Package 'LAMPA'

August 11, 2019

Title LArge Multidomain Protein Annotator

Version 1.0.0

Author Anastasia A. Gulyaeva, Andrey I. Sigorskih, Elena S. Ocheredko, Alexander E. Gorbalenya

Description Functional annotation of large multidomain proteins by iterative HH-suite-based procedure. The procedure utilizes fragments of a large multidomain protein sequence as queries to profile databases. It allows to avoid underestimation of statistical support that is often observed when standard tools are used to annotate large multidomain proteins.

License GPL (>=2)

Encoding UTF-8

LazyData true

SystemRequirements HH-suite, TMHMM

Imports seqinr, IRanges

RoxygenNote 6.1.1

R topics documented:

`LAMPA` *LArge Multidomain Protein Annotator*

Description

Conducts iterative HH-suite-based annotation procedure for a specified query protein sequence using specified target database(s).

Usage

```
LAMPA(seqF, DBs, hhmakepath, hhsearchpath, addsspath = NULL,
  tmhmmpath = NULL, out_path = ".", out_name = NULL, tm_gap = 100,
  qp_len = 1, ap_len = 300, P = 95, E = 10, L = 50, cpu = 1)
```

Arguments

`seqF`	character string. FASTA file containing query sequence.
`DBs`	named list: elements, paths to databases; names, names of databases.
`hhmakepath`	character string. HHmake executable. HHmake will be run with `-M first` argument.
`hhsearchpath`	character string. HHsearch executable.
`addsspath`	character string. Script addss.pl (from HH-suite software package) executable. If specified, secondary structure of the query sequence will be predicted and included into query profiles. Script will be run with `-fas` argument.
`tmhmmpath`	character string. TMHMM executable. If specified, transmembrane helices of the query sequence will be predicted and taken into account while delineating query sequence fragments to be analysed on second HH-suite-based iteration.
`out_path`	character string. Destination where output folder will be placed. Working directory by default.
`out_name`	character string. Name of the output folder. If NULL, `seqF` filename without path and extension. If a folder with this name already exists at the selected destination, it will be deleted.
`tm_gap`	integer value. Transmembrane helices separated by less than `tm_gap` amino acids will be clustered.
`qp_len`	integer value. Minimal length of query sequence fragments that will be analysed during the QP-specific iterations. Shorter query sequence fragments will be discarded.
`ap_len`	integer value. Length of query sequence fragments that will be analysed during the AP-specific iterations.
`P`	numeric value. Probability threshold: only hits characterised by Probability above `P` will be considered.
`E`	numeric value. E-value threshold: only hits characterised by E-value below `E` will be considered.
`L`	integer value. Length threshold: only hits longer than `L` (in residues of the query sequence) will be considered.
`cpu`	integer value. Number of CPUs to be used by HHsearch.

Details

Protein annotation procedure consists of the following stages:

0. *Transmembrane (TM) regions prediction.* TM helices are predicted by TMHMM. TM helices separated by a distance less than `tm_gap` are grouped into TM clusters.

1. *Application of HHsearch to the whole-length protein sequence.* This is the first iteration of the homology annotation procedure. On every iteration HHsearch hits are filtered: only hits satisfying `P`, `E` and `L` thresholds are retained for consideration. Retained hits are clustered if they overlap, and the hit characterised by the highest Probability value in its cluster is reported as domain annotation.

2. *Query-protein-specific (QP-specific) iterations.* Initially, protein is split into fragments by clusters of hits obtained on the first iteration, as well as clusters of TM helices, followed by application of HHsearch to the delineated fragments. On each subsequent iteration, protein is further split by the clusters of hits obtained on preceding iteration, and HHsearch is applied to the delineated fragments. The procedure is repeated until iteration when no hits, satisfying `P`, `E` and `L` thresholds, are identified. Only query sequence fragments whose length is above or equal to `qp_len` are considered.

3. *Average-protein-size-specific (AP-specific) iterations.* Query regions from previous iterations, for which no annotations were obtained (whole protein sequence is considered only if there were neither TM, nor homology annotations obtained for it), are split into fragments of equal length `ap_len`, starting from the N-terminus (first AP-specific iteration) and the N-terminus with `ap_len%/%2` inset (second AP-specific iteration). The most C-terminal fragments are extended to include the remaining part of the region under consideration, if the remaining part is shorter than `ap_len%/%2` and if the extended fragment does not cover the entire region under consideration. HHsearch is applied to the delineated fragments.

Output of the function is placed into directory `out_path/out_name/`. Output consists of the following:

- Plot summarising obtained annotation, `<query ID>_annotation_plot.pdf`:
  - black numbers, iterations
  - grey bars, fragments of query sequence
  - red lines, clusters of predicted TM helices
  - red numbers, indices of clusters of predicted TM helices
  - blue lines, clusters of hits
  - blue numbers, indices of clusters of hits
- Table summarising TM predictions, `<query ID>_TM.tsv`
  Each row of the table corresponds to a predicted TM helix, table has the following columns:
  - `tm_helix_from`, `tm_helix_to`, coordinates of a TM helix
  - `tm_cl_index`, index of the cluster to which TM helix belongs
- Table summarising homology annotations, `<query ID>_annotation_table.tsv`
  Each row of the table corresponds to a cluster of hits, table has the following columns:
  - `q_id`, query sequence ID
  - `q_from`, `q_to` and `q_len`, coordinates and length of the analysed query sequence fragment (in residues of the query sequence)
  - `iterat_num`, iteration number
  - `iterat_type`, iteration type (stage): 1 = 1st iteration, 2 = QP-specific iteration and 3 = AP-specific iteration
  - `cl_index`, index of a cluster of hits
  - `cl_from`, `cl_to` and `cl_len`, coordinates and length of the cluster (in residues of the query sequence)
  - `DB`, target database

  - `Hit`, ID of the target profile that yielded top-scoring hit of the cluster
  - `Prob`, `E_value` and `Score`, statistics characterising the top-scoring hit
  - `h_from`, `h_to` and `h_len`, coordinates and length of the top-scoring hit in residues of the query sequence
  - `TemplateHMM`, coordinates of the top-scoring hit in match states of the target profile (profile length is specified in parentheses)
- Files with information about hits constituting each cluster:
  table - `<query ID>_hits_cluster_<cluster ID>.tsv`,
  alignments - `<query ID>_hits_cluster_<cluster ID>_alignments.txt`
- Folder `utility_data/` that contains raw data generated by the procedure.

Examples

```
LAMPA(
          seqF = '/path1/query_pp.fasta',
          DBs  = list(pfam = '/path2/pfamA_28.0_hhm_db',
                      pdb  = '/path3/pdb70_06Sep14_hhm_db'),
          hhmakepath   = 'hhmake',
          hhsearchpath = 'hhsearch -p 0 -norealign -alt 10',
          addsspath    = 'addss.pl',
          tmhmmpath    = 'tmhmm'
)
```

Index