Using the Duplication-Divergence Network Model to Predict Protein-Protein Interactions

Nicolás López-Rozo(✉), Jorge Finke, and Camilo Rocha

Department of Electronics and Computer Science, Pontificia Universidad Javeriana,
Cali, Colombia
nicolaslopez@javerianacali.edu.co, jfinke@javerianacali.edu.co,
camilo.rocha@javerianacali.edu.co

Abstract. Interactions between proteins are key to most biological processes, but thorough testing can be costly in terms of money and time. Computational approaches for predicting such interactions are an important alternative. This study presents a novel approach to this prediction using calibrated synthetic networks as input for training a decision tree ensemble model with relevant topological information. This trained model is later used for predicting interactions on the human interactome, as a case study. Results show that deterministic metrics perform better than their stochastic counterparts, although a random forest model shows a feature combination case with comparable precision results.

Keywords: Duplication-Divergence model · Protein Interaction Prediction · Edge Embeddings · Human Interactome.

1 Introduction

Interactions between proteins are key to most biological processes inside cells [12]. Protein-protein interactions (PPIs) underlie a variety of interdependent mechanisms, including signal transduction, homeostasis control, and stress responses. They play an important role in physiological and developmental processes such as protein phosphorylation, transcriptional co-factor recruitment, and transporter activation [32]. Based on the outcome of numerous Yeast-2-Hybrid testing (Y2H), networks of PPIs can be constructed [20,25]. However, identifying protein interactions through Y2H is costly both in terms of time and resources [11,15]. For example, constructing a PPI network for an organism with 2,000 proteins requires testing of about 2 million potential interactions. Moreover, it has been shown that a large amount of false negatives and positives (sometimes near 20%) are found in PPIs generated from several techniques such as Y2H, Tandem Affinity Purification Mass Spectrometry, and ChIP-Seq [14]. In that regard, computational tools for narrowing down the combinatorial search

H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2016 2022, SCI 1077, pp. 322–334, 2023.
https://doi.org/10.1007/978-3-031-21127-0_27

space of such interactions offer a cost-effective alternative, especially for organisms that produce a large number of proteins. A further limitation is that many interactomes, such as the human one, lack many relations [27].

This paper explores the prediction of PPIs using a computational approach and information available from an existing PPI network (in-silico prediction). The proposed approach leverages the intrinsic topological information of the neighborhood of a protein using deterministic and non-deterministic method to train a classification model in charge of predicting the existence of interactions. Several decision-tree models are tested, in particular random forests [22]; they seem to perform better at PPI prediction. The trained model is used on a PPI constructed from experimental evidence; the precision of the model is validated against a more recent version of the PPI for the same organism.

This study applies the proposed approach to the human interactome. XGBoost and random forest classifiers are selected for the prediction task, and are trained using synthetic networks obtained from the duplication-divergence (DD) graph model [9]. The topological information from the synthetic networks is extracted with two different approaches: the first one computes higher degree neighborhood scores (closures), and the second one uses random walks and an unsupervised deep learning model (embeddings). The prediction PPI is the interactome from the Human Reference Protein Interactome Mapping Project [13] and the validation PPI is an assembly of 12 databases constructed by Gysi et al [5]. The results show that deterministic approaches achieve greater precision than stochastic ones, but also suggest that a greater spectrum of classification models and embedding techniques must be explored. Ultimately, these results highlight the need of broader exploration of in-silico approaches that, supported by biologically relevant information, elucidate the interactions of proteins in any given organism.

Outline. The remainder of the paper is organized as follows. Section 2 presents preliminary notions used throughout the paper. Section 3 presents the proposed approach, and introduces the steps required for the generation of the synthetic networks. In Section 4 a case study on the human interactome is developed. Section 5 summarizes related work and concludes the paper.

2 Preliminaries

This section presents preliminaries on closures on graphs, feature learning with node2vec, and ensemble learning.

2.1 Network Closures

Let $G = (V, E)$ be an undirected graph with $n = |V|$ proteins as *nodes* and $m = |E|$ pairwise interactions between nodes as *edges*. The graph G can be represented as a Boolean adjacency matrix A_G of size n^2: for $0 \leq i, j < n$, the entry $A_G(i, j)$ is 1 if there is a direct interaction between proteins i and j, and 0 otherwise. Since G is undirected, A_G is symmetric.

Computing the number of paths of length 2 and 3 in G can be done with $(A_G)^2$ and $(A_G)^3$, respectively. Another important metric in PPI analysis is the degree-normalized count of the paths of length 3, which is computed as in [10]:

$$L3(i,j) = \sum_{p,q \in V} \frac{A_G(i,p) \cdot A_G(p,q) \cdot A_G(q,j)}{\sqrt{k(p) \cdot k(q)}},$$

where $k(i)$ is the number of neighbors of v_i, known as its *degree*. Local-community information extracted from the induced graph of the paths of length 3 is also relevant to predict PPI networks [17]. This metric, known as $CH_2 - L_3$, maximizes internal links in local communities and minimizes external ones, according to a topology-driven mechanistic model. These four metrics will be used throughout this paper as *closures*.

2.2 Feature Learning of a Network and node2vec

The unsupervised algorithm node2vec for feature learning in networks is based on natural language processing [4]. It treats the problem of feature learning in a network as a maximum likelihood optimization problem. The function $f : V \rightarrow \mathbb{R}^d$ to be learned maps each node u to the d-tuple $f(u)$ of features of u. Therefore, f is represented as a matrix of size $|V| \times d$. The optimization problem becomes feasible when the technique assumes *conditional independence* and *spatial symmetry*. Conditional independence means that the likelihood of observing a node does not depend on any other node's observation. Spatial symmetry means that two neighbor nodes have a symmetric effect over each other in the d-dimensional feature space.

For each node $u \in V$, the expression $N(u)$ denotes the *network neighborhood* of u. This set is generated up to a fixed sampling strategy and a fixed size. The sampling strategy is a combination of depth-first (DFS) and breadth-first (BFS) sampling. Starting from vertex u, the former selects the nodes farthest in a path from u, while the latter selects the ones closest to u. Based on these two strategies, node2vec computes random walks for each node u in combination with user-provided weights for back-edges and forward edges to build $N(u)$. Finally, node2vec uses deep learning networks to learn the associated function $f : V \rightarrow \mathbb{R}^d$ of features of G from all the network neighborhood sets $N(u)$.

Since the goal of this work is link prediction, the function to be learned needs to have the edges E of G as its domain. In particular, the feature function has type $g : E \rightarrow \mathbb{R}^d$ and it is defined to obtain a d-tuple representation $g(u,v) = g(v,u)$ for each link $(u,v) \in E$ based on the d-tuples $f(u)$ and $f(v)$ computed by node2vec. Several binary operators can be considered for learning edge features with this approach. For instance, piece-wise average $\left(\frac{f_i(u)+f_i(v)}{2}\right)$, piece-wise multiplication $(f_i(u) * f_i(v))$, weighted-L1 $(|f_i(u) - f_i(v)|)$, and weighted-L2 $|f_i(u) - f_i(v)|^2$, where f_i denotes the i-th projection over $f(u)$. The piece-wise multiplication alternative, also known as Hadamard operator, is used in this study for obtaining the d-dimensional edge representation in g.

2.3 Random Forest, Gradient Boosted Trees, and XGBoost

Ensemble learning is a branch of supervised learning algorithms in which several base estimators are trained and their predictions combined in order to improve robustness over a single estimator. Two big groups are usually distinguished: *averaging methods* and *boosting methods*. Random forest is an example of averaging methods, while gradient boosted trees are representative for boosted methods.

In the random forest implementation of scikit-learn [19], each decision tree is built from a bootstrap sample from the training set (i.e. sampling with replacement). The node splitting process during the tree construction finds the best split either for a random subset of features or from all input features. On the other hand, the XGBoost library implements supervised learning models based on gradient boosted decision forests [2]. The main difference with random forest is that an added estimator tries to minimize the current emsemble error, instead of being an independent predictor.

A set $E' \supseteq E$ is fabricated for training purposes and a dataset T of size $|E'| \times d$: for each $e \in E'$, the corresponding row is $g(e)$, even if $e \notin E$. The algorithms in XGBoost are used to infer the Boolean function $l : T \rightarrow \mathbb{B}$ so that $l(g(e))$ is the supervisory signal for $e \in E'$. The labeling function l is then used to select candidate edges in $E' \setminus E$ to add to G.

The problem of learning the optimal structure of a decision tree is known to be NP-complete [7]. In practice, it is usual to define an objective function to be minimized for learning l, so that it has an internal structure approximating the optimal one. In this case, *training loss* and a *regularization term* are added for the objective function. Training loss is usually calculated as the mean squared error between prediction and observation. The regularization term is, intuitively, a measure penalizing the depth of the decision tree.

3 Training with the Duplication-Divergence Model

This section presents an overview of the duplication-divergence network model [8] and proposes an approach for using the model to predict links in protein-protein interaction networks.

3.1 The Model

From a biological perspective, it is widely accepted that a new protein appears as a copy of an existing one in the interactome [26]. This process is known as *duplication* and is the main driving mechanism of evolution in PPI networks. Random mutations also occur, leading to differences between the source and duplicated proteins; thus, a certain degree of *divergence* is expected.

The duplication-divergence (DD) model was first proposed by Kim et al. in [9] as a network model, but the variant used in this study can be attributed to Ispolatov et al [8]. The model takes as input a given (undirected) connected graph G_0 and a probability δ. For any given time $t \geq 0$, the graph G_t evolves to the graph G_{t+1} as follows:

Duplication. A node u from the graph G_t is chosen at random to create v, which is connected to all neighbors of u, so that $N(u) = N(v)$.

Divergence. Each of the edges from v to $N(u)$ is deleted with probability δ. If at least one edge remains, the replica is preserved; otherwise, the attempt is considered futile and the network does not change.

The criteria of failed attempts is biologically relevant, since it assures that the resulting graph is connected. Leaving nodes with no edges produces disconnected nodes whose survival during the evolution process is questionable. Critical biological pathways, such as degradation, are run by housekeeping proteins that are highly connected and in principle have links to all proteins in a species [8].

3.2 Parameter Estimation

It is key to define the removal probability δ to use the DD model. As addressed by several authors, the DD model preserves the most relevant topological properties of the PPI networks: average degree, power-law exponent, average clustering coefficient, average path length, and observed bipartite cliques [8,18,23]. Given that the number of nodes is already a direct parameter of the DD model, the number of edges is calibrated only with the removal probability δ.

Let $G_p = (V_p, E_p)$ be the PPI network to be predicted and $G_v = (V_v, E_v)$ the target PPI network. The idea is to generate a synthetic network $G_s = (V_s, E_s)$ by finding the parameter δ so that $G_s := \mathrm{DD}(|V_p|, \delta)$ has a similar amount of edges (i.e., $|E_v| \approx |E_s|$). Due to the intrinsic stochastic nature of the model, the considered number of edges is the average of K models generated for the same parameters (i.e., $\overline{|E_s|}$).

Once parameter δ is estimated, a number of repetitions R is defined and then R synthetic networks are created, having in mind a maximum admissible deviation factor $\varDelta_E$ from the expected number of edges $|E_v|$:

$$(1 - \varDelta_E)|E_v| \leq \overline{|E_s|} \leq (1 + \varDelta_E)|E_v|.$$

On average, the number of edges generated by the DD model has a monotonic behavior and is inversely proportional to the removal probability δ. However, variance among repetitions with the same parameters is considerable, so that it is impractical to find δ with a precision lower than 0.001.

3.3 The Approach

After using the DD model with the calibrated parameters to obtain the R networks G_i, $0 \leq i < R$, the following procedure is carried out for each of them. Keep in mind that this description is intended to be general. Details about the application of this approach on the human interactome are given in Section 4.

1. From the complete synthetic network G_i some edge features are extracted, deterministically (e.g., L_3) or stochastically (e.g., node2vec embeddings).

2. A binary classification model $M^{(i)}$ is trained with a subset of those features, using all existing edges as positive evidence (Class 1) and a random subset of non-existing edges as negative evidence (Class 0). To reduce bias, the number of edges of each class is the same (balanced dataset).
3. The same edge features computed for G_i are now computed for the PPI network to be predicted G_p.
4. The model $M^{(i)}$ is used on all the unlabeled edges of the PPI network G_p; that is, on $(V \times V) \setminus G_p(E)$, using the computed edge features.
5. The predicted edges are ranked in descending order by their probability of existence according to the model and the top B are selected.
6. The top B predictions are compared with the target PPI network G_v and the precision is then computed.

Figure 1 shows a visual representation of the approach. Note that for the real PPI network, node features are computed independent to the synthetic networks. The hyperdimensional representation of the edges on the real PPI network is used to feed the classification models trained on the synthetic networks.

4 Prediction of Interactions on the Human Interactome

This section presents the approach for predicting PPIs on the human interactome using synthetic networks. The goal of this case study is to exhibit the proposed approach, having in mind the predictive limitations due to the incompleteness of the current human interactome [5,27].

4.1 Data

Two versions of the human interactome are used: the interactome used for the prediction G_p corresponds to HI-union and is available at the website of the Human Reference Protein Interactome Mapping Project [13]. It consists of 9,094 proteins and 64,006 interactions. The second interactome G_v, which is used to validate the predicted interactions, corresponds to a dataset published by Gysi et al. consisting of 18,505 proteins and 327,924 interactions [5]. It is an assembly of 21 public databases compiling experimentally derived PPI data.

It is worth noting that each interactome has a different identifier type: HI-union (G_p) uses Ensembl GeneID and HI-2021 (G_v) uses EntrezID. Conversion of G_v from EntrezID to Ensembl GeneID was done using BioDBnet web services [16]. Not all proteins could be translated, so the resulting network after translation had 18,173 proteins and 321,360 interactions.

4.2 Data Pre-processing

Since prediction depends almost entirely on the PPI network being connected, the greatest connected component from G_p was used for this study. This leaves G_p with $|V_p| = 8,986$ proteins connected by $|E_p| = 63,203$ interactions. On the other hand, the greatest connected component in G_v has 321,360 edges.

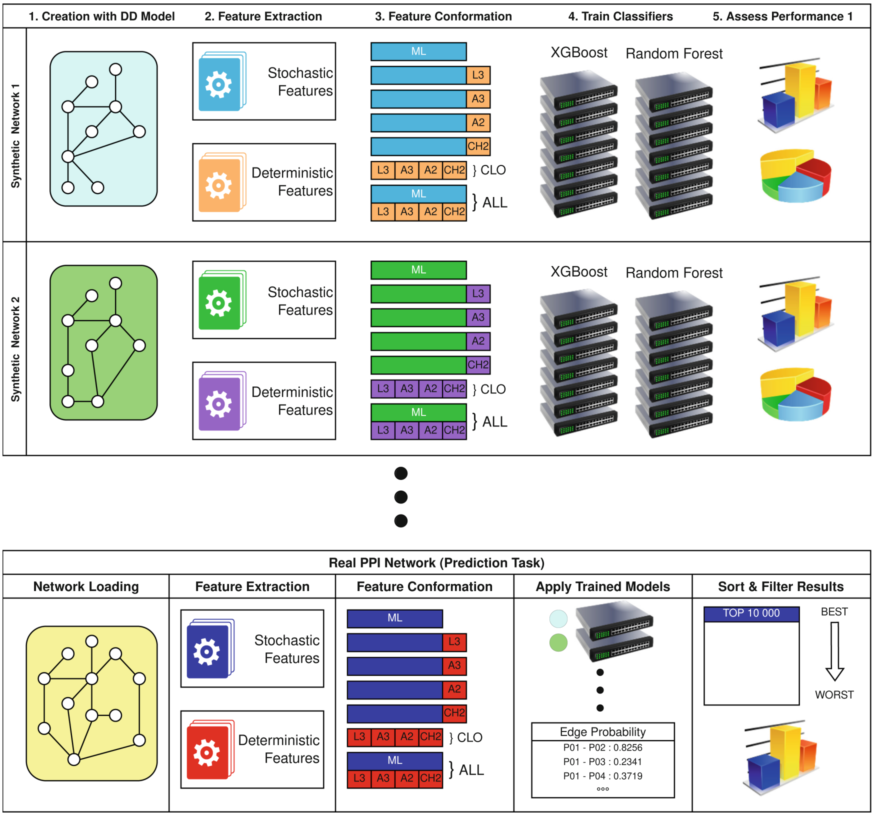

Fig. 1. Diagram of the proposed approach for PPI prediction using synthetic networks.

4.3 DD Model Calibration

The calibration process was carried out by a binary search of the parameter δ. The goal was a network with the expected amount of edges of the greatest connected component in G_v. The maximum admissible deviation factor was set to $\Delta_E = 0.1$ and the number of repetitions considered to average the number of edges was set to $K = 100$. The calibrated value is $\delta = 0.263$ and its computation took around 3 minutes.

4.4 Feature Conformations

The feature conformations used in this study are:

ML. After obtaining node2vec vector representations of the nodes, the Hadamard operator is used to obtain the edge embeddings. In total, 128 dimensions were retrieved with an unbiased search ($p = q = 1$ in node2vec) and default parameters. The implementation used in this study can be found in the karateclub library [21]. This model type is addressed as $M_{\mathrm{ML},\cdot}$.

L3. It uses the features from ML and includes the L_3 score. This model type is addressed as $M_{\mathrm{L3},\cdot}$.

A3. It includes the features from ML, as well as the raw count of paths of length 3 ($A_3 = (A_G)^3$). This model type is addressed as $M_{\mathrm{A3},\cdot}$.

A2. It includes ML features and the raw count of paths of length 2 ($A_2 = (A_G)^2$). This model type is addressed as $M_{\mathrm{A2},\cdot}$.

CH2. This conformation uses the features from ML and includes the score for the local communities of length 3 ($CH_2 - L_3$). This model type is addressed as $M_{\mathrm{CH2},\cdot}$.

CLO. It includes only the deterministic features: L_3, A_3, A_2 and $CH_2 - L_3$. This model type is addressed as $M_{\mathrm{CLO},\cdot}$.

ALL. This conformation uses all the features from ML and CLO. This model type is addressed as $M_{\mathrm{ALL},\cdot}$.

4.5 Classification Models

The classification models used in this study are XGBClassifier from the XGBoost library [2] and Random Forest Classifier from scikit-learn library [19], further addressed here as $M_{\cdot,\mathrm{XGB}}$ and $M_{\cdot,\mathrm{RF}}$, respectively.

4.6 Using the Proposed Approach

The prediction task consists of several steps, as mentioned in Section 3.3. A total of $R = 10$ DD model networks were generated using the calibrated parameters for a more robust assessment of PPI interactions. The creation of the synthetic networks took around 40 seconds. In total, 140 models ${M^{(i)}}_{F,C}$ were trained: $R = 10$ repetitions for each of the 7 feature conformations (F) and each of the classification models (C). After that, each model was used to predict the unlabeled edges of G_p and the top $B = 10,000$ unlabeled edges with the highest existence probability were selected. Finally, the evaluation metrics were computed. Each repetition of all models took around 9 hours of computing time.

The prediction results are presented in Table 1 as a 10-fold average precision for each of the 14 combinations of classification model and feature conformation. Precision values for the deterministic prediction are presented in the last column. Note that predictions overall present a low precision. The highest precision is achieved by the deterministic approach using CH_2-L_3. Out of the methods using machine learning predictors, the random forests model trained to consider only the information from the 4 closures ($M_{\mathrm{CLO},\mathrm{RF}}$) achieves the highest precision, which is comparable with the lowest precision from the deterministic subset of predictions. The set of features to give the lowest average prediction by using the XGBoost predictor is the combination of the 4 closure-based metrics. For the random forest classifier, combining all closure features and the embedding results in the lowest precision model.

Table 1. Precision results for top-10,000 edges for the combination of the 2 models and 7 feature conformations (average over 10 models trained on synthetic networks), as well as the deterministic prediction using each metric

Prediction:	Stochastic		Deterministic
Trained model:	XGBoost (%)	Random forest (%)	None (%)
L3	0.228	0.220	1.090
A3	0.230	0.230	0.450
A2	0.196	0.216	1.210
CH2	0.302	0.142	1.380
ML	0.223	0.214	–
ALL	0.227	0.106	–
CLO	0.115	0.481	–

5 Related Work and Concluding Remarks

Important work on modeling networks governed by duplication of nodes has been done under several approaches. Some authors find that the DD model is one of the models that best fits the distribution on some PPI datasets [24] and that it preserves the observed bipartite cliques [23]. Chung et al. analytically derive relationships associating the power-law distribution exponent β and the DD parameters using combinatorial probabilistic methods [3]. Their work involves both full and partial duplication of a node, and conclude that partial duplication (i.e., retaining only a fraction of the duplicated edges) generates networks consistent with biological networks. They also show that the power-law exponent for large graphs does not depend on the starting graph, but on the growth process itself.

Other important advances in the DD model are made by [8,9,18]. Kim et al. work on a DD model which –besides gene duplication– allows nodes to connect to the duplicated ones with a probability $\frac{r}{t}$ (process called *mutation*). A main finding is that in a mutation-dominated growth, disconnected components merge into larger ones (known as percolation), while in a duplication-dominated environment the process is not self-averaging and each outcome is itself singular [9]. Pastor-Satorras et al. find the characteristic degree distributions for such DD models, and compare the topological properties of the yeast PPI with a calibrated synthetic network in terms of average degree, power-law exponent, average clustering coefficient, and average path length [18]. Ispolatov et al. assess a model as presented in this paper, in which a duplicated node that disconnects from the network is discarded and the network does not change [8]. They analyze the properties of total duplication models, as well as highly divergent ones where the resulting networks are trees because only one edge is added to each duplicated node. They find a range of values for the DD parameter that generates a self-averaging network. To the best of the authors' knowledge, no use of synthetic networks with the DD network model has found a way for edge prediction in PPIs.

Prediction of PPIs is a highly active topic in bioinformatics. Several reviews make the effort of compiling different techniques and methods available [1,22,31]. Most techniques are based on one or several of the following approaches: to exploit the existing network connectivity [10,29], to leverage from the amino acid sequences [28], to use functional annotations [6,33], and to apply machine learning techniques to extract features or predict PPIs [30].

Two complementary approaches support this work. First, the work by Kovacs et al. elucidates the connectivity-based mechanics of PPIs by computing paths of specific distances, being paths of length 2 and 3 the most relevant [10]. Furthermore, they define the degree-normalized count of paths of length 3 (L_3), which performs consistently to predict novel PPIs on several organisms in a deterministic way. On the other hand, Xiao and Deng leverage the usage of deep learning models for the extraction of embeddings [29]. Based on that high-degree neighborhood information, they design a model for predicting PPIs using graph convolutional networks and consider CH_2-L_3 as an important deterministic approach to PPI prediction. These ideas are merged with the DD model for creating synthetic networks that mimic the structure of real PPI networks. The reason to use synthetic networks is that its connectivity is final. That is, an edge either exist or not, whereas real PPI networks mainly have certainty on the observed edges because the rest of the edges are considered unlabeled. On the other hand, decision tree classification methods usually have the best performance when predicting PPIs compared to other machine learning methods [22]. However, great care should be taken because they are prone to overfitting and sensitive to noise and correlated features given the nature of PPIs.

The results in this paper suggest that the deterministic approaches perform better for predicting PPIs. In general, metrics based on paths of length 3 (A3, L3, and CH2-L3) seem to perform better than the paths of length 2 (A2), which is consistent with [10]. Two of the most recent human PPI networks found in literature were used for the prediction (G_p) and performance evaluation (G_v), comprising 9.7% and 49.4% of the expected size of the human interactome ($\sim 650,000$ edges), respectively [27]. These datasets, in comparison with the HI-II-14 version used in other studies, may show a better picture on the real PPI prediction performance for any given model. Although G_v contains many interactions more than G_p, no total assessment can be done on the edges which are reported as non-existing, because some of those interactions might actually occur (but have not been documented). The main contribution is the proposed approach itself; to the best of the authors' knowledge, no PPI prediction task has attempted the use of synthetic networks. This seems a tractable approach from a resource- and computational-wise perspective.

As future work, the exploration of other tree-based classification models may show a more significant precision result. Namely, ensemble alternatives combining different types of models or models with different parameters may overcome the overfitting problems of random forest classifiers, as well as their sensitivity to noise. Furthermore, the exploration of datasets for other organisms might elucidate hidden biological mechanisms related to relevant closures or feature combinations.

Acknowledgments. This work was funded by the OMICAS program: Optimización Multiescala In-silico de Cultivos Agrícolas Sostenibles (Infraestructura y Validación en Arroz y Caña de Azúcar), anchored at the Pontificia Universidad Javeriana in Cali and funded within the Colombian Scientific Ecosystem by The World Bank, the Colombian Ministry of Science, Technology and Innovation, the Colombian Ministry of Education, and the Colombian Ministry of Industry and Tourism, and ICETEX, under GRANT ID: FP44842-217-2018.

References

1. Chang, J.W., Zhou, Y.Q., Ul Qamar, M.T., Chen, L.L., Ding, Y.D.: Prediction of protein-protein interactions by evidence combining methods. Int. J. Mol. Sci. **17** (2016)
2. Chen, T., Guestrin, C.: Xgboost: A scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785-794. KDD '16, ACM, New York, NY, USA (2016)
3. Chung, F., Lu, L., Dewey, T.G., Galas, D.J.: Duplication models for biological networks. J. Comput. Biol. J. Comput. Mol. Cell Biol. **10**, 677–87 (2003)
4. Grover, A., Leskovec, J.: node2vec: Scalable feature learning for networks (2016). https://arxiv.org/abs/1607.00653
5. Gysi, D.M., Ítalo do Valle, Zitnik, M., Ameli, A., Gan, X., Varol, O., Ghiassian, S.D., Patten, J.J., Davey, R.A., Loscalzo, J., Barabási, A.L.: Network medicine framework for identifying drug-repurposing opportunities for COVID-19. Proc. Nat. Acad. Sci. **118**(19) (2021)
6. Halder, A.K., Bandyopadhyay, S.S., Chatterjee, P., Nasipuri, M., Plewczynski, D., Basu, S.: JUPPI: A multi-level feature based method for PPI prediction and a refined strategy for performance assessment. IEEE/ACM Trans. Comput. Biol. Bioinform. **19**, 531–542 (2022)
7. Hyafil, L., Rivest, R.L.: Constructing optimal binary decision trees is NP-complete. Inform. Process. Lett. **5**(1), 15–17 (1976)
8. Ispolatov, I., Krapivsky, P.L., Yuryev, A.: Duplication-divergence model of protein interaction network. Physical review. E-Stat. Nonlin. Soft Matt. Phys. **71**, 061911 (2005)
9. Kim, J., Krapivsky, P.L., Kahng, B., Redner, S.: Infinite-order percolation and giant fluctuations in a protein interaction network. Physical review. E-Stat. Nonlin. Soft Matt. Phys. **66**, 055101 (2002)
10. Kovács, I.A., Luck, K., Spirohn, K., Wang, Y., Pollis, C., Schlabach, S., Bian, W., Kim, D.K., Kishore, N., Hao, T., Calderwood, M.A., Vidal, M., Barabási, A.L.: Network-based prediction of protein interactions. Nat. Commun. **10**, 1240 (2019)
11. Laraia, L., McKenzie, G., Spring, D.R., Venkitaraman, A.R., Huggins, D.J.: Overcoming chemical, biological, and computational challenges in the development of inhibitors targeting protein-protein interactions. Chem. Biol. **22**, 689–703 (2015)
12. Lin, J.S., Lai, E.M.: Protein-protein interactions: Co-immunoprecipitation. In: Journet, L., Cascales, E. (eds.) Bacterial Protein Secretion Systems: Methods and Protocols, pp. 211–219. Springer, New York, New York, NY (2017)

13. Luck, K., Kim, D.K., Lambourne, L., Spirohn, K., Begg, B.E., Bian, W., Brignall, R., Cafarelli, T., Campos-Laborie, F.J., Charloteaux, B., Choi, D., Coté, A.G., Daley, M., Deimling, S., Desbuleux, A., Dricot, A., Gebbia, M., Hardy, M.F., Kishore, N., Knapp, J.J., Kovács, I.A., Lemmens, I., Mee, M.W., Mellor, J.C., Pollis, C., Pons, C., Richardson, A.D., Schlabach, S., Teeking, B., Yadav, A., Babor, M., Balcha, D., Basha, O., Bowman-Colin, C., Chin, S.F., Choi, S.G., Colabella, C., Coppin, G., D'Amata, C., De Ridder, D., De Rouck, S., Duran-Frigola, M., Ennajdaoui, H., Goebels, F., Goehring, L., Gopal, A., Haddad, G., Hatchi, E., Helmy, M., Jacob, Y., Kassa, Y., Landini, S., Li, R., van Lieshout, N., MacWilliams, A., Markey, D., Paulson, J.N., Rangarajan, S., Rasla, J., Rayhan, A., Rolland, T., San-Miguel, A., Shen, Y., Sheykhkarimli, D., Sheynkman, G.M., Simonovsky, E., Taşan, M., Tejeda, A., Tropepe, V., Twizere, J.C., Wang, Y., Weatheritt, R.J., Weile, J., Xia, Y., Yang, X., Yeger-Lotem, E., Zhong, Q., Aloy, P., Bader, G.D., De Las Rivas, J., Gaudet, S., Hao, T., Rak, J., Tavernier, J., Hill, D.E., Vidal, M., Roth, F.P., Calderwood, M.A.: A reference map of the human binary protein interactome. Nature **580**(7803), 402–408 (2020)
14. Ma, C.Y., Liao, C.S.: A review of protein-protein interaction network alignment: From pathway comparison to global alignment. Comput. Struct. Biotechnol. J. **18**, 2647–2656 (2020)
15. Macalino, S.J.Y., Basith, S., Clavio, N.A.B., Chang, H., Kang, S., Choi, S.: Evolution of In Silico Strategies for Protein-Protein Interaction Drug Discovery. Molecules. Basel, Switzerland, pp. 23 (2018)
16. Mudunuri, U., Che, A., Yi, M., Stephens, R.M.: biodbnet: The biological database network. Bioinformatics **25**, 555–6 (2009)
17. Muscoloni, A., Abdelhamid, I., Cannistraci, C.V.: Local-community network automata modelling based on length-three-paths for prediction of complex network structures in protein interactomes, food webs and more. BioRxiv (2018)
18. Pastor-Satorras, R., Smith, E., Solé, R.V.: Evolving protein interaction networks through gene duplication. J. Theor. Biol. **222**, 199–210 (2003)
19. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: machine learning in python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
20. Rajagopala, S.V.: Mapping the protein-protein interactome networks using yeast two-hybrid screens. Adv. Experiment. Med. Biol. **883**, 187–214 (2015)
21. Rozemberczki, B., Kiss, O., Sarkar, R.: Karate Club: An API oriented open-source python framework for unsupervised learning on graphs. In: Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM '20), pp. 3125-3132. ACM (2020)
22. Sarkar, D., Saha, S.: Machine-learning techniques for the prediction of protein-protein interactions. J. Biosci. **44** (2019)
23. Schweiger, R., Linial, M., Linial, N.: Generative probabilistic models for protein-protein interaction networks-the biclique perspective. Bioinformatics **27**, i142-8 (2011)
24. Shao, M., Yang, Y., Guan, J., Zhou, S.: Choosing appropriate models for protein-protein interaction networks: a comparison study. Brief. Bioinform. **15**, 823–38 (2014)
25. Shokri, L., Inukai, S., Hafner, A., Weinand, K., Hens, K., Vedenko, A., Gisselbrecht, S.S., Dainese, R., Bischof, J., Furger, E., Feuz, J.D., Basler, K., Deplancke, B., Bulyk, M.L.: A comprehensive drosophila melanogaster transcription factor interactome. Cell Rep. **27**, 955-970.e7 (2019)

26. Sreedharan, J.K., Turowski, K., Szpankowski, W.: Revisiting parameter estimation in biological networks: Influence of symmetries. IEEE/ACM Trans. Comput. Biol. Bioinform. **18**, 836–849 (2021)
27. Stumpf, M.P.H., Thorne, T., de Silva, E., Stewart, R., An, H.J., Lappe, M., Wiuf, C.: Estimating the size of the human interactome. Proc. Nat. Acad. Sci. U.S.A. **105**, 6959–64 (2008)
28. Sun, T., Zhou, B., Lai, L., Pei, J.: Sequence-based prediction of protein protein interaction using a deep-learning algorithm. BMC Bioinform. **18**, 277 (2017)
29. Xiao, Z., Deng, Y.: Graph embedding-based novel protein interaction prediction via higher-order graph convolutional network. PLOS ONE **15**, e0238915 (2020)
30. Yao, Y., Du, X., Diao, Y., Zhu, H.: An integration of deep learning with feature embedding for protein-protein interaction prediction. Peer J. **7**, e7126 (2019)
31. Zahiri, J., Emamjomeh, A., Bagheri, S., Ivazeh, A., Mahdevar, G., Sepasi Tehrani, H., Mirzaie, M., Fakheri, B.A., Mohammad-Noori, M.: Protein complex prediction: a survey. Genomics **112**, 174–183 (2020)
32. Zhang, Y., Gao, P., Yuan, J.: Plant protein-protein interaction network and interactome. Curr. Genom. **11**(1), 40–46 (2010)
33. Zhong, X., Rajapakse, J.C.: Graph embeddings on gene ontology annotations for protein-protein interaction prediction. BMC Bioinform. **21**, 560 (2020)