



SOFTWARE TOOL ARTICLE

RhierBAPS: An R implementation of the population clustering algorithm hierBAPS [version 1; referees: 2 approved]

Gerry Tonkin-Hill ¹, John A. Lees ², Stephen D. Bentley¹, Simon D.W. Frost^{3,4},
Jukka Corander ^{1,5,6}

¹Parasites and Microbes, Wellcome Trust Sanger Institute, Hinxton, Cambridgeshire, CB10 1SA, UK

²Department of Microbiology, New York University School of Medicine, New York, NY, 10016, USA

³The Alan Turing Institute, London, NW1 2DB, UK

⁴Department of Veterinary Medicine, University of Cambridge, Cambridge, Cambridgeshire, CB3 0ES, UK

⁵Department of Biostatistics, University of Oslo, Blindern, 0317, Norway

⁶Department of Mathematics and Statistics, University of Helsinki, Helsinki, 00014, Finland

v1 First published: 30 Jul 2018, 3:93 (doi: [10.12688/wellcomeopenres.14694.1](https://doi.org/10.12688/wellcomeopenres.14694.1))
Latest published: 30 Jul 2018, 3:93 (doi: [10.12688/wellcomeopenres.14694.1](https://doi.org/10.12688/wellcomeopenres.14694.1))

Abstract

Identifying structure in collections of sequence data sets remains a common problem in genomics. hierBAPS, a popular algorithm for identifying population structure in haploid genomes, has previously only been available as a MATLAB binary. We provide an R implementation which is both easier to install and use, automating the entire pipeline. Additionally, we allow for the use of multiple processors, improve on the default settings of the algorithm, and provide an interface with the ggtree library to enable informative illustration of the clustering results. Our aim is that this package aids in the understanding and dissemination of the method, as well as enhancing the reproducibility of population structure analyses.

Keywords

clustering, population structure, R

Open Peer Review

Referee Status:

| | Invited Referees | |
|--|------------------|------------|
| | 1 | 2 |
| version 1 published 30 Jul 2018 | report | report |

1 **Emmanuel Paradis** , University of Montpellier, CNRS, EPHE, IRD, France

2 **Sebastian Duchene** , University of Melbourne, Australia

Discuss this article

Comments (0)

Corresponding author: Gerry Tonkin-Hill (ggt20@cam.ac.uk)

Author roles: **Tonkin-Hill G:** Software, Writing – Original Draft Preparation, Writing – Review & Editing; **Lees JA:** Validation, Writing – Review & Editing; **Bentley SD:** Supervision, Writing – Review & Editing; **Frost SDW:** Supervision, Writing – Review & Editing; **Corander J:** Supervision, Writing – Review & Editing

Competing interests: No competing interests were disclosed.

Grant information: This work was supported by the Wellcome Trust [206194] and [204016; to GTH; a Wellcome Trust PhD scholarship grant]; and SDWF is supported in part by The Alan Turing Institute via an EPSRC grant EP/510129/1.

The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Copyright: © 2018 Tonkin-Hill G *et al.* This is an open access article distributed under the terms of the [Creative Commons Attribution Licence](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

How to cite this article: Tonkin-Hill G, Lees JA, Bentley SD *et al.* **RhierBAPS: An R implementation of the population clustering algorithm hierBAPS [version 1; referees: 2 approved]** Wellcome Open Research 2018, 3:93 (doi: [10.12688/wellcomeopenres.14694.1](https://doi.org/10.12688/wellcomeopenres.14694.1))

First published: 30 Jul 2018, 3:93 (doi: [10.12688/wellcomeopenres.14694.1](https://doi.org/10.12688/wellcomeopenres.14694.1))

Introduction

Identifying sub-populations in collections of genetic sequences is a common problem in population genetics, molecular ecology, epidemiology and microbiology. In general, the aim of genetic clustering algorithms is to identify separate panmictic clusters within a broader, more heterogeneous population. In large sequence data sets, it is helpful to identify smaller subpopulations which can be further analysed for associations with particular phenotypes as well as recombination^{1,2}, as long as potential biases introduced through taking clusters from a larger population are taken into account³.

A frequently used model assumes that each individual sequence is drawn from one of K distinct subpopulations with each cluster having its own set of allele frequencies. The aim is then to identify which cluster each sequence originates from and the corresponding allele frequencies within that cluster.

There are a number of methods for solving this problem including STRUCTURE^{4,5}, snapclust⁶ and BAPS (Bayesian Analysis of Population Structure⁷⁻¹⁰). The BAPS algorithm^{9,11} is distinct in that it attempts to estimate the partition of individual sequences into clusters directly by analytically integrating over the allele frequencies parameters for each subpopulation. This allows for the latent number of underlying subpopulations, K , to be estimated as part of the model fitting procedure. The hierBAPS algorithm extends this approach by enabling the investigation of a population at multiple resolutions. This is achieved by initially clustering the entire dataset using the BAPS algorithm before iteratively applying the algorithm to each of the resulting clusters.

Similar to other approaches⁴, BAPS assumes that alleles are drawn independently from a multinomial distribution with a Dirichlet prior. However, unlike STRUCTURE, which uses Gibbs sampling to estimate the posterior distribution, BAPS attempts to find the partition of the data S that maximises the posterior probability of an allocation over all other possible allocations. A partition S is defined as the allocation of each sequence to one of K possible clusters. The maximum possible value of K is given in the hierBAPS algorithm. Here \mathcal{S} indicates the set of all possible partitions with up to K_{\max} clusters. The hierBAPS algorithm attempts to choose S to maximise

$$P(S|\text{data}) = \frac{P(\text{data}|S)P(S)}{\sum_{S \in \mathcal{S}} P(\text{data}|S)P(S)}$$

where $P(\text{data}|S)$ is the marginal likelihood of having the allele frequency parameters analytically integrated out leading to

$$P(\text{data}|S) = \prod_{i=1}^K \prod_{j=1}^{N_j} \left(\frac{\Gamma(\sum_l \alpha_{i,jl})}{\Gamma(\sum_l \alpha_{i,jl} + n_{i,jl})} \prod_{l=1}^{N_{A(j)}} \frac{\Gamma(\alpha_{i,jl} + n_{i,jl})}{\Gamma(\alpha_{i,jl})} \right)$$

where $n_{i,jl}$ is the count for allele l at locus j in cluster i and $\alpha_{i,jl}$ is the corresponding hyper-parameter for the Dirichlet prior. The BAPS algorithm attempts to find the partition S that maximizes the posterior probability using a greedy stochastic

search approach. A discretised uniform distribution of the cluster size K ($K = 1, \dots, K_{\max}$) is used in hierBAPS to provide the prior probability of each partition $P(S)$. The Dirichlet hyperparameters are set at $\frac{1}{N_{A(j)}}$ where $N_{A(j)}$ is the number of distinct alleles at locus j .

Currently, hierBAPS is only available as a MATLAB binary, which can be both difficult to install and use as separate runtime libraries are generally needed for different OS versions for MacOS X, Windows and Linux systems. The documentation is also lacking, making it difficult for less computationally experienced researchers to use. There is currently no clear guide on how to use the output of the MATLAB binary to produce informative plots for interpretation. Whilst there are other algorithms available to cluster genetic data in R, such as snapclust⁶ and DAPC¹², neither make use of the partition approach used in BAPS. By providing an R implementation of hierBAPS, we aim to increase its usability and the reproducibility of analyses using the software.

Methods

Implementation

RhierBAPS is implemented in the R language¹³. The core program relies upon the R packages ape¹⁴, dplyr, gmp, purrr and ggplot2. Additional plotting functionality makes use of ggtree¹⁵ and phytools¹⁶. The structure of the code is very similar to the original MATLAB code and has similar runtimes. The development version of the package can be installed using devtools.

```
install.packages("devtools")
devtools::install_github("gtonkinhill/rhierbaps")
```

Unlike the MATLAB version, rhierBAPS by default only considers SNP loci that have a minor allele in at least two sequences. This has been found to improve the results of the analysis as although singleton SNPs are important when constructing phylogenies they introduce noise into the model used in hierBAPS leading to poorer quality clusterings. It is currently recommended that singletons SNPs are removed before running the MATLAB version of the software.

Operation

RhierBAPS can be installed on any computer where R versions 3.5 and above can be installed. The package can be run using just a few lines of R code where the variable "fasta.file.name" should be replaced with the location of the FASTA formatted multiple sequence alignment of the sequences of interest.

```
library(rhierbaps)

fasta.file.name <- system.file("extdata",
"seqs.fa", package = "rhierbaps")
snp.matrix <- load_fasta(fasta.file.name)
hb.results <- hierBAPS(snp.matrix, max.depth = 2,
n.pops = 20, quiet = TRUE)
```

Use cases

RhierBAPS requires a multiple sequence alignment in FASTA format. In all examples we make use of sequences from the *Bacillus cereus* Multi Locus Sequence Typing website (<https://pubmlst.org/bcereus/>)¹⁷. The sequences used are included as part of the R package.

The algorithm requires an initial number of clusters to be set which should be higher than the maximum number of expected clusters in the dataset. If a dataset is likely to contain many distinct lineages, for example, if there are many samples from many locations, then a higher initial number of clusters should be set. Conversely, if the samples are from only a small number of sites and little variation is expected then a smaller initial cluster size can be set. To get an idea of a good initial cluster size, agglomerative clustering with complete linkage using pairwise SNP distances can be performed initially. The number of levels over which clustering should be performed is also required as input to the algorithm.

In the preceding example, we ran rhierBAPS with 20 initial clusters at two clustering levels. Additional parameters that can be set include the number of cores to use and whether the program should generate progress information. The hierBAPS function generates a data frame indicating the assignment of sequences to clusters at each level. This, along with the marginal log likelihoods can be saved to file.

```
write.csv(hb.results$partition.df, file =
  "hierbaps_partition.csv", col.names = TRUE,
  row.names = FALSE)
```

```
save_lml_logs(hb.results, "hierbaps_logML.txt")
```

Finally, as the program is written in R we are able to take advantage of the excellent plotting capabilities available. Given a phylogenetic tree generated using IQTREE¹⁸ with model selection¹⁹ using the command `iqtree -s`, we can annotate it with the BAPS clusters using `ggtree`¹⁵ (Figure 1).

```
newick.file.name <- system.file("extdata",
  "seqs.fa.treefile", package = "rhierbaps")
iqtree <- phytools::read.newick(newick.file.name)

gg <- ggtree(iqtree, layout="circular")
gg <- gg%<+%hb.results$partition.df
gg <- gg+geom_tippoint(aes(color=factor('level 1')))
```

Additionally, the `plot_sub_cluster` function allows for the user to focus on one higher level cluster and investigate the sub cluster present within it. Here we investigate cluster 9 (highlighted in red), at the top most level (Figure 2).

```
plot_sub_cluster(hb.results, iqtree, level = 1,
  sub.cluster = 9)
```

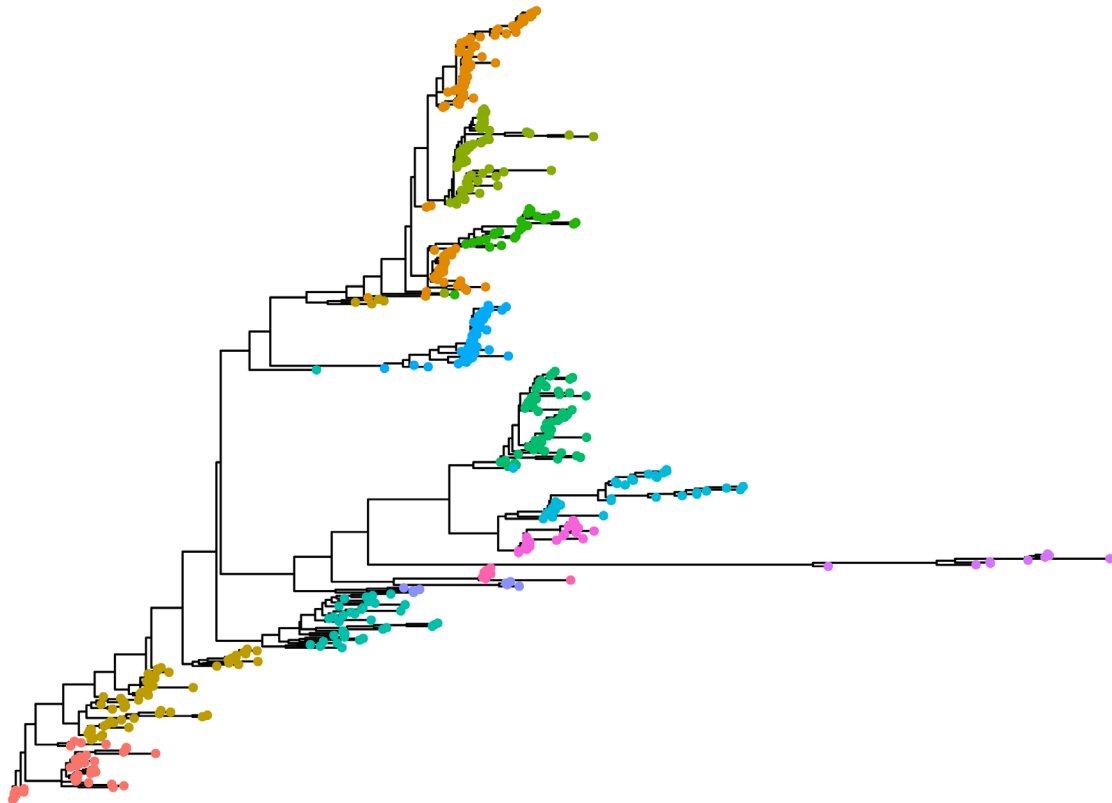


Figure 1. Phylogenetic tree built using `iqtree` and annotated with the top level clusters identified using `rhierBAPS`.

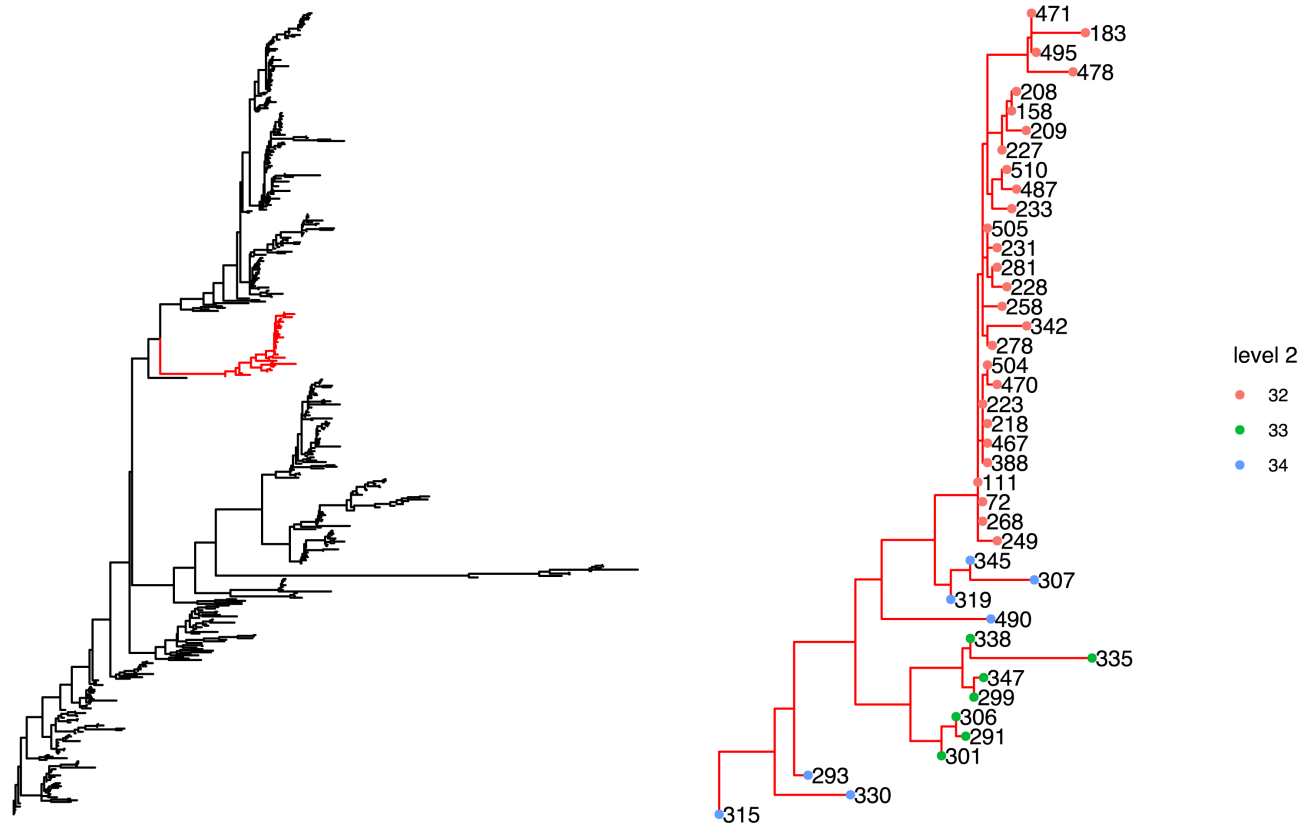


Figure 2. Phylogenetic tree focusing on the 9th cluster at the top level identified using rhierBAPS and plotted using the plot_sub_cluster function. The subsequent clustering at the 2nd level is indicated in the sub-tree to the right.

Summary

Clustering is an essential component of many genetic analysis pipelines. We have presented rhierBAPS, an R package that implements the hierBAPS algorithm for clustering genetic sequence data. It is both easy to install and use, whilst providing additional plotting capabilities and the ability to run using multiple cores. We believe it will aid in the reproducibility of population structure analysis.

Software availability

The package is available on CRAN: <https://cran.r-project.org/web/packages/rhierbaps/index.html>

Source code available from: <https://github.com/gtonkinhill/rhierbaps>

Archived source code as at time of publication: <http://doi.org/10.5281/zenodo.1318958>²⁰

License: MIT

Competing interests

No competing interests were disclosed.

Grant information

This work was supported by the Wellcome Trust [206194] and [204016; to GTH; a Wellcome Trust PhD scholarship grant]; and SDWF is supported in part by The Alan Turing Institute via an Engineering and Physical Sciences Research Council grant [EP/510129/1].

The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Acknowledgments

The authors thank a number of users who highlighted small bugs in the initial version of the software.

References

1. Chewapreecha C, Harris SR, Croucher NJ, *et al.*: **Dense genomic sampling identifies highways of pneumococcal recombination.** *Nat Genet.* 2014; **46**(3): 305–309.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
2. Marttinen P, Hanage WP, Croucher NJ, *et al.*: **Detection of recombination events in bacterial genomes from large population samples.** *Nucleic Acids Res.* 2012; **40**(1): e6.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
3. Dearlove BL, Xiang F, Frost SDW: **Biased phylodynamic inferences from analysing clusters of viral sequences.** *Virus Evol.* 2017; **3**(2): vex020.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
4. Pritchard JK, Stephens M, Donnelly P: **Inference of population structure using multilocus genotype data.** *Genetics.* 2000; **155**(2): 945–959.
[PubMed Abstract](#) | [Free Full Text](#)
5. Anderson EC, Thompson EA: **A model-based method for identifying species hybrids using multilocus genetic data.** *Genetics.* 2002; **160**(3): 1217–1229.
[PubMed Abstract](#) | [Free Full Text](#)
6. Beugin MP, Gayet T, Pontier D, *et al.*: **A fast likelihood solution to the genetic clustering problem.** *Methods Ecol Evol.* 2018; **9**(4): 1006–1016.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
7. Corander J, Waldmann P, Sillanpää MJ: **Bayesian analysis of genetic differentiation between populations.** *Genetics.* 2003; **163**(1): 367–374.
[PubMed Abstract](#) | [Free Full Text](#)
8. Corander J, Waldmann P, Marttinen P, *et al.*: **BAPS 2: enhanced possibilities for the analysis of genetic population structure.** *Bioinformatics.* 2004; **20**(15): 2363–2369.
[PubMed Abstract](#) | [Publisher Full Text](#)
9. Corander J, Marttinen P: **Bayesian identification of admixture events using multilocus molecular markers.** *Mol Ecol.* 2006; **15**(10): 2833–2843.
[PubMed Abstract](#) | [Publisher Full Text](#)
10. Cheng L, Connor TR, Sirén J, *et al.*: **Hierarchical and spatially explicit clustering of DNA sequences with BAPS software.** *Mol Biol Evol.* 2013; **30**(5): 1224–1228.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
11. Corander J, Marttinen P, Mäntyniemi S: **A Bayesian method for identification of stock mixtures from molecular marker data.** *Fish Bull.* 2006; **104**(4): 550–558.
[Reference Source](#)
12. Jombart T, Devillard S, Balloux F: **Discriminant analysis of principal components: a new method for the analysis of genetically structured populations.** *BMC Genet.* 2010; **11**: 94.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
13. R Core Team: **R: A language and environment for statistical computing.** 2013.
[Reference Source](#)
14. Paradis E, Claude J, Strimmer K: **APE: Analyses of Phylogenetics and Evolution in R language.** *Bioinformatics.* 2004; **20**(2): 289–290.
[PubMed Abstract](#) | [Publisher Full Text](#)
15. Yu G, Smith DK, Zhu H, *et al.*: **ggtree: an r package for visualization and annotation of phylogenetic trees with their covariates and other associated data.** *Methods Ecol Evol.* 2017; **8**(1): 28–36.
[Publisher Full Text](#)
16. Revell LJ: **phytools: an R package for phylogenetic comparative biology (and other things).** *Methods Ecol Evol.* 2012; **3**(2): 217–223.
[Publisher Full Text](#)
17. Jolley KA, Maiden MC: **BIGSdb: Scalable analysis of bacterial genome variation at the population level.** *BMC Bioinformatics.* 2010; **11**: 595.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
18. Nguyen LT, Schmidt HA, von Haeseler A, *et al.*: **IQ-TREE: a fast and effective stochastic algorithm for estimating maximum-likelihood phylogenies.** *Mol Biol Evol.* 2015; **32**(1): 268–274.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
19. Kalyaanamoorthy S, Minh BQ, Wong TKF, *et al.*: **ModelFinder: fast model selection for accurate phylogenetic estimates.** *Nat Methods.* 2017; **14**(6): 587–589.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
20. Tonkin-Hill G: **gtonkinhill/rhierbaps: status at time of publication on CRAN (Version v1.0.1).** *Zenodo.* 2018.
<http://www.doi.org/10.5281/zenodo.1318958>

Open Peer Review

Current Referee Status:  

Version 1

Referee Report 05 October 2018

doi:[10.21956/wellcomeopenres.16000.r33842](https://doi.org/10.21956/wellcomeopenres.16000.r33842)



Sebastian Duchene 

Department of Biochemistry and Molecular Biology, University of Melbourne, Parkville, Vic, Australia

The authors report an implementation of hierBAPS for R, RhierBAPS, for determining the optimal number of clusters in population DNA sequence data. The program does not extend the methods in hierBAPS, but I appreciate that R is the default language in many bioinformatics pipelines.

I have a few suggestions:

- The program is easy to use, but I think that it is worth including a few data sets that can be readily available when the package is loaded into R. This would make the examples easier to run and understand.

- I think that it would be valuable to see some results of choosing different minimum values of K to illustrate the sensitivity of the clustering to this parameter.

Is the rationale for developing the new software tool clearly explained?

Yes

Is the description of the software tool technically sound?

Partly

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?

Yes

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?

Partly

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?

Partly

Competing Interests: No competing interests were disclosed.

Referee Expertise: Phylogenetics, phylodynamics, molecular evolution

I have read this submission. I believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.

Referee Report 17 September 2018

doi:[10.21956/wellcomeopenres.16000.r33752](https://doi.org/10.21956/wellcomeopenres.16000.r33752)



Emmanuel Paradis 

ISEM,, University of Montpellier, CNRS, EPHE, IRD, Montpellier, France

The identification of groups (or clusters) with genetic or genomic data is one the basic questions asked by population geneticists. Several methods exist with various software implementations. The software described in this note is a valuable addition to the set of R packages for population genetics and related fields. I greatly appreciate to have this method implemented in R. The analytical integration is a really great feature and that makes this method attractive.

I tried the package and the examples run very smoothly as expected. I also tried the main function with a small DNA alignment (the 'woodmouse' data in ape) and the results made sense. The graphical tools provided with the package are helpful, and the results output as a list in R make easy to use a custom graphical display. For instance, I was able to make my own display with functions in ape in just four lines of code.

I have a few comments or suggestions that the Authors may find useful for future versions of their article and/or package.

At present, it seems that the package handles SNP data. Does this mean that only biallelic DNA loci can be analysed? Can other types of biallelic genetic data be handled? What if more than two alleles are observed at a DNA site?

One suggestion for future developments would be to better integrate with other data classes, particularly from ape and adegenet which are more and more widely used. Also, ape has now efficient links with the data classes used in BioConductor, which makes possible to integrate a wide range of approaches (bioinformatics, genomics, phylogenetics, population genetics) within R.

It seems that the present package does not calculate the individual relative probabilities of assignment to the different clusters as done in Corander *et al.*¹. This might be a valuable addition for future versions, and it would help to compare the results from different methods, for instance using the nice compoplots function in adegenet.

References

1. Corander J, Marttinen P, Sirén J, Tang J: Enhanced Bayesian modelling in BAPS software for learning genetic structures of populations. *BMC Bioinformatics*. 2008; **9**: 539 [PubMed Abstract](#) | [Publisher Full Text](#)

Is the rationale for developing the new software tool clearly explained?

Yes

Is the description of the software tool technically sound?

Yes

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?

Yes

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?

Partly

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?

Yes

Competing Interests: No competing interests were disclosed.

I have read this submission. I believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.
