

Mutual interactors as a principle for phenotype discovery in molecular interaction networks

Sabri Eyuboglu,^{1,*} Marinka Zitnik,^{2,3,4,*} Jure Leskovec^{1†}

¹*Department of Computer Science, Stanford University, Stanford, CA 94305*

²*Department of Biomedical Informatics, Harvard University, Boston, MA 02115*

³*Broad Institute of MIT and Harvard, Cambridge, MA 02142*

⁴*Harvard Data Science, Cambridge, MA 02138*

*Equal Contribution. †Corresponding author. Email: jure@cs.stanford.edu

Biological networks are powerful representations for the discovery of molecular phenotypes. Fundamental to network analysis is the principle—rooted in social networks—that nodes that interact in the network tend to have similar properties. While this long-standing principle underlies powerful methods in biology that associate molecules with phenotypes on the basis of network proximity, interacting molecules are not necessarily similar, and molecules with similar properties do not necessarily interact. Here, we show that molecules are more likely to have similar phenotypes, not if they directly interact in a molecular network, but if they interact with the same molecules. We call this the mutual interactor principle and show that it holds for several kinds of molecular networks, including protein-protein interaction, genetic interaction, and signaling networks. We then develop a machine learning framework for predicting molecular phenotypes on the basis of mutual interactors. Strikingly, the framework can predict drug targets, disease proteins, and protein functions in different species, and it performs better than much more complex algorithms. The framework is robust to incomplete biological data and is capable of generalizing to phenotypes it has not seen during training. Our work represents a network-based predictive platform for phenotypic characterization of biological molecules.

Keywords: Protein-protein interactions, Molecular phenotypes, Network medicine, Graph neural networks

1. Introduction

Molecules in and across living cells are constantly interacting, giving rise to complex biological networks. These networks serve as a powerful resource for the study of human disease, molecular function and drug-target interactions.^{1,2} For instance, evidence from multiple sources suggests that causative genes from the same or similar diseases tend to reside in the same neighborhood of protein-protein interaction networks.^{3–6} Similarly, proteins associated with the same molecular functions form highly-connected modules within protein-protein interaction networks.⁷

These observations have motivated the development of bioinformatics methods that use molecular networks to infer associations between proteins and molecular phenotypes, including diseases, molecular functions, and drug targets.^{8–11} Many of these methods assume that molecular networks obey the organizing principle of homophily: the idea that similarity breeds connection (see Figure

1b).¹² However, while this principle has been well-documented in social networks of many types (e.g., friendship, work, co-membership), it is unclear whether it captures the dynamics of biological networks. If not, existing bioinformatics methods that assume homophily may not realize the full potential of biological networks for scientific discovery.

To better understand the place for homophily in bioinformatics, we consider groups of phenotypically similar molecules (e.g., molecules associated with the same disease, involved in the same function, or targeted by the same drug) and study their interactions in large-scale biological networks. We find that most molecules associated with similar phenotypes do not interact directly in molecular networks, a result which puts into question the assumption of homophily, an assumption that is taken for granted by so many bioinformatics methods.

In fact, a different principle better explains how phenotypic similarity relates to network structure in biology. On average, two molecules that interact directly with one another will have less in common than two molecules that share many *mutual interactors*, just as people in a social network may share mutual friends. We call this the Mutual Interactor principle and validate it empirically on a diverse set of biological networks (see Figure 1c).

Motivated by our findings, we develop a machine learning framework, *Mutual Interactors*, that can predict a molecule's phenotype based on the mutual interactors it shares with other molecules. We demonstrate the power, robustness, and scalability of *Mutual Interactors* on three key prediction tasks: disease protein prediction, drug target identification, and protein function prediction. With experiments across three different kinds of molecular networks (protein-protein interaction, signaling and genetic interaction) and four species (*H. sapiens*, *S. cerevisiae*, *A. thaliana*, *M. musculus*), we find that *Mutual Interactors* substantially outperforms existing methods, with gains in recall up to 61%. Additionally, we show that the weights learned by our method provide insight into the functional properties and druggability of mutual interactors.

Mutual Interactors is an approach based on a different network principle than existing bioinformatics methods. That it can outperform state-of-the-art approaches suggests a need to rethink the fundamental assumptions underlying machine learning methods for network biology.

2. Network connectivity of molecular phenotypes

One way we measure phenotypic similarity between two molecules is by comparing the set of phenotypes (e.g., diseases or functions) associated with each molecule and quantifying their similarity with the Jaccard index. We find that the average Jaccard index of the 62,084 molecule pairs that interact in the human reference interactome (HuRI) is significantly smaller than the average Jaccard index of the 62,084 molecule pairs with most degree-normalized mutual interactors ($p = 2.00 \times 10^{-59}$, dependent *t*-test).¹³ We replicate this finding on three other large-scale interactomes: a PPI network derived from the BioGRID database¹⁴ ($p = 3.56 \times 10^{-26}$) another derived from the STRING database¹⁵ ($p = 1.29 \times 10^{-10}$) and the PPI network compiled by Menche *et al.* ($p = 1.02 \times 10^{-4}$).¹⁶

To further evaluate these two principles (i.e., homophily and Mutual Interactor), we collect 75,744 disease-protein associations¹⁷ and analyze their interactions in the protein-protein interaction network (see Figure 1d-f and Figure A4). For each disease-protein association we compute the fraction of the protein's direct interactors that are also associated with the disease. In only 17.8% of disease-protein associations is this fraction statistically significant ($P < 0.05$, permutation test).

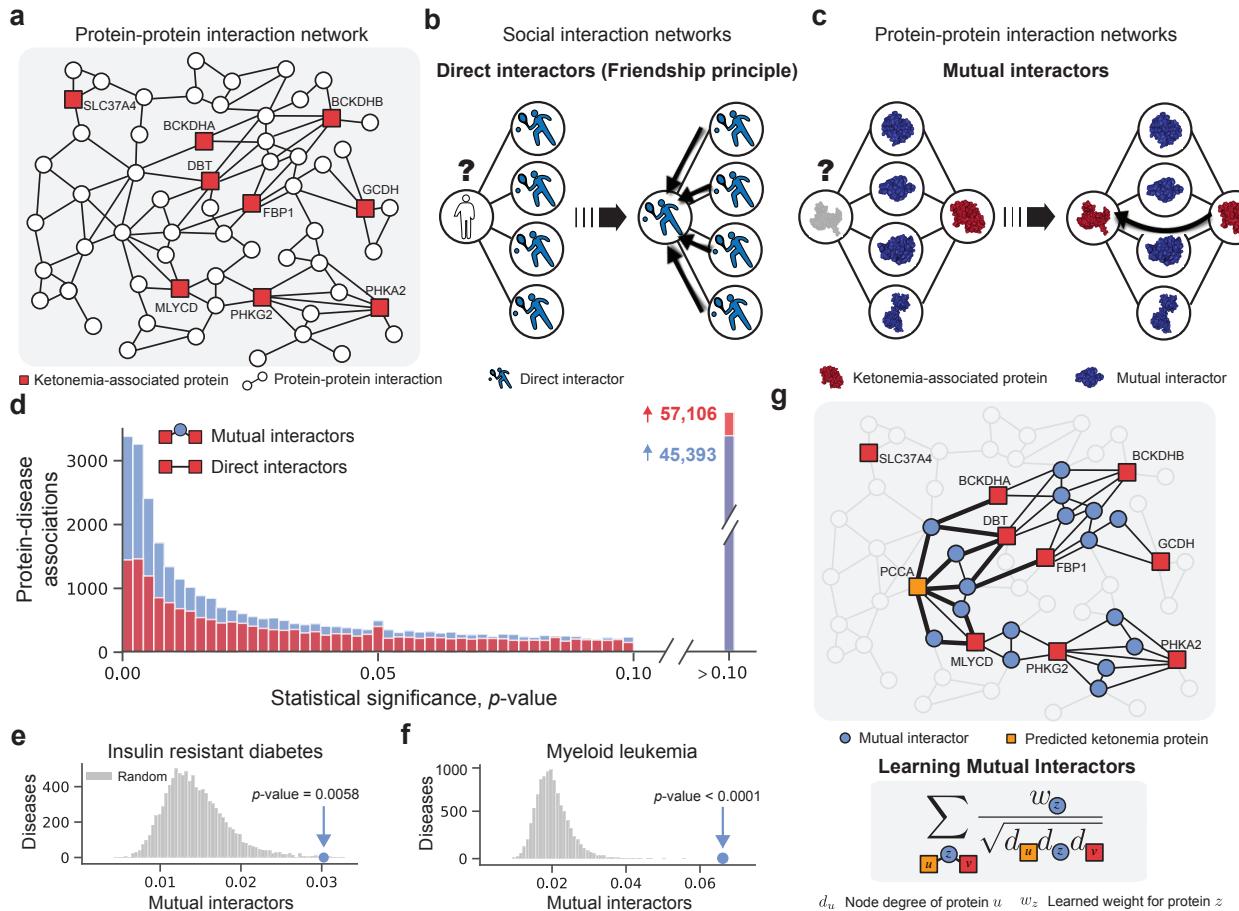


Fig. 1: The mutual interactor principle. (a) The human protein-protein interaction (PPI) network with proteins associated with ketonemia highlighted (in red). (b) Schematic illustration of the friendship principle (*i.e.*, network homophily¹²) in a social network of five individuals. (c) Schematic illustration of the mutual interactor principle in a PPI network. According to the *mutual interactor* principle, the grey protein is likely associated with ketonemia because it interacts with the same proteins as a known ketonemia protein (in red); the two proteins share four mutual interactors (in blue). (d) Comparison of mutual interactors and direct interactors as principles of disease protein connectivity in a human PPI network. For 75,744 disease-protein associations, the statistical significance (*p*-value) of the mutual interactor score (in blue) and the direct interactor score (in red) is computed and plotted for comparison (see Methods). (e-f) We calculate the average mutual interactor score of proteins associated with (e) insulin resistant diabetes and (f) myeloid leukemia (see Methods). The observed mutual interactor scores (in blue) are significantly larger than random expectation (in grey).

Moreover, in 46.5% of disease-protein associations, the protein does not interact directly with any other proteins associated with the same disease. For each disease-protein association, we also compute the degree-normalized count of mutual interactors between the protein and other proteins associated with the disease. We call this the association's *mutual interactor score* (see Methods). In 31.0% of disease-protein associations, this score is significant (permutation test, $P < 0.05$). For other molecular phenotypes, we get similar results: proteins targeted by the same drug have a significant direct interactor score 35.1% of the time and a significant mutual interactor score 67.5% of the time (see Figure 3b).¹⁸ In only 31.0% of the protein-function associations in the Gene Ontology is the direct interactor score significant, compared with 56.7% for the mutual interactor score (see Figure A1a).¹⁹ For biological processes in the Gene Ontology, these fractions are 26.7% and 46.3% for the direct and mutual interactor scores, respectively (see Figure A1b). These results suggest that, in biological networks, there is more empirical evidence for the Mutual Interactor principle than there is for the principle of homophily.

3. Mutual Interactors as a machine learning method for predicting molecular phenotypes

Based on the mutual interactor principle, we develop a machine learning method for inferring associations between proteins and molecular phenotypes. Below, we describe how our method can predict disease-protein associations using the protein-protein interaction network.

In network-based disease protein prediction, the objective is to discover new disease-protein associations by leveraging the network properties of proteins we already know to be involved in the disease. Our method, *Mutual Interactors*, scores candidate disease-protein associations by evaluating the mutual interactors between the candidate protein and other proteins already known to be associated with the disease. Rather than score candidate disease-protein associations according to the raw count of these mutual interactors, our method learns to weight each mutual interactor differently. Intuitively, this makes sense: the significance of a mutual interactor depends on its profile. For example, that two proteins both interact with the same hub-protein is probably less significant than two proteins both interacting with a low-degree signalling receptor. Rather than hard-code which mutual interactors we deem significant, through training on a large set of disease pathways, *Mutual Interactors* learns which proteins often interact with multiple proteins in the same disease pathway. Specifically, *Mutual Interactors* maintains a weight w_z for every protein z in the interactome. This allows *Mutual Interactors* to down-weight spurious mutual interactors when evaluating a candidate disease-protein association.

To further ground our method, we consider its application to a specific disease pathway. Ketonemia is a condition wherein the concentration of ketone bodies in the blood is abnormally high.^{20,21} In Figure 1a, we show the Ketonemia pathway in the human protein-protein interaction network. In red are the proteins known to be associated with Ketonemia, including MLYCD and BCKDHA.^{22,23} We see that Ketonemia-associated proteins rarely interact with one another. In Figure 1g, we show the same network and disease pathway, but now we've highlighted in blue the mutual interactors between Ketonemia-associated proteins. Of all 21,557 proteins in the human protein-protein interaction network, *Mutual Interactors* predicts that PCCA, shown in orange, is the most likely to be associated with Ketonemia. PCCA is a protein involved in the breakdown of fatty acids, a process which produces ketone bodies as a byproduct. PCCA shares mutual interactors with four proteins known to be associated with Ketonemia: BCKDHA, DBT, FBP1, and MLCYD. Further, two of these mutual interactors, MCEE and PCCB, are of very low degree (with 7 and 21 interactions respectively) and are weighted highly by our *Mutual Interactors*. Below, we provide a technical, task-agnostic description of the method.

3.1. Problem Formulation

Though *Mutual Interactors* was motivated by the molecular phenotype prediction problem, it is a general model that can be applied in any setting where we'd like to group nodes on a graph. Suppose we have a graph $G = \{V, E\}$ and a set of node sets $S = \{S_1, S_2, \dots, S_k\}$ where each set S_i is a subset of the full node set $S_i \subseteq V$. Note that the node sets need not be disjoint. For example, G could be a PPI network and each S_i could be the set of proteins associated with a different phenotype. We can split each node set S_i into a set of training nodes $\tilde{S}_i \subset S_i$ and a set of test nodes $S - \tilde{S}_i$. Given \tilde{S}_i and the network G , we're interested in uncovering the full set of nodes S_i . Formally, this means computing a probability $Pr(u \in S | \tilde{S})$ for each node $u \in V$.

3.2. The Mutual Interactors model

The mutual interactors of two nodes u and v are given by the set $M_{u,v} = N(u) \cap N(v)$, where $N(u)$ is the set of u 's one-hop neighbors. For each node $z \in V$, *Mutual Interactors* maintains a weight w_z . As we discussed above, these weights are meant to capture the degree to which each node in the graph acts as a mutual interactor in the node sets of \mathbf{S} . With a weight w_z for every possible mutual interactor in the network, we model the probability that a query node u is in a full node set S given the training set $\tilde{S} \subseteq S$ as

$$Pr(u \in S | \tilde{S}) = \sigma \left(a \left(\sum_{v \in \tilde{S}} \frac{1}{\sqrt{d_v d_u}} \sum_{z \in M_{v,u}} \frac{w_z}{\sqrt{d_z}} \right) + b \right) \quad (1)$$

where d_u is the degree of node u , $\sigma(x) = \frac{1}{1+e^{-x}}$ is the sigmoid function, a is a scale parameter, b is a bias parameter, and w_z is a learned weight for node z . With sparse matrix multiplication we can efficiently compute the probability for every node in the network with respect to a batch of k training sets $\{\tilde{S}_1, \dots, \tilde{S}_k\}$. Let's encode training sets with a binary matrix $\mathbf{X} \in \{0, 1\}^{k \times n}$, where $x_{ij} = 1$ if and only if $j \in \tilde{S}_i$. With \mathbf{X} , we can efficiently compute the probability matrix \mathbf{P} where $P_{ij} = Pr(j \in S_i | \tilde{S}_i)$ with

$$\mathbf{P} = \sigma(a(\mathbf{X}\mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{W}\mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}) + b) \quad (2)$$

where \mathbf{A} is the adjacency matrix, \mathbf{D} is the diagonal degree matrix and \mathbf{W} is a diagonal matrix with the weights w_z on the diagonal.

3.3. Training the Mutual Interactors model

Given a meta-training set of k node sets $\mathbf{S} = \{S_1, \dots, S_k\}$, we can learn the model's weights \mathbf{W} , a , and b that maximize the likelihood of observing the node sets in the meta-training set. During meta-training we simulate node set expansion by splitting each set S_i into a training set \tilde{S}_i encoded by $\mathbf{X} \in \{0, 1\}^{m \times n}$ and a target set $S_i - \tilde{S}_i$ encoded by $\mathbf{Y} \in \{0, 1\}^{m \times n}$. For each epoch, we randomly sample 90% of associations for the training set and use the remaining 10% for the test set. The input associations \mathbf{X} are fed through our model to produce association probabilities \mathbf{P} . We update model weights by minimizing weighted binary cross-entropy loss

$$\ell(\mathbf{X}, \mathbf{Y}) = \sum_{i=1}^m \sum_{j=1}^n -[\alpha_p Y_{ij} \log P_{ij} + (1 - Y_{ij}) \log(1 - P_{ij})] \quad (3)$$

where α_p is the weight given to positive examples. Since there are far more positive examples than negative examples, we set $\alpha_p = \frac{\# \text{ negative examples}}{\# \text{ positive examples}}$.

We can minimize the loss using a gradient-based optimizer. First, we compute the gradient of the loss with respect to model parameters via backpropagation. Then, we use ADAM with a learning rate of 1.0 to update parameters. We train *Mutual Interactors* with weight decay 10^{-5} and a batch size of 200.²⁴ We train for five epochs and use $\frac{1}{9}$ of the training labels as a validation set for early stopping.

4. Predicting disease-associated proteins with *Mutual Interactors*

Mutual Interactors accurately predicts disease-associated proteins. We systematically evaluate our method by simulating disease protein discovery on 1,811 different disease pathways. In ten-fold

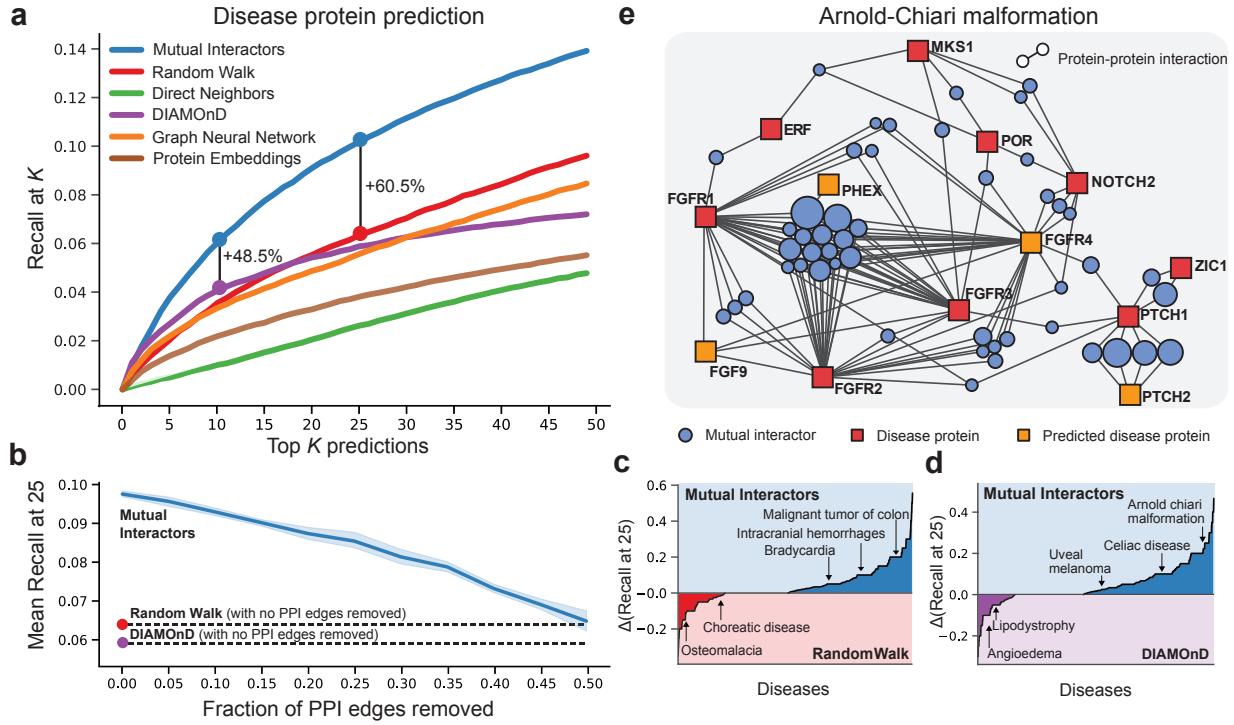


Fig. 2: Uncovering disease proteins with the mutual interactor principle. (a) Overall performance evaluation. The plot shows the fraction of disease proteins recovered within the top k predictions for $k = 1$ to $k = 50$ (recall-at- k). The dotted lines at $k = 10$ and $k = 25$ show the percent-increase in recall over the next best performing method. (b) Effect of data incompleteness on performance. Shown is Mutual Interactors' recall-at-25 as a function of the fraction of protein-protein interactions randomly removed from the network. Dotted lines indicate performance of random walks and DIAMOnD on a full PPI network with no PPIs removed. (c-d) Comparison of Mutual Interactors and baseline methods across diseases. For each disease in our dataset (x-axis), we plot the difference in recall-at-25 (y-axis) between Mutual Interactors and two baseline methods: (c) random walks, (d) DIAMOnD.²⁵ (f) Comparison of the degree-normalized Mutual Interactor weights of drug targets and non-targets. Shown is the distribution of degree-normalized Mutual Interactor weights for 2,212 drug targets¹⁸ (in blue), and, for comparison, the distribution of degree-normalized Mutual Interactor weights for 2,212 random proteins that are not targets of any drug (in grey). (g) Mutual Interactor neighborhood for Arnold-Chiari (AC) malformation. The neighborhood includes known disease proteins (red squares), Mutual Interactors' top predictions (orange squares), and the mutual interactors between them (blue circles). Mutual interactors are sized proportional to their learned Mutual Interactor weight, w_z .

cross-validation, we find that *Mutual Interactors* recovers a larger fraction of held-out proteins than do existing disease protein discovery methods. Specifically, for 10.2% of disease-protein associations our method ranks the held-out protein within the first 25 proteins in the network (recall-at-25 = 0.102). *Mutual Interactors*'s performance represents an improvement of 60.9% in recall-at-25 over the next best performing method, random walks. Other network-based methods of disease protein discovery including DIAMOnD¹⁰ (recall-at-25 = 0.059), random walks²⁶ (recall-at-25 = 0.063), and graph convolutional neural networks²⁵ (recall-at-25 = 0.057) recover considerably fewer disease-protein associations (see Figure 2a,c-d). Moreover, *Mutual Interactors* maintains its advantage over existing methods across disease categories: in all seventeen that we considered *Mutual Interactors*'s mean recall-at-100 exceeds random walks' (see Section F and Figure F1). We also study whether *Mutual Interactors* can generalize to a new disease that is unrelated to the diseases it was trained on. To do so, we train *Mutual Interactors* in the more challenging setting where similar diseases are kept from straddling the train-test divide (see Section E and Figure E1). In this setting, *Mutual Interactors* achieves a recall-at-25 of 0.096, a 50.7% increase in performance over the next best method, random walks. *Mutual Interactors* can naturally be extended to incorporate other sources of protein data.²⁷ In Section app:parametric, we describe a parametric *Mutual Interactors* model that incorporates functional profiles from the Gene Ontology when evaluating mutual interactors. Instead of learning

a weight w_z for every protein z , this model learns one scalar-valued function mapping gene ontology embeddings to mutual interactor weights. We show that parametric *Mutual Interactors* performs on par with the original *Mutual Interactors* model, outperforming baseline methods by 45.5% in recall-at-25 (see Figure G1).

The experimental data we use to construct molecular interaction networks is often incomplete or noisy: it is estimated that state-of-the-art interactomes are missing 80% of all the interactions in human cells.¹⁶ In light of this, we test if our method is tolerant of data incomplete networks. We find that *Mutual Interactors* exhibits stable performance up to the removal of 50% of known PPI interactions. *Mutual Interactors*'s performance with only half of all known interactions exceeds the performance of existing methods that use all known interactions (Figure 2b).

We perform an ablation study to assess the benefits of meta-learning mutual interactor weights w_z (See Figure A8). In the study, we compare our model with *Constant Mutual Interactors* where $w_z = 1 \forall k$. On tasks for which we have a large dataset of phenotypes (i.e. disease protein prediction and molecular function prediction in humans), meta-learning w_z improves performance by up to 16.6% in recall-at-25. However, on tasks for which data is scarce (i.e. drug-target prediction and non-human molecular function prediction) learning w_z does not provide a significant benefit. For these tasks, we report performance on *constant Mutual Interactors* where $w_z = 1 \forall k$.

Learned weights provide insight into the function and druggability of mutual interactors. Next we analyze the mutual interactor weights learned by our method. Recall that *Mutual Interactors* learns a weight w_z for every protein z in the interactome. This allows *Mutual Interactors* to down-weight spurious mutual interactors when evaluating a candidate disease-protein association. Here, we study what insights into biological mechanisms these weights reveal. We find that normalized *Mutual Interactors* weight $\frac{w_z}{\sqrt{d_z}}$ is correlated with neither degree ($r = 0.0359$) nor triangle clustering coefficient ($r = 0.0127$) (see Figure A9). However, we do find that proteins with high weight are often involved in cell-cell signaling. We perform a functional enrichment analysis on the 75 proteins with the highest normalized weight $\frac{w_z}{\sqrt{d_z}}$. Of the fifteen functional classes most enriched in these proteins, ten including *signaling receptor activity* and *cell surface receptor signaling pathway* are directly related to transmembrane signaling and the other five including *plasma membrane part* are tangentially related to signaling (see Figure A6). Further, we find that highly-weighted proteins are often targeted by drugs. Among the 500 proteins with the highest degree-normalized weight, 33.6% are targeted by a drug in the DrugBank database¹⁸ (See Figure 2f). By contrast only 10.9% of proteins in the wider protein-protein interaction network are targeted by those drugs. This represents a significant increase ($p \leq 6.43 \times 10^{-24}$, Kolmogorov-Smirnov test). Indeed, although no drug-target interaction data was used, training our method to predict disease proteins gives us biologically meaningful insights into which proteins are druggable.

5. Identifying drug targets with *Mutual Interactors*

Our method can also be used to identify drug targets. Recall that mutual interactors between proteins targeted by the same drug are statistically overrepresented in the protein-protein interaction network (Figure 3a). Like with disease-protein associations, *Mutual Interactors* can score candidate drug-target interactions by evaluating the mutual interactors between the candidate target protein and other proteins already known to be targeted by the drug (see *Methods* for a technical description

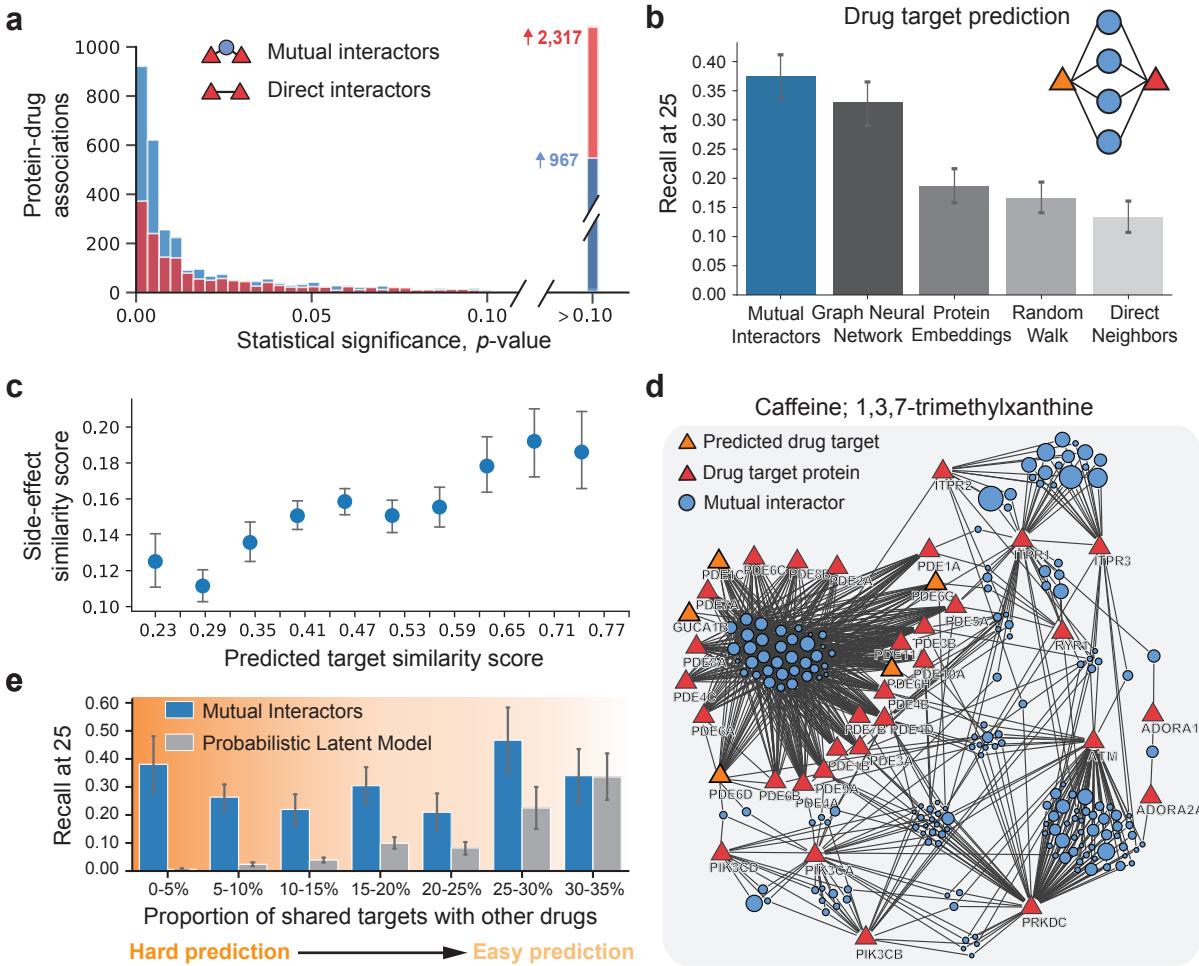


Fig. 3: Identifying drug targets using the principle of mutual interactors and the validity of the principle in non-biological networks. (a) Comparison of Mutual Interactors (in blue) and direct interactors (in red) as principles of drug-target connectivity in a human PPI network. For 4,403 drug-target associations,²⁸ the statistical significance (p -value) of the mutual interactor score (in blue) and the direct interactor score (in red) is computed and plotted for comparison (see Methods). (b) Drug target identification. Shown is mean recall-at-25 across 190 drugs. (c) The side-effect similarity of drugs²⁹ (y-axis) is linearly related to the similarity of Mutual Interactors' predictions for those drugs (x-axis). (d) Mutual Interactors' neighborhood for proteins targeted by Caffeine. The neighborhood includes caffeine-targeted proteins (red triangles), Mutual Interactors' top predictions for novel caffeine targets (orange triangles), and the mutual interactors between them (blue circles). Mutual interactors are sized proportional to their learned Mutual Interactors weight, w_z (see Methods). (e) The fraction of a drug's targets recovered within the top 25 predictions (recall-at-25) vs. the maximum Jaccard similarity between the drug's targets and targets of other drugs in the training set used for machine learning. Bars indicate average recall-at-25 in each bucket.

of the approach). When we simulate drug-target identification with ten-fold cross validation on the drugs and targets in the DrugBank database,¹⁸ we find that our method outperforms existing network-based methods of drug-target identification (recall-at-25=0.374), including graph neural networks (recall-at-25=0.329) and random walks (recall-at-25=0.166). We also compare *Mutual Interactors* with probabilistic non-negative matrix factorization (NMF).^{30–32} On aggregate, our method's performance is comparable to NMF's. However, on the hardest examples, drugs that share few targets with the drugs in the training set, our method (recall-at-25=0.381) significantly outperforms NMF (recall-at-25=0.006) (see Figure 3e). Further, our method provides insight into the side-effects caused by off-target binding. For each drug in DrugBank, we use *Mutual Interactors* to identify potential protein targets that are not already known targets of the drug. Pairs of drugs for which our method makes similar target predictions tend to have similar side effects, an observation supported by prior evidence^{33–36} (Figure 3c).

6. Predicting molecular function across species and molecular networks

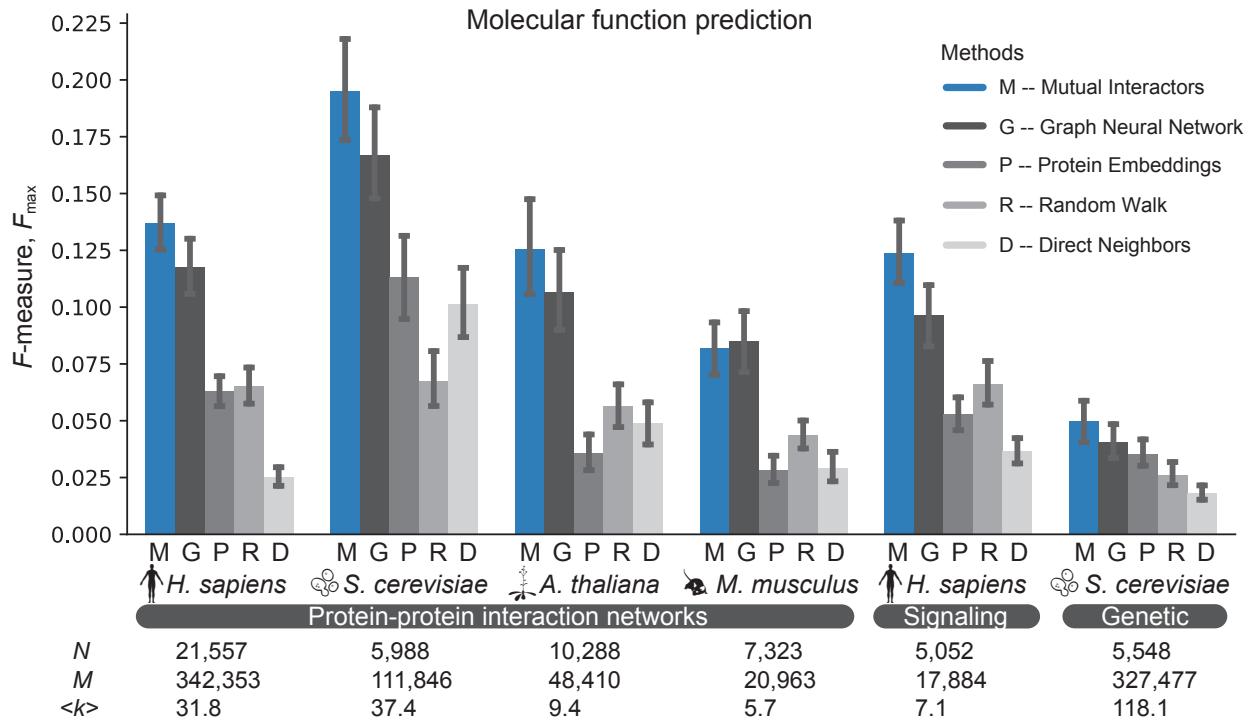


Fig. 4: **Predicting protein functions across species and molecular networks using mutual interactors.** Overall protein function prediction performance across four species and six molecular networks. We predict Molecular Function Ontology³⁷ terms using PPI, signaling, and genetic interaction networks for human, yeast *S. cerevisiae*, mouse *M. musculus*, and thale cress *A. thaliana*. We show average maximum F -measure.³⁸ A perfect predictor would be characterized by $F_{max} = 1$. Confidence intervals (95%) were determined using bootstrapping with $n = 1,000$ iterations. N – number of nodes, M – number of edges, $\langle k \rangle$ – average node degree.

Molecules associated with the same molecular function (e.g., RNA polymerase I activity) or involved in the same biological process (e.g., nucleosome mobilization) tend to share mutual interactors in molecular networks of various type and species (see Figure 4a-b). For example, the eleven proteins involved in the formation of the secondary messenger cAMP (cyclase activity, GO:0009975) do not interact directly with one another in the protein-protein interaction network, but almost all of them interact with the same group of twenty-five mutual interactors (see Figure A3). Using the Mutual Interactor principle, we can predict the molecular functions and biological processes of molecules. Via ten-fold cross validation, we compare *Mutual Interactors* to existing methods of molecular function prediction, including Graph Neural Networks³⁹ and Random Walks.²⁶ Across all four species (*H. sapiens*, *S. cerevisiae*, *A. thaliana*, *M. musculus*) and in three different molecular networks (protein-protein interaction, signaling, and genetic interaction), we find that *Mutual Interactors* is the strongest predictor of both molecular function (see Figure 3c) and biological process (see Figure 3d).

7. Conclusion

This work demonstrates the importance of rooting biomedical network science methods in principles that are empirically validated in biological data, rather than borrowed from other domains. This

need for more domain-specific methodology in biomedical network science is also demonstrated by Kovács *et al.*, who find that social network principles do not apply for link prediction in PPI networks.⁴⁰ This study complements these findings: with experiments across three different kinds of molecular networks (protein-protein interaction, signaling and genetic interaction), and four species (*H. sapiens*, *S. cerevisiae*, *A. thaliana*, *M. musculus*) we show that a method designed specifically for biological data can better predict disease-protein associations, drug-target interactions and molecular function than can general methods of greater complexity. The power of *Mutual Interactors* to predict molecular phenotypes lies not in its algorithmic complexity—it outperforms far more involved methods—but rather in the simple, yet fundamental, principle that underpins it. Motivated by our findings that molecules with similar phenotypes tend to share mutual interactors, we formalize the Mutual Interactor principle mathematically with machine learning. *Mutual Interactors* is fast, easy to implement, and robust to incomplete network data—its foundational formulation makes it ripe for extension to new domains and problems.

Supplementary Material and Code. Supplementary materials are available online at .Code for reproducing the results are available at Code available at <https://github.com/seyuboglu/milieu>

References

1. E. E. Schadt, Molecular networks as sensors and drivers of common human diseases, *Nature* **461**, 218 (September 2009).
2. P. Bork *et al.*, Protein interaction networks from yeast to human, *Current Opinion in Structural Biology* **14**, 292 (June 2004).
3. K. Lage *et al.*, A human phenome-interactome network of protein complexes implicated in genetic disorders, *Nature Biotechnology* **25**, p. 309 (2007).
4. A.-L. Barabási *et al.*, Network medicine: A network-based approach to human disease, *Nature Reviews Genetics* **12**, 56 (January 2011).
5. L. D. Wood *et al.*, The genomic landscapes of human breast and colorectal cancers, *Science* **318**, 1108 (2007).
6. J. Lim *et al.*, A protein–protein interaction network for human inherited ataxias and disorders of purkinje cell degeneration, *Cell* **125**, 801 (2006).
7. J. Chen *et al.*, Detecting functional modules in the yeast protein–protein interaction network, *Bioinformatics* **22**, 2283 (September 2006).
8. X. Wu *et al.*, Network-based global inference of human disease genes, *Molecular Systems Biology* **4**, p. 189 (2008).
9. W. Peng *et al.*, Improving protein function prediction using domain and protein complexes in PPI networks, *BMC Systems Biology* **8**, p. 35 (March 2014).
10. S. D. Ghiassian *et al.*, A DIseAse MOdule Detection (DIAMOnD) algorithm derived from a systematic analysis of connectivity patterns of disease proteins in the human interactome, *PLoS Computational Biology* **11**, p. e1004120 (2015).
11. M. Agrawal *et al.*, Large-scale analysis of disease pathways in the human interactome, *Pacific Symposium on Biocomputing* **1**, 111 (2018).
12. M. McPherson *et al.*, Birds of a feather: Homophily in social networks, *Annual Review of Sociology* **27**, 415 (2001).
13. K. Luck *et al.*, A reference map of the human binary protein interactome, *Nature* **580**, 402 (April 2020).
14. R. Oughtred *et al.*, The BioGRID database: A comprehensive biomedical resource of curated protein, genetic, and chemical interactions, *Protein Science* **30**, 187 (2021).

15. D. Szklarczyk *et al.*, The string database in 2021: customizable protein–protein networks, and functional characterization of user-uploaded gene/measurement sets, *Nucleic Acids Research* **49**, D605 (2021).
16. J. Menche *et al.*, Uncovering disease-disease relationships through the incomplete interactome, *Science* **347**, p. 1257601 (February 2015).
17. J. Piñero *et al.*, The disgenet knowledge platform for disease genomics: 2019 update, *Nucleic acids research* **48**, D845 (2020).
18. D. S. Wishart *et al.*, DrugBank 5.0: a major update to the DrugBank database for 2018, *Nucleic Acids Research* **46**, D1074 (January 2018).
19. T. G. Ontology, The Gene Ontology Resource: 20 years and still GOing strong, *Nucleic Acids Research* **47**, D330 (January 2019).
20. K. Ennis *et al.*, Hyperglycemia accentuates and ketonemia attenuates hypoglycemia-induced neuronal injury in the developing rat brain, *Pediatric Research* **77**, 84 (2015).
21. O. Arisaka *et al.*, Iron, ketone bodies, and brain development, *The Journal of Pediatrics* **222**, 262 (2020).
22. X. Li *et al.*, Eleven novel mutations of the BCKDHA, BCKDHB and DBT genes associated with maple syrup urine disease in the Chinese population: report on eight cases, *European Journal of Medical Genetics* **58**, 617 (2015).
23. S. H. Lee *et al.*, A Korean child diagnosed with malonic aciduria harboring a novel start codon mutation following presentation with dilated cardiomyopathy, *Molecular Genetics & Genomic Medicine* **8**, p. e1379 (2020).
24. D. P. Kingma *et al.*, Adam: A Method for Stochastic Optimization, *arXiv:1412.6980 [cs]* (December 2014), arXiv: 1412.6980.
25. T. N. Kipf *et al.*, Semi-supervised classification with graph convolutional networks, *International Conference on Learning Representations* (2017).
26. *Analysis of protein-protein interaction networks using random walks* 2005.
27. M. Zitnik *et al.*, Machine learning for integrating data in biology and medicine: Principles, practice, and opportunities, *Information Fusion* **50**, 71 (2019).
28. D. S. Wishart *et al.*, Drugbank 5.0: a major update to the drugbank database for 2018, *Nucleic Acids Research* **46**, D1074 (2017).
29. N. P. Tatonetti *et al.*, Data-driven prediction of drug effects and interactions, *Science Translational Medicine* **4**, 125ra31 (2012).
30. *Probabilistic matrix factorization* 2008.
31. D. D. Lee *et al.*, Learning the parts of objects by non-negative matrix factorization, *Nature* **401**, 788 (October 1999).
32. *Structured non-negative matrix factorization with sparsity patterns* October 2008.
33. M. Campillos *et al.*, Drug Target Identification Using Side-Effect Similarity, *Science* **321**, 263 (July 2008).
34. R. Santos *et al.*, A comprehensive map of molecular drug targets, *Nature Reviews Drug Discovery* **16**, 19 (2017).
35. L. H. Calabrese *et al.*, Rheumatic immune-related adverse events from cancer immunotherapy, *Nature Reviews Rheumatology* **14**, 569 (2018).
36. F. Cheng *et al.*, Network-based prediction of drug combinations, *Nature Communications* **10**, 1 (2019).
37. M. Ashburner *et al.*, Gene Ontology: tool for the unification of biology, *Nature Genetics* **25**, p. 25 (2000).
38. P. Radivojac *et al.*, A large-scale evaluation of computational protein function prediction, *Nature Methods* **10**, p. 221 (2013).
39. T. N. Kipf *et al.*, Semi-Supervised Classification with Graph Convolutional Networks (September 2016).
40. I. A. Kovács *et al.*, Network-based prediction of protein interactions, *Nature communications* **10**, 1 (2019).
41. UniProt Consortium, Activities at the Universal Protein Resource (UniProt), *Nucleic Acids Research* **42**, D191 (January 2014).

42. L. Cowen *et al.*, Network propagation: a universal amplifier of genetic associations, *Nature Reviews Genetics* **18**, p. 551 (2017).
43. A. Grover *et al.*, node2vec: Scalable Feature Learning for Networks (July 2016).
44. A.-L. Barabási *et al.*, Network medicine: a network-based approach to human disease., *Nature reviews. Genetics* **12**, 56 (January 2011).
45. S. Navlakha *et al.*, The power of protein interaction networks for associating genes with diseases, *Bioinformatics* **26**, 1057 (2010).
46. S. Kohler *et al.*, Walking the Interactome for Prioritization of Candidate Disease Genes, *The American Journal of Human Genetics* **82**, 949 (April 2008).
47. S. D. Ghiassian *et al.*, A DIseAse MOdule Detection (DIAMOnD) Algorithm Derived from a Systematic Analysis of Connectivity Patterns of Disease Proteins in the Human Interactome, *PLOS Computational Biology* **11**, e1004120 (April 2015).
48. H. Laurberg *et al.*, Theorems on Positive Data: On the Uniqueness of NMF, *Computational Intelligence and Neuroscience* **2008** (2008).
49. M. Zitnik *et al.*, NIMFA: A Python Library for Nonnegative Matrix Factorization, *Journal of Machine Learning Research* , p. 5 (2012).
50. E. Guney *et al.*, Network-based in silico drug efficacy screening, *Nature Communications* **7**, 10331 (February 2016).
51. D. V. Klopfenstein *et al.*, GOATOOLS: A Python library for Gene Ontology analyses, *Scientific Reports* **8**, p. 10872 (July 2018).
52. V. Matys *et al.*, TRANSFAC ® : transcriptional regulation, from patterns to profiles, *Nucleic Acids Research* **31**, 374 (January 2003).
53. A. Ceol *et al.*, MINT, the molecular interaction database: 2009 update, *Nucleic Acids Research* **38**, D532 (January 2010).
54. B. Aranda *et al.*, The IntAct molecular interaction database in 2010, *Nucleic Acids Research* **38**, D525 (January 2010).
55. M. Giurgiu *et al.*, Corum: the comprehensive resource of mammalian protein complexes—2019, *Nucleic acids research* **47**, D559 (2019).
56. M. Costanzo *et al.*, The Genetic Landscape of a Cell, *Science* **327**, 425 (January 2010).
57. M. Costanzo *et al.*, A global genetic interaction network maps a wiring diagram of cellular function, *Science* **353**, p. aaf1420 (September 2016).
58. D. Türei *et al.*, OmniPath: guidelines and gateway for literature-curated signaling pathway resources, *Nature Methods* **13**, 966 (December 2016).
59. S. Choobdar *et al.*, Assessment of network module identification across complex diseases, *Nature Methods* **16**, 843 (2019).
60. J. Piñero *et al.*, DisGeNET: a discovery platform for the dynamical exploration of human diseases and their genes., *Database : the journal of biological databases and curation* **2015**, bav028 (2015).
61. A. P. Davis *et al.*, The Comparative Toxicogenomics Database: update 2013, *Nucleic Acids Research* **41**, D1104 (January 2013).
62. W. A. Kibbe *et al.*, Disease Ontology 2015 update: an expanded and updated database of human diseases for linking biomedical knowledge through disease data., *Nucleic acids research* **43**, D1071 (January 2015).
63. A. Chatr-Aryamontri *et al.*, The BioGRID interaction database: 2015 update., *Nucleic acids research* **43**, D470 (January 2015).
64. D. Szklarczyk *et al.*, STRING v10: protein-protein interaction networks, integrated over the tree of life., *Nucleic acids research* **43**, D447 (January 2015).

Appendix A. Extended Figures

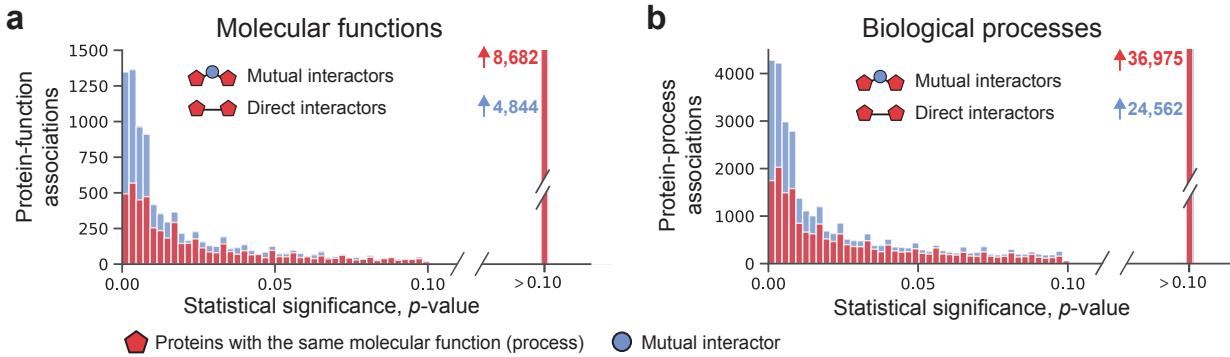


Fig. A1: Network connectivity of protein functions and biological process. (a-b) Comparison of mutual interactors and direct interactors as principles of PPI connectivity for human proteins associated with the same (a) molecular function³⁷ and (b) biological process.³⁷ For each of 13,983 molecular function associations and 55,884 biological process associations, the statistical significance of the mutual interactor score (in blue) is calculated using a non-parametric permutation test. The mutual interactor score of a protein-function association is the degree-normalized count of mutual interactors between the protein and other proteins associated with the same function (see Methods). Also calculated for each protein-function association is the statistical significance of a protein's direct interactions with other proteins associated with the function (in red). We plot the distribution of p -values < 0.10 over all 13,983 protein-function associations. Results show that proteins with the same molecular function (or, proteins involved in the same biological process) interact with the same proteins (*i.e.*, share mutual interactors) significantly more often than they directly interact with each other.

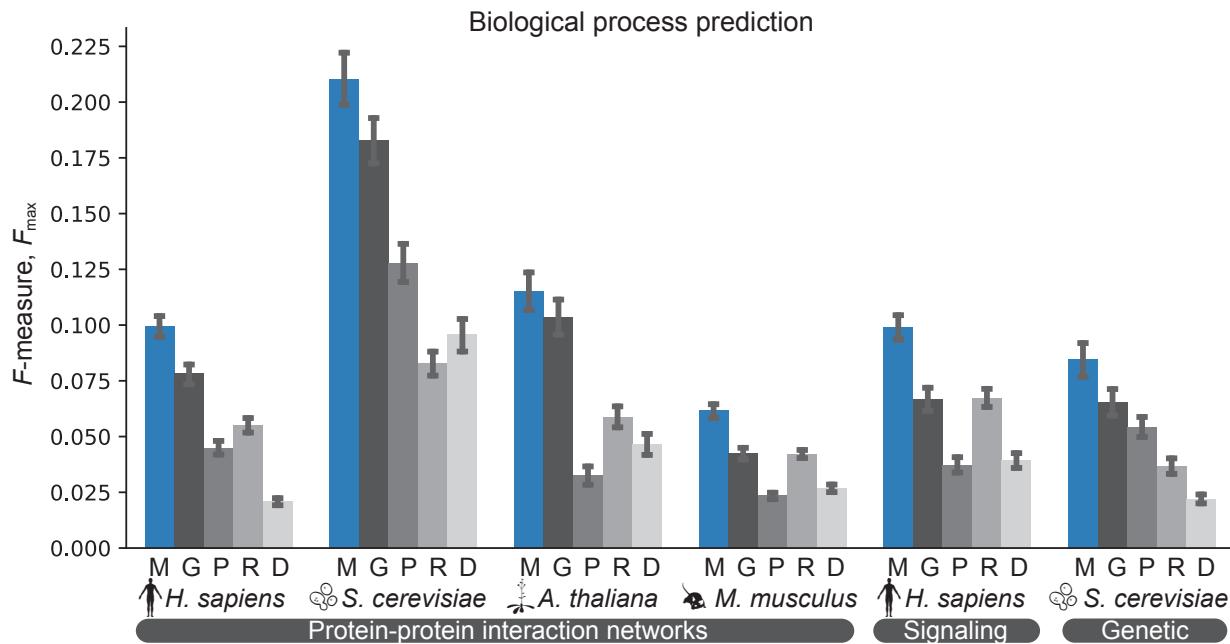


Fig. A2: Predicting biological process across species and molecular networks using mutual interactors. Overall biological process prediction performance across four species and six molecular networks. We predict Biological Process Ontology³⁷ terms using PPI, signaling, and genetic interaction networks for human, yeast *S. cerevisiae*, mouse *M. musculus*, and thale cress *A. thaliana*. We show average maximum F -measure.³⁸ A perfect predictor would be characterized by $F_{max} = 1$. Confidence intervals (95%) were determined using bootstrapping with $n = 1,000$ iterations. N – number of nodes, M – number of edges, $\langle k \rangle$ – average node degree.

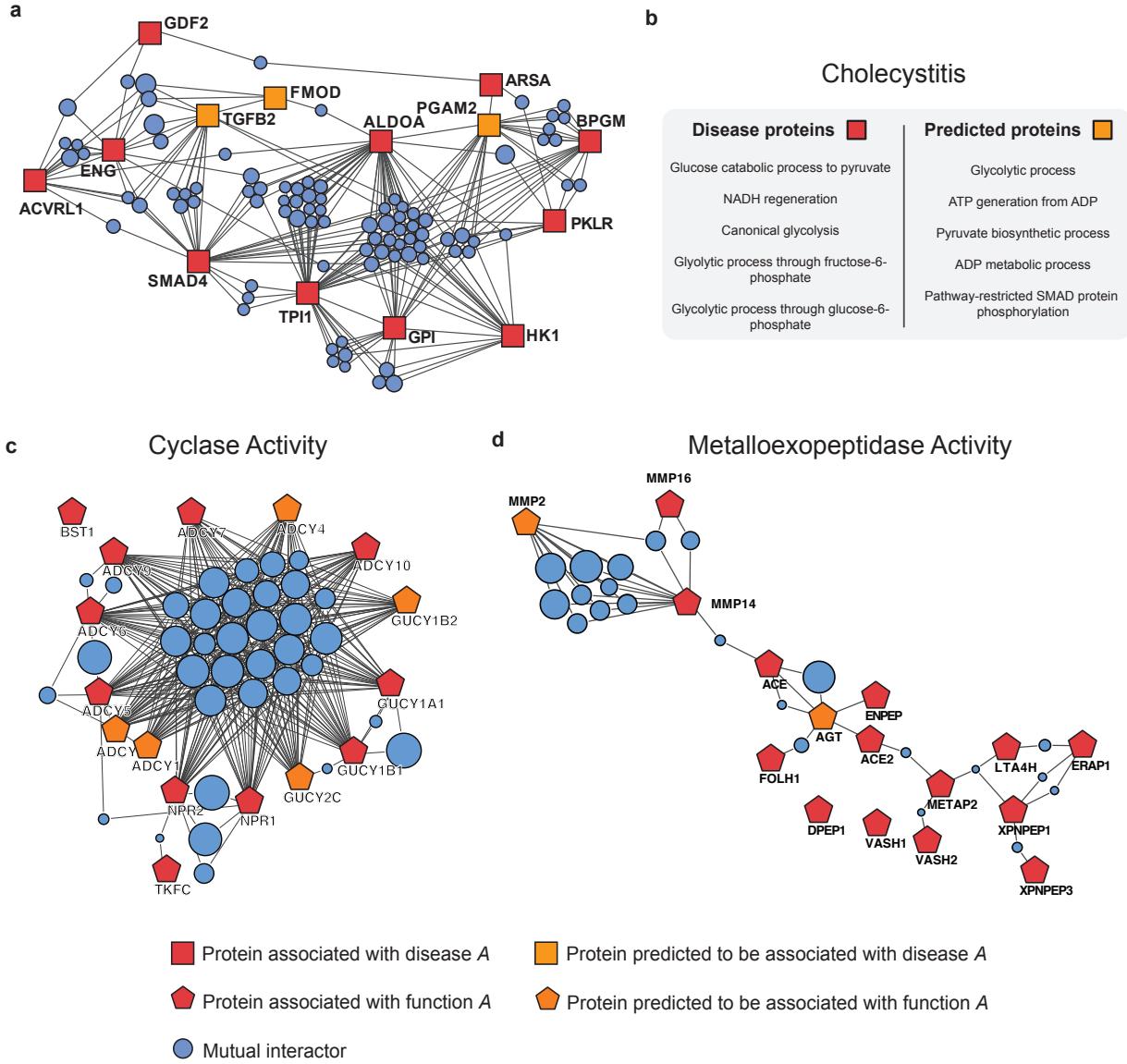


Fig. A3: Mutual interactor neighborhoods in a human PPI network for proteins associated with (a) Cholecystitis, (b) cyclase activity and (c) metalloexopeptidase activity. (a) The Cholecystitis neighborhood includes known Cholecystitis proteins (red squares), *Mutual Interactors'* top predictions (orange squares), and the mutual interactors between them (blue circles). Listed in the table are the cellular functions most significantly enriched in known Cholecystitis proteins and *Mutual Interactors'* predictions. (b-c) The cyclase activity and metalloexopeptidase activity neighborhoods include proteins associated with the function (red pentagons), *Mutual Interactors'* top predictions (orange pentagons) and the mutual interactors between them (blue circles). Mutual interactors are sized proportional to their learned Mutual Interactor weight, w_z .

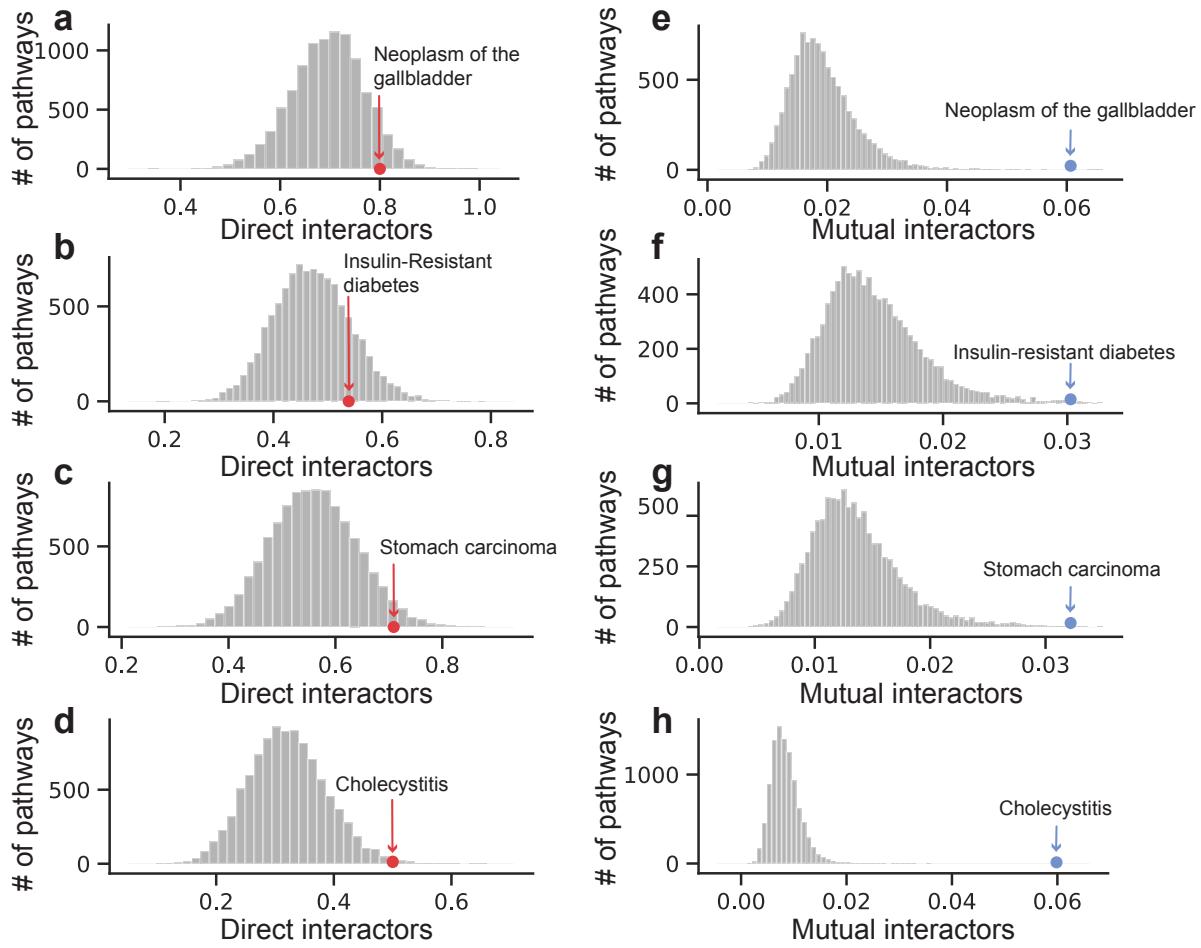


Fig. A4: (a-d) For four diseases (a) neoplasm of the gallbladder, (b) insulin-resistant diabetes, (c) stomach carcinoma, and (d) cholecystitis we calculate the direct interactor score of the disease's proteins. To generate a null distribution, the same is done for 10,000 random protein sets of matching size and degree distribution (see Methods). Shown are the direct interactor scores for each of the three diseases (red) and the null distribution (grey). (e-h) For the same four diseases (e) neoplasm of the gallbladder, (f) insulin-resistant diabetes, (g) stomach carcinoma, and (h) cholecystitis we calculate the mutual interactor score of the disease's proteins and generate a null distribution. The observed mutual interactor scores (in blue) are significantly larger than random expectation (in grey).

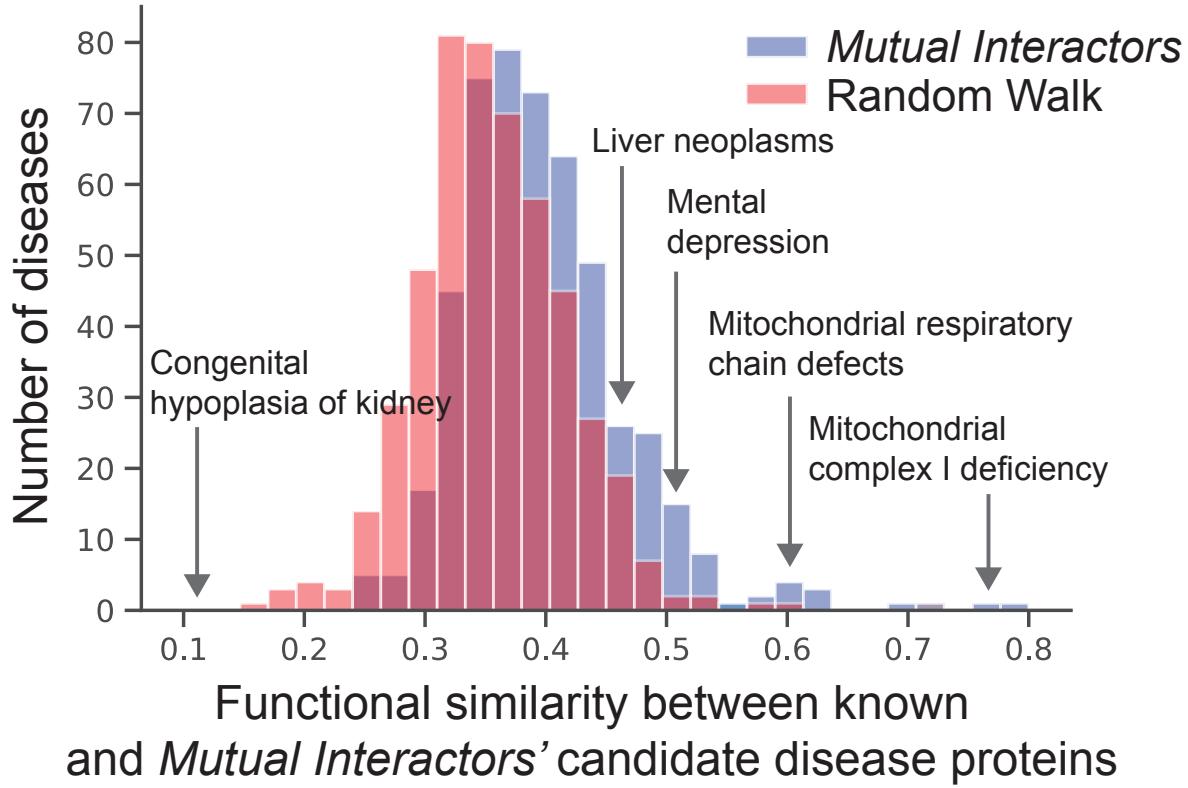


Fig. A5: Functional similarity between known disease proteins and predicted disease proteins. For each disease, we rank cellular functions by their enrichment in the set of proteins S associated with the disease. A cellular function's enrichment in a set of proteins is given by the probability that the function would be associated with at least as many proteins in the set as it actually is. We compute this p -value with Fisher's exact test and consider cellular functions from the Gene Ontology.⁴¹ We repeat this process for the top $|S|$ candidate proteins from each method, producing a ranking of cellular functions for each. We then compute the Spearman rank-order correlation between these rankings to quantify the functional similarity of the disease pathway and predicted proteins. In the histogram, the x -axis is the Spearman rank-order correlation and the y -axis indicates the number of diseases with that degree of similarity. Similarity distribution for *Mutual Interactors* is shown in blue and for Random Walks⁴² in red.

Appendix B. Extended Methods

Appendix B.1. Molecular phenotype prediction and experimental setup

Molecular phenotype prediction can be framed as an information retrieval task on the nodes of a molecular network: given a set \tilde{S} of nodes known to be associated with a phenotype, recover the full set S of nodes associated with that phenotype. Computational methods of molecular phenotype prediction accept known associations \tilde{S} as input and output an association probability for each node u in the network $Pr(u \in S|\tilde{S})$.

Node-wise cross validation. We simulate molecular phenotype prediction using ten-fold cross validation. For each phenotype, we randomly split its associated nodes into ten equally-sized folds. We feed nine of the folds to a molecular phenotype prediction method and record how the method ranks the nodes of the last fold. We repeat this method ten times, holding out a different fold each time.

To evaluate method performance, we compute recall-at- k , the fraction of held-out nodes ranked

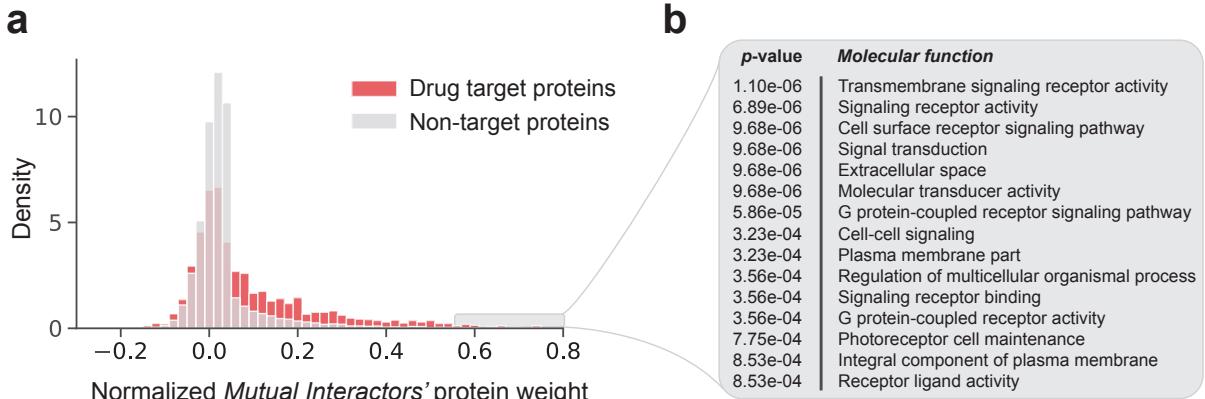


Fig. A6: (a) Comparison of degree-normalized mutual interactor weights of drug targets and non-targets. The mutual interactor weight of protein z is the weight w_z that *Mutual Interactors* learns for protein z (see Methods and Figure 1). Shown is the distribution of degree-normalized mutual interactor weights for 2,212 drug targets¹⁸ (in red) and proteins that are not targeted by any drug (in grey). (b) List of the fifteen molecular functions most enriched in the 75 proteins with the highest degree-normalized mutual interactor weight. A molecular function's enrichment in a set of proteins is given by the probability that the function would be associated with at least as many proteins in the set as it actually is. We compute this p -value with Fisher's exact test and consider molecular functions from the Gene Ontology.⁴¹ Notably, proteins with highest weights w_z are enriched in molecular transducer activity ($p = 9.68e-6$), transmembrane signaling receptors ($p = 1.10e-6$), signal transduction ($p = 9.68e-6$), and G protein-coupled receptors ($p = 5.86e-5$).

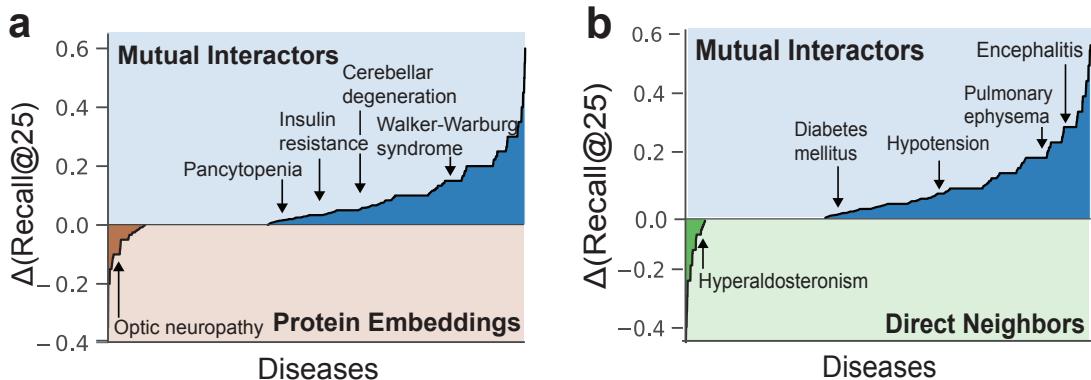


Fig. A7: Comparison of *Mutual Interactors* and baseline methods on the task of disease protein discovery. For each disease, the baseline's recall-at-25 is subtracted from *Mutual Interactors'*. The diseases are plotted in order of increasing difference and a few diseases are annotated. Shown are results for two methods: (a) Protein Embeddings⁴³ and (b) Direct Neighbors.⁴⁴

among the first k nodes in the network. Computational methods of molecular phenotype prediction discovery are useful when they rank associated nodes within the first few nodes in the network, so recall-at- k for some small k (e.g. $k = 10$ or $k = 25$) is an apt evaluation metric. We also compute maximum F-measure, the maximum F1-score achieved by a method across all prediction thresholds.

Phenotype-wise cross validation. *Mutual Interactors* is trained on a large set of phenotypes and subsequently applied to predict associations in other, unseen phenotypes. We wrap node-wise 10-fold cross validation in phenotype-wise 10-fold cross validation. Specifically, we randomly split the phenotypes in our dataset into ten equal-sized folds. We train *Mutual Interactors* with the phenotypes in nine of the folds, and evaluate performance on the phenotypes in the held-out fold using node-wise cross validation described above. By repeating this process ten times and holding out a different fold each time, we can evaluate *Mutual Interactors* on every disease in our dataset.

We also present results for a more challenging setting where similar phenotypes are placed in

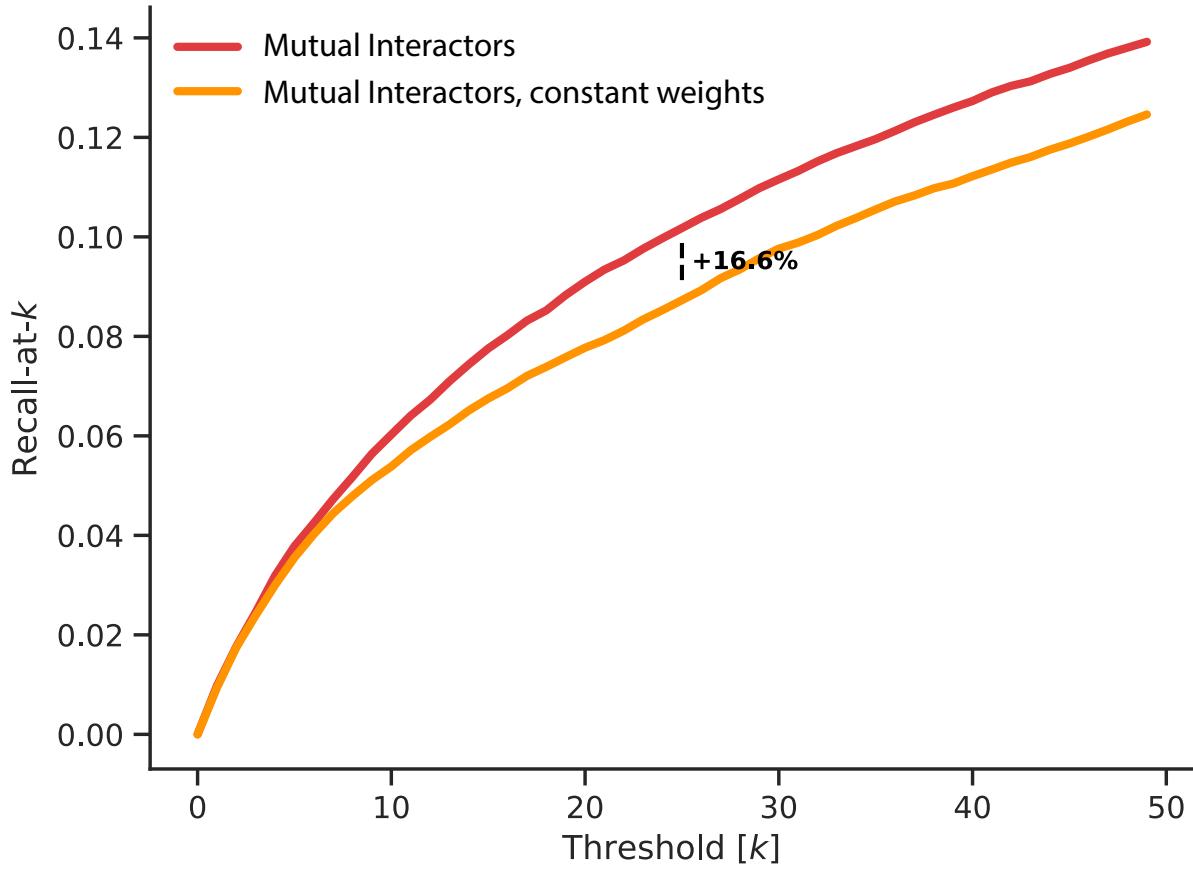


Fig. A8: Effect of learned mutual interactor weights on disease protein prediction performance. Comparison of *Mutual Interactors* with learned weights and *Mutual Interactors* with uniform weights, $w_k = 1 \quad \forall k$. We plot average recall-at- k across all diseases for $k = 0$ to $k = 50$. We include the percent-increase in recall-at-25. *Mutual Interactors* with learned weights achieves a recall-at-25 16.6% higher than *Mutual Interactors* with uniform weights.

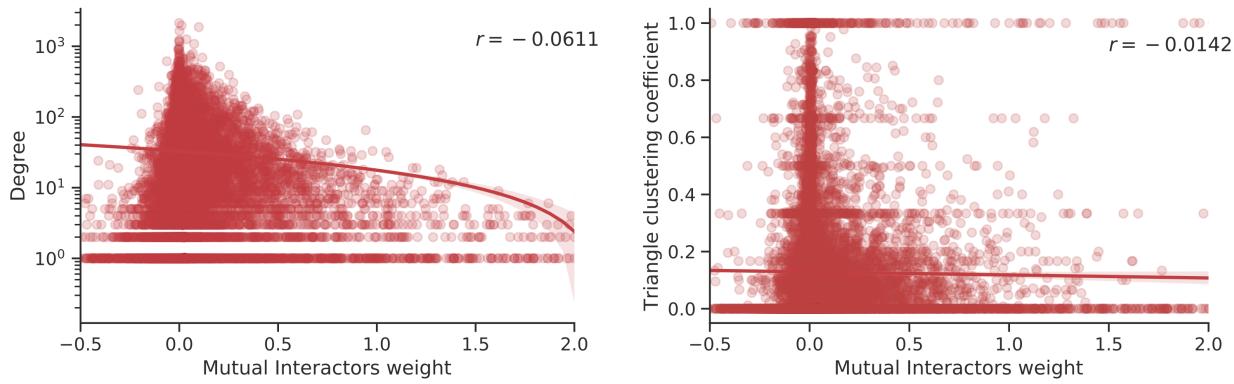


Fig. A9: Degree-normalized *Mutual Interactors* weight $\frac{w_z}{\sqrt{d_z}}$ plotted against two key properties of the human PPI network: the network degree and the triangle clustering coefficient. We report the Pearson correlation coefficient, r . Degree-normalized *Mutual Interactors* weight is largely uncorrelated with these two network properties.

the same folds (See E).

Appendix B.2. Computational methods for predicting molecular phenotypes

On all tasks, *Mutual Interactors*'s performance is evaluated against four network-based methods of molecular phenotype prediction. Additionally, we evaluate *Mutual Interactors* against two task-specific methods: DIAMOnD for disease protein prediction and probabilistic latent models for drug-target interaction prediction.

1) Random walk. Random walks can be used to expand a set of nodes in a graph.²⁶ By simulating a random walker on the graph, we can compute the relevance of each node in the graph with respect to the set. Several previous studies have found random walks to be the most powerful network-based method of disease protein discovery.^{11,45,46}

Formally, given a set of known phenotype associations \tilde{S} , we can model the probability of association $Pr(u \in S|\tilde{S})$ with the stationary distribution \mathbf{p} of a random walk with restart. The initial distribution \mathbf{p}^0 is seeded by known disease associations \tilde{S} such that all of the probability is distributed equally among the proteins in \tilde{S} . One step of the random walk is given by

$$\mathbf{p}^{t+1} = \alpha \hat{\mathbf{A}} \mathbf{p}^t + (1 - \alpha) \mathbf{p}^0 \quad (\text{B.1})$$

where α is the damping factor and $\hat{\mathbf{A}} = \mathbf{AD}^{-1}$ is the column-normalized adjacency matrix of the network. The stationary distribution \mathbf{p} is computed via power iteration until $\frac{1}{n} \|\mathbf{p}^{(t+1)} - \mathbf{p}^{(t)}\|_1 < 10^{-6}$ or we reach 100 iterations. We find that random walks perform best with a damping factor of $\alpha = 0.25$, which is consistent with previous studies.^{11,45}

2) Graph Neural Networks. For each phenotype and its set of known associations \tilde{S} , we train a graph convolutional network (GCN) to predict the full association set S . We train our GCNs in a semi-supervised, binary setting: each protein in \tilde{S} is labeled positive and an equal number of randomly-sampled proteins are labeled negative. On each epoch we resample the negative proteins.

The GCN's initial embedding $\mathbf{H}^{(0)}$ is set to the identity matrix. With each graph convolutional layer we apply the update rule

$$\mathbf{H}^{(l+1)} = \text{ReLU}(\tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}} \mathbf{H}^{(l)} \mathbf{W}^{(l)}) \quad (\text{B.2})$$

where $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ is the adjacency matrix with added self-loops, $\tilde{\mathbf{D}}$ is the diagonal degree matrix for $\tilde{\mathbf{A}}$, and $\mathbf{W}^{(l)}$ is a trainable weight matrix.³⁹ After the graph convolutional layers, we apply a fully connected layer followed by a softmax classification layer, which outputs a probability of association $Pr(u \in S|\tilde{S})$ for each node u in the network. We train the model with cross-entropy loss.

We used one graph convolutional layer with 128 hidden units and one fully-connected layer with 16 hidden units. We train for 40 epochs with a learning rate of 0.001, dropout rate of 0.5 and L2 weight decay 5.0×10^{-4} . We evaluated deeper GCNs with two or three graph convolutional layers, but achieved the best performance with just one.

3) Protein embeddings. We generate neighborhood-preserving embeddings for the nodes in our networks with *node2vec*.⁴³ The embeddings are 128-dimensional and generated with the return and in-out parameters both set to 1.

For each phenotype and its set of known associations \tilde{S} , we train a logistic regression model to predict the full association set S . For each node u , the model accepts u 's *node2vec* embedding as input and outputs u 's probability of association $Pr(u \in S|\tilde{S})$. To train the model, we assign

a positive label to the nodes in \tilde{S} and randomly sample an equal number of molecules to use as negative examples.

4) Direct neighbor scoring. Given a phenotype and its set of associated molecules \tilde{S} , we rank molecules by the fraction of their direct interactors that are in \tilde{S} .

5) DIAMOnD (connectivity significance). DIAMOnD is a disease protein discovery method that predicts novel disease-protein associations by measuring the statistical significance of the interactions between a query protein and known disease associations.⁴⁷ Given a disease and its set of known disease associations \tilde{S} , the connectivity p -value for a query protein u is given by

$$p\text{-value}(u, \tilde{S}) = \sum_{k=|N(u) \cap \tilde{S}|}^{d_u} \frac{\binom{|\tilde{S}|}{k} \binom{n-|\tilde{S}|}{d_u-k}}{\binom{n}{d_u}} \quad (\text{B.3})$$

where d_u is the degree of protein u and n is the total number of proteins in the network. On each iteration of the algorithm, the protein with the lowest connectivity p -value is added to the association set. When calculating p -values on later iterations, we weight the original association set by α . We use $\alpha = 10$, as is recommended by the authors of DIAMOnD.⁴⁷ For each disease, we run DIAMOnD for 300 iterations and rank proteins in the order that they are added.

6) Probabilistic latent model. We use probabilistic non-negative matrix factorization to predict drug-target interactions. First, we construct a binary matrix $\mathbf{V} \in \{0, 1\}^{m \times n}$ where m is the number of drugs, n is the number of proteins and

$$v_{ij} = \begin{cases} 1 & \text{if drug } i \text{ targets protein } j \\ 0 & \text{otherwise} \end{cases} \quad (\text{B.4})$$

The goal is to factorize \mathbf{V} into two non-negative matrices $\mathbf{W} \in \mathbb{R}^{m \times d}$ and $\mathbf{H} \in \mathbb{R}^{d \times n}$ such that $\mathbf{V} \approx \mathbf{WH}$. This objective can be formalized probabilistically and solved for with an expectation-maximization algorithm.^{30,31,48} With approximations for the latent variables \mathbf{H} and \mathbf{W} , we can predict new drug-target interactions by computing $\tilde{\mathbf{V}} = \mathbf{WH}$. We use an implementation of probabilistic non-negative matrix factorization from the NIMFA library.⁴⁹

Appendix B.3. Network analysis

Here, we provide details on the methods used in our network analysis.

Mutual interactor neighborhoods. We introduce *mutual interactor neighborhoods*, a new way to analyze phenotypes in molecular networks. A mutual interactor neighborhood is the induced subgraph of a phenotype's nodes and mutual interactors. We compute a mutual interactor neighborhood for every phenotype in our study and compute standard network metrics on these subgraphs including density and conductance.

Most importantly, we find that visualizing mutual interactor neighborhoods can deliver insights into the mechanisms behind molecular phenotypes. In Figures 1, 2 and 3 and Figure A3, we show a few illustrative examples of mutual interactor neighborhood visualization. We implement a software package that allows users to quickly visualize a phenotype's mutual interactor neighbor in the context of *Mutual Interactors*' predictions for that phenotype.

Network metrics. We consider two network metrics defined over graphs $G = \{V, E\}$. These metrics are important for reasoning about the network structure of molecular phenotypes.

- (1) **Direct Interactor (DI) score.** Consider a set of nodes $S \subseteq V$. The direct interactor score of a node $u \in S$ is given by

$$\text{DI}(u, S) = \sum_{v \in S \setminus \{u\}} \frac{1}{\sqrt{d_u d_v}} \quad \mathbf{1}[(u, v) \in E] \quad (\text{B.5})$$

where d_i is the degree of node i . Intuitively, the direct interactor score tells us how well-connected u is to the other nodes in the set S .

- (2) **Mutual Interactor (MI) score.** The mutual interactor score of a node $u \in S$ is given by

$$\text{MI}(u, S) = \sum_{v \in S \setminus \{u\}} \frac{1}{\sqrt{d_u d_v}} \sum_{k \in M_{u,v}} \frac{1}{\sqrt{d_k}} \quad (\text{B.6})$$

where d_i is the degree of node i and $M_{u,v} = N(u) \cap N(v)$ is the set of proteins that interact with both u and v . Intuitively, the mutual interactor score is a degree-normalized count of the mutual interactors between u and the other nodes in the set S . Roughly speaking, higher mutual interactor scores imply more shared neighbors between u and the rest of S . Note that we can compute pairwise mutual interactor scores for an entire graph very efficiently with sparse matrix multiplication:

$$\text{MI}(u, \{v\}) = (\mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}})_{u,v} \quad (\text{B.7})$$

where \mathbf{A} is the adjacency matrix and \mathbf{D} is the diagonal degree matrix. With these pairwise mutual interactor scores pre-computed, we can in linear time compute the mutual interactor score for any node u and full set S since

$$\text{MI}(u, S) = \sum_{v \in S \setminus \{u\}} \text{MI}(u, \{v\}). \quad (\text{B.8})$$

Computing Mutual Interactor scores in this way is critical for efficiently analyzing large molecular networks, especially when performing permutation tests with thousands of iterations.

We cannot directly compare DI and MI scores because their scales are drastically different. To compare them in a fair way, we normalize each score into a p -value via permutation test. We describe this approach in the statistical analysis section below.

Appendix B.4. Statistical analysis

Throughout the study we use rigorous statistical analysis with non-parametric permutation tests, exact tests and conservative corrections for multiple hypothesis testing.

Evaluating the statistical significance of network structures. We conduct permutation tests to determine the statistical significance of the network structures measured by the metrics described above. For each network metric, we determine whether the molecules associated with the phenotypes in our dataset score significantly higher than molecules drawn randomly from the network.

For each phenotype in our study, we collect the set of proteins S that are associated with the phenotype. Then, for each protein $u \in S$, we compute both $\text{DI}(u, S)$ and $\text{MI}(u, S)$. We then randomly

sample 1,000 proteins \hat{u} with degree similar to u and compute $\text{DI}(\hat{u}, S \setminus \{u\})$ and $\text{MI}(\hat{u}, S \setminus \{u\})$. For each network metric, we compute a p -value equal to the fraction of random proteins \hat{u} with a more extreme value than the true associated protein u .

As an additional analysis, for each set S , we randomly generate 1,000 node sets \hat{S} that match S in size and degree distribution. We compute the average direct interactor and mutual interactor scores on the node sets and calculate a p -value equal to the number of random node sets \hat{S} with a more extreme value than the true set S .

To prevent repeatedly sampling the same proteins when u is of high degree, we use the sampling approach proposed by Guney *et al.*⁵⁰ Specifically, we group proteins of similar degree into bins of fixed width. Bin widths are chosen so as to maximize the total number of bins while maintaining that all have at least 250 nodes.

Functional enrichment analysis. For each disease in our dataset we conduct a functional enrichment analysis to determine which molecular functional are enriched in its set of associated proteins. A molecular function c is enriched in a set of proteins S if proteins in S are associated with c or its gene ontology descendants more frequently than other proteins. For each molecular function, we compute a p -value with Fisher's exact test. Since we are performing a statistical test for large number of molecular functions, we use Benjamini-Hochberg false discovery rate correction, which controls the false positive rate (false positive rate $\leq 5\%$).⁵¹

For each disease, we consider the top $|S|$ predictions made by *Mutual Interactors* and perform a gene ontology enrichment analysis on the predicted proteins, ranking molecular functions by their enrichment p -value. We repeat this process for the disease's verified associations and compute the Spearman rank-order correlation between the predicted and verified association enrichment rankings. The results of this analysis are shown in ??.

Appendix C. Extended Datasets and Preprocessing

We start by describing the molecular network data considered in this study. We then proceed with an overview of our data on disease proteins, drug targets and molecular functions.

Appendix C.1. Protein-protein interaction (PPI) networks

Our PPI networks are composed of experimentally-verified interactions between proteins. The interactions we consider are diverse in nature: they include signalling, regulatory, metabolic-pathway, kinase-substrate and protein complex interactions. An edge (u, v) in a PPI network indicates that one or more of these interaction types have been experimentally documented between the proteins u and v . Note that we make no distinction between interaction types when constructing the network: each edge is unweighted, undirected and unlabeled.

We use the human PPI network compiled by Menche *et al.* as our PPI primary network.¹⁶ This network integrates several protein-protein interaction databases, including TRANSFAC for regulatory interactions,⁵² MINT and IntAct for yeast to hybrid binary interactions,^{53,54} and CORUM for protein complex interactions.⁵⁵ The network has $n = 21,557$ proteins and $m = 342,353$ edges. Unless specified, we report results from this network. To ensure that our results are not an artifact of study bias, we replicate our results on two additional human PPI networks derived from the BioGRID¹⁴

and STRING¹⁵ databases (See D). We also validate the mutual interactor principle on the human reference interactome (HuRI) generated by Luck *et al.*¹³ Specifically, we use HI-union a combination of HuRI and several related efforts to systematically screen for protein-protein interactions. The HI-union PPI network consists of $n = 8,992$ proteins and $m = 62,841$ edges.

We build PPI networks for three additional species (*A. thaliana*, *M. musculus*, and *S. cerevisiae*) using BioGRID, a database of protein-protein interaction data curated from primary biomedical literature.¹⁴ BioGRID interactions are supported by various forms of experimental evidence including protein-fragment complementation assays and two-hybrid screens. For *A. thaliana*, we consider every BioGRID interaction and build a PPI network with $n = 10,288$ nodes and $m = 48,410$ edges. We do the same for *M. musculus*, generating a network with $n = 7,323$ nodes and $m = 20,963$ edges. For *S. cerevisiae*, we consider only physical interactions and build a network with $n = 5,988$ nodes and $m = 111,846$ edges.

Appendix C.2. Genetic interaction network

Two genes share a genetic interaction if mutations in both produce a phenotype that neither would produce individually. When mutations in both genes combine to cause cell death, we say the genes share a *negative genetic interaction*. We consider a network of negative genetic interactions in *S. cerevisiae* constructed with quantitative genetic interaction scores derived from double mutant screens.^{56,57} The network consists of $n = 5,548$ genes and $m = 327,477$ interactions.

Appendix C.3. Signaling network

We use a network of human signaling interactions from OmniPath, an integration of 27 signaling pathways found in literature.^{58,59} These pathways include both causal and undirected interactions. With this data, we construct an undirected signaling network of $n = 5,052$ nodes and $m = 17,884$ edges.

Appendix C.4. Disease protein dataset

We use the DisGeNet database of gene-disease associations.⁶⁰ In DisGeNet, a gene is associated with a disease when an alteration in the gene's code, expression, or protein product underpins the disease. We only consider DisGeNet associations that are expert-curated (i.e. associations from either the Comparative Toxicogenomics Database⁶¹ or the Universal Protein Resource⁴¹). We consider every human disease in DisGeNet with at least ten expert-curated gene associations. To avoid duplicate diseases in our dataset, if two diseases have a Jaccard similarity of 0.7 or greater between their associated gene sets, we discard the one with fewer associated genes. This leaves us with a total of 1,811 diseases and 75,744 associations. The diseases in our dataset have a median of 21 associated genes. The number of associated genes varies widely with a standard deviation of 69.3. For example, there are 474 genes associated with prostatic neoplasms, but only 10 associated with Dyslexia. For our analysis, associated genes are mapped to their protein products in the PPI network. We say a protein is associated with a disease if it is a product of a gene that is associated with that disease.

Disease categories.

Using the Disease Ontology, we group diseases into 17 categories from the second level of the ontology. Categories include nervous system diseases, respiratory system diseases, and cancers.⁶² Of the 1,811 diseases we considered, only the 510 found in the ontology were categorized. Nervous system diseases represent the largest disease category with 101 diseases, while diseases of the reproductive system form the smallest with only 8 diseases. The median category size was 22 diseases.

Appendix C.5. *Molecular function dataset*

The Gene Ontology (GO) is a structured knowledge-base that associates genes with biological processes and molecular functions.¹⁹ For each molecular function in the Gene Ontology, we collect the set of genes associated with the function. We only consider associations with an experimental evidence code (i.e. EXP, IDA, IPI, IMP, IGI, IEP) or a high-throughput evidence code (i.e. HTP, HDA, HMP, HGI, HEP). We build a dataset of all molecular functions with between 10 and 100 associated genes. If two molecular functions share a Jaccard similarity of 0.7 or greater between their associated gene sets, we discard the one with fewer associated genes. We do this for four species: *H. sapiens* (499 molecular functions), *M. musculus* (197 molecular functions), *A. thaliana* (161 molecular functions), and *S. cerevisiae* (255 molecular functions).

We repeat this process for each biological process in the Gene Ontology. Again, we consider four species: *H. sapiens* (1,928 biological processes), *M. musculus* (1,764 biological processes), *A. thaliana* (691 biological processes), and *S. cerevisiae* (788 biological processes).

Appendix C.6. *Drug-target interaction dataset*

We use drug-target interaction data from DrugBank,¹⁸ a database of drug mechanisms and targets. In DrugBank, a drug targets a protein if the drug has an experimentally-verified, physiological or pharmaceutical effect on the protein. We consider every drug with at least ten targets. Drugs in our dataset target 21.27 proteins on average and 19 target thirty or more. Of the 21,557 proteins in the human PPI network, only 2,212 are targeted by a drug in DrugBank.

Drug side effect similarity. We use a dataset of drug side-effect similarities compiled by Tatonetti *et al.*²⁹ It includes pairwise side-effect similarity scores between all drugs in the OFFSIDES and SIDER databases. To compute side-effect similarity scores between two drugs they use a simplified version of the approach proposed by Campillos *et al.*³³ By extracting side effects from the Unified Medical Language System (UMLS), they can compute side-effect similarity scores based on distances in the UMLS ontology. Their approach then down-weights commonly co-occurring side effects (e.g. nausea and vomiting) and normalizes similarity scores into *p*-values via permutation test.

Appendix D. Replicating the disease protein prediction experiments on other interaction networks

Existing protein-protein interaction networks are incomplete. It is estimated that state-of-the-art interactomes are missing 80% of all the interactions in human cells.¹⁶ To ensure that our results are not an artifact of study bias in one network, we replicate disease protein prediction experiments on two additional PPI networks. These networks are culled from different sources and vary substantially

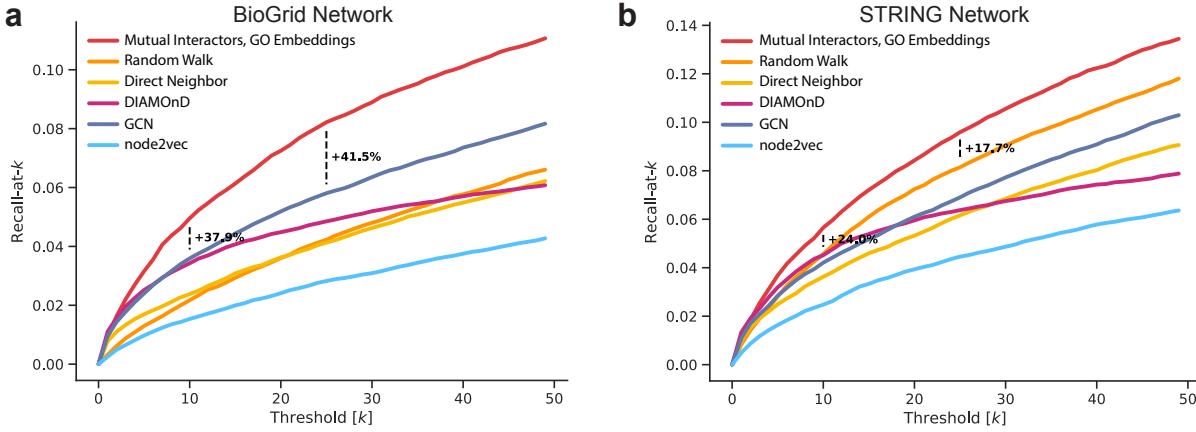


Fig. D1: (a-b) Disease protein prediction performance on two supplementary human, protein-protein interaction networks. Included are *Mutual Interactors* and existing disease protein prediction methods. Plot shows average recall-at- k across all diseases for $k = 0$ to $k = 50$. We include the percent-increase in recall over the next best performing method at $k = 10$ and $k = 25$. (a) BioGrid⁷ Network. *Mutual Interactors* achieves a recall-at-10 37.9% higher than the next best performer, a 1-Layer Graph Convolutional Network (GCN).²⁵ (b) STRING⁷ Network. *Mutual Interactors* achieves a recall-at-10 24.0% higher than the next best performer, Random Walks.⁴²

from our primary network, the one compiled by Menche *et al.*¹⁶

The first network comes from BioGRID, a database of protein-protein interaction data curated from primary biomedical literature.⁶³ BioGRID interactions are supported by experimental evidence such as protein-fragment complementation assays and two-hybrid screens. We take every BioGRID interaction between human proteins and build a PPI network with 16,577 proteins and 271,497 edges. Note that 25% of the interactions in the BioGRID network are not found in our primary network. The Jaccard similarity between the interactions of our primary network and the BioGRID network is only 0.504.

We repeat our disease protein discovery experiments on the BioGRID network using ten-fold cross validation. We find that *Mutual Interactors* recovers a larger fraction of held-out proteins than do existing disease protein discovery methods. Specifically, in 8.21% of disease-protein associations *Mutual Interactors* ranks the protein within the first 25 proteins in the network (recall-at-25 = 0.0821). *Mutual Interactors*'s performance represents an improvement of 41.5% in recall-at-25 over the next best performing method, a 1-layer GCN (recall-at-25=0.058). Other network-based methods of disease protein discovery including DIAMOnD (recall-at-25= 0.0485) and random walks (recall-at-25 = 0.0413) recover considerably fewer disease-protein associations. See ??.

The second network is derived from the STRING interaction database.⁶⁴ We consider all STRING interactions with at least one source of experimental evidence and construct a PPI network with 13,821 proteins and 144,648 edges. Note that 15% of the interactions in the STRING network are not represented in our primary network. The Jaccard similarity between the interactions of our primary network and the STRING network is only 0.333.

As with the two other networks, we simulate disease protein discovery using ten-fold cross validation, and again we find that *Mutual Interactors* outperforms existing disease protein discovery methods. Specifically, in 9.59% of disease-protein associations *Mutual Interactors* ranks the protein within the first 25 proteins in the network (recall-at-25 = 0.0959). *Mutual Interactors*'s performance represents an improvement of 17.7% in recall-at-25 over the next best performing method, random walks (recall-at-25=0.0814). Other network-based methods of disease protein discovery including

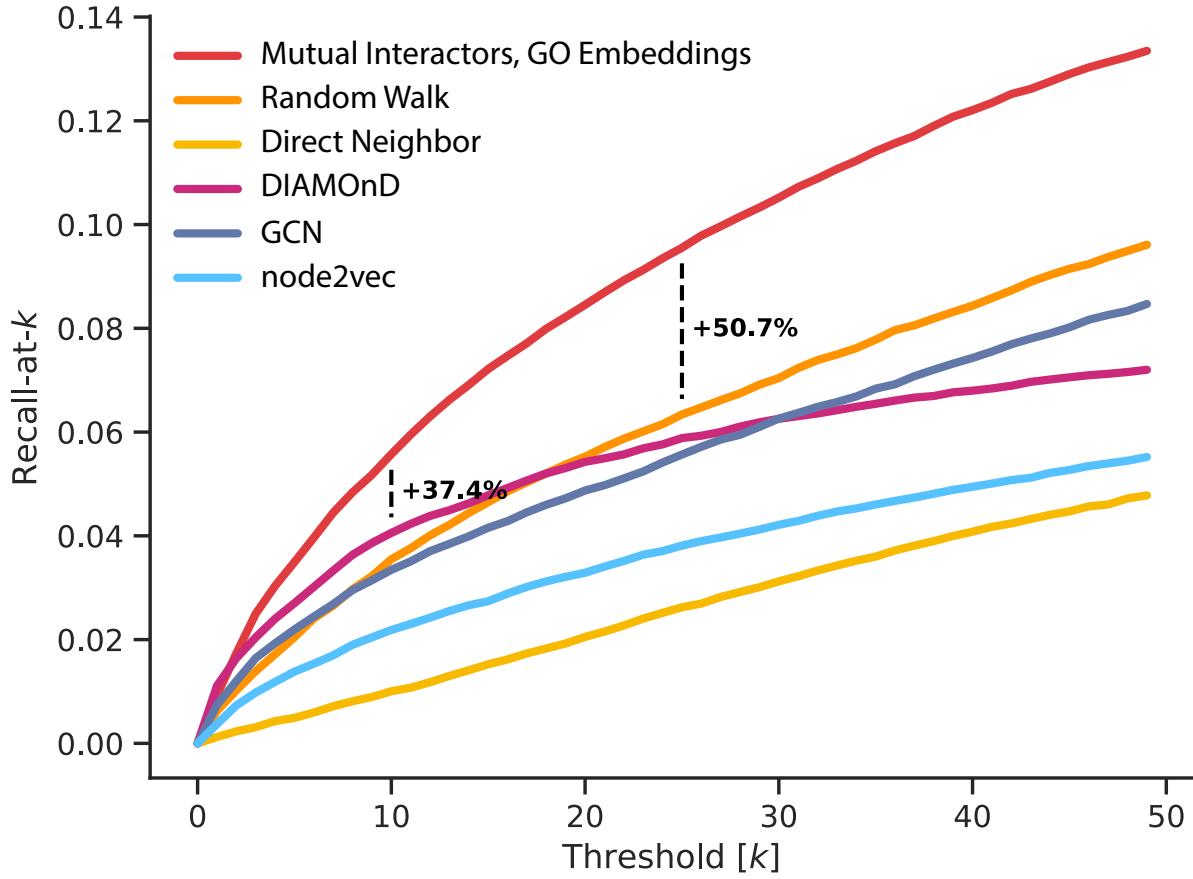


Fig. E1: Disease protein prediction performance with challenging train/test splits. We evaluate *Mutual Interactors* in a more challenging setting where similar diseases do not straddle splits. The plot shows average recall-at- k across all diseases for $k = 0$ to $k = 50$. We include the percent-increase in recall over the next best performing method at $k = 10$ and $k = 25$. *Mutual Interactors* achieves a recall-at-10 37.4% higher than the next best performer, DIAMOnD.¹⁰ *Mutual Interactors*'s advantage is slightly less pronounced in this setting, though it remains the strongest method by a significant margin.

DIAMOnD (recall-at-25= 0.0638) and graph convolutional neural networks (recall-at-25 = 0.0690) recover considerably fewer disease-protein associations. See ??.

Indeed, even as we change the network, *Mutual Interactors* maintains a considerable advantage. Other methods are less consistent. Random walks, the second best performing method on our primary network, is outpaced by nearly all other methods when we use the BioGRID network. Our 1-layer GCN, the second best performing method on the BioGRID network, is only the fourth best performing method when we use our primary network. *Mutual Interactors*'s consistency suggests that it is robust to incomplete PPI networks.

Appendix E. Disease-wise Test Split

Many of the diseases in the DisGeNet database are closely related. Oftentimes two different entries are only slight variants of the same disease. For example, DisGeNet contains entries for both Brain Edema and Cerebral Edema. To avoid double-counting diseases, we find and remove duplicates (see Methods). Specifically, if two disease pathways have a Jaccard similarity of 0.7 or greater, we remove the smaller disease pathway.

However, consider these two diseases with Jaccard similarity of 0.68: hereditary nonpolyposis colorectal cancer and malignant tumor of the colon. These diseases are substantively different, so we should not treat them as duplicates, yet they are highly related and share genetic underpinnings. Since *Mutual Interactors* is trained with disease-wise cross validation (see Methods), and the other methods are not, one could argue it is unfair for *Mutual Interactors* to train on one and evaluate on the other.

To address these concerns, we devise a more challenging disease-wise split where diseases with similar genetic underpinnings are placed in the same split. To generate such a split, we build a network of diseases with an edge between every pair of diseases whose associated genes have a Jaccard similarity greater than 0.3. We then partition the connected-components of this network into ten folds, each with the same number of diseases. Using this split with disease-wise cross validation, we guarantee that the diseases with which we train *Mutual Interactors* will be substantively different than those with which we evaluate it.

As expected, *Mutual Interactors*'s performance dips slightly. Yet even in this more challenging setting, *Mutual Interactors* maintains a substantial advantage over existing methods. *Mutual Interactors*'s recall-at-25 with the challenging split is 0.095, down 6% from 0.102 in the original setting. This represents a performance gain of 50.7% over the next best performing method Random Walks (recall-at-25=0.063). See ??.

Appendix F. Analyzing *Mutual Interactors* performance across disease categories

Here, we explore how *Mutual Interactors*'s disease protein prediction performance varies across disease categories. Using the Disease Ontology, we group diseases into 17 categories from the second level of the ontology. Categories include nervous system diseases, respiratory system diseases, and cancers.⁶² Of the 1,811 diseases we considered, only the 510 found in the ontology were categorized. Nervous system diseases represent the largest disease category with 101 diseases, while diseases of the reproductive system form the smallest with only 8 diseases. The median category size was 22 diseases.

Mutual Interactors outperforms existing methods in every disease category considered. Specifically, in all seventeen categories, *Mutual Interactors*'s median recall-at-100 exceeds Random Walk's. *Mutual Interactors* provides the largest gains for cancers and benign neoplasms, achieving a recall-at-100 at least as high as random walks' on 85.0% of cancers and 95.0% of benign neoplasms. *Mutual Interactors* provides the smallest gains for diseases of the urinary, reproductive and gastrointestinal system. *Mutual Interactors* performs at least as well as random walks on 62.5% of reproductive system diseases, 65.7% of gastrointestinal system diseases and 66.7% of urinary system diseases. See F1.

Appendix G. Parametric *Mutual Interactors*

Because the *Mutual Interactors* model includes a parameter w_z for every node z in the network, the number of parameters in the model scales linearly with the size of the network. Here, we describe a parametric *Mutual Interactors* model where the number of parameters is independent of network size and show that it performs on par with the original *Mutual Interactors* model.

Instead of learning a weight w_z for each node z in the network, we learn one scalar-valued

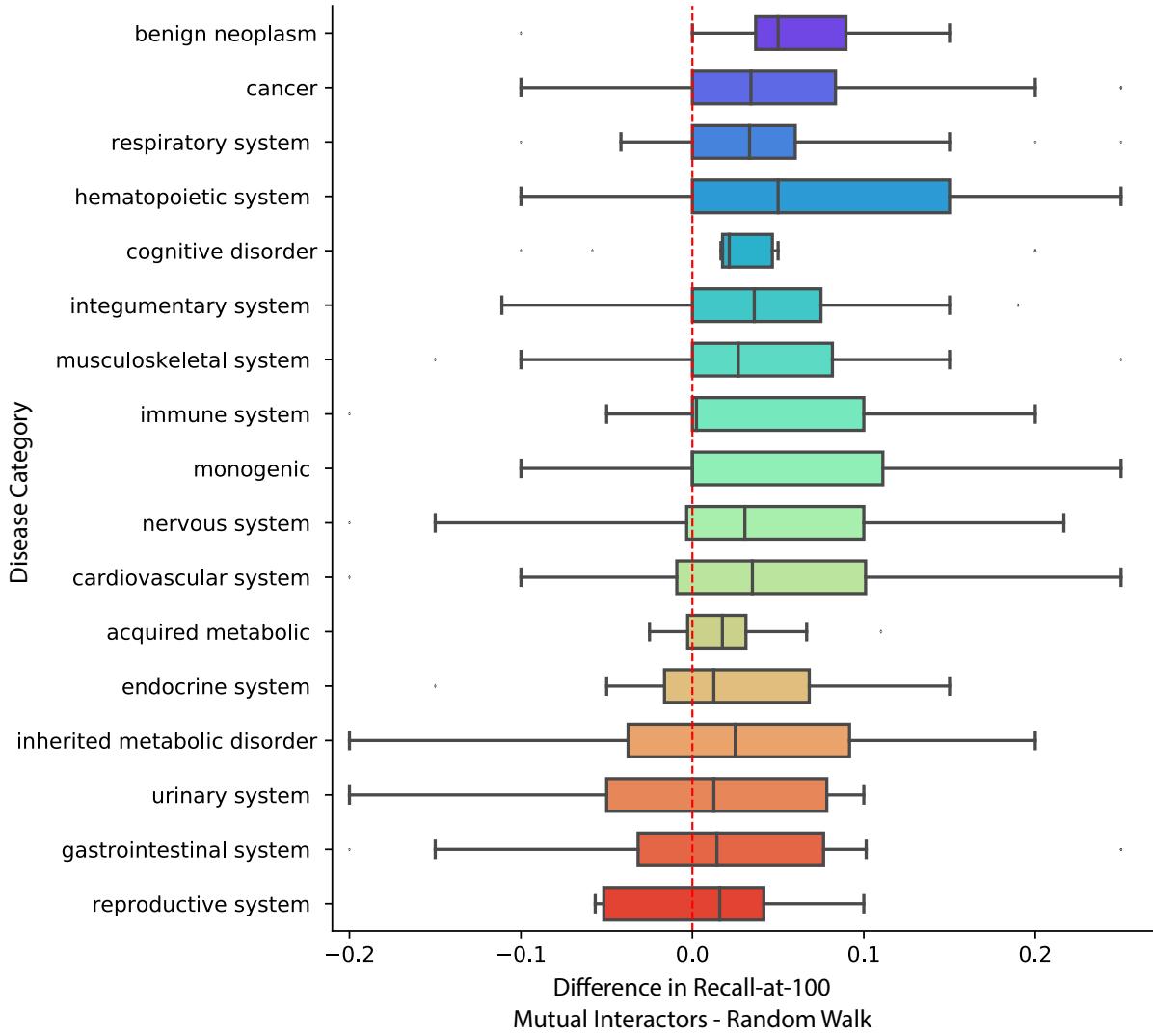


Fig. F1: Comparison of *Mutual Interactors* and Random Walks⁴² by disease category. For each disease, we compute the difference between *Mutual Interactors* and Random Walks recall-at-100. Then, for each disease category, we show a box-plot of the differences within the category. The dotted red line represents equal performance.

function $g(\mathbf{x}_z)$ mapping a node's features \mathbf{x}_z to a scalar that takes the place of w_z in the model. Specifically, we model the probability that a query node u is in a phenotype's complete association set S given its known associations $\tilde{S} \subseteq S$ as

$$Pr(u \in S | \tilde{S}) = \sigma \left(a \left(\sum_{v \in S} \frac{1}{\sqrt{d_v d_u}} \sum_{z \in C_{v,u}} \frac{g(\mathbf{x}_z)}{\sqrt{d_z}} \right) + b \right) \quad (\text{G.1})$$

We use gene ontology embeddings for our protein features \mathbf{x}_z . These embeddings encode information on the molecular functions of proteins in human cells. We generate an embedding $\mathbf{x}_z \in \mathbb{R}^{1024}$ for each protein z by first constructing a high-dimensional, binary vector that encodes the protein's annotations from the first five levels of the gene ontology.⁴¹ We then reduce the dimensionality of this vector with principal component analysis. In G1c, we use t-SNE to illustrate how proteins with similar functional profiles tend to cluster in our embedding space.

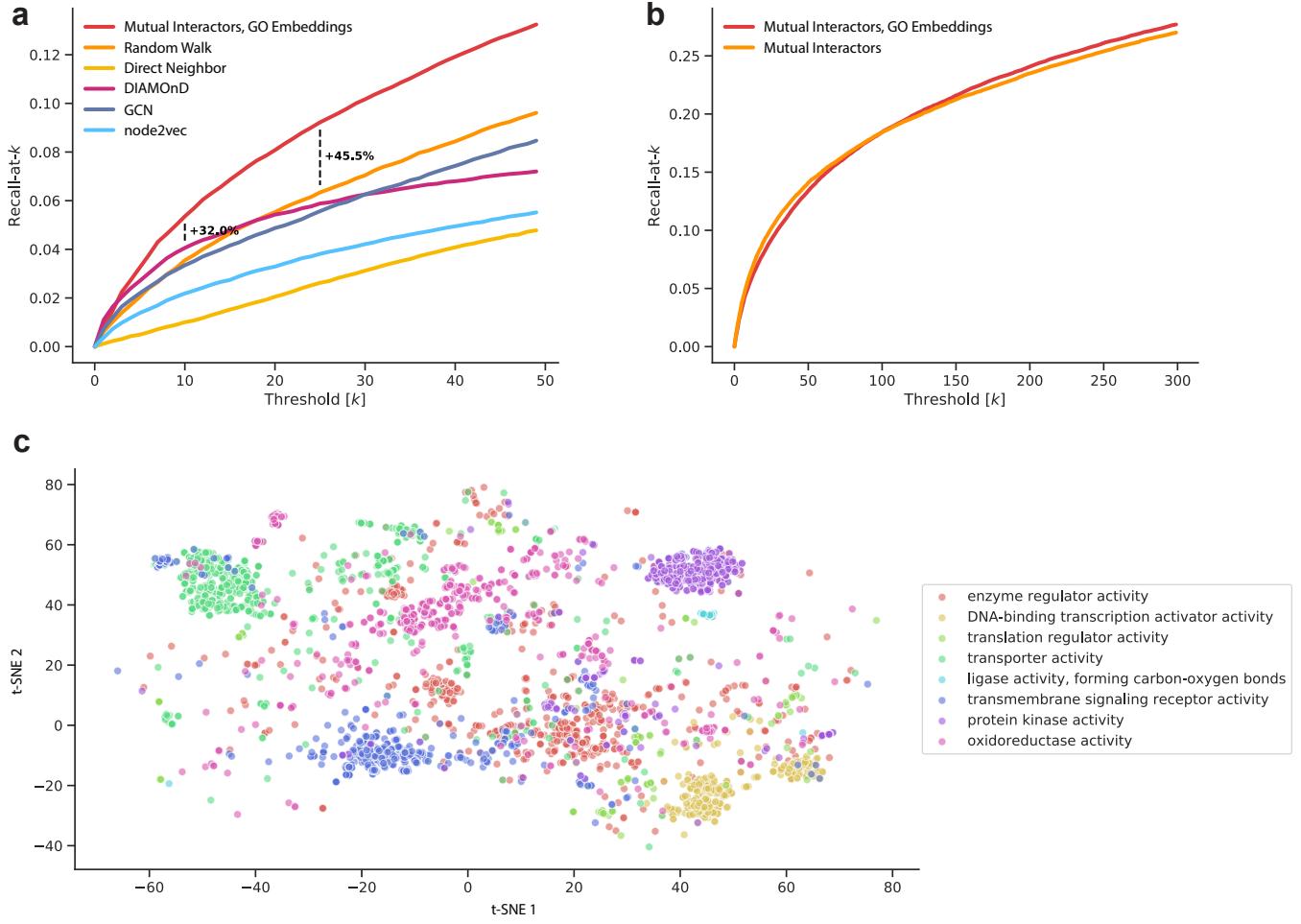


Fig. G1: (a) Evaluation of parametric *Mutual Interactors*' disease protein prediction performance. Plot shows average recall-at- k across all diseases for $k = 0$ to $k = 50$. The parametric model achieves a recall-at-25 45.5% higher than the next best performer, Random Walks.⁴² (b) We evaluate parametric *Mutual Interactors* against the standard *Mutual Interactors* model. We plot average recall-at- k across all diseases for $k = 0$ to $k = 300$. Parametric *Mutual Interactors* outperforms *Mutual Interactors* at high thresholds. (c) t-SNE visualization of our GO Ontology³⁷ Embeddings. We plot proteins associated with 8 different molecular functions to show how proteins with similar functional profiles cluster in the embedding space.

Our scalar-valued function $g(\mathbf{x}_z)$ is a feed-forward neural network with one hidden layer,

$$g(\mathbf{x}_z) = \mathbf{W}_{\text{final}}^T \text{ReLU}(\mathbf{W}_{\text{hidden}}^T \mathbf{x}_z)) \quad (\text{G.2})$$

where $\mathbf{W}_{\text{hidden}} \in \mathbb{R}^{1024 \times 512}$ and $\mathbf{W}_{\text{final}} \in \mathbb{R}^{512 \times 1}$ are learnable weights. We train these weights by minimizing the weighted binary cross-entropy loss, as in the original *Mutual Interactors* model. We use ADAM with a learning rate of 1×10^{-2} , weight decay 10^{-5} , and a batch size of 200 diseases.²⁴

Proteins involved in cell-cell signaling are overrepresented in the set of proteins with highest *Mutual Interactors* weight. This suggests that mutual interactors in human disease favor specific functional roles and motivates using feature vectors that encode functional information. This way, $g(x_z)$ can learn the functional profiles most representative of mutual interactors in disease.

We find that parametric *Mutual Interactors* with GO Embeddings outperforms existing disease protein discovery methods. Specifically, in 9.22% of disease-protein associations parametric *Mutual Interactors* ranks the protein within the first 25 proteins in the network (recall-at-25 = 0.0959). Parametric *Mutual Interactors*'s performance represents an improvement in recall-at-25 of 45.5% over

the next best performing method, random walks (recall-at-25=0.0633). See G1b.

In G1b we compare parametric *Mutual Interactors* with the standard *Mutual Interactors* model. Though the standard *Mutual Interactors* model achieves slightly higher recall at low thresholds, parametric *Mutual Interactors* begins to outpace standard *Mutual Interactors* at higher thresholds.