

##

## 1. ROC, KS, AUC


### 1. TP, FP, FN, TN

	预测1	预测0	合计
真实1	True Positive (TP)	False Negative (FN)	Actual Positive(TP+FN)
真实0	False Positive (FP)	True Negative(TN)	Actual Negative(FP+TN)
合计	Predicted Positive(TP+FP)	Predicted Negative(FN+TN)	TP+FP+FN+TN

- True Positive Rate (TPR) , 计算公式为 $TPR=TP/(TP+FN)$ ; 所有真实的“1”中, 有多少被模型成功选出
- False Positive Rate (FPR) , 计算公式为 $FPR=FP/(FP+TN)$ ; 所有真实的“0”中, 有多少被模型误判为1了;
- Precision= $TP/(TP+FP)$ , 或 $2TP/((TP+FN)+(TP+FP))$ 。所有判为1的用户, 判对的比例
- 好的模型: TPR尽量高而FPR尽量低

## 2. ROC

- ROC(Receiver Operating Characteristic Curve):接受者操作特征曲线。
- ROC曲线: 设定不同的阈值, 计算不同的点(FPR,TPR), 连成曲线
- ROC曲线确定阈值的方法:
  - 给出ROC曲线的拟合函数表达式, 然后计算出最优的阈值, 这个目前通过软件实现难度不大: 如何给出最优拟合函数, 计算数学上有很多方法;
  - 计算出 $\Delta TPR \approx \Delta FPR$ 的点即为最优的阈值;
  - 从业务上给出最优的阈值。

1560344119508

## 3. AUC

- AUC: ROC曲线下方的面积Area Under the ROC Curve, 简称为AUC。这是评价模型的另一个方法, AUC值越大, 说明模型的分辨效果越好
- gini系数: 在SAS的评分模型输出中, 常用来判断收入分配公平程度, 此时 $gini=2*AUC-1$

XGB中

```
double sum_pospair = 0.0;
double sum_npos = 0.0, sum_nneg = 0.0, buf_pos = 0.0, buf_neg = 0.0;
```

```

for (size_t j = 0; j < rec.size(); ++j) {
    const float wt = info.GetWeight(rec[j].second);
    const float ctr = info.labels[rec[j].second];
    // keep bucketing predictions in same bucket
    if (j != 0 && rec[j].first != rec[j - 1].first) { // 遍历所有的预测值
        sum_pospair += buf_neg * (sum_npos + buf_pos * 0.5); // 逐个梯形计算
        sum_npos += buf_pos;
        sum_nneg += buf_neg;
        buf_neg = buf_pos = 0.0f;
    }
    buf_pos += ctr * wt; // 累计加权TP
    buf_neg += (1.0f - ctr) * wt; // 累计加权FP
}
sum_pospair += buf_neg * (sum_npos + buf_pos * 0.5);
sum_npos += buf_pos;
sum_nneg += buf_neg;
// check weird conditions
utils::Check(sum_npos > 0.0 && sum_nneg > 0.0,
              "AUC: the dataset only contains pos or neg samples");
// this is the AUC
sum_auc += sum_pospair / (sum_npos * sum_nneg); // 计算AUC

```

R语言中的计算方法

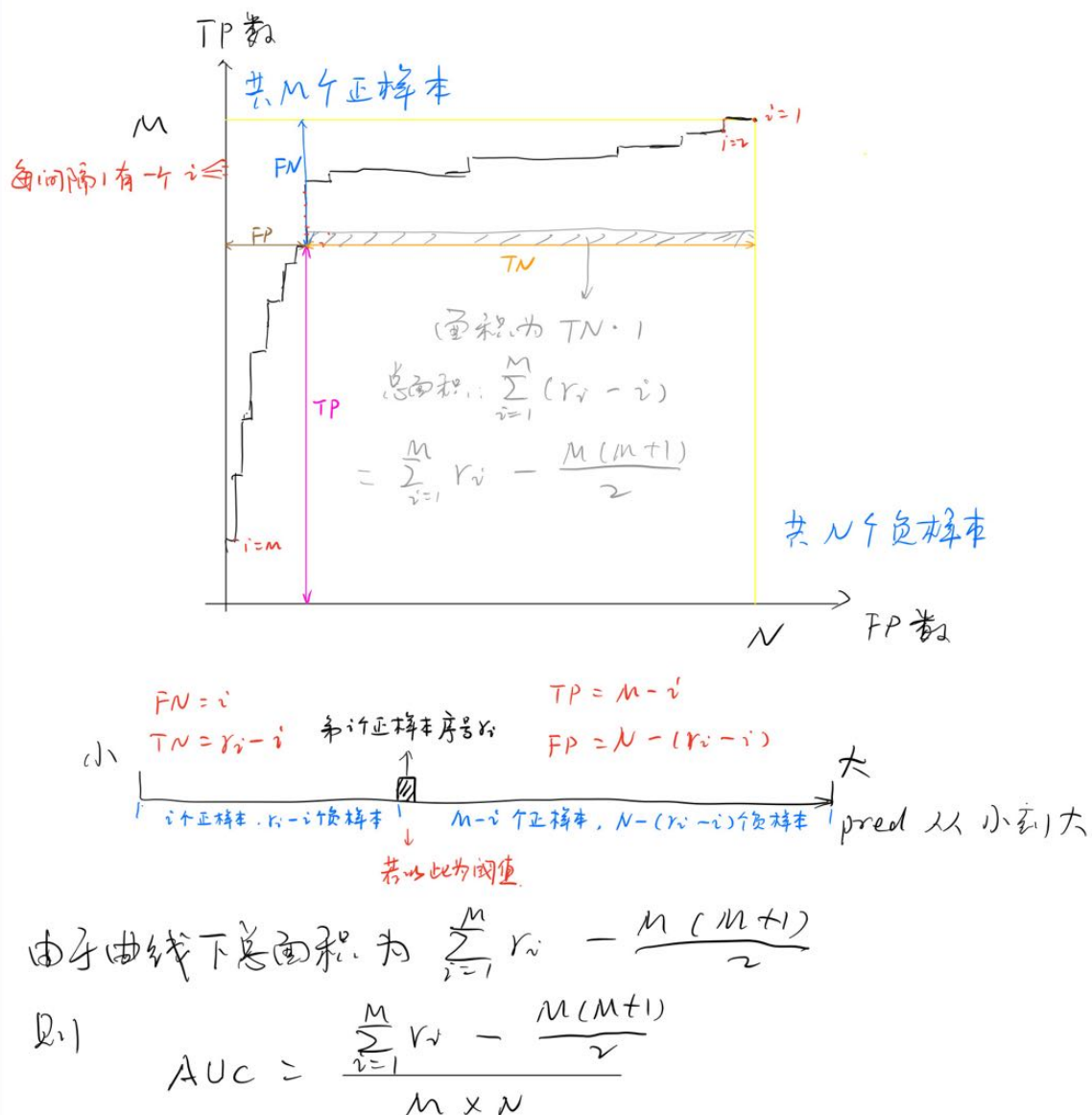
```

function (y_pred, y_true)
{
    rank <- rank(y_pred) # rank[i] 为 y_pred[i] 从小到大的排序号，最小为1,两个数并列第5，则
    都为5.5
    n_pos <- sum(y_true == 1)
    n_neg <- sum(y_true == 0)
    AUC <- (sum(rank[y_true == 1]) - n_pos * (n_pos + 1)/2)/(n_pos *
    n_neg)
    return(AUC)
}

```


$$AUC = \frac{\sum_{i \in \text{positiveClass}} rank_i - \frac{M(1+M)}{2}}{M \times N}$$

原因：



#### 4. KS

- K-S曲线：它和ROC曲线的画法异曲同工。以Logistic模型为例，首先把Logistic模型输出的概率从大到小排序，然后取10%的值（也就是概率值）作为阈值，同理把10%\*k (k=1,2,3,...,9) 处的值作为阈值，计算出不同的FPR和TPR值，以10%\*k (k=1,2,3,...,9) 为横坐标，分别以TPR和FPR的值为纵坐标，就可以画出两个曲线，这就是K-S曲线。
- KS值：KS=max(TPR-FPR)，即是两条曲线之间的最大间隔距离。当(TPR-FPR)最大时，也就是 $\Delta TPR - \Delta FPR = 0$ ，这和ROC曲线上找最优阈值的条件 $\Delta TPR = \Delta FPR$ 是一样的。从这点也可以看出，ROC曲线、K-S曲线、KS值的本质是相同的。

 1560344142780

- K-S曲线能直观地找出模型中差异最大的一个分段，比如评分模型就比较适合用KS值进行评估；
- KS值只能反映出哪个分段是区分度最大的，不能反映出所有分段的效果。

因此，在实际应用中，模型评价一般需要将ROC曲线、K-S曲线、KS值、AUC指标结合起来使用。

## 2. PSI

### 1. 含义

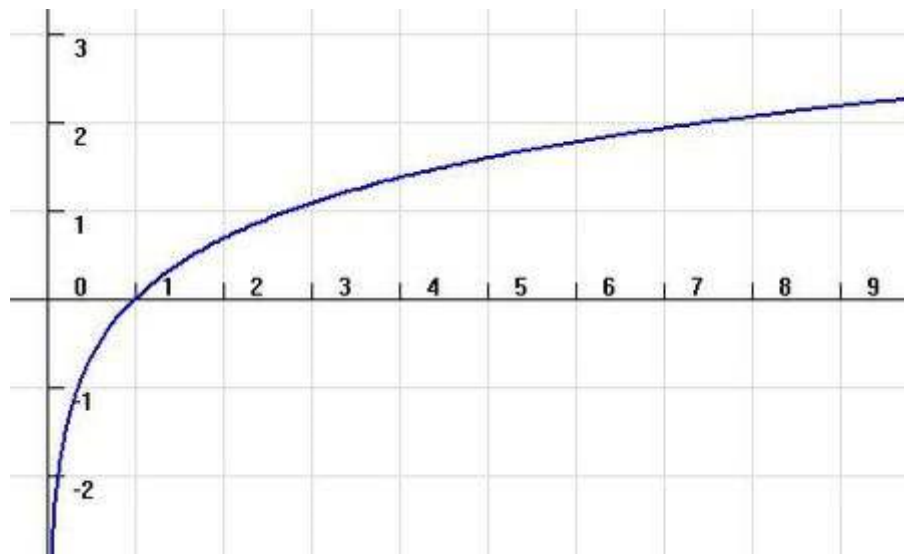
Population Stability Index (PSI) 群体稳定性指标

### 2. 公式

$$\text{psi} = \sum (\text{实际占比} - \text{预期占比}) * \ln(\text{实际占比} / \text{预期占比})$$

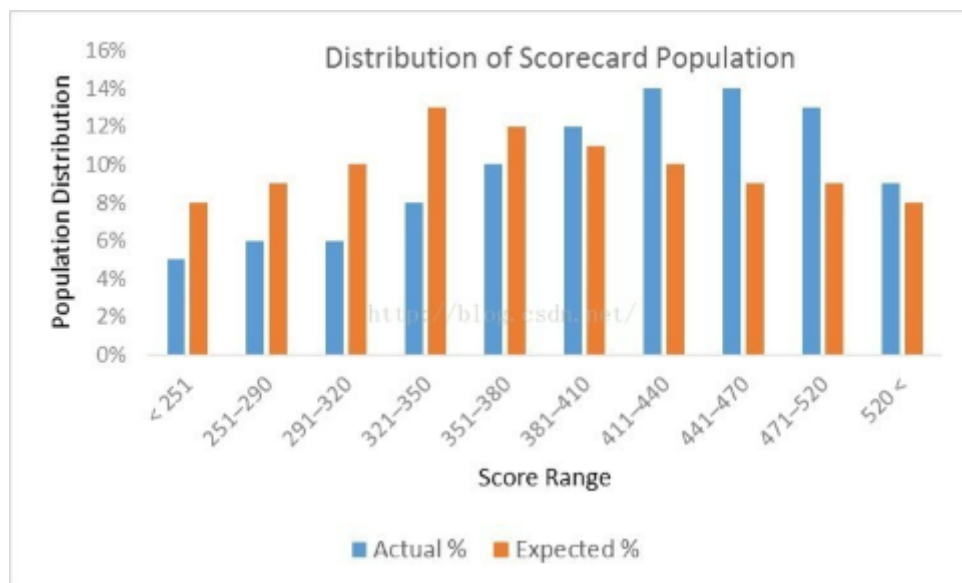
数学原理：

- 平衡符号
- 占比小的区间权重小



### 3. 计算

形式上比较像WoE和IV，下面是**计算**举例：



计算表：

Score bands	Actual %	Expected %	Ac-Ex	ln(Ac/Ex)	Index
< 251	5%	8%	-3%	-0.47	<b>0.014</b>
251–290	6%	9%	-3%	-0.41	<b>0.012</b>
291–320	6%	10%	-4%	-0.51	<b>0.020</b>
321–350	8%	13%	-5%	-0.49	<b>0.024</b>
351–380	10%	12%	-2%	-0.18	<b>0.004</b>
381–410	12%	11%	1%	0.09	<b>0.001</b>
411–440	14%	10%	4%	0.34	<b>0.013</b>
441–470	14%	9%	5%	0.44	<b>0.022</b>
471–520	13%	9%	4%	0.37	<b>0.015</b>
520 <	9%	8%	1%	0.12	<b>0.001</b>
<b>(PSI)=</b>					<b>0.1269</b>

指标取值解释说明：

PSI Value	Inference	Action
Less than 0.1	无关紧要的差距	不需要进一步操作
0.1 – 0.25	有一点差距	检查一下其他度量
Greater than 0.25	差距较大	需要进一步研究

## 4. 使用

## 3. VIF

### 1. 含义

方差膨胀因子 (Variance Inflation Factor, VIF)

- 容忍度的倒数，VIF越大，显示共线性越严重。经验判断方法表明：当 $0 < VIF < 10$ ，不存在[多重共线性](#)；当 $10 \leq VIF < 100$ ，存在较强的多重共线性；当 $VIF \geq 100$ ，存在严重多重共线性。

## 4. LIFT

### 1. 什么是LIFT

Lift是评估一个预测模型是否有效的一个度量；它衡量的是一个模型（或规则）对目标中“响应”的预测能力优于随机选择的倍数，以1为界线，大于1的Lift表示该模型或规则比随机选择捕捉了更多的“响应”，等于1的Lift表示该模型的表现独立于随机选择，小于1则表示该模型或规则比随机选择捕捉了更少的“响应”。维基百科中提升度被解释为“Target response divided by average response”。

### 2. 计算方法

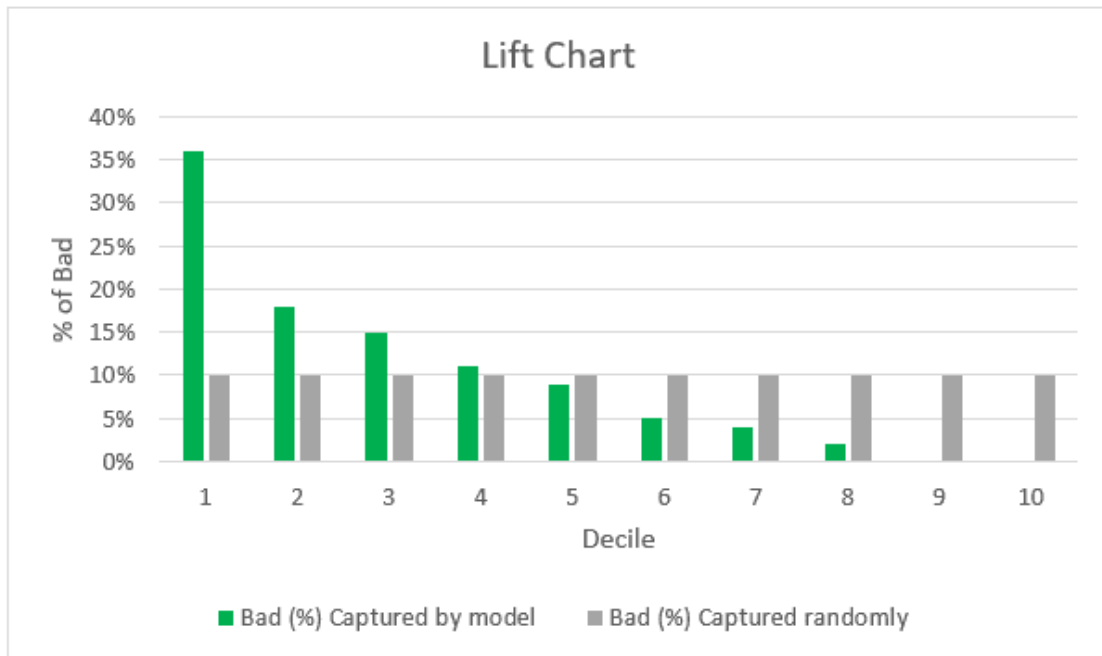
在模型评估中，我们常用到增益/提升（Gain/Lift）图来评估模型效果，其中的**Lift是“运用该模型”和“未运用该模型”所得结果的比值**。以信用评分卡模型的评分结果为例，我们通常会将打分后的样本按分数从低到高排序，取10或20等分（有同分数对应多条观测的情况，所以各组观测数未必完全相等），并对组内观测数与坏样本数进行统计。用评分卡模型捕捉到的坏客户的占比，可由该组坏样本数除以总的坏样本数计算得出；而不使用此评分卡，以随机选择的方法覆盖到的坏客户占比，等价于该组观测数占总观测数的比例（分子分母同时乘以样本整体的坏账率）。对两者取累计值，取其比值，则得到提升度Lift，即该评分卡抓取坏客户的能力是随机选择的多少倍。

### 3. 示例

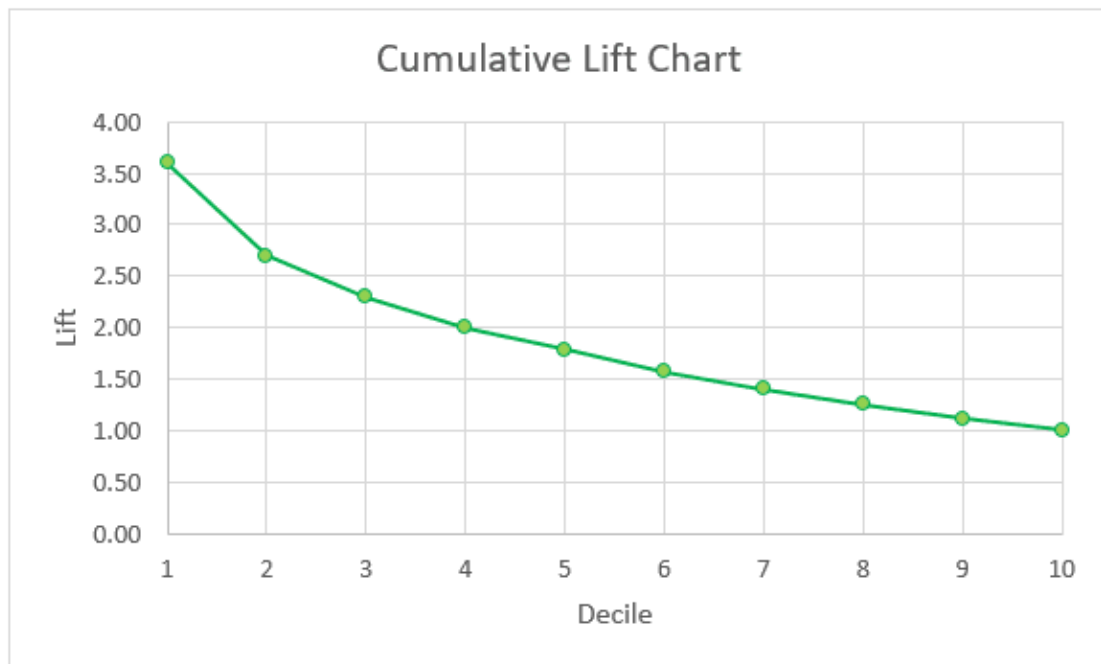
下表是一个提升表（Lift Table）的示例：

Decile	<u>Obs</u>	Bad	Bad (%) Captured by model	Bad (%) Captured randomly	Cumulative Bad (%) by model	Cumulative Bad (%) randomly	Lift
1	100	20	36%	10%	36%	10%	3.60
2	100	10	18%	10%	54%	20%	2.70
3	100	8	15%	10%	69%	30%	2.30
4	100	6	11%	10%	80%	40%	2.00
5	100	5	9%	10%	89%	50%	1.78
6	100	3	5%	10%	94%	60%	1.57
7	100	2	4%	10%	98%	70%	1.40
8	100	1	2%	10%	100%	80%	1.25
9	100	0	0%	10%	100%	90%	1.11
10	100	0	0%	10%	100%	100%	1.00
Total	1000	55					

以分数段为横轴，以捕捉到的“坏”占比为纵轴，可绘制出提升图，示例如下：



以分数段为横轴，以提升度为纵轴，可绘制出累计提升图，示例如下：



有了累计提升图，我们就能直观地去比较不同模型或策略给我们带来的区分能力增益程度。

## 5. RMSE, $R^2$

### 1. RMSE

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (X_{obs,i} - X_{model,i})^2}{n}}$$

## 2. $R^2$

$$R^2 = 1 - \frac{\sum (y - \hat{y})^2}{\sum (y - \bar{y})^2}$$

## 6. 变异系数

---

### 1. 概念

变异系数 (Coefficient of Variation)：当需要比较两组数据**离散程度**大小的时候，如果两组数据的测量尺度相差太大，或者数据**量纲**的不同，直接使用**标准差**来进行比较不合适，此时就应当消除测量尺度和量纲的影响，而变异系数可以做到这一点，它是原始数据标准差与原始数据**平均数**的比。CV没有量纲，这样就可以进行客观比较了。事实上，可以认为变异系数和极差、标准差和**方差**一样，都是反映数据离散程度的绝对值。其数据大小不仅受变量值离散程度的影响，而且还受变量值平均水平大小的影响。

### 2. 计算公式

标准差与平均值之比：

$$C_v = \frac{\sigma}{\mu}$$

## 7. WOE

---

### 1. 什么是WOE

### 2. 计算公式

WOE (Weight of Evidence)

某个变量第i个属性对应的WOE值计算公式如下：

$$\begin{aligned} WOE_i &= \ln\left(\frac{\text{好用户占比}}{\text{坏用户占比}}\right) \\ &= \ln\left(\frac{\frac{g_i}{g_T}}{\frac{b_i}{b_T}}\right) \\ &= \ln\left(\frac{g_i}{b_i}\right) - \ln\left(\frac{g_T}{b_T}\right) \end{aligned}$$

其中： $g_i$ 为第i个属性上好用户数， $g_T$ 表示总好人数， $b_i$ 为第i个属性上坏用户数， $b_T$ 表示总坏人数

**WOE**的值**越高**，代表着该分组中客户是坏客户的**风险越低**

## 8. IV

---

### 1. IV是什么



IV值是用来衡量某个变量对好坏客户区分能力的一个指标

## 2. 计算公式

IV值公式如下：

$$\begin{aligned} IV &= \sum_i \left( \frac{g_i}{g_T} - \frac{b_i}{b_T} \right) WOE_i \\ &= \sum_i \left( \frac{g_i}{g_T} - \frac{b_i}{b_T} \right) \ln \left( \frac{\frac{g_i}{g_T}}{\frac{b_i}{b_T}} \right) \\ &= \sum_i (P_g - P_b) \ln \left( \frac{P_g}{P_b} \right) \end{aligned}$$

$P_g$ 表示如果我是个好用户，我属于第*i*个属性的概率

$$P_g = P(x \in i | x \in g) = \frac{g_i}{g_T}$$

## 3. 取值经验

KL散度与IV见 九-4

# 9. KL散度

## 1. 什么是KL散度

在概率论或信息论中，KL散度(Kullback-Leibler divergence)，又称相对熵 (relative entropy)，是描述两个概率分布P和Q差异的一种方法。它是非对称的，这意味着 $D(P||Q) \neq D(Q||P)$ 。特别的，在信息论中， $D(P||Q)$ 表示当用概率分布Q来拟合真实分布P时，产生的信息损耗，其中P表示真实分布，Q表示P的拟合分布。有人将KL散度称为KL距离，但事实上，KL散度并不满足距离的概念，应为：1) KL散度不是对称的；2) KL散度不满足三角不等式。

## 2. 计算公式

$$D(P||Q) = \sum_{i \in X} P(i) * \left[ \log \left( \frac{P(i)}{Q(i)} \right) \right]$$

$$D(P||Q) = \int_x P(x) * \left[ \log \left( \frac{P(x)}{Q(x)} \right) \right] dx$$

## 3. 信息论含义

KL散度在信息论中有自己明确的物理意义，它是用来度量使用基于Q分布的编码来编码来自P分布的样本平均所需的额外的Bit个数。而其在机器学习领域的物理意义则是用来度量两个函数的相似程度或者相近程度，在泛函分析中也被频繁地用到[2]。在香农信息论中，用基于P的编码去编码来自P的样本，其最优编码平均所需要的比特个数（即这个字符集的熵）为：

$$H(x) = \sum_{x \in X} \underbrace{P(x)}_{\text{P中各字符出现的频率}} * \underbrace{\log\left(\frac{1}{P(x)}\right)}_{\text{P中该字符对应的编码长度}}$$

#### 4. KL散度与IV

$$\begin{aligned} IV &= \sum_i (P_g - P_b) \ln\left(\frac{P_g}{P_b}\right) \\ &= \sum_i P_g \ln\left(\frac{P_g}{P_b}\right) + \sum_i P_b \ln\left(\frac{P_b}{P_g}\right) \\ &= KL(P_g || P_b) + KL(P_b || P_g) \end{aligned}$$

即：好用户落在一个特征某个段上概率和坏用户落在这个段上的概率差别越大，IV值越大

即：好坏用户落在同一个属性上的概率越小（指 $P_g$ 和 $P_b$ 的分布差异越大）则IV值越大

## 10. F1-score

---

### 1. 混淆矩阵

### 2. 二级指标

## 11. AMS

---

<https://www.kaggle.com/c/higgs-boson/overview/evaluation>