University of Colorado **Boulder**

Unsupervised Learning

Geena Kim

Clustering

Clustering

- PCA: finds a low-dimensional representation

- Clustering: finds subgroups among observations

What Clustering is for

- Get a meaningful intuition of the structure of the data

- Cluster-then-predict

(ex) clustering patients into different subgroups and build a model for each subgroup to predict the probability of the risk of having certain disease

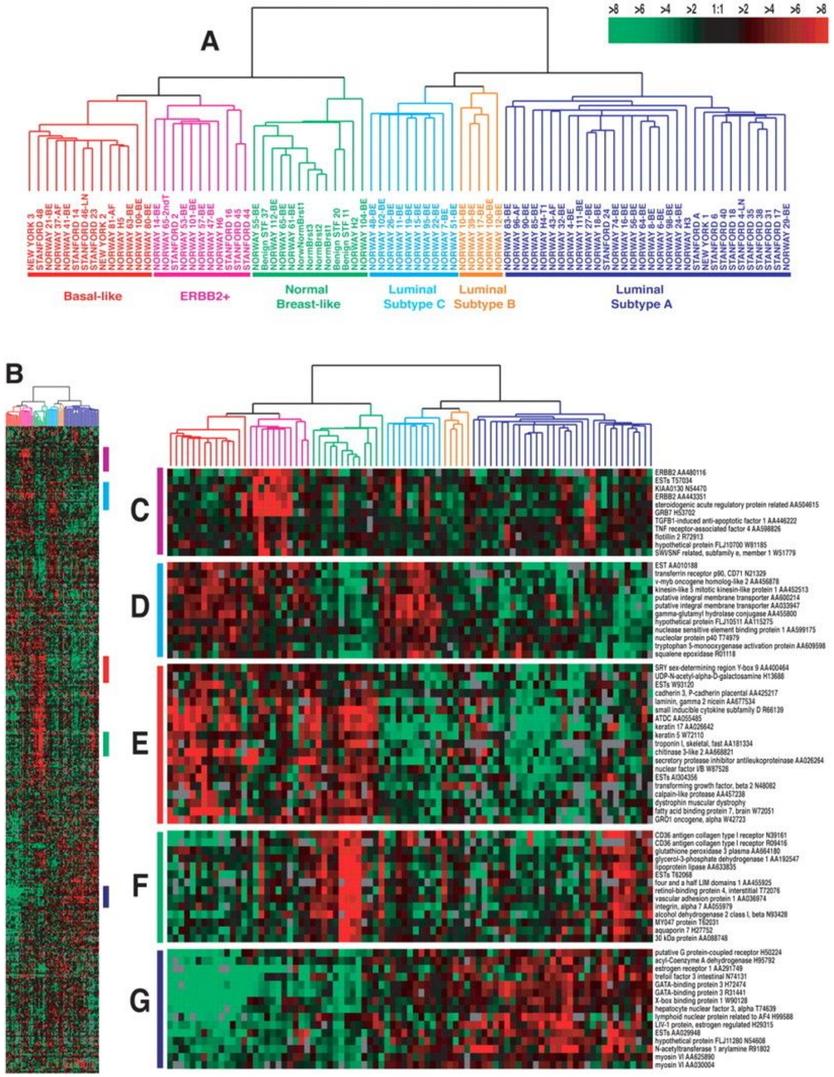
Clustering Applications

- Marketing and sales
  - Customer segmentation: identifying subgroups of people who might like to purchase particular types of products
  - Advertising: identifying subgroups of people who might respond to particular types of advertising

Clustering Applications

- Disease subtypes discovery
- Genomic research

Ahlqvist et al. (2018)

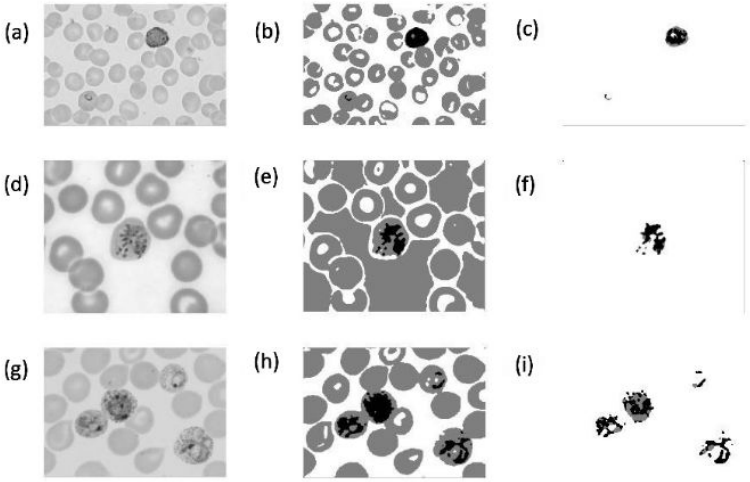


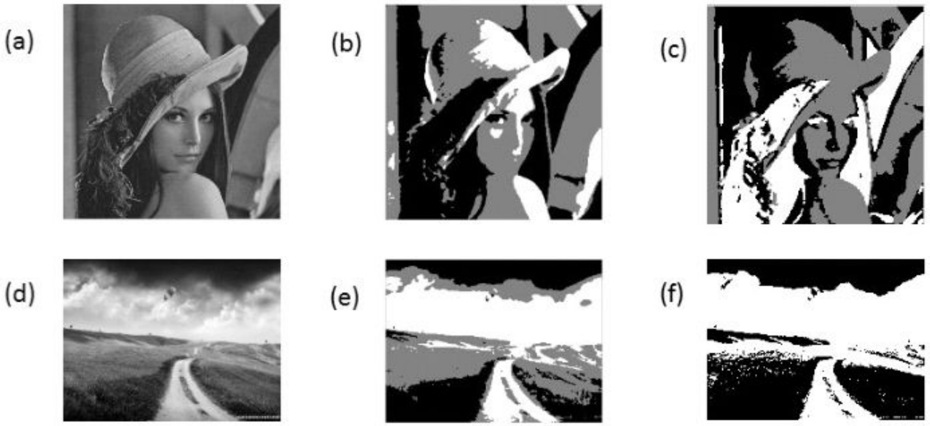
T. Sørlie et al (2001)

Clustering Applications

- Document clustering
  - identifying documents (or movies/music) that are similar

- Image segmentation or preprocessing





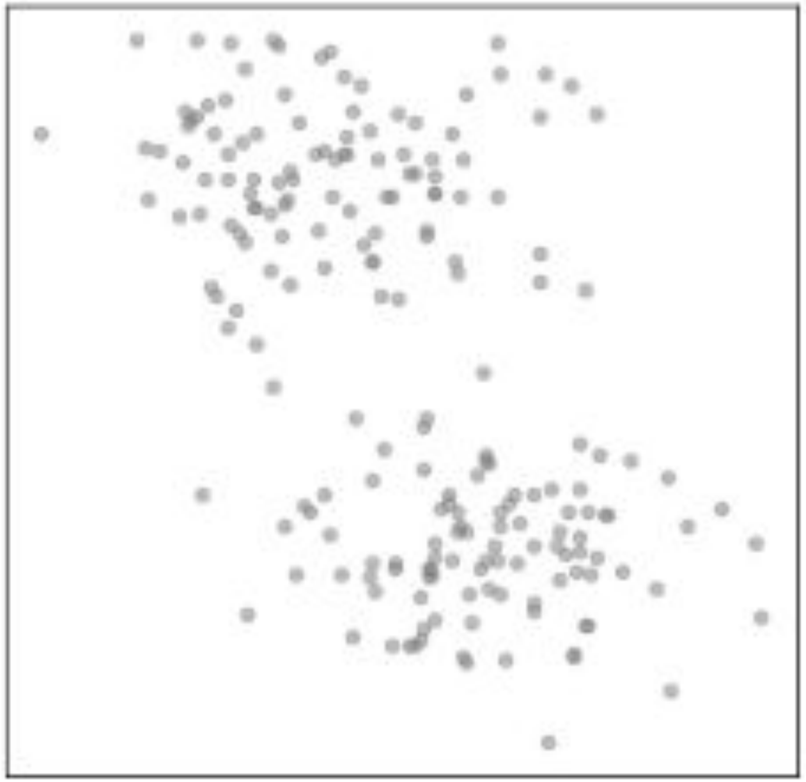
Dhanachandra et al., (2015)

Popular Clustering Methods

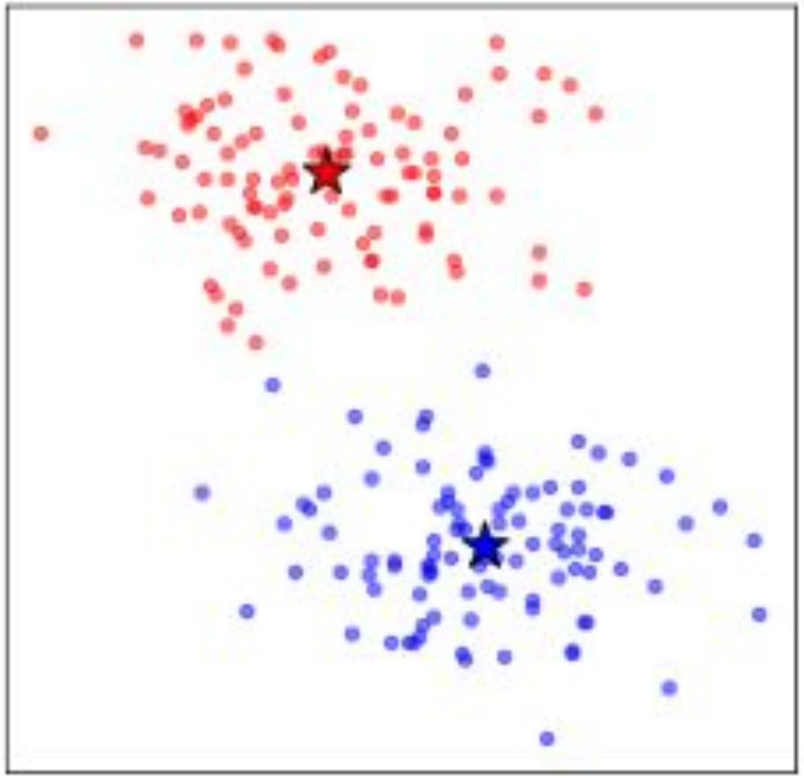
- K-means clustering
- hierarchical clustering

K-means Clustering

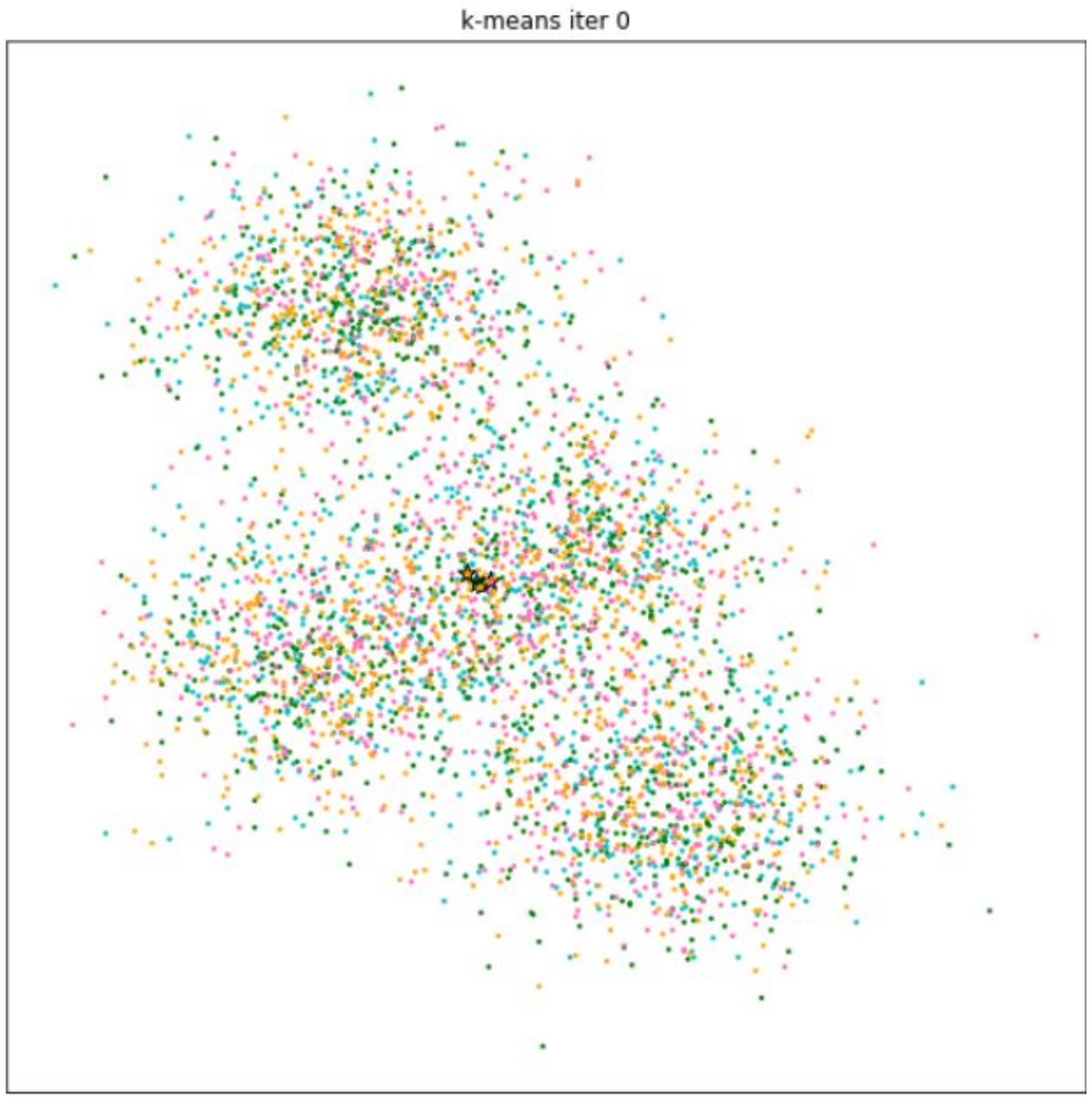
What is a cluster?



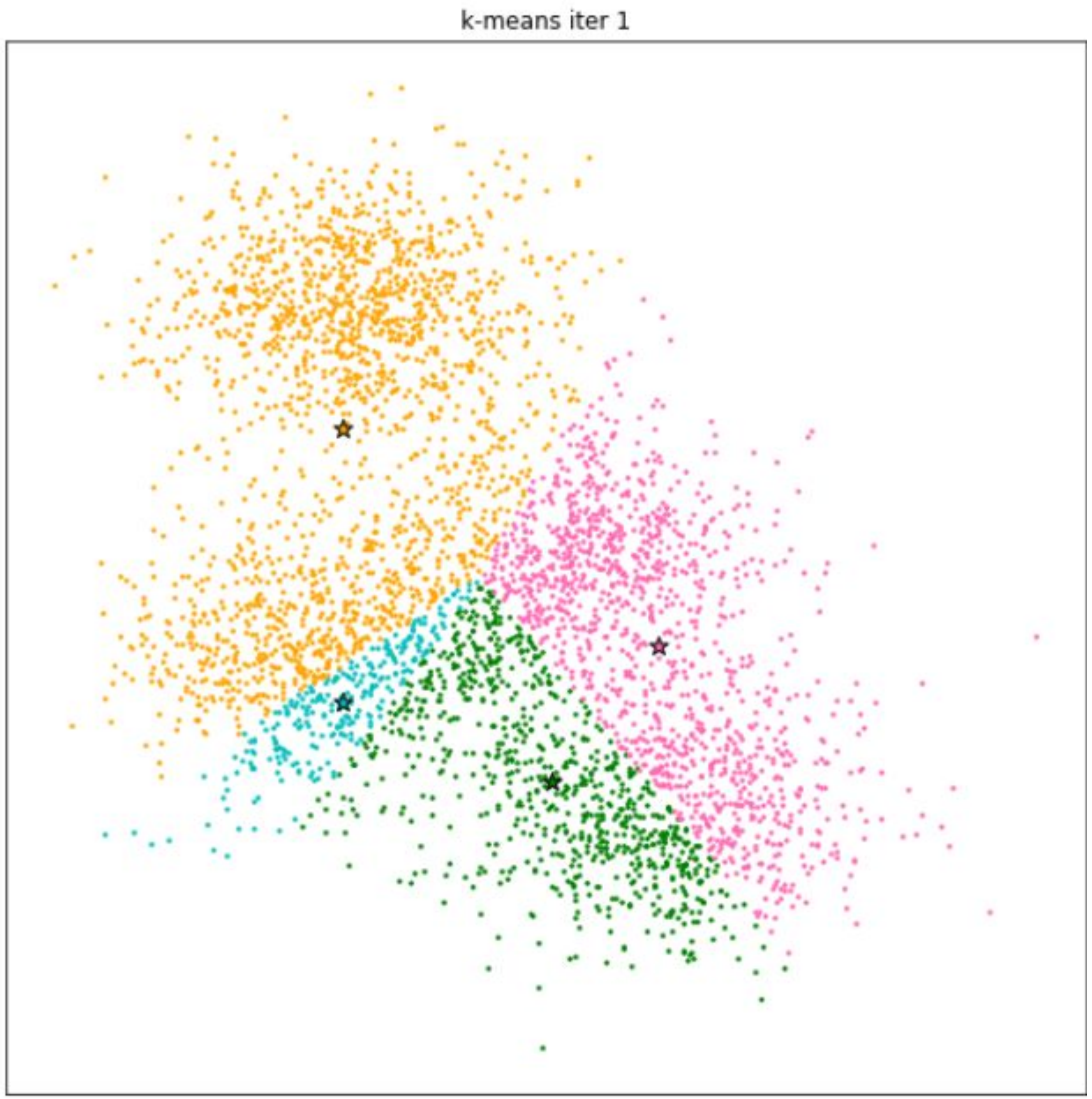
What is a centroid?



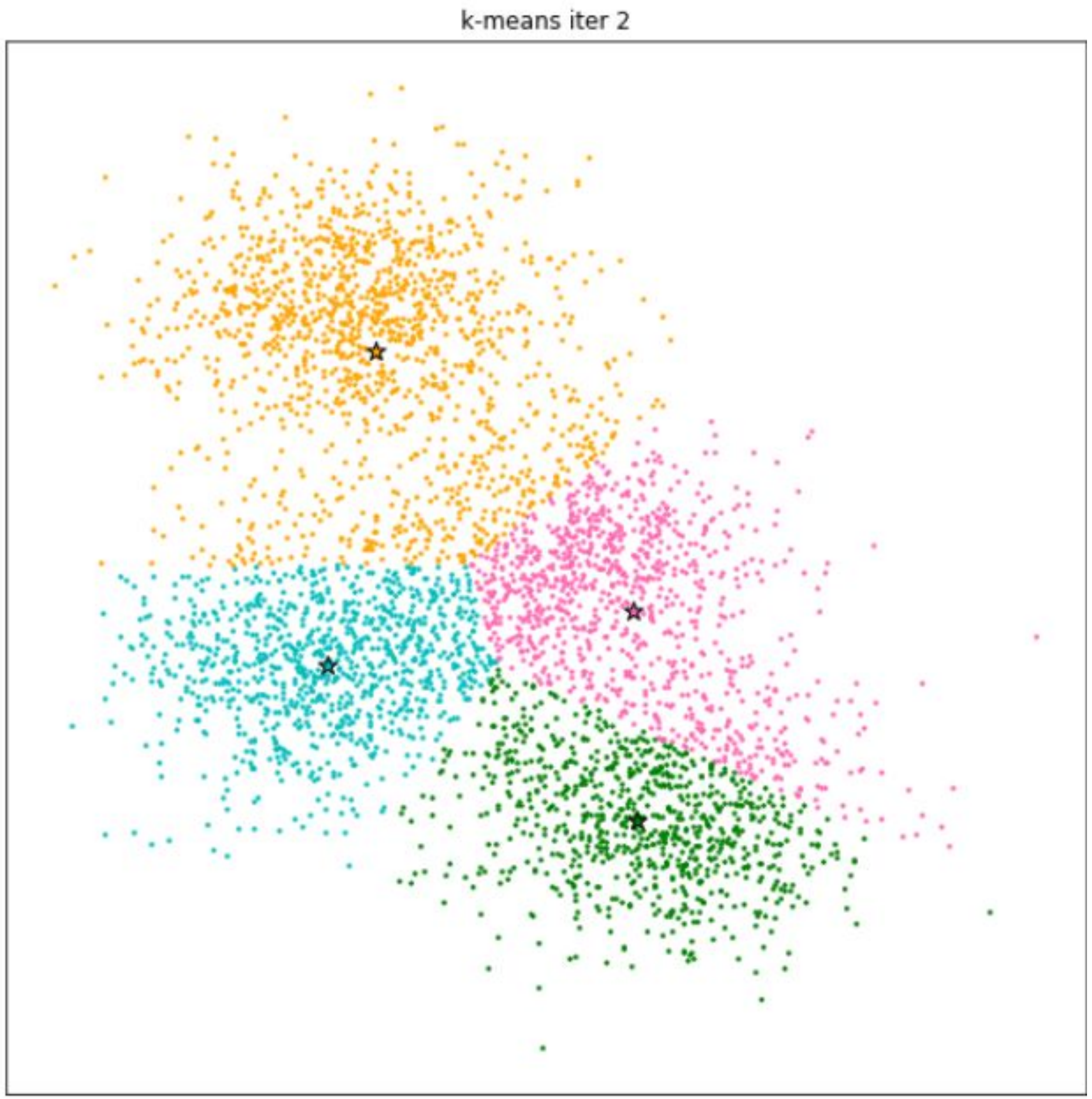
K-means Algorithm



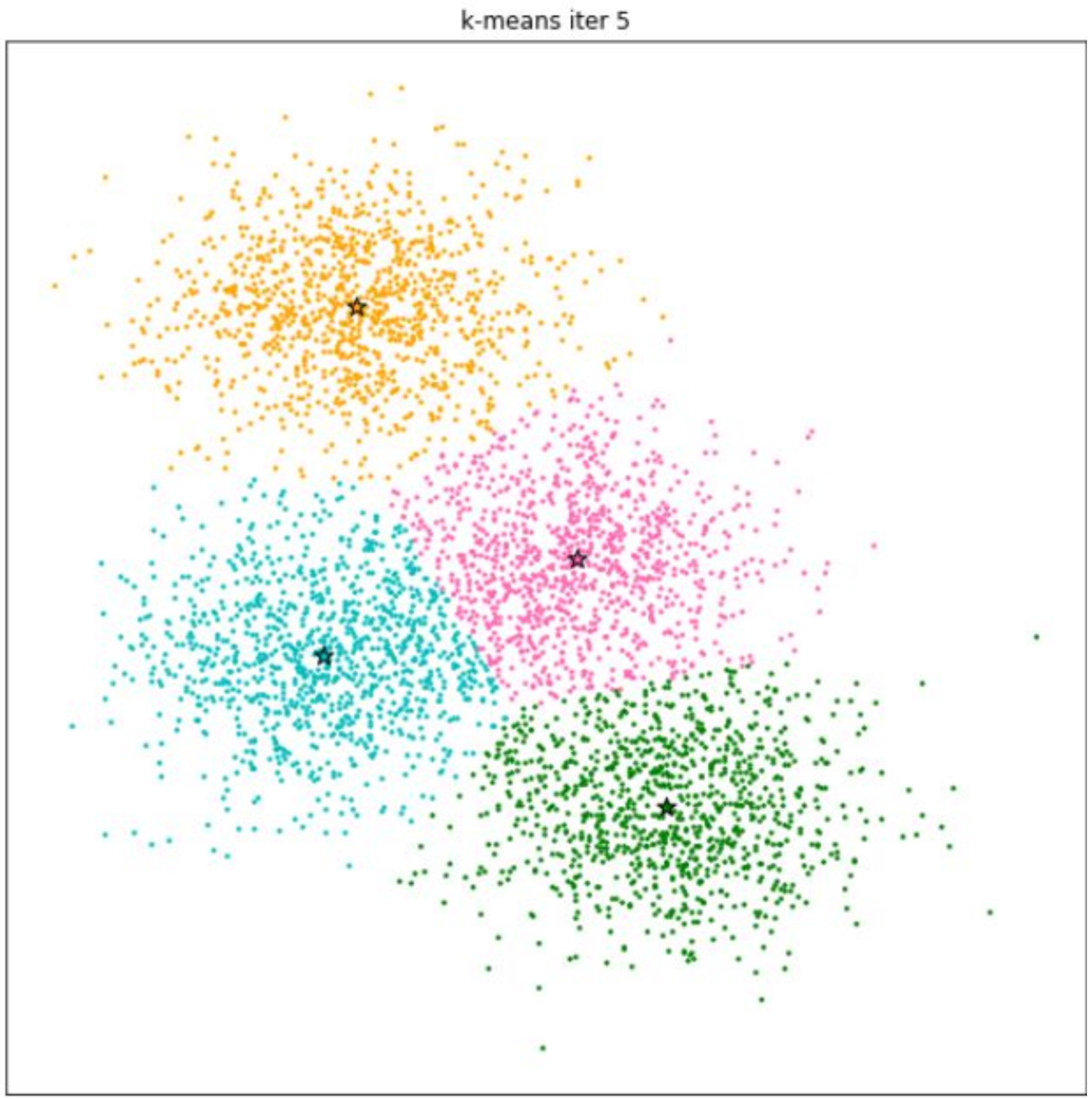
K-means Algorithm



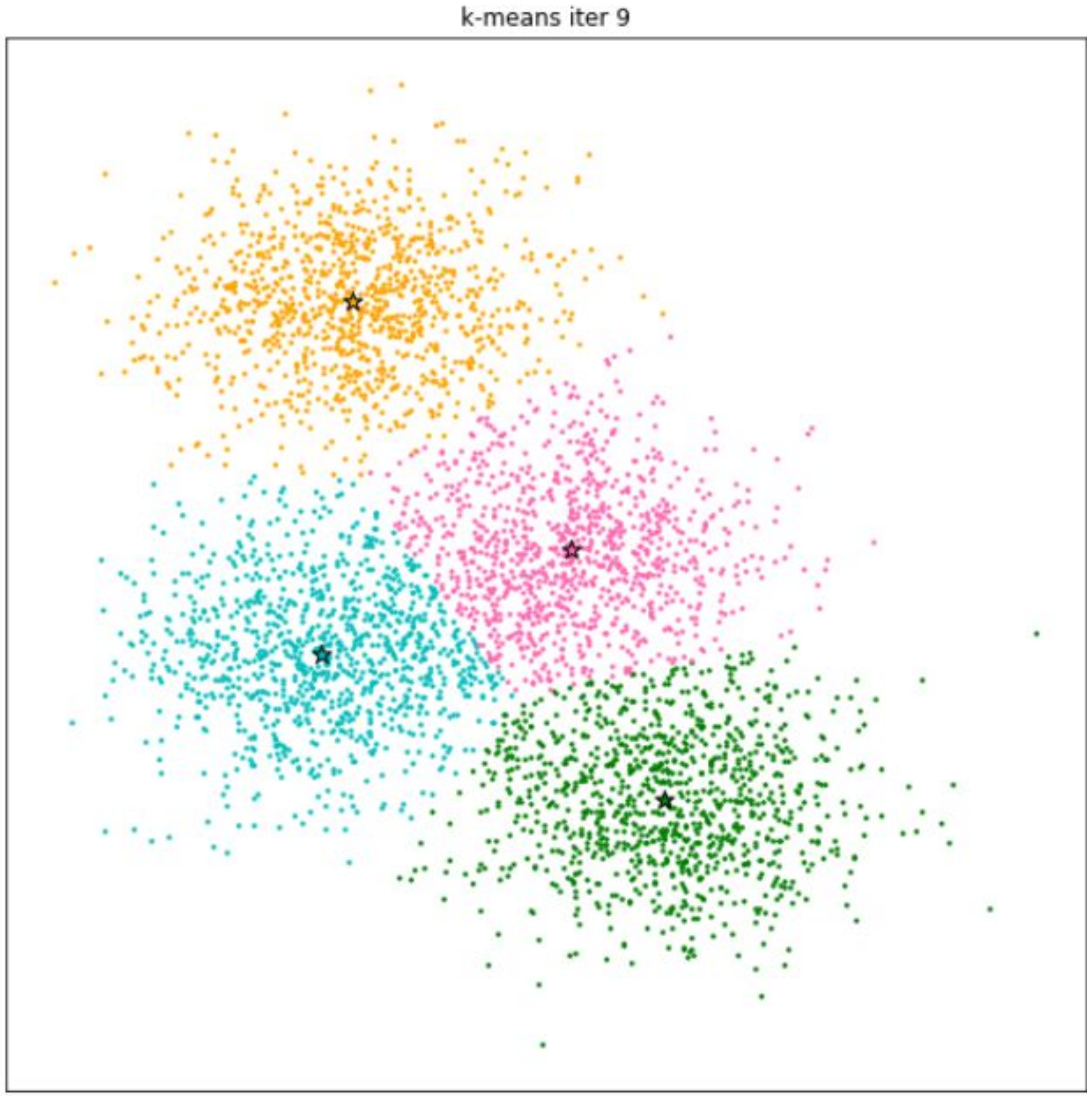
K-means Algorithm



K-means Algorithm



K-means Algorithm



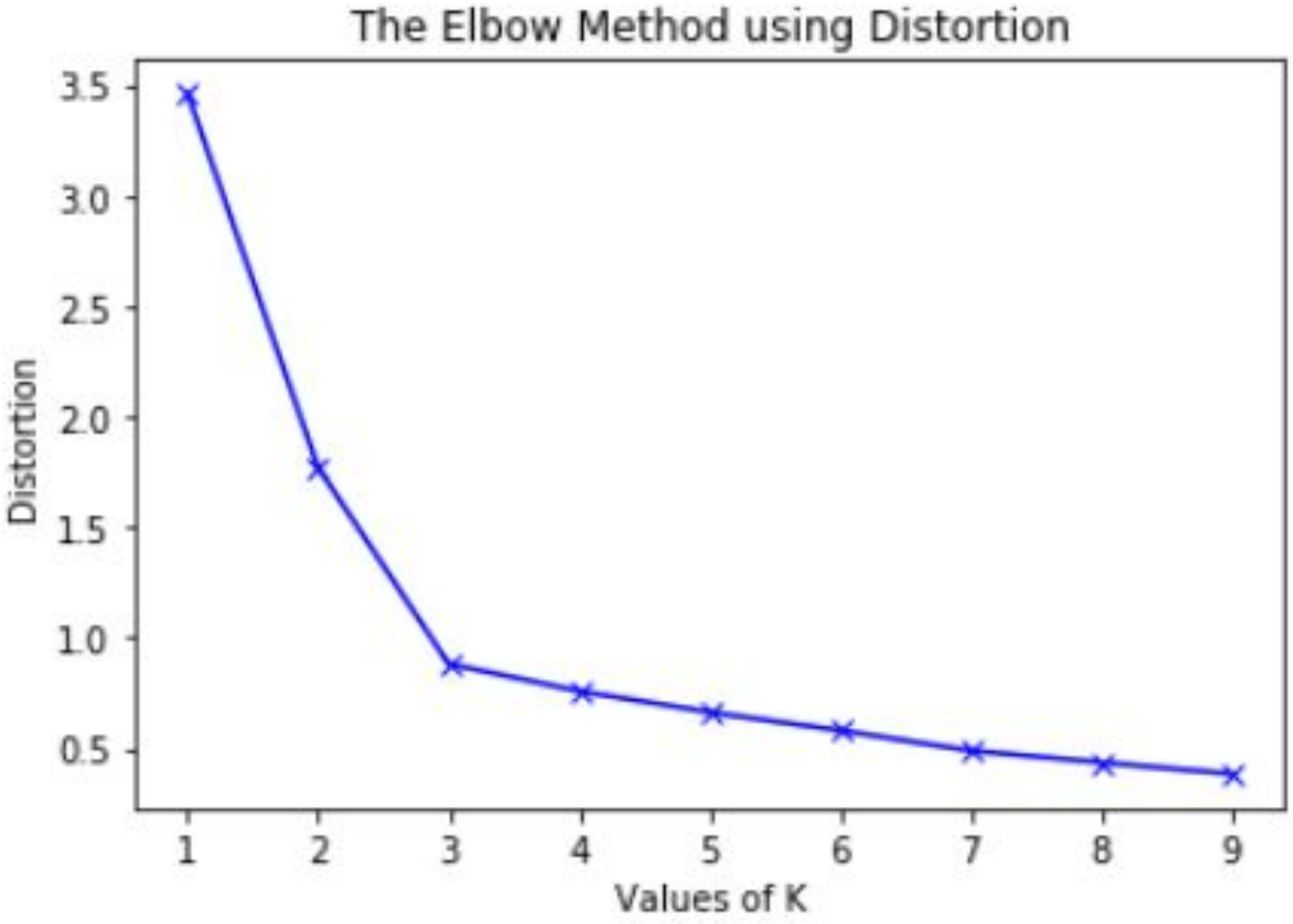
Distance metrics

Metrics

Distortion (the mean of square distance)

Inertia (the sum of square distance)

How to choose K?



K-means Clustering

Need to decide how many clusters (K) before trying

Vulnerable to curse of dimensionality   PCA preprocessing helps

Given enough time, K-means will always converge

Finds local minimum, not global minimum

The local minimum is highly dependent on the initialization

sklearn's Kmeans (`sklearn.cluster.KMeans`) can initialize better if `init='k-means++'` is used

MiniBatchKmeans uses mini-batches to reduce the computation time

Clustering-continued

Hierarchical Clustering Properties

It does not need to know K in advance!

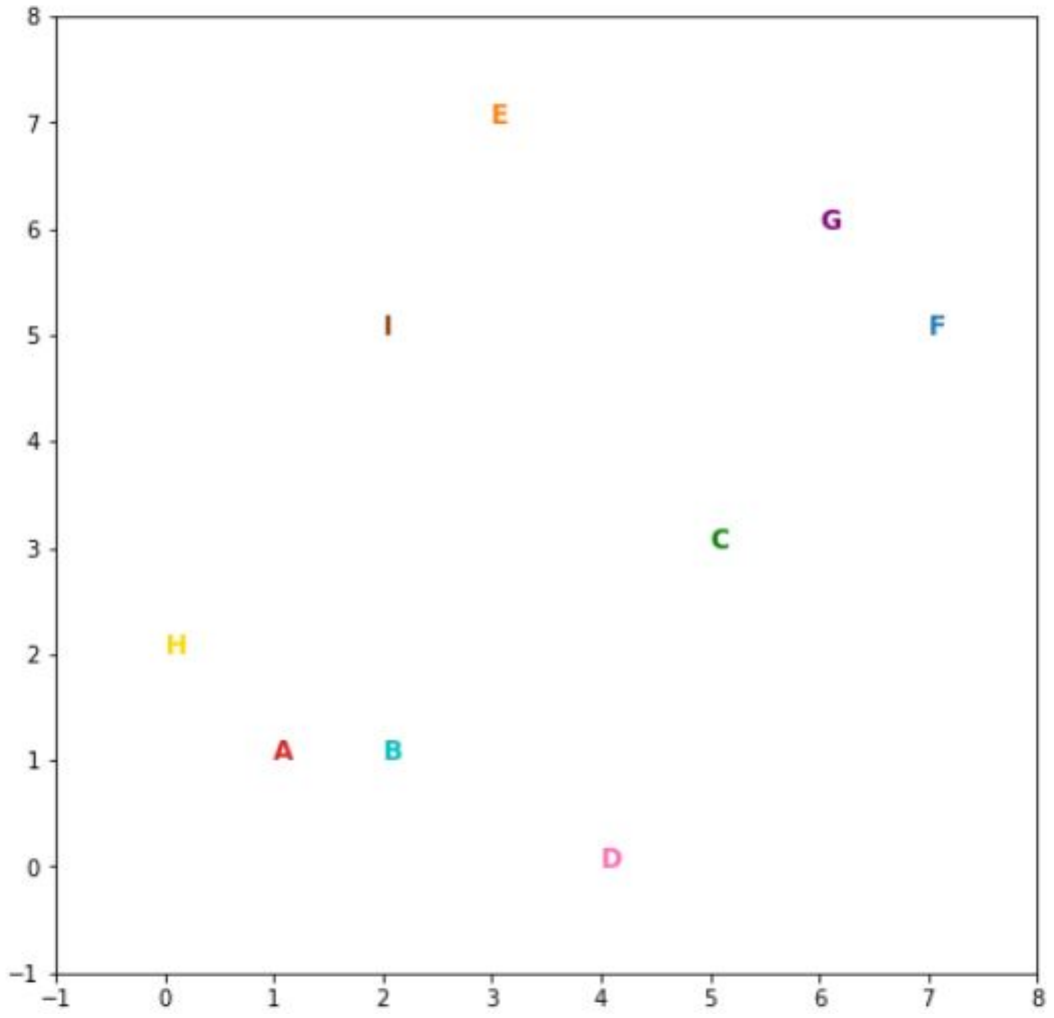
Use (di)similarity or distance metric

Use Dendrogram (upside down tree)

Deterministic (Reproducible)

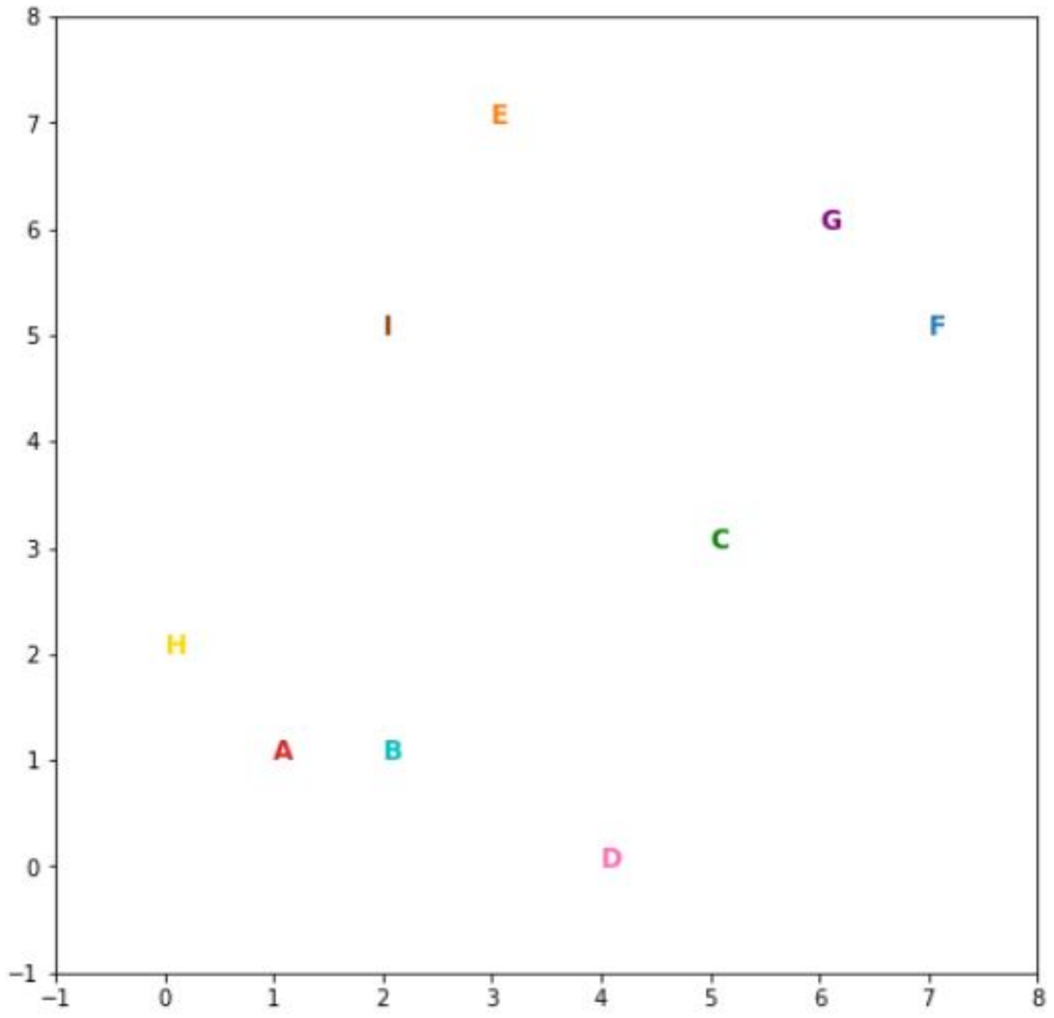
Greedy (local solutions)

Hierarchical Clustering Algorithm



- Measure the (distance) metric among all cluster points
- Merge the closest clusters
- Determine the new cluster's representative point

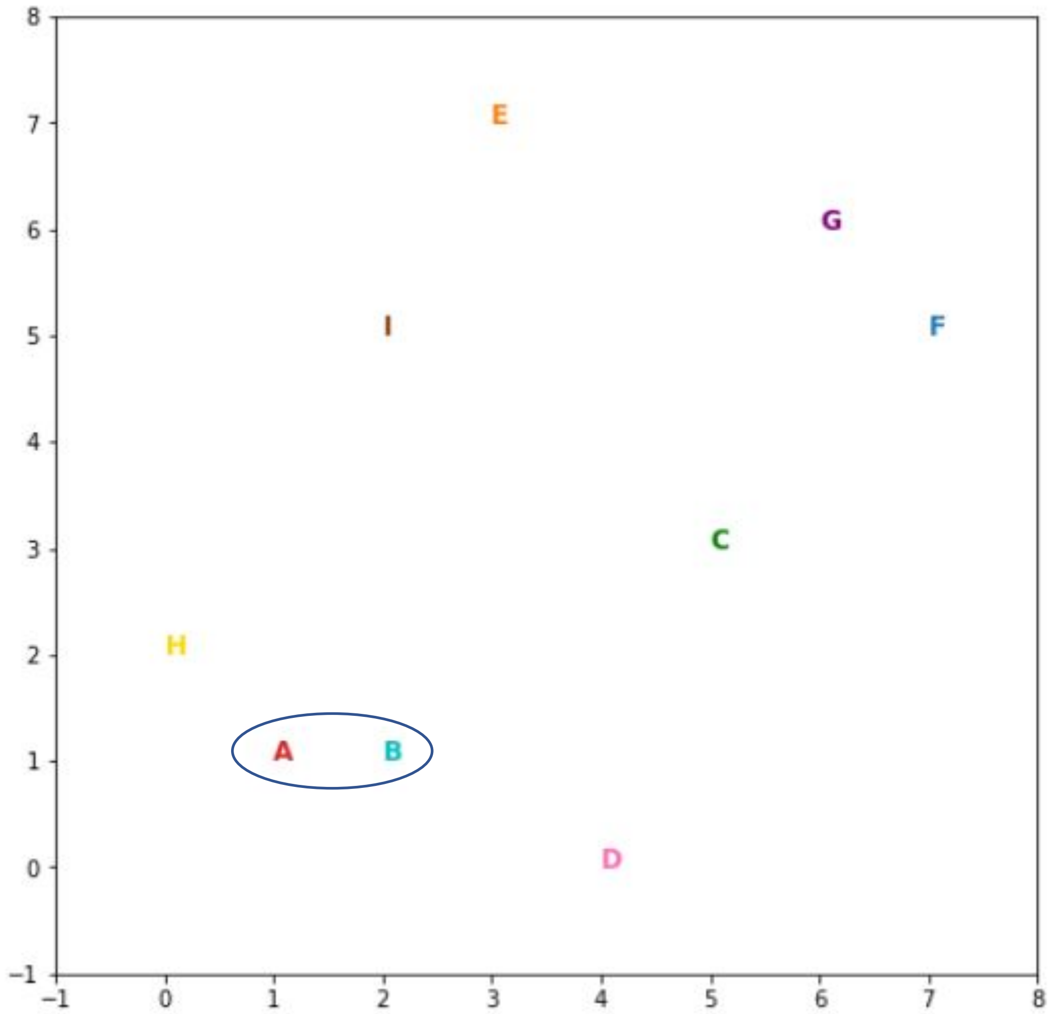
Hierarchical Clustering Algorithm



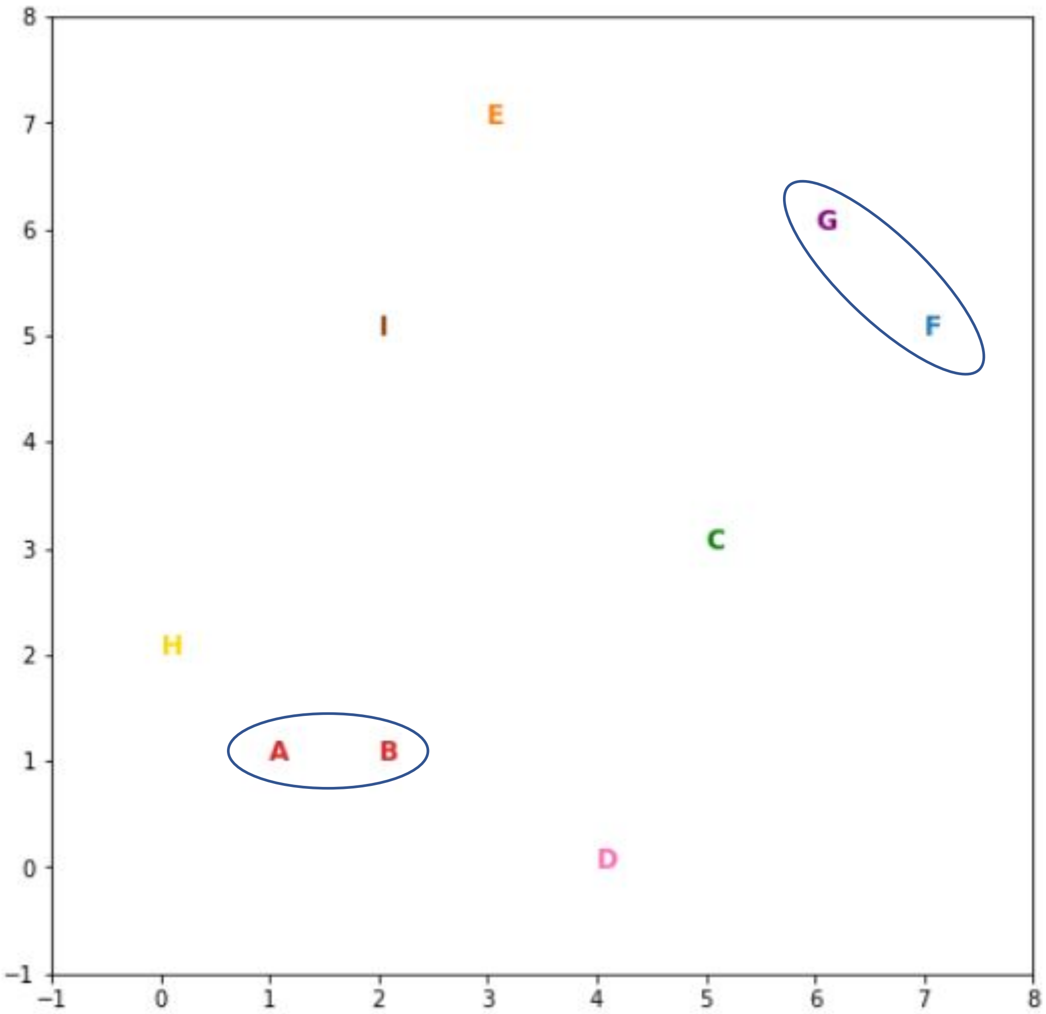
Choice of linkage type (metric) matters!

- Complete
- Single
- Average
- Centroid
- Ward

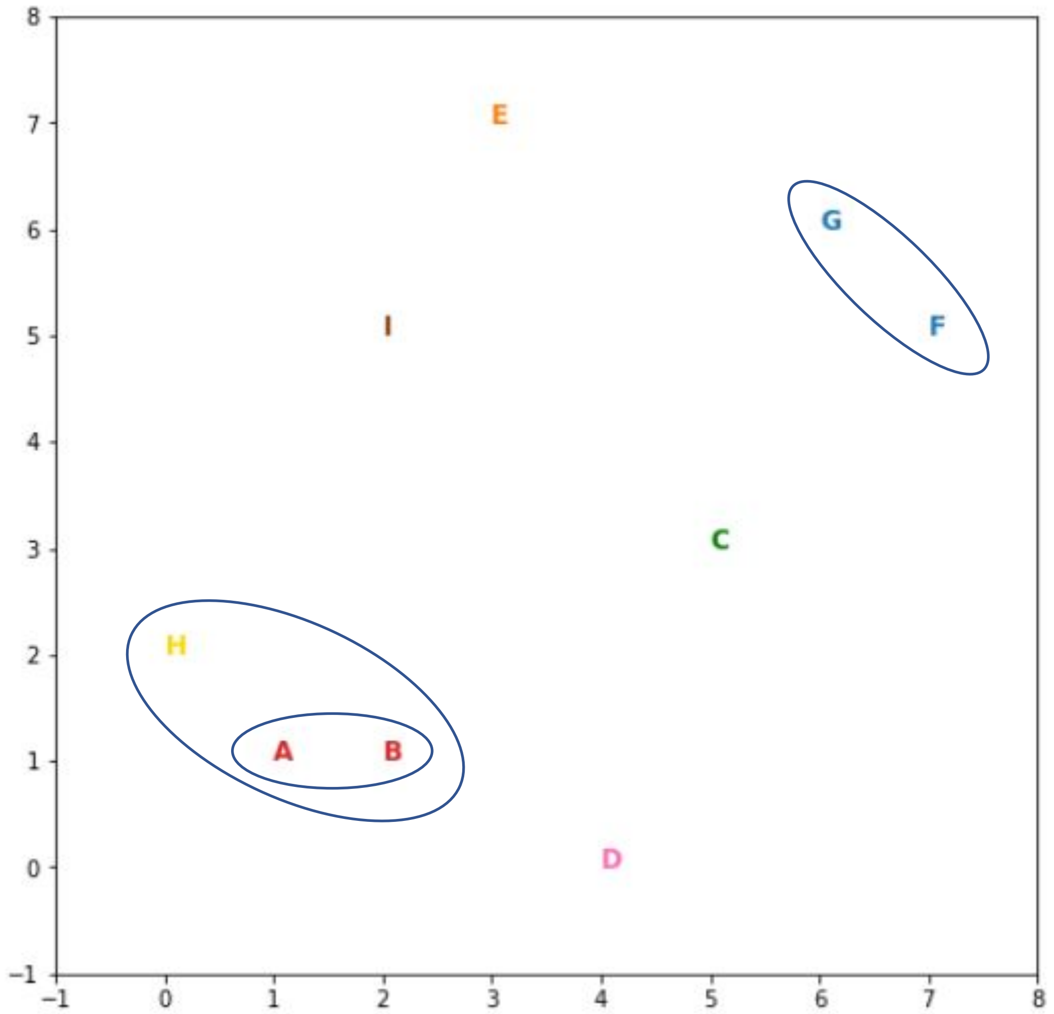
Complete Linkage

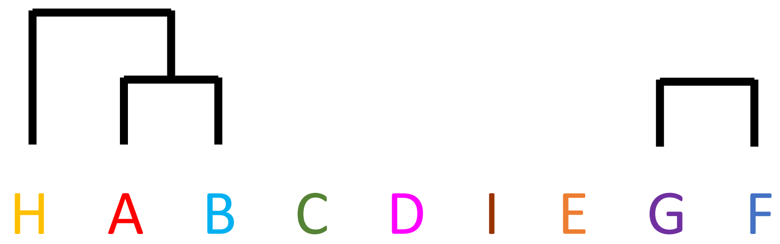


Complete Linkage

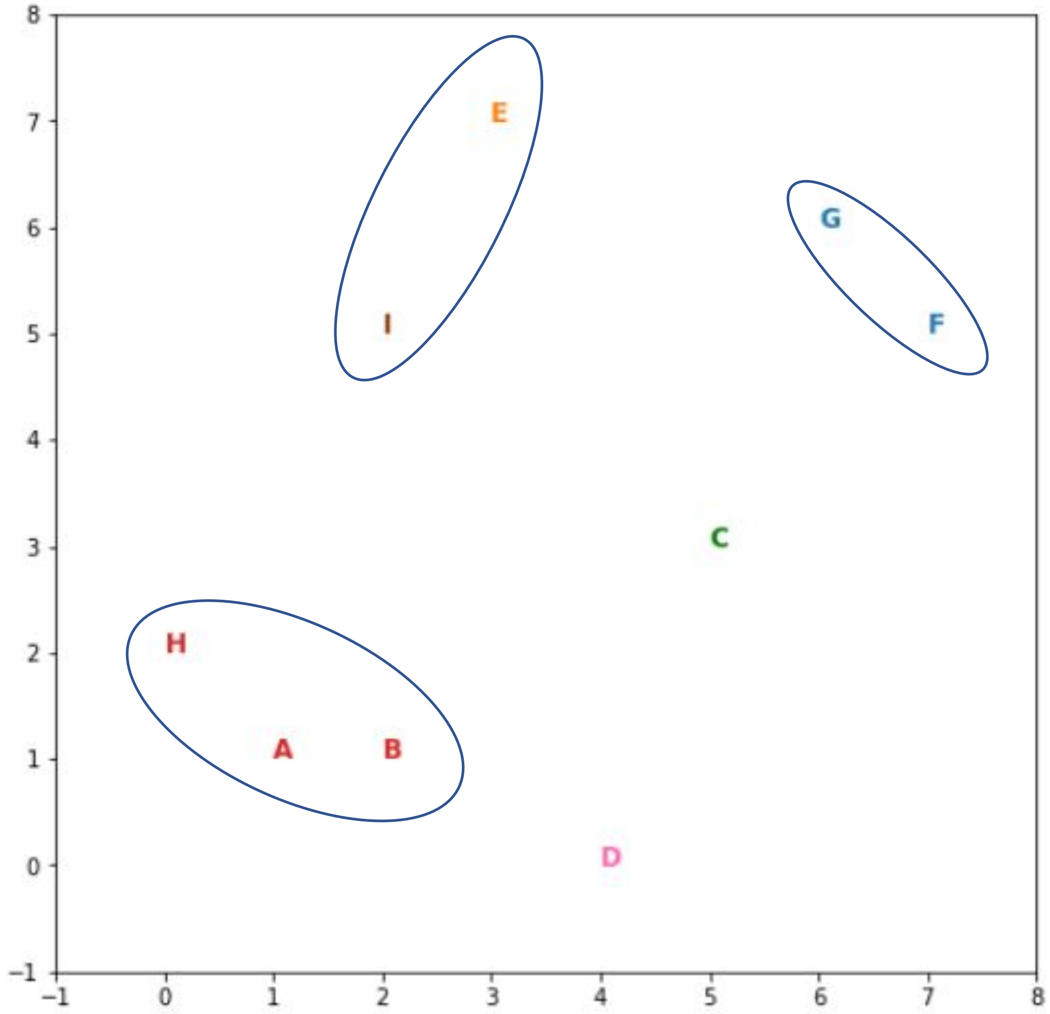


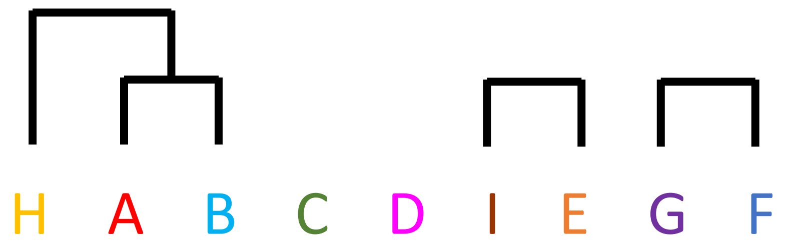
Complete Linkage



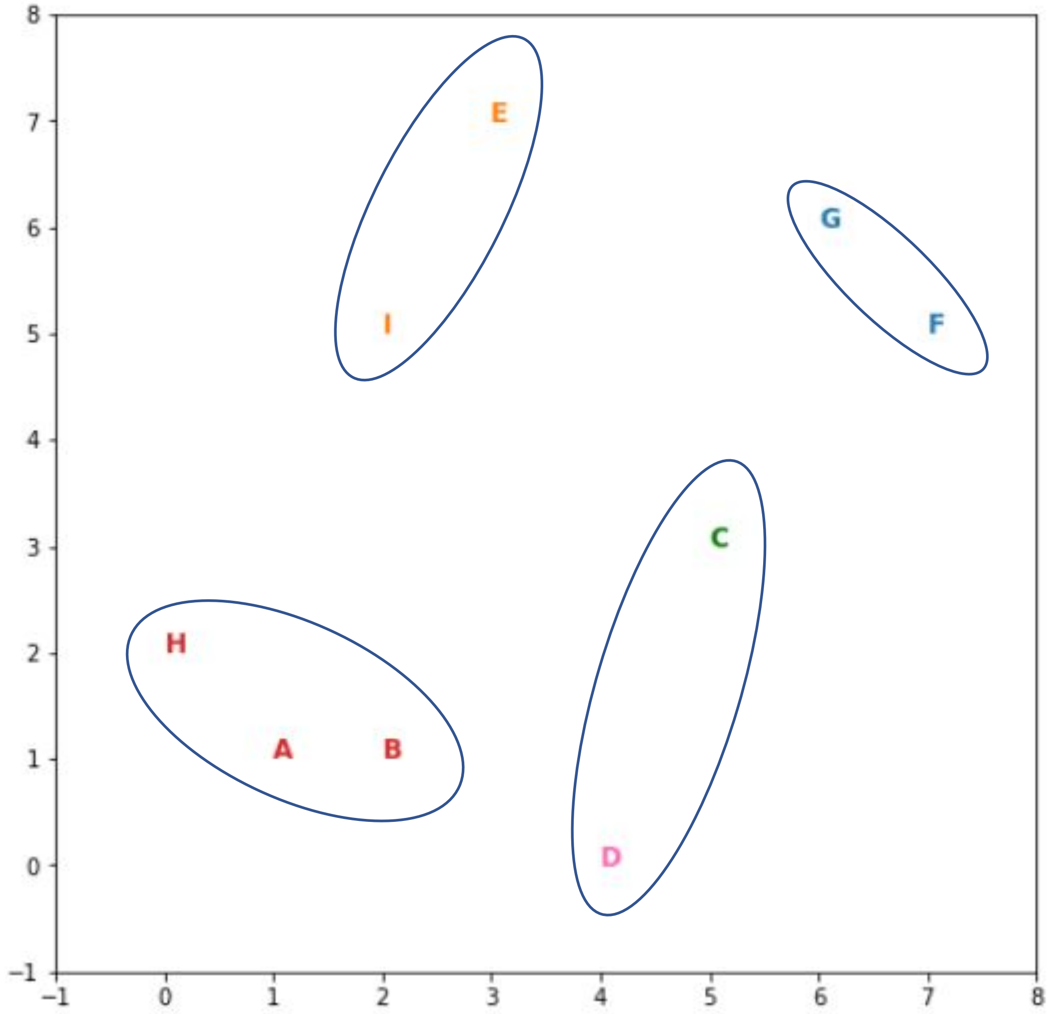


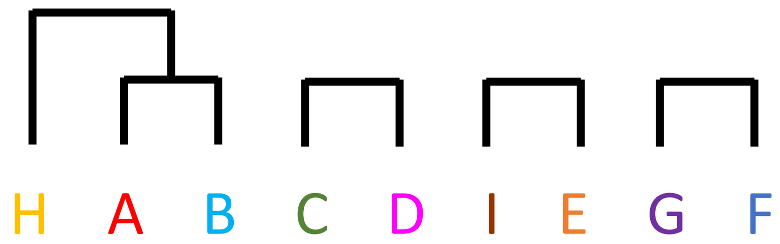
Complete Linkage



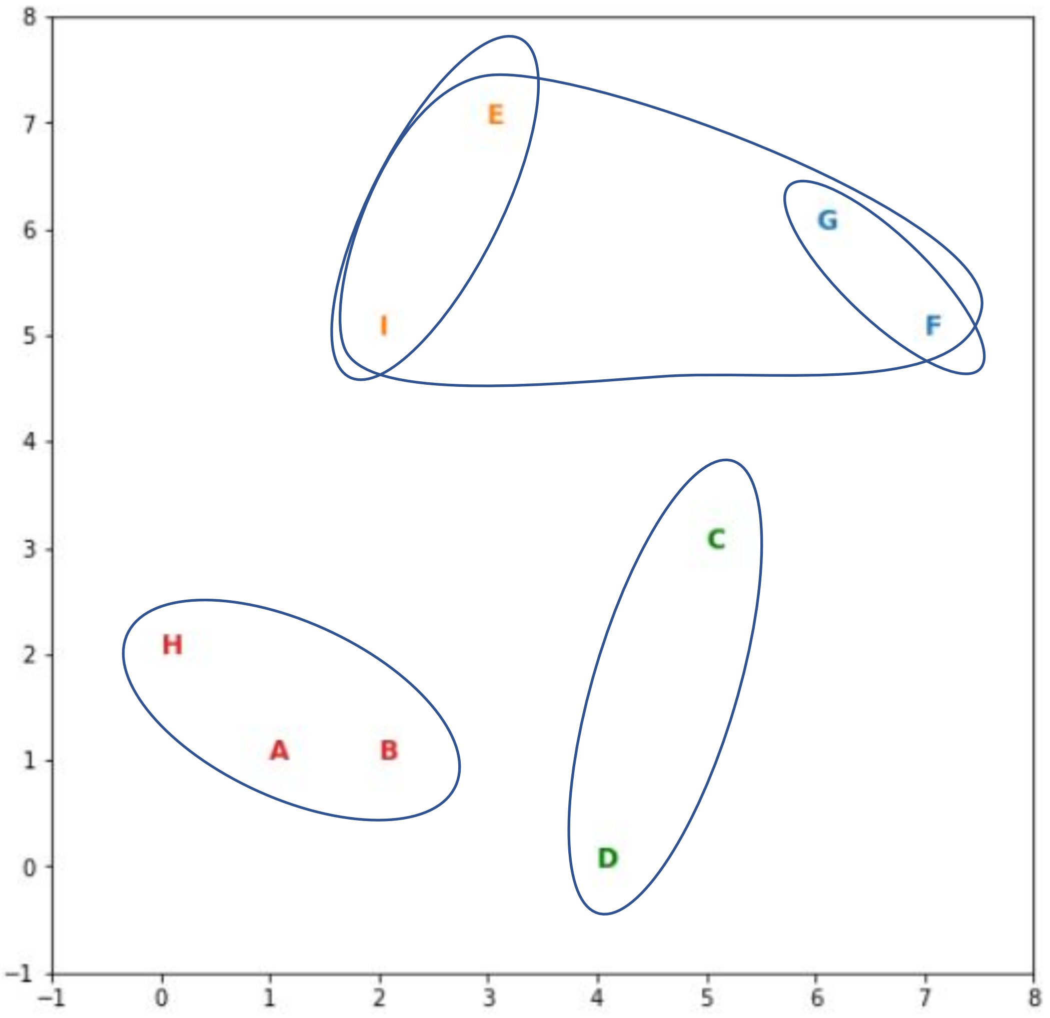


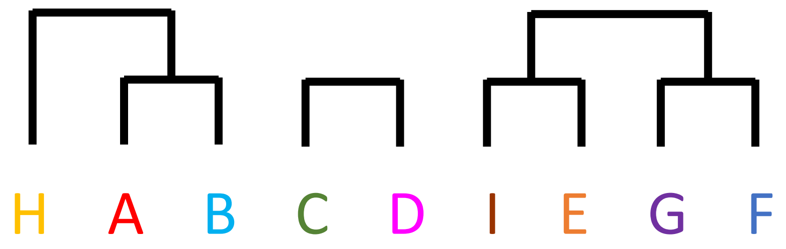
Complete Linkage



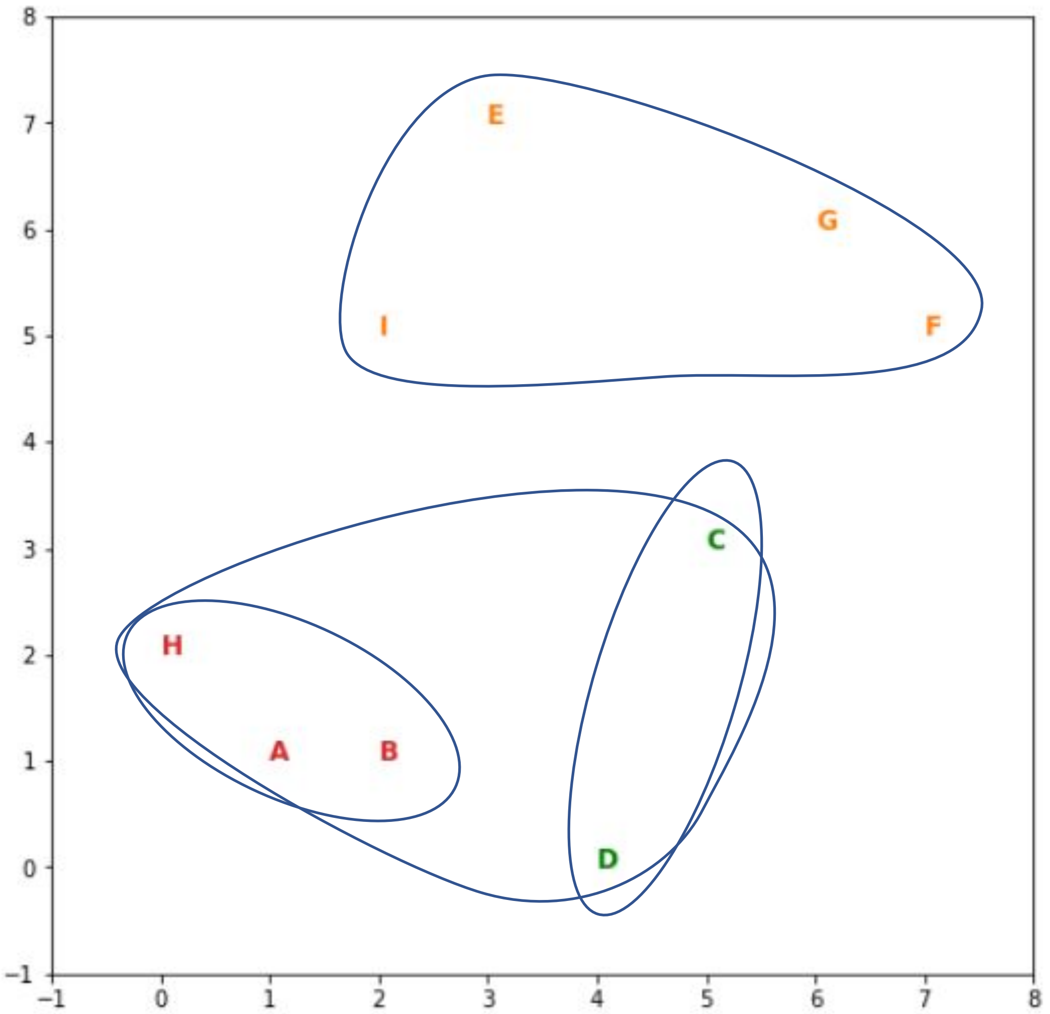


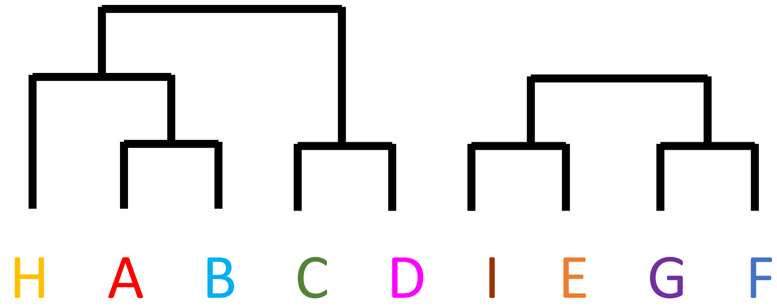
Complete Linkage



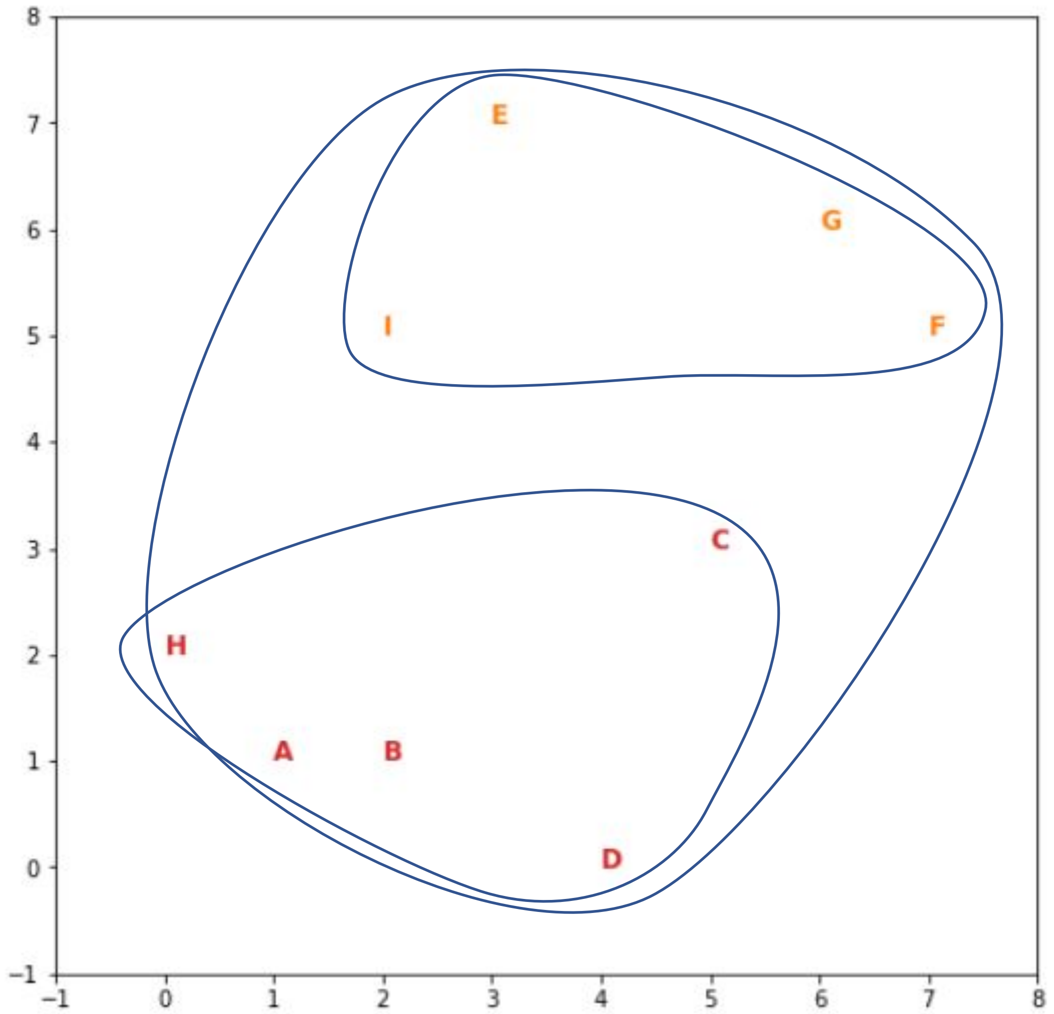


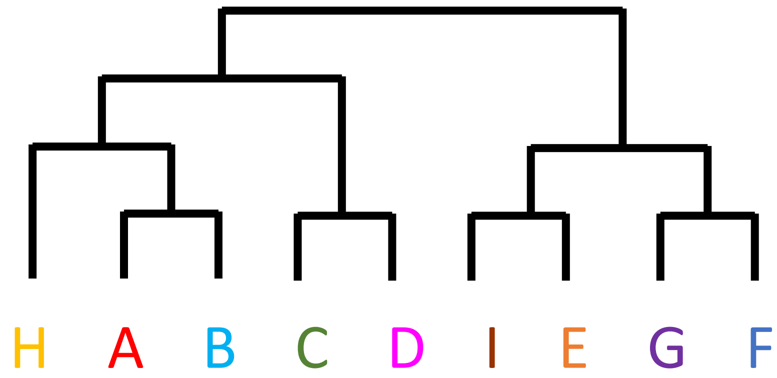
Complete Linkage



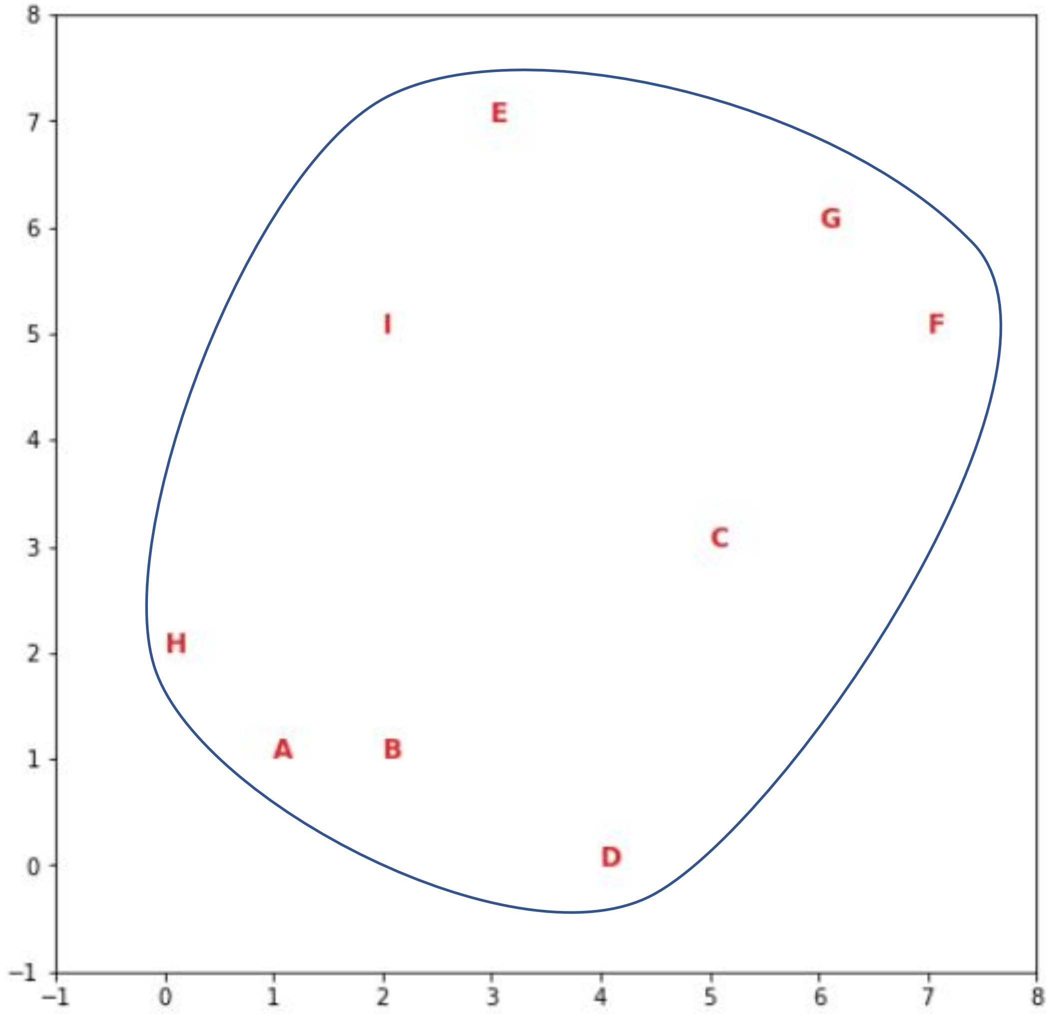


Complete Linkage



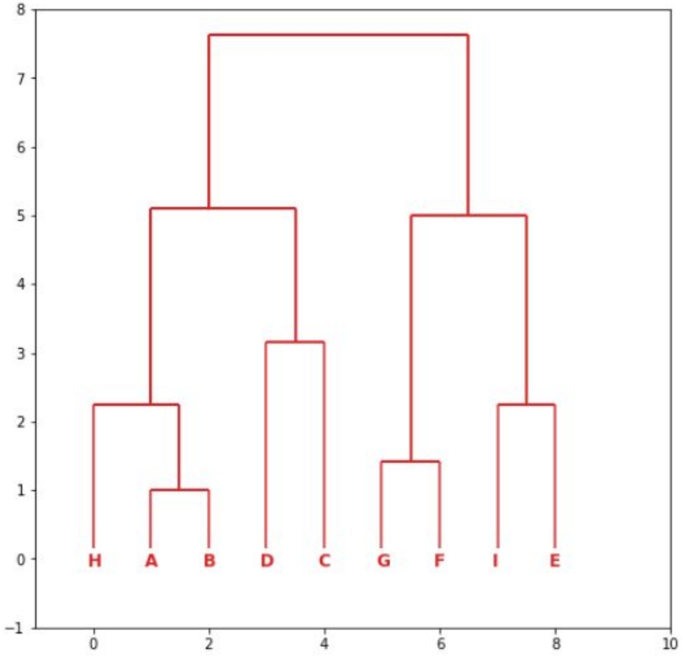


Complete Linkage

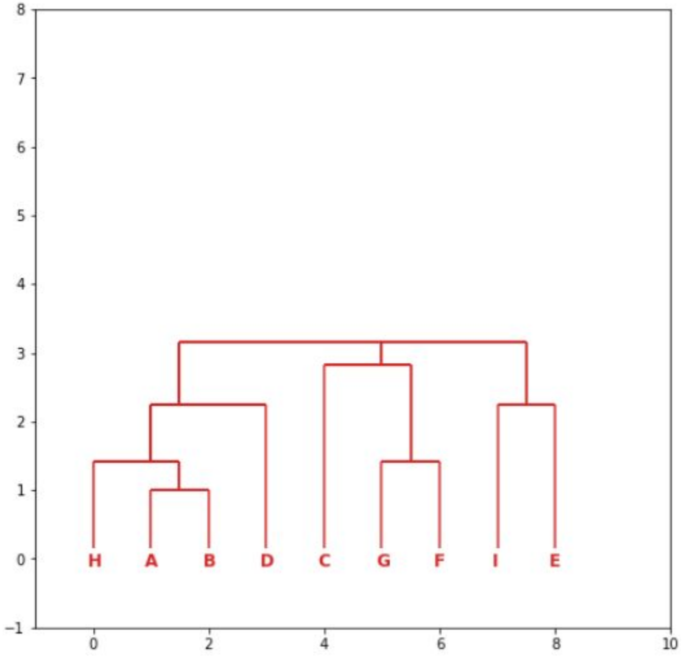


Results from different metrics

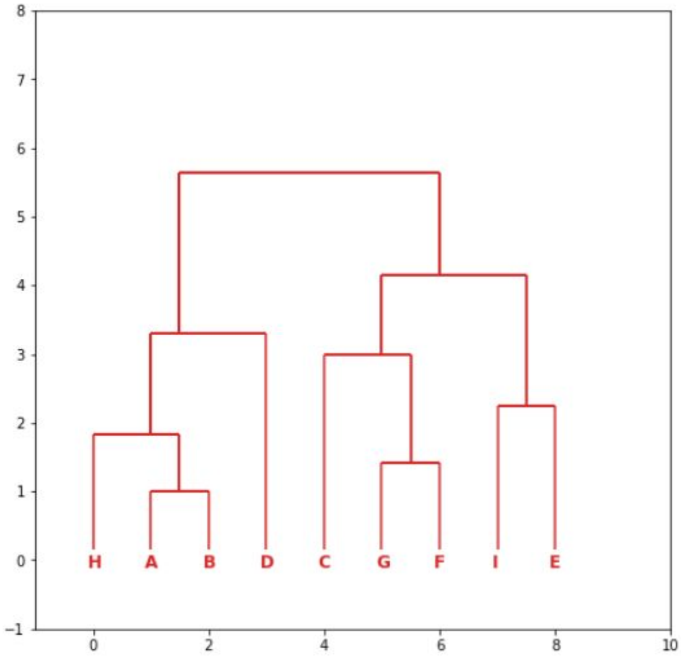
Complete Linkage



Single Linkage

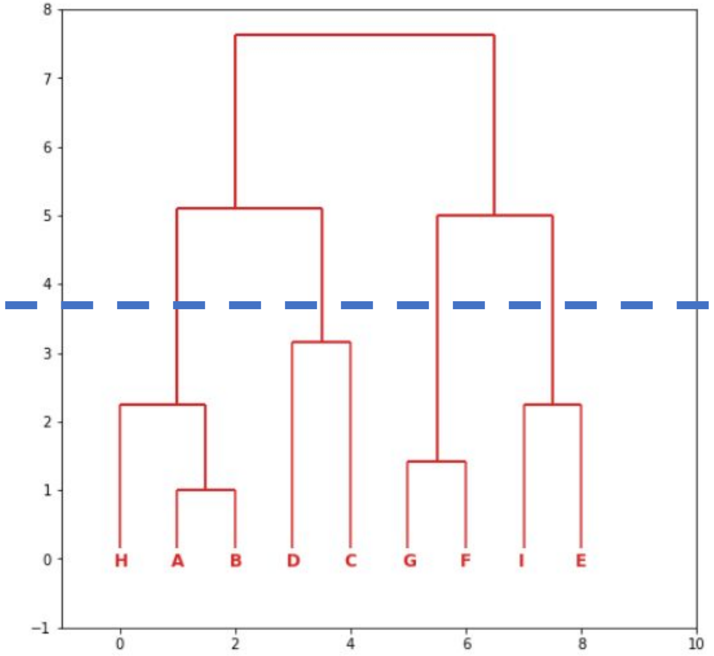


Average Distance

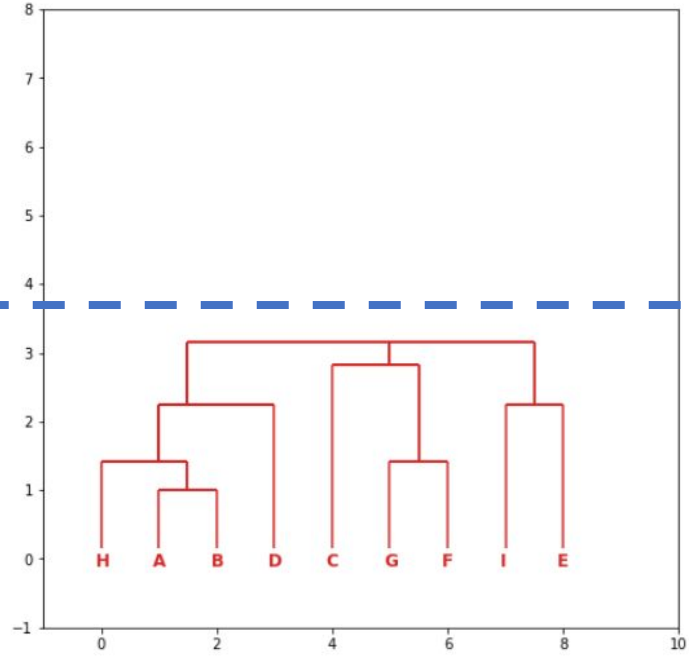


Finding clusters from the dendrogram

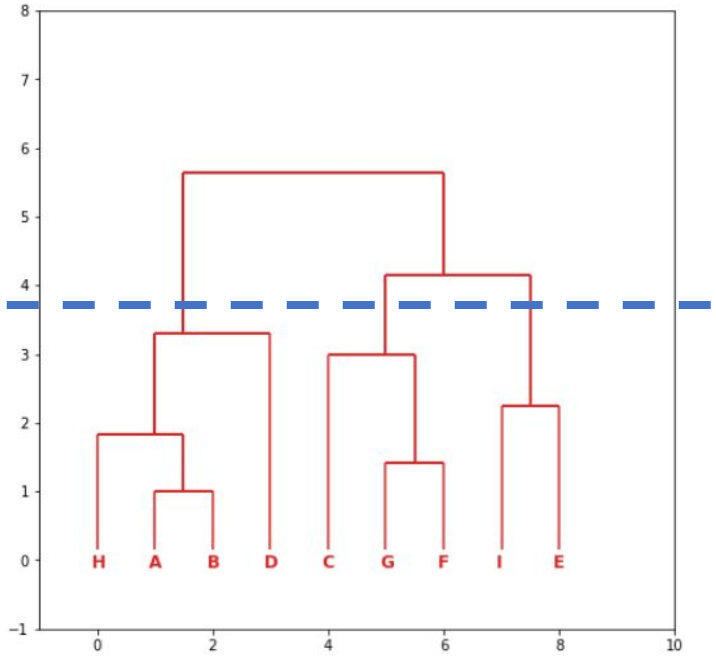
Complete Linkage



Single Linkage



Average Distance



Effect of (dis)similarity metric choice

Choice of similarity metric is very important

Example: identifying subgroups of shoppers

Data-> 100 millions of shoppers (rows) and 500 millions of items

What happens if we use Euclidean distance?

What if we use correlation?

Effect of feature scaling

Features may have very different range of values

Consider shopping frequency of certain items
(e.g.) AA battery vs. laptop

The solution: standardize