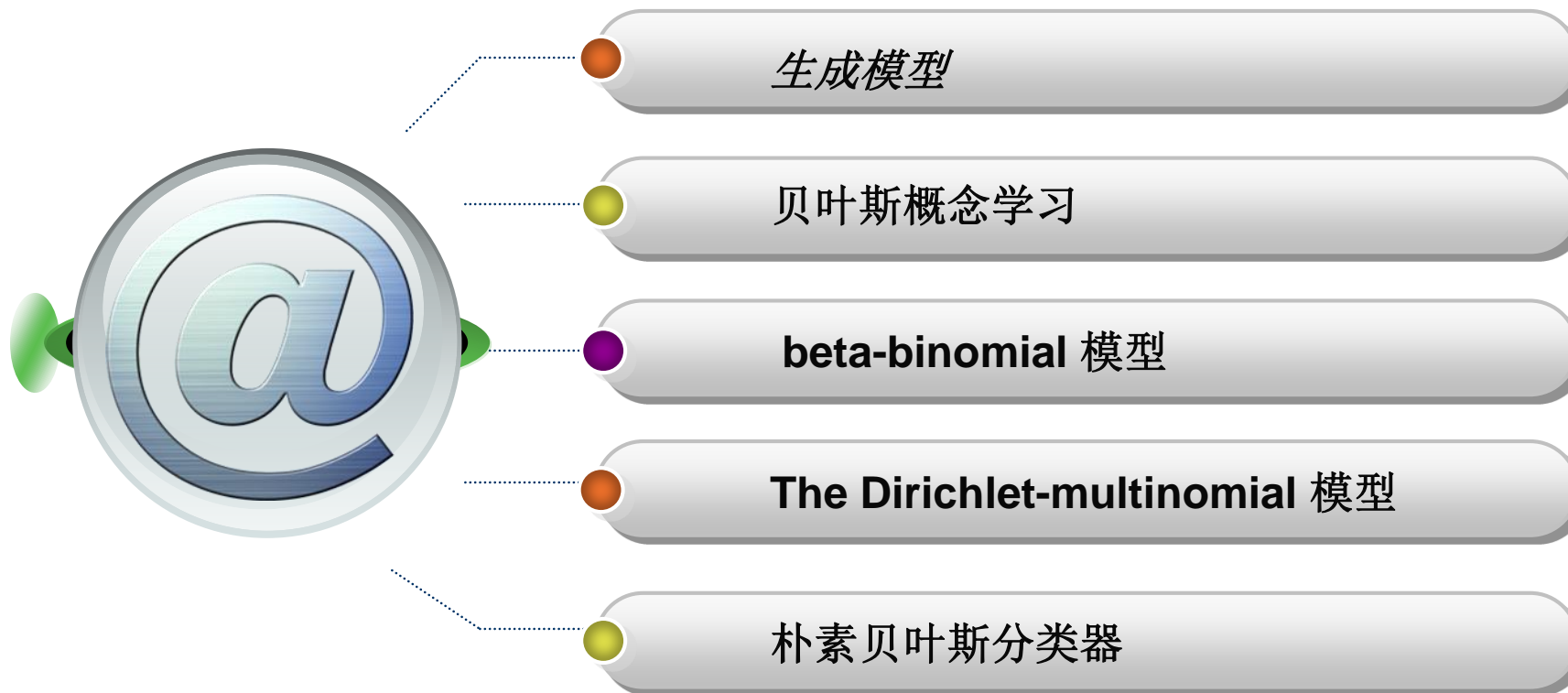


第2章：离散数据的生成模型



目录



生成式分类器

$$p(y = c | x, \theta) \propto p(x | y = c, \theta) p(y = c | \theta)$$

❖ 使用此类模型的关键:

- 确定合适的类条件密度形式 $p(\mathbf{x}/y = c, \theta)$,
- 如何推导出这类模型的未知参数 θ .



贝叶斯概念学习



小孩学习词汇

❖ 例如，孩子学习理解“狗”一词的含义

- 孩子的父母可能会给出正面的例子，说：

- ⑩ “看那只可爱的狗！”

- ⑩ 或“小心狗狗”等



概念学习

❖ 学习一个词汇的意义等价于概念学习

❖ 它相当于二值分类：

- $f(x) = 1$ ，如果 x 是概念 C 的实例
- $f(x) = 0$ ，否则

❖ 目标是学习指标函数 f

- f 定义了元素是否在集合 C 中



数字游戏

❖ 这个游戏是这样的

- 一个简单的数学概念 C ，如“素数”
- 给定一系列随机选择的正面样例
 - ⑩ $D = \{x_1, \dots, x_N\}$ ，从 C 中抽样出来
- 问新的测试样例 x 是否属于 C



仅用一个样本玩数字游戏

- ❖ 假设所有数字都是1到100之间的整数。
- ❖ 概念的一个正样本：16，问
 - 哪些是正的？17？6？32？99？
- ❖ 仅凭一个样本很难判断，你的预测可能会很模糊
 - 17是相似的，因为它是“近在咫尺”
 - 6是相似的，因为它与16有一位数是相同的
 - 32是相似的，因为它也是偶数和2的幂
 - 99似乎并不相似。



用多个样本玩数字游戏

❖ 正样本 $D=\{16, 8, 2, 64\}$

- 你认为哪些属于正样本？17？6？32？99？
- 你可能会猜到隐藏的概念是“二的幂”
- 这是归纳法的一个例子

❖ 你为什么不说概念是“偶数”，这两个概念都与证据一致？



归纳法：一种经典方法

❖ 概念的假设空间 H :

- 例如：奇数、偶数、二的幂等。

❖ 版本空间:

- 与数据 D 一致的 H 的子集

❖ 版本空间缩小，我们对这个概念越来越确定



利用贝叶斯定理实现归纳法

❖ 贝叶斯理论:

$$p(h | D) = \frac{p(D | h) p(h)}{p(D)}$$

❖ 根据: $p(h|D)$ (后验概率)

- 后验估计

❖ 根据: $p(D|h)$ (似然函数)

- 似然估计.

❖ $p(h)$ 为 先验概率



数字游戏模型的似然函数

❖ 似然函数:

$$p(D \mid h) = \left[\frac{1}{size(h)} \right]^N = \left[\frac{1}{\|h\|} \right]^N$$

其中, 假设 $h =$ 属于该假设的所有数字的集合

❖ 它表示在假设 h 为真的条件下, 观测到的数据集多大可能属于这个 h



最大似然估计 (MLE)

- ❖ 设 $h_{two} = \text{“2的幂”} = \{2, 4, 8, 16, 32, 64\}$;
- ❖ $h_{even} = \text{“偶数”} = \{2, 4, 6, \dots, 98, 100\}$;
- ❖ 样本集 $D = \{16, 8, 2, 64\}$
- ❖ 计算似然

$$p(D | h_{two}) = \left[\frac{1}{6} \right]^4 = 7.7 \times 10^{-4} \quad p(D | h_{even}) = \left[\frac{1}{50} \right]^4 = 1.6 \times 10^{-7}$$

- ❖ 似然比 几乎是 5000:1
 - 所以, 预测 h 为 h_{two}
- ❖ 这意味着仅用单个最好的假设做预测



先验

❖ 给定样本 $D = \{16, 8, 2, 64\}$,

- “2的幂但不含32” 这个概念与 “2的幂” 相比，可能性更大

❖ 经验告诉我们

- $h_1 = \text{“2的幂但不含32”}$ 似乎 “在概念上不自然”

❖ 我们把这种经验叫做**先验(prior)**

- 先验： $p(h_1)$
- 通过将小先验概率 $p(h_1)$ 分配给不自然的概念，我们可以来捕捉这种直觉。



后验

❖ 后验概率:

$$p(h \mid D) = \frac{p(D \mid h) p(h)}{\sum_{h' \in H} p(D, h')} = \frac{p(h) \mathbf{I}(D \in h) / |h|^N}{\sum_{h' \in H} p(h') \mathbf{I}(D \in h') / |h'|^N}$$

❖ 在大多数概念的情况下先验是一致的

- 因此，后验与似然成比例

❖ “不自然的”概念，如“2的幂加37”和“2的幂但不含32”等

- 尽管似然性很高，但一般来说，先验概率小
- 当有足够的数据，后验概率 $p(h|D)$ 往往在单个概念上达到峰值，用这种方法估计，叫MAP估计



后验预测分布

❖ 贝叶斯平均模型

- 根据边缘概率规则，可得到：

$$p(\hat{x} \in C|D) = \sum_i p(y = 1|\hat{x}, h_i)p(h_i|D) = \sum_i p(X|h_i)p(h_i|d)$$

❖ 这意味着不只是用“最佳”假设做预测

❖ 每个假设的概率，可以通过贝叶斯规则得到：

$$P(h_i | \mathbf{d}) = \alpha P(\mathbf{d} | h_i)P(h_i) .$$

- 这样，学习就简化为概率推理.

❖ 贝叶斯平均模型的插件式近似（**plug-in approximation**）

$$p(\hat{x} \in C|D) = \sum_i p(y = 1|\hat{x}, h_i)\delta_{\hat{h}}(h_i) = p(\hat{x}|\hat{h})$$



一个“买糖果”的例子

❖ 我们最喜欢的糖果

- 两种口味：樱桃和酸橙。

❖ 每一块糖果，不管味道如何，都用相同的不透明纸包装

❖ 糖果装在大袋子里出售，有五种装法，但从外部无法区分

❖ 假设：装法

- h1: 100% 樱桃,
- h2: 75% 樱桃 + 25%酸橙,
- h3: 50% 樱桃+ 50%酸橙,
- h4: 25% 樱桃+ 75%酸橙,
- h5: 100%酸橙.



准备预测糖果的口味

❖ 给定每个假设的先验分布

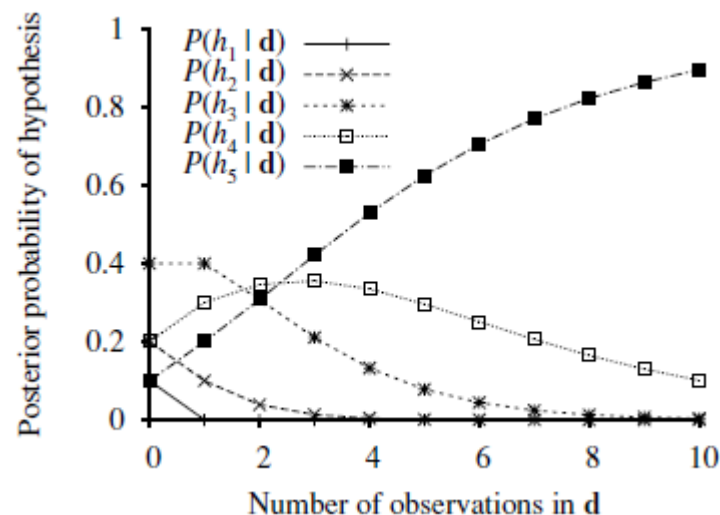
- $P(h_1) = 0.1, P(h_2)=0.2, P(h_3)=0.4, P(h_4) = 0.2, P(h_5)=0.1$

❖ 从袋中拿出10个糖果都是酸橙味的

- $P(\mathbf{d} | h_1) = ; P(\mathbf{d} | h_2) = ; P(\mathbf{d} | h_3) = 0.5*0.5*...*0.5 = 0.5^{10}, P(\mathbf{d} | h_4) = ; P(\mathbf{d} | h_5) = ;$

❖ 每个假设的后验概率 ($P(h_i | \mathbf{d}) = \alpha P(\mathbf{d} | h_i)P(h_i)$)

- $P(h_1 | \mathbf{d}) = 0$
- $P(h_2 | \mathbf{d}) = 0.25^{10} * 0.2$
- $P(h_3 | \mathbf{d}) = 0.5^{10} * 0.4$
- $P(h_4 | \mathbf{d}) = 0.75^{10} * 0.2$
- $P(h_5 | \mathbf{d}) = 0.1$



预测再拿出一颗糖果是哪种口味

❖ 用贝叶斯平均模型进行预测

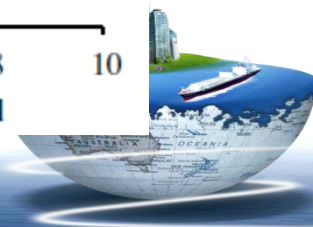
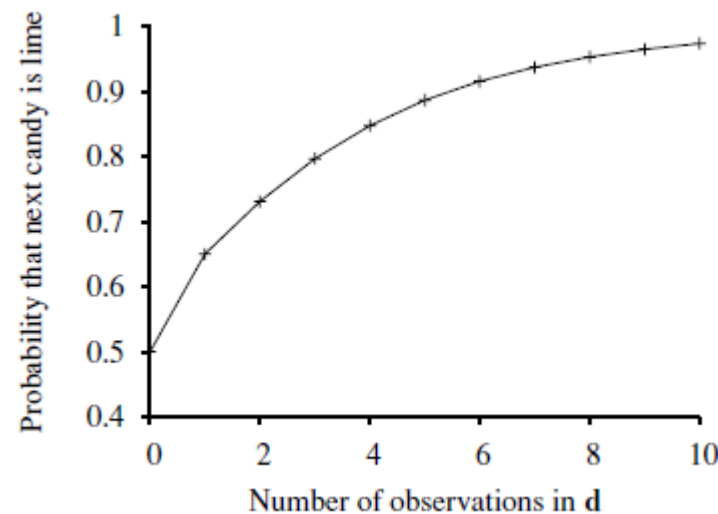
公式:
$$p(\hat{x} \in C|D) = \sum_i p(y = 1|\hat{x}, h_i)p(h_i|D) = \sum_i p(X|h_i)p(h_i|d)$$

计算:
$$p(x \in h_5|D) = 0 + 0.25 * 0.25^{10} * 0.2 + 0.5 * 0.5^{10} * 0.4 + 0.75 * 0.75^{10} * 0.2 + 1 * 0.1$$

❖ 插件式近似 (plug-in approximation)

公式:
$$p(\hat{x} \in C|D) = p(\hat{x}|\hat{h})$$

计算:
$$p(x \in h_5|D) = p(x|h_5) = 1$$



贝叶斯预测方法的特点

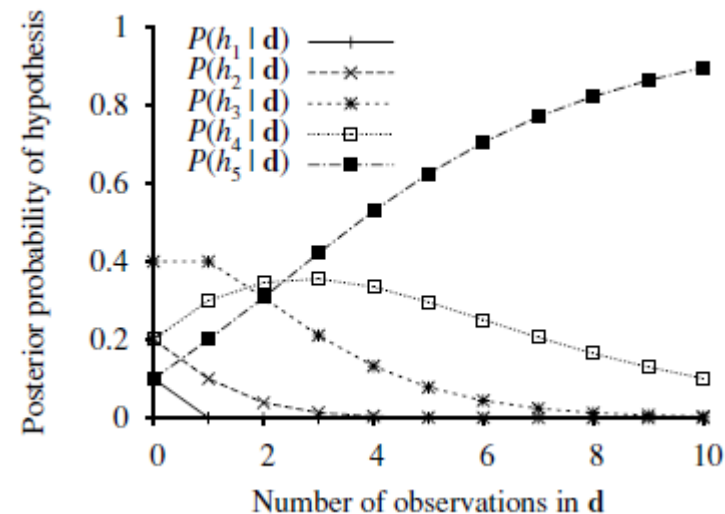
- ❖ 无论数据集大小，贝叶斯预测都是最优的。
- ❖ 贝叶斯学习的最优性是有代价的，
 - 对于实际学习问题，假设空间通常非常大或无限大。
- ❖ 在大多数情况下，我们必须采用近似或简化的方法。

- 常见的近似算法：

⑩ **最大后验(MAP) 假设** — h_{MAP} = 最大化 $P(h_i | \mathbf{d})$.

✓ 在糖果例子中，如果连续拿到三个酸橙味糖果，则有： $h_{\text{MAP}} = h_5$

⑩ **最大似然(ML) 假设** — h_{ML} = 最大化 $P(\mathbf{d} | h_i)P(h_i) = P(\mathbf{d} | h_i)$



β 二项分布模型



回顾一下数字游戏

❖ 数字游戏：

- 假设空间 $h \in H$ 是**有限的**
- 给定一系列离散观测值，计算特别简单：
 - ⑩ 只需要加、乘和除运算

❖ 在许多应用中，未知参数是连续的

- 假设空间是 \mathbf{R}^K 的（某个子集）
- 计算变得复杂：须用积分代替求和



另一种游戏：抛硬币

- ❖ 观察一系列抛硬币，推断正面朝上的概率。
- ❖ 该模型是许多方法的基础
- ❖ 该模型具有历史重要性
 - 它最早是1763年贝叶斯的论文中提出的



抛硬币模型的似然

- ❖ 设 $X_i \sim \text{Ber}(\theta)$,
 - $X_i = 1$ 表示“正面”, $X_i = 0$ 表示“反面”,
 - $\theta \in [0, 1]$ 为正面朝上的概率

- ❖ 如果数据是独立同分布, 则 似然:

$$p(D | \theta) = \theta^{N_1} (1 - \theta)^{N_0} \quad (\text{基于伯努利分布})$$

- $N_1 = \sum_{i=1}^N \mathbb{I}(x_i = 1)$ 正面朝上的个数, $N_0 = \sum_{i=1}^N \mathbb{I}(x_i = 0)$ 反面朝上的个数



抛硬币模型的先验

❖ 如果先验具有与似然相同的形式

- 计算起来更容易、更方便

❖ 如果先验看起来像：

$$p(\theta) \propto \theta^{\gamma_1} (1 - \theta)^{\gamma_0}$$

❖ 通过简单地将指数相加，就轻松评估后验值：

$$p(\theta \mid D) \propto p(D \mid \theta)p(\theta) = \theta^{N_1} (1 - \theta)^{N_0} \theta^{\gamma_1} (1 - \theta)^{\gamma_0} = \theta^{N_1 + \gamma_1} (1 - \theta)^{N_0 + \gamma_0}$$



共轭先验

❖ 当先验与后验具有相同的形式

- 先验称为对应似然的共轭先验

❖ 对于伯努利分布，共轭先验是 β 分布：

$$p(\theta \mid D) \propto \theta^{N_1} (1 - \theta)^{N_0} \times \frac{1}{B(a, b)} \theta^{a-1} (1 - \theta)^{b-1} = \frac{1}{B(a, b)} \theta^{N_1+a-1} (1 - \theta)^{N_0+b-1}$$



β 分布

❖ β 分布密度函数:

$$\text{Beta}(x | a, b) = \frac{1}{B(a, b)} x^{a-1} (1-x)^{b-1} \quad 0 \leq x \leq 1$$

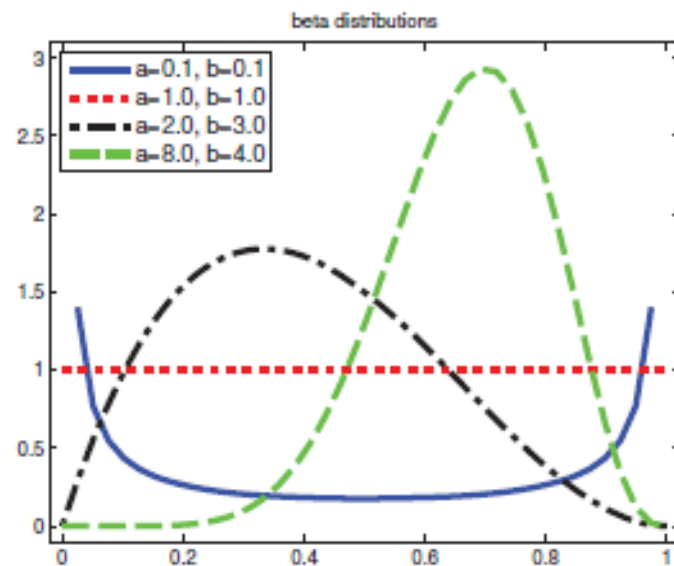
- $a = b = 1$, 则成为了均匀分布
- $a, b < 1$, 则成为了双峰分布, 分别在0和1处有“尖峰”
- $a, b > 1$, 则成为了单峰分布

$$\text{mean} = \frac{a}{a+b}, \quad \text{var} = \frac{ab}{(a+b)^2(a+b+1)}$$

- $B(a, b)$: β 函数
- $\Gamma(x)$: γ 函数

$$B(a, b) = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}$$

$$\Gamma(x) = \int_0^{\infty} u^{x-1} e^{-u} du$$



抛硬币模型的后验

❖ 参数的后验估计:

$$p(\theta \mid D) \propto \text{Bin}(N_1 \mid \theta, N_0 + N_1) \text{Beta}(\theta \mid a, b) = \text{Beta}(\theta \mid N_1 + a, N_0 + b)$$

❖ 后验预测分布

- 考虑预测单个未来试验中出现头部朝上的概率:

$$\begin{aligned} p(\tilde{x} = 1 \mid D) &= \int_0^1 p(x = 1, \theta \mid D) d\theta = \int_0^1 p(x = 1 \mid \theta) p(\theta \mid D) d\theta \\ &= \int_0^1 \theta \text{Beta}(\theta \mid a, b) d\theta = E[\theta \mid D] = \frac{a}{a + b} \end{aligned}$$



贝叶斯推理适合于在线学习

❖ 设：数据集 $D=D_a + D_b$

$$\begin{aligned} p(\theta \mid D_a, D_b) &= p(\theta, D_a, D_b) = p(D_a)p(\theta \mid D_a)p(D_b \mid \theta, D_a) \\ &\propto p(D_b \mid \theta)p(\theta \mid D_a) \\ &\propto \text{Bin}(N_1^b \mid \theta, N_1^b + N_0^b)\text{Beta}(\theta \mid N_1^a + a, N_0^a b) \\ &\propto \text{Beta}(\theta \mid N_1^a + N_1^b + a, N_0^a + N_0^b + b) \end{aligned}$$

❖ 这可以看出，贝叶斯推理特别适合在线学习



狄利克雷—多项式分布模型



掷骰子模型

- ❖ 我们想估计掷骰子（有 K 个面）游戏，出现第 i 个面的概率。
- ❖ 这种模型还被广泛用于
 - 文本数据分析、生物序列数据分析等。



- ❖ 这种模型又被称为多努利实验



独热编码 (one-hot encoding)

❖ **0-1向量** (每1项只能是0或1, 且只能一项为1)

- $(1, 0, 0, 0), (0, 1, 0, 0), (0, 0, 1, 0)$, 和 $(0, 0, 0, 1)$.

❖ **独热编码**

- 变量 $\mathbf{x} \in \{1, 2, \dots, k\}$
- 用独热编码表示 \mathbf{x}

$$\textcircled{10} \mathbf{x} = [I(\mathbf{x} = 1), \dots, I(\mathbf{x} = K)]$$



掷一次骰子(K 面)的概率分布

❖ 用独热编码表示骰子每个面出现的概率：

- $\theta = (\theta_1, \theta_2, \dots, \theta_K)$

❖ 概率分布可表示为

$$Cat(x | \theta) = Mu(x | 1, \theta) = \prod_{j=1}^K \theta_j^{I(x_j=1)}$$

- 这个分布称为 **分类分布 (categorical distribution)**
- 这个分布也称为 **多努利分布 (multinoulli distribution)**

❖ 如果 $x \sim Cat(\theta)$, 则 $p(x=j|\theta) = \theta_j$.



掷n次骰子(K面)的概率分布

❖ 设 $\mathbf{n} = (n_1, \dots, n_K)$ 为随机向量, 表示每个面出现的次数

■ $n_j \geq 0, n_1 + \dots + n_K = n.$

❖ 掷n次骰子服从多项分布

$$Mu(\mathbf{n} \mid n, \theta) = \binom{n}{n_1 \dots n_k} \prod_{j=1}^K \theta_j^{x_j} \quad \text{where} \quad \binom{n}{n_1 \dots n_k} = \frac{n!}{n_1! n_2! \dots n_K!}$$



掷n次骰子(K面)模型的似然

❖ 设：掷n次骰子(K面), $D = \{x_1, \dots, x_N\}$,

■ $x_i \in \{1, \dots, K\}$ 出现的概率为 θ_i

❖ 如果数据是独立同分布的, 则似然为：

$$p(D | \theta) = \prod_{k=1}^K \theta_k^{N_k}$$

■ $N_k = \sum_{i=1}^N I(y_i = k)$ 为第k面朝上的次数

❖ 多项分布的似然与该似然, 只相差一个常量系数



掷 n 次骰子(K 面)模型的先验

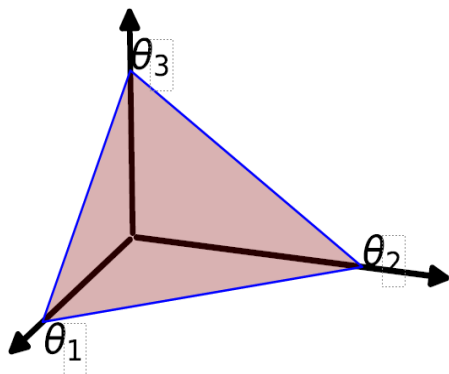
- ❖ 我们希望找到一个共轭先验
- ❖ 狄利克雷分布正好符合条件
- ❖ 因此，我们采用狄利克雷分布作为先验:

$$Dir(\theta | \alpha) = \frac{1}{B(\alpha)} \prod_{k=1}^K \theta_k^{\alpha_k - 1} I(x \in S_k)$$

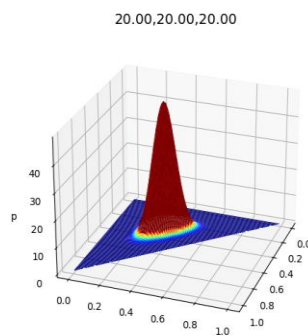


狄利克雷分布(Dirichlet distribution)

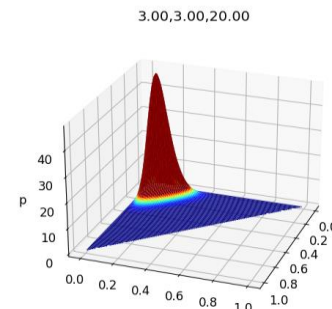
❖ Dirichlet distribution: $Dir(x | \alpha) = \frac{1}{B(\alpha)} \prod_{k=1}^K x_k^{\alpha_k - 1} I(x \in S_k) S_k = \left\{ \mathbf{x} : 0 \leq x_k \leq 1, \sum_{k=1}^K x_k = 1 \right\}$



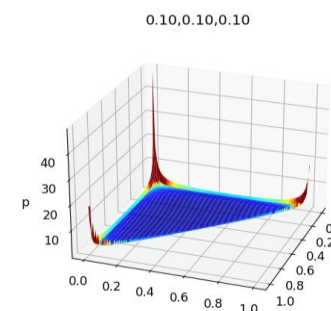
定义域: 等边三角面



$\alpha = (20, 20, 20).$



$\alpha = (3, 3, 20).$



$\alpha = (0.1, 0.1, 0.1).$

❖ $B(\alpha_1, \dots, \alpha_K)$ 为贝塔函数的自然泛化:

$$B(\alpha) = \frac{\prod_{k=1}^K \Gamma(\alpha_k)}{\Gamma(\alpha_0)} \quad \text{where } \alpha_0 = \sum_{k=1}^K \alpha_k$$



掷 n 次骰子(K 面)模型的后验

❖ 如果，似然为多项分布，先验为狄利克雷分布

■ 后验也是狄利克雷分布：

$$p(\theta | D) \propto p(D | \theta) p(\theta) = \text{Dir}(\theta | \alpha_1 + N_1, \dots, \alpha_K + N_K)$$

❖ 可以采用最大后验（MAP）估计分布的参数

■ 求解这个最优问题，可以采用拉格朗日乘数法（**Lagrange multiplier**）

$$l(\theta, \lambda) = \sum_k N_k \log \theta_k + \sum_k (\alpha_k - 1) \log \theta_k + \lambda (1 - \sum_k \theta_k)$$

❖ 求出最终估计值：

$$\hat{\theta}_k = \frac{N_k + \alpha_k - 1}{N + \alpha_0 - K}$$



掷n次骰子(K面)模型的预测分布

❖ 后验预测分布

- n次掷骰子后，再掷1次骰子第j面朝上的概率：

$$\begin{aligned} p(X = j \mid D) &= \int p(X = j, \theta \mid D) d\theta \\ &= \int p(X = j \mid \theta) p(\theta \mid D) d\theta = \frac{\alpha_j + N_j}{\alpha_0 + N} \end{aligned}$$



基于词袋模型表示语言序列

- ❖ 这是掷骰子模型的应用例子
- ❖ 观察下面序列（儿童童谣的一部分）：
 - Mary had a little lamb, little lamb, little lamb,
 - Mary had a little lamb, its fleece as white as snow
- ❖ 预测一下，序列后接下来可能出现的单词
- ❖ 基于词袋模型，我们将文本序列表示为向量



词袋模型

- ❖ 构造一个**词汇表**，包括我们观测文本中的所有单词
- ❖ 构造一个向量，其每个属性为词汇表中的一个单词，不考虑词的顺序
- ❖ 假设第*i*个单词是独立于其他单词进行采样
- ❖ 向量各属性的值 = 该单词出现次数

Token	1	2	3	4	5	6	7	8	9	10
Word	mary	lamb	little	big	fleece	white	black	snow	rain	unk
Count	2	4	4	0	1	1	0	1	0	4

- ❖ 向量中，做如下处理：
 - **unk**（未知）：文本中其他地方未出现的所有其他单词
 - 删除任何小词，如“a”、“as”、“the”等



预测文本序列后可能出现的词汇

❖ 如果 θ 的先验分布为 $Dir(\theta | \alpha)$,

❖ N_j . 表示单词序列中第 j 个单词出现的次数

❖ 后验预测分布:
$$p(\tilde{x} = j | D) = E[\theta_j | D] = \frac{\alpha_j + N_j}{\sum_i \alpha_i + N}$$

❖ 设 $\alpha_j = 1$, 有:

$$p(\tilde{X} = j \mid D) = \left(\begin{matrix} \text{mary} & \text{lamb} & \text{little} & \text{big} & \text{fleece} & \text{white} & \text{black} & \text{snow} & \text{rain} & \text{unk} \\ \frac{3}{27} & \frac{5}{27} & \frac{5}{27} & \frac{1}{27} & \frac{2}{27} & \frac{2}{27} & \frac{1}{27} & \frac{2}{27} & \frac{1}{27} & \frac{5}{27} \end{matrix} \right)$$

❖ 根据预测分布, 得到的预测结果:

■ $X = 2$ (“*lamb*”), $X = 3$ (“*little*”) and $X = 10$ (“*unk*”).



特别注意

❖ 单词 “big”, “black” 和 “rain” 预测概率非零

- 注意：它们在观测的文本中没有出现过

Token	1	2	3	4	5	6	7	8	9	10
Word	mary	lamb	little	big	fleece	white	black	snow	rain	unk
Count	2	4	4	0	1	1	0	1	0	4

$$p(\tilde{X} = j \mid D) = \left(\frac{3}{27}, \frac{5}{27}, \frac{5}{27}, \frac{1}{27}, \frac{2}{27}, \frac{2}{27}, \frac{1}{27}, \frac{2}{27}, \frac{1}{27}, \frac{5}{27} \right)$$



朴素贝叶斯分类器



朴素贝叶斯分类器(NBC)

- ❖ 特征向量 $\mathbf{x} \in \{1, \dots, K\}^D$
- ❖ 求一个类条件概率分布, $p(\mathbf{x}/y = c)$.
 - 这个问题也可以用掷骰子(K面)模型表达
- ❖ 设 $p(\mathbf{x}/y = c)$ 条件独立

$$p(\mathbf{x} | y = c, \boldsymbol{\theta}) = \prod_{j=1}^D p(x_j | y = c, \theta_{jc})$$

- ❖ 这个推理模型称为朴素贝叶斯分类器(NBC).



朴素贝叶斯分类器的常用条件密度分布

❖ 特征为实数时，可采用高斯分布：

$$p(\mathbf{x}/y = \mathbf{c}, \boldsymbol{\theta}) = \prod_{j=1}^D N(x_j | \mu_{jc}, \sigma_{jc}^2)$$

❖ 特征为二值时 $x_j \in \{0, 1\}$ ，可采用伯努利分布：

$$p(\mathbf{x}/y = \mathbf{c}, \boldsymbol{\theta}) = \prod_{j=1}^D \text{Ber}(x_j | \mu_{jc})$$

■ 这个模型又称为多变量伯努利朴素贝叶斯模型

❖ 特征为离散值时 $x_j \in \{1, \dots, K\}$ ，可采用多努利分布（分类分布）：

$$p(\mathbf{x}/y = \mathbf{c}, \boldsymbol{\theta}) = \prod_{j=1}^D \text{Cat}(x_j | \mu_{jc})$$

■ μ_{jc} 是 \mathbf{c} 类中 x_j 的 K 个可能值的直方图



训练朴素贝叶斯分类器

❖ 通常训练朴素贝叶斯分类器的方法.

- 最大似然估计法（MLE），估计模型参数
- 最大后验估计法（MAP），估计模型参数



基于最大似然估计的朴素贝叶斯分类器训练

❖ 单个数据的概率

$$p(\mathbf{x}_i, y_i | \boldsymbol{\theta}) = p(y_i | \boldsymbol{\pi}) \prod_j p(x_{ij} | \theta_j) = \prod_c \pi_c^{I(y_i=c)} \prod_j \prod_c p(x_{ij} | \theta_{jc})^{I(y_i=c)}$$

- 概率分布形式为狄利克雷分布

❖ log似然函数:

$$\log p(D | \boldsymbol{\theta}) = \sum_{c=1}^C N_c \log \pi_c + \sum_{j=1}^D \sum_{c=1}^C \sum_{i: y_i=c} \log p(x_{ij} | \theta_{jc})$$

❖ 最大似然求得参数结果:

$$\hat{\pi}_c = \frac{N_c}{N} \quad \hat{\theta}_{jc} = \frac{N_{jc}}{N_c}$$



案例：垃圾邮件检测

训练数据:

❖ 垃圾邮件（共9个词）

- Offer is secret
- Click secret link
- Secret sports link

❖ 正常邮件（共15个词）

- Play sports today
- Went play sports
- Secret sports event
- Sport is today
- Sport costs money

测试数据:

❖ 给定邮件

- M = "Secret is secret"

❖ 请问是垃圾邮件还是正常邮件？



模型参数估计：最大似然估计

❖ D : x=SSSHHHHH, y=11100000

❖ 设 : $P(S|\theta)=\theta$, $P(H|\theta)=1-\theta$

❖ 计算最大似然 :

$$p(y_i) = \begin{cases} \theta, & \text{if } y_i = 1 \\ 1 - \theta & \text{if } y_i = 0 \end{cases} = \theta^{y_i} \cdot (1 - \theta)^{1-y_i}$$

❖ 似然函数 :

$$p(D|\theta) = \theta^{\text{count}(y_i=1)} \cdot (1 - \theta)^{\text{count}(1-y_i)}$$

❖ 本问题使用log似然 :

$$\log(p(D|\theta)) = 3\log\theta + 5\log(1 - \theta)$$

$$\frac{\partial \log P(D|\theta)}{\partial \theta} = \frac{3}{\theta} + \frac{5}{1 - \theta} = 0$$

$$\frac{\partial \log P(D|\theta)}{\partial \theta} = \frac{3}{\theta} - \frac{5}{1 - \theta} = 0$$

$$\theta = \frac{3}{8}$$



预测给定邮件是否是垃圾邮件

❖ 给定邮件: $M = \text{"Secret is secret"}$

已知: $P(spam) = \frac{3}{8} \cdot$ $P(ham) = \frac{5}{8} \cdot$

计算: $P(M|spam) = P("secret"|spam) \cdot P("is"|spam) \cdot P("secret"|spam) = \frac{3}{9} \cdot \frac{1}{9} \cdot \frac{3}{9} = \frac{1}{81}$

$$P(M|ham) = \frac{1}{15} \cdot \frac{1}{15} \cdot \frac{1}{15} = \frac{1}{3375}$$

根据贝叶斯公式: $P(spam|M) = \frac{P(spam, M)}{P(M)} = \frac{P(spam) \cdot P(M|spam)}{P(M|spam)P(spam) + P(M|ham)P(ham)} = \frac{25}{26}$

❖ 所以, 预测该邮件是垃圾邮件



MLE vs. MAP

❖ **MLE:**

$$\hat{\theta} = \operatorname{argmax}_{\theta} p(D|\theta)$$

❖ **MAP:**

$$\hat{\theta} = \operatorname{argmax}_{\theta} \frac{p(D|\theta)p(\theta)}{p(D)} = \operatorname{argmax}_{\theta} p(D|\theta)p(\theta)$$



最大似然估计存在过拟合问题

❖ 对于文本分类，如果采用最大似然估计，我们得到：

$$\hat{\theta}_{jc} = \frac{N_{jc}}{N_c}$$

❖ 可以看出，这里只考虑词汇在样本中出现的次数

❖ 如果属于c类的样本中，没有第j个词，则会有： $\hat{\theta}_{jc} = 0$

■ 这就出现了过拟合

❖ 可以用拉普拉斯平滑法，解决这个问题



拉普拉斯平滑

❖ 似然函数：

$$P(x) = \text{count}(x) / N$$

❖ 为了克服过拟合问题，对似然做拉普拉斯平滑

- $P(x) = (\text{count}(x) + 1) / (N + 2)$

- **基本思路**：假设实际采样前，已有两个样本，一个正样本，一个负样本，因此，分子加1和分母加2

❖ 判定邮件是否垃圾邮件：M = “Today is secret”

- 若采用似然估计： $P(\text{“Today is Secret”} | \text{Spam}) = 0$ ，表明过拟合了

- 用拉普拉斯平滑：

$$P(M | \text{spam}) = \frac{1}{11} \cdot \frac{2}{11} \cdot \frac{3}{11} = 0.0045$$

$$P(M | \text{ham}) = \frac{3}{17} \cdot \frac{2}{17} \cdot \frac{2}{17} = 0.0024$$

❖ 所以，预测该邮件是垃圾邮件



特征选择

- ❖ 朴素贝叶斯分类器是拟合多个特征的联合分布
 - 它可能受到过拟合的影响
 - 运行时间复杂度为 $O(D)$ ，对于某些应用程序来说可能太高。
- ❖ 一种常用的方法是做特征选择，删除“不相关”特征
- ❖ 最简单的方法是分别度量每个特征的相关性
 - 估计相关性的一种方法是计算特征 X_j 和类标签 Y 之间的互信息
 - 取前 K 个最相关的特征， K 的选择要基于平衡精确性与复杂性
 - 这种方法被称为变量排序、过滤或筛选。



熵 (*Entropy*)

❖ 熵是一种随机变量的不确定性度量

$$Entropy(X) = \int P(X) uncertainty(X) dX$$

❖ 定义不确定性

$$uncertainty(X) = \log_2 \frac{1}{P(X)}$$

❖ 设离散随机变量 \mathbf{X} 有 \mathbf{K} 个状态，其熵表示为：

$$H(X) = \sum_{k=1}^K P(X = k) \log_2 \frac{1}{P(X = k)} = - \sum_{k=1}^K P(X = k) \log_2 P(X = k)$$



熵计算的例子

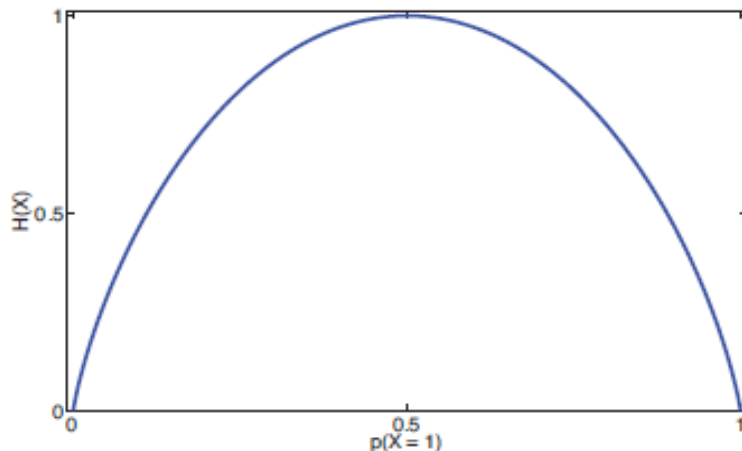
❖ 对于二值随机变量 $X \in \{0, 1\}$, 其概率:

■ $p(X = 1) = \theta$ and $p(X = 0) = 1 - \theta$.

❖ 熵计算如下

$$\begin{aligned} H(X) &= -[P(X = 1)\log_2 P(X = 1) + P(X = 0)\log_2 P(X = 0)] \\ &= -[\theta\log_2 \theta + (1 - \theta)\log_2 (1 - \theta)] \end{aligned}$$

❖ 熵随概率值变化曲线:



KL 散度(*divergence*)

❖ 一种度量两个随机变量 p 和 q 的分布的不相似性方法

- *the Kullback-Leibler divergence*
- 也称为相对熵 (*relative entropy*)

$$KL(p \parallel q) = \sum_k p_k \log \frac{p_k}{q_k}$$

❖ 交叉熵(*cross entropy*) :

$$H(p, q) = -\sum_k p_k \log q_k$$

❖ KL散度与熵和交叉熵的关系 :

$$KL(p \parallel q) = \sum_k p_k \log p_k - \sum_k p_k \log q_k = -H(p) + H(p, q)$$



KL 距离

❖ 希望利用KL散度定义两个分布之间的距离

- 但 **KL 散度**本身不是距离度量

- ⑩ KL散度不具有对称性: $KL(p||q) \neq KL(q||p)$

- ⑩ KL散度不是恒大于等于0

❖ 我们可以定义KL距离如下:

- $0.5KL(p||q) + 0.5KL(q||p)$



互信息(Mutual information)

❖ 如何判定联合分布 $p(X, Y)$ 与因子化分布 $p(X)p(Y)$ 的相似性

■ 可以运用互信息:

$$I(X, Y) = KL(p(X, Y) \parallel p(X)p(Y)) = \sum_{x_j} \sum_y p(x_j, y) \log \frac{p(x_j, y)}{p(x_j)p(y)}$$

❖ $I(X; Y) \geq 0$ (iff $p(X, Y) = p(X)p(Y)$, 式子取等号)



度量相关性

- ❖ 度量相关性的一种方法就是计算特征 X_j 和类标签 Y 之间的互信息：

$$I(X, Y) = \sum_{x_j} \sum_y p(x_j, y) \log \frac{p(x_j, y)}{p(x_j)p(y)}$$

- ❖ 如果特征是二值的，则互信息可写成：

$$I_j = \sum_c \left[\theta_{jc} \pi_c \log \frac{\theta_{jc}}{\theta_j} + (1 - \theta_{jc}) \pi_c \log \frac{1 - \theta_{jc}}{1 - \theta_j} \right]$$

- ❖ 其中， $\pi_c = p(y=c)$, $\theta_{jc} = p(x_j=1|y=c)$, $\theta_j = p(x_j=1) = \sum_c \pi_c \theta_{jc}$



分类文本

- ❖ 采用词袋模型
- ❖ 从不同的角度去看这个问题，可以得到不同的分类器
- ❖ 我们来看3个不同的文本分类器
 - 简单分类器
 - 更精确些的分类器
 - 更好的分类器



简单文本分类器

- ❖ 基于词袋模型，用二值向量表示文档，记录每个单词是否存在于文档中
 - 如果单词 j 出现在文档 i 中，则 $x_{ij} = 1$ ，否则 $x_{ij} = 0$
- ❖ 可以用类条件概率来分类：

$$p(x_i | y_i = c, \theta) = \prod_{j=1}^D \text{Ber}(x_{ij} | \theta_{jc}) = \prod_{j=1}^D \theta_{jc}^{I(x_{ij})} (1 - \theta_{jc})^{I(1-x_{ij})}$$

- ❖ 这种方法的问题
 - 忽略了每个单词出现的次数
 - 丢失了一些信息



更精确些的文本分类器

- ❖ 考虑了每个单词出现在文档中的次数
- ❖ 设 \mathbf{x}_i 为向量，表示每个单词出现在文档 i 中的次数, $x_{ij} \in \{0, 1, \dots, N_{ij}\}$,
 - N_i 为文档 i 中总单词数 ($\sum_{j=1}^D x_{ij} = N_i$)
- ❖ 可以使用多项分布来分类.

$$p(\mathbf{x}_i | y_i = c, \theta) = \text{Mu}(\mathbf{x}_i | N_i, \theta_c) = \frac{N_i!}{\prod_{j=1}^D x_{ij}!} \prod_{j=1}^D \theta_{jc}^{x_{ij}}$$

- ❖ 假设
 - 文档大小 N_i 不依赖于类别.
 - 单词 j 存在于 C 类文档的概率为: θ_{jc}
 - 对于每个文档类别, 参数满足约束: ($\sum_{j=1}^D \theta_{jc} = 1$)



更好的文本分类器

- ❖ 基于多项分布的分类器对于文档分类不是特别好。
 - 单词出现具有突发性。
 - 大多数单词从未出现在任何给定文档中
 - 如果它们出现一次，则可能出现多次
- ❖ 我们可以使用**Dirichlet**平均多项分布的方法进行分类

$$p(x_i | y_i = c, \alpha) = \int \text{Mu}(x_i | N_i, \theta_c) \text{Dir}(\theta_c | \alpha_c) d\theta_c = \frac{N_i!}{\prod_{j=1}^D x_{ij}!} \frac{B(x_i + \alpha_c)}{B(\alpha_c)}$$

