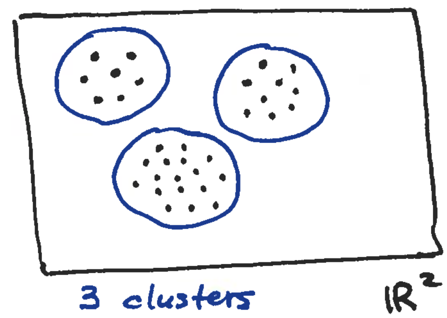
Lecture 5 The K-means Algorithm for Clustering Vectors

Last time we introduced the notion of an innerproduct, discussed how to measure the length of a vector, and how to compute angles and distances between vectors. In this lecture, we will discuss the clustering problem, which often appears in applications whenever we have many vectors in a normed vector space (so that we can compute distances between them). The clustering problem is then to divide vectors into groups (clusters) of vectors that are close to each other. In particular, we will discuss a famous clustering method, called the K-means algorithm, which is widely used in applications.



Remark 1 K-means will also illustrate one old and very useful idea often used in numerical optimization

Remark 2 Clustering can be discussed in general normed vector spaces, but for the sake of simplicity, we will focus on $(\mathbb{R}^d, \|\cdot\|_2)$.

Clustering Problem

Suppose we have n vectors $v_1, \dots, v_n \in \mathbb{R}^d$.

Goal: partition $v_1 \dots v_n$ into K clusters $(K \ll n)$, with the vectors in each cluster close to each other.

Applications:

$n \sim 10^2 - 10^6$

$K \sim 2, 3 - 10^2$

① Topic discovery
We have n documents (web, text, etc) and a list of d "key-words". $(v_i)_j = \#$ word j in document i
Then clusters are groups of similar documents.

② Patient clustering
Coordinates of v_i are features (height, weight, age, etc) associated with patients admitted to a hospital.
Then clusters are groups of similar patients.

③ Market segmentation
We have n customers and d products.
$(v_i)_j = \#\$$ customer i spent on product j

Then clusters are market segments.

Why is clustering difficult?

1. $d \gg 1$ (if $d = 2 \Rightarrow$ easy, use your eyes :-))
2. Not clear what the best value of K is
3. Real data are not "cleanly" clustered.

Problem Formalization

Assume K is given (how to choose K we will discuss at the end).
Let's describe any clustering by a vector $c \in \mathbb{R}^n$, where

$$c_i = \text{cluster number that } v_i \text{ is assigned to } \quad (\in \{1, \dots, K\})$$

Q: How many possible clusterings?

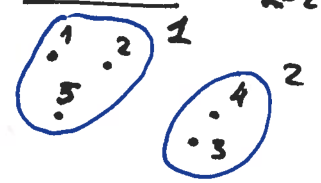
Anyway, the number is huge.

- K^n (if clusters are "labeled")

are viewed as diff.

- $\underbrace{\left\{ {n \atop K} \right\} = \frac{1}{K!} \sum_{j=0}^{K} (-1)^{K-j} \binom{K}{j} j^n}_{\text{Stirling number of the second kind.}}$ (if clusters are "unlabeled")

Example $n=5$, $K=2$



$c = (1, 1, 2, 2, 1)$

To find a good clustering, we should be able to comare different clusterings. To compare clusterings, we need to introduce a natural measure of the clustering quality.

Idea: (K-means) Let's associate with each cluster k a cluster representative $r_k \in \mathbb{R}^d$.
We want all vectors in cluster k to be close to the representative r_k:

can be any vector in $\mathbb{R}^d$ (not necessarily one of $v_1 \dots v_n$)

we want $\|v_i - r_{c_i}\|$ to be small for all $i = 1, \dots, n$.

Given clustering c and representatives $r_1, \dots, r_K$, we can measure the quality of $(c; r_1 \dots r_K)$ by the following objective function:

$$\boxed{\rho(c; r_1 \dots r_K) = \frac{1}{n} \sum_{i=1}^{n} \|v_i - r_{c_i}\|^2}$$

The smaller ρ is, the better the clustering.

Thus, we formalized the clustering problem to the following optimization problem.
(reduced)

$$(*)\quad \begin{cases} \rho(c; r_1 \dots r_K) \to \min \\ r_1, \dots, r_K \in \mathbb{R}^d \\ c \in \mathbb{R}^n, \; c_i \in \{1, \dots, K\} \end{cases}$$

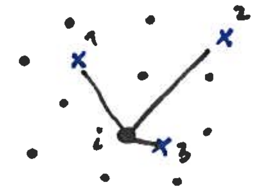
The solution of this problem c^* is called optimal clustering.

In practice, it is impossible to find the optimal clustering: the optimization problem is to difficult for large n.

The K-means algorithm allows to find a suboptimal clustering, i.e. clustering which is not optimal, but close to optimal, $\rho_{K\text{-mean}} > \rho_{opt}$, $\rho_{K\text{-means}} \approx \rho_{opt}$

K-means is based on the following idea which is very often used in numerical optimization.

① We can solve $(*)$ exactly if $r_1, \dots, r_K$ are fixed.



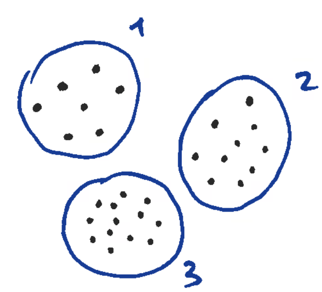
$$\rho(c; r_1 \dots r_K) = \frac{1}{n} \sum_{i=1}^{n} \| v_i - r_{c_i} \|^2 \to \min$$

We need to find $c = (c_1, \dots, c_n)$, which minimizes the objective function. Key observation: c_i affects only i^{th} term.

$\Rightarrow$ to minimize ρ, we need to minimize each term independently:

$$f(c_i) = \| v_i - r_{c_i} \|^2 \to \min \quad \Rightarrow \quad \boxed{c_i = \arg\min_k \| v_i - r_k \|^2} \quad k = 1 \dots K$$

② We can solve $(*)$ exactly if c is fixed.



$$\rho(c; r_1 \dots r_K) = \frac{1}{n} \sum_{i=1}^{n} \| v_i - r_{c_i} \|^2$$

$$= \rho_1 + \rho_2 + \dots + \rho_K$$

vectors associated with cluster k

$$\rho_k = \frac{1}{n} \sum_{i : c_i = k} \| v_i - r_k \|^2$$

contribution from cluster k.

that is why squared distance is convenient

Representative r_k affects only ρ_k $\Rightarrow$ to minimize ρ, we need to minimize each ρ_k by choosing r_k:

~~[illegible]~~ $\rho_k(r_k) \to \min$

The solution is simple:

r_k is the centroid (average) of vectors in cluster k

$$\boxed{r_k = \frac{1}{n_K} \sum_{i : c_i = k} v_i}, \quad n_K = \#\text{ vectors in cluster } k = |\{i : c_i = k\}|$$

The K-means algorithm

So, we can minimize $\rho(c, \{r\})$ over c for fixed $\{r\}$ and over $\{r\}$ for fixed c, but we can't minimize it over c and $\{r\}$ simultaneously.

Standard solution: iterate between the two minimizations
(popular strategy)

$$\{r\} \rightleftarrows c$$

Algorithm:

1. Pick initial representatives $r_1, \dots, r_K$ randomly from the original vectors $v_1 \dots v_n$

There are more sophisticated methods, but this is beyond the scope of acm 104.

Repeat untill convergence

2. Partition $v_1 \dots v_n$ into K clusters
Assigne v_i to the cluster associated with the nearest r_k

3. Update representatives $r_1 \dots r_K$

r_k = mean of v_i in claster k.

Remark 1 Let $c^{(1)}, c^{(2)}, \dots$ be clusterings obtained in these iterations.
If $c^{(m)} = c^{(m-1)} \Rightarrow \{r\}^{(m)} = \{r\}^{(m-1)}$ and both c and $\{r\}$ will not change in future ($>m$) iterations.

"Convergence" $= \{c^{(m)} = c^{(m-1)}\}$

Remark 2 At each iteration, the value of ρ decreases $\Rightarrow$ the k-means alg converges in a finite number of steps.
Depending on the initial choice of $r_1 \dots r_K$, it may converge to different final clusterings with different values of ρ.
$\Rightarrow$ it is common to run the algorithm several times with different $r_1 \dots r_K$ and choose clustering with the smallest value of ρ.

Find min

$$\{r\}^{(0)} \rightsquigarrow \{r\}^{(m)}, c^{(m)} \qquad \rho_{\text{final}}$$

$$\{r'\}^{(0)} \rightsquigarrow \{r'\}^{(m')}, c'^{(m')} \qquad \rho'_{\text{final}}$$

$$\vdots$$

Remark 3 How to choose K ?

It is difficult to choose the right value of K in advance.

Common Strategy:

Run K-means with different values of K,
for each K, compute the corresponding minimized value of the objective function $\rho_{min}(K)$. This is a decreasing function ($\rho_{min}(n) = 0$)

Look for the following pattern