k-means

Authors

Alexander Jung*, Department of Computer Science, Aalto University, Espoo, Finland
*Corresponding author: alex.jung@aalto.fi

Synonyms

k-means clustering

Abstract

The k-means method is a hard clustering method for data points with numeric feature vectors. It partitions a dataset into k clusters. Each cluster is represented in the feature space by a cluster centroid. The quality of a clustering is measured by the clustering error: the average squared Euclidean distance between a data point and the nearest cluster centroid. Minimizing the clustering error is a non-convex and NP-hard optimization problem. Practical k-means methods use an approximate iterative optimization method known as Lloyd's algorithm: a fixed-point iteration that alternates between assigning each data point to its nearest cluster centroid and re-averaging. No iteration increases the clustering error, and the iterates reach a fixed point after finitely many iterations.

Definition

Consider the task of grouping the days of a year by their weather at some location, say the town Krems in Austria. Each day yields a data point whose feature vector is obtained by stacking temperature measurements recorded during that day, e.g., the minimum and the maximum air temperature. Days in the same group should have similar feature vectors. k-means produces such a grouping: with $k = 2$, it partitions the days into a cold-season and a warm-season cluster.

k-means is an optimization-based hard clustering method for data points with a numeric feature vector (Jung, 2022, Sect. 8.1). It partitions a dataset $\mathcal{D} = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}\}$ with $\mathbf{x}^{(r)} \in \mathbb{R}^d$ into k disjoint clusters indexed by $c = 1, \ldots, k$. Each cluster is represented by a cluster centroid $\mathbf{w}^{(c)} \in \mathbb{R}^d$, obtained by averaging the feature vectors of the data points assigned to that cluster (see Fig. 1 for an example with $k = 2$ cluster centroids). The quality of a given choice of cluster centroids is measured by the clustering error: the average squared Euclidean distance between the feature vector of a data point and the nearest cluster centroid. The k-means principle is to choose cluster centroids that minimize the clustering error,

$$\min_{\mathbf{w}^{(1)},\ldots,\mathbf{w}^{(k)}} \frac{1}{m} \sum_{r=1}^{m} \min_{c=1,\ldots,k} \left\| \mathbf{x}^{(r)} - \mathbf{w}^{(c)} \right\|_2^2.$$

The inner minimization assigns each data point to its nearest cluster centroid. The outer minimization chooses the cluster centroids so that the average squared Euclidean distance (between each data point and its nearest cluster centroid) is small.

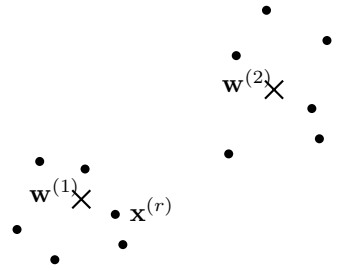

Figure 1: Scatterplot of data points with feature vectors $\mathbf{x}^{(r)} \in \mathbb{R}^2$ for $r = 1, \ldots, m$. The crosses mark the two cluster centroids $\mathbf{w}^{(1)}, \mathbf{w}^{(2)}$ produced by k-means with $k = 2$

Solving the k-means optimization problem exactly is NP-hard (Mahajan et al., 2009). The difficulty of the optimization problem is also reflected by the fact that the objective function is non-convex in the cluster centroids. With a single data point at $x = 0$ and two scalar cluster centroids $w^{(1)}, w^{(2)} \in \mathbb{R}$, the objective is $f(w^{(1)}, w^{(2)}) = \min\big((w^{(1)})^2, (w^{(2)})^2\big)$. Both $(w^{(1)}, w^{(2)}) = (1, 0)$ and $(0, 1)$ are global optima with value 0, meaning at least one centroid perfectly coincides with the data point x, yet their midpoint $(0.5, 0.5)$ gives $f = 0.25 > 0$. A convex combination of optima is therefore worse than either optimum, which violates the defining property of a convex function.

Practical implementations use iterative alternating-minimization methods that converge to a local optimum. A widely used choice is Lloyd's algorithm, which alternates between assigning each data point to its closest cluster centroid and recomputing each cluster centroid as the mean of the currently assigned data points. One iteration of Lloyd's algorithm is the application of an operator $\mathcal{F}$ to the stacked cluster centroids $\mathbf{w} := \big(\mathbf{w}^{(1)}, \ldots, \mathbf{w}^{(k)}\big)$, so the method is the fixed-point iteration $\mathbf{w}^{(t+1)} = \mathcal{F}\big(\mathbf{w}^{(t)}\big)$. Its fixed points are cluster centroids that each coincide with the mean of the data points assigned to them. Applying $\mathcal{F}$ never increases the clustering error, and since a finite dataset admits only finitely many partitions, the iterates reach a fixed point after finitely many iterations (Selim and Ismail, 1984).

The k-means principle is a special case of the empirical risk minimization (ERM) principle. Indeed, k-means uses a hypothesis space $\mathcal{H}_k$ that consists of all maps $h : \mathcal{X} \to \mathcal{X}$ of the piecewise-constant form

$$h(\mathbf{x}) = \mathbf{w}^{(\,c^\star(\mathbf{x}))}, \qquad c^\star(\mathbf{x}) = \operatorname*{arg\,min}_{c=1,\ldots,k} \Big\| \mathbf{x} - \mathbf{w}^{(c)} \Big\|_2,$$

parameterized by the k cluster centroids $\mathbf{w}^{(1)}, \ldots, \mathbf{w}^{(k)} \in \mathcal{X}$. Each such hypoth-

esis maps every feature vector to its nearest cluster centroid. The loss function is the squared Euclidean distance between the feature vector of each data point and its nearest cluster centroid, $L(\mathbf{x}, h) = \|\mathbf{x} - h(\mathbf{x})\|_2^2$. With these choices, ERM on $\mathcal{H}_k$ recovers the k-means optimization problem above (Jung, 2022, Sect. 8.1).

Applications of k-means include customer segmentation (grouping the customers of a retailer by feature vectors such as monthly spending and number of purchases), outlier detection, compression, and image segmentation. Fig. 2 illustrates the latter two on a subsampled photo of the Ötscher massif: k-means on the pixel colors with $k = 4$ replaces each pixel's color by the nearest of k palette colors, compressing 24 bits per pixel to 2 bits per pixel plus a small palette; with $k = 2$, the cluster assignments form a segmentation mask that separates the sky and the summit from the vegetation.

(a) (b) (c)

Figure 2: Image compression and image segmentation via k-means on pixel colors. (a) A photo of the Ötscher massif, subsampled to 97×129 pixels. (b) Each pixel's color replaced by the nearest of $k = 4$ palette colors, compressing 24 bits per pixel to 2 bits per pixel plus the palette — a compression factor of approximately 12. (c) Segmentation mask from $k = 2$: sky and summit (white) versus vegetation (black). Data generated by `pythondemos/kmeans.py`

Cross References

hard clustering, cluster, cluster centroid, clustering error, Lloyd's algorithm, k-means++, Euclidean distance, optimization problem, ERM, hypothesis space, loss function

References

1. Jung (2022). Machine Learning: The Basics. Springer Nature, Singapore, Singapore.
2. Mahajan et al. (2009). The Planar k-Means Problem is NP-Hard. In: WALCOM: Algorithms and Computation, pp. 274–285. Springer-Verlag.
3. Selim and Ismail (1984). K-Means-Type Algorithms: A Generalized Convergence Theorem and Characterization of Local Optimality. IEEE Trans. Pattern Anal. Mach. Intell., 6(1), pp. 81–87.