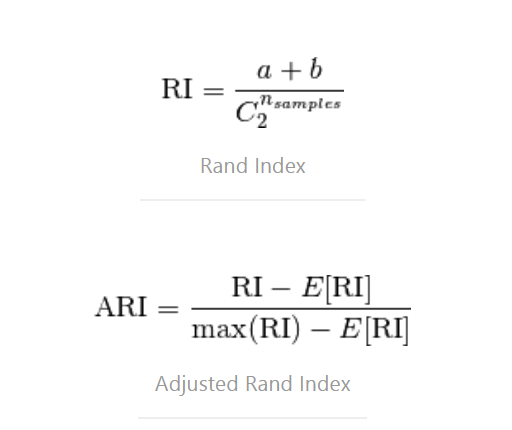
## 外部信息指标

**假设数据已经标注完毕，作为参考簇C，算法聚类形成簇D，样本总数为 m。下面以D作为Different的缩写，S为Same的缩写。那么，**

**a = 在C中为相同簇在D中为相同簇的样本数量（SS）**

**b = 在C中为相同簇在D中为不同簇的样本数量（SD）**

**调整后的 Rand 指数（Adjusted Rand Index，ARI）**



优点：

1. 范围有界，[-1, 1]，值越大聚类效果越好
2. 随机标签获得接近 0 的值。
3. 对簇的结果不需要做任何假设。换标签名，调整标签顺序没有影响

缺点：

1. 需要人工标注参考簇C，大数据几乎不可用

**Fowlkes-Mallows 指数（Fowlkes-Mallows index，FMI）**

优点与缺点和 ARI 基本一致，不过 FMI 的范围是 0 到 1。

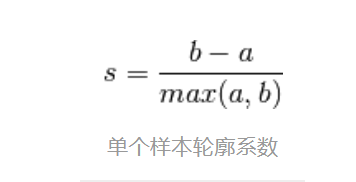
## 内部信息指标

外部信息评价指标的缺点均为人工标注在大规模数据下不可行，因此，我们需要借助其他的数字指标，来分析聚类的内部信息。聚类是为了让同一个簇内样本尽可能的近，不同簇的样本尽可能远（**类内高聚合**、**类间低耦合**）

**轮廓系数（Silhouette Coefficient ）**

轮廓系数由两个得分组成，一是样本与同一个簇中其他点之间的平均距离（记为a）。二是样本与下一个距离最近的簇中其他点的平均距离（记为b），根据公式计算出单个样本的轮廓系数，然后计算整组的轮廓系数平均值。对于不正确的聚类，分数的范围是-1，对于高度密集的聚类，分数的范围是+1。零附近的分数表示重叠的群集。

当群集密集且分隔良好时，分数会更高，这与群集的标准概念有关。



优点

簇紧凑且分离度高时分数更高。

缺点

凸簇比其他类型的簇分数更高。比如基于密度的 DBSCAN 算法在这种情况下表现更好。

这里的凸簇如果降维到一定程度，会接近于凸集。根据定义，对于集合内的每一对点，连接该对点的直线段上的每个点也在该集合内。凸集的边界总是凸曲线。

Sklearn实现：

轮廓系数（sklearn.metrics.silhouette\_score）、

**>>> import** **numpy** **as** **np**

**>>> from** **sklearn.cluster** **import** KMeans

**>>>** kmeans\_model = KMeans(n\_clusters=3, random\_state=1).fit(X)

**>>>** labels = kmeans\_model.labels\_

**>>>** metrics.silhouette\_score(X, labels, metric='euclidean')

0.55...

**Calinski-Harabaz 指数**

**DBI指数**

它的作用是评估K-means算法中k值的取值。

### 其他一些观点

应该嵌入到问题中进行评价

很多实际问题中，聚类仅仅是其中的一步，可以对比不聚类的情形（比如人为分割、随机分割数据集等等），所以这时候我们评价『聚类结果好坏』，其实是在评价『聚类是否能对最终结果有好的影响』

## 如何评价数据是否合适聚类

**Hopkins统计量**

霍普金斯统计量，评估给定数据集是否存在有意义的可聚类的非随机结构。如果一个数据集是有随机的均匀的点生成的，虽然也可以产生聚类结果，但该结果没有意义。聚类的前提需要数据是**非均匀分布**的。该值在区间[0, 1]之间，[0.01, 0.3]表示数据结构regularly spaced，该值为0.5时数据是均匀分布的，[0.7, 0.99]表示聚类趋势很强。

有开源实现