phyloflows: Calculating sources, onward transmissions and flow ratios

Xiaoyue Xi and Oliver Ratmann

2019-09-10

This vignette describes how a number of key summary statistics of inferred transmission flows can be easily calculated with **phyloflows** `source.attribution.mcmc.getKeyQuantities` function. Please work through the vignette *phyloflows: Estimating transmission flows under heterogeneous sampling – a first example* before you go ahead here.

Getting started

We continue our "First_Example". The following code chunk contains all code needed, up to running **phyloflows** MCMC routine. The only change is that the number of iterations is now $50,000$. The MCMC should take about 2 minutes to run.

```
require(data.table)
require(phyloflows)
data(twoGroupFlows1, package="phyloflows")
dobs <- twoGroupFlows1$dobs
dprior <- twoGroupFlows1$dprior
mcmc.file <- file.path(getwd(),'twoGroupFlows1_mcmc.RData')
control <- list(seed=42, mcmc.n=5e4, verbose=0)
mc <- phyloflows:::source.attribution.mcmc(dobs, dprior, control)
```

Sources of transmission for each group

OK, so now we have samples from posterior distribution of transmission flows within and between the two population groups,
$$\pi = (\pi_{11}, \pi_{12}, \pi_{21}, \pi_{22}).$$

One important summary statistic are the sources of transmissions into each recipient group, defined by

$$\eta = (\eta_{11}, \eta_{21}, \eta_{12}, \eta_{22})$$

where $\eta_{ij} = \pi_{ij} / \sum_s \pi_{sj}$.

Onward transmissions from each group

Another important summary statistic are the proportions of transmissions that originate from each group, defined by

$$\nu = (\nu_{11}, \nu_{21}, \nu_{12}, \nu_{22})$$

where $\nu_{ij} = \pi_{ij} / \sum_s \pi_{is}$.

Transmission flow ratios

Yet another important summary statistic are ratios of transmission flows, defined by

$$\rho_{ij} = \pi_{ij} / \pi_{ji}.$$

Calculating key quantities

phyloflow has a function to calculate the above summary statistics. The basic syntax is as follows:

```
#   specify list of user options
#
#   burnin.p: proportion of samples to discard as burn-in
#            (only needed when the burn-in was not already removed)
#
#   thin: keep every thin-nth iteration
#        (only needed when thinning was not already performed)
#
#   quantiles: quantiles of the marginal posterior distributions
#             that will be computed
#
#   flowratios: list of vectors of 3 elements. The 3 elements
#              specify the name of the flow ratio (first element),
#              the enumerator of the flow ratio (second element),
#              and the denominator of the flow ratio (third element)
control <- list(  burnin.p=0.05,
                  thin=NA_integer_,
                  quantiles= c('CL'=0.025,'IL'=0.25,'M'=0.5,'IU'=0.75,'CU'=0.975),
                  flowratios= list( c('1/2', '1 2', '2 1'), c('2/1', '2 1', '1 2'))
                )
ans <- source.attribution.mcmc.getKeyQuantities(mc=mc,
      dobs=dobs, control=control)
#>
#> Removing burnin in set to  5 % of chain, total iterations= 625
#> Computing flows...
#> Computing WAIFM...
#> Computing sources...
#> Computing flow ratios...
ans
#>    TR_TARGETCAT REC_TARGETCAT        CL         CU         IL         IU          M
#>  1:            1             1 0.28195785 0.39197078 0.31601498 0.35443253 0.33486791 33.5%\n[28.2%
#>  2:            1             2 0.02758612 0.07486695 0.04005584 0.05609872 0.04762430    4.8%\n[2.8%
#>  3:            2             1 0.04018723 0.09412431 0.05469187 0.07330903 0.06347945     6.3%\n[4%
#>  4:            2             2 0.49018456 0.61277059 0.52964502 0.57339978 0.55150290   55.2%\n[49%
#>  5:            1             1 0.80990329 0.92567905 0.85462460 0.89464407 0.87550587   87.6%\n[81%
#>  6:            1             2 0.07432095 0.19009671 0.10535593 0.14537540 0.12449413   12.4%\n[7.4
#>  7:            2             1 0.06529071 0.15353627 0.08875570 0.11934096 0.10311869  10.3%\n[6.5%
#>  8:            2             2 0.84646373 0.93470929 0.88065904 0.91124430 0.89688131  89.7%\n[84.6%
#>  9:            1             1 0.77239794 0.89672712 0.81795244 0.86118964 0.84087458 84.1%\n[77.2%
#> 10:            2             1 0.10327288 0.22760206 0.13881036 0.18204756 0.15912542 15.9%\n[10.3%
#> 11:            1             2 0.04608185 0.12514978 0.06677337 0.09366908 0.07952006    8%\n[4.6%
#> 12:            2             2 0.87485022 0.95391815 0.90633092 0.93322663 0.92047994  92%\n[87.5%
#> 13:           NA            NA 0.37863188 1.45436845 0.59242003 0.94657717 0.75002692  0.75\n[0.38
#> 14:           NA            NA 0.68758367 2.64108877 1.05643790 1.68799153 1.33328548  1.33\n[0.69
#>                  LABEL2      STAT FLOWRATIO_CAT
#>  1: 33.5% (28.2%-39.2%)     flows          <NA>
#>  2:    4.8% (2.8%-7.5%)     flows          <NA>
#>  3:      6.3% (4%-9.4%)     flows          <NA>
#>  4:   55.2% (49%-61.3%)     flows          <NA>
#>  5:   87.6% (81%-92.6%)     waifm          <NA>
#>  6:    12.4% (7.4%-19%)     waifm          <NA>
```

```
#>  7:  10.3% (6.5%-15.4%)      waifm         <NA>
#>  8: 89.7% (84.6%-93.5%)      waifm         <NA>
#>  9: 84.1% (77.2%-89.7%)    sources         <NA>
#> 10: 15.9% (10.3%-22.8%)    sources         <NA>
#> 11:     8% (4.6%-12.5%)    sources         <NA>
#> 12:   92% (87.5%-95.4%)    sources         <NA>
#> 13:    0.75 (0.38-1.45) flow_ratio          1/2
#> 14:    1.33 (0.69-2.64) flow_ratio          2/1
```

Note

Note it is also possible to specify a file name to MCMC output or aggregated MCMC output, and it is also possible to input aggregated MCMC output. Please look up the package help for further instructions.

```
?phyloflows:::source.attribution.mcmc.getKeyQuantities
```