BIOINFORMATICS ORIGINAL PAPER

Vol. 27 no. 4 2011, pages 487–494
doi:10.1093/bioinformatics/btq700

Structural bioinformatics

Advance Access publication December 17, 2010

DROP: an SVM domain linker predictor trained with optimal features selected by random forest

Teppei Ebina[1], Hiroyuki Toh[2,*] and Yutaka Kuroda[1,*]

[1]Department of Biotechnology and Life Science, Tokyo University of Agriculture and Technology, 12-24-16 Nakamachi, Koganei-shi, Tokyo 184-8588, Japan and [2]Computational Biology Research Center, AIST Tokyo Waterfront Bio-IT Research Building 2-4-7 Aomi, Koto-ku, Tokyo, 135-0064, Japan

Associate Editor: Anna Tramontano

ABSTRACT

Motivation: Biologically important proteins are often large, multidomain proteins, which are difficult to characterize by high-throughput experimental methods. Efficient domain/boundary predictions are thus increasingly required in diverse area of proteomics research for computationally dissecting proteins into readily analyzable domains.

Results: We constructed a support vector machine (SVM)-based domain linker predictor, DROP (Domain linker pRediction using OPtimal features), which was trained with 25 optimal features. The optimal combination of features was identified from a set of 3000 features using a random forest algorithm complemented with a stepwise feature selection. DROP demonstrated a prediction sensitivity and precision of 41.3 and 49.4%, respectively. These values were over 19.9% higher than those of control SVM predictors trained with non-optimized features, strongly suggesting the efficiency of our feature selection method. In addition, the mean NDO-Score of DROP for predicting novel domains in seven CASP8 FM multidomain proteins was 0.760, which was higher than any of the 12 published CASP8 DP servers. Overall, these results indicate that the SVM prediction of domain linkers can be improved by identifying optimal features that best distinguish linker from non-linker regions.

Availability: DROP is available at http://tuat.ac.jp/~domserv/DROP.html

Contacts: toh-hiroyuki@aist.go.jp; ykuroda@cc.tuat.ac.jp

Supplementary information: Supplementary data are available at *Bioinformatics* online.

Received on June 22, 2010; revised on November 9, 2010; accepted on November 27, 2010

1 INTRODUCTION

Biologically significant proteins are often large and consist of many domains, which make them difficult to characterize by high-throughput experimental methods (Brenner, 2000; Christendat *et al.*, 2000; Yokoyama *et al.*, 2000). Efficient methods for dissecting proteins into structural domains that can be readily analyzed are gaining practical importance in proteomics research (Hondoh *et al.*, 2006). Experimental approaches for identifying structural domains are mostly based on limited proteolysis, but these methods require significant quantities of proteins, and large amount of time and effort. Computational domain prediction methods are thus being actively investigated (Joshi, 2007).

Methods that can predict domain regions without using sequence similarity to an existing domain cataloged in reference databases such as Pfam (Bateman *et al.*, 2002) and PROSITE (Hulo *et al.*, 2004) are particularly useful, as they may lead to the discovery of 'novel' domains, which are preferred targets of proteomics projects. Many domain prediction methods first detect domain boundaries or linkers, and in turn assign the location of the domain regions. This strategy takes advantage of the local nature of the boundary/linker sequence characteristics (Dumontier *et al.*, 2005; George and Heringa, 2002; Suyama and Ohara, 2003; Tanaka *et al.*, 2003). Initially, simple domain linker predictors that use variations in amino acid composition between domains and domain linkers were proposed [e.g. UMA (Udwary *et al.*, 2002), DomCut (Suyama and Ohara, 2003), Armadillo (Dumontier *et al.*, 2005) and DLiP (Tanaka *et al.*, 2006)]. More recently, domain linker prediction performances were improved by the use of machine-learning methods (Ebina *et al.*, 2009; Miyazaki *et al.*, 2002; Miyazaki *et al.*, 2006; Ye *et al.*, 2008). Additionally, position specific scoring matrix (PSSM) could further improve the performances of domain linker predictions [e.g. Nagarajan's method (Nagarajan and Yona, 2004), CHOPnet (Liu and Rost, 2004) and PPRODO (Sim *et al.*, 2005)].

Though the ability of machine-learning methods for predicting domain linkers appears well established, their performances might be further improved by selecting optimal features for distinguishing linkers from non-linkers. The previously derived domain linker properties (Ebina *et al.*, 2009; Tanaka *et al.*, 2006) combined with PSSM elements as well as 544 recently cataloged amino acid properties (Kawashima *et al.*, 2008) would represent a vast feature space from which the best or the nearly best subsets could be searched. However, no systematic search from such a large number of features has yet been reported, and when systematic searches were carried out, they were applied to feature sets of modest sizes (Ye *et al.*, 2008). This is probably because a huge number of feature combinations need to be tested by trial-and-error, which requires a significant amount of computational time. Random forest, which is based on random sampling, could potentially provide a method for rapidly screening the optimal features (Saeys *et al.*, 2007). It was originally developed as an ensemble classifier based on a collection of decision trees, and with each decision tree, randomly chosen

*To whom correspondence should be addressed.

features are scored according to their importance for classifying vectors into, e.g. linkers and non-linkers.

Here, we report a domain linker predictor based on a support vector machine (SVM) trained with optimal features, DROP (Domain linker pRediction using OPtimal features). We selected optimal features from a set of 3000, which included PSSMs and over 2000 physicochemical properties, using a random forest algorithm complemented by a stepwise selection protocol. The selected features were mostly related to secondary structures, PSSM elements of hydrophilic residues and prolines. DROP performances were superior to previously developed domain linker predictors trained without systematic optimization of the features. In addition, the efficiency of DROP for predicting novel domains was confirmed by predicting linkers in CASP8 FM multidomain protein targets.

2 METHODS

2.1 Domain linker dataset

We constructed a domain linker dataset according to our previously reported protocol (Ebina *et al.*, 2009; Tanaka *et al.*, 2006). According to our definition, a domain linker is a loop region separating two structural domains, and a structural domain is defined as a protein fragment that can fold in isolation. Domain boundary sequences containing α-helices or β-strands were not included in our study, because their amino acid compositions differ from domain linkers formed by coils only (George and Heringa, 2002). Discontinuous domains were also discarded from our dataset as our definition of a structural domain was originally motivated by the practical need of detecting autonomously folded novel protein domains, whose crystal or solution structure could be readily analyzed by biophysical methods (Hondoh *et al.*, 2006; Kuroda *et al.*, 2000; Miyazaki *et al.*, 2006). Our domain linker dataset, DS-All, contained 169 protein sequences, with a maximum sequence identity of 28.6%, and 201 linkers.

2.2 Vector encoding

Residues were encoded into a 3000-dimensional real-valued vector, where each element represented a different property (properties are described in Supplementary Methods). Vector elements were assigned to the following features: 544 amino acid indices describing physicochemical properties (Kawashima *et al.*, 2008), 20 PSSM elements (see PSSM construction details in Supplementary Methods), three probabilities of secondary structure (PSS) by PSI-PRED (Jones, 1999), two α-helix/β-sheet core propensities (Chou and Fasman, 1978), one sequential hydrophobic cluster index (Coeytaux and Poupon, 2005), sequence complexity as defined by Shannon's entropy (one element; Shenkin *et al.*, 1991), one expected contact order (Garbuzynskiy *et al.*, 2004), amino acid compositions (20 elements), three domain/coil/linker propensity indices (Suyama and Ohara, 2003), two linker likelihood scores (Tanaka *et al.*, 2006) and three newly defined scores quantifying the amino acid composition similarity between domain and linker regions (see Supplemental Methods). Vector elements were averaged with windows of ±5, ±10, ±15 or ±20 residues around the considered residue for including local and semi-local information into the vectors. For each residue, a 3000-dimensional real-valued vector [(544+20+3+2+1+1+1+20+3+2+3) features × (4 averaging window size + un-averaged element) = 3000] was generated. The total number of the vectors encoding linkers and domains were, respectively, 2230 and 52 335.

2.3 Random forest feature selection

We first assessed the vector elements using the mean decrease Gini index (MDGI) calculated by random forest [R-Random Forest package (Liaw and Wiener, 2002); http://cran.r-project.org/]. The MDGI represents the importance of individual vector elements for correctly classifying a residue into linker and non-linker regions. The MDGI was calculated by classifying 200 randomly selected linker vectors and 200 non-linker vectors, and the mean MDGI was calculated as the averaged MDGI over 100 trials. The mean MDGI *Z*-Score of each vector element was calculated as:

$$\text{MDGI } Z\text{-Score} = \frac{x_i - \bar{x}}{\sigma}$$

where x_i is the mean MDGI of the *i*-th feature; and σ is the SD of all mean MDGI. Vector elements with MDGI *Z*-Score larger than 3.0 were selected as optimal feature candidate-1 (OFC-1) and those with MDGI *Z*-Score between 2.0 and 3.0 were selected as OFC-2.

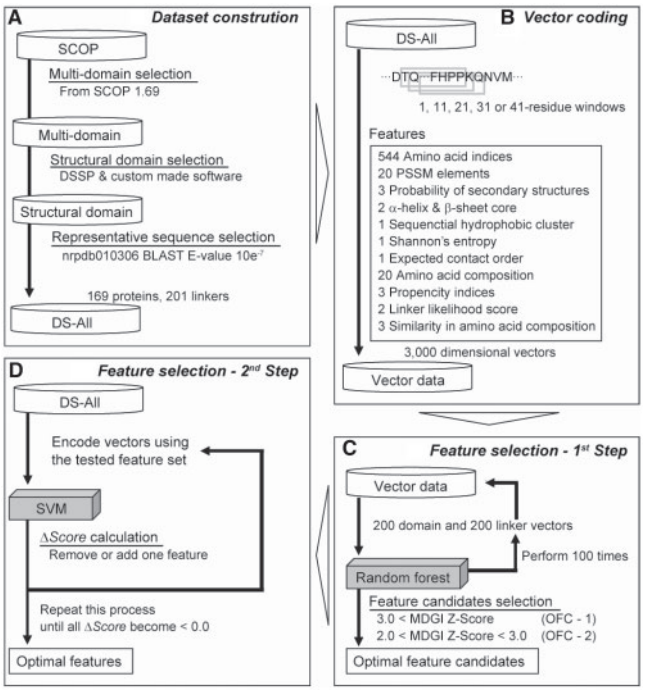

Fig. 1. Schematic representation of the feature selection. (**A**) Dataset Construction: we first constructed a domain linker dataset (DS-All), which contained 169 strictly selected multidomain protein sequences and 201 linkers. (**B**) Vector Encoding: we encoded the amino acid sequences using 3000 features representing PSSMs, PSSs and other amino acid's physicochemical properties. (**C**) Feature Selection—first step. The first step of the feature selection was performed using the mean MDGI computed with a random forest algorithm. We selected 54 features subdivided into 22 features classified according to their mean MDGI values as OFC-1 and 32 features as OFC-2. (**D**) Feature Selection—second step. SVM-based domain linker predictors were trained with various combinations of OFC-1 and OFC-2 features candidates using a stepwise selection. At each round of stepwise selection, we selected the feature set with the highest Δ*Score* and repeated the selection process until Δ*Score* became negative for all SVMs (i.e. when the SVM was not improved by adding or removing any one feature).

2.4 Stepwise feature selection

We performed a stepwise selection by constructing and assessing several SVM linker predictors trained with different sets of optimal feature candidates. The stepwise selection was performed by training an original SVM with an original data set, which was OFC-1 for the first round of the stepwise selection (Fig. 1D). The performances of the original dataset were compared, using Δ*Score*, to that of test SVMs (54-test SVMs). The test SVMs were trained using a test dataset consisting of either the original dataset from which one feature was removed (22 test SVMs in the first round)

or the original dataset to which one feature from a feature candidate dataset, which was OFC-2 for the first round, was added (32 test SVMs in the first round). The best test SVM, as evaluated by $\Delta Score$, was selected as the original SVM for the next round of selection ($\Delta Score$ is described in the next paragraph). Features removed from an original set during a selection round were included in the test dataset of the next round, so that a feature discarded in an early round of the stepwise selection could still appear in the final SVM. Consequently, the number of tested feature combinations, and thus of SVMs, remained constant at any round of the stepwise selection, as they corresponded to the number of features contained in the original dataset plus those in the test dataset (54 test SVMs, in our case). The stepwise selection process was repeated until $\Delta Score$ became negative for all tested SVMs (i.e. the original SVM was better than any newly tested SVMs).

The prediction performance was evaluated by defining a score difference ($\Delta Score$) defined as:

$$\Delta Score = T_{\text{prec}} \times T_{\text{sens}} \times T_{\text{AUC}} - O_{\text{prec}} \times O_{\text{sens}} \times O_{\text{AUC}}$$

where O_{prec} and O_{sens} are, respectively, the precision and the sensitivity of the original SVM predictor; T_{prec} and T_{sens} are the corresponding values of the test SVM. T_{AUC} and O_{AUC} are the area under curve (AUC) (see next section) of the test SVM and the original SVM predictor, respectively. We assessed the SVMs using $\Delta Score$ because it is easy to calculate and useful for simultaneously monitoring the changes of all of the three parameters. $\Delta Score$ becomes 0.0 when the prediction performances, as assessed by the product of the prediction's precision, sensitivity and AUC, remain unchanged; 1.0 when the test SVM is perfect and the original one fails completely; and -1.0 in the opposite case. We defined $\Delta Score$ using the product of T_{prec}, T_{sens} and T_{AUC} rather than their sum, because their increase/decrease increments differed significantly (T_{AUC} variation is about 1/10 of that of T_{prec} or T_{sens}).

2.5 SVM parameter optimization

We used the SVM$^{\text{light}}$ package with a RBF kernel for the classifiers (Joachims, 1999). Residues were encoded using the above identified optimal feature values as their elements. The SVM parameters (γ, C and E) were optimized using SVMLab software (http://rubygems.org/gems/ svmlab) with a dataset of 400 randomly selected vectors containing the same number of linker and non-linker vectors. The smoothing window size was optimized according to the AUC. The AUC is the area under the receiver operating characteristic (ROC) curve, which is a plot of R_{FP} against R_{TP}, where R_{FP} is the ratio of the number of correctly classified linker residues to the total number of residues classified as linker residue, and R_{TP} is the ratio of the number of correctly classified linker residues to the total number of linker residues. ROC curves were calculated for each SVM predictions with smoothing window sizes of 1, 5, 9 and 13 residues with a five-fold cross-validation test.

2.6 Domain linker prediction

The raw SVM output values, which represent a residue's domain linker propensity, were smoothed using 5-residue moving averages. The region with the highest smoothed output value was predicted as a linker if its output value was larger than the threshold value (TV). The default TV was chosen so as to maximize $R_{TP} - R_{FP}$.

2.7 Prediction assessment

The prediction performances were assessed using the sensitivity and the precision of the prediction. Sensitivity is the ratio of correctly predicted linkers to all of the structure-derived linkers listed in DS-All. Precision is the ratio of correctly predicted linkers to all of the predicted linkers. The prediction was defined as correct when the predicted linker residue with the highest SVM value overlapped with a structure-defined linker residue. This criterion is significantly more stringent than those previously used for assessing domain linker prediction (Dumontier, *et al.*, 2005; Ebina, *et al.*, 2009; Miyazaki, *et al.*, 2002; Sim, *et al.*, 2005; Tanaka, *et al.*, 2006).

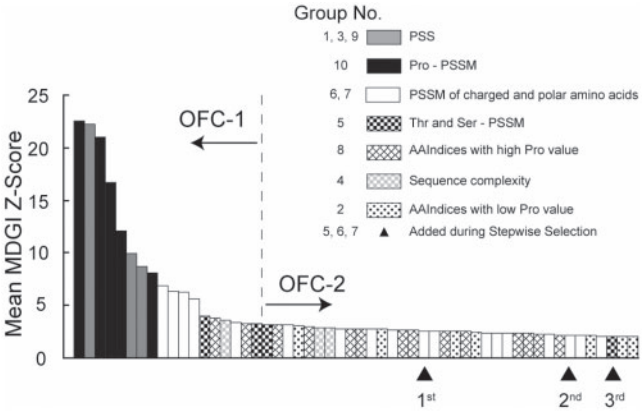

Fig. 2. Mean MDGI *Z*-Score. The bar represents the feature's mean MDGI *Z*-Score. The feature's grouping is the same as the clustering reported in Supplementary Figure S-2, except that we consolidated PSS-E, -C and -H into PSS. MDGI *Z*-score values are listed in Supplementary Table S-2. DROP used all of the OFC-1 features and the three features, which were added during the stepwise selection and are indicated by arrows.

Additionally, we assessed the performances of DROP using the average overlapped score (AOS; Jones *et al.*, 1998), and the normalized domain overlap (NDO) score (Tai *et al.*, 2005). The AOS is the ratio of correctly assigned residue number to the total number of residues. The AOS of DROP was calculated by assuming a prediction as correct when it coincided with the CAFASP4 assignment. The NDO-Score is useful as it provides a single value that evaluates (penalize/prioritize) both over- and under-predictions. The NDO-Score and AOS were also used to compare the prediction performances of DROP to those of other domain boundary predictors using CAFASP4 DP targets (http://www.cs.bgu.ac.il/~dfischer/CAFASP4/) and CASP8 (Ezkurdia, I. *et al.*, 2009) protein targets including FM (Free Modeling) domains.

3 RESULTS

3.1 Selection of optimal feature candidates by random forest

DROP is a domain linker predictor trained with 25 optimal features that best distinguish linkers from non-linkers (Supplementary Table S-1, Table S-2 and Fig. 2). The 25 features were selected in two steps. In the first step, we used a random forest protocol and evaluated the importance of 3000 features for distinguishing linker from non-linker regions using the mean MDGI *Z*-Score (Fig. 1). Fifty four features with *Z*-Scores >2.0 were selected as optimal feature candidates (Supplementary Table S-1 and Figure S-2).

In order to provide insight into the feature's charateristics, we clustered the optimal feature candidates according to their vector elements using a complete linkage algorithm. The feature candidates clustered into 10 groups (Supplementary Table S-1 and Figure S-2): predicted secondary structure element (PSS) by PSI-PRED (Jones, 1999) (Groups 1, 3 and 9 corresponding to PSS-H, -E and -C, respectively); indices that attribute a low value to Pro (Group 2); amino acid indices that attribute a high value to Pro [Group 8, which included a linker likelihood score defined in our previous reports (Tanaka *et al.*, 2006)]; sequence complexity and linker's amino acid composition (Group 4); PSSM elements of Ser and Thr (Group 5); PSSM elements of His and Arg with small window

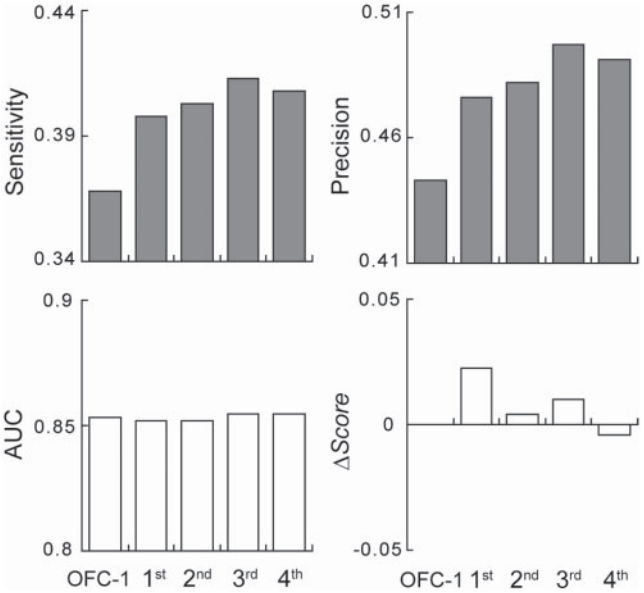

Fig. 3. Improvement of the prediction performances during the stepwise selection. The vertical axes indicate the prediction performances of the SVM with the highest $\Delta Score$ in the round of the stepwise selection indicated on the horizontal axes. The performances were estimated using a five-fold cross-validation test.

sizes (Group 6); PSSM elements of charged and polar amino acid (Group 7); and PSSM elements of Pro (Group 10).

3.2 Stepwise selection

The second step of our selection protocol was a stepwise selection, where features that decreased the prediction performances were removed whereas those that increased them were added. To this end, the above set of 54 feature candidates was divided into 22 features with *Z*-Scores larger than 3.0 (OFC-1), and 32 features with *Z*-Scores between 2.0 and 3.0 (OFC-2; Supplementary Table S-1, Table S-2 and Fig. 2). OFC-1 was used as the initial original training set (see 'Methods' section). Three features were added in four rounds of stepwise selection (Figs 2 and 3) and no feature was removed. The added features were PSSM elements of Glu, Arg and Ser with 21-, 41- and 21-residue windows, respectively.

3.3 Performance improvements by choosing optimal features

The efficiency of our feature selection was tested using five SVM domain linker predictors trained with various combinations of features. Besides DROP, trained with the 25 optimal features, SVM-Af was trained with all 3000 features described in the 'Methods' section, and SVM-SD3.0 and SVM-SD2.0 were trained using features with *Z*-Scores larger than 3.0 and 2.0, respectively. Further, SVM-PeP was trained using only the PSSM (17 elements) and PSS (3 elements) elements among the 25 elements used in DROP.

According to a five-fold cross-validation test, the sensitivity, precision, NDO-Score and AOS of DROP were, respectively, 41.3%, 49.4%, 0.766 and 0.796. These values represented 19.9% sensitivity, 23.7% precision, 0.079 NDO-Score and 0.062 AOS improvements over SVM-Af, and were over 1.0%, 0.3%, 0.03 and 0.04 higher than the respective values of SVMs developed without stepwise feature selection (Table 1).

Table 1. Prediction performance of SVM-based domain linker predictor

Predictor	AUC	Sensitivity	Precision	NDO	AOS
DROP	0.854	0.413	0.494	0.766	0.796
DROP-SD5.0	0.850	0.428	0.515	0.774	0.809
DROP-SD8.0	0.840	0.418	0.503	0.756	0.787
SVM-PeP	0.843	0.403	0.491	0.763	0.777
SVM-SD3.0	0.853	0.373	0.446	0.757	0.791
SVM-SD2.0	0.852	0.353	0.420	0.603	0.792
SVM-Af	0.784	0.214	0.257	0.687	0.734
Random	–	0.050	0.060	0.697	0.629

The performances were measured using the first-ranked prediction with a five-fold cross-validation test. All the SVMs were trained using DS-All, but encoded with different features (details are shown in the 'Results' section). DROP was trained with the 25 optimal features listed in Supplementary Table S-1, and DROP-SD5.0 and DROP-SD8.0 were trained with 15 and 9 optimal features, respectively (see 'Results' section). SVM-Pep was trained using 20 PSSM and PSS features, which were obtained by removing non-PSSM and non-PSS features from the 25 optimal features. SVM-SD3.0 and -SD2.0 were trained, respectively, with 22 OFC-1 features, and 56 features with mean MDGI *Z*-Scores larger than 2.0. SVM-Af was trained with all 3000 features. A random guess was performed according to the same prediction protocol as for the SVM-based predictors but using a randomly selected 11-residue region in place of the predicted region. For each protein, the random selection was performed 1000 times and the results were averaged.

3.4 Dependence of the prediction performances on the initial feature set of the stepwise selection

To assess the prediction dependency on the initial state of the stepwise selection, we constructed two additional domain linker predictors, DROP-SD5.0 and DROP-SD8.0. DROP-SD5.0 was trained with 15 optimal features derived from OFC-1a and OFC-2a. OFC-1a and OFC-2a contained, respectively, features with MDGI *Z*-Scores larger than 5.0 and between 2.0 and 5.0. Similarly, DROP-SD8.0 was trained with nine optimal features derived from OFC-1b and OFC-2b, which contained features with MDGI *Z*-Scores larger than 8.0 and between 2.0 and 8.0, respectively.

The optimal features in DROP-SD5.0 included PSSM elements of Pro, Lys, Arg and Thr, PSSs, and the Shannon's entropy (Supplementary Table S-1), whereas those in DROP-SD8.0 included PSSM elements of Pro, PSSs, Shannon's entropy and one AAIndex (BLAM930101). Overall, the sensitivity and precision of DROP-SD5.0 were slightly higher than those of DROP and DROP-SD8.0 (Table 2).

3.5 Comparison to publicly available predictors

We compared the prediction performances of DROP to those of publicly available domain boundary predictors that, similarly to DROP, do not use sequence similarity to domain databases, such as SMART, Pfam PROSITE etc (Fig. 4). DROP's sensitivity and precision were, respectively, at least 11.4 and 12.1% higher than those of other predictors when assessed with DS-All. In addition, the AOS and NDO-Score of DROP were also higher than those of all other predictors except PPRODO.

Furthermore, we assessed the characteristics of DROP with an independent set of proteins, BDS (benchmarking dataset), which contained multidomain proteins and single domain proteins in a

Table 2. Dependency of prediction performances on the averaging window size of the vector elements

Window size	AUC	Sensitivity	Precision
1	0.823	0.348	0.414
11	0.800	0.338	0.433
21	0.799	0.299	0.382
31	0.776	0.244	0.325
41	0.744	0.204	0.283
All	0.854	0.413	0.494

Predictors were constructed using the same protocol as that of DROP, but by averaging the vector element using a single window size. The prediction performances were calculated with a five-fold cross-validation test using DS-AII.

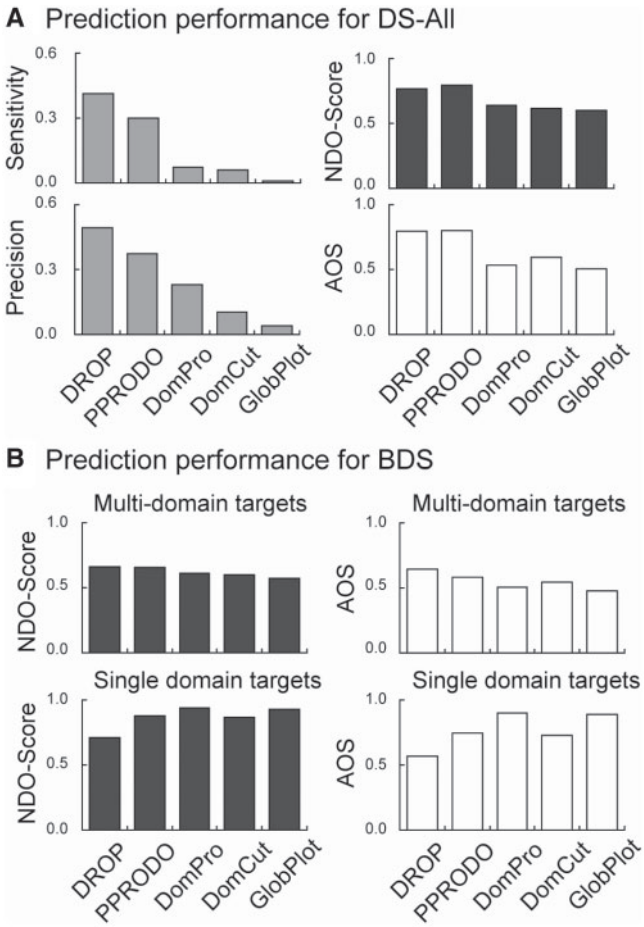

Fig. 4. Prediction performances of four publicly available domain boundary/linker predictors that do not use sequence similarity to domain database. Predictions were performed with (**A**) DS-All and (**B**) BDS. The predictions of PPRODO, DomPro and GlobPlot were performed using their downloadable packages. DomCut was reconstructed according to the methods described in the original article. The prediction performances of PPRODO and DomCut were calculated using the first-ranked predictions. We defined the prediction peaks of DomPro and GlobPlot as the center of the predicted boundary residues. DROP's performances for DS-All were calculated using a five-fold cross-validation test.

proportion and a size distribution similar to those computed in whole genomes (detail of the BDS construction are shown in the Supplementary Methods). BDS contained 275 multidomain and 183 single-domain protein sequences, and their ratio was set to 60.0%, which is an estimate calculated over several genomes (Yeats *et al.*, 2010). The AOS and NDO-Score of DROP calculated with BDS were the highest with, respectively, 0.661 and 0.646 (Supplementary Figure S-5).

Furthermore, we found very similar results when we assessed DROP's performance using CAFASP4 protein targets, which contains 27 multidomain and 51 single-domain proteins, and seven CASP8 FM (Free modeling) multidomain proteins (http://predictioncenter.org/casp8/). DROP's AOS calculated with CAFASP4 multidomain proteins was 0.666, which was higher than any of the 12 CAFASP4 DP predictors except Robetta-Ginzu (Fig. 5A). Additionally, the mean NDO-Score of DROP was 0.760, which was also higher than any value reported for the 12 CASP8 DP servers (Fig. 5C).

4 DISCUSSION

Feature selection can significantly influence the performances of machine learning-based predictors (Kernytsky and Rost, 2009). However, because of the computational time required for performing an exhaustive search, features are usually selected by trial-and-error from a relatively small set of intuitively pre-selected features (Ye *et al.*, 2008). Our two step approach, which combines random forest and a stepwise selection, provides a realistic approach for selecting an optimal set of features within a reasonable computational time. In our present setting, the random forest and the stepwise selection assessed, respectively, 3000 features in 7 h and 54 features in 69 h (for 4 rounds) on an 8 Xeon processors Linux server.

Stepwise selection is not an exhaustive search and may overlook the very best set of features, but it assesses a sufficient number of combinations to yield one of the best (54 feature combinations were tested in each round of stepwise selection). Furthermore, the correlated effect of multiple features is partially assessed by the stepwise selection, whereas a simple forward or backward selection would merely assess the effect of adding or removing a single feature. Indeed, the re-inclusion of a feature that is discarded from the training set in a previous round of selection into the next round's test dataset enables the recovery of a feature that might not be useful with a given set of features but becomes useful when combined with a different set of features.

Another advantage related to our fast two-step feature selection approach is that it enables to test several averaging windows: we encoded 600 properties averaged with five different windows into a 3000-dimensional vector. Indeed, it is not usual to include elements containing the same properties averaged over different window sizes (Hirose,S. *et al.*, 2007). One possible reason for the scarce use of multiple averaging windows is the time-consuming search, which might counterbalance the advantages of including additional window sizes, especially when the search space is as large as in our case. However, the prediction performances of DROP significantly improved by using multiple size windows (Table 2), and this suggest that using multiple averaging window sizes might be a useful approach for improving predictions of properties determined by both local and non-local nature.

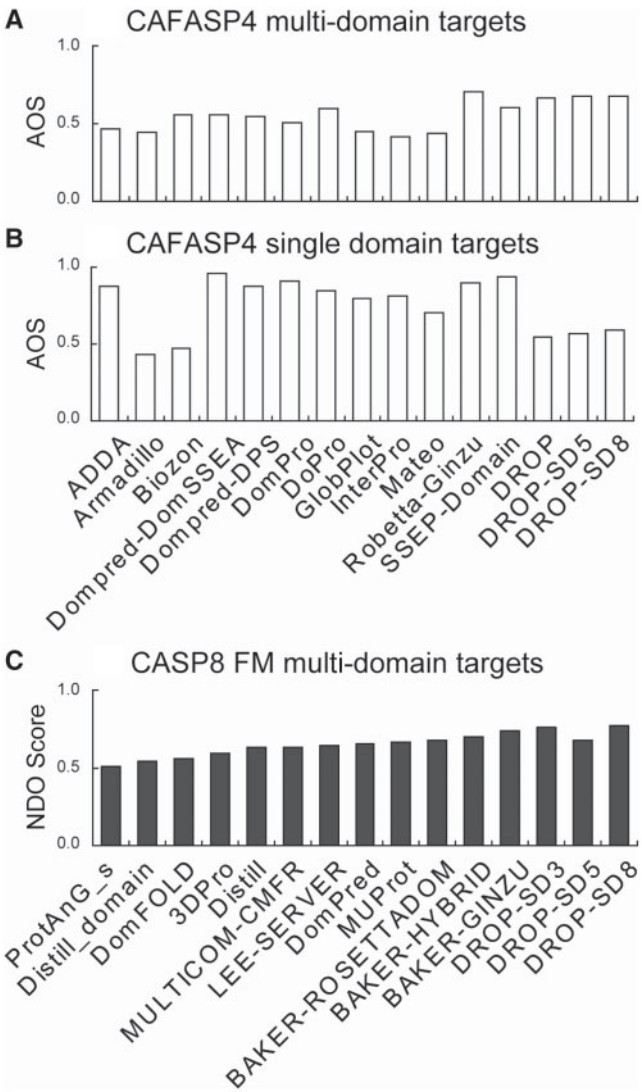

Fig. 5. Comparison with other domain boundary predictors. Mean AOS calculated using (**A**) 26 multidomain and (**B**) 51 single-domain CAFASP4 targets. (**C**) Mean NDO-Scores calculated using CASP8 FM multidomain proteins. DROP's NDO-Score was calculated by considering the residue with the highest SVM output value and located within the predicted linker as the domain boundary residue separating two adjacent domains. Other predictor's AOS and NDO-Scores were adapted from the CAFASP4 and CASP8 web pages.

We also assessed the prediction performance dependency on the initial state of the stepwise selection. DROP, DROP-SD5.0 and DROP-SD8.0 stepwise selection processes were started with different initial feature sets and eventually yielded the respective 25, 15 and 9 optimal features. The discrepancies between the final numbers of optimal features indicate that the feature space is not exhaustively searched, and that 'the optimal sets' are local minima. However, in all three cases, similar, though not identical, features were chosen (Supplementary Table S-1), and the performances were improved when compared with those of DROP-Af (Table 1). Eight features, which were related to PSSM elements of Pro, PSSs and Shannon's entropy, were present in all of the three optimal feature sets.

Previous reports have shown that domain linker regions preferentially contain Pro and hydrophilic residues and/or secondary structure breaker (George and Heringa, 2002; Miyazaki *et al.*, 2002; Suyama and Ohara, 2003; Tanaka *et al.*, 2006). In particular, linkers contain more Pro and Lys but less Gly, Asp and Asn than non-linker loops. The selected optimal features for training DROP roughly corroborated these observations: the mean MDGI *Z*-Scores of Pro and Lys PSSMs and PSSs were among the highest (Supplemntary Tables S-1 and S-2 and Fig. 2). On the other hand, PSSMs of Gly and Asn were not selected. Additionally, five non-PSSM or non-PSS features, such as Shanon's entropy and domain linker likelihood, were found among the optimal features. Though their *Z*-Scores were relatively low (Supplementary Table S-2), and they were typically not used for linker prediction, they nevertheless contributed to improve the prediction performances of DROP (Table 1).

The inclusion into DROP of features not typically used for boundary prediction (two AAIndex and a linker likelihood score) may help predicting domain boundary regions in novel proteins. For instance, features that contributed most to the identification of the linker sequence in T0496, by DROP and DROP-SD8.0 but not DROP-SD5.0, were the PSSM elements of Pro, PSS-Coil, PSS-Helix and three features that were not used in training DROP-SD5.0 (Fig. 6).

In this and previous studies, we focus on loop regions separating continuous structural domains (See 'Methods' section). Our choice to discard discontinuous domains was motivated by our original and practical need of detecting autonomously folded domains (Hondoh *et al.*, 2006; Kuroda *et al.*, 2000; Miyazaki *et al.*, 2006). Additionally, domain boundary regions containing α-helices and/or β-strands, which represent a minority (Supplementary Table S-4), were also excluded from the training dataset. This is because their amino acid compositions differ from those of domain linkers formed by coils (George and Heringa, 2002), and SVMs trained with a dataset containing sequences with differing properties are likely to perform poorly. In our study, this idea was corroborated by DROP's significantly better ability to recognize domain than other predictors (Fig. 4).

The possibility of applying DROP to large protein sequence datasets and possibly whole proteomes was analyzed using BDS, CASP and CAFASP protein targets, though their domains are not strictly structural domains. The prediction performances of DROP assessed using BDS and CASP8 FM multidomain targets were the highest and second highest when assessed using CAFASP4 multidomain targets (Figs 4B, 5A and 5C). On the other hand, DROP had a tendency to overpredict domain linkers in single-domain targets of BDS and CAFASP4 (Figs 4B and 5B). Overprediction could be decreased simply by increasing the default threshold level or by including non-local features (as discussed below). However from an experimental viewpoint, over-prediction can be considered less detrimental than missing an existing domain, since the experimental cost for assessing a putative domain linker is nearly equivalent to that of testing a new domain terminus (Chikayama *et al.*, 2010). In addition, there is usually little reason to apply domain prediction to small proteins since many are single-domain proteins.

Finally, domain linker predictions that consider only local sequence characteristics cannot yield a perfect predictor. This is because domain boundaries are most likely determined by the

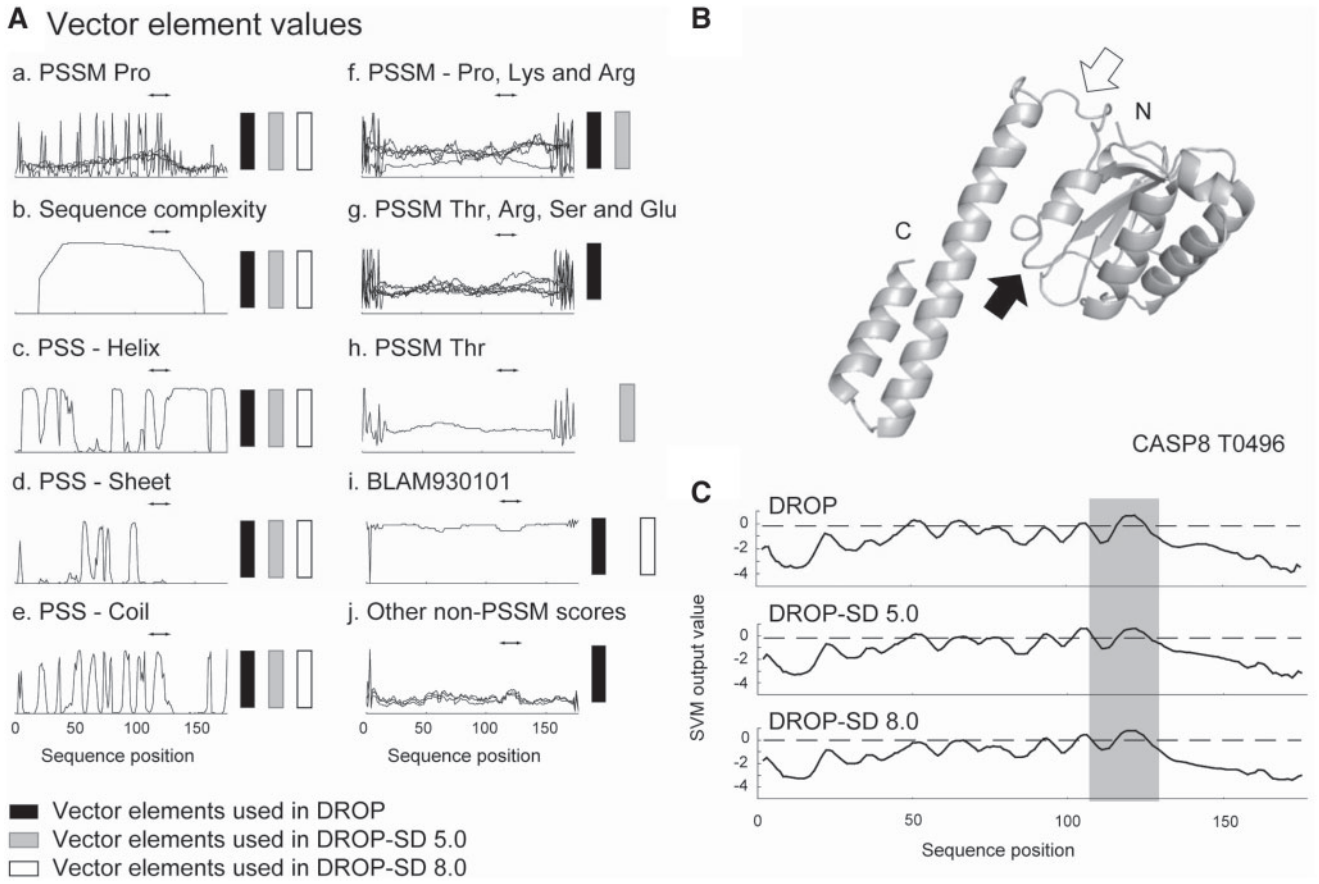

Fig. 6. Delineation of SVM outputs and prediction example of CASP8 T0496 target. (**A**) Vector element values used for DROP prediction. The horizontal axis represents the sequence position and the solid lines represent the vector element values at each residue. The horizontal arrow indicates a loop region, which overlaps with the CASP8 defined domain boundary. Black, gray and white boxes indicate that the corresponding vector elements were included in DROP, DROP-SD5.0 and DROP-SD8.0, respectively. (**a**–**e**) PSSM elements, PSSs and Shannon's Entropy with different window sizes. (**f**) PSSM elements of Pro with a 41-residue window, and of Lys and Arg. (**g**) PSSM elements of Thr, Arg, Ser and Glu. (**h**) PSSM element of Thr with residue window of 41. (**i**) AAIndex, BLAM930101, with residue window of 21. (**j**) Other non-PSSM Scores that appeared in DROP. (**B** and **C**) Prediction example of CASP8 T0496 protein. (**B**) Cartoon representation of CASP8 T0496 (PDB ID 3DEE_A) created with PyMOL (http://www.pymol.org/). The loop region that overlaps with the CASP8 defined domain boundary is indicated with an open arrow (residues 110–129, which is predicted by DROP and DROP-SD8.0). The filled arrow indicates residue 105, which is predicted by DROP-SD5.0. 'N' and 'C' represent, respectively, the N- and C-termini of the protein. (**C**) SVM-outputs calculated with DROP, DROP-SD5.0 and DROP-SD8.0. The horizontal axis represents the residue number. The broken line and the solid line indicate the TV and the smoothed SVM output value, respectively. The gray box represents the structure defined inter-domain loop region. DROP predicted residues 117–128 as a domain linker, and we assigned residue 122, which had the maximum smoothed SVM output value, as a domain boundary residue. Residues 1–121 and 123–178 were thus assigned to domain-1 and domain-2, respectively, resulting in a NDO-Score of 0.955. Similarly, residues 105 and 121 were predicted as domain boundary by DROP-SD5.0 and DROP-SD8.0, respectively. The NDO-Score of DROP-SD5.0 and -SD8.0 were, respectively, 0.740 and 0.944.

foldability of the domain region as much as by the local information encoded in the linker region. Previous reports suggest that non-local information may have a strong potential for improving domain prediction (George *et al.*, 2005; Zhang, 2009). Our random forest-based approach can readily accommodate multiple window sizes, and it may be particularly suitable for selecting optimal non-local features, such as Cα density or foldability index, because the number of candidate non-local features is anticipated to be large, probably much larger than that of local features.

In conclusion, our approach efficiently selected optimal sets of features from a large number of features, by combining two types of methods: the random forest and the stepwise selection. Using this approach, a vast feature space consisting of 3000 features was searched within a realistic computational time, which improved the performances of DROP. Constructed on such premises, we expect DROP to help analyzing the enormous amount of sequential data by assisting the discovery of novel domains.

ACKNOWLEDGEMENTS

We thank Dr Shuichi Hirose (CBRC) for discussion, Yuta Kumagai, Takao Arai, Tomohiro Furuyama, Shun Iwasaki and Ryotaro Tsuji (TUAT; Kuroda's lab) for their help with dataset construction, and Dr Frederik Johanson (Kyushu University) for advices on SVMLab software.

Funding: Japan Society for Promotion of Science (JSPS-18500225 and 21300110).

Conflict of Interest: none declared.

REFERENCES

Bateman,A. *et al.* (2002) The Pfam protein families database. *Nucleic Acids Res.*, **30**, 276–280.

Brenner,S.E. (2000) Target selection for structural genomics. *Nat. Struct. Biol.*, **7** (Suppl.), 967–969.

Chikayama, E. *et al.* (2010) Mathematical model for empirically optimizing large scale production of soluble protein domains. *BMC Bioinformatics*, **11**, 113.

Chou,P.Y. and Fasman,G.D. (1978) Prediction of the secondary structure of proteins from their amino acid sequence. *Adv. Enzymol. Relat. Areas Mol. Biol.*, **47**, 45–148.

Christendat,D *et al.* (2000) Structural proteomics: prospects for high throughput sample preparation. *Prog. Biophys. Mol. Biol.*, **73**, 339–345.

Coeytaux,K. and Poupon,A. (2005) Prediction of unfolded segments in a protein sequence based on amino acid composition. *Bioinformatics*, **21**, 1891–1900.

Dumontier,M. *et al.* (2005) Armadillo: domain boundary prediction by amino acid composition. *J. Mol. Biol.,* **350**, 1061–1073.

Ebina,T. *et al.* (2009) Loop-length-dependent SVM prediction of domain linkers for high-throughput structural proteomics. *Biopolymers*, **92**, 1–8.

Ezkurdia,I. *et al.* (2009) Assessment of domain boundary predictions and the prediction of intramolecular contacts in CASP8. *Proteins*, **77** (Suppl. 9), 196–209.

Garbuzynskiy,S.O. *et al.* (2004) To be folded or to be unfolded? *Protein Sci.*, **13**, 2871–2877.

George,R.A. and Heringa,J. (2002) An analysis of protein domain linkers: their classification and role in protein folding. *Protein Eng.*, **15**, 871–879.

George,R.A. *et al.* (2005) Scooby-domain: prediction of globular domains in protein sequence. *Nucleic Acids Res.*, **33**, W160–W163.

Hirose,S. *et al.* (2007) POODLE-L: a two-level SVM prediction system for reliably predicting long disordered regions. *Bioinformatics*, **23**, 2046–2053.

Hondoh,T. *et al.* (2006) Computer-aided NMR assay for detecting natively folded structural domains. *Protein Sci.*, **15**, 871–883.

Hulo,N. *et al.* (2004) Recent improvements to the PROSITE database. *Nucleic Acids Res.*, **32**, D134–D137.

Joachims,T. (1999) Making large-Scale SVM learning practical. In Schölkopf,B. *et al.* (eds) *Advances in Kernel Methods - Support Vector Learning*. The MIT-Press, Cambridge, MA.

Jones,D.T. (1999) Protein secondary structure prediction based on position-specific scoring matrices. *J. Mol. Biol.*, **292**, 195–202.

Jones,S. *et al.* (1998) Domain assignment for protein structures using a consensus approach: characterization and analysis. *Protein Sci.*, **7**, 233–242.

Joshi,R.R. (2007) A decade of computing to traverse the labyrinth of protein domains. *Curr. Bioinfo.*, **2**, 113.

Kawashima,S. *et al.* (2008) AAindex: amino acid index database, progress report 2008. *Nucleic Acids Res.*, **36**, D202–D205.

Kernytsky,A. and Rost,B. (2009) Using genetic algorithms to select most predictive protein features. *Proteins*, **75**, 75–88.

Kuroda,Y. *et al.* (2000) Automated search of natively folded protein fragments for high-throughput structure determination in structural genomics. *Protein Sci.*, **9**, 2313–2321.

Liaw,A. and Wiener,M. (2002) Classification and regression by randomForest. *R news*, **2**, 18–22.

Liu,J. and Rost,B. (2004) Sequence-based prediction of protein domains. *Nucleic Acids Res.*, **32**, 3522–3530.

Miyazaki,S. *et al.* (2002) Characterization and prediction of linker sequences of multi-domain proteins by a neural network. *J. Struct. Funct. Genomics*, **2**, 37–51.

Miyazaki,S. *et al.* (2006) Identification of putative domain linkers by a neural network - application to a large sequence database. *BMC Bioinformatics*, **7**, 323.

Nagarajan,N. and Yona,G. (2004) Automatic prediction of protein domains from sequence information using a hybrid learning system. *Bioinformatics*, **20**, 1335–1360.

Saeys,Y. *et al.* (2007) A review of feature selection techniques in bioinformatics. *Bioinformatics*, **23**, 2507–2517.

Shenkin,P.S. *et al.* (1991) Information-theoretical entropy as a measure of sequence variability. *Proteins*, **11**, 297–313.

Sim,J. *et al.* (2005) PPRODO: prediction of protein domain boundaries using neural networks. *Proteins*, **59**, 627–632.

Suyama,M. and Ohara,O. (2003) DomCut: prediction of inter-domain linker regions in amino acid sequences. *Bioinformatics*, **19**, 673–674.

Tai,C.H. *et al.* (2005) Evaluation of domain prediction in CASP6. *Proteins*, **61** (Suppl. 7), 183–192.

Tanaka,T. *et al.* (2003) Characteristics and prediction of domain linker sequences in multi-domain proteins. *J. Struct. Funct. Genomics*, **4**, 79–85.

Tanaka,T. *et al.* (2006) Improvement of domain linker prediction by incorporating loop-length-dependent characteristics. *Biopolymers*, **84**, 161–168.

Udwary,D.W. *et al.* (2002) A method for prediction of the locations of linker regions within large multifunctional proteins, and application to a type I polyketide synthase. *J. Mol. Biol.*, **323**, 585–598.

Yeats,C. *et al.* (2010) A fast and automated solution for accurately resolving protein domain architectures. *Bioinformatics*, **26**, 745–751.

Ye,L. *et al.* (2008) Sequence-based protein domain boundary prediction using BP neural network with various property profiles. *Proteins*, **71**, 300–307.

Yokoyama,S. *et al.* (2000) Structural genomics projects in Japan. *Nat. Struct. Biol.*, **7** (Suppl.), 943–945.

Zhang,Y. (2009) I-TASSER: fully automated protein structure prediction in CASP8. *Proteins*, **77** (Suppl. 9), 100–113.