机器学习与深度学习

目录

- LR和最大熵
- HMM和EM
- CRF
- Topic Model

很多资料来自于《统计学习方法》--李航

逻辑斯特回归和最大熵

Logistic Regression

- Sigmoid函数：

$$f(z) = \frac{1}{1 + \exp(-z)}$$
$$f'(z) = f(z)(1 - f(z))$$

[0,1]

二分类

- Binomial logistic regression model
  - 由条件概率P(Y|X)表示的分类模型
  - 形式化为logistic distribution
  - X取实数，Y取值1,0

$$P(Y=1|x)=\frac{\exp(w\cdot x+b)}{1+\exp(w\cdot x+b)}$$

$$P(Y=0|x)=\frac{1}{1+\exp(w\cdot x+b)}$$

多分类

- 设Y的取值集合为

$$\{1,2,\cdots,K\}$$

- 多项logistic回归模型

$$P(Y=k|x)=\frac{\exp(w_k\cdot x)}{1+\sum_{k=1}^{K-1}\exp(w_k\cdot x)},\quad k=1,2,\cdots,K-1$$

$$P(Y=K|x)=\frac{1}{1+\sum_{k=1}^{K-1}\exp(w_k\cdot x)}$$

二项逻辑斯蒂回归

- 事件的几率odds：事件发生与事件不发生的概率之比为

$$\frac{p}{1-p}$$

- 称为事件的发生比(the odds of experiencing an event)，对数几率：

$$\mathrm{logit}(p) = \log\frac{p}{1-p}$$

- 对逻辑斯蒂回归

$$\log\frac{P(Y=1|x)}{1-P(Y=1|x)} = w \cdot x$$

$$P(Y=1|x) = \frac{\exp(w \cdot x)}{1+\exp(w \cdot x)}$$

参数估计

- 那么对于上述m个观测事件，设

$$P(Y=1|x)=\pi(x), \qquad P(Y=0|x)=1-\pi(x)$$

- 其联合概率密度函数，即似然函数为：

N个样本发生某种情况的概率

$$\prod_{i=1}^{N}[\pi(x_i)]^{y_i}\,[1-\pi(x_i)]^{1-y_i}$$

- 目标：求出使这一似然函数的值最大的参数估，$w_1, w_2, \cdots, w_n$，使得L(w)取得最大值。

模型参数估计

- 对数似然函数

$$L(w)=\sum_{i=1}^{N}[y_i\log\pi(x_i)+(1-y_i)\log(1-\pi(x_i))]$$

$$=\sum_{i=1}^{N}[y_i\log\frac{\pi(x_i)}{1-\pi(x_i)}+\log(1-\pi(x_i))]$$

$$=\sum_{i=1}^{N}[y_i(w\cdot x_i)\ -\log(1+\exp(w\cdot x_i)]$$

- 对L(w)求极大值，得到w的估计值。
- 通常采用梯度下降法及拟牛顿法，学到的模型:

$$P(Y=1|x)=\frac{\exp(\widehat{w}\cdot x)}{1+\exp(\widehat{w}\cdot x)}\quad P(Y=0|x)=\frac{1}{1+\exp(\widehat{w}\cdot x)}$$

最大熵模型

- 最大熵原理: 学习概率模型时，在所有可能的概率模型(分布)中，熵最大的模型是最好的模型，表述为在满足约束条件的模型集合中选取熵最大的模型。
- 假设离散随机变量X的概率分布是P(X)， 熵:

$$\mathrm{H}(P) = -\sum_{x} P(x)\log P(x)$$

- 且:

$$0 \leq \mathrm{H}(P) \leq \log|X|$$

- |X|是X的取值个数，X均匀分布时右边等号成立。

最大熵例子

- 假设随机变量X有5个取值{A,B,C,D,E},估计各个值的概率。
- 解：满足 P(A)+P(B)+P(C)+P(D)+P(E)=1
- 等概率估计：

$$P(A) = P(B) = P(C) = P(D) = P(E) = \frac{1}{5}$$

- 加入一些先验：

$$P(A) + P(B) = \frac{3}{10}$$
$$P(A) + P(B) + P(C) + P(D) + P(E) = 1$$

- 于是：

$$P(A) = P(B) = \frac{3}{20}$$
$$P(C) = P(D) = P(E) = \frac{7}{30}$$

再加入约束：

$$P(A) + P(C) = \frac{1}{2}$$
$$P(A) + P(B) = \frac{3}{10}$$
$$P(A) + P(B) + P(C) + P(D) + P(E) = 1$$

最大熵原理

- 输入：X，类别：Y，条件概率：P(Y|X)
- 给定数据集：

$$T=\{(x_1,y_1),(x_2,y_2),\cdots,(x_N,y_N)\}$$

- 联合分布P(Y|X)的经验分布，边缘分布P(X)的经验分布：

$$\tilde{P}(X,Y) \Rightarrow \tilde{P}(X=x,Y=y)=\frac{\nu(X=x,Y=y)}{N}$$

$$\tilde{P}(X) \Rightarrow \tilde{P}(X=x)=\frac{\nu(X=x)}{N}$$

- 特征函数：

$$f(x,y)=\begin{cases}1, & x\text{与}y\text{满足某一事实}\\ 0, & \text{否则}\end{cases}$$

最大熵原理

- 特征函数f(x,y)关于经验分布的期望值：

$$E_{\tilde{P}}(f) = \sum_{x,y} \tilde{P}(x,y) f(x,y)$$

- 特征函数f(x,y)关于模型P(Y|X)与经验分布$\tilde{P}(X)$的期望值：

$$E_P(f) = \sum_{x,y} \tilde{P}(x) P(y|x) f(x,y)$$

如果模型能够获取训练数据中的信息，可以假设这两个期望值相等，即

$$E_P(f) = E_{\tilde{P}}(f) \quad \Rightarrow \quad \sum_{x,y} \tilde{P}(x) P(y|x) f(x,y) = \sum_{x,y} \tilde{P}(x,y) f(x,y)$$

- 假设有n个特征函数$f_i(x,y)$，$i = 1,2,\cdots,n$，那么就有n个约束条件

最大熵模型

- 假设满足所有约束条件的模型集合为:

$$\mathcal{C} = \{P \in \mathcal{P} | E_P(f_i), \qquad i = 1,2,\cdots,n\}$$

- 定义在条件概率分布P(Y|X)上的条件熵:

$$H(P) = -\sum_{x,y} \tilde{P}(x) P(y|x) \log P(y|x)$$

- 则模型集合C中条件熵H(P)最大的模型称为最大熵模型

最大熵模型的学习

- 最大熵模型的学习等价于约束最优化问题:

$$\max_{p\in C} H(P) = -\sum_{x,y} \tilde{P}(x)P(y|x)\log P(y|x)$$

$$s.t.\ \ E_P(f_i) = E_{\tilde{P}}(f_i), \qquad i = 1,2,\cdots,n$$

$$\sum_y P(y|x) = 1$$

$$\min_{p\in C} -H(P) = \sum_{x,y} \tilde{P}(x)P(y|x)\log P(y|x)$$

$$s.t.\ \ E_P(f_i) - E_{\tilde{P}}(f_i) = 0, \qquad i = 1,2,\cdots,n$$

$$\sum_y P(y|x) = 1$$

最大熵模型的学习

- 引进拉格朗日乘子，定义拉格朗日函数，转化为无约束优化问题:

$$L(P,w) \equiv -H(P) + w_0\left(1 - \sum_y P(y|x)\right) + \sum_{i=1}^{n} w_i\left(E_{\tilde{P}}(f_i) - E_P(f_i)\right)$$

$$= \sum_{x,y} \tilde{P}(x)P(y|x)\log P(y|x) + w_0\left(1 - \sum_y P(y|x)\right)$$

$$+ \sum_{i=1}^{n} w_i\left(\sum_{x,y} \tilde{P}(x,y)f_i(x,y) - \sum_{x,y} \tilde{P}(x)P(y|x)f_i(x,y)\right)$$

- 最优化原始问题 到 对偶问题:

$$\min_{p\in C}\max_{w} L(P,w) \Rightarrow \max_{w}\min_{p\in C} L(P,w)$$

最大熵模型的学习

- 先求极小化问题: $\min_{P \in C} L(P, w)$是w的函数,

$$\psi(w) = \min_{P \in C} L(P, w) = L(P_w, w)$$

$$P_w = \arg\min_{P \in C} L(P, w) = P_w(y|x)$$

最大熵模型的学习

- 求L(P,w)对P(y|x)的偏导数:

$$\frac{\partial L(P,w)}{\partial P(y|x)} = \sum_{x,y} \tilde{P}(x)\,(\log P(y|x) + 1) - \sum_{y} w_0$$

$$- \sum_{x,y} \left(\tilde{P}(x) \sum_{i=1}^{n} w_i f_i(x,y) \right)$$

$$= \sum_{x,y} \tilde{P}(x) \left(\log P(y|x) + 1 - w_0 - \sum_{i=1}^{n} w_i f_i(x,y) \right)$$

- 得:

$$P(y|x) = \exp\left(\sum_{i=1}^{n} w_i f_i(x,y) + w_0 - 1 \right) = \frac{\exp(\sum_{i=1}^{n} w_i f_i(x,y))}{\exp(1 - w_0)}$$

最大熵模型的学习

- 由：

$$\sum_{y} P(y|x) = 1$$

- 得：

$$P_w(y|x) = \frac{1}{Z_w(x)} \exp\left(\sum_{i=1}^{n} w_i f_i(x,y)\right) \tag{6.22}$$

- 规范化因子:

$$Z_w(x) = \sum_{y} \exp\left(\sum_{i=1}^{n} w_i f_i(x,y)\right) \tag{6.23}$$

- 模型$P_w = P_w(y|x)$就是最大熵模型
- 求解对偶问题外部的极大化问题:

$$\max_{w} \Psi(w) \qquad w^* = arg \max_{w} \Psi(w) \qquad P^* = P_{w^*} = P_{w^*}(\mathrm{y}|\mathrm{x})$$

例子

- $y_1 y_2 y_3 y_4 y_5$分别代表A,B,C,D,E，其中P(A)+P(B)=3/10
- 最大熵模型:

$$\min -H(P) = \sum_{i=1}^{5} P(y_i)\log P(y_i)$$

$$s.t.\ \ P(y_1) + P(y_2) = \tilde{P}(y_1) + \tilde{P}(y_2) = \frac{3}{10}$$

$$\sum_{i=1}^{5} P(y_i) = \sum_{i=1}^{5} \tilde{P}(y_i) = 1$$

$$L(P,w) = \sum_{i=1}^{5} P(y_i)\log P(y_i) \ + w_1\left(P(y_1) + P(y_2) - \frac{3}{10}\right) + w_0\left(\sum_{i=1}^{5} P(y_1) - 1\right)$$

$$\max_{w} \min_{P} L(P,w)$$

例子：

$$\frac{\partial L(P,w)}{\partial P(y_1)} = 1 + \log P(y_1) + w_1 + w_0$$
$$\frac{\partial L(P,w)}{\partial P(y_2)} = 1 + \log P(y_2) + w_1 + w_0$$
$$\frac{\partial L(P,w)}{\partial P(y_3)} = 1 + \log P(y_3) + w_0$$
$$\frac{\partial L(P,w)}{\partial P(y_4)} = 1 + \log P(y_4) + w_0$$
$$\frac{\partial L(P,w)}{\partial P(y_5)} = 1 + \log P(y_5) + w_0$$

解得：

$$P(y_1) = P(y_2) = e^{-w_1 - w_0 - 1}$$
$$P(y_3) = P(y_4) = P(y_5) = e^{-w_0 - 1}$$

例子：

$$\min_{P} L(P,w) = L(P_w,w) = -2e^{-w_1-w_0-1} - 3e^{-w_0-1} - \frac{3}{10}w_1 - w_0$$

- 得：

$$\max_{w} L(P_w,w) = -\,2e^{-w_1-w_0-1} - 3e^{-w_0-1} - \frac{3}{10}w_1 - w_0$$

- 对wi求偏导并令为0:

$$e^{-w_1-w_0-1} = \frac{3}{20}$$

$$e^{-w_0-1} = \frac{7}{30}$$

$$\Rightarrow$$

$$P(y_1) = P(y_2) = \frac{3}{20}$$

$$P(y_3) = P(y_4) = P(y_5) = \frac{7}{30}$$

隐马尔科夫模型

马尔科夫模型

- 存在一类重要的随机过程：如果一个系统有 N 个状态 $S_1, S_2, \ldots, S_N$, 随着时间的推移，该系统从某一状态转移到另一状态。如果用 q_t 表示系统在时间 t 的状态变量，那么，t 时刻的状态取值为 S_j $(1 \le j \le N)$ 的概率取决于前 t-1 个时刻 (1, 2, …, t-1) 的状态，该概率为：

$$P(q_t = S_j \mid q_{t-1} = S_i, q_{t-2} = S_k, \cdots)$$

- 系统在时间 t 的状态只与其在时间 t-1 的状态相关

$$P(q_t = S_j \mid q_{t-1} = S_i, q_{t-2} = S_k, \cdots) = P(q_t = S_j \mid q_{t-1} = S_i)$$

$$P(q_t = S_j \mid q_{t-1} = S_i) = a_{ij}, \quad 1 \le i, j \le N \qquad a_{ij} \ge 0 \qquad \sum_{j=1}^{N} a_{ij} = 1$$

例子

$$P(S_1,\cdots,S_T)=P(S_1)P(S_2\mid S_1)P(S_3\mid S_1,S_2)\cdots P(S_T\mid S_1,\cdots,S_{T-1})$$

$$=P(S_1)P(S_2\mid S_1)P(S_3\mid S_2)\cdots P(S_T\mid S_{T-1})$$

$$=\pi_{S_1}\prod_{t=1}^{T-1}a_{S_tS_{t+1}}$$

$$P(t,i,p)=P(S_1=t)\times P(S_2=i\mid S_1=t)\times P(S_3=p\mid S_2=i)$$

$$=1.0\times 0.3\times 0.6$$

$$=0.18$$

隐马尔可夫模型

- 隐马尔可夫模型是关于时序的概率模型，是一个双重随机过程
- 描述由一个隐藏的马尔可夫链随机生成不可观测的状态随机序列(state sequence)， 再由各个状态生成一个观测而产生观测随机序列（observation sequence）的过程,序列的每一个位置又可以看作是一个时刻。

隐马尔科夫模型

- 组成
  - 初始概率分布
  - 状态转移概率分布
  - 观测概率分布
  - Q： 所有可能状态的集合
  - V： 所有可能观测的集合
  - I: 长度为T的状态序列

$$I = \{i_1, i_2, \cdots, i_T\}, \qquad Q = \{q_1, q_2, \cdots, q_N\},$$

  - O： 对应的观测序列

$$V = \{v_1, v_2, \cdots, v_M\}, \qquad O = \{o_1, o_2, \cdots, o_T\}$$

隐马尔科夫模型

- A： 状态转移概率矩阵

$$A = [a_{ij}]_{N\times N}$$

$$a_{ij} = P(i_{i+1} = q_j | i_i = q_i), \qquad \mathrm{i} = 1,2,\cdots,N;\ j = 1,2,\cdots,N$$

时刻t处于状态q_i的条件下在时刻t+1转移到状态q_j的概率

- B: 观测概率矩阵

$$B = [b_j]_{N\times M}$$

$$b_j(k) = P(o_t = v_k | i_t = q_j), \qquad \mathrm{k} = 1,2,\cdots,M;\ j = 1,2,\cdots,N$$

在时刻t处于状态q_j的条件下生成观测v_k的概率

- π: 初始状态概率向量

$$\pi_i = P(i_1 = q_i), \qquad i = 1,2,\cdots,N$$

时刻t=1处于状态q_i的概率

隐马尔科夫模型

- 关键

$$\lambda = (\mathrm{A}, \mathrm{B}, \pi)$$

- 基本假设

  - 齐次马尔科夫性假设， 隐马尔可分链t的状态只和t-1状态有关：

$$P(i_t | i_{t-1}, o_{t-1}, \cdots, i_1, o_1) = P(i_t | i_{t-1}), \qquad t = 1,2,\cdots,T$$

  - 观测独立性假设， 观测只和当前时刻状态有关；

$$P(o_t | i_T, o_T, i_{T-1}, o_{T-1}, \cdots, i_{t+1}, o_{t+1}, i_t, i_{t-1}, o_{t-1}, \cdots, i_1, o_1) = P(o_t | i_t)$$

例子

- 盒子： 1 2 3 4
- 红球： 5 3 6 8
- 白球： 5 7 4 2

例子

- 转移规则：
  - 盒子1 下一个 盒子2
  - 盒子2或3 下一个 0.4 左， 0.6右
  - 盒子4 下一个 0.5 自身， 0.5盒子3
- O={ 红， 红， 白， 白， 红}
- 状态集合： Q={盒子1， 盒子2， 盒子3， 盒子4}， N=4
- 观测集合： V={红球， 白球} M=2
- 初始化概率分布：

- 状态转移矩阵：$\pi = (0.25,0.25.0.25,0.25)^T$ 观测矩阵：

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0.4 & 0 & 0.6 & 0 \\ 0 & 0.4 & 0 & 0.6 \\ 0 & 0 & 0.5 & 0.5 \end{bmatrix} \qquad B = \begin{bmatrix} 0.5 & 0.5 \\ 0.3 & 0.7 \\ 0.6 & 0.4 \\ 0.8 & 0.2 \end{bmatrix}$$

隐马尔柯夫模型观察序列生成

输入：隐马尔可夫模型$\lambda = (A, B, \pi)$，观测序列长度T；
输出：观测序列$O = (o_1, o_2, \cdots, o_T)$.
(1) 按照初始状态分布π产生状态i_1
(2) 令t=1
(3) 按照状态i_t的观测概率分布$b_{i_t}(k)$生成o_t
(4) 按照状态i_t的状态转移概率分布$\{a_{i_t, i_{t+1}}\}$产生状态i_{t+1}，$i_{t+1} = 1,2, \cdots, N$
(5) 令t=t+1；如果t<T，转步(3)；否则，终止

三个关键问题

- 在给定模型 $\mu=(A,B,\pi)$ 和观察序列 $O=O_1O_2\ldots O_T$ 的情况下，怎样快速计算概率 $P(O\mid\mu)$ ？（概率计算问题）
- 在给定模型 $\mu=(A,B,\pi)$ 和观察序列 $O=O_1O_2\ldots O_T$ 的情况下，如何选择在一定意义下“最优” 的状态序列 $Q=q_1q_2\cdots q_T$ ，使得该状态序列“最好地解释”观察序列 $O=O_1O_2\ldots O_T$？（预测问题）
- 给定一个观察序列 $O=O_1O_2\ldots O_T$，如何根据最大似然估计来求模型的参数值？即如何调节模型 $\mu=(A,B,\pi)$ 的参数，使得 $P(O\mid\mu)$ 最大？（学习问题）

概率计算方法

- 直接计算法
- 状态序列$I=(i_1,i_2,\cdots,i_T)$概率：$P(I|\lambda)=\pi_{i_1}a_{i_1i_2}a_{i_2i_3}\cdots a_{i_{T-1}i_T}$
- 对固定的状态序列I，观测序列O的概率：$P(O|I,\lambda)$
- O和I同时出现的联合概率为：$P(O|I,\lambda)=b_{i_1}(o_1)b_{i_2}(o_2)\cdots b_{i_T}(o_T)$
- 对所有可能的状态序列I求和，得到观测O的概率：

$$P(O,I|\lambda)=P(O|I,\lambda)P(I|\lambda)$$
$$=\pi_{i_1}b_{i_1}(o_1)a_{i_1i_2}b_{i_2}(o_2)\cdots a_{i_{T-1}i_T}b_{i_T}(o_T)$$

$$P(O|\lambda)=\sum_I P(O|I,\lambda)P(I|\lambda)$$
$$=\sum_{i_1,i_2,\cdots,i_T}\pi_{i_1}b_{i_1}(o_1)a_{i_1i_2}b_{i_2}(o_2)\cdots a_{i_{T-1}i_T}b_{i_T}(o_T)$$

概率计算方法

如果模型μ=(A, B, π) 有 N 个不同的状态，时间长度为 T，那么有 N^T 个可能的状态序列，搜索路径成指数级组合爆炸。

$O(TN^T)$

前向算法

- 前向概率定义： 给定隐马尔科夫模型λ， 定义到时刻t部分观测序列为：$o_1, o_2, \cdots, o_t$ ， 且状态为q_i的概率为前向概率，记作：

$$\alpha_T(i) = P(o_1, o_2, \cdots, o_t, i_T = q_i|\lambda)$$

前向算法（动态规划）

输入：隐马尔科夫模型λ，观测序列O；

输出：观测序列概率$P(O|\lambda)$。

- 初值：$\alpha_1(i) = \pi_{i_1} b_{i_1}(o_1), \quad i = 1,2,\cdots,N$
- 递推：$\alpha_{i+1}(i) = \left[\sum_{j=1}^{N} \alpha_t(j)\alpha_{ji}\right] b_i(o_{t+1}), \quad i = 1,2,\cdots,N$
- 终止： $P(O|\lambda) = \sum_{j=1}^{N} \alpha_T(i)$

前向算法

- 因为：$\alpha_T(i) = P(o_1, o_2, \cdots, o_t, i_T = q_i|\lambda)$
- 所以：$P(O|\lambda) = \sum_{j=1}^{N} \alpha_T(i)$
- 递推：

$O(TN^2)$

前向算法

- 减少计算量的原因在于每一次计算， 直接引用前一个时刻的计算结果，避免重复计算。

图 10.2　观测序列路径结构

例子

$$A = \begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.3 & 0.5 & 0.2 \\ 0.2 & 0.3 & 0.5 \end{bmatrix}, B = \begin{bmatrix} 0.5 & 0.5 \\ 0.4 & 0.6 \\ 0.7 & 0.3 \end{bmatrix}, \pi = (0.2,0.4,0.4)^T$$

设T=3，O=（红，白，红），试用前向算法计算$P(O|\lambda)$

(1) 计算初值

$$\alpha_1(1) = \pi_1 b_1(o_1) = 0.10$$
$$\alpha_1(2) = \pi_2 b_2(o_1) = 0.16$$
$$\alpha_1(3) = \pi_3 b_3(o_1) = 0.28$$

例子

(2) 递推计算

$$\alpha_2(1) = \left[\sum_{i=1}^{3} \alpha_1(i)\alpha_{i1}\right] b_1(o_2) = 0.154\times0.5 = 0.077$$

$$\alpha_2(2) = \left[\sum_{i=1}^{3} \alpha_1(i)\alpha_{i2}\right] b_2(o_2) = 0.184\times0.6 = 0.1104$$

$$\alpha_2(3) = \left[\sum_{i=1}^{3} \alpha_1(i)\alpha_{i3}\right] b_3(o_2) = 0.202\times0.3 = 0.0606$$

$$\alpha_3(1) = \left[\sum_{i=1}^{3} \alpha_2(i)\alpha_{i1}\right] b_1(o_3) = 0.04187$$

$$\alpha_3(2) = \left[\sum_{i=1}^{3} \alpha_2(i)\alpha_{i2}\right] b_2(o_3) = 0.03551$$

$$\alpha_3(3) = \left[\sum_{i=1}^{3} \alpha_2(i)\alpha_{i3}\right] b_3(o_3) = 0.05284$$

(3) 终止

$$P(O|\lambda) = \sum_{i=1}^{3} \alpha_3(i) = 0.13022$$

后向算法

- 后向概率： 给定隐马尔科夫模型λ， 定义在时刻t状态为qi的条件下， 从t+1到T的部分观测序列为$o_{t+1}, o_{t+2}, \cdots, o_T$的概率为后向概率， 记作：

$$\beta_t(i) = P(o_{t+1}, o_{t+2}, \cdots, o_T \,|i_t = q_i, \lambda)$$

可以用递推的方法求得后向概率$\beta_t(i)$及观测序列概率$P(O|\lambda)$

后向算法

输入：隐马尔科夫模型λ，观测序列O；

输出：观测序列概率$P(O|\lambda)$

(1) $\beta_T(i) = 1, i = 1,2,\cdots,N$

(2) 对$\mathrm{t} = \mathrm{T} - 1, \mathrm{T} - 2, \cdots, 1$

$$\beta_t(i) = \sum_{j=1}^{N} \alpha_{ij} b_j(o_{t+1}) \beta_{t+1}(j), i = 1,2,\cdots,N$$

(3) $P(O|\lambda) = \sum_{j=1}^{N} \pi_i b_i(o_1) \beta_1(i)$

归纳顺序： $\beta_T(x), \beta_{T-1}(x), \cdots, \beta_1(x)$ $\longrightarrow$

$O(N^2T)$

后向算法

$$P(O|\lambda) = \sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_t(i)\alpha_{ij}b_j(o_{t+1})\beta_{t+1}(j)\,, t = 1,2,\cdots,T-1$$

Viterbi 搜索算法

- 在给定模型 $\mu=(A,B,\pi)$ 和观察序列 $O=O_1O_2\ldots O_T$ 的情况下，如何选择在一定意义下“最优” 的状态序列 $P(O|\mu)$ ，使得该状态序列 “最好地解释”观察序列 $O=O_1O_2\ldots O_T$？（预测问题）
- 状态序列中的每个状态都单独地具有概率，即：对于每个时刻 t ($1\le t\le T$)，寻找 q_t 使得 $\gamma_t(i)=P(q_t=S_i|O,\mu)$ 最大

$$\gamma_t(i)=P(q_t=S_i|O,\mu)=\frac{P(q_t=S_i,O|\mu)}{P(O|\mu)}$$

HMM 的输出序列 O，并且在时间 t 到达状态 i 的概率。

Viterbi 搜索算法

- HMM 在时间 t 到达状态 i, 并且输出 $O = O_1 O_2 \cdots O_T$ 。 根据前向变量的定义，实现这一步的概率为：$\alpha_t(i)$
- 从时间 t，状态 Si 出发，HMM 输出 $O = O_1 O_2 \cdots O_T$ ，根据后向变量定义，实现这一步的概率为 $\beta_t(i)$

$$P(q_t = S_i, O \mid \mu) = \alpha_t(i) \times \beta_t(i)$$

$$P(O \mid \mu) = \sum_{i=1}^{N} \alpha_t(i) \times \beta_t(i)$$

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{\sum_{i=1}^{N} \alpha_t(i) \times \beta_t(i)}$$

$$\hat{q}_t = \arg\max_{1 \le i \le N}(\gamma_t(i))$$

Viterbi 搜索算法

- 问题：
- 每一个状态单独最优不一定使整体的状态序列最优，可能两个最优的状态 $\hat{q}_t$ 和 $\hat{q}_{t+1}$ 之间的转移概率为0，即 $a_{\hat{q}_t \hat{q}_{t+1}} = 0$

Viterbi 搜索算法

Viterbi 搜索算法

Viterbi 搜索算法

Viterbi 搜索算法

与深度优先算法

- 在给定模型μ 和观察序列O 的条件下求概率最大的状态序列：

$$\hat{Q} = \arg\max_{Q} P(Q \mid O, \mu)$$

- Viterbi algorithm: 动态搜索最优状态序列
- 定义：Viterbi 变量 $\delta_t(i)$ 是在时间 t 时，HMM 沿着某一条路径到达 S_i，并输出观察序列 $O = O_1 O_2 \cdots O_T$ 的最大概率：

$$\delta_t(i) = \max_{q_1, q_2, \cdots, q_{t-1}} P(q_1, q_2, \cdots, q_t = S_i, O_1 O_2 \cdots O_t \mid \mu)$$

Viterbi 搜索算法

- 递归计算：

$$\delta_{t+1}(i) = \max_j [\delta_t(j) \cdot a_{ji}] \cdot b_i(O_{t+1})$$

- Viterbi 算法
- 初始化：

$$\delta_1(i) = \pi_i b_i(O_1), \quad 1 \le i \le N$$

  - 概率最大的路径变量：$\psi_1(i) = 0$
- 递推计算：

$$\delta_t(j) = \max_{1 \le i \le N} [\delta_{t-1}(i) \cdot a_{ij}] \cdot b_j(O_t), \quad 2 \le t \le T, \quad 1 \le j \le N$$

$$\psi_t(j) = \arg\max_{1 \le i \le N} [\delta_{t-1}(i) \cdot a_{ij}] \cdot b_j(O_t), \ 2 \le t \le T, \ 1 \le i \le N$$

Viterbi 搜索算法

- 结束

$$\hat{Q}_T = \arg\max_{1\le i\le N}[\delta_T(i)], \qquad \hat{P}(\hat{Q}_T) = \max_{1\le i\le N} \delta_T(i)$$

- 通过回溯得到路径

$$\hat{q}_t = \psi_{t+1}(\hat{q}_{t+1}), \quad t = T-1, T-2, \cdots, 1 \qquad O(N^2T)$$

例子

$$A=\begin{bmatrix}0.5 & 0.2 & 0.3\\ 0.3 & 0.5 & 0.2\\ 0.2 & 0.3 & 0.5\end{bmatrix}, B=\begin{bmatrix}0.5 & 0.5\\ 0.4 & 0.6\\ 0.7 & 0.3\end{bmatrix}, \pi=(0.2,0.4,0.4)^T$$

O=（红，白，红），试求最优状态序列，即最优路径$I^*=(i_1^*,i_2^*,i_3^*)$

- 初始化： 在t=1时， 对每一个状态i， i=1,2,3， 求状态i观测Q_1为红的概率， 记为：$\delta_1(i)$

$$\delta_1(i)=\pi_i b_i(o_1)=\pi_i b_i(\text{红})\text{，}i=1,2,3$$
$$\delta_1(i)=0.10,\delta_1(2)=0.16,\delta_1(3)=0.28$$
$$\psi_1(i)=0,\qquad i=1,2,3$$

例子

- 在t=2时， 对每一个状态i， i=1,2,3， 求在t=1时状态为j观测Q_1为红并在t=2时状态为i观测Q_2位白的路径的最大概率， 记为$\delta_1(i)$

$$\delta_2(i) = \max_{1 \le j \le 3} \left[\delta_1(i) a_{ji}\right] b_i(o_2)$$

- 同时， 对每个状态i， 记录概率最大路径的前一个状态j

$$\psi_2(i) = arg \max_{1 \le j \le 3} [\delta_1(i) a_{ji}] \ , \quad i = 1,2,3$$

例子

- 计算：

$$\delta_2(1) = \max_{1\le j\le 3}\left[\delta_1(i)a_{j1}\right]b_1(o_2)$$

$$= \max_j\{0.01\times 0.5, 0.16\times 0.3, 0.28\times 0.2\}\times 0.5 = 0.028$$

$$\psi_2(1) = 3$$

$$\delta_2(2) = 0.0504, \psi_2(2) = 3$$

$$\delta_2(3) = 0.042, \psi_2(3) = 3$$

同样，在t=3时 $\delta_3(3) = \max_{1\le j\le 3}\left[\delta_2(i)a_{ji}\right]b_i(o_3)$

$$\psi_3(i) = arg\max_{1\le j\le 3}[\delta_2(j)a_{ji}]$$

$$\delta_3(1) = 0.00756, \qquad \psi_3(1) = 2$$

$$\delta_3(2) = 0.01008, \qquad \psi_3(2) = 2$$

$$\delta_3(3) = 0.0147, \qquad \psi_3(3) = 3$$

例子

- 以P*表示最优路径的概率：

$$P^* = \max_{1 \le i \le 3} \delta_3(i) = 0.0147$$

- 最优路径的终点是：

$$i_3^* = arg \max_i [\delta_3(i)] = 3$$

- 由最优路径的终点i_3^* ，逆向找到i_2^*, i_1^*

在t=2时， $i_2^* = \psi_3(i_3^*) = \psi_3(3) = 3$

在t=1时， $i_1^* = \psi_2(i_2^*) = \psi_2(3) = 3$

- 于是求得最优路径，即最优状态序列：

$$I^* = (i_1^*, i_2^*, i_3^*) = (3,3,3)$$

例子

图 10.4　求最优路径

学习算法

- 监督学习方法
- Baum-Welch 算法
- Baum-Welch模型参数估计公式

学习算法

- 给定一个观察序列$O = O_1 O_2 \ldots O_T$，如何根据最大似然估计来求模型的参数值？即如何调节模型 $\mu = (A, B, \pi)$ 的参数，使得$P(O \mid \mu)$最大？（学习问题）

- 监督学习方法：
  - 假设训练数据是包括观测序列O和对应的状态序列I

$$\{(O_1, I_1), (O_2, I_2), \cdots, (O_s, I_s)\}$$

  - 可以利用极大似然估计法来估计隐马尔可夫模型参数。

- 非监督学习方法：
  - 假设训练数据只有S个长度为T的观测序{O1,O2,…Os},
  - 采用Baum-Welch算法

监督学习方法

- 转移概率a_{ij}的估计：$\{(O_1, I_1), (O_2, I_2), \cdots, (O_s, I_s)\}$
- 设样本中时刻t处于状态i， 时刻t+1转移到状态j的频数为A_{ij}， 那么状态转移概率a_{ij}的估计是：

$$\hat{a}_{ij} = \frac{A_{ij}}{\sum_{j=1}^{N} A_{ij}}, i = 1,2,\cdots,N; j = 1,2,\cdots,N$$

监督学习方法

- 观测概率bj(k)的估计：

$$\{(O_1, I_1), (O_2, I_2), \cdots, (O_s, I_s)\}$$

- 设样本中状态为j并观测为k的频数是$B_j(k)$，那么状态为j观测为k的概率

$$\hat{b}_i(k) = \frac{B_{jk}}{\sum_{k=1}^{M} B_{jk}}, \qquad j = 1,2,\cdots,N; k = 1,2,\cdots,M$$

- 初始状态概率π_i的估计$\hat{\pi}_i$为S个样本中初始状态为q_i的频率。

Baum-Welch算法

- 假定训练数据只包括{O1,O2,...Os},
- 求模型参数λ=（A,B,π)
- 实质上是有隐变量的概率模型： EM算法

$$P(O|\lambda) = \sum_{I} P(O|I,\lambda)P(I|\lambda)$$

-
- 确定完全数据的对数似然函数
- 完全数据 $(O,I) = (o_1, o_2, \cdots, o_T, i_1, i_2, \cdots, i_T)$
- 完全数据的对数似然函数 $\log P(O,I|\lambda)$

Baum-Welch算法

- 给定 HMM 模型μ和观察序列$O = O_1 O_2 \cdots O_T$，那么，在时间 t 位于状态 Si，时间 t+1 位于状态 Sj 的概率

$$\xi_t(i,j) = P(q_t = S_i, q_{t+1} = S_j \mid O, \mu) = \frac{P(q_t = S_i, q_{t+1} = S_j, O \mid \mu)}{P(O \mid \mu)}$$

$$= \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{P(O \mid \mu)}$$

$$= \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}$$

Baum-Welch算法

Baum-Welch算法

- 那么，给定模型 μ 和观察序列 $O = O_1 O_2 \cdots O_T$ ，在时间 t 位于状态 Si 的概率为：

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i, j)$$

Baum-Welch算法

$$\pi_i = \gamma_1(i)$$

$$\bar{a}_{ij} = \frac{Q\text{中从状态}q_i\text{转移到}q_j\text{的期望次数}}{Q\text{中所有从状态}q_i\text{转移到下一状态(包括}q_j\text{自身)的期望次数}}$$

$$= \frac{\sum_{i=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)}$$

$$\bar{b}_j(k) = \frac{Q\text{中从状态}q_j\text{输出符号}v_k\text{的期望次数}}{Q\text{到达}q_j\text{的期望次数}}$$

$$= \frac{\sum_{t=1}^{T} \gamma_t(j) \times \delta(O_t, v_k)}{\sum_{t=1}^{T} \gamma_t(j)}$$

Baum-Welch算法

- 初始化：随机地给 π_i， a_{ij}， $b_j(k)$ 赋值

$$\begin{cases} \sum_{i=1}^{N} \pi_i = 1 & \\ \sum_{j=1}^{N} a_{ij} = 1 & 1 \le i \le N \\ \sum_{k=1}^{M} b_i(k) = 1 & 1 \le i \le N \end{cases}$$

- 由此得到模型μ_0，令 i = 0。

Baum-Welch算法

- EM 算法：
- E-步：计算期望值$\xi_t(i, j)$、$\gamma_t(i)$ 。
- M-步：用E-步中所得到的期望值，重新估计π_i, a_{ij}, $b_j(k)$得到模型μ_{i+1} 。
- 循环：i = i+1，重复执行 E-步和M-步，直至π_i, a_{ij}, $b_j(k)$的值收敛：

$$|\log P(O|\mu_{i+1}) - \log P(O|\mu_i)| < \varepsilon$$

EM算法

三硬币模型

- 三硬币模型： 硬币A、 B、 C， 正面概率π， p， q，
- A正面时选B， 反面选C，然后再扔
- 得到结果： 1101001011
- 问题： 只能看结果， 不能看中间过程， 估算π， p， q，

$$P(y|\theta) = \sum_z P(y,z|\theta) = \sum_z P(z|\theta)\,P(y|z,\theta)$$
$$= \pi p^y (1-p)^{1-y} + (1-\pi) q^y (1-q)^{1-y}$$

- 随机变量Y是观测变量， 表示一次试验观测的结果是1或0， 随机变量z是隐变量， 表示未观测到的掷硬币A的结果， 这一模型是以上数据的生成模型。 现在需要知道三个硬币分别正面出现的概率

三硬币模型

- 观测数据：$Y = (Y_1, Y_2, \cdots, Y_n)^T$
- 未观测数据：$Z = (Z_1, Z_2, \cdots, Z_n)^T$
- 似然函数：$P(Y|\theta) = \sum_Z P(Z|\theta)\, P(Y|Z, \theta)$
- 即：

$$P(Y|\theta) = \prod_{i=1}^{n} [\pi p^{y_j} (1-p)^{1-y_j} + (1-\pi) q^{y_j} (1-q)^{1-y_j}]$$

- 极大似然估计：

$$\hat{\theta} = arg \max_{\theta} \log P(Y|\theta)$$

EM算法

- 选取初值：$\theta^{(0)} = (\pi^{(0)}, p^{(0)}, q^{(0)})$
- 第i步的估计值：$\theta^{(i)} = (\pi^{(i)}, p^{(i)}, q^{(i)})$
- EM算法第i+1次迭代：
- E步： 计算在模型参数$\pi^{(i)}, p^{(i)}, q^{(i)}$下观测数据yi来自掷硬币B的概率：

$$\mu^{(i+1)} = \frac{\pi^{(i)}(p^{(i)})^{y_j}(1-p)^{1-y_j}}{\pi^{(i)}(p^{(i)})^{y_j}(1-p)^{1-y_j} + (1-\pi^{(i)})(q^{(i)})^{y_j}(1-q)^{1-y_j}}$$

- M步: 计算模型参数的新估计值

$$\pi^{(i+1)} = \frac{1}{n}\sum_{j=1}^{n}\mu_j^{(i+1)} \quad p^{(i+1)} = \frac{\sum_{j=1}^{n}\mu_j^{(i+1)}y_j}{\sum_{j=1}^{n}\mu_j^{(i+1)}} \quad q^{(i+1)} = \frac{\sum_{j=1}^{n}(1-\mu_j^{(i+1)})y_j}{\sum_{j=1}^{n}(1-\mu_j^{(i+1)})}$$

EM算法

- 初值：

$$\pi^{(0)} = 0.5, p^{(0)} = 0.5, q^{(0)} = 0.5$$

1101001011

$$\text{对} y_j = 1 \text{与} y_j = 0 \text{均有} \mu_j^{(1)} = 0.5$$

- 利用迭代公式，得：$\pi^{(1)} = 0.5, p^{(1)} = 0.6, q^{(1)} = 0.6$

$$\mu_j^{(2)} = 0.5, \qquad j = 1,2,\cdots,10$$

- 继续迭代，得

$$\pi^{(2)} = 0.5, p^{(2)} = 0.6, q^{(2)} = 0.6$$

- 得到模型参数的极大似然估计：

$$\hat{\pi} = 0.5, \hat{p} = 0.6, \hat{q} = 0.6$$

EM算法

- 输入： 观测变量数据Y,隐变量数据Z， 联合分布P(Y,Z|Θ)
  条件分布P(Z|Y, Θ)
  输出： 模型参数Θ
- (1) 选择参数的初值$\theta^{(0)}$,开始迭代；
- (2) E步：记$\theta^{(i)}$为第i次迭代参数θ的估计值，
  在第i+1次迭代的E步，计算

$$Q\left(\theta,\theta^{(i)}\right)=E_Z\left[\log P(Y,Z|\theta)\,\middle|Y,\theta^{(i)}\right]$$

当前已知的参数条件下，
Y和Z的联合分布的期望

$$=\sum_Z \log P(Y,Z|\theta)\,P(Z|Y,\theta^{(i)})$$

- 给定观测数据Y和当前参数估计Θ
- Q函数定义：完全数据的对数似然函数$\log P(Y,Z|\theta)$关于在给定观测数据Y和当前函数$\theta^{(i)}$下对未观测数据Z的条件概率分布$P(Z|Y,\theta^{(i)})$的期望称为Q函数

EM算法

- (3) M步：求使$Q(\theta,\theta^{(i)})$极大化的θ，
  确定第i+1次迭代的参数的估计值$\theta^{(i+1)}$

$$\theta^{(i+1)} = arg\max_{\theta} Q(\theta,\theta^{(i)})$$

- 停止迭代的条件

$$||\theta^{(i+1)} - \theta^{(i)}|| < \varepsilon_1 \text{ 或 } ||Q(\theta^{(i+1)} - \theta^{(i)}) - Q(\theta^{(i)} - \theta^{(i)})|| < \varepsilon_2$$

EM算法近似实现对观测数据的极大似然估计

- 极大化(不完全数据)Y关于参数Θ的极大似然函数：

$$L(\theta) = \log P(Y|\theta) = \log \sum_{Z} P(Y, Z|\theta)$$

$$= \log(\sum_{Z} P(Y|Z, \theta)\, P(Z|\theta))$$

- EM通过迭代逐步近似极大化L(Θ)
-

$$L(\theta) > L\left(\theta^{(i)}\right)$$

EM算法近似实现对观测数据的极大似然估计

- 考虑二者的差：

$$L(\theta)-L\left(\theta^{(i)}\right)=\log\left(\sum_{Z}P(Y|Z,\theta)\,P(Z|\theta)\right)-\log P\left(Y\middle|\theta^{(i)}\right)$$

- Jason不等式：

$$L(\theta)-L\left(\theta^{(i)}\right)=log\left(\sum_{Z}P\left(Y\middle|Z,\theta^{(i)}\right)\frac{P(Y|Z,\theta)P(Z|\theta)}{P\left(Y\middle|Z,\theta^{(i)}\right)}\right)-\log P\left(Y\middle|\theta^{(i)}\right)$$

$$\geq\sum_{Z}P\left(Y\middle|Z,\theta^{(i)}\right)log\frac{P(Y|Z,\theta)P(Z|\theta)}{P\left(Y\middle|Z,\theta^{(i)}\right)}-\log P\left(Y\middle|\theta^{(i)}\right)$$

$$=\sum_{Z}P\left(Y\middle|Z,\theta^{(i)}\right)log\frac{P(Y|Z,\theta)P(Z|\theta)}{P\left(Y\middle|Z,\theta^{(i)}\right)P\left(Y\middle|\theta^{(i)}\right)}$$

EM算法近似实现对观测数据的极大似然估计

- 令：

$$B\left(\theta,\theta^{(i)}\right)=L\left(\theta^{(i)}\right)+\sum_{Z}P\left(Z\middle|Y,\theta^{(i)}\right)\log\frac{P(Y|Z,\theta)P(Z|\theta)}{P\left(Z\middle|Y,\theta^{(i)}\right)P\left(Y\middle|\theta^{(i)}\right)}$$

- 则：

$$L(\theta)\geq B\left(\theta,\theta^{(i)}\right)$$
$$L\left(\theta^{(i)}\right)=B\left(\theta^{(i)},\theta^{(i)}\right)$$

任何可以使$B\left(\theta^{(i)},\theta^{(i)}\right)$增大的$\theta$，也可以使$L(\theta)$增大

$$\theta^{(i+1)}=arg\max_{\theta}B\left(\theta,\theta^{(i)}\right)$$

EM算法近似实现对观测数据的极大似然估计

- 省去和Θ无关的项：

$$\theta^{(i+1)} = arg\max_{\theta}(L(\theta^{(i)}) + \sum_{Z} P(Z|Y,\theta^{(i)})\log\frac{P(Y|Z,\theta)P(Z|\theta)}{P(Z|Y,\theta^{(i)})P(Y|\theta^{(i)})})$$

$$= arg\max_{\theta}(\sum_{Z} P(Z|Y,\theta^{(i)})\log(P(Y|Z,\theta)P(Z|\theta)))$$

$$= arg\max_{\theta}(\sum_{Z} P(Z|Y,\theta^{(i)})\log P(Y|Z,\theta))$$

$$= arg\max_{\theta} Q(\theta,\theta^{(i)})$$

条件随机场（CRF）

概率无向图模型

- 概率无向图模型(Probabilistic Undirected Graphical Model) ，又称马尔可夫随机场(Markov Random Field)，是一个可以由无向图表示的概率联合分布

模型定义

- 概率无向图模型：
  - 设有联合概率分布P(Y),由无向图G=(V,E)表示，在图G中，结点表示随机变量，边表示随机变量之间的依赖关系,
  - 如果联合概率分布P(Y)满足成对、局部或全局马尔可夫性，就称此联合概率分布为概率无向图模型(probability undirected graphical model)，或马尔可夫随机场(Markov random field).
  - 问题关键：求联合概率，为对联合概率进行因子分解。

概率无向图模型的因子分解

- 无向图G中任何两个结点均有边连接的结点子集称为团(clique)。
- 若C是无向图G的一个团，并且不能再加进任何一个c的结点使其成为一个更大的团，则称此C为最大团(maximal clique).

概率无向图模型的因子分解

- 将概率无向图模型的联合概率分布表示为其最大团上的随机变量的函数的乘积形式的操作，称为概率无向图模型的因子分解(Factorization).
- 给定概率无向图模型，设其无向图为G，C为G上的最大团，Yc表示C对应的随机变量，那么概率无向图模型的联合概率分布P(Y)可写作图中所有最大团C上的函数$\Psi_c(Y_c)$的乘积形式，即

$$P(Y)=\frac{1}{Z}\prod_c \Psi_c(Y_c)$$

- Z是归一化因子(normalization factor)

$$Z=\sum_Y \prod_c \Psi_c(Y_c)$$

$$\Psi_c(Y_c)=\exp\{-E(Yc)\}$$

势函数

条件随机场

- 设X与Y是随机变量，P(Y|X)是在给定X的条件下Y的条件概率分布，若随机变量Y构成一个由无向图G=(V,E)表示的马尔可夫随机场，即满足马尔科夫性

$$P\left(Y_v \middle| X, Y_w, w \neq v\right) = P\left(Y_v \middle| X, Y_W, w \sim v\right)$$

- 对任意结点v成立，则称条件概率分布P(Y|X)为条件随机场，式中w~v表示在图G=(V,E)中与结点v有边连接的所有结点w，w≠v表示结点v以外的所有结点。

条件随机场

- 线性链条件随机场(linear chain conditional random field)
  - 在条件概率模型P(Y|X)中，Y是输出变量，表示标记序列，X是输入变量，表示需要标注的观测序列，也把标记序列称为状态序列。

条件随机场的定义与形式

- 定义(线性链条件随机场)
  - 设 $X=(X_1,X_2,\cdots,X_n)$, $Y=(Y_1,Y_2,\cdots,Y_n)$ 均为线性链表示的随机变量序列，若在给定随机变量序列X的条件下，随机变量序列Y的条件概率分布P(Y|X)构成条件随机场。即满足马尔可夫性

$$P(Y_i|X,Y_1,\cdots,Y_{i-1},Y_{i+1},Y_n)=P(Y_i|X,Y_{i-1},Y_{i+1})$$

$$i=1,2,\cdots,n$$ （在i=1和n时只考虑单边）

  - 则称P(Y |X)为线性链条件随机场。
  - 在标注问题中，X表示输入观测序列，Y表示对应的输出标记序列或状态序列.

条件随机场

- 条件随机场(conditional random field)三个主要问题：
  - 概率计算
  - 模型学习
  - 推测状态

条件随机场的参数化形式

设P(Y|X)为线性链条件随机场，则在随机变量X取值为x的条件下，随机变量Y取值为y的条件概率具有如下形式：

$$P(y|x)=\frac{1}{Z(x)}\exp\left(\sum_{i,k}\lambda_k t_k(y_{i-1},y_i,x,i)+\sum_{i,l}\mu_l s_l(y_i,x,i)\right)$$

其中：

$$Z(x)=\sum_{y}\exp\left(\sum_{i,k}\lambda_k t_k(y_{i-1},y_i,x,i)+\sum_{i,l}\mu_l s_l(y_i,x,i)\right)$$

- t_k定义在边上的特征函数，转移特征，依赖于前一个和当前位置，
- s_l定义在结点上的特征函数，状态特征，依赖于当前位置。

例子

- 输入观测为：X=(X1,X2,X3)，输出标记为Y=(Y1,Y2,Y3)，Y1,Y2,Y3 取值于{1,2}
- 假设特征和对应权值，只注明特征取值为1，为0省略

$$t_1 = t_1(y_{i-1} = 1, y_i = 2, x, i),\ i = 2,3,\ \lambda_1 = 1$$

$$t_1(y_{i-1}, y_i, x, i) = \begin{cases} 1, & y_{i-1} = 1, y_i = 2, x, i, (i = 2,3) \\ 0, & \text{其他} \end{cases}$$

$$\begin{array}{ll} t_2 = t_2(y_1 = 1, y_2 = 1, x, 2) & \lambda_2 = 0.5 \\ t_3 = t_3(y_2 = 2, y_3 = 1, x, 3) & \lambda_3 = 1 \\ t_4 = t_4(y_1 = 2, y_2 = 1, x, 2) & \lambda_4 = 1 \\ t_5 = t_5(y_2 = 2, y_3 = 2, x, 3) & \lambda_5 = 0.2 \end{array}$$

$$\begin{array}{lll} s_1 = s_1(y_1 = 1, x, 1), & & \mu_1 = 1 \\ s_2 = s_2(y_1 = 2, x, i), & i = 1,2 & \mu_2 = 0.5 \\ s_3 = s_3(y_i = 1, x, i), & i = 2,3 & \mu_3 = 0.8 \\ s_4 = s_4(y_3 = 2, x, 3), & & \mu_4 = 0.5 \end{array}$$

条件随机场的参数化形式

$$P(y|x) \propto \exp\left[\sum_{k=1}^{5}\lambda_k \sum_{i=2}^{3} t_k(y_{i-1}, y_i, x, i) + \sum_{k=1}^{4}\mu_k \sum_{i=2}^{3} s_k(y_i, x, i)\right]$$

- 对给定的观测序列x，标记序列Y=(1, 2, 2)

$$P(y1=1, y2=2, y3=2|x) \propto \exp(3.2\ )$$

$$t_1 = t_1(y_{i-1}=1, y_i=2, x, i), \quad i=2,3, \quad \lambda_1 = 1$$

条件随机场的简化形式

$$f_k(y_{i-1}, y_i, x, i) = \begin{cases} t_k(y_{i-1}, y_i, x, i), & k = 1,2,\cdots,K_1 \\ s_l(y_i, x, i), \quad k = K_1 + 1; & l = 1,2,\cdots,K_2 \end{cases}$$

$$f_k(y, x) = \sum_{i=1}^{n} f_k(y_{i-1}, y_i, x, i), \quad k = 1,2,\cdots,K \qquad w_k = \begin{cases} \lambda_k, & k = 1,2,\cdots,K_1 \\ \mu_l, \quad k = K_1 + l; & l = 1,2,\cdots,K_2 \end{cases}$$

$$P(y|x) = \frac{1}{Z(x)} \exp \sum_{k=1}^{K} w_k f_k(y, x)$$

$$Z(x) = \sum_{y} \exp \sum_{k=1}^{K} w_k f_k(y, x)$$

条件随机场的简化形式

- 若w表示权值向量：

$$w=(w_1,w_2,\cdots,w_K)^T$$

- 以F(y,x)表示全局特征向量，即

$$F(y,x)=(f_1(y,x),f_1(y,x),\cdots,f_K(y,x))^T$$

- 条件随机场写成内积：

$$P_w(y|x)=\frac{\exp(w\bullet F(y,x))}{Z_w(x)}$$

$$Z_w(x)=\sum_y \exp(w\bullet F(y,x))$$

条件随机场的矩阵形式

- 线性链条件随机场，引进特殊的起点和终点状态标记Y_0=start, Y_n+1=stop，这时$P_w(y|x)$可以通过矩阵形式表示。
- 对观测序列x的每一个位置i=1,2,..,n+1，定义一个m阶矩阵(m是标记Y_i取值的个数)

$$M_i(x)=\left[M_i\left(y_{i-1},y_i|x\right)\right]$$

$$M_i\left(y_{i-1},y_i|x\right)=\exp\left(W_i\left(y_{i-1},y_i|x\right)\right)$$

$$W_i\left(y_{i-1},y_i|x\right)=\sum_{i=1}^{K}w_k f_k\left(y_{i-1},y_i,x,i\right)$$

条件随机场的矩阵形式

- 条件概率$P_w(y|x)$： $$P_w(y|x)=\frac{1}{Z_w(x)}\prod_{i=1}^{n+1}M_i(y_{i-1},y_i|x)$$

- $Z_w(x)$为规范化因子，是n+1个矩阵的乘积的(start, stop)元素

$$Z_w(x)=\left(M_1(x)M_2(x)\cdots M_{n+1}(x)\right)_{start,stop}$$

条件随机场的概率计算问题

- 条件随机场的概率计算问题
  - 给定条件随机场P(Y|X),输入序列x和输出序列y，
  - 计算条件概率：

$$P(Y_i|X), P(Y_{i-1}=y_{i-1}, Y_i=y_i|x)$$

- 引进前向-后向向量，递归计算（前向后向算法）

条件随机场的概率计算问题

- 前向-后向算法：
  - 对每个指标i=0,1,…,n+1，定义前向向量$\alpha_i(x)$

$$\alpha_0(y|x)=\begin{cases}1, & y=start\\ 0, & 否则\end{cases}$$

  - 递推公式：

$$\alpha_i^T(y_i|x)=\alpha_{i-1}^T(y_{i-1}|x)M_i(y_{i-1},y_i|x),\quad i=1,2,\cdots,n+1$$

  - 又可表示为：

$$\alpha_i^T(x)=\alpha_{i-1}^T(x)M_i(x)$$

  - 即表示在位置i的标记是y_i，且到位置i的前部分标记序列的非规范化概率， y_i可取的值m个，所以 $\alpha_i(x)$ 是m维列向量。

条件随机场的概率计算问题

- 前向-后向算法：
  - 同样，对每个指标i=0,1,…,n+1，定义前向向量 $\beta_i(x)$

$$\beta_{n+1}(y_{n+1}|x)=\begin{cases}1, & y_{n+1}=stop\\ 0, & \text{否则}\end{cases}$$

$$\beta_i(y_i|x)=M_i(y_i,y_{i+1}|x)\beta_{i-1}(y_{i+1}|x)$$

  - 又可表示为：

$$\beta_i(x)=M_{i+1}(x)\beta_{i+1}(x)$$

  - 即表示在位置i的标记是y_i，且从位置i+1到n的后部分标记序列的非规范化概率

$$Z(x)=\alpha_n^T(x)\bullet 1=1^T\bullet\beta_1(x)$$

条件随机场的概率计算问题

- 可计算标记序列在位置i是标记y_i的条件概率
- 和在位置i-1与i是标记y_{i-1}和y_i的条件概率：

$$P(Y_i = y_i|x) = \frac{\alpha_i^T(y_i|x)\beta_i(y_i|x)}{Z(x)}$$

$$P(Y_{i-1} = y_{i-1}, Y_i = y_i|x) = \frac{\alpha_{i-1}^T(y_{i-1}|x)M_i(y_{i-1}, y_i|x)\beta_i(y_i|x)}{Z(x)}$$

$$Z(x) = \alpha_n^T(x) \cdot \mathbf{1}$$

条件随机场的概率计算问题

- 利用前向-后向向量， 得到特征函数关于联合分布P(X,Y)和条件分布P(Y|X)的数学期望。
- 特征函数f_k关于条件分布P(Y|X)的数学期望是：

$$E_{P(Y|X)}[f_k] = \sum_y P(y|x) f_k(y,x)$$

$$= \sum_{i=1}^{n+1} \sum_{y_{i-1} y_i} f_k(y_{i-1}, y_i, x, i) \frac{\alpha_{i-1}^T(y_{i-1}|x) M_i(y_{i-1}, y_i|x) \beta_i(y_i|x)}{Z(x)}$$

$$k = 1,2,\cdots,K$$

$$Z(x) = \alpha_n^T(x) \cdot \mathbf{1}$$

条件随机场的概率计算问题

- 假设经验分布为：$\tilde{P}(X)$ 特征函数fk关于联合分布P(X,Y)的数学期望是：

$$E_{P(Y|X)}[f_k]$$

$$= \sum_{x,y} P(x,y) f_k(y,x) \sum_{i=1}^{n+1} f_k(y_{i-1}, y_i, x, i)$$

$$= \sum_{x} \tilde{P}(x) \sum_{y} P(y|x) \sum_{i=1}^{n+1} f_k(y_{i-1}, y_i, x, i)$$

$$= \sum_{x} \tilde{P}(x) \sum_{i=1}^{n+1} \sum_{y_{i-1} y_i} f_k(y_{i-1}, y_i, x, i) \frac{\alpha_{i-1}^T(y_{i-1}|x) M_i(y_{i-1}, y_i|x) \beta_i(y_i|x)}{Z(x)}$$

$$Z(x) = \alpha_n^T(x) \cdot \mathbf{1}$$

条件随机场的学习算法

- 拟牛顿法：

$$P_w(y|x) = \frac{\exp(\sum_{i=1}^{n} w_i f_i(x,y))}{\sum_y \exp(\sum_{i=1}^{n} w_i f_i(x,y))}$$

- 学习的优化目标函数：

$$\min_{w \in R^n} f(w) = \sum_x \tilde{P}(x) \log \sum_y \exp(\sum_{i=1}^{n} w_i f_i(x,y)) - \sum_{x,y} \tilde{P}(x,y) \sum_{i=1}^{n} w_i f_i(x,y)$$

- 梯度函数：

$$g(w) = \sum_{x,y} \tilde{P}(x) P_w(y|x) f(x,y) - E_{\tilde{P}}[f]$$

$$g(w) = E_P[f] - E_{\tilde{P}}[f]$$

解码算法

- 给定条件随机场P(Y|X)和输入序列(观测序列)x，
- 求：条件概率最大的输出序列(标记序列)y*,
- 维特比算法：

$$P_w(y|x)=\frac{\exp(w\bullet F(y,x))}{Z_w(x)} \Longrightarrow y^*=\arg\max_y P_w(y|x)$$

$$=\arg\max_y \frac{\exp(w\bullet F(y,x))}{Z_w(x)}$$

$$=\arg\max_y \exp(w\bullet F(y,x))$$

$$=\arg\max_y (w\bullet F(y,x))$$

解码算法

- Viterbi 搜索算法

条件随机场的预测算法

- 路径表示标记序列：

$$w=(w_1,w_2,\cdots,w_K)^T$$

$$F(y,x)=(f_1(y,x),f_2(y,x),\cdots,f_K(y,x))^T$$

$$f_k(y,x)=\sum_{i=1}^{n}f_k(y_{i-1},y_i,x,i),\quad k=1,2,\cdots,K$$

$$\max_y\sum_{i=1}^{n}w\bullet F_i(y_{i-1},y_i,x)$$

$$F_i(y_{i-1},y_i,x)=(f_1(y_{i-1},y_i,x,i),f_2(y_{i-1},y_i,x,i),\cdots,F_K(y_{i-1},y_i,x,i))^T$$

条件随机场的预测算法

- 维特比算法：
- 首先求出位置1的各个标记j=1,2..m的非规范化概率：

$$\delta_i(j) = w \bullet F_i(y_0 = start, y_1 = j, x), \quad j = 1,2,\cdots,m$$

- 递归计算：

$$\delta_i(l) = \max_{1 \le f \le m} \{\delta_{i-1}(j) + w \bullet F_i(y_{i-1} = j, y_i = l, x)\}, \quad l = 1,2,\cdots,m$$

$$\Psi_i(l) = \arg\max_{1 \le f \le m} \{\delta_{i-1}(j) + w \bullet F_i(y_{i-1} = j, y_i = l, x)\}, \quad l = 1,2,\cdots,m$$

条件随机场的预测算法

- 直到i=n时终止，这时求得非规范化概率的最大值为：

$$\max_{y}\left(w\bullet F\left(y,x\right)\right)=\max_{1\le j\le m}\delta_n\left(j\right)$$

- 及最优路径的终点：

$$y_n^*=\arg\max_{1\le j\le m}\delta_n\left(j\right)$$

- 回溯路径：

$$y_i^*=\Psi_{i+1}\left(y_{i+1}^*\right),\quad i=n-1,n-2,\cdots,1$$

$$y^*=\left(y_1^*,y_2^*,\cdots,y_n^*\right)^T$$

Topic Model

Topic Model

Seeking Life's Bare (Genetic) Necessities

COLD SPRING HARBOR, NEW YORK—How many genes does an organism need to survive? Last week at the genome meeting here,* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.

Although the numbers don't match precisely, those predictions "are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing an

Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.

* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

文档的主题分布

Seeking Life's Bare (Genetic) Necessities

COLD SPRING HARBOR, NEW YORK—How many genes does an organism need to survive? Last week at the genome meeting here,* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.

Although the numbers don't match precisely, those predictions "are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing an

* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.

SCIENCE • VOL. 272 • 24 MAY 1996

词的主题分布

human	evolution	disease	computer
genome	evolutionary	host	models
dna	species	bacteria	information
genetic	organisms	diseases	data
genes	life	resistance	computers
sequence	origin	bacterial	system
gene	biology	new	network
molecular	groups	strains	systems
sequencing	phylogenetic	control	model
map	living	infectious	parallel
information	diversity	malaria	methods
genetics	group	parasite	networks
mapping	new	parasites	software
project	two	united	new
sequences	common	tuberculosis	simulations

实际上可以理解成

- 每一个Topic是一个类别
- 多个Topic张成了一个语义空间
- 每一篇文档可以用Topic进行表示，每一个分量的大小可以认为是该文档在该主题上的分量
- 每一词也可以用Topic进行表示，词表示中的分量可以认为是词的语义分布

Latent Semantic Analysis (LSA)

- 统计词与文档的共现矩阵

text

	d1	d2	d3
word w1	1	1	3
w2	2	2	1
w3	4	2	1
w4		3	

SVD分解

Probabilistic Latent Semantic Analysis (PLSA)

n(d,w): text-d 与word-w共现的次数

$$\mathcal{L} = \sum_{d \in \mathcal{D}} \sum_{w \in \mathcal{W}} n(d,w) \log P(d,w)$$

P(d,w): d与w的联合分布

$$P(d,w) = P(d)P(w|d) \qquad P(w|d) = \sum_{z \in \mathcal{Z}} P(w|z)P(z|d)$$

$$P(d,w) = \sum_{z \in \mathcal{Z}} P(z)P(w|z)P(d|z)$$

E-Step

$$P(z|d,w) = \frac{P(z)P(d|z)P(w|z)}{\sum_{z' \in \mathcal{Z}} P(z')P(d|z')P(w|z')}$$

M-Step

$$P(w|z) \propto \sum_{d \in \mathcal{D}} n(d,w)P(z|d,w),$$

$$P(d|z) \propto \sum_{w \in \mathcal{W}} n(d,w)P(z|d,w),$$

$$P(z) \propto \sum_{d \in \mathcal{D}} \sum_{w \in \mathcal{W}} n(d,w)P(z|d,w)$$

Probabilistic Latent Semantic Analysis (PLSA)

human	evolution	disease	computer
genome	evolutionary	host	models
dna	species	bacteria	information
genetic	organisms	diseases	data
genes	life	resistance	computers
sequence	origin	bacterial	system
gene	biology	new	network
molecular	groups	strains	systems
sequencing	phylogenetic	control	model
map	living	infectious	parallel
information	diversity	malaria	methods
genetics	group	parasite	networks
mapping	new	parasites	software
project	two	united	new
sequences	common	tuberculosis	simulations

Latent Dirichlet Allocation (LDA)

Dirichlet Distribution

- Probability of the probability
- A story
  - 掷骰子，按照常理，每面出现的概率为1/6
  - 当某次事件观测到“6”时

		Probability of each face under each hypothesis about how the die is loaded						
	Belief	Face	1	2	3	4	5	6
A	.5	Probability	1/7	1/7	1/7	1/7	1/7	2/7
B	.25	Probability	1/8	1/8	1/8	1/8	1/8	3/8
C	.25	Probability	1/6	1/6	1/6	1/6	1/6	1/6

P(A) = 0.5

Dirichlet Distribution (Cont’)

- 置信度的值设为X
- X最自然的分布是 Dirichlet Distribution

$$\mathcal{X} \sim Dir(\alpha)$$

- α为先验分布

$$\mathrm{Dir}(\boldsymbol{p};\boldsymbol{\alpha}) = \frac{1}{\mathrm{B}(\boldsymbol{\alpha})} \prod_{v=1}^{|\boldsymbol{\alpha}|} p_t^{\alpha_t - 1} \quad ,$$

$$\mathrm{B}(\boldsymbol{\alpha}) = \frac{\prod_{i=1}^{|\boldsymbol{\alpha}|} \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^{|\boldsymbol{\alpha}|} \alpha_i)}$$

$$\Gamma(n) = (n-1)!$$

共轭分布 Conjugate Distribution

- What is conjugate distribution

$$p(\vartheta|x) \propto p(x|\vartheta)p(\vartheta)$$

Same function format
(different parameters)

$p(\vartheta)$ is conjugate to $p(x|\vartheta)$

例子

- 假设文本中的词之间是条件独立的

$$p(X|p) = \prod_{n=1}^{N} p(x_n|p) = \prod_{i=1}^{V} p_i^{n(i)}$$

- 参数符合Dirichlet Distribution

$$p(p|\alpha) = \frac{1}{B(\alpha)} \prod_{i=1}^{V} p_i^{\alpha_i - 1}$$

- Posterior is:

$$p(p|X) = \frac{p(X|p)p(p|\alpha)}{\int p(X|p)p(p|\alpha)dp}$$

$$= \frac{\frac{1}{B(\alpha)} \prod_{i=1}^{V} p_i^{n(i)+\alpha_i-1}}{\frac{1}{B(\alpha)} \int \prod_{i=1}^{V} p_i^{n(i)+\alpha_i-1} dp}$$

$$= \frac{1}{B(n(i)+\alpha)} \prod_{i=1}^{V} p_i^{n(i)+\alpha_i-1}$$

$$\int p(p|\alpha)dp = 1$$

$$\frac{1}{B(\alpha)} \int \prod_{i=1}^{V} p_i^{\alpha_i-1} dp = 1$$

$$\int \prod_{i=1}^{V} p_i^{\alpha_i-1} dp = B(\alpha)$$

Dirichlet Distribution

文本建模

- 文本生成的过程：上帝扔骰子
  - 我们只观测到了结果（看到了组成文档的词）
  - 我们想知道词生成的过程
    - 上帝有什么样的骰子
    - 上帝是怎么样扔骰子的

文档建模

(a) unigram

(b) mixture of unigrams

(c) pLSI/aspect model

$$p(D) = \prod_{m-1}^{M} \prod_{n=1}^{N_m} p(w_n)$$

$$p(D) = \prod_{m-1}^{M} \sum_{k-1}^{K} p(z) \prod_{n=1}^{N_m} p(w_n|z)$$

$$p(D,W) = \prod_{m-1}^{M} \prod_{n=1}^{N_m} p(d) \sum_{z} p(w_n|z)p(z|d)$$

LDA

Doc
Topic
Word
$\vec{\vartheta}_m \sim \mathrm{Dir}(\vec{\alpha})$
z
$\vec{\vartheta}_m$
$\vec{\varphi}_k$
$\vec{\varphi}_k \sim \mathrm{Dir}(\vec{\beta})$
1
2
3
4
K
$\vec{\varphi}_k \quad k \in [1, K]$

LDA

ϑ is a $k \times M$ matrix
$\vartheta_{i,j} = p(z = j | d = i)$

$$p(\theta \mid \alpha) = \frac{\Gamma\left(\sum_{i=1}^{k} \alpha_i\right)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} \theta_1^{\alpha_1 - 1} \cdots \theta_k^{\alpha_k - 1},$$

φ is a $k \times V$ matrix
$\varphi_{i,j} = p(w = j | z = i)$

- For all topics
  - Sample mixture components $\varphi_k \sim Dir(\beta)$
- For all documents
  - Sample mixture components $\vartheta_m \sim Dir(\alpha)$
  - Sample document length $N_m \sim Poiss(\varepsilon)$
  - For all words in document m
    - Sample topic index $z_{m,n} \sim Mult(\vartheta_m)$
    - Sample term for word $w_{m,n} \sim Mult(\varphi_{z_{m,n}})$

LDA

Topics
gene 0.04
dna 0.02
genetic 0.01
...
life 0.02
evolve 0.01
organism 0.01
...
brain 0.04
neuron 0.02
nerve 0.01
...
data 0.02
number 0.02
computer 0.01
...
Documents
Topic proportions and assignments
Seeking Life's Bare (Genetic) Necessities
COLD SPRING HARBOR, NEW YORK—How many genes does an organism need to survive? Last week at the genome meeting here,* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.
Although the numbers don't match precisely, those predictions
"are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing an
Haemophilus genome 1703 genes
Genes in common 233 genes
Mycoplasma genome 469 genes
Genes needed for biochemical pathways +22 genes
Redundant and parasite-specific genes removed – 4 genes
Redundant and modern genes removed –122 genes
258 genes
Minimal gene set 250 genes
128 genes
Ancestral gene set
ADAPTED FROM NCBI
* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.
Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.
SCIENCE • VOL. 272 • 24 MAY 1996

Inference

- Task: Infer the optimal latent parameter

$$p(\vartheta, z, \varphi | w, \alpha, \beta) = \frac{p(\vartheta, z, \varphi, w | \alpha, \beta)}{p(w | \alpha, \beta)}$$

word plate

$$p(\vartheta, z, \varphi, w | \alpha, \beta) = \prod_{k=1}^{K} p(\varphi_k | \beta) \prod_{m=1}^{M} p(\vartheta_m | \alpha) \prod_{n=1}^{N_m} p(w_{m,n} | \varphi_k, z_{m,n}) p(z_{m,n} | \vartheta_m)$$

Topic plate

document plate

$$p(w | \alpha, \beta) = \iiint p(\vartheta, z, \varphi, w | \alpha, \beta) d\varphi d\vartheta dz = \iint p(\vartheta_m | \alpha) p(\Phi | \beta) \prod_{n=1}^{N_m} \sum_{z} p(w_{m,n} | \varphi_k, z_{m,n}) p(z_{m,n} | \vartheta_m) \, d\vartheta d\Phi$$

$$p(D | \alpha, \beta) = \prod_{m=1}^{M} \iiint p(\vartheta, z, \varphi, w | \alpha, \beta) d\varphi d\vartheta dz = \prod_{m=1}^{M} \iint p(\vartheta_m | \alpha) p(\Phi | \beta) \prod_{n=1}^{N_m} \sum_{z} p(w_{m,n} | \varphi_k, z_{m,n}) p(z_{m,n} | \vartheta_m) \, d\vartheta d\Phi$$

Inference (Cont’)

- How to estimate the value of parameter
  - Exact inference is intractable
    - BP
  - Approximate inference
    - Gibbs Sampling
    - Variational Inference
    - Laplace Approximation
    - …

Gibbs Sampling

$$\vartheta_{m,k} = \frac{n_m^k + \alpha_k}{\sum_{k=1}^{K}(n_m^k + \alpha_k)} \qquad \varphi_{k,v} = \frac{n_k^v + \beta_v}{\sum_{v=1}^{V}(n_k^v + \beta_v)}$$

?

- If we know the exact z for each document, it's trivial to estimate ϑ and Φ

$$p(z_i = k | z_{-i}, w),\ z_{-i} \text{ is all } z_j \ and\ j \neq \text{i}$$

Gibbs Sampling (Cont')

$$p(z_i = k|Z_{-i}, W_{-i}, \alpha, \beta) = \frac{p(z_i = k, Z_{-i}, W|\alpha, \beta)}{p(z_{-i}, W|\alpha, \beta)} = \frac{p(Z, W|\alpha, \beta)}{p(Z_{-i}, W_{-i}|\alpha, \beta)} where\ z_i = k$$

$$p(Z, W|\alpha, \beta) = p(W|Z, \beta)p(Z|\alpha) = \prod_{k=1}^{K} \frac{B(n_k + \beta)}{B(\beta)} \prod_{m=1}^{M} \frac{B(n_m + \alpha)}{B(\alpha)}$$

$$p(W|Z, \beta) = \int p(W|Z, \Phi)p(\Phi|\beta)\, d\Phi = \int \prod_{k=1}^{K} \frac{1}{B(\beta)} \prod_{v=1}^{V} \varphi_{k,v}^{n_k^v + \beta_v - 1}\, d\Phi = \prod_{k=1}^{K} \frac{B(n_k + \beta)}{B(\beta)}$$

Multinomial Distribution: $p(W|Z, \Phi) = \prod_{k=1}^{K} \prod_{v=1}^{V} \varphi_{k,v}^{n_k^v}$

Dirichlet Distribution: $p(\Phi|\beta) = \prod_{k=1}^{K} \frac{1}{B(\beta)} \prod_{v=1}^{V} \varphi_{k,v}^{\beta_v - 1}$

$$\int \prod_{i=1}^{V} p_i^{\alpha_i - 1}\, dp = B(\alpha)$$

$$p(Z|\alpha) = \int p(Z|\Theta)p(\Theta|\alpha)\, d\Theta = \int \prod_{m=1}^{M} \frac{1}{B(\alpha)} \prod_{k=1}^{K} \vartheta_{m,k}^{n_m^k + \alpha_k - 1}\, d\Theta = \prod_{m=1}^{M} \frac{B(n_m + \alpha)}{B(\alpha)}$$

Multinomial Distribution: $p(Z|\Theta) = \prod_{m=1}^{M} \prod_{k=1}^{K} \vartheta_{m,k}^{n_m^k}$

Dirichlet Distribution: $p(\Theta|\alpha) = \prod_{m=1}^{M} \prod_{k=1}^{K} \vartheta_{m,k}^{\alpha_k - 1}$

Gibbs Sampling (Cont')

$$p(z_i = k | Z_{-i}, W_{-i}, \alpha, \beta) = \frac{p(Z, W | \alpha, \beta)}{p(Z_{-i}, W_{-i} | \alpha, \beta)}$$

$$= \frac{\prod_{k=1}^{K} B(n_k + \beta) \prod_{m=1}^{M} B(n_m + \alpha)}{\prod_{k=1}^{K} B(n_{k,-i} + \beta) \prod_{m=1}^{M} B(n_{m,-i} + \alpha)}$$

$$= \frac{B(n_k + \beta)}{B(n_{k,-i} + \beta)} \cdot \frac{B(n_m + \alpha)}{B(n_{m,-i} + \alpha)}$$

$$= \frac{\frac{\prod_{v=1}^{V} \Gamma(n_k^v + \beta_v)}{\Gamma(\sum_{v=1}^{V} n_k^v + \beta_v)}}{\frac{\prod_{v=1}^{V} \Gamma(n_{k,-i}^v + \beta_v)}{\Gamma(\sum_{v=1}^{V} n_{k,-i}^v + \beta_v)}} \cdot \frac{\frac{\prod_{k=1}^{K} \Gamma(n_m^k + \alpha_k)}{\Gamma(\sum_{k=1}^{K} n_m^k + \alpha_k)}}{\frac{\prod_{k=1}^{K} \Gamma(n_{m,-i}^k + \alpha_k)}{\Gamma(\sum_{k=1}^{K} n_{m,-i}^k + \alpha_k)}}$$

$$\boxed{\Gamma(y) = (y - 1)!}$$

$$= \frac{\frac{\Gamma(n_k^v + \beta_v)}{\Gamma(\sum_{v=1}^{V} n_k^v + \beta_v)}}{\frac{\Gamma(n_{k,-i}^v + \beta_v)}{\Gamma(\sum_{v=1}^{V} n_{k,-i}^v + \beta_v)}} \cdot \frac{\frac{\Gamma(n_m^k + \alpha_k)}{\Gamma(\sum_{k=1}^{K} n_m^k + \alpha_k)}}{\frac{\Gamma(n_{m,-i}^k + \alpha_k)}{\Gamma(\sum_{k=1}^{K} n_{m,-i}^k + \alpha_k)}}$$

$$= \frac{n_k^v + \beta_v - 1}{\sum_{v=1}^{V}(n_k^v + \beta_v) - 1} \frac{n_m^k + \alpha_k - 1}{\sum_{k=1}^{K}(n_m^k + \alpha_k) - 1}$$

$$\propto \frac{n_k^v + \beta_v - 1}{\sum_{v=1}^{V}(n_k^v + \beta_v) - 1} (n_m^k + \alpha_k - 1)$$

Gibbs Sampling (Cont')

- If z is assigned, the unknown parameters are ϑ and Φ, Bayesian Estimation

$$p(\vartheta_m|w,z,\alpha) = \frac{p(w,z|\vartheta_m,\alpha)p(\vartheta_m|\alpha)}{\int p(w,z|\vartheta_m,\alpha)p(\vartheta_m|\alpha)d\vartheta_m}$$

$$= \frac{1}{\mathcal{Z}_{\vartheta_m}} \prod_{n=1}^{N_m} p(z_{m,n}|\vartheta_m)p(\vartheta_m|\alpha) = Dir(n_m + \alpha)$$

$$p(\varphi_k|w,z,\beta) = \frac{p(w,z|\varphi_k,\beta)p(\varphi_k|\beta)}{\int p(w,z|\varphi_k,\beta)p(\varphi_k|\beta)d\varphi_k}$$

$$= \frac{1}{\mathcal{Z}_{\vartheta_m}} \prod_{k=1}^{K} p(w|\varphi_k)p(\varphi_k|\beta) = Dir(n_k + \beta)$$

$$< Dir(\alpha) >= \frac{\alpha_i}{\sum_i \alpha_i}$$

$$\vartheta_{m,k} = \frac{n_m^k + \alpha_k}{\sum_{k=1}^{K}(n_m^k + \alpha_k)}$$

$$\varphi_{k,v} = \frac{n_k^v + \beta_v}{\sum_{v=1}^{V}(n_k^v + \beta_v)}$$

⊡ initialisation
zero all count variables, $n_m^{(k)}, n_m, n_k^{(t)}, n_k$
for all documents $m \in [1, M]$ **do**
  for all words $n \in [1, N_m]$ in document m **do**
    sample topic index $z_{m,n}=k \sim \text{Mult}(1/K)$
    increment document–topic count: $n_m^{(k)} + 1$
    increment document–topic sum: $n_m + 1$
    increment topic–term count: $n_k^{(t)} + 1$
    increment topic–term sum: $n_k + 1$
  end for
end for
⊡ Gibbs sampling over burn-in period and sampling period
while not finished **do**
  for all documents $m \in [1, M]$ **do**
    for all words $n \in [1, N_m]$ in document m **do**
      ⊡ for the current assignment of k to a term t for word $w_{m,n}$:
      decrement counts and sums: $n_m^{(k)} - 1; n_m - 1; n_k^{(t)} - 1; n_k - 1$
      ⊡ multinomial sampling acc. to Eq. 79 (decrements from previous step):
      sample topic index $\tilde{k} \sim p(z_i|\vec{z}_{\neg i}, \vec{w})$
      ⊡ use the new assignment of $z_{m,n}$ to the term t for word $w_{m,n}$ to:
      increment counts and sums: $n_m^{(\tilde{k})} + 1; n_m + 1; n_{\tilde{k}}^{(t)} + 1; n_{\tilde{k}} + 1$
    end for
  end for
  ⊡ check convergence and read out parameters
  if converged and L sampling iterations since last read out **then**
    ⊡ the different parameters read outs are averaged.
    read out parameter set $\underline{\Phi}$ according to Eq. 82
    read out parameter set $\underline{\Theta}$ according to Eq. 83
  end if
end while

MCMC和Gibbs Sampling

- 随机模拟（蒙特卡洛方法，Monte Carlo Simulation）
- 问题：给定一个概率分布p(x)，如何在计算机中生成一个符合这个分布的样本？
  - 伪随机数
  - 假设U_0, U_1独立同分布

$$U_1, U_2 \sim Uniform[0,1]$$

$$Z_0 = \sqrt{-2\ln U_1} cos(2\pi U_2)$$
$$Z_1 = \sqrt{-2\ln U_1} sin(2\pi U_2)$$

  - 则Z_0, Z_1独立且符合标准正态分布

部分资料来自于靳志辉《LDA数学八卦》

马尔可夫状态转移及其平稳分布

- 马尔可夫性

$$P(X_{t+1}=x|X_t,X_{t-1},\cdots)=P(X_{t+1}=x|X_t)$$

- 例子：人口按照经济状况分为：下层、中层和上层，一个人的经济收入的重要决定因素是其父母的经济收入

		子代		
	State	1	2	3
父代	1	0.65	0.28	0.07
	2	0.15	0.67	0.18
	3	0.12	0.36	0.52

$$P=\begin{bmatrix}0.65 & 0.28 & 0.07\\ 0.15 & 0.67 & 0.18\\ 0.12 & 0.36 & 0.52\end{bmatrix}$$

平稳分布

- $\pi_0 = [\pi_0(1), \pi_0(2), \pi_0(3)]$
- $\pi_n = \pi_{n-1}P = \cdots = \pi_0 P^n$
- 假设：$\pi_0 = [0.21, 0.68, 0.11]$

第n代人	下层	中层	上层
0	0.210	0.680	0.110
1	0.252	0.554	0.194
2	0.270	0.512	0.218
3	0.278	0.497	0.225
4	0.282	0.490	0.226
5	0.285	0.489	0.225
6	0.286	0.489	0.225
7	0.286	0.489	0.225
8	0.289	0.488	0.225
9	0.286	0.489	0.225
10	0.286	0.489	0.225
...	...	...	...

平稳分布

- $\pi_0 = [0.75, 0.15, 0.1]$

第n代人	下层	中层	上层
0	0.75	0.15	0.1
1	0.522	0.347	0.132
2	0.407	0.426	0.167
3	0.349	0.459	0.192
4	0.318	0.475	0.207
5	0.303	0.482	0.215
6	0.295	0.485	0.220
7	0.291	0.487	0.222
8	0.289	0.488	0.225
9	0.286	0.489	0.225
10	0.286	0.489	0.225
...	...	...	...

$$P^{20} = P^{21} = \cdots = P^{100} = \cdots = \begin{bmatrix} 0.286 & 0.489 & 0.225 \\ 0.286 & 0.489 & 0.225 \\ 0.286 & 0.489 & 0.225 \end{bmatrix}$$

定理

- 定理：如果一个非周期马氏链具有转移矩阵P，且它的任何两个状态是联通的，那么$\lim_{n\to\infty} P_{ij}^n$存在且与$i$无关，记为$\lim_{n\to\infty} P_{ij}^n = \pi(j)$，则有

$$\lim_{n\to\infty} P^n = \begin{bmatrix} \pi(1) & \pi(2) & \cdots & \pi(j) & \cdots \\ \pi(1) & \pi(2) & \cdots & \pi(j) & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \pi(1) & \pi(2) & \cdots & \pi(j) & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \end{bmatrix}$$

$$\pi(j) = \sum_{i=0}^{\infty} \pi(i) P_{ij}$$

π 是方程$\pi P = \pi$ 的唯一非负解

$$\pi = [\pi(1), \pi(2), \cdots, \pi(j), \cdots], \quad \sum_{i=0}^{\infty} \pi_i = 1$$

- 这时，我们称π是马氏链平稳分布（所有MCMC方法的理论基础）

样本生成

- X_i的概率分布为π_i，从初始概率π_0出发

$$
\begin{aligned}
X_0 &\sim \pi_0(x) \\
X_1 &\sim \pi_1(x) \\
&\cdots \\
X_n &\sim \pi_n(x) = \pi(x) \\
X_{n+1} &\sim \pi(x) \\
X_{n+2} &\sim \pi(x) \\
&\cdots
\end{aligned}
$$

- 则$X_n, X_{n+1}, X_{n+2,\ldots\ldots} \sim \pi(x)$
- 如果从一个具体的初始状态x_0出发，沿着马氏链按照状态转移矩阵跳转，那么我得到一个转移状态的序列$x_0, x_2, \ldots, x_n, x_{n+1}, \ldots$，则$x_n, x_{n+1}, \ldots$ 都是平稳分布$\pi(x)$的样本

Markov Chain Monte Carlo

- 关键：构建一个状态转移矩阵P，使得其马氏链上的平稳分布恰好是$\pi(x)$
- 定理（细致平稳条件）：如果非周期马氏链的转移矩阵P和分布$\pi(x)$满足

$$\pi(i)P_{ij} = \pi(j)P_{ji} \qquad \text{for all} \quad i, j$$

- 则$\pi(x)$是马氏链的平稳分布

$$\sum_{i=1}^{\infty} \pi(i)P_{ij} = \sum_{i=1}^{\infty} \pi(j)P_{ji} = \pi(j)\sum_{i=1}^{\infty} P_{ji} = \pi(j)$$

$$\Rightarrow \pi P = \pi$$

由于π是方程$\pi P = \pi$的解，所以π是一个平稳分布

Markov Chain Monte Carlo

- 假设一个状态转移矩阵Q，从状态i到状态j的转移概率为$q(j|i)$或者$q(i \to j)$

$$p(i)q(i,j) \neq p(j)q(j,i)$$

- 引入$\alpha(i,j)$，使得

$$p(i)q(i,j)\alpha(i,j) = p(j)q(j,i)\alpha(j,i)$$

$$\alpha(i,j) = p(j)q(j,i) \quad \alpha(j,i) = p(i)q(i,j)$$

- 状态转移矩阵为$Q'(i,j)$

$$p(i)\underbrace{q(i,j)\alpha(i,j)}_{Q'(i,j)} = p(j)\underbrace{q(j,i)\alpha(j,i)}_{Q'(j,i)}$$

MCMC采样算法

1: 初始化马氏链初始状态$X_0 = x_0$

2: 对$t = 0, 1, 2, \cdots$, 循环以下过程进行采样

- 第t个时刻马氏链状态为$X_t = x_t$, 采样$y \sim q(x|x_t)$
- 从均匀分布采样$u \sim Uniform[0, 1]$
- 如果$u < \alpha(x_t, y) = p(y)q(x_t|y)$ 则接受转移$x_t \to y$，即$X_{t+1} = y$
- 否则不接受转移，即$X_{t+1} = x_t$

Metropolis-Hastings算法

- $\alpha(i,j)$太小，收敛太慢
- 解决：等比例放大$\alpha(i,j)$和$\alpha(j,i)$，直到两数中最大的一个放大到1

1: 初始化马氏链初始状态$X_0 = x_0$

2: 对$t = 0, 1, 2, \cdots$，循环以下过程进行采样

- 第t个时刻马氏链状态为$X_t = x_t$，采样$y \sim q(x|x_t)$
- 从均匀分布采样$u \sim Uniform[0,1]$
- 如果$u < \alpha(x_t, y) = \min\left\{\frac{p(y)q(x_t|y)}{p(x_t)p(y|x_t)}, 1\right\}$则接受转移$x_t \to y$，即$X_{t+1} = y$
- 否则不接受转移，即$X_{t+1} = x_t$

Gibbs Sampling

- 高维情况下如何采样？
- 二维情况：假设一个概率分布p(x,y)，有x坐标相同的两个点 $A(x_1, y_1), B(x_1, y_2)$，则有

$$p(x_1, y_1)p(y_2|x_1) = p(x_1)p(y_1|x_1)p(y_2|x_1)$$

$$p(x_1, y_2)p(y_1|x_1) = p(x_1)p(y_2|x_1)p(y_1|x_1)$$

$$p(x_1, y_1)p(y_2|x_1) = p(x_1, y_2)p(y_1|x_1)$$

$$p(A)p(y_2|x_1) = p(B)p(y_1|x_1)$$

- A和B两个点之间的转移满足细致平稳条件

Gibbs Sampling

- 对于$A(x_1, y_1), C(x_2, y_1)$，也有

$$p(A)p(x_2|y_1) = p(C)p(x_1|y_1)$$

- 则可以构造任意两点的状态转移概率矩阵Q

$$Q(A \to B) = p(y_B|x_1) \qquad \text{如果} \quad x_A = x_B = x_1$$

$$Q(A \to C) = p(x_C|y_1) \qquad \text{如果} \quad y_A = y_C = y_1$$

$$Q(A \to D) = 0 \qquad \text{其它}$$

Gibbs Sampling

- 对于任意两点X,Y，基于上述所构造的状态转移矩阵Q，则有

$$p(X)Q(X \to Y) = p(Y)Q(Y \to X)$$

1: 随机初始化$X_0 = x_0 Y_0 = y_0$

2: 对$t = 0, 1, 2, \cdots$ 循环采样

1. $y_{t+1} \sim p(y|x_t)$
2. $x_{t+1} \sim p(x|y_{t+1})$

Gibbs Sampling（高维）

1: 随机初始化$\{x_i : i = 1, \cdots, n\}$

2: 对$t = 0, 1, 2, \cdots$ 循环采样

1. $x_1^{(t+1)} \sim p(x_1 | x_2^{(t)}, x_3^{(t)}, \cdots, x_n^{(t)})$
2. $x_2^{(t+1)} \sim p(x_2 | x_1^{(t+1)}, x_3^{(t)}, \cdots, x_n^{(t)})$
3. $\cdots$
4. $x_j^{(t+1)} \sim p(x_j | x_1^{(t+1)}, \cdots, x_{j-1}^{(t+1)}, x_{j+1}^{(t)}, \cdots, x_n^{(t)})$
5. $\cdots$
6. $x_n^{(t+1)} \sim p(x_n | x_1^{(t+1)}, x_2^{t}, \cdots, x_{n-1}^{(t+1)})$

Questions