A Hierarchical Mixture of Markov Models for Finding Biologically Active Metabolic Paths using Gene Expression and Protein Classes

Hiroshi Mamitsuka
Institute for Chemical Research
Kyoto University
Gokasho, Uji 611-0011, Japan
mami@kuicr.kyoto-u.ac.jp

Yasushi Okuno
Graduate School of Pharmaceutical Sciences
Kyoto University
Sakyo-ku, Kyoto 606-8501, Japan
okuno@pharm.kyoto-u.ac.jp

Abstract

With the recent development of experimental high-throughput techniques, the type and volume of accumulating biological data have extremely increased these few years. Mining from different types of data might lead us to find new biological insights. We present a new methodology for systematically combining three different datasets to find biologically active metabolic paths/patterns. This method consists of two steps: First it synthesizes metabolic paths from a given set of chemical reactions, which are already known and whose enzymes are co-expressed, in an efficient manner. It then represents the obtained metabolic paths in a more comprehensible way through estimating parameters of a probabilistic model by using these synthesized paths. This model is built upon an assumption that an entire set of chemical reactions corresponds to a Markov state transition diagram. Furthermore, this model is a hierarchical latent variable model, containing a set of protein classes as a latent variable, for clustering input paths in terms of existing knowledge of protein classes. We tested the performance of our method using a main pathway of glycolysis, and found that our method achieved higher predictive performance for the issue of classifying gene expressions than those obtained by other unsupervised methods. We further analyzed the estimated parameters of our probabilistic models, and found that biologically active paths were clustered into only two or three patterns for each expression experiment type, and each pattern suggested some new long-range relations in the glycolysis pathway.

1. Introduction

A metabolic pathway represents biochemical activities, and is generally illustrated as a directed graph in current pathway databases [16, 8, 17]. A node in the graph corresponds to a chemical compound, and a directed edge extending a node to another node corresponds to a chemical reaction of generating a compound attached to the latter node from another compound attached to the former. An edge is labeled by a protein, which catalyzes a chemical reaction corresponding to this edge. We here note that a metabolic pathway graph is merely a set of reactions. In fact, in order to draw the graph of a metabolic pathway, a number of chemical reactions (edges) are gathered and connected by putting a same compound in them together as a same node. This property of a metabolic pathway casts serious doubt as to whether a path beginning from a node, which is distant from an ending node in a pathway, is a biologically meaningful (active) path.

The purpose of this paper is to find biologically active paths and/or patterns in a metabolic pathway. For this purpose, we focus on using microarray expression and existing knowledge on proteins, as well as a graph of a metabolic pathway. Microarray expression reveals co-expressed proteins under a certain experimental condition, and the co-expressed proteins will be able to provide a strong clue to find biologically active paths. We then focus on edges, all of which are labeled by co-expressed proteins. Furthermore, co-expressed proteins can be regarded as a set of proteins, which hold a same property. In other words, they fall into a same class. We will then be able to use existing knowledge of protein classes, such as protein structures, families, motifs, functions, etc., for our purpose. These protein classes are in some sense contrast to microarray expression. More concretely, microarray expression is obtained directly from recent high-throughput experiments and very noisy, while protein classes have been accumulated as part of databases for a long time and are very noise-tolerant. Thus it will be effective to use both of microarray expression and known protein classes to find biologically active paths.

In order to develop a method for our purpose, we first assume that all biologically active paths are contained in exist-

Proceedings of the 2004 IEEE Computational Systems Bioinformatics Conference (CSB 2004)
0-7695-2194-0/04 $20.00 © 2004 IEEE

ing graphs of metabolic pathways. This assumption would be valid for a pathway, which has been well investigated. Fortunately we can say that most of major metabolic pathways were already well-studied [18]. A big advantage of this assumption is that we do not need to estimate the graph structure of a metabolic pathway, and we can use all existing metabolic pathways as they are. We further assume that a chemical compound does not appear more than once in a biologically active path we would obtain. Under these assumptions, we present an efficient algorithm to synthesize a path, whose proteins are co-expressed and which is contained in a given metabolic pathway. A path can be regarded as a Markov chain, based on the property that chemical reactions in a metabolic pathway are independent of each other. In other words, we can consider a graph of a metabolic pathway as a Markov state transition diagram, in which a state corresponds to a chemical compound. Furthermore, we consider a probabilistic model for the Markov state transition, and by estimating its probability parameters from the synthesized paths, we can obtain patterns, which frequently appear in the input paths. A key feature of our model is that it is a hierarchical latent variable model, containing a set of protein classes as a latent variable, and clustering input paths is done with this latent variable.

We evaluated our proposed methodology from a variety of viewpoints. First, we examined the predictive performance of our method in a supervised learning manner. We split given binary microarray expression profiles into training and test sets, which are both positives, and randomly generated negatives as a negative test set. Evaluation was then done by checking the accuracy of whether our method, trained by a positive training set only, can discriminate positives from negatives in the test set. We compared this accuracy, which was averaged over ten runs, of our method with those of other methods, including two simple probabilistic models and two types of support vector machines. Obtained results indicate that our method can achieve higher predictive performance than those of the other unsupervised methods in all cases, being statistically significant in roughly 60% cases of all.

Secondly, we assessed the biological significance of the trained models by examining their probability parameter values. We found that input paths were clustered into a relatively small number of patterns, for each experiment type of microarray expression and each protein class. We further found that one or more clear long-range (in the glycolysis pathway) relations/correlations exist for each of these biologically active patterns. We emphasize that these clear findings have never been done, to the best of our knowledge. Over all, our method combined three different types of datasets to effectively find these patterns and long-range correlations, in particular for each given protein class.

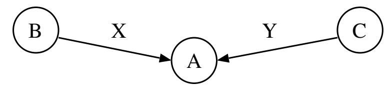

Figure 1. An example of a pathway.

2. Related work

Bayesian and boolean networks [10, 24] have been popularly used to represent biological networks, particularly a gene regulatory network [5]. A Bayesian network is suitable for representing a set of random variables, in which each random variable depends on more than one random variables, and is a graphical representation of these dependencies. We here note that a metabolic pathway should be represented by neither a Bayesian nor a boolean network. Figure 1 shows a simple example of a metabolic pathway containing three chemical compounds, A, B and C. This pathway indicates that chemical compound A can be generated from either B or C, and A does not need both B and C. In other words, the edge extending B to A is independent of the edge extending C to A, and a chemical compound depends on only one chemical compound. However, if we regard this pathway as a graphical representation of a Bayesian network, a variable taking A depends on both a variable taking B and another variable taking C. This is clearly different from what should be represented by this example of a pathway. In addition, a metabolic pathway may have a cycle of reactions, whereas a Bayesian network is an acyclic graph. Thus, we can say that a metabolic pathway should be represented by neither a Bayesian nor a boolean network, when we regard a chemical compound as a node in the network.

A path in a metabolic pathway is a sequence of reactions, namely a time-series sequence, and so our model should take time-series sequences as inputs. A Markov model, which allows to deal with time-series sequences, has been used in a number of applications, such as speech recognition [23], natural language processing [20], biological sequence analysis [7] and web mining [1]. In particular, a mixture of Markov models has been successfully applied to mining web access patterns [4], modeling online customer behavior [2], and biological network modeling [19]. We note that our model hierarchically extends the probabilistic structure of this mixture model, allowing us to combine protein classes with microarray expression and a metabolic pathway.

Identifying and analyzing co-expressed genes are currently the cornerstone of microarray research [11]. A variety of research activities have been already done to analyze metabolic pathways using microarray expression profiles. These include mapping co-expressed proteins on a metabolic pathway [13, 14], co-clustering proteins with microarray expression and metabolic pathways [12], clustering proteins by microarray expression to generate pathways [26] and analyzing microarray data by scoring pathways [29], etc. Our objective is to develop a systematic method for finding biologically active metabolic paths with expression profiles and existing protein classes (Note that the purpose of our previous work [19] is similar, but this method cannot deal with protein classes.). We claim that this objective is different from those of existing methods, to the best of our knowledge. Furthermore, we emphasize that our framework is very systematic by estimating probability parameters of our model with paths efficiently synthesized from expression profiles, and is powerful to find biologically active paths, which are hidden in a given metabolic pathway.

3. Method

3.1. Notations

Let X $(= \{x_1, \cdots, x_L\})$ and Y $(= \{y_1, \cdots, y_M\})$ be a set of chemical compounds and a set of enzymes, respectively. A metabolic pathway $\mathcal{R} = \{R_1, R_2, \cdots, R_J\}$ is a set of reactions, and a reaction R is a triple (x, y, x'), where x and x' are chemical compounds and y is an enzyme (protein) attached to an edge extending compound x to compound x'. Let $\mathcal{F} = \{F_1, \cdots, F_N\}$ be a set of microarray expression profiles, where $F = \{v_1, \cdots, v_M\}$ is a set of real values, and each v_j is a profile of y_j. We call each F_i a *real-value profile record*. Let $\mathcal{G} = \{G_1, \cdots, G_N\}$ be a set of binary microarray expression profiles, where $G = \{b_1, \cdots, b_M\}$ is a set of binary numbers, and b_j is one if v_j is larger than a certain cut-off value, and otherwise zero. We call each G_i a *profile record*. Let $\mathcal{I} = \{I_1, \cdots, I_N\}$ be a set of co-expressed protein sets, and each co-expressed protein set I_i contains all proteins whose binary numbers are one in G_i. We call each I_i a *profile set*. Let $C = \{c_1, \cdots, c_H\}$ be a set of protein classes, and a protein is allowed to belong to more than one classes. Let $\mathbf{x} = x_1 y_2 ... y_T x_T$ be a sequence (metabolic path), where x and y be a chemical compound and a protein, respectively. Let Ξ be a set of sequences.

3.2. Synthesizing metabolic paths

We here describe a method for synthesizing time-series discrete sequences (paths), each of which begins with a beginning compound and ends with an ending compound, from a set of co-expressed protein sets $\mathcal{I}$ and a metabolic pathway $\mathcal{R}$. The output of this method is two: A set of possible sequences, and how many times each of these sequences

In: $\mathcal{I}$, $\mathcal{R}$, x_s, x_f
Out: Ξ $(= \{\mathbf{x}^0, \cdots, \mathbf{x}^T\})$, $E(\mathbf{x}^0), \cdots, E(\mathbf{x}^T)$

Mpath()
Global variable: t
for each $I \in \mathcal{I}$ **do**
    $H = \{x_s\}$; $t = 0$;
    Sub_Mpath(x_s, x_s, H, I)
$T = t$

Sub_Mpath($\mathbf{x}$, x, H, I)
for each $y \in I$ **do**
    if $(x, y, x') \in \mathcal{R}$ and $x' \notin H$
        $\mathbf{x}' = \mathbf{x} y x'$
        if $x' == x_f$
            if $\mathbf{x}' \in \Xi$
                $E(\mathbf{x}') = E(\mathbf{x}') + 1$
            else
                $\mathbf{x}^t = \mathbf{x}'$; $E(\mathbf{x}') = 1$; $t = t + 1$
        else
            $H = \{H, x'\}$
            Sub_Mpath($\mathbf{x}'$, x', H, I)
            $H = H - x'$
return

Figure 2. Pseudocode of algorithm Mpath

appears in $\mathcal{I}$. We note that this method does not depend on particular compounds, and any pair of beginning and ending compounds can be used. We assume that a chemical compound appears no more than once in each sequence.

We first explain a bruteforce approach under this assumption. In this approach, we first synthesize all possible paths from given $\mathcal{R}$ and $\mathcal{I}$, and then scan $\mathcal{I}$ to check whether or not each protein set in $\mathcal{I}$ can synthesize each of all these paths. The time complexity of this straight-forward procedure is $O(f(\mathcal{R}, \mathcal{I}) \cdot N)$, where $f(\mathcal{R}, \mathcal{I})$ is the number of all possible paths synthesized from $\mathcal{R}$ and $\mathcal{I}$, and is an exponential function.

Normally $|I|$, the number of co-expressed proteins, can be considerably smaller than the number of all proteins in $\mathcal{I}$. Considering this situation, we then present a more time- and space-efficient algorithm, which we call Mpath, to count up the number of each of all sequences, which can be synthesized from given $\mathcal{I}$ and $\mathcal{R}$. Let $E(\mathbf{x})$ be the number of times when sequence $\mathbf{x}$ is synthesized from given $\mathcal{I}$ and $\mathcal{R}$. Let x_s and x_f be the beginning and ending compounds, respectively. Figure 2 shows a pseudocode of Mpath, which searches all possible paths in a depth-first manner, taking $\mathcal{I}$, $\mathcal{R}$, x_s and x_t as inputs. The key point of Mpath is that it scans $\mathcal{I}$ to synthesize possible paths, considering only pro-

Proceedings of the 2004 IEEE Computational Systems Bioinformatics Conference (CSB 2004)
0-7695-2194-0/04 $20.00 © 2004 **IEEE**

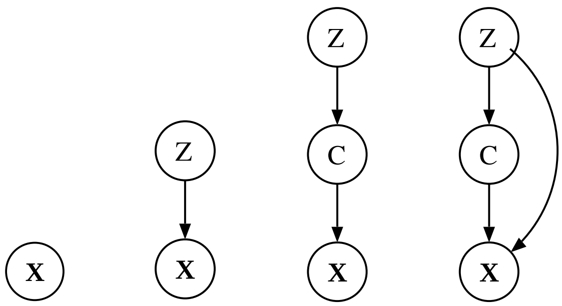

Figure 3. Graphical models of (a) 2M, (b) 3M, (c) H3M and (d) H3M+.

teins in each $I \in \mathcal{I}$. For each I, Mpath repeatedly extends a sequence if possible, checking both I and $\mathcal{R}$ (lines 0 to 2). If it reaches the ending compound, then Mpath increments the number that this path appears in $\mathcal{I}$ (lines 3 to 7). To satisfy the assumption that a compound appears no more than once in a path, Mpath needs to store already appeared compounds (in H in Figure 2) while searching a possible path (line 9). The time complexity of this method is $O(g(\mathcal{R}, \mathcal{I}) \cdot N)$, where $g(\mathcal{R}, \mathcal{I}) = max_{I \in \mathcal{I}} g'(\mathcal{R}, I)$ and $g'(\mathcal{R}, I)$ is the complexity of searching all paths, which can be synthesized from $\mathcal{R}$, considering only proteins in I. We note that $g'(\mathcal{R}, I)$ is extremely smaller than $f(\mathcal{R}, \mathcal{I})$, since as mentioned earlier, $|I|$ is considerably smaller than the number of all proteins contained in $\mathcal{I}$. Thus we emphasize that Mpath significantly reduces the computation time of a bruteforce approach, under the normal situation that $|I|$ is considerably smaller than the number of all proteins in $\mathcal{I}$. We further emphasize that this approach is efficient in terms of space complexity, because Mpath does not count up a sequence which cannot be generated from given datasets.

3.3. Probabilistic models

Clustering synthesized paths is done through estimating the parameters of our probabilistic model using these paths as inputs. Our model is a finite mixture model [21], having two different types of latent variables. One is an usual latent variable, and the other corresponds to a set of protein classes.

3.3.1. Probabilistic Markov model We first describe a probabilistic (first-order) Markov model (2M for short), which does not have a latent variable and is used as a component of our model. This model can be represented as a directed graph, containing nodes and directed edges, and we call a node in this model a *state*. Each state of this model corresponds to a chemical compound in a metabolic pathway, and an edge connecting two states corresponds to a chemical reaction for the two compounds attached to these two states. This model has two types of probability parameters: The initial state probability $p(i)$ is the probability that the first compound in a sequence is compound i. The state transition probability $p(i|m, j)$ is the probability of going from a state j to a state i through an edge labeled by m. In other words, it is the probability that a chemical compound i is synthesized from a compound j, catalyzed by a protein m. More simply, $p(i|m, j)$ is the probability that a reaction R $(= (j, m, i))$ proceeds under a certain condition, and so if this $R \notin \mathcal{R}$, $p(i|m, j)$ is fixed at zero.

We note that a state transition probability of an usual probabilistic first order Markov model is $p(i|j)$, which is specified by only two states, i and j, while that of our model is $p(i|m, j)$, specified by two states and an edge. This is because this probability is the probability of a reaction, which is specified by two states and an edge connecting them. In other words, a reaction of i from j can be synthesized more than one proteins.

This model can be described as follows:

$$p(\mathbf{x}; \theta) \quad = \quad p(x_1; \theta) \prod_{t=2}^{T} p(x_t | y_{t-1}, x_{t-1}; \theta).$$

3.3.2. Mixture of probabilistic Markov models We then describe a mixture of 2M (we call 3M for short), which is a simpler model of our proposed model. This model has a latent variable, and let Z be a discrete-valued latent variable taking on values $z_1, ..., z_K$, each of which corresponds to a latent cluster. This model has a latent cluster probability $p(z)$ of a latent cluster z, adding to an initial state probability and a state transition probability. We note that the latter two probabilities are conditional probabilities, given a latent value z. This model can be described as follows:

$$\begin{aligned} p(\mathbf{x}; \theta) &= \sum_k p(z_k; \theta) p(\mathbf{x} | z_k; \theta) \\ &= \sum_k p(z_k; \theta) p(x_1 | z_k; \theta) \\ &\quad \prod_{t=2}^{T} p(x_t | y_{t-1}, x_{t-1}, z_k; \theta). \end{aligned}$$

Graphical models of 2M and 3M are shown in Figure 3 (a) and (b), respectively. We note that 3M has a representational power, which is basically the same as those in [4, 2, 19], although as mentioned, the component model, 2M, of 3M is different from the component of a general mixture of Markov models.

3.3.3. Hierarchical mixture of probabilistic Markov models In order to consider a set of protein classes, which will be effective for clustering synthesized paths, we extend 3M to a hierarchical latent variable model by adding

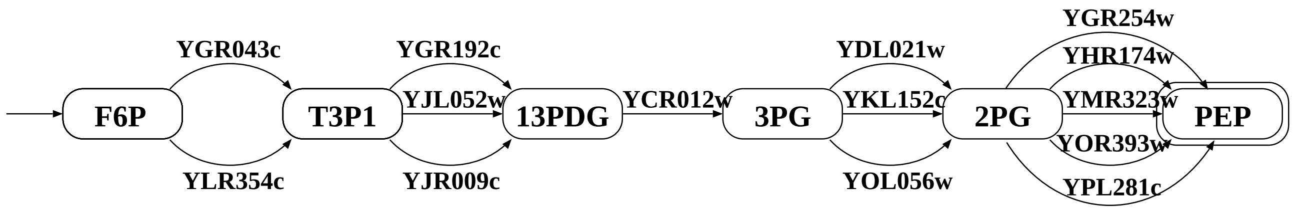

Figure 4. Glycolysis pathway used in our experiments: F6P→PEP.

a latent variable C, taking on a protein class as a discrete value. Thus this model has a latent conditional probability $p(c|z)$, which is a probability of a latent value c, given a latent cluster z. In addition, an initial state probability and a state transition probability depends on a latent value c, instead of z.

This model has the following form:

$$\begin{aligned} p(\mathbf{x};\theta) &= \sum_{k,h} p(z_k;\theta)p(c_h|z_k;\theta)p(\mathbf{x}|c_h;\theta) \\ &= \sum_{k,h} p(z_k;\theta)p(c_h|z_k;\theta)p(x_1|c_h;\theta) \\ &\quad \prod_{t=2}^{T} p(x_t|y_{t-1},x_{t-1},c_h;\theta) \end{aligned}$$

We call this model H3M, standing for *a hierarchical mixture of Markov models*, and a graphical model is shown in Figure 3 (c). In H3M, we just obtain only one component model for each protein class. However, it is more natural that for each protein class, input sequences will be again clustered into some groups. Thus we consider another hierarchical latent variable model, in which probability parameters of a component model depend on both two latent variables, C and Z, at the same time, as follows:

$$\begin{aligned} p(\mathbf{x};\theta) &= \sum_{k,h} p(z_k;\theta)p(c_h|z_k;\theta)p(\mathbf{x}|c_h,z_k;\theta) \\ &= \sum_{k,h} p(z_k;\theta)p(c_h|z_k;\theta)p(x_1|c_h,z_k;\theta) \\ &\quad \prod_{t=2}^{T} p(x_t|y_{t-1},x_{t-1},c_h,z_k;\theta) \end{aligned}$$

We call this model H3M+, and a graphical model of H3M+ is shown in Figure 3 (d). We will examine the performance of H3M+, 3M and 2M in our experiments.

3.3.4. Estimating probability parameters We here show a method for estimating probability parameters of H3M+, and note that the parameters of other models can be also estimated in a similar manner. A possible criterion for estimating probability parameters of H3M+ is the maximum likelihood, in which parameters are obtained to maximize the likelihood of given training data. In order to obtain the maximum likelihood parameters, we apply a general scheme, called EM (Expectation-Maximization) algorithm [6], to H3M+. The algorithm starts with random initial parameter values and iterates both an expectation step (E-step) and a maximization step (M-step) alternately until a certain convergence criterion is satisfied. The EM algorithm for a more general hierarchical model was already developed [3], and the following equations are consistent with it.

In E-step, we estimate the latent classes using the complete data log-likelihood as follows:

$$\begin{aligned} & w(z_k,c_h|\mathbf{x}^d;\theta) \\ &\quad = \frac{p(z_k;\theta)p(c_h|z_k;\theta)p(\mathbf{x}^d|z_k,c_h;\theta)}{\sum_{h',k'} p(z_{k'};\theta)p(c_{h'}|z_{k'};\theta)p(\mathbf{x}^d|z_{k'},c_{h'};\theta)}. \end{aligned}$$

Here let $n_{x,y\to x'}(\mathbf{x})$ be a binary value computed from $\mathbf{x}$ as follows: If a subsequence xyx' is in $\mathbf{x}$ then $n_{x,y\to x',e}(\mathbf{x})$ is 1, otherwise it is 0. In M-step, we take sums over $w(z_k,c_h|\mathbf{x}^d;\theta)$ with $n_{i\to j,e}(\mathbf{x}^d)$, as follows[1]:

$$\begin{aligned} \theta_{i|e,j,c_h,z_k} &\propto \sum_d n_{j\to i,e}(\mathbf{x}^d)\; w(z_k,c_h|\mathbf{x}^d;\theta_{old}). \\ \theta_{c_h|z_k} &\propto \sum_d w(z_k,c_h|\mathbf{x}^d;\theta_{old}). \\ \theta_{z_k} &\propto \sum_{h,d} w(z_k,c_h|\mathbf{x}^d;\theta_{old}). \end{aligned}$$

4. Experimental results

Throughout our experiments, we focus on the proteins of yeast (more concretely, *Saccharomyces cerevisiae*), and all of data used in our experiments were those of yeast. We fixed $K = 10$.

1 We omit to show the updating rules of an initial state probability, but they can be derived similarly.

Experiment types	# profile records	# sequences	
		0.1	0
α-f	18	79	158
Elu	14	66	129
Cdc	25	56	100
Spo	13	60	69
Dia	7	15	51

Table 1. The number of sequences generated when we set the cut-off value at 0 and 0.1.

4.1. Data

4.1.1. Metabolic pathway Figure 4 shows a pathway, which begins with F6P (β-D-Fructose 6-Phosphate) and ends with PEP (Phosphoenolpyruvate). We used this pathway, because it is the most major path of glycolysis and/or gluconeogenesis, and is one of the most fundamental pathways in metabolism. All reactions in this pathway are downloaded from www.cpb.dtu.dk/models/yeastmodel.html [9]. As shown in the figure, this pathway is a simple left-to-right type pathway, i.e. F6P $\rightarrow$ T3P1 $\rightarrow$ 13PDG $\rightarrow$ 3PG $\rightarrow$ 2PG $\rightarrow$ PEP, but each step of generating a compound from another has more than one edges, except for 13PDG $\rightarrow$ 3PG. Thus, this pathway contains totally 90 (=2 $\times$ 3 $\times$ 1 $\times$ 3 $\times$ 5) different paths and fourteen edges, each of which is labeled by a different protein.

4.1.2. Gene expression We used a so-called Eisen dataset, which can be downloaded from rana.lbl.gov/EisenData.htm [27]. This expression dataset contains 80 real-value profile records, all of which differ in experimental conditions and are classified into five different types, 'Cell-cycle α-factor (α-f)', 'Cell-cycle cdc15 (Cdc)', 'Cell-cycle Elutriation (Elu)', 'Sporulation (Spo)' and 'Diauxic shift (Dia),' and more minor types.

We synthesized sequences from both each type of this dataset and a graph given in Figure 4. To do this, we first convert each of the real-value profile records into a profile set, by using a certain cut-off value. A real-value in each of these records indicates how much a corresponding protein is expressed, but it is a relative value to that obtained under a normal condition. Thus, a real-value profile of a protein may be small if this protein is expressed under a normal condition, and most proteins in a main glycolysis pathway can be expressed under a normal condition. We then consider a cut-off value of approximately zero. As with being smaller this cut-off value, the number of sequences to be synthesized increases, because the number of proteins,

Proteins	Functional Codes (Classes)
YCR012w	01.05.01 02.01 40.03
YDL021w	01.05.01 02.01
YGR043c	01.05.01 02.07
YGR192c	01.05.01 02.01 40.03
YGR254w	01.05.01 02.01 40.03
YHR174w	01.05.01 02.01 40.03
YJL052c	01.05.01 02.01 40.03
YJR009c	01.05.01 02.01 40.03
YKL152c	01.05.01 02.01 40.03
YLR354c	01.05.01 02.07 40.03
YMR323w	01.05.01 02.01
YOL056w	01.05.01 02.01
YOR393w	01.05.01 02.01
YPL281c	01.05.01 02.01

Classes	Biological Functions
01.05.01	C-compound carbohydrate utilization
02.01	Glycolysis and gluconeogenesis
02.07	Pentose-phosphate pathway
40.03	Cytoplasm

Table 2. (a) Protein class codes and (b) their functions, for all proteins appeared in F6P$\rightarrow$PEP.

whose profiles are larger than this cut-off, increases. However, if the cut-off value is too small, less than zero in particular, synthesized sequences become extremely noisy [19]. On the other hand, if this value is larger, the number of sequences becomes smaller to use them as inputs to estimate our probabilistic model. We then examined two cut-off values, zero and 0.1. Table 1 shows the number of sequences synthesized when we used these cut-off values. As implied by this table, if we set a cut-off value at 0.2 or larger, we could not obtain sequences whose number is large enough to estimate probability parameters of our model.

4.1.3. Protein classes We used protein functional classes in our model, and the dataset of these classes was downloaded from ftpmips.gsf.de/yeast/catalogues/funcat/ [22]. Table 2 shows the classes taken by each of proteins in Figure 4 and a biological function of each class. Each of all these fourteen proteins belonged to more than one classes, and totally four classes, '01.05.02', '02.01', '02.07' and '40.03', appeared. However, both two proteins of edge from F6P to T3P1 do not belong to '02.01', and this means that it is impossible to synthesize a path whose all proteins belong to '02.01.' This is also true of '02.07'. As a result, we cannot use these two classes to clustering in-

Exp. Types	Cut-off	H3M+	3M	2M	SVC	OC.SVM
α-f	0.1	**72.5**	70.0 (1.58)	70.0 (1.58)	71.7 (0.45)	61.3 (**3.33**)
	0.0	**85.0**	80.0 (**2.35**)	76.3 (**3.07**)	75.0 (**3.26**)	51.7 (**12.6**)
Elu	0.1	90.0	90.0 (0.0)	90.0 (0.0)	**100.0** (**2.86**)	56.7 (**10.0**)
	0.0	90.0	87.5 (1.63)	80.0 (**3.06**)	**97.5** (**2.37**)	71.3 (**2.91**)
Cdc	0.1	**95.0**	**95.0** (0.0)	81.7 (**6.32**)	86.7 (1.96)	66.8 (**5.97**)
	0.0	**92.5**	90.0 (0.95)	68.8 (**3.73**)	72.5 (**3.06**)	53.8 (**6.32**)
Spo	0.1	**85.0**	80.0 (**2.35**)	80.0 (**2.35**)	75.0 (**2.85**)	50.0 (**4.83**)
	0.0	**95.0**	**95.0** (0.0)	**95.0** (0.0)	85.0 (1.87)	65.0 (**3.87**)
Dia	0.1	**95.0**	**95.0** (0.0)	90.0 (1.08)	80.0 (1.85)	50.0 (**9.49**)
	0.0	**92.5**	90.0 (1.05)	90.0 (1.05)	90.0 (1.05)	67.5 (**5.0**)

Table 3. Average prediction accuracies (%) and t-values of mean difference significance test.

put paths. On the other hand, all proteins are in '01.05.01'. There can be a path whose all proteins belong to '40.03'. That is, one example is 'YLR354c - YJL052c - YCR012w - YKL152c - YGR254w.' In this example, YJL052c for 'T3P1 $\rightarrow$ 13PDG' and YGR254w for '2PG $\rightarrow$ PEP' can be replaced with two other proteins, YGR192c or YJR009c, and another protein YHR174c, respectively. Thus, two protein classes, '01.05.01' and '40.03', are used in our experiments, and so $H = 2$. Hereafter we call '01.05.01' and '40.03' as class 01 and class 40, respectively.

4.2. Discriminating positive expressions from negatives

4.2.1. Procedure We first evaluated the performance of H3M+ in a supervised learning manner as follows: We first randomly split a given set of profile records into training and test, and used these two as positive datasets. We then randomly synthesized a negative test dataset, keeping the number of negatives the same as that of positives. We further repeated both these random splitting and random negative example generation ten times, and the performance was averaged over these ten runs. We generated a negative training dataset for a supervised learning approach, which was used to compare with H3M+, in a similar manner as done for a negative test dataset.

Evaluation was done by how exactly positive profile records were discriminated from test negatives. We note that each of H3M+, 3M and 2M predicts (outputs) the likelihood for a sequence, and so the prediction for a test record was done by them as follows: First, a profile record was converted into a profile set, and one or more sequences were synthesized from this profile set. Then, the likelihood of each sequence was computed, and the likelihood (score) for a profile record is given as the average over the likelihoods of all the sequences, synthesized from this profile record[2].

We here note that profile records, i.e. microarray expressions, were used for training all methods tested. The pathway given in Figure 4 was used only for H3M+, 3M and 2M, and the information of protein classes was used only for H3M+.

4.2.2. Competing methods

1) One-class Support Vector Machine (OC.SVM)

The goal of OC.SVM [25] is to find a hyper plane h that separates given binary profile records from the origin in a hyper space at a threshold ρ. For this goal, the following quadratic problem is solved:

$$\min \; \frac{1}{2}||h||^2 + \frac{1}{\mu N}\sum_i \xi_i$$

subject to

$$(h \cdot \Phi(G_i)) \geq \rho - \xi_i \;\; (i = 1, \cdots, N), \;\; \xi_i \geq 0,$$

where Φ be a kernel map.

In our experiments, we used LIBSVM ver.2.36, which has an option to run the OC.SVM and can be downloaded from http://www.csie.ntu.edu.tw/~cjlin/libsvm/. We note that in prediction, for each example, LIBSVM gives only a binary output, indicating whether this example belongs to the class of training data or not.

2) Support Vector Classifier (SVC)

SVC is a well known and the most high-performance supervised learning approach. In

2 We note that when a sequence, whose length is variable, is synthesized, we need to correct the computed likelihood depending on its length. However, a sequence, which was synthesized for the pathway shown in Figure 4, had an equal length, and we did not need to take care of this matter.

our experiments, we used SVM^{light} [15], which can be downloaded from http://svmlight.joachims.org/. For both OC.SVM and SVC, we used a linear kernel in our experiments.

4.2.3. Average prediction accuracy We computed the prediction accuracy for each of the methods, as follows: We first computed a score (or an average likelihood) for each of positives and negatives and sorted all of the scores obtained. We varied a cut-off value for discriminating positives from negatives, and the maximum discrimination accuracy obtained by changing this cut-off value was given as the prediction accuracy. We performed these predictions ten times, and then the obtained prediction accuracies were averaged over the ten runs as the final average prediction accuracy. We evaluated each method by this average prediction accuracy.

We further used 't' values of the (pairwise) mean difference significance test for statistically comparing the prediction accuracy of H3M+ with that of each of the other four methods. The t values are calculated using the following formula:

$$t = \frac{|ave(D)|}{\sqrt{\frac{var(D)}{n}}},$$

where we let D denote the accuracy difference between our method and another method for each of ten runs, $ave(W)$ the average of W, $var(W)$ the variance of W, and n the number of datasets (ten in our case). For $n = 10$, if t is greater than 2.262 then it is more than 95% statistically significant that H3M+ achieves a higher prediction accuracy than another method compared.

Table 3 shows the average prediction accuracies of H3M+ and the other four methods tested and t-values between H3M+ and each of the other four methods. In this table, the highest accuracy for each cut-off value and experimental type is emphasized, and t-values are also emphasized if they are larger than 2.262.

We first note that SVC is a supervised learning approach, and it used negative in training, but these negatives were not used for all of the other methods. Thus, we should compare H3M+ with 3M, 2M and OC.SVM, all of which are unsupervised learning approaches. As shown in the table, H3M+ outperformed (or achieved equal performance of) each of these other unsupervised methods in all 30 cases, being statistically significant in 17 cases, i.e. 56.7% of all cases. In particular, H3M+ outperformed OC.SVM in all of 10 cases, being all statistically significant. OC.SVM used profile records only in training, and so it would be natural that H3M+, which used the metabolic pathway information as well as profile records, outperformed OC.SVM. Furthermore, H3M+ outperformed 3M and 2M in fourteen cases out of all twenty cases, and there were no cases in which 3M and/or 2M outperformed H3M+. 3M and 2M used both microarray expression and the pathway information in training, and in addition to these, H3M+ used the information of protein classes in training. Thus, this performance advantage of H3M+ over 3M and 2M would be also natural. From these results, we can say that H3M+ reasonably combined three different types of datasets to improve the performance of the other methods, which cannot use all of those three datasets.

SVC is a supervised learning approach, which used profile records as negatives, and the other methods did not, since they are unsupervised approaches. Thus, if SVC outperformed another method except for OC.SVM, we can say that the negatives were more effective than the pathway information for discriminating positives from negatives. Interestingly, the performance advantage of SVC over H3M+, 3M and 2M depends on the types of experiments. For example, in Elu, SVC outperformed 3M, 2M and even H3M+, and this implies that microarray expression under Elu is not so strongly correlated with the metabolic pathway of glycolysis, and as a result, the information of pathway is not so effective as negative examples. On the other hand, in Cdc, Spo and Dia, even 3M outperformed SVC, and this result implies that microarray expression under these experimental types are related with glycolysis. In other words, we can say that the effectiveness of combining microarray expression and metabolic pathways strongly depends on experiment types. That is, if an experiment type is correlated with a function of a pathway, it would be useful to combine this pathway with microarray expression under this experiment type.

4.3. Analyzing glycolysis pathway

We further checked the parameter values of H3M+, which was trained by using all synthesized sequences for each experiment type, when we set the cut-off value for expression profiles at zero. From the results obtained, we found that $p(c|h)$ of protein class 40 was almost zero in α-f, Spo and Dia, and this indicates that there were no paths whose proteins are all in class 40. Thus, out of these three types, we first checked probability parameters of the models trained for α-f, since the number of training sequences of α-f was the largest among those of them (as shown in Table 1). We then checked parameters obtained for Elu and Cdc, in which class 40 was used.

4.3.1. Analysis for α-f We obtained ten component Markov models for protein class 01. Figure 5 shows three typical patterns obtained from these ten components. Each of these three patterns had characteristics to be distinguished from the others. More concretely, Figure 5 (a), which was found six cases out of the ten components, was the major pattern, in which only YLR354c

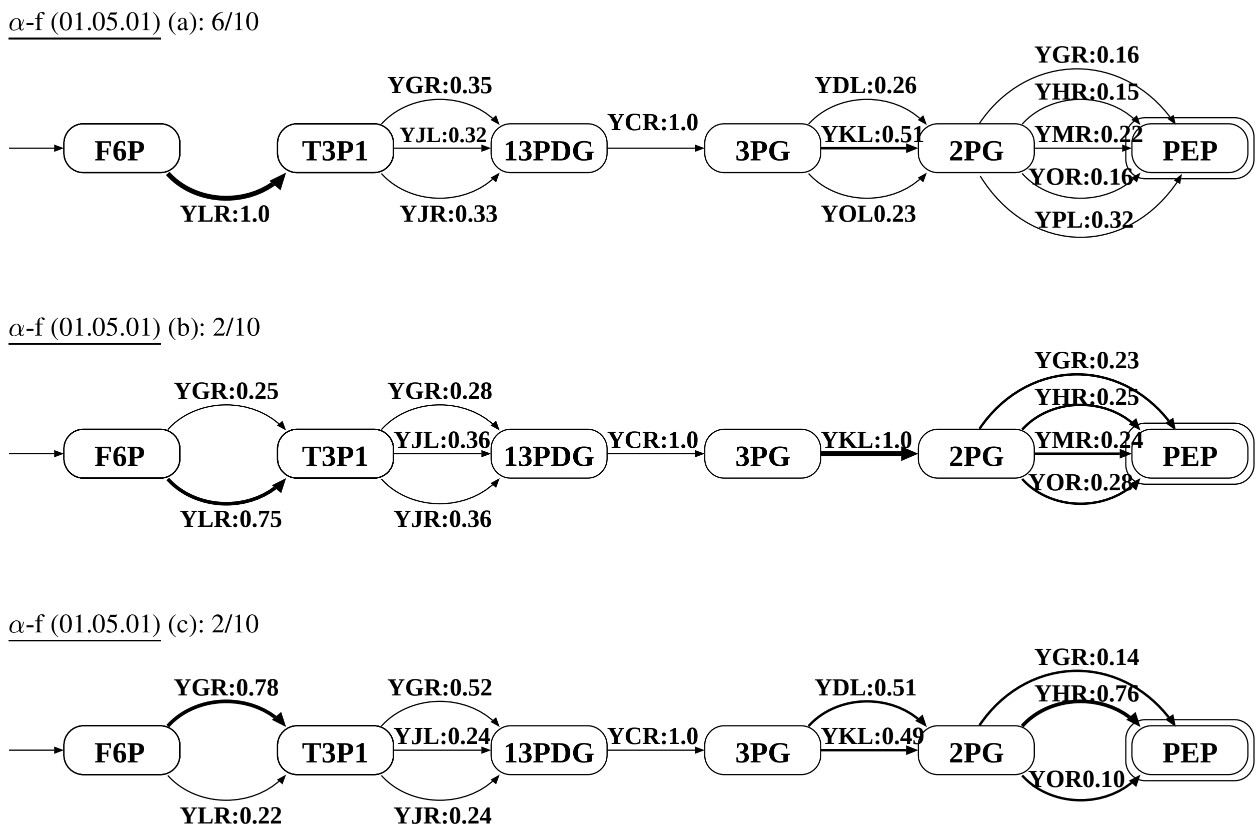

Figure 5. Three patterns obtained for α-f.

was used for 'F6P $\rightarrow$ T3P1,' and all five proteins were used for '2PG $\rightarrow$ PEP.' Figure 5 (b), which was found two cases out of ten, was a minor pattern, in which only YKL152c was used for '3PG $\rightarrow$ 2PG'. We may say that (b) was somewhat similar to (a), because in (a), the probability value of YKL152c for '3PG $\rightarrow$ 2PG' was relatively larger than the others for '3PG $\rightarrow$ 2PG', and the transition probability of YLR354c is larger than that of YGR043c in (b). On the other hand, Figure 5 (c), which was also found two cases out of ten, was a clearly different pattern, in which the probability of YGR043c was larger than that of YLR354c, and YHR174w was mainly used for '2PG $\rightarrow$ PEP.' From these results, we can say that H3M+ captured three patterns, each of which consists of long-range relations between 'F6P $\rightarrow$ T3P1' and '3PG $\rightarrow$ 2PG' (or '2PG $\rightarrow$ PEP') in the glycolysis pathway.

4.3.2. Analysis for Elu We have obtained ten components for each of two protein classes, 01 and 40. Figure 6 shows two typical patterns representing ten components of protein class 01. Figure 6 (a), which was found eight cases out of ten components, was the major pattern, in which only YGR043c was used for 'F6P $\rightarrow$ T3P1.' Interestingly, this pattern is similar to α-f (c), a minor pattern of α-f. Figure 6 (b), which was found two out of ten, was the minor pattern, in which both two proteins are used for 'F6P $\rightarrow$ T3P1,' and only two out of five proteins are equally used for '2PG $\rightarrow$ PEP.' From these (a) and (b), we can see again that H3M+ captured long-range relations between 'F6P $\rightarrow$ T3P1' and '3PG $\rightarrow$ 2PG' (or '2PG $\rightarrow$ PEP'). Another point is that these (a) and (b) are somewhat similar to each other, compared with the difference between Figure 5 (a) and (c). More concretely, two patterns had almost same probability values for 'T3P1 $\rightarrow$ 13PDG' and '3PG $\rightarrow$ 2PG,' and the differences in 'F6P $\rightarrow$ T3P1' and '2PG $\rightarrow$ PEP' were rather small. In other words, biologically active paths in Elu could be represented as only one simple pattern, which would also be very simple at the level of microarray expression. This fact might cause the result that SVC outperformed H3M+ in discriminating positives from random negatives for Elu, as shown in Table 3.In ten components for protein class 40, three probabilities for 'T3P1 $\rightarrow$ 13PDG' and two probabilities for '2PG $\rightarrow$ PEP' were both almost uniform distribu-

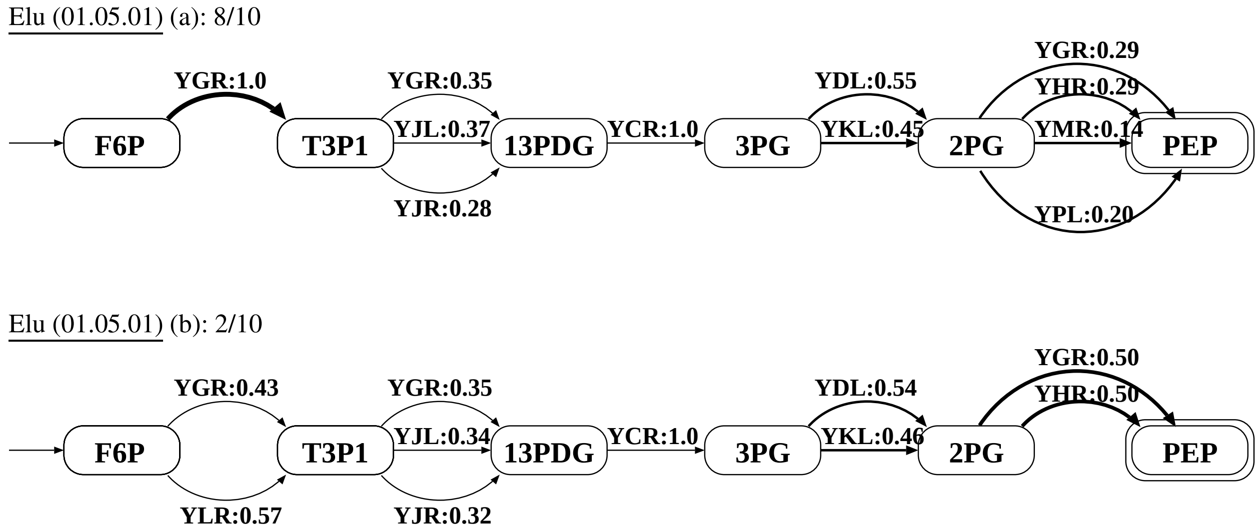

Figure 6. Two patterns obtained for Elu.

tions, and this result might also cause the result that SVC outperformed H3M+.

4.3.3. Analysis for Cdc From ten components for protein class 01, we again obtained a small number of patterns and long-range relations within each of these patterns. We here do not show the details of these patterns, because these patterns have the same nature as those obtained for α-f and Elu. For protein class 40, we found an interesting long-range correlation between 'T3P1 $\rightarrow$ 13PDG' and '2PG $\rightarrow$ PEP.' Figure 7 shows the ten plots (each of which corresponds to one of ten components) on a two-dimensional space of probability values of YJR009c for 'T3P1 $\rightarrow$ 13PDG' and those of YHR174w for '2PG $\rightarrow$ PEP.' As clearly shown in this figure, the probability value of YHR174w linearly increased as with increasing that of YJR009c, indicating a strong positive correlation between these two probability parameters, attached to two edges, which are distant from each other. This result indicates that these two proteins cooperatively work in the metabolic pathway of glycolysis, under the conditions of Cdc.

We note that this finding of a clear long-range correlation as well as finding of biologically active paths for three microarray experiment types have never been reported both experimentally and computationally in existing analysis on glycolysis, e.g. [28], to the best of our knowledge. We emphasize that H3M+ automatically discovered these biologically active patterns containing new knowledge on a long-range relation of proteins in a metabolic pathway. We further emphasize that H3M+ could obtain such patterns and implications separately for each known protein class.

5. Concluding remarks

We have presented a new methodology for combining three different types of data, to find biologically active metabolic patterns. Our method first synthesizes biologically active paths from microarray expression and a metabolic pathway in a time-efficient manner. It then represents these synthesized paths in a compact and comprehensive form by estimating probability parameters of a hierarchical latent variable model. We have empirically showed that our method outperformed the other three unsupervised methods in all cases tested. We further showed that we obtained a relatively small number of biologically active paths for each protein class, and discovered a variety of significant biological implications from them. We can say that our method effectively combined three different datasets to find biologically active metabolic patterns. This combination has never been successfully done, to the best of our knowledge.

Our methodology can be applied to any pathway, unless synthesizing biologically active paths becomes computationally infeasible. It would be important to assess our methodology by finding biologically active paths in another metabolic system, which was already well investigated experimentally. It would be also interesting to characterize more systematically the conditions (e.g. the size of Markov models) under which our proposed method works well. A wider variety of protein classes can be used in our method, and this will reveal the protein classes, which should be combined with microarray expression.

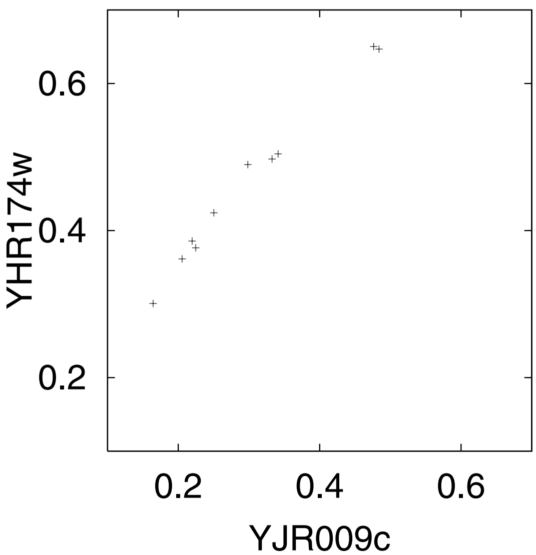

Figure 7. A long-range correlation between 'T3P1 $\rightarrow$ 13PDG' and '2PG $\rightarrow$ PEP' in Cdc (40.03).

6. Acknowledgements

This work was partially supported by Grant-in-Aid for Scientific Research on Priority Areas (C) "Genome Information Science" from the Ministry of Education, Culture, Sports, Science and Technology of Japan.

References

[1] P. Baldi, P. Frasconi, and P. Smyth. *Modeling the Internet and the Web: Probabilistic Methods and Algorithms*. John Wiley & Sons, 2003.

[2] D. Bertsimas, A. Mersereau, and N. Patel. Dynamic classification of online customers. In *Proceedings of the Third SIAM International Conference on Data Mining*, 107-118, 2003.

[3] C. Bishop and M. Tipping. A hierarchical latent variable model for data visualization. *IEEE Pattern Analysis and Machine Intelligence*, 20:281-293, 1998.

[4] I. Cadez, D. Heckerman, C. Meek, P. Smyth and S. White. Model-based clustering and visualization of navigation patterns. *Data Mining and Knowledge Discovery*, 7:399-404, 2003.

[5] H. de Jong. Modeling and Simulation of Genetic Regulatory Systems: A Literature Review. *Journal of Computational Biology*, 9:67-103, 2002.

[6] A. Dempster, N. Laird and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B,* 39:1-38, 1977.

[7] R. Durbin, S. Eddy, A. Krogh and G. Mitchison. *Biological Sequence Analysis*. Cambridge University Press, 1998.

[8] L. Ellis, B. Hou, W. Kang and L. Wackett. The University of Minnesota biocatalysis/biodegradation database. *Nucleic Acids Research*, 31:262-265, 2003.

[9] J. Förster, I. Famili, P. Fu, B. Palsson and J. Nielsen. Genome-scale reconstruction of the *Saccharomyces cerevisiae* metabolic network. *Genome Research*, 13:244-253, 2003.

[10] N. Friedman, M. Linial, I. Hachman and D. Pe'er. Using Bayesian networks to analyze expression data. *Journal of Computational Biology*, 7:601-620, 2000.

[11] D. Gershon. Microarray technology: an array of opportunities. *Nature*, 416:885-891, 2002.

[12] D. Hanisch, A. Zien, R. Zimmer and T. Lengauer. Co-clustering of biological networks and gene expression data. *Bioinformatics*, 18:S145-S154, 2002.

[13] T. Ideker et al. Integrated genomic and proteomic analysis of a systematically perturbed metabolic network. *Science*, 292:929-934, 2001.

[14] J. Ihmels, R. Levy and N. Barkai. Principles of transcriptional control in the metabolic network of *Saccharomyces cerevisiae*. *Nature Biotechnology*, 22:86-92, 2004.

[15] T. Joachims. Making large-scale SVM learning practical. In *Advances in Kernel Methods - Support Vector Learning*. MIT Press, 1999.

[16] M. Kanehisa, S. Goto, S. Kawashima, Y. Okuno and M. Hattori. The KEGG resource for deciphering the genome. *Nucleic Acids Research*, 32:D277-D280, 2004.

[17] P. Karp et al. The EcoCyc database. *Nucleic Acids Research*, 30:56-58, 2002.

[18] K. Koeller and C. Wong. Enzymes for chemical synthesis. *Nature*, 409:232-240, 2002.

[19] H. Mamitsuka, Y. Okuno and A. Yamaguchi. Mining biologically active patterns in metabolic pathways using microarray expression profiles. *ACM SIGKDD Explorations*, 5:113-121, 2003.

[20] C. Manning and H. Schültze. *Foundations of Statistical Natural Language Processing*. MIT Press, 1999.

[21] G. McLachlan and D. Peel. *Finite Mixture Models*. Wiley Inter-Science, 2000.

[22] H. Mewes et al. MIPS: analysis and annotation of proteins from whole genomes. *Nucleic Acids Research*, 32:D41-D44, 2004.

[23] L. Rabiner. A tutorial on hidden Markov models and selected applications. *Proceedings of IEEE*, 75:257-286, 1989.

[24] I. Schmulevich, E. Dougherty and W. Zhang. From boolean to probabilistic boolean networks as models of gene regulatory networks. *Proceedings of IEEE*, 90:1778-1792, 2002.

[25] B. Schölkopf, J. Platt, J. Shawe-Taylor, A. Smola and R. Williamson. Estimating the support of a high-dimensional distribution. *Neural Computation*, 13:1443-1471, 2001.

[26] E. Segal, H. Wang and D. Koller. Discovering molecular pathways from protein interaction and gene expression data. *Bioinformatics*, 19:i264-i272, 2003.

[27] G. Sherlock et al. The Stanford microarray database. *Nucleic Acids Research*, 29:152-155, 2001.

[28] B. ter Kuile and H. Westerhoff. Transcriptome meets metabolome: Hierarchical and metabolic regulation of the glycolytic pathway. *FEBS Letter*, 500:169-171, 2001.

[29] A. Zien, R. Küffner, R. Zimmer and T. Lengauer. Analysis of gene expression data with pathway score. In *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology*, 407-417, 2000.