J Mol Model (2013) 19:4059–4070
DOI 10.1007/s00894-013-1886-9

SOFTWARE REPORT

PPIcons: identification of protein-protein interaction sites in selected organisms

Brijesh K. Sriwastava · Subhadip Basu · Ujjwal Maulik · Dariusz Plewczynski

Received: 28 February 2013 / Accepted: 6 May 2013 / Published online: 2 June 2013

Abstract The physico-chemical properties of interaction interfaces have a crucial role in characterization of protein–protein interactions (PPI). *In silico* prediction of participating amino acids helps to identify interface residues for further experimental verification using mutational analysis, or inhibition studies by screening library of ligands against given protein. Given the unbound structure of a protein and the fact that it forms a complex with another known protein, the objective of this work is to identify the residues that are involved in the interaction. We attempt to predict interaction sites in protein complexes using local composition of amino acids together with their physico-chemical characteristics. The local sequence segments (LSS) are dissected from the protein sequences using a sliding window of 21 amino acids. The list of LSSs is passed to the support vector machine (SVM) predictor, which identifies interacting residue pairs considering their inter-atom distances. We have analyzed three different model organisms of *Escherichia coli*, *Saccharomyces Cerevisiae* and *Homo sapiens*, where the numbers of considered hetero-complexes are equal to 40, 123 and 33 respectively. Moreover, the unified multi-organism PPI meta-predictor is also developed under the current work by combining the training databases of above organisms. The PPIcons interface residues prediction method is measured by the area under ROC curve (AUC) equal to 0.82, 0.75, 0.72 and 0.76 for the aforementioned organisms and the meta-predictor respectively.

Keywords Amino acids · AAindex · Machine learning · Pattern analysis · Proteins · Protein-protein complexes · Protein-protein interactions · Sequence · Structure · Support vector machine

Electronic supplementary material The online version of this article (doi:10.1007/s00894-013-1886-9) contains supplementary material, which is available to authorized users.

B. K. Sriwastava
Department of Computer Science and Engineering,
Government College of Engineering and Leather Technology,
Kolkata 700098, India
e-mail: sriwastavabrijesh@yahoo.co.in

S. Basu · U. Maulik (✉)
Department of Computer Science and Engineering,
Jadavpur University, Kolkata 700032, India
e-mail: umaulik@cse.jdvu.ac.in

S. Basu
e-mail: subhadip@cse.jdvu.ac.in

D. Plewczynski (✉)
Interdisciplinary Centre for Mathematical and Computational Modelling, University of Warsaw, 02-106 Warsaw, Poland
e-mail: darman@icm.edu.pl

D. Plewczynski
Department of Physical Chemistry, Faculty of Pharmacy,
Medical University of Warsaw, 02-097 Warsaw, Poland

Introduction

Proteins perform a biological function by interacting with other proteins, small chemical compounds, metabolites, and RNA/DNA molecules. Interactions are completed by the formation of complexes, either transient or more permanent, e.g., the replisome, RNA polymerases, spliceosome, ribosome, haperonins and various complexes formed along signal transduction pathways, or during enzyme catalysis and inhibition. Such interaction networks mediate biological processes. Protein-protein interactions (PPI) are at the core of the entire interaction system of any living cell, making them the central hubs or major mediators for virtually every bio-chemical process. Therefore for a given protein, in order to understand its biological function, it is important to identify its likely

interactions with other proteins. Detailed information of protein-protein interactions, metabolic and signal transduction networks improves our understanding of diseases, perturbation of healthy states or processes, providing the theoretical basis for new therapeutic approaches, mutant engineering and design, high throughput screening for drug design [1].

Large molecular machines carry out most of molecular processes in the cell, like DNA replication. The topological organization and connectivity of the components within a protein complex are given by structural alignment of their interfaces. Understanding the characteristics of interfacial sites is a requirement for modeling the molecular recognition process. It is observed that the recognition sites have very similar common properties. Interaction sites share specific chemical and physical characteristics, which contribute to a molecular recognition process, for example hydrophobic, planar, globular and protruding properties.

Currently developed high-throughput experimental methods, such as Yeast two-hybrid, or mass spectrometry provided the global view of the whole interaction network for model organisms (interactome) [2–6]. The growing number of observed PPI interactions, makes it increasingly important to distinguish true physical interactions from experimental methods' artifacts or purely functional complexes. The mapping between identified protein-protein interactions and its atomic level structural details is essential for understanding the PPI molecular functions, and for designing drugs that can inhibit formation of a complex. Typically, X-ray crystallography, or NMR techniques are used for assigning the three-dimensional structure for given protein-protein complex, allowing for detailed structural analysis in the context of molecular organization and its dynamics [7].

Predicting residues that participate in protein–protein interactions helps to suggest, which amino acids are located at the interface, and further experimentally verifying using mutational analysis. Moreover, it can be used in the virtual screening of ligands to find potent inhibitors that are able to alter the protein-protein interaction for therapeutics discovery. First, it is important to find which properties of protein–protein interfaces differentiate them from non-interface surface regions. Analysis of physical and chemical properties required selecting distinctive features as observed in known three dimensional structures of complexes. Those features can be used for building statistical models and further *in silico* prediction using the variety of machine learning algorithms. Two major types of complexes are observed, namely homodimers and heterodimers, where homodimers mostly form permanent and highly optimized complexes, generally by aligning hydrophobic interfaces. In contrast, in the case of hetero-complexes, hydrophobicity is indistinguishable from the rest of the surface [8–10]. Jones and Thornton [11] suggested importance of differentiating between aforementioned types of complexes, when analyzing their intermolecular interfaces.

Typically, PPI recognition methods use either protein-protein docking studies for structural fitting of complexes' members [12–14], or exploit structural and physico-chemical characterization of the interface. Structural properties of interior surface and interfaces residues of oligometric proteins were compared by [15–18] and they found that hydrophobicity, accessible surface area (ASA), shape and residues' preferences are the most important factors. Two protein subunits may interact and form a protein-protein interface by aligning two relatively flat surfaces, or can form non-planar curved interface. Therefore we need to describe a curvature of interface that is how far the interface residues are deviated from a plane. The planarity of the surfaces can be calculated by root mean square deviation of all interface atoms from the corresponding least squares plane as calculated using positions of those atoms. It has been observed that interfaces of hetero-complexes are more planar than homodimers, and for them it is difficult to find single parameter sufficient to distinguish interface residues from other surface patches. However, further studies suggest [11] that accessible surface area has the high impact on the differentiation. Protein-protein interfaces can be identified by the change in their solvent ASA, when going from monomeric to the dimeric state. Interface residues are defined as those, where ASA is decreased by 1 Å. Jones and Thornton observed that protein-protein interfaces for permanent complexes are more closely packed, but less planar with fewer inter sub-unit hydrogen bonds than the nonobligatory complexes [11].

Fariselli et al. [19] defined surface residues if their ASA is larger than 16 % of its nominal maximum area [20]. The DSSP program [21] helps to calculate ASA values for each residue in unbound chain. Liu et al. [22] recognized a surface residue as an interface one, if the distance between its Cα atom and any residue's Cα atom from its molecular partner is less than 1.2 nm. Transient protein-protein interactions have an important role in many biological processes, such as cell regulation and signal transduction. In their study, the temperature factor (B-factor) was observed to differentiate between an interface and the rest of the protein surface. Therefore, apart from two well-known features, such as sequence profiles and ASA, the temperature factor is also important. In our work, we show that incorporation of a great variety of different physico-chemical properties, together with other structural attributes, allows for further improving the quality of characterization of protein-protein interactions.

Several web servers have been recently developed for protein-protein interaction sites prediction, using different computational methodology and providing different levels of accuracy:

a) Cons-PPISP http://pipe.scs.fsu.edu/ppisp.html method uses PSI-Blast sequence profile and solvent accessibility as the input to the artificial neural network [23];

b) Promate http://bioportal.weizmann.ac.il/promate is based on Bayesian representation of secondary structure, atoms distribution, amino-acid pairing, and sequence conservation [24];
c) PINUP http://sparks.informatics.iupui.edu/PINUP/ uses an empirical scoring function; consisting of the side-chain energy term, the term proportional to solvent accessible area, and the term accounting for sequence conservation; to predict protein binding site [25];
d) PPI-Pred http://bmbpcu36.leeds.ac.uk/ppi_pred/ takes six properties (including surface shape and its electrostatic potential) as input to the support vector machine approach [26];
e) SPPIDER http://sppider.cchmc.org/ is based on artificial neural network method, which uses predicted solvent accessibility [27];
f) Meta-PPISP http://pipe.scs.fsu.edu/meta-pisp.html provides the meta web server that is built on top of the raw scores from cons-PPISP, Promate and PINUP [28].

Zhou and Shan [29] predicted protein–protein interaction sites using artificial neural network (ANN) classifier trained with sequence profiles of neighboring residues and solvent exposure as the input. The main strength of the ANN predictor lies in the fact that neighbors' lists and solvent exposure are relatively insensitive to structural changes upon a complex formation, therefore performing equally well for bound or unbound structures of interacting partners.

In one of the recent works, Jang et al. [30] proposed a domain-based PPI prediction model using intra-protein domain cohesion and intra-protein domain combination coupling interactions. The technique uses hybrid inter/intra-domain interaction information for improvement of the prediction accuracy. The work by Guo et al. [31] uses SVM and auto covariance which accounts for the interactions between amino acids within 30 amino acids apart in the sequence. This method considers effect of neighboring amino acids, similar to the sliding window scheme used in many earlier works [32–35]. However, they are not considering inter-residue interactions between two interacting proteins and therefore Guo et al. cannot specifically predict the interacting residue fragments in a pair of interacting protein. They have also curated the negative data samples, leading to over estimation of prediction accuracy [36].

Summarizing, significant research was done in the area of protein-protein interactions, yet the problem of interaction sites prediction is still not fully understood. Major unresolved issues are, among others, linked with the problem of selection of biological and physico-chemical features crucial for protein-protein interactions [37]. The main problem in terms of theoretical analysis and machine learning algorithms typically is linked with non-balanced training dataset, namely the number of interaction sites is typically very small in comparison to non-interacting sites [38]. Moreover, any single physico-chemical feature is not sufficient to distinguish interface and non-interface residues, the complex nonlinear combinations of features are needed to describe an interaction site.

The PPI prediction is not the balanced learning problem; therefore the optimal set of computational methods' parameters is not easy to obtain. To select the proper subset of descriptors, we applied the consensus fuzzy clustering technique [38] to extract high quality physico-chemical indices from the set of 544 indices provided by the AAindex1 database (http://www.genome.jp/aaindex/). The selected subset of the most informative features is proved to be very useful for local representation of protein sequence characteristics in various machine learning applications [39]. Deng et al. [39] proposed ensemble learning method in order to overcome the misbalancing problem in PPI and effectively utilize a wide variety of features. He combined bootstrap sampling technique, SVM-based fusion classifiers and weighted voting strategy.

Other works in this domain include extraction of PPIs from biomedical literature [40, 41]. The challenge here is to find a suitable compromise between the biological relevance of the results and a comprehensive coverage of the analyzed networks. Zhang et al. [34] have used the graph kernel to compute dependency graphs representing the sentence structure for PPI extraction task, which can efficiently make use of full graph structural information, and particularly capture the contiguous topological and label information, ignored before. PPI networks can be grouped in two categories, one allowing a protein to participate in different clusters and the other generating only non overlapping clusters. Pizzuti et al. [35] present a co-clustering based technique to generate both overlapping and non overlapping clusters from the input PPI networks.

In view of the above facts, the goal of our paper is to predict the interacting residues for a pair of proteins given their unbound structures. The interface residues define the interaction site for those two proteins. More specifically, we attempt to predict interaction sites in protein complexes more accurately using selective high quality index physico-chemical features (HQI) extracted from AAindex1 dataset. We have used the sliding window algorithm with the length of 21 amino acids to select sets of local sequence segments for each protein, then identifying interacting residue pairs by considering their inter-atom distances. We have trained our method on three datasets of interacting proteins for *Escherichia coli*, *Saccharomyces Cerevisiae* and *Homo sapiens* and evaluated the PPI sites prediction performance on unknown test samples using SVM classifier. The PPIcons software is available for public domain at http://code.google.com/p/cmater-bioinfo/ under Apache License 2.0. The meta-predictor is also designed by combining the interacting proteins from all considered organisms. PPIcons

therefore is able to perform identification of interaction sites: 1) using organism specific prediction by the classifiers designed separately for three aforementioned organisms, and 2) using unified organism-independent meta-prediction. The dataset design principles, selected HQI features, and classifier design methodology are described in detail in the following section. The Results section provides the performance evaluation metrics and analysis of the prediction results for PPIcons software.

Methods

Training dataset

There are several databases available online containing proteins pairs that are experimentally observed to interact. We can divide these resources roughly into two groups: providing either sequence or structural details. The first group of experimentally confirmed protein-protein pairings also involves transient interactions, and the second focuses on real protein-protein complexes, i.e., stable, permanent interactions. In almost all databases, the developers use their own format for the data storage and processing, making the integration across different datasets difficult. However, the theoretical analysis of interactions depends on heterogeneous sources of biological information, such as sequence (genomic) and structural (crystallographic) databases, the literature mining, and experimental data. For our analysis, we selected two major databases containing experimental information about protein-protein interactions, namely Protein Data Bank (PDB) [42], where one can find the three-dimensional structures of protein complexes, and Database of Interacting Proteins (DIP, http://dip.doe-mbi.ucla.edu/dip) [43], where the known interactions among protein pairs are stored.

Initially we started with 12606 number of protein-protein interactions of *E. coli* organism, which are given in the file Ecoli20100614.txt of DIP database. Among these interactions, some entries did not have UniProt KB signatures, matching PDB code, or even their primary sequences. Therefore, we applied the multi-stage refinement. After removing the interactions with incomplete information (unavailability of primary sequence and some missing UniProtKb id), the DIP database for *E. coli* is reduced to 8740 interactions (step 2). In step 3, we have checked the PDB entry for these known interactions by mapping the PDB id from their UniProtKb id. This process further reduces the PPI data to 2256 interaction pairs. Then the interactions are verified for availability of the same PDB entry for both interacting proteins, therefore known bound structures, and in this step the database size gets reduced to 312 entries (step 4). Each entry is now comprised of a valid PDB database identifier (for the protein-protein complex), with multiple UniProtKb codes. Further, the entries for homodimers are also removed (step5) and we finally get only 40 valid hetero-interactions as our training dataset. The amino acid sequences are extracted from file dip20091230.seq file (http://dip.doe-mbi.ucla.edu/dip/) using the corresponding UniProtKb id. A schematic description of the database preparation phase is shown in Fig. 1.

In the case of PPI interactions for *Yeast*, we started with 22,208 entries from the Scere20100614.txt file of DIP database. After processing them through step 1 and step 2, as discussed above, the database first remains the same number 22,208 and after applying step 3 it was reduced to

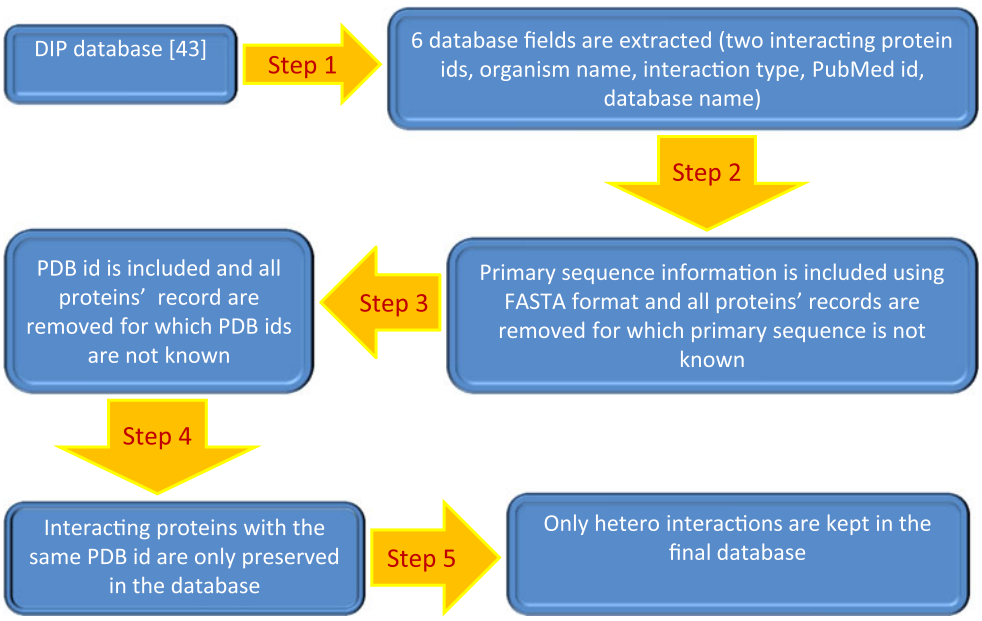

Fig. 1 A schematic diagram shows the training data preparation steps for PPI organism-specific database

1372, which further ended up as 204 entries, following step 4. After removing the homo-complexes protein (same protein) interactions (step 5), we finally get only 123 hetero-complex (different proteins) interactions in our *Yeast* training dataset.

Similarly for *Homo sapiens* we started with 2251 entries from the Hsapi20100614.txt file of DIP database. After processing them through step 1 and step 2, as discussed above, the database first remains the same number 2251 and after applying step 3 it reduced to 1007, which further reduces to 168 entries, following step 4. After removing the homo protein interactions (step 5), we finally get only 33 hetero interactions in our *Homo sapiens* training dataset.

The database format, used for our work is shown below, along with three valid interactions for the three organisms. The statistics recognized of PPI networks of *E. coli*, *Yeast* and *Homo sapiens* are also shown in the Figures 1, 2 and 3 respectively (see Supplementary material). The complete databases are available freely to download for academic users from our website http://code.google.com/p/cmater-bioinfo/.

Choice of the amino acid feature set

In conjunction with earlier machine and statistical learning approaches, Saha et al. [38] have performed an extensive search to derive, optimize, and evaluate physico-chemical features that can best discriminate between interacting and non-interacting sites. These features can be roughly divided into eight groups, namely electric properties, hydrophobicity, alpha and turn propensities, physico-chemical properties, residue propensity, composition, beta propensity and intrinsic propensities. Currently, 544 amino acid indices are released in AAindex1 database. These features previously were clustered into different high-quality-indices (HQI) by co-authors [38] . In the current work, we have used eight HQIs (HQI8) with names: BLAM930101, BIOV880101, MAXF760101, TSAJ990101, NAKH920108, CEDJ970104, LIFS790101, and MIYS990104. Detailed description of the clustering method, software and Supplementary material are available for academic users at http://sysbio.icm.edu.pl/aaindex/AAindex/.

Representation of PPI features

Here, we are working with interacting protein pairs (say, P_A and P_B) from our aforementioned training datasets. Let P_A and P_B be described by their own amino acid sequences as $a_1, a_2, \ldots, a_M$ and $b_1, b_2, \ldots, b_N$ respectively, where

$$a_i, b_j \in \{A, R, N, D, L, K, M, F, C, Q, E, G, H, I, P, S, T, W, Y, V\},$$
$$\forall i = 1 \text{ to } M \text{ and } \forall j = 1 \text{ to } N.$$

In the next step, we compute inter-atom distances between P_A and P_B. Please note that we consider only the heavy atoms (as given in respective PDB entry) from each amino acid for this purpose. We define the distance measures as follows:

$$D_P(a_i, b_j) = \left(dr\left(a_{ik}, b_{jl}\right)\right), \quad \forall k = 1 \text{ to } P \text{ and } \forall l = 1 \text{ to } Q,$$

where P and Q are number of heavy atoms in the residues a_i and b_j respectively and $d_r(a_{ik},b_{jl})$=inter-atom Euclidean distances between the kth heavy atom of a_i and lth heavy atom of b_j. If $D_P(a_i,b_j)$ is lower than 3.5 Å [44], then corresponding residue pair (a_i, b_j) corresponding belonging to the protein pair (P_A, P_B) is said to be interacting, otherwise they are said to be non-interacting.

The protein sequences of hetero-complexes are therefore divided into multiple overlapping segments of subsequences, each consisting of 21 amino acids. Please note that the results from our current study strongly support selection of 21 window size, providing optimal results for protein-protein interaction prediction as tested on sample subsets of pairs of interacting proteins. For each pair of local sequence segments (LSS) from proteins P_A and P_B we consider all residues from $a_1, a_2, \ldots, a_{21}$ and $b_1, b_2, \ldots, b_{21}$ respectively, and check whether any of the residue pairs has $D_P(a_i,b_j)<3.5$ Å. If found, we annotate the given pair of sub-sequences (obtained from P_A and P_B respectively) as positive, i.e., confirmed interaction and extract HQI8 features for the 42 residues, resulting in a 428=336 dimensional feature vector representing positive training case. The overlapping subsequences are then shifted, as a sliding window, to check for further interactions. In all cases, where two sub-sequences have no interacting residue pair, then such sub-sequence pair is said to be non-interacting, and we recognized it as negative training cases described by 336 dimensional vector of features using also HQI8 features. These positive and negative vectors are then used by the machine learning procedure to train the support vector machine algorithm, designed separately to produce optimal recall, precision and area under ROC curve (AUC) scores.

Support vector machine

Support vector machine (SVM) is the pattern classification technique proposed by Vapnik and co-workers [45]. Traditional methods generally minimize the empirical training error, while SVM aims to minimizing the upper bound of the generalization error by maximizing the margin between the separating hyperplane and the data, providing the structure risk minimization principle protocol. Striking feature of SVM is the property of compacting information contained in the training data, and providing a sparse representation even using a small number of data points.

SVM performs both linear and nonlinear classification in the parameter space. Nonlinear classification is done by mapping the space S={x} of the input data into a high-dimensional feature space $F=\{\Phi(x)\}$ and this is achieved by choosing an appropriate mapping f so that the data points become almost linearly separable in the high-dimensional space. For this, there is no need to compute the mapped patterns $\Phi(x)$ explicitly, instead only the dot products between mapped patterns are calculated. This can be done easily by choosing different kernel function, which generates $\Phi(x)$, e.g., radial basis function (RBF), polynomial, sigmoid and multi-layer perceptron classifiers [46–48]. Typically the performance of SVM mostly depends on the appropriate kernel function, yet there is no regular way to choose appropriate kernel functions within a data-driven approach.

In the case of our prepared dataset of interacting and non-interacting fragments, one group contains vectors of features representing positives denoted by (+*ve*) and the second group include negatives (−*ve*). Therefore, using those two clusters the problem is finally becoming the binary classification, which can be handled by nonlinear support vector machine with polynomial kernel function. Input training samples are nonlinearly mapped into higher dimensional space, where they are separated using hyperplane, which is at maximum margin from each of the two clusters. Given the training set of n input points $\{x_i, y_i\}$; i=1, 2, 3, 4,……,n; x_i represents input feature vectors and y_i represents corresponding class label with two values {+1, −1}.

The separating hyperplane is represented as a linear combination of the training samples and classification of unknown test pattern x is done by the cost function: $f(x) = \sum_{i=1}^{1} \alpha_i y_i k(x_i, y_i) + b$, where $k(x_i,y_i)$ is SVM kernel function and b is the bias that can be optimized on given training data. Note that if $k(x_i, y_i)$ becomes small as it grows further away from x_i, each element in the sum measures the degree of closeness of the test point x to the corresponding point x_i. The sum of kernels above can be used to measure the relative nearness of each test point to the data points originating in one or the other of the sets to be discriminated. The optimal hyperplane is found by varying α_i and data point x_i. Finally, the sign of $f(x)$ function determines the class membership of the input query point. Here, we have used polynomial kernel function after testing different types of the kernels, observing that it provides the best results for our datasets.

Results

The current work, reported in this paper, involves 3254 positive interactions and 4948 negative interactions for *E. coli* proteome, 3490 positive interactions and 5082 negative interactions for *Yeast* proteome, and 3923 positive interactions and 6153 negative interactions for *Homo sapiens* proteome. We have also prepared a meta-dataset consisting of all three species mentioned above, which results in 9667 positive interactions, and 16,183 negative ones. It may be noted that the number of positive and negative interactions, considered in the training dataset for any proteome, are only a subset of all possible positive and negative interactions. This is done so to limit the computational complexity of the training algorithm, during the multi-fold cross validation (CV) process. Each interacting or non-interacting residue fragments are represented using HQI8 amino acids indices for both positive and negative data samples for the three organisms. We discuss here the training and testing prediction results for these organisms. Finally, the results with the meta-predictor that combines the training and test datasets from all three organisms are discussed in this section. In our

Table 1 Result on AUC optimized network over *E. coli* CV set and test set

Run	Accuracy	Recall	Precision	Specificity	AUC	MCC	F-measure
CV run#1	80.3571	0.737024	0.760714	0.84738	0.792202	0.587729	0.748682
CV run#2	81.3443	0.716263	0.793103	0.877273	0.796768	0.60559	0.752727
CV run#3	80.9328	0.709343	0.788462	0.875	0.792171	0.596719	0.746812
CV run#4	82.3288	0.741379	0.799257	0.877273	0.809326	0.627545	0.769231
CV run#5	79.6982	0.695502	0.770115	0.863636	0.779569	0.570494	0.730909
CV run#6	82.4417	0.754325	0.792727	0.870455	0.81239	0.630533	0.77305
CV run#7	83.0137	0.758621	0.80292	0.877273	0.817947	0.642617	0.780142
CV run#8	81.07	0.705882	0.793774	0.879545	0.792714	0.599392	0.747253
CV run#9	80.5213	0.743945	0.759717	0.845455	0.7947	0.591595	0.751748
CV run#10	78.4932	0.724138	0.731707	0.825	0.774569	0.550128	0.727903
CVAverage	**81.02011**	**0.728642**	**0.77925**	**0.863829**	**0.796236**	**0.600234**	**0.752846**
Test Set	**83.07692**	**0.736842**	**0.818462**	**0.892532**	**0.814687**	**0.642583**	**0.77551**

Bold entries represent average CV results and Test set results

Table 2 Result on AUC optimization over *Yeast* CV set and test set

Run	Accuracy	Recall	Precision	Specificity	AUC	MCC	F-measure
CV run#1	75.5585	0.683871	0.706667	0.804878	0.744375	0.49141	0.695082
CV run#2	74.4094	0.654839	0.697595	0.80531	0.730074	0.465255	0.675541
CV run#3	74.4094	0.687097	0.684887	0.783186	0.735141	0.470046	0.68599
CV run#4	74.574	0.684887	0.68932	0.787611	0.736249	0.472979	0.687097
CV run#5	75.3281	0.703226	0.694268	0.787611	0.745418	0.489872	0.698718
CV run#6	72.6675	0.680645	0.659375	0.758315	0.71948	0.436918	0.669841
CV run#7	75.3604	0.697749	0.697749	0.792035	0.744892	0.489785	0.697749
CV run#8	74.5407	0.7	0.68239	0.776549	0.738274	0.474736	0.691083
CV run#9	75.4593	0.690323	0.701639	0.798673	0.744498	0.490283	0.695935
CV run#10	74.443	0.700965	0.68125	0.774336	0.73765	0.473305	0.690967
CVAverage	**74.67503**	**0.68836**	**0.689514**	**0.78685**	**0.737605**	**0.475459**	**0.6888**
Test Set	**75.60463**	**0.741602**	**0.684964**	**0.765957**	**0.75378**	**0.502249**	**0.712159**

Bold entries represent average CV results and Test set results

case we used the following evaluation metrics, based on the *TP* (*true positives*), *TN* (*true negatives*), *FP* (*false positive*), and *FN* (*false negative*) numbers:

$$Accuracy = (1-Error) = \frac{TP+TN}{TP+FP+TN+FN},$$

$$TruePositiveRate(TPR)\,or\;\; Recall(R)\,or\;\; Sensitivity = \frac{TP}{TP+FN},$$

$$Precision\;\; (P) = \frac{TP}{TP+FP},$$

$$Specificity = \frac{TN}{FP+TN},$$

$$FalsePositiveRate\;\; (FPR)\;\; or\;\; (1-specificity) = \frac{FP}{FP+TN},$$

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

$$F-measure = 2\times\frac{(precision\times recall)}{(precision+recall)}$$

AUC is calculated by using an average of a number of trapezoidal approximations over TPR versus FPR curve [49, 50]. The Matthews correlation coefficient is used in our work as a measure of the quality of binary (two-class) classifications and F-measure is used as a measure of the test's accuracy.

To analyze the performance of the developed technique, we have designed a two-stage evaluation strategy. In the first stage, the overall dataset is divided into two parts with the ratio 88:12 to define the CV set and the test set respectively. Then ten-fold cross validation is done with the CV set. In the second stage, the optimum network (chosen from the best of the ten runs during the CV experiment) is selected to evaluate the performance over the independent test set. For CV experiments with the *E. coli* proteome, we have randomly chosen 2893 positive interactions out of total 3254 interactions, and 4399 negative interactions from 4948 data samples. The nonlinear support vector machine classifier with *polynomial* kernel function of degree 5 is used during classification experiments over the CV set and the test set. A comparative result with different kernel functions (together with *polynomial* kernel function of degree 3) on the dataset for *E. coli* proteome is included in Supplementary Table S1. It may be observed that the current choice of *polynomial* kernel function of degree 5 gives

Table 3 Result on AUC optimized network over *Homo sapiens* CV set and test set

Run	Accuracy	Recall	Precision	Specificity	AUC	MCC	F-measure
CV run#1	83.6872	0.741379	0.821656	0.897623	0.819501	0.652729	0.779456
CV run#2	83.9286	0.73639	0.831715	0.904936	0.820663	0.657941	0.781155
CV run#3	84.0402	0.74212	0.830128	0.903108	0.822614	0.660444	0.783661
CV run#4	84.8214	0.739255	0.851485	0.917733	0.828494	0.677197	0.791411
CV run#5	83.2589	0.74212	0.811912	0.890311	0.816216	0.644075	0.775449
CV run#6	84.581	0.767241	0.824074	0.895795	0.831518	0.672559	0.794643
CV run#7	82.1429	0.74212	0.787234	0.872029	0.807075	0.621285	0.764012
CV run#8	81.9196	0.716332	0.798722	0.884826	0.800579	0.614878	0.755287
CV run#9	85.2679	0.767908	0.840125	0.906764	0.837336	0.687094	0.802395
CV run#10	84.1518	0.759312	0.820433	0.893967	0.82664	0.663478	0.78869
CVAverage	**83.77995**	**0.745418**	**0.821748**	**0.896709**	**0.821064**	**0.655168**	**0.781616**
Test Set	**72.27191**	**0.721839**	**0.624254**	**0.72328**	**0.722559**	**0.436224**	**0.66951**

Bold entries represent average CV results and Test set results

Table 4 Result on AUC optimized network over the multi-organism meta-data CV set and test set

Run	Accuracy	Recall	Precision	Specificity	AUC	MCC	F-measure
CV run#1	78.5415	0.718354	0.735421	0.829624	0.773989	0.550257	0.726788
CV run#2	80.2681	0.739451	0.757838	0.844336	0.791894	0.586334	0.748532
CV run#3	81.4489	0.742887	0.779867	0.86171	0.802298	0.609998	0.760928
CV run#4	80.176	0.712025	0.771429	0.860918	0.786472	0.58178	0.740538
CV run#5	80.4858	0.724974	0.770437	0.85754	0.791257	0.589146	0.747014
CV run#6	79.3884	0.690928	0.766979	0.86171	0.776319	0.564125	0.72697
CV run#7	80.1341	0.739451	0.755388	0.842142	0.790797	0.583781	0.747335
CV run#8	80.1508	0.748156	0.751323	0.836692	0.792424	0.585272	0.749736
CV run#9	79.5978	0.71519	0.757542	0.849201	0.782195	0.570451	0.735757
CV run#10	79.6482	0.726027	0.753005	0.842946	0.784487	0.572722	0.73927
CVAverage	**79.98396**	**0.725744**	**0.759923**	**0.848682**	**0.787213**	**0.579387**	**0.742287**
Test Set	**76.13293**	**0.722739**	**0.69063**	**0.786748**	**0.754744**	**0.510918**	**0.708903**

Bold entries represent average CV results and Test set results

superior performance in the current experimental setup. For all subsequent experiments we therefore use this setup and report the classification performances accordingly.

During CV, training is performed on three different optimization criterion, viz., Recall, precision and AUC scores. Ten CV experiment runs are marked as *run*#1, *run*#2… run#10. We present the results over the *E. coli* CV set for all ten runs, the average CV performance and the performance over the test set using AUC optimized network in Table 1. Corresponding performances with the recall and precision optimized networks are given in Tables S2 and S3 respectively, in the Supplementary material. AUC, recall and precision optimized training strategies are discussed in our earlier works [51–54].

To allow some flexibility in the training program, SVM models have a cost parameter, c, that controls the trade-off between allowing training errors and forcing rigid margins. It creates a *soft margin* that permits some misclassifications. Increasing the value of c increases the cost of misclassifying points and forces the creation of a more accurate model that may not generalize well. In the support-vector networks algorithm, one can control the trade-off between complexity of decision rule and frequency of error by changing the parameter c [55]. In our work we have varied the value of c in the range (16, 316). The *gamma* (γ) and *degree* of polynomial kernel determine the ability of the resulting SVM in fitting the data. We can also vary the kernel coefficient (r), to make the kernel non-symmetric. The intuitive meaning of *gamma* is the amount of influence of a support vector upon its surroundings. In the current work we have heuristically chosen: $0 \leq \gamma \leq 32$, and $0 \leq r \leq 300$. During any run of the CV experiment (run_i), the optimum set of kernel parameters are estimated as P_i. We generate one-model files (m_1), over the complete CV set using the best set of kernel parameters (chosen on the basis of best AUC scores during CV). For three different optimization experiments, three different model files (m_1,m_2,m_3) are generated. Performances of these model files (m_i) over the *E. coli* test set are evaluated and the best results for the AUC optimization is reported in the last row of Table 1. Figure S4 in the Supplementary data sheet shows a performance analysis over the *E. coli* dataset during the CV experiment.

In the same way, we have prepared interacting and non-interacting residue fragments and extracted HQI8 features for both positive and negative data samples for the *Yeast* organism. For the CV experiment, 3103 positive interactions have been randomly chosen from 3490 positive interactions, and 4518 negative interactions are chosen for 5082 data samples. AUC optimized results of ten-fold CV experiment and over the independent test set are shown in Table 2, and the corresponding results with the recall and precision optimization are shown in Tables S4 and S5 respectively (see Supplementary data). Figure S5 (see Supplementary data) shows the performance analysis over the *Yeast* dataset during the CV experiment using three different optimization strategies.

For the CV experiment with the *Homo sapiens* organism, 3488 positive and 5470 negative interactions are randomly selected from the total set of 3923 and 6153 interactions respectively. The AUC optimized results of ten-fold CV

Table 5 Comparison of our current work with the existing techniques

	Methods	AUC	Sensitivity	Specificity
1	Wang et. al. [56]	0.71933	0.68640	0.65417
2	Nguyen et. al. [57]	0.74943	0.3598	0.92949
3	Deng et. al. [39]	0.79761	0.76765	0.63158
4	Borderner et. al. [58]	–	0.57	0.26
5	Singh et. al. [44]	–	0.6	0.75
6	PPIcons(*E.coli*)	0.814687	0.736842	0.892532
7	PPIcons(*Yeast*)	0.75378	0.741602	0.765957
8	PPIcons(*Homo sapiens*)	0.722559	0.721839	0.72328
9	PPIcons (meta-data)	0.754744	0.722739	0.786748

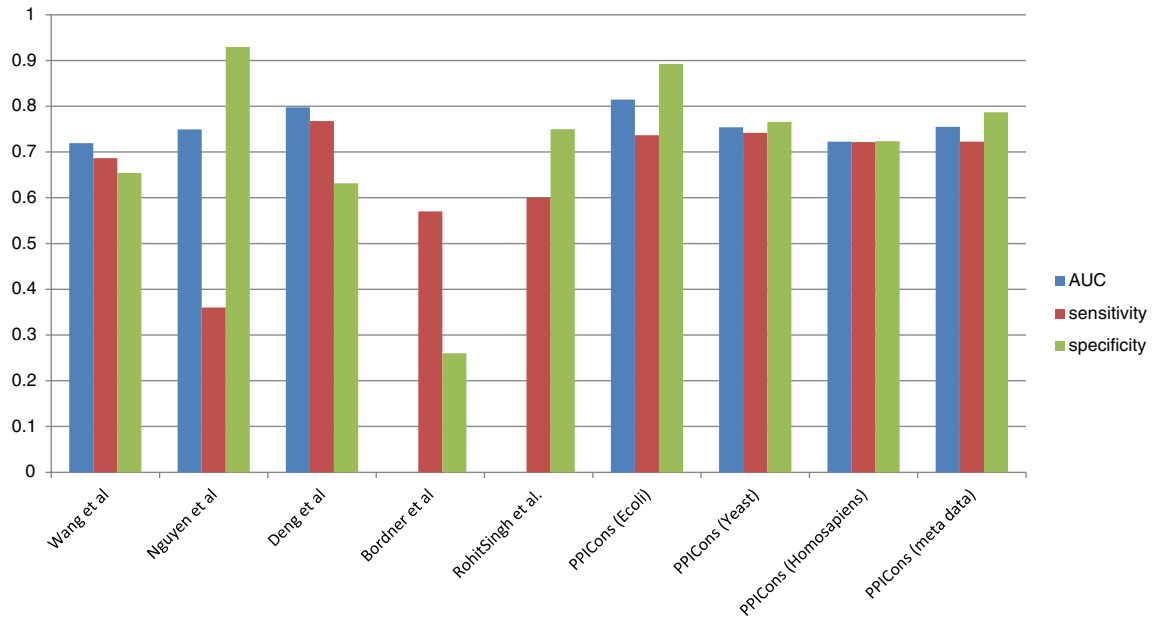

Fig. 2 The performance on testing dataset of PPIcons in comparison with the existing state-of-the-art tools

experiment and over the independent test set are shown in Table 3, and the recall and precision optimized results are shown in Tables S6 and S7, in the Supplementary data sheet. Figure S7 (see in Supplementary data sheet) shows a performance analysis over the *Homo sapiens* dataset during the CV experiment using three different optimization strategies.

Finally, in the case of meta-dataset experiment, 9484 positive and 14,387 negative interactions are randomly selected from the combined data set (collected from the aforementioned three organisms in the same ratio of train and test as discussed above in individual cases) of 9667 positive and 16,183 negative interactions respectively, in the ratio of 88:12 to generate the CV set and the test set. The AUC optimized results of the ten-fold CV experiment and over the independent test set are shown in Table 4 and the corresponding recall and precision results are shown in Tables S8 and S9 respectively (see Supplementary data). Figure S8, in Supplementary sheet, shows a performance analysis of the meta dataset during the CV experiment, over the independent test set using three different optimization strategies.

We compared our results with similar works reported previously in the literature. In the work of Wang et al. [56] position specific scoring matrices (PSSMs) were used along with evolutionary conservation score for 11 neighbor residues. They obtained 71.9 % AUC, 68.6 % sensitivity and 65.4 % specificity over their dataset of 113 pairs of interacting proteins. Nguyen et al. [57] used PSSMs and accessible surface areas (ASA) with 15 neighbor residue to get 74.9 % AUC, 35.9 % sensitivity and 92.9 % specificity scores over 77 individual proteins collected from the Protein Data Bank. Both the above methods used SVM pattern classifier. Deng et al. [39] used an ensemble method with weighted voting strategy along with SVM approach and achieved 79.7 % AUC, 76.7 % sensitivity and 63.1 % specificity over 54 hetero-complexes. Bordner and Abagyan [58] achieved 76 % accuracy, 57 % recall and 26 % precision over 1494 protein-protein interfaces, of which 518 were homodimers, 114 were heterodimers and 862 were multimers. Singh et al. [44] obtained 60 % sensitivity and 75 % specificity in their Struct2Net web server.

In comparison, our results are prepared using 196 hetero-complexes (40 for *E. coli*, 123 for *Yeast*, 33 for *Homo sapiens*) and obtained up to 81.46 % AUC, 73.68 % sensitivity (or recall) and 89.25 % specificity (see Table 1) over our *E. coli* test dataset. For *Yeast* test data, we have obtained 75.4 % AUC, 74.2 % sensitivity (or recall) and 76.6 % specificity (see Table 2). For *Homo sapiens* test data, we have obtained 72.3 % AUC, 72.2 % sensitivity (or recall) and 72.3 % specificity (see Table 3). Finally, in the case of meta-dataset, we have 75.5 % AUC, 72.3 % sensitivity, 78.7 % specificity (see Table 4). We have also calculated the MCC for all three organisms *E. coli*, *Yeast*, *Homo sapiens* which are 64.26 %, 50.23 %, 43.62 % (given in Tables 1, 2 and 3) respectively and finally for meta dataset, it is 50.59 % (see Table 4). The F-measures are also calculated for all three organisms *E. coli*, *Yeast*, *Homo sapiens* which are 77.55 %, 71.22 %, 66.95 % (given in Tables 1, 2 and 3) respectively and finally for meta dataset, it is 70.63 % (see Table 4). We have also added Table 5 and Fig. 2 for easy comparison of our work with the existing ones available in the literature.

Although the performances of different existing techniques are not evaluated over an identical test-bed (due to large variations in the datasets), the PPIcons results over the

196 hetero-complexes are found to be comparable with the existing state-of-the-art tools. In fact, the reported numbers show that PPIcons performance is better than most of the other prediction tools. For example, the AUC score of the meta-data PPIcons is higher than all but the one designed by Deng et al. Our *E. coli* specific PPIcons on the other hand has better AUC score than Deng et al. It may also be noted that work of Deng et al. has higher sensitivity value, but lower specificity value in comparison to our work. Similarly, the work of Nguyen et al. has lower sensitivity but higher specificity in comparison to PPIcons. In general, our predictors are found to be stable and reports balanced prediction results in comparison to the existing systems.

Conclusions

In the present work, we introduce the PPIcons software as a novel and accurate tool for PPI site prediction, using only protein sequences. In the training dataset we have used three dimensional structures of interacting proteins, yet the predictor uses only sequence composition in order to predict which local sequence segments from both proteins are interacting. The distance between all atom pairs are calculated, if it is equal or less than 3.5 Å, the pair is considered as interacting. The local sequence neighborhoods are then considered and HQI8 features vectors are used to represent the continuous, overlapping sliding window of length 21 residues. Finally, support vector machine algorithm with polynomial kernel function of the degree 5 is used to build the statistical learning model for individual organisms and the meta-predictor. This prediction model allows annotating unknown interactions, enriching the biological knowledge about proteins' partners. The current work also provides datasets of interacting hetero-complexes collected from three organisms, viz., *E. coli*, *Yeast* and *Homo sapiens*. Moreover, the results of meta-predictor show that the method is stable over different organisms. The training datasets and the source code for PPIcons tool are available in public domain at http://code.google.com/p/cmater-bioinfo/. The performance of our predictor is better than most of the methods discussed in this paper. Although the datasets used in different works are sometimes different, up to now the general performance scores from different publications are compared in evaluation of different *in silico* methods in PPI domain.

In this paper, we have worked with three different organism specific databases, as well as a combined meta-database. We would like to improve the database by including more organisms in the near future. Due to limitation of computing resources, all interactions could not be considered for training. Despite certain constraints, the current version of PPIcons is observed to generate a steady and balanced prediction result (in terms of AUC score, sensitivity and specificity) over labeled test samples of different organisms. As evident from the discussion in the Results section, the performance of the PPIcons program is found to be comparable or better than the state-of-the-art tools available today. For most of the existing predictors their performances are not balanced, producing high sensitivity, yet low specificity, or vice-versa. Avoiding such a biasing is often difficult in a complex binary classification problem. Considering that, the balanced prediction potential of our developed algorithm may be considered as a good statistical learning characteristic. The PPIcons software tool is also made available for free download in the public domain. In the future we plan to incorporate a larger training/test datasets, incorporating more proteins from *E. coli*, *Yeast*, *Homo sapiens* and other organisms, for design of improved versions of PPIcons. Design of an effective classifier ensemble, for meta-analysis of classification results different experimental sources, may be incorporated in future. Brainstorming consensus [59] or weighted Markov chain based rank aggression approach [60] may be used for the in future to achieve such an objective.

Acknowledgments This work is partially supported by the Polish Ministry of Education and Science and ERASMUS Mundus EC funding. Contributions of second and third authors are also supported by the PURSE project of Computer Science and Engineering Department of Jadavpur University, India.

References

1. Chelliah V, Chen L, Blundell T et al. (2004) Distinguishing structural and functional restraints in evolution inorder to identify interaction sites. J Mol Biol 342:1487–1504
2. Uetz P, Giot L, Cagney G (2000) A comprehensive analysis of protein-protein interactions in Saccharomyces cerevisiae. Nature 403:623–627
3. Ito T, Chiba T, Ozawa R et al. (2001) A comprehensive two-hybrid analysis to explore the Yeast protein interactome. Proc Natl Acad Sci USA 98(8)
4. Gavin A, Bosche M (2002) Functional organization of the Yeast proteome by systematic analysis of protein complexes. Nature 415:141–147
5. Yuen H, Gruhler A, Heilbut A (2002) Systematic identification of protein complexes in Saccharomyces Cerevisiae by mass spectrometry. Nature 415:180–183
6. Gavin A, Aloy P, Grandi P (2006) Proteome survey reveals modularity of the Yeast cell machinery. Nature 440:631–636
7. Krogan N, Cagney G, Yu H et al. (2006) Global landscape of protein complexes in the Yeast Saccharomyces cerevisiae. Nature 440:637–643

8. Korn A, Burnett R (1991) Distribution and complementarity of hydropathy in multi-subunit proteins. Protein Struct Funct Bioinforma 9:37–55
9. Jones S, Thornton JM (1997) Analysis of protein-protein interaction sites using surface patches. JMB 272:121–132
10. Lo Conte L, Chothia C, Janin J (1999) The atomic structure of protein– protein recognition sites. J Mol Biol 285:2177–2198
11. Jones S, Thornton JM (1996) Principles of protein-protein interactions. Proc Natl Acad Sci USA 93:13–20
12. Walls PH, Sternberg MJE (1992) New algorithm to model protein-protein recognition based on surface complementarity: applications to antibody-antigen docking. J Mol Biol 228:277–297
13. Helmer-Citterich M, Tramontano A (1994) A new method for automated protein docking based on surface shape complementarity. J Mol Biol 235:1021–1031
14. Zielenkiewicz P, Rabczenko A (1988) Methods of molecular modelling of protein-protein interactions. Biophys Chem 29:219–224
15. Janin J, Miller S, Chothia C (1988) Surface, subunit interfaces and interior of oligomericproteins. J Mol Biol 204:155–164
16. Miller S (1989) The structure of interfaces between subunits of dimeric and tetrameric proteins. Protein Eng 3:77–83
17. Argos P (1988) An investigation of protein subunit and domain interfaces. Protein Eng 2:101–113
18. Jones S, Thornton J (1995) Protein-protein interactions: a review of protein dimer structures. Prog Biophys Mol Biol 63:31–65
19. Fariselli P, Pazos F, Valencia A et al. (2002) Prediction of protein-protein interaction sites in heterocomplexes with neural networks. Eur J Biochem FEBS 269:1356–1361
20. Rost B, Sander C (1994) Conservation and prediction of solvent accessibility in protein families. Proteins 20:216–226
21. Kabsch W, Sander C (1983) Dictionary of protein secondary structure: pattern of hydrogen-bonded and geometrical features. Biopolymers 22:2577–2637
22. Liu R, Jiang W, Zhou Y (2010) Identifying protein–protein interaction sites in transient complexes with temperature factor, sequence profile and accessible surface area. Amino Acids 38:263–270
23. Chen H, Zhou H-X (2005) Prediction of interface residues in protein–protein complexes by a consensus neural network method: test against NMR data. Proteins 61:21–35
24. Neuvirth H, Raz R, Schreiber G (1980) ProMate: a structure based prediction program to identify the location of protein–protein binding sites. J Mol Biol 338:181–199
25. Liang S, Zhang C, Liu S et al. (2006) Protein binding site prediction using an empirical scoring function. Nucleic Acids Res 34:3698–3707
26. Bradford JR, Westhead DR (2005) Improved prediction of protein-protein binding sites using a support vector machines approach. Bioinformatics 21:1487–1494
27. Porollo A, Meller J (2007) Prediction-based fingerprints of protein–protein interactions. Proteins Struct Funct Bioinforma 66:630–645
28. Qin SB, Zhou H-X (2007) meta-PPISP: a meta web server for protein-protein interaction site prediction. Bioinformatics 23(24):3386–3387
29. Zhou H-X, Shan Y (2001) Prediction of protein interaction sites fromsequence profile and residue neighbor list. Proteins Struct Funct Genet 44:336–343
30. Jang W-H, Jung S-H, Han D-S (2012) A computational model for predicting protein interactions based on multidomain collaboration. IEEE/ACM Trans Comput Biol Bioinforma 9(4):1081–1090
31. Guo Y, Yu L, Wen Z et al. (2008) Using support vector machine combined with auto covariance to predict protein–protein interactions from protein sequences. Nucleic Acids Res 36(9):3025–3030
32. Jordan RA (2012) Structure-based prediction of protein-protein interaction sites. BMC Bioinformatics doi:10.1186/1471-2105-13-41
33. Darby C, Yu-Tang S, Po-Chang L (2010) Predicting the protein-protein interactions using primary structures with predicted protein surface. BMC Bioinformatics doi:10.1186/1471-2105-11-S1-S3
34. Shen XL, Chen YH (2011) Predicting protein interaction sites based on a new integrated radial basis functional neural network. Adv Mater Res 183:387–391
35. Xiong Y, Liu J, Zhang W et al. (2012) Prediction of heme binding residues from protein sequences with integrative sequence profiles. Proteome Sci 10(Suppl 1):S20
36. Ben-Hur A, Noble WS (2006) Choosing negative examples for the prediction of protein-protein interactions. BMC Bioinforma 7(Suppl 1):S2
37. Chen X, Jeong J (2009) Sequence-based prediction of protein interaction sites with an integrative method. Bioinformatics 25(5):585–591
38. Saha I, Maulik U, Bandyopadhyay S et al. (2011) Fuzzy clustering of physicochemical and biochemical properties of amino acids. Amino Acids doi:10.1007/s00726-011-1106-9
39. Deng L, Guan J, Dong Q et al. (2009) Prediction of protein-protein interaction sites using an ensemble method. BMC Bioinforma 10:426
40. Zhang Y, Lin H, Yang Z et al. (2012) Hash subgraph pairwise kernel for protein-protein interaction extraction. IEEE/ACM Trans Comput Biol Bioinformatics doi:10.1109/TCBB.2012.50
41. Pizzut C, Rombo SE (2012) A coclustering approach for mining large protein-protein interaction networks. IEEE/ACM Trans Comput Biol Bioinforma doi:10.1109/TCBB.2011.158
42. Berman H, Westbrook J, Feng Z et al. (2000) The protein data bank. Nucleic Acids Res 28:235–242
43. Salwinski L, Miller CS, Smith AJ et al. (2004) The database of interacting proteins: 2004 update. Nucleic Acids Res 32:D449–D451
44. Singh R, Park D, Xu J et al. (2010) Struct2Net: a web service to predict protein–protein interactions using a structure-based approach. Nucleic Acids Res 38:W508–W515
45. Vapnik V (1995) The nature of statistical learning theory. Springer, New York
46. Suykens JA, Vandewalle J (1999) Least squares support vector machine classifiers. Neural Proces Lett 9(3):293–300
47. Van Der Malsburg C (1986) Frank Rosenblatt: Principles of neurodynamics: perceptrons and the theory of brain mechanisms. Brain Theory 245–248. Springer, Heidelberg
48. Rumelhart DE, McClelland JL (1987) Parallel distributed processing: explorations in the microstructure of cognition. Fondations. MIT press, Cambridge
49. Pruessner JC, Kirschbaum C, Meinlschmid G et al. (2003) Two formulas for computation of the area under the curve represent measures of total hormone concentration versus time-dependent change. Psychoneuroendocrinology 28(7):916–931
50. Fekedulegn DB, Andrew ME, Burchfiel CM et al. (2007) Area under the curve and other summary indicators of repeated waking cortisol measurements. Psychosom Med 69(7):651–659
51. Basu S, Plewczynski D (2010) AMS3.0: prediction of post-translational modifications. BMC Bioinformatics doi:10.1186/1471-2105-11-210
52. Chatterjee P, Basu S, Kundu M et al. (2011) PPI_SVM: prediction of protein-protein interactions using machine learning, do-main-domain affinities and frequency tables. Cell Mol Biol Lett 16(2):264–278
53. Chatterjee P, Basu S, Kundu M et al. (2011) PSP_MCSVM: brainstorming consensus prediction of protein secondary structures using two-stage multiclass support vector machine. J Mol Model 17(9):2191–2201
54. Plewczynski D, Basu S, Saha I (2012) AMS 4.0: consensus prediction of post-translationalmodifications in protein sequences. Amino Acids doi:10.1007/s00726-012-1290-2

55. Cortes C, Vapnik V (1995) Support vector networks. Mach Learn 20(3):273–297
56. Wang B, Chen P, Huang D-S et al. (2006) Predicting protein interaction sites from residue spatial sequence profile and evolution rate. Fed Eur Biochem Soc Lett 580:380–384
57. Nguyen MN, Rajapakse JC (2006) Protein-protein interface residue prediction with SVM using evolutionary profiles and accessible surface areas. CIBCB doi:10.1109/CIBCB.2006.331008
58. Bordner AJ, Abagyan R (2005) Statistical analysis and prediction of protein-protein interfaces. Proteins Struct Funct Bioinforma 60:353–366
59. Plewczynski D (2010) Brainstorming: weighted voting prediction of inhibitors for protein targets. J Mol Model 17:2133–2141
60. Sengupta D, Maulik U, Bandyopadhyay S (2012) Weighted markov chain based aggregation of biomolecule orderings. IEEE/ACM Trans Comput Biol Bioinforma 9(3):924–933