## NAME

CUTEST_csgr − CUTEst tool to evaluate constraints gradients and gradient of objective/Lagrangian function.

## SYNOPSIS

CALL CUTEST_csgr( data, status, n, m, X, Y, grlagf, nnzj, lj, J_val, J_var, J_fun )

## DESCRIPTION

The CUTEST_csgr subroutine evaluates the gradients of the general constraints and of either the objective function or the Lagrangian function $l(x,y) = f(x) + y^T c(x)$ corresponding to the problem decoded from a SIF file by the script *sifdecode* at the point $(x,y) =$ (X,Y). It also evaluates the Hessian matrix of the Lagrangian function at $(x,y)$. The gradients are stored in a sparse format.

The problem under consideration is to minimize or maximize an objective function $f(x)$ over all $x \in R^n$ subject to general equations $c_i(x) = 0$, $(i \in 1, \ldots, m_E)$, general inequalities $c_i^l(x) \le c_i(x) \le c_i^u(x)$, $(i \in m_E + 1, \ldots, m)$, and simple bounds $x^l \le x \le x^u$. The objective function is group-partially separable and all constraint functions are partially separable.

## ARGUMENTS

The arguments of CUTEST_csgr are as follows

**data** [inout] - CUTEST_data_type derived type
problem-specific private data,

**status** [out] - integer
the outputr status: 0 for a succesful call, 1 for an array allocation/deallocation error, 2 for an array bound error, 3 for an evaluation error,

**n** [in] - integer
the number of variables for the problem,

**m** [in] - integer
the total number of general constraints,

**X** [in] - real/double precision
an array which gives the current estimate of the solution of the problem,

**Y** [in] - real/double precision
an array which should give the Lagrange multipliers whenever grlagf is set .TRUE. but need not otherwise be set,

**grlagf** [in] - logical
a logical variable which should be set .TRUE. if the gradient of the Lagrangian function is required and .FALSE. if the gradient of the objective function is sought,

**nnzj** [out] - integer
the number of nonzeros in J_val,

**lj** [in] - integer
the actual declared dimensions of J_val, J_var and J_fun,

**J_val** [out] - real/double precision
an array which gives the values of the nonzeros of the gradients of the objective, or Lagrangian, and general constraint functions evaluated at X and Y. The i-th entry of J_val gives the value of the derivative with respect to variable J_var(i) of function J_fun(i),

**J_var** [out] - integer
an array whose i-th component is the index of the variable with respect to which J_val(i) is the derivative,

**J_fun** [out] - integer
an array whose i-th component is the index of the problem function whose value J_val(i) is the derivative. J_fun(i) = 0 indicates the objective function whenever grlagf is .FALSE. or the Lagrangian

function when grlagf is .TRUE., while J_fun(i) = j > 0 indicates the j-th general constraint function.

## AUTHORS

I. Bongartz, A.R. Conn, N.I.M. Gould, D. Orban and Ph.L. Toint

## SEE ALSO

*CUTEr (and SifDec): A Constrained and Unconstrained Testing Environment, revisited*,
N.I.M. Gould, D. Orban and Ph.L. Toint,
ACM TOMS, **29**:4, pp.373-394, 2003.

*CUTE: Constrained and Unconstrained Testing Environment*, I. Bongartz, A.R. Conn, N.I.M. Gould and Ph.L. Toint, TOMS, **21**:1, pp.123-160, 1995.

cutest_ugr(3M), cutest_cgr(3M), sifdecode(1).

## NAME

CUTEST_csgr − CUTEst tool to evaluate constraints gradients and gradient of objective/Lagrangian function.

## SYNOPSIS

CALL CUTEST_csgr( data, status, n, m, X, Y, grlagf, nnzj, lj, J_val, J_var, J_fun )

## DESCRIPTION

The CUTEST_csgr subroutine evaluates the gradients of the general constraints and of either the objective function or the Lagrangian function $l(x, y) = f(x) + y^T c(x)$ corresponding to the problem decoded from a SIF file by the script *sifdecode* at the point $(x, y) =$ (X,Y). It also evaluates the Hessian matrix of the Lagrangian function at $(x, y)$. The gradients are stored in a sparse format.

The problem under consideration is to minimize or maximize an objective function $f(x)$ over all $x \in R^n$ subject to general equations $c_i(x) = 0$, $(i \in 1, \ldots, m_E)$, general inequalities $c_i^l(x) \le c_i(x) \le c_i^u(x)$, $(i \in m_E + 1, \ldots, m)$, and simple bounds $x^l \le x \le x^u$. The objective function is group-partially separable and all constraint functions are partially separable.

## ARGUMENTS

The arguments of CUTEST_csgr are as follows

**data** [inout] - CUTEST_data_type derived type
problem-specific private data,

**status** [out] - integer
the outputr status: 0 for a succesful call, 1 for an array allocation/deallocation error, 2 for an array bound error, 3 for an evaluation error,

**n** [in] - integer
the number of variables for the problem,

**m** [in] - integer
the total number of general constraints,

**X** [in] - real/double precision
an array which gives the current estimate of the solution of the problem,

**Y** [in] - real/double precision
an array which should give the Lagrange multipliers whenever grlagf is set .TRUE. but need not otherwise be set,

**grlagf** [in] - logical
a logical variable which should be set .TRUE. if the gradient of the Lagrangian function is required and .FALSE. if the gradient of the objective function is sought,

**nnzj** [out] - integer
the number of nonzeros in J_val,

**lj** [in] - integer
the actual declared dimensions of J_val, J_var and J_fun,

**J_val** [out] - real/double precision
an array which gives the values of the nonzeros of the gradients of the objective, or Lagrangian, and general constraint functions evaluated at X and Y. The i-th entry of J_val gives the value of the derivative with respect to variable J_var(i) of function J_fun(i),

**J_var** [out] - integer
an array whose i-th component is the index of the variable with respect to which J_val(i) is the derivative,

**J_fun** [out] - integer
an array whose i-th component is the index of the problem function whose value J_val(i) is the derivative. J_fun(i) = 0 indicates the objective function whenever grlagf is .FALSE. or the Lagrangian

function when grlagf is .TRUE., while J_fun(i) = j > 0 indicates the j-th general constraint function.

## AUTHORS

I. Bongartz, A.R. Conn, N.I.M. Gould, D. Orban and Ph.L. Toint

## SEE ALSO

*CUTEr (and SifDec): A Constrained and Unconstrained Testing Environment, revisited*,
N.I.M. Gould, D. Orban and Ph.L. Toint,
ACM TOMS, **29**:4, pp.373-394, 2003.

*CUTE: Constrained and Unconstrained Testing Environment*, I. Bongartz, A.R. Conn, N.I.M. Gould and Ph.L. Toint, TOMS, **21**:1, pp.123-160, 1995.

cutest_ugr(3M), cutest_cgr(3M), sifdecode(1).