Journal of Theoretical Biology 385 (2015) 153–159

Contents lists available at ScienceDirect

Journal of Theoretical Biology

journal homepage: www.elsevier.com/locate/yjtbi

Identification of microRNA precursor with the degenerate K-tuple or Kmer strategy

Bin Liu [a,b,c,*], Longyun Fang [a], Shanyi Wang [a], Xiaolong Wang [a,b], Hongtao Li [d], Kuo-Chen Chou [c,e]

[a] School of Computer Science and Technology, Harbin Institute of Technology Shenzhen Graduate School, Shenzhen, Guangdong, China
[b] Key Laboratory of Network Oriented Intelligent Computation, Harbin Institute of Technology Shenzhen Graduate School, Shenzhen, Guangdong, China
[c] Gordon Life Science Institute, Boston, MA 0478, USA
[d] Wendeng Marine Environmental Monitoring Station, Station Oceanic Administration Wendeng, Weihai, Shandong, China
[e] Center of Excellence in Genomic Medicine Research (CEGMR), King Abdulaziz University, Jeddah 21589, Saudi Arabia

HIGHLIGHTS

- microRNA (miRNA) plays an important role in gene expression.
- Identification of real pre-miRNAs is important miRNA-based therapy.
- A novel predictor was developed for fast and effectively identifying miRNA.

ARTICLE INFO

Article history:
Received 16 July 2015
Received in revised form
21 August 2015
Accepted 24 August 2015
Available online 9 September 2015

Keywords:
MicroRNA precursor
True pre-miRNA
False pre-miRNA
Degenerate Kmer
deKmer web-server
Long-range effect

ABSTRACT

The microRNA (miRNA), a small non-coding RNA molecule, plays an important role in transcriptional and post-transcriptional regulation of gene expression. Its abnormal expression, however, has been observed in many cancers and other disease states, implying that the miRNA molecules are also deeply involved in these diseases, particularly in carcinogenesis. Therefore, it is important for both basic research and miRNA-based therapy to discriminate the real pre-miRNAs from the false ones (such as hairpin sequences with similar stem-loops). Most existing methods in this regard were based on the strategy in which RNA samples were formulated by a vector formed by their Kmer components. But the length of Kmers must be very short; otherwise, the vector's dimension would be extremely large, leading to the "high-dimension disaster" or overfitting problem. Inspired by the concept of "degenerate energy levels" in quantum mechanics, we introduced the "degenerate Kmer" (deKmer) to represent RNA samples. By doing so, not only we can accommodate long-range coupling effects but also we can avoid the high-dimension problem. Rigorous jackknife tests and cross-species experiments indicated that our approach is very promising. It has not escaped our notice that the deKmer approach can also be applied to many other areas of computational biology. A user-friendly web-server for the new predictor has been established at http://bioinformatics.hitsz.edu.cn/miRNA-deKmer/, by which users can easily get their desired results.

1. Introduction

MicroRNAs (miRNAs) are small single-strand and non-coding RNAs (ncRNAs), which play important roles in gene regulation by targeting messenger RNAs (mRNAs) for cleavage or translational repression (Fig.1). Their lengths are about 17–25 nt (Lopes et al., 2014). The miRNAs are also involved in many important biological processes, such as affecting stability, translation of mRNAs, and negatively regulating gene expression in post-transcriptional processes. Therefore, it is fundamentally important to identify the real pre-miRNAs from the false ones. Unfortunately, it is difficult to

[*] Correspondence to: Harbin Institute of Technology Shenzhen Graduate School, HIT Campus Shenzhen University Town, Xili, Shenzhen 518055, China. Tel.: +86 0755 2603 3283.
E-mail addresses: bliu@insun.hit.edu.cn (B. Liu), dragoncloudest@gmail.com (L. Fang), wangshanyiwsy@gmail.com (S. Wang), wangxl@insun.hit.edu.cn (X. Wang), lht760@126.com (H. Li), kcchou@gordonlifescience.org (K.-C. Chou).

http://dx.doi.org/10.1016/j.jtbi.2015.08.025
0022-5193/© 2015 Elsevier Ltd. All rights reserved.

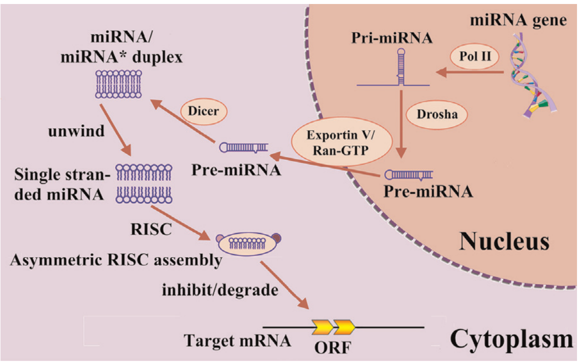

Fig. 1. MicroRNAs (miRNAs) are small single-strand and non-coding RNAs (ncRNAs), which play important roles in gene regulation by targeting messenger RNAs (mRNAs) for cleavage or translational repression.

use the traditional experimental techniques for timely and systematically detecting miRNAs from a genome (Xuan et al., 2011). Facing the avalanche of genome sequences generated in the postgenomic age, it is imperative to develop computational methods (Li et al., 2010) for detecting miRNAs according to their sequence information alone.

At present, the most successful computational approaches in this field were using the Kmer composition to represent RNA samples (Wei et al., 2014). But the length of Kmers practically really useful in this area is less than 6 nucleobases. This is because any Kmers longer than that would result in using extremely high-dimension vectors to represent the statistical samples (Chen et al., 2014b, 2014a; Lin et al., 2014), leading to the “high-dimension disaster” (Wang et al., 2008) or overfitting problem that would significantly reduce the deviation tolerance or cluster tolerant capacity (Chou, 1999) so as to lower down the success rate of prediction. However, the miRNAs can vary from 17 to 25 nucleobases. Therefore, the Kmer approach can be only used to represent the short-range or local information of miRNA sequences but not their long-range or global information. Particularly, most of the pre-miRNAs have the characteristic of stem-loop hairpin structures (Xue et al., 2005). In view of this, some novel approaches are definitely needed to relax the aforementioned limitation imposed on the length of Kmers for miRNA sequences. The present study was initiated in an attempt to address these problems.

2. Methods

2.1. Benchmark dataset

The benchmark dataset S used in this study can be formulated as

$$S = S^{+} \cup S^{-} \tag{1}$$

where the positive subset S^{+} contains pre-miRNA samples only, which were extracted from the latest version of miRBase (release 21: June 2014). Furthermore, the CD-HIT software (Li and Godzik, 2006; Li et al., 2009) was used to make sure that none of the pre-miRNA samples included in S^{+} has ≥80% pairwise sequence identity to any other. By doing so, we finally obtained 1 612 pre-miRNA samples for the positive subset S^{+}.

The negative subset S^{-} also contained 1 612 samples, which were randomly picked from the 8489 false pre-miRNAs in (Xue et al., 2005). Again, none of the negative samples included in S^{-} has ≥80%pairwise sequence identity to any other.

Since the most stringent cutoff threshold for DNA sequences by CD-HIT is 75%, to our best knowledge, the aforementioned benchmark dataset is so far the most stringent and largest benchmark dataset constructed for studying the prediction of pre-miRNAs.

Also, as pointed out in a comprehensive review (Chou and Shen, 2007), there is no need to separate a benchmark dataset into a training dataset and a testing dataset if a prediction method is to be validated by the jackknife or subsampling (K-fold) cross-validation since the outcome thus obtained is actually from a combination of many different independent dataset tests.

The benchmark dataset S as well as its subsets $S^{+}$and S^{-}, along with the corresponding detailed sequences are given in Supporting information S1.

As pointed in Chou (2011) and concurred in a series of recent publications (see, e.g., Chen et al., 2012; Min and Xiao, 2013; Xiao et al., 2013a, 2015; Xu et al., 2013b, 2014b; Liu et al., 2014a, 2015a; Qiu et al., 2014, 2015; Jia et al., 2015), one of the keys in successfully developing a sequence-based statistical predictor is how to effectively formulate the sequence samples concerned with an effective mathematical expression that can truly capture their intrinsic correlation with the target to be predicted. Below we are to address this problem.

2.2. Use degenerate Kmer composition to represent RNA samples

Suppose an RNA sequence $\mathbf{R}$ with L nucleobases (nitrogenous bases or nucleic acid residues); i.e.,

$$\mathbf{R} = \mathrm{B}_1\mathrm{B}_2\mathrm{B}_3\mathrm{B}_4\mathrm{R}_5\mathrm{B}_6\mathrm{B}_7 \cdots \mathrm{B}_L \tag{2}$$

where

$$\mathrm{B}_i \in \left\{ \mathrm{A}\,(\text{adenine}) \quad \mathrm{C}(\text{cytosine}) \quad \mathrm{G}(\text{guanine}) \quad \mathrm{U}(\text{uracil}) \right\} \tag{3}$$

denotes the nucleobase at sequence position $i\,(= 1,2, \ \cdots, L)$.

The most straightforward method to represent an RNA sample is just using its entire nucleobase sequence as shown in Eq. (2). In order to identify whether the RNA sample belongs to pre-miRNA or false pre-miRNA, one may use various sequence-similarity-search-tools, such as BLAST (Altschul et al., 1997; Schaffer et al., 2001), to search RNA database for those sequences that have high sequence similarity to the query RNA sample $\mathbf{R}$. Subsequently, the attributes of the RNAs thus found were used to deduce the attribute concerned for $\mathbf{R}$. Unfortunately, this kind of straightforward sequential model, although quite intuitive and without missing any of the sample's information, failed to work when it did not have significant sequence similarity to any character-known RNA. To overcome such a difficulty, one had to consider using non-sequential or discrete vector models to formulate RNA samples. Actually, the other important reasons to embrace the vector models is that all the existing computational algorithms can only handle vectors but not sequences, as elaborated in a recent paper Chou (2015).

Here we are to propose a completely different vector model to represent RNA sample, as described below.

First of all, formulating the RNA sequence of Eq. (2) according to its secondary structure derived from the Vienna RNA software package (released 2.1.6) (Hofacker, 2003), we have

$$\mathbf{R} = \Psi_1\Psi_2\Psi_3\Psi_4\Psi_5\Psi_6\Psi_7 \cdots \Psi_L \tag{4}$$

where Ψ_1 denotes the secondary structure state of B_1, Ψ_2 the structure state of B_2, and so forth. They can be any of the following seven structure states; i.e.,

$$\Psi_i \in \{\mathrm{A},\ \mathrm{C},\ \mathrm{G},\ \mathrm{U},\ \mathrm{A-U},\ \mathrm{G-C},\ \mathrm{U-G}\} \tag{5}$$

where A, C, G, U represent the structure states of the four unpaired nucleobases, while A–U, G–C, U–G represent the structure states of the three paired bases. Note that, in order to reduce computational cost, here we do not discriminate the A–U with U–A, G–C with C–G and G–U with U–G.

Based on the seven structure states, if the RNA sequence is represented by a vector of Kmer (or K-tuple) composition (Chen et al., 2014c; Liu et al., 2015d), we have

$$\mathbf{R} = \begin{bmatrix} f_1^{\text{Kmer}} & f_2^{\text{Kmer}} & f_3^{\text{Kmer}} & f_4^{\text{Kmer}} & \cdots & f_{7^K}^{\text{Kmer}} \end{bmatrix}^{\mathbf{T}} \tag{6}$$

where the symbol $\mathbf{T}$ is the transpose operator, f_i^{Kmer}represents the normalized occurrence frequency of the ith Kmer. As we can see from Eq. (6), with the incensement of K values, although longer-range information can be incorporated, the vector's dimension will increase rapidly. For example, when $K = 8$, its dimension will be $7^8 = 10^{8\log_{10}7} > 5.75 \times 10^6$,causing the so-called "high-dimension disaster" (Wang et al., 2008) or overfitting problem that will significantly reduce the deviation tolerance or cluster-tolerant capacity (Chou, 1999) so as to lower down the prediction success rate or stability. Therefore, Eq. (6) is useful only when the value of K is very small. In other words, it can only be used to incorporate the local or short-range sequence-order information, but certainly not the global or long-range sequence-order information. To approximately cover the long-range sequence-order effects, one popular and well-known method is to use the pseudo components that were originally introduced in dealing with protein/peptide sequences (Chou, 2001, 2005) and recently extended to deal with DNA/RNA sequences (Chen et al., 2014b, 2014a, 2014c, 2015a, 2015b; Liu et al., 2015a, 2015b, 2015c, 2015d).

In this study, we would like to introduce a different approach, which was inspired by the concept of "degenerate energy levels". As is well known, in quantum mechanics an energy level is deemed as degenerate if it corresponds to two or more different measurable states of a quantum system. The new concept introduced here is called "Degenerate Kmer" or just "deKmer" for short, by which we can still accommodate long-range information but meanwhile also able to avoid the high dimension problem, as elaborated below.

For an RNA sequence generated by the Vienna software (Hofacker, 2003) as given by Eq. (4), its degenerated Kmers or deKmers ($K \geq 2$) possess the following feature: two deKmers can be deemed no different if they each have at least two base pairs that are sequentially pairwise identical to each other regardless whether the remaining $(K-2)$base pairs have the same match or not.

Thus, according to the concept of deKmer, instead of Eq. (6), we have

$$\boldsymbol{R} = \begin{bmatrix} f_1^{\text{deKmer}} & f_2^{\text{deKmer}} & f_3^{\text{deKmer}} & f_4^{\text{deKmer}} & \cdots & f_{\Omega}^{\text{deKmer}} \end{bmatrix}^{\mathbf{T}} \tag{7}$$

where

$$\Omega = 7^2 \cdot C_K^2 = 7^2 \cdot \frac{K!}{(K-2)!2!} \tag{8}$$

is the number of all the possible different deKmers. The vector formed by the deKmer components as defined in Eq. (7) is called "deKmer vector". Its dimension will be significantly reduced in comparison with the dimension of the Kmer vector as defined in Eq. (6).

For example, when $K = 4$, the dimension of the deKmer composition vector (Eq. (7)) is $\Omega = 7^2 \times [4!/(4-2)!2!] = 294$, whereas the dimension of the corresponding Kmer composition vector (Eq. (6)) is $7^4 = 2401$. The latter is more than 8 times the size of the former. For the case of $K = 5$, the dimension of the deKmer composition vector is $\Omega = 7^2 \times [5!/(5-2)!2!] = 490$, whereas the dimension of the corresponding Kmer composition vector is $7^5 = 16807$. The latter is more than 34 times the size of the former. In other words, compared with Kmer, using deKmer can substantially reduce the composition vector's dimension. This is particularly true when the value of K continues increasing (Table 1).

Table 1
Comparison of the dimension between Kmer vector (Eq. (6)) and DeKmer vector (Eq. (7)).

K	Dimension of Kmer vector [a]	Dimension of DeKmer vector [b]	Ratio γ [c]
2	49	49	1
3	343	147	~3
4	2401	294	~8
5	16,807	490	~34
6	117,649	735	~160
7	823,543	1029	~800
8	5,764,801	1372	~4201
9	40,353,607	1764	~22,876
10	282,475,249	2205	~128,107
11	1,977,326,743	2695	~733,702
12	13,841,287,201	3234	~4,279,928
⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮

[a] Calculated by (see Eq. (6)).
[b] Calculated by Eq. (8).
[c] The quotient of the number in column 2 over that in column 3; it is equal to $\gamma = 7^{K-2} \cdot \frac{K!}{(K-2)!2!}$.

Accordingly, hereafter, we shall use the deKmer vector (Eq. (7)) to formulate the RNA samples. By doing so, we can relax the aforementioned limitation imposed on the value of K without facing the high-dimension disaster trouble.

2.3. Support vector machine (SVM)

With the high quality benchmark dataset as given in Supporting information S1 and the RNA samples defined in a vector space (Eq. (7)) to avoid the high-dimension disaster problem (Wang et al., 2008), the next step is what kind of algorithm or operation engine should be used to conduct the prediction (Chou, 2011).

SVM is a machine-learning algorithm based on the statistical learning theory. It has been widely used in the realm of bioinformatics (see, e.g., Chen et al., 2014a, 2014b; Lin et al., 2014; Liu et al., 2014a, 2014b; Feng et al., 2013; Ding et al., 2014; Guo et al., 2014; Fan et al., 2014; Xu et al., 2014a; Qiu and Xiao, 2014). The basic idea of SVM is to construct a separating hyper-plane so as to maximize the margin between the positive dataset and negative dataset. The nearest two points to the hyper-plane are called support vectors. SVM first constructs a hyper-plane based on the training dataset, and then maps an input vector $\vec{X}$ from the input space into a vector in a higher dimensional Hillbert space, where the mapping is determined by a kernel function. A trained SVM can output a class label (in our case, pre-miRNA or false pre-miRNA) based on the mapping vector of the input vector. In the current study, the LIBSVM algorithm (Chang, 2009) was employed, which is a software for SVM classification and regression. The kernel function was set as Radial Basis Function (RBF) and the two parameters C and γ were optimized on the benchmark dataset by adopting the grid tool provide by LIBSVM (Chang, 2009.). For a brief formulation of SVM and how it works, see the papers (Chou and Cai, 2002; Cai and Zhou, 2003); for more details about SVM, see a monograph (Cristianini and Shawe-Taylor, 2000).

The prediction method thus developed is called the deKmer predictor.

3. Results and discussion

3.1. Metrics used to measure the prediction quality

The current study belongs to a binary (two-lass) classification problem; i.e., for a given segment of RNA sequence, whether its outcome is positive (pre-miRNA) or negative (false pre-miRNA). For this kind of binary classification problem, the following set of metrics were often used to measure the prediction quality

$$\begin{cases} \mathrm{Sn} = \dfrac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \\ \mathrm{Sp} = \dfrac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} \\ \mathrm{Acc} = \dfrac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \\ \mathrm{MCC} = \dfrac{(\mathrm{TP} \times \mathrm{TN}) - (\mathrm{FP} \times \mathrm{FN})}{\sqrt{(\mathrm{TP} + \mathrm{FP})\,(\mathrm{TP} + \mathrm{FN})(\mathrm{TN} + \mathrm{FP})(\mathrm{TN} + \mathrm{FN})}} \end{cases} \tag{9}$$

where TP represents the true positive; TN, the true negative; FP, the false positive; FN, the false negative; Sn, the sensitivity; Sp, the specificity; Acc, the accuracy; MCC, the Mathew's correlation coefficient (Chen et al., 2007). The metrics formulated in Eq. (9) is not easy-to-understand for most experimental scientists, and hence here we would prefer to use the following formulation as done by many investigators in a series of recent publications (see, e.g., Lin et al., 2014; Chen et al., 2012, 2013; Xu et al., 2013a, 2013b, 2014a, 2014b; Qiu et al., 2014, 2015; Xiao et al., 2015; Guo et al., 2014):

$$\begin{cases} \mathrm{Sn} = 1 - \dfrac{N_-^+}{N^+} 0 \ \le \mathrm{Sn} \le 1 \\ \mathrm{Sp} = 1 - \dfrac{N_+^-}{N^-} \quad 0 \le \mathrm{Sp} \le 1 \\ \mathrm{Acc} = \Lambda = 1 - \dfrac{N_-^+ + N_+^-}{N^+ + N^-} \quad 0 \le \mathrm{Acc} \le 1 \\ \mathrm{MCC} = \dfrac{1 - \left(\frac{N_-^+ + N_+^-}{N^+ + N^-}\right)}{\sqrt{\left(1 + \frac{N_+^- - N_-^+}{N^+}\right)\left(1 + \frac{N_-^+ - N_+^-}{N^-}\right)}} \quad -1 \le \mathrm{MCC} \le 1 \end{cases} \tag{10}$$

where $N^+$is the total number of the positive samples or pre-miRNAs investigated while N_-^+ the number of pre-miRNA samples incorrectly predicted to be of false pre-miRNA; $N^-$the total number of the negative samples or false pre-miRNAs investigated while N_+^- the number of the false pre-miRNAs incorrectly predicted to be of pre-miRNA.

According to Eq. (10), it is crystal clear to see the following. When $N_-^+ = 0$ meaning none of the pre-miRNAs was incorrectly predicted to be a false pre-miRNA, we have the sensitivity Sn = 1. When N_-^+=N^+ meaning that all the pre-miRNAs were incorrectly predicted to be the false pre-miRNAs, we have the sensitivity Sn = 0. Likewise, when $N_+^- = 0$ meaning none of the false pre-miRNAs was mispredicted, we have the specificity Sp = 1; whereas $N_+^- = N^-$ meaning that all the false pre-miRNAs were incorrectly predicted as true pre-miRNAs, we have the specificity Sp = 0. When N_-^+=N_+^-=0 meaning that none of pre-miRNAs in the positive dataset and none of the false pre-miRNAs in the negative dataset were incorrectly predicted, we have the overall accuracy Acc = 1 and MCC = 1; when N_-^+=N^+ and $N_+^- = N^-$ meaning that all the pre-miRNAs in the positive dataset and all the false pre-miRNAs in the negative dataset were incorrectly predicted, we have the overall accuracy Acc = 0 and MCC = −1; whereas when N_-^+=N^+/2 and $N_+^- = N^-/2$ we have Acc = 0. 5 and MCC = 0 meaning no better than random guess. As we can see from the above discussion, it would make the meanings of sensitivity, specificity, overall accuracy, and Mathew's correlation coefficient much more intuitive and easier-to-understand by using the formulation of Eq. (10), particularly for the meaning of MCC.

It should be pointed out, however, the set of metrics as defined in Eq. (10) is valid only for the single-label systems. For the multi-label systems whose emergence has become more frequent in system biology (Chou et al., 2012; Lin et al., 2013; Xiao and Wu, 2011; Wang et al., 2015) and system medicine (Chou, 2015; Xiao et al., 2013b), a completely different set of metrics as defined in Chou (2013) is needed.

3.2. Method used to conduct cross validation

With the evaluation metrics available, the next thing is what validation method should be used to derive the metrics values.

In statistical prediction, the following three cross-validation methods are often used to derive the metrics values for a predictor: independent dataset test, subsampling (or K-fold cross-validation) test, and jackknife test (Chou and Zhang, 1995). Of the three methods, however, the jackknife test is deemed the least arbitrary that can always yield a unique outcome for a given benchmark dataset as elucidated in Chou (2011) and demonstrated by Eqs. (28)–(32) therein. Accordingly, the jackknife test has been widely recognized and increasingly used by investigators to examine the quality of various predictors (see, e.g., (Xiao and Wu, 2011; Zhou and Assa-Munt, 2001; Hajisharifi et al., 2014; Mondal and Pai, 2014; Dehzangi et al., 2015; Khan et al., 2015; Kumar et al., 2015).

Accordingly, in this study we also use the jackknife test to evaluate the accuracy of the current predictor. During the jackknife test, each of the samples in the benchmark dataset is in turn singled out as an independent test sample and all the rule-parameters are calculated without including the sample being identified. Although the jackknife test may take more computational time, it is worthwhile because it will yield a unique outcome for a given benchmark dataset.

3.3. Tested results of deKmer predictor and comparison with its counterparts

At first, we investigated how different values of K would affect the performance of the deKmer predictor. In this step, for reducing computational time, the predictor was examined by the 5-fold cross validation on the benchmark dataset S (cf. Eq. (1) as well as the Online Supporting information S1). The results thus obtained are given in Fig.2, from which we can see that, when K=8, the predictor's accuracy (Acc) reaches its peak, indicating that the optimal K value for deKmer is 8 when it is trained by the current benchmark dataset.

Subsequently, with K fixed at 8, the rigorous jackknife tests were performed to calculate the Sn, Sp, Acc, and MCC as defined in Eq. (10) for the deKmer predictor on the same benchmark dataset. The results thus obtained are listed in Table 2, where for facilitating comparison, the corresponding results by the Kmer approach and Triplet-SVM predictor (Xue et al., 2005) are also given. As we can see from the table, the new deKmer predictor outperformed its counterparts in all the four metrics.

Furthermore, the comparison was also made via a graphic plot because it can provide intuitive insights useful for in-depth analyses of complicated biological systems (see, e.g., (Chou and Forsen, 1980; Althaus et al., 1993; Chou, 2010; Zhou, 2011)). Depicted in Fig.3 is the ROC (Receiver Operating Characteristic) plot (Fawcett, 2005); the larger the area under the curve, the better the

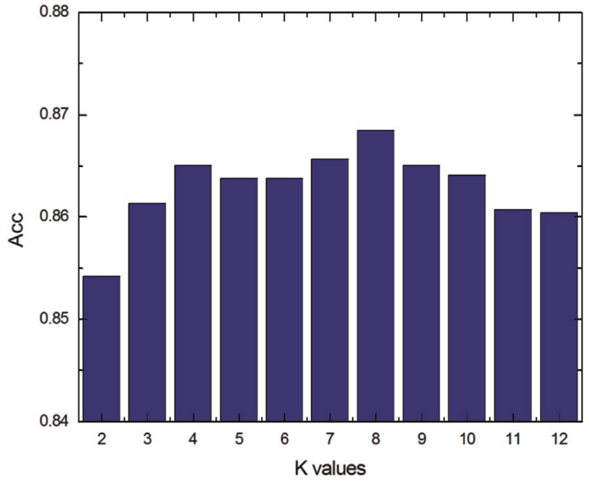

Fig. 2. An illustration to show how the accuracy (Acc) achieved by the deKmer predictor varies with the values of *K*. See the text for further explanation.

Table 2
Comparison of the current deKmer predictor with its counterparts by the jackknife tests on the same benchmark dataset (cf. the Online Supporting information S1).

Method	Sn (%)	Sp (%)	Acc (%)	MCC	AUC[a]	Running time (s)[b]
Kmer[c]	81.00	80.73	80.86	0.62	0.891	1258
Triplet-SVM[d]	78.47	85.20	81.85	0.64	0.894	35
deKmer[e]	**85.36**	**88.46**	**86.91**	**0.74**	**0.941**	41

[a] AUC is the abbreviation of the “Area Under the Curve” for ROC (Receiver Operating Characteristic) plot (Fawcett, 2005); the larger the value of AUC, the better the corresponding predictor. See the main text for further explanation.

[b] The running (or CPU) time of converting all the 3224 samples in the benchmark dataset into the feature vectors.

[c] See the paper (Fletez-Brant et al., 2013) and Eq. (6) for Kmer's definition. For the current benchmark dataset S the highest success rates were reached when $K=5$.

[d] Results obtained by the in-house implemented Triplet-SVM (Xue et al., 2005) on the same benchmark dataset.

[e] See Eqs. (7) and (8) with $K=8$.

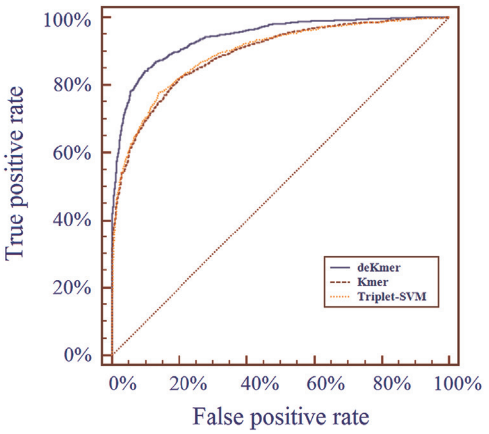

Fig. 3. A graphical illustration showing the performance of the new predictor deKmer in comparison with its counterparts via the ROC (Receiver Operating Characteristic) plot (Fawcett, 2005). See the footnote d of Table 2 as well as the main text for further explanation.

corresponding predictor (Fawcett, 2005). As we can see from the figure, the area under the ROC curve of the new deKmer predictor is remarkably greater than those of their counterparts, clearly indicating a remarkable improvement of the deKmer predictor over its counterparts.

3.4. Comparison of deKmer with other methods in computational cost

To further indicate the advantage of the current method over the existing ones, a comparison of deKmer with its counterparts in CPU time was also made, as given in column 7 of Table 2. As we can see there, the CPU time for Kmer (Fletez-Brant et al., 2013) is 1258 s, which is about 30-fold for the deKmer and 36-fold for Triplet-SVM. This is because, as shown in Table 1, it needs to generate 16,807-D feature vectors when $K=5$, the optimal state for Kmer (see the footnote “a” of Table 2). In contrast, for deKmar, even when $K=8$, it only needs to generate 1029-D feature vectors (see Table 1). As for the Triplet-SVM, it only needs to generate 32-D vectors. Although the CPU time is also quite short, only very local or short-range coupling effects are incorporated, and hence its prediction quality is lower than that of deKmer as shown in Table 2. Therefore, the degenerate Kmer approach as formulated in Eq. (8) not only has the advantage to enhance the prediction quality by accommodating some longer-range coupling effects but also can significantly reduce the computational time.

3.5. Cross-species experiments that show more power of deKmer

The current deKmer predictor was trained by the dataset that contains the human samples only (cf. Supporting information S1). What would happen when using it to identify the microRNA precursors from other species or organisms? To address this problem, extended tests were performed for the predictor by 4022 pre-miRNA samples, of which 962 from Mus musculus, 277 from Rattus norvegicus, 659 from Gallus gallus, 291 from Danio rerio, 175 from Caenorhabditis briggsae, 250 from Caenorhabditis elegans, 210 from Drosophila pseudoobscura, 256 from Drosophila melanogaster, 592 from Oryza sativa, 325 Arabidopsis thaliana, and 25 Epstein Barr Virus (see Supporting information S2). All these samples were taken from the latest miRBase (release 21: June 2014), and none of them has ≥80% pairwise sequence identity with any other in the same species.

Listed in Table 3 are the results identified by the deKmer on the 4022 samples from the aforementioned eleven species, respectively. From the table we can observe the following. (1) Of the 4022 miRNA precursors, 3497 were correctly identified; the overall accuracy is 86.95%. (2) For the samples from some species, such as Oryza sativa, Caenorhabditis briggsae, Arabidopsis thaliana, and Epstein Barr Virus, the success rates were remarkably high, ranged from 93% to 100%. All these results indicate that, although the DeKmer predictor was trained by the samples from human species, it can be quite effectively applied to identify the miRNA precursors for many other species as well.

Why the current deKmer trained with the samples from human species can be so successfully used to identify the miRNA precursors of the other species? To address this problem, we calculated the sequence similarity scores between the human miRNAs and the other eleven species' miRNAs respectively. The results thus obtained are given in columns 5 and 6 of Table 3, from which we can observe the following trend: the average scores for the correctly predicted samples is higher than those of the incorrectly ones, implying that the sequence similarity still plays a dominant role in using deKmer to identify the miRNA precursors of the other species although some sort of flexibility would be allowed due to the degenerate nature of the current deKmer.

3.6. Web-server for deKmer

To enhance its practical application value, a web-server for deKmer has been established at http://bioinformatics.hitsz.edu.cn/miRNA-deKmer/. Furthermore, to maximize the convenience of

Table 3
Results obtained by the deKmer predictor trained with human samples in identifying the microRNA precursors from eleven other species.

Species	Number of microRNA precursors	Number of correct identification	Acc (%)	Similarity score A[a] (%)	Similarity score B[b] (%)
Mus musculus	962	794	82.54	20.69	14.68
Rattus norvegicus	277	240	86.64	33.33	16.05
Gallus gallus	659	504	76.48	20.42	14.22
Danio rerio	291	271	93.13	25.20	22.2
Caenorhabditis briggsae	175	165	94.29	15.73	13.8
Caenorhabditis elegans	250	226	90.40	16.60	13.21
Drosophila pseudoobscura	210	186	88.57	15.19	13.83
Drosophila melanogaster	256	221	86.33	15.12	12.57
Oryza sativa	592	551	93.07	18.03	12.95
Arabidopsis thaliana	325	314	96.62	19.30	12.55
Epstein Barr Virus	25	25	100	15.32	N/A
Total	4022	3497	86.95		

[a] The average sequence alignment score between the correctly predicted samples in the cross-species datasets and the samples in the benchmark dataset S of deKmer taken from human species.

[b] The average sequence alignment score between the incorrectly predicted samples in cross-species datasets and the samples in the benchmark dataset S of deKmer taken from human species.

most experimental scientists, a built-in *Guide* window is provided therein. By clicking it, users can easily get their desired results without the need to go through the detailed mathematical equations presented in this paper. By clicking the *Benchmark Data* window, users can download the data used to train and test the deKmer predictor.

Since the idea of deKmer approach can also be applied to many other areas of computational biology, its software package is available upon request.

Conflicting interests

The authors declare no competing financial interests.

Acknowledgments

The authors wish to thank the two anonymous Reviewers for their constructive comments, which were very useful for strengthening the presentation of this study. This work was supported by the National Natural Science Foundation of China (61300112 and 61272383), the Scientific Research Innovation Foundation in Harbin Institute of Technology (Project no. HIT.NSRIF.2013103), the Natural Science Foundation of Guangdong Province (2014A030313695), the Project Sponsored by the Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry, Shenzhen Municipal Science and Technology Innovation Council (Grant no. CXZZ20140904154910774), and National High Technology Research and Development Program of China (863 Program) (2015AA015405).

Appendix A. Supplementary material

Supplementary data associated with this article can be found in the online version at http://dx.doi.org/10.1016/j.jtbi.2015.08.025.%0d.

References

Althaus, I.W., Chou, J.J., Gonzales, A.J., Diebel, M.R., Aristoff, P.A., Tarpley, W.G., Reusser, F., 1993. Kinetic studies with the nonnucleoside HIV-1 reverse transcriptase inhibitor U-88204E. Biochemistry 32, 6548–6554.

Altschul, S.F., Madden, T.L., Schaffer, A.A., Zhang, J., Zhang, Z., Miller, W., Lipman, D. J., 1997. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res. 25, 3389–3402.

C. Chang, C.J. Lin, 2009. LIBSVM – A Library for Support Vector Machines. Available at ⟨http://www.csie.ntu.edu.tw/~cjlin/libsvm/⟩.

Cai, Y.D., Zhou, G.P., 2003. Support vector machines for predicting membrane protein types by using functional domain composition. Biophys. J. 84, 3257–3263.

Chen, J., Liu, H., Yang, J., 2007. Prediction of linear B-cell epitopes using amino acid pair antigenicity scale. Amino Acids 33, 423–428.

Chen, W., Feng, P.M., Lin, H., 2013. iRSpot-PseDNC: identify recombination spots with pseudo dinucleotide composition. Nucleic Acids Res. 41, e68.

Chen, W., Feng, P.M., Lin, H., 2014b. iSS-PseDNC: identifying splicing sites using pseudo dinucleotide composition. Biomed. Res. Int. 2014, 623149.

Chen, W., Lin, H., Chou, K.C., 2015b. Pseudo nucleotide composition or PseKNC: an effective formulation for analyzing genomic sequences. Mol. BioSyst. 11, 2620–2634, http://dx.doi.org/10.1039/c5mb00155b.

Chen, W., Lin, H., Feng, P.M., Ding, C., 2012. iNuc-PhysChem: a sequence-based predictor for identifying nucleosomes via physicochemical properties. PLoS One 7, e47843.

Chen, W., Feng, P.M., Deng, E.Z., Lin, H., 2014a. iTIS-PseTNC: a sequence-based predictor for identifying translation initiation site in human genes using pseudo trinucleotide composition. Anal. Biochem. 462, 76–83.

Chen, W., Lei, T.Y., Jin, D.C., Lin, H., 2014c. PseKNC: a flexible web-server for generating pseudo K-tuple nucleotide composition. Anal. Biochem. 456, 53–60.

Chen, W., Zhang, X., Brooker, J., Lin, H., 2015a. PseKNC-General: a cross-platform package for generating various modes of pseudo nucleotide compositions. Bioinformatics 31, 119–120.

Chou, K.C., 1999. A key driving force in determination of protein structural classes. Biochem. Biophys. Res. Commun. 264, 216–224.

Chou, K.C., 2001. Prediction of protein cellular attributes using pseudo amino acid composition. Protiens: Struct. Funct. Genet. 43, 246–255 (Erratum: ibid., 2001, Vol.44, 60).

Chou, K.C., 2005. Using amphiphilic pseudo amino acid composition to predict enzyme subfamily classes. Bioinformatics 21, 10–19.

Chou, K.C., 2010. Graphic rule for drug metabolism systems. Curr. Drug Metab. 11, 369–378.

Chou, K.C., 2011. Some remarks on protein attribute prediction and pseudo amino acid composition (50th anniversary year review). J. Theor. Biol. 273, 236–247.

Chou, K.C., 2013. Some remarks on predicting multi-label attributes in mMolecular biosystems. Mol. Biosys. 9, 1092–1100.

Chou, K.C., 2015. Impacts of bioinformatics to medicinal chemistry. Med. Chem. 11, 218–234.

Chou, K.C., Forsen, S., 1980. Graphical rules for enzyme-catalyzed rate laws. Biochem. J. 187, 829–835.

Chou, K.C., Zhang, C.T., 1995. Review: prediction of protein structural classes. Crit. Rev. Biochem. Mol. Biol. 30, 275–349.

Chou, K.C., Cai, Y.D., 2002. Using functional domain composition and support vector machines for prediction of protein subcellular location. J. Biol. Chem. 277, 45765–45769.

Chou, K.C., Shen, H.B., 2007. Review: recent progresses in protein subcellular location prediction. Anal. Biochem. 370, 1–16.

Chou, K.C., Wu, Z.C., Xiao, X., 2012. iLoc-Hum: using accumulation-label scale to predict subcellular locations of human proteins with both single and multiple sites. Mol. Biosys. 8, 629–641.

Cristianini, N., Shawe-Taylor, J., 2000. An Introduction of Support Vector Machines and Other Kernel-Based Learning Methodds. Cambridge University Press, Cambridge, UK.

Dehzangi, A., Heffernan, R., Sharma, A., Lyons, J., Paliwal, K., Sattar, A., 2015. Gram-positive and gram-negative protein subcellular localization by incorporating evolutionary-based descriptors into Chou's general PseAAC. J. Theor. Biol. 364, 284–294.

Ding, H., Deng, E.Z., Yuan, L.F., Liu, L., 2014. iCTX-Type: a sequence-based predictor for identifying the types of conotoxins in targeting ion channels. BioMed. Res. Int. 2014, 286419.

Fan, Y.N., Xiao, X., Min, J.L., 2014. iNR-Drug: predicting the interaction of drugs with nuclear receptors in cellular networking. Int. J. Mol. Sci. 15, 4915–4937.

Fawcett, J.A., 2005. An introduction to ROC analysis. Pattern Recognit. Lett. 27 (8), 861–874.

Feng, P.M., Chen, W., Lin, H., 2013. iHSP-PseRAAAC: identifying the heat shock protein families using pseudo reduced amino acid alphabet composition. Anal. Biochem. 442, 118–125.

Fletez-Brant, C., Lee, D., McCallion, A.S., Beer, M.A., 2013. kmer-SVM: a web server for identifying predictive regulatory sequence features in genomic data sets. Nucleic Acids Res. 41, W544–W556.

Guo, S.H., Deng, E.Z., Xu, L.Q., Ding, H., 2014. iNuc-PseKNC: a sequence-based predictor for predicting nucleosome positioning in genomes with pseudo k-tuple nucleotide composition. Bioinformatics 30, 1522–1529.

Hajisharifi, Z., Piryaiee, M., Mohammad Beigi, M., Behbahani, M., Mohabatkar, H., 2014. Predicting anticancer peptides with Chou's pseudo amino acid composition and investigating their mutagenicity via Ames test. J. Theor. Biol. 341, 34–40.

Hofacker, I.L., 2003. Vienna RNA secondary structure server. Nucleic Acids Res. 31, 3429–3431.

Jia, J., Liu, Z., Xiao, X., Chou, K.C., 2015. iPPI-Esml: an ensemble classifier for identifying the interactions of proteins by incorporating their physicochemical properties and wavelet transforms into PseAAC. J. Theor. Biol. 377, 47–56.

Khan, Z.U., Hayat, M., Khan, M.A., 2015. Discrimination of acidic and alkaline enzyme using Chou's pseudo amino acid composition in conjunction with probabilistic neural network model. J. Theor. Biol. 365, 197–203.

Kumar, R., Srivastava, A., Kumari, B., Kumar, M., 2015. Prediction of beta-lactamase and its class by Chou's pseudo-amino acid composition and support vector machine. J. Theor. Biol. 365, 96–103.

Li, C., Feng, Y., Coukos, G., Zhang, L., 2009. Therapeutic microRNA strategies in human cancer. AAPS J. 11, 747–757.

Li, L., Xu, J., Yang, D., Tan, X., Wang, H., 2010. Computational approaches for microRNA studies: a review. Mamm. Genome 21, 1–12.

Li, W., Godzik, A., 2006. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. Bioinformatics 22, 1658–1659.

Lin, H., Deng, E.Z., Ding, H., 2014. iPro54-PseKNC: a sequence-based predictor for identifying sigma-54 promoters in prokaryote with pseudo k-tuple nucleotide composition. Nucleic Acids Res. 42, 12961–12972.

Lin, W.Z., Fang, J.A., Xiao, X., 2013. iLoc-animal: a multi-label learning classifier for predicting subcellular localization of animal proteins. Mol. BioSys. 9, 634–644.

Liu, B., Liu, F., Fang, L., 2015. repRNA: a web server for generating various feature vectors of RNA sequences. Mol. Genet. Genom. . http://dx.doi.org/10.1007/s00438-015-1078-7

Liu, B., Liu, F., Fang, L., Wang, X., 2015b. repDNA: a Python package to generate various modes of feature vectors for DNA sequences by incorporating user-defined physicochemical properties and sequence-order effects. Bioinformatics 31, 1307–1309.

Liu, B., Xu, J., Lan, X., Xu, R., Zhou, J., 2014a. iDNA-Prot|dis: identifying DNA-binding proteins by incorporating amino acid distance-pairs and reduced alphabet profile into the general pseudo amino acid composition. PLoS One 9, e106691.

Liu, B., Zhang, D., Xu, R., Xu, J., Wang, X., 2014b. Combining evolutionary information extracted from frequency profiles with sequence-based kernels for protein remote homology detection. Bioinformatics 30, 472–479.

Liu, B., Liu, F., Wang, X., Chen, J., Fang, L., 2015d. Pse-in-One: a web server for generating various modes of pseudo components of DNA, RNA, and protein sequences. Nucleic Acids Res. 43, W65–W71.

Liu, Z., Xiao, X., Qiu, W.R., 2015a. iDNA-Methyl: identifying DNA methylation sites via pseudo trinucleotide composition. Anal. Biochem. 474, 69–77 (also Data in Brief, 2015, 4, 87-89).

Lopes, Id.O., Schliep, A., Carvalho, A.Cd.Ld, 2014. The discriminant power of RNA features for pre-miRNA recognition. BMC Bioinform. 15, 124.

Min, J.L., Xiao, X., 2013. iEzy-Drug: a web server for identifying the interaction between enzymes and drugs in cellular networking. BioMed. Res. Int. 2013, 701317.

Mondal, S., Pai, P.P., 2014. Chou's pseudo amino acid composition improves sequence-based antifreeze protein prediction. J. Theor. Biol. 356, 30–35.

Qiu, W.R., Xiao, X., 2014. iRSpot-TNCPseAAC: identify recombination spots with trinucleotide composition and pseudo amino acid components. Int. J. Mol. Sci. 15, 1746–1766.

Qiu, W.R., Xiao, X., Lin, W.Z., 2014. iMethyl-PseAAC: identification of protein methylation sites via a pseudo amino acid composition approach. Biomed. Res. Int. 2014, 947416.

Qiu, W.R., Xiao, X., Lin, W.Z., 2015. iUbiq-Lys: prediction of lysine ubiquitination sites in proteins by extracting sequence evolution information via a grey system model. J. Biomol. Struct. Dyn. 33, 1731–1742.

Schaffer, A.A., Aravind, L., Madden, T.L., Shavirin, S., Spouge, J.L., Wolf, Y.I., Koonin, E. V., Altschul, S.F., 2001. Improving the accuracy of PSI-BLAST protein database searches with composition-based statistics and other refinements. Nucleic Acids Res. 29, 2994–3005.

Wang, T., Yang, J., Shen, H.B., 2008. Predicting membrane protein types by the LLDA algorithm. Protein Pept. Lett. 15, 915–921.

Wang, X., Zhang, W., Zhang, Q., Li, G.Z., 2015. MultiP-SChlo: multi-label protein subchloroplast localization prediction with Chou's pseudo amino acid composition and a novel multi-label classifier. Bioinformatics 31, 2639–2645.

Wei, L.Y., Liao, M.H., Gao, Y., Ji, R.R., He, Z.Y., Zou, Q., 2014. Improved and promising identification of human microRNAs by incorporating a high-quality negative set. Comput. Biol. Bioinform. 11.

Xiao, X., Wu, Z.C., 2011. iLoc-Virus: a multi-label learning classifier for identifying the subcellular localization of virus proteins with both single and multiple sites. J. Theor. Biol. 284, 42–51.

Xiao, X., Min, J.L., Wang, P., 2013a. iGPCR-Drug: a web server for predicting interaction between GPCRs and drugs in cellular networking. PLoS One 8, e72234.

Xiao, X., Wang, P., Lin, W.Z., 2013b. iAMP-2L: a two-level multi-label classifier for identifying antimicrobial peptides and their functional types. Anal. Biochem. 436, 168–177.

Xiao, X., Min, J.L., Lin, W.Z., Liu, Z., 2015. iDrug-Target: predicting the interactions between drug compounds and target proteins in cellular networking via the benchmark dataset optimization approach. J. Biomol. Struct. Dyn. 33, 2221–2233.

Xu, Y., Ding, J., Wu, L.Y., 2013a. iSNO-PseAAC: predict cysteine S-nitrosylation sites in proteins by incorporating position specific amino acid propensity into pseudo amino acid composition. PLoS One 8, e55844.

Xu, Y., Shao, X.J., Wu, L.Y., 2013b. iSNO-AAPair: incorporating amino acid pairwise coupling into PseAAC for predicting cysteine S-nitrosylation sites in proteins. PeerJ 1, e171.

Xu, Y., Wen, X., Shao, X.J., 2014a. iHyd-PseAAC: predicting hydroxyproline and hydroxylysine in proteins by incorporating dipeptide position-specific propensity into pseudo amino acid composition. Int. J. Mol. Sci. 15, 7594–7610.

Xu, Y., Wen, X., Wen, L.S., Wu, L.Y., 2014b. iNitro-Tyr: prediction of nitrotyrosine sites in proteins with general pseudo amino acid composition. PLoS One 9, e105018.

Xuan, P., Guo, M., Liu, X., Huang, Y., Li, W., Huang, Y., 2011. PlantMiRNAPred: efficient classification of real and pseudo plant pre-miRNAs. Bioinformatics 27, 1368–1376.

Xue, C., Li, F., He, T., Liu, G.P., Li, Y., Zhang, X., 2005. Classification of real and pseudo microRNA precursors using local structure-sequence features and support vector machine. BMC Bioinform. 6, 310.

Zhou, G.P., 2011. The disposition of the LZCC protein residues in wenxiang diagram provides new insights into the protein-protein interaction mechanism. J. Theor. Biol. 284, 142–148.

Zhou, G.P., Assa-Munt, N., 2001. Some insights into protein structural class prediction. Protiens: Struct. Funct. Genet. 44, 57–59.