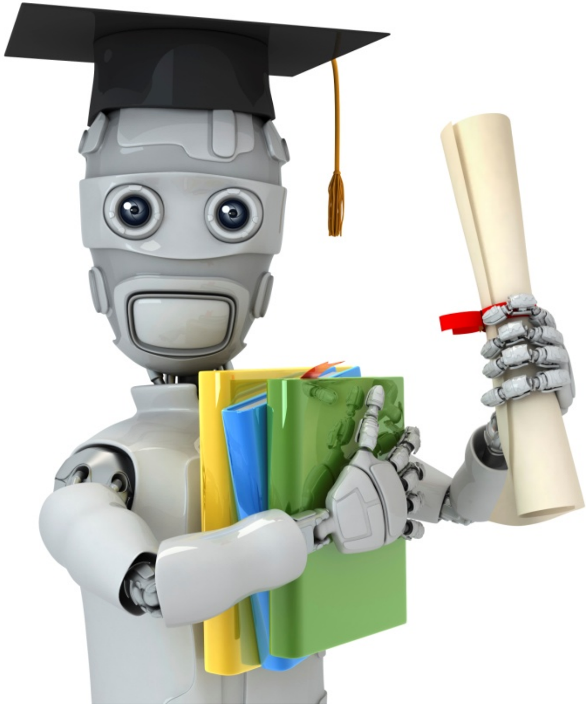
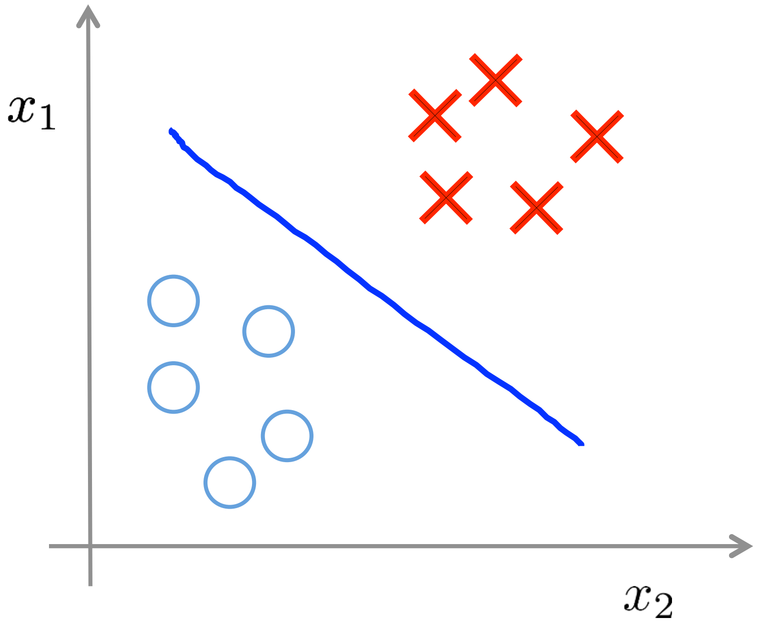


Machine Learning

Clustering

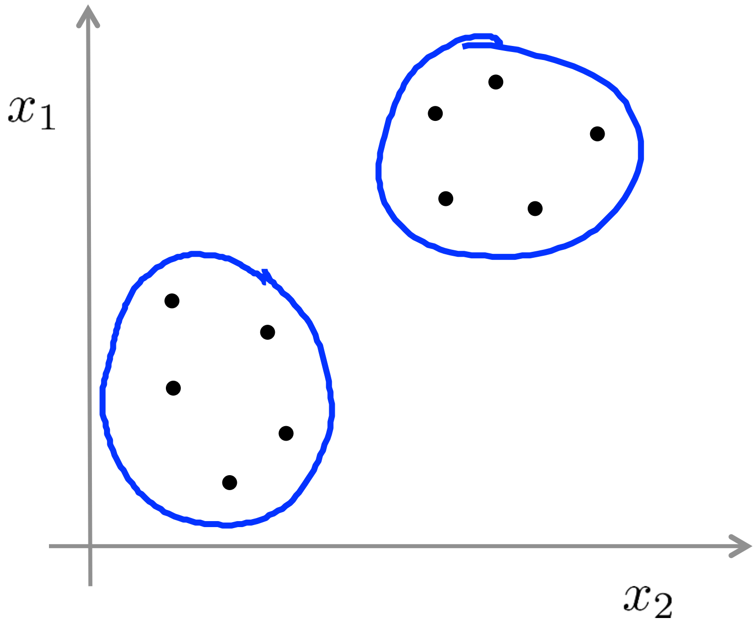
Unsupervised learning introduction

Supervised learning



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

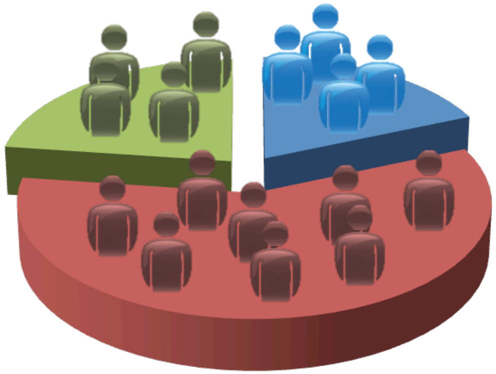
Unsupervised learning



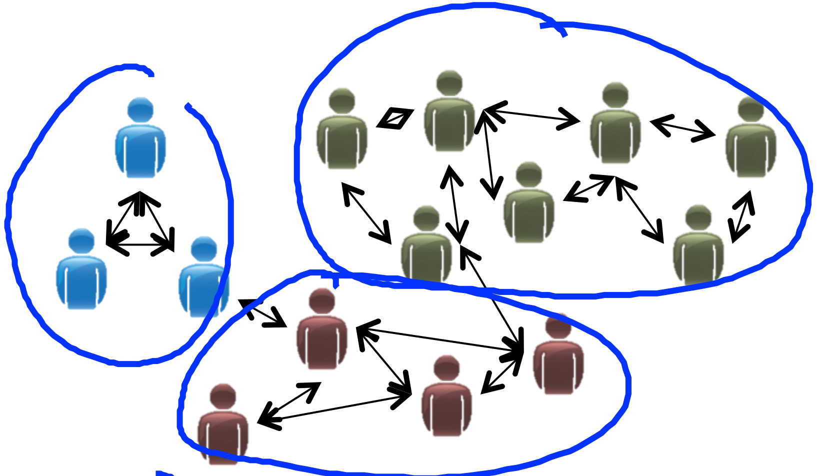
Clustering algorithm

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

Applications of clustering



Market segmentation



Social network analysis

Organize computing clusters

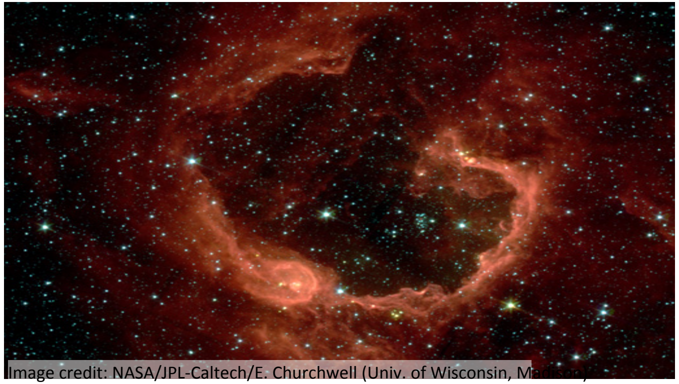
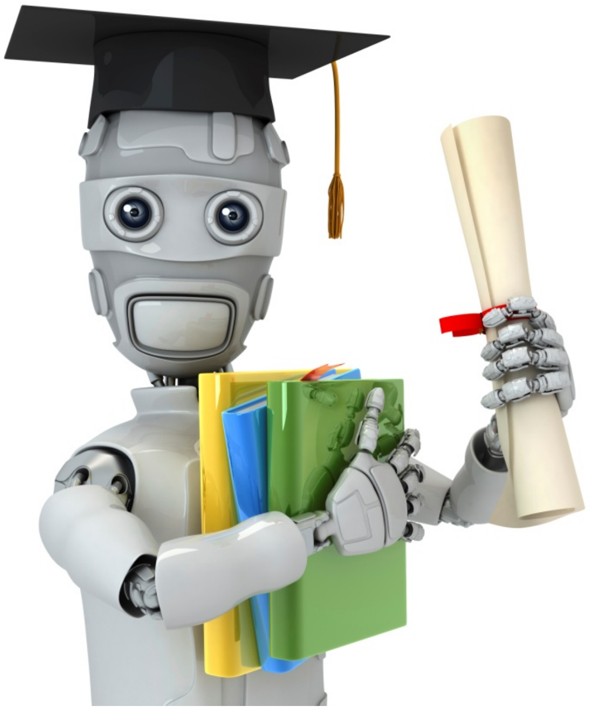


Image credit: NASA/JPL-Caltech/E. Churchwell (Univ. of Wisconsin, M

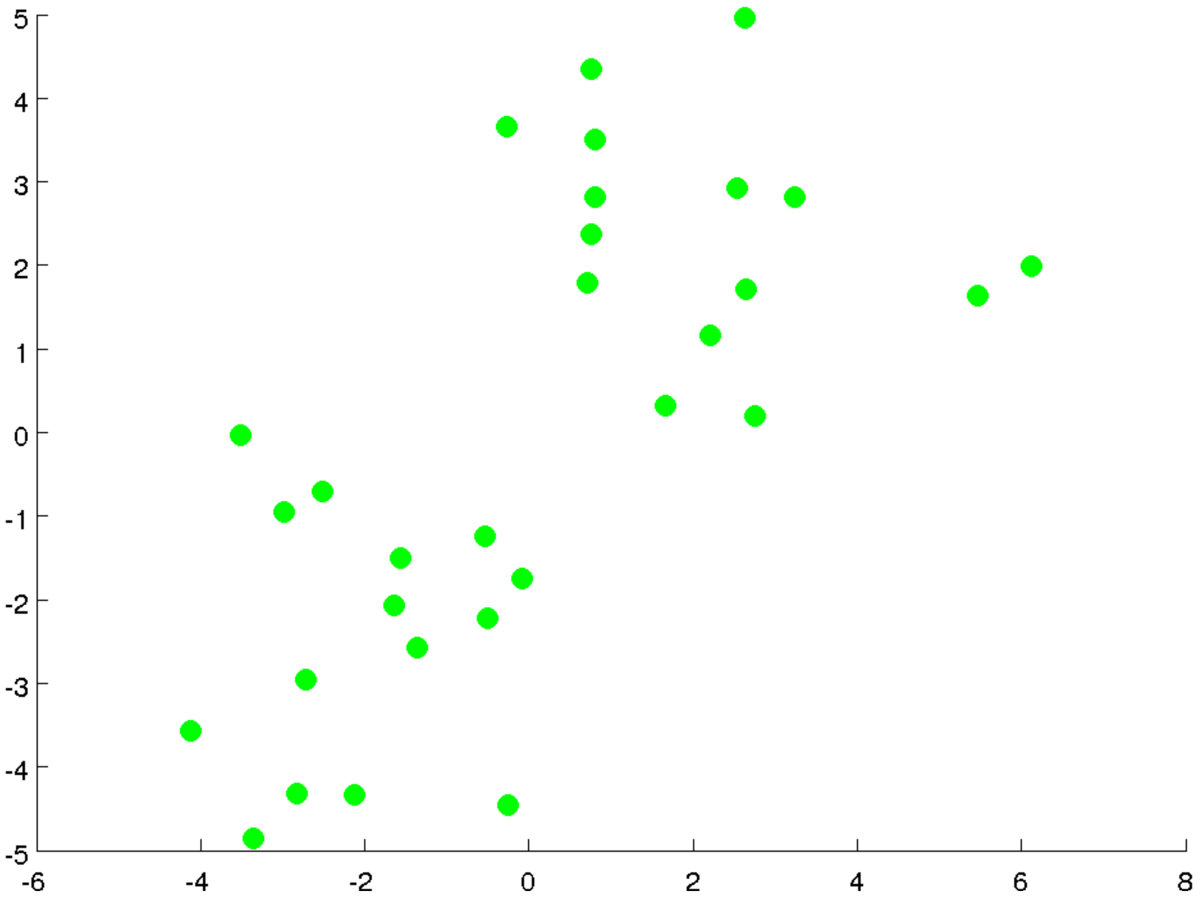
Astronomical data analysis



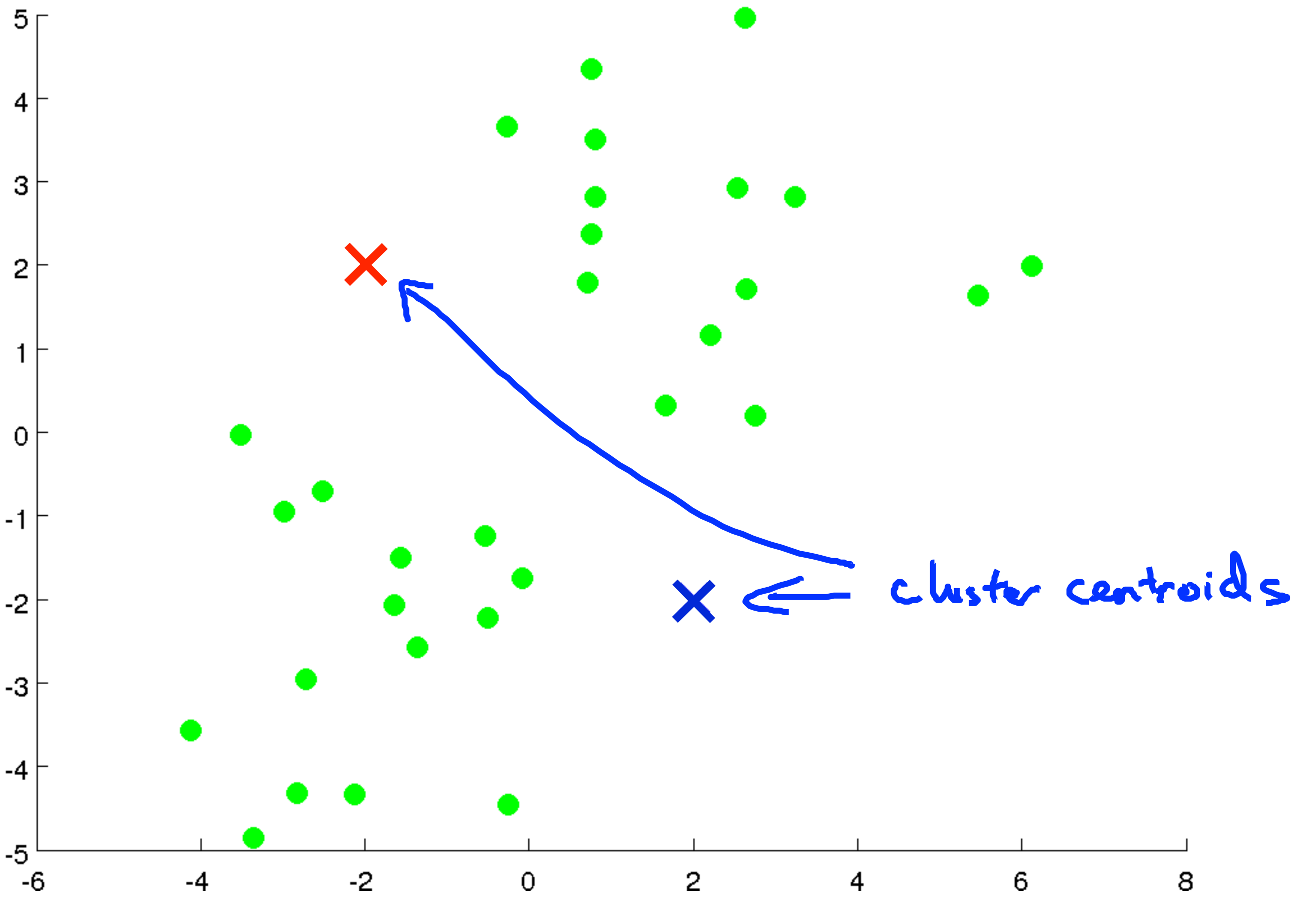
Machine Learning

Clustering

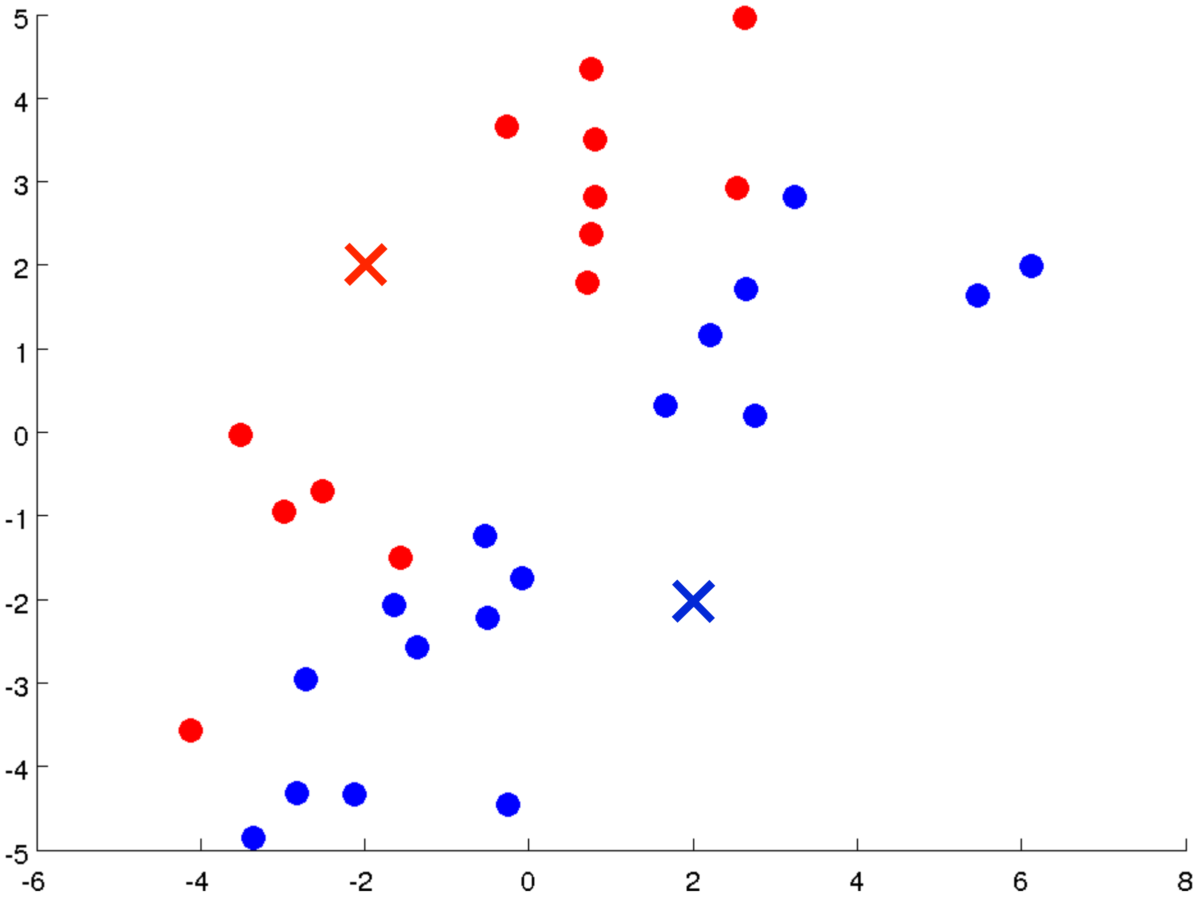
K-means algorithm



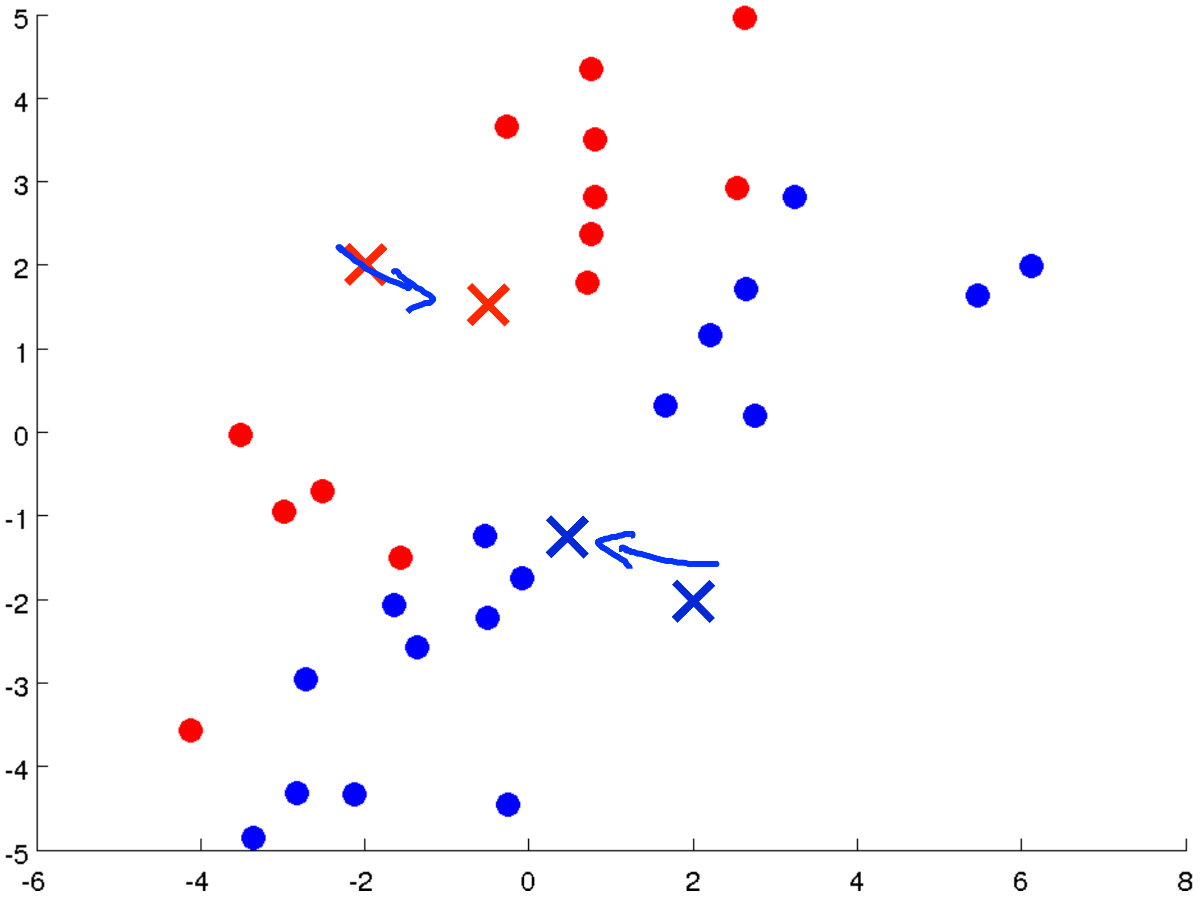
Andrew Ng



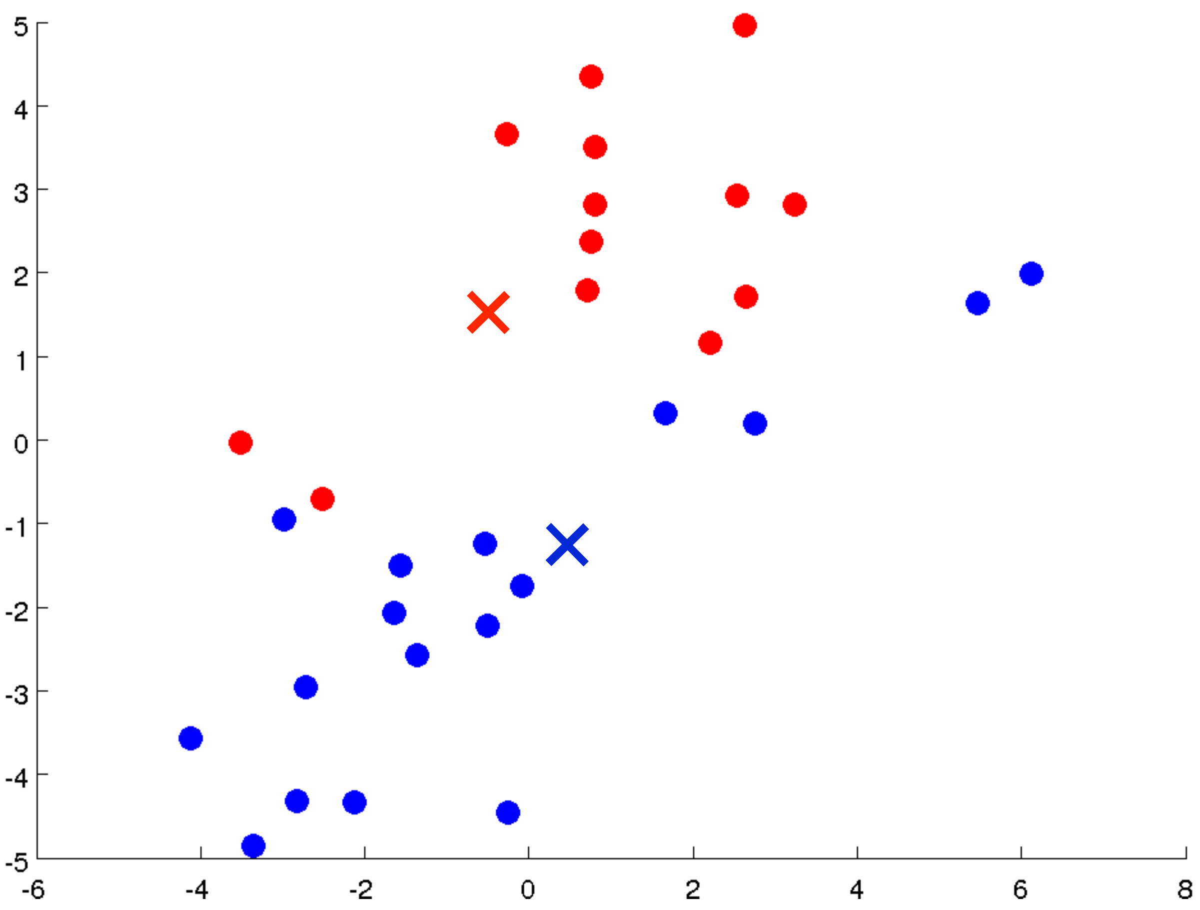
cluster centroids
5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8



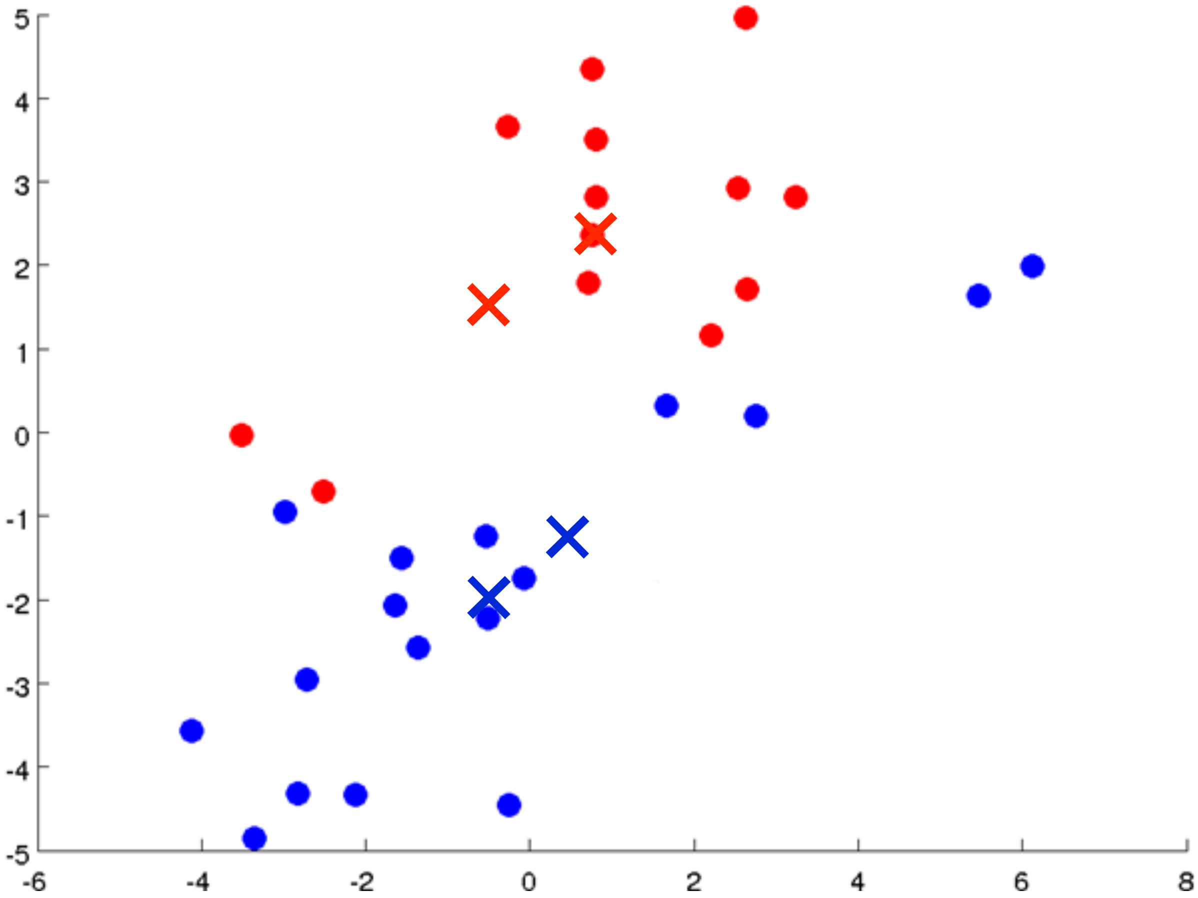
5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8

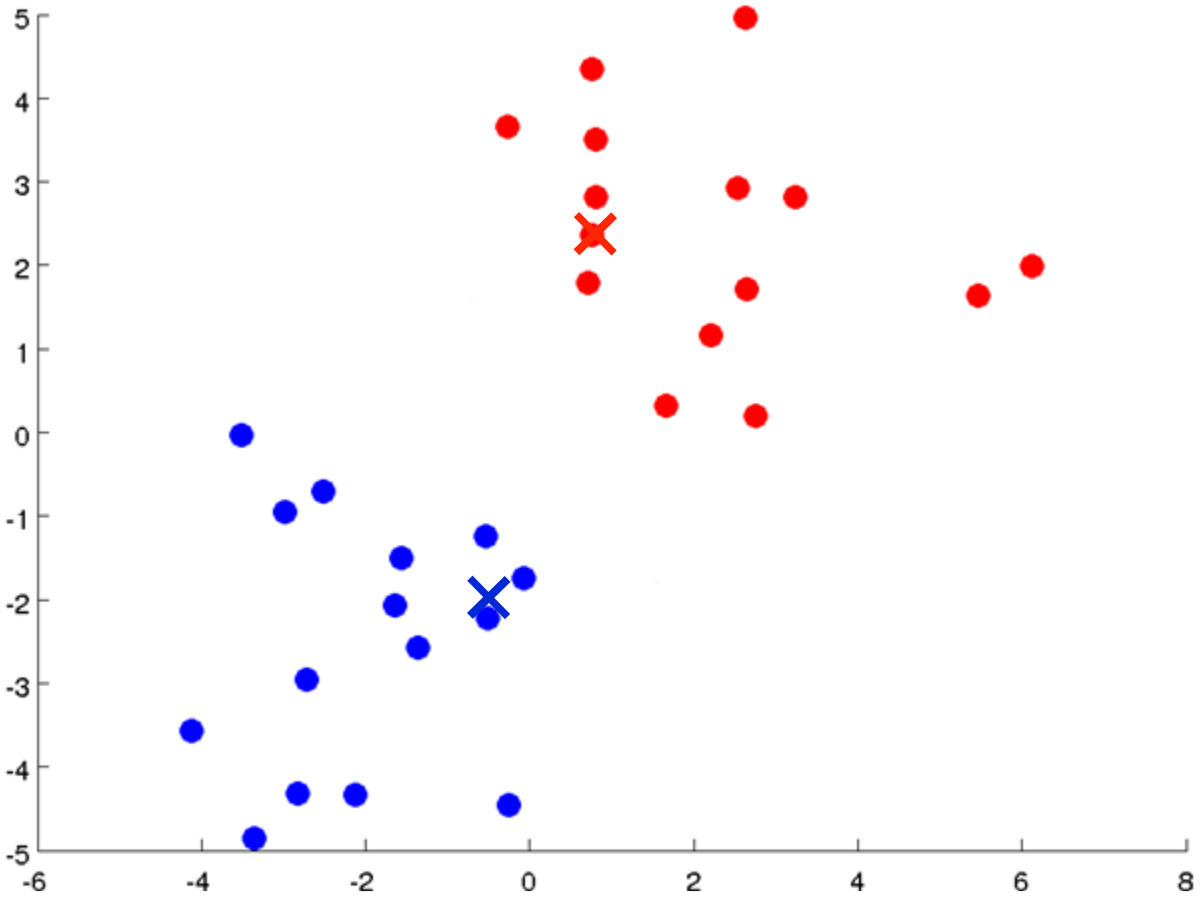


Andrew Ng

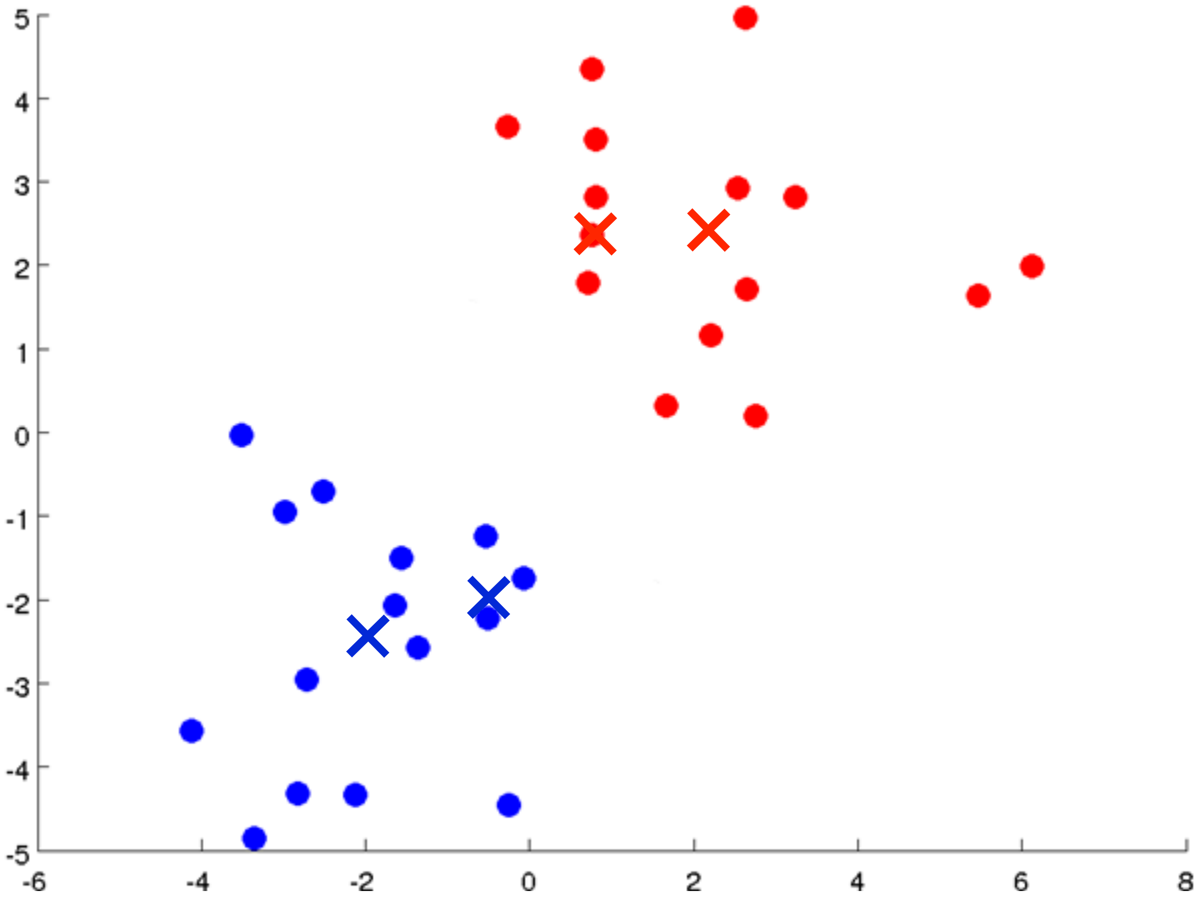


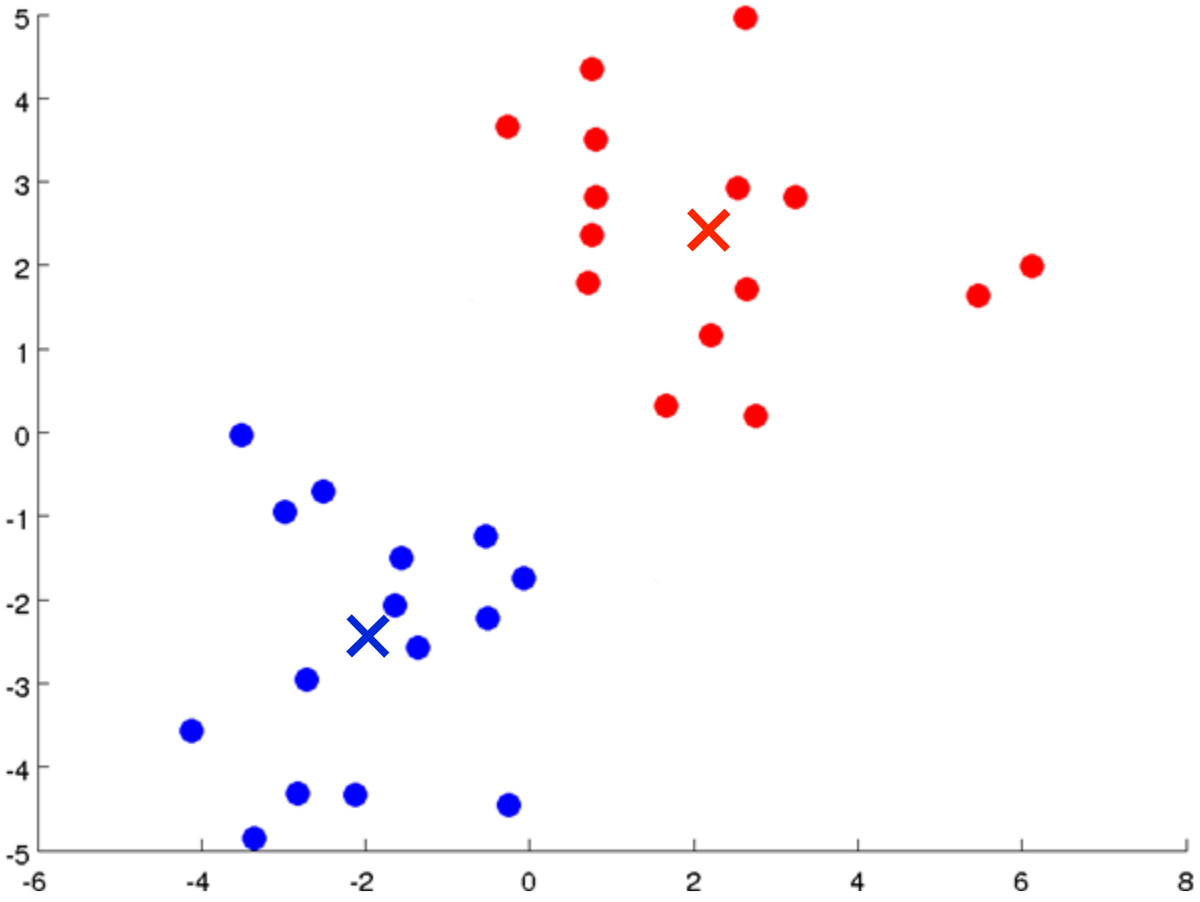
5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8


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



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


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

μ_1 $\times$ μ_2 $\times$

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

Repeat {

Cluster assignment step

for i = 1 to m assign each point to the closest centriod

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

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

$k \to c^{(i)}$

Move centroid

for k = 1 to K

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

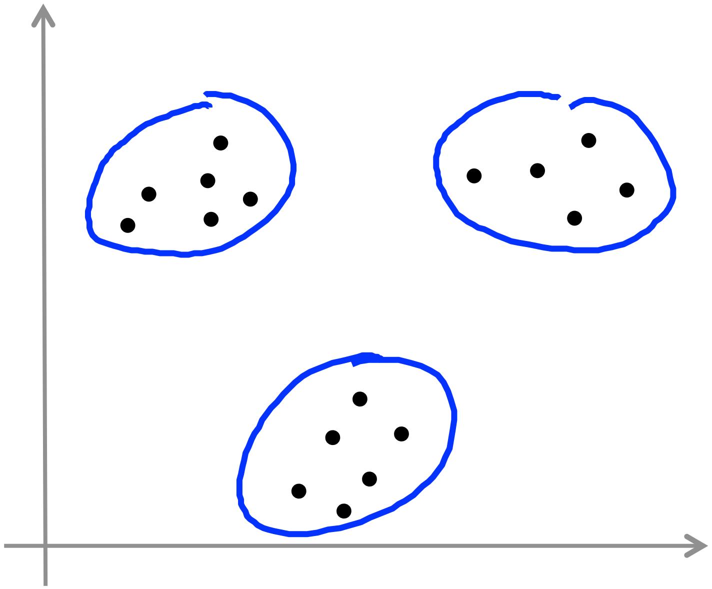
$x^{(1)}, x^{(5)}, x^{(6)}, x^{(10)}$ → $c^{(1)}=2, c^{(5)}=2, c^{(6)}=2, c^{(10)}=2$

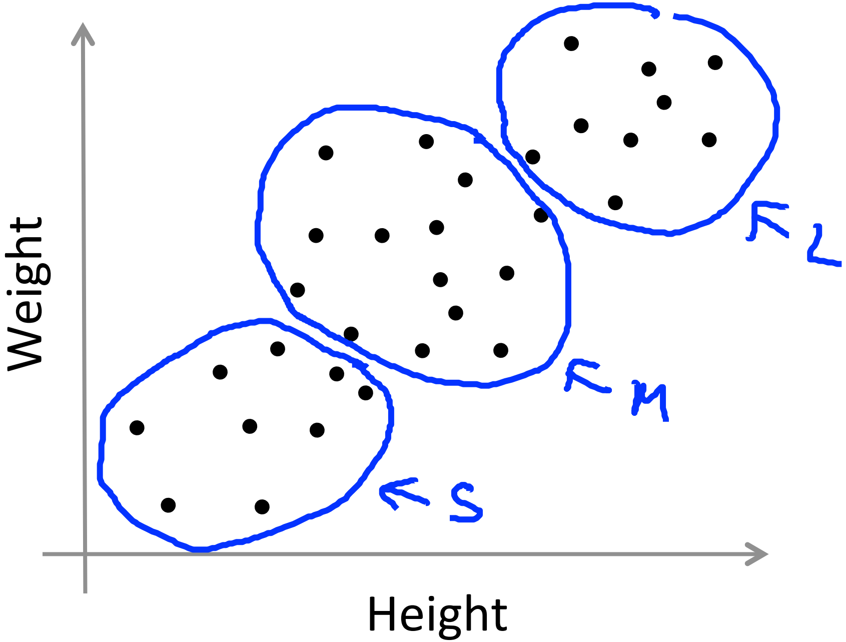
$$\mu_2 = \frac{1}{4}\left[x^{(1)} + x^{(5)} + x^{(6)} + x^{(10)}\right] \in \mathbb{R}^n$$

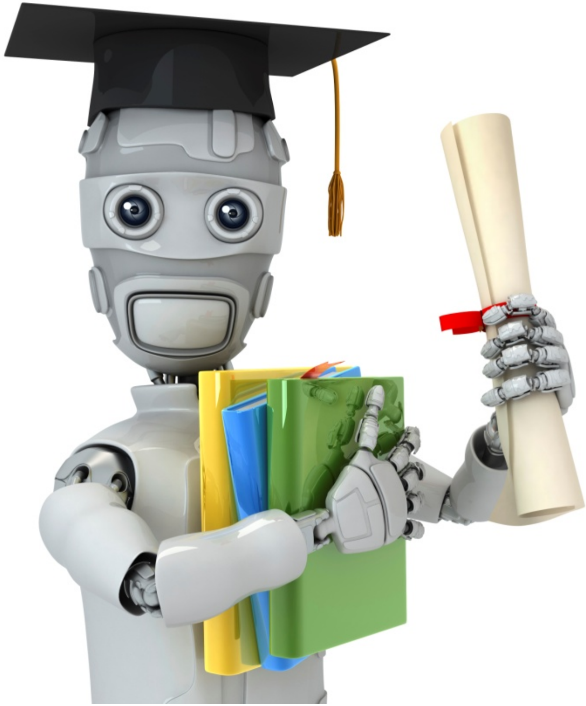
}

K-means for non-separated clusters

S, M, L







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 $\underline{k}$ ($\mu_k \in \mathbb{R}^n$) K $k \in \{1,2,\ldots,K\}$

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

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

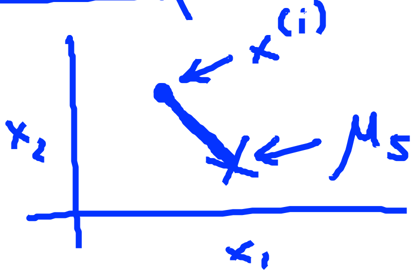
Optimization objective:

cohesiveness

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

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

Distortion



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 $x^{(i)}$

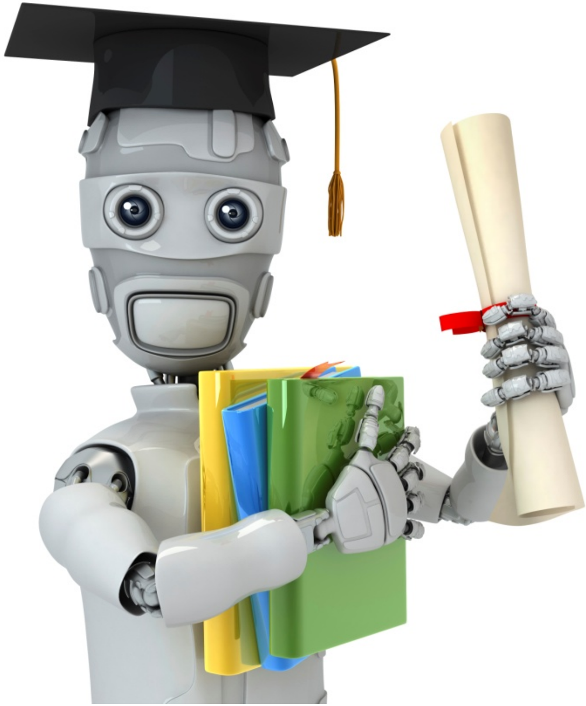
move centroid

for k = 1 to K

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

}

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



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 $K < m$

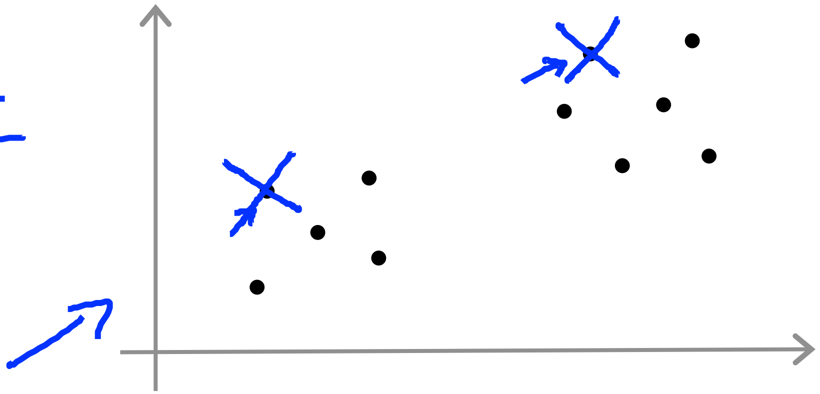
Randomly pick K training examples.

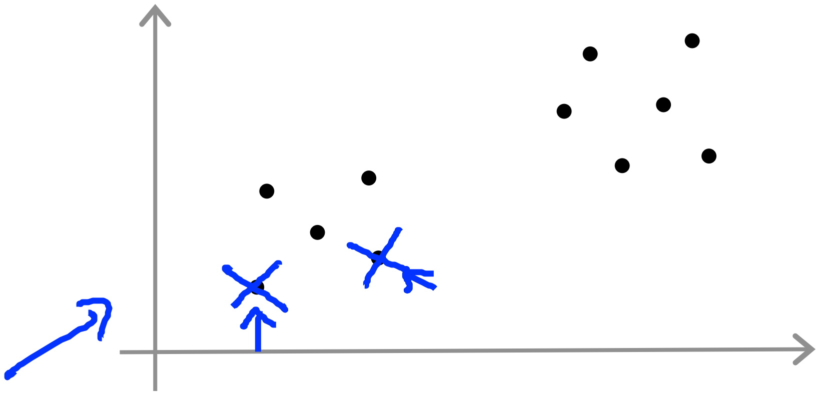
Set $\mu_1, \ldots, \mu_K$ equal to these K examples.

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

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

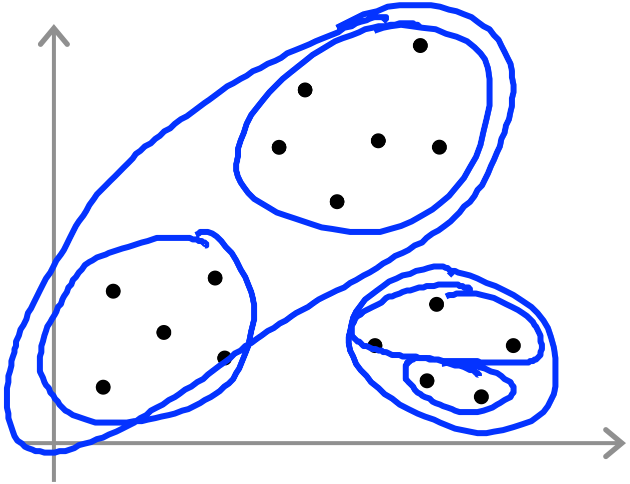
$\vdots$



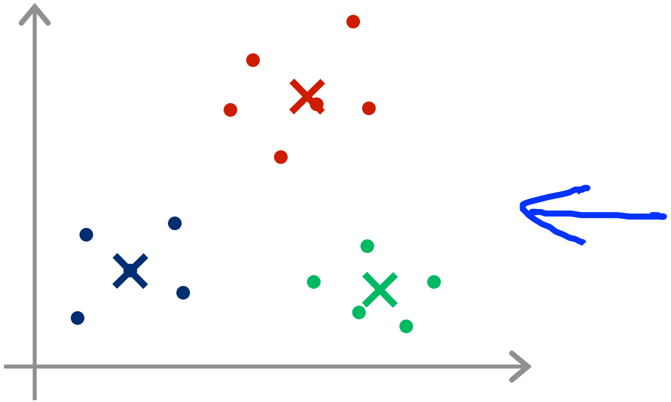


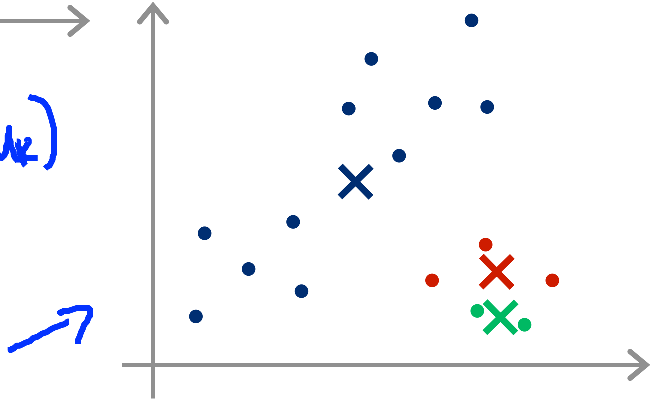
in order to avoid K-means stucked into the local optimation
we should do random initialize lots times

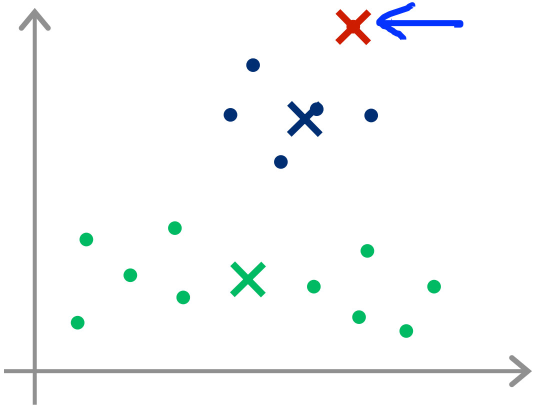
Local optima



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







Random initialization

100 times random initialization

For i = 1 to 100 {

Randomly initialize K-means.

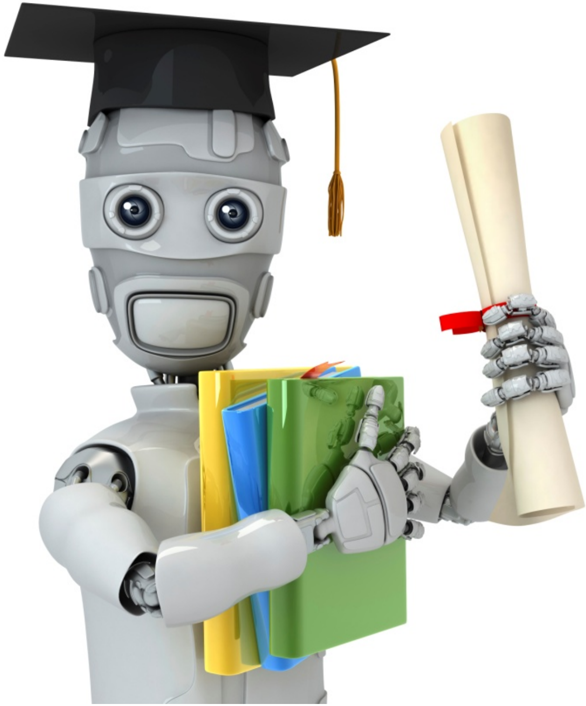
Run K-means. Get $c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K$.

Compute cost function (distortion)

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

}

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

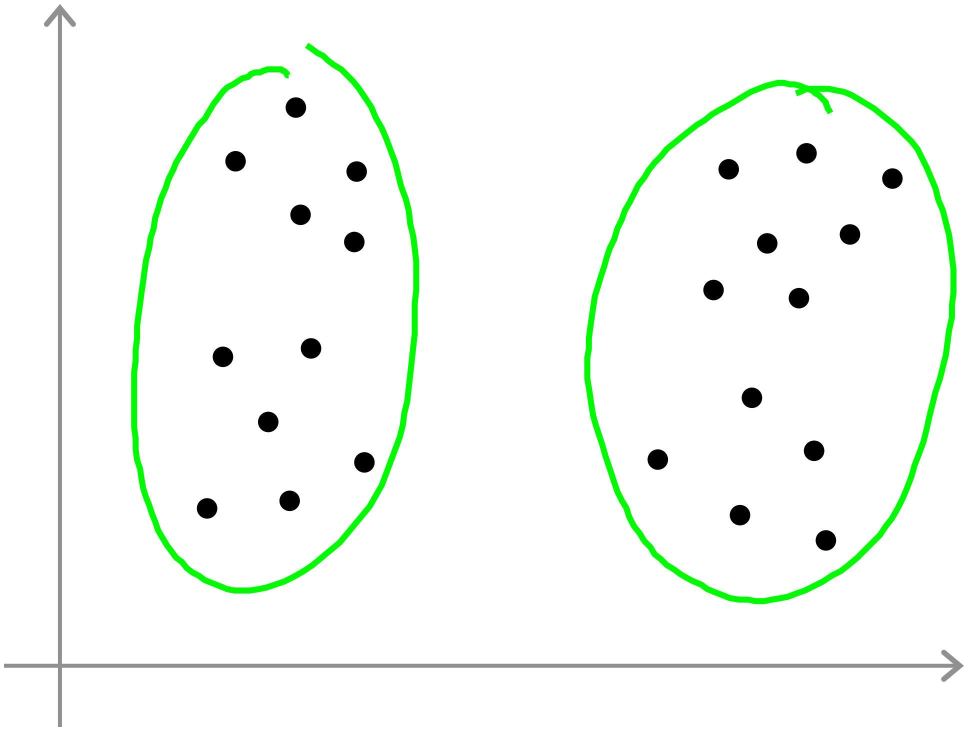


Clustering

Choosing the number of clusters

Machine Learning

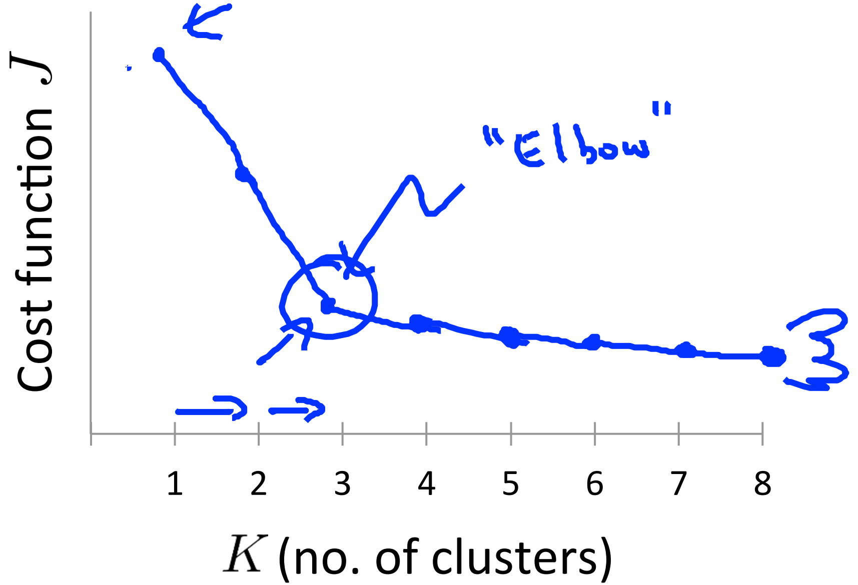
What is the right value of K?

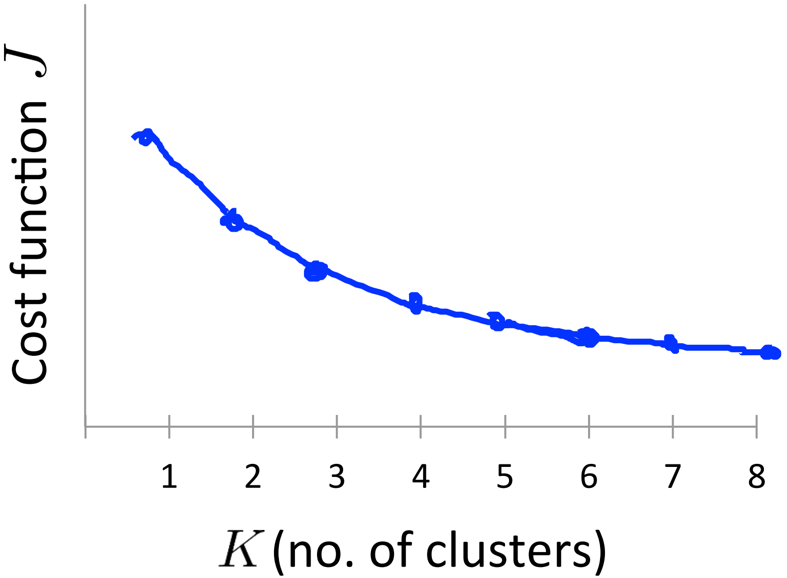


Andrew Ng

Choosing the value of K

Elbow method:





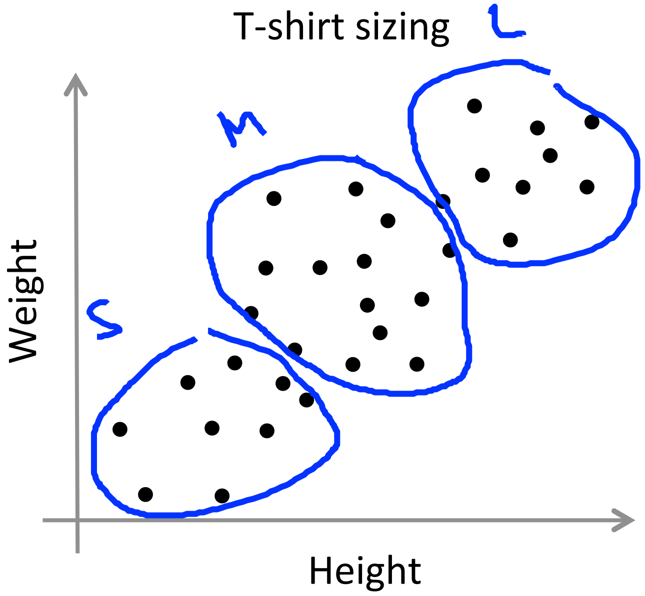
Choosing the value of K

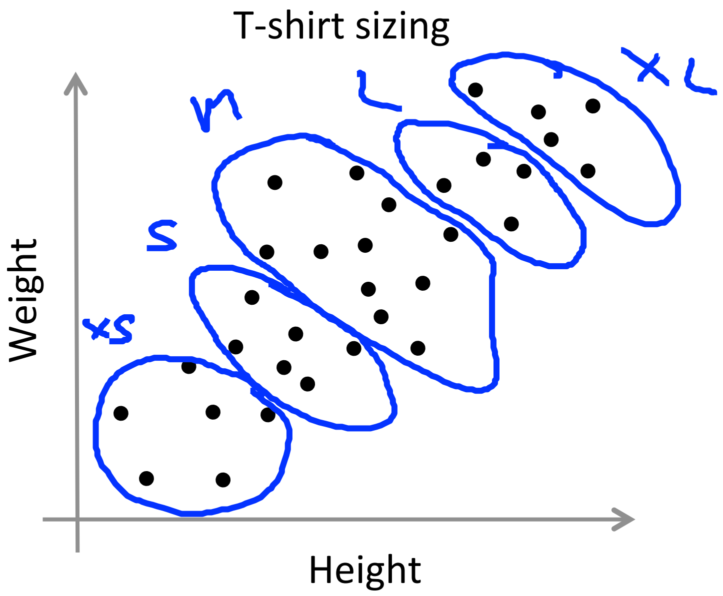
Sometimes, you're running K-means to get clusters to use for some later/downstream purpose. Evaluate K-means based on a metric for how well it performs for that later purpose.

K=3 S, M, L

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

E.g.





Andrew Ng