Load Clustering Method Based on Improved Density Peak and Gaussian Mixture Model

Dong Zhe1,*, Wang Ya1, Ma Chuanxiao1, Ai Weihan1
1School of Electrical and Control Engineering, North China University of Technology, Beijing 100043, China
*email: dongzhe@ncut.edu.cn

Abstract. A load clustering method based on improved density peak and Gaussian mixture was proposed to solve the problems that standard Gaussian mixture clustering algorithm (GMM) with random initial clustering center was easy to fall into local optimum in the clustering analysis of residential electricity behavior, and the density peak algorithm (DPC) needed to manually select the clustering center. The method combines the Cosine distance and Euclidean distance, which reflect the similarity of curve shape, and selects the clustering center automatically through the sine value relation of relative Angle between two points in γ descending graph. The clustering center selected by the improved DPC algorithm was used to initialize the parameters of the GMM model. Finally, load clustering is carried out by GMM model. The results of calculation examples show that the proposed method can effectively improve the efficiency of daily load curve clustering.

1. Introduction

With the construction of smart grids, projects such as intelligent buildings and intelligent communities have emerged. Power companies can collect load data of power users by installing various data collection terminal equipment and setting different sampling periods [1]. The power data contains valuable information such as energy consumption characteristics. Reasonable analysis and use of power data can make power companies and power users a win-win situation [2].

Therefore, power load classification has gradually become a research focus. Load classification is divided into supervised classification and unsupervised classification. Supervised classification usually refers to artificial intelligence neural network method [3,4], and unsupervised classification usually refers to cluster analysis. Cluster analysis generally uses fuzzy C-Means clustering algorithm (FCM) [5,6], hierarchical clustering algorithm [7], K-MEANS clustering algorithm [8-11], Gaussian mixture model (GMM) [12] and other methods. Among them, GMM clustering belongs to "soft classification", which is not simply a sum of multiple Gaussian distributions, but the superposition of each Gaussian distribution after assigning a weight. In the end, each cluster is represented by a Gaussian mixture model composed of multiple Gaussian distributions, so GMM clustering has a more flexible cluster shape. However, the random initialization of the GMM clustering model parameters makes it easy to fall into the local optimum.

In GMM, the number of clusters needs to be manually determined and its parameters need to be initialized randomly, which makes it easy to fall into the problem of local optimization and high-dimensional data reducing the efficiency of the clustering algorithm. This paper proposes a load clustering method based on improved peak density and Gaussian mixture model for cluster analysis of daily load data.
2. Cluster analysis of typical electricity consumption behavior

2.1. Gaussian mixture model clustering
The Gaussian Mixture Model (GMM) is a generative model based on probability. It assumes that all samples in the data set are generated by a multivariate Gaussian distribution with given parameters. The process of clustering samples is actually to project each sample to each Gaussian mixture model to obtain the probability that each sample belongs to each cluster, select the Gaussian mixture model with the highest probability, and divide the sample into the cluster. For Gaussian mixture models with unknown parameters, expectation-maximization algorithm (EM) is usually used to estimate the parameters of GMM.

![Fig. 1 Gaussian mixture model](image)

Given the number of clusters K, for the sample x in the sample set, the probability density function of a GMM can be represented by a mixed distribution composed of K multivariate Gaussian distributions:

$$p(x) = \sum_{i=1}^{K} w_i \cdot p(x | \mu_i, \Sigma_i)$$  \hspace{1cm} (1)

Among them, \( p(x | \mu, \Sigma) \) is the mean vector of \( \mu \), the probability density function of the multivariate Gaussian distribution of the covariance matrix, \( w_i \) is the weight of the i-th multivariate Gaussian distribution in the mixture model, and \( \sum w_i = 1 \).

The calculation steps are as follows:
(1) Initialize the model parameters \( w_i, \mu_i, \Sigma_i \) of the Gaussian mixture distribution.
(2) E-Step: Calculate the posterior probability of the sample \( x_j \) generated by each mixed component, that is, the probability of the sample \( x_j \) generated by the i-th sub-model \( p(x_j = i | x_j) \) and mark it as \( \gamma_{ji} \), where \( z_j \) is the component in the mixed model, that is, a distribution.

$$\gamma_{ji} = \frac{w_i \cdot p(x_j | \mu_i, \Sigma_i)}{\sum_{i=1}^{K} w_i \cdot p(x_j | \mu_i, \Sigma_i)}$$  \hspace{1cm} (2)

(3) M-Step: Calculate new model parameters.

$$\mu_i = \frac{\sum_{j=1}^{m} \gamma_{ji} x_j}{\sum_{j=1}^{m} \gamma_{ji}}$$  \hspace{1cm} (3)

$$\Sigma_i = \frac{\sum_{j=1}^{m} \gamma_{ji} (x_j - \mu_i)'(x_j - \mu_i)}{\sum_{j=1}^{m} \gamma_{ji}}$$  \hspace{1cm} (4)

$$w_i = \frac{\sum_{j=1}^{m} \gamma_{ji}}{m}$$  \hspace{1cm} (5)

Where \( m \) is the total number of data in the data set.
(4) Repeat steps 2 and 3 according to the new model parameters until convergence.
(5) Divide each sample into corresponding clusters according to $\lambda_j = \arg \max_{i \in [1, 2, \cdots, K]} \gamma_{ij}$. That is, each sample will be assigned to the cluster of the sub-model with a high probability of coming from the sub-model, and finally K clusters are obtained.

2.2. DPC algorithm

Clustering by fast search and find of density peaks (DPC) algorithm is based on the assumption that the cluster center is surrounded by surrounding samples with low local density, and at the same time, the cluster center is at a relatively large distance from any other point with high local density, and the number of clustering clusters is determined according to the decision graph [13].

Given data set $D = \{x_1, x_2, \cdots, x_n\}$, for any sample $x_i$, DPC calculates two quantities: local density $\rho_i$ and minimum distance $\delta_i$. The local density $\rho_i$ of sample $x_i$ is defined as:

$$\rho_i = \sum_{j \neq i} \chi(d_{ij} - d_c)$$  \hspace{1cm} (6)

Among them, when $x < 0$, $\chi(x) = 1$, when $x \geq 0$, $\chi(x) = 0$; $d_{ij}$ represents the Euclidean distance between samples $x_i$ and $x_j$; $d_c$ represents the cut-off distance, which is generally considered to be within the range of $d_c$. The value of $d_c$ is appropriate when the average number of samples around each sample is 1%-2% of the total number of samples in the dataset.

Since the cut-off distance will affect the local density, which will affect the clustering result to a certain extent, the Gaussian kernel function is adopted to increase the local density:

$$\rho_i = \sum_{j \neq i} \exp \left( -\left( \frac{d_{ij}}{d_c} \right)^2 \right)$$  \hspace{1cm} (7)

The minimum distance $\delta_i$ of sample $x_i$ is defined as:

$$\delta_i = \begin{cases} \min_{i \neq j, i \neq i} (d_{ij}) & \text{else} \\ \max_i (d_{ij}) & \rho_i = \max(\rho) \end{cases}$$  \hspace{1cm} (8)

Calculate $\gamma_i = \rho_i \ast \delta_i, i \in [1, n], \gamma = [\gamma_1, \gamma_2, \cdots, \gamma_n]$, sort $\gamma$ in descending order and draw $\gamma$ descending order decision diagram. Finally, a point with a larger value of $\gamma$ is artificially selected from the decision diagram.

2.3. DPC algorithm based on automatic cluster center selection

The traditional DPC algorithm uses Euclidean distance to calculate the distance between samples. The Euclidean distance measures the absolute distance between each sample, which is strongly correlated with the position coordinates of each sample, and focuses on reflecting the absolute difference in numerical characteristics between samples. However, as the dimensionality increases, the metric of Euclidean distance will gradually deteriorate, and the Euclidean distance between two high-dimensional vectors with higher morphological similarity may be very large. Therefore, this article uses cosine similarity to correct the Euclidean distance to make the similarity difference of high-dimensional data more significant and improve the rationality of sample distance measurement in the traditional DPC algorithm.

2.3.1. Cosine distances. The cosine distance of the load is defined as $D_c = 1 - S_c$. $S_c$ is the cosine similarity of the load, that is, the cosine of the angle between the two vectors. The greater the cosine similarity, the more similar the two vectors, and the more attention is paid to the difference in orientation between the vectors, rather than the difference in value. The cosine similarity of the load is defined as:
$$S_c = \frac{\sum_{i=1}^{n} a_i b_i}{\sqrt{\sum_{i=1}^{n} a_i^2 \sum_{i=1}^{n} b_i^2}}$$  \quad (9)$$

Among them, $a_i$ is the i-th component in the load vector $A = [a_1, a_2, \ldots, a_n]$, $b_i$ is the i-th component in the load vector $B = [b_1, b_2, \ldots, b_n]$. The larger the cosine distance, the lower the similarity between the two load vectors.

2.3.2. Two-scale distance measurement. Aiming at the problems of Euclidean distance in high-dimensional vector distance measurement, this paper combines cosine distance and Euclidean distance, comprehensively considering the absolute distance between loads and the similarity of load shapes, and proposes a two-scale distance measurement method, which is defined as $D_{EC} = D_{2} \cdot D_{e}$. Among them, $D_{2}$ is the Euclidean distance between the two loads, and $D_{e}$ is the cosine distance between the two loads. The Euclidean distance between vectors will increase as the vector dimension increases, and the cosine distance has a fixed value range $[0, 2]$. The higher the similarity between the vectors, the closer the cosine distance is to 0, otherwise the cosine distance is closer to 2. Therefore, the cosine distance has the effect of scaling, reducing the Euclidean distance between loads with high morphological similarity, and making the difference in similarity between high-dimensional vectors more significant.

2.3.3. Automatic selection of cluster centers. The traditional DPC algorithm calculates $\gamma = p_i \cdot \delta_i$, draws a $\gamma$ descending decision diagram and artificially selects a point with a larger value of $\gamma$ as the cluster center. However, when the data set distribution is complex, the accuracy of clustering center selection will be greatly reduced due to the subjective factors. In response to this problem, this paper proposes an automatic selection of cluster centers based on the sine relationship of the relative angles between two points in a $\gamma$ descending graph.

Relative angle $\alpha_i$ is defined as the smaller angle between the line of the minimum point of point $\text{point}_i$, and $\gamma$ in the $\gamma$ descending decision diagram and the horizontal line, and $\gamma_{\Delta}$ is the descending sequence of $\gamma$. As can be seen from Figure 2, the greater the distance between point $\text{point}_i$, and point $\text{point}_{i+1}$ in the decision diagram, the greater the ratio of $\alpha_i$ to $\alpha_{i+1}$. And since the decision graph is a descending graph of $\gamma$, we know that there is $\alpha_i < \frac{\pi}{2}$, $\sin \alpha_i < 1$ for any i. According to the nature of the sine function between $[0, \frac{\pi}{2}]$, it is known that $\frac{\sin \alpha_i}{\sin \alpha_{i+1}} < \frac{\alpha_i}{\alpha_{i+1}}$, that is, the difference between $\alpha_i$ and $\alpha_{i+1}$ is greater than the difference between $\alpha_{i+1}$ and $\sin \alpha_{i+1}$. Therefore, it is more robust to select cluster centers based on the ratio of the sine values of the relative angles between adjacent points in the decision graph.

**Fig. 2** The relative angle between two points in the decision diagram
Calculate the corresponding $\sin \alpha$, for each point in $\gamma_{ik}$, to get $\gamma_{ik} = [\gamma_{1k}, \gamma_{2k}, \ldots, \gamma_{mk}] = [\sin \alpha_1, \sin \alpha_2, \ldots, \sin \alpha_m]$. For point point, in $\gamma_{ik}$, this article uses the following two steps to determine whether the sample is the cluster center:

Step 1: When $k = 1$, the sample $x_i$ corresponding to point $i$ is included in the cluster center candidate set candidates. When $k > 1$, if $\frac{\gamma_{i1}}{\gamma_{ik}} > 2$, then the sample $x_i, i \in [2,k]$ corresponding to point, $i \in [2,k]$ is included in the cluster center candidate set candidates. The remaining samples are directly classified as ordinary points.

Step 2: Calculate the distance between each data point in the cluster center candidate set candidates, and define $\delta_{min}$ as the distance between the two closest samples in the cluster center candidate set. For the sample $x_j$ in the candidate cluster center set candidates, when $j = 1$, $x_j$ is included in the cluster center set; when $0 < j < k$ and $\frac{\delta_j}{\delta_{min}} > 1.05$, $x_j$ is included in the cluster center set. When $j = k$ and $\frac{\delta_j}{\delta_{min}} > 0.8$, $x_j$ is included in the cluster center set. If the above conditions are not met, $x_j$ is classified as a normal point.

3. Clustering quality assessment

It is generally believed that high-quality clusters have the characteristics of "high similarity within clusters and low similarity between clusters". This paper uses Silhouette Coefficient, Calinski Harabasz Score, and Davies-Bouldin Index (DBI) as the evaluation indicators of clustering quality.

3.1. Silhouette Coefficient

The calculation formula of the SC score is:

$$s(i) = \frac{B(i) - A(i)}{\max[A(i), B(i)]} \quad (10)$$

Among them, $A(i)$ represents the average distance between data $i$ and other datasets in the same cluster; $B(i) = \min(B_1, B_2, \ldots, B_k)$, $B_j$ represents the average distance between data $i$ and all datasets in cluster $j$, $B(i)$ represents the dataset $i$ is the smallest distance among the average distances of all datasets from all other clusters. The range of $s(i)$ is [-1,1], and its value is positively correlated with the clustering result.

3.2. Calinski Harabasz Score

The formula for calculating CH score is:

$$\text{CH}(K) = \frac{\text{tr}(B(K-1)^{-1})}{\text{tr}(W(N-K)^{-1})} \quad (11)$$

Among them, $\text{tr}(B) = \sum_i j_i \left\| z_j - z_i \right\|^2$, $\text{tr}(B) = \sum_i j_i \sum_{x \in x_i} \left\| x_i - z_i \right\| ^2$; $z$ represents the mean value of all data in the dataset, $z_j$ is the mean value of the $j$th cluster, and $N$ is the cluster value. The number, $K$ is the current cluster. The value of $\text{CH}(K)$ is positively correlated with the clustering result.

3.3. Davies-Bouldin Index

The calculation formula of DBI is:

$$\text{DBI} = \frac{1}{N} \sum_{i=1}^{N} \max_{j \neq i} \left( \frac{d(X_i) + d(X_j)}{d(c_i, c_j)} \right) \quad (12)$$

Among them, $d(X_i)$ and $d(X_j)$ represent the sum of the average distances from all datasets in the cluster to the central sample of the cluster; $d(c_i, c_j)$ represents the distance between the central samples.
in two different clusters; The number of clustering categories is set to N. The value of DBI is negatively correlated with the clustering results.

4. Calculation example analysis
This paper selects the historical electricity load of some residents collected from a certain community in Beijing as the experimental data, and 500 data points are collected for each household every day. After preprocessing the original daily load data of users, the variation of daily load data of 249 households is shown in the figure.

Fig. 3 Daily load data curve

4.1. Analysis of clustering results
This paper uses the K-means algorithm, the traditional GMM algorithm, and the AD-DPC-GMM algorithm that automatically selects the cluster center based on the improved density peak to perform cluster analysis on the collected data, and the clustering results of these three methods Perform analysis and comparison to verify the rationality and superiority of the AD-DPC-GMM algorithm.

| Arithmetic | Number of clusters | SC   | CH     | DBI  |
|------------|--------------------|------|--------|------|
| K-means    | 3                  | 0.226| 113.370| 1.523|
| GMM        | 3                  | 0.237| 115.867| 1.518|
| AD-DPC-GMM | 3                  | 0.314| 116.058| 1.516|

Table 1 shows that the index values of AD-DPC-GMM algorithm are significantly improved compared with other clustering methods, indicating that in the clustering results of AD-DPC-GMM algorithm, the distance between clusters is relatively far and the distance within the clusters is relatively close. At the same time, the effectiveness of ad-DPC method to automatically search cluster centers and the improvement of AD-DPC on GMM clustering performance are also proved.

Fig. 4 γ descending decision diagram
Fig. 5 AD-DPC-GMM algorithm clustering results

Figure 4 shows the three cluster centers selected by the AD-DPC algorithm in the descending decision diagram. Figure 5 shows the clustering results of the AD-DPC-GMM algorithm on the daily load curve of 249 users, where the black curve represents the overall average of each type of load. In the clustering results, the number of load curves of each cluster is 80, 78, and 91, respectively. The
average load of these three types of users is analyzed: the peak power consumption of the first type of load occurs at 11:00–13:00 and 18:00–20:00, which are usually meal time and conform to the life rules of office workers from 9 to 5. The main peak of power consumption of the second type of load appears after 18:00, but there is a small peak of power consumption in the morning and afternoon, which is consistent with the characteristics of group tenants. The overall power consumption of the third type of load is lower than that of the first two types of load, and the end time of power consumption is earlier, which is consistent with the behavior of elderly households.

4.2. Two-scale EC distance validity proof

![Figure 6 Three user daily load curves](image)

| User       | Euclidean distance | Cosine distance | EC distance  |
|------------|--------------------|----------------|-------------|
| User1-User2| 11528.226          | 0.606          | 6986.105    |
| User1-User3| 11622.856          | 0.409          | 4753.748    |

The three curves in Figure 6 are the three user daily load curves in the data set used in this article. It can be seen from the figure that the shape similarity of the daily load curves of user 1 and user 3 is more similar than the shape similarity of the daily load curves of user 1 and user 2. high. However, by calculating the Euclidean distance between the three curves, it is found that the Euclidean distance between user 1 and user 3 is greater than the Euclidean distance between user 1 and user 2. From the Euclidean distance, the curve similarity between user 1 and user 2 is higher, which is inconsistent with the actual situation. This is because with the increase of the dimension, the Euclidean distance becomes larger and larger, which makes the Euclidean distance between the two curves with higher similarity become very large.

Multiply the Euclidean distance between the three curves by their corresponding cosine distances to obtain the dual-scale EC distance. As shown in Table 2, the dual-scale EC distance between user 1 and user 3 is significantly smaller than that between user 1 and user 2. The two-scale EC distance of is consistent with the actual situation, indicating that the measurement of the two-scale EC distance for the distance between high-dimensional data is significantly better than the Euclidean distance.

5. Conclusion

In this paper, a load clustering method based on improved density peak and Gaussian mixture is proposed, and the collected daily load data of users is taken as an example for cluster analysis. The experimental results prove the effectiveness of the algorithm to automatically select cluster centers, effectively avoiding the problem of the traditional GMM clustering algorithm easily falling into the local optimum, improving the clustering quality, and having better engineering feasibility.

Acknowledgments

This work was supported by National Key R&D Program Project "Research on the Prevention Technology of the Whole Chain of Food Toxic and Hazardous Substances in Major Events" (2018YFC1602703).
References

[1] Li Q 2019 An Efficient Power User Classification Method Based on Dimensionality Reduction and Load Clustering North China Electric Power University (Beijing)

[2] Wang H, Sun Y, Xu B 2019 Optimization of Clustering Analysis of Residential Electricity Consumption Behavior 2019 14th IEEE Conference on Industrial Electronics and Applications (ICIEA). IEEE

[3] Liu Y, Xu L 2018 High performance back propagation neural network algorithm for massive load data classification Automation of Electric Power Systems 42(21) pp 96-105

[4] Saraiva F, Bernardes W, Asada E N 2015 A framework for classification of non-linear loads in smart grids using Artificial Neural Networks and Multi-Agent Systems Neurocomputing 170 (dec.25) pp 328-338

[5] Liu Y, Sun C, Niu Z 2014 Research on the improved fuzzy c-means clustering algorithm based power load characteristic classification technology Electrical Measurement & Instrumentation 51(18) pp 5-9

[6] Zeng W 2018 Modified algorithm of image segmentation based on FCM ELECTRONIC MEASUREMENT TECHNOLOGY 37(03) pp 145-147 p 152

[7] Jia H, He G, Fang C 2007 Multi-level clustering method for hierarchical clustering and bidirectional capping combination for load forecasting Power System Technology 31(23) pp 33-36 (in Chinese)

[8] Zhao W, Gong Y 2016 Load curve clustering based on Kernel K-means Electric Power Automation Equipment 36(6) pp 203-207

[9] Xu Y, Li G, Guo K 2017 Research on parallel clustering of power load based on improved KMeans algorithm. Computer Engineering and Applications 53(17) pp 260-265

[10] Liu Q, Ruan Y, Shi X, Li Z 2018 Hourly Power Load Forecasting of Office Building based on Kmeans Clustering and BP Neural Network Algorithm JOURNAL OF ENGINEERING FOR THERMAL ENERGY AND POWER 33(03) pp 138-144

[11] Song J, Cui Y, LI X 2020 Load curve clustering method based on Euclidean-dynamic time warping distance and entropy weight method Automation of Electric Power Systems 44(15) pp 87-98

[12] Zang M, Li L, Yang X 2020 A Load Classification Method Based on Gaussian Mixture Model Clustering and Multi-dimensional Scaling Analysis Power System Technology 44(11) pp 4283-4296

[13] Rodriguez A, Laio A 2014 Clustering by fast search and find of density peaks Science 44(6191) pp 1492-1496