VIT BHOPAL
www.vitbhopal.ac.in

Introduction to clustering

(K-Mode Clustering Algorithm)

Dr. Virendra Singh Kushwah

Assistant Professor Grade-II

School of Computing Science and Engineering

Virendra.Kushwah@vitbhopal.ac.in

7415869616

K-Means

(Continuous data)

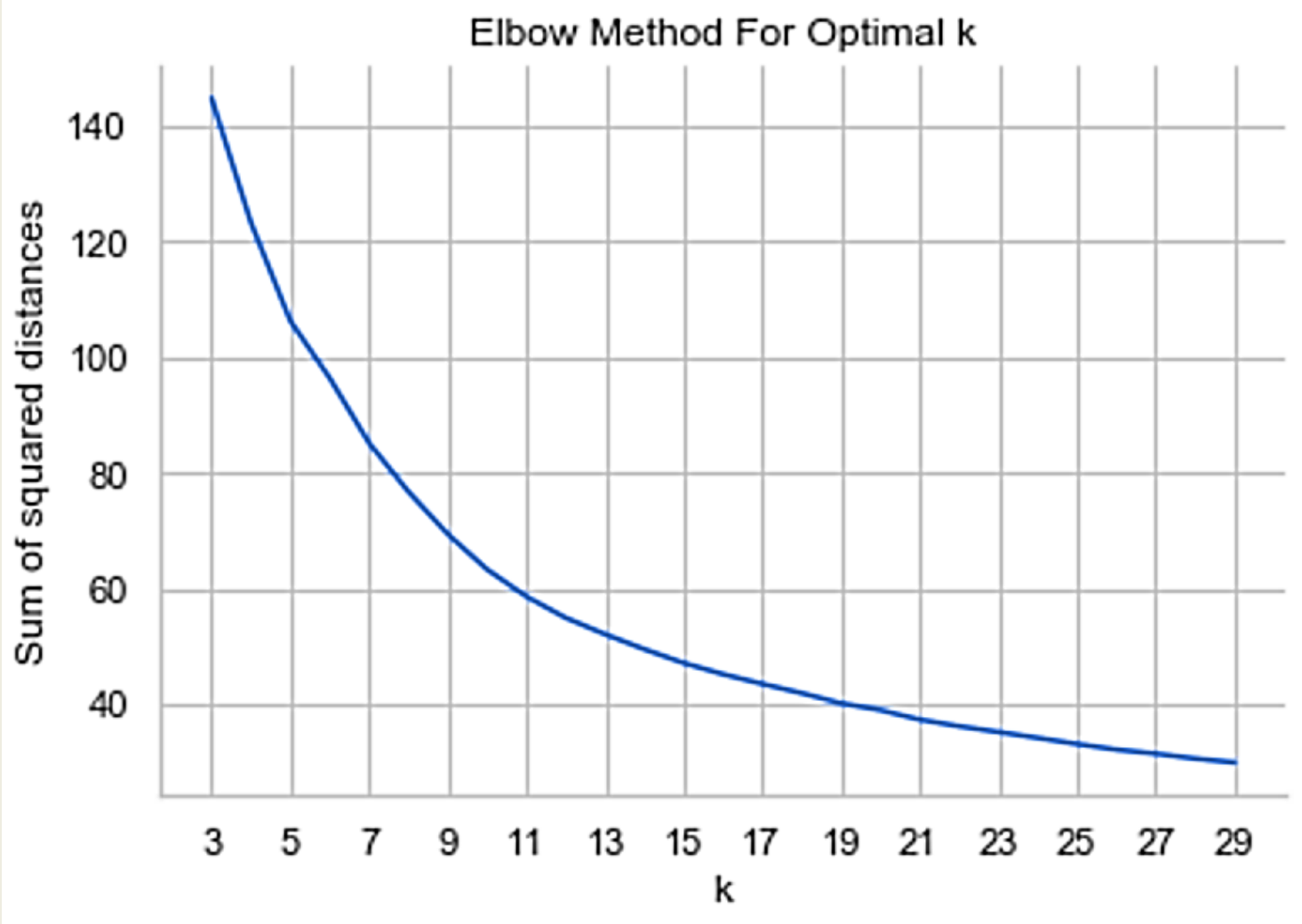
- K-Means is probably the most popular clustering algorithm. Thanks to this, as well as its simplicity and its ability to scale, it has become the go-to option for most data scientists.

The Algorithm

- The user decides the number of resulting clusters (denoted K). K points are randomly assigned to be the cluster centers.
- From there, the algorithm assigns all the other points in the dataset to one of the clusters by taking the cluster whose Euclidean distance with the point is minimal.
- Following this, the cluster centers are re-calculated by taking the average of each points' coordinates.
- The algorithm reassigns every point to the closest cluster and repeats the process until the clusters converge and don't change more.

- Note that because of the random initialization, results may depend on which points are randomly selected to initialize the clusters.
- Most implementations of the algorithm thus provide the ability to run the algorithms multiple times with different “random starts” so as to select the clustering that minimizes the sum of squared errors (the inertia) of the points and their cluster centers.

- Using elbow plots, it is also very easy to select the right number of clusters (if that is not predetermined by the problem at stake)



Best for …

- General cases where interpretability of the clusters may not be required (i.e. when using as a feature of a supervised problem)
- Problems where a quick solution is sufficient to generate insights for most cases. K-Means' algorithm is relatively efficient.

K-Modes

((Categorical data)

- K-Means also doesn't perform well when in the presence of categorical variables. As for K-medians, an implementation exists to leverage the efficiency of K-Means on categorical data.

The Algorithm

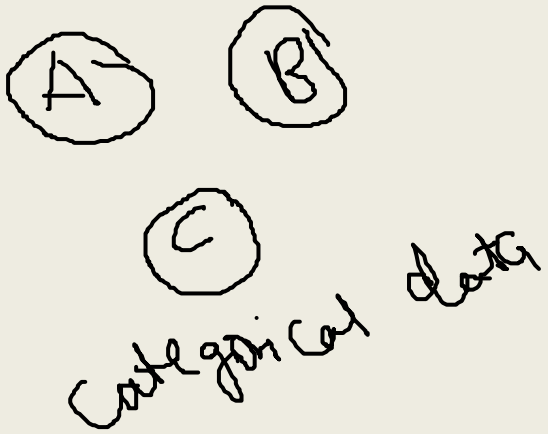
- While K-Means calculates the Euclidean distance between two points, K-Modes attempts to minimize a dissimilarity measure: it counts the number of "features" that are not the same. Using modes in lieu of means, K-Modes becomes able to handle efficiently categorical data

Best for …

- When the dataset contains categorical data exclusively

Analytical record

Individual	Q1	Q2	Q3	Q4	Q5
1	A	B	A	B	C
2	A	A	A	B	B
3	C	A	B	B	A
4	A	B	B	A	C
5	C	C	C	B	A
6	A	A	A	A	B
7	A	C	A	C	C
8	C	A	B	B	C
9	A	A	B	C	A
10	A	B	B	A	C

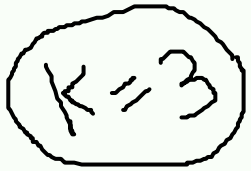


STEPS

1. Pick an observation (instance) at random and use that as a cluster

K-mean → decide the centroid

Clusters

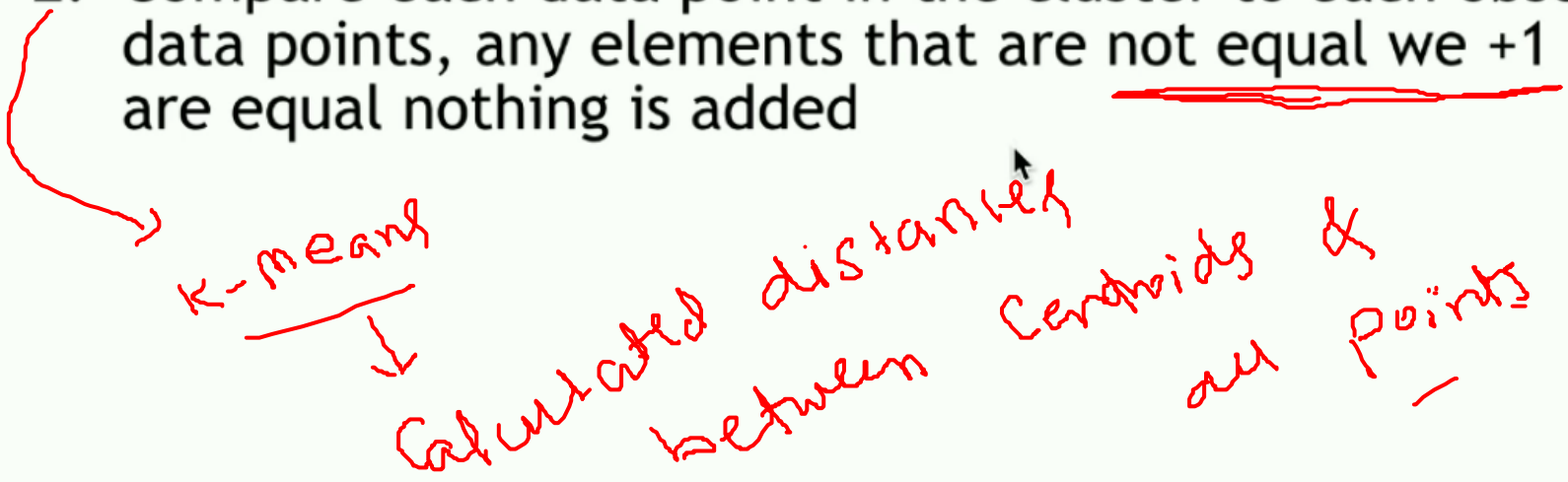


Cluster	Q1	Q2	Q3	Q4	Q5
1 (1)	A	B	A	B	C
2 (5)	C	C	C	B	A
3 (10)	A	B	B	A	C

Individual	Q1	Q2	Q3	Q4	Q5
1	A	B	A	B	C
2	A	A	A	B	B
3	C	A	B	B	A
4	A	B	B	A	C
5	C	C	C	B	A
6	A	A	A	A	B
7	A	C	A	C	C
8	C	A	B	B	C
9	A	A	B	C	A
10	A	B	B	A	C

STEPS

1. Pick an observation (instance) at random and use that as a cluster
2. Compare each data point in the cluster to each observation data points, any elements that are not equal we +1 if they are equal nothing is added



Cluster	Q1	Q2	Q3	Q4	Q5
1 (1)	A	B	A	B	C
2 (5)	C	C	C	B	A
3 (10)	A	B	B	A	C

Individual	Q1	Q2	Q3	Q4	Q5	C1	C2	C3
1	A	B	A	B	C	0	4	2
Cluster 1	A	B	A	B	C			
3	C	A	B	B	A			
4	A	B	B	A	C			
5	C	C	C	B	A			
6	A	A	A	A	B			
7	A	C	A	C	C			
8	C	A	B	B	C			
9	A	A	B	C	A			
10	A	B	B	A	C			

Cluster	Q1	Q2	Q3	Q4	Q5
1 (1)	A	B	A	B	C
2 (5)	C	C	C	B	A
3 (10)	A	B	B	A	C

Individual	Q1	Q2	Q3	Q4	Q5	C1	C2	C3
1	A	B	A	B	C	0		
Cluster 1	A	B	A	B	C			
3	C	A	B	B	A			
4	A	B	B	A	C			
5	C	C	C	B	A			
6	A	A	A	A	B			
7	A	C	A	C	C			
8	C	A	B	B	C			
9	A	A	B	C	A			
10	A	B	B	A	C			

STEPS

1. Pick an observation (instance) at random and use that as a cluster
2. Compare each data point in the cluster to each observation data points, any elements that are not equal we +1 if they are equal nothing is added

Cluster	Q1	Q2	Q3	Q4	Q5
1 (1)	A	B	A	B	C
2 (5)	C	C	C	B	A
3 (10)	A	B	B	A	C

Individual	Q1	Q2	Q3	Q4	Q5	C1	C2	C3
1	A	B	A	B	C	0	4	
Cluster 2	C	C	C	B	A			
3	C	A	B	B	A			
4	A	B	B	A	C			
5	C	C	C	B	A			
6	A	A	A	A	B			
7	A	C	A	C	C			
8	C	A	B	B	C			
9	A	A	B	C	A			
10	A	B	B	A	C			

Individuals

Individual	Q1	Q2	Q3	Q4	Q5	C1	C2	C3
1	A	B	A	B	C	0	4	1+1 = 2
Cluster 3	A	B	B	A	C			
4	A	B	B	A	C			
5	C	C	C	B	A			
6	A	A	A	A	B			
7	A	C	A	C	C			
8	C	A	B	B	C			
9	A	A	B	C	A			
10	A	B	B	A	C			

Individuals

Individual	Q1	Q2	Q3	Q4	Q5	C1	C2	C3
1	A	B	A	B	C	0	4	2
Cluster 3	A	B	B	A	C			
4	A	B	B	A	C			
5	C	C	C	B	A			
6	A	A	A	A	B			
7	A	C	A	C	C			
8	C	A	B	B	C			
9	A	A	B	C	A			
10	A	B	B	A	C			

Cluster	Q1	Q2	Q3	Q4	Q5
1 (1)	A	B	A	B	C
2 (5)	C	C	C	B	A
3 (10)	A	B	B	A	C

Individual	Q1	Q2	Q3	Q4	Q5	C1	C2	C3
1	A	B	A	B	C	0	4	2
Cluster 2	C	C	C	B	A			
3	C	A	B	B	A			
4	A	B	B	A	C			
5	C	C	C	B	A			
6	A	A	A	A	B			
7	A	C	A	C	C			
8	C	A	B	B	C			
9	A	A	B	C	A			
10	A	B	B	A	C			

Cluster	Q1	Q2	Q3	Q4	Q5
1 (1)	A	B	A	B	C
2 (5)	C	C	C	B	A
3 (10)	A	B	B	A	C

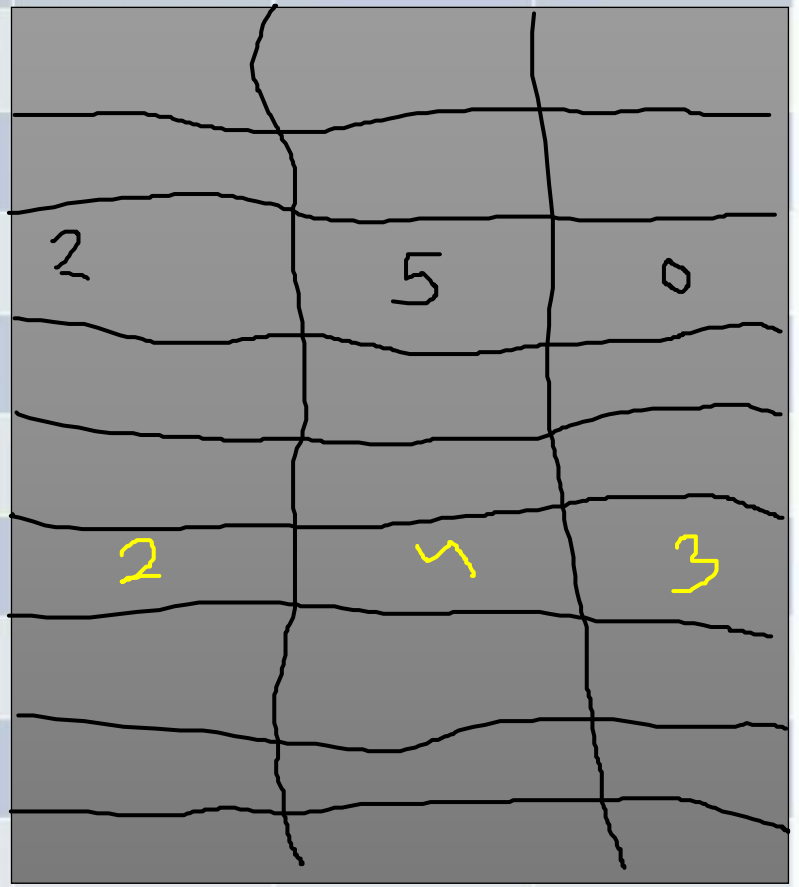
Individual	Q1	Q2	Q3	Q4	Q5	C1	C2	C3
1	A	B	A	B	C	0	4	2
Cluster 2	C	C	C	B	A			
3	C	A	B	B	A			
4	A	B	B	A	C			
5	C	C	C	B	A			
6	A	A	A	A	B			
7	A	C	A	C	C			
8	C	A	B	B	C			
9	A	A	B	C	A			
10	A	B	B	A	C			

Cluster	Q1	Q2	Q3	Q4	Q5
1 (1)	A	B	A	B	C
2 (5)	C	C	C	B	A
3 (10)	A	B	B	A	C

Individual	Q1	Q2	Q3	Q4	Q5
1	A	B	A	B	C
2	A	A	A	B	B
3	C	A	B	B	A
4	A	B	B	A	C
5	C	C	C	B	A
6	A	A	A	A	B
7	A	C	A	C	C
8	C	A	B	B	C
9	A	A	B	C	A
10	A	B	B	A	C

Individuals

Individual	Q1	Q2	Q3	Q4	Q5	C1	C2	C3
1	A	B	A	B	C	0	4	2
2	A	A	A	B	B			
3	C	A	B	B	A			
4	A	B	B	A	C			
5	C	C	C	B	A			
6	A	A	A	A	B			
7	A	C	A	C	C			
8	C	A	B	B	C			
9	A	A	B	C	A			
10	A	B	B	A	C			



Individuals

Individual	Q1	Q2	Q3	Q4	Q5	C1	C2	C3
1	A	B	A	B	C	0	4	2
2	A	A	A	B	B	2	4	4
3	C	A	B	B	A	4	2	4
4	A	B	B	A	C	2	5	0
5	C	C	C	B	A	4	0	5
6	A	A	A	A	B	3	5	4
7	A	C	A	C	C	2	4	3
8	C	A	B	B	C	3	3	3
9	A	A	B	C	A	4	4	3
10	A	B	B	A	C	2	5	0

STEPS

1. Pick an observation (instance) at random and use that as a cluster
2. Compare each data point in the cluster to each observation data points, any elements that are not equal we +1 if they are equal nothing is added
3. Assign each individual to the closest centroid

Individuals

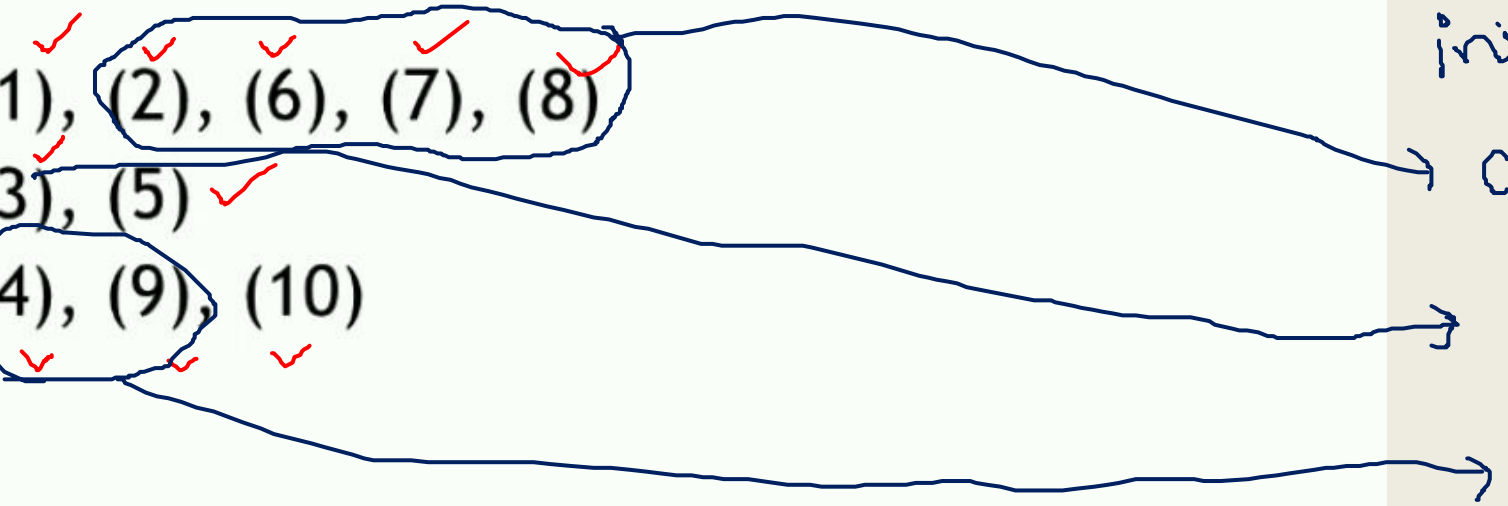
Individual	Q1	Q2	Q3	Q4	Q5	C1	C2	C3
1	A	B	A	B	C	0	4	2
2	A	A	A	B	B	2	4	4
3	C	A	B	B	A	4	2	4
4	A	B	B	A	C	2	5	0
5	C	C	C	B	A	4	0	5
6	A	A	A	A	B	3	5	4
7	A	C	A	C	C	2	4	3
8	C	A	B	B	C	3	3	3
9	A	A	B	C	A	4	4	3
10	A	B	B	A	C	2	5	0

Assign individuals to clusters

Cluster 1: (1), (2), (6), (7), (8)

Cluster 2: (3), (5)

Cluster 3: (4), (9), (10)



at initial

$c_1 = 1$

$c_2 = 5$

$c_3 = 10$

next step

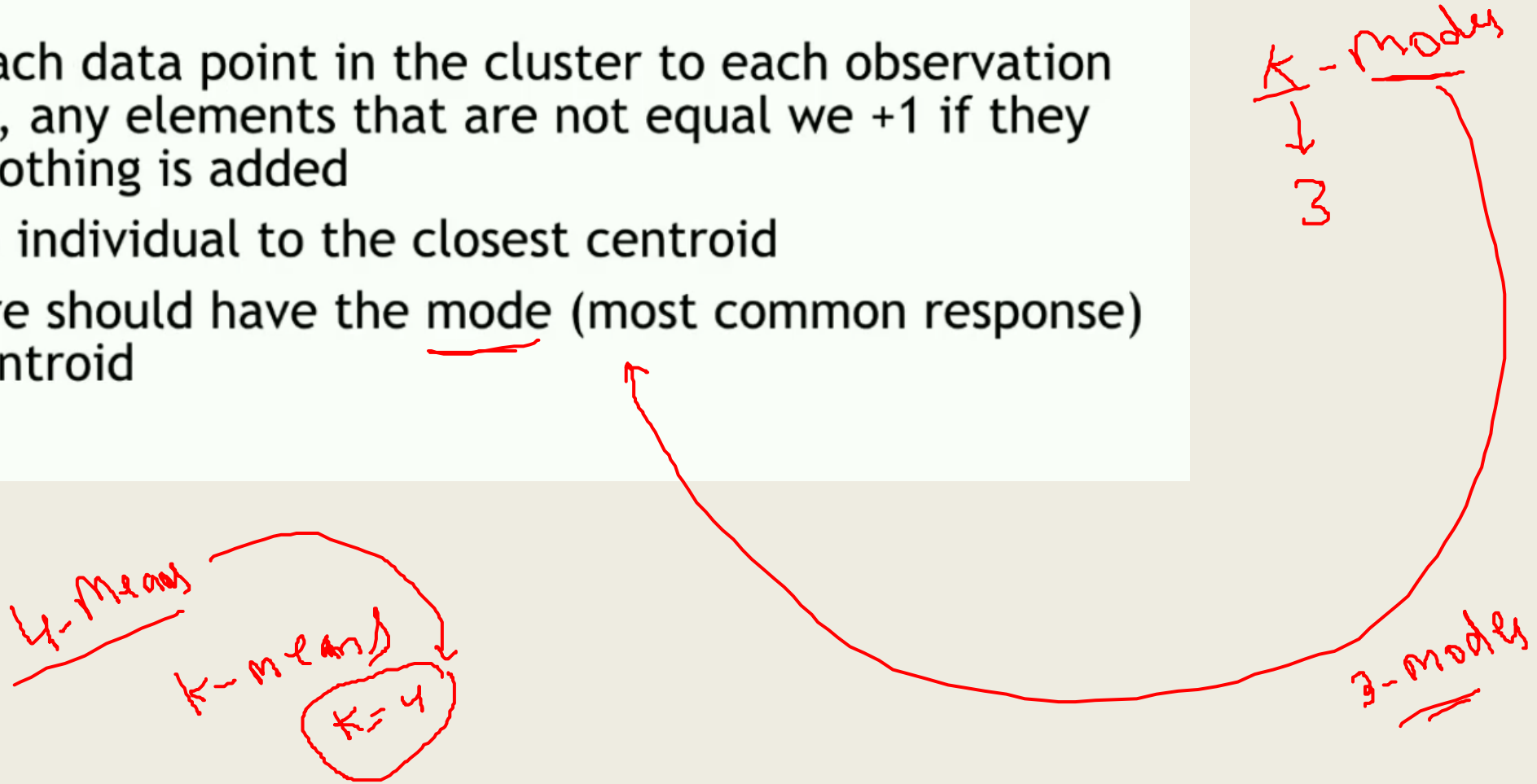
update the clusters or centroids

Individuals

Individual	Q1	Q2	Q3	Q4	Q5	C1	C2	C3
1	A	B	A	B	C	0	4	2
2	A	A	A	B	B	2	4	4
3	C	A	B	B	A	4	2	4
4	A	B	B	A	C	2	5	0
5	C	C	C	B	A	4	0	5
6	A	A	A	A	B	3	5	4
7	A	C	A	C	C	2	4	3
8	C	A	B	B	C	3	3	3
9	A	A	B	C	A	4	4	3
10	A	B	B	A	C	2	5	0

STEPS

1. Pick an observation (instance) at random and use that as a cluster
2. Compare each data point in the cluster to each observation data points, any elements that are not equal we +1 if they are equal nothing is added
3. Assign each individual to the closest centroid
4. Each feature should have the mode (most common response) for each centroid



Individual	Q1	Q2	Q3	Q4	Q5	C1	C2	C3
1	A	B	A	B	C	0	4	2
2	A	A	A	B	B	2	4	4
3	C	A	B	B	A	4	2	4
4	A	B	B	A	C	2	5	0
5	C	C	C	B	A	4	0	5
6	A	A	A	A	B	3	5	4
7	A	C	A	C	C	2	4	3
8	C	A	B	B	C	3	3	3
9	A	A	B	C	A	4	4	3
10	A	B	B	A	C	2	5	0
Cluster 1	A	A	A	B	C			

Individual	Q1	Q2	Q3	Q4	Q5	C1	C2	C3
1	A	B	A	B	C	0	4	2
2	A	A	A	B	B	2	4	4
3	C	A	B	B	A	4	2	4
4	A	B	B	A	C	2	5	0
5	C	C	C	B	A	4	0	5
6	A	A	A	A	B	3	5	4
7	A	C	A	C	C	2	4	3
8	C	A	B	B	C	3	3	3
9	A	A	B	C	A	4	4	3
10	A	B	B	A	C	2	5	0
Cluster 1	A	A	A	B	C			

C2

C

A/C

B/C

B

A

Updation will be done by mode

updated cluster mode table / centroid

updated values after mode

Cluster	Q1	Q2	Q3	Q4	Q5
1 (1), (2), (6), (7), (8)	A	A	A	B	C
2 (3), (5)	C	A	B	B	A
3 (4), (9), (10)	A	B	B	A	C

Cluster	Q1	Q2	Q3	Q4	Q5
1 (1)	A	B	A	B	C
2 (5)	C	C	C	B	A
3 (10)	A	B	B	A	C

Old centroid Table

STEPS

1. Pick an observation (instance) at random and use that as a cluster
2. Compare each data point in the cluster to each observation data points, any elements that are not equal we +1 if they are equal nothing is added
3. Assign each individual to the closest centroid
4. Each feature should have the mode (most common response) for each centroid
5. Repeat steps 2-4 until no changes are made in the assignment of individuals to the closest centroid

Individuals

Individual	Q1	Q2	Q3	Q4	Q5	C1	C2	C3
1	A	B	A	B	C	1	4	3
Cluster 1	A	A	A	B	C	1	3	4
						3	0	4
4	A	B	B	A	C	3	4	0
5	C	C	C	B	A	4	2	5
6	A	A	A	A	B	2	4	3
7	A	C	A	C	C	2	5	3
8	C	A	B	B	C	2	1	3
9	A	A	B	C	A	3	2	3
10	A	B	B	A	C	3	4	0

Assign individuals to clusters

Cluster 1: (1), (2), (6), (7), (8)

Cluster 2: (3), (5) (8)(9)

Cluster 3: (4), (9), (10)

→ next we need to do again get mode of each centroid.