常用聚类算法及原理

一． 基本概念

聚类分析又称群分析，它是研究（样品或指标）分类问题的一种统计分析方法，同时也是数据挖掘的一个重要算法。
聚类（Cluster）分析是由若干模式（Pattern）组成的，通常，模式是一个度量（Measurement）的向量，或者是多维空间中的一个点。
聚类分析以相似性为基础，在一个聚类中的模式之间比不在同一聚类中的模式之间具有更多的相似性

二． 常用聚类算法

K 均值聚类算法(k-means)

K 均值聚类算法是先随机选取 K 个对象作为初始的聚类中心。然后计算每个对象与各个种子聚类中心之间的距离，把每个对象分配给距离它最近的聚类中心。聚类中心以及分配给它们的对象就代表一个聚类。每分配一个样本，聚类的聚类中心会根据聚类中现有的对象被重新计算。这个过程将不断重复直到满足某个终止条件。终止条件可以是没有（或最小数目）对象被重新分配给不同的聚类，没有（或最小数目）聚类中心再发生变化，误差平方和局部最小

聚类是一个将数据集中在某些方面相似的数据成员进行分类组织的过程，聚类就是一种发现这种内在结构的技术，聚类技术经常被称为无监督学习。

k 均值聚类是最著名的划分聚类算法，由于简洁和效率使得他成为所有聚类算法中最广泛使用的。给定一个数据点集合和需要的聚类数目 k，k 由用户指定，k 均值算法根据某个距离函数反复把数据分入 k 个聚类中

K-Means 算法是无监督的聚类算法，它实现起来比较简单，聚类效果也不错，因此应用很广泛。K-Means 算法有大量的变体，本文就从最传统的 K-Means 算法讲起，在其基础上讲述 K-Means 的优化变体方法。包括初始化优化 K-Means++，距离计算优化 elkan K-Means 算法和大数据情况下的优化 Mini Batch K-Means 算法

K-Means 算法的思想很简单，对于给定的样本集，按照样本之间的距离大小，将样本集划分为 K 个簇。让簇内的点尽量紧密的连在一起，而让簇间的距离尽量的大。

如果用数据表达式表示，假设簇划分为(C1,C2,...Ck)(C1,C2,...Ck)，则我们的目标是最小化平方误差 E:

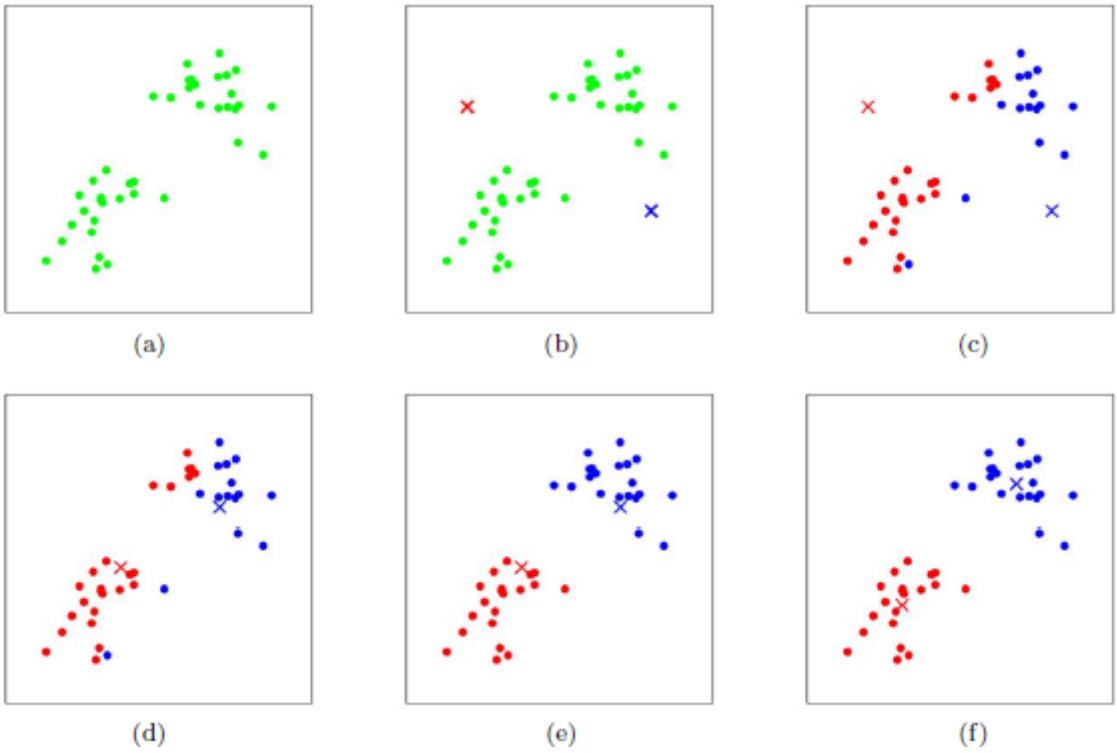
E=∑i=1k∑x∈Ci||x−μi||22E=∑i=1k∑x∈Ci||x−μi||22

其中 μiμi 是簇 CiCi 的均值向量，有时也称为质心，表达式为:

μi=1|Ci|∑x∈Cixμi=1|Ci|∑x∈Cix

如果我们想直接求上式的最小值并不容易，这是一个 NP 难的问题，因此只能采用启发式的迭代方法。

K-Means 采用的启发式方式很简单，用下面一组图就可以形象的描述。



上图 a 表达了初始的数据集，假设 k=2。在图 b 中，我们随机选择了两个 k 类所对应的类别质心，即图中的红色质心和蓝色质心，然后分别求样本中所有点到这两个质心的距离，并标记每个样本的类别为和该样本距离最小的质心的类别，如图 c 所示，经过计算样本和红色质心和蓝色质心的距离，我们得到了所有样本点的第一轮迭代后的类别。此时我们对我们当前标记为红色和蓝色的点分别求其新的质心，如图 4 所示，新的红色质心和蓝色质心的位置已经发生了变动。图 e 和图 f 重复了我们在图 c 和图 d 的过程，即将所有点的类别标记为距离最近的质心的类别并求新的质心。最终我们得到的两个类别如图 f。

当然在实际 K-Mean 算法中，我们一般会多次运行图 c 和图 d，才能达到最终的比较优的类别

GMM(高斯混合模型)算法

高斯混合模型（Gaussian Mixed Model）指的是多个高斯分布函数的线性组合，理论上 GMM 可以拟合出任意类型的分布，通常用于解决同一集合下的数据包含多个不同的分布的情况（或者是同一类分布但参数不一样，或者是不同类型的分布，比如正态分布和伯努利分布）。

如图 1，图中的点在我们看来明显分成两个聚类。这两个聚类中的点分别通过两个不同的正态分布随机生成而来。但是如果没有 GMM，那么只能用一个的二维高斯分布来描述图 1 中的数据。图 1 中的椭圆即为二倍标准差的正态分布椭圆。这显然不太合理，毕竟肉眼一看就觉得应该把它们分成两类。

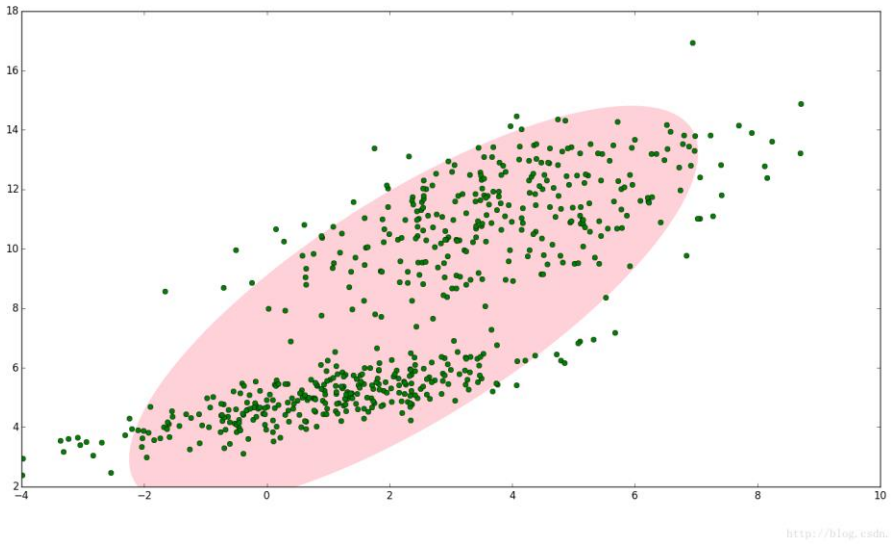


图 1

这时候就可以使用 GMM 了！如图 2，数据在平面上的空间分布和图 1 一样，这时使用两个二维高斯分布来描述图 2 中的数据，分别记为 N(μ1,Σ1)和 N(μ2,Σ2). 图中的两个椭圆分别是这两个高斯分布的二倍标准差椭圆。可以看到使用两个二维高斯分布来描述图中的数据显然更合理。实际上图中的两个聚类的中的点是通过两个不同的正态分布随机生成而来。如果将两个二维高斯分布 N(μ1,Σ1)和 N(μ2,Σ2)合成一个二维的分布，那么就可以用合成后的分布来描述图 2 中的所有点。最直观的方法就是对这两个二维高斯分布做线性组合，用线性组合后的分布来描述整个集合中的数据。这就是高斯混合模型（GMM）。

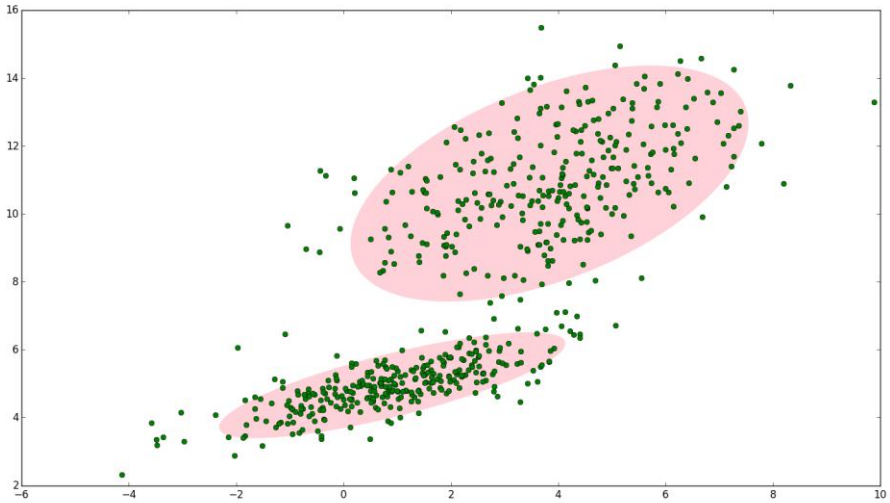


图 2

GMM 常用于聚类。如果要从 GMM 的分布中随机地取一个点的话，实际上可以分为两步：首先随机地在这 K 个 Component 之中选一个，每个 Component 被选中的概率实际上就是它的系数 πk ，选中 Component 之后，再单独地考虑从这个 Component 的分布中选取一个点就可以了——这里已经回到了普通的 Gaussian 分布，转化为已知的问题。

将 GMM 用于聚类时，假设数据服从混合高斯分布（Mixture Gaussian Distribution），那么只要根据数据推出 GMM 的概率分布来就可以了；然后 GMM 的 K 个 Component 实际上对应 K KK 个 cluster 。根据数据来推算概率密度通常被称作 density estimation 。特别地，当我已知（或假定）概率密度函数的形式，而要估计其中的参数的过程被称作『参数估计』。

例如图 2 的例子，很明显有两个聚类，可以定义 K=2 K=2K=2. 那么对应的 GMM 形式如下：

$$p(\boldsymbol{x}) = \pi_1 \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1) + \pi_2 \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$$

上式中未知的参数有六个：

$$(\pi_1, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1; \pi_2, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$$

之前提到 GMM 聚类时分为两步，第一步是随机地在这 K 个分量中选一个，每个分量被选中的概率即为混合系数 πk. 可以设定 π1=π2=0.5，表示每个分量被选中的概率是 0.5，即从中抽出一个点，这个点属于第一类的概率和第二类的概率各占一半。但实际应用中事先指定 πk 的值是很笨的做法，当问题一般化后，会出现一个问题：当从图 2 中的集合随机选取一个点，怎么知道这个点是来自 N(x｜μ1,Σ1)还是 N(x｜μ2,Σ2)呢？换言之怎么根据数

据自动确定 π1 和 π2 的值？这就是 GMM 参数估计的问题。要解决这个问题，可以使用 EM 算法。通过 EM 算法，我们可以迭代计算出 GMM 中的参数：(πk,xk,Σk)