CS 181 Clustering March 2021

New task $\underline{x} \rightarrow \text{summary}(\underline{x})$

Motivations understanding communication
pre-training for supervised
organizing

Data $\{\underline{x}_1, \ldots, \underline{x}_N\}$ often $\underline{x}_n \in \mathbb{R}^D$
could be $\underline{x}_n \in \{0,1\}^D$

Number of clusters K (may not be given)

Output: assignment of each example to a cluster

$\underline{z}_n$ 1-hot "assignment" $\begin{cases} z_{nk} = 0 & \text{if } \underline{x}_n \text{ is not in cluster } k \\ z_{nk} = 1 & \text{if } \underline{x}_n \text{ is in cluster } k \end{cases}$

What is a good clustering?

Idea: examples to be more similar to examples in same cluster than to examples in other clusters

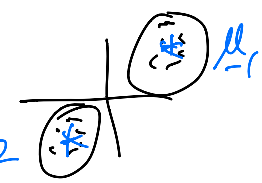
Need measure of similarity:

e.g., $d(\underline{x}, \underline{x}') = \| \underline{x} - \underline{x}' \|_2$ (L2)

edit distance (strings)

Hamming distance (bit vectors)

$\vdots$



$\boxed{\text{Alg 1}}$ K-means clustering

- Define "prototype" $\underline{\mu}_k \in \mathbb{R}^D$ per cluster
- Goal assign $\underline{z}_n$ + find $\{\underline{\mu}_1, \dots, \underline{\mu}_K\}$ s.t.

$$\min_{\{\underline{\mu}\}\{\underline{z}\}} \sum_n \sum_k z_{nk} \| \underline{x}_n - \underline{\mu}_k \|_2^2$$

Note: Non-convex

NP-hard

$$z_{nk} = \begin{cases} 1 & \text{if assigned } k \\ 0 & \text{o.w.} \end{cases}$$

Addendum

Objective criterion should be $\| \underline{x}_n - \underline{\mu}_k \|_2^2$ not $\| \underline{x}_n - \underline{\mu}_k \|_2$ as written in lecture

Lloyd's algorithm

1) Randomly initialize prototypes $\{\underline{\mu}_k\}$

2) Repeat:

Step 1 Assign each example to its closest prototype

$$\left.\arg\min_k \|\underline{x}_n - \underline{\mu}_k\|_2\right\} \text{for each } n$$

Step 2 For each k, set $\underline{\mu}_k$ to the centroid (mean) of assigned examples

$$\underline{\mu}_k := \frac{1}{N_k}\sum_n z_{nk}\,\underline{x}_n$$

where $N_k = \sum_n z_{nk}$

\# examples assigned to k

Typical restart this multiple times & take the "best" solution

Understanding Lloyd's algorithm

Recall objective

$$\min_{\{\underline{\mu}\}\{\underline{z}\}} \sum_n \sum_k z_{nk} \|\underline{x}_n - \underline{\mu}_k\|_2^2 \qquad (*)$$

"Coordinate descent", alternating $\{\underline{\mu}\}$ + $\{\underline{z}\}$ updates

(Repeat)

Step 1: Fixing $\{\underline{\mu}\}$, minimize the loss $(*)$ by assigning each $\underline{x}_n$ to closest prototype

Step 2: Fixing $\{\underline{z}\}$, minimize loss $(*)$ by choosing prototypes $\{\underline{\mu}\}$

for k: $$L(\underline{\mu}_k) = \sum_n z_{nk} (\underline{x}_n - \underline{\mu}_k)^T (\underline{x}_n - \underline{\mu}_k)$$

$$\frac{\partial L(\underline{\mu}_k)}{\partial \underline{\mu}_k} = -2 \sum_n z_{nk} (\underline{x}_n - \underline{\mu}_k) = 0$$

$$\Leftrightarrow \underline{\mu}_k = \frac{1}{N_k} \sum_n z_{nk} \underline{x}_n$$

N_k # examples assigned to k

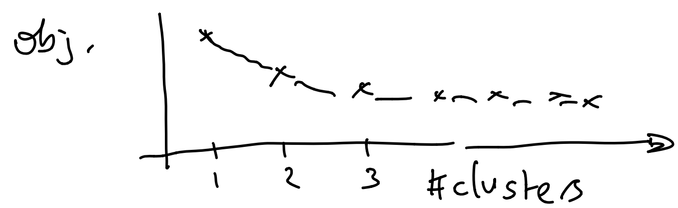
Again minimizing!

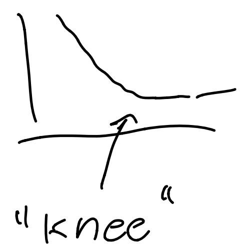
Considerations

1) How many cluster?

smaller, better interpretation

larger, better extraction of concepts





2) Parametric method

$D \times K$ parameters (prototypes)

3) Inflexible (linear decision boundary)

4) Fast! Assignment z_{nk} step parallelizes by $\underline{x}_n$; prototype step parallelizes by $\underline{\mu}_k$

Variations

1) k-means++ to initialize

2) K-medoids

replace $\underline{\mu}_k$ step with:

$\underline{\mu}_k :=$ example assigned to k that minimizes total distance to the other examples in the cluster

$$\arg\min_{\substack{\underline{x}_n \text{ s.t.} \\ z_{nk}=1}} \sum_{n'} z_{n'k} \| \underline{x}_n - \underline{x}_{n'} \|_2^2$$

3) L1 norm in place of L2 norm

$\underline{\mu}_k :=$ median of the points assigned to cluster k

Alternate Hierarchical Agglomerative clustering (HAC)

Data $\{\underline{x}_n\}$ Distance $\underbrace{d(G, G')}_{\text{distance between groups}}$

HAC:

1) Every example starts in its cluster

2) While the # clusters > 1,
Merge the two "closest" clusters

Notes

Forms a hierarchy of clusters

No need to specify K (# cluster)

Deterministic

Need two concepts

1) $d(\underline{x}, \underline{x}')$ distance between points

2) "linkage" function, min, max, average, centroid

$\mapsto$ Get $d(G, G')$

Comments

1) Average, centroid compromises between min + max

2) Non-parametric (distance-based)

   Arbitrary cluster shapes

3) $\left.\begin{array}{l}\text{Scales as } O(n^2) \\ \text{Pairwise calculation}\end{array}\right\}$ k-means $O(n)$

4) Can sometimes lead to overfitting

K-means

compares $\underline{x}_n$ to prototype $\underline{\mu}_k$

+ prototypes are averages

HAC

compares pairs of examples

(no averaging effect)

Concept check

Data $\quad \underbrace{0 \cdots 0}_{\ell} \quad \underbrace{\text{Random } 0/1}_{D-\ell} \quad$ " $\underline{x}_0$ "

$\quad \underbrace{1 \cdots 1}_{\ell} \quad \underbrace{\text{Random } 0/1}_{D-\ell} \quad$ " $\underline{x}_1$ "

$\boxed{\text{Problem}}$:

Given $\underline{x}$, $\underline{x}'$ in "0" cluster + $\underline{z}$ in "1" cluster, the probability $\underline{x}$ is closer to $\underline{x}'$ than to $\underline{z}$ goes to $1/2$ as $D \to \infty$ (fixing ℓ)

- Noise comes to dominate } + HAC fails
- K-means works OK here

$$\underline{\mu}_0 = (0, \ldots, 0, 1/2, 1/2, \ldots, 1/2)$$
$$\underline{\mu}_1 = (1, \ldots, 1, 1/2, 1/2, \ldots, 1/2)$$

With these prototypes we can correctly cluster data + K-means converges why the problem with noise in HAC?

The "curse of dimensionality"

Suppose 1000 examples are uniform randomly distributed in a D-dimensional unit hypercube

Consider the squared-distance between random examples $\underline{x}$, $\underline{z}$:

$$\|\underline{x} - \underline{z}\|_2^2 = \sum_{j=1}^{D} (x_j - z_j)^2$$ } sum of D, i.i.d. random variables

[1] By central limit theorem, concentrates around $D \times E\left[(x_j - z_j)^2\right]$ where $x_j, z_j \sim U(0,1)$. Distance is sqrt of this.

(compare to min distance 0, max distance $\sqrt{D}$)

[2] Distr. of inter-example distances

$\longrightarrow$ increasing concentration