K-Means Algorithm for Clustering

Robert Jacobs
Department of Brain & Cognitive Sciences
University of Rochester
Rochester, NY 14627, USA

August 22, 2008

This is a very simple (and very common) iterative algorithm for clustering a set of data items. Suppose we have a set of N data items $\{\mathbf{x}_i\}_{i=1}^N$ which we would like to cluster into K clusters (note that the number of clusters is selected by the user of the algorithm—that is, it is not determined by the algorithm). Each cluster is represented by a "prototypical" vector, denoted $\boldsymbol{\mu}_j$ for the j^{th} cluster. The algorithm seeks to partition the data items into disjoint subsets $\mathcal{S}_j$ containing N_j data items in such a way so as to minimize the function

$$J = \sum_{j=1}^{K} \sum_{i \in \mathcal{S}_j} \|\mathbf{x}_i - \boldsymbol{\mu}_j\|^2 \tag{1}$$

where $\boldsymbol{\mu}_j$ is the mean of the data items in set $\mathcal{S}_j$ and is given by

$$\boldsymbol{\mu}_j = \frac{1}{N_j} \sum_{i \in \mathcal{S}_j} \mathbf{x}_i. \tag{2}$$

The algorithm begins by first assigning random values to the prototypical vectors $\{\boldsymbol{\mu}_j\}_{j=1}^K$. Next the data items are assigned to disjoint subsets according to each item's nearest prototypical vector: if $\boldsymbol{\mu}_j$ is the closest prototypical vector to data item $\mathbf{x}_i$, then $\mathbf{x}_i$ is assigned to subset $\mathcal{S}_j$. (As an aside, note that this is a "hard" assignment—a data point is assigned to one and only one subset.) Then the prototypical vectors are re-computed using Equation 2. Then the data items are re-assigned to subsets based on the closest prototypical vector to each item. This procedure is repeated until there is no further change in the grouping of the data items. It can be shown that at each iteration the value of J (Equation 1) will not increase.