

Intro. to Machine Learning | Chpt5. Eval. of Mach. Learning

Evaluation of Machine Learning

Lecturer: MaoQiang Xie

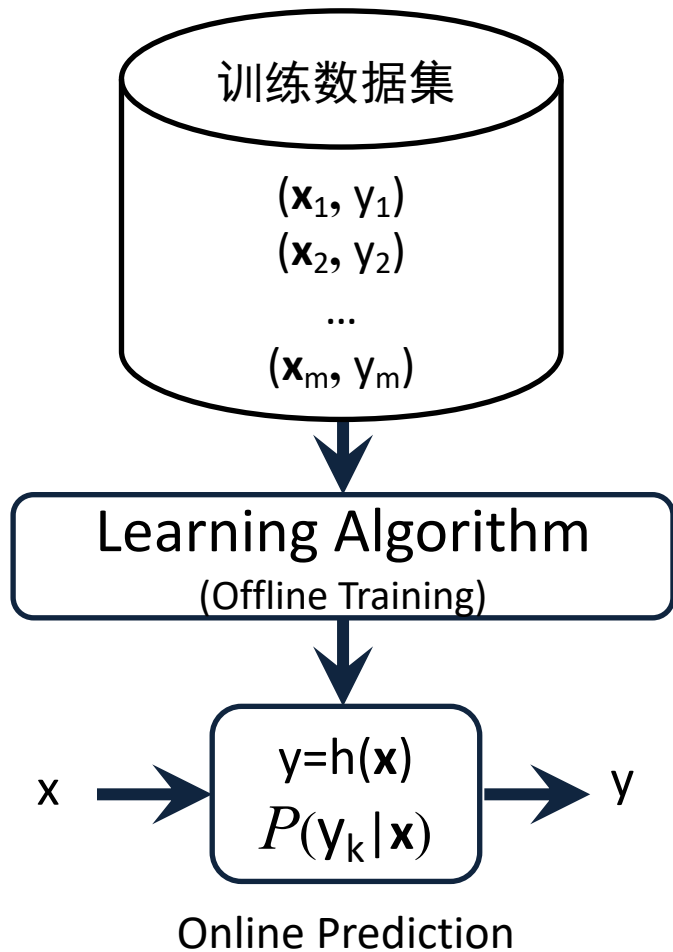
College of Software, Nankai University

Outline

- 机器学习回顾
- 机器学习预测结果的评估
 - 准确率、查准率、查全率、 F_1 Measure、ROC和AUC
 - 其他：调参、代价敏感错误率、代价敏感分类器
- 显著性检验
- 交叉验证

What's Machine Learning

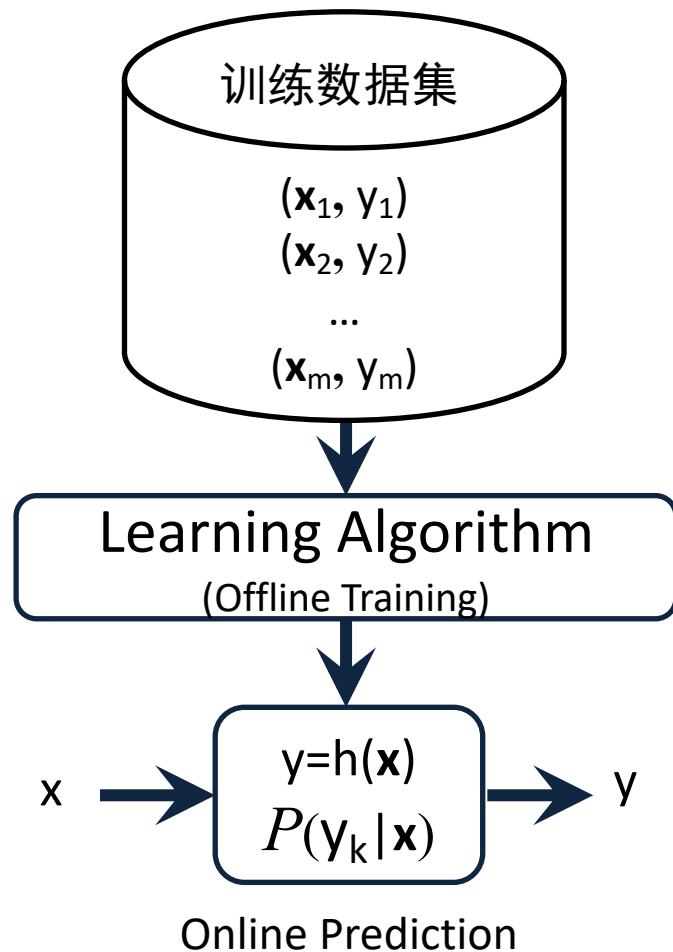
- **Definition**_[Tom Mitchell, 1997]: A computer program is said to *learn* from **experience** E with respect to some class of **tasks** T and **performance measure** P, if its performance at tasks in T, as measured by P, improves with experience E.
- **Machine Learning** Addresses the question of how to **build computer programs that improve their performance at some task through experience.**



- 目的：利用预测模型 $y=h(\mathbf{x})$ 或 $\operatorname{argmax}_k \{P(y_k|\mathbf{x})\}$ ，为用户提供的 x 预测输出 y
- 预测模型从何而来：从历史经验（带专家标注的训练数据 $\{(\mathbf{x}_i, y_i)\}$, $i=1, 2, \dots, m$ ）中学习

预测模型的性能

- 工程 v.s. 科学
- 量化评估是工程的基石
- 个人认为机器学习是体现自身优化最直观的技术
- 实验能力



Outline

- 机器学习回顾
- 机器学习预测结果的评估
 - 准确率、查准率、查全率、 F_1 Measure、ROC和AUC
 - 其他：调参、代价敏感错误率、代价敏感分类器
- 机器学习模型的评估
 - 期望风险、经验风险、结构风险
 - 交叉检验
 - 偏差与方差

分类准确率(Accuracy)

将 $err(h_{\Theta}(x), y)$ 的返回值控制在 $\{0, 1\}$,那么

$$\text{Accuracy} = \left(1 - \frac{1}{m} \sum_{i=1}^m err(h_{\Theta}(x^{(i)}, y^{(i)}))\right) \times 100\%$$

Ground Truth v.s. Prediction (Hypothesis Test)

		Predicted condition	
Total population		Predicted Condition positive	Predicted Condition negative
True condition	condition positive	True positive	False Negative (Type II error)
	condition negative	False Positive (Type I error)	True negative

Ground Truth v.s. Prediction

		Predicted condition	
		Predicted Condition positive	Predicted Condition negative
True condition	condition positive	True positive	False Negative (Type II error)
	condition negative	False Positive (Type I error)	True negative

TP: True Positive, 被判定为正样本, 事实上也是正样本。

TN: True Negative, 被判定为负样本, 事实上也是负样本。

FN: False Negative, 被判定为负样本, 但事实上是正样本。

FP: False Positive, 被判定为正样本, 但事实上是负样本。

(相对地, 用户更为关心positive结果和true样本的重合度)

Precision and Recall (Table 2.1 P.30)

Precision: 查准率，即在**分类器预测出的正例中**，真正正确的个数占整个结果的比例。

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

Recall: 查全率，即在预测结果中真正正确的个数**占整个标注数据集中正例个数**的比例。

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

Precision and Recall

- Precision和Recall相矛盾
 - 使用阈值(threshold)决定判别, 如LR, NN
 - 想将所有感兴趣的样本识别出来, 需要降低阈值, 扩大接受范围
 - 想提高识别感兴趣样本的查准率, 需要提高阈值, 提高拒绝率

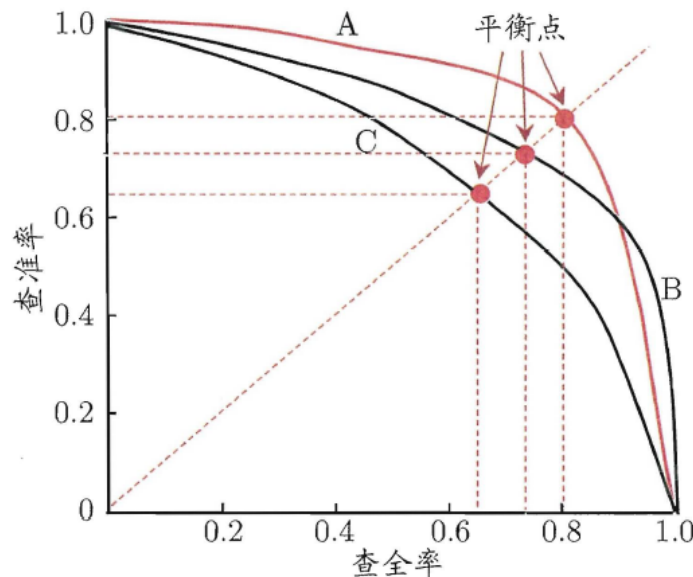


图 2.3 P-R曲线与平衡点示意图

Precision and Recall

- Precision和Recall相矛盾
 - 好的方法应该能够尽量包住对比方法的P-R曲线
 - 平衡点（Break-Event Point）
“查准率=查全率”时的取值
可以用来作为衡量分类器性能
的指标之一

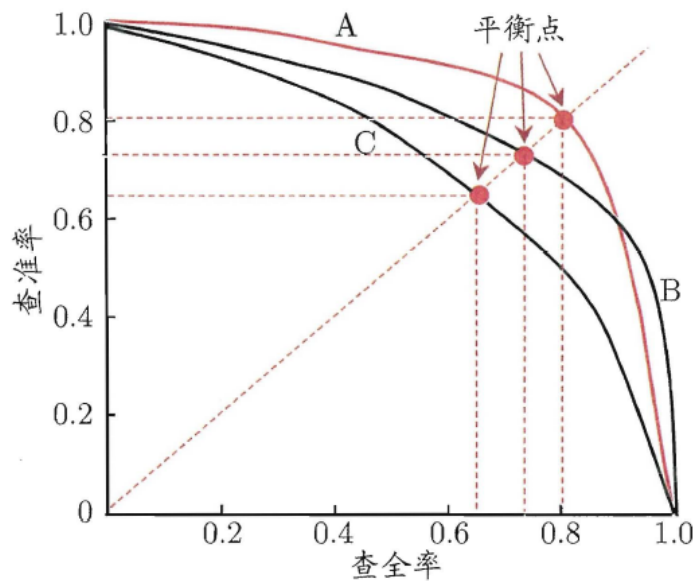


图 2.3 P-R曲线与平衡点示意图

F_1 measure

相比于P-R曲线的平衡点， F_1 度量更为专业一些，它是precision和recall的调和平均（harmonic mean），最差为0，最好为1

$$F_1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} = \frac{2TP}{2TP + FP + FN}$$

调和平均（harmonic mean）又叫倒数平均

$$H_n = \frac{n}{\frac{1}{a_1} + \frac{1}{a_2} + \dots + \frac{1}{a_n}}$$

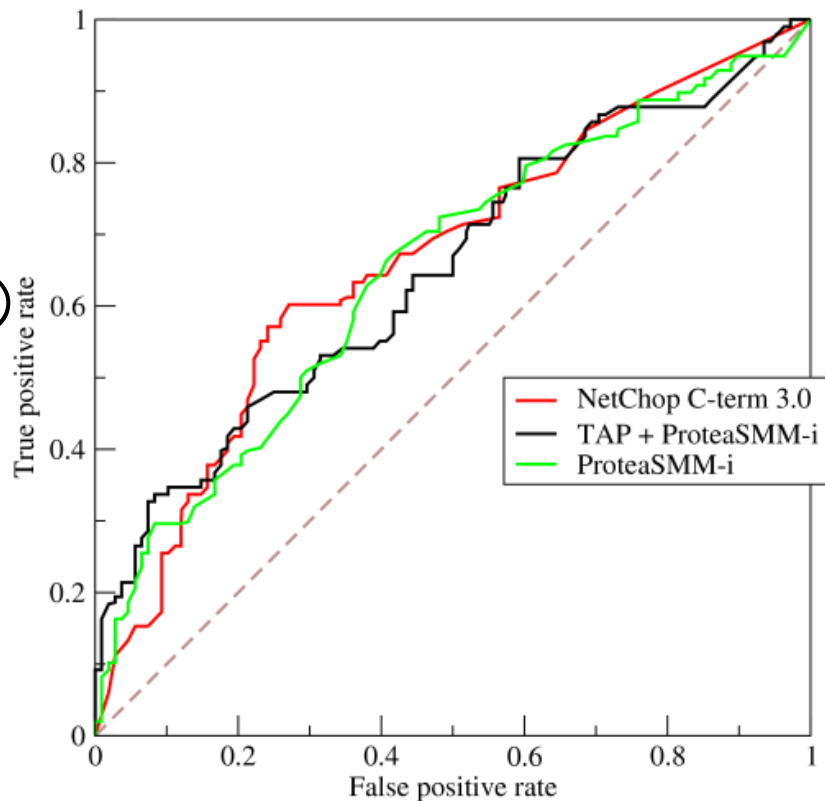
ROC

预测方法：

- 1) 使用阈值（阈值确认风险大）
- 2) 使用排序（搜索引擎v.s.问答系统）

排序做法：

将最可能的正例排在前，最不可能是正例的排在后面，通过cut-point区分预测结果是否为正

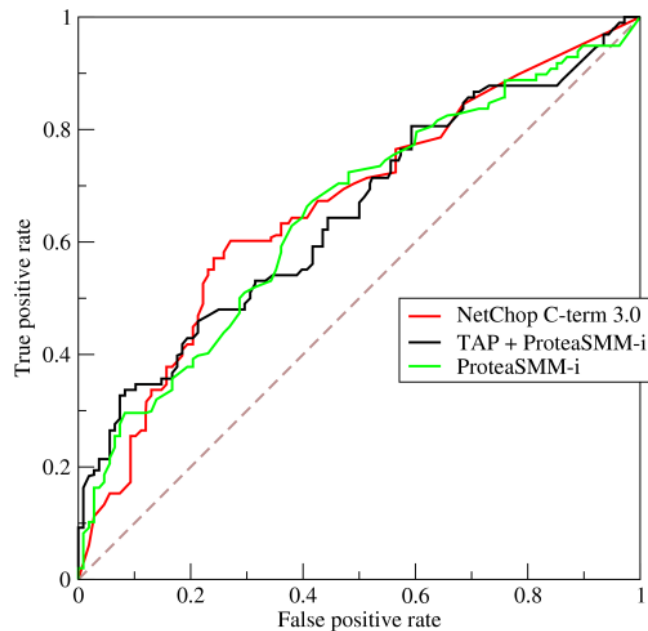


ROC

- 通过不断调整分割点，使用排序更能全方位反映分类器性能（泛化性能）。
- ROC曲线则可用于评估不同分割点下的排序性能

Y轴: $TPR = TP / (TP + FN)$

X轴: $FPR = FP / (FP + TN)$



True positive	False Negative (Type II error)
False Positive (Type I error)	True negative

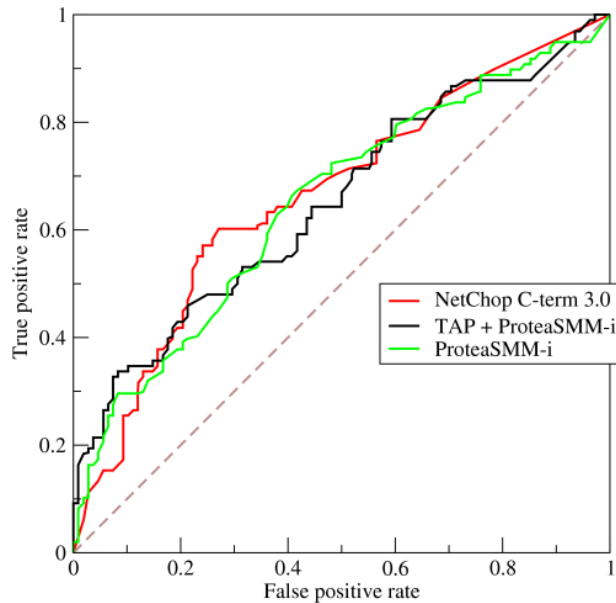
ROC

接收者操作特征曲线（Receiver Operating Characteristic curve，或者叫ROC曲线

Y轴： $TPR = TP / (TP + FN)$

X轴： $FPR = FP / (FP + TN)$

通过TPR和FPR之间的关系来判断分类器性能的好坏。偏向左上好，偏向右下不好



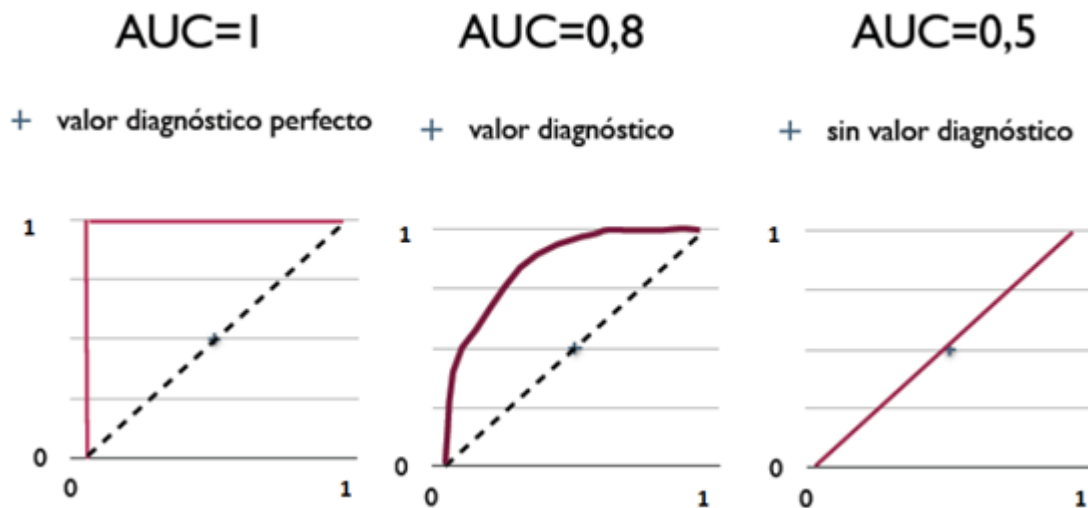
True positive	False Negative (Type II error)
False Positive (Type I error)	True negative

AUC(Area Under Curve) of ROC

若随机抽取一个true样本和一个false样本，AUC表示分类器接受true样本高于接受false样本的概率

AUC取值在0~1之间

AUC值越大的分类器，
正确率越高



A

TP=63	FP=28	91
FN=37	TN=72	109
100	100	200

TPR = 0.63

FPR = 0.28

ACC = 0.68

B

TP=77	FP=77	154
FN=23	TN=23	46
100	100	200

TPR = 0.77

FPR = 0.77

ACC = 0.50

C

TP=24	FP=88	112
FN=76	TN=12	88
100	100	200

TPR = 0.24

FPR = 0.88

ACC = 0.18

C'

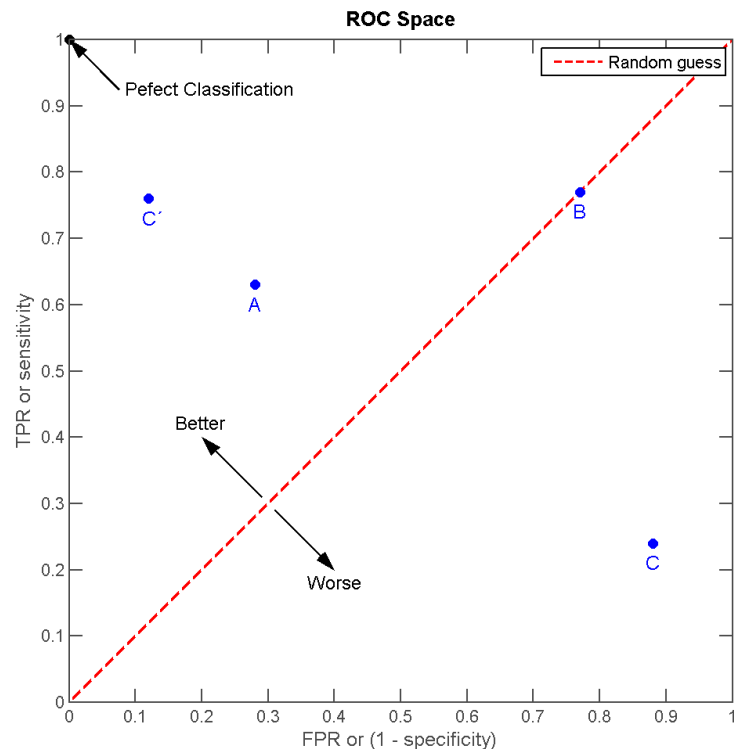
TP=76	FP=12	88
FN=24	TN=88	112
100	100	200

TPR = 0.76

FPR = 0.12

ACC = 0.82

ROC



Calculation of AUC

梯形法（trapezoid method）：简单地将每个相邻的点以直线连接，计算连线下方的总面积。因为每一线段下方都是一个梯形，所以叫梯形法。

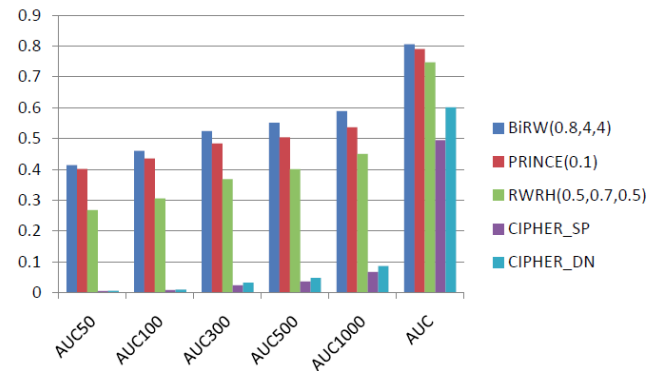
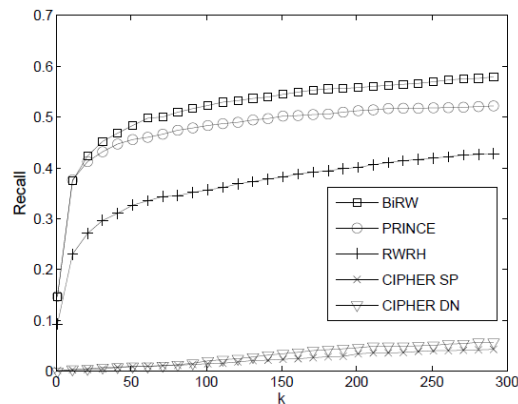
可以看作积分

具体实现：随着FP的增加，累加TP

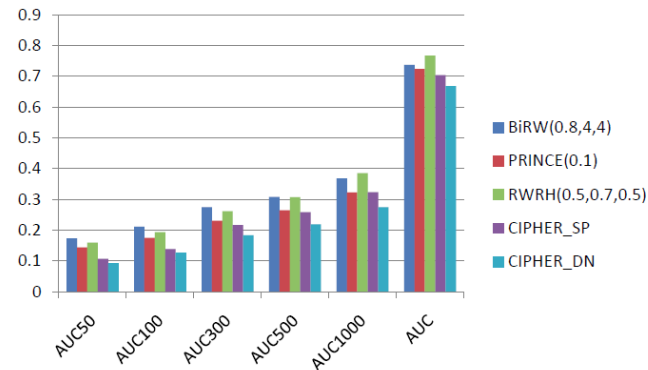
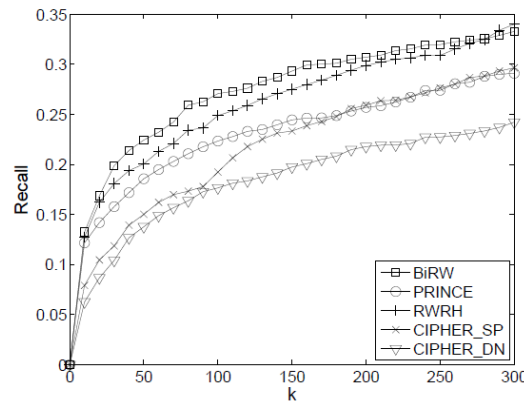
Examples of AUC

AUC@K

Recall@K



(A) 100-fold Cross-validation



(B) Test Data

Sensitive Cost

例1. “将患者诊断为健康人” 和 “把健康人诊断为患者” 的后果不同

例2. “降低信用评估标准贷款” 和 “提高信用评估标准拒绝贷款” 的后果不同

之前提到的算法将不同类型的错误赋予 “等价错误”

Sensitive Cost

例1. “将患者诊断为健康人” 和 “把健康人诊断为患者” 的后果不同

例2. “降低信用评估标准贷款” 和 “提高信用评估标准拒绝贷款” 的后果不同

之前提到的算法将不同类型的错误赋予 “等价错误”

Sensitive Cost

- 以二分类为例，可以计算不同类型的错误，将FP和FN分别计算，记为 $cost_{10}$ 和 $cost_{01}$ 。
- 解决方法
 - 按照任务的关注，选择 $cost_{ij}$ 小的模型
 - 按照任务关注，调整损失函数

表 2.2 二分类代价矩阵

真实类别	预测类别	
	第 0 类	第 1 类
第 0 类	0	$cost_{01}$
第 1 类	$cost_{10}$	0

Cost Sensitive Error Rate

$$E(f; D; cost) = \frac{1}{m} \left(\sum_{\mathbf{x}_i \in D^+} \mathbb{I}(f(\mathbf{x}_i) \neq y_i) \times cost_{01} + \sum_{\mathbf{x}_i \in D^-} \mathbb{I}(f(\mathbf{x}_i) \neq y_i) \times cost_{10} \right) .$$

Student's T-Test

主要用于判断所提方法(proposed method)与基准方法(baseline methods)之间的性能差异是否明显 (significantly)

例如：这些方法在多个任务（数据集）上的性能结果作比较。

$\text{mean}(\text{PM}) > \text{mean}(\text{BM})$ ，则对 $[h, p, ci] = \text{ttest2}(\text{PM}, \text{BM})$;

- 当 $h=1$ 时，表明可以从统计上断定算法A1的结果大于A2的结果（即两组数据均值的比较是有意义的）
- $h=0$ 则表示不能根据平均值来断定两组数据的大小关系（因为区分度小）

Conclusion

- 机器学习回顾
- 机器学习预测结果的评估
 - 准确率、查准率、查全率、 F_1 Measure、ROC和AUC
 - 其他：调参、代价敏感错误率、代价敏感分类器
- 显著性检验
- 交叉验证

- Training/Empirical Error

$$\text{Training Error} = \frac{1}{m_{\text{train}}} \sum_{i=1}^{m_{\text{train}}} \text{err}(h_{\Theta}(x_{\text{train}}^{(i)}), y_{\text{train}}^{(i)})$$

$$\text{CV Error} = \frac{1}{m_{\text{CV}}} \sum_{i=1}^{m_{\text{CV}}} \text{err}(h_{\Theta}(x_{\text{CV}}^{(i)}), y_{\text{CV}}^{(i)})$$

- Test/Generalization Error

$$\text{Test Error} = \frac{1}{m_{\text{test}}} \sum_{i=1}^{m_{\text{test}}} \text{err}(h_{\Theta}(x_{\text{test}}^{(i)}), y_{\text{test}}^{(i)})$$

Cross Validation

- **N-fold Cross-Validation (Fig. 2.2, page 26)**

将整个训练数据集划分成 n 份，选取其中一份作为验证数据集，其余 $n-1$ 份作为训练数据集。

进行轮换让每一份数据都有机会作为验证数据集。尽可能避免overfitting

- **Leave One Out Cross-Validation**

是N-fold Cross-Validation的特例，每个样本都当作一份验证数据集

Evaluating your hypothesis

Dataset:

Size	Price	
2104	400	60% Training set
1600	330	
2400	369	
1416	232	
3000	540	
1985	300	
1534	315	20% Cross validation set (cv)
1427	199	
1380	212	20% test set
1494	243	

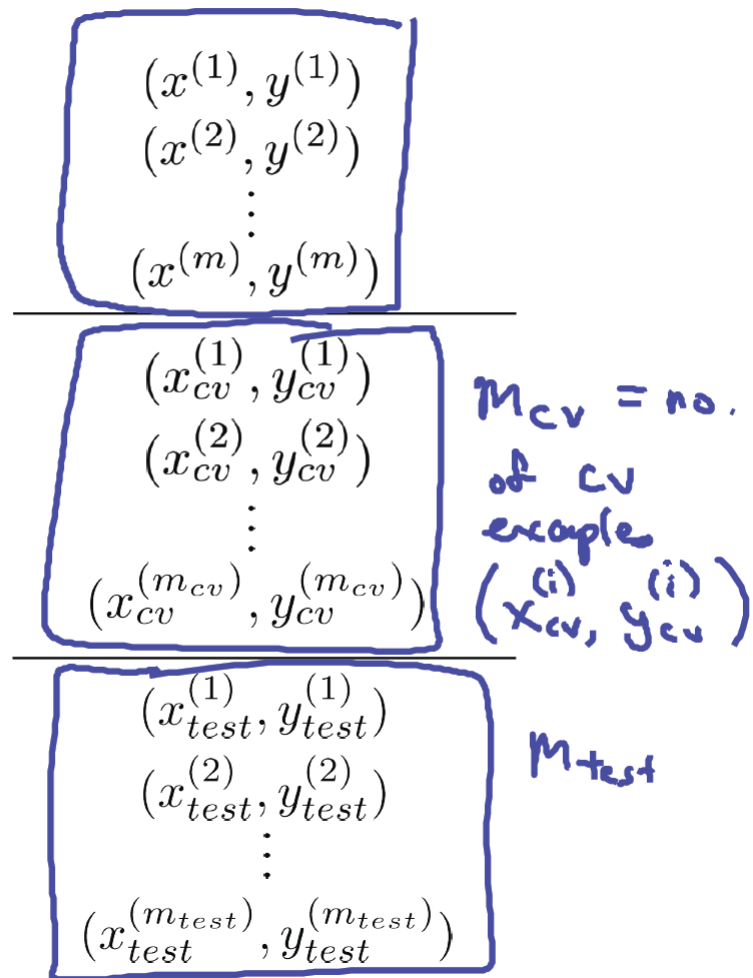


Illustration of Cross Validation

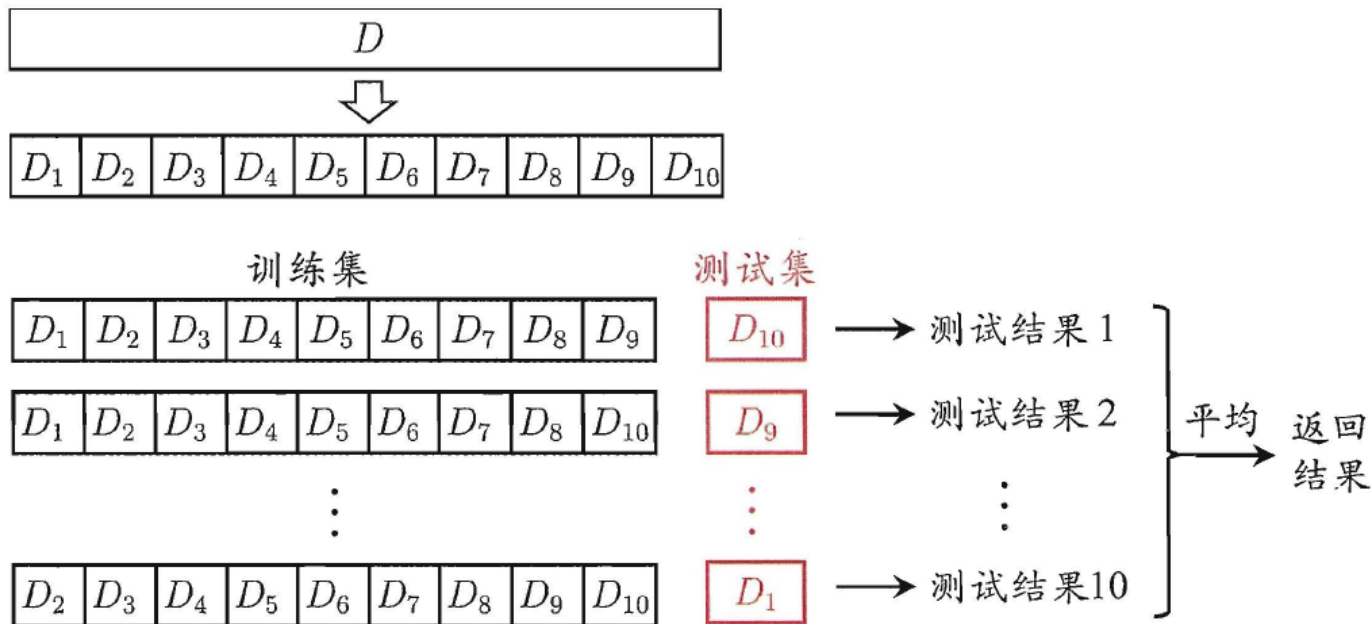


图 2.2 10 折交叉验证示意图

An Implementation of 10-fold Cross-Validation

for i = 1 : loop↓

read_begin = round((i - 1) * rows / loop) + 1;↓

read_end = round(i * rows / loop);↓

↓

tmp_buffer = g_p_network(read_begin : read_end, :);↓

g_p_network(read_begin : read_end, :) = 0;↓

↓

tmpR = birw_mn(phenotype_logistic, ppi_network, g_p_network, m, n, alpha,

↓

R(read_begin:read_end, :) = tmpR(read_begin:read_end, :);↓

↓

g_p_network(read_begin : read_end, :) = tmp_buffer;↓

↓

end↓

Leave One Out Cross-Validation

- 是N-fold Cross-Validation的特例，每个样本都当作一份验证数据集
- 对于耗时不是很长的，建议使用LOOCV

Conclusion

- 机器学习回顾
- 机器学习预测结果的评估
 - 准确率、查准率、查全率、 F_1 Measure、ROC和AUC
 - 其他：调参、代价敏感错误率、代价敏感分类器
- 显著性检验
- 交叉验证