BIOINFORMATICS ORIGINAL PAPER Vol. 24 no. 18 2008, pages 2002–2009
doi:10.1093/bioinformatics/btn353

Structural bioinformatics

Accurate prediction of stability changes in protein mutants by combining machine learning with structure based computational mutagenesis

Majid Masso and Iosif I. Vaisman*

Laboratory for Structural Bioinformatics, Department of Bioinformatics and Computational Biology, George Mason University, 10900 University Blvd., MSN 5B3, Manassas, VA 20110, USA

Received on November 26, 2007; revised on June 1, 2008; accepted on July 9, 2008

Advance Access publication July 16, 2008

Associate Editor: Anna Tramontano

ABSTRACT

Motivation: Accurate predictive models for the impact of single amino acid substitutions on protein stability provide insight into protein structure and function. Such models are also valuable for the design and engineering of new proteins. Previously described methods have utilized properties of protein sequence or structure to predict the free energy change of mutants due to thermal ($\Delta\Delta G$) and denaturant ($\Delta\Delta G^{H2O}$) denaturations, as well as mutant thermal stability (ΔT_m), through the application of either computational energy-based approaches or machine learning techniques. However, accuracy associated with applying these methods separately is frequently far from optimal.

Results: We detail a computational mutagenesis technique based on a four-body, knowledge-based, statistical contact potential. For any mutation due to a single amino acid replacement in a protein, the method provides an empirical normalized measure of the ensuing environmental perturbation occurring at every residue position. A feature vector is generated for the mutant by considering perturbations at the mutated position and it's ordered six nearest neighbors in the 3-dimensional (3D) protein structure. These predictors of stability change are evaluated by applying machine learning tools to large training sets of mutants derived from diverse proteins that have been experimentally studied and described. Predictive models based on our combined approach are either comparable to, or in many cases significantly outperform, previously published results.

Availability: A web server with supporting documentation is available at http://proteins.gmu.edu/automute

Contact: ivaisman@gmu.edu

1 INTRODUCTION

Experimental assessments of changes in protein stability, which result from amino acid residue substitutions, represent an important area of research in biochemistry and molecular biology. A more complete understanding of the factors that influence protein folding can be developed through the analysis of this information, including the persistence or elimination of non-covalent contacts (hydrophobic and van der Waals interactions; hydrogen and ionic bonds) upon mutation as well as the secondary structure and solvent accessibility of each substituted position. Such studies are also useful for the elucidation of catalytic residues in enzymes and for developing a clearer understanding of the functional roles of other residue positions in proteins. Additionally, these data are keys for designing new proteins that possess desired attributes, such as specific levels of stability or enzymatic activity, avoidance of protein aggregation and enhancement or diminution of protein–protein interactions or DNA binding capability. Given the abundance and significance of applications, and due to the costly nature with respect to both time and expense of performing exhaustive wet-lab mutagenesis studies, accurate predictive models of protein stability changes upon single point substitutions are in great demand.

Predictions have been carried out by some groups through the application of force fields based on physical effective energy functions derived from molecular mechanics (Lazaridis and Karplus, 2000; Moult, 1997; Wang *et al.*, 1996), which are often combined with molecular dynamics or Monte Carlo simulations (Duan and Kollman, 1998; Duan *et al.*, 1998; Kollman *et al.*, 2000; Pitera and Kollman, 2000; Prevost *et al.*, 1991). However, these approaches are computationally expensive, limiting their practical utility to small sets of protein mutants. Alternatively, methods described in the literature that utilize force fields based on pseudo-energy functions have been effectively applied to the stability analysis of large mutant datasets. These techniques are derived either from knowledge-based statistical potentials (Gilis and Rooman, 1996, 1997; Hoppe and Schomburg, 2005; Kwasigroch *et al.*, 2002; Meyerguz *et al.*, 2007; Ota *et al.*, 1995; Parthiban *et al.*, 2007; Topham *et al.*, 1997; Wang *et al.*, 1998; Zhou and Zhou, 2002), or from physical descriptions of possible interactions combined with experimentally obtained empirical data (Bordner and Abagyan, 2004; Guerois *et al.*, 2002; Saraboji *et al.*, 2006).

Recently, supervised classification and regression machine learning techniques have been used successfully to predict the direction (increased or decreased) and value of mutant stability change, respectively (Capriotti *et al.*, 2004, 2005a, b; Cheng *et al.*, 2006; Frenz, 2005; Huang *et al.*, 2006, 2007). These approaches are capable of utilizing local and non-local interactions that impact protein stability by learning complex nonlinear functions based on large training sets of protein mutants with experimentally measured stability changes. Independent variables (i.e. predictors) include

*To whom correspondence should be addressed.

information about the mutation as well as the protein sequence or structure, and each protein mutant is encoded as an ordered vector of these attributes.

In this article, we have merged a four-body, knowledge-based statistical potential and machine learning techniques, yielding a novel and accurate method for predicting stability changes in proteins upon single point mutations. Briefly, the potential was developed by abstracting every amino acid to a point in a training set of protein structures and applying a computational geometry technique known as Delaunay tessellation to each discretized structure. The points associated with the residues are utilized as vertices for generating a compact tiling of the space into tetrahedral simplices, which objectively identify all quadruplets of nearest-neighbor residues in the protein. A computational mutagenesis based on this multibody potential yields a normalized environmental change (EC) score measuring perturbations at every residue position in a protein structure resulting from a single amino acid substitution. Among the ordered attributes defining each protein mutant for analysis with machine learning algorithms, we have included the EC scores of the mutated position and its six nearest neighbors in a 3-dimensional (3D) protein structure defined by the tessellation. By employing machine learning tools and large protein mutant datasets that were used in previously published reports; we illustrate the significantly improved performance of models designed with these predictors. To our knowledge, this is the first reported application combining attributes obtained explicitly from a knowledge-based potential with supervised classification and regression machine learning techniques, for the prediction of mutant protein stability changes.

2 METHODS

2.1 Delaunay tessellation and the four-body potential

A non-homologous training set of over 1400 high-resolution crystallographic protein structures with low primary sequence identity was selected from the Protein Data Bank (PDB) (Berman *et al.*, 2000) for developing the knowledge-based potential. Each structure was represented as a discrete set of points in 3D space, corresponding to the C_α atomic coordinates of each of the constituent amino acid residues in the protein. A computational geometry construct known as Delaunay tessellation, applied to each discretized protein structure, generates an aggregate of non-overlapping, space-filling, irregular tetrahedral simplices by utilizing the points as vertices (Singh *et al.*, 1996; Vaisman *et al.*, 1998). Hence, this approach objectively defines quadruplets of nearest neighbor amino acids in a protein structure based on the residue identities represented by the vertices of the simplices formed by a protein tessellation (Fig. 1). As an added measure to ensure physically meaningful interactions, we only considered simplices in protein tessellations for which all six edge-lengths were <12 Å. The Quickhull algorithm was used to perform the Delaunay tessellation of each protein structure (Barber *et al.*, 1996). An in-house suite of Java and Perl programs was used for preprocessing of the PDB structure files, which included checking for the absence of gaps, and post-processing of the Quickhull output data.

Without regard to order, there are 8855 distinct quadruplet types that can be formed from the 20 amino acids naturally occurring in proteins (Singh *et al.*, 1996; Vaisman *et al.*, 1998). For each quadruplet, we determined the observed proportion of simplices among all training set protein tessellations whose vertices represented the four amino acids. We also computed a rate expected by chance for each quadruplet based on a multinomial reference distribution that utilized the individual amino acid frequencies in the training set proteins. Modeled after the inverse Boltzmann law, an empirical potential of quadruplet interaction (log-likelihood score) was calculated as a logarithm

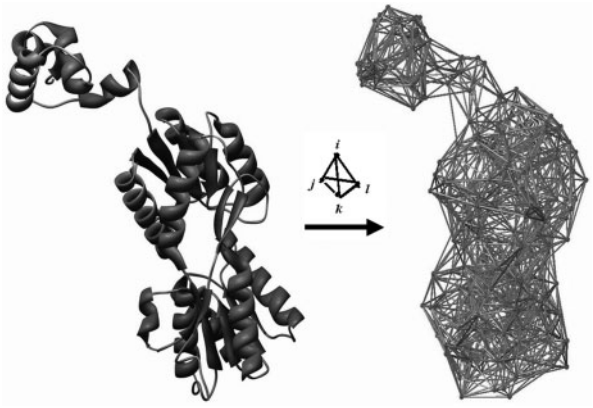

Fig. 1. Delaunay tessellation (right) of a monomer of the *Escherichia coli lac* repressor subject to a 12 Å edge-length cutoff. Ribbon diagram (left) produced with Chimera (Pettersen *et al.*, 2004).

of the ratio of observed to expected values. The four-body statistical potential is defined as the collection of 8855 quadruplet (or simplex) types along with each of their respective log-likelihood scores (Singh *et al.*, 1996; Vaisman *et al.*, 1998).

Given any tessellated protein structure of interest, the four-body potential can be used to calculate a *residue environment score* for each amino acid position, by locally summing the log-likelihood scores of all simplices that share the C_α coordinate of the residue position as a vertex. The vector of residue environment scores for all amino acid positions in a protein structure, ordered by the primary sequence, is referred to as a *3D-1D potential profile* (Bowie *et al.*, 1991; Masso and Vaisman, 2003; Masso *et al.*, 2006).

2.2 Computational mutagenesis and predictors of mutant protein stability changes

Given the abstraction of amino acids to single points in 3D for the purpose of tessellation, coupled with the robustness of Delaunay tessellation with respect to small shifts in the point coordinates as well as unavailability of solved structures for all single point protein mutants, potential profiles for protein mutants were calculated by using the wild type (wt) structure tessellation. For each single point mutant, the amino acid identity was altered at the C_α coordinate representing the mutated position, and the log-likelihood scores of the simplices sharing the point were recomputed. In the wt potential profile vector, such a substitution only alters the residue environment scores of the mutated position itself, as well as all positions whose C_α coordinates participate as vertices in simplices with the C_α coordinate of the mutated position (Masso and Vaisman, 2003; Masso *et al.*, 2006).

The *residual profile* of a mutant is defined as the difference between the mutant and wt protein 3D-1D potential profile vectors, and the value of each residual profile component is referred to as an *EC score*. Specifically, the residual profile components with nonzero EC scores identify the mutated position and all of its structural nearest neighbors defined by the tessellation, and the values of these nonzero EC scores signify the degree of environmental perturbation at those positions caused by the specific type of residue replacement at the mutated position. Due to its significance, the EC score at the mutated position component in a mutant residual profile is referred to as the *residual score* of the mutant.

Assuming that each C_α coordinate in a protein structure tessellation participates in at least two simplices as the only shared vertex, every amino acid position is expected to have a minimum of six nearest neighbor residues defined by the tessellation. An attribute vector was generated for every single point protein mutant under consideration, consisting of the residual score (i.e. EC score of the mutated position), followed by the EC scores of the six nearest neighbors, ordered by the 3D Euclidean distances of the neighboring

C_α coordinates from that of the mutated position. Additional attributes that we evaluated include the identities of the wt and substituted amino acids at the mutated position, the ordered identities of the amino acids at the six nearest neighbor positions and the ordered primary sequence distances between the nearest neighbors and the mutated position. Finally, in order to perform direct comparisons with published reports, we also included where appropriate the thermodynamic parameters of pH and temperature under which experimental mutagenesis and stability measurements were performed, in addition to relative accessibility and secondary structure of the mutated residue, as provided by those studies and described more fully in subsequent sections. For the protein structures under consideration, tessellations were performed only on single chains of multimeric proteins, and for NMR structures, only a single model was tessellated unless a minimized average structure was available.

2.3 Datasets

2.3.1 The Capriotti datasets The dataset S1948 consists of 1948 distinct single amino acid substitutions in 58 proteins with solved structures in the PDB, which are also uniformly distributed among the four major SCOP structural classifications (http://scop.mrc-lmb.cam.ac.uk/scop/) (Capriotti *et al.*, 2005b). Mutants with experimentally measured and published free energy changes due to thermal denaturations ($\Delta\Delta G$), for which the experimental conditions of pH and temperature were also reported, were obtained from the ProTherm database (Bava *et al.*, 2004). An RSA value (Relative Solvent Accessible Area) was also calculated for each mutant using the DSSP program (Capriotti *et al.*, 2004, 2005b; Kabsch and Sander, 1983). We identified two structures with missing residues (PDB codes 1CAH and 1TPK), which eliminated 12 mutants from our set. We also eliminated one degenerate mutant from a protein with numerous other bona fide mutants (PDB code 1PGA, mutation T53T). Finally, we eliminated 10 mutants for which the mutated position had fewer than six nearest neighbors based on Delaunay tessellation with a 12 Å distance cutoff, three of which constituted the only mutants from a particular protein (PDB code 1ARR). Hence, our version of S1948 consists of 1925 single point mutants in 55 proteins.

The dataset S1615 is similarly defined and consists of 1615 distinct single point mutants in 42 proteins with solved structures (Capriotti *et al.*, 2004). We eliminated 11 mutants for which the mutated position had fewer than six nearest neighbors, which again resulted in the loss of all three mutants associated with 1ARR. Our version of S1615 therefore consists of 1604 single point mutants in 41 proteins. Finally, the dataset S388 is a subset of S1615 that contains only experiments performed under physiological conditions (temperature: 20–40°C and pH: 6–8) and consists of 388 mutants in 17 protein structures (Capriotti *et al.*, 2004). Likewise, our version of S388 consists of 382 mutants in 16 protein structures.

2.3.2 The Gromiha datasets The dataset S1791 consists of 1791 distinct single point mutants with experimentally determined thermal stability (ΔT_m), as well as secondary structure (helix, strand, coil, turn) and accessible surface area (ASA) ($0 \leq ASA \leq 2$, buried; $2 < ASA \leq 50$, partially buried; $ASA > 50$, exposed; M. Gromiha, personal communication) at the mutated positions, and were obtained from the ProTherm database (Saraboji *et al.*, 2006). Due to the elimination of PDB structures with missing residues, as well as mutations at positions with fewer than six nearest neighbors, our version of S1791 consists of 1749 single point mutants.

The datasets S1396 and S2204 consist of 1396 and 2204 distinct single point mutants with experimentally determined free energy change due to thermal ($\Delta\Delta G$) and denaturant ($\Delta\Delta G^{H2O}$) denaturations, respectively, as well as secondary structure and ASA at the mutated positions, collected from the ProTherm database (Huang *et al.*, 2007; Saraboji *et al.*, 2006). Since 174 of the mutants in the S1396 dataset are not associated with any solved protein structure, and due to elimination of an additional 18 mutants as a result of either missing residues in PDB structures or mutated positions with fewer than six nearest neighbors, our version of S1396 consists of 1204 single point mutants. Although all mutants in the S2204 dataset are associated with solved structures, 242 mutants were eliminated due to missing residues in PDB structures and mutated positions with fewer than six nearest neighbors. Hence, our version of S2204 consists of 1962 single point mutants.

2.4 Machine learning algorithms and evaluation of performance

The S1948 dataset was previously used to train support vector machine (SVM) classification and regression models, utilizing a radial basis function (RBF) kernel and a 20-fold cross-validation (CV) testing procedure (Capriotti *et al.*, 2005b). In the case of classification only the sign of $\Delta\Delta G$ for each mutant was considered for prediction ($\Delta\Delta G < 0$, decreased stability; $\Delta\Delta G \geq 0$, increased stability), while the actual $\Delta\Delta G$ values were used for SVM regression. The S1615 and S388 datasets were previously evaluated with neural network (NN) (Capriotti *et al.*, 2004) and SVM (Cheng *et al.*, 2006) classifiers, again based on a 20-fold CV procedure. Additionally, SVM regression and 20-fold CV was applied to the S1615 dataset (Cheng *et al.*, 2006), and S1615 was recently used to train iPTREE (Huang *et al.*, 2006), a C4.5 decision tree classifier (Quinlan, 1993) augmented by the Adaboost adaptive boosting algorithm (Freund and Schapire, 1996), along with 10-fold CV. Capriotti *et al.* (2004) also applied NN with 20-fold CV on the S388 dataset; however, they subsequently used S388 as a validation test set to obtain prediction accuracies for other existing methods and compared their 20-fold CV results on S388 to predictions made by the other methods (i.e. a comparison of different testing procedures). Similarly, Cheng *et al.* (2006) applied SVM with 20-fold CV on the S388 dataset and compared his results to those of Capriotti and the other methods. Instead of machine learning, an average assignment method, combined with leave-one-out CV (known as the jackknife) and self-consistency (also referred to as 'back-check,' i.e. training and testing with the full dataset) testing procedures, was recently implemented for classifying mutants in the S1791, S1396 and S2204 datasets (Saraboji *et al.*, 2006). On the other hand, a classification and regression tree (CART) algorithm (Breiman *et al.*, 1984), using 4- and 5-fold CV testing, was also recently applied to the S1396 and S2204 datasets (Huang *et al.*, 2007).

In cases where SVM classification or regression and iPTREE were applied, we used the same algorithms and testing procedures in order to directly compare the performance of our predictors to those that have been reported. SVM uses a kernel function to nonlinearly map training set examples into a higher-dimensional feature space, where an optimal separating hyperplane is constructed that provides a maximal margin of separation between examples from two differing classes and corresponds to a nonlinear decision boundary in the original space. However, we also implemented alternative machine learning tools, including random forest (RF) learning (Breiman, 2001) and reduced error pruning tree (REPTree) regression, in order to try and improve further on performance. RF utilizes bagging (bootstrap aggregating) to generate multiple bootstrapped datasets, each of which trains a classification tree by random selection of a fixed-size subset of the available predictors for splitting at each node, and predictions are made by majority vote. Similarly, we compared Adaboost/C4.5 to the average assignment method on the S1791 dataset, while RF was compared to average assignment on the S1396 and S2204 datasets, all using the self-consistency and jackknife performance measures. Boosting with Adaboost is similar to bagging, in that it uses voting to combine the output of multiple models (C4.5 decision trees in this case); however, boosting is iterative, so that new models are influenced by the performance of previous ones, and model contributions are weighted by their performance. Finally, we compared our performance of RF and REPTree using 4- and 5-fold CV to that of CART on the S1396 and S2204 datasets. All algorithms were implemented using the Weka suite of machine learning tools (Frank *et al.*, 2004).

The following information was calculated for evaluating algorithm performance and making comparisons with previously published reports. Depending on the dataset, mutants are classified as 'increased stability' ('+' and 'stabilizing' are terms also used) if ΔT_m, $\Delta\Delta G$, or $\Delta\Delta G^{H2O} \geq 0$;

otherwise the mutants belong to the 'decreased stability' (i.e. '–' or 'destabilizing') class. With the understanding that TP (TN) = total number of correctly predicted 'increased stability' ('decreased stability') mutants, and FN (FP) = total number of respectively misclassified mutants, the overall accuracy is defined as Q = (TP + TN)/(TP + TN + FP + FN), which may also be reported as a percentage. Also, for the 'increased stability' class, S(+) = sensitivity = TP/(TP + FN) and P(+) = precision = TP/(TP + FP), while for the 'decreased stability' class, S(−) = TN/(TN + FP) and P(−) = TN/(TN + FN). The remaining measures that follow were calculated due to their robustness with respect to unequal class distributions. The balanced error rate is defined as BER = 0.5 × [FN/(FN + TP) + FP/(FP + TN)], Matthew's correlation coefficient is given by

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

and AUC refers to area under the receiver operating characteristic (ROC) curve, a plot of true positive rate (i.e. sensitivity) versus false positive rate (i.e. 1-specificity) for a particular class. Class predictions are based on probabilities assigned to mutants by a decision function associated with each machine learning tool. ROC for a given class is obtained by ranking mutants according to their predicted probabilities for belonging to the class, then plotting successive points based on the actual class memberships of mutants that lie above a steadily decreasing predicted probability threshold.

3 RESULTS AND DISCUSSION

3.1 Predictions with the S1948 dataset

Among the 1925 mutants in our version of this dataset, there were 582 'increased stability' mutants with a mean residual score of 0.58 ± 1.38, and 1343 'decreased stability' mutants with a mean residual score of −0.61 ± 1.87. Application of the F-test shows that the class variances differ significantly, and a t-test verifies that the difference in the mean residual scores of the classes is statistically significant ($P = 4.81 \times 10^{-51}$). Figure 2 suggests that non-conservative amino acid substitutions (Dayhoff *et al.*, 1978) are primarily responsible for the trend, a property that we previously observed among large datasets of protein-specific single point mutants with experimentally determined activity classes (Masso *et al.*, 2006).

Each of the 1925 mutants was represented as an attribute vector consisting of the following features:

(1) wt and replacement amino acid identities at the mutated position;

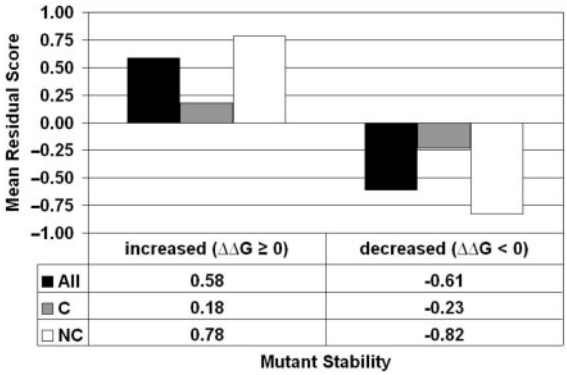

Fig. 2. Structure (mean residual score)—function (stability change) correlation among the mutants in the S1948 dataset.

(2) Residual score (i.e. EC score at the mutated position obtained from the residual profile vector);

(3) For the six nearest neighbors of the mutated position, defined by the Delaunay tessellation of the structure and ordered by 3D Euclidean distance, we include: EC scores obtained from the residual profile vector, amino acid identities at those positions and their primary sequence distances away from the mutated position;

(4) Computed RSA and experimental temperature, pH and ΔΔG (sign for classification, actual value for regression).

The features in (1) and (4) are common to both our attribute vectors as well as those of Capriotti *et al.* (2005), while the features in (2) and (3) are specifically based on our four-body potential. In order to make a direct comparison with the previous study, we trained an SVM classifier using an RBF kernel and a 20-fold CV testing procedure. The results using all the mutant attributes (Table 1) reveal an increase of 0.04 (5%) in accuracy over Capriotti's SVM, assisted significantly by a 0.14 (25%) sensitivity increase in the '+' class and leading to a 0.08 (29%) drop in BER and a 0.10 (20%) increase in MCC. We also applied a RF classifier with 20-fold CV, where parameters chosen included growing 50 trees from bootstrapped datasets and selecting five random attributes (from among all the attributes described above) to split at every node of every tree. As shown in Table 1, accuracy improved by 0.06 (8%) over Capriotti's SVM as a result of significant increases in all sensitivity and precision measures, resulting in a 0.10 (36%) drop in BER and a 0.15 (29%) increase in MCC. When only seven mutant attributes were provided, corresponding to the mutant residual score and the ordered EC scores of the six nearest neighbors, our RF classifier (50 trees, four random attributes/node) again outperformed the SVM classifier of Capriotti *et al.* (2005) with respect to all reported measures (Table 1).

To further investigate the robustness of high performance of SVM and RF using all predictors, ROC curves were plotted based on 20-fold CV (Fig. 3), for which AUC values were 0.86 (SVM) and 0.91 (RF). Controls in Figure 3 are based on initially performing a random shuffling of the 582 '+' and 1343 '–' stability class labels among the 1925 mutants in the dataset prior to applying SVM and RF. Next, we performed 10 iterations of 20-fold CV to assess the degree of variability between runs. Mean values of Q, BER and MCC were 0.84 ± 0.003, 0.21 ± 0.004 and 0.60 ± 0.01 using SVM, and 0.86 ± 0.003, 0.19 ± 0.003 and 0.65 ± 0.01 using RF. We also performed 10 stratified random split iterations to investigate the performance of trained models on separate validation test sets. With every iteration, 66% of the mutants in the original dataset were randomly selected for training a model, and the remaining mutants that were held-out (34% of the original dataset) and blindly predicted by the trained classifier. Mean values of Q, BER and MCC were

Table 1. Comparison of 20-fold CV prediction performance on S1948

Method	Q	S(+)	P(+)	S(−)	P(−)	BER	MCC
RF (all attributes)	0.86	0.70	0.81	0.93	0.88	0.18	0.66
RF (EC scores)	0.82	0.61	0.75	0.91	0.84	0.24	0.55
SVM (all attributes)	0.84	0.70	0.75	0.90	0.87	0.20	0.61
Capriotti (SVM)	0.80	0.56	0.73	0.91	0.83	0.28	0.51

0.80 ± 0.01, 0.25 ± 0.02 and 0.52 ± 0.03 using SVM, and 0.84 ± 0.02, 0.21 ± 0.02 and 0.61 ± 0.04 using RF. Lastly, application of RF with the jackknife method on the dataset gave Q = 0.86, BER = 0.19, MCC = 0.65 and AUC = 0.91.

In order to gauge the impact of training set size on accuracy, learning curves were prepared (Fig. 4). For each size increment, 10 mutant sets were generated by selecting with replacement from among the full dataset, and 20-fold CV accuracy was obtained with each set using SVM and RF. Points on the learning curves represent the mean 20-fold CV accuracy at each increment, and error bars represent ±1 SD. The graphs reveal how the RF and SVM classifiers clearly benefit by learning from larger training sets.

Finally, mutant class labels obtained from the sign of ΔΔG were replaced with the actual ΔΔG values in the mutant attribute vectors, and we applied SVM regression (SVMreg) to our dataset, using an RBF kernel and 20-fold CV for direct comparison. As shown in Table 2, the correlation (r) of the predicted and experimental data is 0.76 with a standard error of 1.2 kcal/mol based on our predictors, a significant improvement over the results of Capriotti *et al.* (2005). We also applied REPTree regression in conjunction with a bagging (bootstrapped aggregating) procedure and 20-fold CV, whereby 50 bootstrapped training sets equal in size to the original dataset were utilized, and the final mutant predictions were obtained by averaging.

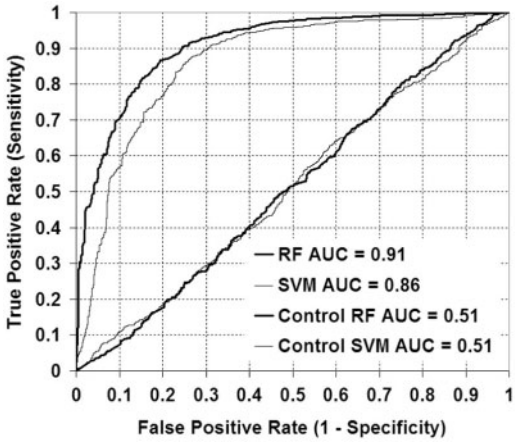

Fig. 3. ROC curves based on 20-fold CV classification of S1948 mutants with RF and SVM (all attributes). Controls were obtained by randomly shuffling class labels among the mutants prior to classification.

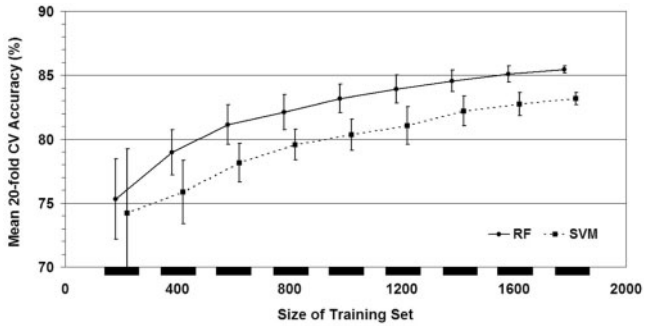

Fig. 4. RF and SVM (all attributes) based learning curves.

The results (Table 2 and Fig. 5) reveal further improvement over our SVMreg method, with $r = 0.79$ and standard error of 1.1 kcal/mol.

3.2 Predictions with the S1615 and S388 datasets

Our version of S1615 was initially used for training an RF classifier with 100 trees and five randomly chosen attributes/node selected from among all the attributes described at the start of the previous section. All of our RF performance measures associated with a 20-fold CV (Table 3; center row, top section) displayed significant improvements over those obtained by Cheng *et al.* (2006) based on an SVM approach with an RBF kernel and the use of sequence and structure (ST) mutant attributes, as well as results obtained by Capriotti *et al.* (2004) based on an NN approach along with attributes defined by sequence and structure (Table 3; center and bottom rows, bottom section). These previous studies also included computed RSA and experimental temperature and pH conditions for each mutant, justifying our inclusion of these attributes for direct comparison to those methods. As done in the previous section, in order to highlight the strongly predictive capabilities of attributes derived strictly from our four-body potential, we considered only seven attributes (EC scores of the mutated position and the ordered six nearest neighbors) and applied RF (100 trees, four random attributes/node) and 20-fold CV. The results (Table 3; bottom row, top section) again reveal improvement over the NN approach.

In order to perform a direct comparison with iPTREE (Huang *et al.*, 2006) on the S1615 dataset (Table 3; top row, bottom section), we used all of our attributes and applied Weka implementations of the AdaBoost algorithm (400 iterations) in conjunction with the C4.5 decision tree algorithm (all default parameters, pruning with confidence factor C = 0.25), and we performed a 10-fold CV

Table 2. Comparison of regression algorithms on S1948

Method	r	Standard error (kcal/mol)	Regression line
REPTree (all attributes)	0.79	1.1	$y = 0.5357x - 0.4376$
SVMreg (all attributes)	0.76	1.2	$y = 0.6287x - 0.3124$
Capriotti (SVMreg)	0.71	1.3	$y = 0.5223x - 0.4705$

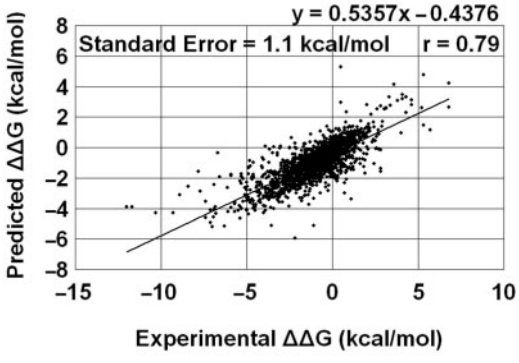

Fig. 5. Correlation plot of the experimental and predicted values of ΔΔG based on the REPTree method.

Table 3. Comparison of classification algorithms on S1615

Method	Q	S(+)	P(+)	S(−)	P(−)	BER	MCC
AdaBoost/C4.5 (all attributes)	0.872	0.722	0.796	0.929	0.898	0.17	0.67
RF (all attributes)	0.862	0.713	0.771	0.919	0.894	0.18	0.65
RF (EC scores)	0.823	0.634	0.697	0.895	0.865	0.24	0.55
Huang (iPTREE)	0.871	0.668	0.836	0.949	0.881	0.19	0.67
Huang (iPTREE)	0.871	0.668	0.836	0.949	0.881	0.19	0.67
Cheng (SVM/ST)	0.847	0.671	0.733	0.910	0.883	0.21	0.60
Capriotti (NN)	0.810	0.520	0.710	0.910	0.830	0.29	0.49

Table 4. Comparison of regression algorithms on S1615

Method	r	Standard error (kcal/mol)	Regression line
REPTree (all attributes)	0.77	1.09	$y=0.5126x-0.5228$
SVMreg (all attributes)	0.74	1.14	$y=0.5876x-0.4221$
Cheng (SVM regression/ST)	0.75	1.09	—

testing procedure. Our results based on the combined Adaboost/C4.5 algorithms (which are iPTREE by definition) and tabulated in the top row of Table 3, reveal that we achieved performance levels comparable to those of iPTREE, with identical correlation coefficients of 0.67.

Replacing the sign of $\Delta\Delta G$ with the actual values for each mutant in S1615, we next turned to application of REPTree and SVM regression with 20-fold CV, using all of our attributes so that we could make a direct comparison to the SVM regression with RBF kernel approach used by Cheng *et al.* (2006). As described earlier, REPTree was implemented with bagging by averaging results from 50 bootstrapped datasets, and an RBF kernel was utilized with SVM regression. The correlation of predicted and experimental $\Delta\Delta G$ values based on REPTree and SVM regression were 0.77 and 0.74, respectively, with standard errors of 1.09 and 1.14 kcal/mol (Table 4), which are comparable to the previously published result.

Lastly, we trained two RF models by using the subset of mutants remaining in S1615 after the removal of the S388 mutants (referred to as S1227 in Table 5). The first RF model (100 trees, five random attributes/node) was trained using mutant feature vectors that incorporated all the attributes described earlier, while the second RF model (100 trees, four random attributes/node) utilized only seven EC scores (mutant residual score and ordered EC scores of the six nearest neighbors). The trained models were subsequently used to blindly predict the mutants in S388, which served as a validation test set. As detailed in the Methods, Capriotti *et al.* (2004) and Cheng *et al.* (2006) reported 20-fold CV results on S388 and compared them to predictions made on S388, as a validation test set, by three other existing methods. It is not clear why they chose to compare

Table 5. Comparison of classification algorithms on S388

Method	Q	S(+)	P(+)	S(−)	P(−)	BER	MCC
RF*, S388 (all attributes)	0.87	0.36	0.42	0.94	0.92	0.35	0.31
RF, S1227 (all attributes)	0.89	0.42	0.56	0.96	0.93	0.31	0.43
RF, S1227 (EC scores)	0.83	0.47	0.33	0.88	0.93	0.33	0.30
Cheng* (SVM/ST)	0.86	0.31	0.40	0.93	0.91	0.38	0.27
Cheng* (SVM/ST)	0.86	0.31	0.40	0.93	0.91	0.38	0.27
Capriotti* (NN)	0.87	0.21	0.44	0.96	0.90	0.42	0.25
PoPMuSiC[a]	0.85	0.25	0.33	0.93	0.90	0.41	0.20
DFIRE[b]	0.68	0.44	0.18	0.71	0.90	0.43	0.11
FOLDX[c]	0.75	0.56	0.26	0.78	0.93	0.33	0.25

*Based on 20-fold CV; all others use S388 as a separate test set for existing models
[a]http://babylone.ulb.ac.be (Gilis and Rooman, 1997; Kwasigroch *et al.*, 2002)
[b]http://sparks.informatics.iupui.edu (Zhou and Zhou, 2002)
[c]http://fold-x.embl-heidelberg.de (Guerois *et al.*, 2002)

20-fold CV results to those based on using S388 as a separate test set; however, our approach of training an RF model without using any of the S388 data, and then using that model to predict the S388 data, allows for an appropriate comparison with predictions made by the three other methods. We also applied RF (100 trees, five random attributes/node) with 20-fold CV to S388 in order to compare our results with those of Capriotti *et al.* (2004) and Cheng *et al.* (2006). The results of all predictions on S388 are detailed in Table 5. The data specific to the utilization of S388 as a validation test set clearly indicate that our trained RF, S1227 (all attributes) model yields a substantial overall improvement in prediction performance, summarized by a 6% BER decrease and 72% MCC increase relative to the best values obtained by the other three methods. Additionally, the RF, S1227 (EC scores) model results suggest that these seven attributes alone provide a substantial portion of the information required for training an accurate predictive model.

3.3 Predictions with the Gromiha datasets (S1791, S1396 and S2204)

Saraboji *et al.* (2006) applied an average assignment method for the classification of mutants in the S1791 dataset as stabilizing or destabilizing. For each of the 380 possible (wt, new) amino acid pairs, the experimental ΔT_m values of all mutants in the dataset defined by the pair were averaged, and this average was assigned to each of those mutants. Accuracy and correlation based on this classification method were evaluated by comparing the experimental and assigned stability values. They reported significant improvement by initially segregating the S1791 mutants into clusters, based on either secondary structure or ASA, and performing the average assignment to each cluster. We implemented Adaboost (20 iterations) together with C4.5 (default parameters, C = 0.25) and evaluated the performance on our mutant attribute vectors, using the full dataset as well as each of the grouped subsets (Table 6). In all cases, except for accuracy of the jackknife applied to mutations in strands, our approach significantly outperformed that of average assignment

Table 6. Comparison of Adaboost/C4.5 with average assignment on S1791

	Number of Mutants	Q (%)		MCC	
		Back-check	Jack-knife	Back-check	Jack-knife
2° Str					
Helix	871 (872)	95 (82)	82 (75)	0.89 (0.68)	0.60
Strand	308 (326)	96 (90)	81 (87)	0.91 (0.79)	0.51
Coil	570 (593)	98 (85)	82 (80)	0.95 (0.65)	0.59
Location					
Buried	414 (429)	99 (89)	91 (84)	0.98 (0.64)	0.69
Partial	778 (794)	97 (84)	82 (80)	0.93 (0.63)	0.57
Exposed	557 (568)	93 (79)	75 (72)	0.86 (0.73)	0.49
Full dataset	1749 (1791)	96 (78)	81 (71)	0.91 (0.60)	0.56

Average assignment data are in parentheses for comparison.

Table 7. Comparison of RF and REPTree with CART on S1396 and S2204

Fold CV	S1396 (ΔΔG)			S2204 ($\Delta\Delta G^{H2O}$)		
	MCC	Q (%)	MAE	MCC	Q (%)	MAE
4	0.6137 (0.5884)	82.31 (80.59)	0.9468 (1.1010)	0.4109 (0.4159)	81.45 (80.22)	1.0755 (1.3816)
5	0.6200 (0.6093)	82.56 (81.08)	0.9407 (1.0794)	0.3780 (0.4401)	80.63 (80.10)	1.0634 (1.3684)

CART data are in parentheses for comparison. MCC and Q were obtained by applying RF, and MAE was obtained by applying REPTree.

[jackknife correlation data not provided in Saraboji *et al.* (2006)]. As expected, the RF algorithm generated Q and MCC performance measures similar to those shown for Adaboost/C4.5 in Table 6 (data not shown).

Likewise, jackknife accuracy values (Q) using the average assignment method applied to S1396 were reported as 74% (the full set), 82% (overall, after clustering by secondary structure) and 84% (overall, after clustering by ASA); corresponding values for the S2204 dataset were 80, 83 and 82%. For comparison, we implemented RF (50 trees, five random attributes/node) on the full S1396 and S2204 datasets using our attributes and obtained jackknife accuracy values of 83% for each case, which is significantly higher than average assignment accuracies on each of these full datasets and is comparable to the overall accuracy results for average assignment based on clustering by ASA or secondary structure.

Huang *et al.* (2006) applied CARTs on S1396 and S2204, and prediction accuracy, correlation and mean absolute error (MAE) were reported based on 4- and 5-fold CV as well as the use of 48 predictor attributes for each mutant. We implemented RF (50 trees. five random attributes/node) with 4- and 5-fold CV in order to obtain classifier accuracy (Q) and correlation (MCC), and we similarly implemented REPTree (100 bagged iterations) regression to calculate MAE based on the difference between experimental and predicted stability change (Table 7). A comparison of the results reveals that RF performed slightly better on S1396 while CART performed slightly better on S2204; however, REPTree performed better on both datasets.

3.4 Significance of the combined approach for making predictions

Energy-based methods learn functions for making predictions by fitting a linear combination of (pseudo-) energy data obtained from experimental or knowledge-based approaches. On the other hand, machine learning techniques learn complex nonlinear functions for making predictions that depend on a common set of measured attributes for all examples in a dataset. Currently, published reports on applications of machine learning to the prediction of activity or stability changes in proteins due to single residue substitutions have all utilized as attributes information about protein sequence or structure, or evolutionary information, without making use of the more strongly correlative data obtained from energy-based methods. Here, for the first time, we have made explicit use of data obtained from a four-body, knowledge-based, statistical contact potential, by defining a computational mutagenesis procedure leading to mutant attributes that quantify the environmental perturbations occurring at the mutated position and its six closest neighbors. In some instances, by leveraging the power of machine learning on as few as these seven energy-based attributes, we have outperformed techniques that utilize a much larger number of predictors. In all cases, our results are at least comparable to those of previous studies when analogous sequence, structure or experimental parameter attributes are included. Unlike other energy-based approaches, the simplicity with which the four-body potential and computational mutagenesis can be applied makes it ideally suited for use in conjunction with machine learning techniques as a way to develop improved models for predicting activity and stability changes in protein mutants.

ACKNOWLEDGEMENTS

The authors thank Andrew Carr for preparing the tessellation graphic, Rita Casadio for encouraging us to test our approach on the Capriotti *et al.* datasets and Michael Gromiha for providing us with the corrected ASA class thresholds in his datasets.

Confict of Interest: none declared.

REFERENCES

Barber,C.B. *et al.* (1996) The quickhull algorithm for convex hulls. *ACM T. Math. Software*, **22**, 469–483.

Bava,K.A. *et al.* (2004) ProTherm, version 4.0: thermodynamic database for proteins and mutants. *Nucleic Acids Res.*, **32**, D120–D121.

Berman,H.M. *et al.* (2000) The protein data bank. *Nucleic Acids Res.*, **28**, 235–242.

Bordner,A.J. and Abagyan,R.A. (2004) Large-scale prediction of protein geometry and stability changes for arbitrary single point mutations. *Proteins*, **57**, 400–413.

Bowie,J.U. *et al.* (1991) A method to identify protein sequences that fold into a known three-dimensional structure. *Science*, **253**, 164–170.

Breiman,L. (2001) Random forests. *Machine Learning*, **45**, 5–32.

Breiman,L. *et al.* (1984) *Classification and Regression Trees*. Wadsworth International Group, Belmont, California.

Capriotti,E. *et al.* (2004) A neural-network-based method for predicting protein stability changes upon single point mutations. *Bioinformatics*, **20** (Suppl. 1), I63–I68.

Capriotti,E. *et al.* (2005a) Predicting protein stability changes from sequences using support vector machines. *Bioinformatics*, **21** (Suppl. 2), ii54–ii58.

Capriotti,E. *et al.* (2005b) I-Mutant2.0: predicting stability changes upon mutation from the protein sequence or structure. *Nucleic Acids Res.*, **33**, W306–W310.

Cheng,J. *et al.* (2006) Prediction of protein stability changes for single-site mutations using support vector machines. *Proteins*, **62**, 1125–1132.

Dayhoff,M.O. *et al.* (eds) (1978) *A Model for Evolutionary Change in Proteins*. National Biomedical Research Foundation, Washington, DC.

Duan,Y. and Kollman,P.A. (1998) Pathways to a protein folding intermediate observed in a 1-microsecond simulation in aqueous solution. *Science*, **282**, 740–744.

Duan,Y. *et al.* (1998) The early stage of folding of villin headpiece subdomain observed in a 200-nanosecond fully solvated molecular dynamics simulation. *Proc. Natl Acad. Sci. USA*, **95**, 9897–9902.

Frank,E. *et al.* (2004) Data mining in bioinformatics using Weka. *Bioinformatics*, **20**, 2479–2481.

Frenz,C.M. (2005) Neural network-based prediction of mutation-induced protein stability changes in Staphylococcal nuclease at 20 residue positions. *Proteins*, **59**, 147–151.

Freund,Y. and Schapire,R.E. (1996) Experiments with a new boosting algorithm. In *Proceedings of the International Conference on Machine Learning,* pp. 148–156. Morgan Kaufmann, San Francisco.

Gilis,D. and Rooman,M. (1996) Stability changes upon mutation of solvent-accessible residues in proteins evaluated by database-derived potentials. *J. Mol. Biol.*, **257**, 1112–1126.

Gilis,D. and Rooman,M. (1997) Predicting protein stability changes upon mutation using database-derived potentials: solvent accessibility determines the importance of local versus non-local interactions along the sequence. *J. Mol. Biol.*, **272**, 276–290.

Guerois,R. *et al.* (2002) Predicting changes in the stability of proteins and protein complexes: a study of more than 1000 mutations. *J. Mol. Biol.*, **320**, 369–387.

Hoppe,C. and Schomburg,D. (2005) Prediction of protein thermostability with a direction- and distance-dependent knowledge-based potential. *Protein Sci.*, **14**, 2682–2692.

Huang,L.T. *et al.* (2006) Knowledge acquisition and development of accurate rules for predicting protein stability changes. *Comput. Biol. Chem.*, **30**, 408–415.

Huang,L.T. *et al.* (2007) Prediction of protein mutant stability using classification and regression tool. *Biophys. Chem.*, **125**, 462–470.

Kabsch,W. and Sander,C. (1983) Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, **22**, 2577–2637.

Kollman,P.A. *et al.* (2000) Calculating structures and free energies of complex molecules: combining molecular mechanics and continuum models. *Acc. Chem. Res.*, **33**, 889–897.

Kwasigroch,J.M. *et al.* (2002) PoPMuSiC, rationally designing point mutations in protein structures. *Bioinformatics*, **18**, 1701–1702.

Lazaridis,T. and Karplus,M. (2000) Effective energy functions for protein structure prediction. *Curr. Opin. Struct. Biol.*, **10**, 139–145.

Masso,M. and Vaisman,I.I. (2003) Comprehensive mutagenesis of HIV-1 protease: a computational geometry approach. *Biochem. Biophys. Res. Commun.*, **305**, 322–326.

Masso,M. *et al.* (2006) Computational mutagenesis studies of protein structure-function correlations. *Proteins*, **64**, 234–245.

Meyerguz,L. *et al.* (2007) The network of sequence flow between protein structures. *Proc. Natl Acad. Sci. USA*, **104**, 11627–11632.

Moult,J. (1997) Comparison of database potentials and molecular mechanics force fields. *Curr. Opin. Struct. Biol.*, **7**, 194–199.

Ota,M. *et al.* (1995) Desk-top analysis of the structural stability of various point mutations introduced into ribonuclease H. *J. Mol. Biol.*, **248**, 733–738.

Parthiban,V. *et al.* (2007) Structural analysis and prediction of protein mutant stability using distance and torsion potentials: role of secondary structure and solvent accessibility. *Proteins*, **66**, 41–52.

Pettersen,E.F. *et al.* (2004) UCSF Chimera–a visualization system for exploratory research and analysis. *J. Comput. Chem.*, **25**, 1605–1612.

Pitera,J.W. and Kollman,P.A. (2000) Exhaustive mutagenesis in silico: multicoordinate free energy calculations on proteins and peptides. *Proteins*, **41**, 385–397.

Prevost,M. *et al.* (1991) Contribution of the hydrophobic effect to protein stability: analysis based on simulations of the Ile-96→Ala mutation in barnase. *Proc. Natl Acad. Sci. USA*, **88**, 10880–10884.

Quinlan,R. (1993) *C4.5: Programs for Machine Learning*. Morgan Kaufman Publishers, San Mateo, CA.

Saraboji,K. *et al.* (2006) Average assignment method for predicting the stability of protein mutants. *Biopolymers*, **82**, 80–92.

Singh,R.K. *et al.* (1996) Delaunay tessellation of proteins: four body nearest-neighbor propensities of amino acid residues. *J. Comput. Biol.*, **3**, 213–221.

Topham,C.M. *et al.* (1997) Prediction of the stability of protein mutants based on structural environment-dependent amino acid substitution and propensity tables. *Protein Eng.*, **10**, 7–21.

Vaisman,I.I. *et al.* (1998) Compositional preferences in quadruplets of nearest neighbor residues in protein structures: statistical geometry analysis. In *Proceedings of the IEEE Symposia on Intelligence and Systems*, pp. 163–168.

Wang,Y. *et al.* (1996) Position-dependent protein mutant profile based on mean force field calculation. *Protein Eng*, **9**, 479–484.

Wang,L. *et al.* (1998) Can one predict protein stability? An attempt to do so for residue 133 of T4 lysozyme using a combination of free energy derivatives, PROFEC, and free energy perturbation methods. *Proteins*, **32**, 438–458.

Zhou,H. and Zhou,Y. (2002) Distance-scaled, finite ideal-gas reference state improves structure-derived potentials of mean force for structure selection and stability prediction. *Protein Sci.*, **11**, 2714–2726.