BIOINFORMATICS ORIGINAL PAPER Vol. 21 no. 8 2005, pages 1383–1388
doi:10.1093/bioinformatics/bti200

Genome analysis

Support vector machines for separation of mixed plant–pathogen EST collections based on codon usage

Caroline C. Friedel[1,2,†], Katharina H. V. Jahn[1,2,†], Selina Sommer[1,2,†], Stephen Rudd[3], Hans W. Mewes[4,5] and Igor V. Tetko[4,*]

[1]Institut fuer Informatik, Ludwig-Maximilians-Universitaet Muenchen, Oettingenstrasse 67, 80538 Muenchen, Germany, [2]Fakultaet fuer Informatik, Technische Universitaet Muenchen, Boltzmannstrasse 3, 85748 Garching b. Muenchen, Germany, [3]Bioinformatics group, Turku Centre for Biotechnology, Finland, [4]Institute for Bioinformatics GSF—Forschungszentrum fuer Umwelt und Gesundheit, GmbH, Ingolstaedter Landstrasse 1, 85764 Neuherberg, Germany and [5]Department of Genome-Oriented Bioinformatics, Wissenschaftszentrum Weihenstephan, Technische Universitaet Muenchen, 85350 Freising, Germany

Received on September 16, 2004; revised on November 16, 2004; accepted on November 30, 2004
Advance Access publication December 7, 2004

ABSTRACT

Motivation: Discovery of host and pathogen genes expressed at the plant–pathogen interface often requires the construction of mixed libraries that contain sequences from both genomes. Sequence identification requires high-throughput and reliable classification of genome origin. When using single-pass cDNA sequences difficulties arise from the short sequence length, the lack of sufficient taxonomically relevant sequence data in public databases and ambiguous sequence homology between plant and pathogen genes.

Results: A novel method is described, which is independent of the availability of homologous genes and relies on subtle differences in codon usage between plant and fungal genes. We used support vector machines (SVMs) to identify the probable origin of sequences. SVMs were compared to several other machine learning techniques and to a probabilistic algorithm (PF-IND) for expressed sequence tag (EST) classification also based on codon bias differences. Our software (ECLAT) has achieved a classification accuracy of 93.1% on a test set of 3217 EST sequences from *Hordeum vulgare* and *Blumeria graminis*, which is a significant improvement compared to PF-IND (prediction accuracy of 81.2% on the same test set). EST sequences with at least 50 nt of coding sequence can be classified using ECLAT with high confidence. ECLAT allows training of classifiers for any host–pathogen combination for which there are sufficient classified training sequences.

Availability: ECLAT is freely available on the Internet (http://mips.gsf.de/proj/est) or on request as a standalone version.

Contact: friedel@informatik.uni-muenchen.de

1 INTRODUCTION

The characterization of interactions between plants and their pathogens is a major contemporary research area and has its roots within agriculture and disease control. To analyze genes expressed within plant defense mechanisms and pathogen virulence at the molecular level, cDNA libraries may be constructed from either infected or challenged tissues. The subsequent single-pass sequencing of these cDNAs produces expressed sequence tags (ESTs). There are currently over 100 000 ESTs within the public sequence databases clearly annotated as coming from mixed plant–pathogen interactions.

Within the realm of comparative plant genomics and gene identification there is a need to simply and reliably identify and filter out the non-plant sequences. Although experimental techniques can be applied for this purpose, they are laborious and time-consuming and therefore become infeasible for large numbers of EST sequences. Hence efficient and reliable computational EST classification methods are required. The canonical approach involves performing a BLAST search against genetic databases such as GenBank to find a significant unambiguous match that resolves either the plant or the pathogen origin of sequence. This approach is based on the assumption that a plant sequence will be more homologous to any other plant sequence than to a pathogen sequence due to taxonomic proximity. However, it has been shown that biased taxa representation in existing databases decreases reliability of this homology approach (Koski and Golding, 2001).

An advanced method (Hsiang and Goodwin, 2003) tackles this problem by using a restricted database for homology search consisting of a single plant and fungal genome, each closely related to the infected plant and the fungal pathogen, respectively. However, in most cases this approach is limited by the lack of complete or adequate genome coverage for related organisms. Moreover, difficulties in classification are increased by the relatively high-sequence homology of plant and fungal genes with conserved functions. Therefore, a method is desired which is independent of sequence homology and the availability of genomic sequences.

A suitable approach to this task employs codon usage preferences that vary significantly between species (Sharp *et al*., 1988) and are correlated to GC-content at the third codon position (Kawabe and Miyashita, 2003; Fennoy and Bailey-Serres, 1993). A probabilistic algorithm based on this observation is PF-IND (Maor *et al*., 2003), which compares the actual number of occurrences of codons of different types (G or C ending versus A or T ending) for particular amino acids in an EST sequence with the expected number of

*To whom correspondence should be addressed.

†The authors wish it to be known that, in their opinion, the first authors should be regarded as joint First Authors.

occurrences in plant and fungus respectively using Poisson distribution. The resulting probabilities are used to classify sequences as either plant or fungal.

In this paper we show that by applying standard machine learning algorithms, classification accuracy can be improved decisively compared to the simple probabilistic approach. We present a novel method for the classification of EST sequences based on support vector machines (SVMs), which is independent of homology criteria and relies only on codon usage differences. This method was used to train a classification scheme to discriminate between EST sequences from *Hordeum vulgare* and *Blumeria graminis* and can easily be extended to other plant–pathogen pairs.

2 METHODS

Support vector machines

The use of SVMs is a prevalent technique for data classification based on linear decision rules (Vapnik, 1995; Burges, 1998; Boser *et al*., 1992). SVMs take as input i.i.d. (independent and identically distributed) training samples $(x_1, y_1), \ldots, (x_n, y_n)$ where x_i represents the sample attributes and $y_i \in \{-1, +1\}$ the class.

SVMs will then find a hyperplane separating the training instances by their classes and maximizing the distance from the closest examples to the hyperplane (maximum-margin hyperplane). The classification of a sample will be determined by the sign of the function

$$f(x) = w^{\mathrm{T}}x + b$$

where w and b are the parameters of the hyperplane. The examples closest to the hyperplane are called support vectors and are crucial for training.

For many training sets it will not be possible to separate samples by a linear function in the original feature space, so training instances are mapped into a higher dimensional space by a function ϕ. SVM will then find a linear maximum-margin hyperplane in this higher dimensional space. For solving this problem it is not necessary to directly define the mapping into higher dimensional space, but it is sufficient to give the dot product of two instances in this space (Burges, 1998). $K(x_i, x_j) = \phi(x_i)^{\mathrm{T}}\phi(x_j)$ is called a kernel function. Commonly used kernel functions comprise linear, polynomial, sigmoid and radial basis functions (RBF). For our purpose we used an RBF kernel, as the linear kernel has been proven to be a special case of the RBF kernel (Keerthi and Lin, 2003) and the sigmoid kernel appears to behave like RBF for some parameters (Lin and Lin, 2003). Moreover, RBF has less hyperparameters than the polynomial kernel and is less difficult numerically than both sigmoid and polynomial kernel. The radial basis function is defined by

$$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2), \quad \gamma > 0.$$

The parameters of the maximum-margin hyperplane are calculated by solving a quadratic programming optimization problem (Boser *et al*., 1992) and there exist several specialized algorithms for solving this problem efficiently (Joachims, 1999; Platt, 1998).

Training sequences

We used a dataset of 3974 unigene sequences of various lengths from barley (*H.vulgare*, 1487 sequences) and blumeria (*B.graminis*, 2487 sequences). Unigene sequences were chosen to avoid redundancy and derived as follows. Public EST sequences from both *B.graminis* and *H.vulgare* were clustered and assembled within the Sputnik EST analysis pipeline (Rudd *et al*., 2003). The Hashed Position Tree clustering algorithm (Heumann and Mewes, 1996) was used to cluster sequences using a similarity threshold of 0.7 and 100 network iterations to describe a cluster. Sequence assembly was performed using CAP3 (Huang and Madan, 1999) with default parameters. For each of the derived unigenes regions of likely coding sequence (CDS) were identified by a BLASTX comparison against a non-redundant sequence database. Best matches exceeding the arbitrary expectation threshold of 1×10^{-15} were filtered and probable CDS was collated in species specific datasets. These collections of sequences were used to derive a species specific codon usage and hexanucleotide probability tables with the FrameFinder application (Slater, 2000) to predict a single uninterrupted CDS for each of the unigenes. Short sequences were excluded (threshold of 21 bp).

Sequence attributes

For each sequence in the dataset codon frequencies are computed from its beginning up to the first stop codon. To account for some codons missing randomly, pseudocounts are used. The frequency for a given codon c is computed as

$$F(c) = \frac{n_c + 1}{\sum_{c' \in \mathrm{Codons}} n_{c'} + 64}$$

where n_i denotes the absolute number of occurrences of a codon i. Therefore, for each sequence 64 attributes are derived for training and classification. Those attributes are computed for each of the six possible frames of a sequence separately so that every sequence provides six training instances, one correct frame and five incorrect frames.

Training

There are two parts in building a classification model. To begin with a support vector model is calculated to distinguish between correct and wrong frames in a sequence. The training instances for computing this model are chosen as follows. From half of the original sequences (chosen randomly) the correct frame is used, whereas from the other half a randomly chosen wrong frame is used. This is done to ensure that correct and incorrect frames are represented equally in the training instances. First, the maximum and minimum values for each attribute over all training instances are determined. These values are then used to scale training instances such that all attribute values lie between -1 and 1. Afterwards they are stored for later use in classification. After scaling the instances, the support vectors are computed using a RBF kernel.

The second step is then to learn a classifier for separating the two possible classes (plant and fungus). In this case only the correct frame of each sequence is used for training. As before maximum and minimum values for each attribute are calculated and the training instances are scaled appropriately to lie in the interval $[-1, 1]$. Then support vectors are calculated in the same fashion as before. Table 1 describes the general procedure for each of the two steps.

Classification

Classifying a sequence also consists of two parts. First, the coding frame is determined and then classified as being of plant or fungus origin. To determine the correct frame, the sequence's attributes are first scaled using the precomputed scaling parameters for frame determination. (Note that now the attributes will no longer necessarily lie in the range of $[-1, 1]$.) Following scaling the first SVM model is then used to classify the six possible frames as being correct or incorrect. As every frame is classified independently, it does happen occasionally that all frames are classified as being incorrect or more than one frame is classified as being correct. In this case the reading frame with the largest predicted margin is chosen. Having selected the coding frame, scaling on the original attribute values is applied again, this time using the scaling parameters for classification. Afterwards this frame is classified by the SVM model for discriminating between plant and fungus origin. See Table 2 for a summary of the steps involved.

Software

ECLAT is an implementation of the described method and is available online (http://mips.gsf.de/proj/est). It consists of a web-frontend and Java packages. Computing of SVMs is done using the freely available software package LIBSVM (Chang and Lin, 2001, http://www.csie.ntu.edu.tw/~cjlin/libsvm), which provides implementations for support vector classification, regression and distribution estimation based on the algorithms SMO (Platt, 1998) and

Table 1. Training procedure: $i = 1$ for frame model, $i = 2$ for class model

Let T be the set of all training instances

(1) For each attribute a_j compute
- $\max_{ij} \leftarrow \max_{t \in T} a_j(t)$
- $\min_{ij} \leftarrow \min_{t \in T} a_j(t)$

(2) Scale training instances using $\max_{ij}$ and $\min_{ij}$ for attribute a_j
(3) Calculate support vectors using scaled training instances

Table 2. Classification procedure

Let t be an unclassified sequence

(1) Calculate codon frequencies for each frame
(2) Scale frames with $\min_{1j}$ and $\max_{1j}$, $j \in [1, 64]$
(3) Determine correct frame using SVM model
(4) Scale correct frame with $\min_{2j}$ and $\max_{2j}$, $j \in [1, 64]$
(5) Classify frame using SVM model

SVMLight (Joachims, 1999). For communication between the frontend and the Java application, an XML format has been defined.

Frontend The user interface consists of two HTML input forms for classification and training mode respectively. The server is based on Tomcat as container. For handling of incoming requests JSP and Java are used. All request parameters are first stored in a Java Bean, tested for validity (e.g. correct sequence format) and transformed into the XML format defined as internal data interface. After processing the request, an XSL template and a modified version of the tag libraries of the Jakarta project (http://jakarta.apache.org) are used to transform XML result data into HTML for presenting it to the user.

Application layer Classes have been implemented in Java for data storage, training and classifying as well as managing the process. For support vector classification Java implementations of LIBSVM are used, which allows for fast training and classification.

The application starts by parsing the XML input to determine the mode (training or classification) which is to be performed. If training mode is chosen, the training sequences are read. For each of the training sequences codon frequencies are computed for all six frames. These attributes are used to perform training as described above. Additional unclassified sequences can be classified afterwards using the trained model. To estimate performance of the trained classifier 10-fold cross-validation can be chosen. This will slow down the training process as 10 additional training rounds have to be performed, each with a training set nine-tenths the size of the original training set.

In classification mode all unclassified sequences are read from the XML input. Codon frequencies are computed for each frame of each sequence. This information is then used to classify the sequence by the presented method. In doing this, the user has the choice to apply either the online available model for *H.vulgare* and *B.graminis* or to use a model previously trained with ECLAT for any other plant–fungus pair.

3 RESULTS

Comparison of different machine learning algorithms and PF-IND

Several machine learning algorithms as well as a probabilistic algorithm were compared with regard to their performance on discriminating plant from fungal ESTs. Performance was measured in terms of prediction accuracy using the mixed barley/blumeria dataset described before but including only sequences longer than 100 bp. Accuracy is defined as the percentage of correctly classified instances. For this comparison, performance was calculated only on the correct frame. The machine learning algorithms applied to the task comprised SVMs (LIBSVM; Chang and Lin, 2001), artifical neural networks (ASNN; Tetko, 2002, http://www.vcclab.org), random forests (WEKA implementation; Witten and Frank, 2000), Naive Bayes (WEKA), decision trees (J4.8 from WEKA) and Rule learning (JRip from WEKA). Additionally PF-IND (Maor *et al*., 2003) was evaluated, an algorithm designed specifically to separate ESTs from mixed plant–fungus EST-pools. To allow high-throughput analysis the method described by Maor *et al*., 2003 was re-implemented using the same codon usage frequencies. We compared the predictions of the original implementation and our version for several sequences. Not only did we get the same predictions for those sequences, but also the same scores for all frames. Thus, the original and our version of PF-IND calculated the same results for this test and both implementations do not differ.

For evaluation three kinds of attributes were calculated for each sequence: codon frequencies (64 attributes), %GC3 (18 attributes) and fungus probabilities (18 attributes). Codon frequencies were computed as described in the Methods section. The %GC3 for an amino acid denotes the ratio of occurrence of GC-ending codons to overall occurrence of the amino acid they are coding for. These attributes were included since the main source of variation in codon bias are the levels of C- and G-ending codons (Fennoy and Bailey-Serres, 1993; Kawabe and Miyashita, 2003). Fungus probabilities describe the probability that a given number of GC- and AT-ending codons coding for the same amino acid will occur in an EST from the fungus species. This was an approach similar to PF-IND, but without previous selection of amino acids. These probabilities were estimated using Poisson distribution and codon usage frequencies obtained from the online database site http://www.kazusa.or.jp/codon/. Note that the last two types of attributes were computed for each amino acid separately, excluding methionine and tryptophan.

To estimate performance, repeated holdout with stratification was applied on the dataset. The holdout procedure consisted of splitting the data randomly into training and test sets. One-third of the dataset was chosen for testing and the remainder for training. The additional use of stratification ensured that both classes were represented by the same proportions in both training and test sets. With each machine learning algorithm a classifier was learned from the training set and performance was estimated on the test set. This was repeated 10 times with different random splits to decrease any bias due to a particular sample choice. For PF-IND no training was necessary as precomputed codon frequencies were used, but to make performance comparable only sequences from the respective test sets were classified. Table 3 shows the average results for the 10 holdouts. No significant differences between the performances of SVMs and artificial neural networks were detected using a paired t-test; therefore, both performed equally well on the given dataset.

Comparison of different subsets of attributes concerning prediction accuracy

We evaluated the attribute types described before separately and combined with regard to performance as instance attributes for support vector classification on the same random training and test splits as

Table 3. Estimates for classification accuracy of machine learning algorithms using repeat holdout with stratification on the barley/blumeria dataset

Accuracy (in %)	Algorithm SVM	ASNN	Random Forest	Naive Bayes	JRip	J4.8	PF-IND
Average	92.9	92.4	88.9	87.8	85.9	82.2	82.0
SD	0.6	0.5	1.0	0.9	1.8	1.6	0.9

Results for the different holdout splits were averaged. Accuracy is defined as the number of correctly classified instances divided by the total number of instances. No abstaining on instances is applied yet, thus error rate and accuracy sum up to one.

Table 4. Estimates for classification accuracy using different subsets of attributes

Attributes Type	#	Accuracy (in %) Average	SD
CF	64	93.5	0.8
%GC3	18	88.9	0.5
FP	18	62.6	2.6
CF + %GC3	82	93.5	0.6
CF + FP	82	93.1	0.5
%GC3 + FP	36	88.6	0.8
All	100	92.9	0.6

Abbreviations: CF, codon frequencies; %GC3, percentage of synonymous codons ending at G or C; FP, probability of fungus origin of sequence (%GC3 and FP computed separately for each amino acid). Results for the 10 random holdout splits were averaged.

above (Table 4). Using only codon frequencies as sequence attributes resulted in an average classification accuracy of 93.5%, which was significantly better than using the complete attribute set. (Significance was tested with paired *t*-tests.) Additional use of %GC3 or fungus probabilities with codon frequencies did have no significant effect on prediction performance. Contrary to codon frequencies the subset containing exclusively probabilities of fungal origin performed poorly with a prediction accuracy at 62.8%. When we tried probabilities of plant origin of sequence or ratios of both instead these results did not improve.

Prediction of the correct reading frame

To assess ECLAT's capability of predicting the correct reading frame, the first step of the classification procedure was evaluated separately. For that purpose frame prediction performance of ECLAT was determined on the previously described 10 holdout splits and compared to PF-IND predictions (Table 5). Analysis showed that ECLAT predictions of the correct frame were highly reliable (average prediction accuracy of 97.7%) whereas PF-IND predicted the frame of only 71.4% of the sequences correctly.

Performance in discriminating plant and fungal sequences

So far the accuracy of each of the two steps of the ECLAT methodology has been examined separately. To estimate performance of the complete procedure 10-fold cross-validation was repeated 10 times. Ten-fold cross-validation consisted of splitting the barley/blumeria dataset randomly into 10 parts and then alternatingly using one part for testing and the remainder for training, excluding sequences

Table 5. Estimates for classification accuracy of ECLAT and PF-IND in predicting the correct reading frame

Algorithm	Accuracy (%) Average	SD
ECLAT	97.7	0.4
PF-IND	71.4	0.7

Results for the 10 random holdout splits were averaged.

shorter than 100 nt from the dataset (leaving 3217 sequences). The average accuracy in predicting the origin of sequence was 93.1% for ECLAT. As 10-fold cross-validation provided a prediction for each single instance in the dataset, performance of PF-IND was estimated on all barley/blumeria sequences longer than 100 bp, which resulted in an estimated prediction accuracy of 81.2% for PF-IND.

Dependence of classification performance on sequence length

The length of an EST sequence determines the accuracy of its classification. In short sequences it will not be possible to estimate the underlying codon frequencies correctly, as only few codons occur at all. In this case pseudocounts will dominate the calculated codon frequencies. To study this effect, different sequence intervals were analyzed in relation to the resulting accuracy. Repeated 10-fold cross-validation was performed with sequence length boundaries of 50, 100, 200, 300, 400 and 500 bp for testing. Results of this analysis can be seen in Table 6. The same intervals were used in estimating the performance of PF-IND.

ECLAT predictions became less precise for shorter sequences, but still for sequence lengths between 50 and 100 the origin of a sequence could be determined with an average accuracy of 90%. The highest prediction accuracies were achieved for sequences between 300 and 500 bp, whereas for sequences longer than 500 bp accuracies decreased slightly. PF-IND performance also increased with sequence length, but for sequences longer than 500 bp classification accuracy dropped as low as 71.8%. We examined if PF-IND results for long sequences could be improved by recalculating codon usage based only on long sequences. However, classification accuracy did not increase.

Performance on EST sequences from GenBank

The performance of ECLAT and PF-IND additionally was tested on EST sequences from GenBank. These sequences were deposited in GenBank in the second part of 2003 or later and did not overlap

Table 6. Estimates for classification accuracy of ECLAT and PF-IND for different sequence length intervals

	Sequence length (bp) [21, 50]	[51, 100]	[101, 200]	[201, 300]	[301, 400]	[401, 500]	≥500
No. of sequences	431	326	748	768	646	502	553
ECLAT							
Average accuracy (%)	78.8	90.0	90.6	92.2	95.6	95.7	94.4
SD (%)	0.9	0.9	0.4	0.4	0.3	0.5	0.5
PF-IND							
Accuracy (%)	71.9	81.6	81.0	83.5	86.7	81.1	71.8

Estimates were performed by repeated 10-fold cross-validation. Results for the 10 repeats were averaged.

Table 7. Estimates for classification accuracy of ECLAT (with and without abstention) and PF-IND on EST sequences from GenBank

Algorithm	Accuracy (%) *H.vulgare*	*B.graminis*
ECLAT	81.0	91.9
ECLAT with abstention	87.6	96.5
PF-IND	56.6	90.1

with the training set used to develop the method. The second test set contained 931 sequences from blumeria and 9312 sequences from barley. The composition of the test set mirrored the taxa bias in EST databases towards plant EST sequences. Furthermore, a minimum sequence length threshold of 100 bp was applied again. The results are shown in Table 7.

In this test classification, accuracy for blumeria was comparable to previous results for ECLAT and increased for PF-IND, whereas for barley classification accuracy dropped significantly for both ECLAT and PF-IND. We also tested if classification accuracy could be increased by abstaining, i.e. not classifying sequences for which no frame is classified as correct. Indeed, this did raise classification accuracy by more than 6% for barley and almost 5% for blumeria. However, ∼19% of the sequences remained unclassified. Contrary to that only 1.3% of the barley/blumeria unigene sequences remained unclassified when using 10-fold cross-validation with abstention, which was not done in the previous tests.

Application to different organism pairs

To test if our methodology is also applicable to a wider range of biological targets, we used a second dataset containing EST sequences from cotton (*Gossypium arboreum*, 2028 sequences) and cotton root knot nematode (*Meloidogyne incognita*, 2040 sequences). This training set was derived using the same method as before, so the methodology remains consistent with what has already been done. The prediction accuracy of a model trained on this dataset as estimated by 10-fold cross-validation is ∼87.3%. This result clearly indicates that the methodology is also applicable to other systems, such as plant/nematode. Since the classification performance is lower compared to the barley/blumeria analysis, we can conclude that the codon compositions of both these species are similar and their ESTs are more difficult to separate.

4 DISCUSSION

Previous approaches to the prediction of species origin for sequences from mixed plant–pathogen EST collections have utilized homology-based methods. The objective of ECLAT was to provide an easy-to-use interface for high-throughput, automatic classification of ESTs derived from pathogen-infected plants, which does not rely on the existence of homologous sequences in public databases. Differences in codon frequencies between plant and fungi have already proven to be a reliable basis for fast computational EST classification (Maor *et al*., 2003). In this paper, we demonstrated that utilization of machine learning methods (SVMs) could improve results decisively compared to the existing probabilistic algorithm PF-IND. Analysis of sequences longer than 100 bp from a mixed barley–blumeria EST dataset showed an average accuracy of 93% for ECLAT compared to 81% for PF-IND. For a second test set containing sequences from GenBank both classifiers performed worse than in previous tests. This drop in accuracy can be explained by vector contaminations or low-quality regions within EST sequences deposited in databases. Further detrimental effects may result from high redundancy within EST databases as well as the fact that many of the tested EST sequences may not contain a coding sequence at all. In fact, it could be shown that classification accuracy could be increased further by abstaining from classifying sequences where no frame was predicted as correct. This suggests that for a large fraction of ESTs, the observed codon frequencies deviate from the expected due to a lack of coding sequence or contamination. This violates basic assumptions of both ECLAT and PF-IND. Therefore, before using ECLAT care should be taken to remove vector contaminations and to assess sequence quality.

Currently ECLAT only provides a pre-built model for the plant–fungus pair barley and blumeria, but the design of the software allows extensions to more organism pairs, as it does not rely on any specific characteristics of barley and blumeria except for codon bias differences. Researchers have the possibility to train ECLAT specifically for plant–fungus pairs of interest using EST sequences from their laboratories. In general, we believe that it will be impossible to specify some objective criteria to predict the accuracy of EST separation 'a priori'. Nevertheless, it is always possible to develop a new classifier using the available data and test its performance with cross-validation to decide if the proposed methodology can or cannot be applied in each particular case.

The use of two additional types of attributes derived from codon frequencies proved to have a neutral or even detrimental effect on prediction accuracy. The results suggest that both types of attributes

hardly contribute any new information compared to the original codon frequencies. Furthermore, the slightly but significantly lower accuracy on the complete attribute set implies overfitting effects. None of the additional attribute sets taken separately did perform as well as codon frequencies. Unexpectedly the probabilistic attributes performed very poorly, although a good algorithm based on these probabilities exists (Maor *et al*., 2003). The exclusion of those attributes provided model simplicity, which is desired to reduce overfitting effects, without decreasing overall prediction accuracy. In the future further attribute types should be considered that are not dependent on codon frequencies such as occurrences of short sequence motifs or GC-content.

Several standard machine learning algorithms were evaluated on the task of training models to classify EST sequence data. Here SVMs and artificial neural networks proved to be equally capable, but SVMs, at least the used implementation, had the advantage of speed, which becomes an important factor when learning with large training sets. Unfortunately SVMs produce 'black box' models, i.e. the reason for misclassification of sequences in most cases is unclear. Nevertheless, some tendencies can be observed, such as a negative correlation between the absolute value of the margin of a sequence and its probability to be classified. PF-IND could establish the origin of 94 of 100 test sequences in a previous test (Maor *et al*., 2003), whereas on the 3217 sequences from barley and blumeria used for our estimates its prediction accuracy reached only 82% on average.

Although programs exist for predicting correct reading frames in ESTs, ECLAT does not use any of these, but instead applies machine learning techniques, reaching an accuracy of 98%. Of course, the task here has been simplified as only two organisms have to be considered. Nevertheless training data for each of the two classes (correct versus incorrect frame) are heterogeneous since they contain sequences from both organisms. The high classification accuracy therefore suggests, that the difference in codon usage is more severe between correct and incorrect frames than between organisms. A possible explanation could be that incorrect frames contain codons, which are never or rarely observed for the correct frames. Since the same problem is relevant for both plant and fungi, the algorithm correctly identifies the coding frames for both of them.

The prediction accuracy of both ECLAT and PF-IND increases with sequence length but decreases again for sequences longer than 500 bp. Yet the drop in prediction accuracy for very long sequences is much more pronounced for PF-IND. This is consistent with the observation that quality within EST sequences rapidly degrades as length exceeds an optimum of $\sim$400 nt (S. Rudd, unpublished data). For sequences of $<$50 nt, the distribution of codon frequencies deteriorates to a uniform distribution with peaks for only a few codons, thus making classification difficult.

ECLAT assists in the rapid and automatic analysis of ESTs. Nevertheless ECLAT should not replace BLAST analysis, which gives additional hint to gene function, but be used complementarily to validate BLAST results and give predictions when no close homolog can be found.

Although ECLAT was developed primarily for the purpose of classification of sequences from mixed plant–fungus EST-pools, our tests have shown that these methods are also applicable to other pairs of evolutionarily distinct organisms such as plant/nematode. Alternative applications of ECLAT may comprise automatic prediction of high- and low-expressed genes, since gene expression levels and codon bias are positively correlated (Duret and Mouchiroud, 1999), or the detection of putative alien sequences in a genome, which have originated from horizontal transfer events.

ACKNOWLEDGEMENTS

This work was supported by grant 031U118A from NGFN (to H.W.M.), BFAM and INTAS 00-0363 grant.

REFERENCES

Boser,B.E., Guyon,I.M. and Vapnik,V. (1992) A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, Pittsburg, PA, USA, pp. 144–152. ACM press, New York, NY.

Burges,C.J.C. (1998) A tutorial on support vector machines for pattern recognition. *Data Mining Knowl. Discov.*, **2**, 121–167.

Chang,C.-C. and Lin,C.-J. (2001) LIBSVM: a library for support vector machines.

Duret,L. and Mouchiroud,D. (1999) Expression pattern and, surprisingly, gene length shape codon usage in *Caenorhabditis*, *Drosophila*, and *Arabidopsis*. *Proc. Natl Acad. Sci., USA*, **96**, 4482–4487.

Fennoy,S.L. and Bailey-Serres,J. (1993) Synonymous codon usage in *Zea mays* L. nuclear genes is varied by levels of C and G-ending codons. *Nucleic Acids Res.*, **21**, 5294–5300.

Heumann,K. and Mewes,H.W. (1996) The Hashed Position Tree (HPT): a suffix tree variant for large data sets stored on slow mass storage devices. In Ziviani,N., Baeza-Yates,A. and Guimaraes,G. (eds), *Proceedings of the Third South American Workshop on String Processing*, Recife, Brazil, pp. 101–115.

Hsiang,T. and Goodwin,P.H. (2003) Distinguishing plant and fungal sequences in ESTs from infected plant tissues. *J. Microbiol. Methods*, **54**, 339–351.

Huang,X. and Madan,A. (1999) CAP3: a DNA sequence assembly program. *Genome Res.*, **9**, 868–877.

Joachims,T. (1999) Making large-scale SVM learning practical. In Schölkopf,B., Burges,C. and Smola,A. (eds), *Advances in Kernel Methods—Support Vector Learning*. MIT Press, Cambridge, MA.

Kawabe,A. and Miyashita,N.T. (2003) Patterns of codon usage bias in three dicot and four monocot plant species. *Genes Genet. Syst.*, **78**, 343–352.

Keerthi,S.S. and Lin,C.-J. (2003) Asymptotic behaviors of support vector machines with Gaussian kernel. *Neural Comput.*, **15**, 1667–1689.

Koski,L.B. and Golding,G.B. (2001) The closest BLAST hit is often not the nearest neighbor. *J. Mol. Evol.*, **52**, 540–542.

Lin,H.-T. and Lin,C.-J. (2003) A study on sigmoid kernels for SVM and the training of non-PSD kernels by SMO-type methods. *Technical report*, Department of Computer Science and Information Engineering, National Taiwan University.

Maor,R., Kosman,E., Golobinski,R., and Goodwin,P. and Sharon,A. (2003) PF-IND: probability algorithm and software for separation of plant and fungal sequences. *Curr. Genet.*, **43**, 296–302.

Platt,J. (1998) Fast training of support vector machines using sequential minimal optimization. In Schölkopf,B., Burges,C. and Smola,A. (eds), *Advances in Kernel Methods—Support Vector Learning*. MIT Press, Cambridge, MA, pp. 185–208.

Rudd,S., Mewes,H.W. and Mayer,K.F. (2003) Sputnik: a database platform for comparative plant genomics. *Nucleic Acids Res.*, **31**, 128–132.

Sharp,P.M., Cowe,E., Higgins,D.G., Shields,D.C., Wolfe,K.H. and Wright,F. (1988) Codon usage patterns in *Escherichia coli*, *Bacillus subtilis*, *Saccharomyces cerevisiae*, *Schizosaccharomyces pombe*, *Drosophila melanogaster* and *Homo sapiens*: a review of the considerable within-species diversity. *Nucleic Acids Res.*, **16**, 8207–8211.

Slater,G. (2000) Algorithms for analysis of ESTs. Ph.D. thesis, University of Cambridge, UK.

Tetko,I.V. (2002) Neural network studies. 4. Introduction to associative neural networks. *J. Chem. Inf. Comput. Sci.*, **42**, 717–728.

Vapnik,V. (1995) *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, NY.

Witten,I.H. and Frank,E. (2000) *Data Mining: Practical Machine Learning Tools with Java Implementations*. Morgan Kaufmann, San Francisco, CA.