This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2014.2316508, IEEE Journal of Biomedical and Health Informatics

Integrative clustering by non-negative matrix factorization can reveal coherent functional groups from gene profile data

Sanja Brdar, Vladimir Crnojević, Blaž Zupan

***Abstract*—Recent developments in molecular biology and techniques for genome-wide data acquisition have resulted in abundance of data to profile genes and predict their function. These data sets may come from diverse sources and it is an open question how to commonly address them and fuse them into a joint prediction model. A prevailing technique to identify groups of related genes that exhibit similar profiles is profile-based clustering. Cluster inference may benefit from consensus across different clustering models. In this paper we propose a technique that develops separate gene clusters from each of available data sources and then fuses them by means of non-negative matrix factorization. We use gene profile data on the budding yeast *S. cerevisiae* to demonstrate that this approach can successfully integrate heterogeneous data sets and yields high-quality clusters that could otherwise not be inferred by simply merging the gene profiles prior to clustering.**

***Index Terms*—Clustering, Data fusion, Gene profiling, Gene set enrichment, Non-negative matrix factorization**

I. Introduction

MODERN experimental approaches in molecular systems biology provide us with data that are rich in the number of observed objects (e.g., genes) and in the conditions where these are studied. Today, a major challenge to exploit available data is addressed by crafting of computational approaches that can propose potentially useful hypotheses from the ever-increasing volume of data repositories and heterogeneity of data sources.

A common task in molecular biology is gene function prediction. We can exploit currently available functional annotations in model organisms in combination with various source of experimental data to infer functions of yet uncharacterized genes. A popular approach for this task is gene clustering [1]. Clustering infers groups of similarly-profiled genes. The experimental data that characterizes genes is considered for the assessment of gene similarity and the function of uncharacterized genes is inferred from the prevailing function of the genes in the cluster. This "guilt by association" principle assumes that gene clusters are also functionally enriched, that is, genes with similar functions will cluster together, making the clusters coherent in terms of functions carried out by genes in the cluster.

Large-scale molecular biology experiments may provide the data for profiling thousands of genes. These profiles may include condition- or development stage-specific gene expressions, mutant-based phenotypes such as growth rates or measurements of fitness, and gene interactions. Profiles that stem from different types of experiments may result in gene clusters of different coherence and hence different utility for gene function prediction. An open question is how to integrate the results of clustering coming from different types of gene profiles to increase the quality of clusters with respect to enrichment of their associated gene functions.

In bioinformatics, integrative approaches are motivated by the desired improvement of robustness, stability and accuracy. Troyanskaya *et al.* introduced a Bayesian integrative framework [2], [3], [4] that examines information from various data sources. Each data source provides information to independently estimate the likelihood that a pair of genes is functionally related. These likelihoods are then merged across data sources via the Bayesian approach. The structure of the Bayesian network and conditional probability tables are often obtained from domain experts or inferred from Gene Ontology (GO) [5]. A related, but methodologically different unsupervised approach to data integration was proposed by Tanay *et al.* [6], where biclustering of genes and their characteristics led to identification of groups of genes with correlated behavior across diverse data sources. The approach proposed in this paper is motivated by consensus clustering [7], a method that originally incorporates resampling to yield diverse data sets of which clustering is a subject to consensus analysis to find groups of genes that consistently co-cluster across data samples. Consensus clustering increases the stability of discovered clusters.

Instead of resampling employed in consensus clustering, we propose to examine gene clusters that are developed from different data sources and different similarity measures. We further propose an alternative technique for cluster integration, where we use non-negative matrix factorization (NMF) [8]. Approaches based on NMF have become widely accepted for the analysis of bioinformatics data [9] and useful tools have emerged [10], [11]. NMF has been applied to reduce dimensions in microarray data and infer reduced features metagenes

Manuscript received on August 31, 2013; revised on November 14, 2013 and February 8, 2014; accepted on March 31. This work was supported by the Serbian Ministry of Education and Science through project III 43002 (SB, VC), and by grants from the Slovenian Research Agency (P2-0209, J2-5480), European Commission (CARE-MI Health-F5-2010-242038), and National Institutes of Health's Program Project PO1 HD39691 (BZ).

S. Brdar and V. Crnojević are with Faculty of Technical Sciences, University of Novi Sad, Trg Dositeja Obradovića 6, 21000 Novi Sad, Serbia. E-mail: {brdars, crnojevic}@uns.ac.rs.

B. Zupan is with Faculty of Computer and Information Science, University of Ljubljana, Tržaska 25, SI-1000 Ljubljana, Slovenia and with Baylor College of Medicine, Department of Molecular and Human Genetics, 1 Baylor Plaza, 77030 Houston, Texas, USA. E-mail: Blaz.Zupan@fri.uni-lj.si.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2014.2316508, IEEE Journal of Biomedical and Health Informatics

that were then used for clustering and visualization [12]. In another example, Wang *et al.* [13] reduced data dimensions by least squares NMF. The authors observed improved results when uncertainty measurements of gene expression data were incorporated in the algorithm. Zheng et al. used NMF for clustering cancer gene expression data [14]. A Specific NMF application was reported by Greene *et al.* [15], where the authors proposed to ensemble non-negative matrix factorizations of proteins pairwise similarity matrices, each obtained with different random initialization of the method. In a text mining study, Chagoyen [16] developed a corpus of gene-relevant documents and relied on NMF to transform the initial high dimensional vocabulary space into reduced semantic representation. Hierarchical clustering was then used to group genes in the new feature space. Discovered groups were functionally coherent, but the authors limited the evaluation to only eight GO terms.

We here describe the study that proposes data integration through gene clustering on possibly heterogeneous data sets and cluster fusion by means of NMF. We show that proposed integration increases cluster coherence estimated through gene function enrichment [17]. The clusters discovered through integration are more representative as they include higher proportion of genes that share common function. We also diversify input data by considering various estimates of gene profile similarity. Integrative approach allows us to better handle noise and other uncertainties by generalizing across multiple data sources. In our study, gene clusters are inferred from gene networks [18] [19] [20], where these can directly represent the original data (for example, for interactions between genes or between proteins) or can be constructed from gene profile data applying some profile similarity measure. For clustering, we use a state-of-the-art network-based algorithm SPICi (Speed and Performance In Clustering) [21] and two well-known Markov Cluster [22] and Affinity Propagation [23] algorithms. Different clustering algorithms provided us an opportunity to study their effects on quality of data fusion. The main contributions of our work include the proposed data fusion, an algorithm for extracting final clusters after NMF, and evaluation of proposed data fusion technique within the scope of functional genomics [24], [25] [26].

II. Data

We considered three different data sets on budding yeast (*Saccharomyces cerevisiae*) that include a collection of gene expressions measured at 36 different time points of the metabolic cycle [24] (YMC), gene interaction data from SGA experiments [25], and gene expression data sets from the Saccharomyces Genome Database - Expression Connection (SGD) [26]. SGA interaction data profiles 3,475 query genes by recording a fitness of a double mutant, where each of the query genes was knocked-out together with another gene chosen from the set of 1,712 genes. In gene expression data from SGD we have merged various SGD data subsets to derive profiles of genes whose expression was observed under 740 different conditions. The selected data collections include different sets of genes; we focused on the subset of 1,799 genes that were present in all three data sources.

III. Methods

A. Inference of Gene Networks

We inferred gene networks from gene profile similarities and considered three alternative measures: mutual information, Pearson correlation coefficient and Euclidean distance. Each inferred network is an undirected weighted graph $G = (V, E, w)$, where V is the set of nodes (genes), $E \subseteq V \mathrm{x} V$ is the set of edges and w are edge weights that refer to estimated similarity. In the case of mutual information and Pearson correlation, two nodes are connected if the profile similarity between their corresponding genes is above the 99^{th} percentile of similarities from ten thousand arbitrarily chosen gene pairs from randomly perturbed data (c.f., [18]). For Euclidean distance, significant edge weights are those below 25^{th} percentile of estimated null-hypothesis distribution. Initial threshold that selects edges below the 1^{st} percentile was too restrictive and would result in a loss of more than half of networks nodes that became singletons after thresholding.

After the thresholding described above the resulting gene networks still include about half a million edges and are too dense for identification of groups by graph-based clustering. Hence, we have additionally removed the edges by retaining at most 100 highest-scored edges for each gene. The choice of this threshold was inspired from results of the studies of yeasts co-expression networks in [27], [28] which exhibit small-world and scale-free typologies with high modularity. The degrees of our resulting metabolic, expression and SGA networks along with the other main properties of inferred graphs are reported in the Table I. Analysis was carried out with the Network Analyzer [29] plug-in for the Cytoscape [30]. These properties are similar to those of the co-expression networks from [28] where clustering coefficient was 0.2 and diameter was 3, and are similar to properties of the networks from [27], where the average node degree was 73.4.

B. Clustering Algorithms

The SPICi [21] network clustering algorithm searches for highly connected regions in the network and uses a greedy heuristic approach. It calculates the density of sub-network $S \subset G$ as the sum of the weights of all edges in S divided by the total number of possible edges that would be present in a complete sub-graph. Another measure used in SPICi is node support provided by a sub-network S, which is defined as the sum of the weights of edges that are incident to nodes in S. The algorithm starts with nodes of the highest-weighted edge and grows the cluster based on two parameters: Ts - the support threshold and Td - the density threshold. The number of clusters is determined by the algorithm. After clustering, some nodes remain as singleton clusters due to their relatively low similarity with adjacent nodes and they are discarded at the end of the process. Our networks were clustered with parameter Ts set to 0.5 and Td adapted to the network properties. The starting value was set to 0.5 and was decreased until coverage, expressed as the ratio between the number of genes included in the clusters and the total number of genes, reached at least 50 % of genes.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2014.2316508, IEEE Journal of Biomedical and Health Informatics

TABLE I: Statistical Properties of Inferred Networks

Data Set	Similarity Score	Number of Nodes	Average Degree	Clustering Coefficient	Network Diameter
YMC	Mutual Inf.	1798	42.32	0.23	6
	Pearson	1797	41.38	0.32	7
	Euclidean	1788	62.74	0.53	14
SGA	Mutual Inf.	1799	76.85	0.07	3
	Pearson	1799	73.36	0.09	3
	Euclidean	1799	67.19	0.17	5
SGD	Mutual Inf.	1799	35.20	0.21	6
	Pearson	1797	31.36	0.24	7
	Euclidean	1428	33.82	0.33	14

The Markov Clustering (MCL) [22] algorithm uses random walks and assumes that longer network paths are more likely to occur for a pair of associated nodes . The algorithm starts with an adjacency matrix that represents a weighted graph, where the diagonal elements are added to include self-loops. The matrix is transformed to a stochastic transition matrix where each column sums to one. After this, expansion and inflation operators are applied in iterative steps. Expansion corresponds to the power of a matrix and provides higher step transition probabilities. The inflation operator takes entry-wise powers with coefficient r and it is followed by re-scaling to keep the matrix stochastic. This operator emphasizes strong connections and further weakens already weak ones. Inflation parameter r affects clustering granularity. In our experiments, we start clustering with r set to 2.0. If the algorithm produced oversized clusters with more than 300 genes, inflation parameter r was increased. For SGA/Mutual information, SGA/Euclidean and YMC/Euclidean networks this parameter was set to 2.0, 2.5 and 4.0, respectively. For all others networks, $r = 2.2$ fulfilled this condition and provided good quality and coverage of clusters. In the initialization step, self-loops were assigned to the graph with weights that equal the maximum weight of incident edges for each node [31]. Compared to the case where the self-loop is left at zero or equal to the sum of incident weights, this setting produced better results in terms of the higher gene function enrichment scores.

The third algorithm, Affinity Propagation [23] (AP), searches for representative nodes (so-called exemplars) that provide seeds for clusters. Seeds are chosen to maximize within-cluster similarities. Nodes exchange messages on availability and responsibility. Responsibility $r(i,k)$ is sent from non-representative nodes to exemplars and inform on the suitability of exemplar k for node i, considering other potential exemplars. Availability $a(i,k)$ is sent from exemplar k to data point i to inform it on how appropriate it would be for point i to choose k as its exemplar. Messages trigger actions on choice of cluster membership, and are exchanged until reaching convergence. The number of exemplars (clusters) emerges through the use of a clustering algorithm.

C. Integration by Non-negative Matrix Factorization

The result of network clustering from different data set/similarity measure combinations can be presented as a matrix of cluster memberships [32], where one dimension represents genes and the other clusters. Cluster memberships by SPICi, AP and MCL are all crisp and the values in membership matrix are either 1 or 0, indicating whether a gene was assigned to a specific cluster. Clustering information from different data sources were merged by concatenating membership matrices in the cluster dimension to obtain the joint cluster membership matrix $R = \{0,1\}^{m\times n}$, where m is the total number of clusters from all clusterings and n is the number of genes considered. NMF finds an approximation $R \approx WH$, where W and H are two non-negative factors such that $W \in \mathbb{R}^{m\times k}$ and $H \in \mathbb{R}^{k\times n}$. Parameter k is a factorization rank and equals to the desired (target) number of clusters. In the resulting factorization the matrix W contains encoding coefficients while rows of H are the basis vectors that can be interpreted as (continuous) memberships to target clusters discovered by factorization.

NMF used an algorithm with multiplicative updates [33]. Since our input matrix is sparse, multiplicative updates also provide sparse solutions and there is no need to include regularization into the process of factorization. Values of H and W are iteratively updated (Eqs. 1 and 2) by multiplying the current values with the factors that depend on the quality of the approximation $R \approx WH$:

$$H \leftarrow H. * ((W^T R)./(W^T W H)), \tag{1}$$

$$W \leftarrow W. * ((R H^T)./(W H H^T)). \tag{2}$$

Under the multiplicative updates, approximation of R improves monotonically in the Frobenius norm of reconstruction error:

$$\|R - WH\|_F^2 = \sum_i \sum_j [R_{ij} - (WH)_{ij}]^2 \tag{3}$$

The optimization starts with matrices W and H computed by non-negative double singular value decomposition (NNDSVD) [34], speeding up the convergence of the optimization and supporting the reproducibility of the results.

The cluster reconstruction process involves setting the threshold on gene cluster memberships. Fig. 1 illustrates NMF decomposition of an example cluster membership matrix. For thresholding, we implement a scaling procedure described below. Namely, the results of non-negative matrix factorization are not necessary unique. There may exist nonsingular matrices $D \in \mathbb{R}^{k\times k}$ that satisfy $WD \geq 0$ and $D^{-1}H \geq 0$, and we can rewrite factorization as:

$$WH = WDD^{-1}H = W^* H^* \tag{4}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2014.2316508, IEEE Journal of Biomedical and Health Informatics

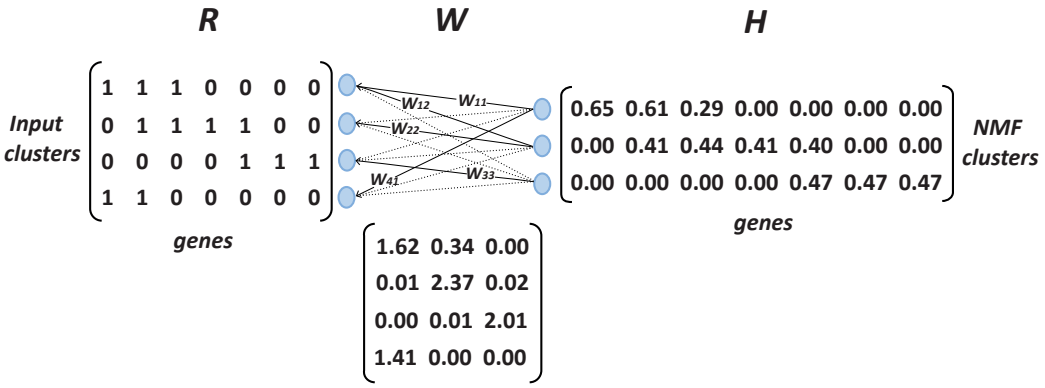


Fig. 1: Example of NMF decomposition. The original matrix with crisp memberships to four clusters R is transformed to new membership matrix H with three clusters and fuzzy memberships.

Matrix D can perform transformations such as scaling or permutation. Difficulty in determination of new clusters comes from a scale variance. Instead of factorization presented in Fig. 1 which results in a pair of coefficients $w_{3,3}$ = 2.10 and $h_{3,5}$ = 0.47, NMF can also result in $w_{3,3}$ = 1.82, $h_{3,5}$ = 0.54 (other values in W and H are also changed). Therefore, it would not be appropriate to assign an absolute threshold value for creation of new clusters. In order to eliminate encoding variations we rescaled the columns of encoding matrix W and rows of basis matrix H, and use the following two diagonal matrices D_W and D_H:

$$D_W = diag([max(w_{:,1}), max(w_{:,2})...max(w_{:,k})]) \quad (5)$$

$$D_H = diag([max(h_{1,:}), max(h_{2,:})...max(h_{k,:})]) \quad (6)$$

Part of the procedure used in binary matrix factorization [35] was suitable for rescaling obtained W and H. For matrices D_W and D_H, the following relations hold:

$$D_W = D_W^{1/2} D_W^{1/2} \quad D_H = D_H^{1/2} D_H^{1/2}$$
$$D_W^{-1} = D_W^{-1/2} D_W^{-1/2} \quad D_H^{-1} = D_H^{-1/2} D_H^{-1/2} \quad (7)$$

$$\tilde{R} = WH = (WD_W^{-1})(D_W D_H)(D_H^{-1} H)$$
$$= (WD_W^{-1/2} D_H^{1/2})(D_H^{-1/2} D_W^{1/2} H) \quad (8)$$

From Equation 8 rescaling matrix D can be expressed as $D = D_W^{-1/2} D_H^{1/2}$:

$$W^* = WD_W^{-1/2} D_H^{1/2} \quad H^* = D_H^{-1/2} D_W^{1/2} H \quad (9)$$

Transformations of W and H into W^* and H^* keep product WH unchanged, but ensure that values in the encoding and basis matrices are comparable and can be interpreted. Each element of W and H is rescaled in the following manner:

$$w^*_{i,k} = w_{i,k} \sqrt{\frac{max(h_{k,:})}{max(w_{:,k})}} =$$
$$\frac{w_{i,k}}{max(w_{:,k})} \sqrt{max(w_{:,k}) max(h_{k,:})} \quad (10)$$

$$h^*_{k,j} = h_{k,j} \sqrt{\frac{max(w_{:,k})}{max(h_{k,:})}} =$$
$$\frac{h_{k,j}}{max(h_{k,:})} \sqrt{max(h_{k,:}) max(w_{:,k})} \quad (11)$$

We infer the membership to k new clusters from coefficients in W^* and H^* in either overlapping or exclusive manner. In overlapping clustering, genes may belong to more than one cluster, while in exclusive clustering, each gene is assigned only to one, most likely cluster. Overlapping clustering assigns genes to clusters according to their membership coefficients in H^*, but only if the membership exceeds the threshold of 0.5. For exclusive clustering, additional ranking is used that takes into account the importance of a gene within cluster and strength of cluster. Importance is derived from H^* and strength from W^*. The ranking algorithm can be summarized by the pseudo code given in Algorithm 1.

Algorithm 1: Extraction of clusters

Inputs: $W^* \in \mathbb{R}^{M \times K}$, $H^* \in \mathbb{R}^{K \times N}$, genes [$g_1$, g_2, ... g_N], T_r = 0.5
Outputs: clusters C = [c_1, c_2, ... c_K]
$WSUM^* \leftarrow$ sum over columns W^*
for $k \leftarrow 1 : K$ **do**
  for $j \leftarrow 1 : N$ **do**
    if clustering = overlapping **then**
      if $h^*_{k,j} \geq T_r$ **then**
        append cluster c_k with gene g_j
      end if
    else
      if ($h^*_{k,j} \geq T_r$) and ($h^*_{k,j} * wsum^*_k$ = max($h^*_{k',j} * wsum^*_{k'}$, for $k' \leftarrow 1 : K$)) **then**
        append cluster c_k with gene g_j
      end if
    end if
  end for
end for

Factorization of the input matrix R is iterative and runs for 500 iterations. This is also the number of iterations that is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2014.2316508, IEEE Journal of Biomedical and Health Informatics

required for to reach a stable results in terms of a clustering structure in number of clusters and involved genes.

D. Cluster Scoring

Any useful clustering should infer gene groups that are coherent in terms of gene function or any other observed gene properties. To test this aspect of the method, we use gene annotations from Gene Ontology [5] (GO) and focus on its 92 yeast slim terms that represent the major branches of the GO. We assume that the quality of the cluster is associated with the enrichment of a subset of slim terms in the annotations of genes from the clusters. Term enrichment, expressed through a p-value, was computed with a hypergeometric test that assesses the probability that, for a particular GO term, the abundance of term-annotated genes in the cluster is not the result of chance. Intuitively, the clusters with no enriched terms are not useful for function prediction and hence are of poor quality. In general, good clusters may have several slim terms that are enriched. Improvements in clustering algorithm should yield clusters with increased proportion of genes that share common function, and thus exhibit higher function enrichment scores [17]. We therefore score the clusters by averaging $-log$(enrichment p-value) of the three most-enriched slim terms.

IV. Experimental Study and Discussion

This section provides in-depth view on different integration scenarios. The properties of individual clustering used in integrations are outlined in Table II and include number of clusters and coverage - the ratio between clustered and total number of genes. We first describe experiments with this set of input clusterings. Later, we evaluate method on larger set created by altering the parameters that affect clustering properties. In the experiments we have varied the factorization rank k according to the average number of clusters inferred by individual clusterings that participate in the integration (bottom row of Table II). We then used $k \in \{150, 200, 250, 300, 350\}$ for SPICi and $k \in \{100, 150, 200, 250, 300\}$ for the other two methods. In this way we could test the effectiveness of representing new clusters by virtue of merging, splitting and combining input clusters.

A. Partial Integration Across Data Sets or Across Different Network-Specific Similarity Scores

We integrated either a single input data set where the clustering was inferred from similarity networks obtained with application of three different similarity measures, or integrated three different data sets where a single similarity measure was considered. Experimental results of these six integration scenarios are summarized in Fig. 2 and corresponding coverages of integrative clusterings can be followed in Fig. 3. The results demonstrate that integration improves enrichment, as we always observe higher scores for the clusterings after integration. The results also suggest that the efficiency of integrative clustering can be boosted not only by considering the integration of different sources of data, but also by considering different measures of similarity. Comparison with baseline enrichment derived from clustering with the same structure of clusters but arbitrary association of gene cluster membership demonstrates that improvement from initial clustering is truly due to integration and appropriate assignment of genes to the clusters, and is not obtained just by changing the size and number of clusters.

B. Integration of Complete Set of Input Clusterings

In the next experiment we tested the effectiveness of integrating the entire set of nine clusterings where all data sets and all similarity measures were involved. This integration (see Fig. 4a) improves the results over previous models of integration. NMF grouped genes into clusters with an average enrichment score from 6.15 to 8.11 for overlapping clustering, and from 4.91 to 6.19 for exclusive clustering. That is significantly higher than the coherence in original clusters since the best clustering that was involved in this integration (SGD data set, Euclidean measure) has an enrichment score of 4.99. Integrated clusters have higher gene function coherence than clusters that served as an input to the integration.

We further tested the behavior of the proposed data fusion with two other clustering algorithms, MCL and AP. Again, clustering was carried out on networks inferred from all three data sets, where we used each of the three similarity measures. The results (Fig. 4b and 4c) demonstrate better performance of overlapping representative clusters compared to all individual clusterings for both MCL an AP. In the case of MCL, the quality of exclusive representative clusters outperforms all individual clusterings when k is set to 100 and 150 and it is at the level of the best used in integration when k is 200. When we increase granularity (250 and 300 clusters), the integrative approach performs slightly worse, with enrichment scores that are still higher than in seven out of nine individual clusterings. In the case of AP, our method

TABLE II: Properties of Individual Network-Based Clusterings (Inputs to Integration)

	Similarity	SPICi		MCL		AP	
Data Set	Score	Clusters	Coverage	Clusters	Coverage	Clusters	Coverage
YMC	Pearson	221	0.77	197	0.94	185	0.99
	Mutual Inf.	183	0.70	214	0.92	252	0.99
	Euclidean	141	0.81	179	0.95	136	0.99
SGA	Pearson	385	0.76	155	0.73	245	1.00
	Mutual Inf.	307	0.86	174	0.91	280	1.00
	Euclidean	285	0.89	118	0.76	162	1.00
SGD	Pearson	256	0.84	232	0.84	195	0.99
	Mutual Inf.	279	0.73	205	0.85	213	1.00
	Euclidean	170	0.61	176	0.76	175	0.78
Average		247	0.77	183	0.85	205	0.97

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2014.2316508, IEEE Journal of Biomedical and Health Informatics

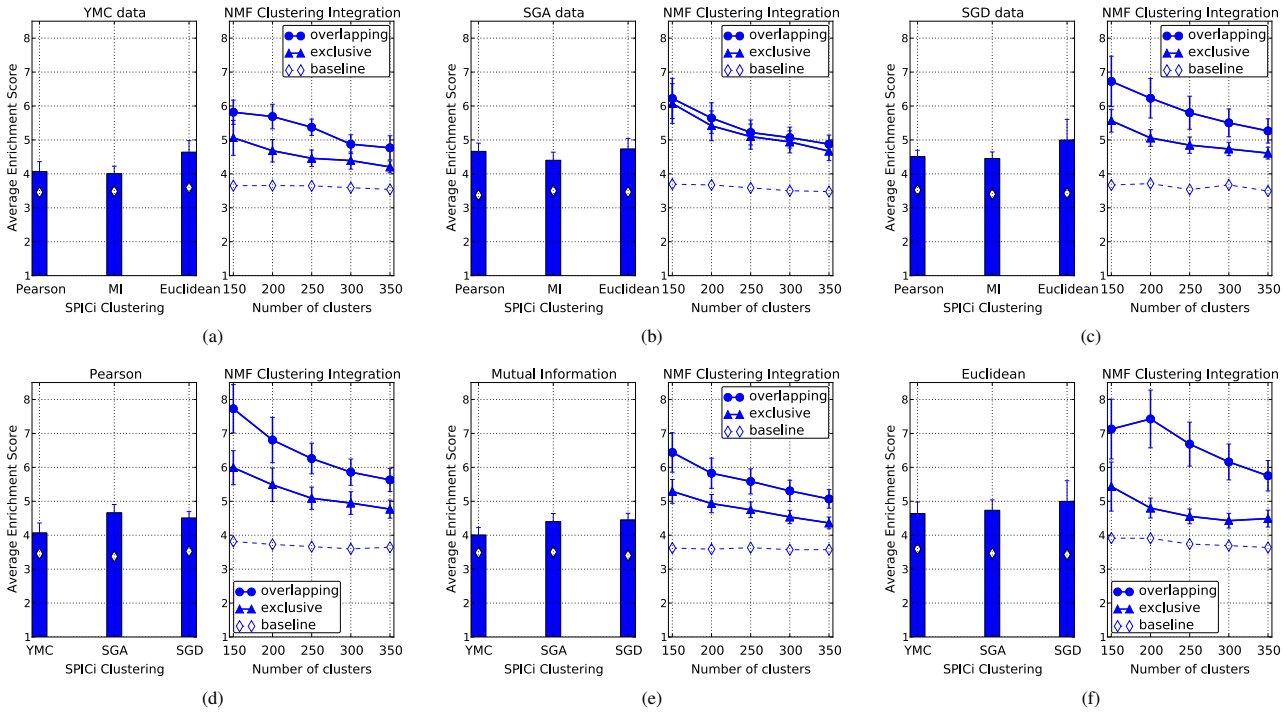


Fig. 2: Comparison of clustering results before and after the integration. The bar charts present the average enrichment scores of SPICi clusters (before the integration), and the line graphs present the enrichment scores after the NMF integration with both overlapping and exclusive clusters at five granularity levels (k). Each panel shows result for specific integration scenario: (a) YMC data x 3 measures, (b) SGA data x 3 measures, (c) SGD data x 3 measures, (d) 3 data sets x Pearson Correlation, (e) 3 data sets x Mutual Information, (f) 3 data sets x Euclidean distance. In all cases the NMF integration results in increased enrichment scores and with this improved quality of clusters. The enrichment scores are compared to the baseline scores (diamond symbol on bars and dashed lines) inferred from clustering with random assignment of genes to the clusters. The graphs provide baseline scores for clustering before integration (bar charts) and for overlapping NMF clustering (line charts); the baseline scores for exclusive clustering were slightly lower and are not shown.

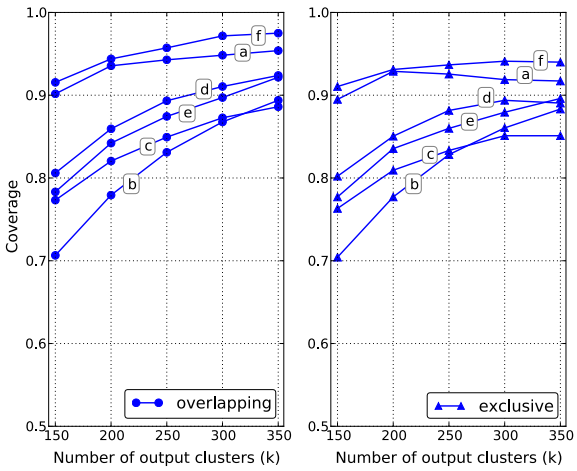


Fig. 3: Coverage of genes as a function of the number of output clusters k. The figure reports on the coverage of overlapping (left) and exclusive NMF clusters (right) from six experiments presented in Fig. 2. Letters on the lines in the graph (from a to f) refer to panels with different integrations scenarios from Fig. 2.

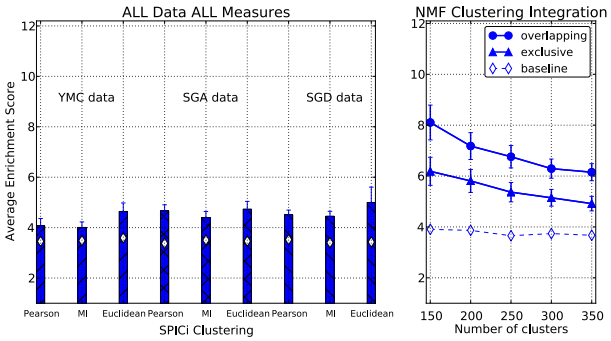
is able to successfully transform input clusters in 100 and 150 exclusive representative clusters. If we additionally increase granularity when creating representative clusters, the quality of the resulting system declines.

C. Choice of the Number of Clusters with Respect to its Effect on Average Accuracy and Gene Coverage

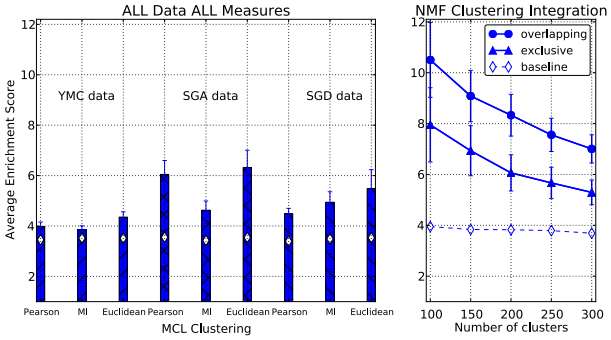
Both average enrichment and gene coverage depend on the choice of the number of output clusters k. Results suggest that both scores improve after integration. For instance, the average number of input SPICi clusters was 247 with gene coverage of 0.77 (Table II, bottom row). At similar number of clusters ($k = 250$), the integration — especially the one with overlapping clusters — improves the average enrichment score (Fig. 2) but has also higher coverage (Fig. 3).

To further study this two-fold benefit of integration, and isolate its dependency on number of clusters, we altered the parameters of our network clustering methods that provide for initial clustering. Our aim was to infer a cluster sets with specific number of input clusters, and then output the same number of clusters after the integration. SPICi ($k = 150$) and MCL ($k = 100$) clustering were considered, as AP clustering is parameter-free. Shrinking the number of clusters when compared to our previous experiments (Table II) slightly improved enrichment for MCL clusters, but had a mixed effect on SPICi-based clusters. Average enrichment score in a set of SPICi-inferred clusters was 4.56 with best individual clustering scoring 5.08, at 0.95 coverage. Integration increased both the coverage to 0.97 and average enrichment score to 5.56 for exclusive, and to coverage of 0.99 and enrichment of 7.93 for overlapping clustering. Average score in a set of clusters by MCL was 5.43 with best individual clustering

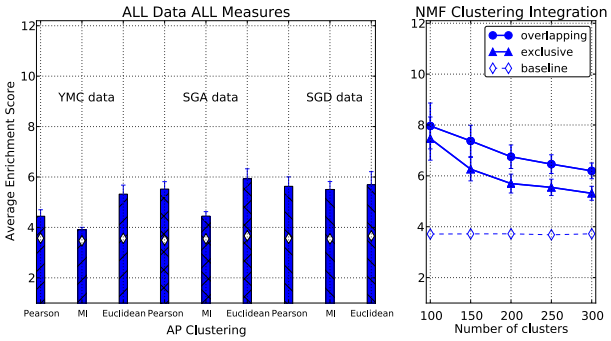
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2014.2316508, IEEE Journal of Biomedical and Health Informatics



(a)



(b)



(c)

Fig. 4: NMF integration of nine clusterings by (a) SPICi, (b) MCL, (c) AP clusterings (3 data sets x 3 measures). In graphs on the left we report on average enrichment scores for clusterings that participate in the integration, and the right part presents average enrichment scores after NMF integration produced at five different granularity levels. Higher enrichment scores indicate better functional coherence of clusters. The enrichment scores after integration are also consistently above the baseline obtained by evaluating random clusterings of the the same clustering structure.

scoring 6.77 at 0.96 coverage, while NMF again increased the coverage and enrichment to 0.99 and 7.97 for exclusive and to 1.00 and 9.65 for overlapping clustering, respectively. This set of experiments further confirms the utility of integration by increasing both average enrichment and coverage. We have obtained qualitatively similar results with cluster reduction by pruning of the smallest clusters in the input clusterings (results not presented for brevity).

The number of clusters k after the integration is a user-specified parameter. When k is small, the effect of integration is stronger, while for higher values of k the initial clusters may be split to smaller ones. The choice of parameter k involves considering the trade-off between enrichment scores and coverage, and may depend on the goals of particular application. For an appropriate starting choice we recommend setting the number of clusters to the average number of clusters in the input set of clusterings. Our experiments suggest that under such setup the clustering integration already has a positive effect by increasing both enrichment scores and coverage.

D. Further Insight into the Effects of Cluster Integration

To further demonstrate the inner workings of the proposed approach, we provide an illustration obtained from our experiment with integration of nine clusterings (3 data sets x 3 similarity measures). Fig. 5 shows part of the input matrix R considered by NMF. Matrix columns correspond to genes and rows to clusters. Information on the data source and corresponding similarity scoring is provided in the last column of the matrix. In the figure, we provide details on two initial clusters c_1 and c_{21} (the first and the last row) that are the best among the 21 presented and compare them with the output clusters after NMF transformation. For each of the clusters we have analyzed we report on the most enriched GO terms. Since only a subset of genes is shown in the figure, we print in black the cluster memberships that comprise only the genes present in the displayed matrix, and in gray those that also comprise some genes outside the displayed matrix. Notice how NMF reorganizes clusters. Based on the supported evidence, NMF prunes initial clusters and creates functionally more consistent groups. For 33 genes in Fig. 5 assigned to 21 input clusters, NMF identified two clusters that are related to this particular set of genes. Genes CAT2, TCB3, YML131W, YNR014W, HXK2, MTO1, SIS2 and YIR024C were excluded from these clusters due to obvious lack of supporting evidence. CAT2 shares label peroxisome - prevailing function assigned to $c1$, but except that cluster none of the other input clusters uphold its connection to genes that remained clustered together after NMF. We have further examined other clusters that included CAT2. Interestingly, this gene was assigned to another group also enriched in peroxisome, but additionally associated with cellular amino acid and derivative metabolic process. Through other NMF clusters, YML131W was additionally associated with membrane, HXK2 and MTO1 with cytoplasm and mitochondrion, SIS2 with enzyme regulator activity and YIR024C with mitochondrion. TCB3 was not assign to any NMF cluster due to small support, only YNR014W was in cluster were did not contribute to the enrichment score. Output clusters with assigned functional labels indicate that not only is the NMF approach able to identify representatives among input clusters, but also succeeds in further improving them.

E. On Initialization of Matrix Factorization Procedure

Although there is no guarantee that NMF with multiplicative updates converges to global optimum, obtained solutions proved useful and improved clustering results. Through the use of deterministic initialization by NNDSVD [34], our procedure always converges to the same solution. Alternatively, we could use a random initialization of matrices W and H. To examine the differences with deterministic initialization in terms of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2014.2316508, IEEE Journal of Biomedical and Health Informatics

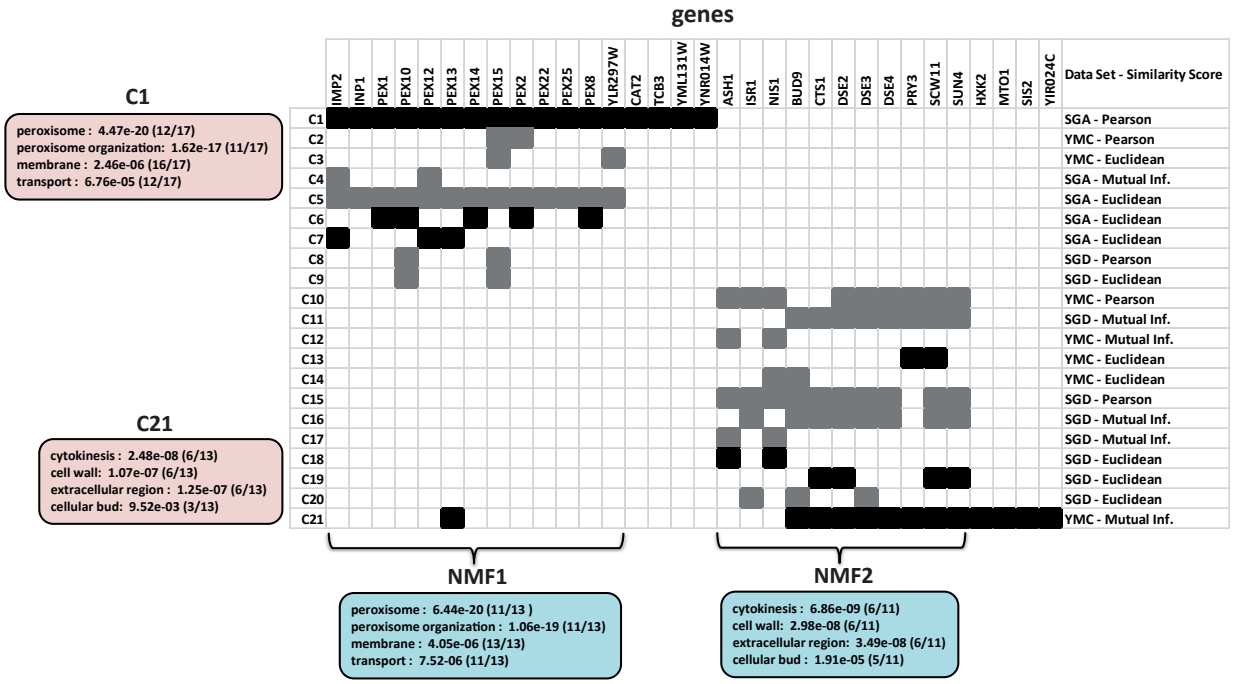


Fig. 5: Integration of information through NMF discovers more meaningful clusters. The figure shows a fragment of integrated cluster membership matrix. The black colour indicates that the fragment of matrix encompasses all members of the cluster, and the grey colour indicates that cluster includes other genes besides those presented. To compare the results we assigned corresponding enriched functional terms to two input clusters (the best in this example) and to output clusters (obtained through NMF framework). Improved enrichment values demonstrate the benefits of the integrative approach.

quality of resulting clusters, we ran 50 experiments with random initialization for 6 integration scenarios from Fig. 2. Results (Fig. 6) indicate that both initialization techniques lead to data integration of similar quality. In some cases random initialization may yield better results and hint at potential utility of assembling of randomly-initialized models. However, considering substantially increased computational requirements of such procedure, we therefore prefer a faster, deterministic, and, as shown in our study, useful initialization by NNDSVD.

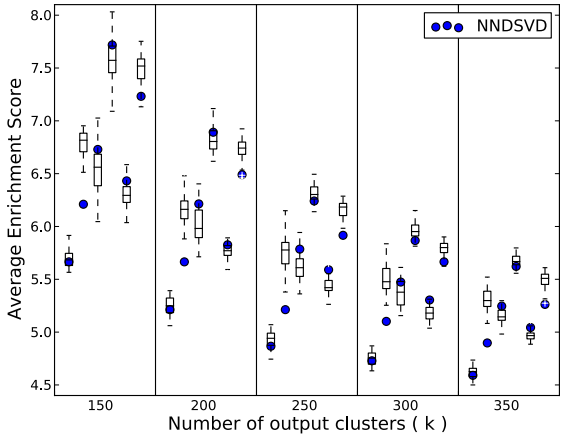


Fig. 6: Comparison of matrix factorization initialization by NNDSVD and random initialization across six different integration scenarios from Fig. 2 and using five different factorization ranks (k). Initialization by NNDSVD is deterministic and using it our data integration procedure converges to a unique solution (blue dots). Results of 50 runs of data integration by random initialization are summarized with box-plots.

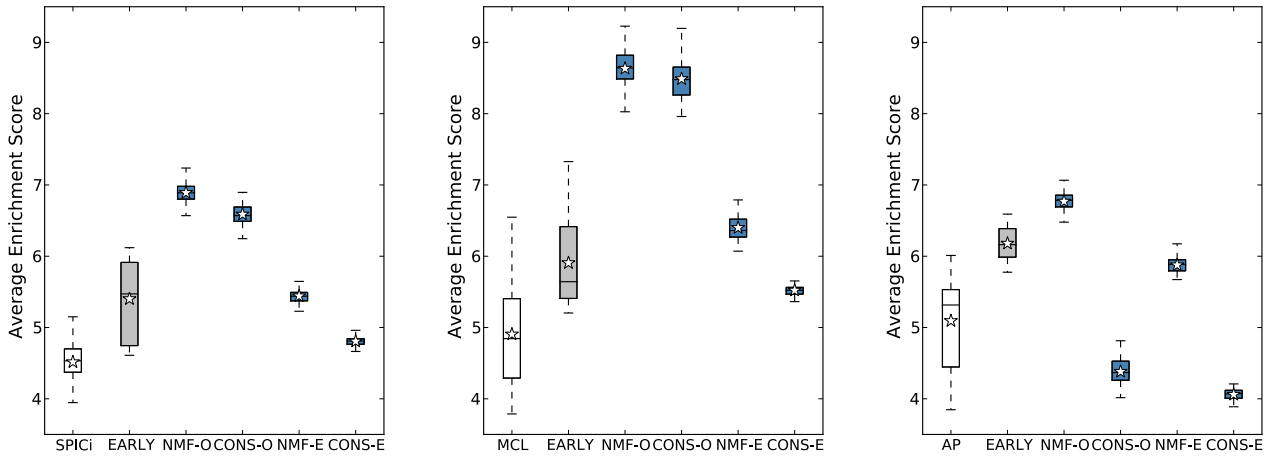
F. On Overlapping vs. Non-Overlapping Cluster Integration

Our proposed integrative method consistently performs better in terms of average enrichment scores when inferring overlapping clusters. This was in part expected as gene annotation terms in general overlap in coverage of the genes, that is, a particular gene may be annotated with more then one term. The problem considered in this paper, that is, finding gene groups with enriched annotations, is therefore biased and benefits from overlapping clustering. We believe that this is with no loss of generality, as many problems from natural sciences deal with objects that are annotated with a set of labels, rather than classified to a single specific class. Being able to infer overlapping clusters should thus be considered a major strength of NMF-based integration. Other studies also indicate that overlapping clustering better address problems in various fields of molecular biology, such as those investigating protein complexes [37], [36] and biological processes [38].

V. Comparison With Other Integration Techniques

Our proposed approach belongs to the late integration type of ensemble techniques, where aggregation is performed after individual clusterings have already been formed. We have compared our method to well-known late integration approach of consensus clustering [7]. Originally proposed for integration of different clusterings obtained from samples of the same data sets, consensus clustering may also be used when different cluster models stem from different data sets or from different preprocessing steps, as in our case. Consensus clustering integrates cluster memberships into a consensus matrix that

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2014.2316508, IEEE Journal of Biomedical and Health Informatics



(a) (b) (c)

Fig. 7: Comparison of clustering integration approaches for initial clustering by SPICi (a), MLC (b) and AP (c). Box plots refer to the baseline approach (no integration, the first box plot in each panel), early integration (EARLY), late integration by NMF (NMF-O for overlapping and NMF-E for exclusive clustering), and consensus clustering (CONS-O for overlapping and CONS-E for exclusive clustering). The length of a box is the interquartile range of the enrichment score distribution, the line across the box represents the median, and the mean is denoted with a star symbol.

can be viewed as a similarity matrix and post-processed through additional methods to obtain final clusters. We used kernel k-means to create exclusive consensus clusters and its soft version to detect overlapping clusters [39]. Soft kernel k-means assigns genes to clusters based on distances to cluster centers. The number of clusters was set to the same level as in the proposed NMF-based integration. Evaluation score for consensus integration in each experiment is averaged across 10 runs due to random initialization of kernel k-means.

A different type of data fusion is an early aggregation, where data is fused before the application of a clustering algorithm by merging gene profiles or by aggregation of similarity matrices [40]. To compare our approach to this technique, we merged gene profiles before clustering and then independently inferred gene similarity networks with all three measures and finally ran individual clustering.

To compare various integration approaches we have first established a collection of different gene networks. We have considered all nine combinations of three data sets and three similarity measures. To additionally diversify the networks, these were pruned so that each node included a maximum of t edges, where $t \in \{80, 85, \ldots 125\}$. Notice that in the previous experiments this parameter was fixed to 100. In this way we have obtained 90 different networks. For the case of early integration, where the data set where first merged, the number of considered networks was 30 (3 similarity measures, 10 choices of t).

Just like in experiments from Fig. 4, we have considered three different clustering methods (SPICi, MCL and AP) to obtain the initial clusters from each of the networks. Fig. 7 reports on the resulting average enrichment scores for the baseline approach (no data integration), early integration (EARLY), and late integration approaches by overlapping and non-overlapping NMF-based integration (NMF-O and NMF-E) and overlapping and non-overlapping consensus integration (CONS-O and CONS-E). Box plots in the figure summarize the average enrichment scores obtained from each of 90 networks for baseline approaches (no data integration, box plots labeled SPICi, MCL, and AP) and scores from clusters from each of 30 networks for early integration. Late integration techniques were run 50 times, each time on a random sample of 9 networks from our collection of 90 networks. For the late integration approaches, box plots in Fig. 7 thus summarize 50 different average enrichment scores. The number of output clusters for each run of late integration methods was set to the average number of clusters in 9 sampled networks.

ANOVA test indicate that significant difference exists among different methods ($p < 10^{-70}$ for all experiments within initial clustering by SPICi, MCL and AP). Post-hoc Tukey test with 99% confidence reveals groups that are significantly different. For integration of clusters proposed by SPICi (Fig. 7.a) the ranking order is (NMF-O, CONS-O, NMF-E, EARLY, CONS-E, SPICi) with corresponding grouping (A, B, C, C, D, E). Groups that do not share the same letter are significantly different. Thus, in results from Fig. 7.a, the score distribution for NMF-O is significantly different than those of other methods, while score distributions of NMF-E and EARLY are different to score distributions of the CONS-E and SPICi but are, between themselves, not significantly different. For integration of clusters proposed by MCL (Fig. 7.b), the ranking is (NMF-O, CONS-O, NMF-E, EARLY, CONS-E, MCL) with corresponding grouping of (A, A, B, C, D, E), and for the integration of AP clusters the ranking is (NMF-O, EARLY, NMF-E, AP, CONS-O, CONS-E) with grouping of (A, B, C, D, E, F). Notice that all types of integration surpasses the clustering where no integration took place, except in experiments with AP where both type of CONS lose in performance. For all three types of initial clustering the best results are achieved by overlapping type of NMF integrative clustering. Scores for NMF-E are higher to those for CONS-E.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2014.2316508, IEEE Journal of Biomedical and Health Informatics

EARLY integration performs comparatively well, but its score depends on an appropriate choice of similarity measure that, in our experience, is the parameter causing high variance in performance of this approach.

VI. Conclusion

Clustering that infers gene groups from their profiles that can be gathered from any of the currently abundant genome-wide experimental techniques is currently one of the most common computational tools in functional genomics. While other more focused and specialized computational approaches exist that could manifest better accuracy by learning from class-labeled data [41], clustering is still the prevailing technique for preliminary and explorative analysis of experimental data in systems biology. Further gains in the quality of discovered clusters may stem from data integration, as different data sources may provide different but complementary insight into the observed system. In this paper we have proposed an integration method that can fuse clusterings stemming from different data sets, different data preprocessing steps or different clustering techniques. The approach based on non-negative matrix factorization is robust and can infer gene groups with high functional enrichment and improved gene coverage. Our proposed method is general and compares favorably to alternative integration approaches.

References

[1] M.B. Eisen, P.T. Spellman, P.O Brown, D. Botstein, "Cluster analysis and display of genome-wide expression patterns", Proc Natl Acad Sci U S A, vol. 95(25), pp. 14863-14868, 1998.

[2] 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*, vol. 100(14), pp. 8348-53, 2003.

[3] O.G. Troyanskaya: "Putting microarrays in a context: integrated analysis of diverse biological data", *Briefings in Bioinformatics*, vol. 6(1), pp. 34-43, 2005.

[4] C.L. Myers, D. Robson, A. Wible, M.A. Hibbs, C. Chiriac, C.L. Theesfeld, K. Dolinski, O.G. Troyanskaya, "Discovery of biological networks from diverse functional genomic data", *Genome Biology* vol. 6(13), pp. R114, 2005.

[5] M. Ashburner et al., "Gene ontology: tool for the unification of biology", *Nature Genetics*, vol. 25, pp. 25–29, 2000.

[6] A. Tanay, R. Sharan, M. Kupiec and R. Shamir, "Revealing modularity and organization in the yeast molecular network by integrated analysis of highly heterogeneous genomewide data", *Proc Natl Acad Sci U S A*, vol. 101(9), pp. 2981, 2004.

[7] S. Monti, P. Tamayo, J.P. Mesirov and T. Golub, "Consensus clustering: A resampling-based method for class discovery and visualization of gene expression microarray data", Functional Genomics Special Issue, *Machine Learning Journal* vol. 52, pp. 91-118, 2003.

[8] D.D. Lee and H.S. Seung, "Learning the parts of objects by non-negative matrix factorization", *Nature*, vol. 401(6755), pp. 788–791, 1999.

[9] K. Devarajan, "Nonnegative matrix factorization: an analytical and interpretive tool in computational biology", *PLoS computational biology*, vol. 4(7), pp. e1000029, 2008.

[10] A. Pascual-Montano, P. Carmona-Saez, M. Chagoyen, F. Tirado, and J.M. Carazo, and R.D. Pascual-Marqui, "bioNMF: a versatile tool for non-negative matrix factorization in biology", *BMC bioinformatics*, vol. 7(1), pp. 366, 2006.

[11] M. Žitnik, and B. Zupan, "Nimfa: A python library for nonnegative matrix factorization", *Journal of Machine Learning Research*, vol 13, pp. 849-853, 2012.

[12] L. Weixiang, Y. Kehong, Y. Datian, "Reducing microarray data via nonnegative matrix factorization for visualization and clustering analysis", *Journal of Biomedical Informatics*, vol. 41(4), pp. 602-606, 2008.

[13] G. Wang, A.V. Kossenkov, M.F. Ochs, "LS-NMF: a modified non-negative matrix factorization algorithm utilizing uncertainty estimates", *BMC Bioinformatics* vol. 7, pp. 175, 2006.

[14] C.H. Zheng, D.S. Huang, L. Zhang, and X.Z. Kong, "Tumor clustering using nonnegative matrix factorization with gene selection", *IEEE Transactions on Information Technology in Biomedicine*, vol. 13(4), pp. 599-607, 2009.

[15] D. Greene, G. Cagney, N. Krogan and P. Cunningham, "Ensemble non-negative matrix factorization methods for clustering protein-protein interactions", *Bioinformatics*, vol. 24(15), pp. 1722-1728, 2008.

[16] M. Chagoyen, P Carmona-Saez, H. Shatkay, J.M. Carazo, and A. Pascual-Montano, "Discovering semantic features in the literature: a foundation for building functional associations", *BMC bioinformatics*, vol. 7(1), pp. 41, 2006.

[17] J.H. Hung, T.H. Yang, Z. Hu, Z. Weng, and C. DeLisi, "Gene set enrichment analysis: performance evaluation and usage guidelines", *Briefings in bioinformatics*, 13(3), pp. 281–291, 2012.

[18] A.J. Butte and I.S. Kohane, "Mutual Information Relevance Networks: Functional Genomic Clustering Using Pairwise Entropy Measurements", *In Proceedings of the Pacific Symposium on Biocomputing 2000*, pp. 418-429, 2000.

[19] J. Ruan, A.K. Dean, W. Zhang, "A general co-expression network-based approach to gene expression analysis: comparison and applications", *BMC Systems Biology*, vol. 4(8), 2010.

[20] A.A. Margolin, I. Nemenman, K. Basso, C. Wiggins, G. Stolovitzky, R. Della Favera, A. Califano, "Aracne: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context",*BMC Bioinformatics*, vol. 7(Suppl 1), pp. S7, 2006.

[21] P. Jiang and M. Singh, "SPICi: a fast clustering algorithm for large biological networks", *Bioinformatics*, vol. 26, pp. 1105-1111, 2010.

[22] A.J. Enright, S. Van Dongen and C.A. Ouzounis, "An efficient algorithm for large-scale detection of protein families", *Nucleic acids research*, vol. 30(7), pp. 1575–1584, 2002.

[23] B.J. Frey and D. Dueck, "Clustering by passing messages between data points", *Science*, vol. 315(5814), pp. 972–976, 2007.

[24] B.P. Tu, A. Kudlicki, M. Rowicka and S.L. McKnight, "Logic of the yeast metabolic cycle: temporal compartmentalization of cellular processes", *Science*, vol. 310(5751), pp. 1152-1158, 2005.

[25] M. Costanzo, et al. "The Genetic Landscape of a Cell", *Science*, vol 327 no. 5964, pp. 425-431, 2010.

[26] The Saccharomyces Genome Database (SGD) Available: http://www.yeastgenome.org

[27] L. Tari, C. Baral, P. Dasgupta, "Understanding the global properties of functionally-related gene networks using the gene ontology", *In Proceedings of the Pacific Symposium on Biocomputing 2005*, pp. 209-20, 2005.

[28] V. van Noort, B. Snel, M.A. Huynen, "The Yeast Coexpression Network has a Small-world, Scale-free Architecture and can be Explained by a Simple Model", *EMBO Reports*, vol. 5(3), pp. 280-284. 2004.

[29] Y. Assenov, F. Ramrez, S.E. Schelhorn, T.Lengauer and M. Albrecht, "Computing topological parameters of biological networks" *Bioinformatics*, vol. 24(2), pp. 282–284, 2008.

[30] M.E. Smoot, K. Ono, J. Ruscheinski, P.L. Wang and T. Ideker, "Cytoscape 2.8: new features for data integration and network visualization", *Bioinformatics*, vol. 27(3), pp. 431–432, 2011.

[31] J. Vlasblom and S. Wodak, "Markov clustering versus affinity propagation for the partitioning of protein interaction graphs", *BMC bioinformatics*, vol. 10(1):99, 2009.

[32] D. Greene "A matrix factorization approach for integrating multiple data views", *Machine Learning and Knowledge Discovery in Databases*, pp. 423–438, 2009.

[33] D.D. Lee and H.S. Seung, "Algorithms for Non-negative Matrix Factorization", *In Proceedings of the Advances in Neural Information Processing Systems*, pp. 556–562, 2000.

[34] C. Boutsidis and E. Gallopoulos, "SVD based initialization: A head start for nonnegative matrix factorization", *Pattern Recognition*, vol. 41(4), pp. 1350–1362, 2007.

[35] Z.Y. Zhang, T. Li, C. Ding, X.W. Ren and X.S. Zhang, "Binary matrix factorization for analyzing gene expression data", *Data Mining and Knowledge Discovery*, vol. 20(1), pp. 28–52, 2010.

[36] T. Nepusz, H. Yu, and A. Paccanaro, "Detecting overlapping protein complexes in protein-protein interaction networks", *Nature methods*, vol. 9(5), pp. 471–472, 2012.

[37] X.F. Zhang, D.Q. Dai, L. Ou-Yang, M.Y. Wu, "Exploring overlapping functional units with various structure in protein interaction networks", vol. 7(8):e43092, *PloS one*, 2012.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2014.2316508, IEEE Journal of Biomedical and Health Informatics

[38] A. Battle, E. Segal, and D. Koller, "Probabilistic discovery of overlapping cellular processes and their regulation", *Journal of Computational Biology*, vol. 12(7), pp. 909–927, 2005.

[39] M. Deodhar, J. Ghosh, "Consensus Clustering for Detection of Overlapping Clusters in Microarray Data", *In Proceedings of the Sixth IEEE International Conference on Data Mining Workshops (ICDMW'06)*, pp. 104–108, 2006.

[40] S. Yu, B. De Moor and Y. Moreau, "Clustering by heterogeneous data fusion: framework and applications", *NIPS workshop on Learning with Multiple Sources*, Whistler, Canada, 2009.

[41] P. Larrañaga et al, "Machine learning in bioinformatics", *Briefings in bioinformatics*, vol. 7(1), pp. 86–112, 2006.

Sanja Brdar received MSc degree in electrical and computer engineering from the University of Novi Sad, Serbia in 2007. She spent two years working in the software industry with major in databases design and development. Currently she is a researcher at the Faculty of Technical Sciences, Novi Sad, where she works toward the PhD degree and participates in projects within BioSense multidisciplinary center. In 2010. she was awarded a ten-month Basileus fellowship and spent it for visiting Bioinformatics Laboratory at University of Ljubljana. Her main research interests encompass machine learning and data mining with applications in biology, agriculture and environmental science.

Vladimir Crnojević received degree in telecommunications and electronics from the Faculty of Technical Sciences, University of Novi Sad, Serbia in 1995, and MSc degree from the Faculty of Electrical Engineering, University of Belgrade, Serbia in 1999. He received PhD degree in communications and signal processing from the University of Novi Sad in 2004, where he is currently an Associate Professor at the Chair for communications and signal processing and the director of BioSense Center - multidisciplinary research center devoted to deployment of state-of-the-art ICT solutions in agriculture, forestry and environment. He is the coordinator and participant of several FP7 projects, EUREKA! research projects for technology transfer, COST and national research projects. His main research interests include image processing, pattern recognition, machine learning and wireless sensor networks.

Blaž Zupan studied computer science at University of Ljubljana, Slovenia, and University of Houston, Texas, USA. He is Professor at University of Ljubljana, and a visiting professor at Baylor College of Medicine in Huston. His research in data mining focuses on methods for data fusion and applications in bioinformatics and systems biology. He is a co-author of Orange, a Python-based and visual programming data mining suite, and several bioinformatics web applications, such as dictyExpress for gene expression analytics and GenePath for epistasis analysis.