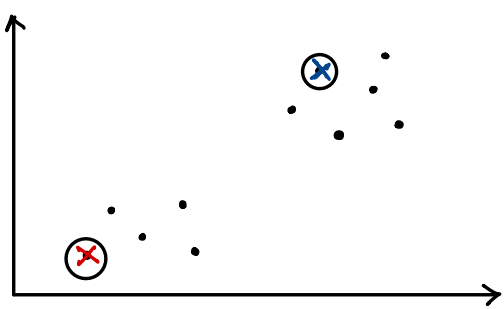
We've been saying that we randomly choose two points as cluster centroids, but there's a different way to do it.

First $K < m$ always because the no. of training examples must be higher than the centroids otherwise there won't be any cluster formation.

Now, out of all of the $m$ training examples pick $K$.

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

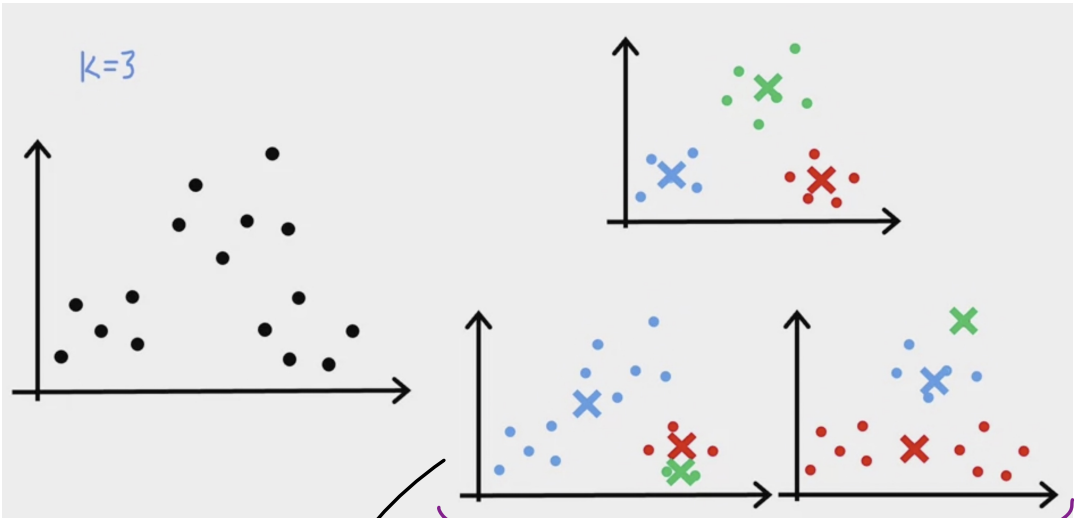
eg.



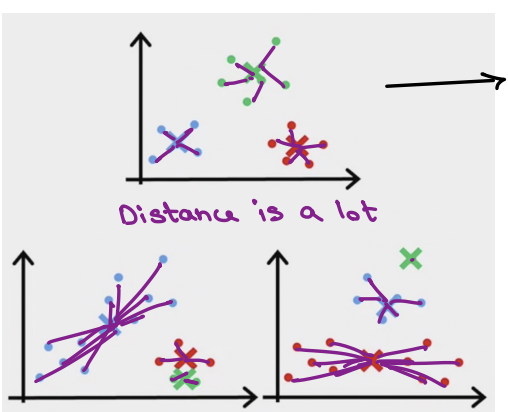
$K = 2$ in this case and these coordinates of these two points was chosen as the cluster centroid.

With this method depending on what is the $K$-means initialization we might get a different group of cluster.

eg. In the next page




these are three different outcomes to the graph on the left when cluster centroid was initialized in different ways.




This is the best scenario.

One way to check whether the cluster centroid is placed at the most appropriate location is to run K-means multiple times for different initializations and check which gives the minimum cost function value.

To incorporate multiple K-means this is the algorithm:-

for i = 1 to 100 {

Choose between 50 - 1000. More than 1000 is computationally expensive and gives diminishing returns

initialize K-means "randomly"

"randomly" refers to choosing k examples from m

Run K-means and get $c^{(1)}, \ldots, c^{(m)}$, $\mu_1, \ldots, \mu_K$ through convergence

Check the distortion cost function $J(c^{(1)}, \ldots, c^{(m)}, \mu_1, \ldots, \mu_K)$ }

Pick the set of clusters that gave the lowest J.