6

Prediction of Combinatorial Protein-Protein Interaction from Expression Data Based on Conditional Probability

Takatoshi Fujiki, Etsuko Inoue,
Takuya Yoshihiro and Masaru Nakagawa
Wakayama University
Japan

1. Introduction

After the entire human DNA sequence was made public, many post-genome researchers began to investigate the systems of living creatures. Creatures consist of vast collections of proteins and their bodies are maintained by complex interactions among genes, proteins, and organic molecules. One major area of interest is how the characteristics of each creature are manifest and what kind of proteins, genes, and their interactions are related to them.

Much research to detect protein–protein interactions has been conducted. The most direct approach to tackle protein–protein interactions is to identify the evidence of the interactions through *in vitro* or *in vivo* experiments. Since several high-throughput experimental methods to detect physical interactions of proteins, such as yeast two-hybrid [1] and tandem affinity purification [2], have been developed, a significant number of protein interactions have been clarified that accelerated the exploration for protein functionality.

As vast amounts of genome sequences became available, computational approaches to infer protein–protein interactions became more focused. They typically assume some hypotheses of biological activity or property, and search biological databases with their own analytical methods for combinations of proteins to satisfy their hypotheses. Initially, many of these methods simply used gene or protein sequences, e.g., the method based on conservation of gene neighborhoods [3], the Rosetta Stone method [4][5], and the sequence-based co-evolution method [6]. Later, as various public databases became available, such as 3D-structures, domains, motifs, pathways, and phylogenetic profiles, various advanced methods to search for protein–protein interactions were developed. These methods and their results are available on the Web [7].

As one computational approach, gene or protein expression-based analysis is widely used to understand gene or protein interactions, which is the focus of this article. These methods were originally developed for microarray experiments that produced gene expression profiles, but they can apply to protein expression data as well. Because we can now obtain the expression profile of genes using high-throughput experiments such as microarray, protein chip, and 2D-electrophoresis, algorithms to derive interactions from expression data

are increasingly valuable. As a basic analysis, the correlation coefficient of expression levels between two proteins is often used to measure the interaction level of protein pairs. (Note that, in this article, we call this type of interaction the *sole effect*, which refers to the effect on a protein from another single protein.) However, since protein interactions have more complex structures, more sophisticated analyses such as Bayesian networks [8] have been used to understand *combinatorial effects* among proteins. A Bayesian network provides the optimal network computed from a set of expression data, which shows the landscape of interaction effects among proteins. Although this network does not infer direct physical interactions, it helps us gain a better understanding of protein functions. However, since the process of Bayesian network analysis considers the sole effects and the combinatorial effects together, it cannot recognize the combinatorial effects alone.

In this article, we treat interactions among three proteins. We derive the combinatorial effect level, which emerges only when the three proteins are together, besides the sole effects that emerge between two proteins. The combinatorial effect level is estimated in a statistical manner, which will lead to a better understanding of protein interactions and a guide to deeper investigations.

The remainder of this paper is organized as follows. In Section 2, we describe related work to understand the current state of the art in this research area. In Section 3, we describe the model of protein–protein interactions used in our method, and present the method to retrieve the combinatorial effect of three proteins. In Section 4, we evaluate our method by applying it to real protein expression data, and finally in Section 5 present the conclusions.

2. Related work

In this section, we give a short introduction of the major approaches used to predict protein–protein interactions.

Many computational methods to predict protein–protein interactions have been proposed. They utilize various kinds of public data such as genome sequences, amino-acid sequences, pathways, domains, 3D-structures, motifs, and phylogenetic profiles, to identify a property of protein pairs in order to predict protein–protein interactions. One typical genome-sequence-based technique is based on conservation of gene neighbourhood [3]. This technique assumes that genes with similar functions or genes that are in the same pathways are transcribed together as a single unit known as an operon. Thus, finding two proteins that are neighbours in several genomes infers that they interact or have similar functions. Another typical sequence-based technique is called the Rosetta Stone method [4][5]. This method is based on the fact that several pairs of proteins interacting with each other have their homologs in other single proteins, called Rosetta Stone proteins. The phylogenetic profile method [6] uses a series of gene sequences in evolution and detects the set of genes that are simultaneously present or absent in the sequences. Since proteins in interaction tend to disappear simultaneously, finding the set of such genes predicts that the corresponding proteins interact. In addition, the in silico two-hybrid system [9] provides a fully alignment-based protein–protein interaction prediction. This technique tries to detect physical interaction of proteins within their 3D structures by means of correlation of sequences of sites among target proteins. Recently, docking analysis using 3D structures of proteins has progressed rapidly. The main difficulty in docking analysis is that there are many potential

ways in which proteins can interact, and protein surfaces are flexible. Currently, one of the major approaches is a global search based on fast Fourier Transform [10]. Including the methods introduced in this brief discussion, there are a tremendous number of techniques to predict protein–protein interactions, and their algorithms and results are available in public databases. For more details, see [7][11].

Boolean networks [12] and Bayesian networks [8] are well known as computational methods to predict interactions from expression data. It is important to note that they treat gene interactions rather than protein interactions since most of them originally suppose microarray data as their source of analysis. However, they can also treat protein expression data.

A Boolean network [12] is a network that represents causal association and it is typically generated from a pattern of time-series expression data. In Boolean networks, a set of expression levels for a sample at time *t* is regarded as "state" at some time *t*, where each expression level is typically represented by "1 (expressed)" or "0 (not expressed)." To compute the network, the time-series state transition is analyzed to learn the functions to determine the state at time *t+1* from the current state at time *t*. As a result, an expression level of a protein at time *t+1* is determined depending on the expression level of several proteins at time *t*. This dependency indicates the protein–protein interaction, although it does not always indicate a direct interaction. There are several versions and extensions of Boolean networks. Akutsu et al. proposed a model and an algorithm of Boolean networks that is generated from non-time-series expression data [13]. Laubenbacher et al. proposed multistate Boolean networks [14]. However, these models cannot treat noise and, thus, often fail in computing networks. To overcome this problem, Shumulevich et al. proposed a model of probabilistic Boolean networks [15] that enables Boolean networks to apply to practical real expression data that includes noise.

A Bayesian network [8] is also a model of interactions often used in computational approaches that is typically built from expression data with discrete expression levels. Bayesian networks represent a joint distribution of random variables, and its direct edge between nodes represents causal association of those nodes. The learning process of a Bayesian network includes the optimization of network topology, where the evaluation of topologies is based on some information criterion, which is typically based on entropy. Note that it evaluates, for each node, the strength of the relationship between the node and its parents in the network, meaning that the sole effects and the combinatorial effects are evaluated together. Later, as an extension of the model, the Dynamic Bayesian network model was proposed [16], which handles time-series expression data. For details of this kind of network learning, there are several survey articles available, such as [17][18].

3. Method to retrieve combinatorial effects

3.1 Expression data used in our method

In this section, we explain the typical representation of protein expression data. Protein expression data represents the expression level of each protein *i* in sample *j*. Typically, the number of proteins in the data are several hundreds to thousands while the number of samples is usually several tens and at most hundreds.

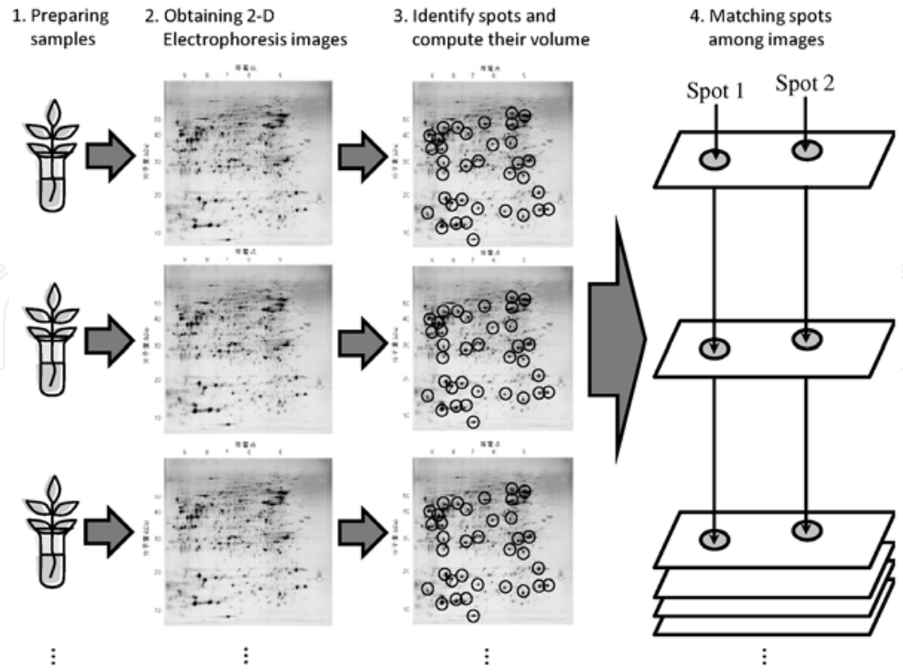


Fig. 1. The process of obtaining Proteome Expression Data.

Sample ID	Protain ID				
	A	B	C	D	…
1	0.000582	0.000107	0.000338	0.000451	…
2	0.000563		0.000475	0.000458	…
3	0.000495	0.000126	0.000433	0.000565	…
4	0.000553	0.000153	0.000382	0.000486	…
5	0.000536	0.000134	0.000536	0.000471	…
6	0.000601	0.000185	0.000457	0.000513	…
:	:	:	:	:	:

Fig. 2. The Data Format for Our Data Mining Process.

Protein expression data is obtained from several methods or devices such as protein arrays, 2D electrophoresis, and mass spectrometry. Among these, we now introduce a 2D electrophoresis-based method [19] as a typical way of generating protein expression data. The process of obtaining protein expression data is somewhat complicated compared to microarray data that measures gene expression levels (see Figure 1). First, we prepare target samples and obtain 2D electrophoresis images from each target sample through an experimental biological process. Second, we identify areas (in the rest of this article we call them *spots*) of separated proteins using image-processing software and measure the expression level of each spot. Third, we match the spots among different images such that the matched spots indicate the same protein. Finally, we normalize the values of expression levels using a normalization method as a preprocess to the data mining processes. As a result, we have a set of protein expression levels as shown in Figure 2, which shows the expression levels of each protein in each sample.

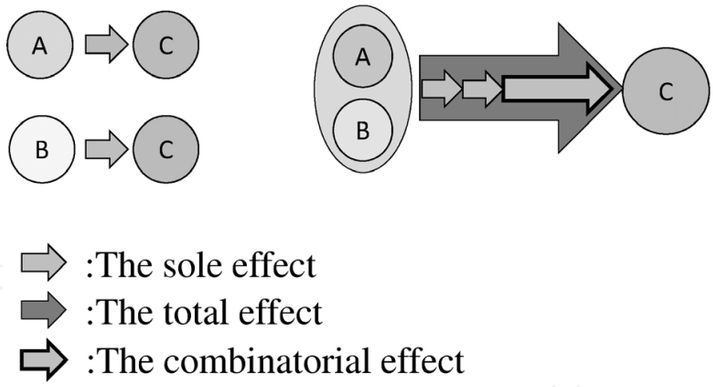


Fig. 3. The Interaction Model to Predict.

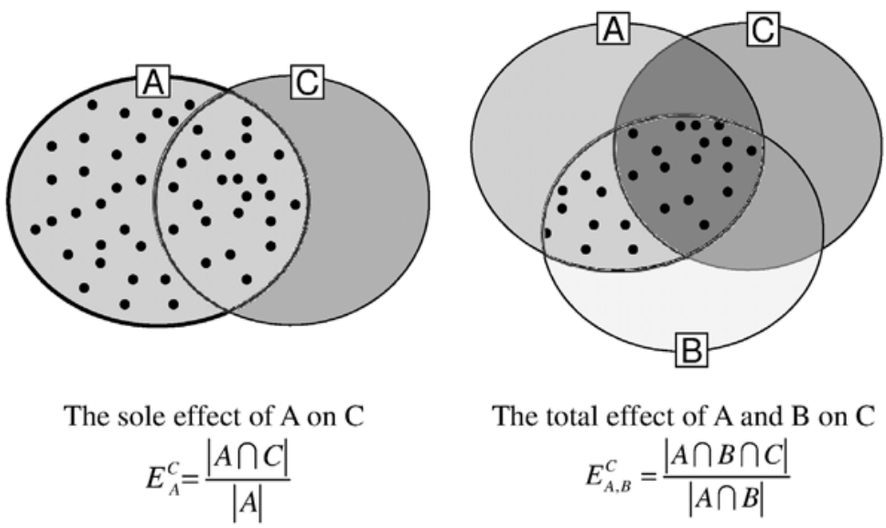


Fig. 4. How to Measure Sole and Total Effect Level of Protein A and B on C.

3.2 Combinatorial protein-protein interaction model

The protein–protein interaction model we try to predict in this paper is shown in Figure 3. Three proteins, A, B, and C, are related to this model, where A and B individually effect the expression level of C, but if both A and B are expressed together, they have a far larger effect on the expression level of C. We call the effect from A to C (resp. B to C) the *sole effect*, and we call the whole effect from A and B on C the *total effect*. Note that the total effect consists of two sole effects and the *combinatorial effect* appears only if both A and B express. What we want to retrieve from expression data is the combinatorial effect of A and B on C.

To measure the combinatorial effect, we first estimate the amount of total effect of A and B on C. Then from the estimated total effect level, we subtract the two sole effects, i.e., the effect of $A - C$ and $B - C$, to obtain the combinatorial effect level.

Note that the three proteins may interact directly or indirectly. We try to extract the three proteins that work in the same functional groups by identifying the behaviour of expression levels following our model of interaction.

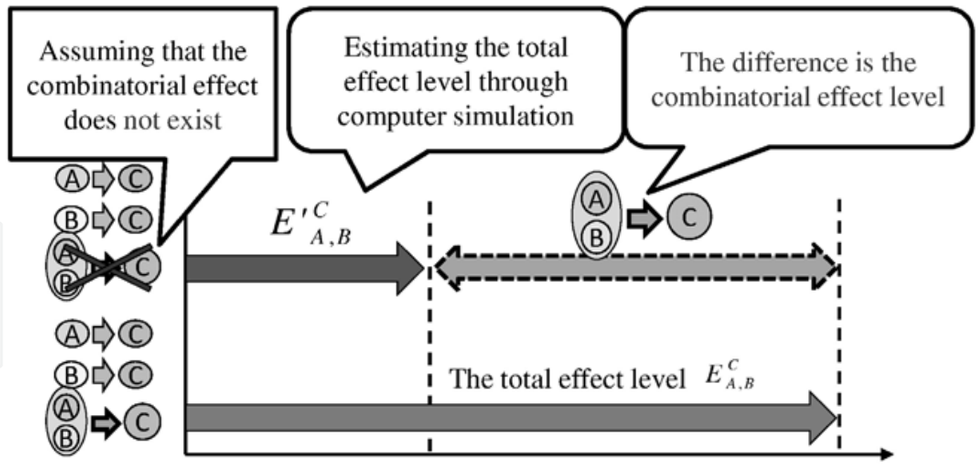


Fig. 5. Dividing Total Effect into Sole and Combinatorial Effect.

3.3 Estimating sole and total interaction levels based on conditional probability

We use conditional probability to retrieve this interaction from expression data. The probability of the sole interactions of $A-C$ and $B-C$ are measured by conditional probability, as shown in Figure 4. Namely, the sole interaction effect level of A on C is measured as the ratio of the number of samples in which the expression levels of both A and C are sufficiently high out of the number of samples in which the expression level of A is sufficiently high. The total interaction effect of A and B on C is also measured in a similar manner, i.e., the ratio of the number of samples in which the expression level of A, B, and C are all sufficiently high out of the number of samples in which the expression levels of both A and B are sufficiently high.

The definitions and formulation of our problems are as follows. We handle proteins i ($1 \le i \le I$) and samples j ($1 \le j \le J$), both of which are included in the input expression data. We also call the proteins A, B, C, ..., and so on. As a parameter, we define r ($0 < r < 1$) as the threshold of the ratio used to judge the expression, i.e., if the expression level of sample j for protein i is within the top r among all the expression levels of protein i, we call the protein i "expressed" in sample j. Let $|A|$ be the number of samples in which protein A is expressed, and similarly, let $|A \cap B|$ be the number of samples in which both protein A and B are expressed. Then, we define $E_A^C = \frac{|A \cap C|}{|A|}$ as the sole effect level of A on C. Similarly, the sole effect level of B on C is defined as $E_B^C = \frac{|B \cap C|}{|B|}$, and the total effect level of A and B on C is defined as $E_{A,B}^C = \frac{|A \cap B \cap C|}{|A \cap B|}$.

3.4 Retrieving combinatorial effect

What we want to estimate is the amount of the combinatorial interaction effect level, which can be estimated from the total interaction level (presented in the previous section) and the sole effect levels of $A-C$ and $B-C$ (see Figure 5). To estimate the combinatorial effect level for the combination of the three proteins A, B, and C, we split the total interaction effect into two parts, i.e., into two sole interaction effects and the combinatorial effect. Then, the

difference between them is regarded as the combinatorial effect level that we wish to compute. To obtain the combinatorial effect level, we compute the statistical distribution of the total effect levels $E'^{C}_{A,B} = \frac{|A \cap B \cap C|}{|A \cap B|}$, which are computed through the simulation executed under the assumption that no combinatorial effect exists over A, B, and C. From the distribution of $E'^{C}_{A,B} = \frac{|A \cap B \cap C|}{|A \cap B|}$ and the total effect score $E^{C}_{A,B} = \frac{|A \cap B \cap C|}{|A \cap B|}$, which is the total effect level presented in the previous subsection, we can estimate the combinatorial effect level.

The computer simulation to compute the distribution of $E'^{C}_{A,B} = \frac{|A \cap B \cap C|}{|A \cap B|}$ is performed as follows. For the corresponding value of α and β, which are the sole effect values for the combination $\mathrm{A} - \mathrm{C}$ and $\mathrm{B} - \mathrm{C}$, we first create distributions of A, B, and C randomly such that the sole effect levels of $\mathrm{A} - \mathrm{C}$ and $\mathrm{B} - \mathrm{C}$ are α and β, respectively. Since those distributions are created randomly, it is possible to assume that they do not include any combinatorial effect. Then we compute the total effect score of the combination A, B, and C. After a sufficient number of repetitions of this process, we obtain the distribution of $E'^{C}_{A,B}$ as the accumulation of the total effect scores. Note that we do not consider what kind of distribution A, B, and C follow in our method since we determine if the protein is expressed using the threshold *r* of the ranking in expression levels.

From this total effect distribution $E'^{C}_{A,B}$, we compute the combinatorial effect as a z-score in the distribution of $E'^{C}_{A,B}$. The z-score $z^{C}_{A,B}$ is defined as $z^{C}_{A,B} = \frac{(E^{C}_{A,B} - \mu)}{\sigma}$, where $E^{C}_{A,B}$ is the total effect level of A, B, and C obtained from the real data, and μ and σ are the average and the standard deviation of the distribution of $E'^{C}_{A,B}$ obtained from the computer simulation, respectively. Namely, the z-score is the difference between the average μ of the distribution of $E'^{C}_{A,B}$ and the real total effect level obtained from the real data, which is measured as the unit value σ. Intuitively, the z-score indicates the probability of the value $E^{C}_{A,B}$ assuming that the combinatorial effect does not exist, which implies the level of the combinatorial effect.

To compute the distribution of the total effect levels through the simulation, however, requires considerable computing time so it is desirable to precompute the distribution. Thus, we prepared a distribution table that shows the average and the standard deviation of the distribution for each value of α and β, as shown in Figure 6. Note that when we compute the distributions in Figure 6, we prepared the data of A, B, and C with 10,000 samples and we perform 5,000,000 trials for each pair of α and β. Because we computed the table for 20 values of α and β between 0 and 1, for obtaining the corresponding values of μ and σ we used the value in the table that is the closest to α and β of A, B, and C.

Now we summarize the proposed method. First, we enumerate every combination of the three proteins A, B, and C from the input data set. For each of the combinations, we compute the total effect level $E^{C}_{A,B}$ of A, B, and C. By referring to the precomputed distribution table, we find the distribution of $E'^{C}_{A,B}$ corresponding to the value α and β of A, B, and C. From the distribution of $E'^{C}_{A,B}$, and the total effect level $E^{C}_{A,B}$, we obtain the combinatorial effect level of A and B on C as the corresponding z-score. Finally, we create a ranking of all the combinations of the three proteins by ordering them by the z-score.

Average table

The sole effect of A on C \ The sole effect of B on C	0.00	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50	0.55	0.60	0.65	0.70	0.75	0.80	0.85	0.90	0.95	1.00
0.00	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.05	0.0000	0.0064	0.0135	0.0212	0.0298	0.0393	0.0500	0.0620	0.0757	0.0913	0.1094	0.1305	0.1556	0.1858	0.2228	0.2694	0.3296	0.4107	0.5255	0.7005	1.0000
0.10	0.0000	0.0135	0.0280	0.0437	0.0609	0.0795	0.1000	0.1225	0.1473	0.1750	0.2059	0.2406	0.2800	0.3251	0.3770	0.4377	0.5093	0.5953	0.7003	0.8315	1.0000
0.15	0.0000	0.0212	0.0437	0.0678	0.0933	0.1207	0.1500	0.1815	0.2154	0.2520	0.2917	0.3348	0.3819	0.4334	0.4901	0.5528	0.6224	0.7002	0.7877	0.8868	1.0000
0.20	0.0000	0.0298	0.0609	0.0933	0.1273	0.1628	0.2000	0.2390	0.2800	0.3231	0.3685	0.4163	0.4668	0.5201	0.5766	0.6365	0.7001	0.7679	0.8401	0.9173	1.0000
0.25	0.0000	0.0393	0.0795	0.1207	0.1628	0.2059	0.2500	0.2952	0.3415	0.3889	0.4376	0.4874	0.5385	0.5910	0.6449	0.7001	0.7569	0.8152	0.8751	0.9367	1.0000
0.30	0.0000	0.0500	0.1000	0.1500	0.2000	0.2500	0.3000	0.3500	0.4000	0.4500	0.5001	0.5501	0.6001	0.6501	0.7001	0.7501	0.8001	0.8501	0.9001	0.9500	1.0000
0.35	0.0000	0.0620	0.1225	0.1815	0.2390	0.2952	0.3500	0.4035	0.4559	0.5069	0.5569	0.6057	0.6534	0.7001	0.7457	0.7904	0.8341	0.8769	0.9188	0.9598	1.0000
0.40	0.0000	0.0757	0.1473	0.2154	0.2800	0.3415	0.4000	0.4559	0.5091	0.5601	0.6088	0.6554	0.7001	0.7429	0.7841	0.8236	0.8616	0.8982	0.9334	0.9673	1.0000
0.45	0.0000	0.0913	0.1750	0.2520	0.3231	0.3889	0.4500	0.5069	0.5601	0.6097	0.6563	0.7001	0.7412	0.7801	0.8167	0.8514	0.8843	0.9154	0.9450	0.9732	1.0000
0.50	0.0000	0.1094	0.2059	0.2917	0.3685	0.4376	0.5001	0.5569	0.6088	0.6563	0.7000	0.7405	0.7779	0.8126	0.8449	0.8750	0.9033	0.9297	0.9546	0.9780	1.0000
0.55	0.0000	0.1305	0.2406	0.3348	0.4163	0.4874	0.5501	0.6057	0.6554	0.7001	0.7405	0.7770	0.8106	0.8412	0.8694	0.8954	0.9194	0.9417	0.9625	0.9819	1.0000
0.60	0.0000	0.1556	0.2800	0.3819	0.4668	0.5385	0.6001	0.6534	0.7001	0.7412	0.7779	0.8106	0.8399	0.8667	0.8909	0.9131	0.9334	0.9520	0.9692	0.9852	1.0000
0.65	0.0000	0.1858	0.3251	0.4334	0.5201	0.5910	0.6501	0.7001	0.7429	0.7801	0.8126	0.8412	0.8667	0.8894	0.9100	0.9286	0.9455	0.9609	0.9750	0.9880	1.0000
0.70	0.0000	0.2228	0.3770	0.4901	0.5766	0.6449	0.7001	0.7457	0.7841	0.8167	0.8449	0.8694	0.8909	0.9100	0.9269	0.9423	0.9561	0.9686	0.9800	0.9904	1.0000
0.75	0.0000	0.2694	0.4377	0.5528	0.6365	0.7001	0.7501	0.7904	0.8236	0.8514	0.8750	0.8954	0.9131	0.9286	0.9423	0.9545	0.9655	0.9754	0.9844	0.9925	1.0000
0.80	0.0000	0.3296	0.5093	0.6224	0.7001	0.7569	0.8001	0.8341	0.8616	0.8843	0.9033	0.9194	0.9334	0.9455	0.9561	0.9655	0.9738	0.9815	0.9882	0.9944	1.0000
0.85	0.0000	0.4107	0.5953	0.7002	0.7679	0.8152	0.8501	0.8769	0.8982	0.9154	0.9297	0.9417	0.9520	0.9609	0.9686	0.9754	0.9815	0.9868	0.9917	0.9960	1.0000
0.90	0.0000	0.5255	0.7003	0.7877	0.8401	0.8751	0.9001	0.9188	0.9334	0.9450	0.9546	0.9625	0.9692	0.9750	0.9800	0.9844	0.9882	0.9917	0.9947	0.9975	1.0000
0.95	0.0000	0.7005	0.8315	0.8868	0.9173	0.9367	0.9500	0.9598	0.9673	0.9732	0.9780	0.9819	0.9852	0.9880	0.9904	0.9925	0.9944	0.9960	0.9975	0.9988	1.0000
1.00	0.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

Standard deviation table

The sole effect of A on C \ The sole effect of B on C	0.00	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50	0.55	0.60	0.65	0.70	0.75	0.80	0.85	0.90	0.95	1.00
0.00	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.05	0.0000	0.0024	0.0032	0.0040	0.0046	0.0053	0.0059	0.0065	0.0070	0.0076	0.0082	0.0088	0.0095	0.0103	0.0111	0.0122	0.0137	0.0156	0.0183	0.0209	0.0000
0.10	0.0000	0.0032	0.0048	0.0055	0.0064	0.0071	0.0078	0.0084	0.0090	0.0096	0.0101	0.0107	0.0112	0.0118	0.0125	0.0133	0.0141	0.0150	0.0155	0.0144	0.0000
0.15	0.0000	0.0040	0.0055	0.0069	0.0076	0.0084	0.0090	0.0096	0.0101	0.0105	0.0109	0.0114	0.0117	0.0121	0.0125	0.0129	0.0132	0.0132	0.0127	0.0108	0.0000
0.20	0.0000	0.0046	0.0064	0.0076	0.0087	0.0092	0.0098	0.0103	0.0107	0.0110	0.0113	0.0115	0.0117	0.0119	0.0120	0.0121	0.0120	0.0116	0.0106	0.0085	0.0000
0.25	0.0000	0.0053	0.0071	0.0084	0.0092	0.0099	0.0103	0.0107	0.0110	0.0112	0.0113	0.0114	0.0115	0.0115	0.0114	0.0112	0.0109	0.0102	0.0091	0.0070	0.0000
0.30	0.0000	0.0059	0.0078	0.0090	0.0098	0.0103	0.0107	0.0109	0.0111	0.0112	0.0112	0.0112	0.0111	0.0109	0.0107	0.0103	0.0098	0.0090	0.0078	0.0059	0.0000
0.35	0.0000	0.0065	0.0084	0.0096	0.0103	0.0107	0.0109	0.0111	0.0111	0.0111	0.0110	0.0108	0.0106	0.0104	0.0100	0.0095	0.0089	0.0080	0.0068	0.0050	0.0000
0.40	0.0000	0.0070	0.0090	0.0101	0.0107	0.0110	0.0111	0.0111	0.0113	0.0109	0.0107	0.0104	0.0101	0.0098	0.0093	0.0087	0.0080	0.0071	0.0060	0.0044	0.0000
0.45	0.0000	0.0076	0.0096	0.0105	0.0110	0.0112	0.0112	0.0111	0.0109	0.0112	0.0103	0.0100	0.0096	0.0092	0.0086	0.0080	0.0073	0.0064	0.0053	0.0038	0.0000
0.50	0.0000	0.0082	0.0101	0.0109	0.0113	0.0113	0.0112	0.0110	0.0107	0.0103	0.0109	0.0095	0.0091	0.0085	0.0080	0.0073	0.0066	0.0057	0.0047	0.0033	0.0000
0.55	0.0000	0.0088	0.0107	0.0114	0.0115	0.0114	0.0112	0.0108	0.0104	0.0100	0.0095	0.0104	0.0085	0.0079	0.0073	0.0067	0.0059	0.0051	0.0042	0.0029	0.0000
0.60	0.0000	0.0095	0.0112	0.0117	0.0117	0.0115	0.0111	0.0106	0.0101	0.0096	0.0091	0.0085	0.0096	0.0073	0.0067	0.0061	0.0054	0.0046	0.0037	0.0026	0.0000
0.65	0.0000	0.0103	0.0118	0.0121	0.0119	0.0115	0.0109	0.0104	0.0098	0.0092	0.0085	0.0079	0.0073	0.0085	0.0061	0.0055	0.0048	0.0041	0.0033	0.0023	0.0000
0.70	0.0000	0.0111	0.0125	0.0125	0.0120	0.0114	0.0107	0.0100	0.0093	0.0086	0.0080	0.0073	0.0067	0.0061	0.0075	0.0049	0.0043	0.0036	0.0029	0.0020	0.0000
0.75	0.0000	0.0122	0.0133	0.0129	0.0121	0.0112	0.0103	0.0095	0.0087	0.0080	0.0073	0.0067	0.0061	0.0055	0.0049	0.0062	0.0038	0.0032	0.0025	0.0017	0.0000
0.80	0.0000	0.0137	0.0141	0.0132	0.0120	0.0109	0.0098	0.0089	0.0080	0.0073	0.0066	0.0059	0.0054	0.0048	0.0043	0.0038	0.0050	0.0027	0.0021	0.0015	0.0000
0.85	0.0000	0.0156	0.0150	0.0132	0.0116	0.0102	0.0090	0.0080	0.0071	0.0064	0.0057	0.0051	0.0046	0.0041	0.0036	0.0032	0.0027	0.0039	0.0018	0.0012	0.0000
0.90	0.0000	0.0183	0.0155	0.0127	0.0106	0.0091	0.0078	0.0068	0.0060	0.0053	0.0047	0.0042	0.0037	0.0033	0.0029	0.0025	0.0021	0.0018	0.0025	0.0010	0.0000
0.95	0.0000	0.0209	0.0144	0.0108	0.0085	0.0070	0.0059	0.0050	0.0044	0.0038	0.0033	0.0029	0.0026	0.0023	0.0020	0.0017	0.0015	0.0012	0.0010	0.0013	0.0000
1.00	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Fig. 6. The Distribution Table of $E'^{C}_{A,B}$ Created through Simulation.

4. Evaluation

4.1 Property of expression data used in our method

In this section, we explain the preprocess applied to the expression data, and also describe the basic property of the data. The expression data used in this experiment originated from the sample of fat near the kidney of black cattle. We performed 2D electrophoresis on each sample and measured the volume of each separated spot that corresponds to each protein. For details of the protocol of the experiment, see [19].

We preprocessed the expression data to improve the reliability of the expression data. Our preprocess consists of the following three steps. First, we removed from the data the

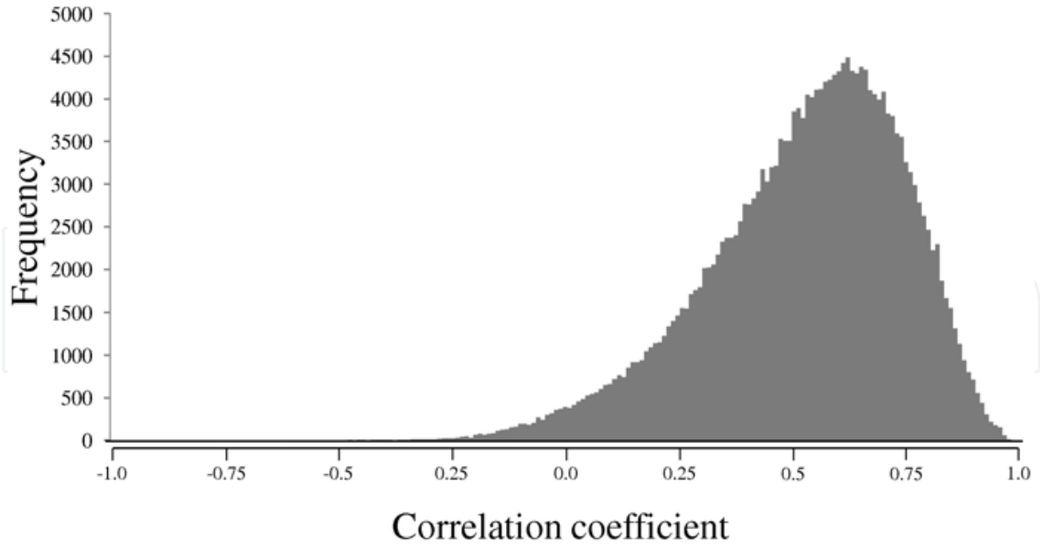


Fig. 7. The histogram of correlation coefficient between proteins.

samples and the proteins that included more than 10% of null expression levels. This was done because samples or proteins with so many null values significantly reduce the reliability of the expression data. Next, we normalized the expression data with the global scaling method [20], where for every sample a scale factor is applied such that the total sum of the protein expression levels in the sample is 1. Finally, we removed the samples with high repetition error. Note that, in fact, in this data set, we performed 2D electrophoresis twice for each sample to confirm the accuracy of each electrophoresis experiment. To maintain the reliability of the data, we removed the sample in which more than 30% of the spots have a high repetation error or null value. Specifically, we consider a spot to have high repetation error if the larger expression level is larger than 1.3 times the value of the smaller expression level. Otherwise, the average of the two expression levels is used for each sample-protein pair. As a result, the expression data used for our evaluation consist of 124 samples and 670 proteins.

In order to indicate a characteristic of this data, we investigated the correlation between proteins. See Figure 7 for the results of calculating correlation coefficients for all pairs of the proteins. Note that the number of pairs is ${}_{670}C_2$ in total. Figure 7 is the histogram where the horizontal axis shows the correlation coefficient separated into classes with 0.05 intervals and the vertical axis shows the frequency of each class. From this result, we can see that most of the correlation coefficients take positive values, and many of them take relatively large values.

4.2 Evaluation experiment of retrieving combinatorial effect

4.2.1 Methods

We performed the experiment to evaluate the performance of the proposed method by applying it to the expression data described in Section 4.1. As a parameter of the experiment, we used the values of 50% and 30% as the threshold r to define the phenomenon that a protein is expressed.

To maintain statistical reliability, we excluded from the analysis the combinations of three proteins where the number of samples was insufficient. Namely, we ignored the

combinations of the three proteins if $|A \cap B|$, which is the denominator in the total effect level $E^C_{A,B}$, was less than 35 in case of r is 50%, and less than 20 in case r is 30%. Similarly, we also removed the combinations if $|A \cap B \cap C|$ was less than 18 in case of r is 50%, and less than 10 in case r is 30%. Furthermore, for the computation, we only used the samples in which all the expression levels of the three proteins are not null.

4.2.2 Results

In this section, we describe the results of the evaluation experiments. Figure 8 shows the histogram of the case of r = 50%, where the horizontal axis indicates the z-scores separated into classes with 0.5 intervals, and the vertical axis indicates the number of combinations in each class. Figure 9 shows the ranking of the top 30 combinations of proteins in terms of z-score. This table includes the columns of the spot numbers of proteins A, B, C, z-score of the combinations, E^C_A and E^C_B (the sole effect levels), $E^C_{A,B}$ (the total effect level), $|A \cap B|$ and $|A \cap B \cap C|$ (the number of samples contained in each phenomenon).

Under the significance level of 1%, we extracted 462,706 combinations in which a strong combinatorial effect is inferred. Here, we caluculate the corresponding p-value to the significance level of 1% using the formula of the Bonferroni correction presented in [21], i.e., p-value $= 1 - e^{\frac{log(1-\gamma)}{n}}$, where n is the number of combinations of three proteins and γ is the significance level. This suggests that if p-value $= 1 - e^{\frac{log(1-0.01)}{149,708,820}} = 6.713 \times 10^{-11}$ or less, the combinatorial effect exists. When the p-value is 6.713×10^{-11}, then the corresponding z-score is 6.423. This is computed as the point in the normal distribution where the probability that the value will become more than the point is p-value = 6.713×10^{-11}. Figure 8 shows only the part where the z-score is larger than 6.423. Note that the probability of a z-score larger than 6.423 is only 6.713×10^{-11} if we assume that there is no combinatorial effect. This and the results of Figure 8 imply that our expression data includes many combinations in which the combinatorial effect exists.

Figure 9 shows that most of the sole effects of the shown combinations occur between 0.4 and 0.45, and the total effects occur between 0.45 and 0.55. Moreover, in most of the combinations, $|A \cap B|$ takes values close to 35, which is the threshold value to judge statistical reliability. This implies that combinations of lower $|A \cap B|$ tend to have larger z-scores. Although it is not shown in Figure 9, the combinations of lower ranks have larger values of $|A \cap B|$.

Figures 10 and 11 show the results with r = 30%. Compared to Figure 8, z-scores tend to have lower values. In addition, the number of combinations with z-scores larger than 6.423 decreases to 167,320. Here, 6.423 is the corresponding p-value with the significance level of 1%. In Figure 11, all of the total effects take a value of 1.0 and all of $|A \cap B|$ take a value of 20, which is the threshold value to judge statistical reliability. Furthermore, about 97.8% of the total effects take 1.0 in the retrieved 167,320 combinations. This means that in most of retrieved combinations, protein C is expressed in all the samples in which both proteins A and B are expressed. This appears to be an unusual tendency. Since in the case of 30% the number of samples in the phenomenon "express" is smaller than in the case of 50%, it is possible that the number of samples is not sufficient to ensure a reliable statistical analysis. One of our future projects will be to clarify why this result appears in the case of r = 30%.

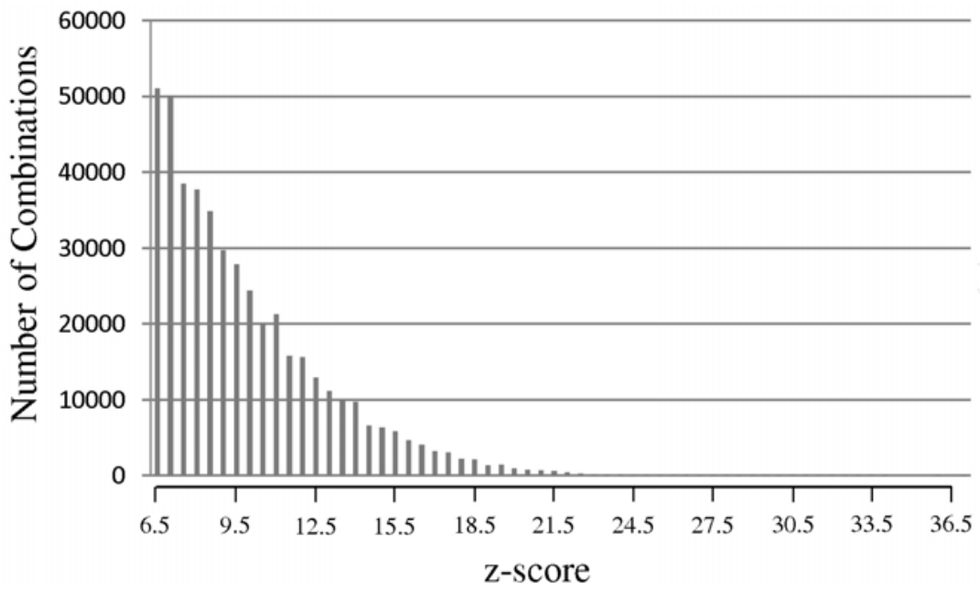


Fig. 8. The histogram of z-score (r=50%).

rank	A (spot No.)	B (spot No.)	C (spot No.)	z-score	E_A^C	E_B^C	$E_{A,B}^C$	|A∩B|	|A∩B∩C|
1	5052	6080	5895	37.3456	0.4583	0.3958	0.5429	35	19
2	3554	5639	5895	36.2082	0.4600	0.4000	0.5429	35	19
3	2742	3554	5895	34.4911	0.4490	0.4490	0.5714	35	20
4	4015	5735	3100	33.7957	0.4348	0.4348	0.5405	37	20
5	5812	5866	1767	33.7458	0.4468	0.4255	0.5429	35	19
6	4798	6080	4849	33.5581	0.4000	0.4000	0.4737	38	18
7	5052	5731	5895	33.4141	0.4468	0.3830	0.5000	36	18
8	5739	6043	4838	33.2666	0.4043	0.4255	0.5000	38	19
9	5812	5866	5895	32.7405	0.4490	0.4286	0.5429	35	19
10	5052	5730	5895	32.6462	0.4375	0.3958	0.5000	36	18
11	3861	6111	5649	32.6423	0.3958	0.3958	0.4615	39	18
12	2318	5940	1765	32.5554	0.4130	0.4348	0.5135	37	19
13	926	5739	5895	32.3921	0.4800	0.4000	0.5429	35	19
14	168	6162	5695	31.9159	0.4667	0.4444	0.5714	35	20
15	5738	6043	3657	31.8151	0.4222	0.4444	0.5278	36	19
16	5639	6242	5895	31.3446	0.3600	0.4400	0.4615	39	18
17	5612	5732	5895	31.3436	0.4375	0.4167	0.5135	37	19
18	6043	6080	4849	31.2987	0.4348	0.3913	0.4865	37	18
19	4015	5735	4838	31.2948	0.4130	0.4565	0.5278	36	19
20	5735	6043	4838	31.2367	0.4348	0.4348	0.5278	36	19
21	4201	5808	3646	31.1739	0.4468	0.4681	0.5714	35	20
22	5726	6242	5895	30.9849	0.4082	0.4490	0.5143	35	18
23	5940	6080	1767	30.7235	0.4255	0.4468	0.5278	36	19
24	2318	4134	5895	30.6533	0.4082	0.5102	0.5714	35	20
25	5734	5866	3467	30.5461	0.4255	0.4043	0.4865	37	18
26	3880	6162	1763	30.5325	0.4444	0.4444	0.5429	35	19
27	5620	5639	5895	30.4974	0.4800	0.3800	0.5135	37	19
28	3554	5621	5895	30.4920	0.4490	0.4694	0.5714	35	20
29	4015	5849	3100	30.4629	0.4222	0.4222	0.5000	36	18
30	5622	5731	5895	30.3865	0.4800	0.4200	0.5526	38	21

Fig. 9. The Top 30 Combinations in z-score (r=50%).

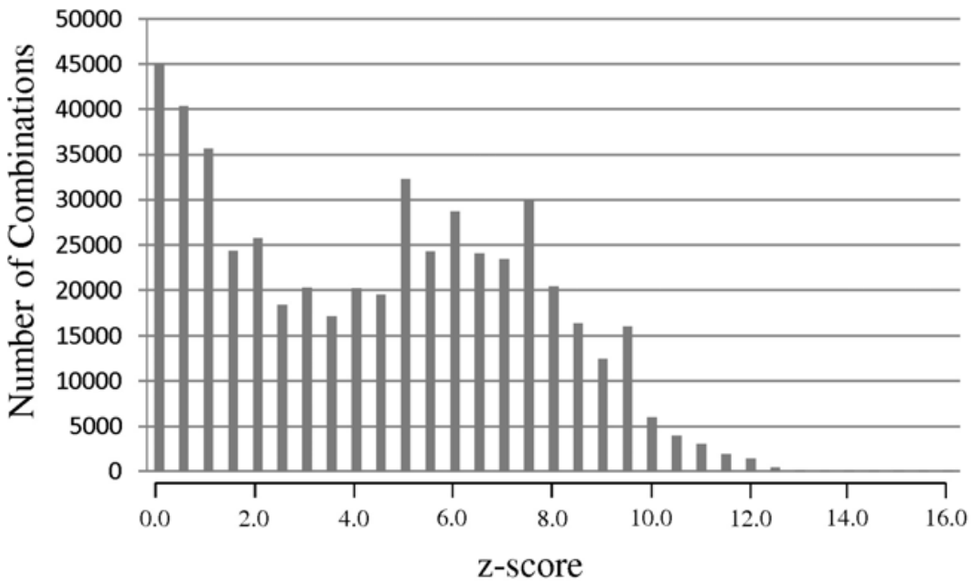


Fig. 10. The histogram of z-score (r=30%).

rank	A (spot No.)	B (spot No.)	C (spot No.)	z-score	E_A^C	E_B^C	$E_{A,B}^C$	|A∩B|	|A∩B∩C|
1	932	4257	4284	16.1614	0.6061	0.6970	1.0000	20	20
2	932	4284	4257	16.1614	0.6061	0.6970	1.0000	20	20
3	934	5140	828	15.9453	0.6176	0.6765	1.0000	20	20
4	932	6240	4134	15.5417	0.7059	0.6176	1.0000	20	20
5	2319	4056	6039	15.5417	0.6176	0.7059	1.0000	20	20
6	934	4284	4257	15.0960	0.6364	0.6970	1.0000	20	20
7	975	4134	4284	15.0960	0.6364	0.6970	1.0000	20	20
8	3998	4795	5045	15.0960	0.6970	0.6364	1.0000	20	20
9	4479	5724	4009	15.0960	0.6970	0.6364	1.0000	20	20
10	5045	5715	5194	15.0960	0.6970	0.6364	1.0000	20	20
11	5573	5954	5218	15.0960	0.6970	0.6364	1.0000	20	20
12	5615	6240	5965	15.0960	0.6970	0.6364	1.0000	20	20
13	2318	4013	5943	15.0958	0.6061	0.7273	1.0000	20	20
14	2318	5943	4013	15.0958	0.6061	0.7273	1.0000	20	20
15	932	4755	5954	14.9425	0.6286	0.7143	1.0000	20	20
16	3972	4755	4476	14.8927	0.7188	0.6250	1.0000	20	20
17	4476	4755	3972	14.8927	0.7188	0.6250	1.0000	20	20
18	4134	6240	5724	14.8927	0.7188	0.6250	1.0000	20	20
19	5724	6240	4134	14.8927	0.7188	0.6250	1.0000	20	20
20	5731	6065	6158	14.8927	0.6250	0.7188	1.0000	20	20
21	5731	6158	6065	14.8927	0.6250	0.7188	1.0000	20	20
22	5733	6065	6158	14.8927	0.6250	0.7188	1.0000	20	20
23	5733	6158	6065	14.8927	0.6250	0.7188	1.0000	20	20
24	934	6240	4134	14.6466	0.7059	0.6471	1.0000	20	20
25	2319	6158	5207	14.6466	0.6471	0.7059	1.0000	20	20
26	5622	5639	5955	14.6466	0.7059	0.6471	1.0000	20	20
27	1762	6034	5965	14.5887	0.7097	0.6452	1.0000	20	20
28	5965	6034	1762	14.5887	0.7097	0.6452	1.0000	20	20
29	1764	3626	5965	14.5887	0.7097	0.6452	1.0000	20	20
30	3626	5965	1764	14.5887	0.6452	0.7097	1.0000	20	20

Fig. 11. The Top 30 Combinations in z-score (r=30%).

4.3 Evaluation experiment of exchangeable proteins

4.3.1 Procedure to exchange proteins

In this section, for the combinations that have high z-scores, we investigate the z-scores when we exchange protein A with protein D in the case where D has a high correlation coefficient with A. Figure 9 shows that many high z-score combinations include C as the common protein, although A and B are also found as common proteins. Since our method defines the samples with the top r expression levels as expressed, having similar z-scores is intuitively inferred if we exchange A with D when D has a high correlation coefficient with A. We believe this is because there are many pairs of proteins in our data set that have a high correlation coefficient allowing us to retrieve so many combinations with a high combinatorial effect. In order to confirm this, we performed an experiment where we exchanged proteins.

The experiment is as follows. First, we create the list of proteins for D that have correlation coefficients against A that are larger than a certain threshold value. Next, we exchange A with D, and calculate the z-score $z^C_{D,B}$ for all combinations of proteins D, B, and C.

4.3.2 Result of exchanging protein

Figure 12 shows the value of the z-scores $z^C_{D,B}$ when A and D are exchanged in the highest z-score combination of A, B, and C in the case r = 50%, where A is exchanged with D if D has the correlation coefficient with A larger than 0.8. This table includes the columns of the spot numbers of proteins A, B, C, protein D exchanged with A, correl(A,D) (the correlation coefficient of A and D), E^C_D (the sole effect level when A and D are exchanged), E^C_B (the sole effect level of before exchanging), $E^C_{D,B}$ (the total effect level), |D ∩ B| and |D ∩ B ∩ C| (the number of samples contained in each phenomenon). In addition, this table is sorted in descending order of z-score.

Figure 12 shows that the lowest z-score as a result of exchanging is 5.503. Note that there are only three combinations that have a z-score less than 6.423, by which the combinatorial effect is inferred under the significance level of 1%. This means that the z-score tends to be high when two proteins with a strong correlation are exchanged. Accordingly, one of the reasons that so many combinations that have a combinatorial effect are retrieved in our data seems to be that our data includes so many pairs of proteins in which the correlation coefficient is high.

5. Conclusion

In this paper, we proposed a method to retrieve the combinatorial protein–protein (or gene-gene) interactions from expression data using statistics of conditional probability. We suppose a model of protein–protein interactions in which the expression level of C takes a large value only if proteins A and B are expressed together. This is the first study to estimate the combinatorial effect level apart from the sole effect. In this study we described our method to treat protein interactions, but note that our method is also applicable to gene expression data generated from microarray experiments.

We evaluated our method using real expression data obtained from a 2D electrophoresis-based experiment. We performed two evaluation experiments with two different parameters, i.e., r = 50% and r = 30%. As a result, the real expression data used in our experiment

A (spot No.)	B (spot No.)	C (spot No.)	D (spot No.)	correl(A,D)	z_score	E_D^C	E_B^C	$E_{D,B}^C$	|D ∩ B|	|D ∩ B ∩ C|
			5142	0.8222	30.7904	0.5306	0.4286	0.6129	31	19
			6019	0.9351	27.2615	0.4898	0.3878	0.5143	35	18
			4275	0.8750	26.9008	0.5600	0.4000	0.5938	32	19
			2312	0.8205	26.3425	0.5000	0.4565	0.5882	34	20
			6043	0.8442	26.2674	0.4600	0.4200	0.5128	39	20
			926	0.8302	26.1577	0.4902	0.4314	0.5526	38	21
			4001	0.8393	25.2836	0.5600	0.4200	0.6061	33	20
			4269	0.8817	25.0023	0.5882	0.4118	0.6250	32	20
			5706	0.9268	24.8728	0.5319	0.3617	0.5161	31	16
			5281	0.8255	24.7993	0.5000	0.3913	0.5152	33	17
			4225	0.8406	24.7963	0.5417	0.3542	0.5172	29	15
			5298	0.8360	24.7883	0.4783	0.4130	0.5161	31	16
			4243	0.8686	24.6493	0.5102	0.3673	0.5000	34	17
			5612	0.8929	24.4295	0.4706	0.4314	0.5250	40	21
			4256	0.8447	24.2406	0.6939	0.4082	0.7308	26	19
			6020	0.9019	24.0511	0.5000	0.3800	0.5000	34	17
			5703	0.9148	24.0087	0.4800	0.4400	0.5405	37	20
			5961	0.8195	23.9841	0.5490	0.4314	0.6000	35	21
			2595	0.8112	23.9176	0.5400	0.4000	0.5588	34	19
			⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
5052	6080	5895	4257	0.8637	14.3240	0.6800	0.4200	0.6774	31	21
			914	0.8112	14.2823	0.5686	0.4314	0.5714	35	20
			4185	0.8303	14.0312	0.6531	0.4286	0.6552	29	19
			4710	0.8103	13.5711	0.5200	0.4400	0.5278	36	19
			5978	0.8501	13.5634	0.7000	0.4400	0.7143	28	20
			5207	0.8104	13.4411	0.5294	0.4314	0.5278	36	19
			2589	0.8287	13.1487	0.5294	0.4314	0.5263	38	20
			921	0.8219	12.5048	0.6000	0.4000	0.5625	32	18
			6057	0.8128	12.4964	0.5652	0.4348	0.5625	32	18
			6012	0.8380	12.4960	0.5625	0.4375	0.5625	32	18
			6181	0.8033	11.6221	0.5714	0.4490	0.5789	38	22
			5060	0.8653	11.1105	0.5800	0.4200	0.5556	36	20
			942	0.8278	10.6944	0.7000	0.4400	0.7000	30	21
			5193	0.8156	8.1077	0.5800	0.4200	0.5405	37	20
			6276	0.8043	8.0482	0.6739	0.4348	0.6538	26	17
			5968	0.8026	7.7789	0.6939	0.4490	0.6875	32	22
			5615	0.8195	6.9890	0.6000	0.4400	0.5758	33	19
			975	0.8314	6.2982	0.7200	0.4400	0.7000	30	21
			4261	0.8536	5.7678	0.6600	0.4200	0.6129	31	19
			978	0.8142	5.5027	0.7000	0.4200	0.6552	29	19

Fig. 12. The ranking of z-score about exchangeable proteins (r=50%).

included a considerable number of combinations in which combinatorial effect is inferred. However, the results are quite different between the two parameters of r that we used in our expeirment. This may be because the number of samples is not sufficient for statistical analysis, and we hope to clarify the validity of our method in detail in our future work. Further, we confirmed that we can exchange protein of A with D when D has strong correlation with A, and we found that the combinatorial effect is still strong even when A is exchanged with D.

In the future, we would like to perform more experiments to further validate our proposed method. In addition, we would like to develop an algorithm for the analytical computation

of the statistical distribution under the assumption of no combinatorial effect, i.e., we would like to compute the distribution shown in Figure 6 without simulation. If such fast computation is possible, it enables us to easily vary the threshold *r*, and it also enables us to compute a more accurate analysis. Finally, we also would like to find the known interactions in our results verify the value of this data-mining method.

6. Acknowledgment

This work was partly supported by the Program for Promotion of Basic and Applied Researches for Innovations in Bio-oriented Industry.

7. References

[1] Fields, S., & Song, O. (1989). A novel genetic system to detect protein-protein interactions, *Nature*, Vol. 340, pp. 245-246.

[2] Rigaut, G., Shevchenko, A., Rutz, B., Wilm, M., Mann, M. & Séraphin, B. (1999). A generic protein purification method for protein complex characterization and proteome exploration, *Nature Biotechnology*, Vol. 17, pp. 1030 - 1032.

[3] Overbeek, R., Fonstein, M., D'Souza, M., Pusch, G.D. & Maltsev, N. (1999). The use of gene clusters to infer functional coupling, *Proc. Natl Acad Sci U S A*, Vol. 96, No. 6, pp. 2896-2901.

[4] Enright, A.J., Iliopoulos, I., Kyrpides, N.C. & Ouzounis, CA. (1999). Protein interaction maps for complete genomes based on gene fusion events, *Nature*, Vol. 402, No. 6757, pp. 86-90.

[5] Marcotte, E.M., Pellegrini, M., Ng, H.L., Rice, D.W., Yeates, T.O.& Eisenberg, D. (1999). Detecting protein function and protein-protein interactions from genome sequences, *Science*, Vol. 285, No. 5428, pp. 751-753.

[6] Pellegrini, M., Marcotte, E.M., Thompson, M.J., Eisenberg, D. & Yeates, T.O. (1999). Assigning protein functions by comparative genome analysis: protein phylogenetic profiles, *Proc. Natl Acad Sci U S A*, Vol. 96, No. 8, pp. 4285-4288.

[7] Tuncbag, N., Kar, G., Keskin, O., Gursoy, A. & Nussinov, R. (2009). A Survey of Available Tools and Web Servers for Analysis of Protein-Protein Interactions and Interfaces, *Briefings in Bioinformatics*, Vol. 10, No.3, pp.217-232.

[8] Friedman, N., Linial, M., Nachman, I. & Pe'er, D. (2000). Using Bayesian Networks to Analyze Expression Data, *Journal of Computational Biology*, Vol. 7, No. 3/4, pp. 601-620.

[9] Pazos, F. & Valencia, A. (2002). In silico two-hybrid system for the selection of physically interacting protein pairs. *Proteins: Structure, Function and Genetics*, Vol. 47, No. 2, pp. 219-227.

[10] Comeau, S.R., Gatchell, D.W., Vajda, S. & Camacho, C.J. (2004). ClusPro: A Fully Automated Algorithm for Protein-protein Docking, *Nucleic Acids Research*, Vol. 32(Web server issue), pp. W96-99.

[11] Jothi, R. & Przytycka, T.M. (2008). Computational approaches to predict protein-protein and domain-domain interactions, In: *Bioinformatics Algorithms: Techniques and Applications*, Mondoiu, I.I. and Zelikovsky, A. of Editors, pp. 465-492, Wiley Press, ISBN 978-047-0097-73-1.

[12] Liang, S., Fuhrman, S. & Somogyi, R. (1998). REVEAL, a General Reverse Engineering Algorithm for Inference of Genetic Network Architectures, *Proc. Pacific Symposium on Biocomputing '98*, pp. 18-29.

[13] Akutsu, T., Kuhara, S., Maruyama, O. & Miyano, S. (1998). A system for identifying genetic networks from gene expression patterns produced by gene disruptions and overexpressions, *Genome Informatics*, Vol. 9, pp. 151-160.

[14] Laubenbacher, R. & Stigler, B. (2004). A Computational Algebra Approach to the Reverse Engineering of Gene Regulatory Network, *Journal of Theoretical Biology*, Vol. 229, No. 4, pp. 523-537.

[15] Shmulevich, I., Dougherty, E.R., Kim, S. & Zhang, W. (2002). Probabilistic Boolean Networks: A Rule-based Uncertainty Model for Gene Regulatory Networks, *Bioinformatics*, Vol. 18, No. 2, pp. 261-274

[16] Husmeier, D. (2003). Sensitivity & Specificity of Inferring Genetic Regulatory Interactions From Microarray Experiments with Dynamic Bayesian Networks, *Bioinformatics*, Vol. 19, No. 17, pp. 2271-2282.

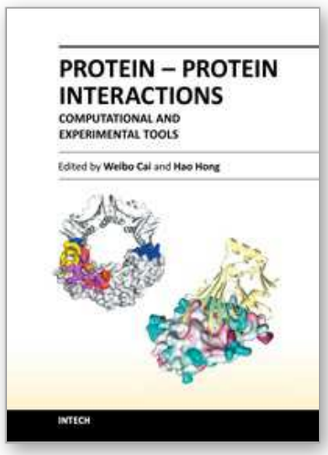
[17] Huang, Y., Tienda-Luna, I.M. & Wang, Y. (2009). A Survey of Statistical Models for Reverse Engineering Gene Regulatory Networks, *IEEE Signal Process Mag*, Vol. 26, No. 1, pp. 76-97.

[18] Sima, C., Hua, J. & Jung, S. (2009). Inference of Gene Regulatory Networks Using Time-Series Data: A Survey, *Current Genomics*, Vol. 10, No. 6, pp. 416-429.

[19] Nagai, K., Yoshihiro, T., Inoue, E., Ikegami, H., Sono, Y., Kawaji, H., Kobayashi, N., Matsuhashi, T., Ohtani, T., Morimoto, K., Nakagawa, M., Iritani, A. & Matsumoto, K. (2008). Developing an Integrated Database System for the Large-scale Proteomic Analysis of Japanese Black Cattle, *Animal Science Journal*, Vol. 79, No. 4. (in Japanese)

[20] Lu, C. (2004). Improving the Scaling Normalization for High-density Oligonucleotide GeneChip Expression microarrays, *BMC Bioinformatics*, Vol. 5, pp. 103.

[21] Spelman, R.J., Coppieters, W., Karim, L., van-Arendonk, J.A.M., & Bovenhuis, H. (1996). Quantitative Trait Loci Analysis for Five Milk Production Traits on Chromosome Six in the Dutch Holstein-Friesian Population, *Genetics,* Vol. 144, No. 4, pp. 1799-1808.



Protein-Protein Interactions - Computational and Experimental Tools

Edited by Dr. Weibo Cai

ISBN 978-953-51-0397-4
Hard cover, 472 pages
Publisher InTech
Published online 30, March, 2012
Published in print edition March, 2012

Proteins are indispensable players in virtually all biological events. The functions of proteins are coordinated through intricate regulatory networks of transient protein-protein interactions (PPIs). To predict and/or study PPIs, a wide variety of techniques have been developed over the last several decades. Many in vitro and in vivo assays have been implemented to explore the mechanism of these ubiquitous interactions. However, despite significant advances in these experimental approaches, many limitations exist such as false-positives/false-negatives, difficulty in obtaining crystal structures of proteins, challenges in the detection of transient PPI, among others. To overcome these limitations, many computational approaches have been developed which are becoming increasingly widely used to facilitate the investigation of PPIs. This book has gathered an ensemble of experts in the field, in 22 chapters, which have been broadly categorized into Computational Approaches, Experimental Approaches, and Others.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Takatoshi Fujiki (2012). Prediction of Combinatorial Protein-Protein Interaction from Expression Data Based on Conditional Probability, Protein-Protein Interactions - Computational and Experimental Tools, Dr. Weibo Cai (Ed.), ISBN: 978-953-51-0397-4, InTech, Available from: http://www.intechopen.com/books/protein-protein-interactions-computational-and-experimental-tools/prediction-of-combinatorial-protein-protein-interaction-from-expression-data-based-on-conditional-pr

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen