# Feature Design for Protein Interface hotspots using KFC2 and Rosetta

Franziska Seeger, Anna Little, Yang Chen, Tina Woolf, Haiyan Cheng and Julie *C*. Mitchell*

**Abstract**

Protein-protein interactions regulate many essential biological processes and play an important role in health and disease. The process of experimentally characterizing protein residues that contribute the most to protein-protein interaction affinity and specificity is laborious. Thus, developing models that accurately characterize hotspots at protein-protein interfaces provides important information about how to inhibit therapeutically relevant protein-protein interactions. During the course of the ICERM WiSDM workshop 2017, we combined the KFC2a protein-protein interaction hotspot prediction features with Rosetta scoring function terms and interface filter metrics. A 2-way and 3-way forward selection strategy was employed to train support vector machine classifiers, as was a reverse feature elimination strategy. From these results, we identified subsets of KFC2a and Rosetta combined features that show improved performance over KFC2a features alone.

---

Franziska Seeger
University of Washington, Institute for Protein Design, Seattle WA, e-mail: fseeger@uw.edu

Anna Little
Michigan State University, East Lansing MI, e-mail: littl119@egr.msu.edu

Yang Chen
University of Michigan, Ann Arbor MI, e-mail: ychenang@umich.edu

Tina Woolf
Jet Propulsion Laboratory, Pasadena CA, e-mail: tina.m.woolf@jpl.nasa.gov

Haiyan Cheng
Willamette University, Salem OR, e-mail: hcheng@willamette.edu

*Correspondence to Julie Mitchell
Oak Ridge National Laboratory, Knoxville TN & University of Wisconsin - Madison, Madison WI
e-mail: mitchelljc@ornl.gov

# 1 Introduction

Protein-protein interactions play a crucial role in biochemical processes. Modulation of protein-protein interactions bears enormous potential for therapeutic drug development. Thus, accurate predictive models of protein-protein interactions will not only enhance our understanding of the molecular basis of protein recognition and specificity but further provide and inform efforts to modulate protein-protein interactions. Certain hotspot residues at protein-protein interfaces contribute more binding energy to the interaction than others. An *alanine mutagenesis hotspot* in a protein-protein interface is an amino acid for which the change in binding energy upon mutation to alanine exceeds 2 kcal/mol. That is, the change in energy upon binding ($\Delta G_{bind}$) is increased by at least 2 kcal/mol ($\Delta \Delta G_{bind} > 2$ kcal/mol). hotspots are known to contribute significantly to the energetics of protein-protein interaction [7, 22, 40]. hotspot analysis has both a long history as well as many recent contributions [1, 2, 4–8, 11, 12, 14–19, 21–25, 28, 29, 33–37, 40, 42, 45–51, 53–57]. Early work on analysis of protein structures in relation to mutagenesis effects established the structural and chemical properties of amino acid residues that significantly alter binding free energy when mutated to alanine [7, 21, 22]. More recent work has begun to characterize hotspot regions, chemical alignment of interfaces, and structural evolution of hotspots [12, 50, 51].

The KFC and KFC2 models for predicting binding interface hotspots [13, 14, 59] have become a gold standard for hotspot prediction. The KFC2 model identifies about 80% of known hotspots [59]. An important recent study of antibody design found the KFC2 model largely in sync with experimental predictions [52]. KFC2 is available via a public web server [14] and has been accessed nearly 80,000 times. The original KFC model examined geometric and biochemical features of a protein-protein interface and used decision trees to develop an accurate predictive model. The KFC2 model pursued a similar line of approach, using support vector machines to train the model and introducing new features that have stronger predictive value than the original ones. In particular, the introduction of interface plasticity measures has significantly improved our ability to distinguish hotspots from non-hotspots.

Rosetta is a molecular modeling and design software suite that has been used for a variety of tasks ranging from protein structure prediction [41] to de novo protein design [26, 30] and protein-protein interface design [9]. Rosetta-based energy calculations [3] have been previously used to create a model for predicting protein-protein interface hotspots [28]. In this work, we will add features from Rosetta to those of KFC2 and train an improved model for protein-protein hotspot prediction. We will combine strategies for feature selection with support vector machine learning in order to achieve an optimal model.
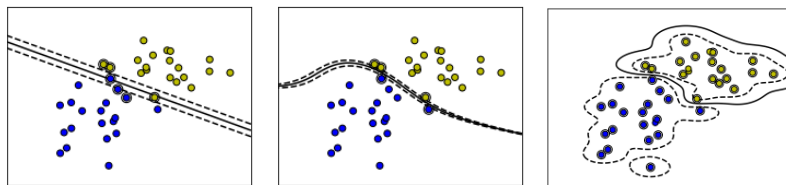
**Fig. 1** Demonstration of SVM classification with linear (left), polynomial (middle), and RBF (right) kernels. The yellow and blue dots correspond to data points simulated from two multivariate normal distributions, i.e. two classes. The solid and dashed lines are the contours of the SVM decision function with levels 0 (solid curve) and $\pm 0.5$ (dashed curves).

## 2 Background on SVM

Support vector machine (SVM) is a widely adopted binary classifier in recent years due to its efficiency and accuracy [10, 39]. As a supervised classification algorithm, SVM uses labeled training data to build a model and infers the two categories of the testing data. The two categories correspond to hotspot or non-hotspot in our case. In SVM, each data point is represented using a $d$-dimensional vector of descriptors/features and a label that denotes the class (hotspot, non-hotspot). Given labeled training data, SVM identifies a separating hyperplane in the high-dimensional feature space, with each side of the hyperplane corresponding to one (predicted) class. This hyperplane can be used to classify testing data for which the class is unknown. In practice, there are multiple valid hyperplanes that separate the training data. The hyperplane that SVM selects maximizes the distance between the hyperplane and the nearest data point to each side of the hyperplane.

The SVM classifier can be linear or nonlinear, depending on the choice of kernel; see the documentation for SVM from `scikit-learn` [43] for more details. Figure 1 gives an example showing SVM classifiers which use linear, polynomial, and (Gaussian) Radial Basis Function (RBF) kernels, respectively. We choose to use the RBF kernel for our data due to its utility in obtaining the best models for this application. There are two parameters controlling the SVM classifier, $C$ which controls the margin between support vectors and the separating hyperplane, and $\gamma$, which controls the shape of the RBF kernel.

For our training runs, we tabulated performance based on a five-fold cross-validation. Each $C$ and $\gamma$ combination is checked using cross-validation, and the combination that leads to the best cross-validation accuracy is selected. See Section 4 for a more detailed description of the SVM implementation and parameter tuning for the hotspot data set.

In the hotspot classification problem, the proportion of hotspots is much smaller than the proportion of non-hotspots. This problem is typically referred to as classification for highly unbalanced data. In this case, the decision function is more driven by the more prevalent class (non-hotspots) instead of the other (hotspots). In order to avoid this issue, we adopt the "class-weighted" SVM: assigning higher misclas-

3

sification penalties to the instances in the rare class and vice versa in the training data so that the decision boundary is almost equally influenced by the two classes. We use the SVC function in `scikit-learn` [43] to implement this.

In interpreting the results of SVM feature selection and parameterization, it is important to understand any resulting model represents *a good choice* rather than *the best choice*. However, we will see some patterns emerge in feature selection if we build a range of models using different parameters.

## 3 Data Sets and Features

The original KFC and KFC2 data sets are described in [13, 14, 59]. For this work, we used a newer expanded data set of alanine mutagenesis hotspots, available from the SKEMPI database [38]. Note that SKEMPI distributes a set of cleaned and renumbered protein structure files that align with their database entries, on which our feature calculations were performed. All KFC2 features were calculated on the structure for the complex, and Rosetta features used relaxed structures and *in silico* mutants.

Structures of the SKEMPI data set of mutant empirical interactions were relaxed in the latest Rosetta full-atom forcefield, REF15, while being constrained to input atomic coordinates [3]. A computational model was generated for each described interface mutation in the SKEMPI dataset by first replacing the native residue with an Alanine residue and performing local side chain minimization within 8 $\mathring{A}$ of the mutated residue. All wild-type and mutant structures were scored with the REF15 Rosetta energy function in addition to seven Rosetta filter terms pertaining to interface characteristics [32]: number of residues participating in the interface, $\Delta\Delta G_{bind}$ of binding, Larence and Colman interface shape complementarity [31], side chain carbon-carbon contact counts, and a count of the buried unsatisfied hydrogen bond donors and accceptors at the interface. A full description of features is given in Table 1.

We created a custom data set by combining the KFC2a data set with these Rosetta features. Each row in the data set refers to an individual mutation and is labeled as a hotspot or non-hotspot residue based on the empirically determined change in binding free energy [38].

## 4 Feature Selection Strategy and Implementation

The presence of redundant and irrelevant features makes careful feature selection essential, especially for high-dimensional data [58]. We implement a *wrapper* method for feature selection, i.e. the features we select optimize the performance of an SVM classifier. As opposed to *filter* methods, where feature selection is independent of the learning algorithm, wrapper methods treat the learning algorithm as a black box

4

| KFC2a Features | Description |
|---|---|
| hydrophobicity | Fauchere and Pliska Hydrophobicity Index of Residue |
| DELTA_TOT | the buried solvent accessible surface area of an amino acid within the protein-protein interface |
| CORE_RIM | indicates a residue's position at the protein-protein interface, at the rim or core of the interface |
| POS_PER | rank order of CORE_RIM values |
| ROTS | total number of side chain rotatable single bonds |
| PLAST4 | measure the potential for local deformations within the protein interface, with 4Å cutoff |
| PLAST5 | measure the potential for local deformations within the protein interface, with 5Å cutoff |
| FADE_Point10 | number of interface grid points in the range 9-10 Angstrom, as calculated by FADE |

| Rosetta Feature | Description |
|---|---|
| buns3 | number of buried unsatisfied h-bond donors and acceptors at the protein-protein interface |
| ddg | Rosetta binding energy of the protein-protein interaction |
| dslf_fa13 | energy of disulfide bridges |
| fa_atr | attractive energy between two atoms on different residues separated by a distance d |
| a_dun | probability that a chosen rotamer is native-like given backbone $\phi, \psi$ angles |
| fa_elec | energy of interaction between two nonbonded charged atoms separated by a distance d |
| fa_intra_rep | repulsive energy between two atoms on the same residue separated by a distance d |
| fa_intra_sol_xover4 | Gaussian exclusion implicit solvation energy between protein atoms in the same residue |
| fa_rep | repulsive energy between two atoms on different residues separated by a distance d |
| fa_sol | Gaussian exclusion implicit solvation energy between protein atoms in different residues |
| hbond_bb_sc | energy of backbone–side-chain hydrogen bonds |
| hbond_lr_bb | energy of long-range hydrogen bonds |
| hbond_sc | energy of side-chain–side-chain hydrogen bonds |
| hbond_sr_bb | energy of short-range hydrogen bonds |
| interface_buried_sasa | buried solvent accessible surface area at the protein-protein interface |
| interface_contact | count of sidechain carbon-carbon contacts at the protein-protein interface |
| interface_sc | Larence and Colman shape complementarity at the protein-protein interface |
| interface_sc_int_area | buried solvent accessible surface area as computed for the Larence and Colman shape |
| lk_ball_wtd | orientation-dependent solvation of polar atoms assuming ideal water geometry |
| omega | backbone-dependent penalty for cis and trans $\omega$ dihedrals |
| p_aa_pp | probability of amino acid identity given backbone $\phi, \psi$ angles |
| pro_close | penalty for an open proline ring and proline $\omega$ bonding energy |
| rama_prepro | probability of backbone $\phi, \psi$ angles given the amino acid type |
| ref | reference energies for amino acid types |
| yhh_planarity | sinusoidal penalty for nonplanar tyrosine $\chi 3$ dihedral angle |

**Table 1** Descriptions of individual KFC2a and Rosetta features used in this study.

that outputs a performance metric associated with a given set of features, which is then optimized by adjusting the training parameters. This is a simple and powerful approach for feature selection [20].

More specifically, our goal is to select the set of features that optimizes the cross-validated F1-score of a Gaussian kernel SVM model. The F1-score is a performance metric for a binary classifier, and is defined as:

$$F1 = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}.$$

Here, recall is the true positive rate, and precision is the percentage of predicted hotspots that are true hotspots [44]. The F1-score is generally considered to be a more useful measure of performance than overall accuracy, especially when the negative class occurs more frequently than the positive class. To avoid over fitting, we optimize the F1-score using a five-fold cross-validation procedure. Such a validation procedure is intended to mimic the performance on independent test data, by successively eliminating subsets of the data, training on the remainder, and test-

ing on the withheld data. By iterating through folds of withheld data, an unbiased prediction can be made for each data point in the entire data set.

Although we seek the set of features giving us the global optimum of the cross-validated F1-score, it is computationally intractable to search the space of all possible subsets. Assuming we start with $d$ features, there are $2^d$ possible subsets, giving a complexity exponential in the number of feature combinations [27]. Even if we restrict our search to feature subsets of cardinality $k \ll d$, a brute force search would require that we train $O(d^k)$ models. For this reason, a greedy algorithm is introduced that selects the highest performing feature and then sequentially grows the feature set; this process is called *forward selection*. This reduces the complexity to $O(kd)$, though it is easy to construct examples where the feature set obtained is not optimal [20]. In this work we implement an approach which leverages the efficiency of forward selection while reducing the optimization error incurred by the greedy algorithm. We employ a semi-greedy algorithm which at each iteration adds in optimal pairs of features, giving complexity $O(kd^2)$. We also compare with the result of adding in optimal triples of features, which has complexity $O(kd^3)$. We will refer to the models obtained from forward selection with pairs and triples as Model 1 and Model 2 respectively.

Our SVMs were trained with a custom Python script using the `scikit-learn` library [43]. The features were scaled using the `scikit-learn` preprocessing to have zero mean and unit variance. For every model evaluation, a randomized grid search using a Gaussian RBF kernel and balanced class-weights, distributed to run eight jobs in parallel, was performed to find ideal estimates for the parameters $C$ and $\gamma$. For each parameter combination $(C, \gamma)$ in the random grid, the F1-score was estimated using five-fold cross-validation.

Our feature 2-way and 3-way forward selection strategies are not deterministic due to the computational challenges of fine-grained parameter search, so we ran these parameter searches five times to look for trends among the features discovered, rather than relying on a single run. We also ran Recursive Feature Elimination Cross Validation (RFECV), a reverse selection algorithm for use with linear SVMs for various values of the penalty parameter, $C$. We then trained non-linear classification models using the feature classes identified from the RFECV analysis.

## 5 Results and Discussion

### 5.1 Pairwise Relationships Among Features

First, let's look at the correlation matrix of features (Figure 2), which are individually described in Table 1. There are two groups of highly correlated features used for KFC2a. The first group (DELTA_TOT, CORE_RIM and POS_PER) are all related in some way to solvent accessibility, and the second group consists of two plasticity features calculated at different distance thresholds (4Å vs. 5Å). Some

6

(mostly weaker) internal correlations exist for Rosetta features. Between KFC2a and Rosetta, the non-trivial correlations were related to buried surface (KFC2a CORE_RIM vs. Rosetta interface_buried_sasa and interface_sc_int_area) and hydrophobicity (KFC2a hydrophobicity vs. Rosetta ref).

| | hydrophobicity | DELTA_TOT | CORE_RIM | POS_PER | ROTS | PLAST4 | PLAST5 | FADE_Point10 | buns3 | ddg | fa_atr | fa_dun | fa_elec | fa_intra_rep | fa_intra_sol_xover4 | fa_rep | fa_sol | hbond_bb_sc | hbond_sc | interface_buried_sasa | interface_contact | interface_sc | interface_sc_int_area | interface_unsat_hbond2 | lk_ball_wtd | omega | p_aa_pp | pro_close | rama_prepro | ref | yhh_planarity |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| hydrophobicity | 1.00 | -0.21 | 0.07 | -0.20 | 0.20 | 0.00 | 0.03 | 0.04 | -0.03 | 0.01 | 0.39 | 0.13 | -0.44 | -0.19 | 0.18 | -0.02 | 0.55 | -0.18 | -0.23 | 0.05 | -0.38 | -0.17 | -0.16 | -0.06 | 0.16 | 0.06 | 0.18 | 0.00 | -0.01 | -0.78 | -0.06 |
| DELTA_TOT | -0.21 | 1.00 | -0.48 | 0.96 | -0.39 | 0.20 | 0.12 | -0.34 | 0.03 | -0.14 | -0.31 | -0.03 | -0.11 | -0.10 | -0.08 | 0.04 | 0.13 | -0.17 | -0.13 | 0.09 | 0.35 | 0.33 | 0.27 | 0.12 | -0.06 | -0.06 | 0.04 | 0.02 | 0.07 | 0.21 | 0.19 |
| CORE_RIM | 0.07 | -0.48 | 1.00 | -0.51 | 0.15 | -0.24 | -0.24 | 0.02 | -0.03 | 0.04 | 0.29 | -0.10 | 0.08 | -0.10 | -0.12 | -0.06 | -0.11 | 0.04 | 0.03 | -0.60 | -0.38 | -0.26 | -0.53 | -0.09 | 0.02 | -0.09 | -0.09 | 0.06 | -0.09 | -0.08 | -0.09 |
| POS_PER | -0.20 | 0.96 | -0.51 | 1.00 | -0.38 | 0.19 | 0.11 | -0.39 | 0.03 | -0.14 | -0.30 | -0.01 | -0.11 | -0.13 | -0.05 | 0.03 | 0.15 | -0.17 | -0.14 | 0.11 | 0.35 | 0.33 | 0.28 | 0.12 | -0.05 | -0.07 | 0.04 | 0.00 | 0.06 | 0.19 | 0.18 |
| ROTS | 0.20 | -0.39 | 0.15 | -0.38 | 1.00 | -0.39 | -0.29 | 0.16 | -0.11 | -0.09 | 0.54 | -0.23 | 0.20 | -0.05 | -0.12 | -0.05 | -0.25 | 0.04 | 0.28 | -0.06 | -0.35 | -0.08 | -0.31 | -0.20 | 0.03 | -0.10 | -0.13 | -0.19 | -0.16 | -0.08 | -0.16 |
| PLAST4 | 0.00 | 0.20 | -0.24 | 0.19 | -0.39 | 1.00 | 0.81 | -0.03 | 0.11 | 0.14 | -0.31 | 0.17 | -0.13 | 0.05 | 0.17 | 0.08 | 0.18 | -0.04 | -0.14 | 0.18 | 0.23 | 0.06 | 0.20 | 0.08 | -0.01 | 0.22 | 0.19 | 0.13 | 0.23 | -0.09 | 0.08 |
| PLAST5 | 0.03 | 0.12 | -0.24 | 0.11 | -0.29 | 0.81 | 1.00 | 0.00 | 0.08 | 0.20 | -0.26 | 0.19 | -0.11 | 0.04 | 0.14 | 0.11 | 0.16 | -0.05 | -0.11 | 0.18 | 0.22 | 0.04 | 0.18 | 0.00 | 0.28 | 0.21 | 0.14 | 0.25 | -0.11 | 0.05 | |
| FADE_Point10 | 0.04 | -0.34 | 0.02 | -0.39 | 0.16 | -0.03 | 0.00 | 1.00 | -0.01 | 0.11 | 0.13 | -0.01 | 0.15 | 0.09 | 0.03 | 0.03 | -0.13 | 0.17 | 0.13 | 0.06 | -0.15 | -0.13 | -0.10 | -0.05 | 0.07 | 0.03 | -0.06 | 0.06 | -0.01 | -0.07 | -0.19 |
| buns3 | -0.03 | 0.03 | -0.03 | 0.03 | -0.11 | 0.11 | 0.08 | -0.01 | 1.00 | 0.15 | -0.18 | 0.03 | 0.10 | 0.05 | -0.03 | 0.07 | 0.03 | 0.17 | 0.12 | 0.12 | 0.13 | -0.18 | 0.26 | 0.29 | -0.12 | 0.03 | 0.09 | 0.05 | 0.05 | 0.06 | -0.03 |
| ddg | 0.01 | -0.14 | 0.04 | -0.14 | -0.09 | 0.14 | 0.20 | 0.11 | 0.15 | 1.00 | -0.10 | 0.09 | 0.04 | 0.04 | 0.03 | 0.32 | 0.02 | 0.18 | 0.06 | 0.08 | 0.05 | -0.21 | 0.00 | -0.02 | -0.08 | 0.26 | 0.04 | 0.10 | 0.01 | -0.03 | -0.05 |
| fa_atr | 0.39 | -0.31 | 0.29 | -0.30 | 0.54 | -0.31 | -0.26 | 0.13 | -0.18 | -0.10 | 1.00 | -0.32 | 0.19 | -0.22 | -0.18 | -0.06 | -0.33 | 0.10 | 0.16 | -0.19 | -0.56 | -0.25 | -0.44 | -0.21 | 0.11 | -0.12 | -0.09 | -0.04 | -0.15 | -0.31 | -0.13 |
| fa_dun | 0.13 | -0.03 | -0.10 | -0.01 | -0.23 | 0.17 | 0.19 | -0.01 | 0.03 | 0.09 | -0.32 | 1.00 | -0.31 | 0.15 | 0.40 | 0.11 | 0.37 | -0.15 | -0.22 | 0.13 | 0.15 | -0.02 | 0.21 | 0.07 | 0.02 | 0.11 | 0.14 | 0.10 | 0.09 | -0.24 | -0.03 |
| fa_elec | -0.44 | -0.11 | 0.08 | -0.11 | 0.20 | -0.13 | -0.11 | 0.15 | 0.10 | 0.04 | 0.19 | -0.31 | 1.00 | 0.08 | -0.21 | 0.07 | -0.76 | 0.41 | 0.56 | -0.10 | -0.04 | -0.15 | -0.09 | 0.01 | -0.32 | -0.07 | -0.15 | -0.07 | -0.05 | 0.43 | -0.18 |
| fa_intra_rep | -0.19 | -0.10 | -0.10 | -0.13 | -0.05 | 0.05 | 0.04 | 0.09 | 0.05 | 0.04 | -0.22 | 0.15 | 0.08 | 1.00 | 0.20 | 0.24 | -0.08 | 0.13 | 0.08 | 0.11 | 0.04 | -0.05 | 0.16 | 0.08 | 0.00 | 0.01 | -0.07 | -0.04 | 0.00 | 0.24 | 0.07 |
| fa_intra_sol_xover4 | 0.18 | -0.08 | -0.12 | -0.05 | -0.12 | 0.17 | 0.14 | 0.03 | -0.03 | 0.03 | -0.18 | 0.40 | -0.21 | 0.20 | 1.00 | 0.04 | 0.34 | -0.01 | -0.18 | 0.03 | 0.02 | -0.10 | 0.11 | -0.01 | -0.01 | 0.09 | 0.07 | 0.11 | 0.06 | -0.24 | 0.00 |
| fa_rep | -0.02 | 0.04 | -0.06 | 0.03 | -0.05 | 0.08 | 0.11 | 0.03 | 0.07 | 0.32 | -0.06 | 0.11 | 0.07 | 0.24 | 0.04 | 1.00 | 0.03 | 0.03 | 0.10 | 0.00 | 0.01 | -0.05 | -0.02 | 0.03 | 0.10 | 0.00 | 0.02 | 0.05 | -0.02 | -0.02 | -0.10 |
| fa_sol | 0.55 | 0.13 | -0.11 | 0.15 | -0.25 | 0.18 | 0.16 | -0.13 | 0.03 | 0.02 | -0.33 | 0.37 | -0.76 | -0.08 | 0.34 | 0.03 | 1.00 | -0.35 | -0.44 | 0.13 | 0.01 | 0.05 | 0.18 | 0.04 | 0.16 | 0.12 | 0.20 | 0.05 | 0.15 | -0.49 | 0.10 |
| hbond_bb_sc | -0.18 | -0.17 | 0.04 | -0.17 | 0.04 | -0.04 | -0.05 | 0.17 | 0.17 | 0.18 | 0.10 | -0.15 | 0.41 | 0.13 | -0.01 | 0.03 | -0.35 | 1.00 | 0.12 | 0.00 | 0.00 | -0.26 | -0.03 | 0.05 | -0.19 | 0.01 | -0.03 | -0.01 | 0.00 | 0.20 | -0.27 |
| hbond_sc | -0.23 | -0.13 | 0.03 | -0.14 | 0.28 | -0.14 | -0.11 | 0.13 | 0.12 | 0.06 | 0.16 | -0.22 | 0.56 | 0.08 | -0.18 | 0.10 | -0.44 | 0.12 | 1.00 | 0.04 | -0.04 | -0.13 | -0.02 | -0.03 | -0.33 | -0.02 | -0.11 | -0.12 | -0.04 | 0.23 | -0.20 |
| interface_buried_sasa | 0.05 | 0.09 | -0.60 | 0.11 | -0.06 | 0.18 | 0.18 | 0.06 | 0.12 | 0.08 | -0.19 | 0.13 | -0.10 | 0.11 | 0.03 | 0.00 | 0.13 | 0.00 | 0.04 | 1.00 | 0.21 | -0.06 | 0.62 | 0.13 | -0.07 | 0.12 | 0.13 | -0.04 | 0.09 | 0.05 | 0.06 |
| interface_contact | -0.38 | 0.35 | -0.38 | 0.35 | -0.35 | 0.23 | 0.22 | -0.15 | 0.13 | 0.05 | -0.56 | 0.15 | -0.04 | 0.04 | 0.02 | 0.01 | 0.01 | 0.00 | -0.04 | 0.21 | 1.00 | 0.35 | 0.36 | 0.21 | -0.11 | 0.09 | 0.01 | -0.04 | 0.11 | 0.27 | 0.07 |
| interface_sc | -0.17 | 0.33 | -0.26 | 0.33 | -0.08 | 0.06 | 0.04 | -0.13 | -0.18 | -0.21 | -0.25 | -0.02 | -0.15 | -0.05 | -0.10 | -0.05 | 0.05 | -0.26 | -0.13 | -0.06 | 0.35 | 1.00 | -0.19 | 0.01 | 0.21 | -0.03 | -0.03 | 0.01 | 0.03 | 0.11 | 0.13 |
| interface_sc_int_area | -0.16 | 0.27 | -0.53 | 0.28 | -0.31 | 0.20 | 0.18 | -0.10 | 0.26 | 0.00 | -0.44 | 0.21 | -0.09 | 0.16 | 0.11 | -0.02 | 0.18 | -0.03 | -0.02 | 0.62 | 0.36 | -0.19 | 1.00 | 0.24 | -0.14 | 0.10 | 0.15 | -0.01 | 0.11 | 0.15 | -0.01 |
| interface_unsat_hbond2 | -0.06 | 0.12 | -0.09 | 0.12 | -0.20 | 0.08 | 0.08 | -0.05 | 0.29 | 0.02 | -0.21 | 0.07 | 0.01 | 0.08 | 0.01 | 0.03 | 0.04 | 0.05 | -0.03 | 0.13 | 0.21 | 0.01 | 0.24 | 1.00 | -0.06 | 0.04 | 0.08 | 0.00 | 0.08 | 0.04 | -0.01 |
| lk_ball_wtd | 0.16 | -0.06 | 0.02 | -0.05 | 0.03 | -0.01 | 0.00 | 0.07 | -0.12 | -0.08 | 0.11 | 0.02 | -0.32 | 0.00 | -0.01 | -0.03 | 0.16 | -0.19 | -0.33 | -0.07 | -0.11 | 0.21 | -0.14 | -0.06 | 1.00 | -0.07 | 0.04 | 0.07 | -0.01 | -0.16 | 0.01 |
| omega | 0.06 | -0.06 | -0.09 | -0.07 | -0.10 | 0.22 | 0.21 | -0.07 | 0.03 | 0.26 | -0.12 | 0.11 | -0.07 | 0.01 | 0.09 | 0.00 | 0.12 | 0.01 | -0.02 | 0.12 | -0.06 | -0.03 | 0.10 | 0.04 | -0.07 | 1.00 | 0.25 | -0.02 | 0.11 | 0.05 | 0.08 |
| p_aa_pp | 0.18 | 0.04 | -0.09 | 0.04 | -0.13 | 0.19 | 0.21 | -0.06 | 0.09 | 0.04 | -0.09 | 0.14 | -0.15 | -0.07 | 0.07 | 0.00 | 0.20 | -0.03 | -0.11 | 0.13 | 0.01 | -0.03 | 0.15 | 0.08 | 0.04 | 0.25 | 1.00 | 0.08 | 0.63 | -0.21 | 0.04 |
| pro_close | 0.00 | 0.02 | 0.06 | 0.00 | -0.19 | 0.13 | 0.14 | 0.06 | 0.05 | 0.10 | -0.04 | 0.10 | -0.07 | -0.04 | 0.11 | 0.02 | 0.05 | -0.01 | -0.12 | -0.04 | -0.04 | 0.01 | -0.01 | 0.00 | 0.07 | -0.02 | 0.08 | 1.00 | 0.10 | -0.02 | 0.02 |
| rama_prepro | -0.01 | 0.07 | -0.09 | 0.06 | -0.16 | 0.23 | 0.25 | -0.01 | 0.05 | 0.01 | -0.15 | 0.09 | -0.05 | 0.00 | 0.06 | 0.02 | 0.07 | 0.00 | -0.04 | 0.09 | 0.11 | 0.03 | 0.11 | 0.08 | -0.01 | 0.11 | 0.63 | 0.10 | 1.00 | -0.09 | 0.02 |
| ref | -0.78 | 0.21 | -0.08 | 0.19 | -0.08 | -0.09 | -0.11 | -0.07 | 0.06 | -0.03 | -0.31 | -0.24 | 0.43 | 0.24 | -0.24 | -0.02 | -0.49 | 0.20 | 0.23 | 0.05 | 0.27 | 0.11 | 0.15 | 0.04 | -0.16 | 0.05 | -0.21 | -0.02 | -0.09 | 1.00 | 0.02 |
| yhh_planarity | -0.06 | 0.19 | -0.09 | 0.18 | -0.16 | 0.08 | 0.05 | -0.19 | -0.03 | -0.05 | -0.13 | -0.03 | -0.18 | 0.07 | 0.00 | -0.10 | 0.10 | -0.27 | -0.20 | 0.06 | 0.07 | 0.13 | 0.12 | -0.01 | 0.01 | 0.08 | 0.04 | 0.02 | 0.02 | 0.02 | 1.00 |

**Fig. 2** Pairwise correlation scores between features, with significant correlations shown in dark orange or blue.

## 5.2 Recursive Feature Elimination

In addition to our forward selection strategies, we looked at results from the Recursive Feature Elimination Cross Validation method. These models are somewhat easier to analyze, as they are linear models based only on the classification penalty parameter, $C$. Because it is hard to draw conclusions from a single run of the RFE method, we varied the value of $C$, and otherwise used the same SVM training parameters applied in the 2-way and 3-way feature selection strategies.

In Table 2, the feature rank for each feature is shown for varied values of $C$. Notice that the number of features required increases as the penalty for incorrect classification increases, which could lead to a more precise model, but also to over-

7

| C | 1 | 1.25 | 1.5 | 1.75 | 2 | 3 | 5 | 10 | 25 |
|---|---|---|---|---|---|---|---|---|---|
| hydrophobicity | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| DELTA_TOT | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| **CORE_RIM** | 1 | 1 | 12 | 11 | 1 | 1 | 1 | 1 | 1 |
| POS_PER | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| ROTS | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| **PLAST4** | 1 | 1 | 10 | 9 | 1 | 1 | 1 | 1 | 1 |
| ~~PLAST5~~ | 7 | 8 | 20 | 19 | 4 | 4 | 2 | 4 | 3 |
| FADE_Point10 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| **buns3** | 1 | 1 | 7 | 6 | 1 | 1 | 1 | 1 | 1 |
| **ddg** | 1 | 1 | 2 | 1 | 1 | 1 | 1 | 1 | 1 |
| ~~fa_atr~~ | 8 | 9 | 21 | 20 | 5 | 5 | 3 | 2 | 2 |
| **fa_dun** | 1 | 1 | 5 | 4 | 1 | 1 | 1 | 1 | 1 |
| ~~fa_elec~~ | 2 | 3 | 15 | 14 | 1 | 1 | 1 | 1 | 1 |
| ~~fa_intra_rep~~ | 3 | 4 | 16 | 15 | 1 | 1 | 1 | 1 | 1 |
| **fa_intra_sol_xover4** | 1 | 1 | 3 | 2 | 1 | 1 | 1 | 1 | 1 |
| ~~fa_rep~~ | 9 | 10 | 22 | 21 | 6 | 6 | 4 | 3 | 4 |
| fa_sol | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| **hbond_bb_sc** | 1 | 1 | 6 | 5 | 1 | 1 | 1 | 1 | 1 |
| **hbond_sc** | 1 | 1 | 8 | 7 | 1 | 1 | 1 | 1 | 1 |
| **interface_buried_sasa** | 1 | 2 | 14 | 13 | 1 | 1 | 1 | 1 | 1 |
| interface_contact | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| interface_sc | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| interface_sc_int_area | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| **interface_unsat_hbond2** | 1 | 1 | 9 | 8 | 1 | 1 | 1 | 1 | 1 |
| ~~lk_ball_wtd~~ | 5 | 6 | 18 | 17 | 2 | 2 | 1 | 1 | 1 |
| **omega** | 1 | 1 | 13 | 12 | 1 | 1 | 1 | 1 | 1 |
| p_aa_pp | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| **pro_close** | 1 | 1 | 4 | 3 | 1 | 1 | 1 | 1 | 1 |
| ~~rama_prepro~~ | 4 | 5 | 17 | 16 | 1 | 1 | 1 | 1 | 1 |
| ~~ref~~ | 6 | 7 | 19 | 18 | 3 | 3 | 1 | 1 | 1 |
| **yhh_planarity** | 1 | 1 | 11 | 10 | 1 | 1 | 1 | 1 | 1 |

**Table 2** Feature rankings returned from RFECV for each of the 31 features we considered, when examined for various values of *C* between 1.0 and 25.0. The features are grouped into three sets: disfavored features shown in strikeout text, lowC features in regular text, and features in the highC but not the lowC group in bold. The lowC group contains only those features that are top-ranked for all *C* values. The highC group includes everything except the disfavored features.

fitting. Some features such as hydrophobicity appear as a top-ranked feature for every choice of *C*, and we will call this feature group lowC, meaning they can perform well for low values of *C*. In this case, 7/8 KFC2a features and 16/23 Rosetta features are selected. Other features such as fa_rep are not highly chosen in any model, and we will refer to such examples as the disfavored feature group. For high *C* values, the RFECV method selects all but a few of the features, and this highC group contains everything but the disfavored features. In Table 2, the lowC features are those in plain text, and the highC features include those in plain and bold text. The features with strikeout text are those in the disfavored group.

Using the lowC and highC feature groups, we performed non-linear SVM training using *C* values in the preferred range (0.5 to 2 for lowC; 5 to 25 for highC). We also examined the KFC2a, Rosetta and all features using an exhaustive *C* and $\gamma$ parameter search, using the entire *C* parameter range (0.5 to 25) and finer sampling. These results are shown in Table 3, and we see that the lowC and highC groups return the highest ROC AUC scores when compared with other feature groups. The highC feature group returns the best result overall, with the lowC model performing worse on the positive (hotspot) class. The highC feature group performs similarly
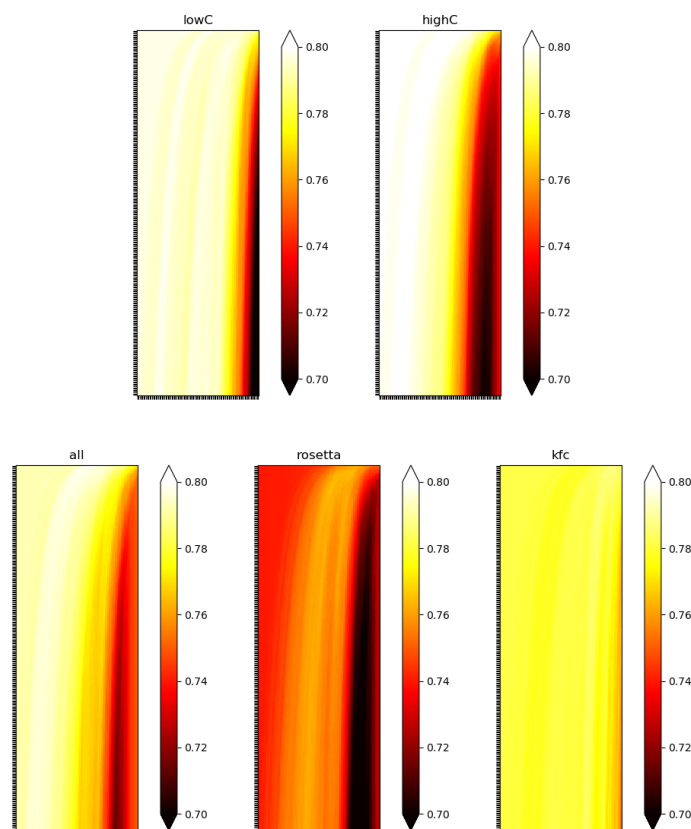
**Fig. 3** The figure shows ROC AUC scores for models trained on lowC features (top left) and highC features (top right), as a function of $C$ and $\gamma$. The $C$ values for lowC vary linearly along the y-axis from 0.5 to 25.0, with 300 samples, and the $\gamma$ values vary logarithmically along the x-axis from 0.00001 to 0.1, with 100 samples. In addition, models trained on all features (bottom left), Rosetta features (bottom middle) and KFC features (bottom right) are shown. The highC feature combination leads to the best chance of finding a high scoring model.

to KFC2a features and to the group of all on the positive class (TP+FN) and gives improved predictions on the negative class (TN+FP). The lowC feature group performs similarly to Rosetta features on the negative class, doing somewhat better on the positive class. In Figure 3, we see the results of exhaustive parameter search using non-linear SVMs with radial basis functions on the different feature groups.

Further examining Table 3, both the lowC and highC feature groups outperform groups consisting of KFC2a features, Rosetta features and all (KFC2a+Rosetta) features. The lowC group performs well on the negative class (TN+FP), while the highC group performs well on the positive class (TP+FN). When comparing the lowC to highC group, the former results in higher precision models and an accuracy similar to the Rosetta feature group, and the latter in higher recall models and specificity similar to KFC2a features or all features combined. The set of all features vs.

9

KFC2a features performs about the same when examined with regard to F1 score and ROC AUC. However, the groups lowC and highC, made by combining KFC2a and Rosetta features, demonstrate improvements in both F1 score and ROC AUC over other feature groups.

| | N | C | gamma | TP | TN | FP | FN | Ac | Pr | Re | Sp | F1 | AUC |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| KFC2a | 8 | 0.8277591973244147 | 0.03593813663804629000 | 104 | 328 | 150 | 36 | 0.699 | 0.409 | 0.743 | 0.686 | 0.528 | 0.7882 |
| Rosetta | 23 | 1.7290969899665551 | 0.00319926713779738460 | 90 | 362 | 116 | 50 | 0.731 | 0.437 | 0.643 | 0.757 | 0.520 | 0.7657 |
| all | 31 | 0.6638795986622074 | 0.00183073828029536980 | 102 | 334 | 144 | 38 | 0.706 | 0.415 | 0.729 | 0.699 | 0.528 | 0.7991 |
| lowC* | 10 | 0.6010101010101010 | 0.00265608778294668680 | 95 | 359 | 119 | 45 | 0.735 | 0.444 | 0.679 | 0.751 | 0.537 | 0.7990 |
| lowC | 10 | 0.5819397993311037 | 0.00265608778294668680 | 93 | 358 | 120 | 47 | 0.730 | 0.437 | 0.664 | 0.749 | 0.527 | 0.7990 |
| highC* | 23 | 5.6060606060606060 | 0.00031257158496882353 | 105 | 338 | 140 | 35 | 0.717 | 0.429 | 0.750 | 0.707 | 0.545 | 0.8031 |
| highC | 23 | 0.5000000000000000 | 0.00319926713779738460 | 102 | 336 | 142 | 38 | 0.709 | 0.418 | 0.729 | 0.7029 | 0.531 | 0.8060 |

**Table 3** The table shows optimized $C$ and $\gamma$ values, the size (N) of each feature group, and confusion matrix entries for cross-validated performance at the optimal $C$ and $\gamma$ values for that feature group. In addition, the Accuracy (Ac), Precision (Pr), Recall (Re), Specificity (Sp), F1-score (F1) and ROC AUC are given. The default search range is $C=(0.5,25)$ with 300 linear divisions and $\gamma=(10^{-5}, 10^{-1})$ with 100 logarithmic divisions; the lowC* and highC* results restrict the $C$-range from 0.5 to 2.0 for lowC* and from 2.0 to 25.0 for highC*. We see from the results that all features are better than simply KFC2a or Rosetta features; however, both the lowC and highC feature groups offer an improvement over all features.

### 5.3 2-way and 3-way Forward Selection of Features

Forward feature selection was performed by adding features in groups of 2 or 3 (2-way or 3-way, respectively) and then selecting the group that best maximizes performance. When doing the forward feature selection, random sampling was used to optimize $C$ and $\gamma$ values. Costly searches such as those shown in Figure 3 are not feasible as part of search strategies, but the large regions of good scores suggest random sampling can identify good solutions. To avoid drawing conclusions from a single run of a stochastic method, we ran our forward feature selection five times. In order to compare results with the highC and lowC groups previously discussed, we ran the algorithm with these restricted $C$ ranges, in addition to unconstrained random sampling.

It is important to remind the reader of several points about the search algorithm: the parameter search is coarse-grained, random, and based on F1 scores that are not cross-validated. These properties allow the search to run efficiently, but the optimized scores are not directly comparable to cross-validated F1 scores for the trained models described previously. Full parameter searches with cross-validation were conducted with parameters identified using forward selection, in order to compare performance directly with the other strategies, as will be demonstrated below. The Supplementary Materials include the non-cross-validated F1 scores and pairs of features chosen for 2-way feature addition.

For 2-way forward selection, the two features identified in the first iteration, for each of the five runs, were the KFC2a CORE_RIM and POS_PER features. Rosetta's interface_contact score was consistently chosen in second ranked initial pairs, as shown in Table 4.

| Pair Rank | non-CV F1 | Feature 1 | Feature 2 |
|---|---|---|---|
| 1 | 0.53112 | CORE_RIM | POS_PER |
| 2 | 0.52944 | DELTA_TOT | interface_contact |
| 3 | 0.52840 | CORE_RIM | interface_contact |

**Table 4** For a single run, at the first iteration, the top three results highlight alternative combinations of features that can perform well.

From the second iteration on, we take the best previous result (in this case CORE_RIM and POS_PER) and search for two additional features to add. KFC2a ROTS is a consistent choice in the second iteration, along with Rosetta omega. At the third iteration, Rosetta's fa_sol was commonly chosen. Subsequent iterations sample a wide range of KFC2a and Rosetta terms, using them to improve the F1 score. Table 5 shows the feature selection process for four iterations of five runs of the algorithm with $C$ sampled between 2 and 25.

| Feature Group | Feature Group 1 | | Feature Group 2 | | Feature Group 3 | | Feature Group 4 | |
|---|---|---|---|---|---|---|---|---|
| Run 1 | CORE_RIM | POS_PER | ROTS | omega | fa_sol | fa_elec | hbond_sc | rama_prepro |
| Run 2 | CORE_RIM | POS_PER | ROTS | ref | fa_sol | rama_prepro | lk_ball_wtd | rama_prepro |
| Run 3 | CORE_RIM | POS_PER | ROTS | omega | fa_sol | hbond_sc | fa_elec | lk_ball_wtd |
| Run 4 | CORE_RIM | POS_PER | ROTS | omega | interface_sc_int_area | hbond_sc | hbond_sc | lk_ball_wtd |
| Run 5 | CORE_RIM | POS_PER | ROTS | omega | fa_sol | DELTA_TOT | fa_elec | fa_intra_sol_xover4 |

**Table 5** Each group of columns shows the two features added to the model at each iteration, across five runs. As significant improvement in non-cross validated F1 score is observed in iterations 1-2. Around iterations 3-4, the models tend to plateau in performance.

Looking back to Table 2, we see that the KFC2a features POS_PER and ROTS are part of the highC group, and CORE_RIM is in the lowC group. Rosetta's fa_sol is in the highC group and omega in the lowC group. By the fourth iteration, forward selection has converged. Features like rama_prepro and lk_ball_wtd, which were eliminated by reverse selection, begin to appear as selected features but do not offer significant improvements to the model based on F1 scores (see Supplementary Materials.) At this point, feature selection becomes noisy, with many combinations of features offering insignificant improvements to the non-CV F1 score.

The fact that KFC2a features related to core vs rim position of a residue (CORE_RIM, POS_PER) were selected first is a good sign, as core-rim is well known to impact the likelihood of a hotspot. The choice of KFC2a ROTS is not surprising, as it likely reflects some entropic penalty in desolvating long side chains. The choice of Rosetta's omega is curious, reflecting backbone $\omega$ angles. However, we see that omega is somewhat correlated to the KFC2a plasticity features, suggesting a correlation for which the cause and effect may be more complex. An unusual omega angle in the
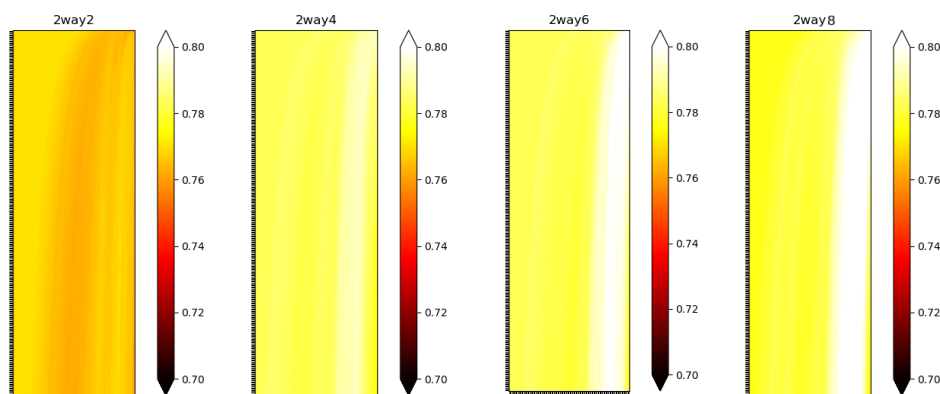
11

**Fig. 4** The figure shows ROC AUC scores for models trained on a progression of features identified using 2-way forward selection.

mutated structure generated by Rosetta may reflect significant reorganization of the local structure, which is implicitly measured by KFC2a using the plasticity features.

In order to compare the forward selection to reverse feature elimination, we trained models using the same parameter search applied to other examples, as displayed in Figure 3. The final ROC AUC for the 8-feature model corresponding to Run 5 of Table 5 was 0.8099, thus exceeding both the lowC and highC feature groups.

The progression of performance is shown in Table 6, after training the models using cross-validated scoring. In Figure 4, the results of a full parameter search are displayed, using the same bounds used to generate the results of Figure 3. Each iteration increases the zone in which high performing solutions can be obtained. Curiously, the favorable parameter region (high $\gamma$) that emerges is nearly opposite to that arising from the parameter searches shown in Figure 3. When $\gamma$ is high, the model is localized, and only nearby points influence the prediction at a given instance, whereas when $\gamma$ is small, many points influence the prediction at a single point.

|  | N | C | gamma | TP | TN | FP | FN | Ac | Pr | Re | Sp | F1 | AUC |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2-way | 2 | 0.5819397993311037 | 0.0015199110829529348 | 109 | 317 | 161 | 31 | 0.689 | 0.404 | 0.779 | 0.663 | 0.532 | 0.7706 |
| 2-way | 4 | 5.0066889632107030 | 0.0359381366380462900 | 117 | 321 | 157 | 23 | 0.709 | 0.427 | 0.836 | 0.672 | 0.565 | 0.7957 |
| 2-way | 6 | 4.5969899665551840 | 0.0475081016210279800 | 109 | 336 | 142 | 31 | 0.720 | 0.434 | 0.779 | 0.703 | 0.558 | 0.8034 |
| 2-way | 8 | 2.5484949832775920 | 0.0830217568131975300 | 114 | 344 | 134 | 26 | 0.741 | 0.460 | 0.814 | 0.720 | 0.588 | 0.8099 |

**Table 6** The table repeats the analysis of Table 3 using the results of 2-way forward selection. Results are shown after selecting 2 features, 4 features, 6 features and 8 features. A 5-fold cross validation was used to generate the predictions.

In addition to forward selection adding two features at a time, we performed 3-way forward selection, which showed a very similar progression in selecting features as the 2-way forward selection (Table 7). The 3-way selection showed more

variation in initial selection. While CORE_RIM, POS_PER and ROTS was chosen as a good combination, the top combination combined POS_PER with two Rosetta features, interface_sc and interface_sc_int_area. Using these three to seed the next iteration, the next three features chosen were DELTA_TOT, ROTS and omega, again largely following the preferences of the 2-way search. At the third iteration, buns3, fa_elec and interface_contact were added. The progression of features selected across five runs of the 3-way forward selection restricting $C$ between 2 and 25 are shown in Table 7. The training results and metrics for the 3-way feature selection were fairly comparable to those observed for the 2-way forward selection, hence we omit these details for brevity.

| Feature Group | Feature Group 1 | Feature Group 2 | | | Feature Group 3 | | |
|---|---|---|---|---|---|---|---|
| Run 1 | POS_PER  interface_sc  interface_sc_int_area | ROTS | DELTA_TOT | omega | buns3 | fa_elec | interface_contact |
| Run 2 | POS_PER  interface_sc  interface_sc_int_area | ROTS | hbond_sc | omega | interface_unsat_hbond2 | fa_dun | hydrophobicity |
| Run 3 | POS_PER  interface_sc  interface_sc_int_area | ROTS | DELTA_TOT | omega | interface_unsat_hbond2 | fa_dun | ref |
| Run 4 | POS_PER  interface_sc  interface_sc_int_area | ROTS | interface_unsat_hbond2 | omega | buns3 | fa_elec | hbond_sc |
| Run 5 | POS_PER  interface_sc  interface_sc_int_area | ROTS | DELTA_TOT | omega | buns3 | fa_elec | interface_contact |

**Table 7** Each group of columns shows the three features added to the model at each iteration, across 5 runs.

## 5.4 Conclusions

We examined two different strategies for feature selection on a data set for alanine mutagenesis hotspots. The features combined those of a popular hotspot model, KFC2a, and a widely used molecular modeling suite, Rosetta. Recursive feature elimination to define the highC group removed very few features from the combined data set, primarily features that were either redundant or uninformative. The lowC group further reduced the set of features, generally achieving better specificity in prediction than the highC group but lower recall/sensitivity.

An alternate strategy applied forward 2-way and 3-way selection with a random search for optimal $C$ and $\gamma$ parameters. These methods converged after just a few iterations, producing a small number of features with significant information content for answering the classification question. The random parameter search was remarkably consistent at finding the top parameter group, CORE_RIM and POS_PER, both of which relate to the "buriedness" of an amino acid within the interface.

The overall preferences for 3-way search versus 2-way search are very similar, but some of the top choices changed. In particular, some Rosetta features that were overshadowed by the dominant choice of CORE_RIM and POS_PER were more prominent in the 3-way search. For example, the CORE_RIM feature, no longer chosen in the initial iteration of 3-way forward selection, is somewhat correlated with both interface_sc and interface_sc_int_area, which were chosen instead. This shows the value of considering 3-way forward selection in addition to 2-way selec-

tion; in particular, the 3-way selection allowed the first iteration to choose a slightly more accurate combination to cover core-rim effects using three terms.

While showing overall consistency in feature selection, the results also demonstrate that many feature combinations can lead to comparable models. There is not a clearly "right" combination, and the results do not allow us to rank order the importance of any individual feature.

# References

[1] Abram ME, Ferris AL, Shao W, Alvord WG, Hughes SH (2010) Nature, position, and frequency of mutations made in a single cycle of HIV-1 replication. Journal of virology 84(19):9864–9878

[2] Ahmad S, Keskin O, Mizuguchi K, Sarai A, Nussinov R (2010) CCRXP: exploring clusters of conserved residues in protein structures. Nucleic Acids Research 38(Web Server issue):W398–401

[3] Alford RF, Leaver-Fay A, Jeliazkov JR, OMeara MJ, DiMaio FP, Park H, Shapovalov MV, Renfrew PD, Mulligan VK, Kappel K, Labonte JW, Pacella MS, Bonneau R, Bradley P, Dunbrack RL, Das R, Baker D, Kuhlman B, Kortemme T, Gray JJ (2017) The Rosetta All-Atom Energy Function for Macromolecular Modeling and Design. Journal of Chemical Theory and Computation 13(6):3031–3048

[4] Assi SA, Tanaka T, Rabbitts TH, Fernandez-Fuentes N (2010) PCRPi: Presaging Critical Residues in Protein interfaces, a new computational tool to chart hot spots in protein interfaces. Nucleic Acids Research 38(6):e86

[5] Bahram F, von der Lehr N, Cetinkaya C, Larsson LG (2000) c-Myc hot spot mutations in lymphomas result in inefficient ubiquitination and decreased proteasome-mediated turnover. Blood 95(6):2104–2110

[6] Ben-Shimon A, Eisenstein M (2010) Computational mapping of anchoring spots on protein surfaces. Journal of Molecular Biology 402(1):259–277

[7] Bogan AA, Thorn KS (1998) Anatomy of hot spots in protein interfaces. Journal of Molecular Biology 280(1):1–9

[8] Bradshaw RT, Patel BH, Tate EW, Leatherbarrow RJ, Gould IR (2011) Comparing experimental and computational alanine scanning techniques for probing a prototypical protein-protein interaction. Protein engineering, design & selection : PEDS 24(1-2):197–207

[9] Chevalier A, Silva DA, Rocklin GJ, Hicks DR, Vergara R, Murapa P, Bernard SM, Zhang L, Lam KH, Yao G, et al (2017) Massively parallel de novo protein design for targeted therapeutics. Nature 550(7674):74–79

[10] Christianini N, Shawe-Taylor J (2000) An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods. Cambridge University Press, Cambridge, England

[11] Chuang GY, Mehra-Chaudhary R, Ngan CH, Zerbe BS, Kozakov D, Vajda S, Beamer LJ (2010) Domain motion and interdomain hot spots in a multidomain enzyme. Protein Science 19(9):1662–1672

[12] Cukuroglu E, Gursoy A, Keskin O (2012) HotRegion: a database of predicted hot spot clusters. Nucleic Acids Research 40(Database issue):D829–33

[13] Darnell SJ, Page D, Mitchell JC (2007) An automated decision-tree approach to predicting protein interaction hot spots. Proteins-Structure Function and Bioinformatics 68(4):813–823

[14] Darnell SJ, LeGault L, Mitchell JC (2008) KFC Server: interactive forecasting of protein interaction hot spots. Nucleic Acids Research 36(Web Server issue):W265–9

15

[15] DeLano W (2002) Unraveling hot spots in binding interfaces: progress and challenges. Curr Opin Struct Biol 12(1):14–20

[16] Donald JE, Zhu H, Litvinov RI, DeGrado WF, Bennett JS (2010) Identification of interacting hot spots in the beta3 integrin stalk using comprehensive interface design. Journal of Biological Chemistry 285(49):38,658–38,665

[17] Fischer A, Arunachalam K, Mangual V, Bakhru S, Russo R, Huang D, Paczkowski M, Lalchandani V, Ramachandra C, Ellison B, Galer S, Shapley J, Fuentes E, Tsai J (2003) The binding interface database (BID): a compilation of amino acid hot spots in protein interfaces. Bioinformatics 19(11):1453–1454

[18] Grosdidier S, Fernandez-Recio J (2008) Identification of hot-spot residues in protein-protein interactions by computational docking. BMC bioinformatics 9:447

[19] Guerois R, Nielsen JE, Serrano L (2002) Predicting changes in the stability of proteins and protein complexes: a study of more than 1000 mutations. Journal of Molecular Biology 320(2):369–387

[20] Guyon I, Elisseeff A (2003) An introduction to variable and feature selection. Journal of machine learning research 3(Mar):1157–1182

[21] Halperin I, Wolfson H, Nussinov R (2004) Protein-protein interactions; coupling of structurally conserved residues and of hot spots across interfaces. Implications for docking. Structure (London, England : 1993) 12(6):1027–1038

[22] Jones S, Thornton JM (1997) Analysis of protein-protein interaction sites using surface patches. Journal of Molecular Biology 272(1):121–132

[23] Kelly L, Fukushima H, Karchin R, Gow JM, Chinn LW, Pieper U, Segal MR, Kroetz DL, Sali A (2010) Functional hot spots in human ATP-binding cassette transporter nucleotide binding domains. Protein Science 19(11):2110–2121

[24] Keskin O, Ma BY, Nussinov R (2005) Hot regions in protein-protein interactions: The organization and contribution of structurally conserved hot spot residues. Journal of Molecular Biology 345(5):1281–1294

[25] Kim D (2009) A feature-based approach to modeling protein-protein interaction hot spots. Nucleic Acids Research 37(8):2672–2687

[26] Koga N, Tatsumi-Koga R, Liu G, Xiao R, Acton TB, Montelione GT, Baker D (2012) Principles for designing ideal protein structures. Nature 491(7423):222–227

[27] Kohavi R, John GH (1997) Wrappers for feature subset selection. Artificial intelligence 97(1-2):273–324

[28] Kortemme TT, Baker DD (2002) A simple physical model for binding energy hot spots in protein-protein complexes. Proceedings of the National Academy of Sciences of the United States of America 99(22):14,116–14,121

[29] Krüger DM, Gohlke H (2010) DrugScorePPI webserver: fast and accurate in silico alanine scanning for scoring protein-protein interactions. Nucleic Acids Research 38(Web Server issue):W480–6

[30] Kuhlman B, Dantas G, Ireton GC, Varani G, Stoddard BL, Baker D (2003) Design of a novel globular protein fold with atomic-level accuracy. science 302(5649):1364–1368

[31] Lawrence MC, Colman PM (1993) Shape complementarity at protein/protein interfaces. Journal of molecular biology 234(4):946–950

[32] Leaver-Fay A, Tyka M, Lewis SM, Lange OF, Thompson J, Jacak R, Kaufman K, Renfrew PD, Smith CA, Sheffler W, Davis IW, Cooper S, Treuille A, Mandell DJ, Richter F, Ban YEA, Fleishman SJ, Corn JE, Kim DE, Lyskov S, Berrondo M, Mentzer S, Popovi? Z, Havranek JJ, Karanicolas J, Das R, Meiler J, Kortemme T, Gray JJ, Kuhlman B, Baker D, Bradley P (2011) Rosetta3: an object-oriented software suite for the simulation and design of macromolecules. Methods in Enzymology 487:545–74

[33] Lichtarge O, Bourne HR, Cohen FE (1996) An evolutionary trace method defines binding surfaces common to protein families. Journal of Molecular Biology 257(2):342–358

[34] Lise S, Archambeau C, Pontil M, Jones DT (2009) Prediction of hot spot residues at protein-protein interfaces by combining machine learning and energy-based methods. BMC bioinformatics 10:365

[35] Liu Q, Li J (2010) Protein binding hot spots and the residue-residue pairing preference: a water exclusion perspective. BMC bioinformatics 11:244

[36] Meenan NAG, Sharma A, Fleishman SJ, Macdonald CJ, Morel B, Boetzel R, Moore GR, Baker D, Kleanthous C (2010) The structural and energetic basis for high selectivity in a high-affinity protein-protein interaction. Proceedings of the National Academy of Sciences of the United States of America 107(22):10,080–10,085

[37] Metternich R, Tarzia G (2010) "Hot spots" in medicinal chemistry. ChemMedChem 5(8):1159–1162

[38] Moal IH, Fernández-Recio J (2012) SKEMPI: a Structural Kinetic and Energetic database of Mutant Protein Interactions and its use in empirical models. Bioinformatics 28(20):2600–2607

[39] Nayak J, Naik B, Behera H (2015) A comprehensive survey on support vector machine in data mining tasks: applications & challenges. International Journal of Database Theory and Application 8(1):169–186

[40] Ofran Y, Rost B (2007) Protein-protein interaction hotspots carved into sequences. PLoS computational biology 3(7):e119

[41] Ovchinnikov S, Park H, Kim DE, DiMaio F, Baker D (2017) Protein structure prediction using Rosetta in CASP12. Proteins: Structure, Function, and Bioinformatics

[42] Ozbabacan SEA, Gursoy A, Keskin O, Nussinov R (2010) Conformational ensembles, signal transduction and residue hot spots: application to drug discovery. Current opinion in drug discovery & development 13(5):527–537

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

[44] Powers DM (2011) Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. International Journal of Machine Learning Technology 2(1):37–63

[45] Pulim V, Berger B, Bienkowska J (2008) Optimal contact map alignment of protein-protein interfaces. Bioinformatics 24(20):2324–2328

[46] Rajamani D, Thiel S, Vajda S, Camacho CJ (2004) Anchor residues in protein-protein interactions. Proceedings of the National Academy of Sciences of the United States of America 101(31):11,287–11,292

[47] Res I, Lichtarge O (2005) Character and evolution of protein-protein interfaces. Physical Biology 2(2):S36–S43

[48] Segura J, Fernandez-Fuentes N (2011) PCRPi-DB: a database of computationally annotated hot spots in protein interfaces. Nucleic Acids Research 39(Database issue):D755–60

[49] Segura Mora J, Assi SA, Fernandez-Fuentes N (2010) Presaging critical residues in protein interfaces-web server (PCRPi-W): a web server to chart hot spots in protein interfaces. Plos One 5(8):e12,352

[50] Shulman-Peleg A, Shatsky M, Nussinov R, Wolfson HJ (2007) Spatial chemical conservation of hot spot interactions in protein-protein complexes. BMC biology 5:43

[51] Shulman-Peleg A, Shatsky M, Nussinov R, Wolfson HJ (2008) MultiBind and MAPPIS: webservers for multiple alignment of protein 3D-binding sites and their interactions. Nucleic Acids Research 36(Web Server issue):W260–4

[52] Tharakaraman K, Robinson LN, Hatas A, Chen YL, Siyue L, Raguram S, Sasisekharan V, Wogan GN, Sasisekharan R (2013) Redesign of a cross-reactive antibody to dengue virus with broad-spectrum activity and increased in vivo potency. Proc Natl Acad Sci USA 110(17):E1555–64

[53] Tuncbag N, Gursoy A, Keskin O (2009) Identification of computational hot spots in protein interfaces: combining solvent accessibility and inter-residue potentials improves the accuracy. Bioinformatics 25(12):1513–1520

[54] Tuncbag N, Keskin O, Gursoy A (2010) HotPoint: hot spot prediction server for protein interfaces. Nucleic Acids Research 38(Web Server issue):W402–6

[55] Ui M, Tanaka Y, Tsumuraya T, Fujii I, Inoue M, Hirama M (2010) Structural and energetic hot-spots for the interaction between a ladder-like polycyclic ether and the anti-ciguatoxin antibody 10C9Fab. Molecular bioSystems

[56] Ward JM, Gorenstein NM, Tian J, Martin SF, Post CB (2010) Constraining binding hot spots: NMR and molecular dynamics simulations provide a structural explanation for enthalpy-entropy compensation in SH2-ligand binding. Journal of the American Chemical Society 132(32):11,058–11,070

[57] Xia JF, Zhao XM, Song J, Huang DS (2010) APIS: accurate prediction of hot spots in protein interfaces by combining protrusion index with solvent accessibility. BMC bioinformatics 11:174

[58] Yu L, Liu H (2004) Efficient feature selection via analysis of relevance and redundancy. Journal of machine learning research 5(Oct):1205–1224

[59] Zhu X, Mitchell JC (2011) KFC2: a knowledge-based hot spot prediction method based on interface solvation, atomic density, and plasticity features. Proteins-Structure Function and Bioinformatics 79(9):1097–0134