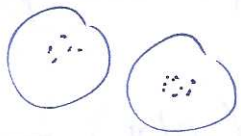
Unsupervised learning

Clustering methods



- **K-means clustering** K = # clusters.

data set $X = \{x_1, \dots x_n\} \subseteq \mathbb{R}^d$ let $C_1, \dots, C_K$ be a partion of $\{1, \dots, n\}$

need a measure of "within cluster variation" $W(C_k)$. small large.

then minimize: $\min_{\text{partition } C_1 \dots C_K} \sum_{k=1}^{K} W(C_k)$

Example $W(C_k)$ = squared Euclidean distance.

$$= \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{P} (x_{ij} - x_{i'j})^2.$$

= sum of all squared distances in cluster / size of cluster.

problem # partitions of $\{1, \dots n\}$ into K sets. approx K^n.

heuristic algorithm that often produces a useful answer:

1. randomly assign a cluster to each observation. (initial assignment)
2. iterate under the clusters stabilize:
   a) for each cluster compute the centroid i.e. the mean vector.
   b) assign each observation to the closest centroid cluster.

Fact this algorithm always reduces complexity

why? $\frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{P} (x_{ij} - x_{i'j})^2 = 2 \sum_{i \in C_k} (x_{ij} - \bar{x}_{kj})^2.$

where $\bar{x}_{kj} = \frac{1}{|C_k|} \sum_{i \in C_k} x_{ij}$ is the cluster mean.

key point do $d=1$. $x_1, \dots x_n$. then $\bar{x} = \frac{1}{n}(x_1 + \dots + x_n)$.

so RHS $= \frac{1}{n} \sum_{i,j} (x_i - x_j)^2 = \frac{1}{2} (x_1-x_1)^2 + (x_1-x_2)^2 + \dots + (x_1-x_n)^2$

$\vdots$

$(x_n-x_1)^2 + \dots + (x_n-x_n)^2$.

LHS $= \sum_i (x_i - \bar{x})^2 = \sum_i \left(x_i - \frac{x_1 + x_2 + \dots x_n}{n} \right)^2$.

$= \left(x_1 - \frac{x_1+\dots+x_n}{n}\right)^2 + \left(x_2 - \frac{x_1+\dots+x_n}{n}\right)^2 + \dots + \left(x_n - \frac{x_1+\dots+x_n}{n}\right)^2$.

$= \frac{1}{n^2} \left[\sum_j (x_1 - x_j)^2 + \sum_j (x_2 - x_j)^2 + \dots + \sum_j (x_n - x_j)^2 \right]$.

this means algorithm finds local minimum.

— so run the algorithm a bunch of times, w different initial conditions.

Q: what is k? A: no idea.

Hierarchical clustering ⇝ make a tree (dendrogram)

Algorithm $X = \{x_1, \dots, x_n\} \subseteq \mathbb{R}^d$.

choose measure/distance, e.g. Euclidean distance.

1) start with n data points and $\binom{n}{2}$ ~~euclr~~ distances. (compute)

2) merge the most similar clusters. ← need measure of distance for groups of points

common choices:

- complete take $\max_{a_i \in A,\ b_j \in B} d(a_i, b_j)$.

- single take $\min d(a_i, b_j)$ (prefer).

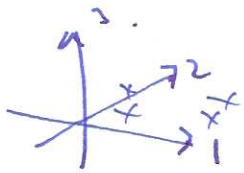
- average $\frac{1}{\binom{n}{2}} \sum_{ij} d(a_i, b_j)$.

- centroid compute closest centroid. (can give inversions - clusters find at heights below. each other).

— could use same measure of correlation.

example: 1 2 3 similar size so close.

correlated but big distances.



choices: normalize?

distance (dissimilarity measure)

linkage

where should we cut the tree? (here)

what is K (K-means).