Lec 3

Dissimilarity

① - Minkowski

② Euklidian = 1 (triangle dist)

③ Manhattan = 2 (block dist)

params : d (dimension), p = (weight ?)

④ - Jaccard Similarity → Manhattan for set differences.

1 when $x_i \neq y_i$ and 0 when $x_i = y_i$

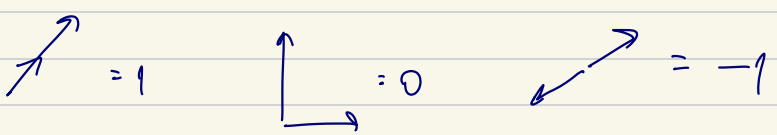
* differ 2M then Manhattan of 1

$$JDist(x,y) = 1 - \frac{|x \cap y|}{|x \cup y|}$$ (same / different)

⑤ Cosine similarity

- take two objects, return large if similar.

$s(x,y) = \cos\theta$



↳ dissimilarity $= \frac{1}{s(x,y)}$ or $1 - s(x,y)$

Lec #4

Clustering - kmeans

Clustering ⇒ groups of assignment

Cost function $\sum_{i}^{k} \sum_{x \in C_i} d(x_i, \mu_i)^2$

* minimize cost

parameters: d: euclidian dist
k: center of clusters

* k=1 k=n is easy bc.
  - k=1 ↓ all is one cluster
  - k=n ↓ each k is own cluster

-> 2 dimensions would be rly hard

Lloyd's algo

1. pick center at random, k
2. assign data point to closet k
3. compute new cluster center.
4. repeat until converge

we could pick outlier as centers :c

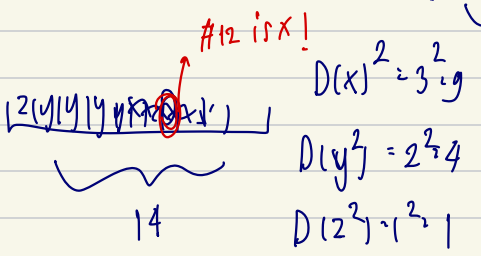
but not always optimal!

Lec #5 / equally fine w/ K mean

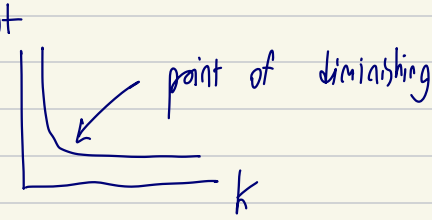
K-mean++

1. pick random center
2. let $D(x)$ be dist of x and closet centers picked. Choose next center w/ prob proportional to $D(x^2)$

- Black box and uniformly chose btw 0 and N.



♥ Choose right K is **elbow method**



Silhouette score

- avg within cluster dist : a
- avg into cluster dist. : b

} if b-a = 0 they identical !?

b-a large → seperate, high quality

b-a small not so much

♡ Silhouette score

a: within avg cluster
b: inter cluster

$$\frac{(b-a)}{\max(a,b)}$$

-1 to 1

-1 ↓ bad

1 ↓ good

K-means variations

1. k medians (L_1 norm / manhatan dist)
2. k-medoid (any dist + center in dataset)
3. weighted k-means (each point diff weight)

Hierarchical clustering
- ① Agglomerative (down → top)
- ② Divisive. (top → down)

- every step record cluster to merge in Dendrogram

parameter : dist between points + dist btw clusters

① Single-Link Distance (min of pairwise dist)

pros:
cons: sensitive to noise, elongated cluster / chain-like

② Complete-Link Distance (max of pairwise)

pros: less susceptible to noise, more balanced, equal diameters
cons: split up large cluster, same, circle sphere

③ Average-Link Distance. (avg. of pairwise dist) ← combination

pros: less susceptible to noise and outliers
cons: bias toward globular clusters

④ centroid Distance (dist btw two centroid of cluster)

⑤ Ward's Distance (diff btw variance of points in cluster)

Lecture # 6

Density-based Clustering

↳ cluster pts that are densely packed together

params : fix radius ε
mins # of point min_pts

1. core : ε-neighborhood w/ at least min_point
2. boarder : in ε but not core.
3. Noise : neither core nor boarder

DBScan algorithm (DFS)

1. find ε-neighboor hood for each point
   - if atleast min_point → core.
2. for each core, assign to same cluster
   all core point in same ε
3. if in core ε-neighbor radius then mark boarder.
4. what left is define all else as noise.
5. assign boarder to nearby cluster

pro : identify diff sizes & shape, resistant to noise
cons : fail to identify clusters of differing density
tend to create clusts of same density
notion of density is problem specific.

Lecture #7 / Normal distribution

Soft-clustering $P(s_j | x_i)$

Maximum likelihood

↳ output is weight / prob that it come from that species

$$P(s_j | x_i) = \frac{P(x_i | s_j) P(s_j)}{P(x_i)}$$

Lecture #8

Clustering_Aggregation

↳ compare & combine
↳ agree or disagree

- Disagreement Dist

$$D(x,y)\begin{cases}1, & \text{if disagree}\\ 0, & \text{if agree}\end{cases}$$

Aggregate clustering :

pos: identify best # of clust
detect outlier
robustest
privacy

cons : NP-hard