Package 'pom'

February 20, 2015

Type Package

Title POM - Patch Occupancy Models

Version 1.1

Date 2012-05-19

Author Fawn Hornsby, Ryan Nielson, and Trent McDonald (www.west-inc.com)

Maintainer Fawn Hornsby <fhornsby@west-inc.com>

Depends matrixcalc

Description This package fits a patch occupancy model

License GNU General Public License

NeedsCompilation no

Repository CRAN

Date/Publication 2013-05-20 00:33:39

R topics documented:

pom-package *POM - Patch Occupancy Models*

Description

The package implements models to analyze site occupancy data with detection error.

Details

Package:	pom
Type:	Package
Version:	1.0
Date:	2012-11-17
License:	GNU General Public License
LazyData:	yes

List of routines:

siteocc	Routine for Estimating the Patch Occupancy Model.
print	Printing Patch Occupancy Model Fits.
F.2nd.deriv	Generic Routine for calculating the hessian matrix of a function.

Author(s)

Fawn Hornsby, Ryan Nielson, and Trent McDonald www.west-inc.com

Maintainer: Fawn Hornsby <fhornsby@west-inc.com>

References

Casella, G., and Berger, R. L. (2002). Statistical inference, Duxbury Press.

MacKenzie, D.I., Nichols, J.D., Lachman, G. B., Droege, S., Royle, J. A., and Langtimm, C. A., 2002. Estimating site occupancy rates when detection probabilities are less than one. Ecology, 83:2248-2255.

MacKenzie, D.I., Nichols, J.D., Royle, J.A., and Pollock, K.H. (2006), Occupancy Estimation and Modeling: Inferring Patterns and Dynamics of Species Occurrence, Academic Press, Burlington, MA.

Royle, J.A., 2006. Site occupancy models with heterogeneous detection probabilities. Biometrics 62:97-102.

F.2nd.deriv *Calculate the Hessian Matrix*

Description

Compute numeric 2nd derivative (Hessian) of the function FUN()

Usage

```
F.2nd.deriv(pt, FUN, ...)
```

Arguments

pt	Vector input to FUN.
FUN	Name of a function to evaluate. Must return a scalar when FUN(pt) is called.
...	Optional Items to FUN()

Details

This function is sourced by the siteocc function.

Value

Returns the Hessian Matrix of the input function.

Author(s)

Fawn Hornsby, Ryan Nielson, and Trent McDonald www.west-inc.com

Maintainer: Fawn Hornsby <fhornsby@west-inc.com>

References

Casella, G., and Berger, R. L. (2002). Statistical inference, Duxbury Press.

See Also

weta.data
siteocc

Examples

```
F.test <- function(x){
    3*x[1]^3 + x[1]^2*x[2]^2 + x[2]^3 + x[1]^2*x[3]^4
}
F.2nd.deriv( c(2,3,4), F.test )

# Another Example using siteocc function to get standard errors.
data(weta.data)

fit1 <- siteocc(~1, ~1, histories=weta.data$detection.histories)
print(fit1)
```

print.mixed.pom *Printing Patch Occupancy Model Fits For a Beta-binomial Mixture*

Description

This function summarizes the results from class `mixed.pom`. This function is called when a "~Beta.mixture" is specified as the model for p (probability of detection).

Usage

```
## S3 method for class 'mixed.pom'
print( x,  digits = max(3, getOption("digits") - 3), ...)
```

Arguments

`x`	an object of class `mixed.pom`; usually, a result of a call to `siteocc`.
`digits`	the number of significant digits to use when printing.
`...`	further arguments passed to or from other methods. They are ignored in this function.

Value

The output includes the matched call, the convergence code, PSI Coefficients with Estimates, Standard Errors, Z-values, and 2-sided p-values, Beta mixture parameters for P, AIC, BIC, the average estimated Probability of Occupancy (PSI).

Author(s)

Fawn Hornsby, Ryan Nielson, and Trent McDonald www.west-inc.com

Maintainer: Fawn Hornsby <fhornsby@west-inc.com>

See Also

`weta.data`
`siteocc`

Examples

```
data(weta.data)

fit1 <- siteocc(~1, ~1, histories=weta.data$detection.histories)
print(fit1)
```

print.pom *Printing Patch Occupancy Model Fits*

Description

This function summarizes the results from class pom. This function is called when covariate(s) or the intercept only model is specified as the model for p (probability of detection).

Usage

```
## S3 method for class 'pom'
print( x,  digits = max(3, getOption("digits") - 3), ...)
```

Arguments

x	an object of class pom; usually, a result of a call to siteocc.
digits	the number of significant digits to use when printing.
...	further arguments passed to or from other methods. They are ignored in this function.

Value

The output includes the matched call, the convergence code, PSI Coefficients with Estimates, Standard Errors, Z-values, and 2-sided p-values, P Coefficients also with Estimates, Standard Errors, and t-values, AIC, BIC, the average estimated Probability of Occupancy (PSI) and the average estimated Probability of Detection (P).

Author(s)

Fawn Hornsby, Ryan Nielson, and Trent McDonald www.west-inc.com

Maintainer: Fawn Hornsby <fhornsby@west-inc.com>

See Also

weta.data
siteocc

Examples

```
data(weta.data)

fit1 <- siteocc(~1, ~1, histories=weta.data$detection.histories)
print(fit1)
```

`siteocc` *Fitting Patch Occupancy Models*

Description

`siteocc` will fit a patch occupancy model to histories of observations and can account for an imperfect probability of detection.

Usage

```
siteocc(psi, p, histories, start=NULL, lower=NULL, ...)
```

Arguments

`psi`	an object of class `formula` (or one that can be coerced to that class): model formula for the site occupancy parameters. Each covariate called should be a vector of size nsites x 1.
`p`	an object of class `formula` (or one that can be coerced to that class): model formula for probability of detection given presence parameters. Each covariate called should be a matrix of size nsites x nvisits. To fit a beta-binomial mixture to probability of detection, specify "~ Beta.mixture" (case-sensitive) as the model for p. This argument specification assumes p's come from a beta-binomial mixture.
`histories`	matrix of encounter histories. One row per site, one column per visit. This argument should be of size nsites x nvisits. A NA may be used in the `histories` dataframe to represent instances when a survey could not be completed at a particular site for a particular visit.
`start`	vector of starting values passed to the `nlminb` function. If `start=NULL`, then the default starting values will be used. See details.
`lower`	vector of lower bounds passed to the `nlminb` function. If `lower=NULL`, then the default lower bounds will be used. See details.
`...`	other arguments passed to the `nlminb` function. See `?nlminb`.

Details

The log-likelihood is maximized using the `nlminb` function. If the user decides to provide starting values, then they will need to specify values for each PSI covariate as well as each P covariate, including intercepts. The starting values should be listed in the order of the specified covariates (i.e., the PSI intercept starting value first, the PSI coefficient(s) starting value(s) next, then the P intercept starting value, and finally the P Coefficient(s) starting value(s)). See below for an example of format. If specifying a Beta-mixture model, then the starting values are 0.5 for all of the covariates including the intercept. Otherwise, 0 is used for the starting value of each covariate.

The same format used to specify starting values should also be used to specify the lower bounds for the `lower` argument. The default lower bounds for a Beta-mixture are -Inf for all PSI covariates and 0 for both of the beta-binomial parameters. If you specify a lower bound of less than 0 for the beta-binomial parameters, your model will most likely not converge. This is because both parameters

must be strictly greater than 0. If a Beta-mixture is not specified, then -Inf will be the lower bound for all covariates.

Unless otherwise specified, the default values of the `nlminb` function are used.

The example datasets are detailed in pages 116-122 of MacKenzie et al. (2006) and also included with the program PRESENCE.

Value

`loglik`	Optimized log-likelihood.
`convergence`	An integer code. 0 indicates successful convergence. See the Value section of `?nlminb`.
`convergence.message`	A character string giving any additional information returned by the optimizer, or NULL. See the Value section of `?nlminb`.
`call`	The matched call.
`naive.psi.est`	Naive Estimate of Occupancy.
`nsites`	Number of Sites.
`nvisits`	Number of Visits.
`psi.coefs`	Esimate(s) of psi coefficient(s).
`p.coefs`	Estimate(s) of p coefficient(s) or the shape parameters if a Beta-Binomial mixture model was specified.
`se.psi.coefs`	Standard Error of the psi coefficient(s).
`se.p.coefs`	Standard Error of the p coefficient(s).
`hessian`	Hessian matrix used to compute the standard error of the psi and p coefficient(s).
`psi.ests`	Psi Estimates corresponding to each site.
`p.ests`	Matrix of P Estimates corresponding to each site and visit.
`aic`	Akaike's information criterion.
`bic`	Bayesian information criterion.

Warning

Be sure to check for convergence. Some tips if you are having trouble getting your models to converge:
1. Choose different starting values.
2. Make sure your covariates are on similar scales.
3. Check for high correlations among covariates.

Note

Currently, this function does not fit visit-specific P-covariates.

Author(s)

Fawn Hornsby, Ryan Nielson, and Trent McDonald www.west-inc.com

Maintainer: Fawn Hornsby <fhornsby@west-inc.com>

References

Royle, J.A., 2006. Site occupancy models with heterogeneous detection probabilities. Biometrics 62:97-102.

MacKenzie, D.I., Nichols, J.D., Lachman, G. B., Droege, S., Royle, J. A., and Langtimm, C. A., 2002. Estimating site occupancy rates when detection probabilities are less than one. Ecology, 83:2248-2255.

MacKenzie, D.I., Nichols, J.D., Royle, J.A., and Pollock, K.H. (2006), Occupancy Estimation and Modeling: Inferring Patterns and Dynamics of Species Occurrence, Academic Press, Burlington, MA.

See Also

weta.data
print.pom
print.mixed.pom
F.2nd.deriv

Examples

```
data(weta.data)

# INTERCEPT-ONLY MODEL
fit1 <- siteocc(~1, ~1, histories=weta.data$detection.histories, start=c(1,1))

# BETA-BINOMIAL MIXTURE MODEL
fit2 <- siteocc(~1, ~Beta.mixture, histories=weta.data$detection.histories,
lower=c(log(0.01),0.0001,0.0001))

# MODEL TESTING BROWSED AND OBSERVER EFFECTS
fit3 <- siteocc(~weta.data$siteCovar$Browsed, ~weta.data$Obs1 + weta.data$Obs2,
histories=weta.data$detection.histories, start=c(0,2,0,-1,0), control=list(iter.max=50))

# MODEL WHICH ALSO FITS A SITE COVARIATE TO THE PROBABILITY OF DETECTION
numvisits=5
p.Browse <- matrix(rep(weta.data$siteCovar$Browsed, numvisits), ncol=numvisits)
fit4 <- siteocc(~1, ~p.Browse + weta.data$Obs1 + weta.data$Obs2,
histories=weta.data$detection.histories)
```

---

weta.data *Occupancy data for the New Zealand Mahoenui Giant Weta*

Description

A list of datasets used in modeling the Mahoenui Giant Weta probability of occupancy. 72 sites were surveyed with up to 5 visits completed for each site. This list also includes covariates to model an observer effect for three observers and whether a site was browsed.

Usage

```
data(weta.data)
```

Format

A list containing 5 datasets:

$ detection.histories: A data frame with 72 observations (one row for each site) on the following 5 variables (one column for each visit to a particular site). A 1 indicates that at least one weta was observed and 0 indicates that no weta were observed on a particular visit. A NA indicates the site was not surveyed for that particular visit.

$ siteCovar : A data frame used to define the effect of browsing.
..$ Browsed : a vector of 1's and 0's, with 1 indicating that a site was browsed and 0 indicating the site was unbrowsed.
..$ Unbrowsed: a vector of 1's and 0's, with 1 indicating that a site was unbrowsed and 0 indicating the site was browsed.

$ Obs1 : Observer 1 Covariate matrix. Each row refers to a specific site and each column refers to one of five total visits. Thus, this matrix is of size 72x5. A 1 indicates that Observer 1 conducted the survey for a particular visit, while a 0 indicates that Observer 1 did not conduct the survey for a particular visit. A NA indicates the site was not surveyed by any observer for that particular visit.

$ Obs2 : Observer 2 Covariate matrix. Each row refers to a specific site and each column refers to one of five total visits. Thus, this matrix is of size 72x5. A 1 indicates that Observer 2 conducted the survey for a particular visit, while a 0 indicates that Observer 2 did not conduct the survey for a particular visit. A NA indicates the site was not surveyed by any observer for that particular visit.

$ Obs3 : Observer 3 Covariate matrix. Each row refers to a specific site and each column refers to one of five total visits. Thus, this matrix is of size 72x5. A 1 indicates that Observer 3 conducted the survey for a particular visit, while a 0 indicates that Observer 3 did not conduct the survey for a particular visit. A NA indicates the site was not surveyed by any observer for that particular visit.

Details

The example datasets are detailed in pages 116-122 of MacKenzie et al. (2006) and also included with the program PRESENCE.

Coding was used to define an observer effect using Obs1 and Obs2 as factor covariates (see table 4.6 on page 118 of MacKenzie et al., 2006). `Obs3` has been included in this package for completeness.

Source

MacKenzie, D.I., Nichols, J.D., Royle, J.A., and Pollock, K.H. (2006), Occupancy Estimation and Modeling: Inferring Patterns and Dynamics of Species Occurrence, Academic Press, Burlington, MA.

See Also

`siteocc`

Examples

```
data(weta.data)

fit1 <- siteocc(~weta.data$siteCovar$Browsed, ~weta.data$Obs1 + weta.data$Obs2,
         histories=weta.data$detection.histories)
print(fit1)
```

Index