Clustering

- Given:
  - a set of examples
  - in some description language (e.g., attribute-value)
  - no labels (-> unsupervised)
- Find:
  - a grouping of the examples into meaningful *clusters*
  - so that we have a high
    - **intra-class similarity:** similarity between objects in same cluster
    - **inter-class dissimilarity:** dissimilarity between objects in different clusters

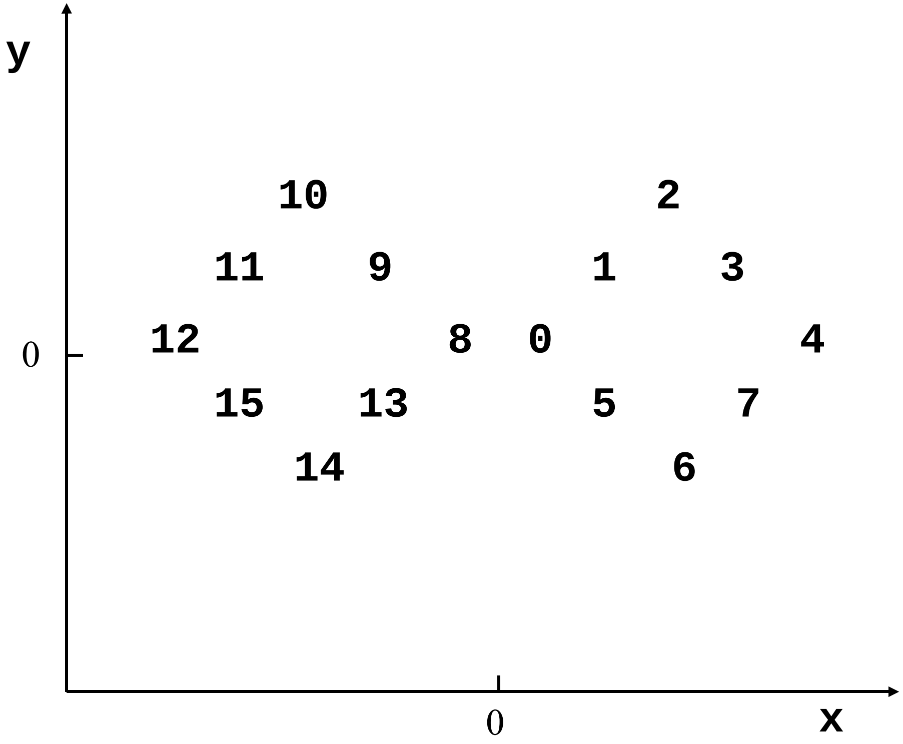
k-means Clustering

```
1. randomly select k cluster centers
2. assign each example to the nearest cluster center
3. compute a new cluster center
      ▪ mean of all examples assigned to that cluster
4. if there was some improvement
      ▪ goto 2.
```

- simple algorithm for finding a fixed number of clusters (k)
  - assumes a similarity function and a user-set value for k
  - optimizes intra-class similarity

k-means: Example

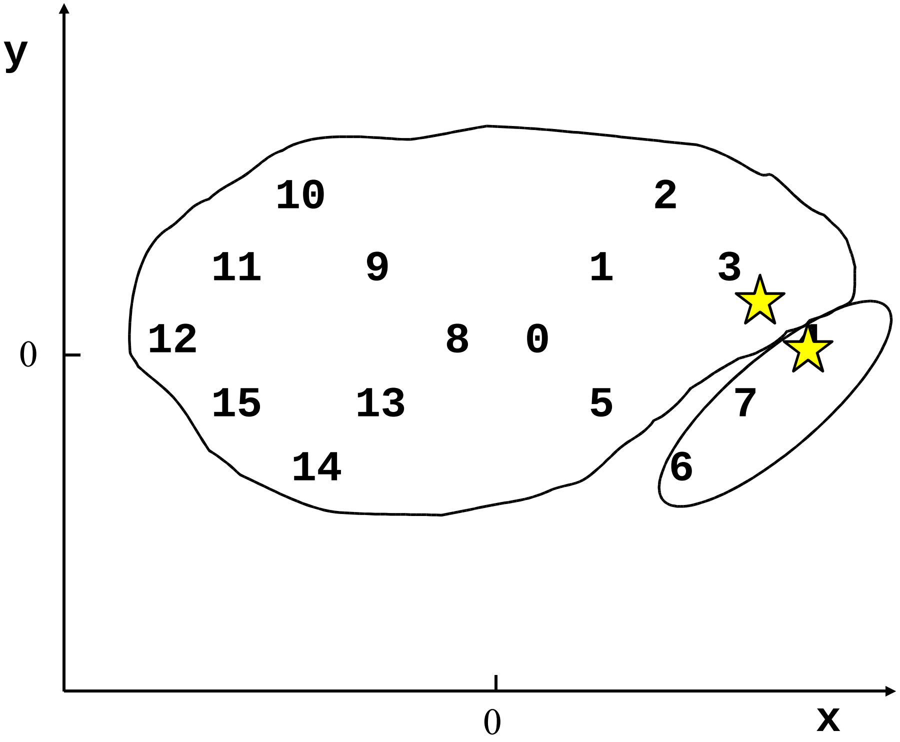
Id	x	y
0:	1.0	0.0
1:	3.0	2.0
2:	5.0	4.0
3:	7.0	2.0
4:	9.0	0.0
5:	3.0	-2.0
6:	5.0	-4.0
7:	7.0	-2.0
8:	-1.0	0.0
9:	-3.0	2.0
10:	-5.0	4.0
11:	-7.0	2.0
12:	-9.0	0.0
13:	-3.0	-2.0
14:	-5.0	-4.0
15:	-7.0	-2.0



- find the best 2 clusters

Seed: (9 0) (8 1)

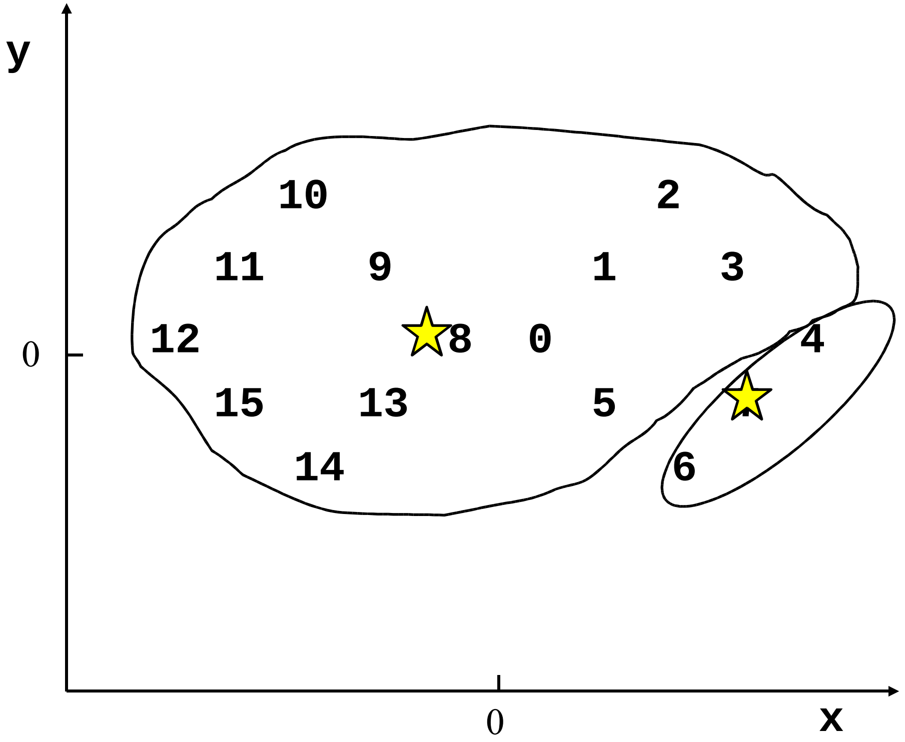
Clustering: (4 6 7) (0 1 2 3 5 8 9 10 11 12 13 14 15)
Cluster Centers: (7.0 -2.0) (-1.61538 0.46153)
Average Distance: 4.35887



Seed: (9 0) (8 1)

Clustering: (4 6 7) (0 1 2 3 5 8 9 10 11 12 13 14 15)
Cluster Centers: (7.0 -2.0) (-1.61538 0.46153)
Average Distance: 4.35887

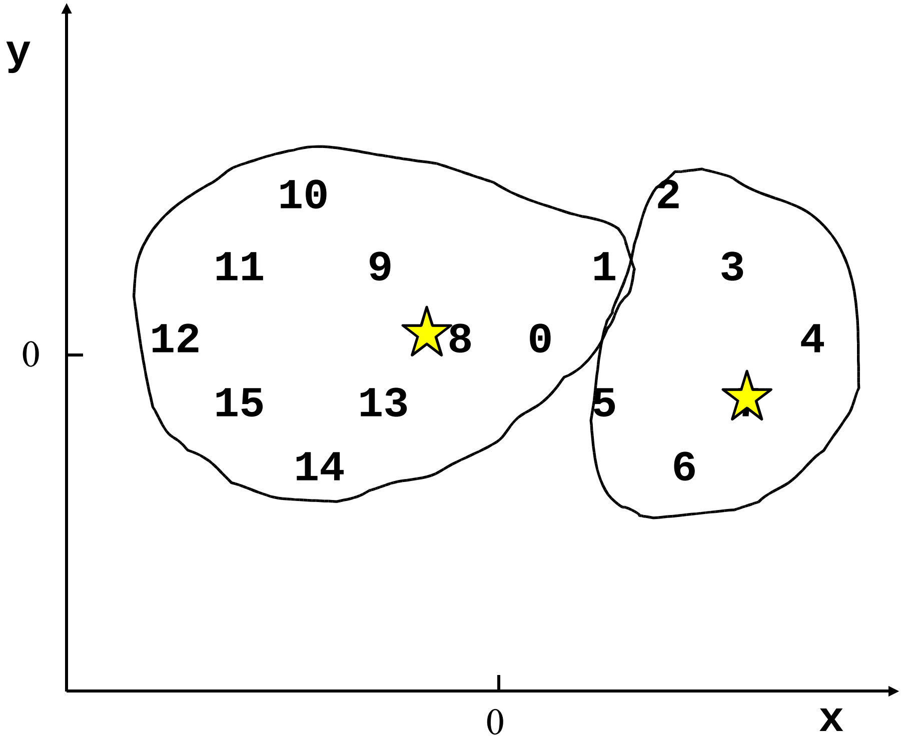
Clustering: (2 3 4 5 6 7) (0 1 8 9 10 11 12 13 14 15)



Seed: (9 0) (8 1)

Clustering: (4 6 7) (0 1 2 3 5 8 9 10 11 12 13 14 15)
Cluster Centers: (7.0 -2.0) (-1.61538 0.46153)
Average Distance: 4.35887

Clustering: (2 3 4 5 6 7) (0 1 8 9 10 11 12 13 14 15)
Cluster Centers: (6.0 -0.33334) (-3.6 0.2)
Average Distance: 3.6928

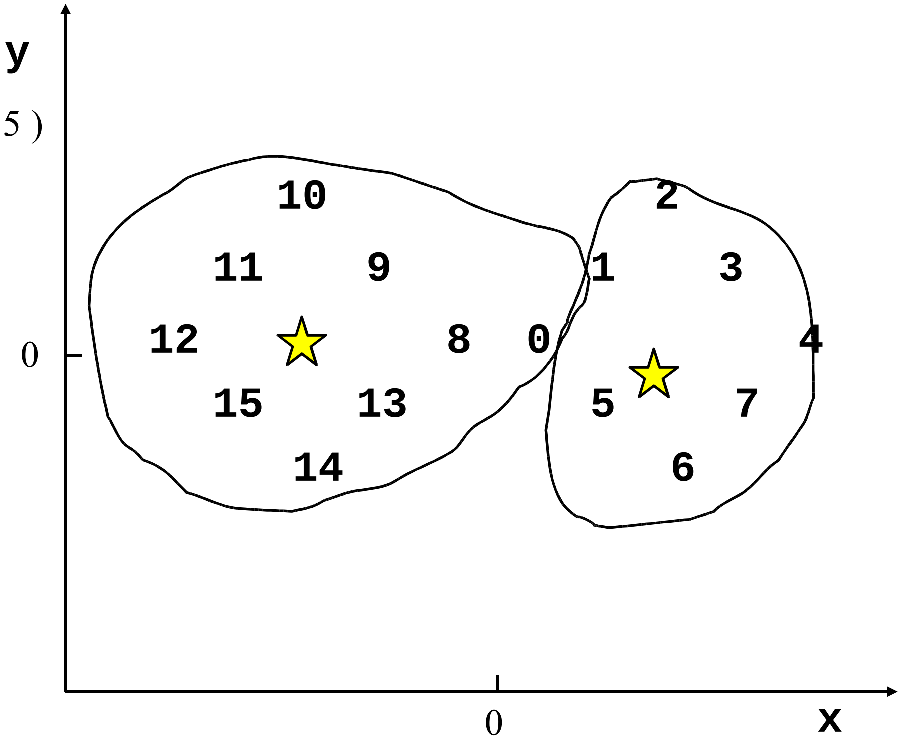


Seed: (9 0) (8 1)

Clustering: (4 6 7) (0 1 2 3 5 8 9 10 11 12 13 14 15)
Cluster Centers: (7.0 -2.0) (-1.61538 0.46153)
Average Distance: 4.35887

Clustering: (2 3 4 5 6 7) (0 1 8 9 10 11 12 13 14 15)
Cluster Centers: (6.0 -0.33334) (-3.6 0.2)
Average Distance: 3.6928

Clustering: (1 2 3 4 5 6 7) (0 8 9 10 11 12 13 14 15)

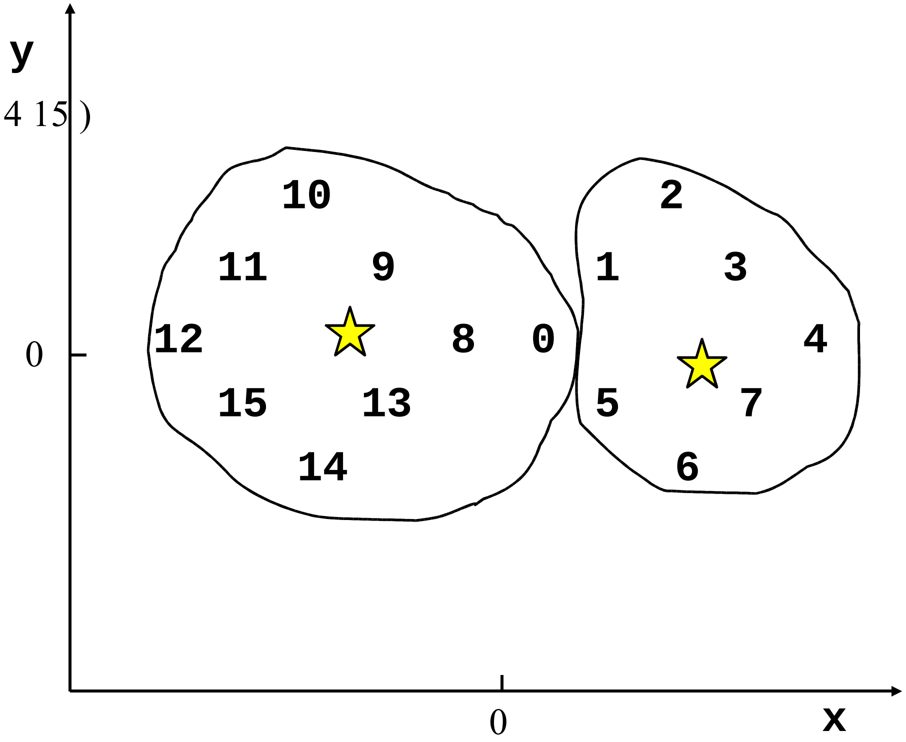


Seed: (9 0) (8 1)

Clustering: (4 6 7) (0 1 2 3 5 8 9 10 11 12 13 14 15)
Cluster Centers: (7.0 -2.0) (-1.61538 0.46153)
Average Distance: 4.35887

Clustering: (2 3 4 5 6 7) (0 1 8 9 10 11 12 13 14 15)
Cluster Centers: (6.0 -0.33334) (-3.6 0.2)
Average Distance: 3.6928

Clustering: (1 2 3 4 5 6 7) (0 8 9 10 11 12 13 14 15)
Cluster Centers: (5.57143 0.0) (-4.33334 0.0)
Average Distance: 3.49115



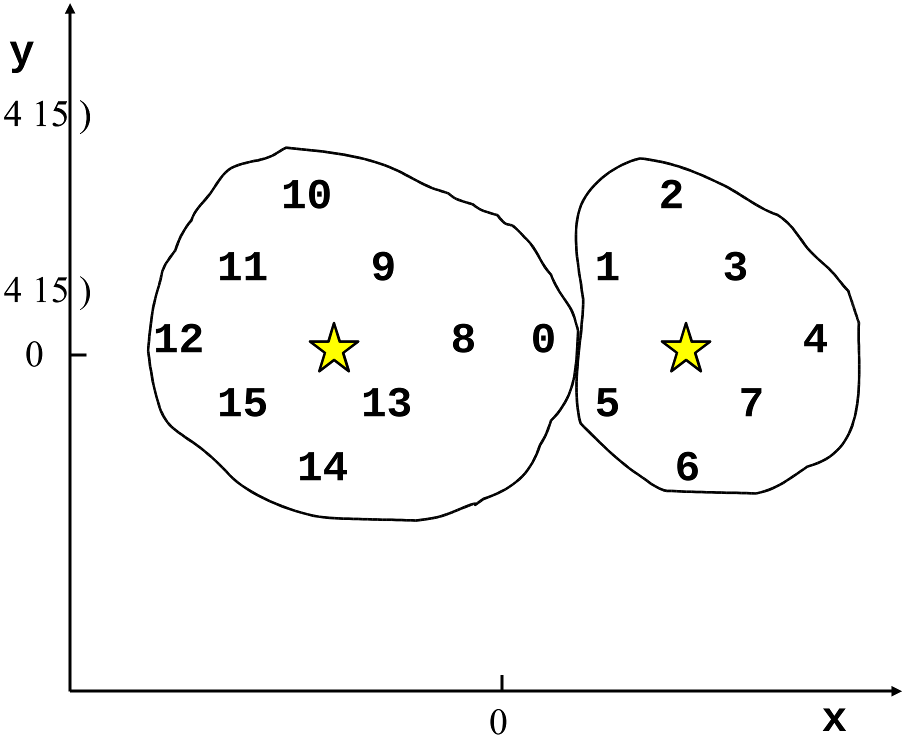
Seed: (9 0) (8 1)

Clustering: (4 6 7) (0 1 2 3 5 8 9 10 11 12 13 14 15)
Cluster Centers: (7.0 -2.0) (-1.61538 0.46153)
Average Distance: 4.35887

Clustering: (2 3 4 5 6 7) (0 1 8 9 10 11 12 13 14 15)
Cluster Centers: (6.0 -0.33334) (-3.6 0.2)
Average Distance: 3.6928

Clustering: (1 2 3 4 5 6 7) (0 8 9 10 11 12 13 14 15)
Cluster Centers: (5.57143 0.0) (-4.33334 0.0)
Average Distance: 3.49115

Clustering: (0 1 2 3 4 5 6 7) (8 9 10 11 12 13 14 15)



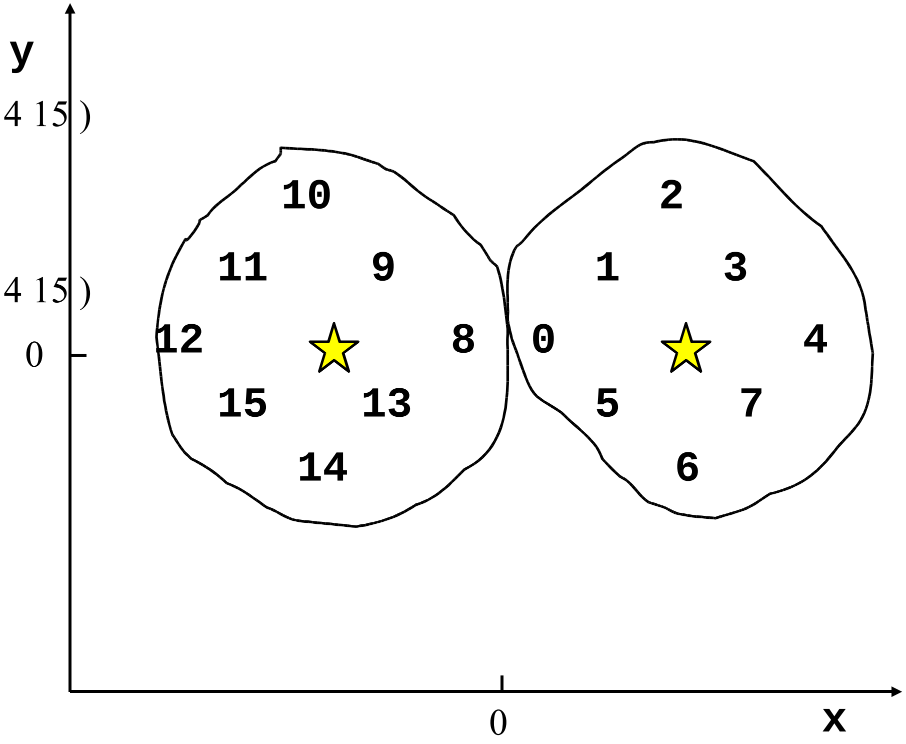
Seed: (9 0) (8 1)

Clustering: (4 6 7) (0 1 2 3 5 8 9 10 11 12 13 14 15)
Cluster Centers: (7.0 -2.0) (-1.61538 0.46153)
Average Distance: 4.35887

Clustering: (2 3 4 5 6 7) (0 1 8 9 10 11 12 13 14 15)
Cluster Centers: (6.0 -0.33334) (-3.6 0.2)
Average Distance: 3.6928

Clustering: (1 2 3 4 5 6 7) (0 8 9 10 11 12 13 14 15)
Cluster Centers: (5.57143 0.0) (-4.33334 0.0)
Average Distance: 3.49115

Clustering: (0 1 2 3 4 5 6 7) (8 9 10 11 12 13 14 15)
Cluster Centers: (5.0 0.0) (-5.0 0.0)
Average Distance: 3.41421



Seed: (9 0) (8 1)

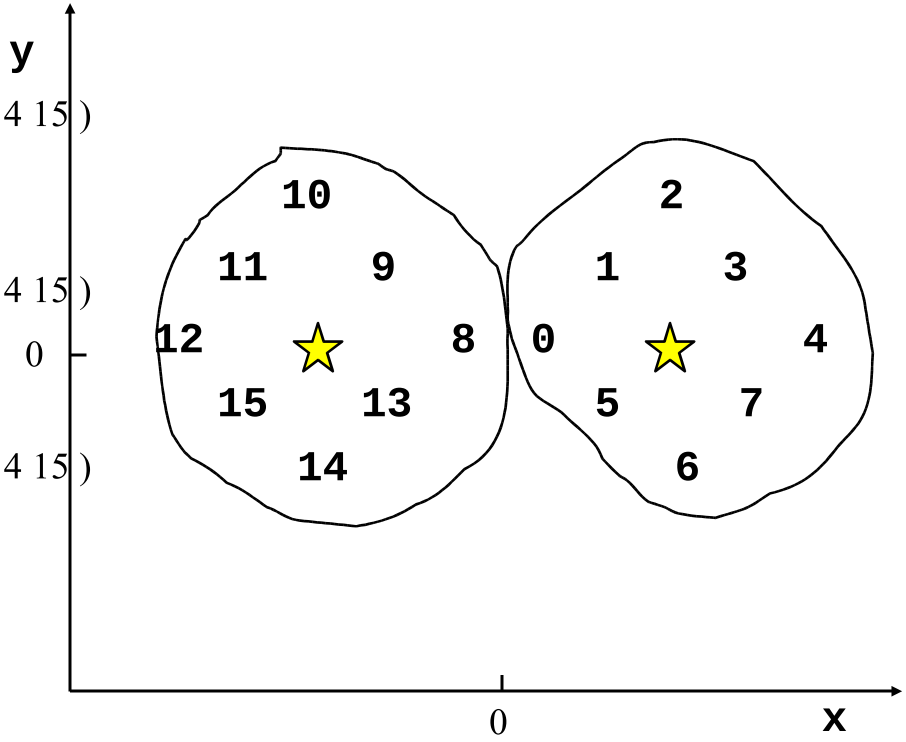
Clustering: (4 6 7) (0 1 2 3 5 8 9 10 11 12 13 14 15)
Cluster Centers: (7.0 -2.0) (-1.61538 0.46153)
Average Distance: 4.35887

Clustering: (2 3 4 5 6 7) (0 1 8 9 10 11 12 13 14 15)
Cluster Centers: (6.0 -0.33334) (-3.6 0.2)
Average Distance: 3.6928

Clustering: (1 2 3 4 5 6 7) (0 8 9 10 11 12 13 14 15)
Cluster Centers: (5.57143 0.0) (-4.33334 0.0)
Average Distance: 3.49115

Clustering: (0 1 2 3 4 5 6 7) (8 9 10 11 12 13 14 15)
Cluster Centers: (5.0 0.0) (-5.0 0.0)
Average Distance: 3.41421

Clustering: (0 1 2 3 4 5 6 7) (8 9 10 11 12 13 14 15)
No improvement.



Hierarchical Clustering

- Produces a tree hierarchy of clusters
  - *root:* all examples
  - *leaves:* single examples
  - *interior nodes:* subsets of examples
- Two approaches
  - **Top-down:**
    - start with maximal cluster (all examples)
    - successively split existing clusters
  - **Bottom-up:**
    - start with minimal clusters (single examples)
    - successively merge existing clusters

Bottom-Up Agglomerative Clustering

1. Start with one cluster for each example: $\boldsymbol{C} = \{\boldsymbol{C}_i\} = \{\{\boldsymbol{o}_i\} \mid \boldsymbol{o}_i \in \boldsymbol{O}\}$

2. compute distance $\boldsymbol{d}(\boldsymbol{C}_i, \boldsymbol{C}_j)$ between all pairs of Cluster $\boldsymbol{C}_i, \boldsymbol{C}_j$

3. Join clusters $\boldsymbol{C}_i$ und $\boldsymbol{C}_j$ with minimum distance into a new cluster $\boldsymbol{C}_p$; make $\boldsymbol{C}_p$ the parent node of $\boldsymbol{C}_i$ and $\boldsymbol{C}_j$:

$$\boldsymbol{C}_p = \{\boldsymbol{C}_i, \boldsymbol{C}_j\}$$
$$\boldsymbol{C} = (\boldsymbol{C} \setminus \{\boldsymbol{C}_i, \boldsymbol{C}_j\}) \cup \{\boldsymbol{C}_p\}$$

4. Compute distances between $\boldsymbol{C}_p$ and other clusteres in $\boldsymbol{C}$

5. If $|C| > 1$, goto 3.

Similarity between Clusters

ways of computing a similarity/distance between clusters C_1 and C_2

- Single-link:
  - minimum distance between two elements of C_1 and C_2

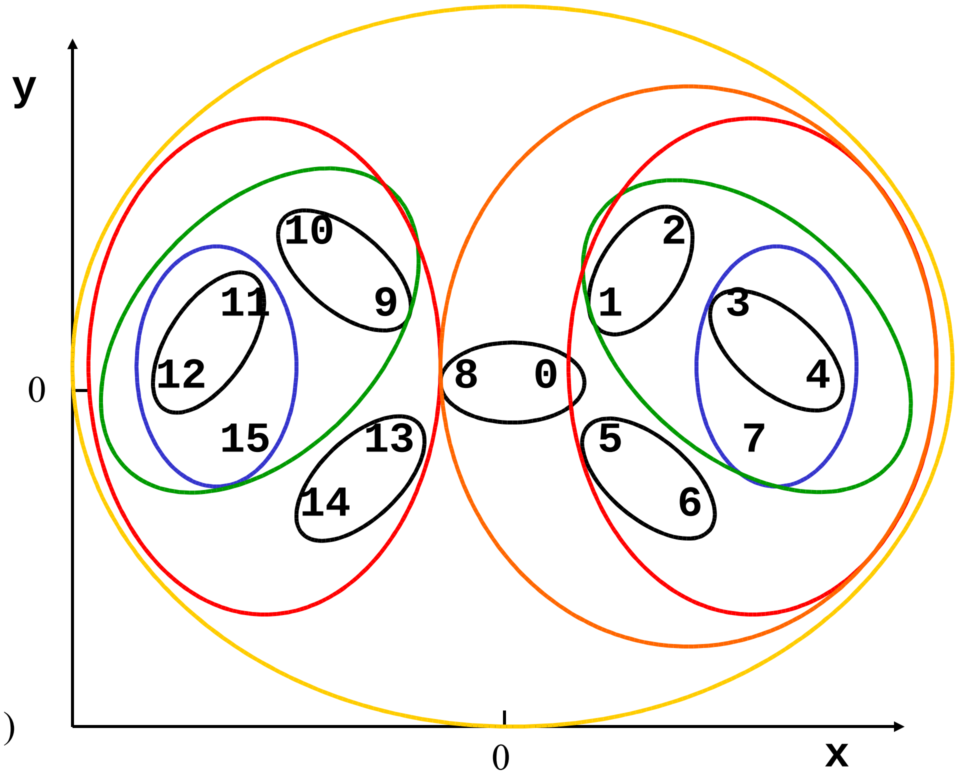
    $d(C_1, C_2) = min\{ d(x, y) \mid x \in C_1 , y \in C_2 \}$
- Complete-link:
  - maximum distance between two elements of C_1 and C_2

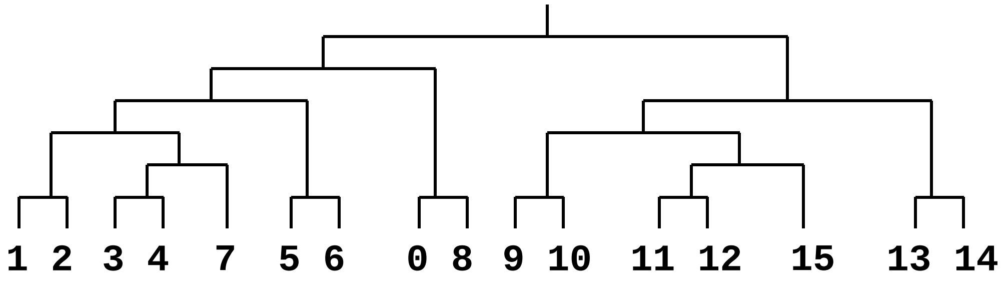
    $d(C_1, C_2) = max\{ d(x, y) \mid x \in C_1 , y \in C_2 \}$
- Average-link:
  - average distance between two elements of C_1 and C_2

    $d(C_1, C_2) = \sum\{ d(x, y) \mid x \in C_1 , y \in C_2 \} / |C_1| / |C_2|$

Bottom-up clustering (average-link):

```
min distance = 2.00000   ( 8 ) ( 0 )
min distance = 2.82843   ( 2 ) ( 1 )
min distance = 2.82843   ( 4 ) ( 3 )
min distance = 2.82843   ( 6 ) ( 5 )
min distance = 2.82843   ( 10 ) ( 9 )
min distance = 2.82843   ( 12 ) ( 11 )
min distance = 2.82843   ( 14 ) ( 13 )
min distance = 3.16228   ( 7 ) ( 3 4 )
min distance = 3.16228   ( 15 ) ( 11 12 )
min distance = 4.73756   ( 3 4 7 ) ( 1 2 )
min distance = 4.73756   ( 11 12 15 ) ( 9 10 )
min distance = 4.74131   ( 1 2 3 4 7 ) ( 5 6 )
min distance = 4.74131   ( 9 10 11 12 15 ) ( 13 14 )
min distance = 5.57143   ( 0 8 ) ( 5 6 1 2 3 4 7 )
min distance = 9.90476   ( 13 14 9 10 11 12 15 ) ( 5 6 1 2 3 4 7 0 8 )
```


y
0
x
0
10
9
11
12
15
13
14
8
0
2
1
3
4
5
7
6


1 2 3 4 7 5 6 0 8 9 10 11 12 15 13 14