Benchmarking phyloregion

Barnabas H. Daru, Piyal Karunarathne & Klaus Schliep

March 07, 2020

Benchmarking `phyloregion` against other packages

In this vignette, we benchmark `phyloregion` against other similar `R` packages in analyses of standard alpha diversity metrics commonly used in conservation, such as phylogenetic diversity and phylogenetic endemism as well as metrics for compositional turnover (e.g., beta diversity and phylogenetic beta diversity). Specifically, we compare `phyloregion`'s functions with available packages for efficiency in memory allocation and computation speed in various biogeographic analyses.

First, load the packages for the benchmarking:

```
library(ape)
library(Matrix)
library(bench)
library(ggplot2)
# packages we benchmark
library(phyloregion)
library(betapart)
library(picante)
library(vegan)
library(hilldiv)
library(BAT)
library(pez)
```

We will use a small data set which comes with `phyloregion`.

```
data(africa)
# subset matrix
X_sparse <- africa$comm[1:30, ]
X_sparse <- X_sparse[, colSums(X_sparse)>0]
X_dense <- as.matrix(X_sparse)
Xt_dense <- t(X_dense)

object.size(X_sparse)
```

```
## 76504 bytes
```

```
object.size(X_dense)
```

```
## 134752 bytes
```

```
dim(X_sparse)
```

```
## [1]  30 401
```

To make results comparable, it is often desirable to make sure that the taxa in different datasets match each other. For example, the community matrix in the `hilldiv` package needs to be transposed. These transformations can influence the execution times, often only marginally. To benchmark `phyloregion` against other packages, we here use the package `bench` because it returns execution times and provides estimates of memory allocations for each computation.

1. Analysis of alpha diversity

1.1. Benchmarking phyloregion for analysis of phylogenetic diversity

For analysis of alpha diversity commonly used in conservation such as phylogenetic diversity - the sum of all phylogenetic branch lengths within an area (Faith 1992) - phyloregion is 31 to 284 times faster and 67 to 192 times memory efficient, compared to other packages!

```
tree <- africa$phylo
tree <- keep.tip(tree, colnames(X_sparse))

pd_picante <- function(x, tree){
    res <- picante::pd(x, tree)[,1]
    names(res) <- row.names(x)
    res
}

pd_pez <- function(x, tree){
    dat <- pez::comparative.comm(tree, x)
    res <- pez::.pd(dat)[,1]
    names(res) <- row.names(x)
    res
}

pd_hilldiv <- function(x, tree) hilldiv::index_div(x, tree, index="faith")
pd_phyloregion <- function(x, tree) phyloregion::PD(x, tree)

res1 <- bench::mark(picante=pd_picante(X_dense, tree),
          hilldiv=pd_hilldiv(Xt_dense,tree=tree),
          pez=pd_pez(X_dense, tree),
          phyloregion=pd_phyloregion(X_sparse, tree))
```

```
## Warning: Some expressions had a GC in every iteration; so filtering is disabled.
```

```
summary(res1)
```

```
## Warning: Some expressions had a GC in every iteration; so filtering is disabled.
```

```
## # A tibble: 4 x 6
##   expression       min   median `itr/sec` mem_alloc `gc/sec`
##   <bch:expr>  <bch:tm> <bch:tm>     <dbl> <bch:byt>    <dbl>
## 1 picante      85.48ms  92.93ms     10.3     59.4MB     0
## 2 hilldiv      934.9ms  934.9ms      1.07   170.1MB     1.07
## 3 pez          86.59ms  88.67ms      9.25    60.2MB     1.85
## 4 phyloregion   2.55ms   2.75ms    348.     883.9KB     0
```

```
autoplot(res1)
```

1.2. Benchmarking phyloregion for analysis of phylogenetic endemism

Another benchmark for phyloregion is in analysis of phylogenetic endemism, the degree to which phylogenetic diversity is restricted to any given area (Rosauer et al. 2009). Here, we found that phyloregion is 160 times faster and 489 times efficient in memory allocation.

```
tree <- africa$phylo
tree <- keep.tip(tree, colnames(X_sparse))

pe_pez <- function(x, tree){
```

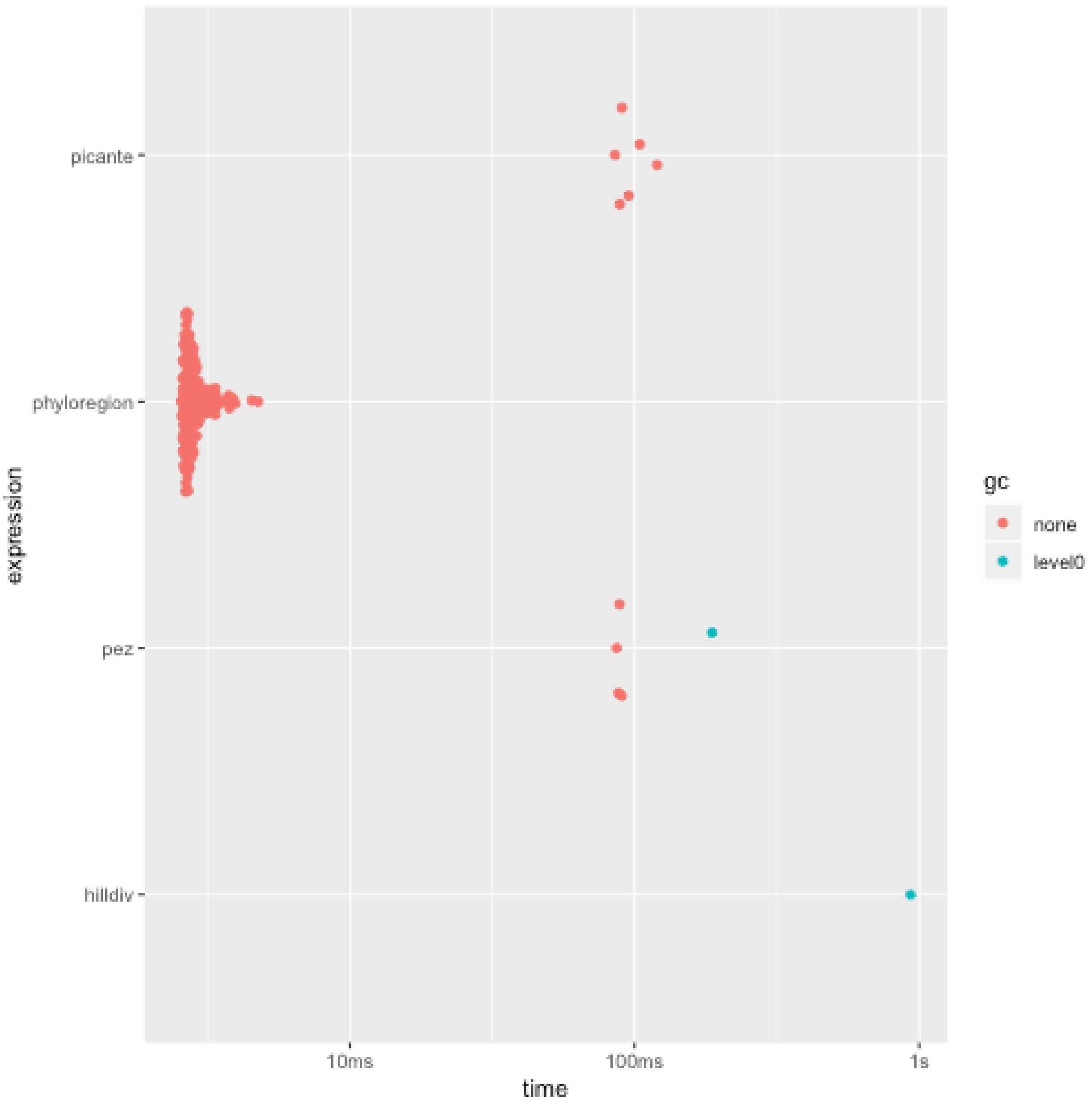


Figure 1: plot of chunk phylo_diversity

```
    dat <- pez::comparative.comm(tree, x)
    res <- pez::pez.endemism(dat)[,1]
    names(res) <- row.names(x)
    res
}

pe_phyloregion <- function(x, tree) phyloregion::phylo_endemism(x, tree)

res2 <- bench::mark(pez=pe_pez(X_dense, tree),
          phyloregion=pe_phyloregion(X_sparse, tree))
```

```
## Warning: Some expressions had a GC in every iteration; so filtering is disabled.
```

```
summary(res2)
```

```
## Warning: Some expressions had a GC in every iteration; so filtering is disabled.
```

```
## # A tibble: 2 x 6
##   expression       min   median `itr/sec` mem_alloc `gc/sec`
##   <bch:expr>  <bch:tm> <bch:tm>     <dbl> <bch:byt>    <dbl>
## 1 pez            523ms    523ms      1.91     499MB     1.91
## 2 phyloregion    3.1ms   3.36ms    285.       975KB     0
```

```
autoplot(res2)
```

2. Analysis of compositional turnover (beta diversity)

2.1. Benchmarking phyloregion for analysis of taxonomic beta diversity

For analysis of taxonomic beta diversity, which compares diversity between communities (Koleff et al. 2003), phyloregion has marginal advantage over other packages. Nonetheless, it is 1-39 times faster and 2 to 110 efficient in memory allocation than other packages.

```
chk_fun <- function(target, current)
    all.equal(target, current, check.attributes = FALSE)

fun_phyloregion <- function(x) as.matrix(phyloregion::beta_diss(x)[[3]])
fun_betapart <- function(x) as.matrix(betapart::beta.pair(x)[[3]])
fun_vegan  <- function(x) as.matrix(vegan::vegdist(x, binary=TRUE))
fun_BAT <- function(x) as.matrix(BAT::beta(x, func = "Soerensen")[[1]])
res3 <- bench::mark(phyloregion=fun_phyloregion(X_sparse),
                    betapart=fun_betapart(X_dense),
                    vegan=fun_vegan(X_dense),
                    BAT=fun_BAT(X_dense), check=chk_fun)
summary(res3)
```

```
## # A tibble: 4 x 6
##   expression       min   median `itr/sec` mem_alloc `gc/sec`
##   <bch:expr>  <bch:tm> <bch:tm>     <dbl> <bch:byt>    <dbl>
## 1 phyloregion 803.53µs 888.13µs    1016.    286.7KB     0
## 2 betapart    890.38µs 994.59µs     948.    575.4KB     2.23
## 3 vegan         1.05ms   1.24ms     683.    893.2KB     2.06
## 4 BAT           35.2ms  44.52ms      22.8    31.7MB     2.28
```

```
autoplot(res3)
```

2.2. Benchmarking phyloregion for analysis of phylogenetic beta diversity

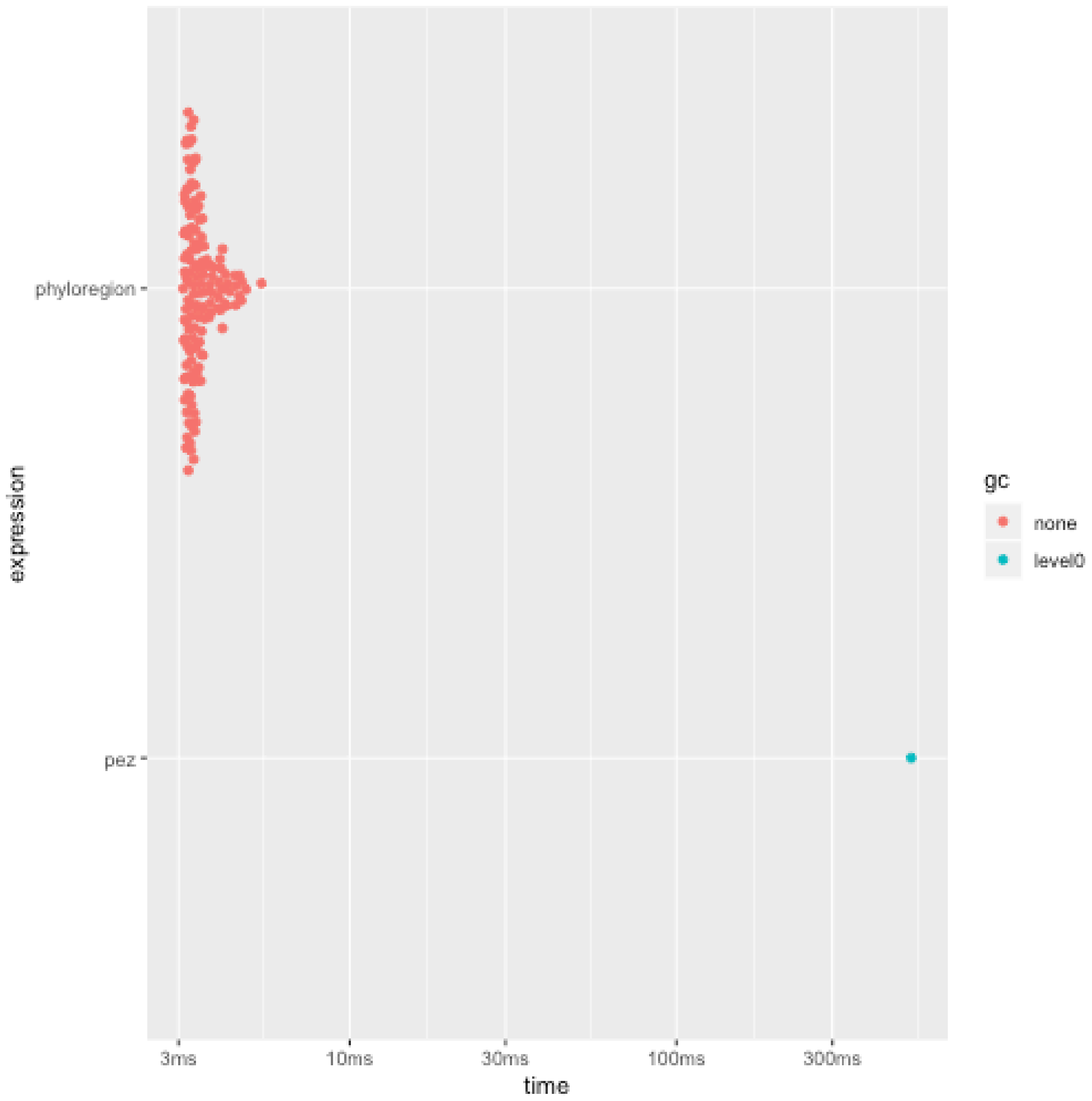


Figure 2: plot of chunk phylo_endemism

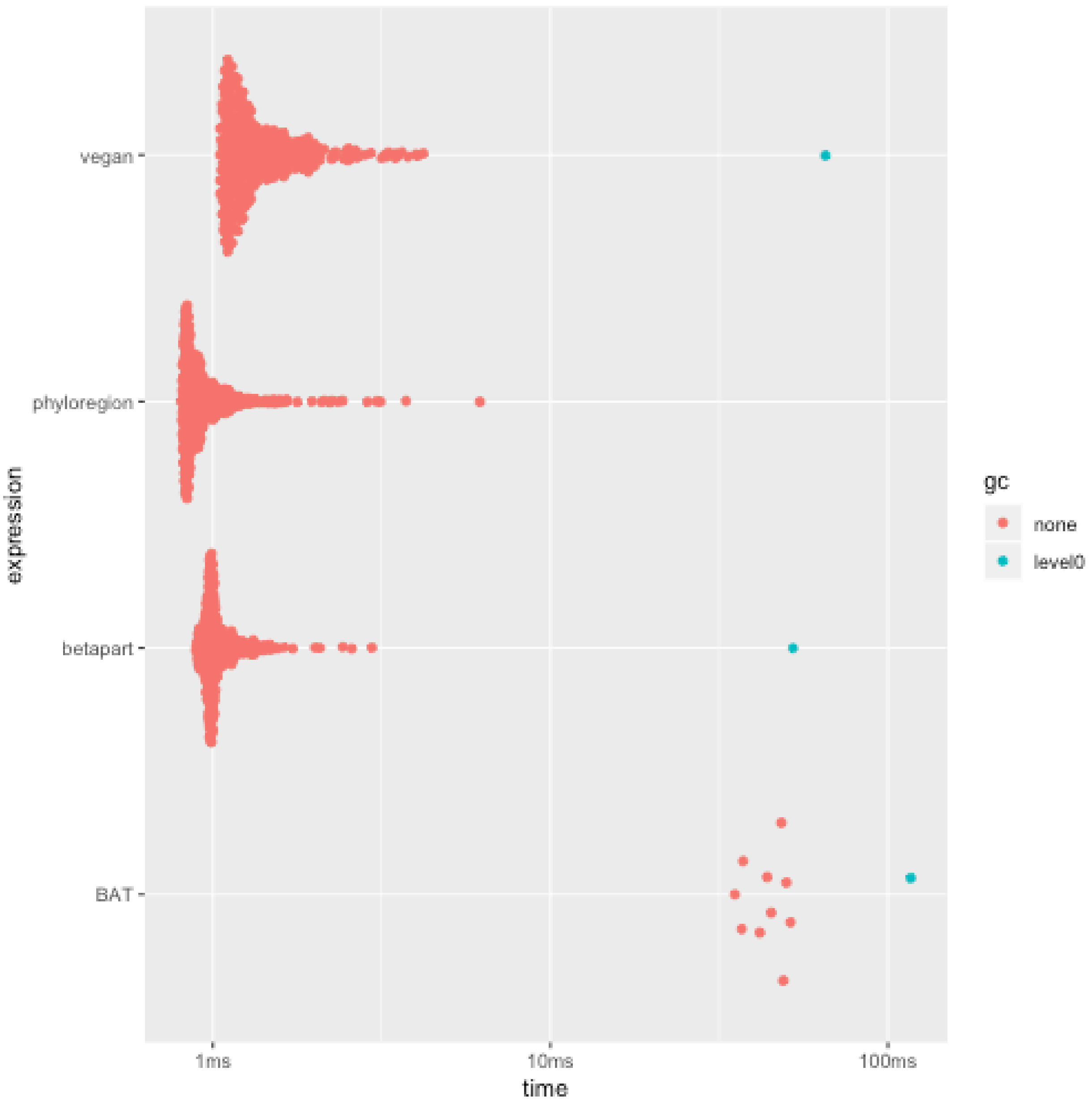


Figure 3: plot of chunk beta_diversity

For analysis phylogenetic turnover (beta-diversity) among communities - the proportion of shared phylogenetic branch lengths between communities (Graham & Fine 2008) - `phyloregion` is 3-7 times faster and 100-600 times efficient in memory allocation!

```
fun_phyloregion <- function(x, tree) phyloregion::phylobeta(x, tree)[[3]]
fun_betapart <- function(x, tree) betapart::phylo.beta.pair(x, tree)[[3]]
fun_picante <- function(x, tree) 1 - picante::phylosor(x, tree)
fun_BAT <- function(x, tree) BAT::beta(x, tree, func = "Soerensen")[[1]]

chk_fun <- function(target, current)
    all.equal(target, current, check.attributes = FALSE)

res4 <- bench::mark(picante=fun_picante(X_dense, tree),
                       betapart=fun_betapart(X_dense, tree),
                       BAT=fun_BAT(X_dense, tree),
                       phyloregion=fun_phyloregion(X_sparse, tree), check=chk_fun)
```

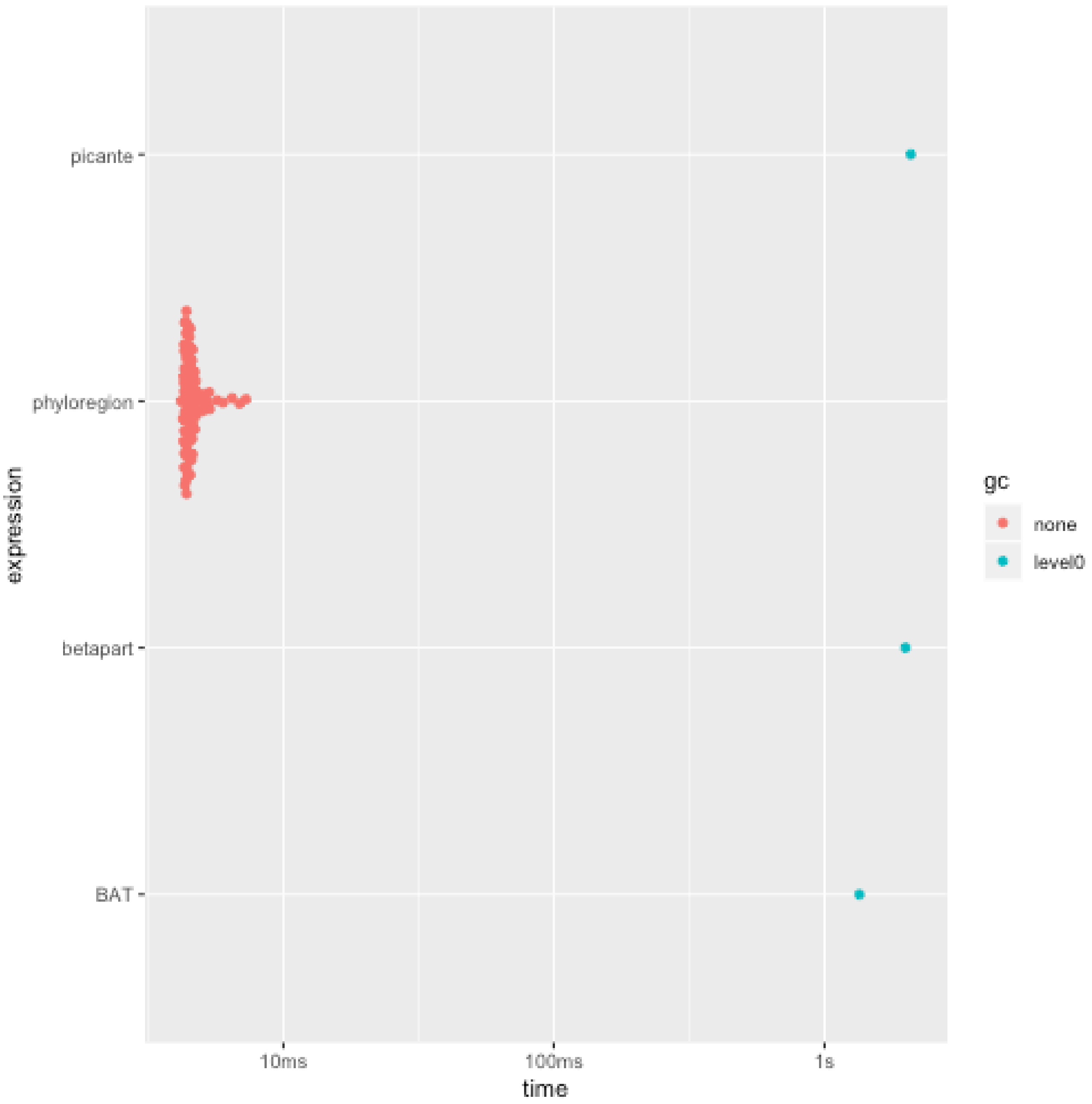
```
## Warning: Some expressions had a GC in every iteration; so filtering is disabled.
```

```
summary(res4)
```

```
## Warning: Some expressions had a GC in every iteration; so filtering is disabled.
```

```
## # A tibble: 4 x 6
##   expression       min   median `itr/sec` mem_alloc `gc/sec`
##   <bch:expr>  <bch:tm> <bch:tm>     <dbl> <bch:byt>    <dbl>
## 1 picante        2.08s    2.08s     0.480    1.24GB    0.959
## 2 betapart          2s       2s     0.501    1.24GB    1.50
## 3 BAT            1.35s    1.35s     0.742  207.32MB    0.742
## 4 phyloregion   4.19ms    4.5ms   216.    1023.17KB    0
```

```
autoplot(res4)
```



Note that for this test, `picante` returns a similarity matrix while `betapart`, and `phyloregion` return a dissimilarity matrix.

REFERENCES

Faith, D.P. (1992) Conservation evaluation and phylogenetic diversity. *Biological Conservation* **61**, 1–10.

Graham, C.H. & Fine, P.V.A. (2008) Phylogenetic beta diversity: linking ecological and evolutionary processes across space in time. *Ecology Letters* **11**, 265–1277.

Koleff, P., Gaston, K.J. & Lennon, J.J. (2003) Measuring beta diversity for presence–absence data. *Journal of Animal Ecology* **72**, 367–382.

Rosauer, D., Laffan, S.W., Crisp, M.D., Donnellan, C. & Cook, L.G. (2009) Phylogenetic endemism: a new approach for identifying geographical concentrations of evolutionary history. *Molecular Ecology* **18**, 4061–4072.

Session Infomation

```
sessionInfo()
```

```
## R version 3.6.1 (2019-07-05)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS Mojave 10.14.6
##
## Matrix products: default
## BLAS:   /System/Library/Frameworks/Accelerate.framework/Versions/A/Frameworks/vecLib.framework/Versi
## LAPACK: /Library/Frameworks/R.framework/Versions/3.6/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] pez_1.2-0         BAT_2.0.0         hilldiv_1.5.1     picante_1.8
##  [5] nlme_3.1-142      vegan_2.5-6       lattice_0.20-38   permute_0.9-5
##  [9] betapart_1.5.1    phyloregion_0.1.0 ggplot2_3.2.1     bench_1.1.1
## [13] Matrix_1.2-18     ape_5.3           knitr_1.26
##
## loaded via a namespace (and not attached):
##   [1] utf8_1.1.4              ks_1.11.7               tidyselect_0.2.5
##   [4] htmlwidgets_1.5.1       grid_3.6.1              combinat_0.0-8
##   [7] munsell_0.5.0           animation_2.6           codetools_0.2-16
##  [10] miniUI_0.1.1.1          withr_2.1.2             profmem_0.5.0
##  [13] colorspace_1.4-1        highr_0.8               rstudioapi_0.10
##  [16] geometry_0.4.5          stats4_3.6.1            ggsignif_0.6.0
##  [19] tensor_1.5              huge_1.3.4              nls2_0.2
##  [22] FD_1.0-12               mnormt_1.5-5            polyclip_1.10-0
##  [25] farver_2.0.1            coda_0.19-3             vctrs_0.2.0
##  [28] clusterGeneration_1.3.4 xfun_0.11               fastcluster_1.1.25
##  [31] R6_2.4.1                ggbeeswarm_0.6.0        pdist_1.2
##  [34] manipulateWidget_0.10.1 spatstat.utils_1.15-0   assertthat_0.2.1
##  [37] promises_1.1.0          scales_1.1.0            nnet_7.3-12
##  [40] rgeos_0.5-2             beeswarm_0.2.3          gtable_0.3.0
##  [43] caper_1.0.1             goftest_1.2-2           phangorn_2.5.5
##  [46] rlang_0.4.2             MatrixModels_0.4-1      zeallot_0.1.0
##  [49] FSA_0.8.27              scatterplot3d_0.3-41    splines_3.6.1
##  [52] lazyeval_0.2.2          acepack_1.4.1           checkmate_2.0.0
##  [55] rgl_0.100.50            yaml_2.2.0              reshape2_1.4.3
##  [58] abind_1.4-5             d3Network_0.5.2.1       crosstalk_1.0.0
##  [61] backports_1.1.5         httpuv_1.5.2            Hmisc_4.3-1
##  [64] tools_3.6.1             psych_1.9.12.31         lavaan_0.6-5
##  [67] cubature_2.0.4          raster_3.0-7            RColorBrewer_1.1-2
##  [70] Rcpp_1.0.3              plyr_1.8.4              base64enc_0.1-3
##  [73] progress_1.2.2          purrr_0.3.3             prettyunits_1.0.2
##  [76] ggpubr_0.2.5            rpart_4.1-15            deldir_0.1-23
##  [79] pbapply_1.4-2           deSolve_1.25            qgraph_1.6.5
##  [82] cluster_2.1.0           magrittr_1.5            data.table_1.12.8
##  [85] SparseM_1.78            mvtnorm_1.0-11          whisker_0.4
##  [88] hms_0.5.2               mime_0.8                evaluate_0.14
##  [91] xtable_1.8-4            jpeg_0.1-8.1            mclust_5.4.5
##  [94] gridExtra_2.3           compiler_3.6.1          tibble_2.1.3
##  [97] maps_3.3.0              KernSmooth_2.23-16      crayon_1.3.4
## [100] hypervolume_2.0.12      htmltools_0.4.0         mgcv_1.8-31
## [103] corpcor_1.6.9           later_1.0.0             Formula_1.2-3
## [106] tidyr_1.0.0             expm_0.999-4            magic_1.5-9
## [109] apTreeshape_1.5-0       subplex_1.5-4           MASS_7.3-51.4
## [112] ade4_1.7-13             cli_2.0.1               quadprog_1.5-8
## [115] parallel_3.6.1          igraph_1.2.4.2          BDgraph_2.62
## [118] pkgconfig_2.0.3         numDeriv_2016.8-1.1     foreign_0.8-72
## [121] sp_1.3-2                pbivnorm_0.6.0          vipor_0.4.5
## [124] webshot_0.5.2           stringr_1.4.0           digest_0.6.23
## [127] phytools_0.6-99         rcdd_1.2-2              spatstat.data_1.4-0
## [130] rmarkdown_1.18          fastmatch_1.1-0         htmlTable_1.13.3
## [133] shiny_1.4.0             gtools_3.8.1            quantreg_5.54
## [136] rjson_0.2.20            geiger_2.0.6.2          lifecycle_0.1.0
## [139] glasso_1.11             jsonlite_1.6            fansi_0.4.1
## [142] pillar_1.4.2            fastmap_1.0.1           plotrix_3.7-7
## [145] survival_3.1-8          glue_1.3.1              fdrtool_1.2.15
## [148] spatstat_1.61-0         png_0.1-7               class_7.3-15
## [151] stringi_1.4.3           latticeExtra_0.6-29     dplyr_0.8.3
## [154] e1071_1.7-3
```