Experimentation with Clustering Algorithms

Corinne Curcie and Shivam Naik

March 5, 2017

Abstract

Clustering algorithms are a popular tool for making sense of big data. Our project involved implementing various algorithms, focusing specifcally on high-dimensional data, to gain better understanding .

1. Introduction
2. Related Work
3. Algorithms Implemented

K-Subspaces

We were intrigued by a 2009 paper that proposed a "K-Subspaces" algorithm similar to K-Means (Wang, Ding, and Li 2009). The well-known K-Means algorithm starts by choosing k points in the dataset to be the initial cluster center points, and then updates on an "expectation-maximization mechanism" (EM).
The "E-step" is the cluster assignment step, where points are labelled with a cluster based on which of the center points they are closest to. The "M-step" is the update step, where the k cluster center points are recalculated to be some average of all of the points that were labeled as belonging to that cluster during the previous "E-step" round. The algorithm stops when the change in centers from one round to the next is less than some predetermined threshold. The objective function that K-Means tries to minimize is the sum of squared Euclidean distances from each point to its assigned cluster center:

$$\arg\min_{\mathbf{S}} \sum_{i=1}^{k} \sum_{\mathbf{x} \in S_i} \|\mathbf{x} - \boldsymbol{\mu}_i\|^2$$

The K-Subspaces algorithm is a variant on this well-known technique. Instead of using Euclidean distances for our objective function, multiple distance measures are used during the "E-step" to determine which centers the points are closest to. The authors of the algorithm decided to focus on three possible subspaces - 1D lines, 2D planes, and 3D spheres - and determines distance functions based on those subspaces. By calculating all three distances for each pair of point and cluster center, the algorithm is better able to determine when a point is within a cluster of a non-standard space. For this reason, the performance is hypothesized to be better than the standard K-means algorithm, which does not perform well on certain cluster shapes. For initializing cluster centers before beginning the EM steps, we used the standard K-means++ algorithm, which probabilistically selects initial clusters (MORE DETAILS NEEDED). The K-means++ initialization is also used in scikits implementation of K-means. For parameter eta, a value of 0.35 was used as specified within the 2009 paper.

For our synthetic data, we wanted to demonstrate the ability to cluster for a variety of shapes. Data includes 1D lines, 2D planes, and 3D spheres, and the goal is for K-subspaces to cluster those shapes together separately even when they are close together.

4. Datasets Used

5. Performance