

# 信息检索导论

## An Introduction to Information Retrieval

### 第10讲 概率检索模型

### Probabilistic Information Retrieval

授课人：林政

中国科学院信息工程研究所/国科大网络空间安全学院

# 提纲

- ① 上一讲及向量空间模型回顾
- ② 基本概率统计知识
- ③ Logistic回归模型
- ④ BIM模型
- ⑤ BM25模型

# 提纲

- ① 上一讲及向量空间模型回顾
- ② 基本概率统计知识
- ③ Logistic回归模型
- ④ BIM模型
- ⑤ BM25模型

# 上一讲内容

- 交互式相关反馈(Interactive relevance feedback): 在初始检索结果的基础上, 通过用户交互指定哪些文档相关或不相关, 然后改进检索的结果
- 最著名的相关反馈方法: Rocchio 相关反馈
- 查询扩展(Query expansion): 通过在查询中加入同义或者相关的词项来提高检索结果
  - 相关词项的来源: 人工编辑的同义词词典、自动构造的同义词词典、查询日志等等。

# 相关反馈的基本思想

- 用户提交一个(简短的)查询
- 搜索引擎返回一系列文档
- 用户将部分返回文档标记为相关的，将部分文档标记为不相关的
- 搜索引擎根据标记结果计算得到信息需求的一个新查询表示。当然我们希望该表示好于初始的查询表示
- 搜索引擎对新查询进行处理，返回新结果
- 新结果可望（理想上说）有更高的召回率

# Rocchio 1971 算法 (SMART系统使用)

实际中使用的公式:

$$\begin{aligned}\vec{q}_m &= \alpha \vec{q}_0 + \beta \mu(D_r) - \gamma \mu(D_{nr}) \\ &= \alpha \vec{q}_0 + \beta \frac{1}{|D_r|} \sum_{\vec{d}_j \in D_r} \vec{d}_j - \gamma \frac{1}{|D_{nr}|} \sum_{\vec{d}_j \in D_{nr}} \vec{d}_j\end{aligned}$$

$q_m$ : 修改后的查询;  $q_0$ : 原始查询;

$D_r$ 、 $D_{nr}$ : 已知的相关和不相关文档集合

$\alpha, \beta, \gamma$ : 权重

- 新查询向相关文档靠拢而远离非相关文档
- $\alpha$  vs.  $\beta/\gamma$  设置中的折中: 如果判定的文档数目很多, 那么  $\beta/\gamma$  可以考虑设置得大一些
- 一旦计算后出现负权重, 那么将负权重都设为0
- 在向量空间模型中, 权重为负是没有意义的。

# 伪相关反馈(Pseudo-relevance feedback)

- 伪相关反馈对于真实相关反馈的人工部分进行自动化
- 伪相关反馈算法
  - 对于用户查询返回有序的检索结果
  - 假定前  $k$  篇文档是相关的
  - 进行相关反馈 (如 Rocchio)

**优点：** 1. 不需要额外的训练数据     2. 平均效果不错

**缺点：** 多次循环迭代后有可能导致主题漂移，如果是有歧义的查询，漂移情况会更严重；伪相关文档的质量和数量影响检索性能。



从Top多少文档中抽取Top多少个词扩充查询，搜索引擎性能最佳？（需要实验验证）

# 查询扩展(Query expansion)

- **基于(用户)相关反馈的查询扩展**：用户参与反馈过程(显式/隐式/伪)，系统利用用户反馈结果对原始查询进行调整。
- **基于自动局部分析的查询扩展**：利用上次检索结果的前N个结果文档进行自动分析
- **基于自动全局分析的查询扩展**：利用全部的文档集合进行分析。
- **基于(外部)资源的方法**：基于已有的词典(wordnet、Hownet、同义词词林)、网络资源(wiki)、基于其他搜索引擎(如google)的结果、搜索日志(search log)等等



# 基于用户相关反馈的查询扩展

- 优点：
  - 为用户提供了方便反馈的接口，但是不参与查询扩展过程
  - 将整个搜索过程分成几个小步骤，每个步骤都很容易把握
  - 提供了一个受控过程来强调或者弱化某些term
- 缺点：
  - 主要的缺点是需要人工的参与，加重用户的负担。
- IR中集中研究不需要用户相关反馈而自动进行QE的方法。

# 基于自动局部分析的查询扩展

- 基于局部聚类(local clustering)的方法
  - 利用局部文档对term进行聚类，即将相关的term聚在一起，聚类的结果称为一个个簇(cluster)，于是利用簇中的相关term对查询 $q$ 进行扩展。
- 基于局部上下文分析(local context analysis, LCA)的方法
  - 局部聚类计算的是 $q$ 中每个term和所有term之间的相似度，应该将 $q$ 看成一个整体！LCA在局部文档中计算出和查询 $q$ 最相近的term进行扩展。

# 基于自动全局分析的查询扩展

---

- 传统全局分析利用所有文档集合来计算 $q$ 中每个term和所有term之间的相似度。直到1990年初，全局分析都被认为是非常失败的一种方法。
- 现代全局分析利用所有文档集合来计算查询 $q$ 和所有term之间的相似度。1990年以后，现代全局分析代替了传统全局分析，并表现出良好的效果。

# 基于外部资源的查询扩展

主要使用的信息：

- 同义词或近义词词典 ([thesaurus](#))
  - 人工构建
  - 自动构建
- 检索日志
  - 例 1: 提交查询 [herbs] (草药) 后, 用户常常搜索 [herbal remedies] (草本疗法)
    - “herbal remedies” 是 “herb” 的潜在扩展查询
  - 例 2: 用户搜索 [flower pix] 时常常点击URL [photobucket.com/flower](http://photobucket.com/flower), 而用户搜索 [flower clipart] 常常点击同样的URL
    - “flower clipart” 和 “flower pix” 可能互为扩展查询

# 向量空间模型回顾

---

- 文档表示成向量
- 查询也表示成向量
- 计算两个向量之间的相似度：余弦相似度、内积等等
- 在向量表示中的词项权重计算方法主要是tf-idf公式，实际考虑tf、idf及文档长度3个因素

# tf-idf权重计算的三要素

词项频率tf		文档频率df		归一化方法	
n(natural)	$tf_{t,d}$	n(no)	1	n(none)	1
l(logarithm)	$1 + \log(tf_{t,d})$	t(idf)	$\log \frac{N}{df_t}$	c(cosine)	$\frac{1}{\sqrt{w_1^2 + w_2^2 + \dots + w_M^2}}$
a(augmented)	$0.5 + \frac{0.5 \times tf_{t,d}}{\max_t(tf_{t,d})}$	p(prob idf)	$\max \left\{ 0, \log \frac{N - df_t}{df_t} \right\}$	u(pivoted unique)	$1/u$ (17.4.4节)
b(boolean)	$\begin{cases} 1 & \text{if } tf_{t,d} > 0 \\ 0 & \text{otherwise} \end{cases}$			b(byte size)	$1/CharLength^a, a < 1$
L(log ave)	$\frac{1 + \log(tf_{t,d})}{1 + \log(\text{ave}_{t \in d}(tf_{t,d}))}$				

- df本身往往较大，所以通常需要将它映射到一个较小的取值范围，可以定义成 $\log(N/df)$
- 计算过程中为了避免出现0值，需要平滑处理

# 向量空间模型的优缺点

- 优点：
  - 简洁直观，可以应用到很多其他领域(文本分类)。
  - 支持部分匹配和近似匹配，结果可以排序
  - 检索效果不错
- 缺点：
  - 理论上不够严谨，往往基于直觉的经验性公式
  - 词项之间的独立性假设与实际不符：实际上，词项的出现之间是有关系的，并不是完全独立的。如：“张继科”、“乒乓球”的出现不是独立的。

# 什么是概率检索模型

- 概率检索模型就是利用概率模型来估计每篇文档和查询的相关概率 $P(R=1|d,q)$ ，然后对结果进行排序。
- 布尔模型和向量空间模型可以给出文档内容和查询是否相关的推测，而概率论的方法可以给这种推测提供一个基本的理论。



# 本讲内容

---

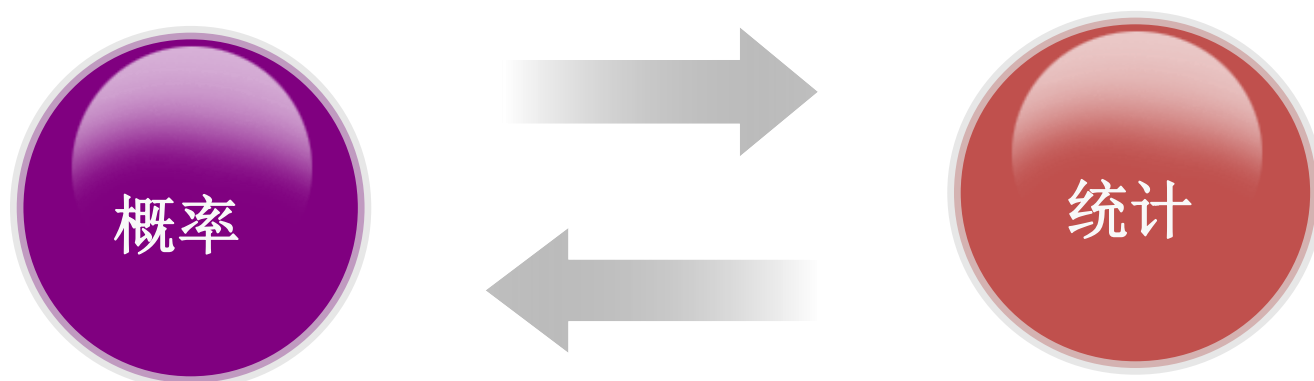
- 概率基础知识
- 基于概率理论的检索模型
- Logistic回归模型
- 二值独立概率模型 BIM：不考虑TF和文档长度
- 考虑TF和文档长度的BM25模型

# 提纲

- ① 上一讲及向量空间模型回顾
- ② 基本概率统计知识
- ③ Logistic回归模型
- ④ BIM模型
- ⑤ BM25模型

# 概率 vs. 统计

概率是统计的**理论基础**



统计是概率的**实际应用**

典型问题：已知某数据  
总体满足某分布，抽样得  
到某数据的概率是多少？

典型问题：已知某抽样数据  
(或总体分布)，判断总体的  
分布(或分布参数) 是多少？

# 概率统计初步

---

- 随机试验与随机事件
- 概率和条件概率
- 乘法公式、全概率公式、贝叶斯公式
- 随机变量
- 随机变量的分布

# 随机试验和随机事件

- **随机试验**：可在相同条件下重复进行；试验可能结果不止一个，但能确定所有的可能结果；一次试验之前无法确定具体是哪种结果出现。
  - 掷一颗骰子，考虑可能出现的点数
- **随机事件**：随机试验中可能出现或可能不出现的情况叫“随机事件”
  - 掷一颗骰子，4点朝上

# 概率和条件概率

- **概率**：直观上来看，事件A的概率是指事件A发生的可能性，记为 $P(A)$ 
  - 掷一颗骰子，出现6点的概率为多少？
- **条件概率**：已知事件A发生的条件下，事件B发生的概率称为A条件下B的条件概率，记作 $P(B|A)$ 
  - 现分别有甲、乙两个容器，在甲容器里分别有 7 个红球和 3 个白球，在乙容器里有 1 个红球和 9 个白球，现已知从这两个容器里任意抽出了一个红球，问这个球来自甲容器的概率是多少？

# 乘法公式、全概率公式和贝叶斯公式

- 乘法公式:

- $P(AB)=P(A)P(B|A)$

- $P(A_1A_2...A_n)=P(A_1)P(A_2|A_1)...P(A_n|A_1...A_{n-1})$

- 全概率公式:  $A_1A_2...A_n$ 是整个样本空间的一个划分

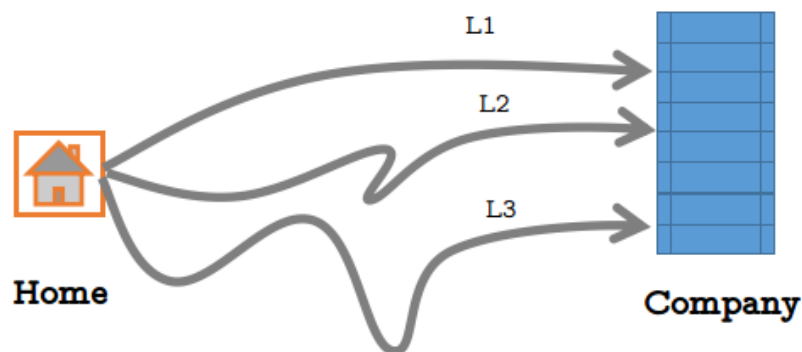
$$P(B)=\sum_{i=1}^n P(A_i)P(B|A_i)$$

- 贝叶斯公式:  $A_1A_2...A_n$ 是整个样本空间的一个划分

$$P(A_j|B)=\frac{P(A_j)P(B|A_j)}{\sum_{i=1}^n P(A_i)P(B|A_i)}, (j=1,...,n)$$

# 全概率公式举例

- 什么时候会用到全概率公式呢？比如达到某个目的，有多种方式，问达到目的的概率是多少？
- **举例：**小明从家到公司上班总共有三条路可以直达，但是每条路每天拥堵的可能性不太一样，由于路的远近不同，选择每条路的概率为0.5,0.3,0.2，每条路不堵车的概率为0.2, 0.4, 0.7,如果堵车会迟到，那不迟到概率是多少？



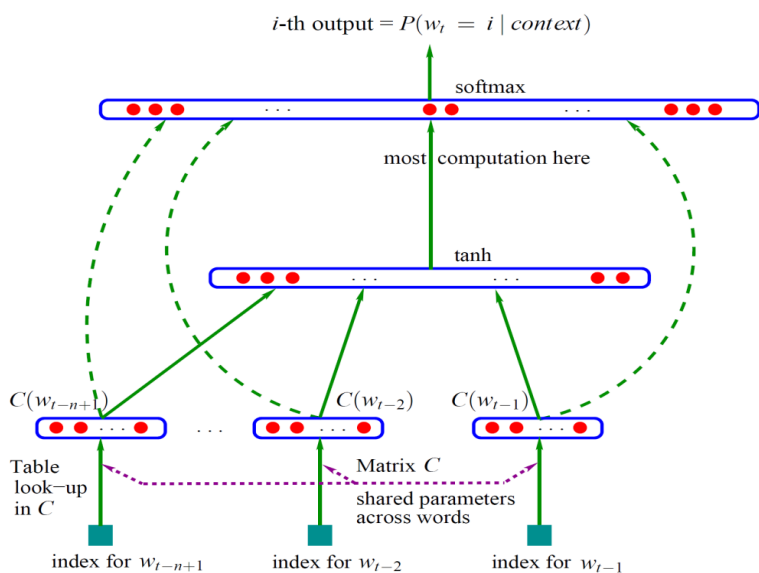


# 乘法公式举例

## n-gram语言模型

$$P(w_1, w_2, \dots, w_n) = P(w_1)P(w_2 | w_1)P(w_3 | w_1, w_2) \dots P(w_n | w_1, w_2, \dots, w_{n-1})$$

语言模型就是判断给定字符串为自然语言的概率 $P(w_1, w_2, \dots, w_n)$ ，其中 $w_1, w_2, \dots, w_n$ 依次表示字符串中的各个词。如果 $P$ 大于某个阈值，就认为该字符串为自然语言。



从下往上依次为输入层、隐藏层和输出层。

- (1) 首先根据训练集生成词典 $D$ ;
- (2) 对于语料中的任意词 $w_t$ ，获取其前面 $n-1$ 个词，输入层将这前 $n-1$ 个词的词向量 $C(w_{t-n+1}), C(w_{t-n+2}), \dots, C(w_{t-1})$ 拼接起来;
- (3) 隐藏层通过激励函数将输入信息进行转换;
- (4) 输出层计算词 $w_t$ 的概率。

# 事件的独立性

○两事件独立：事件A、B，若 $P(AB)=P(A)P(B)$ ，则称A、B独立

○三事件独立：事件A B C，若满足

$$P(ABC)=P(A)P(B)P(C)$$

$$P(AB)=P(A)P(B)$$

$$P(AC)=P(A)P(C)$$

$$P(BC)=P(B)P(C)$$

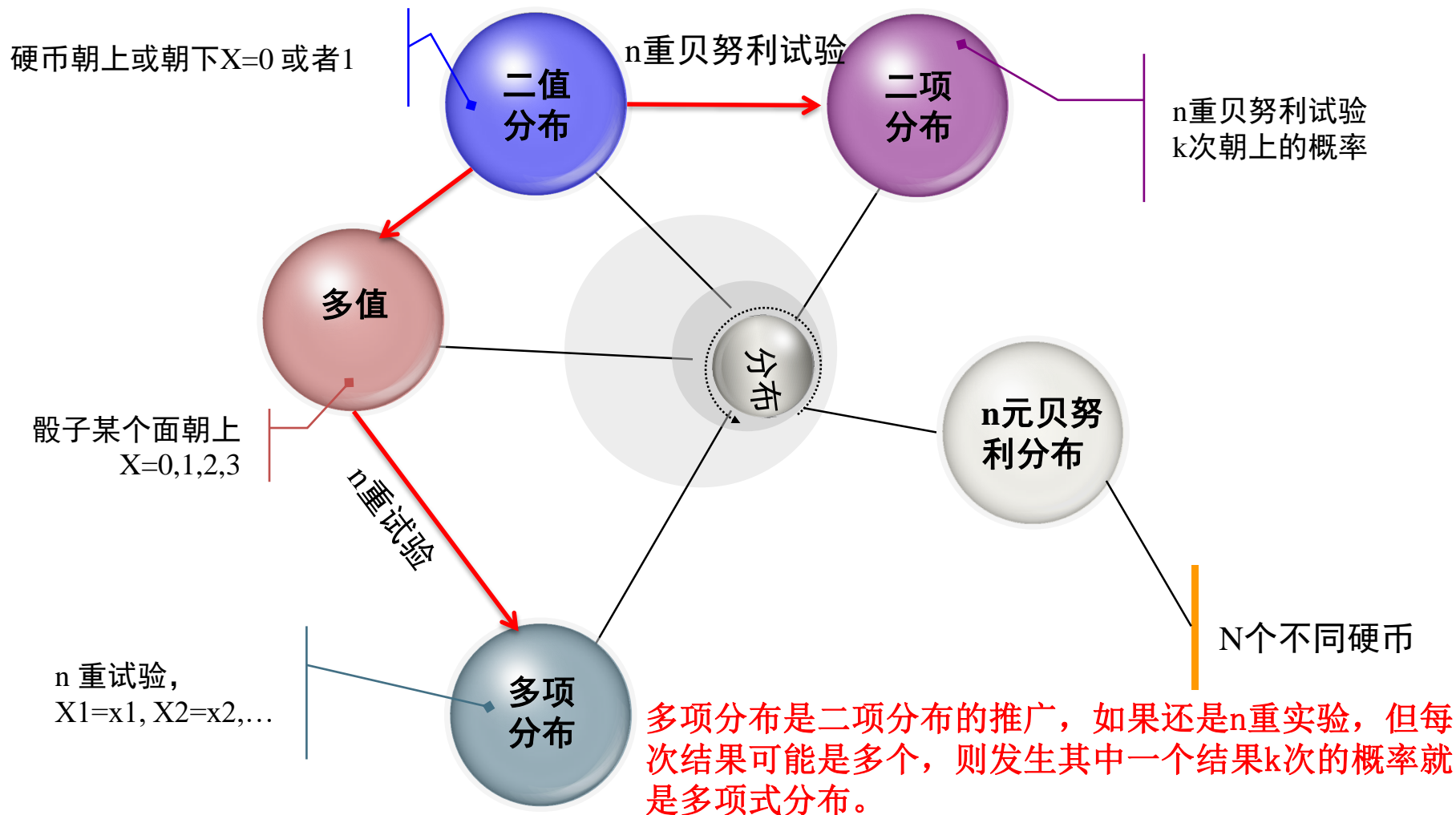
则称A、B、C独立

○多事件独立：两两独立、三三独立、四四独立….

# 随机变量

- 随机变量：若随机试验的各种可能的结果都能用一个变量的取值（或范围）来表示，则称这个变量为随机变量，常用 $X$ 、 $Y$ 、 $Z$ 来表示
  - (离散型随机变量)：掷一颗骰子，可能出现的点数 $X$  (可能取值1、2、3、4、5、6)
  - (连续型随机变量)：北京地区的温度(-15~45)

# 各种分布关系图



# 贝努利

- 瑞士数学家家族，产生过11位数学家
- 在科学史上，父子科学家、兄弟科学家并不鲜见，然而，在一个家族跨世纪的几代人中，众多父子兄弟都是科学家的较为罕见，瑞士的伯努利家族最为突出
- 雅可比-贝努利(Jacob Bernoulli)：1654-1705，积分“integral”这一术语即由他首创
- 贝努利试验、贝努利分布



# 概率检索模型

---

- 检索系统中，给定查询，计算每个文档的相关度
- 检索系统对用户查询的理解是非确定的(uncertain)，对返回结果的猜测也是非确定的
- 而概率理论为非确定推理提供了坚实的理论基础
- 概率检索模型可以计算文档和查询相关的可能性

# 概率检索模型

- 概率检索模型是通过概率的方法将查询和文档联系起来
  - 定义3个随机变量 $R$ 、 $Q$ 、 $D$ ：相关度 $R=\{0,1\}$ ，查询 $Q$ 可以是 $q_1, q_2, \dots$ 中的一个查询，文档 $D$ 可以是 $d_1, d_2, \dots$ 中的一篇文档，则可以通过计算条件概率 $P(R=1|Q=q, D=d)$ 来度量文档和查询的相关度。
- 概率检索模型包括一系列模型，如Logistic Regression(回归)模型及最经典的二值独立概率模型BIM、BM25模型等等(还有贝叶斯网络模型)。
- 1998出现的基于统计语言建模的信息检索模型本质上也是概率模型的一种。

# 概率检索模型排序原理

- 概率排序原理
- 基本思想：给定一个用户查询，若搜索系统能在搜索结果排序时按照文档和用户查询的**相关性由高到低排序**，那么这个搜索系统的准确性是最优的。
- 实际实现
- 根据用户的查询将文档集合划分为两个集合：相关文档子集和不相关文档子集。
- 将相关性衡量转换为分类问题，对某个文档D来说，若其属于相关文档子集的概率大于属于不相关文档的概率，就认为它与查询相关。
- 形式化表示： $P(R|D)$ 代表给定一个文档D对应的相关性概率，而 $P(NR|D)$ 代表该文档的不相关概率，**若 $P(R|D) > P(NR|D)$ 我们就认为此文档与查询相关。**



# 概率排序原则(PRP)

- **简单地说：**如果文档按照与查询的相关概率大小返回，那么该返回结果是所有可能获得结果中效果最好的。
- **严格地说：**如果文档按照与查询的相关概率大小返回，而**这些相关概率又能够基于已知数据进行尽可能精确的估计**，那么该返回结果是所有基于已知数据获得的可能的结果中效果最好的。

# 几种概率检索模型

---

- 基于Logistic回归的检索模型
- 经典的二值独立概率模型BIM
- 经典的BM25模型 (BestMatch25)
- 贝叶斯网络模型：本讲义不介绍，请参考有关文献。
- 基于语言建模的检索模型：1998年兴起，研究界的热点。下一讲介绍。

# 提纲

- ① 上一讲及向量空间模型回顾
- ② 基本概率统计知识
- ③ Logistic回归模型
- ④ BIM模型
- ⑤ BM25模型

# 机器学习基本问题

## 分类问题

决策树

贝叶斯

支持向量机

逻辑回归

集成学习

## 回归问题

线性回归

岭回归

**Lasso**回归

## 聚类问题

K-means

高斯混合聚类

密度聚类

层次聚类

谱聚类

## 其他问题

条件随机场

隐马尔可夫模型

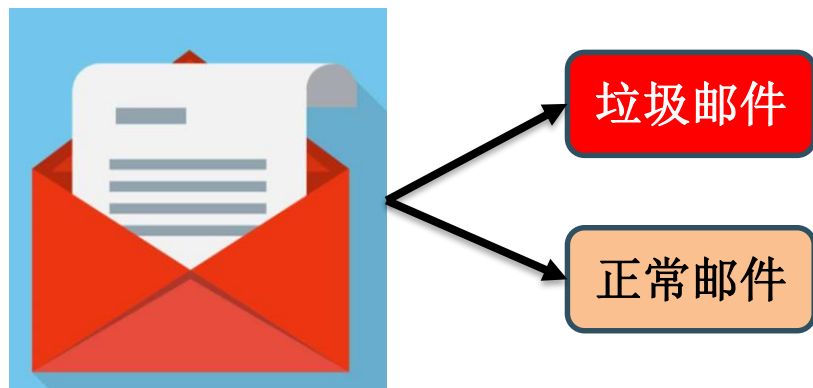
LDA主题模型

# 分类问题

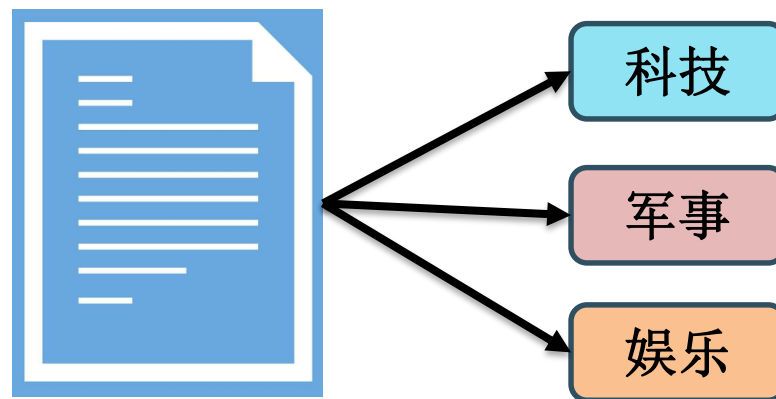
**分类问题**是**监督学习**的一个核心问题，它从数据中学习一个分类决策函数或分类模型(分类器 (classifier) )，对新的输入进行输出预测，输出变量取有限个离散值。

## □ 分类在我们日常生活中很常见

✓ 二分类问题



✓ 多分类问题

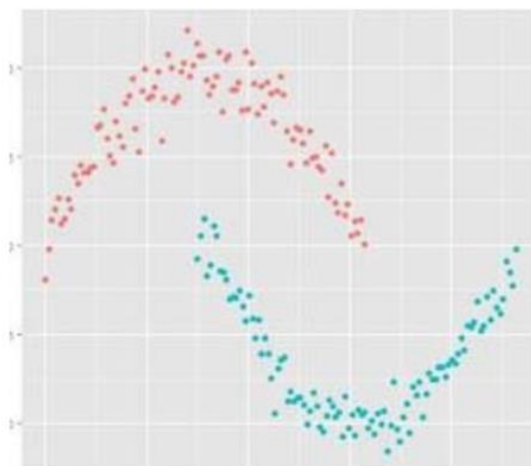
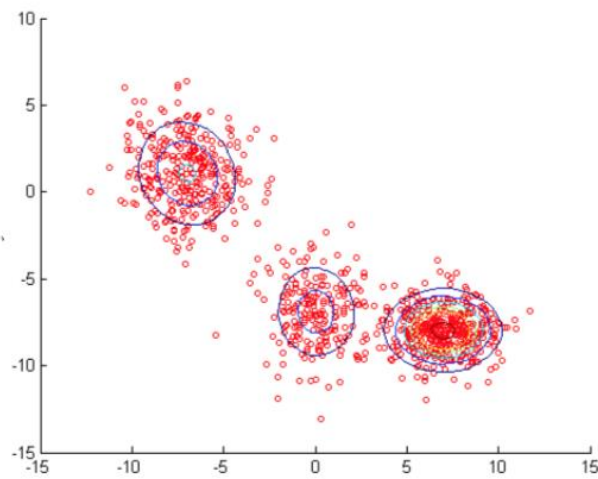
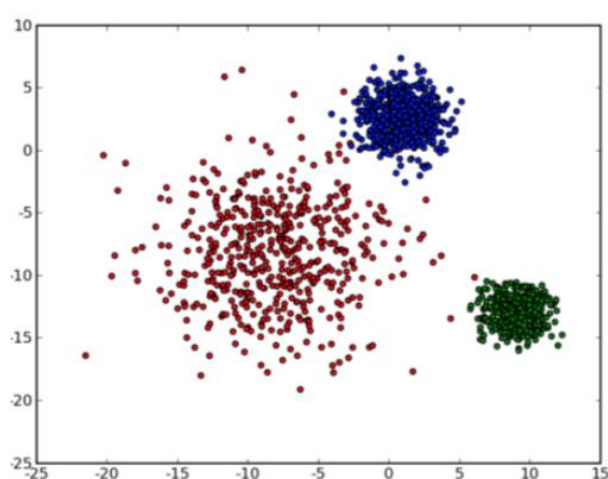


## □ 核心算法

✓ 决策树、贝叶斯、SVM

# 聚类问题

**聚类问题**是**无监督学习**的问题，算法的思想就是“物以类聚，人以群分”。聚类算法感知样本间的相似度，进行类别归纳，对新的输入进行输出预测，输出变量取有限个离散值。



- ✓ 可以作为一个单独过程，用于寻找数据内在的分布结构
- ✓ 可以作为分类、稀疏表示等其他学习任务的前驱过程

# 回归问题

**回归分析**用于预测输入变量（自变量）和输出变量（因变量）之间的关系，特别是当输入变量的值发生变化时，输出变量值随之发生变化。直观来说回归问题等价于**函数拟合**，选择一条函数曲线使其很好地拟合已知数据且很好地预测未知数据。

回归  
分析

自变量个数

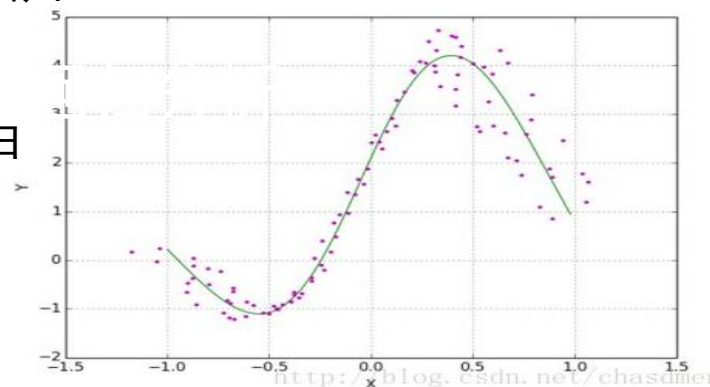
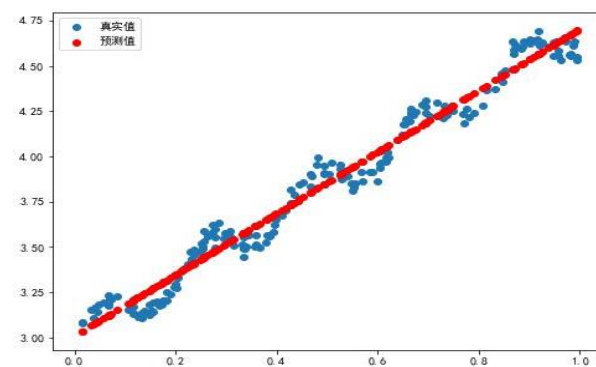
一元回归、多元回归

自变量与因变量  
关系

线性回归、非线性回归

因变量个数

简单回归、多重回归



# 线性回归

**线性回归算法**假设特征（自变量）和结果（因变量）满足线性关系。这就意味着可以将输入项分别乘以一些常量，再将结果加起来得到输出。

## 线性回归算法流程

- ① 选择拟合函数形式

$$h_{\theta}(x) = \sum_{i=0}^n \theta_i x_i = \theta^T x$$

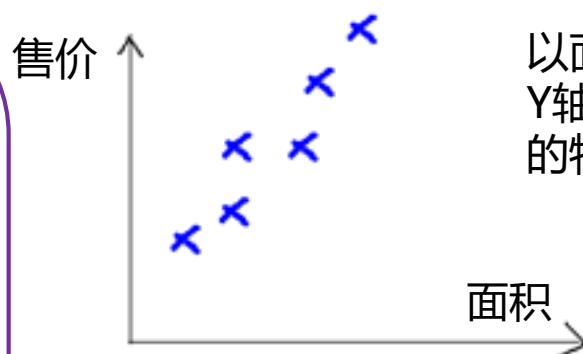
- ② 确定损失函数形式

$$\min_{\theta} J(\theta) = \frac{1}{2} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

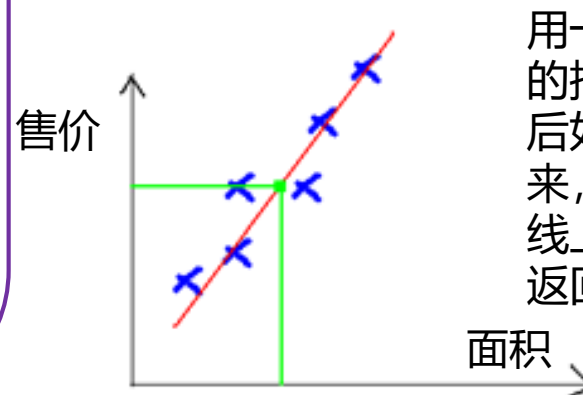
- ③ 训练算法，找到回归系数  
如最小二乘、梯度下降等

- ④ 使用算法进行数据预测

$$y = 10 * x + 3$$



以面积为X轴，售价为Y轴建立房屋销售数据的特征空间表示图。



用一条曲线去尽量准的拟合这些数据，然后如果有新的输入过来，我们可以在将曲线上这个点对应的值返回。



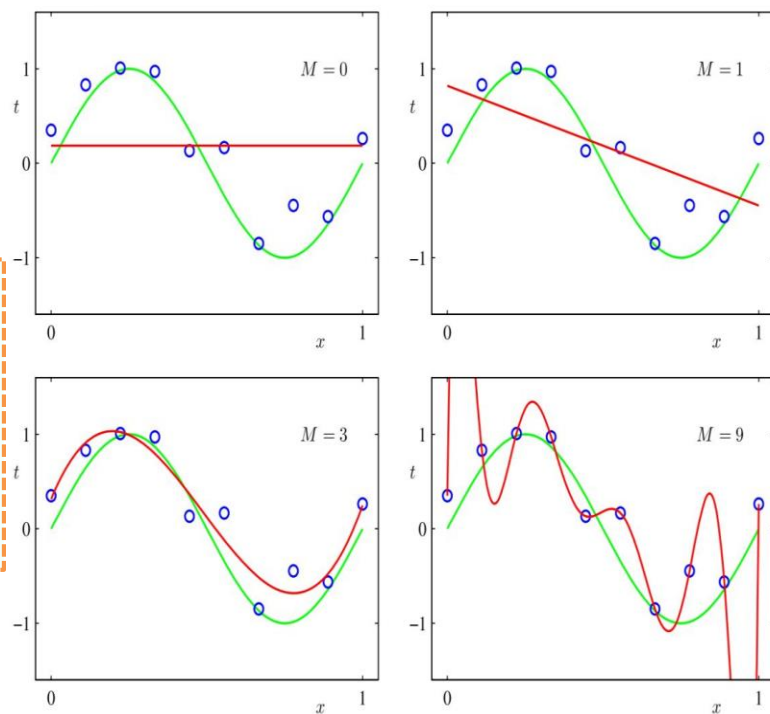
# 岭回归

**岭回归**在经验风险最小化的基础上加入正则化因子。当正则化因子选择为模型参数的二范数的时候，整个回归的方法就叫做岭回归。

$$\underset{\beta_0 \in \mathbb{R}, \beta \in \mathbb{R}^p}{\text{minimize}} \frac{1}{n} \sum_{i=1}^n (y_i - \beta_0 - x_i^T \beta)^2 \quad \text{subject to} \quad \|\beta\|^2 \leq t$$

$$\underset{\beta_0 \in \mathbb{R}, \beta \in \mathbb{R}^p}{\text{minimize}} \frac{1}{n} \sum_{i=1}^n (y_i - \beta_0 - x_i^T \beta)^2 + \lambda \|\beta\|^2$$

$\lambda$ 越大，说明偏差就越大，原始数据对回归求取参数的作用就越小，当 $\lambda$ 取到一个合适的值，就能在一定意义上解决过拟合的问题：原先过拟合的特别大或者特别小的参数会被约束到正常甚至很小的值，但不会为零。

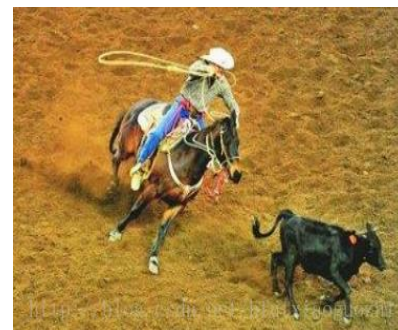


# Lasso回归

**Lasso回归**是一种压缩估计（正则化因子选择1范数）。它通过构造一个惩罚函数得到一个较为精炼的模型，使得它压缩一些系数，同时设定一些系数为零。

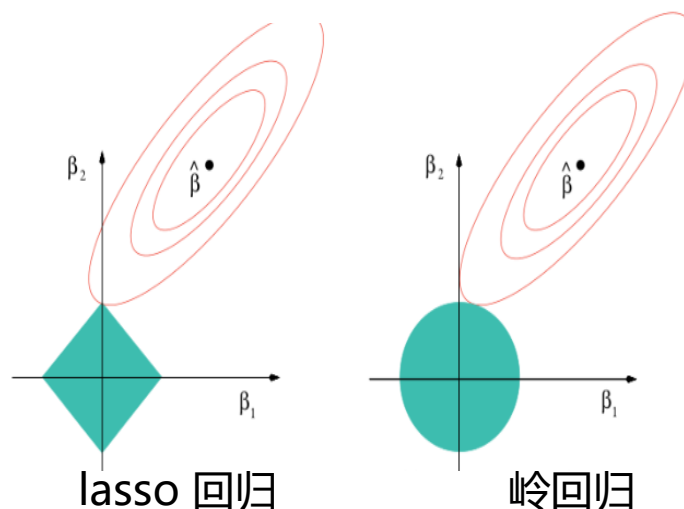
Lasso回归翻译成中文叫套索，就是拿这个东西把动物脖子套住，不要它随便跑。lasso 回归差不多这个意思，就是让回归系数不要太大，以免造成过度拟合（overfitting）。

$$\underset{\beta_0 \in \mathbb{R}, \beta \in \mathbb{R}^p}{\text{minimize}} \frac{1}{n} \sum_{i=1}^n (y_i - \beta_0 - x_i^T \beta)^2 \quad \text{subject to} \quad \|\beta\|_1 \leq t$$



**lasso回归可以适应的情况：**

样本量比较小，但是特征非常多。适用于高维统计，传统的方法无法应对这样的数据。并且lasso可以进行特征选择。

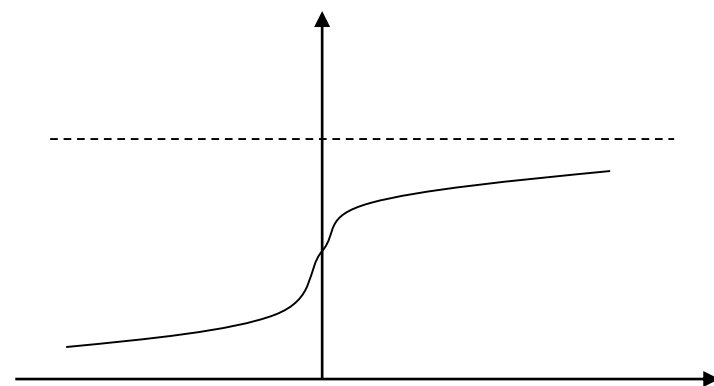


图中的红色线圈表示，损失函数的等值线，可以看到在Lasso第一范数约束下， $\beta_1$ 可以被约束成0。

# Logistic 回归（分类模型）

- Logistic回归是一种非线性回归
- Logistic (也叫Sigmoid)函数(S型曲线):

$$y = f(x) = \frac{1}{1 + e^{-(\alpha + \beta x)}} = \frac{e^{\alpha + \beta x}}{1 + e^{\alpha + \beta x}}$$



- Logistic回归可以转化成线性回归来实现

$$\frac{y}{1-y} = e^{\alpha + \beta x}, \quad \ln \frac{y}{1-y} = \alpha + \beta x$$

**线性回归扩展：**用简单的基函数  $\Phi(x)$  替换输入变量  $x$ 。这样就把线性拟合形式扩展到了固定非线性函数的线性组合。

# Logistic 回归IR模型

- 基本思想：为了求 $Q$ 和 $D$ 相关的概率 $P(R=1|Q,D)$ ，通过定义多个特征函数 $f_i(Q,D)$ ，认为 $P(R=1|Q,D)$ 是这些函数的组合。
- Cooper等人提出一种做法\*：定义 $\log(P/(1-P))$ 为多个特征函数的线性组合。则 $P$ 是一个Logistic函数，即：

$$\log \frac{P}{1-P} = \beta_0 + \sum_i \beta_i f_i(Q,D)$$

$$P = \frac{1}{1 + e^{-\beta_0 - \sum_i \beta_i f_i(Q,D)}}$$

\*William S. Cooper , Fredric C. Gey , Daniel P. Dabney, Probabilistic retrieval based on staged logistic regression, Proceedings of ACM SIGIR'92, p.198-210, June 21-24, 1992, Copenhagen, Denmark

# 特征函数 $f_i$ 的选择

$$X_1 = \frac{1}{M} \sum_1^M \log QAF_{t_j}$$

$$X_2 = \sqrt{QL}$$

$$X_3 = \frac{1}{M} \sum_1^M \log DAF_{t_j}$$

$$X_4 = \sqrt{DL}$$

$$X_5 = \frac{1}{M} \sum_1^M \log IDF_{t_j}$$

$$IDF = \frac{N - n_{t_j}}{n_{t_j}}$$

$$X_6 = \log M$$

AF代表absolute frequency

QAF代表出现在query中的term的AF

DAF代表出现在document中的term的AF

DL是文档长度

QL是查询长度

# Logistic 回归IR模型(续)

- 求解和使用过程：
  - 通过训练集合拟和得到相应系数  $\beta_0 \sim \beta_6$
  - 对于新的文档，代入公式计算得到概率 $P$
  - *Learning to Rank*中*Pointwise*方法中的一种
  - 判别式(discriminate)模型
- 优缺点：
  - 优点：直接引入数学工具，形式简洁。
  - 缺点：特征选择非常困难，实验中效果一般。

# 提纲

- ① 上一讲及向量空间模型回顾
- ② 基本概率统计知识
- ③ Logistic回归模型
- ④ BIM模型
- ⑤ BM25模型

# 二值独立概率模型BIM

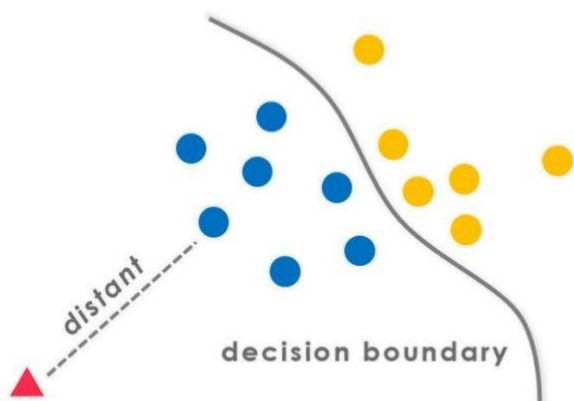
- 二值独立概率模型(Binary Independence Model, 简称BIM): 伦敦城市大学Robertson及剑桥大学Sparck Jones 1970年代提出, 代表系统OKAPI
- 贝叶斯公式
$$P(A|B) = \frac{P(A,B)}{P(B)} = \frac{P(B|A)P(A)}{P(B)}$$
- BIM模型通过贝叶斯公式对所求条件概率 $P(R=1|Q,D)$ 展开进行计算。BIM是一种生成式(generative)模型
- 对于同一 $Q$ ,  $P(R=1|Q,D)$ 可以简记为 $P(R=1|D)$



# 生成模型 VS 判别模型（扩展）

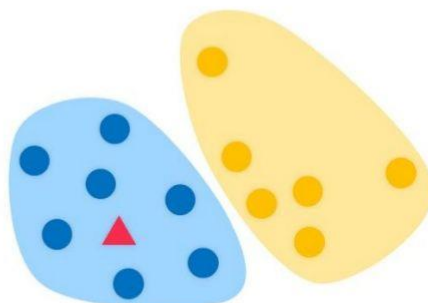
## Discriminative vs. Generative

Discriminative



- Only care about estimating the conditional probabilities
- Very good when underlying distribution of data is really complicated (e.g. texts, images, movies)

Generative



- Model observations  $(x, y)$  first, then infer  $p(y|x)$
- Good for missing variables, better diagnostics
- Easy to add prior knowledge about data

机器学习任务是从属性 $X$ 预测标记 $Y$ ，即求概率 $P(Y|X)$

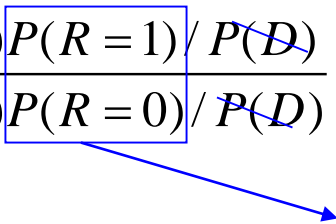
▣ 判别式模型求的是 $P(Y|X)$ ，对未见示例 $X$ ，根据 $P(Y|X)$ 可以求得标记 $Y$ （线性回归、SVM）

▣ 生成式模型求的是 $P(Y, X)$ ，对于未见示例 $X$ ，要求出 $X$ 与不同标记之间的联合概率分布，然后大的获胜（朴素贝叶斯、HMM）

# BIM模型(续)

- 对每个 $Q$ 定义排序(Ranking)函数RSV( $Q, D$ ):

$$\log \frac{P(R=1|D)}{P(R=0|D)} = \log \frac{P(D|R=1)P(R=1)/P(D)}{P(D|R=0)P(R=0)/P(D)}$$

$$\propto \log \frac{P(D|R=1)}{P(D|R=0)}$$


对同一 $Q$ 是常量,  
对排序不起作用

其中,  $P(D|R=1)$ 、 $P(D|R=0)$ 分别表示在相关和不相关情况下生成文档 $D$ 的概率。Ranking函数显然是随着 $P(R=1|D)$ 的增长而增长。

# 文档是怎么生成的？

- 钢铁是怎么炼成的？
- 博士是怎么读成的？
- .....
- 概率的观点：
  - 词项满足某个总体分布，然后从该总体分布中抽样，将抽样出的词项连在一起，组成文档
  - 对于 $P(D|R=1)$ 或者 $P(D|R=0)$ ，可以认为 $R=1$ 或 $0$ 的文档的词项满足某个总体分布，然后抽样生成 $D$

# 两种常用的文档生成的总体分布

- 多元贝努利分布(Multi-variate Bernoulli distribution)
    - 贝努利试验是只有两种可能结果的单次随机试验， $n$ 个不同硬币就是多元贝努利实验
    - 词项词典大小为 $M$ ， $M$ 个不规则硬币分别对应 $M$ 个词项，第 $i$ 个硬币朝上的概率为 $p_i$
    - 多元贝努利分布不考虑出现位置
    - 多元贝努利分布考虑出现和不出现
    - 多元贝努利分布不考虑出现次数
    - 假设 $M=4$ (四个词项分别为 I you can fly),  $p_1=0.7$ ,  $p_2=0.4$ ,  $p_3=0.1$ ,  $p_4=0.05$
- 则：  $P(\text{I can fly fly}) = (X_1=1, X_2=0, X_3=1, X_4=1) = ?$

# 两种常用的文档生成的总体分布(续)

- 多项式分布(Multinomial distribution)

- 词项大小为M, 某个不规则骰子共有M个面, 每个面对应一个词项(假设每次抛掷必有某个面稳定朝上或下), 第*i*个面朝上的概率为 $p_i$
- 多项式分布考虑词项的多次出现
- 多项式分布不考虑词项的不出现
- 多项式分布同样不考虑词项的出现位置和次序

假定 $M=4$  (四个词项分别为 I you can fly),  $p_1=0.4$ ,  $p_2=0.3$ ,  $p_3=0.2$ ,  $p_4=0.1$

则:  $P(\text{I can fly fly})=P(X_1=1, X_2=0, X_3=1, X_4=2)=?$

# BIM中 $P(D|R=1)$ 或 $P(D|R=0)$ 的计算

- 类比： $M$ 次独立试验 (多元贝努利模型)
  - 假想词项空间中有 $M$ 个词项，相当于有 $M$ 个不规则硬币，第 $i$ 个硬币对应词项 $i$ ，正面写着“出现 $t_i$ ”，反面写着“不出现 $t_i$ ”，独立地抛这 $M$ 个硬币，然后记录下每个硬币朝上的面对应的词项便组成文档 $D$ 。
  - 因此，求 $P(D|R)$ 就是抛这个 $M$ 个硬币得到 $D$ 的概率。假设抛不同硬币之间是独立的(独立性假设)，并且不考虑 $t_i$ 出现的次数，只考虑 $t_i$ 要么出现要么不出现(二值)。同时，也不考虑抛硬币的次序(词袋模型)
  - $P(D|R=1)$ 和 $P(D|R=0)$ 相当于有两组硬币，因此需要求解 $2M$ 个概率参数

# 一个例子——BIM模型

- 查询为：信息 检索 教程

所有词项的在相关、不相关情况下的概率 $p_i$ 、 $q_i$ 分别为：

词项	信息	检索	教材	教程	课件
R=1时的概率 $p_i$	0.8	0.9	0.3	0.32	0.15
R=0时的概率 $q_i$	0.3	0.1	0.35	0.33	0.10

文档D1： 检索 课件

则：  $P(D|R=1)=(1-0.8)*0.9*(1-0.3)*(1-0.32)*0.15$

$$P(D|R=0)= (1-0.3)*0.1*(1-0.35)*(1-0.33)*0.10$$

$$P(D|R=1)/P(D|R=0)=4.216$$

# BIM模型公式的推导

将 $D$ 看成

$$P(D | R = 1) = \prod_{t_i \in D} P(t_i | R = 1) \prod_{t_i \notin D} P(\bar{t}_i | R = 1)$$

$$= \prod_{t_i} p_i^{e_i} (1 - p_i)^{1-e_i}, \text{if } t_i \in D \text{ then } e_i = 1, \text{else } e_i = 0$$

$$P(D | R = 0) = \prod_{t_i \in D} P(t_i | R = 0) \prod_{t_i \notin D} P(\bar{t}_i | R = 0)$$

$$= \prod_{t_i} q_i^{e_i} (1 - q_i)^{1-e_i}, \text{if } t_i \in D \text{ then } e_i = 1, \text{else } e_i = 0$$

$$p_i = P(t_i | R = 1)$$

$$q_i = P(t_i | R = 0)$$

**注：**  $P(t_i|R=1)$ 表示在相关情况下， $t_i$ 出现在某个文档中的概率（出现或不出现）。  
**注意：** 不是在相关文档集中出现的概率，因此所有 $P(t_i|R=1)$ 的总和不为1。



# BIM模型公式的推导

- 继续推导，去掉公式中的只依赖查询 $Q$ 的常数项（和 $D$ 无关），得所有出现在文档 $D(e_i=1)$ 中的词项的某个属性值之和。再假定对于不出现在 $Q$ 中的词项，有 $p_i=q_i$ ，则得到所有出现在 $Q \cap D$ 中的词项的属性值之和

$$\begin{aligned}
 \log \frac{P(D | R=1)}{P(D | R=0)} &= \log \frac{\prod_{t_i \in D \cup \bar{D}} p_i^{e_i} (1-p_i)^{1-e_i}}{\prod_{t_i \in D \cup \bar{D}} q_i^{e_i} (1-q_i)^{1-e_i}} = \sum_{t_i \in D \cup \bar{D}} \log \left( \frac{p_i}{q_i} \right)^{e_i} \left( \frac{1-p_i}{1-q_i} \right)^{1-e_i} \quad \text{常数} \\
 &= \sum_{t_i \in D \cup \bar{D}} \left( e_i \log \frac{p_i}{q_i} + (1-e_i) \log \frac{1-p_i}{1-q_i} \right) = \sum_{t_i \in D \cup \bar{D}} \left( e_i \log \frac{p_i}{q_i} - e_i \log \frac{1-p_i}{1-q_i} + \log \frac{1-p_i}{1-q_i} \right) \quad \text{假设对不属于Q的 term, } p_i=q_i, \text{ 则此项为零} \\
 &\stackrel{t_i \text{ 在 } D \text{ 中, 权重取0或1}}{\approx} \sum_{t_i \in D \cup \bar{D}} \left[ e_i \log \frac{p_i/(1-p_i)}{q_i/(1-q_i)} \right] = \sum_{t_i \in D} \log \frac{p_i/(1-p_i)}{q_i/(1-q_i)} = \sum_{t_i \in Q \cap D} \log \frac{p_i/(1-p_i)}{q_i/(1-q_i)} + \sum_{t_i \notin Q \wedge t_i \in D} \log \frac{p_i/(1-p_i)}{q_i/(1-q_i)} \\
 &\approx \sum_{t_i \in Q \cap D} \log \frac{p_i/(1-p_i)}{q_i/(1-q_i)} \\
 &= \sum_{t_i \in D \cap Q} W_i^{BIM} \quad \text{在 } Q \text{ 中权重, 只与 } Q \text{ 相关}
 \end{aligned}$$

最原始的BIM模型的计算公式，其中最关键是 $p_i$ 、 $q_i$ 的计算！

# $p_i$ $q_i$ 参数的计算

理想情况下，可以将整个文档集合根据是否和查询相关、是否包含 $t_i$ 分成如下四个子集合，每个集合的大小已知。

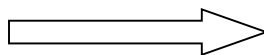
	相关 $R_i$ (100)	不相关 $N-R_i$ (400)
包含 $t_i$ $n_i$ (200)	$r_i$ (35)	$n_i - r_i$ (165)
不包含 $t_i$ $N-n_i$ (300)	$R_i - r_i$ (65)	$N - R_i - n_i + r_i$ (235)

其中， $N$ 、 $n_i$ 分别是总文档以及包含 $t_i$ 的文档数目。 $R_i$ 、 $r_i$ 分别是相关文档及相关文档中包含 $t_i$ 的文档数目。括号中列举的数值是给出的一个总文档数目为500的计算例子。则：

$$p_i = \frac{r_i}{R_i} = \frac{35}{100} = 0.35$$

$$q_i = \frac{n_i - r_i}{N - R_i} = \frac{165}{400} = 0.413$$

平滑



$$p_i = \frac{r_i + 0.5}{R_i + 1}$$

$$q_i = \frac{n_i - r_i + 0.5}{N - R_i + 1}$$

# RSJ权重

- Robertson & Spärck Jones权重(RSJ权重)

$$W_i^{RSJ} = \log \frac{(r_i + 0.5)(N - R - n_i + r_i + 0.5)}{(n_i - r_i + 0.5)(R - r_i + 0.5)}$$

BM25是在信息检索系统中根据query对document进行评分的算法。It is based on the probabilistic retrieval framework developed in the 1970s and 1980s by Stephen E. Robertson, Karen Spärck Jones, and others

## $p_i$ $q_i$ 参数的计算(续)

- 由于在真实情况下，对于每个查询，无法事先得到相关文档集和不相关文档集，所以无法使用理想情况下的公式计算，因此必须进行估计
- 有多种估计方法
  - 初始检索：第一次检索之前的估计
  - 基于检索结果：根据上次检索的结果进行估计

# $p_i$ $q_i$ 参数的计算(续)

- **初始情况**：检索初始并没有相关和不相关文档集合，此时可以进行假设： $p_i$ 是常数， $q_i$ 近似等于term  $i$ 在所有文档集合中的分布(**假定相关文档很少**)

$$p_i = 0.5$$

$$q_i = \frac{n_i}{N}$$

包含 $t_i$   $n_i$  (200)  
不包含 $t_i$   $N-n_i$  (300)

相关  $R_i$  (100)      不相关  $N-R_i$  (400)

$r_i$ (35)	$n_i - r_i$ (165)
$R_i - r_i$ (65)	$N - R_i - n_i + r_i$ (235)

$$\sum_{t_i \in D \cap Q} \log \frac{p_i / (1 - p_i)}{q_i / (1 - q_i)} = \sum_{t_i \in D \cap Q} \log \frac{N - n_i}{n_i}$$

$$\approx \sum_{t_i \in D \cap Q} \log \frac{N - n_i + 0.5}{n_i + 0.5} = \sum_{t_i \in D \cap Q} W_i^{IDF}$$

因此，BIM在初始假设情况下，其检索公式实际上相当于对**所有同时出现在 $q$ 和 $d$ 中的词项**的IDF的求和

# $p_i q_i$ 参数的计算(续)

- 基于前面的检索结果：假定检索出的结果集合 $V$ (可以把 $V$ 看成全部的相关文档集合)，其中集合 $V_i$ 包含term  $i$ ，则可以进一步进行计算
- 为避免较小的 $V$ 和 $V_i$ 集合，加入常数或非常数平滑因子(以下用 $V$ 和 $V_i$ 表示同名集合的大小)

$$\begin{array}{ccc}
 p_i = \frac{V_i}{V} & \xrightarrow{\text{平滑方式1}} & p_i = \frac{V_i + 0.5}{V + 1} \\
 q_i = \frac{n_i - V_i}{N - V} & \xrightarrow{\text{平滑方式2}} & q_i = \frac{n_i - V_i + \frac{n_i}{N}}{N - V + 1}
 \end{array}$$

# BIM模型小结

---

- BIM计算过程：目标是求排序函数  $P(D|R=1)/P(D|R=0)$ 
  - 首先估计或计算每个term分别在相关文档和不相关文档中的出现概率  $p_i=P(t|R=1)$  及  $q_i=P(t|R=0)$
  - 然后根据独立性假设，将  $P(D|R=1)/P(D|R=0)$  转化为  $p_i$  和  $q_i$  的某种组合，将  $p_i$  和  $q_i$  代入即可求解。

# BIM模型的优缺点

---

- 优点：
  - BIM模型建立在数学基础上，理论性较强
- 缺点：
  - 需要估计参数
  - 原始的BIM没有考虑TF、文档长度因素
  - BIM中同样存在词项独立性假设



# 提纲

- ① 上一讲及向量空间模型回顾
- ② 基本概率统计知识
- ③ Logistic回归模型
- ④ BIM模型
- ⑤ BM25模型

# Okapi BM25: 一个非二值模型

- BIM是最简单的文档评分方式:

$$RSV(Q, D) = \sum_{t_i \in D \cap Q} W_i^{IDF}$$

- 考虑词项在文档中的tf权重, 有:

$$RSV(Q, D) = \sum_{t_i \in D \cap Q} W_i^{IDF} \frac{(k_1 + 1)tf_{t_i, D}}{k_1((1 - b) + b \times (L_D / L_{ave})) + tf_{t_i, D}}$$

- $tf_{t_i, D}$ : 词项 $t_i$ 在文档 $D$ 中的词项频率
- $L_D$  ( $L_{ave}$ ): 文档 $D$ 的长度(整个文档集的平均长度)
- $k_1$ : 用于控制文档中词项频率比重的调节参数
- $b$ : 用于控制文档长度比重的调节参数

# Okapi BM25: 一个非二值模型

- 如果查询比较长，则加入查询的tf

$$RSV(Q, D) = \sum_{t_i \in D \cap Q} W_i^{IDF} \cdot \frac{(k_1 + 1)tf_{ti,D}}{k_1((1-b) + b \times (L_D / L_{ave})) + tf_{ti,D}} \cdot \frac{(k_3 + 1)tf_{ti,Q}}{k_3 + tf_{ti,Q}}$$

- $tf_{ti,Q}$ : 词项 $t_i$ 在Q中的词项频率
- $k_3$ : 用于控制查询中词项频率比重的调节参数
- 没有查询长度的归一化 (由于查询对于所有文档都是固定的)
- 理想情况下，上述参数都必须在开发测试集上调到最优。一般情况下，实验表明， $k_1$  和  $k_3$  应该设在 1.2到2之间， $b$  设成 0.75。

# 参考资料

---

- 《现代信息检索》 第11章
  
- BM25模型的推导
  - S.E Robertson and S. Walker, Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval, SIGIR'94
  - S.E Robertson, S. Walker, S. Jones, Okapi at TREC-3, in Proceedings of TREC-3