Software

Sirius PSB: A Generic System for Analysis of Biological Sequences

Chuan Hock Koh, Sharene Lin and Limsoon Wong

Address: School of Computing, National University of Singapore, COM1, Law Link, Singapore 117590

Email: Chuan Hock Koh* – kohchuanhock@alumni.nus.edu.sg; Sharene Lin – sharene@alumni.nus.edu.sg; Limsoon Wong – wongls@comp.nus.edu.sg

*Corresponding author

Abstract

Background

Functional sites such as transcription start sites, translation initiation sites and polyadenylation sites influence virtually all aspects of the gene expression process. A general approach for computational recognition of these sites consists of feature generation, feature selection, feature integration and possibly also the construction of cascade classifiers. In this paper, we describe a software tool, Sirius Prediction Systems Builder (PSB), that supports this approach.

Results

Using PSB, we have built two prediction models, one for the recognition of Arabidopsis polyadenylation sites and another for the subcellular localization of proteins. Both systems are competitive against current state-of-the-art models based on evaluation of public datasets.

Conclusion

On top of being able to produce high-quality results, PSB is hassle-free and it greatly reduces the time required to build a prediction model. In place of programming languages, it has a user-friendly graphical user interface, taking the burden of programming off users. It also has a genetic algorithm that assists in the feature generation step, so that users no longer need to spend extended periods deciding on the features to generate. Prediction models built can easily be saved and reused, and can even be put online as prediction servers.

Background

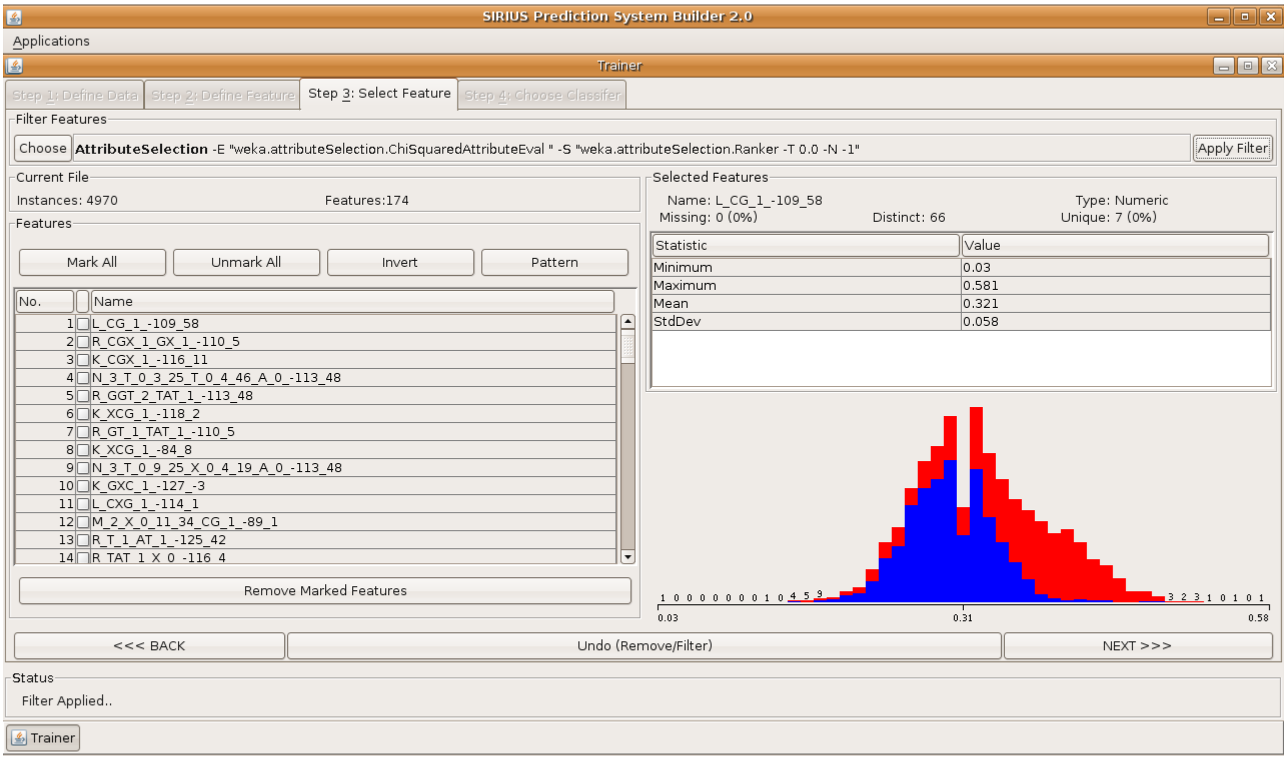
With the advancement of sequencing technologies and mass spectrometry, large amount of data in the form of genome sequences and protein sequences are produced daily [1,2,3]. With so many bio-sequences widely and easily accessible, the next challenge is to mine information from them. In particular, to be able to determine from a DNA sequence its functional sites such as transcription start sites, translation initiation sites, and polyadenylation sites has always been of interest to biologists as functional sites influence virtually all aspects of the gene expression process. In another example, the ability to determine the subcellular localization of the protein from a protein sequence can give biologists clues to the functions of that protein.

There exist numerous approaches in building computer models to carry out bio-sequence analysis. Here, we focus on one particular approach that has successfully built many high-quality computer models [4,5,6,7,8]. The approach consists of the following sequential steps: 1) Feature Generation, 2) Feature Selection, 3) Feature Integration [6] and 4) Cascade Classifier [7].

One popular machine learning package that is often used to carry out the feature selection and feature integration step is Waikato Environment for Knowledge Analysis (WEKA) [9]. The biggest strength of WEKA is that it has implemented a wide variety of feature selection and machine learning algorithms. However, being a general machine learning package, it does not support the feature generation and cascade classifier step which is often used in analyzing biological sequences. As such, the use of WEKA in analyzing biological sequences is less robust and comprehensive.

Another software, FunSiP [10], which was recently developed, is a software platform that implemented the first three steps of the approach but not the fourth step. Partially due to that, the system does not support prediction of functional sites that do not have conserved sequences characterizing the site. As many functional sites of organisms do not have conserved sequences [7], FunSiP is restrictive in that it can only be used for selected problems.

In an attempt to have a more comprehensive software that will enable a wide range of functional sites prediction systems to be built, saved and deployed with ease and speed, we have developed a tool named Sirius Prediction System Builder (Sirius PSB).



AttributeSelection -E "weka.attributeSelection.ChiSquaredAttributeEval " -S "weka.attributeSelection.Ranker -T 0.0 -N -1"

Instances: 4970 Features:174

No.		Name
1	□	L_CG_1_-109_58
2	□	R_CGX_1_GX_1_-110_5
3	□	K_CGX_1_-116_11
4	□	N_3_T_0_3_25_T_0_4_46_A_0_-113_48
5	□	R_GGT_2_TAT_1_-113_48
6	□	K_XCG_1_-118_2
7	□	R_GT_1_TAT_1_-110_5
8	□	K_XCG_1_-84_8
9	□	N_3_T_0_9_25_X_0_4_19_A_0_-113_48
10	□	K_GXC_1_-127_-3
11	□	L_CXG_1_-114_1
12	□	M_2_X_0_11_34_CG_1_-89_1
13	□	R_T_1_AT_1_-125_42
14	□	R_TAT_1_X_0_-116_4

Name: L_CG_1_-109_58 Type: Numeric
Missing: 0 (0%) Distinct: 66 Unique: 7 (0%)

Statistic	Value
Minimum	0.03
Maximum	0.581
Mean	0.321
StdDev	0.058

Filter Applied..

Figure 1. Screen shot of Sirius PSB.

Implementation

In this section, we discuss the general approach that Sirius PSB has been developed to support — feature generation, feature selection, feature integration and cascade classification — which has been used repeatedly in the building of computer models that perform bio-sequence analysis, especially for functional sites prediction on DNA sequences. The current version of Sirius PSB is 2.1.

Feature Generation

In the feature generation step, a set of candidate features are generated based on the bio-sequences as sequences are usually not suitable to be used directly by machine learning techniques.

In this step, one feature type that is frequently used is the *k-gram* feature. It is typically used because it is an easy type of feature to extract and compute. It has also been shown that high-quality classifiers can be produced just by using them [5,7].

A *k-gram* feature is simply a string of k consecutive characters and the frequency of that string in a window location. The characters are usually one of the symbols according to the International Union of Pure and Applied Chemistry (IUPAC) depending on whether the bio-sequence of interest is a genomic or peptide sequence. The window location indicates the part of the sequence that we consider when calculating the number of occurrences of the string.

Other features that are closely related to *k-gram* feature include *ratio of k-gram* feature and *multiple k-gram* feature. *Ratio of k-gram* feature is where we calculate the ratio of two *k-gram* features in a certain window location. *Multiple k-grams* are two or more *k-gram* features that occur one after another within a specific distance range of each other in the window.

Although Sirius PSB currently only supports three feature types namely *k-gram, ratio of k-gram* and *multiple k-gram*, the feature generation step is not limited to only those three feature types. It really depends on the individual's reasoning to decide what type of features to extract when given a sequence. Therefore supporting additional feature types are what we will work on in the near future.

It is not difficult to see that the feature generation step is the most critical step of all. When given a "correct" set of candidate features, one can easily build a high accuracy classifier from them. However, the task of finding the "correct" set of candidate features is hard. Therefore, Sirius PSB also provides an option to do auto-generation of features using genetic algorithm given a set of training datasets. We have also shown in this paper that high-quality prediction models can be built with features auto-generated by Sirius PSB genetic algorithm.

Feature Selection

Usually, during the feature generation step, a large set of candidate features is generated. This poses two immediate problems. First, with too many candidate features comes the curse of dimensionality. Second, many of these features are noise or irrelevant features. Having such features for training often leads to over-fitting by most machine learning techniques.

Therefore, the purpose of employing this feature selection step is to significantly reduce noise and irrelevant features. Various techniques may be used to carry out the differentiation between meaningful and useless features. Techniques like signal-to-noise measure, statistical measure, entropy measure, information gain measure, correlation-based measure can all be used for this cause. For this step, we imported all the feature selection techniques that are already implemented in WEKA [9].

Feature Integration

In the feature integration step, a machine learning method is chosen to be trained using features that remain after the feature selection step. A classifier is then ready to be used. Machine learning methods that are available in Sirius PSB are imported from WEKA [9].

For sequence prediction, there are generally three different categories. One is where prediction is made only once for each sequence (e.g. subcellular localization of proteins). Another is where there is a known anchor motif and prediction is only made when the anchor motif is encountered. The last type is where no anchor motif exists and every position of the sequence is a candidate site and prediction has to be made on every position of the sequence.

For the first two types of problems, simply using the first 3 steps of the approach is usually sufficient to build decent prediction models. For the last category, an additional step is often employed, that is, cascade classification.

Cascade Classification

The reason for an additional step is that problems tackled previously using the feature generation, feature selection and feature integration steps always had anchor points to base the predictions upon. For example, [5] does prediction of polyadenylation sites of humans. They first scan for an AATAAA motif in the sequence and then predict if that is a true polyadenylation site. However, using such a scanning method has a disadvantage: although 58.2% of human polyadenylation site contains an AATAAA upstream, some motifs other than AATAAA can also be a true polyadenylation site [11]. Therefore, the prediction model cannot recognize those sites.

Also, not all functional sites have an anchor motif. For example, the most frequently occurring motif (AATAAA) for Arabidopsis polyadenylation sites is found only in about 10% of Arabidopsis genes [12].

For sequences without an anchor motif, feature integration is first used to create an output with a prediction score for each position on the sequence. Thereafter, these prediction scores are used as a feature vector to train a cascade classifier using machine learning. Any machine learning technique that can handle numerical features can be used here.

Results and Discussion

After creating Sirius PSB, the next step is to prove that Sirius PSB is indeed capable of producing decent prediction systems for real-life applications. Therefore, we have built two prediction models using Sirius PSB. One is for the prediction of subcellular localization of proteins. The other one is for recognition of Arabidopsis polyadenylation sites. It took us two days to build the prediction model for subcellular localization of proteins and five days for the model for Arabidopsis polyadenylation sites.

Subcellular Localization of Proteins

Given a protein sequence, it is of interest to know the subcellular localization of the protein because it helps us better understand its functions. Many prediction models have been constructed previously to predict a protein's subcellular localization based on its sequence. In particular, TargetP [13] is one such model. It has achieved a high sensitivity (>85%) and is still often used by biologists today. Hence, we compare our protein localization model to TargetP.

Datasets of Subcellular Localization of Proteins

The dataset used here is downloaded from the TargetP website. All sequences were extracted from SWISS-PROT and redundancy reduced. Please refer to [13] for more details on the preparation of the dataset.

The dataset has two versions, plant and non-plant. For the plant version, it contains 141 cTP, 368 mTP, 269 SP and 162 "other" sequences. For the non-plant version, it contains 371 mTP, 715 SP and 1652 "other" sequences.

The abbreviations used subsequently are as follows: cTP stands for chloroplast transit peptides, mTP stands for mitochondrial targeting peptides, SP stands for signal peptides and "other" stands for peptides in other localizations.

Method of Subcellular Localization of Proteins

TargetP is built using neural networks and consists of two layers. The first layer is a dedicated network for each presequence (cTP, mTP, SP), and the second layer is an integrating network that outputs the actual prediction. A non-plant version of TargetP that differentiates only between mTP, SP and "other" has also been constructed.

For our protein localization model (PL model), we employed the first 3 steps of the approach — feature generation, feature selection and feature integration.

For the feature generation step, we used straightforward features of 1, 2 and 3-gram with window (0,100). Note that there are 20 different amino acids. This means there are $20 + 20^2 + 20^3 = 8420$ features. We then calculated the occurrence of the 8420 features for the first 101 characters of each sequence. For the feature selection step, we filtered away those with chi-square value ≤ 0.

As for the feature integration step, we used Support Vector Machine [14] with polynomial kernel of degree two and buildLogisticModels (set to True) for all the sequences except for non-plant SP presequence, where Naive Bayes algorithm was used instead. Like TargetP, we have seven different classifiers for each type of presequence. All of them used SVM except for the non-plant SP preseqeuence. Naive Bayes was used for non-plant SP presequence because poor performance was observed when SVM was used for this particular presequence. This shows the flexibility of Sirius PSB in that the choice of machine learning method can be changed without fuss.

Results of Subcellular Localization of Proteins

The results for TargetP were extracted from [13]. As the authors of TargetP used 5-fold cross-validation, we also ran 5-fold cross-validation on PL model in order to compare with TargetP on equal ground.

			TargetP			PL model		
Set	Category	Size	TP	FN	SN	TP	FN	SN
Plant	cTP	141	120	21	0.851	124	17	0.879
	mTP	368	300	68	0.815	303	65	0.823
	SP	269	245	15	0.911	250	19	0.929
	other	162	137	25	0.846	140	22	0.864
Plant Sensitivity			0.856 (0.853)			**0.874 (0.869)**		
Non-plant	mTP	371	330	41	0.889	337	34	0.908
	SP	715	683	32	0.955	622	93	0.870
	other	1652	1451	201	0.878	1610	42	0.975
Non-plant Sensitivity			0.907 (0.900)			**0.918 (0.938)**		
Overall Sensitivity			0.878 (0.888)			**0.893 (0.921)**		

Table 1. Prediction performance based on 5-fold cross-validation of TargetP and PL model. Plant senitivity, Non-Plant sensitivity and Overall sensitivity based on equal weightage of each category (based on absolute numbers of each category).

Discussion of Subcellular Localization of Proteins

From the results, it is clear that using the approach (feature generation, feature selection and feature integration) produces superior results in the prediction of subcellular localization of proteins compared to TargetP.

Finally, it is important to note that all this was done within a timeframe of just two days using Sirius PSB. It is remarkable to see that a high-quality prediction model could be developed in such a short time using Sirius PSB.

Recognition of Polyadenylation Sites from Arabidopsis Genomic Sequences

Polyadenylation is a post-transcriptional process, which basically cleaves and adds approximately 200-300 adenosine residues to the pre-mRNA 3' end. This process is an essential processing event and an integral part of gene expression [12]. Having the ability to accurately predict them allows us to define gene boundaries, predict the number of genes as well as better understand the process.

Currently, the best prediction model for recognition of polyadenylation site for Arabidopsis sequences is designed by us [7]. Here, we show that with Sirius PSB, we can build a better model in considerably lesser time. We call the new model Arabidopsis Polyadenylation Site model (APS model).

Datasets of Recognition of Polyadenylation Sites from Arabidopsis Genomic Sequences

The datasets used here were provided by Qingshun Quinn Li [15]. For any two sequences with more than 70% similarity using pair-wise global alignment, one is removed. After redundancy is reduced, the dataset contains 6209 sequences with EST-supported polyadenylation sites, 1501 coding region sequences, 864 5'UTR region sequences and 1581 intronic region sequences. Each sequence is of length 400 and for the EST-supported sequences; the polyadenylation site is at position 301.The dataset are split and used in the following ways:

Dataset A (Used for training Feature Integration Classifier)

2640 (+ve) sequences with EST-supported polyadenylation sites
900 (-ve) coding region sequences
476 (-ve) 5'UTR region sequences
954 (-ve) intronic region sequences

Dataset B (Used for training Cascade Classifier)

1500 (+ve) sequences with EST-supported polyadenylation sites
100 (-ve) coding region sequences
100 (-ve) 5'UTR region sequences
100 (-ve) intronic region sequences

Dataset C (Used for testing)

2069 (+ve) sequences with EST-supported polyadenylation sites
501 (-ve) coding region sequences
288 (-ve) 5'UTR region sequences
527 (-ve) intronic region sequences

Both models use the same number of sequences in the same way. This is possible because both Koh et al (2007) model [7] and APS model follow the same general approach.

Method of Recognition of Polyadenylation Sites from Arabidopsis Genomic Sequences

Feature generation, feature selection, feature integration and cascade classification is the methodology used by both Koh et al (2007) model [7] and APS model. The settings for feature selection (chi-square with threshold 0) and feature integration (support vector machine) for both models are the same. The only difference between the two models is in the feature generation step.

Koh et al (2007) model [7] generates 261 candidate features based on biological knowledge from literature. APS model uses 144 candidate features auto-generated by running genetic algorithm on training Dataset A.

Results of Recognition of Polyadenylation Sites from Arabidopsis Genomic Sequences

The performance measure used is equal-error-rate value (i.e. the points where sensitivity = specificity).

Sensitivity (SN) = TP/ (TP + FN), Specificity (SP) = TN / (TN + FP)

where TP (True Positive) is the total number of EST-supported polyadenylation sites that are correctly predicted. FN (False Negative) is the total number of EST-supported polyadenylation sites that are not identified. TN (True Negative) is the total number of sites with prediction score ≤ threshold in the (-ve) sequences. FP (False Positive) is the total number of sites with score > threshold in the (-ve) sequences.

SN_0 means that the predicted polyadenylation site is exactly the same as the EST-supported polyadenylation site. SN_10 means the EST-supported polyadenylation site is within 10 nucleotides of the predicted polyadenylation site. SN_30 means the EST-supported polyadenylation site is within 30 nucleotides of the predicted polyadenylation site.

Control Sequences		Koh et al (2007) model [7] Sensitivity & Specificity	APS model Sensitivity & Specificity
Coding	SN_0	0.943	**0.955**
	SN_10	0.965	**0.974**
	SN_30	0.975	**0.985**
5'UTR	SN_0	0.849	**0.857**
	SN_10	0.892	**0.901**
	SN_30	0.915	**0.931**
Intronic	SN_0	0.711	**0.746**
	SN_10	0.788	**0.829**
	SN_30	0.830	**0.872**

Table 2. Equal-error-rate of Koh et al (2007) model [7] and APS model.

Discussion of Recognition of Polyadenylation Sites from Arabidopsis Genomic Sequences

From the results, APS model has shown improved performance over Koh et al (2007) model [7]. What we would like to stress here is that even though both methods followed the same approach, but Koh et al (2007) model [7] was designed by writing many programs and having to change the codes in the programs whenever the authors wanted to try different settings or generate different features. In contrast, the APS model produced by Sirius PSB encompasses settings that can be changed instantly via a few mouse clicks in the Graphical User Interface of Sirius PSB.

Another important difference is that the 261 candidate features generated were decided upon after spending a lot of time and effort searching and reading literature about Arabidopsis polyadenylation process. Compare this with the APS model that auto-generated 144 features simply by running the genetic algorithm (provided by Sirius PSB) using training Dataset A.

Due to these differences, Koh et al (2007) model [7] took us about nine months to complete whereas APS model took us only five days.

Conclusion

In this paper, we have described a software tool named Sirius Prediction System Builder. Sirius PSB helps users build high-quality computer models using the feature generation, feature selection, feature integration and cascade classification approach in a manner that is hassle-free.

Having the easy-to-use graphical user interface, even a person without prior programming knowledge can build a prediction model. As demonstrated, we built two prediction models using Sirius PSB, and not only did those prediction models outperform current state-of-the-art models in terms of accuracy, but the time required to build them is also significantly reduced.

Furthermore, with the genetic algorithm integrated into Sirius PSB, a user will not even need to worry about what features to generate. As demonstrated, the genetic algorithm is able to generate "useful" features, where excellent prediction models can be subsequently built from them.

Although Sirius PSB is able to assist users in generating “useful” features using its implemented genetic algorithm, there also exist numerous motif finding programs easily available that could also greatly assist users in generating “useful” features. Therefore, for future enhancements of Sirius PSB's feature generation capabilities, we are considering incorporating outputs of popular motif finding programs like Pfam, PROSITE and MEME into Sirius PSB as features. This will not only boost results, but also makes it more convenient for the user.

With Sirius PSB, we are confident that more high-quality prediction models will be produced using the feature generation, feature selection, feature integration and cascade classification methodology.

Availability and Requirements

Project name: Sirius Prediction System Builder

Project homepage: http://www.comp.nus.edu.sg/~wongls/projects/dnafeatures/SiriusPSB-v2_1.zip

Operating system: Platform independent (Developed in Linux – Ubuntu 7.10)

Programming language: Java

Other requirements: None

License: Access to the software is open to all academics.

Any restrictions to use by non-academics: Commercial use license can be obtained by contacting the authors.

Authors' contributions

Chuan Hock carried out the programming, software design and drafted the manuscript. Sharene prepared all user manuals for the software and helped to draft the manuscript. Limsoon played a supervision role for the study. All authors read and approved the final manuscript.

Acknowledgements

This work is supported in part by Singapore National Research Foundation grant NRF-G-CRP-2997-04-082(d) (Wong) and a National University of Singapore NGS scholarship (Koh).

References

1. Shendure J, Mitra RD, Varma C, Church GM: **Advanced Sequencing Technologies: Methods and Goals.** Nature Reviews Genetics 2004, **5**:335-344.
2. Chi KR: **The year of sequencing.** Nature Methods 2008, **5**:11-14.
3. Aebersold R, Mann M: **Mass spectrometry-based proteomics.** Nature 2003, **422**:198-207
4. Liu H, Han H, Li J, Wong L: **DNAFSMiner: A Web-Based Software Toolbox to Recognize Two Types of Functional Sites in DNA Sequences.** Bioinformatics 2005, **21**:671-673.
5. Liu H, Han H, Li J, Wong L: **An In-Silico Method for Prediction of Polyadenylation Signals in Human Sequences.** In *Proceedings of 14th International Conference on Genome Informatics: December 2003; Yokohama;* 2003:84-93.
6. Liu H, Wong L: **Data Mining Tools for Biological Sequences.** Journal of Bioinformatics and Computational Biology 2003, **1**:139-167.
7. Koh CH, Wong L: **Recognition of Polyadenylation Sites from Arabidopsis Genomic Sequences.** In *Proceedings of 18th International Conference on Genome Informatics: 3-5 December 2007; Singapore;* 2007:73-82.
8. Liu H, Han H, Li J, Wong L: **Using Amino Acid Patterns to Accurately Predict Translation Initiation Sites.** In silico Biology 2004, **4**:255-269.
9. Witten, I.H. and Frank, E: *Data Mining: Practical Machine Learning Tools and Techniques, Second Edition.* Morgan Kaufmann: San Francisco; 2005.
10. Bel M V, Saeys Y, Peer Y V: **FunSiP: a modular and extensible classifier for the prediction of functional sites in DNA**. Bioinformatics 2008, **24**:1532-1533.
11. Beaudoing E, Freier S, Wyatt JR, Claverie DG, Gautheret D: **Patterns of Variant Polyadenylation Signal Usage in Human Genes.** Genome Research 2000, **10**:1001-1010.
12. Loke CJ, Stahlberg EA, Strenski DG, Haas BJ, Wood PC, Li QQ: **Compilation of mRNA Polyadenylation Signals in Arabidopsis Revealed a New Signal Element and Potential Secondary Structures.** Plant Physiology 2005, **138**:1457-1468.
13. Emanuelsson O, Nielsen H, Brunak S, Heijne GV: **Predicting Subcellular Localization of Proteins Based on Their N-Terminal Amino Acid Sequence.** Journal of Molecular Biology 2000, **300**:1005-1016.
14. Cortes C, Vapnik V: **Support-Vector Networks.** Machine Learning 1995, **20**:273-297.
15. Ji G, Zheng J, Shen Y, Wu X, Jiang R, Lin Y, Loke JC, Davis KM, Reese GJ, Li QQ: **Predictive Modeling of Plant Messenger RNA Polyadenylation Sites.** BMC Bioinformatics 2007, Vol.8, No.43, February 2007.
16. Prlic A, Domingues FS, Sippl MJ: **Structure-Derived Substitution Matrices for Alignment of Distantly Related Sequences.** Protein Engineering 2000, **13**:545-550.
17. **TargetP 1.1 Server** [http://www.cbs.dtu.dk/services/TargetP/]
18. **Koh et al (2007) Model [7] Datasets & Source codes** [http://www.comp.nus.edu.sg/~wongls/projects/dnafeatures/giw07-supplement/]
19. **Pfam** [http://pfam.janelia.org/]
20. **PROSITE** [http://expasy.org/prosite/]
21. **MEME** [http://meme.sdsc.edu/meme/intro.html]