Bioinformation by Biomedical Informatics Publishing Group *open access*

www.bioinformation.net **Prediction Model**

CpGIF: an algorithm for the identification of CpG islands

Sujuan Ye[1], Asai Asaithambi[1] and Yunkai Liu[1,*]

[1]Department of Computer Science, University of South Dakota, Vermillion, SD, USA;
Yunkai Liu[*] - E-mail: Yunkai.Liu@usd.edu; [*] Corresponding author

received May 01, 2008; revised May 13, 2008; accepted May 15, 2008; published May 20, 2008

Abstract:
CpG islands (CGIs) play a fundamental role in genome analysis and annotation, and contribute to improving the accuracy of promoter prediction. Besides, CGIs in promoter regions are abnormally methylated in cancer cells and thus can be used as tumor markers. However, current methods for identifying CGIs suffer from various drawbacks. We present a new algorithm for detecting CGIs, called CpG Island Finder (CpGIF), which combines the best features in the most commonly used algorithms and avoids their disadvantages as much as possible. Five public tools for CpG island searching are used to compare with CpGIF for the assessment of accuracy and computational efficiency. The results reveal that CpGIF has higher performance coefficient and correlation coefficient than these previous methods, which indicates that CpGIF is able to provide high sensitivity and specificity at the same time. CpGIF is also faster than those methods with comparable prediction accuracy.

Keywords: CpG islands; CpG dinucleotides; clustering algorithm

Background:
A CpG island is an unmethylated region in which CpG dinucleotides occur more frequently than in bulk DNA **[1]**. CpG islands are often associated with the promoters of most house-keeping genes and many tissue-specific genes, and thus have important regulatory functions and can be used as gene markers **[2]**. Most CGIs are non-methylated at any stage of development with the exception of methylated CGIs associated with transcriptionally silent genes on the inactive X chromosome and imprinted genes **[3]**. In cancer cells, the DNA methylation patterns are altered. Many non-island CpG sites in the bulk genome become unmethylated, while promoters containing CGIs are abnormally methylated **[4]**. Methylation of promoter-related CGIs is associated with abnormal silencing of transcription and is a common mechanism of inactivation of tumor-suppressor genes **[4]**. Since methylation of promoter CGIs is common in all types of cancer, the hypermethylated CGIs in promoter regions can be used as molecular tumor markers and make the early detection of cancer possible **[4]**. Identification of potential CGIs helps to find candidate regions for aberrant DNA methylation and therefore has contributed to the understanding of the epigenetic causes of cancer. CpG islands can be identified experimentally **[5, 6]** or computationally **[7-11]**. Recent studies have incorporated additional information, such as DNA sequence properties, DNA structure, and epigenetic states, into computational methods to predict the strength of each CGI quantitatively **[12]**. However, sequence-criteria-based approaches are still crucial to generate an initial genome-wide map of CpG islands.

There are several commonly used programs developed for locating CGIs in DNA sequences, including CpGPlot/CpGReport **[7]**, CpGProd **[8]**, CpGIS **[9]**, CpGIE **[10]**, and CpGcluster **[11]**. Most of those employ the sliding window technique with the exception of CpGCluster. The programs using the sliding window approach have the high capability combining small CpG islands. However, these methods suffer from several disadvantages: 1) the number and length of CGIs found depend on the window size and step size. If the window size is big, several short and loosely distributed CGIs might be clustered together to form a big one. 2) CGIs identified by those methods generally do not start and end with a CpG dinucleotide. 3) Because the window is moved in only one direction, those tools may not be able to locate CGIs accurately. 4) Longer running time. CpGCluster avoids the problems stated above and is much faster and more computationally efficient since it focuses on CpG dinucleotides and clusters neighboring CpG sites based on the physical distance between them. But several problems still exist in CpGCluster: 1) search results are dependent on the composition of the sequence scanned, i.e. a CGI identified in one sequence may be discarded when planted in another sequence with different composition. 2) Low prediction sensitivity. The CGIs detected by CpGClutser are usually short fragments

with high GC content and CpG observed/expected (*o/e)* ratio. This is the reason why the CGIs predicted by CpGCluster exhibit lower degree of overlap with Alu and higher degree of overlap with PhastCons.

To overcome the shortcomings of these commonly used tools for CGI finding mentioned above, we propose a novel algorithm for CpG-island finding, called CpG Island Finder (CpGIF). Instead of using the sliding window approach, CpGIF first searches regions with high CpG density, named as seeds. The seeds are then extended and clustered into the final CGIs. CpGIF combines the best features in current algorithms and avoids their disadvantages mentioned above. All CGIs predicted by CpGIF start and end with a CpG dinucleotide.

Methodology:

Our algorithm, CpGIF, was implemented in PERL, including a UNIX command-line application and a Common Gateway Interface (CGI) program. A web service and the source codes are available to public at http://www.usd.edu/~sye/cpgisland/CpGIF.htm. In CpGIF, a density cutoff is applied to exclude "mathematical CpG islands" caused by high G/C (or C/G) ratio. The same cutoff was also used in some previous tools, such as CpGIS.

Our algorithm consists of four major steps. First, we scan the DNA sequence from 5′ to 3′ end to find all CpG dinucleotides and record their positions. Then, we try to identify all initial seeds with default density of 0.10. In this step, an array is built to record the numbers of Gs and Cs in each initial seed and in the region located between two adjacent seeds. In the following steps, we will keep updating the array to calculate the GC content and CpG *o/e* ratio. Next, initial seeds are extended iteratively by decreasing the density cutoff from 0.09 to 0.05. The cutoff is reduced by 0.01 in each iteration. Finally, two neighboring extended seeds are clustered together if the distance between them is less than the maximum length of two adjacent extended seeds or 100 nt, whichever is smaller.

To assess the prediction performance of CpGIF and compare it to other programs, we created a set of test sequences by using the same method described by Hackenberg and colleagues **[11]**. The length of each known CGI in our test sequence is at least 200 nt since the same length criterion was used in all programs tested except CpGCluster.

Results:

The prediction accuracy of five commonly used programs and our algorithm CpGIF were evaluated with regard to nucleotide-level sensitivity (nSn), nucleotide-level specificity (nSp), nucleotide-level positive predictive value (nPPV), nucleotide-level performance coefficient (nPC), and nucleotide-level correlation coefficient (nCC). Table 1 (supplementary material) shows the results of five statistics.

The results reveal that the measures of correctness of CpGIF are higher than other programs. CpGCluster shows superb specificity and positive predictive value (99.98% and 99.9%, respectively), while the sensitivity, performance coefficient, and correlation coefficient are surprisingly low. This demonstrates that the results of CpGCluster depend on the composition of input sequences. If the length of non-island sequences is much longer than that of known CGIs, the probability of observing a CpG in the test sequence and thus the p-value of each CGI would be low. That may be the reason why CpGCluster had a relatively high sensitivity in **[11]**. CpGReport is a tool with high specificity and positive predictive value but moderate sensitivity, performance coefficient, and correlation coefficient. CpGProD only has a high sensitivity, but the other four statistics are low. The prediction accuracy of CpGIS and CpGIE are about the same. They have a slightly better sensitivity than CpGIF, but CpGIF has higher values for all other four measures. As shown in Table 1 (under supplementary material), our algorithm CpGIF outperforms the other tools by three or more measures.

One significant improvement in CpGIF is that it takes much less time to complete the search for CGIs. Table 1 (supplementary material) shows that CpGIS and CpGIE perform better than the other three commonly used tools. The algorithm in CpGIE basically followed that in CpGIS. Since CpGIE runs slower than CpGIS, we only compared the running time used by CpGIF with that of CpGIS. The results showed in Table 2 (supplementary material) indicate that CpGIF runs much faster than CpGIS. The reason is that CpGIF only scans the sequence once for counting G and C nucleotides (in step 2). In the extension and clustering steps, the GC content and CpG *o/e* ratio are calculated using the array that stores the G and C counts. This eliminates the redundant search for G and C and therefore greatly improves its computational efficiency.

The average length of the CGIs returned by CpGIF is much longer than those of CGIs detected by CpGIS. This is due to the higher capability of CpGIF in combining short CGIs. We should note that most of CGIs detected by CpGIS do not start or end with a CpG dinucleotide. If the non-CpG sites are removed from both sides of CGIs, only 2163 (chromosome 21) and 4614 (chromosome 22) CGIs would meet the length criterion. For chromosome 22, the total length of CGIs in the result of CpGIS is longer than that of CGIs returned by CpGIF. However, we observed that there are 2576 CGIs with length less than 210 nt and low CpG density. After excluding these

CGIs, the total length would be about the same and the properties of CGIs become better when compared to the complete CGI list, but still worse than those of CGIs returned by CpGIF (Table 2 in supplementary material). The results show that CGIs predicted by CpGIS have the same or shorter length, lower CpG *o/e* ratio, and lower CpG density than those identified by CpGIF, indicating that CpGIF locates CpG islands more accurately. The difference between CpGIF and CpGIS in prediction accuracy comes from two major improvements in CpGIF. First, all CGIs identified by CpGIF start and end with a CpG dinucleotide. Thus, the island boundaries are accurately defined. Second, when CpGIF extends seeds, the CpG density cutoff is decreased by 0.01 at each of five iterations, which can ensure that segments with higher CpG density can be clustered together first and therefore circumvent the problems caused by the single moving direction.

Conclusion:

We designed and developed a new algorithm, named CpGIF, to predict CGIs in DNA sequences. It takes the advantages of widely used tools and improves the accuracy and performance significantly. According to the length and accuracy of CGIs predicted and the running time needed, CpGIF is superior to other existing algorithms.

Acknowledgement:

This project was supported by NIH Grant Number 2 P20 RR016479 from the INBRE Program of the National Center for Research Resources and a subproject from the NIH National Center for Research Resources grant P20 RR015567 which is designed as a Center of Biomedical Research Excellence (COBRE).

References:

[01] F. Antequera and A. Bird, *Curr. Biol.,* 9: 661 (1999) [PMID: 10508580]

[02] F. Larsen *et al.*, *Genomics,* 13: 1095 (1992) [PMID: 1505946]

[03] A. Bird, *Genes Dev.,* 16: 6 (2002) [PMID: 11782440]

[04] J. G. Herman and S. B. Baylin, *N Engl J Med*., 349: 2042 (2003) [PMID: 14627790]

[05] L. E. Heisler *et al*., *Nucleic Acids Res.,* 33: 2952 (2005) [PMID: 15911630]

[06] R. Illingworth *et al., PLoS Biol.,* 6: e22 (2008) [PMID: 18232738]

[07] P. Rice *et al*., *Trends Genet*., 16: 276 (2000) [PMID: 10827456]

[08] L. Ponger and D. Mouchiroud, *Bioinformatics,* 18: 631 (2002) [PMID: 12016061]

[09] D. Takai and P. A. Jones, *Proc Natl Acad Sci.,* 99: 3740 (2002) [PMID: 11891299]

[10] Y. Wang and F. Leung, *Bioinformatics,* 20: 1170 (2004) [PMID: 14764558]

[11] M. Hackenberg *et al., BMC Bioinformatics,* 7: 446 (2006) [PMID: 17038168]

[12] C. Bock *et al., PLoS Comput Biol*., 3: e110 (2007) [PMID: 17559301]

Edited by P. Kangueane

Citation: Ye *et al.*, Bioinformation 2(8): 335-338 (2008)

Bioinformation by Biomedical Informatics Publishing Group *open access*

www.bioinformation.net **Prediction Model**

Supplementary material

Statistical measures used to assess program prediction accuracy:

The correctness of each tool is assessed at nucleotide level. The true positive (nTP), false negative (nFN), false positive (nFP), and true negative (nTN) at the nucleotide level are defined as followers:

- nTP is the number of nucleotides in both known CGIs and predicted CGIs
- nFN is the number of nucleotides in known CGIs but not in predicted CGIs
- nFP is the number of nucleotides not in known CGIs but in predicted CGIs
- nTN is the number of nucleotides in neither known CGIs nor predicted CGIs

At nucleotide level, the sensitivity, specificity, positive predictive value, performance coefficient (nPC), and the correlation coefficient (nCC) are defined as:

- Sensitivity: nSn=nTP / (nTP + nFN)
- Specificity: nSp=nTN / (nTN + nFP)
- Positive predictive value: nPPV = nTP / (nTP + nFP)
- Performance coefficient: nPC = nTP / (nTP + nFN + nFP), and
- Correlation coefficient: $nCC = \frac{nTP \times nTN - nFN \times nFP}{\sqrt{(nTP + nFN)(nTN + nFP)(nTP + nFP)(nTN + nFN)}}$

Program	nSn	nSp	nPPV	nPC	nCC
CpGIF	0.905±0.002	0.855±0.004	0.892±0.002	0.816±0.003	0.761±0.005
CpGIS	0.916±0.004	0.772±0.002	0.843±0.001	0.783±0.002	0.703±0.005
CpGIE	0.924±0.003	0.752±0.001	0.832±0.001	0.779±0.002	0.694±0.004
CpGCluster	0.113±0.003	1.000	0.999±0.001	0.113±0.003	0.227±0.003
CpGProD	0.953±0.002	0.198±0.010	0.613±0.004	0.595±0.004	0.237±0.012
CpGReport	0.759±0.005	0.958±0.001	0.961±0.002	0.737±0.004	0.714±0.004

Table 1: Prediction accuracy of 6 programs. The criteria used in programs except CpGCluster are, length ≥ 200 nt, GC% ≥ 50%, and CpG observed/expected ratio ≥ 0.60. In CpGcluster, the 75th distance was used as the distance threshold and the p-value cutoff is 10^{-5}.

	CpGIF		CpGIS		
	Chr21	Chr22	Chr21	Chr22	Chr22*
Number of CGIs	3371	6282	3704	9451	6875
Total length of CGIs (nt)	1,469,179	2,833,741	1,280,505	3,363,395	2,842,255
Average length of CGIs (nt)	436	451	346	356	413
GC %	56.94±5.09	57.0±5.289	57.98±4.22	54.78±5.38	55.12±5.59
CpG *o/e* ratio	0.68±0.09	0.67±0.074	0.66±0.09	0.63±0.061	0.64±0.069
CpG density	0.055±0.013	0.054±0.013	0.054±0.012	0.044±0.013	0.048±0.014
Running time (seconds)	109	155	19836	19237	

Table 2: Comparison of CpGIF and CpGIS for identifying CGIs. The subject sequence is human chromosome 21 (length= 46,944,329 nt) and chromosome 22 (length=49,691,432 nt). The search criteria are: length ≥ 200 nt, GC% ≥ 50%, and CpG observed/expected ratio ≥ 0.60.

*: The CGIs with length lesser than 210 nt were excluded. If the non-CpG sites at both ends of CGIs are removed, these CGIs would not meet the length criterion.