OXFORD

*Briefings in Bioinformatics*, 17(5), 2016, 758–770

**doi: 10.1093/bib/bbv074**
Advance Access Publication Date: 22 September 2015
Paper

# Discretization of gene expression data revised

Cristian A. Gallo, Rocio L. Cecchini, Jessica A. Carballido, Sandra Micheletto and Ignacio Ponzoni

Corresponding author: Ignacio Ponzoni, Laboratorio de Investigación y Desarrollo en Computación Científica (LIDeCC), Dpto. de Cs. e Ing. de la Computación, UNS, 1253 Av. Alem, Bahía Blanca, Argentina 8000. Tel.: +542914595135; Fax: +542914595136; E-mail: ip@cs.uns.edu.ar



## Abstract

Gene expression measurements represent the most important source of biological data used to unveil the interaction and functionality of genes. In this regard, several data mining and machine learning algorithms have been proposed that require, in a number of cases, some kind of data discretization to perform the inference. Selection of an appropriate discretization process has a major impact on the design and outcome of the inference algorithms, as there are a number of relevant issues that need to be considered. This study presents a revision of the current state-of-the-art discretization techniques, together with the key subjects that need to be considered when designing or selecting a discretization approach for gene expression data.

**Key words**: discretization; data preprocessing; gene expression data; gene expression analysis; data mining; machine learning

## Introduction

Recent developments in mRNA quantification techniques enable the simultaneous measurement of the expression level of a large number of genes for a given experimental condition. Both microarray and RNA-seq technologies are providing an unprecedented amount of meaningful biological data. In this regard, numerous machine learning methods have been extensively used in the analysis of gene expression data (GED) obtained from these experiments [1–7]. The input data are required to be discrete in several cases, as with any modeling algorithm using discrete-state models [3, 7, 8].

Data discretization is a technique used in computer science and statistics, frequently applied as a preprocessing step in the analysis of biological data. In general, the aim of GED discretization is to allow the application of algorithms for the inference of biological knowledge that requires discrete data as an input, by mapping the real data into a typically small number of finite values. The biological problems that can be addressed by discretizing the GED are roughly the same as those addressed in the continuous domain. The main difference lies in the final modeling of the extracted knowledge, in which the discrete states favor the inference of qualitative models, whereas the continuous values allow the inference of quantitative models [3]. However, knowledge inference from discrete data has several advantages when data-driven analysis is performed. The learning process from discrete data is more efficient and effective [9–11], requiring a reduced amount of data as compared with other methods that use continuous values [3]. Also, the reduction and simplification of the data make the learning process faster, hence yielding more compact and shorter results [12],

**Cristian A. Gallo** has a PhD in Computer Science and has a postdoctoral position at Laboratory of Research and Development in Scientific Computing. Her research interests is in the area of machine learning applied to bioinformatics optimization problems.
**Rocío L. Cecchini** has a PhD in Computer Science. She is a researcher at Laboratory of Research and Development in Scientific Computing. Her Research interests focus on data-mining and text-mining methods with application to bioinformatic problems.
**Jessica A. Carballido** has a PhD in Computer Science and is a Professor and a Researcher at the Universidad Nacional del Sur. His research group focuses on machine learning techniques applied to bioinformatic optimization problems.
**Sandra Micheletto** has a PhD in Agronomy with a minor in Bioinformatics. Her research lies in microarrays data analysis and computational methods to study plant gene expression under drought stress conditions.
**Ignacio Ponzoni** has a PhD in Computer Science. He is the current president of the Argentinean Association on Computational Biology and Bioinformatics (A2B2C) and he is also a member of the ISCB. His research interests focus on evolutionary computing and machine learning methods applied to systems biology and molecular informatics.
**Submitted:** 22 May 2015; **Received (in revised form):** 26 July 2015

and allowing the inference of large-size models with a higher speed of analysis [3]. For researchers, discrete values are easier to understand, use and explain [3, 12]. Another advantage is the homogenization of different data sets in terms of interpretability; if the same semantics is used for discretization into states of heterogeneous data sets, it is easier to contrast the discretized values, which can be analyzed beyond the discretization thresholds used for each data set [13].

There are other specific advantages related to data discretization itself, as well as the benefits of performing the inference process in the discrete domain. By using discrete states, a significant portion of the biological and technical noise presented in the raw data is absorbed [7]. For time series data, Dimitrova *et al.* [14] demonstrate how discretization algorithms perform in the presence of typical levels of noise in the experimental data. They found that the discretization of the data showed more robustness in the presence of noise the higher the variance of a time series when compared with the continuous values. Moreover, discretization of GED may also lead to better prediction accuracy [15]. Ding and Peng [15] carried out experiments using five data sets of gene expression profiles, including two of leukemia data [16] and colon cancer data [17] and they showed that the best continuous features lead to more errors when compared with the best discretized features. They also demonstrate that discretization of the gene expressions leads to better classification accuracy than the original continuous data [15].

Nevertheless, the choice of an appropriate discretization method is not a trivial task. In general, any discretization process implies loss of information [12]. Discretization represents the transition from the continuous to the discrete data world and plays a crucial step in the construction of discrete models. Thus, if the transition is not done well, all subsequent steps are defective [14]. The qualitative nature of the discrete data entails that different discretization strategies may yield to distinct discrete-state models. Therefore, the biological semantic and interpretation of the resulting models might differ, even when the subjacent real-valued data are the same [8]. Consequently, the selected discretization procedure determines the success of the posterior inference task in accuracy and/or simplicity of the model [12]. By this means, the discretization approach for GED should consider the intrinsic nature of the biological data in addition to the technology involved in the measurements, to provide the most accurate representation of the data with a reduced loss of information. It is also important to consider the particular features of the computational method that will be applied to the discretized data, as it will determine the scheme used by the discretization approach.

Over the past years, several approaches have been proposed in the literature to deal with the discretization of GED. In this regard, there are already a few studies that revised some of the discretization approaches for GED in the literature [8, 18, 19]. Madeira and Oliveira [8] were the first to revise this subject, reviewing and assessing simple unsupervised techniques, and proposing a classification for the methods regarding the sample type of the GED. Li *et al.* [18] also revised simple unsupervised approaches and they include more complex clustering techniques, proposing a method that performs better than the reviewed ones. Finally, Mahanta *et al.* [19] reviewed and assessed some of the discretization approaches revised by Madeira and Oliveira [8] and Li *et al.* [18], proposing an extended classification of those unsupervised methods.

A complete understanding of the semantics of discretization approaches for GED is always required to choose the method that best suits a particular case of interest. In this review, a further revision of the classical and state-of-the-art approaches for discretization of GED is proposed, reviewing the most recent research in the field and including supervised discretization for GED that has not been addressed previously. Additionally, an in-depth analysis of the main features of the discretization of GED is also presented, providing a valuable tool for guidance in the selection of an appropriate discretization approach. Finally, a software package that implements most of the GED discretization methods is also provided.

In the following section, the problem definition will be introduced followed by the key issues to be considered when dealing with the discretization of GED. Next, the state-of-the-art approaches used for GED will be summarized. Finally, a discussion is elaborated.

## Discretization problem in gene expression analysis

The discretization process transforms quantitative data into qualitative data, i.e. mRNA concentrations into a finite number of intervals, obtaining a nonoverlapping partition of the continuous domain as a result. An association between each interval with a discrete value is then established. In practice, the discretization can be viewed as a data reduction technique because it maps data from a vast spectrum of numeric gene expression values to a greatly reduced subset of discrete state values. Once the discretization is performed, the data can be used in any inference process that requires a discrete representation. Many existing inference algorithms for GED are designed only to learn from this kind of discrete state data, whereas real-world measurements of gene expression involve continuous values. So, these numerical features have to be discretized before using such algorithms.

In this regard, let us begin with a stricter formulation of the problem. Let A' be a GED matrix of $N$ rows and $M$ columns, where $a'_{ij}$ represents the expression level of the gene $g_i$ under the condition $j$. The matrix A' is defined by its set of rows, $I$, and its set of columns, $J$. A discretized matrix A results from the application of a discretization function, $f_D$, on A', that maps each element $a'_{ij}$ in A' to one of the elements of an alphabet $\sum$. Here the alphabet $\sum$ consists of a set of $k$ symbols that may represent a distinct gene activation level, with $1 < k \leq M$. Now assume, without loss of generality, that the discretization algorithm considers the values of each gene $g_i$ of the GED A' independently. In this way, the discretization algorithm ($f_D$(A', $g_i$)) transforms the continuous gene expression values of $g_i$ in the data set A', into $k$ intervals $D = \{[p_0, p_1], (p_1, p_2], \ldots, (p_{k-1}, p_k]\}$, where $p_0$ and $p_k$ are the minimal and maximal values in the gene expression profile of $g_i$, respectively, and $p_r < p_{r+1}$, for $r = 0, 1,., k-1$. The set of intervals $D$ is called a discretization scheme for the gene $g_i$, and $P = \{p_1, p_2, \ldots, p_{k-1}\}$ is the set of cut points for $g_i$ in A'. Then, all the values on each interval $p_r - p_{r+1}$ for $g_i$ are mapped to the $r$-$th$ symbol of $\sum$, thus transforming the matrix A' into a matrix A, where $a_{iq}$ represents the discretized value of the expression level of $g_i$ under the $q$ experimental conditions, $q \in J$, that satisfies $p_r < a'_{iq} \leq p_{r+1}$. Figure 1 shows an example of the workflow involved in the discretization process when dealing with two discrete gene states.

Obtaining the optimal discretization is an NP-complete problem [10, 12]. Thereby, several discretization techniques have been developed for expression data analysis. According to previous studies [8, 18, 19], there are some well-established

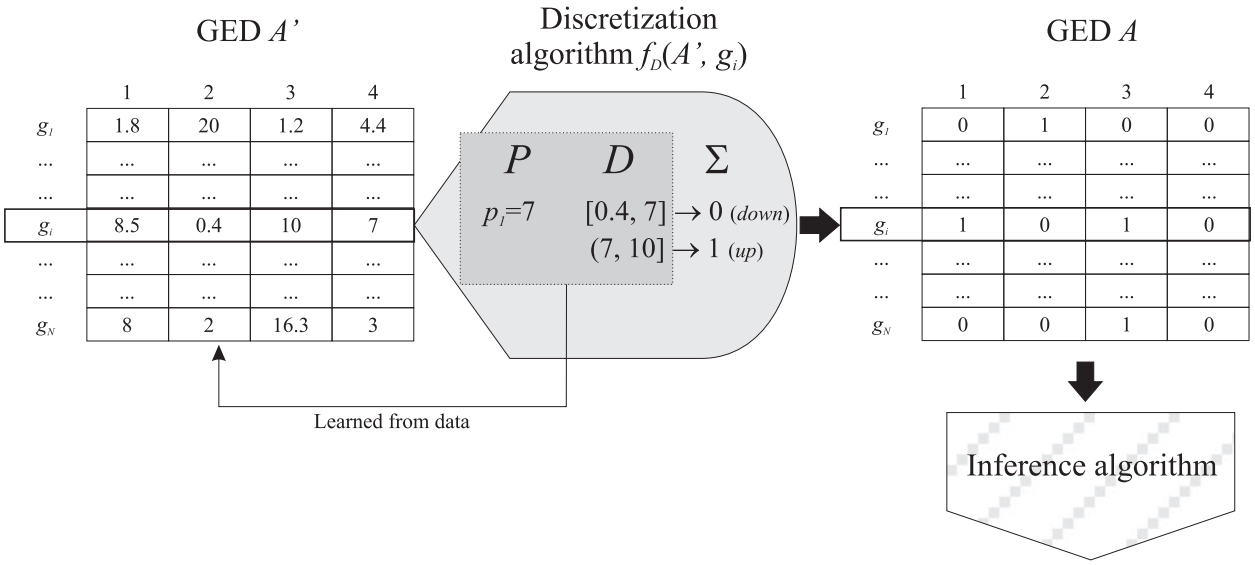


**Figure 1.** Workflow of the discretization process with two discrete states ($\Sigma = \{0, 1\}$) for the gene $g_i$. In the example, the GED A' and the discretized GED A are composed by N genes and four experimental conditions. The discretization algorithm $f_d$ takes A' and $g_i$ and infers the cut point $P = \{7\}$ and the discretization scheme $D = \{[0.4, 7], (7, 10]\}$ to obtain the discretized expression profile of $g_i$ in the GED A. Then, the discretized GED A is the input of an inference algorithm that will extract some kind of information of interest.

features that characterize these approaches. In the next section, a further in-depth analysis of those features will be performed, together with the presentation of other remarkable issues associated with new approaches that are currently emerging.

## Main features of gene expression discretization

In this section, the main features of the gene expression discretization will be presented and analyzed as follows:

### Supervision

The two high-level features of the discretization of GED, namely 'supervised' and 'unsupervised', define whether the particular approach relies on class label information to perform the discretization. So, in the 'supervised' discretization, the values of $g_i$ are assigned regarding the class label information of a specific knowledge domain. The manner in which the discretization method considers the class information depends on the relation between the data and the class labels, and on the heuristic measure used to determine the best cut points. However, most discretization approaches proposed in the literature for GED are 'unsupervised' and follow the problem definition explained in the previous section.

### Level of discretization

Another characteristic to consider about the discretization of GED is the 'level of discretization'. In the simplest case, the alphabet $\sum$ may contain only two symbols (see Figure 1), where one symbol is used for 'upregulation' (or 'activation') and the other symbol is used for 'downregulation' (or 'inhibition'). Thereby, the expression matrix is usually transformed into a binary matrix, where 1 means 'upregulation' and 0 means 'downregulation'. Another widely used scheme is to consider a ternary set of discretization symbols, {−1, 1, 0}, meaning 'downregulated', 'upregulated' or 'no-change'. Nevertheless, the values in matrix A' may be discretized in a multilevel manner to an arbitrary number of symbols. The 'level of discretization' mostly depends on the inference algorithm that will rely on the discretized data. However, the trade-off between the loss of information and the computational complexity may also play an important role in the determination of the 'level of discretization' because, as $k$ increases in value, the loss of information decreases at the cost of increasing the computational complexity of the inference algorithm [3].

### Data technology type

This feature may also influence the selection of a discretization approach. In general, almost all discretization methods were developed for microarray GED [8, 18, 19], without taking into consideration the particular characteristic of the data that is being discretized. This allows the application of the methods for both microarray and RNA-seq GED. However, RNA-seq technology offers many advantages over conventional microarrays, such as a low background signal and an increased dynamic range of measurements [20, 21]. Thus, the consideration of the particular characteristics of the technology involved in the extraction of the biological data may lead to the development of more reliable discretization approaches [22], although it may compromise the application to other platforms.

### Sample type

The 'sample type' is related to the previous feature, i.e. the type of experiment that determines the meaning of the columns in the data matrix. There are basically two types of samples, the equilibrium (steady state) expression levels that correspond to a static situation, and the time series expression levels that are gathered during a phenotypic phase, such as in the cell cycle [23]. Generally, in the steady state expression data, different experimental conditions refer to different tissues, temperatures, chemical compounds or any other condition that may produce different regulatory behavior among the sampled genes. Each element $a'_{ij}$ of A' contains the expression value of gene $g_i$ in the sample or experimental condition $j$. On the other hand, the time

series data are also represented by means of a GED matrix A', except that the rows and columns represent genes and time points, respectively. That is, the different columns represent the expression values of each sampled gene at different times under the same experimental condition, during some phenotypic phase. The sampling intervals at which the genes are sampled are determined by the researcher in regard to the nature of the study, and they are not necessarily defined at the same equidistant interval. Taking the sample type into consideration may lead to the selection of a specific discretization method, e.g. in the case of time series GED, it is not unusual to compute the discretization using expression variations between time points, but this approach is clearly not applicable to steady state GED. On the other hand, in general, any discretization approach for steady state data can be applied to time series data, without taking account of the time correlation between the samples.

### Data scope

Another common issue in the discretization of GED is related to the 'data scope' used to compute the discretized values. In other words, the discrete value of $a'_{ij}$ may be determined regarding the gene profile, the experimental condition profile or the whole matrix. Also, as stated previously, in the case of time series data, it is possible to consider the expression variations between successive time points, thus leading, in general, to a highly reduced scope for the computation of the discretized value. The selection of a specific strategy is related to the kind of biological information that the inference algorithm will try to extract from the data, e.g. in the case of inference of gene regulatory networks based on association rules, the most common approach is to use the gene expression profile to determine the discrete states of a gene $g_i$ [24]. It is also important to consider that the more reduced is the 'data scope' for the computation of the discrete states, the discretization approach will be more sensitive to noise, but it will also be more capable of capturing small variations in the gene expression pattern than those with greater 'data scopes'.

Figure 2 resumes the abovementioned features for the characterization of gene expression discretization. Other criteria more related to general data mining approaches may also be applied. For example, the 'Static versus Dynamic' characteristic refers to the moment and independence at which the discretization process operates in relation to the inference algorithm. A dynamic discretization process acts when the learner is building the model, while a static discretization method proceeds before the inference task and it is independent of the learning algorithm [12]. However, almost all known discretization processes applied over GED are static. For other features applied to discretization in general data mining methodologies, please refer to Garcia *et al.* [12].

## Algorithms for discretization of GED

In this section, the classical and current state-of-the-art methods for discretization of GED will be briefly reviewed, starting with the unsupervised approaches and followed by the supervised procedures.

## Unsupervised discretization of GED

As was previously stated, the unsupervised discretization does not rely on any class label information for the computation of the discrete states of the genes. The discrete values are only calculated from GED. The simplest approach uses some measure to compute a threshold from which the state of a gene is determined. Madeira and Oliveira [8] proposed a classification for these approaches based on the 'sample type' at which they are aimed. The first one is the 'discretization using absolute values', which can be used in all GED because they discretize the absolute gene expression values directly using different techniques. The second one is the 'discretization using expression variations between time points', which is only applicable to time series expression data and computes variations between each pair of consecutive time points.

For simplicity in the formulation of the measures used to describe the discretization approaches, some metrics need to be introduced. Let $a'_{IJ}$ denote the average value in the expression matrix A', and let $a'_{iJ}$ and $a'_{Ij}$ represent the mean of row $i$ and column $j$, respectively. Let $H_{IJ}$ ($U_{IJ}$) refer to the maximum (minimum) value in the expression matrix A', and let $H_{iJ}$ ($U_{iJ}$) and $H_{Ij}$ ($U_{Ij}$) be the maximum (minimum) value of row $i$ and column $j$, respectively. In the same way, let $M_{IJ}$ stand for the median value in the expression matrix A', and $M_{iJ}$ and $M_{Ij}$ represent the median value of row $i$ and column $j$, respectively.

### Discretization using absolute values

This subsection describes those approaches that discretize the absolute gene expression values directly using different techniques. In this article, these methods will be further classified into 'discretization based on metrics', 'discretization based on ranking' and 'discretization based on clustering'.

#### *Discretization based on metrics*

The approaches based on metrics use a measure to compute the cut points $P$ for the gene $g_i$ in A' to determine the corresponding discrete state. In general, the metric can be computed with different 'data scopes', i.e. the discrete value $a_{ij}$ might be determined regarding the gene profile, the experimental

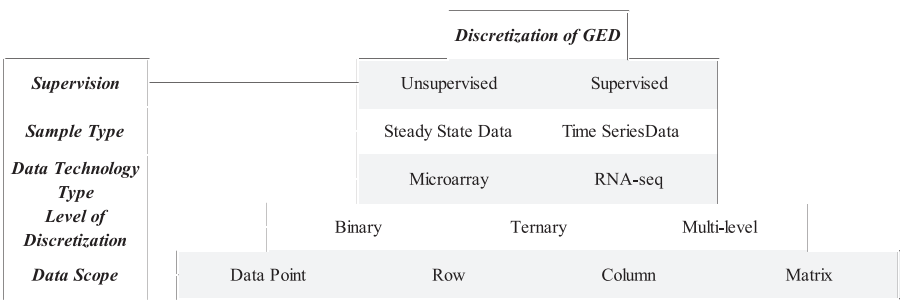


**Figure 2.** Main features of gene expression discretization with their multiple variants.

condition profile or the whole matrix. When the goal is to discretize the matrix $A'$ with a 'level of discretization' of two, these approaches follow the basic formulation given in Equation 1 to determine $a_{ij}$, where $\delta$ represents the metric used in the computation:

$$a_{ij} = \begin{cases} 1 & if \quad a'_{ij} \geq \delta \\ 0 & otherwise \end{cases} \quad (1)$$

In other words, a binary matrix with two symbols, one for 'activation' and another one for 'inhibition' (for instance, 1 and 0 as in Equation 1) is constructed. The simplest approach is to define $\delta$ as the average expression value of a specific 'data scope', i.e. averaging all the values in the matrix ($a'_{IJ}$), the values of the rows ($a'_{iJ}$) or the values of the columns ($a'_{Ij}$) [8, 24]. Some examples of the application of this approach for gene expression profiles can be found in Soinov *et al.* [25], Li *et al.* [26] and Ponzoni *et al.* [27]. These studies use the average value of the gene expression profiles aiming at reconstructing gene regulatory networks, discretizing the target genes of such interactions and inferring the relations by means of decision trees [25, 26] or by combinatorial optimization [27].

Other variations of the previous approach were assessed in Becquet *et al.* [28] and Pensa *et al.* [29]. For instance, $\delta$ of the Equation 1 can be defined as the median value $M$ (known as 'Mid-Range'), or as some sort of expression considering a fixed proportion $x$ regarding the max value $H$ (known as *Max - X%Max*) [8]. Also, as before, $M$ and $H$ can be computed in a specific 'data scope', that is, with respect to the gene expression profile (with $M_{Ij}$ and $H_{Ij}$), the condition expression profile (with $M_{iJ}$ and $H_{iJ}$) or the whole expression matrix (with $M_{IJ}$ and $H_{IJ}$). Becquet *et al.* [28] used these approaches to perform association rule mining on GED, whereas Pensa *et al.* [29] assessed these methods in the context of hierarchical clustering of GED.

When considering a 'level of discretization' of three, the most common approach is given in the Equation 2:

$$a_{ij} = \begin{cases} -1 & if \quad a'_{ij} < \delta \\ 1 & if \quad a'_{ij} > \delta \\ 0 & otherwise \end{cases} \quad (2)$$

The GED $A'$ is discretized using three symbols (for instance, $-1$, 0 and 1) meaning 'downregulated', 'upregulated' or 'no-change'. In this case, $\delta$ is defined as the average expression value combined with its standard deviation. Let $\alpha$ be a parameter used to tune the desired deviation from average and let $\sigma_{IJ}$, $\sigma_{iJ}$ and $\sigma_{Ij}$ be the standard deviations regarding different 'data scopes'. Then, $\delta$ can be defined as $a'_{IJ} \pm \alpha\sigma_{IJ}$, $a'_{iJ} \pm \alpha\sigma_{iJ}$ or $a'_{Ij} \pm \alpha\sigma_{Ij}$, i.e. by means of the values in the matrix, the values in the row $i$ or the values in the column $j$, respectively [8, 24].

Another possibility is to allow a multilevel discretization. This can be achieved by the 'Equal Width Discretization' (*EWD*) in which each cut point $p_r$ of $P$ is calculated by means of $p_{r+1} = p_r + (H - U)/k$, with $p_0 = U$, assigning the corresponding $r \in \sum$ to the values $a'_{ij}$ that satisfy $p_r < a'_{ij} \leq p_{r+1}$. In other words, the *EWD* divides the difference between the maximum and minimum values $H$ and $U$, respectively, into $k$ intervals of equal width, with $k$ being the user-defined parameter that determines the 'level of discretization'. As before, this can be done regarding different 'data scopes': the gene expression profile (with $H_{iJ}$ and $U_{iJ}$), the conditions expression profile (with $H_{Ij}$ and $U_{Ij}$) or the whole expression matrix (with $H_{IJ}$ and $U_{IJ}$).

Examples of applications of these two approaches for the biclustering of time series GED can be found in Madeira and Oliveira [8] and Mahanta *et al.* [19].

### *Discretization based on ranking*

Let us assume that the expression values are sorted in decreasing order on a list $L$. A simple approach is to assign the first $x$% values of $L$ to 1, whereas the other values are assigned to 0. This approach is known as *Top %X* [8, 12]. Another related approach that allows a multilevel discretization is based on the equal frequency principle. This method, known as 'Equal Frequency Discretization' (*EFD*) [2], considers a given number $k$ of symbols into which the expression values will be discretized. Then $L$ is split in $k$ segments of length $|L|/k$ containing the same number of data points per symbol, thus assigning the $k$ discrete states accordingly to the decreasing order of the segments. The approach based on the median value $M$, described earlier, corresponds to the special case when only two symbols are considered. As before, these methods can be applied using different 'data scopes'. The studies of Madeira and Oliveira [8], Mahanta *et al.* [19] and Lonardi *et al.* [30] contain examples of applications of these methods in the biclustering of GED.

### *Discretization based on clustering*

Other approaches that deal with the discretization of GED are based on clustering [7, 14, 18, 19, 24]. The way to achieve this is to consider each value $a'_{ij}$ of the GED $A'$ as an element of a single-dimensional space $\Omega$. Then, a clustering algorithm is applied to the $S$ elements of $\Omega$ that corresponds to a specific 'data scope' (a gene profile, a column profile or a matrix profile) to obtain groups of values, where the values belonging to the same group are assigned to the same discrete state. The groups are calculated by maximizing the similarity within the elements of each cluster, while minimizing this value among elements in different clusters. A common quality metric for the clusters is the *WCSS* (Within-Cluster Sum of Squares), defined as follows for a given discretization scheme $D$:

$$WCSS(D) = \sum_{a'_{ij} \in [p_0, p_1]} |a'_{ij} - \mu_0|^2 + \sum_{r=1}^{k-1} \sum_{a'_{ij} \in (p_r, p_{r+1}]} |a'_{ij} - \mu_r|^2 \quad (3)$$

Where $\mu_r$ is the mean of the $a'_{ij} \in (p_r, p_{r+1}]$. Basically, the *WCSS* is the sum of the squared Euclidean distance between the elements within a cluster and the mean of that cluster, where lower values mean higher similarities between the elements of the clusters.

Regarding the 'level of discretization', it depends on the clustering algorithm used in the task, although it is clear that multiple discrete states may be allowed with the appropriate approach. However, let us first consider the simplest case of a 'level of discretization' of two. Because the elements of $S$ are in a single-dimensional space, there is a total ordering between them. Therefore, when the number of cluster is two, a partition of $S$ can be optimally found with the following procedure [7]: first, let us define $L$ as a sorted list of the elements of $S$, with $L(e)$ representing the $e$-*th* element in the list, $0 \leq e < |S|$. Then, the optimal discretization scheme $D = \{[L(0), L(p)], (L(p), L(|S|)]\}$ for $S$ is calculated by finding the cut point $L(p)$, with $0 < p < |S| - 1$, so that $WCSS(D)$ has the minimum value. Finally, the expression values $a'_{ij}$ in $S$ that satisfy $a'_{ij} \leq L(p)$ are discretized to 0 (*inhibition*), and the expression values $a'_{ij}$ in $S$ that satisfy $a'_{ij} > L(p)$ are discretized to 1 (*activation*). In this approach, $p$ varies between (not inclusive) 0 and $|S| - 1$ to avoid the effect of outliers [7]. Figure 3 depicts an example of the

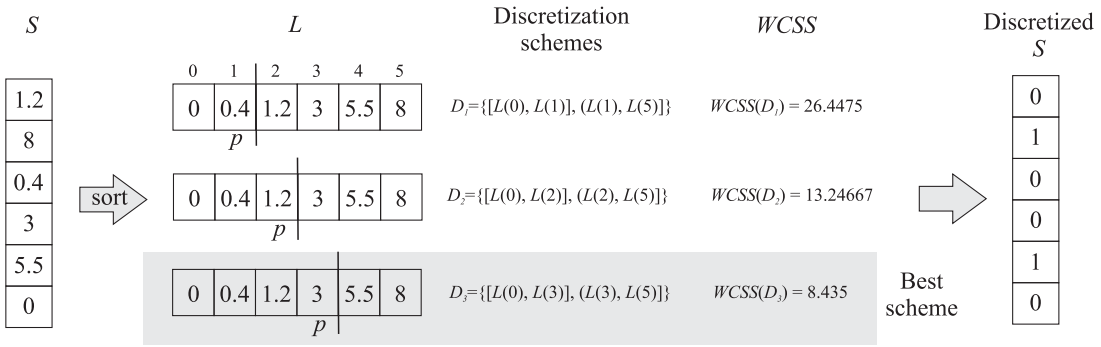


**Figure 3.** Simple clustering approach for a 'level of discretization' of two, where S represents the expression values to be discretized. After sorting the expression values in a list L, the WCSS of all the discretization schemes $D_i$ such that $1 < p < 4$ are calculated. Then, the best scheme is the one with lower WCSS, thus given the discretized expression values of S as shown.

**Table 1.** An example of the *Bikmeans* discretization with $k = 3$ regarding the discretization obtained with the $(k+1)$-*means* algorithm in the gene profile ($a^g_{ij}$) and in the condition profile ($a^c_{ij}$). The discrete state $a_{ij}$ (bolded), with $1 \leq a_{ij} \leq k$, is assigned to $a'_{ij}$ if $(a_{ij})^2 \leq a^g_{ij}\, a^c_{ij} < (a_{ij}+1)^2$.

| Discretized condition profile value $a^c_{ij}$ | Discretized gene profile value $a^g_{ij}$ | | | |
|---|---|---|---|---|
| | 1 | 2 | 3 | 4 |
| 1 | $1*1=1 \rightarrow \mathbf{a_{ij}=1}$ | $1*2=2 \rightarrow \mathbf{a_{ij}=1}$ | $1*3=3 \rightarrow \mathbf{a_{ij}=1}$ | $1*4=4 \rightarrow \mathbf{a_{ij}=2}$ |
| 2 | $2*1=2 \rightarrow \mathbf{a_{ij}=1}$ | $2*2=4 \rightarrow \mathbf{a_{ij}=2}$ | $2*3=6 \rightarrow \mathbf{a_{ij}=2}$ | $2*4=8 \rightarrow \mathbf{a_{ij}=3}$ |
| 3 | $3*1=3 \rightarrow \mathbf{a_{ij}=1}$ | $3*2=6 \rightarrow \mathbf{a_{ij}=2}$ | $3*3=9 \rightarrow \mathbf{a_{ij}=3}$ | $3*4=12 \rightarrow \mathbf{a_{ij}=3}$ |
| 4 | $4*1=4 \rightarrow \mathbf{a_{ij}=2}$ | $4*2=8 \rightarrow \mathbf{a_{ij}=3}$ | $4*3=12 \rightarrow \mathbf{a_{ij}=3}$ | $4*4=16 \rightarrow \mathbf{a_{ij}=3}$ |

previous procedure. This approach was applied in the work of Gallo *et al.* [7] to discretize gene expression profiles in the inference of gene association rules.

In the case of a multilevel discretization, the previous procedure is not applicable and the problem of finding the optimal partition becomes NP-Hard [31]. This means that the optimal partition of S cannot always be determined in a useful time and must be computed by algorithms that may not give the best solution. A widely used algorithm for this task is the *k-means clustering* [32]. The *k-means* uses the Squared Euclidean distance as a similarity measure, trying to yield a partition of elements with the least WCSS, as before. However, it follows a greedy approach to simplify the computation owing to the NP-Hardness of the problem. The main steps of the algorithm can be summarized as follows: first, the algorithm takes a set of points S and a fixed integer k as input. Then, it splits S into k subsets by choosing a set of k initial centroid points, where the elements of S are grouped regarding its nearest centroid to form the clusters. The next step is the recalculation of the centroids from the elements within the clusters. These two steps, i.e. cluster formation and centroid recomputation, are iterated until some stopping criterion is met (generally convergence). The choice of the initial centroid points is a key aspect of this algorithm, because it may influence the final structure of the partition. Given that a common approach is to start with random centroids, a different clustering of S may result every time the algorithm is run [24]. When dealing with GED, the most common approach is to use the *k-means* algorithm to discretize either the gene expression profiles or the condition expression profiles [18, 19]. In both cases, given a 'level of discretization' of k, the algorithm processes each expression profile independently, to discretize its values to one of the k discrete states. This requires N runs of the clustering algorithm to discretize the gene expression profiles, or M runs in the case of the condition expression profiles, thus increasing the computational complexity regarding the algorithms described in the previous sections.

Another approach, known as 'Bidirectional K-means Discretization' (*Bikmeans*) [18], uses both the clustering of gene profiles and column profiles using the *k-means* algorithm. That is, for a given 'level of discretization' of k, the algorithm computes the $(k+1)$-*means* clusters for the gene profiles, and for the condition profiles, independently. This gives two possible discrete states for each $a'_{ij}$, one for the gene profile and one for the condition profile, namely $a^g_{ij}$ and $a^c_{ij}$, respectively, with $1 \leq a^g_{ij} \leq k+1$ and $1 \leq a^c_{ij} \leq k+1$. Then, the discrete state $a_{ij}$, with $1 \leq a_{ij} \leq k$, is assigned to $a'_{ij}$ if $(a_{ij})^2 \leq a^g_{ij}\, a^c_{ij} < (a_{ij}+1)^2$. Table 1 shows an example of the possible discrete states for $a'_{ij}$ with $k = 3$, regarding $a^g_{ij}$ and $a^c_{ij}$. Note that in this case, for a given GED A', the *k-means* algorithm needs to be run $N + M$ times because both the gene profiles and condition profiles are clustered. This method was used by Li *et al.* [18] to discretize GED in the inference of gene regulatory networks.

Graph-based clustering algorithms can also be applied to the discretization of GED. In Dimitrova *et al.* [14], a method called 'Short Series Discretization' (*SSD*) was proposed for the multilevel discretization of short time series GED. *SSD* is a top down hierarchical clustering algorithm of gene profiles that define the distance between two clusters as the minimal distance between any pair of objects that do not belong to the same cluster simultaneously [14]. These objects are the real-value $a'_{ij}$ entries of the gene profile to be discretized, and the distance function that measures the distance between two gene profile entries $a'_{ij}$ and $a'_{il}$ is the one-dimensional Euclidean distance $|\, a'_{ij} - a'_{il}\, |$. As *SSD* follows a top down approach, it starts from the entire gene profile and iteratively splits it until either the degree of similarity reaches a certain threshold or every group consists of only one object. For the purpose of GED discretization, it is impractical to let the clustering algorithm produce too many clusters containing only one element. Thereby, the iteration at which the algorithm is terminated is crucial and determines the 'level of discretization'. The basic steps for the *SSD* algorithm are as follows: for each gene profile of M conditions, a completely

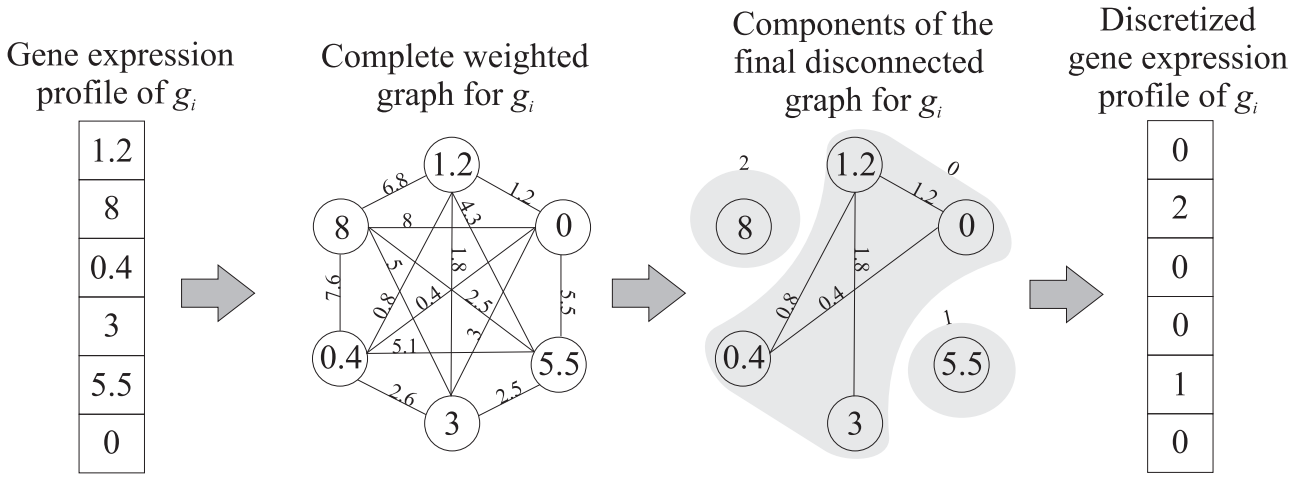


**Figure 4.** An example of the discretization of a gene expression profile $g_i$ with six experimental conditions using the *SSD* algorithm. First, a complete weighted graph for the gene $g_i$ is constructed, where each vertex is an expression value and each edge is the Euclidean distance between the vertexes. Then, the graph becomes disconnected until three components are obtained, discretizing the values according to the alphabet $\Sigma = \{0, 1, 2\}$.

weighted graph on $M$ vertices is constructed, where a vertex represents an entry on the gene profile and each edge has a weight of the Euclidean distance between its endpoints. The discretization process starts by deleting the edge(s) of highest weight until the graph becomes disconnected. If there is more than one edge labeled with the current highest weight, then all of the edges with this weight are deleted. The order in which the edges are removed leads to components, in which the distance between any two vertices is smaller than the distance between any two components [14]. The output of the algorithm is a discretization of the gene profile, in which each cluster corresponds to a discrete state and the gene profile entries that belong to one component are discretized into the same state. Owing to the computational cost involved in the process of recalculating the components of the graph on each edge deletion, this method is only aimed at time series data with few samples [14]. Figure 4 shows an example for a gene profile $g_i$ with six experimental conditions, discretized to an alphabet $\Sigma = \{0, 1, 2\}$. This method was assessed in the context of gene regulatory network inference from time series data [14].

So far, all the approaches described earlier were developed with the discretization of microarray GED in mind, without taking any special characteristic of the microarray data into consideration. Thus, they are also applicable to RNA-seq data. However, contemplating the particular characteristics of the GED that is being analyzed may lead to the development of better approaches. In Qu *et al.* [22], a new method for the discretization of RNA-seq GED was developed that combines data fitting an exponential distribution with a hierarchical clustering, to obtain a multilevel discretization with a matrix 'data scope'. Let us assume a given level of discretization of $k$. In essence, the algorithm consists of three steps: first, the raw RNA-seq GED is fitted to an exponential distribution, estimating the corresponding single parameter $\mu$. The second step is the partition of the estimated distribution in $k_1$ intervals of equal width, replacing the expression values $a'_{ij}$ in a certain interval with the mean of the values of that interval. Here $k_1$ acts as a sampling rate for the estimated distribution, where a large enough value allows for better robustness of the hierarchical clustering procedure [22]. Finally, the $k_1$ mean values are merged with the $k$ clusters by means of hierarchical clustering. Figure 5 depicts the workflow of the algorithm. Qu *et al.* [22] compared this method for discretization against *k-means* [31] (for gene and conditions profiles), *bikmeans* [19] and *EFD* [2] in the context of featured- and non-featured-based clustering of GED, and the results were assessed with several measures. In general, the method performs better than the other approaches, showing the importance of considering the specific characteristics of the data that are being discretized.

## Discretization Using Expression Variations between Time Points

A different approach for the discretization of GED is to consider the variation between time points, instead of the absolute gene expression values as was described previously. In this case, the methods are only applicable to time series GED, as they rely on the columns representing different time points in the same experiment, thus computing how the expression profiles evolve through time to perform the discretization. Therefore, the only meaningful 'data scopes' for these methods are the gene expression profile or the data point scope depending on the approach involved. There are several proposed discretization techniques based on the transitions in expression values between successive time points [8, 24]. Usually, these methods only allow a 'level of discretization' of two or three discrete states, depending on how they are formulated. In this case, the discrete state indicates the change over time of the gene expression, i.e. the changing tendencies of the genes. Also, the discretization of a GED matrix A' using these approaches produces a discretized matrix A with $M-1$ conditions [8, 24].

The first and simplest approach applied to GED that follows this idea is called 'Transitional State Discrimination' (*TSD*) [33], which is a method that discretizes gene profiles of GED with a 'level of discretization' of two. The main steps of the algorithm can be summarized as follows: first, the gene profiles of the GED A' are standardized using z-scores, scaling the expression values to a mean of zero and a unit of standard deviation. Then, each gene expression profile is discretized using the following scheme:

$$a_{ij} = \begin{cases} 1 & \text{if} \quad a'_{ij} - a'_{i(j-1)} \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

In this way, the GED matrix A' is transformed to a discrete matrix A of $N$ genes and $M-1$ conditions. This method was developed by Moller-Levet *et al.* [33] to perform GED clustering

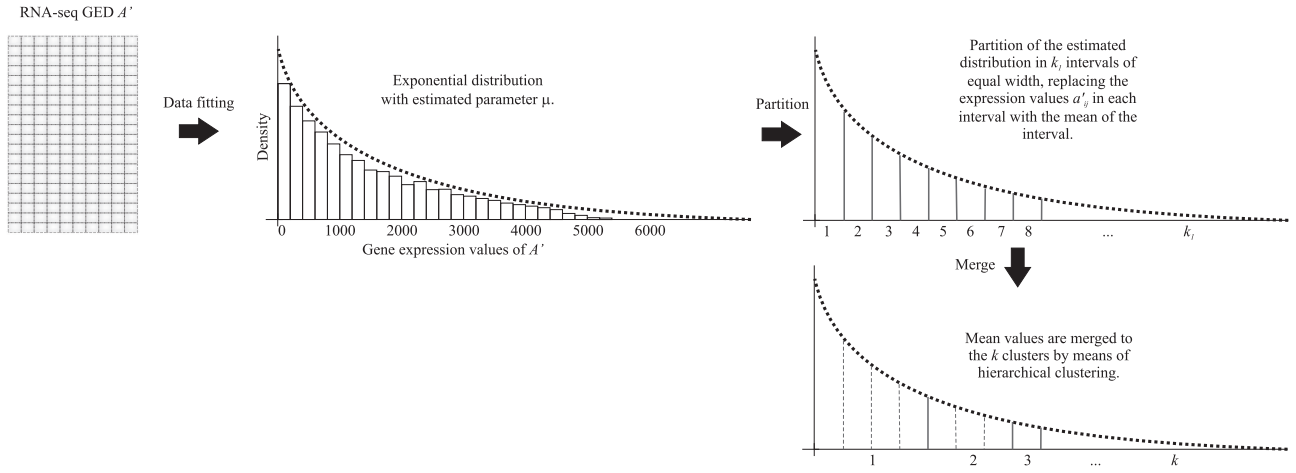


**Figure 5.** Workflow of the RNA-seq discretization approach proposed by Qu *et al.* [22].

based on temporal variation. A related method was developed by Erdal *et al.* [34], also applied to GED clustering, in which they compute the absolute differences between successive time points, and introduce a threshold $t$ for the 'upregulated' discrete state, as follows:

$$a_{ij} = \begin{cases} 1 & if \quad |a'_{ij} - a'_{i(j-1)}| \geq t \\ 0 & otherwise \end{cases} \tag{5}$$

Note that in both previous approaches the 'data scope' involved in the calculation of each discrete state consists of only one point.

Now consider a 'level of discretization' of three. A simple approach to achieve this is to combine the mean discretization with the variations between time points [25, 27]. In this case, the first step is to discretize the GED matrix A' using absolute values, with the mean discretization approach for the gene profile scope, as described earlier. This gives an intermediate discrete matrix A''. Then, each discrete state is calculated as follows:

$$a_{ij} = (a''_{ij} - a''_{i(j-1)}) \tag{6}$$

This approach gives a discretized matrix A of $N$ genes and $M-1$ conditions, in which each $a_{ij}$ may have one of three discrete states: 1, −1 and 0, meaning 'increase', 'decrease' and 'no-change' respectively. This method was used by Soinov *et al.* [25] and Ponzoni *et al.* [27] to infer changed state rules in the reconstruction of gene regulatory networks.

Another approach consists of analyzing variations between successive time points as before, but considering that these variations are significant whenever they exceed a given preset threshold [8, 24, 35, 36]. Thus, the discretized matrix A can be obtained after two steps: the first step transforms the GED matrix A' into a matrix A'' of variations such that:

$$a''_{ij} = \begin{cases} \dfrac{a'_{ij} - a'_{i(j-1)}}{|a'_{i(j-1)}|} & if \quad a'_{i(j-1)} \neq 0 \\ 1 & if \quad a'_{i(j-1)} = 0 \wedge a'_{ij} > 0 \\ -1 & if \quad a'_{i(j-1)} = 0 \wedge a'_{ij} < 0 \\ 0 & if \quad a'_{i(j-1)} = 0 \wedge a'_{ij} = 0 \end{cases} \tag{7}$$

In the second step, once the matrix A'' is calculated, the final discretized matrix A is obtained considering a threshold $t > 0$ as follows:

$$a_{ij} = \begin{cases} 1 & if \quad a''_{ij} \geq t \\ -1 & if \quad a''_{ij} \leq -t \\ 0 & otherwise \end{cases} \tag{8}$$

There are several examples of this approach in the context of clustering and biclustering of time series GED [8, 24, 35, 36].

All the methods described in this section discretize the GED by only considering the expression values of the genes. In the next section, another kind of approach will be described that uses additional information besides the expression values to perform the discretization.

## Supervised discretization of GED

As it was aforementioned, most methods developed to deal with the discretization of GED are unsupervised. Nonetheless, there are some approaches that use supervised methods and, in general, they consider prior biological knowledge for performing the discretization. Given a GED matrix A' of $N$ genes and $M$ conditions, a set of classes $\Gamma$ and a matrix $C$ (of the same dimensionality as A'), a supervised discretization approach will take A' and $C$ as input, where $C$ maps each $a'_{ij}$ of A' into a target class label $c \in \Gamma$. Then, the supervised approach will try to obtain a discretized matrix A that best fits the target class label information of $C$ with the continuous expression values of A'. In this way, the level of discretization will be given by the number of classes (i.e. $|\Gamma| = k$).

Usually, the supervised approaches are aimed to discretize GED in the context of sample classification of GED, i.e. given a steady state GED A', the set of samples $J$ can be partitioned into $k$ classes, where each $J_l \subseteq J$ set, with $0 < l \leq k$, corresponds to an experimental condition (i.e. class). Thus, the main goal is to build a sample classifier to determine the corresponding class of a given condition profile. The typical examples are those of GED related to cancer, where a set of conditions corresponds to healthy samples (control), and the other set of conditions corresponds to cancerous samples (typically of a specific type). Here, the discretization of the GED allows the application of classifiers that require discretized data as input. Although it is

clear that these GED can be discretized using unsupervised approaches, the idea is to improve the outcomes of the discretization by using the class label information available, leading to better sample classifiers for the GED.

To describe some proposed supervised discretization approaches, let us look at some useful definitions, extending the concepts given previously. Let $S$ be the list of $N \times M$ pairs of elements, $S = \{S_1, S_2, \ldots, S_{N\times M}\}$, such that each element $S_t$ represents the mapping function from the element $a'_{ij}$ of $A'$ to the corresponding element $c_{ij}$ of $C$. Consider that $S$ is sorted in ascending order of the $a'_{ij}$ elements, which means that for all $t$ from 1 to $N \times M$, if $S_{t-1} = (a'_{ij}, c_{ij})$ and $S_t = (a'_{i'j'}, c_{i'j'})$ then $a'_{ij} \leq a'_{i'j'}$. Let $L[e_f : e_l]$ be the sub-list of first elements from the $e_f$-$th$ pair to the $e_l$-$th$ pair in $S$, with $1 \leq e_f < e_l \leq N \times M$. That is, if $S_e = (a'_{ij}, c_{ij})$ and $S_{e'} = (a'_{i'j'}, c_{i'j'})$, then $L[e_f{:}e_l]$ defines the expression values of $A'$ going from $a'_{ij}$ to $a'_{i'j'}$ in ascending order. In particular, $L[1{:}NxM]$ represents all the expression values of the GED $A'$ sorted in ascending order, and from now on they will be referred to simply as $L$. Thus, a discretization scheme of $L$ can be represented by the set of $k$ intervals: $D_{e_k} = \{L[1 : e_1], L[(e_1 + 1) : e_2], \ldots, L[(e_{(k-1)} + 1) : e_k]\}$. For example, a two-interval discretization of $L = \{0.5, 0.7, 0.9, 1.2, 1.6, 2\}$ is $D_{e_2} = \{L[1 : e_1], L[(e_1 + 1) : e_2]\}$. If $e_1 = 3$ and $e_2 = 6$, then $D_{e_2} = D_6 = \{L[1 : 3], L[4 : 6]\} = \{\{0.5, 0.7, 0.9\}, \{1.2, 1.6, 2\}\}$ is a possible discretization. The discretization scheme $D_{e_k}$ defines $k - 1$ cut points. In the previous example, $D_6$ defined a cut point between 0.9 and 1.2.

These concepts were used in a well-known supervised discretization approach developed by Fayyad and Irani [37], and applied to GED in Lustgarten *et al.* [38]. The *FI* method [37] (for its author's initials) selects a cut point, $p$, in a given interval in a greedy way and continues recursively in the subintervals defined by $p$. The procedure is undertaken in two principal steps:

i) *Calculate the score of each interval in $D_{e_k}$ as the entropy of the target values ($c_{ij}$'s).* For an interval $L[e_{k-1} : e_k]$ derived from the values of a gene expression matrix $A'$, and a target class label $c$ belonging to $\Gamma$, which can take $k = |\Gamma|$ values, the entropy can be defined as:

$$\varepsilon(L[e_{k-1} : e_k]) = \sum_{j=1}^{|\Gamma|} P(c = c_j)\log_2(P(c = c_j)) \tag{9}$$

Where $P(c = c_j)$ is the probability that an instance in the current interval takes the value $c_j$.

ii) *Discretize each interval recursively into two new subintervals.* Given an interval $L[e_f : e_l]$ and its entropy $\varepsilon(L[e_{k-1} : e_k])$, a cut point $p$ can be greedily specified if we try to minimize the joint entropy of the subintervals defined by $p$ in $L[e_{k-1} : e_k]$:

$$\varepsilon(p; L[e_{k-1} : e_k]) = \frac{|L[e_{k-1} : e_p]|}{L[e_{k-1} : e_k]}\varepsilon(L[e_{k-1} : e_p]) + \frac{|L[e_{p+1} : e_k]|}{L[e_{k-1} : e_k]}\varepsilon(L[e_{p+1} : e_k]) \tag{10}$$

Where $L[e_{k-1} : e_p]$ and $L[e_{p+1} : e_k]$ are the new two subintervals defined by the cut point $p$ in $L[e_k : e_{k+1}]$.

The *FI* method does not guarantee that the optimal cut point will be discovered with minimum entropy because it does not accomplish an exhaustive search. However, when only dealing with two classes, the optimal cut point with minimum entropy can be computed in a greedy manner [7, 27]. Gallo *et al.* [7] and Ponzoni *et al.* [27] used this approach to compute the discretization of the regulator genes in the inference of gene association rules from time series GED.

Another supervised discretization approach for GED, called Efficient Bayesian Discretization (*EBD*), was proposed by Lustgarten *et al.* [38]. It is based on the method developed by Boullé [39] that uses dynamic programming to search all the possible discretizations and a Bayesian measure to score each one of them, thus ensuring the optimal one is found. In the case of *EBD*, it improves the method proposed by Boullé [39] by allowing the incorporation of prior knowledge and decreasing the algorithm time complexity [38]. The *EBD* algorithm consists of two principal steps:

a) *Evaluate each discretization by means of the score.*

$$Score(M) = P(M) \cdot P(S/M) \tag{11}$$

Which is the numerator of the 'posterior probability' given by 'Bayes rules', where $M$ is the discretization model corresponding to the discretization $D_{e_k}$ and is defined as $M \equiv \{\,|\,D_{e_k}\,|\,, D_{e_k}, \Theta\}$. In other words, the model is conformed by the number of intervals in the discretization $D_{e_k}$, the discretization $D_{e_k}$ itself and the set of probabilistic parameters corresponding to a multinomial distribution. In Equation 11, $P(M)$ is the prior probability of $M$ and $P(S/M)$ is the marginal likelihood of the data in $S$ given the model $M$. In Lustgarten *et al.* the authors use:

$$P(S/M) = \prod_{i=1}^{I} \frac{(\,|\,\Gamma\,|\,-1)!}{(\,|\,\Gamma\,|\,-1+n_i)!} \prod_{j=1}^{|\Gamma|} n_{ij}! \tag{12}$$

Where $I$ is the number of intervals in the discretization of the model $M$, $|\Gamma|$ is the number of possible values for the target variable, $n_i$ is the number of instances in the interval $I$ and $n_{ij}$ is the number of instances in the interval $i$ that have taken the target value $j$.

b) *Search all the possible discretizations using dynamic programming.* This strategy allows reusing the previously computed optimal solutions that have been obtained in a smaller instance of the same problem.

In Lustgarten *et al.* [38], both the *FI* and *EBD* methods were used to discretize GED obtained from high-throughput transcriptomic and proteomic studies, to build classifiers of GED samples.

The last supervised discretization approach that we will consider was proposed by Wang *et al.* [40]. In this study, the authors take advantage of gene expression information to locate the interval's cut points. Let the expression range of a gene be divided into $m$ left-side-half-open segments. Let $l_i = 1, 2, \ldots, m$, be the superior boundaries corresponding to each segment. Let $v_i = (-\infty, l_i]$ be the $i$-th half open interval. Wang *et al.* defined the 'Class Distribution Diversity' (*CDD*) of $v_i$, denoted $CDD(v_i)$, for a binary classification problem as:

$$CDD(v_i) = \frac{n_1(v_i)}{N_1} - \frac{n_2(v_i)}{N_2} \tag{13}$$

Where $n_1(v_i)$ and $n_2(v_i)$ are the number of cases belonging to *class 1* and *class 2* in the interval $v_i$, and $N_1$ and $N_2$ are the total number of samples of class 1 and class 2, respectively. Depending on the gene expression values, the *CDD* allows the presence of zero and one or two possible cut points in a gene expression range. Let us suppose that $v_{max}$ ($v_{min}$) is the interval

**Table 2.** Summary of the revised methods' main features

| Method | Supervision | Sample type | Data technology type | Level of discretization | Data scope | Evaluation measure | Requires input parameter | Application examples on GED |
|---|---|---|---|---|---|---|---|---|
| *Bikmeans* | Unsupervised | Steady State/ Time Series | RNA-seq/Microarray | Multilevel | Row/Column | Euclidean distance | Yes | Li *et al.* [18], Mahanta *et al.* [19] |
| *EBD* | Supervised | Steady State/ Time Series | RNA-seq/Microarray | Multilevel | Row | Bayesian score | Yes | Lustgarten *et al.* [38], Wang *et al.* [40] |
| *EFD* | Unsupervised | Steady State/ Time Series | RNA-seq/Microarray | Multilevel | Row/Column/ Matrix | Equal frequency | Yes | Madeira and Oliveira [8], Li *et al.* [18], Mahanta *et al.* [19] |
| Entropy based discretization | Supervised | Steady State/ Time Series | RNA-seq/Microarray | Binary | Row | Partition entropy | No | Gallo *et al.* [7], Ponzoni *et al.* [27] |
| Erdal's *et al.* method | Unsupervised | Time Series | RNA-seq/Microarray | Binary | Data point | Absolute difference between consecutive samples | Yes | Madeira and Oliveira [8], Erdal *et al.* [34] |
| *EWD* | Unsupervised | Steady State/ Time Series | RNA-seq/Microarray | Multilevel | Row/Column/ Matrix | Equal width | Yes | Madeira and Oliveira [8], Li *et al.* [18], Mahanta *et al.* [19] |
| *FI* | Supervised | Steady State/ Time Series | RNA-seq/Microarray | Multilevel | Row | Entropy and MDL | Yes | Lustgarten *et al.* [38], Wang *et al.* [40] |
| Gallo *et al.* method | Unsupervised | Steady State/ Time Series | RNA-seq/Microarray | Binary | Row | Euclidean distance | No | Gallo *et al.* [7] |
| Ji and Tan method | Unsupervised | Time Series | RNA-seq/Microarray | Ternary | Data point | Ratio between consecutive samples | Yes | Madeira and Oliveira [8], Ji and Tan [35] |
| *K-means clustering* | Unsupervised | Steady State/ Time Series | RNA-seq/Microarray | Multilevel | Row/Column/ Matrix | Euclidean distance | Yes | Li *et al.* [18], Mahanta *et al.* [19] |
| Max - X%Max | Unsupervised | Steady State/ Time Series | RNA-seq/Microarray | Binary | Row/Column/ Matrix | Max value | Yes | Madeira and Oliveira [8], Becquet *et al.* [28], Pensa *et al.* [29] |
| Mean | Unsupervised | Steady State/ Time Series | RNA-seq/Microarray | Binary | Row/Column/ Matrix | Mean | No | Madeira and Oliveira [8], Mahanta *et al.* [19], Li *et al.* [26], Ponzoni *et al.* [27] |
| MeanPlusEstDev | Unsupervised | Steady State/ Time Series | RNA-seq/Microarray | Ternary | Row/Column/ Matrix | Mean plus standard deviation | Yes | Madeira and Oliveira [8] |
| Median | Unsupervised | Steady State/ Time Series | RNA-seq/Microarray | Binary | Row/Column/ Matrix | Median | No | Madeira and Oliveira [8], Becquet *et al.* [28], Pensa *et al.* [29] |
| Qu *et al.* method | Unsupervised | Steady State/ Time Series | RNA-seq | Multilevel | Matrix | Exponential distribution | Yes | Qu *et al.* [22] |
| Soinov's change state | Unsupervised | Time Series | RNA-seq/Microarray | Ternary | Row | Mean and difference between consecutive samples | No | Soinov *et al.* [25], Ponzoni *et al.* [27] |
| *SSD* | Unsupervised | Time Series | RNA-seq/Microarray | Multilevel | Row | Euclidean distance | No | Dimitrova *et al.* [14] |
| Top %X | Unsupervised | Steady State/ Time Series | RNA-seq/Microarray | Binary | Row/Column/ Matrix | Max value | Yes | Madeira and Oliveira [8], Becquet *et al.* [28], Pensa *et al.* [29] |
| *TSD* | Unsupervised | Time Series | RNA-seq/Microarray | Binary | Data point | Difference between consecutive samples | No | Madeira and Oliveira [8], Möller-Levet *et al.* [33] |
| Wang *et al.* method | Supervised | Steady State/ Time Series | RNA-seq/Microarray | Ternary | Row | Class distribution diversity | Yes (2) | Wang *et al.* [40] |

upper bounded by $l_{max}$ ($l_{min}$) with maximum (minimum) *CDD*, denoted by $CDD_{max}$ ($CDD_{min}$). Then, given a gene expression profile, its values of $CDD_{max}$ and $CDD_{min}$ could be in one of the next cases: (i) $CDD_{max} > 0$, $CDD_{min} = 0$; (ii) $CDD_{max} = 0$, $CDD_{min} < 0$; and (iii) $CDD_{max} > 0$ and $CDD_{min} < 0$. Thus, the authors defined the 'discriminative power of a gene' as the absolute value of the difference between $CDD_{max}$ and $CDD_{min}$, and they used that value to determine the number of cut points (zero, one or two) and the cut points themselves to accomplish a possible discretization. Wang *et al.* [40] assessed this method in the context of binary classification scenarios, and showed that it performs better in comparison with the *FI* and *EBD* methods described earlier.

## Discussion

Advances in microarray and RNA-seq technologies allow the simultaneous measurement of the expression of thousands of genes under different experimental conditions, enabling the unraveling and reverse-engineering of the interactions of the genes in an organism. Several data mining and machine learning algorithms have been developed to discover those interactions from the GED, and in several cases they require discrete data as inputs to make the inference. In this regard, the discretization of the data plays a key role in the outcomes of the gene expression analysis.

A direct benefit of using a discrete view of the data is that it emphasizes the inference of knowledge only on a relevant pattern of the gene expression values. This may lead to better prediction models because the inherent noise of the data is removed. Also, the discretization improves the efficiency of the inference algorithms owing to the reduced search space in which the extraction of knowledge is performed. Furthermore, it helps with the interpretation of the results because each discrete state has a direct meaning in its value. Nonetheless, some of these asseverations depend mainly on the capabilities of the discretization method to capture the real pattern of gene expression. Even more, as every kind of discretization implies loss of information, a careful evaluation must be performed before proceeding with this preprocessing of the data.

In this article, we presented the main features of the discretization of GED and reviewed the classical and state-of-the-art approaches that deal with this task. These methods vary from simple formulations such as the average, to more complex approaches involving clustering, distribution fitting and in some cases class label information. Table 2 resumes all the revised methods together with their main features, aimed at helping the reader to further explore a particular approach of interest. In this regard, and to the best of our knowledge, there are no free software tools that provide a reasonable set of discretization methods for GED. The available free software for GED analysis focuses on the knowledge inference stage and provides only one or two of the most common discretization methods (such as *EFD*, *EWD*, *k-means* and *FI*). Thereby, we provide a free and open-source software called GEDPROTOOLS (http://lidecc.cs.uns.edu.ar/files/gedprotools.zip) that allows visualization and preprocessing of GED, providing most of the discretization approaches revised in this article.

As the reader might notice, the selection of a discretization method is not a trivial task. There are several topics to consider to make the correct choice for a particular instance in the addressed biological problem. First of all, there is the issue related to the 'level of discretization' used in the modeling, i.e. how many discrete states will be used to represent the expression levels of the genes. A discretization with two levels, 'upregulated' and 'downregulated', gives the simplest case for representation and the easiest way of interpretation of the results [3], although it does not allow any modeling of intermediate states. This last asseveration greatly limits the possibility of dynamic modeling given the case [3]. The important issue to consider here is that as the level of discretization increases, so does the complexity of the algorithms involved in the expression analysis (discretization and inference algorithm) and, at the same time, increasing the difficulties in interpreting the results.

Another issue to consider is the inherent computational complexity in the discretization approach, as this may add an undesirable load in the computational time required to perform the expression analysis. In general, a discretization performed on the gene profiles (condition profiles) requires $N$ ($M$) runs of the discretization algorithm to discretize the entire GED. If the GED has to be discretized as a whole (i.e. with a data matrix scope), then the genome-wide context of some GED with thousands of genes may impose an additional limit on the applicability of the discretization algorithm. The simplest methods based on metrics and ranking provide an efficient way of performing the discretization, as little computation effort is required. Nonetheless, these approaches may not be the best suited for the task. More elaborate methods, such as those based on clustering, tend to perform better in the discretization of GED [7, 14, 18, 19, 22], although they also require a significantly higher computational effort. Such is, for example, the case of the *SSD* algorithm that it is only applicable to short time series GED owing to its high computational complexity [14].

The data type of the GED, and the kind of information intended to infer from it, may also play an important role in the determination of the discretization approach. For example, if the goal is to study the change on the expression levels of the genes in time series GED, the methods based on expression variations between time points may represent an alternative worth exploring. On the other hand the supervised approaches may be more appropriate in the case of sample classification of healthy and cancerous steady state data.

Finally, there are other issues to consider that may not be so clear. For example, how are the results affected given a particular 'data scope' used in the discretization? Or how should the most adequate approach between algorithms of similar features be chosen? The difficulty lies in the fact that there may be approaches that perform well in some instances of the same biological problem and poorly in others, given the strong dependence on the particularities of each case. Therefore, a methodology for the selection of a discretization algorithm may be as follows: first, the essential features of the biological problem instance (level of discretization, data type, etc.) need to be determined. Then, the methods that satisfy those features are selected. Finally, if some indecisions persist that cannot be solved with an understanding of the chosen methods, the discretization approaches can be compared by analyzing the outcomes of the inference algorithm in each case and selecting the one that performs best. In this way, the ultimate impact of a specific discretization approach will be based on the predictive quality achieved by the inference algorithm responsible of extracting the biological knowledge. Thus, the metrics and validation procedures for predictive models will be adequate to indirectly assess the impact of the chosen discretization.

**Key Points**

- In gene expression data analysis, the discretization of the data is an important step when discrete states are required in the inference of knowledge, and plays a major role in the outcomes of the analysis.
- All types of discretization involve some degree of loss of information, and therefore, different variants of discretization may lead to different knowledge extraction (sometimes contradictory between them).
- The choice of a suitable discretization scheme may improve the performance of predictive models by reducing the noise inherent to the experimental data.
- There are several approaches to discretize gene expression data, each one requiring specific features of the particular gene expression analysis problem.
- A straightforward way to choose a discretization method is to determine the main characteristics of the gene expression analysis problem by following the features described in the article, and then selecting the approach that best meets those requirements.

## Funding

This work was supported by Consejo Nacional de Investigaciones Científicas y Técnicas [Grant number 112-2012-0100471] and Secretaria de Ciencia y Tecnología (UNS) [Grant numbers 24/N032, 24/ZN26].

## References

1. Friedman N, Goldszmidt M. Discretization of continuous attributes while learning Bayesian networks. In: Saitta L (ed). *ICML96. Proceedings of the 13th International Conference on Machine Learning; 1996 July 3-6; Bari, Italy*. San Francisco CA: Morgan Kauffman Publishers, 1996, 157–65.
2. Dougherty J, Kohavi R, Sahami M. Supervised and unsupervised discrimination of continuous Features. In: Prieditis A, Russell S (eds). *ICML95. Proceedings of the 12th International Conference on Machine learning; 1995 July 9-12; Tahoe City, United States*. San Francisco CA: Morgan Kauffman Publishers, 1995, 194–202.
3. Karlebach G, Shamir R. Modelling and analysis of gene regulatory networks. *Nat Rev Mol Cell Biol* 2008;**9**:770–80.
4. Alves R, Rodriguez-Baena DS, Aguilar-Ruiz JS. Gene association analysis: a survey of frequent pattern mining from gene expression data. *Brief Bioinform* 2009;**11**(2):210–24.
5. Vignes M, Vandel J, Allouche D, *et al*. Gene regulatory network reconstruction using bayesian networks, the Dantzig selector, the Lasso and their meta-analysis. *PLoS One* 2011;**6**(12):e29165.
6. Vijesh N, Chakrabarti SK, Sreekumar J. Modeling of gene regulatory networks: a review. *J Biomed Sci Eng* 2013;**6**:223–31.
7. Gallo CA, Carballido JA, Ponzoni I. Discovering time-lagged rules from microarray data using gene profile classifiers. *BMC Bioinformatics* 2011;**12**:123.
8. Madeira SC, Oliveira AL. An evaluation of discretization methods for non-supervised analysis of time-series gene expression data [Internet]. Lisboa: Instituto de Engenharia de Sistemas e Computadores Investigação e Desenvolvimento em Lisboa (INESC-ID); 2008 December [cited 2014 Dec 29]. Report No.: 42/2005. http://algos.inesc-id.pt/~jpa/InscI/poisson/varwwwhtml/portal/ficheiros/publicacoes/3369.pdf
9. Richeldi M, Rossotto M. Class-driven statistical discretization of continuous attributes. In: Lavrač N, Wrobel S (eds). *ECML '95. Proceedings of the 8th European Conference on Machine learning; 1995 April 25-27; Heraclion, Crete, Greece*. Berlin Heidelberg: Springer, 1995, 335–38.
10. Chlebus B, Nguyen SH. On finding optimal discretizations for two attributes. In: Polkowski L, Skowron A, (eds). *RSCTC '98. Proceedings of the First International Conference on Rough Sets and Current Trends in Computing; 1998 June 22-26; Warsaw, Poland*. Berlin Heidelberg: Springer, 1995, 537–44.
11. Cios KJ, Pedrycz W, Swiniarski RW, *et al*. *Data Mining: A Knowledge Discovery Approach* (1st edn). Springer: New York, 2007.
12. Garcia S, Luengo J, Sáez JA, *et al*. A survey of discretization techniques: taxonomy and empirical analysis in supervised learning. *IEEE Trans Knowl Data Eng* 2013;**25**:734–50.
13. Lazar C, Meganck S, Taminau J, *et al*. Batch effect removal methods for microarray gene expression data integration: a survey. *Brief Bioinform* 2012;**14**(4):469–90.
14. Dimitrova ES, Vera Licona MP, McGee J, *et al*. Discretization of time series data. *J Comput Biol* 2010;**17**(6):853–69.
15. Ding C, Peng H. Minimun redundancy feature selection from microarray gene expression data. *J Bioinform Comput Biol* 2005;**3**:185–93.
16. Golub TR, Slonim DK, Tamayo P, *et al*. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science* 1999;**286**:531–7.
17. Alon U, Barkai N, Notterman DA, *et al*. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *PNAS* 1999;**96**:6745–50.
18. Li Y, Liu L, Bai X, *et al*. Comparative study of discretization methods of microarray data for inferring transcriptional regulatory networks. *BMC Bioinformatics* 2010;**11**:520.
19. Mahanta P, Ahmed HA, Kalita JK, Bhattacharyya DK. Discretization in gene expression data analysis: a selected survey. In: *CCSEIT '12. Proceedings of the Second International Conference on Computational Science, Engineering and Information Technology; 2012 Oct 26-28; Coimbatore, India*. New York: ACM, 2012, 69–75.
20. Marioni JC, Mason CE, Mane SM, *et al*. RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome Res* 2008;**18**(9):1509–17.
21. Wang Z, Gerstein M, Snyder M. RNA-Seq: a revolutionary tool for transcriptomics. *Nat Rev Genet* 2009;**10**(1):57–63.
22. Qu J, Zhang J, Huang C, *et al*. A novel discretization method for processing digital gene expression profiles. In: *ISB 2013. 7th International Conference on Systems Biology; 2013 Aug 23-25; Rio Grande do Sul, Brasil*. Los Alamitos: IEEE, 2013, 134–8.
23. Spellman PT, Sherlock G, Zhang MQ, *et al*. Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization. *Mol Biol Cell* 1998;**9**(12):3273–97.
24. Gallo CA, Carballido JA, Ponzoni I. *Biological Knowledge Discovery Handbook: Preprocessing, Mining, and Postprocessing of Biological Data (Vol. 36)*. In: Elloumi M, Zomaya AY (eds). John Wiley & Sons: Hoboken, 2013, 803–40.
25. Soinov LA, Krestyaninova MA, Brazma1 A. Towards reconstruction of gene networks from expression data by supervised learning. *Genome Biol* 2003;**4**(1):R6.
26. Li X, Rao S, Jiang W, *et al*. Discovery of time-delayed gene regulatory networks based on temporal gene expression profiling. *BMC Bioinformatics* 2006;**7**:26.

27. Ponzoni I, Azuaje F, Augusto J, *et al*. Inferring adaptive regulation thresholds and association rules from gene expression data through combinatorial optimization learning. *IEEE/ACM Trans Comp Biol Bioinform* 2007;**4**(Suppl 4):624–34.
28. Becquet C, Blachon S, Jeudy B, *et al*. Strong-association-rule mining for large-scale gene-expression data analysis: a case study on human sage data. *Genome Biol* 2002;**3**(12): research0067.
29. Pensa RG, Leschi C, Besson J, *et al*. Assessment of discretization techniques for relevant pattern discovery from gene expression data. In: Zaki MJ, Morishita S, Rigoutsos I (eds). *BIOKDD 2004. Proceedings of the 4th Workshop on Data Mining in Bioinformatics; 2004 Aug. 22; Seattle, United States*. New York: ACM, 2004, 24–30.
30. Lonardi S, Szpankowski W, Yang Q. Finding biclusters by random projections. In: Sahinalp SC, Muthukrishnan S, Dogrusoz U (eds). *CPM 2004. Proceedings of the 15th Annual Symposium on Combinatorial Pattern Matching; 2004 Jul 5-7; Istanbul, Turkey*. Berlin: Springer Berlin Heidelberg, 2004, 102–16.
31. Aloise D, Deshpande A, Hansen P, *et al*. NP-hardness of Euclidean sum-of-squares clustering. *Mach Learn* 2009;**75**:245–9.
32. MacQueen J. Some methods for classification and analysis of multivariate observations. In: *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability; 1965 Jun 21-July 18, 1965 Dec 27, 1966 Jan 7; Berkeley, United States*. Berkeley: University of California Press, 1967, 281–97.
33. Möller -Levet C, Cho KH, Wolkenhauer O. Microarray data clustering based on temporal variation: FCV and TSD preclustering. *Appl Bioinform* 2003;**2**(1):35–45.
34. Erdal S, Ozturk O, Armbruster D, *et al*. A time series analysis of microarray data. In: *BIBE 2004. Proceeding of the 4rd IEEE Symposium on Bioinformatics and Bioengineering; 2004 May 19-21; Taichung, Taiwan*. Los Alamitos: IEEE, 2004, 366–75.
35. Ji L, Tan K. Mining gene expression data for positive and negative co-regulated gene clusters. *Bioinformatics* 2004;**20**(16):2711–18.
36. Madeira SC, Teixeira MC, Sa-Correia I, *et al*. Identification of regulatory modules in time series gene expression data using a linear time biclustering algorithm. *IEEE/ACM Trans Comput Biol Bioinform* 2010;**7**(1):153–65.
37. Fayyad U, Irani K. Multi-interval discretization of continuous-valued attributes for classification learning. In: *Proceedings of the International Joint Conference on Uncertainty in AI; 1993 Sep 1; Chambery, France*. Berlin Heidelberg: Springer, 1993, 1022–7.
38. Lustgarten JL, Visweswaran S, Gopalakrishnan V, *et al*. Application of an efficient Bayesian discretization method to biomedical data. *BMC Bioinformatics* 2011;**12**:309.
39. Boullé M. MODL: a Bayes optimal discretization method for continuous attributes. *Mach Learn* 2006;**65**:131–65.
40. Wang HQ, Jing GJ, Zheng C. Biology-constrained gene expression discretization for cancer classification. *Neurocomputing* 2014;**145**:30–6.