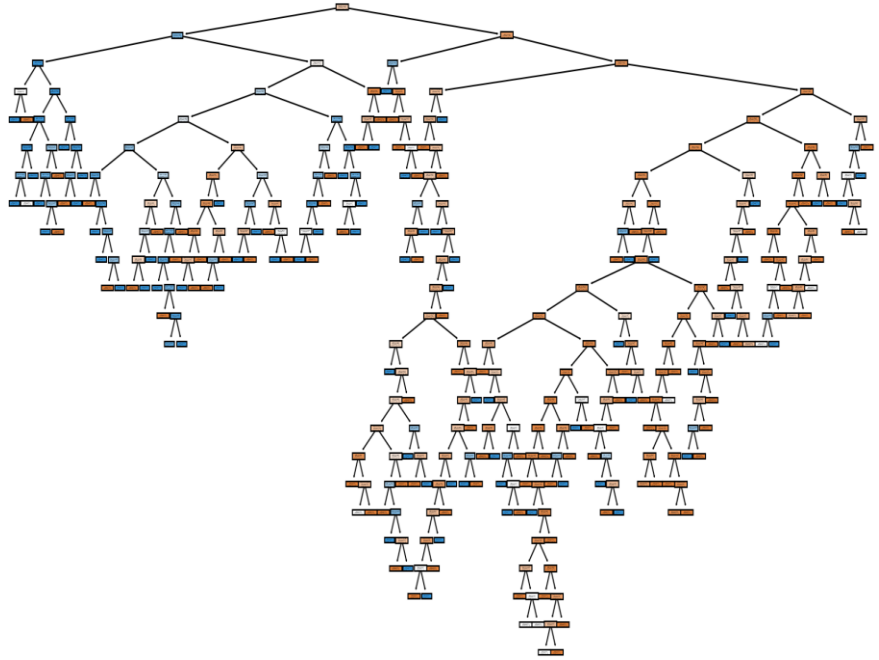
# WK6 目录Contents

# 评估


Business Understanding
Data Understanding
Data Preparation
Modeling
Deployment
Data
Evaluation

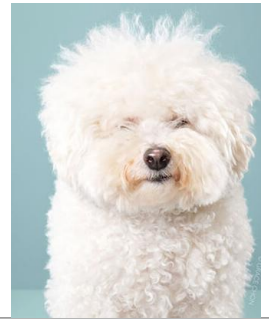
# 继续: 决策树 Decision Tree

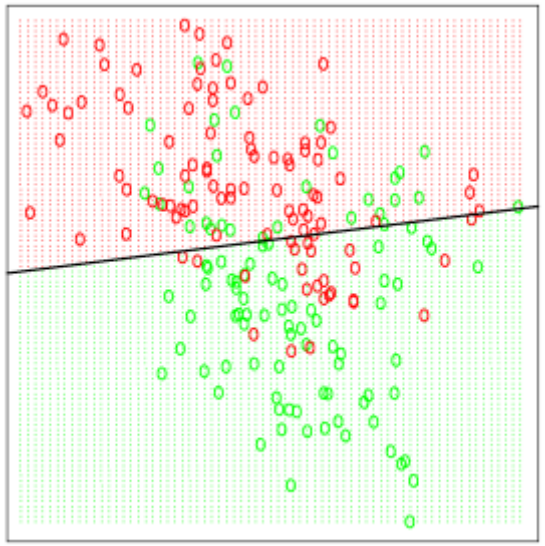
- 当达到最大纯度时（即，所有到达该节点的训练样本都属于同一类别），划分过程将停止。
- 进一步划分无法获得信息增益。



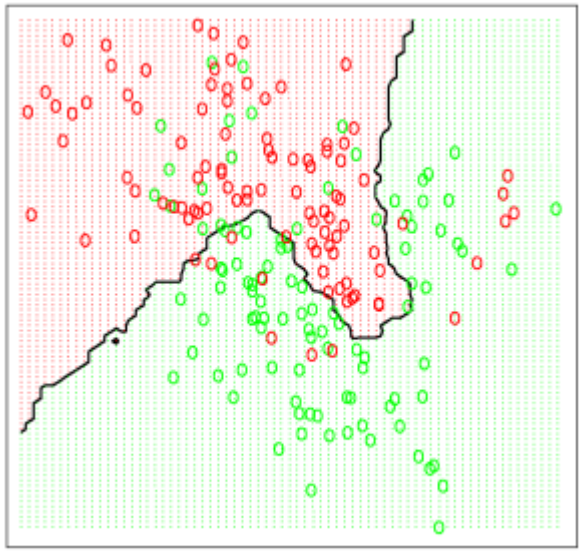
Question: 我们真的需要把决策树完全生长吗?

# 过度拟合

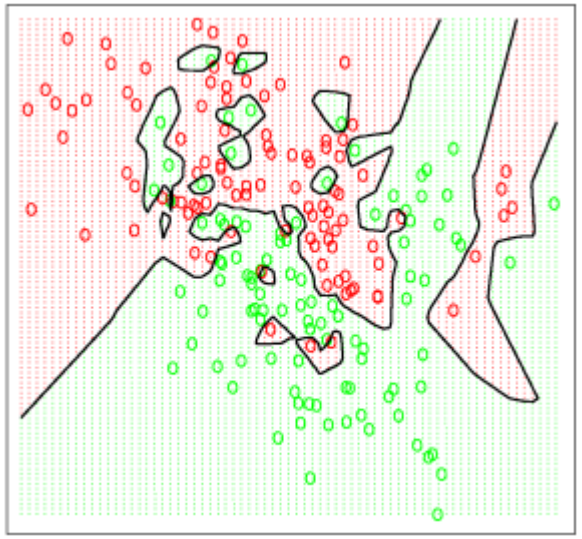




Under-fitting



Good



Over-fitting

# 过度拟合

- “如果你对数据严刑拷打，它最终会招供。”
  ——这句话形象地说明了数据挖掘过程中，模型往往会过度拟合训练数据，从而牺牲了对新数据的泛化能力。
- 过拟合的现象：

  模型在训练数据上的表现非常好，但在未见过的新数据上表现不佳。

  这是因为模型过于“记住”了训练集的细节和噪声，而没有学到数据的普遍规律。
- 我们追求的目标：

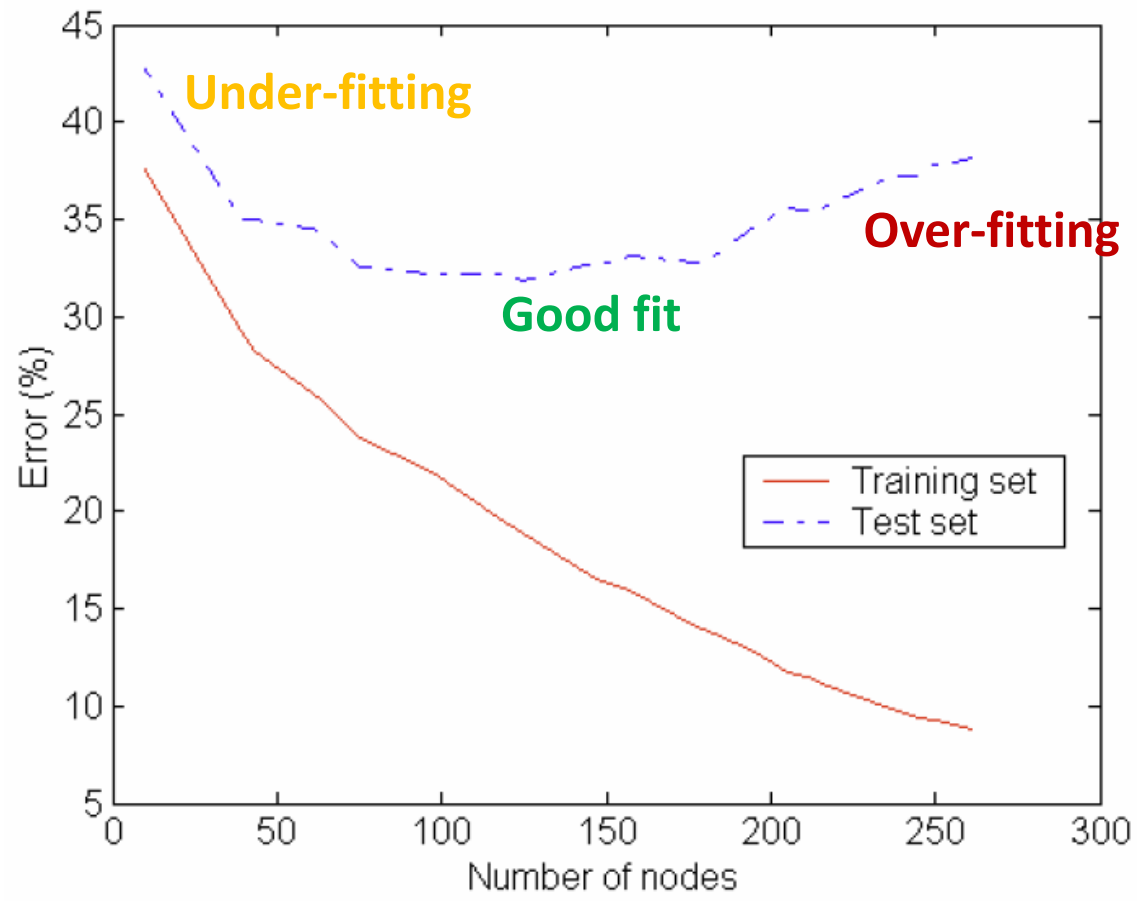
  希望模型不仅仅适用于训练集，而是能够推广到训练数据所代表的总体（general population）。

  这就是“泛化”（Generalization）：模型对新、未见过的数据也能做出准确预测的能力。
- 总结：

  泛化能力强的模型，才是真正有用的模型。我们要避免过度拟合，追求模型在新数据上的良好表现。

泛化是指模型对新的、以前未见过的数据能够做出正确预测的能力，这些预测与用于创建模型的数据类似。.

# 过拟合的症状Symptom of Overfitting




**Error rate = 1 - Accuracy**

# 如何避免过拟合？——剪枝（Pruning）

剪枝通过简化决策树来防止对数据中的噪声产生过拟合。

1) 预剪枝：在决策树变得过于复杂之前停止生长

   例如，分裂后每个节点的数据点太少（比如少于总数的5%）；

   树太深。

1) 后剪枝：先生成完整的决策树，然后再“剪枝”，减少树的节点数。

   实际应用中更倾向于使用后剪枝。

# 1) 停止树的生长

- 通过以下方式停止树的生长：

  限制树的最大深度

  树的深度：从最深的节点到根节点的边数

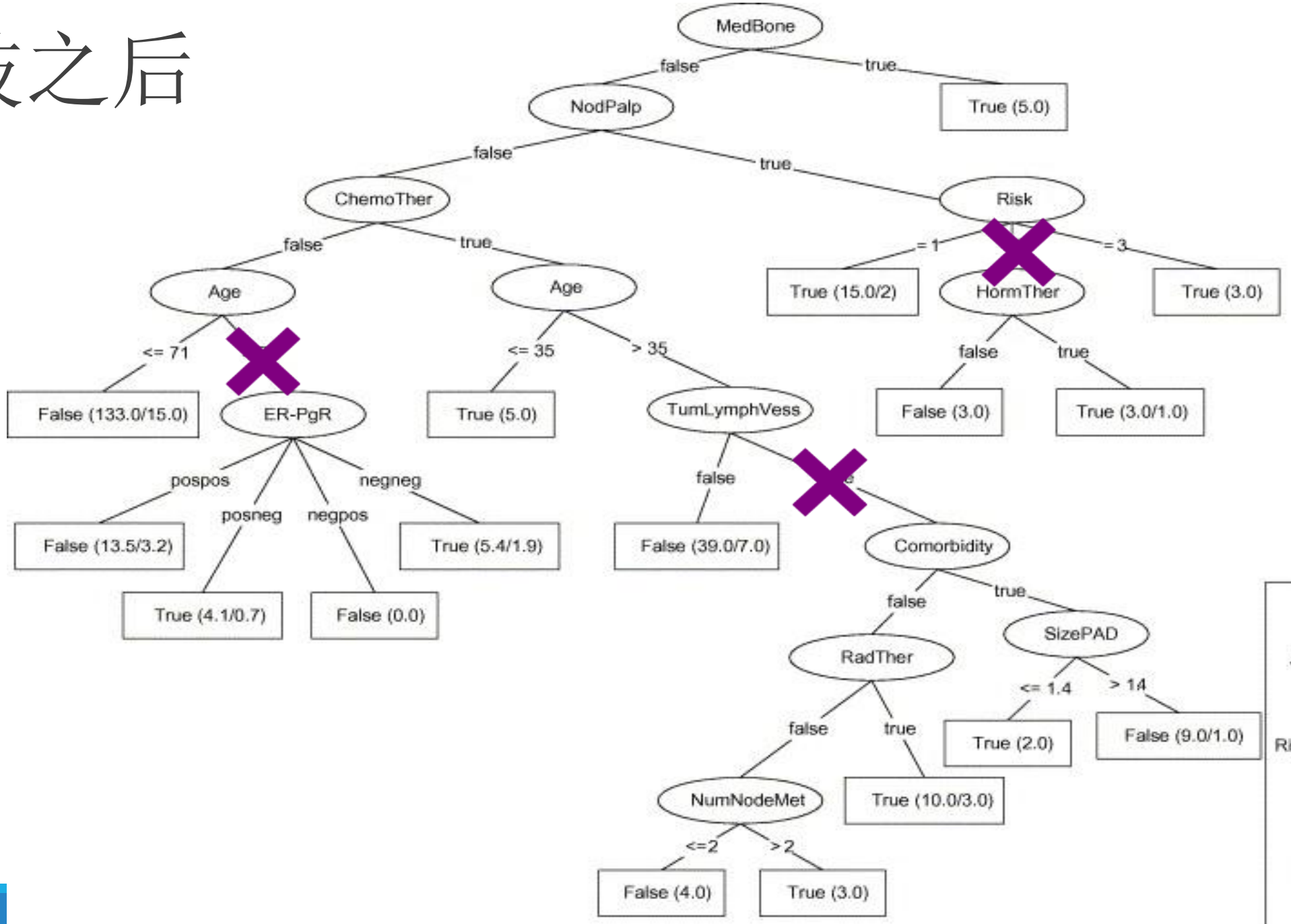
  限制每个节点分裂所需的最小样本数

  限制分裂所需的最小不纯度减少量

- 在Python中，可以通过调整 max_depth、min_samples_split、min_impurity_decrease 等参数来微调决策树。

- 这些方法不是完全基于数据驱动的，但可以结合领域知识进行设置。

# 2) 剪枝之后

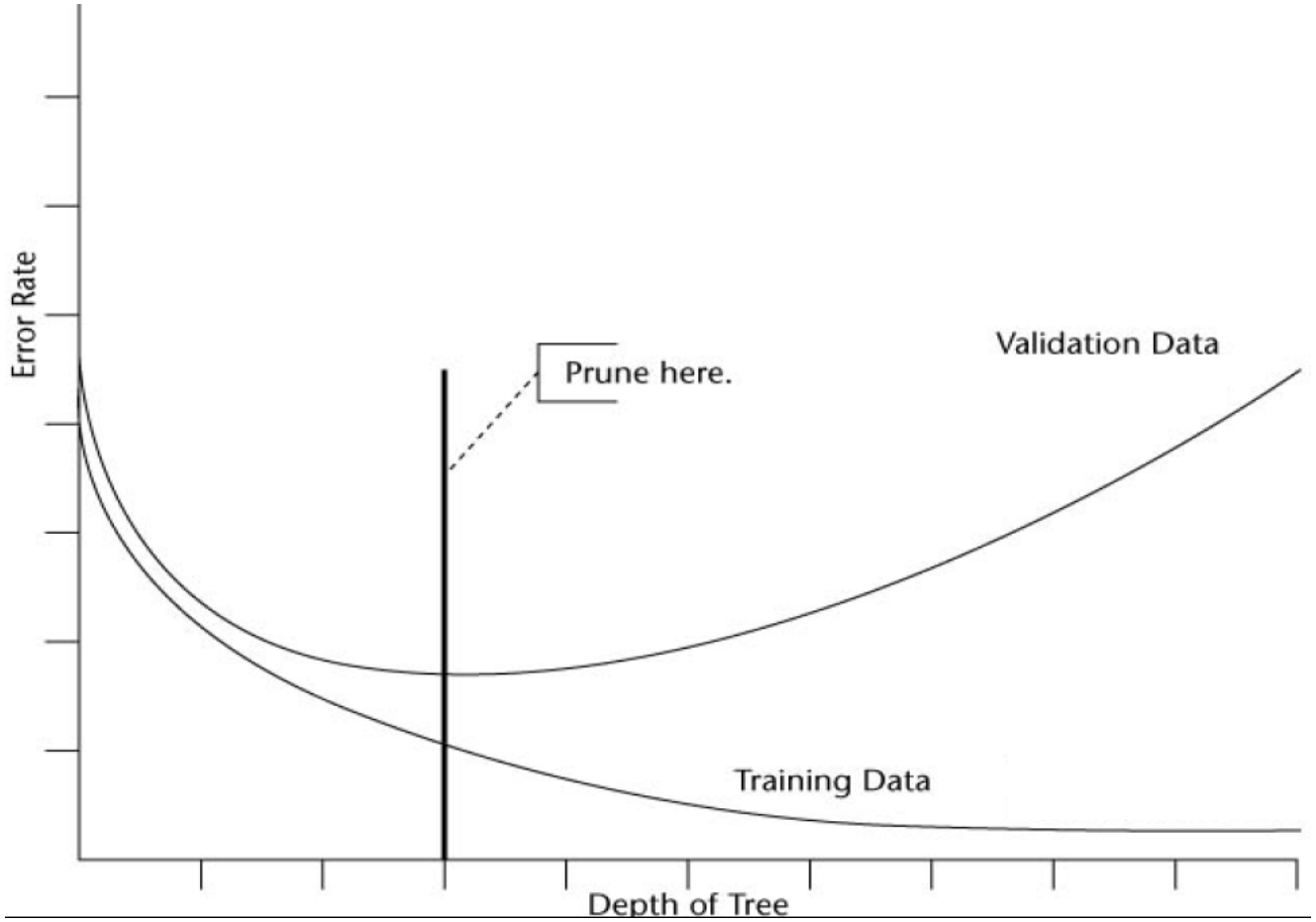

MedBone
false
true
True (5.0)
NodPalp
false
true
ChemoTher
Risk
false
true
= 1
= 3
Age
Age
True (15.0/2)
HormTher
True (3.0)
<= 71
<= 35
> 35
false
true
False (133.0/15.0)
ER-PgR
True (5.0)
TumLymphVess
False (3.0)
True (3.0/1.0)
pospos
posneg
negpos
negneg
false
False (13.5/3.2)
True (5.4/1.9)
False (39.0/7.0)
Comorbidity
True (4.1/0.7)
False (0.0)
false
true
RadTher
SizePAD
<= 1.4
> 14
false
true
True (2.0)
False (9.0/1.0)
NumNodeMet
True (10.0/3.0)
<=2
>2
False (4.0)
True (3.0)
J48 Decision tree
MedSpec
Class attribute
5-Year-State
Risk feature included
PC 76%
Kappa 0.34
AUC 0.64
Sensitivity 0.40
Specificity 0.91

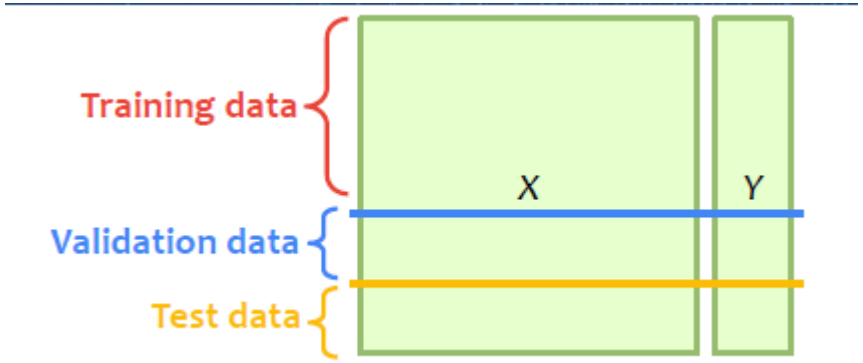
# 2) 后剪枝（Post-Pruning）决策树

简化误差剪枝（Reduced error pruning）

- 将训练数据分为训练集和验证集
- 验证集只用于参数调优，不用于测试！
- 在训练集上生成决策树
- 以自底向上的方式修剪决策树
- 如果剪枝后的树在验证集上的表现不比原来的差，就去除该子树
- 一旦确定了超参数和模型结构，最终用测试集评估模型性能

# 减少错误剪枝

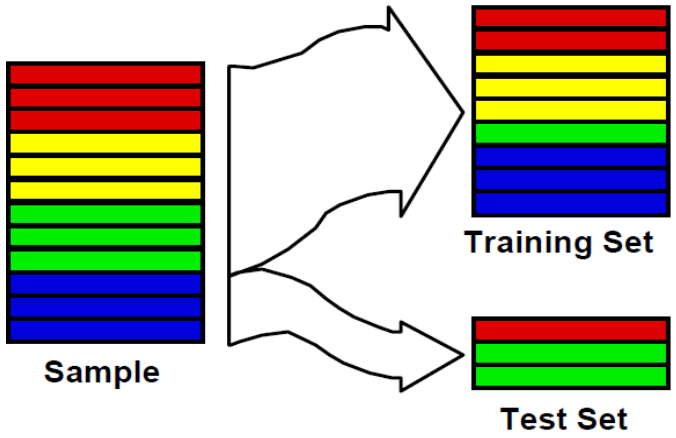

Error Rate
Prune here.
Validation Data
Training Data
Depth of Tree

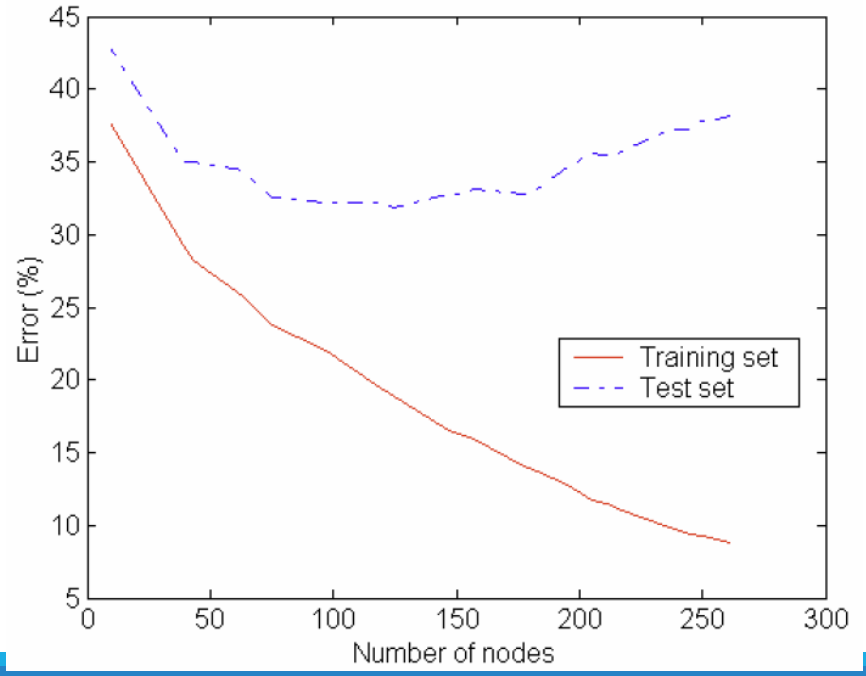

Training data
X
Y
Validation data
Test data

# 留出法评估Holdout Evaluation

当只有一个数据集时，我们会保留一部分数据（即 holdout），这些数据的目标变量已知，用于模型评估。

用于最终评估的这部分保留数据称为测试集（test set）。



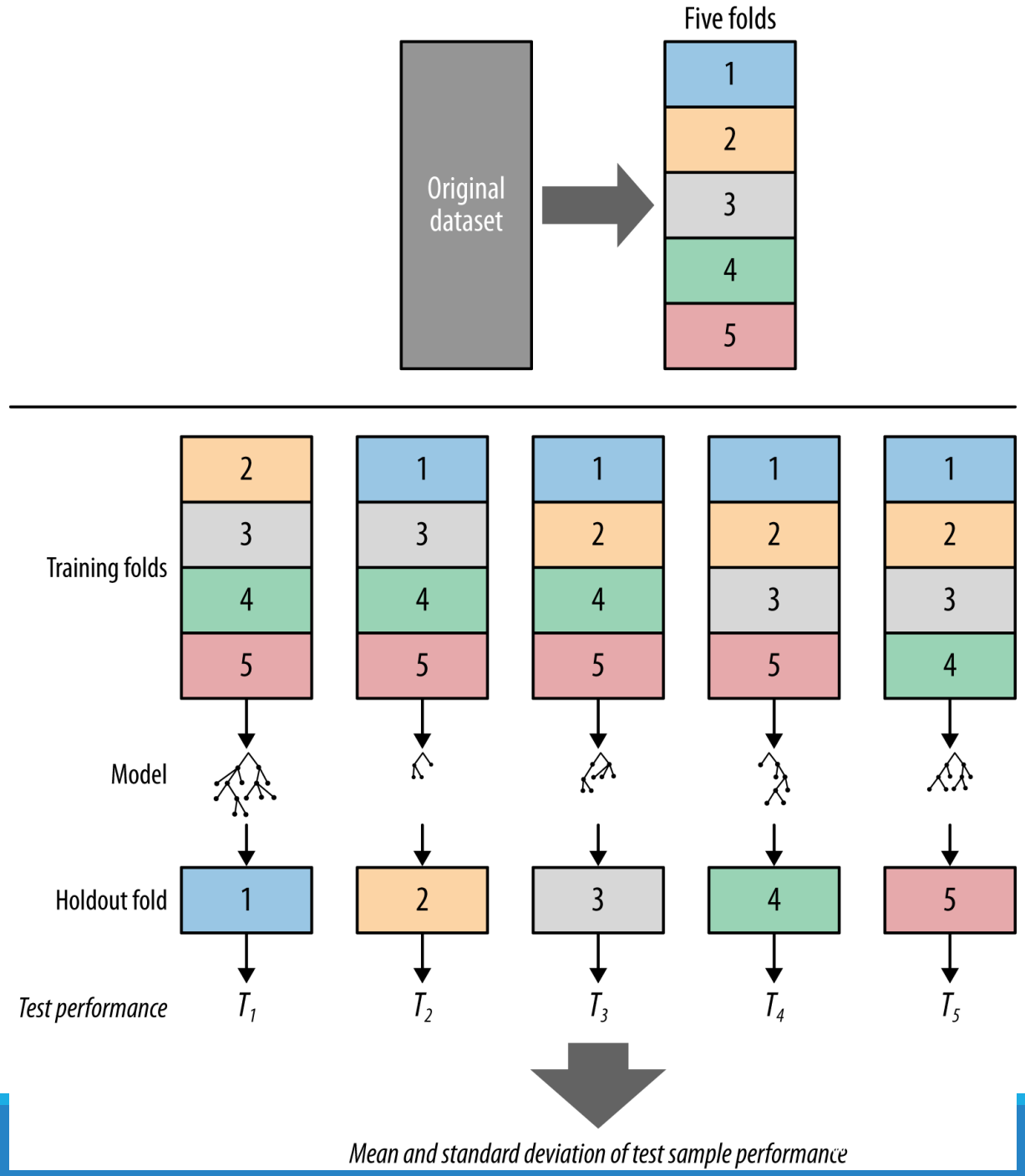




- 训练数据上的准确率：称为“样本内”（in-sample）准确率
- 测试数据上的准确率：称为“样本外”（out-of-sample）准确率，也叫泛化准确率（generalization accuracy）

# 交叉验证Cross-Validation (CV)

- 留出法的误差估计在不同的数据划分下可能会有较大波动，尤其是在数据集较小时更为明显。
- 针对数据有限的情况，可以采用 k 折交叉验证（k-fold cross validation）：

  - 将数据随机分成 k 个子集（folds）
  - 进行 k 次训练/测试评估
  - 每次实验用其中 k-1 个子集作为训练集，剩下的 1 个子集作为测试集
  - 这样可以更好地估计模型的真实性能
  - 可以统计性能估计的均值和方差等指标
  - 评估对性能估计的置信度




- ✓每个折（fold）都会被用作一次测试集（其余折合并为训练集）。
- ✓这样可以保证所有数据都被测试一次（每个数据点都作为测试集出现一次）。
- ✓最终可以计算准确率等评估指标的平均值和方差。

# 需要多少折?

在实际应用中，折数的选择取决于数据集的大小：

- - 对于大型数据集，即使使用 3 折交叉验证也能获得较为准确的评估结果。
- - 对于非常稀疏或较小的数据集，为了让模型尽可能多地在训练样本上学习，通常需要使用更多的折数（例如 k=N，即留一交叉验证，leave-one-out cross-validation）。
- 这样可以充分利用有限的数据，提高模型评估的可靠性。

k 折交叉验证的常见做法是选择 k=5（Python 中的默认值）或 k=10。

# 决策树的优缺点

优点：

- 生成透明的规则，易于理解和解释
- 需要很少的数据预处理，变量选择是自动完成的
- 能够处理属性与目标之间的非线性关系

缺点：

- 树结构不稳定，对数据的微小变化很敏感
- 可能发生过拟合，因此需要剪枝步骤，并且通常需要较大的数据集来构建较好的模型
- 每次分裂只基于一个属性，无法很好地覆盖属性之间的组合关系（例如回归模型中的交互项）

# 表现评估

## MORE THAN JUST ACCURACY

# 两种类型的评估

数据驱动的评估

- 例如：训练-测试集划分、交叉验证

领域知识评估

- 将模型作为建模者与利益相关者之间的接口
- 拥有一个利益相关者能够理解的模型非常重要
- 与现有知识进行对比
- 让专家对模型进行评估

# 分类模型评估：准确率 / 错误率

**准确率（Accuracy）：** 预测正确的样本数占总样本数的比例。

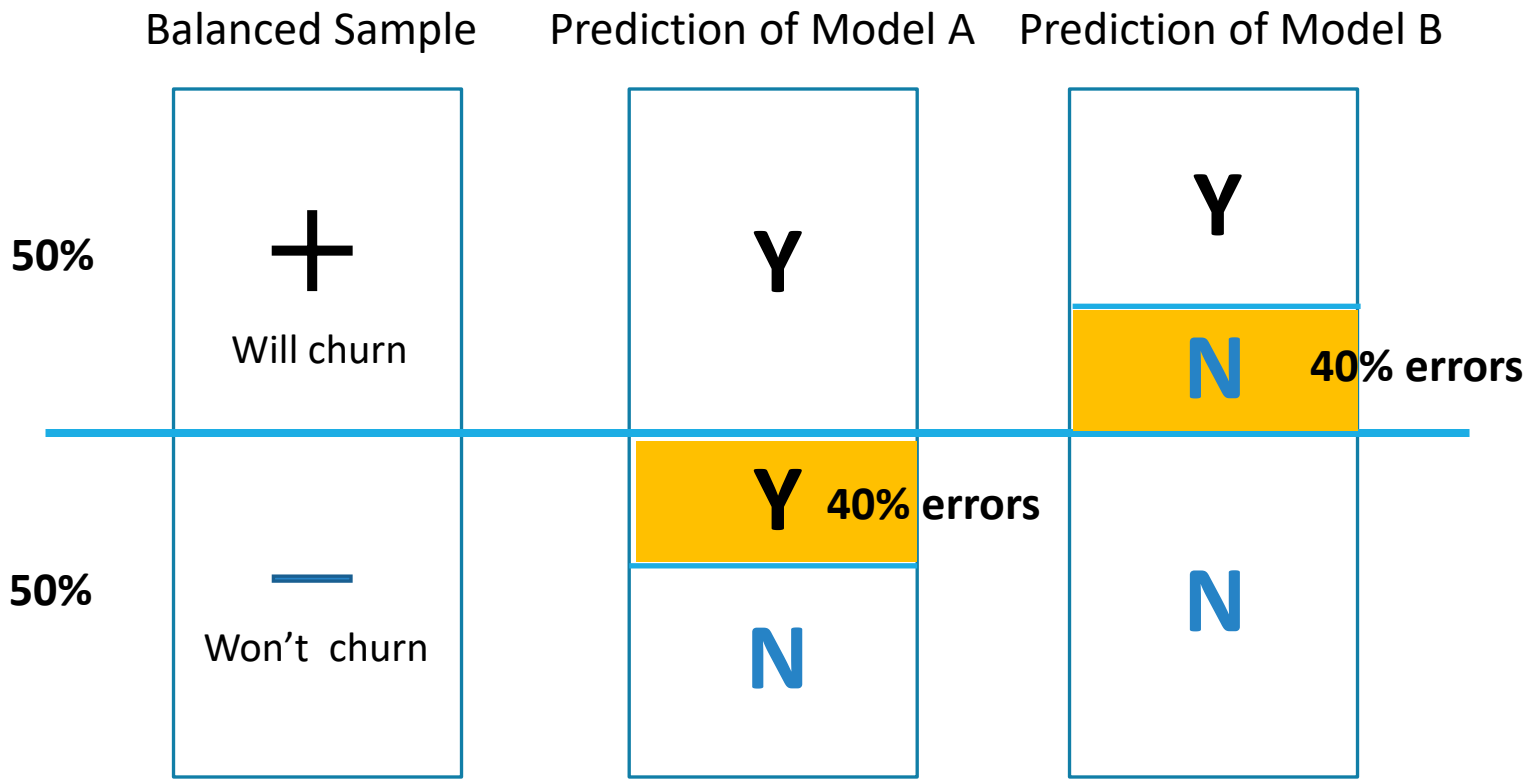
**错误率（Error rate）：** 预测错误的样本数占总样本数的比例，等于 **1 - 准确率**。

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of instances in testing set}}$$

$$= 1 - \text{Error rate}$$

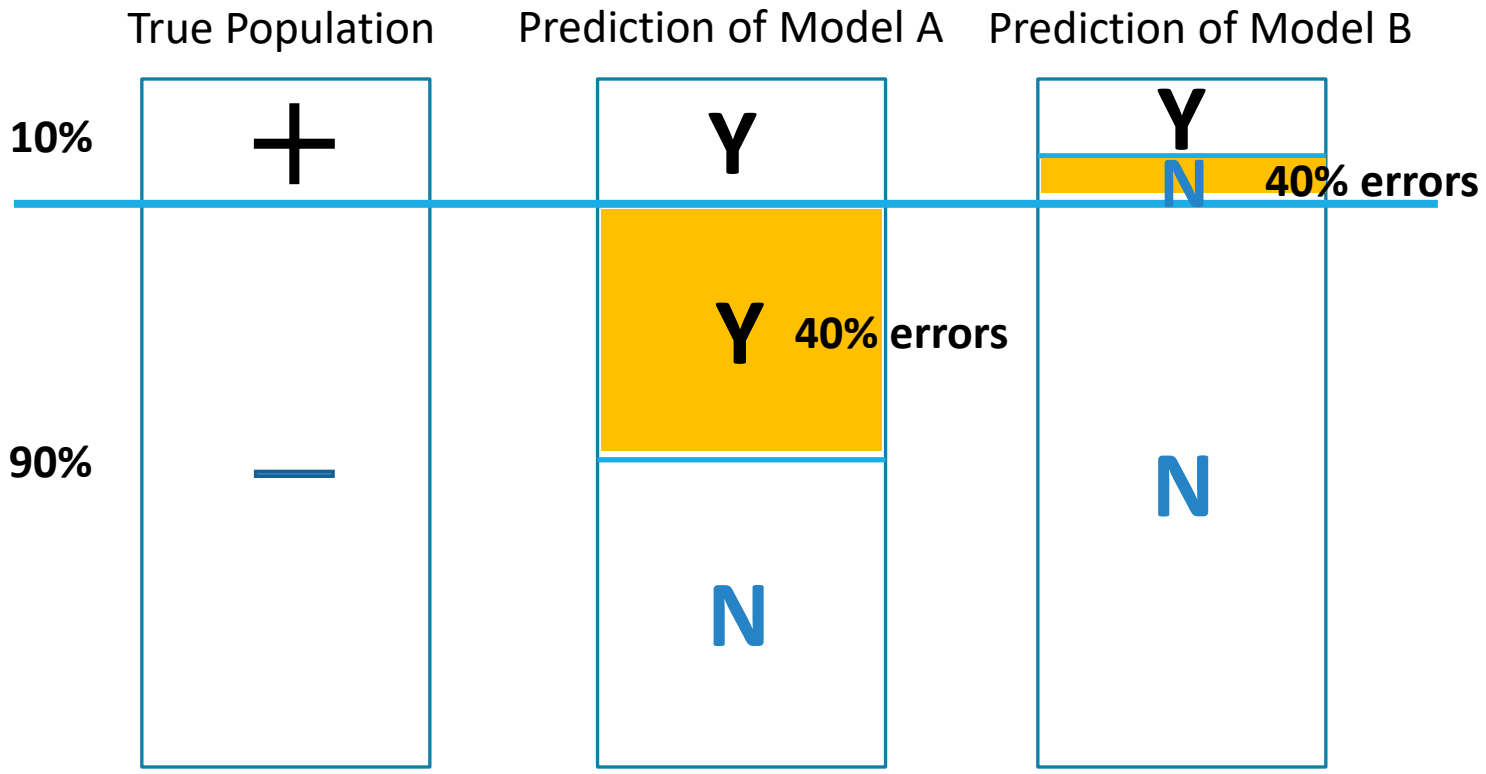
过于简单……有时会产生误导，尤其是在类别分布不均衡或偏斜的情况下。

# 样本类别分布均衡的准确率




两种模型都能正确分类平衡样本中80%的人群。

# 样本类别分布不均衡的准确率




模型A的准确率下降到64%，而模型B的准确率上升到96%。

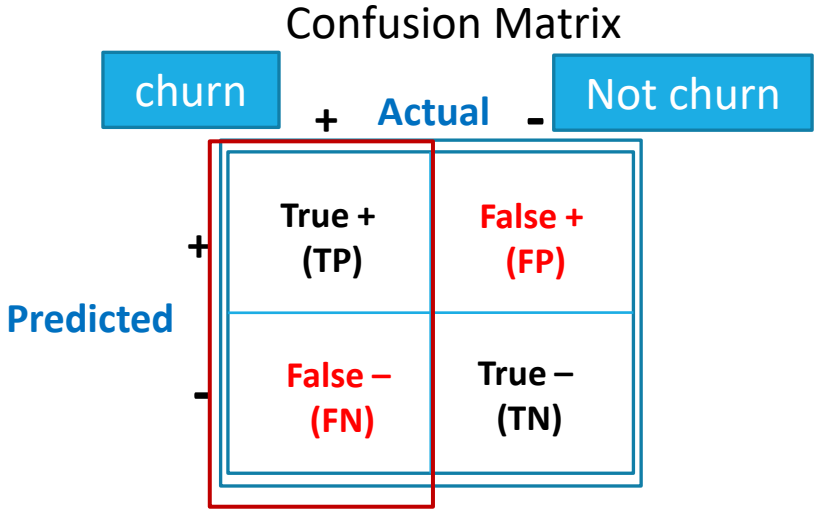
# 精准率与召回率指标

- **精准率（Precision）**：在所有被模型预测为正类的样本中，实际为正类的比例。
  公式：Precision = TP / (TP + FP)
- **召回率（Recall）**：在所有实际为正类的样本中，被模型正确预测为正类的比例。
  公式：Recall = TP / (TP + FN)

其中，TP为真正例，FP为假正例，FN为假负例。精准率和召回率常用于类别分布不均衡的分类问题，可以更全面地评估模型性能。

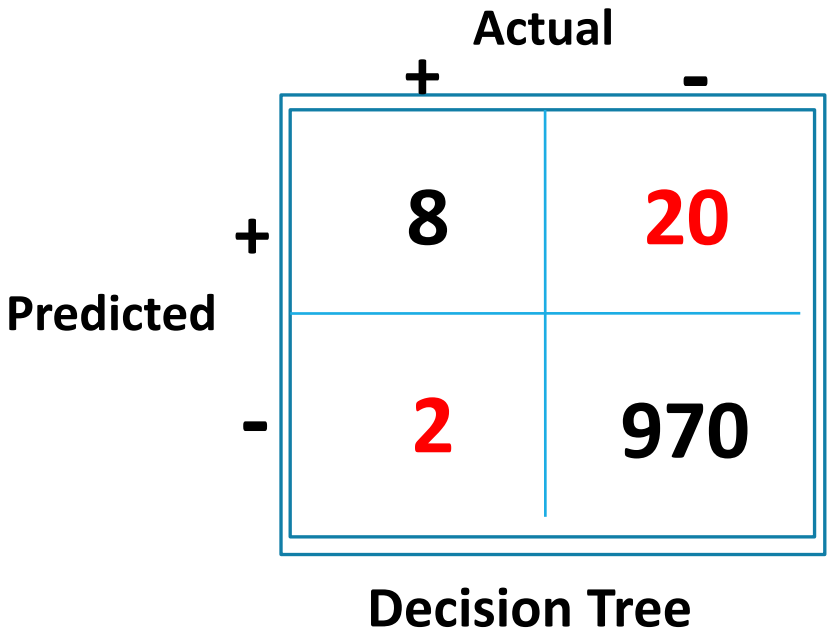
$$\text{Precision (+)} = \frac{TP}{TP + FP}$$

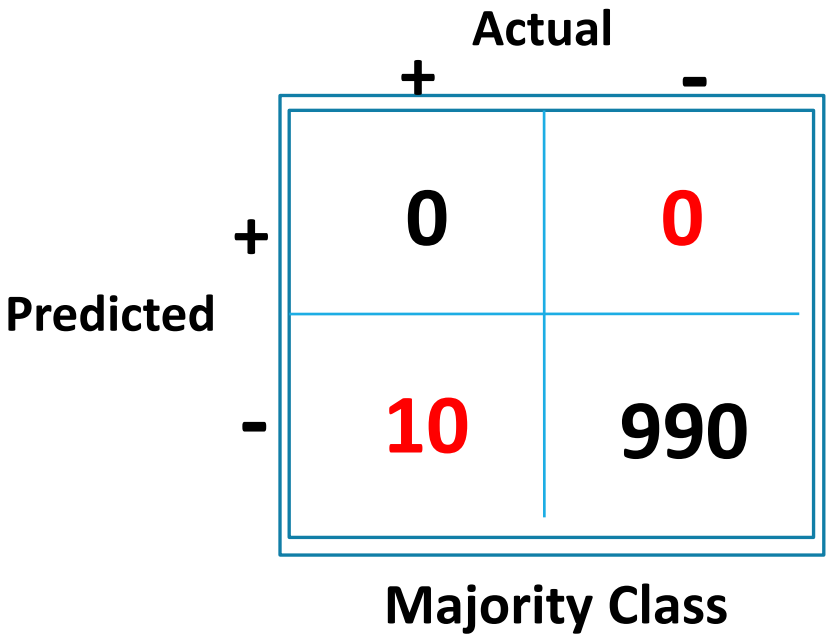
$$\text{Recall (+)} = \frac{TP}{TP + FN}$$



# 练习

Q: 每个模型的准确率、精准率和召回率分别是多少？？

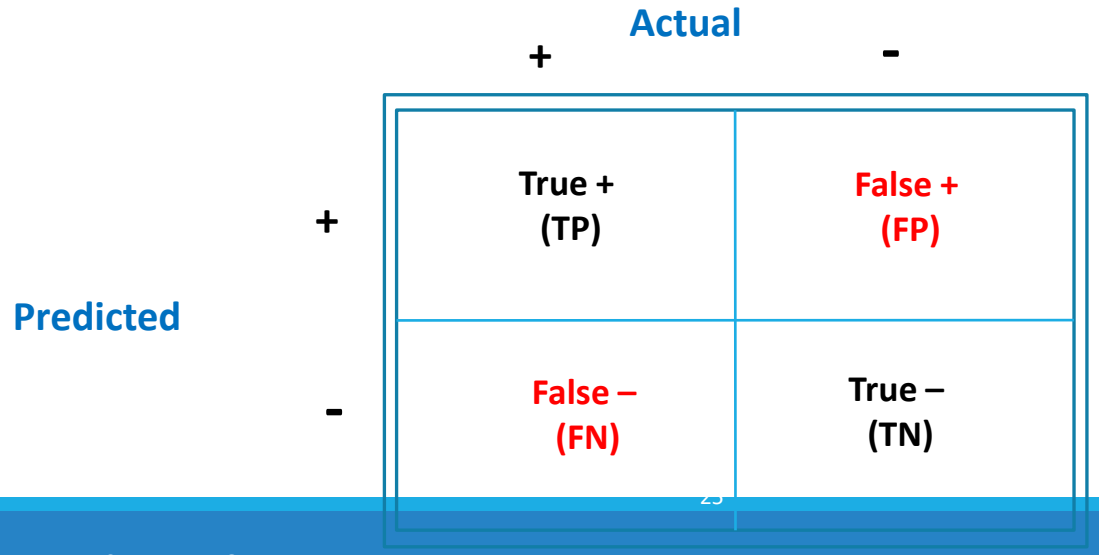


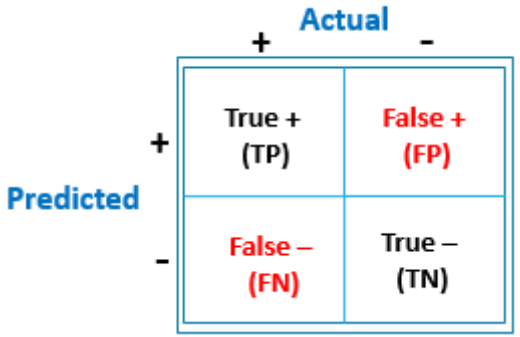
# 两类错误的代价不均等

在许多应用中，不同类型的错误具有不同的代价。

- 错误批准一份信用卡申请的代价远高于错误拒绝一份申请。
- 错误过滤掉一封正常邮件的代价远高于错误接受一封垃圾邮件。
- 错误地将患病患者诊断为正常的代价远高于错误地将正常人诊断为患病。



# 有害的正例与无害的负例




正例

表示出现了异常情况，例如检测到疾病、发现欺诈案件等。

通常较为罕见，值得关注或警惕。

负例

表示正常或无特殊意义的结果。

在实际应用中，针对负例的错误（即假阳性错误，false positive）可能数量较多，但针对正例的错误（即假阴性错误，false negative）所带来的代价通常更高。

# 用期望值来衡量模型评估

**Confusion Matrix (N=110)**

| | Actual + | Actual - |
|---|---|---|
| **Predicted +** | 56 | 7 |
| **Predicted -** | 5 | 42 |

**Cost/Benefit Matrix**

| | Actual + | Actual - |
|---|---|---|
| **Predicted +** | 99 | -1 |
| **Predicted -** | 0 | 0 |

$$EV=p(TP)*v(TP)+p(FP)*v(FP)+p(TN)*v(TN)+p(FN)*v(FN)$$

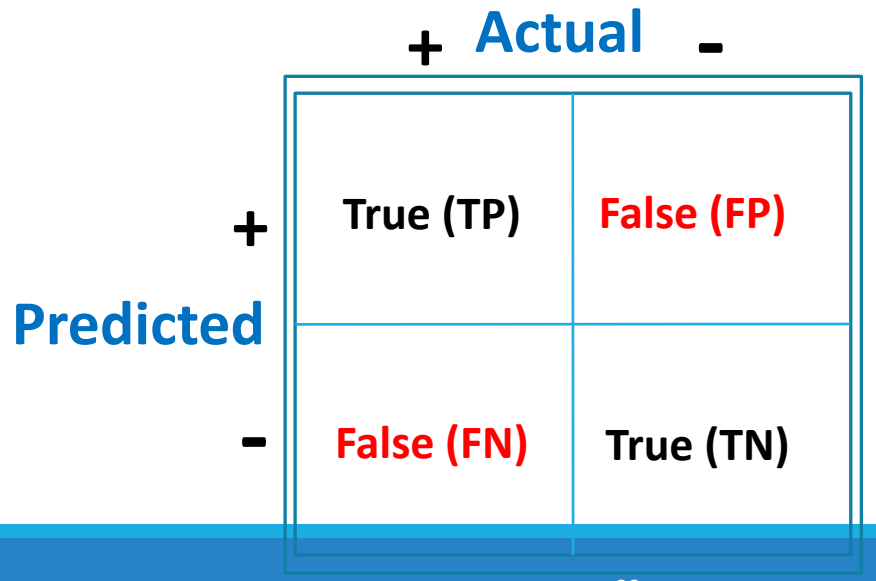
# ROC 分析

ROC曲线（接收者操作特征曲线）

- - 最初于20世纪50年代在信号检测理论中提出，用于分析噪声信号。
- - 是一种系统性的方法，用于评估概率预测的质量。
- - ROC曲线的评估结果不受类别比例和错误代价结构的影响。
- - 对于许多业务相关人员来说，ROC曲线并不是最直观的可视化方式。

$$\text{True Positive Rate(TPR), Recall} = \frac{TP}{TP + FN}$$

$$\text{False Positive Rate(FPR)} = \frac{FP}{FP + TN}$$



# 决策阈值Decision Threshold

给定类别概率估计（PE），调整决策阈值会影响真正例（TP）和假正例（FP）的比例。

具体来说：

- 降低决策阈值：更多样本会被判为正类，TP数量可能增加，但FP数量也会增加（召回率提高，精准率可能下降）。
- 提高决策阈值：更少样本会被判为正类，TP数量可能减少，但FP数量也会减少（精准率提高，召回率可能下降）。

因此，通过调整决策阈值，可以在TP率和FP率之间进行权衡，以满足不同业务需求。

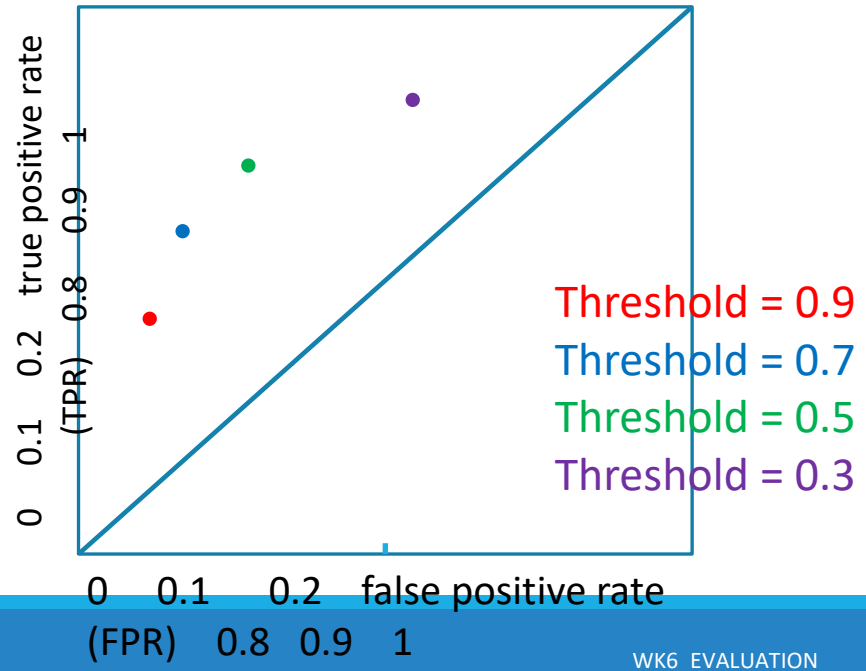
4 examples

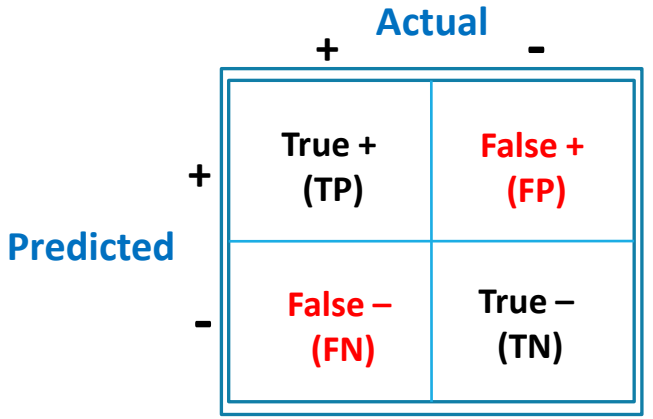
| PE for + | Prediction (0.5 threshold) | Prediction (0.8 threshold) | True value |
|---|---|---|---|
| 0.3 | - | - | - |
| 0.55 | + | - | + |
| 0.75 | + | - | - |
| 0.9 | + | + | - |
| True positive rate | 1/1 | 0/1 | |
| False positive rate | 2/3 | 1/3 | |
| Accuracy rate | 2/4=0.5 | 2/4=0.5 | |

# ROC 分析 Analysis

- 对于某个特定模型，每一个决策阈值都对应着一组TPR（真正例率，纵轴）和FPR（假正例率，横轴）的数值。
  - 当你调整决策阈值时，这个点在坐标系中的位置也会随之变化。

  换句话说，
  - - 不同的阈值会产生不同的TPR和FPR组合，
  - - 这些点连起来就形成了ROC曲线。



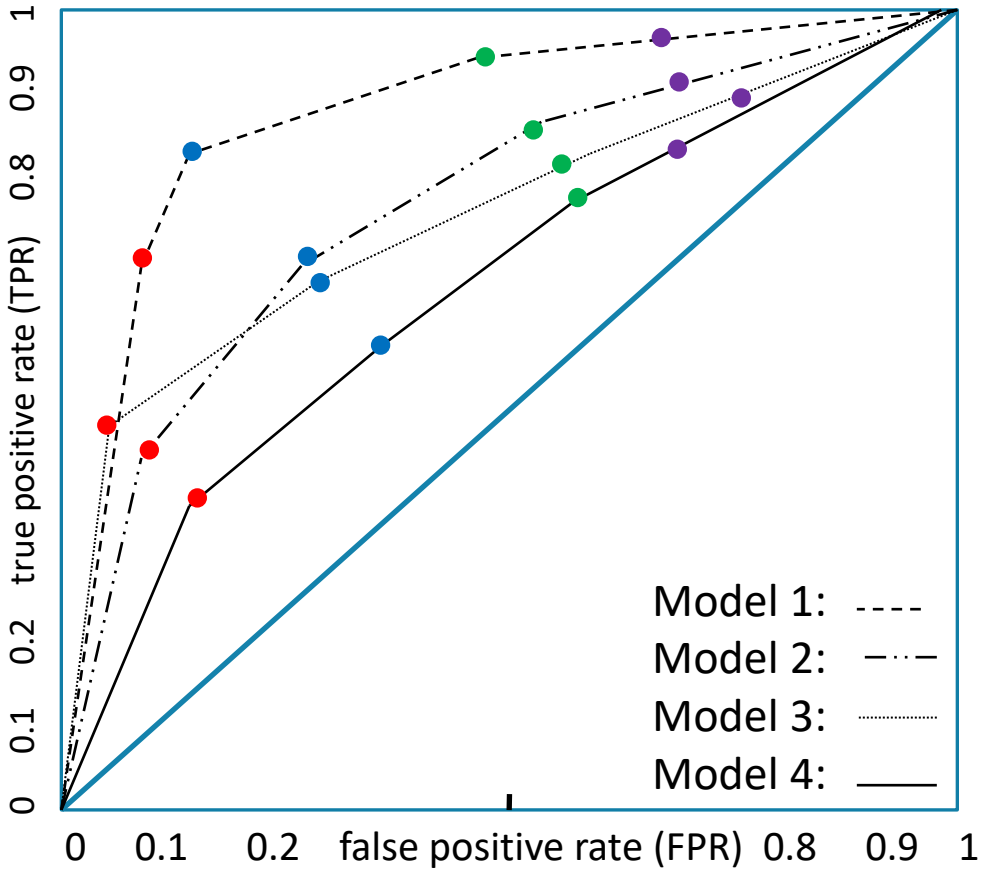
# ROC曲线 ROC Curve

将这些点连接起来，就可以得到一个模型的曲线（ROC曲线）。

不同的模型会有不同的ROC曲线。

ROC曲线下的面积（AuROC，Area under ROC Curve）越大，说明模型的性能越好。

**ROC曲线上的几个关键点和对角线分别代表什么含义？**



# 表现评估Performance Evaluation

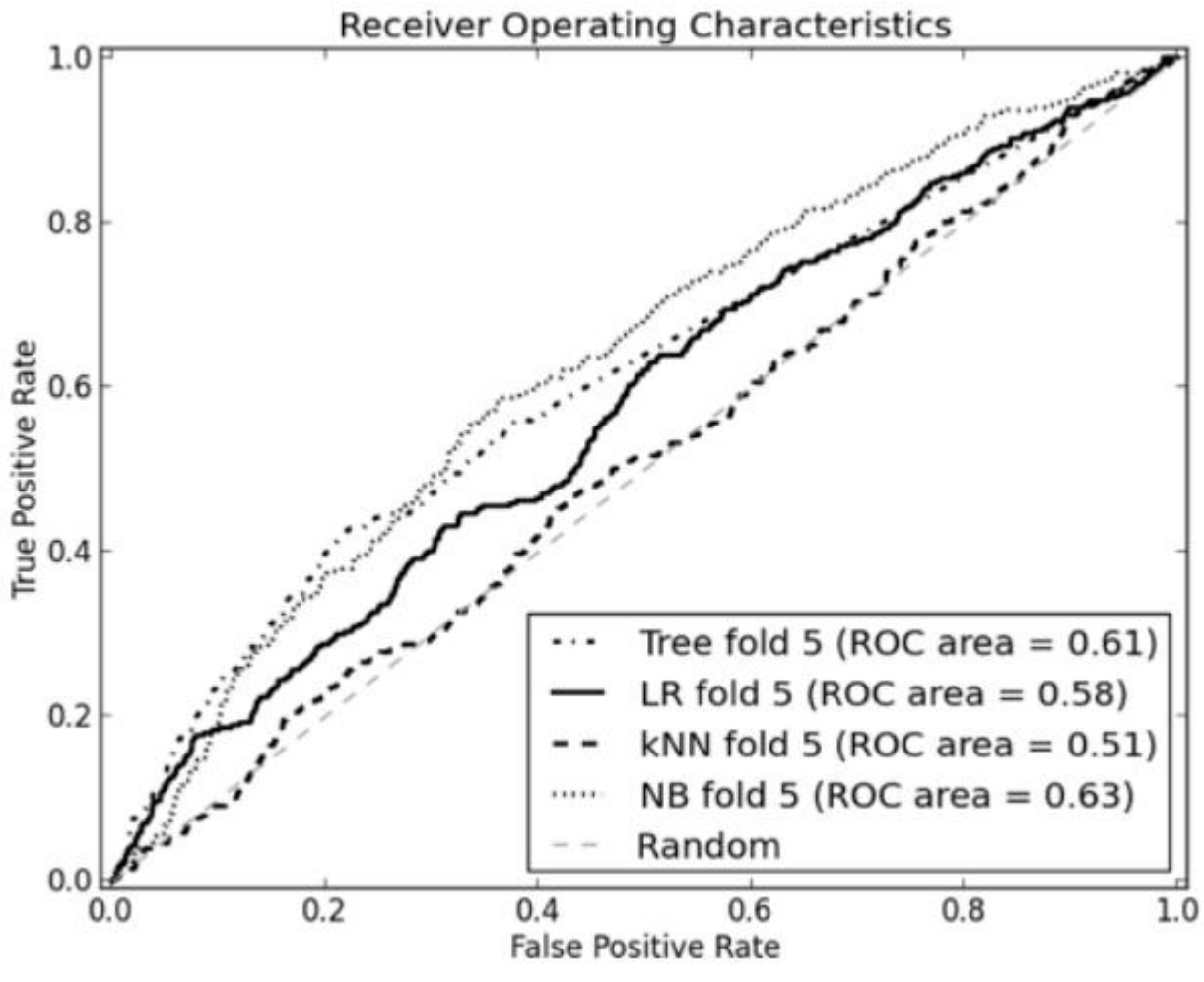
训练集Training Set:

| Model | Accuracy |
|---|---|
| Classification Tree | 95% |
| Logistic Regression | 93% |
| $k$-Nearest Neighbors | 100% |
| Naïve Bayes | 76% |

测试集Test Set:

| Model | Accuracy | AUC |
|---|---|---|
| Classification Tree | 91.8%±0.0 | 0.614±0.014 |
| Logistic Regression | 93.0%±0.1 | 0.574±0.023 |
| $k$-Nearest Neighbors | 93.0%±0.0 | 0.537±0.015 |
| Naïve Bayes | 76.5%±0.6 | 0.632±0.019 |

# ROC曲线 ROC Curve




**Reference：**
Business analytics: The science of data-driven decision making / U. Dinesh Kumar, New Delhi Wiley India, 2022