〈研究ノート〉

On Indices of Clustering Method

Yuusaku Kamura

Abstract Clustering is a fundamental and important method in data science. A large data set is categorized and divided into some subsets that each has a same property. To deal with subsets partitioned of an original data set makes handling data easier. But we are always confronted with the problems how many subsets are appropriate, and is the division good partition. Many indices have been proposed to evaluate the quality of the divided subsets. In this research, we focus on indices of clustering, especially for K-means method.

Keywords index of clustering; K-means method

1. Introduction

Clustering methods are effective for a data analysis. They are categorized into hierarchical and non hierarchical one. As a non hierarchical and unsupervised learning algorithm, K-means is the most famous and widely used.

When we use K-means, we need to give an appropriate K. But we can not guess which number K is suitable to make a good partition. Fortunately K-means does not require a long time for calculation, hence we can make subsets for different K by applying K-means again and again. However we need some criteria that show how well the partition.

In section 3 we show some simple methods to find the proper number K of a partition. In section 4 we report criteria that indicate goodness of a partition.

2. K-means method

At first, we show the problem that K-means method solves.

Problem 1

X : a set of n vectors $\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n$. $\boldsymbol{x}_i$'s dimension is m.
K : given. The number of subsets, that is, clusterings.
C_i : subsets divided of X. Each $\boldsymbol{x}_j$ belongs to exactly one C_i.
$\boldsymbol{\mu}_i$: mean value of $\boldsymbol{x}_j \in C_i$.

Define the squared error in C_i as follows:

$$f(C_i) = \sum_{\boldsymbol{x}_j \in C_i} \|\boldsymbol{x}_j - \boldsymbol{\mu}_i\|^2.$$

Then find a partition that minimizes the sum of the squared error for all i,

$$f(X) = \sum_{i=1}^{K} \sum_{\boldsymbol{x}_j \in C_i} \|\boldsymbol{x}_j - \boldsymbol{\mu}_i\|^2.$$

□

This problem is known to an NP-hard problem. K-means method is based on a greedy algorithm, hence it gives an approximation solution for Problem 1.

Many algorithms of K-means method have been proposed. They are essentially same. The differences are expectations to improve a computational complexity and to get a better solution.

K-means, outline

Step 1: Select an initial partition with K clusters. Repeat Step 2 and Step 3 until each C_i stabilizes.
Step 2: Make a new partition by assigning each pattern to its closest C_i's center.
Step 3: Compute the new C_is' centers.

3. The number of partitions

When we apply K-means to X, we have to give the number of partitions, K. We can not know which K is appropriate. In this section we show some methods to determine K. They do not give the correct K essentially.

Elbow method is well-known one. It is a primitive and old technique to find K.

A. *Rule of thumb*

It is known that

$$K \cong \sqrt{\frac{n}{2}}.$$

There are no theoretical grounds for this number.

B. *Based on distortion*

This is a theoretical method and highly effective for many problems. The procedure is based on distortion in cluster dispersion.

Step 1: Apply K-means using different numbers of clusters K. Then calcuate the distortions

$$\hat{d}_K = \frac{1}{m} \min_{\boldsymbol{c}_1,\ldots,\cdots,\boldsymbol{c}_K} E[(\boldsymbol{x}_i - \boldsymbol{c}_{\boldsymbol{x_i}})^T \Gamma^{-1} (\boldsymbol{x}_i - \boldsymbol{c}_{\boldsymbol{x_i}})]$$

for each K, where $\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_K$ are the center of K clusters and $\boldsymbol{c}_{\boldsymbol{x_i}}$ is the closest to $\boldsymbol{x}_i$. Γ is a covariance matrix.
Step 2: Select a transformation power $Y > 0$. $Y = p/2$ is a typical value.
Step 3: Calculate the “jumps”

$$J_K = \hat{d}_K^{-Y} - \hat{d}_{K-1}^{-Y}.$$

Step 4: Estimate the number of clusters in the dataset by $K^* = \arg\max_K J_K$. K^* is the largest jump and gives the value we seek.

C. *Elbow Method*

For $k = 2, 3, 4, \ldots$, solve Problem 1 by K-means. If $f(X)$’s value decreases sharply at some value k, such k is the value that we search and called an Elbow point. This procedure is simple and easy. But $f(X)$ do not always have an Elbow point. That is, if $f(X)$ decreases gradually, we can not identify such a point.

4. Indices

Many indices are proposed to evaluate the result of clustering. Most of indices are a measure of the compactness and separation of clusters. Here we enumerate them and make their definitions clear.

Notation

$X \ni \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n$: Vectors to be partitioned.
$d(\boldsymbol{x}, \boldsymbol{y})$: distance between $\boldsymbol{x}$ and $\boldsymbol{y}$.
$C_1, C_2, \ldots, C_m$: subsets of X by a clustering. Use $C(\boldsymbol{x}_i)$ to denote the subset that $\boldsymbol{x}_i$ belongs to.
$n_1, n_2, \cdots, n_m$: the number of points in $C_1, C_2, \cdots, C_m$, respectively.

A. *Silhouette index*

$$Sil = \frac{1}{n} \sum_{\boldsymbol{x}_i \in X} s(\boldsymbol{x}_i)$$

where

$a(i)$: mean of $d(\boldsymbol{x}_i, \boldsymbol{x}'_i)$ for $\boldsymbol{x}'_i \in C(\boldsymbol{x}_i)$,

$b(i)$: $\displaystyle\min_{C_k \neq C(\boldsymbol{x}_i)} \{\mathrm{mean}_{\boldsymbol{x}'_i \in C_k} d(\boldsymbol{x}_i, \boldsymbol{x}'_i)\}$,

$$s(\boldsymbol{x}_i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}.$$

A high value of Sil indicates a partion is good.

B. The C index

$$c = \frac{S_w - S_{\min}}{S_{\max} - S_{\min}}$$

where

S_w : sum of the within cluster distances. C_i has n_i points, hence there are $n_i(n_i-1)/2$ distinct pairs in C_i. Let $n_w = n_i(n_i-1)/2$.
$S_{\min}$: sum of the smallest n_w distances between all pairs of points in X. X has $n(n-1)/2$ distinct pairs
$S_{\max}$: sum of the greatest n_w distances between all pairs of points in X.

c statisfies $c \in [0,1]$. A low value indicates a partion is good.

C. The Baker-Hubert Gamma index

For two indieces i, i', we define $u_{ii'}$ as follows:

$u_{ii'} = 1$ if $\boldsymbol{x}_i$ and $\boldsymbol{x}_{i'}$ are in the same cluster,
$u_{ii'} = 0$ otherwise.

If

(i) $d(\boldsymbol{x}_i, \boldsymbol{x}'_i) < d(\boldsymbol{x}_j, \boldsymbol{x}'_j)$ and $u_{ii'} < u_{jj'}$
or
(ii) $d(\boldsymbol{x}_i, \boldsymbol{x}'_i) > d(\boldsymbol{x}_j, \boldsymbol{x}'_j)$ and $u_{ii'} > u_{jj'}$,

then we call a quadruple (i, i', j, j') concordant. On the other hand if

(iii) $d(\boldsymbol{x}_i, \boldsymbol{x}'_i) < d(\boldsymbol{x}_j, \boldsymbol{x}'_j)$ and $u_{ii'} > u_{jj'}$
or
(iv) $d(\boldsymbol{x}_i, \boldsymbol{x}'_i) > d(\boldsymbol{x}_j, \boldsymbol{x}'_j)$ and $u_{ii'} < u_{jj'}$,

then we call a quadruple (i, i', j, j') discordant.

We take quadruples (i, i', j, j') for all $\boldsymbol{x}_i \in X$. Then we count the concordants and the discordants.

The Baker-Hubert Gamma index is given as follow:

$$\Gamma = \frac{S^+ - S^-}{S^+ + S^-},$$

where

S^+ : the number of concordant quadruples,
S^- : the number of discordant quadruples.

Γ statisfies $\Gamma \in [-1,1]$. A high value indicates a partion is good.

D. Yule's index

For $\boldsymbol{x}_i, \boldsymbol{x}_j \in X, (i \neq j)$, take $d(\boldsymbol{x}_i, \boldsymbol{x}_j)$. We denote the number of $d(\boldsymbol{x}_i, \boldsymbol{x}_j)$ within same cluster by n_w and that of between clusters by n_b. $n_w + n_b = n(n-1)/2$.

We take n_w smallest $d(\boldsymbol{x}_i, \boldsymbol{x}_j)$, then we define a as the number of them within same cluster and b as between clusters. Similary we take n_b largest $d(\boldsymbol{x}_i, \boldsymbol{x}_j)$, then we define c as the number of them within same cluster and d as between clusters.

For the numbers a, b, c, d, Yule index is defined as follows:

$$yule = \frac{ad - bc}{ad + bc}.$$

A high value indicates a partion is good.

E. Dunn's index

We denote the miminal distance between points of different clusters by $d_{\min}$, and the largest distance within a cluster distance by $d_{\max}$.

The Dunn index is given as the quotient of $d_{\min}$ and $d_{\max}$:

$$dunn = \frac{d_{\min}}{d_{\max}}.$$

$dunn \in [0, \infty)$. Good partitions are indicated by high values of $dunn$.

F. Kendall's tau

This index is based on the quadruple counts as for Baker-Hubert Gamma index.

$$tau = \frac{S^+ - S^-}{N(N-1)/2},$$

where

S^+ : The number of concordant quadruples,

S^- : The number of discordant quadruples.

tau statisfies $tau \in [-1, +1]$. A high value indicates a g partition is good.

5. Conclusion

Firstly, we show some simple methods to find the appropriate number of subsets. Secondly, we report criteria to evaluate a partition.

Apply these methods and use criteria for sample data, then show effects of them is left for further studies.

References

[1] Fahad, A, Alshatri, N., Tari, Z., Alamri A., Khalil, I., Zomaya, A.Y., Foufou, S., Bouras, A : A Survey of Clustering Algorithms for Big Data: Taxonomy & Empirical Analysis. EMERGING TOPICS IN COMPUTING **2**, 267–279 (2014)

[2] Fraley, Chris, Raftery, Adrian E. : Model-Based Clustering, Discriminant Analysis, and Density Estimation. J. of the American Statical Association **97**, 611–631 (2002)

[3] Jain, Anil K. : Data clustering: 50 years beyond K-means. Pattern Recognition Letters. **31**,651-666, (2010)

[4] Kaufmann, Leonard, Rousseeuw, Peter J.: Clustering by Means of Medoids. Dodge, Y. (ed.),Statistical Data Analysis Based on the L1-Norm and Related Methods, North-Holland, 405–416 (1987)

[5] Kodinariya, Trupti M., Makwana, Prashant R.: Review on determining number of Cluster in K-Means Clustering. International J. of Advance Research in Computer Science and Management Studies.**1**,90–95 (2013)

[6] Roux, Maurice : Which indeces reveal the right number of cluster?, Private research paper (2005)

[7] Sugar, Catherine A., James, Gareth M.: Finding the Number of Clusters in a Dataset: An Information-Theoretic Approach. J. of the American Statistical Association. **98**, 750–763 (2003)

[8] Zhao,Wan-Lei, Deng,Cheng-Hao, Ngo, Chong-Wah : k-means: A revist. Neurocomputing **291**, 195–206 (2018)

Author

Yuusaku Kamura
Email: kamura.yuusaku@internet.ac.jp
Joint Research Laboratory, Tokyo Online University,
1-7-3, Nishi-Shinjuku, Shinjuku, Tokyo 160-0023, Japan