决策树、Adaboost

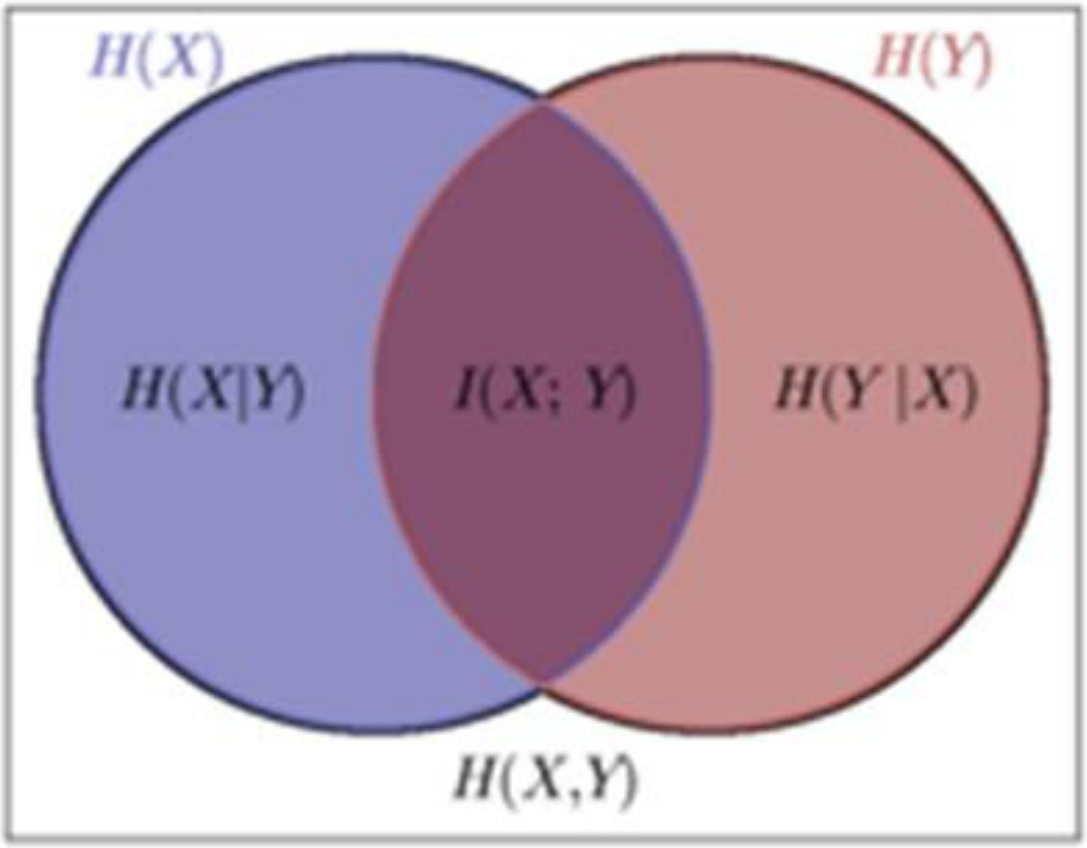
北京10月机器学习班 邹博

2014年11月1日

复习：熵

- sqrt(1-4x)<exp(-2x),0<x<1/4
- H(Y|X) = H(X,Y) - H(X)
  - 条件熵定义
- H(Y|X) = H(Y) - I(X,Y)
  - 根据互信息定义展开得到
  - 有些文献将I(X,Y)=H(Y) – H(Y|X)作为互信息的定义式
- 对偶式
  - H(X|Y)= H(X,Y) - H(Y)
  - H(X|Y)= H(X) - I(X,Y)
- I(X,Y)= H(X) + H(Y) - H(X,Y)
  - 有些文献将该式作为互信息的定义式
- 试证明：H(X|Y) ≤H(X)，　H(Y|X) ≤H(Y)

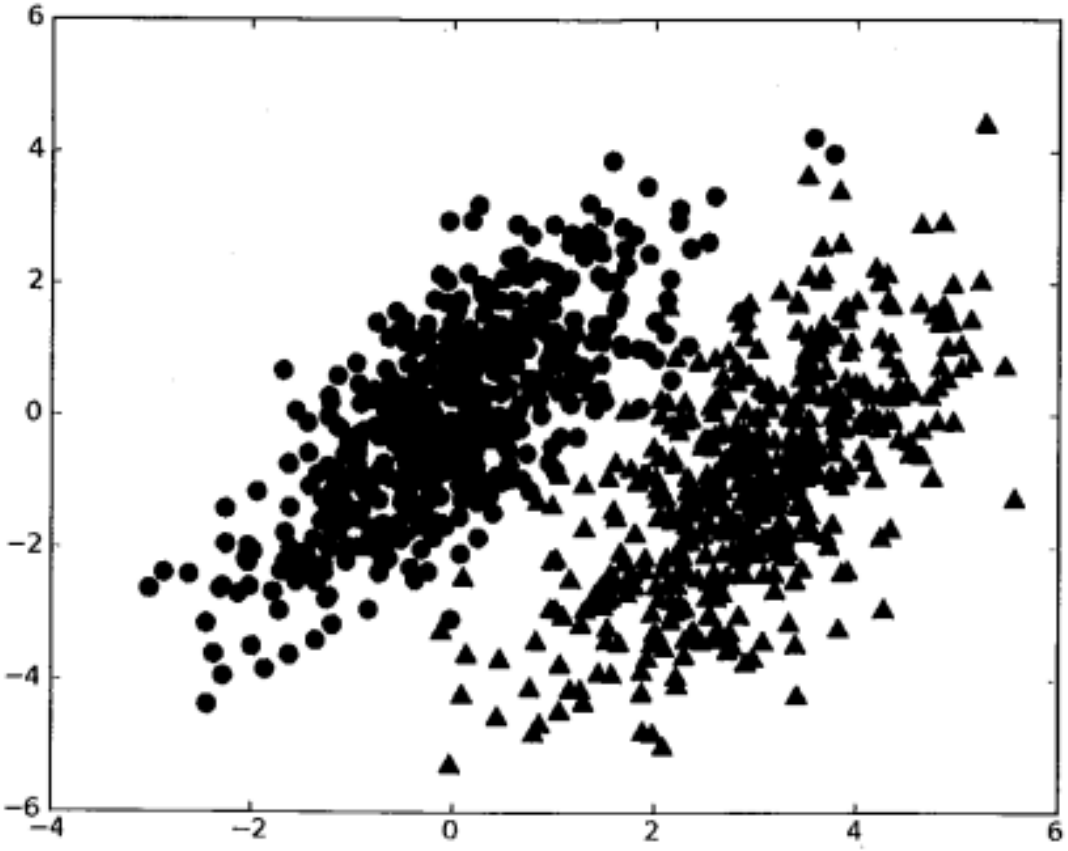
强大的Venn图：帮助记忆



等式变化

- 根据$H(Y|X) = H(Y) - I(X,Y)$
- 得到$I(X,Y) = H(Y) - H(Y|X)$
- $I(X,Y)$：在X中包含的关于Y的信息

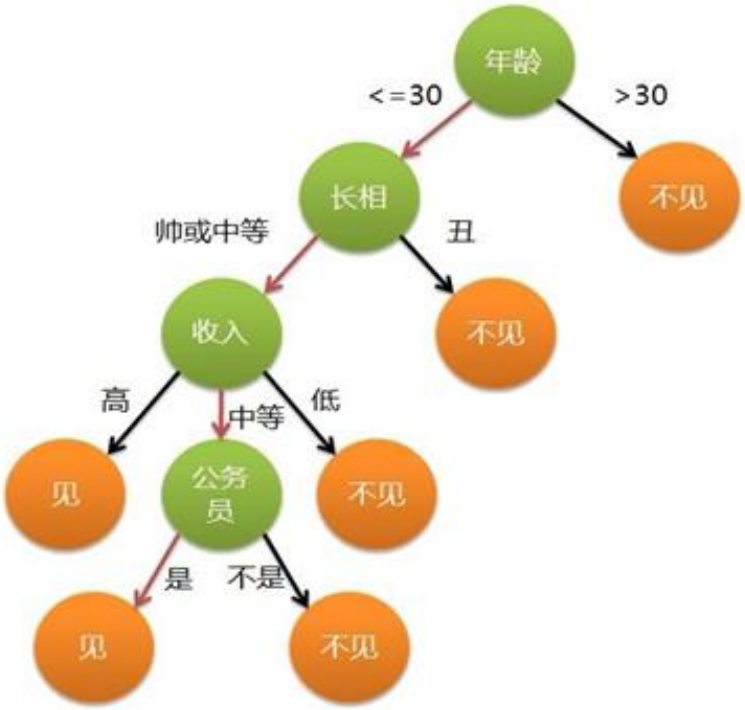
k近邻分类

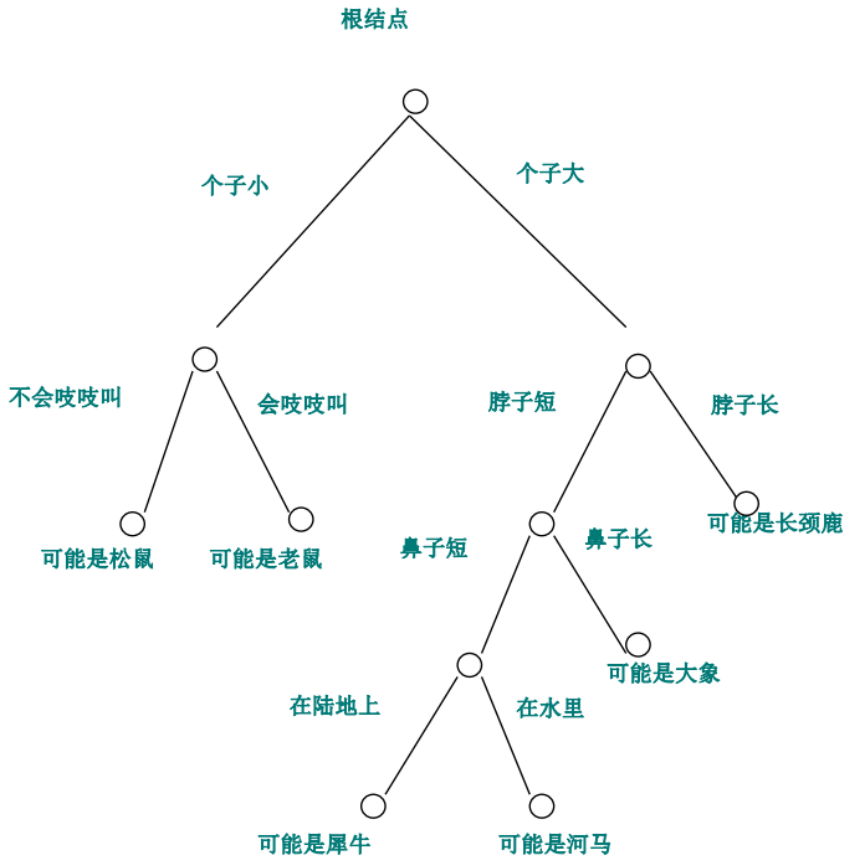


决策树(Decision Tree)

- 一种描述概念空间的有效的归纳推理办法。基于决策树的学习方法可以进行不相关的多概念学习，具有简单快捷的优势，已经在各个领域取得广泛应用。
- 决策树是一种树型结构，其中每个内部结点表示在一个属性上的测试，每个分支代表一个测试输出，每个叶结点代表一种类别。

决策树示意图


年龄
<=30
>30
长相
不见
帅或中等
丑
收入
不见
高
中等
低
见
公务员
不见
是
不是
见
不见


根结点
个子小
个子大
不会吱吱叫
会吱吱叫
脖子短
脖子长
可能是松鼠
可能是老鼠
鼻子短
鼻子长
可能是长颈鹿
可能是大象
在陆地上
在水里
可能是犀牛
可能是河马

决策树学习

- 决策树学习是以实例为基础的归纳学习。
- 决策树学习采用的是自顶向下的递归方法，其基本思想是以信息熵为度量构造一棵熵值下降最快的树，到叶子节点处的熵值为零，此时每个叶节点中的实例都属于同一类。

决策树学习算法的特点

- 决策树学习算法的最大优点是，它可以自学习。在学习的过程中，不需要使用者了解过多背景知识，只需要对训练例子进行较好的标注，就能够进行学习。
  - 从一类无序、无规则的事物（概念）中推理出决策树表示的分类规则。

决策树学习的生成算法

- ID3
- C4.5
- CART

信息增益

- 当熵和条件熵中的概率由数据估计（特别是极大似然估计）得到时，所对应的熵和条件熵分别称为经验熵和经验条件熵。
- 信息增益表示得知特征A的信息而使得类X的信息的不确定性减少的程度。
- 定义：特征A对训练数据集D的信息增益g(D,A)，定义为集合D的经验熵H(D)与特征A给定条件下D的经验条件熵H(D|A)之差，即：$g(D,A)=H(D)-H(D|A)$
  - 即训练数据集类别和特征的互信息。

基本记号

- 设训练数据集为D，|D|表示其容量，即样本个数。设有K个类Ck，k=1,2,…,K，|Ck|为属于类Ck的样本个数。$\Sigma_k|Ck|=|D|$。设特征A有n个不同的取值{a1,a2…an}，根据特征A的取值讲D划分为n个子集D1,D2,…Dn,|Di|为Di的样本个数，$\Sigma_i|Di|=D$。记子集Di中属于类Ck的样本的集合为D_{ik}，$|D_{ik}|$为D_{ik}的样本个数。

信息增益的计算方法

- 计算数据集D的经验熵 $H(D)=-\sum_{k=1}^{K}\frac{C_k}{D}\log\frac{C_k}{D}$
- 计算特征A对数据集D的经验条件熵H(D|A)
- 计算信息增益：g(D,A)=H(D) – H(D|A)

经验条件熵H(D|A)

$$H(D \mid A) = -\sum_{i,k} p(D_k, A_i) \log p(D_k \mid A_i)$$

$$= -\sum_{i,k} p(A_i) p(D_k \mid A_i) \log p(D_k \mid A_i)$$

$$= -\sum_{i=1}^{n} \sum_{k=1}^{K} p(A_i) p(D_k \mid A_i) \log p(D_k \mid A_i)$$

$$= -\sum_{i=1}^{n} p(A_i) \sum_{k=1}^{K} p(D_k \mid A_i) \log p(D_k \mid A_i)$$

$$= -\sum_{i=1}^{n} \frac{|D_i|}{|D|} \sum_{k=1}^{K} \frac{|D_{ik}|}{|D_i|} \log \frac{|D_{ik}|}{|D_i|}$$

其他目标

- 信息增益率：$g_r(D,A) = g(D,A) / H(A)$
- 基尼指数：

$$Gini(p) = \sum_{k=1}^{K} p_k (1 - p_k) = 1 - \sum_{k=1}^{K} p_k^2$$

$$= 1 - \sum_{k=1}^{K} \left(\frac{|C_k|}{D} \right)^2$$

讨论

- 考察基尼指数的图像、熵、分类误差率三者之间的关系
  - 使用1-x近似代替-lnx

三种决策树学习算法

- 适应信息增益来进行特征选择的决策树学习过程，即为ID3决策。
- 所以如果是取值更多的属性，更容易使得数据更“纯”（尤其是连续型数值），其信息增益更大，决策树会首先挑选这个属性作为树的顶点。结果训练出来的形状是一棵庞大且深度很浅的树，这样的划分是极为不合理的。
- C4.5：信息增益率
- CART：基尼系数
- 一个属性的信息增益越大，表明属性对样本的熵减少的能力更强，这个属性使得数据由不确定性变成确定性的能力越强。

提升方法

- 一个概念如果存在一个多项式的学习算法能够学习它，并且正确率很高，那么，这个概念是强可学习的；
- 一个概念如果存在一个多项式的学习算法能够学习它，并且学习的正确率仅比随机猜测略好，那么，这个概念是弱可学习的；
- 强可学习与弱可学习是等价的。
- 在学习中，如果已经发现了“弱学习算法”，能否将他提升为“强学习算法”。

Adaboost

- 设训练数据集T={(x1,y1), (x2,y2)…(xN,yN)}
- 初始化训练数据的权值分布

$$D_1 = (w_{11}, w_{12} \cdots w_{1i} \cdots, w_{1N}), \ w_{1i} = \frac{1}{N}, \ i = 1,2,\cdots, N$$

Adaboost：对于m=1,2,…M

- 使用具有权值分布Dm的训练数据集学习，得到基本分类器

$$G_m(x): \chi \to \{-1,+1\}$$

- 计算Gm(x)在训练数据集上的分类误差率

$$e_m = P(G_m(x_i) \neq y_i) = \sum_{i=1}^{N} w_{mi} I(G_m(x_i) \neq y_i)$$

- 计算Gm(x)的系数

$$\alpha_m = \frac{1}{2}\log\frac{1-e_m}{e_m}$$

Adaboost：对于m=1,2,…M

- 更新训练数据集的权值分布

$$D_{m+1} = (w_{m+1,1}, w_{m+1,2} \cdots w_{m+1,i} \cdots, w_{m+1,N}),$$

$$w_{m+1,i} = \frac{w_{mi}}{Z_m} \exp(-\alpha_m y_i G_m(x_i)), \quad i = 1,2,\cdots,N$$

- 这里，Zm是规范化因子

$$Z_m = \sum_{i=1}^{N} w_{mi} \exp(-\alpha_m y_i G_m(x_i))$$

- 它使D_{m+1}成为一个概率分布

Adaboost

- 构建基本分类器的线性组合

$$f(x)=\sum_{m=1}^{M}\alpha_m G_m(x)$$

- 得到最终分类器

$$G(x)=sign(f(x))=sign\left(\sum_{m=1}^{M}\alpha_m G_m(x)\right)$$

举例

- 给定下列训练样本，试用AdaBoost算法学习一个强分类器。

序号	1	2	3	4	5	6	7	8	9	X
X	0	1	2	3	4	5	6	7	8	9
Y	1	1	1	−1	−1	−1	1	1	1	−1

解

- 初始化训练数据的权值分布

$$D_1 = (w_{11}, w_{12} \cdots w_{1i} \cdots, w_{1N}),\ w_{1i} = \frac{1}{N},\ i = 1,2,\cdots,N$$

- W1i = 0.1

m=1

- 对于m=1
- 在权值分布为D1的训练数据上，阈值v取2.5时误差率最低，故基本分类器为：

$$G_1(x)=\begin{cases}1, & x<2.5\\ -1, & x>2.5\end{cases}$$

m=1

- G1(x)在训练数据集上的误差率
  e1=P(G1(xi)≠yi) = 0.3
- 计算G1的系数：

$$\alpha_1 = \frac{1}{2}\log\frac{1-e_1}{e_1} = 0.4236$$

m=1

- 更新训练数据的权值分布：

$$D_{m+1} = (w_{m+1,1}, w_{m+1,2} \cdots w_{m+1,i} \cdots, w_{m+1,N}),$$

$$w_{m+i} = \frac{w_{mi}}{Z_m} \exp(-\alpha_m y_i G_m(x_i)), \quad i = 1,2,\cdots,N$$

- D2=(0.0715, 0.0715, 0.0715, 0.0715, 0.0715, 0.0715, 0.1666, 0.1666, 0.1666, 0.0715)
- f1(x)=0.4236G1(x)
- 分类器sign(f1(x))在训练数据集上有3个误分类点。

m=2

- 对于m=2
- 在权值分布为D2的训练数据上，阈值v取8.5时误差率最低，故基本分类器为：

$$G_2(x)=\begin{cases}1, & x<8.5\\ -1, & x>8.5\end{cases}$$

m=2

- G2(x)在训练数据集上的误差率
  e2=P(G2(xi)≠yi) = 0.2143
- 计算G2的系数：

$$\alpha_2 = \frac{1}{2}\log\frac{1-e_2}{e_2} = 0.6496$$

m=2

- 更新训练数据的权值分布：

$$D_{m+1} = (w_{m+1,1}, w_{m+1,2} \cdots w_{m+1,i} \cdots, w_{m+1,N}),$$

$$w_{m+i} = \frac{w_{mi}}{Z_m} \exp(-\alpha_m y_i G_m(x_i)), \quad i = 1,2,\cdots,N$$

- D3=(0.0455, 0.0455, 0.0455, 0.1667, 0.1667, 0.01667, 0.1060, 0.1060, 0.1060, 0.0455)
- f2(x)=0.4236G1(x) + 0.6496G2(x)
- 分类器sign(f2(x))在训练数据集上有3个误分类点。

m=3

- 对于m=3
- 在权值分布为D3的训练数据上，阈值v取5.5时误差率最低，故基本分类器为：

$$G_3(x)=\begin{cases}1, & x<5.5\\ -1, & x>5.5\end{cases}$$

m=3

- G3(x)在训练数据集上的误差率
  e3=P(G3(xi)≠yi) = 0.1820
- 计算G3的系数：

$$\alpha_3 = \frac{1}{2}\log\frac{1-e_3}{e_3} = 0.7514$$

m=3

- 更新训练数据的权值分布：

$$D_{m+1} = (w_{m+1,1}, w_{m+1,2} \cdots w_{m+1,i} \cdots, w_{m+1,N}),$$

$$w_{m+i} = \frac{w_{mi}}{Z_m} \exp(-\alpha_m y_i G_m(x_i)), \quad i = 1,2,\cdots,N$$

- D4=(0.125, 0.125, 0.125, 0.102, 0.102, 0.102, 0.065, 0.065, 0.065, 0.125)
- f3(x)=0.4236G1(x) + 0.6496G2(x)+0.7514G3(x)
- 分类器sign(f3(x))在训练数据集上有0个误分类点。

误差上限

$$\frac{1}{N}\sum_{i=1}^{N} I\left(G\left(x_i\right)\neq y_i\right)\leq \frac{1}{N}\sum_{i}\exp\left(-y_i f\left(x_i\right)\right)=\prod_{m} Z_m$$

- 当G(xi)≠yi时，yi*f(xi)<0，因而exp(-yi*f(xi))≥1，前半部分得证。

后半部分

$$\frac{1}{N}\sum_i \exp\left(-y_i f(x_i)\right)$$

$$= \frac{1}{N}\sum_i \exp\left(-\sum_{m=1}^{M}\alpha_m y_i G_m(x_i)\right)$$

$$= w_{1i}\sum_i \exp\left(-\sum_{m=1}^{M}\alpha_m y_i G_m(x_i)\right)$$

$$= w_{1i}\prod_{m=1}^{M}\exp\left(-\alpha_m y_i G_m(x_i)\right)$$

$$= Z_1\sum_i w_{2i}\prod_{m=2}^{M}\exp\left(-\alpha_m y_i G_m(x_i)\right)$$

$$= Z_1 Z_2\sum_i w_{3i}\prod_{m=3}^{M}\exp\left(-\alpha_m y_i G_m(x_i)\right)$$

$$= Z_1 Z_2 \cdots Z_{M-1}\sum_i w_{Mi}\exp\left(-\alpha_M y_i G_M(x_i)\right)$$

$$= \prod_{m=1}^{M} Z_m$$

$$w_{m+1,i} = \frac{w_{mi}}{Z_m}\exp\left(-\alpha_m y_i G_m(x_i)\right)$$

$$Z_m w_{m+1,i} = w_{mi}\exp\left(-\alpha_m y_i G_m(x_i)\right)$$

训练误差界

$$\prod_{m=1}^{M} Z_m = \prod_{m=1}^{M} \left(2\sqrt{e_m(1-e_m)}\right) = \prod_{m=1}^{M} \sqrt{\left(1-4\gamma_m^2\right)} \leq \exp\left(-2\sum_{m=1}^{M}\gamma_m^2\right)$$

$$\gamma_m = \frac{1}{2} - e_m$$

训练误差界

$$Z_m = \sum_{i=1}^{N} w_{mi} \exp\left(-\alpha_m y_i G_m\left(x_i\right)\right)$$

$$= \sum_{y_i = G_m\left(x_i\right)} w_{mi} e^{-\alpha_m} + \sum_{y_i \neq G_m\left(x_i\right)} w_{mi} e^{\alpha_m}$$

$$= \left(1 - e_m\right) e^{-\alpha_m} + e_m e^{\alpha_m}$$

$$= 2\sqrt{e_m\left(1 - e_m\right)}$$

$$= \sqrt{1 - 4\gamma_m^2}$$

$$\alpha_m = \frac{1}{2} \log \frac{1 - e_m}{e_m}$$

取γ1, γ2… 的最大值，记做γ

$$\frac{1}{N}\sum_{i=1}^{N} I\left(G\left(x_i\right) \neq y_i\right) \leq \exp\left(-2M\gamma^2\right)$$

总结

- AdaBoost的训练误差是以指数速率下降的
- AdaBoost算法不需要事先知道下界γ，AdaBoost具有自适应性，它能适应若分类器格子的训练误差率。（“适应”Adaptive的由来）

感谢大家！

恳请大家批评指正！