K-Medoids Clustering Algorithm

With Data Analysis

Kristina Pestaria Sinaga

Master in Information System Management
Bina Nusantara University

January 04, 2021

Summary

1 Why K-Medoids?

2 How Does K-Medoids Algorithm Works?

3 Case Analysis
- Vendor Segmentation
- Customer Segmentation

Why K-Medoids?

Illustrations

To explain why we need K-Medoids or why the concept of medoid over mean, let's seek an analogy.

Table: Example Problem of Credit Write-Off Prediction.

Customer	Balance	Age	Employed	Write-off
Mike	$200,000	42	No	Yes
Mary	$35,000	33	Yes	No
Claudio	$115,000	40	No	No
Robert	$29,000	23	Yes	Yes
Dora	$72,000	31	No	No

Centers of Clusters: Means

$$\bar{X}_{Bal} = \frac{\$200,000 + \$35,000 + \$115,000 + \$29,000 + \$72,000}{5} = \$90,200$$

$$\bar{X}_{Bal_2} = \frac{\$35,000 + \$115,000 + \$29,000 + \$72,000}{4} = \$62,750$$

Centers of Clusters: Means

$$\bar{X}_{Bal} = \frac{\$200,000 + \$35,000 + \$115,000 + \$29,000 + \$72,000}{5} = \$90,200$$

$$\bar{X}_{Bal_2} = \frac{\$35,000 + \$115,000 + \$29,000 + \$72,000}{4} = \$62,750$$

Clustering 1: K-Means

1. K-Means considers the center as $\$90,200$ which is greater than the 60% of individual balances in the Example
2. The mean excluding Mike's balance($\$200,000$) is $\$62,750$

Clustering 2: K-Medoids Procedure

1. K-Medoids is another kind of clustering algorithm. It uses another way to compute the centers.
2. Here's how to find Medoids as the center, sort the balance-feature data in ascending order, which gives $\$29,000, \$35,000, \$72,000, \$115,000, \$200,000$
3. For K-Medoids, we take each balance and compute its distance with the other balances. Then we select the balance with the minimum distance.
4. This time, we chose $\$72,000$ as the center. We call it a medoid. It is a better option in our case. How?

Center of Clusters: Medoids

$$|(\$35,000 - \$29,000)| + ... + |(\$200,000 - \$29,000)| = \$306,000$$

$$\Downarrow$$

$$|(\$29,000 - \$35,000)| + ... + |(\$200,000 - \$35,000)| = \$288,000$$

$$\Downarrow$$

$$|(\$29,000 - \$72,000)| + ... + |(\$200,000 - \$72,000)| = \$251,000$$

Center of Clusters: Medoids

$$|(\$35,000 - \$29,000)| + ... + |(\$200,000 - \$29,000)| = \$306,000$$

$$\Downarrow$$

$$|(\$29,000 - \$35,000)| + ... + |(\$200,000 - \$35,000)| = \$288,000$$

$$\Downarrow$$

$$|(\$29,000 - \$72,000)| + ... + |(\$200,000 - \$72,000)| = \$251,000$$

Center of Clusters: Medoids

$$|(\$35,000 - \$29,000)| + ... + |(\$200,000 - \$29,000)| = \$306,000$$

$$\Downarrow$$

$$|(\$29,000 - \$35,000)| + ... + |(\$200,000 - \$35,000)| = \$288,000$$

$$\Downarrow$$

$$|(\$29,000 - \$72,000)| + ... + |(\$200,000 - \$72,000)| = \$251,000$$

Center of Clusters: Medoids

$$|(\$29,000 - \$115,000)| + ... + |(\$200,000 - \$115,000)| = \$294,000$$

$$\Downarrow$$

$$|(\$29,000 - \$200,000)| + ... + |(\$115,000 - \$200,000)| = \$549,000$$

Center of Clusters: Medoids

$$|(\$29,000 - \$115,000)| + ... + |(\$200,000 - \$115,000)| = \$294,000$$

$$\Downarrow$$

$$|(\$29,000 - \$200,000)| + ... + |(\$115,000 - \$200,000)| = \$549,000$$

Clustering 2: K-Medoids

A medoid as a median is not sensitive to outliers. But a medoid is not a median.

Too hard, let's try with a simple example

1. Here's a list of weights of 5 students of our class, 62, 64, 65, 62, 120.
2. With our 5 students, K-means considers the center as 74.6

$$\frac{62+64+65+62+120}{5} = 74.6$$

which is greater than the 80% of individual weights in the population and what's funnier is that the mean excluding the last item(120) is 63.5,

$$\frac{62+64+65+62}{4} = 63.5$$

63.5 represents the set quite well. These values, which lye way further than the general distribution of the data points are called outliers and as seen above, the mean is worst affected by such outliers.

Now think of another measure: K-Medoids

1. Here's a list of weights of 5 students of our class after sort the data in ascending order, which gives 62, 62, 64, 65, 120
2. For K-Medoids, we take each weight and compute its distance with the other weights. Then we select the weight with the minimum distance.
3. Distance between student with weight 62 and the other students

$$(62 - 62) + (64 - 62) + (65 - 62) + (120 - 62) = 63$$

4. Distance between student with weight 64 and the other students

$$|(62 - 64)| + |(62 - 64)| + |(65 - 64)| + |(120 - 64)| = 61$$

Now think of another measure: K-Medoids

1. Distance between student with weight 65 and the other students

$$(62-65)+(62-65)+(64-65)+(120-65)=62$$

2. Distance between student with weight 120 and the other students

$$|(62-120)|+|(62-120)|+|(64-120)|+|(65-120)|=227$$

3. This time, we chose 64 as the center. We call it a medoid. It is a better option in our case.
4. Note that the medoids idea as centers almost accurately represents the data even at the presence of an outlier. Statistically, medoids are less prone to outliers which makes them a more reliable measure of center. Hopefully, this made sense to why K-Medoids are preferred over K-Means.

How Does K-Medoids Algorithm Works?

Step 1: Select Initial Medoids

Calculate the distance between every pair of all objects based on the chosen dissimilarity measure (Euclidean distance in our case).

$\Downarrow$

Calculate v_j for object j as follows:

$$v_j = \sum_{i=1}^{n} \frac{d_{ij}}{\sum_{l=1}^{n} d_{il}}, j = 1, ..., n$$

$\Downarrow$

Sort v_j's in ascending order. Select k objects having the first k smallest values as initial medoids.

Step 1: Select Initial Medoids

Calculate the distance between every pair of all objects based on the chosen dissimilarity measure (Euclidean distance in our case).

$$\Downarrow$$

Calculate v_j for object j as follows:

$$v_j = \sum_{i=1}^{n} \frac{d_{ij}}{\sum_{l=1}^{n} d_{il}}, j = 1, ..., n$$

$$\Downarrow$$

Sort v_j's in ascending order. Select k objects having the first k smallest values as initial medoids.

Step 1: Select Initial Medoids

Calculate the distance between every pair of all objects based on the chosen dissimilarity measure (Euclidean distance in our case).

$\Downarrow$

Calculate v_j for object j as follows:

$$v_j = \sum_{i=1}^{n} \frac{d_{ij}}{\sum_{l=1}^{n} d_{il}}, j = 1, ..., n$$

$\Downarrow$

Sort v_j's in ascending order. Select k objects having the first k smallest values as initial medoids.

Step 1: Select Initial Medoids

Obtain the initial cluster result by assigning each object to the nearest medoid.

$\Downarrow$

Calculate the sum of distances from all objects.

Step 1: Select Initial Medoids

Obtain the initial cluster result by assigning each object to the nearest medoid.

$$\Downarrow$$

Calculate the sum of distances from all objects.

Step 2: Update medoids

Find a new medoid of each cluster, which is the object minimizing the total distance to other objects in its cluster. Update the current medoid in each cluster by replacing with the new medoid.

Step 3: Assign objects to medoids

Assign each object to the nearest medoid and obtain the cluster result.

$\Downarrow$

Calculate the sum of distance from all objects to their medoids. If the sum is equal to the previous one, then stop the algorithm. Otherwise, go back to the Step 2

Step 3: Assign objects to medoids

Assign each object to the nearest medoid and obtain the cluster result.

$$\Downarrow$$

Calculate the sum of distance from all objects to their medoids. If the sum is equal to the previous one, then stop the algorithm. Otherwise, go back to the Step 2

Case Analysis
Vendor Segmentation

A Graphical Depiction of the MapReduce Process

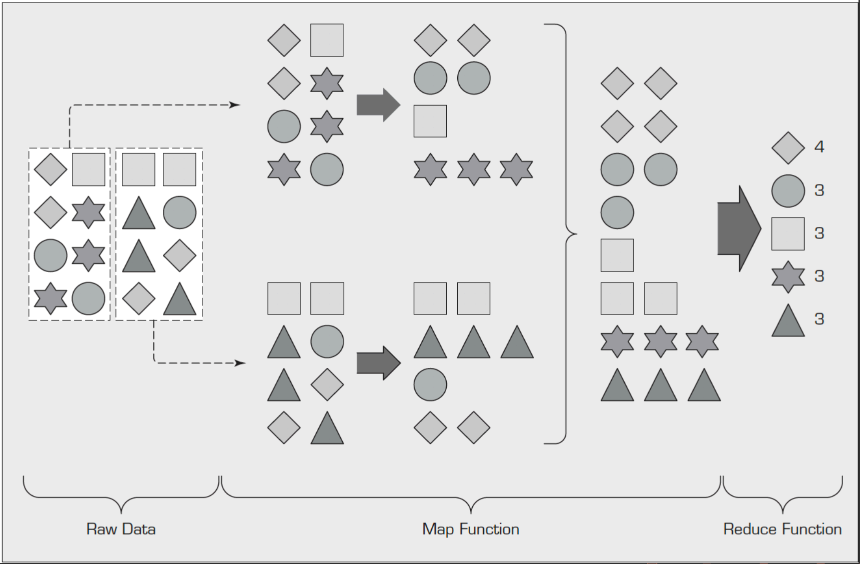
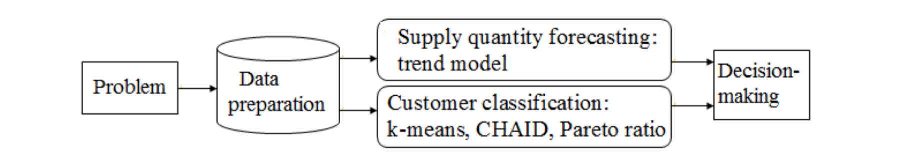


Table 2. Top 20 Vendors in Big Data Market

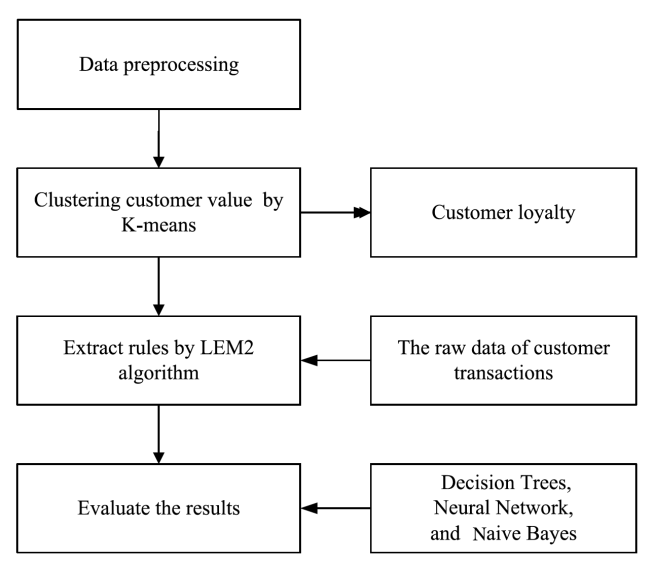
2012 Worldwide Big Data Revenue by Vendor ($US millions)

Vendor	Big Data Revenue	Total Revenue	Big Data Revenue as % of Total Revenue	% Big Data Hardware Revenue	% Big Data Software Revenue	% Big Data Services Revenue
IBM	$1,352	$103,930	1%	22%	33%	44%
HP	$664	$119,895	1%	34%	29%	38%
Teradata	$435	$2,665	16%	31%	28%	41%
Dell	$425	$59,878	1%	83%	0%	17%
Oracle	$415	$39,463	1%	25%	34%	41%
SAP	$368	$21,707	2%	0%	67%	33%
EMC	$336	$23,570	1%	24%	36%	39%
Cisco Systems	$214	$47,983	0%	80%	0%	20%
Microsoft	$196	$$71,474	0%	0%	67%	33%
Accenture	$194	$29,770	1%	0%	0%	100%
Fusion-io	$190	$439	43%	71%	0%	29%
PwC	$189	$31,500	1%	0%	0%	100%
SAS Institute	$187	$2,954	6%	0%	59%	41%
Splunk	$186	$186	100%	0%	71%	29%
Deloitte	$173	$31,300	1%	0%	0%	100%
Amazon	$170	$56,825	0%	0%	0%	100%
NetApp	$138	$6,454	2%	77%	0%	23%
Hitachi	$130	$112,318	0%	0%	0%	100%
Opera Solutions	$118	$118	100%	0%	0%	100%
Mu Sigma	$114	$114	100%	0%	0%	100%

Decision-making Framework



Research Model



Data Preparation

The following attributes are used as independent variables: “Big Data Revenue”, “Total Revenue”, “Big Data Revenue as % of Total Revenue”, “% Big Data Hardware Revenue”, “% Big Data Software Revenue”, “% Big Data Services Revenue”.

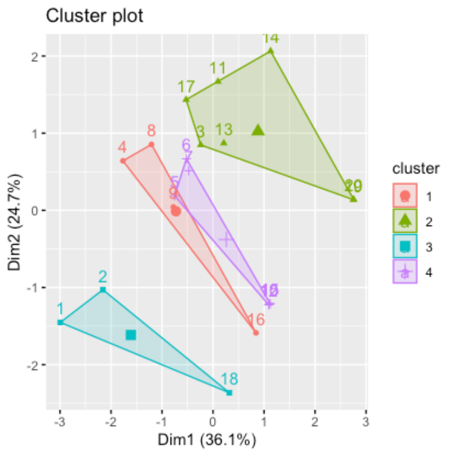
The results of clustering Two categories by K-Means.

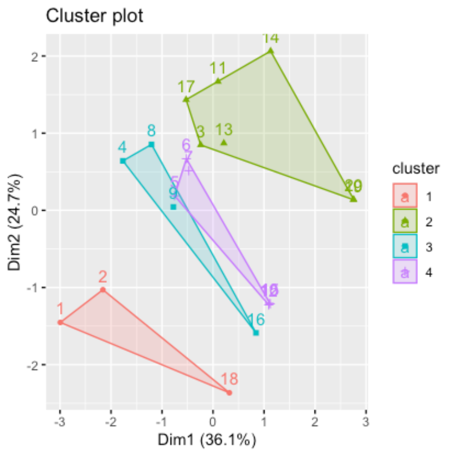
	Category 1	Category 2
Big Data Revenue	$489.50	$232.64
Total Revenue	$87,387	$17,016
Big Data Revenue as % of Total Revenue	0.0050	0.2671
% Big Data Hardware Revenue	0.2317	0.2200
% Big Data Software Revenue	0.2150	0.2107
% Big Data Services Revenue	0.5533	0.5686

The results of clustering Two categories by K-Means and K-Medoids (Original Data)

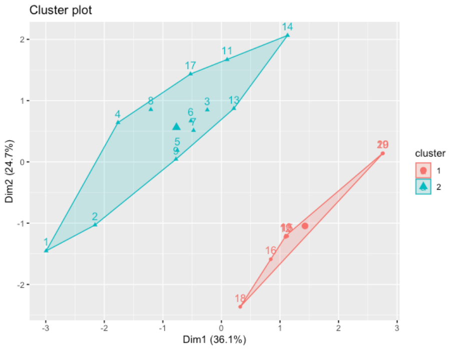
The results of clustering Three categories by K-Means and K-Medoids (Original Data)

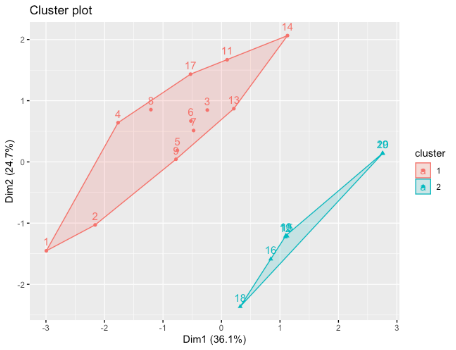
The results of clustering Four categories by K-Means and K-Medoids (Original Data)



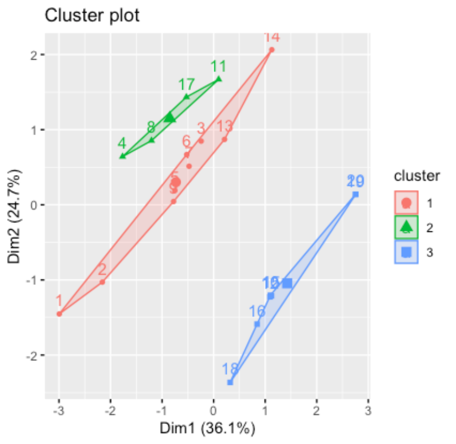


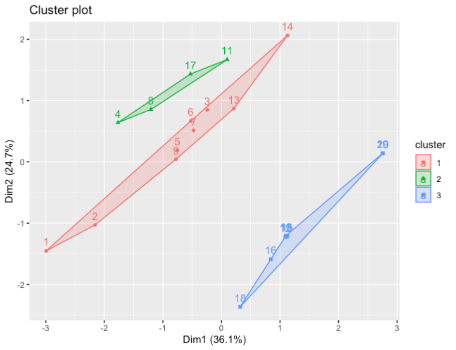
The results of clustering Two categories by K-Means and K-Medoids (Scale Data)



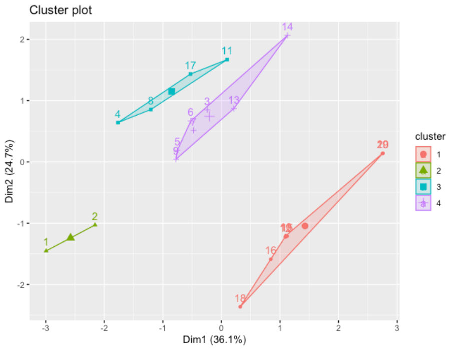


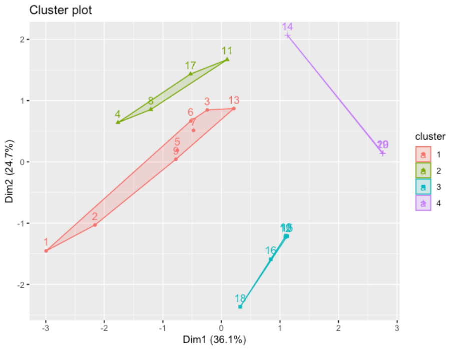
The results of clustering Three categories by K-Means and K-Medoids (Scale Data)





The results of clustering Four categories by K-Means and K-Medoids (Scale Data)



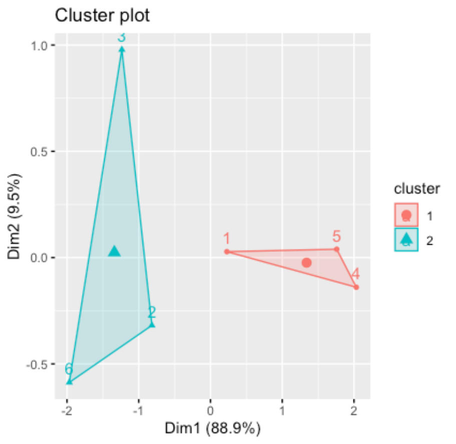


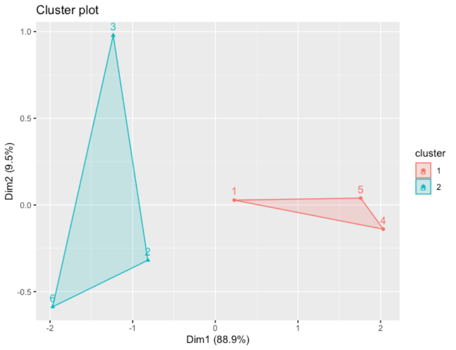
Case Analysis
Customer Segmentation

Table 3. Problem of Credit Write-Off Prediction

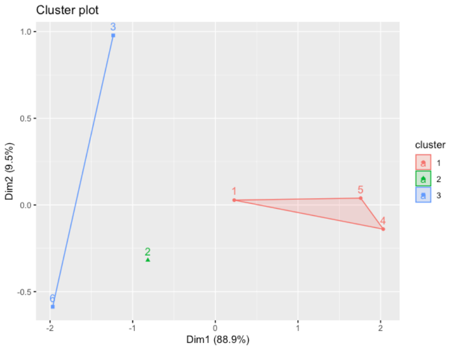
Customer	Age	Income ($1000)	Cards
David	37	50	2
John	35	35	3
Rachael	22	50	2
Ruth	63	200	1
Jefferson	59	170	1
Norah	25	40	4

The results of clustering Two categories by K-Means and K-Medoids (Original Data)





The results of clustering Three categories by K-Means and K-Medoids (Original Data)



References

- Asuncion, A. & Newman, D. *UCI machine learning repository.* 2007.
- Cheng, C.-H. & Chen, Y.-S. Classifying the segmentation of customer value via RFM model and RS theory. *Expert systems with applications* **36,** 4176–4184 (2009).
- Park, H.-S. & Jun, C.-H. A simple and fast algorithm for K-medoids clustering. *Expert systems with applications* **36,** 3336–3341 (2009).
- You, Z. *et al.* A decision-making framework for precision marketing. *Expert Systems with Applications* **42,** 3357–3367 (2015).

The End