DOI: 10.19783/j.cnki.pspc.190144

基于分位数半径动态 *K*-means 的分布式负荷聚类算法

刘季昂[1]，刘友波[1]，程明畅[2]，余莉娜[3]

(1. 四川大学电气信息学院，四川 成都 610065；2. 西南财经大学统计学院，四川 成都 611130；
3. 中国三峡新能源有限公司西南分公司，四川 成都 610023)

摘要：针对电力负荷曲线聚类中传统的 *K*-means 算法对初始值敏感以及需给定类数目的缺陷，将一种基于分位数半径的动态 *K*-means 算法应用到日负荷曲线的聚类分析中，并进行了分布式的改进以优化计算效率。此算法结合了两种思想：分布式聚类中的局部聚类与全局聚类，以及层次 *K*-means 中以多次 *k* 取定值时 *K*-means 运算所得到的中心点来表示该类。将多次的 *K*-means 运算分配到不同子站点，并使每次 *K*-means 运算中 *k* 不断改变。再从类的几何特征出发，引入了分位数半径的概念，规定样本点与各类中心点间距的分位数表示该类的半径，于主站点中对各类的中心点间距与类的半径进行大小比较，并进行筛选融合来获得新的类，从而实现较为快速地识别类数目，并且得到新的聚类初始中心与结果。最终以某地区 606 个用户某月的日负荷数据为研究对象，验证了该算法在电力负荷曲线聚类分析中的有效性。

关键词：电力大数据；聚类分析；负荷曲线聚类；分位数半径；分布式聚类

A distributed load clustering algorithm based on quantile radius dynamic *K*-means

LIU Ji'ang[1], LIU Youbo[1], CHENG Mingchang[2], YU Lina[3]

(1. School of Electrical Engineering and Information, Sichuan University, Chengdu 610065, China;
2. School of Statistic, Southwestern University of Finance and Economics, Chengdu 611130, China;
3. China Three Gorges New Energy Limited Company Southwest Branch, Chengdu 610023, China)

Abstract: Aiming at the defects of traditional *K*-means algorithm in the power load curves clustering, such as sensitive to initial values and given the number of clusters, a dynamic *K*-means algorithm based on quantile radius is applied to the clustering analysis of daily load curves, and distributed improvement is made to optimize the computational efficiency. This algorithm combines two ideas: local clustering and global clustering in distributed clustering, and the central point obtained by *K*-means operation when *k* is repeatedly set as the fixed value for many times in the hierarchical *K*-means. Multiple *K*-means operations are assigned to different subsites, and the *k*-value of each *K*-means operation is changed. Then from the geometrical characteristics of the clusters, the concept of quantile radius is introduced. Quantile radius defines that the quantile of the distance between the sample point and the cluster center point represents the radius of the cluster. At the main site, the distance between the center points of each cluster with the quantile radius of the cluster is compared to filter and merge to get new clusters, so that the number of clusters can be quickly identified and a good initial center and result of the cluster are given. Finally, the daily load data of 606 users in a certain area and in a certain month is taken as the research object, and the effectiveness of the algorithm in the cluster analysis of power load curves is verified.

This work is supported by National Key Research and Development Program of China (No. 2017YFE0112600).

Key words: power big data; cluster analysis; load curves clustering; quantile radius; distributed clustering

0 引言

建设中国特色智能电网，是立足于经济社会对我国电网发展要求的战略性选择。随着智能电网的不断建设，智能电表的普及率逐渐升高，将采集到海量的用户用电数据。不同于传统的每月抄表，时间粒度更细的用户用电数据中蕴含了更多的用户用电信息，也能够反映更多的用户用电规律。如何充

基金项目：国家重点研发技术项目资助(2017YFE0112600)

分合理地使用这些用户用电数据，并且从中提取出有价值的信息来为提高电网运行可靠性与经济社会效益提供帮助，变成了眼下电力系统急需解决的一个问题[1]。

聚类属于机器学习当中一种无监督的学习方式，即将数据集中的样本划分成多个是互不相交的子集，各子集为一个“簇”，其中各个簇对应于一些未知的概念(类别)。电力负荷曲线聚类能够有效地辨识典型的负荷曲线，并且将具有相似用电模式的用户归为同一类，是配用电领域的基础，也是不良数据识别[2]、需求侧管理和能效管理[3]、用电客户精细分类、电力负荷特性分析[4]等多种配用电数据挖掘应用的基础[5]。因此，对适合电力负荷曲线聚类的算法进行研究是必要的，也是实现智能电网用电控制的关键之一。

传统的聚类算法主要被分为基于划分、基于层次、基于密度、基于网格和基于模型等不同种类[6]。考虑到电力负荷曲线作为一种时间序列，用于处理空间数据的基于网格的聚类算法通常不会被采用。在电力负荷曲线聚类中应用比较成熟的算法包括*K*-means 算法和 FCM、层次聚类、SOM 等方法[7-10]，其中，基于划分的聚类算法因其原理简单、编程易于实现、运行速率较快等特性应用较为普遍，但同时也存在着 *k* 值的确定、局部收敛性(依赖初始数据的选择)、结果不稳定、对异常用电模式敏感[6]等问题。对于该类算法所存在的上述问题，众多学者提出了改进。文献[11]提出的 *K*-means++算法对初始中心点的选择进行了优化以增加算法的准确度。文献[12]提出了一种将层次聚类和划分聚类相结合的用户用电曲线集成聚类算法，兼顾了效率与精度，但依旧未能解决 *k* 值选择的问题，仅能确保 *k* 取某个固定值时可以获得相对稳定的分类。除此之外，也有学者提出通过几何特性来提升算法的运算效率，例如文献[13]提出通过三角不等式对是否需要进行距离计算来进行判断。针对 *k* 值确定的问题，文献[14-16]中均提出了能够自适应地确定类数目的新算法。

值得注意的是，传统的聚类算法是集中式聚类，即对集中存放在单个站点的数据集进行聚类。而在实际应用中，一方面用户用电的数据随时间不断增加，另一方面数据采集点具有极强的分散性。大量的数据对数据分析的时间和空间效率提出了更高的要求，而分散的数据则为数据通信带来了挑战。因此针对用户用电数据的聚类，不仅需要考虑提高聚类算法的准确度，还需考虑调高聚类算法的计算效率和数据的通信成本。不同于传统的负荷曲线聚类算法要将数据集中于单个数据中心，分布式聚类算法不仅能够对数据进行并行处理从而在一定程度上提高数据的处理效率，而且能够极大程度地减少数据的通信成本[18]。

本文考虑到了类的几何特性，提出一种引入了分位数半径概念的动态 *K*-means 算法，即将样本点相互间的关系简化为各类的分位数半径与其类的中心点之间的关系，并且以中心点间距与分位数半径间的大小关系为判据进行筛选融合，从而快速获得良好的聚类结果。在此基础上，结合了分布式聚类算法的基本思想，将基于分位数半径的动态 *K*-means 算法进行一定改进并应用于电力负荷曲线的聚类分析，再以某地区 606 个用户的实测数据进行算例分析，并与传统算法进行比较。

1 算法设计

1.1 *K*-means 算法与层次 *K*-means 算法

K-means 算法是由 MacQueen 于 1967 年提出的一种基于划分的聚类算法[20]，也是相对基础且应用最普遍的一种聚类算法，其主要步骤如下。

(1) 任意选取 *k* 个样本点作为初始聚类中心(可不包含在数据集中)。

(2) 分别计算每个样本点到 *k* 个中心的距离，并将各个样本点均划分到距离最近的中心点所在的类中。

(3) 计算各类的中心点，若得到的中心点与前一次为同一个点，则输出；否则以新得到的点作为初始点重新执行步骤(2)。

尽管该算法因其算法思想简单、聚类速度较快、聚类效果良好以及处理大数据集时的可伸缩性与高效性等特点获得了广泛应用，但是也存在着很多公开的问题。例如类数目 *k* 值的确定、局部收敛性等，使得该算法在一些情况下无法提供好的聚类结果。

层次 *K*-means 算法是 Arai 和 Barakbah 于 2007 年提出的混合算法[21]，通过 *K*-means 算法来弥补层次算法在大规模数据计算中时间消耗大的缺陷，能够在一定程度上解决初始聚类中心的选取问题。该算法的核心思想是以重复进行的若干次 *K*-means 计算所得到的聚类中心来替代大量样本的分布，并且对得到的全部中心点层次聚类给出更优的初始点，最终以这些初始点进行一次 *K*-means 运算求得更优良的聚类效果。其主要步骤如下。

(1) 进行 *p* 次 *k* 取定值的 *K*-means 计算，且将每次得到的聚类中心存入集合 *C*。

(2) 对集合 *C* 中的数据点进行层次聚类，存储分类中心。

(3) 以步骤(2)得到的点为聚类中心进行 *K*-means 计算，输出结果。

1.2 基于分位数半径的动态 *K*-means 算法

考虑到层次 *K*-means 算法未能解决 *k* 值的选取问题，且未能充分利用类的几何信息，本文结合层次 *K*-means 的思想提出了一种基于分位数半径的动态 *K*-means 算法(QRD *K*-means)。图 1 给出了该算法的基本流程，主要包含 4 个步骤。

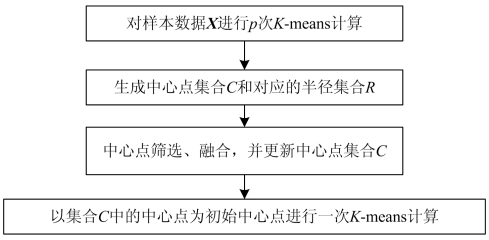


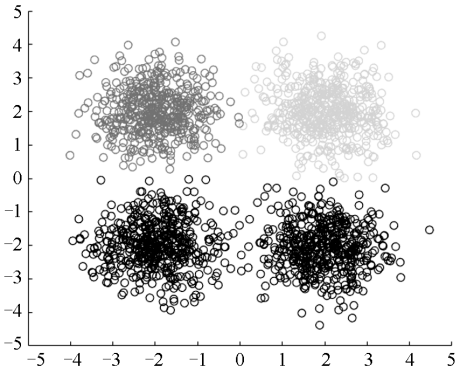
图 1 基于分位数半径的动态 *K*-means 算法基本流程

Fig. 1 Flow chart of dynamic *K*-means algorithm based on quantile radius

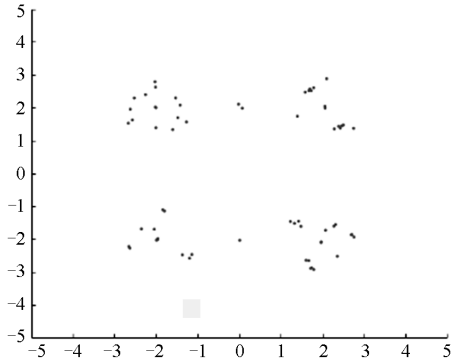
该算法通过让 *k* 值变动(如 *k* 从 2 取到 *p*+1)，并对每个 *k* 值依次进行一次 *K*-means 计算，以计算得到的聚类中心替代原始数据点。随着 *k* 值的增大，各次 *K*-means 生成的聚类中心则会更加分散，其涵盖的信息也会不断增加。如图 2(a)所示，给出了一组实际类数目为 4 的模拟数据，对该组数据进行 10 次 *k* 取值分别为 2~11 的 *K*-means 计算，得到了如图 2(b)所示的中心点。可以看出，中心点的分布状况能够明显地反映数据集中各类的分布状况。

1.2.1 分位数半径

此外，该算法还引入了分位数半径的概念，即规定样本点与各类中心点间距的分位数表示该类的半径 *r*，且半径可随着分位数选取的不同而改变[17]。具体来说，将某一类包含的全部数据点根据其与中心点间距的大小排列，那么 *t*%分位数对应的数据点与中心点间距即代表了该类的 *t*%分位数半径。选取不同的分位数会得到不同的类半径，从而影响最终



(a) 类数目为 4 的模拟数据



(b) 10 次 *K*-means 计算得到的中心点

图 2 中心点集合的产生

Fig. 2 Generating of center sets

的聚类结果。当分位数选取过大时，类的半径过大，不同的类之间的信息相互重叠导致欠拟合；反之，分位数选取过小时则可能导致过拟合。因此，本文选取了 25%分位数半径来表示其类的规模。

1.2.2 中心点筛选与融合

当某个较大的类半径与另外分属于两个类的半径相交时，可能会对类的数目和最终的聚类结果产生不良影响。为了筛除那些可能会对聚类结果造成负面影响的中心点，增强聚类准确性，该算法结合中心点间距与类的分位数半径对中心点进行筛选，设定条件如下。

$$d(c_i,c_j)+\min(r_i,r_j)<T\cdot\max(r_i,r_j) \tag{1}$$

式中：$c_i\in C$ 表示第 *i* 个类的中心点；$r_i\in R$ 表示第 *i* 个类中心点所对应的半径；*T* 表示人为选取的控制参数，*T* >1。符合此约束条件，即两个中心点之中相对大的半径大于相对小的半径与两个中心点间距之和时，可以认为半径相对大的类中发生了不相同类之间的信息交叉，使得原本应区分开的类被合并，因此将半径相对大的中心点去除，再对中心点集合 *C* 进行更新。

为了能够自适应地得到适合的 *k* 值，并且能得到更加好的初始中心点，算法对筛选之后剩余的聚类中心进行融合。融合中心点的核心思想就是将集合 *C* 中间距相近的中心点融合为一类，并将该类各个中心点的均值设为新获得的类的中心点。因此设定条件如下。

$$d(c_i,c_j)<r_i+r_j \tag{2}$$

式中：$c_i\in C$ 表示第 *i* 个类的中心点；$r_i\in R$ 表示第 *i* 个类中心点所对应的半径。满足此约束条件，即两个中心点的半径和大于两个中心点的间距时，可以将两个中心点融合为一类。值得注意的是，新类的产生应当满足[17]：(1) 传递性，即中心点 *a* 和 *b*

合并为一类，且中心点 b 和 c 合并为一类，则中心点 a 和 c 合并为一类；(2) 互不相容性，即每个中心点仅属于一个类。

为了实现中心点的融合，文献[17]中设定了指针矩阵 $\boldsymbol{pointer} \in \boldsymbol{R}^{n\times n}$，$n$ 表示中心点的个数。指针矩阵 $\boldsymbol{pointer}$ 中的元素可以表示为

$$\boldsymbol{pointer}(i,j)=\begin{cases}1 & d(c_i,c_j)<r_i+r_j \\ 0 & d(c_i,c_j)>r_i+r_j \text{ or } i=j\end{cases} \tag{3}$$

可以得到这一指针矩阵：

$$\boldsymbol{pointer}=\begin{bmatrix}0&1&0&1\\1&0&0&0\\0&0&0&1\\1&0&1&0\end{bmatrix}_{4\times4} \tag{4}$$

当 $\boldsymbol{pointer}(i,j)=1$ 时，第 i 个中心点与第 j 个中心点应当合并。结合其传递性与互不相容性，中心点融合的过程可以概括为以下 3 个步骤。

(1) 设定集合 $cen=\{1,\cdots,n\}$ 来存储仍未被分类的中心点，集合 s={}来存储已被分类的中心点。

(2) 对 cen 进行判断：若 cen 中仍有元素，则取 i=cen (1)执行步骤(3)；否则输出为最终分类结果。

(3) 将第 i 个中心点存储进 s，并从 cen 中删去该中心点。对指针矩阵 $\boldsymbol{pointer}$ 的第 i 行元素进行判断。若元素全为 0，则该中心点为单独的一类，执行步骤(2)，否则记录该行所有非零元素列标，对列的脚标 j 进行判断(当所有列的脚标判断完毕后，回到步骤(2))。若 j 属于集合 s 则跳过，否则与第 i 个中心点合并，并记 i=j 且重复执行步骤(3)。

2　算法的分布式处理过程

基于分位数半径的动态 K-means 算法能够较好地识别 k 值，并且还能给出较为良好的初始中心，以确保聚类准确性。对中心点的筛选和融合也能够有效地提高算法的计算效率。但考虑到用户用电数据数量庞大、采集点分散、依赖数据通信等特性，在实际应用中基于分位数半径的动态 K-means 算法计算效率的优势并不能得到很好的发挥。因此，针对电力负荷曲线聚类，本文结合分布式聚类的基本思想对上述算法进行改进，即先对子站点的样本点进行局部聚类，再将各局部聚类得到的中心点发送至主站点统一进行中心点的筛选、融合，最终对筛选、融合后获得的初始中心点进行 K-means 聚类。

2.1　分布式聚类概述

分布式聚类是一种适用于从大规模、分散储存的数据集中获取分类模式的聚类算法。目前较为常见的分布式聚类算法大致可以分为基于密度、基于划分、基于特征向量、基于分型维度和基于隐私保护等几类算法[19]。其基本思想可被划分为以下三个部分：

(1) 对分散在各子站点的局部数据分别进行聚类；

(2) 将各个子站点的局部聚类结果传给主站点以进行全局聚类；

(3) 将主站点的全局聚类结果传回各个子站点，子站点再根据全局聚类结果进行调整、更新。

分布式的聚类算法在数据处理之初就可以单独地处理各局部数据，各子站点同时进行、互不干扰，以提高数据分析处理的效率。最终只需要将处理后的结果传输到主站点即可，降低了实时通信需求，因此更加适合数据量巨大且采集点相对分散的情况。值得注意的是，由于子站点传输的是局部聚类结果，即本站点数据集中的部分代表点，可能无法涵盖数据集的全部信息，因此相较于集中式聚类算法，分布式聚类算法的准确性相对较低。这就意味着聚类准确性与站间通信量之间无法兼顾，而如何良好地平衡聚类准确性和站间通信量之间的关系正是分布式聚类算法研究所面对的困难。

2.2　基本原理

分布式 QRD K-means 算法的流程如图 3 所示。

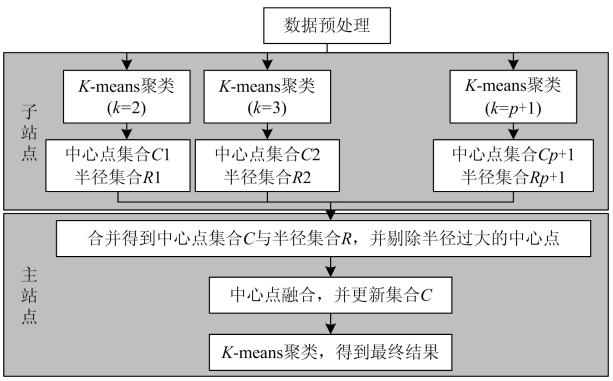


图 3　分布式 QRD K-means 算法基本流程

Fig. 3 Flow chart of QRD K-means algorithm

首先需要对原始负荷数据进行校验和归一化处理，即删除包含有空缺值负荷采集点的原始负荷数据。其次将所有待聚类的负荷数据分别传送至 N 个子站点，每个子站点的数据标记为局部数据 $i(i=1,2,\cdots,N)$。在上文中提到，QRD K-means 算法将对样本数据 X 进行 p 次 K-menas 计算。为了提高计算效率对算法进行分布式处理，设定参数 p，对各子站点的数据分别进行 k 值取 $p+1,p,\cdots,2$ 的 K-means 运算，即将原本的计算量同时分配到各子

站点中。通常来说，子站点个数 N 远大于设定的参数 p，但是在子站点较少的情况下(即 $p \geq N$ 时)，p 次 *K*-means 运算将无法均匀分配给 N 个子站点，则此时 k 值较小的子站点需进行多次 k 值不同的 *K*-means 计算以分摊计算量。各子站点计算结束后将得到的 N 个中心点集合 $C_i(i=1,2,\cdots,N)$ 与其对应的半径集合 $R_i(i=1,2,\cdots,N)$ 传送至主站点。主站点接收到 N 个中心点集合与半径集合后，根据设定条件对中心点进行筛选、融合得到中心点集合 C，以集合 C 中的中心点为初始中心点，再对负荷数据进行 *K*-means 计算，可以得到最终的聚类结果。

2.3 聚类评价指标

聚类结果的质量评价和最佳聚类数确定均是通过确定聚类有效性指标来完成的。常见的聚类有效性指标有误差平方和(Sum of Squared Error, SSE)、Calinski-Harabasz 指标(CHI)、Davies-Bouldi 指标(DBI)等[12]。

(1) SSE 指标

某个子类到所在类聚类中心的距离平方和即为误差平方和 SSE 指标 I_{SSE}。

$$I_{\mathrm{SSE}}=\sum_{k=1}^{K}\sum_{\boldsymbol{x}\in\boldsymbol{X}_i}\left\|\boldsymbol{x}-\boldsymbol{c}_k\right\|^2 \tag{5}$$

式中，$\boldsymbol{c}_k$ 表示了类 $\boldsymbol{X}_k$ 的聚类中心。I_{SSE} 随着聚类数的增多而降低，且 I_{SSE} 越小则聚类得到的效果就越好，因此将 SSE 曲线拐点的聚类数定义为最佳聚类数。

(2) CHI 指标

CHI 指标考虑了类之间的分散性(用 D 表示)及各个类的内部紧凑性(用 T 表示)。

$$D=\sum_{k=1}^{K}\sum_{i=1}^{n}w_{k,i}\left\|\boldsymbol{c}_k-\bar{\boldsymbol{x}}\right\|^2 \tag{6}$$

$$T=\sum_{k=1}^{K}\sum_{i=1}^{n}w_{k,i}\left\|\boldsymbol{x}_i-\boldsymbol{c}_k\right\|^2 \tag{7}$$

式中：$\bar{\boldsymbol{x}}$ 为所有样本点的均值；$w_{k,i}$ 表示第 i 个样本点对第 k 个类的隶属关系。则 CHI 指标公式为

$$I_{\mathrm{CHI}}=\frac{D/(k-1)}{T/(n-k)} \tag{8}$$

显然，I_{CHI} 越大，类之间的分离程度和类中的紧凑程度就越优良，聚类质量越高。

(3) DBI 指标

DBI 指标 I_{DBI} 的计算公式为

$$I_{\mathrm{DBI}}=\frac{1}{K}\sum_{k=1}^{K}\max_{k\neq j}\frac{d(\boldsymbol{X}_k)+d(\boldsymbol{X}_j)}{\left\|\boldsymbol{c}_k-\boldsymbol{c}_j\right\|} \tag{9}$$

式中，$d(\boldsymbol{X})$ 为类 $\boldsymbol{X}$ 内样本间的平均距离。I_{DBI} 越小，则聚类后得到的结果就越优良。

3　算例分析

本文实验数据来源于某地区的实测数据，数据覆盖了 606 个用户，且每 15 min 采集一次，每日共 96 个采集点，采集时间为 30 天。

3.1 数据预处理

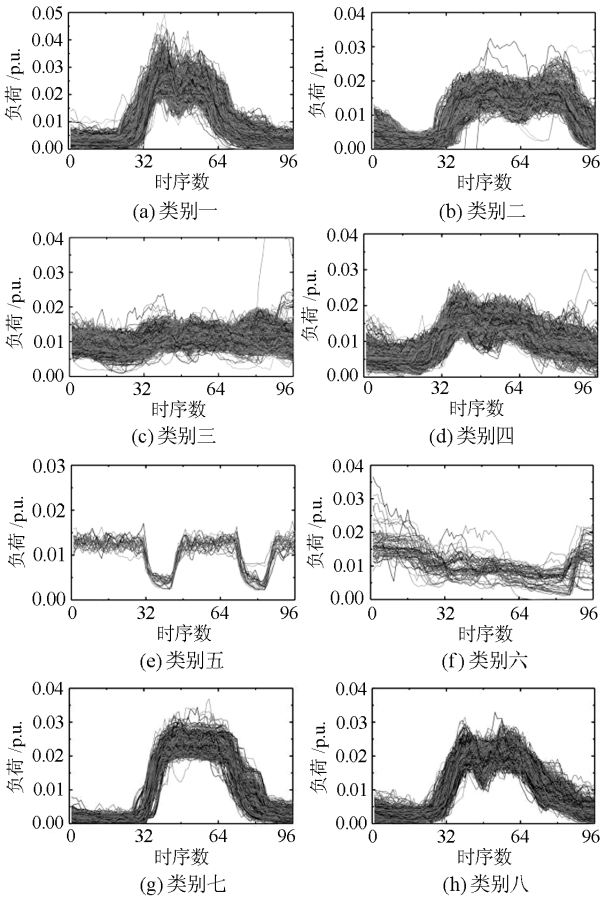
由于存在软硬件故障、通信中断和信号干扰等影响，负荷数据无法避免地会出现失真或缺失的情况。针对以上问题，本文剔除了其中数据不全的日负荷曲线，并采用高斯滤波对失真数据进行初步处理。对于高斯滤波未能处理的失真数据，本文采用 DBSCAN 算法分别对各个用户 30 天的负荷数据进行处理来剔除异常负荷曲线。此外，为了防止负荷曲线的幅值对聚类结果造成不良影响，本文对负荷曲线进行了归一化处理，归一化的方法为

$$x'(i)=\frac{x(i)}{\sum_{i=1}^{96}x(i)} \tag{10}$$

式中：$x(i)$ 表示处理前第 i 个采集点的功率；$x'(i)$ 表示处理后第 i 个采集点的平均功率。

3.2 聚类结果分析

对预处理后的数据进行分布式 QRD *K*-means 聚类，得到的聚类结果如图 4 所示。



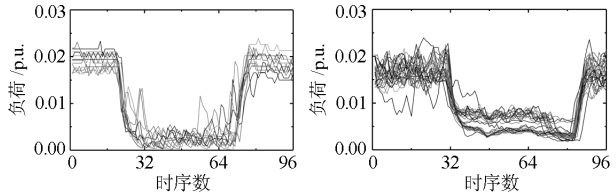


图 4 分布式 QRD *K*-means 算法聚类结果

Fig. 4 Clustering results of distributed QRD *K*-means algorithm

如图 4 所示，日负荷曲线被分为十类，且每一类均有明显的特征。其中，类别一、七、八中所包含的用户负荷在 8:00~16:00 时段(即时序数 32~64 时)为高峰期，其中类别一所包含的负荷用电量在 11:00 左右略有下降且在 16:00 左右下降快速；类别七中所包含的用户负荷用电量在高峰期较为平缓且在 16:00 左右下降快速；类别八则在 11:00 左右略有下降而在 16:00 左右下降缓慢。类别二中所包含用户的负荷在 10:00~20:00 时段(即时序数 40~80 时)为高峰期，且有两个明显的峰值。类别三中所包含的用户负荷整体较为平缓，没有明显的峰谷值。类别四中所包含的用户负荷用电量于 6:00 开始上升且在 10:00 左右达到峰值，之后逐渐下降。类别五中所包含的用户负荷用电量整体较为平稳，且在 10:00 左右与 20:00 左右处有明显的低谷。类别六中所包含的用户负荷用电量于每日 20:00 左右开始上升且在 24:00 左右达到峰值，次日整体呈现逐渐下降的趋势。类别九中所包含的用户负荷用电量在当日 16:00 左右至次日 6:00 左右为高峰期，其余时间为低谷期。类别十中所包含的用户负荷用电量在当日 20:00 左右至次日 8:00 左右为高峰期，其余时间为低谷期。

3.2.1 聚类有效性评价

本文共计采用了 SSE、CHI 和 DBI 三种指标对分布式 QRD *K*-means 聚类算法的有效性进行评价，并采用平滑处理后的数据进行指标计算，评价结果如图 5 所示。

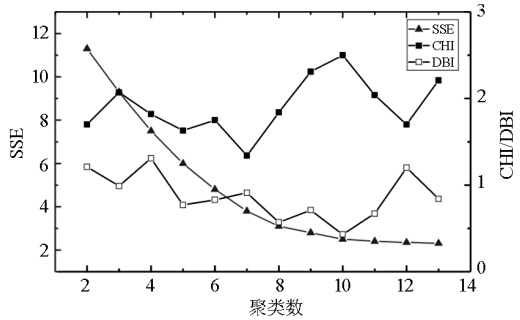


图 5 聚类数与有效性指标关系

Fig. 5 Relationships between cluster number and validity index

观察图 5 可以发现，当聚类数 k=10 时，SSE 指标曲线出现了明显的拐点，而 CHI 和 DBI 指标曲线分别达到了各自的极大值与极小值，与分布式 QRD *K*-means 算法聚类得到的结果类数相同，验证了该算法自动寻找 k 值的功能。

3.2.2 聚类结果比较

为验证分布式 QRD *K*-means 聚类算法的结果，对同一组预处理后的数据进行 k=10 的 *K*-means 聚类进行比较，*K*-means 聚类得到的聚类结果如图 6 所示。

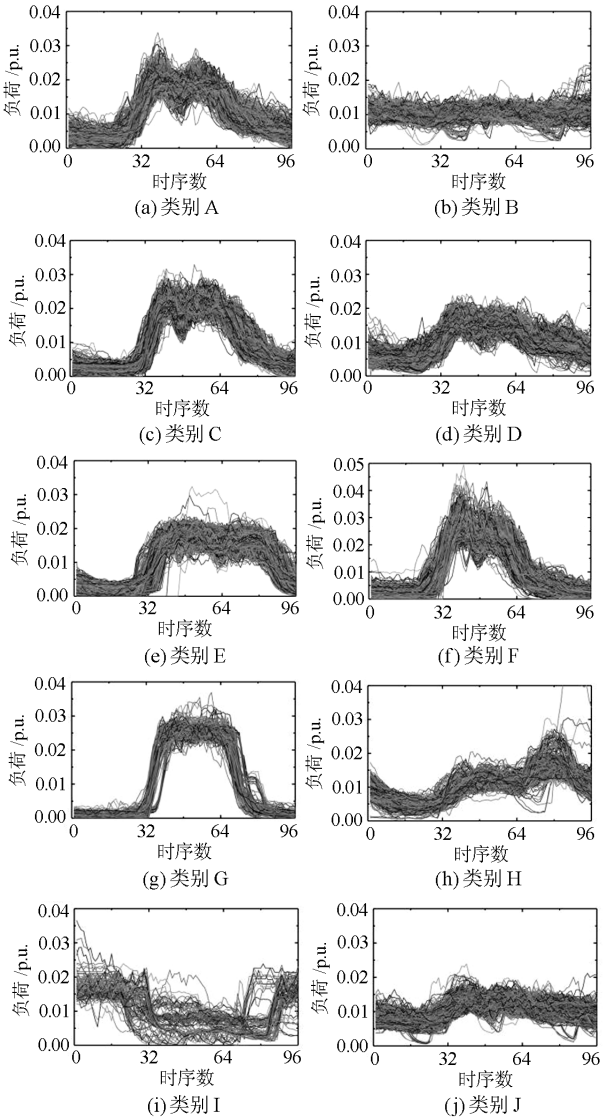


图 6 k=10 时 *K*-means 算法聚类结果

Fig. 6 Clustering results of *K*-means algorithm when k=10

如图 6 所示，日负荷曲线被分为十类。其中图 4 中的类别三被划分为形态相似的类别 B 和类别 J 两类；图 4 中的类别二被划分为类别 E 和类别 H 两类；而类别 C 对应了图 4 中类别八的一部分，类别 F 对应了图 4 中类别一的一部分，其余部分则被划

分进了类别 A；类别 D 与图 4 中的类别四相似；类别 G 与图 4 中的类别七相似；图 4 中的类别五、六、九、十这四类日负荷曲线则被划分进了类别 I。

显然，*K*-means 算法聚类得到各个簇中的样本点数量相对平均，这就意味着相互间形态差异较大，但样本数量相对较少的样本可能会被划分在同一类中。而分布式 QRD *K*-means 算法能够通过调整参数(即分位数半径)的设置使得聚类结果更为细致，可以更好地分辨样本数量较少的不同形态的日负荷曲线。

3.2.3 算法计算时间

为了验证算法的计算时间，分别使用分布式 QRD *K*-means 与 QRD *K*-means、层次 *K*-means、*K*-means 四种算法对同组数据进行 *p* 取值为 2~10 的聚类计算(*K*-means 与层次 *K*-means 中 *k* 的取值等于 *p*)，记录各自的计算时间，结果如图 7 所示。显然，*K*-means 算法计算所用时间最少，QRD *K*-means 与分布式 QRD *K*-means 算法的计算时间明显小于层次 *K*-means 算法的计算时间。随着参数 *p* 取值的增大，分布式 QRD *K*-means 算法耗时小于 QRD *K*-means 算法的耗时。

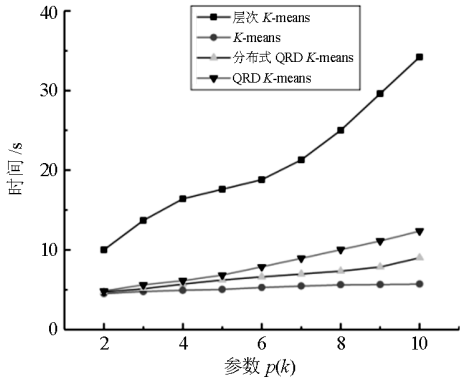


图 7 聚类算法的计算时间

Fig. 7 Computing time of clustering algorithms

3.2.4 *k* 值识别的准确性

如前文所述，该算法通过进行 *p* 次 *k* 取值为 2 到 *p*+1 的 *K*-means 计算后，根据所给出的中心点集合来识别 *k* 值。但是当实际的 *k* 值接近或大于 *p*+1 时，得到的中心点集合可能无法包含足够的数据信息，从而影响最终聚类结果。为了验证上述观点，对 *p* 进行多次不同取值，得到表 1。

表 1 对 *k* 值的识别情况

Table 1 Identification of *k*

p	2	3	4	5	6	7	8	9	10
k	2	2	4	5	6	6	7	9	10
p	11	12	13	14	15	16	17	18	19
k	10	10	10	12	10	10	15	10	10

由表 1 可以看出：当 *p* 的取值接近或小于实际的 *k* 值时，该算法 *k* 值的识别效果不佳，无法得到正确的聚类数；而当 *p* 的取值大于实际 *k* 值时，该算法对 *k* 值的识别效果有了明显提升。因此在实际情况中，可通过选取足够大的 *p* 值来获得较为准确的聚类数，但这同时会导致计算成本上升、计算时间增加。

4 结论

针对电力负荷曲线的特性，提出了一种基于分位数半径的动态 *K*-means 算法并进行分布式改进，将其应用于日负荷曲线的聚类分析中。算例结果表明，该算法具有良好的 *k* 值识别能力，而且能够得到较为理想的聚类初始中心点。通过各个中心点的间距与类半径对类进行筛选合并，在一定程度上降低了算法的计算时间，使其能够适用于大规模的日负荷曲线聚类分析。此外，结合分布式算法的思想所进行的分布式改进，进一步优化了算法的计算步骤，提高聚类效率。

值得注意的是，算法中所涉及的分位数半径选取问题仍需要被讨论，选取过大或过小都会对最终的聚类结果产生不良影响。当选取过小时，会发生过拟合，聚类得到的类数过大；当选取过大时，数量较少的样本则无法被有效识别并划分在某一类中。算例经过多次试验最终选取了较为平衡的第一四分位数(即 25%分位数半径)，使其能够可见地识别出部分数量较少的样本，但在样本数量较多的类的划分上没有明显的优势，这一点有待更深入的思考和研究。

参考文献

[1] HU Zhuangli, HE Tong, ZENG Yihui, et al. Fast image recognition of transmission tower based on big data[J]. Protection and Control of Modern Power Systems, 2018, 3(3): 149-158. DOI: 10.1186/s41601-018-0088-y.

[2] 李洁珊, 王朝硕, 章禹, 等. 基于历史信息挖掘的变压器健康状态聚类方法[J]. 电力系统保护与控制, 2018, 46(14): 94-99.
LI Jieshan, WANG Chaoshuo, ZHANG Yu, et al. A clustering method for transformer health state based on historical information mining[J]. Power System Protection and Control, 2018, 46(14): 94-99.

[3] 陈明照, 毛坚, 杜宗林, 等. 基于聚类法的工业用户需求侧管理(DSM)方案分析与研究[J]. 电力系统保护与控制, 2017, 45(7): 84-89.
CHEN Mingzhao, MAO Jian, DU Zonglin, et al. Analysis on demand side management scheme of industrial enterprise based on clustering method[J]. Power System Protection

and Control, 2017, 45(7): 84-89.

[4] 陈俊艺, 丁坚勇, 田世明, 等. 基于改进快速密度峰值算法的电力负荷曲线聚类分析[J]. 电力系统保护与控制, 2018, 46(20): 85-93.
CHEN Junyi, DING Jianyong, TIAN Shiming, et al. An improved density peaks clustering algorithm for power load profiles clustering analysis[J]. Power System Protection and Control, 2018, 46(20): 85-93.

[5] 陈俊艺, 丁坚勇, 田世明, 等. 基于改进快速密度峰值算法的电力负荷曲线聚类分析[J]. 电力系统保护与控制, 2018, 46(20): 85-93.
CHEN Junyi, DING Jianyong, TIAN Shiming, et al. An improved density peaks clustering algorithm for power load profiles clustering analysis[J]. Power System Protection and Control, 2018, 46(20): 85-93.

[6] CHICCO G, NAPOLI R, PIGLIONE F. Comparisons among clustering techniques for electricity customer classification[J]. IEEE Transactions on Power Systems, 2006, 21(2): 933-940.

[7] LI R, GU C H, LI F R, et al. Development of low voltage network templates--part I: substation clustering and classification[J]. IEEE Transactions on Power Systems, 2015, 30(6): 3036-3044.

[8] GULBINAS R, KHOSROWPOUR A, TAYLOR J. Segmentation and classification of commercial building occupants by energy-use efficiency and predictability[J]. IEEE Transactions on Smart Grid, 2015, 6(3): 1414-1424.

[9] 刘友波, 刘俊勇, 赵岩, 等. 基于多目标聚类的用电集群特征属性计算[J]. 电力系统自动化, 2009, 33(19): 46-51.
LIU Youbo, LIU Junyong, ZHAO Yan, et al. Calculation of characteristic attributes of consumer aggregations based on multi-objective clustering[J]. Automation of Electric Power Systems, 2009, 33(19): 46-51.

[10] RODRIGUEZ A, LAIO A. Clustering by fast search and find of density peaks[J]. Science, 2014, 344(6191): 1492-1496.

[11] ARTHUR D, VASSILVITSKII S. K-means++: the advantages of careful seeding[C] // Proceeding of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms, Pennsylvania, PA, USA: SIAM, 2007: 1027-1035.

[12] 张斌, 庄池杰, 胡军, 等. 结合降维技术的电力负荷曲线集成聚类算法[J]. 中国电机工程学报, 2015, 35(15): 3741-3749.
ZHANG Bin, ZHUANG Chijie, HU Jun, et al. Ensemble clustering algorithm combined with dimension reduction techniques for power load profiles[J]. Proceedings of the CSEE, 2015, 35(15): 3741-3749.

[13] HAMERLY G. Making k-means even faster[C] // Proceeding of the 2010 SIAM International Conference on Data Mining, 2010, Pennsylvania, PA, USA: 130-140.

[14] LIKAS A, VLASSIS N, VERBEEK J J. The global k-means clustering algorithm[J]. Pattern Recognition, 2003, 36(2): 451-461.

[15] MASHOR M Y. Hybrid training algorithm for RBF network[J]. International Journal of the Computer, the Internet and Management, 2000, 8(2): 50-65.

[16] PELLEG D, MOORE A W. X-means: extending K-means with efficient estimation of the number of clusters[C] // Proceedings of the 17th International Conference on Machine Learning, 2000, San Francisco, CA, USA: 727-734.

[17] 程明畅, 刘友波, 张程嘉, 等. 基于分位数半径的动态*K*-means 算法[J]. 南京大学学报, 2018, 54(1): 48-55.
CHENG Mingchang, LIU Youbo, ZHANG Chengjia, et al. Dynamic *K*-means algorithm based on quantile radius[J]. Journal of Nanjing University (Natural Science), 2018, 54(1): 48-55.

[18] 朱文俊, 王毅, 罗敏, 等. 面向海量用户用电特性感知的分布式聚类算法[J]. 电力系统自动化, 2016, 40(12): 21-27.
ZHU Wenjun, WANG Yi, LUO Min, et al. Distributed clustering algorithm for awareness of electricity consumption characteristics of massive consumers[J]. Automation of Electric Power Systems, 2016, 40(12): 21-27.

[19] 海沫, 张书云, 马燕林. 分布式环境中聚类问题算法研究综述[J]. 计算机应用研究, 2013, 30(9): 2561-2564.
HAI Mo, ZHANG Shuyun, MA Yanlin. Algorithm review of distributed clustering problem in distributed environments[J]. Application Research of Computers, 2013, 30(9): 2561-2564.

[20] MACQUEEN J B. Some methods for classification and analysis of multivariate observations[C] // Proceeding of the 5th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, 1967, CA, USA: 281-297.

[21] ARAI K, BARAKBAH A R. Hierarchical k-means: an algorithm for centroids initialization for K-means[R]. Japan: Saga University, 2007.

收稿日期：2019-01-31； 修回日期：2019-06-20

作者简介：

刘季昂(1995—)，男，硕士研究生，研究方向为电力系统大数据分析；E-mail: ymm_liujiang@163.com

刘友波(1983—)，男，博士，副教授，研究生导师，研究方向为电力系统数据挖掘与分析。

(编辑 许 威)