# K-means Clustering

Mengta Chung, PhD

Department of Management Sciences
Tamkang University

# K-means

- An unsupervised learning algorithm
- A heuristic algorithm advanced by Stuart Lloyd of Bell Labs in 1957
- The term “k-means" was first used by James MacQueen in 1967.

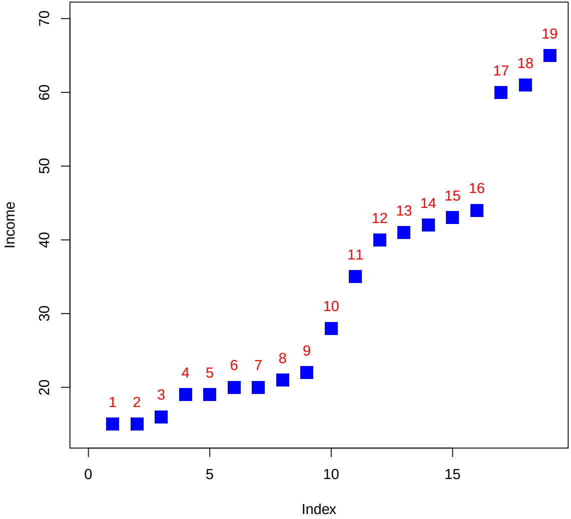
  MacQueen, J. Some methods for classification and analysis of multivariate observations. Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics, 281--297, University of California Press, Berkeley, Calif., 1967.

# Example 1 (One Attributes)

Suppose we wanna group visitors to a website using their ages:

15, 15, 16, 19, 19, 20, 20, 21, 22, 28, 35, 40, 41, 42, 43, 44, 60, 61, 65

# Plot



Income
Index
20
30
40
50
60
70
0
5
10
15
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19

# Algorithm (cluster data into K groups)

The algorithm uses an iterative refinement technique:

- Predefine K
- Select K points ~~(at random)~~ as cluster centers
- Assign points to their closest cluster center according to some distance function
- Calculate the centroid or mean of all points in each cluster
- Repeat above steps until convergence

# Initial Values

- Decide # of clusters
  - $k = 2$
- Select ~~(random)~~ initial centroids
  - $(C_1, C_2) = (16, 22)$
- Calculate the Manhattan distance (or other distance measures)
  - Let $A_i$ be the $i$th point of the data
  - $d_{i1} = |A_i - C_1|$
  - $d_{i2} = |A_i - C_2|$
- If $d_{i1} < d_{i2}$, then $A_i$ belongs to $C_1$

# (Sensible) Initial Values

The algorithm does not guarantee convergence to the global optimum. The result may depend on the initial clusters. As the algorithm is usually fast, it is common to run it multiple times with different starting conditions.

# Iteration 1

| $A$ | $C_1$ | $C_2$ | $d_{i1}$ | $d_{i2}$ | Nearest Cluster | New Centroid |
|---|---|---|---|---|---|---|
| 15 | 16 | 22 | 1 | 7 | 1 | $\frac{(15+15+16)}{3} = 15.33$ |
| 15 | | | 1 | 7 | 1 | |
| 16 | | | 0 | 6 | 1 | |
| 19 | | | 3 | 3 | 2 | $\frac{(19+\cdots+65)}{16} = 36.25$ |
| 19 | | | 3 | 3 | 2 | |
| 20 | | | 4 | 2 | 2 | |
| 20 | | | 4 | 2 | 2 | |
| 21 | | | 5 | 1 | 2 | |
| 22 | | | 6 | 0 | 2 | |
| 28 | | | 12 | 6 | 2 | |
| 35 | | | 19 | 13 | 2 | |
| 40 | | | 24 | 18 | 2 | |
| 41 | | | 25 | 19 | 2 | |
| 42 | | | 26 | 20 | 2 | |
| 43 | | | 27 | 21 | 2 | |
| 44 | | | 28 | 22 | 2 | |
| 60 | | | 44 | 38 | 2 | |
| 61 | | | 45 | 39 | 2 | |
| 65 | | | 49 | 43 | 2 | |

# Iteration 2

| $A$ | $C_1$ | $C_2$ | $d_{i1}$ | $d_{i2}$ | Nearest Cluster | New Centroid |
|---|---|---|---|---|---|---|
| 15 | 15.33 | 36.25 | 0.33 | 21.25 | 1 | $\frac{(15+\cdots+22)}{9} = 18.56$ |
| 15 | | | 0.33 | 21.25 | 1 | |
| 16 | | | 0.67 | 20.25 | 1 | |
| 19 | | | 3.67 | 17.25 | 1 | |
| 19 | | | 3.67 | 17.25 | 1 | |
| 20 | | | 4.67 | 16.25 | 1 | |
| 20 | | | 4.67 | 16.25 | 1 | |
| 21 | | | 5.67 | 15.25 | 1 | |
| 22 | | | 6.67 | 14.25 | 1 | |
| 28 | | | 12.67 | 8.25 | 2 | $\frac{(28+\cdots+65)}{10} = 45.9$ |
| 35 | | | 19.67 | 1.25 | 2 | |
| 40 | | | 24.67 | 3.75 | 2 | |
| 41 | | | 25.67 | 4.75 | 2 | |
| 42 | | | 26.67 | 5.75 | 2 | |
| 43 | | | 27.67 | 6.75 | 2 | |
| 44 | | | 28.67 | 7.75 | 2 | |
| 60 | | | 44.67 | 23.75 | 2 | |
| 61 | | | 45.67 | 24.75 | 2 | |
| 65 | | | 49.67 | 28.75 | 2 | |

# Iteration 3

| $A$ | $C_1$ | $C_2$ | $d_{i1}$ | $d_{i2}$ | Nearest Cluster | New Centroid |
|---|---|---|---|---|---|---|
| 15 | 18.56 | 45.9 | 3.56 | 30.9 | 1 | $\frac{(15+\cdots+28)}{10} = 19.5$ |
| 15 | | | 3.56 | 30.9 | 1 | |
| 16 | | | 2.56 | 29.9 | 1 | |
| 19 | | | 0.44 | 26.9 | 1 | |
| 19 | | | 0.44 | 26.9 | 1 | |
| 20 | | | 1.44 | 25.9 | 1 | |
| 20 | | | 1.44 | 25.9 | 1 | |
| 21 | | | 2.44 | 24.9 | 1 | |
| 22 | | | 3.44 | 23.9 | 1 | |
| 28 | | | 9.44 | 17.9 | 1 | |
| 35 | | | 16.44 | 10.9 | 2 | $\frac{(35+\cdots+65)}{9} = 47.89$ |
| 40 | | | 21.44 | 5.9 | 2 | |
| 41 | | | 22.44 | 4.9 | 2 | |
| 42 | | | 23.44 | 3.9 | 2 | |
| 43 | | | 24.44 | 2.9 | 2 | |
| 44 | | | 25.44 | 1.9 | 2 | |
| 60 | | | 41.44 | 14.1 | 2 | |
| 61 | | | 42.44 | 15.1 | 2 | |
| 65 | | | 46.44 | 19.1 | 2 | |

# Iteration 4

| $A$ | $C_1$ | $C_2$ | $d_{i1}$ | $d_{i2}$ | Nearest Cluster | New Centroid |
|---|---|---|---|---|---|---|
| 15 | 19.5 | 47.89 | 4.5 | 32.89 | 1 | $\frac{(15+\cdots+28)}{10} = 19.5$ |
| 15 | | | 4.5 | 32.89 | 1 | |
| 16 | | | 3.5 | 31.89 | 1 | |
| 19 | | | 0.5 | 28.89 | 1 | |
| 19 | | | 0.5 | 28.89 | 1 | |
| 20 | | | 0.5 | 27.89 | 1 | |
| 20 | | | 0.5 | 27.89 | 1 | |
| 21 | | | 1.5 | 26.89 | 1 | |
| 22 | | | 2.5 | 25.89 | 1 | |
| 28 | | | 8.5 | 19.89 | 1 | |
| 35 | | | 15.5 | 12.89 | 2 | $\frac{(35+\cdots+65)}{9} = 47.89$ |
| 40 | | | 20.5 | 7.89 | 2 | |
| 41 | | | 21.5 | 6.89 | 2 | |
| 42 | | | 22.5 | 5.89 | 2 | |
| 43 | | | 23.5 | 4.89 | 2 | |
| 44 | | | 24.5 | 3.89 | 2 | |
| 60 | | | 40.5 | 12.11 | 2 | |
| 61 | | | 41.5 | 13.11 | 2 | |
| 65 | | | 45.5 | 17.11 | 2 | |

# Results

- 2 centroids are 19.5 and 47.89 (do not update at iteration 3).
- 2 clusters have been identified (15-28) and (35-65).

# please note ...

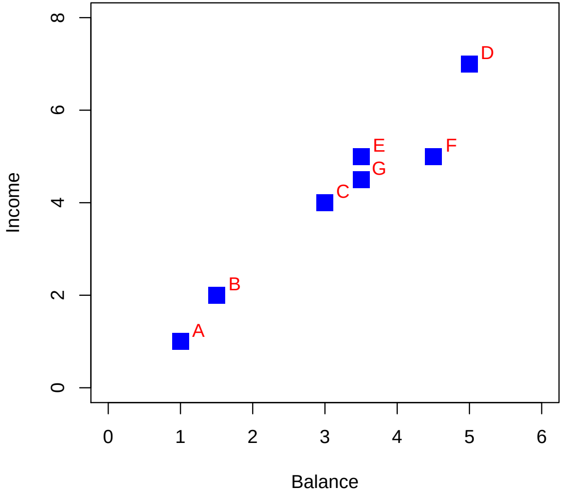
- The initial choice of centroids can affect the outcome, so sensible initial values are important.
- The algorithm is often run multiple times with different initial values in order to get a fair view of what the clusters should be.

# Example 2 (Two Attributes)

Cluster 7 people into two groups using balance and income:

| # | Balance | Income |
|---|---|---|
| A | 1 | 1 |
| B | 1.5 | 2 |
| C | 3 | 4 |
| D | 5 | 7 |
| E | 3.5 | 5 |
| F | 4.5 | 5 |
| G | 3.5 | 4.5 |

# Plot



# Initial Values

- Use the 2 subjects with longest Manhattan distance as 2 centroids
  - $C_1 : (1, 1)$ (subject A)
  - $C_2 : (5, 7)$ (subject D)
- Initial centroids do not have to be selected from data points.

# Iteration 1

$$d_{(1,1)\to C_2} = |1-5| + |1-7| = 10$$

| $(B,I)$ | $d_{(B,I)\to C_1:(1,1)}$ | $d_{(B,I)\to C_2:(5,7)}$ | Cluster | New $C$ |
|---|---|---|---|---|
| $(1,1)$ | 0 | 10 | 1 | $(1.25, 1.5)$ |
| $(1.5,2)$ | 1.5 | 8.5 | 1 | |
| $(3,4)$ | 5 | 5 | 2 | $(3.9, 5.1)$ |
| $(5,7)$ | 10 | 0 | 2 | |
| $(3.5,5)$ | 6.5 | 3.5 | 2 | |
| $(4.5,5)$ | 7.5 | 2.5 | 2 | |
| $(3.5,4.5)$ | 6 | 4 | 2 | |

New centroids:

- New $C_1 = \left(\frac{1+1.5}{2} = 1.25, \frac{1+2}{2} = 1.5\right)$
- New $C_2 = \left(\frac{3+\cdots+3.5}{5} = 3.9, \frac{4+\cdots+4.5}{5} = 5.1\right)$

# Iteration 2

| $(A, B)$ | $d_{(A,B)\to C_1:(1.25,1.5)}$ | $d_{(A,B)\to C_2:(3.9,5.1)}$ | Cluster | New $C$ |
|---|---|---|---|---|
| $(1, 1)$ | 0.75 | 7 | 1 | $(1.25, 1.5)$ |
| $(1.5, 2)$ | 0.75 | 5.5 | 1 | |
| $(3, 4)$ | 4.25 | 2 | 2 | $(3.9, 5.1)$ |
| $(5, 7)$ | 9.25 | 3 | 2 | |
| $(3.5, 5)$ | 5.75 | 0.5 | 2 | |
| $(4.5, 5)$ | 6.75 | 0.7 | 2 | |
| $(3.5, 4.5)$ | 5.25 | 1 | 2 | |

- New $C_1 = \left(\frac{1+1.5}{2} = 1.25, \frac{1+2}{2} = 1.5\right)$
- New $C_2 = \left(\frac{3+\cdots+3.5}{5} = 3.9, \frac{4+\cdots+4.5}{5} = 5.1\right)$
- The algorithm converges after 2 iterations.

# Advantage of K-means over HC

Hierarchical clustering requires the computation and storage of an $N \times N$ distance matrix. For a very large dataset, this can be expensive and slow.

# Lab

Analyze HC.csv using K-means clustering in Python