Computational and Structural Biotechnology Journal 19 (2021) 2269–2278

COMPUTATIONAL AND STRUCTURAL BIOTECHNOLOGY JOURNAL

journal homepage: www.elsevier.com/locate/csbj

Improved cluster ranking in protein–protein docking using a regression approach

Shahabeddin Sotudian [a,1], Israel T. Desta [b,1], Nasser Hashemi [a], Shahrooz Zarbafian [a], Dima Kozakov [c], Pirooz Vakili [a], Sandor Vajda [b,e], Ioannis Ch. Paschalidis [a,b,d,*]

[a] Division of Systems Engineering, Boston University, Boston, USA
[b] Department of Biomedical Engineering, Boston University
[c] Laufer Center for Physical and Quantitative Biology, Institute for Advanced Computational Sciences, Stony Brook University, Stony Brook, USA
[d] Department of Electrical & Computer Engineering, and Faculty for Computing & Data Sciences, Boston University
[e] Department of Chemistry, Boston University

ARTICLE INFO

Article history:
Received 31 January 2021
Received in revised form 8 April 2021
Accepted 9 April 2021
Available online 20 April 2021

Keywords:
Protein docking
Machine learning
Ranking

ABSTRACT

We develop a *Regression-based Ranking by Pairwise Cluster Comparisons (RRPCC)* method to rank clusters of similar protein complex conformations generated by an underlying docking program. The method leverages robust regression to predict the relative quality difference between any pair or clusters and combines these pairwise assessments to form a ranked list of clusters, from higher to lower quality. We apply RRPCC to clusters produced by the automated docking server ClusPro and, depending on the training/validation strategy, we show improvement by 24–100% in ranking acceptable or better quality clusters first, and by 15–100% in ranking medium or better quality clusters first. We compare the RRPCC–ClusPro combination to a number of alternatives, and show that very different machine learning approaches to scoring docked structures yield similar success rates. Finally, we discuss the current limitations on sampling and scoring, looking ahead to further improvements. Interestingly, some features important for improved scoring are internal energy terms that occur only due to the local energy minimization applied in the refinement stage following rigid body docking.

1. Introduction

Besides comprising the largest portion of the dry mass in each cell, proteins interact with one or multiple other proteins to organize, regulate and drive cellular functions and communication [1]. Protein–protein interactions govern activities ranging from degradation of specific proteins [2] to complex events such as cell proliferation [3]. Due to their importance, elucidating such interactions up to atomic detail is necessary for understanding large multicomponent complexes like ribosomes and discovering protein-based drugs (antibodies, nanobodies, peptides, etc.). While the current golden standard for determining the structure of protein complexes is X-ray crystallography, crystallizing protein complexes is extremely time consuming. The latest experimental methods, such as cryo-EM, are mostly financially restrictive and unfeasible when considering that there are tens of thousands of interactions that are yet to be resolved.

Computational *docking* offers an alternative approach to predicting the structures of protein–protein complexes based on the structures of the interacting component proteins. While the computational approach is generally less reliable than X-ray crystallography, it generates models that can be validated by simpler experimental techniques such as cross-linking or site-directed mutagenesis. The Vajda lab at Boston University has developed ClusPro, one of the most popular protein–protein docking servers currently available [4]. Although registration is not required, ClusPro has 17,000 registered users, and in 2020 has performed over 180,000 docking calculations. The server uses the rigid body docking program PIPER based on the fast Fourier transform (FFT) approach to evaluating interaction energies. With PIPER, the larger protein is placed at the origin of the coordinate system on a fixed grid, the second protein is placed on a movable grid, and the interaction energy is written as a linear combination of a few correlation functions. These functions describe the repulsive and attractive contributions to the van der Waals interaction energy,

* Corresponding author.
E-mail address: yannisp@bu.edu (I.Ch. Paschalidis).
[1] These authors contributed equally to this work.

https://doi.org/10.1016/j.csbj.2021.04.028

the electrostatic interaction energy, and a pairwise structure-based potential constructed by the Decoys as the Reference State (DARS) approach [5], representing solvation effects. The weights of the energy terms are adjustable parameters and will be discussed further in the sequel. Using FFTs, the energy function can be evaluated for all translations in a single calculation, and only rotations need to be considered explicitly. Due to the extremely fast sampling, one can explore billions of the conformations of the two interacting proteins, and thus perform global docking without any a priori information on the structure of the complex.

Although PIPER uses an FFT-based approach, which makes it similar to a number of other FFT-based docking programs, including FTDOCK [6], ZDOCK [7], GRAMM-X [8], and HDOCK [9], the algorithm powering ClusPro has two main differences from these programs. First, PIPER is the first FFT-based method that uses a pairwise structure-based potential called DARS [5], which is converted by eigenvalue analysis to the correlation form required for the FFT approach. The use of the DARS potential substantially improved the performance of the method. Second, while most docking methods select the lowest energy structures as the best models, in ClusPro we consider the centers of the most populated clusters of the docked structure as predictions of the complex. This approach enables us to account for the role of entropy in the recognition process, leading to more reliable predictions than methods using energy values alone. Indeed, we have shown in a recent paper [10] that using cluster centers rather than the lowest energy structures improves the accuracy of docking. In this paper we also compare the performance of ClusPro to that of the two most successful docking servers, HADDOCK and SwarmDock [11,12]. We note that the performance of servers is continuously compared in the ongoing CAPRI (Critical Assessment of Predicted Interactions) docking experiments. ClusPro and most of the other popular servers all participate in CAPRI since 2009, demonstrating their relative performance.

It is generally recognized that the protein–protein interfaces in complexes are packed essentially as well as the interiors of proteins, possibly with more polar atoms but without enclosed cavities and definitely without any steric clashes [13]. Rigid body docking methods are based on the assumption that the structures of proteins do not change upon their association. In reality some conformational changes always occur, and hence rigid body methods must use "soft" energy functions that allow for moderate clashes without increasing the calculated value of the interaction energy. Reducing the sensitivity of the energy function to steric clashes implies that the near-native docked structures do not necessarily have the lowest energies, and hence one has to retain a large number of conformations for further processing. Attempts to identify near-native structures among the retained structures may include re-scoring using more accurate energy functions. The more accurate energy evaluation frequently requires refinement of the structures using energy minimization, thus beyond the scope of rigid body docking. Since energy minimization is computationally expensive, many docking methods use clustering for reducing the number of structures prior to refinement.

In ClusPro, we retain 1000 structures generated by the rigid body docking. The unique feature of our server is that the selection of clusters that are most likely near-native is based on the size of the resulting clusters rather than their energy values. As has been shown, this approach is meaningful if we assume that the energy range of the retained 1000 lowest energy conformations is comparable to the error in the calculation of the energy, and hence does not allow for further discrimination between near-native and non-native structures [14]. It is reasonable to assume that ranking the clusters based on the number of structures is not the best approach to cluster discrimination, and that there must exist empirical energy functions to improve the scoring. However, we have shown that ranking clusters based on population yields better models than based on PIPER energies, in spite of the success of the latter in guiding the search in the process of docking [10]. We made considerable effort to further optimize the potential for specific classes of protein–protein complexes. In particular, we have developed an asymmetric version of the pairwise DARS potential for docking and scoring antibody-antigen pairs, thereby improving the results for this type of interactions [15]. However, we failed to obtain substantial improvement for the general case of complexes categorized as "others" [11], and even for enzyme-inhibitor pairs.

In this paper we describe the use of a *Machine Learning (ML)*-based method to improve the ranking of clusters generated by ClusPro, and show that the approach substantially increases the number of proteins for which the top prediction, denoted as T_1, is of acceptable or better quality (see Methods for the definition of quality measures). ML methods have previously been used in protein–protein docking, both for ranking the clusters and for improving the scoring of models. The Bates group [16] employed a randomized tree classifier (an ensemble of randomly constructed decision trees based on [17]) with 109 molecular descriptors to rank the clusters of docked structures. The descriptors included statistics of cluster properties, which were then combined into a pairwise cluster comparison model to discriminate near-native from incorrect clusters. The results have shown improved discrimination, but were provided only for a few protein complexes. As will be shown, we adopted some ideas from this paper, but in the context of a very different ML method. In other papers the ML approach has been used to directly score docked structures rather than clusters. An influential scoring method called IRaPPA (Integrative Ranking of Protein–Protein Assemblies) [12] has been developed by Fernadez-Recio and co-workers. The method used support vector machines to combine a large selection of metrics, including biophysical models, statistical potentials and composite energy functions. IRaPPA has been used to re-score structures generated by several docking methods, and showed almost a doubling of acceptable solutions in the top 10 models for SWARMDOCK and 30% to 75% improvement on pyDock, SDOCK and ZDOCK. While the increase in performance is impressive, [12] uses on the order of 90 different features, incurring a significant computational cost. Basu and Wallner [18] also used a support vector machine algorithm for scoring complex models using a potential function they called ProQDock. They were able to reduce the number of features to 12. ProQDock was combined with the traditional scoring functions ZRANK [19] and ZRANK2 [20], to form ProQDockZ with improved performance. More recently, Eismann et al. [21] reported the use of a neural network-based method called PAUL for selecting models of protein complexes. The network architecture combined multiple ingredients that together enabled end-to-end learning from molecular structures using a point-based representation of atoms, equivariance with respect to rotation and translation, local convolutions, and hierarchical subsampling operations. The method has been combined with previously developed scoring functions, and improved the identification of accurate structural models.

The method we have developed to rank the clusters generated by ClusPro is based on a robust regression approach, not used in the above applications. Some features depended on the statistical properties of the clusters, and others were determined by the conformation selected as the cluster representative. The proposed method, called *Regression-based Ranking by Pairwise Cluster Comparisons (RRPCC)*, performs pairwise comparisons of clusters, an approach also used by Bates and co-workers [16]. Learning through pairwise comparisons sidesteps challenges associated with having to learn from few samples based on an imbalanced dataset. In particular, more direct methods would attempt to predict the quality of a cluster based on the features. However, for any given complex,

the number of clusters is relatively small (about 50) and most of them are of low-quality. Instead, for such a complex, one can generate a balanced dataset of about 2,500 pairwise comparisons. As will be described, we trained RRPCC on the complexes in version 5.0 of the protein–protein benchmark (abbreviated as BM5) [11]. The method's performance was evaluated by firstly training it on 2 subsets of protein complexes after splitting BM5 randomly into three and tested on the third subset (3-fold cross-validation). Secondly, it was trained using a subset of the complexes (training set) and tested on a separate subset of 51 protein pairs that were added to version 4.0 of the Benchmark (BM4) to form BM5 (historical). The importance of this selection is that the ML based scoring methods IRaPPA [12], ProQDock, ProQDockZ, PAUL, and RRPCC were all applied to the same test set, providing information on the performance of scoring based on SVMs, neural networks, and regression methods. As will be discussed, the interesting result is that all methods yield very similar success rates, in spite of the very different ML approaches and the different programs used for generating the docked structures to be scored. Although the scoring always improves the ranking of near-native structures among the many structures generated by the docking, the success rate remains slightly below 50%, even when considering the top 10 models. As will be shown, in the majority of cases when the modeling of the complex fails, the culprit is the insufficient sampling. However, good conformations are also lost when selecting the final models. Although in this paper we focus on the development and characterization of a specific protocol to improve the scoring of clusters of structures generated by ClusPro, we hope to also answer some general questions related to the properties of ML-based approaches used for improving the discrimination of near-native structures.

2. Materials and Methods

Notation: We use boldfaced lowercase letters to denote vectors, ordinary lowercase letters to denote scalars, boldfaced uppercase letters to denote matrices, and calligraphic capital letters to denote sets. All vectors are column vectors. For space saving reasons, we write $\mathbf{x} = (x_1, \ldots, x_n)$ to denote the column vector $\mathbf{x} \in \mathbb{R}^n$. For any matrix $\mathbf{A}$, we let a_{ij} denote its (i,j) element, $\mathbf{A}_i$ the ith row and, with some abuse of our conventions, $\mathbf{A}_j$ the jth column. $\mathbf{I}$ denotes the identity matrix. We use prime to denote the transpose of a vector, and $\|\mathbf{x}\|_p = (\sum_{i=1}^{n} |x_i|^p)^{1/p}$ for the ℓ_p norm, where $p \geqslant 1$. For any matrix $\mathbf{A} \in \mathbb{R}^{n \times m}$, we will refer to a vector $(a_{ij};\ i = 1, \ldots, n,\ j = 1, \ldots, m)$ containing all entries of the matrix as the vectorized form and by $\|\mathbf{A}\|_p$ we will denote the ℓ_p norm of that vector, i.e., $\|\mathbf{A}\|_p = (\sum_{i=1}^{n} \sum_{j=1}^{m} |a_{ij}|^p)^{1/p}$.

2.1. Scoring Functions

A collection of physicochemical scoring functions and statistical potentials were used to generate 129 features we examined to derive our cluster ranking model. We leveraged the docking program PIPER [22,14] which calculates van der Waals (repulsive and attractive) interactions, electrostatic (Coulombic and Born) forces, and a statistical potential called DARS [5]. PIPER uses a linear combination of these five scores to compute a composite score and select the 1,000 lowest scoring models. These conformations are then clustered using the all-atom pairwise *Root Mean Squared Deviation (RMSD)* as the distance metric, generating clusters with a radius of up to 10 Å RMSD and a minimum number of 10 structures per cluster. From each cluster, a *representative* is selected to be the *cluster center* – the conformation with the largest number of neighbors within the cluster. Two conformations are considered neighbors if their iRMSD – the RMSD of backbone atoms on the interface between the receptor and ligand among the two conformations – is no more than 10 Å.

ClusPro features. From each cluster we use its cardinality (which is the only feature ClusPro uses for ranking), the total ClusPro composite score and individual energy terms for the cluster representative, statistics (mean, variance, skewness, kurtosis) of these energy metrics over all cluster members, and statistics of RMSD distances between cluster members.

SOAP features. A second statistical potential named *Statistically Optimized Atomic Potential (SOAP)* [23] of each cluster representative was obtained. SOAP was calculated with MODELLER release 9.21 developed by Šali and coworkers. Three different SOAP potentials were considered: (*i*) a pairwise score of atomic interactions within the protein–protein interface; (*ii*) an atom-specific score of solvent accessibility for all atoms; and (*iii*) a score using (*i*) and (*ii*) to obtain an assessment score for the target model.

ROSETTA features. The conformation of the cluster representative structures was also evaluated using ROSETTA release version 3.11 – a suite for computational modeling and analysis of protein structures [34]. We calculated energy terms that included: (*i*) interaction potentials between bonded and non-bonded atoms; (*ii*) intramolecular terms that involved bonded atoms, such as intra-residue *Lennard-Jones (LJ)* potentials between atoms, the likelihood of backbone angles, and penalties for unlikely atomic arrangements [24,25]; and (*iii*) terms that play significant role on interface formation, such as intra-residue attractive and repulsive LJ potential between atoms, solvation energy, electrostatic potential, and hydrogen bonding.

CHARMM features. For each cluster representative we used CHARMM22 [26] to extract features related to bond stretches, bond angles, torsion angles, and improper angles (out of plane bending) contributions, as well as non-bonded interaction terms like two-atom and three-atom Van der Waals energies calculated with a LJ potential.

Stability features. A final set of features is related to the stability of the cluster representatives. To that end, we started from the cluster representative and separated the two proteins. An in–house program (libmol2), based on ideas from [27,28], was used to perform 10 steps of limited memory Broyden-Fletcher-Goldfarb-Shanno (LBFGS) minimization with the alpha carbons fixed to remove potential clashes at the interface. [2] Both ROSETTA and CHARMM22 energy terms were calculated before and after performing such minimization. The CHARMM22 parameter set is the extended version of the CHARMM19 united atom parameter set, which uses only polar hydrogens, whereas the nonpolar hydrogens are represented by the increased radii of the nonpolar group. This type of parameterization reduces the total number of atoms by almost 50% with very little impact on accuracy, particularly because we use the minimization of the CHARMM potential only for the removal of steric clashes rather than structure prediction. Since the computing time changes with the square of the number of atoms, the united atom representation leads to major reduction of the computing time. In fact, minimization of the large number of structures generated by the docking would not be possible using an all-hydrogen potential. After evaluating the ROSETTA energy terms of the minimized structure, the ligand was moved 100 Å away from original position near the receptor. After obtaining the CHARMM22 energy terms of the separated structure, it was minimized in the same way as before using libmol2. Finally, the CHARMM22 and ROSETTA energy terms of the separated and minimized structure were obtained. A complete list and brief description of all features is in supplementary Table S4.

[2] Code is available upon request.

2.2. Evaluation Criteria and DockQ

CAPRI, the community-wide competition for protein docking, uses three conventional parameters to assess submitted conformations: (*i*) Fnat (fraction of the number of true interface residue contacts accurately predicted to that of falsely predicted ones), (*ii*) iRMSD (RMSD of backbone atoms on the interface when the model is superposed on the native), and (*iii*) LRMSD (RMSD between ligand backbone atoms when the receptors are aligned). Based on these parameters, there are four categories of accuracy. A prediction is considered *high quality* if Fnat $\geqslant$0.5 and LRMSD $\leqslant$1.0 Å or iRMSD $\leqslant$1.0 Å. It is of *medium quality* if 0.3 $\leqslant$Fnat $<$ 0.5, and LRMSD $\leqslant$5.0 Å or iRMSD $\leqslant$2.0 Å, or if Fnat $\geqslant$0.5 but LRMSD $>$ 1.0 Å and iRMSD $>$ 1.0 Å. It is of *acceptable quality* if 0.1 $\leqslant$Fnat $<$ 0.3 and LRMS $\leqslant$10.0 Å or iRMSD $\leqslant$4.0 Å, or if Fnat $\geqslant$0.3 but LRMSD $>$ 5.0 Å and iRMSD $>$ 2.0 Å. Finally, the prediction is *incorrect* if none of the above categories apply. DockQ [29] is a program [3] which combines the three metrics non-linearly to produce a score and categorizes models as acceptable if that score is $\geqslant$0.23, as medium quality with values $\geqslant$0.49, and as high quality with values $\geqslant$0.8. [29] verified that DockQ matched with previous CAPRI experiments well.

2.3. Benchmark and Decoy Generation

We consider here 38 antibody-antigen (ab-ag) complexes, 88 enzyme-containing complexes, and 101 others-containing complexes from version 5.0 of the well-established protein docking benchmark (BM5) [11]. Note that two targets of the 40 ab-ag complexes, and one of the others-containing complexes are excluded because DockQ failed to evaluate the quality of the models due to the large size of the complexes. Since these three classes of proteins fundamentally differ in how they interact with their complementary proteins and in their respective energetics, the models were trained and tested separately. Two *validation strategies* were used to test the algorithm. First, each class of complexes was split into 3 approximately equal sets. In a process called *3-fold cross-validation*, the algorithm was trained on two of the folds and tested on the remaining fold; this was repeated three times, each leaving out a different fold for testing. In a second validation strategy, which we call *historical*, the algorithm was trained on Benchmark 4.0 (BM4) [30] and tested on the 51 new cases added onto BM4 to form BM5 (of the 55 cases, two failed in both SWARMDOCK and ClusPro, and two more failed in SWARMDOCK due to their size).

2.4. Enriching Starting Poses

Due to the imbalance in terms of extremely low number of good (acceptable or better) models compared to incorrect models, we sought to enrich the number of good models in the top 30 clusters retained from PIPER. As mentioned in Section 2.1, PIPER was used to provide 1,000 models for a specific composite score that is a linear combination of five component scores. Typically, PIPER is run using a fixed composite score determined by a set of five coefficients that multiply the component scores. For our purposes, however, we run PIPER using 50 different coefficient sets, each set resulting into 30 clusters of docked conformations. Given that computing the individual energy scores is by far the most computationally expensive task, we note that using multiple coefficient sets does not add any significant computational overhead; all that is needed is computing different combinations of individual energy scores that are computed once, and re-clustering each time.

The cluster centers of each of the clusters produced (50×30) were re-clustered using the same clustering procedure. In this clustering of cluster centers, we used used a 9 Å radius for antibodies and a 19 Å radius for enzyme-containing and others-containing complexes. Once these clusters of cluster centers were formed, we added back the conformations in the original cluster associated with each cluster center, removing any duplicate models. This produced "mixed" clusters, containing conformations scored by different coefficient sets and, possibly, multiple original cluster centers. For each mixed cluster, we selected as representative the original cluster center associated with the largest original cluster. With this strategy, the number of unique cases with acceptable quality increased by 3, 9, and 7 in antibodies, enzymes, and others, respectively. The number of unique cases with medium quality increased by 1, 10, and 5 in antibodies, enzymes, and others, respectively. The details of the impact of this strategy are shown in supplementary Table S5.

2.5. The RRPCC Method

Suppose we have a training set of N protein complexes, indexed by $l = 1, \ldots, N$. For each complex l, we generate a set of n_l clusters containing PIPER-selected conformations, as described earlier. Each cluster $i = 1, \ldots, n_l$, of complex l has an associated cluster representative and a vector $\boldsymbol{p}_i^l \in \mathbb{R}^d$ of d features associated with the cluster and the cluster representative, generated to include all features we outlined earlier. In addition, for every cluster i of complex l, we denote by q_i^l the DockQ score of the cluster representative.

Rather than using regression to predict the DockQ scores, which would equally penalize deviations for either low quality and high quality clusters, we seek to predict the difference of DockQ scores for any pair of clusters corresponding to the same complex. Specifically, for any clusters i,j of a complex l we define $s_{ij}^l = q_i^l - q_j^l$ and seek to find a matrix of coefficients $\mathbf{x} \in \mathbb{R}^{d \times d}$ which solves the following regularized regression problem:

$$min_{\mathbf{x}} \frac{1}{\eta} \sum_{l=1}^{N} \sum_{i=1}^{n_l} \sum_{j=1, i \neq j}^{n_l} \left[(\mathbf{p}_i^l)\prime \mathbf{X} \mathbf{p}_j^l - s_{ij}^l \right]^2 + \mu \|\mathbf{X}\|_2^2, \tag{1}$$

where $\eta = \sum_{l=1}^{N} n_l(n_l - 1)$ and $\mu > 0$ is some scalar regulating the strength of the regularizer. It is straightforward to see that the above problem can be written in a more conventional *ridge regression* form:

$$min_{\mathbf{x}} \frac{1}{\eta} \sum_{l=1}^{N} \sum_{i=1}^{n_l} \sum_{j=1, i \neq j}^{n_l} \left[(\mathbf{v}_{ij}^l)\prime \mathbf{x} - s_{ij}^l \right]^2 + \mu \|\mathbf{x}\|_2^2, \tag{2}$$

where $\mathbf{v}_{ij}^l$ is the vectorized form of $\mathbf{p}_i^l(\mathbf{p}_j^l)\prime$, and $\mathbf{x}$ is the vectorization of $\mathbf{X}$. We elect to use ridge regressions, rather than the ordinary least squares regression, because it has been shown to be robust to potential outliers in the training set, assuming a dense model (i.e., almost all features are informative) [31,32].

Define the matrix $\mathbf{V} \in \mathbb{R}^{\eta \times d^2}$ with rows $\{\mathbf{v}_{ij}^l;\ l = 1, \ldots, N,\ i,j = 1, \ldots, n_l,\ i \neq j\}$ and the vector $\mathbf{s} = (s_{ij}^l;\ l = 1, \ldots, N,\ i,j = 1, \ldots, n_l,\ i \neq j)$. By solving the optimality conditions, it can be easily shown that an optimal solution of (2) can be written in closed-form as:

$$\mathbf{x} = (\mathbf{V}\prime\mathbf{V} + \mu\eta\mathbf{I})^{-1}\mathbf{V}\prime\mathbf{s}. \tag{3}$$

For large enough $\mu\eta$, $\mathbf{V}\prime\mathbf{V} + \mu\eta\mathbf{I}$ is positive definite and the inverse exists.

Given $\mathbf{x}$, we can use it to rank clusters in a newly presented protein complex as follows. Suppose the new "test" complex denoted

[3] http://github.com/bjornwallner/DockQ/

by t has n_t clusters. For any pair of clusters i,j of complex t, we can estimate the relative merit of these clusters by

$$\hat{s}_{ij}^t = (\mathbf{p}_i^t)\prime \mathbf{X} \mathbf{p}_j^t.$$

Comparing a cluster i with every other cluster, we compute a score for cluster i given by

$$\hat{s}_i^t = \frac{1}{n_t - 1} \sum_{j=1, j\neq i}^{n_t} (\mathbf{p}_i^t)\prime \mathbf{X} (\mathbf{p}_j^t), \qquad i = 1, \ldots, n_t. \tag{4}$$

Ordering these scores provides a ranking of the clusters.

2.6. Feature Selection

To improve the prediction performance and avoid overfitting during training the predictive model, we select a subset of the features using three different feature selection and dimensionality reduction methods:

1. *Univariate linear regression test:* We regress the output variable in (2) on each feature in $\mathbf{V}_{ij}^l$ and, using the mean and standard deviation of the corresponding coefficient in $\mathbf{x}$, we compute a p-value associated with the null hypothesis that the coefficient is zero. We remove features (i.e., set to zero the corresponding coefficient) whose p-value exceeds 0.05.
2. *Principal Component Analysis (PCA):* After standardizing the input features, we use Singular Value Decomposition (SVD) to project the standardized data to a lower dimensional space. The dimension of the subspace is selected by the least number of principal component which contribute 98% of the total variation in the dataset. Then, we select the top 45 original features which have the largest contribution to the principal components.
3. *Elastic net regularization:* We add an ℓ_1-norm regularizer to (2) and tune its strength using 5-fold cross-validation. As is well known, such an ℓ_1-norm term can be seen as convexifying the original subset selection problem (which is an integer programming problem)[33] , and suppresses the coefficients of features that are not included in the optimal subset. We remove features whose absolute coefficient is below a certain threshold.

3. Results

We designed, evaluated, and trained RRPCC for antibodies, enzymes, and other complexes, separately. We present results from the two validation strategies, 3-fold cross-validation and historical, outlined in Section 2.3. We present the key results here, while further details on parameter tuning, experimental settings, and the complete list of complexes can be found in the supplementary materials.

In case we do not have enough information to determine the class of proteins the structure being docked belongs, we developed a combined model for enzymes and other complexes; results from this model are included in the supplementary materials. Antibodies were not included in this combined model because antibodies can be differentiated from the other two categories (i.e., enzymes and other complexes) based on their amino acid sequences. This is due to their unique features such as: (1) the presence of heavy and/or light chain (note: both chains need not be present for identifying a sequence as an antibody), (2) the presence of complementarity determining regions (CDR loops) which are easily identifiable due to the unusual number of aromatic residues, and (3) the presence of the constant region which contains essentially the same amino acid sequence in all antibodies of the same class.

3.1. 3-Fold Cross-Validation

The complexes were split into three subsets of roughly equal size. Two subsets were used for training and one for testing. This gives rise to three *folds*, depending on the subset retained for testing. Table 1 presents the ranking results for antibodies, enzymes, and others. In this table, the RRPCC columns use our method and the ClusPro columns rank the clusters in decreasing order of their size. $T_k, k \in \{1, 5, 10\}$, denotes the total number of (test) complexes where a (ranking) model is able to raise at least one medium or acceptable cluster to the top k clusters. We also provide the total number of complexes in each fold with an acceptable or better (#Acc) and medium or better (#Med) cluster, which provides an upper bound on what a ranking method can achieve. The rows labeled 'Total' simply add the number of complexes in each column for each type of complex, providing a metric of performance across folds and protein complexes. The last row simply adds all partial totals. Percentages within parentheses indicate the percentage improvement of RRPCC over ClusPro for the corresponding metric. As seen in Table 1, RRPCC is able to improve T_1 for Acceptable or better predictions by 14%–50% for antibodies, enzymes, and others across the three folds and by 24% for the corresponding 'Total' metric. For Medium or better predictions, T_1 improves by 10%–25% across the three types of complexes and folds, and by 15% for the corresponding 'Total' metric. According to the overall 'Total', T_5 an T_{10} are equal or better for RRPCC.

3.2. Benchmark 5 as the Test Set

In the second validation setting, we train the model on Benchmark 4.0 (BM4) [30] and test on complexes added in Benchmark 5.0 (BM5). Table 2 presents the performance comparison of ClusPro against RRPCC. Interestingly, our ranking algorithm improves ClusPro's T_1 performance ('Total') by 100% for both Acceptable or better predictions and Medium or better predictions, while T_5 and T_{10} performance is equal or slightly better for RRPCC. A more detailed comparison of RRPCC and ClusPro, including how they fare in ranking the highest quality cluster for each complex, can be found in the supplementary materials.

3.3. Comparing Against Other Methods

We compared the RRPCC ranking of the ClusPro-generated clusters with the original ClusPro and the neural network implementation by Dror et al. [21] in Table 3. In the table, a '*' in the T_k column indicates that the method places a cluster with an acceptable or better representative in the top k clusters. [21] used the ATTRACT docking system and the Statistically Optimized Atomic Potential (SOAP) for ranking the models. Upon implementation of their neural network method, PAUL, they reported improvements on 23 of the 55 complexes added to BM5 (ATTRACT did not yield any good solutions in the top 1,000 poses for the rest of the complexes). As shown in Table 3, ClusPro actually performs better than $\text{ATTRACT}_{\text{PAUL}-\text{SOAP}}$ for both T_5 and T_{10} in regards to the number of acceptable or better solutions in these 23 cases. RRPCC ranking raises ClusPro performance above these alternatives. It should be noted that this set of complexes is biased, in the sense that it only contains complexes where ATTRACT was able to find acceptable or better solutions.

In addition to PAUL, the total number of cases with high, medium and acceptable quality from ClusPro and RRPCC were compared with SWARMDOCK before and after implementation of their Machine Learning (ML) enhancement method (IRaPPA) [12] (we refer to this method as SWARMDOCK-ML). The results are in Table 4, using 51 of the 55 added cases to BM5 [10]. While

Table 1
Performance comparison of ClusPro and RRPCC using 3-fold cross-validation.

		Acceptable or better						**Medium or better**							
		RRPCC			**ClusPro**			**RRPCC**			**ClusPro**				
	μ	T_1	T_5	T_{10}	T_1	T_5	T_{10}	T_1	T_5	T_{10}	T_1	T_5	T_{10}	**#Acc**	**#Med**
Antibody															
Fold 1	1	3	6	7	1	6	6	2	3	3	1	3	3	8	3
Fold 2	1	3	6	7	3	7	8	3	4	4	3	4	4	10	7
Fold 3	1	3	7	7	2	5	8	2	5	5	2	3	5	11	5
Total		9 (50%)	19	21	6	18	22	7 (17%)	12	12	6	10	12	29	15
Enzyme															
Fold 1	10	9	16	17	6	16	16	4	7	9	3	7	7	21	14
Fold 2	10	9	14	18	8	14	17	4	6	10	4	7	9	23	10
Fold 3	10	3	12	17	3	13	17	3	5	7	3	5	7	22	12
Total		21 (24%)	42	52	17	43	50	11 (10%)	18	26	10	19	23	66	36
Others															
Fold 1	10	3	8	9	2	8	9	1	2	2	0	2	2	11	4
Fold 2	10	5	8	10	4	8	9	1	3	4	1	2	4	16	6
Fold 3	10	8	10	12	8	10	12	3	5	5	3	5	5	19	7
Total		16 (14%)	26	31	14	26	30	5 (25%)	10	11	4	9	11	46	17
Total		**46 (24%)**	**87**	**104**	**37**	**87**	**102**	**23 (15%)**	**40**	**49**	**20**	**38**	**46**	**141**	**68**

Table 2
Performance comparison of ClusPro and RRPCC using BM5 additions as the test set.

		Acceptable or better						**Medium or better**					
		RRPCC			**ClusPro**			**RRPCC**			**ClusPro**		
	μ	T_1	T_5	T_{10}	T_1	T_5	T_{10}	T_1	T_5	T_{10}	T_1	T_5	T_{10}
Antibody	10	4 (100%)	7	8	2	7	7	3 (50%)	3	3	2	3	3
Enzyme	1	4 (100%)	7	7	2	7	7	2 (100%)	4	4	1	4	4
Others	1	2 (100%)	7	7	1	6	7	1	1	1	0	1	1
Total		**10 (100%)**	**21**	**22**	**5**	**20**	**21**	**6 (100%)**	**8**	**8**	**3**	**8**	**8**

Table 3
Case by case results for ClusPro with and without RRPCC-based ranking compared with ATTRACT results ranked by SOAP and PAUL-SOAP.

		ClusPro			**RRPCC**			**SOAP**			**PAUL-SOAP**		
PDBID	Type	T_1	T_5	T_{10}	T_1	T_5	T_{10}	T_1	T_5	T_{10}	T_1	T_5	T_{10}
2VXT	A	–	*	*	–	*	*	*	*	*	*	*	*
3L5W	A	–	–	–	–	–	*	*	*	*	*	*	*
3MXW	A	*	*	*	*	*	*	*	*	*	*	*	*
4DN4	A	–	*	*	–	*	*	–	–	*	–	*	*
4G6J	A	–	–	–	–	–	–	–	–	–	–	–	–
4G6M	A	–	*	*	*	*	*	–	*	*	–	*	*
2A1A	E	–	–	–	–	–	–	–	–	–	–	–	–
2GAF	E	–	*	*	–	–	–	–	–	–	–	–	–
2YVJ	E	*	*	*	*	*	*	*	*	*	*	*	*
3A4S	E	*	*	*	*	*	*	–	*	*	*	*	*
3H11	E	–	*	*	*	*	*	–	*	*	–	*	*
3K75	E	–	–	–	–	–	–	–	–	–	–	–	–
3PC8	E	–	–	–	*	*	*	*	*	*	*	*	*
3VLB	E	–	*	*	–	*	*	*	*	*	*	*	*
4HX3	E	–	–	–	–	–	–	–	–	–	–	–	–
1M27	O	–	*	*	–	*	*	–	–	–	–	–	–
2GTP	O	–	–	–	–	–	–	–	–	–	–	–	–
2X9A	O	–	*	*	–	*	*	–	–	–	–	–	-
3BX7	O	–	–	*	–	*	*	–	–	–	–	–	-
3DAW	O	–	*	*	*	*	*	–	–	*	–	–	*
3F1P	O	–	–	–	–	*	*	–	*	*	–	*	*
3L89	O	–	*	*	–	*	*	–	–	*	–	*	*
3S9D	O	*	*	*	*	*	*	*	*	*	*	*	*
Antibody		1	4	4	2	4	5	3	4	5	3	5	5
Enzyme		2	5	5	4	5	5	3	5	5	4	5	5
Others		1	5	6	2	7	7	1	2	4	1	3	4
Total		4	14	15	8	16	17	7	11	14	8	13	14

Type: Antibody containing (A), Enzyme containing (E), and Others (O).

Table 4
Comparing RRPCC-based ranking with the original ClusPro and SWARMDOCK.

	ClusPro			RRPCC			SWARMDOCK			SWARMDOCK-ML		
	T_1	T_5	T_{10}	T_1	T_5	T_{10}	T_1	T_5	T_{10}	T_1	T_5	T_{10}
High	0	1	1	0	1	1	0	0	0	0	0	0
Medium	3	7	7	6	7	7	5	9	9	5	10	12
Acceptable	2	12	13	4	13	14	0	5	6	7	8	10
Total	5	20	21	10	21	22	5	14	15	12	18	22

SWARMDOCK-ML places an acceptable or better solution at the top for more complexes, RRPCC ranking significantly improves ClusPro in this category and is slightly below SWARMDOCK-ML. RRPCC outperforms SWARMDOCK-ML in the T_5 and T_{10} metrics, benefiting from the fact that ClusPro was already competitive there (better in T_5 and slightly below in T_{10}). The detailed results are shown in Table S6 of the Supplement.

It is worth noting that with fewer features and only 2 minimization steps (the only computationally expensive steps; cf. Section 2.1 and the generation of the stability features), RRPCC improves on ClusPro as much and often more than IRaPPA does on SWARMDOCK. Comparing these ML-based enhancements to the impact of the neural network on ATTRACT, it is noteworthy that the former actually lead to a more significant improvement compared to the original docking method they are based upon. PAUL leads to performance improvements of 14% in T_1, 18% in T_5, and 0% in T_{10}. This improvement is significantly lower when compared to the improvements by RRPCC which doubles T_1 and IRaPPA which improves SWARMDOCK's T_1 metric by 140%.

3.4. Predictive features

One of the advantages of the regression-based method we used is that it becomes possible to compare the importance of different features, which is not possible with neural network-based methods (used in [21]) or ensembles of decision trees (used in [16]). In order to compute the contribution of each feature in the model, we define a 'feature' score. Specifically, solving the regression problem in (1), yields a matrix of coefficients $\mathbf{x} = (x_{ij})_{i,j=1}^{D}$. Feature $i, i = 1, \ldots, D$, is involved in the ith row and ith column of $\mathbf{x}$. We compute a cumulative score for feature i by:

$$z_i = \sum_{j=1}^{D} (x_{ij} + x_{ji}) - x_{ii}, \qquad i = 1, \ldots, D. \tag{5}$$

Figs. 1–3 show the score of each feature in Enzymes, Antibodies and Others, respectively. Fig. 4, shows eight common features which are important for all three sets of complexes. The description of all features (and the corresponding labels used) are provided in the supplemental Table S4; which uses the following common abbreviation: var (variance), mem (members), bb (backbone), PREMIN (before minimization), and POSTMIN (after minimization).

The feature importance scores shown in Figs. 1–3 reflect differences among the three sets of complexes. Enzymes, known to be more rigid and inflexible upon complex formation, have as the two most dominant features metrics pertinent to intermolecular interactions (long-range hydrogen bond energy and Coulombic electrostatic potential). Antibodies, on the other hand, show that intramolecular energies evaluating phi and psi backbone angles based on Ramachandran maps before and after CHARMM22 minimization have the most impact. Similarly, Others show intramolecular energy terms such as reference energy of each amino acid, bonded angles' energy, and pre- and post-minimization proline ring closure energies. This indicates that protein minimization and ensuing energy calculations are essential for protein complexes showing flexibility on the protein–protein interface.

Despite the difference in the top 2–3 most dominant features, it is still surprising that non-interface related potentials are preva-

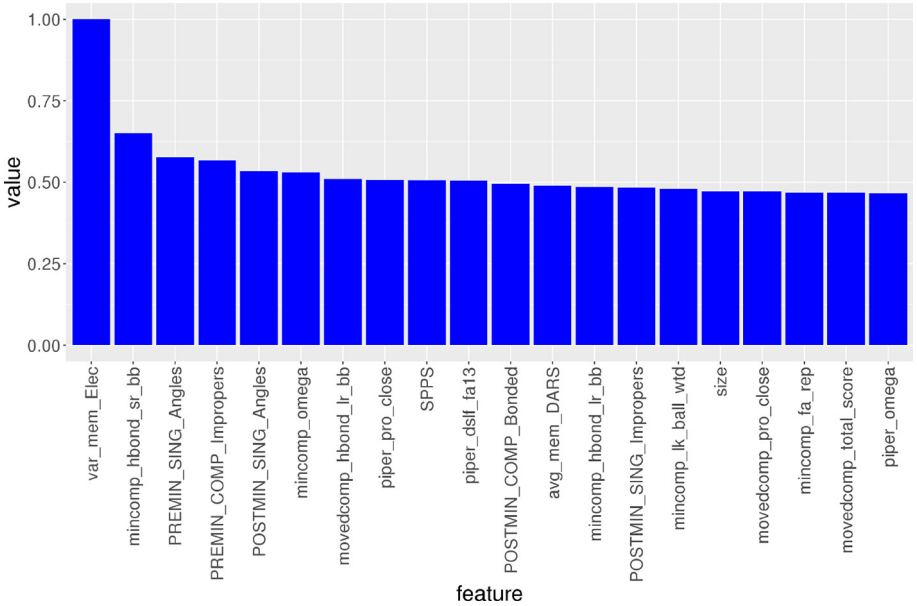

Fig. 1. The most important features in Enzymes. The first two features, var-mem-Elec (variance of Coulombic electrostatics potential of each member of the cluster) and mincomp-hbond-sr-bb (backbone-backbone hydrogen bonds close in primary sequence), are both relevant to intermolecular interactions.

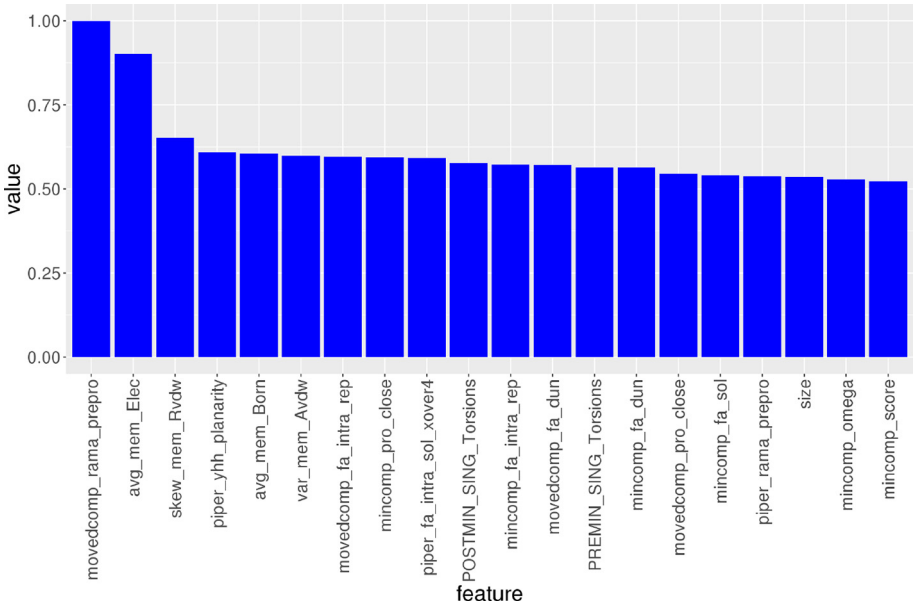

Fig. 2. The most important features in Antibodies. The first two features are movedcomp-rama-prepro (Ramachandran preferences) and var-mem-Elec (variance of Coulombic electrostatics potential of each member of the cluster).

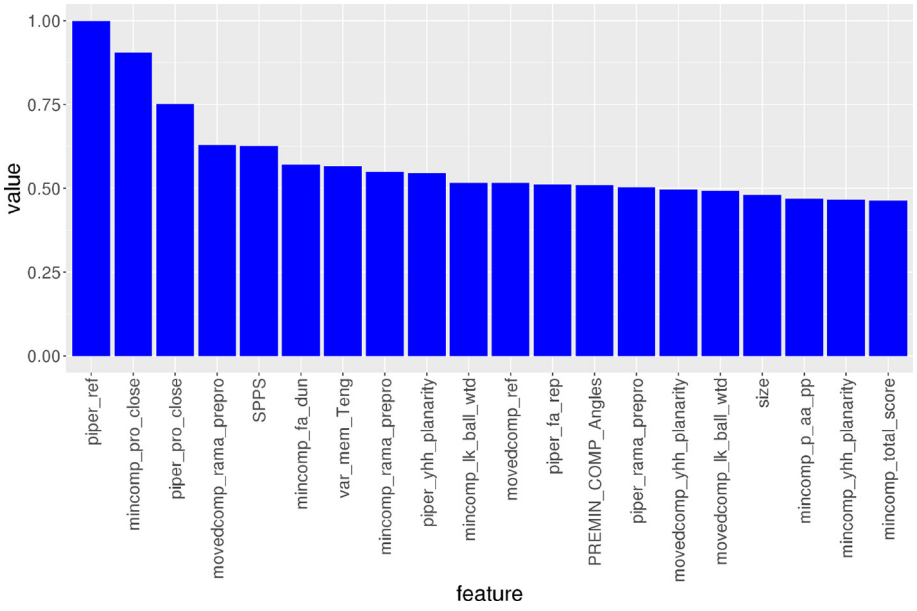

Fig. 3. The most important features in Others. The first three features are piper-ref (reference energy of each amino acid), mincomp-pro-close and piper-pro-close (proline ring closured energy and energy of psi angle of preceding residue).

lent in all three protein classes. Though it is difficult to draw definitive conclusions, it is evident that both pre- and post-minimization potential values of non-interface bonded and non-bonded interactions of good docking poses are distinct from bad ones. This might signify that refinement of protein complex models, and their respective response to the minimization, are strong indicators to how good the model originally was. Since RRPCC is doing a pairwise comparison, it is not hard to imagine the change (instead of the actual value) in these potentials pre- and post-minimization might be indicative of good models.

According to Fig. 4, the common eight features among the three classes of proteins are: (1) statistically optimized atomic potential developed by Sali and colleagues (SPPS) [23], (2) improper dihedral angle energy, (3) population of cluster (size), (4) proline ring closure energy and psi angle of preceding residue (pro_close) after separating the proteins and minimization with CHARMM, (5) Lennard-Jones repulsive energy between atoms of different residues after minimization, (6) Lennard-Jones repulsive energy between atoms of different residues before minimization, (7) proline ring closure energy and psi angle of preceding residue (pro_close) of the complexes (before separation) and after minimization with CHARMM, and (8) total weighted score of ROSETTA calculated energies. This suggests that size, the only feature that ClusPro technically uses for ranking, is a potent feature but is not alone in determining better ranking. Five ROSETTA scores are also prominent as they offer a more fine-tuned perspective of interactions between atoms of different residues. This is expected since ClusPro's energy potentials, especially the DARS potential,

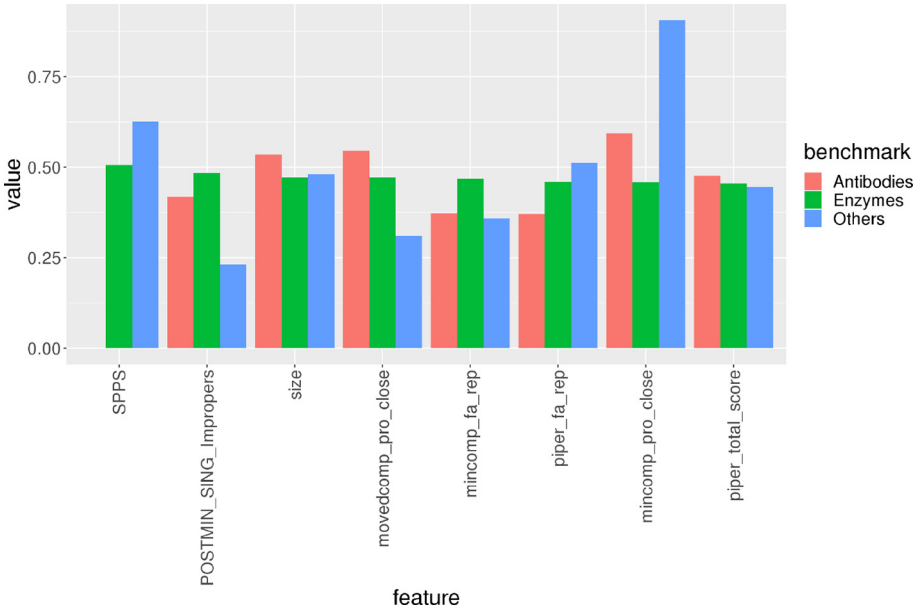

Fig. 4. The common eight features in all three classes of protein complexes.

are more coarse-grained. Since the goal is to discriminate between poses with different interfaces, the results show, as expected, that interface-related potentials (features 1, 3, 4, 5 and 8) dominate the list of common important features.

4. Discussion

As shown in Tables 1 through 4, RRPCC significantly improves the ranking of ClusPro clusters. When training on a randomly selected subset of complexes containing 2/3 of BM5 and testing on the remaining 1/3 of the complexes, RRPCC increases by 24% the number of test complexes where a cluster of acceptable or better quality is ranked first. The corresponding improvement for clusters of medium or better quality is 15%. When training on the version 4.0 of the benchmark [30] and testing on the new complexes added in BM5, the number of test complexes where an acceptable or better quality cluster is ranked first increases by 100%, and the number of test complexes where a medium or better quality cluster is placed at the top is also increased by 100%. RRPCC does not impact as much the quality of the top 5 or top 10 clusters; it mostly pulls the better cluster to the top of the rank.

The combination of RRPCC–CluPro compares well against a number of alternative methods, but there is a remarkable convergence of performances. Although ClusPro, RRPCC, SwarmDock, and PAUL-SOAP were all tested on the BM5 set, the comparison to the neural network based PAUL-SOAP is somewhat limited, because the latter was used to score docked structures generated by the program ATTRACT [21], which produced near-native structures only for 23 of the 55 complexes. Thus, scoring was limited to this subset of 23 targets, and ClusPro was already competitive with PAUL-SOAP in terms of T_5 and T_{10} (Table 3). However, using RRPCC was required to match T_1. RRPCC also improved the T_5 and T_{10} results, but only by adding one good solution for each.

The comparison between ClusPro, RRPCC, and SwarmDock were more interesting (see Table 4), since these methods were tested on the 51 (out of 55) complexes added to BM5 that we could process using the SWARMDOCK server [10]. In addition, from the literature [18], we also had T_1 and T_{10} results for ZRANK, ZRANK2, ProQDock, and ProQDockZ mentioned in the Introduction. RRPCC–ClusPro outperformed SWARMDOCK-ML in terms of T_5 (21 vs. 18), the two methods had the same performance in T_{10} (22 for both), but SWARMDOCK-ML outperformed RRPCC–ClusPro in T_1 (12 vs. 10). Based on Basu and Wallner [18], the T_1 values were 8 for both ProQDock and ProQDockZ, and 4 and 10, respectively, for ZRANK and ZRANK2. Looking at the T_{10} results, ZRANK2 had a correct model for 16 targets, ZRANK had 17, ProQDock had 20 and ProQDockZ had 23. Considering these values as well, the best T_1 performance of 12 is achieved by SWARMDOCK-ML, the best T_5 of 21 by ClusPro-RRPCC (we have no data for ZRANK, ZRANK2, ProQDock and ProQDockZ), and the best T_{10} value of 23 by ProQDockZ, but SWARMDOCK-ML and ClusPro-RRPCC were very close with 22 good structures each. This comparison shows that in spite of the different methodologies, all scoring tools provide very similar success rates, with a difference of one or two complexes. However, it is also necessary to note that even the best success rates are somewhat moderate, e.g., in T_{10} acceptable or better models are obtained at most for 23/51 = 45.1% of the complexes in BM5. The different methods do not all fail on the same targets, and either ClusPro-RRPCC or SWARMDOCK-ML are able to identify an acceptable or better model for 28 of the 51 targets in T_{10}. It is interesting to understand where the failures are coming from. The 51 targets in the test set are categorized as 29 easy (rigid body), 16 medium difficulty, and 6 difficult (flexible). Among the total number of clusters generated by ClusPro and enriched by the use of multiple parameters as described in 2.4, we find acceptable or better models for 20, 6, and 3 in these classes. Thus, the sampling produces acceptable or better structures only for a total of 29 targets. As shown in Table 4, using cluster ranking by RRPCC, we find such structures in T_{10} only for 22 of these 29 targets. Thus, further improvements are important both in sampling and ranking, since good solutions are lost for 22 and 7 targets, respectively, in these computational steps.

The training of RRPCC selects different features to use for ranking, depending on the type of the complex, thus providing insight on what "discriminates" good clusters. Over the entire set of complexes, RRPCC leverages several features in addition to cluster size used by ClusPro, including a statistically optimized potential by Sali et al. [23]. However, the most important features differ among the different types of complexes. Enzymes tend to form more rigid complexes, and leverage electrostatic and long-range hydrogen

bonding energy terms. The antibody-antigen pairs leverage intramolecular energy terms evaluating the likelihood of psi and phi backbone angles. Finally, the other-type complexes rely on energy terms for specific amino acids, bonded angles, and proline ring closure. We emphasize that some of these features are constant when assuming rigid proteins, and become variable only due to the local minimization of the CHARMM energy. Thus, the improved discrimination is partly achieved due to going beyond the rigid body approximation. We attempted to apply the regression method by restricting consideration to rigid body features, and observed a drop in performance. The number of acceptable or better models in T_1 and T_5 was reduced to 7 and 18, respectively, from 10 and 21 shown in Table 4 (data shown in the Supplement, Table S7). While the need for features representing internal energy changes due to local minimization was unexpected, the results shows that the rigid body assumption limits the accuracy of energy evaluation and hence the ability to locate the most native-like docked structures.

CRediT authorship contribution statement

Shahabeddin Sotudian: Conceptualization, Methodology, Software, Validation, Investigation, Formal analysis, Data curation, Writing - original draft, Visualization. **Israel T. Desta:** Conceptualization, Methodology, Software, Validation, Investigation, Formal analysis, Data curation, Writing - original draft, Visualization. **Nasser Hashemi:** Conceptualization, Software, Investigation, Formal analysis, Data curation, Writing - original draft, Visualization. **Shahrooz Zarbafian:** Conceptualization, Investigation, Formal analysis, Data curation, Visualization. **Dima Kozakov:** Conceptualization, Writing - review & editing, Supervision, Funding acquisition. **Pirooz Vakili:** Conceptualization, Writing - review & editing, Supervision, Funding acquisition. **Sandor Vajda:** Conceptualization, Writing - review & editing, Supervision, Funding acquisition.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The research was partially supported by the NSF under grants DMS-1664644, CNS-1645681, DBI 1759277, AF 1645512, and IIS-1914792, by the ONR under grant N00014-19-1-2571, by the NIGMS under grants R21GM127952 and RM1135136, by the NIH under grants R01 GM135930, R35 GM118078, and UL54 TR004130, and by the DOE under grant DE-AR-0001282.

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at https://doi.org/10.1016/j.csbj.2021.04.028.

References

[1] Jones S, Thornton J. Principles of protein-protein interactions. Proc Natl Acad Sci USA 1996;93:13–20.
[2] Alberts B. The cell as a collection of protein machines: Preparing the next generation of molecular biologists. Cell 1998;92:291–4.
[3] Petry S. Mechanisms of mitotic spindle assembly. Annu Rev Biochem 2016;85:659–83.
[4] Camacho CJ, Gatchell DW, Kimura SR, Vajda S. Scoring docked conformations generated by rigid-body protein-protein docking, Proteins: Structure. Funct Gen 2000;40:525–37.
[5] Chuang G-Y, Kozakov D, Brenke R, Comeau SR, Vajda S. DARS (decoys as the reference state) potentials for protein-protein docking. Biophys J 2008;95:4217–27.
[6] Gabb HA, Jackson RM, Sternberg MJ. Modelling protein docking using shape complementarity, electrostatics and biochemical information. J Mol Biol 1997;272:106–20.
[7] Chen R, Li L, Weng Z. Zdock: an initial-stage protein-docking algorithm, Proteins: Structure. Funct Genet 2003;52:80–7.
[8] Tovchigrechko A, Vakser IA. GRAMM-X public web server for protein–protein docking. Nucl Acids Res 2006;34:W310–4.
[9] Yan Y, Tao H, He J, Huang S-Y. The HDOCK server for integrated protein–protein docking. Nat Protocols 2020;15:1829–52.
[10] Desta IT, Porter KA, Xia B, Kozakov D, Vajda S. Performance and its limits in rigid body protein-protein docking. Structure 2020;28:1071–81.
[11] Vreven T, Moal IH, Vangone A, Pierce BG, Kastritis PL, Torchala M, Chaleil R, Jimenez-Garcia B, Bates PA, Fernandez-Recio J, Bonvin AM, Weng Z. Updates to the integrated protein-protein interaction benchmarks: Docking benchmark version 5 and affinity benchmark version 2. J Mol Biol 2015:3031–41.
[12] Moal IH, Barradas-Bautista D, Jiménez-García B, Torchala M, van der Velde A, Vreven T, Weng Z, Bates PA, Fernández-Recio J. IRaPPA: information retrieval based integration of biophysical models for protein assembly selection. Bioinformatics 2017;33:1806–13.
[13] Conte LL, Chothia C, Janin J. The atomic structure of protein-protein recognition sites. J Mol Biol 1999;285:2177–98.
[14] Kozakov D, Hall DR, Xia B, Porter KA, Padhorny D, Yueh C, Beglov D, Vajda S. The ClusPro web server for protein-protein docking. Nat Protoc 2017;12:255–78.
[15] Brenke R, Hall DR, Chuang G-Y, Comeau SR, Bohnuud T, Beglov D, Schueler-Furman O, Vajda S, Kozakov D. Application of asymmetric statistical potentials to antibody–protein docking. Bioinformatics 2012;28:2608–14.
[16] Pfeiffenberger E, Chaleil RA, Moal IH, Bates PA. A machine learning approach for ranking clusters of docked protein-protein complexes by pairwise cluster comparison, Proteins: Structure. Funct Bioinf 2017;85:528–43.
[17] Geurts P, Ernst D, Wehenkel L. Extremely randomized trees. Mach Learn 2006;63:3–42.
[18] Sankar B, Wallner B. Finding correct protein–protein docking models using proqdock. Bioinformatics 2016;32:i262–70.
[19] Pierce B, Weng Z. Zrank: Reranking protein docking predictions with an optimized energy function, Proteins: Structure. Funct Gen 2007;67:1078–86.
[20] Pierce B, Weng Z. A combination of rescoring and refinement significantly improves protein docking performance, Proteins: Structure. Funct Genet 2008;72:270–9.
[21] Eismann S, Townshend RJ, Thomas N, Jagota M, Jing B, Dror R. Hierarchical, rotation-equivariant neural networks to predict the structure of protein complexes, arXiv preprint arXiv:2006.09275 (2020)..
[22] Kozakov D, Brenke R, Comeau SR, Vajda S. PIPER: An FFT-based protein docking program with pairwise potentials. Proteins 2006;65:392–406.
[23] Dong GQ, Fan H, Schneidman-Duhovny D, Webb B, ali A. Optimized atomic statistical potentials: assessment of protein interfaces and loops. Bioinformatics 2013;29:3158–66.
[24] Alford RF, Leaver-Fay A, Jeliazkov JR, O'Meara MJ, DiMaio FP, Park H, Shapovalov MV, Renfrew PD, Mulligan VK, Kappel K, Labonte JW, Pacella MS, Bonneau R, Bradley P, Dunbrack RLJ, Das R, Baker D, Khulman B, Kortemme T, Gray JJ. The rosetta all-atom energy function for macromolecular modeling and design. J Chem Theory Comput 2017;13:3031–48.
[25] Park H, Bradley P, Greisen Jr P, Liu Y, Mulligan VK, Kim DE, Baker D, DiMaio F. Simultaneous optimization of biomolecular energy functions on features from small molecules and macromolecules. J Chem Theory Comput 2016;12:6201–12.
[26] Brooks BR, Brooks III CL, Mackerell Jr AD, Nilsson L, Petrella RJ, Roux B, Won Y, Archontis G, Bartels C, Boresch S, Caflisch A, Caves L, Cui Q, Dinner A, Feig M, Fischer S, Gao J, Hodoscek M, Im W, Kuczera K, Lazaridis T, Ma J, Ovchinnikov V, Paci E, Pastor R, Post C, Pu J, Schaefer M, Tidor B, Venable RM, Woodcock HL, Wu X, Yang W, York D, Karplus M. CHARMM: the biomolecular simulation program. J Comput Chem 2009;30:1545–614.
[27] Mirzaei H, Beglov D, Paschalidis IC, Vajda S, Vakili P, Kozakov D. Rigid body energy minimization on manifolds for molecular docking. J Chem Theory Comput 2012;8:4374–80.
[28] Mirzaei H, Zarbafian S, Villar E, Mottarella S, Beglov D, Vajda S, Paschalidis IC, Vakili P, Kozakov D. Energy minimization on manifolds for docking flexible molecules. J Chem Theory Comput 2015;11:1063–76.
[29] Basu S, Wallner B. DockQ: a quality measure for protein-protein docking models. PloS One 2016;11.
[30] Hwang H, Vreven T, Janin J, Weng Z. Protein–protein docking benchmark version 4.0. Proteins: Struct Funct Bioinf 2010;78:3111–4.
[31] Chen R, Paschalidis IC. A robust learning approach for regression models based on distributionally robust optimization. J Mach Learn Res 2018;19.
[32] Chen R, Paschalidis IC. Distributionally robust learning. Found Trends Optim 2020;4:1–243.
[33] Bertsimas D, King A, Mazumder R. Best subset selection via a modern optimization lens. Ann Stat 2016;44:813–52.
[34] Leaver-Fay A, Tyka M, Lewis SM, Lange OF, Thompson J, Jacak R, et al. ROSETTA3: an object-oriented software suite for the simulation and design of macromolecules. In: Methods in Enzymology, 487. Elsevier; 2011. p. 545–74.