k-means Clustering

Shivaram Kalyanakrishnan

February 20, 2019

Abstract

We introduce the k-means clustering problem, describe the k-means clustering algorithm, and provide a proof of convergence for the algorithm.

1 The k-means Clustering Problem

We are given a data set $(\boldsymbol{x}^1, \boldsymbol{x}^2, \ldots, \boldsymbol{x}^n)$, where for $i \in \{1, 2, \ldots, n\}$, $\mathbf{x}^i \in \mathbb{R}^d$. Here $d \geq 2$ is the dimension of the data set. We are also specified an integer $k \geq 2$. The objective of k-means clustering is to partition the data set into k clusters, such that each cluster is as "tight" as possible. We define this objective more precisely.

A *clustering* $\mathcal{C} : \{1, 2, \ldots, n\} \rightarrow \{1, 2, \ldots, k\}$ assigns one of k clusters to each point in the data set. Each cluster $k' \in \{1, 2, \ldots, k\}$ is also associated with a centre $\boldsymbol{\mu}_{k'} \in \mathbb{R}^d$. If we take a clustering $\mathcal{C}$ along with the sequence $\boldsymbol{\mu}$ representing the centres of its k clusters—$\boldsymbol{\mu} = (\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_k)$—we can define "tightness" in terms of the aggregate distance between the data points and the centres of the clusters to which they are assigned by $\mathcal{C}$. If $\mathcal{C}(i)$ is the cluster in $\{1, 2, \ldots, k\}$ to which $\mathcal{C}$ assigns input point i, the Euclidean distance between the point and its cluster center is $\|\boldsymbol{x}^i - \boldsymbol{\mu}_{\mathcal{C}(i)}\|$. The most common measure of the tightness of a clustering $\mathcal{C}$ (along with cluster centres $\boldsymbol{\mu}$) is the sum squared error (SSE), defined as

$$\sum_{i=1}^{n} \|\boldsymbol{x}^i - \boldsymbol{\mu}_{\mathcal{C}(i)}\|^2.$$

Other definitions of tightness may also be used, but this particular one enjoys nice mathematical properties, as we shall shortly see.

The k-means clustering problem is the problem of finding a clustering among the set of all clusterings, along with a sequence of cluster centres, such that the corresponding SSE is minimal. Unfortunately, even for $k = 2$, this problem is NP-hard for general d and n [2]. If we revise our aim to find a "reasonable", rather than optimal, clustering, it turns out we can do quite nicely by applying the k-means clustering algorithm. This algorithm is an iterative one, which provably converges to a local minimum.

2 k-means Clustering Algorithm

Before we specify the k-means clustering algorithm, we settle one relevant matter. Recall that a clustering algorithm must return both a clustering and a centre for each cluster. The following lemma shows that for any fixed clustering, the SSE is minimised when the centre associated with each cluster is the mean (or centroid) of the set of points assigned to that cluster.

Lemma 1. *Consider the points $\boldsymbol{z}^1, \boldsymbol{z}^2, \ldots, \boldsymbol{z}^m$, where $m \geq 1$, and for $i \in \{1, 2, \ldots, m\}$, $\boldsymbol{z}^i \in \mathbb{R}^d$. Let $\bar{\boldsymbol{z}} = \frac{1}{m}\sum_{i=1}^m \boldsymbol{z}^i$ be the mean of these points, and let $\boldsymbol{z} \in \mathbb{R}^d$ be an arbitrary point in the same (d-dimensional) space. Then*

$$\sum_{i=1}^m \|\boldsymbol{z}^i - \boldsymbol{z}\|^2 \geq \sum_{i=1}^m \|\boldsymbol{z}^i - \bar{\boldsymbol{z}}\|^2.$$

Proof.

$$\begin{aligned}
\sum_{i=1}^m \|\boldsymbol{z}^i - \boldsymbol{z}\|^2 &= \sum_{i=1}^m \|(\boldsymbol{z}^i - \bar{\boldsymbol{z}}) + (\bar{\boldsymbol{z}} - \boldsymbol{z})\|^2 \\
&= \sum_{i=1}^m \big(\|\boldsymbol{z}^i - \bar{\boldsymbol{z}}\|^2 + \|\bar{\boldsymbol{z}} - \boldsymbol{z}\|^2 + 2(\boldsymbol{z}^i - \bar{\boldsymbol{z}}) \cdot (\bar{\boldsymbol{z}} - \boldsymbol{z})\big) \\
&= \sum_{i=1}^m \|\boldsymbol{z}^i - \bar{\boldsymbol{z}}\|^2 + \sum_{i=1}^m \|\bar{\boldsymbol{z}} - \boldsymbol{z}\|^2 + 2\sum_{i=1}^m (\boldsymbol{z}^i \cdot \bar{\boldsymbol{z}} - \boldsymbol{z}^i \cdot \boldsymbol{z} - \bar{\boldsymbol{z}} \cdot \bar{\boldsymbol{z}} + \bar{\boldsymbol{z}} \cdot \boldsymbol{z}) \\
&= \sum_{i=1}^m \|\boldsymbol{z}^i - \bar{\boldsymbol{z}}\|^2 + m\|\bar{\boldsymbol{z}} - \boldsymbol{z}\|^2 + 2(m\bar{\boldsymbol{z}} \cdot \bar{\boldsymbol{z}} - m\bar{\boldsymbol{z}} \cdot \boldsymbol{z} - m\bar{\boldsymbol{z}} \cdot \bar{\boldsymbol{z}} + m\bar{\boldsymbol{z}} \cdot \boldsymbol{z}) \\
&= \sum_{i=1}^m \|\boldsymbol{z}^i - \bar{\boldsymbol{z}}\|^2 + m\|\bar{\boldsymbol{z}} - \boldsymbol{z}\|^2 \\
&\geq \sum_{i=1}^m \|\boldsymbol{z}^i - \bar{\boldsymbol{z}}\|^2.
\end{aligned}$$

□

The k-means clustering algorithm, shown below, is rather straightforward. We begin with an arbitrary clustering, and in line with Lemma 1, set the cluster centres to be the means of the points in each cluster. Thereafter, we examine each point. If it so happens that the closest cluster centre to a point is not the centre of its current cluster, the point is shifted to the cluster to whose centre it is closest. The change in cluster assignments now calls for a corresponding recalculation of the cluster centres; this process iterates until convergence.

k-means Clustering Algorithm

Let $\mathcal{C}(0)$ be an arbitrary clustering, and let $\boldsymbol{\mu}(0) = (\boldsymbol{\mu}^1, \boldsymbol{\mu}^2, \ldots, \boldsymbol{\mu}^k)$ be a sequence of centres such that for $k' \in \{1, 2, \ldots, k\}$, $\boldsymbol{\mu}(0)_{k'}$ is the centroid of the points in the k'-th cluster.
$t \leftarrow 0$.
converged $\leftarrow$ false.
While ¬converged
    converged $\leftarrow$ true.
    for $i \in \{1, 2, \ldots, n\}$
        $\mathcal{C}(t+1)(i) \leftarrow \mathcal{C}(t)(i)$.
        for $k' \in \{1, 2, \ldots, k\}$
            If $k' \neq \mathcal{C}(t+1)(i)$ and $\|\boldsymbol{x}^i - \boldsymbol{\mu}(t)_{k'}\| < \|\boldsymbol{x}^i - \boldsymbol{\mu}(t)_{\mathcal{C}(t+1)(i)}\|$
                $\mathcal{C}(t+1)(i) \leftarrow k'$.
                converged $\leftarrow$ false.
    for $k' \in \{1, 2, \ldots, k\}$
        Set $\boldsymbol{\mu}(t+1)_{k'}$ to be the centroid of all points i such that $\mathcal{C}(t+1)(i) = k'$.
    $t \leftarrow t + 1$.
Return $\mathcal{C}(t), \boldsymbol{\mu}(t)$.

As per the procedure outlined above, it is entirely possible to achieve clusterings that assign *no* points to some of the k clusters. In such a case, the corresponding cluster centre can be set arbitrarily (since the mean is undefined). In practice, though, it is common to use all k clusters effectively—for instance, one could set the centre of an empty cluster to be one of the points in the data set, which would ensure that the cluster will not be empty in the next iteration.

It should also be noted that the choice of the initial clustering, $\mathcal{C}(0)$, can make a significant difference to the SSE of the final clustering obtained. Specialised initialisation strategies (such as **k-means++** [1]) are often used to good effect. It exceeds the scope of this discussion to describe initialisation procedures in detail. Rather, we proceed to prove that regardless of the initialisation, the algorithm will necessarily converge.

Theorem 2. *The k-means clustering algorithm converges.*

Proof. Suppose that the algorithm proceeds from iteration t to iteration $t+1$. It suffices to show that $\mathrm{SSE}(\mathcal{C}(t+1),\ \boldsymbol{\mu}(t+1)) < \mathrm{SSE}(\mathcal{C}(t),\ \boldsymbol{\mu}(t))$. To see why, consider that if that was true, no clustering can be visited twice; since the number of possible clusterings is finite (k^n), the algorithm must necessarily terminate. By the construction of the algorithm, we know that it terminates when no point has a cluster centre closer than the centre of its current cluster: in other words, the current clustering is *locally* optimal.

We show that $\mathrm{SSE}(\mathcal{C}(t+1),\ \boldsymbol{\mu}(t+1)) < \mathrm{SSE}(\mathcal{C}(t),\ \boldsymbol{\mu}(t))$ in two steps. First, we show that

$$\mathrm{SSE}(\mathcal{C}(t+1), \boldsymbol{\mu}(t)) < \mathrm{SSE}(\mathcal{C}(t), \boldsymbol{\mu}(t)), \tag{1}$$

and next, we show that

$$\mathrm{SSE}(\mathcal{C}(t+1), \boldsymbol{\mu}(t+1)) \leq \mathrm{SSE}(\mathcal{C}(t+1), \boldsymbol{\mu}(t)). \tag{2}$$

The first step follows from the logic of the algorithm: $\mathcal{C}(t)$ and $\mathcal{C}(t+1)$ are different only if there is a point that finds a closer cluster centre in $\boldsymbol{\mu}(t)$ than the one assigned to it by $\mathcal{C}(t)$:

$$\mathrm{SSE}(\mathcal{C}(t+1), \boldsymbol{\mu}(t)) = \sum_{i=1}^{n} \|\boldsymbol{x}^i - \boldsymbol{\mu}(t)_{C(t+1)(i)}\|^2 < \sum_{i=1}^{n} \|\boldsymbol{x}^i - \boldsymbol{\mu}(t)_{C(t)(i)}\|^2 = \mathrm{SSE}(\mathcal{C}(t), \boldsymbol{\mu}(t)).$$

The second step puts Lemma 1 to use:

$$\begin{aligned}
\mathrm{SSE}(\mathcal{C}(t+1), \boldsymbol{\mu}(t+1)) &= \sum_{i=1}^{n} \|\boldsymbol{x}^i - \boldsymbol{\mu}(t+1)_{C(t+1)(i)}\|^2 \\
&= \sum_{k'=1}^{k} \sum_{i \in \{1,2,\ldots,n\}, \mathcal{C}(t+1)(i)=k'} \|\boldsymbol{x}^i - \boldsymbol{\mu}(t+1)_{C(t+1)(i)}\|^2 \\
&< \sum_{k'=1}^{k} \sum_{i \in \{1,2,\ldots,n\}, \mathcal{C}(t+1)(i)=k'} \|\boldsymbol{x}^i - \boldsymbol{\mu}(t)_{C(t+1)(i)}\|^2 \\
&= \sum_{i=1}^{n} \|\boldsymbol{x}^i - \boldsymbol{\mu}(t)_{\mathcal{C}(t+1)(i)}\|^2 \\
&= \mathrm{SSE}(\mathcal{C}(t+1), \boldsymbol{\mu}(t)).
\end{aligned}$$

□

References

[1] David Arthur and Sergei Vassilvitskii. k-means++: the advantages of careful seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2007)*, pages 1027–1035. SIAM, 2007.

[2] Sanjoy Dasgupta. The hardness of k-means clustering. Technical Report CS2008-0916, Department of Computer Science and Engineering, University of California, San Diego, 2008. Available at `http://cseweb.ucsd.edu/~dasgupta/papers/kmeans.pdf`.