Project Category: Life Sciences
SUNet ID: bentyeh

Predicting Protein Interactions of Intrinsically Disordered Protein Regions

Benjamin Yeh, Department of Computer Science, Stanford University

CS 229 Project, Autumn 2018

https://github.com/bentyeh/disprot

Abstract

Recent research has increasingly demonstrated the ubiquity and functional importance of intrinsically disordered proteins (IDPs). Characterized by fluctuations through large conformational space, IDPs engage in dynamic protein-protein interactions (PPIs) that have not been well-understood through current structure-based analyses. We build on previous work on IDP PPI prediction solely using sequence information and analyze the performance of various machine learning algorithms. We achieve top performance on a previously published IDP PPI dataset by using new featurization and data augmentation techniques. However, the results are difficult to interpret in terms of concrete protein pair characteristics that are favorable for interactions, and more work still needs to be done towards improved feature considerations.

Introduction

Over the last two decades, many algorithms have been developed to predict regions of disorder (where there is no stable secondary or tertiary structure) within protein sequences[1,2,3,4]. However, less is known about how these disordered regions interact with other proteins. Such research is important for several reasons: 1) a recent estimate[5] suggests that over a third of human proteins are intrinsically disordered; and 2) these intrinsically disordered proteins (IDPs) have widespread roles in cellular processes, such as cell signaling and regulation[6,7]. While there are many protein-protein interaction (PPI) prediction algorithms[8], they are largely based on knowledge from curated databases or models of energetically favorable interactions, both of which tend to rely on known protein structures. IDPs thus pose a unique challenge for PPI prediction.

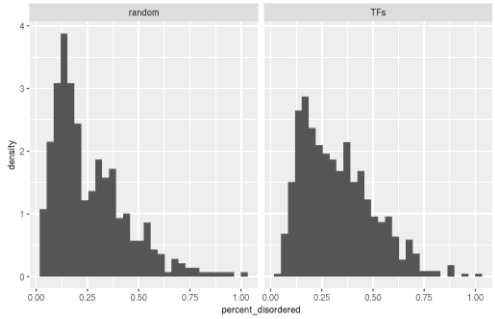

Figure 1. Comparison of disorder prevalence between transcription factors and control (random) sequences from the human genome. A motivating example in the study of disordered PPI's is the difference in percent disorder between transcription factors (TFs) and random sequences in the human genome. TFs are thought to recruit transcriptional complexes (such as mediator) via their disordered domains.

Related Work

Many protein-protein interaction prediction programs have been developed in the past, utilizing various heuristic methods as well as standard machine learning algorithms such as support vector machines and random forest (RF) algorithms.[9] However, much of the available training data comes from structural interactions, such as those found in the Protein Data Bank (PDB).[10]

In July 2018, Perovic et al.[11] published an analysis of "intrinsically-disordered protein" (IDP)-specific interaction predictors. They found an RF predictor to be the best among the models they tested, including gradient boosting machines, SVMs, and other linear models. They achieved an area-under-the-receiver-operating-curve (AUROC; a plot of true-positive versus false-positive rates at different binary decision thresholds) of 0.745, which at the time was the highest score achieved so far, with other algorithms all scoring below 0.7.

Dataset and Features

The labeled dataset was borrowed from Perovic et al., consisting of 90253 unique protein-protein pairs where at least one protein was considered "intrinsically disordered" by the DisProt protein disorder database.[12] Within this dataset, 19796 (22%) pairs were considered to be interacting (positive) and 70457 (78%) to be non-interacting (negative) based on the highly curated Human Integrated Protein-Protein Interaction rEference (HIPPIE) database.[13] (Non-interaction is difficult to validate experimentally, but it is commonly assumed that most proteins do not interact at any significant level. Therefore, for this dataset, non-interaction is defined as a lack of experimentally observed interaction.) This dataset was then filtered for proteins with length greater than 50 amino acids to avoid trivial length-dependent auto-correlative feature descriptors.

Each protein-protein interaction pair was featurized by concatenating the feature vectors of its constituent proteins. Individual proteins were featurized based on techniques used by Perovic et al. and additional methods available in the protr[14] R package. These features can be broadly classified into length-independent features and length-dependent features. The length-independent features describe compositional distributions, such as included amino acid, dipeptide, and transition frequencies. The length-dependent features describe distribution of amino acid properties along the sequence, including (amphiphilic) pseudo-amino acid composition (PAAC) descriptors and several auto-correlative measures. In total, this yields a 2449-dimensional vector for each protein; thus a single protein-protein pair is represented as a 4898-dimensional vector. Note that the dataset was also readily augmented: since whether two proteins interact should not depend on the order of the proteins, both orderings of concatenation of the individual protein feature vectors were included. Therefore, the fully-featurized augmented dataset was a 176548-samples by 4898-features matrix.

All data were normalized as z-scores (0 mean, 1 variance) then visualized through PCA plots to understand how well featurization separated the binary-labelled data. To reduce the feature complexity, only the top 446 principal components (corresponding to singular values > 1) were retained. Finally, the dataset was renormalized as z-scores and split 60-20-20 into training, validation, and test sets.

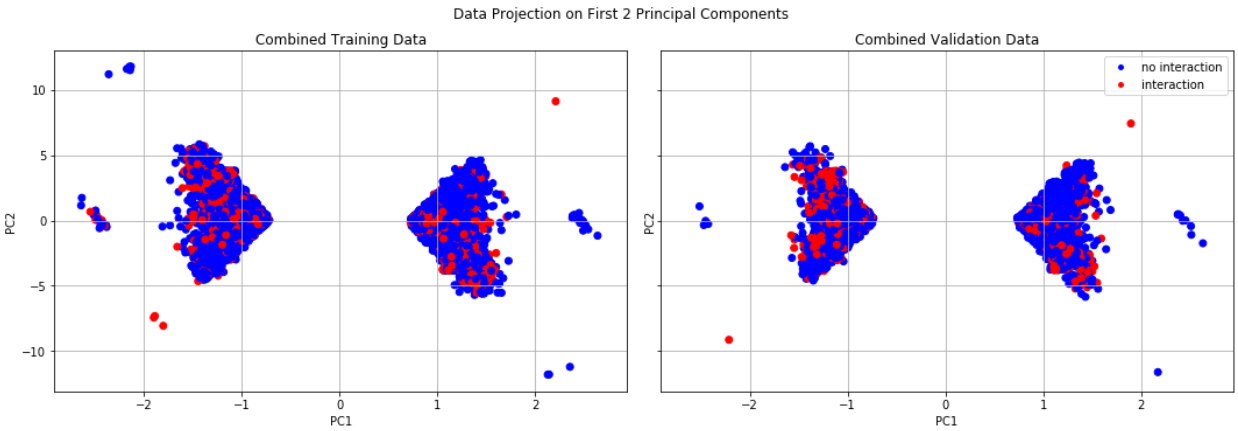

Figure 2: Projection of training and validation data onto first 2 principal components. The symmetry of the plots is a likely consequence of data augmentation procedures.

Methods

The PCA plots (Figure 2) did not reveal any clear linear decision boundary. However, since PCA merely looks for high-variance dimensions without specifically attempting to separate the data, there still could be other linear boundaries that would separate the data. Therefore, both linear and non-linear models were tested. The Python package scikit-learn[15] was used to build and train the models and evaluate their accuracy. Following Perovic et al., we used AUROC as the primary metric of comparison.

The linear models tested included L2-regularized logistic regression and support vector machines (SVM). L2-regularized logistic regression aims to minimize logistic loss over all training examples

$$\text{loss}_{\text{logistic}}\left(x^{(i)}, y^{(i)}; w, C\right) = \frac{1}{2}\left|\left|w\right|\right|_2^2 + C\log\left(\exp\left(-y^{(i)}w^\top x^{(i)}\right) + 1\right)$$

by taking negative gradient steps towards the global minimum. SVMs solve the optimization problem

$$\min_{\gamma,w,b} \frac{1}{2}\left|\left|w\right|\right|_2^2 + C\sum_{i=1}^{m}\xi_i$$

$$\text{s.t. } y^{(i)}\left(w^\top x^{(i)} + b\right) \geq 1 - \xi_i, \qquad \xi_i \geq 0, \qquad i = 1, \ldots, m$$

For both logistic regression and SVMs, the coefficient C trades off the objectives of regularizing the weight vector *w* and accurately classifying the training examples (for SVMs, specifically with functional margin at least 1). A higher value of *C* will therefore result in higher accuracy on the training set but with potentially worse generalization performance. In this project, C was varied 100-fold between 0.1 and 10.

The non-linear models tested included random forest (RF) classifiers, Gaussian- (or radial-basis function)-kernel SVMs, and neural networks. RFs are ensemble classifiers that average a large number of (relatively high-variance) decision trees (here, 10-100), each trained over a bootstrapped sample of the original data and a subset of the features (here, the top $\sqrt{n_{\text{features}}}$). Enforcing maximum depth (here, 5-50, or unconstrained) and minimum leaf size (here, 5) constraints further reduces variance. The Gaussian-kernel $K(s,z) = \exp\left(-\frac{\left|\left|x-z\right|\right|^2}{2\sigma^2}\right)$ SVMs were regularized similarly to the linear SVMs. The neural networks were setup with a 100-node hidden layer, ReLU activation, and parametrized by L2-regularization strengths between 0.01 and 0.0001, a constant learning rate, and a batch size of 200.

Results

Interestingly, the linear models consistently demonstrated less variance (overfitting) than the nonlinear models, with the exception of the Gaussian-kernel SVMs, which failed to converge within 50 iterations. (The maximum number of iterations was imposed due to time constraints and the quadratic time complexity of the algorithm used.)[15] Nonetheless, the RF models, which fit the training data very well and demonstrated low bias (but higher variance), performed very well. The model that achieved the highest AUROC score on the validation set was a random forest classifier of 100 trees and a max depth of 50. To analyze how well it likely generalizes to unseen data, the model was re-run on an unseen test set on which it achieved an AUROC score of 0.8268, very similar to its performance on the validation set.

The results were not surprising, given that the PCA plots failed to show strong evidence of linear decision boundaries, therefore suggesting an advantage for non-linear models. Furthermore, Perovic et al. had recorded their top performance with an RF model as well. However, the AUROC achieved here (0.8268) was higher than that reported in their paper (0.745), which is likely due to data augmentation described previously.

Unfortunately, interpreting the performance result of the top RF model is difficult. This is in part due to the ensemble nature of the RF model and even more so due to the PCA dimensionality-reduction step prior to training. It is therefore almost impossible to concretely explain what protein pair characteristics are favorable for interactions versus non-interactions.

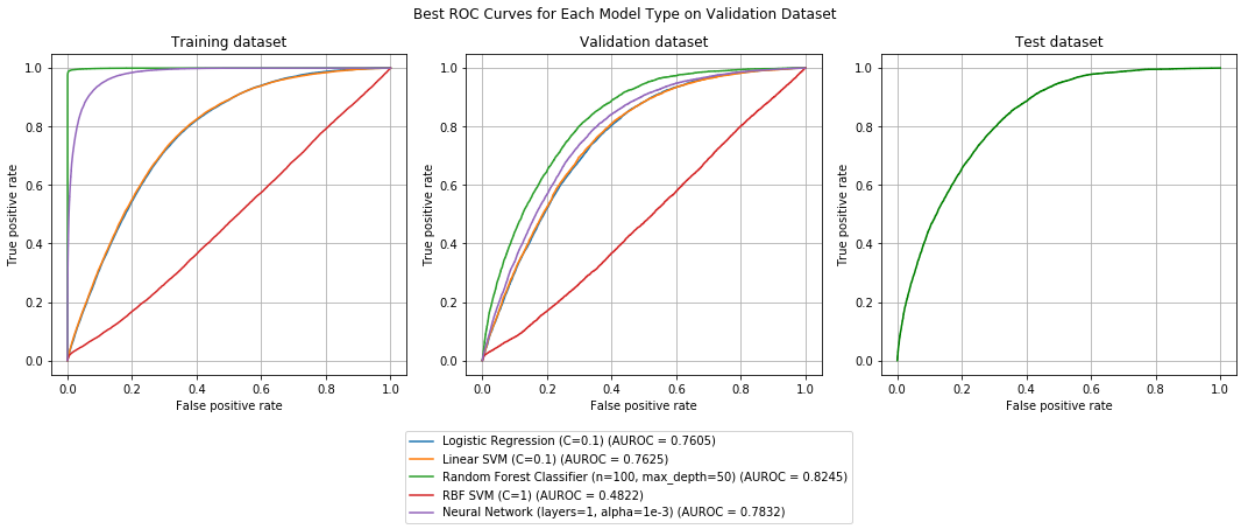

Figure 3: ROC curves across training and validation datasets for the best-performing (on the validation dataset) model of each model type. The ROC curve for the overall best-performing model is also shown for the test dataset. AUROC values in the legend correspond to performance on the validation dataset.

Model	Hyperparameters	AUROC (train)	AUROC (validation)	AUROC (test)
Logistic regression	C=0.1	0.7713	0.7605	--
Logistic regression	C=1	0.7713	0.7605	--
Logistic regression	C=10	0.7713	0.7605	--
SVM (linear)*	C=0.1	0.7731	0.7625	--
SVM (linear)*	C=1	0.7346	0.7292	--
SVM (linear)*	C=10	0.6195	0.6165	--
Random forest	10 trees, max depth n/a	0.9932	0.7811	--
Random forest	10 trees, max depth 10	0.8306	0.7657	--
Random forest	10 trees, max depth 50	0.9932	0.7812	--
Random forest	100 trees, max depth n/a	0.9995	0.8243	--
Random forest	100 trees, max depth 10	0.8545	0.7846	--
Random forest	100 trees, max depth 50	0.9995	0.8245	0.8268
Random forest	200 trees, max depth 5	0.7707	0.7527	--
Random forest	200 trees, max depth 10	0.858	0.7869	--
SVM (RBF)**	C=0.1	0.48	0.4817	--
SVM (RBF)**	C=1	0.4794	0.4822	--
SVM (RBF)**	C=10	0.4774	0.4799	--
Neural network	α=0.0001	0.9783	0.7829	--
Neural network	α=0.001	0.9775	0.7832	--
Neural network	α=0.01	0.9756	0.783	--

Table 1: Results for all models tested. *Did not converge after 1000 iterations. **Did not converge after 50 iterations.

Future Work

The moderate prediction accuracies achieved through this project demonstrate large potential for improvement. There are several simple extensions of the current project that deserve more attention. First, more (or all) of the original features could be considered, rather than the dimensionality-reduced set of 446 features along the principal components, which was enforced largely due to time and computational constraints. In addition, more advanced neural networks may also be capable in identifying better nonlinear decision boundaries, and by using more granular software packages (like Tensorflow or PyTorch), it would be possible to output loss gradients with respect to individual features to produce saliency maps, thereby allowing improved feature analysis. Finally, finer hyperparameter tuning would almost certainly yield better predictions on the training and validation sets. Many of the non-linear models demonstrated significant overfitting, which could be curbed through different regularization techniques: stricter (larger) minimum leaf sizes for the Random Forest Classifiers; early-stopping criteria and larger L2-regularization penalties (alpha) for the neural networks.

Different data sources could also be incorporated. The human proteome consists of over 20,000 proteins. Given that a third of them are predicted to be disordered, and disordered proteins participate in an average of over 100 PPIs each, there are a lot more PPIs that can be studied (and predicted). Proteomes from other species (especially well-studied model species like mice, yeast, and fruit flies) can contribute even more data. Many different databases have been set up to capture data produced by protein-protein interaction experiments and computational analyses. BioGRID, for example, currently contains 353,521 human PPIs[16], while STRING boasts 1.38 billion PPIs across over 2000 organisms.[17] Data quality, however, is still a major concern, especially since studies of PPIs often miss non-structured interactions. However, by filtering for biochemical techniques that are more likely to identify disordered PPIs (such as cross-linking mass spectrometry), more balanced datasets can be curated.

Finally, new featurization strategies appear to be crucial to improving prediction. Current techniques of featurizing individual proteins and then concatenating their feature vectors as a representation of a potential protein-protein interaction have been unsuccessful in producing visual separation of the binary data. Some possible future considerations include incorporating co-evolution information and energy models. Specifically, some disordered domains are known to stabilize upon interactions with other proteins;[18] such information can be matched with predicted or known protein surface geometries to improve predictions. Even more recently, a paper from September this year explores embeddings of protein complexes derived from PPI networks.[19]

Ultimately, improved understanding of disordered PPIs has the potential to elucidate many complexities of gene regulation, signal transduction, and other cellular processes. Such information can be invaluable for therapeutic development and further biomedical research.

[1] Romero, Pedro, et al. "Sequence complexity of disordered protein." *Proteins: Structure, Function, and Bioinformatics* 42.1 (2001): 38-48.
[2] Peng, Kang, et al. "Length-dependent prediction of protein intrinsic disorder." *BMC Bioinformatics* 7.1 (2006): 208.
[3] Ishida, Takashi, and Kengo Kinoshita. "PrDOS: prediction of disordered protein regions from amino acid sequence." *Nucleic Acids Research* 35. (2007): W460-W464.
[4] Dosztanyi, Zsuzsanna, et al. "The pairwise energy content estimated from amino acid composition discriminates between folded and intrinsically unstructured proteins." *Journal of Molecular Biology* 347.4 (2005): 827-839.
[5] Ali, Muhammad, and Ylva Ivarsson. "High-throughput discovery of functional disordered regions." *Molecular Systems Biology* 14.5 (2018): e8377.
[6] Wright, Peter E., and H. Jane Dyson. "Intrinsically Disordered Proteins in Cellular Signalling and Regulation." *Nature Reviews Molecular Cell Biology* 16.1 (2015): 18–29.
[7] Liu, Jiangang et al. "Intrinsic Disorder in Transcription Factors." *Biochemistry* 45.22 (2006): 6873–6888.
[8] Singh, Rohit, et al. "Struct2Net: a web service to predict protein–protein interactions using a structure-based approach." *Nucleic Acids Research* 38. (2010): W508-W515.
[9] Park, Yungki, and Edward M. Marcotte. "Flaws in evaluation schemes for pair-input computational predictions." *Nature methods* 9.12 (2012): 1134.
[10] Berman, Helen M., et al. "The protein data bank." *Nucleic acids research* 28.1 (2000): 235-242.
[11] Perovic, Vladimir et al. "IDPpi: Protein-Protein Interaction Analyses of Human Intrinsically Disordered Proteins." *Scientific Reports* 8.1 (2018): 10563.
[12] Piovesan, Damiano, et al. "DisProt 7.0: a major update of the database of disordered proteins." *Nucleic acids research*45.D1 (2016): D219-D227.
[13] Schaefer, Martin H., et al. "HIPPIE: Integrating protein interaction networks with experiment based quality scores." *PloS one* 7.2 (2012): e31826.
[14] Xiao, Nan, et al. "protr/ProtrWeb: R package and web server for generating various numerical representation schemes of protein sequences." *Bioinformatics* 31.11 (2015): 1857-1859.
[15] Pedregosa, Fabian, et al. "Scikit-learn: Machine learning in Python." *Journal of Machine Learning Research* 12.Oct (2011): 2825-2830.
[16] Chatr-Aryamontri, Andrew, et al. "The BioGRID interaction database: 2017 update." *Nucleic acids research* 45.D1 (2017): D369-D379.
[17] Szklarczyk, Damian, et al. "The STRING database in 2017: quality-controlled protein–protein association networks, made broadly accessible." *Nucleic acids research* (2016): gkw937.
[18] Dosztányi, Zsuzsanna, Bálint Mészáros, and István Simon. "ANCHOR: web server for predicting protein binding regions in disordered proteins." *Bioinformatics* 25.20 (2009): 2745-2746.
[19] Liu, Xiaoxia, et al. "Identifying protein complexes based on node embeddings obtained from protein-protein interaction networks." *BMC bioinformatics* 19.1 (2018): 332.