Improving Multi-Relief for detecting specificity residues from multiple sequence alignments

Elena Marchiori

Radboud University Nijmegen, The Netherlands
elenam@cs.ru.nl

Abstract. A challenging problem in bioinformatics is the detection of residues that account for protein function specificity, not only in order to gain deeper insight in the nature of functional specificity but also to guide protein engineering experiments aimed at switching the specificity of an enzyme, regulator or transporter. The majority of the state-of-the art algorithms for this task use multiple sequence alignments (MSA's) to identify residue positions conserved within- and divergent between- protein subfamilies. In this study, we focus on a recent method based on this approach called multi-RELIEF. We analyze and modify the two core parts of the method in order to improve its predictive performance. A parametric generalization of the popular RELIEF machine learning algorithm for weighting residues is introduced and incorporated in multi-RELIEF. The ensemble criterion of multi-RELIEF for merging the weights of multiple runs is simplified. Finally, the method used by multi-RELIEF for exploiting tertiary structure information is modified by incorporating prior information describing the confidence of the original scores assigned to residues. Extensive computational experiments on six real-life datasets show improvement of both robustness and detection capability of the new multi-RELIEF over the original method.

1 Introduction

Many homologous protein families have a common biological function but different specificity towards substrates, ligands, effectors, proteins and other interacting molecules. All these interactions require a certain specificity. Identifying crucial residues for this specificity is important for understanding the nature of functional specificity, for planning experiments on functional analysis or protein redesign, and for guiding point mutations aimed at switching the specificity of an enzyme, regulator or transporter.

In order to detect specificity residues, advanced computational techniques are used, because of a great variety of functional specificities observed in nature and the vast amount of protein sequence data.

Many algorithms have been proposed in recent years for this task (e.g., [1,2,4,5,6,8,10,13,22,23]). Most of them employ information-entropy related scoring functions [18] to rank residue positions according to the association with the subfamilies (see for instance the overview contained in [21]). Many methods require either a predefined subdivision of the MSA into classes, while unsupervised methods induce also a grouping during execution. The SDPpred method [10] uses mutual information to identify residue positions in which amino acid distributions correlate with the sub-family grouping [14]. In [9] the authors extended this approach to the problem of predicting protein functional sites. The Two-entropies analysis algorithm (TEA) [23] creates a 2-dimensional plot of residue conservation in terms of Shannon entropy at both superfamily and subfamily level. Recently, another method based on correlated mutation analysis [12] has been introduced to determine networks of functionally related residues in enzymes that upon mutation influence enzyme specificity and/or activity. The TreeDet approach [4] contains three algorithms for detecting so-called tree-determinant residues from an un-partitioned MSA. The Sequence Harmony (*SH*) method [5,16] scores compositional overlap between two user-specified groups. In a recent work [3], state-of-the-art methods for specificity residue detection, including *MR*, have been experimentally compared. An ensemble approach that combines predictions of the three best performing methods was used to identify new potential specificity determining sites.

In this paper we focus on a recent method for this task is multi-RELIEF (*MR*) [22], which identifies specificity residues from a given MSA and predefined multiple classes using 'local' conservation properties. *MR* employs an ensemble approach based on a machine learning algorithm for feature weighting, called RELIEF [11], that it applied multiple times to pairs of classes. Weights resulting from multiple runs are then merged.

The merging criterion assigns equal (best) score to those residues yielding perfect discrimination of at least two classes, that is, having complete within-class conservation and between-class divergence. This is undesirable, because residues discriminating well many pairs of classes should be considered more relevant than those discriminating only one pair of classes. Furthermore, RELIEF assigns equal (zero) score to residues that are either fully conserved or fully divergent. However, fully conserved residues are more relevant than fully divergent ones with respect to protein functionality. We tackle these two drawbacks of multi-RELIEF by introducing a parametric generalization RELIEF, that incorporates multiplicative factors in the formula used by RELIEF to weight residues. These factors are used to bias the weight of each residue, depending on its conservation composition in the protein sub-families considered when computing its weight. Furthermore, we replace the criterion used for merging weights with the simpler one that ensembles weights by means of their average. We call the resulting method *new-MR*.

Next, we refine the criterion used in *MR*, here called *3D*, for boosting the scores assigned to residues using information on tertiary structure, when available. Specifically, we incorporate prior information, in the form of a scaling factor associated to the original scores, describing the confidence assigned by the user to these scores. Then the new score is computed as the mean of the scaled original score and the average of the original scores of the 3D neighbors. We call the resulting criterion *new-3D*.

To assess the effectiveness of the resulting method thoroughly, six experimentally determined benchmark sets are considered, taken from five widely studied protein families: G protein-coupled receptors (GPCRs), the LacI family of bacterial transcription factor, the Ras-superfamily of small GTP-ases, the MIP-family of integral membrane transporters and the Smad family of transcription factors. We compare experimentally *MR*, and its modifications obtained by using either *new-MR* or *new-3D*, or both. Using ROC curves we show that *new-MR* identifies specificity residues as good as or better than *MR*. Moreover, when using *new-3D* overall robustness and improved predictive performance is achieved.

The rest of the paper is organized as follows. Section 2 describes the multi-RELIEF approach. In Section 3 we introduce the new methods. Section 4 contains the experimental analysis. Finally, in Section 5 we briefly summarize the results and point to future work.

2 Setting the stage: the Multi-RELIEF approach

Multi-RELIEF uses as core procedure RELIEF [11], a successful two-class feature weighting algorithm, and an ensemble approach for handling multiple classes, based on random sub-sampling pairs of classes. Random sampling of pairs of classes is mainly employed for efficiency reasons, while random sub-sampling of sequences is applied for handling unbalanced classes as well as for gaining efficiency.

Multi-RELIEF [22] is illustrated below in pseudo-code, where nr_positions denotes the total number of positions in the considered MSA. The algorithm works as follows. Multiple runs (*nr_iter*) of RELIEF are performed. At each run i, first two classes are randomly selected. Next, *nr_sample* sequences from each class are randomly selected. Finally, RELIEF is applied to the resulting two classes, yielding an output vector W_i.

```
Multi-RELIEF
input:  X1,..,Xm (m classes of an MSA), nr_iter, nr_sample
output: multi_W (weights assigned to residues)
for i=1: nr_iter
    select randomly two classes Xj, Xk
    X = select randomly nr_sample sequences from Xj and from Xk
    W_i = apply RELIEF to X
end;
for s=1: nr_positions
multi_W(s) = (see formula below);
end;
return multi_W
```

When the multiple runs are completed, the weight $multi\_W(s)$ of a residue s is computed using the formula below, where $N^+ = |\{W_i(s) > 0 \;\; \forall \;\; i\}|$ and $N^- = |\{W_i(s) < 0 \;\; \forall \;\; i\}|$.

$$multi\_W(s) = \begin{cases} \frac{1}{N^+}\sum_i \{W_i(s) > 0 \ \ \forall \ \ i\} & \text{for} \ \ N^+ > 0 \\ \frac{1}{N^-}\sum_i \{W_i(s) < 0 \ \ \forall \ \ i\} & \text{for} \ \ N^+ = 0 \ \wedge \ N^- > 0 \\ 0 & \text{for} \ \ N^+ = 0 \ \wedge \ N^- = 0 \end{cases}$$

When computing multi_W(s), all runs yielding zero weight for s are discarded. If this results in an empty set, then s is assigned weight equal to zero. Otherwise, $multi\_W(s)$ is set to the average over only the positive weights assigned to s. If there are only zero or negative weights, then $multi\_W(s)$ is the average of the negative weights assigned to s.

The core part of multi-RELIEF is RELIEF, considered one of the most successful multivariate feature weighting algorithms [7], due to its simplicity and effectiveness [11]. Recent bioinformatics applications employing (modifications of) RELIEF include, for instance, genome-wide genetic analysis [15] and gene selection [24]. RELIEF constructs a vector of weights, one for each position, by means of an iterative procedure, illustrated in pseudo-code below for two classes having the same number of sequences.

```
RELIEF
input:  X (samples of aligned proteins from two classes)
output: W (weights assigned to residues)
W = zero vector of size nr_positions
for each seq in X
     W = W + (seq - miss(seq)) - (seq - hit(seq))
end;
return W
```

The weights vector W is initialized to zero. At each iteration, one sequence *seq* is selected. The weights vector is updated by adding the 'difference' between *seq* and its nearest neighbor computed across sequences of the other class, called by $miss(seq)$, and subtracting the difference between *seq* and its nearest neighbor computed across sequences of the same class, called $hit(seq)$. This procedure is iterated over all sequences of the dataset X. The difference between two sequences $seq1 - seq2$ is a vector representing matches (`0`) and mismatches (`1`) between residues. For instance, $\texttt{ALM} - \texttt{VLM} = \texttt{100}$.

2.1 Exploiting structural information: 3D

Multi-RELIEF can optionally include tertiary structure information, if available. It does this by employing the following heuristic based on the assumption that a specificity residue does not evolve in isolation, but within a functional cluster in the protein structure [22]. This means that a residue would be more likely to be a specificity residue if its neighboring residues are also specific.

We use 3D neighbors, that is, residues that share surface with a given residue as calculated by the web server at http://ligin.weizmann.ac.il/cma/ [19].

Alternatively, the tools available at http://www.infobiotics.org/ could be used. For instance, the `PSP` server could be used to either calculate, if the structure is known, or predict when the structure is unknown, many topological and geometrical 3D features. Moreover, the `ProCKSI` server could be used to calculate multiple versions of contact maps, and compare contact maps of the structures of the MSA sequences.

The weight of a residue is adjusted by adding the average weight of its 3D neighbors. In this way the score of a residue is boosted if its neighbors have a high average score.

3 Improving Multi-RELIEF

In multi-RELIEF, $multi\_W(s)$ assigns a high score to position s discriminating at least two classes. In particular, a maximum weight is assigned if s fully discriminates two specific classes but does not differentiate (i.e. weight less than or equal to zero) any other pair of classes. Furthermore, RELIEF assigns equal (zero) score to residues that are either fully conserved or fully divergent. However, fully conserved residues are more relevant than fully divergent ones with respect to protein functionality. Example of these two cases is shown in Table 1, where residues b and c have equal maximum score, and a and d have both score equal to zero. We propose the following approach for overcoming these two drawbacks of multi-RELIEF.

3.1 new-multi-RELIEF

We introduce a parametric formula for computing weights assigned to position s. Let $cl(seq)$ denotes the class label (protein sub-family) of seq, and $c1, c2$ the labels of the two classes. Denote the element of a sequence seq in position s by seq_s. Let

$$W_{miss}(s) = \sum_{seq \in X}(seq_s - miss(seq)_s),$$
$$W_{hit,min}(s) = \min(W_{c1}(s), W_{c2}(s)),$$
$$W_{hit,max}(s) = \max(W_{c1}(s), W_{c2}(s))$$

and

$$W_{c1}(s) = \sum_{seq \in X, cl(seq)=c1}(seq_s - hit(seq)_s,$$
$$W_{c2}(s) = \sum_{seq \in X, cl(seq)=c2}(seq_s - hit(seq)_s).$$

Then we define

$$W_{new}(s) = \alpha_0 W_{miss}(s) - \alpha_1 W_{hit,min}(s) - \alpha_2 W_{hit,max}(s),$$

with $\alpha_0, \alpha_1, \alpha_2$ in $(0, 1]$ We call $W_{hit,min}$ and $W_{hit,max}$ *within-one-class weights*, and the parameters α *factors*, which can be viewed as multiplicative factors measuring the relevance given by the user to each term of the sum. Their values can be chosen depending on the specific type of application.

One can easily check that, for $\alpha_0 = \alpha_1 = \alpha_2 = 1$ one obtains the formula used in RELIEF to compute weights.

For each residue s, we assign more relevance (that is, high α_1) to the highest within-one-class conservation, that is, to the minimum within-one-class weight of s. Moreover, we account for uncertainty stemming, for instance, from the incompleteness of the considered protein sub-families, the quality of the alignments, and the possible presence of class noise (that is, proteins assigned to incorrect sub-families). We address uncertainty by choosing α_1 smaller than 1 (in our experimental analysis, the value 0.8 is selected). The remaining two terms of $W_{new}(s)$ are considered to have half the relevance of the minimum within-one-class weight (in our experiments $\alpha_0 = \alpha_2 = 0.4$).

Using W_{new} instead of W in Multi-RELIEF and by averaging the resulting weights for computing multi_W(s), we obtain the new-RELIEF algorithm shown below in pseudo-code.

```
new-Multi-RELIEF
input:  X1,..,Xm (m classes of an MSA), nr_iter, nr_sample
output: multi_W (weights assigned to positions)
for i=1: nr_iter
    select randomly two classes
    X = select randomly nr_sample sequences from each selected class
    W_i = W_new computed using X
end;
for s=1: nr_positions
new_multi_W(s) = average of W_i(s);
end;
return new_multi_W
```

The example given in Table 1, slightly adapted from [22], shows the (normalized) weights generated by multi-RELIEF and by new-multi-RELIEF. The MSA of four sub-families, $C_1, \ldots, C_4$, is considered, where each protein sequence has six positions $a, \ldots, f$.

The two feature weighting methods induce different rankings of the residues. Indeed, multi-RELIEF ranks both b and c as most relevant positions, followed by f, then both a and d which are considered equally relevant, followed by e. Instead new-multi-RELIEF generates a different ranking, namely c, b, a, f, d, e. In particular, new-multi-RELIEF considers residue c more relevant than b and the fully conserved residue a more relevant than d.

3.2 new-3D

In the boosting procedure of multi-RELIEF $multi\_W(s)$ is adjusted by adding the average weight of its 3D neighbors. Here we use 3D neighbors to change the weight of position s by associating the scaling factor β for the original weights, which can be viewed as prior confidence of these weights. Then the new weights

	a	b	c	d	e	f
C_1	R R R R	F F F F	T T T T	I Q N A	T F V D	T F V D
C_2	R R R R	F F F F	Y Y Y Y	S F D V	T F V D	N N N N
C_3	R R R R	Y Y Y Y	D D D D	E V W G	T F V D	T F V D
C_4	R R R R	Y Y Y Y	H H H H	H P Y C	T F V D	T F V D
Multi-RELIEF weights	0	1	1	0	-1	0.5
new Multi-RELIEF weights	0	0.3	0.4	-0.8	-1.2	-0.6

Table 1. Weights computed by multi-RELIEF applied to a toy example.

are obtained by computing the mean of the original weight scaled by its prior importance, and the average of the original weights of its 3D neighbors:

$$new\_3D\_W(s) = \frac{1}{2}(\beta multi\_W(s) + \frac{\sum_{s' \in 3Dnn(s)} multi\_W(s')}{|3Dnn(s)|}),$$

where $3Dnn(s)$ are the positions of the 3D neighbors of s.

In the experimental analysis conducted in the sequel we use $\beta = 3$, that is, we assign high confidence to the original weights. (In general, higher values of β lead to similar results.)

4 Experimental Analysis

To assess the performance of the proposed modifications of the multi-RELIEF method, we considered the following six algorithms:

1. *MR*: the original multi-RELIEF algorithm from [22];
2. *new-MR*: the new multi-RELIEF algorithm;
3. *MR 3D*: multi-RELIEF with 3D weight boosting procedure;
4. *MR new-3D*: multi-RELIEF with the new 3D weight boosting procedure;
5. *new-MR 3D*: new multi-RELIEF with the 3D weight boosting procedure;

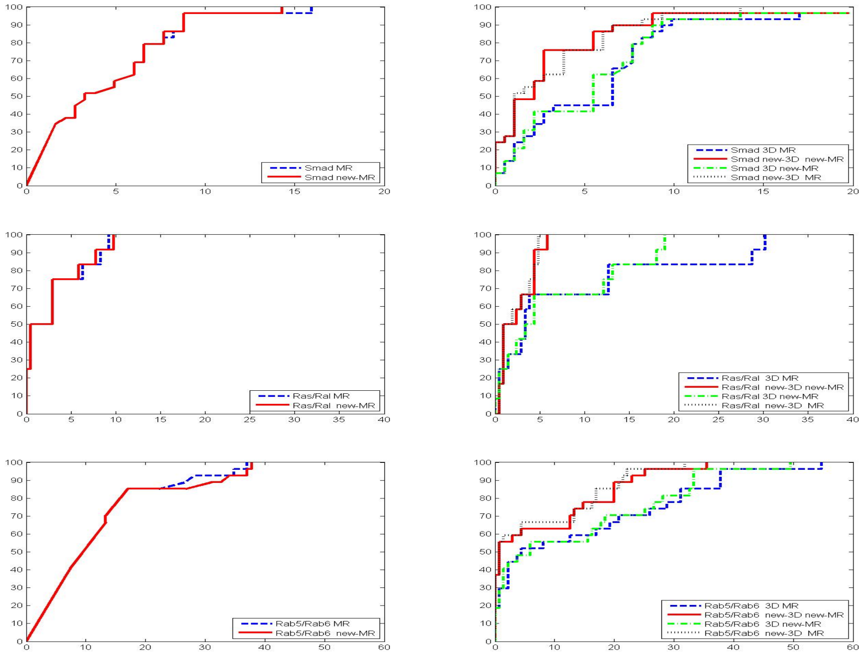

Fig. 1. ROC curves of the algorithms. From top to bottom: dataset Smad, Ras/Ral and Rab5/Rab6.

6. *new-MR new-3D*: new multi-RELIEF with the new 3D weight boosting procedure.

We conducted experiments datasets previously used in [22], containing different protein families with various associated functional specificity properties. Properties of these datasets are summarized in Table 2; they are thoroughly described in [22].

The following parameter values were used: $\alpha_1 = 0.8, \alpha_2 = \alpha_0 = 0.4, \beta = 3$, $nr\_iter = 2000$ and $nr\_samples = 5$. In general, a high value of nr_iter and a reasonably small value of $nr\_samples$ can be used. Ties were broken by sorting residue positions with equal score in increasing sequence position.

The Receiver-operator characteristic (ROC) curve is used for testing the capability of an algorithm to separate true and false positives [20,17]. Known functional specificity residues are considered true positives, the other ones true negatives. The weight values are used as threshold for generating the ROC curve. For each weight value v the set of residues with weight higher than or equal to v is considered: the true positive percentage is reported on the y-axis (sensitivity, or coverage), and the false positive percentage (1$-$specificity, or error) on the

dataset	nr of classes	avg class size	standard deviation	max class size	min class size	sequence length	formation
GPCR	77	26.8	34	189	3	214	ligand
LacI	15	3.6	2.5	12	2	339	ligand and DNA
Ras/Ral	2	44.5	24.5	69	20	218	protein
Rab5/Rab6	2	5.0	1	4	6	163	protein
MIP	2	30.0	18	48	12	430	protein
Smad	2	10.0	2	12	8	211	protein

Table 2. Properties of the datasets used for testing the algorithms.

x-axis. The ROC curve thus describes the goodness of a method in giving higher ranking to *known* functionally specific residues.

Results of experiments are given in Figures 1, 2. They can be summarized as follows:

- Significant improvement of new-*MR* is achieved on the GPCR dataset, which contains many classes, and on the MIP one. On the other datasets results of new-*MR* and *MR* do not differ significantly (see left column of Figures 1, 2).
- When also tertiary structure information is used, results improve on all datasets except LacI (see right column of Figures 1, 2). Specifically, on Smad, Ras/Ral and Rab5/Rab6 new-3D new-*MR* outperforms significantly all other *MR* variants employing 3D information. On the other three datasets performances of the *MR* variants do not significantly differ from each other, except for GPCR, where both 3D new-*MR*and new-3D new-*MR* ROC curves significantly dominate the other ones. On the LacI dataset, new-3D new-*MR* show slightly worse performance than 3D *MR*. However, classes in this dataset are rather small, with one class containing 12 elements and all other classes having very few elements (≤ 4). We can take into account this characteristic of the dataset and set the parameter values in such a way that importance of the maximum within-one-class conservation is strengthen (smaller values for α_0, α_2), and the confidence of the original weights is decreased (smaller value for β). For instance, improvement is achieved by choosing $\alpha_1 = 0.8, \alpha_0 = \alpha_2 = 0.2, \beta = 0.5$ (see Figure 2, bottom plots).
- *new-3D new-MR* shows best overall results across all datasets. Performance of *3D* new-*MR* is also very good on the GPCR, MIP and LaCI datasets (see Figure 1), but is significantly worse than the one of the other algorithms on the Smad, Ras/Ral and Rab5/Rab6 datasets (see Figure 2).

In general, the results substantiate the effectiveness of the proposed new method for detecting of specificity residues from multiple sequence alignment. Indeed, new-3D new-*MR* achieves best overall performance, in particular significantly improving robustness and performance of the state-of-the-art method multi-RELIEF when tertiary structure information is used.

5 Conclusion

We proposed a new algorithm for detecting specificity residues from multiple sequence alignments, which is obtained by modifying the core parts of the recent state-of-the-art method for this task multi-RELIEF. Results of extensive experiments showed improved overall performance and robustness of the new method, especially when tertiary structure information is used. In future work, we want to extend the proposed method to predict functional sites.

Acknowledgements

Thanks to Leonid A. Mirny and Mikhail S. Gelfand for making their LacI dataset available, and to Anton Feenstra, Walter Pirovano and Kai Ye for the other datasets and for the useful past discussions on the subject of this paper. This work was partially supported by the NWO project 639.023.604.

References

1. P.J. Bickel, K.J. Kechris, P.C. Spector, G.J. Wedemayer, and A.N. Glazer. Finding important sites in protein sequences. *Proc. Natl Acad. Sci. USA*, 99:14764–71, 2002.
2. A. Carro, M. Tress, D. de Juan, F. Pazos, P. Lopez-Romero, A. Del Sol, A. Valencia, and A.M. Rojas. Treedet: a web server to explore sequence space. *Nucleic Acids Res.*, 35(Web Server Issue):99, 2006.
3. S. Chakrabarti and A.R. Panchenko. Ensemble approach to predict specificity determinants: benchmarking and validation. *BMC Bioinformatics*, 10:207, 2009.
4. A. Del Sol Mesa, F. Pazos, and A. Valencia. Automatic methods for predicting functionally important residues. *J Mol Biol*, 326(4):1289–1302, 2003.
5. K.A. Feenstra, W. Pirovano, K. Krab, and J. Heringa. Sequence harmony: detecting functional specificity from alignments. *Nucleic Acids Res.*, 35(web server issue):W495–W498, 2007.
6. X. Gu. A simple statistical method for estimating type-ii (cluster-specific) functional divergence of protein sequence. *Mol Biol Evol.*, 23:1937–45, 2006.
7. I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.
8. S.S. Hannenhalli and R.B. Russell. Analysis and prediction of functional sub-types from protein sequence alignments. *J Mol Biol*, 303(1):61–76, 2000.
9. O.V. Kalinina, M.S. Gelfand, and R.B. Russell. Combining specificity determining and conserved residues improves functional site prediction. *BMC Bioinformatics*, 2009.
10. O.V. Kalinina, P.S. Novichkov, A.A. Mironov, M.S. Gelfand, and A.B. Rakhmaninova. SDPpred: a tool for prediction of amino acid residues that determine differences in functional specificity of homologous proteins. *Nucleic Acids Res*, 32(Web Server issue):W424–8, 2004.
11. I. Kononenko. Estimating attributes: Analysis and extensions of relief. In Springer, editor, *European Conference on Machine Learning*, volume LNCS 784, pages 171–182, 1994.

12. R.K. Kuipers, H.-J.J. Joosten, E. Verwiel, S. Paans, J. Akerboom, J. van der Oost, N.G. Leferink, W.J. van Berkel, G. Vriend, and P.J. Schaap. Correlated mutation analyses on super-family alignments reveal functionally important residues. *Proteins*, 76(3):608–616, 2009.
13. I. Mihalek, I. Res, and O. Lichtarge. A family of evolution-entropy hybrid methods for ranking protein residues by importance. *J Mol Biol*, 336(5):1265–1282, 2004.
14. L.A. Mirny and M.S. Gelfand. Using orthologous and paralogous proteins to identify specificity-determining residues in bacterial transcription factors. *J Mol Biol*, 321(1):7–20, 2002.
15. J.H. Moore and B.C. White. Tuning relieff for genome-wide genetic analysis. In *Evolutionary Computation,Machine Learning and Data Mining in Bioinformatics (EvoBIO 2007)*, volume 4447 of *LNCS*, pages 166–175. Springer, 2007.
16. W. Pirovano, K.A. Feenstra, and J. Heringa. Sequence comparison by sequence harmony identifies subtype specific functional sites. *Nucleic Acids Res.*, 34:6540–48, 2006.
17. F. Provost and R. Kohavi. Guest editors' introduction: On applied research in machine learning. *Machine Learning*, 30:127–132, 1998.
18. P.S. Shenkin, B. Erman, and L.D. Mastrandrea. Information-theoretical entropy as a measure of sequence variability. *Proteins*, 11(4):297–313, 1991.
19. V. Sobolev, A. Sorokine, J. Prilusky, E.E. Abola, and M. Edelman. Automated analysis of interatomic contacts in proteins. *Bioinformatics*, 15:327–332, 1999.
20. J.A. Swets. Measuring the accuracy of diagnostic systems. *Science*, 240:1285–1293, 1988.
21. J.C. Whisstock and A.M. Lesk. Prediction of protein function from protein sequence and structure. *Quart Rev Biophys*, 36(3):307–340, 2003.
22. K. Ye, K.A. Feenstra, J. Heringa, A.P. IJzerman, and E. Marchiori. Multi-relief: a method to recognize specificity determining residues from multiple sequence alignments using a machine-learning approach for feature weighting. *Bioinformatics*, 24(1):18–25, 2008.
23. K. Ye, E.W. Lameijer, M.W. Beukers, and A.P. IJzerman. A two-entropies analysis to identify functional positions in the transmembrane region of class a g protein-coupled receptors. *Proteins*, 63:1018–30, 2006.
24. Y. Zhang, C. Ding, and T. Li. Gene selection algorithm by combining relieff and mrmr. *BMC Genomics*, 9(Suppl 2), 2008.

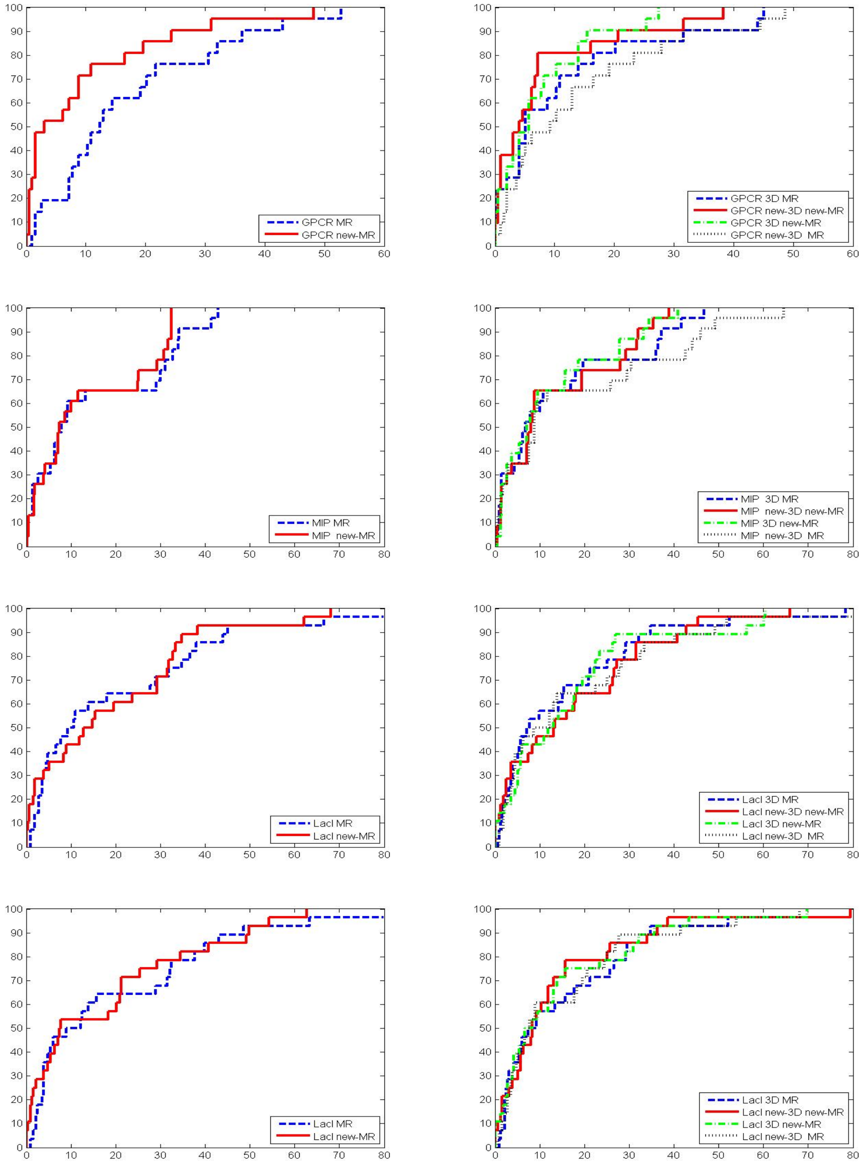

Fig. 2. ROC curves of the algorithms. From top to bottom: dataset GPCR, MIP, LacI, and LacI with $\alpha_1 = 0.8$, $\alpha_0 = \alpha_2 = 0.2$, $\beta = 0.5$.