

Machine Learning Approaches for Prediction of Expansin Gene Family in *Indica* Rice

N. Hemalatha · M. K. Rajesh · N. K. Narayanan

Received: 7 September 2012 / Accepted: 20 September 2013 / Published online: 15 October 2013
© NAAS (National Academy of Agricultural Sciences) 2013

Abstract Expansin refers to a family of proteins present in the plant cell wall which has important roles in plant cell growth, emergence of root hairs, meristem function and other developmental processes. A major constraint to rice production is submergence of rice by flash flooding. In our earlier study, we had identified 21 novel sequences related to expansin gene families in the genome of *indica* rice using genome-wide analysis. Development of a tool for the prediction of these expansin genes using computational approaches might significantly enhance rice gene annotation. ExpansinPred, a novel computational method based on radial basis function (RBF) and support vector machines (SVMs) for prediction of α -expansins (EXPA) and β -expansins (EXPB), is presented in this work. Two large families of expansin genes have been discovered in plants, namely EXPA and EXPB. The experimental data are curated from NCBI and include 24 EXPA and 20 EXPB, of *indica* rice, after redundancy elimination. The proper window length for a potential expansin was optimized as 4 for EXPA and EXPB with prediction accuracies 100 % each for both classifiers for RBF classifier. For SVM, the window length was optimized as 3 for EXPA and 4 for EXPB with prediction accuracies 90 and 100 %, respectively. To evaluate the prediction performance of ExpansinPred, cross-validation, independent dataset validation and jackknife validation were carried out. ExpansinPred was also compared with four more algorithms namely Naïve Bayes, sequential minimal optimization, J48 and random forest. To further prove that species-specific predictor is much better than general tool, ExpansinPred was compared with an All-plant tool and also with plants other than rice as test set. The different statistical analyses carried out demonstrated that the proposed algorithm is a useful computational tool for rice genome annotation, specifically for predicting expansin gene family, and can benefit rice research community.

Keywords Expansin · J48 · SMO · Naïve Bayes · Rice

Introduction

Flooding is one of the most important environmental stresses worldwide and has a dramatic effect on the growth

and yield of various crops. When subjected to submergence, one of the biggest problems encountered by plants is energy deficit caused by inhibition of respiration as the oxygen supply is severely hampered. Among the 42 biotic and abiotic stresses affecting rice production, submergence has been identified as the third most important constraint for higher rice productivity in India, because it sometimes results in near total yield loss [33].

Deep-water rice responds to submergence with rapid stem elongation, which is preceded by large increase in expansin mRNA [8]. Expansins are proteins that break the crosslinks (hydrogen bonds) between cellulose microfibrils and other cell wall constituents, loosening the cell wall fabric. These plant cell wall loosening proteins are known to be involved in a variety of

N. Hemalatha (✉)
AIMIT, St. Aloysius College, Mangalore, India
e-mail: hemasree71@gmail.com

M. K. Rajesh
Division of Crop Improvement, Central Plantation Crops
Research Institute, Kasaragod 671124, India

N. K. Narayanan
School of Information Science and Technology, Kannur
University, Kannur, India

other developmental processes in which cell wall enlargement and modification occur [13]. They characteristically cause cell wall stress relaxation and irreversible wall extension [24]. This process is essential for cell enlargement. They are typically 250–275 amino acids long and are made up of two domains preceded by a signal peptide. Four families of expansins are currently recognized in plants on the basis of phylogenetic sequence analysis [14]. They are designated α -expansin (EXPA), β -expansin (EXPB), expansin-like A (EXLA) and expansin-like B (EXLB). Among these, α -expansin and β -expansin proteins have been demonstrated experimentally to cause cell wall loosening [15], whereas EXLA and EXLB proteins are known only from their gene sequences.

Twenty-one novel sequences related to expansin families were identified in our early work on genome-wide analysis in *Oryza sativa* spp. *indica* [18]. Development of a genome-wide prediction tool for expansins will significantly help in the advancement of rice genome annotation. In the present study, we explored the performance of basic local alignment search tool (BLAST) for the annotation of expansin genes from *indica* rice, which produced less accurate results. Therefore, it was decided to explore machine learning algorithms to predict expansin genes. Here, we propose a novel gene prediction tool called ExpansinPred, for predicting EXPA and EXPB. The models for creating this tool were developed using the classifiers radial basis function (RBF) in WEKA and support vector machine in SVM^{light} [32]. Statistical accuracy and prediction of this tool were tested using cross-validation, independent data test and jackknife validation.

Methods

Dataset

Twenty-four EXPA and twenty EXPB, belonging to *indica* rice, from NCBI database, were used in this study. For conducting independent data test, around 10 % of the EXPA and EXPB were kept aside, which means three EXPA and four EXPB were randomly selected from the original set for the creation of test set and remaining were used for the training set.

Features

Initially, for all four nucleotides (A, T, C and G), different window lengths were generated [2]. In order to produce different features based on nucleotides, the frequency of occurrence of the four nucleotide bases was considered, and this method also created a standard window size among all the selected sequences. In the first method of feature extraction (3-mer), 64 feature vectors were

generated based on the occurrence of three nucleotide bases at a time in a sequence. Similarly to generate higher dimensional feature vectors, 256 features (4-mer), 1,024 features (5-mer) and 4,096 features (6-mer) were considered in the source dataset using a PERL program.

Support Vector Machine (SVM)

Support vector machine, a machine learning method, has been applied for many kinds of pattern recognition problems. The principle of the SVM is to transform the samples into a higher dimension space called Hilbert space. A separating hyperplane is sought in this space called the optimal separating hyperplane in such a way as to maximize its distance from the closest training samples. SVM is a supervised machine learning technology founded theoretically on statistical learning theory [31]. Recently, SVM has been successful in solving many biological problems, such as predicting protein subcellular locations [19] and protein secondary structures [16]. In this work, the feature vector of each expansin gene was transformed into a higher dimension space through linear kernel function.

The Machine Learning Algorithms

We have used WEKA machine learning workbench [20] to implement Naïve Bayes, sequential minimal optimization (SMO), RBF, random forest (RF) and J48 decision tree algorithms.

Naïve Bayes

Naïve Bayes is a classification model based on Bayes theorem [25]. This classifier assumes that the effect of a variable value on a given class is independent of the other variables values which are called class conditional independence. It is made to simplify the computation, and hence, the word “Naïve” is considered. Naïve Bayes is actually a special case of a Bayesian Network, in which each attribute node (corresponding to an attribute variable) has the class node (corresponding to the class variable) as its parent, but does not have any other parent. Given the class variable, Naïve Bayes does not represent any variable dependencies.

Sequential Minimal Optimization (SMO)

Sequential minimal optimization [26] is one of the most popular algorithms for large margin classification by SVM. It is a new SVM learning algorithm which is conceptually simple, easy to implement, often faster and has better scaling properties than a standard SVM algorithm. It is an algorithm for efficiently solving the optimization problem

which arises during the training of SVMs. It is a fast and an efficient version of SVM implemented in WEKA.

Radial Basis Function (RBF)

Radial basis function network is a kind of multilayer, feed-forward artificial neural network [3]. An RBF network has three layers, namely the input layer, the hidden layer and the output layer. The input layer in RBF network broadcasts the coordinates of the input vector to each of the nodes in the hidden layer. Based on the associated RBF, each node in the hidden layer produces activation. Finally, each node in the output layer computes a linear combination of the activations of the hidden nodes. The activation functions associated with the hidden nodes and the weights associated with the links between the hidden layer and the output layer determine how an RBF network reacts to a given input stimulus. In our model, once feature vectors were fed into input layers, then the links between nodes are iteratively updated until convergence. The output layer finally produced the decision of “expansin” or “non-expansin.”

Random Forest (RF)

Random forest has been applied effectively in various biological problems [5, 6, 21, 23, 29, 30]. It is an ensemble method generating many trees using recursive partitioning and then aggregating the results [4]. Using a bootstrap sample of the training data, each tree is independently constructed. For each tree, two-third of the training samples are used for constructing the tree and the remaining one-third of the samples are used to test the tree. These left out data, named “Out of Bag,” is used to attune the performance of each tree. The generalization error of a forest of tree classifiers depends on the strength of the individual trees present in the forest and the correlation between them.

J48 Decision Tree

A J48 decision tree is a predictive machine learning model that decides the target value (dependent variable) of a new sample based on various attribute values of the available data. The internal nodes of a decision tree denote the different attributes, and the branches between the nodes tell the possible values that these attributes can have in the observed samples, while the terminal nodes tell the final value (classification) of the dependent variable. The attribute that is to be predicted is known as the dependent variable, since its value depends upon, or is decided by, the values of all the other attributes. The other attributes, which help in predicting the value of the dependent variable, are known as the independent variables in the dataset.

Similarity Search

For sequence similarity searches, we have used BLAST which compares a set of given data against a database of sequences [1]. This tool conducts similarity search for predicting the function of a new sequence against a database of annotated sequences. Here, we conducted a tenfold cross-validation for predicting α -expansin and β -expansin using BLAST, and result of the same is analyzed in the results section.

Performance Evaluation

In statistical prediction, three methods often used to examine a predictor for its effectiveness are independent dataset test, cross-validation test and jackknife test [22]. In the independent dataset test, although none of the data to be tested occurs in the training dataset used to train the predictor, the selection of data for the testing dataset could be quite arbitrary unless it is sufficiently large. For the cross-validation test, the practical procedure often used in literature is the fivefold, eightfold or tenfold cross-validation. In k-fold cross-validation, data are divided into k subsets of (approximately) equal size. The net is trained k times, each time leaving out one of the subsets from training, but using only the omitted subset to compute whatever error criterion of interest. If k equals the sample size, this is called “leave-one-out” cross-validation. The problem with the cross-validation examination is that the number of possible selections in dividing a target dataset is an astronomical figure even for a very simple dataset [17, 28]. Therefore, any result by the cross-validation test only represents one of the many possible results and hence cannot avoid the uncertainty either.

Leave-one-out cross-validation is easily confused with jackknifing. Both involve omitting each training case in turn and retraining the network on the remaining subset. But cross-validation is used to estimate generalization error, while jackknifing is used to estimate the bias of a statistic. In the jackknife cross-validation, each of the data in the benchmark dataset is in turn singled out as a tested one and the predictor is trained by the remaining ones. During the jackknifing process, both the training dataset and testing dataset are actually open, and a data will in turn move from one to the other. The jackknife cross-validation can exclude the memory effects during entire testing stage, and also the outcome thus obtained is always unique for a given benchmark dataset. Jackknifing is usually considered as the most rigorous and objective test among the other tests [9]. In the jackknife, some statistic of interest is computed in each subset of the data. The average of these subset statistics is compared with the corresponding statistic computed from the entire sample in order to estimate

Table 1 Comparison of prediction performances of different algorithms for α - and β -expansins using various cross-validation techniques

Algorithm	Window length	Threefold cross-validation				Fivefold cross-validation				Eightfold cross-validation				LOO cross-validation				Average MCC
		Sensitivity (%)	Specificity (%)	Accuracy (%)	MCC (%)	Sensitivity (%)	Specificity (%)	Accuracy (%)	MCC (%)	Sensitivity (%)	Specificity (%)	Accuracy (%)	MCC (%)	Sensitivity (%)	Specificity (%)	Accuracy (%)	MCC (%)	
α -Expansins																		
RBF	3	100	100	100	1	100	100	100	1	100	100	100	1	100	100	100	1	1
	4	100	100	100	1	100	100	100	1	100	100	100	1	100	100	100	1	1
Naive Bayes	5	100	0	40	0	100	0	40	0	100	0	40	0	100	0	40	0	0
	6	100	0	40	0	100	0	40	0	100	0	40	0	100	0	40	0	0
	3	25	100	70	0.4	0	1	0	0	25	100	70	0.4	0.25	100	0.7	0.48	0.32
	4	50	100	80	0.61	50	100	80	0.61	50	100	80	0.61	0.5	100	0.8	0.62	0.61
J48	5	100	100	100	1	100	100	100	1	100	100	100	1	100	100	100	1	1
	6	100	100	100	1	100	100	100	1	100	100	100	1	100	100	100	1	1
	3	75	100	90	0.8	75	100	90	0.8	75	100	90	0.8	0.75	100	0.9	0.8	0.8
	4	50	100	80	0.61	50	100	80	0.61	50	100	80	0.61	0.5	100	0.8	0.62	0.61
	5	50	50	50	0	50	50	50	0	50	50	50	0	0.5	0.5	0.5	0	0
	6	25	100	70	0.4	20	100	70	0.4	20	100	70	0.4	0.25	100	0.7	0.48	0.4
β -Expansins																		
RBF	3	100	100	100	1	100	100	100	1	100	100	100	1	100	100	100	1	1
	4	100	100	100	1	100	100	100	1	100	100	100	1	100	100	100	1	1
Naive Bayes	5	25	100	70	0.4	25	100	70	0.4	25	100	70	0.4	100	0.6	0.8	0.67	0.46
	6	100	0	40	0	100	0	40	0	100	0	40	0	100	0	40	0	0
	3	0	100	60	0	0	100	60	0	0	100	60	0	100	0	0.4	0	0
	4	0	100	60	0	0	100	60	0	0	100	60	0	100	100	100	1	0.25
J48	5	100	100	100	1	100	100	100	1	100	100	100	1	100	100	100	1	1
	6	100	50	70	0.5	100	50	70	0.5	100	50	70	0.5	100	100	100	1	0.63
	3	100	100	100	1	100	100	100	1	100	100	100	1	100	100	100	1	1
	4	75	100	90	0.8	75	100	90	0.8	75	100	90	0.8	100	100	100	1	0.85
	5	100	100	100	1	100	100	100	1	100	100	100	1	100	100	100	1	1
	6	75	100	90	0.8	75	100	90	0.8	75	100	90	0.8	100	100	100	1	0.85

the bias of the latter. Therefore, of the above three examination methods, the jackknife test has been widely recognized [27, 29] and used by investigators to examine the accuracy of various predictors [7, 8, 10–12]. On applying jackknifing for the dataset with regard to EXPA and EXPB, it can be concluded from the tables (Tables 1, 2, 3) that data used for validation are unbiased.

Performance Evaluation

To compare the gene prediction accuracy, the standard performance measures on prediction by Burset and Guigo [33] were applied. The measures sensitivity (S_n), specificity (S_p), accuracy (Acc), precision (Pr) and Mathew correlation coefficient (MCC) were applied. Sensitivity and specificity relate to the test's ability to identify positive results and negative results. Accuracy (Ac) is the proportion of the total number of predictions that were predicted

correct, and precision (Pr) is the proportion of the positive cases that were predicted correct. However, when the number of positive results and negative results differs too much from each other, the MCC should be included to evaluate the prediction performance. The value of MCC ranges from -1 to 1 , and a positive MCC value stands for better prediction performance.

Among the data with positive prediction by ExpansinPred, the real positives are defined as true positives, (TP) (the number of correctly predicted genes), while the others are defined as false positives, (FP) (the number of pseudogenes (non-expansins) wrongly predicted). Among the data with negative hits by ExpansinPred, the real positives are defined as false negatives, (FN) (the number of true genes missed in the prediction), while the others are defined as true negatives, TN (the number of correctly predicted pseudogenes). The performance measurements of sensitivity (S_n), specificity (S_p), accuracy (Acc), precision (Pr) and MCC are given below in Eqs. 1–5:

Table 2 Comparison of the prediction performances of different algorithms for α - and β -expansins using independent dataset test and jackknife validations

Algorithm	Window length	Independent data test validation					Jackknife validation				
		Sensitivity (%)	Specificity (%)	Accuracy (%)	Precision (%)	MCC	Sensitivity (%)	Specificity (%)	Accuracy (%)	Precision (%)	MCC
α -Expansins											
RBF	3	100	100	100	100	1	97	100	98	100	0.98
	4	100	100	100	100	1	100	100	100	100	1
	5	100	0	40	40	0	100	100	100	100	1
	6	100	0	40	40	0	100	100	100	100	1
Naïve Bayes	3	25	100	70	100	0.4	100	100	100	100	1
	4	50	100	80	100	0.61	100	100	99	100	0.86
	5	100	100	100	100	1	100	100	100	100	1
	6	100	100	100	100	1	100	100	100	100	1
J48	3	75	100	90	100	0.8	100	100	100	100	1
	4	50	100	80	100	0.61	96	88	97	77	0.94
	5	50	50	50	40	0	96	91	93	92	0.87
	6	25	100	70	100	0.4	87	96	98	86	0.97
β -Expansins											
RBF	3	100	100	100	100	1	100	100	100	100	1
	4	100	100	100	100	1	100	100	100	100	1
	5	25	100	70	100	0.4	98	93	99	93	0.98
	6	100	0	40	40	0	100	0	100	100	0
Naïve Bayes	3	0	100	60	0	0	100	100	100	100	1
	4	0	100	60	0	0	99	100	90	0	0.59
	5	100	100	100	100	1	92	100	99	100	0.8
	6	100	50	70	57	0.5	100	98	99	100	0.99
J48	3	100	100	100	100	1	100	100	100	100	1
	4	75	100	90	100	0.8	95	100	98	100	0.98
	5	100	100	100	100	1	85	96	92	89	0.83
	6	75	100	90	100	0.8	92	100	89	62	0.7

Table 3 Classification accuracies of three kernels using SVM (for α - and β -expansin prediction models)

Kernel	Window length	Independent data test validation					Jackknife validation				
		Sensitivity (%)	Specificity (%)	Accuracy (%)	Precision (%)	MCC	Sensitivity (%)	Specificity (%)	Accuracy (%)	Precision (%)	MCC
α-Expansins											
Linear	3	75	100	90	100	0.8	93	100	97	100	0.95
	4	50	100	80	100	0.61	100	1	100	100	1
	5	25	100	70	100	0.48	100	100	100	100	1
	6	100	0	40	40	0	100	100	100	100	1
Polynomial	3	25	100	70	100	0.48	53	100	78	100	0.68
	4	50	100	80	100	0.61	100	1	100	100	1
	5	25	100	70	100	0.48	100	100	100	100	1
	6	0	100	60	0	0	100	100	100	100	1
RBF	3	100	0	40	40	0	69	0	40	53	0.24
	4	100	0	40	40	0	100	100	100	100	1
	5	100	0	40	40	0	100	100	100	100	1
	6	100	0	40	40	0	100	100	100	100	1
β-Expansins											
Linear	3	75	100	90	100	0.8	100	100	100	100	1
	4	100	100	100	100	1	98	100	99	100	0.98
	5	100	100	100	100	1	93	100	97	93	0.93
	6	50	100	80	100	0.61	100	100	100	100	1
Polynomial	3	75	100	90	100	0.8	88	100	95	100	0.91
	4	0	100	60	100	0	100	100	100	100	1
	5	0	100	60	100	0	100	100	100	100	1
	6	0	100	60	0	0	100	100	100	100	1
RBF	3	100	0	40	40	0	100	100	100	100	1
	4	100	0	40	40	0	100	100	100	100	1
	5	100	0	40	40	0	100	100	100	100	1
	6	100	0	40	40	0	100	100	100	100	1

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \times 100 \quad (1)$$

$$\text{Specificity} = \frac{\text{TN}}{\text{FP} + \text{TN}} \times 100 \quad (2)$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \times 100 \quad (3)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \times 100 \quad (4)$$

$$\text{MCC} = \frac{(\text{TP} \times \text{TN}) - (\text{FP} \times \text{FN})}{\sqrt{(\text{TP} + \text{FP})(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TN} + \text{FN})}} \quad (5)$$

ROC Curves

The ROC curve is a fundamental tool used for diagnostic test evaluation. In a ROC curve, the true positive rate (sensitivity) is plotted in function of the false positive rate

(1-specificity) for different cutoff points of a parameter. Each point on the ROC curve represents a sensitivity/specificity pair corresponding to a particular decision threshold. The area under the curve (AUC) is a measure of how well a parameter can distinguish between two diagnostic groups (abnormal/normal).

Results and Discussion

The prediction performances of ExpansinPred were evaluated by independent data test, cross-validation test and leave-one-out cross-validation test, and the biasedness of the dataset was further validated by jackknife validation. The various statistical results of ExpansinPred conducted for EXPA and EXPB using WEKA and SVM are reported in tables (Tables 1, 2, 3).

Performance of ExpansinPred in WEKA

We performed threefold, fivefold, eightfold and leave-one-out cross-validations along with independent data test with varying window length of 3–6 to evaluate the performance of ExpansinPred. For EXPA, in the case of various cross-validation tests and independent data tests, ExpansinPred achieved positive average MCC value for all the classifiers but best values were obtained for RBF network model with a window length of 4 and prediction accuracy of 100 % (Tables 1, 2). Hence, window length four can be considered optimal for EXPA which has good prediction accuracy.

Similarly for EXPB, threefold, fivefold, eightfold and leave-one-out cross-validations along with independent data test with varying window length of 3–6 were performed. Experimental results show that ExpansinPred achieved positive average MCC value for all window lengths with all machine learning algorithms for both cross-validation test and independent data test. Analyzing the results, we have taken window length 3 or 4 as optimal for EXPB with RBF network as the classifying algorithm and prediction accuracy of 100 % (Tables 1, 2).

Performance of ExpansinPred in SVM^{light}

We carried out independent data test using SVM with window length varying from 3 to 6. Test was carried out with linear, polynomial and RBF kernels. For each kernel, different parameters were tested and one with the best result was reported. In the case of EXPA, linear kernel was found to have the best prediction accuracy of 90 % with window length of 3 and for β -expansins linear kernel with 4 or 5 window length had best prediction accuracy of 100 % (Table 3). Hence, from the table, it can be concluded that linear kernel has the best average value for all

statistical measures for both expansins with window length 3 for α -expansins and 4 or 5 for EXPB.

Prediction Using BLAST and ExpansinPred

BLAST allows comparing our sequence against a database of sequences and informs if our sequence matches any of the sequences in the database [1]. In this study, we have conducted a tenfold cross-validation of EXPA and EXPB and compared the performance of BLAST against ExpansinPred which is reported in Table 4. Each sequence in the testing dataset was used as a BLAST query sequence, and remaining sequences (training dataset) were used as a BLAST database. It was observed from the table that α -expansins obtained an accuracy of 88 % for BLAST which was less when compared to ExpansinPred as the number of hits obtained was low. Similarly for EXPB, BLAST obtained an accuracy of only 86 % compared to ExpansinPred. This implies that similarity-based search method may not be the best method for the annotation of expansin genes.

Comparison with All-Plant Model

To prove the advantage of a species-specific predictor, an All-plant model was built and compared with ExpansinPred. In the case of EXPA, to build an All-plant model sequences from four plants viz. maize (*Zea mays*), Arabidopsis (*Arabidopsis thaliana*), castor (*Ricinus communis*) and tomato (*Solanum lycopersicum*) were selected. These were trained using the RBF model and SVM linear model with window size 4 for EXPA. All-plant dataset for EXPA contained 57 sequences. For EXPB, a total of 31 sequences from three plants namely maize, Arabidopsis and tomato were chosen for building an All-plant model. Here, the plant model was trained using RBF model and SVM linear

Table 4 Prediction result of α - and β -expansins with similarity search tool BLAST (tenfold cross-validation used)

Test	α -Expansins			β -Expansins		
	No. of sequences given	Correctly predicted	Accuracy (%)	No. of sequences given	Correctly predicted	Accuracy (%)
1	10	9	90	8	6	75
2	10	9	90	8	7	87.5
3	10	10	100	8	6	75
4	10	9	90	8	7	87.5
5	10	7	70	8	8	100
6	10	8	80	8	7	87.5
7	10	9	90	8	6	75
8	10	8	80	8	7	87.5
9	12	11	91	8	7	87.5
10	12	12	100	9	9	100
			88.1			86

model with window size 5 which was found to be the best model for EXPB.

On comparison of ExpansinPred with corresponding All-plant method based on rice independent training set, it was reported for EXPA that ExpansinPred was 50 % more accurate than All-plant method in the case of RBF model and 30 % more accurate than SVM linear model (Table 5). Taking the case of EXPB, ExpansinPred was found to be more accurate than All-plant method with an increase in accuracy of 40 % (Table 5). Similarly for SVM too, ExpansinPred performed better than All-plant method with accuracy of 100 % and MCC value of 1. Here, we have taken enough datasets from different plant species, where they were trained with the best classifier for EXPA and EXPB, respectively. Methodology applied for creating and validating the model was also identical. All these results clearly revealed that a model specific to a species is always more advantageous than a general one.

Performance of ExpansinPred with Other Plants

To further prove that species-specific predictor are much accurate than general predictor, we cross-checked the performance of ExpansinPred with plants individually. For this, ExpansinPred with RBF and SVM model was cross-checked with four plants viz. maize, Arabidopsis, castor and tomato for EXPA. Similarly for EXPB, ExpansinPred with RBF and SVM model was cross-checked with three plants viz. maize, Arabidopsis and tomato. The results obtained for both the models in the case of EXPA and EXPB (independent data test) for different plants are tabulated in the Table 6. The table reveals low accuracy and MCC value for each plant of EXPA and EXPB as compared to ExpansinPred with *indica* rice. The difference in the performance of ExpansinPred with *Oryza sativa* and other plants again indicates that species-specific predictor is always superior to general prediction systems.

Table 5 Comparison of All-plant model with ExpansinPred model

Method	Algorithm	Sn (%)	Sp (%)	Acc (%)	Prec (%)	MCC
All-Plant(WEKA)						
α -Expansins	RBF	0	100	50	0	0.00
β -Expansins		0	90	60	0	-0.19
ExpansinPred (WEKA)						
α -Expansins	RBF	100	100	100	100	1.00
β -Expansins		100	100	100	100	1.00
All-Plant (SVM)						
α -Expansins	Linear kernel	75	50	60	50	0.25
β -Expansins		100	93.33	95	83	0.88
ExpansinPred (SVM)						
α -Expansins	Linear kernel	75	100	90	100	0.8
β -Expansins		100	100	100	100	1.00

Table 6 Performance of ExpansinPred (α - and β -expansins) on other plants

Plants	SVM(linear)			RBF		
	Sn	Acc	MCC	Sn	Acc	MCC
α -Expansin						
Arabidopsis	11.8	33	-0.23	73.3	64	0.24
Maize	20	47	-0.21	71.4	71	0.41
Castor	11.8	33	-0.23	52.9	59	0.22
Tomato	0	41	-0.42	100	82	0.70
β -Expansin						
Arabidopsis	0	63	0	0	56	-0.20
Maize	21.4	54	0.32	78.6	83	0.68
Tomato	0	48	0	0	43	-0.23

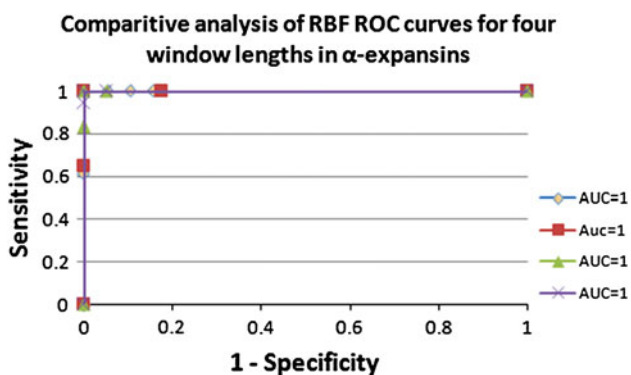


Fig. 1 ROC curves for the RBF network algorithm with respect to EXPA. This curve is drawn for the best algorithm with four window lengths

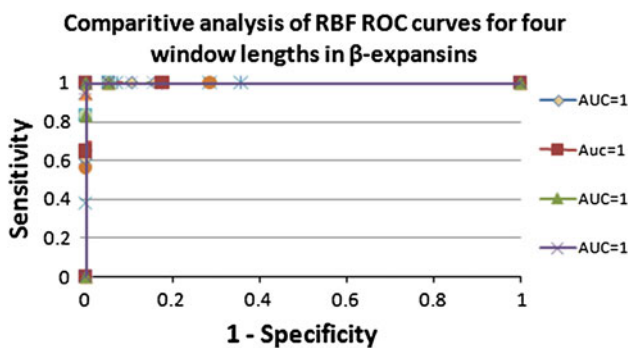


Fig. 2 ROC curves for the RBF network algorithm with respect to EXPB. This curve is drawn for the best algorithm with four window lengths

ROC curves

To evaluate the classifier stringently, we further plotted the ROC curves with AUC based on independent test performance of ExpansinPred with respect to EXPA and EXPB. The prediction performance of RBF network algorithm for EXPA and EXPB with different window lengths is represented in the figures (Figs. 1, 2). The ROC curve for the perfect classifier will produce a curve along the left and top boundary of the square and will receive a score of one.

Description of Web Server

ExpansinPred, a dynamic web server, has been implemented using the best performing module of SVM and WEKA and can be accessed at <http://14.139.158.118/expansin>. The common gateway interface (CGI) scripts of the web server have been written in PERL and PHP, and web interface to access user requests has been designed using HTML. The web page has the option for the user to enter the queries in the FASTA format either by pasting the query or by uploading the sequence in the form of a file. The output will be obtained in a user-friendly format. The overall description of the methodology applied in constructing ExpansinPred has been represented in the Fig. 3.

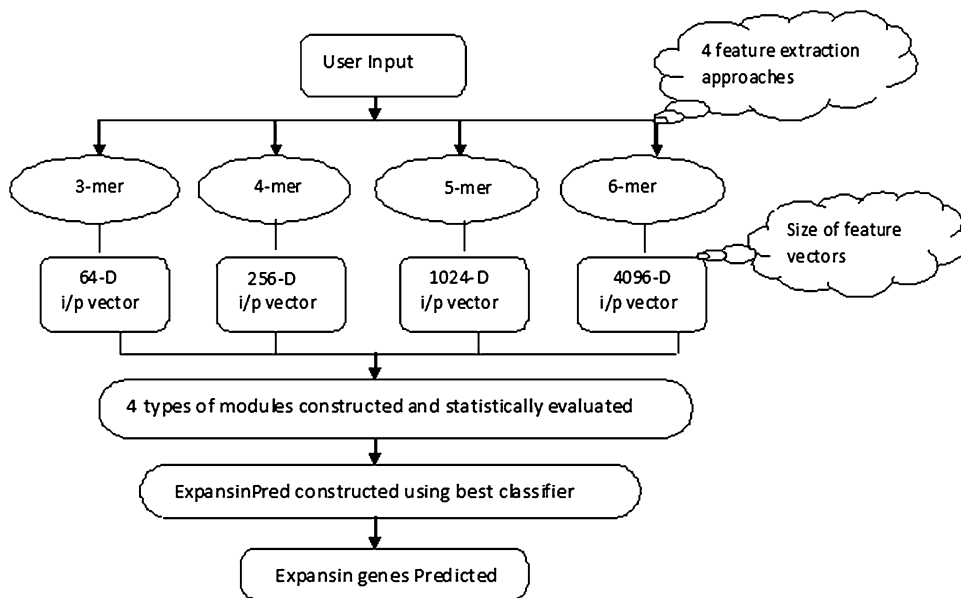


Fig. 3 Schematic diagram showing the architecture of web-based tool. The figure shows the flow diagram of the architecture of the ExpansinPred tool

Conclusions

Tools and resources are being developed to maximally construe the rice genome sequence. A major difficulty with rice annotation is the lack of accurate gene prediction programs. Similar to all completed genomes, rice has a substantial number of genes that are hypothetical in that they are predicted solely on the basis of gene prediction programs making it vital that the quality of gene prediction programs for rice be improved further.

Identification of expansin genes from sequence databases is difficult due to poor sequence similarity. In this work, we present a new method for expansin prediction based on RBF network for EXPA and EXPB implemented in WEKA and linear kernel in SVM for both EXPA and EXPB implemented in SVM^{light}. The performance was found to be highly satisfactory. Comparisons between different machine learning algorithms (Naive Bayes, RBF network, Random forest, SMO and J48) were also carried out. Very high prediction accuracies for the three validation tests show that ExpansinPred is a potentially useful tool for the prediction of EXPA and EXPB.

References

- Altschul S, Gish W, Miller W, Myers E, Lipman D (1990) Basic local alignment search tool. *J Mol Biol* 215(3):403–410
- Anwar F, Baker SM, Jabid T, Mehedi Hasan M, Shoyab M, Khan H, Walshe R (2008) Pol II promoter prediction using characteristic 4-mer motifs: a machine learning approach. *BMC Bioinformatics* 9:414
- Bishop CM (1995) *Neural networks for pattern recognition*. Oxford University Press, Oxford
- Breiman L (2001) Random forests. *Mach Learn* 45:5–32
- Burset M, Guigo R (1996) Evaluation of gene structure prediction programs. *Genomics* 34:353–367
- Chen C, Chen LX, Zou XY, Cai PX (2008) Predicting protein structural class based on multi-features fusion. *J Theor Biol* 253:388–392
- Chen C, Chen L, Zou X, Cai P (2009) Prediction of protein secondary structure content by using the concept of Chou's pseudo amino acid composition and support vector machine. *Protein Peptide Lett* 16:27–31
- Cho HT, Kende H (1997) Expansins in deepwater rice internodes. *Am Soc Plant Biol* 113:1137–1143
- Chou KC, Zhang CT (1995) Prediction of protein structural classes. *Crit Rev Biochem Mol Biol* 30:275–349
- Chou KC, Shen HB (2008) Cell-PLoc a package of web servers for predicting sub-cellular localization of proteins in various organisms. *Nat Protocols* 3:153–162
- Chou KC, Shen HB (2010) A new method for predicting the subcellular localization of eukaryotic proteins with both single and multiple sites. *PLoS One* 5:e9931
- Chou KC, Shen HB (2010) Plant-mPLoc a top-down strategy to augment the power for predicting plant protein subcellular localization. *PLoS One* 5:e11335
- Cosgrove DJ (2000) Loosening of plant cell walls by expansins. *Nature* 407:321–326
- Cho D, Cho HT, Lee Y (2006) Expansins: expanding importance in plant growth and development. *Physiol Plant* 126:511–518
- Cosgrove DJ, Li LC, Cho HT, Benning SH, Moore RC, Blecker D (2002) The growing world of expansins. *Plant Cell Physiol* 43:1436–1444
- Guo J, Chen H, Sun Z, Lin Y (2004) A novel method for protein secondary structure prediction using dual-layer SVM and profiles. *Proteins* 54:738–743
- Geisser S (1975) The predictive sample reuse method with applications. *J Amer Statist Assoc* 70:320–328
- Hemalatha N, Rajesh MK, Narayanan NK (2011) Genome-wide analysis and identification of genes related to expansion gene family in indica rice. *Int J Bioinformatics Res Appl* 7:162–167
- Hua S, Sun Z (2001) A novel method of protein secondary structure prediction with high segment overlap measure: support vector machine approach. *J Mol Biol* 308:397–407
- Ian H Witten, Eibe Frank, Mark A Hall (2011) *Data Mining: Practical machine learning tools and techniques*. In: Kaufmann M, Burlington MA (eds) Third Edition
- Kandaswamy KK, Pugalenth G, Hartmann E, Kalies KU, Moller S, Suganthan PN, Martinetz T (2010) SPRED: a machine learning approach for the identification of classical and non-classical secretory proteins in mammalian genomes. *Biochem Biophys Res Commun* 391:1306–1311
- Kumar KK, Pugalenth G, Suganthan PN (2009) DNA-ProT: identification of DNA binding proteins from protein sequence information using random forest. *J Biomol Struct Dyn* 26:679–686
- Lee JW, Lee JB, Park M, Song SH (2005) An extensive comparison of recent classification tools applied to microarray data. *Comp Stat Data Anal* 48:869–885
- McQueen-Mason SJ, Cosgrove DJ (1995) Expansin mode of action on cell walls- Analysis of wall hydrolysis, stress relaxation, and binding. *Plant Physiol* 107:87–100
- Peng F, Schuurmans D, Wang S (2004) Augmenting naive bayes classifiers with statistical language models. *Inf Retrieval* 7:317–345
- Platt JC (1999) Fast training of support vector machines using sequential minimal optimization. In: Schölkopf B, Burges C, Smola A (eds) *Advances in Kernel Methods—Support vector learning*. MIT Press, Cambridge, pp 185–208
- Quenouille M (1949) Approximate tests of correlation in time series. *J Roy Statist Soc Ser B* 11:18–84
- Stone M (1974) Cross-validatory choice and assessment of statistical predictions. *J Roy Statist Soc Ser B* 36:111–147
- Tukey JW (1958) Bias and confidence in not quite large samples. *Ann Math Stat* 29:614
- Uriarte RD, Andres SA (2006) Gene selection and classification of microarray data using random forest. *BMC Bioinformatics* 7:3–16
- Wu B, Abbott T, Fishman D, McMurray W, Mor G, Stone K, Ward D, Williams K, Zhao H (2003) Comparison of statistical methods for classification of ovarian cancer using a proteomics dataset. *Bioinformatics* 19:1636–1643
- Vapnik VN (1995) *The Nature of Statistical Learning Theory*. Springer Verlag, New York
- Widawsky DA, O'Toole JC (1990) Prioritizing the rice biotechnology research agenda for Eastern India. *The Rockefeller Foundation*, New York, pp 39–59