MI tech
Multimedia Imaging Technology

university of groningen

THE UNIVERSITY OF AUCKLAND
FACULTY OF SCIENCE

Recovery Rate of Clustering Algorithms

Fajie Li and Reinhard Klette

Outline and Objective

Abstract

We provide a simple and general way for defining the recovery rate of clustering algorithms using a given family of old clusters for evaluating the performance when calculating a family of new clusters. The recovery rate may be calculated by using an approximate and efficient algorithm.

Our Approach

Our general (!) evaluation of clustering algorithms is very intuitive. Our method does not need to introduce other functions such as, for example, an F-function as in [Larsen/Aone 1999], or entropy as in [Borgelt 2006] or [Crabtree et al. 2005].

Recovery Rate and Exact Calculation

Clustering Algorithm

A clustering algorithm $\mathcal{A}$ maps a finite set $C = \cup_{i=1}^{n} C_i$, with $N = \text{card}\, C$ of old clusters) into a family of (new) clusters:

$$\cup_{k=1}^{m} G_k = C$$

Definition of Recovery Rate

Definition 2. *Assume that* $G_{1_{t'_1}}, G_{2_{t'_2}}, \ldots, G_{m_{t'_m}}$ *satisfy*

(i) For $i, j \in \{1_{t'_1}, 2_{t'_2}, \ldots, m_{t'_m}\}$, *there exist two old clusters* C_i *and* C_j *such that* $G_{i_{t'_i}} \subseteq C_i$ *and* $G_{j_{t'_j}} \subseteq C_j$; *and*

(ii) $\sum_{k=1}^{m} \frac{cardG_{k_{t'_k}}}{cardC_k} = \max\{\sum_{k=1}^{m} \frac{cardG_{k_{t_k}}}{cardC_k} : t_k = 1, 2, \ldots, s_k\}$

The value

$$\frac{\sum_{k=1}^{m} \frac{cardG_{k_{t'_k}}}{cardC_k}}{m} \times 100\%$$

is called the recovery rate *of the clustering algorithm* $\mathcal{A}$ *with respect to the input* $\cup_{i=1}^{n} C_i$.

Algorithm 1: Exact Recovery Rate

Input: Old clusters C_i, where $i = 1, 2, \ldots, n$; and new clusters G_j, where $j = 1, 2, \ldots, m$, obtained from a clustering algorithm $\mathcal{A}$.

Output: The recovery rate of $\mathcal{A}$ with respect to C_i, where $i = 1, 2, \ldots, n$.

1. Let M be an $m \times n$ matrix, initially with zeros in all of its elements.
2. For each $j \in \{1, 2, \ldots, m\}$ and for each $x \in G_j$, if there exists an $i \in \{1, 2, \ldots, n\}$ such that $x \in C_i$, then update M as follows: $M(j,i) = M(j,i) + 1$, where $M(j,i)$ is the (j,i)-th entry of M.
3. Find m different integers (i.e., column indices) $i_k \in \{1, 2, \ldots, n\}$ such that

$$\sum_{k=1}^{m} \frac{M(k,i_k)}{cardC_{i_k}} = \max\{\sum_{j=1}^{m} \frac{M(j,i_j)}{cardC_{i_j}} : i_j \in \{1, 2, \ldots, n\}\}$$

4. Output the recovery rate as being the value

$$\frac{\sum_{k=1}^{m} \frac{M(k,i_k)}{cardC_{i_k}}}{m} \times 100\%$$

Time-Efficient Approximate Calculation

This exact algorithm requires exponential run-time. The following is only an approximate algorithm for computing the recovery rate, but with $O(mn)$ run-time. Below we also provide an example (from astronomy: synthesized galaxies) for applying these two algorithms. There are five old clusters with 10,000 3D points each. The figure shows a 2D projection of the union of all five old clusters. We apply an adaptive mean-shift clustering algorithm (see Georgescu et al. 2003), but further optimized (see Li/Klette, ICONIP 2008). The recovery rate equals 58.18%, and the approximate algorithm estimates this rate as 51.76%.

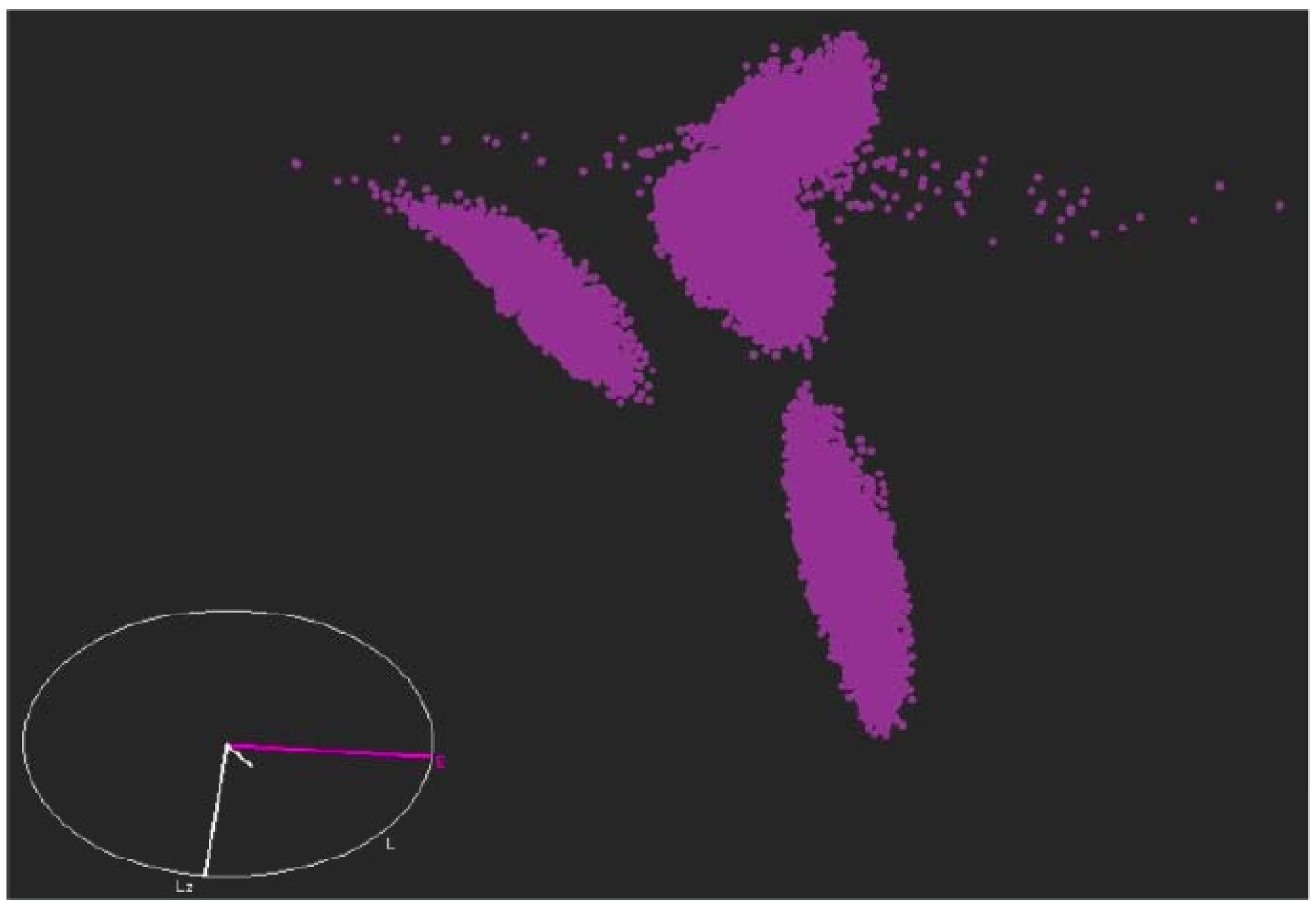
Algorithm 2: Approximate Recovery Rate

Input and Steps 1 and 2 are the same as in Algorithm 1.

Output: The approximate recovery rate of $\mathcal{A}$ with respect to C_i, where $i = 1, 2, \ldots, n$.

3.0. For each entry $M(i,j)$ of M, let $M(i,j) = \frac{M(i,j)}{cardC_j}$, where $i = 1, 2, \ldots, m$; $j = 1, 2, \ldots, n$.
3.1. For each $j \in \{1, 2, \ldots, m\}$, find the maximum entry of M, denoted by $m_j = M(i,k)$.
3.2. Update M by removing the i-th row and k-th column of M and go to Step 3.1.
4. Output the approximate recovery rate as the value

$$\frac{\sum_{j=1}^{m} m_j}{m} \times 100\%$$



Clustering is often used in image and video processing, and the specified technique allows to compare the performance of various clustering algorithms (for image segmentation, learning, bags-of-features representations of images, video retrieval etc.).