

Package ‘orloca.es’

February 14, 2012

Type Package

Depends orloca (>= 3.2)

Title Spanish version of orloca package

Version 3.2

Date 2010-05-22

Author Fernando Fernandez-Palacin <fernando.fernandez@uca.es> and
Manuel Munoz-Marquez <manuel.munoz@uca.es>

Maintainer Manuel Munoz-Marquez <manuel.munoz@uca.es>

Description Version espanola del paquete orloca que trata de los
modelos de Localizacion en Investigacion Operativa (Operations
Research LOCational Analysis). En particular, el problema de
encontrar el punto que minimiza la suma ponderada de las
distancias a un conjunto de puntos dados.

License GPL (>= 2)

URL <http://knuth.uca.es/orloca>

Repository CRAN

Date/Publication 2010-05-26 16:54:51

R topics documented:

orloca.es-package	2
as-methods	3
czsum	5
czsummin	5
loca.p-class	6
plot-methods	7
plot.zsum	8
rloca.p	9

zsum	11
zsuml2	11
zsuml2min	12
zsumlp	13
zsumlpmin	14
zsummin	15

Index	17
--------------	-----------

orloca.es-package	<i>Spanish version of orloca package - Version española del paquete orloca</i>
-------------------	--

Description

El paquete ofrece modelos de Análisis de Localización. Esta versión aborda problemas de localización min-sum. El problema min-sum, también conocido como problema de Weber, localiza un punto tal que la suma ponderada de las distancias entre dicho punto y los puntos de demanda sea mínima.

Details

Package: orloca.es
 Type: Package
 Version: 3.2
 Date: 2010-05-22
 License: GPL (>= 2)

El paquete proporciona una clase (`loca.p`) que representa un problema de localización con un conjunto finito de puntos de demanda sobre el plano. También es posible representar los puntos y la función objetivo. Dicha función objetivo representa la suma de los desplazamientos de los usuarios a un servicio.

Para una demostración, cargue el paquete con `library(orloca.es)` y use `demo(orloca)`.

El paquete está preparado para su internacionalización. Las traducciones de los ficheros `.mo` recibidas serán añadidas en próximas versiones del paquete.

Index:

<code>loca.p</code> :	descripción de la clase <code>loca.p</code> .
<code>rloca.p</code> :	instancias aleatorias de objetos de la clase <code>loca.p</code> .
<code>zsum</code> :	función para evaluar la función objetivo para modelos min-sum.
<code>zsummin</code> :	para encontrar el mínimo del problema de localización.
<code>plot.loca.p</code> :	para hacer gráficos de los objetos de la clase <code>loca.p</code> .
<code>plot.zsum</code> :	para hacer gráficos de la función objetivo.

Author(s)

Fernando Fernandez-Palacin <fernando.fernandez@uca.es> and Manuel Munoz-Marquez <manuel.munoz@uca.es>
Maintainer: Manuel Munoz-Marquez <manuel.munoz@uca.es>

References

- [1] Love, R. F., Morris, J. G., Wesolowsky, G. O. *Facilities Location: Chapter 2: Introduction to Single-Facility Location*, 1988, North-Holland
- [2] <http://knuth.uca.es/orloca>

See Also

For the English version of the package see [orloca-package](#).

Examples

```
# Un objeto loca.p no ponderado
o <- new("loca.p", x = c(-1, 1, 1, -1), y = c(-1, -1, 1, 1))

# Calcula la función objetivo en el punto c(3, 4)
zsum(o, 3, 4)

# Resuelve el problema de localización
zsummin(o)

# Curvas de nivel
contour(o)

# Ejecuta una demo del paquete
demo(orloca)
```

as-methods

Conversion entre objetos de la clase loca.p y otros objetos

Description

Métodos para convertir desde y hacia la clase loca.p.

Usage

```
## S3 method for class 'data.frame'
as.loca.p(x, ...)
## S3 method for class 'matrix'
as.loca.p(x, ...)
```

```
## S3 method for class 'loca.p'  
as.data.frame(x, row.names = NULL, optional = FALSE, ...)  
## S3 method for class 'loca.p'  
as.matrix(x, ...)
```

Arguments

x	es el objeto para convertir a una nueva clase.
row.names	No usado.
optional	No usado.
...	Otros argumentos, no usados.

Details

Valores NA's no están permitidos en ninguno de los argumentos.

La `matrix` a convertir en `loca.p` debe tener al menos dos columnas. la primera columna será considerada como la coordenada x, la segunda como la coordenada y, y la tercera (si existe) como los valores de w.

El `data.frame` para convertir en `loca.p` debe tener al menos una columna x para la coordenada x, y una columna y para la coordenada y. Opcionalmente, puede tener una columna w, como los valores de w.

Value

Si los argumentos tienen valores válidos, devuelve un nuevo objeto de la nueva clase.

See Also

Ver también [loca.p](#).

Examples

```
# Un objeto loca.p no ponderado  
loca <- loca.p(x = c(-1, 1, 1, -1), y = c(-1, -1, 1, 1))  
  
# Conversión a matriz  
m <- as.matrix(loca)  
m  
  
# Conversión desde una matriz  
as.loca.p(m)
```

`czsum`*czsum* y *czsumgra* en el paquete *orloca*

Description

Funciones `zsum` y `zsumgra` calculadas en el clúster snow dado.

Usage

```
czsum(o, cluster, x=0, y=0)
czsumgra(o, cluster, x=0, y=0)
```

Arguments

<code>o</code>	Un objeto de la clase <code>loca.p</code> .
<code>cluster</code>	Un objeto snow cluster.
<code>x</code>	La coordenada x del punto para ser evaluada.
<code>y</code>	La coordenada y del punto para ser evaluada.

Value

`czsum` devuelve la función objetivo del problema de localización min-sum, $\sum_{a_i \in o} w_i d(a_i, (x, y))$, donde $d(a_i, (x, y))$ representa la distancia euclídea entre a_i y (x, y) .

`czsumgra` devuelve el vector gradiente de la función `zsum`.

El cálculo se realiza en el clúster snow dado.

See Also

Ver <http://www.stat.uiowa.edu/~luke/R/cluster/cluster.html> para información sobre computación en clusters.

Véase también [zsum](#), [zsumgra](#) y [zsummin](#)

`czsummin`*czsummin* en el paquete *orloca*

Description

Es la versión para clúster de `zsummin`.

Usage

```
czsummin(o, cluster, x=0, y=0, max.iter=100, eps=1.e-3,
         verbose=FALSE, algorithm="gradient")
```

Arguments

<code>o</code>	Un objeto de la clase <code>loca.p</code> .
<code>cluster</code>	Un objeto <code>snow cluster</code> .
<code>x</code>	La coordenada <code>x</code> del punto inicial.
<code>y</code>	La coordenada <code>y</code> del punto inicial.
<code>max.iter</code>	Máximo número de iteraciones permitido.
<code>eps</code>	El módulo del gradiente en la regla de parada.
<code>verbose</code>	Si es <code>TRUE</code> la función proporciona una salida detallada.
<code>algorithm</code>	El algoritmo que se usará. Los valores válidos son: "gradient" o "g" para un algoritmo de gradiente, y "search" o "s" para un algoritmo de búsqueda local. "gradient" es el valor por defecto.

Value

`czsummin` devuelve un vector con las coordenadas del punto solución.
Los cálculos se realizan en el cluster dado.

See Also

Ver <http://www.stat.uiowa.edu/~luke/R/cluster/cluster.html> para información sobre computación en clusters.

Ver también [zsummin](#), [loca.p](#) y [zsum](#).

`loca.p-class`

Clase de objetos `loca.p` para Localizacion en Investigacion Operativa

Description

Un objeto de la clase `loca.p` representa un problema de localización ponderado con un conjunto de puntos de demanda. El [orloca.es-package](#) está principalmente dedicado a abordar problemas de localización.

Details

Las longitudes de los vectores `x` e `y` deben ser iguales. La longitud de `w` debe ser igual a los anteriores o 0. Los valores NA no están permitidos en ninguno de los argumentos.

Value

Si los argumentos son valores válidos, devuelve un objeto de la clase `loca.p`, en caso contrario devuelve un error. `summary(x)` devuelve un resumen del objeto `x` de la clase `loca.p` y `print(x)` imprime un resumen del objeto `x` de la clase `loca.p`.

Generators

El principal generador es `loca.p(x, y, w = numeric(0), label = "")` o alternativamente `new("loca.p", x, y, w = numeric(0), label = "")`.

x es un vector con las coordenadas x de los puntos de demanda.

y es un vector con las coordenadas y de los puntos de demanda.

w es un vector de pesos de los puntos de demanda. Si **w** se omite entonces todos los pesos se consideran iguales a 1.

label Si se explicita, es la etiqueta del nuevo objeto.

See Also

Véase también [orloca.es-package](#).

Examples

```
# Un objeto loca.p no ponderado
loca <- loca.p(x = c(-1, 1, 1, -1), y = c(-1, -1, 1, 1))
# o
loca <- new("loca.p", x = c(-1, 1, 1, -1), y = c(-1, -1, 1, 1))

# Un ejemplo con pesos y nombre
locb <- new("loca.p", x = c(-1, 1, 1, -1), y = c(-1, -1, 1, 1),
  w = c(1, 2, 1, 2), label = "Caso Ponderado")
```

plot-methods

Grafico de un objeto de la clase loca.p

Description

Este método proporciona una representación gráfica de un objeto de la clase `loca.p`.

Usage

```
## S3 method for class 'loca.p'
plot(x, xlab="", ylab="", main="Plot of loca.p object", ...)
```

Arguments

<code>x</code>	El objeto <code>loca.p</code> a representar.
<code>xlab</code>	La etiqueta para el eje x.
<code>ylab</code>	La etiqueta para el eje y.
<code>main</code>	El título principal del gráfico.
<code>...</code>	Otras opciones gráficas.

Details

Gráfico de los puntos de demanda con límites de evaluación automáticos.

Value

La representación gráfica de los puntos de demanda.

See Also

Véase también [orloca.es-package](#), [loca.p](#) y [plot.zsum](#).

Examples

```
# Un objeto de la clase loca.p no ponderado
loca <- new("loca.p", x = c(-1, 1, 1, -1), y = c(-1, -1, 1, 1))

# El gráfico del objeto loca.p
plot(loca)
```

plot.zsum

Grafica de la funcion objetivo min-sum

Description

Proporciona dos representaciones gráficas de la función objetivo del problema min-sum (zsum).

Usage

```
## S3 method for class 'loca.p'
contour(x, lp=numeric(0), xmin=min(x@x), xmax=max(x@x),
        ymin=min(x@y), ymax=max(x@y), n=100, ...)
## S3 method for class 'loca.p'
persp(x, lp=numeric(0), xmin=min(x@x), xmax=max(x@x),
       ymin=min(x@y), ymax=max(x@y), n=100, ...)
```

Arguments

x	El objeto loca.p para calcular el objetivo.
lp	Si se proporciona, entonces se usa la norma l_p en vez de la euclídea.
xmin	El valor mínimo del eje x.
xmax	El valor máximo del eje x.
ymin	El valor mínimo del eje y.
ymax	El valor máximo del eje y.
n	El número de divisiones para la rejilla.
...	Otras opciones.

Details

Si $p < 1$ entonces l_p no es norma, por tanto, sólo $p \geq 1$ es válido.

Value

La función `contour` representa un gráfico de curvas de nivel y la función `persp` un gráfico 3D.

See Also

Véase también [orloca.es-package](#), [plot.loca.p](#) y [loca.p](#).

Examples

```
# Un objeto loca.p no ponderado
loca <- loca.p(x = c(-1, 1, 1, -1), y = c(-1, -1, 1, 1))

# El gráfico de curvas de nivel del objeto loca.p
contour(loca)

# El gráfico 3D del objeto loca.p
persp(loca)
```

rloca.p

Generador de instancias aleatorias de objetos de la clase loca.p

Description

Devuelve una instancia aleatoria de un objeto de la clase `loca.p` en una determinada región.

Usage

```
rloca.p(n, xmin=0, xmax=1, ymin=0, ymax=1, groups=numeric(0),
        xmin=xmin, xmax=xmax, ymin=ymin, ymax=ymax)
```

Arguments

n	El número de puntos de demanda.
xmin	Mínimo valor para la coordenada x de los puntos de demanda.
xmax	Máximo valor para la coordenada x de los puntos de demanda.
ymin	Mínimo valor para la coordenada y de los puntos de demanda.
ymax	Máximo valor para la coordenada y de los puntos de demanda.
groups	El número de grupos (de aproximadamente igual tamaño) o una lista con los tamaños de los grupos a generar.
xgmin	Mínimo valor para la coordenada x de los puntos de demanda respecto del punto referencia del grupo.

xgmax	Máximo valor para la coordenada x de los puntos de demanda respecto del punto referencia del grupo.
ygmin	Mínimo valor para la coordenada y de los puntos de demanda respecto del punto referencia del grupo.
ygmax	Máximo valor para la coordenada y de los puntos de demanda respecto del punto referencia del grupo.

Details

n debe ser al menos 1.

xmin debe ser menor o igual que xmax.

ymin debe ser menor o igual que ymax.

Cuando se suministra valor para groups los puntos se generan en dos fases, en la primera se genera un punto de referencia, en la segunda se genera un desplazamiento sobre dicho punto de referencia que se suma a éste.

Obsérvese que groups = 1 no es equivalente a groups = numeric(0), debido a que en el primer caso se genera un punto de referencia en la primera etapa.

Value

Si los argumentos son valores válidos, devuelve un nuevo objeto de la clase loca_p, en otro caso informa de un error.

See Also

Véase también [orloca-package](#) y [loca.p](#)

Examples

```
# Un objeto aleatorio loca.p en el cuadrado unidad con 5 puntos de demanda
rloca.p(5)

# En otra región
rloca.p(10, xmin=-2, xmax=2, ymin=-2, ymax=2)

# Cinco grupos
rloca.p(48, groups=5)

# Tres grupos de distinto tamaño
rloca.p(1, groups=c(10, 7, 2))
```

zsum *zsum* y *zsumgra* del paquete *orloca*

Description

La función objetivo y la función gradiente para el problema de localización min-sum. Para uso interno.

Usage

```
zsum(o, x=0, y=0, lp=numeric(0))
zsumgra(o, x=0, y=0, lp=numeric(0), partial=F)
```

Arguments

o	Un objeto de la clase <code>loca.p</code> .
x	La coordenada x del punto a ser evaluado.
y	La coordenada y del punto a ser evaluado.
lp	Si se proporciona, entonces se usa la norma l_p en vez de la euclídea.
partial	Si (x,y) es un punto de demanda <code>partial=T</code> significa que se ignore dicho punto para el cálculo del gradiente. Principalmente para uso interno.

Value

`zsum` devuelve la función objetivo para el problema de localización min-sum, $\sum_{a_i \in o} w_i d(a_i, (x, y))$, donde $d(a_i, (x, y))$ es la distancia euclídea o la distancia l_p entre a_i y (x, y) .

`zsumgra` devuelve el vector gradiente de la función `zsum`.

See Also

Véase también [orloca.es-package](#) y [zsummin](#).

zsuml2 *zsuml2* y *zsuml2gra* del paquete *orloca*

Description

Las funciones `zsum` y `zsumgra` para la norma euclídea (l_2).

Usage

```
zsuml2(o, x=0, y=0)
zsuml2gra(o, x=0, y=0, partial=F)
```

Arguments

o	Un objeto de la clase <code>loca.p</code> .
x	La coordenada x del punto a ser evaluado.
y	La coordenada y del punto a ser evaluado.
partial	Si (x,y) es un punto de demanda <code>partial=T</code> significa que se ignore dicho punto para el cálculo del gradiente. Principalmente para uso interno.

Value

`zsuml2` devuelve la función objetivo para el problema de localización min-sum, $\sum_{a_i \in O} w_i d(a_i, (x, y))$, donde $d(a_i, (x, y))$ es la distancia euclídea entre a_i y (x, y) .

`zsumgra` devuelve el vector gradiente de la función `zsum`.

See Also

Véase también [orloca.es-package](#), `zsum`, `zsumgra` y `zsummin`.

`zsuml2min`

zsuml2min en el paquete orloca

Description

La función `zsummin` para la norma euclídea (l_2).

Usage

```
zsuml2min(o, x=0, y=0, max.iter=100, eps=1.e-3, verbose=FALSE,
          algorithm="weiszfeld")
```

Arguments

o	Un objeto de la clase <code>loca.p</code> .
x	La coordenada x del punto inicial.
y	La coordenada y del punto inicial.
max.iter	Número máximo de iteraciones permitido.
eps	La norma del gradiente en la regla de parada.
verbose	Si es TRUE la función proporciona salida detallada.
algorithm	El algoritmo a utilizar. Los valores válidos son: "gradient" o "g" para un algoritmo de gradiente, "search" o "s" para un algoritmo de búsqueda local, "weiszfeld" o "w" para el algoritmo de Weiszfeld o cualquiera de los métodos válidos para la función <code>optim</code> , a saber, "Nelder-Mead", "BFGS", "CG", "L-BFGS-B", "SANN". "Weiszfeld" es el valor por defecto.

Value

zsummin devuelve un vector con las coordenadas del punto solución.

See Also

Vea también [orloca.es-package](#), [loca.p](#) y [zsum](#).

Examples

```
# Un objeto loca.p no ponderado
loca <- new("loca.p", x = c(-1, 1, 1, -1), y = c(-1, -1, 1, 1))

# Calcula el mínimo
sol<-zsummin(loca)

# Muestra los resultados
sol

# Evaluación de la función objetivo en el punto solución
zsum(loca, sol[1], sol[2])
```

zsumlp

zsumlp y zsumlpgra del paquete orloca

Description

Las funciones zsum y zsumgra con norma l_p .

Usage

```
zsumlp(o, x=0, y=0, p=2)
zsumlpgra(o, x=0, y=0, p=2, partial=F)
```

Arguments

o	An object of loca.p class.
x	The x coordinate of the point to be evaluated.
y	The y coordinate of the point to be evaluated.
p	The l_p norm to use.
partial	If (x,y) is a demand point partial=T means ignore such point to compute the gradient. This option is mainly for internal use.

Details

If $p < 1$ then l_p ara not a norm, so only $p \geq 1$ are valid values.

Value

zsumlp returns the objective function of the min-sum location problem with l_p norm, $\sum_{a_i \in o} w_i d(a_i, (x, y))$, where $d(a_i, (x, y))$ gives the distances between a_i and the point (x, y) using l_p norm.

zsumlpgra returns the gradient vector of the function zsumlp.

Note

Since l_2 norm is the Euclidean norm, when $p = 2$ zsumlp are equal to zsum, and zsumlpgra are equal to zsumgra. But the computations involved are greater for the firsts form.

See Also

Véase [zsum](#), [orloca.es-package](#) y [zsumlpmin](#).

zsumlpmin

zsumlpmin en el paquete orloca

Description

La función zsummin con norma l_p .

Usage

```
zsumlpmin(o, x=0, y=0, p=2, max.iter=100, eps=1.e-3,
          verbose=FALSE, algorithm="weiszfeld")
```

Arguments

o	Un objeto de la clase <code>loca.p</code> .
x	La coordenada x del punto inicial.
y	La coordenada y del punto inicial.
p	Valor de p para la norma l_p .
max.iter	Número máximo de iteraciones permitido.
eps	El módulo del gradiente para la regla de parada.
verbose	Si es TRUE la función proporciona salida detallada.
algorithm	El algoritmo a utilizar. Los valores válidos son: "gradient" o "g" para un algoritmo de gradiente, "search" o "s" para un algoritmo de búsqueda local, y "weiszfeld" o "w" para el algoritmo de Weiszfeld o cualquiera de los métodos válidos para la función optim, a saber, "Nelder-Mead", "BFGS", "CG", "L-BFGS-B", "SANN". "Weiszfeld" es el valor por defecto.

Details

Si $p < 1$ entonces l_p no es una norma, por tanto, sólo valores $p \geq 1$ son válidos.

Value

zsummin devuelve un vector con las coordenadas del punto solución.

Note

Dado que l_2 es la norma euclídea, para $p = 2$ zsumlpmin es equivalente a zsummin. Pero los cálculos involucrados son mayores en la primera forma.

See Also

See also [zsummin](#), [orloca.es-package](#), [loca.p](#) and [zsum](#).

zsummin

zsummin en el paquete orloca

Description

Resuelve el problema de localización min-sum para un objeto dado de la clase `loca.p`.

Usage

```
zsummin(o, x=0, y=0, max.iter=100, eps=1.e-3, verbose=FALSE,
        algorithm="weiszfeld")
```

Arguments

o	Un objeto de la clase <code>loca.p</code> .
x	La coordenada x del punto inicial.
y	La coordenada y del punto inicial.
max.iter	Número máximo de iteraciones permitido.
eps	La norma del gradiente en la regla de parada.
verbose	Si es TRUE la función proporciona salida detallada.
algorithm	El algoritmo a utilizar. Los valores válidos son: "gradient" o "g" para el método basado en el gradiente, "search" o "s" para método de busca local, y "weiszfeld" o "w" para el método de Weiszfeld o cualquiera de los métodos válidos para la función optim, a saber, "Nelder-Mead", "BFGS", "CG", "L-BFGS-B", "SANN". "Weiszfeld" es el valor por defecto.

Details

Si $p < 1$ entonces l_p no es una norma, por tanto, sólo $p \geq 1$ es válido.

Value

zsummin devuelve un vector con las coordenadas del punto solución.

See Also

Vea también [orloca.es-package](#), [loca.p](#) y [zsum](#).

Examples

```
# Un objeto loca.p no ponderado
loca <- new("loca.p", x = c(-1, 1, 1, -1), y = c(-1, -1, 1, 1))

# Calcula el mínimo
sol<-zsummin(loca)

# Muestra los resultados
sol

# Evaluación de la función objetivo en el punto solución
zsum(loca, sol[1], sol[2])
```

Index

- *Topic **classes**
 - as-methods, 3
 - czsum, 5
 - czsummin, 5
 - loca.p-class, 6
 - plot-methods, 7
 - plot.zsum, 8
 - zsum, 11
 - zsuml2, 11
 - zsuml2min, 12
 - zsumlp, 13
 - zsumlpmin, 14
 - zsummin, 15
- *Topic **datagen**
 - rloca.p, 9
- *Topic **hplot**
 - plot-methods, 7
 - plot.zsum, 8
- *Topic **methods**
 - as-methods, 3
- *Topic **optimize**
 - czsum, 5
 - czsummin, 5
 - loca.p-class, 6
 - orloca.es-package, 2
 - zsum, 11
 - zsuml2, 11
 - zsuml2min, 12
 - zsumlp, 13
 - zsumlpmin, 14
 - zsummin, 15
- *Topic **package**
 - orloca.es-package, 2
- as, loca.p-method (as-methods), 3
- as-methods, 3
- as.data.frame (as-methods), 3
- as.data.frame, loca.p-method (as-methods), 3
- as.data.frame.loca.p (as-methods), 3
- as.loca.p (as-methods), 3
- as.loca.p, data.frame-method (as-methods), 3
- as.loca.p, matrix-method (as-methods), 3
- as.loca.p.data.frame (as-methods), 3
- as.loca.p.matrix (as-methods), 3
- as.matrix (as-methods), 3
- as.matrix, loca.p-method (as-methods), 3
- as.matrix.loca.p (as-methods), 3
- contour, loca.p-method (plot.zsum), 8
- contour.loca.p (plot.zsum), 8
- czsum, 5
- czsum, loca.p-method (czsum), 5
- czsumgra (czsum), 5
- czsumgra, loca.p-method (czsum), 5
- czsummin, 5
- czsummin, loca.p-method (czsummin), 5
- initialize, loca.p-method (loca.p-class), 6
- loca.p, 2, 4, 6, 8–10, 13, 15, 16
- loca.p (loca.p-class), 6
- loca.p-class, 6
- orloca (orloca.es-package), 2
- orloca-package, 10
- orloca-package, 3
- orloca.es-package, 6–9, 11–16
- orloca.es-package, 2
- persp, loca.p-method (plot.zsum), 8
- persp.loca.p (plot.zsum), 8
- plot, loca.p-method (plot-methods), 7
- plot-methods, 7
- plot.loca.p, 2, 9
- plot.loca.p (plot-methods), 7
- plot.zsum, 2, 8, 8
- print, loca.p-method (loca.p-class), 6
- print-method (loca.p-class), 6

`print.loca.p`(`loca.p-class`), 6

`rloca.p`, 2, 9

`summary`, `loca.p-method` (`loca.p-class`), 6

`summary-method` (`loca.p-class`), 6

`summary.loca.p` (`loca.p-class`), 6

`zsum`, 2, 5, 6, 11, 12–16

`zsum`, `loca.p-method` (`zsum`), 11

`zsumgra`, 5, 12

`zsumgra` (`zsum`), 11

`zsumgra`, `loca.p-method` (`zsum`), 11

`zsuml2`, 11

`zsuml2`, `loca.p-method` (`zsuml2`), 11

`zsuml2gra` (`zsuml2`), 11

`zsuml2gra`, `loca.p-method` (`zsuml2`), 11

`zsuml2min`, 12

`zsuml2min`, `loca.p-method` (`zsuml2min`), 12

`zsumlp`, 13

`zsumlp`, `loca.p-method` (`zsumlp`), 13

`zsumlpgra` (`zsumlp`), 13

`zsumlpgra`, `loca.p-method` (`zsumlp`), 13

`zsumlpmin`, 14, 14

`zsumlpmin`, `loca.p-method` (`zsumlpmin`), 14

`zsummin`, 2, 5, 6, 11, 12, 15, 15

`zsummin`, `loca.p-method` (`zsummin`), 15