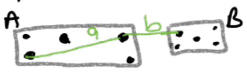
cluster analysis

- maximize inter-cluster distance and minimize intra-cluster distance
  ↳ between clusters ↳ inside the cluster elements

types of clustering

- partitional = non-overlapping clusters
- hierarchical = tree like : traditional / non-traditional (size first → merge 1's first then 2's)
- non-exclusive = a point may belong more than one cluster
  ↳ fuzzy clustering ⇒ x → 0.3 cluster A, 0.4 cluster B, 0.3 cluster C

types of clusters

- well-saparated = any point in a cluster is closer to every point in the cluster than to any point not in the cluster 

  a>b → not-well saparated
- prototype-based = " closer to the prototype / center of its cluster than others centers
  ↳ centroid, medoid: most representetive
- contiguous (nearest neighbor / transitive) = a point is closer to one or more other points in the cluster 

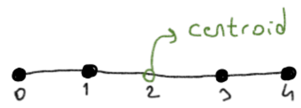
- density-based = dense regions are clusters, separated by low-density regions
  used when clusters are irregular, where there are outliers
- described by an objective function = find clusters that min/maximize an objective function

clustering algorithms

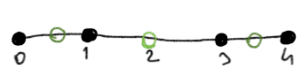
- k-means = complexity O(#points × #clusters × #iterations × # attributes) O(N)
  ↳ repeat until sum of squared error - euclidean func reaches minimum
  ↳ k-means++ = select initial centroids = next centroid it the point having max distance from the nearest centroid (most probably - logn converge guarantee)
  ↳ bisecting kmeans (ciKiye bölmek) = split some of points into two, choose one → like hierarchical clustering
- hierarchicol clustering = complexity $O(N^2)$ space → proximity / similarity matrix, $O(N^3)$ time → worst than k-means
  ↳ dendrogram, agglomerative (merge, n → 1 cluster) / divisive (spilit at each step, 1 ⇒ n clusters)
  ↳ single linkage = min dist → sensitive to noise
  ↳ complete linkage = among max distances choose min one
  ↳ average linkage = avg of distances between all datapoints among two clusters
  ↳ ward's method = sum of squared error, variance → like avg dist, but do not divide and squared
- DBSCAN = density based, merge core points until no left, assign borders to clusters at the end
  ↳ core point = if the point has at least N (minN) points within circle
  ↳ border point = neigboor of a core point $O(N^2)$
  ↳ noise point = neither core nor border point

measurement of the goodness of the clusters

- cluster cohesion = how closely related are objects in a cluster → SSE
- cluster separation = how distinct / well separated clusters are → SSB between clusters
- silhouette coefficient = considers both cohesion and separation
  ↳ $s = \frac{b-a}{\max(a,b)}$ $(-1 \leq s \leq 1)$ (1 → better), a = avg dist of a point to others in its cluster, b = avg dist in another cluster
- correlation = measurement of the corr between proximity matrix and ideal similarity matrix
  ↳ must be high in magnitude (can be negative too)
  ↳ 1: if same clusters
  ↳ 0: else



k = 1
$SSE = (0-2)^2 + (1-2)^2 + (3-2)^2 + (4-2)^2 = 10$
$SSB = 4(3-3)^2 = 0$
} + 10



k = 2
$SSE = (0-0.5)^2 + (1-0.5)^2 + (3-3.5)^2 + (4-3.5)^2 = 1$
$SSB = 2 \times (2-0.5)^2 + 2 \times (3.5-2)^2 = 9$
(↳ n in the cluster ↳ centroid ↳ cluster center)
} + 10

} SSE + SSB = constant