A user's guide to estimating dietary parameters using IsotopeR 0.3.1

November 7, 2011

Contents

IsotopeR is a stable isotope mixing model used to estimate dietary parameters at the population, group, and individual levels. IsotopeR includes most features common to such analyses. We intend to make the IsotopeR user interface simple and intuitive and we welcome any feedback that can help to continue to refine the interface. An example folder with data files formatted for IsotopeR is available at `http://people.biology.ufl.edu/troutinthemilk/R_software_files/IsotopeR_Data.zip`.

Installing IsotopeR

- Install JAGS from `http://sourceforge.net/projects/mcmc-jags/` (IsotopeR has been tested at various times on JAGS v3.1, 2.2.0, 2.1.0 and 1.0.4 under R v2.14, 2.13, and 2.12)
- 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 run IsotopeR and analyze an example data set. Once the IsotopeR package is installed using the instructions above, download the example dataset from `http://people.biology.ufl.edu/troutinthemilk/R_software_files/IsotopeR_Data.zip` and extract the data files. Note: when analyzing your own data these files can be used as a format guide.

Load IsotopeR and view the user interface using the following R code:

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

If everything is correctly installed, a window resembling Figure 1 will pop up

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

For this analysis we will use the example isotopic mixture data and diet source isotope values. To enter data input files, click on each field (button) on the left hand side of the screen. This will navigate you to your hard drive and allow you to select the appropriate data file. To include the isotopic mixture data click the

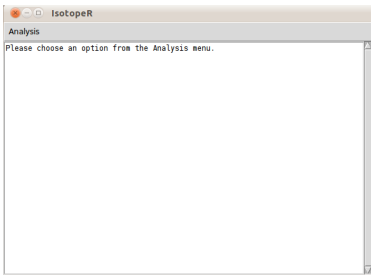

Figure 1: Main IsotopeR analysis window.

button labeled `Mixtures`, and select the Mixtures.csv file. To include the source isotope values, click on the button labeled `Sources`, select the Sources.csv file. Data is provided for the subsequent fields as well, but these files are not required to obtain parameter estimates. Note: Click the question mark next to each field to get more detailed information.

We will keep the same output file name, `SampleOutput.Rdata`, but we do want to increase the MCMC runs; change this value to 50000. We will keep plot observations and plot mixing estimates on, but will turn off the dietary source contributions plot, so click FALSE on this option. We can now click the `Run IsotopeR` button and the estimation procedure will begin. Once clicked, a progress bar will appear in the R terminal. When the estimation has completed plots will appear,diagnostic output and parameter estimates will appear in the 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`.

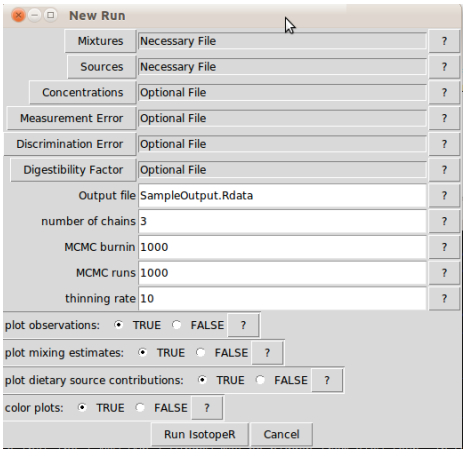

Figure 2: Data analysis interface.

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 models and large datasets that may take a long time to run. Previous analyses can be opened from the `Analysis -> Load Previous Run` menu from the main IsotopeR window, creating a window like Figure 3. Load the previous analysis, `SampleOutput.Rdata`. To make greyscale plots for publication, click FALSE in the `color plots` field.

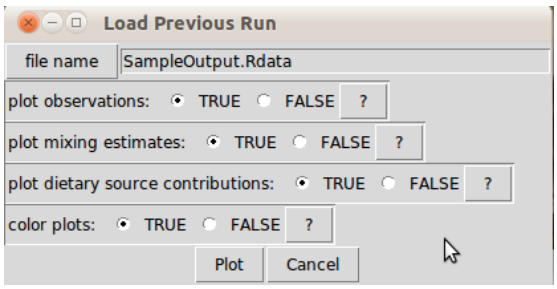

Figure 3: Previous run window.

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 the column labels δX and δY will be used as axis labels.

δX	δY	group	individual
-21.1	5.8	1	1
-21.8	4.3	1	2
-21.0	4.7	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.

δX	δY	source	subsource
-22.2	2.9	plants	1
-21.9	2.7	plants	1
-21.7	2.6	plants	2
-21.2	3.4	plants	2
-21.4	3.7	animals	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.

[X]	[Y]	source	subsource
45	5	plants	1
42	6	plants	1
42	4	plants	2
45	5	plants	2
10	3	animals	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).

δX	δY
-12.7	5.7
-12.6	5.9
-12.9	5.7
-12.7	5.6

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.

δX	δY	source
1.1	1.1	plants
2.6	0.9	animals
0.9	0.7	human food

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.

digestX	digestY	source
1	.52	plants
1	.47	animals
1	1	human food

Table 6: Digestibility example

Control Parameters

The user can change the control parameters for an MCMC run using the following fields:

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.

A user may receive several warnings during a model run. These errors are associated with JAGS and are not 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 '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, an `.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 the observed isotope source and mixture values. Each source is identified by different plotting symbols. When the color option is turned on, subsources can be distinguished by different shades of the same color.

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 estimates.

Parameters

mu.conc(y,z): Mean concentration for element y, source z.

mu.mix(x,z): Isotopic mixture value for individual x, source z (note: these are joint estimation values that include all sources of error).

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

p(x,z): Individual-level proportional dietary contribution for individual x, source z.

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

rho.source(z): Correlation values between isotopes in source z.

sd.conc(i, z): Concentration standard deviations for isotope i and source z.

sd.me(i,z): Measurement error standard deviations for isotope i and source z.

sd.res(i,j): The residual error term (standard deviation) for isotopes i and j.

sd.source(i,z) Source standard deviation for isotope i and source z.