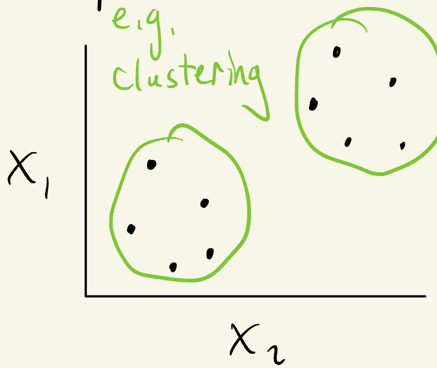
Unsupervised Learning:

May 29, 2021



Training Set: $\{x^{(1)}, x^{(2)} \dots x^{(m)}\}$
→ no $y^{(i)}$
→ find structure/patterns in the data

K-Means Algorithm:

Steps:

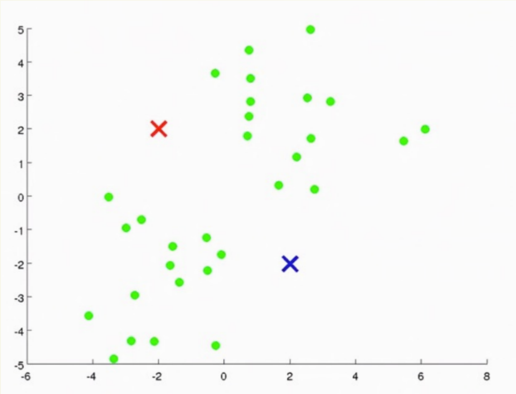
1. Randomly initialize cluster centroids
2. Cluster assignment
3. Move centroids
4. Repeat 2. and 3.

Inputs:

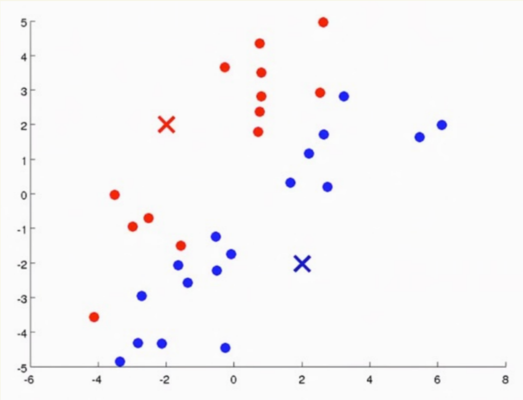
→ K (# clusters)
→ training set $\{x^{(1)}, x^{(2)}, \dots x^{(m)}\}$
↳ $x_0 = 1$ by convention, $x^{(i)} \in \mathbb{R}^n$

e.g.

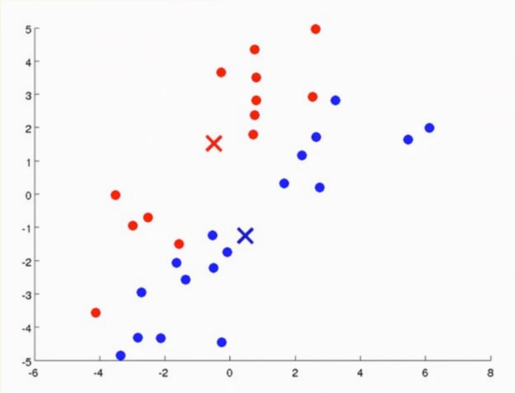
1.



2.



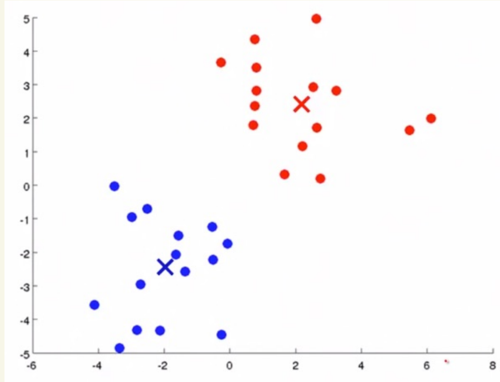
3.



4. repeat 2, 3

Eventually:

Cluster centroids will not move anymore.
K-means has converged.



K-Means Algorithm:

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

Repeat {

cluster assignment: for $i = 1 : m$

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

$\rightarrow \min_k \|x^{(i)} - \mu_k\|, \ k \in [1, K]$

move centroid: for $k = 1 : K$

$\mu_k :=$ mean of points assigned to cluster k

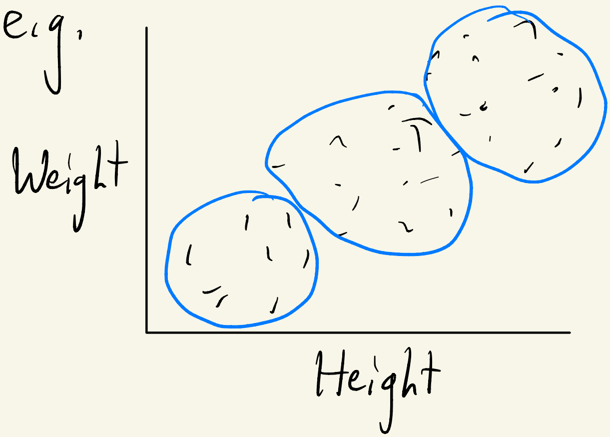
}

$\mu_k \in \mathbb{R}^n$

Non-Separated Clusters:

K-means also works for non-separated clusters.

e.g.



Notation:

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

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

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

e.g. $x^{(i)} \rightarrow$ Cluster 5

$c^{(i)} = 5$

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

Optimization Objective:

a.k.a. Distortion Function

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

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

Algorithm:

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

Repeat {

cluster assignment

for i = 1 : m

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

minimize $J(\dots)$ w.r.t. $c^{(1)}, c^{(2)} \dots c^{(m)}$; $\mu_1 \dots \mu_K$ fixed

move centroid

for k = 1 : K

μ_k := mean of points assigned to cluster k

minimize $J(\dots)$ w.r.t. $\mu_1 \dots \mu_K$

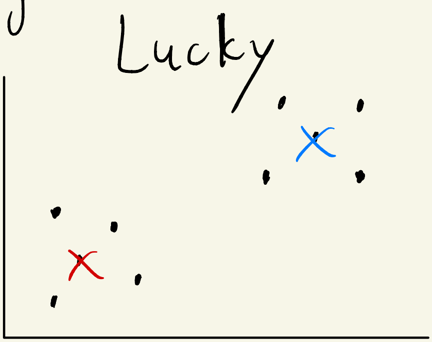
}

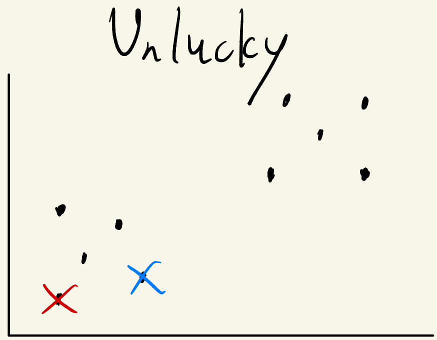
Random Initialization:

$K < m$

Randomly pick K training examples, assign $\mu_1, \mu_2 \dots \mu_K$ to these examples.

e.g. $K=2$





- Depending on the random initialization, K-means can result in different solutions.
  - → Different local minima
- To increase chances of converging to a good local minima, run K-means multiple times.

i.e. For $i = 1{:}100$ {

Randomly initialize K-means.

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

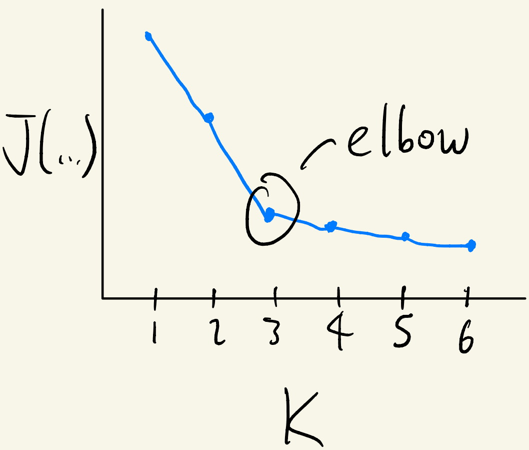
Compute cost function $J(c^{(1)}, \dots, c^{(m)}, \mu_1, \dots, \mu_K)$.

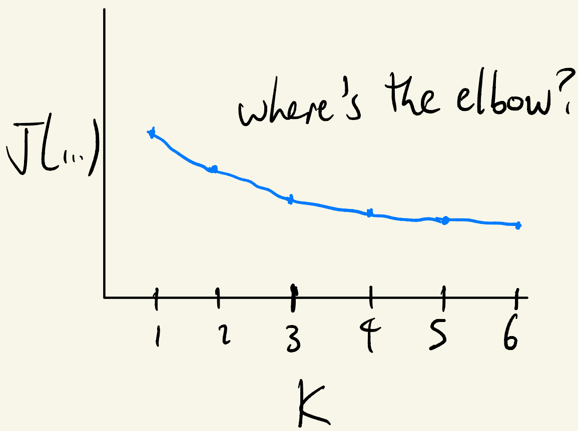
}

Pick clustering that gives lowest $J(\dots)$.

Choosing K (# of Clusters):

Elbow method: *Not super reliable





Choose based on how well it performs for a later purpose;

e.g., Clothing sizes

↳ If only S, M, L are available, choose $K=3$

↳ If more sizes are available, increase K