Computers in Biology and Medicine 104 (2019) 149–162

Contents lists available at ScienceDirect

Computers in Biology and Medicine

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

Genetic algorithm for assigning weights to gene expressions using functional annotations

Shubhra Sankar Ray, Sampa Misra*

Machine Intelligence Unit, Indian Statistical Institute, 203 B.T. Road, Kolkata, 700108, India

ARTICLE INFO

Keywords:
Gene expression
Weighting
Saccharomyces cerevisiae
Genetic algorithm
Gene annotation
Computational biology

ABSTRACT

A method, named *genetic algorithm for assigning weights to gene expressions using functional annotations* (GAAWGEFA), is developed to assign proper weights to the gene expressions at each time point. The weights are estimated using functional annotations of the genes in a genetic algorithm framework. The method shows gene similarity in an improved manner as compared with other existing methods because it takes advantage of the existing functional annotations of the genes. The weight combination for the expressions at different time points is determined by maximizing the fitness function of GAAWGEFA in terms of the positive predictive value (PPV) for the top 10,000 gene pairs. The performance of the proposed method is primarily compared with Biweight mid correlation (BICOR) and original expression values for the six *Saccharomyces cerevisiae* datasets and one *Bacillus subtilis* dataset. The utility of GAAWGEFA is shown in predicting the functions of 48 unclassified genes (using *p*-value cutoff 10^{-13}) from *Saccharomyces cerevisiae* microarray data where the expressions are weighted using GAAWGEFA and are clustered using *k*-medoids algorithm. The related code along with various parameters is available at http://sampa.droppages.com/GAAWGEFA.html.

1. Introduction

The process by which coded information of a gene is converted into the structures present and operating in the cell is called gene expression [1]. The analysis and interpretation of the gene expression data is one of the major challenges in bioinformatics [2,3]. In general, the gene expression data can be obtained from microarray experiments [4]. A typical gene expression microarray contains tens of thousands of probes tethered onto a solid surface [5] where each probe corresponding to the fragment of a gene is employed to infer the expression level of the target gene [6,7]. The gene expression data are usually represented as an expression matrix where each column represents all the gene expression levels from a single experiment and each row represents the expression of a gene across all the experiments. Generally, it is considered that the genes with similar expression profiles are functionally related. In many cases, accurate similarity between two gene expressions may not be obtained properly using the raw data as they may contain noise, missing values, and technical or experimental errors [8–10]. Therefore, it is important to preprocess the raw time series gene expression data prior to any analysis [11]. The goal of preprocessing is to reduce the technical or experimental errors to find out appropriate functional similarity between two expression profiles and thereby inferring the underlying biological process. However, discretization [12,13] and weighting expressions [14,15] are two of the important methods for preprocessing the data. Herein, we propose a method for weighting the expressions using functional annotations so that the expression similarity among the genes can truly reflect their functional similarity. The weight for each of the expressions (time points) is determined via genetic algorithm (GA) [16]. The positive predictive value (PPV), computed using the functional annotations of the genes, for the top 10,000 gene pairs is considered as the fitness function of the chromosome (weight combination) in GA.

A key step in the analysis of the gene expression data is the identification of the groups of genes that manifest similar expression patterns [17]. If two expression profiles are similar, one can hypothesize that the respective genes are functionally related and they can be helpful in gene function prediction [18,19], gene regulatory network analysis [20–23], functional module prediction [24,25], and disease prediction [26,27]. There exist a number of measures to find out the expression similarity between the genes, namely, Pearson correlation (PC) [10,28], Euclidean distance (ED) [29], Manhattan distance/City-block distance (MD) [9], and Spearman rank correlation (SRC) [30]. The Pearson correlation [31] is one of the widely used measures to calculate the similarity between two gene expression profiles [32]. Let us consider the expression values of the two genes A_i and A_j as $A_i = \{a_{i1}, \cdots, a_{in}\}$ and $A_j = \{a_{j1}, \cdots, a_{jn}\}$, where n is the total number of

* Corresponding author.
E-mail addresses: shubhra@isical.ac.in (S.S. Ray), sampamisra1989@gmail.com (S. Misra).

https://doi.org/10.1016/j.compbiomed.2018.11.011
Received 31 July 2018; Received in revised form 13 November 2018; Accepted 13 November 2018

expression values for both the genes. It is defined as

$$P(A_i, A_j) = \frac{\frac{1}{n}\sum_{k=1}^{n}(A_{i_k} - \overline{A_i})(A_{j_k} - \overline{A_j})}{\sigma_{A_i}\sigma_{A_j}}, \quad (1)$$

where A_{i_k} and A_{j_k} are the expression values of the gene A_i and the gene A_j at the k^{th} time point, respectively. Whereas, σ_{A_i} & σ_{A_j} are the standard deviation and $\overline{A_i}$ & $\overline{A_j}$ are the mean of the respective genes.

Additionally, it is worth mentioning about some commonly used weighted expression similarity based methods. Zhou et al. in Ref. [15] proposed a method called weighted correlation (WC) where the weights are assigned to all the time points under each experiment type (e,g., cell cycle) instead of separate weights to individual time points. In this method, the number of expressions in each experiment type decide the value of weight. For example, if a dataset consists of 10 expressions and out of them 5 expressions belong to cell cycle, 3 expressions belong to sporulation, and 2 expressions belong to diauxic shift experiments. Then, each of the 5 cell cycle samples, 3 sporulation samples, and 2 diauxic shift samples are assigned a weight of 1/5, 1/3, and 1/2, respectively. Hence, the weighted correlation (WC) between two genes A_i and A_j with a weighting vector w is defined as

$$WC(A_i, A_j) = \frac{\sum_{k=1}^{n} w_k A_{i_k} A_{j_k} - (1/\sum w_k)\sum w_i A_{i_k} \sum w_k A_{j_k}}{\sqrt{\sum w_k A_{i_k}^2 - (1/\sum w_k)\sum (w_k A_{i_k})^2}} \times \frac{1}{\sqrt{\sum w_k A_{j_k}^2 - (1/\sum w_k)\sum (w_k A_{j_k})^2}}. \quad (2)$$

The biweight midcorrelation (BICOR) [14] is also a weighted similarity based method where the weights are assigned to each expression using median absolute deviation. It is defined as

$$BICOR(A_i, A_j) = \frac{\sum_{k=1}^{n}(A_{i_k} - med(A_i))w_k^{(A_i)}}{\sqrt{\sum_{k=1}^{n}[A_{i_k} - med)w_k^{(A_i)}](A_i)^2}} \times \frac{\sum_{k=1}^{n}(A_{j_k} - med(A_j))w_k^{(A_j)}}{\sqrt{\sum_{k=1}^{n}[A_{j_k} - med)w_k^{(A_i)}](A_j)^2}}, \quad (3)$$

where $w_k^{(A_i)} = (1 - u_k^2)^2 I(1 - |u_k|)$ and $u_k = \frac{A_{i_k} - med(A_i)}{9MAD(A_i)}$.

Here, $med(A_i)$ and $MAD(A_i)$ are the median and median absolute deviation of A_i, respectively. If $1 - |u_k| > 0$ then $I(1 - |u_k|)$ becomes 1, otherwise it will be 0.

However, these methods cannot take advantage of the existing functional annotations available for many genes. Furthermore, the gene expression data are generally noisy and are not very reliable because of several factors affecting the measurements during experiments [33]. In this regard, incorporation of biological knowledge (gene annotations) to the expression profile to find out accurate similarity between two genes will be helpful to surpass this limitation to some extent.

The relationship between gene annotations with gene expression as well as some similarity measures based on gene annotations are discussed in Ref. [34]. In our previous report [35], a weighted Pearson correlation (WPC) method is designed where the weights are assigned to experiment types rather than each expression (time point) using functional annotations. The method developed in Ref. [35] is applicable when multiple types of experiments are available for the same set of genes and all the expressions within a particular experiment type are assigned the same weight. However, the importance of individual time point is not considered in that investigation [35]. It is worth noting that, the methods discussed earlier [15,35] are useful and are applicable when multiple types of experimental conditions like cell cycle, sporulation, and diauxic shift are present in the same dataset and the weights are assigned to a group of expressions belonging to same experimental condition. Under this situation, to overcome the limitation of the existing methods like WPC [35] and WC [15], which cannot assign proper weights to each of the expressions under same experimental condition, we propose *genetic algorithm for assigning weights to gene expressions using functional annotations* (GAAWGEFA) to reduce the technical or experimental errors in each time point. Over decades, researchers have suggested many algorithms, namely, genetic algorithm (GA) [16], tabu search [36], particle swarm optimization [37], and ant colonies for optimization [38]. In the present study, the popular optimization method, GA, is employed due to its effectiveness in finding near optimal solutions in short computational time for combinatorial optimization problems [39]. In GA, each chromosome is created by randomly selecting n numbers (genes in chromosome of GA) having the values within 0–1 as weight combination for the time points, where n is the number of time points in the gene expression dataset. The weight combination for different time points is determined by maximizing the fitness function of the chromosome in GAAWGEFA in terms of the positive predictive value (PPV), computed using the functional annotations of the genes, for the top 10,000 gene pairs. All the expressions under each time point in the dataset are then multiplied with the weight combination for different time points to obtain the weighted gene expression data. The detail descriptions related to PPV are provided in the Section 2.2. The utility of GAAWGEFA is shown in predicting the function of unknown genes in *Saccharomyces cerevisiae*.

The proposed GAAWGEFA deals with functional time series gene expression data where individual expressions in time series for various experiments (like Cell Cycle and Diauxic Shift) have no label information and hence they cannot be considered as features to predict the class/function of a gene. In functional gene expression data, a gene's label/function can only be predicted from its expression profile similarity with other genes (through clustering mechanism and functional enrichment by observing cluster related p value), but not by observing its expression values at various time points as features. On the other hand, the proposed GAAWGEFA is not applicable for the sample based data (e.g., diseased gene expression data involving normal and cancer patients) where the samples are the features and are used for classifying/discriminating the normal and the diseased genes/patients. Because the working principle for GAAWGEFA is not based on classifying/labeling the genes based on sample (feature) values. Further, the sample based gene expression data (say normal vs. cancer) can only be handled using classification based methodologies where the objective is feature selection (to select a subset of features) and thereby ranking the genes according to their relevance to the disease or improving the classification accuracy in discriminating the normal and the diseased genes/patients.

The rest of the article is organized as follows: brief description of the datasets, the PPV computation method, details of GA, and the proposed methodology are provided in Section 2. The experimental results for various datasets, cross validation, statistical analysis, function prediction of unknown genes, and the biological relevance of the predicted functions are discussed in Section 3. The performance of our method as compared with other existing methods are discussed in Section 4. Finally, Section 5 concludes this investigation.

2. Methods

In this section, brief description of the datasets, the PPV computation method, basics of genetic algorithm, and various steps involved in the proposed GAAWGEFA are provided.

2.1. Dataset description

We used seven datasets, namely, All Yeast [40], Cell Cycle All Yeast (CCAY) [40], Diauxic Shift All Yeast (DSAY) [40], Sporulation All Yeast (SAY) [40], Yeast Complex [41], Cell Cycle [4], and Bacillus [4] datasets in this investigation. Out of these seven datasets, first six are

Table 1
Summary for different gene expression data sets.

Dataset	All Yeast	CCAY	DSAY	SAY	Yeast Complex	Cell Cycle	Bacillus
Organism	*Saccharomyces cerevisiae*	*Saccharomyces cerevisiae*	*Saccharomyces cerevisiae*	*Saccharomyces cerevisiae*	*Saccharomyces cerevisiae*	*Saccharomyces cerevisiae*	*Bacillus subtilis*
No. of genes	6072	6072	6072	6072	979	634	1055
Total number of time points	80	60	7	13	79	184	81

Saccharomyces cerevisiae datasets obtained from microarray experiments and the seventh is the *Bacillus subtilis* dataset obtained from microarray experiment. The total number of time point for All Yeast, CCAY, DSAY, SAY, Yeast Complex, Cell Cycle, and Bacillus are 80, 60, 7, 13, 79, 184, and 81, respectively. Table 1 shows the name of the datasets, the number of genes in each dataset, organism, and total number of expressions for a particular dataset. The method, described in LSimpute [42], is employed to predict the missing gene expression values for each dataset.

2.2. Computing PPV

The method of computing the positive predictive value (PPV) at a particular similarity value of the gene pairs is described in this section. The proportion of true positive (TP) gene pairs at various similarity values (computed from a data set) can be used to compute the PPV. The TP gene pair is defined as a pair of genes having an overlapping GO (Gene Ontology) term annotations. For a particular similarity value, the positive predictive value (proportion of TP pair) can be defined as [19]

$$\text{PPV} = \frac{\text{Number of pairs with common GO term}}{\text{Total number of pairs}}. \quad (4)$$

The higher PPV value implies higher functional similarity between the gene-pairs predicted by a similarity measure. In addition, the PPV can be interpreted as being proportional to the accuracy of the datasets and their ability to predict the cellular/biological processes involved in a given similarity value. In the present study, the common GO term for the *Saccharomyces cerevisiae* dataset is computed using Yeast GO-Slim process (SGD) annotations [43]. The Munich Information for Protein Sequences (MIPS) annotations [44] are used for *Bacillus subtilis* dataset.

We have chosen yeast GO-Slim process annotations because our ultimate aim is to predict the gene functions, not their location or molecular activity. The use of other types of annotations, such as Yeast GO-Slim component, will lead to prediction of the location of gene's activity rather than their involvement in biological process and hence they are not used in the present investigation.

2.3. Genetic algorithm

Genetic algorithm (GA) [16] is a randomized search and optimization method to find out near optimal solutions for large combinatorial optimization problems. In GA, the parameters of the search space are encoded in the form of chromosomes and a collection of such chromosome is termed as population. A random population is created initially to represent different points in the search space. A certain number of chromosomes are then selected based on some methodologies like roulette wheel selection, tournament selection, and linear normalization selection [16,45]. The crossover and mutation [46] are then applied to these selected chromosomes probabilistically. The process of selection followed by crossover and mutation continues for a fixed number of generations or until a termination condition is satisfied. The Pseudo-code of genetic algorithm (GA) is shown in Algorithm 1.

Algorithm 1.

Algorithm 1 The pseudo-code of genetic algorithm (GA).

```
1: begin GA
2: create initial population
3: while iteration_count < K (K= max. number of iteration) do
4:     selection and elitism
5:     crossover
6:     mutation
7:     increment iteration_count
8: end while
9: output best chromosome
10: end GA
```

In this study, GA is used to find out a weight combination which maximizes the positive predictive value (PPV) for the top 10,000 gene pairs through its fitness function. Although for different time points different weights are assigned, but for a particular time point the weight remains the same for the expressions belonging to different genes. We used a randomly chosen real number in between 0 and 1 for each value of weight and the sequence of weights (weight combination) for various time points of the gene expression data represents a chromosome in GA. A collection of such chromosomes forms the initial population of GA. Among the different selection methods, we used linear normalization selection [16]. Here, an individual is ranked according to its fitness and then allowed to generate a number of offspring proportional to its rank position. A new population is thus created at each generation/iteration. Elitism [16] is used to preserve the fittest chromosome, where the fittest chromosome of previous generation randomly replaces a chromosome of new generation if the fitness of the fittest chromosome of the new generation is less compared to the previous generation. The single point crossover [47] is applied as a crossover technique where a single crossover point on both chromosomes is selected probabilistically and all the data beyond that point in both chromosomes are swapped between them. Then, a simple inversion mutation [48] is performed on each chromosome probabilistically. In this mutation, a point is selected randomly and it is then inverted. For example, if the value is 0.4 at the start then after mutation it will be 1–0.4 = 0.6.

2.4. Proposed method

The goal of the current study is to reduce the experimental errors related to gene expressions so that the expression similarity among the genes can properly reflect their functional similarity. In this regard, we propose a methodology, named *genetic algorithm for assigning weights to gene expressions using functional annotations* (GAAWGEFA), where functional annotations are used in a genetic algorithm (GA) framework to find out a combination of weights for the expressions which maximizes the fitness of the chromosomes in GA. First, the random

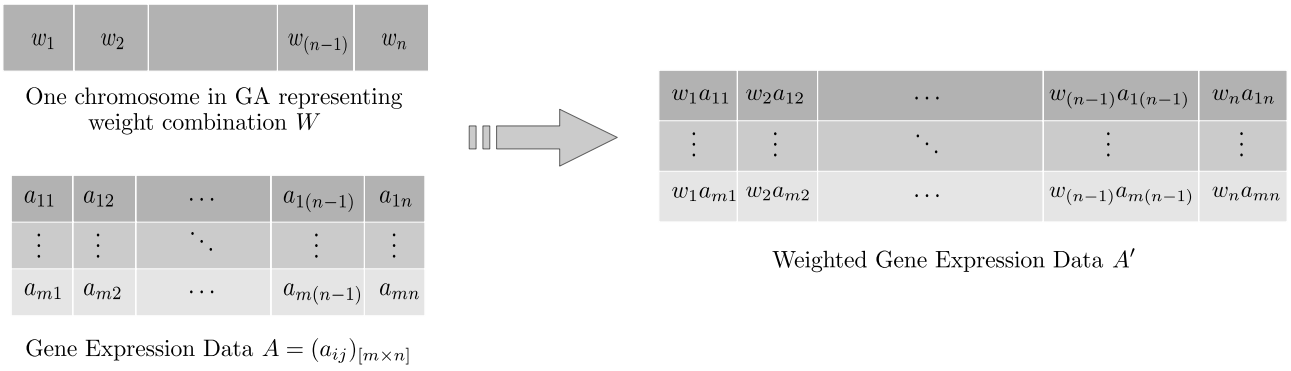

Fig. 1. Construction of weighted gene expression data using original gene expression data and weight combination.

Table 2
Values of different parameters for GAAWGEFA.

Parameters	NP	K	CR	MR
	(Population size)	(Max. number of iterations)	(Crossover rate)	(Mutation rate)
Values	20	1000	0.85	0.1

combination of weights is used to construct a chromosome. Next, those weights are used to define the weighted expressions. The weighted expressions are then used in computing the pairwise gene similarity and the top 10000 gene similarity values are used to compute the fitness for each chromosome in terms of the PPV. The weighted expressions will be able to reflect the functional similarity between the genes more accurately as compared with the functional similarity obtained using original expressions. The steps to determine the optimum weight combination using GA are described below:

S1. Create each chromosome by randomly selecting 'n' numbers (genes in chromosome of GA) having the values within 0–1 as weight combination for different time points, where 'n' is the number of time points in the gene expression dataset.
S2. All the expressions under each time point in the dataset is then multiplied with the corresponding gene values (weights) of the chromosome to obtain the weighted gene expression data.
S3. Construct similarity matrix (for all possible pairs) for each chromosome using weighted gene expression and Pearson correlation.
S4. For each chromosome, compute the fitness value in terms of the PPV (see Eq. (4)) for a given number of gene pairs (for example, top 10,000 gene pairs) obtained from the similarity matrix.
S5. Compute the fitness for all the chromosomes in the population and use elitism so that the fittest chromosome will pass through to the next iteration.
S6. Use linear normalized selection to select chromosomes for the new population.
S7. Use single point crossover between two chromosomes probabilistically.
S8. Perform inverse mutation on each chromosome probabilistically.
S9. Repeat step 2 to 8 for maximum number of iterations (K).

Fig. 1 shows how a weight combination W (w_1, w_2, ⋯⋯,$w_{(n-1)}$, w_n), represented by a chromosome in GA, is used to weight a gene expression data A (a_{11}, a_{12}, ⋯⋯,$a_{1(n-1)}$, a_{1n}, ⋯⋯⋯,a_{m1}, a_{m2}, ⋯,$a_{m(n-1)}$, a_{mn}) to obtain a weighted gene expression data A' (w_1a_{11}, w_2a_{12}, ⋯⋯,$w_{(n-1)}a_{1(n-1)}$, w_na_{1n}, ⋯⋯⋯,w_1a_{m1}, w_2a_{m2}, ⋯⋯⋯$w_{(n-1)}$ $a_{m(n-1)}$, w_na_{mn}). This weighted expression data is used to construct the similarity matrix and thereby computing the fitness value (PPV).

The GAAWGEFA is executed with various parameter settings and the best results are obtained for the parameters as specified in Table 2. The best result (fitness values) for each dataset is obtained over 20 runs for GAAWGEFA and the results for 20 runs are shown in Fig. 2. From the figure, it is evident that the difference among minimum fitness values (worst) and maximum fitness values (best) lies within 0.02–0.04 for different datasets, which shows the robustness of the proposed GAAWGEFA. The best, average, and worst fitness values over 20 runs are shown in Table 3. The robustness of the proposed method is also evident from the table. This combination of weights is assigned to the gene expressions for the evaluation purpose. The weight combinations for different datasets are available at http://www.isical.ac.in/~shubhra/GAAWGEFA/Weightcombination.xlsx. We have also de-

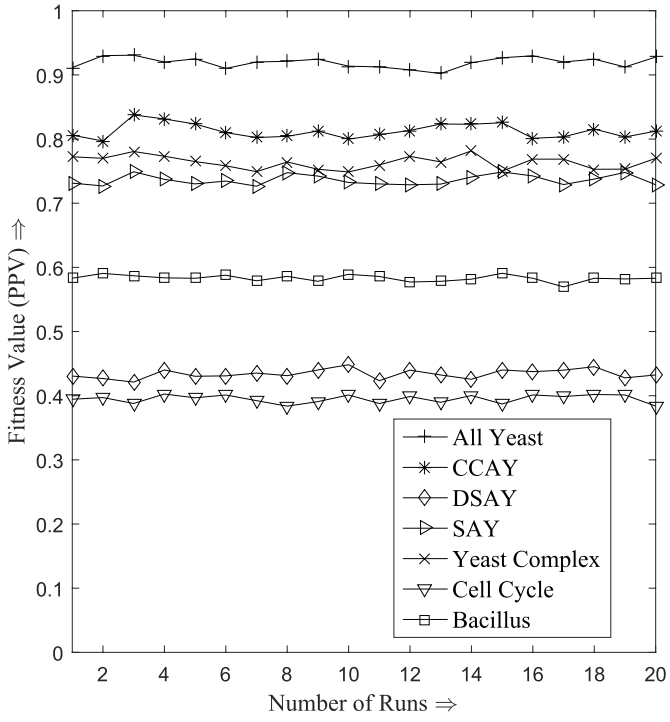

Fig. 2. Variation of fitness value (PPV) with number of runs for all the datasets.

Table 3
Best, average, and worst results (fitness value) for proposed GAAWGEFA over 20 runs. The best performance is highlighted in bold font.

Dataset	All Yeast	CCAY	DSAY	SAY	Yeast Complex	Cell Cycle	Bacillus
Best	**0.9308**	**0.8376**	**0.4483**	**0.7490**	**0.7816**	**0.4026**	**0.5908**
Average	0.9201	0.8135	0.4353	0.7372	0.7638	0.3923	0.5831
Worst	0.9021	0.7969	0.4212	0.7265	0.7492	0.3837	0.5696

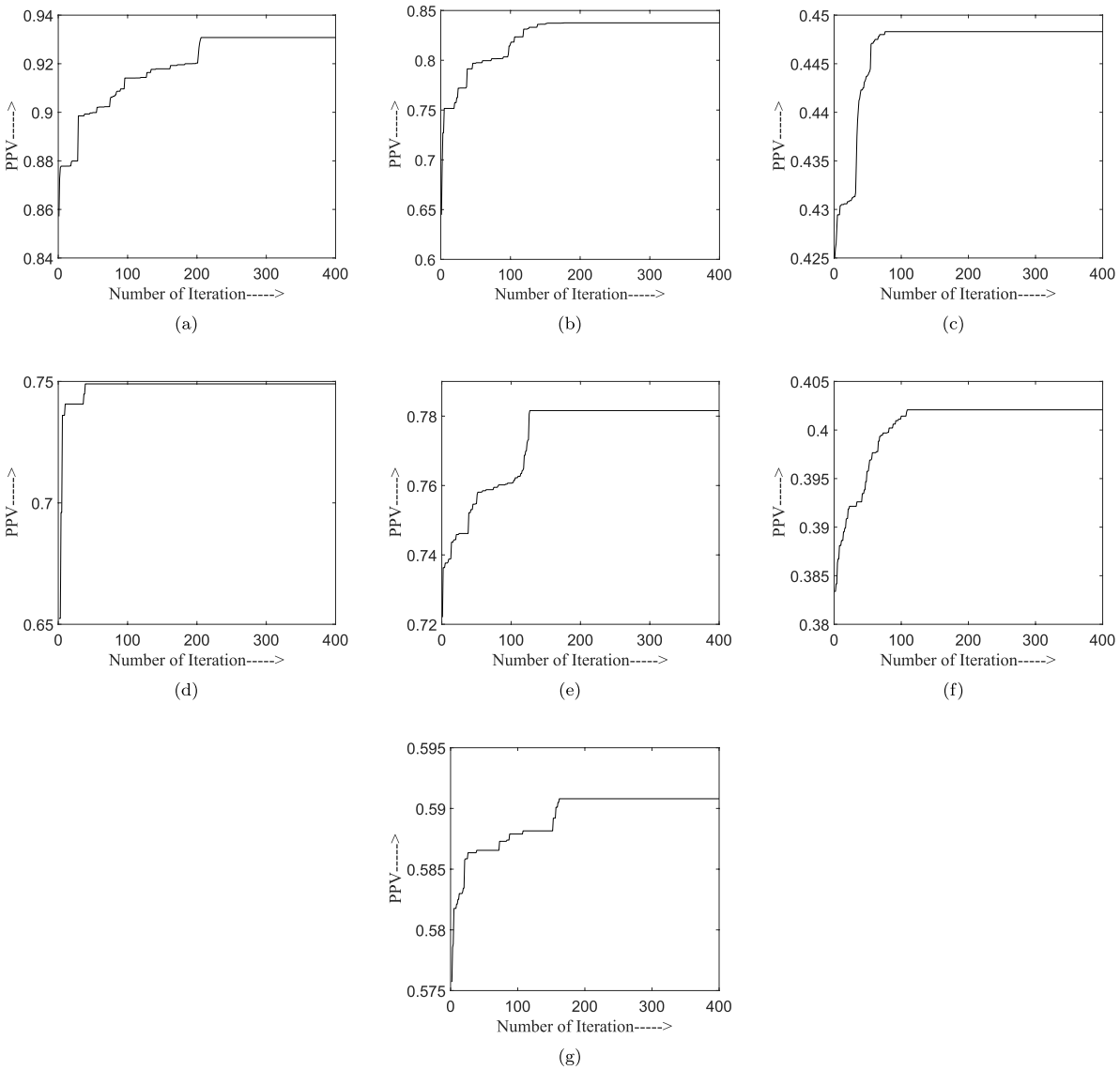

Fig. 3. Variation of PPV (computed using Yeast GO-Slim process (SGD) annotations) with number of iterations of genetic algorithm for (a) All Yeast, (b) CCAY, (c) DSAY, (d) SAY, (e) Yeast Complex, (f) Cell cycle, and (g) Bacillus datasets. For better visualization, we have shown till 400 iterations in the plots.

signed a webpage at http://www.sampa.droppages.com/GAAWGEFA.html, where the source code for the main program for calculating the weight combination for the expression values using genetic algorithm is available.

Let us consider the expression values for the two genes A_i and A_j as $A_i = a_{i1}, \cdots, a_{in}$ and $A_j = a_{j1}, \cdots, a_{jn}$, where n is the total number of time point.

If the optimum weight combination (W) is determined using genetic algorithm as $W = w_1, \cdots, w_n$, then after assigning weights, they will be $A'_i = w_1 a_{i1}, \cdots, w_n a_{in}$ and $A'_j = w_1 a_{j1}, \cdots, w_n a_{jn}$. Hence, the Pearson correlation between two gene expression will be

$$P_{(A'_i, A'_j)} = \frac{\frac{1}{n}\sum_{k=1}^{n}(A'_{i_k} - \overline{A'_i})(A'_{j_k} - \overline{A'_j})}{\sigma_{A'_i}\sigma_{A'_j}}. \quad (5)$$

Example: Let us consider a gene A_i with the expression values 0.21, 0.89, 0.82, 0.23 and other gene A_j with the expression values 0.3, 0.52, 0.1, and 0.9. The optimum weights are determined from the genetic algorithm for each time point as 0.2, 0.1, 0.9, and 0.5. Therefore, after assigning the weights the expression values for the two genes will be 0.042, 0.089, 0.7380, 0.1150 and 0.06, 0.05, 0.09, 0.45. Hence, the Pearson correlation between two genes A_i and A_j will be -0.1848.

The Fig. 3(a)–3(g) represents how the PPV values for the top 10,000 gene pairs i.e., the fitness of GA, vary with each of its iteration for All

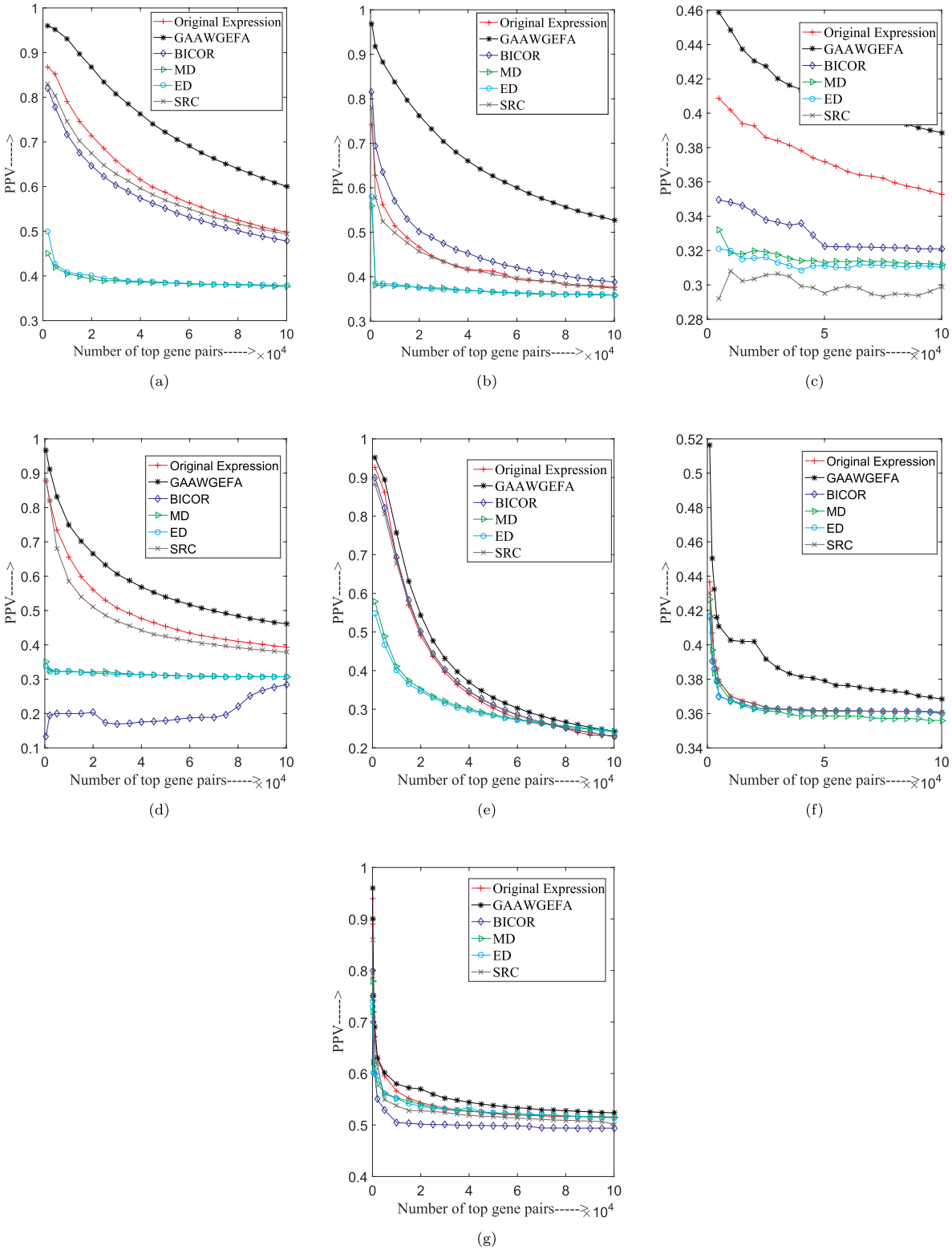

Fig. 4. PPVs (computed using Yeast GO-Slim process (SGD) annotations) versus the number of top gene pairs for different methods for (a) All Yeast, (b) CCAY, (c) DSAY, (d) SAY, (e) Yeast Complex, (f) Cell cycle, and (g) Bacillus datasets.

Yeast, CCAY, DSAY, SAY, Yeast Complex, Cell Cycle, and Bacillus datasets, respectively. It is evident from the curves that the PPV value increases for the first hundred iterations in most of the cases and then it saturates at a certain value. For example, for the All Yeast dataset the PPV value increases from 0.74 to 0.86 for the first 200 iterations and thereafter remains unchanged.

The optimum weight combination is the one for which the PPV for the top 10,000 gene pairs is maximized. From the figures, it can be observed that optimum weight combination is obtained after 150 iterations for All Yeast, CCAY, DSAY, Yeast Complex, and Bacillus data.

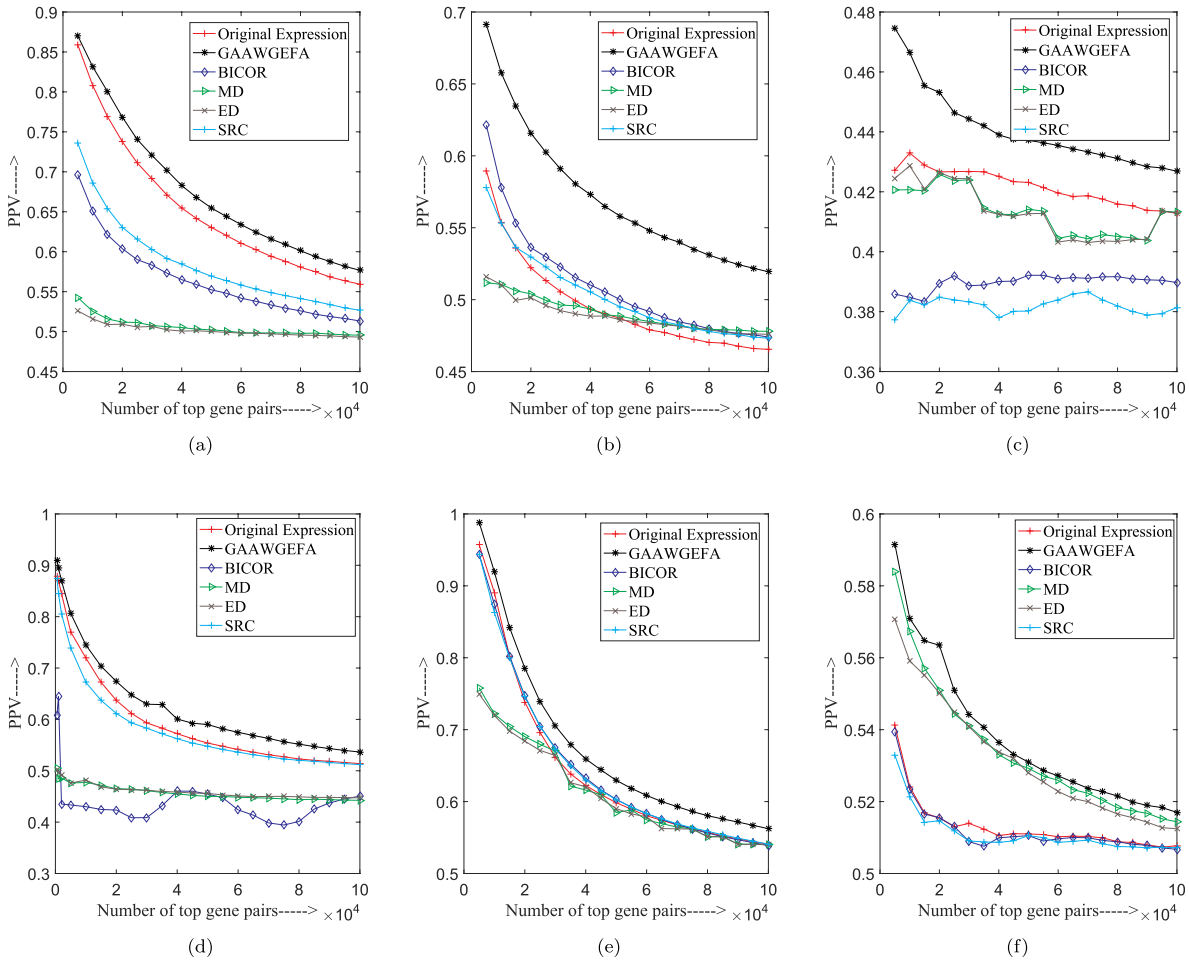

Fig. 5. PPVs (computed using MIPS annotations) versus the number of top gene pairs for different methods using (a) All Yeast, (b) CCAY, (c) DSAY, (d) SAY, (e) Yeast Complex, and (f) Cell cycle datasets.

For SAY and Cell Cycle data, the optimum weight combination is obtained within 100 iterations.

As mentioned earlier, the original gene expression data may contain noise, technical or experimental errors and thereby not properly reflecting the actual biological/functional similarity between the genes. Therefore, a new method is developed, where the weights are determined by maximizing the positive predictive value using gene annotations for the top 10,000 gene pairs in terms of fitness function.

3. Experimental results

In this section, the experimental results using our proposed GAAWGEFA are reported followed by its application in predicting the function of unclassified genes.

3.1. Performance evaluation

The efficiency of our proposed method is evaluated based on two criteria: the PPV vs. top gene pairs and the PPV vs. similarity measure. Further, for the PPV vs. top gene pairs, the comparisons are carried out in two ways i.e., in one case the PPV is computed using Yeast GO-Slim process (SGD) annotations and in other case it is computed using Munich Information for Protein Sequences (MIPS) annotations. As the primary goal of our investigation is to enhance gene similarity using gene expression, it is only fair to compare it with the methods that can handle gene expression similarity, unlike GO induced methods (Resnik [57], Jiang & Conrath [58], and Lin [59]).

3.1.1. PPV vs. top gene pairs

The performance of the top gene pairs in terms of PPV for All Yeast, CCAY, DSAY, SAY, Yeast Complex, Cell Cycle, and Bacillus datasets are shown in Fig. 4(a)–4(g), respectively. It is observed that the increase in PPV value is inversely proportional to the number of top gene pairs. Furthermore, the performance of the weighted expression obtained using proposed GAAWGEFA method is found to be superior compared to the results obtained using original expression as well as BICOR for all the datasets as the curves corresponding to GAAWGEFA appear at the top. However, the performance of BICOR for Yeast Complex and Cell Cycle datasets almost remains the same like original expressions. For CCAY dataset, the performance is slightly improved and for All Yeast, DSAY, SAY, and Bacillus datasets the performance seems to be inferior to original expressions. For example, the PPVs for the top 50,000 gene pairs obtained using the original expressions are 0.58, 0.41, 0.37, 0.45, 0.30, 0.36, and 0.52 for All Yeast, CCAY, DSAY, SAY, Yeast Complex, Cell Cycle, and Bacillus dataset, respectively. On the other hand, those values are obtained as 0.72, 0.63, 0.40, 0.54, 0.33, 0.38, and 0.54 using the proposed GAAWGEFA and 0.55, 0.43, 0.32, 0.2, 0.31, 0.36, and 0.49 for the BICOR. The performance of Euclidean distance (ED) and Manhattan distance (MD) are noticeably inferior for all the datasets. The curves for Spearman rank correlation (SRC) suggest that their performance is slightly superior or comparable to original expressions.

As Yeast GO-Slim process (SGD) annotations are used for the computation of the PPV in the optimization process, we additionally used Munich Information for Protein Sequences (MIPS) annotations for the computation of the PPV in the evaluation process involving the PPV vs.

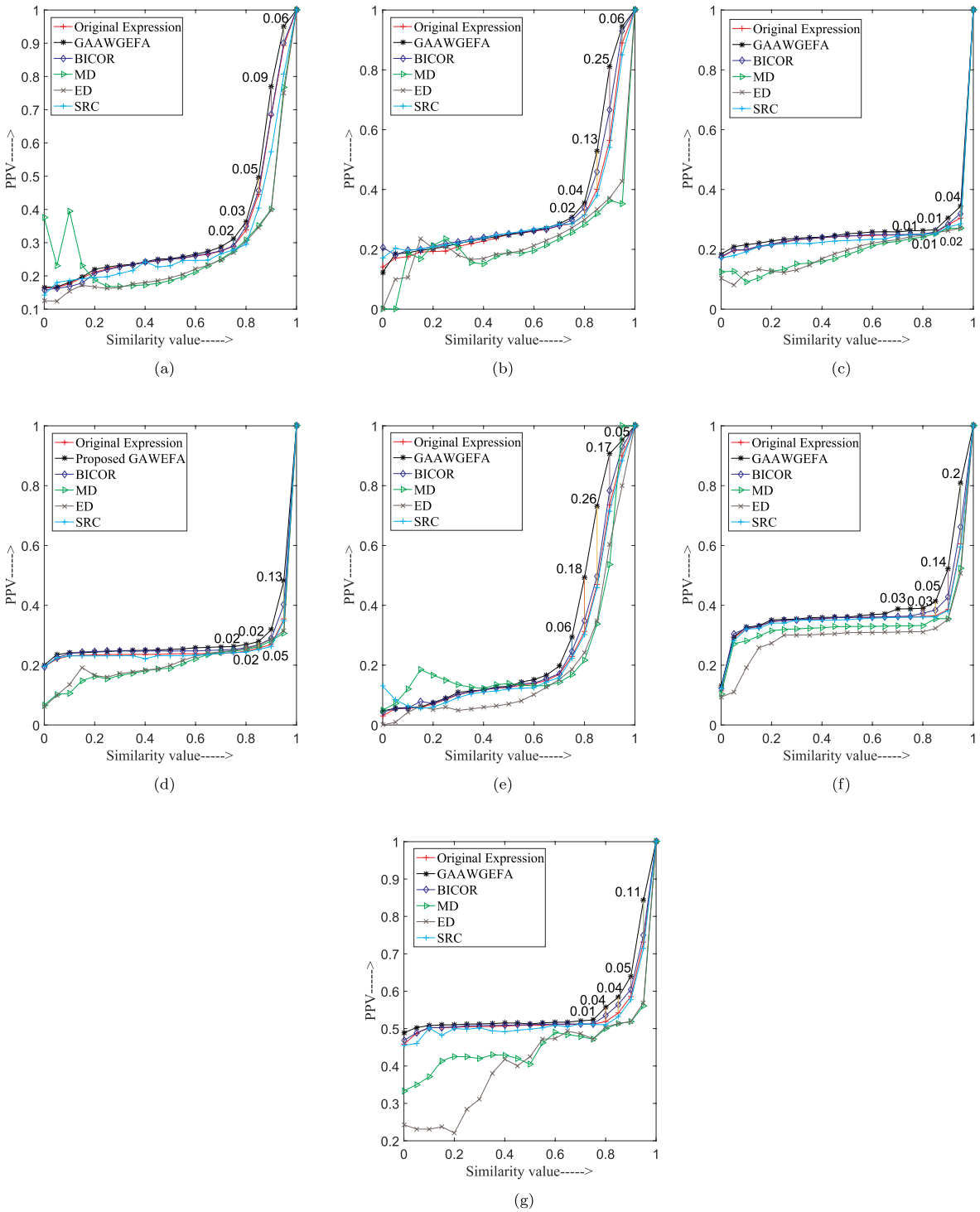

Fig. 6. Comparing PPVs (computed using Yeast GO-Slim process (SGD) annotations) versus the similarity values between GAAWGEFA and different similarity measures for (a) All Yeast, (b) CCAY, (c) DSAY, (d) SAY, (e) Yeast Complex, (f) Cell cycle, and (g) Bacillus datasets. The vertical lines between curves obtained using GAAWGEFA (at the top) and original expression show the differences between them in terms of PPV at a particular similarity value.

top gene pairs curves for different methods. The PPV vs. top gene pairs curves for different methods and different datasets using MIPS annotations are shown in Fig. 5(a)–5(f). From the figure (Fig. 5(a)–5(f)), it can be observed that the performance of the proposed GAAWGEFA method using MIPS annotations is also found to be superior compared to the results obtained using other existing methodologies for all the datasets. It is worth noting that, for the Bacillus dataset, the gene annotations are only available for MIPS and hence we could not use two different annotations databases for the optimization and the evaluation purpose.

3.1.2. *PPV vs. similarity measures*

The performance of GAAWGEFA is also compared with original expressions and BICOR in terms of the PPVs at various similarity values. The Fig. 6(a)–6(g) demonstrates the performance of proposed GAAWGEFA for All Yeast, CCAY, DSAY, SAY, Yeast Complex, Cell Cycle, and

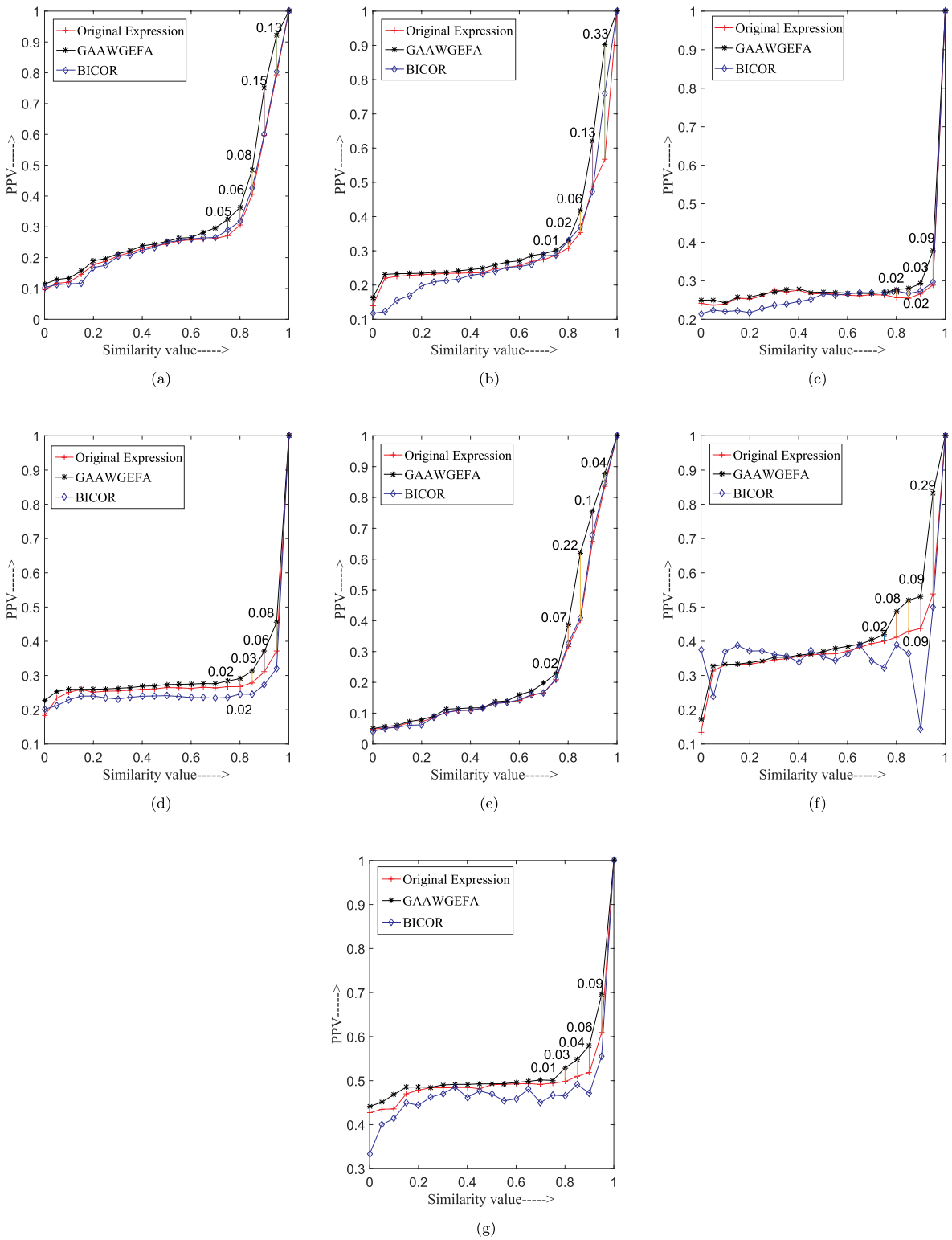

Fig. 7. Cross validation results comparing GAAWGEFA with original expressions and BICOR in terms of PPV versus similarity value. The curves are shown for one of the instances in 5-fold cross validation using (a) All Yeast, (b) CCAY, (c) DSAY, (d) SAY, (e) Yeast Complex, (f) Cell Cycle, and (g) Bacillus datasets. The vertical lines between curves obtained using GAAWGEFA and original expression show the differences between them in terms of PPV at a particular similarity value.

Bacillus datasets, respectively. The improved performance of the GAAWGEFA is evident in these figures as compared with the results obtained using original expressions above the similarity values of 0.5, 0.6, 0.5, 0.3, 0.3, 0.6, and 0.5 for All Yeast, CCAY, DSAY, SAY, Yeast Complex, Cell Cycle, and Bacillus dataset, respectively. The performance of GAAWGEFA is also appeared to be superior compared to the results obtained using BICOR for all the datasets. For example, the PPV values at 0.85 similarity value using the original expression for All Yeast, CCAY, DSAY, SAY, Yeast Complex, Cell Cycle, and Bacillus datasets are 0.44, 0.4, 0.25, 0.25, 0.46, 0.36, and 0.54, respectively.

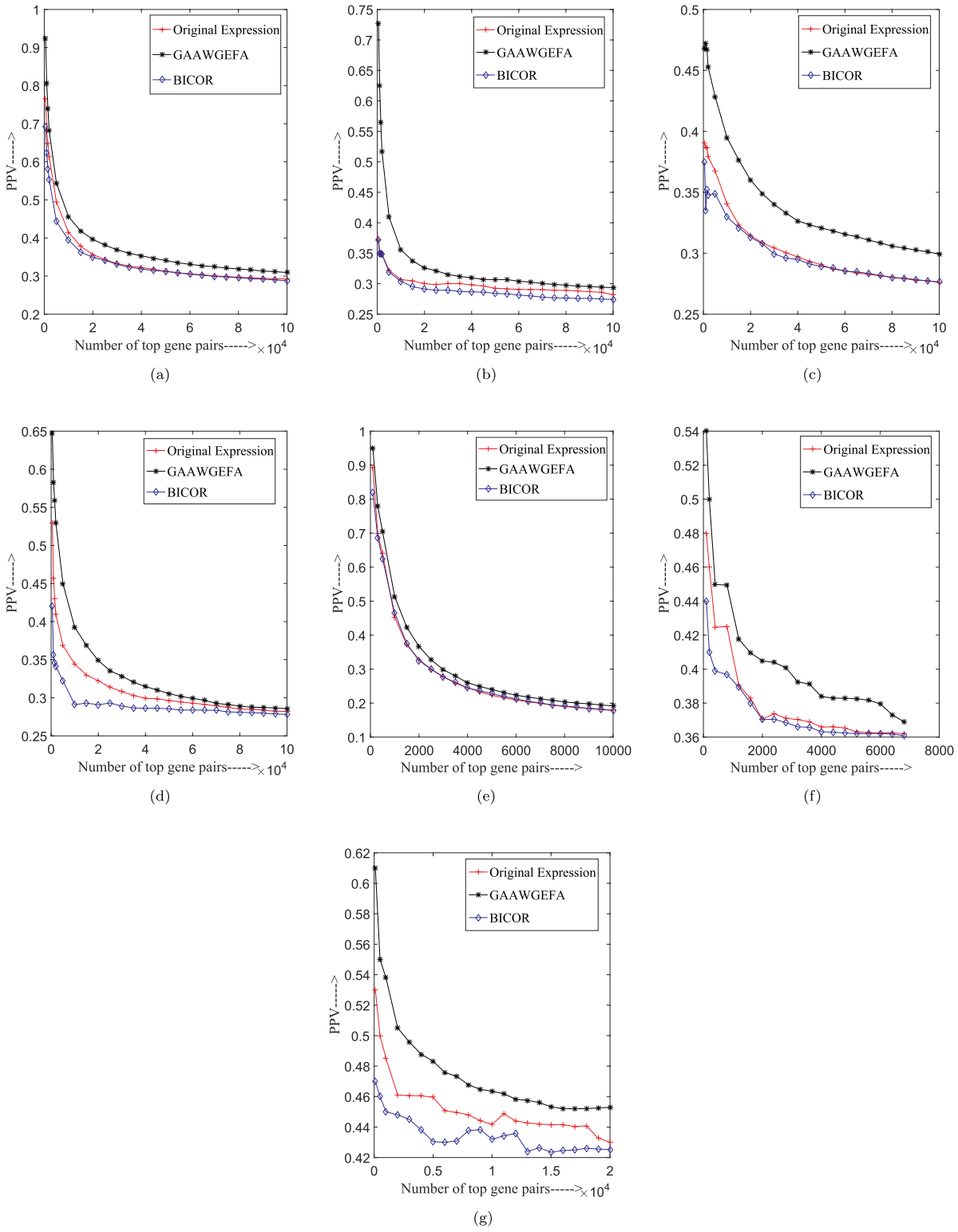

Fig. 8. Cross validation results comparing GAAWGEFA with original expressions and BICOR in terms of PPV versus the number of top gene pairs. The curves are shown for one of the instances in 5-fold cross validation using (a) All Yeast, (b) CCAY, (c) DSAY, (d) SAY, (e) Yeast Complex, (f) Cell Cycle, and (g) Bacillus datasets.

Whereas, by employing GAAWGEFA, the PPVs at 0.85 similarity value are obtained to be 0.5, 0.52, 0.27, 0.28, 0.73, 0.4, and 0.57, respectively for the above datasets. For BICOR, these values are observed to be 0.45, 0.46, 0.26, 0.26, 0.49, 0.49, and 0.55, respectively. It can be noticed from these figures that the performance of BICOR is slightly better than the original expressions for all the datasets. Whereas, the performance of SRC is slightly superior or comparable to the original expressions within the similarity value range of 0.5–0.9. However, the performance of ED and MD measures are noticeably inferior.

Table 4
Results of *t*-test for GAAWGEFA & original expressions and GAAWGEFA & BICOR.

			All Yeast	CCAY	DSAY	SAY	Yeast Complex	Cell Cycle	Bacillus
PPVs at various similarity values	GAAWGEFA & original expressions	*t*	3.16	2.07	4.69	2.98	2.89	2.51	2.68
		p	0.007	0.038	0.001	0.009	0.010	0.018	0.014
	GAAWGEFA & BICOR	*t*	2.88	1.98	6.14	2.54	2.62	2.60	2.37
		p	0.010	0.041	0.001	0.017	0.015	0.016	0.022
PPVs at various number of top gene pairs	GAAWGEFA & original expressions	*t*	30.50	19.01	40.98	33.94	9.29	7.29	12.44
		p	0.001	0.001	0.001	0.001	0.001	0.001	0.001
	GAAWGEFA & BICOR	*t*	23.47	21.38	35.82	11.10	7.21	6.29	9.43
		p	0.001	0.001	0.001	0.001	0.001	0.001	0.001

3.2. Cross validation

The performance of GAAWGEFA is evaluated in a conventional way via 5-fold cross validation. Here, 4 folds of the genes along with their annotations profile are randomly selected for training. The purpose of training is to find out a weight combination for the gene expressions in a genetic algorithm framework so that the PPV can be maximized for the top 10,000 gene pairs. The remaining one fold of the genes is used for the evaluation process. In the evaluation process, the weight combination obtained from the training phase is assigned to the expressions of the genes in the remaining one fold and their performance is compared with original expressions and BICOR in terms of the PPVs obtained at various similarity values and the PPVs at various number of top gene pairs. The process is repeated 5 times for each dataset and the result for one of those instances using All Yeast, CCAY, DSAY, SAY, Yeast Complex, Cell Cycle, and Bacillus datasets for the PPV vs. similarity value is presented in Fig. 7(a)–7(g), respectively. The performance of the top gene pairs in the cross validation procedure for All Yeast, CCAY, DSAY, SAY, Yeast Complex, Cell Cycle, and Bacillus datasets is shown in Fig. 8(a)–8(g), respectively. From the figures, it is clear that the performance of the proposed GAAWGEFA is superior as compared with the results obtained using original expressions and BICOR. It is worth noting that, the curves for GAAWGEFA, original expressions, and BICOR are only used for cross validation because they appear above the other curves for most of the datasets and clearly superior to others.

3.3. Statistical analysis

To statistically compare the performance of the proposed GAAWGEFA with original expressions for the PPVs at various similarity values, *t*-tests are performed with the PPV values above 0.5 similarity value (as shown in Fig. 6(a)–6(g)) using

$$t = \frac{\overline{SI_1} - \overline{SI_2}}{\sqrt{\frac{VarianceSI_1}{n_1} + \frac{VarianceSI_2}{n_2}}}. \quad (6)$$

Here, $\overline{SI_1}$ and $VarianceSI_1$ are the mean and the variance of all the available PPV values above 0.5 similarity value for the proposed GAAWGEFA method for each dataset shown in Fig. 6(a)–6(g). We used the PPV values obtained above 0.5 similarity value because the genes with similarity more than 0.5 are treated as highly similar gene pair. The highly similar gene pairs are generally clustered together and are used for function prediction in the later stage (in Section 3.4). SI_2 is used for original expressions. We used $n_1 = n_2 = 9$, because there are 9 values available above 0.5 similarity value for each of the dataset. Therefore, the degrees of freedom for the *t*-test are $9\times2 - 2 = 16$. Similarly, the *t*-test is also performed for GAAWGEFA and BICOR. Note that, *t*-tests are not performed for GAAWGEFA & ED, GAAWGEFA & MD, and GAAWGEFA & SRC as the curves for ED, MD, and SRC appear noticeably lower compared to GAAWGEFA. The alternative hypothesis, that implies the PPV value for GAAWGEFA is superior as compared with other related methods (original expressions or BICOR), is used for the calculation of *t*-statistics. The *t* values and related *p* values are shown in the first 4 rows of Table 4. The observed *p* values (Table 4) for the corresponding *t* values eventually reject the null hypothesis that there is no difference of the PPVs at various similarity values between GAAWGEFA and other related methods (original expressions or BICOR) with significance level of 0.05 for all the datasets in favor of the alternative.

Similar types of *t*-tests for the PPVs at various number of top gene pairs of the GAAWGEFA and related methods (original expressions and BICOR) are also performed. The results are shown in the last 4 rows of Table 4. For each method and each dataset, there are 20 values and hence $20 \times 2 - 2 = 38^{\circ}$ of freedom are available for each *t*-test. The alternative hypothesis, that implies the PPV of top gene pairs using GAAWGEFA is superior as compared with the related methods (original expressions or BICOR), is used for the calculation of *t*-statistics. The observed *p* values (Table 4) for the corresponding *t* values reject the null hypothesis that there is no difference between the PPVs at various number of top gene pairs for the GAAWGEFA and related methods (original expressions or BICOR) with significance level of 0.0001 in favor of the alternative.

3.4. Function prediction

The function prediction of unclassified genes is carried out with All Yeast dataset. The MIPS [44] annotations are used for function prediction because the SGD annotations are used to determine the weight of expression at each time point. The weighted gene expression dataset, obtained by the proposed method, is clustered using *k*-medoids [49] algorithm where the value of *k* is chosen to be 510. This *k* value is in accordance with the 510 functional categories in Munich Information for Protein Sequences (MIPS) annotations. After clustering, the biological function of an unclassified gene is predicted from the functional enrichment of the cluster using MIPS annotations. The steps for gene function prediction using the All Yeast data are as follows:

S1) Assign weights to the expressions of each time point using GAAWGEFA method.
S2) Calculate pairwise similarity for the all possible pairs using Pearson correlation from weighted expressions and cluster the genes using *k*-medoids algorithm. The value of *k* is chosen to be 510 as there exist 510 functional categories in MIPS.
S3) 46 clusters are identified with functional enrichment in MIPS categories and *p*-value less than 10^{-5}.
S4) From these 46 clusters, functions of 143 unclassified genes are predicted by assigning the function related to the cluster.

The functions of 143 unclassified genes are predicted with a p-value cutoff of 10^{-5} using Munich Information for Protein Sequences (MIPS) annotations for the All Yeast data. The details regarding the functions of 143 unclassified genes are available at http://www.isical.ac.in/~shubhra/GAAWGEFA/predictionAY-p-5.xls. Out of these genes, functional categories of 48 genes are reported in Table 5 from the 12 clusters with a *p*-value cutoff of 10^{-13}. For each cluster, the unclassified

Table 5
Function Predictions of Top 48 Unclassified Genes at p-value Less Than 10^{-13} for All Yeast Dataset using MIPS Annotations.

Cluster number	Unclassified Genes	Functional Category	p-value	No of Genes Within Cluster	No. of Genes Within the Genome
1	YGR210C	ribosomal proteins	8.9999e-15	15	221
2	YCLX11W	amino acid metabolism	9.355e-16	14	239
3	YHR131C YIL017W YOR292C YBL101W-A YER080W YFL-TYA YFL-TYB	transposable elements, viral, and plasmid proteins	2.2717e-14	9	34
4	YKR011C	proteasomal degradation (ubiquitin/ proteasomal pathway)	2.06e-16	16	126
5	YMR031C	respiration	9.2198e-16	12	124
6	YOR007C YKR018C YJL021C YPR158W YCR097WB	protein folding and stabilization	1.2956e-16	13	91
7	YHR049W	ribosomal proteins	8.9851e-15	11	221)
8	YAL018C YNR014W YDL186W YOL047C YOR365C YFR032C YGL138C YCRX06W YNL018C YBR063C YNL019C YER085C YJL037W YJL038C YNL033W YPR027C YJL043W YOL024W YER033C YER182W	development of asco- basidio- or zygospore	5.2106e-14	17	161
9	YJL213W YCRX17W YMR295C	glycolysis and gluconeogenesis	2.8781e-16	11	58
10	YLR077W YER0561C	ribosomal proteins	5.55e-18	26	221
11	YKR077W YHR218W YPR202W YPR203W YLR464W	DNA topology	1.2394e-16	15	52
12	YER0191C	ribosome biogenesis	2.7408e-15	15	284

genes, the predicted function, the related p-value, the number of genes within the cluster, and the number of genes within the genome are also presented in Table 5. The detail information related to Table 5 is available at http://www.isical.ac.in/~shubhra/GAAWGEFA/predictionAY-p-13.xls.

3.5. Biological relevance

Now we will discuss about the biological relevance of the predicted functions for some of the unclassified genes presented in Table 5.

The predicted function for YGR210C in cluster 1 is “ribosomal protein”. The gene YGR210C encodes a protein of 411 amino acids with a strong homology with the *Sz. pombe* protein SPBC428.15 [50] which interacts with rix7 gene to produce ribosome assembly ATPase Rix7 [51]. Hence, our prediction for YGR210C as “ribosomal protein” is a likely one.

In cluster 6, the gene YOR007C is found to be involved in “protein folding and stabilization”. The gene YOR007C was thought to be the homologue of vertebrate SGT (small glutamine tetratricopeptide) containing protein [52]. The SGT has been known to interact with stress-induced 70-kDa heat shock protein Hsp70 which performs many cellular functions like protein transport into organelles and refolding of denatured proteins [53,54]. Therefore, our prediction for YOR007C as “protein folding and stabilization” is a promising one.

The function for gene YJL213W is predicted as “glycolysis and gluconeogenesis” in cluster 9. The gene YJL213W shows similarity to *Methanobacterium aryldialkylphosphatase* related protein [55]. In general, *aryldialkylphosphatase* related proteins are involved in the hydrolase activity on the carbon-nitrogen bonds except peptide. As gluconeogenesis (GNG) is a metabolic pathway that results in the generation of glucose from certain non-carbohydrate carbon substrates [55,56], therefore our function prediction for YJL213W as “glycolysis and gluconeogenesis” is a possible one.

4. Discussion

In this article, six microarray datasets of *Saccharomyces cerevisiae*, namely, All Yeast, Cell Cycle All Yeast, Diauxic Shift All Yeast, Sporulation All Yeast, Yeast Complex, Cell Cycle and one microarray dataset of *Bacillus subtilis*, namely, Bacillus are used. Here, we proposed an expression weighting method, named GAAWGEFA, using functional annotations of the genes so that the expression similarity among the genes can properly reflect their functional similarity. The performance of the proposed GAAWGEFA is found to be superior as compared with the existing similarity measures like Pearson correlation (PC) [10,28], Euclidean distance (ED) [29], Manhattan distance (MD) [9], Spearman rank correlation (SRC) [30], and BICOR [14] in terms of the PPVs at various similarity values and also at various number of top gene pairs. The GAAWGEFA increases the expression similarity for the functionally similar gene pairs. This is demonstrated by plotting the PPVs of the gene pairs for different number of the top gene pairs (Fig. 4(a)–4(g) & 5(a)-5(f)) and the PPVs of the gene pairs at various similarity values (Fig. 6(a)–6(g)). From the figures, it is evident that the expression weighting using functional annotations performs better compared to the related existing methods as for all the datasets the curves for GAAWGEFA appear above the curves for the related existing methods beyond 0.5 similarity value and for various number of the top gene pairs. In this investigation, the proposed GAAWGEFA provides a framework to utilize GO annotations for improving the gene expression similarity which is not possible using GO-induced methods [57–59]. In addition, these methods explore various possibilities of GO function similarity rather than exploring the expression similarities between two genes [60]. As our investigation is focused on measuring the expression similarity, any comparison with the aforementioned ontology based (or functional) similarity methods is not feasible. Further, if the genes are clustered using functional annotations, then it will not be possible to

cluster any unclassified gene along with the classified genes, which is required for function prediction of unclassified gene.

5. Conclusion

In this study, a novel method, named *genetic algorithm for assigning weights to gene expressions using functional annotations* (GAAWGEFA), is developed. The pairwise similarity for the all possible pairs is calculated from the weighted expressions using Pearson correlation. The combination of weights is determined by maximizing the 'PPV for the top 10,000 gene pairs' (fitness function of GA, computed using functional annotations) via genetic algorithm. The superiority of the proposed GAAWGEFA is demonstrated by plotting i) the PPV vs. top gene pairs and ii) the PPV vs. similarity values. The comparisons for the PPV vs. top gene pairs are performed using SGD GO Slim process annotations as well as MIPS annotations. The curves for GAAWGEFA appear at the top for All Yeast, Cell Cycle All Yeast, Diauxic Shift All Yeast, Sporulation All Yeast, Yeast Complex, Cell Cycle, and Bacillus dataset. The *k*-medoids clustering technique is applied to the All Yeast dataset to predict the function of unknown genes. Here, the functions of 143 unclassified *Saccharomyces cerevisiae* genes are predicted from 46 clusters using a *p*-value cutoff of 10^{-5} from the All Yeast dataset. Out of these 143 genes, functional categories of 48 unclassified genes are reported in this investigation with a *p*-value cut off of 10^{-13}. Our results indicate that the GAAWGEFA is capable of identifying the similarity between the gene expression profiles in an improved manner in terms of functional annotations. The GAAWGEFA is unique as it assigns proper weights to every time point in a functional gene expression data using functional annotations in a genetic algorithm framework. It can be applied to functional time series data of any species for improved similarity where functional annotations of some genes and expression values for different time points are available.

References

[1] S. Brenner, F. Jacob, M. Meselson, An unstable intermediate carrying information from genes to ribosomes for protein synthesis, Nature 190 (4776) (1961) 576–581.
[2] A. Brazma, J. Vilo, Gene expression data analysis, FEBS Lett. 480 (1) (2000) 17–24.
[3] Y. Christinat, B. Wachmann, L. Zhang, Gene expression data analysis using a novel approach to biclustering combining discrete and continuous data, IEEE ACM Trans. Comput. Biol. Bioinf 5 (Oct-Dec. 2008) 583–593.
[4] G. Sherlock, et al., The stanford microarray database, Nucleic Acids Res. 29 (1) (2001) 152–155.
[5] V. Filkov, S. Skiena, J. Zhi, Analysis techniques for microarray time-series data, J. Comput. Biol. 9 (2) (2002) 317–330.
[6] J.D. Hoheisel, Microarray technology: beyond transcript profiling and genotype analysis, Nat. Rev. Genet. 7 (3) (2006) 200–210.
[7] A. Schulze, J. Downward, Navigating gene expression using microarraysa technology review, Nat. Cell Biol. 3 (8) (2001) E190–E195.
[8] J. Herrero, R. Díaz-Uriarte, J. Dopazo, Gene expression data preprocessing, Bioinformatics 19 (5) (2003) 655–656.
[9] S.S. Ray, S. Bandyopadhyay, S.K. Pal, Dynamic range-based distance measure for microarray expressions and a fast gene-ordering algorithm, IEEE Trans. Syst. Man Cybern. B Cybern. 37 (Jun. 2007) 742–749.
[10] T. Biedl, B. Brejová, E.D. Demaine, A.M. Hamel, T. Vinar, Optimal Arrangement of Leaves in the Tree Representing Hierarchical Clustering of Gene Expression Data, Technical report Department of Computer Science, University of Waterloo, Canada, 2001.
[11] C.A. Gallo, R.L. Cecchini, J.A. Carballido, S. Micheletto, I. Ponzoni, Discretization of gene expression data revised, Briefings Bioinf. 17 (5) (2015) 758–770.
[12] P. Berka, I. Bruha, Discretization and grouping: preprocessing steps for data mining, in: J.M. Żytkow, M. Quafafou (Eds.), Principles of Data Mining and Knowledge Discovery, Springer Berlin Heidelberg, Berlin, Heidelberg, 1998, pp. 239–245.
[13] S. Misra, S.S. Ray, Finding optimum width of discretization for gene expressions using functional annotations, Comput. Biol. Med. 90 (2017) 59–67.
[14] L. Song, P. Langfelder, S. Horvath, Comparison of co-expression measures: mutual information, correlation, and model based indices, BMC Bioinf. 13 (1) (2012) 1–21.
[15] Y. Zhou, J.A. Young, A. Santrosyan, K. Chen, S.F. Yan, E.A. Winzeler, In silico gene function prediction using ontology-based pattern identification, Bioinformatics 21 (7) (2004) 1237–1245.
[16] D.E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Machine Learning, Addison-Wesley, New York, 1989.
[17] S. Selvaraj, J. Natarajan, Microarray data analysis and mining tools, Bioinformation 6 (3) (2011) 95–99.
[18] S.S. Ray, S. Bandyopadhyay, S.K. Pal, A weighted power framework for integrating multisource information: gene function prediction in yeast, IEEE (Inst. Electr. Electron. Eng.) Trans. Biomed. Eng. 59 (Apr. 2012) 1162–1168.
[19] O.G. Troyanskaya, K. Dolinski, A.B. Owen, R.B. Altman, D. Botstein, A bayesian framework for combining heterogeneous data sources for gene function prediction (in Saccharomyces cerevisiae), Proc. Natl. Acad. Sci. U. S. A. 100 (14) (2003) 8348–8353.
[20] J.L. Lustgarten, S. Visweswaran, V. Gopalakrishnan, G.F. Cooper, Application of an efficient bayesian discretization method to biomedical data, BMC Bioinf. 12 (2011) 1–15.
[21] T. Schlitt, A. Brazma, Current approaches to gene regulatory network modelling, BMC Bioinf. 8 (6) (2007) 1–22.
[22] I. Shmulevich, J.D. Aitchison, Deterministic and stochastic models of genetic regulatory networks, Methods Enzymol. 467 (2009) 335–356.
[23] I. Ponzoni, F. Azuaje, J. Augusto, D. Glass, Inferring adaptive regulation thresholds and association rules from gene expression data through combinatorial optimization learning, IEEE ACM Trans. Comput. Biol. Bioinf 4 (Oct-Dec. 2007) 624–634.
[24] E.M. Marcotte, M. Pellegrini, H.-L. Ng, D.W. Rice, T.O. Yeates, D. Eisenberg, Detecting protein function and protein-protein interactions from genome sequences, Science 285 (5428) (1999) 751–753.
[25] V. Spirin, L.A. Mirny, Protein complexes and functional modules in molecular networks, Proc. Natl. Acad. Sci. Unit. States Am. 100 (21) (2003) 12 123–12 128.
[26] T.R. Golub, et al., Molecular classification of cancer: class discovery and class prediction by gene expression monitoring, Science 286 (5439) (1999) 531–537.
[27] S. van Dam, U. Võsa, A. van der Graaf, L. Franke, J.P. de Magalhães, "Gene co-expression analysis for functional classification and gene–disease predictions, Briefings Bioinf. 19 (4) (2017) 575–592.
[28] H.J. Bussemaker, H. Li, E.D. Siggia, Regulatory element detection using correlation with expression, Nat. Genet. 27 (2) (2001) 167–171.
[29] D.K. Slonim, From patterns to pathways: gene expression data analysis comes of age, Nat. Genet. 32 (2002) 502–508.
[30] J. Hardin, A. Mitani, L. Hicks, B. VanKoten, A robust measure of correlation between two genes on a microarray, BMC Bioinf. 8 (1) (2007) 1–13.
[31] P.T. Spellman, G. Sherlock, M.Q. Zhang, V.R. Iyer, K. Anders, M.B. Eisen, P.O. Brown, D. Botstein, B. Futcher, "Comprehensive identification of cell cycle–regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization, Mol. Biol. Cell 9 (12) (1998) 3273–3297.
[32] R.C. Baishya, R. Sarmah, D.K. Bhattacharyya, M.A. Dutta, A similarity measure for clustering gene expression data, International Conference on Applied Algorithms, Springer, 2014, pp. 245–256.
[33] Y.H. Yang, S. Dudoit, P. Luu, T.P. Speed, Normalization for cDNA microarry data, Microarrays: Optical Technologies and Informatics, vol. 4266, International Society for Optics and Photonics, 2001, pp. 141–153.
[34] J.L. Sevilla, et al., Correlation between gene expression and go semantic similarity, IEEE ACM Trans. Comput. Biol. Bioinf 2 (Oct-Dec. 2005) 330–338.
[35] S.S. Ray, S. Misra, A supervised weighted similarity measure for gene expressions using biological knowledge, Gene 595 (2) (2016) 150–160.
[36] F. Glover, M. Laguna, Tabu search, Handbook of Combinatorial Optimization, Springer, 1998, pp. 2093–2229.
[37] J. Kennedy, Particle swarm optimization, Encyclopedia of Machine Learning, Springer, 2011, pp. 760–766.
[38] M. Dorigo, M. Birattari, Ant colony optimization, In Encyclopedia of Machine Learning, Springer, 2011, pp. 36–39.
[39] S.S. Ray, S. Bandyopadhyay, S.K. Pal, Genetic operators for combinatorial optimization in tsp and microarray gene ordering, Appl. Intell. 26 (3) (2007) 183–195.
[40] M.B. Eisen, P.T. Spellman, P.O. Brown, D. Botstein, Cluster analysis and display of genome-wide expression patterns, Proc. Natl. Acad. Sci. Unit. States Am. 95 (25) (1998) 14 863–14 868.
[41] Z. Bar-Joseph, D.K. Gifford, T.S. Jaakkola, Fast optimal leaf ordering for hierarchical clustering, Bioinformatics 17 (1) (2001) S22–S29.
[42] T.H. Bø, B. Dysvik, I. Jonassen, LSimpute: accurate estimation of missing values in microarray data with least squares methods, Nucleic Acids Res. 32 (3) (2004) 1–8.
[43] S.S. Dwight, et al., Saccharomyces Genome Database (SGD) provides secondary gene annotation using the Gene Ontology (GO), Nucleic Acids Res. 30 (1) (2002) 69–72.
[44] H.-W. Mewes, C. Amid, R. Arnold, D. Frishman, U. Güldener, G. Mannhaupt, M. Münsterkötter, P. Pagel, N. Strack, V. Stümpflen, MIPS: analysis and annotation of proteins from whole genomes, Nucleic Acids Res. 32 (suppl_1) (2004) D41–D44.
[45] D.E. Goldberg, K. Deb, A comparative analysis of selection schemes used in genetic algorithms, Foundations of Genetic Algorithms, 1 1991, pp. 69–93.
[46] M. Srinivas, L.M. Patnaik, Genetic algorithms: A survey, Computer 27 (6) (1994) 17–26.
[47] R. Poli, W.B. Langdon, Genetic programming with one-point crossover, in: P.R. Chawdhry, K. P, R. Roy (Eds.), Soft Computing in Engineering Design and Manufacturing, Springer, 1998, pp. 180–189.
[48] S.N. Sivanandam, S.N. Deepa, Introduction to Genetic Algorithms, Springer Science & Business Media, 2007.
[49] Y. Dodge, Statistical Data Analysis Based on the L1-norm and Related Methods, Springer, Birkhuser, Basel, 2012.
[50] P. Guerreiro, C. Rodrigues-Pousada, Disruption and phenotypic analysis of six open reading frames from chromosome VII of Saccharomyces cerevisiae reveals one essential gene, Yeast 18 (9) (2001) 781–787.
[51] T.V. Vo, et al., A proteome-wide fission yeast interactome reveals network evolution principles from yeasts to human, Cell 164 (1) (2016) 310–323.
[52] S.-T. Liou, M.-Y. Cheng, C. Wang, SGT2 and MDY2 interact with molecular chaperone YDJ1 in Saccharomyces cerevisiae, Cell Stress & Chaperones 12 (1) (2007) 59–70.

[53] B. Bukau, A.L. Horwich, The Hsp70 and Hsp60 chaperone machines, Cell 92 (3) (1998) 351–366.
[54] K. Lee, M.-K. Sung, J. Kim, K. Kim, J. Byun, H. Paik, B. Kim, W.-K. Huh, T. Ideker, Proteome-wide remodeling of protein location and function by stress, Proc. Natl. Acad. Sci. Unit. States Am. 111 (30) (2014) E3157–E3166.
[55] B.P. Tu, A. Kudlicki, M. Rowicka, S.L. McKnight, Logic of the yeast metabolic cycle: temporal compartmentalization of cellular processes, Science 310 (5751) (2005) 1152–1158.
[56] S. Jadhav, S. Russo, S. Cottier, R. Schneiter, L.A. Cowart, M.L. Greenberg, Valproate induces the unfolded protein response by increasing ceramide levels, J. Biol. Chem. 291 (42) (2016) 22253–22261.
[57] P. Resnik, Using Information Content to Evaluate Semantic Similarity in a Taxonomy, (1995) cmp-lg/9511007.
[58] J.J. Jiang, D.W. Conrath, Semantic similarity based on corpus statistics and lexical taxonomy, Proceedings of International Conference Research on Computational Linguistics, Taiwan, 1997.
[59] D. Lin, An information-theoretic definition of similarity, ICML, vol. 98, 1998, pp. 296–304.
[60] H. Al-Mubaid, A. Nagar, Comparison of four similarity measures based on go annotations for gene clustering, IEEE Symposium on Computers and Communications, 2008, pp. 531–536.