the FEBS Journal

PROSO II – a new method for protein solubility prediction

Pawel Smialowski[1,2], Gero Doose[1], Phillipp Torkler[1], Stefanie Kaufmann[1] and Dmitrij Frishman[1,2]

1 Department of Genome Oriented Bioinformatics, Technische Universität Muenchen, Freising, Germany
2 Helmholtz Zentrum Munich, National Research Center for Environment and Health, Institute for Bioinformatics, Neuherberg, Germany

Keywords
classification; feature selection; machine learning; prediction; protein solubility

Correspondence
P. Smialowski, Department of Genome Oriented Bioinformatics, Technische Universität Muenchen
Wissenschaftszentrum Weihenstephan,
Am Forum 1, 85350 Freising, Germany
Fax: +49 8161 712186
Tel: +49 8161 712136
E-mail: pawelsm@gmail.com

(Received 13 December 2011, revised 12 March 2012, accepted 12 April 2012)

doi:10.1111/j.1742-4658.2012.08603.x

Many fields of science and industry depend on efficient production of active protein using heterologous expression in *Escherichia coli*. The solubility of proteins upon expression is dependent on their amino acid sequence. Prediction of solubility from sequence is therefore highly valuable. We present a novel machine-learning-based model called PROSO II which makes use of new classification methods and growth in experimental data to improve coverage and accuracy of solubility predictions. The classification algorithm is organized as a two-layered structure in which the output of a primary Parzen window model for sequence similarity and a logistic regression classifier of amino acid k-mer composition serve as input for a second-level logistic regression classifier. Compared with previously published research our model is trained on five times more data than used by any other method before (82 000 proteins). When tested on a separate holdout set not used at any point of method development our server attained the best results in comparison with other currently available methods: accuracy 75.4%, Matthew's correlation coefficient 0.39, sensitivity 0.731, specificity 0.759, gain (soluble) 2.263. In summary, due to utilization of cutting edge machine learning technologies combined with the largest currently available experimental data set the PROSO II server constitutes a substantial improvement in protein solubility predictions. PROSO II is available at http://mips.helmholtz-muenchen.de/prosoII.

Introduction

Protein solubility is an important prerequisite for success in many biophysical studies and industrial applications, including the production of the ever more important protein-based drugs such as antibodies, interleukins and others. Efficient production of soluble and active proteins still remains a major challenge. Many proteins heterologously expressed in *Escherichia coli* are insoluble. Solubilization attempts are plagued by relatively low success rates [1] and often lead to the loss of biological activity [2]. Various experimental approaches aimed at improving protein solubility during heterologous expression include use of weak promoters, low temperature, modified growth media [3] and fusion with solubility enhancing tags [4]. Other methods are based on large-scale screening and random mutagenesis for solubility optimization [5].

To focus experimental work on easily soluble proteins and avoid recalcitrant targets one can study recurrent patterns in amino acid sequences of soluble proteins. Under a given set of experimental conditions, including the expression host, temperature etc., protein solubility is a trait ultimately determined by its sequence. Since the most widely used method for protein production is heterologous expression in *E. coli*, it is particularly beneficial to study factors determining protein solubility upon overexpression in this specific host.

Abbreviations
AI, aliphatic index; AUROC, area under the receiver operating characteristic curve; FI, fold index; GRAVY, grand average of hydropathicity index; MCC, Matthew's correlation coefficient; PDB, Protein Data Bank; pI, isoelectric point; SVM, support vector machine.

The correlation between protein solubility and primary structure has been the subject of active research in the past 20 years. The first proposed methods for deriving protein solubility from sequence were based on the content of charged (Asp, Glu, Lys and Arg) and turn-forming (Asn, Gly, Pro and Ser) residues [4,6]. Those methods were trained on a sparse number of instances (~ 100) and therefore did not have sufficient generalization power. With the advent of structural biology initiatives high-throughput technologies and electronic data storage became a commodity in protein expression and purification endeavors. This resulted in an unprecedented increase in the amount of experimental data documenting both successes (soluble proteins) and failures (insoluble proteins). Already in 2000 Christendat and coworkers [7] examined experimental results from a pilot project on 424 non-membrane proteins from *Methanobacterium thermoautotrophicum*. They observed that insoluble proteins more frequently contained hydrophobic stretches of 20 or more residues, had lower glutamine content (Q < 4%), fewer negatively charged residues (DE < 17%) and a higher percentage of aromatic amino acids (FYW > 7.5%) than soluble proteins. In 2001 Bertone *et al.* [8] re-analyzed 562 proteins from the same organism and confirmed that a high content of negative residues (DE > 18%) and absence of hydrophobic patches were correlated with improved solubility. Moreover they showed that low content of aspartic acid, glutamic acid, asparagines and glutamine (DENQ < 16%) was coincident with protein insolubility. In a study of 10 167 proteins overexpressed in *E. coli,* Luan *et al.* [9] reported that proteins homologous to those with known structures have higher chances of being soluble. Goh *et al.* [10] studied 27 000 protein sequences from multiple organisms using the decision tree formalism and found that protein solubility is influenced by a number of primary structure features including (in decreasing order of importance) content of serine (S < 6.4%), fraction of negatively charged residues (DE < 10.8%), percentage of S, C, T or M amino acids, and length (< 516 amino acids). In 2006 Idicula-Thomas *et al.* [11] presented a solubility prediction method based on a support vector machine (SVM) trained on a small (62 soluble and 130 insoluble proteins) unbalanced data set. According to the authors the method is able to predict correctly the increase/decrease in solubility upon mutation. The method was discussed and evaluated in Smialowski *et al.* [12]. Our previous method for solubility prediction PROSO [12] was built upon ~ 14 000 instances (half soluble and half insoluble) from the structural genomics database TargetDB [13] and the Protein Data Bank (PDB) [14]. The classification algorithm was organized as a two-level structure with an SVM on the first level and a naive Bayes classifier on the second level. Proteins were represented by frequencies of k-mers of 20 amino acids and by compressed alphabets. The accuracy of this method was 71.1% as calculated using 10-fold cross-validation. Using a formalized feature evaluation algorithm [15] we identified the content of R, D, E, G, S, C, M and L to be relevant for the solubility of single and multiple domain proteins. Amongst dipeptide frequencies, five of them (RE, EG, KG, QA, HM) seem to be the most important in solubility determination.

Magnan *et al.* [16] published a solubility prediction method (SOLpro) based on 17 000 instances with a classification algorithm of the same architecture as used in PROSO (SVM on the first and naive Bayes on the second level of classifier). Similar to the PROSO approach frequencies of k-mers as well as reduced amino acid alphabets were used as input. In addition to frequencies some other sequence derived features were included: length, molecular weight, grand average of hydropathicity index (GRAVY), aliphatic index (AI), fraction of turn-forming residues, absolute charge per residue, fraction of beta, alpha and exposed residues, and the number of domains. The authors found that the best single group of features was the content of the 20 amino acids. The overall accuracy of SOLpro was 74%, but as we discuss below this result may be due to a missing correction for length distribution. In 2011 Agostini *et al.* [17] presented the CCSOL method which utilizes 28 physicochemical properties in a sliding window to predict protein solubility. Their study was based on a data set of 3034 *E. coli* proteins for which solubility was measured experimentally *in vitro* [18]. The authors found that features most important for solubility were disorder, coil, hydrophilicity, β–sheet and α–helix.

In this paper we present a novel prediction server PROSO II constructed to assess the chance of a protein being soluble upon heterologous expression in *E. coli.* Our method exploits the experimentally measured solubility of 82 299 proteins, five times more data than any other available method, and thus provides better coverage of the currently known protein sequence space. PROSO II employs a model based on a logistic function and an adapted Parzen window algorithm trained on experimental data extracted from the pepcDB [19] and PDB [14] databases. It is superior in discriminatory ability (accuracy 75.4%) on data with the same soluble/insoluble class distribution as observed in pepcDB. Additionally, we analyze the significance of features and their correlation with protein solubility.

Results and Discussion

Performance of primary classifiers

The performance of the primary classifiers and of the entire PROSO II method is presented in Table 1. The primary classifier built on frequencies of dimers was the single best performing method in this comparison with accuracy, Matthew's correlation coefficient (MCC) and area under the receiver operating characteristic curve (AUROC) equal to 68.6%, 0.379 and 0.753, respectively. It was followed by the Parzen window method. The low performance of the sequence-length-based model confirmed the absence of any length bias in our data sets. Simple protein features such as isoelectric point (pI), length, fold index (FI), AI and GRAVY did not perform well. Even their combination reached merely 53.1% accuracy and an MCC of 0.063. Therefore, we believe that none of these simple features is significantly correlated with protein solubility.

In addition to k-mers of 20 amino acids we also tested (data not shown) compressed alphabets Sol07, Sol14, Sol17, We2 and We4 from our previous work [12]. Their performance was similar but not better than the use of the original 20 letter amino acids. Therefore, for simplicity we decided not to use them. Nevertheless, it should be noted that they were not much worse and that in the case of huge amounts of data it could be sensible to switch to compressed alphabets for computational reasons. We also tested frequencies of tripeptides using 25% sequence identity clustered data (data not shown) and decided not to use them as they did not perform substantially better than dimers and the use of tripeptides required much more computational power.

Feature selection results

The set of the best performing k-mers of length 1 and 2 was selected using the wrapper method [15] as described in Methods. There were 18 amino acid frequencies that were correlated with protein solubility (R, N, D, C, Q, E, G, H, I, K, M, F, P, S, T, W, Y, V). This result of feature selection combined with relatively low performance of the single amino acid based classification implies that protein solubility cannot be attributed to a single or a small group of amino acid frequencies.

Sixteen out of 400 dipeptide frequencies were selected as most important for model performance: AK, CV, EG, GN, GH, HE, IH, IW, MR, MQ, PR, TS and WD. Eight out of 13 selected dimers contain electrically charged (negatively, D, E; positively, R, H, K) side chains, which is in good agreement with Wilkinson–Harrison work [4]. Other frequently occurring amino acid groups include hydrophobic aromatic (F, W, H, Y) and hydrophobic aliphatic (I, M, P) residues. Five out of 13 dimers contain aromatic amino acids. As demonstrated before by Christendat *et al.* [7] a high percentage of aromatic residues FYW > 7.5% is coincident with insolubility. Also a high content of hydrophobic dimers seems to be an important factor for protein solubility although we removed from our data all proteins containing even a single transmembrane segment (as predicted by TMHMM 2.0c).

Table 1. Performance of different methods for predicting protein solubility. We evaluated both primary classifiers and the complete PROSO II method at a level of sequence identity equal to 90%. An additional holdout set was used to further examine the performance of threshold-adjusted PROSO II. We also evaluated how strongly protein solubility is correlated with simple sequence features: AI, FI, GRAVY and pI. All values are provided for 90% identity clustered data if not stated differently in the column header. Except for the holdout set all values presented were obtained using stratified 10-fold cross-validation as described in the text. The letters P and N in parentheses refer to the positive (soluble) and negative (insoluble) classes, respectively.

	Amino acid frequencies	Dipeptide frequencies	Parzen window	Length	pI	AI, FI, GRAVY, pI	PROSO II	PROSO I on holdout data set	SOLpro on holdout data set	PROSO II on holdout data set
Instances	82 299	82 299	82 299	82 299	82 299	82 299	82 299	1764	1764	1764
Accuracy	59.7	68.6	64.3	52.9	53.7	53.1	71.0	66.0	65.0	75.4
MCC	0.194	0.379	0.286	0.081	0.075	0.063	0.421	0.127	0.068	0.390
TP rate	0.598	0.783	0.617	0.184	0.481	0.467	0.754	0.459	0.389	0.731
TN rate	0.596	0.589	0.669	0.874	0.594	0.596	0.666	0.700	0.698	0.76
Precision (P)	0.597	0.655	0.651	0.59	0.542	0.536	0.693	0.234	0.192	0.377
Gain (P)	1.194	1.311	1.302	1.189	1.084	1.072	1.386	1.406	1.233	2.263
Precision (N)	0.597	0.730	0.636	0.517	0.534	0.528	0.730	0.866	0.861	0.934
Gain (N)	1.194	1.461	1.272	1.035	1.067	1.055	1.460	1.039	1.020	1.121
AUROC	0.635	0.753	0.688	0.520	0.541	0.548	0.785	–	–	–

Performance of the entire method

The performance of our entire method (PROSO II) was first evaluated by 10-fold cross-validation (Table 1). It achieved an accuracy of 71%, AUROC of 0.785, MCC of 0.421, sensitivity of 0.754, specificity of 0.666, precision (soluble) of 0.693, gain (soluble) of 1.383, precision (insoluble) of 0.73 and gain (insoluble) of 1.46.

To further improve our method we adjusted the threshold of the classifier to 0.6 using a separate data set derived as described in Methods to account for non-equal distribution of soluble and insoluble instances in pepcDB. In other words any query sequence with a PROSO II score ≥ 0.6 is classified as soluble. This threshold was selected to balance sensitivity and specificity. Based on our analysis of pepcDB (Methods) we see that approximately one out of six proteins is soluble upon heterologous expression in *E. coli*. In the next step we tested PROSO II using a holdout set of sequences with this real-life class distribution (one in six proteins are soluble). The holdout set was not utilized at any point of method development. On these data our server attained higher performance values (Table 1) compared with another recent method – SOLpro [16]: an accuracy of 75.4% versus 65%, MCC of 0.39 versus 0.068, sensitivity of 0.731 versus 0.389, specificity of 0.759 versus 0.698, precision (soluble) of 0.377 versus 0.192, gain (soluble) of 2.263 versus 1.233, precision (insoluble) of 0.934 versus 0.861, and gain (insoluble) of 1.121 versus 1.02. High values of MCC and gain suggest that PROSO II is a better performing and better balanced method. Given the unequal class distribution in the holdout set MCC or gain are much better suited to quantify classifier efficiency than accuracy.

In order to better assess the accuracy of SOLpro we studied its input data and features. The authors of SOLpro trained and evaluated a two-level k-mer frequencies based model (similar to the one reported by our group in 2007 [12]) over 17 407 sequences (half soluble, half insoluble). They reported an accuracy of 74.1% and an AUROC of 0.742. Because the authors chose not to adjust the data for sequence length distribution we tested whether any part of the performance can be attributed to this single feature. To this end we calculated sequence length for all proteins from the SOLpro data set and used it as input to build a solubility classifier. A naive Bayes model [20] was trained and evaluated using 10-fold cross-validation. This experiment yielded an accuracy and AUROC of 58.3% and 0.618, respectively. Adding pI values as a second input feature led to a slight increase in accuracy (58.8%) and AUROC (0.621). This means that between one-third (accuracy-wise) and a half (AUROC-wise) of the SOLpro performance can be achieved using a simple naive Bayes classifier and the sequence length as the only input. The same experiment on our data led to an accuracy of 52.9% and an AUROC of 0.52. We believe that in real-life applications, when scientists need to select a target protein amongst orthologous sequences originating from different species or protein variants designed to improve solubility by point mutations, it is crucial to have a solubility prediction method which operates in a sequence-length-independent fashion. If the classifier is mainly or substantially driven by protein length it could be less reliable when confronted with such a problem then would be expected based on the evaluation on a length-biased data set. Therefore, we believe that it is beneficial to normalize the length distribution even though it leads to a more difficult classification problem. To check for easily detectable biases in our data we evaluated the classification performance of five simple global protein features (pI, length, FI, AI and GRAVY) using the same naive Bayes classification method [20] and found that the combination of all five features yielded an accuracy of 53.1% and an AUROC of 0.548. We were thus unable to find obvious biases in our data.

We evaluated the performance of the CCSOL method on our holdout set. Since CCSOL only provides numerical scores and does not categorize instances into soluble/insoluble class we assign the soluble label to each sequence with a CCSOL score of 70 or greater and insoluble to those with lower scores. This particular threshold was chosen to minimize the difference between the sensitivity and specificity values. All performance measures for threshold values ranging from 40 to 100 are provided in Table S1. Both the accuracy (55.1%) and MCC (0.0659) of CCSOL were lower then those of PROSO II (accuracy 75.4%, MCC 0.39) and SOLpro (accuracy 65%, MCC 0.068). However, we note that this comparison is not fully justifiable as CCSOL was trained only on prokaryotic sequences while our data set contains both prokaryotic and eukaryotic proteins. We next compared PROSO II with CCSOL based on the data set used to construct the latter method (initially created by Niwa *et al*. [18]). Following the approach of Niwa *et al*. we derived the sets of ‘aggregation prone’ and ‘highly soluble’ proteins, defined as one-third of all proteins having the highest and the lowest experimental solubility, respectively. The CCSOL threshold for the soluble class was set to 50. The CCSOL method scored higher in terms of accuracy (76.1% versus 63.9%), MCC (0.519 versus 0.26) and gain (soluble) (1.623 versus 1.42) than PROSO II (Table S3). Although on this specific data

Table 2. Basic performance measurements of the PROSO II method for four levels of sequence clustering at 25%, 50%, 75% and 90% identity. We evaluated the PROSO II method at four different levels of sequence identity. All results presented were obtained using stratified 10-fold cross-validation as described in the text. The letters P and N in parentheses refer to positive (soluble) and negative (insoluble) class, respectively.

Level of sequence identity clustering (%)	Accuracy	MCC	AUROC	Gain (P)	Gain (N)
90	71.0	0.422	0.785	1.386	1.460
75	70.5	0.414	0.777	1.354	1.484
50	70.0	0.405	0.77	1.345	1.475
25	69.9	0.401	0.766	1.356	1.450

set CCSOL performed better than PROSO II our method still achieved quite good performance which can be seen on comparing MCC and gain values. The crucial difference between these two tests – CCSOL on the PROSO II data and PROSO II on the CCSOL data – is that the PROSO II data set is a holdout data set never used for training PROSO II while the complete CCSOL data set was used to train the CCSOL method. This naturally gives the CCSOL method advantage when tested on its own training data set; the performance of PROSO II on the CCSOL data set was somewhat weaker, although still quite good.

Because of ongoing extensive genome sequencing efforts the protein sequence space is expanding rapidly. It is crucial to include the highest possible amount of data into protein classification models. By virtue of relying on a five times larger sequence data set PROSO II has better coverage of the protein universe and is therefore expected to be more reliable and robust than other methods when confronted with a newly discovered sequence.

Interestingly, we found that the performance of our method was only slightly dependent on the clustering level of the training/evaluation data. For example, the accuracy for the data clustered at 25% and 90% identity was 69.99% and 71%, respectively, with an MCC of 0.401 and 0.422 (Table 2). It seems that our classification approach is well adapted to both stringent and relaxed levels of sequence clustering. Because the 90% clustered data set had the highest number of instances and supposedly provided the biggest coverage of sequence space we chose it when building the PROSO II web server.

Conclusions

We provide a public web server implementing a novel sequence composition and similarity-based model to classify proteins into 'soluble' and 'insoluble' classes. Our approach allows us to categorize proteins with low or no sequence similarity to the training data. PROSO II outperforms (accuracy of 75.4% and gain on soluble class of 2.263) previously reported methods as demonstrated using the holdout set with a real-life-like class distribution. Based on the fact that it relies on the largest experimental data set we expect it to be particularly robust in selecting soluble proteins as well as in filtering out recalcitrant targets. Moreover, in our work we identified the subset of sequence features having the strongest impact on protein solubility.

An apparent limitation of PROSO II is that it is only applicable to non-membrane proteins of size between 20 and 2004 residues. It is also unable to take into account factors unrelated to protein sequence such as buffer composition, temperature or presence of nucleic acids. In the future it would be interesting to integrate our prediction algorithm with the methods evaluating the impact of point mutations on protein stability [21–23].

The PROSO II server is available with no registration requirements under http://mips.helmholtz-muenchen.de/prosoII.

Methods

Data

The pepcDB database [19] (http://pepcdb.sbkb.org/) stores target and protocol information contributed by Protein Structure Initiative centers as well as targets imported from the TargetDB database. Each protein target can be associated with multiple amino acid sequences corresponding to different constructs. For example, full-length protein, N- or C-terminal truncated proteins and single domains are amongst available constructs. For each construct experimental results and status history are recorded. The status description includes the following principal stages: Selected, Cloned, Expressed, Soluble, Purified, Crystallized, HSQC (heteronuclear single quantum coherence), Structure, and In PDB. All constructs that achieved the Soluble status or subsequent stages (including native_diffraction-data, NMR_assigned, phasing_diffraction-data, diffraction, in_BMRB, NMR_structure, crystal_structure, diffraction-quality_crystals, in_PDB, crystallized, HSQC) may be considered soluble. Comparing the experimental status at two time points, September 2009 and May 2010, we were able to derive a set of insoluble proteins defined as those which were not soluble in September 2009 and still remained in that state 8 months later. We did not consider 'test' targets as some of them were entered in the database just to test the IT infrastructure. To remove targets dropped as a result of competitors having a structure submitted to the PDB [14] all proteins with 100% identity to any PDB sequences were removed from the insoluble set. Although many proteins from pepcDB lack a description of an expression system it was shown before [12] that more than 75% of those proteins that reach the 'In PDB' status are indeed produced by heterologous expression in *E. coli*. An additional set of heterologously expressed soluble proteins was derived from the PDB database (release July 2010). We selected all proteins with the annotation 'Expression Organism: ESCHERICHIA COLI'. The majority of proteins from pepcDB are expressed as single proteins. Around 5% of the final soluble data set originate from PDB entries of heterologous complexes. There is no obvious way to find out whether they were co-expressed or expressed as single proteins and then mixed. Therefore we decided to keep them in our analysis.

Each of the described data sets was refined by removing proteins with one or more transmembrane segments as predicted by TMHMM 2.0c [24] and those sequences containing more than one contiguous 'X' character. The number of instances and other details on each data set used in this work can be found in Table S2. Finally we reduced sequence redundancy of the soluble and insoluble sets separately by homology clustering at the 90%, 75%, 50%, 25% sequence identity level using the CD-HIT program [25]. We abstained from doing any cross-class clustering because it artificially simplifies classification leading to overoptimistic estimates of performance.

To prevent sequence length and class distribution from becoming prevalent features in our model we adjusted class and sequence length distribution of soluble and insoluble data sets such that length alone was non-decisive in classification and both classes were equally represented. To this end insoluble and soluble sequences were sorted and divided into eight bins (equal ranges) according to protein length. For each bin, the number of sequences from the least populated class was used as a limit for the number of sequences to be considered in each of the two classes. Both soluble and insoluble sequences fall into the length range of 20–2004 residues (proteins outside this range were very few and therefore rejected). This preprocessing step helps assure similar length distributions between insoluble and soluble classes and thus removes any significant length-related bias.

Length-adjusted soluble and insoluble data sets clustered at four different sequence identity levels were used for model building and evaluation by 10-fold cross-validation.

A separate data set was built to model the real-life class distribution with a ratio of 1 to 5 between soluble and insoluble proteins, as observed in the pepcDB database releases between May and December 2010. To do this we used insoluble sequences removed during the adjustment of length distribution described above from the data set clustered at 25% maximal allowed identity. We made sure that no sequence was identical to either soluble or insoluble sequences used in the 10-fold cross-validation. Moreover, all proteins identical to soluble sequences from pepcDB (December 2010) or PDB (December 2010) were removed. The soluble data set consisted of new PDB entries from October and November 2010. All new PDB sequences had been heterologously expressed in *E. coli,* and their sequences were filtered against sequences used in the cross-validation and transmembrane segments (TMHMM 2.0c). After an additional 50% sequence identity clustering using CD-HIT, half of the proteins from both soluble and insoluble data sets were saved to form a holdout set and the rest were used for threshold selection. The holdout sets of soluble and insoluble proteins were derived to allow for additional evaluation of our classification model.

In summary, in the process of careful and restrictive data selection from the pepcDB and PDB databases we built the currently largest available (more then 82 000 proteins) input data set used for model building and evaluation. Furthermore, we constructed a holdout set with the natural class distribution as observed in pepcDB and used it for an independent model validation.

Features

Amino acid sequences were represented by the frequencies of monopeptides, dipeptides and tripeptides. Although the final classifier uses only frequencies of monopeptides and dipeptides we also tested a compressed alphabet representation described in our previous publications [12,26]. Additionally, we calculated the following global sequence features: length, pI, GRAVY [27,28], AI [29] and FI [30]. A separate naive Bayes classifier was trained and evaluated with these features to check whether any of them could result in a reasonably good classification performance. Additionally, the set of AI, FI, GRAVY and pI was also used.

Classification

We classified data using the two-level framework described in detail in our previous publications [12,26] but this time we used the threshold selector classifier [31] (optimized for accuracy by an internal 10-fold cross-validation using only the training data) with a multinomial logistic regression model [32] on both levels. Briefly, the input data were first classified using k-mer-based and Parzen window classifiers. A second-level classifier aggregates results of primary classifiers. Ten-fold stratified cross-validation over input data was performed over both levels to obtain class assignment for each protein and to estimate the accuracy of the entire method.

We built a sequence-similarity-based model using an adapted Parzen window approach [33]. It relies on a BLASTP [34] score to calculate a local approximation of the probability function [33] using the modified Cauchy kernel. For

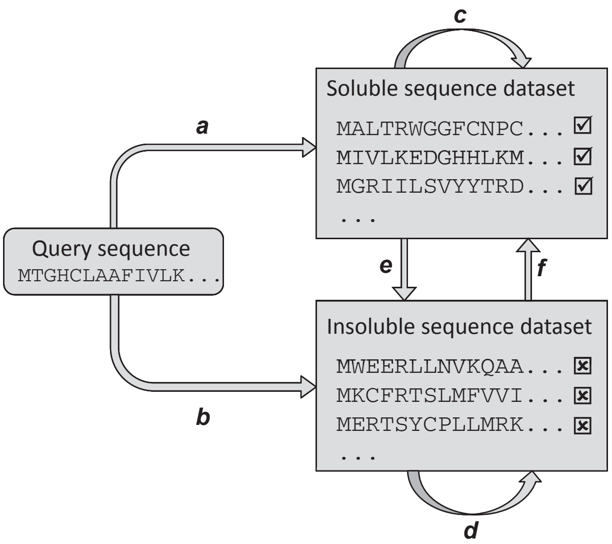

Fig. 1. Flowchart of the Parzen window method. Each arrow represents the calculated BLAST score when comparing *a*, the query sequence against the soluble data set; *b*, the query sequence against the insoluble data set; *c*, a soluble instance against the soluble data set; *d*, an insoluble instance against the insoluble data set; *e*, a soluble instance against the insoluble data set; and *f*, an insoluble instance against the soluble data set.

each query/test protein, BLASTP scores to the soluble and insoluble data sets (training data set in the case of 10-times cross-validation) were calculated. Moreover, for each member of the soluble and insoluble data sets themselves, the same score was calculated against both data sets (Fig. 1). Each query/test protein is then characterized by two similarity values:

similarity to soluble data set

$$\mathrm{SS} = \frac{1}{hn}\sum_{i=0}^{n}\frac{1}{\pi}\left[1+\left(\frac{a-c_i}{h}\right)^2+\left(\frac{b-e_i}{h}\right)^2\right]$$

similarity to insoluble data set

$$\mathrm{SI} = \frac{1}{hm}\sum_{j=0}^{m}\frac{1}{\pi}\left[1+\left(\frac{b-d_j}{h}\right)^2+\left(\frac{a-f_j}{h}\right)^2\right]$$

where *i*, *j* are elements and *m*, *n* are sizes (equal in our case) of the soluble/insoluble training sets, respectively; SS and SI reflect the similarity of the query sequence to the soluble and insoluble data sets, respectively; *a*, *b*, *c*, *d*, *e* and *f* are BLASTP scores (Fig. 1) when comparing *a*, the query sequence against the soluble data set; *b*, the query sequence against the insoluble data set; *c*, a soluble instance against the soluble data set; *d*, an insoluble instance against the insoluble data set; *e*, a soluble instance against the insoluble data set; and *f*, an insoluble instance against the soluble data set; *h* is the bandwidth or smoothness parameter which was set to 0.65.

Solubility was then computed using the following equation:

$$f(t) = \frac{\mathrm{SS}}{\mathrm{SS}+\mathrm{SI}}$$

When the value of the function was higher than the threshold (0.5) then the instance was marked as soluble by the Parzen window classifier.

Feature selection

For feature selection we used the wrapper method [15] with the threshold selector model configured as described above for the first level of classification as a classification procedure and the 'best first' approach [15] as a search algorithm. The detailed procedure can be found in Smialowski *et al.* [12].

Classification evaluation

In order to quantify the performance of the classifiers the following measures were calculated.

- AUROC: The receiver operating characteristic curve portrays the relation between the true positive rate and the false positive rate of the classifier. AUROC measures the discriminating ability of the model and it takes values between 0.5 for random drawing and 1.0 for perfect classifier. It is often interpreted as a probability that if you randomly draw one positive and one negative instance the one scored higher by the model will be actual positive. It was calculated using the algorithm implemented in the WEKA package [31].
- Accuracy: the number of correctly classified instances divided by the total number of instances.
- True positive rate (TP rate) (also called sensitivity or recall of the positive class): the number of correctly classified instances from the positive class divided by the number of all instances from the positive class (TP + FN).
- True negative rate (TN rate) (also called sensitivity or recall of the negative class): the number of correctly classified instances from the negative class divided by the number of all instances from the negative class (TN + FP).
- Specificity: the ratio of the number of correctly classified negative (TN) instances to the sum of all negative instances (TN + FP).
- Precision (selectivity): the ratio of the number of correctly classified positive (TP) or negative (TN) instances to the number of all instances classified as positive (TP + FP) or negative (TN + FN), for positive and negative class respectively.
- Gain: a ratio of the given class precision (selectivity) to the proportion of the given class in the full data set.

- MCC, calculated as

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

where TP, FP and FN are the numbers of true positive, false positive and false negative instances, respectively.

In particular, gain is an important performance measure that quantifies how much better the decision is when guided by the classifier in comparison with random drawing of instances. MCC indicates the correlation between the classifier assignments and the actual class in the two-class case. It is a good measure of classifier performance even when classes are unbalanced.

Web server

The PROSO II web server was built using the JBOSS SEAM framework (2.0) (Red Hat, Raleigh, NC, USA) based on AJAX technology. The server side application is run on the JBOSS server (4.2) (Red Hat).

Acknowledgements

We would like to thank Philip Wong for careful reading of the manuscript and helpful advice. Pawel Smialowski acknowledges financial support of the Deutsche Forschungsgemeinschaft.

References

1 Chow MK, Amin AA, Fulton KF, Fernando T, Kamau L, Batty C, Louca M, Ho S, Whisstock JC, Bottomley SP *et al.* (2006) The REFOLD database: a tool for the optimization of protein expression and refolding. *Nucleic Acids Res* **34**, D207–D212.

2 Singh SM & Panda AK (2005) Solubilization and refolding of bacterial inclusion body proteins. *J Biosci Bioeng* **99**, 303–310.

3 Georgiou G & Valax P (1996) Expression of correctly folded proteins in *Escherichia coli*. *Curr Opin Biotechnol* **7**, 190–197.

4 Davis GD, Elisee C, Newham DM & Harrison RG (1999) New fusion protein systems designed to give soluble expression in *Escherichia coli*. *Biotechnol Bioeng* **65**, 382–388.

5 Waldo GS (2003) Genetic screens and directed evolution for protein solubility. *Curr Opin Chem Biol* **7**, 33–38.

6 Wilkinson DL & Harrison RG (1991) Predicting the solubility of recombinant proteins in *Escherichia coli*. *Biotechnology (NY)* **9**, 443–448.

7 Christendat D, Yee A, Dharamsi A, Kluger Y, Savchenko A, Cort JR, Booth V, Mackereth CD, Saridakis V, Ekiel I *et al.* (2000) Structural proteomics of an archaeon. *Nat Struct Biol* **7**, 903–909.

8 Bertone P, Kluger Y, Lan N, Zheng D, Christendat D, Yee A, Edwards AM, Arrowsmith CH, Montelione GT & Gerstein M (2001) SPINE: an integrated tracking database and data mining approach for identifying feasible targets in high-throughput structural proteomics. *Nucleic Acids Res* **29**, 2884–2898.

9 Luan CH, Qiu S, Finley JB, Carson M, Gray RJ, Huang W, Johnson D, Tsao J, Reboul J, Vaglio P *et al.* (2004) High-throughput expression of *C. elegans* proteins. *Genome Res* **14**, 2102–2110.

10 Goh C-S, Lan N, Douglas SM, Wu B, Echols N, Smith A, Milburn D, Montelione GT, Zhao H, Gerstein M *et al.* (2004) Mining the structural genomics pipeline: identification of protein properties that affect high-throughput experimental analysis. *J Mol Biol* **336**, 115–130.

11 Idicula-Thomas S, Kulkarni AJ, Kulkarni BD, Jayaraman VK & Balaji PV (2006) A support vector machine-based method for predicting the propensity of a protein to be soluble or to form inclusion body on overexpression in *Escherichia coli*. *Bioinformatics* **22**, 278–284.

12 Smialowski P, Martin-Galiano AJ, Mikolajka A, Girschick T, Holak TA & Frishman D (2007) Protein solubility: sequence based prediction and experimental verification. *Bioinformatics* **23**, 2536–2542.

13 Chen L, Oughtred R, Berman HM & Westbrook J (2004) TargetDB: a target registration database for structural genomics projects. *Bioinformatics* **20**, 2860–2862.

14 Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, Shindyalov IN & Bourne PE (2000) The Protein Data Bank. *Nucleic Acids Res* **28**, 235–242.

15 Kohavi R & John G (1997) Wrappers for feature subset selection. *Artifi Intell J* **97**, 273–324.

16 Magnan CN, Randall A & Baldi P (2009) SOLpro: accurate sequence-based prediction of protein solubility. *Bioinformatics* **25**, 2200–2207.

17 Agostini F, Vendruscolo M & Tartaglia GG (2011) Sequence-based prediction of protein solubility. *J Mol Biol* doi: 10.1016/j.jmb.2011.12.005.

18 Niwa T, Ying B-W, Saito K, Jin W, Takada S, Ueda T & Taguchi H (2009) Bimodal protein solubility distribution revealed by an aggregation analysis of the entire ensemble of *Escherichia coli* proteins. *Proc Natl Acad Sci USA* **106**, 4201–4206.

19 Berman HM, Westbrook JD, Gabanyi MJ, Tao W, Shah R, Kouranov A, Schwede T, Arnold K, Kiefer F, Bordoli L *et al.* (2009) The protein structure initiative structural genomics knowledgebase. *Nucleic Acids Res* **37**, D365–D368.

20 Domingos P & Pazzani M (1996) Beyond independence: conditions for the optimality of the simple Bayesian classifier. In *Proceedings of the 13th International Conference on Machine Learning (Vol. 29)* (Saitta L ed.), pp. 105–112. Morgan Kaufmann, San Francisco, CA (retrieved from http://www.cs.washington.edu/homes/pedrod/papers/mlc96.pdf).

21 Capriotti E, Fariselli P, Rossi I & Casadio R (2008) A three-state prediction of single point mutations on protein stability changes. *BMC Bioinformatics* **9**(Suppl 2), S6.

22 Cheng J, Randall A & Baldi P (2006) Prediction of protein stability changes for single-site mutations using support vector machines. *Proteins* **62**, 1125–1132.

23 Parthiban V, Gromiha MM & Schomburg D (2006) CUPSAT: prediction of protein stability upon point mutations. *Nucleic Acids Res* **34**, W239–W242.

24 Krogh A, Larsson B, von Heijne G & Sonnhammer EL (2001) Predicting transmembrane protein topology with a hidden Markov model: application to complete genomes. *J Mol Biol* **305**, 567–580.

25 Huang Y, Niu B, Gao Y, Fu L & Li W (2010) CD-HIT Suite: a web server for clustering and comparing biological sequences. *Bioinformatics* **26**, 680–682.

26 Smialowski P, Schmidt T, Cox J, Kirschner A & Frishman D (2006) Will my protein crystallize? A sequence-based predictor. *Proteins* **62**, 343–355.

27 Gasteiger E, Gattiker A, Hoogland C, Ivanyi I, Appel RD & Bairoch A (2003) ExPASy: the proteomics server for in-depth protein knowledge and analysis. *Nucleic Acids Res* **31**, 3784–3788.

28 Kyte J & Doolittle RF (1982) A simple method for displaying the hydropathic character of a protein. *J Mol Biol* **157**, 105–132.

29 Ikai A (1980) Thermostability and aliphatic index of globular proteins. *J Biochem* **88**, 1895–1898.

30 Prilusky J, Felder CE, Zeev-Ben-Mordehai T, Rydberg EH, Man O, Beckmann JS, Silman I & Sussman JL (2005) FoldIndex: a simple tool to predict whether a given protein sequence is intrinsically unfolded. *Bioinformatics* **21**, 3435–3438.

31 Frank E, Hall M, Trigg L, Holmes G & Witten IH (2004) Data mining in bioinformatics using Weka. *Bioinformatics* **20**, 2479–2481.

32 Le Cessie S & Van Houwelingen JC (1992) Ridge estimators in logistic regression. *Appl Stat* **41**, 191–201.

33 Parzen E (1962) On estimation of a probability density function and mode. *Ann Math Stat* **33**, 1065–1076.

34 Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z, Miller W & Lipman DJ (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res* **25**, 3389–3402.

Supporting information

The following supplementary material is available:

Table S1. Performance of the CCSOL method as tested on 1743 instances of the holdout set (21 instances were rejected by the method based on their length or presence of low complexity regions).

Table S2. Additional information about the data set used for construction and evaluation of PROSO II.

Table S3. Performance of the CCSOL and PROSO II methods as tested on 1230 'aggregation prone' and 1013 'highly soluble' sequences derived as described by Niwa *et al.*

This supplementary material can be found in the online version of this article.

Please note: As a service to our authors and readers, this journal provides supporting information supplied by the authors. Such materials are peer-reviewed and may be reorganized for online delivery, but are not copy-edited or typeset. Technical support issues arising from supporting information (other than missing files) should be addressed to the authors.