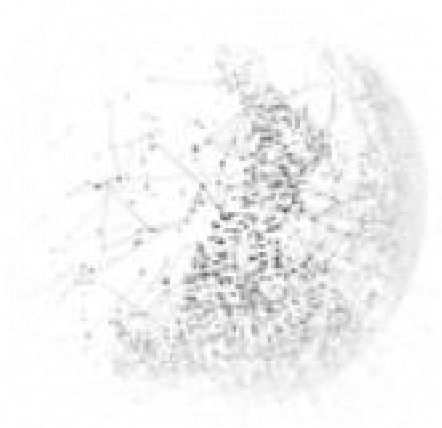


# 八、集成学习

主讲教师：周志华

机器学习导论

# 集成学习

**Ensemble Learning (集成学习):**
Using multiple learners to solve the problem


**Demonstrated great performance in real practice**

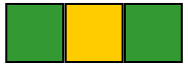
- KDDCup'07: 1st place for "… Decision Forests and …"
- KDDCup'08: 1st place of Challenge1 for a method using Bagging; 1st place of Challenge2 for "… Using an Ensemble Method"
- KDDCup'09: 1st place of Fast Track for "Ensemble … "; 2nd place of Fast Track for "… bagging … boosting tree models …", 1st place of Slow Track for "Boosting … "; 2nd place of Slow Track for "Stochastic Gradient Boosting"
- KDDCup'10: 1st place for "… Classifier ensembling"; 2nd place for "… Gradient Boosting machines … "
- KDDCup'11: 1st place of Track 1 for "A Linear Ensemble … "; 2nd place of Track 1 for "Collaborative filtering Ensemble", 1st place of Track 2 for "Ensemble …"; 2nd place of Track 2 for "Linear combination of …"
- KDDCup'12: 1st place of Track 1 for "Combining… Additive Forest…"; 1st place of Track 2 for "A Two-stage Ensemble of…"
- KDDCup'13: 1st place of Track 1 for "Weighted Average Ensemble"; 2nd place of Track 1 for "Gradient Boosting Machine"; 1st place of Track 2 for "Ensemble the Predictions"
- KDDCup'14: 1st place for "ensemble of GBM, ExtraTrees, Random Forest…" and "the weighted average"; 2nd place for "use both R and Python GBMs"; 3rd place for "gradient boosting machines… random forests" and "the weighted average of…"
- KDDCup'15: 1st place for "Three-Stage Ensemble and Feature Engineering for MOOC Dropout Prediction"
- KDDCup'16: 1st place for "Gradient Boosting Decision Tree"; 2nd place for "Ensemble of Different Models for Final Prediction"
- KDDCup'17: 1st and 2nd place of Task 1 for "XGBoost"; 1st place of Task 2 for "XGBoost", 2nd place of Task 2 for "Weighted Average of Multiple Models"
- KDDCup'18: 1st place for "Gradient Boosting"; 2nd place for "Two-stage stacking"; 3rd place for "Weighted Average of Multiple Models"

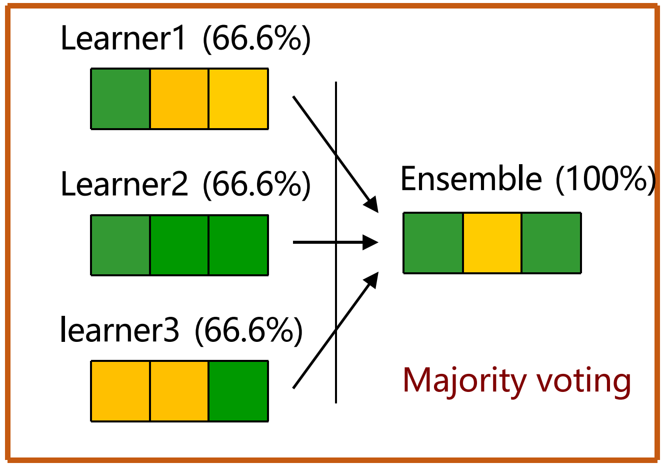
During the past decade, almost all winners of KDDCup, Netflix competition, Kaggle competitions, etc., utilized ensemble techniques in their solutions

**To win? Ensemble !**

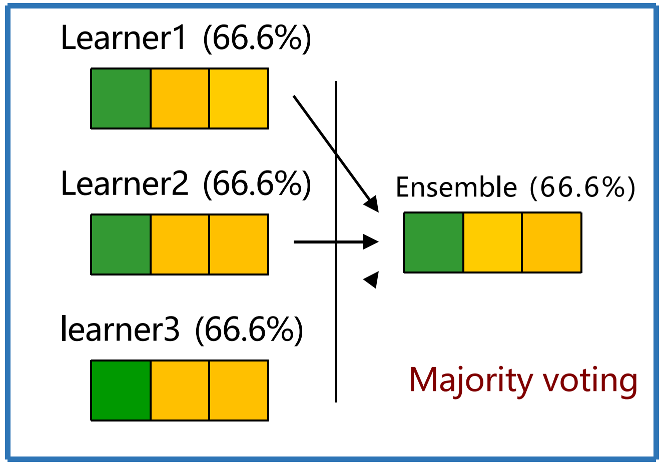
# 如何得到好的集成

**Some intuitions:**

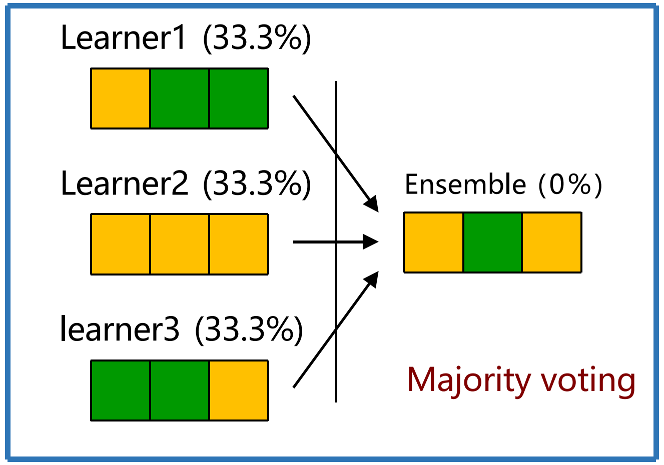

Ground-truth




**Ensemble really helps**




**Individuals must be different**




**Individuals must be not-bad**

令个体学习器 “好而不同”

# "多样性"（diversity）是关键

误差-分歧分解 (error-ambiguity decomposition):

$$\boxed{E} = \bar{E} - \bar{A}$$

Ensemble error　Ave. error of individuals　Ave. "ambiguity" of individuals　( "ambiguity" later called "diversity" )

The more **accurate** and **diverse** the individual learners,
the better the ensemble

[Krogh and Vedelsby, NIPS95]

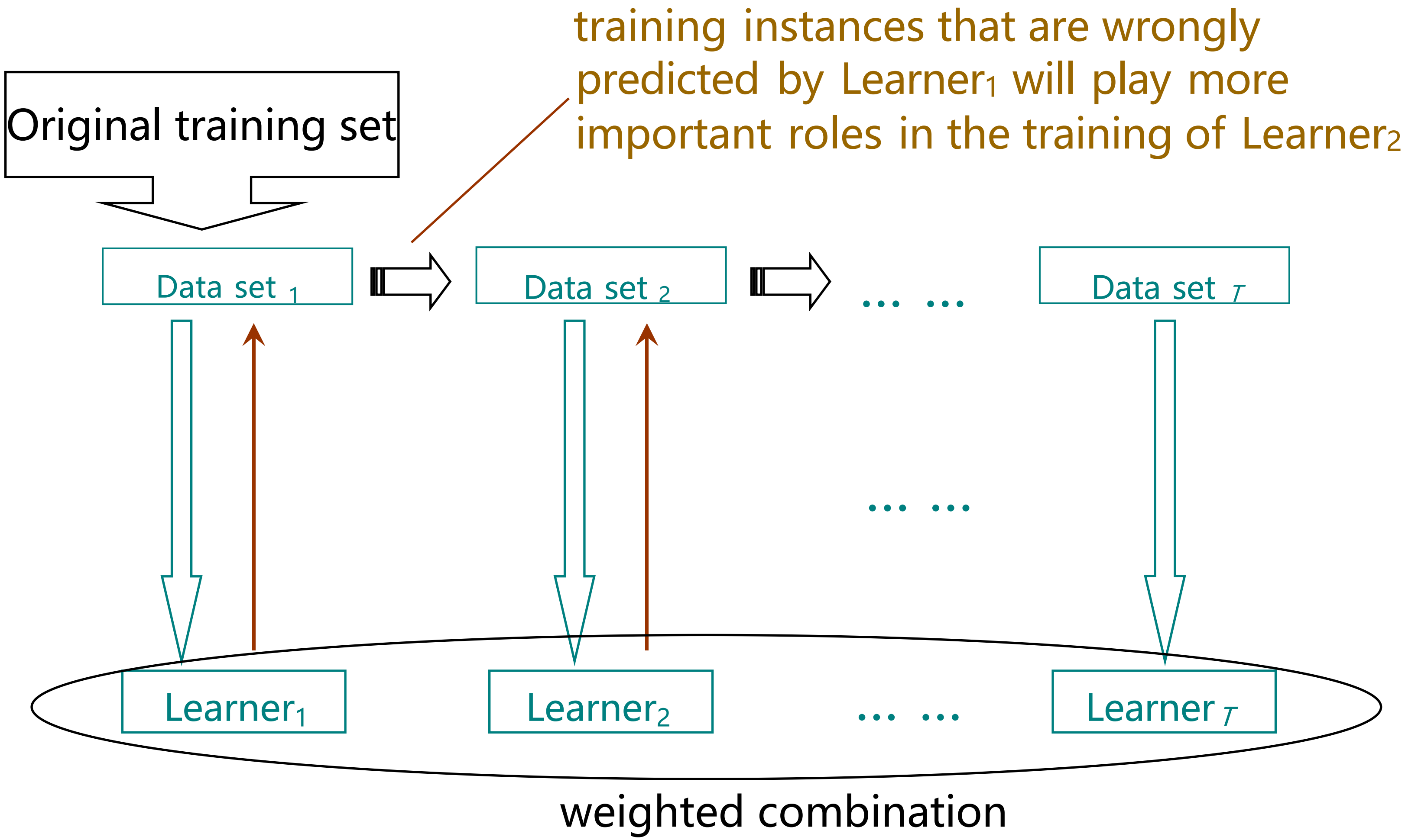
However,

- The "ambiguity" does not have an operable definition
- The error-ambiguity decomposition is derivable only for regression setting with squared loss

# 很多成功的集成学习方法

- 序列化方法
  - **AdaBoost** [Freund & Schapire, JCSS97]
  - GradientBoost [Friedman, AnnStat01]
  - LPBoost [Demiriz, Bennett, Shawe-Taylor, MLJ06]
  - … …
- 并行化方法
  - **Bagging** [Breiman, MLJ96]
  - Random Forest [Breiman, MLJ01]
  - Random Subspace [Ho, TPAMI98]
  - … …

# Boosting: A flowchart illustration



AdaBoost 算法步骤：二分类$y \in \mathcal{Y} = \{+1, -1\}$

1. 每个训练数据的权值为$w_i = \frac{1}{N}$, $D_1(i) = (w_1, \dots, w_N) = \left(\frac{1}{N}, \dots, \frac{1}{N}\right)$
2. 进行迭代$t = 1, \dots, T$

a) 选取一个当前误差最低的弱分类器作为第$t$个基本分类器$H_t$，计算在当前分布$D_t(i)$上的误差：

$$e_t = P(H_t(x_i) \neq y_i) = \sum_{i=1}^{N} w_{ti} I(H_t(x_i) \neq y_i)$$

a) 计算该弱分类器在最终分类器所占权重：$\alpha_t = \frac{1}{2}\ln\left(\frac{1-e_t}{e_t}\right)$

b) 更新训练样本的权值分布$D_{t+1}$：

$$D_{t+1}(i) = \frac{D_t(i)\exp(-\alpha_t y_i H_t(x_i))}{Z_t}, Z_t = \sum_{i=1}^{N} w_{ti}\exp(-\alpha_t y_i H_t(x_i))$$

$$Z_t = \sum_{i=1}^{N} w_{t,i} e^{-y_i \alpha_t h_t(x)} = \sum_{y_i = h_t(x_i)} w_{t,i} e^{-\alpha_t} + \sum_{y_i \neq h_t(x_i)} w_{t,i} e^{\alpha_t} = (1-e_t)e^{-\alpha_t} + e_t e^{\alpha_t} = 2\sqrt{(1-e_t)e_t}$$

3. 按照弱分类器权重$\alpha_t$组合各个弱分类器，即$f(x) = \sum_{t=1}^{T} \alpha_t H_t(x)$,通过$sign$函数得到强分类器

$$H_{final} = sign[f(x)] = sign\left[\sum_{t=1}^{T} \alpha_t H_t(x)\right]$$

PS: 因为权重更新依赖于权重$\alpha$与误差$e$，所以对样本权重更新公式：

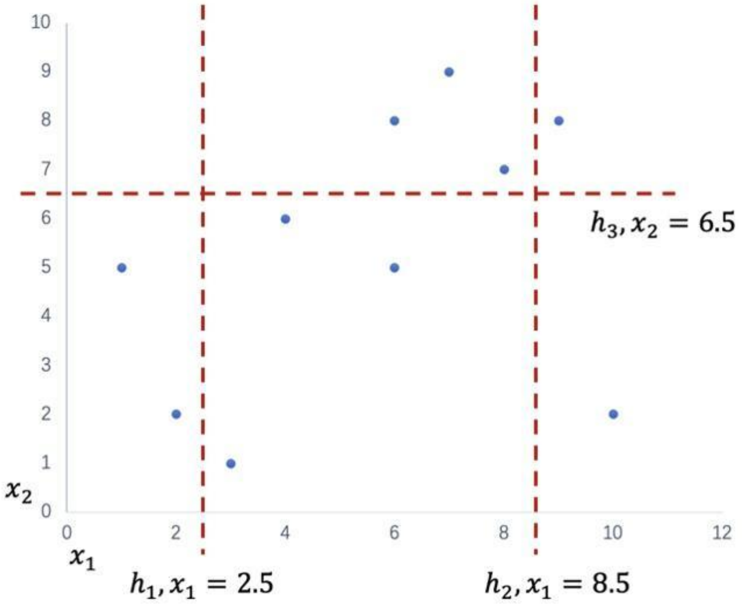
$$D_{t+1} = \frac{D_t(i)\exp(-\alpha_t y_i H_t(x_i))}{Z_t}, Z_t = 2\sqrt{e_t(1-e_t)}$$

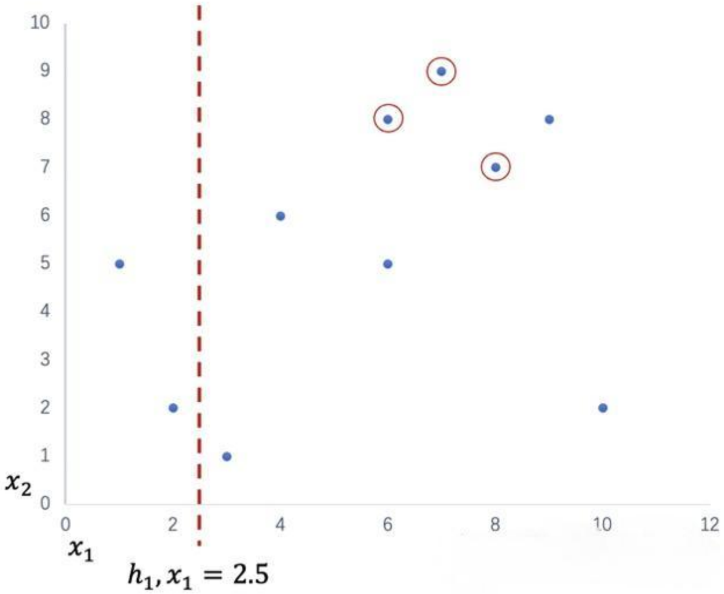
a) 当样本分错时，$y_i H_t(x_i) = -1$: $D_{t+1}(i) = \frac{D_t(i)\exp(-\alpha_t y_i H_t(x_i))}{Z_t} = \frac{D_t(i)}{Z_t}\sqrt{\frac{1-e_t}{e_t}} = \frac{D_t(i)}{2e_t}$

b) 当样本分对时，$y_i H_t(x_i) = 1$: $D_{t+1}(i) = \frac{D_t(i)\exp(-\alpha_t y_i H_t(x_i))}{Z_t} = \frac{D_t(i)}{Z_t}\sqrt{\frac{e_t}{1-e_t}} = \frac{D_t(i)}{2(1-e_t)}$

| 样本序号 | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
|---|---|---|---|---|---|---|---|---|---|---|
| 样本点$\boldsymbol{X}$ | (1,5) | (2,2) | (3,1) | (4,6) | (6,8) | (6,5) | (7,9) | (8,7) | (9,8) | (10,2) |
| 类别$\boldsymbol{Y}$ | 1 | 1 | -1 | -1 | 1 | -1 | 1 | 1 | -1 | -1 |
| 权值分布$\boldsymbol{D_1}$ | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 |

$$h_1(x)=\begin{cases}1, x_1<2.5\\ -1,\ \ x_1>2.5\end{cases};h_2(x)=\begin{cases}1, x_1<8.5\\ -1,\ \ x_1>8.5\end{cases};h_3(x)=\begin{cases}1, x_2>6.5\\ -1,\ \ x_2<6.5\end{cases}$$




在权值分布为 $D_1$ 时，不难验证三个弱分类器中，

$h_1$ 分类错5，7，8； $h_2$ 分类错3，4，6； $h_3$ 分类错1，2，9，

它们的分错误差为0.3

第一轮迭代， $t = 1$

那么在权值分布✦为 $D_1$ 时，三个弱分类器✦分错误差相等，我们选择 $h_1$ 分类误差最小故：

$$H_1(x) = h_1(x) \begin{cases} 1, & x_1 < 2.5 \\ -1, & x_1 \geq 2.5 \end{cases}$$

在分类器 $H_1$ 下的分类错误的事样本点5，7，8误差为： $e_1 = (0.1 + 0.1 + 0.1) = 0.3$

则 $H_1$ 在总分类器中的权重为： $\alpha_1 = \frac{1}{2} ln(\frac{1-e_1}{e_1}) = 0.4236$

得到强分类函数 $sign[f_1(x)] = sign[0.4236 H_1(x)]$

对分类正确训练样本1，2，3，4，6，9，10更新：
$D_2(i) = \frac{D_1(i)}{2(1-e_1)} = \frac{1}{10} \cdot \frac{1}{2 \cdot (1-0.3)} = \frac{1}{14}$

对于分类错误的训练样本：5，7，8更新: $D_2(i) = \frac{D_1(i)}{2e_1} = \frac{1}{10} \cdot \frac{1}{2 \cdot 0.3} = \frac{1}{6}$

因为 $sign[f_1(x)]$ 分类结果为：

| 样本序号 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| 权值分布$D_3$ | 1/22 | 1/22 | 1/6 | 1/6 | 1/66 | 1/6 | 7/66 | 7/66 | 1/22 | 1/22 |
| sign$[f_2(x)]$ | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 |

第二轮迭代，$t=2$

在权值为 $D_2$ 的情况下，三个弱分类器 $h_1, h_2, h_3$ 中，

$h_1$ 分类错5，7，8，分类误差：1/6+1/6+1/6=1/2；

$h_2$ 分类错3，4，6，分类误差：1/14+1/14+1/14=3/14

$h_3$ 分类错1，2，9，分类误差：1/14+1/14+1/14=3/14

我们取最小误差率中第二个基本分类器 $h_2$ 作为 $H_2(x)$ ，误差率：
$e_2=(1/14+1/14+1/14)=3/14$

则 $H_2(x)$ 在总分类器中的权重为： $\alpha_2=\frac{1}{2}ln(\frac{1-e_2}{e_2})=0.6496$ ，

得到强分类函数 $sign[f_2(x)]=sign[0.4236H_1(x)+0.6496H_2(x)]$

对分类正确训练样本1，2，5，7，8，9，10更新： $D_3(i)=\frac{D_2(i)}{2(1-e_2)}=\frac{7}{11}\cdot D_2(i)$

对于分类错误的训练样本：3，4，6更新： $D_3(i)=\frac{D_2(i)}{2e_2}=\frac{7}{3}\cdot D_2(i)$

| 样本序号 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| 权值分布$D_3$ | 1/22 | 1/22 | 1/6 | 1/6 | 1/66 | 1/6 | 7/66 | 7/66 | 1/22 | 1/22 |
| $sign[f_2(x)]$ | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 |

第三轮迭代✦， $t=3$

在权值为 $D_3$ 的情况下，三个弱分类器 $h_1, h_2, h_3$ 中，

$h_1$ 分类错5，7，8，分类误差：7/66+7/66+7/66=7/22

$h_2$ 分类错3，4，6，分类误差：1/6+1/6+1/6=1/2

$h_3$ 分类错1，2，9，分类误差：1/22+1/22+1/22=3/22

我们取最小误差率中第三个基本分类器 $h_3$ 作为 $H_3(x)$ ，误差率✦：
$e_3 = (1/22+1/22+1/22) = 3/22$

则H_3 (x)在总分类器中的权重为： $\alpha_3 = \frac{1}{2} ln(\frac{1-e_3}{e_3}) = 0.9229$ ，

得到强分类函数
$sign[f_3(x)] = sign[0.4236H_1(x) + 0.6496H_2(x) + 0.9229H_3(x)]$

对分类正确训练样本3，4，5，6，7，8，10更新： $D_4(i) = \frac{D_3(i)}{2(1-e_3)} = \frac{11}{9} \cdot D_3(i)$

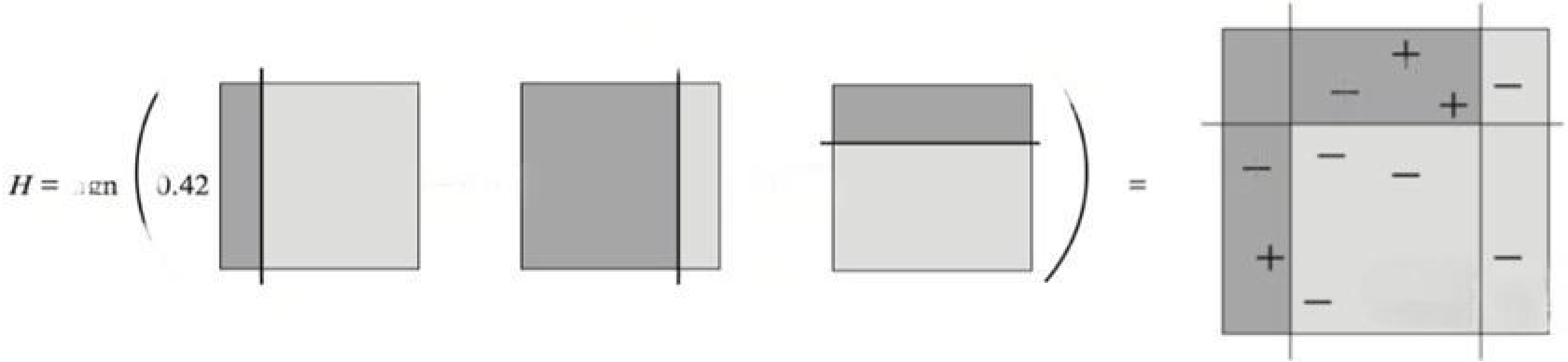
对于分类错误的训练样本：1，2，9更新: $D_4(i) = \frac{D_3(i)}{2e_3} = \frac{11}{3} \cdot D_3(i)$

| 样本序号 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| 权值分布 $D_4$ | 1/6 | 1/6 | 11/114 | 11/114 | 7/114 | 11/114 | 7/114 | 7/114 | 1/6 | 1/38 |
| $sign[f_3(x)]$ | 1 | 1 | -1 | -1 | 1 | -1 | 1 | 1 | -1 | -1 |

| 样本序号 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| 样本点***X*** | (1,5) | (2,2) | (3,1) | (4,6) | (6,8) | (6,5) | (7,9) | (8,7) | (9,8) | (10,2) |
| 类别***Y*** | 1 | 1 | -1 | -1 | 1 | -1 | 1 | 1 | -1 | -1 |
| 权值分布$D_1$ | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 |
| 权值分布$D_2$ | 1/14 | 1/14 | 1/14 | 1/14 | 1/6 | 1/14 | 1/6 | 1/6 | 1/14 | 1/14 |
| sign$[f_1(x)]$ | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 |
| 权值分布$D_3$ | 1/22 | 1/22 | 1/6 | 1/6 | 1/66 | 1/6 | 7/66 | 7/66 | 1/22 | 1/22 |
| sign$[f_2(x)]$ | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 |
| 权值分布$D_4$ | 1/6 | 1/6 | 11/114 | 11/114 | 7/114 | 11/114 | 7/114 | 7/114 | 1/6 | 1/38 |
| sign$[f_3(x)]$ | 1 | 1 | -1 | -1 | 1 | -1 | 1 | 1 | -1 | -1 |

通过线性组合 $h_1, h_2, h_3$

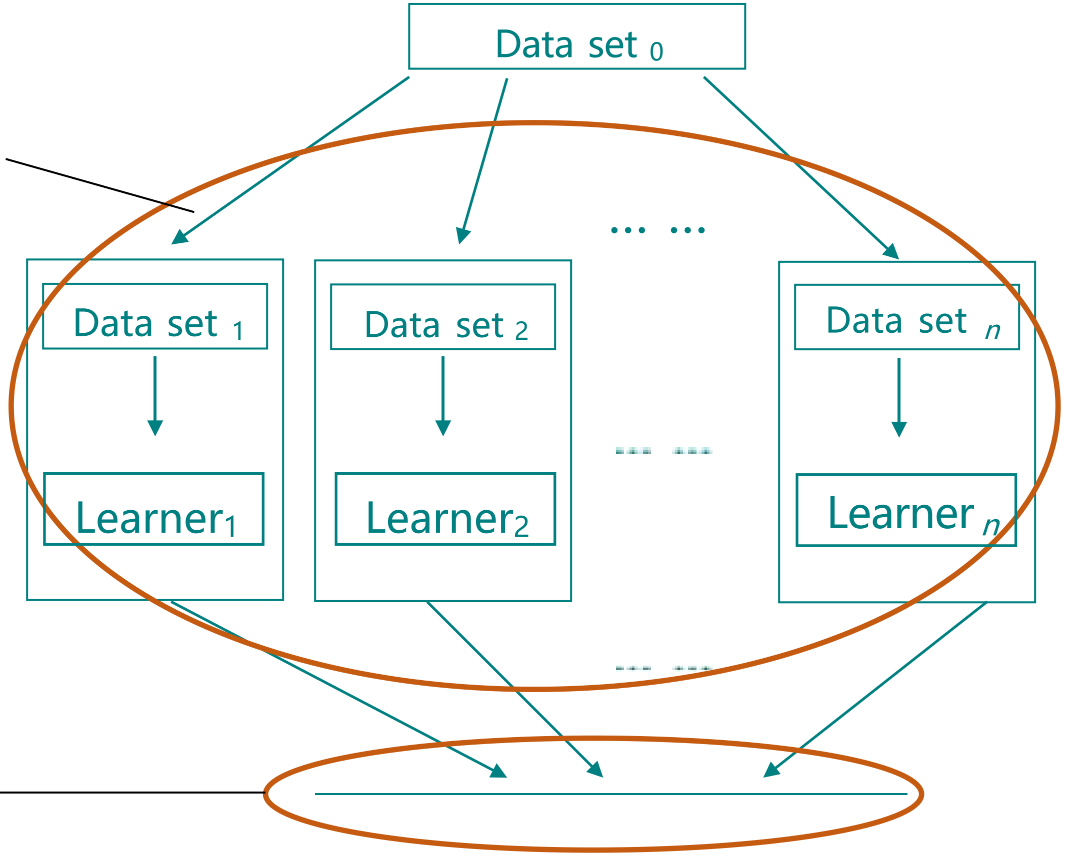
$sign[f_3(x)] = sign[0.423H_1(x) + 0.6496H_2(x) + 0.9229H_3(x)]$，此时分类误差为0



# Bagging

bootstrap a set of learners

generate many data sets from the original data set through bootstrap sampling (random sampling with replacement), then train an individual learner per data set




voting for classification

the output is the class label receiving the most number of votes

averaging for regression

the output is the average output of the individual learners

# 学习器结合

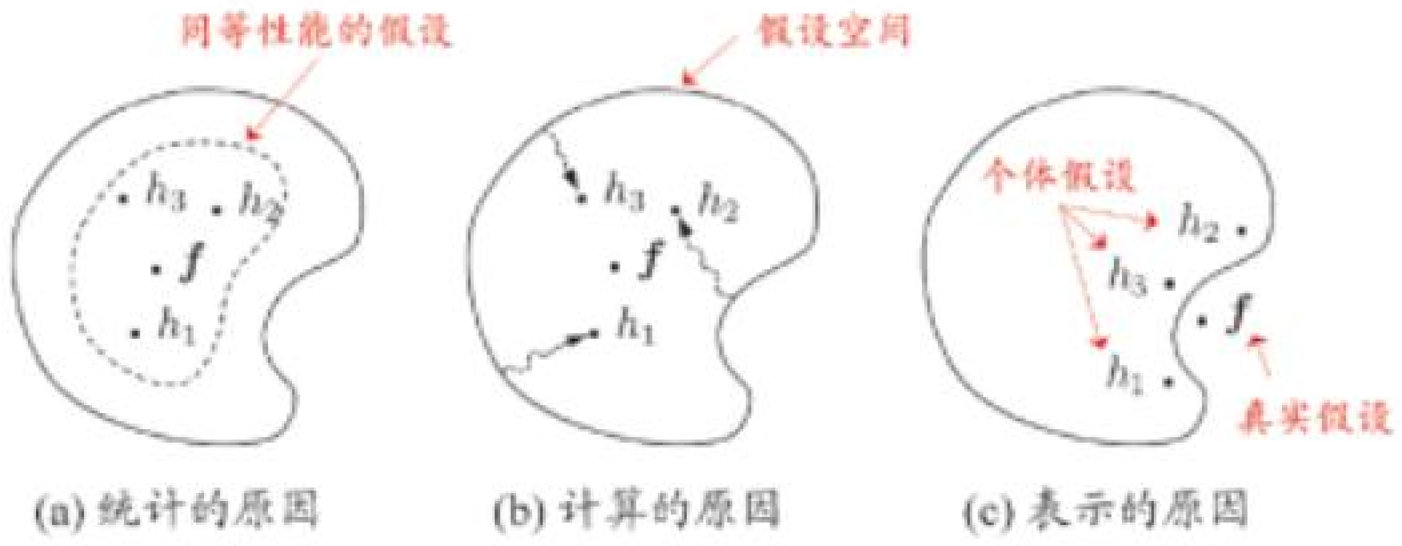



图 8.8 学习器结合可能从三个方面带来好处 [Dietterich, 2000]

常用结合方法：

- 投票法
  - 绝对多数投票法
  - 相对多数投票法
  - 加权投票法
- 平均法
  - 简单平均法
  - 加权平均法
- 学习法

# Stacking

**输入：** 训练集 $D=\{(\boldsymbol{x}_1,y_1),(\boldsymbol{x}_2,y_2),\ldots,(\boldsymbol{x}_m,y_m)\}$;
初级学习算法 $\mathfrak{L}_1,\mathfrak{L}_2,\ldots,\mathfrak{L}_T$;
次级学习算法 $\mathfrak{L}$.

**过程：**

```
for t = 1,2,...,T do
    h_t = 𝔏_t(D);
end for
D' = ∅;
for i = 1,2,...,m do
    for t = 1,2,...,T do
        z_it = h_t(x_i);
    end for
    D' = D' ∪ {((z_i1, z_i2, ..., z_iT), y_i)};
end for
h' = 𝔏(D');
```

使用初级学习算法 $\mathfrak{L}_t$ 产生初级学习器 $h_t$.

生成次级训练集.

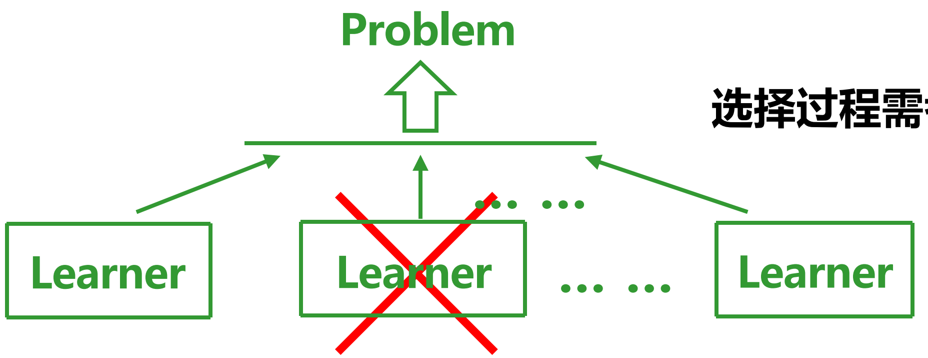
**输出：** $H(\boldsymbol{x})=h'(h_1(\boldsymbol{x}),h_2(\boldsymbol{x}),\ldots,h_T(\boldsymbol{x}))$

**图 8.9** Stacking 算法

## "越多越好"？

**选择性集成** (selective ensemble)：

给定一组个体学习器，从中选择一部分来构建集成，经常会比使用所有个体学习器更好 (更小的存储/时间开销，更强的泛化性能)




**选择过程需考虑个体 性能 与 多样性/互补性**

仅选出"精度最高的"通常不好!

集成修剪 (ensemble pruning) [Margineantu & Dietterich, ICML ' 97]较早出现，针对序列型集成

减小集成规模、降低泛化性能

选择性集成 [Zhou et al., AIJ' 02] 稍晚，针对并行型集成，MCBTA (Many could be better than all)定理

减小集成规模、增强泛化性能

目前"集成修剪"与"选择性集成"基本被视为同义词

# 多样性

"多样性"(diversity) 是集成学习的关键

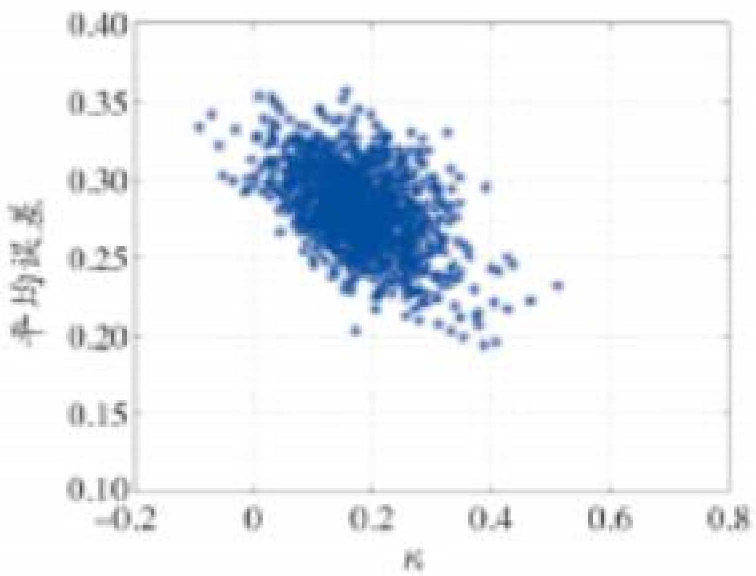
多样性度量

一般通过两分类器的预测结果列联表定义

| | $h_i = +1$ | $h_i = -1$ |
|---|---|---|
| $h_j = +1$ | $a$ | $c$ |
| $h_j = -1$ | $b$ | $d$ |

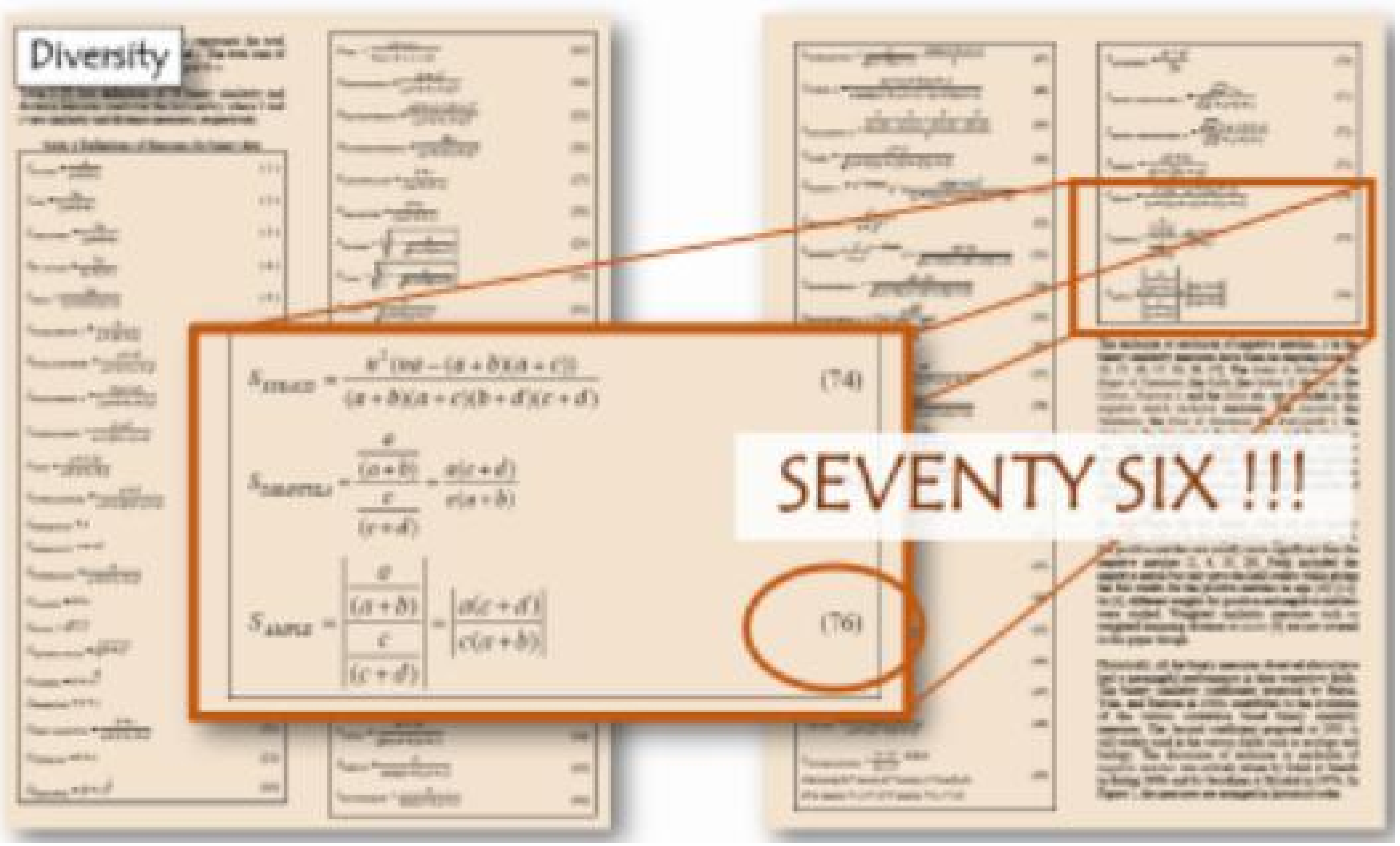
- 不合度量 (disagreement measure)
- 相关系数 (correlation coefficient)
- Q-统计量 (Q-statistic)
- r-统计量 (r-statistic)
- … …

r-误差图




每一对分类器作为图中的一个点

# 研究者提出了很多 Diversity measure




From [L. Kuncheva, ICPRAM’ 16 keynote]

# However,

...

- [Kuncheva & Whitaker, MLJ 2003]: Empirical study shows that there seems no clear relation between many diversity measures and the ensemble performance
- [Tang, Suganthan, Yao, MLJ 2006]: Exploiting many diversity measures explicitly is ineffective in constructing consistently stronger ensembles

**There is no well-accepted definition/formulation of diversity**

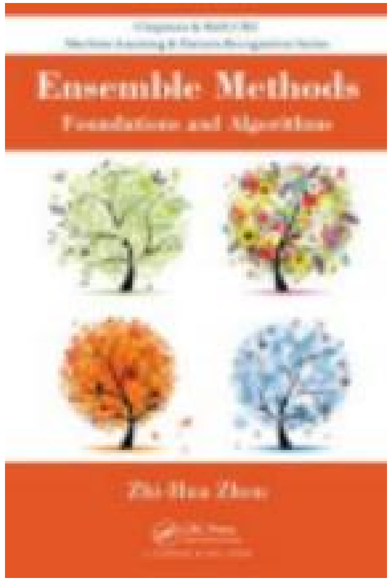
**" What is diversity " remains the holy grail problem of ensemble learning**

# 多样性增强常用策略

- 数据样本扰动
  - 例如 AdaBoost 使用 重要性采样、 Bagging 使用自助采样
  - 注意：对“不稳定基学习器” (如决策树、神经网络等) 很有效
    不适用于“稳定基学习器” (如线性分类器、SVM、朴素贝叶斯等)
- 输入属性扰动
  - 例如 随机子空间 (Random Subspace)
- 输出表示扰动
  - 例如 输出标记随机翻转、分类转回归、 ECOC
- 算法参数扰动

# 更多关于集成学习的内容，可参考：

Z.-H. Zhou.
Ensemble Methods: Foundations and Algorithms, Boca Raton, FL: Chapman & Hall/CRC, Jun. 2012.
(ISBN 978-1-439-830031)




# AdaBoost 算法综合应用题

## 题目背景

AdaBoost 是经典的集成学习算法，核心通过迭代训练弱分类器并加权组合得到强分类器。本题以一维特征二分类问题为例，要求基于 AdaBoost 算法（以**决策桩**为弱分类器，决策桩形式：$h(x)=\begin{cases}a, & x\le c\\ b, & x>c\end{cases}$，其中 $a,b\in\{+1,-1\}$，$c$为划分阈值）完成分类器构建。

| 样本序号 | 特征$x$ | 真实标签$y$（二分类） |
|---|---|---|
| 1 | 1 | -1 |
| 2 | 2 | +1 |
| 3 | 3 | -1 |
| 4 | 4 | +1 |

（1）初始化所有样本的权重；

（2）迭代训练 2 轮弱分类器，每轮完成：

① 确定本轮最优弱分类器（分类误差最小）；

② 计算该弱分类器的分类误差率$\varepsilon_m$；

③ 计算该弱分类器的权重$\alpha_m$；

④ 更新样本权重分布$D_m$；

（3）组合 2 个弱分类器，写出最终的 AdaBoost 强分类器表达式；

（4）验证最终分类器对 4 个样本的分类结果是否正确。

# 九、聚类

主讲教师：周志华

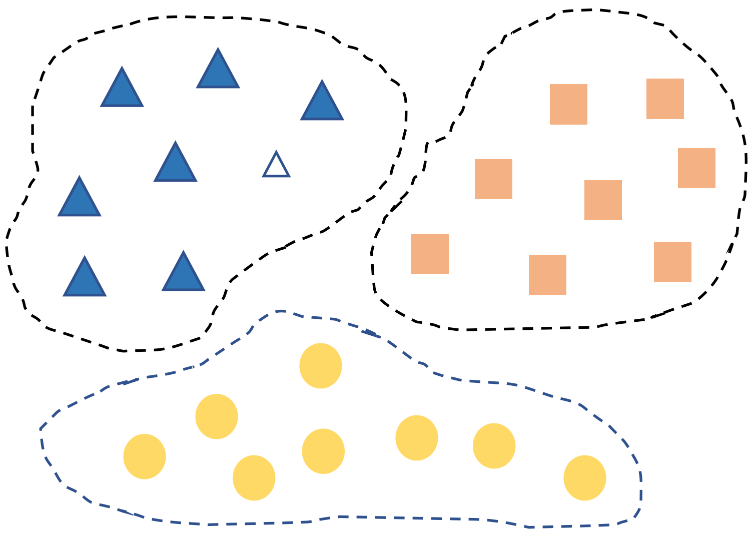
## 聚类 (Clustering)

在“无监督学习”任务中研究最多、应用最广

目标：将数据样本划分为若干个通常不相交的“簇”(cluster)

既可以作为一个单独过程（用于找寻数据内在的分布结构）

也可作为分类等其他学习任务的前驱过程



# 距离计算

- **距离度量 (distance metric) 需满足的基本性质：**

  非负性: dist(or,ar3)>0;

  同一性: $\mathrm{dist}(\boldsymbol{x}_i, \boldsymbol{x}_j) = 0$ 当且仅当 $\boldsymbol{x}_i = \boldsymbol{x}_j$ ;

  对称性:dist(ar,arj)=dist(r;,ac);

  直递性:dist(ars,arj)≤dist(as,ax)+dist(arr,3).

- **常用距离形式:**

闵可夫斯基距离 (Minkowski distance)

$$\mathrm{dist}_{\mathrm{mk}}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \left( \sum_{u=1}^{n} |x_{iu} - x_{ju}|^p \right)^{\frac{1}{p}}$$

$p = 2$: 欧氏距离(Euclidean distance)

$p = 1$: 曼哈顿距离(Manhattan distance)

# 距离计算

- **对无序(non-ordinal)属性，可使用 VDM (Value Difference Metric)**

令 表示属性 u 上取值为 a 的样本数，TTlu,a,i 表示在第 i 个样本簇中在属性 u 上取值为 a 的样本数，k 为样本簇数，则属性 u 上两个离散值 a 与 b 之间的 VDM 距离为

$$VDM_p(a,b) = \sum_{i=1}^{k} \left| \frac{m_{u,a,i}}{m_{u,a}} - \frac{m_{u,b,i}}{m_{u,b}} \right|^p$$

- **对混合属性，可使用 MinkovDM**

$$\mathrm{MinkovDM}_p(\boldsymbol{x}_i, \boldsymbol{x}_j) = \left( \sum_{u=1}^{n_c} |x_{iu} - x_{ju}|^p + \sum_{u=n_c+1}^{n} \mathrm{VDM}_p(x_{iu}, x_{ju}) \right)^{\frac{1}{p}}$$

# 必须记住

聚类的“好坏”不存在绝对标准

**The goodness of clustering depends on the opinion of the user.**

# 故事一则

聚类的故事：

老师拿来苹果和梨，让小朋友分成两份。

小明把大苹果大梨放一起，小个头的放一起，老师点头，恩，体量感。

小芳把红苹果挑出来，剩下的放一起，老师点头，颜色感。

小武的结果？不明白。小武掏出眼镜：最新款，能看到水果里有几个籽，左边这堆单数，右边双数。

老师很高兴：新的聚类算法诞生了。

**聚类也许是机器学习中“新算法”出现最多、最快的领域**
**总能找到一个新的“标准”，使以往算法对它无能为力**

# 常见聚类方法

## 原型聚类

- 亦称“基于原型的聚类”（prototype-based clustering)
- 假设：聚类结构能通过一组原型刻画
- 过程：先对原型初始化，然后对原型进行迭代更新求解
- 代表：k均值聚类，学习向量量化(LVQ)，高斯混合聚类

## 密度聚类

- 亦称“基于密度的聚类”（density-based clustering)
- 假设：聚类结构能通过样本分布的紧密程度确定
- 过程：从样本密度的角度来考察样本之间的可连接性，并基于可连接样本不断扩展聚类簇
- 代表： DBSCAN, OPTICS, DENCLUE

## 层次聚类 (hierarchical clustering)

- 假设：能够产生不同粒度的聚类结果
- 过程：在不同层次对数据集进行划分，从而形成树形的聚类结构
- 代表：AGNES (自底向上)， DIANA (自顶向下)