Package ‘mads’

October 13, 2022

Depends R (>= 3.5.0), mrds, stats

Suggests testthat

Type Package

Title Multi-Analysis Distance Sampling

Version 0.1.6

Author Laura Marshall

Maintainer Laura Marshall `<lhm@st-and.ac.uk>`

Description Performs distance sampling analyses on a number of species at once and can account for unidentified sightings, model uncertainty and covariate uncertainty. Unidentified sightings refer to sightings which cannot be allocated to a single species but may instead be allocated to a group of species. The abundance of each unidentified group is estimated and then prorated to the species estimates. Model uncertainty should be incorporated when multiple models give equally good fit to the data but lead to large differences in estimated density / abundance. Covariate uncertainty should be incorporated when covariates cannot be measured accurately, for example this is often the case for group size in marine mammal surveys. Variance estimation for these methods is via a non parametric bootstrap. The methods implemented are described in Gerodette T. and Forcada J. (2005) <doi:10.3354/meps291001> Non-recovery of two spotted and spinner dolphin populations in the eastern tropical Pacific Ocean.

BugReports `https://github.com/DistanceDevelopment/mads/issues`

License GPL (>= 2)

Collate 'accumulate.results.R' 'calculate.dht.R' 'check.bootstrap.options.r' 'check.convergence.R' 'check.fitted.R' 'check.covar.uncertainty.R' 'check.ddf.models.R' 'check.species.code.definitions.R' 'check.species.presence.r' 'create.obs.table.R' 'create.param.arrays.r' 'create.result.arrays.R' 'execute.multi.analysis.R' 'fit.ddf.models.R' 'format.dht.results.R' 'get.datasets.R' 'glossary.R' 'mae.warning.R' 'model.description.R' 'object.description.R'

'process.bootstrap.results.R' 'process.warnings.R'
'prorate.unidentified.R' 'renumber.duplicates.R'
'resample.covariates.R' 'resample.data.R' 'rtpois.R'
'store.param.ests.R' 'summary.ma.allspecies.R'
'summary.ma.allunid.R' 'summary.ma.analysis.R' 'summary.ma.R'
'summary.ma.species.R' 'summary.ma.unid.R' 'mads-package.R'

RoxygenNote 7.1.0

Encoding UTF-8

NeedsCompilation no

Repository CRAN

Date/Publication 2020-05-27 13:10:02 UTC

R topics documented:

`mads-package` *Multi-Analysis Distance Sampling (mads)*

Description

Can perform distance sampling analyses on a number of species at once and can account for unidentified sightings. Unidentified sightings refer to sightings which cannot be allocated to a single species but may instead be allocated to a group of species. The abundance of each unidentified group is estimated and then prorated to the species estimates. The multi-analysis engine can also incorporate model and covariate uncertainty. Variance estimation is via a non parametric bootstrap. The methods implemented are described in Gerodette T. and Forcada J. (2005) <10.3354/meps291001> "Non-recovery of two spotted and spinner dolphin populations in the eastern tropical Pacific Ocean".

Details

The main function in this package is execute.multi.analysis.

Further information on distance sampling methods and example code is available at `http://distancesampling.org/R/`.

We are also in the process of setting up a new area of the website for vignettes / example code at `http://examples.distancesampling.org`.

For help with distance sampling and this package, there is a Google Group `https://groups.google.com/forum/#!forum/distance-sampling`.

Author(s)

Laura Marshall <lhm@st-and.ac.uk>

`accumulate.results` *Enters the prorated results into the bootstrap.results array*

Description

Enters the prorated results into the bootstrap.results array

Usage

```
accumulate.results(n, bootstrap.results, formatted.results, clusters)
```

Arguments

`n`	index of the current bootstrap iteration
`bootstrap.results`	list of 4-dimensional arrays containing the bootstrap results
`formatted.results`	list of data objects similar to the dht class
`clusters`	boolean are the observations clusters of individuals bootstrap results

Value

list of 4-dimensional arrays containing the updated

Author(s)

Laura Marshall

`calculate.dht` *Calculates the abundance for each species code including the unidentified codes if supplied.*

Description

Calculates the abundance for each species code including the unidentified codes if supplied.

Usage

```
calculate.dht(
  species.name,
  species.field.name,
  model.index,
  ddf.results,
  region.table,
  sample.table,
  obs.table,
  dht.options
)
```

Arguments

`species.name`	character vector of species codes
`species.field.name`	character vector giving the field name of the ddf data that contains the species codes
`model.index`	named character vector which acts as a look up table for duplicate detection function models
`ddf.results`	a list of ddf objects
`region.table`	dataframe of region records - Region.Label and Area
`sample.table`	dataframe of sample records - Region.Label, Sample.Label, Effort
`obs.table`	dataframe of observation records with fields object, Region.Label, and Sample.Label which give links to sample.table, region.table and the data records used in `model`
`dht.options`	a list of the options to be supplied to mrds::dht

Value

a list of dht objects, one for each species code

Author(s)

Laura Marshall

See Also

`mrds::dht`

`check.convergence` *Checks whether the model has converged*

Description

Checks whether the model has converged

Usage

```
check.convergence(ddf.model)
```

Arguments

`ddf.model`	ddf object

Value

boolean

Author(s)

Laura Marshall

`check.fitted` *Checks whether the model's fitted values make sense*

Description

Checks whether the model's fitted values make sense

Usage

```
check.fitted(ddf.model)
```

Arguments

`ddf.model`	ddf object

Value

boolean

Author(s)

Laura Marshall

`create.obs.table` *Creates a subsetted observation table*

Description

Subsets the obs.table dataframe supplied to only contain the observations of interest.

Usage

```
create.obs.table(obs.table, ddf.data, subset.variable, subset.value)
```

Arguments

`obs.table`	dataframe of observation records with fields object, Region.Label, and Sample.Label which give links to sample.table, region.table and the data records used in `model`
`ddf.data`	dataframe containing the observations
`subset.variable`	variable name supplied as a character
`subset.value`	character value on which to subset the data

Value

dataframe containing the subset of the obs.table

Note

Internal function not intended to be called by user.

Author(s)

Laura Marshall

`create.param.arrays` *Creates a list of arrays for storing the ddf results*

Description

Creates a list of arrays. These are used to store the parameter estimates, a record of convergence, selection criteria values and which model was selected in the case of model uncertainty.

Usage

```
create.param.arrays(model.names, ddf.models, n, criteria)
```

Arguments

`model.names`	a list of character vectors of model names with the elements named by species code
`ddf.models`	a list of all the ddf models named in model.names.
`n`	the number of bootstrap iterations to be completed.
`criteria`	the name of the model selection criteria.

Value

list of arrays

Note

Internal function not intended to be called by user.

Author(s)

Laura Marshall

`create.result.arrays` *Creates a list of arrays for storing the dht results*

Description

Creates a list of arrays. These are used to store the summary, abundance and density outputs of the `dht` routine called from `mrds`.

Usage

```
create.result.arrays(
  species.name,
  species.code.definitions,
  region.table,
  clusters,
  n
)
```

Arguments

`species.name`	a list of all the species in the analysis
`species.code.definitions`	a list with an element for each unidentified code which contains a vector of corresponding identified species codes or NULL if not required
`region.table`	dataframe of region records - Region.Label and Area
`clusters`	boolean, TRUE if observations are of cluster, FALSE if observations are of individuals.
`n`	the number of bootstrap iterations to be completed.

Value

list of arrays

Note

Internal function not intended to be called by user.

Author(s)

Laura Marshall

`execute.multi.analysis`

Performs Multiple Analyses on Distance Data

Description

Analyses are performed for multiple species contained within the same dataset. Individual detection function analyses of each species must have already been completed using the `ddf` function in the `mrds` library. This function may then perform additional tasks such as assessing variance via a non-parametric bootstrap, including covariate variability via a parametric bootstrap, including model uncertainty and dealing with species codes which relate to unidentified sightings.

Usage

```
execute.multi.analysis(
  species.code,
  unidentified.sightings = NULL,
  species.presence = NULL,
  covariate.uncertainty = NULL,
  models.by.species.code,
  ddf.model.objects,
  ddf.model.options = list(criterion = "AIC", species.field.name = "species"),
  region.table,
  sample.table,
  obs.table,
  dht.options = list(convert.units = 1),
  bootstrap,
  bootstrap.options = list(resample = "samples", n = 1, quantile.type = 7),
  silent = FALSE
)
```

Arguments

`species.code`	vector of all the species codes to be included in the analysis
`unidentified.sightings`	a list with an element for each unidentified code which contains a vector of corresponding identified species codes or NULL if not required
`species.presence`	must be specified if species.code.definitions is specified. A list with an element for each strata which contains the vector of species codes present in that strata
`covariate.uncertainty`	a dataframe detailing the variables to be resampled - variable.layer, variable.name, cor.factor.layer, cor.factor.name , uncertainty.layer, uncertainty.name, uncertainty.measure, sampling.distribution. or NULL if not required
`models.by.species.code`	a list of character vectors of model names with the elements named by species code

`ddf.model.objects`	a list of all the ddf models named in models.by.species.code
`ddf.model.options`	a list of options 1) selection.criterion either "AIC", "AICc" or "BIC" 2) species.field.name describing the field name in the ddf dataset containing species codes.
`region.table`	dataframe of region records - Region.Label and Area
`sample.table`	dataframe of sample records - Region.Label, Sample.Label, Effort
`obs.table`	dataframe of observation records with fields object, Region.Label, and Sample.Label which give links to sample.table, region.table and the data records used in `model`
`dht.options`	list containing option for dht: convert.units indicated if the distance measurement units are different from shapefile and transect coordinate units.
`bootstrap`	if TRUE resamples data to obtain variance estimate
`bootstrap.options`	a list of options that can be set 1) n: number of repetitions 2) resample: how to resample data ("samples", "observations")
`silent`	boolean used to suppress progress counter output

Details

This is a new package with limited testing on real data, please drop me a line if you plan on using it (lhm[at]st-and.ac.uk).

The model fitting code in this function obtains its data and the model descriptions from the ddf objects passed in via the `ddf.models` argument.

If you wish to include model uncertainty then each model which you wish to be included in the analyses must have already been run and should be provided in the `ddf.models` argument. The `model.names` argument tells this function which `"ddf"` objects are associated with which species code in the dataset. This object must be constructed as a list of vectors. Each element in the list must be named corresponding to one of the species codes in the dataset and contain a character vector of object names.

For the majority of analyses the variance will be estimated using a non-parametric bootstrap, indicated by the `bootstrap` argument. You may select options for the bootstrap using the `bootstrap.options` argument. This is a list with elements specifying the number of repetitions and whether to resample samples within strata (`$resample = "samples"`) or observations withing strata (`$resample = "observations"`). In addition, the `bootstrap.covariates` is a boolean argument specifying whether or not a parametric bootstrap should be performed on any of the covariates. The details of which variables should be resampled and from which distributions should be entered in the `covariate.uncertainty` dataframe. This dataframe should contain 7 columns with the following names: `variable.layer`, `variable.name`, `cor.factor.layer`, `cor.factor.name`, `uncertainty.layer`, `uncertainty.name`, `uncertainty.measure` and `sampling.distribution`. [Currently this is only implemented for the observation layer]. The `variable.name` and `uncertainty.name` should be the names of the variable in the dataset giving the covariate to be resampled and the variable containing the uncertainty respectively. The `cor.factor.layer` specifies the data layer which contains the correction factor variable, although alternatively "numeric" can be entered. The `cor.factor.name` specifies the name of the correction factor variable or the correction factor value if "numeric" was specified for the correction factor layer. The `uncertainty.name` should specify what values the uncertainty variable contains and should be one of `"sd"`, `"var"` or `"CV"`. The `sampling.distribution` should specify one of the following distributions to parametrically resample from `"Normal"`, `"Normal.Absolute"`, `"Lognormal.BC"`, `"Poisson"` or `"TruncPoissonBC"`. The remaining column in this dataset, `variable.correction.factos` allows the user to specify a value by which the variable should be scaled. If this is not required this should be set to 1.

If there are unidentified sightings in the dataset then the `unidentified.sightings` argument should be `true` and a `species.code.definitions` list should be provided. This list must contain one element for every unidentified species code which should be named according to this code. Each element will contain a vector of identified species codes corresponding to those species which the unidentified code could have potentially been. This function uses this information to prorate the abundance estimated from the unidentified species codes to the relevant abundances from the identified codes. The prorating is done individually for each strata. The function can be forced not to prorate to any given species in any selected strata using the `species.presence` argument. This is a list containing one element for each strata, each must be named using the appropriate strata name. Each element should contain a vector of identified species codes corresponding to which species are present in each strata.

Value

object of class "ma" which consists of a list of objects of class "ma.element". Each "ma.element" consists of the following elements:

`individuals`	Summary, N (abundance) and D (density) tables
`clusters`	Summary, N (abundance) and D (density) tables
`Expected.S`	Expected cluster size table
`ddf`	Model details including a summary of convergence and selection as well as parameter estimates for selected models.

Author(s)

Laura Marshall

References

Marques, F.F.C. and S.T. Buckland. 2004. Covariate models for the detection function. In: Advanced Distance Sampling, eds. S.T. Buckland, D.R.Anderson, K.P. Burnham, J.L. Laake, D.L. Borchers, and L. Thomas. Oxford University Press. Gerodette, T. and Forcada, J. 2005 Non-recovery of two spotted and spinner dolphin populations in the eastern tropical Pacific Ocean. Marine Ecology Progress Series, 291:1-21.

Examples

```
#Load the example data
data("mads.data")
ddf.data <- mads.data$dist.data
region.table <- mads.data$region.table
sample.table <- mads.data$sample.table
```

```
obs.table <- mads.data$obs.table

# Fit candidate detection function models using ddf in mrds
# Fit a half normal model
df.all.hn <- ddf(dsmodel = ~mcds(key = "hn", formula = ~ 1),
                 method='ds', data=ddf.data, meta.data=list(width=1))
summary(df.all.hn)
plot(df.all.hn)
# Fit a hazard rate model
df.all.hr <- ddf(dsmodel = ~mcds(key = "hn", formula = ~ 1),
                 method='ds', data=ddf.data, meta.data=list(width=1))
summary(df.all.hr)
plot(df.all.hr)

# Set up mads data:

# A vector of the species names
species.codes <- c("CD", "WSD", "Unid")

# A list defining which species the unidentified categories could be
unid.defs <- list("Unid" = c("CD", "WSD"))

# Specify which models are to be tried for each species code
mod.uncert <- list("CD" = c("df.all.hn", "df.all.hr"),
                   "WSD" = c("df.all.hn", "df.all.hr"),
                   "Unid" = c("df.all.hn", "df.all.hr"))

# Provide the models in a named list and the selection criteria
models <- list("df.all.hn" = df.all.hn,
               "df.all.hr" = df.all.hr)
model.opts <- list(criterion = "AIC")


# Bootstrap options
bootstrap.opts <- list(resample = 'samples', n=999)

#Warning this will take some time to run!
results<- execute.multi.analysis( species.code = species.codes,
                                  unidentified.sightings = unid.defs,
                                  models.by.species.code = mod.uncert,
                                  ddf.model.objects = models,
                                  ddf.model.options = model.opts,
                                  region.table = region.table,
                                  sample.table = sample.table,
                                  obs.table = obs.table,
                                  bootstrap = TRUE,
                                  bootstrap.option = bootstrap.opts)


#Short example to run as per CRAN requirements -
#   warning only 1 repetition, results not interpretable!
bootstrap.opts <- list(resample = 'samples', n=1)
```

```
results<- execute.multi.analysis( species.code = species.codes,
                                  unidentified.sightings = unid.defs,
                                  models.by.species.code = mod.uncert,
                                  ddf.model.objects = models,
                                  ddf.model.options = model.opts,
                                  region.table = region.table,
                                  sample.table = sample.table,
                                  obs.table = obs.table,
                                  bootstrap = TRUE,
                                  bootstrap.option = bootstrap.opts)

#These are simulated data and true abundances are:
#  CD (common dolphins) = 3000
#  WSD (white sided dolphins) =  1500

summary(results)
```

`fit.ddf.models` *Refits the detection functions to the resampled data*

Description

Fits all the models named in model.names to the associated data supplied in ddf.dat.working. If more than one model is supplied for any species the model with the minimum selection criteria will be selected.

Usage

```
fit.ddf.models(
  ddf.dat.working,
  model.names,
  ddf.models,
  criterion,
  bootstrap.ddf.statistics,
  rep.no,
  MAE.warnings
)
```

Arguments

`ddf.dat.working`	list of dataframes containing the data to which the models will be fitted
`model.names`	list of unique character vectors giving the names of the ddf objects for each species.
`ddf.models`	a list of ddf objects
`criterion`	character option specifying the model selection criteria - "AIC", "AICc" or "BIC".

`bootstrap.ddf.statistics`	array storing parameter estimates
`rep.no`	numeric value indicating iteration number
`MAE.warnings`	character vector of warning messages

Value

list of ddf objects

Note

Internal function not intended to be called by user.

Author(s)

Laura Marshall

`format.dht.results` *Formats the estimated abundances of all species categories, to be consistent with the prorated results.*

Description

Formats the estimated abundances of all species categories, to be consistent with the prorated results.

Usage

```
## S3 method for class 'dht.results'
format(dht.results, species.name, clusters)
```

Arguments

`dht.results`	a list of objects of class dht
`species.name`	a character vectors detailing the species codes
`clusters`	boolean whether observations are clusters of individuals

Value

a list of results with an element for each species

Note

Internal function not intended to be called by user.

Author(s)

Laura Marshall

`mads.data` *Example simulated data used to demonstrate the package functionality*

Description

These data were generated using DSsim. Two populations were generated inside a rectangular study region, one of these is called ’CD’ (common dolphin) and the other is ’WSD’ (white-sided dolphin). Density was assumed to be equal across the study area and the population sizes for the CD and WSD populations were 3000 and 1500, respectively. Detections of individuals were simulated based on half normal detection functions with a scale parameter of 0.5 and a truncation distance of 1. A systematic parallel line transect design was used. Once both sets of data had been generated they were combined and 10 randomly selected to be in the unidentified sightings category.

Format

This is a list of 4 items. The first is dist.data a dataframe with distance sampling data including the columns object, transect.ID, distance, x, y, true.species, unid, species, observer. The other items are the region, sample and observations tables as per the definitions in mrds.

`mae.warning` *Warning function*

Description

Writes or stores messages for various situations that can occur

Usage

```
mae.warning(warning.msg = NULL, warning.mode = "store", MAE.warnings)
```

Arguments

`warning.msg`	the message to be stored/printed (optional)
`warning.mode`	report or print errors (default report)
`MAE.warnings`	character vector of existing warning messages

Value

None

Author(s)

Dave Miller & Laura Marshall

`model.description` *Extracts the model description*

Description

Returns a description of the model fitted in the ddf object.

Usage

```
model.description(model)
```

Arguments

`model`	a ddf object

Value

mod.str a string describing the fitted model

Author(s)

Jeff Laake & Laura Marshall

`process.bootstrap.results` *Summarises the bootstrap results.*

Description

Creates summary statistics for each species. These consist of dataframes relating to summaries, abundance (N) and density (D) for both individuals and clusters. In addition, summary statistics for expected cluster size (Expected.S) are also calculated.

Usage

```
process.bootstrap.results(
  bootstrap.results,
  model.index,
  clusters,
  bootstrap.ddf.statistics,
  quantile.type,
  analysis.options = list(bootstrap, n, covariate.uncertainty, clusters,
   double.observer, unidentified.species, species.code.definitions, model.names)
)
```

Arguments

`bootstrap.results`	list of arrays containing results from the repeated analyses.
`model.index`	named character vector which acts as a look up table for duplicate detection function models.
`clusters`	boolean whether observations are clusters of individuals
`bootstrap.ddf.statistics`	array storing parameter estimates from ddf models
`quantile.type`	numeric value describing which quantile algorithm to use
`analysis.options`	list describing the type of analysis carried out

Value

ma object a list of summary statistics for each species

Note

Internal functions not intended to be called by user.

Author(s)

Laura Marshall

`process.warnings` *Summarises warnings*

Description

Summarises warnings generated during the bootstrap and removes the MAE.warnings global object.

Usage

```
process.warnings(MAE.warnings)
```

Arguments

`MAE.warnings`	character vector of warning messages

Note

Internal function not intended to be called by user.

Author(s)

Laura Marshall

`prorate.unidentified` *Prorate the estimated abundances of the unidentified sightings to the other identified species categories.*

Description

The prorating is done individually for each strata. It will prorate the unidentified abundance between the species as defined in the species.code.definitions except where specified that a given species is not present in that strata as defined in the species.presence argument.

Usage

```
prorate.unidentified(
  dht.results,
  species.code.definitions,
  species.presence,
  clusters
)
```

Arguments

`dht.results`	a list of objects of class dht
`species.code.definitions`	a list of character vectors detailing the species codes associated with the unidentified code given as the element name.
`species.presence`	a list of character vectors defining the species present in each strata.
`clusters`	boolean whether observations are clusters of individuals of identified species codes corresponding to which species are present in each strata.

Value

a list of pro-rated results with an element for each species

Note

Internal function not intended to be called by user.

Author(s)

Laura Marshall

`rtpois` *Randomly generates values from a zero-truncated Poisson distribution*

Description

Generates values from a zero-truncated Poisson distribution with mean equal to that specified. It uses a look up table to check which value of lambda will give values with the requested mean.

Usage

```
rtpois(N, mean = NA)
```

Arguments

`N`	number of values to randomly generate
`mean`	mean of the generated values

Note

Internal function not intended to be called by user.

Author(s)

Laura Marshall

`summary.ma` *Summary of multi-analysis object*

Description

Provides a summary of the fitted detection probability model parameters, model selection criterion, and optionally abundance in the covered (sampled) region and its standard error for all species.

Usage

```
## S3 method for class 'ma'
summary(object, description = FALSE, glossary = FALSE, ...)
```

Arguments

`object`	an object of class `ma`
`description`	boolean if you would like
`glossary`	a `ma` model object would like a glossary to be displayed
`...`	unspecified and unused arguments for S3 consistency

Value

list of extracted and summarized objects

Note

This function is called by the generic function `summary` for any `ma` object.

Author(s)

Laura Marshall

`summary.ma.allspecies` *Summary of multi-analysis object*

Description

Provides a summary of the fitted detection probability model parameters, model selection criterion, and optionally abundance in the covered (sampled) region and its standard error for all species.

Usage

```
## S3 method for class 'ma.allspecies'
summary(object, ...)
```

Arguments

`object`	a `ma` model object
`...`	unspecified and unused arguments for S3 consistency

Value

list of extracted and summarized objects

Note

This function is called by the generic function `summary` for any `ma` object.

Author(s)

Laura Marshall

`summary.ma.allunid` *Summary of multi-analysis object*

Description

Provides a summary of the fitted detection probability model parameters, model selection criterion, and optionally abundance in the covered (sampled) region and its standard error for all species.

Usage

```
## S3 method for class 'ma.allunid'
summary(object, ...)
```

Arguments

`object`	a `ma` model object
`...`	unspecified and unused arguments for S3 consistency

Value

list of extracted and summarized objects

Note

This function is called by the generic function `summary` for any `ma` object.

Author(s)

Laura Marshall

`summary.ma.analysis` *Summary of multi-analysis object*

Description

Provides a summary of the fitted detection probability model parameters, model selection criterion, and optionally abundance in the covered (sampled) region and its standard error for all species.

Usage

```
## S3 method for class 'ma.analysis'
summary(object, ...)
```

Arguments

`object`	a `ma` model object
`...`	unspecified and unused arguments for S3 consistency

Value

list of extracted and summarized objects

Note

This function is called by the generic function `summary` for any `ma` object.

Author(s)

Laura Marshall

`summary.ma.species` *Print a summary of an element of a multi-analysis result corresponding to a single species included in the analyses.*

Description

Provides a summary of the fitted detection probability model parameters, model selection criterion, and optionally abundance in the covered (sampled) region and its standard error. What is printed depends on the corresponding call to summary.

Usage

```
## S3 method for class 'ma.species'
summary(object, species = NULL, ...)
```

Arguments

`object`	a summary of `ma` model object
`species`	optional character value giving the species name, solely for display purposes
`...`	unspecified and unused arguments for S3 consistency

Author(s)

Laura Marshall

See Also

`summary.ma`

`summary.ma.unid` *Summary of multi-analysis object*

Description

Provides a summary of the fitted detection probability model parameters, model selection criterion, and optionally abundance in the covered (sampled) region and its standard error for all species.

Usage

```
## S3 method for class 'ma.unid'
summary(object, species = NULL, ...)
```

Arguments

`object`	an object of class `ma.unid`
`species`	optional character value giving the species name, solely for display purposes
`...`	unspecified and unused arguments for S3 consistency

Value

list of extracted and summarized objects

Note

This function is called by the generic function `summary` for any `ma` object.

Author(s)

Laura Marshall

Index