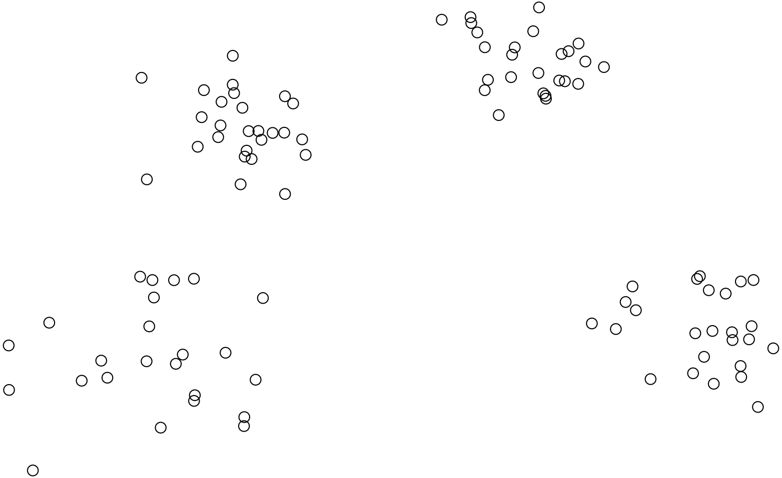
Mini lecture

k-means clustering

Arthur Van Camp

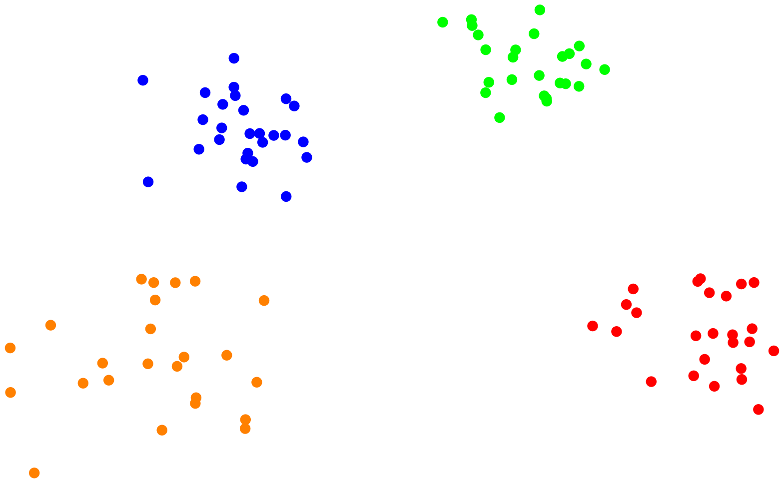
Monday 15 May 2023

Introduction



Slides: arthurvancamp.github.io/mini-lecture/mini-lecture-slides.pdf

Introduction



Slides: arthurvancamp.github.io/mini-lecture/mini-lecture-slides.pdf

Clustering is widely used

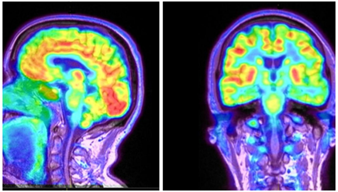
Social networks

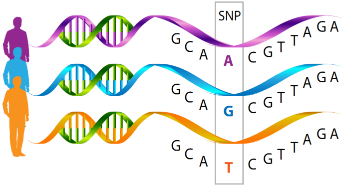
(Medical) imaging

Market analysis

Chemistry

Gene sequencing



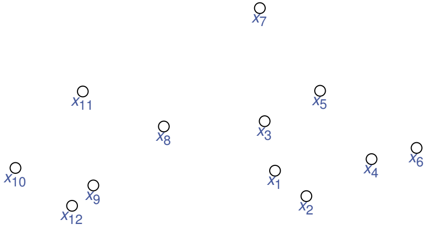


Clustering

Given: $x_1, \ldots, x_n$ vectors,
and k: number of clusters, with $n \gg k$.

Required: Partition $\{x_1, \ldots, x_n\}$ into $\{C_1, \ldots, C_k\}$ such that each group (cluster) C_ℓ contains vectors that are close to each other.

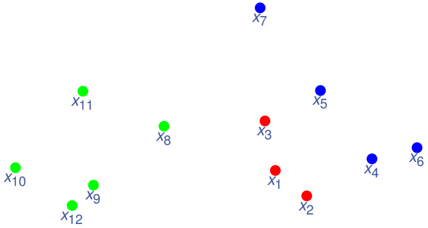
$\{C_1, \ldots, C_k\}$ is a **partition** of $\{x_1, \ldots, x_n\}$ if $C_1 \cup C_2 \cup \cdots \cup C_k = \{x_1, \ldots, x_n\}$ and $C_i \cap C_j = \emptyset$ for all $i \neq j$.



Clustering

Given: $x_1, \ldots, x_n$ vectors,
and k: number of clusters, with $n \gg k$.

Required: Partition $\{x_1, \ldots, x_n\}$ into $\{C_1, \ldots, C_k\}$ such that each group (cluster) C_ℓ contains vectors that are close to each other.



$\{C_1, \ldots, C_k\}$ is a **partition** of $\{x_1, \ldots, x_n\}$ if
$C_1 \cup C_2 \cup \cdots \cup C_k = \{x_1, \ldots, x_n\}$ and
$C_i \cap C_j = \emptyset$ for all $i \neq j$.

We will study k-means clustering.

k-means clustering

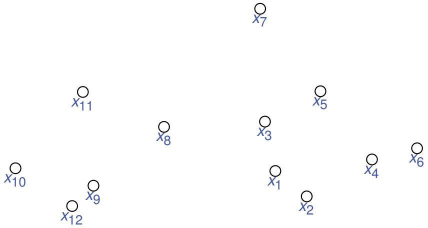
Idea: Find a partition $\{C_1, \ldots, C_k\}$ such that

$$\sum_{\ell=1}^{k} \frac{1}{|C_\ell|} \sum_{x,y \in C_\ell} \|x - y\|^2$$

is as small as possible.

$\|\bullet\|$ is the **Euclidean norm**: $\|x\| = \sqrt{\sum_{i=1}^{d} x_i^2}$.

Exercise 1: Why is there a factor $\frac{1}{|C_\ell|}$?



k-means clustering

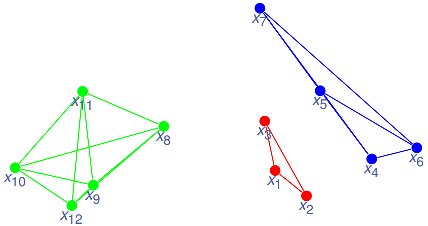
Idea: Find a partition $\{C_1, \ldots, C_k\}$ such that

$$\sum_{\ell=1}^{k} \frac{1}{|C_\ell|} \sum_{x,y \in C_\ell} \|x - y\|^2$$

is as small as possible.

$\|\bullet\|$ is the **Euclidean norm**: $\|x\| = \sqrt{\sum_{i=1}^{d} x_i^2}$.

Exercise 1: Why is there a factor $\frac{1}{|C_\ell|}$?



A useful alternative expression

Find a partition $\{C_1, \ldots, C_k\}$ that minimises $\sum_{\ell=1}^{k} \frac{1}{|C_\ell|} \sum_{x,y \in C_\ell} \|x - y\|^2$.

A useful alternative expression

Find a partition $\{C_1, \ldots, C_k\}$ that minimises $\sum_{\ell=1}^{k} \frac{1}{|C_\ell|} \sum_{x,y \in C_\ell} \|x - y\|^2$.

Define $\mu_\ell = \sum_{x \in C_\ell} \frac{x}{|C_\ell|}$, the centroid of cluster C_ℓ. Then the same $\{C_1, \ldots, C_k\}$ minimises

$$\sum_{\ell=1}^{k} \sum_{x \in C_\ell} \|x - \mu_\ell\|^2. \qquad \text{(Exercise 2)}$$

However, finding a partition that minimises $\sum_{\ell=1}^{k} \sum_{x \in C_\ell} \|x - \mu_\ell\|^2$ is NP-hard.

Exercise 2: solution

Compare

$$\sum_{x,y\in C_\ell}\|x-y\|^2 = \sum_{x,y\in C_\ell}\Big(\|x\|^2+\|y\|^2-2\langle x,y\rangle\Big)$$

$$= 2|C_\ell|\sum_{x\in C_\ell}\|x\|^2 - 2\sum_{x,y\in C_\ell}\langle x,y\rangle = 2|C_\ell|\sum_{x\in C_\ell}\|x\|^2 - 2\sum_{x\in C_\ell}\langle x,\mu_\ell\rangle|C_\ell| = 2|C_\ell|\Big(\sum_{x\in C_\ell}\|x\|^2\Big) - 2|C_\ell|^2\|\mu_\ell\|^2$$

with

$$\sum_{x\in C_\ell}\|x-\mu_\ell\|^2 = \sum_{x\in C_\ell}\Big(\|x\|^2+\|\mu_\ell\|^2-2\langle x,\mu_\ell\rangle\Big)$$

$$= \sum_{x\in C_\ell}\|x\|^2 + |C_\ell|\|\mu_\ell\|^2 - 2\sum_{x\in C_\ell}\langle x,\mu_\ell\rangle = 2\sum_{x\in C_\ell}\|x\|^2 + |C_\ell|\|\mu_\ell\|^2 - 2|C_\ell|\|\mu_\ell\|^2 = \Big(\sum_{x\in C_\ell}\|x\|^2\Big) - |C_\ell|\|\mu_\ell\|^2$$

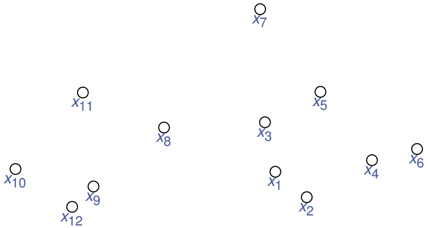
to infer that $\sum_{x\in C_\ell}\|x-\mu_\ell\|^2 = \frac{1}{2|C_\ell|}\sum_{x,y\in C_\ell}\|x-y\|^2$. Therefore minimising $\sum_{\ell=1}^k\sum_{x\in C_\ell}\|x-\mu_\ell\|^2$ is tantamount to minimising $\sum_{\ell=1}^k\frac{1}{|C_\ell|}\sum_{x,y\in C_\ell}\|x-y\|^2$, as the former is a factor 2 larger than the latter.

A heuristic: Lloyd's algorithm

Reformulation: Find a partition $\{C_1, \ldots, C_k\}$ such that

$$\sum_{\ell=1}^{k} \sum_{x \in C_\ell} \|x - \mu_\ell\|^2$$

is as small as possible.



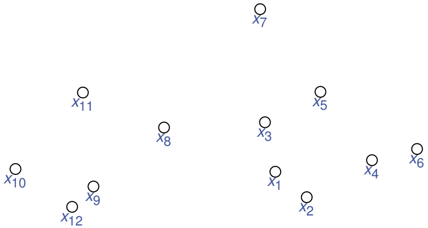
A heuristic: Lloyd's algorithm

Reformulation: Find a partition $\{C_1, \ldots, C_k\}$ such that

$$\sum_{\ell=1}^{k} \sum_{x \in C_\ell} \|x - \mu_\ell\|^2$$

is as small as possible.

Step 1: Choose k points (centroids) $\mu_1, \ldots, \mu_k$ from $\{x_1, \ldots, x_n\}$.



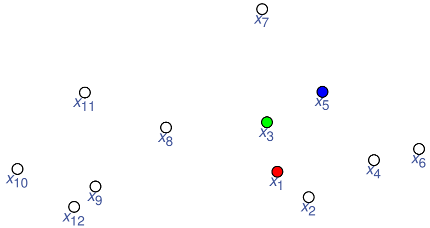
A heuristic: Lloyd's algorithm

Reformulation: Find a partition $\{C_1, \ldots, C_k\}$ such that

$$\sum_{\ell=1}^{k} \sum_{x \in C_\ell} \|x - \mu_\ell\|^2$$

is as small as possible.

Step 1: Choose k points (centroids) $\mu_1, \ldots,$ μ_k from $\{x_1, \ldots, x_n\}$.



A heuristic: Lloyd's algorithm

Reformulation: Find a partition $\{C_1, \ldots, C_k\}$ such that

$$\sum_{\ell=1}^{k} \sum_{x \in C_\ell} \|x - \mu_\ell\|^2$$

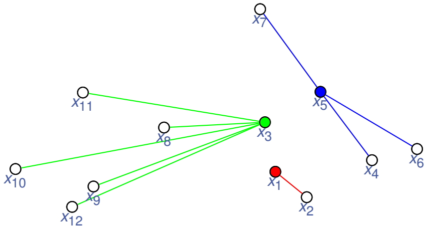
is as small as possible.

Step 1: Choose k points (centroids) $\mu_1, \ldots, \mu_k$ from $\{x_1, \ldots, x_n\}$.

Step 2: While centroids are not stable:

a. Define clusters: add x_i to cluster C_ℓ if μ_ℓ is the centroid closest to x_i.

b. Update μ_ℓ by computing the new centroid of C_ℓ.



A heuristic: Lloyd's algorithm

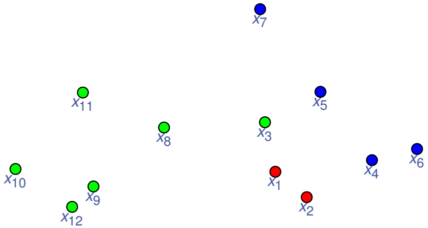
Reformulation: Find a partition $\{C_1, \ldots, C_k\}$ such that

$$\sum_{\ell=1}^{k} \sum_{x \in C_\ell} \|x - \mu_\ell\|^2$$

is as small as possible.

Step 1: Choose k points (centroids) $\mu_1, \ldots, \mu_k$ from $\{x_1, \ldots, x_n\}$.

Step 2: While centroids are not stable:
a. Define clusters: add x_i to cluster C_ℓ if μ_ℓ is the centroid closest to x_i.
b. Update μ_ℓ by computing the new centroid of C_ℓ.



A heuristic: Lloyd's algorithm

Reformulation: Find a partition $\{C_1, \dots, C_k\}$ such that

$$\sum_{\ell=1}^{k} \sum_{x \in C_\ell} \|x - \mu_\ell\|^2$$

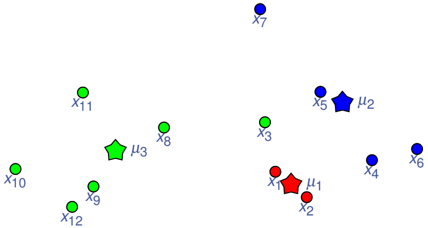
is as small as possible.

Step 1: Choose k points (centroids) $\mu_1, \dots, \mu_k$ from $\{x_1, \dots, x_n\}$.

Step 2: While centroids are not stable:

a. Define clusters: add x_i to cluster C_ℓ if μ_ℓ is the centroid closest to x_i.

b. Update μ_ℓ by computing the new centroid of C_ℓ.



A heuristic: Lloyd's algorithm

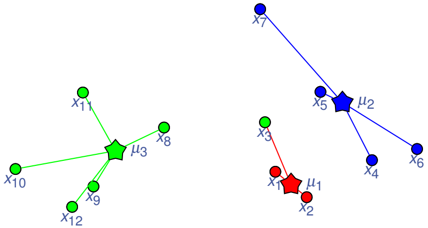
Reformulation: Find a partition $\{C_1, \ldots, C_k\}$ such that

$$\sum_{\ell=1}^{k} \sum_{x \in C_\ell} \|x - \mu_\ell\|^2$$

is as small as possible.

Step 1: Choose k points (centroids) $\mu_1, \ldots, \mu_k$ from $\{x_1, \ldots, x_n\}$.

Step 2: While centroids are not stable:
a. Define clusters: add x_i to cluster C_ℓ if μ_ℓ is the centroid closest to x_i.
b. Update μ_ℓ by computing the new centroid of C_ℓ.



A heuristic: Lloyd's algorithm

Reformulation: Find a partition $\{C_1, \ldots, C_k\}$ such that

$$\sum_{\ell=1}^{k} \sum_{x \in C_\ell} \|x - \mu_\ell\|^2$$

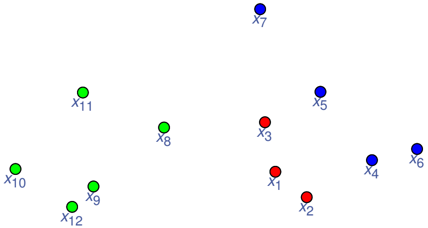
is as small as possible.

Step 1: Choose k points (centroids) $\mu_1, \ldots, \mu_k$ from $\{x_1, \ldots, x_n\}$.

Step 2: While centroids are not stable:

a. Define clusters: add x_i to cluster C_ℓ if μ_ℓ is the centroid closest to x_i.

b. Update μ_ℓ by computing the new centroid of C_ℓ.

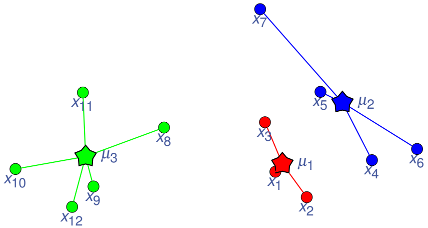


A heuristic: Lloyd's algorithm

Reformulation: Find a partition $\{C_1, \ldots, C_k\}$ such that

$$\sum_{\ell=1}^{k} \sum_{x \in C_\ell} \|x - \mu_\ell\|^2$$

is as small as possible.



Step 1: Choose k points (centroids) $\mu_1, \ldots, \mu_k$ from $\{x_1, \ldots, x_n\}$.

Step 2: While centroids are not stable:
a. Define clusters: add x_i to cluster C_ℓ if μ_ℓ is the centroid closest to x_i.
b. Update μ_ℓ by computing the new centroid of C_ℓ.

Exercise 3: Implement Lloyd's algorithm

Implement Lloyd's algorithm in Python in two dimensions ($d = 2$).

Given an input `X` of the form `X=[X[0], X[1], ..., X[n]]` and a `k`, implement Lloyd's algorithm.

Create some data sets to test your implementation on.

See arthurvancamp.github.io/mini-lecture/k-means.py