Steven T. Hoekman. 2021. Multi-observer methods for estimating uncertain species identification. *Ecosphere*.

Metadata S3: R computer code for simulation analyses of multi-observer methods

---

Author

Steven T. Hoekman
Wild Ginger Consulting
P. O. Box 182
Langley, WA, USA 98260
steven.hoekman@protonmail.com

File list (found within DataS3-R_Code_for_Simulations)

R code files

`generate_simulation_data.R`	Function for generating simulated survey data
`optimization_functions.R`	Functions for likelihood optimization
`supplemental_functions.R`	Supplemental functions for analyses and summaries
`simulations_r_code.R`	R code for conducting simulation analyses
`likelihood_equations.R`	Generates pre-computed values for computations

Example simulation input/output files

`omnibus_m_22_example.csv`	Example simulation input file
`omnibus_m_22_example_output.csv`	Example simulation output file

Dependencies (**required)**

R packages (available at <https://cran.r-project.org/web/packages/>)

`plyr`	Programming tools, data manipulation
`dplyr`	Data frame manipulation
`doSNOW`	Back end for parallel execution
`mrds`	Delta method for arbitrary functions
`actuar`	Truncated Poisson distributions
`nnet` (only for multinomial model sims)	Log-linear multinomial models
`gtools` (only for likelihood_equations.R)	Numeric tools

Files description

generate_simulation_data.R

Function ‘generate.simulation.data.f’ that accepts inputs specifying model structure and true parameter values and returns simulated survey data.

optimization_functions.R

Functions that accept input of simulated survey data for multi-observation method (MOM) and single-observation method (SOM) models and returns the minimized −log(likelihood) (plus any penalty terms), estimated parameter values, and the Hessian matrix.

supplemental_functions.R

Functions providing supplemental services, such as: random sampling for simulating surveys; likelihood computations; and formatting, summary, and error-checking of data and output.

simulations_r_code.R

R computer code for conducting simulation analyses, including loading required R packages, specifying input files, and executing statistical analyses.

likelihood_equations.R

Functions for generating R list objects ‘likelihood.equations’ containing pre-computed values for likelihood equations calculating probabilities for observed groups. List objects required for simulation analyses are provided in DataS2-Input_Files_for_Simulations.

omnibus_m_22_example.csv

An example .csv format file specifying simulation inputs for MOM models with 2 observation states, 2 true species states. Includes distinct models with groups of size one, homogeneous groups, and heterogeneous groups. See MetadataS2.pdf for specifications for input files.

omnibus_m_22_example_output.csv

An example .csv format file containing output from simulation analyses for the example input file above. These data can be replicated by executing the R command “set.seed(1859) prior to executing simulation code to specify the seed for generating random numbers.

Additional to general documentation below, each included .R file contains detailed documentation comments in code describing specific code, functions, and variables.

Simulation engine description

The 'omnibus', 'covariate', 'distinct observer' statistical simulations described in the companion article and Appendix S2 can be conducted in the R statistical computing environment (R Development Core Team 2020) using the files provided here, in conjunction with input files provided in DataS2-Input_Files_for_Simulations. To conduction these or other user-specified simulations, R code in `simulations_r_code.R` requires:

- Functions provided in files `generate_simulation_data.R`, `optimization_functions.R`, and `supplemental_functions.R`
- A .csv format file specifying simulation inputs (see MetadataS2.pdf)
- The 'likelihood.equations' list objects providing pre-computed values (provided in DataS2-Input_Files_for_Simulations)
- R packages listed in the "Required R packages" section at the beginning of `simulations_r_code.R`

The simulation engine supports analyses for multi-observation method (MOM) and single observation method (SOM) models. Models may include: groups of size 1, homogeneous groups, and heterogeneous groups; covariates predicting true species probabilities (psi) and classification probabilities (theta); secondary observers with identical or distinct classification probabilities; and un-modeled heterogeneity in data arising from a covariate predicting true species probabilities, a covariate predicting classification probabilities, or distinct classification probabilities among secondary observers. Simulations support:

- One primary and 1-4 secondary observers.
- One (SOM models) or 2 (MOM models) observation methods.
- Without covariates, 2-4 true species states and observation states.
- With a covariate predicting true species probabilities *or* classification probabilities, 2-3 true species states and observation states.
- With a covariate predicting true species probabilities *or* classification probabilities, boot-strapped estimates of standard errors and confidence intervals for overall mean probabilities.
- With separate covariates predicting true species probabilities *and* classification probabilities, 2 true species states and observation states.
- Heterogeneous groups composed of individuals of 2 true species states.
- With heterogeneous groups, models with a covariate predicting true species states use the "encounter" model described in Appendix S2 of the companion article; all other models with heterogeneous groups use the "constant" model described in the main article.

Also supported are multinomial logit models that estimate observed species proportions assuming no species misidentification.

<u>Simulation inputs</u>

All MOM and SOM models use the R 'optim' function to minimize the –log(likelihood) using the L-BFGS-B optimization method of Byrd et al. (1995), which allows box constraints giving lower and upper bounds to values of individual parameters. Appropriate lower box constraints

are generated automatically. Upper box constraints default to 'Inf' (i.e. no constraint), but can be specified for classification probabilities by users as described in MetadataS2.pdf.

Additional to box constraints, penalty functions in model optimization functions constrain optimization by adding penalty terms to the –log(likelihood) if true species probabilities or heterogeneous group probabilities take inadmissible or unrealistic values. MetadataS1.pdf provides description and examples of penalty functions.

Five sets of initial parameter values are automatically generated for each distinct model. For each simulation repetition, each set is used in succession until achieving acceptable optimization, or the repetition is removed from simulation output.

Simulation inputs specified in the .csv format files are summarized in data frame 'sim_profiles', which is used by the function 'generate.simulation.data.f' to produce R list object 'sim_data' with 4 elements. The first element 'survey_data' is a data frame with simulated survey data and, if applicable, fields for covariate values and key values (Table 1). Key values either match observed groups (i.e. counts of individuals classified to each observation state) for each observer to a key table of unique observed groups in list element 'observation key' (Table 2), or match binned values for a covariate (described in Appendix S2 of the companion article) predicting true species probabilities to key table in list element 'psi_key'. Key tables define a unique key value for each unique observed group or unique covariate value, so that computational load during optimization can be substantially reduced by only computing probabilities for keyed items.

The final list element 'mean_probabilities' contains two vectors giving overall mean values (from the true model) of true species probabilities and classification probabilities across all groups for models with covariates predicting these probabilities. Named vector elements are 'psi_[*b*]' and 'theta.[*observer*].[*ab*]', where bracketed suffixes denote true species state *b*, observation state *a*, and observer identity for primary 'p' and secondary observers "s1" to "s4".

Simulation outputs

List 'model_results' stores statistical output from optimization functions, consisting of: parameter estimates ('par'), the −log(likelihood) + any penalty term(s) ('value'), and the Hessian matrix for estimation of the variance-covariance matrix ('hessian').

Data frame 'output_global' contains summarized statistical output, with rows giving results for each distinct model. The left-most fields give simulation inputs (see MetadataS2.pdf for descriptions of fields and naming conventions). To the right are statistics for estimated parameters (Table 3). Field 'rep.tru' is the count of successful repetitions for each distinct model. For simulations including predictive covariates, output includes estimates of overall means for true species or classification probabilities (see Appendix S2 in the companion article) and associated mean error, root mean squared error, and 95% confidence interval coverage. If inputs specify bootstrap resamples, output for these overall means also includes bootstrapped estimates of standard errors and 95% confidence interval coverage (see Appendix S2 in the companion article). Overall mean parameters follow naming conventions in MetadataS2.pdf.

For simulations including un-modeled heterogeneity in data, 'output_global' includes fields for the true parameter values used to generate data and field 'heterogeneity', which takes value 1 for 'covariate" simulations with an un-modeled predictive covariate and takes value 2 for "distinct observer" simulations with un-modeled differences in classification probabilities among secondary observers.

Upon completion, reports on simulation speed and any error messages are printed to the console window, and output is written to the hard drive in .csv file format with "_output" appended the name of the input file. For lengthy simulations, simulation output is periodically written to the hard drive.

Workflow for conducting simulation analyses

To execute statistical analyses for multi-observer method simulations:

1) Place the .R format files in DataS3 in the R working directory.
2) Place the .R and .csv format files in DataS2-Input_Files_for_Simulations in the R working directory.
3) Select a simulation input .csv file provided in DataS2.
   Alternatively, use the example file `omnibus_m_22_example.csv` provided here or create your own input file.
4) In an R editor or integrated R development environment (e.g., *RStudio* by RStudio, Inc., Boston, Massachusetts, USA, <https://rstudio.com>), open the first four .R script files above.
5) Execute all R code in `generate_simulation_data.R`, `optimization_functions.R`, and `supplemental_functions.R` to load these functions into the R environment.
6) In `simulations_r_code.R`, install and load the R packages in the 'Required R packages' section.
7) In the following section, execute R code to register the 'doSNOW' parallel execution backend.
   Parallel processing dramatically increases speed of simulations.
8) In the 'Import simulation profiles' section, specify the simulation input .csv from (3), and then execute code in this section to define profile(s) for simulation analyses and automatically load the appropriate 'likelihood_equations' list.
   I recommend executing code in this section (except loading 'likelihood.equations') prior to each simulation analysis.
9) Conduct simulation analyses by executing R code in one of the following code sections:

MOM/SOM model simulations

Conducts "omnibus," "covariate," and "distinct observer" simulations for MOM and SOM models *without* un-modeled heterogeneity in data. Supports ≤4 true species states and observation states without covariates, supports ≤3 with one covariate, and supports 2 with two covariates. Without covariates, more states are possible; input to function 'theta4.f' specifies the maximum number of states.

MOM/SOM model simulations: Un-modeled heterogeneity (covariates)

Conducts "covariate" simulations for MOM and SOM models *with* un-modeled heterogeneity in data arising from covariates predicting true species probabilities ***or*** classification probabilities. Supports ≤3 true species states and observation states.

MOM/SOM model simulations: Un-modeled heterogeneity (observers)
Conducts "distinct observer" simulations for MOM and SOM models *with* un-modeled heterogeneity in data arising from differing classification probabilities among secondary observers. Supports ≤3 true species states and observation states; does not support covariates.

Multinomial model simulations: Estimating observed species proportions
For MOM models in 'omnibus" simulations, estimates observed species proportions using a multinomial logit model assuming no species misidentification. Supports ≤4 true species states and observation states; does not support covariates or SOM models. Output file includes field 'observed_p' with value of 1.

`Likelihood_equations.R`

Computing probabilities for observed groups with sizes >1, especially for heterogeneous groups, can be computationally intensive. File `likelihood_equations.R` contains R code for generating list objects 'likelihood.equations' containing pre-computed values describing structure of equations for calculating probabilities in a computationally efficient form. Lists are specific to the number of observation states *A*, true species states *B*, and a range of group sizes *g*.

Saved list objects provided in DataS2-Input_Files_for_Simulations follow the naming convention 'likelihood_equations_A[#]B[#]G[#].RData', where [#] denotes an integer for *A*, *B*, or maximum group size (with group sizes of 1 to the maximum).

Lists contain 3 types of named elements:

1) 'true.combinations.g.[size]', where [size] is an integer 1 to maximum group size

   An integer matrix for each group size *g* containing possible combinations of true species states *b* on separate rows, with columns denoting state(s) for each individual.

2) 'true.permutations.count.g.[size]', where [size] is an integer 1 to maximum group size

   An integer vector for each group size *g* with (*via* the multinomial coefficient) counts of possible permutations for each true group. Vector elements correspond to rows describing true groups for the same group size in (1) above.

3) ‘observed.[*observed group*]’, where [*observed group*] denotes counts for individuals in each observation state 1 to A (separated by decimals).

   Data frame summarizing equations giving the conditional probability of an observed group, with columns:

   a) ‘index’

      Integer corresponding to rows for true groups of the same size in (1) above.

   b) ‘coefficient’

      Integer to be multiplied by the equation terms in (c) below to account for the count for possible permutations of observation states giving rise to the classifications in (c.).

   c) ‘1.1’, ‘2.1’, . . .

      Integers in each of *A* * *B* columns named ‘*a.b*’ give counts of individuals in an observed group belonging to true species state *b* classified to observation state *a*.

   Rows in each data frame in (3) correspond to the possible permutations of observation states possibly giving rise to the true group specified by ‘index’. Each integer in an ‘*a.b*’ column specifies a count of individuals of true species state *b* classified to observation state *a*, with the integer value giving the exponent for the corresponding classification probability. All probability terms specified in each row are multiplied by ‘coefficient’. Rows with the same ‘index’ value are summed, and the overall likelihood for the observed group is the product of these summed probabilities across all index values. These data frames are a computationally efficient method of organizing equations for classification probabilities, where are illustrated in the main article in Table 2 under the ‘Observed group **y**’ columns. For an observed group, matrix rows correspond to each group of multiplied terms within a table cell, with the ‘coefficient’ multiplied by the classification probabilities and integers in columns ‘*a.b*’ representing exponents. Data frame rows with identical ‘index’ values are summed terms within a table cell, while groups of rows with different ‘index’ values represent terms in each table cell within a column.

Workflow for generating 'likelihood.equations' list objects

Time to produce these list files depends on user-specified inputs and computing power, but increases rapidly with higher group sizes.

1) In an R editor or integrated R development environment, open the R script file `likelihood_equations.R`.
2) Install and load packages in the "Required R packages" section.
3) In the following section, execute R code to register the 'doSNOW' parallel execution backend.
4) Execute the code in the "Required supporting functions" section.
5) In "User-specified inputs", specify the true species states, observation states, and range of group sizes.
6) Execute all code in the "Generate a list" section.

   Depending on your computer specs and user inputs, you may have time to prepare: tea, dinner, your dissertation, etc.

7) After inspection, rename the list to 'likelihood.equations" and save to the hard drive.

Literature Cited

Byrd, R. H., Lu, P., Nocedal, J. and Zhu, C. 1995. A limited memory algorithm for bound constrained optimization. SIAM Journal on Scientific Computing. 16:1190–1208.

R Development Core Team. 2020. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. www.r-project.org.

Table 1. Naming convention and description of fields in the data frame 'survey_data' containing simulated survey data. Suffixes (in brackets, separated by underscores) for fields starting with 'y_' denote observer identities and observation states *a*.

Field name	Field values	Description
y_[observer]_[*a*]	Integer	Counts of murrelets classified by survey observers to each observation state. Text suffix [observer] denotes primary observers 'p' or individual secondary observers 's1', 's2', and 's3', and the integer suffix [*a*] denotes observation state *a*.
covariate_theta	Numeric	Group-level covariate predicting classification probabilities
covariate_psi	Numeric	Group-level covariate predicting true species probabilities
group_size	Integer	Count of murrelets in a group
id	1 ≤ Integer ≤ # of simulations	Identifier for data from each simulation repetition.
count	Integer	Count of identical observation histories
y_[observer]_key	Integer	Key identifying unique observed groups for each observer. Text suffix [observer] denotes primary observers 'p' or individual secondary observers 's1', 's2', and 's3'.

Table 2. Naming conventions and description of fields in the data frame 'observation_key' containing unique key values for unique observed groups. For fields starting with "A", the suffix (in brackets, separated by an underscore) denotes observation state *a*.

Field name	Field values	Description
A_[*a*]	Integer	Count of individuals classified to the observation state specified by the integer suffix [*a*].
key	Integer	Unique key identifying the unique observed group.
B_states	Integer	Number of possible combinations of true groups for the observed group.
sum	Integer	Count of classifications in the observed group.

Table 3. Prefixes describing summary statistics for estimated parameters and derived parameters in simulation output. See MetadataS2.pdf for parameter descriptions and naming conventions.

Prefix	Description
m.	Mean
sd.	Standard deviation
se.	Mean standard error
cv.	Coefficient of variation (SE/mean)
me.	Mean error
rm.	Root mean squared error
ci.	95% confidence interval coverage
ci.lg.	95% confidence interval coverage for derived parameters, with confidence intervals estimated in logit scale using estimated standard error (see Appendix S2 in the companion article)
se.bt.	Mean standard error estimated from bootstrap resamples
ci.btlg.	95% confidence interval coverage, with confidence interval estimated in logit scale using the bootstrap estimate of standard error