AP clustering and TS clustering

Huang Junjun

CONTENT

Affinity Propagation Clustering

Affinity Propagation Clustering（简称AP算法），2007发表在《Science》的“Clustering by Passing Messages Between Data Points”[1]。

基本思想：将全部样本看作网络的节点，然后通过网络中各条边的消息传递计算出各样本的聚类中心。聚类过程中，共有两种消息在各节点间传递，分别是吸引度(responsibility)和归属度(availability) 。AP算法通过迭代过程不断更新每一个点的吸引度和归属度值，直到产生m个高质量的Exemplar（相当于质心），同时将其余的数据点分配到相应的聚类中。

Affinity Propagation Clustering

Similarity（相似度）： 数据点i和点j的相似度记为s(i, j)，是指点j与点i的相似度。论文点之间相似度值采用的是平方距离的负值；相似度值越大说明点与点的距离越近(最大为0)。文章中指出，样本点之间相似度可以不必对称

Exemplar：聚类代表，类似于Kmeans中的质心，AP算法不需要事先指定聚类数目,它将所有的数据点作为潜在的聚类中心。

Preference：数据点i的参考值称为p(i)或s(i,i)，是指点i作为聚类中心的参考度，以S矩阵的对角线上的数值s (k, k)作为k点能否成为聚类中心的评判标准,p(i)值越大,点成为聚类中心的可能性也就越大。一般取s相似度值的中值。聚类的数量受到参考度p的影响,如果认为每个数据点都有可能作为聚类中心,那么p就应取相同的值。如果取输入的相似度的均值作为p的值,得到聚类数量是中等的。如果取最小值,得到的是簇数较少的聚类结果。

Affinity Propagation Clustering

吸引度(Responsibility：r(i,k)用来描述点k适合作为数据点i的聚类中心的程度

归属度(Availability：a(i,k)用来描述点i归属于点k作为其聚类中心的程度。

Damping factor(阻尼系数)：主要是起收敛作用。

注：实际应用中，最重要的两个参数（也是需要手动指定）是Preference和Damping factor。前者Preference决定了聚类数量的多少(正相关)，值越大聚类数量越多；后者Damping factor控制算法收敛效果。

Affinity Propagation Clustering

- AP算法流程：
- 1：初始化，将吸引度矩阵R和归属度矩阵A初始化为0矩阵；
- 2：更新吸引度矩阵R(i→k)

$$r_{t+1}(i,k)=\begin{cases} S(i,k)-\max_{j\neq k}\{a_t(i,j)+S(i,j)\}, & i\neq k \\ S(i,k)-\max_{j\neq k}\{S(i,j)\}, & i=k \end{cases}$$

- 3：更新归属度矩阵A(k→i)

$$a_{t+1}(i,k)=\begin{cases} \min\left\{0, r_{t+1}(k,k)+\sum_{j\notin\{i,k\}}\max\{r_{t+1}(j,k),0\}\right\}, & i\neq k \\ \sum_{j\neq k}\max\{r_{t+1}(j,k),0\}, & i=k \end{cases}$$

Affinity Propagation Clustering

- AP算法流程：
- 4：根据阻尼系数λ对两个R和A进行衰减(避免数值震荡,默认λ=0.5)

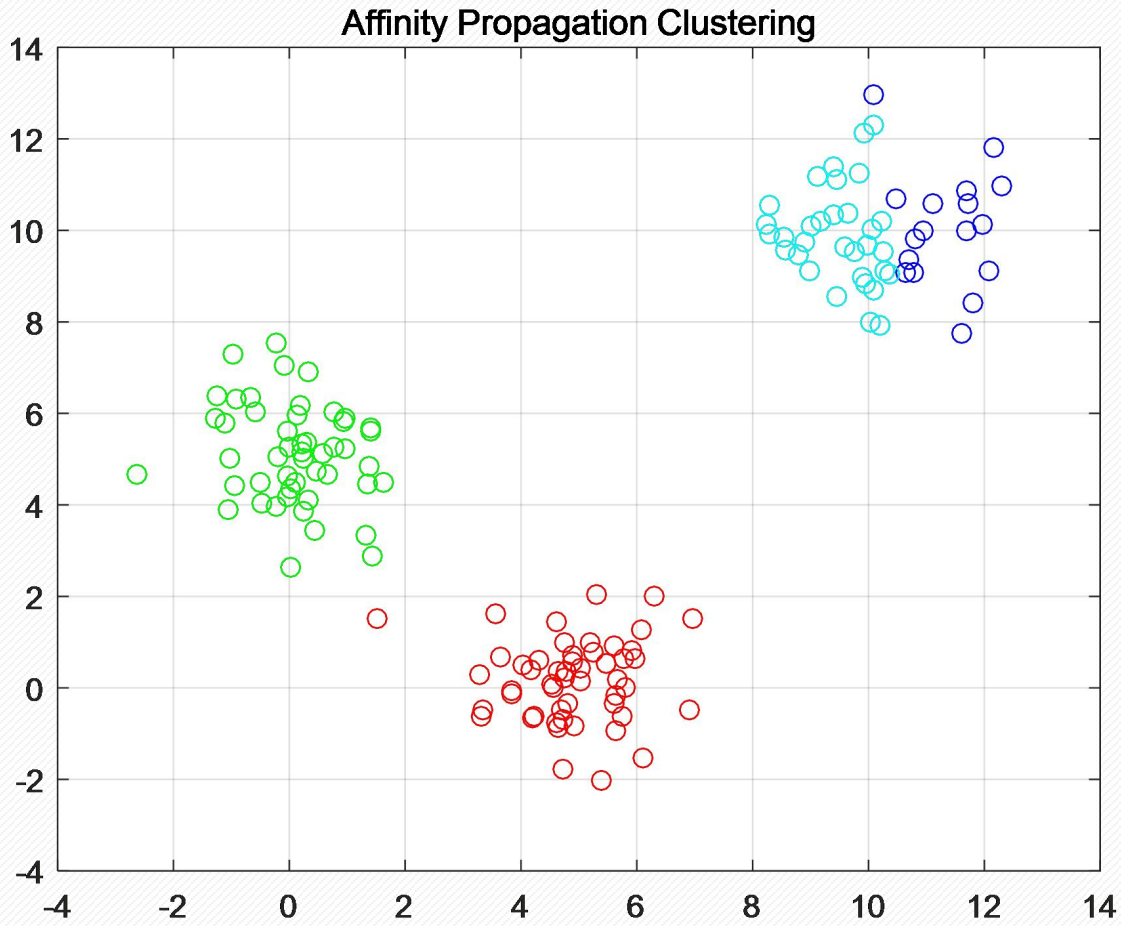
$$r_{t+1}(i,k) = \lambda * r_t(i,k) + (1-\lambda) * r_{t+1}(i,k)$$
$$a_{t+1}(i,k) = \lambda * a_t(i,k) + (1-\lambda) * a_{t+1}(i,k)$$

- 5: 重复步骤2-4直至R和A稳定或者达到最大迭代次数，算法结束
- 6:输出：计算Z=R+A,最终取Z每行i最大的值对应的索引k作为该样本i隶属的簇k。

Affinity Propagation Clustering

- AP算法特点：
- 1：无需预先指定聚类数目
- 2：簇中心选取来自样本中的某些点，而不是由多个数据点平均得到（如kmeans）
- 3：对距离相似矩阵S的对称性、三角不等性没要求
- 4：初始值不敏感。多次执行AP聚类算法，结果完全一样
- 5：以误差平方和来衡量算法优劣，AP聚类误差平方和相对较低
- 6：算法复杂度较高，为 $O(N^2*logN)$
- 7：需事先计算每对数据点之间的相似度S,且需一直保存在内存中
- 8：需要手动指定Preference和Damping factor，聚类结果受到二者影响

Case1:AP Clustering(K=4)

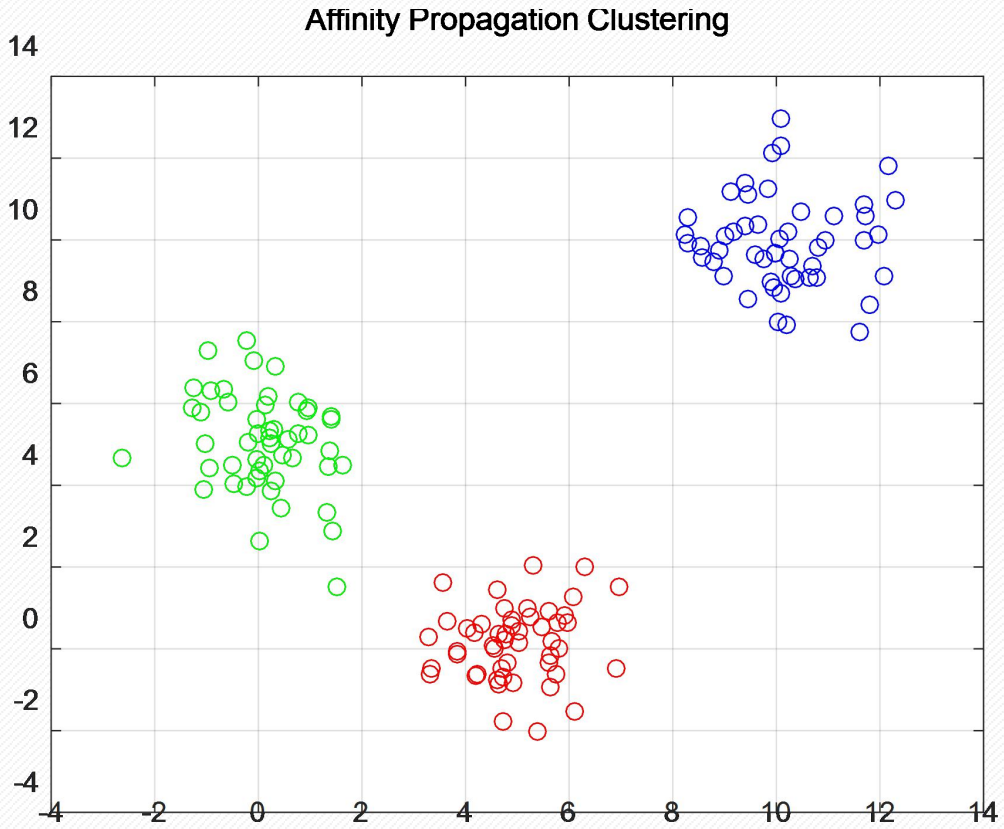


Preference=min(S);

Damping factor=0.5;

S为相似度矩阵

Case1:AP Clustering(K=3)

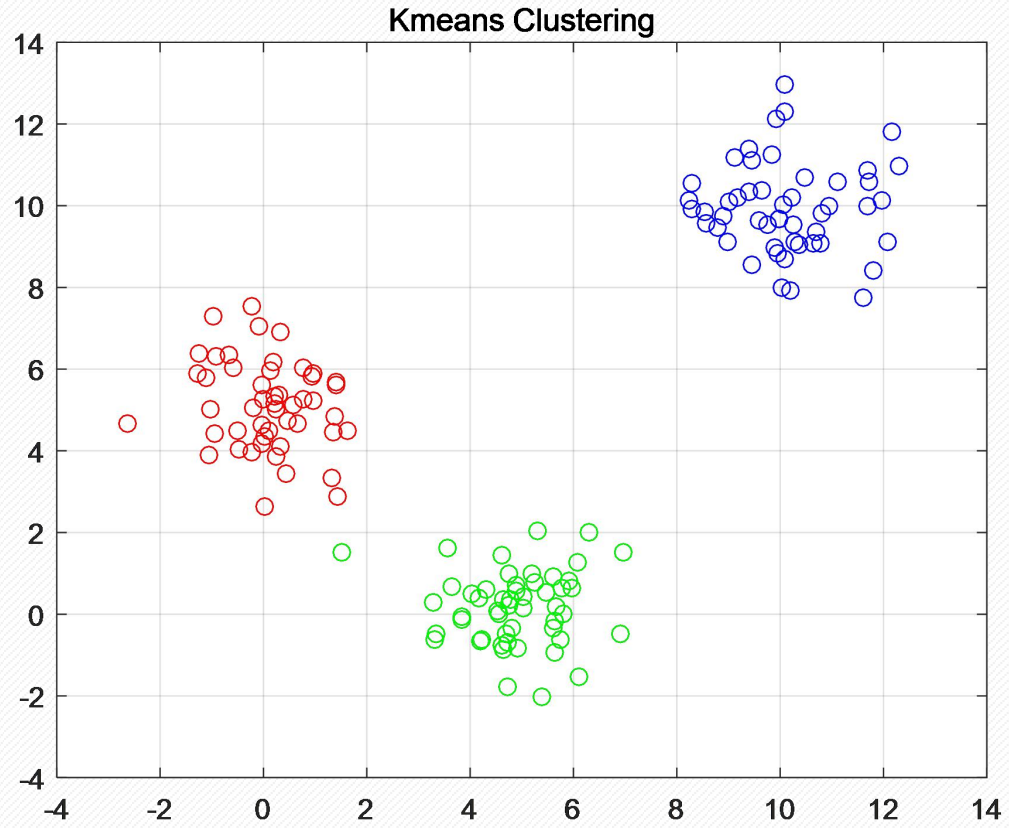


Preference=-2*abs(min(S));

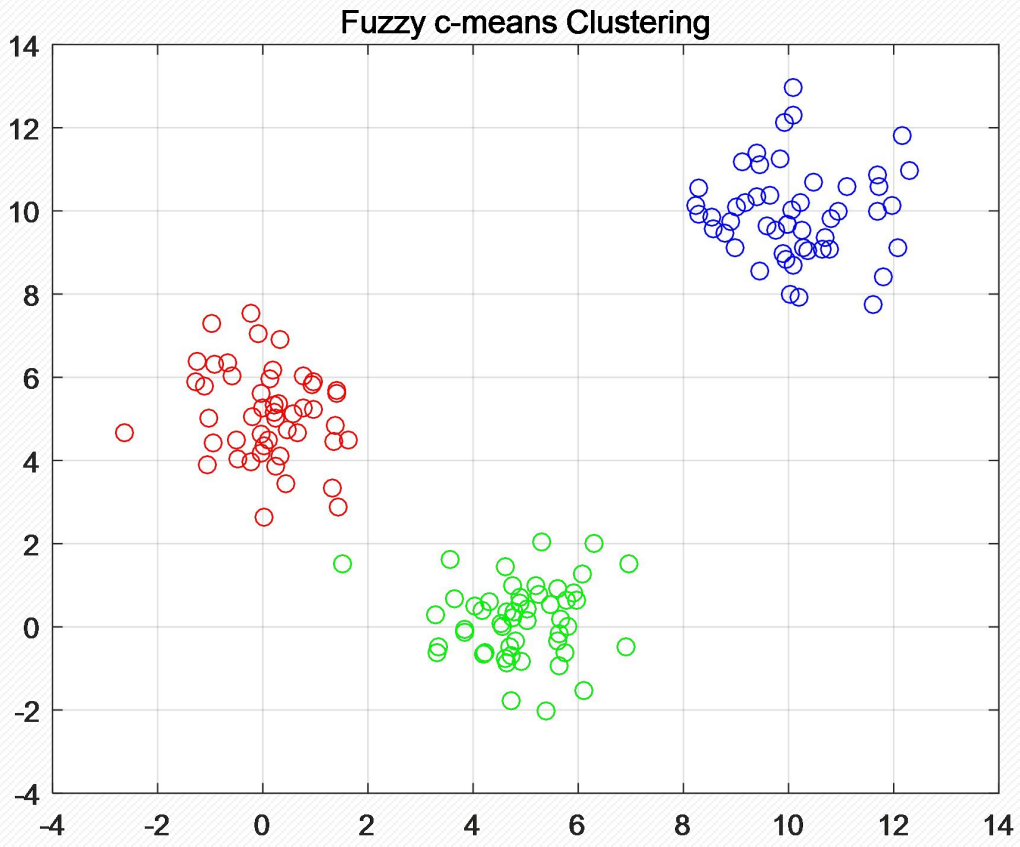
Damping factor=0.5;

S为相似度矩阵

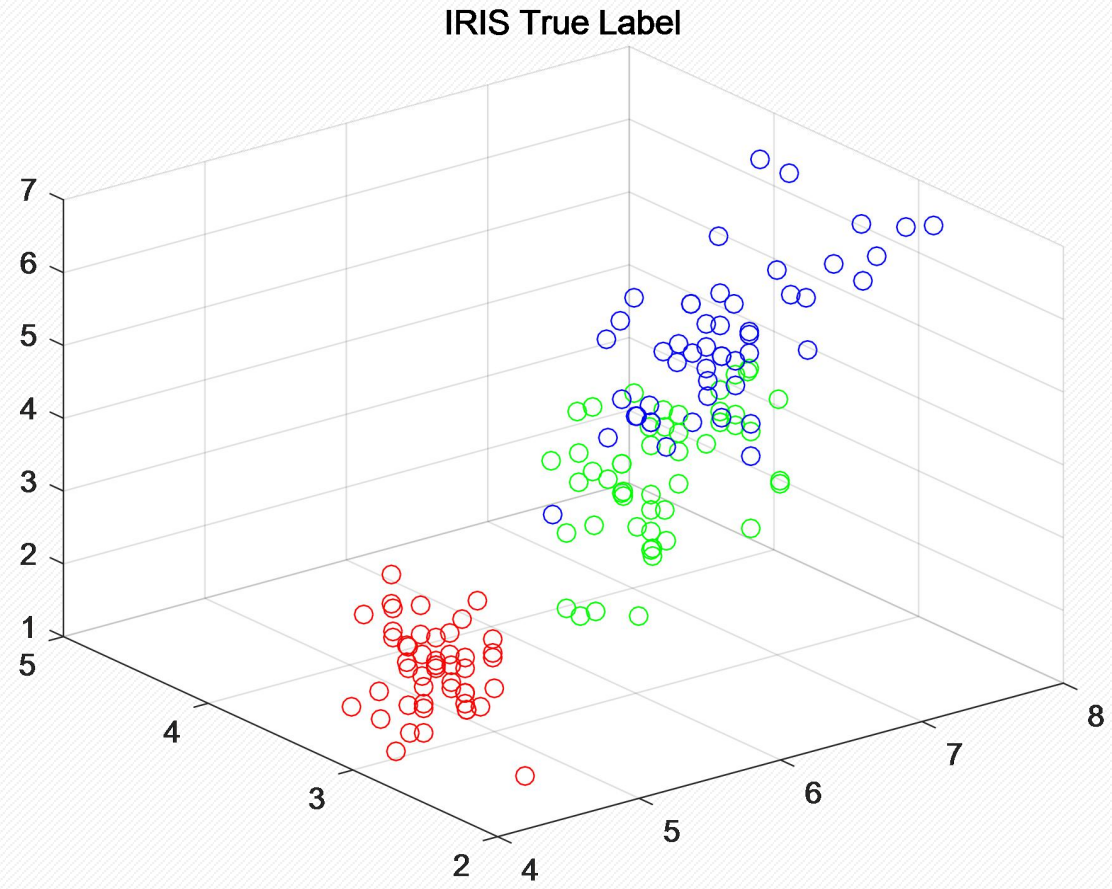
Case1:K=3,Kmeans Clustering



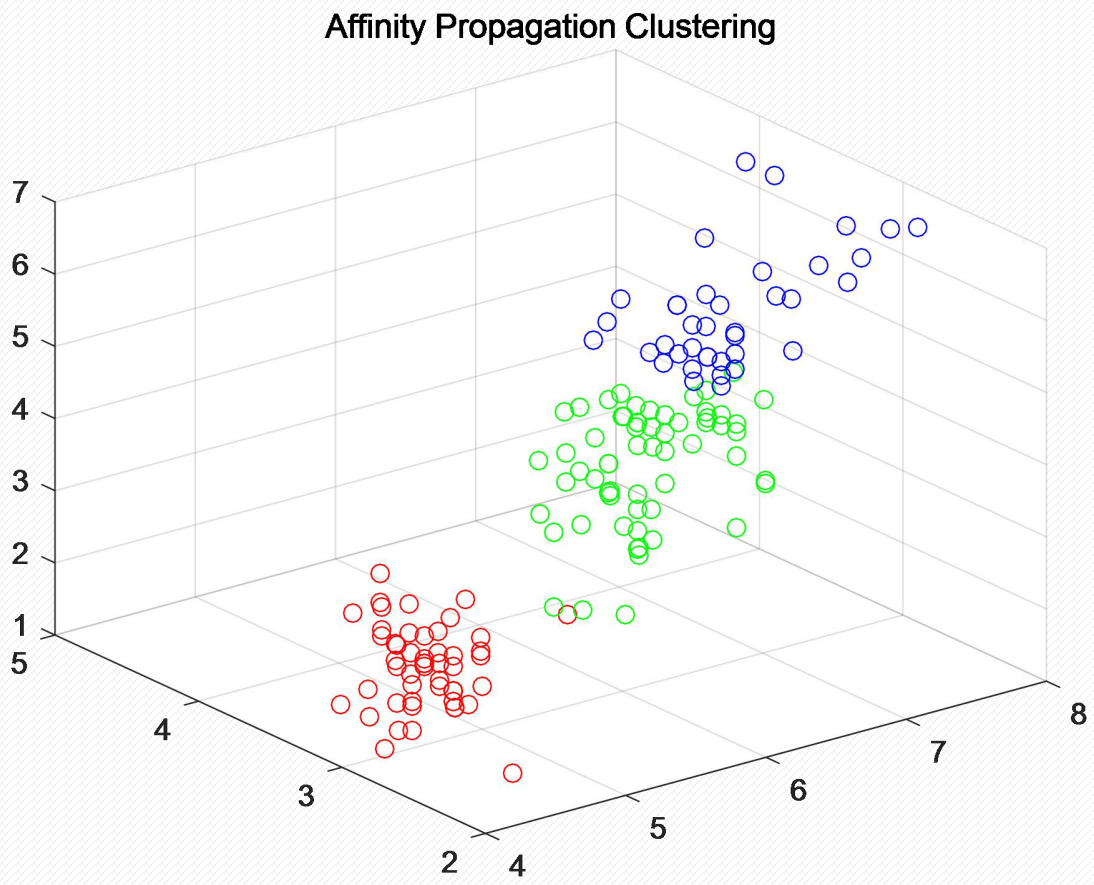
Case1:K=3,Fuzzy c-means Clustering



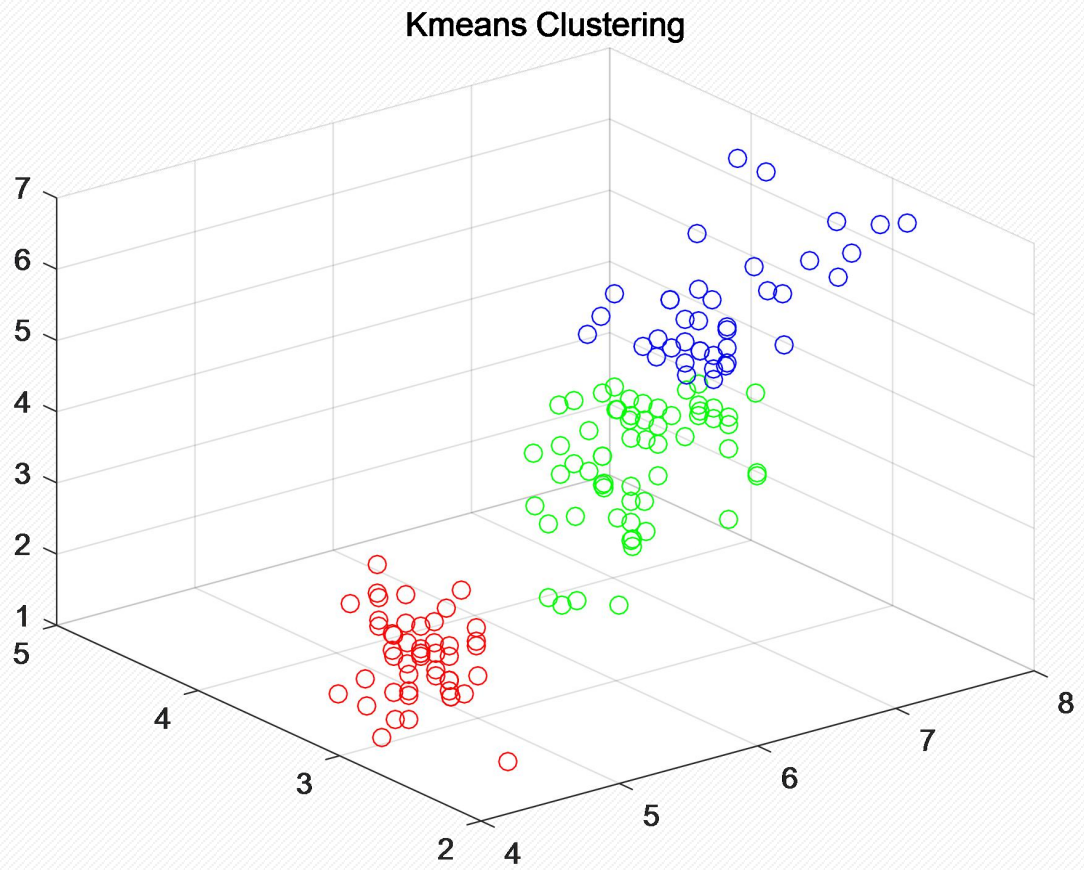
Case 2 (Iris):K=3,True Label



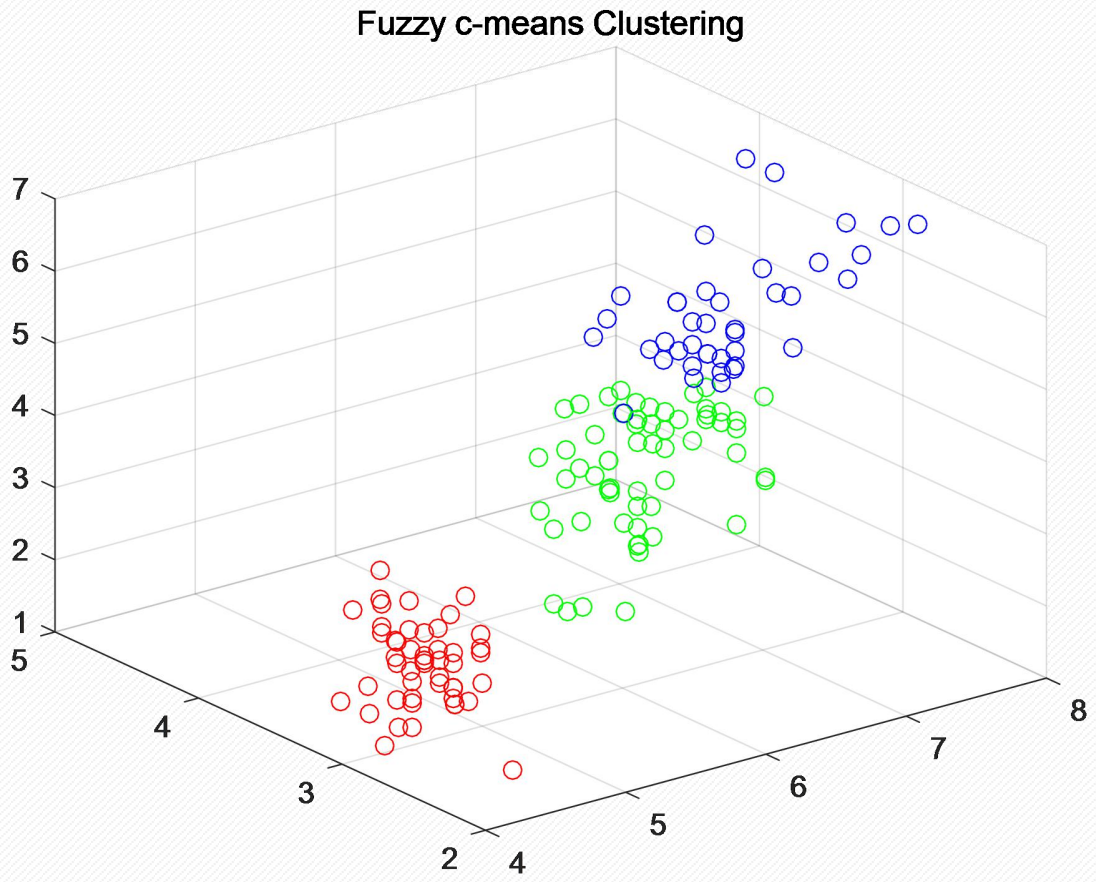
Case 2 (Iris):K=3,AP Clustering



Case 2 (Iris):K=3,Kmeans Clustering



Case 2 (Iris):K=3,Fuzzy c-means Clustering



Time Series Clustering

时间序列的聚类介绍，主要参考两篇综述文章，分别是
“Time-series clustering – A decade review”([2],2015年,information systems),
“Clustering of time series data—a survey”([3],2005,Pattern Recognition)

时间序列特点：
本质上归为动态数据，因为它的特征值随着时间的变化而变化；
维度高、数据量大（如一个小时的心电图数据占用1G，一个经典的网页博客每周需要5G，航天飞机数据库有200G，更新它需要每天2G）；
应用面广：在科学、工程、商业、金融、经济、医疗保健、政府等各个领域中无处不在且备受关注。

Time Series Clustering

时间序列数据聚类的重要性和必要性：

（1）时间序列的数据库可以通过模式发现获得的有价值的信息→聚类

（2）时间序列数据库非常庞大，用户习惯处理结构化的数据，通过聚类将相似的时间序列对象放到一起，表示为抽象的结构数据。

（3）在规则发现、索引、分类和异常检测等复杂数据挖掘技术前，聚类通常是最初的一种必要的探索性工具。

Time Series Clustering

时间序列聚类的应用：

（1）Anomaly,novelty or discord detection

例如在传感器数据库中，为了发现特定事件[35]，通过移动机器人的传感器读数来产生时间序列的聚类。

（2）Recognizing dynamic changes in time-series

时间序列之间相关性的测定[36]。例如，在金融数据库中，它可以用来查找股价变动相似的公司。

（3）Prediction and recommendation:

将聚类和每个聚类的函数逼近模型相结合，可以帮助用户预测和推荐[37-40]。例如，在科学数据库中，它可以解决诸如寻找太阳磁风模式以预测当下模式。

（4）Pattern discovery:

发现数据库中有趣或有价值的模式。例如，在市场数据库中，可以发现商店中特定产品的每日不同的销售模式。

Time Series Clustering

Table 1
Samples of objectives of time-series clustering in different domains.

Category	Clustering application	Research works
Aviation/ Astronomy	Astronomical data (star light curves) – pre-processing for outlier detection	[41]
Biology	Multiple gene expression profile alignment for microarray time-series data clustering	[42]
	Functional clustering of time series gene expression data	[43]
	Identification of functionally related genes	[44–46]
Climate	Discovery of climate indices	[47,48]
	Analysing PM_{10} and $PM_{2.5}$ concentrations at a coastal location of New Zealand	[49]
Energy	Discovering energy consumption pattern	[50,51]
Environment and urban	Analysis of the regional variability of sea-level extremes	[52]
	Earthquake - Analysing potential violations of a Comprehensive Test Ban Treaty (CTBT) – Pattern discovery and forecasting	[53,54]
	Analysis of the change of population distribution during a day in Salt Lake County, Utah, USA	[55]
	Investigating the relationship between the climatic indices with the clusters/trends detected based on clustering method.	[56]
Finance	Finding seasonality patterns (retail pattern)	[57]
	Personal income pattern	[58]
	Creating efficient portfolio (a group of stocks owned by a particular person or company)	[59]
	Discovery patterns from stock time-series	[60]
	Risk reduced portfolios by analyzing the companies and the volatility of their returns	[61]
	Discovery patterns from stock time-series	[29,62]
	Investigate the correlation between hedging horizon and performance in financial time-series.	[63]
Medicine	Detecting brain activity	[64,65]
	Exploring, identifying, and discriminating pathological cases from MS clinical samples	[66]
Psychology	Analysis of human behaviour in psychological domain	[67]
Robotics	Forming prototypical representations of the robot's experiences	[68,69]
Speech/voice recognition	Speaker verification	[70]
	Biometric voice classification using hierarchical clustering	[71]
User analysis	Analysing multivariate emotional behaviour of users in social network with the goal to cluster the users from a fully new perspective-emotions	[72]

Time Series Clustering

定义：Time-series clustering

给定一个n个时间序列数据集D = { F1,F2,...,Fn } ,基于一定的相似性度量，同质时间序列被分组在一起,将D聚类成K个簇C_k(k=1,2,...K)。

时间序列聚类的挑战：首先，时间序列数据通常比内存大得多，因此它们存储在磁盘上，这导致聚类过程的速度呈指数级下降；

第二，时间序列数据往往都是高维，这使得许多聚类算法处理这些数据很困难；

最后，如何构建一个恰当的相似性度量，用于时间序列聚类(时间序列有时不等长

Time Series Clustering

时间序列聚类大体分类三类：全时间序列聚类，子序列时间聚类，时间点聚类

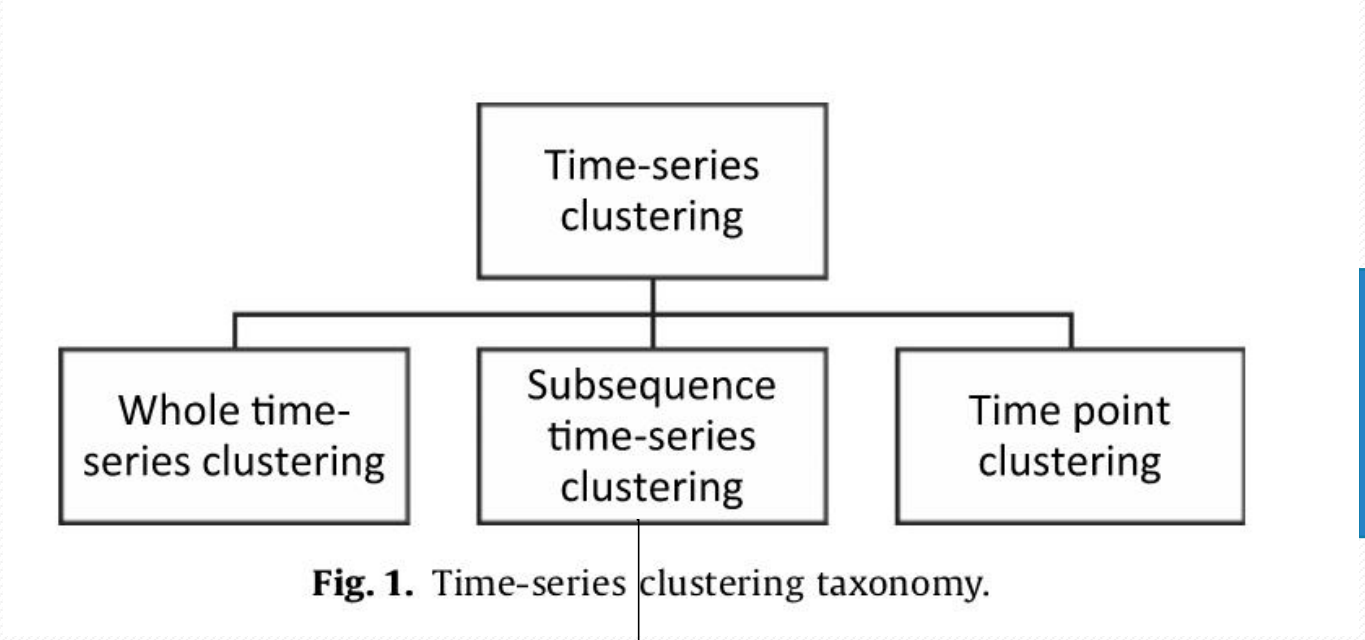


Fig. 1. Time-series clustering taxonomy.

Keogh and Lin [242]：Clustering of Time Series Subsequences is meaningless!

Time Series Clustering

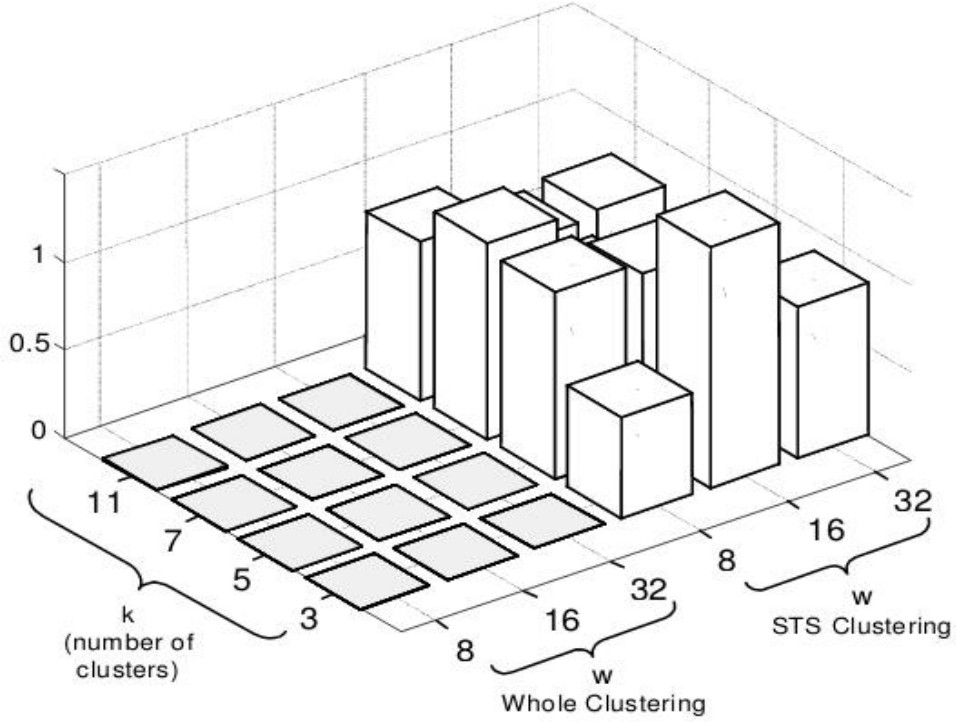


Figure 2. A comparison of the clustering meaningfulness for whole clustering, and STS clustering, using k-means with a variety of parameters. The two datasets used were Standard and Poor's 500 Index closing values and random walk data.

为什么子序列时间聚类无意义？

Whole time-series clustering

- 除了兼有传统聚类的特征外，通常有三种时间序列聚类方法，即基于形状的聚类、基于特征的聚类和基于模型的聚类。
- (1)基于形状的方法中，两个时间序列的形状通过时间轴的非线性拉伸和收缩尽可能匹配，这种方法也被称为基于原始数据的方法，因为它通常直接处理原始时间序列数据。基于形状的聚类算法通常采用传统的聚类方法，该方法与静态数据兼容，同时对距离/相似度的测度进行了适当的修正以适应时间序列
- (2)基于特征的方法中，将原始时间序列转换为低维特征向量。然后将传统的聚类算法应用到提取的特征向量上。通常在这种方法中，首先计算每个时间序列的等长度特征向量，然后计算欧式距离
- (3)基于模型的方法中，将原始时间序列转换为模型参数，然后选择合适的模型距离和聚类算法(通常是传统的聚类算法)对提取的模型参数[16]进行应用。然而，研究表明，通常基于模型的方法存在可伸缩性问题[78]，当簇彼此接近时，其性能下降[79]。

Whole time-series clustering

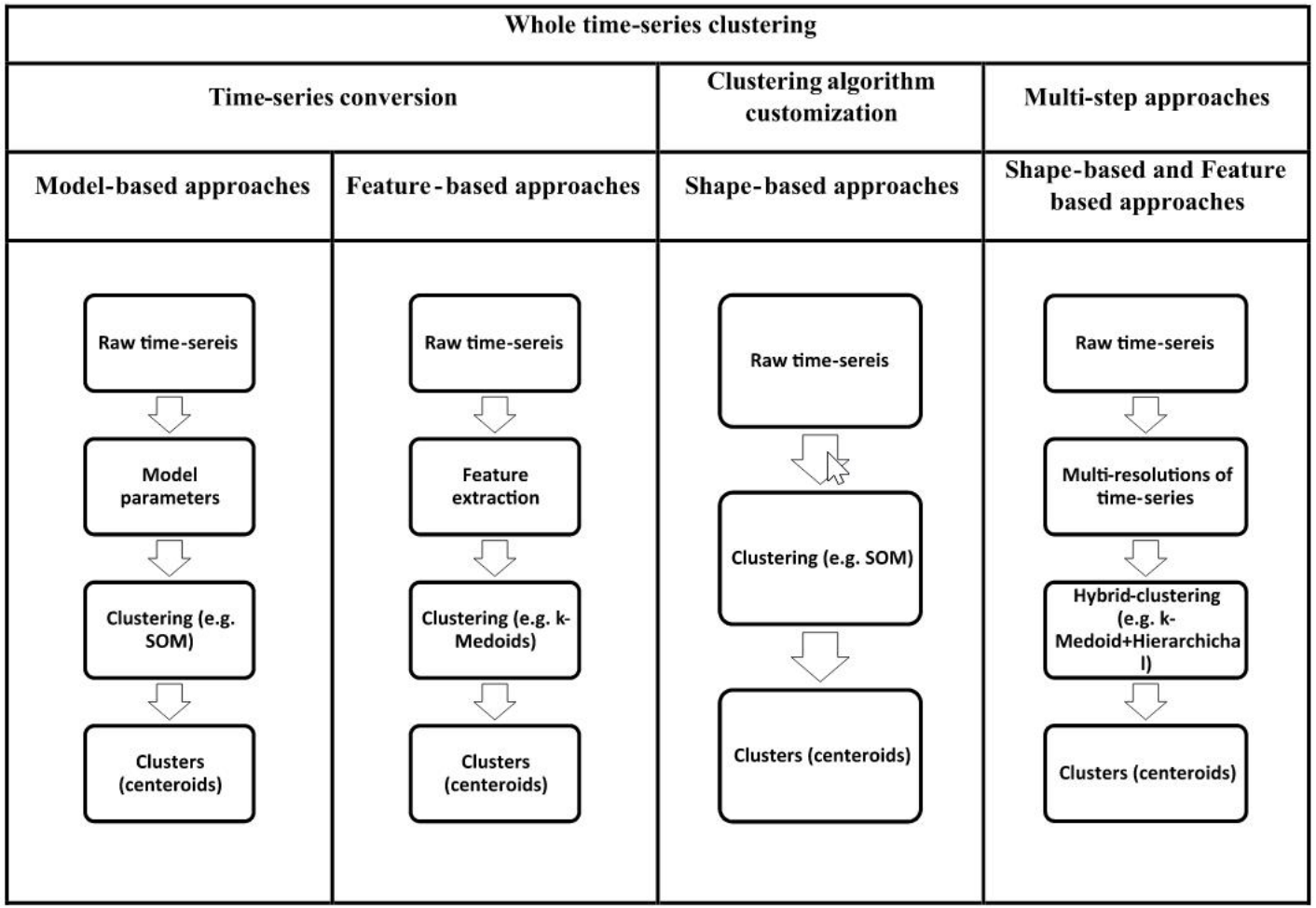


Fig. 2. The time-series clustering approaches.

SOM(self organizing maps):
是一种竞争学习的神经网络，Kohonen(1990)提出，可用于聚类。
相关论文[7]参考：
“Essentials of the self-organizing map”(2013,Neural Networks)

ART是另一种可用于聚类的神经网络， Carpenter and Grossberg（1987）

Whole time-series clustering

- 回顾已有的文献，时间序列聚类大概包含四个部分:降维或表示方法、相似性或距离度量、聚类算法、质心定义和评价。下图显示了这些概述

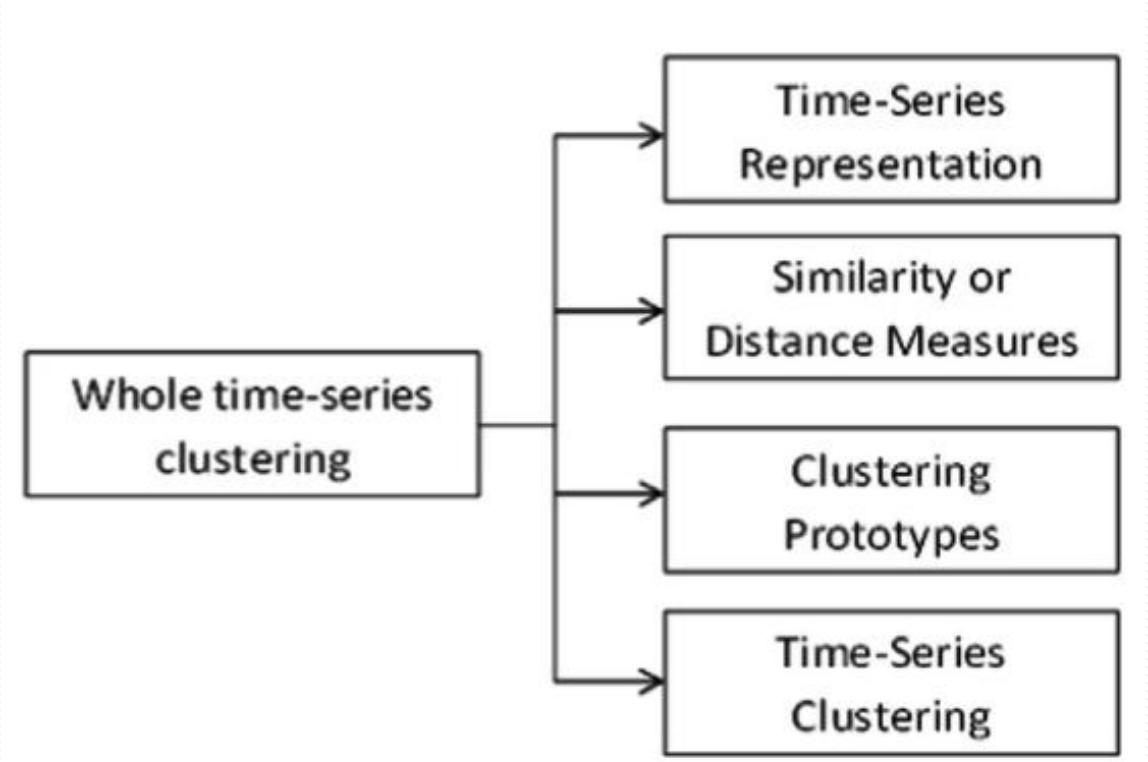


Fig. 3. An overview of four components of whole time-series clustering.

Whole time-series clustering

- **关于时间序列表示方法(Represention)：**
- 数据自适应:对任意不等长的时间序列段时间重构，使得重构后的误差最小化。奇异值分解,Symbolic Aggregate ApproXimation(SAX),分片多项式插值,分片线性逼近...
- 非数据自适应:一种适用于具有固定大小(等长)分段的时间序列的表示方法。
- Discrete Wavelet Transform，spectral Chebyshev Polynomials，spectral DFT，Piecewise Aggregate Approximation...
- 基于模型:以随机的方式表示时间序列，如马尔可夫模型和隐马尔可夫模型(HMM)，统计模型，自回归移动平均(ARMA)
- 数据指导法(Data Dictated)：压缩比是基于原始时间序列(如Clipped[83,92])自动定义的。

- H. Ding等[91]对38个数据集的8种表示方法进行了全面的比较，他们提出下界的紧密性来比较各个表示方法，他们指出表示法虽利于索引效率和聚类，但各个表示法之间差别不是很大。

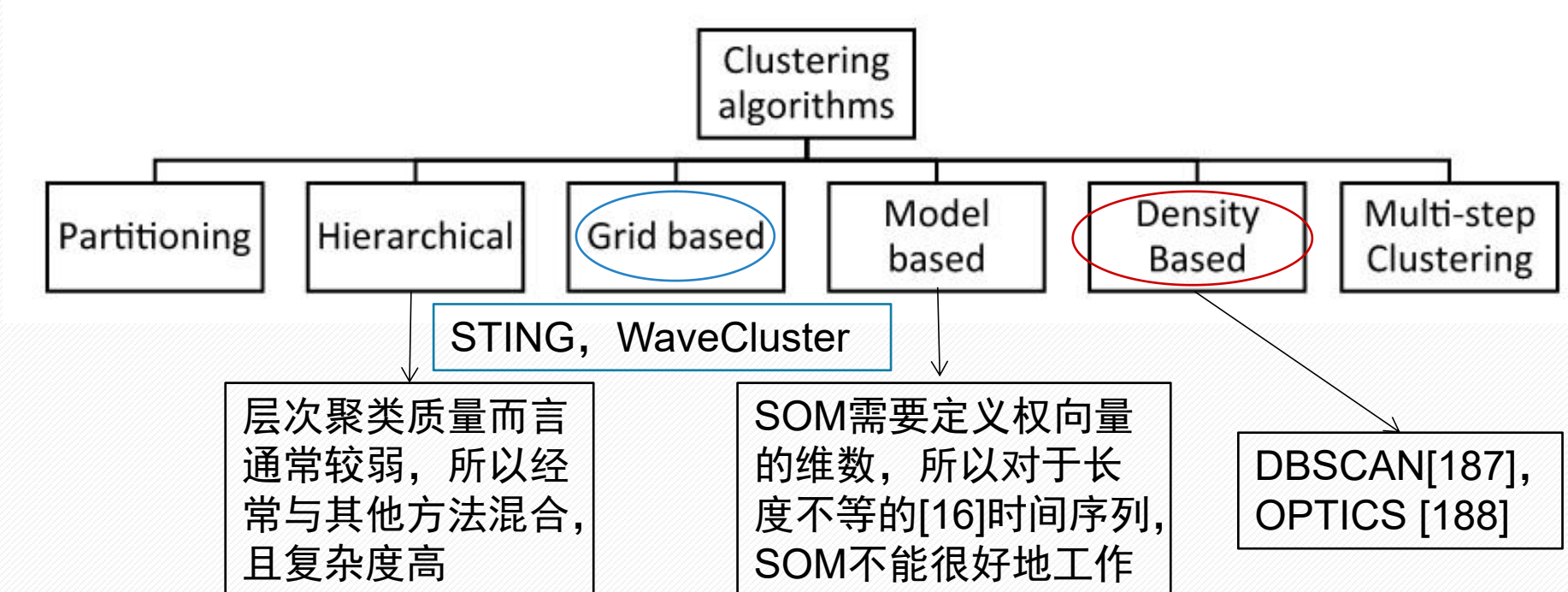
Whole time-series clustering

- **关于时间序列相似距离度量(Similarity measures)：**
- 时间序列聚类领域中，选择一个足够精确的距离度量是有争议的。
- （1）距离度量最有效和准确的方法是基于动态规划(DP)的方法，但计算代价非常高。
- （2）距离度量与表示方法的不兼容性，例如用于时间序列分析的一种常用方法是基于频域分析的方法[85,109]（频谱分析），而在使用这个频域空间，很难找到序列之间的相似性，并产生基于值的差异，来用于聚类。
- （3）研究表明：欧氏距离和DTW(Dynamic time Warping,Itakura)是时间序列聚类中最常用的相似性度量方法。在时间序列分类精度方面，欧氏距离具有惊人的竞争力[145]，然而，DTW在相似性度量方面也具有不可忽视的优势（DTW可以度量两个不等长的时间序列之间的距离）。

Whole time-series clustering

- **关于时间序列质心(Prototypes)：**
- 通常文献质心定义方式有：The medoid sequence of the set，The average sequence of the set，The local search

- **时间序列上聚类算法大致分类：**



Whole time-series clustering

Table 4
Whole time-series clustering algorithms.

Article	Representation method	Distance measurement	Clustering algorithm	Comments (P:Positive, N:Negative)
Košmelj and Batagelj [50]	Raw time-series	Euclidean	Modified relocation clustering	P: Multiple variable support
Golay et al. [132]	Raw time-series	Euclidean and two cross correlation-based	FCM	P: Noise Robustness
Kakizawa, Shumway, and Taniguchi [192]	Raw time-series	J divergence	Agglomerative hierarchical	P: Multiple variable support
Van Wijk and Van Selow [166]	Raw time-series	Root mean square	Agglomerative hierarchical	N: Single variable, using raw time-series
Policker and Geva [193]	Raw time-series	Euclidean	Fuzzy clustering	N: Single, using raw time-series
Qian, Dolled-Filhart, Lin, Yu, and Gerstein [194]	Raw time-series	Ad hoc distance	Single-linkage	N: using raw time-series Sensitive to noise
Kumar and Patel [57]	Raw time-series	Gaussian models of data errors	Agglomerative hierarchical	–
Liao et al. [152]	Raw time-series	DTW and Kullback–Liebler distance	*k*-Medoids-based genetic clustering	P: Support unequal time-series N: Single variable support Sensitive to noise
Wismüller et al. [64]	Raw time-series	***	Neural network clustering	N: Single variable support, using raw time-series
Möller-Levet, Klawonn, Cho, and Wolkenhauer [44]	piecewise linear function	STS	Modified FCM	–
Vlachos, Lin, and Keogh [165]	DWT (Discrete Wavelet Transform) Haar wavelet	Euclidean	k-means,	P: Incremental N: Sensitive to noise
Shumway [53]	Raw time-series	Kullback–Leibler discrimination information Measures	Agglomerative hierarchical	P: Multiple variable support
Lin, Vlachos, Keogh, and Gunopulos [18]	Wavelets.	Euclidean Distance	partitioning clustering, k-Means and EM	P: Incremental N: Sensitive to noise
Z.J. Wang and Willett [195]	Raw time-series	GLR (generalized likelihood ratio)	two stages approach	N: Subsequence Segmentation. Sensitive to noise

Table 4
Whole time-series clustering algorithms.

Article	Representation method	Distance measurement	Clustering algorithm	Comments (P:Positive, N:Negative)
Košmelj and Batagelj [50]	Raw time-series	Euclidean	Modified relocation clustering	P: Multiple variable support
Golay et al. [132]	Raw time-series	Euclidean and two cross correlation-based	FCM	P: Noise Robustness
Kakizawa, Shumway, and Taniguchi [192]	Raw time-series	J divergence	Agglomerative hierarchical	P: Multiple variable support
Van Wijk and Van Selow [166]	Raw time-series	Root mean square	Agglomerative hierarchical	N: Single variable, using raw time-series
Policker and Geva [193]	Raw time-series	Euclidean	Fuzzy clustering	N: Single, using raw time-series
Qian, Dolled-Filhart, Lin, Yu, and Gerstein [194]	Raw time-series	Ad hoc distance	Single-linkage	N: using raw time-series Sensitive to noise
Kumar and Patel [57]	Raw time-series	Gaussian models of data errors	Agglomerative hierarchical	–
Liao et al. [152]	Raw time-series	DTW and Kullback–Liebler distance	*k*-Medoids-based genetic clustering	P: Support unequal time-series N: Single variable support Sensitive to noise
Wismüller et al. [64]	Raw time-series	***	Neural network clustering	N: Single variable support, using raw time-series
Möller-Levet, Klawonn, Cho, and Wolkenhauer [44]	piecewise linear function	STS	Modified FCM	–
Vlachos, Lin, and Keogh [165]	DWT (Discrete Wavelet Transform) Haar wavelet	Euclidean	k-means,	P: Incremental N: Sensitive to noise
Shumway [53]	Raw time-series	Kullback–Leibler discrimination information Measures	Agglomerative hierarchical	P: Multiple variable support
Lin, Vlachos, Keogh, and Gunopulos [18]	Wavelets.	Euclidean Distance	partitioning clustering, k-Means and EM	P: Incremental N: Sensitive to noise
Z.J. Wang and Willett [195]	Raw time-series	GLR (generalized likelihood ratio)	two stages approach	N: Subsequence Segmentation. Sensitive to noise

Whole time-series clustering

- **Multi-step clustering：多步聚类(一般是一种混合方法)。**
- 在大多数模型中，都是直接使用时间序列数据原始数据或降维后的数据，再使用经典的传统聚类算法去做。
- 很明显，这种不经过任何优化就使用蛮力的方法应用于时间序列中，不一定适用于现实世界的问题，尤其是在大型数据库中它们表现非常缓慢或不准确。
- 因此，迫切希望找到一种定制化的聚类算法来专门处理时间序列数据。Multi-step clustering就是基于此目标提出的。
- 例如，Aghabozorgi and Wah[2014,Expert Systems with Applications]针对股票市场的协同运动， 提出了一个三阶段时间序列聚类(3PTC)。
- (a.时间序列的粗略聚类 b.优化上步预聚类结果并汇总 c.合并质心得到最终聚类)

Whole time-series clustering

- **Multi-step clustering：多步聚类(3PTC model,**[8])
- (a.时间序列的粗略聚类 b.优化上步预聚类结果并汇总 c.合并质心得到最终聚类)
- 该模型有助于时间序列数据集的精确聚类，是专门为非常大的时间序列数据集设计的。在模型的第一阶段，对数据进行预处理，转换为低维空间，并近似分组；
- 在第二阶段利用精确聚类方法对预聚类时间序列进行优化，并用质心进行表示；
- 最后，在第三阶段，合并质心以构建最终的集群。为了评估提出的模型的准确性，3PTC对来自不同领域发布的时间序列数据集进行测试。结果表明，所提出的方法可以更好地预测和理解上市公司股价的协同运动，甚至是局部的变化。
- **时间序列常用测试数据库链接：**(截至2018秋，共发布128个数据集)
- http://www.cs.ucr.edu/~eamonn/time_series_data/
- 后面是聚类算法的评估介绍[略]

（Rand index,entropy,Normalized Mutual Information）

GMM based on Mahalanobis distance

主要参考如下文章(想法，尝试用模糊聚类估计高斯混合的协方差矩阵和均值)

1.“Gaussian Mixture Modeling by Exploiting the Mahalanobis Distance”
（[4],2008,IEEE Transactions on Signal Processing）

2. “Fuzzy Gaussian Mixture Models” ([5],2012,Pattern Recognition)

3.”Faster Mahalanobis K-means clustering for Gaussian distributions”
([6],2016,International Conference on Advances in Computing. IEEE)

Reference

[1].Frey B J , Dueck D . Clustering by Passing Messages Between Data Points[J]. Science, 2007, 315(5814):972-976.

[2].Aghabozorgi S , Seyed Shirkhorshidi A , Ying Wah T . Time-series clustering – A decade review[J]. Information Systems, 2015, 53:16-38.

[3].Liao T W . Clustering of time series data—a survey[J]. Pattern Recognition, 2005, 38(11):1857-1874.

[4].Ververidis D , Kotropoulos C . Gaussian Mixture Modeling by Exploiting the Mahalanobis Distance[J]. IEEE Transactions on Signal Processing, 2008, 56(7):2797-2811.

[5].Ju Z , Liu H . Fuzzy Gaussian Mixture Models[J]. Pattern Recognition, 2012, 45(3):1146-1158.

[6].Chokniwal A , Singh M . Faster Mahalanobis K-means clustering for Gaussian distributions[C]// International Conference on Advances in Computing. IEEE, 2016.

[7]Kohonen T . Essentials of the self-organizing map[J]. Neural Networks, 2013, 37(none):52---65.

Reference

[8].Aghabozorgi S , Teh Y W . Stock market co-movement assessment using a three-phase clustering method[J]. Expert Systems with Applications, 2014, 41(4):1301-1314.

[242]E. Keogh, J., Lin, Clustering of time-series subsequences is mean-

ingless: implications for previous and future research, Knowledge

and information systems 8 (2) (2005) 154–177.