Datamonkey Methods and Tools Job Queue Usage statistics API Citations Help Classic

Help

Documentation for Datamonkey's Analyses

For persistent errors or questions not answered here please open an issue on ourGithub

Example Files

INFLUENZA A H5N1 HEMAGLUTTININ HIV-1 POL (RECOMBINANT DATA)

Data Files

General Remarks

For help reading the .json results file generated by DataMonkey please refer to this .

To perform a selection analysis, DataMonkey needs a multiple alignment of at least three homologous coding nucleotide sequences. Codon based methods for estimating dN and dS can be applied to any sequence alignment, but there are several considerations to keep in mind:

- Ideally, the alignment should represent a single gene, or protein product, sampled over multiple taxa (e.g. mammalian interferon genes), or a diverse population sample (e.g. Influenza A viruses infecting different individuals). Because comparative methods estimate relative rates of synonymous and non-synonymous substitution, substantial sequence diversity is needed for reliable inference.
- For example when,Suzuki and Nei applied a REL-type method to a very low divergence (1 or 2 substitutions per sequence along a star phylogeny) sample of the Human T-lymphotropic virus (HTLV), they found that the method performed poorly.
- Yang and colleagues have suggested that the total length of the phylogenetic tree should be at least one expected substitution per codon site, but it is impossible to give a generally valid range for desirable sequence divergence.
- However, sequences that are too divergent could lead to saturation, i.e. our inability to reliably infer branch lengths and substitution parameters. The number of sequences in the alignment is important: too few sequences will contain too little information for meaningful inference, while too many may take too long to run.
- As a rule of thumb, at least 10 sequences are needed to detect selection at a single site (SLAC/FEL/REL) with any degree of reliability, while as few as 4 may be sufficient for alignment-wide inference (BUSTED). For information about typical datasets sizes gathered from other DataMonkey users see: Usage Statistics
- Comparative methods are ill suited to study certain kinds of selection. For example, they should not be applied to the detection of selective sweeps (rapid replacement of one allele with a more fit one, resulting in a homogeneous population), unless sequences sampled prior to and following the selective sweep are included in the sample. A number of publications have dealt with this issue extensively (e.g. Selection using HyPhy), and we refer an interested reader to one of these works for further insight.

It is a good practice to visually inspect your data to make sure that the sequences are alignment correctly. Of course, one can never be sure that an alignment is objectively correct, but gross misalignments (e.g. sequences that are out of frame) are easy to spot with software that provides a graphical visualization of the alignment. Datamonkey uses the HyPhy package as its processing engine, and if an alignment does not open in HyPhy on your machine (see HyPhy for info about running HyPhy), then it will not be properly read by Datamonkey.

- You should verify that the alignment is in frame, i.e. that it does not contain stop codons, including premature stop codons (indicative of a frame shift, e.g. due to misalignment, or a non-functional coding sequence) and the terminal stop codon.
- Your alignment should exclude any non-coding region of the nucleotide sequence, such as introns or promoter regions, for which existing models of codon substitution would not apply.
- When coding nucleotide sequences are aligned directly, frameshifting (i.e. not in multiples of 3) gaps may be inserted, since the alignment program often does not take the coding nature of the sequence into account. Therefore it is generally a good idea to use a codon-aware aligner like Codon-MSAor align translated protein sequences and then map them back onto constituent nucleotides.
- Datamonkey will perform a number of checks when it receives coding sequences and report all problems it encounters.

If the alignment contains identical sequences, Datamonkey will discard all but one copy before proceeding. This is done to speed up the analyses, because identical sequences do not contribute any information to the likelihood inference procedure (except via base frequencies), but the computational complexity of phylogenetic analyses grows with the number of sequences.

Finally, Datamonkey may rename some of the sequences to conform to HyPhy naming conventions for technical reasons (all sequence names must be valid identifiers, e.g. they cannot contain spaces). This is done automatically and has no effect on the subsequent analyses.

Datamonkey expects sequence alignments to be uploaded as text files ie, .fasta, .nex, .txt. Any other format (Word, RTF, PDF) will not be recognized and must be converted into plain text prior to submission.

Nonstandard characters in the alignment

For instance, BioEdit may use the tilde ('~') character to denote a gap. The dot ('.') character is sometimes used as "match the first sequence" character and sometimes as the gap character. Datamonkey will acceptIUPAC nucleotide characters (ACGT/U and ambiguity characters) and '?', 'X', 'N' or '-' for gap or missing data (Datamonkey is not case sensitive). All other characters in sequence data will be skipped and could result in frame shifts.

Uploading an amino-acid alignment

Current Datamonkey methods do **not** support amino scid alignments.

Termination codons

Datamonkey will reject any alignments that contains stop codons, even if the stop codon is at the end of the sequence (i.e. is a proper termination codon). Please strip all stop codons out of the alignment prior to uploading it. This can be done with the command line version of HyPhy using the CleanStopCodons.bf

Alignments that are too gappy

If an alignment contains more than 50% of indels, it may not be properly processed (e.g. it could be read as a protein alignment, depending on the alignment format).

Alignments that are too large

If your alignment exceeds the size currently allowed by Datamonkey, consider running your analysis locally in HyPhy. A detailed discussion of how HyPhy can be used for that purpose can be found in Tutorial.

Incorrect genetic code

If the genetic code is misspecified (e.g. the mitochondrial code is applied to nuclear sequences), valid alignments may fail to upload and if they do, then the results may be compromised (because codons are mistranslated). Make sure the correct genetic code is selected on the data upload page.

Genetic Codes

Universal Genetic Code

Amino acid	Codons
Phe	TTC,TTT
Leu	CTA,CTC,CTG,CTT,TTA,TTG
Ile	ATA,ATC,ATT
Met	ATG
Val	GTA,GTC,GTG,GTT
Ser	AGC,AGT,TCA,TCC,TCG,TCT
Pro	CCA,CCC,CCG,CCT
Thr	ACA,ACC,ACG,ACT
Ala	GCA,GCC,GCG,GCT
Tyr	TAC,TAT
His	CAC,CAT
Gln	CAA,CAG
Asn	AAC,AAT

Asp	GAC,GAT
Glu	GAA,GAG
Cys	TGC, TGT
Trp	TGG
Arg	AGA,AGG,CGA,CGC,CGG,CGT
Gly	GGA,GGC,GGG,GGT
Stop	TAA,TAG,TGA

Other genetic codes are defined in terms of differences with the Universal code.

Vertebrate mtDNA

Codon	New translation
AGA	Stop
AGG	Stop
ATA	Met
TGA	Trp

Yeast mtDNA

Codon	New translation
ATA	Met
CTA	Thr
CTC	Thr
CTG	Thr
CTT	Thr
TGA	Trp

Mold, Protozoan and Coelenterate mtDNA

Codon	New translation
TGA	Trp

Invertebrate mtDNA

Codon	New translation
AGA	Ser
AGG	Ser
ATA	Met

Ciliate Nuclear Code

Codon	New translation
TAA	Gln
TAG	Gln

Echinoderm mtDNA

Codon	New translation
AAA	Asn
AGA	Ser
AGG	Ser
TGA	Trp

Euplotid mtDNA

Codon	New translation
TGA	Cys

Alternative Yeast Nuclear

Codon	New translation
CTG	Ser

Ascidian mtDNA

Codon	New translation
AGA	Gly
AGG	Gly
AGG	Met
TGA	Trp

Flatworm mtDNA

Codon	New translation
AAA	Asn
AGA	Ser
AGG	Ser

Blepharisma Nuclear

Codon	New translation
TAG	Gln

Data Formats

Datamonkey automatically recognizes five aligned sequence data formats and also autodetects whether the data is nucleotide (codon) or aminoacid.

NEXUS

The following NEXUS blocks are supported: `DATA`, `CHARACTERS`,`TAXA`, `ASSUMPTIONS` (for data partitioning) and `TREES`.

PHYLIP

PHYLIP option characters in the first line are ignored for both sequential and interleaved formats.

FASTA

- **Sequential:** Taxa names are preceded by `>` (or `#`), and complete sequence data follow the name of the taxon.
- **Interleaved:** List of taxa names preceded by `>` (or `#_` , and blocks of sequence data follow in the same order as the names of the taxa.

For examples of each format, please visit the hyphy wiki page

Analyzing Data

Selecting a nucleotide model

Complete model selection procedure details can be found in this MBE paper

General Advice

We recommend that you run a model selection procedure, which sifts all 203 possible time-reversible models through a hierarchical testing procedure combining nested LRT tests with AIC selection to pick a single "best-fitting" rate matrix. Model selection is processed on a remote cluster, and should take no more than a few minutes to complete.

To allow the most general model of nucleotide substituion, select the General Reversible Model (REV), since it does not add much to the overall processing time. However, if your data set is small, it may not be possible to accurately estimate nucleotide substitution bias rates, and HKY85 might not be a bad choice. You can also try several different models and see if the location of inferred sites changes depending on the nucleotide model (it rarely does, unless the model is very wrong).

Handling Ambiguities

For more details see MBE paper

Averaged (default)

All possible resolutions of an ambiguous character contribute, in a weighted fashion, to the computation of EN, ES, NN and NS (see methods paper). Characters without any information (all gaps or all missing) are NOT counted though, to avoid artifically high dN and dS estimates.

Resolved

The most likely resolution *for the given site* is used in the computation of EN, ES, NN and NS. Ties are broken randomly.

Skip

GapMM

```
ACA ACG ACG ACR
```

For the resolved option, only most frequent resolution *based on the data in the site only*, will be considered. In this case, the resolution is 'ACG'

For the averaged option, all four possible resolutions ('ACA' and 'ACG') will be considered. The weight factor for each resolved is determined by the relative frequency of that codon to all possible resolutions. If f(xyz) denotes the frequency of codon xyz in the entire data file, then the contribution of ACA will be f(ACA)/(f(ACA)+f(ACG)) and of 'ACG' : f(ACG)/(f(ACA)+f(ACG)).

Choosing significance levels

For more details see MBE paper

Datamonkey infrastructure and development has received support from the NIH (R01GM151683, U01GM110749, U24AI183870, R01GM093939), and the NSF (2027196, 2419522).