01.112/50.007 Machine Learning

Lecture 6

K-Medoids Clustering

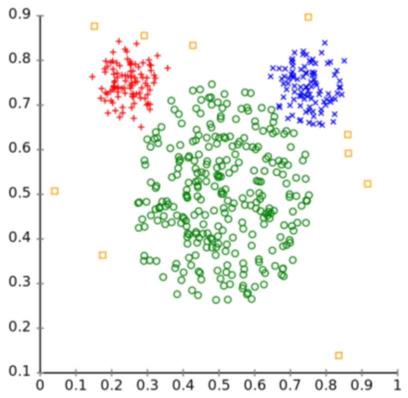
What is clustering

- Form of *unsupervised* learning - no information from teacher
- The process of partitioning a set of data into a set of meaningful (hopefully) sub-classes, called *clusters*

Cluster:

- collection of data points that are "similar" to one another and collectively should be treated as group
- as a collection, are sufficiently different from other groups

What is clustering



Clustering Problem.

Input.

Training data $\mathcal{S}_n = \left\{x^{(i)}; i = 1, 2, \ldots, n\right\}$, each $x^{(i)} \in \mathbb{R}^d$. Integer k.

Output.

Clusters $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_k \subset \{1, 2, \ldots, n\}$ such that every data point is in one and only one cluster.

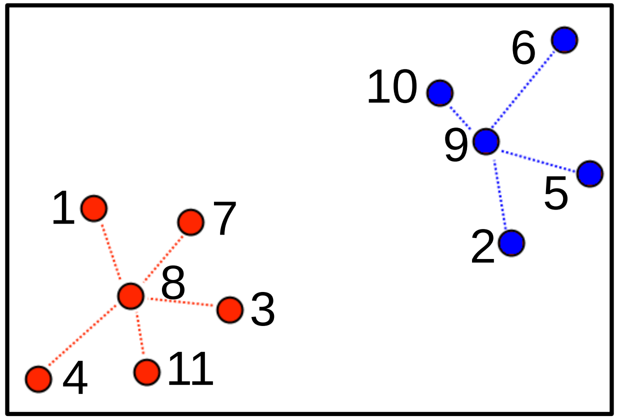
Some clusters could be empty!

How to Specify a Cluster

- By listing all its **elements**

$$\mathcal{C}_1 = \{1,3,4,7,8,11\}$$

$$\mathcal{C}_2 = \{2,5,6,9,10\}$$



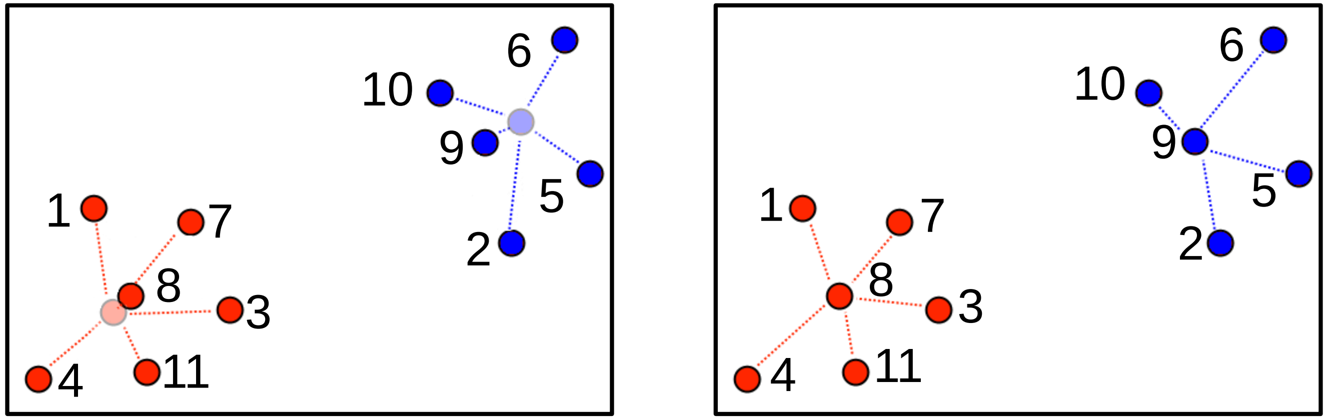
How to Specify a Cluster

- Using a **representative**
  a. A point in center of cluster (centroid)
  b. A point in the training data (exemplar)

Each point $x^{(i)}$ will be assigned the closest representative.

$z^{(1)} = \binom{1}{1}, z^{(2)} = \binom{5}{4}$

$z^{(1)} = 8, z^{(2)} = 9$



centroid

exemplar

K-Means Algorithm

1. Initialize centroids $z^{(1)}, \ldots, z^{(k)}$ from the data.
2. Repeat until no further change in training loss:

   a. For each $j \in \{1, \ldots, k\}$,

   $$\mathcal{C}_j = \{\, i \text{ such that } x^{(i)} \text{ is closest to } z^{(j)} \,\}.$$

   b. For each $j \in \{1, \ldots, k\}$,

   $$z^{(j)} = \frac{1}{|\mathcal{C}_j|} \sum_{i \in \mathcal{C}_j} x^{(i)} \quad \text{(cluster mean)}$$

Animation: https://towardsdatascience.com/k-means-clustering-introduction-to-machine-learning-algorithms-c96bf0d5d57a

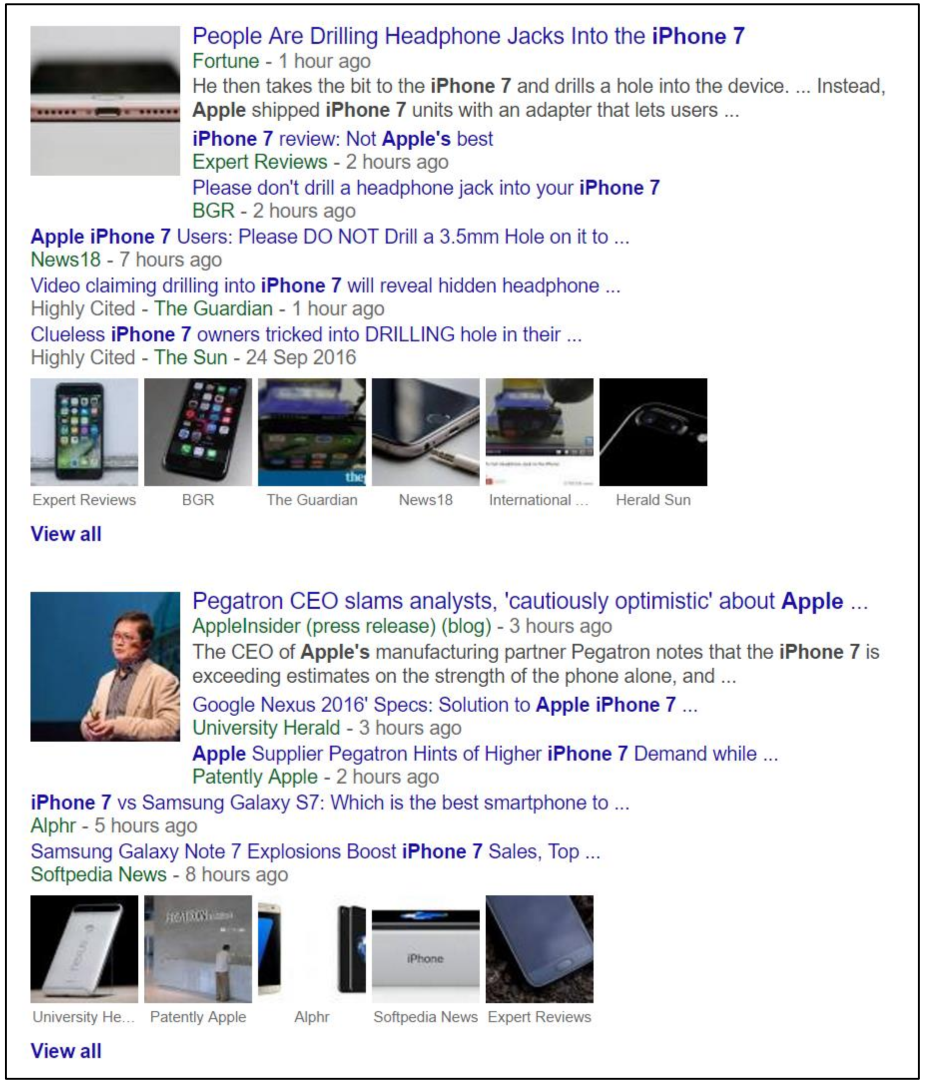
K-Medoids

Use exemplars
instead of centroids.

e.g. Google News.

Repeat until convergence:

- Find best clusters given exemplars
- Find best exemplars given clusters



K-Medoids Algorithm

1. Initialize exemplars $z^{(1)}, \ldots, z^{(k)} \subseteq \{x^{(1)}, \ldots x^{(n)}\}$
2. Repeat until no further change in training loss:

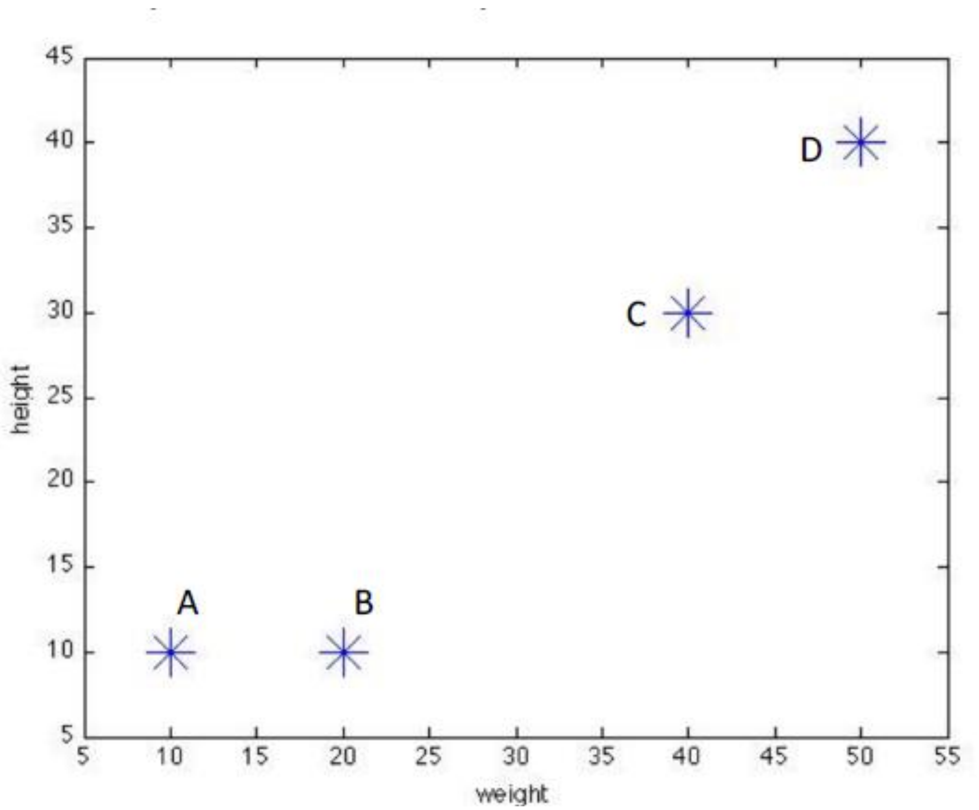
   a. For each $j \in \{1, \ldots, k\}$,
      $\mathcal{C}^{\mathrm{j}} = \{\, i$ such that $x^{(i)}$ is closest to $z^{(j)}\,\}$.

   b. For each $j \in \{1, \ldots, k\}$, set $z^{(j)}$ to be the point in $C^{(j)}$ that minimizes $\sum_{i \in \mathcal{C}^{\mathrm{j}}} d(x^{(i)}, z^{(j)})$

Example: K-Means

Example: K-Means

- Suppose we have 4 boxes of different sizes and we want to divide them into 2 clusters. Each box represents one point with two attributes (X,Y):



$$A = (10,10),\\ B = (20,10),\\ C = (40,30),\\ D = (50,40)$$

Example: K-Means

- Initial centers: suppose we choose points A and B as the initial centers, so c1 = (10, 10) and c2 = (20, 10)
- Object - centre distance: calculate the Euclidean distance between cluster centers and the objects.

- We obtain the following distance matrix:

	A	B	C	D
Centre 1				
Centre 2				

Example: K-Means

- Initial centers: suppose we choose points A and B as the initial centers, so c1 = (10, 10) and c2 = (20, 10)
- Object - centre distance: calculate the Euclidean distance between cluster centers and the objects. For example, the distance of object C from the first center is:

$$\sqrt{(40-10)^2+(30-10)^2} = 36.06$$

- We obtain the following distance matrix:

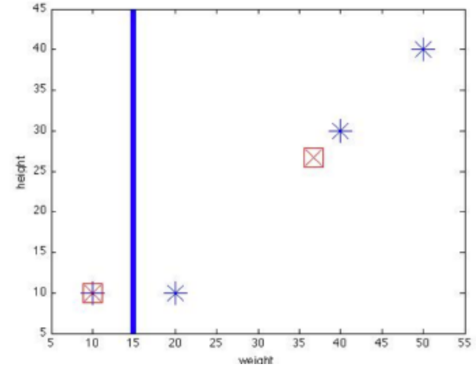
	A	B	C	D
Centre 1	0	10	36.06	50
Centre 2	10	0	28.28	43.43

Example: K-Means

- Object clustering: We assign each object to one of the clusters based on the minimum distance from the centre:

	A	B	C	D
Centre 1	1	0	0	0
Centre 2	0	1	1	1

- Determine centers: Based on the group membership, we compute the new centers



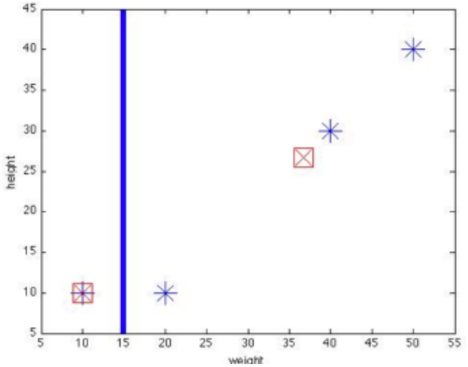
Example: K-Means

- Object clustering: We assign each object to one of the clusters based on the minimum distance from the centre:

	A	B	C	D
Centre 1	1	0	0	0
Centre 2	0	1	1	1

- Determine centers: Based on the group membership, we compute the new centers

$$c_1 = (10, 10), c_2 = \left(\frac{20+40+50}{3}, \frac{10+30+40}{3}\right) = (36.7, 26.7)$$



Example: K-Means

- Recompute the object-center distances: We compute the distances of each data point from the new centers:

	A	B	C	D
Centre 1				
Centre 2				

- Object clustering: We reassign the objects to the clusters based on the minimum distance from the center:

	A	B	C	D
Centre 1				
Centre 2				

Example: K-Means

- Recompute the object-center distances: We compute the distances of each data point from the new centers:

	A	B	C	D
Centre 1	0	10	36.06	50
Centre 2	31.4	23.6	4.7	18.9

- Object clustering: We reassign the objects to the clusters based on the minimum distance from the center:

	A	B	C	D
Centre 1	1	1	0	0
Centre 2	0	0	1	1

Example: K-Means

- Determine the new centers:

- Recompute the object-centers distances

	A	B	C	D
Centre 1				
Centre 2				

- Object clustering

	A	B	C	D
Centre 1				
Centre 2				

Example: K-Means

- Determine the new centers:

$$c_1 = \left(\frac{10+20}{2}, \frac{10+10}{2}\right) = (15, 10)$$
$$c_2 = \left(\frac{40+50}{2}, \frac{30+40}{2}\right) = (45, 35)$$

- Recompute the object-centers distances

	A	B	C	D
Centre 1	5	5	32	46.1
Centre 2	43	35.4	7.1	7.1

- Object clustering

	A	B	C	D
Centre 1	1	1	0	0
Centre 2	0	0	1	1

- The cluster membership did not change from one iteration to another. So the k-means computation terminates.

Example: K-Medoids

K-Medoids Algorithm

1. Initialize exemplars $z^{(1)}, \ldots, z^{(k)} \subseteq \{x^{(1)}, \ldots x^{(n)}\}$
2. Repeat until no further change in training loss:

   a. For each $j \in \{1, \ldots, k\}$,
   $\mathcal{C}^{\mathrm{j}} = \{\, i$ such that $x^{(i)}$ is closest to $z^{(j)}\,\}$.

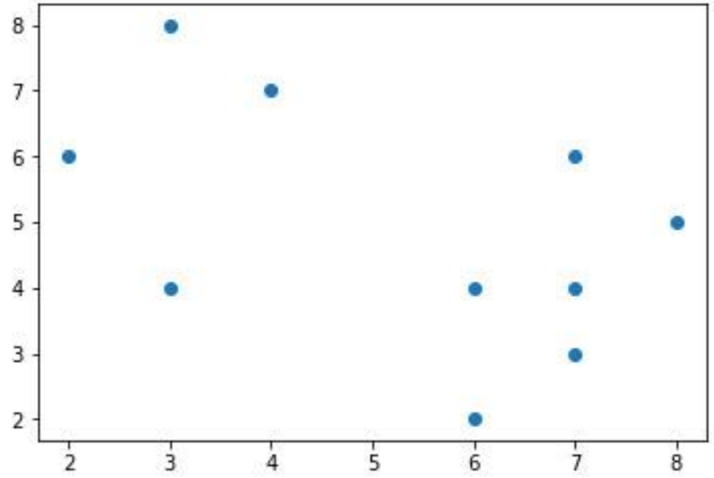
   b. For each $j \in \{1, \ldots, k\}$,
   set $z^{(j)}$ to be the point in $C^{(j)}$ that minimizes $\sum_{i \in \mathcal{C}^{\mathrm{j}}} d(x^{(i)}, z^{(j)})$

For each data point, $x^{(i)}$ which is not a medoid:

1. Swap $z^{(j)}$ and $x^{(i)}$, associate each data point to the swapped medoid, recompute the cost.
2. If the total cost is more than that in the previous step, undo the swap.

Example: K-Medoids

- Consider the following set of points



$x^{(1)}$	2	6
$x^{(2)}$	3	4
$x^{(3)}$	3	8
$x^{(4)}$	4	7
$x^{(5)}$	6	2
$x^{(6)}$	6	4
$x^{(7)}$	7	3
$x^{(8)}$	7	4
$x^{(9)}$	8	5
$x^{(10)}$	7	6

- We will consider L1 distance $d\left(x^{(i)}, z^{(j)}\right) = |x^{(i)} - z^{(j)}|$

Source: https://en.wikipedia.org/wiki/K-medoids

Example: K-Medoids

- Let the randomly selected 2 medoids be

$$z^{(1)} = (3,4)$$
$$z^{(2)} = (7,4)$$

- The cost of each non-medoid point with the medoids is calculated and tabulated:

Data object		Distance to	
i	$x^{(i)}$	$z^{(1)} = (3,4)$	$z^{(2)} = (7,4)$
1	(2, 6)	3	7
2	(3, 4)	0	4
3	(3, 8)	4	8
4	(4, 7)	4	6
5	(6, 2)	5	3
6	(6, 4)	3	1
7	(7, 3)	5	1
8	(7, 4)	4	0
9	(8, 5)	6	2
10	(7, 6)	6	2
Cost			

Example: K-Medoids

- Let the randomly selected 2 medoids be

$$z^{(1)} = (3,4)$$
$$z^{(2)} = (7,4)$$

- The total cost of this clustering is:

Cluster 1: (3+0+4+4) = 11

Cluster 2: (3+1+1+0+2+2) = 9

Total: 20

Data object		Distance to	
i	$x^{(i)}$	$z^{(1)} = (3,4)$	$z^{(2)} = (7,4)$
1	(2, 6)	**3**	7
2	(3, 4)	**0**	4
3	(3, 8)	**4**	8
4	(4, 7)	**4**	6
5	(6, 2)	5	**3**
6	(6, 4)	3	**1**
7	(7, 3)	5	**1**
8	(7, 4)	4	**0**
9	(8, 5)	6	**2**
10	(7, 6)	6	**2**
Cost		**11**	**9**

Example: K-Medoids

- Updating Z_2 with a non-medoid point, O'

$$z^{(1)} = (3,4)$$
$$O' = (7,3)$$

- The total cost of this clustering is:

Cluster 1: (3+0+4+4) = 11

Cluster 2: (2+2+0+1+3+3) = 11

Total: 22

i	$z^{(1)}$		$x^{(i)}$		dist
1	3	4	2	6	**3**
3	3	4	3	8	**4**
4	3	4	4	7	**4**
5	3	4	6	2	5
6	3	4	6	4	3
8	3	4	7	4	4
9	3	4	8	5	6
10	3	4	7	6	6

i	O'		$x^{(i)}$		dist
1	7	3	2	6	8
3	7	3	3	8	9
4	7	3	4	7	7
5	7	3	6	2	**2**
6	7	3	6	4	**2**
8	7	3	7	4	**1**
9	7	3	8	5	**3**
10	7	3	7	6	**3**

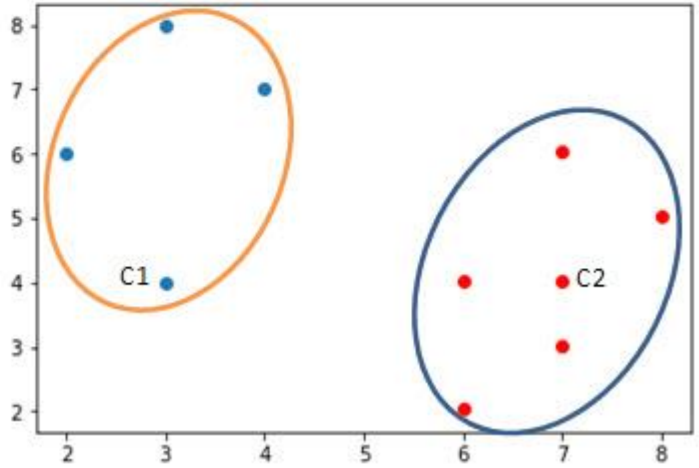
Example: K-Medoids

- The total cost of this clustering is:

Cluster 1: (3+0+4+4) = 11

Cluster 2: (2+2+0+1+3+3) = 11

Total: 22 > 20 → No swapping



Discussion

Number of Clusters

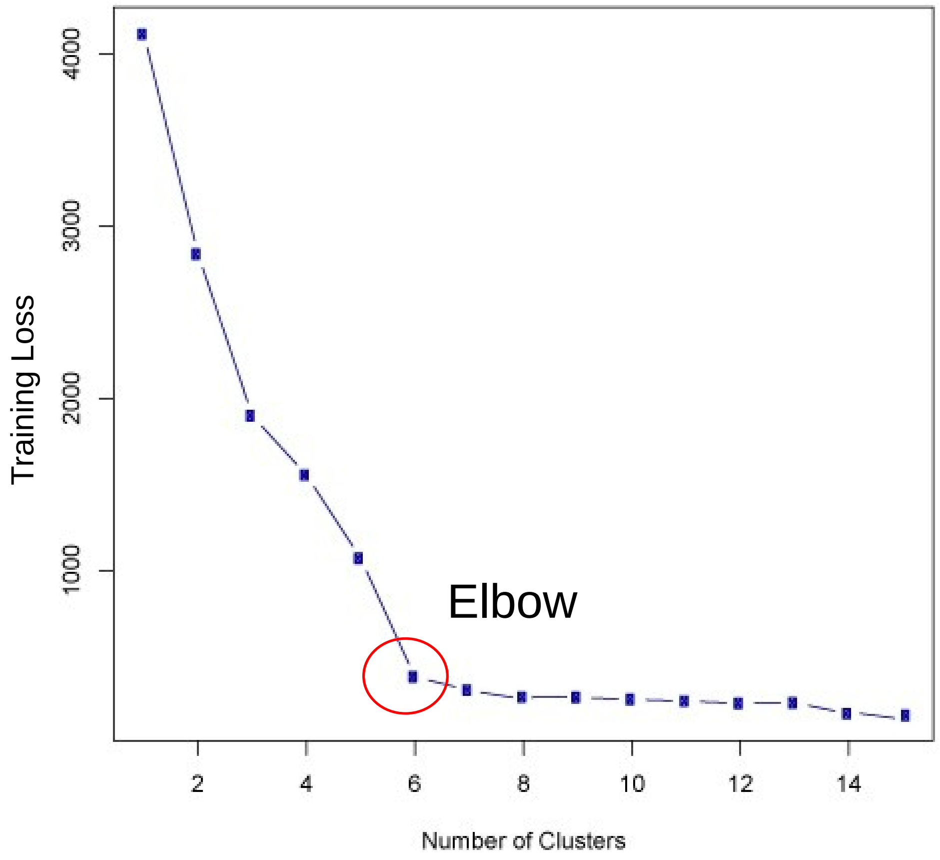
Generalization

How do we choose k, the optimal number of clusters?

- Elbow method
  - Training Loss
  - Validation Loss
- Semi-supervised learning
  - Accuracy in supervised task

Elbow Method

Generalization

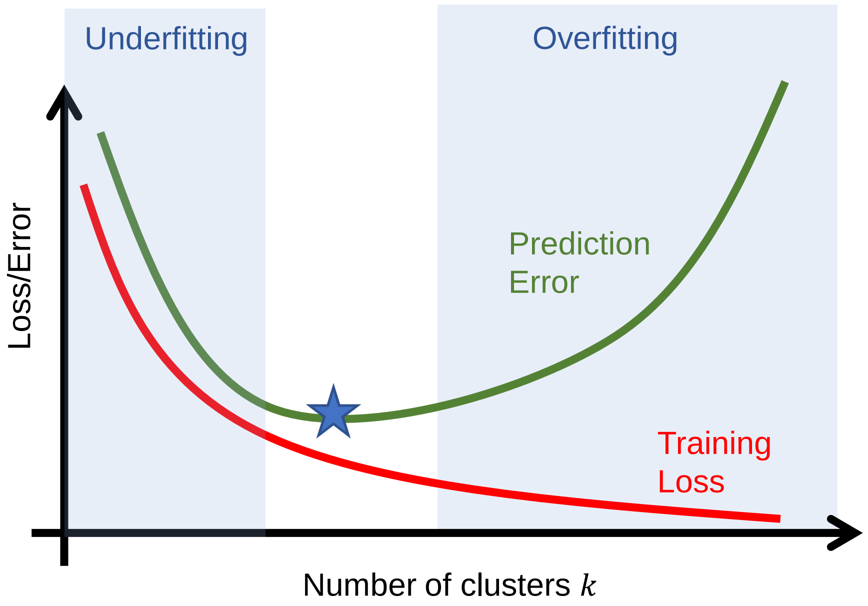


Semi-Supervised Learning

Supervised task with small *labeled* data $\mathcal{S}'$

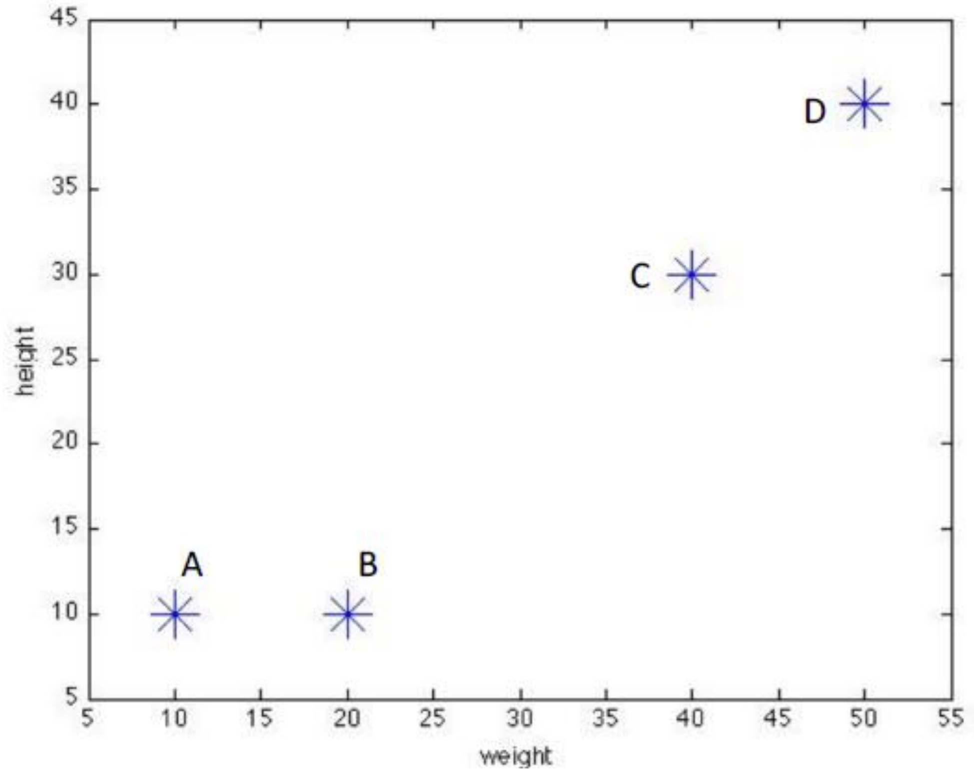
- For each number of clusters k,
  1. Perform k–means on *unlabeled* data.
  2. Transform $\mathcal{S}'$ using learned clusters e.g. compute distance to each centroid.
  3. Use new features for supervised task, and compute prediction error.
- Pick k with smallest prediction error.

Generalization



Example

1. Perform k–means on *unlabeled* data.



Unlabelled data:

$$A = (10,10), B = (20,10),$$
$$C = (40,30), D = (50,40)$$

Centroids:

c1 = (15,10), c2 = (45,35)

Distances:

	A	B	C	D
Centre 1	5	5	32	46.1
Centre 2	43	35.4	7.1	7.1

Example

2. Transform $\mathcal{S}'$ using learned clusters
e.g. compute distance to each centroid.

Labelled original data:

$$\mathcal{S}' = \{((50,30),+1),((15,20),-1)\}$$

Unlabelled data:

$$A = (10,10), B = (20,10),$$
$$C = (40,30), D = (50,40)$$

Centroids:

c1 = (15,10), c2 = (45,35)

	A	B	C	D	**E**	**F**
Centre 1	5	5	32	46.1		
Centre 2	43	35.4	7.1	7.1		

Example

3. Use new features for supervised task, and compute prediction error.

Labelled original data:

$$\mathcal{S}' = \{((50,30), +1), ((15,20), -1)\}$$

Transformed labelled data:

$$\mathcal{S}' = \{((40.3, 7.07), +1), ((10, 33.54), -1)\}$$

Unlabelled data:

$$A = (10,10), B = (20,10),$$
$$C = (40,30), D = (50,40)$$

Centroids:

c1 = (15,10), c2 = (45,35)

	A	B	C	D	**E**	**F**
Centre 1	5	5	32	46.1	**40.3**	**10**
Centre 2	43	35.4	7.1	7.1	**7.07**	**33.54**

Points belonging to same cluster have similar features

Summary

- The **K-medoids algorithm** shares the properties of K-means that we discussed (each iteration **decreases the cost**; the algorithm always **converges**; different starts gives different final answers; it **does not achieve the global minimum**)

- **K-medoids is computationally harder** than K-means (because of step 2: computing the medoid is harder than computing the average)

- Remember, **K-medoids** has the (potentially important) property that the **centers are located among the data points themselves**