VisRD—visual recombination detection

Kristoffer Forslund1,∗, Daniel H. Huson2 and Vincent Moulton1,†

1The Linnaeus Centre for Bioinformatics, BMC, Box 598, Uppsala 751 24, Sweden and
2Center for Bioinformatics Tübingen (ZBIT), Sand 14, Tübingen 72076, Germany

Received on January 23, 2004; revised on June 15, 2004; accepted on July 7, 2004
Advance Access publication July 9, 2004

ABSTRACT
Summary: VisRD, a program for visual recombination detection in a sequence alignment is presented. VisRD is written in Java and is designed to complement the multi-purpose phylogenetic software package SplitsTree4.
Availability: The software is freely available from http://www.lcb.uu.se/~vmoulton/software/visrd/
Contact: krfo6395@student.uu.se

BACKGROUND
Recombination is an important evolutionary process, and is a key factor in shaping the structure of genes and genomes. Since sequence analysis can be severely biased by recombination, it is useful to have tools that can detect recombination in sequence data.

DETECTING RECOMBINATION
The rationale used by most approaches that seek to identify recombination in sequences is that recombination breaks down the correlation in evolutionary history between different regions of the sequences. Several methods for detecting recombination are based on the explicit reconstruction of phylogenetic trees for different parts of a sequence alignment and subsequent testing for changes in tree topology. Any changes provide indication of the underlying recombination events. The most widely used method of this kind is bootscanning (Salminen et al., 1995). Other methods only aim to determine the presence or absence of recombination, without trying to infer recombination breakpoints. This is usually achieved by looking for patterns in the sequence data that contradict the null hypothesis of a single evolutionary history (Worobey, 2001).

VISUAL RECOMBINATION DETECTION
In (Strimmer et al., 2003), an exploratory approach to detect recombination and associated breakpoints in a sequence alignment is introduced. This method emphasizes visualization and can be regarded as a synthesis of bootscanning (Salminen et al., 1995) and the quartet-mapping approach for analyzing tree-likeness of sequence data (Nieselt-Struwe and von Haeseler, 2001). The method uses quartet-trees to rapidly scan for phylogenetic inhomogeneity along an alignment of nucleotide or amino acid sequences. Using a filtering strategy, information gathered in the scan is condensed into two diagrams, the highway plot and the occupancy plot. This allows the rapid exploration of an alignment for recombination.

METHODS
An alignment is analysed using a sequence of overlapping windows. In each window, a measure of support for each of the three possible tree topologies is computed for all quartets of sequences, or for a random sample of quartets for large datasets. Supports can be calculated in various ways—the current implementation employs either maximum parsimony or statistical geometry. The information so gathered is then represented in a two-dimensional quartet-mapping triangle. The vertices of this triangle correspond to each possible tree topology. Each quartet gives rise to a point in this triangle, the position of which indicates the level of support for each of the three tree topologies.

The information contained in the sequence of quartet-mapping triangles is compressed into a highway plot (Fig. 1a). This graph allows the user to identify regions along the alignment where quartets change topology. In particular, the three main ‘lanes’ of this plot correspond to the three possible quartet tree topologies, and curves to the positions of points in the quartet-mapping triangles computed along the alignment. If many curves simultaneously cross into a different lane at some position, this indicates the possibility of a breakpoint. An occupancy plot can also be generated (Fig. 1b), which summarizes the information contained in the highway plot. Additionally, we provide an animated quartet-mapping triangle (Fig. 1c), so that positions of points in consecutive quartet-mappings can be viewed along the alignment. To reduce noise various filtering techniques such as trajectory filtering are employed—see Strimmer et al. (2003) for more details.

∗To whom correspondence should be addressed.
†Present address: School of Computing Science, University of East Anglia, Norwich NR4 7TJ, UK.
VisRD

Fig. 1. (a) A highway plot, displaying 25 quartets arising from an alignment of HIV virus sequences, including a known recombinant BR93029 (Strimmer et al., 2003; Gao et al., 1998). The breakpoints between the two histories (BR93029 being more closely related to HIV subtype B and F, respectively) have been marked in the plot with vertical lines. (b) The occupancy plot corresponding to the plot in (a). (c) Quartet-mappings for three windows of the same dataset.

IMPLEMENTATION

VisRD is written in Java, and the most recent version (v2.2) has been designed to compliment the multipurpose phylogenetic package SplitsTree4. VisRD can be run as a standalone application provided the SplitsTree4 libraries are installed. The most recent version of VisRD may be downloaded from http://www.lcb.uu.se/~vmoulton/software/visrd. SplitsTree4 is a Java re-implementation of SplitsTree (Huson, 1998) and is freely available from http://www-ab.informatik.uni-tuebingen.de/software/jsplits.

USAGE

VisRD is started as a standard Java application. The user is prompted to input a sequence alignment in Nexus format. Once such an alignment has been loaded, the user may change parameter values in the program defaults if desired, execute the analysis algorithm and then display the results using one or more of the provided visualizations.

The user must primarily decide which quartets are to be included in the analysis. Program default is all possible quartets. Support also exists for customizing the choice, such as random sampling of quartets for large datasets. Other parameters may also be varied as needed.

The most time-consuming step in the procedure is initial scanning. After this has been performed, the resulting data may be visualized in any of three ways described above. It is possible to change a number of plot parameters while viewing the plots, such as number of quartets included in the analysis, and the sequence range of the plot. Results may then be saved to .EPS vector graphics files.

ACKNOWLEDGEMENTS

K.F. and V.M. thanks The Swedish Research Council for its support. D.H.H. and V.M. thank ARI (project number HPRI-CT-2001-00153), and the Erasmus exchange program.

REFERENCES


