In silico prediction of noncoding RNAs using supervised learning and feature ranking methods

Stephen J. Griesmer

Bioinformatics Program,
New Jersey Institute of Technology,
Newark, NJ 07102, USA
E-mail: sjg7@njit.edu

Miguel Cervantes-Cervantes

Department of Biological Sciences,
Rutgers University,
Newark, NJ 07102, USA
E-mail: miguelcc@andromeda.rutgers.edu

Yang Song and Jason T.L. Wang*

Department of Computer Science,
New Jersey Institute of Technology,
Newark, NJ 07102, USA
E-mail: ys54@njit.edu E-mail: wangj@njit.edu
*Corresponding author

Abstract: We propose here a new approach for ncRNA prediction. Our approach selects features derived from RNA folding programs and ranks these features using a class separation method that measures the ability of the features to differentiate between positive and negative classes. The target feature set comprising top-ranked features is then used to construct several classifiers with different supervised learning algorithms. These classifiers are compared to the same supervised learning algorithms with the baseline feature set employed in a state-of-the-art method. Experimental results based on ncRNA families taken from the Rfam database demonstrate the good performance of the proposed approach.

Keywords: noncoding RNA; classification and prediction; feature generation and ranking.

Reference to this paper should be made as follows: Griesmer, S.J., Cervantes-Cervantes, M., Song, Y. and Wang, J.T.L. (2011) 'In silico prediction of noncoding RNAs using supervised learning and feature ranking methods', *Int. J. Bioinformatics Research and Applications*, Vol. 7, No. 4, pp.355–375.

Biographical notes: Stephen J. Griesmer received a BSE in Electrical Engineering and Computer Science from Princeton University, an MS in Interdisciplinary Studies from University of British Columbia, and an MS in Bioinformatics from New Jersey Institute of Technology. His research interests include machine learning, data mining, data networking, and bioinformatics.

Miguel Cervantes-Cervantes is a Licentiate in Biology from the Instituto Politécnico Nacional in Mexico City, where he also completed a Master of Science Degree in Biochemistry. In 1991, he received a PhD in Biochemistry from Rutgers University and the University of Medicine and Dentistry of New Jersey (UMDNJ). He is the Coordinator of Undergraduate Studies at the Federated Department of Biological Sciences of Rutgers University-Newark Campus and New Jersey Institute of Technology. His research focuses on plant metabolism and bioinformatics.

Yang Song received a BE in Computer Science from the University of Science and Technology of China. He is currently a PhD candidate in the Computer Science Department at New Jersey Institute of Technology, and a student intern with the UMDNJ-New Jersey Medical School Cancer Centre in Newark, New Jersey. His research interests include data mining and bioinformatics.

Jason T.L. Wang received a BS in Mathematics from National Taiwan University, Taipei, Taiwan, and a PhD in Computer Science from the Courant Institute of Mathematical Sciences at New York University in 1991. He is Professor of Computer Science and Bioinformatics at New Jersey Institute of Technology and Director of the university's Data and Knowledge Engineering Laboratory. His research interests include data mining, databases, computational biology and bioinformatics. He has published 6 books and over 120 papers in these areas.

1 Introduction

Non-coding RNAs (ncRNAs) are RNA transcripts that are not translated to proteins. They include transfer RNA (tRNA), ribosomal RNA (rRNA), and many other functionally important RNA molecules (Gesteland et al., 2005). The sise of ncRNAs varies from 20 to several thousand nucleotides in length. Tens of thousands of ncRNAs are produced in human cells – more than protein-coding RNAs. The ENCODE project has revealed that up to 93% of nucleotides in the human genome are transcribed to RNA, with the majority of transcripts belonging to long ncRNAs (Chang et al., 2009; Valadkhan and Nilsen, 2010). This finding has impact on several areas of study. The non-coding transcriptome displays a rapid rate of evolution, potentially having a role in genomic differences that lead to increasing complexity among species but that are not reflected in the proteome. The production of non-coding transcripts undergoes temporal and spatial regulation, thus indicating that ncRNAs do not merely result from background transcription. In recent years, several studies have shown that ncRNAs may have crucial roles in a number of phenomena, including embryo neurodevelopment in humans; T-cell stimulation in vitro; and possibly, in *cis*-regulation of protein-coding genes. Experimental work will shed more light on the physiological roles of ncRNAs and their many potential therapeutic applications.

The detection of ncRNAs is difficult because unlike protein-coding genes, ncRNAs lack statistical signals for reliable detection from primary sequences (Menzel et al., 2009). A commonly used approach is to consider both the primary and secondary structures of RNA in predicting and detecting the presence of ncRNAs. The secondary structure of an ncRNA molecule determines its function. This secondary structure can be conserved even with substantial changes to the primary DNA sequence that encodes it.

Several methods have been proposed for ncRNA prediction. QRNA (Rivas and Eddy, 2001) uses Hidden Markov Models (HMMs) and Stochastic Context-Free Grammars (SCFGs) in the production of evolutionary models for three regions of DNA, namely those encoding structural RNA, polypeptides, or other. The models and grammars are paired in that they examine or emit two aligned sequences to determine the probability of the existence of ncRNA under the different models. These sequences are presumed to be evolutionarily related and the emission probabilities of the QRNA program are based on this assumption. The structural RNA model for determining the probability of a structure is based on a stochastic context-free grammar that incorporates probabilities of structural elements like hairpins, internal loops, and bulges. The protein-coding model assumes that aligned RNA sequences encode homologous proteins based on the expectation of conservative amino acid substitutions and synonymous mutations. The model for protein coding uses a pair-hidden Markov model that emits a codon (three nucleotides) at once for two aligned sequences. The other model is a 'catch-all' that assumes that there is no structural or sequence conservation and that mutations are position-independent. This model uses a pair-hidden Markov model that emits a pair of nucleotides at a time. The three models also factor in the insertion and deletion of nucleotides in the sequences as well as boundary conditions that arise in the beginning and ending positions of an alignment. In order to classify an input alignment as structural RNA, protein-coding, or other, QRNA calculates the Bayesian posterior probability, under the assumption that the three models are equally likely in ncRNA prediction.

EvoFold (Pedersen et al., 2006) uses phylogenetic stochastic context-free grammars (phylo-SCFGs) to create a probabilistic model of RNA secondary structure and sequence evolution to predict noncoding RNAs. The input to EvoFold is a Multiple Sequence Alignment (MSA) and a phylogenetic tree. EvoFold takes these inputs and calculates probabilities of two different phylo-SCFGs: a model for functional RNAs and a model for background sequences, the null hypothesis. The model for functional RNAs is composed of two types of phylogenetic submodels: a dinucleotide submodel for base pairs and a single nucleotide submodel for unpaired bases. Each of these models contains a substitution process to examine the multiple sequence alignment. The substitution process for the dinucleotide model favours compensatory mutations among base pairs; the substitution process for the single nucleotide model is based on the evolutionary process for unpaired nucleotides. The output of EvoFold includes the functional RNA detected and the log likelihood of the input MSA to contain the functional RNA. The EvoFold program is trained on data from the Rfam database (Griffiths-Jones et al., 2003). Human entries from Rfam are aligned to the human genome using BLAT available from the UCSC Genome Browser. Human-mouse syntenic matches that overlap with these human entries are selected. Aligned sequences with poor secondary structures are removed. The alignments are expanded to six additional sequences – chimp, rat, dog, chicken, fugu, and zebra fish. Alignments that result in less than four sequences are discarded. Parameters for the phylo-SCFG are found using the EM algorithm and the quasi-Newton method.

MSARI (Coventry et al., 2004) computes the statistical significance of short, contiguous base-paired regions of potential structural RNAs in Multiple Sequence Alignments (MSAs). It handles a wide range of evolutionary distances within the MSAs by varying the null hypothesis model from sequence to sequence. MSARI uses RNAfold (Hofacker, 2003) as a preprocessor to locate probable base pairs. For each pair of

positions in a base pair, MSARI examines windows of length 7 for complementary mutations. Misalignments of orthologous base pairs up to two nucleotides can be tolerated by MSARI. A Bonferroni-style test for rejection of the null hypothesis is used. Significant base pairings are sorted. Pseudoknots are eliminated. The probabilities of the selected pairs are multiplied together to determine the score of the aligned sequences. With 10- and 15-sequence alignments, the performance of MSARI in terms of sensitivity and specificity was shown to be better than other methods.

Dynalign (Uzilov et al., 2006) employs a dynamic programming algorithm for computing the minimum free energy for secondary structure formation, and the structural alignment of a pair of sequences. The tool simultaneously optimises the common secondary structure between the pair of sequences and aligns their structures. A Support Vector Machine (SVM) was used together with Dynalign to classify ncRNA sequences.

ddbRNA (Di Bernardo et al., 2003) is a program that makes use of multiple alignments for the detection of conserved RNA secondary structures. The algorithm used by ddbRNA counts the compensatory mutations for every possible stem-loop and compares them to the average obtained from the aligned sequences that have their columns shuffled. Gaps are not included in the analysis; all bases that align to a gap are removed. Di Bernardo et al. (2003) showed that ddbRNA detected 5S rRNA genes while QRNA did not, though the sensitivity of ddbRNA drops with increasing pairwise identity percentages in sequences.

RNAz (Washietl et al., 2005) combines multiple alignments of 2–4 sequences with measures of secondary structure conservation and thermodynamic stability of base pairs to predict the presence of noncoding RNAs. RNAz builds on other programs to accomplish its goal. These programs include RNAfold (Hofacker, 2003) for folding single sequences, RNAalifold (Hofacker et al., 2002) for predicting the consensus structure of aligned sequences, and the SVM tool Libsvm (Fan et al., 2005). Thermodynamic stability is measured by minimum free energy. RNAz compares the minimum free energy of base pairing within a given sequence to random sequences of the same length and base composition. These are combined into a z-score, where negative z-scores indicate that a sequence is more stable than expected by chance. The minimum free energy of the consensus structure, as calculated by RNAalifold, is compared to the average minimum free energy of the secondary structures of the individual sequences in a multiple alignment through the usage of a Structure Conservation Index (SCI). The z-score and SCI are combined in the SVM learning algorithm, Libsvm. This SVM algorithm, trained on a large set of cross-species ncRNAs taken from the Rfam database (Griffiths-Jones et al., 2003), is able to predict an alignment as "structural noncoding RNA" or 'other'.

All the above methods use covariance of compensatory mutations in secondary structures of multiple sequences to discriminate RNA secondary structures (Babak et al., 2007). QRNA incorporates a pair-Hidden Markov Model to predict ncRNA structures based on compensatory mutations. EvoFold computes the probability of paired compensatory mutations against the probability of unpaired nucleotides. MSARI evaluates compensatory mutations nearby predicted paired bases in secondary structures. ddbRNA compares compensatory mutations for every possible stem-loop to shuffled sequences. Dynalign minimises the folding free energy change of a pair of input sequences and compares the energy change to control sequences generated specifically for that input pair. However, none of these tools uses a feature-based machine learning method for multiple sequence alignments, and hence it is difficult for one to try different

features for performance tuning. In contrast, RNAz uses a feature-based machine learning technique to classify multiple sequence alignments, and hence its methodology can be easily extended to include structural or conservational features in addition to the z-score and SCI to improve classification performance. These additional features may be derived from different folding algorithms or incorporate new measures. The purpose of this study is to investigate possible features and machine learning algorithms that could be used to enhance the performance of RNAz.

The approach proposed here follows closely the methodology of RNAz. We assign a given Multiple Sequence Alignment (MSA) into a positive class or a negative class based on features extracted from the given MSA. MSAs in the positive class are predicted as "structural noncoding RNA" and MSAs in the negative class are predicted as 'other'. Our approach includes features such as energy and structural parameters, which can be ranked according to their ability to differentiate between positive and negative classes through a Class Separation Measure (CSM). Features with the highest CSM scores are included in the prediction algorithms as described in the next section. We will compare our approach with RNAz to see if we can extend the features of RNAz and explore different machine learning algorithms to improve classification.

2 Materials and methods

2.1 Feature generation

In designing our ncRNA prediction algorithms, we examined features used by RNAz, features derived from the RNA folding programs RNAalifold (Hofacker et al., 2002), CentroidFold (Hamada et al., 2009), and RSpredict (Spirollari et al., 2009), as well as other features suggested in the literature. The z-score, SCI, the average minimum free energy (MFE_{avg}), and the consensus minimum free energy ($MFE_{consensus}$) were computed as in RNAz. As a measure of thermodynamic stability, the z-score compares the minimum free energy of a given MSA to the average minimum free energy of a set of shuffled MSAs of the same length and base composition. In this study, 100 shuffled MSAs were computed for this average. The z-score is calculated as follows:

$$Z = \frac{m - \mu}{\sigma}$$

where m is the minimum free energy of the given MSA, μ is the average minimum free energy of the shuffled MSAs, and σ is the standard deviation of the minimum free energies of the shuffled MSAs. Negative z-scores indicate that an MSA is more stable than expected by chance.

The SCI is the ratio of the minimum free energy of the consensus secondary structure of the given MSA ($MFE_{consensus}$), determined by RNAalifold (Hofacker et al., 2002), to the average Minimum Free Energy (MFE_{avg}) for the sequences in the MSA. An SCI score near 1 indicates that there is structural stability across the sequences in the MSA (Washietl et al., 2005). MFE_{avg} and $MFE_{consensus}$ are also retained in the feature pool as potential model parameters.

The next set of features is derived from the RNA folding programs RNAalifold (Hofacker et al., 2002), CentroidFold (Hamada et al., 2009), and RSpredict (Spirollari et al., 2009). RNAalifold implements the Zuker-Stiegler algorithm (Zuker, 1989) for

computing the Minimum Free Energy (MFE) structures of given sequences assuming a nearest-neighbour model. It uses empirical estimates of thermodynamic parameters for neighbouring interactions and loop entropies to score structures (Gardner and Giegerich, 2004). The MFE structure is a maximum likelihood estimator that provides the highest probability solution over the probabilistic distribution of all solutions (McCaskill, 1990). However, the MFE structure generally has a low probability of being the real structure and a number of other folding alternatives may have nearly the same probability. These alternatives may differ in the number of base pairs, which thus has an impact on the minimum free energy estimates and features such as SCI. Because of this, a better estimator of secondary structure may be one that maximises the expectations of an objective function related to the prediction accuracy.

CentroidFold (Hamada et al., 2009) provides an averaged gamma-centroid estimator for common secondary structure prediction. For MSAs, the estimator maximises the expected value of

$$\alpha_1 \mathrm{TP} + \alpha_2 \mathrm{TN} - \alpha_3 \mathrm{FP} - \alpha_4 \mathrm{FN}$$

where TP is the number of true positive base pairs, TN is the number of true negative base pairs, FP is the number of false positive base pairs, FN is the number of false negative base pairs, and α_i are user-determined constants.

RSpredict (Spirollari et al., 2009) computes the RNA structure with the minimum normalised free energy. The overall RNA structure is divided into substructures based on the loop decomposition scheme used in the nearest neighbour energy model. RSpredict accepts an MSA as input and predicts the consensus secondary structure of the MSA by minimising pseudo-energy. Pseudo-energy includes both normalised free energy and covariance scores of the sequences in the MSA.

For each of the RNA folding programs above, common features were chosen to characterise the predicted consensus secondary structure of the input MSA. These features include the number of base pairs, number of loops, number of bulges, hairpin length, and longest (maximum) consecutive base pairs. In addition, the normalised value of each feature, which equals the feature value divided by the multiple sequence alignment length, was also computed. These features are chosen according to previously published literature, cf. Yousef et al. (2006) and Hertel and Stadler (2006).

Other features included in the model's feature pool were energy density, or normalised free energy, from RSpredict, Shannon entropy, and base-pair distance. Shannon entropy and base-pair distance have been previously suggested as differentiating features (Freyhult et al., 2005), and both measure properties of the ensemble of structures that may exist for a given RNA sequence in vivo. Shannon entropy is defined as:

$$Q(x) = -\sum_{i<j} p_{ij} \log_2 p_{ij} / L$$

where x is the base-pair sequence, p_{ij} is the base-pair probability (i.e., the probability that base x_i pairs with base x_j) and L is the length of the given RNA sequence. Average base-pair distance is defined as:

$$D(x) = \sum_{i<j} (p_{ij} - p_{ij}^2) / L.$$

As our experimental results show later, Shannon entropy and average base-pair distance show a high degree of correlation.

2.2 Feature ranking

The key to feature ranking is to select features with the greatest potential for discriminating between positive and negative classes. We propose to use the Class Separation Measure (CSM) for selection and ranking of candidate features prior to classification. **CSM** is a *d*-dimensional feature vector:

$$\mathbf{CSM} = (\mathrm{CSM}^k), \quad \forall k \in \{1, 2, \ldots, d\}.$$

The CSM score for feature *k* is defined as:

$$\mathrm{CSM}^k = \frac{\left|\mu_1^k - \mu_2^k\right|}{(\sigma_1^k)^2 + (\sigma_2^k)^2}$$

where μ_1^k is the mean of the positive class for feature *k*, μ_2^k is the mean of the negative class for feature *k*, σ_1^k is the standard deviation of the positive class for feature *k*, and σ_2^k is the standard deviation of the negative class for feature *k*. The CSM score indicates the extent to which feature *k* separates the positive class from the negative class.

Once the CSM score has been calculated for all features, it is then sorted from the largest to the smallest value to create a ranked feature vector, X_{ranked}. The top ranked features can then be used in a supervised learning algorithm for prediction of positive or negative data. One drawback with this approach is that it does not consider the correlation of features. Its downside is that discriminating features that are highly correlated could be included in the feature set. Thus, we may be able to further reduce the feature set by eliminating a feature that is highly correlated with another. However, the class separation measure provides a useful starting point for feature selection where the feature set can then be reduced to a smaller set by considering feature correlation.

2.3 Supervised learning algorithms

Supervised learning, also known as classification, is a widely used data mining technique for predicting the label of an unknown object (Tan et al., 2005). We examined different machine learning or classification algorithms for prediction, including naïve Bayes (Tan et al., 2005), Fisher's Linear Discriminant (FLD) (Duda et al., 2001), and SVMs with different kernel functions (Fan et al., 2005; Wang and Wu, 2006). The FLD algorithm is good for linear classification of samples. SVMs provide a means of classifying data into different classes or categories. The goal of SVMs is to find a hyperplane with the maximum margin that separates two classes of data. By choosing this classification criterion, the number of false positives can be minimised and the impact of changes in the underlying model can be reduced.

Each value in the SVM classifier is represented by a tuple $(\mathbf{x}_i, y_i)$ where $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{id})^T$ corresponds to the feature vector for the *i*th training sample; y_i can be either 1 or –1 to denote the binary choice. The decision boundary of an SVM linear classifier has the form:

$$\mathbf{w} \bullet \mathbf{x} + b = 0$$

where **w** and b are parameters in the model. The boundary is determined from training data that has already been classified. For a linear model, the training data is used to set **w** and b (after scaling) such that:

$$\min f(\mathbf{w}) = \|\mathbf{w}\|^2 / 2$$

subject to $y_i(\mathbf{w} \bullet \mathbf{x}_i + b) \geq 1$. The parameters $\mathbf{w}$ and b must be chosen to meet the following conditions:

$\mathbf{w} \bullet \mathbf{x} + b \geq 1$ if $y_i = 1$ (i.e., for known ncRNAs)

$\mathbf{w} \bullet \mathbf{x} + b < 1$ if $y_i = -1$ (i.e., for known non-ncRNAs).

Two additional issues related to SVMs need to be addressed:

What if training data are not outside of margin because of their intrinsic noise? What if two classes cannot be separated by a line?

To handle the first issue, positive slack variables are added into the constraints of the f($\mathbf{w}$) optimisation such that:

$$\min f(\mathbf{w}) = \|\mathbf{w}\|^2 / 2 + C(\sum_{i=1}^{N} \xi_i)^k$$

subject to $y_i(\mathbf{w} \bullet \mathbf{x}_i + b) \geq 1 - \xi_i$ where $\xi_i \geq 0$ are the slack variables, and C and k represent penalties for erroneously classifying training samples.

To handle the second issue, we transform the data from its original space to a transformed space with a mapping function $\Phi(\mathbf{x})$ where there is a linear hyperplane between the two datasets. This mapping has the property:

$$K(\mathbf{u},\mathbf{v}) = \Phi(\mathbf{u}) \bullet \Phi(\mathbf{v}) = (\mathbf{u} \bullet \mathbf{v} + 1)^2$$

where K is a *kernel function* of vectors $\mathbf{u}$, $\mathbf{v}$. Only certain kernel functions can be used. Some common kernel functions include those previously described by Fan et al. (2005) and Wang and Wu (2006):

Polynomial: $K(\mathbf{x}_i, \mathbf{x}_j) = (\gamma \mathbf{x}_i^T \mathbf{x}_j + r)^d$

Radial basis function: $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2)$

where r, d, and γ are user-determined parameters. In the next section, we will compare the SVM algorithms, using linear, polynomial and radial basis function kernels, with the other classification algorithms.

3 Experiments and results

3.1 Experimental setup

The datasets used in our experiments are derived from the process described by Washietl and Hofacker (2004). Sequences from 12 ncRNA families were taken from the Rfam database (Griffiths-Jones et al., 2003), including 5S rRNA (RF00001), U2 spliceosomal RNA (RF00004), tRNA (RF00005), hammerhead ribozyme III (RF00008), RNAseP (RF00010), U3 snoRNA (RF00012), Signal Recognition Particle (SRP) RNA (RF00017), U5 spliceosomal RNA (RF00020), tmRNA (RF00023), group II catalytic intron (RF00029), microRNA mir-10 (RF00104), and U70 snoRNA (RF00156). Within each family, clusters are formed using NCBI's BLASTClust, requiring 60% similarity for a cluster. Clusters with only one sequence were eliminated. Multiple sequence alignments were formed from each cluster. Random combinations of 2, 3, and 4 sequences were created. A limit of 2000 combinations per cluster was set so no single cluster is over

represented in the test set. The resulting combinations were aligned, and only those with 65%–85% similarity were retained. Nine families had enough diversity to constitute a large enough sample, which included RF00001, RF00004, RF00010, RF00012, RF00017, RF00020, RF00023, RF00029, and RF00156. Three families – RF00005, RF00008, and RF00104 – did not generate a large enough pool of sequences and were not included. A random sample of around 1800 MSAs per family was chosen, with about 600 each from MSAs with 2, 3, and 4 sequences in each family. Negative datasets were derived from the MSAs by shuffling columns in the alignments using the RNAz program, rnazRandomiseAln.pl, while preserving length, base composition, gap pattern, and local conservation patterns. Both positive and negative data were divided equally into training and testing data sets.

The z-score is calculated for each MSA using the following procedure. The program RNAalifold (Hofacker et al., 2002), set to its standard parameters, is used to find the consensus structure and minimum free energy for the MSA. For the minimum free energy, the energy contributions of the single sequences in the MSA are averaged and added to a covariance term that accounts for compensatory mutations. To calculate the z-score, the program rnazRandomiseAln.pl is used to generate 100 random alignments, following the work of Washietl and Hofacker (2004). The z-score is then calculated as the MFE of the MSA obtained from RNAalifold (m) minus the mean of the MFEs of the random alignments (μ) divided by the standard deviation of the MFEs of the random alignments (σ), i.e., z-score = $(m - \mu)/\sigma$. The SCI, which measures structural conservation, is the ratio of the minimum free energy of the consensus secondary structure of the MSA, as calculated by RNAalifold, to the average minimum free energy of the individual sequences in the MSA. The consensus minimum free energy ($MFE_{consensus}$) and the average minimum free energy (MFE_{avg}) are evaluated as potential model parameters.

Folding features are derived from three different RNA folding programs: RNAalifold (Hofacker et al., 2002), CentroidFold (Hamada et al., 2009), and RSpredict (Spirollari et al., 2009). Common features are derived from the predicted consensus secondary structure for each MSA, which include:

- number of base pairs
- number of loops
- number of bulges
- hairpin length
- maximum consecutive base pairs.

Normalised values based on the length of the common features are computed. A customised program parses the structures for these features. Shannon entropy and base-pair distance are derived from the *–p* option of RNAalifold that provides the base-pair probabilities p_{ij} that base x_i pairs with base x_j. A customised program calculates these parameters.

The naïve Bayes and FLD classifiers were computed manually from the training data using Microsoft Excel 2007 matrix functions. Both algorithms are based on the probability distributions in the data sets used for this study. The support vector machine algorithms were taken from Libsvm (Fan et al., 2005). Model parameters were scaled to the interval [–1,1]. The penalty parameter k is set to 1. For the polynomial kernel,

the kernel is given by $K(\mathbf{x}_i, \mathbf{x}_j) = (\gamma \mathbf{x}_i^T \mathbf{x}_j + r)^d$ where $\gamma = 1$, $r = 1$, and $d = 2$. For the radial basis function kernel, the kernel is given by $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2)$. The values of γ and the penalty parameter C are determined by grid search using cross-validation with the grid.py utility supplied by Libsvm.

The measures used to evaluate the relative performance of the classifiers include sensitivity (Sn), specificity (Sp) and the Matthews Correlation Coefficient (MCC) (Matthews, 1975). Let *TP* be the number of true positives, *TN* be the number of true negatives, *FP* be the number of false positives, and *FN* be the number of false negatives. Sensitivity and specificity are defined as follows:

$$\mathrm{Sn} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}$$

$$\mathrm{Sp} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}}.$$

The Matthews correlation coefficient combines Sn and Sp into a single measure:

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

MCC ranges in value from –1 to +1, where +1 indicates a perfect prediction and –1 indicates a totally incorrect prediction.

We evaluate the supervised learning algorithms, which include naïve Bayes, FLD and SVMs with different kernel functions, for distinct feature sets using these performance measures.

3.2 Experimental results

The baseline feature set considered here contains the two features used by RNAz, namely the *z*-score and the SCI. Table 1 shows the results of these baseline feature sets for all classification algorithms. The overall MCC score for the naive Bayes classifier, across all families, was 86.88%. Considering the analysed ncRNA families separately, the U70 snoRNA family (RF000156) had the lowest score with 83.78%, whereas the RNAseP family (RF00010) had the highest score at 98.56%.

The FLD classifier shows a marked improvement over the naive Bayes classifier with an overall MCC score of 91.98%. The MCC scores for the nine different families range from 86.68% to 98.12%. With this classifier, the U70 snoRNA family (RF00156) continued to have the lowest score, and the RNAseP family (RF00010) had the highest.

Table 1 also provides the results of the baseline feature set using SVMs with different kernel functions. All the SVM classifiers used the Libsvm toolset (Fan et al., 2005) as the classifier platform. For the linear kernel, the overall MCC score is 92.80%. The MCC scores for the nine different families range from 86.56% to 99.56%. For the polynomial kernel (with Degree 2), the overall MCC score is 92.88%. The MCC scores of the nine different families range from 86.47% to 99.56%; this is very close to the linear kernel method. For the radial basis function kernel, the overall MCC score is 92.78%. The range of the nine families is 86.47% to 99.44%. For all kernel types, U70 snoRNA (RF00156) had the lowest score, and RNAseP (RF00010) had the highest score, consistent with the naive Bayes and FLD classifiers.

Table 1 A comparison of the sensitivities, specificities, and Matthews Correlation Coefficient (MCC) values of the classification algorithms for different ncRNA families using the baseline feature set

RNA family	*Method*	*Sn* (%)	*Sp* (%)	*MCC* (%)
RF00001	Naïve Bayes	97.10	95.09	92.21
	Fisher's Linear Discriminant	98.33	96.67	95.01
	SVM Linear	98.22	97.33	95.56
	SVM Polynomial	98.22	97.33	95.56
	SVM Radial Basis Function	98.00	97.67	95.67
RF00004	Naïve Bayes	91.00	98.56	89.81
	Fisher's Linear Discriminant	96.67	96.56	93.22
	SVM Linear	95.33	97.56	92.91
	SVM Polynomial	95.78	97.00	92.78
	SVM Radial Basis Function	96.33	96.11	92.44
RF00010	Naïve Bayes	99.89	98.67	98.56
	Fisher's Linear Discriminant	99.89	98.22	98.12
	SVM Linear	99.78	99.78	99.56
	SVM Polynomial	99.89	99.67	99.56
	SVM Radial Basis Function	99.78	99.67	99.44
RF00012	Naïve Bayes	95.13	94.68	89.81
	Fisher's Linear Discriminant	95.02	96.21	91.24
	SVM Linear	96.15	96.15	92.29
	SVM Polynomial	96.26	95.58	91.84
	SVM Radial Basis Function	97.96	96.71	94.68
RF00017	Naïve Bayes	92.81	98.66	91.63
	Fisher's Linear Discriminant	98.15	97.43	95.58
	SVM Linear	99.28	96.40	95.72
	SVM Polynomial	98.66	95.89	94.59
	SVM Radial Basis Function	98.97	97.23	96.21
RF00020	Naïve Bayes	97.10	95.09	92.21
	Fisher's Linear Discriminant	99.89	97.88	97.79
	SVM Linear	97.44	96.43	93.87
	SVM Polynomial	97.32	96.54	93.87
	SVM Radial Basis Function	97.21	96.88	94.09
RF00023	Naïve Bayes	98.00	98.22	96.22
	Fisher's Linear Discriminant	98.33	98.11	96.44
	SVM Linear	97.67	98.44	96.11
	SVM Polynomial	97.56	98.56	96.12
	SVM Radial Basis Function	98.56	98.33	96.89

Table 1 A comparison of the sensitivities, specificities, and Matthews Correlation Coefficient (MCC) values of the classification algorithms for different ncRNA families using the baseline feature set (continued)

RNA family	*Method*	*Sn* (%)	*Sp* (%)	*MCC* (%)
RF00029	Naïve Bayes	92.81	98.66	91.63
	Fisher's Linear Discriminant	97.51	97.02	94.53
	SVM Linear	96.72	98.17	94.89
	SVM Polynomial	96.27	98.08	94.36
	SVM Radial Basis Function	95.48	98.53	94.05
RF00156	Naïve Bayes	91.89	91.89	83.78
	Fisher's Linear Discriminant	94.22	92.44	86.68
	SVM Linear	93.67	92.89	86.56
	SVM Polynomial	94.33	92.11	86.47
	SVM Radial Basis Function	94.33	92.11	86.47
All	Naïve Bayes	91.15	95.64	86.88
	Fisher's Linear Discriminant	96.30	95.68	91.98
	SVM Linear	96.23	96.58	92.80
	SVM Polynomial	96.29	96.59	92.88
	SVM Radial Basis Function	96.21	96.56	92.78

Overall, the best classification method for the baseline feature set is the SVM classifier using the polynomial kernel with an overall MCC score of 92.88%. Still, the scores of all the SVM classification methods were close with a range of 92.78% to 92.88%.

From this baseline case, we seek to enhance the classification accuracy by incorporating additional features. To do this, in Table 2, we calculate the Class Separation Measure (CSM) score for each feature and sort the features – 44 in total – from the highest to the lowest score. Not surprisingly, the z-score was found at the top of the list of differentiating features, followed by normalised CentroidFold base pairs, SCI, Shannon entropy, base-pair distance, normalised RSpredict energy density, and CentroidFold maximum consecutive base pairs as the next six differentiating features. It is interesting to note that the CentroidFold program seems to be a better source of differentiation than RNAalifold or RSpredict. In addition, we note that some of these features are highly correlated (e.g., Shannon entropy and base-pair distance). A classifier may be simplified by choosing among highly correlated features.

We chose the top 11 features listed in Table 2 as the target feature set, and used them together with the different classification algorithms to determine the best method for ncRNA prediction. The results of the classification algorithms for the target feature set are shown in Table 3. This feature set shows substantial improvements over the baseline feature set for all the SVM classifiers. The best classification algorithm for this feature set is the SVM using the Radial Basis Function (RBF) kernel with an overall MCC score of 96.69%. The polynomial kernel has the next highest overall MCC score of 94.80% and the linear kernel has an overall MCC score of 94.29%. The range for the radial basis function kernel for the nine different families is 89.47% to 99.44%; for the polynomial kernel, 89.14% to 99.67%; and, for the linear kernel, 88.46% to 99.44%.

The FLD classifier had an overall MCC score of 93.16% across all families. The naive Bayes classifier had an overall MCC score of 85.01%.

Table 2 Class Separation Measure (CSM) values for all ncRNA features where the CSM score of a feature indicates the ability of the feature to differentiate between positive and negative classes

Feature	*CSM*
z score	4.365
normalised CentroidFold base pairs	2.094
SCI	1.709
Shannon entropy	1.488
base-pair distance	1.205
normalised RSpredict energy density	0.732
CentroidFold max consecutive base pairs	0.637
normalised CentroidFold hairpin length	0.617
normalised RSpredict base pairs	0.590
normalised RNAalifold base pairs	0.555
CentroidFold base pairs	0.505
RSpredict max consecutive base pairs	0.372
normalised RSpredict number of loops	0.266
normalised CentroidFold max consecutive base pairs	0.256
CentroidFold largest loop	0.246
consensus MFE	0.245
RSpredict number of loops	0.198
RNAalifold max consecutive base pairs	0.191
normalised RSpredict max consecutive base pairs	0.183
normalised RNAalifold hairpin length	0.181
normalised CentroidFold largest loop	0.176
normalised CentroidFold number of loops	0.157
normalised RNAalifold max consecutive base pairs	0.101
CentroidFold number of loops	0.070
normalised RNAalifold largest bulge	0.067
normalised CentroidFold largest bulge	0.055
CentroidFold largest bulge	0.053
RSpredict energy density	0.053
normalised RSpredict largest bulge	0.049
average MFE	0.041
CentroidFold hairpin length	0.041
RNAalifold largest bulge	0.040

Table 2 Class separation measure (CSM) values for all ncRNA features where the CSM score of a feature indicates the ability of the feature to differentiate between positive and negative classes (continued)

Feature	*CSM*
RNAalifold largest loop	0.039
RNAalifold base pairs	0.030
RSpredict base pairs	0.022
normalised RNAalifold number of loops	0.017
normalised RNAalifold largest loop	0.013
normalised RSpredict largest loop	0.008
RSpredict largest bulge	0.006
normalised RSpredict hairpin length	0.005
RSpredict largest loop	0.004
RNAalifold number of loops	0.003
RNAalifold hairpin length	0.002
RSpredict hairpin length	0.000

It is worthwhile to note that the naive Bayes classifier with the top 11 features has a lower overall MCC score than the baseline model (86.88%). This may be due to the fact that correlated features can degrade the performance of the naive Bayes classifiers because the assumption of independence no longer holds (Tan et al., 2005). We will further address the issue related to feature correlation in the Discussion section.

Table 3 A comparison of the sensitivities, specificities, and Matthews Correlation Coefficient (MCC) values of the classification algorithms for different ncRNA families using the target feature set

RNA family	*Method*	*Sn* (%)	*Sp* (%)	*MCC* (%)
RF00001	Naïve Bayes	98.00	99.89	97.91
	Fisher's Linear Discriminant	98.00	98.00	96.00
	SVM Linear	98.89	98.45	97.33
	SVM Polynomial	99.00	99.00	98.00
	SVM Radial Basis Function	98.89	99.33	98.22
RF00004	Naïve Bayes	94.56	95.19	89.78
	Fisher's Linear Discriminant	97.47	97.47	94.95
	SVM Linear	97.67	97.34	95.00
	SVM Polynomial	98.56	97.26	95.79
	SVM Radial Basis Function	98.78	98.02	96.78
RF00010	Naïve Bayes	99.22	98.13	97.34
	Fisher's Linear Discriminant	100.00	99.56	99.56
	SVM Linear	99.78	99.67	99.44
	SVM Polynomial	99.89	99.78	99.67
	SVM Radial Basis Function	99.78	99.67	99.44

Table 3 A comparison of the sensitivities, specificities, and Matthews Correlation Coefficient (MCC) values of the classification algorithms for different ncRNA families using the target feature set (continued)

RNA family	*Method*	*Sn* (%)	*Sp* (%)	*MCC* (%)
RF00012	Naïve Bayes	92.07	95.87	88.18
	Fisher's Linear Discriminant	97.39	96.84	94.21
	SVM Linear	93.43	94.94	88.46
	SVM Polynomial	93.77	95.28	89.14
	SVM Radial Basis Function	94.11	95.30	89.47
RF00017	Naïve Bayes	94.44	92.25	86.53
	Fisher's Linear Discriminant	99.07	98.97	98.04
	SVM Linear	99.59	98.57	98.15
	SVM Polynomial	99.38	98.97	98.35
	SVM Radial Basis Function	98.76	99.38	98.15
RF00020	Naïve Bayes	98.66	99.33	97.99
	Fisher's Linear Discriminant	98.66	98.22	96.88
	SVM Linear	99.22	98.56	97.77
	SVM Polynomial	99.22	98.78	97.99
	SVM Radial Basis Function	99.00	98.89	97.88
RF00023	Naïve Bayes	97.00	97.00	94.00
	Fisher's Linear Discriminant	98.56	97.79	96.34
	SVM Linear	98.11	98.66	96.78
	SVM Polynomial	99.11	99.00	98.11
	SVM Radial Basis Function	99.44	98.68	98.11
RF00029	Naïve Bayes	98.75	96.44	95.11
	Fisher's Linear Discriminant	97.74	97.30	94.99
	SVM Linear	98.87	97.54	96.36
	SVM Polynomial	99.43	97.56	96.94
	SVM Radial Basis Function	99.55	97.99	97.50
RF00156	Naïve Bayes	90.22	87.88	77.81
	Fisher's Linear Discriminant	95.44	94.60	90.00
	SVM Linear	95.44	95.23	90.67
	SVM Polynomial	95.44	95.23	90.67
	SVM Radial Basis Function	98.78	95.90	94.60
All	Naïve Bayes	93.93	91.31	85.01
	Fisher's Linear Discriminant	96.67	96.50	93.16
	SVM Linear	97.04	97.25	94.29
	SVM Polynomial	97.45	97.35	94.80
	SVM Radial Basis Function	98.52	98.17	96.69

Table 4 shows the classification results of RNAz using the default threshold of $P > 0.5$ for ncRNA prediction. The RNAz tool is pre-trained using a large set of cross-species ncRNAs taken from the Rfam database (Griffiths-Jones et al., 2003). The overall MCC score for RNAz is 86.71%. The classification scores for the nine different families range from 46.35% to 97.58%. With RNAz, the RNAseP family (RF00010) had the highest score whereas the tmRNA family (RF00023) had the lowest score. Comparing Tables 3 and 4, we can see that the SVM classifier with the radial basis function (RBF) kernel using the target feature set outperforms RNAz, except for U3 snoRNA (RF00012). This may happen due to the fact that RNAz, tuned to detect many types of ncRNA, is trained using ncRNA families with greater diversity, while the SVM classifier we developed here is trained specifically using the nine ncRNA families under analysis.

Table 4 A comparison of the sensitivities, specificities, and Matthews Correlation Coefficient (MCC) values for different ncRNA families using the RNAz prediction tool

RNA family	*Sn* (%)	*Sp* (%)	*MCC* (%)
RF00001	87.80	96.56	84.59
RF00004	97.67	97.78	95.44
RF00010	100.00	97.56	97.58
RF00012	99.00	97.96	96.98
RF00017	94.96	96.51	91.48
RF00020	94.33	99.11	93.53
RF00023	42.00	96.78	46.35
RF00029	99.21	96.23	95.45
RF00156	86.22	97.89	84.70

3.3 ROC curves and AUC

A Receiver Operating Characteristic (ROC) curve shows true positive rate (sensitivity) as a function of false positive rate (1-specificity) of a classifier (Tan et al., 2005). Each point on the ROC curve corresponds to one of the models induced by the classifier. We calculate the area under the ROC curve (AUC), which allows for evaluating which model is better on average (Tan et al., 2005). The larger AUC value a classifier has, the better that classifier is. Figure 1 shows the ROC curves for RNAz, the SVM with the radial basis function (RBF) kernel using the target feature set and baseline feature set respectively. (ROC curves for the other classifiers are not shown for space reasons.) Figure 2 compares the AUC values for RNAz, the classification algorithms with the baseline features and the classification algorithms with the top 11 CSM features for the nine Rfam families under analysis. The naïve Bayes classifier was not included due to its inferior performance. For the baseline features, the SVM algorithms are identical in performance, and appear to be slightly superior to the FLD model. All models (FLD and SVM) using the target feature set show improvements over the baseline model according to the AUC values in Figure 2. All the classification algorithms we developed, either with the top 11 CSM features or with the baseline feature set, outperform RNAz. The SVM algorithm with the RBF kernel using the target feature set tops the list with an AUC of 0.998.

Figure 1 ROC curves showing true positive rate as a function of false positive rate for RNAz, the SVM with the radial basis function kernel using the target feature set (Top11-rbf) and baseline feature set (Z-SCI-rbf) respectively (see online version for colours)

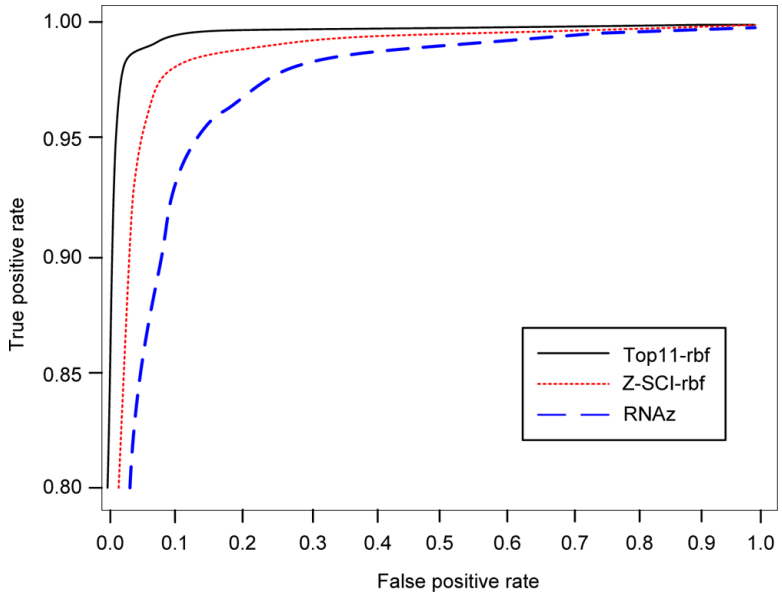
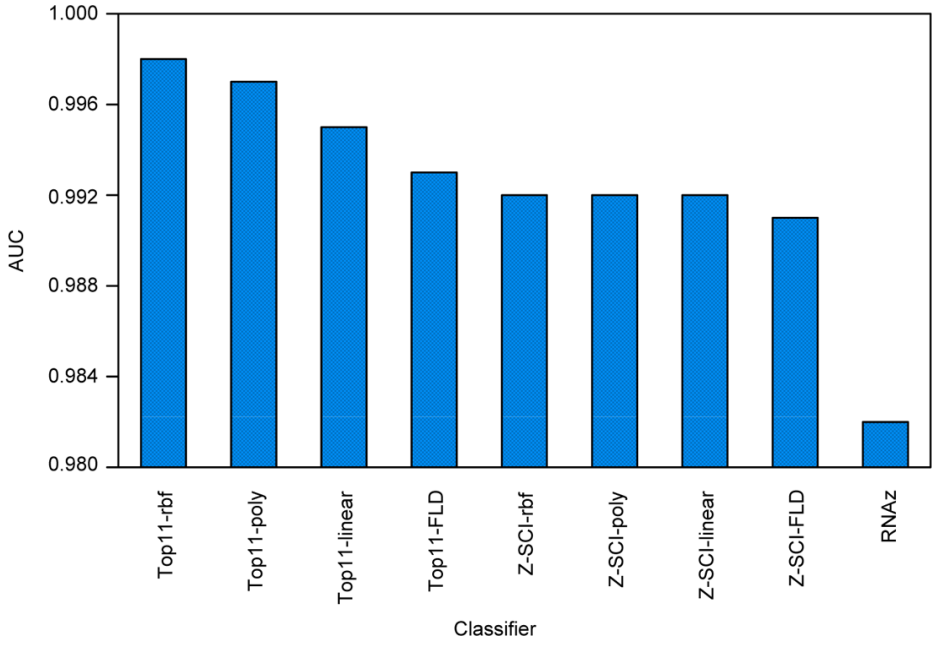


Figure 2 The AUC values of RNAz and the classifiers we developed with abbreviations as follows: the SVM with the polynomial kernel using the target feature set (Top11-poly) and baseline features (Z-SCI-poly), the SVM with the linear kernel using the target feature set (Top11-linear) and baseline features (Z-SCI-linear), Fisher's linear discriminant using the target feature set (Top11-FLD) and baseline features (Z-SCI-FLD) (see online version for colours)



4 Discussion and conclusions

Our experimental results indicated that incorporating additional features to classification algorithms can help to improve their performance. The top 11 features were chosen because their CSM scores exceeded an arbitrary CSM score of 0.5, cf. Table 2. There are opportunities to look beyond this feature set for further classification improvements. First, some of the top features chosen may be highly correlated. For example, Shannon entropy (the 4th feature or feature 4 in Table 2) and base-pair distance (the 5th feature or feature 5 in Table 2) are correlated with a correlation coefficient of 0.981, as shown in Table 5 in which the correlation coefficients for all top 11 features are listed. Freyhult et al. (2005) also made this observation. These two features (Shannon entropy and base-pair distance) have the highest correlation coefficient. Other correlated features include the normalised base pair measures: normalised RNAalifold base pairs, normalised CentroidFold base pairs, and normalised RSpredict base pairs. These highly correlated features may be removed to simplify the classification algorithms without greatly impacting their predictive power.

Table 5 Correlation coefficients for the features in the target feature set where the feature numbers correspond to the ordering in Table 2

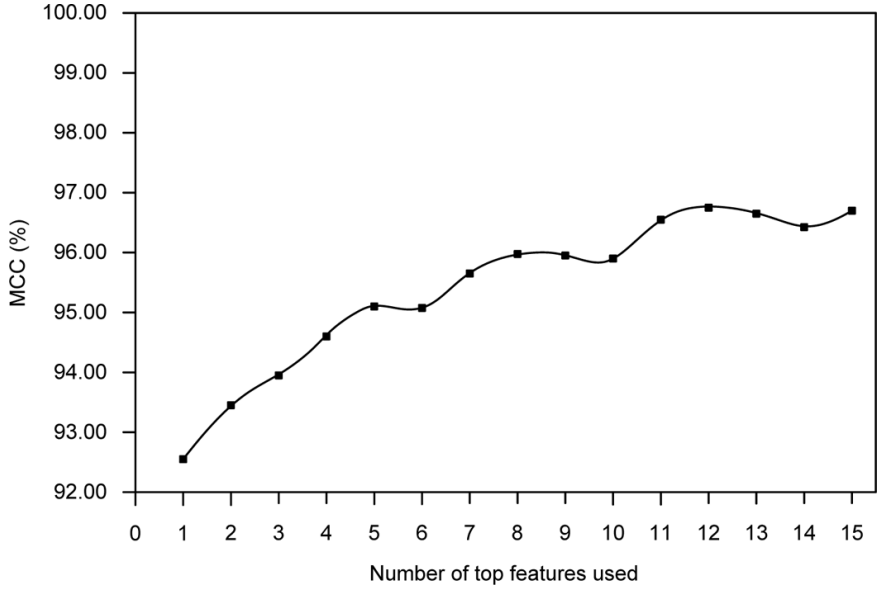
Feature	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*
1	1.000	–0.160	–0.612	0.481	0.497	0.268	–0.158	–0.242	–0.293	–0.219	–0.434
2	–0.160	1.000	0.391	–0.465	–0.378	–0.455	0.370	0.194	0.550	0.659	0.051
3	–0.612	0.391	1.000	–0.479	–0.459	–0.628	0.132	0.205	0.599	0.500	0.155
4	0.481	–0.465	–0.479	1.000	0.981	0.190	–0.134	–0.129	–0.307	–0.325	0.002
5	0.497	–0.378	–0.459	0.981	1.000	0.147	–0.108	–0.137	–0.252	–0.232	–0.019
6	0.268	–0.455	–0.628	0.190	0.147	1.000	–0.220	–0.266	–0.757	–0.595	–0.272
7	–0.158	0.370	0.132	–0.134	–0.108	–0.220	1.000	0.167	0.292	0.305	0.402
8	–0.242	0.194	0.205	–0.129	–0.137	–0.266	0.167	1.000	0.126	0.061	0.426
9	–0.293	0.550	0.599	–0.307	–0.252	–0.757	0.292	0.126	1.000	0.690	0.186
10	–0.219	0.659	0.500	–0.325	–0.232	–0.595	0.305	0.061	0.690	1.000	0.170
11	–0.434	0.051	0.155	0.002	–0.019	–0.272	0.402	0.426	0.186	0.170	1.000

In addition, it may be worthwhile to add additional features whose CSM scores are lower than 0.5 to boost the predictive power, although the additional features may have a diminishing impact and may cause an increase in false positives or negatives. To test this hypothesis, we ran the classification experiments, using the SVM algorithm with the RBF kernel, with the top n features for n from 1 to 15 as ranked by CSM scores shown in Table 2. Figure 3 shows the experimental results. The results indicate that the addition of features based on CSM scores improves prediction based on MCC – levelling off at the top 11 or 12 features.

Four of the 11 target features arise from CentroidFold parameters. This RNA folding program appears to provide greater differentiation of the positive and negative classes than the other RNA folding programs – although features from RNAalifold also populate the list. Many of the published ncRNA prediction methods use RNAfold or

RNAalifold for parameter generation. It may be advantageous to look to CentroidFold to provide predictive improvements.

Figure 3 MCC values for the SVM classifier with the radial basis function kernel using the top-ranked features in Table 2. The number *i* on the X-axis means that the top *i* features in Table 2 are used in the classifier



Both Shannon entropy and base-pair distance are included in the top five differentiating features according to CSM scores. These features were suggested by Freyhult et al. (2005). As stated above, these features are highly correlated. As Freyhult et al. (2005) pointed out, both features are computed using McCaskill base-pair probabilities. The measures Q and D show whether a sequence folds into a unique secondary structure or into several alternative structures (Mathews, 2004). Freyhult et al. (2005) also indicated that it would be sufficient to use only the *z*-score and Shannon entropy to predict how well an RNA folds. Our experimental results confirm the importance of Shannon entropy (and/or base-pair distance) for ncRNA predictions, but a number of other folding features from CentroidFold also contribute to the predictive power over the ncRNA families tested in this work.

In the study presented here, the classifiers including RNAz use the default thresholds provided in the underlying programs. One can use the Matthews correlation coefficient (MCC) to determine the optimal thresholds for these classifiers. Table 6 shows the optimal thresholds based on the MCC values for the SVM classifiers we developed. For the SVM algorithms with the baseline features, the optimal threshold is between 0.509 and 0.523. For the SVM algorithms with the top 11 features based on class separation measure (CSM) scores, the optimal threshold is between 0.526 and 0.545. Thus, it may be better to choose a slightly higher threshold for the models with the top 11 features. These threshold values may be used for subsequent testing and genome-wide discovery of ncRNAs.

Table 6 Thresholds to maximise MCC values for the different SVM algorithms using baseline and target feature sets

Feature set	*Model*	*Optimal threshold*	*MCC* (%)
Baseline	SVM with linear kernel	0.513	92.89
Baseline	SVM with polynomial kernel	0.523	92.94
Baseline	SVM with RBF kernel	0.509	92.89
Target	SVM with linear kernel	0.526	94.38
Target	SVM with polynomial kernel	0.530	95.05
Target	SVM with RBF kernel	0.545	96.81

Our empirical study indicates that the SVM algorithm with the RBF kernel using the target feature set determined by the class separation measure is a superior predictor of ncRNA when compared with other classification algorithms and RNAz for the set of ncRNA families tested in the study reported here. The techniques used by this SVM classifier can be expanded to other ncRNA families and refined by including additional features. The SVM classifier can be used to search multiple sequence alignments from genomic data for novel ncRNAs. Moreover, the classifier might be able to be modified to search for particular families of ncRNAs by incorporating different features into its algorithm as the class separation measure varies with different ncRNA families.

Acknowledgment

We thank the anonymous reviewers for their constructive suggestions, which helped improve the presentation and content of this paper.

References

Babak, T., Blencowe, B.J. and Hughes, T.R. (2007) 'Considerations in the identification of functional RNA structural elements in genomic alignments', *BMC Bioinformatics*, Vol. 8, No. 33.

Chang, S., Wen, S., Chen, D. and Jin, P. (2009) 'Small regulatory RNAs in neuro developmental disorders', *Human Molecular Genetics*, Vol. 18, No. 1, pp.18–26.

Coventry, A., Keitman, D.J. and Berger, B. (2004) 'MSARI: multiple sequence alignments for statistical detection of RNA secondary structure', *Proc. of the National Academy of Sciences USA*, Vol. 104, No. 33, pp.12102–12107.

Di Bernardo, D., Down, T. and Hubbard, T. (2003) 'ddbRNA: detection of conserved secondary structures in multiple alignments', *Bioinformatics*, Vol. 9, No. 13, pp.1606–1611.

Duda, R.O., Hart, P.E. and Stork, D.G. (2001) *Pattern Classification*, 2nd ed., John Wiley & Sons, Inc., New York, NY, USA.

Fan, R-E., Chen, P-H. and Lin, C-J. (2005) 'Working set selection using the second order information for training SVM', *Journal of Machine Learning Research*, Vol. 6, pp.1889–1918.

Freyhult, E., Gardner, P.P. and Moulton, V. (2005) 'A comparison of RNA folding measures', *BMC Bioinformatics*, Vol. 6, No. 241.

Gardner, P.P. and Giegerich, R. (2004) 'A comprehensive comparison of comparative RNA structure prediction approaches', *BMC Bioinformatics,* Vol. 5, No. 140.

Gesteland, R.F., Atkins, J.F. and Cech, T.R. (Eds.) (2005) *The RNA World*, 3rd ed., Cold Spring Harbor Laboratory Press, New York, NY, USA.

Griffiths-Jones, S., Bateman, A., Marshall, M., Khanna, A. and Eddy, S.R. (2003) 'Rfam: an RNA family database', *Nucleic Acids Research*, Vol. 31, No. 1, pp.439–441.

Hamada, M., Kiryu, H., Sato, K., Mituyama, T. and Asai, K. (2009) 'Prediction of RNA secondary structure using generalised centroid estimators', *Bioinformatics*, Vol. 25, No. 4, pp.465–473.

Hertel, J. and Stadler, P.F. (2006) 'Hairpins in a haystack: recognising microRNA precursors in comparative genomics data', *Bioinformatics*, Vol. 22, pp.e197–e202.

Hofacker, I.L. (2003) 'Vienna RNA secondary structure server', *Nucleic Acids Research*, Vol. 31, pp.3429–3431.

Hofacker, I.L., Fekete, M. and Stadler, P.F. (2002) 'Secondary structure prediction for aligned RNA sequences', *J. Mol. Biol*., Vol. 319, pp.1059–1066.

Mathews, D.H. (2004) 'Using an RNA secondary structure partition function to determine confidence in base pairs predicted by free energy minimisation', *RNA,* Vol. 10, pp.1178–1190.

Matthews, B.W. (1975) 'Comparison of the predicted and observed secondary structures of T4 phage lysozyme', *Biochim. Biophys. Acta*, Vol. 405, pp.442–451.

McCaskill, J.S. (1990) 'The equilibrium partition function and base pair binding probabilities for RNA secondary structures', *Biopolymers*, Vol. 29, pp.1105–1119.

Menzel, P., Gorodkin, J. and Stadler, P.F. (2009) 'The tedious task of finding homologous noncoding RNA genes', *RNA*, Vol. 15, pp.2075–2082.

Pedersen, J.S., Bejerano, G., Siepel, A., Rosenbloom, K., Lindblad-Toh, K., Lander, E., Rogers, J., Kent, J., Miller, W. and Haussler, D. (2006) 'Identification and classification of conserved RNA secondary structures in the human genome', *PLoS Computational Biology*, Vol. 2, No. 4, p.e33.

Rivas, E. and Eddy, S.R. (2001) 'Noncoding RNA gene detection using comparative sequence analysis', *BMC Bioinformatics*, Vol. 2, No. 8.

Spirollari, J., Wang, J.T.L., Zhang, K., Bellofatto, V., Park, Y. and Shapiro, B.A. (2009) 'Predicting consensus structures for RNA alignments via pseudo-energy minimisation', *Bioinformatics and Biology Insights*, Vol. 3, pp.51–69.

Tan, P-N., Steinbach, M. and Kumar, V. (2005) *Introduction to Data Mining,* Addison-Wesley, New York, NY, USA.

Uzilov, A.V., Keegan, J.M. and Mathews, D.H. (2006) 'Detection of noncoding RNAs on the basis of predicted secondary structure formation free energy change', *BMC Bioinformatics*, Vol. 7, No. 173.

Valadkhan, S. and Nilsen, T.W. (2010) 'Reprogramming of the non-coding transcriptome during brain development', *Journal of Biology*, Vol. 9, No. 5.

Wang, J.T.L. and Wu, X. (2006) 'Kernel design for RNA classification using support vector machines', *International Journal of Data Mining and Bioinformatics*, Vol. 1, pp.57–76.

Washietl, S. and Hofacker, I.L. (2004) 'Consensus folding of aligned sequences as a new measure for the detection of functional RNAs by comparative genomics', *J. Mol. Biol.*, Vol. 342, pp.19–30.

Washietl, S., Hofacker, I.L. and Stadler, P.F. (2005) 'Fast and reliable prediction of noncoding RNAs', *Proc. of the National Academy of Sciences USA*, Vol. 102, No. 7, pp.2454–2459.

Yousef, M., Nebozhyn, M., Shatkay, H., Kanterakis, S., Showe, L.C. and Showe, M.K. (2006) 'Combining multi-species genomic data for microRNA identification using a naïve Bayes classifier', *Bioinformatics*, Vol. 22, No. 11, pp.1325–1334.

Zuker, M. (1989) 'On finding all suboptimal foldings of an RNA molecule', *Science*, Vol. 244, pp.48–52.