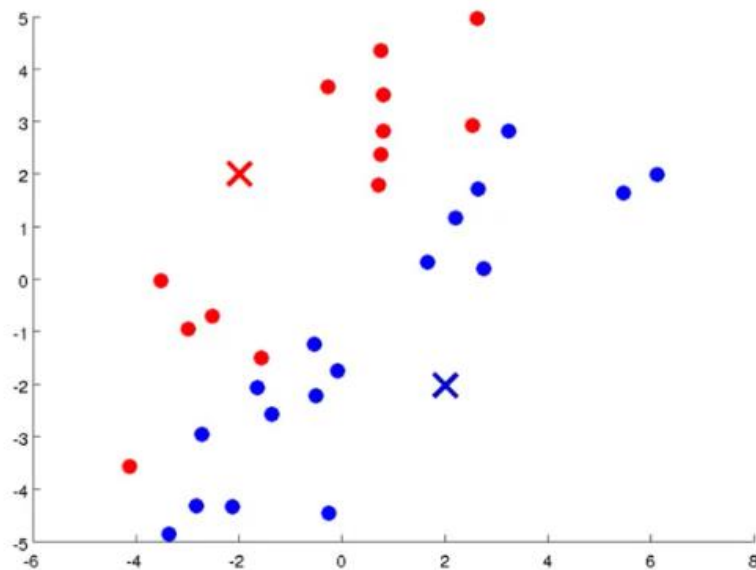
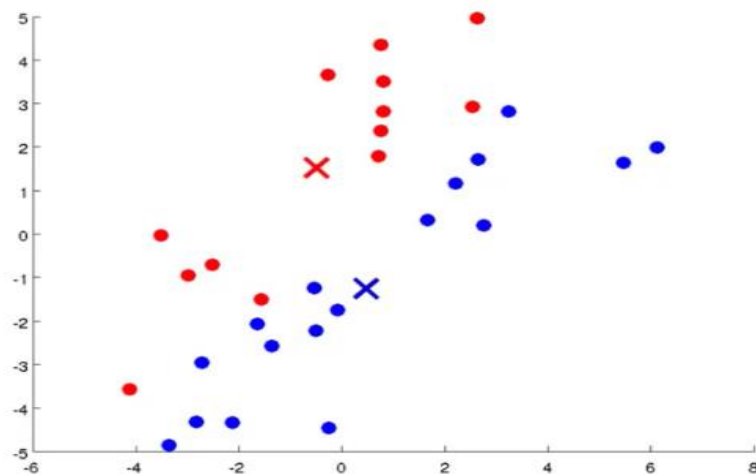


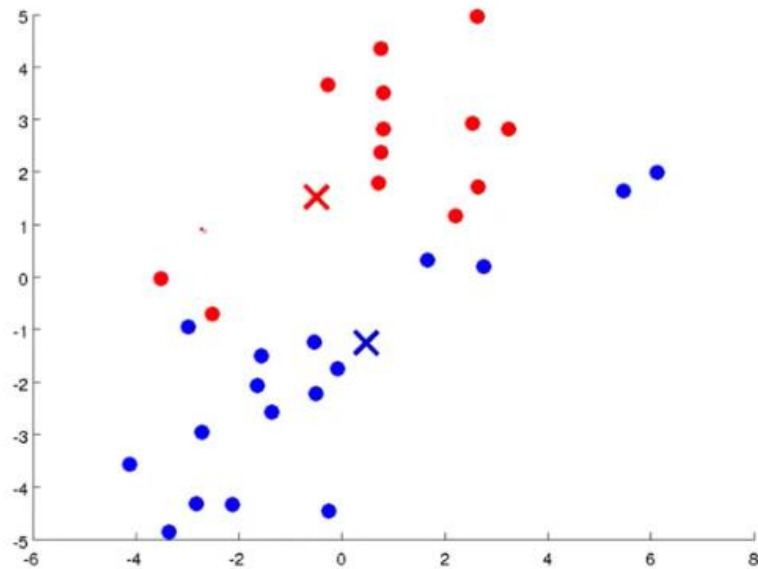
2 Now, the points near the blue ~~dots~~ ^{cross} are organised into a blue cross group & near the red cross are organised into a red cross ~~group~~ group.



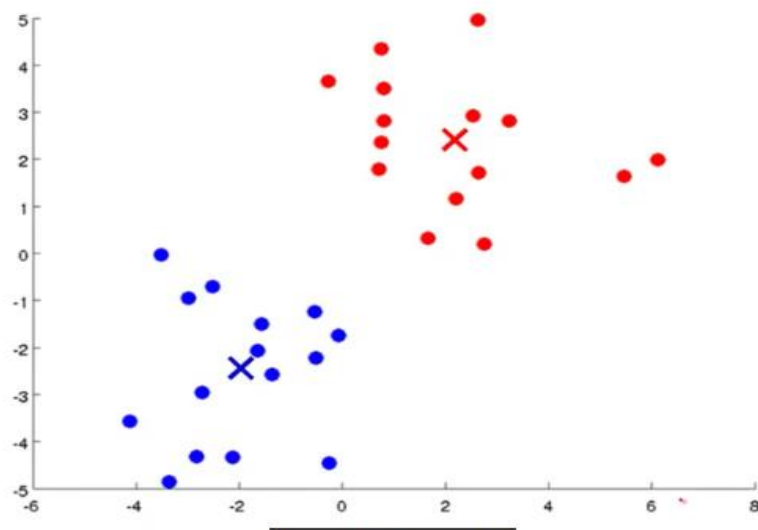
3 Then we take the mean of the points present in blue group & the mean is the new position for the blue cluster centroid. We do the same for red cluster centroid.



4 Then we again find points near the blue cross & label them as blue group & points near the red cross as red group.



We repeat the above-mentioned process again and again until we arrive at the right classification:



If you keep running additional iterations of K means from here the cluster centroids will not change any further and the colours of the points will not change any further. And so, this is the, at this point, K means has converged and it's done a pretty good job finding .

K-means algorithm explained mathematically:

Note:

1. $k=1$ to K represents number of cluster centroids.
2. $i=1$ to m represents the number of datasets/rows of data

Steps in which k means algorithm works:

1. We first randomly initialise K cluster centroids at positions $\mu_1, \mu_2, \dots, \mu_k$

Note that $\mu_1, \mu_2, \dots, \mu_n$ are n row vectors:



$$\underline{\mu}_1, \underline{\mu}_2, \dots, \underline{\mu}_K \in \mathbb{R}^n$$

2. Let us consider we have only 2 columns/features in X i.e. x_1 and x_2 . Now for each row of X , we plot x_1 vs x_2 and find the distance of these points from μ_1 and μ_2 and the points belong to the cluster centroid from which they have the minimum linear distance.

3. Say X^1 has the least linear distance from μ_2 and X^2 from μ_1 and X^3 and X^4 from μ_1 and μ_2 respectively and so on.

4. Then we find the mean of the points belonging to μ_1 and μ_2 respectively. Since X^1, X^2, \dots, X^n are 2 row vectors (since they have 2 columns each representing 2 different features), their mean is also a 2 row vector. The mean of the points belonging to the two groups μ_1 and μ_2 indicates the new coordinates of μ_1 and μ_2 .

5. We then again find the distance of the points on the graph from μ_1 and μ_2 and the points belong to the cluster centroid from which they have the minimum linear distance.

6. Then we repeat steps 3 and 4.

The process is illustrated below:

K-means algorithm

$$\mu_1 \quad \mu_2$$

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

Repeat {

Cluster assignment step

for $i = 1$ to m

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

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

for $k = 1$ to K

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

}

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

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

Instead of taking $\min_k \|x^{(i)} - \mu_k\|$, we can also take $\min_k \|x^{(i)} - \mu_k\|^2$ i.e. square of the linear distance between cluster centroids and the points to find which points belong to which cluster centroid.

Question:

K-means

Randomly initialize

Repeat

Cluster assignment step

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

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

☒ The third example $x^{(3)}$ has been assigned to cluster 5.

Correct

☒ The first and second training examples $x^{(1)}$ and $x^{(2)}$ have been assigned to the same cluster.

Correct

☐ The second and third training examples have been assigned to the same cluster.

Un-selected is correct

☒ Out of all the possible values of $k \in \{1, 2, \dots, K\}$ the value $k = 3$ minimizes $\|x^{(2)} - \mu_k\|^2$.

Correct

Continue

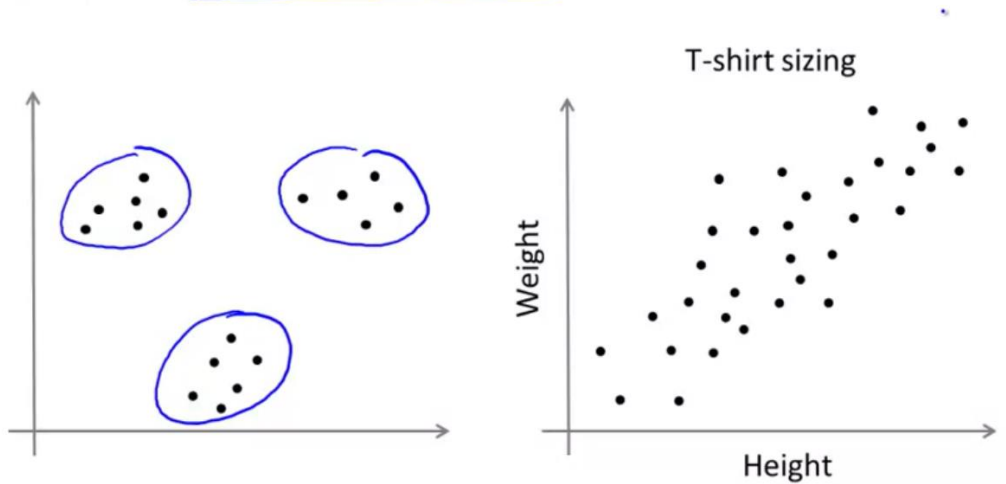
people actually tend to write this as the squared distance.

Andrew Ng

K-means for non-separated clusters:

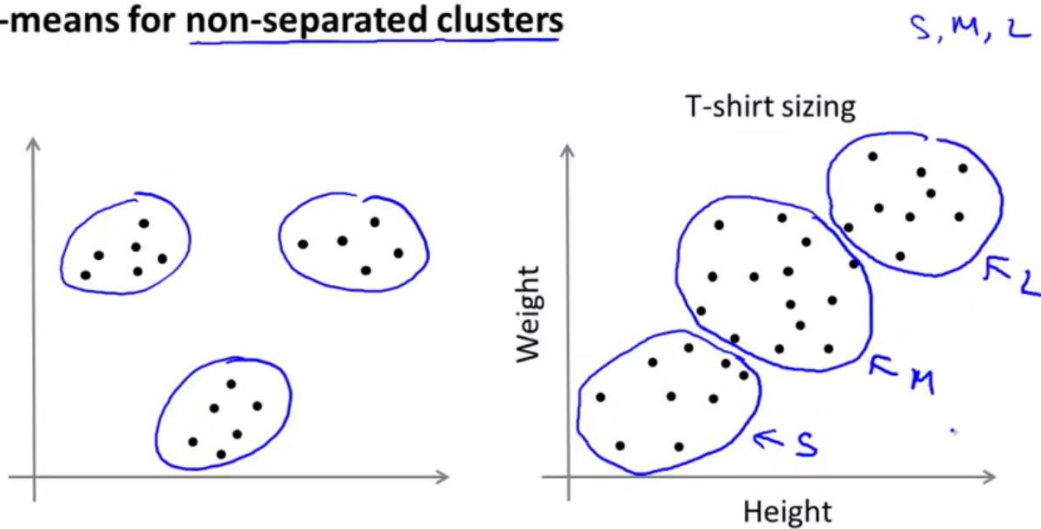
In the graph shown on the right side of the figure , we see data that doesn't seem to be easily separable into clusters:

K-means for non-separated clusters



The K-means algorithm can separate this data into following cluster:

K-means for non-separated clusters



The clusters represent small, medium and large t-shirt sizes.

Optimisation Objective:

Refer to the image below:

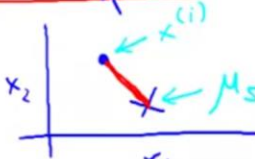
K-means optimization objective

- $c^{(i)}$ = index of cluster $(1, 2, \dots, K)$ to which example $x^{(i)}$ is currently assigned
 - μ_k = cluster centroid k ($\mu_k \in \mathbb{R}^n$)
 - $\mu_{c^{(i)}}$ = cluster centroid of cluster to which example $x^{(i)}$ has been assigned
- Handwritten notes:* K $k \in \{1, 2, \dots, K\}$
 $x^{(i)} \rightarrow 5$ $c^{(i)} = 5$ $\mu_{c^{(i)}} = \mu_5$

Optimization objective:

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

Handwritten notes: $\min_{c^{(1)}, \dots, c^{(m)}, \mu_1, \dots, \mu_K} J(c^{(1)}, \dots, c^{(m)}, \mu_1, \dots, \mu_K)$



Andrew Ng

We want to minimize the cost function J by obtaining values of $\mu_1, \mu_2, \dots, \mu_n$ and c^1, c^2, \dots, c^n such that when we find the sum of the square of distance between the mean of the cluster centroid and a point say x^i belonging to that cluster for all x^1 to x^m , the sum obtained is the minimum possible one.

The above lines mean that we want to classify the points in the best possible way.

Showing how the cost function is calculated:

1. In the first step we randomly initialise μ values and find cost function for all i ranging from 1 to m .
2. In the second step we find mean of points assigned to cluster k and repeat step 1.

K-means algorithm

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

Repeat {

Cluster assignment step
Minimize $J(\dots)$ w.r.t $c^{(1)}, c^{(2)}, \dots, c^{(m)}$ ←
(holding μ_1, \dots, μ_K fixed)

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

} *minimize $J(\dots)$ w.r.t μ_1, \dots, μ_K*

Andrew Ng

Random initialisation:

How do we initialise the $\mu_1, \mu_2, \dots, \mu_n$?

Random initialization

Should have $K < m$

Randomly pick K training examples.

Set μ_1, \dots, μ_K equal to these K examples.



Now, it can be that we end up on different local optima values depending on how we initialise the μ values.

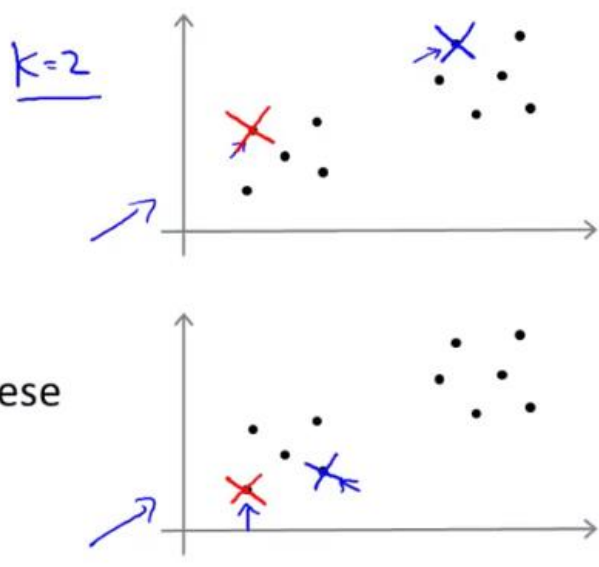
Random initialization

Should have $K < m$

Randomly pick K training examples.

Set μ_1, \dots, μ_K equal to these K examples.

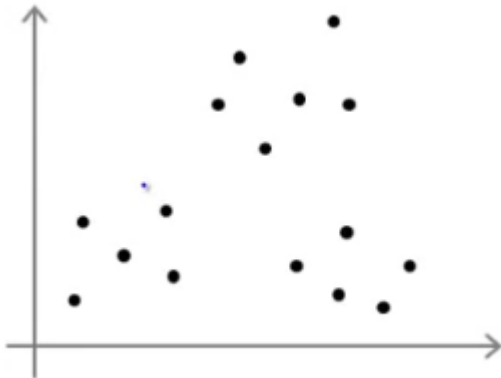
$$\begin{aligned}\mu_1 &= x^{(i)} \\ \mu_2 &= x^{(j)} \\ &\vdots\end{aligned}$$



Above 2 graphs show 2 different initialisations of μ_1 and μ_2 .

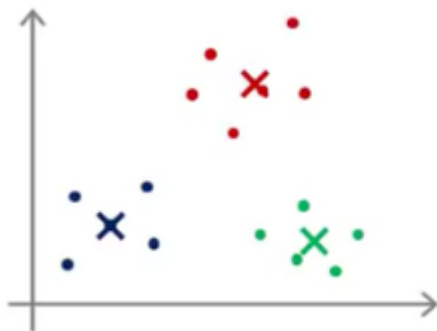
The case of different local optima:

Consider the graph below.

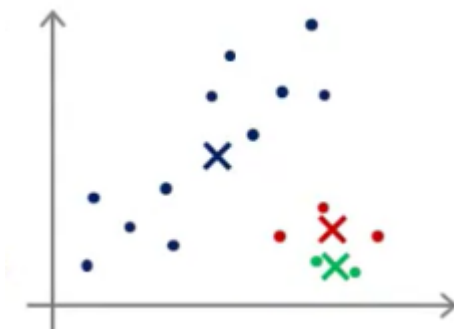


Now depending on how we initialise μ values, we may end up on different local optima and hence different clusters:

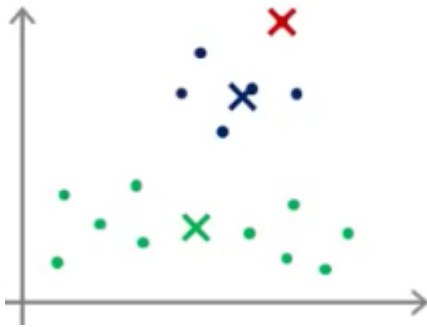
1. Initialisation 1:



2. Initialisation 2:



3. Initialisation 3:



Solving the above problem

We see above that different initialisations can lead to different clusters. So how do we solve this problem?

There are usually two cases that occur:

a) If $K=2-10$

Run the whole process of k-means algorithm anywhere between 50-1000 times depending on requirement. Then pick the value of clustering for which the cost is minimum. This prevents the algorithm from staying on the wrong local optima.

Random initialization

```
For i = 1 to 100 { 50 - 1000
    → Randomly initialize K-means.
    Run K-means. Get  $c^{(1)}, \dots, c^{(m)}, \mu_1, \dots, \mu_K$ .
    Compute cost function (distortion)
    →  $J(c^{(1)}, \dots, c^{(m)}, \mu_1, \dots, \mu_K)$ 
}
```

Pick clustering that gave lowest cost $J(c^{(1)}, \dots, c^{(m)}, \mu_1, \dots, \mu_K)$
 $K=2-10$

b) If $K > 10$

Usually doing the whole process pertaining to the algorithm once with single time initialisation of mu values produces the correct result. Although doing the whole process pertaining to the algorithm multiple times can lead to a little better result.

Questions:

Q1:

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

- ☐ Pick a random integer i from $\{1, \dots, k\}$. Set $\mu_1 = \mu_2 = \dots = \mu_k = x^{(i)}$.
- ☐ Pick k distinct random integers i_1, \dots, i_k from $\{1, \dots, k\}$.
Set $\mu_1 = x^{(i_1)}, \mu_2 = x^{(i_2)}, \dots, \mu_k = x^{(i_k)}$.
- ☒ Pick k distinct random integers i_1, \dots, i_k from $\{1, \dots, m\}$.
Set $\mu_1 = x^{(i_1)}, \mu_2 = x^{(i_2)}, \dots, \mu_k = x^{(i_k)}$.
- ☐ Set every element of $\mu_i \in \mathbb{R}^n$ to a random value between $-\epsilon$ and ϵ , for some small ϵ .

Correct

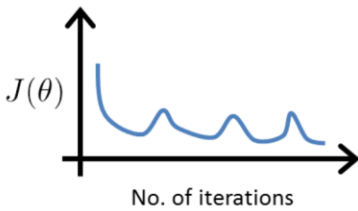
Continue

Sending request to www.facebook.com...

Andrew Ng

Q2:

Suppose you have implemented k-means and to check that it is running correctly, you plot the cost function $J(c^{(1)}, \dots, c^{(m)}, \mu_1, \dots, \mu_k)$ 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.

Correct

Continue

move centroid

cluster k

Andrew Ng

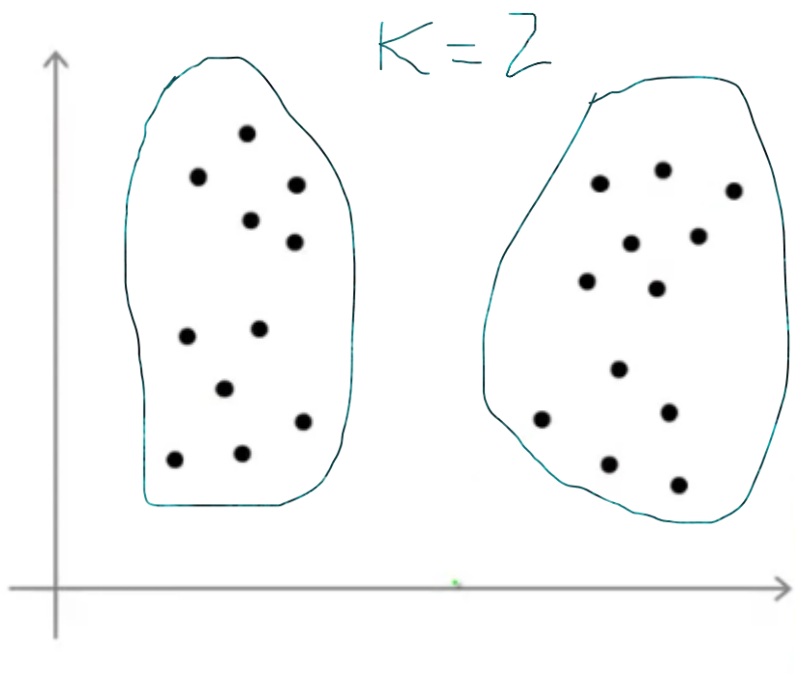
Choosing the value of K:

There is no particular way of choosing the value of K. Consider the figure below:

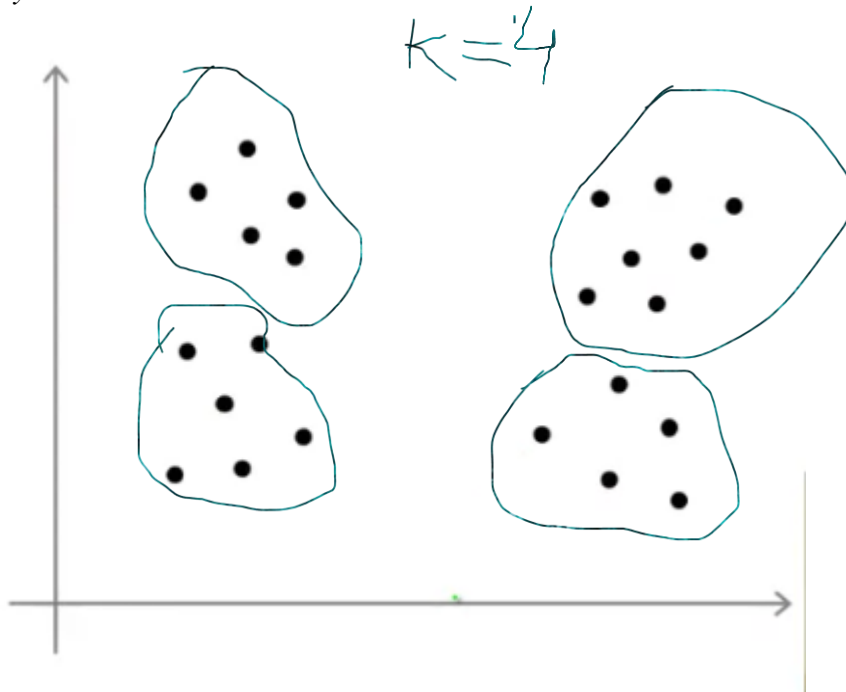


Now this can be made into clusters in many ways, two of which are shown below:

Way 1:



Way 2:

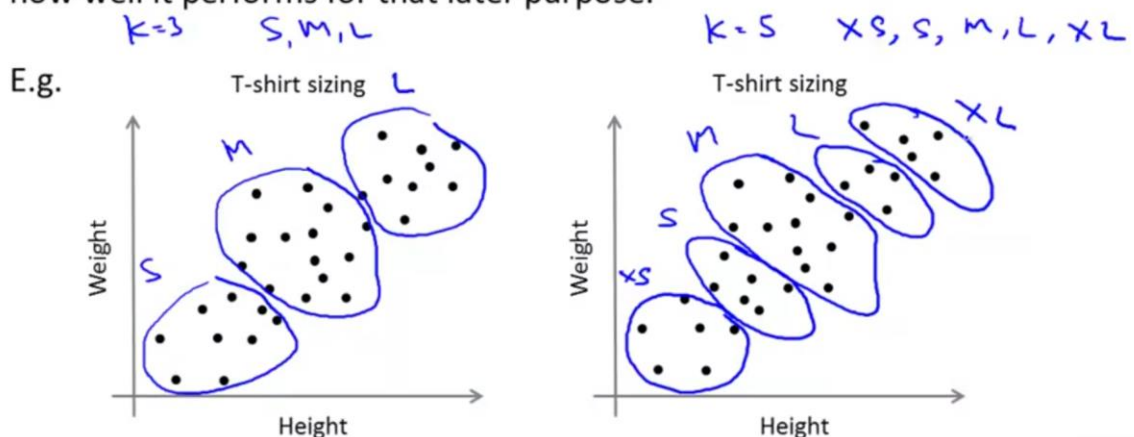


So, we see that there is no particular correct way of choosing a value of K and usually we choose a value by hand. Though there are two ways that may be used to choose a value of K :

1. Choosing K value according to need.

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.



Transferring data from b6sc.co...

Andrew Ng

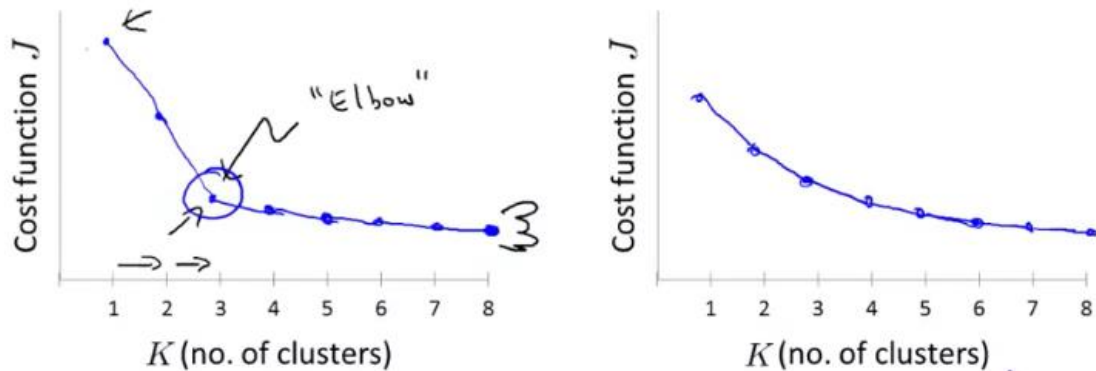
Now, if you are a T-shirt seller then you can choose a value of K according to your need. For example if you feel that using $K=3$ which classifies the t-shirts into 3 groups that is S, M or L

is more useful than using $K=5$ which classifies the t-shirts into 5 groups that is XS,S,M,L or XL then you can choose $K=3$.

2. Using the elbow method:

Choosing the value of K

Elbow method:



If we plot cost function vs K and get the graph on the left side of above figure then the elbow point can be taken as value of K .

However usually we get the graph shown on the right side of the figure and there is no clear elbow. Hence this method is not always useful.

Questions:

Choose the correct answer for the following question. Some questions may have multiple correct answers. E.g.

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 $k = 5$.

Correct

Continue

Transferring data from b.6sc.co... Andrew Ng

1. For which of the following tasks might K-means clustering be a suitable algorithm? Select all that apply.

1 point

- ☐ Given historical weather records, predict if tomorrow's weather will be sunny or rainy.
- ☒ From the user usage patterns on a website, figure out what different groups of users exist.
- ☐ Given many emails, you want to determine if they are Spam or Non-Spam emails.
- ☒ Given a set of news articles from many different news websites, find out what are the main topics covered.

2. Suppose we have three cluster centroids $\mu_1 = \frac{1}{2}$, $\mu_2 = \frac{-3}{0}$ and $\mu_3 = \frac{4}{2}$. Furthermore, we have a training example $x^{(i)} = \frac{3}{1}$. After a cluster assignment step, what will $c^{(i)}$ be?

- ☐ $c^{(i)} = 2$
- ☒ $c^{(i)} = 3$
- ☐ $c^{(i)} = 1$
- ☐ $c^{(i)}$ is not assigned

3. K-means is an iterative algorithm, and two of the following steps are repeatedly carried out in its inner-loop. Which two?

- ☐ Feature scaling, to ensure each feature is on a comparable scale to the others.
- ☐ Using the elbow method to choose K.
- ☒ The cluster assignment step, where the parameters $c^{(i)}$ are updated.
- ☒ Move the cluster centroids, where the centroids μ_k are updated.

4. Suppose you have an unlabeled dataset $\{x^{(1)}, \dots, x^{(m)}\}$. You run K-means with 50 different random

initializations, and obtain 50 different clusterings of the

data. What is the recommended way for choosing which one of

these 50 clusterings to use?

- ☐ Always pick the final (50th) clustering found, since by that time it is more likely to have converged to a good solution.
- ☐ The answer is ambiguous, and there is no good way of choosing.
- ☐ The only way to do so is if we also have labels $y^{(i)}$ for our data.
- ☒ For each of the clusterings, compute $\frac{1}{m} \sum_{i=1}^m \|x^{(i)} - \mu_{c(i)}\|^2$, and pick the one that minimizes this.

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

1 poin

- ☐ Since K-Means is an unsupervised learning algorithm, it cannot overfit the data, and thus it is always better to have as large a number of clusters as is computationally feasible.
- ☐ The standard way of initializing K-means is setting $\mu_1 = \dots = \mu_k$ to be equal to a vector of zeros.
- ☒ If we are worried about K-means getting stuck in bad local optima, one way to ameliorate (reduce) this problem is if we try using multiple random initializations.
- ☒ For some datasets, the "right" or "correct" value of K (the number of clusters) can be ambiguous, and hard even for a human expert looking carefully at the data to decide.

