Package ‘prc’

January 23, 2019

LazyLoad yes
LazyData yes
Version 2019.1-23
Title Paired Response Curve
Author Youyi Fong
Maintainer Youyi Fong <youyifong@gmail.com>
Depends R (>= 3.3.0), kyotil, nlme
Suggests RUnit, Rmosek, Matrix

Imports

Description Estimation, prediction and testing for analyzing serial dilution assay data (Fong and Yu 2015 <DOI:10.1080/19466315.2015.1093019>) using paired response curve, which is a model of the nonlinear functional relationship between experimental outcomes measured at two different dilutions.

License GPL (>= 2)
NeedsCompilation yes
Repository CRAN
Date/Publication 2019-01-23 18:40:02 UTC

R topics documented:

mtct.eg ............................................................ 2
prc .............................................................. 2

Index 7
Example Dataset

Description
from MTCT correlates study

Usage
data(mtctNeg)

Format
A data frame with 248 observations on the following 4 variables.

V3_BioV3B_500 a numeric vector
V3_BioV3B_2500 a numeric vector

References
Permar et al.

-four parameter paired response curve

Description
Estimate four-parameter paired response curve using both total least squares and semiparametric methods.

Usage
prc(xvar, dil.x, yvar, dil.y, model=c("4P","3P"), method=c("TLS","naive"),
init.method=c("gnls","optim"), opt.method=c("gnls","optim"), reltol=1e-3, max.iter=50,
init=NULL, verbose=FALSE)

prcsp (xvar, dil.x, yvar, dil.y, model=c("sp","struct"),
stop.when.dropping=FALSE, grid.density=200,
init=NULL, method.init=c("TLS","naive"), reltol=1e-3, max.iter=20,
try.additiona.support.sets=FALSE,
keep.history=FALSE, verbose=FALSE)

prcstruct(xvar, dil.x, yvar, dil.y,
grid.density=200, method.init=c("TLS","naive"), reltol=1e-3, max.iter=20, init=NULL,
keep.history=FALSE, verbose=FALSE)
## S3 method for class 'prc'
coef(object, ...)
## S3 method for class 'prc'
predict(object, new.dilution, xvar=NULL, dil.x=NULL, ret.sd=FALSE, ...)
## S3 method for class 'prc'
mixlik(object, ...)

## S3 method for class 'prc'
print(x, ...)
## S3 method for class 'prc'
plot(x, type=c("b","l","p"), add=FALSE, diag.line=TRUE, lcol=2, pcol=1,
     log.axis=TRUE, xlab=NULL, ylab=NULL, lwd=2, xlim=NULL, ylim=NULL, ...)
## S3 method for class 'prc'
lines(x, col=1, ...)

four_pl_prc(c,d,b,f, xx, k, call.C=TRUE)
s.dot.f(c,d,b,f, r, k)
mixlik (object, ...)

compute.A (logc,logd,b,f, dil.r, sigma.sq, support, xvar, yvar)

quadratic.eiv (xvar, yvar, init=NULL, reltol=1e-3, opt.method=c("optim"),
stop.after.init=FALSE, max.iter=50, verbose=FALSE)

quadratic.eiv.sp (xvar, yvar, grid.density=200, init=NULL, reltol=1e-3,
opt.method=c("optim"), max.iter=50, fix.sigma.sq=FALSE, verbose=FALSE)

## S3 method for class 'quad'
coef(object, ...)
## S3 method for class 'quad'
print(x, ...)
## S3 method for class 'quad'
plot(x, type=c("b","l"), add=FALSE, lcol=2, pcol=1,
     xlab=NULL, ylab=NULL, lwd=2, x.range=NULL, log.axis=TRUE,...)
## S3 method for class 'quad'
lines(x, col=1, x.range=NULL, ...)

quad.f(a,b,c,x)
Arguments

xvar vector of numbers on the ln(FI) scale
dil.x numeric.
yvar vector of numbers on the ln(FI) scale
dil.y numeric.
init.method glns uses glns to find initial parameter estimate and optim uses optim. In the initial estimation, measurement errors in the xvar are ignored.
reltol relative tolerance convergence criterion used in optimization
opt.method optimization method
method TLS minimizes total least squares. Naive minimizes least squares, ignoring measurement errors in the xvar.
model 4P is a four-parameter model proposed in Fong et al. 3P is a three-parameter model that assumes f=1 in 4P.
max.iter maximum number of iterations to perform in optimization. Each iteration comprises finding the best r given theta and finding the best theta given r
verbose Boolean. If TRUE, print messages during execution
object an object of type prc
new.dilution a new dilution
type "b" for both points and line, "l" for line only
add add to an existing plot or not
lcol line color
pcol points color
col color
diag.line whether to add a diagonal line
log.axis whether the axis should be on the log scale
xlab xlab
ylab ylab
lwd line width. Does not affect points
x object of class prc or quad depending on the functions
xx vector of numbers on the log scale
k dilution ratio: dilution of x axis/dilution of y axis, e.g. 2500/500
c parameter c in the four-parameter paired response model
d parameter d in the four-parameter paired response model
b parameter b in the four-parameter paired response model
f parameter f in the four-parameter paired response model
a parameter a in the quadratic model
r the x axis of a point on the prc on the log scale
init initial parameter value
ret.sd whether to return standard deviation
xlim xlim
ylim ylim
keep.history Boolean
stop.when.dropping Boolean
method.init string
logc log(c) in the four-parameter paired response model
logd log(d) in the four-parameter paired response model
dil.r dilution ratio
sigma.sq sigma square
support the support of the distribution
try.additiona.support.sets boolean
stop.after.init
  Boolean. If true (not recommended), a naive curve estimate which pretends there is no measurement error in xvar is returned. If false, a proper least squares estimate accounting for errors-in-variables is returned.
x.range a pair of low and high
grid.density number of grid points
fix.sigma.sq whether to assume simgasq is fixed
call.C Boolean
... more args

Details

prc() operates on the scale on which the distance is defined. For example, for MBA/Luminex readouts, this is the log(FI) scale.

In prcs, if verbose>=2, will make plots of nonparametric distribution

s.dot.f() is created with deriv3 and it returns prc function value, as well as gradient and hessian at the point (r,s(r))

quadratic.eiv and quadratic.eiv.sp deal with a quadratic model as an illustration

Value

Both prc and prcs return an object of type prc.

coefficients curve estimate
sigma.sq error variance estimate
dilution.ratio dilution.x/dilution.y
dilution.x sample dilution for the readouts plotted on the x axis
dilution.y sample dilution for the readouts plotted on the y axis
xvar observations on the x axis
yvar observations on the y axis
rvar closest point to (x,y) on the curve is (r,s(r))

Both quadratic.eiv and quadratic.eiv.sp return an object of type quad, which has a similar structure as type prc.
compute.A returns a matrix.

Author(s)
Youyi Fong

References

Examples

# estimation
fit=prc (mtct.eg$V3_BioV3B_2500, 2500, mtct.eg$V3_BioV3B_500, 500, verbose=TRUE)
plot(fit)

# prediction
logfi.1000 = predict(fit, new.dilution=1000)
points(exp(fit$rvar), exp(logfi.1000), col=2, cex=.5)

theta=coef(fit)
four_pl_prc(theta[“c”], theta[“d”], theta[“b”], theta[“f”], log(500), k=5)
Index

coeff.prc (prc), 2
coeff.quad (prc), 2
compute.λ (prc), 2

dfour.pl.prc (prc), 2

lines.prc (prc), 2
lines.quad (prc), 2

mixlik (prc), 2
mtct.eg, 2

plot.prc (prc), 2
plot.quad (prc), 2
prc, 2
prcs.p (prc), 2
prcstruct (prc), 2
predict.prc (prc), 2
print.prc (prc), 2
print.quad (prc), 2

quad.f (prc), 2
quadratic.eiv (prc), 2

s.dot.f (prc), 2