AI Implications for Business Strategy

5. Machine Learning – Looking for Patterns

Supervised & Unsupervised Learning

- Supervised Learning
  - We train the model on ‘training set’ with known labels
  - Then use the model to make predictions
  - Validate predictions using a testing set
- Unsupervised Learning
  - There is NO training
  - Discover interesting patterns / groupings of data
  - Derived from data mining

Unsupervised Learning

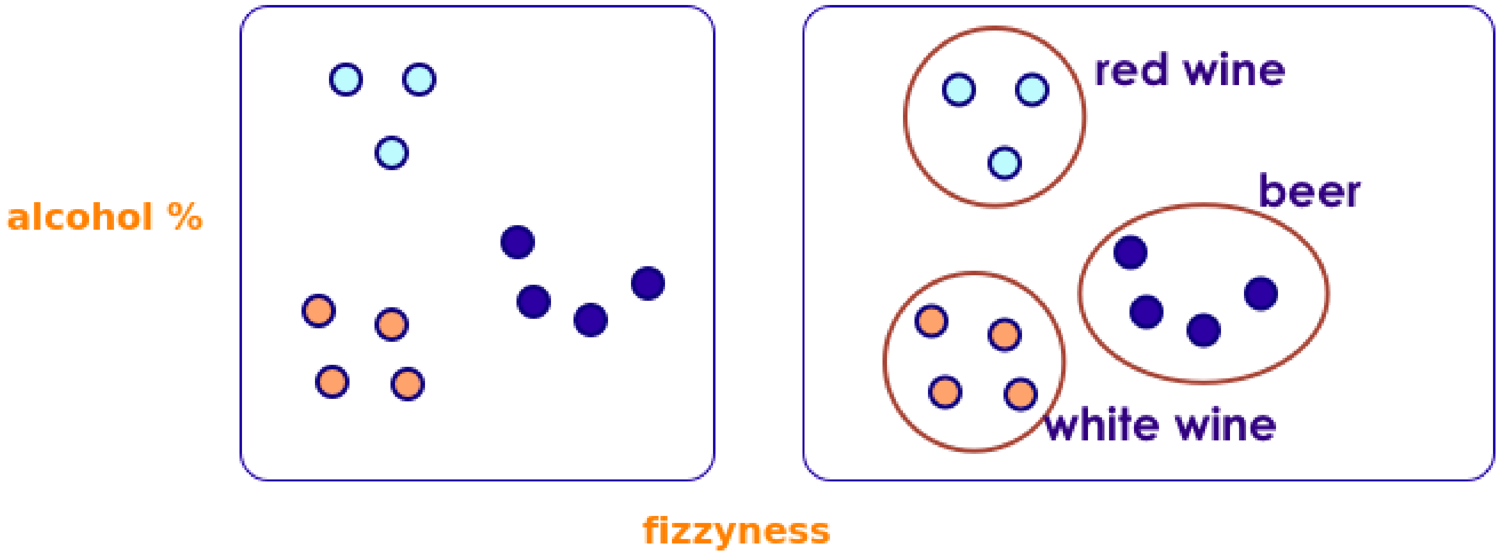
- Goal is to extract meaning from data without training a model on labelled data
  - Big part of exploratory data analysis
  - Sift through large amount of data or large amount of variables
- Cold-start: when we start out a project, we may not have ‘training data’
  - We can use clustering to identify segments or partitions that can be analyzed independently
- Models cannot be tested but can be correlated
  - We have no test data, but we can see if the results extrapolate to other data sets or the population as a whole

Clustering versus Categorization

- Categorization assigns data points to predefined categories
- Clustering looks for similarities between data points
  - Needs the concept of a distance to measure how closely related points are
  - Usually represented as some combination of the features
- For example, looking for clusters of consumer behaviour
  - Reasonable to assume that closeness could be measured in terms of similarity of income and profession, but there might be other clusters that we don’t expect
- Cluster analysis is useful for discovering natural groupings
  - Provides a hypothesis for what features to use for categorization models

Clustering

- Clustering finds natural groupings in data
  - Here we are grouping alcohol beverages according to 2 dimensions (alcohol %, fizziness); And we see similar drinks fall into natural groups
  - In real world applications, we could be clustering by many dimensions (10s or 100s)

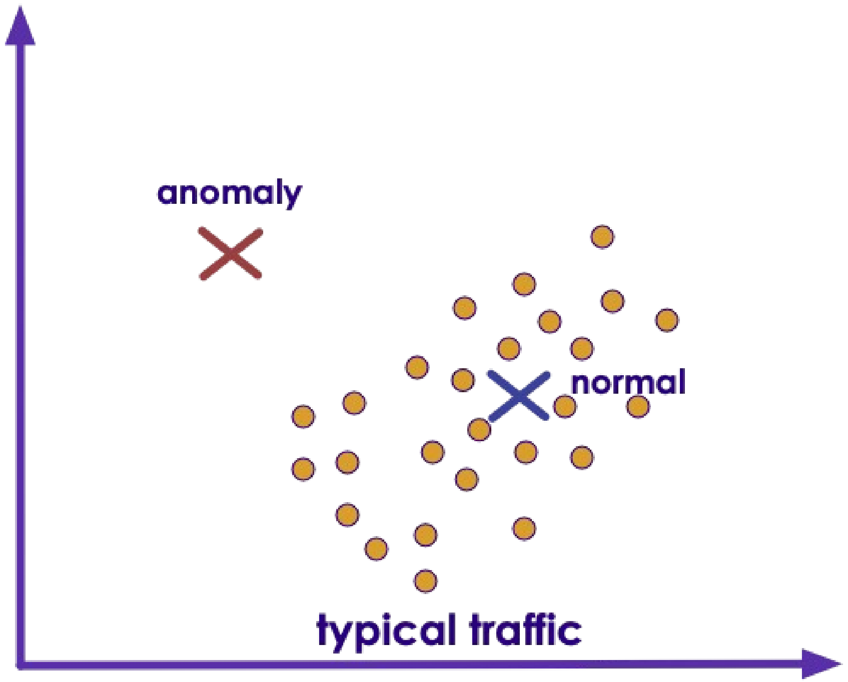


Anomaly Detection

- Extension of clustering where we look for data points that are outliers which are points that are not within the natural clusters detected
  - Eg. A person's purchasing history may cluster along dimensions of amount of purchase and location of purchase (under $200 and usually in and around Toronto)
  - A sudden charge comes through for $10,000 in Moscow
  - This is an outlier and may be indicative of fraud

Clustering Use Cases: Fraud / Anomaly Detection

- Anomaly detection
  - Find fraud
  - Detect network intrusion attack
  - Discover problems on servers
- Clustering does not necessarily detects fraud
  - But it points to unusual data
  - And the need for further investigation

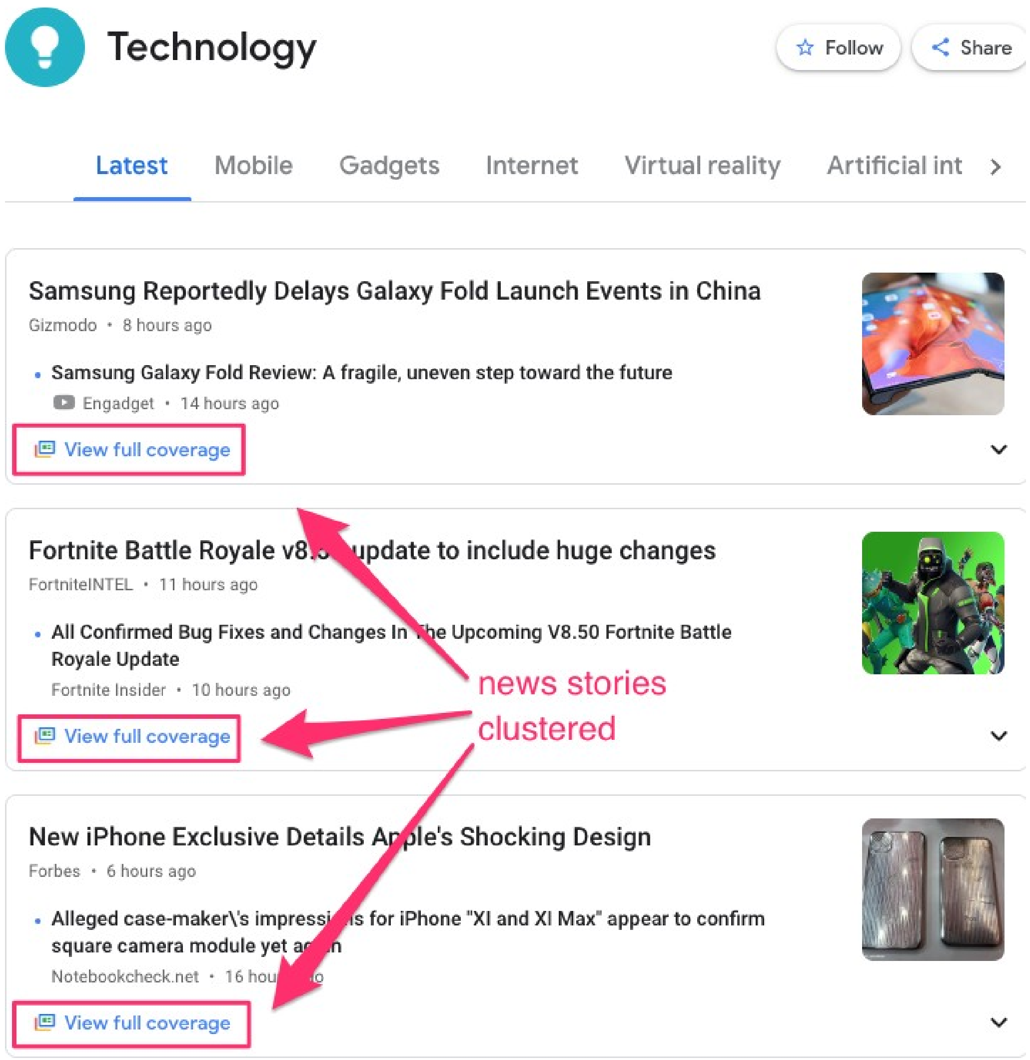


Recommendation Systems

- A number of user choices form a cluster
  - For example, products bought on line by a consumer
  - We want to recommend other products that are close to consumer's choices
- A programmer watches a YouTube video on Java microservices
  - Then other videos that are related to that cluster of topics are recommended
  - Videos on Java
  - Videos on microservices
- Keeping track of all the user's choices allows the cluster to be refined
  - One of the reasons loyalty programs are so important to retailers

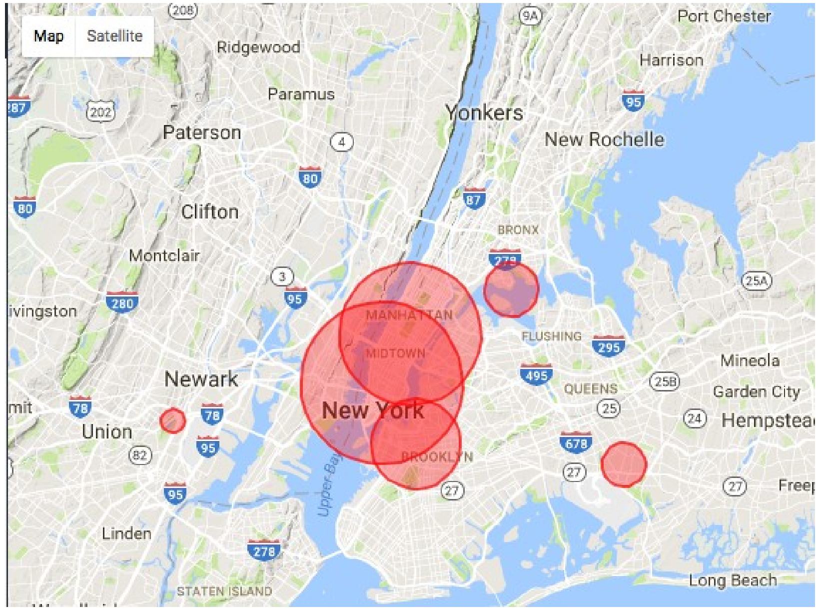
Clustering Example: Google News

- Uses clustering to find related news stories
- The unsupervised learning question is
  - “What do we mean by related?”
    - *Same topic?*
    - *Same people mentioned?*
    - *Same location?*



Clustering Use Case: Uber Trips

- Figure out where demand is so more cars can be sent to that area
  - Customers may be clustering by location and time
  - Example, theatre district on weekends when shows end
- Can be used to make predictions
  - How many cars and drivers are needed where
  - What the pricing should be



Clustering Use Case: Shopping Basket

- Shows which items tend to be bought together
  - Used in product placement
- Or how purchases cluster across time
  - Allows for promotions
- Can be used to look at types of customers based on product clusters
  - Singles, new parents, seniors

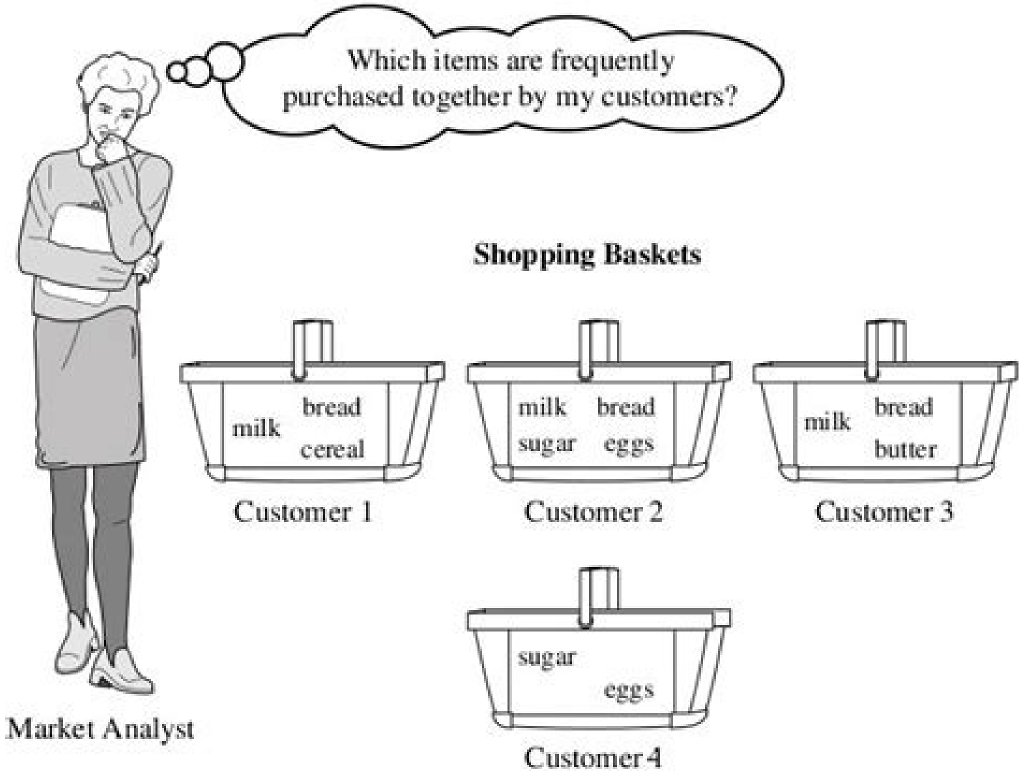


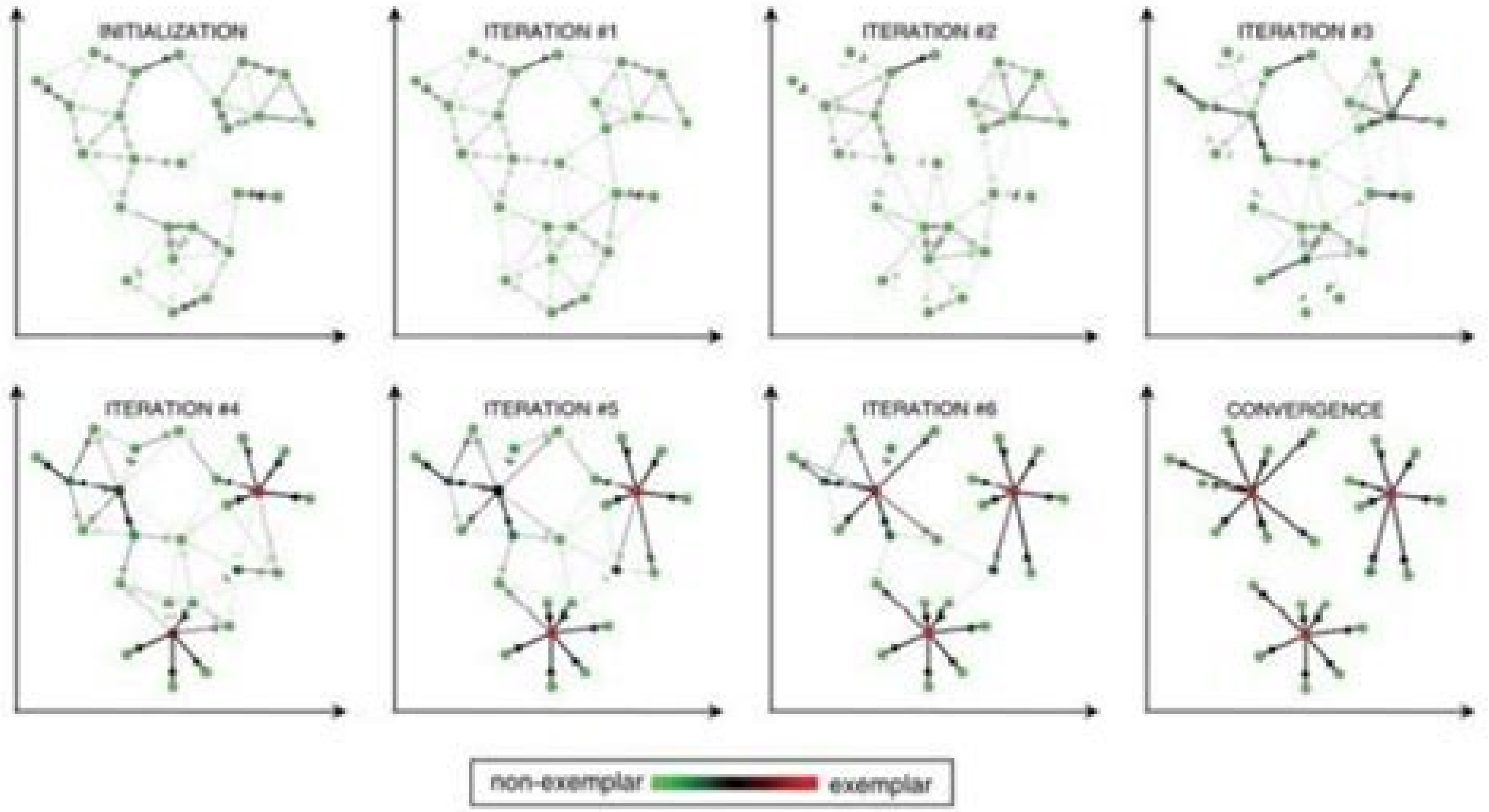
Image Credit: https://www.researchgate.net/figure/Shopping-Basket-Analysis-example-source-33_fig3_320427692

Clustering Algorithms

- A cluster is
  - An area of density in the feature space
  - Data point is “in” a cluster when it is closer to that cluster than other clusters
  - The cluster may have a centre (the centroid) that is a sample or a point at the center of gravity of a feature space and may have a boundary or extent
- The challenges are:
  - Determining the features on which to define a cluster
  - Define what is meant by the distance between two points
  - Assessing the validity of a result may require an assessment by a domain expert

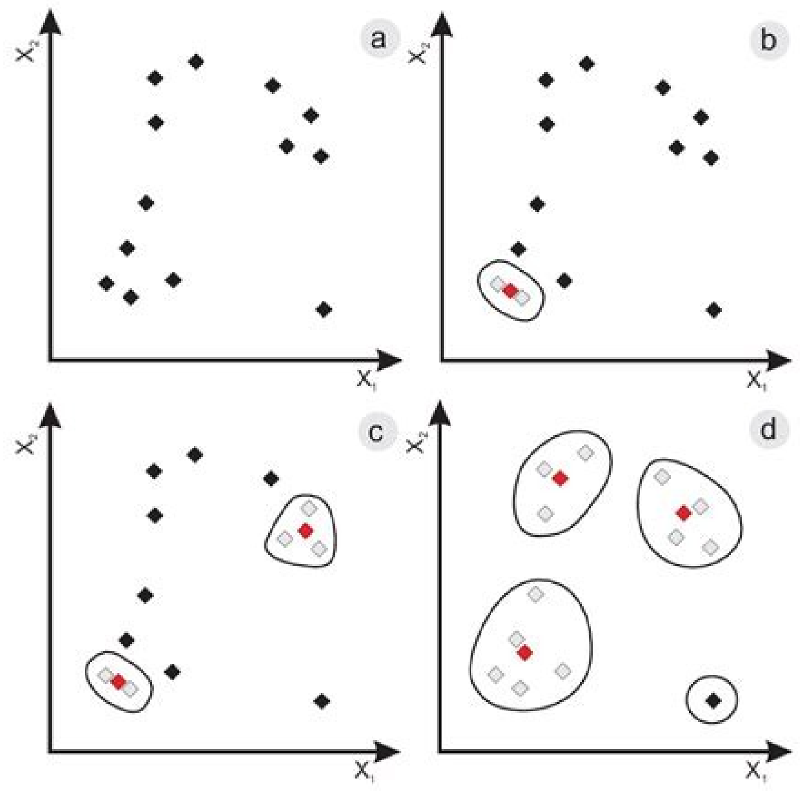
Affinity Propagation

- Involves finding a set of exemplars that best exemplify the data



Agglomerative Clustering

- Merging examples until the desired number of clusters is achieved
  - Bottom up approach
  - One of a number of general hierarchical clustering methods



BIRCH - Balanced Iterative Reducing and Clustering

- Involves constructing a tree structure from which cluster centroids are extracted
  - Designed to be computationally efficient where resources are limited

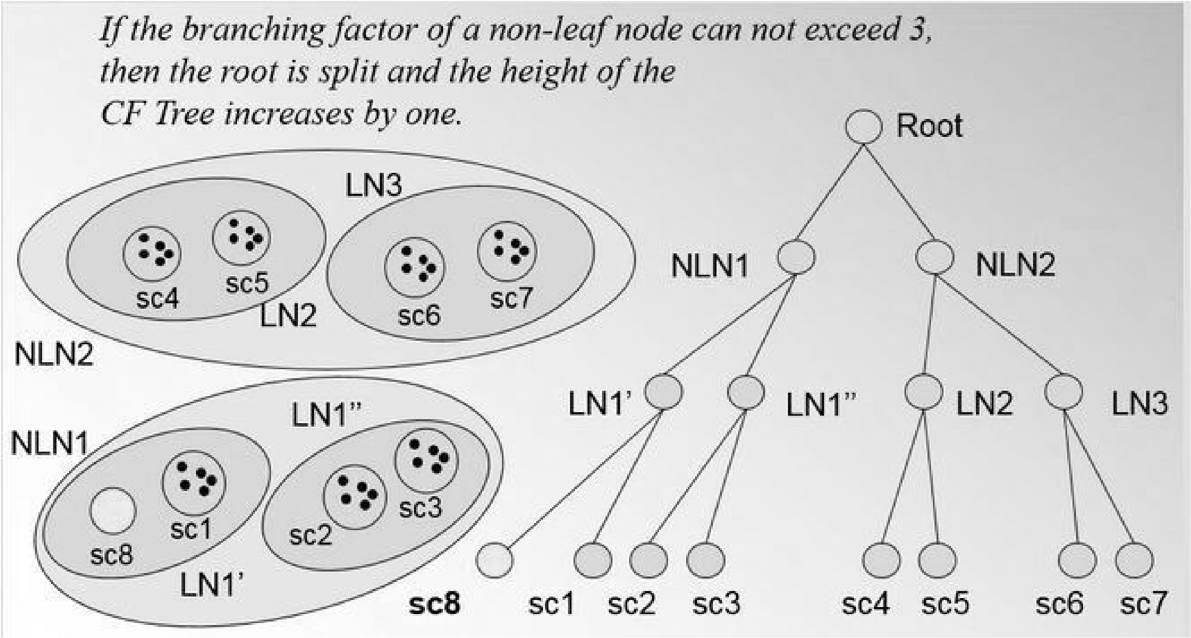


Image Credit:https://laptrinhx.com/birch-clustering-clearly-explained-1946056246/

Divisive Clustering

- Converse of agglomerative clustering
  - Top down
  - Start with one cluster and then
    perform splits recursively

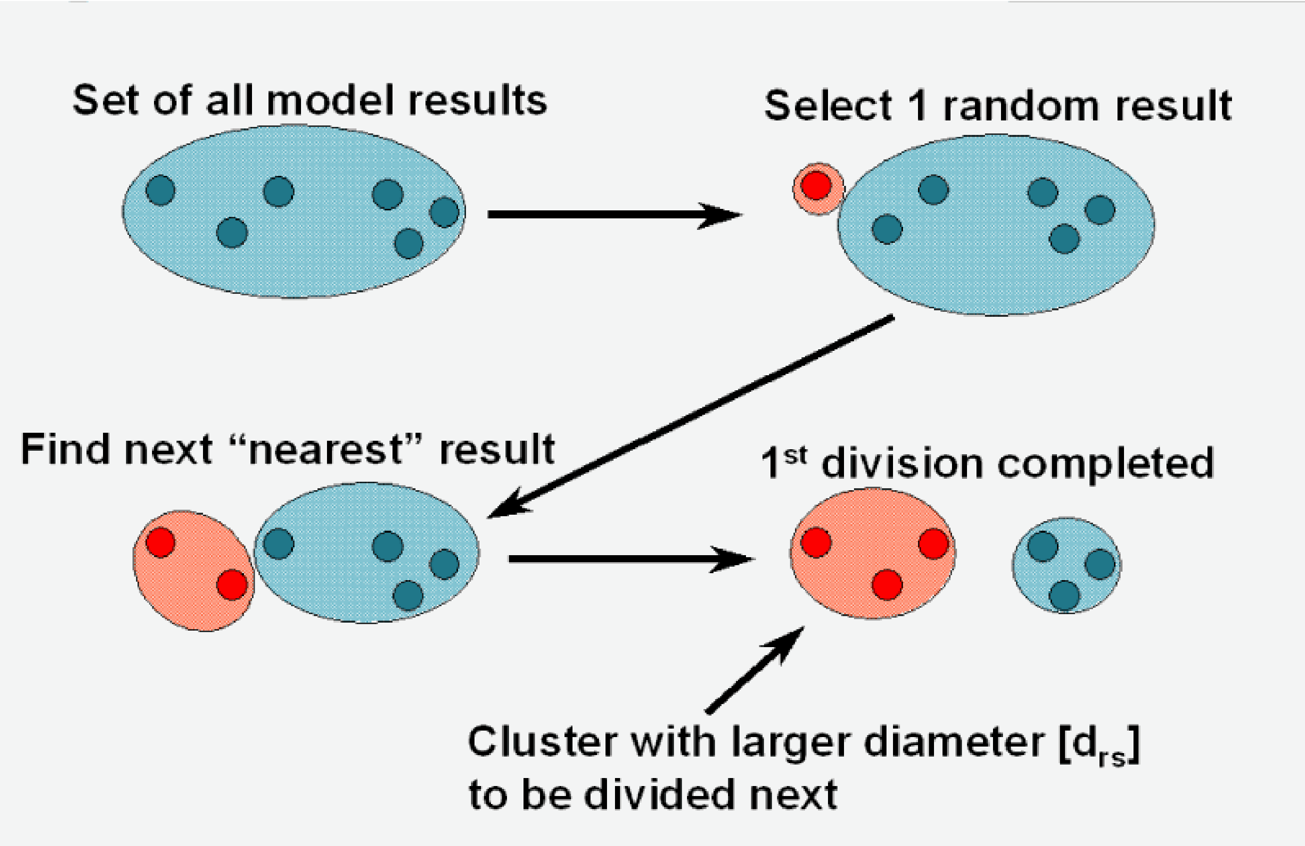
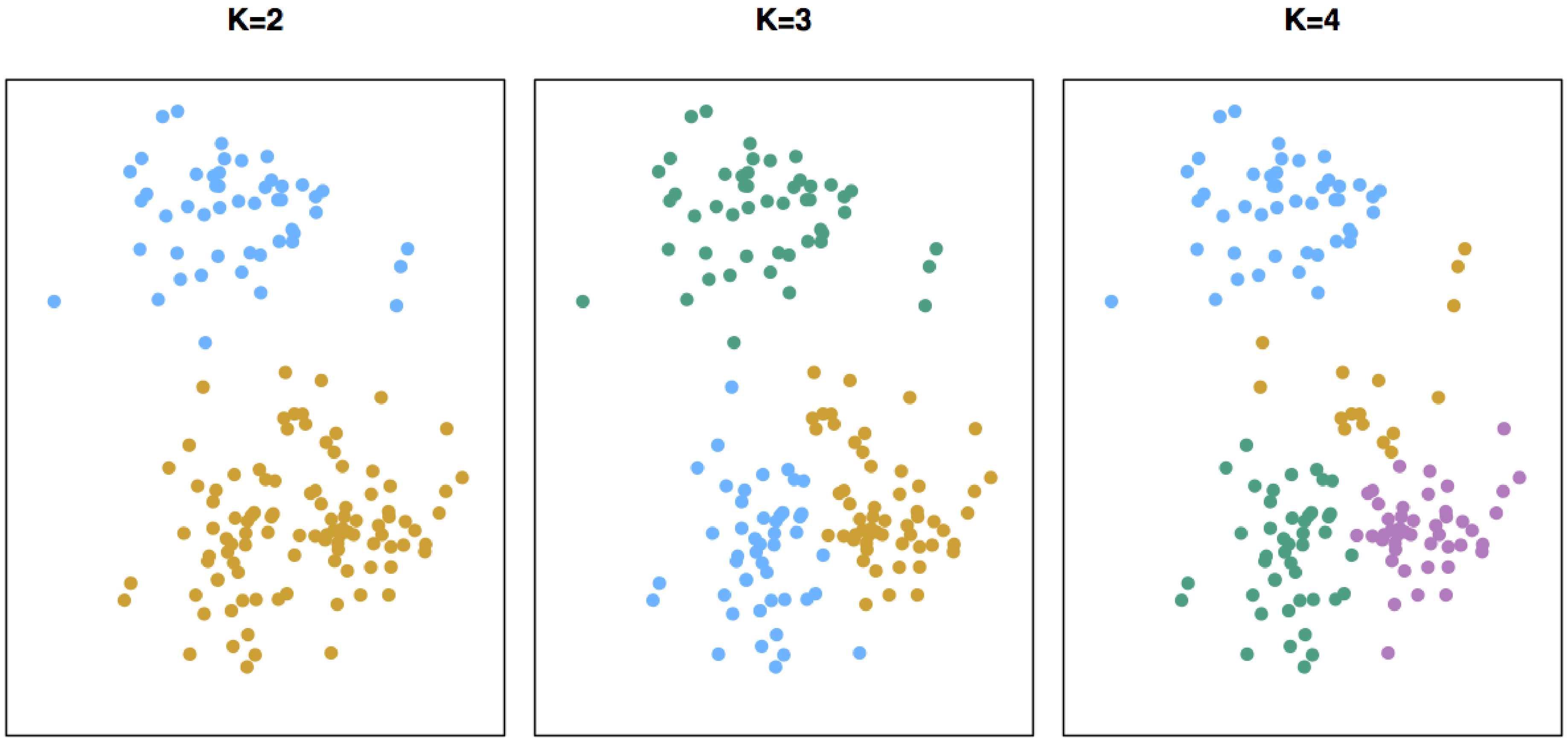


Image Credit:https://www.sedris.org/stc/2004/pp/miv/sld013.htm

K-Means Clustering

- K-means clustering is a simple and elegant approach for partitioning a data set into K distinct, non-overlapping clusters
  - To start K-Means, we need to specify the number of clusters (K)
  - Then the algorithm will assign each observation to exactly one cluster (no overlapping)
  - Not all clusters will have the same size, but the clusters are optimally separated

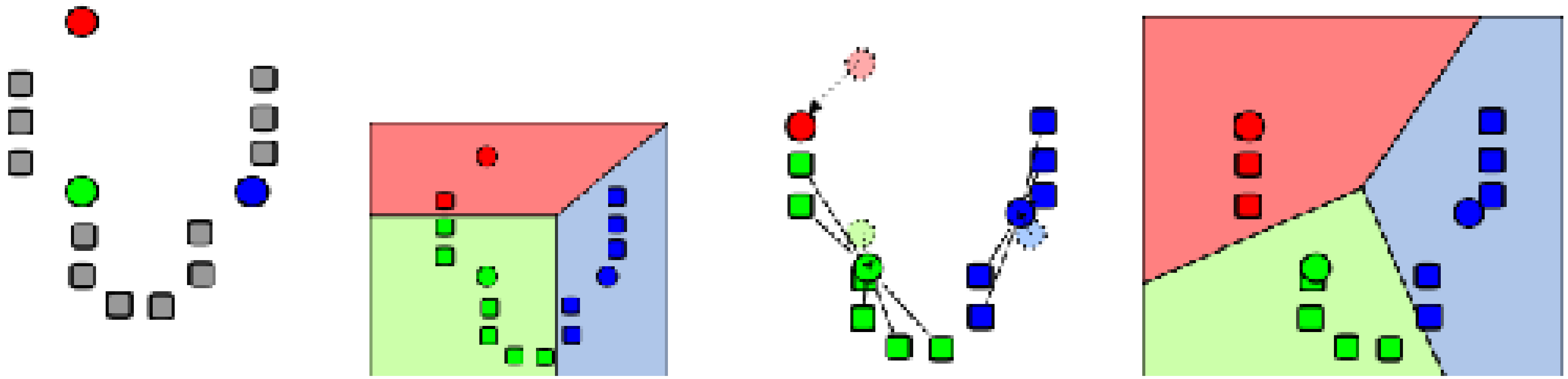
K-Means Visualized



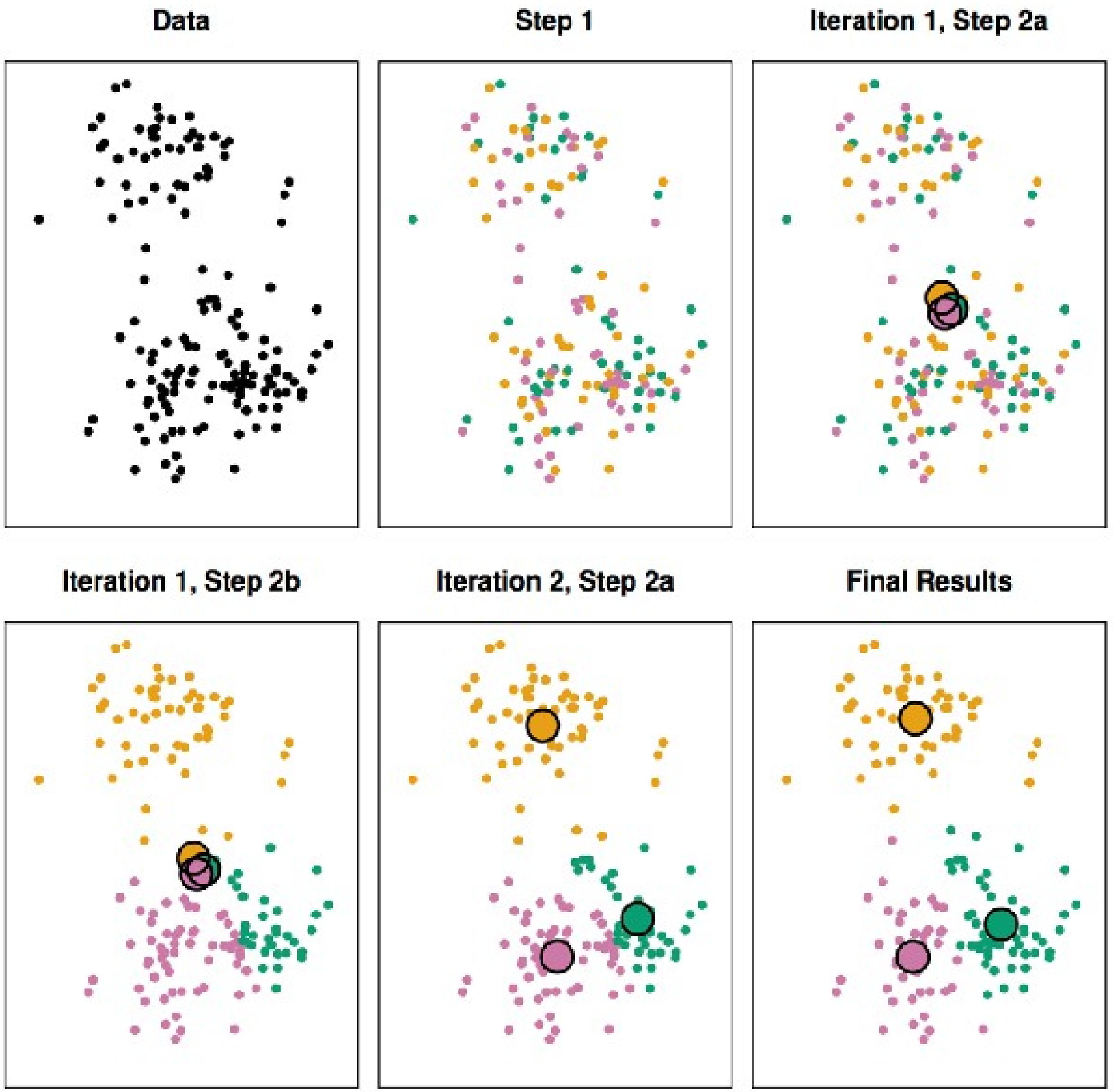
Evaluating K-means Performance

- **Step 1:** Centroids are randomly seeded in the data.
  - Example: 3 centroids (red, green, blue)
- **Step 2:** Each point in the dataset is associated with its nearest centroid, as determined by a distance measurement
- **Step 3:** The centroid (geometric center) of the clustered points becomes the new centroid of that cluster
  - Each centroid updated.
- Step 4: Repeat steps 2 and 3 until convergence is reached (the points move less than a threshold amount).

K-Means Clustering Walkthrough



K-Means Clustering Walkthrough



Evaluating K-means Performance

- We use a method called WSSSE - Within cluster sum of squares by cluster / Within-Cluster-SS
- Remember K-Means works like this
  - Each observation is allocated to closest cluster
  - Measure distance between observation and cluster center
  - Keep iterating until max-K is reached or change in successive WSSSE is less than the threshold value
- Goal is to minimize WSSSE
  - Sum of squared distance from each point to the centroid of its cluster

Example

- We are going to cluster cars using two attributes: MPG & CYL
- 32 data points
- Attributes
  - name - name of the car
  - mpg - Miles/(US) gallon
  - cyl - Number of cylinders
  - disp - Displacement (cu.in.)
  - hp - Gross horsepower
  - drat - Rear axle ratio

	row.names	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
1	Mazda RX4	21.0	6	160.0	110	3.90	2.620	16.46	0	1	4	4
2	Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875	17.02	0	1	4	4
3	Datsun 710	22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1
4	Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1
5	Hornet Sportabout	18.7	8	360.0	175	3.15	3.440	17.02	0	0	3	2
6	Valiant	18.1	6	225.0	105	2.76	3.460	20.22	1	0	3	1
7	Duster 360	14.3	8	360.0	245	3.21	3.570	15.84	0	0	3	4
8	Merc 240D	24.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2
9	Merc 230	22.8	4	140.8	95	3.92	3.150	22.90	1	0	4	2
10	Merc 280	19.2	6	167.6	123	3.92	3.440	18.30	1	0	4	4
11	Merc 280C	17.8	6	167.6	123	3.92	3.440	18.90	1	0	4	4
12	Merc 450SE	16.4	8	275.8	180	3.07	4.070	17.40	0	0	3	3
13	Merc 450SL	17.3	8	275.8	180	3.07	3.730	17.60	0	0	3	3
14	Merc 450SLC	15.2	8	275.8	180	3.07	3.780	18.00	0	0	3	3
15	Cadillac Fleetwood	10.4	8	472.0	205	2.93	5.250	17.98	0	0	3	4

Starting With K = 2 (Find 2 clusters)

Group 0 (16 cars out of 32)		
Car Name	MPG ⬇	CYL
Lincoln Continental	10.4	8
Camaro Z28	13.3	8
Maserati Bora	15	8
Merc 450SL	17.3	8
Merc 280	19.2	6
…		
Pontiac Firebird	19.2	8

Group 1 (16 cars out of 32)		
Car Name	MPG ⬇	CYL
Mazda RX4	21	6
Merc 230	22.8	4
Porsche 914-2	26	4
Fiat X1-9	27.3	4
…		
Honda Civic	30.4	4
Toyota Corolla	33.9	4

K	WSSSE
2	425

Starting With K = 4 (Find 4 clusters)

Group 0 (2 cars / 32)		
Car Name	MPG ⬇	CYL
Lincoln Continental	10.4	8
Cadillac Fleetwood	10.4	8

Group 1 (10 cars / 32)		
Car Name	MPG ⬇	CYL
Camaro Z28	13.3	8
Maserati Bora	15	8
Merc 450SE	16.4	8
Merc 450SL	17.3	8

Group 2 (14 cars / 32)		
Car Name	MPG ⬇	CYL
Merc 280C	17.8	6
Pontiac Firebird	19.2	8
Mazda RX4	21.0	6
…		
Merc 240D	24.4	4

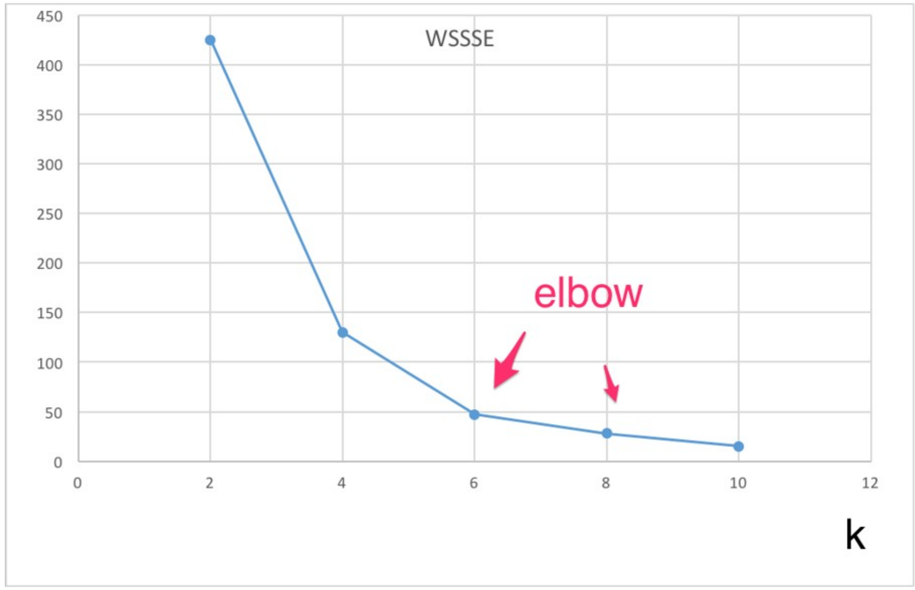
Group 3 (6 cars / 32)		
Car Name	MPG ⬇	CYL
Porsche 914-2	26	4
Fiat X1-9	27.3	4
Honda Civic	30.4	4
…		
Toyota Corolla	33.9	4

K	WSSSE
2	425
4	130

Evaluating K-Means With WSSSE

- Goal is to minimize WSSSE with reasonable effort
  - We look for elbows - that indicates a reasonable clustering
  - After the elbow, the improvement is minimal
  - At k=32 we have achieved WSSSE=0 , as in perfect fit

K	WSSSE
2	425
4	130
6	47
8	28
10	15
16	2
32	0.01



K-Means Drawbacks

- Initial centroid positions are very important
  - Badly initialized centroids can lead to
    - *sub-optimal solution ("local minima" phenomenon)*
    - *can take too long to converge*
- No deterministic way to guarantee the clustering is optimal (NP hard)
  - Choose centroid randomly
  - Do several runs
  - Compare WSSSE score
- Lloyd's algorithm can be used overcome some of these issues
  - Also called Voronoi iteration or relaxation
  - Finds evenly spaced sets of points in subsets of Euclidean spaces
  - Partitions of these subsets into well-shaped and uniformly sized convex cells

K-Means Strengths & Weaknesses

- Strengths
  - K-Means is simple, well-understood
  - Verification straightforward
  - Easy to parallelize, scales to large datasets
- Weaknesses:
  - Value of k must be known in advance, which may mean running the exercise many times to get optimum results.
  - Initial centroid positions are important; may cause long convergence.
  - Outliers may bias results.
  - Clusters not broadly (hyper)spherical don’t work well for k-means.
    - *Use hierarchical clustering for these situations.*

End of Module