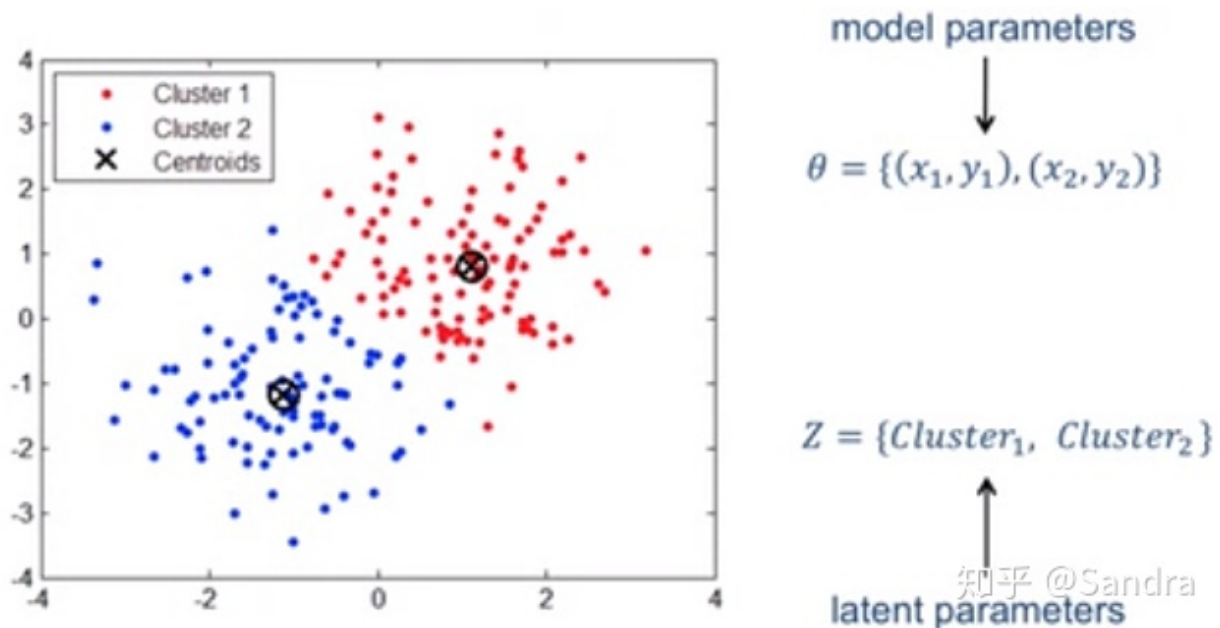


EM (期望最大化) 算法:

引入: (K-means)



z描述的是每一个点到底属于哪一个簇 (隐含参数, 辅助解决问题), 模型参数是要求的Z是未知的, 但是如果知道了模型参数 (中心点坐标) 就知道了Z; 已知Z就可以求得中心点坐标--> 可以假设中心点坐标, 就可以根据距离远近分类得到Z, 再利用Z的分类反推中心点坐标。

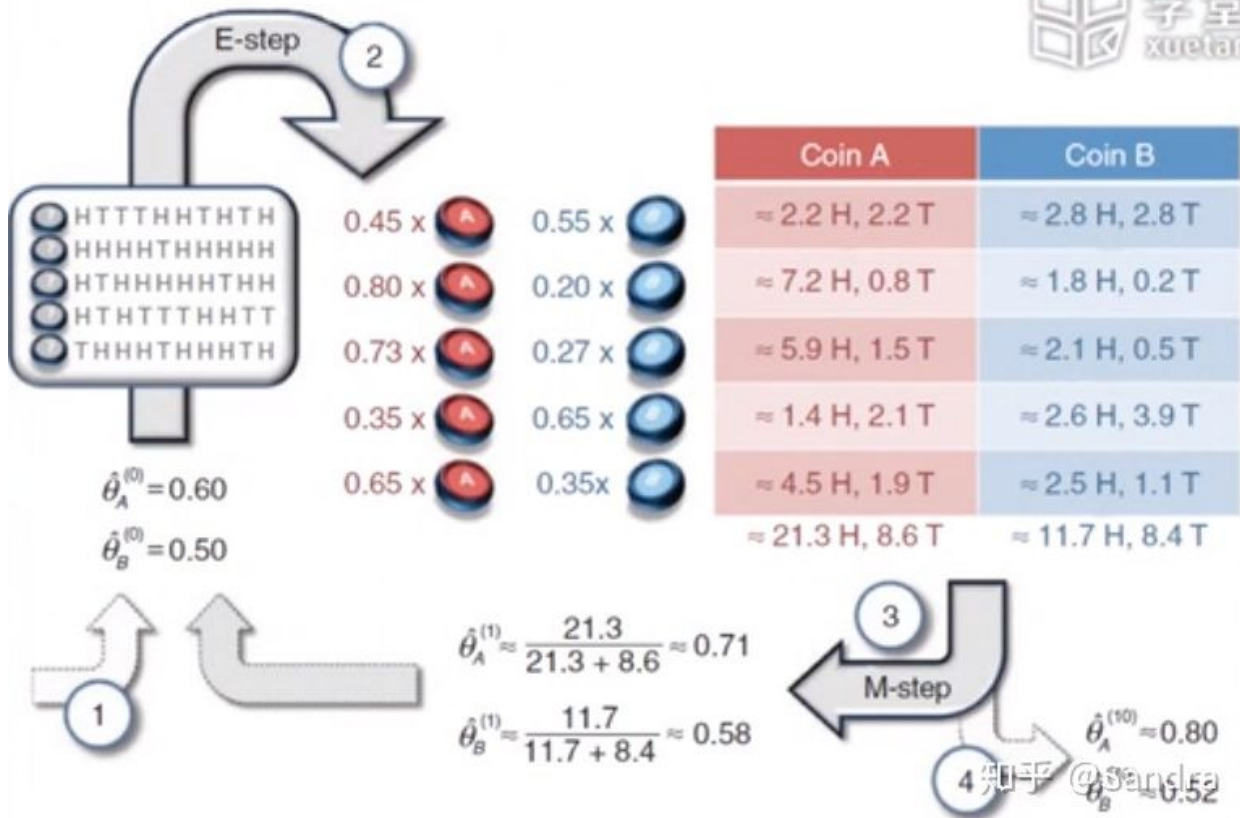
抛硬币 (不均匀) --> EM的基本工作流程:

想知道不同硬币A, B头向上的概率, 利用最大似然估计:



不断地抛硬币, 记录  
给两个硬币, 继续抛硬币, 但不知道哪个是A, B

## b Expectation maximization



### EM算法

首先猜测不同硬币头向上的概率，在猜测硬币是A还是B（贝叶斯），将参测结果拆分到A与B，求出不同硬币头向上的概率（不断的迭代，收敛）

$$P(A|E) = \frac{P(E|A)P(A)}{P(E)}$$

类似于梯度下降，可能会有局部最优（与初始值的选取有关）。

高斯混合模型中的EM算法：

$m$ : the number of data points

$n$ : the number of mixture components

$z_{ij}$ : whether instance  $i$  is generated by the  $j$ th Gaussian

$$E[z_{ij}] = \frac{p(x = x_i | \mu = \mu_j) \alpha_j}{\sum_{k=1}^n p(x = x_i | \mu = \mu_k) \alpha_k} = \frac{e^{-\frac{1}{2\sigma^2}(x_i - \mu_j)^2} \alpha_j}{\sum_{k=1}^n e^{-\frac{1}{2\sigma^2}(x_i - \mu_k)^2} \alpha_k}$$
$$\mu_j \leftarrow \frac{\sum_{i=1}^m E[z_{ij}] x_i}{\sum_{i=1}^m E[z_{ij}]}$$
$$\alpha_j \leftarrow \frac{1}{m} \sum_{i=1}^m E[z_{ij}]$$

知乎 @Sandra

未知参数，高斯的均值与权重；利用中间参数 $Z$ 进行迭代收敛

先假设高斯的参数（

$\mu, \alpha$

）就是

$\theta$

，然后求 $Z$ 。

但由于高斯的分布特点，两侧的尾巴可以无限的延长，所以理论上空间中所有的点都可以被任意高斯生成，所以设定

$Z_{ij}$

——第 $i$ 个样本是有第 $j$ 个高斯生成的，是一个连续值，概率问题，可能性大小是不同的（

$E[Z_{ij}]$

其实就是相当于权重）。

当已知每个高斯的函数情况，就可以知道每个点由特定高斯生成出来的概率是多少，返回来对每一个样本进行加权，计算出每一个高斯的均值（加权求均值）。

每一个高斯特有的一个 $\alpha$ 值，是由每一个点是特定高斯生成的概率加和平均。

知道了高斯参数，就可以去更新 $Z$ 。