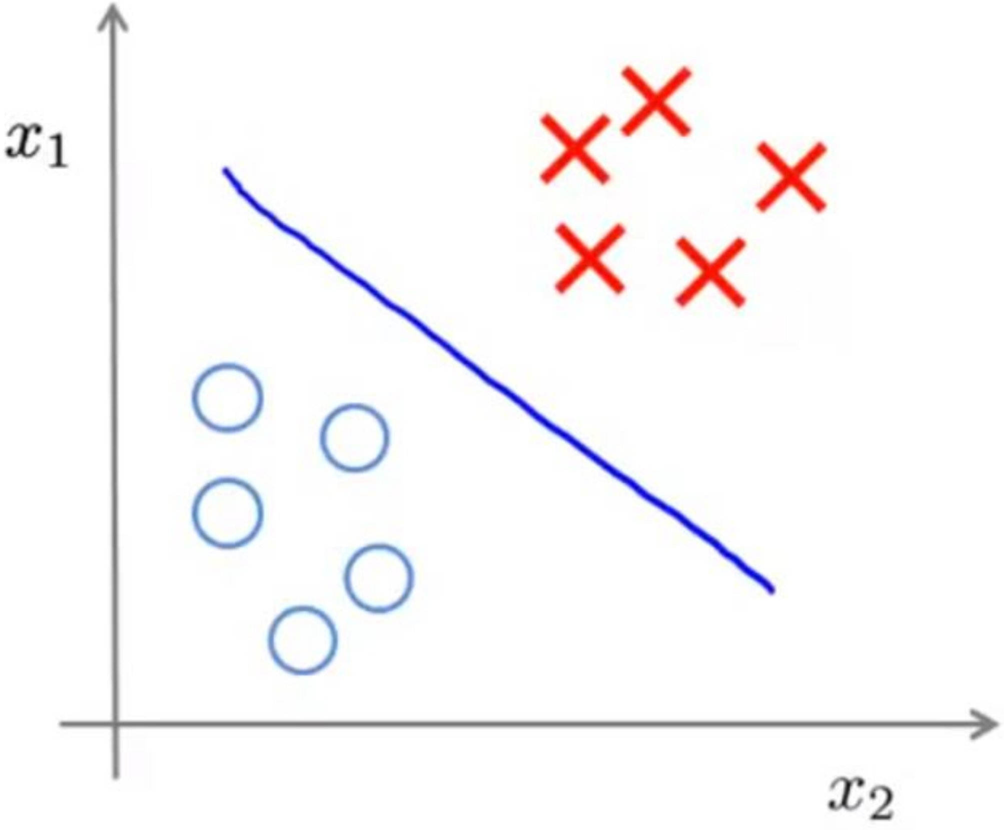
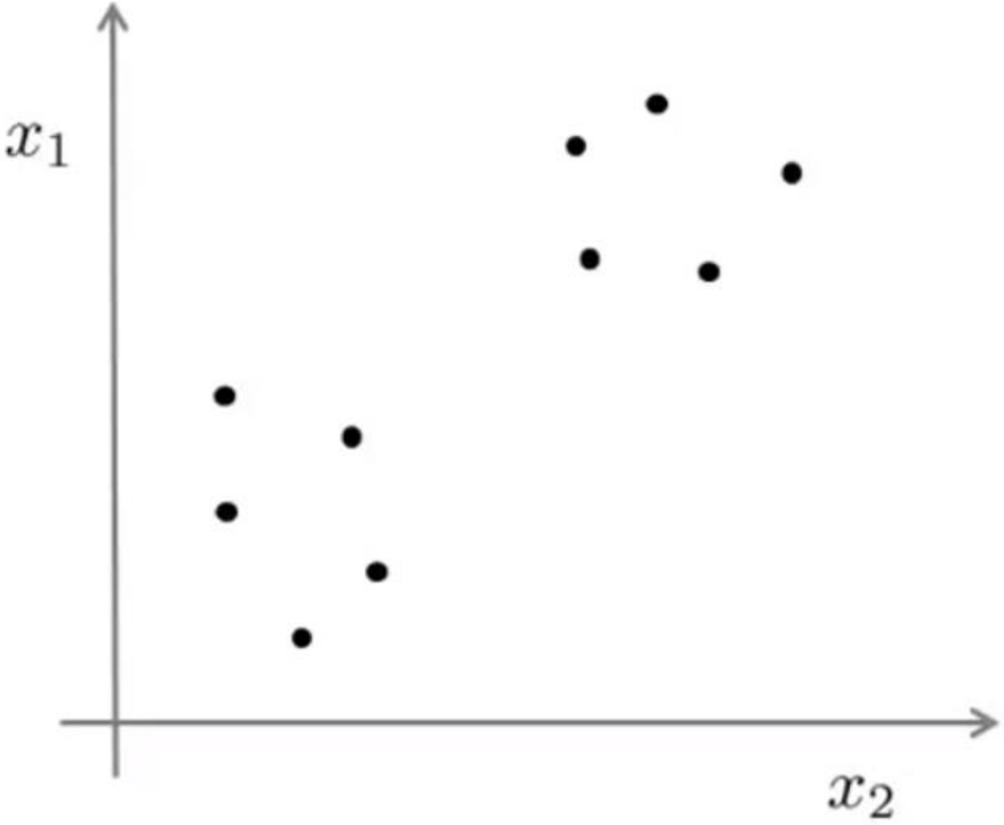
Unsupervised Learning: Clustering with K-Means

Supervised learning

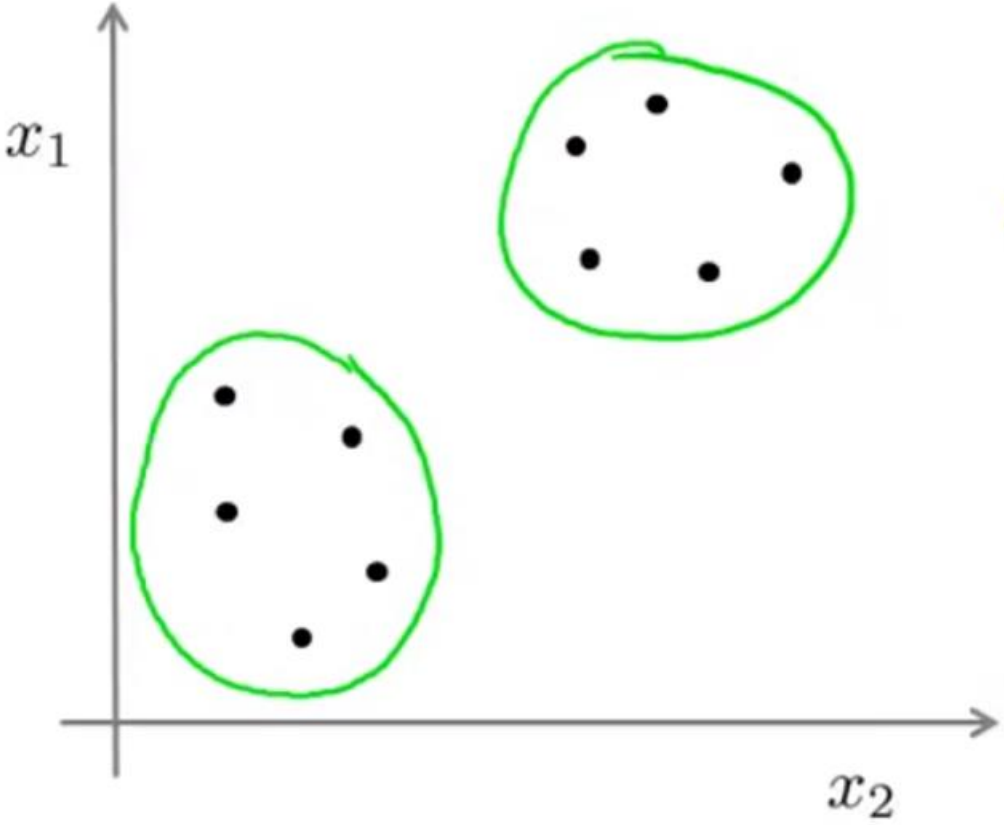


Unsupervised learning



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

Unsupervised learning



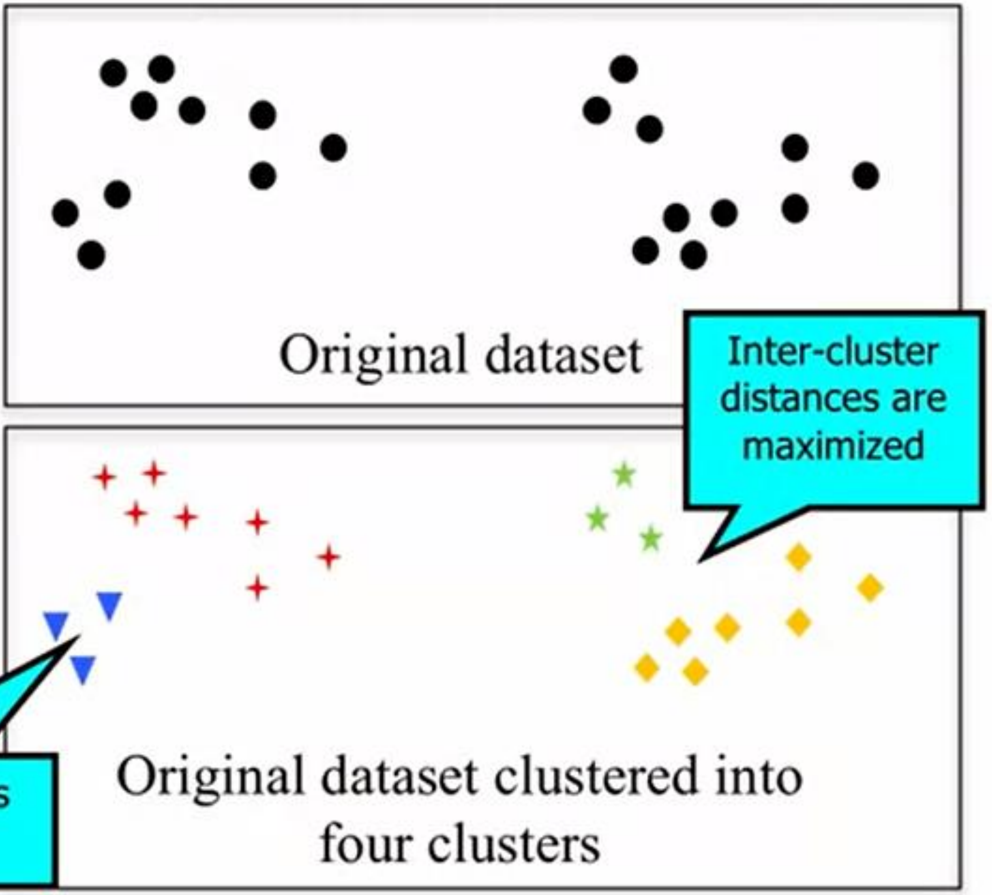
Clustering algorithm

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

Clustering:

Finding a way to divide a dataset into groups ('clusters')

- Data points within the same cluster should be 'close' or 'similar' in some way.
- Data points in different clusters should be 'far apart' or 'different'
- Clustering algorithms output a cluster membership index for each data point:
  - *Hard clustering: each data point belongs to exactly one cluster*
  - *Soft (or fuzzy) clustering: each data point is assigned a weight, score, or probability of membership for each cluster*



Original dataset

Original dataset clustered into four clusters

Inter-cluster distances are maximized
Intra-cluster distances are minimized

K-means Clustering

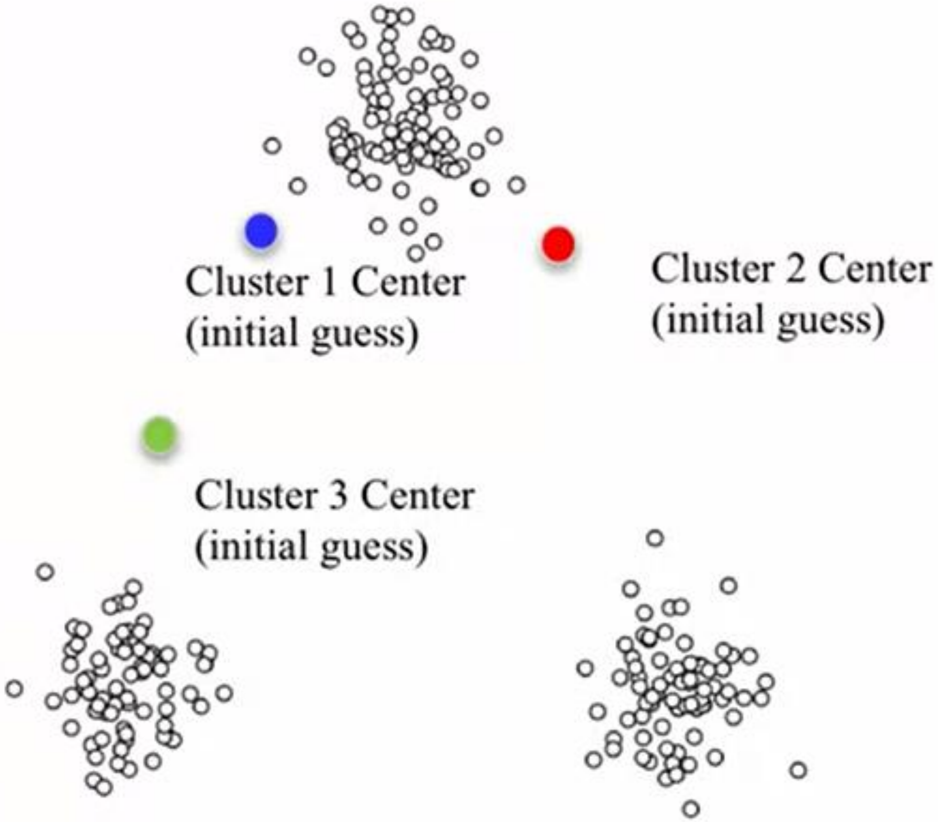
The k-means algorithm

Initialization Pick the number of clusters *k* you want to find. Then pick *k random* points to serve as an initial guess for the cluster centers.

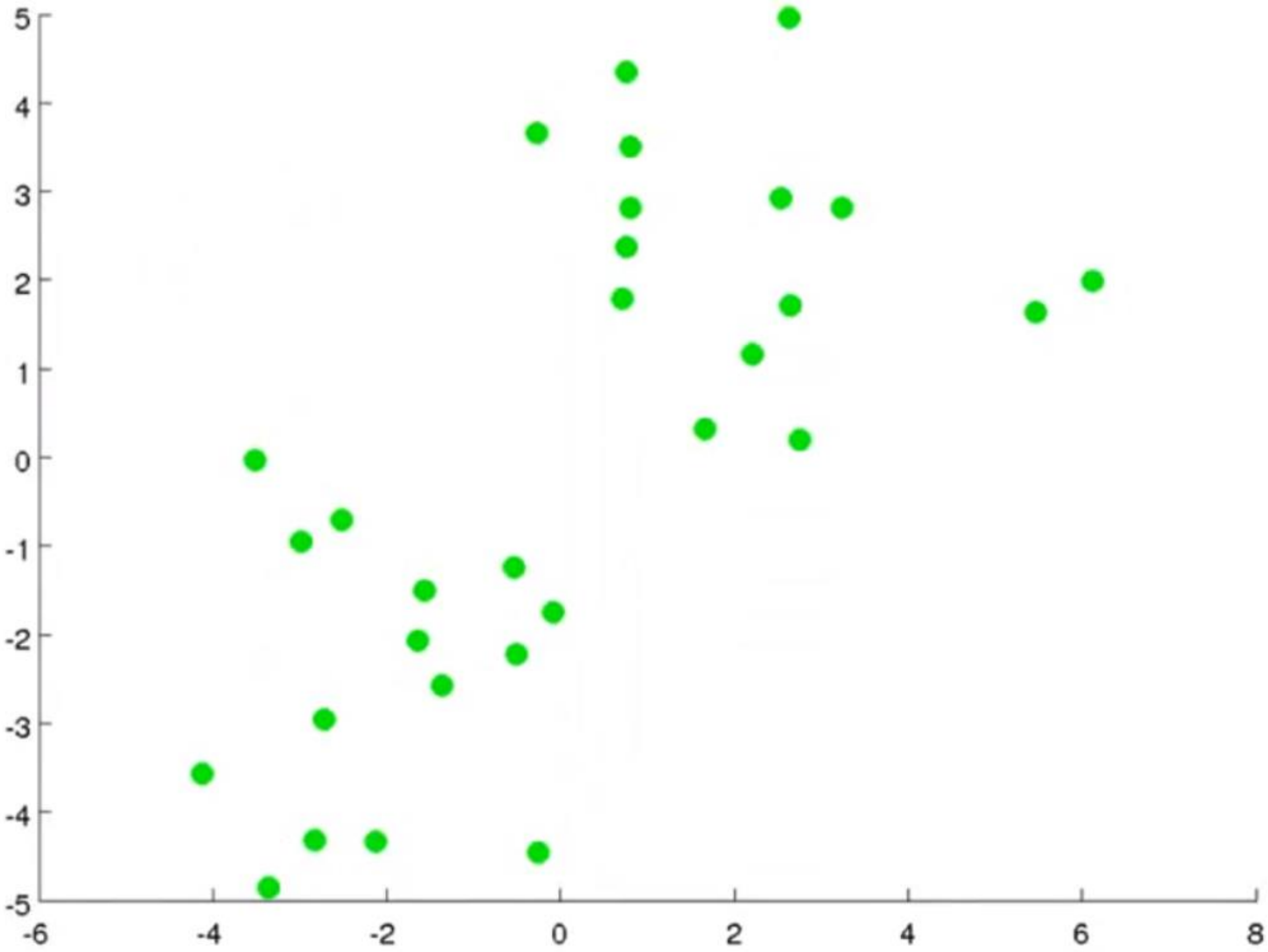
Step A Assign each data point to the nearest cluster center.

Step B Update each cluster center by replacing it with the mean of all points assigned to that cluster (in step A).

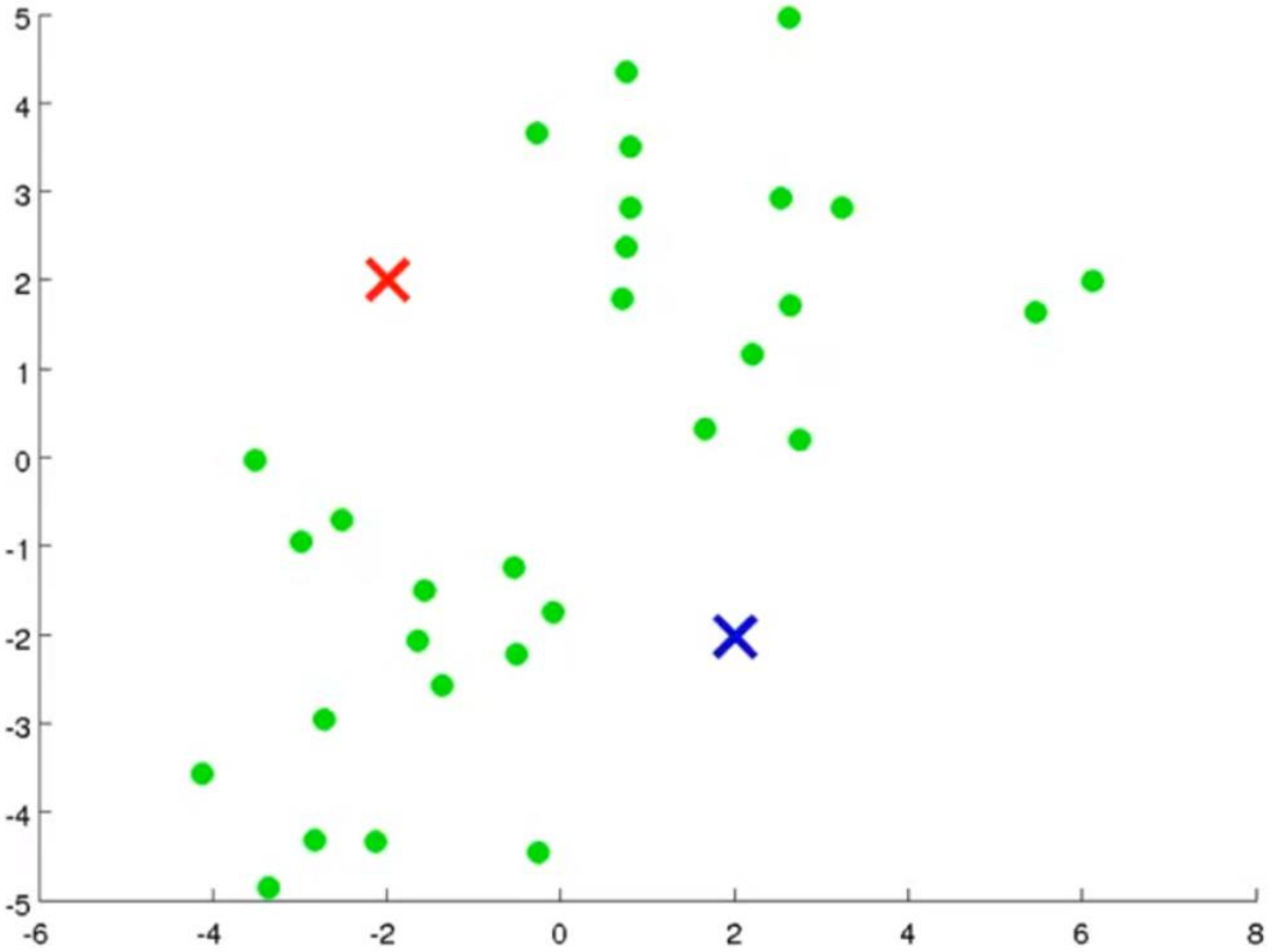
Repeat steps A and B until the centers converge to a stable solution.



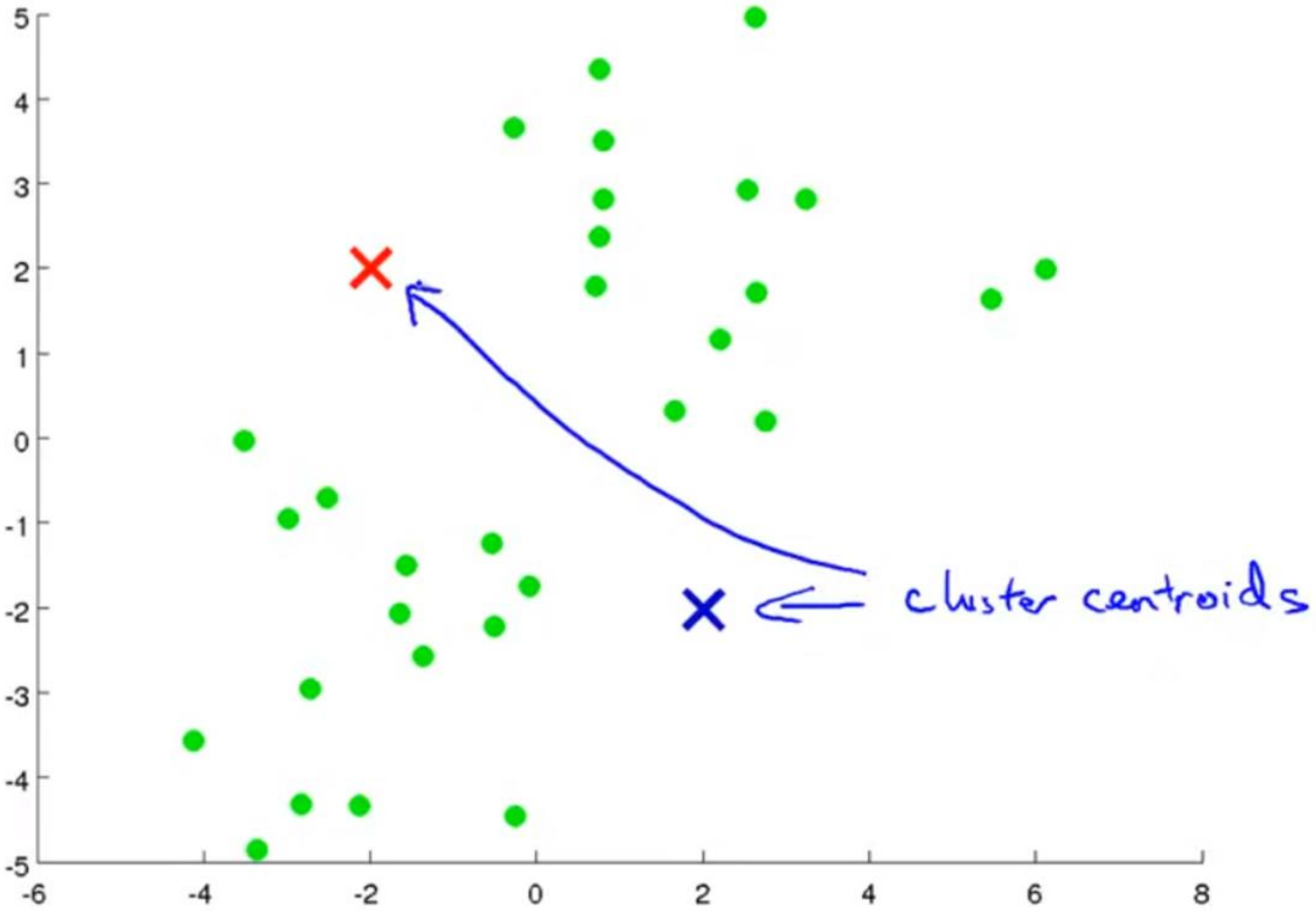
Demo: https://www.naftaliharris.com/blog/visualizing-k-means-clustering/

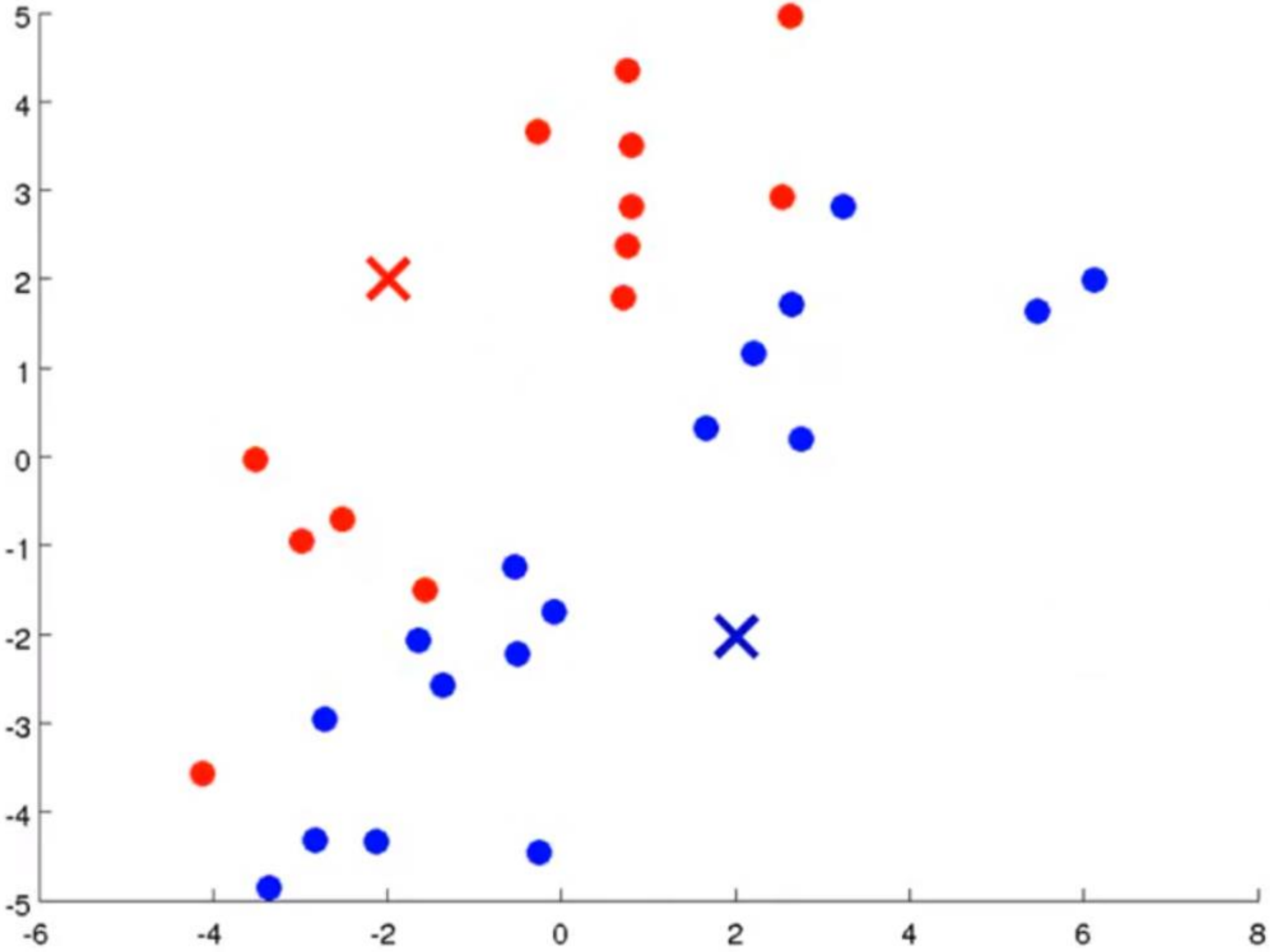


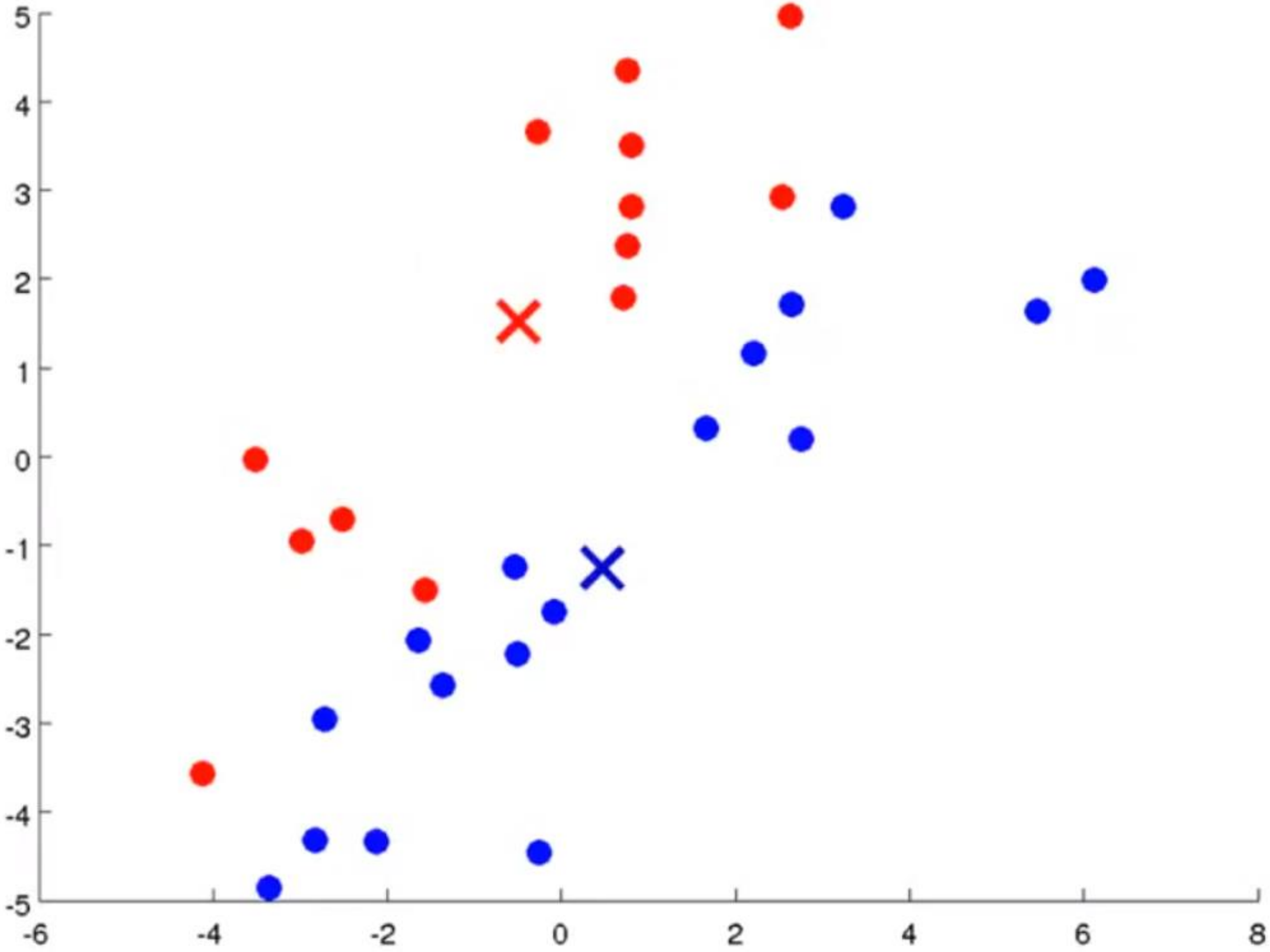
Andrew Ng

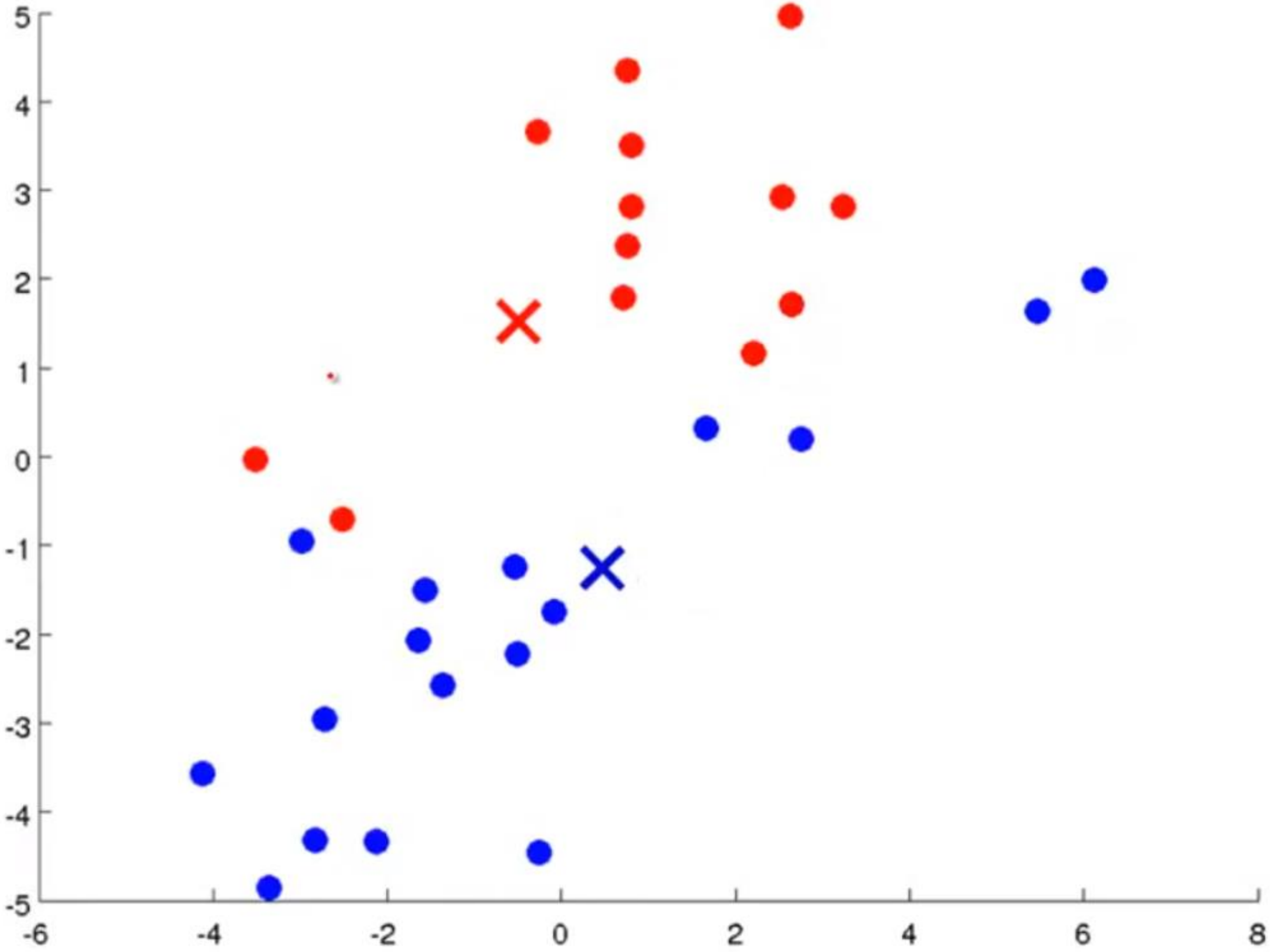


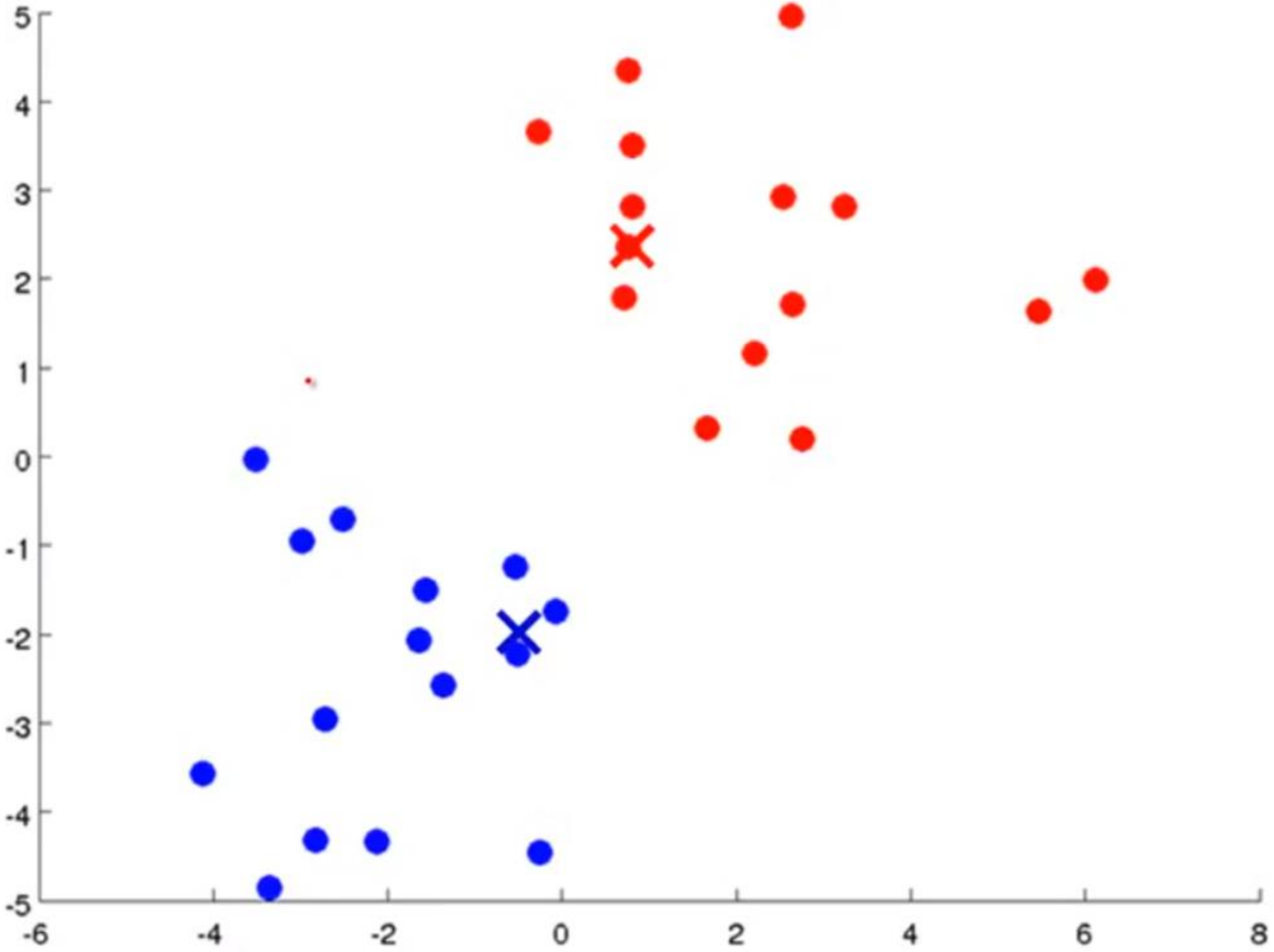
Andrew Ng

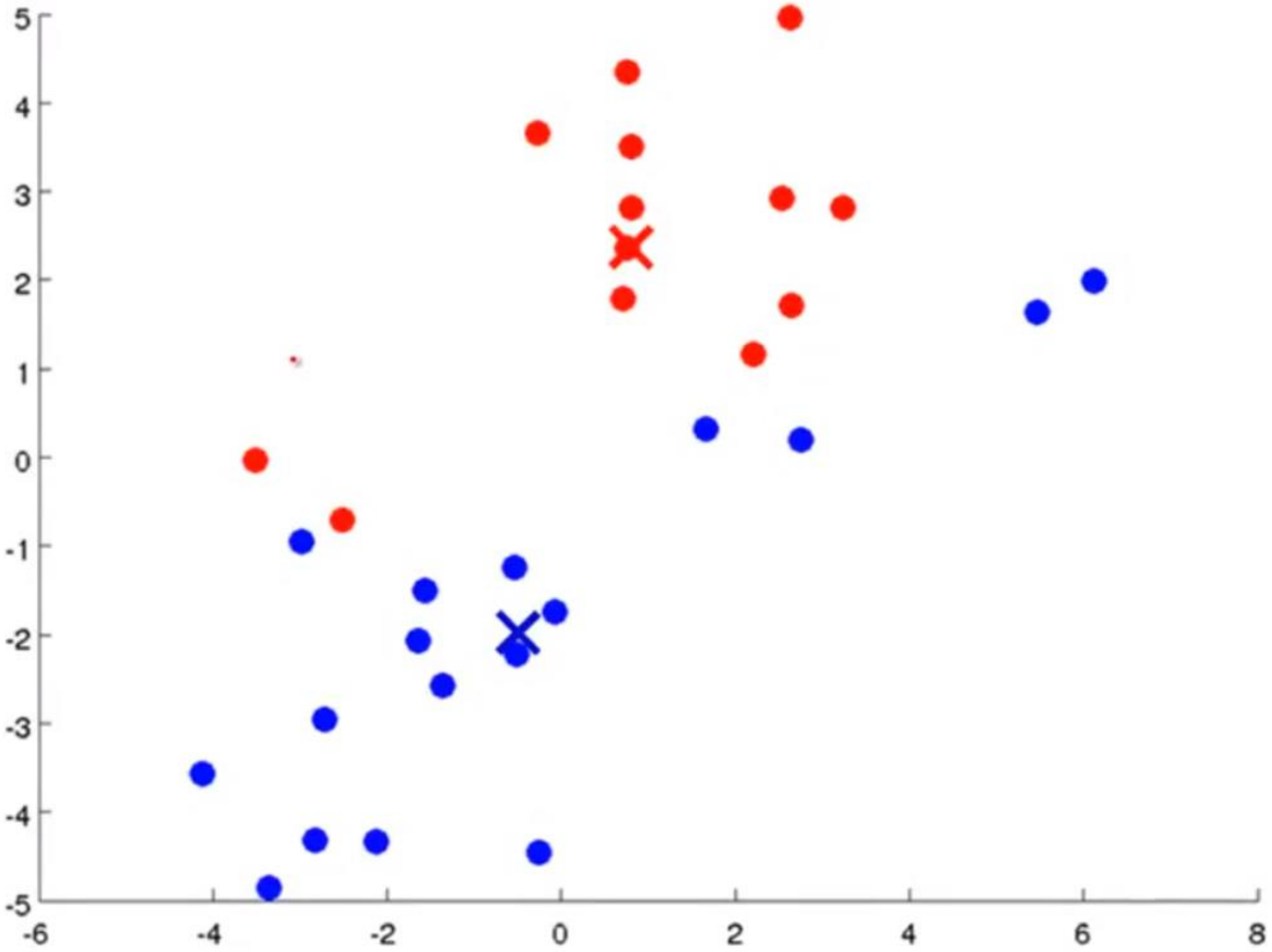

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8
cluster centroids

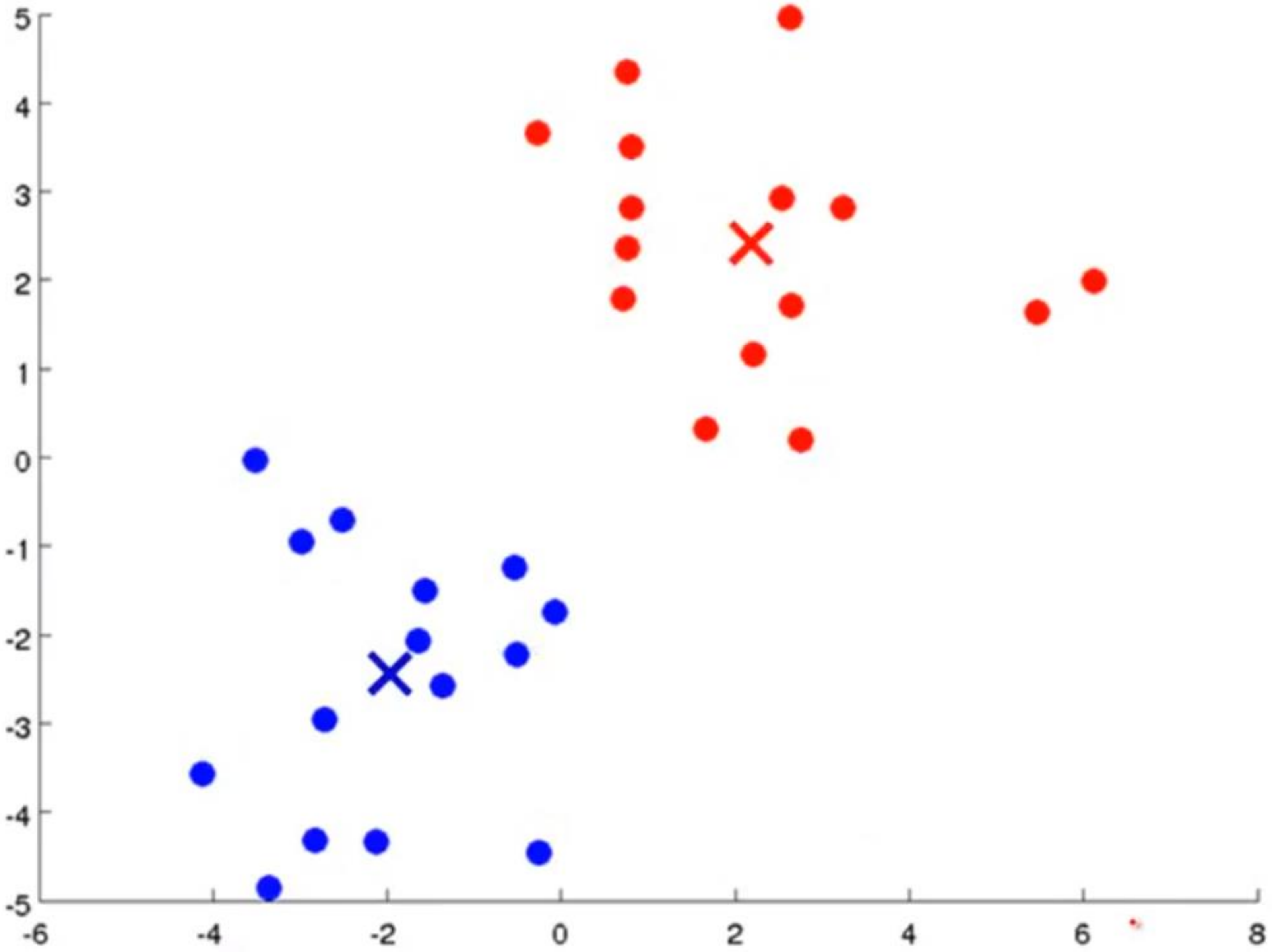

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8


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

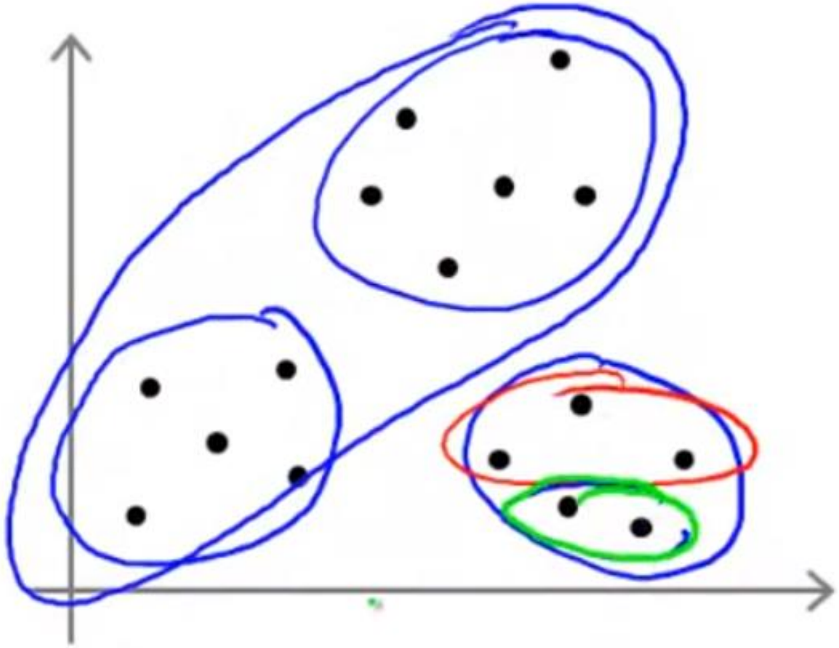

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8

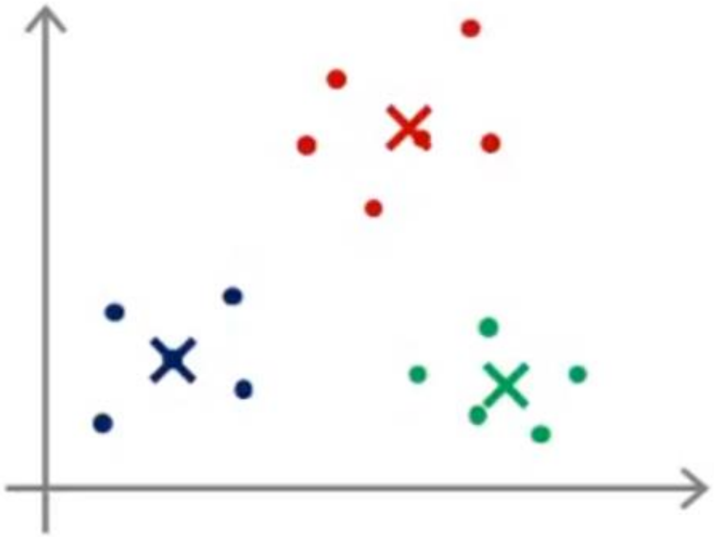

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8

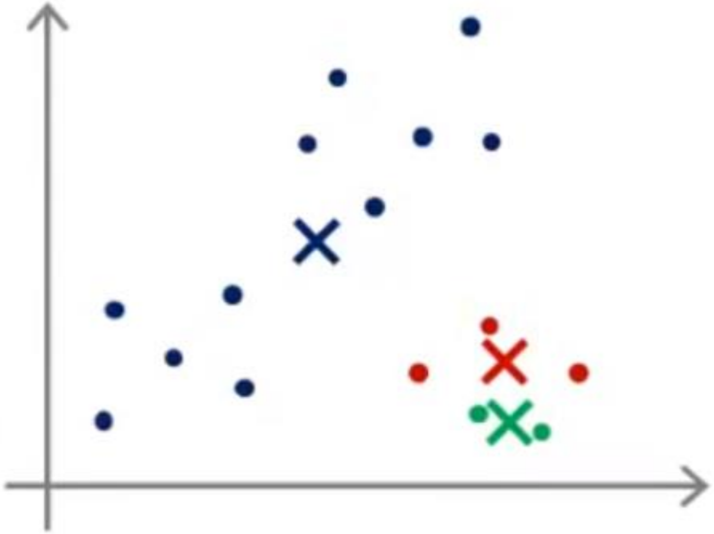

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8

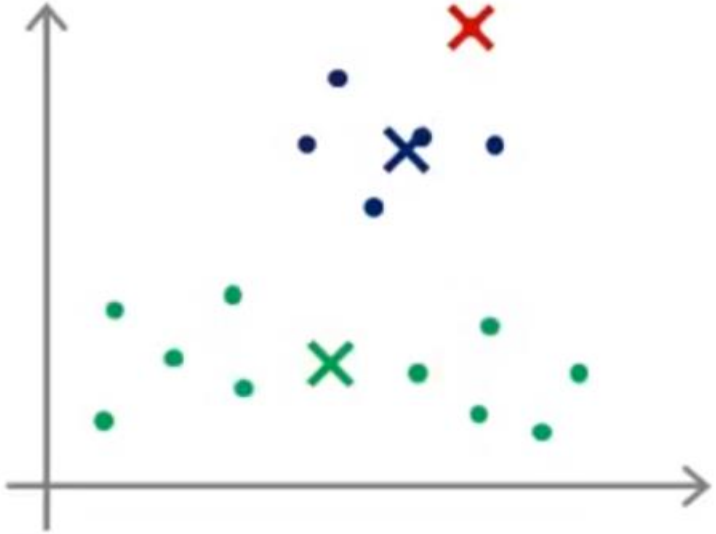

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8

Local optima





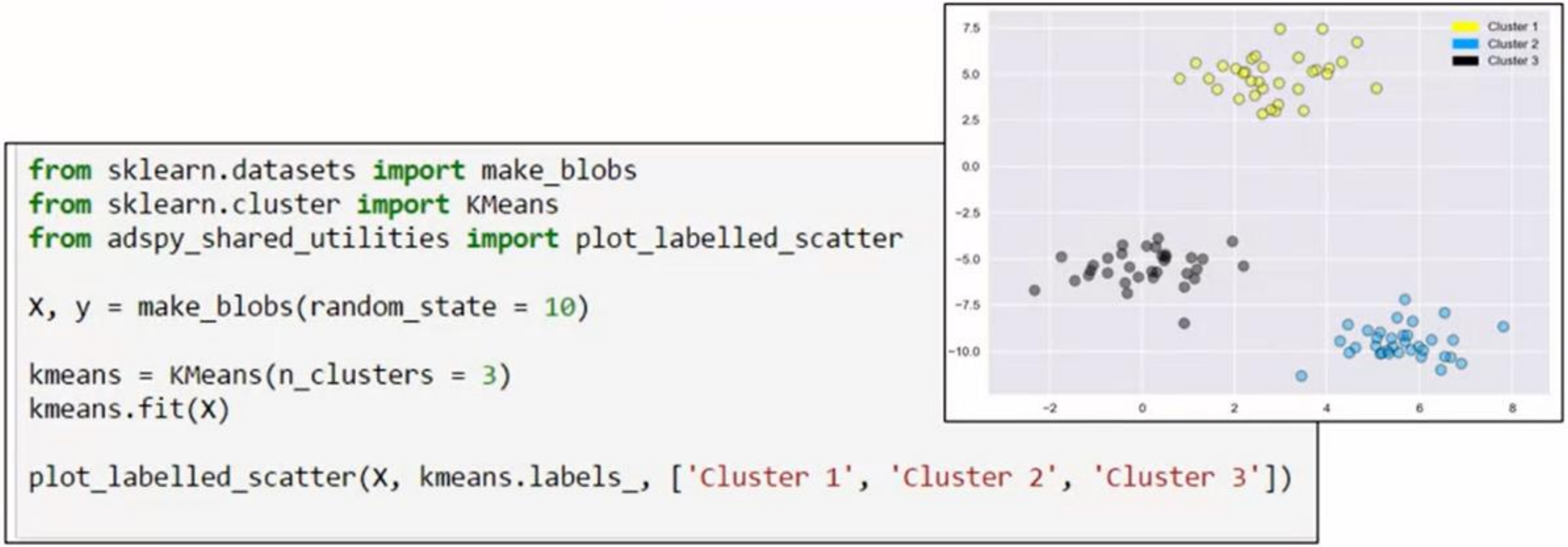




Random initialization

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

k-means Example in Scikit-Learn



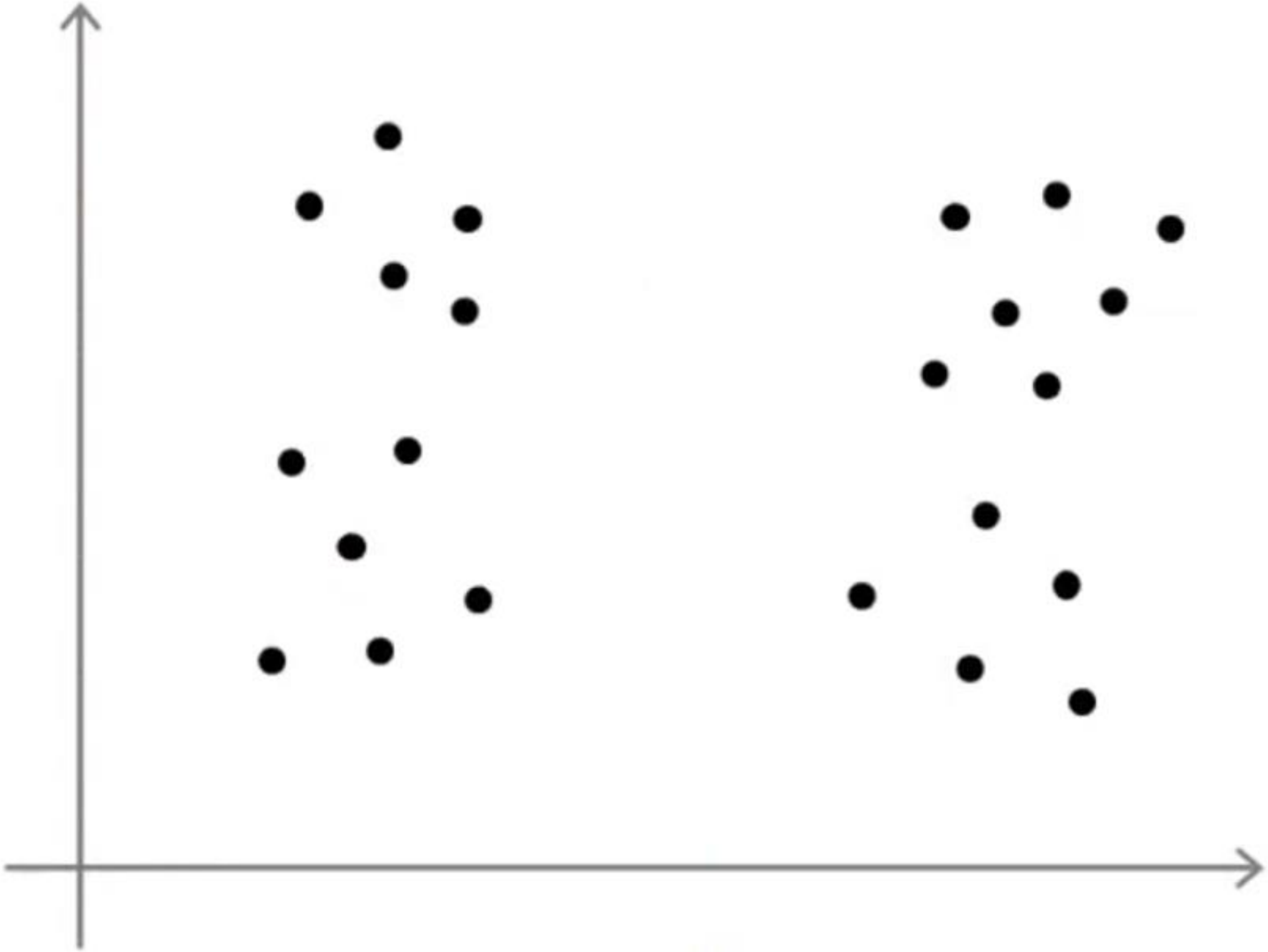
```
from sklearn.datasets import make_blobs
from sklearn.cluster import KMeans
from adspy_shared_utilities import plot_labelled_scatter

X, y = make_blobs(random_state = 10)

kmeans = KMeans(n_clusters = 3)
kmeans.fit(X)

plot_labelled_scatter(X, kmeans.labels_, ['Cluster 1', 'Cluster 2', 'Cluster 3'])
```

Choosing K

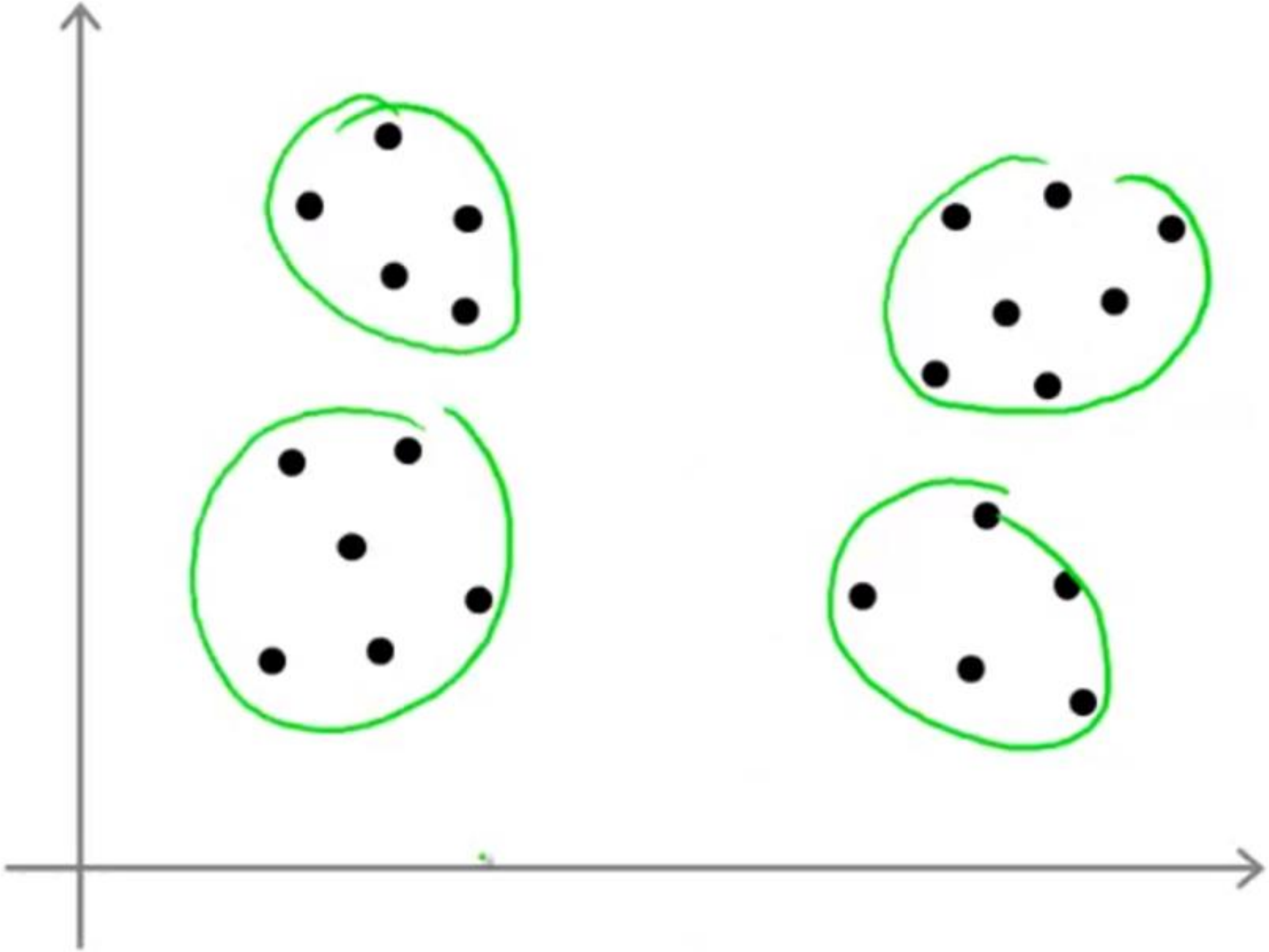
Clustering

Unsupervised Learning

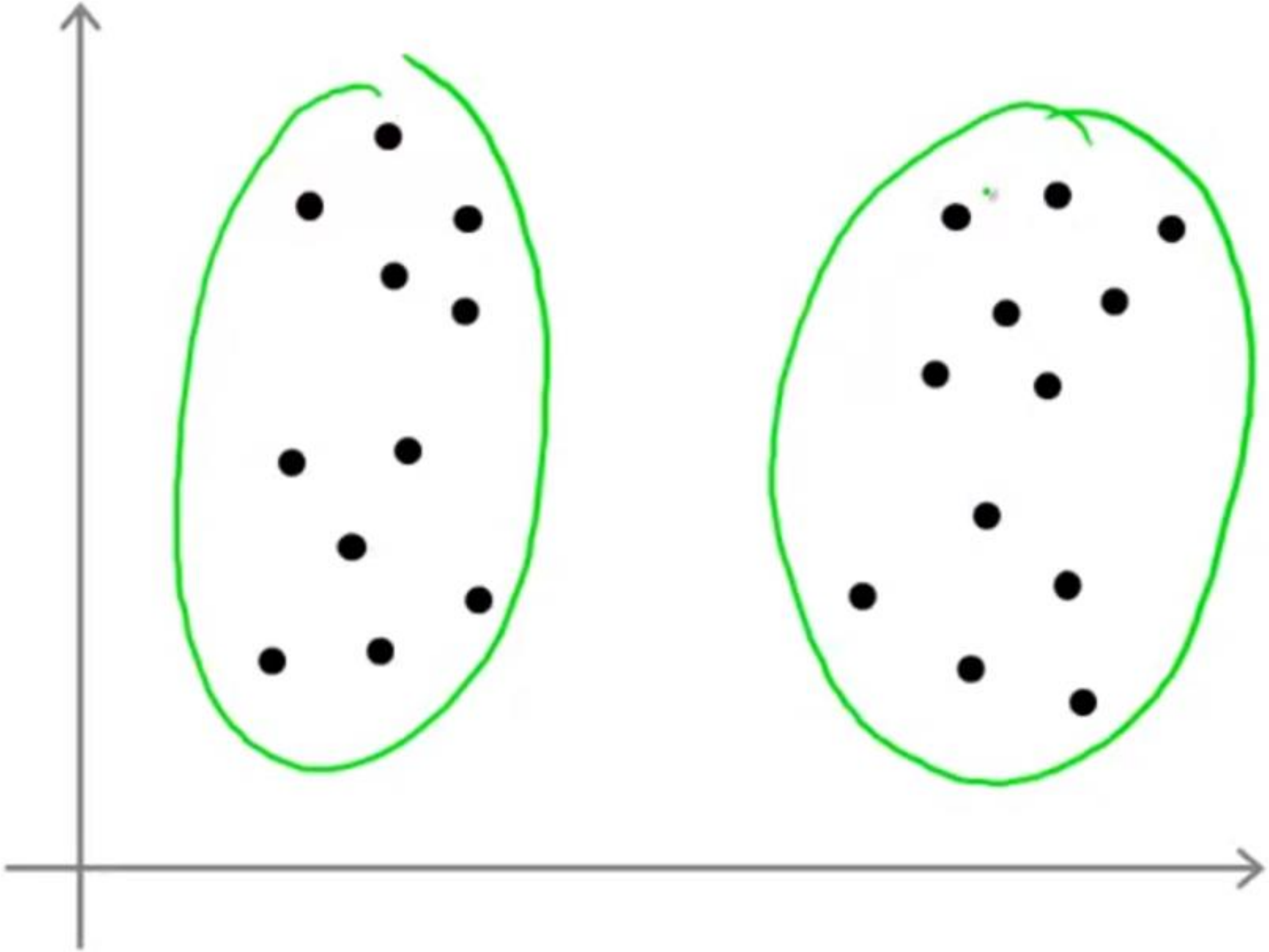
What is the right value of K?



What is the right value of K?

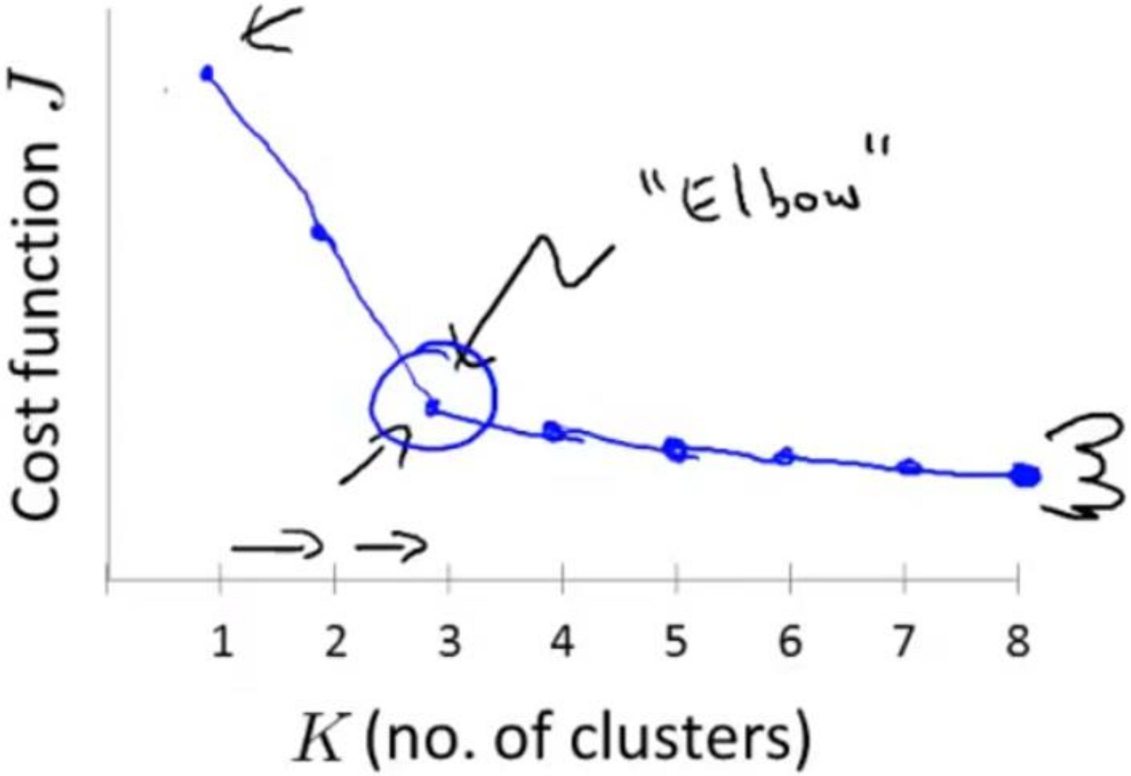


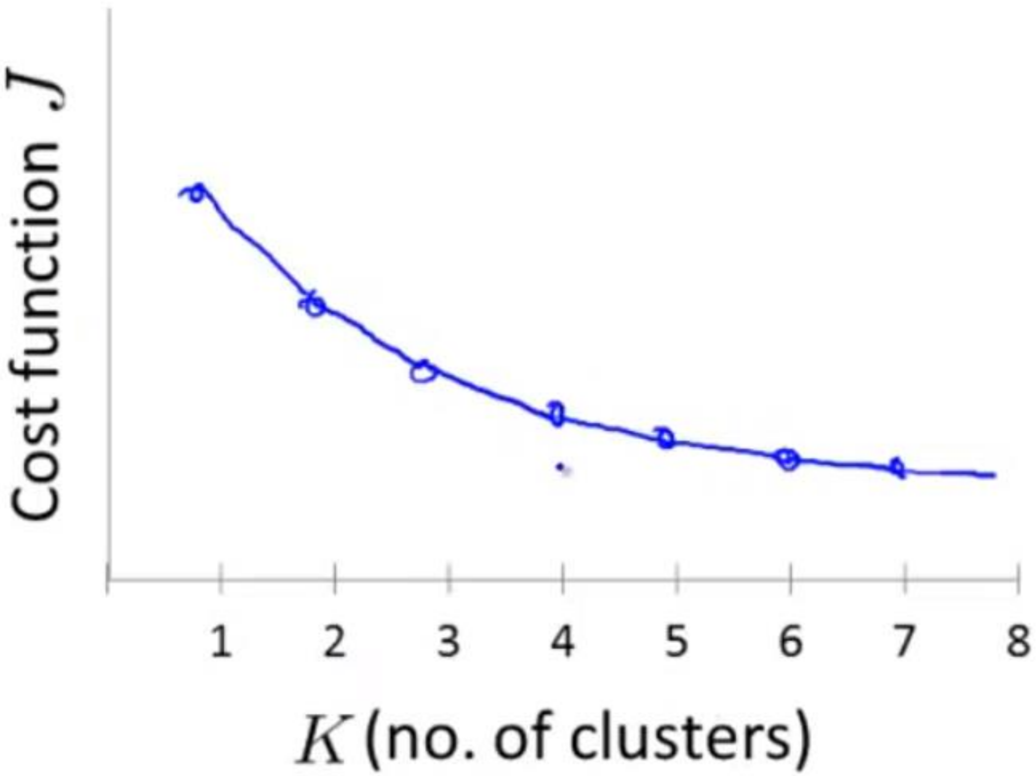
What is the right value of K?



Choosing the value of K

Elbow method:


Cost function J
"Elbow"
K (no. of clusters)


Cost function J
K (no. of clusters)