C1-11

Machine Learning
by Andrew Ng, Stanford Engineering

Xiaojie Zhou

szxjzhou@163.com

2016.9.15

第十二集：K-means算法

- 聚类与K-均值算法(Clustering and K-means Algorithm)
- 高斯混合(Mixture of Gaussians)
- 最大期望算法(Expectation Maximization (EM) Algorithm)

第十二集：K-means算法

- 聚类与K-均值算法
  - 之前讲到的分类算法属于监督学习，原因在于在分类中我们给每个训练集中的样本规定了类标
  - 而非监督学习研究的是另一种问题
    - 给定若干没有类标的点组成的数据集合，我们需要发现这些样本点的内在结构
    - 而聚类算法就是一种典型的非监督学习算法，它根据样本点的特征将样本点聚集成多个不同的类
      - 其中最有代表性的为K-均值算法，其流程如下
        - 1、随机初始化k个聚类中心(cluster centroids){μ1,μ2,...,μk}
        - 2、重复进行下列过程直至收敛（其中ci表示第i个样本点所属的类）

For every i, set

$$c^{(i)} := \arg\min_j ||x^{(i)} - \mu_j||^2.$$

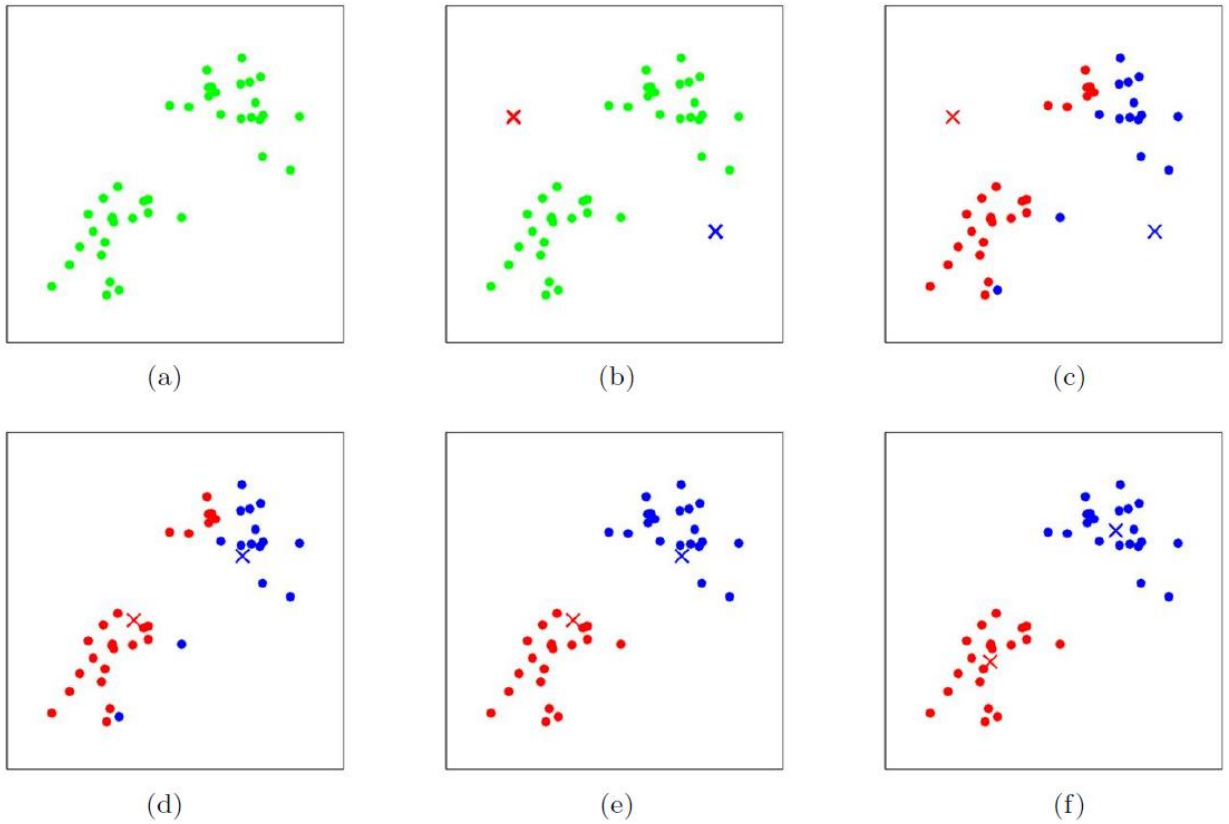
For each j, set

$$\mu_j := \frac{\sum_{i=1}^{m} 1\{c^{(i)} = j\} x^{(i)}}{\sum_{i=1}^{m} 1\{c^{(i)} = j\}}.$$

在第一步中每个样本点将根据到各聚类中心的距离决定各自所属的类别；而在第二步中将根据属于这个类的样本点的均值更新这个类的中心

第十二集：K-means算法

- 聚类与K-均值算法
  - K-均值算法的过程如下图所示



此图展示了K-均值算法的运行过程，x为聚类中心，红蓝实心点分别为两个类的样本点。在(a)中对于聚类中心进行了随机的初始化，然后经过了(b)-(e)的迭代最终得到了(f)的结果

而这个结果是一定会收敛的，原因在于下面这个式子（这也被称为失真函数(distortion function)，它表示的是每个样本点到所属聚类中心的距离之和），它在每次迭代的过程中只会减少而不会增大（因为整个过程相当于对该函数用坐标上升法对聚类中心μ和样本所属类c进行优化）。但与此同时也可以看到这个式子非凸，因此收敛的解并不一定是全局最优解（解会随着初始化聚类中心的位置和个数的变化而变化，对此的解决办法一般是尝试使用不同的聚类中心多运行几次算法）

$$J(c,\mu) = \sum_{i=1}^{m} ||x^{(i)} - \mu_{c^{(i)}}||^2$$

第十二集：K-means算法

- 高斯混合
  - K-均值算法最后会将所有的样本归到一个确定的类中，但这样往往显得过于绝对，我们更加希望的聚类结果是告诉我们有多大概率属于这个类
    - 这时候就可以采用密度估计(density estimation)的方法，用样本点训练出一个概率密度函数p(x)，然后对于测试样本x′通过计算p(x′)即可反映出该测试样本属于这一分布所代表的聚类的概率
    - 这里面一个重要的问题在于数据可能不是单一高斯分布、单一泊松分布这些单一标准的分布的。而且由于数据量太大，即使属于这些分布，我们也很难判断出来
      - 这时候就可以使用一种称为高斯混合的模型(mixture of Gaussians model)，这个模型顾名思义就是众多个高斯分布的叠加，而每个高斯可以看成是对于某一个聚类的拟合
        - 这时如果我们是知道类标的，那么我们可以将各个类的数据用极大似然分别计算出对应的高斯分布，然后再进行叠加
        - 但不幸的是我们并不知道类标，因此我们没办法直接计算出对应的高斯分布

第十二集：K-means算法

- 高斯混合
  - 既然只要我们知道了类标以后就可以用极大似然求出每个类对应的高斯分布，现在的问题在于就是不知道每个样本所属的类标是什么
    - 因此我们不妨假设一个随机变量z来表示样本的类标
      - 对于由m个样本组成的训练集{x1, x2, ..., xm}，存在一个隐随机变量(latent random variable)集合{z1, z2, ..., zm}表示样本的类标
        - 隐变量的意思是我们知道这个变量是存在的，但是现在还不知道是什么
      - 与此同时假设样本中要被聚到k个分布中，因此对于样本x所属的类标z来说，服从向量φ的多项式分布，并要求向量φ满足下式要求

$$\phi_j \geq 0, \sum_{j=1}^{k} \phi_j = 1$$

这里面φj指概率p(z=j)

第十二集：K-means算法

- 高斯混合
  - 这时候类比之前高斯判别分析的方法，写出似然方程

$$\ell(\phi, \mu, \Sigma) = \sum_{i=1}^{m} \log p(x^{(i)}; \phi, \mu, \Sigma)$$

这个似然方程和高斯判别分析的略有不同，原因在于在高斯判别分析中我们不仅给出了样本x，同时还给出了x对应的类标y，因此要极大化x,y的联合概率p(x,y)，即极大化对应类标y的样本x的抽取可能性。但在这里，只给出了样本x，没有给出类标y，因此只需极大化p(x)，即极大化样本x的抽取可能性

  - 这时当样本的类标z是可知的时，即可直接类比高斯判别分析写出类似的结论

$$\ell(\phi, \mu, \Sigma) = \sum_{i=1}^{m} \log p(x^{(i)}|z^{(i)}; \mu, \Sigma) + \log p(z^{(i)}; \phi)$$

这里面将可知的类标z引入了进来，从而我们极大化的目标也要相应从极大化p(x)变回了极大化x,z的联合概率p(x,z)=p(x|z)p(z)

  - 其中：

$$\mathrm{p(zi;\phi)} = \phi 1^{1\{zi=1\}} \phi 2^{1\{z2=1\}} \ldots \phi \mathrm{k} - 1^{1\{zk-1=1\}} (1 - \sum_{c=1}^{k-1} \phi \mathrm{c})^{1-\sum_{c=1}^{k-1} 1\{\mathrm{zi=c}\}}$$

$$\mathrm{p(x|z;\mu,\Sigma)} = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\right)$$

其中p(z)满足多项式分布，而p(x|z)满足混合高斯分布

第十二集：K-means算法

- 高斯混合
  - 接下来是对于参数的求解
    - 首先是对于参数φ的求解

由于

$$\ell(\phi,\mu,\Sigma)=\sum_{i=1}^{m}\log p(x^{(i)}|z^{(i)};\mu,\Sigma)+\log p(z^{(i)};\phi)$$

不难看出这里面与φ有关的只有右边一项，因此求偏微分时左边一项变为系数。因此令φ的偏微分为 0，相当于令右边一项的偏微分为 0，而不用考虑左边一项，因此有：

$$\nabla_{\varphi j}\ell(\varphi)=\nabla_{\varphi j}\sum_{i=1}^{m}\left(1\{zi=1\}\log\varphi 1+\cdots+1\{zi=k-1\}\log\varphi k-1+\left[1-\sum_{c=1}^{k-1}1\{\mathrm{zi}=\mathrm{c}\}\right]\log\left(1-\sum_{c=1}^{k-1}\phi\mathrm{c}\right)\right)$$

$$=\sum_{i=1}^{m}\left(1\{zi=j\}\frac{1}{\varphi j}-\left[1-\sum_{c=1}^{k-1}1\{\mathrm{zi}=\mathrm{c}\}\right]\frac{1}{1-\sum_{c=1}^{k-1}\phi\mathrm{c}}\right)$$

令$\nabla_{\varphi j}\ell(\varphi)=0$有：

$$0=\sum_{i=1}^{m}\left(1\{zi=j\}\frac{1}{\varphi j}-\left[1-\sum_{c=1}^{k-1}1\{\mathrm{zi}=\mathrm{c}\}\right]\frac{1}{1-\sum_{c=1}^{k-1}\phi\mathrm{c}}\right)$$

$$=\sum_{i=1}^{m}\left(1\{zi=j\}\left(1-\sum_{c=1}^{k-1}\phi\mathrm{c}\right)+\varphi j\left[\sum_{c=1}^{k-1}1\{\mathrm{zi}=\mathrm{c}\}-1\right]\right)=\sum_{i=1}^{m}\left(1\{zi=j\}-\varphi j\right)$$

由此可得：

$$\phi_j=\frac{1}{m}\sum_{i=1}^{m}1\{z^{(i)}=j\}$$

- 而对于系数μ和系数Σ的求解和高斯判别分析中的求解方法完全相同，因此不再赘述
  - 详情可见Machine Learning C1-4的15页

第十二集：K-means算法

- 高斯混合
  - 由此可得最终的参数

$$
\begin{aligned}
\phi_j &= \frac{1}{m}\sum_{i=1}^{m} 1\{z^{(i)} = j\}, \\
\mu_j &= \frac{\sum_{i=1}^{m} 1\{z^{(i)} = j\} x^{(i)}}{\sum_{i=1}^{m} 1\{z^{(i)} = j\}}, \\
\Sigma_j &= \frac{\sum_{i=1}^{m} 1\{z^{(i)} = j\} (x^{(i)} - \mu_j)(x^{(i)} - \mu_j)^T}{\sum_{i=1}^{m} 1\{z^{(i)} = j\}}
\end{aligned}
$$

Φ的含义是所有类标为j的样本个数占样本总数的比值；均值μj的含义是所有类标为j的样本x的和除以所有类标为j的样本总数；方差Σj的含义是所有类标为j的样本减去对应均值的平方除以样本总数

第十二集：K-means算法

- 最大期望算法
  - 但现在的问题是样本的类标z是不可知的
    - 也就是说我们现在只知道样本x的值，并已知类标z满足多项式分布
      - 我们将上面的内容具体化，类比一个实际生活中的例子：假设抽样统计得到了一部分同学的身高样本构成了训练集（这里面同时混有男生身高，也混有女生身高），现在给出一个测试集中的身高，问这个身高是男生的还是女生的？
        - 如果这时候给出了训练集里面所有身高x的类标z，那么很简单，就像高斯判别分析做的那样，将男生的身高单独拿出来计算对应的高斯分布（中心极限定理保证了认为该分布是高斯分布是有意义的），然后再将女生的身高单独拿出来计算对应的高斯分布，再计算比较该测试集中的身高在两个分布下的概率
        - 如果没有给出类标我们无法进行计算对应的分布，但与此同时我们也发现，如果我们知道了男生身高和女生身高的分布是可以估计出一个新的身高属于男生的概率和属于女生的概率
        - 由此这个问题的解决关键在于解决下面两个相互依赖的问题
          - 1、给定一个身高，用男生身高和女生身高的分布计算某个身高属于男生的概率和属于女生的概率（从而得到类标）
          - 2、通过给定的类标用极大似然法计算男生身高和女生身高的分布（从而得到分布）

第十二集：K-means算法

- 最大期望算法
  - 这时回忆K-均值聚类中的做法
    - 在K-均值聚类中我们也有同样的问题，一开始我们既不知道数据属于哪个类，也不知道各个类的中心是什么。这时采取的做法是，先随机定下各个类的中心，然后根据数据到各个类的距离判断所属的类别，之后再重新计算各个类的中心...如此一直迭代下去，即可得到最终收敛的结果
  - 对此我们将K-均值聚类的思想类比过来不难得到下面的解决方案
    - 首先初始化随机指定分布，从而可以用这个分布计算数据属于各个类的概率

$$w_j^{(i)} := p(z^{(i)} = j | x^{(i)}; \phi, \mu, \Sigma) \quad p(z^{(i)} = j | x^{(i)}; \phi, \mu, \Sigma) = \frac{p(x^{(i)} | z^{(i)} = j; \mu, \Sigma) p(z^{(i)} = j; \phi)}{\sum_{l=1}^{k} p(x^{(i)} | z^{(i)} = l; \mu, \Sigma) p(z^{(i)} = l; \phi)}$$

    - 其次通过给定的类标用极大似然法再重新计算分布

$$\begin{aligned}
\phi_j &:= \frac{1}{m} \sum_{i=1}^{m} w_j^{(i)}, \\
\mu_j &:= \frac{\sum_{i=1}^{m} w_j^{(i)} x^{(i)}}{\sum_{i=1}^{m} w_j^{(i)}}, \\
\Sigma_j &:= \frac{\sum_{i=1}^{m} w_j^{(i)} (x^{(i)} - \mu_j)(x^{(i)} - \mu_j)^T}{\sum_{i=1}^{m} w_j^{(i)}}
\end{aligned}$$

而这就构成了最大期望算法的核心思想（但请注意上面的这些公式只是最大期望算法应用在高斯混合问题中的特例，并不是最大期望算法的一般形式），计算ωj的过程被称为E步(E-step)，在这一步中将采用贝叶斯公式对后验概率进行计算，其中p(x|z)由高斯分布所定义，p(z)由多项式分布所定义；计算分布权重φ和分布参数μ,Σ的过程称为M步(M-step)，在这过程中将采用极大似然法对于上述参数进行更新

第十二集：K-means算法

- 最大期望算法
  - 在介绍EM算法的一般形式前先介绍一个重要的基本不等式
    - 基本不等式 $f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta) f(y)$

当不等式中≤变成＜时，对应的函数f是严格凸的

称为Jensen不等式，如果在此基础上再要求f的定义域是凸集且0≤θ≤1，,那么这个不等式就构成了凸函数的定义式

- 因此，对于凸函数来说必定满足Jensen不等式，而除此以外还有一个更广泛的结论：如果θ1,...,θk≥0且θ1+...+θk=1，下面不等式成立（用数学归纳法进行证明）

$$f(\theta_1 x_1 + \cdots + \theta_k x_k) \leq \theta_1 f(x_1) + \cdots + \theta_k f(x_k)$$

- 而这又可以进一步扩展到无穷项和、积分以及期望上

$$p(x) \geq 0 \text{ on } S \subseteq \mathbf{dom}\, f,\ \int_S p(x)\,dx = 1 \longrightarrow f\left(\int_S p(x)x\,dx\right) \leq \int_S f(x)p(x)\,dx,$$

如果f是一个凹函数也有类似结论，只不过不等号方向相反

事件x在f的定义域上发生的概率为1 $\longrightarrow$ $f(\mathbf{E}\,x) \leq \mathbf{E}\,f(x)$

第十二集：K-means算法

- 最大期望算法
  - Jensen不等式的证明

$$f(\theta_1 x_1 + \cdots + \theta_k x_k) \le \theta_1 f(x_1) + \cdots + \theta_k f(x_k)$$

下面将用数学归纳法对于$f(\theta 1x1 + \cdots + \theta kxk) \le \theta 1f(x1) + \cdots + \theta kf(xk)$进行证明

1、当 k=2 时：对于凸函数一定有

$$f(\theta 1x1 + \theta 2x2) = f(\theta 1x1 + (1-\theta 1)x2) \le \theta 1f(x1) + (1-\theta 1)f(x2) = \theta 1f(x1) + \theta 2f(x2)$$

2、假设 k=n 时成立，即：

$$f(\theta 1x1 + \cdots + \theta nxn) \le \theta 1f(x1) + \cdots + \theta nf(xn)$$

现在要证 k=n+1 时成立：

$$f(\theta 1x1 + \cdots + \theta n+1xn+1) = f\left(\theta 1x1 + (1-\theta 1)\sum_{i=2}^{n+1}\frac{\theta i}{1-\theta 1}xi\right) \le \theta 1f(x1) + (1-\theta 1)f(\sum_{i=2}^{n+1}\frac{\theta i}{1-\theta 1}xi)$$

由于：

$$\sum_{i=2}^{n+1}\frac{\theta i}{1-\theta 1} = 1$$

因此根据假设有：

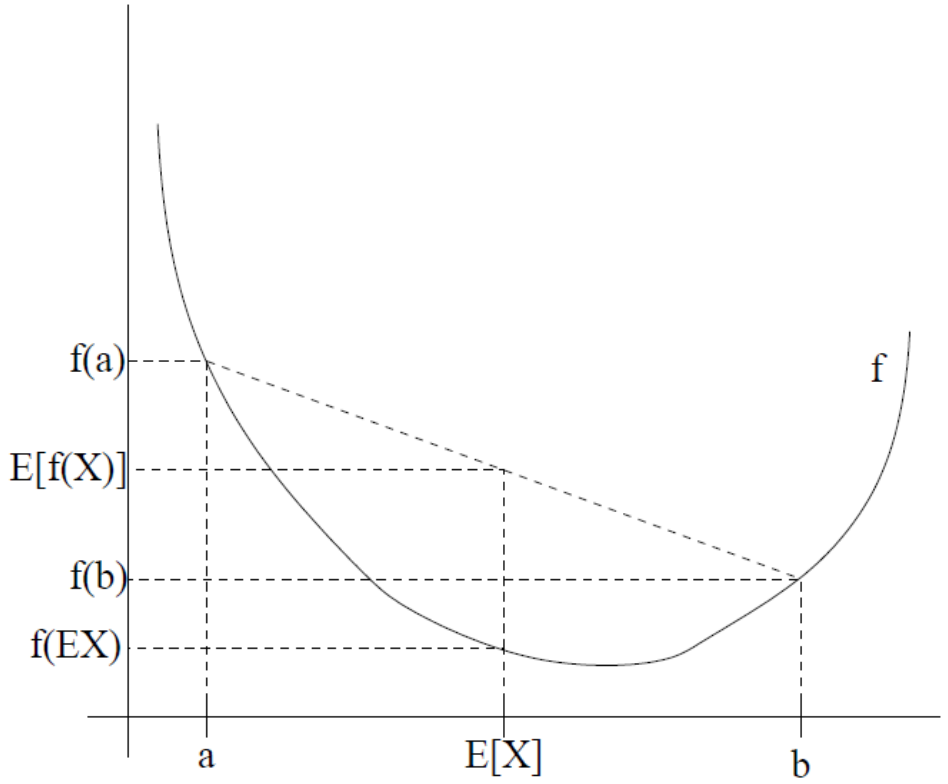
$$(1-\theta 1)f\left(\sum_{i=2}^{n+1}\frac{\theta i}{1-\theta 1}xi\right) \le (1-\theta 1)\sum_{i=2}^{n+1}\frac{\theta i}{1-\theta 1}f(xi) = \sum_{i=2}^{n+1}\theta i f(xi)$$

将其代入原式有：

$$f(\theta 1x1 + \cdots + \theta n+1xn+1) \le \theta 1f(x1) + \sum_{i=2}^{n+1}\theta i f(xi) = \theta 1f(x1) + \cdots + \theta n+1f(xn+1)$$

第十二集：K-means算法

- 最大期望算法
  - 下图展示了Jensen不等式对于凸函数在期望上的推广



这幅图像很好地展示了对于一个凸函数f来说f(EX)≤E[f(X)]

而与此同时我们也可以进一步得到对于一个严格凸的凸函数f来说如果f(EX)=E[f(X)]，当且仅当EX=X，即X是一个常数

第十二集：K-means算法

- 最大期望算法
  - 接下来将在Jensen不等式的基础上推导EM算法
    - 首先先列出极大似然表达式（再一次注意在这个问题中只给出了样本x，没有给出类标y，因此只需极大化样本x出现的概率p(x)，即极大化样本x的抽取可能性）

$$\begin{aligned}\ell(\theta) &= \sum_{i=1}^{m} \log p(x;\theta) \\ &= \sum_{i=1}^{m} \log \sum_{z} p(x,z;\theta)\end{aligned}$$

在这里我们引入一个隐随机变量z，而直接根据边缘概率和联合概率的关系，可得$\mathrm{p(x;\theta)} = \sum_z p(x,z;\theta)$

第十二集：K-means算法

- 最大期望算法

$$\ell(\theta) = \sum_{i=1}^{m} \log p(x;\theta)$$
$$= \sum_{i=1}^{m} \log \sum_{z} p(x,z;\theta)$$

$$\sum_i \log p(x^{(i)};\theta) = \sum_i \log \sum_{z^{(i)}} p(x^{(i)},z^{(i)};\theta) \quad (1)$$
$$= \sum_i \log \sum_{z^{(i)}} Q_i(z^{(i)}) \frac{p(x^{(i)},z^{(i)};\theta)}{Q_i(z^{(i)})} \quad (2)$$
$$\geq \sum_i \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)},z^{(i)};\theta)}{Q_i(z^{(i)})} \quad (3)$$

$$\sum_{z^{(i)}} Q_i(z^{(i)}) \left[\frac{p(x^{(i)},z^{(i)};\theta)}{Q_i(z^{(i)})} \right]$$

通过左上的推导易得(1)式，但对于(1)式来说一大问题在于zi的分布我们是不知道的，这就对应到了EM算法中的一个核心问题：在不知道每个样本的类标的时候，我们没办法通过极大似然法计算得到样本x所属分布的参数。但幸运的是我们可以保证这样的分布是存在且有物理意义的（z的物理意义在于它显示了对应样本x的类标情况，而对于每个样本xi来说它们对应的类标zi分布很可能是不同的。就拿身高的例子来说200cm有可能90%属于男生的分布10%属于女生的分布，而150cm有可能30%属于男生的分布70%属于女生的分布）。因此在这里采取的解决办法是，将这个类标z的分布假设出来，记为Q.根据上面的讨论不难得出，对于每个类标zi来说都有对应的分布为Qi(zi).由于Q是一个分布，因此要求Q满足$\sum_z Qi(z) = 1, Qi(z) \geq 0$，由此可以通过乘上$Qi(z)$，除以一个$Qi(z)$的方法将(1)式等价转换为(2)式

通过乘除同一个$Qi(z)$的方法将(1)式写成(2)式看上去毫无意义，但实际上这样是在想办法往Jensen不等式上面靠拢（后面会讲到利用Jensen不等式的好处）。我们将(2)式的log后面的部分单独拿出来看（如左下角所示），发现中括号里面可以看成是关于随机变量z的函数（中括号部分的分子中的参数只有隐随机变量z，x为样本值是确定的；θ为参数，虽然我们暂时不知道这个参数的值是什么但是用统计学派的观点看它也是一个确定值），而Q是z的分布，因此整个式子相当于求中括号部分的期望。由此利用Jensen不等式可以进一步得到(3)式的表达形式（log是一个严格凹函数）

第十二集：K-means算法

- 最大期望算法

$$f\left(\mathrm{E}_{z^{(i)}\sim Q_i}\left[\frac{p(x^{(i)},z^{(i)};\theta)}{Q_i(z^{(i)})}\right]\right) \geq \mathrm{E}_{z^{(i)}\sim Q_i}\left[f\left(\frac{p(x^{(i)},z^{(i)};\theta)}{Q_i(z^{(i)})}\right)\right]$$

$$\frac{p(x^{(i)},z^{(i)};\theta)}{Q_i(z^{(i)})} = c$$

$$\begin{aligned} Q_i(z^{(i)}) &= \frac{p(x^{(i)},z^{(i)};\theta)}{\sum_z p(x^{(i)},z;\theta)} \\ &= \frac{p(x^{(i)},z^{(i)};\theta)}{p(x^{(i)};\theta)} \\ &= p(z^{(i)}|x^{(i)};\theta) \end{aligned}$$

通过之前的推导得到了如左上的不等式，而这个式子给出了原似然方程的下界，且这个下界是和类标z对应的分布Q有关的。对于我们来说，自然希望有一个更加紧的下界（由于我们的目标是极大化似然方程，因此下界越大越好）。不难发现，这个最大的下界在等于时取得。根据之前Jensen不等式的性质，对于一个严格凸函数f来说如果f(EX)=E[f(X)]，当且仅当EX=X，即X是一个常数，由此可以进一步得到如左所示的结论（这也就说明了不论zi的值是什么，p(xi,zi;θ)/Qi(zi)的值均相同，Qi(zi)总是正比于p(xi,zi;θ)）

由于Q是一个分布，因此$\sum_z Qi(z) = 1$，由此进一步可以得到如左下的原似然方程最紧下界对应的类标z的分布Q的表达形式。得到分布Q的形式后即可代回原式继续求解如下的极大似然问题

$$\sum_i \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)},z^{(i)};\theta)}{Q_i(z^{(i)})}$$

第十二集：K-means算法

- 最大期望算法
  - 由此得到了最大期望算法一般情况下的表达形式
    - 重复迭代下面两步直至收敛
      - 1、E-step：更新每个样本x对应类标z的分布Q
        - 此时已知参数θ，用贝叶斯公式即可得到新的分布Q

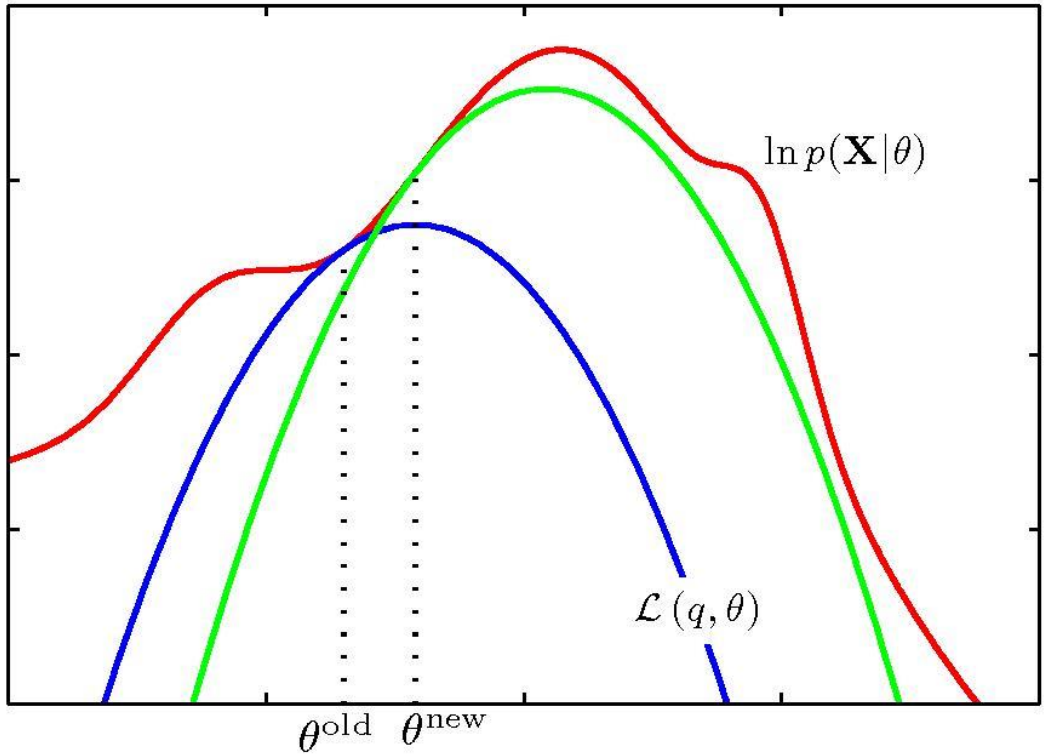
$$Q_i(z^{(i)}) := p(z^{(i)}|x^{(i)};\theta)$$

      - 2、M-step：更新参数θ的取值
        - 此时已知类标z的分布Q，用极大似然法即可得到新的参数θ

$$\theta := \arg\max_{\theta} \sum_i \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})}$$

第十二集：K-means算法

- 最大期望算法
  - 下面通过图片的方式展示EM算法的具体过程



这里的红色曲线为我们希望极大似然的式子（即似然方程）$\ell(\theta)=\sum_{i=1}^{m}\log p(x;\theta)$.在E-step中先随机初始化参数θ（也就是一开始假设了分布），求解类标z所属的分布Q，从而得到了图中的蓝色曲线。根据之前所述，此时我们可以求得原似然方程的下界，该下界是最优的（从图像中可以看出蓝色曲线在初始化的参数θ处和原似然方程在该处的取值相同）

而在M-step中则在已知类标z所属的分布Q的情况下优化如图中的蓝色曲线所示的下界函数$\sum_i\sum_{zi}Qi(zi)\log{}^{p(xi,zi;\theta)}/_{Qi(zi)}$，从而求得新的分布参数θ，这一新的分布参数可以得到下界函数的最优值。然后再将新的分布参数θ代回M-step中，即可得到如图中绿色曲线所示的新的类标z所属的分布Q

但强调一点，EM算法的极大似然的目标$\ell(\theta)=\sum_{i=1}^{m}\log p(x;\theta)$和下界函数$\sum_i\sum_{zi}Qi(zi)\log{}^{p(xi,zi;\theta)}/_{Qi(zi)}$并不保证一定是凹函数，因此EM算法也可能得到局部最优解

第十二集：K-means算法

- 最大期望算法
  - 下面将对最大期望算法收敛性进行证明，最大期望算法收敛的关键在于证明后一次迭代的参数θ比前一次迭代的参数θ对于似然方程只增不减

$$\ell(\theta^{(t)}) \leq \ell(\theta^{(t+1)})$$

$$\begin{aligned}
\sum_i \log p(x^{(i)};\theta) &= \sum_i \log \sum_{z^{(i)}} p(x^{(i)}, z^{(i)};\theta) && (1)\\
&= \sum_i \log \sum_{z^{(i)}} Q_i(z^{(i)}) \frac{p(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})} && (2)\\
&\geq \sum_i \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})} && (3)
\end{aligned}$$

左上为对于似然方程的推导，并通过Jensen不等式得到了似然方程在给定参数θ的情况下的下界。此时，我们希望下界是最优的，因此令不等号直接变成了等号，从而在第t次迭代的E-step中得到了左下的等式

$$\ell(\theta^{(t)}) = \sum_i \sum_{z^{(i)}} Q_i^{(t)}(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)};\theta^{(t)})}{Q_i^{(t)}(z^{(i)})}.$$

第十二集：K-means算法

- 最大期望算法
  - 最大期望算法的收敛性证明

$$\ell(\theta^{(t+1)}) \geq \sum_i \sum_{z^{(i)}} Q_i^{(t)}(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}; \theta^{(t+1)})}{Q_i^{(t)}(z^{(i)})} \quad (4)$$

$$\geq \sum_i \sum_{z^{(i)}} Q_i^{(t)}(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}; \theta^{(t)})}{Q_i^{(t)}(z^{(i)})} \quad (5)$$

$$= \ell(\theta^{(t)}) \quad (6)$$

$$J(Q, \theta) = \sum_i \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}; \theta)}{Q_i(z^{(i)})}$$

根据之前所述，我们知道$\sum_i \sum_{zi} Qi(zi) \log {}^{p(xi,zi;\theta)}/_{Qi(zi)}$只是一个下界函数，因此它的值不会大于原似然方程$\ell(\theta)$，由此可得如(4)式所示的不等式；而在第t次迭代的M-step中，我们找到了下界函数$\sum_i \sum_{zi} Qi(zi) \log {}^{p(xi,zi;\theta)}/_{Qi(zi)}$取得极大值时对应的参数θ(t+1)，因此参数θ(t+1)在该下界函数$\sum_i \sum_{zi} Qi(zi) \log {}^{p(xi,zi;\theta)}/_{Qi(zi)}$上的值不小于其它参数在该下界函数上的值，由此可得如(5)式所示的不等式；而又根据之前所述，在第t次迭代的E-step中通过Jensen不等式得到了似然方程在给定参数θ的情况下的最优下界，这也就意味着该下界函数$\sum_i \sum_{zi} Qi(zi) \log {}^{p(xi,zi;\theta)}/_{Qi(zi)}$在给定参数θ下与原似然方程$\ell(\theta)$的值相等，由此可得如(6)式所示的不等式，从而保证了后一次迭代的参数θ比前一次迭代的参数θ对于似然方程$\ell(\theta)$只增不减，即保证了算法的收敛性

实际上最大期望算法的算法过程还可看作是如左下所示的函数J(Q,θ)的坐标上升法的优化过程（EM算法的E-step可看作是固定参数θ，找到最优的参数Q；而M-step可看作是固定参数Q，找到最优的参数θ）