法律声明

- 课程详情请咨询
  - 微信公众号：北风教育
  - 官方网址：http://www.ibeifeng.com/

人工智能之机器学习

主题模型

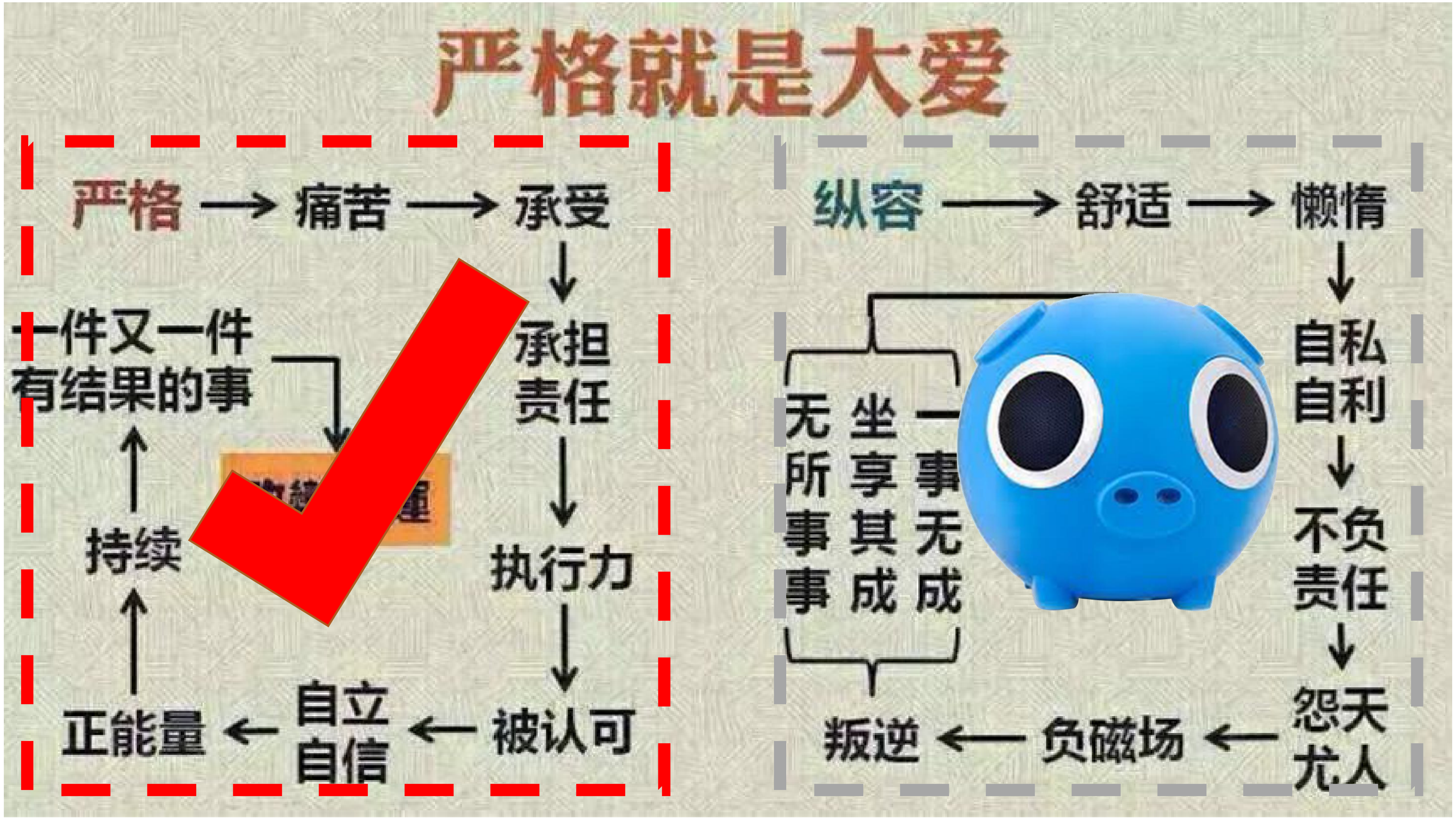
主讲人：Gerry

上海育创网络科技有限公司

课程要求

- 课上课下“九字”真言
  - 认真听，善摘录，勤思考
  - **多温故，乐实践**，再发散
- 四不原则
  - 不懒散惰性，不迟到早退
  - 不请假旷课，不拖延作业
- 一点注意事项
  - 违反“四不原则”，不包就业和推荐就业

严格是大爱


严格就是大爱
严格
痛苦
承受
承担
责任
执行力
被认可
自立
自信
正能量
持续
一件又一件
有结果的事
纵容
舒适
懒惰
自私
自利
不负
责任
怨天
尤人
负磁场
叛逆
无所事事
坐享其成
一事无成

寄语

做别人不愿做的事，

做别人不敢做的事，

做别人做不到的事。

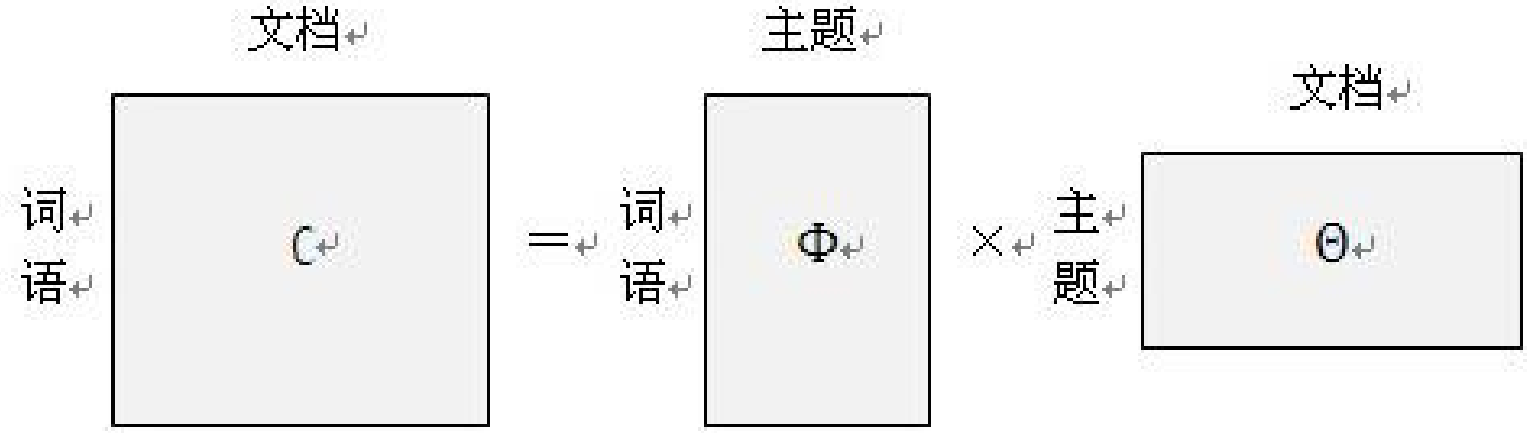
课程内容

- 主题模型
- LSA
- LDA

主题模型

- 怎样才能生成主题？对文章的主题应该怎么分析？这是主题模型要解决的问题。

$$p(\text{词语}|\text{文档})=\sum_{\text{主题}} p(\text{词语}|\text{主题})\times p(\text{主题}|\text{文档})$$



LSA

- 潜在语义分析(Latent Semantic Analysis, LSA)，也叫做Latent Semantic Indexing, LSI. 是一种常用的简单的主题模型。LSA是基于奇异值分解(SVD)的方法得到文本主题的一种方式。

$$A_{m*n} = U_{m*m} \Sigma_{m*n} V^T_{n*n} \qquad A_{m*n} \approx U_{m*k} \Sigma_{k*k} V^T_{n*k}$$

- 总结：我们输入的有m个文本，每个文本有n个词。而A_{ij}则对应第i个文本的第j个词的特征值。k是我们假设的主题数，一般要比文本数少。SVD分解后，U_{il}对应第i个文本和第l个主题的相关度。V_{jm}对应第j个词和第m个词义的相关度。Σ_{lm}对应第l个主题和第m个词义的相关度。

LSA案例

- 假设有10个词、3个文本对应的词频TF矩阵如下：

$$
\begin{array}{l c c}
\text{Terms} & & \begin{array}{ccc} d1 & d2 & d3 \end{array} \\
\begin{array}{l} \text{a} \\ \text{arrived} \\ \text{damaged} \\ \text{delivery} \\ \text{fire} \\ \text{gold} \\ \text{in} \\ \text{of} \\ \text{shipment} \\ \text{silver} \\ \text{truck} \end{array}
& A = &
\begin{bmatrix}
1 & 1 & 1 \\
0 & 1 & 1 \\
1 & 0 & 0 \\
0 & 1 & 0 \\
1 & 0 & 0 \\
1 & 0 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 0 & 1 \\
0 & 2 & 0 \\
0 & 1 & 1
\end{bmatrix}
\end{array}
\qquad
\begin{array}{c c}
 & q \\
q = &
\begin{bmatrix}
0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1
\end{bmatrix}
\end{array}
$$

LSA案例

- 假定主题数为2，通过SVD降维后的三个矩阵分布为：

$$k = 2$$

$$U \approx U_k = \begin{bmatrix} -0.4201 & 0.0748 \\ -0.2995 & -0.2001 \\ -0.1206 & 0.2749 \\ -0.1576 & -0.3046 \\ -0.1206 & 0.2749 \\ -0.2626 & 0.3794 \\ -0.4201 & 0.0748 \\ -0.4201 & 0.0748 \\ -0.2626 & 0.3794 \\ -0.3151 & -0.6093 \\ -0.2995 & -0.2001 \end{bmatrix}$$

$$\Sigma \approx \Sigma_k = \begin{bmatrix} 4.0989 & 0.0000 \\ 0.0000 & 2.3616 \end{bmatrix}$$

$$V \approx V_k = \begin{bmatrix} -0.4945 & 0.6492 \\ -0.6458 & -0.7194 \\ -0.5817 & 0.2469 \end{bmatrix}$$

$$V^T \approx V_k^T = \begin{bmatrix} -0.4945 & -0.6458 & -0.5817 \\ 0.6492 & -0.7194 & 0.2469 \end{bmatrix}$$

LSA

- 通过SVD矩阵分解我们可以得到文本、词与主题、语义之间的相关性，但是这个时候计算出来的内容存在负数，我们比较难解释，所以我们可以对LSI得到文本主题矩阵使用余弦相似度计算文本的相似度的计算。最终我们得到第一个和第三个文档比较相似，和第二个文档不太相似。(备注：这个时候直接在文本主题矩阵的基础上直接应用聚类算法即可)

$$sim(d_1,d_2)=\frac{(-0.4945)*(-0.6458)+0.6492*(-0.7194)}{\sqrt{(-0.4945)^2+0.6492^2}*\sqrt{(-0.6458)^2+(-0.7194)^2}}=-0.1872$$

$$sim(d_1,d_3)=\frac{(-0.4945)*(-0.5817)+0.6492*0.2469}{\sqrt{(-0.4945)^2+0.6492^2}*\sqrt{(-0.5817)^2+0.2469^2}}=0.8686$$

LSA主题模型总结

- 除非数据规模比较小，而且希望快速的粗粒度的找出一些主题分布关系，否则我们一般不会使用LSA主题模型。
- 优点：
  - 原理简单，一次SVD分解即可得到主题模型，可以同时解决词义的问题。
- 缺点：
  - SVD分解的计算非常耗时，对于高维度矩阵做SVD分解非常困难；
  - 主题模型数量的选取对于结果的影响非常大，很难选择合适的k值；
  - LSA模型不是概率模型，缺乏统计基础，结果难以直观的解释。

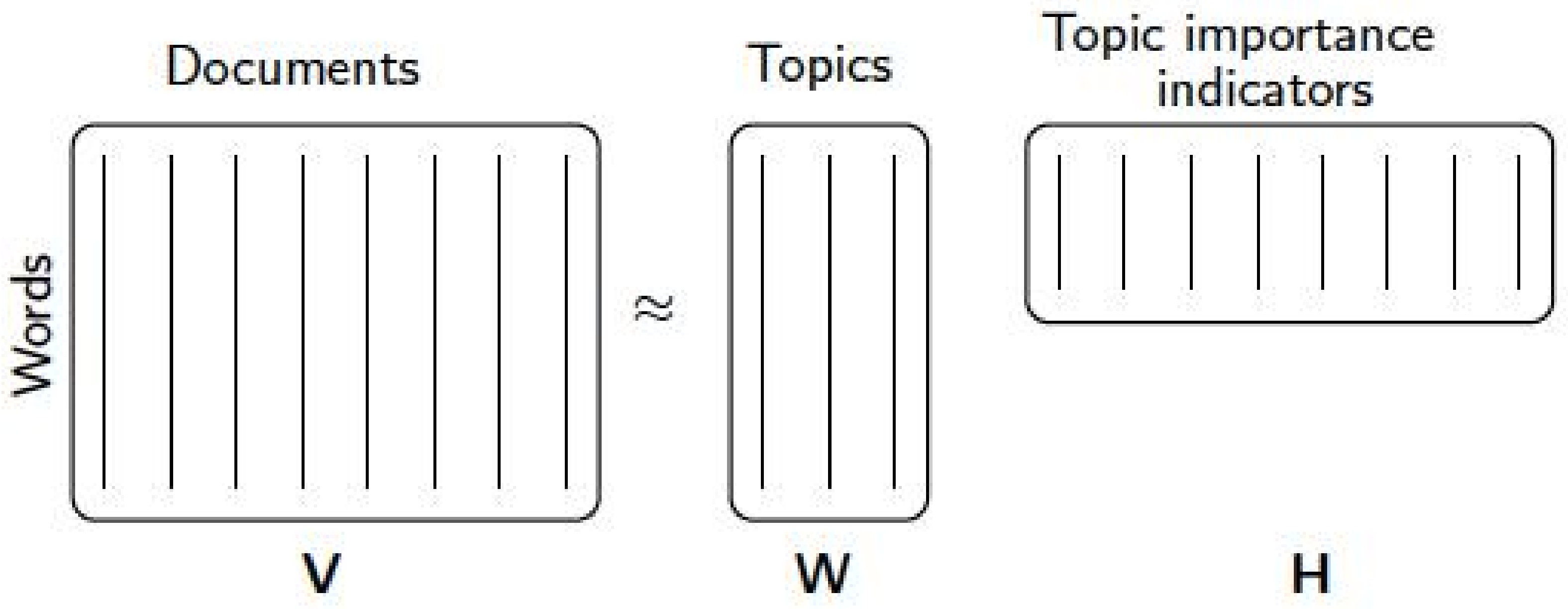
NMF

- 非负矩阵分解(Non-negative Matrix Factorization, NMF)是一种常用的矩阵分解方式，常用于矩阵分解、降维、主题模型等应用场景。
- NMF虽然和SVD一样都是矩阵分解，但是NMF不同的是：它的目标希望是将矩阵分解成为两个子矩阵。
- 参考：https://www.csie.ntu.edu.tw/~cjlin/papers/pgradnmf.pdf

$$V_{m*n} \approx W_{m*k} H_{k*n}$$

NMF

- 在NMF中求解出来的W和H，分别体现的是文本和主题的概率相关度，以及词和主题的概率相关度；



NMF

- NMF的期望是找到两个W、H矩阵，使得WH的矩阵乘积结果和对应的原矩阵V对应位置的值相比误差尽可能的小。

$$\begin{aligned} \min_{W,H} \quad & f(W,H) \equiv \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{m}\left(V_{ij}-(WH)_{ij}\right)^2 \\ \text{subject to} \quad & W_{ia} \geq 0,\ H_{bj} \geq 0,\quad \forall i,a,b,j. \end{aligned}$$

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\left(V_{ij}-(WH)_{ij}\right)^2 = \|V-WH\|_F^2$$

NMF

- NMF的目标函数中总共包含了m*k+k*n个参数，可以直接使用梯度下降法或者拟牛顿法来进行求解。

$$
\begin{aligned}
W^{k+1} &= \max(0, W^k - \alpha_k \nabla_W f(W^k, H^k)) \\
H^{k+1} &= \max(0, H^k - \alpha_k \nabla_H f(W^k, H^k))
\end{aligned}
$$

NMF

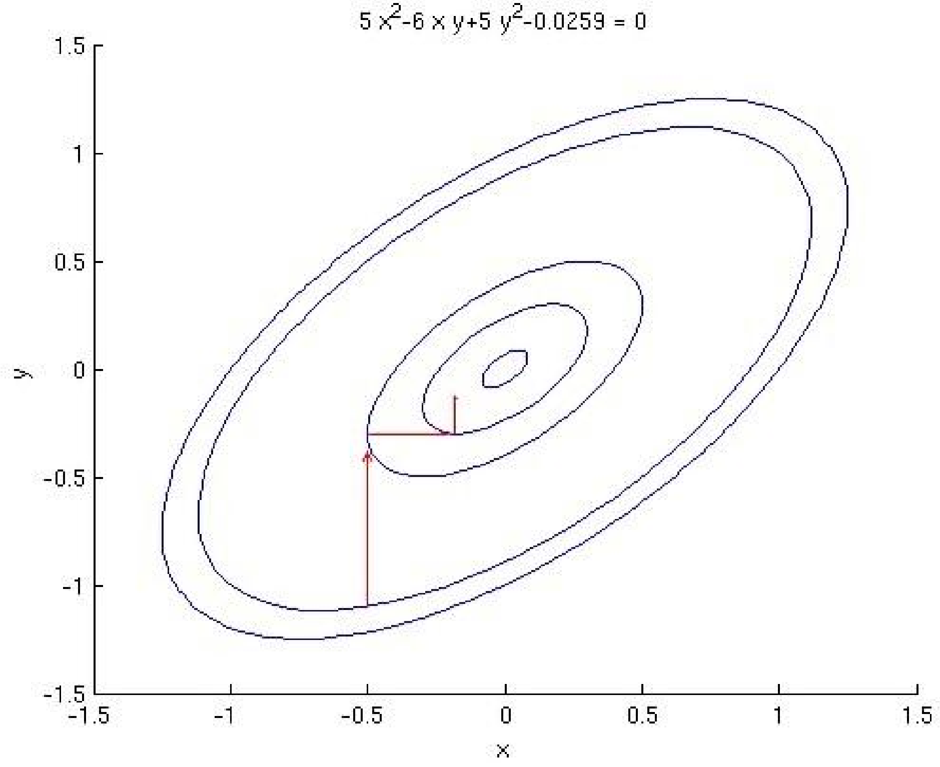
- 为了防止过拟合，也可以在NMF的目标函数的基础上添加一个正则化项

$$\frac{1}{2}||X - WH||_{Fro}^2 + \alpha\rho||W||_1 + \alpha\rho||H||_1 + \frac{\alpha(1-\rho)}{2}||W||_{Fro}^2 + \frac{\alpha(1-\rho)}{2}||H||_{Fro}^2$$

- 但是当加入L1正则项后，由于没法求解出正常的导函数出来(导函数不是连续的)，也就没法使用梯度下降法和拟牛顿法求解参数，此时一般采用坐标轴下降法来进行参数的求解。

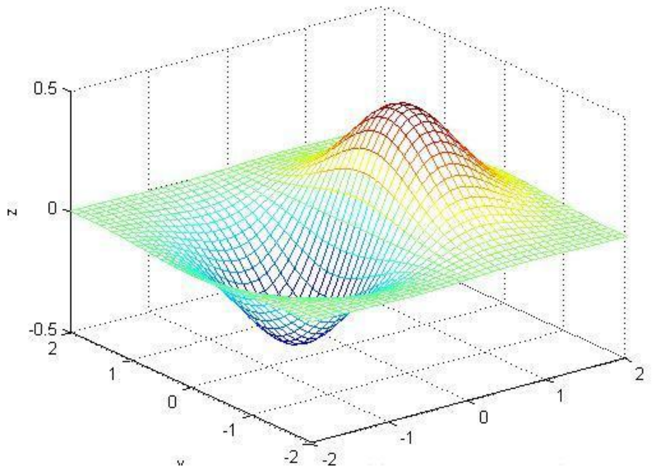
坐标轴下降法

- 坐标轴下降法(Coordinate Descent， CD)是一种迭代法，通过启发式的方法一步步的迭代求解函数的最小值，和梯度下降法(GD)不同的时候，坐标轴下降法是沿着坐标轴的方向去下降，而不是采用梯度的负方向下降。



坐标轴下降法

- 坐标轴下降法利用EM算法的思想，在参数更新过程中，每次均先固定m-1个参数值，求解剩下的一个参数的局部最优解；然后进行迭代式的更新操作。
- 坐标轴下降法的核心思想是多变量函数F(X)可以通过每次沿着一个方向优化来获取最小值；其数学依据是：对于一个可微凸函数f(θ)，其中θ为n*1的向量，如果对于一个解$\theta=(\theta_1,\theta_2,\ldots,\theta_n)$，使得f(θ)在某个坐标轴$\theta_i(i=1,2,..,n)$上都能达到最小值，则$\theta=(\theta_1,\theta_2,\ldots,\theta_n)$就是的f(θ)全局的最小值点



坐标轴下降法

- 在坐标轴下降法中，优化方向从算法的一开始就固定了，即沿着坐标的方向进行变化。在算法中，循环最小化各个坐标方向的目标函数。即：如果x^k给定，那么x^{k+1}的第i维度为:

$$\mathbf{x}_i^{k+1} = \arg\min_{y\in\mathbb{R}} f(x_1^{k+1}, \dots, x_{i-1}^{k+1}, y, x_{i+1}^k, \dots, x_n^k);$$

- 因此，从一个初始的x0求得函数F(x)的局部最优解，可以迭代获取x0、x1、x2...的序列，从而可以得到:

$$F(\mathbf{x}_0) \geq F(\mathbf{x}_1) \geq F(\mathbf{x}_2) \geq \cdots$$

坐标轴下降法算法过程

- 1. 给θ向量随机选取一个初值，记做θ^0；
- 2. 对于第k轮的迭代，从θ_1^k开始计算，θ_1^k到为止，计算公式如下：

$$\theta_1^k = \arg\min_{\theta_1} J\left(\theta_1, \theta_2^{k-1}, \theta_3^{k-1}, \ldots, \theta_n^{k-1}\right)$$

$$\theta_2^k = \arg\min_{\theta_2} J\left(\theta_1^k, \theta_2, \theta_3^{k-1}, \ldots, \theta_n^{k-1}\right)$$

……

$$\theta_n^k = \arg\min_{\theta_n} J\left(\theta_1^k, \theta_2^k, \theta_3^k, \ldots, \theta_n\right)$$

- 检查θ^k和θ^{k-1}向量在各个维度上的变化情况，如果所有维度的变化情况都比较小的话，那么认为结束迭代，否则继续k+1轮的迭代。
- 备注：在求解每个参数局部最优解的时候可以求导的方式来求解。

二项分布

- 二项分布是从伯努利分布推进的。伯努利分布，又称两点分布或0-1分布，是一个离散型的随机分布，其中的随机变量只有两类取值，非正即负{+，-}。而二项分布即重复n次的伯努利试验，记为 $X \sim b(n,p)$。简言之，只做一次实验，是伯努利分布，重复做了n次，是二项分布。二项分布的概率密度函数为：

$$P(K=k)=\binom{n}{k}p^k(1-p)^{n-k} \qquad \binom{n}{k}=\frac{n!}{k!(n-k)!}$$

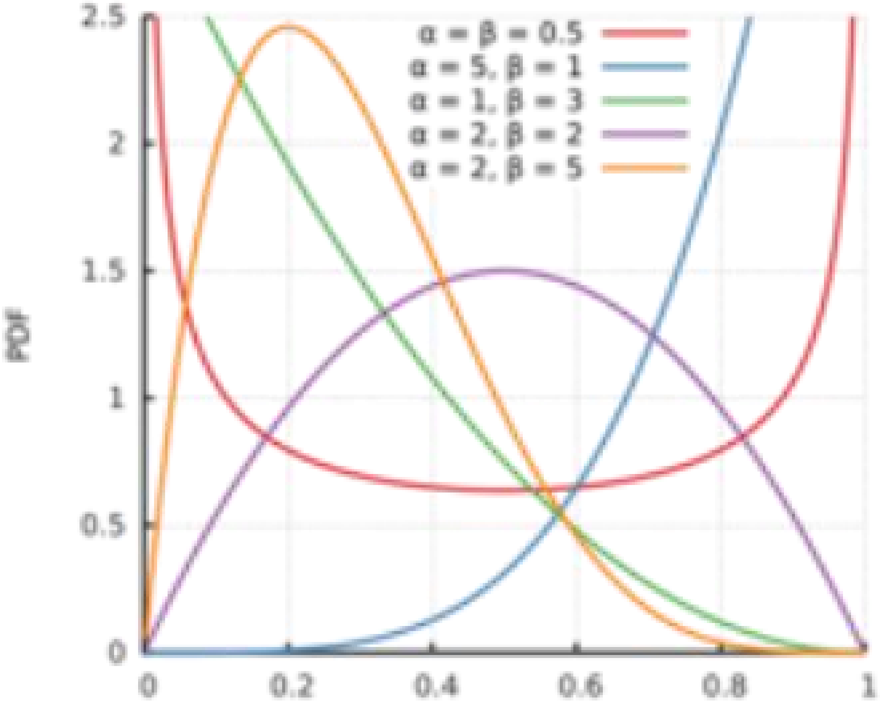
Beta分布

- Beta分布是二项分布的共轭分布，条件概率公式如下：

$$f(x)=\frac{1}{B(\alpha,\beta)}x^{\alpha-1}(1-x)^{\beta-1},x\in[0,1]$$

$$\frac{1}{B(\alpha,\beta)}=\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}$$

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



多项分布

- 是二项分布扩展到多维的情况
- 多项分布是指单次试验中的随机变量的取值不再是0-1的，而是有多种离散值可能（1,2,3...,k）。比如投掷6个面的骰子实验，N次实验结果服从K=6的多项分布。其中

$$\sum_{i=1}^{k} p_i = 1, p_i > 0$$

- 多项分布的概率密度函数为：

$$P(x_1, x_2, \ldots, x_k; n, p_1, p_2, \ldots, p_k) = \frac{n!}{x_1! \cdots x_k!} p_1^{x_1} \cdots p_k^{x_k}$$

Dirichlet分布

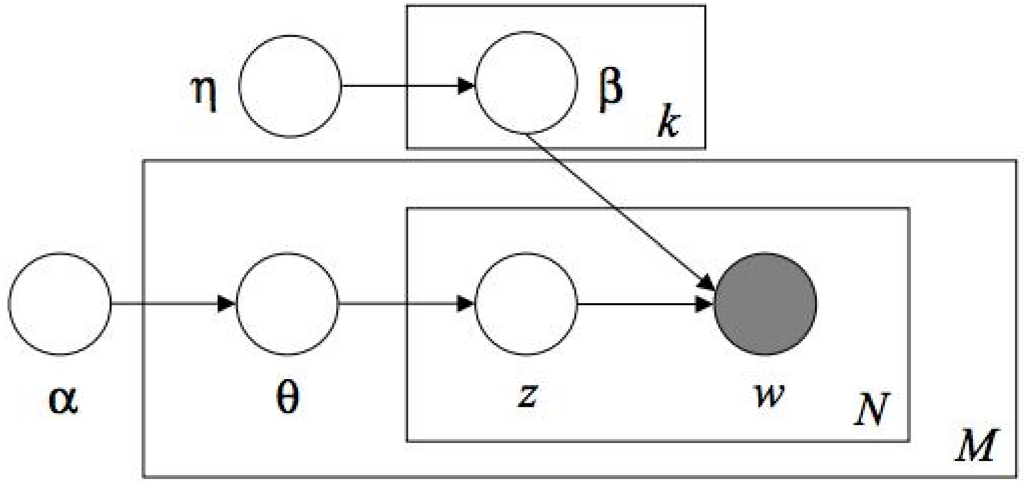
- 是Beta分布扩展到多维的情况

$$f(x_1,\cdots,x_K;\alpha_1,\cdots,\alpha_K)=\frac{1}{B(\alpha)}\prod_{K}^{i=1}x_i^{\alpha_i-1},$$

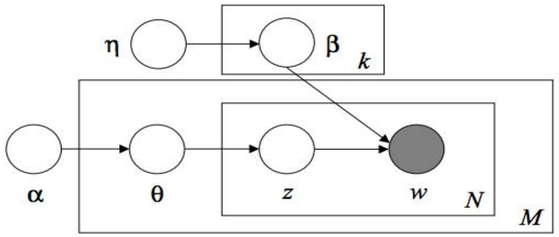
$$B(\alpha)=\frac{\prod_{i=1}^{K}\Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^{K}\alpha_i)},\alpha=(\alpha_1,\cdots,\alpha_K)$$

LDA

- 隐含狄利克雷分布(Latent Dirichlet Allocation, LDA)是一种基于贝叶斯算法模型，利用先验分布对数据进行似然估计并最终得到后验分布的一种方式。LDA是一种比较常用的主题模型。
- LDA假设文档主题是多项分布，多项分布的参数(先验分布)是服从Dirichlet分布，其实LDA是一种三层的贝叶斯模型。

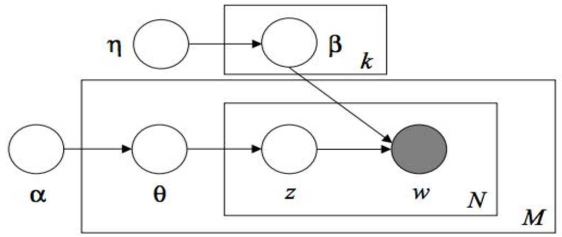


LDA



- 共有M篇文档，每个文档有N_m个单词，一共涉及到K个主题；
- 每篇文档都有各自的主题，主题分布是多项式分布，该多项式分布的参数服从Dirichlet分布，该Dirichlet分布的参数为α；
- 每个主题都有各自的词分布，词分布为为多项式分布，该多项式分布的参数服从Dirichlet分布，该Dirichlet分布的参数为η；
- 对于某篇文档d中的第n个词，首先从该文档的主题分布中采用一个主题，然后再这个主题对应的词分布中采用一个词，不断重复该操作，直到m篇文档全部完成上述过程。

LDA详细解释



- 词汇表中共有V个term(不可重复);
- 语料库中共有m篇文档$d_1,d_2,..,d_m$;对于文档d_i，是由N_i个word组成的(word可重复);语料库共有K个主题$T_1,T_2,...,T_k$;
- α和η是先验分布(Dirichlet分布)的参数；
- θ是每篇文档的**主题分布**，是一个K维的向量；
- 对于第i篇文档d_i，在主题分布$θ_i$下，可以确定一个具体的主题$z_{ij}=k$
- β是每个主题的**词分布**，是一个V维的向量；
- 由z_{ij}选择$β_{zij}$，表示由词分布$β_{zij}$确定term，即可得到最终的观测值w_{ij}。

LDA参数学习-Gibbs采样

- 对于一个n维的概率分布$\pi(x_1,x_2,\ldots,x_n)$，可以通过在n个坐标上轮换采样，来得到新的样本，对于轮换到任意一个坐标x_i上的转移，马尔可夫链的状态转移概率为$p(x_i|x_1,x_2,\ldots,x_{i-1},x_{i+1},\ldots,x_n)$，即固定n-1个坐标轴，在某一个坐标上移动。
- Gibbs采样算法在高维空间采样的时候具有比较高的优势，GIbbs采样的过程比较类似这个坐标轴下降法。

LDA参数学习-Gibbs采样算法流程

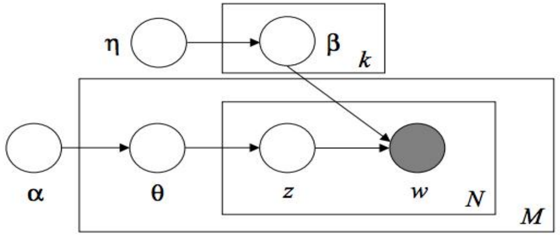
- 1. 输入稳定的分布$\pi(x_1,x_2,\ldots,x_n)$或者对应特征的条件概率分布，设定状态转移次数阈值n_1，需要的样本数n_2；
- 2. 随机初始化状态值$(x_1^1,x_2^1,\ldots,x_n^1)$;
- 3. 进行迭代数据采样(迭代n_1+n_2-1次)
  - 从条件概率分布中采样得到对应的样本

$$x_j^{t+1} \to p\left(x_j \middle| x_1^{t+1}, x_2^{t+1}, \ldots, x_{j-1}^{t+1}, x_{j+1}^{t}, \ldots, x_n^{t}\right)$$

- 4. 最终得到的样本集为:

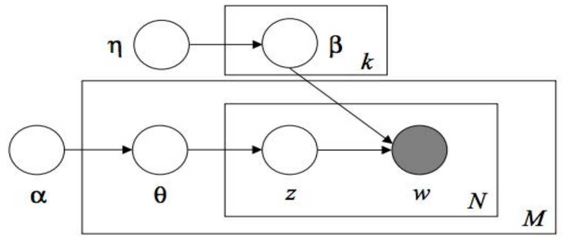
$$\left\{\left(x_1^{n_1}, x_2^{n_1}, \ldots, x_n^{n_1}\right), \ldots, \left(x_1^{n_1+n_2-1}, x_2^{n_1+n_2-1}, \ldots, x_n^{n_1+n_2-1}\right)\right\}$$

LDA参数学习-Gibbs采样



- 给定一个文档集合，w是可以观察到的值，α和η是根据经验给定的先验参数，其它的各个z，θ、β都是未知的隐含变量，都是需要根据观测到的数据进行学习的。
- 具体来讲，所有文档联合起来形成的词向量w是已知数据，但是不知道语料库的主题z的分布。假设可以先求解出w、z的联合分布p(w,z)，进而就可以求出某个词w_i对应主题特征z_i的条件概率分布$p(z_i=k|w,z_{-i})$，其中z_{-i}表示去掉下标为i后的主题分布，有了条件概率，那么就可以使用Gibbs采样，最终可以得到第i个词的主题。
- 如果通过采样得到所有词的主题，那么可以通过统计所有词的主题数，从而得到各个主题的词分布。接着统计各个文档对应词的主题数，从而可以得到各个文档的主题分布。

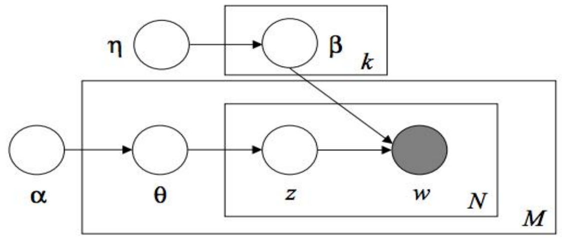
LDA参数学习-Gibbs采样



- 简化Dirichlet分布表达式：

$$Dirichlet(\vec{p}|\vec{\alpha}) = \frac{\Gamma(\sum_{k=1}^{K} \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \prod_{k=1}^{K} p_k^{\alpha_k - 1} = \frac{1}{\Delta(\vec{\alpha})} \prod_{k=1}^{K} p_k^{\alpha_k - 1}$$

LDA参数学习-Gibbs采样



- 计算文档的主题条件分布：

$$p(\vec{z}_d|\vec{\alpha}) = \int p(\vec{z}_d|\vec{\theta}_d)p(\theta_d|\vec{\alpha})d\vec{\theta}_d$$

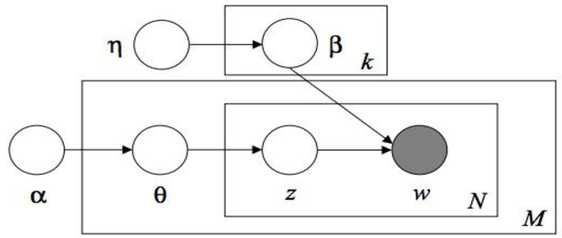
$$= \int \prod_{k=1}^{K} p_k^{n_d^{(k)}} Dirichlet(\vec{\alpha})d\vec{\theta}_d$$

$$= \int \prod_{k=1}^{K} p_k^{n_d^{(k)}} \frac{1}{\Delta(\vec{\alpha})} \prod_{k=1}^{K} p_k^{\alpha_k - 1} d\vec{\theta}_d$$

$$= \frac{1}{\Delta(\vec{\alpha})} \int \prod_{k=1}^{K} p_k^{n_d^{(k)} + \alpha_k - 1} d\vec{\theta}_d$$

$$= \frac{\Delta(\vec{n}_d + \vec{\alpha})}{\Delta(\vec{\alpha})}$$

LDA参数学习-Gibbs采样



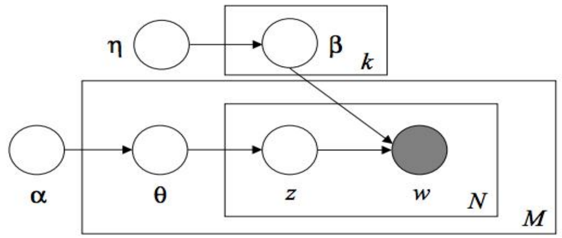
- 在第d个文档中，第k个主题的词的个数表示为：$n_d^{(k)}$, 对应的多项分布的计数可以表示为：

$$\vec{n}_d = (n_d^{(1)}, n_d^{(2)}, \dots n_d^{(K)})$$

- 有了一个文档的主题条件分布，则可以得到所有文档的主题条件分布为:

$$p(\vec{z}|\vec{\alpha}) = \prod_{d=1}^{M} p(\vec{z}_d|\vec{\alpha}) = \prod_{d=1}^{M} \frac{\Delta(\vec{n}_d + \vec{\alpha})}{\Delta(\vec{\alpha})}$$

LDA参数学习-Gibbs采样



- 使用同样的方式，可以得到第k个主题对应的词的条件分布p(w|z,η)为：

$$p(\vec{w}|\vec{z},\vec{\eta}) = \prod_{k=1}^{K} p(\vec{w}_k|\vec{z},\vec{\eta}) = \prod_{k=1}^{K} \frac{\Delta(\vec{n}_k+\vec{\eta})}{\Delta(\vec{\eta})}$$

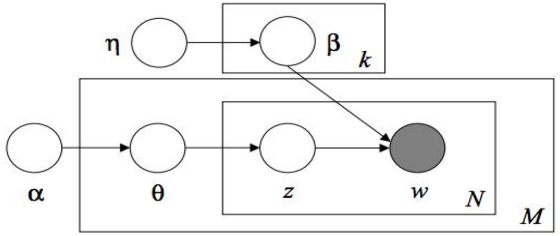
- 其中第k个主题中，第v个词的个数表示为n_k^v;对应的多项式分布计数表示为：

$$\vec{n}_k = (n_k^{(1)}, n_k^{(2)}, \ldots n_k^{(V)})$$

- 最终得到主题和词向量的联合分布为：

$$p(\vec{w},\vec{z}) \propto p(\vec{w},\vec{z}|\vec{\alpha},\vec{\eta}) = p(\vec{z}|\vec{\alpha})p(\vec{w}|\vec{z},\vec{\eta}) = \prod_{d=1}^{M} \frac{\Delta(\vec{n}_d+\vec{\alpha})}{\Delta(\vec{\alpha})} \prod_{k=1}^{K} \frac{\Delta(\vec{n}_k+\vec{\eta})}{\Delta(\vec{\eta})}$$

LDA参数学习-Gibbs采样



- 基于联合分布，就可以使用求解Gibbs采样所需要的条件分布p(z_i=k|w,z_{-i})；对于下标i，由于它对应的词wi是可以观察到的，因此有公式如下：

$$p(z_i = k|\vec{w}, \vec{z}_{\neg i}) \propto p(z_i = k, w_i = t|\vec{w}_{\neg i}, \vec{z}_{\neg i})$$

- 对于zi=k，wi=t，只涉及到第d篇文档和第k个主题两个Dirichlet共轭，即：

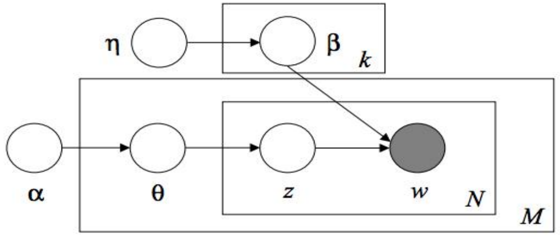
$$\vec{\alpha} \rightarrow \vec{\theta}_d \rightarrow \vec{z}_d$$
$$\vec{\eta} \rightarrow \vec{\beta}_k \rightarrow \vec{w}_{(k)}$$

- 至于其他的Dirichlet共轭和这两个是互相独立的，也就是说从语料库中去掉zi和wi后，并不会改变共轭结构。所以对应的后验分布为：

$$p(\vec{\theta}_d|\vec{w}_{\neg i}, \vec{z}_{\neg i}) = Dirichlet(\vec{\theta}_d|\vec{n}_{d,\neg i} + \vec{\alpha})$$

$$p(\vec{\theta}_d|\vec{w}_{\neg i}, \vec{z}_{\neg i}) = Dirichlet(\vec{\theta}_d|\vec{n}_{d,\neg i} + \vec{\alpha})$$

LDA参数学习-Gibbs采样



- 开始计算Gibbs采样的条件概率：

$$p(z_i = k|\vec{w}, \vec{z}_{\neg i}) \propto p(z_i = k, w_i = t|\vec{w}_{\neg i}, \vec{z}_{\neg i})$$

$$= \int p(z_i = k, w_i = t, \vec{\theta}_d, \vec{\beta}_k|\vec{w}_{\neg i}, \vec{z}_{\neg i})d\vec{\theta}_d d\vec{\beta}_k$$

$$= \int p(z_i = k, \vec{\theta}_d|\vec{w}_{\neg i}, \vec{z}_{\neg i})p(w_i = t, \vec{\beta}_k|\vec{w}_{\neg i}, \vec{z}_{\neg i})d\vec{\theta}_d d\vec{\beta}_k$$

$$= \int p(z_i = k|\vec{\theta}_d)p(\vec{\theta}_d|\vec{w}_{\neg i}, \vec{z}_{\neg i})p(w_i = t|\vec{\beta}_k)p(\vec{\beta}_k|\vec{w}_{\neg i}, \vec{z}_{\neg i})d\vec{\theta}_d d\vec{\beta}_k$$

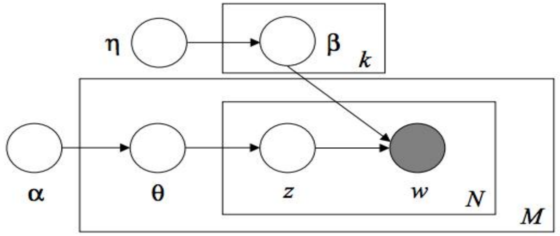
$$= \int p(z_i = k|\vec{\theta}_d)Dirichlet(\vec{\theta}_d|\vec{n}_{d,\neg i} + \vec{\alpha})d\vec{\theta}_d$$

$$* \int p(w_i = t|\vec{\beta}_k)Dirichlet(\vec{\beta}_k|\vec{n}_{k,\neg i} + \vec{\eta})d\vec{\beta}_k$$

$$= \int \theta_{dk}Dirichlet(\vec{\theta}_d|\vec{n}_{d,\neg i} + \vec{\alpha})d\vec{\theta}_d \int \beta_{kt}Dirichlet(\vec{\beta}_k|\vec{n}_{k,\neg i} + \vec{\eta})d\vec{\beta}_k$$

$$= E_{Dirichlet(\theta_d)}(\theta_{dk})E_{Dirichlet(\beta_k)}(\beta_{kt})$$

LDA参数学习-Gibbs采样



- Dirichlet分布的期望公式如下，带入条件概率中，可以得到最终的条件概率公式：

$$E_{Dirichlet(\theta_d)}(\theta_{dk}) = \frac{n_{d,\neg i}^{k} + \alpha_k}{\sum_{s=1}^{K} n_{d,\neg i}^{s} + \alpha_s} \qquad E_{Dirichlet(\beta_k)}(\beta_{kt}) = \frac{n_{k,\neg i}^{t} + \eta_t}{\sum_{f=1}^{V} n_{k,\neg i}^{f} + \eta_f}$$

$$p(z_i = k | \vec{w}, \vec{z}_{\neg i}) = \frac{n_{d,\neg i}^{k} + \alpha_k}{\sum_{s=1}^{K} n_{d,\neg i}^{s} + \alpha_s} \frac{n_{k,\neg i}^{t} + \eta_t}{\sum_{f=1}^{V} n_{k,\neg i}^{f} + \eta_f}$$

LDA参数学习-Gibbs采样训练流程

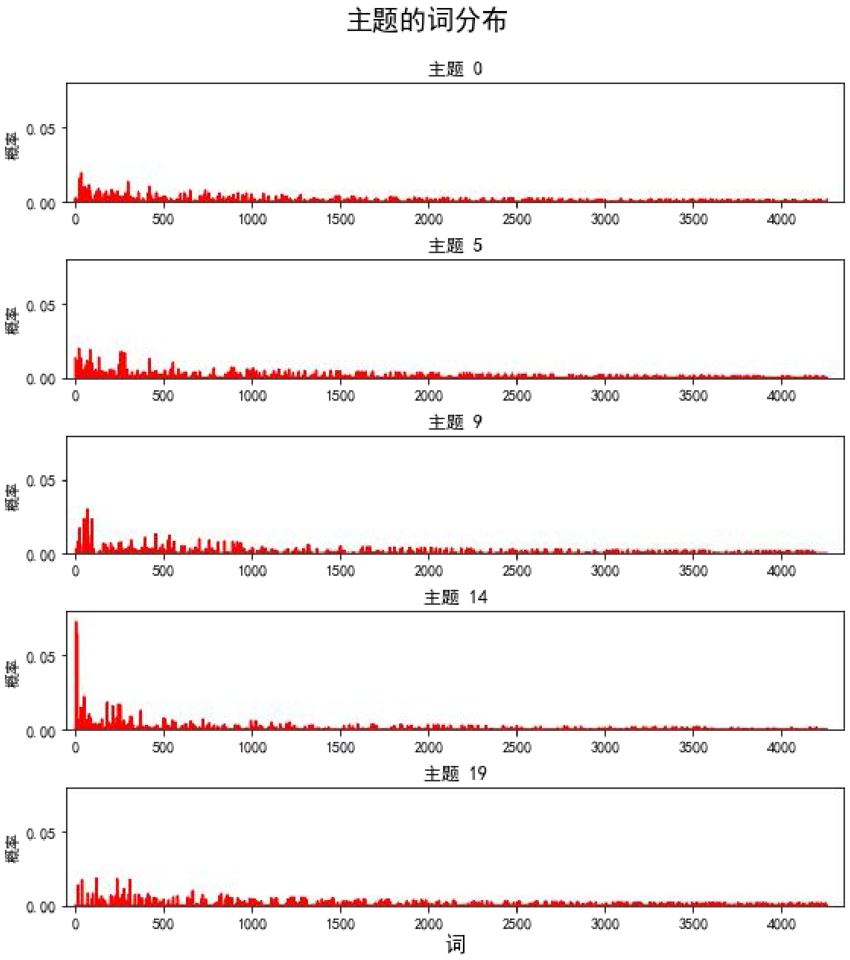
- 1. 选择合适的主题数K，选择合适的超参数α、η
- 2. 对于语料库中每一篇文档的每一个词，随机的赋予一个主题编号z
- 3. 重新扫描语料库，对于每一个词，利用Gibbs采样公式更新它的topic的编号，并更新语料库中该词的编号
- 4. 重复第三步中基于坐标轴轮询的Gibbs采样，直到Gibbs采样收敛。
- 5. 统计语料库中各个文档各个词的主题，得到文档主题分布；然后统计语料库中各个主题词的分布，得到主题与词的分布。

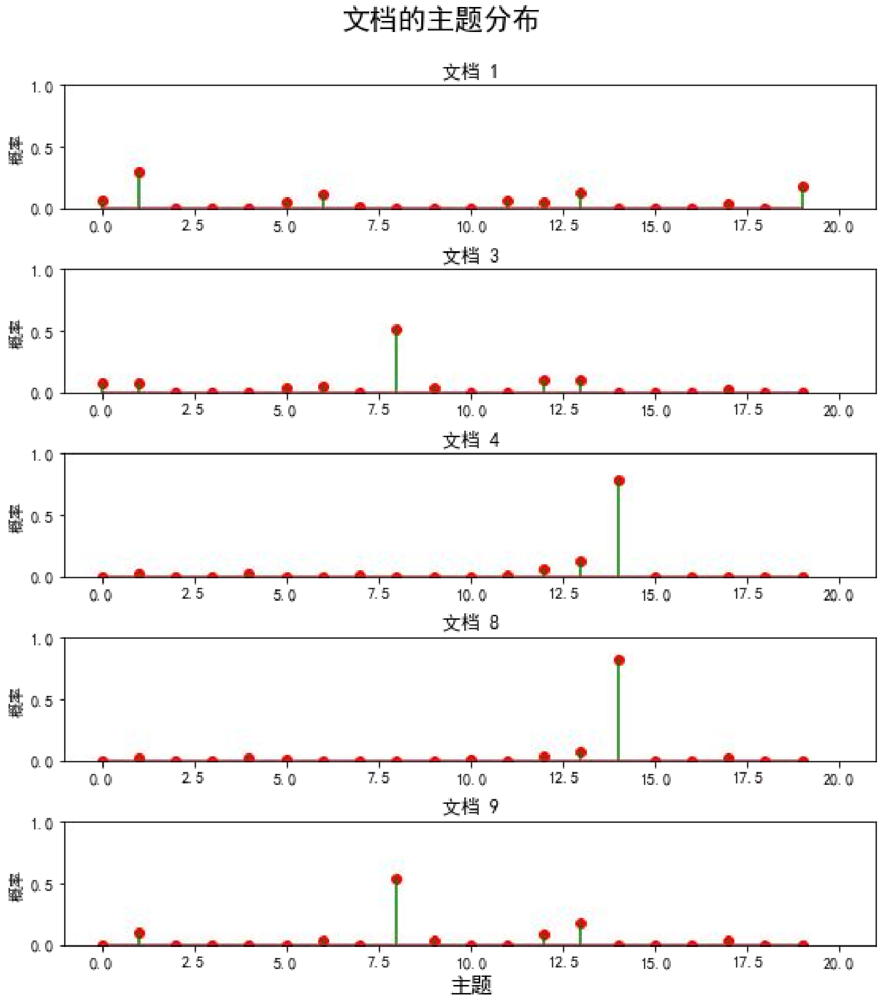
LDA参数学习-Gibbs采样预测流程

- 1. 对应当前文档的每一个词，随机的赋予一个主题编号z
- 2. 重新扫描当前文档，对于每一个词，利用Gibbs采样算法更新它的topic编号
- 3. 重复第二步的基于坐标轴轮换的Gibbs采样，直到Gibbs采样收敛
- 4. 统计文档中各个词的主题，得到该文档主题分布。

LDA主题模型案例

- 安装lda， eg: pip install lda





THANK YOU

上海育创网络科技有限公司