MAFFT2ASR-Analysis.py Manual

Introduction

MAFFT2ASR-Analysis.py is a script (pipeline) to continuously analyze amino acid sequence data saved in Fasta format using MAFFT, PhyML, and PAML, and to output ancestral sequences in a fully automated manner.

Required environment

Linux (CentOS 6 or 7)

Python 3.7 or higher, PhyML version 20120412, PAML (codeml), MAFFT, BioPython

Required input data

Protein sequence data in Fasta format (at least 3 arrays)

Note: Too many sequences would require much of computation time to predict the ancestral sequences.

Procedure

Setting up the execution environment

1. Download and install the MAFFT source data (https://mafft.cbrc.jp/alignment/software/source.html). Type "mafft" from the command line and confirm that no error appears.
2. Download and install the source data of PhyML (http://www.atgc-montpellier.fr/phyml/download.php). Type phyml on the command line and confirm that no error appears. After confirming, type “Ctrl+C” key to leave the interactive mode.
3. Download and install the PAML source data from http://abacus.gene.ucl.ac.uk/software/paml.html. After setting the path, type codeml from the command line, and then confirm that following message would be appeared in the terminal: ” tell me the full path-name of the file?”.
4. Open MAFFT2ASR-Analysis.py with a text editor and change the path of aaRatefile=/***/ in line 127 to match your installation environment.

Preparation of the input files

0. Make a directory for analysis. Move the sequence data which would be a template file to reconstruct ancestral sequence into it (mv command). Perform the following operations under the analysis directory.
1. Open the sequence data with a text editor. First, make sure there are no extra comments on the first line.
2. Make sure that there is a blank line between arrays. Make sure it is in the following format.

```
>BBE38586.1
MLAMLAKISLLSLASMAAATSYDYIVVGGGTSGLVIANRLSEDSSVSVLIVERGDSVLNNAL
VYNTSDYGAAFGSSIDYAYQSVPQAYAGNKVQTLRAGKALGGTSTINGMAYTRAEDIQIDA
WGQIGNNGWNWKNLFPYYEKSEDFQVPTPAQYDAGANYNPSYNGESGPLLVGWTYDM
QNSSIHTELNITYQNLGISYLPDVNGGKMHGYSMFPRTVNRAENVREDAARAYYYPFDSR
PNLSAMLNTTGNRILWAPQTSTSSAAVASGLEVTLSDGTVETITANKEVILSAGSLISPAILER
SGVGNPAVLAQHSIPLVVNLTTVGENLQDQVNTEFIYTSNVSYSGAGTYLGHPTASDIFGS
NTTNVANDVKNNLANYAAQVSAASNGTMSAANLLALFNIQYDIIFENPTPIAEVLVTPKGTN
YYSEYWGLLPFARGNVHIASTDPLQQPTINPNYMMLEWDMQQQIGSGKFLRTLYNTAPMS
AYTTGESTPGYTTLPADATDAQWASWINSVSRSNFHPVGTAAMMPRDMGGVVDTNLMVY
GTANVRVVDASVLPFQVCGHLTSTLYAVAERAADLIKAGDV

>PYH94523.1
MLRSLTLLGALSALASAATPEYDYVIIGGGTSGLVVANRLSENPDVSVLVIEAGDSVYDNYN
VTDVDGYGLAFGTDIDWQYETVLQPYAGNVTQVLRAGKALSGTSAINGMAYTRAEDVQID
AWQAIGNEGWTWDSLLPYYLKSENLTAPTTAQAEAGATFDAAVNGEDGPLAVGWPELPLS
NLTSTVNATFAALGVPWTADVNGGKMRGFNVFPSTIDYAEYVREDAARAYYFPFDTRANL
HVLLNTFANRIVWSDAATAGDHVTAAGVEITYANGTTSVVGAREEVIVSAGSLKSPAILELS
GVGNPAVLEPLNITVKVDLPTVGENLQDQTNAGAYANATSDLTGGKTVAYPNVYDVYGNET
SAVARSVRHQLRQWARETAEVSSGTMTASDLEALFQVQYDLIFTDKAPIAEILYYPGGGNE
LAVQFWGLLPFARGTVHIASADPTTFPTIDPNYWKFDWDIDSTIAIAKYIRKTLQTAPLKDLIA
VETSPGAAVATDAEESVWEDWLLTEYRSNFHPVGTAAMMPKAKGGVVSEQLTVYGTSNV
RVVDASVLPFQVCGHLTSTLYAVAERASDLIKAESSLF

>AER13599.1
MKNLIPLSLLATTVAARPGSAPRDQAAATAYDYIVIGGGTSGLVVANRLSEDASVSVLVI…
```

3. Type the following command to run the script.

```
sudo python MAFFT2ASR-Analysis.py -INPUTFILE <array data processed in step 2> -OUTPUTFILE <output file name>
```

Analysis of the output data

0. All of the output data is saved under the directory named “result-PAMLlog/”. The files in this directory should be analyzed necessarily.
1. Open summary.out in a text editor.
   Ancestral sequences which are near to the common ancestor in the library (input data)

are listed in the following order: >node#1, >node#3, >node#2. For example, if the sequences are >node#1, >node#3, >node#2, then >node#1 is closest to the common ancestor and >node#2 is closest to the current sequence.

#Sequence identity which bears the highest value when we compared between an ancestral sequence and a library sequence: The sequence identity which was most similar between an ancestral sequence and a library sequence. The following table shows the sequence identity which bears the highest value when we compared between an ancestral sequence and a library sequence. For example, if 91.4;node#2, then node#2 (the ancestral sequence) has 91.4% sequence identity with the most similar sequence in the sequence library.

#Sequence identity which bears the lowest value when we compared between an ancestral sequence and a library sequence. The following indicates the match between the ancestral sequence and the least similar sequence between the sequence library and the ancestral sequence. For example, 87.4; node For example, if 87.4;node#2, then node#2 has 87.4% homology with the most remote sequence in the sequence library.

2. Next, open nodex-ancestraldata.log (where x is the node number). Here you can see the results of the nodex analysis. The output format is as follows.

```
#The ancestral sequence name: node4
#The average posterior probability values was:      0.954
#Total number of residues was: 631
#Number of residues which bear >0.90 of PP value: 559
#Number of residues which bear >0.80 of PP value: 572
#Number of residues which bear >0.70 of PP value: 582
#Number of residues which bear >0.50 of PP value: 615
#The residue number, posterior probability
   1,     1.000
   2,     1.000
   3,     1.000
.
.
.
>node#x
*****
```

Line 1: Name of the ancestor sequence.
Line 2: Posterior probability (PP) value of the mean.
Line 3: Number of residues in the ancestral sequence
Lines 4, 5, 6, 7: Show the number of residues with PP values >0.90, >0.80, >0.70, >0.50 or higher
Line 8 onwards: PP values for each residue.
Last line: Array data for >node#x

For reference, we recommend sequences with an average PP value of 0.95 or higher (maybe higher probability of expression).

3. Open commonanc.fasta in a text editor. It contains the following information.

   >node#x;0.9543

   The result of the analysis shows that the common ancestor sequence is node#x, and the average PP value is 0.9543.
4. The best sequence should be selected and sent for artificial gene synthesis.

[Additional]

If you want to change the output method in the Ancestral Type Design section & analyze the data differently from the default values, please change the lines 119-143 in the script.

IUPAC is no longer supported in some versions of Biopython (Biopython 1.75 or later?). If you get an error with this, you should use MA. If you get an error with this, use MAFFT2ASR-noIUPAC.py.