**apex: phylogenetics with multiple genes**

Thibaut Jombart1, Klaus Schliepaaa, Zhian Kamvarbbb, Eric Archerxyz, Rebecca Harrisccc, Emmanuel Paradisddd, Jérome Goudeteee, Hilmar Lappfff

1MRC Centre for Outbreak Analysis and Modelling, Department of Infectious Disease Epidemiology, School of Public Health, Imperial College London, United Kingdom.

xxx

**Corresponding author:**

Thibaut Jombart

MRC Centre for Outbreak Analysis and Modelling  
Department of Infectious Disease Epidemiology  
Imperial College - School of Public Health  
St Mary’s Campus  
Norfolk Place  
London W2 1PG  
United Kingdom  
Tel. : 0044 (0)20 7594 3658  
t.jombart@imperial.ac.uk

**ABSTRACT**

**Keywords:** R, package, software, genetics, phylogenies

**INTRODUCTION**

The constant improvement of sequencing technologies provides ever-increasing amounts of genetic sequences for a wide range of organisms including viruses, bacteria and a variety of eukaryotes. As a consequence, organisms are more frequently sequenced for multiple genes, and full-genome data are becoming increasingly common [(Pettersson *et al.* 2009; Pareek *et al.* 2011)](https://paperpile.com/c/b4EFig/geMM+DLo7). This is especially true for viruses and bacteria, whose smaller genomes allow for large collections of complete genome sequences to be assembled relatively easily [(e.g. Harris *et al.* 2013; Weinert *et al.* 2015)](https://paperpile.com/c/b4EFig/vLzV+NPQg/?prefix=e.g.,).

While it is tempting to treat multiple genes as a single evolutionary unit [(de Queiroz & Gatesy 2007)](https://paperpile.com/c/b4EFig/j9fe), a number of biological processes can lead different genes to undergo distinct evolution and exhibit effectively different phylogenies [(Rokas *et al.* 2003; Beiko *et al.* 2005; McBreen & Lockhart 2006; Pollard *et al.* 2006; Nelson *et al.* 2008)](https://paperpile.com/c/b4EFig/n8J9+o2gx+6zza+TKYV+f2Rq). Therefore, multiple gene data should ideally be first analysed separately to look for congruent genealogies, and then possibly concatenated to derive a single phylogenetic tree. In practice, however, such an approach may not be trivial to implement.

Indeed, handling and analysing multiple gene alignments can be a daunting task. For instance, while the R software is growing as a platform of choice for phylogenetic analyses [(Paradis *et al.* 2004; Kembel *et al.* 2010; Jombart *et al.* 2010a; Schliep 2011; Popescu *et al.* 2012)](https://paperpile.com/c/b4EFig/KMHQ+ynAF+BerG+MLi0+jLoz), tools for handling and analysing multiple gene data are missing. In this paper, we introduce *apex*, a new R package which fills this gap. *apex* implements new object classes for storing and handling multiple gene data. *apex* is fully integrated with existing R standards for phylogenetic reconstruction, and makes the analysis of multiple genes as straightforward as for single genes. In the following, we provide an overview of these functionalities and illustrate the main features using a worked example.

**FUNCTIONALITIES**

**New object classes**

The main feature of *apex* is to provide classes for storing and handling multiple gene data. This is implemented through two novel formal (S4) classes, multidna and multiphyDat, which are respectively extensions of the DNAbin objects from the package *ape* [*(ape, Paradis et al. 2004; Popescu et al. 2012)*](https://paperpile.com/c/b4EFig/KMHQ+ynAF/?prefix=ape%2C%20,), optimized for distance-based methods, and phyDat objects from the package *phangorn* [(Schliep 2011)](https://paperpile.com/c/b4EFig/BerG), better suited for parsimony and likelihood-based approaches.

As in any formal class, the content of multidna and multiphyDat is pre-defined by a number of ‘slots’ containing specific information, such as the number of individuals in the dataset, the various gene sequences, and various meta-information. For simplicity, the two new classes have identical slots (Table 1), and only differ in the way DNA sequences are stored internally: multidna uses bytes to code nucleotides (DNAbin objects), while multiphyDat enumerates variable patterns in the sequences (phyDat objects). In both cases, aligned DNA sequences are stored inside a list (slot @dna, Table 1), in which each element corresponds to a specific gene/locus. Besides storing genetic sequences, multidna and multiphyDat can also store labels for the individuals sequenced, as well as any meta-data regarding the individuals or the genes of the dataset (Table 1).

Both classes ensure that data is stored in a consistent way. When creating new multidna or multiphyDat objects, all individuals with at least one sequence are first enumerated and sorted alphanumerically. For each gene, gap-only sequences are created for each missing individual, and new alignments containing all individuals, sorted identically, are created. As a result, the different gene alignments and their analyses can be readily compared, which greatly facilitates the assessment of congruence amongst the loci.

**Handling data**

The fact that different gene data are stored in a consistent way also makes data manipulation easier. In both classes, we implemented matrix-like subsetting using the syntax ‘[’ operator. Assuming ‘x’ is a multidna or multiphyDat object, then ‘x[i , j]’ is a subset of ‘x’ where ‘i’ indicates the individuals to be kept, and ‘j’ the retained genes. This subsetting follows R standards and allows for vector of integers, characters or logical to be used, so that handling of multidna or multiphyDat objects should be as natural as usual objects (vectors and matrices).

In addition to the easy subsetting and reordering or individuals and genes, the generic function ‘concatenate’ can be used to merge several genes into a single alignment. By default, all genes in the object are used, but an optional argument permits to select which genes to include in the alignment.

**Importing and exporting data**

Building on resources provided in *ape* [(Paradis *et al.* 2004; Popescu *et al.* 2012)](https://paperpile.com/c/b4EFig/KMHQ+ynAF), *phangorn* [(Schliep 2011)](https://paperpile.com/c/b4EFig/BerG) and *adegenet* [(Jombart 2008; Jombart & Ahmed 2011)](https://paperpile.com/c/b4EFig/jftb+qcRk), *apex* provides functions to import data from and export them to a variety of formats (Table 2).

multidna and multiphyDat objects can both be created in R using the constructor ‘new(a, ...)’, where ‘a’ is a character string indicating the class of the object (“multidna” or “multiphyDat”), and ‘...’ is a list of arguments passed to the constructor, the main one being a list of objects (character, DNAbin or phyDat matrices) storing DNA sequences. However, it is likely that in most cases, DNA sequences of different genes will be stored in separate text files, using one file per gene. Three functions permit to import data from a list of files directly into *apex*. The functions ‘read.multidna’ and ‘read.multiFASTA’ build upon *ape*’s procedures to read data in interleaved, sequential, clustal or fasta formats. In addition, the function ‘read.multiphyDat’ enables imports of amino-acid sequences with interleaved, sequential or fasta format. Note that in this case, the resulting multiphyDat object can no longer be converted to multidna, which is restricted to nucleotide sequences only.

Once imported in *apex*, data can also be converted in various formats (Table 2). Conversion from multidna to multiphyDat is implemented by multidna2multiphyDat, while the reverse operation is performed by multiphyDat2multidna. Single Nucleotide Polymorphism (SNPs) can be extracted from the alignments and translated into a genind object, thereby providing access to a wide range of multivariate analyses [(Jombart *et al.* 2008, 2009, 2010b)](https://paperpile.com/c/b4EFig/goGU+jLCH+XQrV). As an alternative, each unique gene sequence can be treated as a separate allele, generalising the Multi Locus Sequence Type (MLST) approach which proved highly useful for classifying clonal organisms in microbiology [(Maiden *et al.* 1998; Aanensen & Spratt 2005; Maiden 2006)](https://paperpile.com/c/b4EFig/Kyur+MboD+85AZ), but could also be quite useful for large gene collections of non-clonal organisms.

**Analysing data**

As each gene is an item of a list (the slot @dna), any operation done on one gene can be applied to all genes using a simple lapply.

**Accessors**

The number of loci in a multidna object and their names can be obtained with the getNumLoci and locusNames functions respectively. The former returns the number of DNAbin objects in the @dna slot, while the latter returns a character vector of their names. The locusNames function can also be used to set the names in the same way that the names, colnames, and rownames functions work on standard R objects.

One can obtain the number of sequences at each locus with the getNumSequences function. With default arguments, it will provide the number of sequences at all loci in the @dna slot. Counts for specific loci can be produced by providing a vector of their names to the loci argument. By default, only sequences that not composed entirely of gaps are counted, however if the exclude.gap.only argument is FALSE, the value returned is the number of all sequences.

Similarly, names of sequences at each locus can be obtained with the getSequenceNames function. Sequence names for specific loci can be obtained by supplying those locus names to the loci argument. This function also has an exclude.gap.only argument that operates the same as in getNumSequences.

The sequences themselves can be returned with the getSequences function. By default, this returns the list of DNAbin-formatted sequences stored in the @dna slot. The return can be filtered for specific individuals and loci by providing a character to vector to the ids and loci arguments respectively. If only a single locus is returned and the simplify argument is TRUE (the default), then the return value is a single DNAbin object. If simplify is FALSE, the function will always return a list of DNAbin objects.

**WORKED** **EXAMPLE**

**DISCUSSION**

**ACKNOWLEDGEMENTS**

*apex* has been created during the “R Popgen Hackathon ”, an event organised by Hilmar Lapp and funded by NESCent. TJ is funded by the Medical Research Council Centre for Outbreak Analysis and Modelling and the National Institute for Health Research - Health Protection Research Unit for Modelling Methodology. We are thankful to github ([http://github.com/](https://github.com/)) and travis (<http://travis-ci.org/>) for providing great resources for software development.

**REFERENCES**

[Aanensen DM, Spratt BG (2005) The multilocus sequence typing network: mlst.net. *Nucleic acids research*, **33**, W728–33.](http://paperpile.com/b/b4EFig/85AZ)

[Beiko RG, Harlow TJ, Ragan MA (2005) Highways of gene sharing in prokaryotes. *Proceedings of the National Academy of Sciences of the United States of America*, **102**, 14332–14337.](http://paperpile.com/b/b4EFig/TKYV)

[Harris SR, Cartwright EJ, Török ME *et al.* (2013) Whole-genome sequencing for analysis of an outbreak of meticillin-resistant Staphylococcus aureus: a descriptive study. *The Lancet infectious diseases*, **13**, 130–136.](http://paperpile.com/b/b4EFig/vLzV)

[Jombart T (2008) adegenet: a R package for the multivariate analysis of genetic markers. *Bioinformatics* , **24**, 1403–1405.](http://paperpile.com/b/b4EFig/jftb)

[Jombart T, Ahmed I (2011) adegenet 1.3-1: new tools for the analysis of genome-wide SNP data. *Bioinformatics* , **27**, 3070–3071.](http://paperpile.com/b/b4EFig/qcRk)

[Jombart T, Balloux F, Dray S (2010a) adephylo: new tools for investigating the phylogenetic signal in biological traits. *Bioinformatics* , **26**, 1907–1909.](http://paperpile.com/b/b4EFig/jLoz)

[Jombart T, Devillard S, Balloux F (2010b) Discriminant analysis of principal components: a new method for the analysis of genetically structured populations. *BMC genetics*, **11**, 94.](http://paperpile.com/b/b4EFig/jLCH)

[Jombart T, Devillard S, Dufour A-B, Pontier D (2008) Revealing cryptic spatial patterns in genetic variability by a new multivariate method. *Heredity*, **101**, 92–103.](http://paperpile.com/b/b4EFig/goGU)

[Jombart T, Pontier D, Dufour A-B (2009) Genetic markers in the playground of multivariate analysis. *Heredity*, **102**, 330–341.](http://paperpile.com/b/b4EFig/XQrV)

[Kembel SW, Cowan PD, Helmus MR *et al.* (2010) Picante: R tools for integrating phylogenies and ecology. *Bioinformatics* , **26**, 1463–1464.](http://paperpile.com/b/b4EFig/MLi0)

[Maiden MCJ (2006) Multilocus sequence typing of bacteria. *Annual review of microbiology*, **60**, 561–588.](http://paperpile.com/b/b4EFig/MboD)

[Maiden MC, Bygraves JA, Feil E *et al.* (1998) Multilocus sequence typing: a portable approach to the identification of clones within populations of pathogenic microorganisms. *Proceedings of the National Academy of Sciences of the United States of America*, **95**, 3140–3145.](http://paperpile.com/b/b4EFig/Kyur)

[McBreen K, Lockhart PJ (2006) Reconstructing reticulate evolutionary histories of plants. *Trends in plant science*, **11**, 398–404.](http://paperpile.com/b/b4EFig/6zza)

[Nelson MI, Viboud C, Simonsen L *et al.* (2008) Multiple reassortment events in the evolutionary history of H1N1 influenza A virus since 1918. *PLoS pathogens*, **4**, e1000012.](http://paperpile.com/b/b4EFig/f2Rq)

[Paradis E, Claude J, Strimmer K (2004) APE: Analyses of Phylogenetics and Evolution in R language. *Bioinformatics* , **20**, 289–290.](http://paperpile.com/b/b4EFig/KMHQ)

[Pareek CS, Smoczynski R, Tretyn A (2011) Sequencing technologies and genome sequencing. *Journal of applied genetics*, **52**, 413–435.](http://paperpile.com/b/b4EFig/DLo7)

[Pettersson E, Lundeberg J, Ahmadian A (2009) Generations of sequencing technologies. *Genomics*, **93**, 105–111.](http://paperpile.com/b/b4EFig/geMM)

[Pollard DA, Iyer VN, Moses AM, Eisen MB (2006) Widespread discordance of gene trees with species tree in Drosophila: evidence for incomplete lineage sorting. *PLoS genetics*, **2**, e173.](http://paperpile.com/b/b4EFig/o2gx)

[Popescu A-A, Huber KT, Paradis E (2012) ape 3.0: New tools for distance-based phylogenetics and evolutionary analysis in R. *Bioinformatics* , **28**, 1536–1537.](http://paperpile.com/b/b4EFig/ynAF)

[De Queiroz A, Gatesy J (2007) The supermatrix approach to systematics. *Trends in ecology & evolution*, **22**, 34–41.](http://paperpile.com/b/b4EFig/j9fe)

[Rokas A, Williams BL, King N, Carroll SB (2003) Genome-scale approaches to resolving incongruence in molecular phylogenies. *Nature*, **425**, 798–804.](http://paperpile.com/b/b4EFig/n8J9)

[Schliep KP (2011) phangorn: phylogenetic analysis in R. *Bioinformatics* , **27**, 592–593.](http://paperpile.com/b/b4EFig/BerG)

[Weinert LA, Chaudhuri RR, Wang J *et al.* (2015) Genomic signatures of human and animal disease in the zoonotic pathogen Streptococcus suis. *Nature communications*, **6**, 6740.](http://paperpile.com/b/b4EFig/NPQg)

**AVAILABILITY**

The stable version of apex is released on the Comprehensive R Archive Network (CRAN):

<http://cran.r-project.org/web/packages/apex/index.html>

and can be installed in R by typing:

install.packages(“apex”)

The development version of apex is hosted on github:

<https://github.com/thibautjombart/apex>

*apex* is distributed under GNU Private Licence (GPL) version 2 or greater. It is fully documented in a vignette accessible by typing:

vignette("apex")

**AUTHOR CONTRIBUTIONS**

TJ, KS, ZK and EA developed the package. RH provided datasets. JG, EP and HL provided advices for the package design. TJ, …. wrote the manuscript.

**TABLES**

**Table 1: content of multidna and multiphyDat objects.** The content of each slot can be accessed using ‘@[*slot name*]’, where ‘[*slot name*]’ is any of the values listed in the first column.

|  |  |  |
| --- | --- | --- |
| **Slot name** | **Data stored** | **Description** |
| dna† | list of DNAbin matrices | a list containing sequences of one locus/gene each; for multidna, sequences are stored as DNAbin matrices; for multiphyDat, sequences are stored as phyDat objects; names are optional, and if provided identify the genes; all matrices have the same individuals in rows, and nucleotide positions in columns |
| labels | character vector | a vector of labels for the individuals |
| n.ind | integer | the number of individuals in the dataset |
| n.seq | integer | the total number of sequences, pooling all genes, and including gap-only sequences |
| n.seq.miss | integer | the total number of gap-only sequences |
| ind.info† | data.frame | a data.frame containing information on the individuals, where individuals are in rows |
| gene.info† | data.frame | a data.frame containing information on the genes, where genes are in rows |

† Slots whose content is NULL when empty.

**Table 2: functions for importing and exporting data in *apex*.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Function** | **Input** | **Output** | **Notes** |
| read.multidna | interleaved,  sequential,  clustal,  fasta files | multidna | based on read.dna (*ape* package) |
| read.multiFASTA | fasta files | multidna | based on read.FASTA (*ape* package) |
| read.multiphyDat | interleaved,  fasta | multiphyDat | based on read.phyDat (*phangorn* package)  can read amino acid sequences |
| multidna2multiphyDat | multidna | multiphyDat |  |
| multiphyDat2multidna | multiphyDat | multidna |  |
| multidna2genind | multidna | genind† | extract either SNPs or MLST |
| multiphyDat2genind | multiphyDat | genind† | extract either SNPs or MLST |

† Base class for genetic markers in the *adegenet* package.

**FIGURES**