k-Means

无监督学习

只给出了数据集，却没有给出数据集的分类标签，需要按照某种算法，把数据集划分为某几类

k均值聚类算法，属于无监督学习算法

算法

1. Initialize **cluster centroids** $\mu_1, \mu_2, \ldots, \mu_k \in \mathbb{R}^n$ randomly.

2. Repeat until convergence: {

For every i, set

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

For each j, set

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

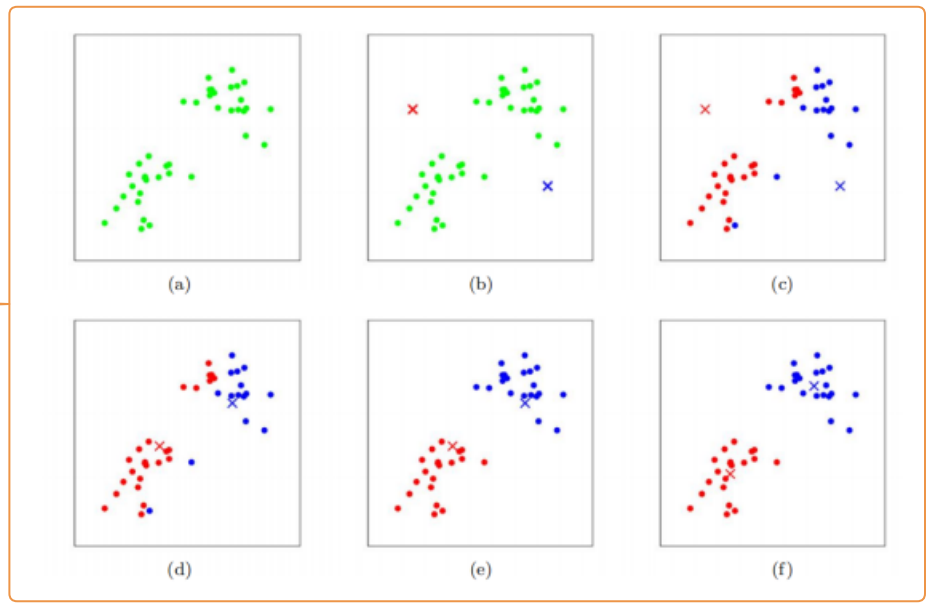
}

1.随机选取k个簇质心(cluster centroids)
2.循环以下过程直到收敛{
对每个质心，算出每个数据集与质心之间的误差（距离），然后将数据集归到误差最小的质心类别下；
数据集已归好类，然后再根据每个簇中的数据集，更新该簇的质心；

}

子主题

效果图



b图随机选取了两个质心，c图将数据集分好类，d图按照分类后的数据集重新计算簇的质心，e图再按照新质心对数据集分类，f图和e图相比无变化，说明达到收敛

算法收敛性

kmeans算法是局部收敛的

失真函数(distortion function)

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

表示某数据和质心的距离的平方和

可以看出kmeans算法的过程就是对J(c,u)函数的坐标下降的过程，固定c改变μ使J变小，再固定μ改变c使J变小，循环往复。

这样就保证了J是单调递减的，所以一定能达到收敛

失真函数 J，是一个非凸函数（non-convex function），所以对 J 进行坐标下降（coordinate descent）并不一定能够收敛到全局最小值（global minimum）。也就是说，k 均值聚类算法可能只是局部最优的（local optima）。

二分kmeans算法可以达到全局最优，这个算法简而言之就是每次选取“最优”的一个质心，将其分裂成两个，直至最后质心个数达到要求。
“最优”就是分裂这个质心，能够使误差降到最小，所以要找到“最优”的质心，就需要遍历当前所有质心进行分裂尝试，最后选取“最优”的那个，有点类似在数组中找最值。
详细算法见《机器学习实战》第10章