A user's guide to estimating dietary parameters using IsotopeR 0.4

November 18, 2012

Contents

IsotopeR is a stable isotope mixing model used to estimate dietary parameters at the population-, group-, and individual-level. IsotopeR allows users to include a number of components common to stable isotope models that are not currently available in a single statistical package. We intend to make the IsotopeR user interface simple and intuitive and we welcome any feedback (jbhopkins3@gmail.com) that will help to continue to refine the tool.

Installing IsotopeR

- Install JAGS from `http://sourceforge.net/projects/mcmc-jags/` (IsotopeR has been tested on JAGS v2.2.0, 2.1.0 and 1.0.4 under R 2.12 and 2.13)
- Install IsotopeR and it's dependencies from CRAN. Type the following to install from the command line:

```
> install.packages("IsotopeR", dep=T)
```

- Mac users must also install the tcltk software, available at: `http://cran.r-project.org/bin/macosx/tools/`.

Using IsotopeR

Here we show how to use IsotopeR by analyzing an example dataset. Once IsotopeR is installed you can download an example dataset `http://people.biology.ufl.edu/troutinthemilk/R_software_files/IsotopeR_Data.zip` and extract the data files (Note: We recommend you format your project data using these files as templates).

Load IsotopeR and view the graphical user interface (Fig.1) using the following R code:

```
> library(IsotopeR)
> IsotopeR()
```

If everything is correctly installed, the main IsotopeR GUI 1 window will appear.

Begin a new run by selecting `Analysis -> New Run` from the menu. A new window resembling the image in Figure 2 will appear.

For the example analysis we will use 4 of the 6 data input files located in the data folder: Mixtures, Sources, SourceCD, and MeasurementError (Note: Click the question mark next to each field to get detailed information about each file; this information is also available below). To enter data input files, click on each

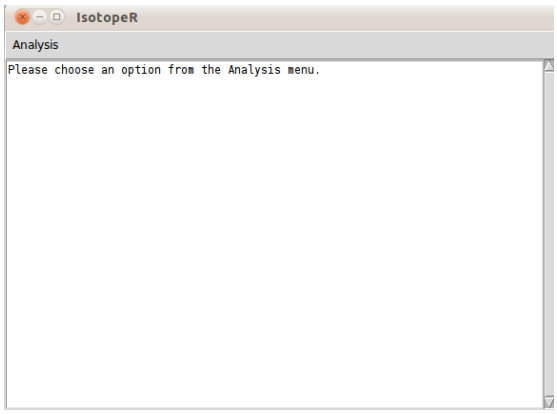

Figure 1: Main IsotopeR GUI window.

button on the left hand side of the IsotopeR analysis window; this will navigate you to your hard drive and allow you to select the appropriate data file.

For the first run, upload Mixtures, Sources, and MeasurementError. Control parameters and plot options are located below the file upload section of the IsotopeR GUI. Do not change any entries for the control parameters or switch off any plots (i.e., keep IsotopeR in default mode) except for MCMC runs; in the interest of time, change the default in this field to 1,000. Click the `Run IsotopeR` button and the estimation procedure will begin. A progress bar will appear in the R console. When the estimation process has terminiated, plots will appear, and parameter estimates and diagnostic output will appear in the R console. These estimates will also be automatically written to a text file in your current working directory in R. In this case, the file is called `SampleOutput.txt`.

IsotopeR also has the ability to open old analyses and build plots without having to rerun the estimation. This is especially useful for more complex modeling activities and larger datasets that may take a long time to run. Previous analyses can be opened from the `Analysis -> Load Previous Run` menu (in main IsotopeR GUI window; Figure 3). Load the previous analysis, `SampleOutput.Rdata`, click FALSE for color plots, and run IsotopeR; this will create grayscale plots for publication. For the last example run, upload the 4 input files listed above and run IsotopeR. Compare figures and parameter estimates when running an analysis that includes and excludes source concentrations.

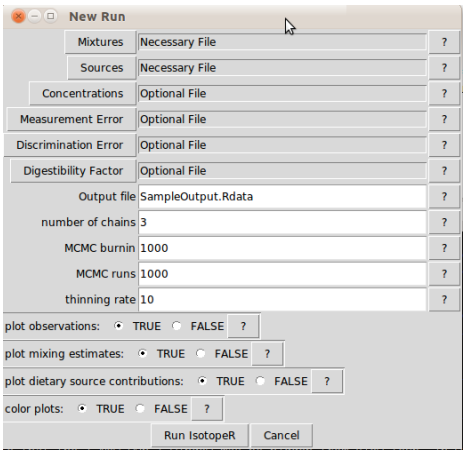

Figure 2: IsotopeR analysis window.

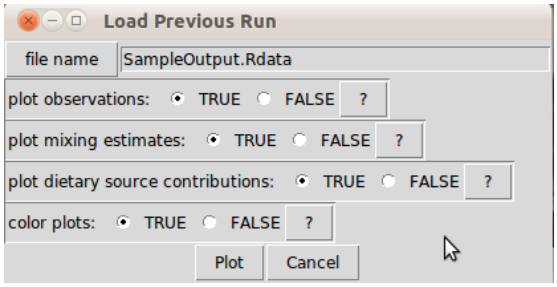

Figure 3: Previous run window.

Input files

Mixtures: The first n columns in this data input file are the isotope values associated with consumers, where n is the number of isotopes used in the analysis. The last two columns designate the group and individual assignments. If there is no group structure, then column $n+1$ will contain "1" for all individuals. If designating multiple groups, the group identity will be determined by the variable in the column. Individuals in the first group should be designated as "1," the second group as "2," etc. The last column identifies each individual. If you have no repeated measurements for individuals, then each individual should be designated by a unique integer (e.g., 1, 2, 3...); individuals with repeated measures should be designated using the same number (e.g., 1, 1, 1, 2, 2, 2...). Note that IsotopeR does not require consecutive integers for the individual column, however the estimated values are necessarily reported as consecutive, so it may be difficult to map the input to output if you use nonconsecutive integer labels in the data file.

δC	δN	group	individual
-22.1	5.2	1	1
-22.2	5.5	1	2
-22.0	5.0	1	3

Table 1: Mixture data example

Sources: Each source is a sample of a consumer's dietary items (may be a sample of the same species or an aggregate of species). The first n columns in this data input file are the isotope values associated with each sampled dietary item, where n is the number of isotopes used in the analysis. Isotope values need to be in the same order as the mixture data file (e.g., column 1 in Mixtures and Sources contain $\delta^{13}C$ values). The next column $(n+1)$ identifies the source to which the sampled dietary item belongs. The last column (subsource) identifies different species or taxa within each source aggregate; this feature assigns equal weight to each subsource.

δC	δN	source	subsource
-22.2	2.9	plants	1
-23.0	2.6	plants	1
-22.6	3	plants	2
-28.4	8.1	ungulates	1
-27.9	7.9	ungulates	1

Table 2: Source data example

SourcesCD: The first n columns in this data input file are the concentration data for each sample, where n is the number of elemental concentrations used in the analysis (e.g., [C], [N]). Columns with elemental concentrations need to match Sources and Mixtures (e.g., column 1 in this file and Sources files contain [C] and $\delta^{13}C$ values, respectively). Column $n+1$ identifies the source in which the set of concentrations belong. The last column links sampled dietary item concentrations to each subsource. This feature assigns equal weight to each sub-source's elemental concentrations and should be consistent with Sources file.

[C]	[N]	source	subsource
45	4	plants	1
45	5	plants	1
45	5	plants	2
40	13	ungulates	1
42	12	ungulates	1

Table 3: Source concentration example.

Measurement Error: This is the error associated with mass spectrometry/EA analysis. This data input file contains all isotopic measurements for standards. Isotope values need to be in the same order as other data files (e.g., column 1 in Measurement Error, Mixtures, and Sources files contain $\delta^{13}C$ values).

δC	δN
-12.7	5.5
-12.6	5.4
-12.5	5.5
-12.5	5.4

Table 4: Measurement error example.

DiscrimSD: This data input file contains the standard deviations associated with the estimated average discrimination factors measured in controlled diet studies. The first n columns are the standard deviations associated with each mean discrimination value for the associated isotope. The last column denotes the source identification for the standard deviations.

δC	δN	source
1.2	1.0	plants
0.6	0.5	ungulates
0.4	0.5	insects

Table 5: Discrimination standard deviation example.

Digest: This input file contains the digestibility of different sources. The first n columns contain the digestibility for n source isotopes. The last column is the source identification code defined in Sources.

digestC	digestN	source
0.5	0.9	plants
0.6	1	ungulates
0.5	1	insects

Table 6: Digestibility example

Control Parameters

A user can change the control parameters for each analysis. Each field is described below:

Number of chains: The number of independent Markov chains.

MCMC burnin: The length of the chain discarded at the beginning of the run. This is interpreted as the length of time it takes for the MCMC to stabilize.

MCMC runs: The total number of iterations per chain, which includes burnin.

Thinning rate: Reduces the sample size to every n^{th} iteration; this is used to reduce autocorrelation in the chain.

Run parallel: If TRUE, runs mcmc chains in parallel on multicore machines. Note that the Deviance Information Criterion (DIC) cannot be reported when using the parallel feature and so it is automatically turned off when this option is TRUE. Otherwise DIC is automatically reported.

Error messages

A user may receive several warnings during a model run. These errors are associated with JAGS and are not always well documented by the package maintainers. Generally, these errors are related to the model not converging. Therefore, you may need to rerun the model with more runs chains and may also need a higher thinning rate.

Further details on the JAGS program can be found in the JAGS manual and available for download at `http://sourceforge.net/projects/mcmc-jags/`. A gentle introduction is provided in the document by N. Thompson Hobbes 'An Ecological Modeler's Primer on JAGS',. JAGS model syntax is compatible with the BUGS language. Users unfamiliar with the BUGS language can find many tutorials at the WinBUGS site `http://www.mrc-bsu.cam.ac.uk/bugs/`.

Output

After a run is finished results will be saved to an image file (`.Rdata` file) and a text file (`.txt`), both located in your current working directory. Files are formatted in a matrix with rows given by the parameter names (defined below). The first two columns are the mean and standard deviation of the posterior probability distribution. Quantiles (2.5%, 25%, 50%, 75%, 97.5%) for this sampling distribution are reported in respective columns, followed by the Rhat values (a metric of convergence that should be less than 1.2 or the model should be rerun with a longer MCMC chain).

Plots

plot observations: A plot of source and mixture isotope values.

plot mixing estimates: A plot of the estimated mixing space. Estimated sources and mixtures are displayed with their 95% credible intervals.

plot dietary source contributions: A plot of the smoothed histograms of the population-level (solid), group-level (dashed), and individual-level (transparent) estimated dietary contributions.

Estimated parameters

mu.source(z,i): Mean isotope value for source z, isotope i.

sd.source(z,i): Mean isotope value of the standard deviation for source z, isotope i.

mu.conc(z,i): Mean elemental concentration for source z, isotope i.

sd.conc(z,i): Mean of the standard deviation for source z, isotope i.

mu.mix(x,i): Isotope value for individual x, isotope i.

rho.mat(z,i,j): Correlation between isotopes i and j in source z.

p(x,z): Proportional dietary contribution for individual x, source z.

p.pop(z): Population-level proportional dietary contribution for source z.

sd.me(i): Measurement error (standard deviations) for isotope i.

sd.res(i): The residual error term (standard deviation) for isotope$i$.