Applied Soft Computing 109 (2021) 107544

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

A metaheuristic multi-objective optimization method for dynamical network biomarker identification as pre-disease stage signal

Veredas Coleto-Alcudia, Miguel A. Vega-Rodríguez *

Department of Computer and Communications Technologies, University of Extremadura, Campus Universitario s/n, 10003 Caceres, Spain

ARTICLE INFO

Article history:
Received 21 December 2020
Received in revised form 25 April 2021
Accepted 22 May 2021
Available online 1 June 2021

Keywords:
Multi-objective optimization
Artificial bee colony algorithm
Pre-filtering
Dynamical network biomarker
Pre-disease stage

ABSTRACT

Deciphering the signals that are attached to the transition from normal to disease stage is crucial in preventive medicine to understand the progression of complex diseases. Between normal and disease stages there exists the pre-disease stage, in which the disease is yet reversible towards the normal stage. Traditionally, molecular biomarkers have been used to identify the pre-disease stage. However, they have limitations because they have an individual and static nature. In complex diseases, the dynamics and interplays of certain genes have to be taken into account in order to identify the pre-disease stage. Therefore, in complex diseases, it is necessary to use dynamical network biomarkers (DNBs). The development of time-course omics data has been crucial to the use of DNBs as biomarker. In this article, a new two-step method is proposed for the identification of DNBs as pre-disease stage signal. In the first step, the relevant genes in the dataset are pre-filtered using a differential gene expression analysis. In the second step, the DNBs are identified, from a multi-objective optimization viewpoint, by using an Artificial Bee Colony based on Dominance (ABCD) algorithm. Specifically, identified DNBs optimize three objectives: they are the smallest gene network that shows the strongest signal in the earliest time-point of the disease progression and best correlates with the disease phenotype. The proposed method has been evaluated with five time-course microarray datasets and the results have been compared with five methods from other authors, surpassing their results. The effectiveness of the proposed method has been also proved with a leave-one-out cross-validation and a Gene Ontology term enrichment. In fact, the proposed method obtains values around 90% for accuracy, precision, recall, and F1 scores.

1. Introduction

The identification of biomarkers that allows the study of disease stages (normal, pre-disease, and disease state) is crucial in the tracking of the complex diseases progression. Generally, the pre-disease stage of a disease is reversible to the normal stage, but in complex diseases this is difficult to detect [1]. Complex diseases as cancer, diabetes, lung injury, influenza, and Alzheimer's disease among others, are diseases caused by both genetic and environmental factors. Because of that, the progression of complex diseases is difficult to study [2].

Traditionally, disease biomarkers have been identified as individual molecules in different samples (control, disease) or conditions [3]. However, the static analysis of isolated molecules is not a proper way to identify biomarkers of a complex disease [4]. In complex diseases, not individual but dynamic changes among molecules occur. Therefore, interaction networks of molecules have been used for the study of complex diseases progression [5].

In the recent last years, due to the development of high-throughput technologies, high amounts of omics data are available as a powerful tool for the study of complex diseases progression [6]. The use of time-course omics data allows to construct interaction networks of biomarkers that change through the time, named dynamical network biomarkers (DNBs). DNBs can be used to identify the pre-disease stage of a complex disease and, therefore, try to discover the mechanisms of the disease progression in order to avoid the disease stage [7]. Taking into account that high-throughput data is usually noisy and contains few samples, the development of an efficient computational method is crucial for the identification of DNBs [8]. In the last years, different approaches have been proposed to manage the large-scale, high-dimensional, and dynamic properties of the biological data, like metaheuristics and evolutionary computation. More specifically, the metaheuristic and evolutionary algorithms have demonstrated their ability to solve a wide variety of bioinformatic problems such as the ones mentioned in [9,10], but very few have faced the DNB identification problem.

* Corresponding author.
E-mail addresses: vcoleto@unex.es (V. Coleto-Alcudia), mavega@unex.es (M.A. Vega-Rodríguez).

https://doi.org/10.1016/j.asoc.2021.107544
1568-4946/

In this article, the identification of DNBs is carried out as a multi-objective optimization problem. More specifically, the identification of the DNB that best reflects the pre-disease stage is treated as a three-objective optimization problem: a DNB has to be the smallest group of molecules that most correlates with the disease phenotype and best reflects the limit of the normal stage before the disease stage, that is, the pre-disease stage. The multi-objective optimization is carried out with an Artificial Bee Colony based on Dominance (ABCD) algorithm, an adaptation of the Artificial Bee Colony (ABC) algorithm. Before the optimization step, a pre-filtering step, only selecting the differentially expressed genes (DEGs) of the data, is developed to keep the genes that offer biological insight into the disease progression. In order to validate both statistically and biologically the proposed method, a leave-one-out cross-validation (LOOCV), a comparison with other different DNB identification methods, and a Gene Ontology enrichment of the identified DNBs are performed. Therefore, the novelty and main contributions of this work can be explained as follows:

- A new method for the identification of DNBs and pre-disease stage in complex diseases has been developed.
- This new method uses a Differential Gene Expression analysis for data pre-filtering in a first step and an ABCD algorithm for multi-objective optimization in a second step.
- Three objectives are optimized: size of the DNB, correlation with the disease phenotype, and earliest time-point with the strongest signal of the disease progression.
- Five time-course gene expression datasets of four complex diseases in different organisms have been used for the evaluation of the proposed method.
- The proposed method has been compared with five methods proposed by other authors in the scientific literature for the identification of DNBs and pre-disease stage in different complex diseases.
- The results of the proposed method have been validated statistically with a leave-one-out cross-validation and biologically with a Gene Ontology term enrichment analysis.

The rest of the article is organized as follows. A review of the scientific literature about DNBs identification is gathered in Section 2. Section 3 gives the definition and formulation about the problem we face in this article. Pre-filtering and optimization steps of the proposed method are described in Section 4. Section 5 collects the datasets used, the experimental settings, the results obtained, the comparisons with other methods proposed in the scientific literature, and the biological relevance and reliability of the results. Finally, Section 6 presents the conclusions and future work.

2. Scientific literature review

The use of dynamical network biomarkers (DNBs) for the identification of the pre-disease stage in complex diseases is recent (since 2012 [7]). In the last years, a number of different approaches for DNB identification have been developed and reported in the scientific literature to be applied to different diseases. The complexity of this problem makes metaheuristic methods a very suitable tool for its study [9,10]. Although metaheuristics field is very active (some recent surveys are [11–15]), very few efficient computational methods have been developed to solve the DNB identification problem [16] since its appearance in 2012 [7]. In this section, different methods proposed in the scientific literature in the recent years that maximize the network score (i.e. composite index) for searching an appropriate DNB in different complex diseases are reviewed.

In order to understand the dynamical organizations of molecules in complex diseases, several methods have been developed. One example is the use of an edge network, performed by X. Yu et al. in [17]. In contrast to conventional networks, in an edge network a node is a pair of molecules and an edge connects two pairs of molecules. This new concept was applied along with the use of second-order statistical information of a gene expression profile of subjects infected with influenza H3N2/Wisconsin. The combination allowed the identification of edge-biomarkers in order to study the prognosis of the influenza infection and to detect biomarkers of the disease. The edge-biomarkers formed a DNB and the time in which a determined DNB achieved the highest network score was the pre-disease stage. The prediction accuracy of a DNB was evaluated with a leave-one-out cross-validation (LOOCV).

In [18], T. Zeng et al. performed a framework named Progressive Module Network (PMN) with data of gene expression of type 1 diabetes mellitus in mice to identify the DNB that represented the pre-disease stage. They constructed tissue-specific and time-specific networks taking into account the biological interactions among molecules. Then, they used the PMN in order to detect the pre-disease biomarkers as the network that achieved the highest network score.

Y. Li et al. in [19], using the knowledge of protein–protein interactions (PPIs), constructed dynamical networks of PPI to identify DNBs. They used gene expression data of H3N2 and H1N1 influenza, acute lung injury, and type 2 diabetes mellitus for the identification of network modules with ClusterONE algorithm. As in previous methods, the DNB that effectively represented the pre-disease stage was the one that got the largest network score.

The use of multi-objective optimization in DNB identification did not appear until F. Vafaee applied it in [16], where NSGA-II algorithm [20] was used. The DNB identification for acute lung injury was treated in her method as a bi-objective optimization problem: maximizing the network score and minimizing the time in which the pre-disease stage occurred. Finally, she showed that the selected DNB was accurate by means of a three-stage analysis of the DNB. First, the DNB core was extended with PPIs. Then, that extension was analyzed in order to show that the extension could be target genes that regulated the genes of the DNB. Finally, she performed a gene ontology term enrichment analysis of the extended DNB.

One of the most recent methods in DNB identification is the one developed by A. D. Torshizi and L. Petzold in [21]. They developed two algorithms with time-course gene expression data of acute lung injury: the simulated annealing-based search algorithm and the pathway-induced dynamic biomarker discovery algorithm. In the first one, the DNB was identified by finding the highest network score in each step of time. In the second algorithm, the information of the biological pathways of the data was included, therefore, the DNB was biologically suitable and got the highest network score. In order to check if the DNB had biological relevance, they correlated the DNB and the phenotype of interest and carried out a gene ontology term enrichment analysis of the DNB.

3. DNB identification problem

The goal of the DNB identification from time-course high-throughput data is to determine the limit between the normal and disease stage in a disease development, named pre-disease stage. Consider a dataset D with the concentrations of a set of molecules over K samples (different subjects, with and without the disease, that is, case and control samples) at different time-points. The concentration of i molecule at t time-point is m_i^t, which can be seen as a vector with K elements (since we have K samples): $m_i^t = \langle m_{i,1}^t, m_{i,2}^t, \ldots, m_{i,K}^t \rangle$.

In [7], L. Chen et al. showed that a DNB is a group of molecules (DNB $\subseteq D$), which has to meet three properties at t time-point to reflect the pre-disease stage, that is, the critical transition between the normal and disease stage. First, the concentrations of these molecules have to drastically vary between the different samples (case and control samples), that is, they have a high standard deviation. Second, the concentrations of the molecules in the group are greatly correlated, that is, they have a high intra-cluster correlation coefficient. And third, the concentrations of these molecules are not greatly correlated with other molecules out of the group, that is, they have a low inter-cluster correlation coefficient. These three properties can be formulated as in Eqs. (1), (2), and (3):

1. Standard deviation of the DNB at t time-point, SD_{DNB}^t. In the pre-disease stage, the DNB shows an increase in its standard deviation:

$$SD_{DNB}^t = \frac{1}{\mid DNB \mid} \sum_{m_i^t \in DNB} \sigma_{m_i^t}, \tag{1}$$

where $\mid DNB \mid$ is the number of molecules in the DNB and $\sigma_{m_i^t}$ is the standard deviation of the different samples of the i molecule at t time-point.

2. Intra-cluster correlation of the DNB at t time-point, $PCintra_{DNB}^t$. The DNB presents an increase in the intra-cluster correlation coefficient in the pre-disease stage:

$$PCintra_{DNB}^t = \frac{1}{\mid DNB \mid \times (\mid DNB \mid -1)} \sum_{m_i^t, m_j^t \in DNB} \rho_{m_i^t, m_j^t}, \tag{2}$$

where $\rho_{m_i^t, m_j^t}$ is the Pearson correlation coefficient between two molecules (i and j) at t time-point.

3. Inter-cluster correlation of the DNB, at t time-point, with the rest of molecules that do not form the DNB, $PCinter_{DNB}^t$. The DNB shows a decrease in the inter-cluster correlation coefficient in the pre-disease stage:

$$PCinter_{DNB}^t = \frac{1}{\mid DNB \mid \times (\mid D \mid - \mid DNB \mid)} \sum_{m_i^t \in DNB, m_j^t \notin DNB} \rho_{m_i^t, m_j^t}, \tag{3}$$

where $\mid D \mid$ is the number of molecules in the entire dataset D and $\rho_{m_i^t, m_j^t}$ is the Pearson correlation coefficient between two molecules (i and j) at t time-point.

As can be seen in Eq. (4), these three properties can be combined into a network score, named composite index (CI), being CI_{DNB}^t the CI of the DNB at t time-point:

$$CI_{DNB}^t = \frac{SD_{DNB}^t \times PCintra_{DNB}^t}{PCinter_{DNB}^t}, \tag{4}$$

where t is the time-point when the DNB is identified.

The identification of DNBs from time-course high-throughput data as an early and strong signal of a disease development can be treated as a multi-objective optimization problem. In the DNB identification problem, a DNB has to be the smallest group of molecules that best reflects the pre-disease stage and best correlates with the phenotype of interest. Therefore, three objectives have to be optimized, as can be noted in Eq. (5).

$$F(DNB) = [F_1(DNB), F_2(DNB), F_3(DNB)], \tag{5}$$

where F is the optimization function and DNB is the solution, in this case, the group of molecules that form the DNB. F consists in three objective functions: F_1, F_2, and F_3.

The first objective tries to maximize the signal difference between the normal and disease stage, that is, it tries to find the time-point that best reflects the pre-disease stage. As said, the signal is quantified by the composite index, CI, for each time-point of the disease development (see Eq. (4)). Therefore, the first objective tries to maximize the difference of the composite index of a DNB ($difCI_{DNB}^t$) between a t time-point and the previous time-point ($t-1$), as can be observed in Eq. (6).

$$F_1(DNB) = difCI_{DNB}^t = CI_{DNB}^t - CI_{DNB}^{t-1}. \tag{6}$$

The second objective tries to maximize the correlation coefficient at t time-point, CC_{DNB}^t, between the expression of the DNB and the phenotype of interest. Therefore, the second objective corresponds to maximizing the intra-cluster correlation of the DNB, as Eq. (7) represents.

$$F_2(DNB) = CC_{DNB}^t = PCintra_{DNB}^t = \frac{1}{\mid DNB \mid \times (\mid DNB \mid -1)} \sum_{m_i^t, m_j^t \in DNB} \rho_{m_i^t, m_j^t}, \tag{7}$$

where $\rho_{m_i^t, m_j^t}$ is the Pearson correlation coefficient between two molecules (i and j) at t time-point.

Finally, the third objective tries to minimize the size of the group of molecules that form the DNB, $|DNB|$. In order to make the $F(DNB)$ optimization function easier, this third objective is turned into a function to maximize, as the first and second objectives. Eq. (8) shows the third objective of the DNB identification problem.

$$F_3(DNB) = \frac{1}{|DNB|}. \tag{8}$$

4. Proposed method for DNB identification

The proposed method consists in two steps. First, due to the high dimensionality of the time-course high-throughput data, the data is pre-filtered. Second, the identification of DNBs is performed as a multi-objective optimization using the Artificial Bee Colony based on Dominance (ABCD) algorithm.

4.1. Data pre-filtering

Data pre-filtering step goal is to remove the data that is not relevant in the problem, in this case, DNB identification. As can be seen in Fig. 1, the dimensionality of the data is reduced first, and then, the information that is not relevant is removed.

First, taking into account that a wide number of probe sets do not have a HUGO (Human Genome Organization [22]) gene symbol associated, those probe sets are removed from the dataset.

The second pre-filtering step is to save the unique genes for the following pre-filtering steps. That is, datasets usually contain multiple probes of the same gene, so they are averaged in order to keep a unique probe per gene in the dataset. Also, the number of samples has to be equal in the different conditions (time-points, in this case). Therefore, the least number of samples in a time-point is taken as reference value and when a time-point has a number of samples that exceeds this reference value, those samples are averaged.

The next pre-filtering step is to perform a differential expression analysis with the unique genes in order to identify the Differentially Expressed Genes (DEGs). DEGs are those genes whose expression levels change between two conditions in a statistically significant manner. In the DNB identification problem, the conditions that are compared to filter DEGs are each time-point. More specifically, a fold-change filtering is applied to select the genes that their expression is double or half in

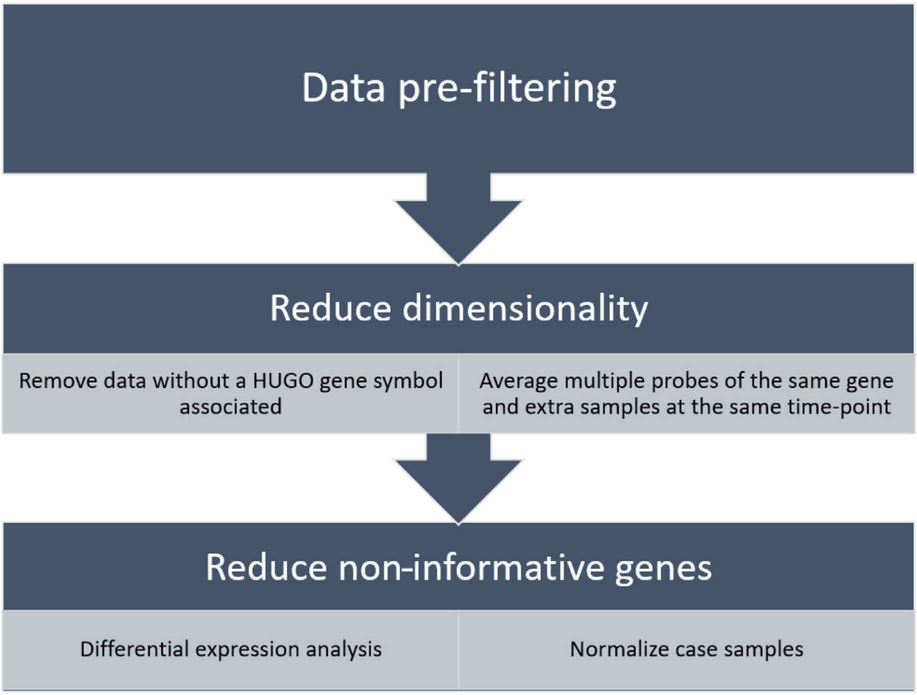
Data pre-filtering
Reduce dimensionality
Remove data without a HUGO gene symbol associated
Average multiple probes of the same gene and extra samples at the same time-point
Reduce non-informative genes
Differential expression analysis
Normalize case samples

Fig. 1. Data pre-filtering procedure.

the case samples compared to control samples. Therefore, unique genes whose *p*-value corrected with false discovery rate (FDR) estimation is higher than 0.01 or the fold-change is lower than 2 are filtered out, that is, those genes are not considered as DEGs.

The last pre-filtering step is the normalization of the case samples with the corresponding control samples to make a fair comparison of expression levels among all the time-points. For each time-point, the expression of each gene is subtracted from its mean expression and then divided by its standard deviation.

4.2. Artificial Bee Colony based on Dominance (ABCD) algorithm

The Artificial Bee Colony (ABC) algorithm was developed by D. Karaboga and B. Basturk in [23]. ABC algorithm is a metaheuristic method for function optimization that is motivated by the intelligent behavior of honey bees. ABC purpose is to find the best solutions in a way that is similar to the way that bees find their best food sources. In a hive, three types of bees exist and each type has a task: employed bees are the bees related to the known food sources or solutions in the ABC algorithm, onlooker bees receive information from the employed bees and choose the best food sources/solutions, and scout bees search randomly new food sources/solutions.

In this article, an adaptation of ABC algorithm, the Artificial Bee Colony based on Dominance (ABCD) algorithm, has been designed and implemented for DNB identification problem. Its choice has been motivated due to ABC has been used in many problems since its development, achieving very good results [24]. Also, the use of ABC is simple due to its reduced number of parameters, so it can be easily adapted to many contexts [25]. The ABCD algorithm is a multi-objective optimization algorithm that uses Pareto dominance between solutions, taking into account that the non-dominated solutions are the best solutions. More specifically, in the DNB identification problem, a solution is a variable-length list of genes (that is, each element of the list is a gene) that is considered as a potential DNB (see Section 3). ABCD pseudocode can be seen in Algorithm 1 and its flowchart in Fig. 2.

Algorithm 1: Pseudocode of ABCD algorithm.

```
Input  : GeneExpr (pre-filtered gene expression dataset), MaxDNBsize
         (maximum number of genes that can form the DNB),
         MinDNBsize (minimum number of genes that can form the DNB),
         MaxIterations (maximum number of iterations), DNBColony
         (colony size, number of DNBs), Tmax (limit for the scout bees)
Output: ParetoSols (different non-dominated solutions)
1  HalfColony ← DNBColony/2
2  solutions ← InitialSols(GeneExpr, HalfColony, MaxDNBsize, MinDNBsize)
3  ParetoSols ← ∅
4  for iteration ← 1 to MaxIterations do
5      EmployedBees(solutions, GeneExpr, HalfColony, MaxDNBsize,
         MinDNBsize)
6      RankCrowding(solutions)
7      CalculateProbs(solutions, HalfColony)
8      OnlookerBees(solutions, GeneExpr, HalfColony + 1, DNBColony,
         MaxDNBsize, MinDNBsize)
9      ScoutBees(solutions, GeneExpr, DNBColony, MaxDNBsize, MinDNBsize,
         Tmax)
10     RankCrowdingSort(solutions)
11     ParetoSols ← UpdateParetoSols(solutions, DNBColony)
12     solutions ← HalfBestSols(solutions, DNBColony)
13 return ParetoSols
```

The first step of ABCD algorithm is the initialization of the solutions. In ABCD, the half of the total colony (*DNBColony*) of solutions (*HalfColony*, Line 1) is associated with employed bees, and therefore, they are initialized in this step (Line 2). Each solution is randomly initialized with a different number of genes between the maximum and minimum values (*MaxDNBsize* and *MinDNBsize*, respectively). The different genes are randomly chosen from the dataset that is already pre-filtered (*GeneExpr*). Besides, the non-dominated solutions will be stored in *ParetoSols*, initialized as empty (Line 3).

The next step of ABCD algorithm is the search of the best solutions after their initialization. The search consists in a loop that is repeated until *MaxIterations* number of iterations is reached (Line 4). The best solutions are searched in each iteration by the management of employed bees, onlooker bees, and scout bees.

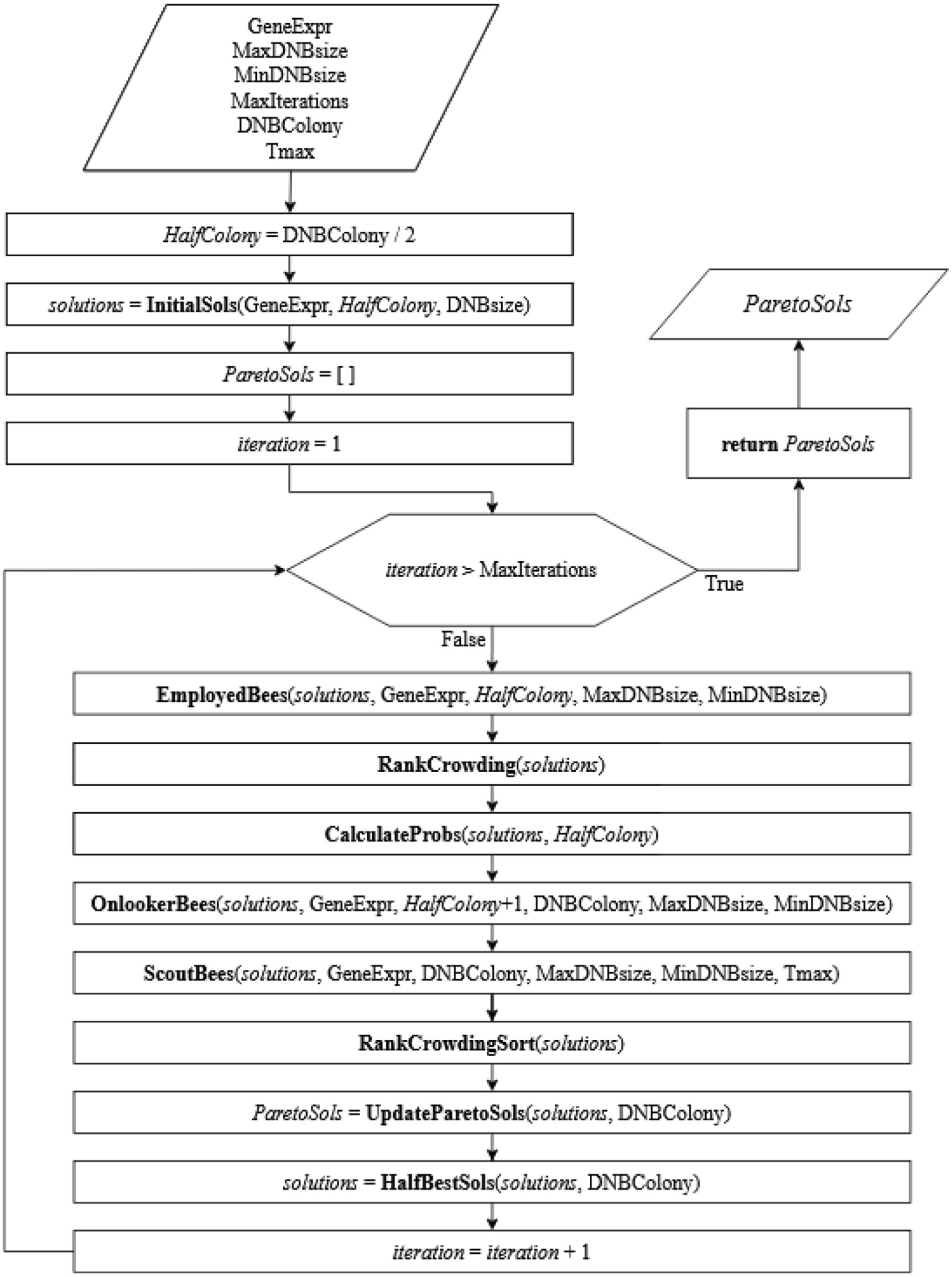
GeneExpr
MaxDNBsize
MinDNBsize
MaxIterations
DNBColony
Tmax
HalfColony = DNBColony / 2
solutions = InitialSols(GeneExpr, HalfColony, DNBsize)
ParetoSols = [ ]
iteration = 1
iteration > MaxIterations
True
False
return ParetoSols
ParetoSols
EmployedBees(solutions, GeneExpr, HalfColony, MaxDNBsize, MinDNBsize)
RankCrowding(solutions)
CalculateProbs(solutions, HalfColony)
OnlookerBees(solutions, GeneExpr, HalfColony+1, DNBColony, MaxDNBsize, MinDNBsize)
ScoutBees(solutions, GeneExpr, DNBColony, MaxDNBsize, MinDNBsize, Tmax)
RankCrowdingSort(solutions)
ParetoSols = UpdateParetoSols(solutions, DNBColony)
solutions = HalfBestSols(solutions, DNBColony)
iteration = iteration + 1

Fig. 2. Flowchart of ABCD algorithm for DNB identification.

Employed bees of ABCD algorithm (Line 5) are the first *HalfColony* solutions in the entire colony, that is, the *HalfColony* number of solutions randomly initialized in the initialization step. Each employed bee makes a search of a new solution using the mutation operator over the original solution (see Section 4.2.1). Then, using a strict dominance, the solution that dominates over the other is kept. That is, if the mutated solution dominates over the original solution, this is replaced by the mutated solution, otherwise, the original solution is kept.

The employed bees that are related to the best solutions have a higher probability to be selected by the onlooker bees. In order to know which employed bees are the best ones, the rank and crowding operators described in [20] are used (Line 6). Therefore, the solutions of the employed bees have a non-domination rank score and a crowding distance score.

Onlooker bees of ABCD algorithm are the second *HalfColony* solutions in the entire colony, that is, the *HalfColony*+1 to *DNBColony* number of solutions. Each onlooker bee chooses an employed bee as its current solution. The selection of employed bees is based on a $Prob(DNB)$ probability value (Line 7) that is associated to the quality of its solution and is calculated with the $RC(DNB)$ function that depends on its non-domination rank and crowding distance scores ($Rank(DNB)$ and $Crowding(DNB)$, respectively) (see Eq. (9)). As can be seen in Eq. (10), the final probability value $Prob(DNB)$ assures that all the solutions have at least a 10% probability of being chosen while the 90% probability depends on the quality of the solution ($RC(DNB)$). Once every onlooker bee has selected its employed bee (Line 8), a new solution is created with the mutation operator (see Section 4.2.1). Finally, a non-strict dominance is used for selecting between the mutated and current solution. More specifically, if the mutated solution is not dominated by the current solution, the mutated solution is selected, otherwise, the current solution is kept.

$$RC(DNB) = \frac{1}{Rank(DNB) + (\frac{1}{1 + Crowding(DNB)})} \tag{9}$$

$$Prob(DNB) = (0.9 \times RC(DNB)) + 0.1 \tag{10}$$

Scout bees of ABCD algorithm check all the *DNBColony* solutions, represented by both employed and onlooker bees (Line 9). If a solution has not been enhanced after *Tmax* number of trials, this solution is abandoned and a scout bee generates randomly a new solution that replaces the abandoned solution.

After that, all the solutions in the entire colony are sorted (Line 10) by quality (non-domination rank and crowding distance scores), that is, from the best to the worst quality.

Finally, the non-dominated solutions are saved and updated in *ParetoSols* (Line 11). Furthermore, the best *HalfColony* solutions will be the employed bees of the next iteration (Line 12).

4.2.1. Mutation operator

The mutation operator (Fig. 3 and Algorithm 2) is applied by employed and onlooker bees in order to create a *Mutated* solution from an *Original* solution. Three types of mutation are allowed: replacing a gene from the original solution by another different gene, adding a gene to the original solution, or removing a gene from the original solution. The choice of the mutation type depends on a random number (*R_Number*), the number of genes of the original solution (*DNBsize*), and if the dominance is strict or not (*Dominance*):

1. The original solution has the minimum number of genes (Line 2 in Algorithm 2). In this case, it is only possible to replace or to add a gene to the solution. If the mutation is carried out by employed bees, the dominance is strict (Line 3) and only replacing a gene is allowed. Otherwise, if the mutation is performed by the onlooker bees, the dominance is not strict (Line 5) and the choice between replacing or adding a gene depends on the random number.
2. The original solution has the maximum number of genes (Line 10). In this case, it is only possible to replace or to remove a gene from the solution. The choice between replacing or removing a gene depends on the *R_Number* random number.
3. The original solution has a size between the minimum and the maximum number of genes, both not included (Line 15). In this case, the three types of mutation are possible. The choice depends on the dominance type and on the random number.

5. Experimental settings and results

5.1. Time-course datasets

The proposed method has been evaluated with five time-course microarray datasets collected from the Gene Expression Omnibus (GEO) of the National Center for Biotechnology Information (NCBI) [26]. Table 1 contains information about these five datasets.

Algorithm 2: Pseudocode of the ABCD mutation operator.

```
   Input  :  GeneExpr (pre-filtered gene expression dataset), MinDNBsize
             (minimum number of genes that can form the DNB), MaxDNBsize
             (maximum number of genes that can form the DNB), Original
             (original solution), DNBsize (number of genes in the original
             solution), Dominance (dominance type, strict or non-strict)
   Output:   Mutated (new mutated solution)
 1 R_Number ← RandomNumber()
 2 if DNBsize == MinDNBsize then
 3     if Dominance == strict then
 4         Mutated ← Replacing(Original, GeneExpr)
 5     else
 6         if R_Number % 2 == 0 then
 7             Mutated ← Replacing(Original, GeneExpr)
 8         else
 9             Mutated ← Adding(Original, GeneExpr)
10 else if DNBsize == MaxDNBsize then
11     if R_Number % 2 == 0 then
12         Mutated ← Replacing(Original, GeneExpr)
13     else
14         Mutated ← Removing(Original)
15 else
16     if Dominance == strict then
17         D_Number = 2
18     else
19         D_Number = 3
20     switch R_Number % D_Number do
21         0: Mutated ← Replacing(Original, GeneExpr)
22         1: Mutated ← Removing(Original)
23         2: Mutated ← Adding(Original, GeneExpr)
24 return Mutated
```

5.2. Settings

The data pre-filtering step has been coded and implemented in R 3.6.1 in Windows 10 Pro. For differential expression analysis, the "limma" package of R is applied [27,28].

The ABCD algorithm has been coded and implemented in Python 3.8.1 in Windows 10 Pro. In order to use the best configuration of *DNBColony* and *MaxIterations* variables, a study to reach 3000 evaluations has been performed using 120 *DNBColony* × 25 *MaxIterations*, 60 *DNBColony* × 50 *MaxIterations*, and 30 *DNBColony* × 100 *MaxIterations* options. The option with a colony size (*DNBColony*) of 120 and 25 iterations (*MaxIterations*) gives the best results and it is used in the ABCD algorithm. The maximum number of trials for the scout bees is set in 10 ($Tmax = 10$). The minimum number of genes of a network is set in 2 ($MinDNBsize = 2$) and the maximum number of genes of a network (*MaxDNBsize*) is established taking into account the size of the DNB identified in other methods for each dataset.

The calculation of DNB accuracy in the dataset has been coded and implemented in R 3.6.1 in Windows 10 Pro. A SVM (Support Vector Machine) classifier is performed using the software LIBSVM [29] of "e1071" R package, Leave-One-Out Cross-Validation (LOOCV) technique, a linear kernel, and a C parameter with minimum and maximum values 2^{-5} and 2^{15} [30], respectively.

5.3. Results

5.3.1. Data pre-filtering results

In the proposed method, the first step consists in a data pre-filtering. First, in order to reduce the dimensionality of the datasets, those genes without HUGO gene symbol associated and duplicated genes are removed (see Section 4.1). Table 2 presents the dimensionality reduction in genes (G) and samples (S) of the datasets carried out in the first two steps of the data pre-filtering.

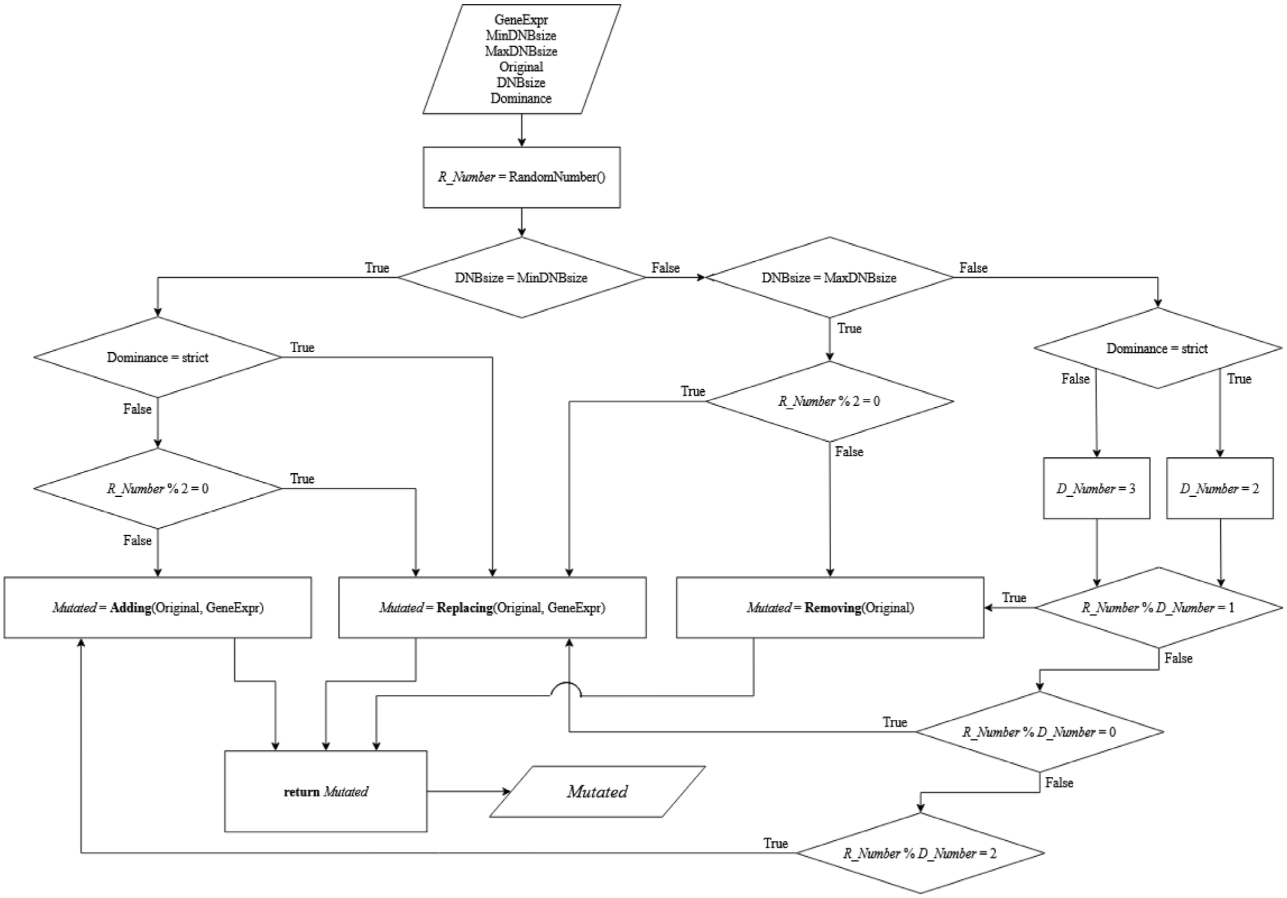

Fig. 3. Flowchart of the ABCD mutation operator for DNB identification.

Table 1
Information about the time-course microarray datasets.

Dataset	Description	Genes	Samples	Time-points
GSE2565	Gene expression changes in lung tissue of mice caused by the exposition of air (control groups) or carbonyl chloride, known as phosgene (case groups). The phosgene-induced lung injury progression is studied with the tissue collected of the mice at 0.5, 1, 4, 8, 12, 24, 48, and 72 h after the exposition.	22,690	104	0.5, 1, 4, 8, 12, 24, 48, and 72 h
GSE13268	Gene expression changes in adipose tissue of GotoKakizake (GK) and WistarKyoto (WK) rats with normal diet (ND — control groups) and high-fat diet (HF — case groups), causing type 2 diabetes (T2D) disease. T2D disease progression is studied with the tissue collected of GK-ND, GK-HF, WK-ND, and WK-HF rats at the age of 4, 8, 12, 16, and 20 weeks.	31,099	101	4, 8, 12, 16, and 20 weeks
GSE15150	Gene expression changes in pancreatic lymph nodes tissue of Non-Obese Diabetic (NOD) mice caused by type 1 diabetes (T1D) disease induction. T1D disease progression is studied with the tissue collected of mice at the age of 10 days (control group), and 4, 8, 12, 16, and 20 weeks (case groups) after the disease induction.	43,790	35	4, 8, 12, 16, and 20 weeks
GSE21884	Gene expression changes in spleen tissue of Non-Obese Diabetic (NOD) mice caused by type 1 diabetes (T1D) disease induction. T1D disease progression is studied with the tissue collected of mice at the age of 10 days (control group), and 4, 8, 12, 16, and 20 weeks (case groups) after the disease induction.	43,790	27	4, 8, 12, 16, and 20 weeks
GSE30550	Gene expression changes in blood of 17 healthy human subjects caused by the inoculation with live influenza (H3N2/Wisconsin) viruses. The disease progression is studied taking blood of the subjects 24 h and immediately prior to inoculation (control groups) and at 0, 5, 12, 21, 29, 36, 45, 53, 60, 69, 77, 84, 93, 101, and 108 h (case groups) after the inoculation.	11,961	268	0, 5, 12, 21, 29, 36, 45, 53, 60, 69, 77, 84, 93, 101, 108 h

After the dimensionality reduction of the datasets, the non-informative genes of the datasets are removed, in order to avoid the use of the genes that do not have a significant biological role in the processes and conditions of interest in the following steps (see Section 4.1). Table 3 shows the results of removing the non-informative genes in the data pre-filtering. In the Supplementary file, we present the collection of the DNB genes that are identified per dataset by the different authors in their respective

Table 2
Results obtained in the dimensionality reduction at the pre-filtering of the five datasets. The number of genes (G) and samples (S) are indicated at the beginning of the pre-filtering step (*Before pre-filtering*), after removing genes without a HUGO gene symbol associated (*Gene symbol associated*), and after removing by averaging the repeated genes and the extra samples found in any time-point (*Non repeated/extra*).

	GSE2565		GSE13268		GSE15150		GSE21884		GSE30550	
	G	S	G	S	G	S	G	S	G	S
Before pre-filtering	22,690	104	31,099	101	43,790	35	43,790	27	11,961	268
Gene symbol associated	22,147	104	15,225	101	21,608	35	21,608	27	11,548	268
Non repeated/extra	13,432	104	15,225	100	21,608	24	21,608	18	11,548	240

methods. In order to make a fair comparison among different methods, the DNB genes of other authors are included in the informative unique genes of this article. Therefore, the number of DEGs corresponds to the first DEGs until reaching 100 or 150 maximum number of informative unique genes, depending on the dataset. The last two datasets (GSE21884 and GSE30550) have less than 100 informative unique genes. The DEGs identified in the proposed method are also collected in the Supplementary file.

5.3.2. ABCD results

After the data pre-filtering step, only the significant data of each dataset are used by the ABCD algorithm for the DNB identification of the different diseases. Table 4 shows the characteristics of the five datasets used by ABCD after data pre-filtering. As can be seen, only the first time-points (always surpassing the 50% of all the available time-points) are used by ABCD due to in the last time-points the disease would be already developed.

Table 5 presents the final results of the proposed method per dataset after the ABCD step: difference of composite index between a time-point and the previous time-point (*difCI*), time-point when the pre-disease stage occurs (T), correlation coefficient with the disease phenotype (CC), number of genes of the DNB (*DNBsize*), and exact genes of the DNB (*DNB*). As can be seen, the identified DNBs are small but they show a strong signal (*difCI* between 0.549 and 1.111) in the earliest time points of the disease progression (T) and correlate well with the disease phenotype (CC between 0.397 and 0.930).

In order to analyze how the genes of each network are related per dataset over all the time-points and how the DNBs emerge in the identified pre-disease stage, the dynamical evolution of DNBs is represented in Figs. 4, 5, 6, 7, and 8. If the DNB genes are not correlated, no arrow is represented. As can be observed, throughout the different time-points the correlation changes among genes. However, in all datasets the time-point selected as pre-disease stage (T in Table 5) shows the earliest time-point with the greatest correlation strength, represented as dark red (if the correlation is positive) or dark blue (if the correlation is negative).

Moreover, a leave-one-out cross-validation (LOOCV) has been performed with all the samples in each dataset (see Table 1 and first rows in Table 6), in order to confirm that the genes of each DNB have the ability to classify the samples in early or later stages of the disease. In this way, the reliability of the proposed method is experimentally validated. As can be observed, the number of samples ranges from 27 (GSE21884) to 268 (GSE30550), ensuring reliable statistics. Table 6 shows the accuracy (*Accuracy* (%)), precision (*Precision* (%)), recall (*Recall* (%)), and F1 ($F1$ (%)) scores per dataset, and the mean of the same scores ($Accuracy_{avg}$ (%), $Precision_{avg}$ (%), $Recall_{avg}$ (%), and $F1_{avg}$ (%)) reached by the proposed method for the five datasets. In all the datasets, in general, the value of all the scores is high, achieving the 100% in three out of five datasets. In conclusion, the proposed method presents good results, with high averages of the scores (from 87.2% in accuracy to 91.3% in recall). Therefore, the identified DNBs are able to recognize the pre-disease stage, allowing a differentiation between previous and posterior stages of each disease. Thus, the proposed method is suitable for an early diagnosis of the diseases.

5.4. Comparison with other methods

In order to validate the proposed method, its results are compared with the results obtained by other methods (from other authors) for DNB identification. Table 7 shows this comparison. It is worth to know that the time-point for the pre-disease stage (T) and the number of genes in the identified DNB (*DNBsize*) have been obtained directly from the different articles, and the difference of composite index (*difCI*) and the correlation coefficient with the disease phenotype (CC) for the different methods have been calculated as in the proposed method in order to make fair comparisons. Table 7 includes two parts for every dataset.

The first part makes the comparison taking into account the three objectives to optimize (*difCI*, CC, and *DNBsize*). As can be observed, in general, the proposed method improves the three objectives in all the datasets regarding all the other methods. In a total of 8 comparisons (some datasets include comparisons with several methods), there are only 2 exceptions: (i) in dataset GSE2565 for *difCI* in [16], although the proposed method improves this method in the other two objectives (CC — correlation coefficient with the disease phenotype and *DNBsize* — number of genes required); and (ii) in dataset GSE30550 for CC in [17], although the proposed method improves this method in the other two objectives (*DNBsize* — number of genes required and *difCI* — difference of composite index, that is, difference of signal between the normal and disease stage).

The second part shows the time-point selected as pre-disease stage (T) by each method. In general, the proposed method detects the earliest time-point for the pre-disease stage, with only 1 exception out of 8 comparisons: in dataset GSE30550, the proposed method detects the same time-point as [19], which is exactly the subsequent time-point to the one detected by [17].

Regarding execution times, the proposed method cannot be compared with the other methods because execution times were not found in all the corresponding articles. However, taking into account the complexity of the problem, the execution time is on the order of minutes. In the case of the proposed method, the running time varies (depending on the dataset) between 8 and 60 min, when using a laptop with a CPU Intel i7-9750H at 2.60 GHz, 16 GB RAM, and Windows 10 Pro as operating system.

Therefore, it can be concluded that the proposed method, in general: (i) requires less genes (which facilitates the posterior biomedical analysis), (ii) improves the correlation coefficient with the disease phenotype, (iii) improves the difference of signal between the normal and disease stage, and (iv) detects the earliest time-point for the pre-disease stage (which is important in preventive medicine).

5.5. Biological relevance of the selected genes

DNBs identified for each dataset are the network that leads to the disease stage. In order to study the association between the DNB genes and the disease they are related to, a disease enrichment has been developed using Gene Ontology (GO) terms. Table 8 shows the biological processes, molecular functions, and cellular components enrichment per dataset performed with GOrilla tool [31].

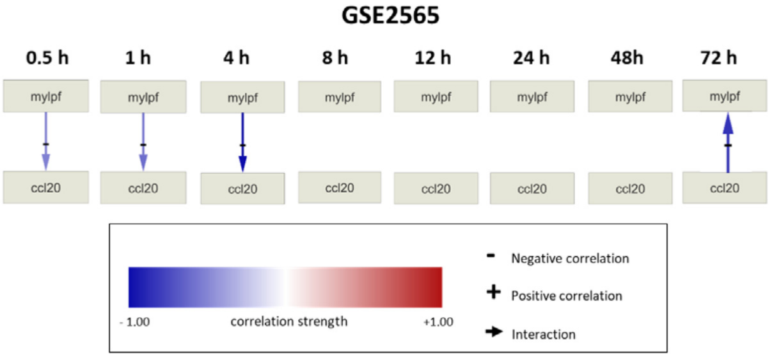

Fig. 4. Dynamical evolution of the DNB genes identified by the proposed method in GSE2565 dataset.

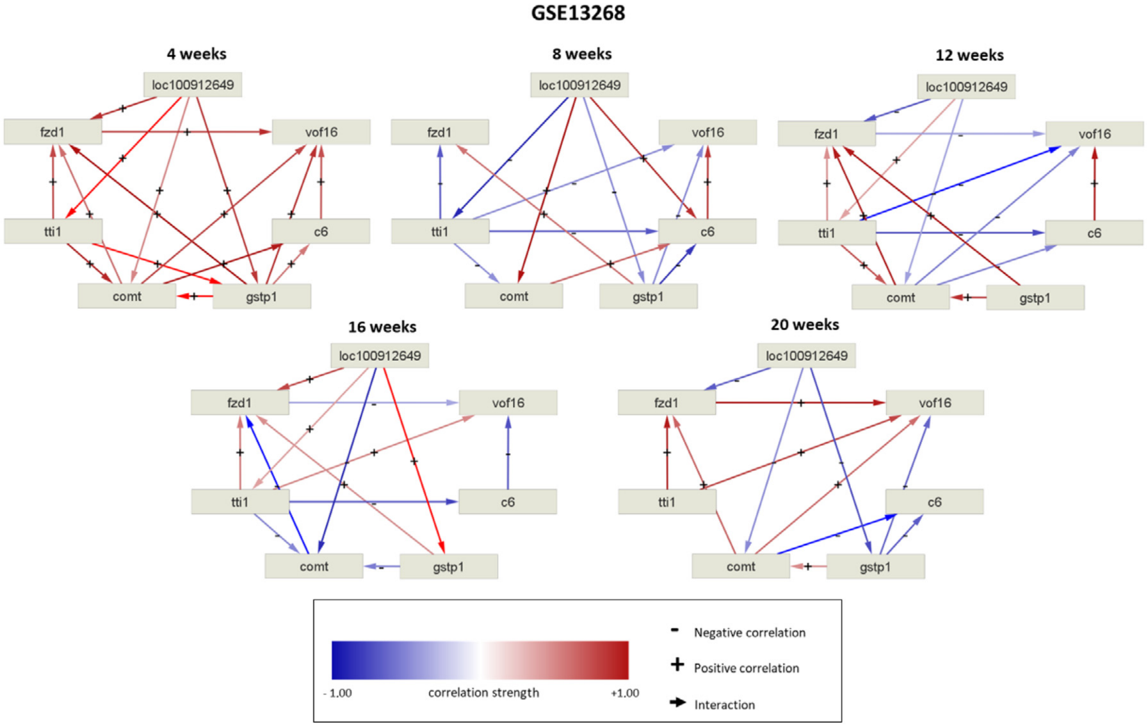

Fig. 5. Dynamical evolution of the DNB genes identified by the proposed method in GSE13268 dataset.

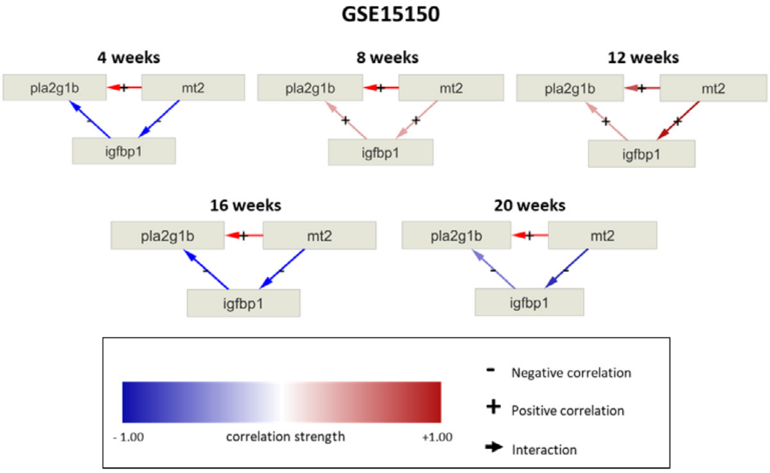

Fig. 6. Dynamical evolution of the DNB genes identified by the proposed method in GSE15150 dataset.

Table 3
Results obtained with the filtering of non-informative genes.

	GSE2565	GSE13268	GSE15150	GSE21884	GSE30550
Unique genes	13,432	15,225	21,608	21,608	11,548
Authors' DNB genes	96	37	7	6	28
DEGs	62	63	93	19	83
Informative unique genes	150	100	100	25	86

Table 4
Characteristics of the datasets used by ABCD for DNB identification.

		GSE2565	GSE13268	GSE15150	GSE21884	GSE30550
DNBsize	MinDNBsize	2	2	2	2	2
	MaxDNBsize	16	37	7	6	22
Time-points used		0.5, 1, 4, 8, 12 h	4, 8, 12 weeks	4, 8, 12 weeks	4, 8, 12 weeks	0, 5, 12, 21, 29, 36, 45, 53 h
Dimensionality	Genes	150	100	100	25	86
	Samples	30	15	12	6	120

Table 5
Results obtained with the proposed method.

Dataset	difCI	T	CC	DNBsize	DNB
GSE2565	0.549	4 h	0.683	2	mylpf, ccl20
GSE13268	0.804	4 weeks	0.397	7	loc100912649, tti1, gstp1, comt, c6, fzd1, vof16
GSE15150	1.111	4 weeks	0.835	3	pla2g1b, mt2, igfbp1
GSE21884	0.674	4 weeks	0.930	3	reg2, 4933400f21rik, c030011g24rik
GSE30550	0.621	45 h	0.866	3	tlr7, tnfaip6, rtp4

Table 6
Results generated with LOOCV, showing the accuracy, precision, recall, and F1 scores obtained by the proposed method.

	GSE2565	GSE13268	GSE15150	GSE21884	GSE30550
Samples	104	101	35	27	268
$Samples_{avg}$	**107**				
Accuracy (%)	62.5	100	100	100	73.3
$Accuracy_{avg}$ (%)	**87.2**				
Precision (%)	62.5	100	100	100	89.5
$Precision_{avg}$ (%)	**90.4**				
Recall (%)	100	100	100	100	56.7
$Recall_{avg}$ (%)	**91.3**				
$F1$ (%)	76.9	100	100	100	69.4
$F1_{avg}$ (%)	**89.3**				

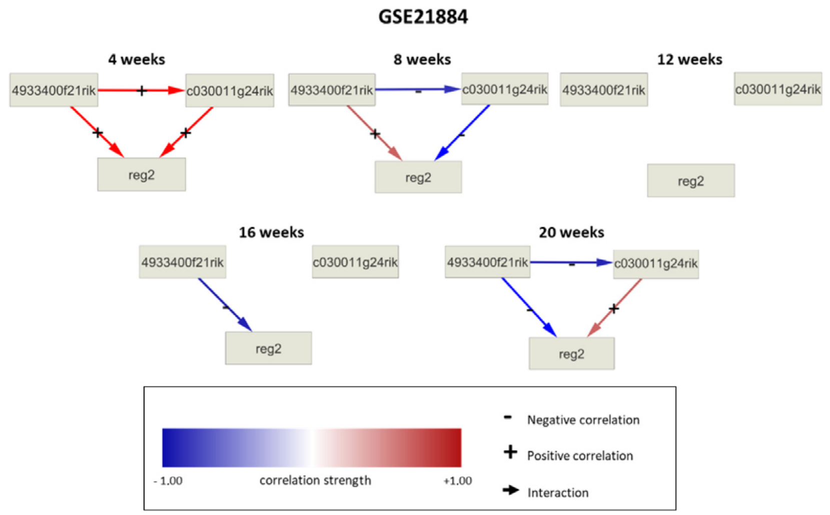

Fig. 7. Dynamical evolution of the DNB genes identified by the proposed method in GSE21884 dataset.

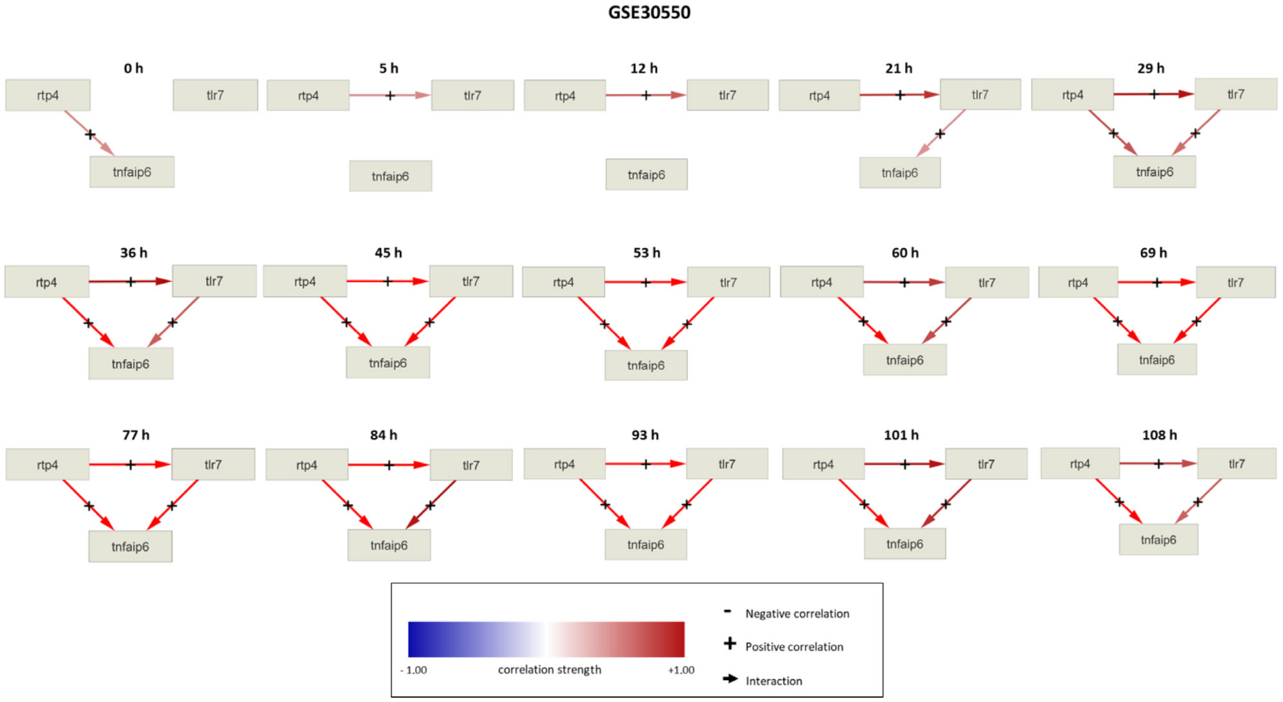

Fig. 8. Dynamical evolution of the DNB genes identified by the proposed method in GSE30550 dataset.

Table 7
Comparison of different methods proposed by different authors for DNB identification. The best results for the three objectives are highlighted in bold.

		Proposed method	Y. Li et al. [19]	T. Zeng et al. [18]	X. Yu et al. [17]	A. D. Torshizi et al. [21]	F. Vafaee [16]
GSE2565	difCI	0.549	0.288	–	–	0.303	**1.692**
	CC	**0.683**	0.313	–	–	0.289	0.605
	DNBsize	**2**	25	–	–	55	16
	....	....	....	....	....	....	....
	T	4 h	8 h	–	–	8 h	4 h
GSE13268	difCI	**0.804**	0.470	–	–	–	–
	CC	**0.397**	0.320	–	–	–	–
	DNBsize	**7**	37	–	–	–	–
	....	....	....	....	....	....	....
	T	4 w	8 w	–	–	–	–
GSE15150	difCI	**1.111**	–	0.215	–	–	–
	CC	**0.835**	–	0.378	–	–	–
	DNBsize	**3**	–	7	–	–	–
	....	....	....	....	....	....	....
	T	4 w	–	4 w	–	–	–
GSE21884	difCI	**0.674**	–	0.334	–	–	–
	CC	**0.930**	–	0.401	–	–	–
	DNBsize	**3**	–	6	–	–	–
	....	....	....	....	....	....	....
	T	4 w	–	4 w	–	–	–
GSE30550	difCI	**0.621**	0.338	–	0.304	–	–
	CC	0.866	0.857	–	**0.890**	–	–
	DNBsize	**3**	22	–	22	–	–
	....	....	....	....	....	....	....
	T	45 h	45 h	–	36 h	–	–

6. Conclusions and future work

In this article, a new method for DNB (Dynamical Network Biomarker) identification in complex diseases has been developed using multi-objective optimization. The identification of a DNB allows to detect the pre-disease stage and, therefore, to know the point of reversibility of a disease. Here, a DNB has been identified as the smallest gene network that shows the strongest signal in the earliest time-point of the disease progression and best correlates with the disease phenotype. The proposed method has been divided into two steps. In the first step, a pre-filtering of the data has been performed in order to keep only the most relevant data. The pre-filtering has been carried out with a differential expression analysis after the dimensionality reduction of the data.

Table 8
GO enrichment found in the five datasets with a *p*-value threshold of 10E-3.

Dataset	GO enrichment	GO term	Description	p-value	Genes
GSE2565	Process	GO:0072679	Thymocyte migration	2.25E−4	ccl20
		GO:0032730	Positive regulation of interleukin-1 alpha production	7.89E−4	ccl20
	Function				
	Component				
GSE13268	Process	GO:0031668	Cellular response to extracellular stimulus	7.14E−4	comt, fzd1, gstp1
		GO:0048662	Negative regulation of smooth muscle cell proliferation	9.96E−4	comt, gstp1
	Function				
	Component				
GSE15150	Process	GO:1904633	Regulation of glomerular visceral epithelial cell apoptotic process	5.07E−4	pla2g1b
		GO:1904635	Positive regulation of glomerular visceral epithelial cell apoptotic process	5.07E−4	pla2g1b
	Function				
	Component				
GSE21884	Process	GO:0044278	Cell wall disruption in other organism	3.38E−4	reg2
		GO:0001967	Suckling behavior	9.02E−4	reg2
	Function	GO:0042834	Peptidoglycan binding	7.33E−4	reg2
		GO:0070492	Oligosaccharide binding	9.58E−4	reg2
	Component				
GSE30550	Process	GO:0002252	Immune effector process	2.85E−4	tlr7, tnfaip6, rtp4
		GO:0006952	Defense response	4.31E−4	tlr7, tnfaip6, rtp4
		GO:0034154	Toll-like receptor 7 signaling pathway	4.88E−4	tlr7
		GO:0051607	Defense response to virus	5.01E−4	tlr7, rtp4
	Function	GO:0031849	Olfactory receptor binding	4.88E−4	rtp4
	Component				

In the second step, the ABCD (Artificial Bee Colony based on Dominance) algorithm has been used due to the DNB identification has been treated as a multi-objective optimization problem.

The proposed method has been evaluated with five time-course gene expression microarray datasets of four different complex diseases in different organisms. Results show that the proposed method identifies small DNBs that are able to reach high signals when pre-disease stage occurs and correlate well with the disease phenotype. Furthermore, the proposed method detects the pre-disease stage in a very early time-point, which is important in preventive medicine. Moreover, the results of the proposed method are compared with results of other different methods (from different authors) for DNB identification. The comparison indicates the effectiveness of the proposed method due to its results are, as a whole, better than the results obtained by the other methods. Finally, a LOOCV (Leave-One-Out Cross-Validation) and a GO (Gene Ontology) term enrichment have been performed to validate the method, showing the good accuracy of the proposed method and the good relation of the DNB genes with the disease they are linked to.

In summary, the proposed method has shown that can identify with effectiveness DNBs as disease initiation signal. The proposed method is a good tool that could be applied in preventive medicine, in order to facilitate the disease prevention and to track the disease progression. More specifically, due to the complexity of some diseases like cancer, Alzheimer, diabetes, etc. (called complex diseases), DNB identification is very valuable to understand the progression of these complex diseases. Beyond biomedicine, the proposed method could be applied in other different biological issues that are accompanied by time-course high-throughput data.

CRediT authorship contribution statement

Veredas Coleto-Alcudia: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data curation, Writing - original draft, Visualization. **Miguel A. Vega-Rodríguez:** Conceptualization, Methodology, Formal analysis, Resources, Writing - review & editing, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was partially funded by Junta de Extremadura (Spain) and the ERDF (European Regional Development Fund, EU), grant GR18090 provided to the research group TIC015. This work was also partially funded by the MCIU (Ministry of Science, Innovation and Universities, Spain), the AEI (State Research Agency, Spain), and the ERDF (European Regional Development Fund, EU), under the contract PID2019-107299GB-I00/AEI/10.13039/501100011033 (Multi-HPC-Bio project).

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.asoc.2021.107544.

References

[1] X. Liu, X. Chang, R. Liu, X. Yu, L. Chen, K. Aihara, Quantifying critical states of complex diseases using single-sample dynamic network biomarkers, PLOS Comput. Biol. 13 (7) (2017) 1–21, http://dx.doi.org/10.1371/journal.pcbi.1005633.

[2] H. Kilpinen, J.C. Barrett, How next-generation sequencing is transforming complex disease genetics, Trends Genet. 29 (1) (2013) 23–30, http://dx.doi.org/10.1016/j.tig.2012.10.001.

[3] Z.-P. Liu, Identifying network-based biomarkers of complex diseases from high-throughput data, Biomark. Med. 10 (6) (2016) 633–650, http://dx.doi.org/10.2217/bmm-2015-0035.

[4] T. Zeng, S.-y. Sun, Y. Wang, H. Zhu, L. Chen, Network biomarkers reveal dysfunctional gene regulations during disease progression, FEBS J. 280 (22) (2013) 5682–5695, http://dx.doi.org/10.1111/febs.12536.

[5] J.X. Hu, C.E. Thomas, S. Brunak, Network biology concepts in complex disease comorbidities, Nature Rev. Genet. 17 (10) (2016) 615–629, http://dx.doi.org/10.1038/nrg.2016.87.

[6] G. Zhu, X.-M. Zhao, J. Wu, A survey on biomarker identification based on molecular networks, Quant. Biol. 4 (2016) 310–319, http://dx.doi.org/10.1007/s40484-016-0084-z.

[7] L. Chen, R. Liu, Z.-P. Liu, M. Li, K. Aihara, Detecting early-warning signals for sudden deterioration of complex diseases by dynamical network biomarkers, Sci. Rep. 2 (342) (2012) 1–8, http://dx.doi.org/10.1038/srep00342.

[8] R. Liu, M. Li, Z.-P. Liu, J. Wu, L. Chen, K. Aihara, Identifying critical transitions and their leading biomolecular networks in complex diseases, Sci. Rep. 2 (813) (2012) 1–9, http://dx.doi.org/10.1038/srep00813.

[9] Z. Chelly Dagdia, P. Avdeyev, M.S. Bayzid, Biological computation and computational biology: survey, challenges, and discussion, Artif. Intell. Rev. (2021) 1–67., http://dx.doi.org/10.1007/s10462-020-09951-1.

[10] S. Cheng, L. Ma, H. Lu, X. Lei, Y. Shi, Evolutionary computation for solving search-based data analytics problems, Artif. Intell. Rev. 54 (2021) 1321–1348, http://dx.doi.org/10.1007/s10462-020-09882-x.

[11] H. Rajabi Moshtaghi, A. Toloie Eshlaghy, M.R. Motadel, A comprehensive review on meta-heuristic algorithms and their classification with novel approach, J. Appl. Res. Ind. Eng. (2021) http://dx.doi.org/10.22105/jarie.2021.238926.1180.

[12] N. Khanduja, B. Bhushan, Recent advances and application of metaheuristic algorithms: A survey (2014–2020), in: Metaheuristic and Evolutionary Computation: Algorithms and Applications, Springer, Singapore, 2021, pp. 207–228, http://dx.doi.org/10.1007/978-981-15-7571-6_10.

[13] R. Kaleche, Z. Bendaoud, K. Bouamrane, Bio-inspired metaheuristics: A comprehensive survey, Int. J. Organ. Collect. Intell. (IJOCI) 10 (4) (2020) 1–18, http://dx.doi.org/10.4018/IJOCI.2020100101.

[14] D.K. Sarmah, A.J. Kulkarni, A. Abraham, Heuristics and metaheuristic optimization algorithms, in: Optimization Models in Steganography using Metaheuristics, Springer International Publishing, Cham, 2020, pp. 49–61, http://dx.doi.org/10.1007/978-3-030-42044-4_3.

[15] K. Hussain, M.N. Mohd Salleh, S. Cheng, Y. Shi, Metaheuristic research: A comprehensive survey, Artif. Intell. Rev. 52 (2019) 2191–2233, http://dx.doi.org/10.1007/s10462-017-9605-z.

[16] F. Vafaee, Using multi-objective optimization to identify dynamical network biomarkers as early-warning signals of complex diseases, Sci. Rep. 6 (22023) (2016) 1–12, http://dx.doi.org/10.1038/srep22023.

[17] X. Yu, G. Li, L. Chen, Prediction and early diagnosis of complex diseases by edge-network, Bioinformatics 30 (6) (2013) 852–859, http://dx.doi.org/10.1093/bioinformatics/btt620.

[18] T. Zeng, C.-c. Zhang, W. Zhang, R. Liu, J. Liu, L. Chen, Deciphering early development of complex diseases by progressive module network, Methods 67 (3) (2014) 334–343, http://dx.doi.org/10.1016/j.ymeth.2014.01.021.

[19] Y. Li, S. Jin, L. Lei, Z. Pan, X. Zou, Deciphering deterioration mechanisms of complex diseases based on the construction of dynamic networks and systems analysis, Sci. Rep. 5 (9283) (2015) 1–11, http://dx.doi.org/10.1038/srep09283.

[20] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2) (2002) 182–197, http://dx.doi.org/10.1109/4235.996017.

[21] A.D. Torshizi, L. Petzold, Sparse pathway-induced dynamic network biomarker discovery for early warning signal detection in complex diseases, IEEE/ACM Trans. Comput. Biol. Bioinform. 15 (3) (2018) 1028–1034, http://dx.doi.org/10.1109/TCBB.2017.2687925.

[22] HUGO, Human Genome Organisation, 2021, http://www.hugo-international.org/, Last accessed: 25-April-2021.

[23] D. Karaboga, B. Basturk, A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm, J. Global Optim. 39 (2007) 459–471, http://dx.doi.org/10.1007/s10898-007-9149-x.

[24] D. Karaboga, B. Gorkemli, C. Ozturk, N. Karaboga, A comprehensive survey: Artificial bee colony (ABC) algorithm and applications, Artif. Intell. Rev. 42 (2014) 21–57, http://dx.doi.org/10.1007/s10462-012-9328-0.

[25] Y. Liu, L. Ma, G. Yang, A survey of artificial bee colony algorithm, in: 2017 IEEE 7th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER), 2017, pp. 1510–1515, http://dx.doi.org/10.1109/CYBER.2017.8446301.

[26] GEO, national center for Biotechnology information (NCBI), 2021, https://www.ncbi.nlm.nih.gov/geo/, Last accessed: 25-April-2021.

[27] M.E. Ritchie, B. Phipson, D. Wu, Y. Hu, C.W. Law, W. Shi, G.K. Smyth, *Limma* powers differential expression analyses for RNA-sequencing and microarray studies, Nucleic Acids Res. 43 (7) (2015) e47, http://dx.doi.org/10.1093/nar/gkv007.

[28] B. Phipson, S. Lee, I.J. Majewski, W.S. Alexander, G.K. Smyth, Robust hyperparameter estimation protects against hypervariable genes and improves power to detect differential expression, Ann. Appl. Stat. 10 (2) (2016) 946–963, http://dx.doi.org/10.1214/16-AOAS920.

[29] C.-C. Chang, C.-J. Lin, LIBSVM: a library for support vector machines, ACM Trans. Intell. Syst. Technol. 2 (3) (2011) 1–27, http://dx.doi.org/10.1145/1961189.1961199.

[30] C.-W. Hsu, C.-C. Chang, C.-J. Lin, A practical guide to support vector classification, Technical Report, National Taiwan University, Taipei, Taiwan, 2016, pp. 1–16.

[31] E. Eden, R. Navon, I. Steinfeld, D. Lipson, Z. Yakhini, Gorilla: A tool for discovery and visualization of enriched GO terms in ranked gene lists, BMC Bioinformatics 10 (48) (2009) 1–7, http://dx.doi.org/10.1186/1471-2105-10-48.