# 基于 Kernel *K*-means 的负荷曲线聚类

赵文清,龚亚强
(华北电力大学 控制与计算机工程学院,河北 保定 071003)

**摘要**:电力负荷曲线聚类是配用电系统的基础,对负荷管理具有重大意义。采用基于核方法的聚类算法提高负荷曲线聚类的准确性,通过点积的方式构造核矩阵,再将数据映射到高维空间中进行聚类,进而加大数据的可分性。同时,针对核矩阵的规模大、计算复杂的问题,提出使用核主成分与缩减矩阵规模对该方法进行优化。实验过程中采用美国能源部开发能源信息网站提供的负荷数据进行聚类,并以 Davies-Bouldin 聚类有效性指标评估效果。结果表明该方法具有较好的划分能力,可以提高负荷曲线聚类的准确性。

**关键词**:负荷曲线;聚类算法;核矩阵;核主成分分析;削减矩阵
**中图分类号**:TM 714 **文献标识码**:A DOI:10.16081/j.issn.1006-6047.2016.06.030

## 0 引言

电力负荷曲线聚类是配用电大数据挖掘的基础[1]。负荷曲线聚类一直是电力负荷预测、分时电价、错峰管理、系统规划的基础,通过负荷分类和负荷特性分析,对于掌握电力负荷的变化规律和发展趋势具有重大意义。高效的负荷聚类方法能为电力规划、错峰管理等提供可靠的依据和准确的指导[2]。因此,研究准确的负荷曲线聚类具有重要意义。

电力负荷曲线的聚类分析,实际上就是衡量不同负荷曲线的相似性,以及把负荷曲线分类到不同的簇中。这就需要根据负荷特性进行准确、科学分类,以确保在同一类中的负荷曲线具有相同或相似的负荷特性,进而合理地确定典型负荷曲线的归类。在良好分类的基础上,可以进一步提高负荷预测精度、降低负荷管理的难度。聚类的结果要满足类内具有较高的紧密性,类间具有较高的分离性,体现出不同类型用户之间的负荷特性差异[3]。

目前的电力负荷曲线聚类的方法很多,比较流行的有 *K*-means 聚类[4]、小波分析[5]、模糊 C 均值聚类算法(FCM)[6]、集成聚类算法[1]、自组织特征映射神经网络(SOM)[7]、极端学习机(ELM)[8]、云模型[9]等,同时还有一些在这些算法的基础上进行改进的算法。

由于智能电网技术的快速发展,各种先进的检测装置和计量设备在配电网中取得了广泛的应用。对负荷的检测的时间越来越短,导致负荷数据的维数大幅提高,加大了负荷曲线聚类的难度。为了解决该问题,本文采用核方法将数据映射到高维空间中,加大数据的可分性,从而提高负荷曲线聚类的效果。

收稿日期:2016-03-01;修回日期:2016-04-06
基金项目:中央高校基本科研业务费专项资金资助项目(12MS121)
Project supported by the Fundamental Research Funds for the Central Universities(12MS121)

## 1 Kernel *K*-means

### 1.1 核函数

核方法是将数据映射到高维空间中,从而使数据可分性在高维空间中增大,聚类的效果相应地得到提升[10]。核方法对映射到高维空间的维度并没有太多的限制,高维空间甚至可以是无限维的。

核函数为核方法提供了一种映射的方式。核函数通过点积的方式将数据在高维空间中的关系表示出来,降低了在高维空间中讨论数据映射的难度。下面将描述核函数是如何表示数据在高维空间的关系。

假设给定一组样本数据 $\boldsymbol{x}_1$、$\boldsymbol{x}_2$、…、$\boldsymbol{x}_n$,$\boldsymbol{x}_i \in \mathbf{R}^D$,即每个样本数据为 $D$ 维向量,映射方程 $\boldsymbol{\phi}(\boldsymbol{x})$ 将 $\boldsymbol{x}_i$ 从空间 $\mathbf{R}^D$ 映射到新空间 $\boldsymbol{Q}$,则核函数在新空间 $\boldsymbol{Q}$ 定义的点积为:

$$H(\boldsymbol{x}_i,\boldsymbol{x}_j)=\boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x}_i)\boldsymbol{\phi}(\boldsymbol{x}_j) \tag{1}$$

其中,无需知道变换函数 $\boldsymbol{\phi}(\boldsymbol{x})$ 的形式和参数,映射实际上是通过核函数 $H(\boldsymbol{x}_i,\boldsymbol{x}_j)$ 完成的,即只需要给定核函数的形式就能完成数据从低维空间到高维空间的映射。常用的核函数有:多项式核函数 $H(\boldsymbol{x}_i,\boldsymbol{x}_j)=(\boldsymbol{x}_i\cdot\boldsymbol{x}_j+1)^d$、高斯核函数 $H(\boldsymbol{x}_i,\boldsymbol{x}_j)=\exp(-r\|\boldsymbol{x}_i-\boldsymbol{x}_j\|^2)$、感知器核函数 $H(\boldsymbol{x}_i,\boldsymbol{x}_j)=\tanh[a(\boldsymbol{x}_i\cdot\boldsymbol{x}_j)+d]$。

通过上述方式,核函数可以轻易地将低维空间与高维空间联系起来。核函数方法可以和不同的算法相结合,形成多种不同的基于核函数技术的方法,而且这两部分的设计可以单独进行,并可以为不同的应用选择不同的核函数和算法。如 SVM(Support Vector Machine)、Kernel *K*-means 算法、核主成分分析 KPCA(Kernel Principal Component Analysis)等都是核函数与不同算法的结合。

### 1.2 算法描述

负荷曲线聚类的经典聚类分析方法有基于层次、基于划分、基于密度、基于模型等聚类算法,经典

聚类算法各有优缺点[1],本文为了进一步提高聚类划分的效果,采用 Kernel *K*-means 算法对负荷数据进行研究。

Kernel *K*-means 算法是 *K*-means 算法的推广,也可以看作是它的一般形式。在 *K*-means 算法中,点与点之间的相似性是用距离来衡量的,但在 Kernel *K*-means 算法中,用核函数代替距离的作用衡量相似性。这样就相当于将数据的整个距离结构改变,同时也可以看作是将数据映射到一个新的空间。Kernel *K*-means 算法描述如下。

设 $\boldsymbol{A}=\{\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_n\}$ 为 $n$ 个样本的数据集,划分 $\boldsymbol{A}$ 为 $K$ 类,$\boldsymbol{C}_k(k=1,2,\cdots,K)$ 代表一个聚类,$\boldsymbol{\delta}(\boldsymbol{x}_i,\boldsymbol{C}_k)$ 表示指示函数,对应各个样本的所属类别 $k$。$\boldsymbol{\phi}(\boldsymbol{x}_i)$ 表示 $\boldsymbol{x}_i(i=1,2,\cdots,n)$ 从空间 $\mathbf{R}^D$ 到新空间 $\boldsymbol{Q}$ 的变换,在核空间 $\boldsymbol{Q}$ 中,聚类 $\boldsymbol{C}_k$ 对应的类中心为 $\boldsymbol{m}_k$,公式如下:

$$\boldsymbol{m}_k=\frac{\sum_{\boldsymbol{x}_i\in\boldsymbol{C}_k}\boldsymbol{\phi}(\boldsymbol{x}_j)}{|\boldsymbol{C}_k|} \tag{2}$$

其中,$|\boldsymbol{C}_k|$ 表示聚类 $\boldsymbol{C}_k$ 中的样本个数。

空间 $\mathbf{R}^D$ 中任意 2 点间的距离通过核函数表示为内积的公式如下:

$$\begin{aligned}D(\boldsymbol{x}_i,\boldsymbol{x}_j)&=\|\boldsymbol{\phi}(\boldsymbol{x}_i)-\boldsymbol{\phi}(\boldsymbol{x}_j)\|^2=\\&\boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x}_i)\boldsymbol{\phi}(\boldsymbol{x}_i)-2\boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x}_i)\boldsymbol{\phi}(\boldsymbol{x}_j)+\boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x}_j)\boldsymbol{\phi}(\boldsymbol{x}_j)=\\&H(\boldsymbol{x}_i,\boldsymbol{x}_i)-2H(\boldsymbol{x}_i,\boldsymbol{x}_j)+H(\boldsymbol{x}_j,\boldsymbol{x}_j)\end{aligned} \tag{3}$$

其中,选取高斯核作为核函数 $H(\boldsymbol{x}_i,\boldsymbol{x}_j)$。

$$H(\boldsymbol{x}_i,\boldsymbol{x}_j)=\exp(-r\|\boldsymbol{x}_i-\boldsymbol{x}_j\|^2)$$

同理,类中每点 $\boldsymbol{x}_i$ 到类中心 $\boldsymbol{m}_k$ 的距离在新空间可表示为 $D(\boldsymbol{x}_i,\boldsymbol{m}_k)=\|\boldsymbol{\phi}(\boldsymbol{x}_i)-\boldsymbol{m}_k\|^2$。

$$\begin{aligned}D(\boldsymbol{x}_i,\boldsymbol{m}_k)&=\boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x}_i)\boldsymbol{\phi}(\boldsymbol{x}_i)-2\boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x}_i)\boldsymbol{m}_k+\boldsymbol{m}_k^{\mathrm{T}}\boldsymbol{m}_k=\\&H(\boldsymbol{x}_i,\boldsymbol{x}_i)-F(\boldsymbol{x}_i,\boldsymbol{C}_k)+G(\boldsymbol{C}_k)\end{aligned} \tag{4}$$

$$F(\boldsymbol{x}_i,\boldsymbol{C}_k)=-\frac{2}{|\boldsymbol{C}_k|}\sum_{\boldsymbol{\phi}(\boldsymbol{x}_l)\in\boldsymbol{C}_k}\boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x}_l)\boldsymbol{\phi}(\boldsymbol{x}_i) \tag{5}$$

$$G(\boldsymbol{C}_k)=\frac{1}{|\boldsymbol{C}_k|^2}\sum_{\boldsymbol{\phi}(\boldsymbol{x}_l)\in\boldsymbol{C}_k}\sum_{\boldsymbol{\phi}(\boldsymbol{x}_s)\in\boldsymbol{C}_k}\boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x}_l)\boldsymbol{\phi}(\boldsymbol{x}_s) \tag{6}$$

### 1.3 算法流程

Kernel *K*-means 描述如下:

**a.** 随机初始化分类向量 $\boldsymbol{\delta}(\boldsymbol{x}_i,\boldsymbol{C}_k)$,构造初始聚类 $\boldsymbol{C}_1$、$\boldsymbol{C}_2$、$\cdots$、$\boldsymbol{C}_K$;

**b.** 计算 $|\boldsymbol{C}_k|$ 与 $G(\boldsymbol{C}_k)$;

**c.** 计算 $F(\boldsymbol{x}_i,\boldsymbol{C}_k)$,将 $\boldsymbol{x}_i$ 分配到最近的聚类中;

$$\boldsymbol{\delta}(\boldsymbol{x}_i,\boldsymbol{C}_k)=\begin{cases}1 & D(\boldsymbol{x}_i,\boldsymbol{C}_k)<D(\boldsymbol{x}_i,\boldsymbol{C}_j)\\0 & \text{其他}\end{cases}$$

**d.** 重复步骤 **b**、**c**,直到 $\boldsymbol{\delta}(\boldsymbol{x}_i,\boldsymbol{C}_k)$ 不再变化。

## 2 算法优化

### 2.1 核主成分分析

主成分分析 PCA(Principal Component Analysis)是一种确定一个坐标系统的直交变换。在这个新的坐标系统下,变换数据点的方差沿新的坐标轴可以得到最大化。这些坐标轴经常被称为是主成分。PCA 运算是一个利用了数据集的统计性质的特征空间变换。这种变换在无损或很少损失数据集的信息的情况下降低了数据集的维数[12]。

KPCA 是线性 PCA 的非线性扩展算法,它采用非线性的方法抽取主成分,即 KPCA 是在通过映射函数把原始向量映射到高维空间 $\boldsymbol{F}$,在 $\boldsymbol{F}$ 上进行 PCA[12]。文献[13]提出一种大规模数据集求解核主成分的计算方法,该方法首先利用 Gram 矩阵构造一个 Gram-power 矩阵,然后将 Gram 矩阵的每一列作为每一步迭代算法的输入样本[13]。利用该方法可以有效解决传统方法在大规模数据集下无法使用的问题。

KPCA 与 PCA 具有本质上的区别:PCA 基于指标,而 KPCA 是基于样本的。KPCA 不仅适合解决非线性特征提取问题,而且它还能比 PCA 提供更多的特征数目和更多的特征质量。因为 KPCA 可提供的特征数目与输入样本的数目是相等的,而 PCA 的特征数目仅为输入样本的维数。KPCA 的优势是可以最大限度地抽取指标的信息,但是 KPCA 抽取指标的实际意义不是很明确[12]。

### 2.2 核矩阵的削减

从 Kernel *K*-means 算法中可以得出,比较样本与类中心距离大小需要计算 2 个样本的内积。本文中将数据集中任意 2 个样本间的内积计算出来,组成核矩阵[12]。核矩阵 $\boldsymbol{H}$ 为 $n\times n$ 阶矩阵,其中:

$$[\boldsymbol{H}]_{ij}=H(\boldsymbol{x}_i,\boldsymbol{x}_j) \tag{7}$$

$N$ 条负荷曲线构成数据集合。其中,每条负荷曲线都是由 $D$ 维的向量 $\boldsymbol{x}_n=\{\boldsymbol{x}_n^{(d)},d=1,2,\cdots,D\}$ 组成。如果要构造这样一个核矩阵,矩阵的储存空间将为 $O(N^2)$。当 $N$ 较大时,矩阵计算量将会很大[11]。文献[14-16]为了提高计算的效率,分别提出不同的解决策略。本文按照保留核矩阵中值较大的项、去除核矩阵中较小值的原则,再依据给定规则将核矩阵的某些项归零[11]。这样,可以减少计算时间,提高运算效率。这种对核矩阵削减的算法描述如下。

**a.** 将核矩阵 $\boldsymbol{H}$ 各行按升序进行排列,每行得到一个排序向量 $\boldsymbol{r}_i$:$r_{ij}(i=1,2,\cdots,n;j=1,2,\cdots,n)$。

**b.** 计算 $r_{ij}$ 的一阶导数,公式如下:

$$r'_{ij}=\frac{1}{3}\sum_{h=1}^{3}\frac{r_{i(j+h)}-r_{i(j-h)}}{2h}\quad j=4,5,\cdots,n-3 \tag{8}$$

**c.** 以降序方式排列 $r'_{ij}(j=1,2,\cdots,n)$。若 $r'_{ij}$ 为排序的前 10%,则令 $v_{ij}=1$;否则,$v_{ij}=0$。10%为所给定的阈值。得到 $\boldsymbol{r}'_i=[r'_{i1},r'_{i2},\cdots,r'_{in}]^{\mathrm{T}}$ 的二值化向量 $\boldsymbol{v}_i$,$\boldsymbol{v}_i=[v_{i1},v_{i2},\cdots,v_{in}]^{\mathrm{T}}$。

**d.** 将所得向量 $\boldsymbol{v}_i$ 求和,得到 $\boldsymbol{v}^*=\sum_{i=1}^{n}\boldsymbol{v}_i$,定义 $v_j^*$ 是

聚类基数 $j$ 所得权值。

**e.** 根据 $\boldsymbol{v}^*$ 计算得分向量 $\boldsymbol{s}$，公式如下：

$$s_j=(1-1/j)\max\left[\mathrm{e}^{-|(v_j^*-[v_j^*/j]j)/j|},\mathrm{e}^{-|(v_j^*-[v_j^*/j]j)/j|}\right] \quad (9)$$

**f.** 取 $\boldsymbol{s}$ 中最大值所属聚类 $j$，即 $w=\arg\max_{j=1}^{n}(s_j)$，定义：对任一数据样本 $\boldsymbol{a}_i$，其二值向量为 $\boldsymbol{v}_i$，如果 $v_{iw}=1$，则 $\boldsymbol{a}_i$ 属于基数为 $w$ 的聚类；否则不属于聚类 $w$。任一数据样本 $\boldsymbol{a}_i$，若使聚类基数 $w$ 值有所增加，必然属于聚类 $w$。

**g.** 令 $\boldsymbol{v}^*=\boldsymbol{v}^*-\boldsymbol{v}_i$，若 $\boldsymbol{v}^*\neq 0$，进行下一次迭代，重复步骤 **d**、**e**；若 $\boldsymbol{v}^*=0$，进入下一行，重复步骤 **b**—**g**。

**h.** 假设 $\boldsymbol{a}_i$ 所属的聚类基数为 $w_i$，在第 $i$ 行中，保留 $[\boldsymbol{H}]_{ij}$ 值中较大的前 $w_i$ 项，其余所有项全部设置为 0。削减后的核矩阵记为 $\boldsymbol{H}^*$，其中的非 0 项个数记为 $n_z$。

## 3　实验分析

### 3.1　聚类评估指标的选取

聚类分析是按照样本的特征将其分类到不同的类的过程，使同一类的个体具有尽可能高的相似性，而类别间则具有尽可能高的互异性。由于聚类分析是在没有先验信息指导下的无监督学习过程，因此评价聚类的效果在聚类分析中至关重要。

本文采用 Davies-Bouldin 指标评价聚类质量并确定最佳聚类数，其计算公式如下：

$$\mathrm{DB}=\frac{1}{K}\sum_{i=1}^{K}R_i \quad (10)$$

其中，$R_i$ 用来衡量第 $i$ 类与第 $j$ 类的相似度。

$$R_i=\max_{j\neq i}R_{ij},\quad R_{ij}=\max_{j=1,2,\cdots,K,i\neq j}\frac{S_i+S_j}{M_{ij}} \quad (11)$$

其中，$S_i$ 用来度量第 $i$ 个类中数据点的分散程度，计算公式如式(12)所示。

$$S_i=\frac{1}{T_i}\sum_{l=1}^{T_i}\|\boldsymbol{X}_l-\boldsymbol{A}_i\|_q=\left(\frac{1}{T_i}\sum_{l=1}^{T_i}|\boldsymbol{X}_l-\boldsymbol{A}_i|^q\right)^{1/q} \quad (12)$$

其中，$\boldsymbol{X}_l$ 为第 $i$ 类中第 $l$ 个数据点；$\boldsymbol{A}_i$ 为第 $i$ 类的中心；$T_i$ 为第 $i$ 类中数据点的个数；$q$ 取 1 时 $S_i$ 为各点到中心的距离的均值，$q$ 取 2 时 $S_i$ 为各点到中心的距离的标准差，它们都可以用来衡量类内分散程度。

$$M_{ij}=\|\boldsymbol{A}_i-\boldsymbol{A}_j\|_p=\left(\sum_{d=1}^{D}|a_{di}-a_{dj}|^p\right)^{1/p} \quad (13)$$

其中，$M_{ij}$ 为第 $i$ 类中心与第 $j$ 类中心的距离；$a_{di}$ 为第 $i$ 类的中心点 $\boldsymbol{A}_i$ 的第 $d$ 个属性的值；$p$ 取 1 时表示 1-范数，$p$ 取 2 时表示 2-范数(表示 2 个类中心的欧氏距离)。

DBI 是类内距离之和与类外距离的比值。类内对象距离越小，类间距离越大，DBI 指标也越小，聚类效果越好。DBI 也可以优化 $K$ 值的选择，最小的 DBI 指标对应的 $K$ 就是最佳聚类个数。

### 3.2　实验数据

实验数据[17]取自由美国能源部于 2009 年 12 月成立的 OpenEI(Open Energy Information)。OpenEI 是为政策制定者、研究人员、技术投资者、风险资本家及市场专业人士提供能源数据、信息等其他资源的网站。部分实验数据如表 1 所示。

**表 1　24 h 居民负荷数据**

Table 1　Hourly data of residential load

| 时间 | 用电负荷/(kW·h) | 时间 | 用电负荷/(kW·h) |
|---|---|---|---|
| 01:00:00 | 0.673 665 839 | 13:00:00 | 0.802 212 349 |
| 02:00:00 | 0.630 061 292 | 14:00:00 | 0.784 326 991 |
| 03:00:00 | 0.629 453 730 | 15:00:00 | 0.809 881 201 |
| 04:00:00 | 0.634 082 613 | 16:00:00 | 0.921 210 551 |
| 05:00:00 | 0.689 802 151 | 17:00:00 | 1.128 449 873 |
| 06:00:00 | 0.832 645 051 | 18:00:00 | 1.211 585 341 |
| 07:00:00 | 1.008 494 723 | 19:00:00 | 1.280 181 308 |
| 08:00:00 | 0.934 340 858 | 20:00:00 | 1.494 505 609 |
| 09:00:00 | 0.866 016 664 | 21:00:00 | 1.635 385 523 |
| 10:00:00 | 0.881 554 665 | 22:00:00 | 1.368 239 538 |
| 11:00:00 | 0.867 876 311 | 23:00:00 | 1.079 166 564 |
| 12:00:00 | 0.834 437 455 | 24:00:00 | 0.783 907 281 |

### 3.3　结果分析

实验采用的机器配置为 Intel(R)Core(TM)i3-3110M 8-core CPU@2.40 GHz，4 GB RAM，MATLAB 版本为 MATLAB R2014b。

首先考察 KPCA 对聚类效果的影响。当采用 KPCA 进行降维后，输出的维度对聚类效果有较大的影响，比如取输出维度为[1,30]时，聚类效果如图 1 所示。

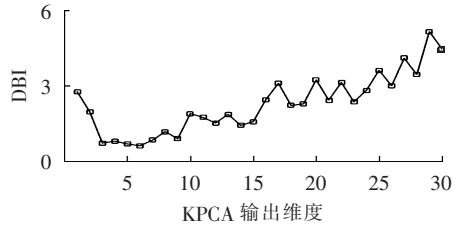


**图 1　输出维度对聚类效果的影响**

Fig.1　Effect of output dimension on clustering

由图 1 可知，输出维度的大小与聚类效果不是正比关系，当输出的维度太大或者太小，都不会得到最优聚类。当输出维度为 6 时，得到 DBI 的值最小，聚类的结果最为理想。因此核主成分分析取 6 作为输出维度。

实验再对聚类数与聚类效果进行分析，分别采用 3 种算法进行对比分析。第 1 种算法是传统的 *K*-means；第 2 种算法是 Kernel *K*-means，即采用了核方法的 *K*-means；第 3 种算法 KPCA-K-*K*-means 首先使用 KPCA 进行降维处理，从而得到降维后的核矩阵，再削减核矩阵，最后使用 *K*-means 进行聚类。利用 MATLAB 对上述 3 种算法编程，考察不同聚类数对

聚类效果的影响，*K*-means、Kernel *K*-means、KPCA-K-*K*-means 以聚类数作为输入，其中 Kernel *K*-means 与 KPCA-K-*K*-means 的核函数都选取高斯核函数，参数分别取 $r_1=-0.1$、$r_2=-1$。聚类数与 DBI 对应关系如表 2 所示。

**表 2 聚类数与 DBI 的关系**

Table 2 Relationship between clustering number and DBI

| 聚类数 | DBI | | |
|---|---|---|---|
| | *K*-means | Kernel *K*-means | KPCA-K-*K*-means |
| 2 | 0.65733 | 0.65733 | 0.66021 |
| 3 | 0.60279 | 0.58053 | 0.56706 |
| 4 | 0.61568 | 0.60053 | 0.54292 |
| 5 | 0.59403 | 0.58287 | 0.54969 |
| 6 | 0.62455 | 0.58053 | 0.54691 |
| 7 | 0.66069 | 0.62222 | 0.54691 |
| 8 | 0.65019 | 0.57792 | 0.54530 |
| 9 | 0.69217 | 0.58416 | 0.53699 |
| 10 | 0.74561 | 0.70116 | 0.54842 |
| 11 | 0.74713 | 0.59715 | 0.58268 |
| 12 | 0.73824 | 0.65476 | 0.61191 |
| 13 | 0.70649 | 0.68816 | 0.57334 |
| 14 | 0.74620 | 0.64809 | 0.58630 |
| 15 | 0.69840 | 0.72528 | 0.56436 |

由于 3 种算法的聚类效果都会受初始点的影响，因此对每个算法每一聚类数 *K*，分别运行 10 次，取最小的 DBI 值作为该算法的性能表现。表 2 对应的曲线图如图 2 所示。

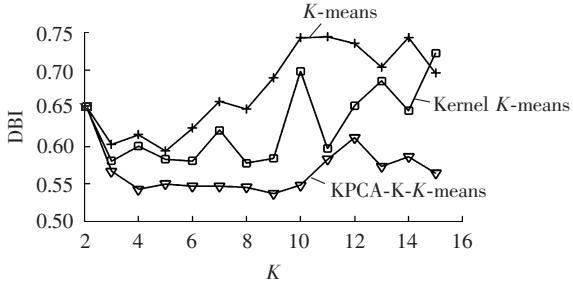


**图 2 聚类数与 DBI 的关系曲线**

Fig.2 Chart of relationship between clustering number and DBI

从图 2 可以看出，当聚类数取值相同时，各算法所得 DBI 值中，*K*-means 算法最大，Kernel *K*-means 次之，KPCA-K-*K*-means 最小。在聚类数的取值不同时，KPCA-K-*K*-means 取得的 DBI 值比 *K*-means、Kernel *K*-means 2 种算法得到的 DBI 值都要小，可以得出 KPCA-K-*K*-means 算法对负荷曲线的划分更加合理，可以提高聚类的准确度。从图中还可得出，当聚类数 $K=5$ 时，曲线具有明显的拐点，由此可以确定最佳聚类数。

## 4 结论

本文针对负荷数据出现的新特点，提出使用 KPCA 方法对负荷数据进行降维，同时保持数据在高维空间中的映射，使数据具有较好的可分性，然后根据聚类算法 Kernel *K*-means 的原理，对负荷曲线进行划分。实验探究了不同聚类数与聚类效果的关系以及输出维数对聚类效果的影响，表明本文方法可以有效地提高负荷曲线聚类的准确性。但该方法聚类易受聚类数和初始分类影响，需要提前确定核函数参数，以及运行时间增大等问题并没有完全解决，这也是进一步的研究方向。

## 参考文献：

[1] 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. Dimensionality reduction technique combined with power load curve integrated clustering algorithm[J]. Proceedings of the CSEE，2015，35(15)：3741-3749.

[2] 朱晓清. 电力负荷的分类方法及其应用[D]. 广州：华南理工大学，2012.
ZHU Xiaoqing. Classification of power load and its application [D]. Guangzhou：South China University of Technology，2012.

[3] 张忠华. 电力系统负荷分类研究[D]. 天津：天津大学，2007.
ZHANG Zhonghua. Load classification of power system [D]. Tianjin：Tianjin University，2007.

[4] 白雪峰，蒋国栋. 基于改进 *K*-means 聚类算法的负荷建模及应用[J]. 电力自动化设备，2010，30(7)：80-83.
BAI Xuefeng，JIANG Guodong. Load modeling based on improved *K*-means clustering algorithm and its application[J]. Electric Power Automation Equipment，2010，30(7)：80-83.

[5] 张平，潘学萍，薛文超. 基于小波分解模糊灰色聚类和 BP 神经网络的短期负荷预测[J]. 电力自动化设备，2012，32(11)：121-125，141.
ZHANG Ping，PAN Xueping，XUE Wenchao. Short-term load forecasting based on wavelet decomposition，fuzzy gray correlation clustering and BP neural network[J]. Electric Power Automation Equipment，2012，32(11)：121-125，141.

[6] 刘永光，孙超亮，牛贞贞，等. 改进型模糊 C 均值聚类算法的电力负荷特性分类技术研究[J]. 电测与仪表，2014，51(18)：5-9.
LIU Yongguang，SUN Chaoliang，NIU Zhenzhen，et al. Research on the improved fuzzy C-means clustering algorithm based power load characteristic classification technology[J]. Electrical Measurement & Instrumentation，2014，51(18)：5-9.

[7] 李智勇，吴晶莹，吴为麟，等. 基于自组织映射神经网络的电力用户负荷曲线聚类[J]. 电力系统自动化，2008，32(15)：66-70，78.
LI Zhiyong，WU Jingying，WU Weilin，et al. Power customers load profile clustering using the SOM neural network[J]. Automation of Electric Power Systems，2008，32(15)：66-70，78.

[8] 张少敏，赵硕，王保义，等. 基于云计算和量子粒子群算法的电力负荷曲线聚类算法研究[J]. 电力系统保护与控制，2014，42(21)：93-98.
ZHANG Shaomin，ZHAO Shuo，WANG Baoyi，et al. Research of power load curve clustering algorithm based on cloud computing and quantum particle swarm optimization[J]. Power System Protection and Control，2014，42(21)：93-98.

[9] 宋易阳，李存斌，祁之强. 基于云模型和模糊聚类的电力负荷模式提取方法[J]. 电网技术，2014，38(12)：3378-3383.
SONG Yiyang，LI Cunbin，QI Zhiqiang. Extraction of power load patterns based on cloud model and fuzzy clustering[J]. Power System Technology，2014，38(12)：3378-3383.

[10] TZORTZIS G F，LIKAS A C. The global Kernel *K*-means algorithm for cluster in feature space[J]. IEEE Transactions on Neural Networks，2009，20(7)：1181-1194.

[11] TSAPANOS N，TEFAS A，NIKOLAIDIS N，et al. A distributed framework for trimmed Kernel *K*-means clustering[J]. Pattern Recognition，2015，48(8)：2685-2698.

[12] 史卫亚，郭跃飞，薛向阳. 一种解决大规模数据集问题的核主成分分析算法[J]. 软件学报，2009，20(8)：2153-2159.
SHI Weiya，GUO Yuefei，XUE Xiangyang. Efficient kernel principal component analysis algorithm for large-scale data set[J]. Journal of Software，2009，20(8)：2153-2159.

[13] 胡中中. 图像信息多层次融合技术的研究[D]. 南昌：南昌大学，2012.
HU Zhongzhong. The research of multi-level fusion technology about image information[D]. Nanchang：Nanchang University，2012.

[14] ZHANG R，RUDNICKY A I. A large scale clustering scheme for Kernel *K*-means[C]//International Conference on Pattern Recognition. Quebec City，Canada：IEEE，2002：289-292.

[15] CHITTA R，JIN R，HAVENS T C，et al. Approximate Kernel *K*-means：solution to large scale kernel clustering[C]//ACM SIGKDD International Conference on Knowledge Discovery and Date Mining. San Diego，California，USA：[s.n.]，2011：895-903.

[16] SARMA T H，VISWANATH P，REDDY B E. A fast approximate Kernel *K*-means clustering method for large data sets[C]//Recent Advances in Intelligent Computational Systems(RAICS). Trivandrum，India：IEEE，2011：545-550.

[17] ERIC W. Commercial and residential hourly load profiles for all TMY3 locations in the United States[EB/OL]. (2013-07-02)[2016-04-25]. http://en.openei.org/datasets/dataset/commercialand-residential-hourly-load-profiles-for-all-tmy3-locations-in-the-united-states.

**作者简介：**

赵文清(1973—)，女，山西朔州人，教授，博士，主要研究方向为数据挖掘与智能技术在电力系统的应用(**E-mail**：jbzwq@126.com)。

## Load curve clustering based on Kernel *K*-means

ZHAO Wenqing，GONG Yaqiang

(School of Control and Computer Engineering，North China Electric Power University，Baoding 071003，China)

**Abstract**：As the basis of distribution and utilization system，the load curve clustering is of great significance for load management. A clustering algorithm based on the kernel method is proposed to improve the accuracy of the load curve clustering，which applies the dot product to construct the kernel matrix and maps the data into a high-dimensional space to increase the data divisibility for clustering. Aiming at the large scale and calculation complexity of the kernel matrix，the kernel principal component analysis and the kernel matrix size reduction are adopted to optimize the proposed method. As an experiment，the load data provided by the United State Department of Energy Development Energy Information Website are clustered and its effectiveness is assessed by the Davies-Bouldin index，which show that，the proposed method has better classification capability and the accuracy of load curve clustering is improved.

**Key words**：load curve；clustering algorithms；kernel matrix；kernel principal component analysis；matrix size reduction