Package ‘idr’

February 15, 2013

Type Package

Title Irreproducible discovery rate

Version 1.1.1

Date 2011-06-08

Author Qunhua Li

Maintainer Qunhua Li <qunhua.li@gmail.com>

Description This is a package for estimating the copula mixture model
and plotting correspondence curves in “Measuring
reproducibility of high-throughput experiments” (2011), Annals

License LGPL (>= 2.0)

LazyLoad yes

Repository CRAN

Date/Publication 2012-10-27 06:50:41

NeedsCompilation no

R topics documented:

  idr-package ...................................................... 2
  est.IDR ......................................................... 3
  get.correspondence ........................................... 4
  salmon ......................................................... 6
  select.IDR ..................................................... 7
  simu.idr ....................................................... 8

Index 10
Description

This package estimates the reproducibility of observations on a pair of replicate rank lists. It consists of three components: (1) plotting the correspondence curve to visualize reproducibility, (2) quantifying reproducibility using a copula mixture model and estimating the posterior probability for each observation to be irreproducible (local irreproducible discovery rate), and (3) ranking and selecting observations by their irreproducibility.

Details

Package: idr
Type: Package
Version: 1.1.1
Date: 2012-10-26
License: GPL-2
LazyLoad: yes

The main functions are est.IDR(), get.correspondence() and select.IDR(). est.IDR estimates the copula mixture model and the posterior probability for each observation to be irreproducible. get.correspondence generates the values for plotting the correspondence curve. select.IDR ranks observations by their reproducibility and reports the number of observations passing the specified IDR thresholds.

Author(s)

Qunhua Li
Maintainer: Qunhua Li <qunhua.li@gmail.com>

References


Examples

data("simu.idr")
x <- cbind(simu.idr$x, simu.idr$y)

mu <- 2.6
sigma <- 1.3
rho <- 0.8
p <- 0.7
est.IDR

idr.out <- est.IDR(x, mu, sigma, rho, p, eps=0.001)

names(idr.out)

---

est.IDR   

Estimate the irreproducible discovery rate using the copula mixture model

Description

Fit a Gaussian copula mixture model.

Usage

est.IDR(x, mu, sigma, rho, p, eps, max.ite=30)

Arguments

x          a m by n numeric matrix, where m= num of replicates, n=num of observations. Numerical values representing the significance of the observations, where signals are expected to have large values, for example, -log(p-value). Currently, m=2.
mu          a starting value for the mean of the reproducible component.
sigma       a starting value for the standard deviation of the reproducible component.
rho         a starting value for the correlation coefficient of the reproducible component.
p           a starting value for the proportion of reproducible component.
eps          Stopping criterion. Interations stop when the increment of log-likelihood is < eps, default=0.001.
max.ite       Maximum number of iterations. Default=30.

Value

para          estimated parameters: p, rho, mu, sigma.
idr           a numeric vector of the local idr for each observation (i.e. estimated conditional probability for each observation to belong to the irreproducible component.
IDR           a numerical vector of the expected irreproducible discovery rate for observations that are as irreproducible or more irreproducible than the given observations.
loglik        log-likelihood at the end of iterations.
loglik.trace  trajectory of log-likelihood.

Author(s)

Qunhua Li
References


Examples

data("simu.idr")

# simu.idr$x and simu.idr$y are p-values
# Transfer them such that large values represent significant ones
x <- cbind(-simu.idr$x, -simu.idr$y)

mu <- 2.6
sigma <- 1.3
rho <- 0.8
p <- 0.7

idr.out <- est.IDR(x, mu, sigma, rho, p, eps=0.01, max.ite=20)
names(idr.out)

get.correspondence

Compute correspondence profiles

Description

Compute the correspondence profiles (\(\Psi\) and \(\Psi'\)) and the corresponding smoothed curve using spline

Usage

get.correspondence(x1, x2, t, spline.df = NULL)

Arguments

x1  Data values or ranks of the data values on list 1, a vector of numeric values. Large values need to be significant signals. If small values represent significant signals, rank the signals reversely (e.g. by ranking negative values) and use the rank as x1.

x2  Data values or ranks of the data values on list 2, a vector of numeric values. Large values need to be significant signals. If small values represent significant signals, rank the signals reversely (e.g. by ranking negative values) and use the rank as x1.

t  A numeric vector between 0 and 1 in ascending order. t is the right-tail percentage.

spline.df  Degree of freedom for spline, to control the smoothness of the smoothed curve.
get.correspondence

**Value**

<table>
<thead>
<tr>
<th>psi</th>
<th>the correspondence profile in terms of the scale of percentage, i.e. between (0, 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>dpsipl</td>
<td>the derivative of the correspondence profile in terms of the scale of percentage, i.e. between (0, 1)</td>
</tr>
<tr>
<td>psi.pl</td>
<td>the correspondence profile in terms of the scale of the number of observations</td>
</tr>
<tr>
<td>dpsipl</td>
<td>the derivative of the correspondence profile in terms of the scale of the number of observations</td>
</tr>
</tbody>
</table>

Each object above is a list consisting of the following items: t: upper percentage (for psi and dpsipl) or number of top ranked observations (for psi.pl and dpsipl) value: psi or dpsipl smoothed.line: smoothing spline ntotal: the number of observations jump.point: the index of the vector of t such that psi(t[jump.point]) jumps up due to ties at the low values. This only happens when data consists of a large number of discrete values, e.g. values imputed for observations appearing on only one replicate.

**Author(s)**

Qunhua Li

**References**


**Examples**

```r
# salmon data
data(salmon)

# get.correspondence() needs the observations with high ranks have
# high correlation and the observations with low ranks have low correlation.
# In this dataset, small values have high correlation and large values
# have low correlation.
# Ranking negative values makes the data follow the structure required
# by get.correspondence().
# There are 28 observations in this data set.
rank.x <- rank(-salmon$spawners)
rank.y <- rank(-salmon$recruits)
uv <- get.correspondence(rank.x, rank.y, seq(0, .99, by=1/28))

# plot correspondence curve on the scale of percentage
plot(uv$psi$t, uv$psi$value, xlab="t", ylab="psi", xlim=c(0, max(uv$psi$t)),
     ylim=c(0, max(uv$psi$value)), cex.lab=2)
lines(uv$psi$smoothed.line, lwd=4)
abline(coef=c(0,1), lty=3)
```
# plot the derivative of correspondence curve on the scale of percentage
plot(uv$dpsi$t, uv$dpsi$value, xlab="t", ylab="psi'", xlim=c(0, max(uv$dpsi$t)), ylim=c(0, max(uv$dpsi$value)), cex.lab=2)
lines(uv$dpsi$smoothed.line, lwd=4)
abline(h=1, lty=3)

# plot correspondence curve on the scale of the number of observations
plot(uv$psi.n$t, uv$psi.n$value, xlab="t", ylab="psi", xlim=c(0, max(uv$psi.n$t)), ylim=c(0, max(uv$psi.n$value)), cex.lab=2)
lines(uv$psi.n$smoothed.line, lwd=4)
abline(coef=c(0,1), lty=3)

# plot the derivative of correspondence curve on the scale of the number
# of observations
plot(uv$dpsi.n$t, uv$dpsi.n$value, xlab="t", ylab="psi'", xlim=c(0, max(uv$dpsi.n$t)), ylim=c(0, max(uv$dpsi.n$value)), cex.lab=2)
lines(uv$dpsi.n$smoothed.line, lwd=4)
abline(h=1, lty=3)

# If the rank lists consist of a large number of ties at the bottom
# (e.g. the same low value is imputed to the list for the observations
# that appear on only one list), it may be desirable to plot only
# observations before hitting the ties. Then it can be plotted using the
# following
plot(uv$psi$t[1:uv$psi$jump.point], uv$psi$value[1:uv$psi$jump.point], xlab="t", ylab="psi", xlim=c(0, max(uv$psi$t[1:uv$psi$jump.point])), ylim=c(0, max(uv$psi$value[1:uv$psi$jump.point])), cex.lab=2)
lines(uv$psi$smoothed.line, lwd=4)
abline(coef=c(0,1), lty=3)

salmon

Salmon data

Description

This data is from Simonoff (1990, p 161). It concerns the size of the annual spawning stock and its production of new catchable-sized fish for 1940 through 1967 for the Skeena river sockeye salmon stock (in thousands of fish). It has three columns, year, spawners and recruits. It was speculated to consist of two different populations.

Usage

data(salmon)

Format

A data frame with 28 observations on the following 3 variables.

year  a numeric vector of the year
select.IDR

spawners a numeric vector of the annual spawning stock
recruits a numeric vector of the production of new catchable-sized fish

Source
Data is salmon.dat in Simonoff (1990). It can be downloaded from the book’s website.

References

Examples
data(salmon)
plot(rank(salmon$spawners), rank(salmon$recruits))

select.IDR Select observations according to IDR

Description
Select observations that exceeding a given IDR level

Usage
select.IDR(x, IDR.x, IDR.level)

Arguments
x a m by n numeric matrix, where m= num of replicates, n=num of observations. Numerical values representing the significance of the observations, where signals are expected to have large values, for example, -log(p-value). Currently, m=2.
IDR.x Irreproducibile discovery rate for each entry of x. It is computed from est.IDR().
IDR.level IDR cutoff, a numerical value between [0,1].

Value
x Observations that are selected.
n Number of observations that are selected.
IDR.level IDR cutoff, a numerical value between [0,1].

Author(s)
Qunhua Li
References

See Also
est.IDR

Examples

data("simu.idr")
x <- cbind(simu.idr$x, simu.idr$y)

mu <- 2.6
sigma <- 1.3
rho <- 0.8
p <- 0.7

idr.out <- est.IDR(x, mu, sigma, rho, p, eps=0.01, max.ite=20)
# select observations exceeding IDR threshold=0.01
IDR.level <- 0.01
x.selected <- select.IDR(x, idr.out$IDR, IDR.level)

Description
This is a simulated dataset for testing the program. Data is first simulated from the copula mixture model with latent structure of 0.65 N(mu, mu, sigma, sigma, rho) + 0.95 N(0, 0, 1, 1, 0), where mu=2.5, sigma=1, rho=0.84. The observations in the dataset are then generated by taking the p-values from a z-test H0: mu=0.

Usage

data(simu.idr)

Format
A data frame with 1000 observations on the following 3 variables.

x a numeric vector, representing p-values on replicate 1
y a numeric vector, representing p-values on replicate 2
labels a binary vector, where 1 represents the reproducible component and 0 represents the irreproducible component.
References

Examples
```r
data(simu.idr)
plot(rank(simu.idr$x), rank(simu.idr$y))
```
Index

*Topic Statistical Models
  est.IDR, 3
  select.IDR, 7
*Topic Statistical models
  get.correspondence, 4
*Topic datasets
  salmon, 6
  simu.idr, 8
*Topic package
  idr-package, 2

est.IDR, 3, 8

get.correspondence, 4

idr (idr-package), 2
idr-package, 2

salmon, 6
select.IDR, 7
simu.idr, 8