

2012

# Structure-based prediction of protein-protein interaction sites

Rafael Armando Jordan  
*Iowa State University*

Follow this and additional works at: <https://lib.dr.iastate.edu/etd>



Part of the [Bioinformatics Commons](#), and the [Computer Sciences Commons](#)

---

## Recommended Citation

Jordan, Rafael Armando, "Structure-based prediction of protein-protein interaction sites" (2012). *Graduate Theses and Dissertations*. 12357.

<https://lib.dr.iastate.edu/etd/12357>

This Dissertation is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Graduate Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact [digirep@iastate.edu](mailto:digirep@iastate.edu).

**Structure-based prediction of protein-protein interaction sites**

by

Rafael A. Jordan

A dissertation submitted to the graduate faculty  
in partial fulfillment of the requirements for the degree of  
DOCTOR OF PHILOSOPHY

Major: Computer Science

Program of Study Committee:

Vasant Honavar, Major Professor

Dreena Dobbs

David Fernández-Baca

Guang Song

Leslie Miller

Iowa State University

Ames, Iowa

2012

Copyright © Rafael A. Jordan, 2012. All rights reserved.

## DEDICATION

To Olga, my wife, and Nicole and Katherine, my daughters, whose love and support were fundamental to complete my thesis.

## TABLE OF CONTENTS

<b>LIST OF TABLES</b> . . . . .	vii
<b>LIST OF FIGURES</b> . . . . .	xii
<b>ACKNOWLEDGEMENTS</b> . . . . .	xviii
<b>ABSTRACT</b> . . . . .	xix
<b>CHAPTER 1. GENERAL INTRODUCTION</b> . . . . .	1
1.1 Introduction . . . . .	1
1.1.1 Protein-protein interaction sites and interface residues . . . . .	1
1.1.2 Prediction of interaction sites using machine learning . . . . .	3
1.1.3 Challenges for predicting protein-protein interaction sites . . . . .	5
1.2 Research aims . . . . .	7
1.3 Dissertation organization . . . . .	8
<b>CHAPTER 2. <i>ProtInDb</i>: A DATA BASE OF PROTEIN-PROTEIN IN-</b> <b>TERFACE RESIDUES</b> . . . . .	11
2.1 Abstract . . . . .	11
2.2 Introduction . . . . .	12
2.3 Databases and servers of protein-protein and domain-domain interfaces . . . . .	14
2.4 Materials and Methods . . . . .	15
2.4.1 Biological assemblies and asymmetric units. . . . .	15
2.4.2 Protein-protein interface residues. . . . .	16
2.4.3 Protein surface residues. . . . .	16
2.4.4 Data collection. . . . .	16
2.4.5 Implementation details. . . . .	18

2.5	Results and Discussion . . . . .	19
2.5.1	User interface. . . . .	19
2.5.2	Updates and content. . . . .	22
2.5.3	Examples of applications that use <i>ProtInDb</i> . . . . .	24
2.6	Summary . . . . .	25
2.7	Availability and requirements . . . . .	26
2.8	List of abbreviations . . . . .	26
2.9	Author's contributions . . . . .	26
2.10	Acknowledgments and funding . . . . .	26
<b>CHAPTER 3. A MODULAR APPROACH TO PREDICT PROTEIN-PROTEIN</b>		
<b>INTERACTION SITES . . . . .</b>		
3.1	Abstract . . . . .	27
3.2	Background . . . . .	28
3.3	Methods . . . . .	31
3.3.1	Surface residues . . . . .	31
3.3.2	Surface patches . . . . .	31
3.3.3	Representative interface patches . . . . .	32
3.3.4	Datasets . . . . .	32
3.3.5	Prediction of protein-protein interaction sites ( <i>PoInterS</i> ) . . . . .	33
3.3.6	Prediction of interface residues . . . . .	35
3.3.7	Performance evaluation . . . . .	37
3.4	Experiments and results . . . . .	40
3.4.1	Cross-validation experiments . . . . .	40
3.4.2	Validation with the ZDOCK dataset . . . . .	47
3.4.3	Comparison with other interface patches predictors . . . . .	52
3.4.4	Web server . . . . .	58
3.5	Conclusions . . . . .	58
3.6	List of abbreviations . . . . .	60
3.7	Authors contributions . . . . .	60

3.8	Acknowledgements . . . . .	60
<b>CHAPTER 4. PREDICTING PROTEIN-PROTEIN INTERFACE RESIDUES</b>		
	<b>USING LOCAL SURFACE STRUCTURAL SIMILARITY . . . . .</b>	<b>61</b>
4.1	Abstract . . . . .	61
4.2	Background . . . . .	62
4.3	Methods . . . . .	64
4.3.1	Structural elements and their representation . . . . .	64
4.3.2	Distance between histogram of atom nomenclatures . . . . .	65
4.3.3	Repository of structural elements . . . . .	66
4.3.4	Retrieving similar structural elements . . . . .	66
4.3.5	<i>PrISE</i> algorithm . . . . .	67
4.3.6	Datasets . . . . .	70
4.3.7	Performance Evaluation . . . . .	71
4.4	Results and discussion . . . . .	72
4.4.1	Comparison of <i>PrISE<sub>L</sub></i> , <i>PrISE<sub>G</sub></i> and <i>PrISE<sub>C</sub></i> . . . . .	72
4.4.2	Impact of homologs of the query protein on the quality of predictions . .	73
4.4.3	Comparison with two prediction methods based on geometric-conserved local surfaces . . . . .	74
4.4.4	Comparison with a prediction method based on protein structural similarity	75
4.4.5	Comparison with other prediction methods . . . . .	80
4.4.6	Prediction performances in the absence of similar proteins . . . . .	85
4.5	Conclusions . . . . .	85
4.6	Competing interests. . . . .	88
4.7	Author's contributions. . . . .	88
4.8	Acknowledgements. . . . .	88
<b>CHAPTER 5. CONCLUSIONS . . . . .</b>		
5.1	Future work . . . . .	92
5.1.1	Extending <i>ProtInDb</i> . . . . .	92

5.1.2	Prediction of interface residues between different macro-molecules . . . . .	92
5.1.3	Using PrISE and PoInterS to assist biological experiments . . . . .	92
5.1.4	Creation of more sophisticated methods to retrieve similar structural elements in PrISE . . . . .	93
5.1.5	Partner-specific versions of <i>PrISE</i> and <i>PoInterS</i> . . . . .	93
5.1.6	Searching for proteins with similar structure . . . . .	94
<b>APPENDIX A. SUPPLEMENTARY MATERIAL FOR CHAPTER 4 . . . . .</b>		<b>95</b>
A.1	Atom nomenclatures . . . . .	95
A.2	Retrieving similar structural elements - additional details . . . . .	95
A.3	Tuning method . . . . .	97
A.3.1	Tuning dataset . . . . .	97
A.3.2	Representative set of similar structural elements . . . . .	98
A.3.3	Performance tuning . . . . .	98
A.4	Selection of a threshold value for performing classification . . . . .	103
A.5	Additional comparisons of <i>PrISE<sub>L</sub></i> , <i>PrISE<sub>G</sub></i> and <i>PrISE<sub>C</sub></i> . . . . .	103
A.6	Additional evaluation of the impact of homologs of the query protein in the predictions . . . . .	106
A.7	Additional comparison with two prediction methods based on geometrical conserved local surfaces . . . . .	106
A.8	Abbreviations . . . . .	109
<b>BIBLIOGRAPHY . . . . .</b>		<b>111</b>

## LIST OF TABLES

2.1	<p><b>Summary of the information stored in <i>ProtInDb</i> at March 3, 2012.</b> The row labeled “Number of protein subunits” indicates the number of protein chains in asymmetric units and the number of protein chains with unique name in biological assemblies (i.e. every chain in a biological assembly is counted once even if it has several copies in the assembly). . . . .</p>	22
3.1	<p><b>Cross-validation results using sequence-based features in windows of 9 residues.</b> The <i>ranking</i> column refers to the method using to rank the patches. The last four columns show the number of successful predictions and their corresponding percentage in reference to the 220 monomers in the dataset. “<i>Patch 1</i>” refers to the results when only the top ranked patch is considered. Similarly, “<i>Patch 2</i>” and “<i>Patch 3</i>” refer to the evaluation using the second and third top-ranked patches respectively. The table has been divided on four blocks depending on the sampling technique used on the training dataset and indicated in the third column. . . . .</p>	42



- 3.2 **Cross-validation results using structure-based features in windows of 9 residues.** The experiments were performed on 220 monomers using patches of size  $1.91n^{0.55}$ . Each of the last four columns show the number and the percentage of successful predictions in reference to the 220 monomers. The table has been divided in four blocks depending on the sampling technique used for the training dataset. Column *Norm* indicates whether the numerical data was or not normalized. Other column labels are explained in Table 3.1. . . . . . 44
- 3.3 **Performances of the interface residues classifiers and their corresponding interface patches predictors.** “*SPP<sup>†</sup>*” refers to *SPPIDER* predictors trained with 10 datasets generated by partitioning the CV dataset into 10 pieces (i.e. each sample in CV appears in exactly one dataset). “*SPP*” denotes *SPPIDER* classifiers trained with 10 datasets generated from 10-cross validation partitions on the CV dataset (i.e. each sample of CV appears in nine training datasets). “*Train. balanced?*” indicates whether the training dataset was balanced or not. “*Train. dimers?*” and “*Test dimers?*” indicate whether the interface residues in the training and testing datasets, respectively, were extracted from dimers in the query protein or from the interacting chains in complexes with sequence identity  $\geq 96\%$  with the query protein. “*IR*” refers to interface residues and “*NIR*” to non-interface residues. “*Overlap %*” is the percentage of correct predictions for the interface patches predictor. “*Overlap Patch 1 %*” is the percentage of correctly predicted patches when only the top-ranked patch is considered. These experiment were performed with patches of size  $1.91n^{0.55}$  ranked with  $probDis_s(p)$ . Windows of nine residues were used for the SVM and LR classifiers. The data is presented according to the performance of the interface patches predictors.

3.4	<b>Comparison of <i>SHARP</i><sup>2</sup> and <i>PoInterS-SVM</i>.</b> <i>PoInterS-SVM</i> ranked patches of size $1.91n^{0.55}$ using $probDis_s(p)$ . <i>Overl</i> , <i>Spec</i> and <i>Sens</i> refers to <i>overlap</i> , <i>specificity</i> , and <i>sensitivity</i> respectively. <i>Patch</i> refers to the first patch with $overlap \geq 70\%$ or to the best among the three top ranked patches if their $overlap < 70\%$ . The probability of randomly select a patch with $overlap \geq 70\%$ is denoted by $p$ , and <i>Prob</i> is the probability of randomly finding the patch specified in the column <i>Patch</i> . . . . .	54
3.5	<b>Comparison using the first patch predicted with <i>PPI-Pred</i> and the first patch predicted with <i>PoInterS-SVM</i>.</b> <i>Patch size</i> refers to the number of residues in the top-ranked patch computed by <i>PPI-Pred</i> and used in <i>PoInterS-SVM</i> . The probability of randomly selected a patch with $overlap \geq 70\%$ is denoted by $p$ . <i>Overl</i> , <i>Spec</i> and <i>Sens</i> refers to <i>overlap</i> , <i>specificity</i> , and <i>sensitivity</i> respectively. Patches were ranked in <i>PoInterS</i> using $probDis_s(p)$ . . . . .	57
4.1	<b>Performance of different methods on the DS24Carl dataset.</b> Performance measures are computed as the average on the set of 24 proteins. Precision and recall values for Carl08 and Carl10 were taken from (26) and (27) respectively. . . . .	75
4.2	<b>Evaluation of <i>PrISE<sub>C</sub></i> and <i>PredUs</i> on DS56bound using different performance measures.</b> The table is divided into two sections depending on which proteins are excluded from the set of similar structures (First column). . . . .	78
4.3	<b>Evaluation of <i>PrISE<sub>C</sub></i> and <i>PredUs</i> on DS56unbound using different performance measures.</b> . . . . .	81

4.4	<b>Evaluation on the datasets DS56bound and DS56unbound.</b> “ <i>PrISE<sub>C</sub></i> spe.” refers to the performance computed after filtering out from the repository samples extracted from homologs from the same species. “ <i>PrISE<sub>C</sub></i> hom.” indicates that samples extracted from homologs were not considered in the prediction process. . . . .	84
4.5	<b>Evaluation on 187 proteins from DS188.</b> “ <i>PrISE<sub>C</sub></i> spe.” refers to the performance computed after excluding from the prediction process samples extracted from homologs of the same species that the query proteins. “ <i>PrISE<sub>C</sub></i> hom.” indicates that samples extracted from homologs were filtered out from the repository. . . . .	85
A.1	<b>Atom nomenclatures used to build the histograms of atom nomenclatures.</b> . . . . .	95
A.2	<b>List of the 50 protein chains included in the tuning dataset.</b> . . . . .	97
A.3	<b>Performance of different methods on the DS24Carl dataset.</b> Performance measures are computed as the average on the set of 24 proteins. Precision and recall values for Carl08 and Carl10 were taken from (26) and (27) respectively. Samples derived from homologs of the query proteins were excluded from the ProtInDb repository. . . . .	109
A.4	<b>Performance of <i>PrISE</i> predictors using different repositories of structural elements and excluding homologs.</b> Performance measures are computed as the average on the set of 24 proteins in the DS24Carl dataset. Samples extracted from homologs (without regarding the species) were excluded from the prediction process. The column “ProtInDb” indicates whether samples were extracted from the ProtInDb repository (marked with a tick), or from the $ProtInDb \cap PQS$ repository.	110

A.5	<b>Performance of <i>PrISE</i> methods using different repositories and excluding homologs of the same species.</b> The performance measures were computed as the averages on the proteins in the DS24Carl dataset. Samples extracted from homologs from the same species than the query proteins were filtered out from the prediction process. The “ProtInDb” column indicates whether the samples were extracted from the ProtInDb repository (marked with a tick), or from the $ProtInDb \cap PQS$ repository. . . . .	110
-----	--	-----

## LIST OF FIGURES

2.1	<b>Flow diagram of the data collection process for a protein.</b> . . . .	17
2.2	<b>Example of the output corresponding to the visualization of interface residues.</b> Interface residues of the subunit A in protein PDB:2f03 are shown in different colors. White spheres indicate non-interface atoms. Atoms in interface residues are colored according to the amino acid to which they belong. . . . .	21
2.3	<b>Screen shot of the Web interface used to generate datasets of protein-protein interface residues.</b> . . . . .	23
3.1	<b>Flow diagram of the <i>PoInterS</i> prediction method.</b> . . . . .	34
3.2	<b>Prediction results using different window sizes for the sequence-based LR predictor.</b> These results correspond to experiments with the cross-validation dataset using patches of size $1.91n^{0.55}$ . “Patch 1” refers to the results obtained considering the top-ranked patch. . . . .	43
3.3	<b>Prediction results using different window sizes for the best structure-based LR classifier.</b> . . . . .	45

3.4	<b>Example of prediction of interface residues using structure and sequence based protein interface residue predictors.</b> Correct predictions are shown in green and white (TP and TN respectively) whereas incorrect predictions are displayed in red and yellow (FP and FN respectively). Sequence-based predictions are presented on the left and structure-based predictions are displayed on the right. Predictions were performed on the metalloenzyme pyruvate: ferredoxin oxidoreductase (PDB:1KEK, chain A). . . . .	46
3.5	<b><i>PoInterS</i> prediction results using the structure-based LR classifier on two patch sizes.</b> These results were obtained using a window of size nine. . . . .	46
3.6	<b>Prediction results for the structure-based LR classifier using four different schemes to rank patches.</b> The predictions were performed based on LR PPIRP, using patches of size $1.91n^{0.55}$ , and structural windows of size nine. . . . .	47
3.7	<b>Comparison of percentages of correct predictions using the ZDOCK and the cross validation datasets.</b> CV refers to the results of the leave-one-protein-out experiments in which the LR were trained and tested using the cross validation dataset. ZDOCK refers to results obtained using structure-based LR and SVM PPIRPs trained with the cross-validation dataset and tested on the ZDOCK dataset. The PPIRPs were built using structural windows of size 9, and normalizing the data. Patches were ranked using the probabilities generated by the PPIRP. Results are grouped into two sections corresponding to different patch sizes. . . . .	49
3.8	<b>Results of <i>PoInterS</i> predictions on the ZDOCK dataset using different ranking schemes.</b> These experiments were performed with normalized data, structure-based PPIRPs using a window with nine residues, and patches of size $1.91n^{0.55}$ . . . . .	49

3.9	<b>Overlap curves for <i>PoInterS-SVM</i> and the best <i>SPPIDER</i>-based predictor of interface patches.</b> Predictions were performed using windows of nine residues and ranking patches of size $1.91n^{0.55}$ using $ProbDis_s(p)$ . . . . .	52
3.10	<b>Comparison of <i>SHARP</i><sup>2</sup> and <i>PoInterS-SVM</i> using <i>overlap curves</i>.</b> The horizontal axis indicate different <i>overlap</i> percentage values whereas the vertical axis shows the number of correct predictions achieved according to the <i>overlap</i> percentage value. . . . .	55
3.11	<b>Comparison of <i>PPI-Pred</i> and <i>PoInterS-SVM</i> using <i>overlap curves</i>.</b> The curves were generated using the predictions corresponding to the top-ranked patch for each protein in the dataset. . . . .	58
4.1	<b>Prediction of interface residues using surface structural similarity.</b> . . . . .	68
4.2	<b>Comparative performances of <i>PrISE<sub>L</sub></i>, <i>PrISE<sub>G</sub></i>, <i>PrISE<sub>C</sub></i>, and randomly generated predictions on the DS188 dataset.</b> . . . . .	73
4.3	<b>Comparison of schemes for filtering out similar proteins from the prediction process.</b> This experiment was performed using <i>PrISE<sub>C</sub></i> with the DS188 dataset. . . . .	74
4.4	<b>Comparison of <i>PredUs</i> and <i>PrISE<sub>C</sub></i> using the dataset DS188, derived from the docking benchmark 3.0.</b> (A) performance of predictions from which homologs from the same species were not used to compute the structural neighbors and the samples used in <i>PredUs</i> and <i>PrISE</i> respectively. (B) performance of predictions that did not consider homologs. Both images show results for the 181 proteins that were predicted by <i>PredUs</i> and <i>PrISE<sub>C</sub></i> and for the 188 proteins predicted by <i>PrISE<sub>C</sub></i> . . . . .	77

- 4.5 **Comparison of *PrISE<sub>C</sub>* and *PredUs* using the dataset DS56bound, derived from CAPRI.** The results in (A) correspond to predictions in which homologs from the same species were excluded from the collection of samples and the set of structural neighbors. The results in (B) were obtained excluding homologs from the sets of similar structures. . . . . 79
- 4.6 **Comparison of *PrISE<sub>C</sub>* and *PredUs* using the DS56unbound dataset , derived from CAPRI.** (A) shows the performance achieved after removing homologs from the same species from the set of similar structures in *PredUs* and *PrISE<sub>C</sub>*. (B) shows the performances when homologs are excluded. The suffixes 53 and 56 indicate the number of proteins that were used in the experiment. . . . . 80
- 4.7 **Performance of different classifiers evaluated on the DS56bound (A) and the DS56unbound (B) datasets.** For the *PrISE* classifiers, “spe.” and “hom.” show predictions in which samples extracted from homologs from the same specie and homologs, respectively, has been excluded from the prediction process. . . . . 82
- 4.8 **Precision-recall curves of different classifiers evaluated on 187 proteins from the DS188 dataset.** For the *PrISE* classifiers, “spe.” and “hom.” show predictions in which homologs from the same species and homologs, respectively, has been excluded from the repository of structural elements. . . . . 83



4.9	<b>Performance computed in absence of similar proteins at different similarity levels.</b> Figures (A) and (B) show the precision recall curves computed after excluding from the sets of similar structures homologs (without regarding the species) sharing $\geq 95\%$ of sequence identity with the query proteins. Similarly, figures (C) and (D) show the performances after excluding proteins sharing $\geq 50\%$ sequence identity, and (E) and (F) display the results after filtering out proteins with sequence identity $\geq 30\%$ . The precision-recall curves corresponding to the DS56bound dataset are shown at (A), (C), and (E), and the results computed using the DS56Unbound dataset are labeled as (B), (D), and (F). Figures (E) and (F) were computed using 55 and 52 proteins respectively given that PredUs could not find structural elements for the protein chain lynt-L. . . . .	86
A.1	<b>Prediction results using majority vote on the top 50 samples according to different definitions of distance between histogram of atom nomenclatures.</b> . . . . .	99
A.2	<b>Prediction results using majority vote with different number of unweighted samples.</b> . . . . .	100
A.3	<b>Prediction results with different number of samples and using majority vote on samples weighted using the city block distance between histogram of atom nomenclatures.</b> . . . . .	101
A.4	<b>Prediction results using different number of samples and general contribution (i.e. <math>PrISE_G</math>).</b> . . . . .	102
A.5	<b>Prediction results using set of samples of different size and local contribution (i.e. <math>PrISE_L</math>).</b> . . . . .	103
A.6	<b>Prediction results using number of samples and combined contribution (i.e. <math>PrISE_C</math>).</b> . . . . .	104

A.7	<b>Prediction results using different weighting schemes. The number in the labels indicates the number of samples used for prediction. . . . .</b>	<b>104</b>
A.8	<b>Comparison of <math>PrISE_L</math>, <math>PrISE_G</math>, and <math>PrISE_C</math> using the dataset DS24Carl. . . . .</b>	<b>105</b>
A.9	<b>Comparison of <math>PrISE_L</math>, <math>PrISE_G</math>, and <math>PrISE_C</math> using the dataset DS56Bound. . . . .</b>	<b>105</b>
A.10	<b>Comparison of <math>PrISE_L</math>, <math>PrISE_G</math>, and <math>PrISE_C</math> using the dataset DS56Unbound. . . . .</b>	<b>106</b>
A.11	<b>Example of the scores generated by <math>PrISE_L</math>, <math>PrISE_G</math>, and <math>PrISE_C</math>.</b> This figure show (in the vertical axis) the score generated by $PrISE_L$ , $PrISE_G$ , and $PrISE_C$ as well as the actual interface residues for the first 28 residues (shown in the horizontal axis) in the sequence of the protein chain 1ohz-B. The horizontal red line signals the threshold computed on the scores (0.34) to differentiate between interfaces and non-interfaces. . . . .	<b>107</b>
A.12	<b>Performance of <math>PrISE_C</math> with DS24Carl using three schemes for excluding similar proteins. . . . .</b>	<b>108</b>
A.13	<b>Performance of <math>PrISE_C</math> with DS56Bound using different schemes for excluding similar proteins. . . . .</b>	<b>108</b>
A.14	<b>Performance of <math>PrISE_C</math> with DS56Unbound using several schemes for excluding similar proteins. . . . .</b>	<b>109</b>

## ACKNOWLEDGEMENTS

I want to express my gratitude to those who helped me to conduct my research: First, to Dr. Vasant Honavar for his guidance, support, and valuable discussions that helped me to define and execute my thesis research; To Dr. Drena Dobbs for her feedback and support; To members of my committee: Dr. David Fernández-Bacca, Dr. Guang Song, and Dr. Leslie Miller, for their feedback on the research proposal and the thesis; To Li Xue, Dr. Yasser El-Manzalawy, Dr. Fadi Towfic, Rasna Walia, Dr. Feihong Wu, and other members of the Artificial Intelligence Research laboratory, the Center for Computational Intelligence, Learning, and Discovery, and the Dobbs Lab, for intellectually stimulating discussions, suggestions, and support; To Dr. Jorge-Francisco Estela, Dr. Andrés Jaramillo-Botero, and Dr. Diego-Luis Linares for encouraging me to pursue doctoral studies and supporting me along the way; and finally, to my father, Rafael Jordan-Sánchez, my mother María-del-Rosario Osorio, and the rest of my family, for their love and encouragement through many years that I have been away from them while pursuing my PhD studies.

## ABSTRACT

Protein-protein interactions play a central role in the formation of protein complexes and the biological pathways that orchestrate virtually all cellular processes. Reliable identification of the specific amino acid residues that form the interface of a protein with one or more other proteins is critical to understanding the structural and physico-chemical basis of protein interactions and their role in key cellular processes, predicting protein complexes, validating protein interactions predicted by high throughput methods, and identifying and prioritizing drug targets in computational drug design. Because of the difficulty and the high cost of experimental characterization of interface residues, there is an urgent need for computational methods for reliable predicting protein-protein interface residues from the sequence, and when available, the structure of a query protein, and when known, its putative interacting partner.

Against this background, this thesis develops improved methods for predicting protein-protein interface residues and protein-protein interfaces from the three dimensional structure of an unbound query protein without considering information of its binding protein partner. Towards this end, we develop (i) *ProtInDb* (<http://protindb.cs.iastate.edu>), a database of protein-protein interface residues to facilitate (a) the generation of datasets of protein-protein interface residues that can be used to perform analysis of interaction sites and to train and evaluate predictors of interface residues, and (b) the visualization of interaction sites between proteins in both the amino acid sequences and the 3D protein structures, among other applications; (ii) *PoInterS* (<http://pointers.cs.iastate.edu/>), a method for predicting protein-protein interaction sites formed by spatially contiguous clusters of interface residues based on the predictions generated by a protein interface residue predictor. *PoInterS* divides a protein surface into a series of patches composed of several surface residues, and uses the outputs of the interface residue predictors to rank and select a small set of patches that are the most likely to constitute the interaction sites; and (iii) *PrISE* (<http://prise.cs.iastate.edu/>), a method for predicting

protein-protein interface residues based on the similarity of the structural element formed by the query residue and its neighboring residues and the structural elements extracted from the interface and non-interface regions of proteins that are members of experimentally determined protein complexes. A structural element captures the atomic composition and solvent accessibility of a central residue and its closest neighbors in the protein structure. *PrISE* decomposes a query protein into a set of structural elements and searches for similar elements in a large set of proteins that belong to one or more experimentally determined complexes. The structural elements that are most similar to each structural element extracted from the query protein are then used to infer whether its central residue is or is not an interface residue. The results of our experiments using a variety of benchmark datasets show that *PoInterS* and *PrISE* generally outperform the state-of-the-art structure-based methods for predicting interaction patches and interface residues, respectively.

## CHAPTER 1. GENERAL INTRODUCTION

### 1.1 Introduction

#### 1.1.1 Protein-protein interaction sites and interface residues

Proteins are fundamental to virtually every process in the cell, including catalysis in biochemical reactions, conformations of the structure of cells and tissues, and control of cellular processes such as DNA replication and signal transduction. To perform their functions, proteins interact with other molecules such DNA, RNA, or other proteins or ligands. The binding sites that define the interaction between proteins are known as interaction sites, and they are composed of a set of amino acid residues, known as interface residues, that form a chemical bond with a component of another molecule. The identification of interaction sites or interface residues can lead to advances in problems such as prediction and validation of protein-protein interactions (7; 170; 13; 165; 150; 208), protein docking (48; 71; 135; 139; 173), identification of hot-spot residues (6; 114; 184), understanding of disease pathways (102; 100; 164), and development of new drugs (189; 197; 5; 215).

Interaction sites and interface residues can be experimentally identified using different methods. Some of the most commonly used methods are:

- X-ray crystallography (17). This method allows scientists to estimate the position of the atoms of a protein from the analysis of the diffracted angles and intensities of X-ray beams applied to a crystallized protein. As a consequence, X-ray crystallography is unsuitable for determining the structures of proteins difficult to crystallize (e.g. several proteins in the cellular membranes, or proteins in some transient complexes) (148). Furthermore, some conformations derived from protein crystals are not biologically relevant (216), which introduces false positives in the set of determined interface residues. Despite these limita-

tions, X-ray crystallography methods have been used to determine the structure of more than 87%<sup>1</sup> of the proteins deposited in the Protein Data Bank (PDB) (12).

- Nuclear magnetic resonance (NMR) spectroscopy (55; 216). Determination of molecular structures using NMR spectroscopy is based on the absorption of different radio frequencies (i.e. resonance) by molecules exposed to a strong magnetic field. This method can be used to determine the structure of proteins in a solution. The solution is analyzed in a NMR spectrometer that measures the nuclear magnetic resonance of protons and some carbon and nitrogen atoms to identify atoms in different amino acids in the protein sequence. Then, the resonance of different atoms is perturbed to infer the internuclear distances between close atoms, which allow modeling the position of the atoms in the protein. Unlike X-ray crystallography, NMR spectroscopy can produce different models of a protein, which provides some insights into its dynamics. NMR spectroscopy methods are generally used to determine the structure of proteins with molecular weight lower than 50 kDa (152) and low to moderate flexibility (66). Around 11% of the proteins structures deposited in the PDB have been determined using NMR as of February 2012.
- Site-specific mutagenesis (196). Using this technique it is possible to identify a subset of interface residues responsible for the stability of protein complexes. This is performed by introducing mutations to specific base pairs in the DNA and evaluating the impact of the mutations on the stability of known protein complexes that contain the protein derived from the mutated DNA. The residues responsible for maintaining the stability in protein complexes are called hot-spot residues.
- Chemical cross-linking and mass spectrometry (160; 8). Using this method, a purified and tagged protein complex is subject to a cross-linking reaction to form cross-links (i.e. covalent bonds that link two proteins in a complex), that can be identified using mass spectrometry. Chemical-cross linking and mass spectrometry can be used to generate low resolution protein structures (174) and to identify interaction sites in transient complexes (128).

---

<sup>1</sup>Information extracted from <http://www.pdb.org/pdb/statistics/holdings.do> on February, 2012.

Given the limitations and the expensive and labor-intensive nature of these methods (53), there is an urgent need for developing computational tools useful for predicting protein-protein interaction sites.

### 1.1.2 Prediction of interaction sites using machine learning

Machine learning techniques provide cost-effective approaches for performing computational prediction of protein-protein interface residues and interaction sites - see reviews in (91; 213; 9; 43; 11; 53; 183; 56). A machine learning-based predictor of interaction sites is a function  $y = h(x)$  that predicts whether a protein site  $x$  is or is not an interaction site. The range of the predicted value  $y$  determines whether the predictor  $h$  is a regression function (i.e.  $y \in \mathbb{R}$ ) or a classification function (i.e.  $y \in \{true, false\}$ ). A regression function generates a score  $y$  that is used to predict whether  $x$  is or is not an interaction site whereas a classifier directly predicts whether  $x$  is or is not an interaction site. Different machine learning algorithms have been used to build predictors of protein-protein interaction sites. Some examples include artificial neural networks (214; 54; 145; 31; 153; 155; 11), support vector machines (204; 205; 206; 19; 22; 35; 190; 192; 200; 47; 155; 191; 70; 116; 44; 121; 124; 123; 210), hidden Markov support vector machines (122), Bayesian networks (205; 21), Naive Bayes (193; 133), conditional random fields (115; 74), random forest (20; 32; 172; 159), clusters (69; 171; 212; 131; 202), and ensemble methods that combine the results of different predictors (168; 138; 158; 191; 44; 41). More details of some of these methods are given in the following chapters.

The construction of a predictor generally involves a process in which the function  $h$  is built from a training dataset, and evaluated using a different testing dataset. In the case of protein-protein interaction sites predictors, these datasets are generally extracted from the Protein Data Bank (PDB) (12), that stores macromolecular structural data that is free and open to the community. Using the three-dimensional position of the atoms in a protein complex deposited in PDB, a user can compute the set of amino acid residues in the interaction sites between every pair of proteins in the complex. Therefore, a dataset of protein-protein interface residues can be defined as a set of pairs  $(x, y)$ , where  $x$  represents a protein site and  $y$  represents the interface/non-interface label associated with the site  $x$ .



A protein site is generally described using features that are useful to discriminate between interaction and non-interaction sites (91; 180; 93; 181; 29; 40; 201; 43; 207; 11; 53; 183; 50; 130). Such features can be derived from the protein sequence (e.g. propensity of the residues to be part of interaction sites, hydrophobicity, electrical charge), the protein structure (e.g. solvent accessible surface area, B-factor, secondary structure, protrusion), or from data derived from conservation analysis (e.g. profiles generated from multiple sequence alignments). Given that no single feature has been found to be sufficient to perform prediction of protein-protein interaction sites (53), it is common to represent a site using a combination of different features.

Methods for predicting protein-protein interaction sites can be divided in those that represent information using the protein sequence and those that represent information using the protein structure (56). Methods based on protein sequences (145; 204; 205; 206; 190; 200; 77; 81; 70; 32; 44; 172; 133; 202) generally represent each amino acid using features derived from it and its neighbors in the sequence. Therefore, an amino acid  $a_i$  may be represented as a tuple  $(a_{i-k}, \dots, a_{i-1}, a_i, a_{i+1}, \dots, a_{i+k})$ , where  $a_{i-j}$  represents a feature associated with the  $j$ -th residue before residue  $a_i$  in the sequence. Methods based on protein structures (54; 31; 24; 35; 115; 155; 158; 191; 116; 121; 122; 124; 147; 172; 80; 123; 210; 211) represent each amino acid using information of its closest residues in the structure. For example, amino acid  $a_i$  can be represented as the tuple  $(a_i, n_1, n_2, \dots, n_l)$  where  $n_j$  represents the closest  $j$ -th residue to  $a_i$  according to metrics such as the Euclidean distance between the closest atoms of residues  $a_i$  and  $a_j$ . The number of neighbors to use in the sequence or structure representations are commonly determined using a grid search approach. The main advantage of sequence-based prediction methods is that the number of known protein sequences is much larger than the number of known protein structures, which potentially allows sequence-based methods to predict interaction sites for a larger number of proteins than structure-based methods. The main advantage of structure-based prediction methods is that they can use more information (derived from the protein structures) than sequence-based methods, which make them very attractive for predicting interaction sites.

Predictors of interaction sites can also be divided into predictors of protein-protein interface residues (214; 54; 57; 31; 24; 35; 113; 110; 115; 155; 158; 191; 79; 116; 165; 11; 49; 122; 121;

124; 172; 107; 123), and predictors of protein-protein interaction patches (94; 22; 119; 132; 90). Predictors of interface residues label each amino acid residue in the protein as belonging to or not belonging to the interaction site. Predictors of interaction patches divide the surface of the protein into patches, generally composed of a central residue and its neighbors in the structure, and select a small set of patches with the highest probability of belonging to the interaction site, allowing researchers to focus on specific sites of a protein.

Of particular interest are recent methods based on similarity between proteins or protein regions, given that the likelihood of success of such methods increases as the number of determined protein structures growth. These methods are motivated by observations suggesting that interaction sites tend to be conserved among proteins with similar structures (35; 182; 39; 69). A limitation of the methods based on proteins with similar structure or sequence to that of a query protein (212; 202; 211) is that they can generate predictions only when similar proteins are found. Methods based on the similarity between protein regions (106; 26; 27; 107) overcome this limitation, but they are computationally intensive and have low predictive performance in comparison with methods based on similarity between proteins.

### 1.1.3 Challenges for predicting protein-protein interaction sites

The creation and evaluation of methods for predicting protein-protein interface residues involve different challenges. Most of these challenges are due to the nature of the data, the complexity of the processes required to build and evaluate reliable predictive models, and the difficulty to objectively compare different predictors. Some of these challenges are presented in this subsection.

The nature of the data required to perform prediction of protein-protein interaction sites impose some challenges for constructing reliable predictors. An initial problem is that some types of protein structures are underrepresented in the PDB, which can result in biased predictions. For example, proteins in cellular membranes represent around 30% of the proteome but account for about 1% of the structures deposited in the PDB (46; 142). Similarly, some protein interactions are underrepresented in the PDB, mainly due to the difficulty of experimentally determining some complexes representing such interactions. Some examples include the

interactions in transient complexes (141; 177; 151; 149), in promiscuous proteins (that in some cases can bind to hundreds of partners) (63; 84; 185), and in disordered proteins (61; 129). Another limitation is that protein structures derived by some experimental techniques (e.g. X-ray crystallography) do not represent the dynamics associated with the interaction between their components but the most likely or stable complex structure. As a consequence of these problems, it is impossible to indicate with certainty which residues do not belong to any interaction site in a protein. Similarly, false positives can be introduced to the set of interface residues due to problems such as the formation of fake interaction sites in crystallography experiments and in the prediction processes used to generate some biological assemblies in the PDB.

There are diverse challenges involved in the processes used to train and test predictive models. A usual problem is overfitting, in which the errors produced after evaluating a predictive model on different datasets are significantly different. Some of the most common causes of overfitting are the creation of complex models that include large numbers of variables, and the construction of models using datasets that are not representative of the actual population of the problem. Techniques to detect or minimize overfitting include the generation of training datasets that attempt to represent the distribution of the data, the evaluation of the predictive error using cross-validation experiments (e.g. 5-fold cross validation) and testing datasets that are independent from the training datasets, and the use of regularization terms that penalize complex models that use a large number of parameters. The performance of a predictor can also be dramatically affected by the selection, representation and combination of the attributes used to describe a protein site. Depending on the number of attributes and attribute representations, sometimes it is essential to perform a large number of experiments to maximize the predictive performance without causing overfitting. Another common problem is the use of inadequate metrics to evaluate the performance of a predictor. Given that the number of interface residues is generally smaller than the number of non-interface residues, measures such as the accuracy, that account for the number of correctly predicted instances, can indicate very good performance for very bad predictions (e.g. high values of accuracy can be produced if all the residues are predicted as non-interacting residues).

The comparison of different predictors of protein-protein interface residues can be a difficult

task, mainly because of the differences in the experimental approaches used to build each predictor (53). The creation and evaluation of prediction methods involve problems such as the selection of the definitions of interface and surface residues, the type of complexes used to train the predictors (e.g. homo-obligomer, hetero-obligomer, homo-transient, or hetero-transient complexes), the evaluation methodology (e.g. cross-validation experiments, testing with an independent dataset), and the performance measures (e.g. correlation coefficient, precision-recall curves, area under the receiver operating characteristic curve). An ideal comparison between different predictors would require to train and test them using the same set of proteins and experimental conditions. In most in the cases this would require a copy of the source code used to build the predictor, or to write a version of the program following the specifications given in the literature. However, these approaches are difficult to follow in practice due to factors such as the lack of the details needed to successfully replicate the predictors from the description given in the literature, the lack of response of some of the authors of the prediction methods (e.g. students that already completed their studies), or the dependency of some predictors on tools that have been deprecated and are no longer available. An alternative to assess different methods is to utilize their web servers to generate predictions for proteins in a given dataset and to evaluate these predictions using the same performance metrics (213; 43; 11). However, this approach can lead to unfair comparisons because it does not consider factors such as the use of different datasets to train different predictors or the selection of different definitions of interface residues.

All these problems make the creation and evaluation of prediction methods a non-trivial, time-consuming, and computationally-intensive task.

## 1.2 Research aims

In light of (i) the costs and limitations of the experimental methods used to determine protein interaction sites, (ii) the rich set of features available for structure-based prediction methods, and (iii) the limitations in predictive performance of the existing predictors of protein-protein interface residues and protein-protein interaction patches; the major aim of this dissertation is the creation of tools and methods for improving structure-based prediction of protein-protein

interaction sites defined in terms of interface residues and interface patches. To achieve this aim, we defined three sub-goals:

1. The creation of a database of protein-protein interface residues that facilitates, among other applications, the creation of well-characterized datasets of protein-protein interface residues used to train and test predictors of interaction sites, and the generation of information derived from protein structures.
2. The construction of a method for predicting protein-protein interaction patches based on the results generated by predictors of interface residues. This method will allow scientist to focus on the development of reliable predictors of interface residues as a way to successfully predict interface patches.
3. The definition of a reliable method for predicting protein-protein interface residues based on the similarity between regions of a query protein and regions extracted from a big dataset of interacting proteins. Unlike existing similarity-based prediction methods, this method should generate predictions for any given protein structure (even in the absence of similar proteins) and should be efficient in terms of the required computational resources.

### 1.3 Dissertation organization

The dissertation is divided into the following sections:

**Chapter 1.** We introduce the problem of predicting protein-protein interaction sites and describe the organization of this thesis.

**Chapter 2.** We present *ProtInDb*, a database of protein-protein interface residues used to visualize interaction sites, and to allow the creation of representative datasets that can be used to train and test predictors of interface residues. The database is accessible using the Web server at <http://protindb.cs.iastate.edu/>, which allows users to: (i) visualize the interface residues in a protein complex deposited in the PDB; (ii) create representative datasets of protein-protein interface residues according to parameters used to determine the definition of interface and surface residues, to select the set of representative proteins according to desired sequence similarity, protein length and quality of the protein structure; and (iii) to download

a reduced version of the database according to user-provided parameters used to define surface and interface residues and to select between complexes represented as asymmetric units or biological assemblies in the PDB. A manuscript describing this database has been submitted to “Database: The journal of biological databases and curation”, an Oxford Journal.

**Chapter 3.** We introduce *PoInterS*, a method for predicting protein-protein interaction sites based on the scores or labels produced by a predictor of interface residues. *PoInterS* decomposes the surface of a protein in as many patches as surface residues, where a patch is composed of a central residue and the closest residues on the surface of the protein, and utilizes the outcomes generated by a predictor of interface residues for scoring each patch using different metrics. *PoInterS* returns the subset of patches with the highest scores as the most likely to be interaction sites. Comparisons using an independent dataset indicate that *PoInterS* outperforms other state-of-the-art predictors of interaction patches, indicating that the problem of predicting interaction patches can be reduced to the problem of predicting interface residues. *PoInterS* is available as a Web server at <http://pointers.cs.iastate.edu/>. A manuscript introducing *PoInterS* is to be submitted to PLoS one.

**Chapter 4.** We propose *PrISE*, a method to predict protein-protein interface residues based on similarity between local structures of proteins. *PrISE* decomposes a protein into structural elements composed of a central residue and its surrounding neighbors. A structural element is represented using data derived from the atomic composition and accessible surface area of its surface residues. This representation allows *PrISE* to efficiently extract, from a dataset of structural elements derived from interacting proteins, a set of similar elements to those of a query protein. Each similar structural element is weighted according to metrics indicating whether they were derived from proteins similar to the query protein, or from local regions in proteins that are similar to local regions in the query protein. These weights are used to compute a score indicating whether the central residue in the structural element is or is not an interface residue. Experiments performed using different test datasets indicate that the performance of *PrISE* is superior or comparable to state-of-the-art structure-based prediction methods. These results indicate that methods based only on local structural similarity are a viable alternative for predicting interface residues. *PrISE* has been implemented as a Web server available at <http://>

[prise.cs.iastate.edu/](http://prise.cs.iastate.edu/). The paper “Predicting protein-protein interface residues using local surface structural similarity” describing *PrISE* was published in BMC Bioinformatics in March 2012 (98).

**Chapter 5.** We summarize the contributions of the dissertations and describe future work.

## CHAPTER 2. *ProtInDb*: A DATA BASE OF PROTEIN-PROTEIN INTERFACE RESIDUES

*Paper submitted to Database: The journal of biological databases and curation*

Rafael A. Jordan, Feihong Wu, Drena Dobbs and Vasant Honavar

### 2.1 Abstract

Protein-protein interactions constitute the physical basis for formation of complexes and pathways that carry out virtually all major cellular processes. Knowledge of the residues in the interface between interacting proteins is of special interest in areas such as drug discovery, protein function prediction and protein docking. Because experimental determination of protein interfaces is expensive in terms of cost and effort involved, there is an increasing focus on computational prediction of protein interfaces e.g., using protein interface predictors trained on datasets extracted from experimentally determined complexes. Such datasets of known interfaces can also be used for guiding docking programs, scoring docked conformations, predicting new complexes, and validating interactions. Against this background, there is an urgent need for datasets of protein-protein interfaces.

We introduce *ProtInDb*, a database of protein-protein interface residues that supports visualization of interface residues on both the amino acid sequence and 3D structure of proteins of interest and the customized extraction of well-characterized datasets of interface residues. *ProtInDb* accommodates a flexible definition of interface residues through user-provided parameters that specify the criteria that need to be met for a residue to be considered a surface residue and an interface residue. It also allows users to extract interface residues from asymmetric units or biological assemblies deposited in the PDB. The datasets returned by *ProtInDb*



contain non-redundant protein chains selected according to user-specified sequence identity cut-off, sequence lengths, and the R-value and resolution of their structures. For each protein chain, *ProtInDb* computes a graph representing the interactions of residues on its surface, bipartite graphs representing the interaction of its residues with residues in other chains in the complex, mappings between the positions of every residue in the sequence and in the structure, and information regarding its sequence and the accessible surface area of its residues.

**Database URL:** <http://protindb.cs.iastate.edu>

## 2.2 Introduction

Interactions between proteins have important roles in almost every cellular process, from DNA replication and transcription, to identification and elimination of pathogens. The identification of the amino acid residues that participate in the interface between interacting proteins has applications in problems such the understanding of disease pathways and drug design. However, the determination of such residues requires methods that are costly and labor intensive (53). Therefore, there is an urgent need to develop computational tools to facilitate the analysis and prediction of interface residues. Interaction sites have been analyzed from the perspectives of their physicochemical and structural properties (92; 180; 181; 37; 4; 104; 103; 68; 201; 33; 207; 63), sequence and structural conservation (187; 25; 161; 76; 26; 34; 69; 105; 212), types of complexes (91; 40; 144; 16; 67), contact preferences (42; 3; 207), interface promiscuity (84; 126), etc. One of the goals of these analyses is to identify a set of features that can be used to differentiate between interface and non-interface residues. Using different combinations of such features, diverse predictors of protein-protein interface residues have been developed (213; 43; 11; 56). These predictors can be classified into sequence-based and structure-based. Sequence-based predictors (62; 145; 205; 200; 47; 193; 32; 133; 202) use information derived from properties associated with the sequence or with amino acids residues to perform predictions, whereas structure-based predictors (31; 155; 158; 110; 116; 122; 172; 147; 123; 107; 127; 211) use information derived from the three dimensional representation of protein complexes. In both cases, the set of interface residues required to train and evaluate such predictors generally is extracted from the structure of the proteins stored in the Protein Data Bank (PDB) (12).

In this context, we introduce *ProtInDb* (protein-protein interface resides data base), a database that allows a user to visualize the interface residues between two or more subunits (chains) in a protein structure and to efficiently generate datasets of protein-protein interface residues derived from interacting protein structures deposited in the PDB. Interface residues can be defined using threshold values on different distance metrics between atoms on two protein subunits. These metrics include distance between the centers of the atoms, distance between the Van der Waals surfaces of the atoms, and distance between the centers of  $\alpha$ -carbon atoms of two residues. In addition, the user can define which residues are on the surface of a chain using thresholds on the relative accessible surface areas. Interface residues can be extracted from asymmetric units or biological assemblies that had not been deprecated in the PDB. *ProtInDb* is updated every two weeks, providing users with up-to-date datasets of protein-protein interface residues. These datasets can be composed of the proteins in a list provided by a user or of proteins selected from *ProtInDb* according to several parameters given by a user. These parameters include sequence identity, R-value, resolution, and length of the protein sequence. Sequence identity can be used to select non-homologous proteins, allowing the creation of non-redundant datasets. Other parameters can be used as filters to exclude proteins that do not satisfy the user preferences. The information about each protein included in a non-redundant dataset includes its sequence and structure, a mapping between each residue in the structure and its corresponding position in the sequence, its accessible surface area before complexation, and graphs representing the neighborhood of each residue on the surface of each subunit and the interaction between residues of two different subunits. In addition, *ProtInDb* allows a user to generate and download a simplified version of the database. This simplified version includes the following information for each protein in the database: sequence, mappings of the positions of each residue in the sequence and in the structure, and chains indicating for each residue in each subunit whether it belongs or not to the interface with another subunit and whether it is or not on the surface of the subunit. Interface and surface residues are computed according to parameters defined by the user. *ProtInDb* has been used in several applications including extraction of datasets used for: training and evaluating several sequence and or structure-based, protein interface predictors (95; 202; 98; 97), assessing techniques for ranking protein

conformations produced by docking programs (203), and for studying conformational changes of antigens after binding with antibodies (96). The information provided by *ProtInDb* can be used in tasks such as analysis of protein interfaces, prediction and validation of protein-protein interactions, and improving protein docking, among other applications.

### 2.3 Databases and servers of protein-protein and domain-domain interfaces

In this section we present a list of databases and server providing protein-protein and domain-domain interface residues, that have been updated on or after 2009. We start enumerating databases of protein-protein interface residues. *ProFace* (166) allows to analyze protein-protein interfaces. It receives as input a PDB file and computes several structural properties such as number of atoms and residues in the interface core and rim, and interface and surface areas of patches of interface residues. *PISA* (108) allows the exploration of interfaces, prediction of quaternary structures, and search for similar interfaces and structures. Options to visualize interfaces and to present additional information of the interaction sites are given to the user. This downloadable database is continually updated. *PDBsum* (111) summarizes the information of the structures deposited in the PDB and provides links to another databases, results of diverse analysis, and schematic diagrams of protein-protein interface residues. *PDBsum* also allows to visualize interactions between proteins, and protein surfaces. Using *PDBsum* it is possible to download a list of interacting atoms for each pair of interacting chains in a PDB complex. *Protorp* (162) allows to analyze some physicochemical properties of protein-protein interaction site as well as to obtain a list of interacting residues defined as the residues that loss  $\geq 1 \text{ \AA}^2$  after complexation. *TCBRP* (82) allows the visualization of interface residues, that are computed as the union of the interface residues of proteins that share  $\geq 95\%$  of sequence identity with a query protein. Interface residues can be defined using threshold values on minimum distance between atoms and on loss of accessible surface area upon complexation. *PICCOLO* (14) is a downloadable relational database that stores 12 different definitions of interfaces using fixed threshold values. *PICCOLO* provides information of interacting sites at chain, residue, and atomic level. Interface residues can be computed from asymmetric units or from biological assemblies extracted using *PISA*.

Several databases of domain-domain interfaces has been created using different definitions of domains. In *SCOPPI* (199), based on the definition of domains given in *SCOP*(134), two domains are defined to interact if at least five pairs of residues are separated by at least 5 Å. *SCOPPI* displays interacting residues on the sequence of the domain. *SCOWLP* (178), based on *SCOP* domains, is oriented to perform analysis of protein interactions at domain level. It also provides some characterization and visualization tools for interface residues, that are computed using predefined distances between atom types. The downloadable database *3DID* (175) stores information about 3D interaction domains extracted from *Pfam*(59). This database provides an option to visualize interface residues. *SNAPPI-DB* (89) is a database of domain interactions that can be downloaded altogether with an application programming interface.

Most of these databases are oriented to the visualization of interface residues, but none of them allow to generate a representative dataset of protein-protein interfaces. In addition, to the best of our knowledge, *ProtInDb* is the only database that provides information about the topology of the protein surface and the protein-protein interaction sites in form of graphs. Finally, *ProtInDb* is the only database that provides together the following functionalities: flexible definition of interface and surface residues (using thresholds given by the user), an option to generate information for a list of protein given by the user, the possibility to extract information from the asymmetric units or the biological assemblies stored in the PDB, and the representation of the information using text files that can be easily processed in any programming language.

## 2.4 Materials and Methods

### 2.4.1 Biological assemblies and asymmetric units.

*ProtInDb* contains information of the interface residues between the chains contained in biological assemblies or in asymmetric units. A biological assembly (*BIA*), or biological unit, is a macromolecular structure that has been shown or is believed to represent a functional protein assembly. *BIAs* can be experimentally determined or computationally defined using software such as PQS(75) and PISA (108). An asymmetric unit (*ASU*) represents the smallest part of a crystal structure such that will generate a unit cell of the crystal after translation and rotation

of copies of the *ASU*. An *ASU* can be composed of a biological assembly, a part of a biological assembly, or several biological assemblies.

#### 2.4.2 Protein-protein interface residues.

Interface residues are commonly defined using measures such as loss of solvent accessible surface area of a residue after the formation of the complex (91), Voronoi Diagrams (156), minimum distance between atoms (144) (or  $\alpha$ -carbon atoms (54)) of amino acid residues in two different proteins, and minimum distance between the Van der Waals surfaces of amino acid residues (179). However, it has been shown that different definitions produce interaction sites that are almost identical in terms of number of residues and accessible surface areas (53). Therefore, *ProtInDb* provides three different definitions of interface residues based on three distance metrics computed between the atoms of two residues: (i) distance between the  $\alpha$ -carbons, (ii) distance between the centers of the atoms, and (iii) distances between the Van der Waals surfaces of the atoms. Given one of this distance metrics, *ProtInDb* defines a residue as an *interface residue* if at least one of its atoms is separated from one of the atoms in a partner protein by at most the threshold value provided by the user.

Interface residues extracted from *ASU* are computed from the subunits belonging to the first model<sup>1</sup> in the PDB file whereas interface residues extracted from *BIA* are computed considering all the subunits in all the models in the PDB file.

#### 2.4.3 Protein surface residues.

An amino acid is defined to be a *surface residue* if its relative accessible surface area (*RASA*) in the isolated protein chain is  $\leq$  than a threshold value defined by the user.

#### 2.4.4 Data collection.

The process used to collect the information stored in *ProtInDb* is summarized in Figure 2.1.

---

<sup>1</sup>Models are used in PDB files to store different structures. For example, each conformation of protein structures determined using nuclear magnetic resonance is represented as a different model. On the other hand, for the case of biological assemblies composed of several copies of the asymmetric unit, each copy can be represented as a model.

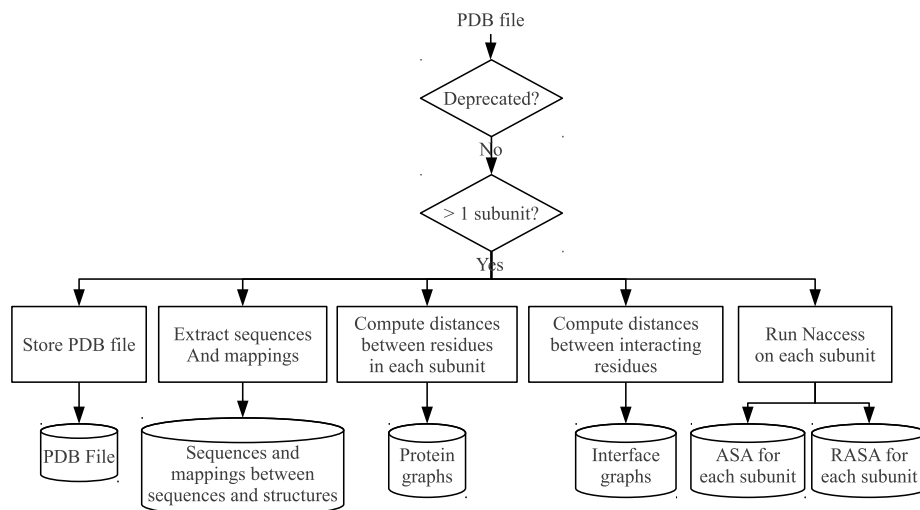


Figure 2.1 **Flow diagram of the data collection process for a protein.**

The main source of information used to collect the data of *ProtInDb* is the Protein Data Bank (12). After identifying non-deprecated proteins representing interacting subunits, the PDB file is stored in *ProtInDb* along with the following information:

- Amino acid sequences for each subunit in a protein complex. These sequences are extracted from the atomic coordinates in the PDB files<sup>2</sup>.
- Mappings between residues in the protein structure and the protein sequence. Each residue in the coordinates section of a PDB file is uniquely identified by a residue sequence number and a code for insertion (this unique identification is referred as *resId* in this document). A mapping between the *resId* and the position of the same amino acid in the sequence is stored to allow the user to efficiently move between structure-based and sequence-based representations of a protein subunit and vice versa.
- Protein Graphs. A protein graph represents distance relationships between the residues in a protein subunit. The nodes in the graph represent the residues in a protein subunit.

Two nodes are connected by an edge if the distance between the Van der Waals surfaces of

<sup>2</sup>Some amino acids of a protein sequence can be absent from the atomic coordinates describing the protein structure in a PDB file.

the closest atoms in the corresponding residues is  $\leq 2.5$  Å. The edges are labeled with such distance. Protein graphs are useful to perform tasks such as analysis of characteristics in the surrounding region of an amino acid, representation of residues using features extracted from the local environment in a protein, generation of protein patches on the surface of a protein, and computation of intra domain-domain interface residues for protein subunits composed of several domains.

- **Interface graphs.** An interface graph is a bipartite graph of the interacting residues between two protein subunits. Nodes in a partition represent residues belonging to a subunit. Two nodes in different partitions (i.e. in different subunits) are connected by an edge if the distance between the corresponding residues is below some predefined threshold. Edges are annotated with the minimum distance between atoms in the residues. The predefined thresholds are: 11 Å for interface graphs based on distances between  $\alpha$ -carbons, 10 Å for graphs generated using distances between the center of the atoms, and 5.5 Å for graphs based on distances between the Van der Waals surfaces. These threshold values were selected to allow efficient computation of the graphs. Interface graphs are useful to compute sets of protein-protein and domain-domain interface residues and to analyze structural and physicochemical properties in sets of interacting residues.
- **Accessible surface areas of atoms and residues.** The accessible surface area (*ASA*) and relative accessible surface area (*RASA*) for atoms and residues in each isolated protein subunit can be used to determine whether a residue or atom is or not on the surface of the subunit. This information is computed using the software *Naccess* (83) with default parameters.

#### 2.4.5 Implementation details.

*ProtInDb* data is stored using text files that are described in the documentation section of the Web server. The computations are performed using Java 1.6 and Jython 2.5 and the Web application is served using Apache Tomcat 6.0 with servlets written in Jython. All the code was generated by the authors with exception of the computations of accessible surface areas

(generated using Naccess (83)) and the generation of lists of representative chains (computed using PISCES (194)). All the data in *ProtInDb* is obtained from public databases and the software used does not impose restrictions for academic use.

## 2.5 Results and Discussion

### 2.5.1 User interface.

*ProtInDb* is accessible as a Web application at <http://protindb.cs.iastate.edu>. This application provides options for visualization of the interface residues of a given protein, generation of datasets of interface residues, and for downloading reduced versions of the database according to user-defined parameters.

To visualize the interfaces in a PDB complex, the user should provide the following inputs: (i) The PDB Id of the protein complex; (ii) The Id of the query protein subunit; (iii) An optional list of the potential interacting subunits in the complex (if this list is not provided the interfaces will be computed considering all the subunits in the protein); (iv) The definition of interface residues (i.e. distance between atom centers, between Van der Waals surfaces, or between  $\alpha$ -carbon atom centers) as well as the threshold value used to determine whether a residue belongs or not to the interface. (v) A threshold value for the *RASA* that defines whether a residue is or not on the surface of the query protein; (vi) A selection that indicates whether the interfaces are computed from the *ASU* or the *BIA*; (vii) A selection indicating whether the interface residues will be computed from the query protein subunit or from its sequence homologs. A sequence homolog is defined as a protein subunit that shares  $\geq 96\%$  sequence identity with the sequence of the query protein subunit. Interface residues are computed from sequence homologs using the following steps: (a) interface residues are computed for every sequence homolog, (b) alignments between the query sequence and the sequence of every homolog are computed, and (c) interface residues in the sequence homologs are mapped into the query sequence using the alignments computed in the previous step. Therefore, the interface residues of the query protein can be seen as the union of the sets of interface residues of its sequence homologs. When interfaces are



computed using sequence homologs, the list of potential interacting subunits provided by the user is ignored.

An example of the results of the output generated by the option to visualize interface residues in the Web application is presented in Figure 2.2. The information presented to the user is: the amino acid sequence, a binary string indicating whether or not each residue in the sequence belongs to the interface, a binary string indicating whether or not each residue in the sequence is on the surface of the protein, the *resId* of the interface residues on the surface of the proteins and the *resId* of the interface residues that are buried in the subunit. A 3D representation of the query subunit and its interface residues is also presented using the software Jmol (78; 73). This software allows to the users to perform operations such as zooming, rotation, measuring of distances between amino acids, and generation of different representations of the protein structures (e.g. ribbons, surfaces, balls and sticks, etc.). If the user selected the option to compute the interface residues from the set of sequence homologous of the query protein, the interface residues for each homolog are displayed as a binary string (as seen in Figure 2.2).

The Web application also allows users to build non-redundant datasets of protein-protein interface residues. An example of the Web interface used to generate these datasets is presented in Figure 2.3. The generation of a list of non-redundant proteins starts by removing non-interacting proteins from the list of all the protein subunits in *ProtInDb* or from a list of proteins provided by the user. Then, the software PISCES (194) is used to filter out proteins with lower structural quality (according to user-defined parameters on the resolution and the R-value) or with sequence lengths outside a range defined by the user. The list of remaining proteins is used by PISCES to build the set of representative proteins according to the following algorithm: the protein with the best resolution and R-value in the list is selected as representative. This protein, and proteins sharing sequence identity  $> i$  (where  $i$  is defined by the user) with it, are excluded from the list. These two steps are repeated until the list is empty. Once the set of representative proteins have been computed, the sequence, maps between the residues in the sequence and the structure, proteins graphs, interfaces graphs, and accessible surface areas, are

### Interface Residues for a given PDB Id

#### Query Parameters

PDB Id: 2f03  
 Chain Id: A  
 Partner chains: All the chains has been taken as possible partners.  
 Distance between any two atoms considering their Van der Waals radius. Threshold: 0.5 Å  
 RASA threshold value used in defining surface residues: 5 %  
 Data base used to compute the interfaces: Asymmetric units  
 interfaces computed using: The given chain and chains with identity >= 96%

#### Query Results

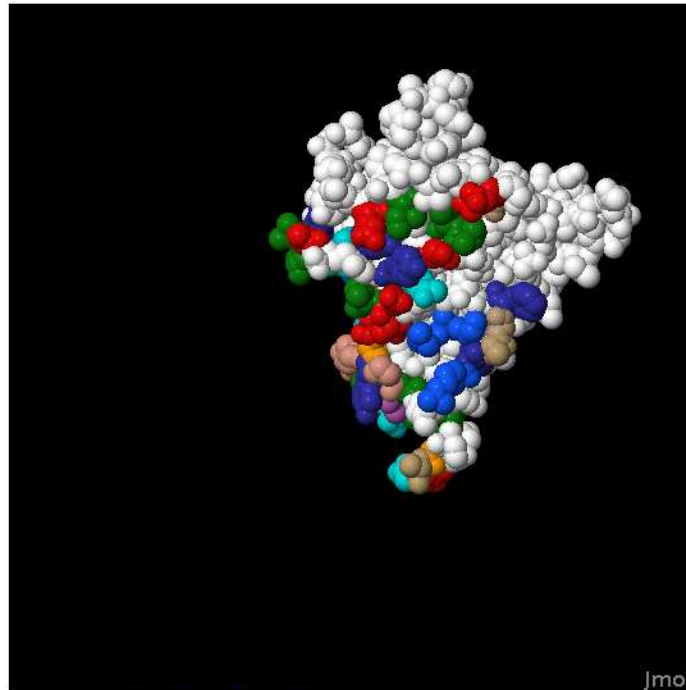
Sequence HQDYRELSLDELESVEKQTLRTIVQALQQYSKEAKSIFETTAADSSGGEVIVLAEDI  
 Interfaces .I.....I...I..II..II..II..I...I.....I..II..I.  
 Surfaces SSS.SS.SSSS.SS..SS..SS..SS..S..SS.SSS..SSSSSSSSSSSS.SS.

#### PDB residue numbers for interface residues on the surface of the protein

3, 15, 19, 22, 23, 26, 27, 30, 34, 38, 45, 48, 49, 52, 53, 56, 59, 60, 63, 64, 65, 66, 67, 68, 69, 70, 73, 74, 75, 76, 77, 81, 82, 85, 86, 87, 88, 89, 92, 144, 146, 147, 148, 149, 152, 248, 264, 265, 266, 267, 268

#### PDB residue numbers for interface residues NOT on the surface of the protein

31



chain A non-interface residues are colored white.  
 chain A interface residues are colored using colors different to white.

Show all interface residues  spin

#### Interface residues computed from similar protein chains

4 chains with at least 96% identity were found:

2ezvA, .....I...I.....I...I..II..I..II..  
 2ezvB, .....I...I.....I.....II..II..I...I..  
 2f03A, .I.....I...I..II..II..I.....I.....  
 2f03C, .I.....I..II..II..I.....I.....

Figure 2.2 Example of the output corresponding to the visualization of interface residues. Interface residues of the subunit A in protein PDB:2f03 are shown in different colors. White spheres indicate non-interface atoms. Atoms in interface residues are colored according to the amino acid to which they belong.

computed according to the parameters specified by the user. Files containing this information, and the PDB files of the representative proteins, are compressed by *ProtInDb*. Finally, an email containing an URL pointing to the compressed file is send to the user. All the data is stored in machine-readable text files organized in several directories. A complete description of these files and directories is provided in the documentation section of the Web application.

The user also can generate and download a copy of the basic information of the dataset. This information includes the sequence, maps between residues in the sequence and the structure, and chains indicating whether each residue in the sequence is or not on the surface of the isolated subunit and whether each residue in the sequence belongs or not to the interface of the subunit with every other subunit in the complex. The previous information is computed according to parameters specified by the user.

### 2.5.2 Updates and content.

The first version of the data base was created in May 2009 and it has been updated every two weeks. The update process removes information of complexes that have been deprecated in the PDB and adds information of new or updated complexes. A summary of the information contained in *ProtInDb* at March 3, 2012 is presented in Table 2.1.

Table 2.1 **Summary of the information stored in *ProtInDb* at March 3, 2012.** The row labeled “Number of protein subunits” indicates the number of protein chains in asymmetric units and the number of protein chains with unique name in biological assemblies (i.e. every chain in a biological assembly is counted once even if it has several copies in the assembly).

Description	<i>ASU</i>	<i>BIA</i>
Number of complexes with > 1 subunit	43,029	39,866
Number of protein subunits	162,145	112,762
Number of subunits present both in <i>ASU</i> and <i>BIA</i>	105,529	

## Generation of data sets of protein-protein interfaces

---

### Select one of the following methods for create the data set:

- Extract the data set from the data base of protein protein interfaces.
- Extract the data set from the following file containing a list of protein chains:

[Example of a valid file](#)

---

### Choose one or more methods and threshold values for distinguishing between interface and non-interface residues:

- Distance between any two atoms (suggested: 4 Å)
- Distance between any two atoms considering their Van der Waals radius (suggested: 0.5 Å)
- Distance between C<sub>α</sub> atoms (suggested: 8 Å)
- 

### Choose a RASA threshold value to define surface residues (suggested value: 5 %):

%

---

### Select at least one of the following data bases to extract the information from:

- Use data base of asymmetric units
- Use data base of biological units
- 

### Options to reduce the size of the data set

Maximum identity  %

Maximum R-value

Resolution: From  to  Å

length of the chain: From  to  residues

---

### Provide a valid email address to be notified when the data set has been generated:

---

Figure 2.3 Screen shot of the Web interface used to generate datasets of protein-protein interface residues.

### 2.5.3 Examples of applications that use *ProtInDb*.

In addition to the application used to visualize interface residues, *ProtInDb* has been used to generate datasets to train and test predictors of interface residues and interaction sites, and to perform analysis concerning interface residues.

The information in *ProtInDb* is used by predictors of interface residues based on sequence and structural similarity. NPS-HomPPI and PS-HomPPI are two predictors of protein-protein binding sites based on protein sequence homology (202). PS-HomPPI predict interface residues that are specific to the interaction between two given proteins whereas NPS-HomPPI predicts interface residues for a query protein without considering its interaction partners. These predictors estimate the interface residues of a query protein from the interface residues of its sequence homologs. The interface residues of such homologs are computed using *ProtInDb*. Given that *ProtInDb* is constantly updated, the predictions of PS-HomPPI and NPS-HomPPI always consider the latest proteins included in the PDB. *PrISE<sub>C</sub>* (98) is a predictor of protein-protein interface residues based on the similarity between local substructures on the surface of a protein. A local substructure represents the atomic composition and the accessible surface area of a patch on the protein surface. Given the local substructures of a query protein, *PrISE<sub>C</sub>* searches for similar substructures in a precomputed database derived from known interacting proteins. The information in this database has been completely extracted from *ProtInDb*.

*ProtInDb* has been utilized to build non-redundant datasets used to train and test several machine learning predictors of interface residues. *PoInterS-SVM* (95; 97) is a predictor of protein-protein interaction sites. An interaction site is defined as a semi-circular patch on the surface of the protein that cover most of the actual interface residues. To predict interaction sites, *PoInterS-SVM* decomposes the surface of the protein in patches that are ranked using information derived from the scores or the interface/non-interface labels generated by a predictor of interface residues. Different predictors of interface residues were trained and tested using non-redundant datasets extracted from *ProtInDb*. The predictor that achieved the highest performance was based on support vector machines.

*ProtInDb* also was used to build a benchmarking dataset of conformational epitopes that included information from bound and unbound structures (96). This dataset was used to compare the performance of different discontinuous B-cell epitopes predictors.

## 2.6 Summary

*ProtInDb* offers a useful resource for the research community interested in analysis and prediction of protein interfaces, validating protein-protein interactions, improving protein-protein docking, and predicting new complexes, among other applications. *ProtInDb* supports visualization of protein-protein interface residues and creation of non-redundant datasets involving interacting proteins in the PDB. Visualization presents a user a graphical representation of protein structures and the set of amino acid residues that form the interface between two or more subunits in a protein complex. These interface residues are determined using a set of parameters defined by the user. Datasets of interface residues returned by *ProtInDb* provide structural information of interacting protein subunits extracted from asymmetric units and/or biological assemblies. Such information includes sequences, maps between residues in the structure and the sequence, protein graphs representing the interactions between the residues in a protein subunit, interface graphs representing interaction between residues in different proteins subunits, and data about the accessible surface area of each isolated subunit. Support for automated and customizable extraction of datasets based on user-specified parameters allows users to save considerable time and effort in tasks such as statistical analysis of protein-protein interfaces or domain-domain interfaces; scoring docked conformations or guiding docking; and retrieving datasets for training and evaluating alternative predictors of interface residues, hot spot residues, conformational epitopes, etc. The information contained in *ProtInDb* also can also be used in applications such as prediction of protein-protein interactions, selection of mutants for experimental verification of protein-protein interactions, understanding of protein functions, prediction of drugability for protein-protein interactions, and development of new therapeutic drugs.

## 2.7 Availability and requirements

*ProtInDb* is periodically updated and is freely accessible for academic use at <http://protindb.cs.iastate.edu>.

## 2.8 List of abbreviations

*ASA*: Accessible surface area.

*ASU*: Asymmetric units.

*BIA*: Biological assembly.

*RASA*: Relative accessible surface area.

*resId*: Identification of a residue in the coordinates section of a PDB file. This identification is composed of a residue sequence number and a code for insertion of residues specified in positions 23 to 27 in the atomic section of the PDB files (see <http://www.wwpdb.org/documentation/format32/sect9.html>)

*ProtInDb*: Data base of protein-protein interface residues.

## 2.9 Author's contributions

All the authors have participated in the design and implementation of the database, and in the writing of this manuscript.

## 2.10 Acknowledgments and funding

This work was funded in part by the National Institutes of Health grant GM066387 to Vasant Honavar and Drena Dobbs and in part by a research assistantship funded by the Center for Computational Intelligence, Learning, and Discovery. The work of Vasant Honavar while working at the National Science Foundation was supported by the National Science Foundation. Any opinion, finding, and conclusions contained in this article are those of the authors and do not necessarily reflect the views of the National Science Foundation.

## CHAPTER 3. A MODULAR APPROACH TO PREDICT PROTEIN-PROTEIN INTERACTION SITES

*Paper to be submitted to PLoS One*

Rafael A. Jordan, Yasser EL-Manzalawy, Drena Dobbs and Vasant Honavar

### 3.1 Abstract

**Background:** Protein-protein interactions play a critical role in protein function. Reliable identification of protein-protein interfaces is important for understanding the physical basis of protein complexes and their role in networks that underly virtually all cellular processes, predicting protein function, guiding protein docking, and developing new drugs. Because of the high cost of experimental determination of protein interfaces, there is an urgent need for reliable computational methods for interface prediction.

**Results:** We present *PoInterS*, a novel modular approach for predicting protein interaction sites using predicted protein-protein interface residues. *PoInterS* decomposes the surface of a protein into patches and ranks each patch based on the predicted protein-protein interface residues in the patch. The top-ranked patches are combined to obtain the predicted interface site of the protein. The modular design of *PoInterS* allows it to use predictions provided by any available protein interface residue predictor for ranking the surface patches of a query protein. Our experiments using leave-one-protein-out cross-validation on a benchmark dataset of 220 proteins show that *PoInterS* is able to correctly identify the interfaces in 81% of the cases. Our experiments using a blind dataset of 24 proteins derived from the first eight rounds in CAPRI show that *PoInterS* outperforms *SHARP*<sup>2</sup> and *PPI-Pred*, which are two state-of-the-art methods for predicting protein interface patches.



**Conclusions:** *PoInterS* offers a modular approach to computational prediction of protein-protein interaction sites that is competitive with the current state-of-the-art methods. An instance of the *PoInterS* method that uses a structure-based support vector machine predictor of interface residues has been implemented as a Web server which is freely available at <http://pointers.cs.iastate.edu>

## 3.2 Background

Protein-protein interactions constitute the physical basis for formation of complexes and pathways that carry out virtually all major cellular processes. Recent advances in high throughput techniques for experimental determination of protein interaction networks (169) have improved our understanding of how proteins interact together to perform different biological processes (186; 64; 65; 117; 176; 52). However, to gain a deeper understanding of the mechanisms involved in protein interactions it is important to identify the sites used for a protein to interact with another protein. Given that the experimental determination of protein binding sites is costly and labor intensive (53), there is an urgent need for reliable computational approaches to identify protein-protein interaction sites. In addition to providing important clues to biological function of novel proteins, computational prediction of protein-protein interaction sites can help to design focused experiments aimed at understanding specific protein interactions, develop new therapeutic drugs that inhibit the interaction between specific proteins involved in disease pathways, and reduce the search space in macromolecular docking.

Prediction of protein-protein interaction sites has been approached from two different perspectives: prediction of interface residues, and prediction of interface patches.

Protein-protein interface residues predictors (PPIRPs) classify each amino acid residue in the protein into interface residues or non interface residues based typically on the features describing the residue and its sequence or structural neighbors and/or its homologs. PPIRPs can be categorized into two major types: sequence-based and structure-based. In sequence-based PPIRPs (62; 145; 204; 205; 206; 190; 200; 77; 81; 70; 32; 44; 172) use sequence neighbors of the target residue to extract features that form the input to the classifier. Structure-based PPIRPs (214; 54; 57; 31; 24; 35; 113; 110; 115; 155; 158; 191; 79; 116; 165; 11; 49; 122; 121; 124; 172; 107;

123) use a set of structural neighbors of the target residue to extract features that represent each residue on the surface. A variety of features (43; 183) derived from the sequence of the target protein (e.g., amino acid identity physico-chemical properties) (91; 180; 93; 29; 40; 207; 130) its structure (e.g. accessible surface area, secondary structure, temperature factor, protrusion, planarity) (91; 180; 93; 181; 29; 40; 201; 207; 130) or its homologs (e.g. profiles generated from multiple sequence alignments) (181; 11; 50) have been explored in the literature. Because no single feature has been found to be sufficient for reliable prediction of protein-protein interaction sites (53), modern methods take advantage of machine learning approaches that can make use of multiple features to achieve good prediction results (213).

Protein-protein interface patch predictors (PPIPPs) deal with the identification of areas on the surface of the protein that contain most of the residues in the interface (interface patches). This approach allows users to focus their studies on a few high-ranked regions on the surface of the protein instead of examining predicted interface residues that can be scattered on different sites on the protein structure. PPIPPs have been developed using two approaches: The first approach constructs patches using clusters of closest atoms or residues on the surface of the protein that are likely to be part of protein interfaces (139; 58; 119; 138; 163; 153; 51). The second approach, on which our work is focused, deals with the selection of patches that are likely to be part of protein interfaces from the set of all pre-computed patches on the protein surface. Another difference is that the methods in the second approach assign a rank to each predicted patch whereas the methods in the first approach do not.

One of the earliest examples of the second class of PPIPPs was proposed by Jones and Thornton (93; 94), who defined a patch as a central residue and its  $m$  closest neighbors on the surface of the protein according to the distance between their  $\alpha$ -carbons, where  $m$  was computed as  $1.9n^{0.6}$ , and  $n$  is the number of residues in the protein. All patches on the surface of the protein were ranked using a score combining solvation potentials, residue interface propensities, hydrophobicity, protrusion, and accessible surface area values. The top-ranked patches constitute the prediction result. An improvement of this method (132; 90) was achieved using patches of size  $1.91n^{0.55}$  and a new scoring function that included solvation potentials, hydrophobicity, accessible surface areas and residue interface propensities. The resulting predictor was called

*SHARP*<sup>2</sup>. Liang et al. (120) used a side-chain energy scoring function to rank patches formed by a central residue and its 20 surrounding residues. The scoring function was defined as a linear combination of features such as atom contact surface, hydrogen bond energy, electrostatic interactions, desolvation energy, rotamer intrinsic energy, and disulfide bond energy. Bradford and Westhead (22) developed *PPI-Pred* using a different approach for representing patches and for predicting interaction sites. They defined basic patches using spheres covering from 6% to 8% of the residues in the protein. These basic patches were extended to include all residues in cavities or protrusions when the patch formed a ring, or decreased to consider only the largest patch from a set of unconnected patches enclosed inside the sphere. Each patch was represented using the means and standard deviations of the normalized values of shape index, curvedness, conservation score, electrostatic potential, hydrophobicity, residue interface propensity and solvent accessible surface area of the patch components. These patches were ranked using the scores generated by a predictor of interface patches based on a support vector machine. A set of non-overlapping top-ranked patches were returned as the predicted protein-protein interface sites. This method was improved in (21) by replacing the support vector machine classifier by a Bayesian network trained using the same dataset, features, and patch definition as in their previous work. Negi and Braun (137) defined a patch as a central residue and its  $n$  closest neighbors in a sphere of radius  $R$ , achieving a good balance between precision and sensitivity (recall) with  $R = 12 \text{ \AA}$ . Each patch in the surface of a protein was ranked using a score function based on the accessible surface area and the interface and surface propensities of the residues in the patch. Finally, a percentage of the top-ranked patches is returned as the prediction result.

Against this background, we introduce *PoInterS* (prediction of protein-protein interaction sites), a fast and modular method for predicting protein-protein interface patches for unbound proteins that allows users to focus their studies in a small set of ranked patches composed of close residues in the structure. *PoInterS* defines a patch as a central residue and its closest neighboring residues on the surface of the protein, which produces as many patches as surface residues on a protein. Each patch is ranked using the outcome produced by a protein-protein interface residues predictor, and three non-overlapping top-ranked patches are selected as the most likely to be interaction sites. Extensive experiments were performed to analyze the impact

that several prediction algorithms and different techniques for representing and sampling data have in the final performance of several *PoInterS* classifiers. We also study the effect of using different patch ranking schemes and the relationship between the observed performance of interface residues predictors and the performance of the interface patches predictors. Based on our experiments, we developed a predictor of protein-protein interface patches (*PoInterS-SVM*) that uses a predictor of interface residues based on a support vector machine. Evaluations using a non-redundant dataset of 24 proteins extracted from the first eight rounds of CAPRI indicate that the performance of *PoInterS-SVM* is superior to that of *SHARP*<sup>2</sup> (132) and *PPI-Pred* (22).

### 3.3 Methods

#### 3.3.1 Surface residues

A residue is considered to be a *surface residue* if its relative accessible surface area in the monomer is  $> 5\%$ . Relative accessible surface areas are computed using the program NACCESS (83) with default parameters.

#### 3.3.2 Surface patches

We used the definition of surface patches proposed by Jones and Thornton (93): A surface patch is composed of a central surface residue and its  $m$  nearest surface residues according to their  $\alpha$ -carbon Euclidean distances in the Brookhaven PDB file. Therefore, there are as many patches as surface residues in a protein. To avoid the construction of patches forming rings of residues around the protein surface, only residues with an angle  $< 110^\circ$  between their solvent vector and the patch central residue solvent vector are considered. A solvent vector is computed as the inverse of the vector between the  $\alpha$ -carbon of a residue and the center of gravity of the  $\alpha$ -carbons of its ten nearest surface residues. The number  $m$  of neighboring residues involved in a patch was computed as an approximate correlation between the number  $n$  of amino-acid residues of a protein and its number of interface residues. In our experiments, we used two different patch sizes,  $m = 1.92n^{0.56}$  and  $m = 1.91n^{0.55}$ , as defined in (92; 94) and

(90) respectively.

### 3.3.3 Representative interface patches

A patch is considered a *representative interface patch* if it covers most of the observed interface residues. A representative interface patch is used as the base for computing the performance of a prediction.

### 3.3.4 Datasets

#### Cross validation dataset

The *cross validation dataset*, or *CV* dataset, is composed of 220 protein chains with more than 40 residues, sharing  $\leq 30\%$  sequence identity, with resolution  $\leq 3.0 \text{ \AA}$ , and R-values  $\leq 0.3$ . In order to produce a small but representative dataset, proteins derived from complexes labeled as homodimers in the Protein Data Bank (PDB) (12) were extracted and subject to several filtering steps. First, complexes with only one protein chain as well as non X-ray determined protein structures were filtered out. Then, we used PISCES (194) with default parameters to select a set of protein chains that satisfied the constraints previously described. In this dataset, an *interface residue* is defined as a surface residue that loss  $> 1 \text{ \AA}^2$  of its accessible surface area after the formation of a dimeric complex. The dimers used to compute the interface residues were composed of the protein chain selected by PISCES and the largest chain in the same PDB complex. The final dataset is composed of 62,795 residues, from which 46,456 are on the surface and 10,373 belong to the interface. An enumeration of these 220 protein chains is available in the additional file cvDataset.txt.

#### Test datasets

We used two docking benchmark datasets to perform a blind validation of *PoInterS* and to compare *PoInterS* with two protein-protein interaction site prediction methods publicly available as online Web servers. The first test dataset, called *ZDOCK*, was used to conduct a blind validation of several *PoInterS* predictors. This dataset is composed of 299 chains derived from all the binary interactions contained in the 124 test cases of the docking Benchmark 3.0

(85). Each test case in the docking benchmark 3.0 is composed of a receptor and one or more ligands. Each protein in this non-redundant benchmark dataset consists of at least 30 amino acids and have resolution better than 3.25 Å. Non-redundancy was achieved using structural classification of proteins by avoiding any two cases to belong to the same family-family pair in the SCOP database. For the *ZDOCK* dataset, an *interface residue* is defined as a surface residue whose  $\alpha$ -carbon is separated by at most 7 Å from the  $\alpha$ -carbon of another residue in a different protein chain. This dataset contains 63,501 residues, from which 47,325 are on the surface, and 8,465 are interface residues. A list of the 299 proteins in this blind dataset is provided in the supplementary file `zdockDataset.txt`.

The second test dataset, called *Capri*, was used to compare *PoInterS* against the methods *PPI-Pred* (22) and *SHARP*<sup>2</sup> (90). This dataset is composed of 24 protein units retrieved from 19 targets used in the first eight rounds of CAPRI (critical assessment of prediction of interactions) (87; 86). Fifteen of these chains were used in (22) to evaluate the performance of *PPI-Pred*, and they share less than 20% sequence identity with the nine chains recently added. Each chain was associated with the partner that produces the largest interaction site in terms of number of residues. *Interface residues* were extracted from the contact residue information provided in CAPRI. This dataset has 5,940 residues, 4,546 surface residues and 582 interface residues. The list of selected dimers are presented in the supplementary file `capriDataset.txt`.

### 3.3.5 Prediction of protein-protein interaction sites (*PoInterS*)

The main idea behind the *PoInterS* method is illustrated in Figure 3.1. Given a query protein structure, *PoInterS* uses the following three-step procedure to predict interaction sites:

1. The surface of the protein is divided into a set of overlapping surface patches.
2. Interface residues in the query protein are predicted using a PPIRP.
3. The patches are ranked according to their potential for containing most interface residues, estimated using the information generated by the PPIRP.

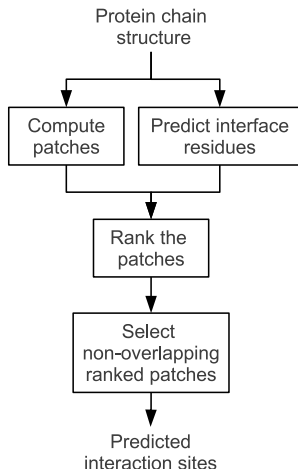


Figure 3.1 **Flow diagram of the *PoInterS* prediction method.**

4. The three top-ranked patches sharing less than 30% of residues are selected as the best candidates to be interacting sites<sup>1</sup>.

We evaluated four different patch ranking schemes based on the prediction generated by the PPIRP for each residue included in the patch. The rationale behind these schemes is that the most predicted interface residues are in a patch, the most relevant the patch is in the interaction between two proteins. The first ranking scheme considers the number of predicted interfaces in the patch to assign a ranking score, and is computed as:

$$Class_s(p) = \frac{\sum_{r \in int(p)} 1}{s}$$

where  $s$  is the number of residues in a patch  $p$  and  $int(p)$  denotes the set of predicted interface residues in  $p$ .

The second scheme considers the concentration of predicted interface residues around the central residue of the patch:

$$ClassDis_s(p) = \frac{\sum_{r \in int(p)} \frac{1}{d(r,c)}}{s}$$

where  $c$  represents the central residue of the patch and  $d(r, c)$  is one when  $r = c$ , or the euclidean distance between the closest atoms in residues  $r$  and  $c$  when  $r \neq c$ .

---

<sup>1</sup>The 30% threshold value was determined experimentally using a leave-one-protein-out cross validation experiment on the cross-validation dataset

The third scheme uses the probabilities estimated by the PPIRP indicating whether the residues in a patch belong to the protein interface:

$$Prob_s(p) = \frac{\sum_{r \in p} probability(r)}{s}$$

where  $r \in p$  represents each residue in the patch  $p$ , and  $probability(r)$  denotes the probability that  $r$  is an interface residue estimated by the PPIRP.

The last scheme weights the probabilities estimated by the PPIRP using the distances from the central residue to each residue in the patch, and is defined as:

$$ProbDis_s(p) = \frac{\sum_{r \in p} \frac{probability(r)}{d(c,r)}}{s}$$

The scores generated by these ranking schemes are in the interval  $[0,1]$ , and the highest the score, the most relevant a patch is in the interface between two proteins.

### 3.3.6 Prediction of interface residues

We built different predictors of protein-protein interface residues to evaluate the performance of *PoInterS* according to different classification algorithms, feature representation schemes, and techniques for dealing with unbalanced and unnormalized data.

We first train protein interface residue predictors that accept, as input, a set of features describing each residue and produce, as output, a label that indicating whether the residue belongs or not to the protein interface. Once such a protein interface residue predictor has been created, it is used to label each residue in a query protein (monomer) as an interface or non-interface residue

The input to the protein interface residue predictor typically consists of features extracted from the residue and its sequence or structural neighbors. In the sequence-based representation, a residue  $a_j$  is described by a sliding window  $(a_{j-k}, a_{j-k+1}, \dots, a_j, \dots, a_{j+k-1}, a_{j+k})$  containing data of a specific feature for the  $2k+1$  adjacent residues in the protein sequence. In the structure-based representation, a surface residue  $s_j$  is described by a tuple  $(s_j, s_{j,1}, s_{j,2}, \dots, s_{j,(2k)})$  in which  $s_{j,r}$  corresponds to the  $r$ -th surface residue closest to  $s_j$ . In this study, each residue  $s_j$  was



described using features that have been successfully applied to predict protein-protein interface residues: Amino acid identity (205), secondary structure (139), crystallographic temperature factor (35; 123) and relative accessible surface area (RASA) (31; 155; 123). The secondary structure was computed using the Stride stand-alone program (60). The temperature factor for each residue was calculated as the averaged temperature factor of its atoms in the PDB file. The RASA was computed using software the NACCESS (83) with default parameters.

We experimented with four different machine learning methods for predicting protein-protein interface residues in the context of protein-protein interface patches prediction: Naive Bayes (NB), decision trees (DT), logistic regression (LR) and support vector machine (SVM). Naive Bayes is a generative model that assumes that the variables used for classification are conditionally independent given the target class. Decision trees use elements of information theory to model dependencies among a set of variables describing instances in a training dataset. These dependencies represent a set of rules that may be used to predict the class associated with every instance in a testing dataset (15). The training process of DT classifiers is very efficient and the resulting models are easy to understand. Logistic regression classifiers model an underlying binomial distribution of the data as a linear function of the variables. LR predictors may produce more accurate results than NB if the independence assumption of NB does not hold (140). Support vector machines (188) compute a set of representative samples (support vectors) that maximize the separation distance between the classes, and use the support vectors to perform classification. In general, the performance of SVM is better than the performance of the other three methods, but the construction of a model takes longer. When a sample space is not linearly separable, Logistic regression and SVM models may use kernels to try to transform the space into one that is linearly separable. We used the implementation of these supervised machine learning algorithms provided by the Weka software (72). The models for naive Bayes, decision trees and logistic regression were trained using default parameters. The decision trees were built using the J48 algorithm. The SVM classifiers were trained using SMO (154) with parameters  $C = 1$ ,  $\epsilon = 1E - 12$ , and using a radial basis function kernel with parameter  $\gamma = 0.01$ . The values for  $C$  and  $\gamma$  were selected using a grid search with steps of  $10^{-i}$  in a subset of 100 proteins in the cross validation dataset for ranges varying from 100 to 0.1 and from 0.1 to 0.001

respectively.

The performance of a classifier may be affected by factors such as the numerical representation of the data or how balanced is the ratio of interface/non-interface residues in the dataset used to train the model. For the problem of predicting protein-protein interface residues, the number of non-interface residues is generally larger than the number of interfaces. Therefore, a bias towards non-interface residues can be introduced in the prediction. This problem may be addressed by using sampling techniques on the training dataset oriented to select an approximately equal number of interface and non-interface samples. Three sampling techniques were tested in this study: (1) under-sampling of non-interface residues by randomly removing non-interface examples until achieving an equal number of interface/non-interface residues; (2) over-sampling of interface residues by introducing 50% of additional synthetic interface examples using SMOTE (30); and (3) balancing first performing over-sampling of interface residues and then under-sampling of non-interface residues. On the other hand, the training time and the performance of several classifiers may be affected by the representation of numerical data. Hence, experiments with raw data and with data scaled to the interval [-1,1] were performed. Given that RASA values can lie in the interval [0,100], the normalization of these values was computed as  $\frac{RASA}{50} - 1.0$ . The transformation of temperature factor values was performed for each individual protein as  $\frac{2 \times (bFactor - minBfactor)}{maxBfactor - minBfactor} - 1$ , where *bFactor* was the temperature factor for each residue, and *minBFactor* and *maxBFactor* were the smallest and largest temperature factor values in the protein respectively. The impact of different choices of the parameters previously described in the performance of the prediction of interaction sites are discussed in the next section.

### 3.3.7 Performance evaluation

We evaluated the performance of protein-protein interface residues predictors using the following metrics:

$$Accuracy = \frac{TP + TN}{N}$$

$$Precision_+ = \frac{TP}{TP + FP}$$

$$Recall_+ = \frac{TP}{TP + FN}$$

$$Precision_- = \frac{TN}{TN + FN}$$

$$Recall_- = \frac{TN}{TN + FP}$$

$$CC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FN) \times (TP + FP) \times (TN + FP) \times (TN + FN)}}$$

Where  $TP$  denotes the number of residues belonging to the interface that are correctly classified,  $TN$  residues that does not belong to the interface and are correctly classified,  $FP$  misclassified residues that does not belong to the interface,  $FN$  misclassified residues that belong to the interface, and  $N = TP + TN + FP + FN$ <sup>2</sup>.  $Precision_+$  refers to the precision of the classification of interface residues and  $Precision_-$  to the precision of the classification of non-interface residues. Similar notation is used for  $Recall$ .  $CC$  refers to the Matthews correlation coefficient.

We evaluate the performance of a prediction of protein-protein interface patches using *overlap*, that is defined as:

$$Overlap(p) = \frac{|obs \cap res_p|}{|int_r|}$$

where  $obs$  is the set of observed interface residues in the protein,  $res_p$  is the set of residues in patch  $p$ ,  $int_r$  is the set of observed interface residues in a representative interface patch, and  $|\circ|$  denotes the number of elements in the set  $\circ$ .  $Overlap$  is equivalent to *RelativeOverlap* in (94; 90), and according to their work, we consider a prediction successfully if  $overlap \geq 0.7$ . However, a more informative performance evaluation is provided in some sections of this paper using *overlap curves* in which each point corresponds to the number (or percentage) or predictions that are considered correct for a specific value of *overlap*.

---

<sup>2</sup>To decide whether or not a residue is predicted as an interface residue, these terms were computed using the default threshold (0.5) on the scores generated by the predictors of interface residues.

The performance of predictors of protein-protein interface patches has also been evaluated in the literature using the following measures:

$$\begin{aligned} \textit{Specificity}(p) &= \frac{|\textit{obs} \cap \textit{res}_p|}{|\textit{res}_p|} \\ \textit{Sensitivity}(p) &= \frac{|\textit{obs} \cap \textit{res}_p|}{|\textit{obs}|} \end{aligned}$$

*Specificity* was defined in (22; 153) and is equivalent to *Precision* in (21). *Sensitivity* was defined in (22) and is equivalent to *Coverage* in (21) and to *PercentageOverlap* in (94; 90). However, the use of these measures may be misleading because they are very sensitive to the size of the patch: Given that small patches may generate low *sensitivity* and high *specificity* values, whereas large patches may cause low *specificity* and high *sensitivity* values, a perfect prediction (i.e. a prediction that covers the same number of interface residues than those in a representative interface patch) may have low *sensitivity* and/or *specificity*, so it could be refused as a correct prediction. Therefore, we used these two measures for the sake of completeness in the comparison of the performances of *PoInterS*, *SHARP*<sup>2</sup>, and *PPI-Pred*.

The predictors of interface residues and interface patches were evaluated using leave-one-protein-out cross validation experiments and also using independent testing datasets. In both cases, the performance measures were computed using the set of residues composed of all the residues in all the proteins in the testing dataset. A detailed description of the experimental conditions of each experiment is provided in the following section .

The probability that a randomly chosen patch correctly predicts interaction sites depends on the size of the patch (e.g. the probability of randomly finding an interface patch is larger for patches covering half of the protein residues than for patches composed by only one residue). This probability may be determined using the method originally described in (22; 21): First, we compute the probability  $p$  of randomly finding a patch that satisfies the definition of a successful prediction as the number of patches that comply with the success definition divided by the total number of patches in a protein. Then, we compute the probability of success when  $i$  predicted patches are selected as the result of the prediction as  $P = 1 - (1 - p)^i$ .

### 3.4 Experiments and results

In this section we present the results of several experiments carried out to build a *PoInterS*-based PPIPP and to compare the performance of this PPIPP against that of *SHARP*<sup>2</sup> and *PPI-Pred*, two predictors of interface patches that are available as Web servers.

We used two steps to build a *PoInterS*-based predictor of interface patches. First, we carried out several leave-one-protein-out cross validation experiments using the *CV* dataset to determine the effect that different configurations commonly used to build PPIRP and different patch sizes and ranking schemes have in the performance of the PPIPPs. We then used the *ZDOCK* dataset to validate the results obtained in the cross-validation experiments and to study the relationship between the performance of different PPIRPs and the performance of the corresponding PPIPP. Based on the results of these experiments, we selected a PPIPP based on a SVM predictor of protein-protein interface residues (*PoInterS-SVM*). We compare the performances of *PoInterS-SVM*, *SHARP*<sup>2</sup>, and *PPI-Pred* using the 24 proteins of the *Capri* dataset.

#### 3.4.1 Cross-validation experiments

To evaluate several *PoInterS* predictors based on various PPIRPs, we conducted several leave-one-protein-out cross-validation tests on the 220 proteins of the cross validation dataset (CV). For producing different PPIRPs we used: (i) three machine learning algorithms (NB, DT, and LR<sup>3</sup>); (ii) four training sampling techniques on the data used to train the PPIRP (as described in the Methods section); (iii) two representations of the data (sequence-based and structure-based). In addition, we experimented with two patch sizes and different approaches for ranking surface patches using the results of the PPIRP predictors.

#### Performance evaluation of *PoInterS* using different sequence-based PPIRPs

As explained in the methods section, a sequence-based predictor uses a sliding window of contiguous residues in the protein sequence as inputs for predictors of interface residues. The

---

<sup>3</sup>SVM algorithms were not considered for these experiments because the time needed to train models in this cross-validation experiments was prohibitive (around a week for each model).

output of these predictors is a probability or a binary label indicating whether each residue is or not an interface residue. These outputs are used to rank all the patches from a query protein, and the top three non overlapping ranked patches are returned as the most likely to be interaction sites. The prediction is considered as a success if the *overlap* between any of the top three patches and the real interaction site is at least 0.7.

In order to determine the best sequence-based *PoInterS* predictors, 48 leave-one protein-out tests (corresponding to all the combinations of classifiers, ranking schemes and sampling techniques) were performed using the following experimental settings: Each residue is represented using the residues in a window of nine amino acids in the sequence; the maximum overlapping allowed between any pair of the ranked top three patches is 30%; and the size of the patch is  $1.91n^{0.55}$ , where  $n$  is the number of residues of the monomer. Table 3.1 summarizes the results of the best sequence-based *PoInterS* classifiers using three PPIRPs (NB, DT, and LR) on the 220 proteins in the cross validation datasets using different ranking schemes and data sampling techniques.

The best performance was obtained using LR as PPIRP, applying under sampling on the training data, and using the *Class<sub>s</sub>* ranking scheme. Predictions using NB outperformed those using DT for all cases, and those using LR when oversampling was used. Predictions using under sampling were significantly better than predictions with raw data or using only over sampling. In particular, the difference between LR with under sampling (136 correctly predicted monomers) and LR using no sampling technique (100 correctly predicted monomers) emphasizes the bias towards non-interface residues that are introduced in the prediction when the dataset is unbalanced.

The results of a second experiment to study the effect of the window size parameter in the prediction performances are shown in Figure 3.2. The results show that the size of the sliding sequence window affects the performance of the PPIPP based on the LR classifier. A window of nine residues provided the best overall performance. In addition, predictions considering only the top ranked patch accounted for 57% to 64% of the correct predictions for the different window sizes, whereas the contributions of the second and third patches combined ranged from 36% to 43%.

Table 3.1 **Cross-validation results using sequence-based features in windows of 9 residues.** The *ranking* column refers to the method using to rank the patches. The last four columns show the number of successful predictions and their corresponding percentage in reference to the 220 monomers in the dataset. “*Patch 1*” refers to the results when only the top ranked patch is considered. Similarly, “*Patch 2*” and “*Patch 3*” refer to the evaluation using the second and third top-ranked patches respectively. The table has been divided on four blocks depending on the sampling technique used on the training dataset and indicated in the third column.

Classifier	Ranking	Sampling	Successful predictions (%)	Patch 1 (%)	Patch 2 (%)	Patch 3 (%)
NB	<i>Class<sub>s</sub></i>	None	98 (44.55)	33 (33.67)	29 (29.59)	36 (36.73)
DT	<i>Prob<sub>s</sub></i>	None	95 (43.18)	33 (34.74)	42 (44.21)	20 (21.05)
LR	<i>Class<sub>s</sub></i>	None	100 (45.45)	41 (41.00)	29 (29.00)	30 (30.00)
NB	<i>Class<sub>s</sub></i>	Under	131 (59.55)	63 (48.09)	38 (29.01)	30 (22.90)
DT	<i>Prob<sub>s</sub></i>	Under	127 (57.73)	74 (58.27)	39 (30.71)	14 (11.02)
<b>LR</b>	<i>Class<sub>s</sub></i>	<b>Under</b>	136 (61.82)	77 (56.62)	35 (25.74)	24 (17.65)
NB	<i>Prob<sub>s</sub></i>	Over	109 (49.55)	59 (54.13)	23 (21.10)	27 (24.77)
DT	<i>Class<sub>s</sub></i>	Over	104 (47.27)	35 (33.65)	34 (32.69)	35 (33.65)
LR	<i>Class<sub>s</sub></i>	Over	105 (47.73)	46 (43.81)	33 (31.43)	26 (24.76)
NB	<i>Class<sub>s</sub></i>	Over&Under	131 (59.55)	63 (48.09)	38 (29.01)	30 (22.90)
DT	<i>Prob<sub>s</sub></i>	Over&Under	125 (56.82)	71 (56.80)	35 (28.00)	19 (15.20)
LR	<i>Class<sub>s</sub></i>	Over&Under	125 (56.82)	67 (53.60)	32 (25.60)	26 (20.80)

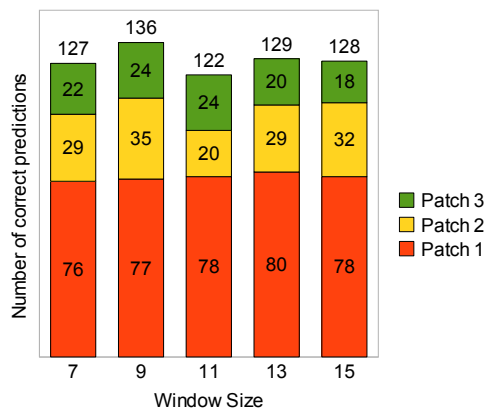


Figure 3.2 **Prediction results using different window sizes for the sequence-based LR predictor.** These results correspond to experiments with the cross-validation dataset using patches of size  $1.91n^{0.55}$ . “Patch 1” refers to the results obtained considering the top-ranked patch.

### Performance evaluation of *PoInterS* using different structure-based PPIRPs

In structure-based classifiers, each residue was represented with information associated with the residue and its  $n$ -nearest surface neighbors. Such information included relative accessible surface area, residue identity, secondary structure and B-factor. A total of 96 different combinations of classifiers, ranking schemes, sampling techniques and representation of numerical values (i.e. normalized and non-normalized) were evaluated using leave-one-protein-out test on the *CV* dataset. The performance comparison of the best 12 predictors is given in Table 3.2. The best performance was achieved by a predictor that used a LR PPIRP trained with under-sampled and normalized data, that ranked the patches using the  $Prob_s$  scheme. This table also shows that predictors based on LR outperform those based on NB and DT for all sampling techniques except over sampling, where the performance of PPIPP based on DT is superior.

We also analyze the effect that the size of the structure window has in the performance of the interface patch predictor based on LR. Figure 3.3 shows that *PoInterS* predictors using structure-based LR PPIRP seem to be less sensitive to the window size parameter than the sequence-based predictors. However, when only the information of the two top-ranked patches is considered, the best performance is achieved using patches of nine residues.



Table 3.2 **Cross-validation results using structure-based features in windows of 9 residues.** The experiments were performed on 220 monomers using patches of size  $1.91n^{0.55}$ . Each of the last four columns show the number and the percentage of successful predictions in reference to the 220 monomers. The table has been divided in four blocks depending on the sampling technique used for the training dataset. Column *Norm* indicates whether the numerical data was or not normalized. Other column labels are explained in Table 3.1.

Classifier	Ranking	Normalized?	Sampling	Successful predictions (%)	Patch 1 (%)	Patch 2(%)	Patch 3(%)
NB	<i>Prob<sub>s</sub></i>	No	None	160 (72.73)	119 (74.38)	27 (16.88)	14 (8.75)
DT	<i>Class<sub>s</sub></i>	Yes	None	166 (75.45)	108 (65.06)	37 (22.29)	21 (12.65)
LR	<i>Prob<sub>s</sub></i>	Yes	None	170 (77.27)	123 (72.35)	33 (19.41)	14 (8.24)
NB	<i>Prob<sub>s</sub></i>	No	Under	159 (72.27)	110 (69.18)	31 (19.50)	18 (11.32)
DT	<i>Prob<sub>s</sub></i>	No	Under	161 (73.18)	119 (73.91)	26 (16.15)	16 (9.94)
<b>LR</b>	<i>Prob<sub>s</sub></i>	<b>Yes</b>	<b>Under</b>	180 (81.82)	142 (78.89)	21 (11.67)	17 (9.44)
NB	<i>Class<sub>s</sub></i>	Yes	Over	156 (70.91)	113 (72.44)	29 (18.59)	14 (8.97)
DT	<i>Prob<sub>s</sub></i>	Yes	Over	166 (75.45)	122 (73.49)	29 (17.47)	15 (9.04)
LR	<i>Class<sub>s</sub></i>	Yes	Over	164 (74.55)	119 (72.56)	25 (15.24)	20 (12.20)
NB	<i>Prob<sub>s</sub></i>	Yes	Over&Under	153 (69.55)	109 (71.24)	30 (19.61)	14 (9.15)
DT	<i>Prob<sub>s</sub></i>	Yes	Over&Under	166 (75.45)	123 (74.10)	31 (18.67)	12 (7.23)
LR	<i>Prob<sub>s</sub></i>	No	Over&Under	174 (79.09)	127 (72.99)	30 (17.24)	17 (9.77)

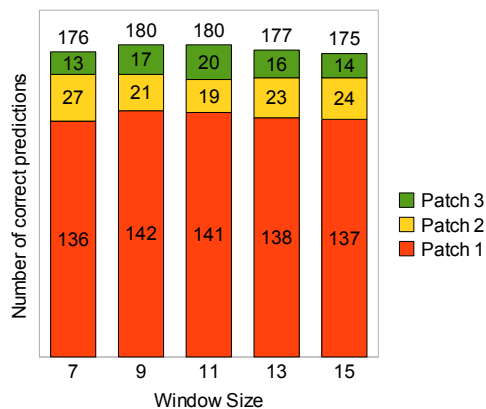


Figure 3.3 Prediction results using different window sizes for the best structure-based LR classifier.

### Structure-based classifiers outperform sequence-based classifiers

Results in Tables 3.1 and 3.2 indicate that the performance of *PoInterS* predictors based on structure-based PPIRPs were superior than those based on sequence-based PPIRPs. Our analysis of the predicted interface residues in several proteins suggests that sequence-based methods tend to generate more false positive predictions than structure-based methods in large non-interacting areas on the protein surface, misleading the process used to rank patches. This is illustrated by the example shown in Figure 3.4. Based on this observation, we chose to use structure-based PPIRPs in the rest of our experiments.

### Evaluation of the performance of *PoInterS* predictors using two patch sizes

The results of experiments using the two patches sizes proposed in (92; 94; 90) on the performance of *PoInterS* using a structure-based LR classifier are presented in Figure 3.5. These results indicate that the overall performance of the predictor that used patches of size  $1.91n^{0.55}$  was similar to those of the predictor that used patches of size  $1.92n^{0.56}$ . However, when only the top-ranked patch was considered, the predictor that used patches of size  $1.91n^{0.55}$  achieved the best performance. These result agree with the findings of Jones and Murakami (90) using a dataset composed of 256 examples. Therefore, most of the results presented in the next sections

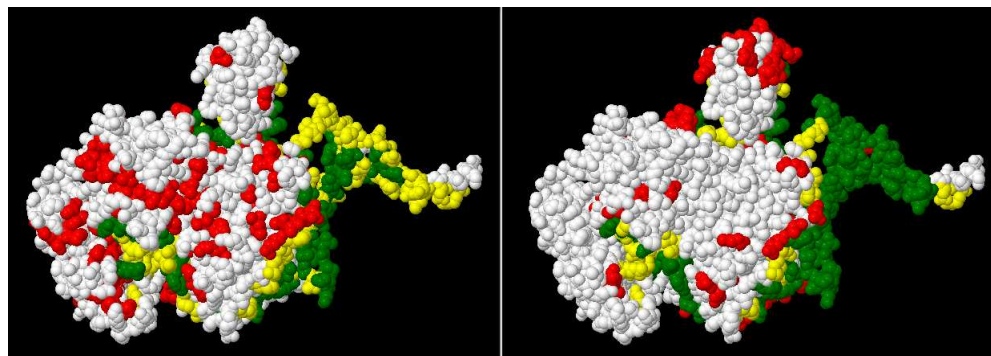


Figure 3.4 **Example of prediction of interface residues using structure and sequence based protein interface residue predictors.** Correct predictions are shown in green and white (TP and TN respectively) whereas incorrect predictions are displayed in red and yellow (FP and FN respectively). Sequence-based predictions are presented on the left and structure-based predictions are displayed on the right. Predictions were performed on the metalloenzyme pyruvate: ferredoxin oxidoreductase (PDB:1KEK, chain A).

were computed using patches of size  $1.91n^{0.55}$ .

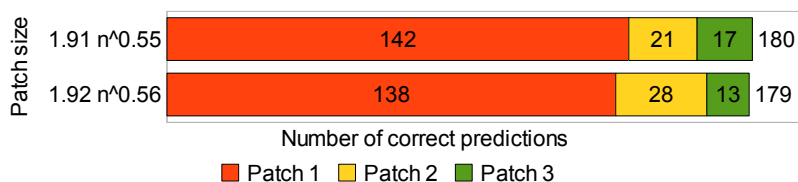


Figure 3.5 ***PoInterS* prediction results using the structure-based LR classifier on two patch sizes.** These results were obtained using a window of size nine.

### Scoring schemes based on estimated probabilities overcomes those based on the predicted binary labels

We proposed four schemes to rank the patches using the results produced by interface residues predictors:  $prob_s(p)$  uses the predicted probabilities that indicates whether the residue is or not an interface whereas  $class_s(p)$  uses the predicted binary labels. The schemes  $probDis_s(p)$  and  $classDis_s(p)$  weight the probabilities and prediction labels according to the inverse of the distances from each residue in the patch to the central residue of the patch. We evaluated the effect that these ranking schemes produce on *PoInterS* predictors using structure-based LR classifiers with windows of size nine and patches of size  $1.91n^{0.55}$ . The results, presented in

Figure 3.6, indicate that the performance obtained using ranking schemes based on the probabilities are superior to that of ranking schemes based on the binary classification labels. This difference is specially large when only the top-ranked patches are considered. In addition, the performance of the schemes that use weights are inferior to those that do not use weights when only the top-ranked patches are considered. In light of these results, we concluded that the ranking scheme that produced the best classification of interaction sites in this dataset was  $prob_s(p)$ .

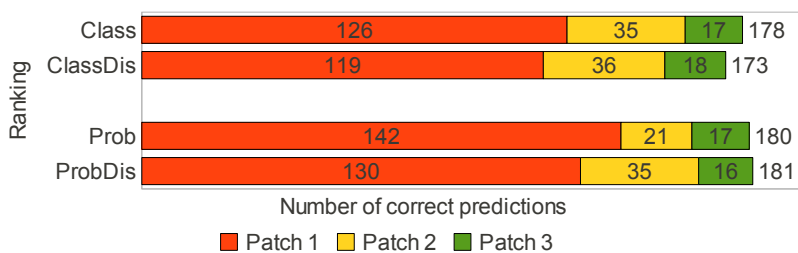


Figure 3.6 **Prediction results for the structure-based LR classifier using four different schemes to rank patches.** The predictions were performed based on LR PPIRP, using patches of size  $1.91n^{0.55}$ , and structural windows of size nine.

In summary, the results of the leave-one-protein-out cross-validation experiments indicate that the best performance of a *PoInterS* classifier was achieved using a structure-based LR protein-protein interface residues predictor trained using normalized and under sampled data and windows of nine residues. The best ranking scheme for this dataset is based on the probabilities generated by the LR PPIRP on patches of size  $1.91n^{0.55}$ . Some of these results were validated in the next section using a blind test dataset.

### 3.4.2 Validation with the ZDOCK dataset

We used the ZDOCK dataset, composed of 299 proteins derived from the docking benchmark 3.0 (85), to compare predictors of interface patches based on SVM, LR, and *SPPIDER* (155), a method to predict protein-protein interface residues. We also studied the relationship between the performance of interface residues predictors and the performance of interface

patches predictors. All the PPIRPs used in this section were trained with the cross validation dataset and tested on the ZDOCK dataset.

As mentioned before, to train SVM predictors with thousand of instances in the leave-one-protein-out cross validation experiments is very time consuming. Therefore, we did not consider *PoInterS* predictors based on SVM PPIRPs in the cross validation experiments. Here, we trained a new structure-based SVM predictor of interface residues using the entire cross validation dataset and used it to generate two predictors of interface patches: One for patches of size  $1.91n^{0.55}$  and the other for patches of size  $1.92n^{0.56}$ .

We evaluated the performance of the interface patches predictors based on SVM and LR PPIRPs using the ZDOCK dataset. The results of these experiments are presented in Figure 3.7, that also include, as a reference, the performances obtained with the LR-based *PoInterS* predictors in the leave-one-protein-out experiments. The results indicate that interface patches predictors based on the SVM PPIRP outperformed those based on the LR predictor on the ZDOCK dataset. In addition, a comparison of the *PoInterS* predictors based on LR PPIRPs indicates that the performance obtained from the ZDOCK dataset was lower than the same on the cross validation dataset. This difference may be explained by the fact that around 90% of the dimers in the cross validation dataset are homo-dimers (i.e. the sequence identities between the interacting proteins is  $\geq 95\%$ ) whereas the proteins in ZDOCK are hetero-dimers (i.e. sequence identities  $\leq 22.81\%$ ).

Given that the complexes in the ZDOCK dataset are hetero-dimers whereas most of the proteins in the cross validation dataset are homo-dimers, we evaluated the impact of the patches ranking schemes in the performance of the PPIPPs using ZDOCK. The result of these evaluations, shown in Figure 3.8, indicate that PPIPPs that use *ProbDis<sub>s</sub>* and *ClassDis<sub>S</sub>* to rank the patches outperform those that use *Prob<sub>s</sub>* and *Class<sub>S</sub>*. These results differ from the obtained in the leave-one-protein-out cross-validation experiment, suggesting that the distribution of interface residues in the patches are different for homo-complexes and for hetero-complexes.

The modular nature of the *PoInterS* method allows to use the predictions generated by any

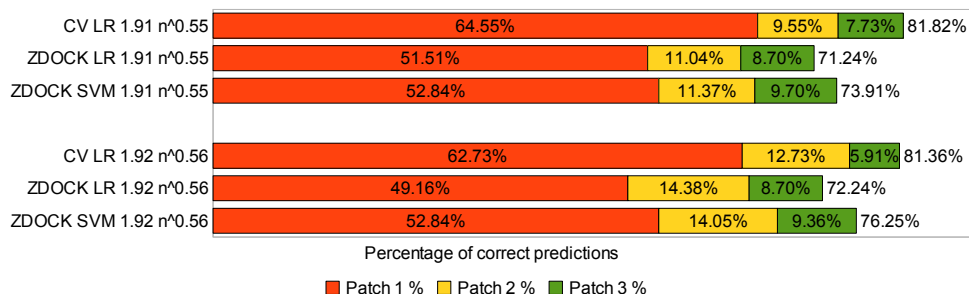


Figure 3.7 **Comparison of percentages of correct predictions using the ZDOCK and the cross validation datasets.** CV refers to the results of the leave-one-protein-out experiments in which the LR were trained and tested using the cross validation dataset. ZDOCK refers to results obtained using structure-based LR and SVM PPIRPs trained with the cross-validation dataset and tested on the ZDOCK dataset. The PPIRPs were built using structural windows of size 9, and normalizing the data. Patches were ranked using the probabilities generated by the PPIRP. Results are grouped into two sections corresponding to different patch sizes.

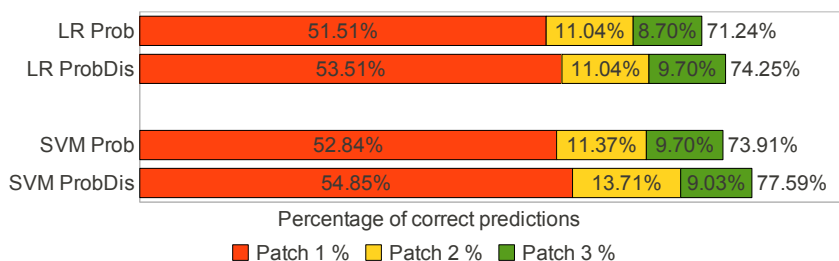


Figure 3.8 **Results of *PoInterS* predictions on the ZDOCK dataset using different ranking schemes.** These experiments were performed with normalized data, structure-based PPIRPs using a window with nine residues, and patches of size  $1.91n^{0.55}$ .

PPIRP to predict interface patches. Therefore, we created several PPIRPs based on *SPPIDER* (155) as an alternative to the structure-based PPIRPs that we defined before. *SPPIDER* was chosen because its high performance reported in (155; 43; 11). *SPPIDER* is a consensus-based method to predict interface residues that uses the results generated by 10 neural network classifiers. The inputs to these neural networks are 19 attributes derived from the sequence and the structure of the protein, and from evolutionary profiles. We trained the neural networks of several *SPPIDER* models using ten partitions of the cross validation dataset defined in two ways. The first definition divided the dataset in 10 non-overlapping parts, and each part was used to train a neural network. The second definition divided the dataset in 10 partitions,

and used the data of nine partitions to train a neural network (i.e. each sample in the dataset was used to train nine neural networks). In addition, we trained some *SPPIDER* models using our definition of interface residues for the cross validation dataset (based on loss of accessible surface area), and other models with the definition of interface residues used by the authors of *SPPIDER* to deal with changes in the conformation of a protein structure after complexation. According to this definition, the sequence homologs of a query protein were aligned, and any aligned residue labeled as interface in the query sequence or any of its homologs, was labeled as an interface residue in the query protein<sup>4</sup>.

In this experiment, we used several PPIPPs based on our implementation of *SPPIDER*, SVM, and LR, to evaluate the relationship between the performances of the PPIPPs and those of their corresponding interface patches predictors. The results of this evaluation are presented in Table 3.3. The rows in this table are sorted according to the *overlap* value. These results indicate that the *PoInterS* predictors based on SVM and LR (with exception of the predictor using a SVM model trained with unbalanced data) outperformed all the interface patches predictors based on a *SPPIDER* PPIPP. However, the main observation deduced from the results presented in Table 3.3 is that, given the performance measures of the PPIPP, it is not completely clear how to select which interface residues predictor will produce the best interface patches predictor. For example, the predictor that achieves the highest *overlap* value was ranked sixth according to accuracy; the predictor with the fourth highest precision value was ranked 11<sup>th</sup> according to *overlap*; the predictor with the highest recall value was ranked third according to *overlap*; and the classifier with the third highest correlation coefficient value was ranked sixth according to *overlap*. We conclude that the selection of the interface residues classifier used to predict interaction sites should be done using the *overlap* measure, not a performance measure associated with the PPIPP. However, this is a minor inconvenience given that the method for computing and ranking the patches is efficient.

---

<sup>4</sup>The interface residues of a protein and its sequence homologs were extracted from ProtInDB (<http://protindb.cs.iastate.edu>), a data base of protein-protein interface that can compute interface residues from sequence homologs sharing  $\geq 96\%$  of sequence similarity with the query protein.

Table 3.3 **Performances of the interface residues classifiers and their corresponding interface patches predictors.** “*SPP*<sup>†</sup>” refers to *SPPIDER* predictors trained with 10 datasets generated by partitioning the CV dataset into 10 pieces (i.e. each sample in CV appears in exactly one dataset). “*SPP*” denotes *SPPIDER* classifiers trained with 10 datasets generated from 10-cross validation partitions on the CV dataset (i.e. each sample of CV appears in nine training datasets). “*Train. balanced?*” indicates whether the training dataset was balanced or not. “*Train. dimers?*” and “*Test dimers?*” indicate whether the interface residues in the training and testing datasets, respectively, were extracted from dimers in the query protein or from the interacting chains in complexes with sequence identity  $\geq 96\%$  with the query protein. “*IR*” refers to interface residues and “*NIR*” to non-interface residues. “*Overlap %*” is the percentage of correct predictions for the interface patches predictor. “*Overlap Patch 1 %*” is the percentage of correctly predicted patches when only the top-ranked patch is considered. These experiment were performed with patches of size  $1.91n^{0.55}$  ranked with  $probDis_s(p)$ . Windows of nine residues were used for the SVM and LR classifiers. The data is presented according to the performance of the interface patches predictors.

IR Predictor	Train. balanced?	Train. dimers?	Test dimers?	Normalized?	Accuracy %	Precision IR %	Recall IR %	Precision NIR %	Recall NIR %	CC %	Overlap %	Overlap Patch 1 %
SVM	✓	✓	✓	✓	72.62	35.42	56.15	89.06	76.46	26.87	77.59	70.69
LR	✓	✓	✓	✓	71.20	32.92	56.09	88.93	74.42	24.55	74.25	72.07
LR	✓	✓	✓		66.83	28.17	61.10	88.60	67.59	21.39	73.24	67.58
SPP <sup>†</sup>	✓			✓	74.53	23.15	57.94	95.40	74.01	23.44	65.20	63.73
SPP	✓		✓	✓	69.85	30.16	48.63	87.50	74.37	18.86	64.86	59.38
SPP	✓			✓	75.06	24.08	58.28	95.51	74.69	24.46	64.53	59.16
SPP	✓	✓	✓	✓	70.94	30.52	46.18	87.44	76.27	18.61	64.21	65.10
SPP		✓	✓	✓	78.79	29.92	23.15	85.13	90.66	13.99	62.21	63.44
SPP				✓	85.41	27.75	32.36	93.05	89.71	20.51	61.49	59.34
SPP <sup>†</sup>	✓	✓	✓	✓	69.10	28.79	50.15	87.35	72.64	18.15	59.20	62.71
SPP <sup>†</sup>		✓	✓	✓	77.65	30.28	25.66	85.33	88.59	14.11	58.19	65.52
SVM	✓	✓	✓		45.17	18.98	71.32	85.71	39.60	7.12	45.48	44.12



An evaluation of the performance of the support vector machine-based interface patches predictor, denoted as *PoInterS-SVM*, and the best *SPPIDER*-based interface patches predictors in terms of the *overlap curve* is presented in Figure 3.9. From this figure it is possible to observe that a threshold value of 50% on *overlap* produced around 90% correct predictions for *PoInterS-SVM* and around 85% for the *SPPIDER*-based predictor, whereas a threshold value of 90% produced around 40% and a 25% of correct predictions for *PoInterS-SVM* and for the *SPPIDER*-based predictor respectively. This figure also indicates that about 20% of the *PoInterS-SVM* predictions were successful independently of the threshold value, so the prediction of interaction sites for the proteins involved in these cases could be considered as trivial for *PoInterS-SVM*.

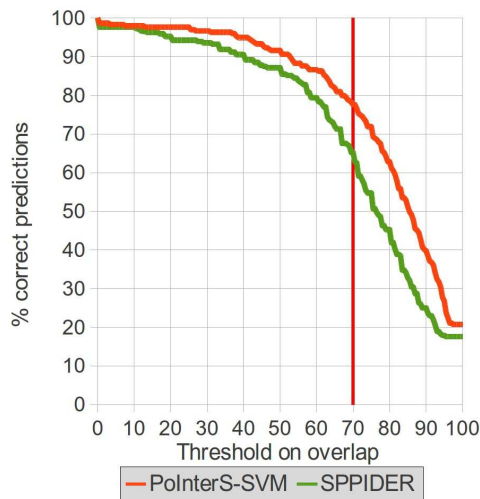


Figure 3.9 **Overlap curves for *PoInterS-SVM* and the best *SPPIDER*-based predictor of interface patches.** Predictions were performed using windows of nine residues and ranking patches of size  $1.91n^{0.55}$  using  $ProbDis_s(p)$ .

### 3.4.3 Comparison with other interface patches predictors

We compared our final proposed predictor of interface patches, *PoInterS-SVM*, with *SHARP*<sup>2</sup> (132) and *PPI-Pred* (22), that are available as Web servers. The comparisons were performed on the *Capri* dataset, composed of 24 proteins extracted from 19 targets in the first eight rounds of CAPRI, as described in the methods section.

### 3.4.3.1 *PoInterS-SVM* versus *SHARP*<sup>2</sup>

*SHARP*<sup>2</sup> predicts interaction sites in a query protein by decomposing its surface into overlapping patches and ranking them using the arithmetic mean of a set of scaled parameters. Two sets of parameters were suggested. The first (94) included solvation potentials, hydrophobicity, accessible surface area, residue interface propensity, protrusion, and planarity. The second set of parameters (90) was composed of solvation potentials, hydrophobicity, accessible surface area, and residue interface propensity. The definition of patches is the same for *SHARP*<sup>2</sup> and *PoInterS*, and patches of size  $1.91n^{0.55}$ , as suggested by Jones and Mukarami in (90), were used for comparing both methods. The predictions of *SHARP*<sup>2</sup> were computed using the available Web server. The performance measures used for the comparison were *overlap*, *specificity*, *sensitivity*, the probability  $p$  of randomly finding a patch that satisfies the definition of success, and the probability *Prob* of finding the  $i$ -th patch, where  $i$  corresponded to the first patch that satisfied the definition of a correct prediction (e.g.  $Prob = 1 - (1 - p)^2$  when the first predicted patch with  $overlap \geq 70\%$  is ranked in the second position) or the patch with the highest *overlap* when none of the three selected patches satisfied such definition. The two sets of parameters of *SHARP*<sup>2</sup> were tested and the best performance in terms of *overlap* was achieved using the set composed of four parameters. Detailed results of the comparison of the best *SHARP*<sup>2</sup> predictor and *PoInterS-SVM* are shown in Table 3.4.

Results in Table 3.4 suggest that *PoInterS-SVM* outperformed *SHARP*<sup>2</sup> on the CAPRI dataset. There were 11 correct predictions according to *overlap* for *SHARP*<sup>2</sup> and 20 for *PoInterS-SVM*. A 65% of these correct predictions were obtained using the top-ranked patch in *PoInterS-SVM*, versus a 36% in *SHARP*<sup>2</sup>. In addition, 12 successful predictions using *PoInterS-SVM* and four using *SHARP*<sup>2</sup> had  $Prob \leq 0.17$ . Using the metrics proposed by Bradford and Westhead in (22) to define successful predictions (i.e. *specificity* > 50 and *sensitivity* > 20) four interaction sites were correctly predicted by *SHARP*<sup>2</sup> whereas *PoInterS-SVM* correctly predicted five. All the *PoInterS-SVM* predictions had  $Prob \leq 0.17$  whereas all the *SHARP*<sup>2</sup> predictions had  $Prob \leq 0.64$ , and one was  $\leq 0.17$ .

Table 3.4 **Comparison of SHARP<sup>2</sup> and PoInterS-SVM.** *PoInterS-SVM* ranked patches of size  $1.91n^{0.55}$  using  $probDis_s(p)$ . *Overl*, *Spec* and *Sens* refers to *overlap*, *specificity*, and *sensitivity* respectively. *Patch* refers to the first patch with  $overlap \geq 70\%$  or to the best among the three top ranked patches if their  $overlap < 70\%$ . The probability of randomly select a patch with  $overlap \geq 70\%$  is denoted by  $p$ , and *Prob* is the probability of randomly finding the patch specified in the column *Patch*.

Target	p	SHARP <sup>2</sup>					PoInterS-SVM				
		Overl.	Spec.	Sens.	Patch	Prob	Overl.	Spec.	Sens.	Patch	Prob
1A	0.16	36.36	12.12	30.77	2	0.30	72.73	24.24	61.54	1	0.16
1H	0.16	100.00	52.17	100.00	1	0.16	100.00	52.17	100.00	1	0.16
2A	0.08	100.00	11.54	100.00	3	0.22	83.33	9.62	83.33	1	0.08
2D	0.11	85.71	16.22	85.71	1	0.11	100.00	18.92	100.00	1	0.11
3A	0.10	71.43	21.74	71.43	3	0.27	92.86	28.26	92.86	1	0.10
3C	0.11	57.14	17.39	57.14	3	0.29	42.86	13.04	42.86	1	0.11
3H	0.07	100.00	23.68	90.00	1	0.07	100.00	23.68	90.00	1	0.07
3L	0.07	55.56	13.51	55.56	1	0.07	100.00	24.32	100.00	1	0.07
4A	0.09	40.91	15.25	39.13	3	0.24	68.18	25.42	65.22	3	0.24
7A	0.07	58.33	18.42	58.33	2	0.14	75.00	23.68	75.00	1	0.07
8A	0.14	87.50	34.15	60.87	2	0.26	75.00	29.27	52.17	1	0.14
8B	0.17	29.41	15.62	22.73	1	0.17	82.35	43.75	63.64	2	0.32
9A	0.15	100.00	63.89	56.10	2	0.28	82.61	52.78	46.34	1	0.15
10A	0.07	34.62	17.65	16.98	3	0.20	76.92	39.22	37.74	3	0.20
11A	0.17	72.22	44.83	50.00	1	0.17	100.00	62.07	69.23	2	0.32
11B	0.40	100.00	55.56	71.43	2	0.64	80.00	44.44	57.14	1	0.40
13F	0.15	64.29	22.50	60.00	3	0.38	85.71	30.00	80.00	1	0.15
14A	0.30	75.00	40.00	34.62	2	0.51	83.33	44.44	38.46	1	0.30
14B	0.20	57.69	34.09	23.08	2	0.36	80.77	47.73	32.31	2	0.36
18A	0.08	38.10	16.00	33.33	3	0.21	100.00	42.00	87.50	2	0.15
18C	0.10	95.65	64.71	68.75	3	0.26	100.00	67.65	71.88	2	0.18
19A	0.06	69.23	36.00	69.23	1	0.06	69.23	36.00	69.23	1	0.06
20A	0.08	31.03	20.93	20.00	3	0.21	65.52	44.19	42.22	3	0.21
20B	0.11	33.33	20.45	24.32	3	0.30	100.00	61.36	72.97	2	0.21

An evaluation of both classifiers in terms of the *overlap curves*, presented in Figure 3.10, indicates that *PoInterS-SVM* outperformed *SHARP<sup>2</sup>* for *overlap* threshold values greater than 30%.

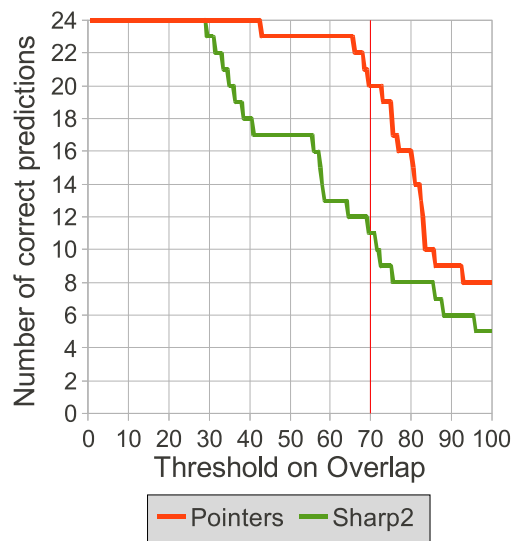


Figure 3.10 **Comparison of *SHARP<sup>2</sup>* and *PoInterS-SVM* using *overlap curves*.** The horizontal axis indicate different *overlap* percentage values whereas the vertical axis shows the number of correct predictions achieved according to the *overlap* percentage value.

### *PoInterS-SVM* versus *PPI-Pred*

*PoInterS-SVM* and *PPI-Pred* (22) have several similarities and differences. Both methods use machine learning classifiers to rank every patch on the protein, and top-ranked non-overlapping patches are returned as the predicted interaction sites. However, *PPI-Pred* uses a SVM classifier of interface patches whereas *PoInterS-SVM* uses a SVM predictor of interface residues; the ranking of *PPI-Pred* is directly produced by the SVM classifier whereas *PoInterS* uses a ranking scheme derived from the prediction performed on each residue in every patch; and the definitions of patches are different for both methods. Specifically, the SVM model used by *PPI-Pred* was trained with an interacting patch and a non-interacting patch extracted from each protein in a dataset of 180 proteins. The prediction generated by this SVM is used to rank

every patch on a query protein. In addition, *PPI-Pred* defines a basic patch as a central atom and the set of atoms included in a sphere centered in the central atom. This basic patch may be extended when it forms a ring on the surface of the protein, or reduced to avoid the inclusion of residues in different sides of the protein or when unconnected patches are formed inside the sphere. Therefore, the patches of a protein may have different sizes, which complicates the task of comparing *PPI-Pred* predictions with ours. Hence, the comparison for each protein was performed using only the top-ranked patch generated by *PPI-Pred* and the top-ranked patch produced by *PoInterS-SVM*. The patches of *PoInterS-SVM* were composed of the same number of residues as the top-ranked patch of *PPI-Pred*. *PPI-Pred*'s predictions were computed using the Web server. The performances of the predictions of *PPI-Pred* and *PoInterS-SVM* on the ZDOCK dataset are shown in Table 3.5. These results indicate that *PoInterS-SVM* produced correct predictions for 12 proteins whereas *PPI-Pred* succeeded in three. Eleven of the correct predictions of *PoInterS-SVM* and the four predictions of *PPI-Pred* had  $p \leq 0.17$ , and the additional correct prediction of *PoInterS* had  $p = 0.29$ . Hence, the number of interaction site predictions performed by *PoInterS-SVM* was almost three times the number of predictions of *PPI-Pred* on this dataset when only the first top ranked patch was used.

Using the definition of successful predictions proposed by the authors of *PPI-Pred* (22) (i.e. *specificity* > 50% and *sensitivity* > 20%), four correct predictions were generated using *PoInterS-SVM* and three using *PPI-Pred*. Three of the four predictions of *PoInterS-SVM* and the three predictions of *PPI-Pred* had  $p \leq 0.17$ . In addition, 10 of the predictions performed with *PPI-Pred* produced zero values for *specificity* and *sensitivity*, whereas *PoInterS-SVM* produced only one. Therefore, when *specificity* and *sensitivity* were used as performance measures, *PoInterS-SVM* results were at least as good as *PPI-Pred* for this dataset.

A comparison between both classifiers using the *overlap curves*, presented in Figure 3.11, indicates that *PoInterS-SVM* outperformed *PPI-Pred* for the dataset extracted from CAPRI.

Table 3.5 **Comparison using the first patch predicted with *PPI-Pred* and the first patch predicted with *PoInterS-SVM*.** *Patch size* refers to the number of residues in the top-ranked patch computed by *PPI-Pred* and used in *PoInterS-SVM*. The probability of randomly selected a patch with  $overlap \geq 70\%$  is denoted by  $p$ . *Overl*, *Spec* and *Sens* refers to *overlap*, *specificity*, and *sensitivity* respectively. Patches were ranked in *PoInterS* using  $probDis_s(p)$ .

Target	Patch size	p	<i>PPI-Pred</i>			<i>PoInterS-SVM</i>		
			Overl.	Spec.	Sens.	Overl.	Spec.	Sens.
1A	25	0.09	0.00	0.00	0.00	63.64	28.00	53.85
1H	11	0.10	63.64	63.64	58.33	72.73	72.73	66.67
2A	24	0.04	0.00	0.00	0.00	0.00	0.00	0.00
2D	34	0.11	0.00	0.00	0.00	100.00	20.59	100.00
3A	39	0.08	0.00	0.00	0.00	92.86	33.33	92.86
3C	32	0.06	0.00	0.00	0.00	76.92	31.25	71.43
3H	25	0.05	44.44	16.00	40.00	100.00	36.00	90.00
3L	30	0.06	0.00	0.00	0.00	66.67	20.00	66.67
4A	44	0.05	0.00	0.00	0.00	57.14	27.27	52.17
7A	23	0.07	0.00	0.00	0.00	70.00	30.43	58.33
8A	42	0.17	0.00	0.00	0.00	75.00	28.57	52.17
8B	40	0.17	60.00	30.00	54.55	20.00	10.00	18.18
9A	34	0.17	86.36	55.88	46.34	81.82	52.94	43.90
10A	91	0.16	32.35	12.09	20.75	50.00	18.68	32.08
11A	16	0.07	0.00	0.00	0.00	53.85	43.75	26.92
11B	9	0.20	25.00	22.22	14.29	62.50	55.56	35.71
13F	43	0.16	42.86	13.95	40.00	85.71	27.91	80.00
14A	76	0.29	39.39	17.11	25.00	72.73	31.58	46.15
14B	50	0.18	58.62	34.00	26.15	34.48	20.00	15.38
18A	54	0.09	80.95	31.48	70.83	80.95	31.48	70.83
18C	19	0.05	100.00	84.21	50.00	18.75	15.79	9.38
19A	23	0.06	53.85	30.43	53.85	100.00	56.52	100.00
20A	35	0.06	48.00	34.29	26.67	4.00	2.86	2.22
20B	99	0.21	68.57	24.24	64.86	31.43	11.11	29.73

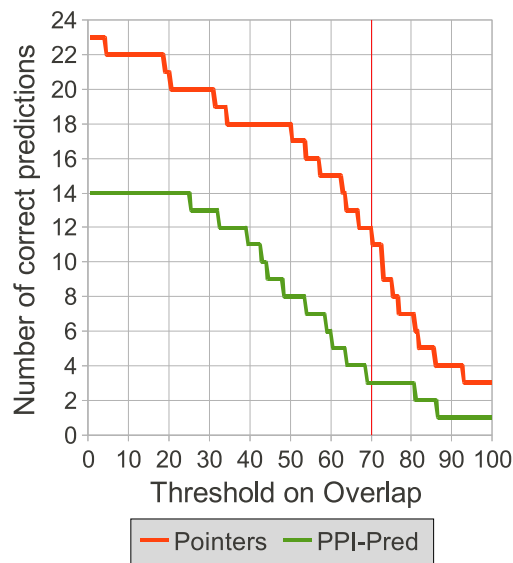


Figure 3.11 Comparison of *PPI-Pred* and *PoInterS-SVM* using *overlap curves*. The curves were generated using the predictions corresponding to the top-ranked patch for each protein in the dataset.

#### 3.4.4 Web server

*PoInterS-SVM* has been implemented as a web server, which is freely available at <http://pointers.cs.iastate.edu>. The server accepts the PDB Id of the query protein or a file with the structure of the protein, the name of protein chain, the size of the patch, the ranking scheme to use, and the maximum allowed overlapping between patches as inputs, and produces as output, a list of the PDB residue codes in each predicted interaction site and a graphical representation of the predicted patches using Jmol(1). The server also allows batch submissions of a list of PDB protein chains on which the user wants to obtain predictions of interface sites.

### 3.5 Conclusions

We presented *PoInterS*, a modular method for predicting protein-protein interaction sites in unbound proteins based on the results produced by interface residues predictors. *PoInterS* computes all the patches on the surface of a given unbound protein and rank them using the results generated by interface residues predictors. Finally, a set of top-ranked patches with overlap  $\leq 30\%$  is returned as the predicted interface patches.

We conducted experiments to evaluate factors that affected the performance of the predictor such as different machine learning prediction methods, patch sizes, structural and sequential representation of each residue in the protein, sampling techniques on the training datasets, normalization of the data, and different schemes to rank patches. As a consequence, we selected a predictor of interaction sites, *PoInterS-SVM*, and implemented it as a Web application. *PoInterS-SVM*, is based on a support vector machine model for prediction of interface residues that was trained using a dataset of 220 non-homologous proteins. Every surface residue in *PoInterS-SVM* is represented using the amino acid identity, secondary structure, crystallographic temperature factor and relative accessible surface area of the residue and its eight nearest neighbors in the structure surface.

Experiments comparing the performance of *PoInterS-SVM*, *SHARP*<sup>2</sup> and *PPI-Pred*, show that our proposed method for predicting protein-protein interface patches using predicted protein-protein interface residues leads to improvements over *SHARP*<sup>2</sup> and *PPI-Pred*. In particular, using a dataset of 24 protein extracted from 19 targets of the first eight rounds of CAPRI, *PoInterS-SVM* successfully predicted 71% of the interaction sites whereas *SHARP*<sup>2</sup> was successful on 46% when the three top-ranked patches were considered. When only the top-ranked patch of each protein was considered, *PoInterS-SVM* correctly predicted 42% of the interaction sites whereas *PPI-Pred* was successful for 13%. The validity of our approach to predict interface patches using predicted interface residues is also supported by the results of an evaluation of the performance of *PoInterS-SVM* computed on a blind dataset of 299 proteins extracted from ZDOCK Benchmark 3.0, achieving a 76% of success in terms of *overlap*.

The modular nature of *PoInterS* allows the use of interface residue predictions from any available method for ranking the patches and predicting the interaction sites. Because the processes of constructing, ranking and selecting the patches in *PoInterS* are relatively fast, it is possible to experiment with different patch sizes or conformations given the predicted interface residues for proteins of interest. This feature is especially useful in light of the fact that it is not easy to determine which residue predictor among a set of candidate predictors is likely to produce a better interface patch predictor based simply on the estimated performance of the interface residue predictors.



Future work aims to improve the prediction of protein-protein interface residues classifiers as a way to improve the prediction of interface patches, to create different schemes to construct or aggregate patches, and to develop of applications of the *PoInterS* predictors in problems such as prediction of conformational B-cell epitopes.

### 3.6 List of abbreviations

SVM - Support vector machine

NB - Naive Bayes

LR - Logistic regression

CV - Cross validation dataset

*PoInterS* - Method for prediction of interaction sites

*PoInterS-SVM* - Predictor of interaction sites based on a SVM interface residues classifier.

PPIRP - Protein-protein interface residues predictor.

PPIPP - Protein-protein interface patch predictor.

### 3.7 Authors contributions

All the authors have directly participated in the design, implementation, or analysis of this research paper.

### 3.8 Acknowledgements

This work was funded in part by the National Institutes of Health grant GM066387 to Vasant Honavar and Drena Dobbs and in part by a research assistantship funded by the Center for Computational Intelligence, Learning, and Discovery. The work of Vasant Honavar while working at the National Science Foundation was supported by the National Science Foundation. Any opinion, finding, and conclusions contained in this article are those of the authors and do not necessarily reflect the views of the National Science Foundation.

## CHAPTER 4. PREDICTING PROTEIN-PROTEIN INTERFACE RESIDUES USING LOCAL SURFACE STRUCTURAL SIMILARITY

*Paper originally published in BMC Bioinformatics, vol 13, 2012*

Rafael A. Jordan, Yasser EL-Manzalawy, Drena Dobbs and Vasant Honavar

### 4.1 Abstract

**Background.** Identification of the residues in protein-protein interaction sites has a significant impact in problems such as drug discovery. Motivated by the observation that the set of interface residues of a protein tend to be conserved even among remote structural homologs, we introduce *PrISE*, a family of local structural similarity-based computational methods for predicting protein-protein interface residues.

**Results.** We present a novel representation of the surface residues of a protein in the form of structural elements. Each structural element consists of a central residue and its surface neighbors. The *PrISE* family of interface prediction methods use a representation of structural elements that captures the atomic composition and accessible surface area of the residues that make up each structural element. Each of the members of the *PrISE* methods identifies for each structural element in the query protein, a collection of *similar* structural elements in its repository of structural elements and weights them according to their similarity with the structural element of the query protein. *PrISE<sub>L</sub>* relies on the similarity between structural elements (i.e. local structural similarity). *PrISE<sub>G</sub>* relies on the similarity between protein surfaces (i.e. general structural similarity). *PrISE<sub>C</sub>*, combines local structural similarity and general structural similarity to predict interface residues. These predictors label the central residue of a structural element in a query protein as an interface residue if a weighted majority of

the structural elements that are similar to it are interface residues, and as a non-interface residue otherwise. The results of our experiments using three representative benchmark datasets show that the *PrISE<sub>C</sub>* outperforms *PrISE<sub>L</sub>* and *PrISE<sub>G</sub>*; and that *PrISE<sub>C</sub>* is highly competitive with state-of-the-art structure-based methods for predicting protein-protein interface residues. Our comparison of *PrISE<sub>C</sub>* with *PredUs*, a recently developed method for predicting interface residues of a query protein based on the known interface residues of its (global) structural homologs, shows that performance superior or comparable to that of *PredUs* can be obtained using only local surface structural similarity. *PrISE<sub>C</sub>* is available as a Web server at <http://prise.cs.iastate.edu/>

**Conclusions.** Local surface structural similarity based methods offer a simple, efficient, and effective approach to predict protein-protein interface residues.

## 4.2 Background

Protein-protein interactions play a central role in many cellular functions. In the past decade, significant efforts have been devoted to characterization as well as discovery of these interactions both in silico and in vivo (65; 117; 109; 209; 118). Of particular interest is the identification of the amino acid residues that participate in protein-protein interactions because of its importance in elucidation of mechanisms that underly biological function and rational drug design (among other applications) (56). However, experimental determination of interface residues is expensive, labor intensive, and time consuming (53). Hence, there is an urgent need for computational methods for reliably identifying from the sequence or structure of a query protein, the subset of residues that are likely to be involved in the interaction of that protein with one or more other proteins.

Several methods for predicting protein-protein interface residues have been proposed in the literature (see the reviews in (213; 43; 11)). A variety of features of the target residue (and often its sequence or structural neighbors) have been explored (143; 183) in combination with machine learning techniques (54; 205; 22; 200; 36; 146; 155; 122; 172; 123; 133) or scoring functions (94; 139; 163; 90; 136; 171) to construct predictors of interface residues. Of particular interest are recent methods for protein interface prediction based on the structural similarity

between a query protein and proteins with known structure. These methods are motivated by observations that suggest that interaction sites tend to be conserved among structurally similar proteins (125; 35; 182; 39; 69). As the number of experimentally determined complexes in the Protein Data Bank (PDB) (12) increases, the likelihood of success of such an approach to interface prediction can be expected to increase as well. Hence, there is growing interest in structural similarity based approaches to protein-protein interface prediction. For example, Konc and Janežič (106) and Carl et al. (26) have developed a method that utilizes a graph based representation of protein surfaces to predict interface residues that exploits the higher degree of conservation of topological and physico-chemical features among interaction sites as compared to non-interaction sites of proteins. Zhang et al. (212) have introduced *PredUs*, a new method that predicts interaction sites using counts of interface residues derived from alignments between the structure of a query protein and the structures of a set of proteins that are structurally similar to the query protein. More recently, *PredUs* has been updated (211) to incorporate a support vector machine that uses accessible surface area of regions on the protein surface and the counts of interface residues derived from the structural alignments to predict interface residues.

A potential limitation of structural similarity based interface prediction methods is that they are effective only to the extent that a set of proteins (with experimentally determined interface residues) that are structurally similar to the query protein can be reliably identified. In light of evidence that the degree of conservation of interfaces tends to be substantially higher than that of non-interfaces (125) and hence that of whole protein structures, there is increasing interest in methods for predicting interface residues based on experimentally determined interface residues in proteins that are locally (as opposed to globally) similar in structure to the query protein (107; 27).

Against this background, we introduce *PrISE* (Predictor of Interface Residues using Structural Elements), a novel family of predictors of protein-protein interface residues based on local structural similarity. The *PrISE* family of interface prediction methods utilizes a repository of structural elements constructed from a dataset of proteins that are part of experimentally determined protein complexes retrieved from PDB. A structural element is defined as a pro-

tein surface residue surrounded by its neighbors on the protein surface. The *PrISE* methods utilize a novel representation of each structural element that captures the distribution of the constituent atoms and the solvent accessible surface areas of residues (calculated from the individual proteins). The prediction of protein-protein interface residues using any of the *PrISE* methods is based on the identification of a collection of structural elements in the repository that are *similar* to the structural elements of a query protein. The *PrISE* predictors label the central residue of each structural element in the query protein as an interface residue if a weighted majority of the similar structural elements are interface residues and as a non-interface residue otherwise. *PrISE<sub>L</sub>* relies on the similarity between structural elements to assign the weights to each query structural element whereas *PrISE<sub>G</sub>* relies on the similarity between protein surfaces in terms of structural elements. *PrISE<sub>C</sub>* combines the local and global approaches of *PrISE<sub>L</sub>* and *PrISE<sub>G</sub>*. We assessed the performance of the *PrISE* family of predictors using several benchmark datasets. The results of experiments show that *PrISE<sub>C</sub>* outperforms *PrISE<sub>L</sub>* and *PrISE<sub>G</sub>*. The three *PrISE* family of predictors outperform two other local structural similarity based interface residue predictors (26; 27). *PrISE<sub>C</sub>* outperforms methods that use diverse structural, evolutionary, and physico-chemical properties to perform prediction of interface residues using machine learning and scoring functions, even in the absence of proteins with similar structure. The performance of *PrISE<sub>C</sub>* is superior or comparable to that of *PredUs* (212; 211), a novel method that predict interface residues using the known interface residues on proteins with similar structure to a query protein. Unlike *PredUs*, that require the existence of structural homologs to perform predictions, *PrISE<sub>C</sub>* is able to generate prediction for all the proteins with known structure.

## 4.3 Methods

### 4.3.1 Structural elements and their representation

A *structural element* is defined by an amino acid residue on the protein surface (referred to as a *surface residue*) and its neighboring surface residues. Thus, the number of structural elements in a protein equals the number of its surface residues. An amino acid residue is considered a

*surface residue* if its accessible surface area in the monomer is greater than zero. Two residues are considered neighbors if the distance between the Van der Waals surface of an *atom* of one residue and the Van der Waals surface of an *atom* of the other residue is  $\leq 1.5 \text{ \AA}$ . Accessible surface areas were computed using Naccess (83).

A structural element is represented using four features: (i) The name of the central residue of the structural element; (ii) the accessible surface area of the central residue of the structural element; (iii) the accessible surface area of the structural element (computed as the addition of the accessible surface areas of its residues); and (iv) an histogram of atom nomenclatures representing the atomic composition of the surface of the structural element. An *histogram of atom nomenclatures* contains the count of the number of atoms on the surface of the structural element for each atom nomenclature (e.g. number of  $\alpha$ -carbons, number of  $\beta$ -carbons, etc.). There are 36 atom nomenclatures (a list is presented in section one of the Appendix A), hence, an histogram of atom nomenclatures has 36 bins. An atom is considered to be in the surface of a protein if its accessible surface area is  $> 0 \text{ \AA}^2$ . The four features that represent a structural element are used to define a similarity measure between structural elements that consider structural and physico-chemical properties. The rationale behind this representation, is that structural elements with similar accessible surface areas and centered on identical residues with similar surface areas have similar structure. In addition, two structural elements with similar atomic composition of the surface of the structural element (represented by the histogram of atom nomenclatures) have similar physico-chemical properties.

### 4.3.2 Distance between histogram of atom nomenclatures

The distance between the histograms of atom nomenclatures of two structural elements provides a measure of their physico-chemical similarity. The distance between two histograms of atom nomenclatures  $x$  and  $y$  is computed using the city block metric:  $\sum_{i=1}^{36} |x_i - y_i|$ , where  $x_i$  and  $y_i$  denote the number of atoms (corresponding to the  $i^{th}$  nomenclature in the histograms) on the surface of the two structural elements (e.g. number of  $\alpha$ -carbons exposed to the solvent)<sup>1</sup>.

---

<sup>1</sup>An explanation of the process used to select the city block metric from a set of different metrics is presented in the Appendix A.

### 4.3.3 Repository of structural elements

A *repository of structural elements* stores all the structural elements extracted from a set of proteins. To perform different experiments, we built two repositories from two different sets of proteins. The first, called the *ProtInDb repository*, was built from the biological assemblies stored in ProtInDb (99), a database of protein-protein interface residues, which in turn was derived from protein complexes in PDB (12). This repository is composed of 21,289,060 structural elements extracted from 88,593 interacting chains (as of February 21, 2011). The second repository, called the *ProtInDB  $\cap$  PQS repository*, is composed of the structural elements extracted from proteins that are common to both *ProtInDb* and the Protein Quaternary Structure database (*PQS*) (75). This repository contains 13,396,420 structural elements extracted from 55,974 interacting chains in 21,786 protein complexes. A protein chain is considered an *interacting chain* if it contains at least five *contact* amino acid residues. An amino acid residue in a protein chain is considered a contact amino acid if the Van der Waals surface of at least one of its heavy atoms is no further than at most 0.5 Å from the Van der Waals surface of some heavy atom(s) of an amino acid residue belonging to another chain.

### 4.3.4 Retrieving similar structural elements

The prediction of interface residues in a query protein is based on the existence of similar structural elements for each structural element in the protein. The process of retrieval similar structural elements from a repository of structural elements should satisfy two requirements: It should be efficient and it should retrieve similar structural elements for every structural element in the query protein. These requirements are satisfied using four constraints that every every structural element  $q_s$  retrieved from the repository and associated with a query structural element  $q_r$  should comply: (i)  $q_r$  and  $q_s$  must not be from the same protein complex; (ii) the central residues  $r$  and  $s$  of the structural elements  $q_r$  and  $q_s$  respectively, must be identical; (iii) the difference between the accessible surface areas of  $r$  and  $s$  should be  $\leq 5\%$  of the maximum accessible surface area of residues identical to  $r$ ; and (iv) the differences between the accessible surface areas of  $q_r$  and  $q_s$  must be  $\leq 15\%$  of the maximum estimated accessible

surface area of any structural element centered on a residue identical to  $r$ . These constraints were experimentally determined, as explained in the Appendix A.

#### 4.3.5 *PrISE* algorithm

The *PrISE* algorithm is summarized in Figure 4.1. First, a query protein structure is decomposed into a collection of structural elements. For each structural element in the query protein, *PrISE* retrieves a collection of similar structural elements (referred as samples) from the repository of structural elements. *PrISE* uses the collection of retrieved samples and information derived from their associated proteins to predict whether the central residue of each structural element is an interface residue. The information derived from the associated proteins can be incorporated into our proposed method using three different approaches (Equations 1-3) that result in three variants of the *PrISE* algorithm for predicting protein interface residues. The first method, *PrISE<sub>L</sub>*, uses similarity between structural elements (i.e. local structural similarity). The second method, *PrISE<sub>G</sub>*, utilizes a measure of similarity between protein surfaces (i.e. general structural similarity). The last method, *PrISE<sub>C</sub>*, combines local and general structural similarity. A detailed description of these approaches as well as the rationales behind them are provided next.

Let  $S$  be a repository of structural elements (where each element is indexed by the protein from which the structural element is derived and the surface residue that it represents). Let  $Q$  be a query protein. Let  $S(Q)$  be the collection of structural elements of  $Q$  (recall that there are as many structural elements in  $S(Q)$  as there are surface residues in  $Q$ ). To predict whether the central residue  $r(q)$  of a structural element  $q \in S(Q)$  is an interface residue, a collection  $S_q$  of structural elements that are most similar to  $q$  is retrieved from the repository  $S$  based on the distance between the histogram of atom nomenclatures  $q$  and that of each element in  $S$ <sup>2</sup>. In the event of a tie, the sample with the lowest difference in accessible surface area between its central residue and residue  $r(q)$  is chosen.

---

<sup>2</sup>Based on results of exploratory experiments, we found that 50, 200, and 500 similar structural elements are adequate (respectively) for performing prediction using *PrISE<sub>L</sub>*, *PrISE<sub>G</sub>*, and *PrISE<sub>C</sub>*. See Figures 4 to 6 and the corresponding discussion in the Appendix A for details.



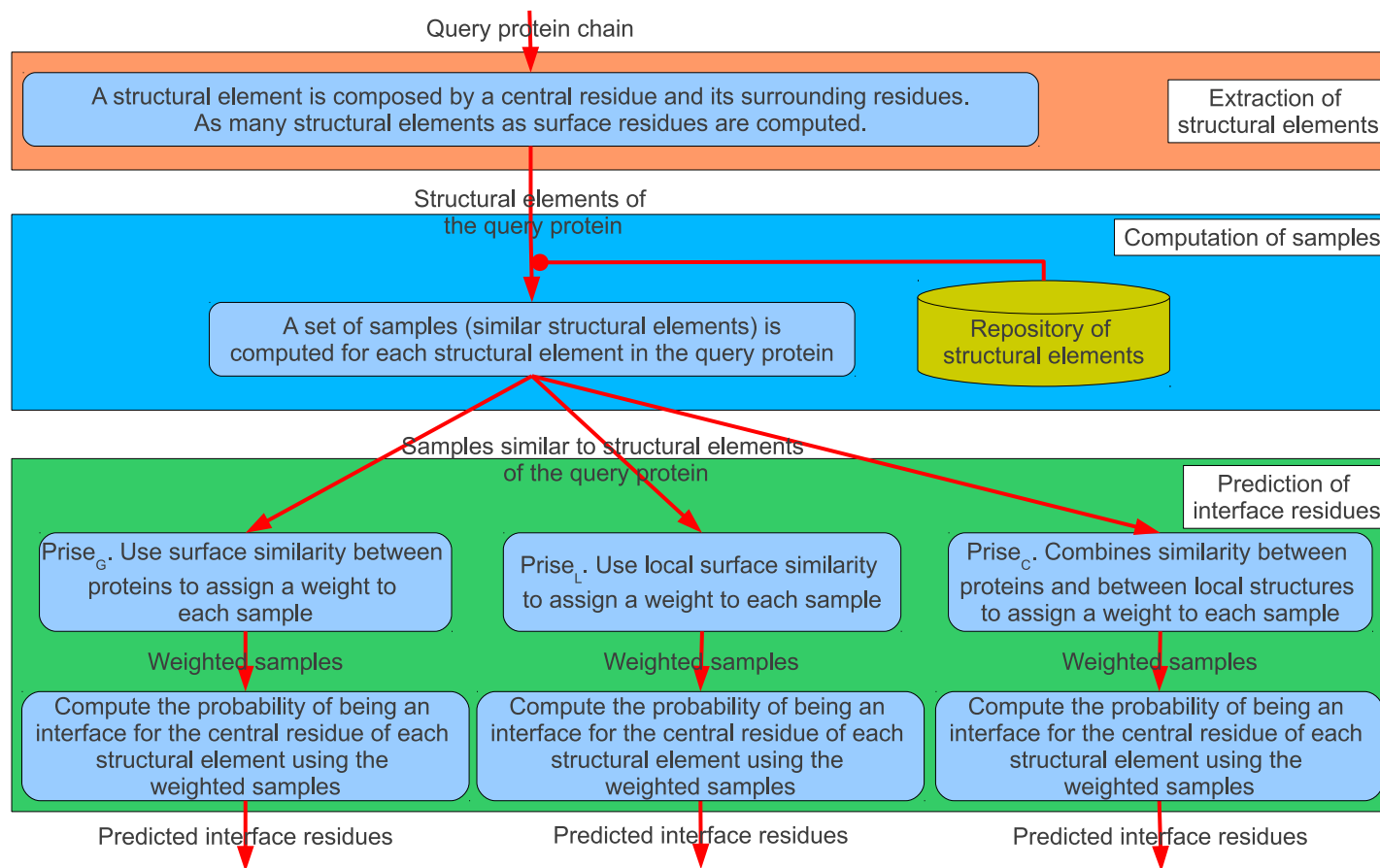


Figure 4.1 Prediction of interface residues using surface structural similarity.

For each structural element  $s$  in  $S$ , let  $\pi(s)$  denote the protein from which  $s$  was extracted. Given a protein  $P$  and an arbitrary collection  $R$  of structural elements, we define the *contribution*,  $cont(P, R)$ , as simply the number of structural elements in  $R$  that are associated with the protein  $P$ . For each  $q \in S(Q)$ , the collection of structural elements of protein  $Q$ , and for each structural element  $s \in S_q$ , we define the *weights*  $w_G(s, q)$ ,  $w_L(s, q)$  and  $w_C(s, q)$  (used by  $PrISE_G$ ,  $PrISE_L$ , and  $PrISE_C$  respectively) as follows:

$$w_G(s, q) = cont(\pi(s), Z_Q) \quad (4.1)$$

where  $Z_Q = \bigcup_{q \in S(Q)} S_q$ . Intuitively, the more similar the query protein  $Q$  containing the structural element  $q$  is to the protein from which the structural element  $s$  was derived, the greater the influence of  $s$  to the prediction on  $q$ .

Given a structural element  $q \in S(Q)$ , let  $Re(q)$  be the set of surface residues of  $Q$  that belong to  $q$ . Let  $N(q)$  be the set of structural elements associated with residues in  $Re(q)$ . Let  $N_q = \bigcup_{n \in N(q)} S_n$  (where  $S_n$ , the collection of structural elements that are most similar to  $n$ , is retrieved from the repository  $S$  of structural elements), we define the weight for  $PrISE_L$  as:

$$w_L(s, q) = cont(\pi(s), N_q) \quad (4.2)$$

Intuitively, the more similar the local surface patch of the structural element  $q$  is to a local surface patch of the protein from which the structural element  $s$  was derived, the greater the influence of  $s$  to the prediction on  $q$ .

For  $PrISE_C$ ,

$$w_C(s, q) = w_G(s, q) \times w_L(s, q) \quad (4.3)$$

Let  $S_+(q) = \{s \in S_q | r(s) \text{ is an interface residue}\}$  and  $S_-(q) = \{s \in S_q | r(s) \text{ is a non-interface residue}\}$ . Thus,  $PrISE_C$  combines the predictions of  $PrISE_L$  and  $PrISE_G$ . Because  $PrISE_L$  and  $PrISE_G$  weight each sample based on different criteria, this allows  $PrISE_C$  potentially to outperform each of them by taking advantage of complementary methods.

In the case of  $PrISE_G$ , the weight of positive samples associated with structural element  $q$  is defined as:

$$W_{G+}(q) = \sum_{s \in S_+(q)} w_G(s, q).$$

Similarly, the weight of negative samples associated with structural element  $q$  is defined as:

$$W_{G-}(q) = \sum_{s \in S_-(q)} w_G(s, q).$$

Finally, classification is performed by selecting a threshold<sup>3</sup> on the probability that indicates whether the central residue  $r(q)$  of the structural element  $q$  is likely to be an interface residue:

$$prob_{G+}(r(q)) = \frac{W_{G+}(q)}{W_{G+}(q) + W_{G-}(q)}$$

In the case of  $PrISE_L$ , and  $PrISE_C$ , the corresponding quantities  $W_{L+}(q)$ ,  $W_{L-}(q)$ , and  $prob_{L+}(r(q))$  and  $W_{C+}(q)$ ,  $W_{C-}(q)$ , and  $prob_{C+}(r(q))$  are defined in terms of the corresponding weights  $w_L$  and  $w_C$  (respectively).

### 4.3.6 Datasets

Four datasets were used to assess the performance of the  $PrISE$  family of interface predictors. The first dataset, DS24Carl (26), is composed of 24 chains: 16 extracted from transient complexes and eight extracted from complexes of different types. In this dataset, a residue is defined as an *interface residue* if the distance of the Van der Waals surface of any of its heavy atoms to a Van der Waals surface in any heavy atom of a different chain is  $\leq 3 \text{ \AA}$ . The other three datasets were defined in (212) from complexes used to evaluate protein docking software. DS188 is composed of 188 proteins chains derived from the Docking Benchmark 3.0 (85) sharing at most 40% sequence identity and containing 39,799 residues and 7,419 interacting residues. The other two datasets, DS56bound and DS56unbound, are composed by 56 protein chains derived from bound and unbound structures from the first 27 targets in CAPRI (88). DS56bound and DS56unbound have a total of 12,123 and 12,173 residues, and 2,154 and 2,112 interacting

---

<sup>3</sup>See the Appendix A for a discussion on the choice of the threshold.

residues respectively. For these three datasets, interface residues are defined as amino acids on two different protein chains with at least a pair of heavy atoms separated by at most 5 Å. These interfaces were computed from complexes extracted from PQS by the authors of (212).

#### 4.3.7 Performance Evaluation

The reliability of a prediction may be evaluated using different performance measures (10). We focused our evaluation on the following measures:

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

where  $TP$  refers to interface residues correctly predicted,  $FP$  to non-interface residues predicted as interfaces, and  $FN$  to interface residues predicted as non-interfaces. *Precision* evaluates the quality of the prediction in reference to the set of predicted interface residues, whereas *recall* measures the quality of the prediction with respect to the set of actual interface residues. When possible, the performance of different classifiers is evaluated by comparison of the precision-recall curve of each classifier. These curves are generated by computing precision and recall using different threshold values on the probability of each residue to be part of the interface. Therefore, these curves provide a more comprehensive evaluation than a pair of precision and a recall values.

For sake of completeness, we computed the following measures:

$$F1 = \frac{2 \times precision \times recall}{precision + recall}$$

$$Accuracy = \frac{TP + TN}{N}$$

$$CC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FN) \times (TP + FP) \times (TN + FP) \times (TN + FN)}}$$

The F1 score computes the harmonic mean between precision and recall. Accuracy measures how well interface and non-interface residues are correctly predicted. CC refers to the Matthews correlation coefficient. In addition, we use the area under the receiver operating characteristic (AUC ROC). This measure computes the area under the curve generated by computing the sensitivity and the false positive rate using different thresholds on the probabilities that indicates whether a residue belongs to the interface.

## 4.4 Results and discussion

We compared the *PrISE* family of algorithms using the DS188, DS24Carl, DS56bound and DS56unbound datasets. We also assessed the extent to which the quality of predictions is impacted by the presence of structural elements derived from homologs of the query protein in the repository of structural elements used to make the predictions. In addition, the performance of *PrISE<sub>C</sub>* was assessed against the performance of several classifiers based on machine learning methods, scoring functions, and local and global structural similarity on different datasets.

### 4.4.1 Comparison of *PrISE<sub>L</sub>*, *PrISE<sub>G</sub>* and *PrISE<sub>C</sub>*

Recall that *PrISE<sub>L</sub>* relies on the similarity between structural elements (i.e. local structural similarity), *PrISE<sub>G</sub>* relies on the similarity between protein surfaces (i.e. general structural similarity), and *PrISE<sub>C</sub>* combines local structural similarity and general structural similarity to predict interface residues. The performance of these three predictors were compared using the DS188 dataset. For this experiment, samples were extracted from the ProtInDb repository. In addition, samples extracted from proteins sharing more than 95% of sequence identity with the query protein and belonging to the same species were excluded from the prediction process to avoid overestimation on the predictions. To simulate a random prediction, the interface/non-interface labels associated with the central residue in each sample in the repository were randomly shuffled. The results of this experiment are presented in Figure 4.2 as precision-recall curves. These results indicate that *PrISE<sub>L</sub>*, *PrISE<sub>G</sub>*, and *PrISE<sub>C</sub>* outperform the random predictor. Furthermore, *PrISE<sub>C</sub>* achieves similar or better performance than *PrISE<sub>G</sub>* whereas *PrISE<sub>G</sub>* predictions are superior to those of *PrISE<sub>L</sub>*. Similar conclusions are supported by

experiments using the DS24Carl, DS56bound and DS56unbound datasets<sup>4</sup>. As a consequence,  $PrISE_C$  was selected to perform the experiments presented in the next subsections.

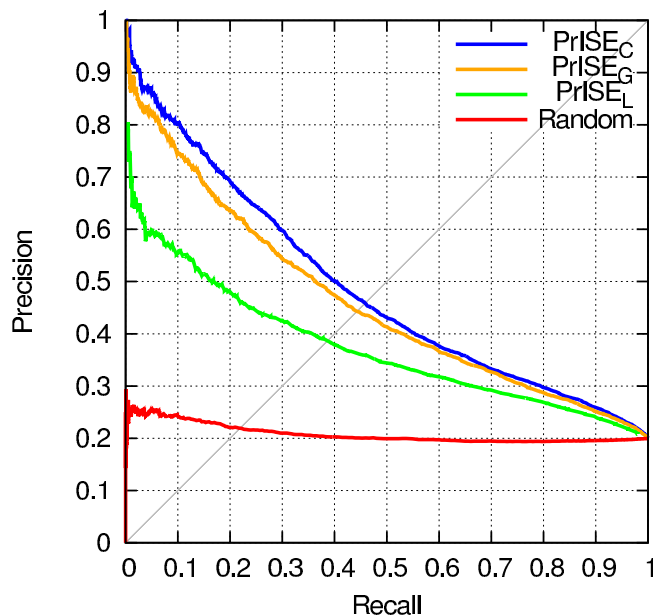


Figure 4.2 Comparative performances of  $PrISE_L$ ,  $PrISE_G$ ,  $PrISE_C$ , and randomly generated predictions on the DS188 dataset.

#### 4.4.2 Impact of homologs of the query protein on the quality of predictions

We assess the extent to which the predictions are impacted by the presence of structural elements derived from sequence homologs of the query protein. The first experiment excludes samples derived from proteins belonging to the same species that share  $\geq 95\%$  of sequence identity with the query protein (called *homologs from the same species*). The second experiment excludes samples from all the proteins that share  $\geq 95\%$  of sequence identity with the query protein (referred to as *homologs*).

Figure 4.3 compares the two methods for excluding homologs with a setup in which only the samples derived from proteins with the same PDB ID as the query proteins are excluded<sup>5</sup>.

<sup>4</sup>See section four of the Appendix A, that also includes an example of the relationship between the scores of the predictors in the  $PrISE$  family.

<sup>5</sup>Additional results using DS24Carl, DS56bound and DS56unbound are presented in section five of the Appendix A.

As seen from Figure 4.3, the prediction performance is better when sequence homologs of the query protein are not excluded from the set of proteins used to generate the repository used for making the predictions. The best performance is achieved by excluding the proteins with the same PDB ID as those of the query proteins.

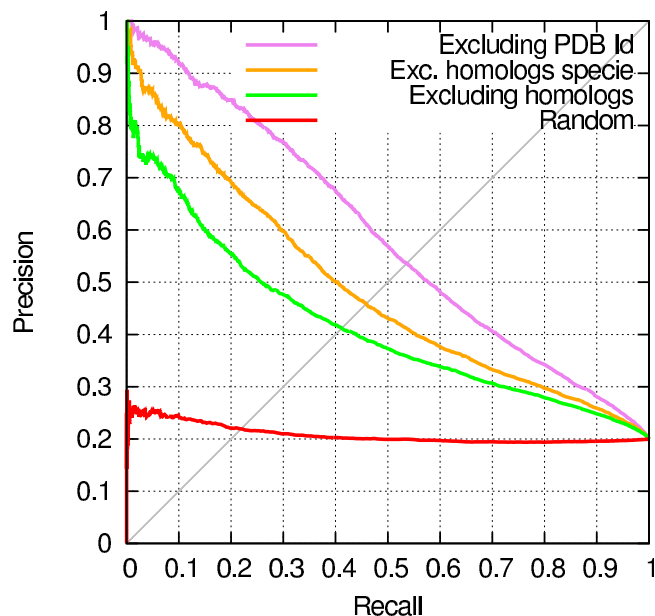


Figure 4.3 **Comparison of schemes for filtering out similar proteins from the prediction process.** This experiment was performed using *PrISE<sub>C</sub>* with the DS188 dataset.

#### 4.4.3 Comparison with two prediction methods based on geometric-conserved local surfaces

We compared the three predictors from the *PrISE* family with the predictors proposed by Carl et al. in (26; 27). These methods rely on conservation of the geometry and the physico-chemical properties of surface patches to predict interfaces. In (26), the conserved regions were extracted from proteins with similar structures. In (27), similar performance was achieved using conserved regions extracted using local structural alignments. This comparison was performed using the DS24Carl dataset composed of 24 proteins and generated in (27). In the case of *PrISE* family of methods, samples were retrieved from the ProtInDb repository.

Samples extracted from proteins sharing more than 95% of sequence identity with the query protein and belonging to the same species were not used in the prediction process. The results of the experiment, presented in Table 4.1, indicate that each of the three predictors from the *PrISE* family outperforms the predictors described in (26; 27). The differences in performances may be explained by the differences in the prediction techniques. In particular, *PrISE* family of predictors, unlike those of Carl et al., exploit the interface / non-interface labels associated with surface patches that share structural similarity with the surface neighborhood of each surface residue of the query protein.

Table 4.1 **Performance of different methods on the DS24Carl dataset.** Performance measures are computed as the average on the set of 24 proteins. Precision and recall values for Carl08 and Carl10 were taken from (26) and (27) respectively.

Predictor	Precision %	Recall %	F1 %	Accuracy %	CC %	AUC %
Carl08	31.5	35.3	33.3	-	-	-
Carl10	32.0	34.0	33.0	-	-	-
<i>PrISE<sub>L</sub></i>	45.1	56.2	50.0	69.1	27.1	70.5
<i>PrISE<sub>G</sub></i>	53.9	58.7	56.2	75.1	36.8	75.6
<i>PrISE<sub>C</sub></i>	58.3	58.3	58.3	77.5	40.6	77.1

Results of a similar experiment excluding samples extracted from homologs of the query proteins, as well as results of experiments using the  $ProtInDb \cap PQS$  repository, are presented in section six of the Appendix A.

#### 4.4.4 Comparison with a prediction method based on protein structural similarity

We compared *PrISE<sub>C</sub>* with *PredUs* (212; 211), a method that relies on protein structural similarity, using the DS188, DS56bound and DS56unbound datasets. *PredUs* is based on the idea that interaction sites are conserved among proteins that are structurally similar to each other. *PredUs* computes a structural alignment of the query protein with every protein in a set of proteins with known interface residues. The alignments are used to extract a *contact frequency map* which indicates for each residue in the query protein, the number of interface residues that are structurally aligned with it. The contact frequency map is then used to predict whether each residue on the query protein is an interface residue. In (212), the prediction was performed using a logistic regression function that receives as inputs the counts contained in



the contact frequency maps. In (211), the logistic regression function was replaced by a support vector machine (SVM) classifier that uses accessible surface areas and the counts contained in the contact frequency maps to perform prediction.

In order to perform a fair comparison between *PrISE* and *PredUs*, the structural elements used by *PrISE* and the structural neighbors used by *PredUs* were extracted from the same dataset of proteins. This dataset corresponds to the subset of proteins that are common to both ProtInDb and PQS which ensures the largest overlap between the proteins used by *PredUs* (which relies on the structural neighbors extracted from PDB and PQS) and *PrISE* (which relies on the proteins extracted from biological assemblies in PDB and deposited in ProtInDb). This resulting dataset, used to create the  $ProtInDB \cap PQS$  repository, includes 55,974 protein chains derived from 21,786 protein complexes. *PredUs* predictions were obtained from the available web server (211). This server allows us to choose the set of structural neighbors to be considered in the prediction process. Using this feature, we were able to exclude from the sets of structural neighbors those proteins that were not in the intersection of ProtInDb and PQS as well as homologs or homologs from the same species.

A first comparison of the *PrISE* family of predictors and *PredUs* was carried out using the DS188 dataset. However, since the SVM used by *PredUs* was trained using this dataset (211), it is likely that the estimated performance of *PredUs* in this case is overly optimistic, resulting in an unfair comparison with *PrISE*. We found that in 7 of 188 cases (corresponding to the PDB Ids and chains 1ghq-A, 1gp2-G, 1t6b-X, 1wq1-G, 1xd3-B, 1z0k-B, and 2ajf-A) *PredUs* failed to find structural neighbors, and hence failed to predict interfaces. In contrast, the *PrISE* predictors found the structural elements needed to produce predictions for the 188 cases. Predictions including these seven cases are labeled as  $PrISE_C$  188 in Figure 4.4, whereas predictions of  $PrISE_C$  and *PredUs* considering the set of 181 proteins are labeled with the suffix 181. The performances of  $PrISE_C$  in the two cases are similar. *PredUs* generally outperforms  $PrISE_C$ , the best performing predictor from the *PrISE* family. This result is not surprising given that the SVM used by *PredUs* was trained on this dataset whereas *PrISE* did not have this advantage.

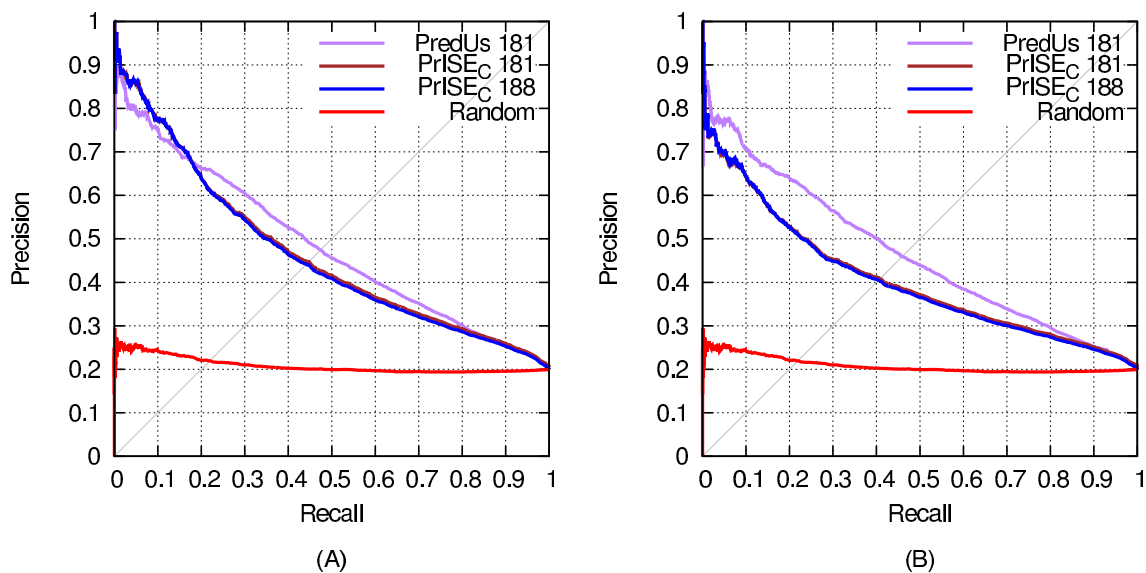


Figure 4.4 **Comparison of *PredUs* and *PrISE<sub>C</sub>* using the dataset DS188, derived from the docking benchmark 3.0.** (A) performance of predictions from which homologs from the same species were not used to compute the structural neighbors and the samples used in *PredUs* and *PrISE* respectively. (B) performance of predictions that did not consider homologs. Both images show results for the 181 proteins that were predicted by *PredUs* and *PrISE<sub>C</sub>* and for the 188 proteins predicted by *PrISE<sub>C</sub>*.

A second comparison of *PrISE<sub>C</sub>* and *PredUs* was performed using the DS56bound dataset. *PrISE<sub>C</sub>* and *PredUs* generated predictions for all the proteins in this dataset. The precision-recall curves presented in Figure 4.5 show that when homologs from the same species are excluded from the collection of similar structures, *PrISE<sub>C</sub>* outperforms *PredUs*, but when homologs are excluded regardless of the species, the performances of *PrISE<sub>C</sub>* and *PredUs* are comparable. These results indicate that the use of local surface structural similarity is a competitive alternative to the use of protein structural similarity for the problem of predicting protein-protein interface residues.

An evaluation considering additional performance measures is presented in Table 4.2. The data in this table indicates that *PrISE<sub>C</sub>* outperforms *PredUs* in terms of F1, correlation coefficient, or area under the ROC. The values for precision, recall, F1, Accuracy and CC were computed using the default cutoff values for *PrISE<sub>C</sub>* and *PredUs*.

Table 4.2 **Evaluation of  $PrISE_C$  and  $PredUs$  on DS56bound using different performance measures.** The table is divided into two sections depending on which proteins are excluded from the set of similar structures (First column).

Filter out	Predictor	Precision %	Recall %	F1 %	Accuracy %	CC %	AUC %
<b>Homologs from the same species</b>	$PredUs$	44.3	39.8	41.9	80.4	30.2	75.1
	$PrISE_C$	46.1	45.4	45.7	80.9	34.1	77.6
<b>Homologs</b>	$PredUs$	44.5	38.5	41.3	80.6	29.8	74.9
	$PrISE_C$	43.6	42.4	43.0	80.0	30.9	76.3

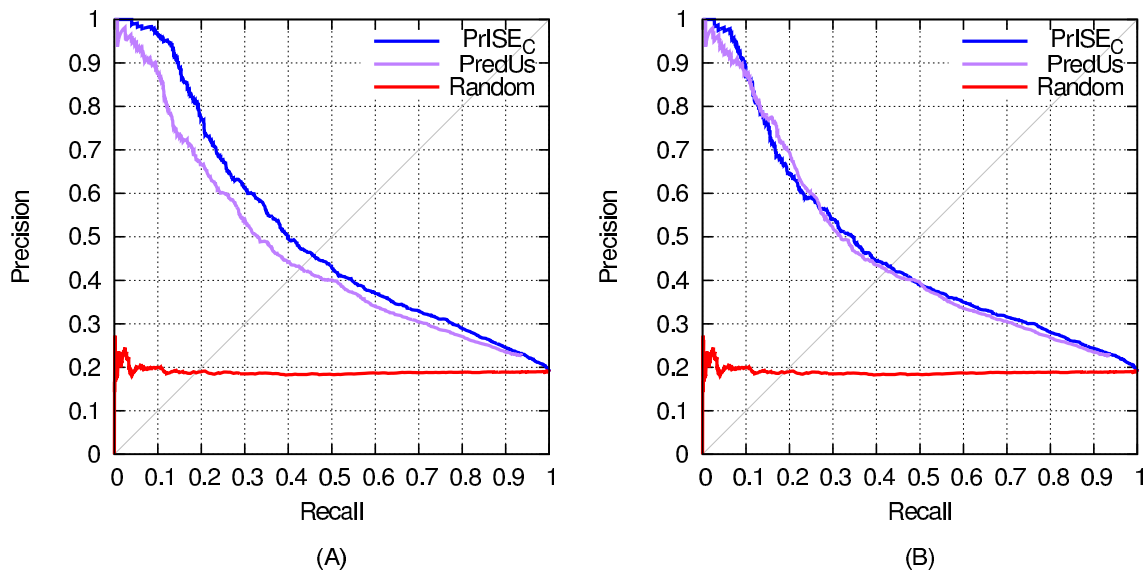


Figure 4.5 **Comparison of  $PrISE_C$  and  $PredUs$  using the dataset DS56bound, derived from CAPRI.** The results in (A) correspond to predictions in which homologs from the same species were excluded from the collection of samples and the set of structural neighbors. The results in (B) were obtained excluding homologs from the sets of similar structures.

A final comparison between  $PrISE_C$  and  $PredUs$  was performed using the DS56unbound dataset. Three out of the 56 proteins (corresponding to the PDB IDs-chains 1ken-H, 1ken-L, and 1ohz-B) were not processed by  $PredUs$  because no structural neighbors were found. Figure 4.6 shows the precision-recall curves of  $PrISE_C$  and  $PredUs$  on the 53 cases covered by  $PredUs$ , as well as the performance of  $PrISE_C$  when all the 56 proteins are considered. A comparison of both predictors using the set of 53 proteins and excluding homologs from the same species, indicates that  $PrISE_C$  outperforms  $PredUs$  for precision values  $> 0.4$ . On the other hand, when homologs are excluded, the performance of  $PredUs$  is better than the performance of  $PrISE_C$  for precision values  $\geq 0.3$ . Finally, the performance of  $PrISE_C$  computed on 56 proteins is, surprisingly, slightly better than the performance computed on 53 proteins. This suggests that idea that local structural similarity based interface prediction methods can be effective even in the absence of globally similar structures in the repository used for making the predictions.

An evaluation of  $PrISE_C$  and  $PredUs$  using additional performance measures is presented in Table 4.3.  $PrISE_C$  outperforms  $PredUs$  in terms of F1, CC and AUC when homologs from

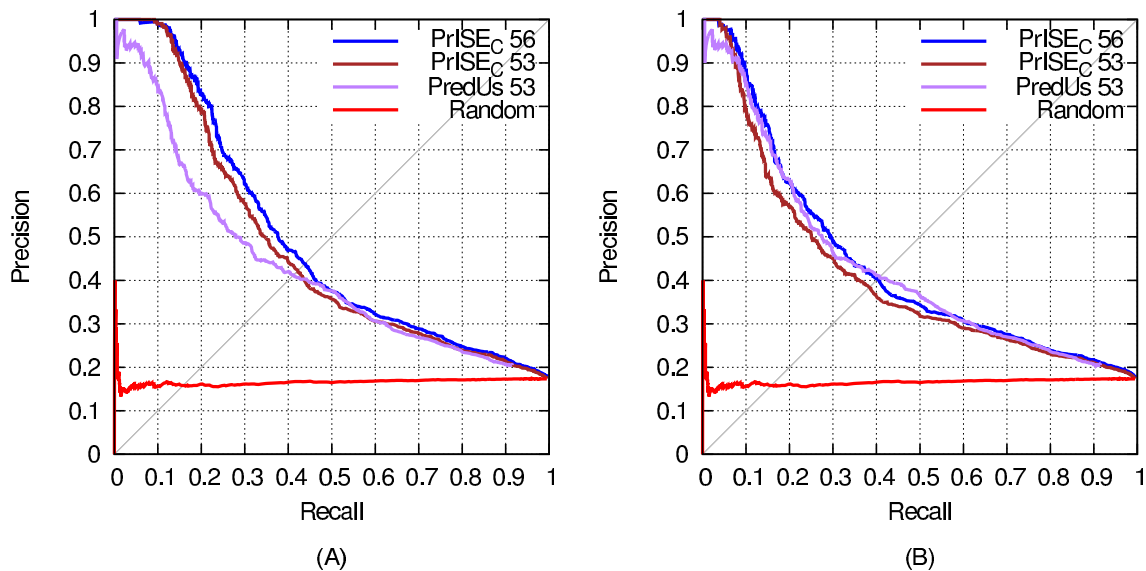


Figure 4.6 **Comparison of  $PrISE_C$  and  $PredUs$  using the DS56unbound dataset , derived from CAPRI.** (A) shows the performance achieved after removing homologs from the set of similar structures in  $PredUs$  and  $PrISE_C$ . (B) shows the performances when homologs are excluded. The suffixes 53 and 56 indicate the number of proteins that were used in the experiment.

the same species are excluded from the set of similar structures. When homologs are excluded,  $PredUs$  outperforms  $PrISE_C$  on the set of 53 proteins predicted by  $PredUs$ .

#### 4.4.5 Comparison with other prediction methods

We compared the performances of  $PrISE_C$ , Promate (139), PINUP(119), Cons-PPISP (31), and Meta-PPISP (158) using all the proteins in the DS56bound and DS56unbound datasets. The choice of the predictors used in this comparison was based on the results of a comparative study in which they were reported to achieve the best performance among the six different classifiers on two different datasets (213). Promate uses a scoring function based on features describing evolutionary conservation, chemical character of the atoms, secondary structures, distributions of atoms and amino acids, and distribution of b-factors. Cons-PPISP’s predictions are based on a consensus between different artificial neural networks trained on conservation sequence profiles and solvent accessibilities. PINUP uses an empirical scoring function based on side chain energy scores, interface propensity and residue conservation. Meta-PPISP uses linear regression on the scores produced by Cons-PPISP, Promate and PINUP.

Table 4.3 **Evaluation of *PrISE<sub>C</sub>* and *PredUs* on DS56unbound using different performance measures.**

Filter out	Predictor	Precision %	Recall %	F1 %	Accuracy %	CC %	AUC %
<b>Homologs from the same species</b>	<i>PredUs</i> 53	43.2	37.2	39.9	81.8	29.4	73.6
	<i>PrISE<sub>C</sub></i> 53	42.3	42.1	42.2	81.2	31.0	74.8
	<i>PrISE<sub>C</sub></i> 56	43.7	44.0	43.8	81.2	32.6	75.5
<b>Homologs</b>	<i>PredUs</i> 53	42.6	36.8	39.5	81.6	28.8	73.5
	<i>PrISE<sub>C</sub></i> 53	38.8	37.9	38.4	80.1	26.5	72.9
	<i>PrISE<sub>C</sub></i> 56	40.5	40.0	40.2	80.2	28.4	73.7

In the experiments presented in this subsection, we considered the performance of two  $PrISE_C$  classifiers according to which proteins were filtered out from the process of extraction of samples: homologs from the same species as the query protein and homologs regardless of the species. The scores used to generate the precision-recall curves of Promate, PINUP, Cons-PPISP and Meta-PPISP were computed using Meta-PPISP’s web server.

The precision-recall curves corresponding to the evaluation of the classifiers on the DS56bound and DS56Unbound datasets are shown in Figure 4.7. On both the datasets,  $PrISE_C$  predictors outperform Meta-PPISP for precision values  $> 0.35$  and achieve performance comparable to that of Meta-PPISP for precision values  $\leq 0.35$ . Furthermore,  $PrISE_C$  outperform Promate, PINUP, and Cons-PPISP over the entire range of precision and recall values.

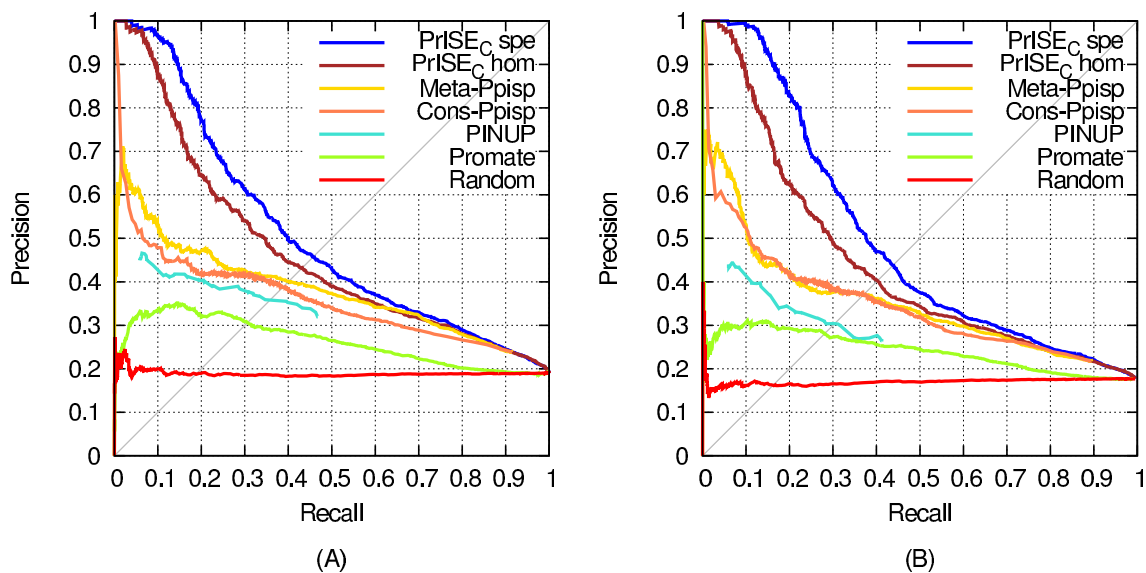


Figure 4.7 **Performance of different classifiers evaluated on the DS56bound (A) and the DS56unbound (B) datasets.** For the  $PrISE$  classifiers, “spe.” and “hom.” show predictions in which samples extracted from homologs from the same specie and homologs, respectively, has been excluded from the prediction process.

An evaluation considering additional performance measures is presented in Table 4.4. All the performance measures, with exception of AUC ROC, were computed using threshold values of 0.56, 0.28, 0.41, 0.34, and 0.34 on the scores generated by Promate, PINUP, Cons-PPISP, Meta-PPISP, and  $PrISE_C$  respectively. These threshold values correspond to the default values

defined in the Meta-PPISP and  $PrISE_C$  web servers. The results show that the  $PrISE_C$  predictors outperform the other predictors on both datasets in terms of F1, correlation coefficient and area under the ROC.

The results of a experiment using 187 proteins from the DS188 dataset is presented in Figure 4.8. Protein chain 2vis-C was excluded from the experiment given that Promate could not generate a prediction. When homologs from the same species are excluded,  $PrISE_C$  outperforms the other predictors except Meta-PPISP.  $PrISE_C$  outperforms Meta-PPISP for precision values  $> 0.4$  and achieves comparable performance to that of Meta-PPISP for precision values  $\leq 0.4$ . When homologs are excluded, the performance of  $PrISE_C$  is superior that the performance of PINUP and Promate.  $PrISE_C$  outperforms Meta-PPISP and Cons-PPISP for precision values  $> 0.5$ , and is outperformed by Meta-PPISP for precision values  $\leq 0.45$ .

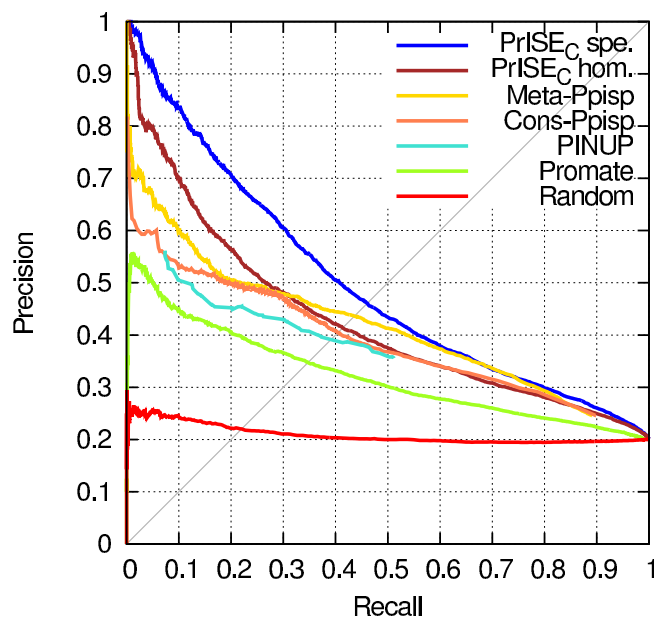


Figure 4.8 **Precision-recall curves of different classifiers evaluated on 187 proteins from the DS188 dataset.** For the  $PrISE$  classifiers, “spe.” and “hom.” show predictions in which homologs from the same species and homologs, respectively, has been excluded from the repository of structural elements.

An evaluation using different performance measures is presented in Table 4.5. According to this table, the performance of both  $PrISE$  predictors is superior that the performance of the



Table 4.4 **Evaluation on the datasets DS56bound and DS56unbound.** “*PrISE<sub>C</sub>* spe.” refers to the performance computed after filtering out from the repository samples extracted from homologs from the same species. “*PrISE<sub>C</sub>* hom.” indicates that samples extracted from homologs were not considered in the prediction process.

Dataset	Predictor	Precision %	Recall %	F1 %	Accuracy %	CC %	AUC %
<b>DS56bound</b>	Promate	31.9	27.3	29.4	76.7	15.6	63.3
	PINUP	37.3	31.9	34.4	78.4	21.7	63.7
	Cons-PPISP	39.8	36.1	37.9	78.9	25.2	72.6
	Meta-PPISP	43.3	25.8	32.3	80.8	22.9	74.4
	<i>PrISE<sub>C</sub></i> spe.	46.1	45.4	45.7	80.9	34.1	77.6
	<i>PrISE<sub>C</sub></i> hom.	43.6	42.4	43.0	80.0	30.9	76.3
<b>Ds56unbound</b>	Promate	28.7	27.3	28.0	76.6	14.0	62.7
	PINUP	30.4	30.1	30.2	76.9	16.4	60.0
	Cons-PPISP	37.4	34.5	35.9	79.5	23.8	71.2
	Meta-PPISP	38.9	24.0	29.7	81.1	20.2	71.5
	<i>PrISE<sub>C</sub></i> spe.	43.7	44.0	43.8	81.2	32.6	75.5
	<i>PrISE<sub>C</sub></i> hom.	40.5	40.0	40.2	80.2	28.4	73.7

other classifiers in terms of F1 and CC. Furthermore, when homologs from the same species are excluded, *PrISE<sub>C</sub>* outperforms the other classifiers in terms of AUC.

Table 4.5 **Evaluation on 187 proteins from DS188.** “*PrISE<sub>C</sub>* spe.” refers to the performance computed after excluding from the prediction process samples extracted from homologs of the same species that the query proteins. “*PrISE<sub>C</sub>* hom.” indicates that samples extracted from homologs were filtered out from the repository.

Predictor	Precision %	Recall %	F1 %	Accuracy %	CC %	AUC %
Promate	36.5	30.3	33.1	77.1	19.5	67.7
PINUP	40.7	34.7	37.5	78.3	24.6	66.0
Cons-PPISP	46.5	30.6	36.9	80.4	26.7	73.2
Meta-PPISP	49.0	26.7	34.6	81.1	26.2	74.6
<i>PrISE<sub>C</sub></i> spe.	48.0	43.2	45.5	80.6	33.8	77.2
<i>PrISE<sub>C</sub></i> hom.	43.2	38.1	40.5	79.0	27.9	74.2

#### 4.4.6 Prediction performances in the absence of similar proteins

To evaluate the extent to which the performances of *PrISE<sub>C</sub>* and *PredUs* depend on the degree of homology between the query proteins and the proteins used to extract samples or structural neighbors, we compare the results obtained using three different sequence homology cutoffs: 95%, 50% and 30%. The results, shown in Figure 4.9, indicate that *PredUs* is more sensitive than *PrISE<sub>C</sub>* to the lack of similar proteins in the sets used to extract similar structures. The figure also shows that the performance of *PrISE<sub>C</sub>* is competitive with that of Meta-PPISP even when the repository used by *PrISE<sub>C</sub>* is composed by proteins sharing < 30% of sequence identity with the query proteins.

## 4.5 Conclusions

We have shown that it is possible to reliably predict protein-protein interface residues using only local surface structural similarity with proteins with known interfaces.

The experiments comparing the performance of the *PriSE* family of predictors with the structural similarity based interface predictors of Carl et al. (26; 27) show that the use of interface / non interface labels of residues in structurally similar surface patches leads to improved predictions by *PrISE*. This observation is also supported by the results obtained using *Pre-*

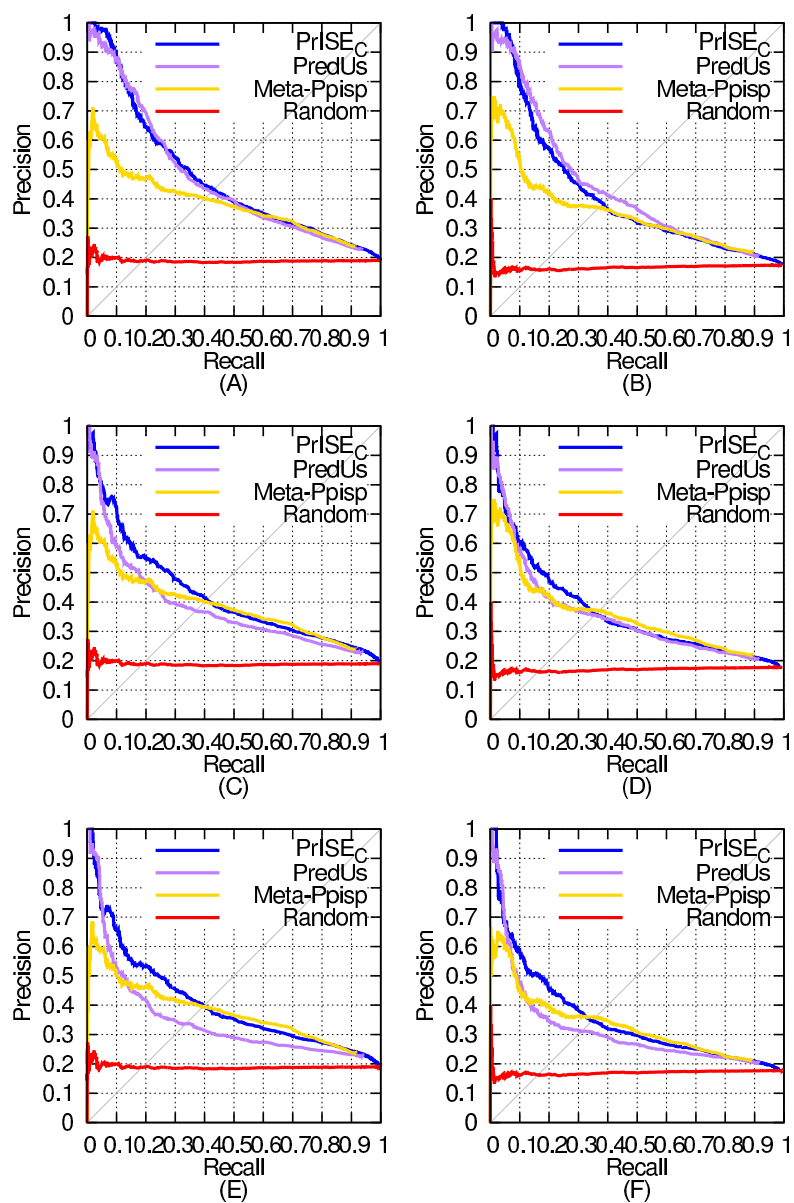


Figure 4.9 **Performance computed in absence of similar proteins at different similarity levels.** Figures (A) and (B) show the precision recall curves computed after excluding from the sets of similar structures homologs (without regarding the species) sharing  $\geq 95\%$  of sequence identity with the query proteins. Similarly, figures (C) and (D) show the performances after excluding proteins sharing  $\geq 50\%$  sequence identity, and (E) and (F) display the results after filtering out proteins with sequence identity  $\geq 30\%$ . The precision-recall curves corresponding to the DS56bound dataset are shown at (A), (C), and (E), and the results computed using the DS56Unbound dataset are labeled as (B), (D), and (F). Figures (E) and (F) were computed using 55 and 52 proteins respectively given that PredUs could not find structural elements for the protein chain 1ynt-L.

*dUs*, that implicitly exploits information about non-interface residues reflected in the contacting frequencies of interface residues.

Surface structural similarity based methods for interface residue prediction may use local similarity, overall similarity, or a combination of both. *PrISE<sub>L</sub>*, which relies on the similarity between structural elements (i.e. local structural similarity) outperforms random prediction; *PrISE<sub>G</sub>* which relies on the similarity between protein surfaces (i.e. general structural similarity) outperforms *PrISE<sub>L</sub>*. This result may not be surprising in light of the influence that regions outside the immediate local environment have on the conformation of protein complexes. However, our results show that the best predictions are achieved by *PrISE<sub>C</sub>*, using a combination of local and overall surface similarity.

Our results indicate that, in general, *PrISE<sub>C</sub>* outperforms several state of the art predictors such as Promate, PINUP, Cons-PPISP, and Meta-PPISP. Blind comparisons of *PrISE<sub>C</sub>* and *PredUs* using the same proteins to extract samples and structural neighbors respectively, indicate that *PrISE<sub>C</sub>* achieves performance that is superior to or comparable with that of *PredUs*. Furthermore, *PrISE<sub>C</sub>* is more robust than *PredUs* at low levels of homology between the query proteins and proteins in the sets used to extract similar structures, while remains competitive with Meta-PPISP.

The interface residue prediction methods such as *PrISE* that use only local surface structural similarity have an advantage relative to methods that rely on global structural similarity: The former can produce predictions whereas the latter cannot in the absence of protein with structures that are sufficiently similar to the structure of the query protein.

Another advantage of the *PrISE* family of predictors is that the information needed to compute similar structural elements (i.e. residues in the structural elements, accessible surface area of these residues and their histogram of atom nomenclatures) can be obtained in a reasonable amount of time. The time required for retrieving the samples associated with a query protein from a repository of 21,289,060 structural elements extracted from 88,593 protein chains is in average 90 seconds using a personal computer (Intel Core2 Duo CPU at 2.40GHz, 4MB of RAM and a hard disk of 232 GB).

We conclude that methods based on local surface structural similarity are a simple yet

effective approach to the problem of prediction of protein-protein interface residues.

#### **4.6 Competing interests.**

The authors declare that they have no competing interests.

#### **4.7 Author's contributions.**

The study was originally conceived by VH and RAJ. RAJ carried out the experiments. All the authors discussed the experimental design, and participated in the analysis and interpretation of the data. RAJ wrote the initial draft of the manuscript. All authors revised and approved the final manuscript.

#### **4.8 Acknowledgements.**

This work was funded in part by the National Institutes of Health grant GM066387 to Vasant Honavar and Drena Dobbs and in part by a research assistantship funded by the Center for Computational Intelligence, Learning, and Discovery. The authors thank Li Xue, Rasna Walia, and Fadi Towfic for useful discussions and suggestions. The work of Vasant Honavar while working at the National Science Foundation was supported by the National Science Foundation. Any opinion, finding, and conclusions contained in this article are those of the authors and do not necessarily reflect the views of the National Science Foundation.

## CHAPTER 5. CONCLUSIONS

The work presented in this thesis focuses on the development of tools and methods for improving the prediction of protein-protein interaction sites. Advancements in prediction of protein-protein interaction sites will lead to advances in problems such as prediction and validation of protein function, prediction of protein quaternary structures (i.e. protein docking), prediction and validation of protein-protein interactions and protein-protein interaction networks, identification of hot-spot residues, prediction of epitopes, and drug design.

We introduced *ProtInDb*, a database of protein-protein interface residues that allows users to visualize the interaction sites in protein structures deposited in the PDB, and that assists users in the creation of representative datasets that simplify the processes used for training, testing, and comparing predictors of interface residues. The format of the data in these datasets allows users to efficiently store and extract the fundamental information required to identify interface residues as well as data about the solvent accessibility and the structural neighborhood of each amino acid residue of proteins of interest. *ProtInDb* also allows users to download a copy of the basic information of all the interacting proteins in the PDB, which can be used to perform comprehensive studies involving interactions between proteins. Such information includes the protein sequence (derived from structural data), mappings between the position of each residue in the sequence and in the structure, and flags indicating whether each residue in a protein is or is not an interface and/or surface residue. *ProtInDb* supports three definitions of interface residues, and allows users to define threshold values that determine whether a residue is or is not an interface residue and whether a residue is or is not on the surface of a protein subunit. *ProtInDb* also allows users to select which type of structure should be used to determine the interaction sites: Asymmetric units, derived directly from experiments performed to determine the protein structures, or biological assemblies, representing the structure that has

been shown or is believed to be biologically functional. *ProtInDb* has been used to construct representative datasets utilized to train and test diverse predictors of interaction sites, and to build a benchmarking dataset of bound-unbound conformational B-cell epitopes. *ProtInDb* also provides the data required by several predictors of interface residues based on similarity between proteins (i.e. *PS-HomPPI* and *NPS-HomPPI* (202), and *PrISE* (98)) that benefit from using the largest amount of information involving interaction sites. *ProtInDb* is accessible at <http://protindb.cs.iastate.edu>.

We proposed *PoInterS*, a method to predict interface patches based on the outcomes produced by predictors of interface residues. Prediction of interface patches allows users to focus their experiments into specific sites on the surface of the protein, which can generate significant savings in time and resources. *PoInterS* decomposes the surface of a protein into a series of patches, ranks them using a scoring function based on the probabilities or the interface/non-interface labels assigned to every surface residue by predictors of interface residues, and returns the three patches that are the most likely to belong to the interaction sites of the given protein. Based on the *PoInterS* method we implemented *PoInterS-SVM*, a predictor of protein-protein interface patches that uses the results generated by a support vector machine predictor of interface residues. Our results indicate that *PoInterS-SVM* outperforms *SHARP*<sup>2</sup> and *PPI-Pred*, two state-of-the-art predictors of interface patches. The modular nature of the method, based on the idea that the outcomes generated by any predictor of interface residues can be used to predict interaction patches, and the experimental results supporting the success of the method in predicting interaction sites, indicate that the creation of improved predictors of interface residues will result in more successful predictors of interface patches. *PoInterS-SVM* has been implemented as a Web application available at <http://pointers.cs.iastate.edu>

We introduced *PrISE* (98), a method for predicting protein-protein interface residues based on the similarity of small protein regions called structural elements. A structural element is composed of a central residue and its closest residues in a protein structure, and is represented using data extracted from the atomic composition and the area accessible to the solvent of its constituent residues. This representation allowed us to create an efficient method to search and retrieve from a large database of structural elements a set of similar elements to those of

a query protein. Each similar structural element is weighted using the idea of contribution of a protein to a set of structural elements, that counts the number of structural elements in the protein that are similar to structural elements in the set. The set of weighted structural elements are used to compute a final score that indicates whether the central residue of every structural element in the query protein is or is not an interface residue. We created predictors of interface residues based on the similarity between proteins ( $PrISE_G$ ), the similarity between protein regions ( $PrISE_L$ ) or a combination of both ( $PrISE_C$ ). Our results indicate that  $PrISE_C$  outperforms  $PrISE_G$  and that  $PrISE_G$  outperforms  $PrISE_L$ . Comparisons using several datasets show that  $PrISE_C$  outperforms a method based on the similarity between protein regions, and achieves a performance that is superior or comparable to that of a state-of-the-art predictor based on the similarity between protein structures (PredUs), and a meta-predictor (meta-PPISP) of protein-protein interface residues that was selected given its high performance in several experiments presented in the literature.  $PrISE_C$  is accessible via Web server at the URL <http://prise.cs.iastate.edu>

The results of this research work can facilitate the development of experiments based on or related to protein-protein interface residues. Biochemists and molecular biologists can use the predictions generated by  $PoInterS$  and  $PrISE$  as a guide to performing in vitro or in vivo experiments oriented to find hot spot residues, to gain a better understanding of the mechanisms involved in protein-protein interactions, and to develop new therapeutic drugs. Bioinformaticians can benefit from the tools provided by  $ProtInDb$  to visualize protein-protein interface residues and to create representative datasets of interface residues. The ideas behind  $PoInterS$  and  $PrISE$ , as well as their predictions, can also be applied to problems such as prediction of interactions between diverse macromolecules, prediction of protein function, prediction and validation of interactions between proteins, molecular docking, and in-silico design of new drugs.



## 5.1 Future work

### 5.1.1 Extending *ProtInDb*

Two changes would allow *ProtInDb* to be useful to a larger number of scientists. *ProtInDb* could be extended to include information of interaction sites between proteins and DNA, RNA and small ligands. This would extend the applicability of *ProtInDb* to different problems concerning interaction between different macromolecules. In addition, including structural information of non-interacting proteins (i.e. protein complexes composed of only one subunit) in *ProtInDb* would allow the generation of datasets of unbound proteins. This could facilitate the development of studies such as the evaluation of conformational changes in molecular structures after formation of complexes, and to carry out more comprehensive evaluations (e.g. assessing the performance of predictors of interaction sites on bound and unbound proteins).

### 5.1.2 Prediction of interface residues between different macro-molecules

Although *PrISE* and *PoInterS* methods were developed to predict protein-protein interaction sites, they could be extended for predicting protein-RNA (157), protein-DNA (45) and protein-small ligands (112) interaction sites.

### 5.1.3 Using *PrISE* and *PoInterS* to assist biological experiments

Though the performance evaluations of *PrISE* and *PoInterS* indicate their effectivity in predicting interaction sites, it would be interesting to use them to assist scientist in in-vitro or in-vivo experiments (e.g. in tasks such as selection of target residues for alanine-scanning mutagenesis experiments oriented to detect hot-spot residues, or the prediction or validation of interactions in protein-protein interaction networks). In addition to the potential benefits of using the predictions generated by our methods, this would allow us to gain a better understanding of their advantages and limitations, which will result in improvements in the reliability of their predictions, and to explore alternative applications of our methods.

#### 5.1.4 Creation of more sophisticated methods to retrieve similar structural elements in PrISE

The method used by *PrISE* to retrieve similar structural elements is based on a measure of the differences in the atomic composition and the accessible surface areas of the query structural element and another element in the repository of structural elements. Despite this method proved to be effective and efficient, it ignores some physico-chemical properties and relationships between the atoms or residues in a structural element that could contribute to retrieve a most suitable set of similar structural elements. For example, the selection of a subset of the atoms included the histogram of atom nomenclatures or the use of weights associated with each atom in the histogram (e.g. according to the relative accessible surface area or the average charge of the atom and its neighbors), or the consideration of topological relationship between functional groups of residues (167; 107), could lead into a new representation of structural elements that produces a more accurate prediction of interaction sites.

#### 5.1.5 Partner-specific versions of *PrISE* and *PoInterS*

*PrISE* and *PoInterS* are non-partner specific prediction methods, in the sense that they predict interface residues and interaction sites for a query protein without considering any information of its specific interacting protein partner(s). However, applications such as protein docking, prediction and validation of protein-protein interactions, and development of drugs that disrupt interactions between particular proteins will benefit of partner-specific prediction methods, that focus on predicting the interaction sites between two or more specific proteins. Diverse partner-specific methods has been devised in the literature, including a method that computes the interaction sites from sets of homo-interologs (i.e. complexes containing interacting proteins that are similar to the query proteins) (202), and other that use machine learning techniques to infer a set of pairs of interaction sites in the partner proteins that are most likely to interact (36; 195; 2). Similar approaches can be developed for *PrISE* and extended to *PoInterS* using a database of interacting structural elements that can be extracted from *ProtInDb*.

### 5.1.6 Searching for proteins with similar structure

Retrieving proteins with similar structures from the PDB is a computational intensive task (101) with applications in problems such as protein design, analysis and prediction of protein functions, prediction of protein structures, and drug discovery. The concept of local structural similarity devised for *PrISE* could be used to create a method to efficiently retrieve from the PDB proteins with similar structure or substructures to that of a query protein. This hypothesis is supported by the existence of methods that retrieve structural neighbors (38; 23) or similar substructures (198; 18) based on protein segments, and by the similarity in the performances of *PrISE<sub>C</sub>* with *PredUs*, based on local structural similarity and protein structural similarity respectively. Such method could serve (i) to discover a reduced number of proteins with similar structures to that of a query protein; (ii) as a filter to decrease the number of pairwise structural alignments required to find proteins with similar structures by excluding proteins with low similarity; or (iii) to retrieve a set of similar substructures for a query substructure. An efficient and reliable method to perform search of similar protein structures will produce a significant impact on the problems mentioned above.

## APPENDIX A. SUPPLEMENTARY MATERIAL FOR CHAPTER 4

This document provides additional information about the process used for building the *PrISE* family of predictors of protein-protein interface residues as well as supplementary results of some of the experiments described in chapter 4. The first two sections describe details about the histograms of atom nomenclatures and the constraints used to retrieve similar structural elements from a repository of structural elements. The next section describe the dataset and the experiments used for tuning the parameters of *PrISE<sub>G</sub>*, *PrISE<sub>L</sub>*, and *PrISE<sub>C</sub>*. The remaining sections show the results of complementary experiments to the reported in chapter 4 performed on different datasets.

### A.1 Atom nomenclatures

A list of the 36 atom nomenclatures used to build the histograms of atom nomenclatures (HAN) is presented in Table A.1. These nomenclatures were extracted from PDB.

Table A.1 **Atom nomenclatures used to build the histograms of atom nomenclatures.**

C	CA	CB	CD	CD1	CD2
CE	CE1	CE2	CE3	CG	CG1
CG2	CH2	CZ	CZ2	CZ3	N
ND1	ND2	NE	NE1	NE2	NH1
NH2	NZ	O	OD1	OD2	OE1
OE2	OG	OG1	OH	SD	SG

### A.2 Retrieving similar structural elements - additional details

As explained in the methods section of chapter 4, we defined four constraints that every structural element retrieved from a repository should comply to be considered similar to a query

structural element:

“(i)  $q_r$  and  $q_s$  must not be from the same protein complex; (ii) the central residues  $r$  and  $s$  of the structural elements  $q_r$  and  $q_s$  respectively, must be identical; (iii) the difference between the accessible surface areas of  $r$  and  $s$  should be  $\leq 5\%$  of the maximum accessible surface area of residues identical to  $r$ ; and (iv) the differences between the accessible surface areas of  $q_r$  and  $q_s$  must be  $\leq 15\%$  of the maximum estimated accessible surface area of any structural element centered on a residue identical to  $r$ ”.

Constraint (iii) requires the computation of the difference between the accessible surface area of the central residues  $r$  and  $s$  of two structural elements  $q_r$  and  $q_s$  respectively. This difference, denoted by  $dASAs$ , is computed as:

$$dASAs(r, s) = \frac{|asaRes(r) - asaRes(s)|}{maxAsaRes(r) - minAsaRes(r)} \times 100\%$$

where  $asares(r_1)$  denotes the accessible surface area of the residue  $r_1$ , and  $minAsaRes(r_1)$  and  $maxAsaRes(r_1)$  denotes the experimental minimum and maximum accessible surface area of the residue  $r_1$  respectively<sup>1</sup>. The values of  $maxAsaRes$  and  $minAsaRes$  were estimated from a dataset of 400 proteins randomly selected from ProtInDb<sup>2</sup>, a database of protein-protein interface residues. the lower the values of  $dASAs$ , the highest the similarity between the accessible surface areas of the residues  $r$  and  $s$ .

Constraint (iv) requires the computation of the difference between the accessible surface areas of two structural elements  $q_1$  and  $q_2$ . This difference, denoted by  $dASAs_e$ , is computed as:

$$dASAs_e(q_1, q_2) = \frac{|asaSe(q_1) - asaSe(q_2)|}{maxAsaSe(q_1) - minAsaSe(q_1)} \times 100\%$$

where  $asaSe(q)$  denotes the summation of the accessible surface area of the surface atoms in the structural element  $q$ . An atom is considered to be a surface atom if its accessible surface area is  $> 0 \text{ \AA}^2$ .  $MinAsaSe(q)$  and  $maxAsaSe(q)$  represent the estimated minimum and maximum

<sup>1</sup>Note that according to constraint (ii) residue  $r$  is identical to residue  $s$ .

<sup>2</sup><http://protindb.cs.iastate.edu>

accessible surface areas of structural elements centered on a residue identical to the central residue of  $q$ . These two values were estimated from the dataset of 400 proteins extracted from ProtInDb. The interpretation of  $dASAse$  is similar to the interpretation of  $dASAses$  (i.e. the lowest the value of  $dASAse(q_1, q_2)$ , the highest the similarity between the accessible surface areas of the structural elements  $q_1$  and  $q_2$ ).

### A.3 Tuning method

We tuned the parameters of the *PrISE* family of predictors in two steps. The goal of the first step was to efficiently retrieve structural elements from the repository of structural elements for all the structural elements in a query protein. The goal of the second step was to maximize the prediction performance. We use the ProtInDb repository of structural elements to perform these experiments.

#### A.3.1 Tuning dataset

The tuning dataset is composed of 50 chains (see Table A.2) with more than 40 residues, resolution  $\leq 2.5$  Å, and sequence identity  $\leq 15\%$ . This dataset has 10,379 residues from which 1,946 are interface residues.

Table A.2 **List of the 50 protein chains included in the tuning dataset.**

1df4A	1risA	2dkoB	2qeeA	3fedA
1dqzA	1s72H	2dw5A	2vn6A	3h7hB
1dysA	1smxA	2hdiB	2vtbA	3hf5A
1euvA	1t0bA	2iihA	2ww2A	3hm4A
1i2cA	1u5kA	2izzA	2xdpA	3k94A
1j34C	1u9dA	2jkhL	2zewA	3kb4A
1kqfC	1uuyA	2o2vA	3ag3I	3kz5B
1kyfA	1v05A	2o70A	3bm3A	3m9lA
1pytA	1yrkA	2odeA	3ct6A	3mcwA
1q7lB	2cchB	2pmuA	3d32B	3pg6A

### A.3.2 Representative set of similar structural elements

We wanted to efficiently obtain a set of similar structural elements (samples) from the repository that allows us to perform predictions for all the structural elements in a query protein. To achieve this goal, we performed a grid search using values equivalent to 5%, 10% and 15% on the parameters  $dASAses$  and  $dASAsel$ . We found that using  $dASAses \leq 5\%$  and  $dASAsel \leq 15\%$  we can retrieve samples for all the structural elements in the dataset. The threshold of 5% on  $dASAses$  allows us to obtain samples whose central residues are as similar as possible to the central residue of a query structural element. The threshold of 15% on  $dASAsel$  allows us to include some flexibility to account for conformational changes in residues in the fringe of the structural elements whereas minimizes the potential problem of lack of samples for query proteins not included in the tuning dataset.

### A.3.3 Performance tuning

We analyzed the impact of different factors in the performance of the *PrISE* family of predictors that extracted samples with  $dASAses \leq 5\%$ , and  $dASAsel \leq 15\%$ . We evaluated several metrics of distance between histogram of atom nomenclatures as well as several schemes used to assign weights to the samples and to find the number of samples that maximized the performance of the predictions.

#### A.3.3.1 Evaluation of distance metrics for histogram of atom nomenclatures

We evaluated six different metrics of distance between histograms selected from a survey presented in (28): Inner product, fidelity, Euclidean distance, city block distance, symmetric Kullback–Leibler divergence, and symmetric Kullback–Leibler divergence with Laplace estimates<sup>3</sup>. We predicted a residue as an interface if the majority of the central residues of the top 50 samples (according to each metric) are interface residues. The results of these experiments, presented as precision-recall curves in Figure A.1, indicate that predictions using the city block and the Euclidean metrics outperform predictions using the other metrics. However the per-

---

<sup>3</sup>The Laplace estimates add 0.0001 to each bin of the HAN. This allows to perform comparisons between empty and non-empty bins in the histograms of the query structural element and a sample using Kullback–Leibler divergence.

formance achieved using city block distance is slightly better than the same using Euclidean distance in the central part of the curves. Hence, we selected the city block metric to compute the distances between histogram of atom nomenclatures (DHAN).

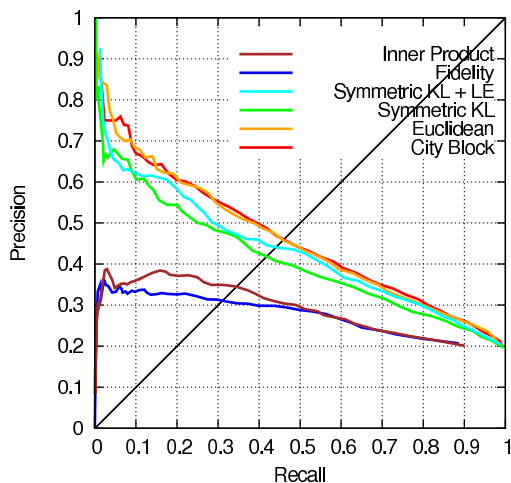


Figure A.1 Prediction results using majority vote on the top 50 samples according to different definitions of distance between histogram of atom nomenclatures.

### A.3.3.2 Evaluation of different schemes to assign weights to the samples

We performed several experiments to evaluate different alternatives to assign weights to the samples and to find an adequate number of samples that maximized the performance of the prediction in the tuning dataset. These experiments were performed with  $dASAs \leq 5\%$ ,  $dASAs \leq 15\%$ , and using the city block metric for comparing distances between histograms of atom nomenclatures.

To set a base case for the comparisons presented in this subsection, we performed predictions using majority vote on the top  $n$  unweighted samples according to the ordering determined by the values of DHAN. The results of these experiments, shown in Figure A.2, indicate that the prediction performance is not significantly affected by the number of unweighted samples.



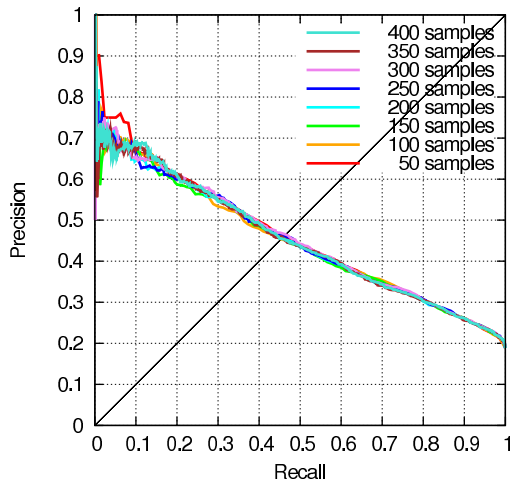


Figure A.2 **Prediction results using majority vote with different number of un-weighted samples.**

A second experiment was performed using majority vote on samples weighted using the normalized DHAN as:

$$w(s, q) = 1 - \frac{DHAN(s, q)}{\max_{r \in S(Q), q_1 \in S_r, \{DHAN(q_1, r)\}}$$

where  $s$  is a sample associated with the query structural element  $q$ ,  $S(Q)$  represents the set of all the structural elements of a query protein  $Q$ , and  $S_r$  represents the set of all the samples associated with a query structural element  $r$ . The normalization term corresponds to the largest  $DHAN$  between any structural element in a query protein and its associated samples. Hence, samples with lower  $DHAN$  values are assigned larger weights. The results of this experiment, presented in Figure A.3, indicate that the best performance was achieved using the top 20 to 30 samples. A comparison between these results and the results presented in Figure A.2 shows that the best performance was achieved when the samples were weighted using  $DHAN$ .

The following experiments evaluated the weighting schemes proposed for each member of the *PrISE* family of predictors. For *PrISE<sub>G</sub>*, the weight of each sample extracted from protein  $p$  (described by equation (4.1) in the methods section in chapter 4) is computed as the total number of samples extracted from  $p$ . Hence, samples extracted from proteins with higher general structural similarity to the query protein (according to the number of samples)

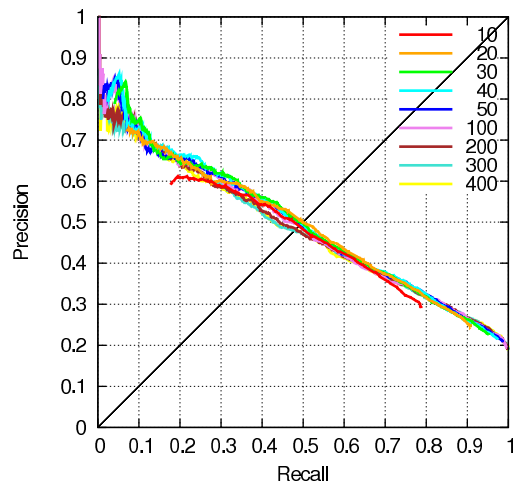


Figure A.3 **Prediction results with different number of samples and using majority vote on samples weighted using the city block distance between histogram of atom nomenclatures.**

are assigned larger weights. For  $PrISE_L$  (see equation (4.2) in chapter 4), the weight of a sample extracted from protein  $p$  is computed as the number of samples extracted from  $p$  that are associated with the structural elements in a region surrounding the query structural element (i.e. local similarity). The  $PrISE_C$  predictor (equation (4,3) in chapter 4), weights each sample using information derived from the combination of local and general similarity.

The results of an evaluation of  $PrISE_G$  using different number of samples are presented in Figure A.4. These results indicate that the best prediction was achieved using 100 to 200 samples.

On the other hand, the best results using  $PrISE_L$  are achieved using as few as 50 samples, as presented in Figure A.5.

The results of experiments using  $PrISE_C$  presented in Figure A.6 show that the prediction performances were similar when more than 300 samples were used. We decided to use 500 samples, which produced slightly better precision than the other alternatives for recall values between 0.6 and 0.75.

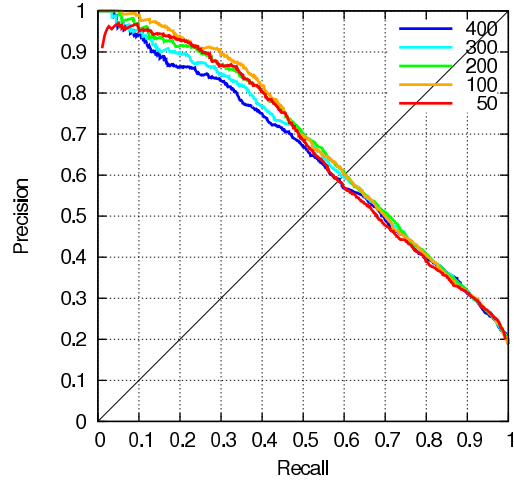


Figure A.4 **Prediction results using different number of samples and general contribution (i.e.  $PrISE_G$ ).**

A comparison of the best results derived from all the previous experiments, as well as the curve computed from a randomized prediction, are shown in Figure A.7. The randomized prediction was achieved by randomly shuffling the interface/non-interface labels of the samples in the repository of structural elements, and performing prediction using samples weighted by combined contribution. From the figure it is possible to deduce that (i) all the prediction schemes are superior than random predictions, (ii) predictions generated using weighted samples are better than predictions produced using unweighted samples, (iii) schemes that incorporate general contribution produces better results than prediction based only in local contribution, and (iv) the best performance is achieved using the contribution scheme that combines local and general information.

As a result, the experiments described in chapter 4 were performed using the top samples based on the city block metric for DHAN, differences  $\leq 5\%$  between the accessible surface areas of the central residues of the samples and the query structural elements, and differences  $\leq 15\%$  between the accessible surface areas of the samples and each query structural element. The

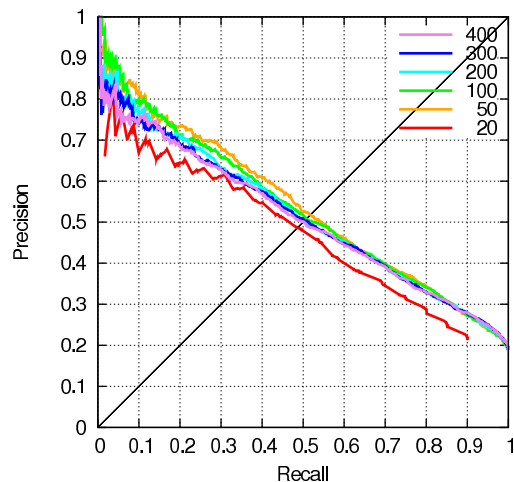


Figure A.5 Prediction results using set of samples of different size and local contribution (i.e.  $PrISE_L$ ).

number of samples used by  $PrISE_G$ ,  $PrISE_L$ , and  $PrISE_C$  were set to 200, 50, and 500 respectively.

#### A.4 Selection of a threshold value for performing classification

The  $PrISE$  family of predictors produce a probability that indicates the likelihood of each residue on the surface of the protein of being an interface residue. The selection of a threshold value on this probability allows to label each residue as interface / non-interface. The lower the threshold value, the more residues are labeled as interfaces. We used the results of the  $PrISE_C$  predictor presented in Figure A.7 to select a threshold value of 0.34, which produced predictions with similar precision and recall values. This value was used as default for all the predictors of the  $PrISE$  family throughout the experiments presented in chapter 4.

#### A.5 Additional comparisons of $PrISE_L$ , $PrISE_G$ and $PrISE_C$

The performances of  $PrISE_L$ ,  $PrISE_G$  and  $PrISE_C$  on the DS24Carl, DS56bound and DS56unbound datasets are shown in Figures A.8 to A.10. Samples extracted from homologs of the same species than the query proteins were filtered out from the repository of structural elements. In terms of performance, the precision recall curves indicate that  $random < PrISE_L < PrISE_G \leq PrISE_C$ .

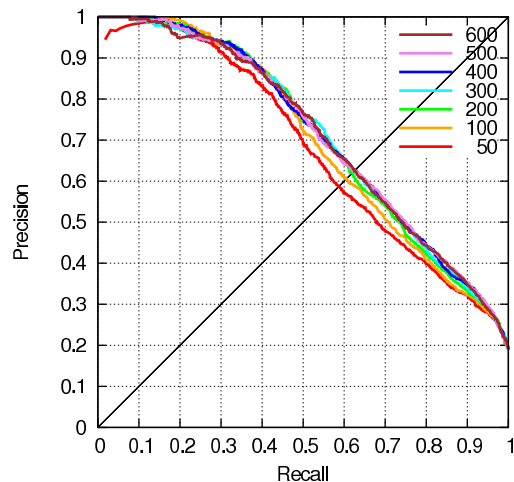


Figure A.6 Prediction results using number of samples and combined contribution (i.e.  $PrISE_C$ ).

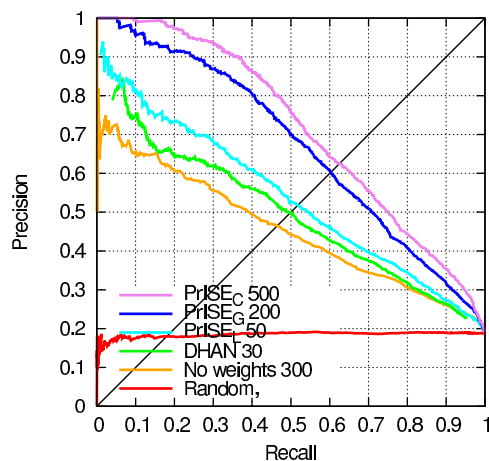


Figure A.7 Prediction results using different weighting schemes. The number in the labels indicates the number of samples used for prediction.

An example of the relationship between the scores of  $PrISE_L$ ,  $PrISE_G$  and  $PrISE_C$ , and the actual interface/non-interface labels for some residues in the protein 1ohz-B is illustrated in Figure A.11. From this figure is clear that  $PrISE_C$  is successful correcting some erroneous predictions generated by both  $PrISE_L$  and  $PrISE_G$  (e.g. residues 19, 25, and 26) or by only

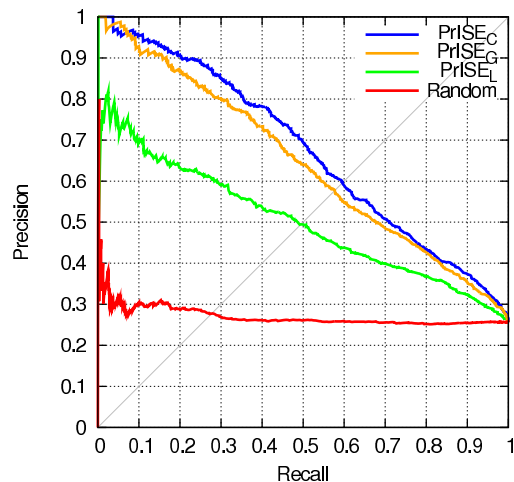


Figure A.8 Comparison of  $PrISE_L$ ,  $PrISE_G$ , and  $PrISE_C$  using the dataset DS24Carl.

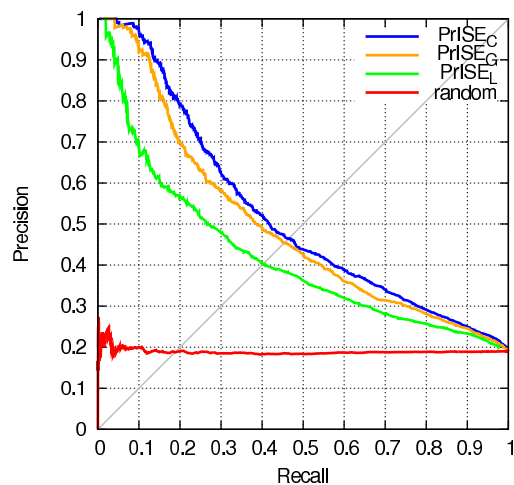


Figure A.9 Comparison of  $PrISE_L$ ,  $PrISE_G$ , and  $PrISE_C$  using the dataset DS56Bound.

one of them (e.g. amino acids 2, 18, and 24).  $PrISE_C$  sometimes generates wrong predictions in cases where  $PrISE_L$  or  $PrISE_G$  make correct predictions (e.g. residues 6, 11, 14, and 20). However, our experimental results indicate that the number of errors fixed by  $PrISE_C$  exceeds the number of errors it introduces.

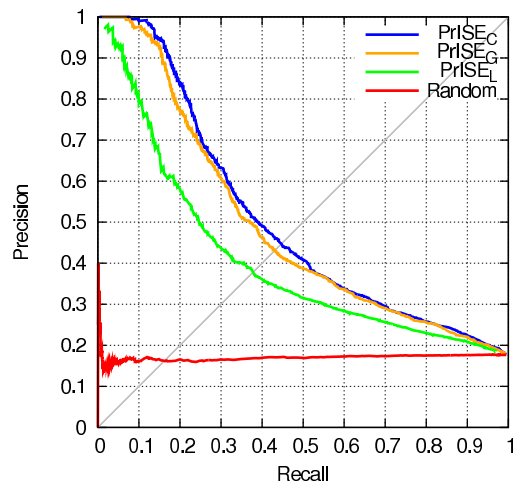


Figure A.10 Comparison of  $PrISE_L$ ,  $PrISE_G$ , and  $PrISE_C$  using the dataset DS56Unbound.

### A.6 Additional evaluation of the impact of homologs of the query protein in the predictions

The impact caused on the predictions by filtering out from the repository samples derived from sequence homologs of the query proteins is presented in Figures A.12 to A.14. This evaluation was performed using  $PrISE_C$  on the DS24Carl, DS56Bound and DS56Unbound datasets. These figures show that the prediction performances are lower when samples extracted from homologs of the query proteins are filtered out from the repository of structural elements.

### A.7 Additional comparison with two prediction methods based on geometrical conserved local surfaces

A comparison of the predictors of the  $PrISE$  family with the methods presented in (26; 27) using the DS24Carl dataset and excluding from the repository of structural elements samples extracted from homologs (without regarding the species) is presented in Table A.3. According to

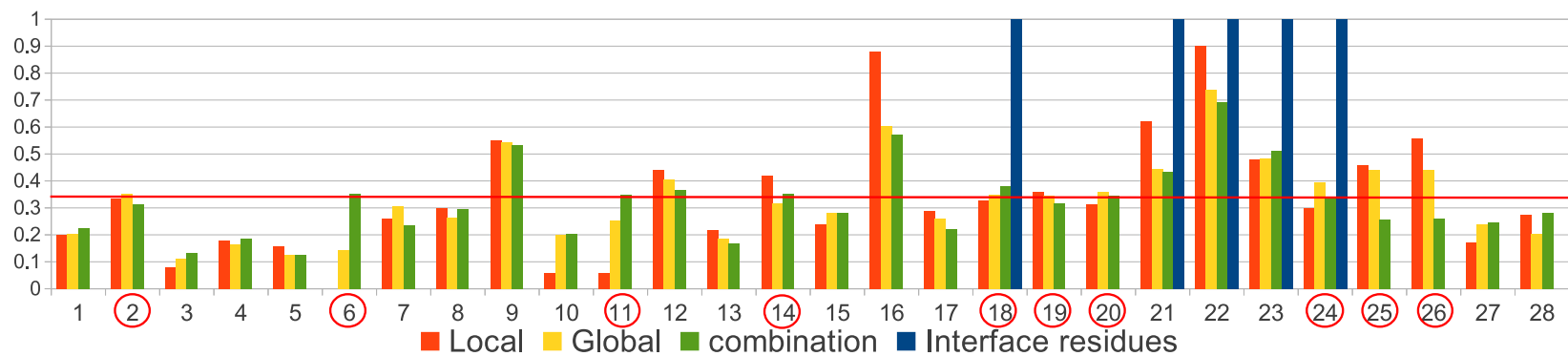


Figure A.11 **Example of the scores generated by  $PrISE_L$ ,  $PrISE_G$ , and  $PrISE_C$ .** This figure show (in the vertical axis) the score generated by  $PrISE_L$ ,  $PrISE_G$ , and  $PrISE_C$  as well as the actual interface residues for the first 28 residues (shown in the horizontal axis) in the sequence of the protein chain 1ohz-B. The horizontal red line signals the threshold computed on the scores (0.34) to differentiate between interfaces and non-interfaces.



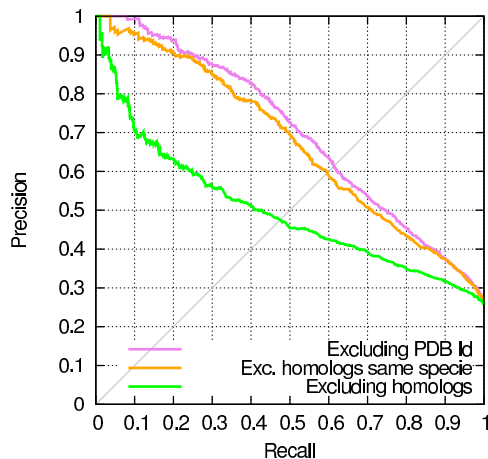


Figure A.12 Performance of *PrISE<sub>C</sub>* with DS24Carl using three schemes for excluding similar proteins.

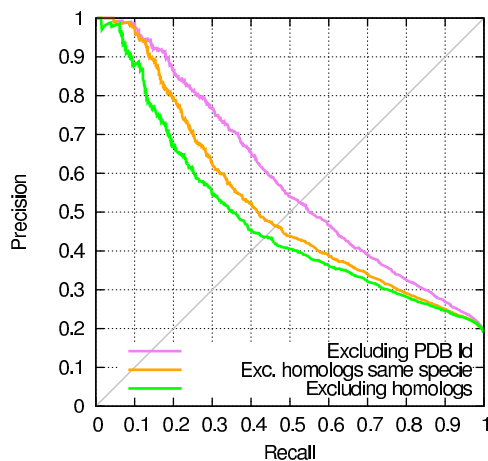


Figure A.13 Performance of *PrISE<sub>C</sub>* with DS56Bound using different schemes for excluding similar proteins.

this table, all the members of the *PrISE* family outperform the classifiers presented in (26; 27) in terms of precision, recall, and F1.

We also evaluated the performance of the *PrISE* family of predictors using the ProtInDb and the  $ProtInDb \cap PQS$  repositories of structural elements. The results of these comparisons, shown in Tables A.4 and A.5, indicate that predictors that use samples extracted from the ProtInDb repository slightly outperform predictors that extract samples from the  $ProtInDb \cap PQS$  repository.

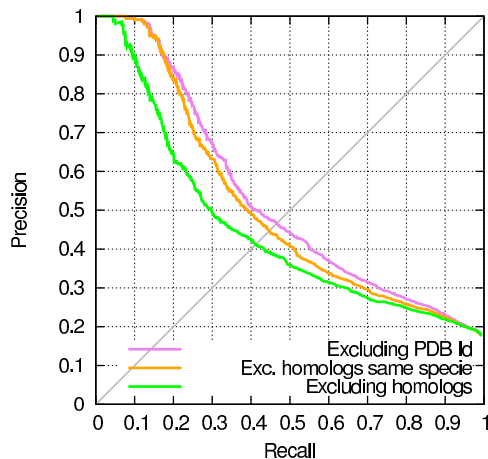


Figure A.14 Performance of  $PrISE_C$  with DS56Unbound using several schemes for excluding similar proteins.

Table A.3 Performance of different methods on the DS24Carl dataset. Performance measures are computed as the average on the set of 24 proteins. Precision and recall values for Carl08 and Carl10 were taken from (26) and (27) respectively. Samples derived from homologs of the query proteins were excluded from the ProtInDb repository.

Predictor	Precision %	Recall %	F1 %	Accuracy %	CC %	AUC %
Carl08	31.5	35.3	33.3	-	-	-
Carl10	32.0	34.0	33.0	-	-	-
$PrISE_L$	41.1	52.3	44.1	66.3	21.1	66.7
$PrISE_G$	45.6	48.6	45.4	69.9	24.0	68.8
$PrISE_C$	48.7	46.4	45.8	72.2	26.3	69.2

## A.8 Abbreviations

$dAS_{Ares}$  - Difference between the accessible surface area of the central residues and of two structural elements.

$dAS_{Ase}$  - Difference between the accessible surface area of two structural elements.

DHAN - distance between two histograms of atom nomenclatures.

Sample - a structural element retrieved from a repository of structural elements.

Table A.4 **Performance of *PrISE* predictors using different repositories of structural elements and excluding homologs.** Performance measures are computed as the average on the set of 24 proteins in the DS24Carl dataset. Samples extracted from homologs (without regarding the species) were excluded from the prediction process. The column “ProtInDb” indicates whether samples were extracted from the ProtInDb repository (marked with a tick), or from the  $ProtInDb \cap PQS$  repository.

Predictor	ProtInDb	Precision %	Recall %	F1 %	Accuracy %	CC %	AUC %
<i>PrISE<sub>L</sub></i>	✓	41.1	52.3	44.1	66.3	21.1	66.7
		41.0	50.7	43.3	66.6	19.2	66.6
<i>PrISE<sub>G</sub></i>	✓	45.6	48.6	45.4	69.9	24.0	68.8
		43.4	47.7	43.8	69.3	21.2	67.3
<i>PrISE<sub>C</sub></i>	✓	48.7	46.4	45.8	72.2	26.3	69.2
		45.5	47.7	45.0	70.4	23.4	69.5

Table A.5 **Performance of *PrISE* methods using different repositories and excluding homologs of the same species.** The performance measures were computed as the averages on the proteins in the DS24Carl dataset. Samples extracted from homologs from the same species than the query proteins were filtered out from the prediction process. The “ProtInDb” column indicates whether the samples were extracted from the ProtInDb repository (marked with a tick), or from the  $ProtInDb \cap PQS$  repository.

Predictor	ProtInDb	Precision %	Recall %	F1 %	Accuracy %	CC %	AUC %
<i>PrISE<sub>L</sub></i>	✓	45.1	56.2	50.0	69.1	27.1	70.5
		46.3	55.7	48.6	69.9	26.8	71.0
<i>PrISE<sub>G</sub></i>	✓	53.9	58.7	56.2	75.1	36.8	75.6
		51.6	56.7	52.5	74.0	33.1	74.3
<i>PrISE<sub>C</sub></i>	✓	58.3	58.3	58.3	77.5	40.6	77.1
		54.4	58.4	54.8	75.5	36.6	76.2

**BIBLIOGRAPHY**

- [1] Jmol: an open-source java viewer for chemical structures in 3d. URL: <http://www.jmol.org/>.  
58
- [2] Ahmad, S. and Mizuguchi, K. (2011). Partner-aware prediction of interacting residues in protein-protein complexes from sequence data. *PLoS One*, 6(12):e29104. 93
- [3] Anashkina, A., Kuznetsov, E., Esipova, N., and Tumanyan, V. (2007). Comprehensive statistical analysis of residues interaction specificity at protein-protein interfaces. *Proteins*, 67(4):1060–1077. 12
- [4] Ansari, S. and Helms, V. (2005). Statistical analysis of predominantly transient protein-protein interfaces. *Proteins*, 61(2):344–355. 12
- [5] Arkin, M. R. and Whitty, A. (2009). The road less traveled: modulating signal transduction enzymes by inhibiting their protein-protein interactions. *Curr Opin Chem Biol*, 13(3):284–290. 1
- [6] Assi, S. A., Tanaka, T., Rabbitts, T. H., and Fernandez-Fuentes, N. (2010). Pcrpi: Presaging critical residues in protein interfaces, a new computational tool to chart hot spots in protein interfaces. *Nucleic Acids Res*, 38(6):e86. 1
- [7] Aytuna, A. S., Gursoy, A., and Keskin, O. (2005). Prediction of protein-protein interactions by combining structure and sequence conservation in protein interfaces. *Bioinformatics*, 21(12):2850–2855. 1
- [8] Back, J. W., de Jong, L., Muijsers, A. O., and de Koster, C. G. (2003). Chemical cross-linking and mass spectrometry for protein structural modeling. *J Mol Biol*, 331(2):303–313.  
2

- [9] Bahadur, R. P. and Zacharias, M. (2008). The interface of protein-protein complexes: analysis of contacts and prediction of interactions. *Cell Mol Life Sci*, 65(7-8):1059–1072. [3](#)
- [10] Baldi, P., Brunak, S., Chauvin, Y., Andersen, C. A., and Nielsen, H. (2000). Assessing the accuracy of prediction algorithms for classification: an overview. *Bioinformatics*, 16(5):412–424. [71](#)
- [11] Bartoli, L., Martelli, P. L., Rossi, I., Fariselli, P., and Casadio, R. (2009). Prediction of protein-protein interacting sites: How to bridge molecular events to large scale protein interaction networks. In Degano, Pierpaolo and Gorrieri, Roberto, editor, *CMSB 09: Proceedings of the 7th International Conference on Computational Methods in Systems Biology*, pages 1–17, Berlin, Heidelberg. Springer-Verlag. [3](#), [4](#), [7](#), [12](#), [28](#), [29](#), [49](#), [62](#)
- [12] Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N., and Bourne, P. E. (2000). The protein data bank. *Nucleic Acids Res*, 28(1):235–242. [2](#), [3](#), [12](#), [17](#), [32](#), [63](#), [66](#)
- [13] Betel, D., Breitkreuz, K. E., Isserlin, R., Dewar-Darch, D., Tyers, M., and Hogue, C. W. V. (2007). Structure-templated predictions of novel protein interactions from sequence information. *PLoS Comput Biol*, 3(9):1783–1789. [1](#)
- [14] Bickerton, G. R., Higuieruelo, A. P., and Blundell, T. L. (2011). Comprehensive, atomic-level characterization of structurally characterized protein-protein interactions: the piccolo database. *BMC Bioinformatics*, 12(1):313. [14](#)
- [15] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer. [36](#)
- [16] Block, P., Paern, J., HÄijllermeier, E., Sanschagrın, P., Sotriffer, C. A., and Klebe, G. (2006). Physicochemical descriptors to discriminate protein-protein interactions in permanent and transient complexes selected by means of machine learning algorithms. *Proteins*, 65(3):607–622. [12](#)
- [17] Blundell, J. and Johnson, L. (1976). *Protein crystallography*. Academic Press, New York.

- [18] Bock, M. E., Garutti, C., and Guerra, C. (2007). Discovery of similar regions on protein surfaces. *J Comput Biol*, 14(3):285–299. [94](#)
- [19] Bordner, A. J. and Abagyan, R. (2005). Statistical analysis and prediction of protein-protein interfaces. *Proteins*, 60(3):353–366. [3](#)
- [20] Bordner, A. J. and Gorin, A. A. (2008). Comprehensive inventory of protein complexes in the protein data bank from consistent classification of interfaces. *BMC Bioinformatics*, 9:234. [3](#)
- [21] Bradford, J. R., Needham, C. J., Bulpitt, A. J., and Westhead, D. R. (2006). Insights into protein-protein interfaces using a bayesian network prediction method. *J Mol Biol*, 362(2):365–386. [3](#), [30](#), [39](#)
- [22] Bradford, J. R. and Westhead, D. R. (2005). Improved prediction of protein-protein binding sites using a support vector machines approach. *Bioinformatics*, 21(8)(8):1487–1494. [3](#), [5](#), [30](#), [31](#), [33](#), [39](#), [52](#), [53](#), [55](#), [56](#), [62](#)
- [23] Budowski-Tal, I., Nov, Y., and Kolodny, R. (2010). Fragbag, an accurate representation of protein structure, retrieves structural neighbors from the entire pdb quickly and accurately. *Proc Natl Acad Sci U S A*, 107(8):3481–3486. [94](#)
- [24] Burgoyne, N. J. and Jackson, R. M. (2006). Predicting protein interaction sites: binding hot-spots in protein-protein and protein-ligand interfaces. *Bioinformatics*, 22(11):1335–1342. [4](#), [28](#)
- [25] Caffrey, D. R., Somaroo, S., Hughes, J. D., Mintseris, J., and Huang, E. S. (2004). Are protein-protein interfaces more conserved in sequence than the rest of the protein surface? *Protein Sci*, 13(1):190–202. [12](#)
- [26] Carl, N., Konc, J., and Janežič, D. (2008). Protein surface conservation in binding sites. *J. Chem. Inf. Model*, 48(6):1279–1286. [ix](#), [x](#), [5](#), [12](#), [63](#), [64](#), [70](#), [74](#), [75](#), [85](#), [106](#), [108](#), [109](#)

- [27] Carl, N., Konc, J., Vehar, B., and Janežič, D. (2010). Protein-protein binding site prediction by local structural alignment. *J Chem Inf Model*, 50 (10):1906–1913. [ix](#), [x](#), [5](#), [63](#), [64](#), [74](#), [75](#), [85](#), [106](#), [108](#), [109](#)
- [28] Cha, S. (2007). Comprehensive survey on distance/similarity measures between probability density functions. *International Journal of mathematical models and methods in applied sciences*, 1(4):300–307. [98](#)
- [29] Chakrabarti, P. and Janin, J. (2002). Dissecting protein-protein recognition sites. *Proteins*, 47(3):334–343. [4](#), [29](#)
- [30] Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357. [37](#)
- [31] Chen, H. and Zhou, H.-X. (2005). Prediction of interface residues in protein-protein complexes by a consensus neural network method: Test against nmr data. *Proteins: Structure, Function, and Bioinformatics*, 61(1):21–35. [3](#), [4](#), [12](#), [28](#), [36](#), [80](#)
- [32] Chen, X. W. and Jeong, J. C. (2009). Sequence-based prediction of protein interaction sites with an integrative method. *Bioinformatics*, 25(5):585–591. [3](#), [4](#), [12](#), [28](#)
- [33] Chen, Y. C. and Lim, C. (2008). Common physical basis of macromolecule-binding sites in proteins. *Nucleic Acids Res*, 36(22):7078–7087. [12](#)
- [34] Choi, Y. S., Yang, J.-S., Choi, Y., Ryu, S. H., and Kim, S. (2009). Evolutionary conservation in multiple faces of protein interaction. *Proteins*, 77(1):14–25. [12](#)
- [35] Chung, J.-L., Wang, W., and Bourne, P. E. (2006). Exploiting sequence and structure homologs to identify protein-protein binding sites. *Proteins*, 62(3):630–640. [3](#), [4](#), [5](#), [28](#), [36](#), [63](#)
- [36] Chung, J.-L., Wang, W., and Bourne, P. E. (2007). High-throughput identification of interacting protein-protein binding sites. *BMC Bioinformatics*, 8:223. [62](#), [93](#)

- [37] Conte, L. L., Chothia, C., and Janin, J. (1999). The atomic structure of protein-protein recognition sites. *J Mol Biol*, 285(5):2177–2198. [12](#)
- [38] Csaba, G., Birzele, F., and Zimmer, R. (2008). Protein structure alignment considering phenotypic plasticity. *Bioinformatics*, 24(16):i98–104. [94](#)
- [39] Dayhoff, J. E., Shoemaker, B. A., Bryant, S. H., and Panchenko, A. R. (2010). Evolution of protein binding modes in homooligomers. *J Mol Biol*, 395(4):860–870. [5](#), [63](#)
- [40] De, S., Krishnadev, O., Srinivasan, N., and Rekha, N. (2005). Interaction preferences across protein-protein interfaces of obligatory and non-obligatory components are different. *BMC Struct Biol*, 5:15. [4](#), [12](#), [29](#)
- [41] de Vries, S. and Bonvin, A. (2011). CPORT: A Consensus Interface Predictor and Its Performance in Prediction-Driven Docking with HADDOCK. *PLoS ONE*, 6(3):e17695. [3](#)
- [42] de Vries, S. J. and Bonvin, A. M. J. J. (2006). Intramolecular surface contacts contain information about protein-protein interface regions. *Bioinformatics*, 22(17):2094–2098. [12](#)
- [43] de Vries, S. J. and Bonvin, A. M. J. J. (2008). How proteins get in touch: interface prediction in the study of biomolecular complexes. *Curr Protein Pept Sci*, 9(4):394–406. [3](#), [4](#), [7](#), [12](#), [29](#), [49](#), [62](#)
- [44] Deng, L., Guan, J., Dong, Q., and Zhou, S. (2009). Prediction of protein-protein interaction sites using an ensemble method. *BMC Bioinformatics*, 10:426. [3](#), [4](#), [28](#)
- [45] Ding, X.-M., Pan, X.-Y., Xu, C., and Shen, H.-B. (2010). Computational prediction of dna-protein interactions: a review. *Curr Comput Aided Drug Des*, 6(3):197–206. [92](#)
- [46] Doerr, A. (2008). Membrane protein structures. *Nature Methods*, 6(1):35–35. [5](#)
- [47] Dong, Q., Wang, X., Lin, L., and Guan, Y. (2007). Exploiting residue-level and profile-level interface propensities for usage in binding sites prediction of proteins. *BMC Bioinformatics*, 8:147. [3](#), [12](#)



- [48] Douguet, D., Chen, H.-C., Tovchigrechko, A., and Vakser, I. A. (2006). Dockground resource for studying protein-protein interfaces. *Bioinformatics*, 22(21):2612–2618. [1](#)
- [49] Du, X., Cheng, J., and Song, J. (2009). Identifying protein-protein interaction sites using covering algorithm. *Int J Mol Sci*, 10(5):2190–2202. [4](#), [28](#)
- [50] Eichborn, J. V., Günther, S., and Preissner, R. (2010). Structural features and evolution of protein-protein interactions. *Genome Inform*, 22:1–10. [4](#), [29](#)
- [51] Engelen, S., Trojan, L. A., Sacquin-Mora, S., Lavery, R., and Carbone, A. (2009). Joint evolutionary trees: a large-scale method to predict protein interfaces based on sequence sampling. *PLoS Comput Biol*, 5(1):e1000267. [29](#)
- [52] Ewing, R. M., Chu, P., Elisma, F., Li, H., Taylor, P., Climie, S., McBroom-Cerajewski, L., Robinson, M. D., O’Connor, L., Li, M., Taylor, R., Dharsee, M., Ho, Y., Heilbut, A., Moore, L., Zhang, S., Ornatsky, O., Bukhman, Y. V., Ethier, M., Sheng, Y., Vasilescu, J., Abu-Farha, M., Lambert, J.-P., Duewel, H. S., Stewart, I. I., Kuehl, B., Hogue, K., Colwill, K., Gladwish, K., Muskat, B., Kinach, R., Adams, S.-L., Moran, M. F., Morin, G. B., Topaloglou, T., and Figeys, D. (2007). Large-scale mapping of human protein-protein interactions by mass spectrometry. *Mol Syst Biol*, 3:89. [28](#)
- [53] Ezkurdia, I., Bartoli, L., Fariselli, P., Casadio, R., Valencia, A., and Tress, M. L. (2009). Progress and challenges in predicting protein-protein interaction sites. *Brief Bioinform*, 10(3):233–246. [3](#), [4](#), [7](#), [12](#), [16](#), [28](#), [29](#), [62](#)
- [54] Fariselli, P., Pazos, F., Valencia, A., and Casadio, R. (2002). Prediction of protein-protein interaction sites in heterocomplexes with neural networks. *Eur J Biochem*, 269(5):1356–1361. [3](#), [4](#), [16](#), [28](#), [62](#)
- [55] Ferentz, A. E. and Wagner, G. (2000). NMR spectroscopy: a multifaceted approach to macromolecular structure. *Q Rev Biophys*, 33(1):29–65. [2](#)
- [56] Fernández-Recio, J. (2011). Prediction of protein binding sites and hot spots. *Wiley Interdisciplinary Reviews: Computational Molecular Science*. [3](#), [4](#), [12](#), [62](#)

- [57] Fernández-Recio, J., Totrov, M., and Abagyan, R. (2004). Identification of protein-protein interaction sites from docking energy landscapes. *J Mol Biol*, 335(3):843–865. [4](#), [28](#)
- [58] Fernández-Recio, J., Totrov, M., Skorodumov, C., and Abagyan, R. (2005). Optimal docking area: a new method for predicting protein-protein interaction sites. *Proteins*, 58(1):134–143. [29](#)
- [59] Finn, R. D., Mistry, J., Tate, J., Coggill, P., Heger, A., Pollington, J. E., Gavin, O. L., Gunasekaran, P., Ceric, G., Forslund, K., Holm, L., Sonnhammer, E. L. L., Eddy, S. R., and Bateman, A. (2010). The pfam protein families database. *Nucleic Acids Res*, 38(Database issue):D211–D222. [15](#)
- [60] Frishman, D. and Argos, P. (1995). Knowledge-based protein secondary structure assignment. *Proteins*, 23(4):566–579. [36](#)
- [61] Gall, T. L., Romero, P. R., Cortese, M. S., Uversky, V. N., and Dunker, A. K. (2007). Intrinsic disorder in the protein data bank. *J Biomol Struct Dyn*, 24(4):325–342. [6](#)
- [62] Gallet, X., Charlotiaux, B., Thomas, A., and Brasseur, R. (2000). A fast method to predict protein interaction sites from sequences. *J Mol Biol*, 302(4):917–926. [12](#), [28](#)
- [63] Gao, M. and Skolnick, J. (2010). Structural space of protein-protein interfaces is degenerate, close to complete, and highly connected. *Proc Natl Acad Sci U S A*, 107(52):22517–22522. [6](#), [12](#)
- [64] Gavin, A.-C., Bösch, M., Krause, R., Grandi, P., Marzioch, M., Bauer, A., Schultz, J., Rick, J. M., Michon, A.-M., Cruciat, C.-M., Remor, M., Höfert, C., Schelder, M., Brajenovic, M., Ruffner, H., Merino, A., Klein, K., Hudak, M., Dickson, D., Rudi, T., Gnau, V., Bauch, A., Bastuck, S., Huhse, B., Leutwein, C., Heurtier, M.-A., Copley, R. R., Edelmann, A., Querfurth, E., Rybin, V., Drewes, G., Raida, M., Bouwmeester, T., Bork, P., Seraphin, B., Kuster, B., Neubauer, G., and Superti-Furga, G. (2002). Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature*, 415(6868):141–147. [28](#)

- [65] Giot, L., Bader, J. S., Brouwer, C., Chaudhuri, A., Kuang, B., Li, Y., Hao, Y. L., Ooi, C. E., Godwin, B., Vitols, E., Vijayadamodar, G., Pochart, P., Machineni, H., Welsh, M., Kong, Y., Zerhusen, B., Malcolm, R., Varrone, Z., Collis, A., Minto, M., Burgess, S., McDaniel, L., Stimpson, E., Spriggs, F., Williams, J., Neurath, K., Ioime, N., Agee, M., Voss, E., Furtak, K., Renzulli, R., Aanensen, N., Carrolla, S., Bickelhaupt, E., Lazovatsky, Y., DaSilva, A., Zhong, J., Stanyon, C. A., Finley, R. L., White, K. P., Braverman, M., Jarvie, T., Gold, S., Leach, M., Knight, J., Shimkets, R. A., McKenna, M. P., Chant, J., and Rothberg, J. M. (2003). A protein interaction map of drosophila melanogaster. *Science*, 302(5651):1727–1736. [28](#), [62](#)
- [66] Gronwald, W. and Kalbitzer, H. R. (2010). Automated protein nmr structure determination in solution. *Methods Mol Biol*, 673:95–127. [2](#)
- [67] Gruber, J., Zawaira, A., Saunders, R., Barrett, C. P., and Noble, M. E. M. (2007). Computational analyses of the surface properties of protein-protein interfaces. *Acta Crystallogr D Biol Crystallogr*, 63(Pt 1):50–57. [12](#)
- [68] Guharoy, M. and Chakrabarti, P. (2007). Secondary structure based analysis and classification of biological interfaces: identification of binding motifs in protein-protein interactions. *Bioinformatics*, 23(15):1909–1918. [12](#)
- [69] Guharoy, M. and Chakrabarti, P. (2010). Conserved residue clusters at protein-protein interfaces and their use in binding site identification. *BMC Bioinformatics*, 11(1):286. [3](#), [5](#), [12](#), [63](#)
- [70] Guo, Y., Yu, L., Wen, Z., and Li, M. (2008). Using support vector machine combined with auto covariance to predict protein-protein interactions from protein sequences. *Nucleic Acids Res*, 36(9):3025–3030. [3](#), [4](#), [28](#)
- [71] GÅijnther, S., May, P., Hoppe, A., FrÅmmel, C., and Preissner, R. (2007). Docking without docking: Isearch–prediction of interactions using known interfaces. *Proteins*, 69(4):839–844. [1](#)

- [72] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. (2009). The weka data mining software: An update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18. [36](#)
- [73] Hanson, R. (2010). Jmol-a paradigm shift in crystallographic visualization. *Journal of Applied Crystallography*, 43(5):1250–1260. [20](#)
- [74] Hayashida, M., Kamada, M., Song, J., and Akutsu, T. (2011). Conditional random field approach to prediction of protein-protein interactions using domain information. *BMC Systems Biology*, 5(Suppl 1):S8. [3](#)
- [75] Henrick, K. and Thornton, J. (1998). PQS: a protein quaternary structure file server. *Trends in biochemical sciences*, 23(9):358. [15](#), [66](#)
- [76] Henschel, A., Kim, W. K., and Schroeder, M. (2006). Equivalent binding sites reveal convergently evolved interaction motifs. *Bioinformatics*, 22(5):550–555. [12](#)
- [77] Henschel, A., Winter, C., Kim, W. K., and Schroeder, M. (2007). Using structural motif descriptors for sequence-based binding site prediction. *BMC Bioinformatics*, 8 Suppl 4:S5. [4](#), [28](#)
- [78] Herrandez, A. (2006). Biomolecules in the computer: Jmol to the rescue. *Biochem Mol Biol Educ*, 34(4):255–261. [20](#)
- [79] Higa, R. H. and Tozzi, C. L. (2008). A simple and efficient method for predicting protein-protein interaction sites. *Genet Mol Res*, 7(3):898–909. [4](#), [28](#)
- [80] Horan, K., Shelton, C. R., and Girke, T. (2010). Predicting conserved protein motifs with sub-hmms. *BMC Bioinformatics*, 11:205. [4](#)
- [81] Hsu, C.-M., Chen, C.-Y., Liu, B.-J., Huang, C.-C., Laio, M.-H., Lin, C.-C., and Wu, T.-L. (2007). Identification of hot regions in protein-protein interactions by sequential pattern mining. *BMC Bioinformatics*, 8 Suppl 5:S8. [4](#), [28](#)
- [82] Hu, J. and Yan, C. (2009). A tool for calculating binding-site residues on proteins from pdb structures. *BMC Struct Biol*, 9:52. [14](#)

- [83] Hubbard, S. and Thornton, J. (1993). Naccess, computer program, department of biochemistry and molecular biology. *University College London*. URL: <http://www.bioinf.manchester.ac.uk/naccess/>. 18, 19, 31, 36, 65
- [84] Humphris, E. L. and Kortemme, T. (2007). Design of multi-specificity in protein interfaces. *PLoS Comput Biol*, 3(8):e164. 6, 12
- [85] Hwang, H., Pierce, B., Mintseris, J., Janin, J., and Weng, Z. (2008). Protein-protein docking benchmark version 3.0. *Proteins*, 73(3):705–709. 33, 47, 70
- [86] Janin, J. (2005). Assessing predictions of protein-protein interaction: the capri experiment. *Protein Sci*, 14(2):278–283. 33
- [87] Janin, J., Henrick, K., Moult, J., Eyck, L. T., Sternberg, M. J. E., Vajda, S., Vakser, I., Wodak, S. J., and of PRedicted Interactions, C. A. (2003). Capri: a critical assessment of predicted interactions. *Proteins*, 52(1):2–9. 33
- [88] Janin, J. and Wodak, S. (2007). The third capri assessment meeting toronto, canada, april 20-21, 2007. *Structure*, 15(7):755–759. 70
- [89] Jefferson, E. R., Walsh, T. P., Roberts, T. J., and Barton, G. J. (2007). Snappi-db: a database and api of structures, interfaces and alignments for protein-protein interactions. *Nucleic Acids Res*, 35(Database issue):D580–D589. 15
- [90] Jones, S. and Mukarami, Y. (2007). Patch prediction of protein interaction sites: Validation of a scoring function for an online server. *Bioinformatics Research and Development*, 4414/2007:303–313. 5, 29, 32, 33, 38, 39, 45, 53, 62
- [91] Jones, S. and Thornton, J. (1996). Principles of protein-protein interactions. *Proceedings of the National Academy of Sciences of the United States of America*, 93(1):13. 3, 4, 12, 16, 29
- [92] Jones, S. and Thornton, J. M. (1995). Protein-protein interactions: a review of protein dimer structures. *Prog Biophys Mol Biol*, 63(1):31–65. 12, 31, 45

- [93] Jones, S. and Thornton, J. M. (1997a). Analysis of protein-protein interaction sites using surface patches. *J Mol Biol*, 272(1):121–132. 4, 29, 31
- [94] Jones, S. and Thornton, J. M. (1997b). Prediction of protein-protein interaction sites using patch analysis. *J Mol Biol*, 272(1):133–143. 5, 29, 31, 38, 39, 45, 53, 62
- [95] Jordan, R. A., El-Manzalawy, Y., Dobbs, D., and Honavar, V. (2010a). A modular method to predict protein-protein interaction patches. Poster presented at the 18th annual international conference on intelligent systems for molecular biology (ISMB), 2010. 13, 24
- [96] Jordan, R. A., EL-Manzalawy, Y., Dobbs, D., and Honavar, V. (2011). Evaluation of discontinuous B-cell epitopes prediction servers. Poster presented at the Second Immunoinformatics and Computational Immunology Workshop (ICIW 2011) - ACM International Conference on Bioinformatics and Computational Biology (ACM-BCB). 14, 25
- [97] Jordan, R. A., EL-Manzalawy, Y., Dobbs, D., and Honavar, V. (2012a). A modular approach to predict protein-protein interaction sites. <http://pointers.cs.iastate.edu>. 13, 24
- [98] Jordan, R. A., EL-Manzalawy, Y., Dobbs, D., and Honavar, V. (2012b). Predicting protein-protein interface residues using local surface structural similarity. *BMC Bioinformatics*, 13:41. 10, 13, 24, 90
- [99] Jordan, R. A., Wu, F., Dobbs, D., and Honavar, V. (2010b). ProtInDb: A data base of protein-protein interface residues. URL: <http://protindb.cs.iastate.edu>. 66
- [100] Keskin, O. and Nussinov, R. (2007). Similar binding sites and different partners: implications to shared proteins in cellular pathways. *Structure*, 15(3):341–354. 1
- [101] Kifer, I., Nussinov, R., and Wolfson, H. J. (2011). Gossip: a method for fast and accurate global alignment of protein structures. *Bioinformatics*, 27(7):925–932. 94
- [102] Kim, P. M., Lu, L. J., Xia, Y., and Gerstein, M. B. (2006a). Relating three-dimensional structures to protein networks provides evolutionary insights. *Science*, 314(5807):1938–1941.

- [103] Kim, W. K., Henschel, A., Winter, C., and Schroeder, M. (2006b). The many faces of protein-protein interactions: A compendium of interface geometry. *PLoS Comput Biol*, 2(9):e124. [12](#)
- [104] Kim, W. K. and Ison, J. C. (2005). Survey of the geometric association of domain-domain interfaces. *Proteins*, 61(4):1075–1088. [12](#)
- [105] Kinjo, A. R. and Nakamura, H. (2010). Geometric similarities of protein-protein interfaces at atomic resolution are only observed within homologous families: an exhaustive structural classification study. *J Mol Biol*, 399(3):526–540. [12](#)
- [106] Konc, J. and Janežič, D. (2007). Protein-protein binding-sites prediction by protein surface structure conservation. *J Chem Inf Model*, 47(3):940–944. [5](#), [63](#)
- [107] Konc, J. and Janežič, D. (2010). Probis algorithm for detection of structurally similar protein binding sites by local structural alignment. *Bioinformatics*, 26(9):1160–1168. [5](#), [12](#), [28](#), [63](#), [93](#)
- [108] Krissinel, E. and Henrick, K. (2007). Inference of macromolecular assemblies from crystalline state. *J Mol Biol*, 372(3):774–797. [14](#), [15](#)
- [109] Krogan, N. J., Cagney, G., Yu, H., Zhong, G., Guo, X., Ignatchenko, A., Li, J., Pu, S., Datta, N., Tikuisis, A. P., Punna, T., Peregrín-Alvarez, J. M., Shales, M., Zhang, X., Davey, M., Robinson, M. D., Paccanaro, A., Bray, J. E., Sheung, A., Beattie, B., Richards, D. P., Canadian, V., Lalev, A., Mena, F., Wong, P., Starostine, A., Canete, M. M., Vlasblom, J., Wu, S., Orsi, C., Collins, S. R., Chandran, S., Haw, R., Rilstone, J. J., Gandi, K., Thompson, N. J., Musso, G., Onge, P. S., Ghanny, S., Lam, M. H. Y., Butland, G., Altaf-Ul, A. M., Kanaya, S., Shilatifard, A., O’Shea, E., Weissman, J. S., Ingles, C. J., Hughes, T. R., Parkinson, J., Gerstein, M., Wodak, S. J., Emili, A., and Greenblatt, J. F. (2006). Global landscape of protein complexes in the yeast *saccharomyces cerevisiae*. *Nature*, 440(7084):637–643. [62](#)
- [110] Kufareva, I., Budagyan, L., Raush, E., Totrov, M., and Abagyan, R. (2007). Pier: protein interface recognition for structural proteomics. *Proteins*, 67(2):400–417. [4](#), [12](#), [28](#)

- [111] Laskowski, R. A. (2009). Pdbsum new things. *Nucleic Acids Res*, 37(Database issue):D355–D359. [14](#)
- [112] Leis, S., Schneider, S., and Zacharias, M. (2010). In silico prediction of binding sites on proteins. *Curr Med Chem*, 17(15):1550–1562. [92](#)
- [113] Li, J.-J., Huang, D.-S., Wang, B., and Chen, P. (2006a). Identifying protein-protein interfacial residues in heterocomplexes using residue conservation scores. *Int J Biol Macromol*, 38(3-5):241–247. [4](#), [28](#)
- [114] Li, L., Zhao, B., Cui, Z., Gan, J., Sakharkar, M. K., and Kanguane, P. (2006b). Identification of hot spot residues at protein-protein interface. *Bioinformatics*, 1(4):121–126. [1](#)
- [115] Li, M.-H., Lin, L., Wang, X.-L., and Liu, T. (2007). Protein-protein interaction site prediction based on conditional random fields. *Bioinformatics*, 23(5):597–604. [3](#), [4](#), [28](#)
- [116] Li, N., Sun, Z., and Jiang, F. (2008). Prediction of protein-protein binding site by using core interface residue and support vector machine. *BMC Bioinformatics*, 9:553. [3](#), [4](#), [12](#), [28](#)
- [117] Li, S., Armstrong, C. M., Bertin, N., Ge, H., Milstein, S., Boxem, M., Vidalain, P.-O., Han, J.-D. J., Chesneau, A., Hao, T., Goldberg, D. S., Li, N., Martinez, M., Rual, J.-F., Lamesch, P., Xu, L., Tewari, M., Wong, S. L., Zhang, L. V., Berriz, G. F., Jacotot, L., Vaglio, P., Reboul, J., Hirozane-Kishikawa, T., Li, Q., Gabel, H. W., Elewa, A., Baumgartner, B., Rose, D. J., Yu, H., Bosak, S., Sequerra, R., Fraser, A., Mango, S. E., Saxton, W. M., Strome, S., Heuvel, S. V. D., Piano, F., Vandenhaute, J., Sardet, C., Gerstein, M., Doucette-Stamm, L., Gunsalus, K. C., Harper, J. W., Cusick, M. E., Roth, F. P., Hill, D. E., and Vidal, M. (2004). A map of the interactome network of the metazoan *c. elegans*. *Science*, 303(5657):540–543. [28](#), [62](#)
- [118] Li, X., Wu, M., Kwoh, C.-K., and Ng, S.-K. (2010). Computational approaches for detecting protein complexes from protein interaction networks: a survey. *BMC Genomics*, 11 Suppl 1:S3. [62](#)



- [119] Liang, S., Zhang, C., Liu, S., and Zhou, Y. (2006). Protein binding site prediction using an empirical scoring function. *Nucl. Acids Res.*, 34(13):3698–3707. [5](#), [29](#), [80](#)
- [120] Liang, S., Zhang, J., Zhang, S., and Guo, H. (2004). Prediction of the interaction site on the surface of an isolated protein structure by analysis of side chain energy scores. *Proteins*, 57(3):548–557. [30](#)
- [121] Liu, B., Wang, X., Lin, L., Dong, Q., and Wang, X. (2009a). Exploiting three kinds of interface propensities to identify protein binding sites. *Comput Biol Chem*, 33(4):303–311. [3](#), [4](#), [28](#)
- [122] Liu, B., Wang, X., Lin, L., Tang, B., Dong, Q., and Wang, X. (2009b). Prediction of protein binding sites in protein structures using hidden markov support vector machine. *BMC Bioinformatics*, 10(1):381. [3](#), [4](#), [12](#), [28](#), [62](#)
- [123] Liu, R., Jiang, W., and Zhou, Y. (2010). Identifying protein-protein interaction sites in transient complexes with temperature factor, sequence profile and accessible surface area. *Amino Acids*, 38(1):263–270. [3](#), [4](#), [5](#), [12](#), [29](#), [36](#), [62](#)
- [124] Liu, R. and Zhou, Y. (2009). Using support vector machine combined with post-processing procedure to improve prediction of interface residues in transient complexes. *Protein J*, 28(7-8):369–374. [3](#), [4](#), [5](#), [28](#)
- [125] Ma, B., Elkayam, T., Wolfson, H., and Nussinov, R. (2003). Protein-protein interactions: structurally conserved residues distinguish between binding sites and exposed protein surfaces. *Proc Natl Acad Sci U S A*, 100(10):5772–5777. [63](#)
- [126] Martin, J. (2010). Beauty is in the eye of the beholder: proteins can recognize binding sites of homologous proteins in more than one way. *PLoS Comput Biol*, 6(6):e1000821. [12](#)
- [127] Mehio, W., Kemp, G. J., Taylor, P., and Walkinshaw, M. D. (2010). Identification of protein binding surfaces using surface triplet propensities. *Bioinformatics*. [12](#)

- [128] Melcher, K. (2004). New chemical crosslinking methods for the identification of transient protein-protein interactions with multiprotein complexes. *Curr Protein Pept Sci*, 5(4):287–296. [2](#)
- [129] Mészáros, B., Tompa, P., Simon, I., and Dosztányi, Z. (2007). Molecular principles of the interactions of disordered proteins. *J Mol Biol*, 372(2):549–561. [6](#)
- [130] Mitra, P. (2010). Identifying the nature of the interface in protein-protein complexes. In *ISB '10: Proceedings of the International Symposium on Biocomputing*, pages 1–8, New York, NY, USA. ACM. [4](#), [29](#)
- [131] Monji, H., Koizumi, S., Ozaki, T., and Ohkawa, T. (2011). Interaction site prediction by structural similarity to neighboring clusters in protein-protein interaction networks. *BMC Bioinformatics*, 12 Suppl 1:S39. [3](#)
- [132] Murakami, Y. and Jones, S. (2006). Sharp2: protein-protein interaction predictions using patch analysis. *Bioinformatics*, 22(14):1794–1795. [5](#), [29](#), [31](#), [52](#)
- [133] Murakami, Y. and Mizuguchi, K. (2010). Applying the naive bayes classifier with kernel density estimation to the prediction of protein-protein interaction sites. *Bioinformatics*, 26(15):1841–1848. [3](#), [4](#), [12](#), [62](#)
- [134] Murzin, A. G., Brenner, S. E., Hubbard, T., and Chothia, C. (1995). Scop: a structural classification of proteins database for the investigation of sequences and structures. *J Mol Biol*, 247(4):536–540. [15](#)
- [135] MÃndez, R., Leplae, R., Maria, L. D., and Wodak, S. J. (2003). Assessment of blind predictions of protein-protein interactions: current status of docking methods. *Proteins*, 52(1):51–67. [1](#)
- [136] Naveed, H., Jackups, R., and Liang, J. (2009). Predicting weakly stable regions, oligomerization state, and protein-protein interfaces in transmembrane domains of outer membrane proteins. *Proc Natl Acad Sci U S A*, 106(31):12735–12740. [62](#)

- [137] Negi, S. S. and Braun, W. (2007). Statistical analysis of physical-chemical properties and prediction of protein-protein interfaces. *J Mol Model*, 13(11):1157–1167. [30](#)
- [138] Neuvirth, H., Heinemann, U., Birnbaum, D., Tishby, N., and Schreiber, G. (2007). Promateus—an open research approach to protein-binding sites analysis. *Nucleic Acids Res*, 35(Web Server issue):W543–W548. [3](#), [29](#)
- [139] Neuvirth, H., Raz, R., and Schreiber, G. (2004). Promate: A structure based prediction program to identify the location of protein-protein binding sites. *Journal of Molecular Biology*, 338(1):181–199. [1](#), [29](#), [36](#), [62](#), [80](#)
- [140] Ng, A. Y. and Jordan, M. I. (2002). On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In *Advances in Neural Information Processing Systems (NIPS)*. [36](#)
- [141] Nooren, I. M. A. and Thornton, J. M. (2003). Diversity of protein-protein interactions. *EMBO J*, 22(14):3486–3492. [6](#)
- [142] Nugent, T. and Jones, D. T. (2011). Membrane protein structural bioinformatics. *J Struct Biol*. [5](#)
- [143] Nussinov, R. (2009). *Computational protein-protein interactions*. CRC. [62](#)
- [144] Ofran, Y. and Rost, B. (2003a). Analysing six types of protein-protein interfaces. *J Mol Biol*, 325(2):377–387. [12](#), [16](#)
- [145] Ofran, Y. and Rost, B. (2003b). Predicted protein-protein interaction sites from local sequence information. *FEBS Lett*, 544(1-3):236–239. [3](#), [4](#), [12](#), [28](#)
- [146] Ofran, Y. and Rost, B. (2007). Isis: interaction sites identified from sequence. *Bioinformatics*, 23(2):e13–e16. [62](#)
- [147] Oh, M., Joo, K., and Lee, J. (2009). Protein-binding site prediction based on three-dimensional protein modeling. *Proteins*, 77 Suppl 9:152–156. [4](#), [12](#)

- [148] Ostermeier, C. and Michel, H. (1997). Crystallization of membrane proteins. *Curr Opin Struct Biol*, 7(5):697–701. [1](#)
- [149] Ozbabacan, S. E. A., Engin, H. B., GURSOY, A., and Keskin, O. (2011). Transient protein-protein interactions. *Protein Eng Des Sel*, 24(9):635–648. [6](#)
- [150] Park, S. H., Reyes, J. A., Gilbert, D. R., Kim, J. W., and Kim, S. (2009). Prediction of protein-protein interaction types using association rule based classification. *BMC Bioinformatics*, 10:36. [1](#)
- [151] Perkins, J. R., Diboun, I., Dessailly, B. H., Lees, J. G., and Orengo, C. (2010). Transient protein-protein interactions: structural, functional, and network properties. *Structure*, 18(10):1233–1243. [6](#)
- [152] Petsko, G. and Ringe, D. (2004). *Protein structure and function*. Sinauer Associates Inc. [2](#)
- [153] Pettit, F. K., Bare, E., Tsai, A., and Bowie, J. U. (2007). Hotpatch: a statistical approach to finding biologically relevant features on protein surfaces. *J Mol Biol*, 369(3):863–879. [3](#), [29](#), [39](#)
- [154] Platt, J. C. (1999). *Advances in kernel methods: support vector learning*, chapter Fast training of support vector machines using sequential minimal optimization, pages 185–208. MIT Press. [36](#)
- [155] Porollo, A. and Meller, J. (2007). Prediction-based fingerprints of protein-protein interactions. *Proteins: Structure, Function, and Bioinformatics*, 66(3):630–645. [3](#), [4](#), [12](#), [28](#), [36](#), [47](#), [49](#), [62](#)
- [156] Poupon, A. (2004). Voronoi and voronoi-related tessellations in studies of protein structure and interaction. *Curr Opin Struct Biol*, 14(2):233–241. [16](#)
- [157] Puton, T., Kozlowski, L., Tuszynska, I., Rother, K., and Bujnicki, J. M. (2011). Computational methods for prediction of protein-rna interactions. *J Struct Biol*. [92](#)

- [158] Qin, S. and Zhou, H.-X. (2007). Meta-ppisp: a meta web server for protein-protein interaction site prediction. *Bioinformatics*, 23(24):3386–3387. [3](#), [4](#), [12](#), [28](#), [80](#)
- [159] Qiu, Z. and Wang, X. (2011). Prediction of protein-protein interaction sites using patch-based residue characterization. *J Theor Biol*. [3](#)
- [160] Rappsilber, J., Siniosoglou, S., Hurt, E. C., and Mann, M. (2000). A generic strategy to analyze the spatial organization of multi-protein complexes by cross-linking and mass spectrometry. *Anal Chem*, 72(2):267–275. [2](#)
- [161] Reddy, B. V. B. and Kaznessis, Y. N. (2005). A quantitative analysis of interfacial amino acid conservation in protein-protein hetero complexes. *J Bioinform Comput Biol*, 3(5):1137–1150. [12](#)
- [162] Reynolds, C., Damerell, D., and Jones, S. (2009). Protorp: a protein-protein interaction analysis server. *Bioinformatics*, 25(3):413–414. [14](#)
- [163] Rossi, A., Marti-Renom, M. A., and Sali, A. (2006). Localization of binding sites in protein structures by optimization of a composite scoring function. *Protein Sci*, 15(10):2366–2380. [29](#), [62](#)
- [164] Royer, L., Reimann, M., Andreopoulos, B., and Schroeder, M. (2008). Unraveling protein networks with power graph analysis. *PLoS Comput Biol*, 4(7):e1000108. [1](#)
- [165] Sacquin-Mora, S., Carbone, A., and Lavery, R. (2008). Identification of protein interaction partners and protein-protein interaction sites. *J Mol Biol*, 382(5):1276–1289. [1](#), [4](#), [28](#)
- [166] Saha, R. P., Bahadur, R. P., Pal, A., Mandal, S., and Chakrabarti, P. (2006). Proface: a server for the analysis of the physicochemical features of protein-protein interfaces. *BMC Struct Biol*, 6:11. [14](#)
- [167] Schmitt, S., Kuhn, D., and Klebe, G. (2002). A new method to detect related function among proteins independent of sequence and fold homology. *J Mol Biol*, 323(2):387–406. [93](#)

- [168] Sen, T. Z., Kloczkowski, A., Jernigan, R. L., Yan, C., Honavar, V., Ho, K.-M., Wang, C.-Z., Ihm, Y., Cao, H., Gu, X., and Dobbs, D. (2004). Predicting binding sites of hydrolase-inhibitor complexes by combining several methods. *BMC Bioinformatics*, 5:205. [3](#)
- [169] Shoemaker, B. A. and Panchenko, A. R. (2007). Deciphering protein-protein interactions. part i. experimental techniques and databases. *PLoS Comput Biol*, 3(3):e42. [28](#)
- [170] Shoemaker, B. A., Panchenko, A. R., and Bryant, S. H. (2006). Finding biologically relevant protein domain interactions: conserved binding mode analysis. *Protein Sci*, 15(2):352–361. [1](#)
- [171] Shoemaker, B. A., Zhang, D., Thangudu, R. R., Tyagi, M., Fong, J. H., Marchler-Bauer, A., Bryant, S. H., Madej, T., and Panchenko, A. R. (2010). Inferred biomolecular interaction server—a web server to analyze and predict protein interacting partners and binding sites. *Nucleic Acids Res*, 38(Database issue):D518–D524. [3](#), [62](#)
- [172] Sikić, M., Tomić, S., and Vlahovicek, K. (2009). Prediction of protein-protein interaction sites in sequences and 3d structures by random forests. *PLoS Comput Biol*, 5(1):e1000278. [3](#), [4](#), [5](#), [12](#), [28](#), [62](#)
- [173] Sinha, R., Kundrotas, P. J., and Vakser, I. A. (2010). Docking by structural similarity at protein-protein interfaces. *Proteins*. [1](#)
- [174] Sinz, A. (2003). Chemical cross-linking and mass spectrometry for mapping three-dimensional structures of proteins and protein complexes. *J Mass Spectrom*, 38(12):1225–1237. [2](#)
- [175] Stein, A., Céol, A., and Aloy, P. (2011). 3did: identification and classification of domain-based interactions of known three-dimensional structure. *Nucleic Acids Res*, 39(Database issue):D718–D723. [15](#)
- [176] Stelzl, U., Worm, U., Lalowski, M., Haenig, C., Brembeck, F. H., Goehler, H., Stroedicke, M., Zenkner, M., Schoenherr, A., Koeppen, S., Timm, J., Mintzlaff, S., Abraham, C., Bock, N., Kietzmann, S., Goedde, A., Toksöz, E., Droege, A., Krobitsch, S., Korn, B., Birchmeier,

- W., Lehrach, H., and Wanker, E. E. (2005). A human protein-protein interaction network: a resource for annotating the proteome. *Cell*, 122(6):957–968. [28](#)
- [177] SzilÁgyi, A., Grimm, V., Arakaki, A. K., and Skolnick, J. (2005). Prediction of physical protein-protein interactions. *Phys Biol*, 2(2):S1–16. [6](#)
- [178] Teyra, J., Doms, A., Schroeder, M., and Pisabarro, M. T. (2006). Scowlp: a web-based database for detailed characterization and visualization of protein interfaces. *BMC Bioinformatics*, 7:104. [15](#)
- [179] Tsai, C. J., Lin, S. L., Wolfson, H. J., and Nussinov, R. (1996a). A dataset of protein-protein interfaces generated with a sequence-order-independent comparison technique. *J Mol Biol*, 260(4):604–620. [16](#)
- [180] Tsai, C. J., Lin, S. L., Wolfson, H. J., and Nussinov, R. (1996b). Protein-protein interfaces: architectures and interactions in protein-protein interfaces and in protein cores. their similarities and differences. *Crit Rev Biochem Mol Biol*, 31(2):127–152. [4](#), [12](#), [29](#)
- [181] Tsai, C. J., Xu, D., and Nussinov, R. (1997). Structural motifs at protein-protein interfaces: protein cores versus two-state and three-state model complexes. *Protein Sci*, 6(9):1793–1805. [4](#), [12](#), [29](#)
- [182] Tuncbag, N., Gursoy, A., Guney, E., Nussinov, R., and Keskin, O. (2008). Architectures and functional coverage of protein-protein interfaces. *J Mol Biol*, 381(3):785–802. [5](#), [63](#)
- [183] Tuncbag, N., Kar, G., Keskin, O., Gursoy, A., and Nussinov, R. (2009). A survey of available tools and web servers for analysis of protein-protein interactions and interfaces. *Brief Bioinform*, 10(3):217–232. [3](#), [4](#), [29](#), [62](#)
- [184] Tuncbag, N., Keskin, O., and Gursoy, A. (2010). Hotpoint: hot spot prediction server for protein interfaces. *Nucleic Acids Res*. [1](#)
- [185] Tyagi, M., Shoemaker, B. A., Bryant, S. H., and Panchenko, A. R. (2009). Exploring functional roles of multibinding protein interfaces. *Protein Sci*, 18(8):1674–1683. [6](#)

- [186] Uetz, P., Giot, L., Cagney, G., Mansfield, T. A., Judson, R. S., Knight, J. R., Lockshon, D., Narayan, V., Srinivasan, M., Pochart, P., Qureshi-Emili, A., Li, Y., Godwin, B., Conover, D., Kalbfleisch, T., Vijayadamodar, G., Yang, M., Johnston, M., Fields, S., and Rothberg, J. M. (2000). A comprehensive analysis of protein-protein interactions in *saccharomyces cerevisiae*. *Nature*, 403(6770):623–627. [28](#)
- [187] Valdar, W. S. and Thornton, J. M. (2001). Protein-protein interfaces: analysis of amino acid conservation in homodimers. *Proteins*, 42(1):108–124. [12](#)
- [188] Vapnik, V. (2000). *The Nature of Statistical Learning Theory, 2nd Edition*. Springer-Verlag. [36](#)
- [189] Veselovsky, A. V., Ivanov, Y. D., Ivanov, A. S., Archakov, A. I., Lewi, P., and Janssen, P. (2002). Protein-protein interactions: mechanisms and modification by drugs. *J Mol Recognit*, 15(6):405–422. [1](#)
- [190] Wang, B., Chen, P., Huang, D.-S., jing Li, J., Lok, T.-M., and Lyu, M. R. (2006a). Predicting protein interaction sites from residue spatial sequence profile and evolution rate. *FEBS Lett*, 580(2):380–384. [3](#), [4](#), [28](#)
- [191] Wang, B., Ge, L. S., Huang, D.-S., and Wong, H. S. (2007a). Prediction of protein-protein interacting sites by combining svm algorithm with bayesian method. In *Proc. Third International Conference on Natural Computation ICNC 2007*, volume 2, pages 329–333. [3](#), [4](#), [28](#)
- [192] Wang, B., Wong, H. S., and Huang, D.-S. (2006b). Inferring protein-protein interacting sites using residue conservation and evolutionary information. *Protein Pept Lett*, 13(10):999–1005. [3](#)
- [193] Wang, C., Cheng, J., Su, S., and Xu, D. (2008). Identification of interface residues involved in protein-protein interactions using naïve bayes classifier. In *ADMA '08: Proceedings of the 4th international conference on Advanced Data Mining and Applications*, pages 207–216, Berlin, Heidelberg. Springer-Verlag. [3](#), [12](#)



- [194] Wang, G. and Dunbrack, R. L. (2003). Pisces: a protein sequence culling server. *Bioinformatics*, 19(12):1589–1591. [19](#), [20](#), [32](#)
- [195] Wang, H., Segal, E., Ben-Hur, A., Li, Q.-R., Vidal, M., and Koller, D. (2007b). Insite: a computational method for identifying protein-protein interaction binding sites on a proteome-wide scale. *Genome Biol*, 8(9):R192. [93](#)
- [196] Wells, J. A. (1991). Systematic mutational analyses of protein-protein interfaces. *Methods Enzymol*, 202:390–411. [2](#)
- [197] Wells, J. A. and McClendon, C. L. (2007). Reaching for high-hanging fruit in drug discovery at protein-protein interfaces. *Nature*, 450(7172):1001–1009. [1](#)
- [198] Weskamp, N., Kuhn, D., HÄijllermeier, E., and Klebe, G. (2004). Efficient similarity search in protein structure databases by k-clique hashing. *Bioinformatics*, 20(10):1522–1526. [94](#)
- [199] Winter, C., Henschel, A., Kim, W. K., and Schroeder, M. (2006). Scoppi: a structural classification of protein-protein interfaces. *Nucleic Acids Res*, 34(Database issue):D310–D314. [15](#)
- [200] Wu, F., Olson, B., Dobbs, D., and Honavar, V. (2006). Comparing kernels for predicting protein binding sites from amino acid sequence. In *IEEE Joint Conference on Neural Networks*, Vancouver, Canada. IEEE Press. [3](#), [4](#), [12](#), [28](#), [62](#)
- [201] Wu, F., Towfic, F., Dobbs, D., and Honavar, V. (2007). Analysis of protein protein dimeric interfaces. In *IEEE International Conference on Bioinformatics and Biomedicine*. [4](#), [12](#), [29](#)
- [202] Xue, L. C., Dobbs, D., and Honavar, V. (2011a). Homppi: A class of sequence homology based protein-protein interface prediction methods. *BMC Bioinformatics*, 12(1):244. [3](#), [4](#), [5](#), [12](#), [13](#), [24](#), [90](#), [93](#)
- [203] Xue, L. C., Jordan, R. A., EL-Manzalawy, Y., Dobbs, D., and Honavar, V. (2011b). Ranking docked models of protein-protein complexes using predicted partner-specific protein-protein interfaces: A preliminary study. In *In Proceedings of the International Conference*

*On Bioinformatics and Computational Biology (ACM-BCB); Chicago, Illinois, August 1-3.*

14

- [204] Yan, C., Dobbs, D., and Honavar, V. (2003). Identification of surface residues involved in protein-protein interaction - a support vector machine approach. In *Proceedings of the Conference on Intelligent Systems Design and Applications (ISDA-03)*. 3, 4, 28
- [205] Yan, C., Dobbs, D., and Honavar, V. (2004a). A two-stage classifier for identification of protein-protein interface residues. *Bioinformatics*, 20 Suppl 1:i371–i378. 3, 4, 12, 28, 36, 62
- [206] Yan, C., Honavar, V., and Dobbs, D. (2004b). Identification of interface residues in protease-inhibitor and antigen-antibody complexes: a support vector machine approach. *Neural Computing & Applications*, 13(2):123–129. 3, 4, 28
- [207] Yan, C., Wu, F., Jernigan, R. L., Dobbs, D., and Honavar, V. (2008). Characterization of protein-protein interfaces. *Protein J*, 27(1):59–70. 4, 12, 29
- [208] Yip, K. Y., Kim, P. M., McDermott, D., and Gerstein, M. (2009). Multi-level learning: improving the prediction of protein, domain and residue interactions by allowing information flow between levels. *BMC Bioinformatics*, 10:241. 1
- [209] Yu, J. and Fotouhi, F. (2006). Computational approaches for predicting protein-protein interactions: a survey. *J Med Syst*, 30(1):39–44. 62
- [210] Zellner, H., Staudigel, M., Trenner, T., Bittkowski, M., Wolowski, V., Icking, C., and Merkl, R. (2011). Prescont: Predicting protein-protein interfaces utilizing four residue properties. *Proteins*. 3, 4
- [211] Zhang, Q. C., Deng, L., Fisher, M., Guan, J., Honig, B., and Petrey, D. (2011). Predus: a web server for predicting protein interfaces using structural neighbors. *Nucleic Acids Res*, Vol. 39:Web Server issue W283–W287. 4, 5, 12, 63, 64, 75, 76
- [212] Zhang, Q. C., Petrey, D., Norel, R., and Honig, B. H. (2010). Protein interface conservation across structure space. *Proc Natl Acad Sci U S A*, 107(24):10896–10901. 3, 5, 12, 63, 64, 70, 71, 75

- [213] Zhou, H.-X. and Qin, S. (2007). Interaction-site prediction for protein complexes: a critical assessment. *Bioinformatics*, 23:3386–3387. [3](#), [7](#), [12](#), [29](#), [62](#), [80](#)
- [214] Zhou, H. X. and Shan, Y. (2001). Prediction of protein interaction sites from sequence profile and residue neighbor list. *Proteins*, 44(3):336–343. [3](#), [4](#), [28](#)
- [215] Zinzalla, G. and Thurston, D. (2009). Targeting protein–protein interactions for therapeutic intervention: a challenge for the future. *Future*, 1(1):65–93. [1](#)
- [216] Zuiderweg, E. R. P. (2002). Mapping protein-protein interactions in solution by nmr spectroscopy. *Biochemistry*, 41(1):1–7. [1](#), [2](#)