Ensemble Prediction of Intrinsically Disordered Regions in Proteins

Ahmed Attia
Stanford University

***Abstract*—Various methods exist for prediction of disorder in protein sequences. In this paper, the author develops and evaluates ensemble predictors of disordered regions in proteins utilizing 11 primary predictors. The ensemble predictors respectively use logistic regression, random forest, and support vector machine models. Ensemble methods were found to be a valuable approach for IDR prediction. An SVM ensemble classifier was the best performer on the dataset constructed from proteins found in DisProt.**

I. Introduction

Intrinsically Disordered Regions (IDRs) are regions of proteins which lack a well-defined three-dimensional structure. IDRs are prevalent in any genome and are known to have functional roles in foundational biological processes including transcription, translation, and cellular signal transduction [1]. Predicting IDRs aids in the structural and functional analysis of uncharacterized protein segments and disorder predictions have been successfully used to guide laboratory science [2].

Reflecting this importance, many methods for protein disorder prediction have been developed. These methods utilize different indicators including physicochemical properties of amino acids, evolutionary information, and interaction energies as well as various machine learning and statistical techniques. Different predictors may have different strengths and weaknesses stemming from their underlying assumptions, the datasets used for training, and the machine learning algorithms used. Ensemble prediction may achieve better accuracy by combining primary predictors to reduce individual weaknesses.

The ensemble predictors presented in this paper take as inputs feature vectors which consist of 11 disorder predictions for a residue in a particular protein. The dimension of the feature vector corresponds to the number of primary predictors used by the ensemble learner. The primary predictors used in this work are PONDR-VSL2b, DisEMBL-465, DisEMBL-hl, FoldIndex, ESpritzD, ESpritzN, EspritzX, GlobPlot, IUPred-Long, IUPred-Short, and JRONN. The output of the model is a binary classification of that feature vector as ordered or disordered. A visualization of feature construction is found in Fig. 1.

II. Related work

Various approaches have been applied to protein disorder prediction. Many of these approaches are based directly on the properties of the protein sequence combined with domain knowledge on protein folding. At the molecular level, when a protein folds, there is entropy loss, where entropy is a measure of disorder and a greater degree of disorder is favorable. Then, the energy stabilization of folding has to be sufficient to make up for the entropy loss due to folding. The IUPred predictors make use of this underlying cause of folding by estimating the interaction energies and predicting sequences with lower interaction energies to be disordered [3].

Similarly, another protein disorder predictor called FoldIndex looks at protein regions that are weakly hydrophobic and have high net charge, because these regions have little energy benefit from adopting an ordered state [4][5].

In addition to predictors like IUPred and FoldIndex, there exist predictors which combine the predictions of several ensemble predictors in order to increase accuracy. For example, Schlessinger et al. take a neural-network approach to disorder prediction using the raw output of four disorder predictors as well as protein sequence characteristics to achieve greater accuracy than any of the four individual predictors [6].

Mizianty et al. take a similar approach to disorder prediction, also utilizing four individual disorder predictors in addition to various sequence characteristics [7]. The architecture of their predictor is composed of an ensemble of three SVMs. One SVM was trained with a dataset containing both regions of long disorder ($\geq$30 residues) and short disorder ($<$30 residues), another was trained with a dataset containing only regions of long disorder, and a final SVM was trained with a dataset containing only regions of short disorder. The training data was derived from the individual predictors and sequence characteristics. Then the probability of a residue being disordered was a maximum over the three SVMs' outputs. This approach performed especially well in proteins with long disordered regions.

III. Dataset and features

There are two main classes of data on IDRs. The first class of data consists of information on protein disorder that is

collected from scientific literature. Data of this source is based on experimental evidence, and manually curated data on IDRs is found in the DisProt database.

Two other experimental sources indirectly provide data on protein disorder. The first indirect source is through x-ray crystallography experiments. X-ray crystallography experiments will try to resolve the structure of a protein but in some cases the position of a residue can't be determined. One explanation for this is that the residue is in a disordered region that isn't adopting a stable three-dimensional structure which can be resolved. Then, this residue is considered disordered.

The second indirect source is through NMR experiments. An NMR experiment will also try to resolve the structure, and often will have as a result multiple conformations of the protein in question. Then, if the amount of change in a residue position is large you would predict that residue to be a disordered one. MobiDB is a database which aggregates the information on NMR and x-ray crystallography experiments.

A dataset of 438 proteins which have annotations in scientific literature stored in the Disprot database and NMR/X-ray crystallography data in MobiDB was constructed. Then, order/disorder predictions were obtained for each residue in each protein using the 11 different IDR predictors, with each IDR predictor taking as input a protein sequence and returning an order/disorder prediction for each residue in the protein. Then, a feature vector was constructed for each residue in each protein where the dimension of the feature vector is the number of individual IDR predictors and each value in the feature vector is the continuous output of an IDR predictor. The construction of continuous feature vectors for each residue in a protein is demonstrated in Fig. 1 with 4 individual IDR predictors as an example. In reality there are 11 IDR predictors.

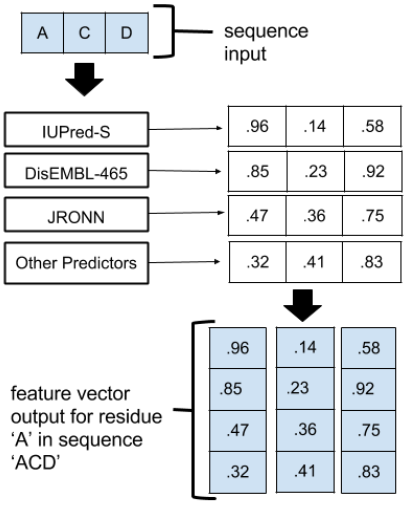

Fig. 1. Construction of feature vectors with continuous output

This yielded a dataset consisting of continuous feature vectors paired with their labels (experimentally determined order/disorder). A dataset consisting of binary feature vectors was also constructed, where each feature vector has dimension 11 and each value in the feature vector is either '1' or '0' where '1' represents a disorder prediction by and IDR predictor and '0' represents an order prediction by an IDR predictor. The binary vector was constructed through interpretation of the raw output. For example, if the raw output of a predictor is a probability of disorder then a probability of over one half would be translated to a '1' in the binary vector. One thing to note is that for the FoldIndex IDR predictor, the outputs of the web server for each residue was only a 0 or a 1. However, I still include the FoldIndex predictor in the continuous features dataset so as not to bias results towards ensemble methods that are trained on the binary dataset.

The ratio of ordered residues to disordered residues in the constructed dataset was 60:40, so there isn't a great imbalance to be adjusted with weighting during optimization. Ensemble methods will be evaluated using each of these datasets and the performance compared to the individual IDR predictors.

TABLE I. DATASET PROPERTIES FOR ENSEMBLE METHODS

Number of Examples	102,007
Number of Features	11
Ratio of Ordered:Disordered Residues	60:40

IV. METHODS

Two baseline models are constructed, one for the binary features dataset and another for the continuous features dataset. These baselines are respectively compared to the three additional models implemented with scikit-learn [10] and applied to both the binary features dataset and the continuous features dataset.

A. Baseline Models

The baseline ensemble models used in this work are majority rule voting and averaging. Majority rule voting is used with the binary feature dataset and averaging is used for the continuous feature dataset. In majority rule voting, an input feature vector containing binary predictions by each of the IDR predictors is classified as disordered if the majority of its features are ones.

For averaging, an input feature vector consisting of continuous predictions is predicted to be disordered if the average of those predictions is greater than half. For one disorder predictor (GlobPlot), the continuous values correspond to disorder propensities rather than to probabilities, so GlobPlot was excluded from the averaging baseline.

B. Logistic Regression

Logistic regression is applied to this binary order/disorder classification problem. Given a feature vector composed of either binary or continuous predictions for a residue in a particular protein, the output is a prediction that this residue is either ordered or disordered.

The regularized risk which must be minimized for logistic regression is given by:

$$J(\Theta) = \frac{1}{m}\sum_{i=1}^{m} log(1 + e^{-y^{(i)}\Theta^T x^{(i)}}) + \lambda\Theta^T\Theta$$

Here l2-regularization is used. Refer to the results section for details on hyperparameter tuning.

C. SVM

Logistic regression will only give linear hyperplanes, moving beyond this to cases where data is not linearly separable can be done with SVMs which produce nonlinear boundaries by transforming the data into a higher dimensional feature space. In this higher dimensional space, linear separation is possible. The transformation into a higher dimensional space is implicit in the kernel function that is chosen.

We may write the regularized risk for support vector machines as:

$$J_\lambda = \frac{1}{m}\sum_{i=1}^{m}\left[1 - y^{(i)}(K^{(i)})^T\alpha\right]_+ + \frac{\lambda}{2}\alpha^T K\alpha$$

Where $K^{(i)}$ is the ith column of kernel matrix K and λ is the regularization parameter. A larger λ represents a simpler model with less overfitting possible. A gaussian (rbf) kernel is used in this work:

$$K(x, x') = -\gamma \left\|x - x'\right\|^2$$

D. Random Forest

Random forest builds a set of de-correlated trees and combines their output in order to solve a classification or regression problem. The random forest algorithm used for classification is similar to that which is presented by Hastie et al. [11]:

1. For b = 1 to numForestTrees:
    a. Draw a bootstrap sample Z* of a set size from the dataset
    b. Grow a random-forest tree T_b using the bootstrapped data by repeating these steps for each leaf of the tree recursively, until a minimum node size is reached:
        i. Select m variables from the p variables
        ii. Pick the best variable among these to serve as the split point
        iii. Split the node into two daughter nodes
2. Now, you have an ensemble of trees T_b where b = 1… n.
3. On a new input feature vector x, you make a prediction by averaging the probability each tree T_b assigns to this feature vector (either binary or continuous) being ordered/disordered.

The difference from Hastie et al. is the use of averaging the trees' probability output for classification rather than using the majority vote of trees.

V. Results and Analysis

A. Experiments

For the logistic regression, random forest, and support vector machine models, hyperparameter tuning was performed. In all three cases hyperparameter tuning was conducted in isolation from the test data, with 30% of the data retained for testing. For logistic regression, l2-regularization outperformed l1-regularization. With random forest, the increase in test accuracy plateaued after roughly 25 trees for both the continuous feature dataset and the binary feature dataset (Fig. 2). Grid search over a regularization parameter, gamma from the radial basis function (refer to Methods), and the kernel function was effective in optimizing SVM performance, and especially on the continuous feature dataset (Table V).

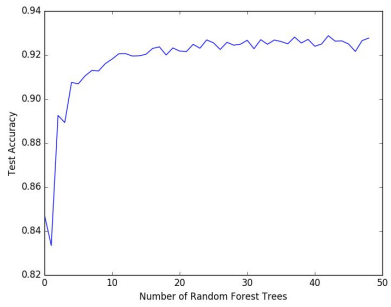

Fig. 2. Optimization of the number of random forest trees

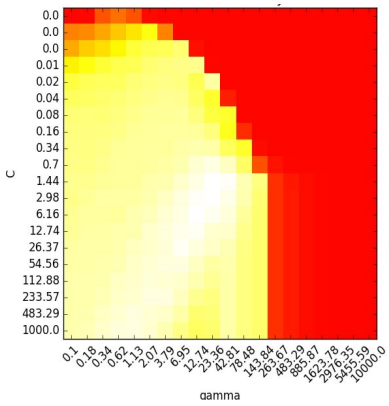

Fig. 3. Optimization of C and gamma for SVM with continuous features

B. Results

The different measures used to evaluate the performance of individual predictors and ensemble methods are sensitivity, specificity, and accuracy.

TABLE II. EVALUATION METRICS

Sensitivity	$\frac{TP}{TP+FN}$
Specificity	$\frac{TN}{TN+FP}$
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$

First the eleven different IDR predictors are evaluated with respect to these metrics (Table III).

TABLE III. PERFORMANCE OF THE PRIMARY PREDICTORS

Individual Predictor	***Sensitivity***	***Specificity***	***Accuracy***
DisEMBL-465	.4191	.9503	.7414
DisEMBL-hl	.5301	.7206	.6456
ESpritzD	.4197	.9574	.7459
EspritzN	.6862	.8081	.7966
EspritzX	.5714	.9553	.8043
FoldIndex	.6223	.7593	.7061
GlobPlot	.4073	.9100	.6952
IUPredL	.6442	.9167	.8056
IUPredS	.5496	.9409	.7870
JRONN	.7258	.8125	.7789
VSL2b	.8067	.7750	.7875

Next, the ensemble methods of majority voting, logistic regression, SVM, and random forest are evaluated on the binary features dataset. As a reminder, each feature vector has dimension eleven and in the binary feature dataset each individual predictors' continuous output is translated into a binary order/disorder prediction to construct a feature vector for each residue in each protein.

TABLE IV. ENSEMBLE PERFORMANCE ON BINARY FEATURE DATASET

Binary Dataset Ensemble Method	***Sensitivity***	***Specificity***	***Accuracy***
Majority Rule	.9257	.6520	.8181
Logistic Regression	.7013	.9153	.8301
SVM	.7230	.9142	.8379
Random Forest	.7125	.9250	.8415

Finally, the ensemble methods of averaging, logistic regression, SVM, and random forest were evaluated on the continuous features dataset.

TABLE V. ENSEMBLE PERFORMANCE ON CONTINUOUS FEATURE DATASET

Continuous Dataset Ensemble Method	***Sensitivity***	***Specificity***	***Accuracy***
Averaging	.9749	.4690	.7785
Logistic Regression	.7337	.9215	.8484
SVM	.8962	.9658	.9388
Random Forest	.8808	.9489	.9229

Overall, we see that ensemble methods tend to outperform the individual predictors as measured by accuracy. In fact, the only ensemble predictor which did not achieve a greater accuracy than every individual predictor in Table II was the averaging baseline using the continuous dataset. One possible reason for the poorer performance of this ensemble method relative to majority voting is the exclusion of GlobPlot due to its continuous values being disorder propensities rather than probabilities. Additionally, the baseline is quite naive in that it doesn't take into account the variance of individual predictors' outputs so predictors that are more often close to 0 or 1 in their probabilities will have a disproportionate effect on the average.

We also see that the models trained with the continuous dataset are better performing than their binary counterparts. The performance increase of using a continuous dataset as measured by accuracy is marginal using logistic regression (<2%) but significant (8-10%) with the nonlinear methods SVM and random forest.

The SVM model trained on the continuous features dataset was found to be the best performing model. It achieves a 13.32% improvement in accuracy over the most accurate primary predictor. Furthermore, it tops both the maximum sensitivity by 8.95% and the maximum specificity of any individual predictor by .84%.

The improvement in performance with ensemble methods is likely a result of eliminating the weaknesses of individual predictors. These weaknesses can stem from different sources of training data [12] and result in highly variable predicted levels of disorder across individual predictors [13].

Of course, the performance of an ensemble predictor is tied to this variability in the underlying predictors. For example, if all of the underlying predictors had few differences then a significant improvement in performance would likely not result. We visualize the relationship between our individual primary predictors using a heatmap of the grid where the value at (i,j) is the l2-norm of the difference between vectors i and j. Vectors i,j will respectively contain the continuous predictions of predictors i and j for every one of the amino acids. It's only possible to compare nine of the eleven predictors in this way

because the other two (FoldIndex and GlobPlot) don't output probabilities as continuous values.

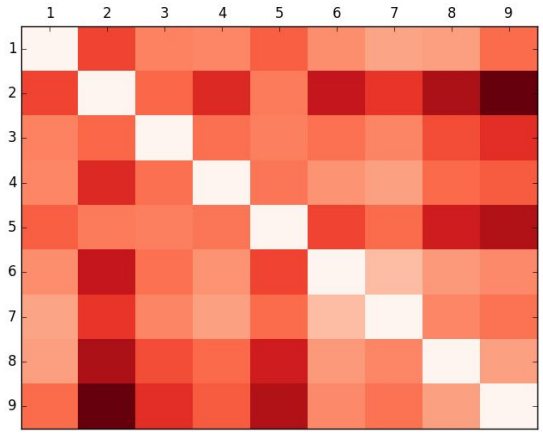

Fig. 4. Illustration of diversity of predictors

The labels 1-9 correspond to primary predictors DisEMBL-465, DisEMBL-hl, ESpritzD, ESpritzN, ESpritzX, IUPredL, IUPredS, JRONN, VSL2b. An area of further research will be building an ensemble predictor with less inputs by excluding one of each pair of similar primary IDR predictors from the ensemble.

VI. CONCLUSION AND FUTURE WORK

This project demonstrated the effectiveness of ensemble methods in protein disorder prediction utilizing 11 individual IDR predictors. Future work includes incorporating additional IDR predictors, incorporating protein properties to improve ensemble performance, and exploring different ensemble methods that can be applied in the field of protein disorder prediction.

ACKNOWLEDGEMENT

I would like to thank the BiocomputingUP Lab at the University of Padua for maintaining MobiDB and providing data on IDR annotations and predictions.

REFERENCES

[1] H. J. Dyson and P. E. Wright, "Intrinsically unstructured proteins and their functions," Nature Reviews Molecular Cell Biology, vol. 6, no. 3, pp. 197–208, 2005.

[2] B. He, K. Wang, Y. Liu, B. Xue, V. N. Uversky, and A. K. Dunker, "Predicting intrinsic disorder in proteins: an overview," Cell Research, vol. 19, no. 8, pp. 929–949, 2009.

[3] Z. Dosztanyi, V. Csizmok, P. Tompa, and I. Simon, "IUPred: web server for the prediction of intrinsically unstructured regions of proteins based on estimated energy content,"Bioinformatics, vol. 21, no. 16, pp. 3433–3434, 2005.

[4] Prilusky, C. Felder, T. Zeev-Ben-Mordehai, E. Rydberg, O. Man, J. Beckmann, I. Silman and J. Sussman, "FoldIndex(C): a simple tool to predict whether a given protein sequence is intrinsically unfolded", Bioinformatics, vol. 21, no. 16, pp. 3435-3438, 2005.

[5] R. van der Lee, M. Buljan, B. Lang, R. Weatheritt, G. Daughdrill, A. Dunker, M. Fuxreiter, J. Gough, J. Gsponer, D. Jones, P. Kim, R. Kriwacki, C. Oldfield, R. Pappu, P. Tompa, V. Uversky, P. Wright and M. Babu, "Classification of Intrinsically Disordered Regions and Proteins", Chemical Reviews, vol. 114, no. 13, pp. 6589-6631, 2014.

[6] A. Schlessinger, M. Punta, G. Yachdav, L. Kajan, and B. Rost, "Improved Disorder Prediction by Combination of Orthogonal Approaches," PLoS ONE, vol. 4, no. 2, Nov. 2009.

[7] M. J. Mizianty, W. Stach, K. Chen, K. D. Kedarisetti, F. M. Disfani, and L. Kurgan, "Improved sequence-based prediction of disordered regions with multilayer fusion of multiple information sources," Bioinformatics, vol. 26, no. 18, pp. i489–i496, Jul. 2010.

[8] M. Sickmeier, J. Hamilton, T. LeGall, V. Vacic, M. Cortese, A. Tantos, B. Szabo, P. Tompa, J. Chen, V. Uversky, Z. Obradovic and A. Dunker, "DisProt: the Database of Disordered Proteins", Nucleic Acids Research, vol. 35, no., pp. D786-D793, 2007.

[9] E. Potenza, T. Domenico, I. Walsh and S. Tosatto, "MobiDB 2.0: an improved database of intrinsically disordered and mobile proteins", Nucleic Acids Research, vol. 43, no. 1, pp. D315-D320, 2014.

[10] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and Duchesnay E. "Scikit-learn: Machine Learning in Python," Journal of Machine Learning Research, 12:2825–2830, 2011

[11] T. Hastie, R. Tibshirani, and J. Friedman, "Random Forests," in The elements of statistical learning, second edition: data mining, inference, and prediction, New York: Springer, 2009.

[12] L. P. Kozlowski and J. M. Bujnicki, "MetaDisorder: a meta-server for the prediction of intrinsic disorder in proteins," BMC Bioinformatics, vol. 13, no. 1, p. 111, 2012.

[13] J. Atkins, S. Boateng, T. Sorensen, and L. Mcguffin, "Disorder Prediction Methods, Their Applicability to Different Protein Targets and Their Usefulness for Guiding Experimental Studies," International Journal of Molecular Sciences, vol. 16, no. 8, pp. 19040–19054, 2015.