4. Computational Complexity of K-Means and K-Medoids

Computation Complexity of K-Means and K-Medoids

0:00 / 0:00 1.25x

Video
Download video file

Transcripts
Download SubRip (.srt) file
Download Text (.txt) file

Computational Complexity of K-Means

1/1 point (graded)
Remember that the K-Means algorithm is given by

1. Randomly select $z_1, \ldots, z_K$
2. Iterate
    1. Given $z_1, \ldots z_K$, assign each $x^{(i)}$ to the closest z_j, so that

    $$\text{Cost}\left(z_1, \ldots z_K\right) = \sum_{i=1}^{n} \min_{j=1,\ldots,k} \left\| x^{(i)} - z_j \right\|^2$$

    2. Given $C_1, \ldots, C_K$ find the best representatives $z_1, \ldots, z_K$, i.e. find $z_1, \ldots, z_K$ such that

    $$z_j = \frac{\sum_{i \in C_j} x^{(i)}}{|C_j|}$$

Assuming that there are n data points $\left\{x_1, \ldots, x_n\right\}$, K clusters and representatives,and each $x_i \in \left\{x_1, \ldots, x_n\right\}$ is a vector of dimension d, what is the computational complexity for one complete iteration of the k-means algorithm? That is, find the time (or the number of steps) it takes to complete steps 2.1 and 2.2.

Note on Big-O notation

We often describe computational complexity using the “Big-O" notation. For example, if the number of steps involved is $5n^2 + n + 1$, then we say it is “of order n^2" and denote this by $\mathcal{O}(n^2)$. When n is large, the highest order term $5n^2$ dominates and we drop the scaling constant 5.

More formally, a function $f(n)$ is of order $g(n)$, and we write $f(n) \sim \mathcal{O}(g(n))$, if there exists a constant C such that

$$f(n) < Cg(n) \qquad \text{as } n \text{ grows large.}$$

In other words, the function f does not grow faster than the function g as n grows large.

The big-O notation can be used also when there are more input variables. For example, in this problem, the number of steps necessary to complete one iteration depends on the number of data points n, the number of clusters K, the dimension d of each vector x_i. Hence, the number of steps required are of $\mathcal{O}(g(n, K, d))$ for some function $g(n, K, d)$.

Hide

- ○ $\mathcal{O}(n)$
- ○ $\mathcal{O}(nK)$
- ○ $\mathcal{O}(nK^2)$
- ◉ $\mathcal{O}(ndK)$

✔

Solution:

In line 2.1, we go through each of the n x_i, and iterate through each of the k z_j's for each x_i (to find the closest z_j). This iteration is $\mathcal{O}(nK)$. And because each x_i has length d, the total iteration is $\mathcal{O}(ndK)$.

Line 2.2 is similar.

Note that because 2.1 and 2.2 both take $\mathcal{O}(ndK)$, one complete iteration takes $\mathcal{O}(ndK)$.

Submit You have used 1 of 2 attempts

Answers are displayed within the problem

Computational Complexity of K-Medoids

2/2 points (graded)

Remember that the K-Medoids algorithm is given by

1. Randomly select $z_1, \ldots, z_K$
2. Iterate
    1. Given $z_1, \ldots z_K$, assign each $x^{(i)}$ to the closest z_j, so that

$$\text{Cost}(z_1, \ldots z_K) = \sum_{j=1}^{n} \min_{j=1,\ldots,k} \text{dist}(x^{(i)}, z_j)$$

    2. Given $C_j \in \{C_1, \ldots, C_K\}$ find the best representative $z_j \in \{x_1, \ldots, x_n\}$ such that

$$\sum_{x^{(i)} \in C_j} \text{dist}(x^{(i)}, z_j)$$

is minimal.

What is the complexity of step 2.1?

- ( ) $\mathcal{O}(n)$
- ( ) $\mathcal{O}(nK)$
- ( ) $\mathcal{O}(nK^2)$
- (•) $\mathcal{O}(ndK)$ ✔

Now what is the complexity of step 2.2?

- ( ) $\mathcal{O}(ndK)$
- ( ) $\mathcal{O}(nK^2)$
- ( ) $\mathcal{O}(nk^2d)$
- (•) $\mathcal{O}(n^2dK)$ ✔

Solution:

Note that step 2.1 of the K-Medoids is the same as that of K-Means, so the time complexity is $\mathcal{O}(ndK)$. Note that step 2.2 of K-Medoids has an additional loop of iterating through the n points $z_j \in \{x_1, \ldots, x_n\}$ which takes $\mathcal{O}(n)$. Thus step 2.2 takes $\mathcal{O}(n^2dK)$.

Submit You have used 2 of 3 attempts

Answers are displayed within the problem

Discussion

Add a Post

Hide Discussion

Topic: Unit 4 Unsupervised Learning (2 weeks) :Lecture 14. Clustering 2 / 4. Computational Complexity of K-Means and K-Medoids

Show all posts

by recent activity

Should the computation complexity of K-Medoids be O(n^2*d) instead of O(n^2*k*d)?

At step 2.2, at each cluster C_j, the number of distance computation is |C_j| * n, so the total ...

3

Doesn't the complexity of calculating the distance depend on the choice of the norm?

I had to think twice about that and recall that a norm is always associated with a positive defi...

3

Community TA