APPLICATION NOTE

Genomics & Informatics
Vol. 9(4) 194-196, December 2011

http://dx.doi.org/10.5808/GI.2011.9.4.194
pISSN 1598-866X eISSN 2234-0742

PromoterWizard: An Integrated Promoter Prediction Program Using Hybrid Methods

Kiejung Park[1] and Ki-Bong Kim[2]*

[1]Division of Bio-Medical informatics, Centers for Genome Science, Korea National Institute of Health, Chungcheongbuk-do 363-951, [2]Department of Biomedical Technology, Sangmyung University, Cheonan 330-720, Korea

Abstract

Promoter prediction is a very important problem and is closely related to the main problems of bioinformatics such as the construction of gene regulatory networks and gene function annotation. In this context, we developed an integrated promoter prediction program using hybrid methods, PromoterWizard, which can be employed to detect the core promoter region and the transcription start site (TSS) in vertebrate genomic DNA sequences, an issue of obvious importance for genome annotation efforts. PromoterWizard consists of three main modules and two auxiliary modules. The three main modules include CDRM (Composite Dependency Reflecting Model) module, SVM (Support Vector Machine) module, and ICM (Interpolated Context Model) module. The two auxiliary modules are CpG Island Detector and GCPlot that may contribute to improving the predictive accuracy of the three main modules and facilitating human curator to decide on the final annotation.

Keywords: regulatory networks, TSS, CDRM, SVM, CpG Island detector, GCPlot

Availability: Executable file of this program is available free of charge for non-commercial use only. Contact the corresponding author.

Introduction

The promoter recognition has an important bearing on the elucidation of gene regulation which is one of the most important research topics in molecular biology, but in which many things are still unclear. It is therefore important to exactly find the regulatory regions, examine them in detail, either computationally or by experiments, and learn the mechanisms that control the expression of genes. The first description of common patterns in eukaryotic promoters, in the form of weight matrices which are equivalent to linear hidden Markov models, can be found in the ground-breaking publication by Bucher (Bucher, 1990). Depending on the goals, computational approaches which deal with promoters can be divided into two classes: the *general* recognition of promoters and the analysis of these regions to identify the regulatory elements in them (or *specific* promoter recognition methods). The primary goal for the *general* methods is to identify TSS and/or core promoter elements for all genes in the genome; the *specific* methods focus on identifying specific regulatory elements (TF sites) that are shared by a particular set of transcriptionally related genes. *Specific* methods can have very high specificity when searching against the whole genome and can provide immediate functional clues to the downstream gene. On the other hand, because of their broad coverage, the *general* methods are extremely useful for large-scale genome annotation.

From the annotation point of view, promoter identification can help gene finding algorithms to identify the 5' UTRs that can span up to tens of thousands of kilobases and to determine the exact 5' boundary of a gene. In most cases, gene finding algorithms do not determine the exact 5' end of a gene since 5' UTRs have a very high variation in length and do not show significant statistical properties. In this respect, to facilitate genome annotation and improve the predictive accuracy in terms of specificity, we developed the integrated promoter prediction program using hybrid methods which have been developed separately in our previous works. The program, PromoterWizard, consists of three main modules and two auxiliary modules (Kim and Park, 2004; Kim, 2007; Kim, 2010).

Features and Results

PromoterWizard is composed of three main modules and two auxiliary modules. Three main modules are CDRM (Composite Dependency Reflecting Model) module, SVM (Support Vector Machine) module, and ICM (Interpolated Context Model) module. The CDRM actually represents a combination of first-, second-, third-

*Corresponding author: E-mail kbkim@smu.ac.kr
Tel +82-41-550-5377, Fax +82-41-550-5184
Received 16 November 2011, Revised 1 December 2011, Accepted 2 December 2011

and even much higher order or long-range dependencies obtained using the maximal dependency decomposition (MDD) procedure, which iteratively decomposes data sets into subsets on the basis of dependency degree and pattern inherent in the target promoter region to be modeled. In addition, decomposed subsets are modeled by using a first-order Markov model, allowing the predictive model to reflect the dependency between adjacent positions explicitly (Kim and Park, 2004). The ICM is essentially a probabilistic decision tree, i.e. a sparse probability distribution expressed as a decision tree. The tree construction is identical to constructing classification trees using information gain as the splitting criteria (Quinlan, 1993). Classification trees associate a class label with each leaf node of the tree. The labels in our case are the four nucleotide values, and our ICM determines a probability distribution for the base to be predicted given the context in which it occurs. Probabilistic decision trees have been designed for other applications (Delcher *et al*., 1999). The SVM is a supervised learning method used for classification and regression analysis. Here we employed the polynomial kernel function of SVMlight (http://svmlight.joachims.org/) (Kim, 2007). Two auxiliary modules are CpG Island Detector and GCPlot. CpG Island Detector can be used for CpG islands determination and GCPlot can give a clue to discriminating the promoter and coding regions from intron sequences. These two modules were brought in PromoterWizard to facilitate the end user to discriminate CpG island-associated promoter from non-CpG island associated promoter.

PromoterWizard is the window-based JAVA application implemented with JBuilder 9.0 which is a JAVA IDE (Integrated Development Environment). Window-based graphical user interface enables users to change the preset default parameter values into the ones tailored to

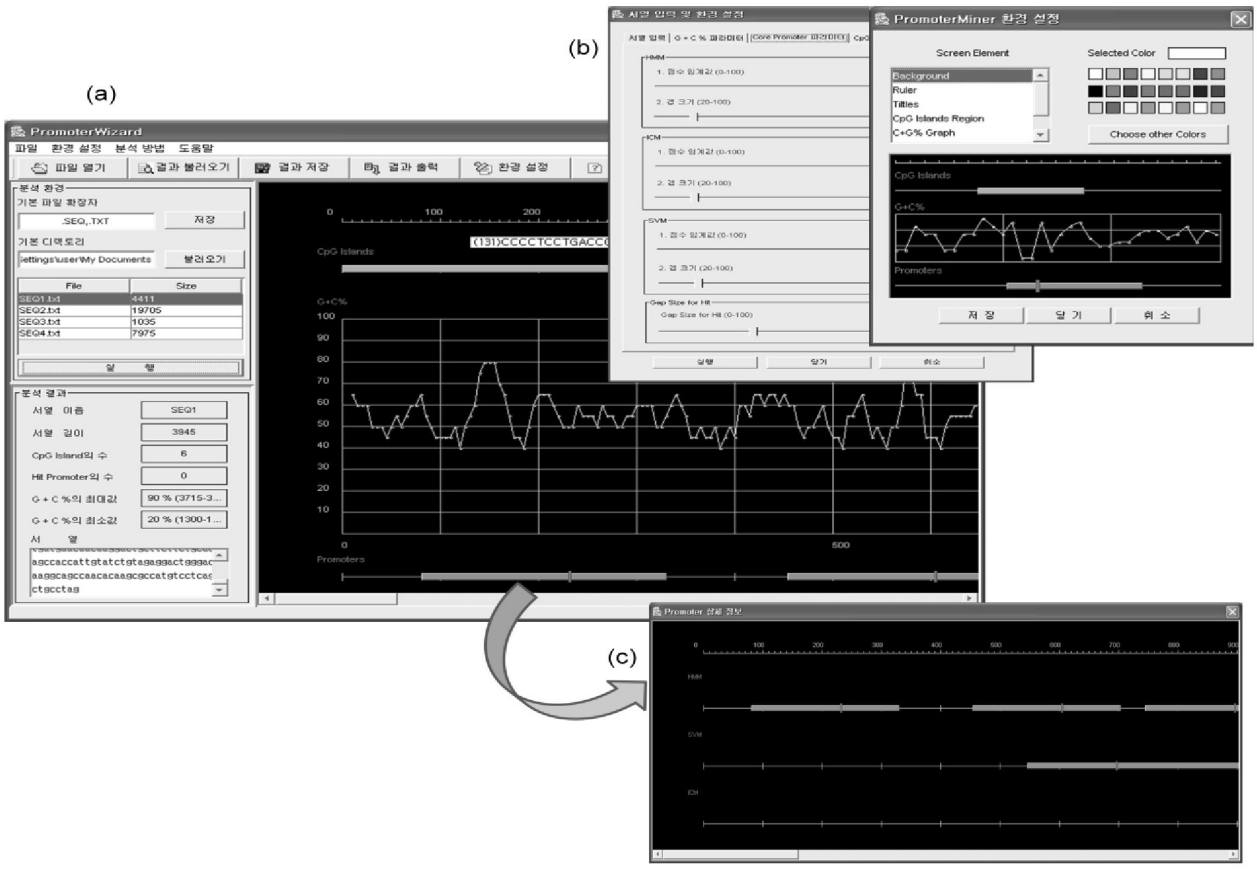

Fig. 1. (a) Window-based graphical user interface of PromoterWizard. The left frame consists of two parts - analysis environment and analysis result. Each provides the information on input sequence files and the summary of the analysis result respectively. The right frame displays CpG islands (top), G+C% plot (middle), and promoters (bottom). (b) Pop-up windows of various parameters setting for analysis optimization. (c) Pop-up window displays all the promoters detected by three different methods separately. This window will show up by double-clicking on the corresponding promoter of main screen.

their analysis intent (Fig. 1). In addition, the user can get the summary of analysis result on the left frame of graphical user interface which comprises two parts such as analysis environment and analysis result. The analysis environment part provides the information on input sequence files and the analysis result part furnishes users with the information on sequence name, sequence length, number of CpG islands, number of hit promoters, maximum value of G+C%, minimum value of G+C%, and input sequence data. The user can get the graphical analysis result on the right frame of graphical user interface, which displays CpG islands (top), G+C% plot (middle), and promoters (bottom) (Fig. 1a). Users can get the detailed information on the corresponding CpG island or promoter through pop-up window which will appear by double-clicking on it. In case of promoter, the pop-up window looks like the one in Fig. 1c.The pop-up window displays all the promoters detected by three different methods separately. The promoters displayed on the main screen are the output of logical product between CDRM, ICM, and SVM, the option of which can be specified through the menu 'Analysis Method'.

Discussion

The important part of computer-based annotation and analysis is concerned with regulatory DNA regions - parts of the sequence that have influence on how and when a gene is activated or *expressed*. Our approach belongs to *general* promoter prediction methods, the primary goal for which is to identify TSS and core promoter region for all protein-coding genes in a genome instead of seeking specific regulatory elements. According to our previous works, the sensitivity of CDRM, SVM, and ICM was 0.87, 0.86, and 0.71 respectively. In addition, the specificity of those methods was 0.72, 0.69, and 0.64 respectively. The result shows that specificity is relatively much lower than sensitivity. In this respect, in order to improve the specificity and to bring end-users promoter analyses tailored to their intent, we developed an integrated promoter prediction program using hybrid methods. It was implemented in JAVA and consists of three main modules (CDRM, SVM, ICM) and two auxiliary modules (CpG Island Detector and GCPlot). Each module can play a complementary role in improving the overall predictive performance and facilitating human curator to decide on the final annotation in terms of promoter determination.

Acknowledgements

This work was supported by Sangmyung University.

References

Bucher, P. (1990). Weight Matrix Description of Four Eukaryotic RNA Polymerase II Promoter Elements Derived from 502 Unrelated Promoter Sequences. *J. Mol. Biol.* 212, 563-578.

Delcher, A.L., Harmon, D., Kasif, S., White, O., and Salzberg, S. (1999). Improved Microbial Gene Identification with GLIMMER. *Nucleic Acids Res.* 27, 4636-4641.

Kim, K.B. and Park, S.H. (2004). Composite Dependency-Reflecting Model for Core Promoter Recognition in Vertebrate Genomic DNA Sequences. *J. Biochem. Mol. Biol.* 37, 648-656.

Kim, K.B. (2007). A Study on the Application Methods of a Support Vector Machine for Gene Promoter Prediction. *J. Life Sci.* 17, 714-718.

Kim, K.B. (2010). CpG Islands Detector: a Window-based CpG Island Search Tool. *Genomics&Informatics* 8, 58-61.

Quinlan, J.R. (1993). *Programs for Machine Learning* (San Matero: Morgan Kaufman Publishers).