MARE (MAtrix REduction)

Manual

version 0.1.2-rc

Bonn, April 2011

1 About MARE

MARE (MAtrix REduction) was designed to find informative subsets of genes and taxa within a large phylogenetic dataset of amino acid sequences. The algorithm can be separated into two parts. (1) Calculation of the potential information content of genes, taxa and matrix, and (2) reduction to an optimized subset of taxa and genes. Therefore, MARE parses a superalignment and a corresponding charset file, which contains the partition (gene) bounds. A superalignment with m taxa and n genes is translated into a data availability matrix $\mathbf{B}_{m\times n}$, with $b_{ij} = 1$ if a sequence of gene j is present for taxon i, otherwise $b_{ij} = 0$. Having generated $\mathbf{B}$, it is checked if a sequence of a gene is available for ≥ 4 taxa, otherwise the gene which is represented by a column of $\mathbf{B}$ is excluded. Subsequently, the potential information content $\mathbf{q}$ of each gene is calculated.

1.1 Tree-likeness

The calculation of potential information content $\mathbf{q}$ of a gene is based on quartet-mapping using extended geometry mapping (eGM) (Nieselt-Struwe and von Haeseler, 2001). eGM calculates support values $\delta_1, \delta_2, \delta_3$ of the three possible topologies T_1, T_2 and T_3 reconstructed from a quartet of sequences. Therefore, distance values between pairs of sequences are calculated for each site and summed up to δ_1, δ_2 and δ_3. Generally, support values can be calculated for aligned sequence quartets with characters from any alphabet of finite length. A quartet of sequences must have at least 50 sites only consisting of amino acids. Distances between sequences can be measured by using any dissimilarity matrix (Nieselt-Struwe and von Haeseler, 2001). In MARE, BLOSUM62 (Henikoff and Henikoff, 1992) is implemented. δ_i can be transformed into a relative support s_i, with $0 \leq s_i \leq 1$ and $\sum_i s_i = 1$ by calculating

$$s_i = \delta_i / (\delta_1 + \delta_2 + \delta_3) \tag{1}$$

Relative support values can be interpreted as baricentric coordinates of a simplex S:

$$S = \{\sum_{i=1}^{3} s_i e_i | s_1 + s_2 + s_3 = 1, 0 \leq s_1, s_2, s_3 \leq 1\} \tag{2}$$

with e_i as unit vectors. Hence, $\mathbf{s} = (s_1, s_2, s_3)$ is a vector that defines a point in S. $(1,0,0)$, $(0,1,0)$ and $(0,0,1)$ are vectors corresponding to the

completely resolved topologies T_1, T_2 and T_3, whereas $(1/3, 1/3, 1/3)$ corresponds to the completely unresolved topology T_* (Figure 1). Thus, by eGM, a one-to-one relationship between the topology of the quartet and a single point in the simplex S is established. For a set of quartets of the same partition, the likeliness to resolve the overall topology increases with the frequency of points closest to T_1, T_2 or T_3. Therefore, in MARE up to 20.000 quartets of each partition are randomly drawn without duplication and simplex vectors are calculated. Since Nieselt-Struwe and von Haeseler (2001) showed that eGM is a conservative estimator of a tree-likeness **t**, MARE also takes the partially resolved tree topologies T_{12}, T_{13} and T_{23} into account and subdivides the simplex S into two areas ϕ_r and ϕ_*. ϕ_* is defined as all simplex points with the smallest distance to T_*, whereas ϕ_r contains all points closest to any other completely or partially resolved tree topology. The tree-likeness **t** of a partition is the number of simplex points within ϕ_r divided by the number of all simplex points.
Graphically, ϕ_r is represented by all simplex points within the region around the inner triangle ϕ_* (Figure 1). In Figure 2, three simplex graphs with different tree-likenesses are shown. After all genes are evaluated, each entry in matrix **B** is multiplied by the corresponding tree-likeness **t**, generating a matrix with $0 \leq b_{ij} \leq 1$.

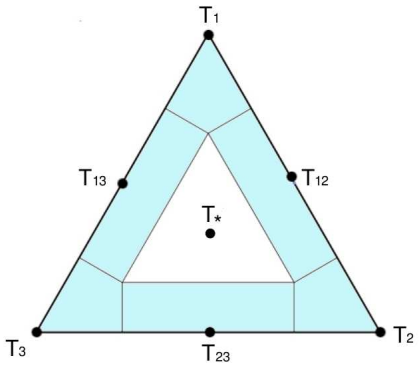

Figure 1: Simplex graph S with vectors of completely resolved trees T_1-T_3, partially resolved trees T_{12}, T_{13}, T_{23} and the unresolved tree T_*. The lightblue area determines ϕ_r and the inner triangle ϕ_*.

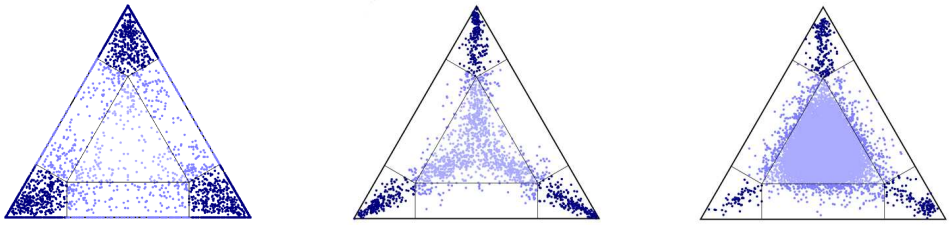

Figure 2: Simplex graphs S with different tree-likeness: 0.94 (left), 0.46 (middle), 0.06 (right). The inner triangle defines ϕ_*. Darkblue simplex points are mapped in the areas of the completely resolved topologies T_1, T_2 or T_3, all other points outside the inner triangle correspond to partially resolved trees.

1.2 Matrix Reduction

For the reduction of **B**, taxa and genes are evaluated by defining the information content **q** of genes (columns) as

$$\mathbf{q}_j = \frac{\sum_{i=1}^{m} \mathbf{b}_{ij}}{m} \tag{3}$$

with m as the number of taxa and the information content of taxa (rows) as

$$\mathbf{p}_i = \frac{\sum_{j=1}^{n} \mathbf{b}_{ij}}{n} \tag{4}$$

with n as the number of genes.
The information content of the matrix **B** is defined as

$$\mathbf{P(B)} = \frac{\sum_{i=1}^{m} \mathbf{p}_i}{m} = \frac{\sum_{j=1}^{n} \mathbf{q}_j}{n} \tag{5}$$

The reduction progress starts with checking the connectivity of **B**. Therefore, in terms of graph theory, genes of **B** are represented by a set of nodes. If gene i and j both contain sequences for at least 3 equal taxa, node i and j are connected by an edge. If the set of nodes and inserted edges represents a connected graph, the connectivity criterion holds. A taxon coverage of 3 was chosen since Steel and Sanderson (2010) showed that 3 overlapping taxa are at least required to consistently build a correct supertree of correct subtrees. Next, an optimality criterion is calculated, which is defined by the following function:

$$f(\mathbf{P}) = 1 - |(\lambda - \mathbf{P}^{\alpha \times (1-\mathbf{P})})| \text{ if } \mathbf{P} < 1 \tag{6}$$

with α as a scaling factor (default = 3) and λ as the size ratio between the reduced matrix $\mathbf{B'}$ and the original matrix $\mathbf{B}$. For the first iteration holds $\mathbf{B'} = \mathbf{B}$. Thus, $\lambda = 1$ since the matrix has not been reduced yet. If the connectivity criterion holds, $f(\mathbf{P'})$ and its corresponding matrix $\mathbf{B'}$ are memorized. The taxon or the gene with the lowest information content ($\mathbf{p}_i$ or $\mathbf{q}_j$, respectively) is excluded. In case of ties, genes will be excluded. Since taxa or genes with the lowest information content will be dropped, $\mathbf{P'} > \mathbf{P}$. Every gene with less than 4 taxa is automatically dropped from the matrix. The following iterations consist of recalculating $\mathbf{p}_i$, $\mathbf{q}_j$, $\mathbf{P'}$, λ and $f(\mathbf{P'})$ plus the exclusion of the gene or taxon with the lowest information content until 1 gene and 4 taxa are left. $\mathbf{B'}$ and $f(\mathbf{P'})$ are kept in memory if the connectivity criterion is fulfilled and if the current result of $f(\mathbf{P'})$ is higher than in the previous iteration. If $f(\mathbf{P'}) = 1$, reduction stops. Since $\mathbf{P'}$ of consecutive $\mathbf{B'}$ continuously increases and λ continuously decreases, the outlined procedure is a simple hill climbing heuristics, which identifies a quasi-biclique with high information content of the matrix. The reduced matrix with highest information content finally contains all taxa and genes that remain in the superalignment. However, because of the interaction of $\mathbf{p}_i$ and $\mathbf{q}_j$, it is not guaranteed to find a global optimum. Thus, MARE can be regarded as a fast approach to approximate optimal subsets of taxa and genes suitable for phylogenetic analyses. The reduction is time efficient with a time complexity of $O(m + n)$.

1.3 Parameter settings

The data availability matrix is reduced in a deterministic way and thus under equal parameter settings MARE will deliver equal results. In order to fit the reduction to datasets with different properties, the reduction can be started with different values of α (default = 3) as scaling factor in $f(\mathbf{P})$ and with/without weighting of taxa. The weighting of taxa is achieved by multiplying the information content of each taxon $\mathbf{p}_i$ with a user defined value > 1 and leads to the retention of more taxa. Changing α has an effect on the size and the information content $\mathbf{P'}$ of the reduced matrix $\mathbf{B'}$. If α is increased, the resulting matrix decreases in size, whereas $\mathbf{P'}$ increases.

2 Program Usage

2.1 Requirements

MARE is written in C++ and currently works on Linux systems. To compile the source code, a g++ compiler is required. To start MARE

1. Open a terminal
2. Unpack MARE
3. Change to the unpacked directory MARE
4. Type 'make' and press <enter>
5. Now, the created binary 'MARE' can be moved to any other directory.

2.2 Input Format

MARE requires two files as command line arguments. The first one determines start and end of each partition in the concatenated multiple sequence alignment (MSA). It must be a plain text file and must have the following format:

charset YourFirstPartition = 1 - 100 ;
charset YourSecondPartition = 101 - 255 ;
...

The second file must be a concatenated multiple sequence alignment of amino acid sequences in non-interleaved FASTA format. Make sure to use UNIX linefeeds in your input files! NOTE: Your charset and your FASTA file must end with a UNIX linefeed: thus the last sequence must also include a linefeed. You can easily check this by using advanced text editors (e.g. `Notepad++`, `Geany` or `SciTE`).

2.3 Running MARE

To run MARE, enter

./MARE charset_file fasta_file [options]

While MARE is running, some information (e.g. information content of each partition from your data set) is dispayed in the terminal. By adding '> logfile.txt' at the end of your MARE command, you create a logfile, and instead using the terminal, the information is written into a plain text file

'logfile.txt'. (NOTE: this is not a MARE command, but a standard command which can be used for any command line program to write information which is displayed in the terminal as standard output, in a file!)

2.4 Output files

MARE results are stored in a directory called *results*. The directory *results* contains the following files:

∗.fas_reduced	reduced multiple sequence alignment
∗.charset_reduced	reduced corresponding charset file
∗_info.txt	list of chosen parameters, matrix saturation, number of taxa/partitions and average information content of the unreduced and reduced matrices
∗matrix.txt	plain text file of the unreduced data matrix with calculated tree-likeness values; can be used to start another MARE reduction without recalculation. Note: In the matrix, absent entries are coded as -1 to distinguish them from entries with a tree-likeness value = 0.
∗matrix_red.txt	plain text file of the optimized data matrix with calculated tree-likeness values
∗.svg	chosen graphical output (see Options)
∗_plot.txt	contains for each reduction step (*red*) the information content P' (#p), size ratio of reduced and unreduced matrix (*lam*), optimality criterion $f(\boldsymbol{P'})$ (f), number of taxa (taxa) and number of genes/partitions (parts). Additionally, the number of clusters of genes connected by three taxa is listed.

∗: filename without extension

2.5 Options

-c [taxon_name]

It is possible to constrain one or more taxa. These taxa cannot be dropped from the matrix during reduction. To use this option, enter the FASTA header of the taxon without '>':

-c taxon_1 -c taxon_2 -c taxon_3

etc.

-g [gene_name]

It is possible to constrain one or more genes. These genes cannot be dropped from the matrix during reduction. To use this option, enter the gene name of the charset file:

-g gene_1 -g gene_2 -g gene_3

etc.

-d [alpha]

By choosing this option, the default weighting of information content of the matrix in MARE (default = 3) is changed to alpha. Generally, higher weights lead to smaller subsets of taxa and genes/partitions with higher information content of the matrix. Lower weights lead to larger subsets with less information content. To get useful subsets, alpha should not exceed 5 or deceed 1.

-h

This option returns a short helpfile to get started with MARE.

-m

This option outputs the reduced and unreduced matrix as *.svg.

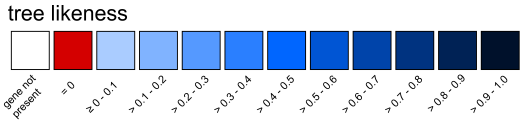

Figure 3: Color code for the graphical matrix output. Genes which are not present (in the plain matrix coded as -1) are white colored. Genes with a tree-likeness = 0: red. The darker the blue, the higher is the tree-likeness.

-r [filename]

By choosing this option, a precalculated data matrix as delivered by MARE (*_matrix.txt) can be used instead of calculating tree-likeness values again. It is required that taxa and genes in the matrix have the same order as in the multiple sequence alignment and the charset file.

-s

This option outputs the simplex graphs as *.svg. WARNING: A *.svg of a simplex graph requires more than 3 MB storage space. So, 1,000 partitions (genes) will take more than 3 GB.

-t [taxon_weight]

This option causes a stronger weighting of taxa while reduction (default = 1). Only values ≥ 1 are allowed. If taxon_weight > 1, more taxa remain in the resulting alignment.

An example, how to use some options in MARE:

./MARE file.charset file.fasta -c Musca_domestica -c Tenebrio_molitor -t 1.5 -g gene11435g -m

Two taxa (*Musca domestica* and *Tenebrio molitor*) and one partition (gene11435g) are retained (see test files), taxa are weighted by a factor of 1.5 and matrices are additionally delivered as $*$.svg. The order of chosen options does not have any influence.

2.6 License/Help-Desk/Citation

MARE version 0.1-rc was developed by Benjamin Meyer in 2010. In MARE version 0.1.2-rc, small bugs and linefeed problems were fixed, error-reports added and the color code has slightly changed by Benjamin Meyer and Karen Meusemann in April 2011. MARE is implemented in C++ and freely available from http://mare.zfmk.de. It can be distributed and/or modified under the terms of the GNU General Public License as published by the Free Software Foundation; either version 3 of the license, or (at

your option) any later version. This program is distributed with the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.
If you have any problems, error-reports or other questions about MARE, feel free to write an email to mail2mare@gmx.de which is the official help desk email account for MARE. Further programs are available from http://software.zfmk.de.

If you use MARE, please cite:
MARE: MAtrix REduction - A tool to select optimized data subsets from supermatrices for phylogenetic inference. Meyer, B. and Misof, B. Zentrum für molekulare Biodiversitätsforschung (zmb) am ZFMK, Adenauerallee 160, 53113 Bonn, Germany. Version 0.1-rc. December 2010.

MARE: MAtrix REduction - A tool to select optimized data subsets from supermatrices for phylogenetic inference. Meyer, B., Meusemann, K. and Misof, B. Zentrum für molekulare Biodiversitätsforschung (zmb) am ZFMK, Adenauerallee 160, 53113 Bonn, Germany. Version 0.1.2-rc. April 2011.

A pre-alpha-version of MARE (Perl), developed by B. Misof has been used for the first time in: Meusemann *et al.* (2010). A phylogenomic approach to resolve the arthropod tree of life. *Molecular Biology & Evolution*, 27(11):2451–2464, 2010. doi: 10.1093/molbev/msq130.

2.7 Release Notes

- December 2010: release vo MARE v0.1-rc
- January 2011: small bug in the provided test.fas file fixed
- April 2011: color code slightly changed: genes with a tree-likeness = 0: red colored

2.8 Copyright

2.9 Acknowledgements

We thank Ralph Peters, Biozentrum Grindel, University of Hamburg for useful comments, and Daniela Bartel, University of Vienna for debugging MARE version 0.1-rc. Also thanks to Karen Meusemann for help with the Manual, suggestions for the MARE version 0.1-rc and 0.1.2-rc developement and debugging. We thank also Markus Gö, Department of Microbiology, DSMZ - German Collection of Microorganisms and Cell Cultures for helpful comments to fix some bugs.

References

Steven Henikoff and Jorja. G. Henikoff. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences of the United States of America*, 89:10915–10919, 1992.

Kay Nieselt-Struwe and Arndt von Haeseler. Quartet-mapping, a generalization of the likelihood-mapping procedure. *Molecular Biology & Evolution*, 18(7):1204–1219, 2001. ISSN 0737-4038.

Mike Steel and Michael J. Sanderson. Characterizing phylogenetically decisive taxon coverage. *Applied Mathematics Letters*, 23(1):82–86, 2010. doi: 10.1016/j.aml.2009.08.009.