

Un-Supervised Machine Learning

Clustering (K-Means)

Supervised Learning

- More accurate
- Labeled data required
- Requires human in the loop

Unsupervised Learning

- Less Accurate
- No labeled data required
- Minimal human effort

	<i>Supervised Learning</i>	<i>Unsupervised Learning</i>
<i>Discrete</i>	classification or categorization	clustering
<i>Continuous</i>	regression	dimensionality reduction

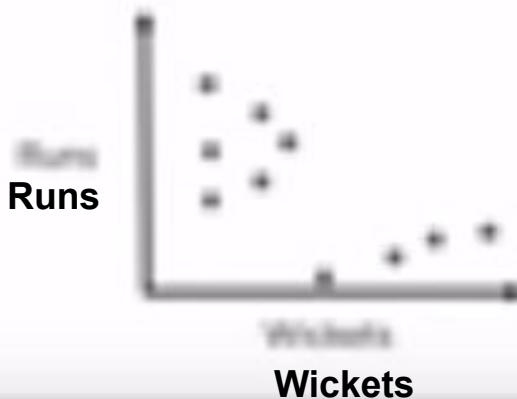


Why do we need 'Clustering'?

Assign data points

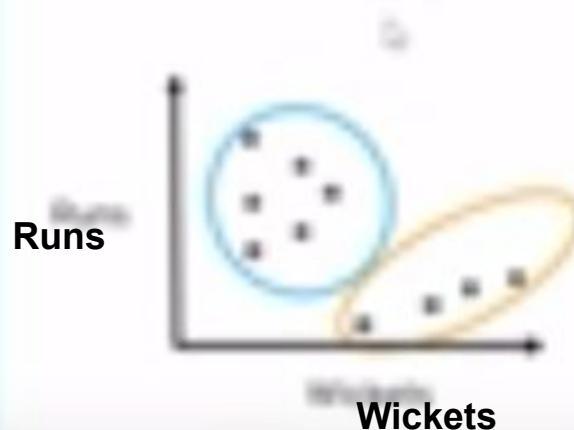
Here, we have our dataset with x and y coordinates.

Now, we want to cluster this data using K-Means.



Cluster 2

And here, we can see that this cluster has players with high wickets and low runs.



Cricket Example:
Cluster "Batsman" and "Bowlers" based on "Runs" and "Wickets" taken.

Batsman = More Runs

Bowlers = More Wickets

Why do we need clustering?

T
O
G
R
O
U
P

Unlabeled Input Dataset

Group 1

Group 2

Group 5

Group 4

Group 3

Clustering images

- For search, group as:
 - Ocean
 - Pink flower
 - Dog
 - Sunset
 - Clouds
 - ...

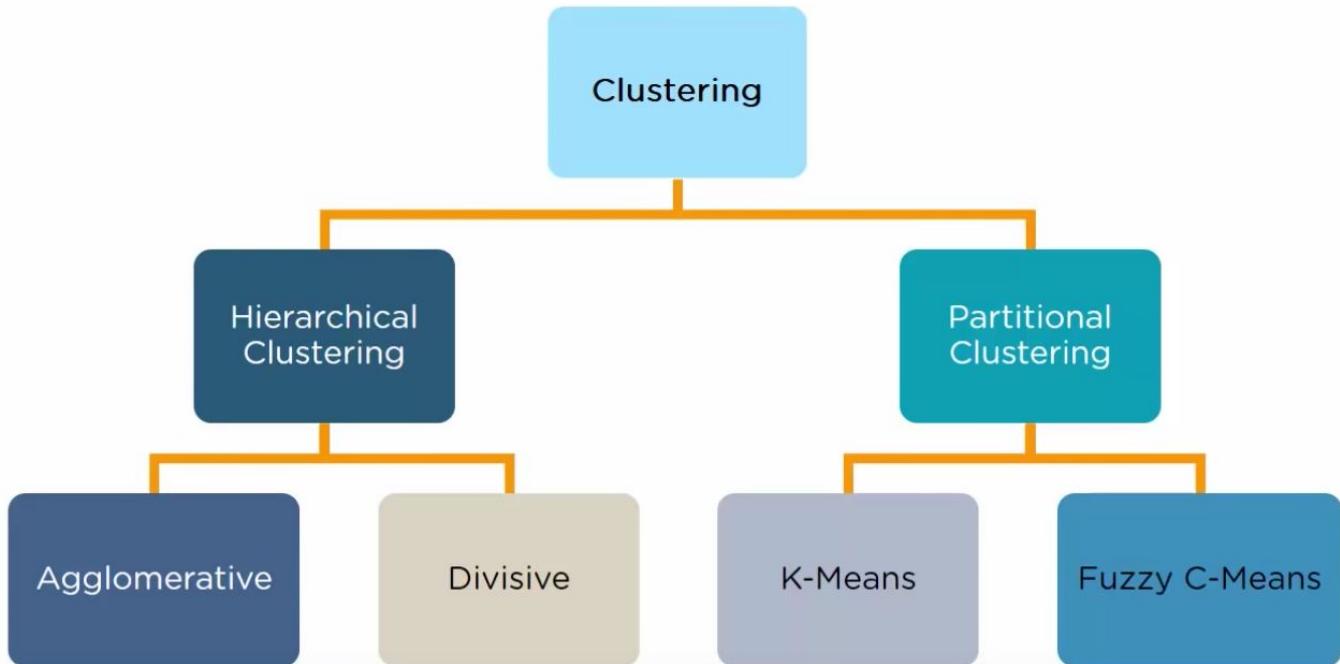




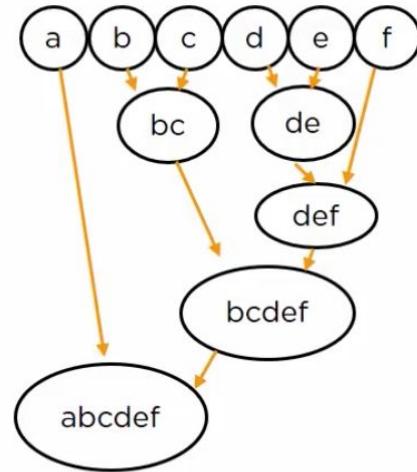
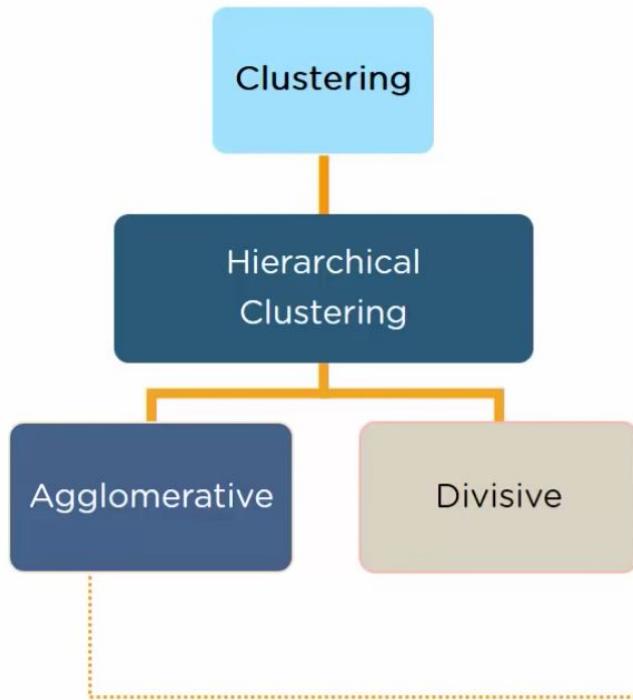
What is 'Clustering' : Details ?

Prepared By: Dr.Mydhili K Nair,Prof,ISE Dept, M S Ramaiah Institute of Technology

Types of Clustering

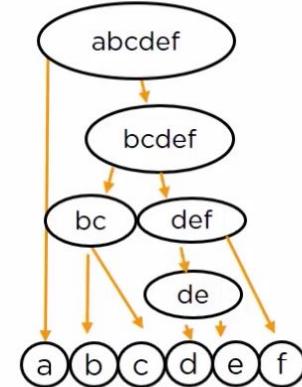
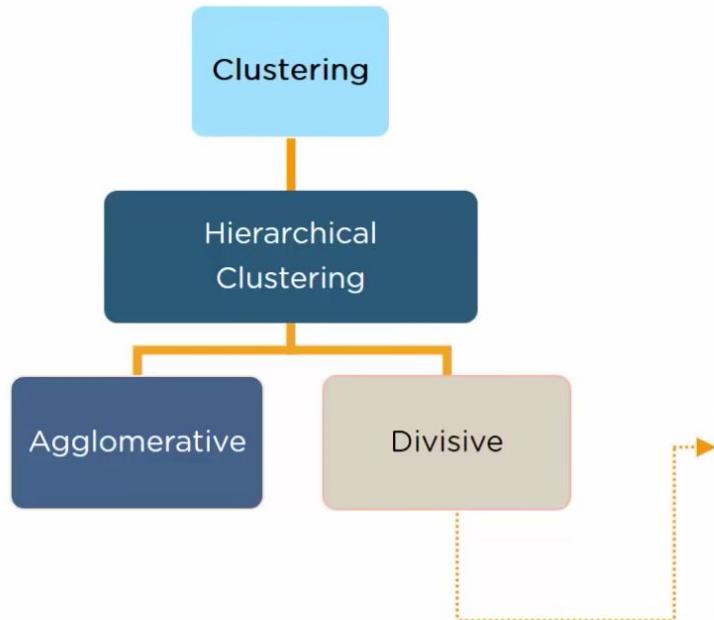


Types of Clustering



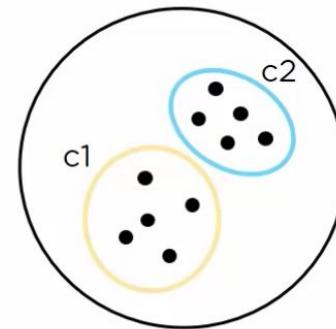
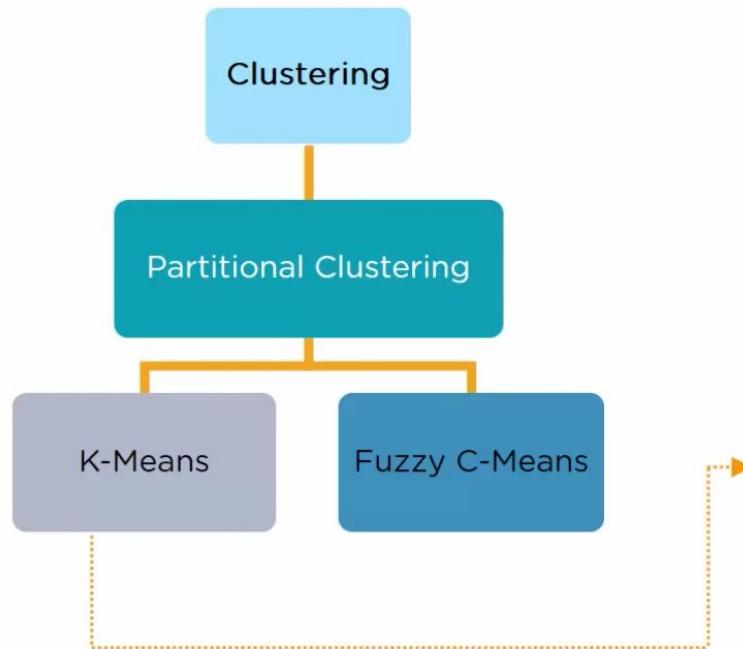
“**Bottom up**” approach: Begin with each element as a separate cluster and merge them into successively larger clusters

Types of Clustering



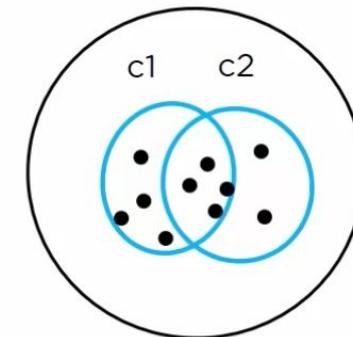
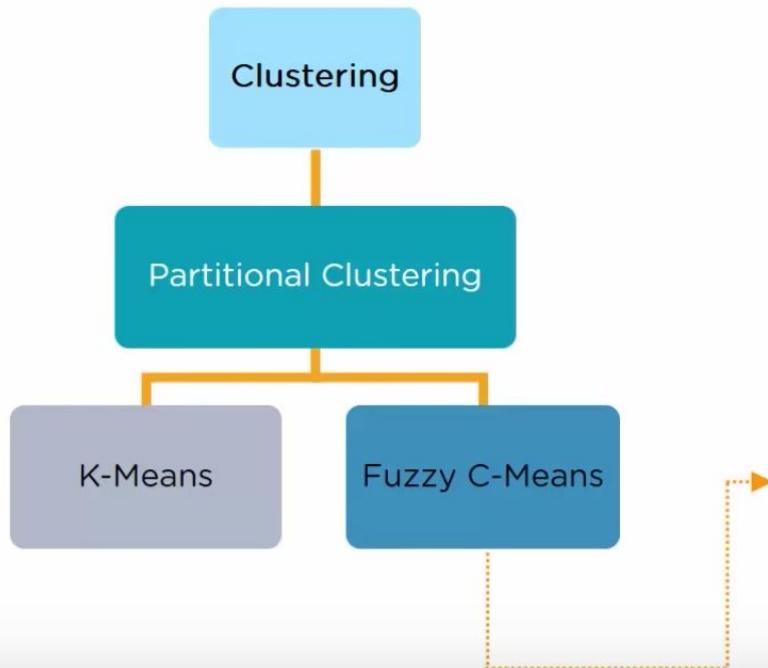
“**Top down**” approach begin with the whole set and proceed to divide it into successively smaller clusters.

Types of Clustering



Division of objects into clusters such that each object is in exactly one cluster, not several

Types of Clustering



Division of objects into clusters such that each object can belong to multiple clusters

K-Means Clustering Algorithm

In Simple
Terms....

Assuming we have inputs x_1, x_2, x_3, \dots , and value of k ,

Step 1 : Pick k random points as cluster centers called centroids

Step 2 : Assign each x_i to nearest cluster by calculating its distance to each centroid

Step 3 : Find new cluster center by taking the average of the assigned points

Step 4 : Repeat Step 2 and 3 until none of the cluster assignments change



What exactly is k-Means Clustering: the details?

Distance Measure

Euclidean
distance
measure

Manhattan
distance
measure

Distance measure will determine the similarity between two elements and it will influence the shape of the clusters

Squared Euclidean
distance measure

Cosine distance
measure

Euclidean Distance Measure

01 Euclidean distance measure

- The Euclidean distance is the "ordinary" straight line
- It is the distance between two points in Euclidean space

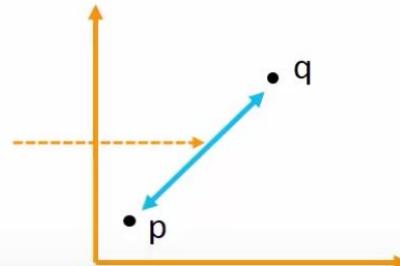
02 Squared euclidean distance measure

03 Manhattan distance measure

04 Cosine distance measure

$$d = \sqrt{\sum_{i=1}^n (q_i - p_i)^2}$$

Euclidian
Distance



Squared Euclidean Distance Measure

01

Euclidean
distance measure

The Euclidean squared distance metric uses the same equation as the Euclidean distance metric, but does not take the square root.

02

Squared euclidean
distance measure

03

Manhattan
distance measure

04

Cosine distance
measure

$$d = \sum_{i=1}^n (q_i - p_i)^2$$

Manhattan Distance Measure

01 Euclidean distance measure

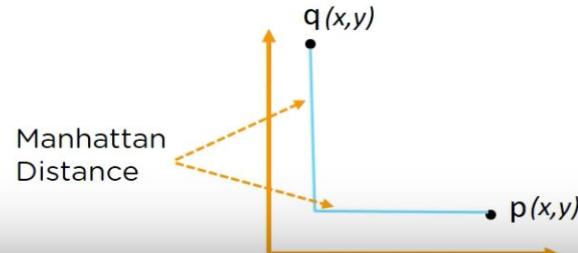
02 Squared euclidean distance measure

03 Manhattan distance measure

04 Cosine distance measure

The Manhattan distance is the simple sum of the horizontal and vertical components or the distance between two points measured along axes at right angles

$$d = \sum_{i=1}^n |q_x - p_x| + |q_y - p_y|$$



Cosine Distance Measure

01 Euclidean distance measure

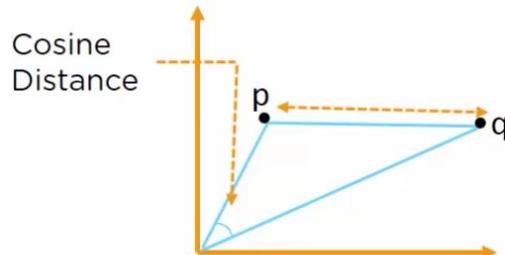
02 Squared euclidean distance measure

03 Manhattan distance measure

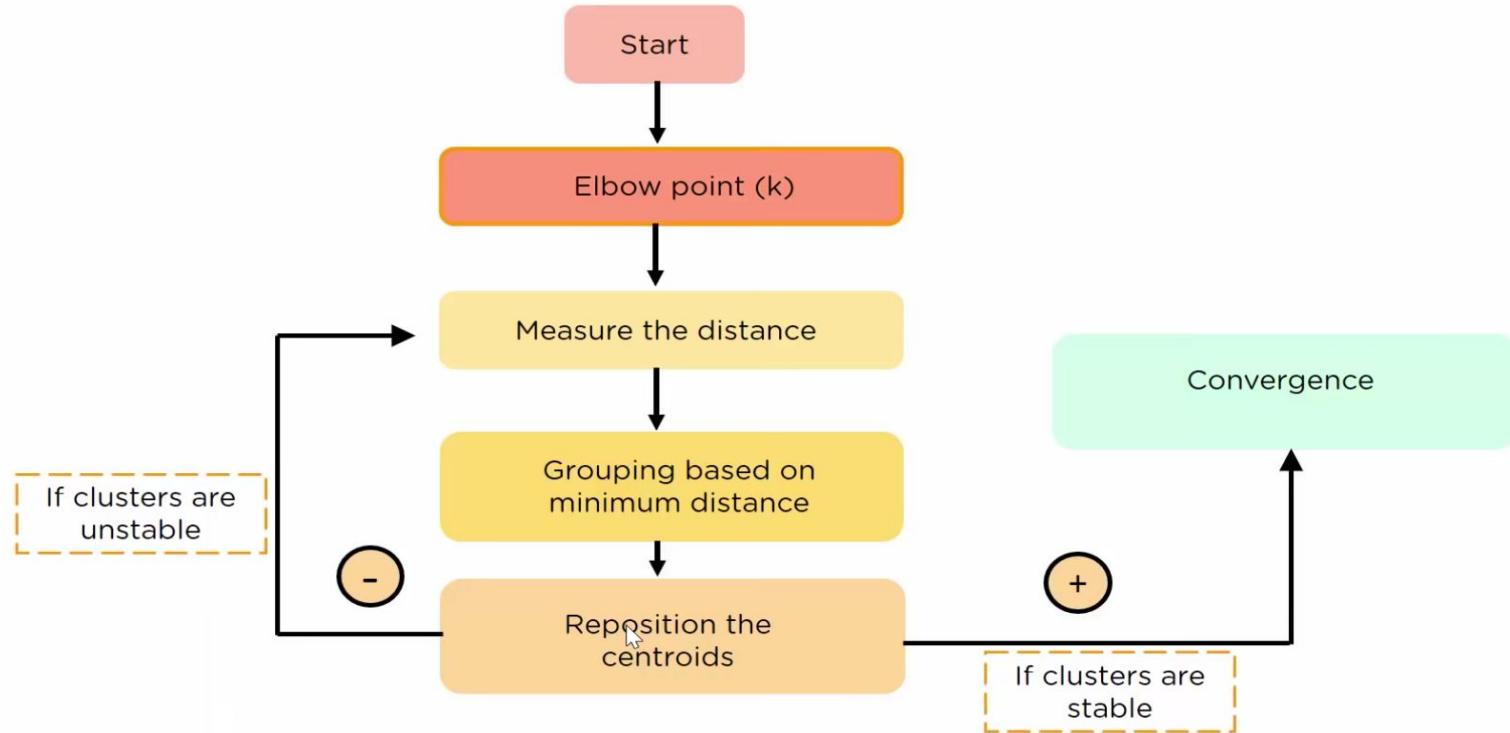
04 Cosine distance measure

The cosine distance similarity measures the angle between the two vectors

$$d = \frac{\sum_{i=0}^{n-1} q_i - p_x}{\sum_{i=0}^{n-1} (q_i)^2 \times \sum_{i=0}^{n-1} (p_i)^2}$$



How does K-Means clustering work?



How does K-Means clustering work?

Elbow point

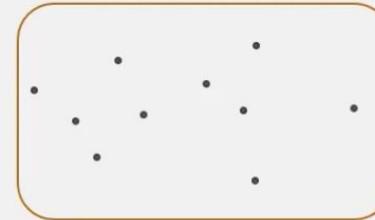
Measure the
distance

Grouping

Reposition the
centroids

Convergence

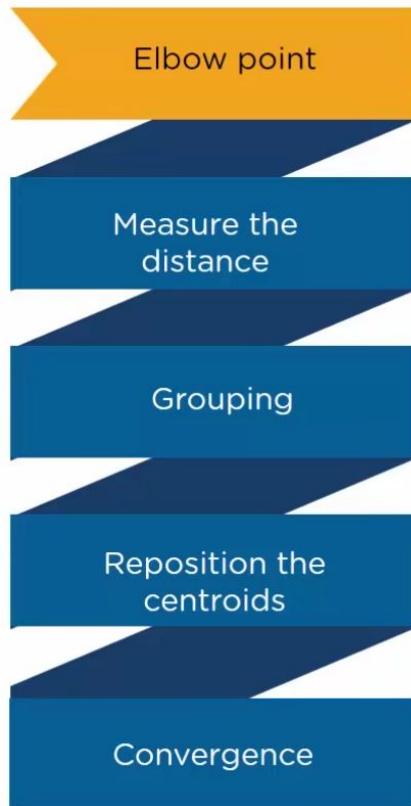
- Let's say, you have a dataset for a Grocery shop



- Now, the important question is, "*how would you choose the optimum number of clusters?*"



How does K-Means clustering work?



- The best way to do this is by **Elbow method**
- The idea of the elbow method is to run K-Means clustering on the dataset where 'k' is referred as number of clusters
- Within sum of squares (WSS) is defined as the sum of the squared distance between each member of the cluster and its centroid



$$WSS = \sum_{i=1}^m (x_i - c_i)^2$$

Where x_i = data point and c_i = closest point to centroid

How does K-Means clustering work?

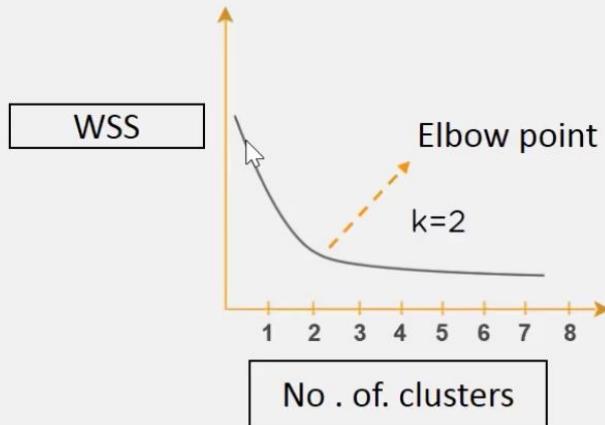
Elbow point

Measure the distance

Grouping

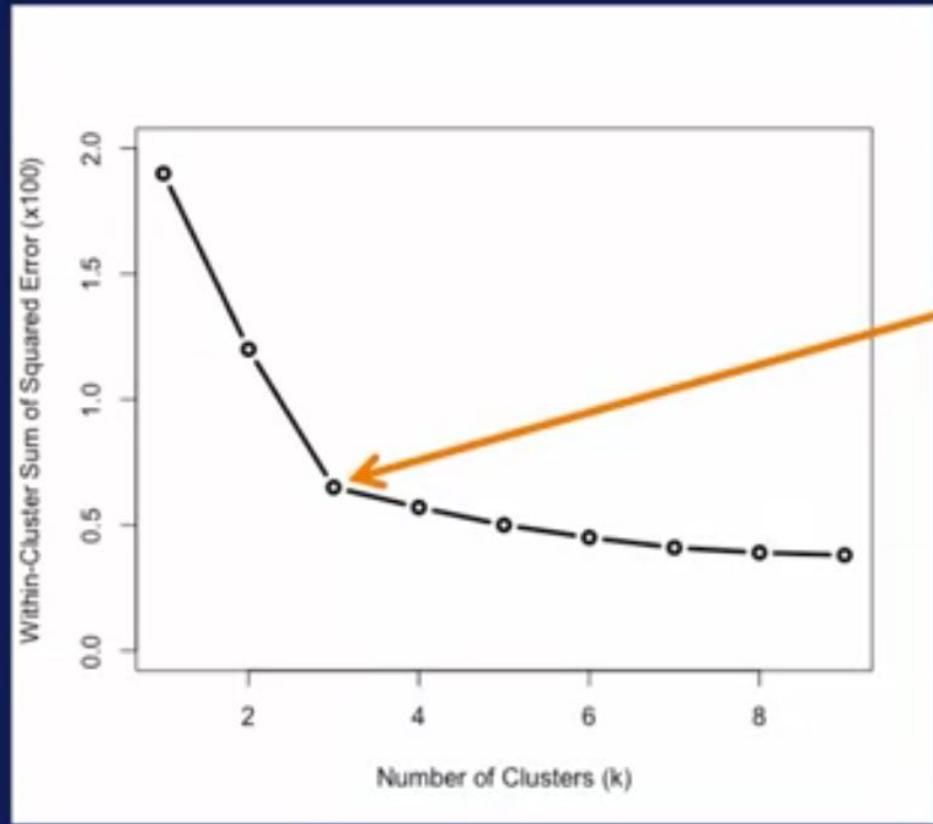
Reposition the centroids

Convergence



- Now, we draw a curve between *WSS* (within sum of squares) and the *number of clusters*
- Here, we can see a very slow change in the value of WSS after $k=2$, so you should take that elbow point value as the final number of clusters

Elbow Method for Choosing k

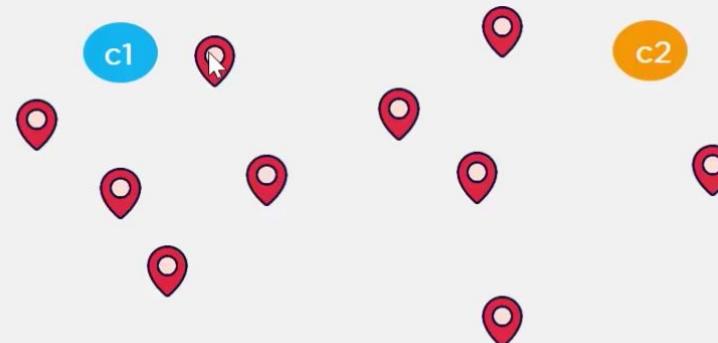


"Elbow" suggests value
for k should be 3

How does K-Means clustering work?



Step 2: We can randomly initialize two points called the cluster centroids,
Euclidean distance is a distance measure used to find out which data point
is closest to our centroids



How does K-Means clustering work?

Elbow point

Measure the distance

Grouping

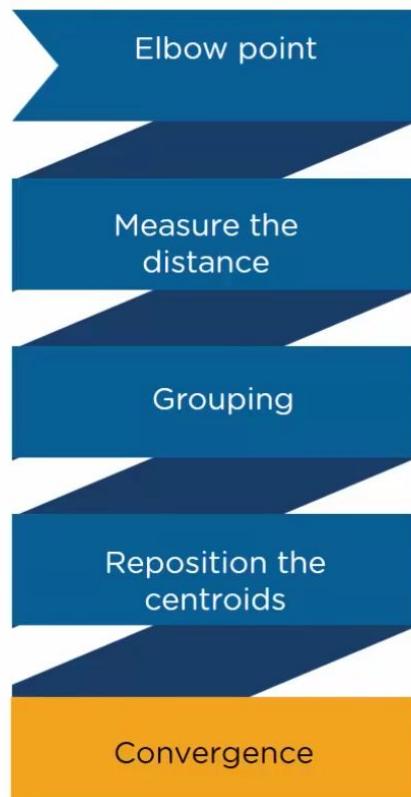
Reposition the centroids

Convergence

Step 3: Based upon the distance from c1 and c2 centroids, the data points will group itself into clusters



How does K-Means clustering work?



Step 8: Once the clusters become static, K-Means clustering algorithm is said to be converged



How does K-Means clustering work?

Elbow point

Measure the distance

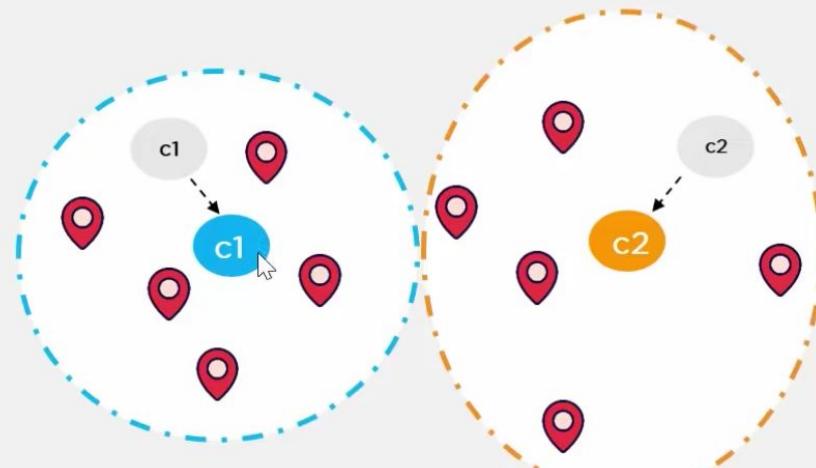
Grouping

Reposition the centroids

Convergence

Step 6: Now, compute the centroid of data points inside the orange cluster

Step 7: Reposition the centroid of the orange cluster to the new centroid





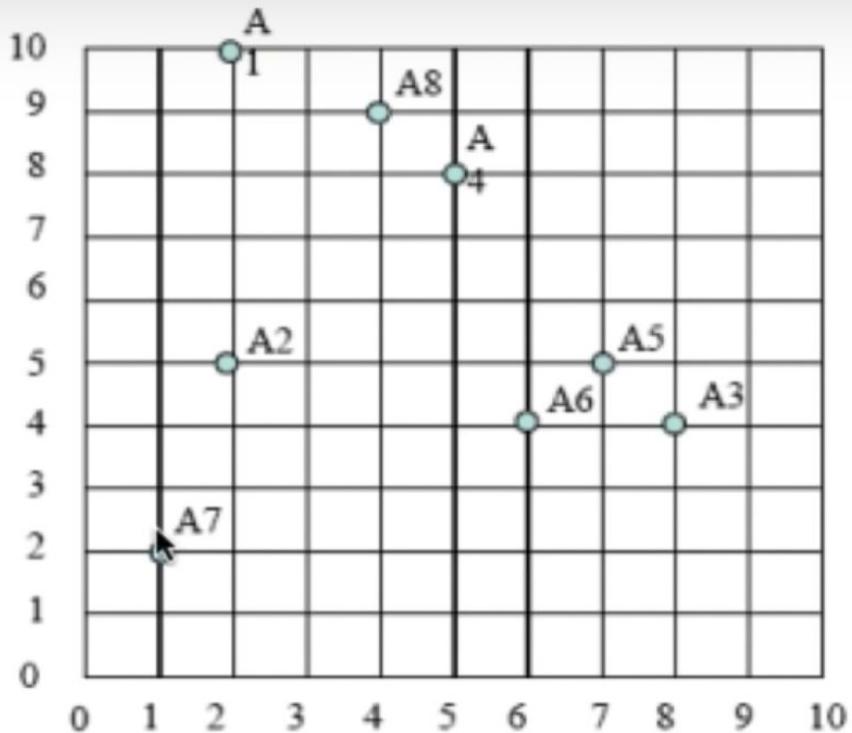
How K-Means Clustering work?

Prepared By: Dr.Mydhili K Nair,Prof,ISE Dept, M S Ramaiah Institute of Technology

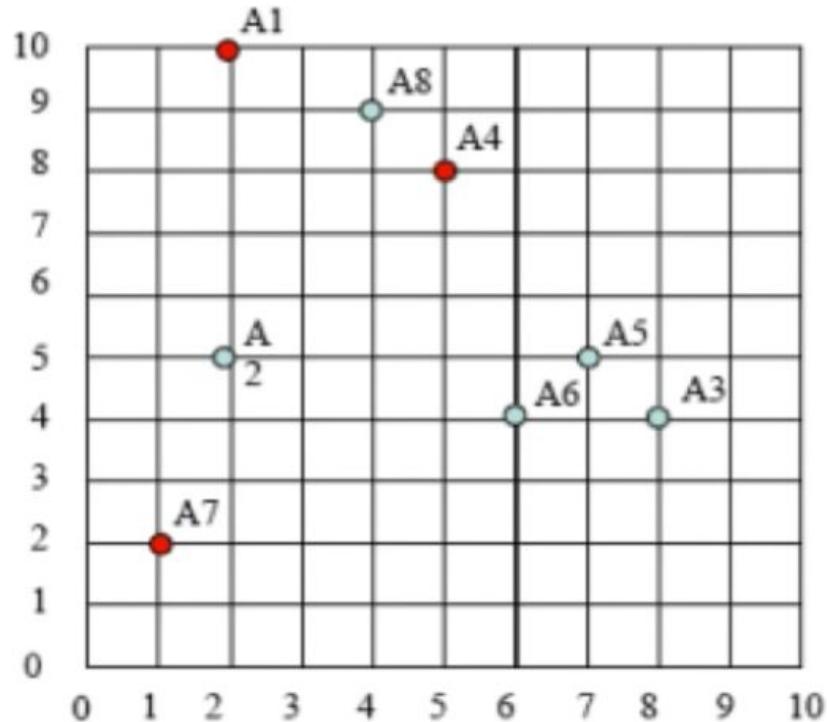
- **Example**

Cluster the following eight points (with (x, y) representing locations) into three clusters A1(2, 10) A2(2, 5) A3(8, 4) A4(5, 8) A5(7, 5) A6(6, 4) A7(1, 2) A8(4, 9).

Initial cluster centers are: A1(2, 10), A4(5, 8) and A7(1, 2).



**Initial Observations -
Dataset Points**



**Randomly selected -
'Centroid' Points**

- **Measuring similarity between observations**
- **Rectilinear Distance:** Most common method to measure distance between observations, when observations include continuous variables is the Rectilinear distance.
- Let observations $u = (u_1, u_2, \dots, u_q)$ and $v = (v_1, v_2, \dots, v_q)$ each comprise measurements of q variables.
- The Rectilinear distance between observations u and v is
- $d_{u,v} = |u_1 - v_1| + |u_2 - v_2| + \dots + |u_q - v_q|$



‘Rectilinear Distance’ is commonly called - “Manhattan Distance”

point
x_l, y_l
(2, 10)

mean1
x₂, y₂
(2, 10)

$$\rho(a, b) = |x_2 - x_l| + |y_2 - y_l|$$

$$\begin{aligned}\rho(\text{point}, \text{mean1}) &= |x_2 - x_l| + |y_2 - y_l| \\ &= |2 - 2| + |10 - 10| \\ &= 0 + 0 \\ &= 0\end{aligned}$$

point
x_l, y_l
(2, 10)

mean2
x₂, y₂
(5, 8)

$$\rho(a, b) = |x_2 - x_l| + |y_2 - y_l|$$

$$\begin{aligned}\rho(\text{point}, \text{mean2}) &= |x_2 - x_l| + |y_2 - y_l| \\ &= |5 - 2| + |8 - 10| \\ &= 3 + 2 \\ &= 5.\end{aligned}$$

Similarity Measure through : Rectilinear Distance / Manhattan Distance

1. Find the centroids
2. Calculate the distance of each data point with the centroid
3. Assign the data point to the cluster of the centroid with the least distance value

		(2, 10)	(5, 8)	(1, 2)	
	Point	Dist Mean 1	Dist Mean 2	Dist Mean 3	Cluster
A1	(2, 10)	0	5	9	1
A2	(2, 5)				
A3	(8, 4)				
A4	(5, 8)				
A5	(7, 5)				
A6	(6, 4)				
A7	(1, 2)				
A8	(4, 9)				

4. If the least distance measure has the same value the data point is allocated to **any one** of the centroid clusters.
5. A data point belongs to only one cluster in k-means

Iteration 1

		(2, 10)	(5, 8)	(1, 2)	
	Point	Dist Mean 1	Dist Mean 2	Dist Mean 3	Cluster
A1	(2, 10)	0	5	9	1
A2	(2, 5)	5	6	4	3
A3	(8, 4)	12	7	9	2
A4	(5, 8)	5	0	10	2
A5	(7, 5)	10	5	9	2
A6	(6, 4)	10	5	7	2
A7	(1, 2)	9	10	0	3
A8	(4, 9)	3	2	10	2

Cluster 1

(2, 10)

Cluster 2

(8, 4)

Cluster 3

(2, 5)

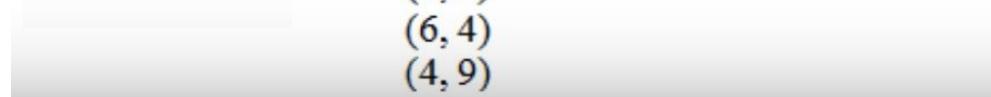
(5, 8)

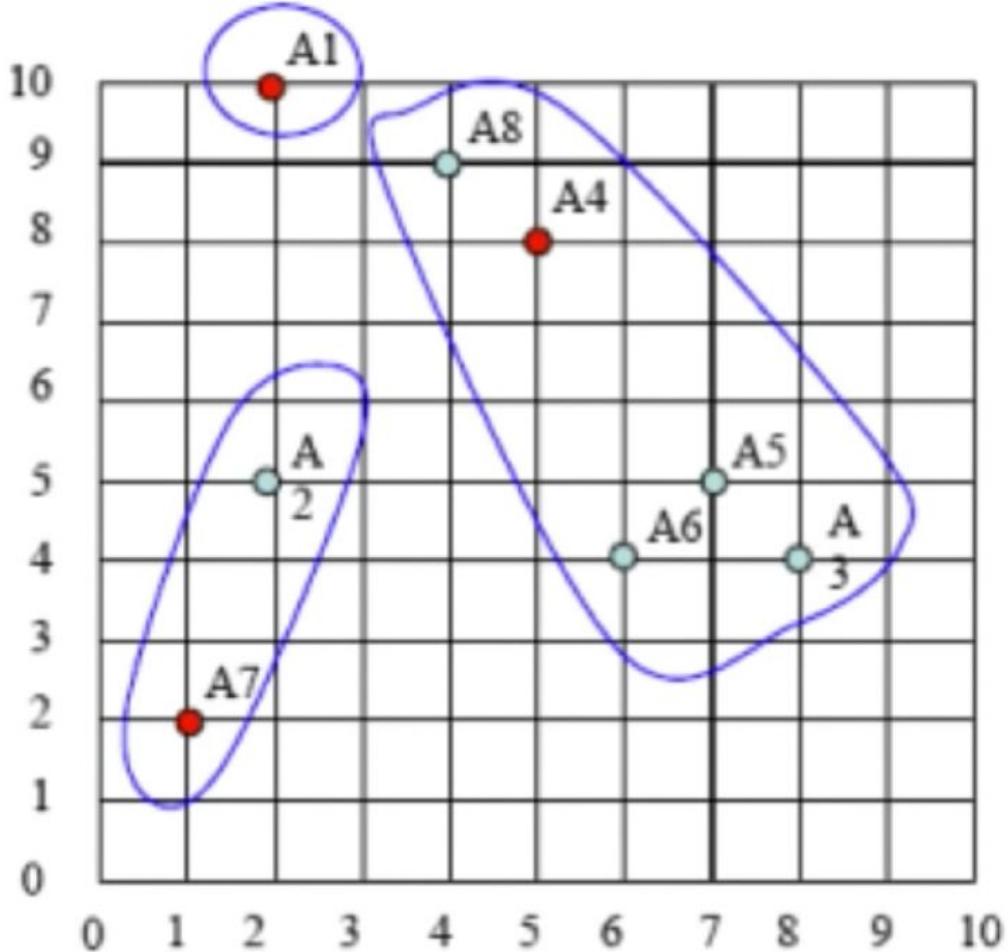
(1, 2)

(7, 5)

(6, 4)

(4, 9)



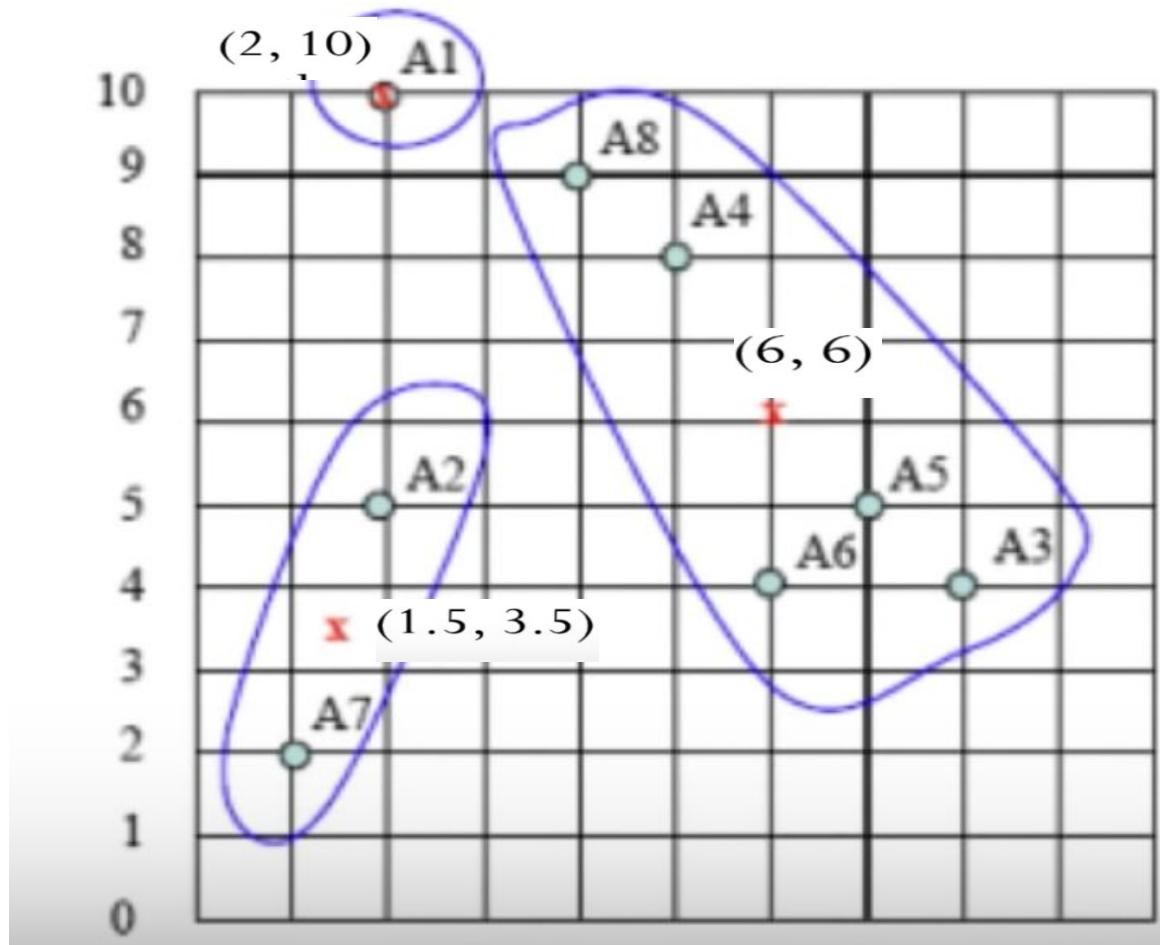


*After
first
iteration*

Second Iteration

Cluster 1	Cluster 2	Cluster 3
(2, 10)	(8, 4)	(2, 5)
	(5, 8)	(1, 2)
	(7, 5)	
	(6, 4)	
	(4, 9)	

- Next, we need to re-compute the new cluster centers (means). We do so, by taking the mean of all points in each cluster.
- For Cluster 1, we only have one point A1(2, 10), which was the old mean, so the cluster center remains the same.
- For Cluster 2, we have $((8+5+7+6+4)/5, (4+8+5+4+9)/5) = (6, 6)$
- For Cluster 3, we have $((2+1)/2, (5+2)/2) = (1.5, 3.5)$



*Second
Iteration:
Centroid
recalculation*

Recalculated Centroids

		(2, 10)	(6, 6)	(1.5, 3.5)	
	Point	Dist Mean 1	Dist Mean 2	Dist Mean 3	Cluster
A1	(2, 10)	0	8	7	1
A2	(2, 5)	5	5	2	3
A3	(8, 4)	12	4	7	2
A4	(5, 8)	5	3	8	2
A5	(7, 5)	10	2	7	2
A6	(6, 4)	10	2	5	2
A7	(1, 2)	9	9	2	3
A8	(4, 9)	3	5	8	1

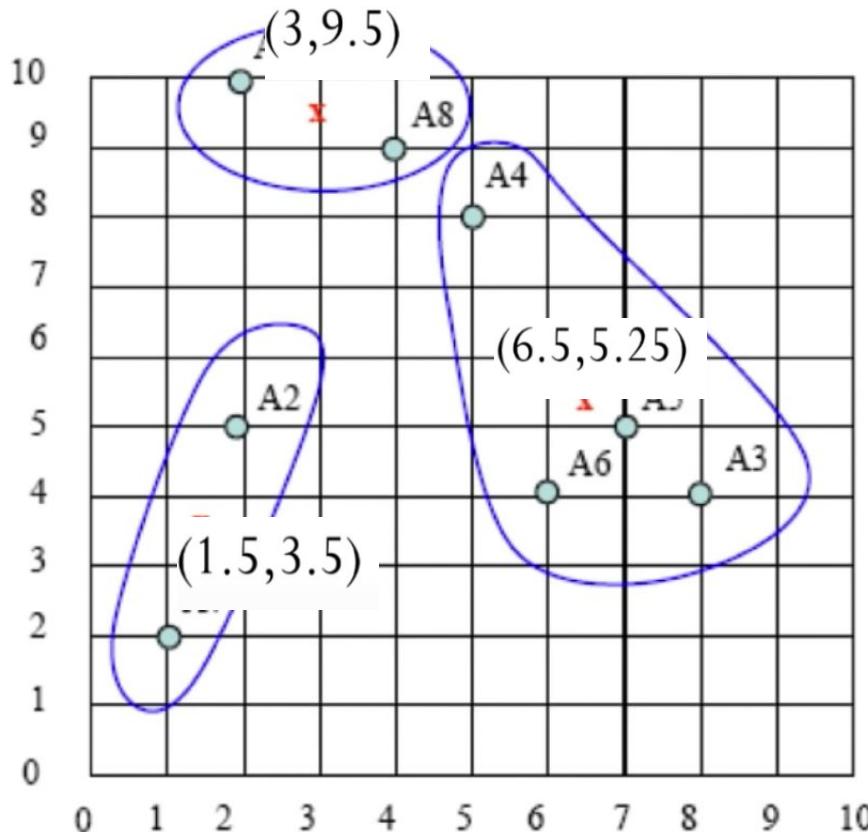
After Second iteration - Manhattan Distances

- Next, we need to re-compute the new cluster centers (means). We do so, by taking the mean of all points in each cluster.

Third Iteration

- In Cluster 1, we have points 1 and 8. Therefore the centroid is: $((2+4)/2, (10+9)/2) = (3, 9.5)$
- In Cluster 2, we have points 3, 4, 5 and 6. Therefore, the centroid is: $((8+5+7+6)/4, (4+8+5+4)/4) = (6.5, 5.25)$
- For Cluster 3, we have points 2 and 7. Therefore, the centroid is: $((2+1)/2, (5+2)/2) = (1.5, 3.5)$

*Third
iteration:
Centroid
Recalculation*



Recalculated Centroids

		(3, 9.5)	(6.5 ,5.25)	(1.5, 3.5)	
	Point	Dist Mean 1	Dist Mean 2	Dist Mean 3	Cluster
A1	(2, 10)	1.5	9.25	7	1
A2	(2, 5)	5.5	4.75	2	3
A3	(8, 4)	10.5	2.75	7	2
A4	(5, 8)	3.5	4.25	8	1
A5	(7, 5)	8.5	0.75	7	2
A6	(6, 4)	8.5	1.75	5	2
A7	(1, 2)	9.5	8.75	2	3
A8	(4, 9)	1.5	6.25	8	1

After Third iteration

- Next, we need to re-compute the new cluster centers (means). We do so, by taking the mean of all points in each cluster.

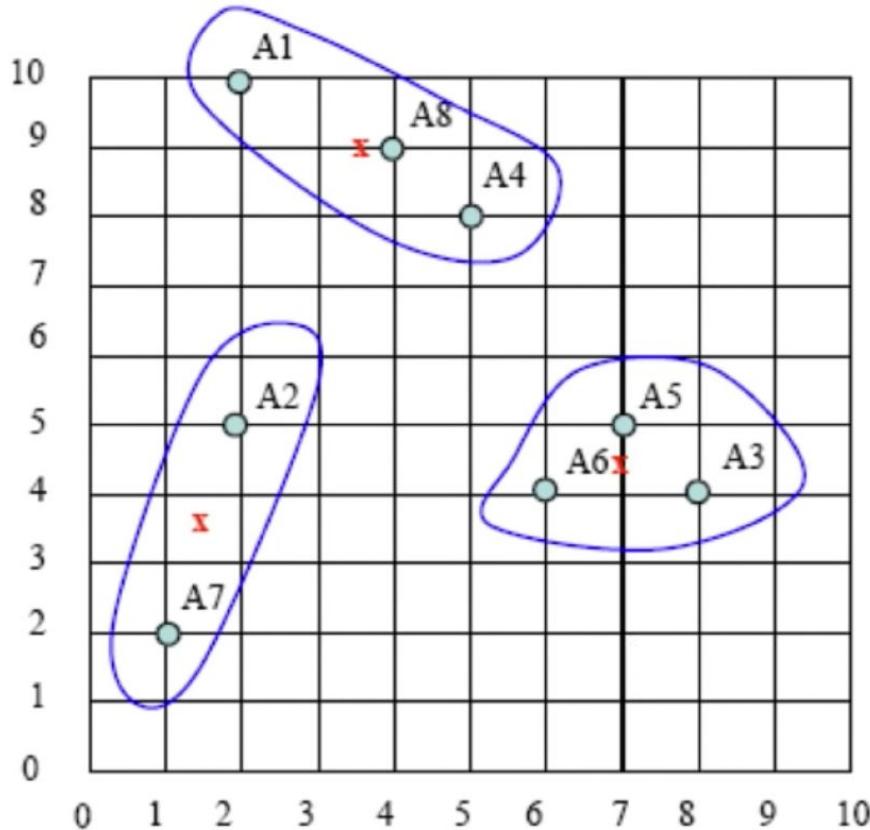
Fourth Iteration

- In Cluster 1, we have points 1, 4, and 8. Therefore the centroid is: $((2+5+4)/2,(10+8+9)/2)=(3.67,9)$
- In Cluster 2, we have points 3,5 and 6. Therefore, the centroid is: $((8+7+6)/4,(4+5+4)/4)=(7,4.3)$
- For Cluster 3, we have points 2 and 7. Therefore, the centroid is: $((2+1)/2,(5+2)/2)=(1.5,3.5)$

Recalculated Centroids

		(3.67, 9)	(7 ,4.3)	(1.5, 3.5)	
	Point	Dist Mean 1	Dist Mean 2	Dist Mean 3	Cluster
A1	(2, 10)	2.67	10.7	7	1
A2	(2, 5)	5.67	5.7	2	3
A3	(8, 4)	9.33	1.3	7	2
A4	(5, 8)	2.33	5.7	8	1
A5	(7, 5)	7.33	0.7	7	2
A6	(6, 4)	7.33	1.3	5	2
A7	(1, 2)	9.67	8.3	2	3
A8	(4, 9)	0.33	7.7	8	1

After Fourth iteration



*After
Fourth
iteration*

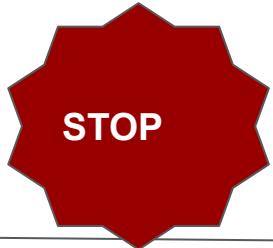
		(3, 9.5)	(6.5 ,5.25)	(1.5, 3.5)	
	Point	Dist Mean 1	Dist Mean 2	Dist Mean 3	Cluster
A1	(2, 10)	1.5	9.25	7	1
A2	(2, 5)	5.5	4.75	2	3
A3	(8, 4)	10.5	2.75	7	2
A4	(5, 8)	3.5	4.25	8	1
A5	(7, 5)	8.5	0.75	7	2
A6	(6, 4)	8.5	1.75	5	2
A7	(1, 2)	9.5	8.75	2	3
A8	(4, 9)	1.5	6.25	8	1

After Third iteration

Same Points in Same Cluster - Only Slight Change in Centroid

		(3.67, 9)	(7 ,4.3)	(1.5, 3.5)	
	Point	Dist Mean 1	Dist Mean 2	Dist Mean 3	Cluster
A1	(2, 10)	2.67	10.7	7	1
A2	(2, 5)	5.67	5.7	2	3
A3	(8, 4)	9.33	1.3	7	2
A4	(5, 8)	2.33	5.7	8	1
A5	(7, 5)	7.33	0.7	7	2
A6	(6, 4)	7.33	1.3	5	2
A7	(1, 2)	9.67	8.3	2	3
A8	(4, 9)	0.33	7.7	8	1

After Fourth iteration



STOP



When to Stop Execution of k-means?

(Stopping Criterion for Convergence)

1. The datapoints assigned to specific cluster remain the same (takes too much time)
To counter the time-consumption issue, usually k-means stops when the cluster points remain in the same cluster for a set threshold number of iterations(e.g.2)
2. Centroids remain the same (time consuming)
This is the most optimal method because it reaches a global solution.
3. The distance of datapoints from their centroid is minimum (the threshold you've set)
This stopping criteria is very specific to the dataset. It is not possible to generalize a threshold value for every dataset.
4. Fixed number of iterations have reached (insufficient iterations → poor results, choose max iteration wisely)
This is a brute force method, results is fast convergence and works in specific cases. Most unreliable and inconsistent, though.

The k -Means Algorithm

- **Initialisation**

- choose a value for k
- choose k random positions in the input space
- assign the cluster centres μ_j to those positions

- **Learning**

- repeat

- * for each datapoint \mathbf{x}_i :

- compute the distance to each cluster centre
 - assign the datapoint to the nearest cluster centre with distance

$$d_i = \min_j d(\mathbf{x}_i, \mu_j). \quad (14.1)$$

- * for each cluster centre:

- move the position of the centre to the mean of the points in that cluster (N_j is the number of points in cluster j):

$$\mu_j = \frac{1}{N_j} \sum_{i=1}^{N_j} \mathbf{x}_i \quad (14.2)$$

- until the cluster centres stop moving

- **Usage**

- for each test point:

- * compute the distance to each cluster centre
 - * assign the datapoint to the nearest cluster centre with distance

$$d_i = \min_j d(\mathbf{x}_i, \mu_j). \quad (14.3)$$

From
Stephan
Marsland

Unsolved Numerical of K-Means.

Question:

Cluster the dataset = { 2,3,4,10,11,12, 20, 25,30 } using k-means algorithm. We need to group into two clusters. Assume the initial centroids as 2 and 12.

Answer to unsolved Numerical of K-Means.

Question:

Cluster the dataset = { 2,3,4,10,11,12, 20, 25,30 } using k-means algorithm. We need to group into two clusters. Assume the initial centroids as 2 and 12.

Answer:

First Cluster: $k_1 = \{ 2,3,4,10,11,12 \}$

First Mean/Centroid: $m_1 = 7$

Second Cluster: $k_2 = \{ 20, 25,30 \}$

Second Mean/Centroid: $m_2 = 25$

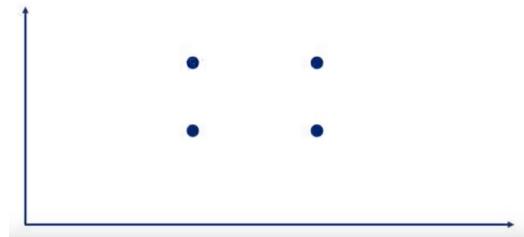
Advantages of K-Means:

1. Running Time
2. Better for high dimensional data.
3. Easy to interpret and Implement.

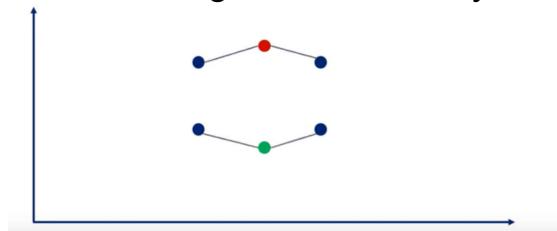
Problems with K-Means Clustering

Problem #1 : K-Means is sensitive to data initialization

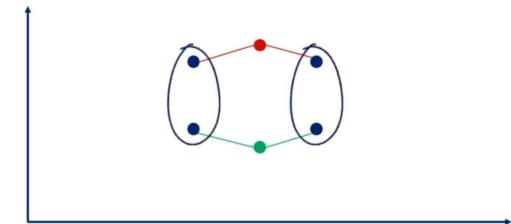
Initial Set of data points



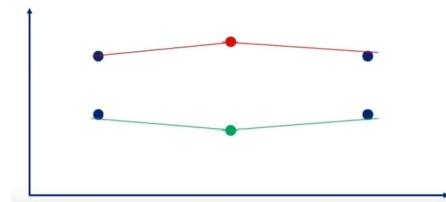
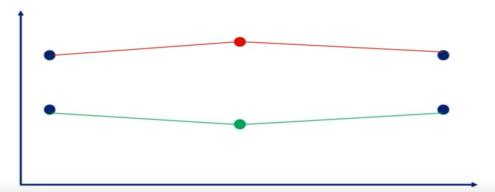
Red & **Green** are the initial centroids chosen for these data points. The centroids are aligned horizontally.



Centroids could have been chosen vertically too - resulting in the clusters shown.



- This problem continues when the data points are more widely spread as shown below.
- This results in very useless clusters if the natural grouping was vertically all along.
- **Inference:** Choosing the centroid during data initialization is a very important step in K-means. If its inappropriately chosen the entire K-Means clustering will be useless.



Hard Assignment: We are certain that particular points belong to particular centroid, when the centroid itself could be inappropriately chosen.

Solution : K-Means ++

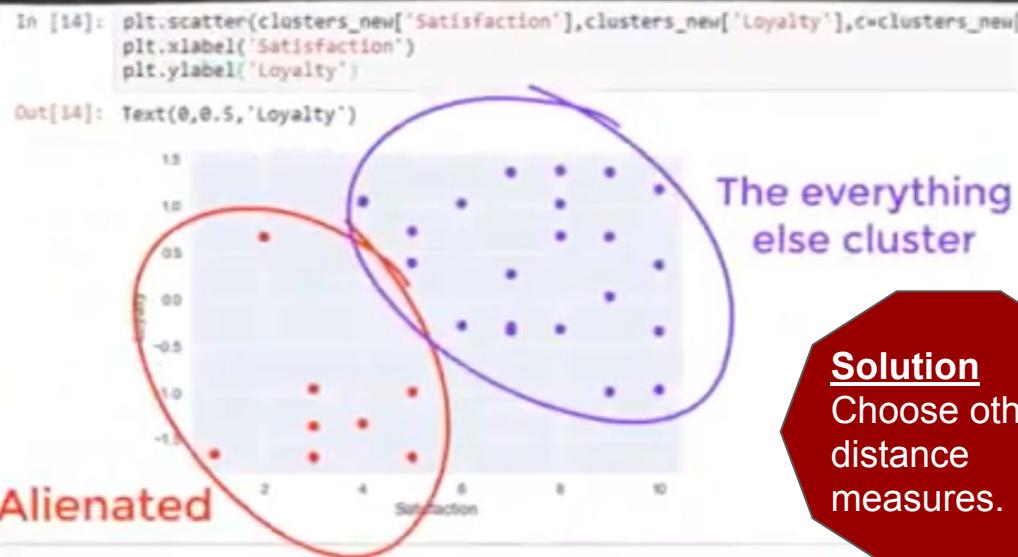
```
In [14]: kmeans.fit(x)
```

```
Out[14]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,  
n_clusters=3, n_init=10, n_jobs=1, precompute_distances='auto',  
random_state=None, tol=0.0001, verbose=0)
```

- Data is pre-processed using various techniques during the initialization phase.
- In Python's sklearn 'K-Means ++' library is preferable over plain k-means

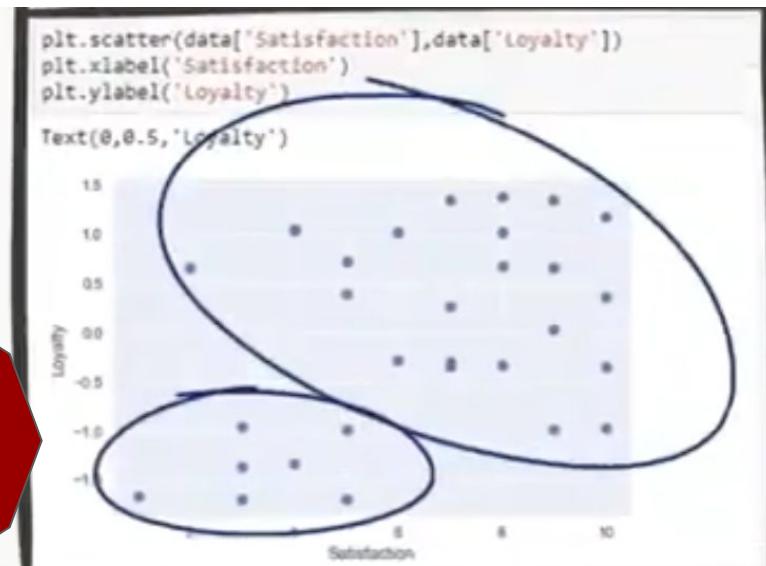
Problem #2 : Default K-Means produces ‘circular’ clusters

- Most popular/default distance measure from centroid in K-Means is ‘Euclidean Distance’ measure.
 - This leads to ‘circular’ clusters
 - The other data points become outliers.
- The natural shape of the clusters commonly is not circular.
 - It could be elipsoidal as shown below.



The everything
else cluster

Solution
Choose other
distance
measures.

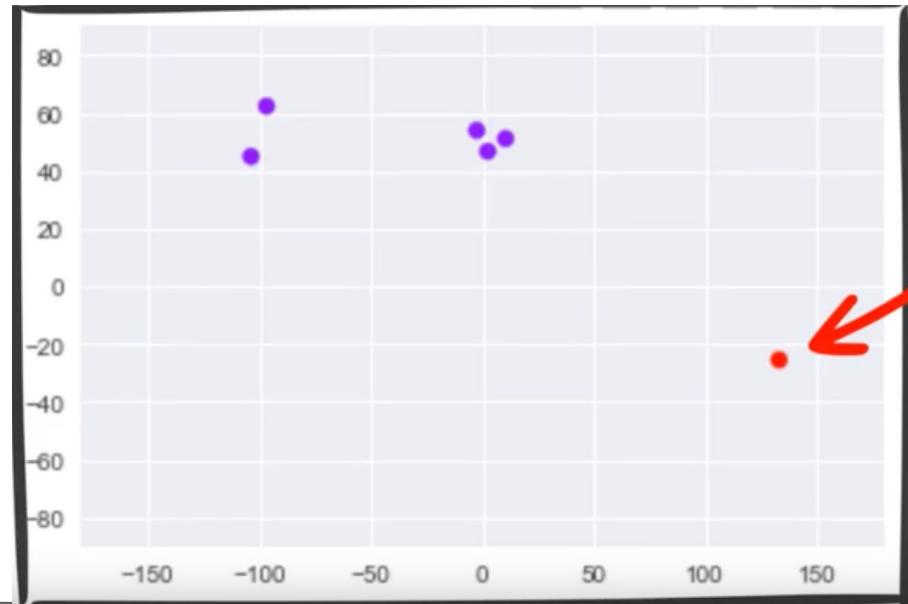


Problem #3 : K-Means is sensitive to Outliers

- In a clustering of countries of the world 'Australia' is always a lonely point , an 'outlier' as shown below, as it is far away from the other data points cluster centroids.
- But, K-means will create single-point clusters too.

Solution:

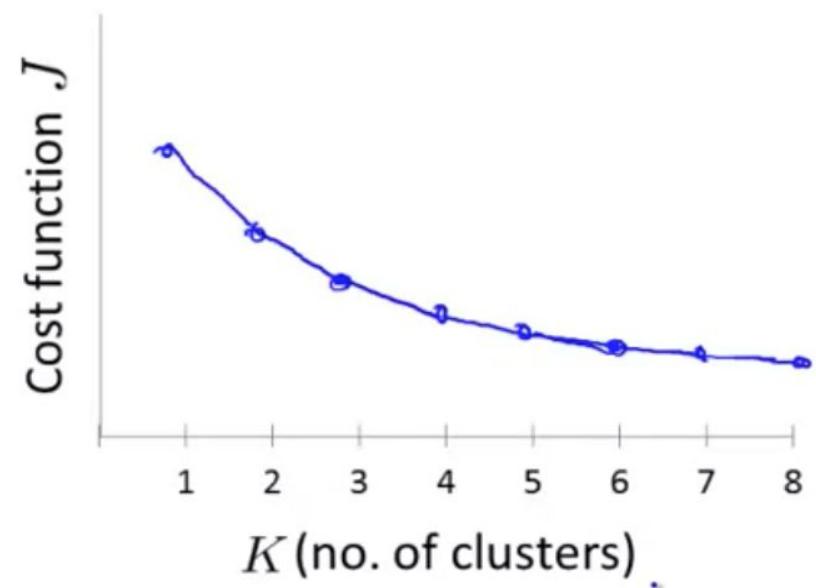
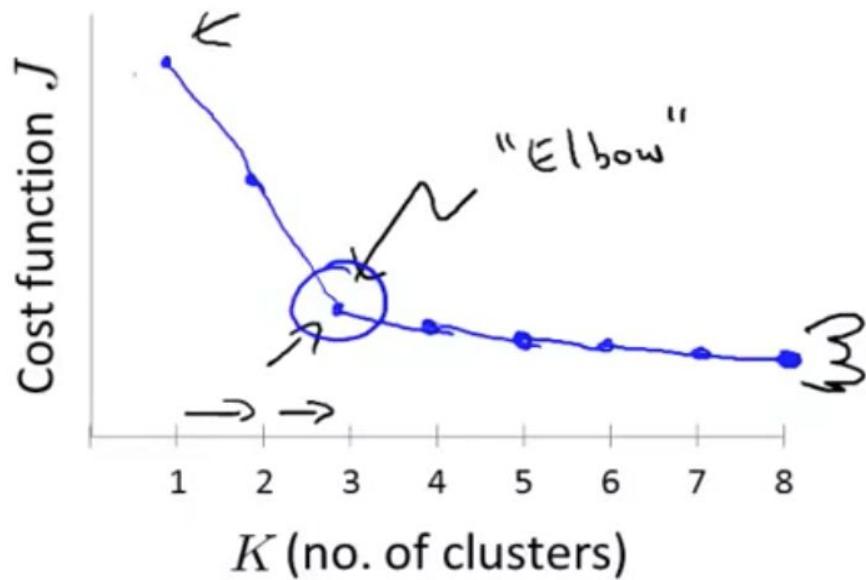
Remove Outliers. Not a very good solution, as these data points may mean something. Information is lost.



Problem #4 : K-Means - Choosing right value of 'K'

Elbow method: Not a very scientific method.

Difficult to determine 'K'



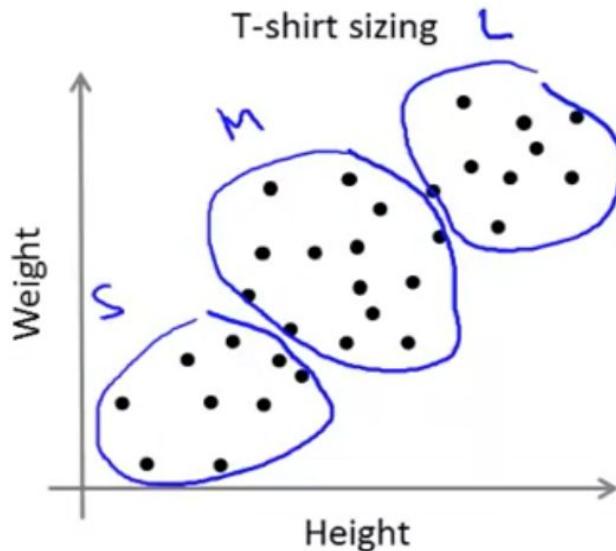
Solution : Trial and Error - Not Scientific, Time Consuming

Choosing the value of K

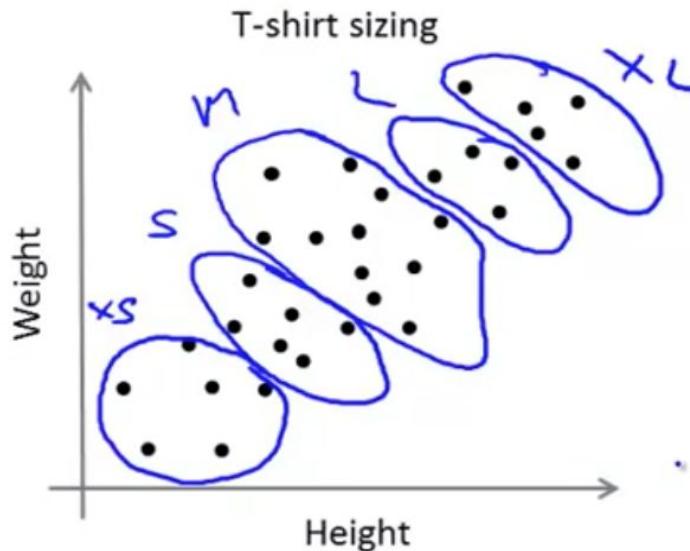
Sometimes, you're running K-means to get clusters to use for some later/downstream purpose. Evaluate K-means based on a metric for how well it performs for that later purpose.

$$K=3 \quad S, M, L$$

E.g.



$$K=5 \quad XS, S, M, L, XL$$





Where do we need 'Clustering'?

Structuring web search results

- Search terms can have multiple meanings
- Example: “**cardinal**”



Cardinal : A Bird



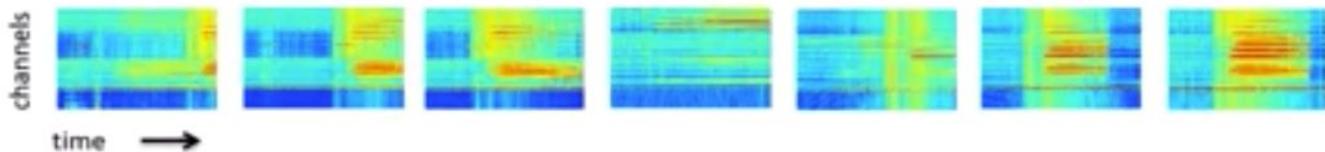
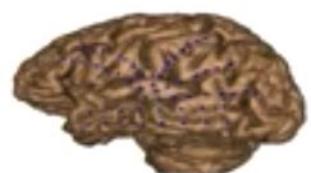
Cardinal : Baseball Team



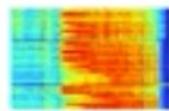
Cardinal : A Church Priest



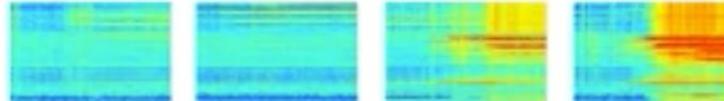
Example: Patients and seizures are diverse



Intracranial EEG Recordings

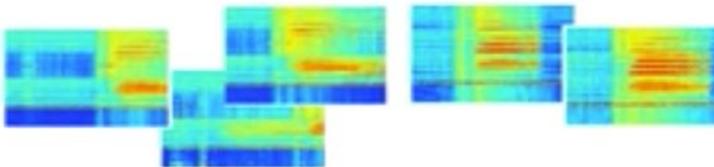


Can we group patients based on exhibited medical conditions?

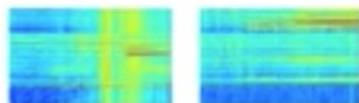


Cluster seizures by observed time courses

Four Clusters
of EEG Seizure
Images



- Types of seizures
- Properties that vary between types
- Helps study a sub-cluster in depth to analyse the disease



Products on Amazon

- Discover product categories from purchase histories



“furniture”



Car Seat

Instead of looking at each product, group “similar” purchases

Products on Amazon

- Discover product categories from purchase histories



~~furniture~~
"baby"



- Or discovering groups of **users**

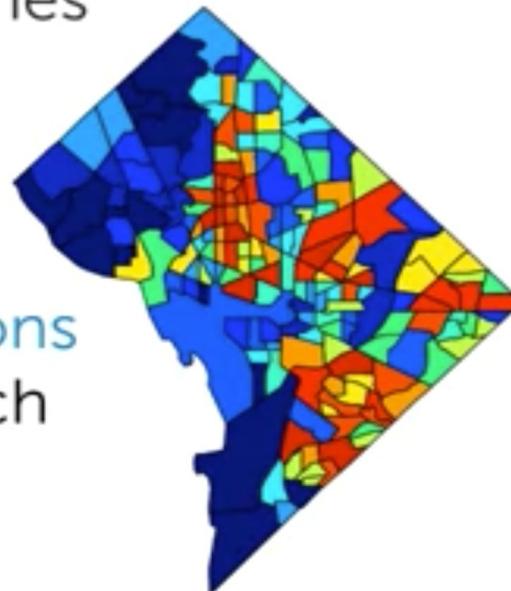
Discovering similar neighborhoods

- **Task 1:** Estimate price at a small regional level
- **Challenge:**
 - Only a few (or no!) sales in each region per month
- **Solution:**
 - Cluster regions with similar trends and share information within a cluster



Discovering similar neighborhoods

- **Task 2:** Forecast violent crimes to better task police
- Again, **cluster regions** and **share information!**
- Leads to **improved predictions** compared to examining each region independently



Washington, DC

SOURCES:

1. Numerical of K-means: https://www.youtube.com/watch?v=_S5tvagaQRU
2. Distance Measures, K-means flow chart: <https://www.youtube.com/watch?v=Xvwt7y2jf5E>
3. Only the Continuous, Discreet Box: <https://www.youtube.com/watch?v=9991JIKnFmk>
4. Applications of k-means:
<https://www.coursera.org/lecture/ml-clustering-and-retrieval/other-applications-of-clustering-ADVKE>
5. Choosing the value of 'k' - Andrew Ng:
<https://www.coursera.org/lecture/machine-learning/choosing-the-number-of-clusters-Ks0E9>
6. Stopping Criteria of k-means:
<https://towardsdatascience.com/k-means-clustering-from-a-to-z-f6242a314e9a>
7. Numerical in Hindi (Slide #34 solution): https://www.youtube.com/watch?v=YWgckSa_2ag