Sharma et al. BMC Bioinformatics (2016) 17:319
DOI 10.1186/s12859-016-1184-5

BMC Bioinformatics

RESEARCH ARTICLE Open Access

Stepwise iterative maximum likelihood clustering approach

Alok Sharma[1,2,3*], Daichi Shigemizu[1,2,4], Keith A. Boroevich[1], Yosvany López[1,4], Yoichiro Kamatani[1], Michiaki Kubo[1] and Tatsuhiko Tsunoda[1,2,4*]

Abstract

Background: Biological/genetic data is a complex mix of various forms or topologies which makes it quite difficult to analyze. An abundance of such data in this modern era requires the development of sophisticated statistical methods to analyze it in a reasonable amount of time. In many biological/genetic analyses, such as genome-wide association study (GWAS) analysis or multi-omics data analysis, it is required to cluster the plethora of data into sub-categories to understand the subtypes of populations, cancers or any other diseases. Traditionally, the *k*-means clustering algorithm is a dominant clustering method. This is due to its simplicity and reasonable level of accuracy. Many other clustering methods, including support vector clustering, have been developed in the past, but do not perform well with the biological data, either due to computational reasons or failure to identify clusters.

Results: The proposed SIML clustering algorithm has been tested on microarray datasets and SNP datasets. It has been compared with a number of clustering algorithms. On MLL datasets, SIML achieved highest clustering accuracy and rand score on 4/9 cases; similarly on SRBCT dataset, it got for 3/5 cases; on ALL subtype it got highest clustering accuracy for 5/7 cases and highest rand score for 4/7 cases. In addition, SIML overall clustering accuracy on a 3 cluster problem using SNP data were 97.3, 94.7 and 100 %, respectively, for each of the clusters.

Conclusions: In this paper, considering the nature of biological data, we proposed a maximum likelihood clustering approach using a stepwise iterative procedure. The advantage of this proposed method is that it not only uses the distance information, but also incorporate variance information for clustering. This method is able to cluster when data appeared in overlapping and complex forms. The experimental results illustrate its performance and usefulness over other clustering methods. A Matlab package of this method (SIML) is provided at the web-link http://www.riken.jp/en/research/labs/ims/med_sci_math/.

Background

In an unsupervised learning procedure, the class label of a training sample is not known and the aim is to partition the data into clusters. The unsupervised learning scheme uses the relationship between samples to perform partitioning. In many biological data (e.g. transcriptome data, genomic data etc.), the number of clusters and class labels are unknown. However, the distribution is sometimes known, which is usually normal. Therefore, it would be an advantage to build a technique that utilizes distance and variance information as it can track clusters with different conformations.

Over last several decades, the *k*-means clustering algorithm has been used quite significantly in partitioning the biological data. In the most recent multi-omics data analysis tools, like iCluster, and iClusterPlus [1], the underlying clustering method used was also *k*-means. Some tools in cancer research, like ConsensusCluster (CC) and CCPlus [2, 3], also utilize *k*-means as one of the common clustering algorithms. Though the *k*-means clustering algorithm has been extensively applied [4] due to its simplicity and reasonable level of accuracy, it cannot track clusters when samples of different groups are overlapping to each other (i.e., data points of adjacent groups are spread in a way that the groups partly coincide over each other). In biological data, this is sometimes the case, and thereby leads to clusters which may not be accurate. This has a significant implication

* Correspondence: alok.fj@gmail.com; tatsuhiko.tsunoda@riken.jp
[1]RIKEN Center for Integrative Medical Sciences, Yokohama 230-0045, Japan
Full list of author information is available at the end of the article

in biological findings, particularly in cancer subtypes analysis, population stratification analysis in GWAS and multi-omics data analysis. In general, *k*-means has played a significant role in carrying out analysis for various types of biological data over several years. Since data complexity and quantity are increasing, it is important to develop techniques that can perform clustering by following data topologies.

In the field of unsupervised learning and clustering, several wonderful techniques have emerged. Some of the techniques are briefly summarized here as follows: 1) clustering using some criterion functions e.g. i) sum-of-squared error criterion; ii) related minimum variance criterion, iii) scattering criterion; iv) trace criterion; v) determinant criterion; and, vi) invariant criterion [5, 6]; 2) clustering using iterative optimization [7–9]; 3) hierarchical clustering [10–13]; several hierarchical-based algorithms can be found in the literature; e.g., single-linkage [14], complete-linkage [15], median-linkage [16] and so on. Single linkage (SLink) [14] merges two nearest-neighbor clusters at a time in an agglomerative hierarchical fashion. It uses Euclidean distance to measure the closeness between two clusters (if it is less than an arbitrary threshold). This method is very sensitive to data position, which sometimes creates problem by forming a cluster in a long chain (known as the chaining effect). The complete linkage (CLink) hierarchical approach [15] depends on the farthest-neighbor and reduces the effects of long chains. This technique is also sensitive to outliers. The use of average or median distance could be a way to overcome this sensitiveness. This was done in median linkage (MLink) hierarchical approach [16]; 4) clustering is also performed using Bayes classifier [17–21]; 5) clustering iterative maximum likelihood [22–24]; and, 6) support vector clustering [25–27].

In the recent literature, support vector clustering has gained a lot of attention [26–31]. However, it is expensive in processing time and sometimes fails to find meaningful clusters. In general, clustering methods based on Bayes classifier and maximum likelihood are still the preferred choice compared to support vector clustering for many applications. There are various approaches to implement these clustering methods.

In this paper, we focus on maximum likelihood estimate. There are three ways to implement the maximum likelihood method. 1) Analytic way: likelihood functions are differentiated and equated to zero and the equations are solved to find extrema. The second derivative is then taken to ensure if maxima has reached rather than minima. 2) Grid search: an exhaustive search over a region is conducted to find the parameters that produce largest likelihood. 3) Numerical analysis: an initial value of parameter is used in a hill climbing algorithm or gradient ascent algorithm (e.g. Newton-Rapson, BHHH, DFP) to find the maxima. Maximum likelihood is also estimated via EM algorithm [5, 22, 32–39].

In general, it is impossible to use an analytic approach to find maximum likelihood estimates as the parametric form of data is unknown. Grid search is only possible when the dimensionality of the data is very small. Most of the time, maximum likelihood is computed by a hill climbing algorithm or by the EM algorithm. The potential problem with gradient algorithms is that when likelihood is not differentiable then it is not possible to find gradient to convergence. Considering this, in this paper, we propose a stepwise iterative maximum likelihood (SIML) procedure which does not require derivatives of likelihood functions. It can find all unknown parameters without solving first derivative and second derivatives of likelihood. The experimental results also show promising when compared to many state-of-the-art clustering methods.

Methods

Description of Maximum Likelihood Clustering

Here, we briefly discuss maximum likelihood method for clustering [5]. Assume a d -dimensional sample set $\chi = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ having n unlabelled samples, and, c is the number of clusters. Let $\Omega = \{\omega_j\}$ (for $j = 1, 2, \ldots, c$) be the state of the nature or class label for j th cluster χ_j. Suppose $\boldsymbol{\theta} = \{\boldsymbol{\theta}_j\}$ (for $j = 1 \ldots c$) is any unknown parameter (having mean $\boldsymbol{\mu}$ and covariance Σ). Then the mixture density is given by

$$p(\mathbf{x}|\boldsymbol{\theta}) = \sum_{j=1}^{c} p(\mathbf{x}|\omega_j, \boldsymbol{\theta}_j) P(\omega_j) \tag{1}$$

where $p(\mathbf{x}|\omega_j, \boldsymbol{\theta}_j)$ is the conditional density, and $P(\omega_j)$ is the a priori probability. The log likelihood can be represented by joint density

$$L = \log p(\chi|\boldsymbol{\theta}) = \log \textstyle\prod_{k=1}^{n} p(\mathbf{x}_k|\boldsymbol{\theta}) = \sum_{k=1}^{n} \log p(\mathbf{x}_k|\boldsymbol{\theta}) \tag{2}$$

Suppose that the joint density $p(\chi|\boldsymbol{\theta})$ is differentiable with respect to $\boldsymbol{\theta}$ then from Eqs. 1 and 2

$$\nabla_{\boldsymbol{\theta}_i} L = \sum_{k=1}^{n} \frac{1}{p(\mathbf{x}_k|\boldsymbol{\theta})} \nabla_{\boldsymbol{\theta}_i} \left[\sum_{j=1}^{c} p(\mathbf{x}_k|\omega_j, \boldsymbol{\theta}_j) P(\omega_j) \right] \tag{3}$$

where $\nabla_{\boldsymbol{\theta}_i} L$ is the gradient of L with respect to $\boldsymbol{\theta}_i$. If $\boldsymbol{\theta}_i$ and $\boldsymbol{\theta}_j$ are independent and suppose a posteriori probability is given as

$$P(\omega_i|\mathbf{x}_k, \boldsymbol{\theta}) = \frac{p(\mathbf{x}_k|\omega_i, \boldsymbol{\theta}_i) P(\omega_i)}{p(\mathbf{x}_k|\boldsymbol{\theta})} \tag{4}$$

then from Eqs. 3 and 4, we have

$$\nabla_{\boldsymbol{\theta}_i} L = \sum_{k=1}^{n} P(\omega_i|\mathbf{x}_k, \boldsymbol{\theta}) \nabla_{\boldsymbol{\theta}_i} \log p(\mathbf{x}_k|\omega_i, \boldsymbol{\theta}_i) \tag{5}$$

The gradient of likelihood (Eq. 5) can be equated to zero ($\nabla_{\boldsymbol{\theta}_i} L = 0$) to obtain maximum likelihood estimate $\hat{\boldsymbol{\theta}}_i$. The solution can be therefore obtained by

$$P(\omega_i) = \frac{1}{n} \sum_{k=1}^{n} P\left(\omega_i|\mathbf{x}_k, \hat{\boldsymbol{\theta}}\right) \tag{6}$$

$$\sum_{k=1}^{n} P\left(\omega_i|\mathbf{x}_k, \hat{\boldsymbol{\theta}}\right) \nabla_{\boldsymbol{\theta}_i} \log p\left(\mathbf{x}_k|\omega_i, \hat{\boldsymbol{\theta}}_i\right) = 0 \tag{7}$$

$$P\left(\omega_i|\mathbf{x}_k, \hat{\boldsymbol{\theta}}\right) = \frac{p\left(\mathbf{x}_k|\omega_i, \hat{\boldsymbol{\theta}}_i\right) P(\omega_i)}{\sum_{j=1}^{c} p\left(\mathbf{x}_k|\omega_j, \hat{\boldsymbol{\theta}}_j\right) P(\omega_j)} \tag{8}$$

In the above equations, $\boldsymbol{\theta}$ is replaced by unknown mean and covariance parameters for normal distribution case, to yield maximum likelihood estimates. In the literature, parameter $\boldsymbol{\theta}$ is iteratively updated to reach the final value $\hat{\boldsymbol{\theta}}$ using hill climbing algorithms such as the Newton-Raphson method. In general, the computation of first and second derivatives of likelihood is required to find the solution. If the likelihood is differentiable and the a priori probability is non-zero, then convergence can be obtained. However, there is always a possibility of being trapped in a local optima.

Stepwise iterative maximum likelihood method

In this section, we describe our proposed method. This method seeks the most optimal partitions in an iterative way. We begin with an initial partition of data and shift a sample from one partition to another partition, and test if such a shift improves the overall log-likelihood. A simple illustration of SIML is given in Fig. 1.

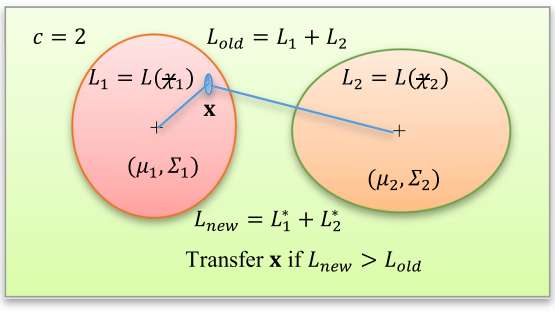

Fig. 1 An illustration of stepwise iterative maximum likelihood method using a $c = 2$ cluster case. In this illustration, two clusters χ_1 and χ_2 are given with likelihood functions L_1 and L_2, respectively. The center of clusters are depicted by μ_1 and μ_2 (shown as '+' inside two clusters). Initial total likelihood is L_{old} which is the sum of two likelihood functions ($L_1 + L_2$). A sample $x \in \chi_1$ is checked for grouping. It is advantageous to shift sample x to cluster χ_2 only if the new likelihood ($L_{new} = L_1^* + L_2^*$) is higher than the old likelihood; i.e., $L_{new} > L_{old}$

If we define class-based log-likelihood of two clusters χ_i and χ_j as

$$L_i = \sum_{\mathbf{x} \in \chi_i} \log[p(\mathbf{x}|\omega_i, \boldsymbol{\theta}_i) P(\omega_i)] \tag{9}$$

and

$$L_j = \sum_{\mathbf{x} \in \chi_j} \log\left[p\left(\mathbf{x}|\omega_j, \boldsymbol{\theta}_j\right) P\left(\omega_j\right)\right], \tag{10}$$

then we would be interested in knowing how the class-based log-likelihood functions (referred as log-likelihood function hereafter) change if a sample is shifted from χ_i to χ_j. In order to know this, let us define mean and covariance of χ_i and χ_j as $\boldsymbol{\mu}_i$ and $\boldsymbol{\mu}_j$, and, as Σ_i and Σ_j, respectively. The following equations describe mean and covariance:

$$\boldsymbol{\mu}_i = \frac{1}{n_i} \sum_{\mathbf{x} \in \chi_i} \mathbf{x} \tag{11}$$

$$\boldsymbol{\mu}_j = \frac{1}{n_j} \sum_{\mathbf{x} \in \chi_j} \mathbf{x} \tag{12}$$

$$\Sigma_i = \frac{1}{n_i} \sum_{\mathbf{x} \in \chi_i} (\mathbf{x} - \boldsymbol{\mu}_i)(\mathbf{x} - \boldsymbol{\mu}_i)^{\mathrm{T}} \tag{13}$$

and

$$\Sigma_j = \frac{1}{n_j} \sum_{\mathbf{x} \in \chi_j} \left(\mathbf{x} - \boldsymbol{\mu}_j\right)\left(\mathbf{x} - \boldsymbol{\mu}_j\right)^{\mathrm{T}} \tag{14}$$

where n_i and n_j are number of samples in χ_i and χ_j, respectively. If the component density is normal and let $P(\omega_i) = n_i/n$ (where n is the total number of samples) then Eqs. 9 and 10 can be written as

$$\begin{aligned} L_i &= \sum_{\mathbf{x} \in \chi_i} \log\left[\frac{1}{(2\pi)^{d/2} |\Sigma_i|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)^{\mathrm{T}} \Sigma_{i^{-1}} (\mathbf{x} - \boldsymbol{\mu}_i)\right]\right] \\ &\quad + n_i \log P(\omega_i) \\ \text{or} &= -\frac{1}{2} tr\left[\Sigma_i^{-1} \sum_{\mathbf{x} \in \chi_i} (\mathbf{x} - \boldsymbol{\mu}_i)(\mathbf{x} - \boldsymbol{\mu}_i)^{\mathrm{T}}\right] - \frac{n_i d}{2} \log 2\pi - \frac{n_i}{2} \log|\Sigma_i| \\ &\quad + n_i \log \frac{n_i}{n} \end{aligned}$$

where $tr()$ is a trace function. Since $tr\left[\Sigma_i^{-1} \sum_{\mathbf{x} \in \chi_i} (\mathbf{x} - \boldsymbol{\mu}_i)(\mathbf{x} - \boldsymbol{\mu}_i)^{\mathrm{T}}\right] = tr(n_i I_{d \times d}) = n_i d$ we can write L_i as

$$L_i = -\frac{1}{2} n_i d - \frac{n_i d}{2} \log 2\pi - \frac{n_i}{2} \log|\Sigma_i| + n_i \log \frac{n_i}{n} \tag{15}$$

Similarly, we can write L_j as

$$L_j = -\frac{1}{2} n_j d - \frac{n_j d}{2} \log 2\pi - \frac{n_j}{2} \log|\Sigma_j| + n_j \log \frac{n_j}{n}, \tag{16}$$

and the total log-likelihood for c clusters can be written as

$$L_{tot} = \sum_{k=1}^{c} L_k \tag{17}$$

where L_k is from Eq. 15 or 16.

If a sample $\hat{\mathbf{x}} \in \chi_i$ is shifted to χ_j, then the mean and covariance will change as follows (from Eqs. 11, 12, 13 and 14):

$$\boldsymbol{\mu}_j^* = \boldsymbol{\mu}_j + \frac{\hat{\mathbf{x}} - \boldsymbol{\mu}_j}{n_j + 1} \tag{18}$$

$$\boldsymbol{\mu}_i^* = \boldsymbol{\mu}_i - \frac{\hat{\mathbf{x}} - \boldsymbol{\mu}_i}{n_i - 1} \tag{19}$$

$$\Sigma_j^* = \frac{n_j}{n_j + 1}\Sigma_j + \frac{n_j}{(n_j + 1)^2}\left(\hat{\mathbf{x}} - \boldsymbol{\mu}_j\right)\left(\hat{\mathbf{x}} - \boldsymbol{\mu}_j\right)^{\mathrm{T}} \tag{20}$$

$$\Sigma_i^* = \frac{n_i}{n_i - 1}\Sigma_i - \frac{n_i}{(n_i - 1)^2}(\hat{\mathbf{x}} - \boldsymbol{\mu}_i)(\hat{\mathbf{x}} - \boldsymbol{\mu}_i)^{\mathrm{T}} \tag{21}$$

where $\boldsymbol{\mu}_i^*$, $\boldsymbol{\mu}_j^*$, Σ_i^* and Σ_j^* are means and covariances after the shift.

In order to find the change in log-likelihood functions L_i and L_j, we first introduce the following Lemma.

Lemma 1 If a sample $\hat{\mathbf{x}} \in \chi_i$ is shifted to cluster χ_j and the changed covariance of χ_j is defined as $\Sigma_j^* = \frac{n_j}{n_j+1}\Sigma_j + \frac{n_j}{(n_j+1)^2}\left(\hat{\mathbf{x}} - \boldsymbol{\mu}_j\right)\left(\hat{\mathbf{x}} - \boldsymbol{\mu}_j\right)^{\mathrm{T}}$ then the determinant of Σ_j^* can be given as $\left|\Sigma_j^*\right| = \left(\frac{n_j}{n_j+1}\right)^d |\Sigma_j| \left(1 + \frac{1}{n_j+1}\left(\hat{\mathbf{x}} - \boldsymbol{\mu}_j\right)^{\mathrm{T}} \Sigma_j^{-1} \left(\hat{\mathbf{x}} - \boldsymbol{\mu}_j\right)\right)$.

Proof By taking determinant of Σ_j^*, we get

$$\left|\Sigma_j^*\right| = \left|\frac{n_j}{n_j + 1}\Sigma_j + \frac{n_j}{(n_j + 1)^2}\left(\hat{\mathbf{x}} - \boldsymbol{\mu}_j\right)\left(\hat{\mathbf{x}} - \boldsymbol{\mu}_j\right)^{\mathrm{T}}\right| \tag{L1}$$

since for $m \times m$ square matrices $|AB| = |A||B|$ and for a scalar c, $|cA| = c^m|A|$. We can write Eq. L1 as

$$\left|\Sigma_j^*\right| = \left(\frac{n_j}{n_j + 1}\right)^d |\Sigma_j| \left|I_{d\times d} + \frac{1}{n_j + 1}\left(\hat{\mathbf{x}} - \boldsymbol{\mu}_j\right)\left(\hat{\mathbf{x}} - \boldsymbol{\mu}_j\right)^{\mathrm{T}} \Sigma_j^{-1}\right| \tag{L2}$$

From Sylvester's determinant theorem, rectangular matrices $A \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{n\times m}$ in $|I_{m\times m} + AB|$ is equal to $|I_{n\times n} + BA|$. Therefore, we can write

$$\left|I_{d\times d} + \frac{1}{n_j + 1}\left(\hat{\mathbf{x}} - \boldsymbol{\mu}_j\right)\left(\hat{\mathbf{x}} - \boldsymbol{\mu}_j\right)^{\mathrm{T}} \Sigma_j^{-1}\right| = 1 + \frac{1}{n_j + 1}\left(\hat{\mathbf{x}} - \boldsymbol{\mu}_j\right)^{\mathrm{T}} \Sigma_j^{-1}\left(\hat{\mathbf{x}} - \boldsymbol{\mu}_j\right) \tag{L3}$$

since $|c| = c$.

Substituting right hand side of Eq. L3 in Eq. L2 proves the Lemma.

As similar to Lemma 1, the determinant of the change in covariance of χ_i can be written as

$$\left|\Sigma_i^*\right| = \left(\frac{n_i}{n_i - 1}\right)^d |\Sigma_i| \left(1 - \frac{1}{n_i - 1}(\hat{\mathbf{x}} - \boldsymbol{\mu}_i)^{\mathrm{T}} \Sigma_i^{-1} (\hat{\mathbf{x}} - \boldsymbol{\mu}_i)\right) \tag{22}$$

We can now observe the change in L_j (Eq. 16) due to the shift of a sample $\hat{\mathbf{x}}$ from χ_i to χ_j as

$$L_j^* = -\frac{1}{2}(n_j + 1)d - \frac{(n_j + 1)d}{2}\log 2\pi - \frac{n_j + 1}{2}\log\left|\Sigma_j^*\right| + (n_j + 1)\log\frac{n_j + 1}{n} \tag{23}$$

From Lemma 1 and Eq. 16, we can rewrite Eq. 23 after doing algebraic manipulation as

$$L_j^* = L_j + (\Delta L_j + C) \tag{24}$$

where ΔL_j is given by

$$\Delta L_j = -\frac{1}{2}\log|\Sigma_j| - \frac{n_j + 1}{2}\log\left(1 + \frac{1}{n_j + 1}\left(\hat{\mathbf{x}} - \boldsymbol{\mu}_j\right)^{\mathrm{T}} \Sigma_{j^{-1}} \left(\hat{\mathbf{x}} - \boldsymbol{\mu}_j\right)\right) + \log\frac{n_j}{n} + (n_j + 1)\left(\frac{d}{2} + 1\right)\log\frac{n_j + 1}{n_j} \tag{25}$$

and constant C is given by

$$C = -\frac{d}{2} - \frac{d}{2}\log 2\pi \tag{26}$$

In a similar manner, change in L_i can be obtained by using Eqs. 15 and 22 as

$$L_i^* = L_i - (\Delta L_i + C) \tag{27}$$

where ΔL_i is given by

$$\Delta L_i = -\frac{1}{2}\log|\Sigma_i| + \frac{n_i - 1}{2}\log\left(1 - \frac{1}{n_i - 1}(\hat{\mathbf{x}} - \boldsymbol{\mu}_i)^{\mathrm{T}} \Sigma_i^{-1} (\hat{\mathbf{x}} - \boldsymbol{\mu}_i)\right) + \log\frac{n_i}{n} - (n_i - 1)\left(\frac{d}{2} + 1\right)\log\frac{n_i - 1}{n_i} \tag{28}$$

and C is same as of Eq. 26.

By adding Eqs. 24 and 27, we can get the change in total log-likelihood (L_{tot}^*) since there is no change in any other clusters apart from χ_i to χ_j; i.e., from Eqs. 17, 24 and 27 we have

$$L_{tot}^* = L_{tot} + \Delta L_{tot} \tag{29}$$

where $\Delta L_{tot} = \Delta L_j - \Delta L_i$. Therefore, the shift of a sample $\hat{\mathbf{x}}$ is advantageous if $\Delta L_{tot} > 0$. This will give the following algorithm (Table 1):

The following sections discuss the characteristic of the SIML method.

Table 1 Stepwise iterative maximum likelihood method procedure

1. *Initialization*: select initial partitions with means $\mu_1, \mu_2, \ldots, \mu_c$ and covariance matrices $\Sigma_1, \Sigma_2, \ldots, \Sigma_c$

2. *Loop*: Select a sample $\hat{\mathbf{x}} \in \chi_i$.

3. If $n_i > 1$ then compute

4. $\delta_j = \begin{cases} \Delta L_j, & j \neq i \\ \Delta L_i, & j = i \end{cases}$

5. Transfer $\hat{\mathbf{x}}$ to χ_k if $\delta_k = \max \delta_j$ for all j.

6. Update L_{tot}, μ_i, μ_k, Σ_i and Σ_k.

7. If L_{tot} doesn't change in n attempts then stop otherwise go to Loop.

Initial settings of the procedure

Similar to any other iterative based optimization technique, this technique also depends on the initial settings. Therefore, it is important to put consideration into the initial settings. In this paper, we implemented three ways of initializing the partitions: 1) random initialization, 2) k-means based initialization, and 3) initialization based on the solution of $c - 1$ partitions and the mean. These schemes are described as follows:

1. Random initialization: In this scheme, we create c random means around the center of the data. This technique works well when the number of clusters is small. If c is very large then it can miss clusters.
2. K-means initialization: In this scheme, the data is first partitioned into c clusters by using the k-means algorithm. The solution of k-means is used as an initial setting for the SIML method. This method works well even if the number of clusters is large. Most of the time this initialization technique provides good results. However, since the k-means algorithm does not track the data based on covariance information, it has limitations.
3. Initialization based on the solution of $c - 1$ clusters: The initialization of c clusters is done by using the solution of $c - 1$ clusters, which would give $c - 1$ locations. The c th location is the mean of the overall data itself. If only two clusters are required to find, then 2 locations around center of the data is used since the solution of 1-cluster is the center of the data itself.

In this paper, we used all the three schemes for initialization and in general schemes 2 and 3 provide satisfactory results for most of the data conformations.

Numerical stability

Due to numerical difficulties the convergences of an iterative algorithm can be missed (e.g. convergence problem for EM algorithm is discussed in [40]). The problem of numeral difficulties is of particular issue when data dimensionality is high. In this situation, iterative algorithms sometimes do not converge properly. This problem usually appears due to the small numerical values of the covariance matrix. If the eigenvalues of a covariance matrix Σ are small, then its determinant can give a value close to zero due to the fixed point architecture of the hardware. However, this problem can be easily overcome by first conducting eigenvalue decomposition of Σ and using the summation of the logarithm of eigenvalues. It is described as follows:

The eigenvalue decomposition of $\Sigma \in \mathbb{R}^{d \times d}$ will give $\mathrm{EDE}^{\mathrm{T}}$ where $\mathrm{E} \in \mathbb{R}^{\mathrm{d} \times \mathrm{d}}$ is the eigenvector matrix and $\mathrm{D} \in \mathbb{R}^{d \times d}$ is the diagonal matrix of eigenvalues. The determinant of Σ will be

$$|\Sigma| = \left|\mathrm{EDE}^{\mathrm{T}}\right| = |\mathrm{D}| = \prod_{k=1}^{d} \lambda_k$$

where λ_k is the k th eigenvalue of Σ. If the values of λ are small then $|\Sigma| = 0$. This problem can be overcome by simply taking logarithm as

$$\log|\Sigma| = \sum_{k=1}^{d} \log\lambda_k$$

In a similar way, the inverse of Σ can cause problems in the term of Eq. 28; i.e., the computation of the term $\log\left(1 - \frac{1}{n_i - 1} P\right)$ (where $P = (\hat{\mathbf{x}} - \boldsymbol{\mu}_i)^{\mathrm{T}} \Sigma_{i^{-1}} (\hat{\mathbf{x}} - \boldsymbol{\mu}_i)$) when the size of the covariance matrix is large. In order to make this numerically stable a small quantity $\epsilon > 0$ can be included as follows:

$$\log\left(1 - \frac{1}{n_i - 1 + \epsilon P}\right)$$

This will ensure that $1 - \frac{P}{n_i + 1 + \epsilon} > 0$.

Small sample size case

When the dimensionality d is much greater than the number of samples n ($d \gg n$) then small sample size problem appears [41–44]. Let a sample set $\chi = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ be drawn independently and let the mean and covariance of χ be denoted by $\boldsymbol{\mu}$ and Σ, respectively. In the normal density we have a term $P = (\mathbf{x} - \boldsymbol{\mu})^{\mathrm{T}} \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu})$ to compute which cannot be solved due to singular covariance matrix as its inverse does not exist. A simple extension could be to use the pseudo-inverse of Σ (denoted here as Σ^{+}). However, this doesn't solve the problem. If samples $\mathbf{x}$ are from χ then $P^{+} = (\mathbf{x} - \boldsymbol{\mu})^{\mathrm{T}} \Sigma^{+} (\mathbf{x} - \boldsymbol{\mu})$ will always be equal to the rank of Σ or basically $n - 1$ (for $d \gg n$). This means that all the samples in a particular cluster would have the same probability and it would not be possible to justify classification of samples based on probability. A second way would be to regularize Σ, however, computing optimal regularization parameter

could be a challenging task. One way could be to apply principal component analysis (PCA) procedure on a d -dimensional sample set $\chi \in \mathbb{R}^d$ to transform it to a parsimonious sample set $Y \in \mathbb{R}^h$ where $h < \min(d, n)$ $h < \min(d, n)$. Thereafter, the clustering procedure can be performed.

Determination of the number of clusters

It is potentially important to estimate the number of clusters c present in the sample set. Since this information is usually not provided, it is important to obtain the value of c with whatever information we have at hand. Basically, the only information we have is the sample itself. In the maximum likelihood procedure we compute likelihood from sample set. Therefore, this information can be utilized to estimate the number of clusters. In order to evaluate c, we can run the SIML algorithm for a range of clusters e.g. $1 \le c \le K$ to see at what point the likelihood function stabilizes or reaches maximum. In this paper, we investigated two ways to compute c. In the first way, we compute the maximum log-likelihood $MaxL_{tot}(c)$ achieved for all values of $c \in [1, K]$. At a particular value of c the $MaxL_{tot}$ reaches maximum and does not change much. This would be the estimated value of c. In the second way, we compute the difference between the maximum log-likelihood $MaxL_{tot}$ achieved and the first value of L_{tot} after SIML procedure (excluding the initial L_{tot} value computed from initial settings as this value is based on the first random guess). Therefore, for a particular number of cluster c, we will get this difference likelihood and we denote it as $DelL_{tot}$ which is equal to $MaxL_{tot} - L_{tot}(1)$ or $\max(L_{tot}) - L_{tot}(1)$, where $L_{tot}(r)$ defines the value of L_{tot} at an iteration r. The curve of $DelL_{tot}$ as a function of c would give a peak at some value of c which would be its best value. In most of the data conformations, $MaxL_{tot}$ gives reliable results. Nonetheless, both the graphs of $MaxL_{tot}$ and $DelL_{tot}$ (as a function of c) are illustrated in the experimental section of the paper.

Results

In order to evaluate the algorithm, we carried out experiments on normal Gaussian data as well as on biological data. We divide this section into 5 subsections. In subsection 1, we illustrate the performance of various methods using three cluster case. Subsection 2 indulges on maximum likelihood plots as a function of number of clusters. In subsection 3, we discuss the processing time of the algorithm. In subsection 4, we discuss the performance in terms of clustering accuracy and rand score of various methods; and, in subsection 5 (parts I and II), we discuss SIML on biological data.

An illustration using three clusters

Since data distribution of GWAS can appear as approximately Gaussian, we generated normal distribution data with 3 different mean and covariance for simulation purposes. Furthermore, if we consider GWAS data from a continent (e.g. Europe) as one cluster, from a country (e.g. Germany) as second cluster and from a city (e.g. Berlin) as a third cluster, then third cluster (Berlin data) will reside inside second cluster (Germany data), and second cluster (Germany data) will reside inside the first cluster (European data). Therefore, clusters will overlap with each other. To simulate this scenario, we generated a sample set with $1,500$ samples, 2 dimensions and 3 clusters as shown in Fig. 2a, and applied various methods on it. Cluster 1 is the least dense (or sparse) and Cluster 3 is the most dense. Cluster 1 has mean $[0.1, 0.1]$ and variance 3 in each direction. Similarly, mean and variance of Cluster 2 and Cluster 3 were $[-1, -1]$ and 0.8, and, $[-0.1, -0.1]$ and 0.05, respectively. The clusters overlap each other and the goal is to track these clusters. It can be seen (from Fig. 2b) that k-means clustered the 3 clusters without considering the distribution information. The processing time to perform the k-means algorithm was 0.82 s. Support vector clustering (CG method) [25] was difficult to perform as it is not possible to provide number of class information. The parameters were tuned so that 3 clusters are outputed. The processing time by this method was 1183.1 s (excluding the tuning time). It can be observed from the Fig. 2c that this method was failed to track the clusters. Next, support vector clustering (using SEP method) [26] was performed. The default parameters gave 45 clusters. Therefore, as similar to the previous CG method, tuning of parameters was carried out to extract only 3 clusters. Processing time was 25.2 s excluding the tuning time. This method also misses the clusters (Fig. 2d). Then we performed the proposed SIML method. This method was able to track all the 3 clusters in 4.49 s per repetition (Fig. 2e). The likelihood plots are discussed in the following section.

Likelihood plots

Here we discussed three plots: log-likelihood (L_{tot}) versus sample (Fig. 3a), maximum log-likelihood ($MaxL_{tot}$) as a function of number of clusters (Fig. 3b) and $DelL_{tot}$ as a function of number of clusters (Fig. 3c).

Figure 3a depicts L_{tot} plot for 3 clusters. When a sample is moved from one cluster to another cluster the value of L_{tot} is updated. This is an increasing function and the maximum value of L_{tot} is defined as $MaxL_{tot}$ in this paper.

Figure 3b depicts $MaxL_{tot}$ plot. Since in general, the number of cluster c information is unknown, it is therefore crucial to estimate this value. In this paper we

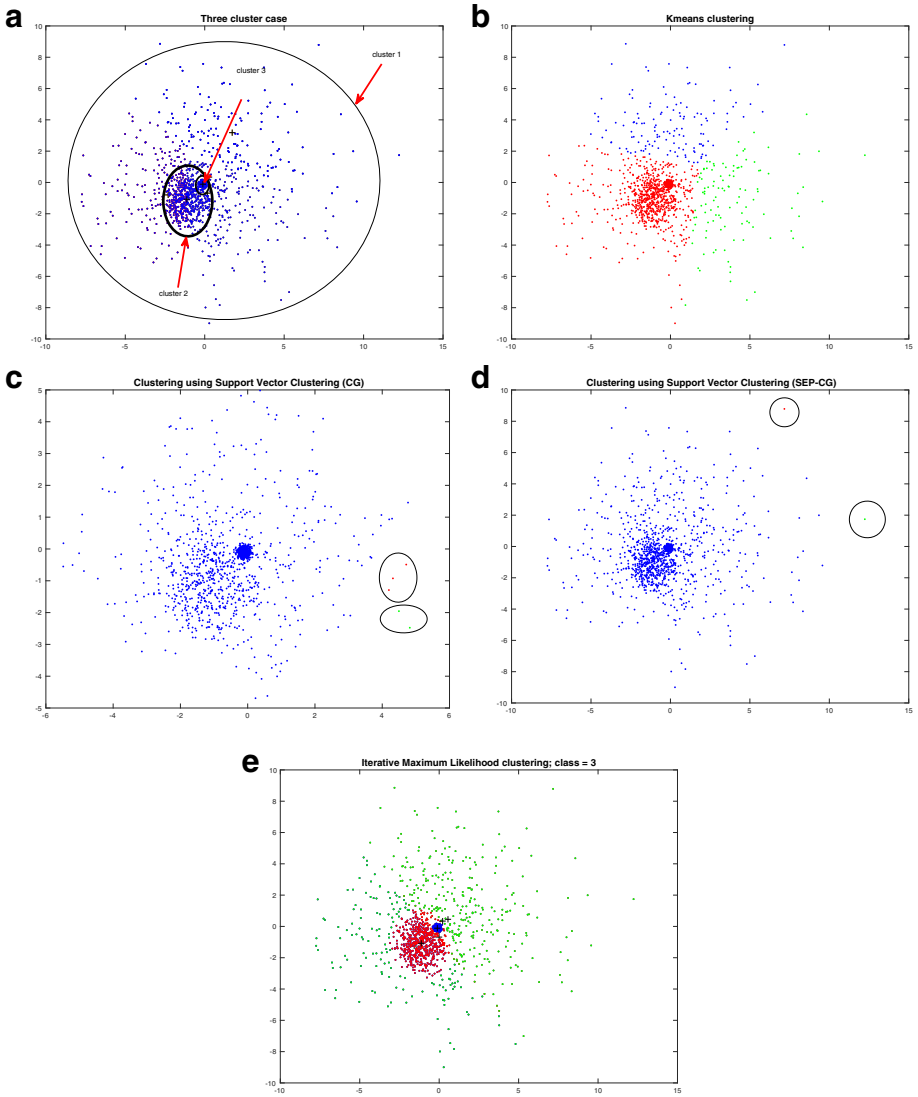

Fig. 2 An illustration using 3 clusters: **a** Three cluster data where $n = 1500$ and $d = 2$; **b** k-means clustering, different colors show different clusters; **c** Support vector clustering (CG method); **d** Support vector clustering (SEP method); **e** Stepwise iterative maximum likelihood (SIML) method

showed that by using $MaxL_{tot}$ plot and $DelL_{tot}$ plot, it is possible to estimate c. For this, one can provide a range of c values and the value for which $MaxL_{tot}$ curve converges (reaches highest peak or does not change much) is the estimated c. We use the same data we generated in Fig. 2a and provide 10 values of c as $1 \le c \le 10$. It can be seen from $MaxL_{tot}$ plot that it converges or peaks at $c = 3$.

Processing time

Here we discuss the processing time of the SIML algorithm. In order to give a complete picture, we investigated the clock time in seconds for samples $n = 3,000$, 9,000, 27,000, 54,000 and 102,000 having 3 clusters. We use the same conformation of data as depicted in Fig. 2a, however, we increased the dimensionality to $d = 10$, 20, 100 and 200. Figure 4 shows the processing time of the algorithm when processed in Linux platform (Ubuntu 14.04 LTS, 64 bit) with 6 processors (Intel Xeon R CPU E5-1660 v2 @ 3.70 GHz) and with 128 GB memory for a repetition.

Clustering on artificial data

We performed clustering accuracy and rand score test on a set of artificial data. For artificial data, we generated d -dimensional, 4 cluster data such that cluster samples are overlapping to each other (in a similar way as shown in Fig. 2). There are in total 2000 samples (where each cluster having 500 samples). We computed cluster accuracy and rand score for various methods. For

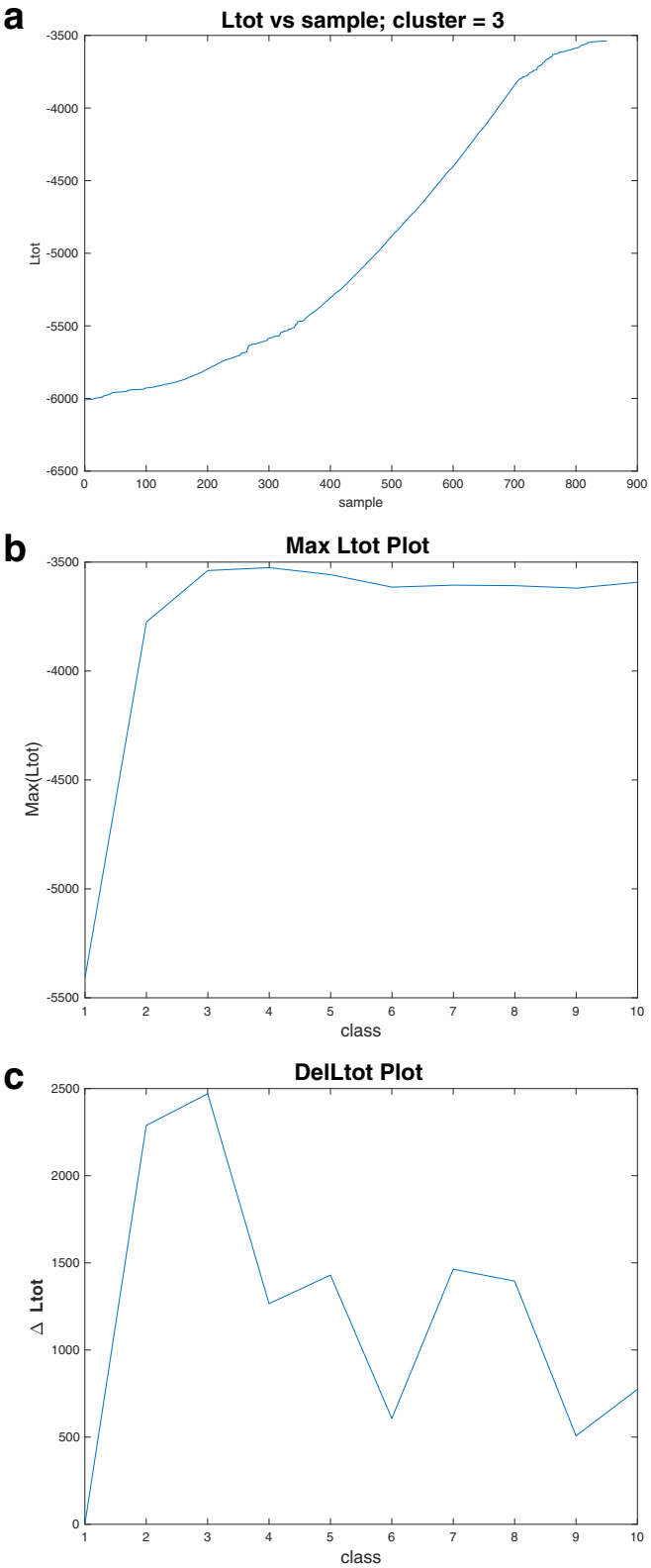

Fig. 3 Likelihood plots **a** L_{tot} plot, **b** $MaxL_{tot}$ plot and **c** $DelL_{tot}$ plot

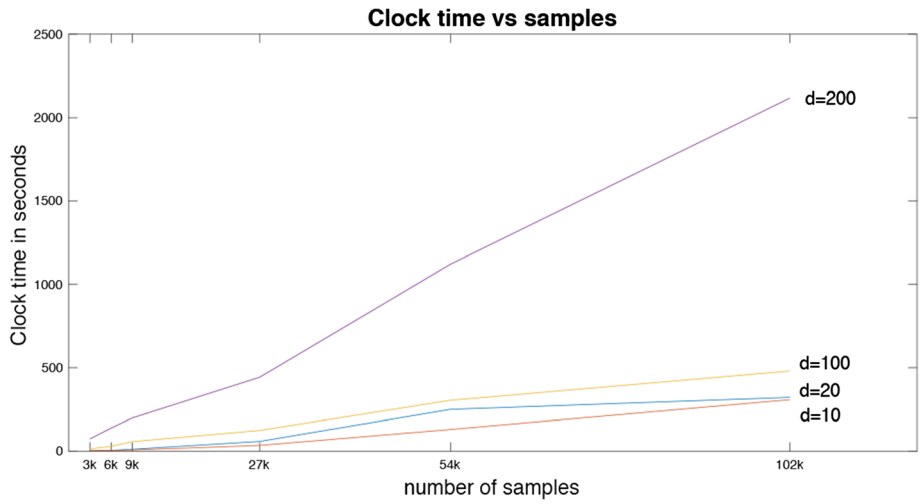

Fig. 4 Processing time of SIML method for $n = 3k - 102k$ and $d = 10 - 200$

statistical stability, we generated data 20 times for a particular dimension d by changing random seed of the normal data. Thereby, we computed average clustering accuracy and average rand score over these 20 attempts for dimension d. We then varied dimension $d = 2, 3, \ldots,$ 20 and reported average clustering accuracy and average rand score in Fig. 5. For comparison, we used centroid-based technique like k-means, hierarchical-based technique like SLink [14], CLink [15] and MLink [16] and model-based technique (using EM algorithm) like mclust [39]. It can be observed from Fig. 5a that mclust and SIML methods perform quite well on Gaussian data. K-means algorithm also performs reasonably well on this data. MLink and SLink couldn't perform well. For average rand score (Fig. 5b), CLink, k-means, SIML and mclust are exhibiting reasonable performance. However, mclust and SIML are superior reaching almost 100 rand score. Since mclust and SIML are derived from Gaussian model, their performance on Gaussian data are well compared to other techniques.

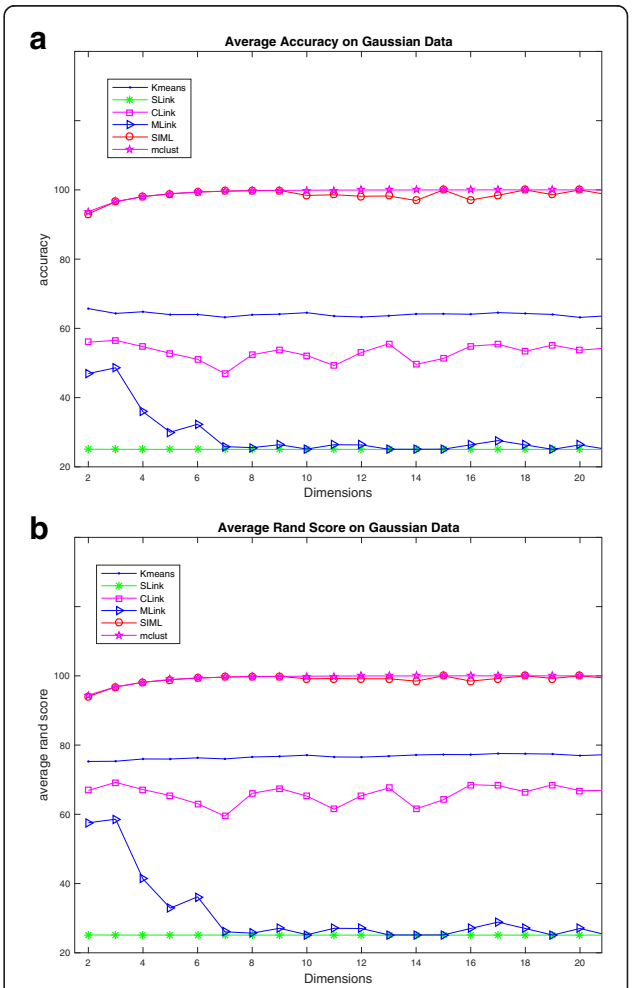

Fig. 5 **a** Average clustering accuracy on Gaussian data. **b** Average rand score on Gaussian data

Clustering on real data-I (publically available biological data)

In this section, we utilized various biological data and reported clustering accuracy and rand score. We employed several methods such as k-means, SLink, CLink, MLink and mclust for comparison. The description of biological data is given as follows:

SRBCT dataset [45]: the small round blue-cell tumor dataset consists of the expression of 2308 genes from 83 samples. This is a four class classification problem. The tumors are Burkitt lymphoma (BL), the Ewing family of tumors (EWS), neuroblastoma (NB) and rhabdomyosarcoma (RMS). The dataset consists of 11, 29, 18 and 25 samples of BL, EWS, NB and RMS respectively.

Table 2 Clustering accuracy on SRBCT dataset

Dim	K-means	SLINK	CLINK	MLINK	mclust	SIML
2	60.4	34.9	62.7	54.2	62.7	**63.9**
3	67.9	39.8	69.9	**71.1**	69.9	66.3
4	77.1	49.4	65.1	67.5	72.3	**81.9**
5	70.3	50.6	**72.3**	50.6	65.1	67.5
6	64.0	39.8	53.0	53.0	57.8	**69.9**

The methods achieving highest results are depicted in bold faces

MLL leukemia [46]: This dataset has 3 classes, namely ALL, MLL and AML leukemia. The dataset contains 24 ALL, 20 MLL and 28 AML. The dimension of MLL dataset is 12,582.

ALL subtype dataset [47]: this dataset consists of the expression of 12,558 genes of subtypes of acute lymphoblastic leukemia. The dataset has seven classes namely BCR-ABL, E2A-PBX1, hyperdiploid >50 chromosomes ALL, MLL, T-ALL, TEL-AML1 and other (contains diagnostic samples that did not fit into any of the former six classes). Samples per class are 15, 27, 64, 20, 43, 79 and 79 respectively.

To vary the data dimensionality (number of features), we utilized Chi-squared feature selection method to rank the attributes. The dimensionality investigated was $d = 2, 3, \ldots, n_m/2$, where n_m is the cluster with minimum number of samples. We then performed cluster analysis (to evaluate clustering accuracy and rand score) on these datasets and compared SIML with the *k*-means, SLink, CLink, MLink and mclust methods. The results are reported in Tables 2 and 3 (for SRBCT dataset), Tables 4 and 5 (for MLL dataset), Tables 6 and 7 (ALL subtype dataset) and Table 8 (for estimation of number of clusters by SIML). Clustering accuracy is depicted in Tables 2, 3, 4, 5, 6 and rand score is shown in Tables 2, 3, 4, 5, 6 and 7. The methods achieving highest results are depicted in bold faces.

It can be observed from Tables 2 and 3 that SIML achieved the highest clustering accuracy and rand score in 3/5 cases, and MLink and CLink achieved the highest performance in 1/5 case each. For the MLL dataset (Tables 4 and 5), mclust achieved the highest clustering accuracy and rand score in 4/9 cases and 3/9 cases, respectively. SIML was able to achieve 4/9 times highest clustering accuracy and rand score. Apart from SIML and mclust, k-means was also able to get reasonable performance especially for higher dimensions. For ALL subtype dataset (Tables 6 and 7), k-means achieved the highest clustering accuracy in 2/7 cases and highest rand score in 3/7 cases. SIML reported the highest clustering accuracy and rand score in 5/7 cases and 4/7 cases, respectively. These results show that SIML can perform reasonably well for many datasets employed in this work. In Table 8, we provided the summary of the number of clusters estimated by SIML. The corresponding $MaxL_{tot}$ plots are given in the Additional file 1. It can be seen from Table 8 that SIML estimates correctly the number of clusters most of the time.

Table 3 Rand score on SRBCT dataset

Dim	K-means	SLINK	CLINK	MLINK	mclust	SIML
2	69.5	32.9	69.9	60.0	62.7	**68.5**
3	77.2	32.0	76.6	**78.4**	69.9	75.3
4	80.5	51.3	71.4	74.8	72.3	**82.7**
5	78.3	53.1	**81.8**	53.1	65.1	75.0
6	72.4	35.8	56.5	56.5	57.8	**78.7**

The methods achieving highest results are depicted in bold faces

Table 4 Clustering accuracy on MLL dataset

Dim	K-means	SLINK	CLINK	MLINK	mclust	SIML
2	56.3	40.3	45.8	45.8	**80.6**	58.3
3	58.8	40.3	50.0	50.0	**68.1**	61.1
4	59.5	43.1	54.2	43.1	**95.8**	72.2
5	81.9	43.1	72.2	69.4	94.4	**95.8**
6	81.9	43.1	81.9	69.4	55.6	**95.8**
7	80.0	41.7	81.9	72.2	91.7	**94.4**
8	81.7	43.1	79.2	68.1	**90.3**	62.5
9	82.8	48.6	80.6	**84.7**	65.3	63.9
10	80.4	43.1	58.3	63.9	61.1	**91.7**

The methods achieving highest results are depicted in bold faces

Clustering on real data-II (SNPs data)

In this section, we attempt to illustrate the use of SIML on real data case. In practical situation, there are two problems to address in a dataset: 1) how many clusters are present; and, 2) what are the locations of the clusters? [48–50]. Sometimes, it is also necessary to identify or remove some sub-population from the data in order to solve the issue of population stratification. The existence of population stratification unmatched between

Table 5 Rand score on MLL dataset

Dim	K-means	SLINK	CLINK	MLINK	mclust	SIML
2	63.6	35.0	41.1	41.1	**80.6**	72.3
3	67.5	35.0	45.7	45.7	68.1	**72.6**
4	64.0	36.3	47.2	36.3	**95.8**	77.5
5	80.4	36.3	75.2	70.2	94.4	**94.7**
6	80.4	36.3	80.4	70.2	55.6	**94.7**
7	79.6	35.3	80.4	75.7	91.7	**93.1**
8	80.6	36.3	78.4	67.7	**90.3**	69.9
9	81.2	41.1	79.3	**82.6**	65.3	71.6
10	80.3	36.3	66.1	73.2	61.1	**90.1**

The methods achieving highest results are depicted in bold faces

Table 6 Clustering accuracy on ALL subtype dataset

Dim	K-means	SLINK	CLINK	MLINK	mclust	SIML
2	**44.8**	32.1	42.8	36.1	34.3	44.0
3	53.3	25.1	45.3	46.2	34.9	**56.9**
4	57.4	25.1	51.7	49.9	33.3	**61.5**
5	60.4	26.0	42.8	34.6	44.7	**62.4**
6	58.9	25.4	38.8	41.0	45.3	**63.6**
7	**58.7**	24.2	47.4	36.1	49.5	56.3
8	54.5	25.7	42.8	34.8	41.9	**61.2**

The methods achieving highest results are depicted in bold faces

cases and controls can produce false positives and negatives in GWAS [51]. For this exercise, we utilize data from a collection of 7001 individuals from the BioBank Japan (BBJ) project and 45 Japanese HapMap (JPT) samples [51]. The total number of single nucleotide polymorphisms (SNPs) was 140,387, genotyped via the Perlegen platform. We also included 45 Han Chinese HapMap (CHB) samples and merged these data using PLINK v1.9 (https://www.cog-genomics.org/plink2) on 140,367 common SNPs. Prior to PCA, we performed filtering using similar criteria as of that used by Yamaguchi-Kabata et al. [51]. We removed SNPs with a call rate < 99 %, a MAF < 0.01, and a Hardy-Weinberg equilibrium (HWE) exact test p-value > 10^{-6}. Individuals with missing calls for > 5 % of SNPs were also removed. After filtering, 6998 BBJ, 44 JPT and 45 CHB samples sharing 117,758 SNPs remained. Consequently, the population consists of 6891 main land Japan (Hondo) samples, 45 CHB samples and 151 Okinawa samples referred as the Ryukyu (RYU) cluster. The Hondo samples can be further subdivided into 628 Kyushu, 908 Kinki, 358 Tokai-Hokoriku, 3975 Kanto-Koshinetsu, 466 Tohoku, 512 Hokkaido and 44 JPT samples. The aim here is to classify RYU and CHB from Hondo so that Hondo only data can be explored for further analysis. We first performed PCA on the filtered data using the R package SNPRelate [52] to reduce the data dimensionality and conducted analysis on 2 dimensional data. Linkage disequilibrium (LD) pruning with a threshold of 0.2 was used to define a representative set of 32,090 SNPs for PCA.

Table 7 Rand score on ALL subtype dataset

Dim	K-means	SLINK	CLINK	MLINK	mclust	SIML
2	**73.1**	37.1	68.2	49.9	34.3	71.8
3	**79.2**	20.5	73.5	62.7	34.9	77.6
4	**81.6**	20.4	78.3	72.4	33.3	81.2
5	79.6	22.0	69.0	47.3	44.7	**82.8**
6	79.9	21.6	69.5	67.5	45.3	**83.1**
7	79.9	21.0	75.2	40.6	49.5	**80.2**
8	77.8	21.7	70.3	60.6	74.9	**82.2**

The methods achieving highest results are depicted in bold faces

Table 8 The estimation of the number of clusters by SIML

Dim	SRBCT	MLL	ALL subtype
2	4	3	7
3	4	2	7
4	4	2	8
5	4	3	4,7
6	2,4	3	7,9
7		3	3,8
8		3	7
9		3	
10		6	

In summary, this two dimensional data contain three clusters: Hondo, RYU and CHB. Here we first computed true positives (and its corresponding accuracy) for Hondo, RYU and CHB clusters. This would provide us information regarding correctly labelled samples in each cluster. For this purpose, we executed all the methods to provide 3 clusters of the data. The true positives for various methods are depicted in Table 9.

From Table 9, we can see that k-means was able to cluster all CHB samples correctly and also attained high true positive for the RYU cluster. However, it displayed comparatively inferior performance for the Hondo cluster. SLink reported very high true positive for Hondo and CHB clusters. However, it completely missed the RYU cluster. CLink, MLink and SIML were able to label all 45 samples of CHB correctly. SIML achieved the highest true positive for RYU among these 3 methods

Table 9 True positives for Hondo, RYU and CHB cluster on BBJ and HapMap data

Methods	Hondo (6891)	RYU (151)	CHB (45)
	71.4 %	85.5 %	100 %
K-means	4922	129	45
	99.9 %	0 %	100 %
SLINK	6886	0	45
	97.9 %	92.7 %	100 %
CLINK	6746	140	45
	95.8 %	92.1 %	100 %
MLINK	6603	139	45
	97.3 %	94.7 %	100 %
SIML	6707	143	45
	66.8 %	94.7 %	0 %
mclust	4602	143	0

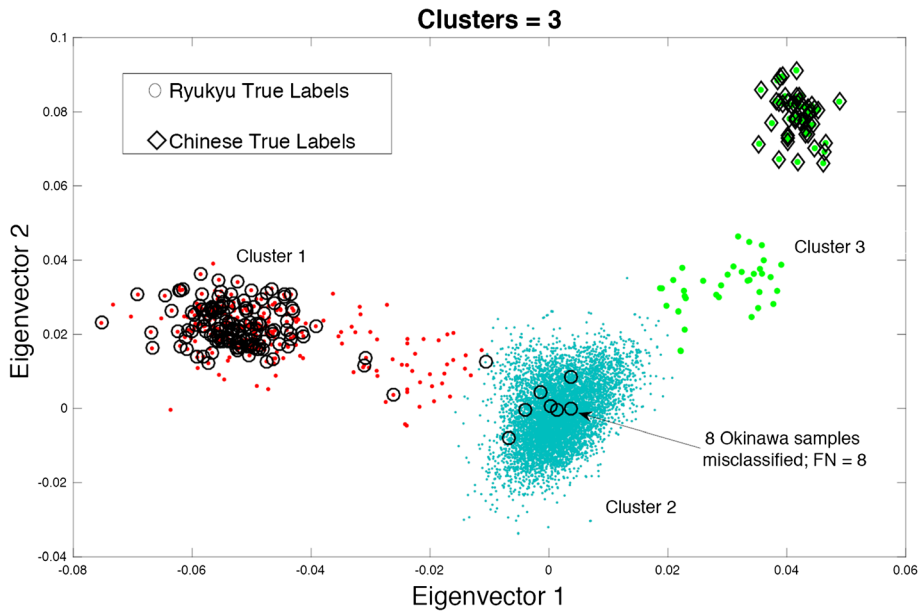

Fig. 6 Clustering by SIML on 2-dimensional BBJ data

and CLink was slightly better (97.9 %) than SIML (97.3 %) for the Hondo cluster. In this case, mclust did not perform well. Nonetheless, mclust gave a high true positive for RYU cluster. It should be noted here that this data is highly imbalanced. Out of 7087 samples, 6891 samples belong to the Hondo cluster (i.e., almost 97 %) leaving only 3 % of samples for the RYU and CHB clusters. This imbalance creates a problem in a way that majority of samples turn to be labelled under the larger cluster leaving the smaller clusters. Nonetheless, SIML has shown encouraging results.

In the next analysis, we did not provide the number of clusters information to study the characteristics of SIML method. The resulting clustering is illustrated in Fig. 6. For this case, the $MaxL_{tot}$ plot gives peak at 3 clusters (Fig. 7) and therefore 3 clusters were used in this case. The true RYU and CHB labels are shown on the plot as circles and diamonds, respectively. Most of Hondo samples are in Cluster 2. There are around 6715 samples in Cluster 2 representing the Hondo region. Almost all CHB are clustered in Cluster 3 and most of RYU are clustered in Cluster 1. Around 8 RYU are clustered in Cluster 2 giving a false negative (FN) error of 8 samples (5.3 %) and no CHB sample is misclassified giving FN error of 0 samples (0 %). Cluster 1 and Cluster 3 can be classified easily and analysis can be conducted on Cluster 2 (Hondo) with very less FN error.

In summary, SIML successfully estimates the number of clusters as well as the locations. The SIML package was tested on Ubuntu 14.04 LTS OS (with 128 GB memory and Intel Xeon R CPU E5-1660 v2 @ 3.7 GHz x 6). The OS type is 64-bit. For Matlab we used 'Statistics and Machine Learning Toolbox'.

Conclusions

In this work, through considering conformations of many biological data, we developed a clustering algorithm based on maximum likelihood estimate. The proposed stepwise iterative maximum likelihood (SIML) method is different from other maximum likelihood methods as it does not require the computation of first and second derivative of likelihood functions. This avoids the necessity to have differentiable likelihood functions for convergence. The SIML method was tested on artificial and real data to evaluate its performance. We show that SIML was able to produce promising results over state-of-the-art methods. The SIML method

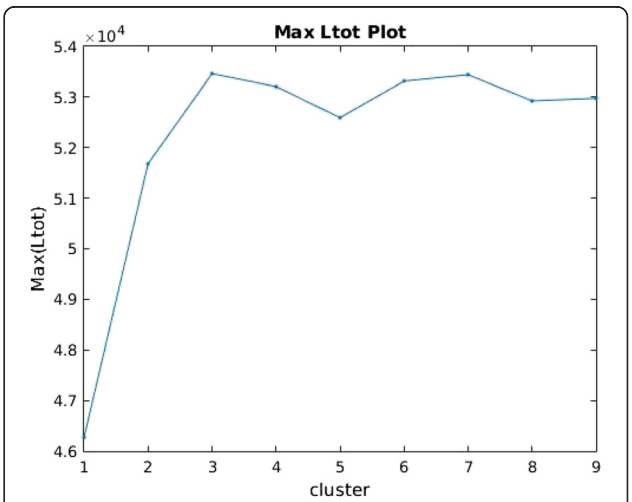

Fig. 7 $MaxL_{tot}$ Plot for 2-dimensional BBJ and HapMap data

was also able to estimate the number of clusters successfully. The Matlab package of SIML is available from our webpage.

Additional file

Additional file 1: Estimation of number of clusters using SIML method. (DOCX 408 kb)

Acknowledgements
This work has been supported by the CREST, JST, Japan.

Funding
The project was funded by JST Grant, Japan.

Availability of data and materials
All the 3 microarray datasets (SRBCT, MLL and ALL subtype) are publically available can can be downloaded via author's webpage or visiting Kent Ridge Bio-medical Repository. The SNP data is managed by RIKEN management only and is not publically available. It is not in our (authors') jurisdiction to make it available. The Matlab package of SIML is available via visiting authors' webpage.

Authors' contributions
AS developed the concept, carried out experiments and written the first draft of the manuscript. DS arranged and assisted in processing the genomic data. KAB: also processed the data and assisted in manuscript write-up. YL performed some experimental tasks. YK and MK provided the data. TT financed and supervised the project. All authors read and approved the final manuscript.

Competing interests
None of the authors have any competing interests.

Consent for publication
Not applicable.

Ethics approval and consent to participate
The Biobank Japan Project collected human genomic DNA after the patients provided written informed consent to participate in this project. This project was approved by the ethical committees at The Institute of Medical Science, The University of Tokyo, and the RIKEN Center for Integrative Medical Sciences (Ref. No. RIKEN Yokohama H17–16).

Author details
[1]RIKEN Center for Integrative Medical Sciences, Yokohama 230-0045, Japan. [2]CREST, JST, Yokohama 230-0045, Japan. [3]Institute for Integrated and Intelligent Systems, Griffith University, Brisbane, Australia. [4]Medical Research Institute, Tokyo Medical and Dental University, Tokyo 113-8510, Japan.

Received: 27 November 2015 Accepted: 12 August 2016
Published online: 24 August 2016

References
1. Mo Q, Wang S, Seshan VE, Olshen AB, Schultz N, Sander C, et al. Pattern discovery and cancer gene identification in integrated cancer genomic data. Proc Natl Acad Sci U S A. 2013;110(11):4245–50.
2. Monti S, Tamayo P, Mesirov J, Golub T. Consensus Clustering: A Resampling-Based Method for Class Discovery and Visualization of Gene Expression Microarray Data. Mach Learn. 2003;52:91–118.
3. Wilkerson MD, Hayes DN. ConsensusClusterPlus: a class discovery tool with confidence assessments and item tracking. Bioinformatics. 2010;26(12):1572–3.
4. Jain AK. Data clustering: 50 years beyond K-means. Pattern Recogn Lett. 2010;31(8):651–66.
5. Duda RO, Hart PE, Stork DG. Pattern Classification. 2nd ed: Wiley-Interscience; 2000.
6. Maimon O, Rokach L. Data Mining and Knowledge Discovery Handbook. 2nd ed. USA: Springer-Verlag New York Incorporated; 2010.
7. Fisher D. Iterative optimization and simplification of hierarchical clusterings. J Intell Res. 1996;4(1):147–79.
8. Dhillon IS, Guan Y, Kogan J, editors. Iterative clustering of high dimensional text data augmented by local search. Maebashi City, Japan: The 2002 IEEE International Conference on Data Mining; 2002.
9. Fayyad UM, Reina CA, Bradley PS, editors. Initialization of Iterative Refinement Clustering Algorithms. Proceedings of the 4th International Conference on Knowledge Discovery & Data Mining (KDD98). Menlo Park: AAAI Press; 1998.
10. Hastie T, Tibshirani R, Friedman J. The Elements of Statistical Learning. 2nd ed. New York: Springer; 2009.
11. Heller KA, Ghahramani Z. Bayesian hierarchical clustering. Bonn, Germany: Twenty-second International Conference on Machine Learning (ICML); 2005.
12. Farrell S, Ludwig C. Bayesian and maximum likelihood estimation of hierarchical response time models. Psychon Bull Rev. 2008;15(6):1209–17.
13. Sharma A, Boroevich K, Shigemizu D, Kamatani Y, Kubo M, Tsunoda T. Hierarchical Maximum Likelihood Clustering Approach. USA: IEEE Transactions on Biomedical Engineering. 2016;PP (99). doi:10.1109/TBME.2016.2542212.
14. Sibson R. SLINK: An optimally efficient algorithm for the single-link cluster method. Comput J (Br Comput Soc). 1973;16(1):30–4.
15. Defays D. An efficient algorithm for a complete link method. Comput J (Br Comput Soc). 1977;20(4):364–6.
16. Everitt BS, Landau S, Leese M, Stahl D. Cluster Analysis. 5th ed. UK: John Wiley & Sons; 2011.
17. Lock EF, Dunson DB. Bayesian consensus clustering. Bioinformatics. 2013;29(20):2610–6.
18. Liu JS, Zhang JL, Palumbo MJ, Lawrence CE. Bayesian Clustering with Variable and Transformation Selections. Bayesian Stat. 2003;7:249–75.
19. Latch EK, Dharmarajan G, Glaubitz JC, Jr OER. Relative performance of Bayesian clustering software for inferring population substructure and individual assignment at low levels of population differentiation. Conserv Genet. 2006;7(2):295–302.
20. Chen C, Durand E, Forbes F, François O. Bayesian clustering algorithms ascertaining spatial population structure: a new computer program and a comparison study. Mol Ecol Notes. 2007;7(5):747–56.
21. Ramoni M, Sebastiani P, Cohen P. Bayesian Clustering by Dynamics. Mach Learn. 2002;47(1):91–121.
22. Dempster AP, Laird NM, Rubin DB. Maximum Likelihood from Incomplete Data via the EM Algorithm. J R Stat Soc Ser B. 1977;39(1):1–38.
23. Misztal I. Comparison of computing properties of derivative and derivative-free algorithms in variance-component estimation by REML. J Anim Breed Genet. 1994;111(1–6):346–55.
24. Denoeux T. Maximum Likelihood Estimation from Uncertain Data in the Belief Function Framework. IEEE Trans Knowl Data Eng. 2013;25(1):119–30.
25. Ben-Hur A, Horn D, Siegelmann HT, Vapnik V. Support Vector Clustering. J Mach Learn Res. 2001;2:125–37.
26. Lee J, Lee D. An improved cluster labeling method for support vector clustering. IEEE Trans Pattern Anal Mach Intell. 2005;27(3):461–4.
27. Lee J, Lee D. Dynamic Characterization of Cluster Structures for Robust and Inductive Support Vector Clustering. IEEE Trans Pattern Anal Mach Intell. 2006;28(11):1869–74.
28. Chiang J-H, Hao P-Y. A new kernel-based fuzzy clustering approach: support vector clustering with cell growing. IEEE Trans Fuzzy Syst. 2003;11(4):518–27.
29. Horng S-J, Su M-Y, Chen Y-H, Kao T-W, Chen R-J, Lai J-L, et al. A novel intrusion detection system based on hierarchical clustering and support vector machines. Expert Systs Appl. 2011;38(1):306–13.
30. Jun S, Park S-S, Jang D-S. Document clustering method using dimension reduction and support vector clustering to overcome sparseness. Expert Systs Appl. 2014;41(7):3204–12.
31. Wang K, Liang C, Liu J, Xiao H, Huang S, Xu J, et al. Prediction of piRNAs using transposon interaction and a support vector machine. BMC Bioinf. 2014;15(1):419.
32. Long JS. Regression Models for Categorical and Limited Dependent Variables. London: Sage Publications; 1997.
33. Felsenstein J, Churchill GA. A Hidden Markov Model Approach to Variation Among Sites in Rate of Evolution. Mol Biol Evol. 1996;13(1):93–104.
34. Jennrich RI, Sampson PF. Newton–Raphson and Related Algorithms for Maximum Likelihood Variance Component Estimation. Technometrics. 1976;18(1):11–7.
35. Adachi J, Hasegawa M. MOLPHY version 2.3: programs for molecular phylogenetics based on maximum likelihood. Comput Sci Monogr. 1996;28:1-150.

36. Berndt ER, Hall BH, Hall RE, Hausman JA. Estimation and Inference in Nonlinear Structural Models. Ann Econ Soc Meas. 1974;3(4):653–65.
37. Davidon WC. Variable Metric Method for Minimization. SIAM J Optim. 1991;1(1):1–17.
38. Fletcher R, Powell MJD. A Rapidly Convergent Descent Method for Minimization. Comput J. 1963;6(2):163–8.
39. Fraley C, Raftery AE. MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering. Seattle, WA, USA: University of Washington; 2006.
40. Cd A, Lee JA, Verleysen M. On Convergence Problems of the EM Algorithm for Finite Gaussian Mixtures. Bruges: European Symposium on Artificial Neural Networks (ESANN); 2003. p. 99–106.
41. Sharma A, Paliwal KK. Fast principal component analysis using fixed-point algorithm. Pattern Recogn Lett. 2007;28(10):1151–5.
42. Sharma A, Paliwal KK. A Gradient Linear Discriminant Analysis for Small Sample Sized Problem. Neural Process Lett. 2008;27(1):17–24.
43. Sharma A, Paliwal KK. Cancer classification by gradient LDA technique using microarray gene expression data. Data Knowl Eng. 2008;66(2):338–47.
44. Sharma A, Imoto S, Miyano S. A Top-r Feature Selection Algorithm for Microarray Gene Expression Data. IEEE/ACM Trans Comput Biol Bioinform. 2012;9(3):754–64.
45. Khan J, Wei JS, Ringnér M, Saal LH, Ladanyi M, Westermann F, et al. Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. Nat Med. 2001;7(6):673–9.
46. Armstrong SA, Staunton JE, Silverman LB, Pieters R, Boer ML, Minden MD, et al. MLL translocations specify a distinct gene expression profile that distinguishes a unique leukemia. Nat Genet. 2002;30(1):41–7.
47. Yeoh E-J, Ross ME, Shurtleff SA, Williams WK, Patel D, Mahfouz R, et al. Classification, subtype discovery, and prediction of outcome in pediatric acute lymphoblastic leukemia by gene expression profiling. Cancer Cell. 2002;1(2):133–43.
48. Rahman MM, Davis DN. Fuzzy Unordered Rules Induction Algorithm Used as Missing Value Imputation Methods for K-Mean Clustering on Real Cardiovascular Data. Lect Notes Eng Comput Sci. 2012;2197(1):391–4.
49. Mirkin B. Clustering for Data Mining: A Data Recovery Approach. Boca Raton: Chapman & Hall; 2005.
50. Elhamifar E, Vidal R. Sparse Subspace Clustering: Algorithm, Theory, and Applications. IEEE Trans Pattern Anal Mach Intell. 2013;35(11):2765–81.
51. Yamaguchi-Kabata Y, Nakazono K, Takahashi A, Saito S, Hosono N, Kubo M, et al. Japanese Population Structure, Based on SNP Genotypes from 7003 Individuals Compared to Other Ethnic Groups: Effects on Population-Based Association Studies. Am J Hum Genet. 2008;83(4):445–56.
52. Zheng X, Levine D, Shen J, Gogarten SM, Laurie C, Weir BS. A high-performance computing toolset for relatedness and principal component analysis of SNP data. Bioinformatics. 2012;28(24):3326–8.