Package ‘shinyrecap’

January 20, 2019

Type    Package
Title   Shiny User Interface for Multiple Source Capture Recapture Models
Version 0.1.0
Author  Ian E. Fellows
Maintainer Ian E. Fellows <ian@fellstat.com>
Description Implements user interfaces for log-linear models, Bayesian model averaging and Bayesian Dirichlet process mixture models.
License MIT + file LICENCE
Imports Rcapture, shiny, shinycssloaders, conting, ggplot2, reshape, CARE1, dga, LCMCR, ipc, future, promises, coda, testthat
URL    https://fellstat.github.io/shinyrecap/
Encoding UTF-8
LazyData true
RoxygenNote 6.1.0
NeedsCompilation no
Repository CRAN
Date/Publication 2019-01-19 23:40:03 UTC

R topics documented:

    shinyrecap-package ............................................. 2
disaggregate ...................................................... 2
formatGraphs ...................................................... 2
launchShinyPopSize .............................................. 3
lcmcrSample ....................................................... 3
simulateCapture ................................................... 4
simulateEstimates ............................................... 4
simulateHeteroNormal ........................................... 5

Index  6
**shinyrecap-package**  
*Shiny User Interface for Multiple Source Capture Recapture Models*

**Description**

Implements user interfaces for log-linear models, bayesian model averaging and bayesian dirichlet process mixture models.

**Author(s)**

Ian E. Fellows <ian@fellstat.com>

**disaggregate**  
*disaggregate data*

**Description**

disaggregate data

**Usage**

`disaggregate(dat, counts)`

**Arguments**

dat  
a data.frame
counts  
frequency counts for each row

**formatGraphs**  
*Format graphs*

**Description**

Format graphs

**Usage**

`formatGraphs(graphs)`

**Arguments**

graphs  
the graphs
launchShinyPopSize

Description
Launches the Shiny Application for Population Size

Usage
launchShinyPopSize(app = c("estimation", "power"))

Arguments
app Which application to launch.

Details
The manual for this shiny application is located at https://fellstat.github.io/shinyrecap/

lcmcrSample

Description
Perform LCMCR sampling with a monitor function

Usage
lcmcrSample(object, burnin = 10000, samples = 1000, thinning = 10,
clear_buffer = FALSE, output = TRUE, nMonitorBreaks = 100,
monitorFunc = function(subs, tot) { })

Arguments
object the samples
burnin MCMC burn in
samples number of samples
thinning MCMC thinning
clear_buffer buffer clear buffer of object
output output progress
nMonitorBreaks number of times to call the monitor function
monitorFunc A function called nMonitorBreaks times taking the number of samples to be taken, and the total samples

Details
An edited version of lcmCR_PostSamp1
simulateCapture Simulate Capture Re-capture with heterogeneity

Description
Simulate Capture Re-capture with heterogeneity

Usage
simulateCapture(hetero, p)

Arguments
hetero The heterogeneity
p A vector of capture event probabilities

Examples
het <- simulateHeteroNormal(1000, 1.1)
cap <- simulateCapture(het, p = c(.05, .1, .05, .1))
summary(cap)

simulateEstimates Simulates capture re-capture estimates

Description
Simulates capture re-capture estimates

Usage
simulateEstimates(nsim, N, p, htype = "None", heteroPerc = 1,
monitorFunc = function(i) { })

Arguments
nsim number of simulations
N Population size
p A vector of capture event probabilities
htype The type of capture heterogeneity. Either "None" or "Normal"
heteroPerc The increase in odds of capture for the perc 90th percentile most likely to be captured individuals, compared to the average individual.
monitorFunc A function called after every iteration. Useful for monitoring simulation progress.
Examples

```r
library(ggplot2)

# Simulate estimates from the Mt model with no population heterogeneity
ests <- simulateEstimates(15, 500, c(.1, .1, .1))

# Simulate estimates from the Mth (Normal) model with no population heterogeneity.
estss2 <- simulateEstimates(20, 500, c(.1, .1, .1), htype="Normal")

df <- data.frame(est = ests[,1], type="Mt")
df <- rbind(df, data.frame(est = ests2[,1], type="Mth (Normal)"))
qplot(x=est, color=type, data=df, geom="density") +
  geom_vline(xintercept=500, color="purple")
```

---

### simulateHeteroNormal

simulate capture heterogeneity

#### Usage

```r
simulateHeteroNormal(N, heteroPerc = 1, perc = 0.9)
```

#### Arguments

- `N` Population size
- `heteroPerc` The increase in odds of capture for the perc 90th percentile most likely to be captured individuals, compared to the average individual.
- `perc` The percentile to use.

#### Examples

```r
het <- simulateHeteroNormal(100, 1.1)
hist(het)
```
Index

disaggregate, 2
formatGraphs, 2
launchShinyPopSize, 3
lcmcrSample, 3
shinyrecap-package, 2
simulateCapture, 4
simulateEstimates, 4
simulateHeteroNormal, 5