A Bayesian Framework for Combining Protein and Network Topology Information for Predicting Protein-Protein Interactions

Adriana Birlutiu, Florence d'Alché-Buc and Tom Heskes

Adriana Birlutiu is with the Institute for Computing and Information Sciences, Radboud University Nijmegen, The Netherlands and the Faculty of Science, "1 Decembrie 1918" University, Alba-Iulia, Romania, e-mail: adrianab@cs.ru.nl.

Florence d'Alché-Buc is with IBISC, Université d'Evry-Val d'Essonne, Genopole, France, email: florence.dalche@ibisc.fr.

Tom Heskes is with the Institute for Computing and Information Sciences, Radboud University Nijmegen, The Netherlands, email: tomh@cs.ru.nl.

A Bayesian Framework for Combining Protein and Network Topology Information for Predicting Protein-Protein Interactions

***Abstract*—Computational methods for predicting protein-protein interactions are important tools that can complement high-throughput technologies and guide biologists in designing new laboratory experiments. The proteins and the interactions between them can be described by a network which is characterized by several topological properties. Information about proteins and interactions between them, in combination with knowledge about topological properties of the network, can be used for developing computational methods that can accurately predict unknown protein-protein interactions. This paper presents a supervised learning framework based on Bayesian inference for combining two types of information: *i)* network topology information, and *ii)* information related to proteins and the interactions between them. The motivation of our model is that by combining these two types of information one can achieve a better accuracy in predicting protein-protein interactions, than by using models constructed from these two types of information independently.**

***Index Terms*—Bayesian methods, protein-protein interaction, network analysis, topology**

I. Introduction

The reconstruction of biological networks is currently an active research subject with important applications ranging from basic biology to medical applications. The term "biological network" is used for graphs in which vertices represent genes or proteins and edges represent various types of interactions between them. Since most of the cellular components exert their functions through interactions with other components, inferring biological networks is crucial for the understanding of the cellular functions and biological processes inside a living cell. The reconstruction of these biological networks is a major challenge with important applications. For example, knowledge about which proteins interact in the human proteome would highlight key proteins that interact with many partners which could be interesting drug targets, this application refers to the new emerging field of network medicine [1], which is a network based approach to human disease.

Research in molecular biology and genetics has already provided a partial view of these biological networks. The presence of edges in the network can only be established by costly and tedious laboratory experiments. However, recent high-throughput technologies, like for example, the yeast two-hybrid systems for protein-protein interactions (PPIs), provide large numbers of likely edges in these graphs, but with a high rate of false positives. Therefore, much work remains to complete (adding currently unknown edges) and correct (removing false positive edges) these partially known networks. These issues can be addressed using computational methods which can make predictions about the presence or absence of edges in the network. Accurately predicting which proteins might interact can help in correcting the wrong edges and in completing the network by designing and guiding future laboratory experiments.

Computational techniques are based on the idea that information about individual genes and proteins, such as their sequence, structure, subcellular localization, or level of expression across several experiments, can provide useful hints about the presence or absence of interactions between them, thus about edges in the network. For example, two proteins are more likely to physically interact if they are expressed in similar experiments, and localized in the same cellular compartment. Following this line of thought, computational methods which take roots in machine learning have been proposed in recent years for inferring biological networks. There are two approaches in computational methods for PPI prediction: *i)* the unsupervised approach which estimates the edges solely from data related to nodes [2], and *ii)* the supervised approach for which the presence or absence of edges is known for part of the network, and this information is used for inferring the unknown edges. This last setting is becoming more realistic as high confidence networks have become increasingly available. In this article we consider the supervised learning setting, in which case PPI prediction can be seen as a pattern recognition problem, namely to find patterns in the interacting protein pairs that do not exist in the non-interacting pairs. In this pattern recognition problem, information about proteins and labels for protein pairs as interacting or not, supervise the estimation of a function that can predict whether an interaction exists or not between two proteins. This can be further framed as a binary classification problem which takes as input a set of features for a protein pair and gives as output a label: interact or non-interact. Binary classification has been studied extensively in machine learning community, and many algorithms designed to solve it have been also applied for predicting PPIs, including Bayesian networks [3], kernel-based methods [4], [5], logistic regression [6], [7], decision trees and random forest based methods [8], [9], [10], metric or kernel learning [5] and [11], [12], [13].

The performance of the computational methods for network reconstruction, i.e., how well they predict the presence/absence of edges in the network, depends on the quantity and quality of the available training data. The more information one has, preferably from a multitude of independent sources, the more accurate one can predict and

the better one can decide [14]. To get the most out of the available information, computational methods should be capable to incorporate any type of data. In addition to information about proteins and interactions between them, PPI networks are characterized by several topological properties [15], [16], [17], [18]. Network topology can uncover important biological information that is independent of other available biological information [19], [20]. One of the most important topological properties is the existence of a few nodes in the network, called hubs, which have many links with the other nodes, while most of the nodes have just a few links. This characteristic is present in PPI networks and also in other real-world networks, such as the internet [21] and citation networks [22]. Summarizing, we can distinguish two types of information that can be used for predicting PPIs: first, information about proteins and labels for protein pairs as interacting or not, and second, information about the topological properties of the network. These two sources of information are both valuable and can complement each other for constructing accurate models for predicting interactions between proteins. We combine methods that have been previously used for modeling each type of information separately. We use a random graph generator for addressing the topology information and a naive Bayes model for addressing the feature information. Computational tractability was the reason behind the model choices made. The combined model is constructed in a Bayesian framework in which the a priori information about network topology is incorporated in a supervised algorithm for PPI prediction. We prove that by making a few simplifying assumptions, both topological and protein information can be incorporated and we show experimentally that this combination improves the prediction accuracy in PPI networks.

This paper is structured as follows. We finish this section by discussing related work and some terminology and notation that will be used throughout the paper. Section 2 describes the Bayesian framework that combines protein and network topology information for predicting protein-protein interactions. Section 3 presents the experimental evaluation of our model and comparisons with other approaches. We conclude in Section 4 and discuss some possible directions for future research.

A. Related work

The two approaches, unsupervised and supervised, for developing computational methods for PPI prediction will be discussed in the following.

On the one hand, the unsupervised approaches reconstruct PPI networks solely based on a set of protein attributes. In this category there are approaches which investigate the use of topology information. Not specifically to PPI networks, but to graphs in generals, [23] define a kernel-based framework for unsupervised structured network inference taking into account the graph topology by defining kernels based on topological properties such as degree, closeness centrality, betweenness centrality and shortest path. [24] propose a multi-way spectral clustering method for link prediction in biological and social networks. [25] reconstruct gene regulatory networks from gene expression data by proposing a structure prior which incorporates the scale-free property. [26] propose a likelihood method in order to fit a hybrid preferential attachment model to some protein-protein interaction networks, obtaining estimates of the model parameters. [27] investigate the incorporation of different types of topology information, such as network motifs. [28], [29] use l_1-type regularization to encourage sparse structures in the graph learned. In [30] we introduced a framework for incorporating both topology and feature information that forms the basis of the current paper, which extends [30] by a substantial analysis of the topology learning using the Bayesian framework and experimental comparisons of our model with other state-of-the-art methods including SVM, decision tree, random forest.

Our approach is different from the other papers mentioned above, in that we are not considering an unsupervised learning setting, but we also use the information about labels of protein pairs, if they interact or not. Topology only has been shown to be able to characterize drug-targets in PPI networks [31], to predict protein functions [32] and PPIs [33], [34] and to complement sequence information in various biological tasks, like for example, homology detection [35].

On the other hand, several supervised learning approaches such as EM-based approach [36], mixture-of-experts approach [37] and metric or kernel learning [38], [11], [12] have been proposed for network inference. Most of them do not use the topology information as an input feature but some of them take it into account during the training phase to structure the output space [38], [11], [12]. Supervised network inference from integrating different types of data is investigated in [5], [39]. Treating PPI prediction as a supervised inference problem is not straightforward since data is typically associated to individual proteins while the labels correspond to pairs of proteins. This issue is addressed by constructing features for pairs of proteins from individual protein features. Features for protein pairs can be constructed from various sources of information, for example, protein sequence, gene expression data, functional properties of proteins, etc. These sources of information are heterogeneous, however they can be combined such that the heterogeneity is taken into account by using a mixture of feature experts method [37], resulting in an enriched set of features. We used these types of features for one data set in the experimental evaluation. Furthermore, informative features for predicting PPIs can be constructed from sequence information only, for example [40]. Each protein is represented by a vector of pairwise similarities against large subsequences of amino acids created by a shifting window which passes over concatenated protein training sequences. Each coordinate of this vector is the E-value of the Smith-Waterman score [41]. In many cases the problem of identifying high-quality features can in itself be quite difficult and has led to the development of new kernels appropriate to encode a protein's properties. Once a kernel

between proteins is defined, pairwise kernels between pairs of unordered proteins can be defined based on tensor products such as proposed by [4] for protein function prediction, further investigated and analyzed by [42]. Finally, extension of supervised network inference methods to other machine learning paradigms as active learning, multi-task learning, and semi-supervised learning, have also been employed for improving the prediction of PPIs [43], [44], [45], [13]. Furthermore, some approaches [46] take into account the unbalancedness of data when predicting PPI and others [47] consider the protein structure when making the predictions.

B. Notation

The terms "network" and "graph" are used synonymously throughout this paper. An undirected network $G = (V, E)$ is a mathematical object defined by a set of nodes $V = \{v_1, v_2, \ldots, v_N\}$ together with a set of undirected edges E consisting of unordered pairs $\{v_i, v_j\}, v_i \neq v_j$ taken from V. Boldface notation is used for vectors and matrices and normal fonts for their components. Upperscripts are used to distinguish between different vectors or matrices and lowerscripts to address their components. Capital letters are used for constants and small letters for indices, e.g., $i = 1, \ldots, I$. The notation $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is used for a multivariate Gaussian with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. The transpose of a matrix $\boldsymbol{M}$ is denoted by $\boldsymbol{M}^T$.

II. Methods

The data that is being considered consists of a list of proteins and the information associated to them: each protein is represented by a numerical vector which encodes the features associated to that protein (this vector can be constructed from, for example, gene expression data, protein sequence, etc.) and labels for protein pairs as interacting or not (the information about the labels can be seen as an adjacency matrix associated to the list of proteins). In addition, there is information about the topological properties of the network, for example, the fact that in PPI networks there are just a few hubs and that the majority of nodes in the network have just a few connections. The goal of the methods that we propose is to show that by using both types of information, topology and feature information, the prediction accuracy in PPI networks can be improved.

The approach that we use to join topology and feature information is graphically summarized in Figure 1. It consists of a random graph generator model and a naive Bayes model which are combined using Bayes' rule to finally arrive to a logistic regression model (we will ignore for the moment the details of this figure but come back to it throughout this section). The random graph generator gives rise to networks which, based on topology can all be plausible hypotheses for the PPI network that we want to reconstruct. Incorporating the actual data will reduce this set of plausible hypotheses to just a few, out of which we can pick the one which has the highest likelihood. We

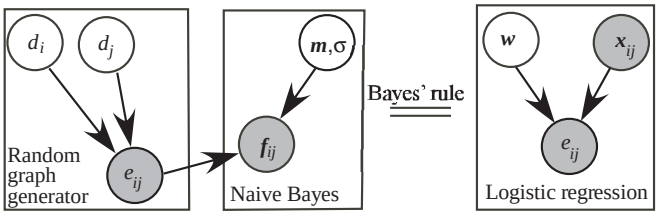

Fig. 1. Left box: random graph generator model. Center box: naive Bayes model. Right box: the result of applying Bayes' rule, the model which combines topology and feature information.

implement this in a Bayesian framework by treating our random graph model as a prior and define a probability model for the features given the absence/presence of an edge and combine these two using Bayes' rule, to finally arrive at a model incorporating both topological and feature information. The way in which each of these models is constructed and then combined is described in the rest of this section.

We deliberately will choose relatively simple and well-known models: naive Bayes for incorporating feature-based classification and a general random graph generator as a prior for PPI networks. As we will see, they combine nicely and such that (approximate) Bayesian inference becomes feasible. Other model choices, e.g., that better incorporate correlations between variables or more subtle properties of PPI network, may lead to even better performance, but then likely at the expense of computational efficiency (see also the discussion in Section IV).

A. Topology of PPI Networks

We will focus on one essential topological characteristics of PPI networks: the *node degree distribution*. The degree of a node represents the number of connections the node has with the other nodes in the network. The probability distribution of these degrees over the whole network, $P(k)$, is defined as the fraction of nodes in the network with degree k,

$$P(k) = \frac{N_k}{N} ,$$

where N is the total number of nodes in the network and N_k is the number of nodes with degree k. The majority of real-world networks have a node degree distribution that is highly right-skewed, which means that most of the nodes have low degrees, while a small number of nodes, known as "hubs", have high degrees. The degree of hubs is typically several order of magnitudes larger than the average degree of a node in the network. This is a distinctive characteristic of PPI networks as well [15]. It has been shown that the connectivity of a protein is related to its function [48], high connectivity is often associated with proteins involved in information storage and processing (transcription in particular) and cellular processes and signaling. Among the non-hubs, there are many proteins that participate in metabolism, while proteins with poorly characterized functions frequently have few or no interactions.

The *clustering coefficient* is a measure of degree to which nodes in a graph tend to cluster together. In most real-world networks, and in particular social networks, nodes tend to create highly inter-connected regions, called clusters [49]. The clustering coefficient has been used in combination with the degree distribution to determine the topology of the interactome [17], showing that the complete interactome is either highly skewed such as in scale-free networks or is at least highly clustered. [16] showed the existence in PPI networks of highly inter-connected regions which are correlated with biological function and large multi-protein complexes.

The distance $d(u, v)$ between two nodes is the path with minimum length, where length is the number of edges. The *average distance* of G is the average over all distances $d(u, v)$ for u and v in G. It represents closeness and measures how quickly information can be transferred in a network The *graph diameter* of G is the maximum distance $d(u, v)$ where u and v are in the same connected components. For scale-free networks with degree exponent $2 < \gamma < 3$ which have been observed for most real-world networks, the average distance is very small, i.e., of the order $\log\log N$ [50], which is known as the small world phenomenon [49] and the diameter is of the order $\log N$ [50]. The PPI networks are shown to adhere also to the small world phenomenon, i.e., most pairs of proteins are connected to each other by a short chain of links involving several intermediate proteins.

B. Random Graph Generator

The first step of our approach is to define a model for generating networks with the node degree distribution similar to the one of PPI networks (the left-hand side box of Figure 1). The random graph generator that we define here is inspired by the general random graph [51]. The general random graph method assigns each node with its expected degree and edges are inserted probabilistically according to a probability proportional to the product of the degrees of the two endpoints, i.e., the probability of an edge between two nodes i and j is proportional to the product of the expected degrees of the nodes i and j. We introduce a latent variable, $d^{(i)}$, related to the degree of node i, i.e., $d^{(i)}$ is roughly proportional to the degree of node i. Let $e^{(ij)}$ be a random variable related to the presence or absence of a link between nodes i and j, $e^{(ij)}$ has two possible values: $e^{(ij)} = 1$ if a link is present between nodes i and j, and $e^{(ij)} = -1$ if a link is not present between nodes i and j. Our model generates links in the network as follows,

$$\begin{aligned} P(e^{(ij)}|d^{(i)}, d^{(j)}) &\propto (\sqrt{d^{(i)}d^{(j)}})^{e^{(ij)}} \\ &= \exp\left[e^{(ij)}\frac{1}{2}(\log d^{(i)} + \log d^{(j)})\right] , \end{aligned} \tag{1}$$

$$\begin{aligned} &P(e^{(ij)} = 1|d^{(i)}, d^{(j)}) \propto \sqrt{d^{(i)}d^{(j)}} , \\ &P(e^{(ij)} = -1|d^{(i)}, d^{(j)}) \propto \frac{1}{\sqrt{d^{(i)}d^{(j)}}} , \\ &P(e^{(ij)} = 1|d^{(i)}, d^{(j)}) \\ &\quad = \frac{P(e^{(ij)} = 1|d^{(i)}, d^{(j)})}{P(e^{(ij)} = 1|d^{(i)}, d^{(j)}) + P(e^{(ij)} = -1|d^{(i)}, d^{(j)})} \\ &\quad = \frac{\sqrt{d^{(i)}d^{(j)}}}{\sqrt{d^{(i)}d^{(j)}} + \frac{1}{\sqrt{d^{(i)}d^{(j)}}}} = \frac{d^{(i)}d^{(j)}}{1 + d^{(i)}d^{(j)}} , \\ &P(e^{(ij)} = -1|d^{(i)}, d^{(j)}) \\ &\quad = \frac{P(e^{(ij)} = -1|d^{(i)}, d^{(j)})}{P(e^{(ij)} = 1|d^{(i)}, d^{(j)}) + P(e^{(ij)} = -1|d^{(i)}, d^{(j)})} \\ &\quad = \frac{\frac{1}{\sqrt{d^{(i)}d^{(j)}}}}{\sqrt{d^{(i)}d^{(j)}} + \frac{1}{\sqrt{d^{(i)}d^{(j)}}}} = \frac{1}{1 + d^{(i)}d^{(j)}} , \end{aligned}$$

In Figure 1, the random variables $d^{(i)}$ and $d^{(j)}$ are represented by white color circles because they are unobserved while the variable $e^{(ij)}$ is represented by a gray color circle since it is observed. The random graph generator can create networks with a desired topology, more specifically with a desired node degree distribution, by assuming a well-chosen distribution for the latent variable associated with the node degree, i.e., $d^{(i)}$. The first choice for the distribution over $d^{(i)}$ would be a power-law which is in general used for modeling the degree distribution of PPI networks [15]. Networks with a power law distribution for node degrees are referred to as scale-free networks [52] and include among others the world wide web, metabolic networks or citation networks. An exponential distribution for $d^{(i)}$ and $d^{(j)}$ gives rise to a scale-free network [51]. A log-normal distribution is another option for modeling the node degree distribution of scale-free networks [53]. Power-law and log-normal distributions are intrinsically connected in the sense that similar generative models can lead to either power law or log-normal distributions [54] and are both suited in our case [55]. For computational reasons which will become clear later, we consider a log-normal distribution for $d^{(i)}$, this means that $\log d^{(i)}$ is normally distributed,

$$P(\log d^{(i)}) = \mathcal{N}(\log d^{(i)}; m_0, \sigma_0^2) , \tag{2}$$

where m_0 is a scaling parameter, and the parameter σ_0 controls the shape of the distribution. These parameters can be set such that the networks randomly generated with the model from Equation (1) have the desired topology. We have defined $d^{(i)}$ to be roughly proportional to the degree of node i, thus a log-normal distribution for $d^{(i)}$ results in a distribution for the degree of node i which is approximately log-normal, which is similar to what is observed in practice. In summary, the random graph generator for a given topology, that we define here, performs the following steps. 1) Choose m_0 and σ_0 the parameters of the log-normal distribution for $d^{(i)}$. 2) Draw from this distribution a random sample $(d^1, \ldots, d^N)$ of size N the number of nodes in the network. 3) Based on this sample construct the network by inserting edges with probability given in

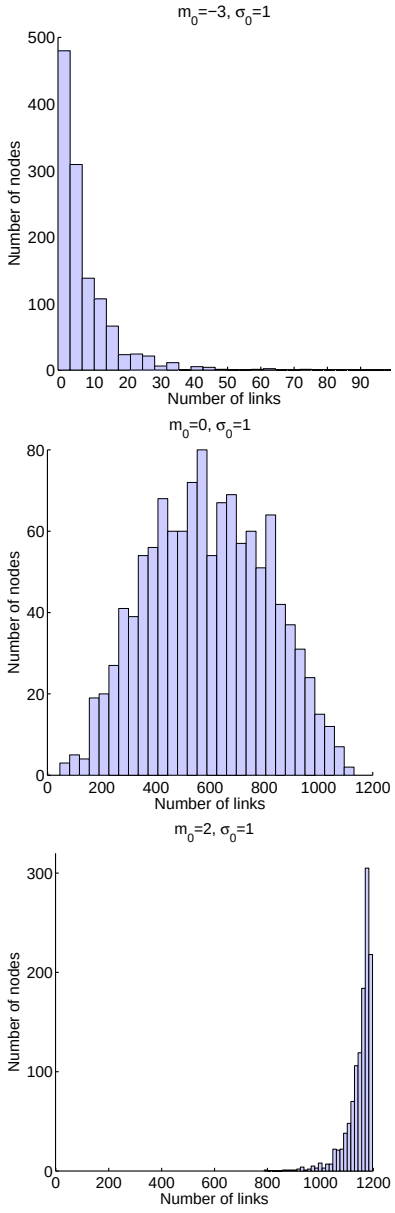

Fig. 2. Node degree distributions of three networks randomly generated from three different log-normal distributions for $d^{(i)}$; top: $m_0 = -3$, $\sigma_0 = 1$, center: $m_0 = 0$, $\sigma_0 = 1$ and bottom: $m_0 = 3$, $\sigma_0 = 1$.

Equation (1). As examples, Figure 2 shows the node degree distributions of three networks randomly generated with the method described above and starting from three settings of the parameters of the log-normal distribution for $d^{(i)}$; top: $m_0 = -3$, $\sigma_0 = 1$, center: $m_0 = 0$, $\sigma_0 = 1$ and bottom: $m_0 = 3$, $\sigma_0 = 1$. The first network is very sparse, with a connectivity of 5% (the connectivity is represented as the percentage of actual links from the total number of possible links). The second and third networks have connectivities of 50% and 95%, respectively, and are quite far from the topology of PPI networks. For $m_0 < -3$ the networks become more sparse and for $m_0 > 3$ the networks become more connected. The parameter σ controls the width of the distribution.

The histograms in Figure 3 compare the node degree distribution in two types of networks: 1) PPI networks observed in two species: yeast and human (the histograms on the left-hand side) and 2) networks randomly generated from the random graph generator defined above (the histograms on the right-hand side). The description of the PPI networks for yeast and human is given in Section III-A. The parameters of the log-normal distribution for $d^{(i)}$ were set such that the histograms of the random networks are similar to the histograms of the PPI networks, for the random network from the first row: $m_0 = -3.7$ and $\sigma_0 = 1$ and for the random network from the second row: $m_0 = -4.2$ and $\sigma_0 = 1.11$. These histograms show that the random graph generator that we defined indeed yields networks with node degree distribution very similar to those observed in practice. Based on Figures 2 and 3, an appropriate choice for the parameters of the log-normal distribution that we will use for the experimental evaluation is $m_0 = -3$ and $\sigma_0 = 1$.

C. Bayesian Framework for Combining Topology and Feature Information

In order to combine topology and feature information, we treat the random graph model as a prior and define a probability model for the features of a protein pair given the absence/presence of an interaction between the proteins. We use a naive Bayes model to express the likelihood of the features for a protein pair given the absence/presence of an interaction. Let D represent the dimension of the feature vectors. The likelihood is thus computed as a product of 1-dimensional Gaussian distributions, each Gaussian distribution expressing the probability of a feature component $f_k^{(ij)}, k = 1, \ldots, D$ given the edge variable $e^{(ij)}$ and the parameters: mean m_k and variance σ,

$$P(\boldsymbol{f}^{(ij)}|e^{(ij)}, \boldsymbol{m}, \sigma) = \prod_{k=1}^{D} \mathcal{N}(f_k^{(ij)}; m_k e^{(ij)}, \sigma) \propto \prod_{k=1}^{D} \exp\left(-\frac{(f_k^{(ij)} - e^{(ij)} m_k)^2}{2\sigma^2}\right). \quad (3)$$

We refer to the center box of Figure 1 for a graphical representation of this model. The naive Bayes model defined above treats the features as independent, which might not be the case in practice. Despite this simplifying assumption, the naive Bayes model is known to be a competitive classification method with similar performance as the closely related logistic regression algorithm.

The posterior distribution for $e^{(ij)}$ which combines topology and feature information is computed using Bayes' rule as the product between the prior defined in Equation (1)

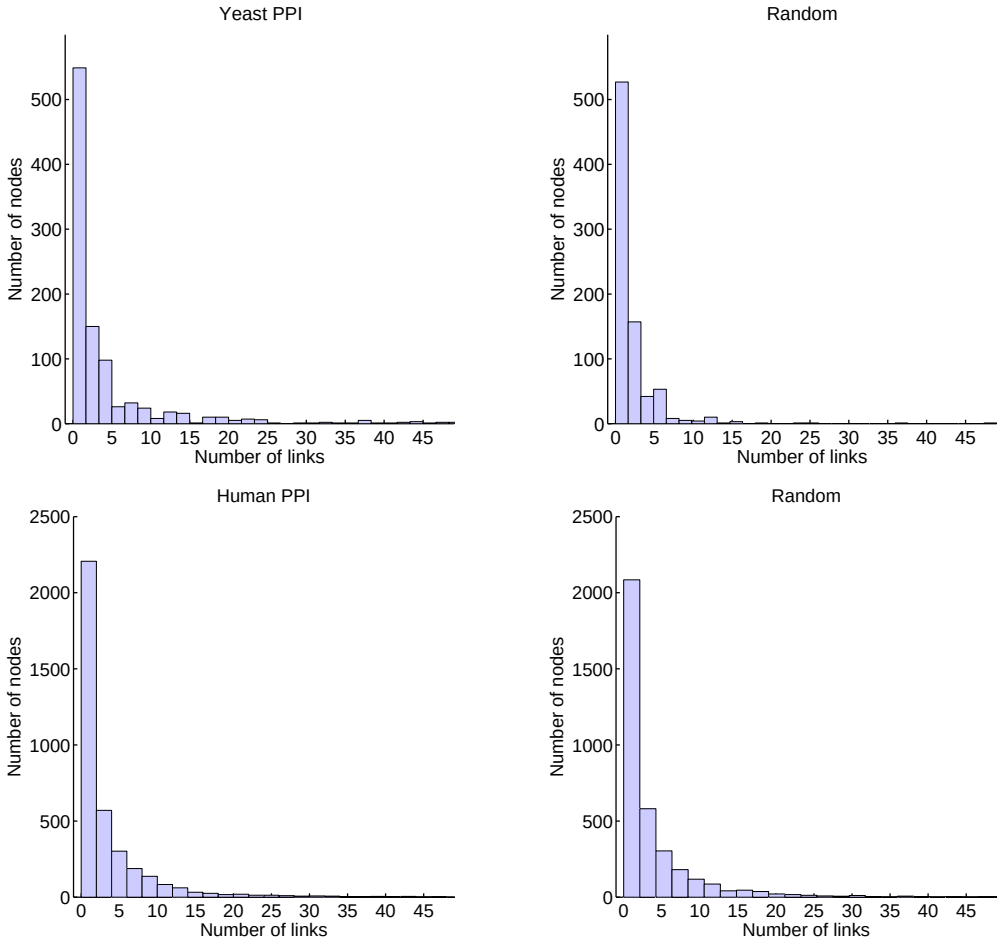

Fig. 3. Left: histograms of node degrees of yeast PPI network (top row) and human PPI network (bottom row); the description of these networks is given in Section III-A. Right: histograms of node degrees of two random networks generated with the model from Section II-B and with the parameters of the log-normal distribution for the latent variables $d^{(i)}$: $m_0 = -3.7$ and $\sigma_0 = 1$ (top row) and $m_0 = -4.2$ and $\sigma_0 = 1.11$ (bottom row).

and the likelihood terms defined in Equation (3), i.e.,

$$
\begin{aligned}
&P(e^{(ij)}|\boldsymbol{f}^{(ij)}, d^{(i)}, d^{(j)}) \\
&\propto P(e^{(ij)}|d^{(i)}, d^{(j)}) P(\boldsymbol{f}^{(ij)}|e^{(ij)}, d^{(i)}, d^{(j)}) \\
&\propto \exp\left(e^{(ij)} \frac{1}{2} (\log d^{(i)} + \log d^{(j)}) - \frac{\sum_k (f_k^{(ij)} - e^{(ij)} m_k)^2}{2\sigma^2} \right) \qquad (4) \\
&\propto \exp\left(e^{(ij)} \frac{1}{2} (\log d^{(i)} + \log d^{(j)}) + \frac{e^{(ij)} \sum_k f_k^{(ij)} m_k}{\sigma^2} \right) \qquad (5) \\
&= \exp\left(e^{(ij)} (\sum_{k=1}^{D} \frac{f_k^{(ij)} m_k}{\sigma^2} + \frac{1}{2} \log d^{(i)} + \frac{1}{2} \log d^{(j)}) \right) \qquad (6)
\end{aligned}
$$

where from (4) to (5) we discarded the square terms which do not depend on $e^{(ij)}$. In the above, we can ignore any term that does not depend on $e^{(ij)}$, since it will only affect the normalization. This includes the term $(e^{(ij)})^2 m_k^2/\sigma^2$, since $e^{(ij)} \in \{-1, 1\}$. The normalization term does play a role and, when incorporated, leads to Equation (8) below. The unknown quantities of our model are $\frac{m_k}{\sigma^2}$, $k = \{1, \ldots, D\}$ and $\log d^{(i)}$, $i = \{1, \ldots, N\}$, and these will be estimated based on the available training data in a learning procedure that we describe below.

The first step is to adjoin the unknown quantities in a single random vector,

$$
\boldsymbol{w} = [\frac{m_1}{\sigma^2}, \ldots, \frac{m_D}{\sigma^2}, \frac{1}{2} \log d^{(1)}, \ldots, \frac{1}{2} \log d^{(N)}] \,, \qquad (7)
$$

and the same for the information available, which is protein features and topological information

$$
\boldsymbol{x}^{(ij)} = [\boldsymbol{f}^{(ij)}, \boldsymbol{t}^{(ij)}] \,,
$$

where $\boldsymbol{t}^{(ij)}$ is the position vector having 1 on positions i and j and 0 everywhere else. Using this notation, the normalized probability from Equation (6) of an interaction between the proteins i and j can be rewritten as

$$
P(e^{(ij)}|\boldsymbol{x}^{(ij)}, \boldsymbol{w}) = \frac{1}{1 + \exp(-2e^{(ij)} \boldsymbol{w}^T \boldsymbol{x}^{(ij)})} \,. \qquad (8)
$$

Note that in the sum

$$
\boldsymbol{w}^T \boldsymbol{x}^{(ij)} = \sum_{k=1}^{D} w_k f_k^{(ij)} + \sum_{k=1}^{N} w_{D+k} t_k^{(ij)} \,, \qquad (9)
$$

the first term on the right-hand side originates from the protein features information and the second term from the topological information.

The unknown parameter $\boldsymbol{w}$ is learned in a Bayesian framework which consists in setting a prior distribution for it, and updating this prior based on the available observations. The update is performed using Bayes' rule, i.e.,

$$P(\boldsymbol{w}|\text{observations}) \propto \prod_{l=1}^{n_{\text{obs}}} P(e_l^{(ij)}|\boldsymbol{x}_l^{(ij)}, \boldsymbol{w}) P(\boldsymbol{w}) \,. \quad (10)$$

where n_{obs} is the size of the training data, i.e., the number of known interacting/non-interacting protein pairs. $P(e_l^{(ij)}|\boldsymbol{x}_l^{(ij)}, \boldsymbol{w})$ is given in Equation (8). $P(\boldsymbol{w})$ is the prior and we choose it to be a Gaussian distribution

$$P(\boldsymbol{w}) = \mathcal{N}(\boldsymbol{w}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) \,.$$

The hyperparameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ of the prior are chosen such that the topological information is incorporated into the model. This is implemented by making the correspondence with the prior for the latent variables $d^{(i)}$. Recall from Equation (7) that $w_{i+D} = \frac{1}{2}\log d^{(i)}$, $i = 1, \ldots, N$ and from Equation (2) that $\log d^{(i)}$ is normally distributed, consequently w_{i+D} will also be normally distributed, i.e.,

$$w_{i+D} \sim \mathcal{N}\left(\frac{m_0}{2}, \frac{\sigma_0^2}{4}\right) \,, \; i = 1, \ldots, N \,.$$

A good choice for the hyperparameters m_0 and σ_0 was discussed in relation to Figures 2 and 3. Thus, we set $\boldsymbol{\mu}_{D+1:N} = \frac{m_0}{2} = -1.5$ which corresponds to a network with a node degree distribution of the form displayed in the histogram from top in Figure 2. We will see in the experimental evaluation, Section III-C2, that the choice of the prior parameters that correspond to the topology information has a big influence on the performance of the models. The hyperparameters μ_i, $i = 1, \ldots, D$ that correspond to the feature information were set to 0, and the covariance matrix $\boldsymbol{\Sigma}$ was chosen to be the identity matrix. This choice for the parameters of the prior corresponding to protein features makes sense when the features are normalized like we do in the experimental evaluation.

In this Bayesian framework, predictions can be done for an unknown interaction between a pair of proteins i', j' characterized by the feature vector $\boldsymbol{x}^{(i'j')}$. These predictions can be done either averaging the posterior over $\boldsymbol{w}$ in Equation (8) or by using a point estimate of this posterior, let $\boldsymbol{w}^*$ be the mean of $P(\boldsymbol{w}|\text{observations})$, and computing $P(e^{(i'j')}|\boldsymbol{x}^{(i'j')}, \boldsymbol{w}^*)$ using Equation (8).

Bayesian inference is known to be computationally expensive. However, in the setting presented in this paper, the computations can be made more efficient by exploiting the sparsity of the input data. The vectors $\boldsymbol{x}^{(ij)}$ are sparse because their components $\boldsymbol{t}^{(ij)}$ of dimension N (the number of proteins) contain only two non-zero elements, on positions i and j.

We refer back to the graphical sketch of our model in Figure 1 at the beginning of this section. The box on the left-hand side, corresponds to the random graph generator model. The observation $e^{(ij)}$, which expresses the presence or absence of an edge between nodes i and j, depends on the latent variables $d^{(i)}$ and $d^{(j)}$ which are roughly proportional to the degrees of nodes i and j. The random graph generator model incorporates feature information through the naive Bayes model with unknown parameters $\boldsymbol{m}$ and σ, represented in the center box. The combination of the two models is obtained using Bayes' rule. The result is shown in the right-hand side box. The unknown quantities $d^{(i)}$, $d^{(j)}$, and $\boldsymbol{m}$, σ are combined in the node $\boldsymbol{w}$ which is unobserved, and $\boldsymbol{f}^{(ij)}$ together with $\boldsymbol{t}^{(ij)}$ which is implicitly expressed by indices i and j, form the observed quantity $\boldsymbol{x}^{(ij)}$.

Summarizing, in order to incorporate topological information for PPI prediction we propose a relatively simple method: logistic regression on an extended feature space. The extended feature space is obtained by adding to the feature vector for a pair of proteins i and j, $\boldsymbol{f}^{(ij)}$, a vector of dimension N with 1 on positions i and j and 0 everywhere else. The regression weights are treated as latent variables and those weights corresponding to the additional topology features are in a one-to-one correspondence with the latent variables $d^{(i)}$ of the random graph generator. The scale-free like architectures of the random graph generator follow from a log-normal prior distribution on these $d^{(i)}$s.

In the experimental evaluation from Section III we will compare four models. These models are based on the same Bayesian framework from Equation (10) with a Gaussian prior and likelihood terms of the form given in Equation (8). The models differ in the type of information they use and how they combine this information. Specifically, the models vary in the way of computing the dot product from Equation (9) and on the parameters of the Gaussian prior.

1) Model 1 (Features+Topology): is the model we introduced above. It makes use of the dot product from Equation (9) and uses a Gaussian prior of dimension $D+N$ with mean $\boldsymbol{\mu}_{1:D} = 0$, $\boldsymbol{\mu}_{D+1:N} = -\frac{3}{2} = -1.5$ and covariance matrix equal to the identity matrix.
2) Model 2 (Features only): uses only information about proteins, thus the dot product is computed as

$$\boldsymbol{w}^T \boldsymbol{x}^{(ij)} = \sum_{k=1}^{D} w_k f_k^{(ij)} + w^{D+1} \,. \quad (11)$$

   The second term on the right-hand side of Equation (11) is a bias term to address the unbalancedness of the data. This bias term also corresponds to the second term on the right-hand side of Equation (9); for an edge $e^{(ij)}$ the contributions in Equation (9) are $w_{D+i} + w_{D+j}$ while in Equation (11) we constraint $w_{D+i} = \frac{1}{2} w_{D+1}$, $\forall i = 1, \ldots, N$. This observation also motivates the choice of the prior for this model: mean $\boldsymbol{\mu}_{1:D} = 0$ and $\mu_{D+1} = -3$ and covariance equal to the identity matrix.
3) Model 3 (Topology only): uses only topology infor-

mation, thus the dot product is computed as

$$\boldsymbol{w}^T \boldsymbol{x}^{(ij)} = \sum_{k=1}^{N} w_k t_k^{(ij)} .$$

The Gaussian prior is of dimension N with mean equal to the vector $\boldsymbol{\mu}_{1:N} = -1.5$ and covariance matrix equal to the identity matrix. The choice for $\boldsymbol{\mu}_{1:N} = -1.5$ corresponds to the log-normal distribution with $m_0 = -3$, thus to a network with a node degree distribution of the form of the top plot from Figure 2.

4) Model 4 (Topology-enriched features): uses the information about proteins and about topology in the following form

$$\begin{aligned} \boldsymbol{w}^T \boldsymbol{x}^{(ij)} &= \sum_{k=1}^{D} w_k f_k^{(ij)} + w_{D+1} \log(\hat{d}^{(i)} + 1) \\ &\quad + w_{D+2} \log(\hat{d}^{(j)} + 1) , \end{aligned}$$

where $\hat{d}^{(i)}$ and $\hat{d}^{(j)}$ are the estimated degrees of nodes i and j computed on the training data. Basically, the features $\boldsymbol{f}^{(ij)}$ for a pair of proteins i and j are being extended by adding two new columns corresponding to the degrees of nodes i and j computed on the training set. For computational reasons we considered the logarithms of node degrees to which we added 1. The idea behind this model is similar to the one used in [56], [44], i.e., the topological features are added to protein features resulting in an enriched set of features. The features are being standardized and the parameters of the Gaussian prior are set to $\boldsymbol{\mu}_{1:D+2} = 0$ and covariance equal to the identity matrix.

III. Results

The four models previously described were empirically evaluated and the results are presented in this section.

A. Data Sets

We used two data sets. Details for each of them are given below.

The yeast data set was used in [36], [12] and it consists of the high confidence physical interactions between proteins highlighted in [57]. The PPI network has 984 nodes (proteins) connected by 2438 links (interactions). We consider all the protein pairs not present in the 2438 interactions as non-interacting. The degree distribution of the nodes is shown in the top-left plot in Figure 3. The yeast PPI graph is very sparse, as a result the data is highly unbalanced, with less than 1% from the total examples belonging to the positive class. We balanced the data set by sampling a number of negative examples equal with the number of positive examples. For each protein pair we considered a vector of features of dimension 5 representing gene expression values under different experimental conditions.

The human data set was created and made available by [37] and consists of protein pairs with an associated label: interact or non-interact. Unlike positive interactions, non-interacting pairs are not experimentally reported. Thus, a common strategy is to consider as non-interacting pairs a randomly drawn fraction from the total set of potential protein pairs excluding the pairs known to interact. The resulting data set has 14,608 interacting pairs and 432,197 non-interacting pairs. The PPI graph consists of $24,380$ nodes connected by $14,608$ edges. As in the case of the yeast data set, the PPI graph of the human data is very sparse. The degree distribution of the nodes is shown in the bottom-left plot on Figure 3. We balanced the data set by sampling a number of negative examples equal with the number of positive examples. Each pair of proteins is characterized by a 27-dimensional feature vector. The features were constructed based on Gene Ontology (GO) cell component (1), GO molecular function (1), GO biological process (1), co-occurrence in tissue (1), gene expression (16), sequence similarity (1), homology based (5) and domain interaction (1), where the numbers in brackets correspond to the number of elements contributed by the feature type to the feature vector. Homology based features were derived from the protein-protein interaction data sets, but more sophisticated approaches based on HMM or Markov random fields [58], [59] could have also been used here. Domain-domain interactions were derived for each candidate protein pair using the method described in [60].

B. Experimental Setup and Evaluation

The experimental setup considered a part of the data for training and the rest for testing. The training data was used to learn the models and the testing data was used to evaluate the performance of these models for predicting PPIs. We randomly sampled 10 training sets containing different percentages $(10\%, 20\%, 30\%, 40\%, 50\%, 60\%, 70\%, 80\%, 90\%)$ from the total observations. The training features were standardized to have mean zero and standard deviation of one. We used area under the receiver operating characteristic curve (AUC) as a measure for evaluating the performance. We can interpret the AUC statistic as the probability that a randomly chosen missing edge (a true positive) is given a higher score by the method than a randomly chosen pair of proteins without an interaction (a true negative). The entire training and testing procedures for all percentages of data were repeated 10 times on different observations in training and testing and average results (mean $\pm$ standard deviation) over the 10 splits of the data into training and testing were reported.

C. Performance

Tables I and II show the performance of the four models discussed in Section II and four other models from the literature for predicting PPIs on the two data sets: yeast (Table I) and human (Table II). Model 1 (Features+Topology) represents the Bayesian framework for combining feature and topology information, Model 2 (Features only) uses

only protein information, Model 3 (Topology only) uses only topology information and Model 4 (Topology-enriched features) uses protein features which are enriched with node degrees. For comparison we used four other approaches: SVM, SVM logistic (which fits logistic models to the SVM outputs), decision tree (the C4.5 decision tree [61]) and random forest classifier [62]. For the implementation of the classifiers used for comparison we used the WEKA toolbox [63]. The protocol described in Section III-B was used for all methods considered and the averaged AUC scores with their standard deviations are reported.

The results show that the combination of the two sources of information, protein features and topology, gives a better performance than using only one type of information. In particular Model 1 (Features+Topology) performs significantly better than Model 2 (Features only) in most of the cases. Model 1 and Model 4 have a similar performance for human data while Model 1 performs better than Model 4 for yeast data. An explanation for this is related to how the protein features were constructed in the two cases; for yeast data the features for a protein pair resulted from summing the feature vectors corresponding to the two proteins, while for human data the protein features are more related to the protein pair than to individual proteins. Model 1 incorporates the topological information through a Bayesian prior, Model 4 just includes the node degrees computed on the training data as features. Both are valid options, but Model 1 empirically outperforms Model 4, arguably because the probabilistic framework incorporates the uncertainty in the degrees instead of considering them fixed and given. The results vary also as a function of the size of the training data. For a small training set the network topology is not well defined, and we can see that in this case the improvement is smaller, but, as we increase the training set, meaning that the knowledge about the network topology, i.e., about which nodes are hubs, increases, the performance obtained by adding the topology information improves. The random forest classifier has been shown to perform very well on predicting PPIs [10], [64]. This is also the case for this comparison, however, we notice that the best performance is obtained by Model 1 (Features+Topology) which is the Bayesian framework for combining topology and feature information that we propose in this work.

1) Topology Learning: The combination between the protein features and topology information from Model 1 has the best performance in comparison with the other models since it is able to learn faster and more accurate the topology of the network, which in this case refers to node degrees. In order to show this, we analyzed how accurate the degrees of the nodes are estimated in the cases of Model 1 (Features+Topology) and Model 2 (Features only). The comparison was performed on the yeast data. Both models were learned on the training data. The predicted node degrees were computed on the test data by summing the predictive probabilities from Equation (8) for edges $e^{(ij)}$ in which one of the indices i or j corresponds to the node of interest. The estimation of the node degrees for the two models is shown in Figure 4 with the x-axis showing the percentage of data used for training, and the y-axis showing the error of the estimates measured using the root mean square of the difference between the predicted and actual node degrees (top plot) and the root mean square of the difference between the logarithms of the predicted and actual node degrees (bottom plot). Where the root mean square of the node degrees themselves measures the absolute error in estimated node degrees, which is quite sensitive to correctly estimating the hub nodes, the root mean square of the logarithms measures the relative error. It can be seen that indeed Model 1 (Features+Topology) gives a much better estimate to the actual node degrees in comparison to Model 2 (Features) and this explains also why the performance of Model 1 is better than that of Model 2 as shown also in Tables I and II.

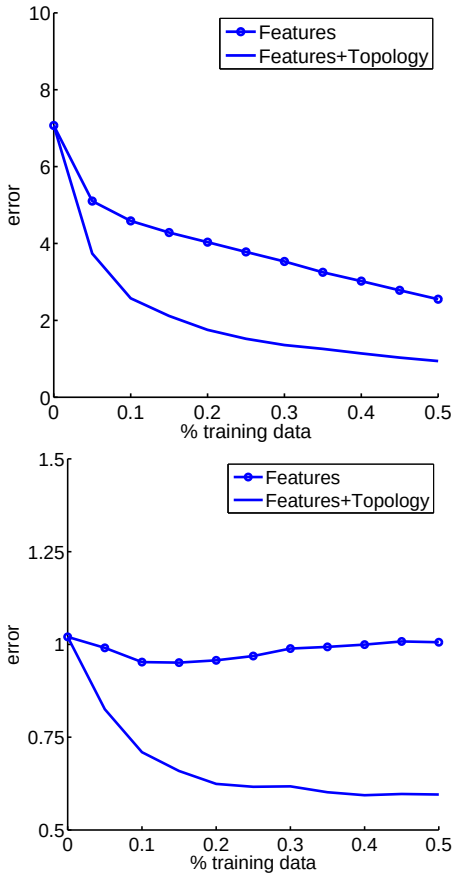

Fig. 4. The root mean square of the difference between the predicted and actual node degrees (top) and between the logarithms of the predicted and actual node degrees (bottom) for the two models: Model 2 (features only) and Model 1 (features+topology) as a function of the percentage of data considered in the training set.

The topology information acts in controlling the node degree distribution and forbidding producing too many nodes with extra large degrees. The topology information enters in our model as a prior which favors networks with degree distributions right-skewed (similar to the top plot from Figure 2 and the plots from Figure 3 and not like the middle and the bottom plot from Figure 2), thus a few hubs and the rest of the nodes with just a few connections.

TABLE I

EXPERIMENTAL RESULTS FOR YEAST DATA SET. AUC IS USED AS A MEASURE FOR EVALUATING THE PERFORMANCE, AVERAGE RESULTS (MEAN ± STANDARD DEVIATION) ARE BEING REPORTED. THE RESULTS ARE SHOWN FOR THE FOUR MODELS DISCUSSED IN THIS PAPER AND ENUMERATED AT THE END OF SECTION II: MODEL 1 (FEATURES+TOPOLOGY), MODEL 2 (FEATURES ONLY), MODEL 3 (TOPOLOGY ONLY) AND MODEL 4 (TOPOLOGY-ENRICHED FEATURES). THE TABLE CONTAINS FOR COMPARISON THE RESULTS OBTAINED FOR FOUR OTHER EXISTING METHODS: SVM, SVM LOGISTIC, DECISION TREE AND RANDOM FOREST. THE COLUMNS REPRESENT DIFFERENT PERCENTAGE OF DATA CONSIDERED FOR TRAINING AND THE REST FOR TESTING.

	10%	20%	30%	40%	50%
Model 1	0.90 ± 0.02	0.94 ± 0.01	0.96 ± 0.00	0.97 ± 0.00	0.97 ± 0.00
Model 2	0.70 ± 0.00	0.71 ± 0.00	0.71 ± 0.00	0.71 ± 0.00	0.71 ± 0.00
Model 3	0.89 ± 0.02	0.92 ± 0.01	0.95 ± 0.01	0.95 ± 0.01	0.96 ± 0.00
Model 4	0.74 ± 0.07	0.80 ± 0.06	0.81 ± 0.07	0.81 ± 0.06	0.81 ± 0.04
SVM	0.64 ± 0.01	0.64 ± 0.01	0.63 ± 0.01	0.63 ± 0.01	0.62 ± 0.01
SVM logistic	0.69 ± 0.00	0.69 ± 0.00	0.70 ± 0.00	0.70 ± 0.00	0.70 ± 0.00
Decision tree	0.70 ± 0.02	0.72 ± 0.01	0.74 ± 0.00	0.74 ± 0.00	0.75 ± 0.01
Random forest	0.76 ± 0.01	0.78 ± 0.00	0.79 ± 0.00	0.80 ± 0.00	0.81 ± 0.01

	60%	70%	80%	90%
Model 1	0.98 ± 0.00	0.98 ± 0.00	0.98 ± 0.00	0.99 ± 0.00
Model 2	0.70 ± 0.00	0.70 ± 0.00	0.70 ± 0.00	0.70 ± 0.01
Model 3	0.97 ± 0.00	0.97 ± 0.00	0.97 ± 0.00	0.98 ± 0.00
Model 4	0.82 ± 0.03	0.82 ± 0.02	0.83 ± 0.02	0.82 ± 0.02
SVM	0.62 ± 0.00	0.62 ± 0.00	0.62 ± 0.00	0.62 ± 0.01
SVM logistic	0.70 ± 0.00	0.70 ± 0.00	0.70 ± 0.00	0.70 ± 0.01
Decision tree	0.75 ± 0.01	0.76 ± 0.01	0.77 ± 0.01	0.76 ± 0.01
Random forest	0.81 ± 0.00	0.81 ± 0.00	0.81 ± 0.01	0.82 ± 0.01

TABLE II

EXPERIMENTAL RESULTS FOR HUMAN DATA SET. AUC IS USED AS A MEASURE FOR EVALUATING THE PERFORMANCE, AVERAGE RESULTS (MEAN ± STANDARD DEVIATION) ARE BEING REPORTED. THE RESULTS ARE SHOWN FOR THE FOUR MODELS DISCUSSED IN THIS PAPER AND ENUMERATED AT THE END OF SECTION II: MODEL 1 (FEATURES+TOPOLOGY), MODEL 2 (FEATURES ONLY), MODEL 3 (TOPOLOGY ONLY) AND MODEL 4 (TOPOLOGY-ENRICHED FEATURES). THE TABLE CONTAINS FOR COMPARISON THE RESULTS OBTAINED FOR FOUR OTHER EXISTING METHODS: SVM, SVM LOGISTIC, DECISION TREE AND RANDOM FOREST. THE COLUMNS REPRESENT DIFFERENT PERCENTAGE OF DATA CONSIDERED FOR TRAINING AND THE REST FOR TESTING.

	10%	20%	30%	40%	50%
Model 1	0.92 ± 0.00	0.95 ± 0.00	0.96 ± 0.00	0.96 ± 0.00	0.97 ± 0.00
Model 2	0.86 ± 0.00	0.86 ± 0.00	0.86 ± 0.00	0.86 ± 0.00	0.86 ± 0.00
Model 3	0.86 ± 0.00	0.91 ± 0.00	0.93 ± 0.00	0.94 ± 0.00	0.95 ± 0.00
Model 4	0.91 ± 0.00	0.94 ± 0.00	0.95 ± 0.00	0.96 ± 0.00	0.96 ± 0.00
SVM	0.80 ± 0.00	0.80 ± 0.00	0.81 ± 0.00	0.81 ± 0.00	0.81 ± 0.00
SVM logistic	0.85 ± 0.00	0.86 ± 0.00	0.86 ± 0.00	0.86 ± 0.00	0.86 ± 0.00
Decision tree	0.84 ± 0.01	0.85 ± 0.01	0.86 ± 0.00	0.87 ± 0.00	0.87 ± 0.00
Random forest	0.92 ± 0.00	0.92 ± 0.00	0.92 ± 0.00	0.92 ± 0.00	0.93 ± 0.00

	60%	70%	80%	90%
Model 1	0.97 ± 0.00	0.97 ± 0.00	0.97 ± 0.00	0.97 ± 0.00
Model 2	0.86 ± 0.00	0.86 ± 0.00	0.86 ± 0.00	0.86 ± 0.00
Model 3	0.95 ± 0.00	0.96 ± 0.00	0.96 ± 0.00	0.96 ± 0.00
Model 4	0.97 ± 0.00	0.97 ± 0.00	0.97 ± 0.00	0.97 ± 0.00
SVM	0.81 ± 0.00	0.81 ± 0.00	0.81 ± 0.00	0.81 ± 0.00
SVM logistic	0.86 ± 0.00	0.86 ± 0.00	0.86 ± 0.00	0.86 ± 0.00
Decision Tree	0.87 ± 0.00	0.88 ± 0.00	0.88 ± 0.00	0.88 ± 0.00
Random forest	0.93 ± 0.00	0.93 ± 0.00	0.92 ± 0.00	0.93 ± 0.00

2) Influence of the Prior: Table III shows the performance obtained for predicting PPIs using Model 1 for a size of the training data of 1% from the total data set and with three parameter settings for the prior. These correspond to the three parameter settings for the $\log$-normal distribution based on which the histograms from Figure 2 were generated. The performance is best for $m_0 = -3$, $\sigma_0 = 1$ which corresponds to the top histogram from Figure 2 and which is also a valid assumption for the topology of PPI networks. The differences between the parameter settings for m_0, i.e., $(-3, 0, 3)$ are huge and correspond to completely different degree distributions. We argue that the results are insensitive to small, reasonable changes in the hyperparameters.

TABLE III
AUC SCORES (MEAN $\pm$ STANDARD DEVIATION) FOR PREDICTING PPIS USING MODEL 1 FOR THREE PARAMETER SETTINGS FOR THE $\log$-NORMAL DISTRIBUTION (1ST COLUMN). THE RESULTS ARE SHOWN FOR THE YEAST DATA AND HUMAN DATA. THE SIZE OF THE TRAINING DATA IS 1% FROM THE TOTAL DATA. $\sigma_0 = 1$ FOR ALL THREE SETTINGS OF THE m_0 PARAMETER.

Prior parameter settings	Yeast data	Human data
$m_0 = -3$	0.639 ± 0.014	0.863 ± 0.006
$m_0 = 0$	0.595 ± 0.015	0.808 ± 0.014
$m_0 = 3$	0.566 ± 0.015	0.742 ± 0.029

IV. DISCUSSIONS

In addition to the node degree distribution, networks in general, and PPI networks in particular, can be characterized by other global topological properties, including the clustering coefficient, the network diameter, and the average shortest path. [16] showed the existence in PPI networks of highly inter-connected regions which are correlated with biological functions and large multi-protein complexes. PPI networks are shown to adhere also to the small world phenomenon, i.e., most pairs of proteins are connected to each other by a short chain of links involving several intermediate proteins. In addition to these global topological properties, PPI networks are also characterized by the so-called network motifs. A network motif is a small subgraph which appears in the network significantly more frequently than in a randomized network. Different types of real-world networks have been shown to have different motifs [65]. We believe that frameworks similar to the one introduced here for incorporating information about the node degree distribution, can be derived for including other types of topological information. The so-called node signature [32], which represents the topology in the neighborhood of a node, might be useful in this direction. In the same direction, other random graph generators have to be investigated, like for example, exponential random graph models [66]. The degree distribution of PPI networks is in general described as scale-free, although this claim has been recently questioned [18], [67]. However, most parties agree that the degree distribution is very broad, hence the existence of hubs, and the $\log$-normal distribution that we consider here can be fitted well to this situation. Another property related to the nodes' degree which has been observed for PPI networks is the anti-correlation between network degrees of interacting proteins [16]. This means that hub-proteins avoid connecting to each other and instead tend to interact with proteins of low connectivity/degree. The random graph generator that we defined in this model does not satisfy this property. Another option for further improvement could be used to incorporate homology-based features along the lines of [58].

When trying to add topological information, the computational complexity becomes a problem. In the framework presented here, we managed to find some simplifying assumptions which reduce the computational complexity and at the same time yield a good performance. Due to the sparsity of the feature vector which incorporates the sparsity, and using an iterative scheme, the computational complexity of the method could be simplified.

We have set the hyperparameters of the prior to roughly match the known characteristics of PPI networks. This already yields excellent results. A more involved approach would estimate these parameters in an empirical Bayesian approach, which is left for future work. The logistic regression classifier is a natural choice in our framework since latent variables in the graph generator can be translated to weights in the logistic regressor. Similar ideas may also work in connection to other classifiers and may be used for the reconstruction of other biological networks, such as, metabolic, gene regulatory or signaling networks.

In this paper, we introduced a framework for predicting PPI by considering the network structure information. This is a Bayesian framework consisting of a prior distribution over the network topology, likelihood terms and using Bayes' rule to compute the posterior distribution.

REFERENCES

[1] A. Barabási, N. Gulbahce, and J. Loscalzo, "Network medicine: a network-based approach to human disease." *Nature reviews. Genetics*, vol. 12, no. 1, pp. 56–68, 2011.
[2] N. Friedman, "Inferring Cellular Networks Using Probabilistic Graphical Models," *Science*, vol. 303, no. 5659, pp. 799–805, 2004.
[3] R. Jansen, H. Yu, D. Greenbaum, Y. Kluger, N. Krogan, S. Chung, A. Emili, M. Snyder, J. Greenblatt, and M. Gerstein, "A Bayesian networks approach for predicting protein-protein interactions from genomic data," *Science*, vol. 302, no. 5644, pp. 449–453, 2003.
[4] A. Ben-Hur and W. S. Noble, "Kernel methods for predicting protein–protein interactions," *Bioinformatics*, vol. 21, no. 1, pp. 38–46, 2005.
[5] Y. Yamanishi, J.-P. Vert, and M. Kanehisa, "Protein network inference from multiple genomic data: a supervised approach," *Bioinformatics*, vol. 20, no. 1, pp. 363–370, 2004.
[6] N. Lin, B. Wu, R. Jansen, M. Gerstein, and H. Zhao, "Information assessment on predicting protein-protein interactions." *BMC Bioinformatics*, vol. 5, p. 154, 2004.
[7] E. Sprinzak, Y. Altuvia, , and H. Margalit, "Characterization and prediction of proteinprotein interactions within and between complexes," *PNAS*, vol. 103, no. 40, p. 1471814723, 2006.
[8] L. V. Zhang, S. L. Wong, O. D. King, and F. P. Roth, "Predicting co-complexed protein pairs using genomic and proteomic data integration." *BMC Bioinformatics*, vol. 5, p. 38, 2004.

[9] Y. Qi, J. Klein-Seetharaman, and Z. Bar-Joseph, "Random forest similarity for protein-protein interaction prediction from multiple sources." in *Pacific Symposium on Biocomputing*, R. B. Altman, T. A. Jung, T. E. Klein, A. K. Dunker, and L. Hunter, Eds. World Scientific, 2005.

[10] X.-W. Chen and M. Liu, "Prediction of protein-protein interactions using random decision forest framework." *Bioinformatics*, vol. 21, no. 24, pp. 4394–4400, 2005.

[11] P. Geurts, L. Wehenkel, and d'Alché Buc F., "Kernelizing the output of tree-based methods," in *Proceedings of the 23th international conference on Machine learning*, 2006, pp. 345–352.

[12] P. Geurts, N. Touleimat, M. Dutreix, and F. d'Alché-Buc, "Inferring biological networks with output kernel trees," *BMC Bioinformatics (PMSB06 special issue)*, vol. 8, no. Suppl 2, p. S4, 2007.

[13] C. Brouard, F. d'Alché Buc, and M. Szafranski, "Semi-supervised penalized output kernel regression for link prediction," in *ICML*, 2011, pp. 593–600.

[14] C. Xing and D. B. Dunson, "Bayesian inference for genomic data integration reduces misclassification rate in predicting protein-protein interactions," *PLoS Comput Biol*, vol. 7, no. 7, p. e1002110, 2011.

[15] H. Jeong, S. Mason, A.-L. Barabási, and Z. Oltvai, "Lethality and centrality in protein networks," *Nature*, vol. 411, no. 6833, pp. 41–42, 2001.

[16] S. Maslov and K. Sneppen, "Specificity and stability in topology of protein networks," *Science*, vol. 296, pp. 910–913, 2002.

[17] C. C. Friedel and R. Zimmer, "Inferring topology from clustering coefficients in protein-protein interaction networks." *BMC Bioinformatics*, vol. 7, p. 519, 2006.

[18] N. Przulj, D. G. Corneil, and I. Jurisica, "Modeling interactome: scale-free or geometric?." *Bioinformatics*, vol. 20, no. 18, pp. 3508–3515, 2004.

[19] A. Sarajli, V. Janji, N. Stojkovi, D. Radak, and N. Prulj, "Network topology reveals key cardiovascular disease genes," *PLoS ONE*, vol. 8, no. 8, p. e71537, 08 2013.

[20] Z.-C. Li, Y.-H. Lai, L.-L. Chen, Y. Xie, Z. Dai, and X.-Y. Zou, "Identifying functions of protein complexes based on topology similarity with random forest," *Mol Biosyst*, no. 10, pp. 514–525, 2014.

[21] A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins, and J. Wiener, "Graph structure in the web," *Comput. Netw.*, vol. 33, no. 1-6, pp. 309–320, 2000.

[22] S. Redner, "How popular is your paper? an empirical study of the citation distribution," *European Physical Journal B*, vol. 4, pp. 131–134, 1998.

[23] C. Lippert, O. Stegle, Z. Ghahramani, and K. Borgwardt, "A kernel method for unsupervised structured network inference," in *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, 2009, pp. 368–375.

[24] P. Symeonidis, N. Iakovidou, N. Mantas, and Y. Manolopoulos, "From biological to social networks: Link prediction based on multi-way spectral clustering," *Data Knowl. Eng.*, vol. 87, pp. 226–242, 2013.

[25] P. Sheridan, T. Kamimura, and H. Shimodaira, "A scale-free structure prior for graphical models with applications in functional genomics," *PLoS ONE*, vol. 5, no. 11, pp. e13 580+, 2010.

[26] C. Wiuf, M. Brameier, O. Hagberg, and M. P. H. Stumpf, "A likelihood approach to analysis of network data," *Proceedings of the National Academy of Sciences*, vol. 103, no. 20, pp. 7566–7570, 2006.

[27] M. Fiori, P. Musé, and G. Sapiro, "Topology constraints in graphical models," in *Advances in Neural Information Processing Systems*, 2012, pp. 800–808.

[28] Q. Liu and A. Ihler, "Learning scale free networks by reweighted l1 regularization," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, AISTATS*, 2011, pp. 40–48.

[29] R. Tandon and P. Ravikumar, "Learning graphs with a few hubs," in *Proceedings of the 31th International Conference on Machine Learning, ICML*, 2014, pp. 602–610.

[30] A. Birlutiu and T. Heskes, "Using topology information for protein-protein interaction prediction," in *Pattern Recognition in Bioinformatics*, ser. Lecture Notes in Computer Science, M. Comin, L. Kll, E. Marchiori, A. Ngom, and J. Rajapakse, Eds. Springer International Publishing, 2014, vol. 8626, pp. 10–22.

[31] M. A. Yildirim, K.-I. Goh, M. E. Cusick, A.-L. Barabasi, and M. Vidal, "Drug-target network," *Nat Biotech*, vol. 25, no. 10, pp. 1119–1126, Oct. 2007.

[32] T. Milenkovic and N. Przulj, "Uncovering biological network function via graphlet degree signatures," *Cancer Informatics*, vol. 6, pp. 257–273, 2008.

[33] O. Kuchaiev, M. Rasajski, D. J. Higham, and N. Przulj, "Geometric de-noising of protein-protein interaction networks," *PLOS Computational Biology*, vol. 5, no. 8, 2009.

[34] C. Lei and J. Ruan, "A novel link prediction algorithm for reconstructing protein-protein interaction networks by topological similarity," *Bioinformatics*, vol. 29, no. 3, pp. 355–364, 2013.

[35] V. Memisevic, T. Milenkovic, and N. Przulj, "Complementarity of network and sequence information in homologous proteins," *Journal of Integrative Bioinformatics*, vol. 7(3):135, 2010.

[36] T. Kato, K. Tsuda, and K. Asai, "Selective integration of multiple biological data for supervised network inference," *Bioinformatics*, vol. 21, no. 10, pp. 2488–2495, May 2005. [Online]. Available: http://dx.doi.org/10.1093/bioinformatics/bti339

[37] Y. Qi, J. Klein-Seetharaman, and Z. Bar-Joseph, "A mixture of feature experts approach for protein-protein interaction prediction," *BMC Bioinformatics*, vol. 8(Suppl 10):S6, 2007.

[38] J.-P. Vert and Y. Yamanishi, "Supervised graph inference," in *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press, 2005, pp. 1433–1440. [Online]. Available: http://eprints.pascal-network.org/archive/00001405/

[39] Y. Yamanishi, J.-P. Vert, and M. Kanehisa, "Supervised enzyme network inference from the integration of genomic data and chemical information," *Bioinformatics*, vol. 21, no. 1, pp. 468–477, Jan. 2005. [Online]. Available: http://dx.doi.org/10.1093/bioinformatics/bti1012

[40] N. Zaki, S. Lazarova-Molnar, W. El-Hajj, and P. Campbell, "Protein-protein interaction based on pairwise similarity." *BMC Bioinformatics*, vol. 10, p. 150, 2009.

[41] T. F. Smith and M. S. Waterman, "Identification of common molecular subsequences." *J Mol Biol*, vol. 147, no. 1, pp. 195–197, 1981. [Online]. Available: http://view.ncbi.nlm.nih.gov/pubmed/7265238

[42] M. Hue and J.-P. Vert, "On learning with kernels for unordered pairs," in *ICML*, 2010, pp. 463–470.

[43] T. Mohamed, C. J.G., and G. M.K., "Active learning for human protein-protein interaction prediction," *BMC Bioinformatics*, vol. 11(Suppl 1):S57, 2010.

[44] Y. Qi, O. Tastan, J. Carbonell, J. Klein-Seetharaman, and J. Weston, "Semi-supervised multi-task learning for predicting interactions between hiv-1 and human proteins," *Bioinformatics*, vol. 26, no. 18, pp. i645–i652, 2010.

[45] H. Kashima, Y. Yamanishi, T. Kato, M. Sugiyama, and K. Tsuda, "Simultaneous inference of biological networks of multiple species from genome-wide data and evolutionary information," *Bioinformatics*, vol. 25, no. 22, pp. 2962–2968, 2009.

[46] C.-Y. Yu, L.-C. Chou, and D. T.-H. Chang, "Predicting protein-protein interactions in unbalanced data using the primary structure of proteins," *BMC Bioinformatics*, vol. 11:167, 2010.

[47] M. Hue, M. Riffle, J. Vert, and W. Noble, "Large-scale prediction of protein-protein interactions from structures," *BMC Bioinformatics*, vol. 11:144, 2010.

[48] D. Ekman, S. Light, A. K. Bjorklund, and A. Elofsson, "What properties characterize the hub proteins of the protein-protein interaction network of saccharomyces cerevisiae?" *Genome Biology*, vol. 7, no. R45, 2006.

[49] D. J. Watts and S. H. Strogatz, "Collective dynamics of /'small-world/' networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.

[50] F. Chung and L. Lu, "The average distances in random graphs with given expected degrees," *Internet Mathematics*, vol. 1, pp. 15 879–15 882, 2002.

[51] ——, "Connected components in random graphs with given expected degree sequences," *Annals of Combinatorics*, vol. 6, no. 2, pp. 125–145, 2002.

[52] A. L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science (New York, N.Y.)*, vol. 286, no. 5439, pp. 509–512, 1999.

[53] D. Pennock, G. W. Flake, S. Lawrence, E. J. Glover, and C. L. Giles, "Winners don't take all: Characterizing the competition for links on the web," in *Proceedings of the National Academy of Sciences*, 2002, pp. 5207–5211.

[54] M. Mitzenmacher, "A brief history of generative models for power law and lognormal distributions." *Internet Mathematics*, vol. 1, no. 2, 2003.

[55] A. Todor, A. Dobra, and T. Kahveci, "Characterizing the topology of probabilistic biological networks," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 10, no. 4, pp. 970–983, 2013.
[56] O. Tastan, Y. Qi, J. Carbonell, and K.-S. J., "Prediction of interactions between hiv-1 and human proteins by information integration," *Proceedings of the Pacific Symposium on Biocomputing*, vol. 14, pp. 516–527, 2009.
[57] C. von Mering, R. Krause, B. Snel, M. Cornell, S. Oliver, S. Fields, and P. Bork, "Comparative assessment of large-scale data sets of protein-protein interactions," *Nature*, vol. 417(6887), pp. 399–403, 2002.
[58] J. Ma, S. Wang, Z. Wang, and J. Xu, "MRFalign: Protein homology detection through alignment of markov random fields," *PLoS Comput Biol*, vol. 10, no. 3, p. e1003500, 03 2014.
[59] J. Söding, "Protein homology detection by HMM–HMM comparison," *Bioinformatics*, vol. 21, no. 7, pp. 951–960, Apr. 2005.
[60] M. Deng, S. Mehta, F. Sun, and C. T., "Inferring domain-domain interactions from protein-protein interactions," *Genome Res*, vol. 12, no. 10, pp. 1540–8, 2002.
[61] J. Quinlan, *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, 1993.
[62] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, Oct. 2001.
[63] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: an update," *SIGKDD Explor. Newsl.*, vol. 11, no. 1, pp. 10–18, Nov. 2009. [Online]. Available: http://doi.acm.org/10.1145/1656274.1656278
[64] Y. Qi, Z. Bar-Joseph, and J. Klein-Seetharaman, "Evaluation of different biological data and computational classification methods for use in protein interaction prediction," *Proteins: Structure, Function, and Bioinformatics*, vol. 63, no. 3, pp. 490–500, 2006.
[65] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon, "Network motifs: Simple building blocks of complex networks," *Science*, vol. 298, no. 5594, pp. 824–827, 2002.
[66] G. Robins, P. Pattison, Y. Kalish, and D. Lusher, "An introduction to exponential random graph (p*) models for social networks," *Social Networks*, vol. 29, no. 2, pp. 173–191, May 2007.
[67] R. Tanaka, Y. T.M., and D. J, "Some protein interaction data do not exhibit power law statistics," *FEBS Letters*, vol. 579, pp. 5140–5144, 2005.

Adriana Birlutiu received her B.Sc. in Mathematics and Computer Science in 2004 and M.Sc. in Computer Science in 2005 from Babeş-Bolyai University, Cluj-Napoca, Romania. She obtained her Ph.D. in 2011 from the Institute for Computing and Information Sciences, Radboud University Nijmegen, the Netherlands. She is currently an assistant professor at the Faculty of Science, "1 Decembrie 1918" University of Alba-Iulia, Romania. Her research interests include Bayesian machine learning, computational intelligence and bioinformatics.

Florence d'Alché-Buc received a diploma of engineering of Telecom ParisTech in 1990 and both the M.Sc. and Ph.D. degrees in Computer Science from University Paris Sud, France, in 1991 and 1993, respectively. After two years at Philips Research laboratories (Limeil-Brévannes, France) as a research scientist, she joined Université Pierre et Marie Curie in Paris, France and worked there as an associate professor until 2004 and then moved to University of Evry, in 2004 as a full professor in Computer Science. Prof. d'Alché-Buc's research interests include machine learning and bioinformatics with a focus on kernel methods and dynamical modeling.

Tom Heskes received both the M.Sc. and Ph.D. degrees in Physics from the University of Nijmegen, The Netherlands, in 1989 and 1993, respectively. After a year of postdoctoral work at the Beckman Institute in Urbana-Champaign, Illinois, he worked from 1994 to 2004 as a researcher for SNN, Nijmegen. Between 1997 and 2004 he headed the company SMART Research BV, specialized in applications of neural networks and related techniques. Since 2004 he works at the Institute for Computing and Information Sciences at the Faculty of Science, Radboud University Nijmegen, where he is now a full professor in artificial intelligence. Prof. Heskes is Editor-in-Chief of Neurocomputing. His research interests include theoretical and practical aspects of neural networks, machine learning, and probabilistic graphical models.