J Mol Model (2008) 14:69–76
DOI 10.1007/s00894-007-0250-3

SOFTWARE REPORT

AutoMotif Server for prediction of phosphorylation sites in proteins using support vector machine: 2007 update

Dariusz Plewczynski · Adrian Tkacz · Lucjan S. Wyrwicz · Leszek Rychlewski · Krzysztof Ginalski

Received: 19 July 2007 / Accepted: 12 October 2007 / Published online: 8 November 2007

Abstract We present here the recent update of AutoMotif Server (AMS 2.0) that predicts post-translational modification sites in protein sequences. The support vector machine (SVM) algorithm was trained on data gathered in 2007 from various sets of proteins containing experimentally verified chemical modifications of proteins. Short sequence segments around a modification site were dissected from a parent protein, and represented in the training set as binary or profile vectors. The updated efficiency of the SVM classification for each type of modification and the predictive power of both representations were estimated using leave-one-out tests for model of general phosphorylation and for modifications catalyzed by several specific protein kinases. The accuracy of the method was improved in comparison to the previous version of the service (Plewczynski et al., "AutoMotif server: prediction of single residue post-translational modifications in proteins", Bioinformatics 21: 2525–7, 2005). The precision of the updated version reached over 90% for selected types of phosphorylation and was optimized in trade of lower recall value of the classification model. The AutoMotif Server version 2007 is freely available at http://ams2.bioinfo.pl/. Additionally, the reference dataset for optimization of prediction of phosphorylation sites, collected from the UniProtKB was also provided and can be accessed at http://ams2.bioinfo.pl/data/.

Keywords Machine learning · Protein kinases · Phosphorylation · Phosphoserine · Phosphothreonine · Phosphotyrosine · Swiss-Prot database · Post-translational modifications · Support vector machine

D. Plewczynski (✉) · K. Ginalski
Interdisciplinary Centre for Mathematical and Computational Modeling, University of Warsaw,
Pawinskiego 5a,
02-106 Warsaw, Poland
e-mail: D.Plewczynski@icm.edu.pl

A. Tkacz · L. S. Wyrwicz · L. Rychlewski
BioInfoBank Institute,
Limanowskiego 24a,
60-744 Poznan, Poland

L. S. Wyrwicz
Departament of Gastroenterology,
Maria Sklodowska-Curie Memorial Cancer Center,
Roentgena 5,
02-781 Warsaw, Poland

Computational methods for prediction of phosphorylation sites in proteins

Phosphorylation is one of the major types of post-tranlational modification of hydroxyl groups in proteins. The attachment of a phosphorus group to aliphatic (serine, threonine) or aromatic (phenylalanine) amino acids influences on the function of proteins by modifying their local general physicochemical properties and by controlling the behavior of a protein in a living cell, for example by activating or inactivating an enzyme [1]. Phosphorylation has proven to be an important mechanism for controlling intracellular processes. Athough many protein kinases are now known, the identification of their potential biological targets is still ongoing research. Relatively high substrate specificity of protein kinases ensures correct transmission of signals in cells. Such methods may provide rapid automatic annotations, which then in turn can be used as guidelines for further experimental discoveries.

The experimental verification of protein kinases' substrates and their corresponding phosphorylation sites is highly non-trivial and time-consuming. The rapid increase in genomic information and the pressure for the translational clinical research requires new automatic techniques

to investigate protein modifications. Although the structural determinants of a phosphrylated amino acids seems to be an important criteria for an effective phosphorylation, the recent studies showed that in most cases protein sequence is quite sufficient to select modified sites. Therefore most in silico methods process local sequence information around phosphorylated sites.

The specificity of protein kinases is largely determined by the primary sequence of the target site. The simplest approach utilized in ELM server at http://elm.eu.org/ [2] represents local sequence neighborhoods of phosphorylated sites by regular expressions. In addition, ELM applies some context-based filters (taxonomic range, cell compartment and globular organization) in order to improve accuracy of the prediction by filtering out 'disallowed' predictions. The overall usability of the method is limited, due to the low information content of the predicted sites. Therefore other approaches focus, instead of single regular expression based description of short sequence fragments, on more advanced statistical description of local neighborhood of known modification sites. These methods allow in more rigorous way to calculate reliability scores.

ScanSite [3] finds sequence motifs that are recognized by signaling domains, phosphorylated by several kinases or which interact with specific proteins or ligands. The position-specific scoring matrixes are constructed from peptide libraries or phage displays. The conserved sequence motifs represent important biochemical properties or biological functions, as it is presented in PRINTS database [4], or eMOTIF [5, 6]. These resources contain multiple sequence alignments from BLOCKS+ database [7–9].

Consensus approaches combine several signature recognition methods to scan a given query protein sequence against observed protein signatures. Each of such methods returns its own lists of hits and then the hits are combined in a consensus prediction. The PROSITE tools [10–12] recognize short sequence motifs by combining ScanProsite [13, 14], PRATT [15], PPSearch, PROSCAN [16, 17] and PPscan. The InterProScan [18, 19] classifies proteins using different member databases by building the consensus from various databases instead of various in silico methods. Balla et al. [20] provide the overview of many short motifs of post-translational modifications, involved in protein-protein interactions or protein trafficking in cell. The motif database comprises over 300 sequence motifs as well as offers a search tool for detection of these motifs in proteins. Ahmad et al. [21] included the knowledge on three-dimensional structures of proteins in the prediction of post-translational modifications, and suggested that structural changes are dynamic and can result in temporary conformational changes. These changes can interfere with many functions of proteins, as it was demonstrated for phosphorylation [21]. In the work of Senawongse et al. [22] a hidden Markov models (HMM) allowed for selection of important sequence motifs and the appropriate machine learning algorithm of the support vector machine (SVM) was used to classify the functional and nonfunctional feature motifs. The authors proved that consensus method of machine learning and sequence motif identification (HMM) can provide much better accuracy than prediction using sequence motifs or machine learning aproaches alone [22].

Those results strongly supported the use of machine learning algorithms for prediction of phosphorylation sites in proteins. The NetPhos 2.0 server applies neural networks trained on fragments of protein sequences flanking post-translational modification sites from PhosphoBase resources [23, 24] to predict serine, threonine, and tyrosine phosphorylation sites in eukaryotic proteins [24, 25]. The sequence specificities of various protein kinases are calculated for nine amino acid segments surrounding a phosphorylation site. In the PPSP (Prediction of PK-specific Phosphorylation site) server a similar approach based on Bayesian decision theory (BDT) was used to predict potential phosphorylation sites [26]. In comparison with previously mentioned tools (Scansite, NetPhosK, KinasePhos and GPS) the service is more accurate and includes phosphorylation motifs of several additional protein kinases (TRK, mTOR, SyK and MET/RON) [26]. Li et al. used various algorithms of machine learning in prediction of specific post-translational modifications. For instance, the k-nearest neighbor (k-NN) method with Manhattan distance was used for prediction of phosphorylated amino acids [27]. Authors used a simple representation of sequence motifs, as calculated by BLOSUM62 similarity scores [27]. In the program PAIL the lysine acetylation is predicted by Bayesian discriminant method (BDM) algorithm with the accuracy of over 85% in the test proided by the authors [28]. Recently, the new resource dbPTM (http://dbPTM.mbc.nctu.edu.tw/) was developed [29]. The dbPTM database compiles information from Swiss-Prot, PhosphoELM, and O-GLYCBASE on various types of protein post-translational modifications. Additionally, it contains detailed physicochemical information on catalytic sites, solvent accessibility, protein secondary and tertiary structures, protein domains and protein variants. All instances are experimentally validated, yet only three types of modifications, i.e., phosphorylation, glycosylation, and sulfation, are assigned for all proteins from Swiss-Prot database.

Those recent advances of in silico methods [20–22, 26–34] clearly justify the directions for further development of computational methods of machine learning trained on available experimentally verified data to predict phosphorylated residues in proteins. The efficient extraction of the most predictive features from amino acid sequence for further computational modeling is now one of the most challenging tasks of the postgenomic era.

Here we present an updated version of AutoMotif Server that includes an revised methodology and an improved training dataset. The service uses a supervised support vector machine approach to predict various types of phosphorylation sites in proteins. It is based on the classification of the available information on phosphorylations and other post-translational modifications obtained from the UniProt database version 06.2007. This classification is used then to predict phosphorylation sites in proteins. The accuracy of the method was significantly improved in comparison to the previous version, the precision is now over 90% for selected types of phosphorylation. The training database of short sequence fragments corresponding to known phosphorylation types is publicly available for download from our web pages.

The new reference database of PTM sites for training machine learning

The AMS method was trained on known experimental instances of post-translational modification sites available in the 06.2007 version of UniProt (Universal Protein Resource) database [35]. In order to maximize the classification accuracy of applied models all sites annotated as "probable", "potential", and "by similarity" were omitted. The remaining sites were assembled to the dataset of positive instances and fragments of nine amino acids in length centered on the annotated residues were dissected from corresponding proteins sequences. If modified amino acids were located in the N- or C-termini of proteins, additional "X' residues were introduced, so the central position of annotated residue in each segment was preserved. Redundant samples were removed from training data (with an exception for different BLAST profile in PROFILE representation as described below). The dataset with negative instances was built from native protein sequences surrounding a matched central residue not annotated as modified. Negative instances were prepared from the same proteins that contained "true" PTMs. The resulting datasets that can be used for training various machine learning algorithms are publicly available at http://ams2.bioinfo.pl/data/.

The two datasets containing positive and negative instances of each type of functional motifs were used for the training of SVM. All sequence segments from both sets were projected (embedded) on the same abstract multidimensional space in order to build detailed sequence models for each modification. In the current version the available representations were restricted to one optimal (in terms of the speed of calculation) generic representation of a short protein sequence segment. This representation (the binary one called here as BINARY) encodes each position of a segment as a long 20 dimensional vector of discrete values (0 and 1); the number of dimensions corresponds to the number of types of amino acids. The 1 value in a vector is taken if the certain type of amino acid is present at the certain position in a segment and 0 for all other types. This representation uses nine residues long segments and thus has dimension equal to 180 (number of positions multiplied by the number of dimensions of each vector). For each given segment only nine coordinates were equal to 1 (one for each position in a segment represented by 20 values), while all other had a value of 0. The vectors are normalized for each position in a segment separately leaving one dimensional scalar value equal to the normalized sequence preference for it. Normalized preferences were calculated separately for nine positions within a segment and the value for a given amino acid at each position of a segment was calculated by dividing the observed probability to find this amino acid at the exact position in positive segments by the observed probability to find it in the negatives [36–38].

Accuracy of updated AMS for phosphorylation sites

The AMS applies a support vector machine to predict post-translational modifications in protein sequences. The supervised machine learning algorithm [39–41] was first trained on known instances after embedding them into multidimensional feature space. In order to extract relevant information from the heterogeneous data, SVM tries to separate a given set of binary labeled training vectors with an optimal hyperplane. The optimum is reached for hyperplane that maximizes the separating margin between the two classes of the training vectors having relatively small number of support vectors. For the detailed description of support vector machine please refer to [36–47].

The performance of a classification was described by three measures of accuracy: classification error [$E = 100\%*(fp+fn)/(tp+fp+tn+fn)$], recall [$R = 100\%*(tp)/(tp+fp)$], and precision [$P = 100\% * (tp)/(tp+fp)$], where tp is the number of true positives, fp is the number of false positives, tn is the number of true negatives, and fn is the number of false negatives. The classification error E was used to provide an overall error measure, whereas recall R corresponded to the percentage of correct predictions (the fraction of correct predictions), and precision P measured the percentage of observed positives that are correctly predicted (the measure of the reliability of prediction of positive instances). These measures of accuracy can be calculated using conservative but easy to compute Xi-Alpha estimates [48] and more precise but computationally intensive the leave-one-out procedure. The leave-one-out test was applied, by removing from the training data one sample, constructing the model on the basis of the

remaining training dataset and then validating the model on the removed sample. The resulting error estimators were averaged over all such models (for all positive and all negative instances).

We collected results for both types of projections of sequence fragments, separately for each considered type of phosphorylation. The accuracy of the previous version is presented in Table 1. Results for SVM with linear kernel and BINARY representation of input data are shown in Table 2. The results for polynomial kernel with PROFILE representation are given in Table 3. According to the obtained results the polynomial kernel with PROFILE representation is the best type of kernel for all types of modifications. The linear kernel with BINARY representation was more effective only when the number of training cases was large. This can be explained by the high sequence similarity between tested instances in the larger collections of positives. The linear kernel function in the case of more complicated sequence signatures of phosphorylated sites is not efficient, or it cannot produce any model at all. However, in some cases (PKA or PKC phosphorylation sites) SVM models of this type reach efficiency of the polynomial kernel.

The AutoMotif Server 2.0 (version 2007)

The AutoMotif Server (AMS) takes the sequence of a query protein as an input and predicts its phosphorylation sites. In our approach we consider only sequence information, because in most cases only the sequence of a potential target protein is known. The server uses the SVM classification models constructed as described in the previous section. Firstly, it dissects a query protein into overlapping short segments of nine amino acids. For each sequence segment a score using SVM model constructed according to its cost function is assigned. Residues that have the score (the value of cost function which is described in detail on the server's Web page) higher than a given cut-off value are annotated as plausible modification sites. The points representing their sequence segments are lying in the region classified as positive by the SVM model's hyperplane within a given cut-off as the margin value. We use only one, the most effective type of the kernel (the polynomial one) in the web server. Our method is a simple one-vote-wins approach, where we annotate all segments with positive verification by at least one model.

AMS accepts input sequences in the one-letter code in capital letters: (ACDEFGHIKLMNPQRSTVWY) with an additional code (X) for marking empty and unknown positions in a protein chain or positions that extends a sequence segment outside chain's ends. The user can input sequences by submiting text file, entering the SWISS-PROT/TrEMBL identifier (or accession number) or providing seqeuences of query proteins in the text box. It is recommended to use the complete protein sequence, not short fragments of it.

By default the server predicts all types of phosphorylation sites that are available in the Swiss-Prot database, such as phosphorylation by PKC, PKA, CK1, CK2, and CDC2 kinases. Users can limit their searches by choosing particular type of functional motif from the drop-down list on the server's main page (for example phosphorylation sites in general or by specified protein kinase). Two types of search procedures are available: identity search and scan based on SVM method. The first one performs a simple search to identify exact (in terms of sequence) matches of nine residues segments from query protein and the database of positives for

Table 1 The training accuracy for support vector machine supervised learning on data representation from AMS 1.0 (ver. 2004)

Functional motif type	Number of proteins	Number of positives/negatives	Recall/precision of the best method	The best method	Recall/precision of the second best method
phosphorylation by PKA	67	86	42%	***LOOKUP***	42%
		14353	86%		77%
phosphorylation by PKC	49	56	18%	***BLOSUM+LOOKUP***	18%
		14368	91%		83%
phosphorylation by CDC2	18	41	29%	***BIN+LOOKUP***	24%
		14375	32%		33%
phosphorylation by CK2	35	62	21%	***SPARSE+LOOKUP***	18%
		11746	39%		48%
phosphorylation by CK	44	85	13%	***SPARSE+LOOKUP***	11%
		11739	41%		36%

The results are obtained using SVM learning with polynomial kernel. The second column presents the number of proteins used in training. The third column shows the number of positive and negative instances in training. The recall and precision values for the best method are in the fourth column. The name of the best method is listed in the fifth column. The last column provides the recall and precision values for the second best method.

AMS - version 2004:

Table 2 The training accuracy for support vector machine supervised learning on data representation from AMS 2.0 (ver. 2007)

PTM type	Protein agent	Kernel	Positives	Negatives	Error	Precision	Recall
Phosphorylation	Autocatalysis	Linear	229	10000	9.09	0	0
Phosphorylation	autocatalysis	Polynomial (s a*b+c)^d	229	10000	7.94	96.15	13.16
Phosphorylation	CDC2	Linear	84	8290	3.14	80.9	85.71
Phosphorylation	CDC2	Polynomial (s a*b+c)^d	84	8290	3.25	95	67.86
Phosphorylation	PKA	Linear	121	10000	4.96	77.78	63.64
Phosphorylation	PKA	Polynomial (s a*b+c)^d	121	10000	4.96	87.88	52.73
Phosphorylation	PKC	Linear	118	7931	6.66	84.85	32.56
Phosphorylation	PKC	Polynomial (s a*b+c)^d	118	7931	7.93	92.31	13.95
Phosphoserine	–	Linear	12373	10000	19.35	81.58	97.14
Phosphoserine	–	Polynomial (s a*b+c)^d	12373	10000	9.48	91.34	97.06
Phosphoserine	autocatalysis	Linear	64	4392	8.94	100	1.67
Phosphoserine	autocatalysis	Polynomial (s a*b+c)^d	64	4392	8.33	85.71	10
Phosphoserine	CK2	Linear	74	2879	7.49	70.97	29.73
Phosphoserine	CK2	Polynomial (s a*b+c)^d	74	2879	7	100	22.97
Phosphoserine	PKA	Linear	105	6750	4.59	78.31	68.42
Phosphoserine	PKA	Polynomial (s a*b+c)^d	105	6750	4.11	98.15	55.79
Phosphoserine	PKC	Linear	105	4146	7.47	80.95	23.29
Phosphoserine	PKC	Polynomial (s a*b+c)^d	105	4146	8.47	69.23	12.33
Phosphothreonine	–	Linear	2295	10000	19.52	60.86	43.22
Phosphothreonine	–	Polynomial (s a*b+c)^d	2295	10000	15.9	75.46	46.12
Phosphothreonine	autocatalysis	Linear	61	2490	9.09	0	0
Phosphothreonine	autocatalysis	Polynomial (s a*b+c)^d	61	2490	9.09	0	0
Phosphotyrosine	–	Linear	1037	10000	14.11	0	0
Phosphotyrosine	–	Polynomial (s a*b+c)^d	1037	10000	13.32	71.2	9.32
Phosphotyrosine	autocatalysis	Linear	92	2772	9.09	0	0
Phosphotyrosine	autocatalysis	Polynomial (s a*b+c)^d	92	2772	9.86	12.5	1.41

that type of modification. The second one runs SVM search with a collection of various embedding methods.

The output page of the service contains two main parts. The first one provides a detailed description of each prediction method and post-translational modification pattern. For each SVM model the server lists a number of positive and negative instances used in training and the precision and recall errors calculated during the training phase. The second part of the output provides results of prediction for each model. It contains information about the protein sequence, a local segment sequence predicted as a modificated site, its position and the output score with value in the range [0.000–5.000], where higher output scores correspond to higher confidence of the prediction.

Table 3 The training accuracy for support vector machine supervised learning with polynomial kernel on PROFILE data representation

Phosphorylation type	Substrate	Recall 2004	Precision 2004	Recall 2007	Precision 2007
Phosphoserine	S	x	x	19%	78%
Phosphothreonine	T	x	x	7%	95%
Phosphotyrosine	Y	x	x	10%	95%
Phosphoserine PKC	S	x	x	4%	100%
Phosphoserine PKA	S	x	x	12%	88%
Phosphotyrosine autocatalysis	Y	x	x	51%	77%
Phosphoserine CK2	S	11%	53%	6%	80%
Phosphothreonine autocatalysis	T	x	x	22%	100%
Phosphoserine autocatalysis	S	x	x	3%	100%
Phospho PKA	S T	41%	75%	14%	86%
Phospho PKC	S T	17%	83%	5%	100%
Phospho autocatalysis	Y H S T	33%	71%	43%	96%
Phospho CDC2	S T	9%	20%	13%	78%

The types of phosphorylation that were not present in the previous version of Swiss-Prot database are marked by "x" in Recall/Precision columns of AMS 1.0 server (ver 2004).

Table 4 The training accuracy for support vector machine supervised learning with polynomial or linear kernel on experimental data from Phospho.ELM

Protein agent	Kernel	Positives	Negatives	Error	Precision	Recall
General	Linear	12103	3546	17.53	83.22	96.88
AMPK_group	Polynomial (s a*b+c)^d	32	320	8.52	100	6.25
ATM	Linear	57	570	2.55	81.54	92.98
ATM	Polynomial (s a*b+c)^d	57	570	2.87	91.49	75.44
CaM-KIIalpha	Linear	36	360	5.81	88.24	41.67
CaM-KII_group	Linear	55	550	8.43	66.67	14.55
CaM-KII_group	Polynomial (s a*b+c)^d	55	550	7.93	88.89	14.55
CDK1	Linear	139	1390	7.52	63.04	41.73
CDK1	Polynomial (s a*b+c)^d	139	1390	7.91	65	28.06
CDK2	Polynomial (s a*b+c)^d	70	700	9.22	45.45	7.14
CDK_group	Linear	102	1020	6.33	67.03	59.8
CDK_group	Polynomial (s a*b+c)^d	102	1020	5.97	79.66	46.08
CK2 alpha	Linear	118	1180	7.24	67.65	38.98
CK2 alpha	Polynomial (s a*b+c)^d	118	1180	7.24	73.08	32.2
CK2_group	Linear	240	2400	6.67	72.22	43.33
CK2_group	Polynomial (s a*b+c)^d	240	2400	6.21	82.76	40
GSK-3beta	Linear	49	490	7.98	75	18.37
GSK-3_group	Linear	32	320	8.52	66.67	12.5
IGF1R	Linear	23	118	12.06	100	26.09
InsR	Polynomial (s a*b+c)^d	45	213	17.05	60	6.67
Lck	Linear	51	510	8.73	60	11.76
MAPK1	Linear	170	1700	6.95	67.24	45.88
MAPK1	Polynomial (s a*b+c)^d	170	1700	7.11	67.29	42.35
MAPK3	Linear	83	830	4.16	78.48	74.7
MAPK3	Polynomial (s a*b+c)^d	83	830	4.71	88.46	55.42
MAPK8	Linear	34	340	9.63	41.67	14.71
MAPK_group	Linear	51	510	5.53	77.78	54.9
MAPK_group	Polynomial (s a*b+c)^d	51	510	7.13	100	21.57
PDK-1	Linear	28	280	5.84	85.71	42.86
PDK-1	Polynomial (s a*b+c)^d	28	280	6.82	81.82	32.14
PKA alpha	Linear	33	330	6.06	82.35	42.42
PKA alpha	Polynomial (s a*b+c)^d	33	330	7.99	83.33	15.15
PKA_group	Linear	325	3250	4.9	82.61	58.46
PKA_group	Polynomial (s a*b+c)^d	325	3250	4.36	90.43	58.15
PKB_group	Linear	84	840	5.74	65.05	79.76
PKB_group	Polynomial (s a*b+c)^d	84	840	4.65	87.27	57.14
PKC alpha	Linear	132	1320	7.99	72.22	19.7
PKC alpha	Polynomial (s a*b+c)^d	132	1320	8.4	77.78	10.61
PKC_group	Linear	238	2380	7.37	80	25.21
PKC_group	Polynomial (s a*b+c)^d	238	2380	7.33	84.85	23.53
Syk	Polynomial (s a*b+c)^d	45	316	13.57	30	6.67

The linear motif size was taken as nine amino acids, and we used BINARY representation of amino acids.

Summary

The current version of AMS server allows for quick and more accurate prediction of protein modification sites. The high overall precision allows a user to gain deep insight in plausible phosphorylation characteristics of proteins of interest. The method was trained on newly released experimental data from Swiss-Prot database. The classification is now optimized for higher precision in trade of lower recall value. The current version of algorithm can be used independently from the Web interface upon request from authors and can be applied to large scale genome analyses.

In addition to previous versions of AutoMotif Server we have developed classification models for protein sequence fragments taken from Phospho.ELM database [49]. The Phospho.ELM resource available at http://phospho.elm.eu.org contains a variety of experimentally verified phosphorylation sites manually curated from the literature. Phospho.ELM constitutes the largest searchable collection of phosphorylation sites available to the research community. The typical

Phospho.ELM entry stores information about substrate proteins with the exact positions of residues known to be phosphorylated by cellular kinases, literature references, subcellular compartment, tissue distribution, and information about the signaling pathways or protein interactions. Phospho.ELM version 2.0 contains 1703 phosphorylation site instances for 556 phosphorylated proteins. Table 4 contain the accuracy of support vector machine algorithm trained by us on various types of phosphorylation taken from the Phospho.ELM dataset.

One of the main problems of post-translational modification prediction is the insufficient number of experimentally verified instances for each type of modifications. On the other hand, even using the updated experimental data, the further development of our automatic method for functional sites annotation should receive a significant improvement when using statistical algorithms in order to quantify in a more rigorous way the results. In our approach the number of support vectors for some models is large, which is explained by the large dimensionality of the embedded space in such cases and the complicated shape of the separation hyperplane between positive and negative instances. The number of support vectors can be lowered when one chooses low dimensional initial encoding of the amino acids into the general physicochemical properties (like polarity, volume, surface area, bulkiness or refractivity - compare work of Lohmann et al. [50]).

Acknowledgements This work was supported by EC (LHSG-CT-2003-503265), NIH (1R01GM081680-01), EMBO Installation, FNP (FOCUS) and MNiSW (PBZ-MNiI-2/1/2005, N401 050 32/1181) grants.

References

1. Web Resources http://www.ncbi.nlm.nih.gov/sites/entrez?Db=pubmed&Cmd=ShowDetailView&TermToSearch=16448870&ordinalpos=5&itool=EntrezSystem2.PEntrez.Pubmed.Pubmed_ResultsPanel.Pubmed_RVDocSum, http://www.plosone.org/article/fetchArticle.action;jsessionid=25641689BCDA437BC10254AB6634C83F?articleURI=info%3Adoi%2F10.1371%2Fjournal.pone.0000656, http://en.wikipedia.org/wiki/Posttranslational_modification. 2007
2. Puntervoll P, Linding R, Gemund C, Chabanis-Davidson S, Mattingsdal M, Cameron S, Martin DM, Ausiello G, Brannetti B, Costantini A, Ferre F, Maselli V, Via A, Cesareni G, Diella F, Superti-Furga G, Wyrwicz L, Ramu C, McGuigan C, Gudavalli R, Letunic I, Bork P, Rychlewski L, Kuster B, Helmer-Citterich M, Hunter WN, Aasland R, Gibson TJ (2003) Nucleic Acids Res 31 (13):3625–3630
3. Obenauer JC, Cantley LC, Yaffe MB (2003) Nucleic Acids Res 31(13):3635–3641
4. Attwood TK, Bradley P, Flower DR, Gaulton A, Maudling N, Mitchell AL, Moulton G, Nordle A, Paine K, Taylor P, Uddin A, Zygouri C (2003) Nucleic Acids Res 31(1):400–402
5. Huang JY, Brutlag DL (2001) Nucleic Acids Res 29(1):202–204
6. Nevill-Manning CG, Wu TD, Brutlag DL (1998) Proc Natl Acad Sci USA 95(11):5865–5871
7. Henikoff JG, Greene EA, Pietrokovski S, Henikoff S (2000) Nucleic Acids Res 28(1):228–230
8. Henikoff JG, Henikoff S, Pietrokovski S (1999) Nucleic Acids Res 27(1):226–228
9. Henikoff S, Henikoff JG, Pietrokovski S (1999) Bioinformatics 15 (6):471–479
10. Falquet L, Pagni M, Bucher P, Hulo N, Sigrist CJ, Hofmann K, Bairoch A (2002) Nucleic Acids Res 30(1):235–238
11. Hofmann K, Bucher P, Falquet L, Bairoch A (1999) Nucleic Acids Res 27(1):215–219
12. Sigrist CJ, Cerutti L, Hulo N, Gattiker A, Falquet L, Pagni M, Bairoch A, Bucher P (2002) Brief Bioinform 3(3):265–274
13. de Castro E, Sigrist CJ, Gattiker A, Bulliard V, Langendijk-Genevaux PS, Gasteiger E, Bairoch A, Hulo N (2006) Nucleic Acids Res 34(Web Server issue):W362–W365
14. Gattiker A, Gasteiger E, Bairoch A (2002) Appl Bioinformatics 1 (2):107–108
15. Jonassen I, Collins JF, Higgins DG (1995) Protein Sci 4(8):1587–1595
16. Blinov NN Jr, Gurzhiev AN, Gurzhiev SN, Kostritskii AV (2004) Med Tekh (5):47
17. Gurhiev AN, Gurzhiev SN, Kirichenko MG, Kostritskii AV (2005) Med Tekh (5):45–48
18. Quevillon E, Silventoinen V, Pillai S, Harte N, Mulder N, Apweiler R, Lopez R (2005) Nucleic Acids Res 33(Web Server issue):W116–W120
19. Zdobnov EM, Apweiler R (2001) Bioinformatics 17(9):847–848
20. Balla S, Thapar V, Verma S, Luong T, Faghri T, Huang CH, Rajasekaran S, del Campo JJ, Shinn JH, Mohler WA, Maciejewski MW, Gryk MR, Piccirillo B, Schiller SR, Schiller MR (2006) Nat Methods 3(3):175–177
21. Ahmad I, Hoessli DC, Walker-Nasir E, Choudhary MI, Rafik SM, Shakoori AR (2006) J Cell Biochem 99(3):706–718
22. Senawongse P, Dalby AR, Yang ZR (2005) J Chem Inf Model 45 (4):1147–1152
23. Blom N, Kreegipuu A, Brunak S (1998) Nucleic Acids Res 26 (1):382–386
24. Kreegipuu A, Blom N, Brunak S (1999) Nucleic Acids Res 27 (1):237–239
25. Blom N, Gammeltoft S, Brunak S (1999) J Mol Biol 294 (5):1351–1362
26. Xue Y, Li A, Wang L, Feng H, Yao X (2006) BMC Bioinformatics 7:163
27. Li A, Wang L, Shi Y, Wang M, Jiang Z, Feng H (2005) Conf Proc IEEE Eng Med Biol Soc 6:6075–6078
28. Li A, Xue Y, Jin C, Wang M, Yao X (2006) Biochem Biophys Res Commun 350(4):818–824
29. Lee TY, Huang HD, Hung JH, Huang HY, Yang YS, Wang TH (2006) Nucleic Acids Res 34(Database issue):D622–D627
30. Chen H, Xue Y, Huang N, Yao X, Sun Z (2006) Nucleic Acids Res 34(Web Server issue):W249–W253
31. Li S, Liu B, Zeng R, Cai Y, Li Y (2006) Comput Biol Chem 30 (3):203–208
32. Monigatti F, Hekking B, Steen H (2006) Biochim Biophys Acta 1764(12):1904–1913
33. Xue Y, Chen H, Jin C, Sun Z, Yao X (2006) BMC Bioinformatics 7:458
34. Zhou F, Xue Y, Yao X, Xu Y (2006) Bioinformatics 22(7):894–896
35. Bairoch A, Apweiler R (1999) Nucleic Acids Res 27(1):49–54
36. Plewczynski D, Tkacz A, Godzik A, Rychlewski L (2005) Cell Mol Biol Lett 10(1):73–89
37. Plewczynski D, Tkacz A, Wyrwicz LS, Godzik A, Kloczkowski A, Rychlewski L (2006) J Mol Model 12(4):453–461
38. Plewczynski D, Tkacz A, Wyrwicz LS, Rychlewski L (2005) Bioinformatics 21(10):2525–2527
39. Cristianini N, Shawe-Taylor J (2000) An introduction to support vector machines : and other kernel-based learning methods. 2000, Cambridge University Press, Cambridge, U.K.; New York, p 189, xiii

40. Vapnik VN (1995) The nature of statistical learning theory. Springer, New York, p 188, xv
41. Vapnik VN (1998) Statistical learning theory. Adaptive and learning systems for signal processing, communications, and control. Wiley, New York, p 736, xxiv
42. Byvatov E, Fechner U, Sadowski J, Schneider G (2003) J Chem Inf Comput Sci 43(6):1882–1889
43. Furey TS, Cristianini N, Duffy N, Bednarski DW, Schummer M, Haussler D (2000) Bioinformatics 16(10):906–914
44. Kim H, Park H (2003) Protein Eng 16(8):553–560
45. Schölkopf B, Burges CJC, Smola AJ (1999) Advances in kernel methods : support vector learning. MIT Press, Cambridge, MA, p 376, vii
46. Zavaljevski N, Stevens FJ, Reifman J (2002) Bioinformatics 18 (5):689–696
47. Zien A, Ratsch G, Mika S, Scholkopf B, Lengauer T, Muller KR (2000) Bioinformatics 16(9):799–807
48. Joachims T (2002) Learning to classify text using support vector machines. Kluwer international series in engineering and computer science ; SECS 668. Kluwer, Boston, p 205, xvi
49. Diella F, Cameron S, Gemund C, Linding R, Via A, Kuster B, Sicheritz-Ponten T, Blom N, Gibson TJ (2004) BMC Bioinformatics 5:79
50. Lohmann R, Schneider G, Behrens D, Wrede P (1994) Protein Sci 3(9):1597–1601