GENERAL ASSEMBLY

DATA SCIENCE

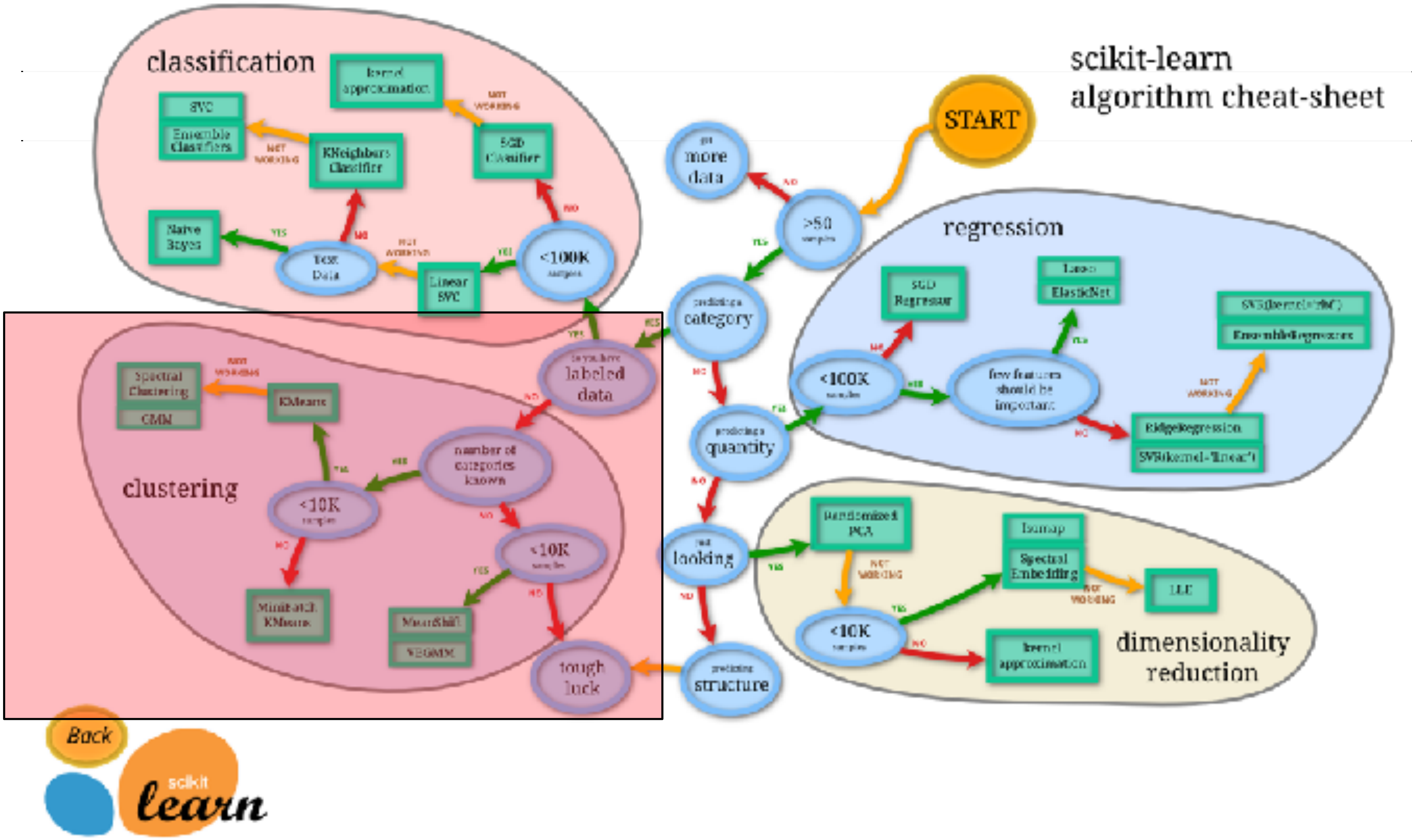
DAT12SYD

Clustering

AGENDA

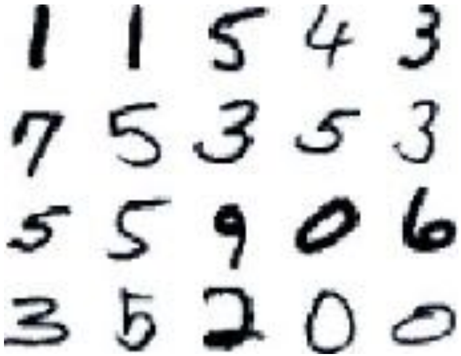
1. **Motivation / Review**
2. **What is Clustering?**
3. **What is K-Means and how does it work?**
4. **Lab**
5. **Discussion**

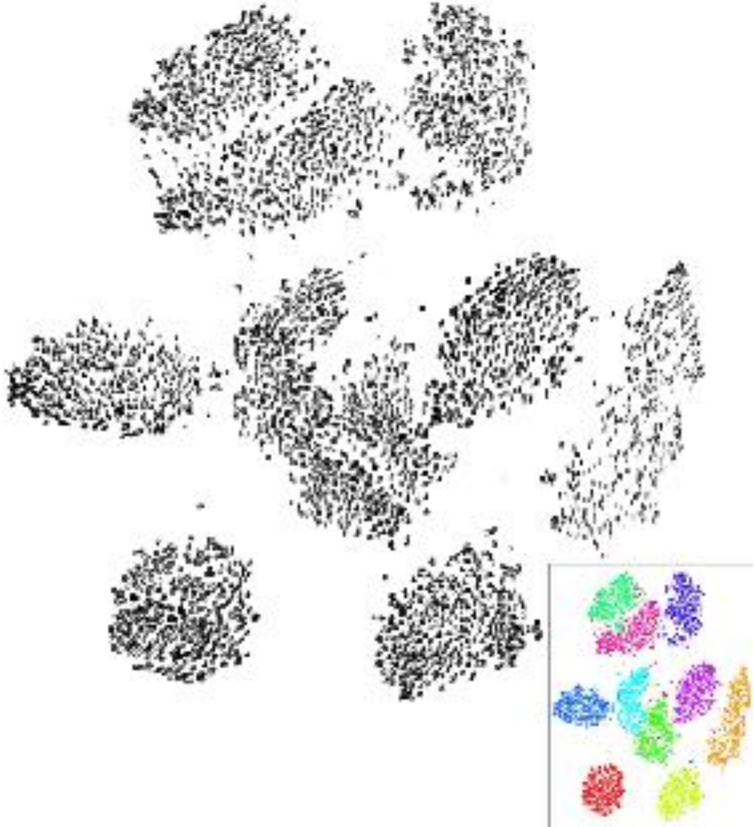
WHAT IS CLUSTERING AND WHY DO IT?



scikit-learn
algorithm cheat-sheet
START
classification
regression
clustering
dimensionality reduction
>50 samples
get more data
predicting a category
do you have labeled data
predicting a quantity
just looking
predicting structure
tough luck
<100K samples
Linear SVC
Text Data
Naive Bayes
KNeighbors Classifier
SVC
Ensemble Classifiers
SGD Classifier
kernel approximation
number of categories known
<10K samples
KMeans
Spectral Clustering
GMM
MiniBatch KMeans
MeanShift
VBGMM
SGD Regressor
few features should be important
Lasso
ElasticNet
RidgeRegression
Randomized PCA
Isomap
Spectral Embedding
LLE
Back
scikit
learn

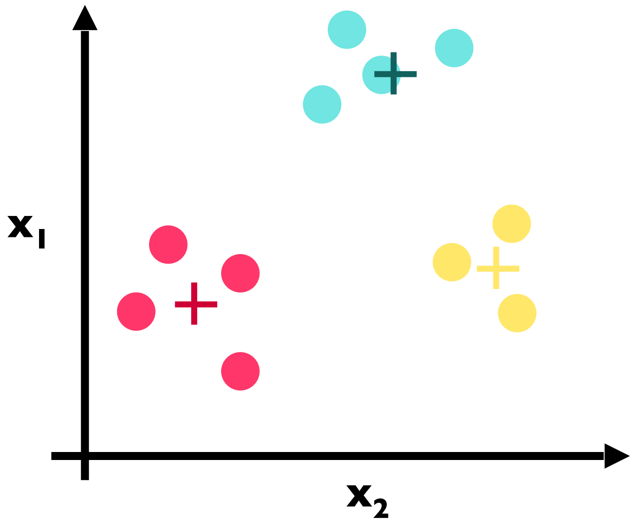
MNIST





Olivetti Faces

- What is a Cluster?
- Why would we do this?
- What is K-Means?



WHAT IS A CLUSTER?

- Unsupervised learning => **find interesting patterns** or groups in data.
- **No** variable we are trying to predict (a **Y value**).
- Clustering discovers **subgroups** in data where the points are similar to each other.
- All points in the same group are similar.
- Points in different groups are different to each other.
- What variables to make groups on. What makes them different (or similar)?

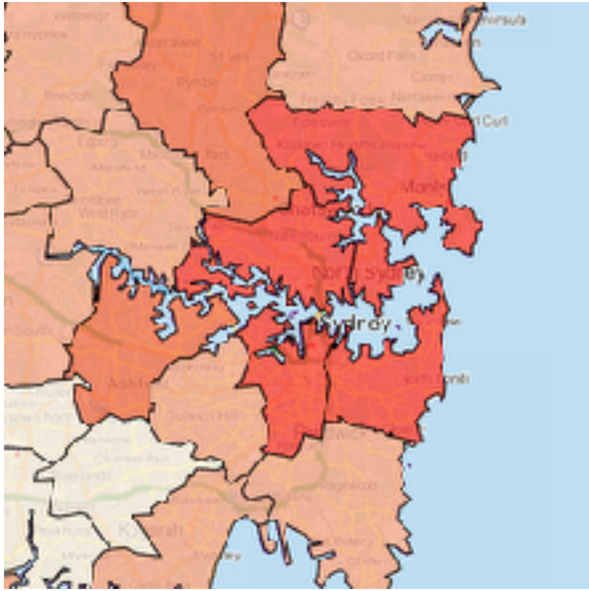
To enhance understanding of a data by **dividing into groups** *(behavioural customer segmentation).*

Clustering provides a **layer of abstraction** *from individual data points (cluster number).*

The goal is to extract and **enhance the natural structure** *of the data*

WHY WOULD WE CLUSTER DATA?

Marketing teams might want to group customers into like groups as a way of summarising the data

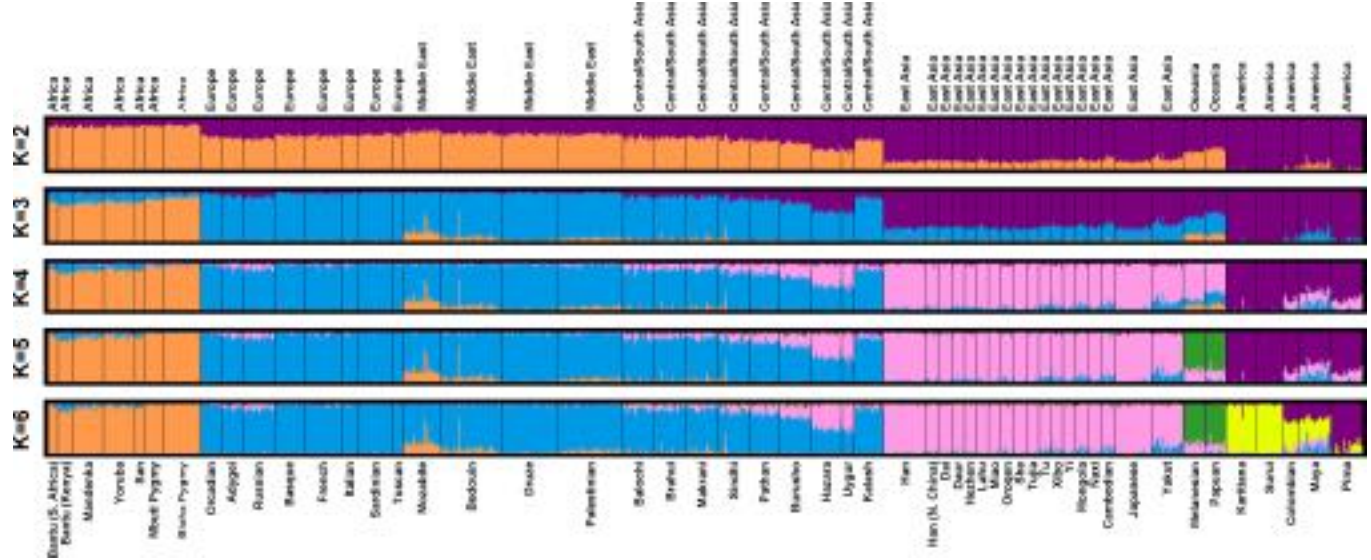


WHY WOULD WE CLUSTER DATA?

Financial groups may want to group transactions into like groups as a way to find unusual payments

WHY WOULD WE CLUSTER DATA?

Genetics data can be clustered to identify ancestry



HOW DO WE CLUSTER DATA?

1) Choose k initial centroids (note that k is an input)

2) For each point:

- find distance to each centroid

- assign point to nearest centroid

3) Recalculate centroid positions

4) Repeat steps 2-3 until stopping criteria met

STEP 1 - CHOOSE CENTROIDS (Whiteboard)

There are several options:

- randomly (but may yield divergent behaviour)
- perform alternative clustering task, use resulting centroids as initial k-means centroids **(warm start)**
- start with global centroid, choose point at max distance, repeat (but might select outlier)

The similarity criterion is determined by the measure we choose.

In the case of k-means clustering, the similarity metric is the **Euclidian distance:**

$$d(x_1, x_2) = \sqrt{\sum_{i=1}^{N} (x_{1i} - x_{2i})^2}$$

Whiteboard distantce btwn 2 pts example

STEP 3 - RECALCULATE CENTROID POSITIONS

Q: How do we re-compute the positions of the centres at each iteration of the algorithm?

STEP 3 - RECALCULATE CENTROID POSITIONS

Q: How do we re-compute the positions of the centres at each iteration of the algorithm?

A: By calculating the centroid (i.e., the geometric centre)

STEP 4 - CONVERGENCE

We iterate until some stopping criteria are met; in general, suitable convergence is achieved in a small number of steps.

Stopping criteria can be based on the centroids (eg, if positions change by no more than ε) or on the points (eg, if no more than x% change clusters between iterations).

1) Choose k initial centroids

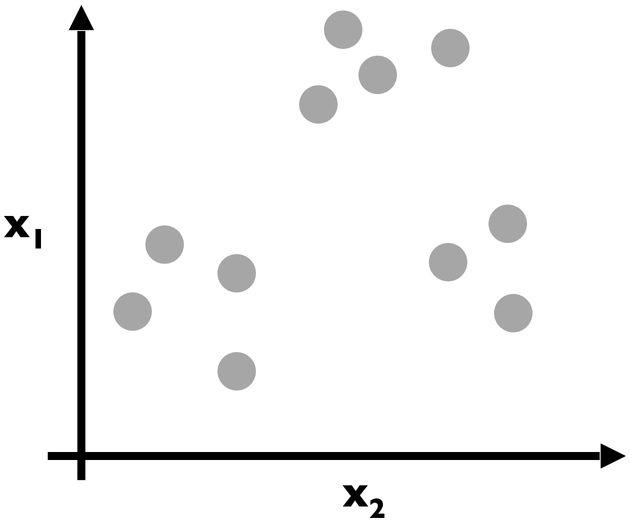
2) For each point:

- find distance to each centroid

- assign point to nearest centroid

3) Recalculate centroid positions

4) Repeat steps 2-3 until stopping criteria met



1) Choose k initial centroids

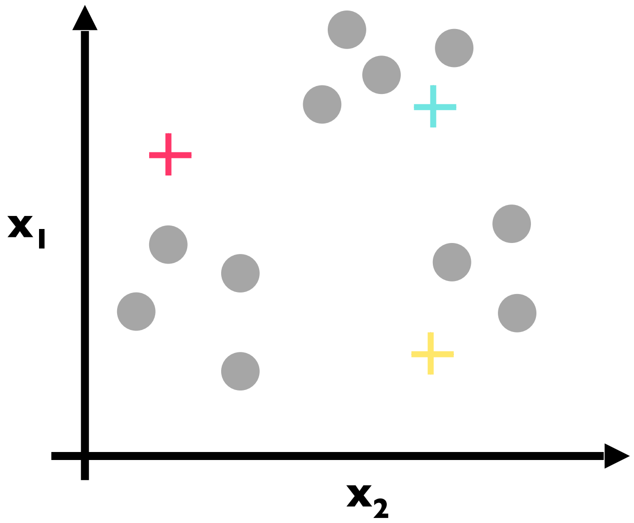
2) For each point:

- find distance to each centroid

- assign point to nearest centroid

3) Recalculate centroid positions

4) Repeat steps 2-3 until stopping criteria met



KMEANS ALGORITHM

1) Choose k initial centroids

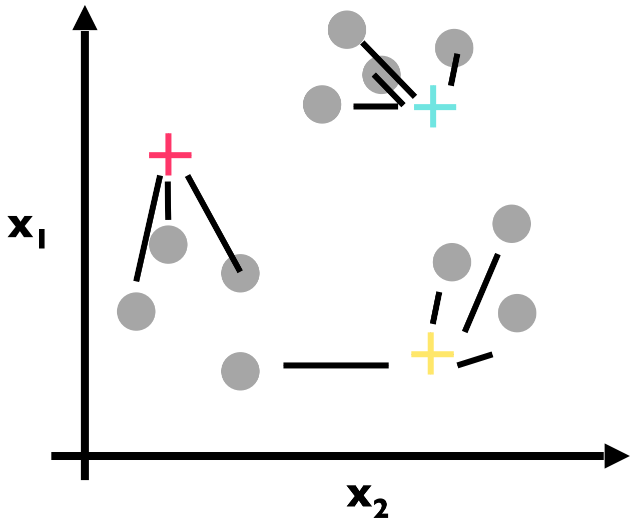
2) For each point:

- find distance to each centroid

- assign point to nearest centroid

3) Recalculate centroid positions

4) Repeat steps 2-3 until stopping criteria met



1) Choose k initial centroids

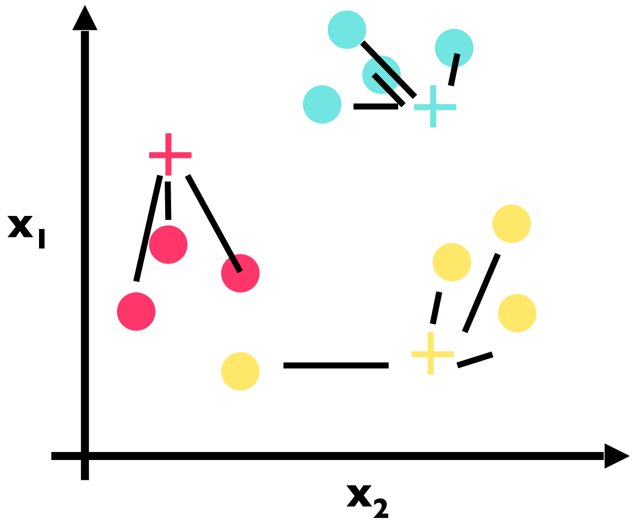
2) For each point:

- find distance to each centroid

- assign point to nearest centroid

3) Recalculate centroid positions

4) Repeat steps 2-3 until stopping criteria met



1) Choose k initial centroids

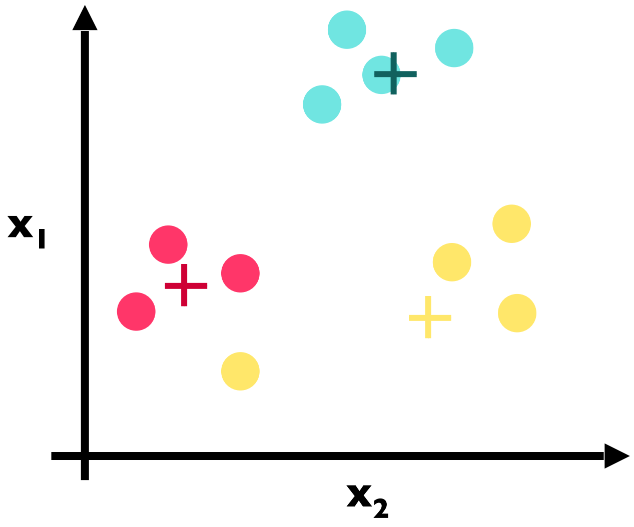
2) For each point:

- find distance to each centroid

- assign point to nearest centroid

3) Recalculate centroid positions

4) Repeat steps 2-3 until stopping criteria met



1) Choose k initial centroids

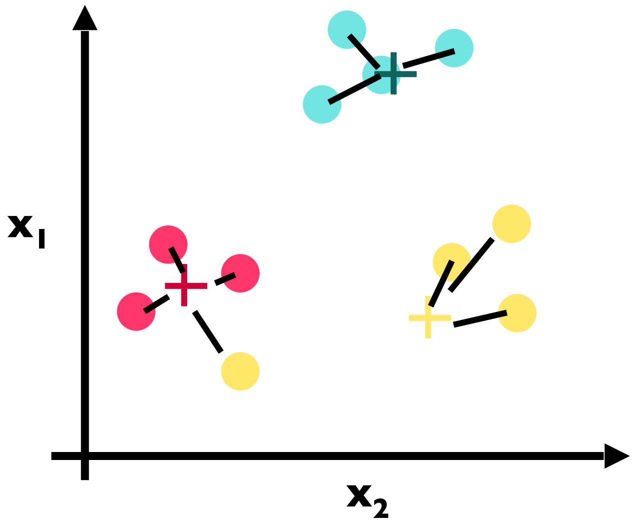
2) For each point:

- find distance to each centroid

- assign point to nearest centroid

3) Recalculate centroid positions

4) Repeat steps 2-3 until stopping criteria met



1) Choose k initial centroids

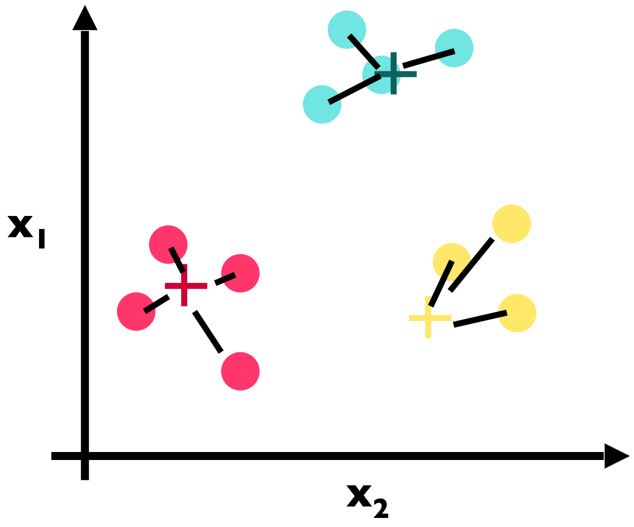
2) For each point:

- find distance to each centroid

- assign point to nearest centroid

3) Recalculate centroid positions

4) Repeat steps 2-3 until stopping criteria met



1) Choose k initial centroids

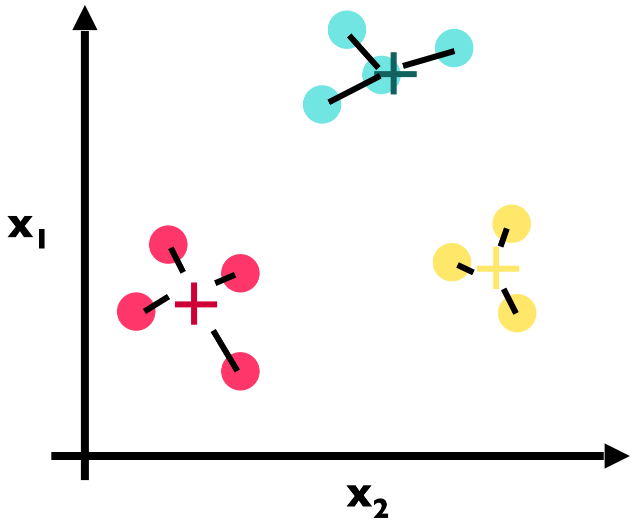
2) For each point:

- find distance to each centroid

- assign point to nearest centroid

3) Recalculate centroid positions

4) Repeat steps 2-3 until stopping criteria met



1) Choose k initial centroids

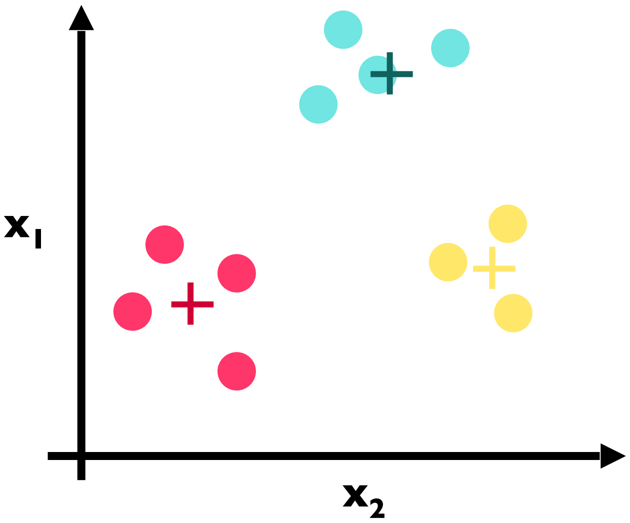
2) For each point:

- find distance to each centroid

- assign point to nearest centroid

3) Recalculate centroid positions

4) Repeat steps 2-3 until stopping criteria met



KMEANS WEB DEMO

http://shabal.in/visuals/kmeans/6.html

KMEANS WEB DEMO

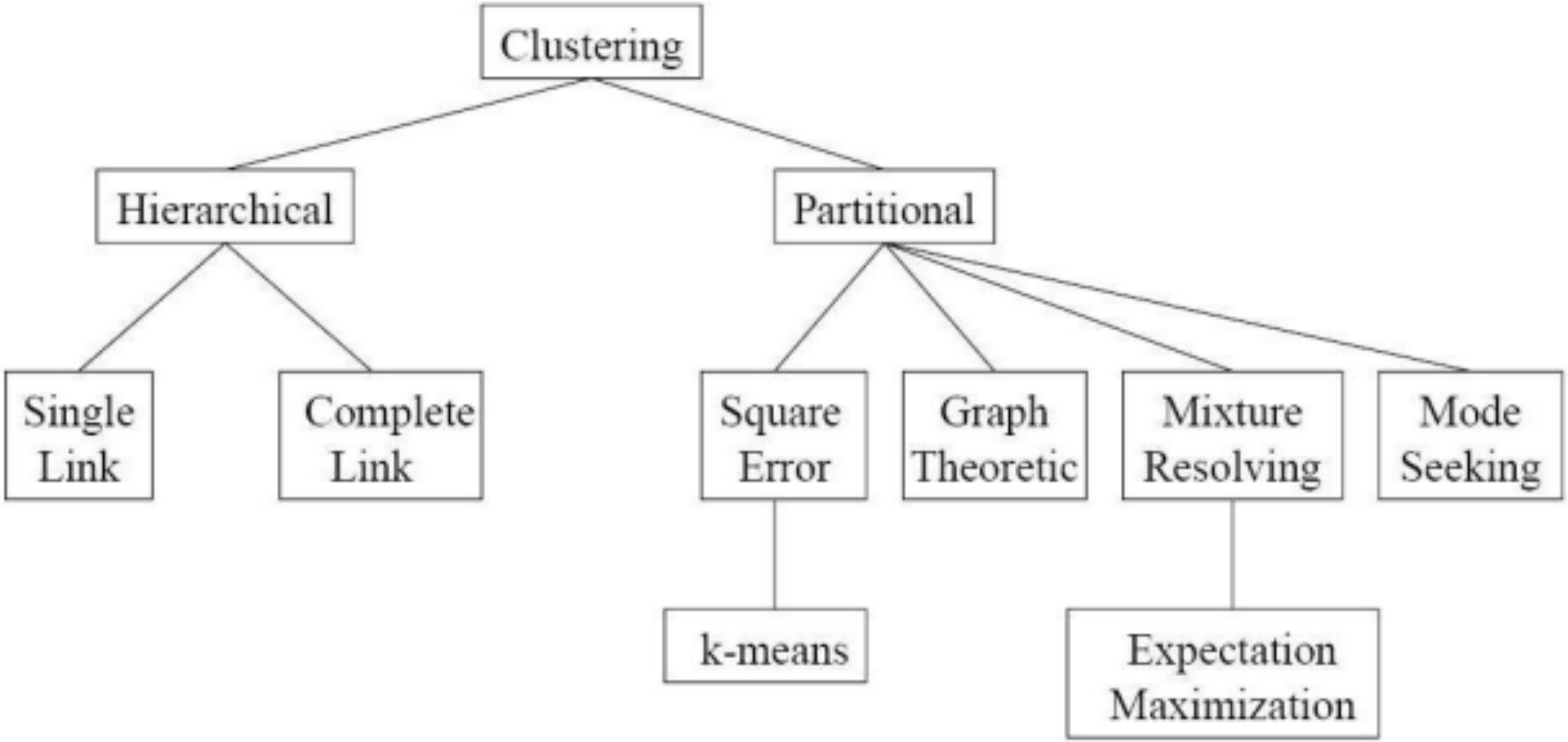
http://shabal.in/visuals/kmeans/6.html

Other good demos:

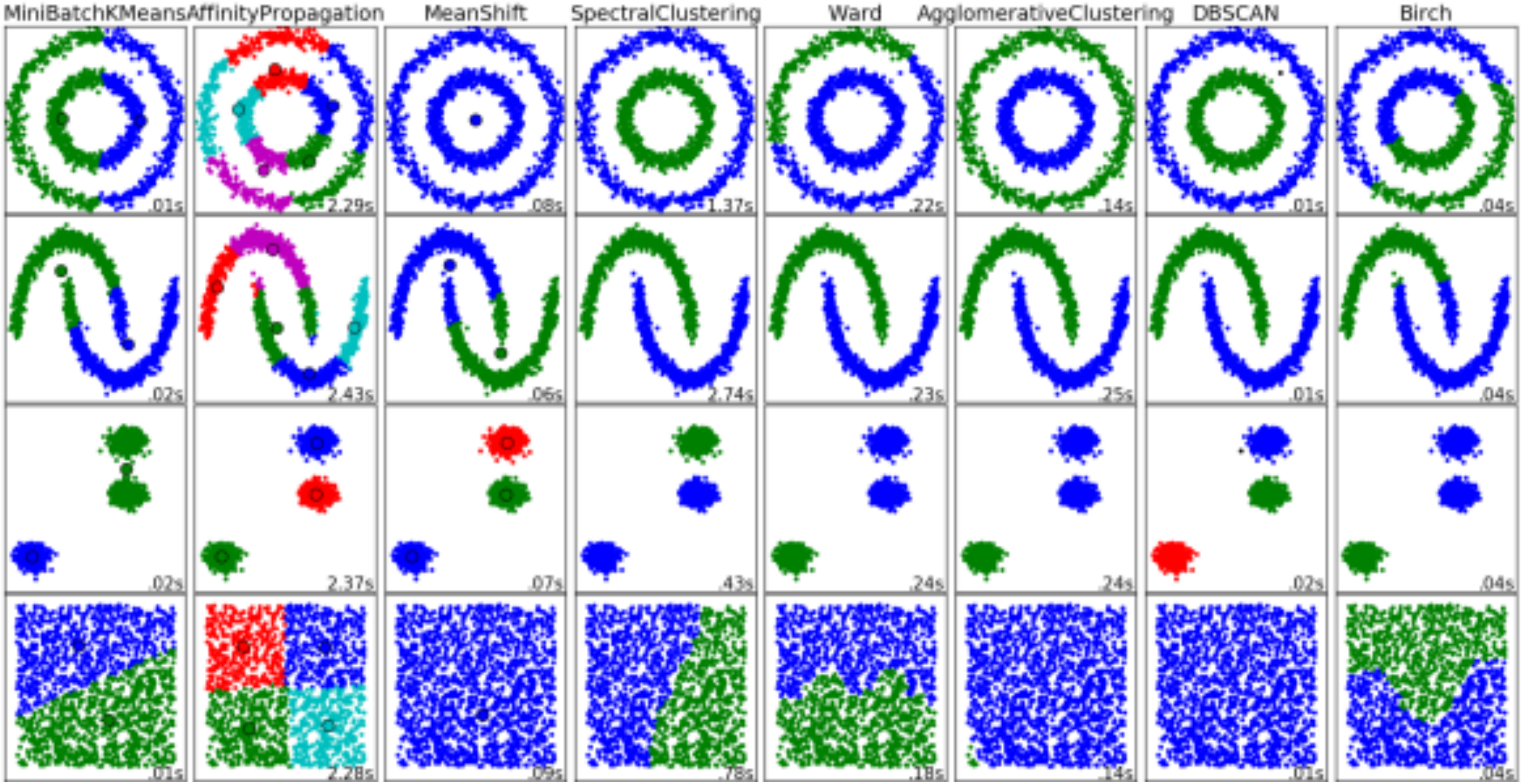
https://www.youtube.com/watch?v=mtkWR8sx0NA
https://www.youtube.com/watch?v=_aWzGGNrcic (especially from minute 4:22)
http://www.onmyphd.com/?p=k-means.clustering

OTHER CLUSTERING ALGORITHMS

VARIETY OF CLUSTERING OPTIONS



VARIETY OF CLUSTERING OPTIONS



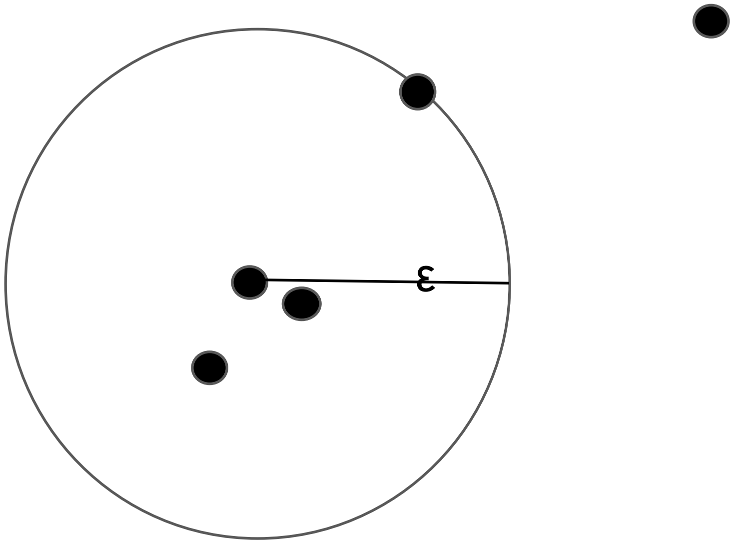
Density-Based Spatial Clustering of Applications with Noise (DBSCAN)

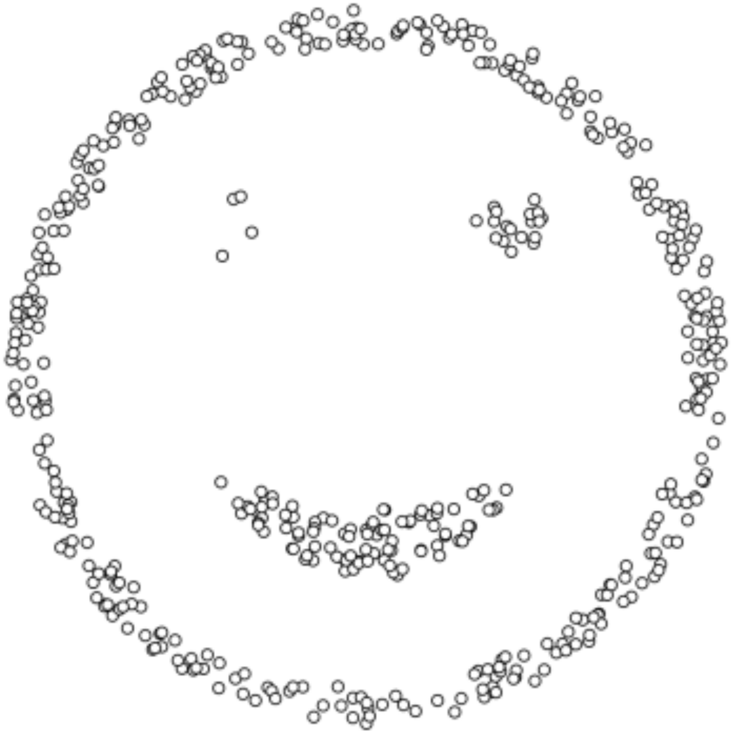
Criteria

- **ε (or Epsilon)** is the radius
- **minPoints** (number of points within the ε-Neighborhood required for classification)

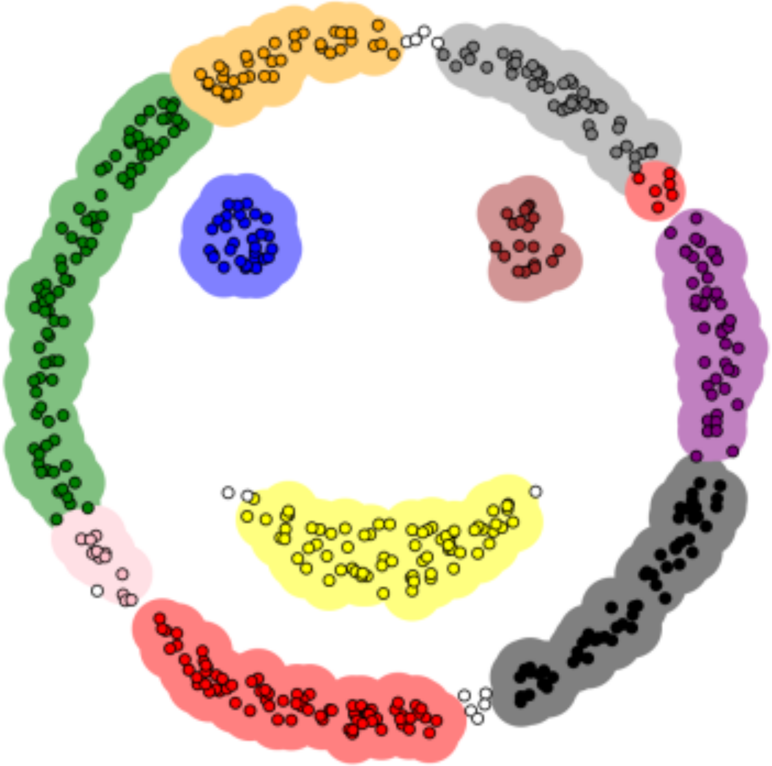
Note

- DBSCAN iterates through every point
- Core object (point meeting the criteria)
- Outlier (outside the radius)



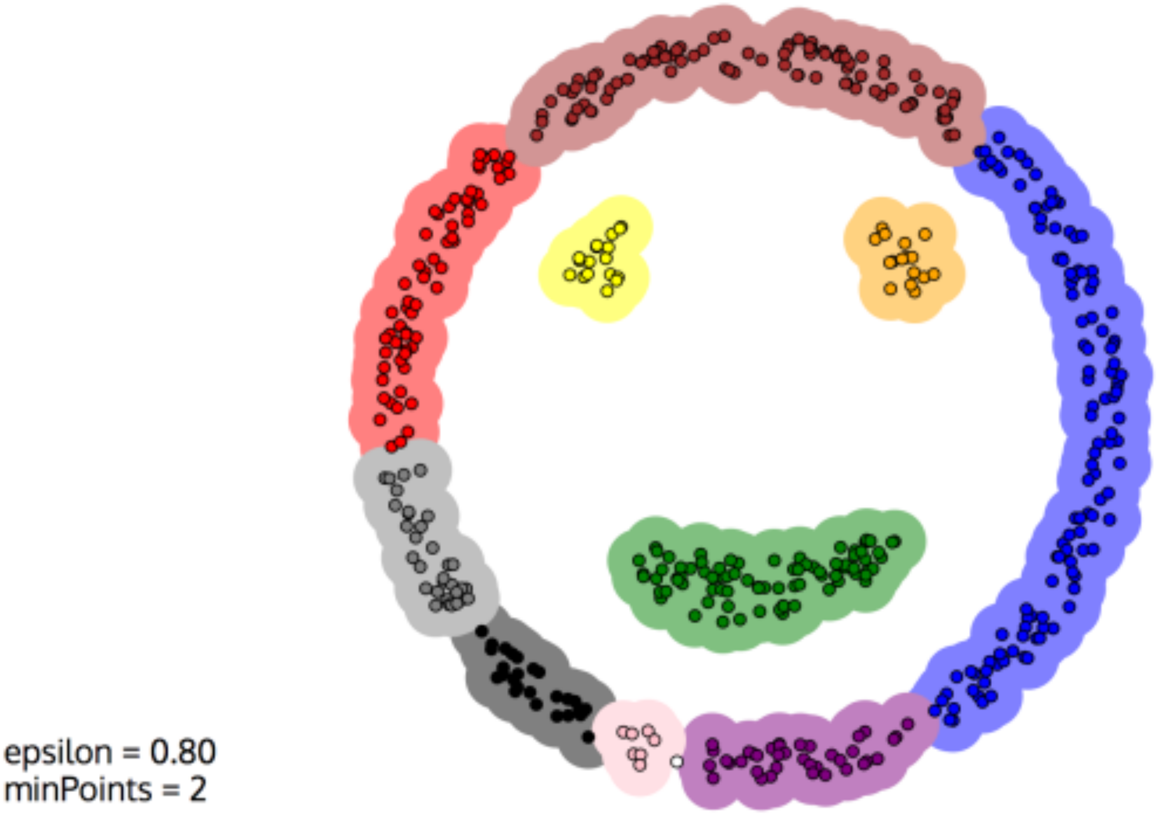


11 Clusters
Patchy

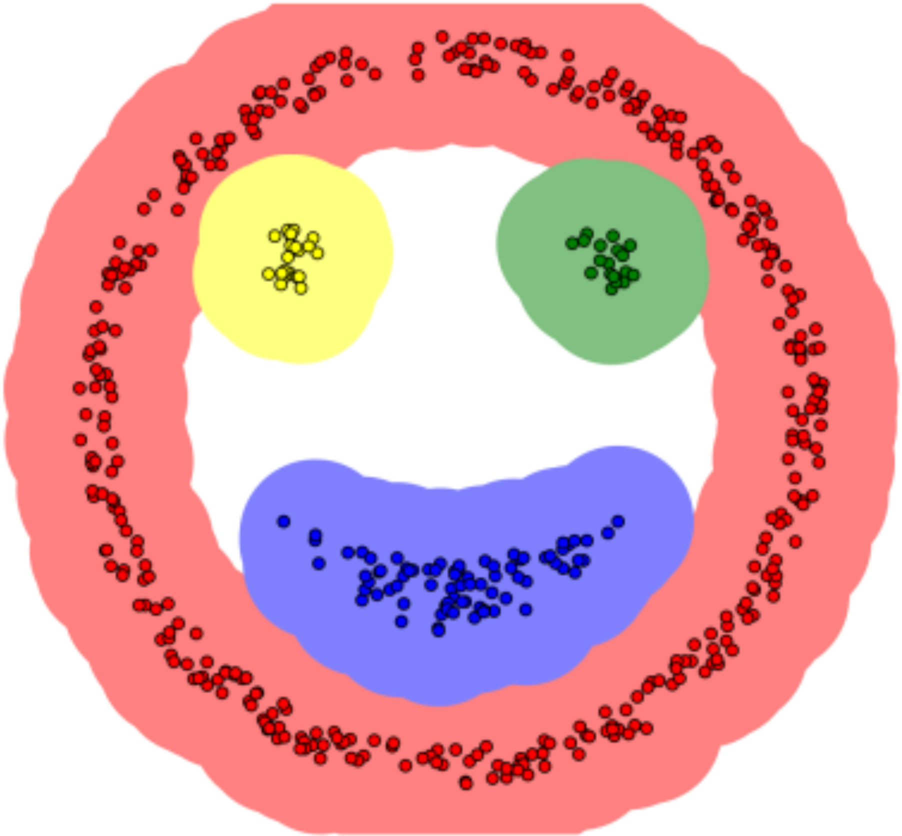


epsilon = 0.80
minPoints = 6

10 Clusters
Less Patchy



4 Clusters
Lion King



epsilon = 1.98
minPoints = 6

https://www.naftaliharris.com/blog/visualizing-dbscan-clustering/

Pros

Recovers more complex cluster shapes
Finds the number of clusters
Automatically find outliers

Cons

Requires a distance function
Not as scalable as K-means
Calculating connected components can be difficult

DATA SCIENCE PART TIME COURSE

HOW DO WE KNOW OUR CLUSTERS

In general, k-means will converge to a solution and return a partition of k clusters, even if no natural clusters exist in the data.

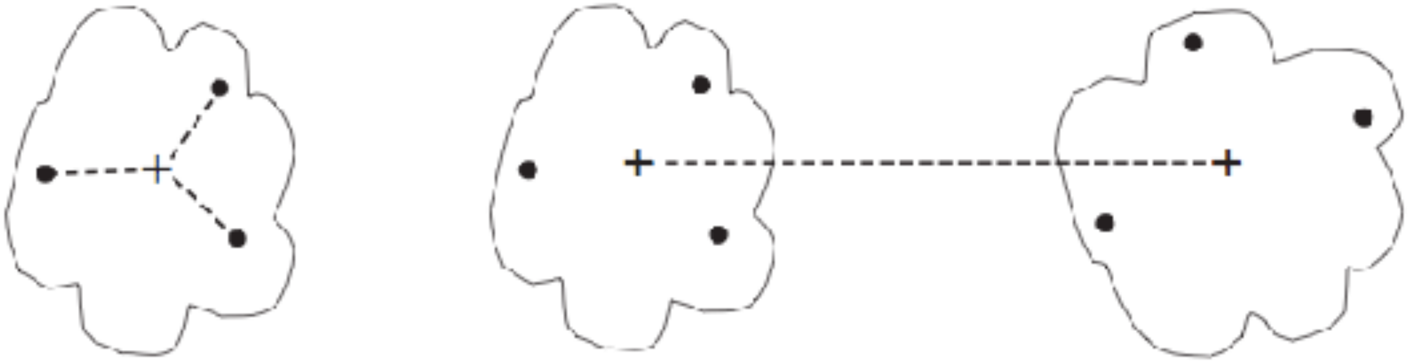
We will look at two validation metrics useful for partitional clustering, cohesion and separation.

Cohesion measures clustering effectiveness within a cluster.

$$\hat{C}(C_i) = \sum_{x \in C_i} d(x, c_i)$$

Separation measures clustering effectiveness between clusters.

$$\hat{S}(C_i, C_j) = d(c_i, c_j)$$



(a) Cohesion. (b) Separation.

Figure 8.28. Prototype-based view of cluster cohesion and separation.

source: http://www-users.cs.umn.edu/~kumar/dmbook/ch8.pdf

A useful measure that combines the ideas of cohesion and separation is the silhouette coefficient. For point x_i, this is given by:

$$SC_i = \frac{b_i - a_i}{max(a_i, b_i)}$$

such that:

a_i = average in-cluster distance to x_i

b_{ij} = average between-cluster distance to x_i

$b_i = min_j(b_{ij})$

The silhouette coefficient can take values between -1 and 1.

In general, we want **separation to be high** and **cohesion to be low**. This corresponds to a value of **SC close to +1.**

A negative silhouette coefficient means the cluster radius is larger than the space between clusters, and thus **clusters overlap**

One useful application of cluster validation is to determine the best number of clusters for your dataset.

Q: How would you do this?

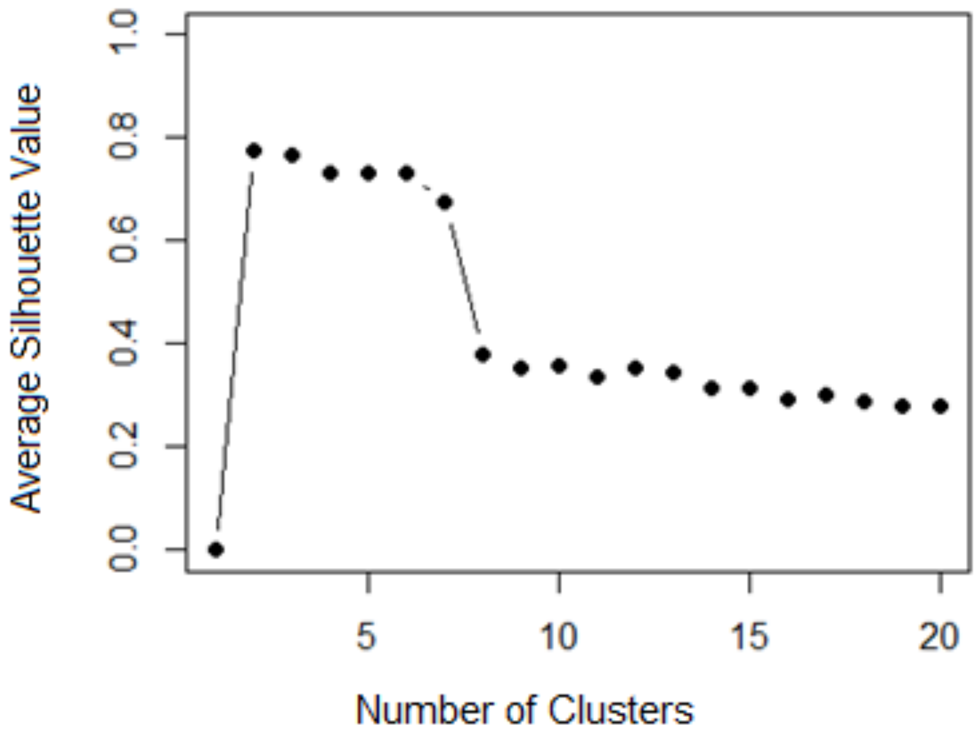
One useful application of cluster validation is to determine the best number of clusters for your dataset.

Q: How would you do this?

A: By computing the Silhouette Coefficient for different values of k.

Ultimately, cluster validation and clustering in general are subjective techniques that rely on human interpretation to be meaningful.

Art


Average Silhouette Value
0.0
0.2
0.4
0.6
0.8
1.0
5
10
15
20
Number of Clusters

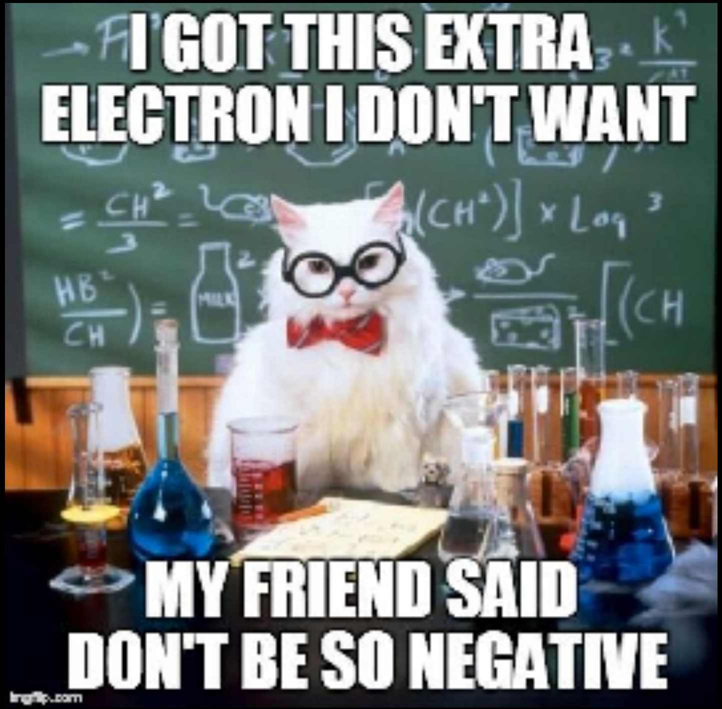
Strengths:

K-means is a popular algorithm because of its computational efficiency and simple and intuitive nature.

Weaknesses:

However, K-means is highly scale dependent, and is not suitable for data with widely varying shapes and densities.

LAB


I GOT THIS EXTRA
ELECTRON I DON'T WANT
MY FRIEND SAID
DON'T BE SO NEGATIVE
imgflip.com

FURTHER READING

Read the following

- Chapter 10.3 of Introduction to Statistical Learning - Clustering Methods in Introduction to Statistical Learning (15 pages)

…OR…

FURTHER READING

- PCA: https://youtu.be/Zbr5hyJNGCs, https://www.youtube.com/watch?v=cnCzY5M3txk
- Clustering (Siraj Raval): https://youtu.be/9991JlKnFmk
- Python Notebook on Clustering => link
- Pre Reading:
  - http://www.slideshare.net/MrChrisJohnson/algorithmic-music-recommendations-at-spotify
  - http://techblog.netflix.com/2012/04/netflix-recommendations-beyond-5-stars.html