Clustering

K-means

Haotian Wang and Yiyan Li

School of Software, Shanghai Jiao Tong University

Introduction to clustering

Clustering

Definition

A cluster is a collection of objects which are “similar” between them and are “dissimilar” to the objects belonging to other clusters.

Definition

Clustering is the algorithm that recognizes clusters from a given data set.

Clustering

Definition

A cluster is a collection of objects which are "similar" between them and are "dissimilar" to the objects belonging to other clusters.

Definition

Clustering is the algorithm that recognizes clusters from a given data set.

Part of common application domains in which the clustering problem arises are as follows:

- Multimedia Data Analysis
- Responding to public health crises
- Intermediate Step for other fundamental data mining problems
- Intelligent Transportation

K-means Algorithm

The k-means clustering problem is one of the oldest and most important questions in all of computational geometry.

Given an integer k and a set of n data points in $\mathbb{R}^d$, the goal of this problem is to choose k centers so as to minimize the total squared distance between each point and its closest center.

The most common K-means algorithm was first proposed by Stuart Lloyd of Bell Labs in 1957.

The objective function to minimize is the within-cluster sum of squares (WCSS) cost:

$$\mathrm{Cost}(C_{1:k}, c_{1:k}) = \sum_{i=1}^{k} \sum_{x \in C_i} \|x - c_i\|^2$$

where c_i is the centroid of cluster

Definition

Cluster centroid is the middle of a cluster.
A centroid is a vector that contains one number for each variable, where each number is the mean of a variable for the observations in that cluster.
The centroid can be thought of as the multi-dimensional average of the cluster.

Lemma

Let C be a cluster of points with its mean to be μ, and let c to be and arbitrary point.Then $\sum_{x \in C} \|x - c\|^2 = \sum_{x \in C} \|x - \mu\|^2 + |C| \cdot \|c - \mu\|^2$

So we denote that:

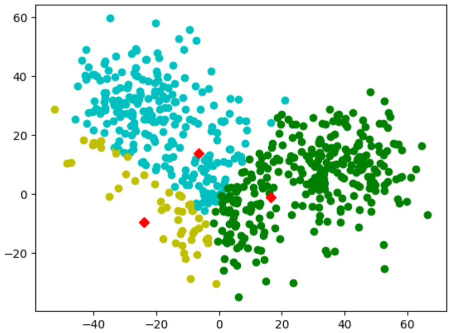
$$
\begin{aligned}
\mathrm{Cost}(C_{1:k}, c_{1:k}) &= \sum_{i=1}^{k} \sum_{x \in C_i} \|x - c_i\|^2 \\
&= \sum_{i=1}^{k} \Big(\sum_{x \in C_i} \|x - \mu_i\|^2 + |C_i| \cdot \|c_i - \mu_i\|^2 \Big) \\
&= \mathrm{Cost}(C_{1:k}, \mathrm{mean}(C_{1:k})) + \sum_{i=1}^{k} |C_i| \cdot \|c_i - \mu_i\|^2
\end{aligned}
$$

Toward a K-means Algorithm

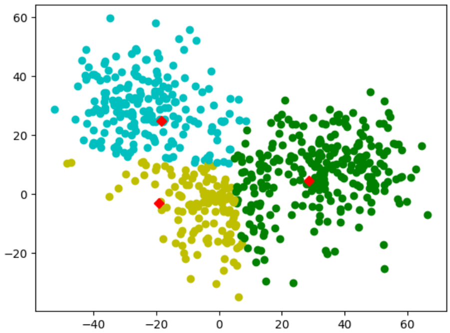
The k-means algorithm iteratively calculates the sum of distance within a cluster and updates the partition.

1. Arbitrarily choose and initial k centroids $\mathcal{C} = \{c_1, c_2 \dots c_k\}$
2. For each $i \in \{1, 2 \dots k\}$, set the cluster C_i to be the set of points that are closer to c_i than they are to c_j for all $j \neq i$
3. For each $i \in \{1, 2 \dots k\}$, set c_i to be the center of all points in C_i where $c_i = \frac{1}{|C_i|} \sum_{x \in C_i} x$
4. Repeat Step 2 and Step 3 until $\mathcal{C}$ no longer changes.

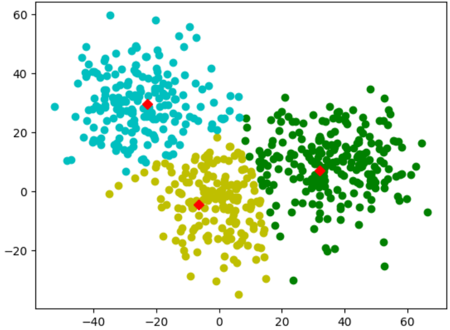
Iteration = 1



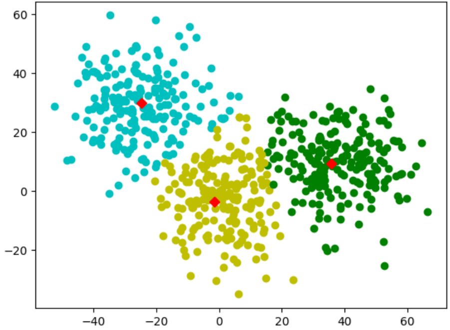
Iteration = 2



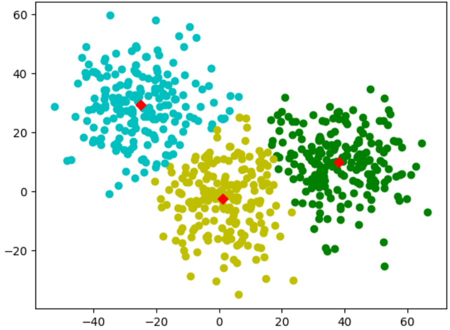
Iteration = 3



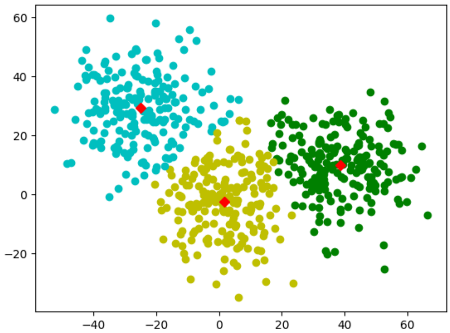
Iteration $= 4$



Iteration = 5



Iteration = 6



Convergence of K-means

Proof.

During the course of the k-means algorithm, the cost monotonically decreases. □

Proof.

During the course of the k-means algorithm, the cost monotonically decreases. $\square$

Let $c_1^{(t)} \ldots c_k^{(t)}, C_1^{(t)} \ldots C_k^{(t)}$ denote the cetnroids and clusters at the start of t^{th} iteration of K-means.

Proof.

During the course of the k-means algorithm, the cost monotonically decreases. □

Let $c_1^{(t)} \ldots c_k^{(t)}, C_1^{(t)} \ldots C_k^{(t)}$ denote the cetnroids and clusters at the start of t^{th} iteration of K-means.

The first step of the iteration assigns each point to its nearest center, therefore:

$$\mathrm{Cost}(C_{1:k}^{(t+1)}, c_{1:k}^{(t)}) \leq \mathrm{Cost}(C_{1:k}^{(t)}, c_{1:k}^{(t)})$$

Proof.

During the course of the k-means algorithm, the cost monotonically decreases. $\square$

Let $c_1^{(t)} \dots c_k^{(t)}, C_1^{(t)} \dots C_k^{(t)}$ denote the cetnroids and clusters at the start of t^{th} iteration of K-means.

The first step of the iteration assigns each point to its nearest center, therefore:

$$\mathrm{Cost}(C_{1:k}^{(t+1)}, c_{1:k}^{(t)}) \leq \mathrm{Cost}(C_{1:k}^{(t)}, c_{1:k}^{(t)})$$

On the second step, each cluster re-centered at its mean. By lemma above:

$$\mathrm{Cost}(C_{1:k}^{(t+1)}, c_{1:k}^{(t+1)}) \leq \mathrm{Cost}(C_{1:k}^{(t+1)}, c_{1:k}^{(t)})$$

Time Complexity

Proof.

Naive K-means algorithm's time complexity is O(kni) □

Proof.

Naive K-means algorithm's time complexity is O(kni) □

In each iteration there are such steps:

- Distance calculation: To calculate the distance from a point to the centroid, we can use the squared Euclidean proximity function, which is thought to be O(1)
- Comparisons betweeen distances.
- Centroid calculation.

Time Complexity

Proof.

Naive K-means algorithm's time complexity is $O(kni)$ $\square$

So the total number of operations in one iteration is:

$$\begin{aligned} C &= \text{distance calculation} + \text{comparisons} + \text{centroids calculation} \\ &= k * n * O(1) + (k-1) * n * O(1) + k * n * O(1) \\ &= O(kn) \end{aligned}$$

where k denotes the number of clusters, n denotes the count of data vectors and d denotes vector dimension.

And the whole process takes i iterations in total so the time complexity of K-means algorithm is $O(kni)$.