BioChip J. (2017) 11(2): 164-171
DOI 10.1007/s13206-017-1210-3

Original Article

miRHunter: A Tool for Predicting microRNA Precursors Based on Combined Computational Method

Insong Koh[1] **& Ki-Bong Kim**[2,*]

Received: 25 December, 2016 / Accepted: 25 January, 2017 / Published online: 15 May, 2017

Abstract MicroRNAs (miRNAs) are small endogenous non-coding RNAs known to post-transcriptionally regulate gene expression in a broad range of organism. Since the discovery of the very first miRNAs, *lin-4* and *let-7*, computational methods have been indispensable tools that complement experimental approaches to understand the biology of miRNAs. In this article, we introduce a web-based computational tool, miRHunter, that identifies potential miRNA precursors (pre-miRNAs) in the genomic sequences by using a combined computational method. The method coupled *ab initio* method with homology-based and hairpin structure-based methods. The miRHunter consists of five modules: 1) a preprocessing module, 2) an evolutionary conservation filter module, 3) a hairpin structure filter module, 4) a support vector machine module that evaluates preliminary pre-miRNA candidates derived from the previous two filtering modules, and 5) a post-processing module. The miRHunter system yielded the following average test results: 96.16%/93.23%, 96.00%/94.68%, and 95.87%/93.57% which are sensitivity (Sn) and specificity (Sp) for animal, plant, and overall categories respectively. The miRHunter system can complement experimental methods and allow wet-lab researchers to screen long sequences for putative miRNAs as well as pre-testing miRNAs of interest. The microarray profiling experiments have supported that the clusters of proximal pairs of miRNAs are generally coexpressed. Therefore, the clustering or spatial localization information will be used to improve the accuracy of our system in further work. The miRHunter is available at http://www.bioinfoworld.com/.

Keywords: MicroRNAs, Gene expression, miRHunter, Combined computational method, Support vector machine

[1]Department of Physiology, College of Medicine, Hanyang University, Seoul, Republic of Korea
[2]Department of Biomedical Technology, Sangmyung University, Cheonan, Republic of Korea
*Correspondence and requests for materials should be addressed to K.-B. Kim (✉kbkim@smu.ac.kr)

Introduction

MicroRNAs are a family of ~22-nucleotide small RNAs that regulate gene expression at the post-transcriptional level and play important roles in a range of processes including development, differentiation, cell growth, cell proliferation, apoptosis and tumorigenesis[1,2]. They regulate the expression of protein coding genes by base-pairing with the transcripts of their target genes, subsequently leading to translational repression[2,3], mRNA cleavage[4] or miRNA-induced degradation[5]. Currently, 28,645 entries representing hairpin precursor miRNAs, expressing 35,828 mature miRNA products, in 223 species have been deposited in the release 21 of miRBase database[6]. Most miRNAs have been identified using experimental techniques such as molecular cloning, Northern Blot or real-time PCR. MicroRNA expression profiling experiments, using microarray or deep sequencing, has proven to be useful to identifying miRNAs that are preferentially expressed in a range of biological processes. More recently, high throughput sequencing (i.e., RNAseq) has been employed as a means of identifying larger numbers of miRNAs. However, the experimental methods are limited by time consuming, high cost, and low efficiency.

An additional limitation is that they are intrinsically biased towards miRNAs which are highly expressed and miRNAs expressed at low levels or in limited cell types may not be easily discovered[7]. As an alternative, computational approaches have been developed to complement experimental methods and accelerate the understanding of miRNA biology. In practice, they have been contributing a lot to affording the complexity of finding putative miRNA genes and their targets, as well as to determining their functions and regulatory mechanisms.

The principles of computational approaches for miRNA or miRNA precursor (pre-miRNA) discovery are based on the major characteristics or features of miRNAs or pre-miRNAs: *structural and thermodynamic features* such as hairpin length, hairpin-loop length, bulge size and location, base-pairing, thermodynamic stability, and distance of the miRNA from the loop of its hairpin precursor, *sequence and genomic features* such as nucleotide content and location, repeat elements, sequence complexity, clustering property, and internal and inverted sequence repeats, and *evolutionary conservation features* such as conserved motif and signature, sequence and structure similarity, and evolutionarily biased sequence composition, from species to species. That is, most existing methods use distinctive properties of known miRNAs as criteria to search for unknown miRNAs. The computational approaches can be classified into three main categories: *comparative and homology-based approaches, ab-initio or non-comparative approaches, and integrated approaches*[8].

Comparative and homology-based approaches use the phylogenetic conservation of pre-miRNAs in their primary sequence and/or their secondary structure to predict new miRNA genes similar to known miRNAs. While these methods are effective for identifying miRNAs from the sequence that are closely conserved in related species, they are less effective for discovering miRNAs in more divergent sequences. The rapid accumulation of known pre-miRNAs in miRBase makes these approaches more powerful. *Ab-initio* methods use only a computational predictive model without extrinsic comparison to existing data, in order to make predictions about biological features of miRNAs. In other words, they make an attempt to identify inherent structural or compositional features of miRNA sequences that can be used to distinguish putative miRNAs from a broader range of candidate sequences. A majority of *ab-initio* methods are based on supervised machine learning techniques that use positive and negative hairpin-shaped precursors as training data set, in order to train a classifier, such that it can identify putative novel miRNAs in 'unseen' sequences. Typically, the inputs for the classifier are a set of features describing a candidate miRNA and the outputs are indicators showing whether the candidate is a genuine miRNA or not. The supervised machine learning algorithms usually are support vector machine (SVM)[9], neural networks[10], hidden Markov model (HMM)[11], and Naïve Bayes (NB)[12]. The drawback is that the methods are computationally intensive and very time-consuming. In recent years, experimental data-driven methods have become the driving force for the discovery of novel miRNA genes. Next generation sequencing and experimental technologies have opened the door to pinpoint known and novel miRNAs. In this context, some approaches, called *integrated approaches*, have combined high-throughput experimental methods with computational approaches to identify a broader range of miRNAs.

In order to complement respective disadvantages of *comparative and homology-based* method and *ab initio* method, in this work, we have developed the miRHunter system that combines the two methods to identify potential pre-miRNAs. Our system uses evolutionary conservation and hairpin structure as preliminary filters to select the preliminary pre-miRNA candidates, thereby allowing relatively large numbers of false-positives followed by a support vector machine classifier to eliminate the false-positive candidates. The evolutionary conservation filter is very powerful since the overwhelming majority of pre-miRNAs show high sequence conservation against the background of non-conserved DNA. The hairpin structure filter is also very effective because all the pre-miRNAs form hairpin structure in biogenesis. However, this filter allows relatively large numbers of false-positives since hairpin secondary structures are common motifs in other types of non-coding RNAs. As for SVM, we first reviewed experimental studies that identified physical characteristics of the pre-miRNA. Based on these characteristics, we defined a number of parameters to describe a sequence and used them as input to a SVM. We trained the SVM against positive and negative data and evaluated this model through 5-fold cross-validation to evaluate predictive ability. Furthermore, we totally evaluated the performance of the miRHunter system through 5-fold cross-validation in terms of sensitivity and specificity. The miRHunter system has the capability of not only recognizing pre-miRNAs for which close homologous cannot be found due to the limitation of current data, or especially due to the availability of rapidly evolving and species-specific miRNAs, but also to be capable of recognizing conserved or homologous miRNAs.

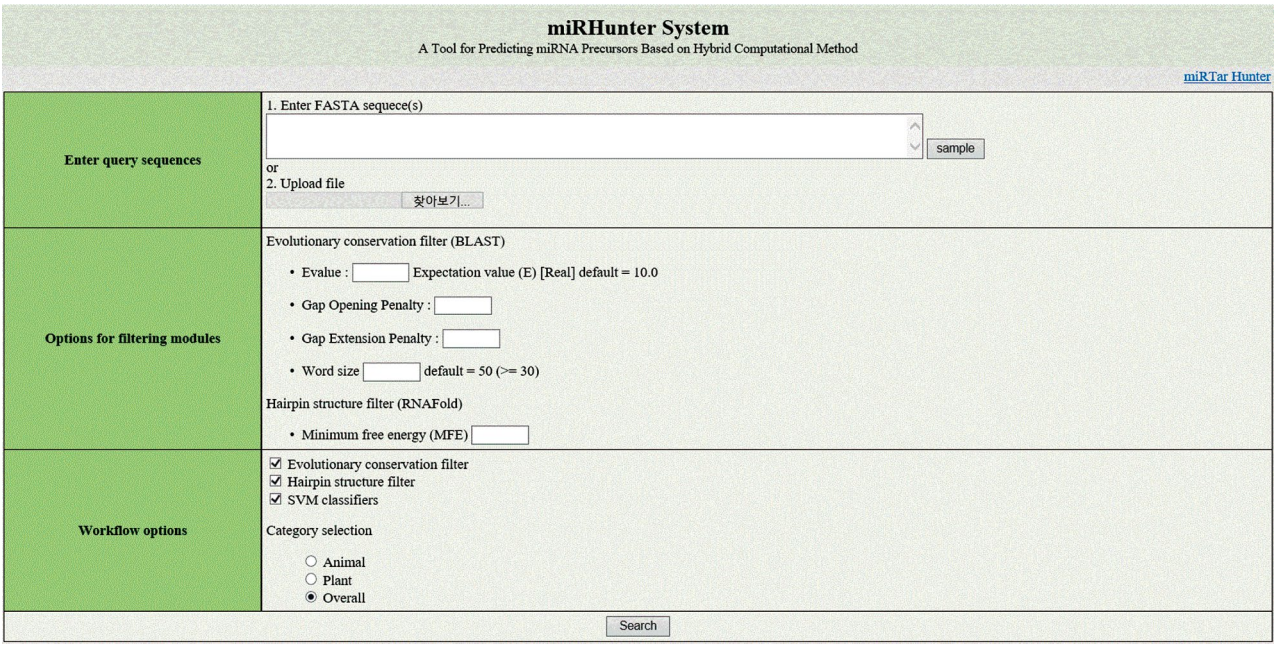

Figure 1. User-friendly web interface of miRHunter. This web interface has three main sections: "Enter query sequences", "Options for filtering modules", and "Workflow options". The web interface is designed to allow customization of various options/parameters for tailored analysis.

Results and Discussion

miRHunter Web Interface and it Diverse Options/ Parameters for Customized Analysis

Figure 1 shows the web interface of the miRHunter system, which consists of three main sections: "Enter query sequences", "Options for filtering modules", and "Workflow options". The "Enter query sequences" section is for entering either the query sequences in multi-FASTA format directly into a textbox or for uploading a file of the query sequences. The "Options for filtering modules" section allows users to input the parameters for evolutionary conservation filter and hairpin structure filter. The evolutionary conservation filter has four parameters: expectation value, gap opening penalty, gap extension penalty, and the word size. The hairpin structure filter has the parameter for minimum free energy. The "Workflow options" section allows users to select modules and category used for their own analyses. All options and parameters provided can be specified by the user to be tailored to his/her intent. The web interface was designed to allow users to perform customized analyses.

Performance Evaluation of the SVM Classifiers through the 5-fold Cross-validation

Using 5-fold cross-validation, as the first experiment, we evaluated three corresponding SVM classifiers for animal, plant, and overall categories respectively. Each classifier was trained with its training dataset to observe the classification performance with its validating dataset by using 29 features mentioned in the materials and methods. For each category, the complete dataset for corresponding category was first partitioned into 5 equally (or nearly equally) sized segments or folds. Subsequently 5 iterations of training and validation were performed such that within each iteration a different fold of the data was held-out for validation while the remaining 4 folds were used for learning. The classification results on the testing datasets were averaged. The same performance evaluation procedure was repeated three times with three separate classifiers for animal, plant, and overall. Data was stratified prior to being split into 5 folds. Stratification is the process of rearranging the data as to ensure each fold is a good representative of the whole. This initial experiment yielded the following average test classification results: 92.77%/90.03%, 92.88%/92.08%, and 93.07%/90.17% which are sensitivity (Sn) and specificity (Sp) for animal, plant, and overall categories respectively. The results of the 5-fold cross-validation are summarized in Table 1. It was obvious that the resulted classifiers performed a little poorly with respect to the negative class compared with the positive one in case of animal

Table 1. Cross validation results of three SVM classifiers for the animal, plant, and overall categories. All three classifiers demonstrated good prediction capability. The last row contains the average values for all the tests. In 5-fold cross-validation, we first divided the training set into 5 subsets of equal size. Sequentially one subset was tested using the classifier trained on the remaining 4 subsets in terms of sensitivity (Sn) and specificity (Sp).

Testing No.	Category					
	Animal		Plant		Overall	
	Sn (%)	Sp (%)	Sn (%)	Sp (%)	Sn (%)	Sp (%)
1	92.25	88.75	93.15	91.78	92.95	90.21
2	93.24	90.21	92.89	91.95	93.25	89.54
3	92.17	90.57	93.68	92.65	92.85	90.86
4	93.15	89.95	92.45	91.78	93.43	89.79
5	93.05	90.65	92.23	92.25	92.89	90.43
Average	92.77	90.03	92.88	92.08	93.07	90.17

and overall categories. That is, these classifiers developed with our imbalanced dataset (21,280/9,248 and 28,337/9,248 that are positives and negatives for animal and overall categories respectively) were biased towards the majority positive class (Sn > > Sp).

Performance Evaluation of the miRHunter System

With 5-fold cross-validation mentioned above, as the second experiment, we evaluated the overall performance of the miRHunter system for animal, plant, and overall categories respectively in terms of a single analysis unit. The same performance evaluation procedure applied to SVM was implemented on this experiment except that all components of the miRHunter system were regarded as one sequential unit which was asked to make predictions about the data in the validation fold. According to the specified default parameter set (word size = 30, expect value = 10.0, gap opening penalty = 5, and gap extension penalty = 2), the evolutionary conservation filter module that searches for preliminary pre-miRNA candidates using the BLAST program[13] was run to hit preliminary pre-miRNAs. At default temperature (37°C), hairpin structures that minimize their free energy were computed by RNAFold program[14] in hairpin structure filter module, which can be used to predict the minimum free energy (MFE) secondary structure of single sequences using the dynamic programming algorithm. Particularly, the logical disjunction between evolutionary conservation filter and hairpin structure filter was applied to yield their preliminary pre-miRNA candidates that were finally classified by SVM classifiers. This experiment yielded the following average test results: 96.16%/93.23%, 96.00%/94.68%, and 95.87%/93.57% which are sensitivity (Sn) and specificity (Sp) for animal, plant, and overall categories respectively. The results of this experiment are summarized in Table 2. The performance evaluation results show that the combined method performs better than otherwise used separately. At the same reason with the previous experiment for SVM classifiers, the miRHunter system performed a little poorly with respect to the negative class compared with the positive one as to animal and overall categories. The class balance learning may lead to better performance with respect to both positive and negative classes.

Table 2. Cross validation results of the miRHunter system for the animal, plant, and overall categories. Compared to the existing methods - miPred and microPred, the miRHunter system performs much better with regard to sensitivity whereas it falls a bit short of them in terms of specificity. The last row contains the average values for all the tests.

Testing No.	Category					
	Animal		Plant		Overall	
	Sn (%)	Sp (%)	Sn (%)	Sp (%)	Sn (%)	Sp (%)
1	95.65	93.75	95.75	94.78	95.95	93.21
2	96.24	92.21	95.89	93.95	95.25	93.54
3	95.67	93.57	96.68	94.65	95.85	93.86
4	96.78	92.95	95.45	94.78	96.43	93.79
5	96.45	93.65	96.23	95.25	95.89	93.43
Average	96.16	93.23	96.00	94.68	95.87	93.57

Compared to the existing representative methods-miPred[15], microPred[16], and MiRNAClassify[17], our miRHunter system performs much better with regard to sensitivity except for MiRNAClassify. However, our method falls a bit short of the existing methods when it comes to specificity. The performance of miPred and microPred has, according to their report, 92.08% (Sn)/97.42% (Sp) and 92.71% (Sn)/94.24% (Sp) respectively. MiRNAClassify is a pre-miRNA classification method based on cost-sensitive ensemble learning. Through a series of iterations, the information of all the positive and negative samples is completely exploited. In the iteration, a new classification instance is trained by the equal number of positive and negative samples, which can relieve the negative effect of class imbalance. The average Sn and Sp demonstrated by the cross-validation of MiRNAClassify are 95.7 and 97.9 respectively. A direct comparison of our algorithm with existing algorithms is challenging, because the different published methods are based on different principles, making it difficult to an accurate comparison. In addition, many methods are not available for download for independent testing on a common dataset, whereas the datasets used by these methods are

highly diverse. MicroRNA expression profiling experiments, using microarray or HST (deep sequencing or RNASeq), have proven that clusters of proximal miRNAs are generally expressed as polycistronic and coregulated units. The clustering of putative proximal miRNA precursors predicted by miRHunter will be used to improve the accuracy of the system in further work.

Conclusion

With the advent of high throughput sequencing (HTS), there is a need for miRNA prediction tool that can support such studies by analyzing large numbers of genomic sequences in a reasonable time and which can be applied to a broad range of species. Besides, the computational approaches for miRNA discovery can complement experimental studies by (i) identifying additional putative miRNAs that can be missed by experimental methods and (ii) in case of HTS experiments, serving as a useful pre-sequencing step to determine the possible yield from such an experiment. In this context, we have developed the miRHunter system for pre-miRNA discovery from genomic sequence. It is primarily designed for use in wet-lab studies to screen genomic sequences for putative miRNAs as well as pre-testing miRNAs of interest to reduce search range. Our system has three major characteristics. First, coupled with homology-based and secondary structure based methods, non-comparative *ab-initio* method was employed to identify pre-miRNAs. Second, instead of training a single classifier for all sequence categories, we trained three separate SVM classifiers for animal, plant, and overall categories respectively. Finally, the complete pre-miRNA data in miRBase, except for viral data, were used to train separate classifiers, rather than restricting ourselves to subsets of the data.

The miRHunter system could be used to predict novel pre-miRNAs in both comparative and non-comparative ways. The comparative and hairpin structure-based prediction is straightforward, while the non-comparative *ab initio* prediction requires additional features analysis. Under comparative and hairpin structure based prediction, miRHunter can be first used to predict whether any region of a genomic sequence is falling into a hairpin secondary structure and/or conservation category or not. If it is predicted as a candidate pre-miRNA hairpin, then it can be further examined and classified by SVM. The BLAST program, a popular sequence similarity search tool, was used for sequence conservation analysis. The RNAFold program was adopted in order to find the structure conservation. There has been much debate as to whether each feature or parameter used in SVM is critical for miRNA processing. However, the SVM is combining these parameters in the training process, rather than analyzing them independently. Importantly, it would be worth trying to incorporate the advanced features identified in the profiling experiment researches with the microarray or deep-sequencing data and signals to develop a better computational method for miRNA discovery. Parameter filtering step is necessary in further study and beyond the scope of this study. In this study, 29 features were introduced in our system. However, selecting the most discriminative set of features would increase the performance, efficiency and comprehensibility of a classifier system by reducing its complexity. Not only all three classifiers but also our whole system showed good performance with high specificity and sensitivity. When used in combination with experimental methods, the miRHunter can provide an additional validation layer for putative miRNAs and have achieved high sensitivity in a genome analysis.

Materials and Methods

Overall Configuration of miRHunter System and its Prediction Pipeline

miRHunter is a web-based system that runs on an Apache web server with a Linux operating system 2.6.18. It has a three-tier architecture composed of a client, an application server, and a back-end database. A schematic overview of the miRHunter system is shown in Figure 2. The client is a user-friendly web interface for the application server. The application server consists of preprocessing module, preliminary filters (evolutionary conservation filter and hairpin structure filter), SVM classifiers, and post-processing module. The back-end database is a secondary database that contains a collection of the pre-miRNAs derived from miRBase database[6]. The prediction pipeline of miRHunter system allows researchers to analyze all possible scenarios to infer pre-miRNA candidates. First, preprocessing is performed on parameters/options and input query sequence to prepare for the following procedures, that is, evolutionary conservation filter and hairpin structure filter. For the evolutionary conservation filter, BLAST program[13] is used to compare query sequence with the back-end database (or BLAST DB) and identify any region of query sequence that has similarity with pre-miRNAs in the back-end database above a certain threshold. All the hits by BLAST are classified as preliminary pre-miRNA candidates and

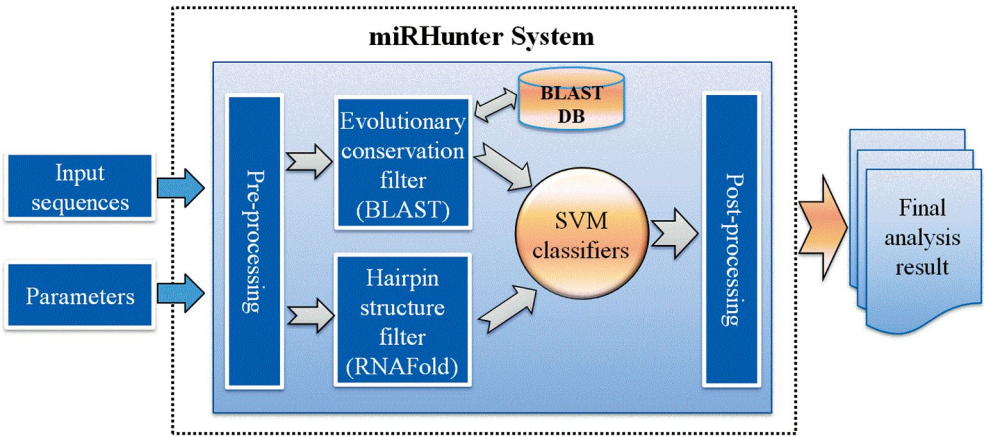

Figure 2. Schematic overview of the prediction pipeline in the miRHunter system. The miRHunter has a three tier architecture consisting of a web interface, application server, and back-end database. The back-end database contains a collection of the pre-miRNAs extracted from the miRBase database. The application server consists of a preprocessing module for input query sequences and parameters, evolutionary conservation filter module to search for preliminary pre-miRNA candidates using BLAST program, hairpin structure filter module to search for hairpin secondary structure of preliminary pre-miRNA candidate, a SVM module to evaluate preliminary pre-miRNA candidates derived from two filter modules, and a post-processing module to display the final analysis results in a web-based browser.

passed in turn to the SVM for classification. For the hairpin structure filter, RNAfold program[14] is used to identify any region of query sequence that can form hairpin structure like pre-miRNA. The hairpin structure filter slides a window of adjustable size over the input query sequence, advancing each window by a given step size. Hairpins with a size above a certain threshold are then identified as preliminary pre-miRNA candidates and passed in turn to the SVM for classification. The preliminary pre-miRNA candidates predicted by the two filters are classified by the SVM classifier as positives and negatives, and the final analysis results are then sorted, ranked, and displayed on the web interface. The web interface is implemented on a Linux server using PHP scripting. A combination of PHP scripts were used for the sequence preparation, SVM training and pre-miRNA predictions with the trained model. The core module of miRHunter, a probabilistic co-learning model, is written in Java version 1.4.2. It uses the library of the program 'RNAFold' to predict the folding of primary RNA sequence (Vienna RNA package version 1.6). The system runs on two dual 2.4 GHz Intel Xeon CPUs with 24 GB RAM module.

BLAST Subject Database and SVM Training Dataset

Excluding 308 viral pre-miRNAs, we classified the total dataset of 28,337 pre-miRNAs in the release 21 of miRBase database into animal (21,280), plant (7,057), and overall categories (28,337) and made BLAST subject database for each category, which the BLAST program in the evolutionary conservation filter module compares input query sequences against. In addition, each category dataset classified into animal, plant, and overall were used as the positive dataset for three corresponding SVM classifiers. The SVM classifier for overall category was trained from all pre-miRNAs and used for predictions on sequences belonging to any category. We also obtained 8,494 non-redundant pseudo hairpin sequences and 754 non-redundant other ncRNA sequences which have been previously used in microPred[16] method. Originally, these pseudo hairpins were extracted from RefSeq genes[18] without going through any experimentally validated alternative splicing event. Therefore, it is more likely that these pseudo hairpin sequences do not contain any annotated or un-annotated pre-miRNA sequences. These data were used as negative data for all three corresponding SVM classifiers without distinction of any category.

Features Selection and Support Vector Machines

Pre-miRNAs have many features about both primary sequence and secondary structure, which are typically employed to make a classifier to classify the real pre-miRNAs and pseudo pre-miRNAs hairpin sequences. Inquiring into the features used by the existing pre-miRNA classification methods, we considered the 29 'global and intrinsic' features introduced in miPred[15] and microPred[16] approaches, which can be calculated irrespective of the type of the secondary structures of

sequences. These features are: 1) 17 base composition variables that are 16 dinucleotide frequencies XY% where X,Y∈{A,C,G,U}, and (G+C)% content; 2) 6 folding measures - adjusted base pairing propensity (dP), adjusted minimum free energy (MFE) of folding (dG), adjusted base pair distance (dD), adjusted Shannon entropy (dQ), MFE index 1 (MFEI1), and MFE index 2 (MFEI2); 3) One topological descriptor, which is the degree of compactness (dF), and 4) Five normalized variants of dP, dG, dQ, dD, and dF. Those are denoted as zP, zG, zQ, zD, and zF respectively.

SVMs deal with classification tasks by finding a hyperplane that separates training instances from two different classes with the maximum margin. The examples used to determine the hyperplane are the support vectors. We chose SVM as our classifier in the miRHunter system due to its high generalization capability, ability to find global classification solutions and successful application in bioinformatics and other practical domains. The model selection for SVMs involves the selection of a kernel function and its parameters which yield the optimal classification performance for a given dataset. Among the available kernel functions, the Radial Basis Function (RBF) is a reasonable choice as it is the most popular and widely used one due to its higher reliability in finding optimal classification solutions in most practical situations[19]. For the training of SVMs, we used a Python interface for the library LIBSVM 3.12 that is an integrated software for support vector classification, regression and distribution estimation[20]. This interface implements the C-SVM algorithm using the RBF kernel. The kernel parameters γ and C were tuned by 5-fold cross validation (CV) over the loose grid search on $C = 2^{-5}, 2^{-3}, ..., 2^{15}$ and $\gamma = 2^{-15}, 2^{-13}, ..., 2^{3}$. The pair (C, γ) that led to the highest CV accuracy was used to train the SVMs using the complete training set. Then a separate testing dataset was used to measure the performance of the developed classifier.

Performance Estimation

Sensitivity (Sn) and specificity (Sp) were used to estimate the performance of the miRHunter system and its SVM classifiers. Sensitivity and specificity can be defined as follows:

$$\text{Sensitivity (Sn)} = \frac{TP}{TP + FN}$$

$$\text{Specificity (Sp)} = \frac{TN}{TN + FP}$$

where TP = number of predicted true positives, TN = number of predicted true negatives, FN = number of predicted false negatives and FP = number of predicted false positives. As suggested by above equations, sensitivity is the proportion of true positives that are correctly classified by the miRHunter system and its classifiers. It shows how good our system is at detecting real pre-miRNAs. Specificity is the proportion of the true negatives correctly classified by the miRHunter system and its classifiers. It indicates how well our system is working at identifying real non-pre-miRNAs. Sensitivity and specificity were calculated through cross-validation. In typical cross-validation, the training and validation sets must cross-over in successive rounds such that each data point has a chance of being validated against. The basic form of cross-validation is *k*-fold cross-validation. In this work, 5-fold cross-validation was used.

Acknowledgements This research was supported by the Bio & Medical Technology Development Program of the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (No. NRF-2012M3A9D1054450) and by a 2016 Research Grant from Sangmyung University.

References

1. Chan, J.A., Krichevsky, A.M. & Kenneth, S.K. MicroRNA-21 is an antiapoptotic factor in human glioblastoma cells. *Cancer Res*. **65**, 6029-6033 (2005).
2. Esquela-Kerscher, A. & Slack, F.J. Oncomirs - microRNAs with a role in cancer. *Nat. Rev. Cancer* **6**, 6:259-269 (2006).
3. Bartel, D.P. MicroRNAs: Genomics, biogenesis, mechanism, and function. *Cell* **116**, 281-297 (2004).
4. Yekta, S., Shih, I.H. & Bartel, D.P. MicroRNA-directed cleavage of HOXB8 mRNA. *Science* **304**, 594-596 (2004).
5. Bagga, S. *et al*. Regulation by let-7 and lin-4 miRNAs results in target mRNA degradation. *Cell* **122**, 553-563 (2005).
6. Kozomara, A. & Griffiths-Jones, S. miRBase: annotating high confidence microRNAs using deep sequencing data. *Nucleic Acids Res*. **42**, D68-D73 (2014).
7. Szittya, G. *et al*. High-throughput sequencing of Medicago truncatula short RNAs identifies eight new miRNA families. *BMC Genomics* **9**, 593. doi: 10.1186/1471-2164-9-593 (2008).
8. Kim, K.B. A survey on computational approaches to the discovery of microRNA genes. *Current Bioinformatics* **9**, 173-181 (2014).
9. Byvatov, E. & Schneider, G. Support vector machine applications in bioinformatics. *Appl. Bioinformatics* **2**, 67-77 (2003).
10. Lancashire, L.J., Lemetre, C. & Ball, G.R. An introduction to artificial neural networks in bioinformatics -

application to complex microarray and mass spectrometry datasets in cancer studies. *Brief. Bioinform.* **10**, 315-329 (2009).
11. Yoon, B.J. Hidden markov models and their applications in biological sequence analysis. *Curr. Genomics* **10**, 402-415 (2009).
12. Webb, G.I., Boughton, J. & Wang, Z. Not so Naïve Bayes: aggregating one-dependence estimators. *Machine Learning* **58**, 5-24 (2005).
13. Altschul, S., Gish, W., Miller, W., Myers, E. & Lipman, D. Basic local alignment search tool. *J. Mol. Biol.* **215**, 403-410 (1990).
14. Hofacker, I.L. Vienna RNA secondary structure server. *Nucleic Acids Res.* **31**, 3429-h3431 (2003).
15. Loong, K. & Mishra, S. De novo SVM classification of precursor microRNAs from genomic pseudo hairpins using global and intrinsic folding measures. *Bioinformatics* **23**, 1321-1330 (2007).
16. Batuwita, R. & Palade, V. microPred: Effective classification of pre-miRNAs for human miRNA gene prediction. *Bioinformatics* **25**, 989-995 (2009).
17. Zhong, Y., Xuan, P., Han, K., Zhang, W. & Li, J. Improved Pre-miRNA Classification by Reducing the Effect of Class Imbalance. *Biomed Res. Int.* **2015**, DOI: 10.1155 (2015).
18. Pruitt, K.D. & Maglott, D.R. RefSeq and LocusLink: NCBI gene-centered resources. *Nucleic Acids Res.* **29**, 137-140 (2001).
19. Keerthi, S. & Lin, C.J. Asymptotic behaviours of support vector machines with Gaussian kernel. *Neural Comput.* **15**, 1667-1689 (2003).
20. Chang, C.C. & Lin, C.J. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology* **2**, 27:1-27:27 (2011).