

# Network-based auto-probit modeling for protein function prediction

Xiaoyu Jiang<sup>1</sup>, David Gold<sup>2</sup>, Eric D. Kolaczyk<sup>1\*</sup>

<sup>1</sup>Department of Mathematics and Statistics  
Boston University  
Boston MA 02215, USA

<sup>2</sup>Department of Biostatistics  
School of Public Health and Health Professions  
University at Buffalo, the State University of New York  
Buffalo, NY 14214, USA

\**email*: kolaczyk@math.bu.edu

## Abstract

Predicting the functional roles of proteins based on various genome-wide data, such as protein-protein association networks, has become a canonical problem in computational biology. Approaching this task as a binary classification problem, we develop a network-based extension of the spatial auto-probit model. In particular, we develop a hierarchical Bayesian probit-based framework for modeling binary network-indexed processes, with a latent multivariate conditional autoregressive (CAR) Gaussian process. The latter allows for the easy incorporation of protein-protein association network topologies – either binary or weighted – in modeling protein functional similarity. We use this framework to predict protein functions, for functions defined as terms in the Gene Ontology (GO) database, a popular rigorous vocabulary for biological functionality. Furthermore, we show how a natural extension of this framework can be used to model and correct for the high percentage of false negative labels in training data derived from GO, a serious short-coming endemic to biological databases of this type. Our method performance is evaluated and compared with standard algorithms on weighted yeast protein-protein association networks, extracted from a recently developed integrative database called *STRING*. Results show that our basic method is competitive with or better than these other methods, and that the extended method – incorporating the uncertainty in negative labels among the training data – can yield nontrivial improvements in predictive accuracy.

**Keywords:** *Auto-probit; Bayesian hierarchical model; Gene Ontology annotation uncertainty; MCMC algorithm; Protein function prediction; STRING*

# 1 Introduction

Inferring the functional role of proteins is a primary task in biology, for purposes ranging from general knowledge to drug discovery and diagnostic development. Protein functions are commonly taken as terms from the Gene Ontology (GO) database, a controlled vocabulary which describes gene and gene product attributes in organisms (*www.geneontology.org*). Whether or not a given protein has a certain GO function can be encoded using a binary variable, with 1 indicating that the protein has the function and 0 indicating that it does not. Proteins can have multiple functional annotations in GO, in part because of their ability to perform multiple biological roles and in part because GO terms are structured as hierarchies, ranging from general to more specific. However, protein-term annotations in GO must follow a *true-path rule*, which states that if a protein is categorized into a more specific functional class, it must also be categorized into all the less specific ancestor functional classes.

Protein-protein interaction (PPI) networks are one of the most commonly used sources of information for predicting protein functions. PPI networks are routinely described by graphs, with vertices corresponding to proteins and edges indicating interactions between a pair of proteins. For a given protein function, the corresponding binary variables describing protein annotation status can be thought of as constituting a binary stochastic process indexed on the PPI network. Protein function prediction is then viewed as a task of predicting binary labels at locations in the network where they are unknown, given the labels observed at other nearby locations. Classifiers are usually built for this purpose and various methodologies of this sort have been proposed, frequently based on the principle of “guilt-by-association”. These include nearest-neighbor algorithms, the methods introduced in [2, 8, 20], and probabilistic approaches such as [4, 11, 14, 15]. For a summary of various methods for protein function prediction, please refer to [21].

Broadly speaking, results in the literature seem to suggest that – using PPI interactions alone – most methods perform similar to nearest-neighbor methods. This phenomenon is presumably a manifestation of the fact that even relatively simple classifiers often can perform reasonably well on hard problems (see [5]). To improve performance in this area, most recent work has therefore concentrated on the problem of “data integration” i.e., combining PPI data with other data sources, such as DNA binding motifs, protein localization, and gene expression data. See, for example, [4, 10, 13], and other references cited therein.

In this paper, the framework we propose is motivated by the desire to pursue two rather different avenues towards improvement in protein function prediction. First, we wish to incorporate weighted –rather than binary – protein-protein association networks in a seamless fashion into a probabilistic framework. An example of a weighted PPI network, and one which we shall use later in the data analyses described in this paper, is that derived from the STRING database in [16], which contains a combination of known and predicted protein-protein associations and corresponding scores. The scores express increased confidence when an association is supported by several types of evidence, which can be highly informative in inferring proteins’ functional characteristics. Taking advantage of the scores in databases such as STRING has become a new challenge as well as an opportunity to improve function prediction accuracy. However, it is not obvious how this challenge can be met by simple adaptations of the “guilt-by-association” principle governing methods like those mentioned earlier.

Second, we wish to model and account for uncertainty in annotations in the Gene Ontology database. Most methods using annotations in the GO database for training classifiers assume that a protein being annotated or not annotated with a function accurately reflects whether or not that protein truly has that function or not. However, while positive annotations – which traditionally have reflected experimentally confirmed protein functions – are generally reliable, negative annotations can be much less so. The reason for this disparity in reliability comes from the fact that negative annotation for a given GO term can reflect either a known lack of positive annotation (perhaps logically implied by certain positive annotations on other GO terms) or simply an absence of knowledge as to the protein status with respect to this term. This observation suggests that, instead of treating the task of protein function prediction as a binary classification problem, we actually have three classes - “having the function”, “not having the function” and “status unknown”. This observation has been made by [15], but otherwise does not appear to have been widely acknowledged in the literature. However, results in this paper show that acknowledging and, moreover, accounting for it appropriately can yield nontrivial improvements in predictive accuracy.

In this paper we develop a network-based hierarchical Bayesian auto-probit model that allows for us to address both of the issues described above i.e., seamless incorporation of weighted networks and accounting for uncertainty in negative associations. The incorporation of weighted networks is facilitated by our use of a latent Gaussian process to encode functional similarity, and the accounting for annotation uncertainty is accomplished by including an additional layer of probabilistic error. The paper is organized

as follows. In Section 2 we develop our model under the assumption that there is no annotation uncertainty in GO. We then generalize this model to tackle the annotation uncertainty problem in Section 3. Results for model fitting, prediction and model comparison on two yeast networks are presented in Section 4. Some discussion follows in Section 5.

## 2 Bayesian hierarchical model for protein function prediction

### 2.1 Network-based auto-probit model

Suppose we have a collection of  $N$  proteins,  $n \leq N$  of which have functional annotations in the Gene Ontology database, and  $N - n \geq 0$  for which we wish to predict functionality. For a given GO term of interest, let the binary variable  $y_i$  denote the functional annotation of protein  $i$ . That is,  $y_i = 1$  if protein  $i$  is annotated with the term in the database, and  $y_i = 0$  otherwise, for  $i = 1, \dots, n$ . In this section, we will assume the annotations in the database to be without error.

Motivated by [22], who develop a spatial auto-probit model for lattice data, we employ a latent Gaussian process  $\mathbf{z}$  to represent the **true** functional status of proteins for the term of interest. That is,  $z_i \geq 0$  if protein  $i$  actually has the function, and  $z_i < 0$ , otherwise, for  $i = 1, \dots, N$ . Since we assume here that there is no annotation error in the GO database, the sign of  $z_i$  fully determines  $y_i$ , i.e.,

$$y_i = \begin{cases} +1, & \text{if } z_i \geq 0; \\ 0, & \text{if } z_i < 0. \end{cases}$$

It is commonly assumed that proteins with the same or similar functions tend to interact more frequently than others. This assumption underlies, for example, the *local density enrichment assumption* in [15]. Therefore, we want to encode protein-protein association network structure into the model to aid in inferring the functional labels. Let  $A$  be the  $N \times N$  adjacency matrix for a protein network. For a binary (i.e., unweighted) network,  $a_{ij} = 1$  for interacting neighbors  $i$  and  $j$ , and 0 otherwise. For a weighted network,  $a_{ij}$  takes on the value of the weight for the edge  $\{i, j\}$ , with a weight of  $a_{ij} = 0$  indicating no edge. For each  $i$ , let  $d_i = \sum_{j \neq i} a_{ij}$  be the degree of protein  $i$ . If the network is binary,  $d_i$  simply counts the number of neighbors of protein  $i$ . Denote by  $D = \text{diag}\{d_i\}$  the diagonal degree matrix.

The  $N \times 1$  latent Gaussian process vector  $\mathbf{z}$  is assumed to follow a multivariate normal distribution

$$\mathbf{z}|\boldsymbol{\mu}, \beta \sim MVN(\boldsymbol{\mu}, (I - \beta D^{\frac{1}{2}} A D^{\frac{1}{2}})^{-1}),$$

where  $\mu_i$  is the location parameter for protein  $i$  and  $\beta$  measures the spatial dependence from the network. We constrain the value of  $\beta$  to ensure that the precision matrix  $I - \beta D^{\frac{1}{2}} A D^{\frac{1}{2}}$  is positive definite. First, we assume that neighbor proteins tend more often than not to agree with each other, according to the *local density enrichment assumption*, and thus restrict  $\beta$  to be nonnegative. Second, writing the determinant of the precision matrix as

$$|I - \beta D^{\frac{1}{2}} A D^{\frac{1}{2}}| = \prod_{i=1}^N (1 - \beta \lambda_i),$$

where  $\lambda_i$  is the  $i$ -th eigenvalue of matrix  $D^{\frac{1}{2}} A D^{\frac{1}{2}}$ , we note that a sufficient condition for this determinant to be positive is that  $\beta < \lambda_{\max}^{-1}$ , where  $\lambda_{\max} = \max_i \lambda_i$ . Hence, we constrain  $\beta$  such that  $0 < \beta < \lambda_{\max}^{-1}$ , which is similar to [22] in the context of spatial auto-probit modeling.

To understand the manner in which this model incorporates the network topology, we note that the partial correlation coefficient  $\rho_{ij}$  for  $z_i$  and  $z_j$  on two neighbor proteins takes the form

$$\rho_{ij} = \begin{cases} \beta \sqrt{d_i d_j} a_{ij}, & \text{if } i \sim j, \\ 1, & \text{if } i = j, \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

This expression indicates that proteins with more neighbors should be more likely to be consistent with the majority of the neighbors; in other words, a protein with a larger neighborhood and hence a bigger  $d_i$  would be more closely correlated with its neighbors controlling other proteins outside the neighborhood. We call this the “one-hop neighborhood effect.” On the other hand, a neighbor of protein  $i$  that itself has more neighbors will likely have greater influence on protein  $i$  than other of  $i$ ’s neighbors, which we refer to as the “two-hop neighborhood effect.”

## 2.2 Prior and posterior distributions for the network auto-probit model

In conducting inferences with our network auto-probit model, we will utilize the joint posterior probability distribution, conditional on the observed annotations  $\mathbf{y}$ , i.e.,

$$P(\mathbf{z}, \boldsymbol{\mu}, \beta | \mathbf{y}) \propto P(\mathbf{y} | \mathbf{z}) \cdot P(\mathbf{z} | \boldsymbol{\mu}, \beta) \cdot P(\boldsymbol{\mu}) \cdot P(\beta),$$

where  $P(\mathbf{y}|\mathbf{z}) = \prod_{i=1}^n P(y_i|z_i) = 1$ , since we assume for now that the GO annotation is correct and the sign of  $z_i$  fully determines the value of  $y_i$ . To fully specify this posterior, we need to equip the parameters  $\boldsymbol{\mu}$  and  $\beta$  with appropriate prior distributions.

We assign a conditional autoregressive (CAR) prior distribution with a hyperparameter  $\tau^2$  to  $\boldsymbol{\mu}$  which models the spatial dependence on the network and smooths individual  $\mu_i$  locally. More specifically, we define a (singular) joint prior distribution on  $\boldsymbol{\mu}$  of the form

$$P(\boldsymbol{\mu}|\tau^2) \propto \exp\left\{-\frac{1}{2\tau^2}\boldsymbol{\mu}^T L\boldsymbol{\mu}\right\} \propto \exp\left\{-\frac{1}{2}\sum_{i\sim j}\frac{(\mu_i - \mu_j)^2}{\tau^2}\right\}.$$

The conditional distribution for individual  $\mu_i$  is therefore

$$\mu_i|\boldsymbol{\mu}_{[-i]}, \tau^2 \sim N\left(\frac{\sum_{j\sim i} a_{ij}\mu_j}{d_i}, \frac{\tau^2}{d_i}\right),$$

where  $L = D - A$  is the so-called graph Laplacian matrix and  $\boldsymbol{\mu}_{[-i]}$  is all of  $\boldsymbol{\mu}$  except the  $i$ th element  $\mu_i$ .

The hyperparameter  $\tau^2$  can be interpreted as the variance of the difference in expected latent characteristics for two neighbor proteins. To better understand the effect of the  $\tau^2$ , we derive the posterior distribution of  $\boldsymbol{\mu}$ , given the Gaussian process  $\mathbf{z}$  and  $\beta$

$$\begin{aligned} & P(\boldsymbol{\mu}|\mathbf{z}, \beta) \\ \propto & P(\mathbf{z}|\boldsymbol{\mu}, \beta) \cdot P(\boldsymbol{\mu}) \\ \propto & \exp\left\{-\frac{1}{2}(\mathbf{z} - \boldsymbol{\mu})^T (I - \beta B)(\mathbf{z} - \boldsymbol{\mu})\right\} \cdot \exp\left\{-\frac{1}{2}\boldsymbol{\mu}^T \frac{L}{\tau^2}\boldsymbol{\mu}\right\} \\ \propto & \exp\left\{-\frac{1}{2}[\boldsymbol{\mu} - \mathbf{z}^T (I - \beta B)]^T (I - \beta B + \frac{L}{\tau^2})[\boldsymbol{\mu} - \mathbf{z}^T (I - \beta B)]\right\} \\ \sim & MVN\left(\mathbf{z}^T (I - \beta B)(I - \beta B + \frac{L}{\tau^2})^{-1}, (I - \beta B + \frac{L}{\tau^2})^{-1}\right). \end{aligned}$$

The parameter  $\tau^2$  controls the extent to which the prior influences the posterior distribution. Therefore, the choice of  $\tau^2$  is important in identifying the location vector  $\boldsymbol{\mu}$ . We discuss this issue further below.

Due to the constraint on  $\beta$  for obtaining a valid precision matrix, we apply a truncated normal prior to  $\beta$ ,

$$\beta \sim TN(\beta_0, \sigma_\beta; 0, \beta_{\max}),$$

where  $\beta_0$  and  $\sigma_\beta$  are the prior mean and standard deviation, respectively;  $\beta_{\max}$  is the maximal possible value for  $\beta$ , as dictated by the largest eigenvalue

of  $D^{1/2}AD^{1/2}$  in our implementation. We simply set  $\beta_0$  as the midpoint of 0 and  $\beta_{\max}$ ;  $\sigma_\beta$  is chosen to be small so that the truncated normal posterior distribution for  $\beta$  could be comfortably fit on the tight constraint.

### 2.3 Complexity of Smoothing Variances

Proper tuning of the prior distributions in this Bayesian auto-probit model is critical to obtaining useful posterior distributions of the location parameter  $\boldsymbol{\mu}$  and the neighborhood effect parameter  $\beta$ , as discussed earlier. Identification of the location vector  $\boldsymbol{\mu}$  in this over-parameterized model would be difficult without strong spatial smoothing. We develop a degree of freedom for  $\boldsymbol{\mu}$  as a function of  $\tau^2$  to have a better idea of balancing model fitting and smoothing to this end.

For the Gaussian process  $\mathbf{z}$ , we have

$$\mathbf{z} \sim MVN(\boldsymbol{\mu}, (I - \beta B)^{-1}),$$

where  $B = D^{\frac{1}{2}}AD^{\frac{1}{2}}$ . Since the parameter  $\beta$  is chosen to guarantee the positive definiteness of the matrix  $(I - \beta B)$ , there exists a full-rank matrix  $V$  such that  $(I - \beta B) = VV^T$ , or  $V(I - \beta B)V^T = I$ . Thus,

$$V\mathbf{z} \sim MVN(V\boldsymbol{\mu}, V(I - \beta B)V^T),$$

$$V\mathbf{z} \sim MVN(V\boldsymbol{\mu}, I).$$

Recalling that the CAR prior distribution for  $\boldsymbol{\mu}$  has the form

$$\boldsymbol{\mu}|\tau^2 \sim \exp\left\{-\frac{1}{2\tau^2}\boldsymbol{\mu}^T L\boldsymbol{\mu}\right\}, \text{ or}$$

$$\boldsymbol{\mu} \sim MVN(\mathbf{0}, \left(\frac{L}{\tau^2}\right)^{-1}),$$

we propose an effective degrees of freedom, in analogy to the degrees of freedom for the smoother matrix in the smoothing splines (e.g., [7]),

$$\begin{aligned} \rho(\tau^2) &= \text{trace}\left[\left(V^T V + \frac{L}{\tau^2}\right)^{-1} V^T V\right] \\ &= \text{trace}\left[\left(I - \beta B + \frac{L}{\tau^2}\right)^{-1} (I - \beta B)\right], \end{aligned}$$

where  $L = D - A$ , as defined earlier. The degree of freedom for  $\boldsymbol{\mu}$  is monotonely increasing and is confined between 0 and  $N$ . In applications, such as those shown later in this paper, we have found it useful to specify  $\tau^2$  so as to impose a fairly low number of degrees of freedom, forcing  $\boldsymbol{\mu}$  to be fairly smooth.

## 2.4 Markov chain Monte Carlo algorithm

In the applications to follow, we use Markov chain Monte Carlo algorithms to draw samples from the joint posterior distribution. We update the individual  $z_i$ ,  $\mu_i$ , and the parameter  $\beta$  one at a time, which requires access to the fully conditional distributions.

The Gibbs sampler is used to update the  $z_i$ 's, based on the conditional probability

$$P(z_i | \mathbf{z}_{[-i]}, \boldsymbol{\mu}, \beta, \mathbf{y}) = \begin{cases} \frac{1}{\sqrt{2\pi\Phi(\bar{z}_i)}} \exp\{-\frac{1}{2}(z_i - \bar{z}_i)^2\}, & z_i \geq 0, \quad y_i = 1, \\ 0, & z_i < 0, \quad y_i = 1, \\ \frac{1}{\sqrt{2\pi(1-\Phi(\bar{z}_i))}} \exp\{-\frac{1}{2}(z_i - \bar{z}_i)^2\}, & z_i < 0, \quad y_i = 0, \\ 0, & z_i \geq 0, \quad y_i = 0, \end{cases} \quad (2)$$

where  $\bar{z}_i = \mu_i + \beta \sum_{j \sim i} \sqrt{d_i d_j} a_{ij} (z_j - \mu_j)$ ,  $\mathbf{z}_{[-i]}$  is all of  $\mathbf{z}$  except the  $i$ th element  $z_i$ , and  $\Phi$  is the standard normal cumulative density function.

The Metropolis-Hasting algorithm is used to update the  $\mu_i$ 's and  $\beta$ . The proposal change from  $\mu_i$  to a new  $\mu'_i$  (or from  $\beta$  to a new  $\beta'$ ) is drawn from a normal distribution centered at the current value with a pre-defined standard deviation.

## 3 Incorporating uncertainty of functional annotation in Gene Ontology

Our model thus far assumes that the annotations from the Gene Ontology database reflect the actual protein functional status. That is, we have assumed that  $y_i$ , the observed annotation for protein  $i$ , is consistent with its true functional label, as captured by the sign of  $z_i$ . However, it has been pointed out that a protein currently not having a functional annotation of interest does not necessarily mean that it does not have the function, but rather that its functional status is simply unknown to the best of our knowledge ([15]). In other words, with respect to the notation established above, the probability that  $y_i = 0$ , given  $z_i \geq 0$ , can be nontrivial.

Therefore, we modify our model, writing

$$P(\mathbf{y} | \mathbf{z}, g) = \prod_{i=1}^n P(y_i | z_i, g),$$

where

$$P(y_i = 0 | z_i, g) = \begin{cases} 1, & z_i < 0, \\ g, & z_i \geq 0. \end{cases}$$

This extended version of our network auto-probit model thus incorporates the probability of being “incorrectly un-annotated”. The device employed here is analogous to that used in [22], in modeling the spatial distribution of toads, to account for the fact that spatial regions for which no toads were observed are not necessarily devoid of toads.

The joint posterior distribution including  $g$  is given by

$$P(\mathbf{z}, \boldsymbol{\mu}, \beta, g | \mathbf{y}) \propto P(\mathbf{y} | \mathbf{z}, g) \cdot P(\mathbf{z} | \boldsymbol{\mu}, \beta) \cdot P(\boldsymbol{\mu}) \cdot P(\beta) \cdot P(g),$$

where we let  $P(g)$  be the uniform prior distribution for  $g$ ,  $P(g) \sim U(0, 1)$ ,  $P(\mathbf{y} | \mathbf{z}, g)$  is as described above, and all other terms are the same as mentioned earlier.

### 3.1 Markov Chain Monte Carlo algorithm with the GO annotation uncertainty

When we consider the Gene Ontology annotation uncertainty and include the probability of being “incorrectly un-annotated”,  $g$ , the fully conditional distribution for updating individual  $z_i$ 's is different from before, being expressed as

$$P(z_i | \mathbf{z}_{[-i]}, \boldsymbol{\mu}, \beta, g, \mathbf{y}) = \begin{cases} \frac{1}{\sqrt{2\pi}\Phi(\bar{z}_i)} \exp\{-\frac{1}{2}(z_i - \bar{z}_i)^2\}, & z_i \geq 0, \quad y_i = 1, \\ 0, & z_i < 0, \quad y_i = 1, \\ \frac{g}{\sqrt{2\pi}[1-\Phi(\bar{z}_i)+g\Phi(\bar{z}_i)]} \exp\{-\frac{1}{2}(z_i - \bar{z}_i)^2\}, & z_i \geq 0, \quad y_i = 0, \\ \frac{1}{\sqrt{2\pi}[1-\Phi(\bar{z}_i)+g\Phi(\bar{z}_i)]} \exp\{-\frac{1}{2}(z_i - \bar{z}_i)^2\}, & z_i < 0, \quad y_i = 0, \end{cases}$$

where  $\bar{z}_i = \mu_i + \beta \sum_{j \sim i} \sqrt{d_i d_j} a_{ij} (z_j - \mu_j)$ , and  $\Phi$  is the standard normal cumulative density function. The derivation of the conditional probability is as follows.

$$\begin{aligned} & P(z_i | \mathbf{z}_{[-i]}, \boldsymbol{\mu}, \beta, g, \mathbf{y}) \\ = & \frac{P(\mathbf{y} | \mathbf{z}, g) \cdot P(z_i | \mathbf{z}_{[-i]}, \boldsymbol{\mu}, \beta)}{\int_{z_i} P(\mathbf{y}, z_i | \mathbf{z}_{[-i]}, \boldsymbol{\mu}, \beta, g) dz_i} \\ = & \frac{P(y_i | z_i, g)}{\int_{z_i} P(y_i | z_i, g) \cdot P(z_i | \mathbf{z}_{[-i]}, \boldsymbol{\mu}, \beta) dz_i} P(z_i | \mathbf{z}_{[-i]}, \boldsymbol{\mu}, \beta) \\ = & C \cdot P(z_i | \mathbf{z}_{[-i]}, \boldsymbol{\mu}, \beta), \end{aligned}$$

where

$$\begin{aligned}
C &= \frac{P(y_i|z_i, g)}{\int_{z_i} P(y_i|z_i, g) \cdot P(z_i|\mathbf{z}_{[-i]}, \boldsymbol{\mu}, \beta) dz_i} \\
&= \frac{P(y_i|z_i, g)}{\int_{z_i \geq 0} P(y_i|z_i, g) \cdot P(z_i|\mathbf{z}_{[-i]}, \boldsymbol{\mu}, \beta) dz_i + \int_{z_i < 0} P(y_i|z_i, g) \cdot P(z_i|\mathbf{z}_{[-i]}, \boldsymbol{\mu}, \beta) dz_i} \\
&= \begin{cases} \frac{\frac{P(y_i|z_i, g)}{\int_{z_i \geq 0} 1 \cdot P(z_i|\mathbf{z}_{[-i]}, \boldsymbol{\mu}, \beta) dz_i + \int_{z_i < 0} 0 \cdot P(z_i|\mathbf{z}_{[-i]}, \boldsymbol{\mu}, \beta) dz_i}}{\frac{P(y_i|z_i, g)}{\int_{z_i \geq 0} g \cdot P(z_i|\mathbf{z}_{[-i]}, \boldsymbol{\mu}, \beta) dz_i + \int_{z_i < 0} 1 \cdot P(z_i|\mathbf{z}_{[-i]}, \boldsymbol{\mu}, \beta) dz_i}}, & \text{when } y_i = 1, \\ \frac{\frac{P(y_i|z_i, g)}{\int_{z_i \geq 0} g \cdot P(z_i|\mathbf{z}_{[-i]}, \boldsymbol{\mu}, \beta) dz_i + \int_{z_i < 0} 1 \cdot P(z_i|\mathbf{z}_{[-i]}, \boldsymbol{\mu}, \beta) dz_i}}{\frac{1}{g\Phi(\bar{z}_i) + 1 - \Phi(\bar{z}_i)}}, & \text{when } y_i = 0, \end{cases} \\
&= \begin{cases} \frac{1}{\Phi(\bar{z}_i)}, & \text{when } y_i = 1, \quad z_i \geq 0, \\ 0, & \text{when } y_i = 1, \quad z_i < 0, \\ \frac{g}{g\Phi(\bar{z}_i) + 1 - \Phi(\bar{z}_i)}, & \text{when } y_i = 0, \quad z_i \geq 0, \\ \frac{1}{g\Phi(\bar{z}_i) + 1 - \Phi(\bar{z}_i)}, & \text{when } y_i = 0, \quad z_i < 0. \end{cases}
\end{aligned}$$

The Gibbs sampler can be used to update  $g$ , the fully conditional distribution of which is a beta distribution,

$$P(g|\mathbf{z}, \boldsymbol{\mu}, \beta, \mathbf{y}) \propto P(\mathbf{y}|\mathbf{z}, g) \cdot P(g) \propto g^{N_{-+}} (1-g)^{N_{++}},$$

where  $N_{-+} = \#\{i : y_i = 0, z_i \geq 0\}$ ,  $N_{++} = \#\{i : y_i = +1, z_i \geq 0\}$ .

## 4 Results

### 4.1 Data

We have implemented our network auto-probit models on two yeast protein-protein association (sub)networks extracted from the STRING database, introduced in [16]. STRING contains known and predicted protein-protein associations, where ‘‘association’’ refers to both direct physical binding and indirect interaction such as participation in the same metabolic pathway or cellular process. Information for associations is obtained from 7 evidence sources: *database imports*<sup>1</sup>, *high-throughput experiments*, *co-expression*, *homology based on phylogenetic co-occurrence*, *homology based on gene fusion events*, *homology based on conserved genomic neighborhood*, and *text mining*.

STRING simplifies the access to protein-protein associations by providing a comprehensive collection of protein-protein associations for a large number of organisms. A score  $S$  is assigned to each interacting pair of proteins by bench-marking against the KEGG pathway from [12]. The score is

---

<sup>1</sup>PPI and pathway databases

calculated by  $1 - S = \prod_i(1 - S_i)$ , where  $i$  indicates the individual evidence type described above, and  $S_i$  is the score from the  $i$ -th source. As a result, STRING database provides users with weighted undirected protein-protein association networks.

For purposes of illustration, we extract two such networks of different sizes from the yeast genome and study different functions for them. The smaller one contains  $N = 211$  genes, all of which are annotated with the term *GO:0007154, cell communication* from the Gene Ontology database, updated as of November 2007. To ease exposition we name this the “CC network”. The term we wish to predict for this network is *GO:0007242, intracellular signaling cascade*, a grandchild term of *cell communication*. The larger network, named the “OOB network”, consists of  $N = 975$  genes, all annotated with *GO:0006996, organelle organization and biogenesis*. We study one of its child terms, *GO:0051276, chromosome organization and biogenesis*. Note that according to the GO annotation in November 2007, there are 226 and 1290 proteins annotated with *cell communication* and *organelle organization and biogenesis*, respectively. We only use the largest connected components, neglecting small connected neighborhoods and isolated proteins, therefore, the network sizes in this paper are smaller than the actual numbers of proteins.

## 4.2 Parameter estimation and prediction by the auto-probit model

Using the model in Section 2.2 and 2.3, we first take the observed annotation  $\mathbf{y}$  to be known for all  $N$  proteins, and examine the issue of fitting the model. The hyperparameters for the two networks are listed in Table 1. We discard the first 1000 iterations as burn-in and run 9000 iterations to get posterior samples. Convergence diagnostics by standard approaches indicate that all chains reach equilibrium.

Estimates of the posterior means of some parameters are given in Table 2 and 3, together with 95% credibility intervals. It can be seen that there is statistically significant positive spatial dependence on both networks. The small estimates for  $\beta$  are a result of the large eigenvalues  $\lambda_{\max}$  (312.9047 and 4297.8 for CC and OOB networks, respectively), and hence a small  $\beta_{\max}$  (0.0032 and 0.0002 for CC and OOB networks, respectively).

Figure 1 contains the histograms of the posterior estimates of probabilities of having the target function, given the observed GO annotations, for the two networks. Blue histograms are based on proteins which are not annotated with the term in question while purple ones are for annotated

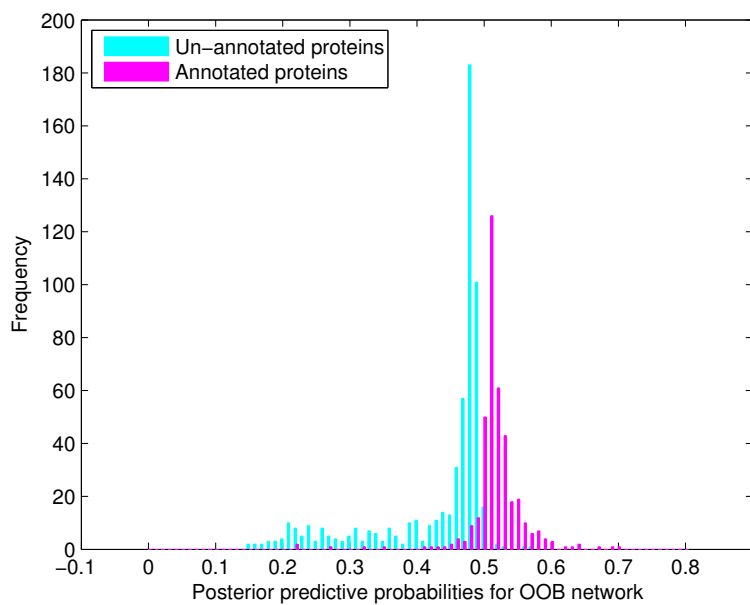
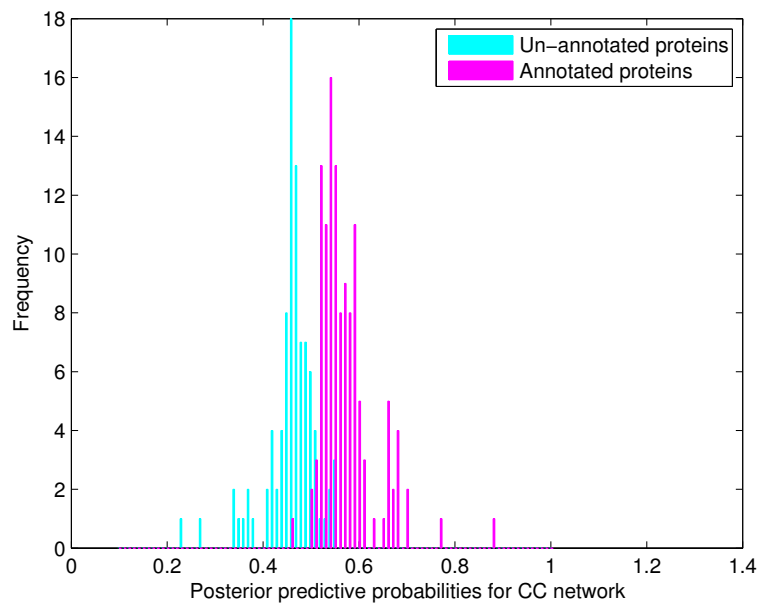


Figure 1: Histogram of the posterior estimates of the probability of having the target function. [Top]: *intracellular signaling cascade* in the CC network; [Bottom]: *chromosome organization and biogenesis* in the OOB network.

Table 1: Hyperparameter setup

	CC network	OOB network
$\tau^2$	0.30	1.00
$\sigma_\mu$ in proposal	0.4	0.4
$\beta_{\min}$	0	0
$\beta_{\max}$	0.0032	0.0002
$\sigma_\beta$ in proposal	0.0001	0.0001

Table 2: Estimated posterior means, Monte Carlo standard errors and 95% credibility intervals for the CC network with the model in Section 2

CC network			
Parameter	Degree	Estimate	95% credibility interval
$\beta$	—	0.0017	(0.0015, 0.0019)
$\mu_{YCR026C}$	1	-0.212	(-0.227, -0.197)
$\mu_{YCR038C}$	15	0.158	(0.153, 0.162)
$\mu_{YGL035C}$	25	0.082	(0.078, 0.086)
$\mu_{YMR037C}$	40	0.136	(0.132, 0.139)
$\mu_{YLR229C}$	58	0.157	(0.154, 0.160)

proteins. The histograms of the posterior estimates of  $\mu$  are shown in the supplementary material, which can be found at [math.bu.edu/people/xiaoyu](http://math.bu.edu/people/xiaoyu).

Interestingly, in those plots, the histograms for the two classes of proteins are well separated, with the posterior mean of the un-annotated proteins lower than that of the annotated proteins in both cases, and little overlapping areas between the two classes. This indicates that the auto-probit model is capable of distinguishing proteins with different functional status, “annotated” and “un-annotated”, by utilizing the network topology and estimating the parameters in a globally coherent fashion.

We perform a sanity check to study the predictive performance of the auto-probit model. We compare the network-based auto-probit model with the nearest-neighbor (NN) algorithm, a standard algorithm representing typical methods using local neighborhood information based on the *guilt-by-association* principal, and the kernel logistic method in [23], which builds a Laplacian kernel matrix with the weighted network by  $L = D - A$  and produces predictive probabilities. Since NN does not use the weights on the

Table 3: Estimated posterior means, Monte Carlo standard errors and 95% credibility intervals for the OOB network with the model in Section 2

OOB network			
Parameter	Degree	Estimate	95% credibility interval
$\beta$	—	$1.4764 \times 10^{-4}$	$(1.2608 \times 10^{-4}, 1.6921 \times 10^{-4})$
$\mu_{YBR172C}$	1	0.434	(0.374, 0.495)
$\mu_{YAL010C}$	15	-0.062	(-0.125, 0.000)
$\mu_{YLR068W}$	50	0.129	(0.072, 0.186)
$\mu_{YBL002W}$	100	-0.351	(-0.411, -0.292)
$\mu_{YLR175W}$	202	0.221	(0.161, 0.282)

edges, we use the induced binary networks. It produces a number between 0 and 1 (i.e., the fraction of a protein’s neighbors in the binary network possessing the term in question). To evaluate and compare method performances, we plot the standard Receiver Operating Characteristic (ROC) curves based on a 10-fold cross validation on both networks. More specifically, we calculate true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN) for varying predictive thresholds based on the proteins in each fold, and then average across folds. The comparisons are presented in Figure 2. The ROC curves show that the auto-probit model works at least equally well for both networks as the simple nearest-neighbor algorithm, and the more sophisticated logistic kernel method. The similarity in performance of these three rather different methods is most likely due, at least in part, to the fact that sequence similarity is the dominating information source for the protein-protein association. As pointed out in [17], it is hard to distinguish methods relying on protein-protein association data when this is the case.

One of the advantages of the auto-probit model is that it provides a natural platform of incorporating the weights on the edges in the networks. Conversely, it is not straightforward for methods based on the neighborhood counting principle. To show such an advantage, we apply the auto-probit model on both the weighted and binary versions of the two networks, and compare the performances by ROC curves, as in Figure 2. In both cases, there is a clear jump from using only the embedded binary networks (the purple curves in Figure 2) to the weighted networks (the red curves in Figure 2).

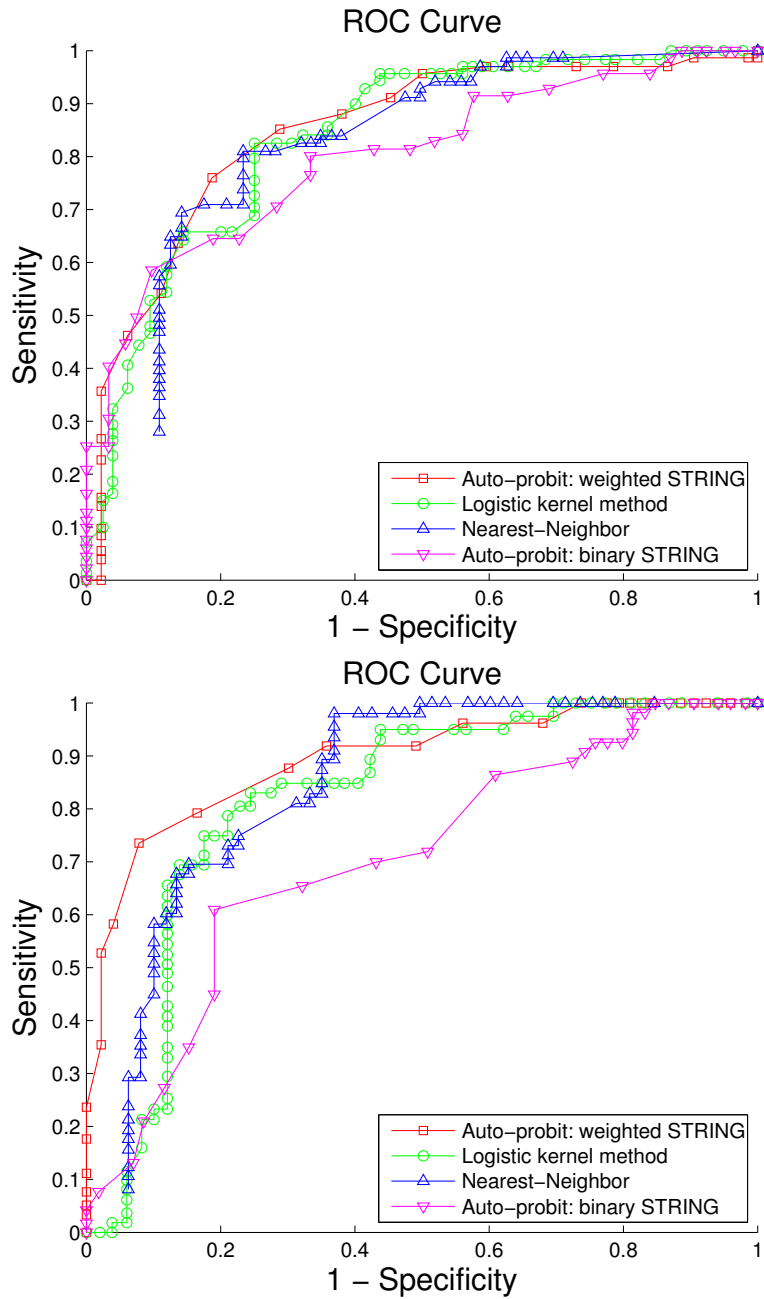


Figure 2: Comparing the auto-probit model and the Nearest-Neighbor algorithm by ROC curves based on a 10-fold cross-validation. [Top]: *intracellular signaling cascade* in the CC network; [Bottom]: *chromosome organization and biogenesis* in the OOB network.

### 4.3 Analysis of the Gene Ontology annotation uncertainty

What significantly distinguishes our network-based auto-probit model for gene function prediction from most other methods is the ease with which uncertainty information can be incorporated, which allows us, in particular, to explore the problem of GO annotation uncertainty.

We first fit the model introduced in Section 3 to the two networks CC and OOB, and conduct posterior inference on  $g$ , i.e., the probability that a protein is currently not annotated with the function of study but actually has the function. The hyperparameter setup here is the same as before. We run 9000 iterations after 1000 burn-in. Results from diagnostic methods indicate that both chains are converged. The trace plots and the histograms of the posterior estimates are shown in Figure 3. The posterior means of  $g$  are 0.1142 and 0.4186 for the CC network and the OOB network, respectively. Hence, we see that while in the CC network the rate of false negative annotations is estimated to be fairly small, that in the OOB network is estimated to be quite substantial – more than two out of every five.

Knowledge of these false negative rates may be in turn propagated through the process of posterior-based prediction to produce more accurate predictions based on currently “flawed” annotations. In order to illustrate, we use a set of annotations from the Gene Ontology database updated in June 2006 to estimate the parameters and to predict the target functions. Then we evaluate the predictions according to the more recently updated annotations used in the analysis of the previous section (i.e., updated in November 2007). There are 16 proteins in the CC network updated with new annotations to *intracellular signaling cascade* in November 2007; 38 proteins for the OOB network. Predictions were made for these new annotations using our network auto-probit model with and without incorporating GO annotation uncertainty. Method performance is evaluated by *recall* (or *sensitivity*), defined as the ratio of the true positive and the predicted positive, or,

$$recall = \frac{TP}{TP + FN} = sensitivity.$$

Our decision to use recall is due to an arguably greater interest among scientists in discovering previously unknown functions of proteins, rather than confirming known functions. A method which correctly detects more proteins annotated with the target function is preferred. Recall from both methods are calculated and plotted versus the different predictive thresholds.

Figure 4 contains the plots of recall versus the predictive thresholds for the two networks. The network auto-probit model with  $g$  (i.e., incorporat-

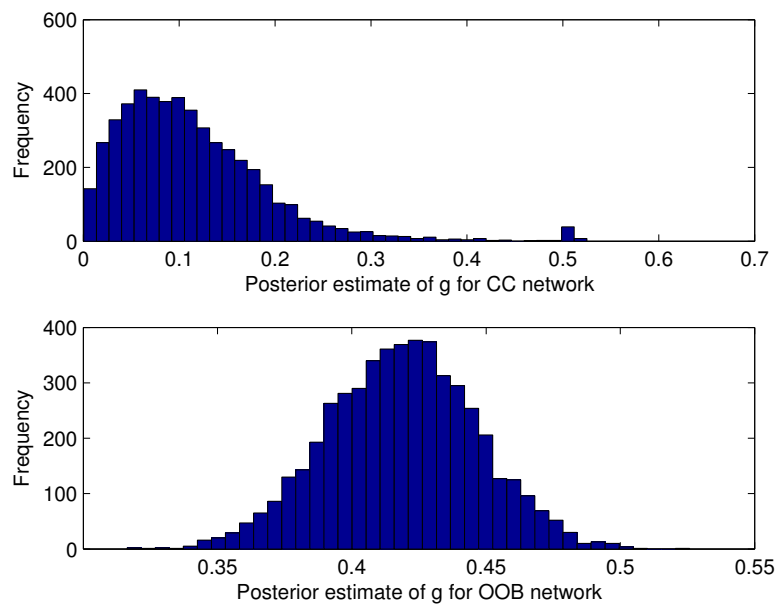


Figure 3: Histograms of the posterior estimates of  $g$ , the probability of being incorrectly un-annotated. [Top]: *intracellular signaling cascade* in the CC network; [Bottom]: *chromosome organization and biogenesis* in the OOB network.

ing annotation uncertainty) provides improvements in both cases. For the CC network, the improvement is mostly for small predictive thresholds (less than 0.5). The improvement for the OOB network is decidedly more substantial. If we use a predictive threshold of 0.5, a common choice, the recall from the auto-probit model with  $g$  is 0.6794, which leads to nearly 200% improvement from the model without  $g$  with recall being 0.2265. These results demonstrate that the auto-probit framework not only provides a solution to the problem of GO annotation uncertainty by modeling the probability of being “incorrectly un-annotated”, but also produces high quality predictions based on low quality annotations. Essentially, incorporation of the additional parameter  $g$  allows the methodology the possibility of rectifying apparent inconsistencies in vertex labeling, as suggested by the graph topology.

## 5 Discussion

This paper introduces a network-based fully Bayesian auto-probit model for protein function prediction. It takes protein-protein association networks as input and employs a latent Gaussian process  $\mathbf{z}$  to encode proteins functional similarity. Using a hierarchical Bayesian model, we assign a conditional autoregressive (CAR) prior distribution with a single hyperparameter  $\tau^2$  to the location vector  $\boldsymbol{\mu}$  of the Gaussian process. There are various extensions that may be of interest.

For example, the use of a single  $\tau^2$  may not be flexible enough for the nodes in the network with greater neighborhood sizes. To allow spatial smoothing to vary more freely, we could extend the prior distribution on  $\boldsymbol{\mu}$  to have a different variance for each neighbor pair, in analogy to [19]. In addition, here we predict protein function only term-by-term, but GO terms are organized according to a directed acyclic graph (DAG), reflecting their ontological relationships with each other. Recent work has shown that some improvement in protein function prediction can be obtained by exploiting the structure among GO terms. See [11], for example, and references therein. In principle, the network auto-probit model proposed here can be extended in an analogous manner.

The incorporation of uncertainty in negative GO term annotations was shown here to yield substantial improvements in predictive accuracy. This observation has powerful implications, since the tendency toward emphasis on “positive results” in science, and the manner in which modern biological databases encode those results, means that this issue is not restricted to the

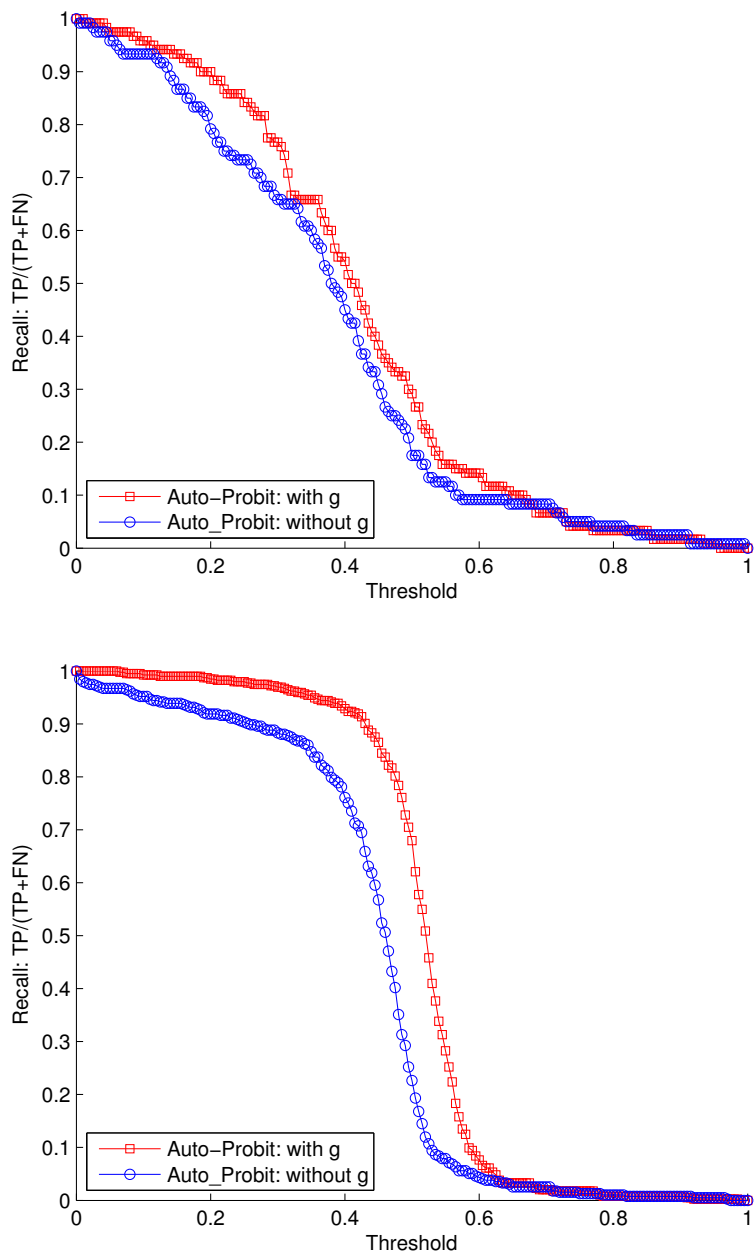


Figure 4: Plots of recall versus thresholds considering the probability of being incorrectly un-annotated  $g$ . [Top]: *intracellular signaling cascade* in the CC network; [Bottom]: *chromosome organization and biogenesis* in the OOB network.

Gene Ontology database alone, but rather is likely endemic to the area as a whole. Other important forms of uncertainty include, for example, inaccuracies in protein interaction networks. The STRING database is an example of work in this area that seeks to use other evidence in addition to interaction experiments to determine protein-protein interactions. Although here we have incorporated this information in the form of a weighted network, our network auto-probit model is structured in such a way that it should facilitate the incorporation of such information in a more nuanced fashion through an additional layer of probability modeling, if desired.

## Acknowledgements

The authors thank Brian Reich for helpful discussion at the start of this project. This work was supported in part by NIH grant R01 HG003367-01A1, NSF ITF 0428715, NSF DMS-0602204 (EMSW21-RTG, BioDynamics at Boston University), and ONR award N00014-06-1-0096.

## References

- [1] Besag JE and York J and Mollié A. (1991). Bayesian image restoration with applications in spatial statistics (with discussion). *Ann. Inst. Statist. Math.* **43**, 21-59.
- [2] Chua HN and Sung WK and Wong L. (2006). Exploiting indirect neighbours and topological weight to predict protein function from protein-protein interactions. *Bioinformatics* **22**, 1623-1630.
- [3] Cressie NAC. (c1991). *Wiley Series in Probability and Statistics*. New York: J Wiley.
- [4] Deng M and Chen T and Sun F. (2004). An integrated analysis of protein function prediction. *Journal of Computational Biology* **11**, 463-475.
- [5] Friedman JH. (1997). On bias, variance, 0/1-loss, and the curse-of-dimensionality. *Data Mining and Knowledge Discovery* **1**, 55-77.
- [6] Gelman A and Rubin DB. (1992). Inference from iterative simulation using multiple sequences. *Statistical Science* **7**, 457-472.
- [7] Hastie T and Tibshirani R and Friedman J. (2001). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* Springer .

- [8] Hishigaki H and Nakai K and Ono T and Tanigami A and Takagi T. (2001.) Assessment of prediction accuracy of protein function from protein-protein interaction data. *Yeast* **18**, 523-531.
- [9] Hodges JS and Sargent DJ. (2001). Counting degrees of freedom in hierarchical and other richly-parameterised models. *Biometrika* **88**, 367-379.
- [10] Jiang X and Nariai N and Steffen M and Kasif S and Gold D and Kolaczyk ED. (2008b). Combining hierarchical inference in ontologies with heterogeneous data sources improves gene function prediction. *Proceedings of the IEEE International Conference on Bioinformatics and Biomedicine*.
- [11] Jiang X and Nariai N and Steffen M and Kasif S and Kolaczyk ED. (2008a). Integration of relational and hierarchical network information for protein function prediction. *BMC Bioinformatics* **9**, 350.
- [12] Kanehisa M and Goto S and Kawashima S and Okuno Y and Hattori M. (2004). The KEGG resource for deciphering the genome. *Nucleic Acids Research* **32**, D277-D280.
- [13] Lanckriet GRC and Cristianini N and Jordan MI and Nobel WS. (2004). A statistical framework for genomic data fusion. *Bioinformatics*, **20**, 2626-2635.
- [14] Lee I and Date SV and Adai AT and Marcotte EM. (2004). A probabilistic functional network of yeast genes. *Science* **306**, 1555-1558.
- [15] Letovsky S and Kasif S. (2003). Predicting protein function from protein-protein interaction data: a probabilistic approach. *Bioinformatics* **19**, i197-i204.
- [16] Mering CV et al. (2005). STRING: known and predicted protein-protein associations, integrated and transferred across organisms. *Nucleic Acids Research* **33**, D433-D437.
- [17] Peña-Castillo et al. (2008). A critical assessment of *Mus musculus* gene function prediction using integrated genomic evidence *Genome Biology* **9**, S2.
- [18] Raftery AE and Lewis SM. (1992). How many iterations in the Gibbs sampler? *Bayesian Statistics 4*, Oxford University Press, 763-773.

- [19] Reich BJ and Hodges JS. (2008). Modeling longitudinal spatial periodontal data: A spatially-adaptive model with tools for specifying priors and checking fit. *Biometrics* **64**, 790-799.
- [20] Schwikowski B and Uetz P and Fields S. (2000). A network of protein-protein interactions in yeast. *Nature Biotechnology* **18**, 1257-1261.
- [21] Sharan R and Ulitsky I and Shamir R (2007). Network-based prediction of protein function *Molecular Systems Biology* **3**, 88.
- [22] Weir IS and Pettitt AN, (2000). Binary Probability Maps Using a Hidden Conditional Autoregressive Gaussian Process with an Application to Finnish Common Toad Data. *Applied Statistics* **49**, 473-484.
- [23] Zhu J and Hastie T. (2005). Kernel logistic regression and the import vector machine. *Journal of Computational and Graphical Statistics* **14**, 185-205.