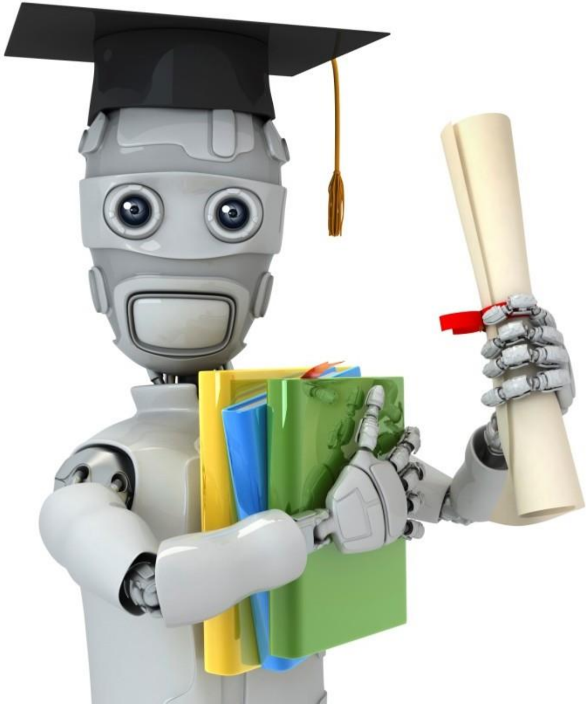
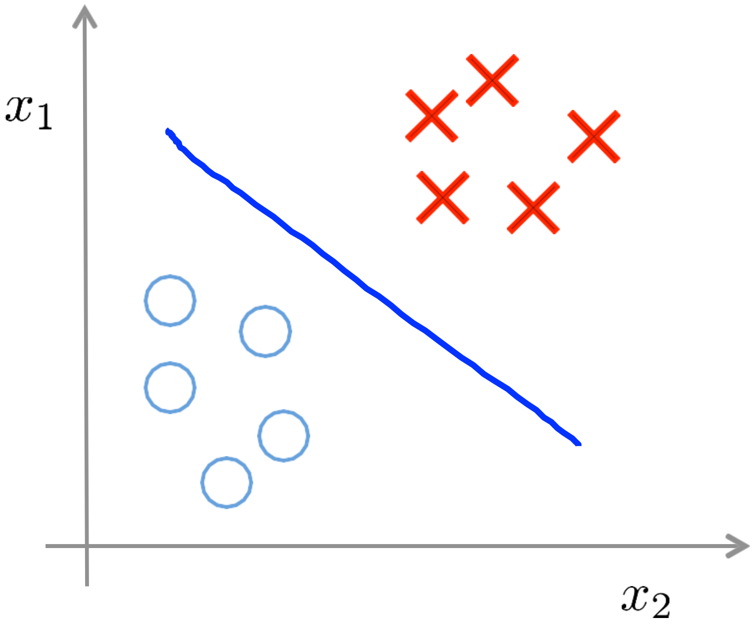


Machine Learning

Clustering

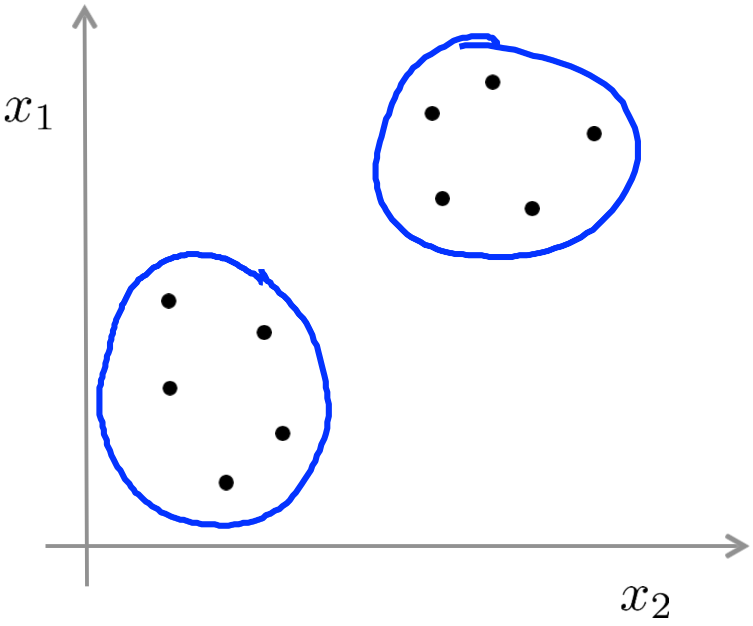
Unsupervised learning introduction

Supervised learning



Training set: $\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), (x^{(3)}, y^{(3)}), \ldots, (x^{(m)}, y^{(m)})\}$

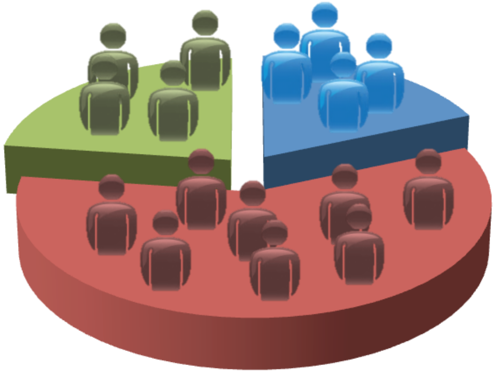
Unsupervised learning



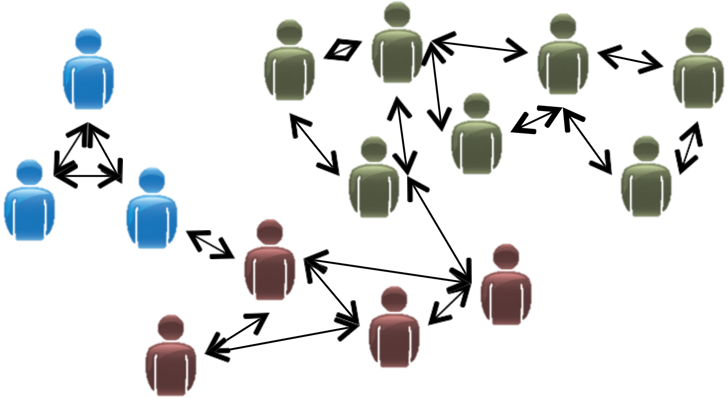
Clustering algorithm

Training set: $\{x^{(1)}, x^{(2)}, x^{(3)}, \ldots, x^{(m)}\}$

Applications of clustering



Market segmentation



Social network analysis

Organize computing clusters

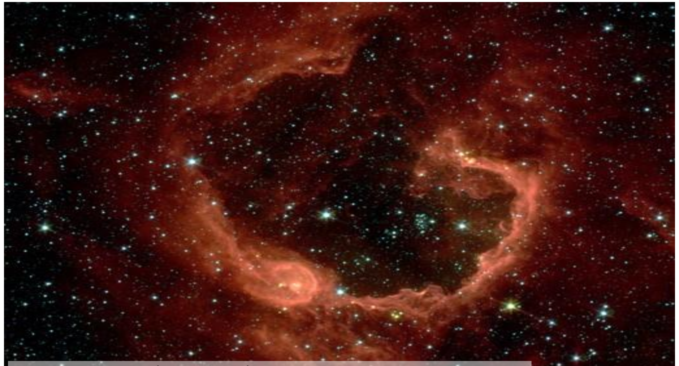
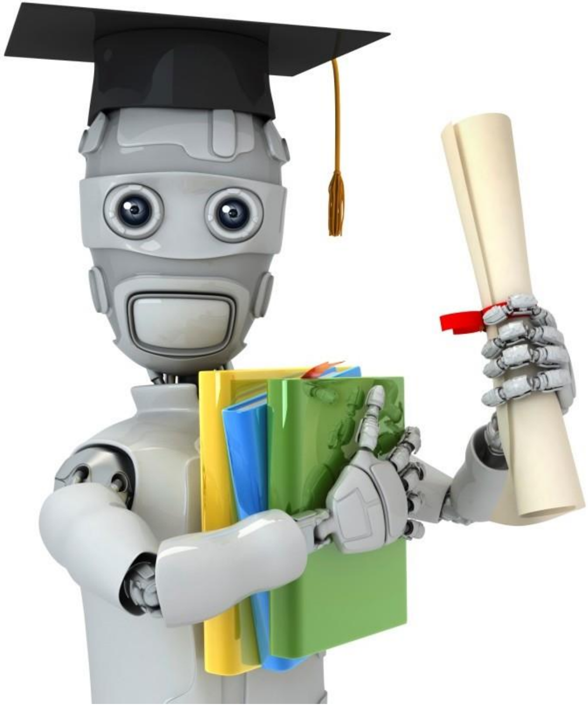


Image credit: NASA/JPL--Caltech/E. Churchwell (Univ. of Wisconsin, Madison)

Astronomical data analysis

Which of the following statements are true? Check all that apply.

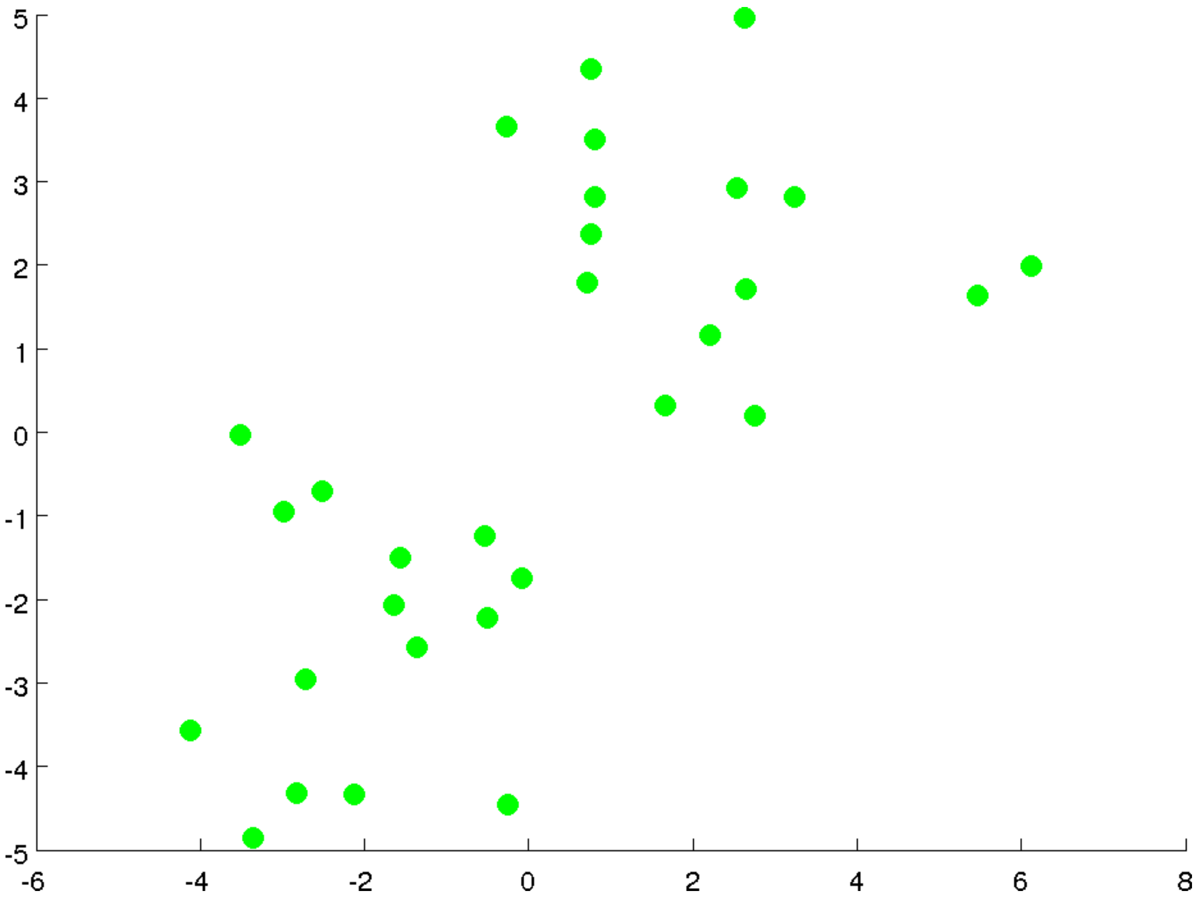
- [ ] In unsupervised learning, the training set is of the form $\{x^{(1)}, x^{(2)}, \ldots, x^{(m)}\}$ without labels $y^{(i)}$.
- [ ] Clustering is an example of unsupervised learning.
- [ ] In unsupervised learning, you are given an unlabeled dataset and are asked to find "structure" in the data.
- [ ] Clustering is the only unsupervised learning algorithm.

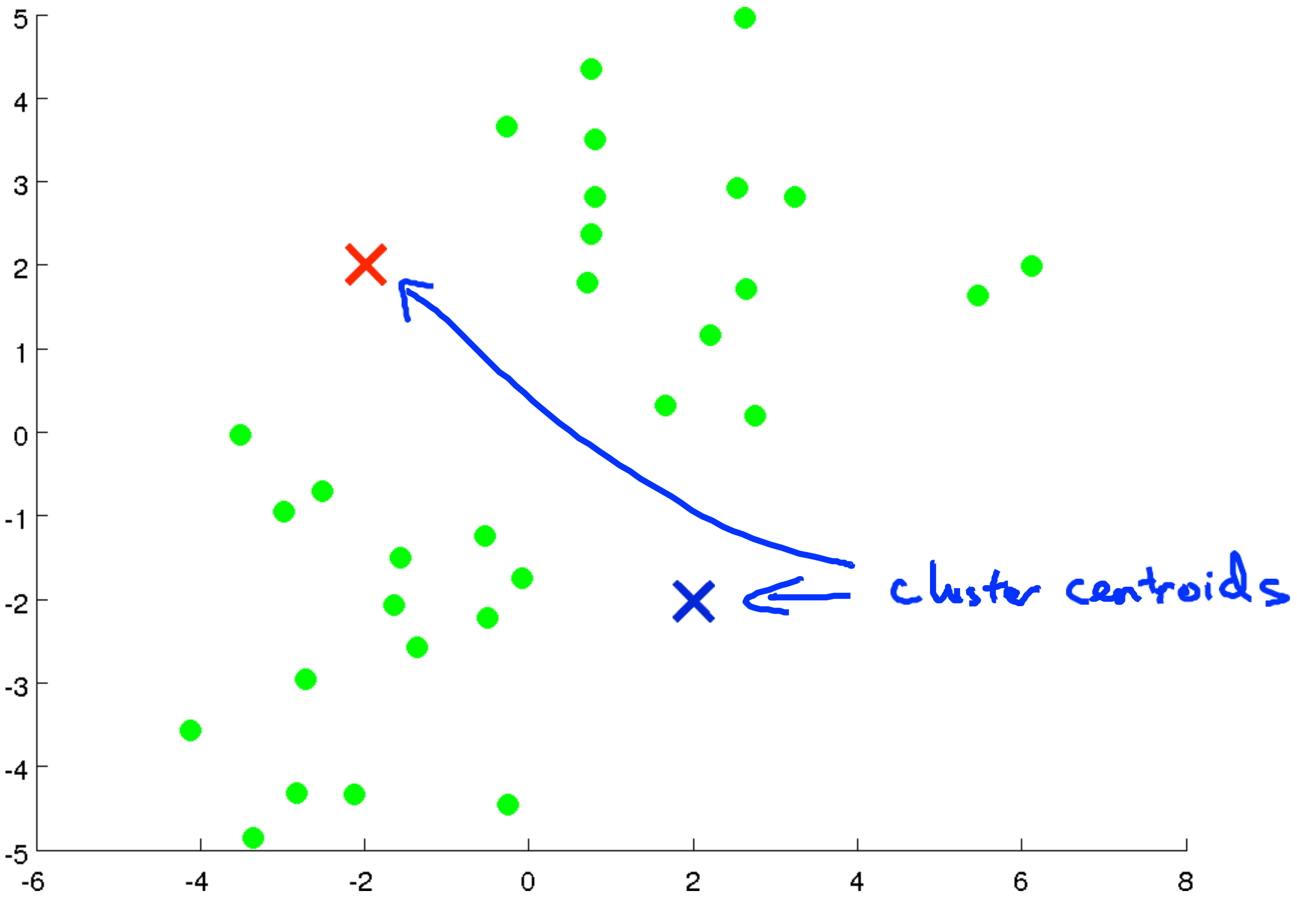


Machine Learning

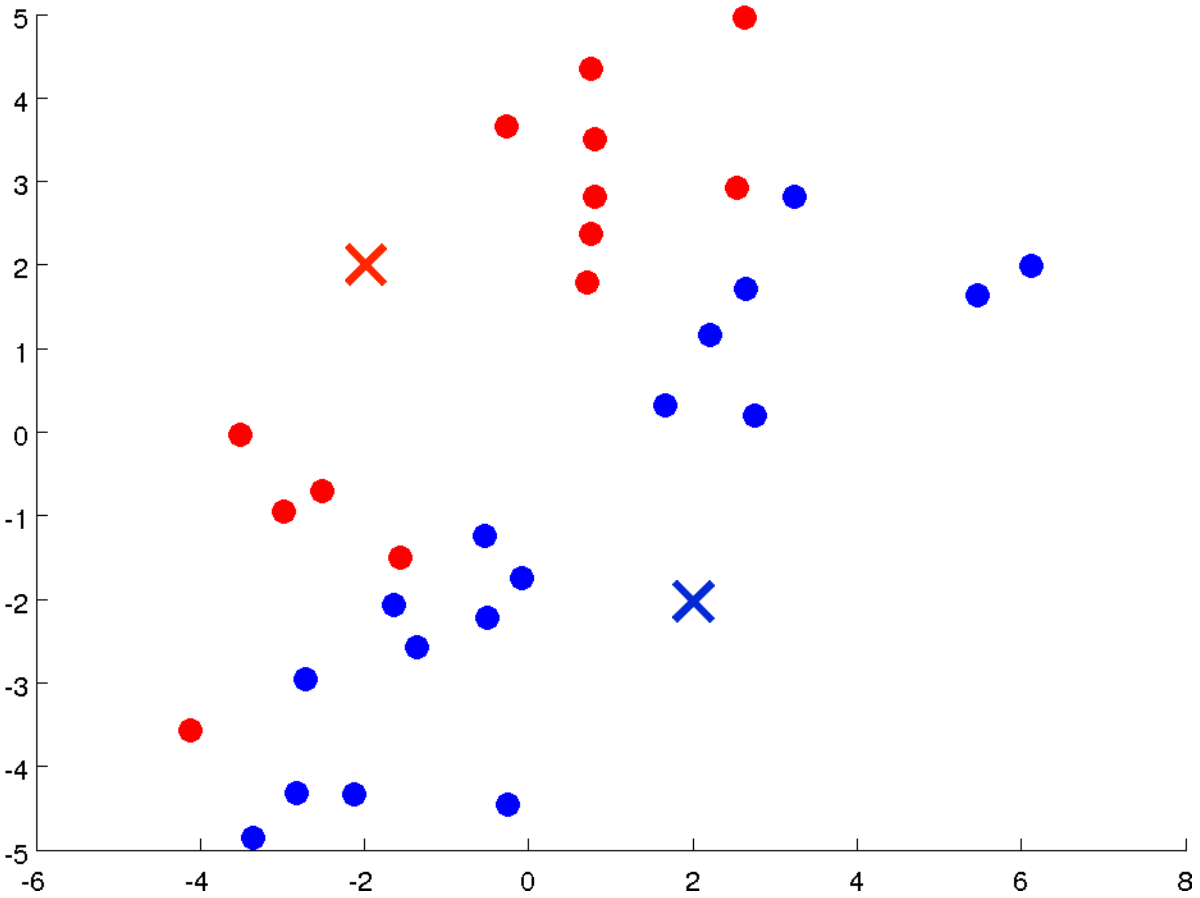
Clustering

K-means algorithm

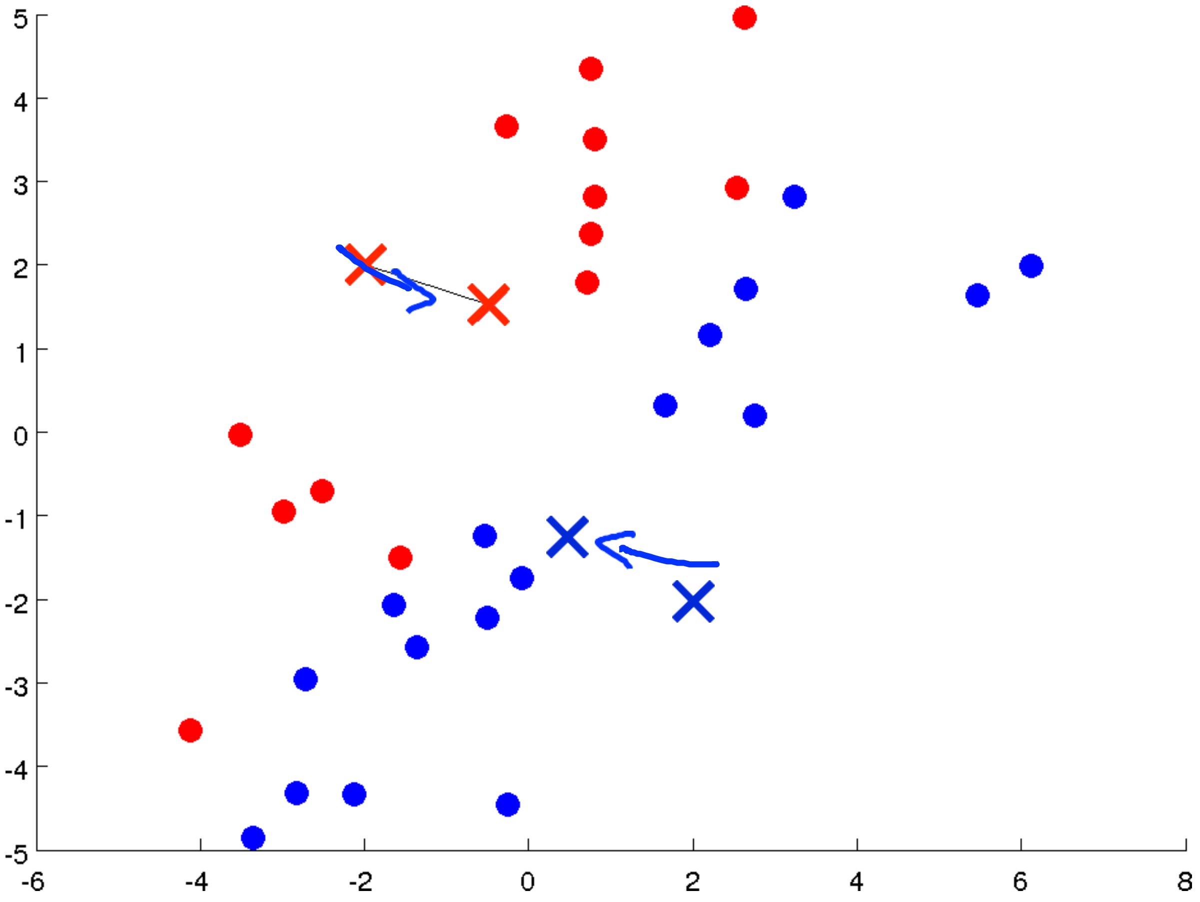

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8

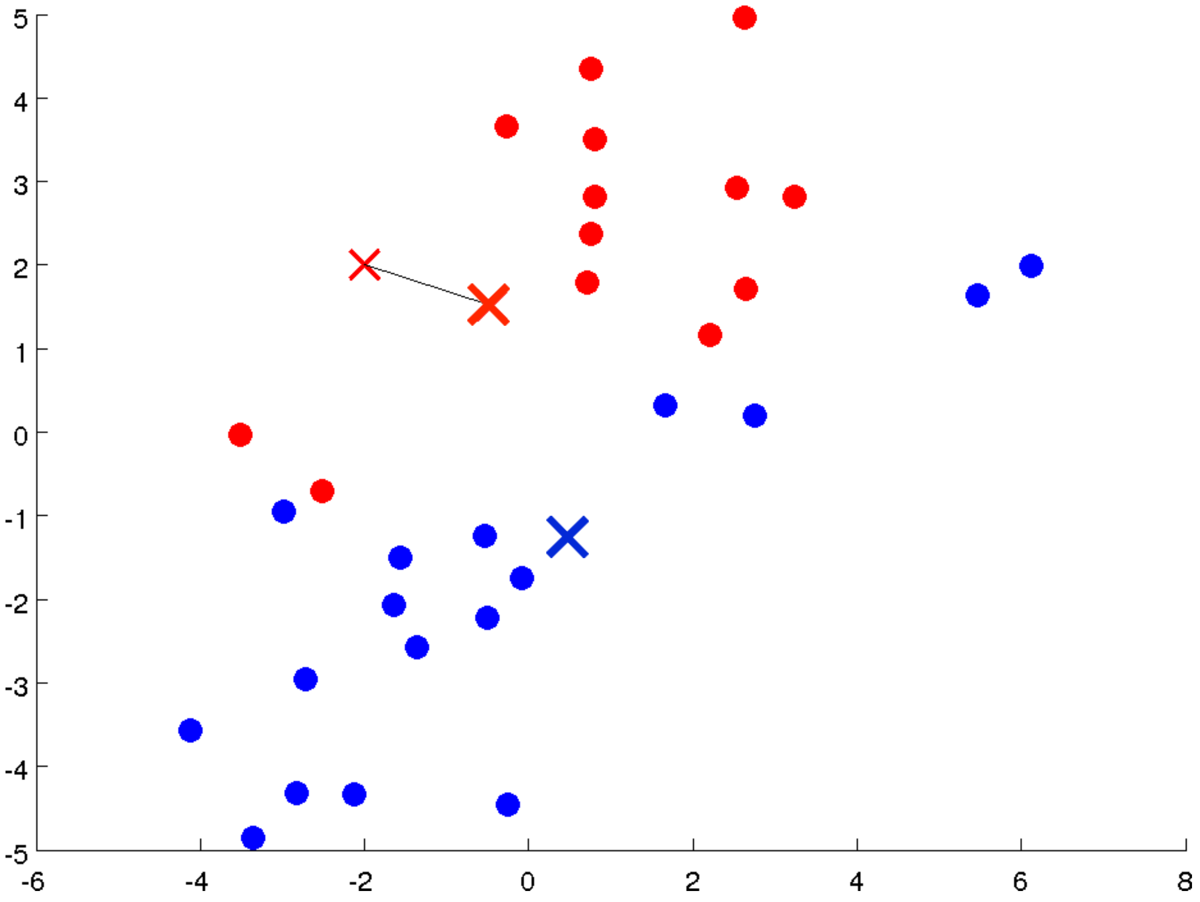


cluster centroids

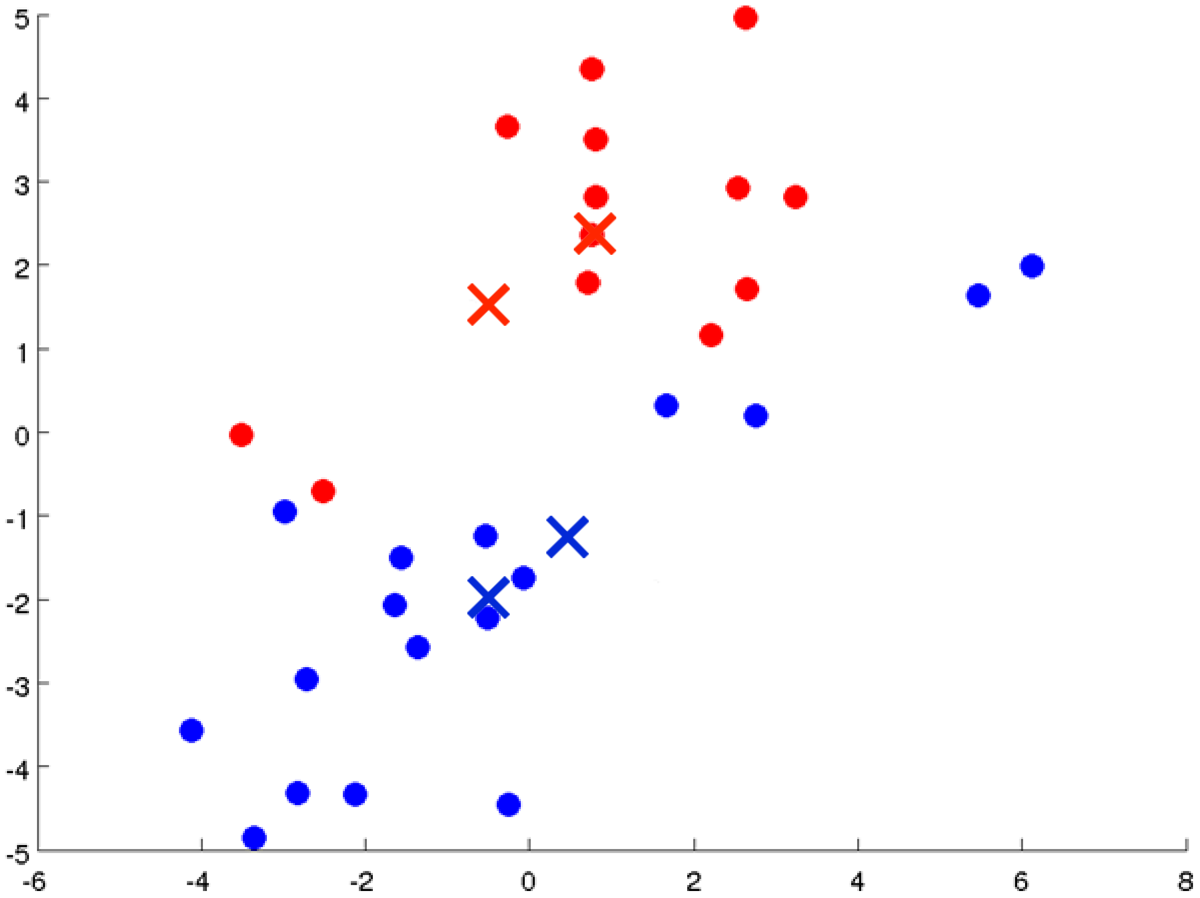


Andrew Ng

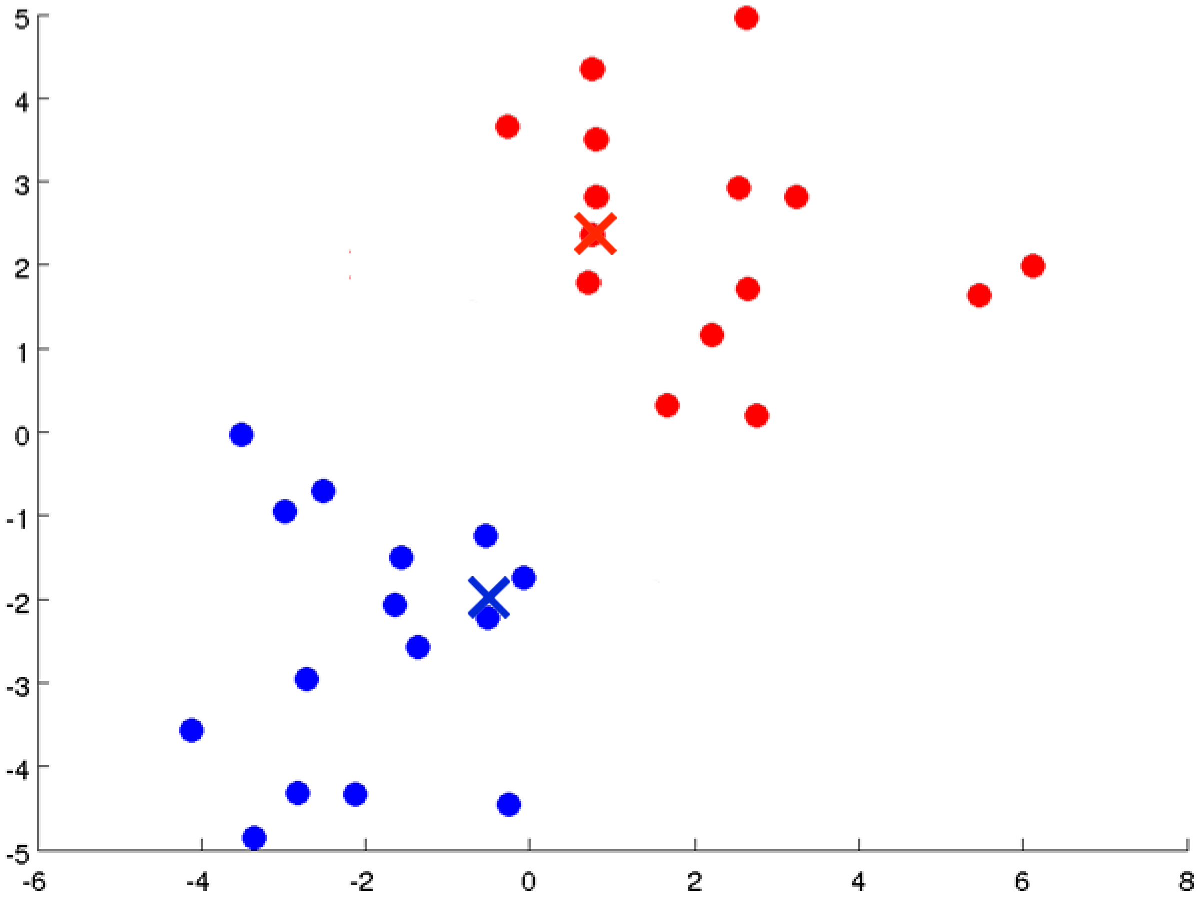

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8



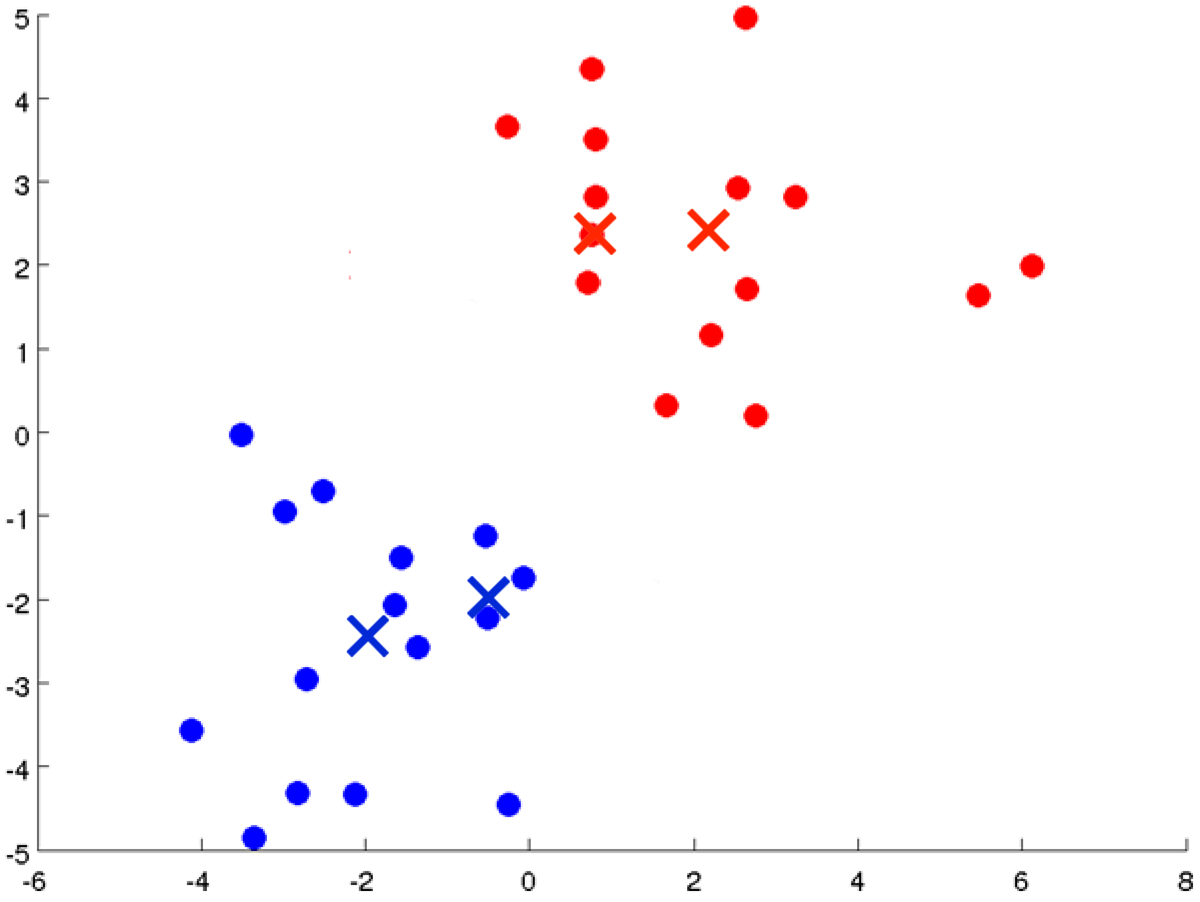
Andrew Ng



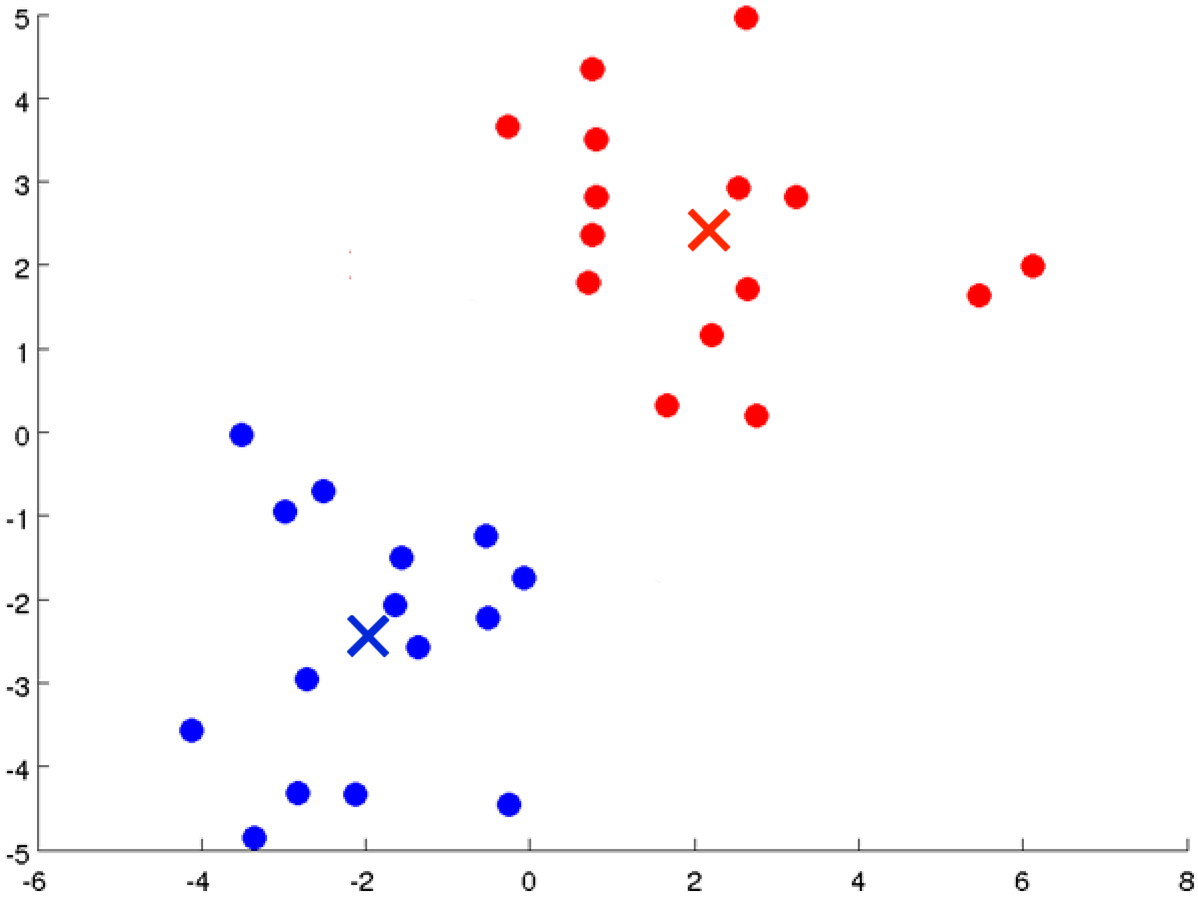
Andrew Ng



Andrew Ng



5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8



Andrew Ng

K-means algorithm

Input:

- K (number of clusters)
- Training set $\{x^{(1)}, x^{(2)}, \ldots, x^{(m)}\}$

$x^{(i)} \in \mathbb{R}^n$ (drop $x_0 = 1$ convention)

K–means algorithm

Randomly initialize K cluster centroids $\mu_1, \mu_2, \ldots, \mu_K \in \mathbb{R}^n$

Repeat {

Cluster assignment step

for i = 1 to m

$c^{(i)}$:= index (from 1 to K) of cluster centroid closest to $x^{(i)}$

$\min_k \|x^{(i)} - \mu_k\|^2 \rightarrow c^{(i)}$

Move centroid

for k = 1 to K

$\rightarrow \mu_k$:= average (mean) of points assigned to cluster k

$x^{(1)}, x^{(5)}, x^{(6)}, x^{(10)} \rightarrow c^{(1)} = 2, \; c^{(5)} = 2, \; c^{(6)} = 2, \; c^{(10)} = 2$

$$\mu_2 = \frac{1}{4}\left[x^{(1)} + x^{(5)} + x^{(6)} + x^{(10)}\right] \in \mathbb{R}^n$$

}

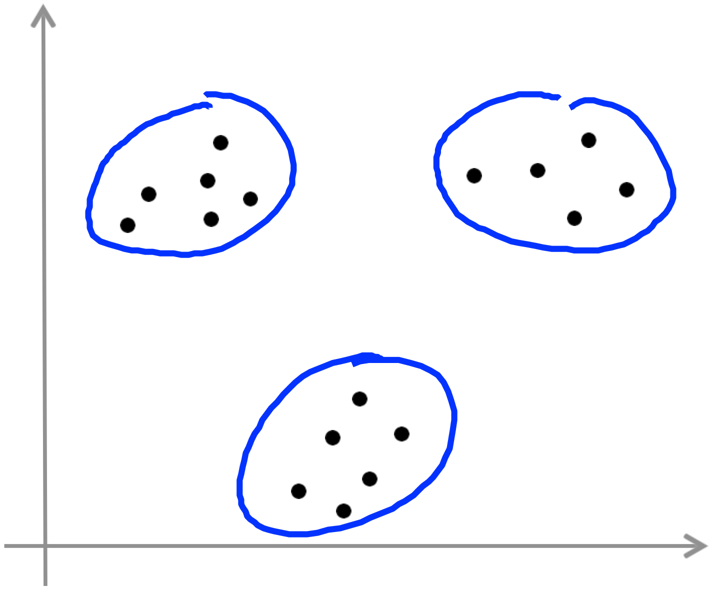
Suppose you run k-means and after the algorithm converges, you have:
$c^{(1)} = 3, c^{(2)} = 3, c^{(3)} = 5, \ldots$

Which of the following statements are true? Check all that apply.

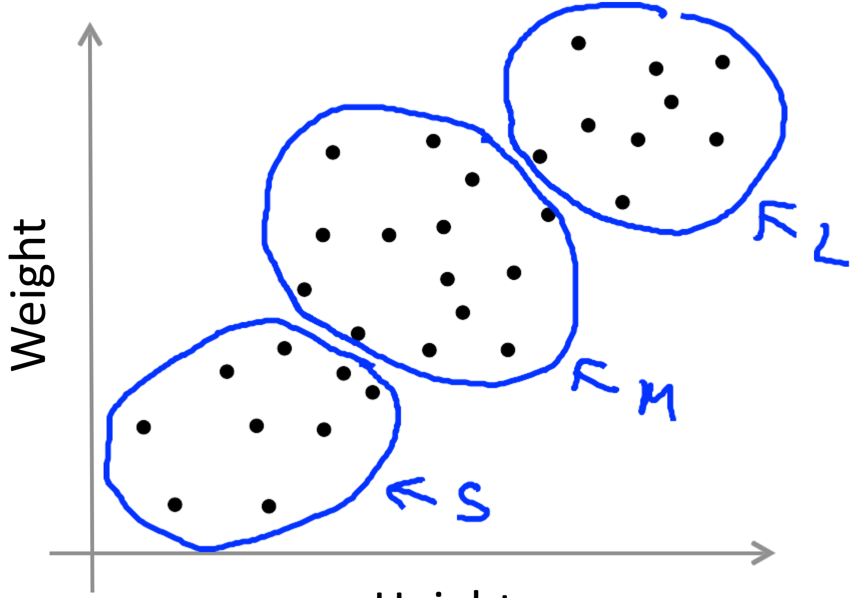
- [ ] The third example $x^{(3)}$ has been assigned to cluster 5.
- [ ] The first and second training examples $x^{(1)}$ and $x^{(2)}$ have been assigned to the same cluster.
- [ ] The second and third training examples have been assigned to the same cluster.
- [ ] Out of all the possible values of $k \in \{1, 2, \ldots, K\}$ the value $k = 3$ minimizes $\|x^{(2)} - \mu_k\|^2$.

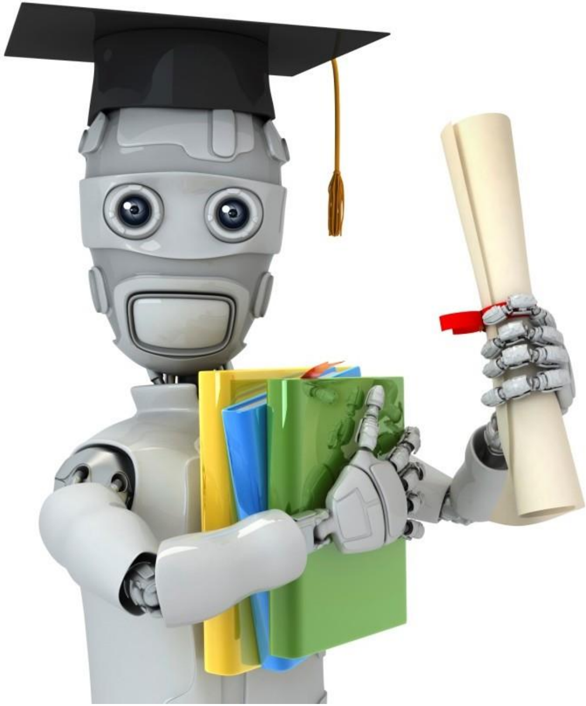
K-means for non-separated clusters

S, M, L



T-shirt sizing





Machine Learning

Clustering

Optimization objective

K-means optimization objective

$c^{(i)}$ = index of cluster (1,2,…,K) to which example $x^{(i)}$ is currently assigned

μ_k = cluster centroid k ($\mu_k \in \mathbb{R}^n$)

K $k \in \{1, 2, \ldots, K\}$

$\mu_{c^{(i)}}$ = cluster centroid of cluster to which example $x^{(i)}$ has been assigned

$x^{(i)} \rightarrow 5$ $c^{(i)} = 5$ $\mu_{c^{(i)}} = \mu_5$

Optimization objective:

$$J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K) = \frac{1}{m} \sum_{i=1}^{m} ||x^{(i)} - \mu_{c^{(i)}}||^2$$

$$\min_{\substack{c^{(1)}, \ldots, c^{(m)}, \\ \mu_1, \ldots, \mu_K}} J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K)$$

K-means algorithm

Randomly initialize K cluster centroids $\mu_1, \mu_2, \ldots, \mu_K \in \mathbb{R}^n$

Cluster assignment step

Minimize $J(\ldots)$ wrt $\boxed{c^{(1)}, c^{(2)}, \ldots, c^{(m)}}$

(holding $\mu_1, \ldots, \mu_K$ fixed)

Repeat {

for i = 1 to m

$c^{(i)}$:= index (from 1 to K) of cluster centroid closest to $x^{(i)}$

move centroid

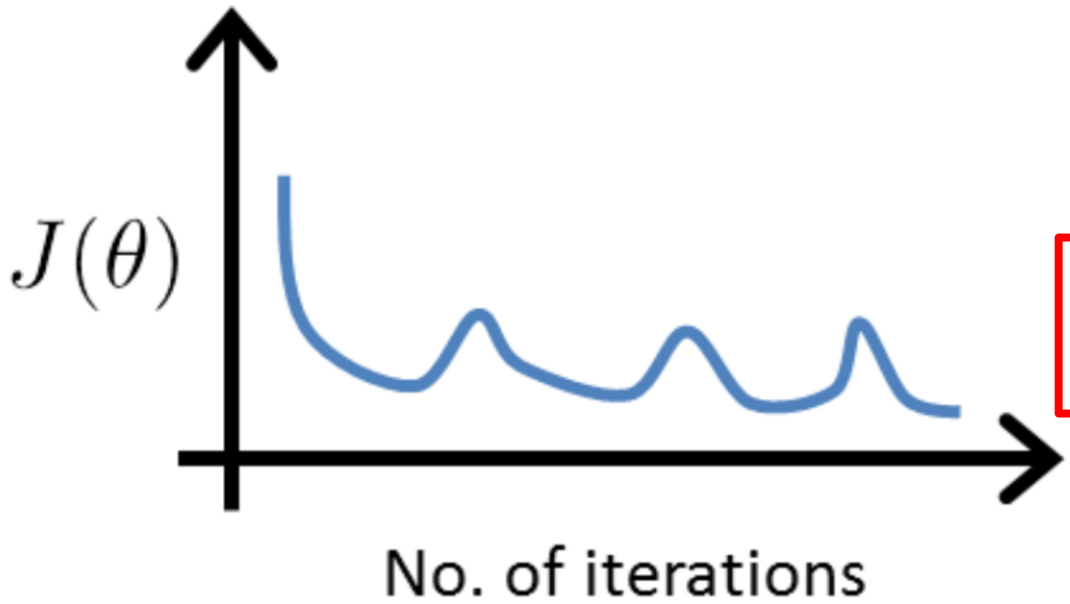
for k = 1 to K

μ_k := average (mean) of points assigned to cluster k

}

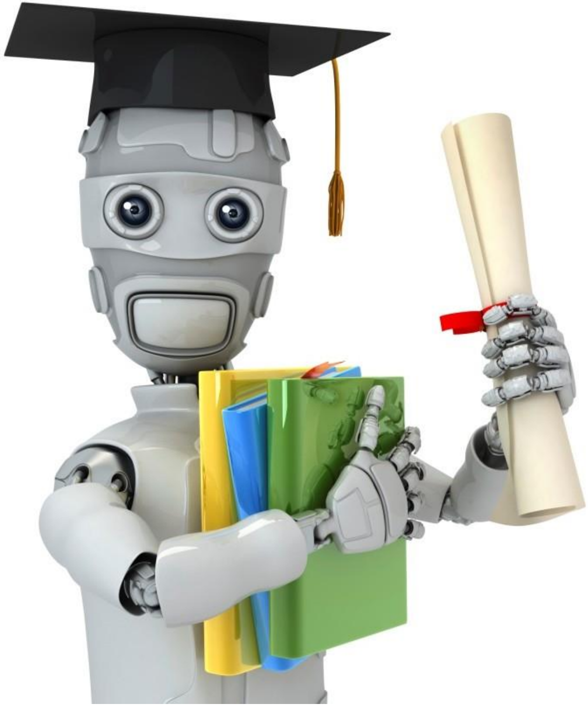
minimize $J(\ldots)$ wrt $\boxed{\mu_1, \ldots, \mu_K}$

Suppose you have implemented k-means and to check that it is running correctly, you plot the cost function $J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_k)$ as a function of the number of iterations. Your plot looks like this:



What does this mean?

- The learning rate is too large.
- The algorithm is working correctly.
- The algorithm is working, but k is too large.
- It is not possible for the cost function to sometimes increase. There must be a bug in the code.



Machine Learning

Clustering

Random initialization

K–means algorithm

Randomly initialize K cluster centroids $\mu_1, \mu_2, \ldots, \mu_K \in \mathbb{R}^n$

Repeat {

for i = 1 to m

$c^{(i)}$:= index (from 1 to K) of cluster centroid closest to $x^{(i)}$

for k = 1 to K

μ_k := average (mean) of points assigned to cluster k

}

Random initialization

$K=2$

Should have $K < m$

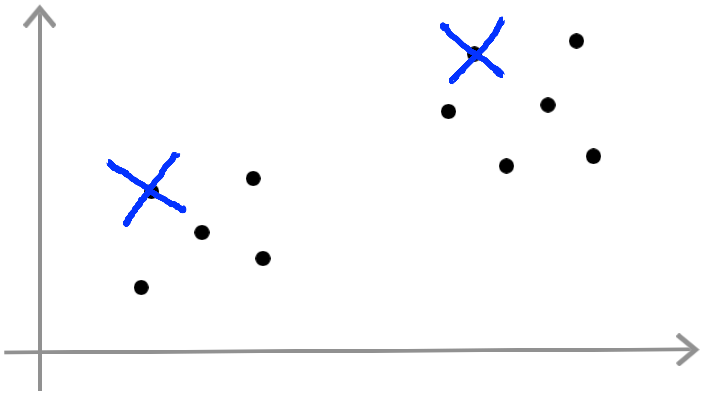
Randomly pick K training examples.

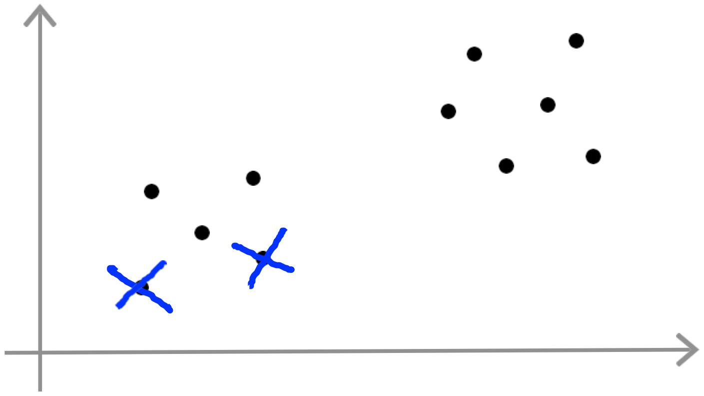
Set $\mu_1, \ldots, \mu_K$ equal to these K examples.

$\mu_1 = x^{(i)}$

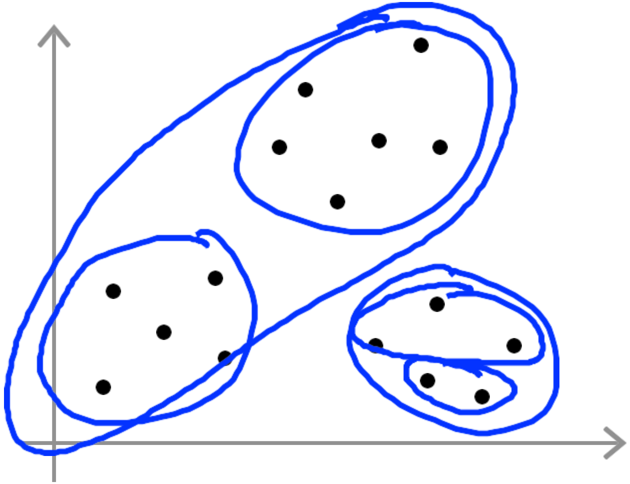
$\mu_2 = x^{(j)}$

$\vdots$

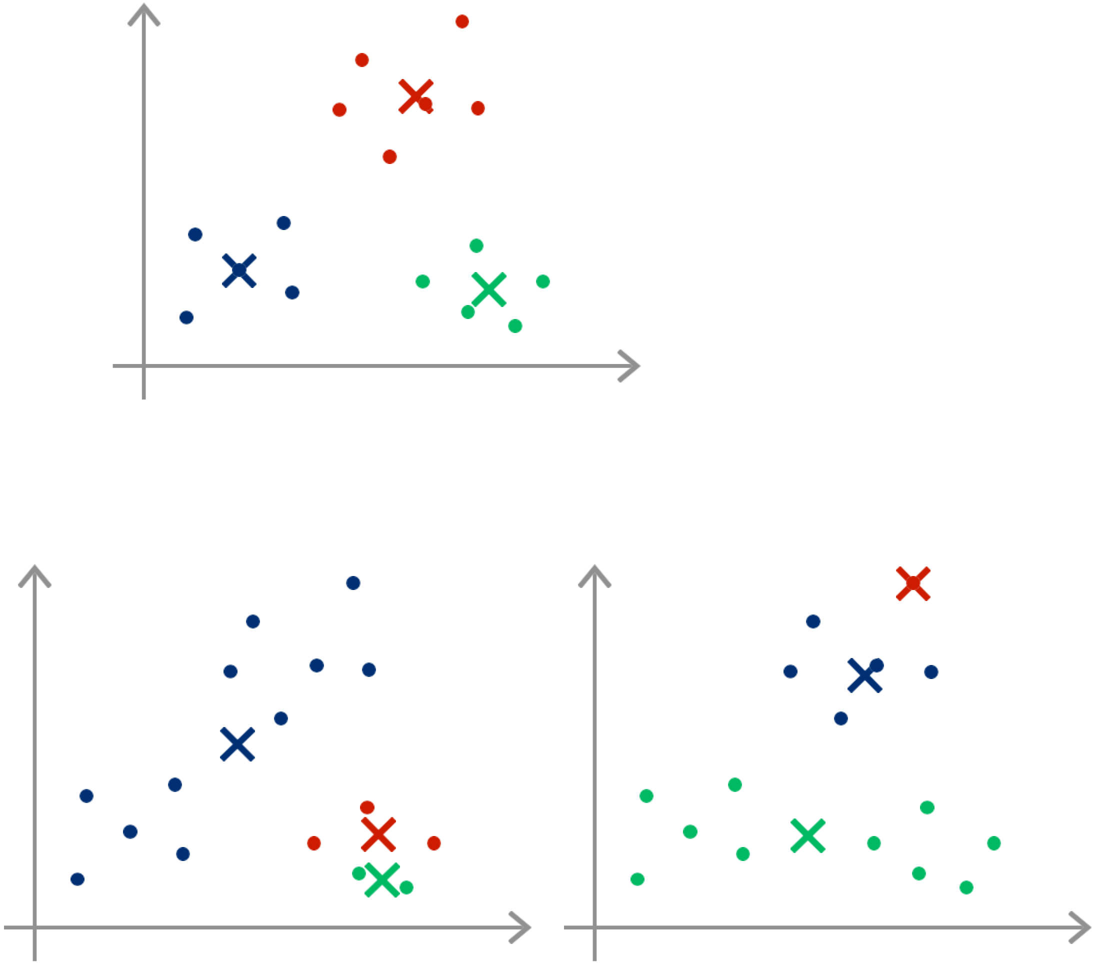




Local optima



$$J(c^{(1)},\dots,c^{(m)},\mu_1,\dots,\mu_K)$$



Random initialization

For i = 1 to 100 {

Randomly initialize K--means.

Run K--means. Get $c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K$.

Compute cost function (distortion)

$$J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K)$$

}

Pick clustering that gave lowest cost $J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K)$

Which of the following is the recommended way to initialize k-means?

- Pick a random integer i from $\{1, \ldots, k\}$. Set $\mu_1 = \mu_2 = \cdots = \mu_k = x^{(i)}$.

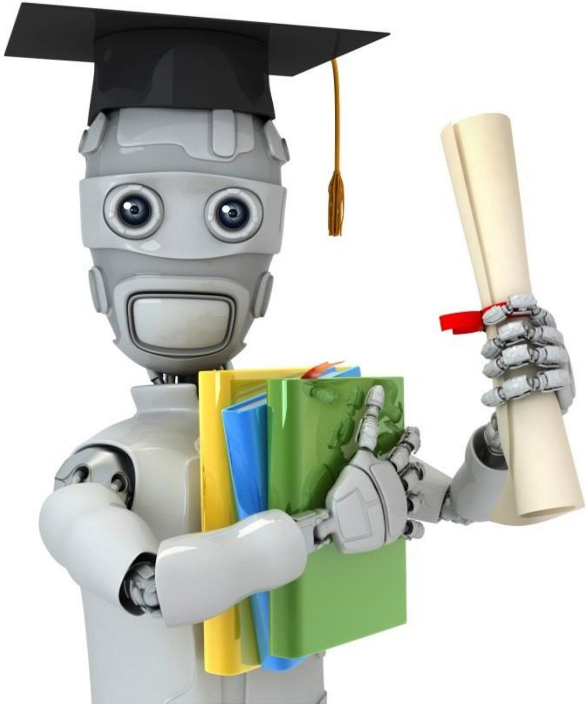
- Pick k distinct random integers $i_1, \ldots, i_k$ from $\{1, \ldots, k\}$.

  Set $\mu_1 = x^{(i_1)}, \mu_2 = x^{(i_2)}, \ldots, \mu_k = x^{(i_k)}$.

- Pick k distinct random integers $i_1, \ldots, i_k$ from $\{1, \ldots, m\}$.

  Set $\mu_1 = x^{(i_1)}, \mu_2 = x^{(i_2)}, \ldots, \mu_k = x^{(i_k)}$.

- Set every element of $\mu_i \in \mathbb{R}^n$ to a random value between $-\epsilon$ and ϵ, for some small ϵ.

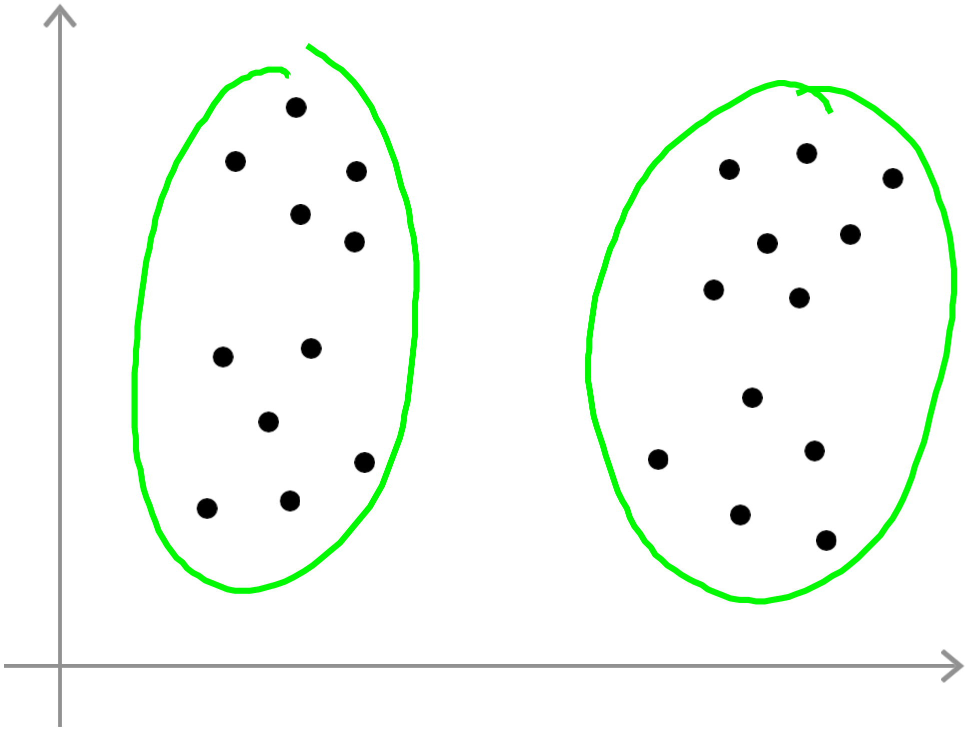


Machine Learning

Clustering

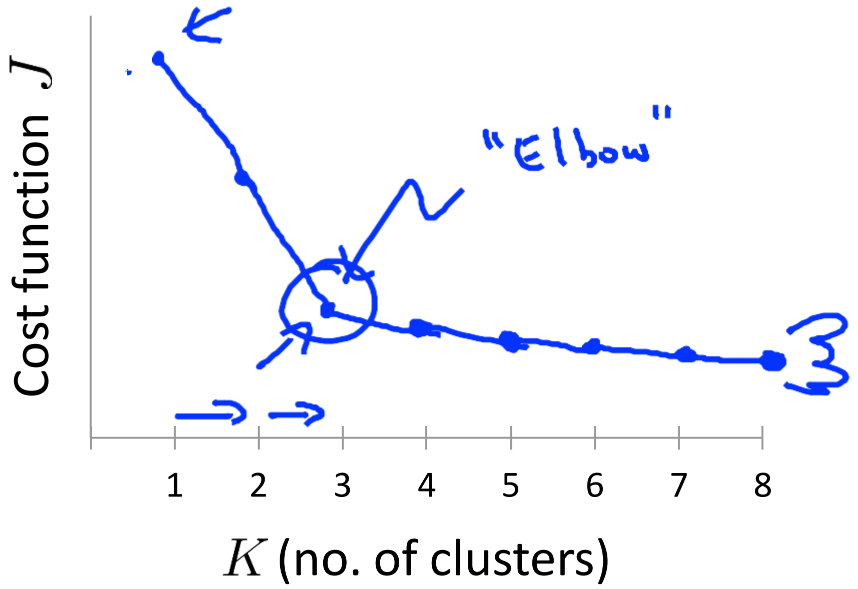
Choosing the number of clusters

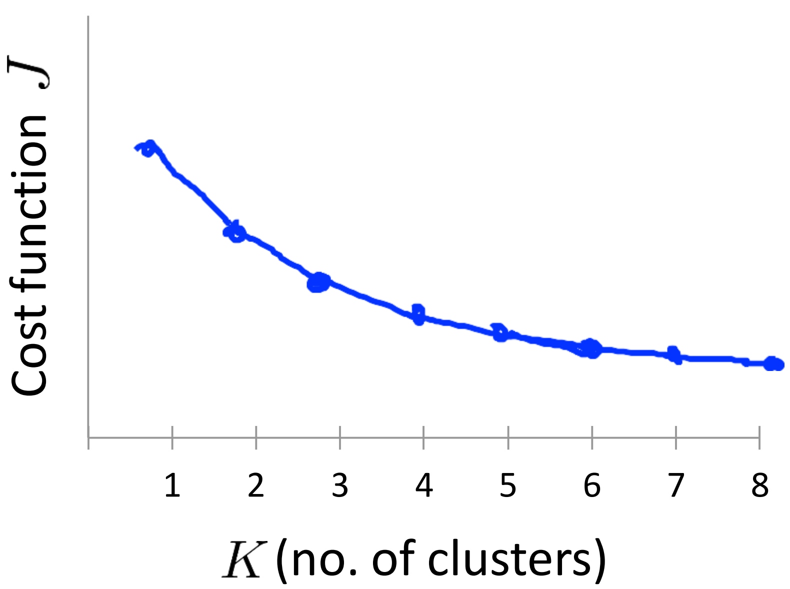
What is the right value of K?



Choosing the value of K

Elbow method:





Suppose you run k-means using k = 3 and k = 5. You find that the cost function J is much higher for k = 5 than for k = 3. What can you conclude?

- This is mathematically impossible. There must be a bug in the code.
- The correct number of clusters is k = 3.
- In the run with k = 5, k-means got stuck in a bad local minimum. You should try re-running k-means with multiple random initializations.
- In the run with k = 3, k-means got lucky. You should try re-running k-means with k = 3 and different random initializations until it performs no better than with k = 5.

Choosing the value of K

Sometimes, you're running K-means to get clusters to use for some later/downstream purpose. Evaluate K-means based on a metric for how well it performs for that later purpose.

K=3 S, M, L   K=5 XS, S, M, L, XL

E.g.