ISSN 2007-9737

Performance of the Classification of Critical Residues at the Interface of BMPs Complexes Pondered with the Ground-State Energy Feature Using Random Forest Classifier

Oscar Roberto Chaparro-Amaro, Miguel de Jesús Martínez-Felipe,
Jesús Alberto Martínez-Castro

Instituto Politécnico Nacional,
Centro de Investigación en Computación,
Mexico

{ochaparroa2019, mmartinezf2020, macj}@cic.ipn.mx

Abstract. This work is focused on implementing and evaluating the Random Forest Classifier (RFC), among other classical machine learning models, on predicting the residues at the interface of protein-protein interactions (PPI) that contribute most of the binding free energy (called hot spots and hot regions). The dataset comprises twenty-nine bone morphogenetic proteins (BMPs) complexes from the Protein Data Bank (PDB). We used just six features such as B-factor, hydrophobicity index, prevalence score, accessible surface area (ASA), conservation score, and the ground-state energy of the amino acids, which were calculated using the Density Functional Theory (DFT). Proving and testing several machine learning methods, we selected the RCF because of its better performance using classical classification metrics and tests. An optimal parameter selection of the RFC reached a better performance using this dataset with around 90 % with the correct class assigned (hot spot & hot region / non-hot spot hot region) residues.

Keywords. Hot spots, hot regions, BMPs, DFT, RFC.

1 Introduction

Protein complexes are compounded by several amino acid chains binding by non-covalent protein-protein interactions [28].

The Bone morphogenetic proteins (BMPs) are a group of similar-structure proteins with short-length amino acid chains and low molecular weight that configures functional growth factors presented in PPI zones [8, 33].

Identifying and classifying these zones, especially the amino acids that thermodynamically convey these interactions, are critical for developing new reaction mechanisms and discovering new drugs inside the Protein complexes [25].

For these reasons, our work is focused on this detection and classification, assuming that the ground-state energy of the amino acids is affected by their interacting zone.

This zone, in which these amino acids are located, is the PPI. The interacting zones between the chain in protein complexes are called interface residues, as shown in Fig. 1, which forms a region where two or more protein chains link themselves by non-covalent interactions such as Van der Waals, electrostatic, hydrogen bonding, ionic, and other forces [12].

1.1 Hot Spots and Hot Regions Classification

The use of thermodynamics to reveal the residues at the interface that mediates the biochemical reaction between protein-protein interactions, combined with machine learning techniques, is well known [23].

These residues have been characterized by employing their free energy ΔG, which contributes to more binding free energy to the interactions than the other ones [20].

A familiar strategy to detect these particular residues is calculating the free energy change $\Delta\Delta G$ function [17, 30].

doi: 10.13053/CyS-27-1-4537

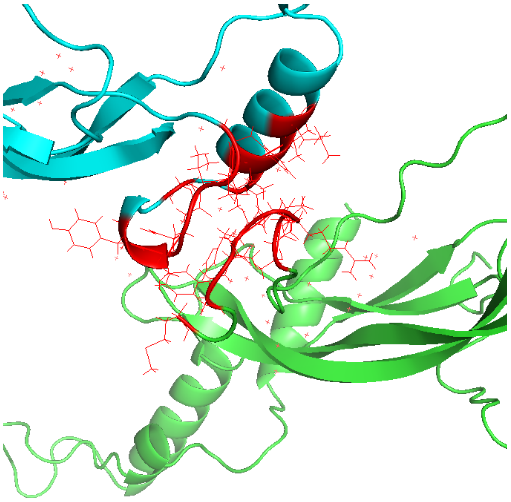

Fig. 1. Interface residues of the 1M4U Protein Complex, which contains two chains (ribbon representation) code A (green) and code L (cyan), respectively. The interface residues of the interaction between them are represented by the red section (visualized using Pymol [34])

Table 1. One entry corresponds to every residue or amino acid (around 2920 entries for training) from the 29 BMPs Complexes

	Features		
ASA	**Ground-State Energy**	**Conservation Score**	**B-factor**
Å^2	E_h	%	Å^2
Features		**Target**	
Hydrophobicity Index	Prevalence Score	$\Delta\Delta G$	$\Delta\Delta G$
%	%	kcal/mol	kcal/mol
		$2 >\rightarrow 1$	$2 <=\rightarrow 0$

Alanine scanning mutagenesis (ASM) is a method to predict $\Delta\Delta G_{bind}$ values using alanine mutation between two non-covalent bonded chains at their interface, measuring the change of free energy in every amino acid [36].

Then, this method is employed for labeling the dataset used in this work. Consequently, the amino acids that obtain a change of free energy $\Delta\Delta G$ more than 2 kcal/mol, are called hot spots [36].

Robetta server is a protein-structure prediction service that calculates the free energy function ΔG and also calculates the binding-mutation function $\Delta\Delta G_{\mathrm{bind},\ n}$ of a specific residue n between two different chains [16].

Hot spots tend to form contact surface areas at the interfaces called hot regions, which are found when the distance between two of their C_α atoms is ≤ 6.5 Å [9]. These regions are critical from a biological activity point of view, so we identified these amino acids by labeling them with these server tools.

Due to the interdisciplinary approach to this problem, identifying the hot spots and the hot regions demands different techniques and approaches. Some of these techniques have been applied with several groups of proteins and datasets using several models [38, 20, 23].

Among popular machine learning algorithms, the Random Forest Classifier has been one of the most useful and successful methods for non-linear classification, and neuronal networks [27, 21]. Therefore, our goal is to show the capability of the ground-state energy of the amino acid molecules for identifying these active sites.

2 Materials and Methods

The general implementation for manipulating and preprocessing the Protein Data Bank (PDB) structures files (.pdb extension) was developed and performed within a C++ framework.

The ASA calculation was supported by GPU GeForce 840M with CUDA (Compute Unified Device Architecture 8.0) API [29]. Part of the DFT calculation and the training process was performed in CPU Intel® Core ™, i7-4510U 2.00GHz (CPU 1).

The remaining DFT calculations were performed in the CPU processors Intel® Core ™and Intel® Core ™2 Quad Processor Q8200 2.33GHz.

To constitute the dataset, we fetched twenty-nine BMP complexes files from the Protein data bank [1] (three-dimensional crystallized structures described in the previous work [7]). Then, this dataset is completed by calculating extra features per amino acid (such as ASA and DFT).

Consequently, even using a small number of proteins, long execution times are required (especially for DFT).

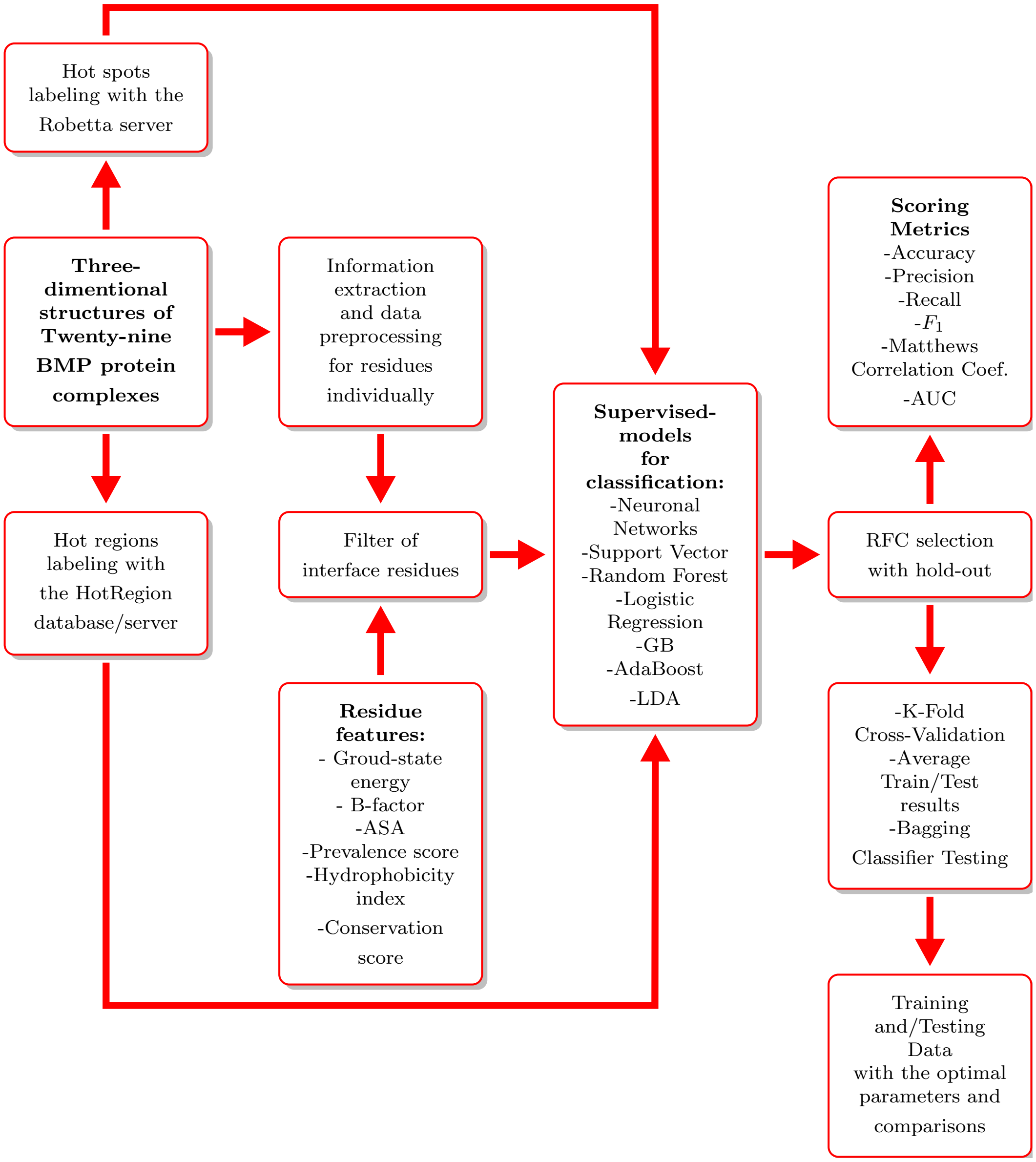
Hot spots labeling with the Robetta server
Three-dimentional structures of Twenty-nine BMP protein complexes
Information extraction and data preprocessing for residues individually
Hot regions labeling with the HotRegion database/server
Filter of interface residues
Residue features: - Groud-state energy - B-factor -ASA -Prevalence score -Hydrophobicity index -Conservation score
Supervised-models for classification: -Neuronal Networks -Support Vector -Random Forest -Logistic Regression -GB -AdaBoost -LDA
Scoring Metrics -Accuracy -Precision -Recall -F_1 -Matthews Correlation Coef. -AUC
RFC selection with hold-out
-K-Fold Cross-Validation -Average Train/Test results -Bagging Classifier Testing
Training and/Testing Data with the optimal parameters and comparisons

Fig. 2. General training, preprocessing and evaluation processes based on supervised machine learning

2.1 Preprocessing

The preprocessing of the protein data can be divided into the .pdb file processing and the calculation and fetching of the training features using either server tools or our methods inside the framework implementation.

Firstly, the .pdb files were ordered, standardized and the Hydrogen atoms were added through the server tool Mol Probity [14] with the flip method (Asn/Gln/His) using electron-cloud $x - H$ bond lengths and Van der Waals radii.

The biochemical properties of amino acids, such as polarity, thermodynamic stability ,and chemical structure, suggest a statistical prevalence between

Table 2. Average performance rates of the additional classifiers. The 0 corresponds to the non-hot spot & non-hot region class, and 1 corresponds to the hot spots and hot region class, respectively

Class	Precision	Recall	F_1-score
	GB classifier		
0	0.86	0.9	0.89
1	0.78	0.7	0.75
	AdaBoost classifier		
0	0.85	0.88	0.86
1	0.72	0.68	0.7
	LDA classifier		
0	0.78	0.86	0.82
1	0.6	0.5	0.55

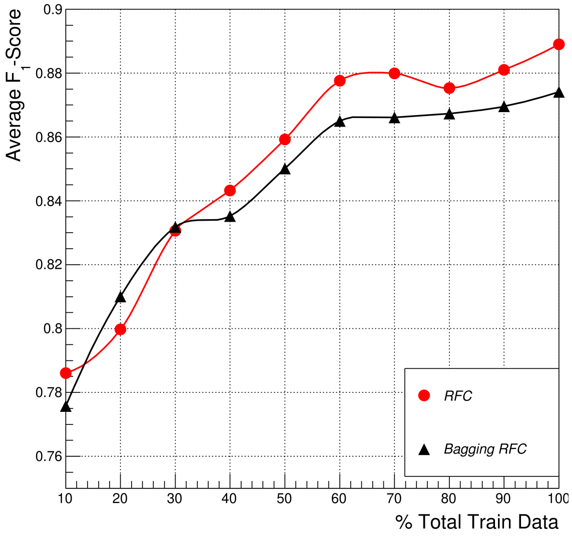

Fig. 3. Average Micro F_1-score performance comparison between a single RFC and a Bagging RFC ensemble (50 estimators) as a function of the dataset size available

them to be more likely hot spots such as valine, leucine and serine [2].

Furthermore, hydrophobicity is a feature that represents the exclusion of solvent from the hot spots and hot regions, which can be obtained as a quantitative measure from [18, 19] for each amino acid type.

Additionally, the evolutionary changes presented in the linear sequence of amino acids of the protein chains over time can be measured using the conservation score, which is convenient for discovering conformational functionality and predicting hot spots under several steps [32].

We used the server ConSurf to calculate this feature by selecting just one iteration for the homolog search algorithm HMMER with E-value=0.0001 of cut-off, using the automatic homologs ConSurf analyses with MAFFT-L-INS-i alignment method and Bayesian calculations.

On the other hand, B-Factor represents the average of the flexibility of the crystallized molecules and has been used to predict hot spots in previous works [39].

The B-factor value is included in the PDB file for every atom and is estimated for every residue using the standardized function from [6].

2.2 DFT and ASA Calculation

The ASA of every amino acid is one of the most important features that can characterize hot spots & hot regions [27].

We implemented the Shrake & Rupley algorithm to calculate the whole ASA protein complexes according to the surface of the Van der Waals atomic spheres [35].

We represent the atomic spheres with a set from one hundred to one thousand points.

To evaluate the solvent exposure, every atomic sphere is in contact with a spherical solvent probe with standard water Van der Waals radii $d_w = 1.4$ Å (insight report is found at [7]).

Then, each ASA residue from each protein complex is extracted and used as an input feature for the classifiers.

An effective method to approximate the ground-state energy (lowest energy value) of a many-body particle system, as amino acids, is using its electronic configuration (DFT procedure), which has been reported with well-correlated results in proteins [11].

In consequence, we used the python module PyQuante2 to calculate the ground-state energy of the individual amino acids of the interfaces with STO-3G basis set, SVWN functional solver, and 0.00001 as tolerance value disregarding effects from neighbor residues [26]. DFT algorithm is heavier and slower for customary scalar processors, especially for large macromolecules formed by amino acid chains.

Therefore, we combined the PyQuante2 methods with the Multiprocessing python module to run the parallel subprocesses concurrently between multiple CPU cores [22].

2.3 Labeling and Training

In total, it was fetched and processed roughly $12,100$ entries (one per amino acid that constitutes each protein in the dataset).

These entries were filtered as interface residues that were in contact between their polypeptide chains using the rules given in [37], since hot spots & hot regions are mandatory residues from the interface.

The dataset of the protein complexes presents 24% of the interface residues, from which 32.4% of these residues are hot spots and hot regions. For training and testing, it is used only the interface residues.

These entries are described in Table 1. Hot spots & hot regions residues were labeled using the ASM computational method from the Robetta server and the HotRegion database [17, 16, 9], following the preprocessing described in Fig. 2.

Comparing several machine learning techniques, we trained a neural network (multilayer perceptron model) with five layers, $\tanh(x)$ activation function, Adam optimization, and binary cross-entropy as loss function using three-hundred epochs with extra data scale preprocessing [13].

Also, we trained a support vector classifier with a radial basis function (rbf): $e^{-\gamma||x-x'||^2}$, in which $\gamma = 0.01$ in this kernel and the regularization parameter of $\rho = 10$.

A hyperparameter-grid search optimization was performed over the parameters of these models using Scikit-learn in the training data.

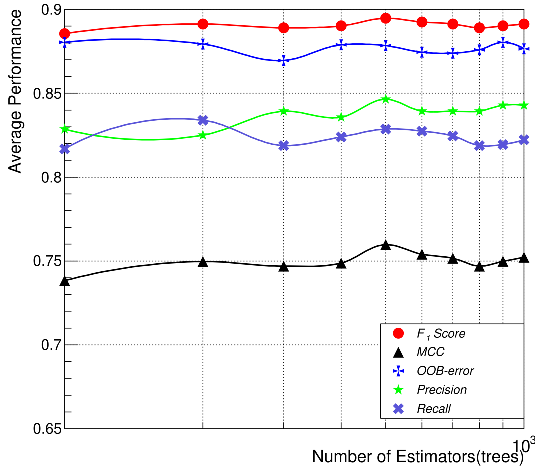

Fig. 4. Average performance comparison between the five performance metrics used, based on the number of estimators or trees in the RFC, in which every point represents a train/test evaluation using 70 and 30, respectively

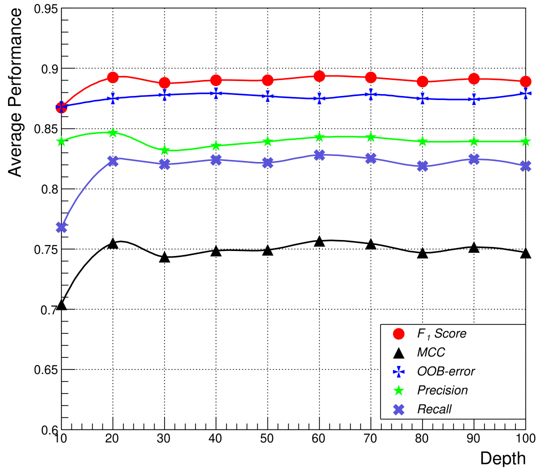

Fig. 5. Comparison of the average performance of the five metrics as a function of the maximum depth of all the trees (fixed as 500 trees) in the RFC, in which every point represents a train/test evaluation using 70 and 30 repeated ten times, respectively

This optimization was performed in the following classification algorithms: Gradient Boosting (GB), with 1000 estimators, a maximum depth of 100, and a learning rate of 1, the AdaBoost using 1000 estimators and the Linear Discriminant Analysis (LDA) with a Singular value decomposition.

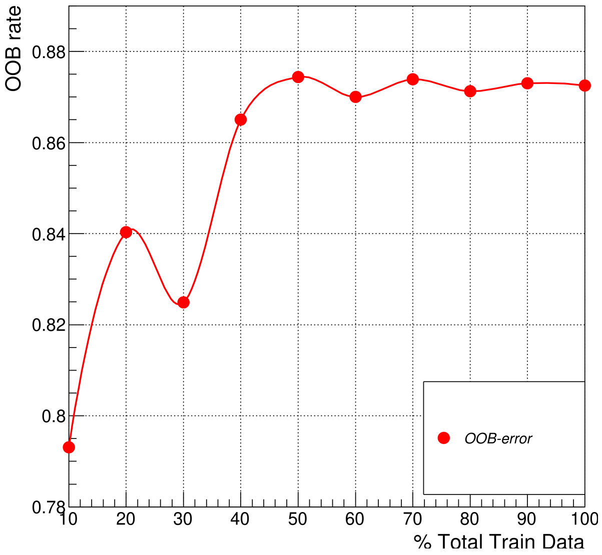

Fig. 6. Average OOB error rates in function of different percentages of data used for training for an RFC with 500 trees and a depth value of 60

Table 3. Average performance rates of the RFC. The mean of the accuracy or micro-F_1-score was ≈ 0.89, with Std. Dev. ≈ 0.00145 and MCC=0.75 with $\sigma = 0.004$. The 0 corresponds to the non-hot spot & non-hot region class, and 1 corresponds to the hot spots and hot region class, respectively

Class	**Precision**	
	μ	σ
0	0.92	0.0017
1	0.82	0.0035
Class	**Recall**	
0	0.91	0.0019
1	0.84	0.004
Class	F_1**-score**	
0	0.92	0.0013
1	0.83	0.003

Besides, an RFC was trained with different parameters and configurations, such as a bagging ensemble estimator [5]. Regarding the whole dataset, we split it randomly (hold-out technique with ten-folds), where 70% and 30% of this dataset were for training and testing, respectively.

3 Results and Discussion

3.1 Accelerated Performance

We implemented CPU/GPU hardware acceleration techniques during the preprocessing as parallel programming to speed up the ASA and DFT calculations.

Our parallel ASA implementation improves the performance in execution time in contrast with the scalar Python Pdbremix API [31] implementation based on the number of points used to represent the Van der Waals atomic spheres (Shrake & Rupley approximation).

In the ASA calculation case, the parallel GPU implementation gets a speed-up ratio between three to twenty-two times, reducing the execution time as the protein chains become larger in relation to the number of atoms.

These were possible by locally distributing a load of data in each available core using a three-dimensional spatial box, calculating the ASA between the adjacent atoms and residues inside these boxes [7].

This implementation is still not as fast as some applications, such as FreeSasa [24]. However, adapting ASA calculation algorithms such as the Linear Combinations of Pairwise Overlaps (LCPO), using these schemes presented, enables scalability on the calculations according to the protein size.

On the other hand, the DFT calculation showed a slight speed improvement of 3.5 to 4 times (according to the number of CPU cores), as reported in [7].

3.2 Models Testing

The classification models were evaluated with the Accuracy, Precision, Recall, F_1-score, and Matthews correlation coefficient (MCC) rates using both micro-average and macro-average methods, respectively [20, 3].

In the same way, we assessed the binary classification with the Receiver Operating Characteristics (ROC) curves and the Area Under The Curve (AUC).

Previously, some machine learning algorithms, such as Support Vector Classifier (SVC) and Neuronal Networks, were tested and assessed, as well as the RFC.

Their respective average AUC values denoted that the better method is the RFC [7]. The results of the GB, AdaBoost, and LDA classifiers are shown in Table 2:

These classifiers do not reach enough performance in comparison with classifiers based on decision trees as the RFC.

The results report of the NN, the SVM, and the Logistic Regression classifiers is found in [7]. Furthermore, a single RFC obtained an $\mathrm{AUC} \approx 0.95$.

The feature importance for this classifier from the most to the least important was: ASA (0.31), ground-state energy (0.18), conservation score (0.16), B-factor (0.12), hydrophobicity index (0.11), and prevalence score (0.09) of feature importance respectively.

3.3 Bagging Random Forest Classifier Test

RFC can be assembled into a bootstrap aggregating ensemble (decision trees), especially when it is necessary to obtain variance reduction [5].

Fig. 3 shows the comparative performance between a single RFC and a Bagging RFC ensemble in the function of the training data percentage, where one hundred percent of the training set represents seventy percent of the total data assigned randomly.

These experiments show the bagging classifiers based on RFC estimators are less efficient than the single RFC, particularly when the total amount of training data available is used (the 100 percent obtained in both techniques is the best result).

This indicates that many estimators based on trees are over-trained with a loss of generalization for this particular problem. Then, we suggest applying a single RFC model to have better results.

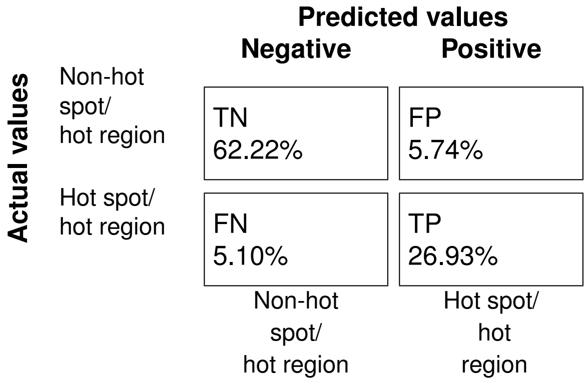

Fig. 7. Confusion matrix of the average RFC. Combining both the true negative (TN) and true positive (TP) values we obtained roughly 90% (correct prediction) of the total evaluation

3.4 RFC Validation

The parameters used in the RFC model were the number of estimators or trees and the depth (maximum of samples per tree in the splinted leaves [4]).

In this particular case, we found that applying more than the minimal number of samples per leaf (one) reduces the accuracy of the prediction.

Therefore, to search for the best parameter values and to see the effect of this variation, we followed the next steps:

Firstly, we measured the average of F_1-score, MCC, and the OOB error (defined only for RFC) metrics with the whole dataset, varying the number of estimators using a depth value of 50. We searched from one hundred to one thousand estimators.

Fig. 4 expresses non-significant changes in the general performance ($\mu \approx 0.89, \approx 0.878$, and ≈ 0.748) for F_1-score, OOB error, and MCC respectively), so we propose the application of 500 estimators.

Secondly, by fixing the number of estimators and varying the depth of the RFC, the metrics reveal a stable behavior after a depth value of 20, reaching an average of $\mu \approx 0.89, 0.875, 0.75$ for F_1-score, OOB error, and MCC, respectively.

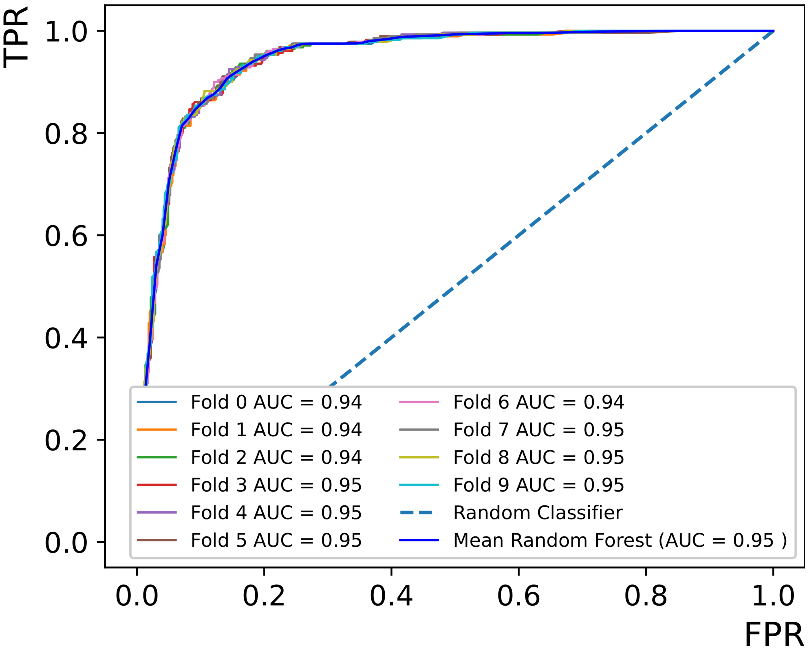

Fig. 8. Average ROC curve performance of ten train/test evaluations. A $\mu = 0.95$ is obtained by interpolating the curves. Y-axis is the True Positive Rate (TPR), and X-axis is the False Positive Rate (FPR)

Subsequently, it is possible to set a depth value of 60, from which is obtained a minimal variation (Fig. 5). Using these RFC parameters, the dataset for training and testing was validated with the OOB error varying the total train data as previously.

This metric also relates the usage of enough quantity of data for the estimators in the RFC [15]. Therefore, the OOB error reveals an average performance of 0.87, starting at 40% of the training data, showing better stability of the testing output results when we used the training data (see Fig. 6).

Thirdly, the K-fold cross-validation showed the average data consistency between the training and testing proofs. It was applied ten splits repeated ten times (one hundred in total) throughout the whole dataset, obtaining an interpolated ROC curve with an AUC metric of 0.94, which expresses a correlation with low variation.

According to the previous results, the ASA and the ground-state energy areas of the hyperplane of the two-dimensional projection are shown in [7]. This projection supports that the ground-state energy improves the classification.

Besides, training an RFC without the ground-state energy feature obtains an $\mathrm{AUC} \approx 0.9$, which is a worse performance. The results of the optimal parameters using the six features are shown in Table 3.

Likewise, the average values of $(\mathrm{TN}, \mathrm{TP}, \mathrm{FN}, \mathrm{FP})$ are represented in the confusion matrix in Fig. 7, as well as the respective ROC curve (Fig. 8), using these parameters.

We should emphasize that the conservation score is kept as the third and the B-factor as the fourth in feature importance in the whole training process, indicating that these features contribute to the total estimation function of the classifiers also.

The training and testing process using the balanced data does not improve the performance since the F_1-score has an accuracy of 0.814 for the hot spot & hot region class, and 0.893 for the complementary.

Since almost 70% are not hot spot & hot regions class, the balance data lost relevant information. An advantage of this model is that the RFC training time is faster than the others.

Conversely, the execution time of the DFT algorithm is the main disadvantage of this method. This problem is treated by applying High-performance computing (HPC) or hardware acceleration techniques using deeper parallel implementations.

In addition, DFT can reveal the relation between the electronic configuration energy in the ground state and the binding free energy of the amino acids.

Moreover, we compared the label test data with the foldX ASM framework [10] (for each complex protein in the test data), in which we got ≈ 0.71 and ≈ 0.79 for precision and recall, respectively.

The method for labeling hot spots (ASM from Robetta) has a similar principle [16]. However, the high deviation corresponds to the addition of hot regions (which foldX ASM does not estimate), so a direct comparison is not allowed in this particular case.

Consequently, the RF comparison with foldX test data was $\approx 0.733, 0.79$ and ≈ 0.8 for accuracy, precision, and recall, respectively. The detection of the active amino acids (class 1) is a crucial part of the work, so a higher TP rate is preferred. Thus, the recall metric should be prioritized.

4 Conclusion and Future Work

The RFC proved to be a suitable method for predicting and detecting hot spots and hot regions using this fetched dataset.

A single RFC brings the best results over the rest of the machine learning models applied with this set of features and configurations, including the Bagging RFC classifier ensemble.

Notable advantages of this method are its non-linearity capability separation and the easy and light implementation according to the metrics used in this work.

The main limitation of this work was the completion of the dataset, which needs the DFT calculation in interface residues.

In this sense, DFT calculation is still a hard computational chemistry challenge, which needs several computational strategies to reduce the execution calculation time.

Consequently, the development of acceleration techniques in software and hardware for calculating protein parameters and model refinement is crucial because of the constant increase of data and the current higher computational resources demanded.

A strong result of this work is that the ground-state energy was the second most relevant feature scored, used by the RFC to classify hot spots & hot regions.

This highlight that this feature reveals important energetic information about the protein-protein interactions of BMPs and could help to clear up biological activity. Therefore, expanding the dataset to continue extracting information (especially adding the ground-state energy) should be reinforced.

Consequently, we explored the parallelization via GPU of the ASA calculation (Shrake and Rupley approximation), improving the execution time in contrast with the scalar process. However, the parallelization of newer ASA calculation algorithms should be adapted to obtain better performance.

Finally, a single RFC trained with the six features mentioned above can describe hot spots and hot regions at the protein-protein interfaces with optimal parameter selection with enough performance described by the classification metrics.

Acknowledgments

The authors declare that no competing financial interests exist. We thank ”Consejo Nacional de Ciencia y Tecnología” (CONACYT).

References

1. **Berman, H., Henrick, K., H.Nakamura, Markley, J. (2007).** The worldwide protein data bank (wwPDB): Ensuring a single, uniform archive of PDB data. Nucleic Acids Research, Vol. 35. DOI: 10.1093/nar/gkl971.
2. **Bogan, A. A., Thorn, K. S. (1998).** Anatomy of hot spots in protein interfaces. Journal of Molecular Biology, Vol. 280, No. 1, pp. 1–9. DOI: 10.1006/jmbi.1998.1843.

3. **Boughorbel, S., Jarray, F., El-Anbari, M. (2017).** Optimal classifier for imbalanced data using matthews correlation coefficient metric. PLoS ONE, Vol. 12, No. 6, pp. 1–17. DOI: 10.1371/journal.pone.0177678.

4. **Breiman, L. (2001).** Random forests. Machine Learning, Vol. 45, No. 1, pp. 5–32. DOI: 10.1023/A:1010933404324.

5. **Bühlmann, P. (2011).** Bagging, Boosting and Ensemble Methods. Springer Berlin Heidelberg, pp. 985–1022. DOI: 10.1007/978-3-642-21551-3_33.

6. **Carugo, O. (2018).** How large B-factors can be in protein crystal structures. BMC Bioinformatics, Vol. 19, No. 61, pp. 1–9. DOI: 10.1186/s12859-018-2083-8.

7. **Chaparro-Amaro, O., Martínez-Felipe, M. J., Martínez-Castro, J. M. (2022).** Hot spots and hot regions detection using classification algorithms in BMPs complexes at the protein-protein interface with the ground-state energy feature. Lecture Notes in Computer Science, pp. 3–14. DOI: 10.1007/978-3-031-07750-0_1.

8. **Chen, D., Zhao, M., Mundy, G. R. (2004).** Bone morphogenetic proteins. Growth Factors, Vol. 22, No. 4, pp. 233–241.

9. **Cukuroglu, E., Gursoy, A., Keskin, O. (2011).** Hotregion: a database of predicted hot spot clusters. Nucleic Acids Research, Vol. 40, No. 22080558, pp. 829–833. DOI: 10.1093/nar/gkr929.

10. **Durme, J. V., Delgado, J., Stricher, F., Serrano, L., Schymkowitz, J., Rousseau, F. (2011).** A graphical interface for the FoldX forcefield. Bioinformatics, Vol. 27, No. 12, pp. 1711–1712. DOI: 10.1093/bioinformatics/btr254.

11. **Fox, S. J., Dziedzic, J., Fox, T., Tautermann, C. S., Skylaris, C.-K. (2014).** Density functional theory calculations on entire proteins for free energies of binding: Application to a model polar binding site. Proteins: Structure, Function, and Bioinformatics, Vol. 82, No. 12, pp. 3335–3346. DOI: 10.1002/prot.24686.

12. **Guo, W., Wisniewski, J. A., Ji, H. (2014).** Hot spot-based design of small-molecule inhibitors for protein–protein interactions. Bioorganic and Medicinal Chemistry Letters, Vol. 24, No. 11, pp. 2546–2554. DOI: 10.1016/j.bmcl.2014.03.095.

13. **Haykin, S., Haykin, S. (2009).** Neural Networks and Learning Machines. Number 10in Neural networks and learning machines. Prentice Hall.

14. **Hintze, B. J., Lewis, S. M., Richardson, J. S., Richardson, D. C. (2016).** Molprobity's ultimate rotamer-library distributions for model validation. Proteins: Structure, Function, and Bioinformatics, Vol. 84, No. 9, pp. 1177–1189. DOI: 10.1002/prot.25039.

15. **Janitza, S., Hornung, R. (2018).** On the overestimation of random forest's out-of-bag error. Public Library of Science One, Vol. 13, No. 8, pp. e0201904. DOI: 10.1371/journal.pone.0201904.

16. **Kortemme, T., Baker, D. (2002).** A simple physical model for binding energy hot spots in protein-protein complexes. PNAS, Vol. 99, No. 22, pp. 14116–14121. DOI: https://doi.org/10.1073/pnas.202485799.

17. **Kortemme, T., Kim, D. E., Baker, D. (2004).** Computational alanine scanning of protein-protein interfaces. Science's STKE, Vol. 2004, No. 219. DOI: 10.1126/stke.2192004pl2.

18. **Kyte, J., Doolittle, R. F. (1982).** A simple method for displaying the hydropathic character of a protein. Journal in Molecular Biology, Vol. 5, No. 157, pp. 105–132. DOI: 10.1016/0022-2836(82)90515-0.

19. **Law, K. Y. (2014).** Definitions for hydrophilicity, hydrophobicity, and superhydrophobicity: Getting the basics right. The Journal of Physical Chemistry Letters, Vol. 5, No. 4, pp. 686–688. DOI: 10.1021/jz402762h.

20. **Lise, S., Archambeau, C., Pontil, M., Jones, D. T. (2009).** Prediction of hot spot residues at protein-protein interfaces by combining machine learning and energy-based methods. BMC Bioinformatics, Vol. 10, No. 1. DOI: 10.1186/1471-2105-10-365.

21. **Liu, S., Liu, C., Deng, L. (2018).** Machine learning approaches for protein-protein interaction hot spot prediction: Progress and comparative assessment. MDPI Molecules, Vol. 23, No. 10, pp. 2535. DOI: 10.3390/molecules23102535.

22. **McKerns, M. M., Strand, L., Sullivan, T., Fang, A., Aivazis, M. A. G. (2012).** Building a framework for predictive science. DOI: 10.48550/ARXIV.1202.1056.

23. **Melo, R., Fieldhouse, R., Melo, A., Correia, J., Cordeiro, M., Gümüş, Z., Costa, J., Bonvin, A., Moreira, I. (2016).** A machine learning approach for hot-spot detection at protein-protein interfaces. International Journal of Molecular Sciences, Vol. 17, No. 8, pp. 1215. DOI: 10.3390/ijms17081215.

24. **Mitternacht, S. (2016).** FreeSASA: An open source c library for solvent accessible surface area calculations. F1000Research, Vol. 5, pp. 189. DOI: 10.12688/f1000research.7931.1.

25. **Morrow, J. K., Zhang, S. (2012).** Computational prediction of protein hot spot residues. Current Pharmaceutical Design, Vol. 18, No. 9, pp. 1255–1265. DOI: 10.2174/138161212799436412.

26. **Muller, R. (2013).** Pyquante2. PyQuante Sourceforge Project Page.

27. **Nguyen, Q. T., Fablet, R., Pastor, D. (2013).** Protein interaction hotspot identification using sequence-based frequency-derived features. IEEE Transactions on Biomedical Engineering, Vol. 60, No. 11, pp. 2993–3002. DOI: /10.1109/TBME.2011.2161306.

28. **Nussinov, R., Schreiber, G.**, editors **(2009).** Computational Protein-Protein Interactions. CRC Press. DOI: 10.1201/9781420070071.

29. **NVIDIA, Vingelmann, P., Fitzek, F. H. (2020).** Cuda, release: 10.2.89.

30. **Ozdemir, E. S., Gursoy, A., Keskin, O. (2018).** Analysis of single amino acid variations in singlet hot spots of protein–protein interfaces. Bioinformatics, Vol. 34, No. 17, pp. i795–i801. DOI: 10.1093/bioinformatics/bty569.

31. **PDBremix (2014).** Calculating the solvent accessible surface area.

32. **Qiao, Y., Xiong, Y., Gao, H., Zhu, X., Chen, P. (2018).** Protein-protein interface hot spots prediction based on a hybrid feature selection strategy. BMC Bioinformatics, Vol. 19, No. 1. DOI: 10.1186/s12859-018-2009-5.

33. **Reeves, A.**, editor **(2017).** In Vitro Mutagenesis: Methods and Protocols. Springer New York. DOI: 10.1007/978-1-4939-6472-7.

34. **Schrödinger, L. (2015).** The PyMOL molecular graphics system, version 1.8.

35. **Shrake, A., Rupley, J. A. (1973).** Environment and exposure to solvent of protein atoms. lysozyme and insulin. Journal of Molecular Biology, Vol. 79, No. 2, pp. 351–371. DOI: 10.1016/0022-2836(73)90011-9.

36. **Tamulewicz, A. (2015).** Methods of hot spot identification in protein complexes. CHEMIK Science, Vol. 69, No. 6, pp. 331–334.

37. **Tuncbag, N., Keskin, O., Gursoy, A. (2010).** HotPoint: hot spot prediction server for protein interfaces. Nucleic Acids Research, Vol. 38, No. Web Server, pp. W402–W406. DOI: 10.1093/nar/gkq323.

38. **Wang, L., Liu, Z. P., Zhang, X. S., Chen, L. (2012).** Prediction of hot spots in protein interfaces using a random forest model with hybrid features. Protein Engineering Design and Selection, Vol. 25, No. 3, pp. 119–126. DOI: 10.1093/protein/gzr066.

39. **Xia, J. F., Zhao, X. M., Song, J., Huang, D. S. (2010).** APIS: accurate prediction of hot spots in protein interfaces by combining protrusion index with solvent accessibility. BMC Bioinformatics, Vol. 11, No. 1. DOI: 10.1186/1471-2105-11-174.

Article received on 21/08/2022; accepted on 01/12/2022. Corresponding author is Jesús Alberto Martínez-Castro.