TABAJARA

A tool for rational design of profile HMMs

A Quick Guide and Tutorial

TABAJARA - A Quick Guide and Tutorial

1 Introduction

TABAJARA is a tool for rational design of profile HMMs. Starting from a multiple sequence alignment (MSA), TABAJARA is able to find blocks that are either (1) conserved in all sequences or (2) discriminative for two specific groups of sequences (Figure 1). For the identification of regions conserved across all protein sequences of an MSA, we implemented a previously described algorithm (Capra & Singh, 2007), based on Jensen–Shannon divergence method (Lin, 1991). This is the method of choice to determine the level of character conservation across all sequences of an MSA. In the case of nucleotide sequences, TABAJARA uses Shannon entropy (Shannon, 1948; Shannon & Weaver, 1949) to calculate position-specific scores. To find group-discriminative blocks, TABAJARA uses a combination of Mutual Information (Adami, 2004; Cover & Thomas, 2006) and Sequence Harmony (Feenstra *et al*., 2007; Pirovano *et al*., 2006) for both, DNA and protein sequences.

What does the name TABAJARA mean? The classical definition is that TABAJARA is the name of one of Tupi tribes of indigenous people, the Tabajaras, that used to live along the Atlantic coast of northeastern Brazil. An alternative definition is that TABAJARA is the name of a fictional company (TABAJARA Organization) created by a Brazilian humoristic TV show in the 1990's, associated with products of doubtful utility and sensationalist advertising. In our view, TABAJARA is an acronym for **T**ool for **A**lignment **B**lock **A**nalysis **J**oining **A**ppropriate **R**ational **A**pproaches. We let the users to choose their favorite definition.

2 Main features

- Fully configurable
- Multiple sequence alignments of either protein or nucleotide (still experimental) sequences can be used
- Identification of the most conserved regions from a set of functionally and taxonomically related proteins using Jensen-Shannon divergence for protein sequences and Shannon entropy for nucleotide sequences

- Identification of group-specific regions from a set of functionally and taxonomically related proteins using combined measures of Mutual Information (MI) and Sequence Harmony (SH)
- Automatic elimination of redundant sequences and gap-only columns from MSAs
- Automatic choice of the best candidate alignment blocks
- Automatic construction of profile HMMs from selected alignment blocks
- Generates different types of text reports

3 The algorithm

`TABAJARA` uses MSA files in both FASTA and Clustal (`aln`) formats as input. According to the user's choice, the program can detect identical sequences and leave only one representative of each identical group. This step avoids biasing the alignment block selection and profile HMM construction. By default, `TABAJARA` identifies and removes gap-only columns. Depending on the dataset (nucleotide or protein sequences) and the chosen task (detecting sequence-conserved or group-specific regions), `TABAJARA` uses one of the aforementioned methods (see Introduction) to calculate position-specific scores across the entire alignment. Once position-specific scores are determined, the program uses a sliding-window to screen the whole alignment and delimit top-scoring regions. A second step can be performed using a smaller sliding window to identify short high-scoring regions. This is particularly interesting if regions highly specific to a particular subset of sequences are sought. The program automatically extracts the selected alignment blocks, discards identical sequences, eliminates gap-only columns and builds the corresponding profile HMMs, which can in turn be used for downstream applications.

4 Self-validation of the models

After building the profile HMMs, `TABAJARA` executes a series of validation steps to discard models that do not fulfill a series of quality criteria. First, `TABAJARA` converts the training set MSA file into a plain-sequence FASTA file without gaps. Either in Conservation or in Discrimination mode, all models are submitted to similarity searches against all sequences of the training set using `hmmsearch` program. After saving the output in both tabular and complete formats, `TABAJARA` inspects the results and checks whether the models fulfilled the

quality control criteria defined by the user. Approved models are stored, while the remaining are discarded.

4.1 Validating models for conservation

TABAJARA can be executed in Conservation mode to generate models that should ideally detect all sequences used in the training set. To validate the profile HMMs, TABAJARA invokes hmmsearch to run similarity searches of the models against the sequences of the training set, using default parameters. TABAJARA determines for each model whether the percentage of detected sequences is equal to or greater than the value defined in parameter pd (see below). If the user declares a list of category prefixes (e.g. DENV and ZIKV) in parameter c (see below), then a valid model is required to detect a percentage of all sequences, plus the same percentage of sequences of each individual category. For example, being a dataset composed of 50 sequences of dengue virus, 40 of Zika virus and 310 sequences of other flaviviruses, if the user chooses parameters c DENV,ZIKV and -pd 80, TABAJARA will only accept models that detect at least 80% of dengue (≥ 40) and Zika (≥ 32) sequences, in addition of detecting at least 80% of the total number of sequences of the training set (80% of 400 - ≥ 320 sequences). Finally, if parameter cs yes is used, TABAJARA inserts a CUTOFF SCORE tag in the profile HMM, using a value corresponding to 80% of the score of the last similarity hit of the training set.

4.2 Validating models for discrimination

When running in Discrimination mode, TABAJARA also validates the models following a series of criteria. Given a dataset composed of sequences of the chosen category (group A) and the remaining sequences (group B), TABAJARA runs hmmsearch with default parameters using each model as a query against the training set, including sequences of groups A and B. The first requirement for a model to be accepted is that the top hit sequence must be a member of group A, otherwise the model is discarded. If all positive sequences belong to the chosen category (group A) and parameter cs yes is used, TABAJARA inserts a CUTOFF SCORE tag in the profile HMM, corresponding to a percentage (defined by parameter pt) of the lowest score.

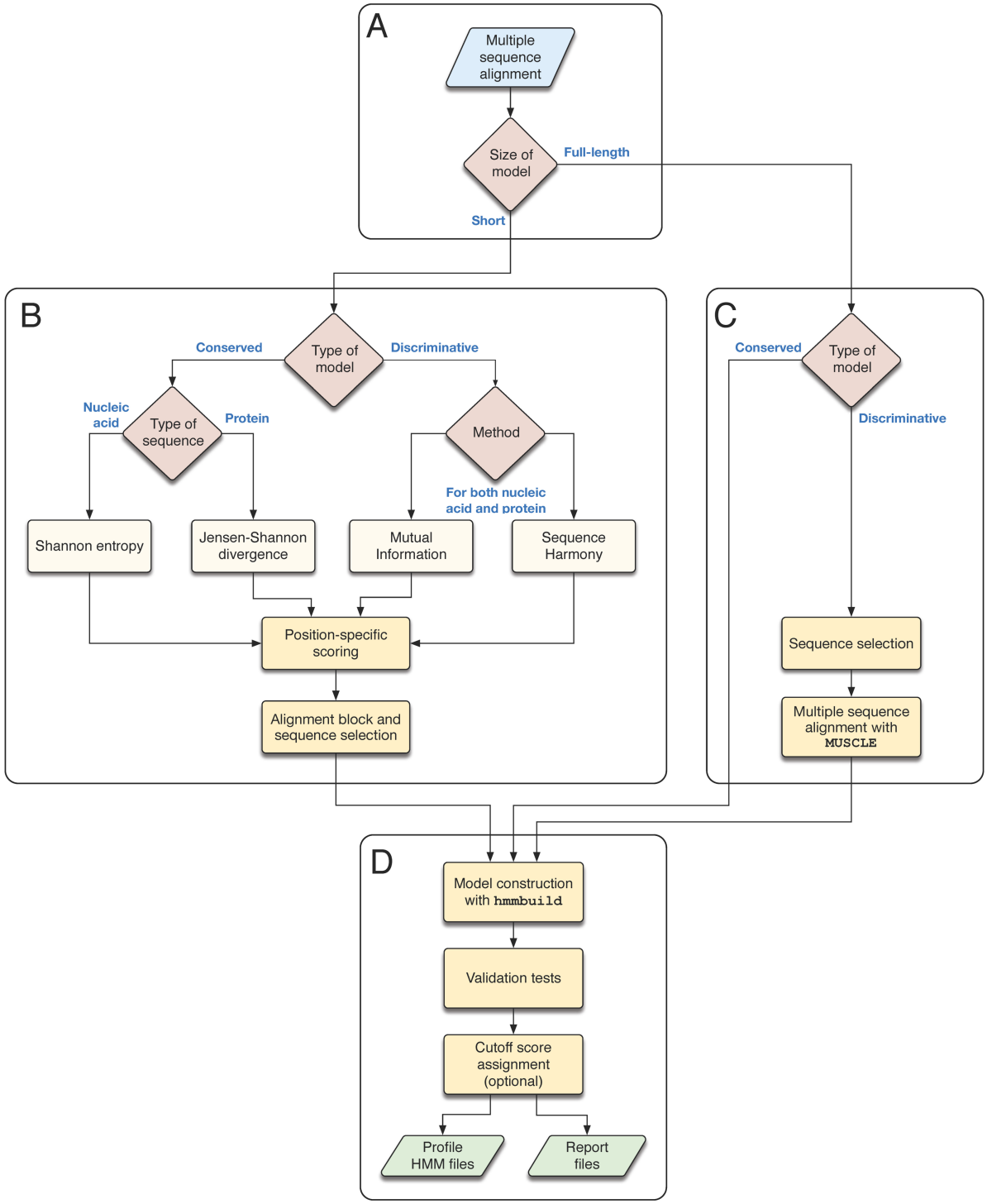


Figure 1 – TABAJARA uses a multiple sequence alignment (MSA) as an input. In Conservation mode, alignment blocks are selected for regions conserved in all sequences using Shannon entropy for nucleic acid and Jensen-Shannon divergence for protein sequences. In Discrimination mode, TABAJARA uses a combination of Mutual Information and Sequence Harmony. Selected alignment blocks are used to build profile HMMs with hmmbuild program. All profile HMMs are then submitted to validation tests and the selected models are stored, together with some report files of the pipeline. Models can also be built directly from the full-length MSA. In this case there is no selection of alignment regions and the models are submitted to the same validation tests.

There are many instances where hits corresponding to sequences of group B can also be found. TABAJARA screens the list of hits in decreasing order of score and calculates the ratio of the score of the first positive sequence of group B to the score of the immediately preceding sequence of group A. This value must be equal to or lower than the value defined by parameter pt, otherwise the model is discarded. For example, given the list of hmmsearch results below and assuming that category GroupA is the chosen group, the ratio of 33.8 to 153.7 would be 21.99. Assuming a pt value of 80, this model would be accepted.

Sequence	**Score**
GroupA_AGX32058.1	158
GroupA_AGX32061.1	155.4
GroupA_AFR34023.1	154.8
GroupA_AGX32060.1	154.6
GroupA_AGX32059.1	154.6
GroupA_AGX32057.1	153.7
GroupB_NC_022595_Murrumbidgee	33.8
GroupB_KJ481929_Buffalo_Creek	33.7
GroupB_KJ867181_Pongola	28.2
GroupB_KJ481923_Mapputta	27.6
GroupB_KF981633_Khurdun	26.1
GroupB_KM272183_Calchaqui	25.1
GroupB_KT950264_Alajuela_GAM_122	24.4

If the criterion above is fulfilled, and parameter cs yes is used, TABAJARA inserts a CUTOFF SCORE tag in the profile HMM. The program now calculates the absolute difference between the scores of the first positive sequence of group B (33.8) and the immediately preceding sequence of group A (153.7). A value corresponding to 80% of this difference (in this case 0.8 x 119.9 = 95.92) is added to the score of the top performer sequence of group B (33.8). Thus, is this example the cutoff score would be 129.72. The rationale behind such heuristics is to guarantee that sequences of the chosen category that present a slightly lower score (80%) would be still detected by the model, while avoiding the detection of sequences not belonging to this category. Finally, using the assigned cutoff values of each model,

TABAJARA verifies whether the percentage of detected sequences of group A is equal to or higher than the value defined by parameter pd. Only models that meet this condition are accepted and stored.

5 Before using TABAJARA

5.1 System requirements

TABAJARA was developed in Perl language and can be used in any POSIX compliant operating system such as UNIX and Linux distributions with an installed Perl interpreter (http://www.perl.org).

5.2 Third-party programs

TABAJARA requires the program hmmbuild (HMMER3 package - (http://hmmer.org/) to build profile HMMs. The program must be located in a directory listed in the PATH of the operating system.

5.3 How to cite

If you use TABAJARA for your publication, please cite the program as:

TABAJARA program (developed by Liliane S. Oliveira and Arthur Gruber, University of São Paulo, Brazil, manuscript in preparation; Ibrahim *et al*., 2018)

Ibrahim, B. et al. (2018). Bioinformatics Meets Virology: The European Virus Bioinformatics Center's Second Annual Meeting. Viruses 10(5):256. doi: 10.3390/v10050256.

6 Understanding TABAJARA parameters

6.1 Mandatory parameters:

TABAJARA has six mandatory parameters that must be specified by the user at the command line (or in the configuration file):

-i <file name>: Input file (multiple sequence alignment - FASTA and Clustal formats are accepted)

-m <c|d>: Method to generate position-specific scores. Options: c (conservation) and d (discrimination).

TABAJARA uses several different methods to calculate position-specific scores, according to the global level of character conservation along all positions of the MSA. Alternatively, the program can evaluate each position for its ability of discriminating two groups of sequences. In this case, TABAJARA uses a combination of two distinct methods, Mutual Information or Sequence Harmony. The following options are valid for parameter m:

c – Conservation. TABAJARA automatically detects the nature of the sequences comprising the MSA. For protein sequences, TABAJARA uses a previously described algorithm (Capra & Singh, 2007), based on Jensen–Shannon divergence method (Lin, 1991). In the case of nucleotide sequences, TABAJARA calculates Shannon entropy (Shannon, 1948; Shannon & Weaver, 1949) and converts the results to normalized Shannon entropy values (Kumar *et al*., 1986). In both cases, position-specific scores are calculated to estimate the overall level of character conservation.

d - Discrimination. TABAJARA calculates position-specific scores using a combination of Mutual Information, as described by Adami (2004) and Cover & Thomas (2006), and complements of Sequence Harmony (Pirovano *et al*., 2006; Feenstra *et al*., 2007) values. Using a sliding window approach, the best regions are selected for each method. Method-specific are stored and in case of overlaps between regions of different methods, these regions are merged and TABAJARA calculates an average score value for each position. Finally, an additional sliding window step is used to find the top scoring window within each merged region.

NOTE:

When using `-m d` (Discrimination mode), the parameter `c` becomes mandatory:

`-c` `<string>`: Name of the category(ies) to be analyzed. Ex. `Alpavirinae`. Sequences belonging to a category are identified in the FASTA header by a prefix composed of an initial string of characters delimited at the end by an underline character. For example, the MSA of *Microviridae* sequences, provided as a dataset for this tutorial, presents the following naming format:

```
>Alpavirinae_Human_feces_C_016
>Gokushovirinae_Human_gut_34_012
>Pichovirinae_Pavin_279
```

For example, using `Alpavirinae` as the string of parameter `c` will imply that all sequences whose headers start with `Alpavirinae` will be treated as members of a single group, while all remaining sequences will compose a second group. Multiple categories can be specified, separated by commas:

```
-c Alpavirinae,Gokushovirinae
```

In this example, after processing and finding `Alpavirinae`-specific blocks, `TABAJARA` will repeat the process for `Gokushovirinae`, treating the respective subsets of sequences as groups to be compared against all remaining sequences.

Name prefixes are also used to guide the model validation step (see section “Self-validation of the models”). When using `TABAJARA` in Conservation mode, declaring sequence categories is optional. However, if these prefixes are used, the program will report the detection rate of the models, discriminated according to each group of sequences. If prefixes are not declared, then `TABAJARA` will only report the detection rate in regard to the whole set of sequences of the training set.

IMPORTANT:

When using `-fl no` (construction of short models - see below), the following parameters are also mandatory:

`-b <integer>`: Minimum block size (value must be ≥ `w`)

`-p <integer>`: Percentage of positions in sliding window with score ≥`t`

`-t <decimal>`: Score threshold of the alignment position for block extraction (valid values: 0 to 1)

`-w <integer>`: Window size for block extraction

6.2 Optional parameters:

`-clean <yes|no>` (default = `yes`) - Delete some processing files.

`-conf <configuration file>` - use a configuration file that lists all parameters for execution, overriding any parameter of the command line. An example of a configuration file follows below:

```
i=peribunyaviridae.aln
o=output_dir
m=b
t=0.5
p=50
c=Alpavirinae,Gokushovirinae
w=15
sa=1
b=20
md=3
gs=20
wg=yes
di=yes
cs=yes
pd=80
gc=30
wg=yes
pt=80
sv=0.8
mb=60
sa=1
sr=yes
```

`-cs <yes|no>`: Insert cutoff scores in the profile HMMs (default = `no`)

`-di <yes|no>`: Discard identical sequences (default = `no`).

-fl <yes|no>: Use full-length sequence for model construction (default = no). IMPORTANT: When using -fl no (default), the following parameters are mandatory: m (method), t (threshold), p (perc3ntage) and w (window size).

-gc <integer>: Gap cutoff (default=30). Percentage of allowed gaps in each position of the MSA. If most of the sequences in a given position are represented by gap characters, this position may introduce noise to the model (the profile HMM). To avoid this occurrence, positions presenting higher percentages than the gc value will receive a zero score. If parameter -wg yes is used, any window containing zero-score positions will be discarded. Therefore, this former parameter allows to discard blocks containing any number of gaps. In Conservation mode, the percentage of gaps in a column is calculated using all sequences of the alignment. Conversely, in Discrimination mode, calculation only takes into account the sequences of the chosen category. This latter feature allows to select group-specific blocks that are absent in the remaining sequences (they correspond to an inserted region).

-gs <integer>: Percentage of allowed gaps in each sequence of the selected alignment block. Any sequence presenting higher percentages of gap characters than gs is discarded from the MSA and not used in the construction of the profile HMM.

-h: Print a help screen.

-hb <string>: Any set of hmmbuild's valid parameters can be entered under double quotes. Example: -hb "--wblosum -wid 0.8". Default: no parameters.

-mb <integer>: Maximum block size. Given an alignment block with a minimum length defined by parameter b, the user can also define the maximum length of this block. TABAJARA scans the region with another sliding window with -mb length and then selects the sub-region which presents the highest score sum.

-md <integer>: Maximum accepted distance for two alignment blocks to be joined as a single block. If TABAJARA finds two contiguous alignment blocks that are separated by a distance equal or shorter than md, the program will join them into a single block, otherwise the blocks will be stored separately.

-o <file name>: Output directory (default = output_dir_#).

-pc <integer>: Minimum percentage of categories in the training set that must meet the criteria defined by the parameter pd for profile HMMs built in Conservation mode (default = 80).

-pd <integer>: Minimum detection rate (in percentage) of training set (MSA) sequences by the constructed profile HMM. This is equivalent to the minimum accepted sensitivity of the model (default = 80).

-pt <integer>: This parameter applies for the validation of HMMs designed for group discrimination. Given two groups, A (selected category) and B (remaining sequences), an hmmsearch is performed with the constructed model against the training set (MSA) sequences. The highest score obtained by group B sequences must be lower than n % (defined by parameter pt) of the lowest score obtained by group A sequences, otherwise the model is not accepted (default = 80).

-sa <integer>: Minimum allowed number of sequences to build a profile HMM (default = 2). This value applies to the original MSA and to the post-scoring selected alignment blocks. In both cases, the redundant sequences are identified and discarded. TABAJARA only builds a profile HMM from an alignment if the total number of sequences remaining after redundancy removal is equal or greater than the number defined by parameter sa. In general, no model should be built with less than two sequences. In fact, hmmbuild, the program that constructs the models, does not accept a single sequence as input. However, if a value of 1 is specified, TABAJARA creates multiple copies of the single sequence and this pseudo set is used to build the profile HMM.

-sr <yes|no>: If a maximum block size (parameter mb) is defined, in addition to the selection of the best alignment block, the program also screens the remaining stretches of sequence to find additional blocks that fulfill the specified scoring selection criteria. This option maximizes the number of models that can be built from a high-scoring region (default = no).

-sv <integer>: Minimum accepted score per alignment position to be ascribed in the CUTOFF SCORE tag of the profile HMM (default = 1). For instance, if a -sv 1 is used for a 20-position block, then the ascribed CUTOFF SCORE will be 20. This parameter supersedes the standard CUTOFF SCORE calculated in the model validation step (see parameter -pt), thus avoiding too low cutoff values.

`-v:` Print the program version.

`-wg <yes|no>`: Discard sliding windows presenting gaps (default = `no`).

7 Running TABAJARA

To install the data for the tutorials, copy the file `tutorial.tar.gz` to a directory of your choice. Decompress the file using the following command:

```
tar xzvf tutorial.tar.gz
```

This command will create the `tutorial` directory. This directory contains the following items:

- Subdirectory `data` - contains two multiple sequence alignment files:
  - `microviridae.fasta` – contains 83 protein sequences of VP1 (major capsid protein) of *Microviridae* phages (Roux *et al.,* 2012).
  - `flavivirus.fasta` – contains 127 polyprotein sequences from viruses of the *Flavivirus* genus, including 10 of ZIKV, 10 of YFV and 20 of DENV (5 of each serotype).
- Subdirectory `cnf` – contains five configuration files for running the TABAJARA program in conservation and discriminative modes for *Microviridae* phages and *Flavivirus* genus.

7.1 Example 1 – Finding regions conserved in all sequences of a multiple sequence alignment of VP1 sequences of *Microviridae* phages

7.1.1 Executing TABAJARA

To execute `TABAJARA`, some configuration files can be used. For example, to run `TABAJARA` in Conservation mode and produce models can be used to generically detect all sequences of a dataset composed of VP1 protein of *Microviridae* phages. We can use the command below:

```
tabajara.pl -conf cnf/cons_microviridae.cnf
```

This configuration file includes the following content:

```
input_file=data/microviridae.fasta
threshold=0.5
percentage=50
window_size=15
minimum_block_size=15
method=c
gap_sequence=20
maximum_distance=3
output=cons_microviridae_dir
cutoff_score=yes
maximum_block_size=20
```

Alternatively, we can execute TABAJARA with exactly the same parameters and options using the line command:

```
tabajara.pl -i data/microviridae.fasta -t 0.5 -p 50 -w 15 -b 15 -mb 15 -m c -gs 20 -md 3 -o cons_microviridae_dir -cs yes -mb 20
```

7.1.2 Understanding the parameters

In a first step, the program estimates the conservation level across all sequences (parameter `-m c`) in all positions of the MSA, using the algorithm described by Capra & Singh (2007). Following scoring, TABAJARA runs a 15-base sliding window (parameter `-w`) where at least 50% of the bases (parameter `-p`) must have scores higher than 0.5 (parameter `-t`). If two contiguous alignment blocks are located up to 3 bases from each other (parameter `-md`), they are joined into a single block, otherwise they are treated separately. Finally, the program scans the selected blocks using a 15-base sliding window (parameter `-mb`) and calculates the score sums base by base. The window presenting the highest score sum is then used to select the final alignment region from each previously designated block. This second sliding window step allows to refine and restrict the selected region to a short and specific alignment block. This short alignment block is then extracted from the MSA. Gap-only columns are deleted and sequences presenting more than 20% gap characters (parameter `-gs`) are discarded. The

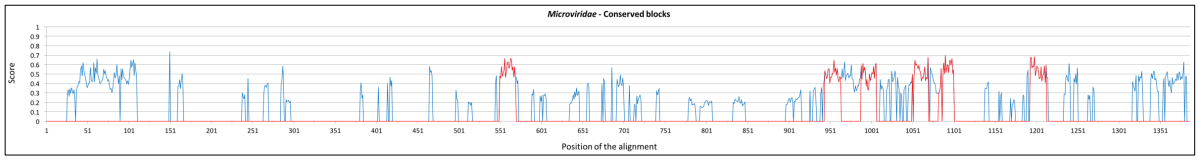
extracted MSA block is used as input by `hmmbuild` to construct a profile HMM. Since the parameter `-cs yes` is used, `TABAJARA` inserts cutoff score tags in the header section of the profile HMMs Finally, all output files are stored within the specified output directory (parameter `-o`).

7.1.3 Inspecting the output files

Once `TABAJARA` finishes the processing, an output directory is created. Within this directory, you will find several files and subdirectories created by `TABAJARA`:

- `logfile.txt`: a log file created by `TABAJARA` on every run. This file stores all the parameters utilized to run `TABAJARA`. A list of all identified alignment blocks and selected regions are also stored.
- `scores.csv`: this file is composed of tab-separated columns listing all positions of the MSA and their corresponding scores, respectively. Score values, ranging from zero to one, are ascribed according to the chosen scoring method. An additional column lists those scores restricted to positions of the alignment blocks selected by the chosen method (Conservation, Mutual Information or Sequence Harmony), with all remaining positions presenting a score of zero. If `–m b` option is used, three additional columns are presented, corresponding to scores of Mutual Information, Sequence Harmony and the combination of both methods, respectively. This file can be used as input for a spreadsheet, to easily generate a graph depicting scores along the MSA, with selected blocks highlighted in a different color, as exemplified in Figure 2.

A



B

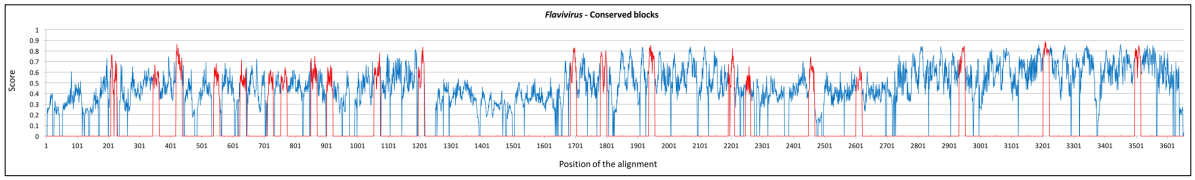


Figure 2 – Position-specific scores along multiple sequence alignments (MSA) of 83 VP1 protein sequences of *Microviridae* phages (A) and 127 *Flavivirus* polyprotein sequences

(B). Selected regions conserved across all sequences of the MSA, according to Jensen–Shannon divergence, are indicated in red.

- `selected_blocks.txt`: this text file lists the MSA coordinates of all selected alignment blocks.
- `original_blocks`: this directory contains multiple-sequence FASTA files, each one containing sequences restricted to the coordinates of extracted alignment blocks. These files may contain identical sequences.
- `blocks_without_redundancy`: this directory contains files that are similar to those within `original_blocks` directory, but without identical sequences and gap-only columns.
- `hmms`: contains two subdirectories `validation` and `valid_HMMs`. The directory `validation` contains the validation tests of each models, including the hmmsearch results and the `results.csv`, that present the quantitative results of the models (number and percentage of detected sequences). The directory `valid_HMMs` contains all validated models. Files are named with a prefix of the respective category, a number corresponding to the location order in the MSA and respective initial and end coordinates of the source alignment blocks.
- In case `TABAJARA` finds identical sequences and/or gap-only columns in the original MSA file, the program also stores an MSA without these occurrences. The naming is composed of the original file name followed by the suffix `_no_redundancy.fasta`.

7.2 Example 2 – Finding regions conserved in all sequences of a multiple sequence alignment of polyprotein sequences of viruses of the genus *Flavivirus*

We will use the MSA provided in this tutorial (file `flavivirus.fasta`) to select alignment blocks and build profile HMMs that are generic to members of the genus *Flavivirus*.

To run `TABAJARA` in Conservation mode for *Flavivirus* sequences, we will use the configuration file `cons_flavivirus.cnf`:

```
tabajara.pl -conf cnf/cons_flavivirus.cnf
```

This configuration file includes the following content:

```
input_file=data/flavivirus.fasta
threshold=0.5
percentage=50
window_size=15
minimum_block_size=20
method=c
category=DENV,ZIKV,YFV
gap_sequence=20
maximum_distance=3
output=cons_flavivirus_dir
cutoff_score=yes
maximum_block_size =60
```

Alternatively, we can execute `TABAJARA` with exactly the same parameters and options using the line command below:

```
tabajara.pl -i data/flavivirus.fasta -t 0.5 -p 50 -w 15 -b 20 -m c -c DENV,ZIKV,YFV -gs 20 -md 3 -o cons_flavivirus_dir -cs yes -mb 60
```

This execution will produce profile HMMs that detect at least 80% of the sequences of the training set. The validation step also determines the detection rate of each of the declared categories. In this case, for a model to be accepted, in addition to the 80% overall detection rate, it must also detect at least 80% of the sequences from each one of the individual categories (DENV, ZIKV and YFV species).

7.3 Example 3 – Finding discriminative regions for the subfamilies *Alpavirinae* and *Gokushovirinae* (*Microviridae* family).

The command below executes `TABAJARA` using the MSA file `microviridae.fasta` as input. The parameter `-m d` is used For Discrimination mode. The parameter `–c` with the option `Alpavirinae,Gokushovirinae` indicates that all sequences starting with prefixes `Alpavirinae` or `Gokushovirinae` should be treated as independent groups and all

remaining sequences as another group. In this case, independent executions are automatically run for each specified category.

To run TABAJARA in Discrimination mode and produce models can be used to specifically detect phages of the subfamilies *Alpavirinae* and *Gokushovirinae*, we can use the command below:

```
TABAJARA.pl -conf cnf/disc_microviridae.cnf
```

This configuration file includes the following content:

```
input_file=data/microviridae.fasta
threshold=0.5
percentage=50
window_size=15
minimum_block_size=15
method=d
category=Alpavirinae,Gokushovirinae
gap_sequence=20
maximum_distance=3
output=disc_microviridae_dir
cutoff_score=yes
maximum_block_size=20
```

Alternatively, we can execute TABAJARA with exactly the same parameters and options using the line command:

```
tabajara.pl -i data/microviridae.fasta -t 0.5 -p 50 -w 15 -b 15 -mb 15 -m d -c Alpavirinae,Gokushovirinae -gs 20 -md 3 -o disc_microviridae_dir -cs yes -mb 20
```

As can be seen in Figure 3, alignment blocks specific for *Alpavirinae* and *Gokushovirinae* were found. Is it worth mentioning that some most of these regions are common to both subfamilies. This is not contradictory, but rather reflects the fact that some alignment regions are conserved within members of the same taxonomic groups, but are divergent from each other, allowing for discrimination between groups.

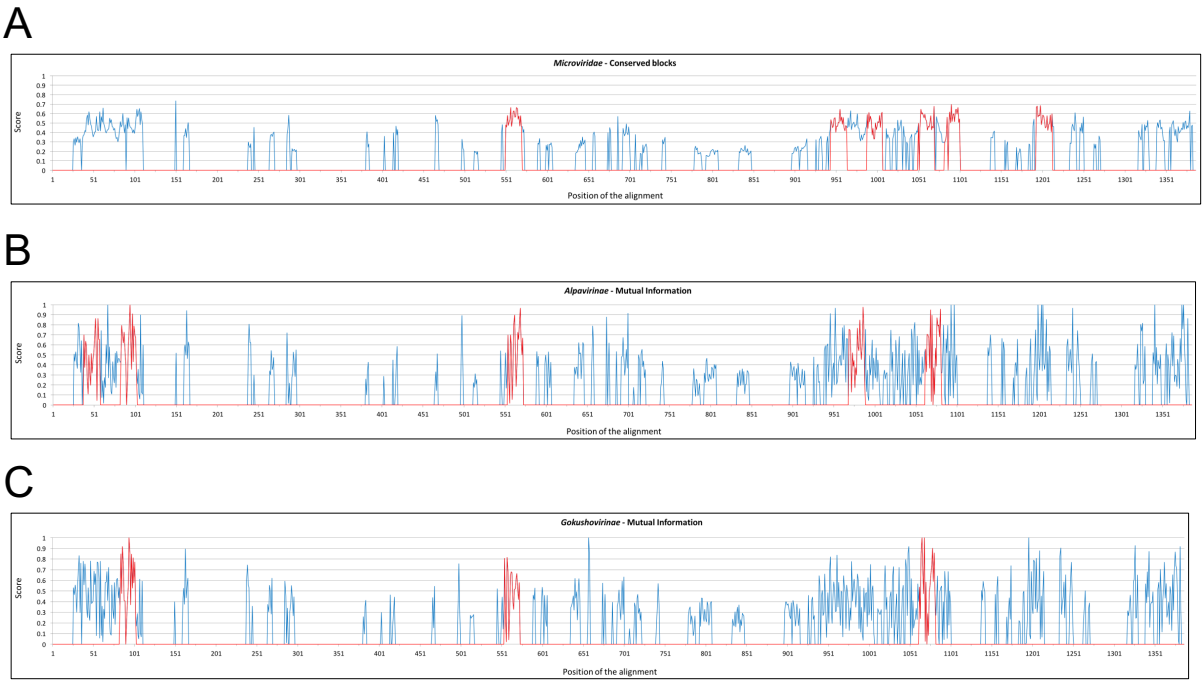


Figure 3 – Position-specific scores along a multiple sequence alignment (MSA) of 83 VP1 protein sequences of *Microviridae* phages. `TABAJARA` selected regions (indicated in red color) that are conserved across all *Microviridae* sequences (A), or regions specific to member of *Alpavirinae* and (B), *Gokushovirinae*.

7.4 Example 4 – Finding discriminative regions for Dengue, Zika and Yellow Fever viruses (genus *Flavivirus*)

Using the MSA provided in this tutorial (file `flavivirus.fasta`), it is possible to select regions and build profile HMMs that are specific for DENV, ZIKV and YFV. As can be seen in Figure 1, members of the genus *Flavivirus* (Fig. 4A) are much more conserved than viruses of the *Microviridae* family (Fig. 3A).

Let's execute `TABAJARA` in Discrimination mode using the configuration file `disc_flavivirus.cnf` to produce models can be used to especifically detect DENV, ZIKV and YFV:

```
TABAJARA.pl -conf cnf/disc_flavivirus.cnf
```

This configuration file includes the following content:

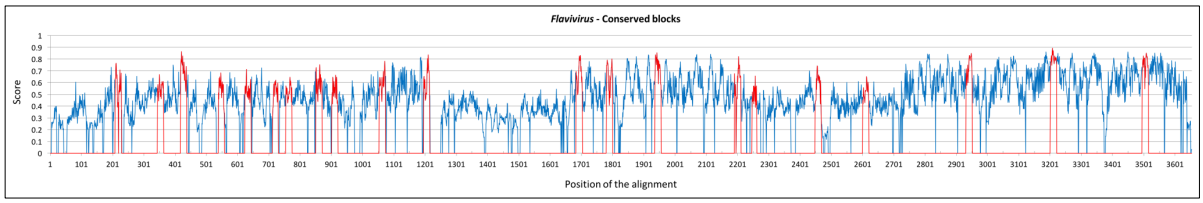
```
input_file=data/flavivirus.fasta
threshold=0.5
percentage=50
window_size=15
minimum_block_size=20
method=d
category=DENV,ZIKV,YFV
gap_sequence=20
maximum_distance=3
output=disc_flavivirus_dir
cutoff_score=yes
maximum_block_size =60
```

Alternatively, we can execute TABAJARA with exactly the same parameters and options using the line command:

```
tabajara.pl -i data/flavivirus.fasta -t 0.5 -p 50 -w 15 -b 20 -m d -c DENV,ZIKV,YFV -gs 20 -md 3 -o disc_flavivirus_dir -cs yes -mb 60
```

Figure 4B shows the results obtained for *Denguevirus* using Discrimination mode.

A



B

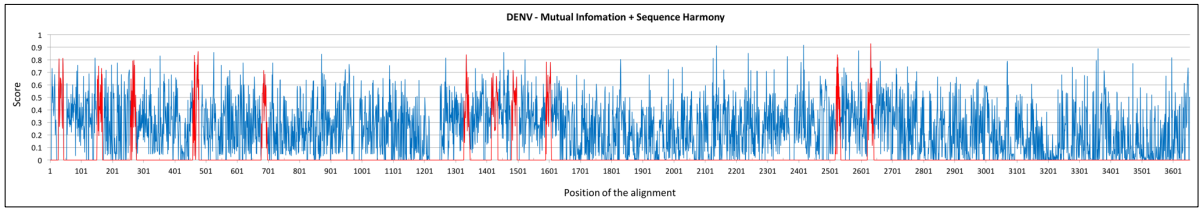


Figure 4 – Position-specific scores along a multiple sequence alignment (MSA) of 127 polyprotein sequences of members of the genus *Flavivirus*. TABAJARA selected regions (indicated in red color) that are conserved across all *Flavivirus* sequences based on Jensen–Shannon divergence (A), or regions specific to *Denguevirus* using a combination of Mutual Information and Sequence Harmony (B).

7.5 Example 5 – Building profile HMMs discriminative for Dengue, Zika and Yellow Fever viruses (genus *Flavivirus*) using full-length polyprotein sequences

Instead of finding the most informative alignment regions to build profile HMMs, TABAJARA also permits to construct discriminative models using the entire MSA, that is, using full-length sequences. In this case, the program just builds the model without measuring any position-specific score. Next, it performs the validation tests to evaluate the sensitivity and specificity of detection and assigns a cutoff score that allows a good balance between detection and discrimination rates.

To build full-length models that especifically detect DENV, ZIKV and YFV, we will execute TABAJARA in Discrimination mode using the configuration file disc_flavivirus_fl.cnf:

```
TABAJARA.pl -conf cnf/disc_flavivirus_fl.cnf
```

This configuration file includes the following content:

```
input_file=data/flavivirus.fasta
method=d
full_length=yes
category=DENV,ZIKV,YFV
output=disc_flavivirus_fl_dir
cutoff_score=yes
```

Alternatively, we can execute TABAJARA with exactly the same parameters and options using the line command:

```
tabajara.pl -i data/flavivirus.fasta -m d -fl yes -c DENV,ZIKV,YFV -o disc_flavivirus_fl_dir -cs yes
```

8 Using profile HMM seeds with GenSeed-HMM

You can use profile HMMs generated by TABAJARA as seeds for progressive assembly using metagenomic datasets and the program GenSeed-HMM (Alves *et al*., 2016). GenSeed-HMM is publicly available at https://sourceforge.net/projects/genseedhmm/. The program is fully

documented and tutorial, available on the site, provides information on some public metagenomic datasets of human fecal samples that can be used to reconstruct *Microviridae* phage genomes using profile HMMs and `GenSeed-HMM`. In case you find that profile HMMs build by `TABAJARA` are not specific to the chosen taxonomic group, try a higher stringency (e.g. increasing for example the values of parameters `-t`, `-w` and `-g`).

9 References

Adami C. Information theory in molecular biology. (2004). Phys Life Rev. 1: 3–22.

Alves JM, de Oliveira AL, Sandberg TO, Moreno-Gallego JL, de Toledo MA, de Moura EM, Oliveira LS, Durham AM, Mehnert DU, Zanotto PM, Reyes A & Gruber A. (2016) GenSeed-HMM: A Tool for Progressive Assembly Using Profile HMMs as Seeds and its Application in *Alpavirinae* Viral Discovery from Metagenomic Data. Front. Microbiol. 7: 269.

Capra JA & Singh M (2007). Predicting functionally important residues from sequence conservation. Bioinformatics 23: 1875–1882.

Cover TM & Thomas JA. (2006). Elements of Information Theory. Second edition. John Wiley & Sons, Inc., Hoboken, New Jersey.

Feenstra KA, Pirovano W, Krab K & Heringa J. (2007). Sequence harmony: detecting functional specificity from alignments. Nucleic Acids Res. 35(Web Server issue): W495-8.

Kumar, U., Kumar, V. & Kapur, J. N. (1986). Normalized measures of entropy. Int. J. Gen. Syst. 12: 55-69.

Lin, J. (1991) Divergence measures based on the Shannon entropy. IEEE Trans. Inf. Theory 37: 145–151.

Pirovano, W., Feenstra, K.A. & Heringa, J. (2006) Sequence comparison by sequence harmony identifies subtype specific functional sites. Nucleic Acids Res. 34: 6540–6548.

Roux S, Krupovic M, Poulet A, Debroas D & Enault F. (2012). Evolution and diversity of the *Microviridae* viral family through a collection of 81 new complete genomes assembled from virome reads. PLoS One 7(7): e40418.

Shannon, C.E. (1948). A Mathematical Theory of Communication. The Bell System Technical Journal 27: 379–423, 623–656.

Shannon, C.E. & Weaver, W. (1949). The Mathematical Theory of Communication. The University of Illinois Press, Urbana.

Shenkin, P.S., Erman, B. & Mastrandrea, L.D. (1991). Information-theoretical entropy as a measure of sequence variability. Proteins 11: 297–313.