Genome analysis

SNPNB: analyzing neighboring-nucleotide biases on single nucleotide polymorphisms (SNPs)

Fengkai Zhang¹ and Zhongming Zhao¹,²,³,*

¹Virginia Institute for Psychiatric and Behavioral Genetics, Virginia Commonwealth University, Richmond, VA 23298, USA, ²Center for the Study of Biological Complexity, Virginia Commonwealth University, Richmond, VA 23284, USA and ³Kunming Institute of Zoology, Chinese Academy of Sciences, Kunming, Yunnan 650223, China

Received on September 26, 2004; revised on February 2, 2005; accepted on March 4, 2005
Advance Access publication March 15, 2005

ABSTRACT

Summary: SNPNB is a user-friendly and platform-independent application for analyzing Single Nucleotide Polymorphism NeighBor-ing sequence context and nucleotide bias patterns, and subsequently evaluating the effective SNP size for the bias patterns observed from the whole data. It was implemented by Java and Perl. SNPNB can efficiently handle genome-wide or chromosome-wide SNP data analysis in a PC or a workstation. It provides visualizations of the bias patterns for SNPs or each type of SNPs.

Availability: SNPNB and its full description are freely available at http://bioinfo.vipbg.vcu.edu/SNPNB/

Contact: zzhao@vcu.edu

Single nucleotide polymorphism (SNP) discovery is of major interest in the post-genome era because SNPs have broad applications in biological fields, such as fine mapping, disease studies, population genetics and molecular evolution (Gibbs et al., 2003). As of November 2004, >16 million unique SNPs from 23 species (Build 123) have been released in the dbSNP database of the National Center for Biotechnology Information (NCBI); meanwhile, millions of SNPs have been available in the private domains, such as Celera’s RefSNP database. This provides us an unprecedented opportunity to examine the local DNA-sequence context of SNPs, and therefore, to understand the molecular mechanisms of genome sequence evolution. Recent studies revealed that neighboring-nucleotide biases on SNPs were strong in the human and mouse genomes (Krawczak et al., 1998; Zhao and Boerwinkle, 2002; Zhang and Zhao, 2004). SNPNB accepts the data in FASTA format which has been widely used in the dbSNP, Celera RefSNP and other databases. It has the options for the analysis of the SNPs with known or unknown mutation direction. After the user chooses the sites or ranges in the flanking sequences, SNPNB subsequently obtains the proportion biases by comparing the frequency values computed from the data with the reference average values, which can be either entered manually by the user or computed from a reference sequence (e.g. human genome sequence). The results may be displayed in a table or in graphics. The user has the options to display the neighboring-nucleotide frequencies or biases by regions (e.g. coding regions) or by SNP types (e.g. A/G, C/T, A/C, G/T, A/T, C/G or all).

The second utility is to evaluate the effective SNP size. This is important for the genome-wide or chromosome-wide analysis because the user would like to evaluate whether the observed patterns

To whom correspondence should be addressed at: Virginia Institute for Psychiatric and Behavioral Genetics, Virginia Commonwealth University, PO Box 980126, Richmond, VA 23298-0126, USA

© The Author 2005. Published by Oxford University Press. All rights reserved. For Permissions, please email: journals.permissions@oupjournals.org
are representative or random in the genome. A small effective size means high confidence of the observed biases. A resampling algorithm was implemented. For the total number of SNPs \( N \), the user sets an initial subsample size \( n \), here \( n \ll N \). The program generates \( m \) random subsamples each of size \( n \). The program then computes the biases for each of the \( m \) samples, compares the biases observed from the whole data and shows the likelihood (%) of the \( m \) samples having the observed bias patterns relative to the user-defined constraints. Theoretically, the next step is an iterative procedure to find the minimum number of SNPs after each round of evaluation of the \( m \) subsamples of size \( n \); however, because the resampling algorithm is usually computationally intensive, it is extremely slow and probably unrealistic for a large dataset like human SNPs (\( N > 8 \) millions). We made two major improvements. First, we improved the resampling algorithm as follows:

1. A stratified sampling strategy is applied to generate random numbers, i.e. the program generates random numbers for each subrange of \( N \). This significantly improves the efficiency, e.g. it reduced the computational time from \(~30 \) to \( 4 \) h to generate 1000 samples of size 50 000 for \( N = 8 \) 043 656 in a PC.

2. When \( N \) is large, it is very time consuming to extract \( m \) samples and then to compute the biases because it has to scan the data file for \( m \) times. In SNPNB, the random numbers indexed by the sample are transferred and sorted into an array indexed by SNP. This needs to read the data file only once and takes \(~1/m \) (e.g. \( 1/1000 \)) computational time.

Second, because it should be sufficient to obtain a number close to the effective SNP size in research, we would suggest the user to evaluate whether the number is a reasonable estimate. The user may set the median likelihood value of all sites on the 5' side, 3' side and both sides to be at least 80%, which is commonly accepted in statistical power analysis. The user provides an initial empirical effective size to run SNPNB and then increases or decreases the size for the next round after an evaluation of the likelihood of the bias patterns. It may take a few rounds for the user to finally obtain a number close to the effective SNP size (e.g. \( 10 000 \rightarrow 100 000 \rightarrow 50 000 \rightarrow 30 000 \) for human SNPs). Although it is still slow despite the algorithm improvements above, it was able to obtain an approximate effective SNP size for humans within a couple of days and for mice within a day in a standard PC.

To illustrate the use of SNPNB, we reanalyzed the neighboring-nucleotide biases and the effective SNP size using human (8 043 656 SNPs) and mouse (469 445 SNPs) data (dbSNP build 121, ftp://ftp.ncbi.nih.gov/snp/). The results confirmed the bias patterns observed in our previous studies (Fig. 1). If we chose 1000 random subsamples, required the median likelihood value of all sites to be at least 80% and set the limit to be 0.3% (i.e. \( \pm 0.3\% \) of the observed biases), we obtained the effective SNP size of 30 000 for humans and mice, which is somewhat larger than our previous estimates (Zhang and Zhao, 2004). Since SNPNB can only obtain a number close to the effective SNP size, the user should interpret the estimated number cautiously. The summarized elapsed time of computation and screenshots of the results are shown in the SNPNB website.

SNPNB provides a powerful tool to examine and compare the patterns of local sequence of SNPs. The evaluation of the effective SNP size provides a confidence level to interpret the bias patterns. New features are being added for analysis of the short fragments and CpG dinucleotides in the SNP flanking sequences, comparative
analysis of neighboring biases for coding versus non-coding SNPs and statistical improvements in estimating the effective SNP size.

ACKNOWLEDGEMENTS
We thank the two anonymous reviewers for their constructive comments. This project was supported in part by a NARSAD Young Investigator Award.

REFERENCES