Computational prediction of novel miRNAs from genome-wide data

G. Stegmayer [1], C. Yones [1], L. Kamenetzky [2], N. Macchiaroli [2] and D. H. Milone [1,*]

[1] sinc(*i*) (CONICET-UNL), Ciudad Universitaria, Santa Fe, Argentina

[2] IMPAM (CONICET-UBA), Facultad de Medicina, Buenos Aires, Argentina

[*] To whom correspondence should be addressed. e-mail: dmilone@sinc.unl.edu.ar

sinc(*i*) Research Institute for Signals, Systems and Computational Intelligence (fich.unl.edu.ar/sinc)
G. Stegmayer, C. Yones, L. Kamenetzky, N. Macchiaroli & D. H. Milone; "Computational Prediction of Novel miRNAs from Genome-Wide Data"
, pp. 29-37, 2017.

Computational prediction of novel miRNAs from genome-wide data

G. Stegmayer, C. Yones, L. Kamenetzky, N. Macchiaroli and D. H. Milone

Abstract The computational prediction of novel microRNAs (miRNAs) within a full genome involves identifying sequences having the highest chance of being bona fide miRNA precursors (pre-miRNAs). These sequences are usually named candidates to miRNA. The well-known pre-miRNAs are usually only a few in comparison to the hundreds of thousands of potential candidates to miRNA that have to be analyzed. Although the selection of positive labeled examples is straightforward, it is very difficult to build a set of negative examples in order to obtain a good set of training samples for a supervised method. In this chapter we describe an approach to this problem, based on the unsupervised clustering of unlabeled sequences from genome-wide data, and the well-known miRNA precursors for the organism under study. Therefore, the protocol developed allows for quick identification of the best candidates to miRNA as those sequences clustered together with known precursors.

Key words: microRNAs prediction; genome-wide data; unsupervised model; clustering; self-organizing map; high class imbalance.

1 Introduction

MicroRNAs (miRNAs) are a class of small non coding RNA molecules, present in both animals and plants, with a major role in regulation of gene expression [1]. Many studies have shown that miRNAs are implied in several important processes, for example, in cancer progression [2] as well as in viral infection progress [3] and parasites development [4]. Given their role in promoting or inhibiting certain diseases and infections, the discovery of new miRNAs is of high interest today.

G. Stegmayer, C. Yones and D. H. Milone
sinc(*i*) (CONICET-UNL), Ciudad Universitaria, Santa Fe, Argentina

L. Kamenetzky and N. Macchiaroli
IMPAM (CONICET-UBA), Facultad de Medicina, Buenos Aires, Argentina

MiRNA precursors (pre-miRNAs, also known as hairpins) generated during biogenesis have well-known RNA secondary structures that have allowed the development of computational algorithms for their identification. They typically exhibit a stem-loop structure or hairpin, with few internal loops or asymmetric bulges. Since large amount of similar hairpins can be folded in a given genome, the identification of those structures having the highest chance of being bona fide pre-miRNAs should be addressed. Due to the difficulty in systematically detecting pre-miRNAs by existing experimental techniques, which have proven to be time consuming and costly, computational methods play an important role nowadays in the identification of novel miRNAs [5, 6]. Machine learning methods essentially identify hairpin structures in non-coding and non-repetitive regions of the genome that are characteristics of miRNA precursor sequences, using structures, properties and features of well-known pre-miRNAs during the learning processes to discriminate between true predictions and false positives [7].

In a realistic scenario, when genome-wide data is used, a huge imbalance is often present between the positive class (a few known pre-miRNAs) and the unlabeled data (hundreds of thousands sequences). This important fact may lead to overlearning the majority class and/or incorrect assessment of classification performance. This means that most existing supervised proposals, although reporting very high accuracies, cannot be really trusted in practical situations.

In this chapter we present a protocol to predict novel pre-miRNAs from genome-wide data, with a classifier based on unsupervised learning. The model can predict the best candidates to pre-miRNAs, as sequences are clustered together with the well-known pre-miRNAs of the genomics data under study. This way, the very-hard to build negative artificial examples must not be defined, making it useful to work with genome-wide data from any organism.

2 Materials

2.1 Input data

- genomic DNA: A fasta file of genomic DNA (for example, genome.fa), with an entry for each chromosome. The genomics data will be mined to identify the best miRNA precursors.
- pre-miRNAs: A fasta file of known pre-miRNA sequences. These sequences are retrieved from specialized databases or reported in the literature as experimentally validated. These pre-miRNAs could be from the organism under study or a phylogenetically related one.
- other known non miRNA RNA sequences (optional): A fasta file of CDSs, tRNAs, rRNAs, non coding RNAs, and other no miRNAs sequences. These sequences can be used for filter out known other non miRNA RNAs.

sinc(i) Research Institute for Signals, Systems and Computational Intelligence (fich.unl.edu.ar/sinc)
G. Stegmayer, C. Yones, L. Kamenetzky, N. Macchiaroli & D. H. Milone; "Computational Prediction of Novel miRNAs from Genome-Wide Data"
, pp. 29-37, 2017.

2.2 Software

- Einverted (EMBOSS package). Program for finding inverted repeats in nucleotide sequences and genome folding.
  Free available from `emboss.sourceforge.net/download/`.
- RNA fold. This program reads RNA sequences, calculates their minimum free energy (MFE) structure and prints the MFE structure in bracket notation and its free energy. It can be downloaded from
  `www.tbi.univie.ac.at/RNA/RNAfold.1.html`.
- MiRcheck. Scripts to call and process einverted and RNAfold outputs. Free available from `bartellab.wi.mit.edu/software.html`
- BLAST. This program finds regions of similarity between biological sequences. Available at
  `ftp.ncbi.nlm.nih.gov/blast/executables/blast+/LATEST/`
- miRNA-SOM. This is a tool for the discovery of pre-miRNAs from genome-wide data. Available at
  `sourceforge.net/projects/sourcesinc/files/mirnasom/`
  (download version 23)
- miRNAfe (optional). It is a comprehensive tool to extract features from RNA sequences, providing almost all state-of-the-art feature extraction methods used today in several works from different authors.
  Available at `fich.unl.edu.ar/sinc/blog/web-demo/mirnafe/`

3 Methods

This section shows in detail the individual steps necessary to carry out the pipeline proposed for the analysis of raw genome-wide data, which is presented in Figure 1. Each step of the pipeline will be described and exemplified with linux commands[1]. Before beginning, the following software must be installed:

- Install einverted:
  ```
  sudo apt-get install emboss
  ```
- Install RNAfold:
  ```
  sudo apt-get install vienna-rna
  ```

3.1 Cut and fold genome-wide data

The input genome-wide data (a multi-fasta file named, for example, genome.fa) is pre-processed by miRcheck scripts, which calls einverted and RNAfold [8]. These steps can be done as follows:

[1] Commmand-line examples for Ubuntu Linux.

sinc(i) Research Institute for Signals, Systems and Computational Intelligence (fich.unl.edu.ar/sinc)
G. Stegmayer, C. Yones, L. Kamenetzky, N. Macchiaroli & D. H. Milone; "Computational Prediction of Novel miRNAs from Genome-Wide Data"
, pp. 29-37, 2017.

- Cut full genome into sequences: the original `run_einverted.pl` script from miRcheck can be used, but previously the gap penalty and other thresholds of einverted must be configured (see Note 1 and 2). A modified version of the script with these parameters is provided in the utils folder of miRNA-SOM (version 23). With the modified script, the following linux command can be used to run einverted:

  ```
  ./run_einverted.pl genome.fa genIR
  ```

- If the fasta file has extra information appart from the chomosomes (for example, mitochondrial DNA), it should be disregarded, leaving the chomosomes information only. For example, suppose that genome.fa has a particular string (such as `Chr_<number>`) that identifies chomosomes. Then, you can run:

  ```
  cat genIR | grep Chr_ > genIR_chr
  ```

- Folding sequences: this step can be done by editing `fold_inverted_repeats.pl`, adding RNAfold options to produce structures without lonely pairs (noLP) and avoid the generation of postscript drawings (noPS)[2]. After that, you can run:

  ```
  ./fold_inverted_repeats.pl genIR_chr genome.fa genIR_chr_f
  ```

3.2 Filter by energy and loops

The sequences obtained in the previous step, from the raw genome-wide data cut and folding procedure, must be filtered to improve prediction. Two filters can be applied: a minimum free energy (MFE) threshold of -20 according to the miRNA biogenesis model [1], and multi-loops sequences can be discarded, obtaining a reduced fasta file. This step can be done by running the script:

```
filterle.m
```

provided with the source code of miRNA-SOM. Inside this matlab script, the mentioned filters are applied and a fasta file named `all_folded_selected_le.fa` is obtained as output.

In this step, the script `filterle.m` also extracts the following features, that are the most commonly used in literature for pre-miRNA prediction [6]:

- Triplets [9]: combines the local contiguous structures with sequence information to characterize the hairpin structure. This feature focus on the information of every 3 adjacent nucleotides. In the predicted secondary structure, there are only two status for each nucleotide, paired or unpaired, indicated by brackets, "(" or ")", and dots, respectively. The left bracket "(" means that the paired nucleotide is located near the 5'-end and can be paired with another nucleotide at the 3'-end, which is indicated by a right bracket ")". For any 3 adjacent nucleotides, there are 2^3 possible structure compositions: `(((`, `((.`, `(..`, `(.(`, `.((`,

[2] A modified version of the script is also provided in the utils folder of miRNA-SOM version 23.

sinc(i) Research Institute for Signals, Systems and Computational Intelligence (fich.unl.edu.ar/sinc)
G. Stegmayer, C. Yones, L. Kamenetzky, N. Macchiaroli & D. H. Milone; "Computational Prediction of Novel miRNAs from Genome-Wide Data"
, pp. 29-37, 2017.

`.(.`, `..(` and `....`. Considering the middle nucleotide among the 3, there are 32 possible structure-sequence combinations, which are denote as `U(((`, `A((.`, etc.
- MFE value [10]: minimum free energy when folding; and
- Sequence length: count of the length of the nucleic acid string.

All these features are saved in the data folder of miRNA-SOM, to train the model as detailed in Section 3.5. Additionaly, any number of features can be extracted and used to train the miRNA-SOM classifier. The web-tool miRNAfe [11], which is a comprehensive tool to extract features from RNA sequences, can be used for features extraction. It provides almost all state-of-the-art feature extraction methods used today in several works from different authors.

3.3 Filter known non miRNA RNA

This is an optional step. If a fasta file of CDS, tRNAs, rRNAs and long non coding RNAs, as well as any other non miRNA sequences of the organism under study is available (for example, in a file named `known_rna.fa`), they can be used for filter out known non miRNAs. This can be done by using BLAST[12] with the following linux script:

```
./delkrna.sh known_rna.fa all_folded_selected_le.fa
    all_folded_to_remove.csv
```

This script is also provided with miRNA-SOM. It generates the file `all_folded_to_remove.csv`, which indicates the indexes of the sequences that must be removed from `all_folded_selected_le.fa`.

3.4 Mark well-known pre-miRNAs

As a result of the previous steps, the files `all_folded_selected_le.fa` and `all_folded_to_remove.csv` are obtained. The first one includes sequences that correspond to well-known pre-miRNAs of the organism under study. These known pre-miRNAs can be identified after a BLAST match against the microRNA hairpins deposited in the most recent version of miRBase[3], and put together into a multi-fasta file, for example named `mirnas.fa`.

These sequences must be labeled as positive class in order to properly train the miRNA-SOM classifier. This step can be done this way:

```
./selmirs.sh mirnas.fa all_folded_selected_le.fa
    all_folded_known_mirna.csv
```

[3] http://www.mirbase.org/

sinc(i) Research Institute for Signals, Systems and Computational Intelligence (fich.unl.edu.ar/sinc)
G. Stegmayer, C. Yones, L. Kamenetzky, N. Macchiaroli & D. H. Milone; "Computational Prediction of Novel miRNAs from Genome-Wide Data"
, pp. 29-37, 2017.

This script is also provided with miRNA-SOM. It generates the file `all_folded_known_mirna.csv`, which has the indexes of the sequences that correspond to well-known pre-miRNAs in `all_folded_selected_le.fa`.

3.5 Train miRNA-SOM and predict novel pre-miRNAs

The `mainsom.m` script provided in miRNA-SOM trains the SOM classifier [13] (shown in Figure 2). It learns the labeled sequences as positive class, and identifies novel candidates to pre-miRNAs. When this main script is run, the miRNA-SOM classifier is trained according to the Algorithm shown in Figure 3, where the following notation is used: G_ℓ and G_u are the labeled and unlabeled input training sequences, respectively, extracted from the input genome-wide data and represented by a feature vector (steps 1 to 4 of the pipeline of Figure 1). Labeled input sequences correspond to well-known pre-miRNAs; n is the initial map size ($n \times n$ neurons); and h_{max} is the maximum deep level.

The miRNA-SOM model training and prediction involves the following steps. While the maximum deep level of SOMs has not been reached (line 4), a SOM map is trained at each level (line 5). The top level SOM, at $h = 1$, is set to the initial map size (see Note 3) and trained with all input training data (labeled and unlabeled data). During training, each input data point is assigned to a map unit (neuron) according to the minimum Euclidean distance between the feature vector representing each sequence and each neuron centroid. Neurons are labeled by taking into account the labeled data only, as follows: if there is at least one labeled input sequence in a neuron (line 6), then this neuron is labeled as a miRNA-neuron, no matter how many other unlabeled data points are clustered there as well. Then, only sequences clustered on miRNA-neurons pass to the next level (line 8). After training all SOM levels, up to h_{max}, only the sequences that are clustered into labeled miRNA-neurons at the deepest level (h_{max}) are predicted as pre-miRNA candidates with a high probability of being miRNA precursors (line 9). This final list of top candidates is saved in the results folder of miRNA-SOM software. For practical applications of this model and the protocol, see Notes 4 and 5.

The deep structure of this classifier is shown in Figure 2. When the root SOM, on the first layer, is trained and becomes stable, only the data in the neurons having clustered together with at least one well-known pre-miRNA are chosen as input data for training the next map, in the second layer. These neurons are marked miRNA-neurons and, although they might contain much more unlabeled data than labeled one, due to the existing high class-imbalance, they are marked as positive class neurons. During model training, only sequences clustered in miRNA-neurons remain for further training the next deep level of miRNA-SOM. After training several layers, the best pre-miRNAs candidates are those sequeces that remained in the miRNA-neurons at the last deep level.

With this approach, each internal map is trained only with a portion of the whole input genome-wide data. This method reduces significantly the number of possi-

ble candidate to pre-miRNAs, level after level, retaining at the last level only the high confidence candidates. In this last level, each well-known pre-miRNA in the miRNA-neurons (in dark blue) is grouped together with unlabeled sequences. They are selected as the best bona-fide candidates to novel pre-miRNAs.

4 Notes

1. In the first step (Section 3.1), the recommended paramenters for einverted are: gap penalty `$GAP = 6`; minimum score threshold `$THRESH = 25`; match score `$M = 3`; mismatch score `$MM = 3`; and maximum separation between the start and end of the inverted repeat `$DIST = 95`.
2. Also in the first step the recommended paramenters to cut sequences are: window size `$WIN = 500000`; and window step `$step = 400000`.
3. It is recommended to start with a large initial SOM map, such as for example $n = 100$. After the first level, a large number of sequences will not pass to the next SOM level and they will be naturally discarded. After that, the map size number can be reduced.
4. A practical example on the application of this protocol to genome-wide data from *Echinococcus multilocularis* can be found in [13] and online in: `http://fich.unl.edu.ar/sinc/web-demo/mirna-som/`. The source code is available for free academic use at: `http://sourceforge.net/projects/sourcesinc/files/mirnasom/` (download version 23)
5. Another example on a model organism (*Caenorhabditis elegans*) is available at: `http://fich.unl.edu.ar/sinc/blog/web-demo/mirna-som-ce/`.

References

[1] Bartel DP (2004) MicroRNAs: genomics, biogenesis, mechanism, and function. Cell 116:281–297

[2] Esquela-Kerscher A, Slack FJ (2006) Oncomirs - microRNAs with a role in cancer. Nature Reviews Cancer 6(1):259–269

[3] Lecellier CH, Dunoyer P, Arar K, Lehmann-Che J, Eyquem S, Himber C, Saib A, Voinnet O (2005) A cellular MicroRNA mediates antiviral defense in human cells. Science 308(5721):557–560

[4] Rosenzvit M, Cucher M, Kamenetzky L, Macchiaroli N, Prada L, Camicia F (2013) MicroRNAs in Endoparasites. Nova Science Publishers

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

sinc(*i*) Research Institute for Signals, Systems and Computational Intelligence (fich.unl.edu.ar/sinc)
G. Stegmayer, C. Yones, L. Kamenetzky, N. Macchiaroli & D. H. Milone; "Computational Prediction of Novel miRNAs from Genome-Wide Data"
, pp. 29-37, 2017.

[6] Ivani de ON Lopes and Alexander Schliep and Andre de Carvalho (2014) The discriminant power of RNA features for pre-miRNA recognition. BMC Bioinformatics 15(1):124+

[7] Liu B, Li J, Cairns M (2014) Identifying mirnas, targets and functions. Briefings in Bioinformatics 15(1):1–19

[8] Lorenz R, Bernhart S, zu Siederdissen CH, Tafer H, Flamm C, Stadler P, Hofacker I (2011) ViennaRNA Package 2.0. Algorithms for Molecular Biology 6(1):26–36

[9] Xue C, Li F, He T, Liu GP, Li Y, Zhang X (2005) Classification of real and pseudo microRNA precursors using local structure-sequence features and support vector machine. BMC Bioinformatics 6(1):310

[10] Zuker M, Stiegler P (1981) Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. Nucleic Acids Research 9(1):133–148

[11] Yones C, Stegmayer G, Kamenetzky L, Milone D (2015) miRNAfe: a comprehensive tool for feature extraction in microRNA prediction. BioSystems 238:1–5

[12] Altschul S, Gish W, Miller W, Myers E, Lipman D (1990) Basic local alignment search tool. Journal of Molecular Biology 215(1):403–410

[13] Kamenetzky L, Stegmayer G, Maldonado L, Macchiaroli N, Yones C, Milone D (2016) MicroRNA discovery in the human parasite echinococcus multilocularis from genome-wide data. Genomics 107(6):274–280

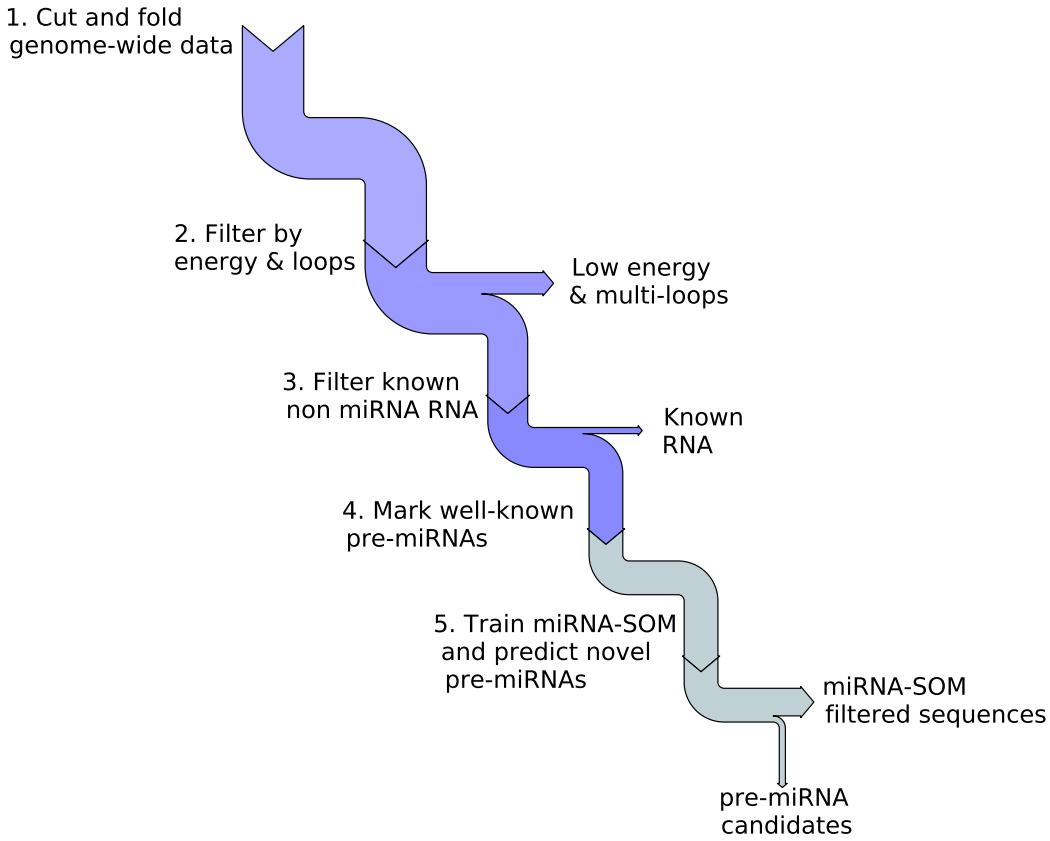

Fig. 1 Flow of the pipeline for novel pre-miRNA discovery from genome-wide data.

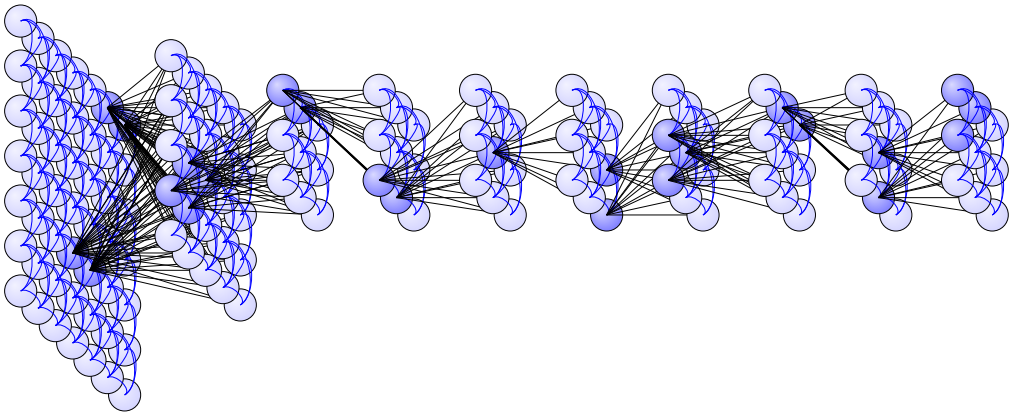

Fig. 2 miRNA-SOM classifier. Dark blue neurons have highly likely pre-miRNA candidates, which are input to the next level SOM (black lines).

Inputs :
  G_l: labeled input sequences (well-known pre-miRNAs)
  G_u: unlabeled input sequences
  n: initial map size ($n \times n$)
  h_{max}: maximum deep level
Outputs:
  C: pre-miRNA candidates at the last level

begin
  $h \leftarrow 1$
  $D_h \leftarrow G_u \cup G_l$
  while $h < h_{max}$ **do**
    Train a SOM with D_h
    Label as miRNA-neuron those neurons having at least one sequence in G_l
    $h \leftarrow h + 1$
    $D_h \leftarrow$ sequences in miRNA-neurons
  $C \leftarrow D_{h_{max}}$

Fig. 3 Unsupervised training and labeling of SOMs for novel pre-miRNA prediction from genome-wide data.

sinc(i) Research Institute for Signals, Systems and Computational Intelligence (fich.unl.edu.ar/sinc)
G. Stegmayer, C. Yones, L. Kamenetzky, N. Macchiaroli & D. H. Milone; "Computational Prediction of Novel miRNAs from Genome-Wide Data"
, pp. 29-37, 2017.