Protein Binding Sites Prediction Using Ensembles

Georgina Mirceva and Andrea Kulakov

Ss. Cyril and Methodius University in Skopje,
Faculty of Computer Science and Engineering, Skopje, Macedonia
{georgina.mirceva, andrea.kulakov}@finki.ukim.mk

Abstract. Protein molecules play essential roles in the living organisms. The knowledge about their functions is very important in order to design new drugs that could be used to control various processes in the organisms. The determination of the protein functions could be performed by detecting the binding sites where interactions between proteins occur. In this paper we focus on predicting the protein binding sites. First, several characteristics of the amino acid residues are extracted. Then, prediction methods are induced. In this research paper we consider several classification methods for inducing models. In order to enhance the predictions, we use ensembles, which combine several classification models. The results show that using ensembles, the prediction power is increased.

Keywords: Protein function, protein interaction, protein binding site, BIND database, ensembles.

1 Introduction

Protein molecules contain one or several protein chains that are constructed by amino acid residues, which fold in particular conformation in 3D space. Further, the amino acids residues are constructed by several atoms. The knowledge about the functions of the protein molecules is very important because this knowledge could be used for designing new drugs in order to control various processes in the living organisms. This importance triggers many research groups to investigate various methods for protein annotation. There are experimental methods for discovering the protein functions. However, these methods are very expensive and time-consuming. On the other side, with the high-throughput technologies numerous protein structures are determined every day, thus many protein molecules with determined structures are not annotated yet. Therefore, there is an obvious need for fast computational methods for protein function prediction.

In the current literature, there are various methods for protein annotation. Different methods consider different information regarding the interacting protein structures. First group of methods [1] examines the sequence and/or structure homology of the protein molecules. Second group of methods [2] identifies the conserved parts of the protein sequences and/or structures, and determines the protein functions based on the

features of the conserved regions. There is a third group of methods [3] that annotate protein structures based on the motifs (signatures) found in their sequences. As a fourth group of methods we identify the methods that annotate the protein structures based on the characteristics of the protein binding sites. The protein binding sites are the regions of the protein structures where interactions with other protein structures occur. In [4], the authors provide a wide survey of available tools and web servers for protein binding sites prediction. As a fifth group of methods [5] we identify the methods that do not consider the protein sequences and structures, but consider information regarding the interacting pairs of protein structures presented by the protein-protein interaction networks. In this research we focus on the fourth group of methods, and we aim to provide models that successfully detect the protein binding sites. Afterward, the predictions about the binding sites obtained by our models could be used for protein annotation.

There are numerous methods for protein binding sites prediction. In order to identify the amino acid residues that are part of a given binding site, different characteristics of the residues could be considered. The Accessible Surface Area (ASA) [6], depth index (DPX) [7], protrusion index (CX) [8] and hydrophobicity [9] are the amino acid residues' characteristics that are used the most. None of the characteristics does not provide sufficient information in order to make perfect prediction of the residues that take part of binding sites. Therefore, several characteristics are considered in the induction of the prediction models. In this research we consider these four features, i.e. ASA, DPX, CX and hydrophobicity.

After extraction of the characteristics of the amino acid residues, then prediction models are induced. In the literature various classification methods are used for this purpose. In this research paper we consider several classifiers for inducing models for protein binding sites prediction.

In order to increase the prediction power of the models, ensembles could be used. An ensemble model combines several models. There are several techniques for inducing ensemble models, including bagging and boosting. In the induction of the models, the samples are randomly chosen with replacement. At the beginning each sample has equal probability to be chosen. In bagging, in the induction of the later models, the probability that a given sample would be randomly chosen in the training dataset for the given model remains uniform. On the other side, in boosting we increase the probability for choosing the samples that are misclassified by the previous model. In this way, the samples that are harder for learning are more frequently presented to the classifiers. Namely, this way we force the learning of the samples that are located in regions in the *N*-dimensional space where more misclassification errors occur. In this research we induce both bagging and boosting ensemble models using various classification methods.

The rest of this research paper is organized in this way. In section 2, first we present how the amino acid residues' characteristics are extracted. Then, we explain the model induction using bagging and boosting techniques. Section 3 provides results of the evaluation of the prediction models. In section 4 we conclude the paper and identify directions for additional improvements.

2 Protein Binding Sites Prediction Models

In this section we present our approach for predicting the protein binding sites. We classify each amino acid residue in one of the two classes (part of binding site or not). First, we extract several characteristics for each amino acid residue of the inspected protein chain. Then, the prediction models are induced using various classification methods. In this paper we induce single models, and also we induce ensembles using bagging and boosting techniques.

2.1 Extraction of the Amino Acid Residues Characteristics

The most widely used characteristics of the amino acid residues are Accessible Surface Area (ASA) [6], depth index (DPX) [7], protrusion index (CX) [8] and hydrophobicity [9]. Since none of the characteristics does not provide sufficient information whether a given residue is a part of a binding site or not, therefore several characteristics are used in the model induction. In this research we consider the four features mentioned below. Since an amino acid residue contains several atoms, thus first we calculate the ASA, DPX and CX for each atom. Then, the corresponding characteristic for the amino acids residue is calculated using some aggregation of the values of the characteristic obtained for each atom.

The Accessible Surface Area (ASA) [6] is usually expressed in $Å^2$ and is calculated as a surface area of the atom that could be reached by a rolling sphere. In this research we use a rolling sphere with radius of 1.4 Å, which is the most common value. The rolling sphere is rolled around the protein structure by making small slices. Small arcs are formed as intersections of the rolling sphere and the slices. The value of the ASA in the *i*-th slice is calculated as in [6]

$$ASA_i = \frac{R}{\sqrt{R^2 - Z_i^2}} * (\Delta Z / 2 + \Delta' Z) * L_i \quad (1)$$

$$\Delta' Z = \min(\Delta Z, R - Z_i), \quad (2)$$

where R is the radius of the inspected atom, Z_i is the distance between the centre of the rolling sphere and the *i*-th section, L_i is the length of the corresponding slice and ΔZ denotes the distance between adjacent slices. In this way we calculate the ASA in each slice. Then, the ASA of the inspected atom is calculated as sum of the ASA values obtained in all slices. Since an amino acid residue contains several atoms, therefore we sum the ASA values of all atoms that constitute the residue.

In the protein structure, numerous atoms are hidden in the interior of the protein, and they could not be reached by the rolling sphere. As a consequence of this, they could not be a part of a binding site. Thus, we filter only the amino acid residues that are located at the protein surface. In this filtering we consider that a given amino acid residue is located at the protein surface if the ratio of its ASA and the total surface area of the residue is not lower than 5%, as suggested in [10].

The depth index (DPX) [7] of an atom denotes the distance from the centre of the atom to the nearest atom (including the inspected atom) that could be reached by the rolling sphere. The depth index of the atoms that could be reached by the rolling sphere is zero, and greater than zero for the remaining atoms. In this way the depth index gives evidence how far a given atom is from the protein surface. After extraction of DPXs of all atoms, the DPX of an amino acid residue is calculated as an average of the DPXs of its atoms.

The protrusion index (CX) is calculated as in [8]. First, we calculate the number of non-hydrogen atoms that are located in the neighboring around the examined atom. In this research we inspect the atoms that are within a sphere with radius of 10 Å, as suggested in [8]. Then, we calculate the volume occupied by the protein structure by multiplying this number of non-hydrogen atoms and the average volume of an atom (20.1 $Å^3$ [8]). Finally, the protrusion index CX is calculated as a ratio of the remaining volume and the occupied volume, where the remaining volume is a difference between the total volume of the inspected sphere and the volume of the inspected sphere that is occupied by the protein structure. In this way, CX gives evidence about the density around given atom. The atoms located in regions with higher density have lower CX, while the atoms located in regions with lower density have higher CX. The CX of an amino acid residue is calculated as a mean of the CXs of the atoms that constitute the inspected residue.

The hydrophobicity characteristic [9] is the last characteristic that we use in this research. Hydrophobicity indicates the hydrophobic properties of the amino acid and is related with the hydrophobic effect. Namely, hydrophobic amino acids are typically found deeply in the protein interior, and hydrophilic amino acids are usually located towards the surface of the protein molecule. There are several scales for expressing the hydrophobic properties of the amino acids. We use the hydrophobicity scale proposed by Kyte and Doolittle [9].

2.2 Induction of Ensemble Models for Protein Binding Sites Prediction

After extraction of the characteristics of the amino acids residues, next we induce prediction models for identifying the amino acid residues that are part of binding sites. We use the following classification methods for building prediction models: C4.5 Tree [11], Alternating Decision Tree (ADTree) [12], Naïve Bayes [13], Naïve Bayes Tree [14] and Bayesian Network [15].

In order to increase the prediction power of the models, we also induce ensembles that combine several models. In this way we aim to induce ensemble models that are capable to overcome the problems of the individual models. We consider two techniques for building ensembles, i.e. bagging [16] and boosting [17]. Both techniques use randomization to choose samples that would be used for building models. Let we have a training dataset D with $|D|$ samples, and let we want to induce m models using some classification method. We generate m training datasets D_i, i=1,…m, with size $|D_i| \leq |D|$ by sampling the dataset with replacement and following the same distribution of the class attribute as in the entire training dataset. Since we resample the samples with replacement, some samples could be considered several times in same training

dataset. Using the training datasets D_i, i=1, 2,…m, we induce m separate models. During testing, the test samples are presented to each of the m models, and the final decision regarding the class attribute is made using voting. In this research the models have equal weights in the voting. In bagging, the samples have equal probabilities to be randomly chosen during the entire process. On the other side, using boosting the samples that are harder for learning have higher probability to be randomly chosen. At the beginning the samples in the training dataset have equal probabilities to be chosen. We randomly chose samples and form the training dataset D_1. Then, the model M_1 is induced using the dataset D_1, and the samples from D_1 are presented to the model M_1 for prediction. For each sample from D_1 that is misclassified by the model M_1 the probability for choosing is increased. On the other side, for each sample in D_1 that is correctly predicted by M_1 the probability for choosing is decreased. Using the new probabilities, the dataset D_2 is generated. Then, the models M_2 is induced, and based on the predictions about the samples in D_2, the new probabilities of the samples are calculated. This procedure is repeated until m models are induced, or while the weight threshold reaches a predefined threshold. In this way, if a given sample is misclassified by the previous models, it has higher chance to be chosen in the training of the next models. In this research paper, we use the implementations of the classification methods and the methods for inducing ensembles provided in the Weka software [18]. For the boosting we use Adaptive Boosting (AdaBoost) method [17]. For each method we use the default settings if it is not otherwise specified.

3 Experimental Results

For evaluation of the prediction models, we use the knowledge stored in the Biomolecular Interaction Network Database (BIND) [19], which contains information about the protein binding sites. The knowledge stored in this database is acquired in experimental manner, and therefore we consider this knowledge as a standard of truth. From the BIND database we filter the protein chains that do not have more than twenty percents similarity in their sequences. For this filtering we use the criterion given in [20]. Then, using the same selection criterion [20] we generate the test set by selecting the protein chains with less than ten percents similarity in their sequences. All protein chains that belong to the first dataset, and do not belong to the second dataset are considered in the training dataset.

Next, we filter the surface amino acid residues. After filtering the surface residues, we obtain a training dataset with 115579 samples, from which only 15696 are part of binding sites. The obtained test dataset contains 625939 amino acid residues from which majority belong to the non-binding sites' class. In order to avoid inducing models that are biased towards the dominant class (the non-binding sites' class), we balance the training dataset until uniform distribution is obtained. However, we do not perform balancing on the test dataset, so in the evaluation of the prediction models we have to use some evaluation measure that is appropriate for unbalanced datasets. After balancing the training dataset, next we normalize the amino acid residues' characteristics thus obtaining values in the interval [0;1].

There are various measures that could be used for evaluation of models for predicting a discrete class attribute. However, the classification accuracy is not appropriate measure for evaluation of a model using unbalanced test dataset. Since the test set in our case is not balanced, in this research paper we use the Area Under the ROC Curve (AUC-ROC) measure to estimate the prediction power of the models. AUC-ROC is calculated as TPR*TNR+TPR*(1-TNR)/2+TNR*(1-TPR)/2= (TPR+TNR)/2, where TPR and TNR denotes the true positive and true negative rates respectively. The true positive rate is calculated as TP/(TP+FN), while the true negative rate is determined by TN/(TN+FP), where TP and TN correspond to the number of true positives and true negatives, while FP and FN denote the number of false positives and false negatives correspondingly. In this way, the AUC-ROC measure achieves values in the interval [0,1]. Value 1 means perfect prediction of the class attribute, while value 0 denotes that the model makes inverse predictions.

First, we evaluate the prediction power of the models obtained using different classification methods. In this analysis we induced single models. The results for AUC-ROC obtained from this analysis are provided in Table 1. The results show that the C4.5 tree obtains highest AUC-ROC, while the Alternating Decision Tree based model has lowest prediction power.

Table 1. The AUC-ROC obtained by the single models using various classification methods.

Classification method	AUC-ROC
C4.5 tree	0,5866
Alternating Decision Tree	0,5455
Naïve Bayes	0,5668
Naïve Bayes Tree	0,5857
Bayesian Network	0,5762

Next, we induce ensemble models using bagging. In this analysis we examine the number of iterations m, which corresponds to the number of models used in the ensembles. Also, we induce several models using datasets D_i, i=1,2,…m, with different sizes $|D|*k/100$, k=5, 10, 20, 50 and 100, where $|D|$ denotes the number of samples in the entire training dataset. The results of this analysis are presented in Table 2. Similarly, we induce ensemble models using boosting. We use different values for the maximal allowed number of iterations (m=10, 20 and 50). The results of the ensemble models using boosting are provided in Table 3.

The results given in Table 2 and Table 3 show that the ensemble model using C4.5 tree obtained by bagging using m=20 and k=50 has highest AUC-ROC among the models that are based on the C4.5 tree classifier. Using bagging, the prediction power of the C4.5 tree based models is increased from 0,5866 to 0,5878. In this analysis the boosting models do not outperformed the single C4.5 tree model. Regarding Alternating Decision Tree (ADTree), using single model we obtained AUC-ROC of 0,5455. Using ensembles we increased AUC-ROC up to 0,5804 with bagging and up to 0,5839 with boosting. Similarly, using Naïve Bayes, the AUC-ROC is increased from 0,5668 up to 0,5740. Also, with the other classification methods using ensembles we

improved the prediction power of the models. Generally, boosting showed as better technique for building ensembles, except for C4.5 tree classifier. Regarding boosted C4.5, it is interesting to mention that for m=10 (using 10 models) higher AUC-ROC is obtained than for m=20 and 50. With boosting, in the later iterations the misclassified samples are more frequently chosen for training, so therefore the models become over-fitted regarding these samples.

Regarding the values of k, from Table 2 we can see that more accurate models are induced when the models are induced using datasets with smaller size than the entire training set (for k<100). In this way using lower k, the models are not over-fitted. However, using too small sets (for k=5), the prediction power is lower. In order to get better picture of the influence of the parameter k, on Figure 1 and Figure 2 we present the results from Table 2 graphically. On the x-axis the examined values of k are given. From these figures we can see that generally the optimal values for k are 10, 20 and 50. From the figures it is also evident that using k=100 (using datasets D_i with same size as the entire training dataset), the prediction is getting worse when C4.5 and ADTree are used since the models are over-fitted. However, this is not a case using the Bayesian methods (Naïve Bayes, Naïve Bayes Tree and Bayesian Network).

Table 2. The AUC-ROC obtained by the ensemble models using bagging technique and various classification methods.

m	k	C4.5 Tree	Alternating Decision Tree	Naïve Bayes	Naïve Bayes Tree	Bayesian Network
10	5	0,5860	0,5781	0,5656	0,5714	0,5717
10	10	0,5855	0,5804	0,5671	0,5717	0,5690
10	20	0,5865	0,5703	0,5675	0,5726	0,5693
10	50	0,5842	0,5533	0,5668	0,5718	0,5718
10	100	0,5708	0,5455	0,5665	0,5777	0,5761
20	5	0,5865	0,5700	0,5667	0,5730	0,5725
20	10	0,5873	0,5691	0,5673	0,5714	0,5702
20	20	0,5876	0,5748	0,5670	0,5730	0,5693
20	50	**0,5878**	0,5613	0,5668	0,5708	0,5708
20	100	0,5731	0,5455	0,5668	0,5781	0,5750

Table 3. The AUC-ROC obtained by the ensemble models using boosting technique and various classification methods.

m	C4.5 Tree	Alternating Decision Tree	Naïve Bayes	Naïve Bayes Tree	Bayesian Network
10	0,5825	0,5837	**0,5740**	**0,5868**	**0,5859**
20	0,5722	**0,5839**	**0,5740**	**0,5868**	**0,5859**
50	0,5722	**0,5839**	**0,5740**	**0,5868**	**0,5859**

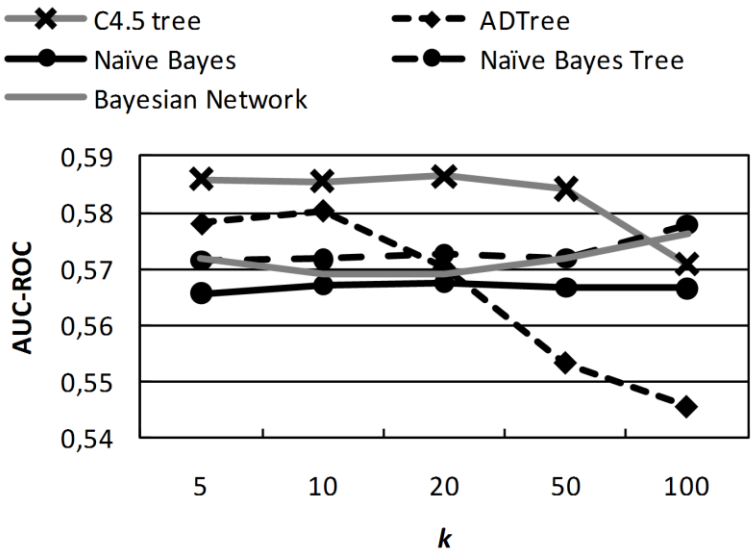

Fig. 1. The AUC-ROC obtained by the ensemble models for *m*=10 using bagging technique and various classification methods. On the x-axis the inspected values of *k* are given.

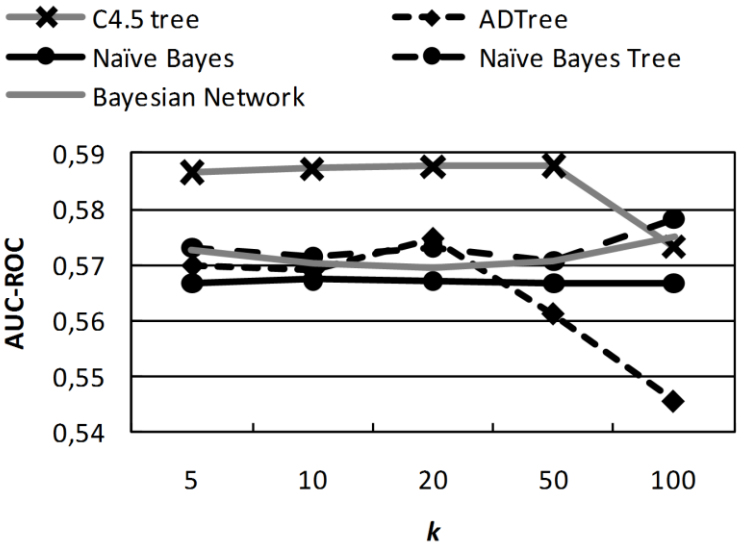

Fig. 2. The AUC-ROC obtained by the ensemble models for *m*=20 using bagging technique and various classification methods. On the x-axis the inspected values of *k* are given.

4 Conclusion and Future Work

In this paper we introduced an approach for inducing models for predicting the protein binding sites. The predictions of our models could be used to determine the functions of the protein structures. First, we extracted several characteristics of the amino acid residues. Then, we induced models using various classification methods. In order to enhance the prediction power of the models, we induced ensemble models that combine several single models. For this purpose we used the bagging and boosting techniques for building ensembles.

The results showed that we improved the predictions using ensembles of models. Generally boosting showed as better technique for building ensembles (except for C4.5 tree) since it forces learning of the samples that are harder for learning. We examined the influence of the parameter *k* that defines the size of the training datasets that are used for building the individual models in bagging. The results showed that for the C4.5 tree and the ADTree, *k* should not be too high. Regarding the values of the parameter *m*, which denotes the number of induced models that are combined in the ensemble, we can conclude that generally as *m* increases also the prediction power increases. However, in the same time also the training and testing times linearly increase.

Further, we plan to induce ensemble models for protein binding sites prediction using other classification methods. Also, we will explore whether the set of four amino acid residues' characteristics that is used in this research is the most relevant one, or maybe other set of characteristics is more suitable for protein binding sites prediction.

Acknowledgments. This work was partially financed by the Faculty of Computer Science and Engineering at the "Ss. Cyril and Methodius University in Skopje", Skopje, R. Macedonia.

References

1. Todd, A.E., Orengo, C.A., Thornton, J.M.: Evolution of function in protein superfamilies, from a structural perspective. J. Mol. Biol. 307, 4, 1113–1143 (2001)
2. Panchenko, A.R., Kondrashov, F., Bryant, S.: Prediction of functional sites by analysis of sequence and structure conservation. Protein Science 13, 4, 884–892 (2004)
3. Sigrist, C.J.A., Cerutti, L., De Castro, E., Langendijk-Genevaux, P.S., Bulliard, V., Bairoch, A., Hulo, N.: PROSITE, a protein domain database for functional characterization and annotation. Nucleic Acids Res. 38, Database issue, D161–D166 (2010)
4. Tuncbag, N., Kar, G., Keskin, O., Gursoy, A., Nussinov, R.: A survey of available tools and web servers for analysis of protein-protein interactions and interfaces. Briefings in Bioinformatics 10, 3, 217–232 (2009)
5. Kirac, M., Ozsoyoglul, G., Yang, J.: Annotating proteins by mining protein interaction networks. Bioinformatics 22, 14, e260–e270 (2006)
6. Lee, B., Richards, F.M.: The interpretation of protein structures: Estimation of static accessibility. J. Mol. Biol. 55, 3, 379–400 (1971)

7. Pintar, A., Carugo, O., Pongor, S.: DPX: for the analysis of the protein core. Bioinformatics 19, 2, 313–314 (2003)
8. Pintar, A., Carugo, O., Pongor, S.: CX, an algorithm that identifies protruding atoms in proteins. Bioinformatics 18, 7, 980–984 (2002)
9. Kyte, J., Doolittle, R.F.: A simple method for displaying the hydropathic character of a protein. J. Mol. Biol. 157, 1, 105–132 (1982)
10. Chothia, C.: The Nature of the Accessible and Buried Surfaces in Proteins. J. Mol. Biol. 105, 1, 1–12 (1976)
11. Quinlan, R.: C4.5: Programs for Machine Learning, 1st ed., Morgan Kaufmann Publishers, San Mateo, CA, USA (1993)
12. Freund, Y., Mason, L.: The alternating decision tree learning algorithm. In: Proceedings of the 7th International Conference on Machine Learning (ICML 1999), Bled, Slovenia, June 27-30, 1999, Morgan Kaufmann, San Francisco, CA, USA, pp. 124–133 (1999)
13. John, G.H., Langley, P.: Estimating Continuous Distributions in Bayesian Classifiers. In: Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence, Montreal, Quebec, Canada, August 18-20, 1995, Morgan Kaufmann, San Francisco, CA, USA, pp. 338–345 (1995)
14. Kohavi, R.: Scaling Up the Accuracy of Naive-Bayes Classifiers: A Decision-Tree Hybrid. In: Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD 1996), Portland, Oregon, USA, August 2–4, 1996, AAAI Press, Menlo Park, CA, USA, pp. 202–207 (1996)
15. Friedman, N., Geiger, D., Goldszmidt, M.: Bayesian Network Classifiers. Mach. Learn. 29, 2-3, 131–163 (1997)
16. Breiman, L.: Bagging predictors. Machine Learning 24, 2, 123–140 (1996)
17. Freund, Y., Schapire, R.E.: Experiments with a new boosting algorithm. In: Thirteenth International Conference on Machine Learning, San Francisco, pp. 148-156 (1996)
18. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA Data Mining Software: An Update. SIGKDD Explorations 11, 1, 10–18 (2009)
19. Bader, G.D., Donaldson, I., Wolting, C., Ouellette, B.F., Pawson, T., Hogue, C.W.: BIND: the Biomolecular Interaction Network Database. Nucleic Acids Res. 29, 1, 242–245 (2001)
20. Chandonia, J.–M., Hon, G., Walker, N.S., Conte, L.L., Koehl, P., Levitt, M., Brenner, S.E.: The ASTRAL Compendium in 2004. Nucleic Acids Res. 32, D189–D192 (2004)