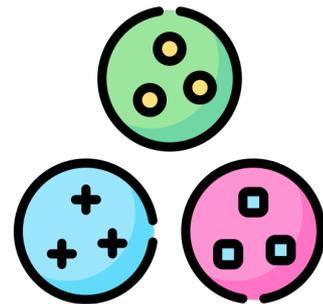


CSC380: Principles of Data Science

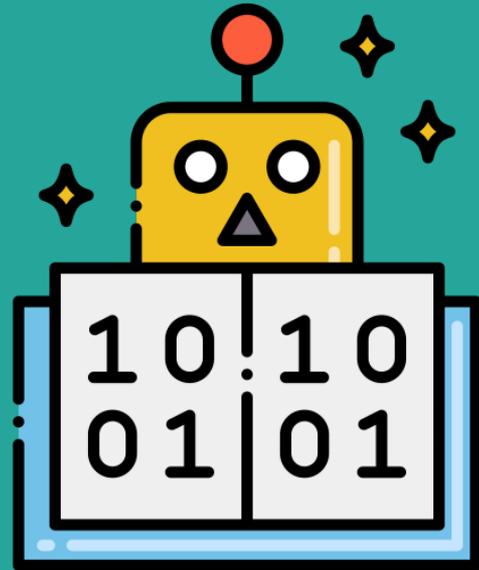
Clustering: K Means

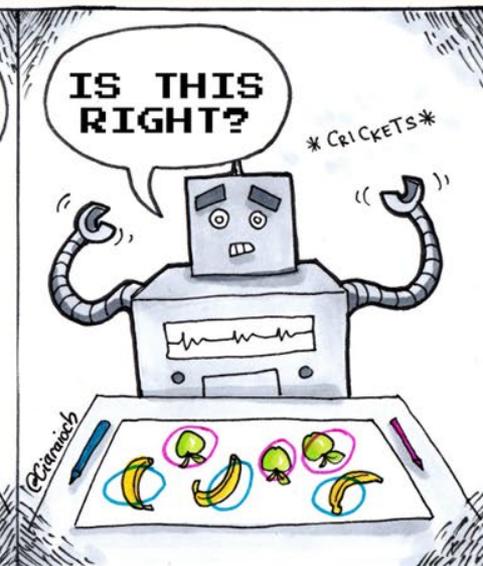
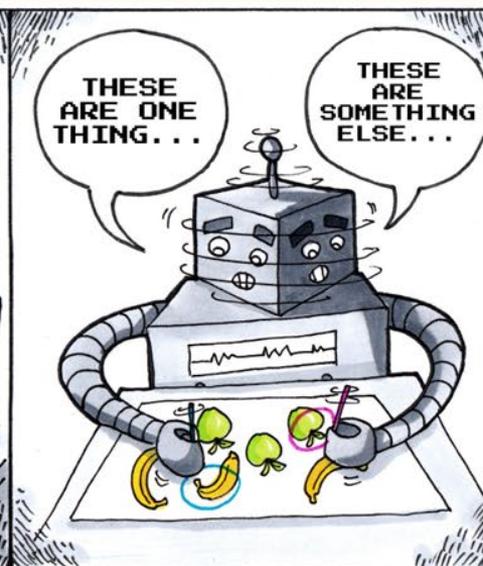
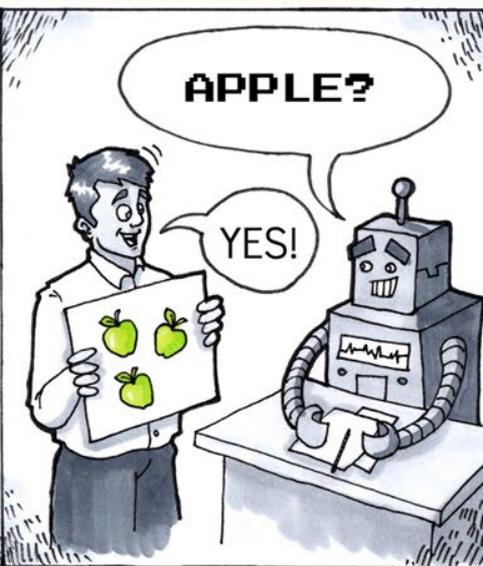
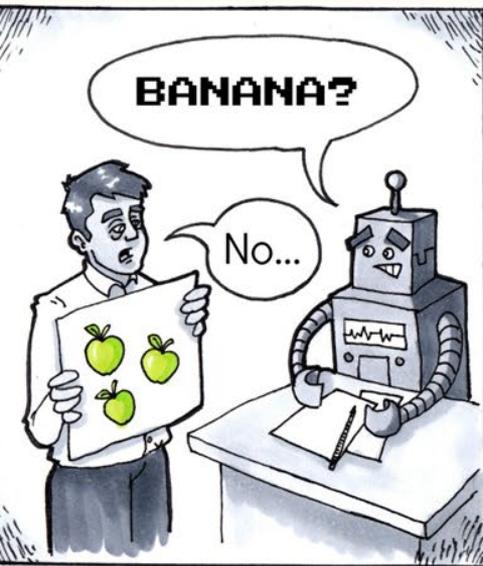


Outline

- Unsupervised Learning
 - Clustering
 - Types of Clustering
 - K Means Intuition
 - Formalising K Means
 - Convergence In K Means
 - Choosing an Optimum Number Of K
 - K-Means++
 - K-Medoids
 - Assumptions Made by KMeans
 - Application Case
 - How to Use K Means?
 - How can We Improve?

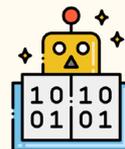
Unsupervised Learning



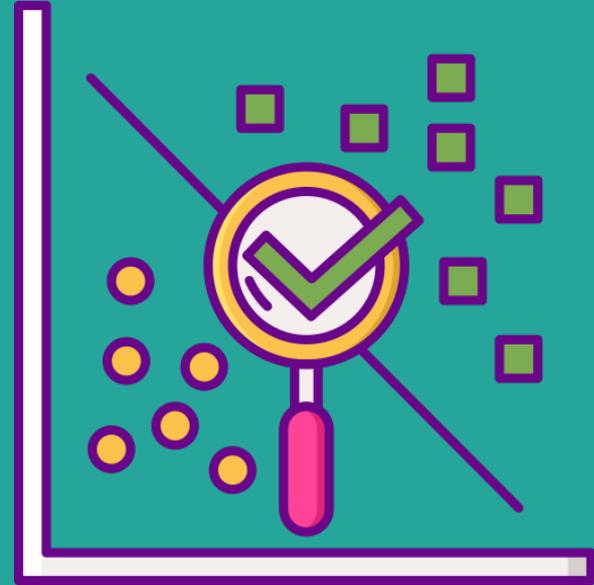


Supervised Learning

Unsupervised Learning



Clustering



Task 1 : Group These Set of Document into 3 Groups based on meaning

Doc1 : Health , Medicine, Doctor

Doc 2 : Machine Learning, Computer

Doc 3 : Environment, Planet

Doc 4 : Pollution, Climate Crisis

Doc 5 : Covid, Health , Doctor



Task 1 : Group These Set of Document into 3 Groups.

Doc1 : Health , Medicine, Doctor

Doc 2 : Machine Learning, Computer

Doc 3 : Environment, Planet

Doc 4 : Pollution, Climate Crisis

Doc 5 : Covid, Health , Doctor



Task 1 : Group These Set of Document into 3 Groups.

Doc1 : Health , Medicine, Doctor

Doc 5 : Covid, Health , Doctor

Doc 3 : Environment,
Planet

Doc 4 : Pollution, Climate
Crisis

Doc 2 : Machine
Learning, Computer



Supervised Learning

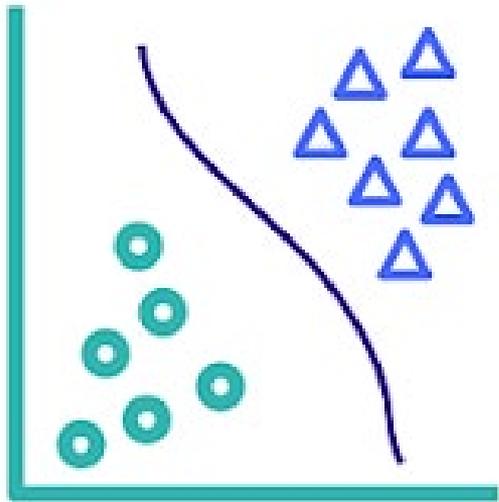
Unsupervised Learning

Discrete

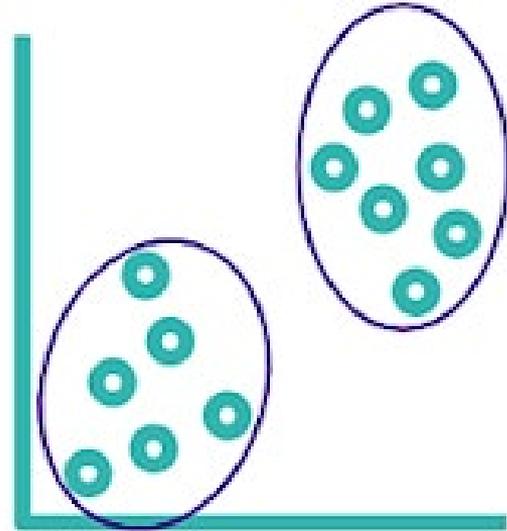
classification or
categorization

clustering

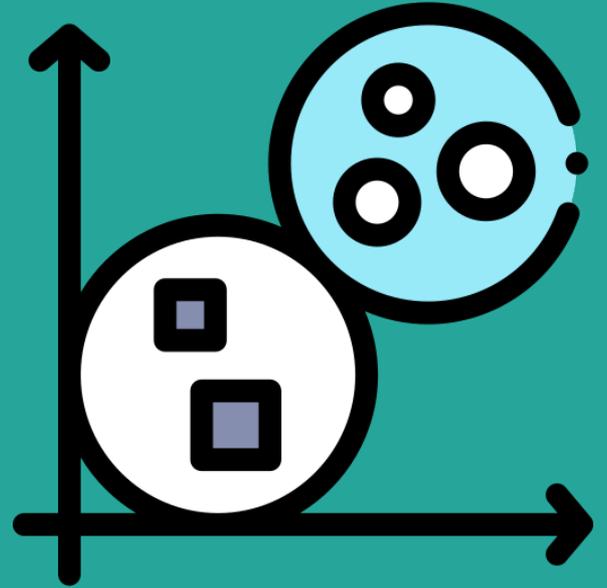




← Supervised
vs.
Unsupervised
Learning →

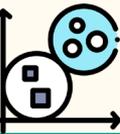
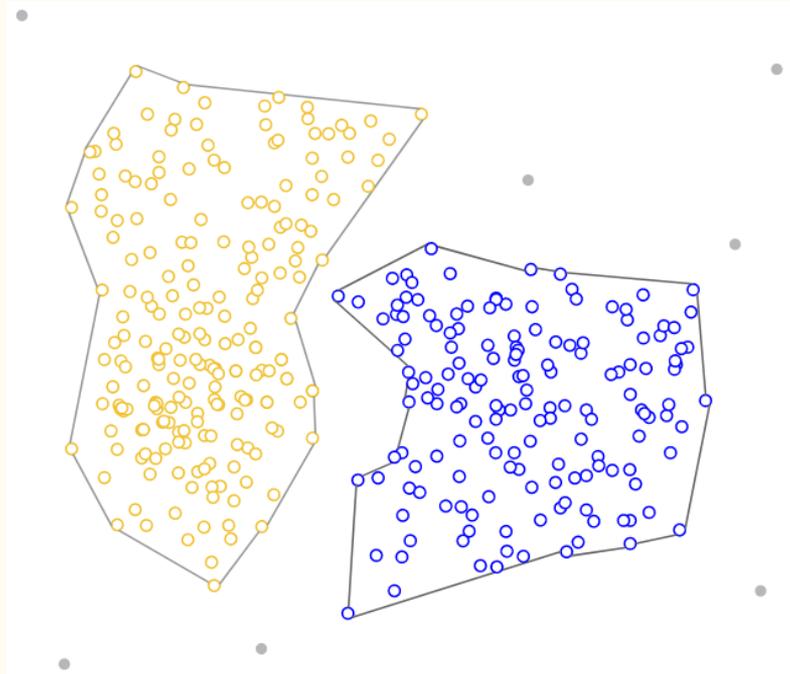


Types of Clustering

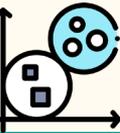
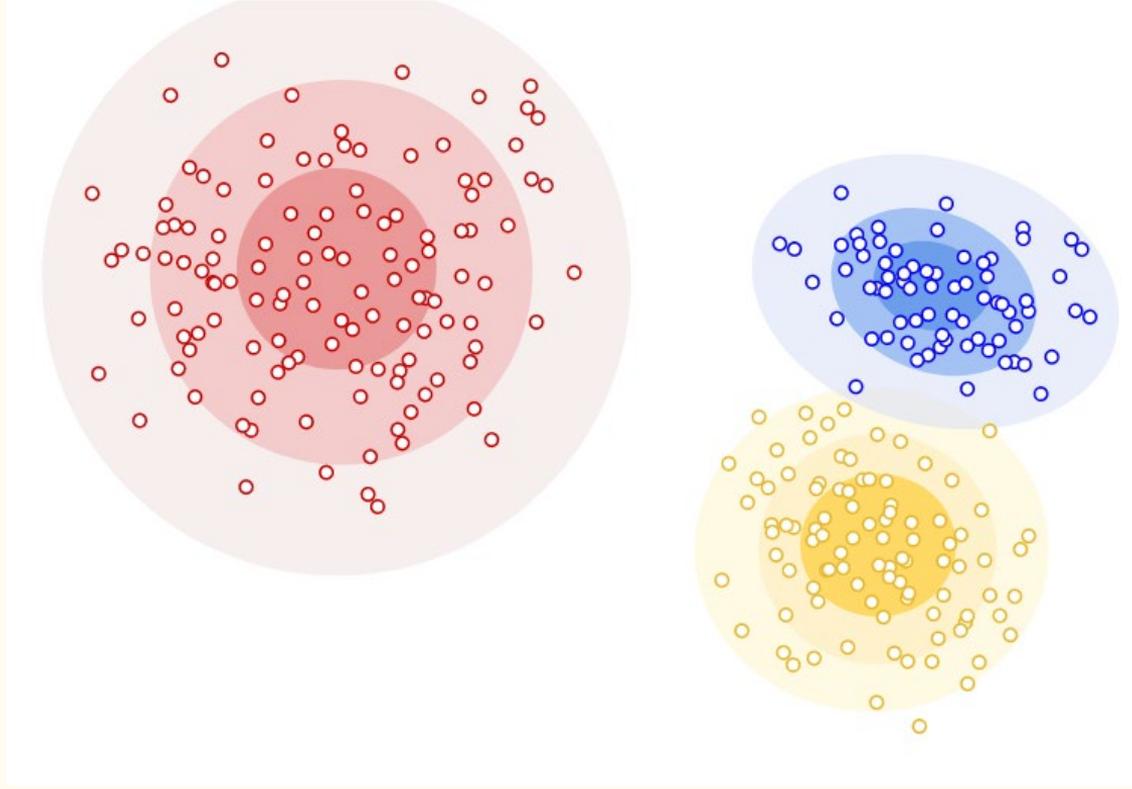


From [Clustering in Machine Learning - Google Developers](#)
For a comprehensive List : [A Comprehensive Survey of Clustering Algorithms](#)

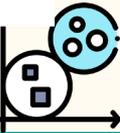
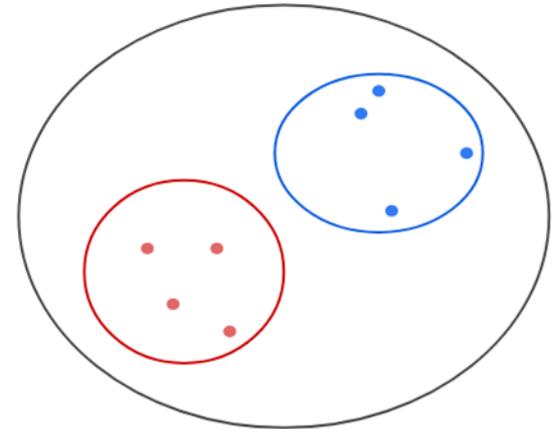
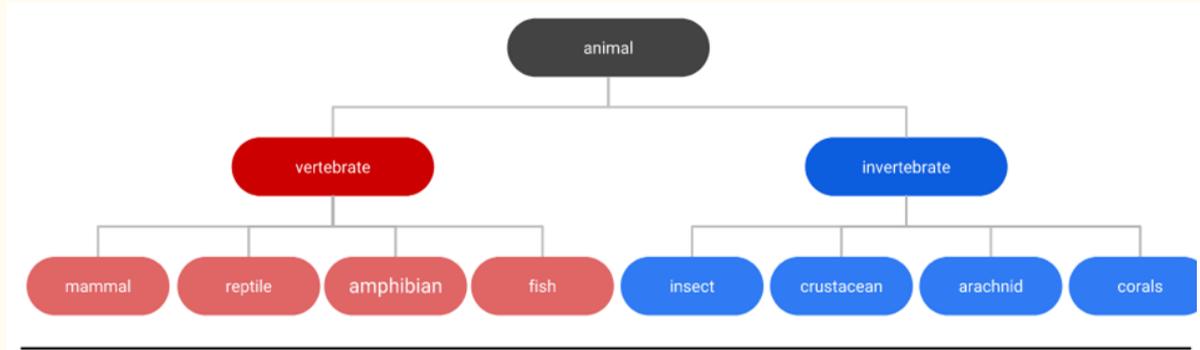
Density Based Clustering



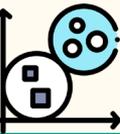
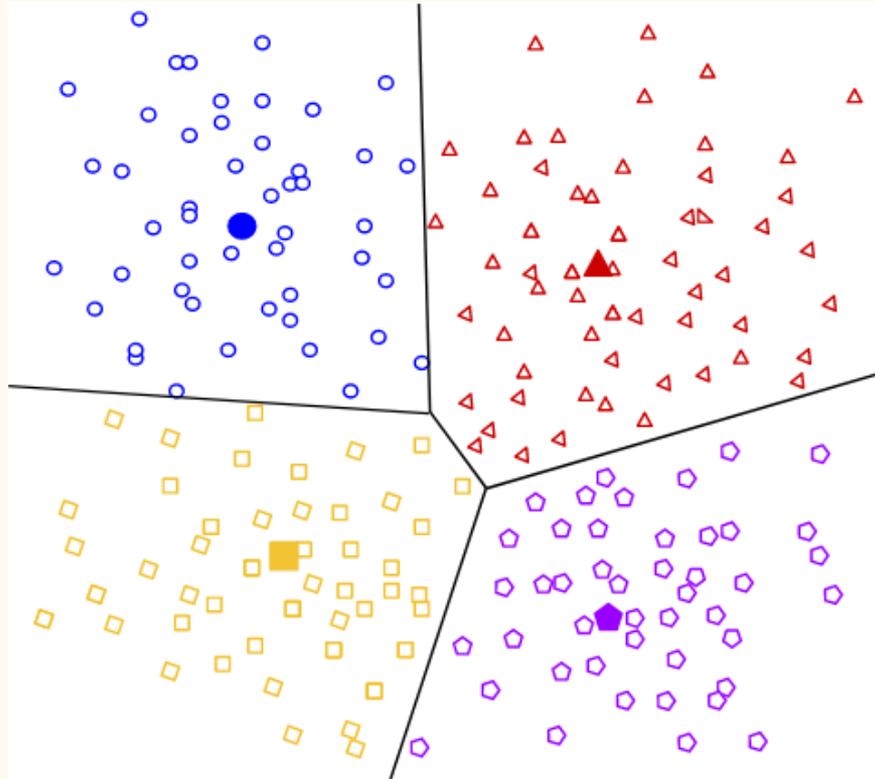
Distribution-based Clustering



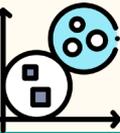
Hierarchical Clustering



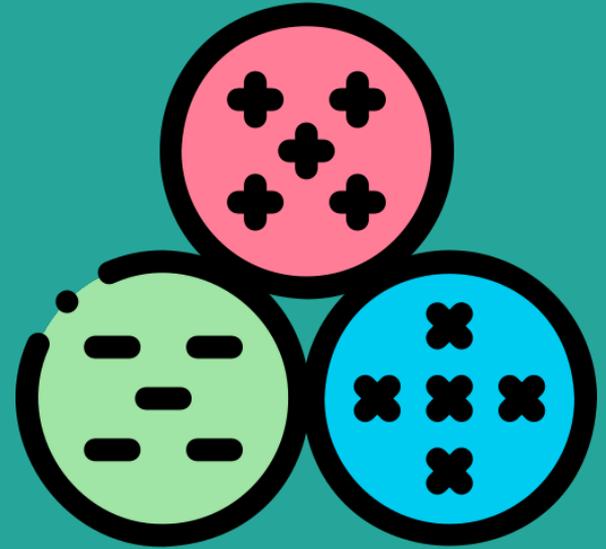
Centroid-based Clustering



One such Centroid Based Clustering Algorithm Is K-Means



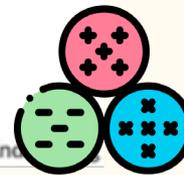
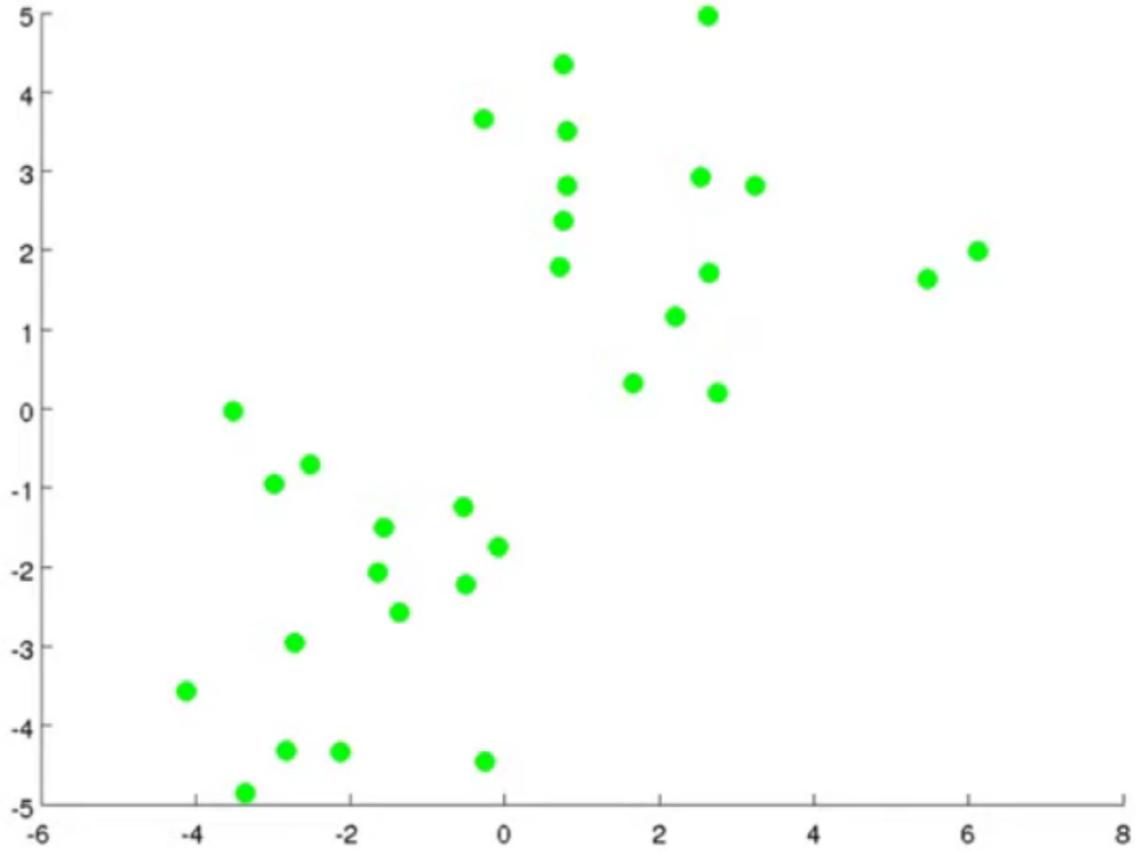
K Means Intuition

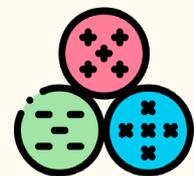
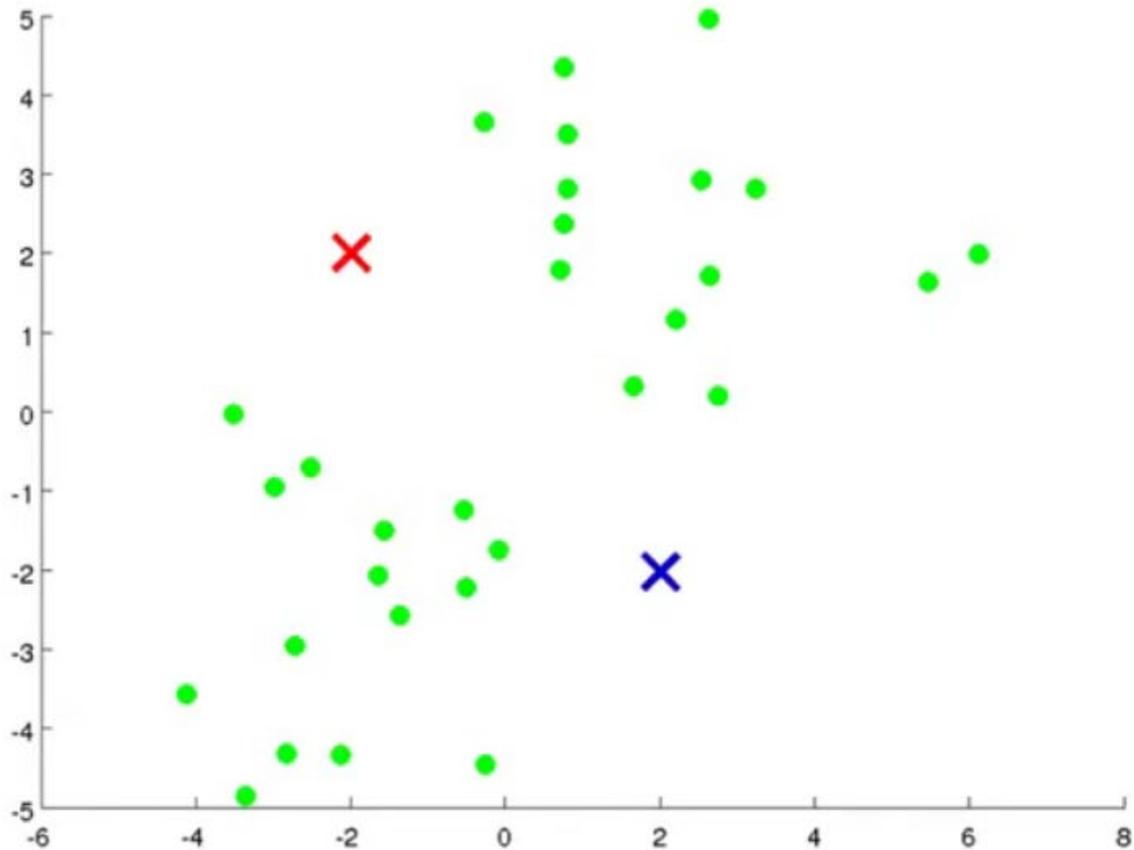


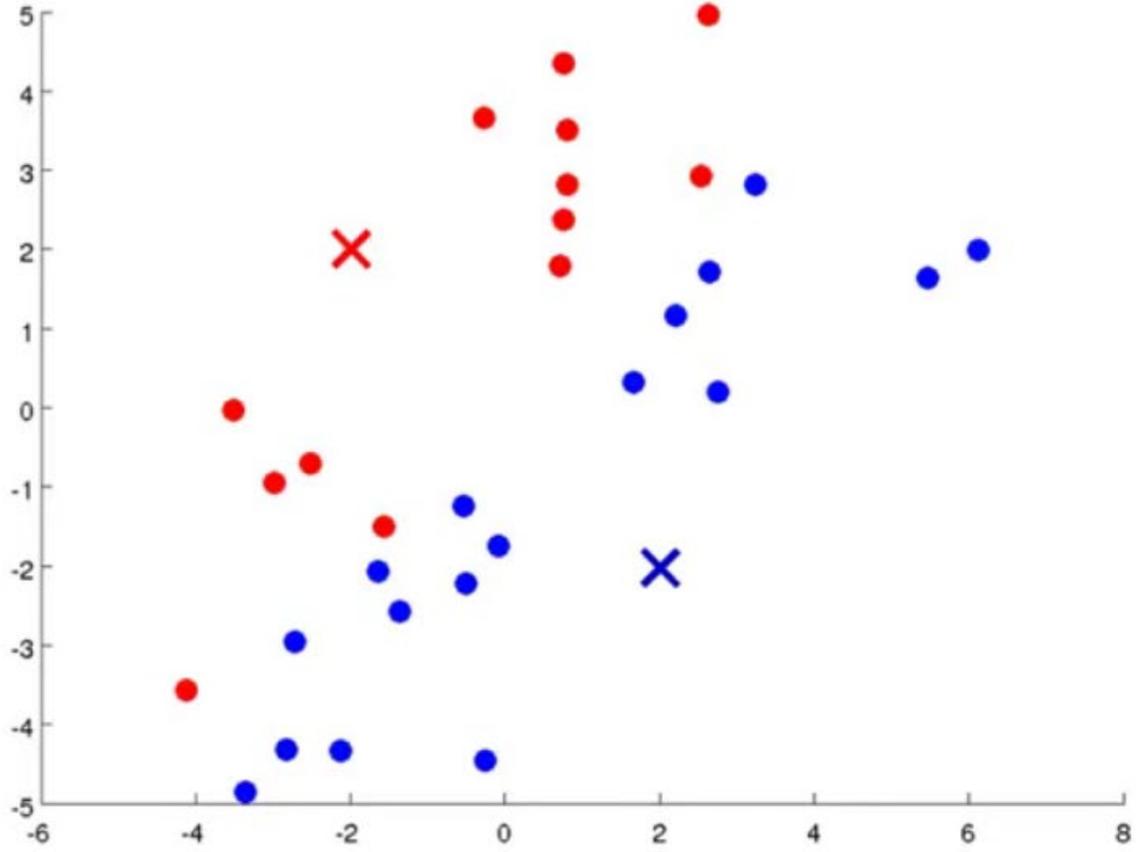
Basic Steps

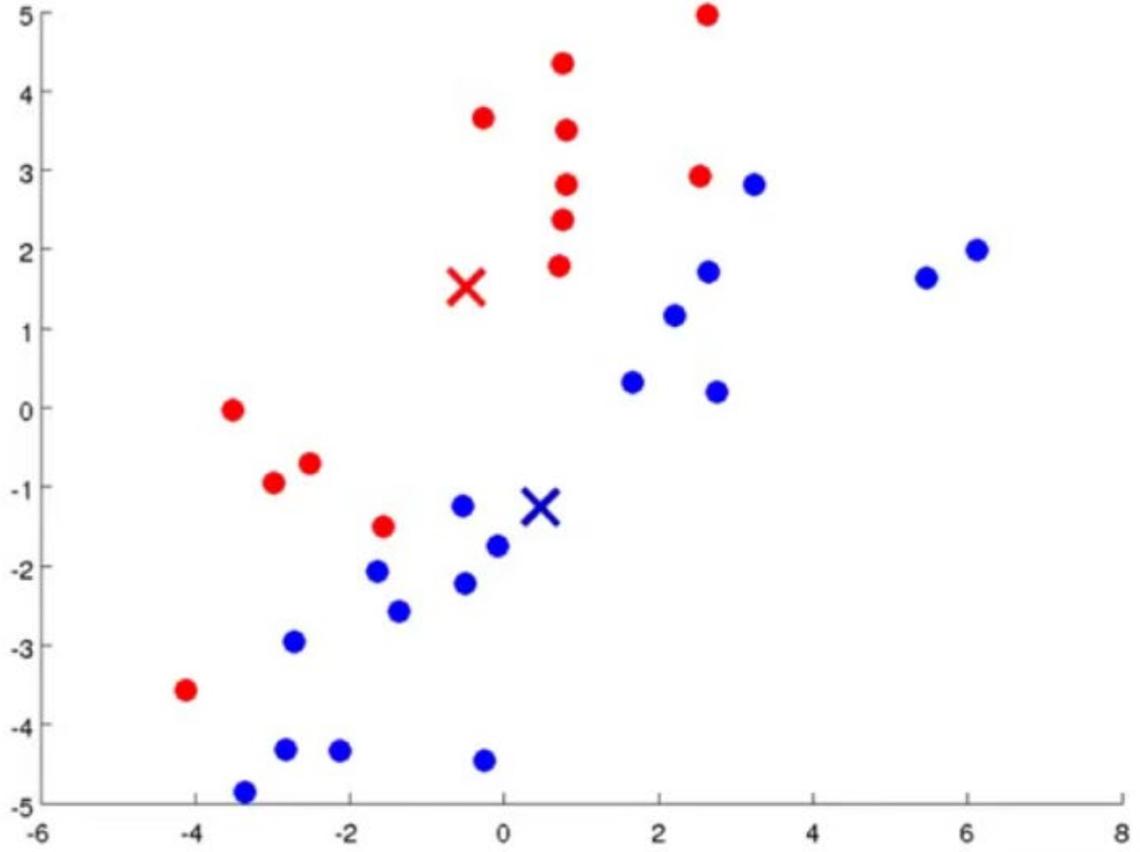
- Assign Cluster Centroids
- Until Convergence :
 - Cluster Assignment Step
 - Re-assigning Centroid Step

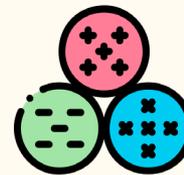
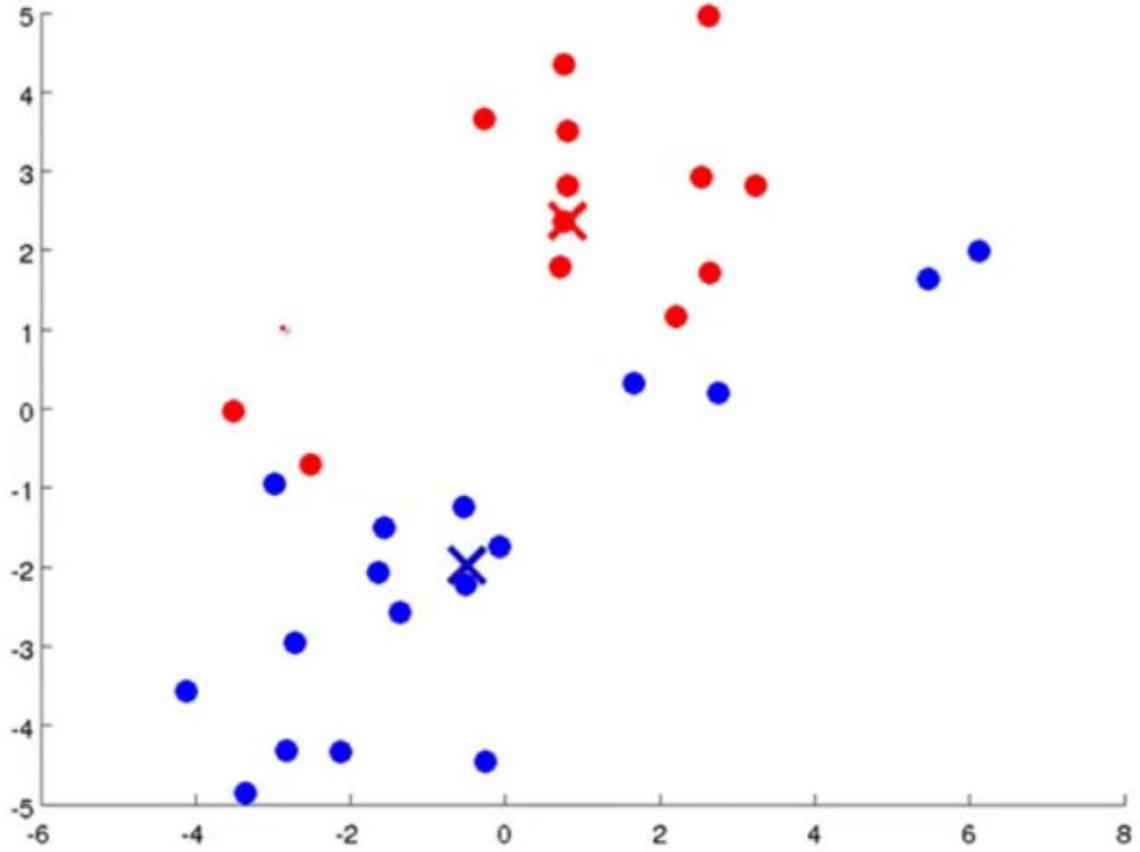


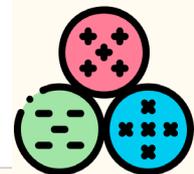
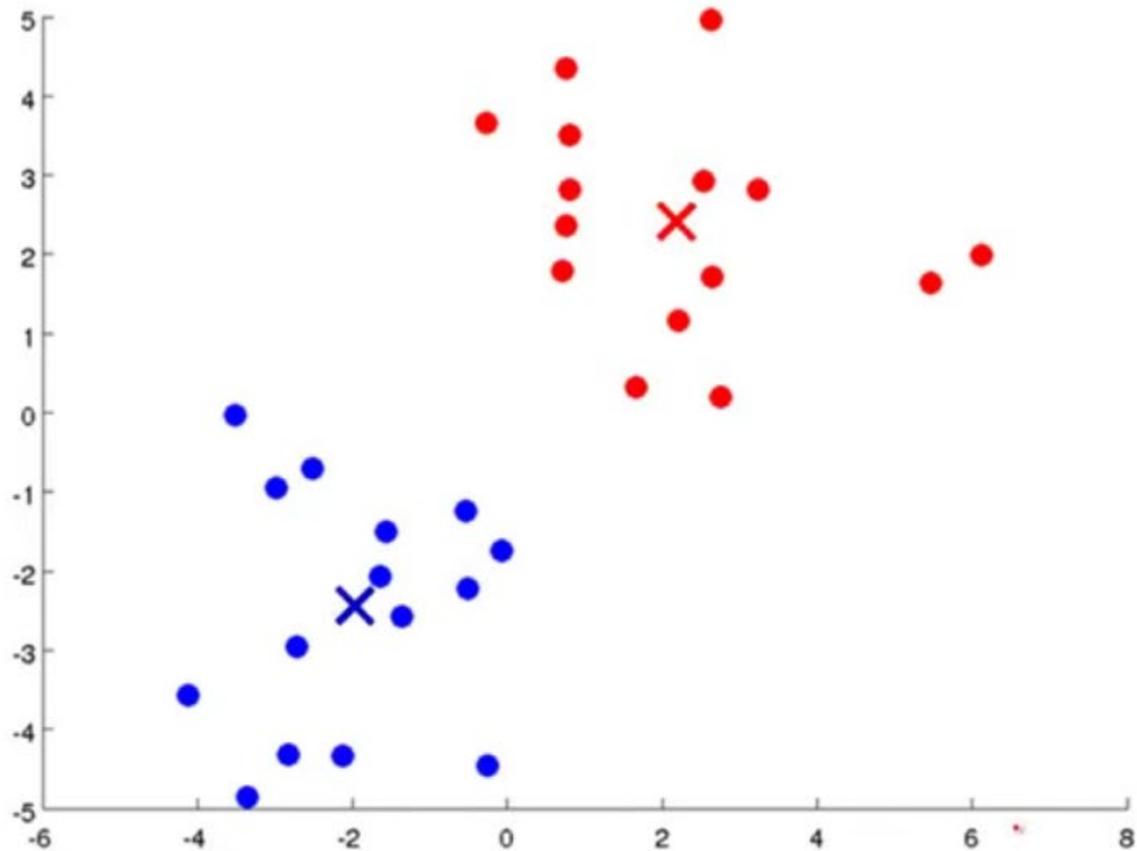


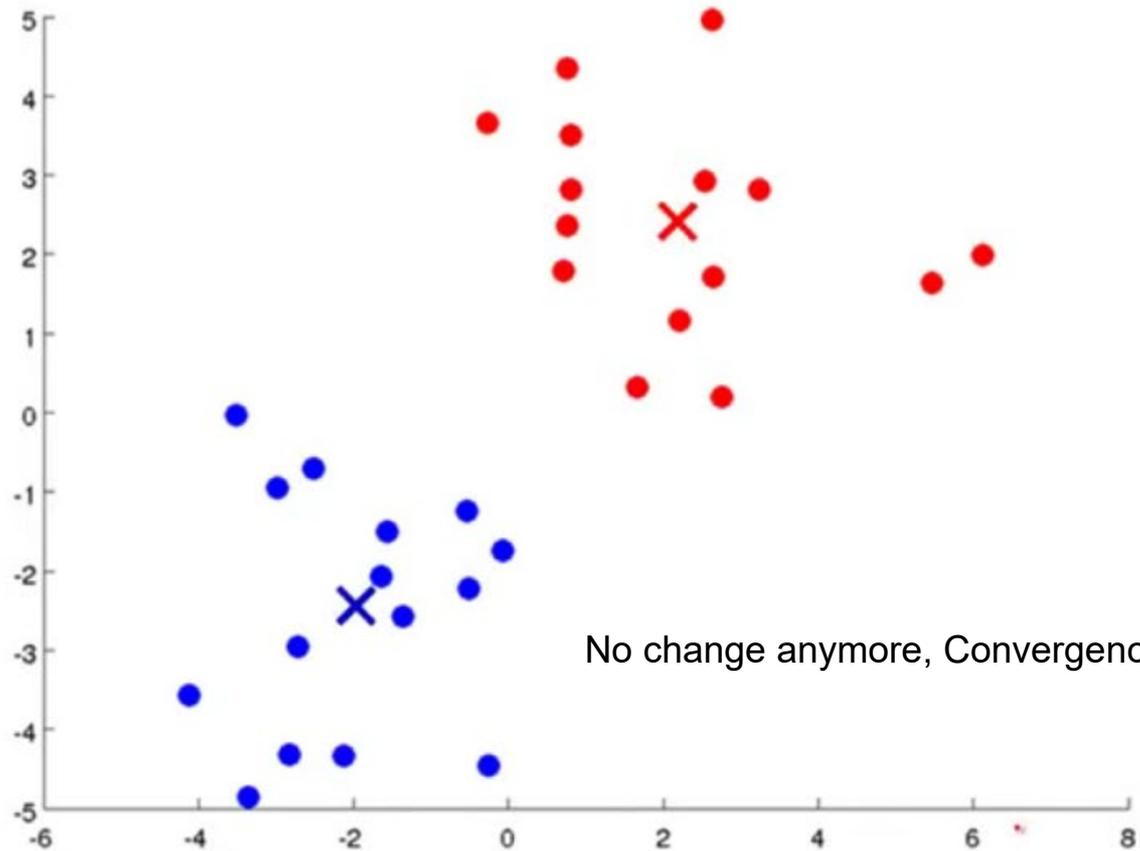




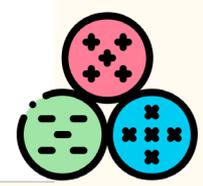








No change anymore, Convergence!



Basic Steps

- Assign Cluster Centroids
- Until Convergence :
 - Cluster Assignment Step
 - Re-assigning Centroid Step



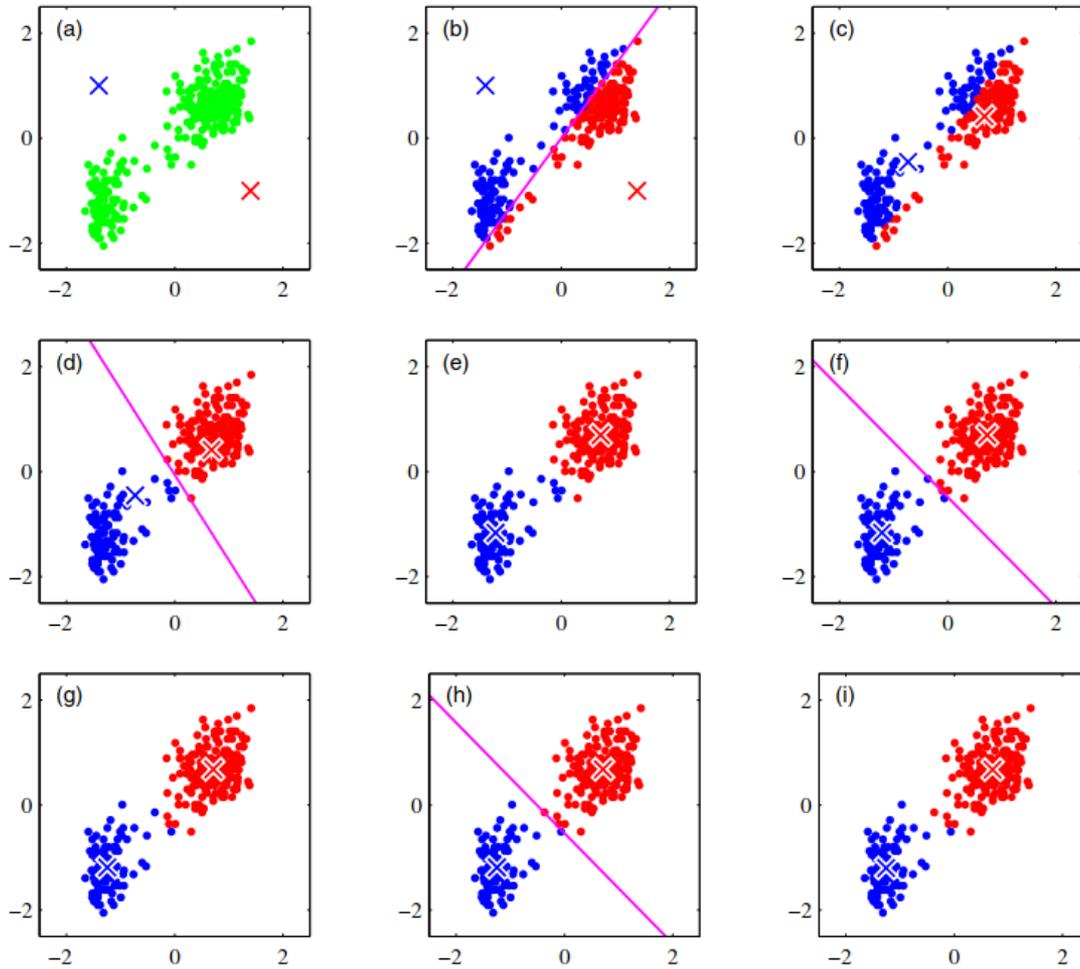
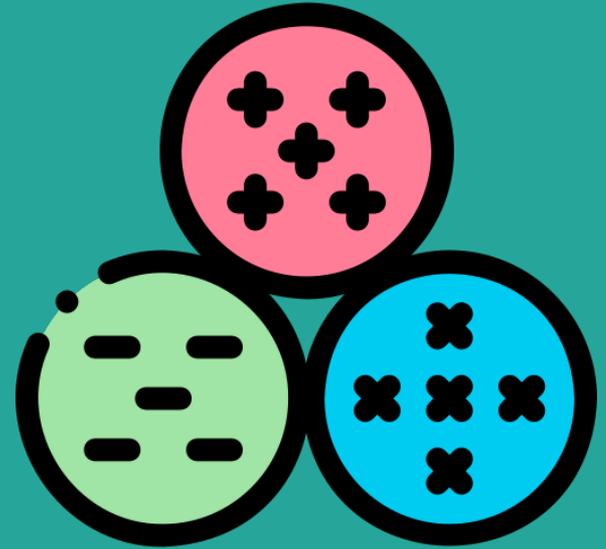


Fig 9.1 from Bishop - Pattern Recognition And Machine Learning

K Means Formalisation



What is a Cluster?

A group of data points whose **inter-point distances are small** compared with the distances to points outside of the cluster.



Clustering?

Consider a set of D -dimensional vectors μ_k , where $k = 1, \dots, K$, in which μ_k is a prototype associated with the k th cluster.



Clustering?

The goal of Clustering is then to find an assignment of data points to clusters, as well as a set of vectors $\{\mu_k\}$, such that the sum of the squares of the distances of each data point to its closest vector μ_k , is a minimum.



First we choose some initial values for the

$$\mu_k$$



Step1 : Assignment of data points to clusters

- 1-of-K coding scheme

For each data point x_n , we introduce a corresponding set of binary indicator variables $r_{nk} \in \{0, 1\}$, where $k = 1, \dots, K$ describing which of the K clusters the data point x_n is assigned to, so that if data point x_n is assigned to cluster k then $r_{nk} = 1$, and $r_{nj} = 0$ for $j \neq k$.



Objective function - Distortion Measure

$$J = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$



Step 2 : Assign Cluster to Each Point.

We minimize J with respect to the r_{nk} , keeping the μ_k fixed

$$r_{nk} = \begin{cases} 1 & \text{if } k = \arg \min_j \|\mathbf{x}_n - \boldsymbol{\mu}_j\|^2 \\ 0 & \text{otherwise.} \end{cases}$$



Step 3 : Reassignment of Centroids

We minimize J with respect to the μ_k , keeping r_{nk} fixed.

function J is a quadratic function of μ_k , and it can be minimized by setting its derivative with respect to μ_k to zero giving

$$2 \sum_{n=1}^N r_{nk} (\mathbf{x}_n - \mu_k) = 0 \quad (9.3)$$

which we can easily solve for μ_k to give

$$\mu_k = \frac{\sum_n r_{nk} \mathbf{x}_n}{\sum_n r_{nk}}. \quad (9.4)$$



This simply means -

μ_k equal to the mean of all of the data points x_n assigned to cluster k



This simply means -

μ_k equal to the mean of all of the data points x_n assigned to cluster k

Hence the name K Means



Kmeans Iteration 1

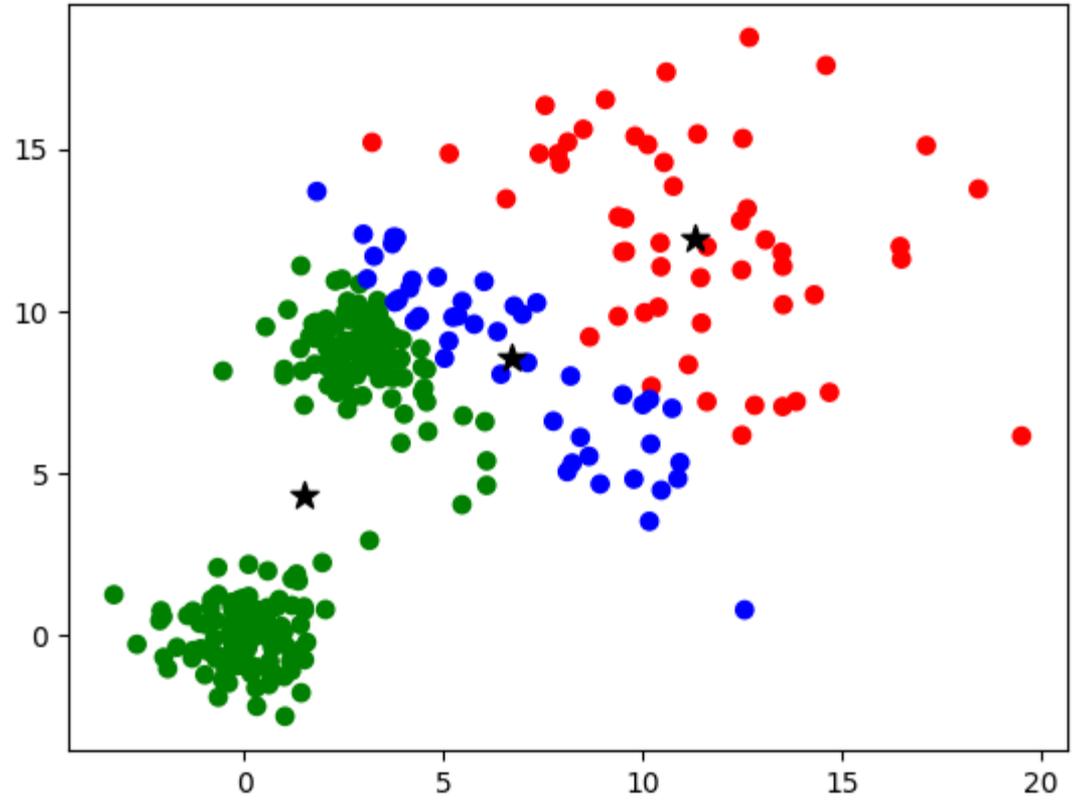


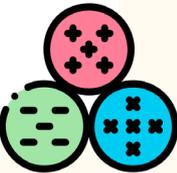
Image from [Floydhub](#)



K-means algorithm

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

```
Repeat {  
  for  $i = 1$  to  $m$   
     $c^{(i)}$  := index (from 1 to  $K$ ) of cluster centroid  
      closest to  $x^{(i)}$   
  for  $k = 1$  to  $K$   
     $\mu_k$  := average (mean) of points assigned to cluster  $k$   
}
```

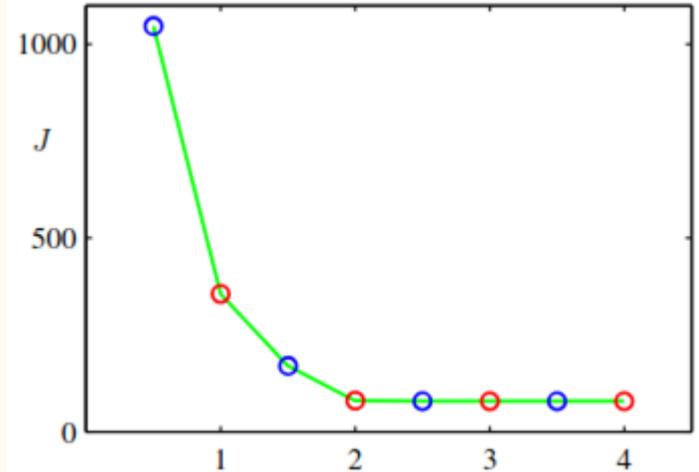


Convergence In K-Means



Promise of Convergence

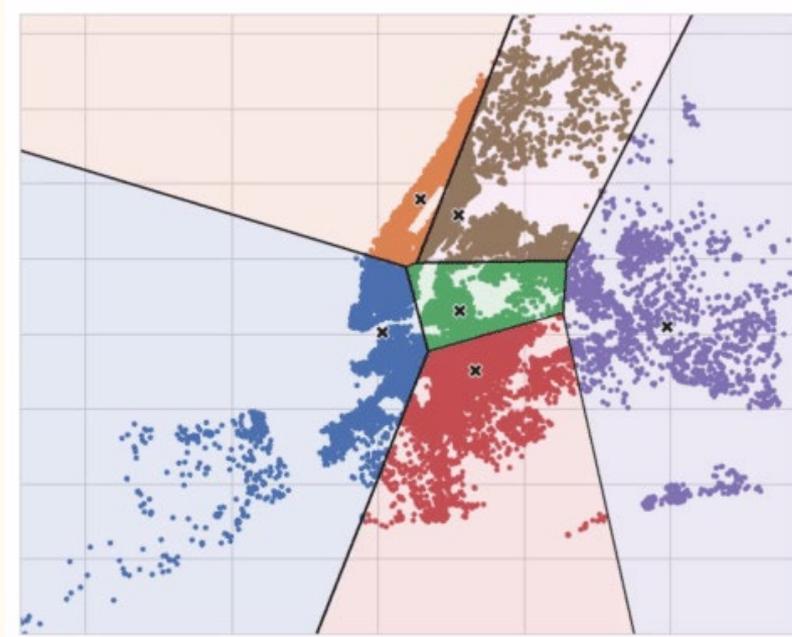
$$J = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$



Plot of the cost function J given by (9.1) after each E step (blue points) and M step (red points) of the K-means algorithm for the example shown in Figure 9.1.



Iterating until Convergence

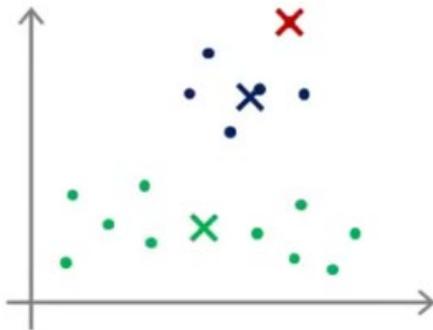
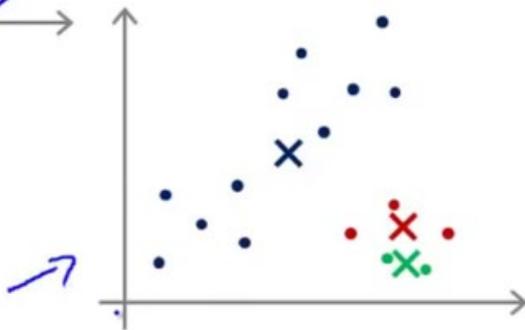
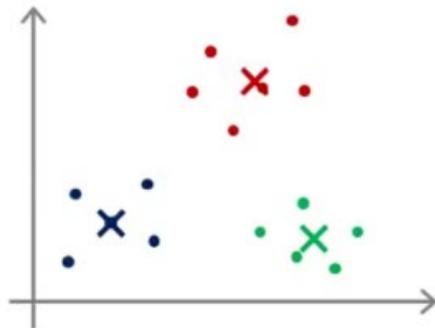
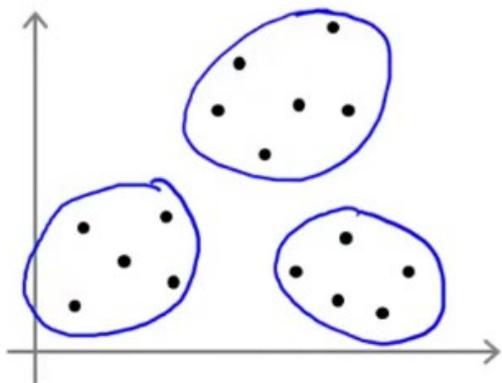


But,

It may converge to a local rather than global minimum of J .



Local optima



Andrew Ng



A cluster Has Just One Point?

Why ?



A cluster Has Just One Point?

Why ?

What to Do?



Choosing The Number Of K

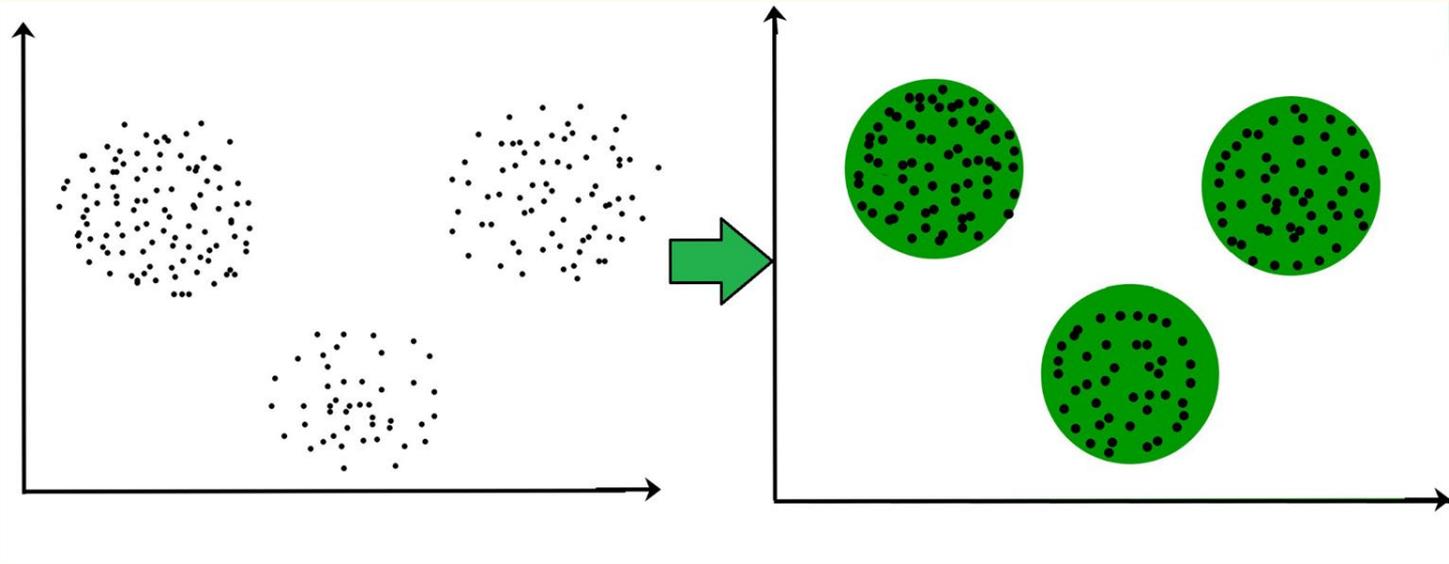


How to Choose Number of K?

- Most common approach is Visualise, and then pick manually

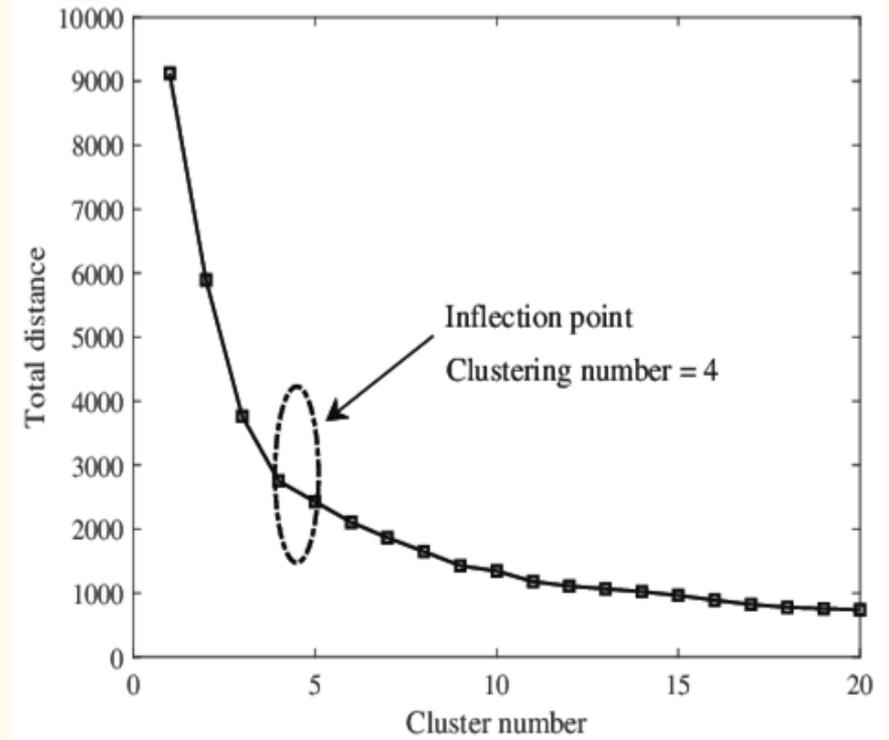


Most common approach is Visualise, and then pick manually



How to Choose Number of K?

The Elbow Method



How to Choose Number of K?

But sometimes it doesn't work

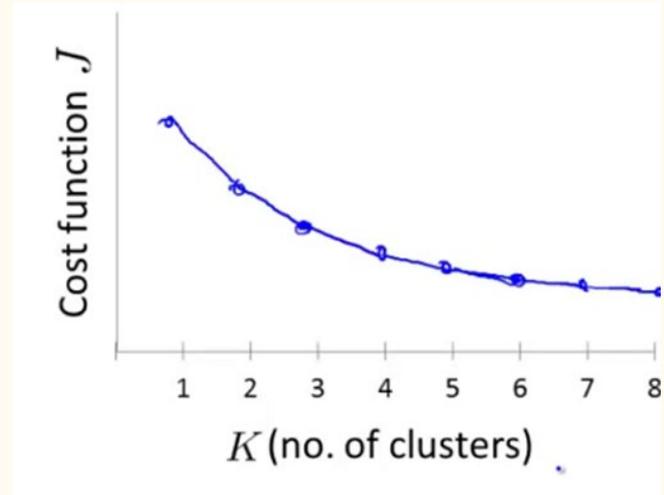
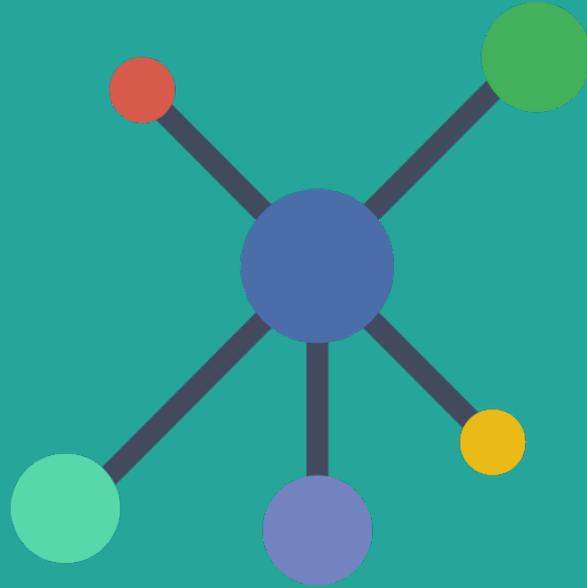


Image Source : Andrew NG Machine Learning



KMeans ++



Is there a way to start Smarter?

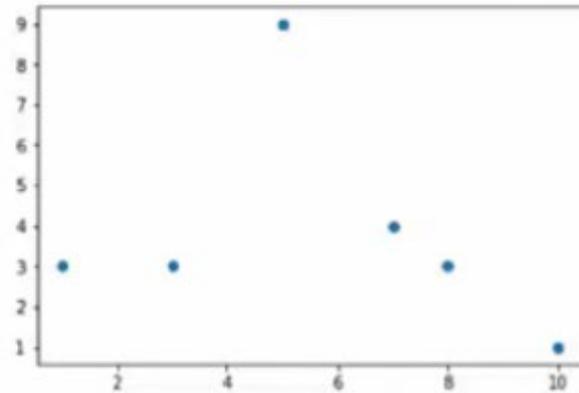


Suppose we have the small dataset

☛ $[(7,4),(8,3),(5,9),(3,3),(1,3),(10,1)]$ to which we wish to assign 3 clusters.

We begin by randomly selecting $(7,4)$ to be a cluster center.

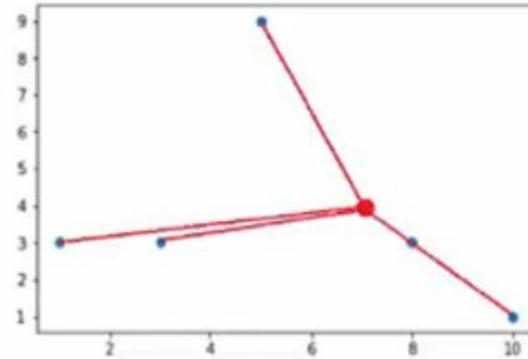
x	$\min(d(x, z_i)^2)$
$(7,4)$	
$(8,3)$	
$(5,9)$	
$(3,3)$	
$(1,3)$	
$(10,1)$	



Suppose we have the small dataset
[[7,4),(8,3),(5,9),(3,3),(1,3),(10,1)] to which we wish to assign 3
clusters.

We begin by randomly selecting (7,4) to be a cluster center.

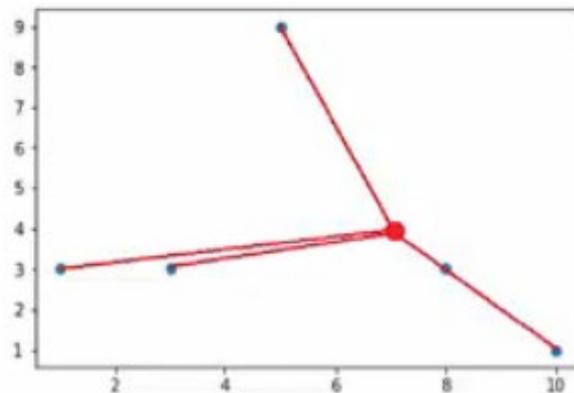
x	$\min(d(x, z_i)^2)$
(7,4)	-
(8,3)	2
(5,9)	29
(3,3)	17
(1,3)	37
(10,1)	18



Suppose we have the small dataset $[(7,4),(8,3),(5,9),(3,3),(1,3),(10,1)]$ to which we wish to assign 3 clusters.

We begin by randomly selecting $(7,4)$ to be a cluster center.

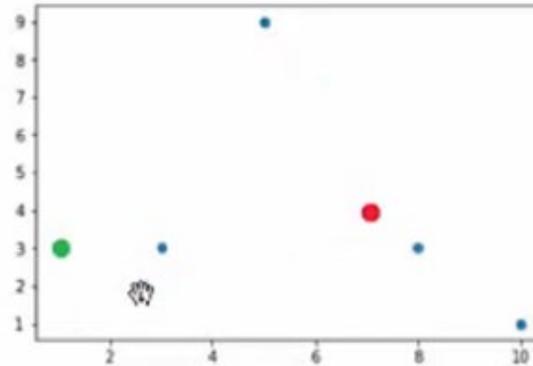
x	prob
$(7,4)$	-
$(8,3)$	$2/103$
$(5,9)$	$29/103$
$(3,3)$	$17/103$
$(1,3)$	$37/103$
$(10,1)$	$18/103$



Suppose we have the small dataset $[(7,4),(8,3),(5,9),(3,3),(1,3),(10,1)]$ to which we wish to assign 3 clusters.

We add $(1,3)$ to the list of cluster centers.

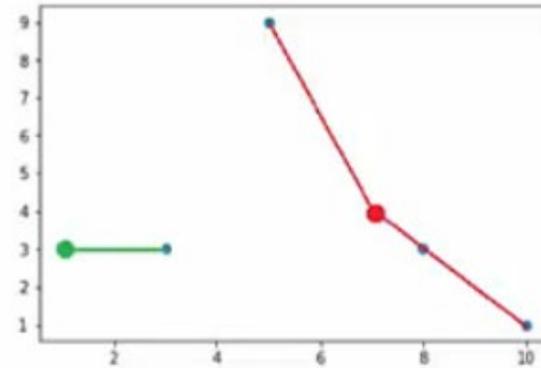
x	$\min(d(x, z_i)^2)$
$(7,4)$	-
$(8,3)$	
$(5,9)$	
$(3,3)$	
$(1,3)$	-
$(10,1)$	



Suppose we have the small dataset $[(7,4),(8,3),(5,9),(3,3),(1,3),(10,1)]$ to which we wish to assign 3 clusters.

We add $(1,3)$ to the list of cluster centers.

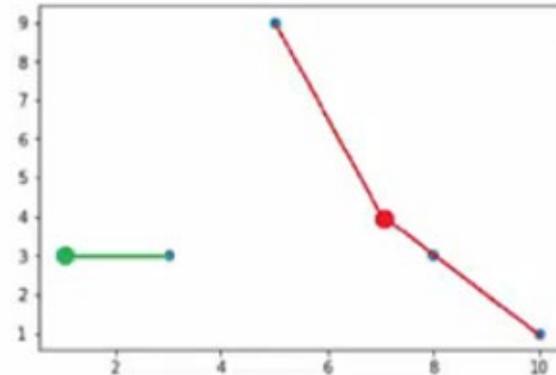
x	$\min(d(x, z_i)^2)$
$(7,4)$	-
$(8,3)$	2
$(5,9)$	29
$(3,3)$	4
$(1,3)$	-
$(10,1)$	18



Suppose we have the small dataset $[(7,4),(8,3),(5,9),(3,3),(1,3),(10,1)]$ to which we wish to assign 3 clusters.

We add $(1,3)$ to the list of cluster centers.

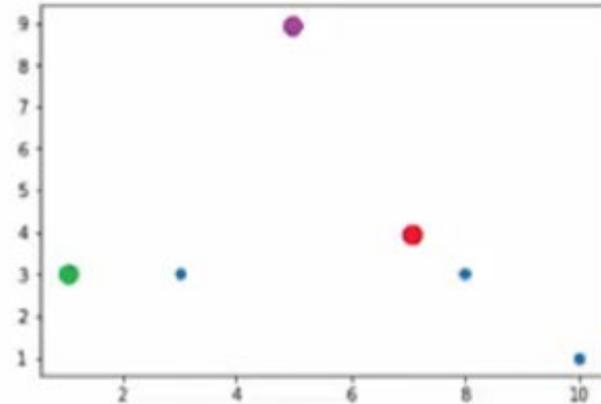
x	prob
$(7,4)$	-
$(8,3)$	$2/55$
$(5,9)$	$29/55$
$(3,3)$	$4/55$
$(1,3)$	-
$(10,1)$	$18/55$



Suppose we have the small dataset
[[7,4),(8,3),(5,9),(3,3),(1,3),(10,1)] to which we wish to assign 3
clusters.

We add (5,9) to the list of cluster centers.

x	prob
(7,4)	-
(8,3)	
(5,9)	-
(3,3)	
(1,3)	-
(10,1)	

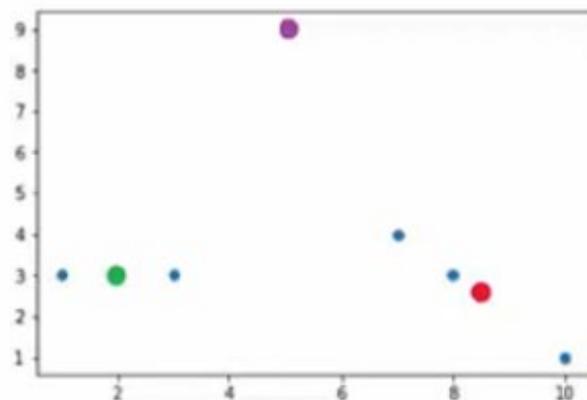


Suppose we have the small dataset

$[(7,4),(8,3),(5,9),(3,3),(1,3),(10,1)]$ to which we wish to assign 3 clusters.

We now run k -means with initialized centers $(7,4)$, $(1,3)$, and $(5,9)$.

x	prob
$(7,4)$	-
$(8,3)$	
$(5,9)$	-
$(3,3)$	
$(1,3)$	-
$(10,1)$	



K Medoids

- The issue with Squared Euclidean Distance
 - Type of Data
 - Reaction to Outliers
- How can we generalise better?



K-medoids algorithm

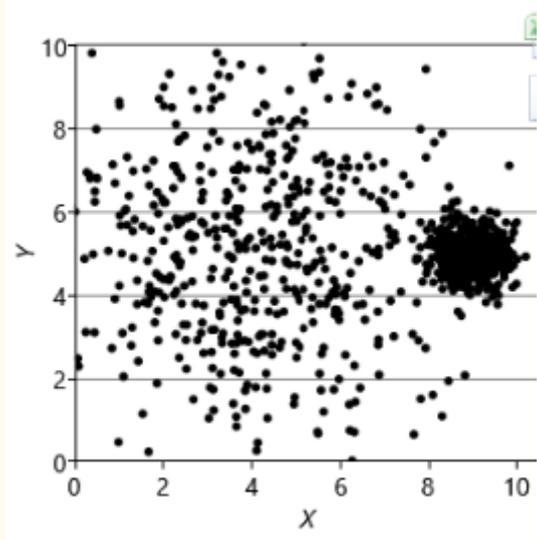
$$\tilde{J} = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \mathcal{V}(\mathbf{x}_n, \boldsymbol{\mu}_k)$$



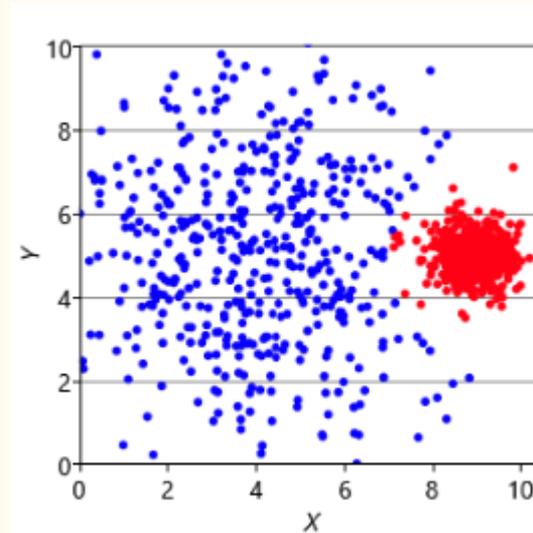
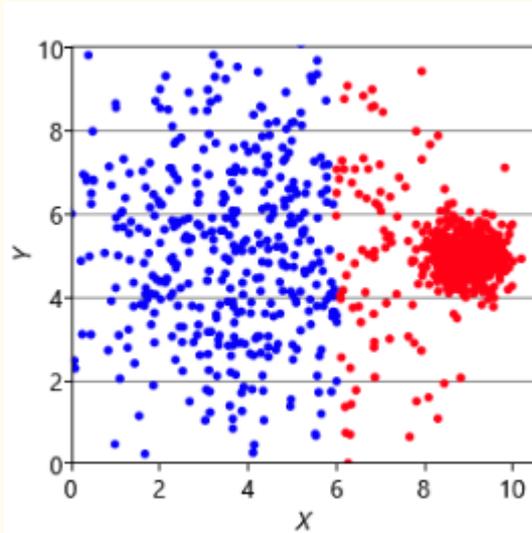
Assumptions made by K Means

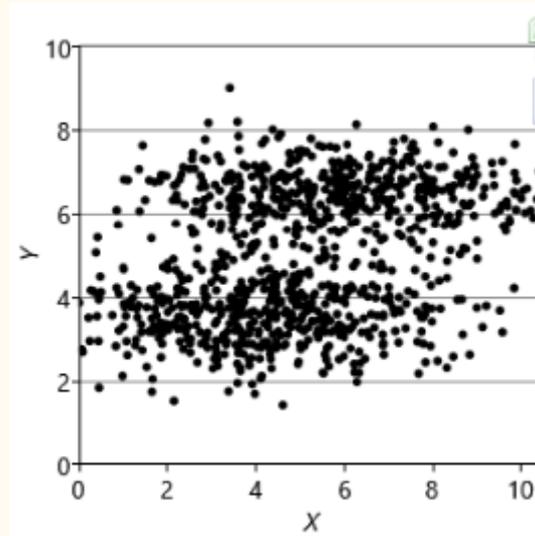


Based on [mlbmlbook](#)

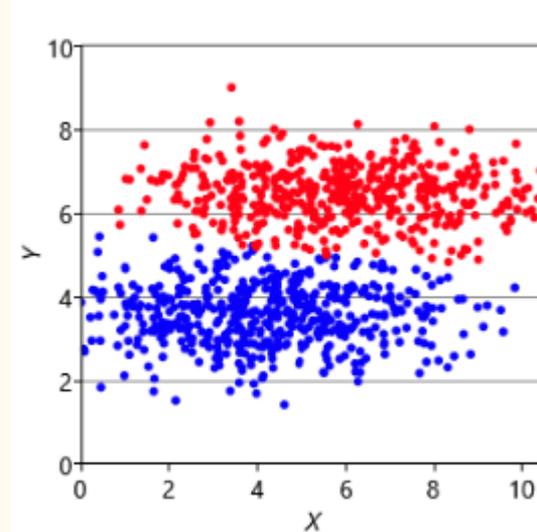
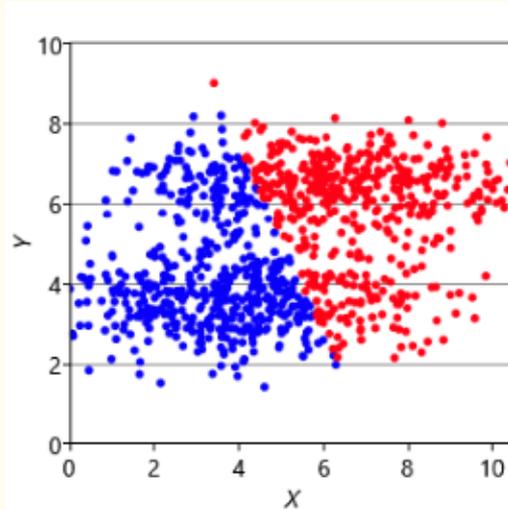


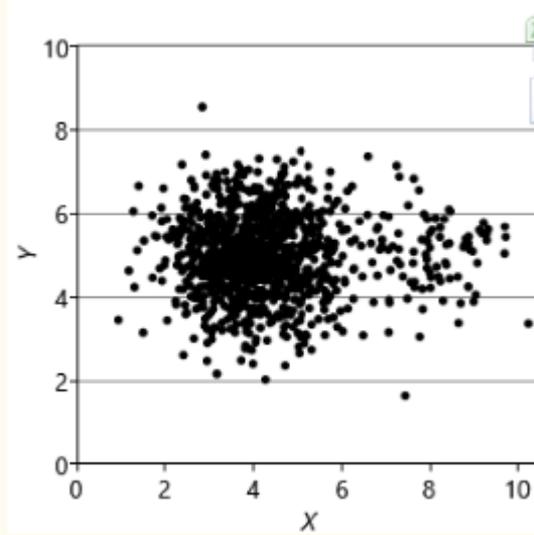
1. All clusters are the same size.(Area not Cardinality)



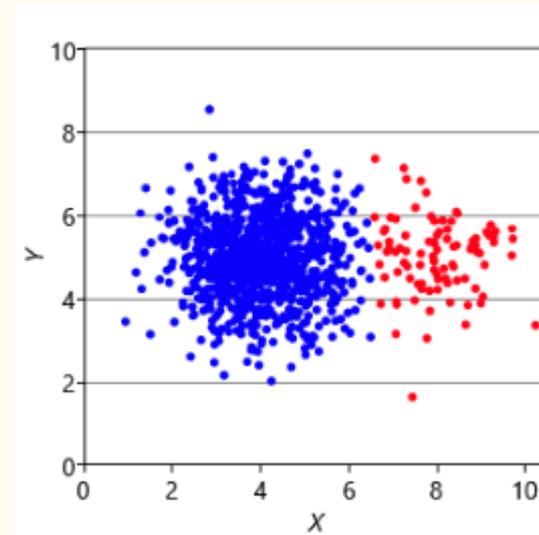
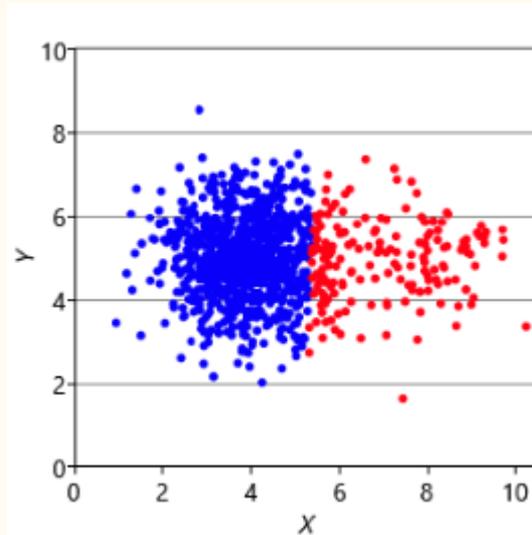


2. Clusters have the same extent in every direction.





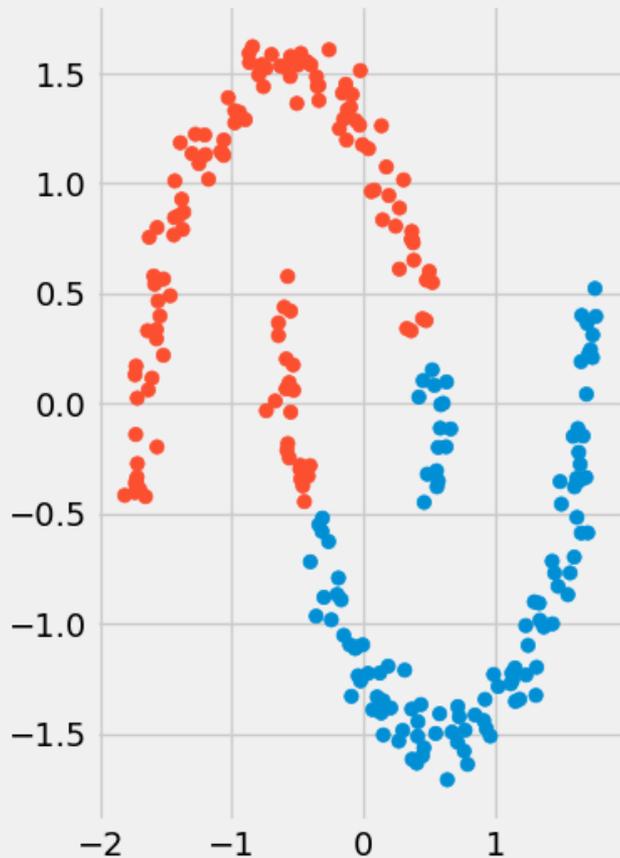
3. Clusters have similar numbers of points assigned to them.



Clustering Algorithm Comparison: Crescents

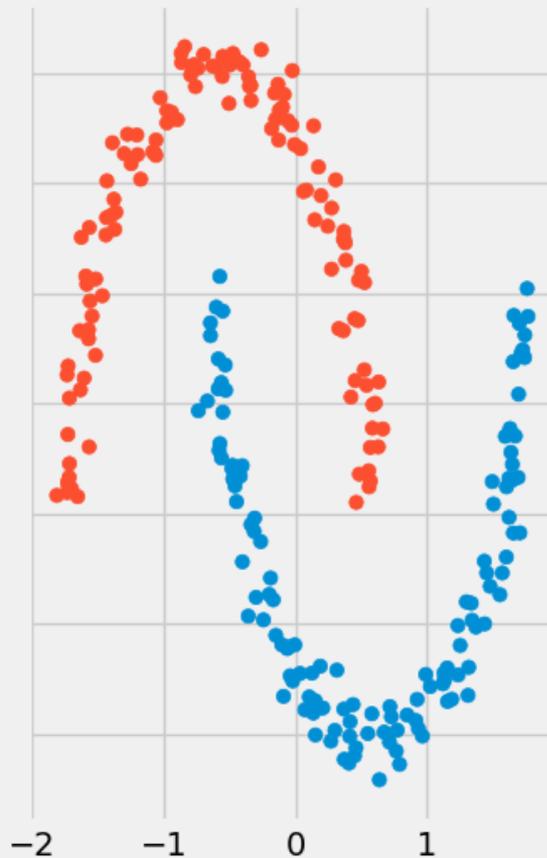
k-means

Silhouette: 0.5



DBSCAN

Silhouette: 0.38



Application Case:

**Image Segmentation and
Compression**



What is Segmentation?

Segmentation is to partition an image into regions each of which has a reasonably homogeneous visual appearance or which corresponds to objects or parts of objects



Original image



$K = 10$



Original image



$K = 3$



Original image



$K = 2$



How to Use?



sklearn.cluster.KMeans

```
class sklearn.cluster.KMeans(n_clusters=8, *, init='k-means++', n_init=10, max_iter=300, tol=0.0001, verbose=0,
random_state=None, copy_x=True, algorithm='auto')
```

[\[source\]](#)

```
>>> from sklearn.cluster import KMeans
>>> import numpy as np
>>> X = np.array([[1, 2], [1, 4], [1, 0],
...             [10, 2], [10, 4], [10, 0]])
>>> kmeans = KMeans(n_clusters=2, random_state=0).fit(X)
>>> kmeans.labels_
array([1, 1, 1, 0, 0, 0], dtype=int32)
>>> kmeans.predict([[0, 0], [12, 3]])
array([1, 0], dtype=int32)
>>> kmeans.cluster_centers_
array([[10.,  2.],
       [ 1.,  2.]])
```

>>>



```
# Importing the dataset
```

```
dataset = pd.read_csv('../input/Mall_Customers.csv', index_col='CustomerID')
```

```
dataset.head()
```

	Genre	Age	Annual_Income_(k\$)	Spending_Score
CustomerID				
1	Male	19	15	39
2	Male	21	15	81
3	Female	20	16	6
4	Female	23	16	77
5	Female	31	17	40

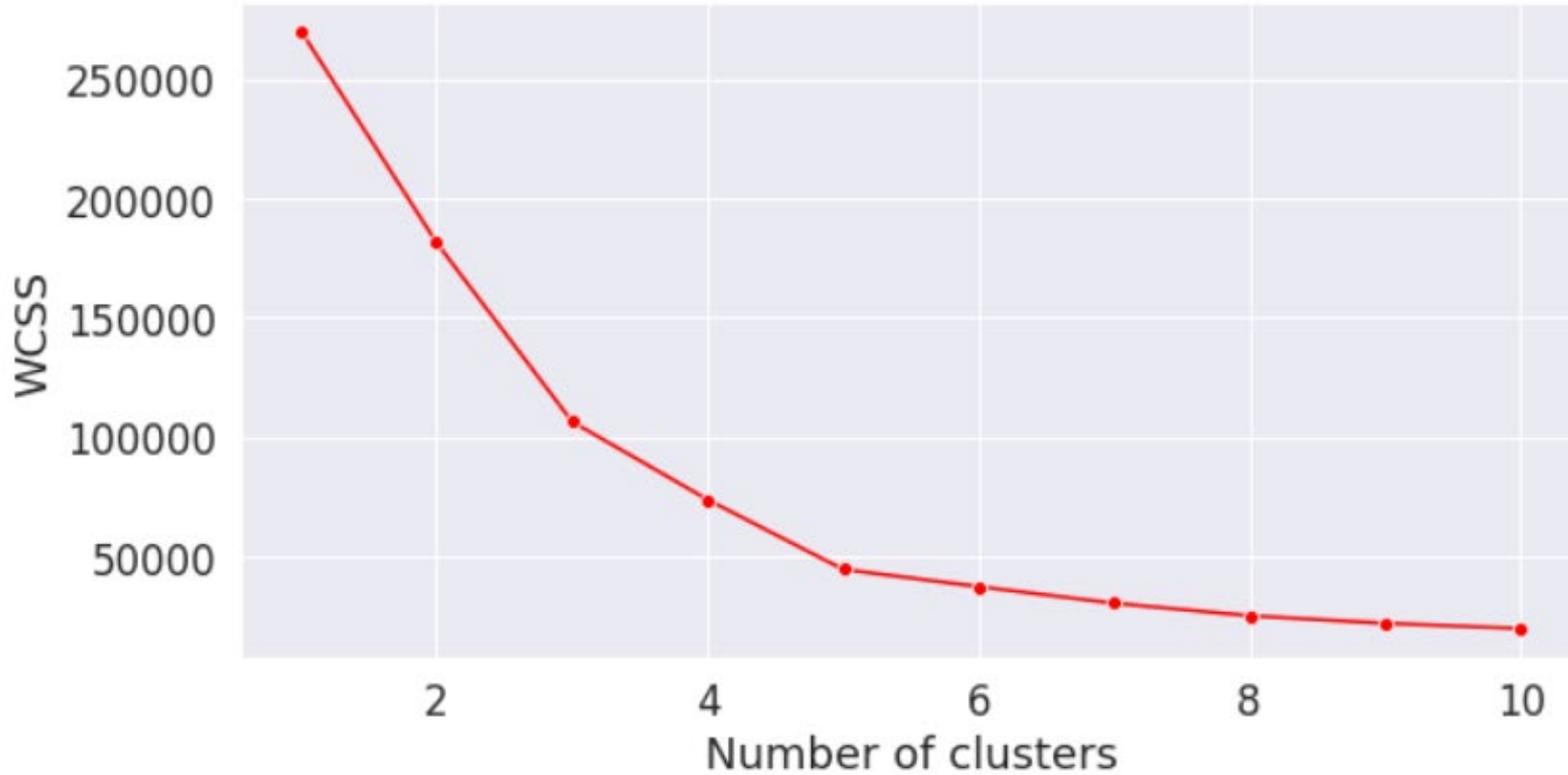


```
# Using the elbow method to find the optimal number of clusters
from sklearn.cluster import KMeans
wcss = []
for i in range(1, 11):
    kmeans = KMeans(n_clusters = i, init = 'k-means++', random_state = 42)
    kmeans.fit(X)
    # inertia method returns wcss for that model
    wcss.append(kmeans.inertia_)
```

```
plt.figure(figsize=(10,5))
sns.lineplot(range(1, 11), wcss,marker='o',color='red')
plt.title('The Elbow Method')
plt.xlabel('Number of clusters')
plt.ylabel('WCSS')
plt.show()
```



The Elbow Method



```
# Fitting K-Means to the dataset
```

```
kmeans = KMeans(n_clusters = 5, init = 'k-means++', random_state = 42)
```

```
y_kmeans = kmeans.fit_predict(X)
```

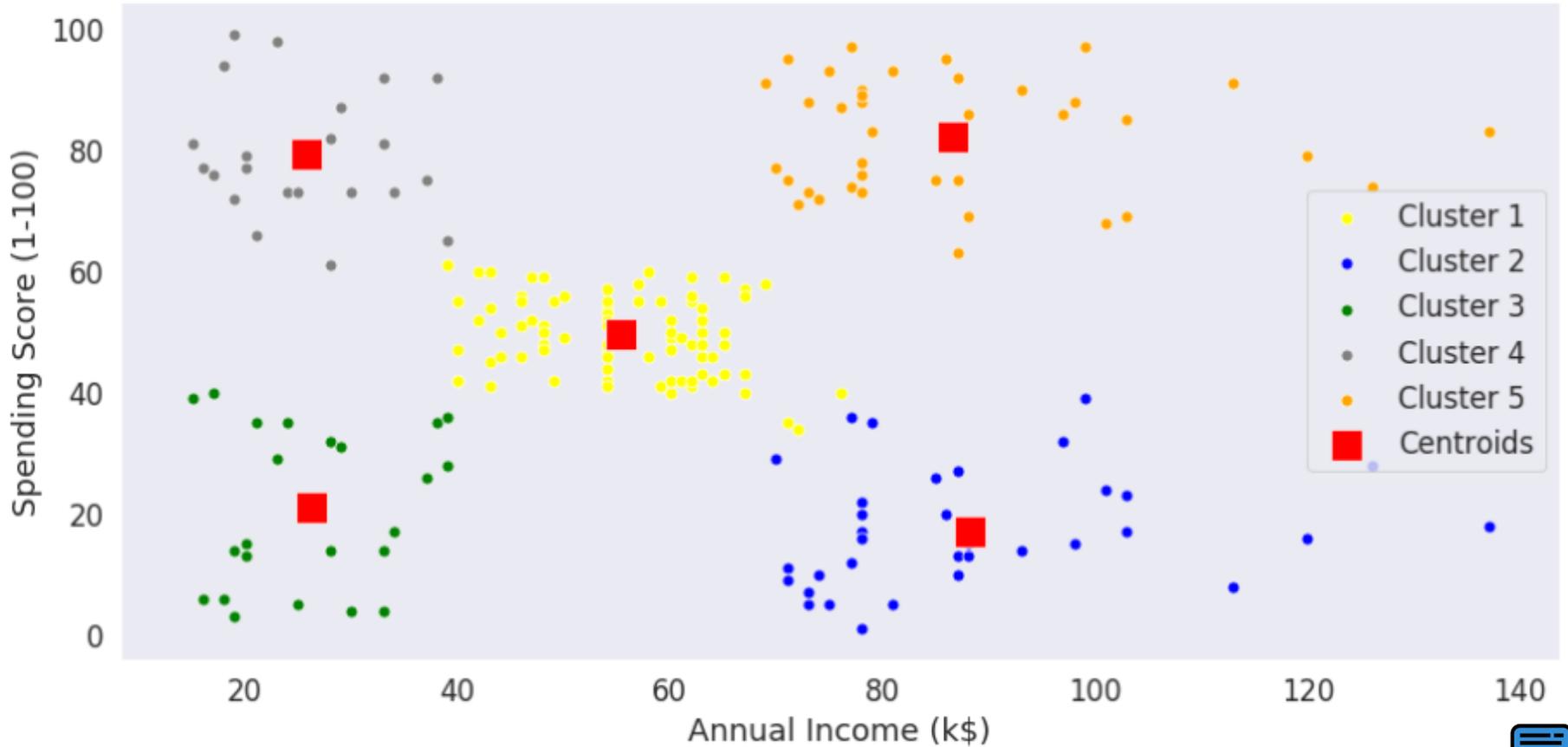


```
# Visualising the clusters
plt.figure(figsize=(15,7))
sns.scatterplot(X[y_kmeans == 0, 0], X[y_kmeans == 0, 1], color = 'yellow', label = 'Cluster 1',
s=50)
sns.scatterplot(X[y_kmeans == 1, 0], X[y_kmeans == 1, 1], color = 'blue', label = 'Cluster 2',s=
50)
sns.scatterplot(X[y_kmeans == 2, 0], X[y_kmeans == 2, 1], color = 'green', label = 'Cluster 3',s
=50)
sns.scatterplot(X[y_kmeans == 3, 0], X[y_kmeans == 3, 1], color = 'grey', label = 'Cluster 4',s=
50)
sns.scatterplot(X[y_kmeans == 4, 0], X[y_kmeans == 4, 1], color = 'orange', label = 'Cluster 5',
s=50)
sns.scatterplot(kmeans.cluster_centers_[ :, 0], kmeans.cluster_centers_[ :, 1], color = 'red',
label = 'Centroids',s=300,marker=',')
plt.grid(False)
plt.title('Clusters of customers')
plt.xlabel('Annual Income (k$)')
plt.ylabel('Spending Score (1-100)')
plt.legend()
plt.show()
```

[Step by Step KMeans Explained in Detail](#)

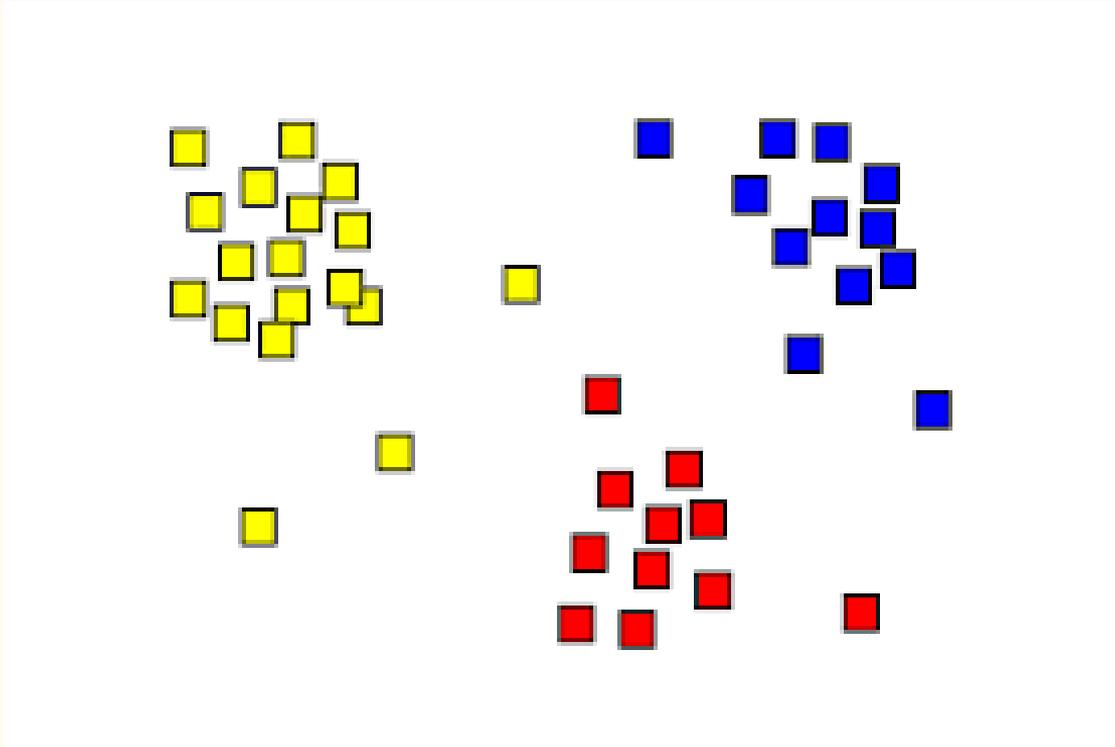


Clusters of customers



**What Can We
improve Further?**





Each Data Point is assigned to just One Cluster ie Hard Assignment.



Each Data Point is assigned to just One Cluster ie Hard Assignment.

Question : Is this the most optimum way to look at the problem?



Question : Is this the most optimum way to look at the problem?

Soft Assignments?



References

- Google Developers - [Clustering in Machine Learning](#)
- KMeans++ - [Sara Jensen](#)
- Pattern Recognition and Machine Learning- Christopher Bishop
- Mlb - MLBook - [How to Read a Model?](#)