Package ‘modopt.matlab’

July 23, 2025

Type Package

Title 'MatLab'-Style Modeling of Optimization Problems

Version 1.0-2

Date 2018-08-17

Maintainer Ronald Hochreiter <ron@hochreiter.net>

Description 'MatLab'-Style Modeling of Optimization Problems with 'R'. This package provides a set of convenience functions to transform a 'MatLab'-style optimization modeling structure to its 'ROI' equivalent.

Depends R (>= 3.4), ROI, ROI.plugin.glpk, ROI.plugin.quadprog

License MIT + file LICENSE

URL http://www.finance-r.com/

RoxygenNote 6.1.0

NeedsCompilation no

Author Ronald Hochreiter [aut, cre]

Repository CRAN

Date/Publication 2018-08-18 16:00:03 UTC

Contents

modopt.matlab-package *MatLab(R)-style Optimization Modeling in R using ROI*

Description

'MatLab'-Style Modeling of Optimization Problems with 'R'. This package provides a set of convenience functions to transform a 'MatLab'-style optimization modeling structure to its 'ROI' equivalent.

Author(s)

Ronald Hochreiter, <ron@hochreiter.net>

References

http://www.finance-r.com/

See Also

Useful links:

- http://www.finance-r.com/

intlinprog *MatLab(R)-style Mixed Integer Linear Programming in R using ROI*

Description

`intlinprog` provides a simple interface to ROI using the optimization model specification of MatLab(R)

minimize in x: f'*x subject to A*x <= b Aeq*x == beq x >= lb x <= ub

Usage

```
intlinprog(f, intcon = NULL, A = NULL, b = NULL, Aeq = NULL,
  beq = NULL, lb = NULL, ub = NULL, x0 = NULL, options = NULL)
```

Arguments

`f`	Linear term (vector) of the objective function
`intcon`	Vector of which variables are integer
`A`	Inequality constraints (left-hand side)
`b`	Inequality constraints (right-hand side)
`Aeq`	Equality constraints (left-hand side)

beq	Equality constraints (right-hand side)
lb	Lower bound
ub	Upper bound
x0	Initial solution
options	Additional optimization parameters

Value

The solution vector in x as well as the objective value in fval.

Author(s)

Ronald Hochreiter, <ron@hochreiter.net>

Examples

```
# minimize 8x1 + x2
# subject to
#   x1 + 2x2 >= -14
#   -4x1 - 1x2 <= -33
#   2x1 + x2 <= 20
#   x1, x2 integer

f <- c(8, 1)
A <- matrix(c(-1, -2, -4, -1, 2, 1), nrow=3, byrow=TRUE)
b <- c(14, -33, 20)

sol <- intlinprog(f, c(1, 2), A, b)
sol <- intlinprog(f, NULL, A, b)

sol$x
```

---

linprog *MatLab(R)-style Linear Programming in R using ROI*

Description

linprog provides a simple interface to ROI using the optimization model specification of MatLab(R)

minimize in x: f'*x subject to: A*x <= b subject to: Aeq*x == beq x >= lb x <= ub

Usage

```
linprog(f, A = NULL, b = NULL, Aeq = NULL, beq = NULL, lb = NULL,
  ub = NULL, x0 = NULL, options = NULL)
```

Arguments

`f`	Linear term (vector) of the objective function
`A`	Inequality constraints (left-hand side)
`b`	Inequality constraints (right-hand side)
`Aeq`	Equality constraints (left-hand side)
`beq`	Equality constraints (right-hand side)
`lb`	Lower bound
`ub`	Upper bound
`x0`	Initial solution
`options`	Additional optimization parameters

Value

The solution vector in `x` as well as the objective value in `fval`.

Author(s)

Ronald Hochreiter, `<ron@hochreiter.net>`

Examples

```
# maximize: 2x1 + x2
# subject to:
#   x1 + x2 <= 5
#   x1 <= 3
#   x1 >= 0, x2 >= 0

f <- c(2, 1)
A <- matrix(c(1, 1, 1, 0), nrow=2, byrow=TRUE)
b <- c(5, 3)

sol <- linprog(-f, A, b)
sol$x
```

`quadprog` *MatLab(R)-style Quadratic Programming in R using ROI*

Description

quadprog provides a simple interface to ROI using the optimization model specification of MatLab(R)

minimize in x: f'*x + 0.5*x'*H*x subject to: A*x <= b Aeq*x == beq x >= lb x <= ub

Usage

```
quadprog(H, f, A = NULL, b = NULL, Aeq = NULL, beq = NULL,
  lb = NULL, ub = NULL, x0 = NULL, options = NULL)
```

Arguments

H	Quadratic term (matrix) of the objective function
f	Linear term (vector) of the objective function
A	Inequality constraints (left-hand side)
b	Inequality constraints (right-hand side)
Aeq	Equality constraints (left-hand side)
beq	Equality constraints (right-hand side)
lb	Lower bound
ub	Upper bound
x0	Initial solution
options	Additional optimization parameters

Value

The solution vector in x as well as the objective value in fval.

Author(s)

Ronald Hochreiter, <ron@hochreiter.net>

Examples

```
# Covariance matrix of four stocks (weekly returns from 2011):
#
#              AAPL          IBM         MSFT         ORCL
# AAPL 0.0014708114 0.0006940036 0.0006720841 0.0008276391
# IBM  0.0006940036 0.0009643581 0.0006239411 0.0011266429
# MSFT 0.0006720841 0.0006239411 0.0009387707 0.0008728736
# ORCL 0.0008276391 0.0011266429 0.0008728736 0.0021489512

covariance = matrix(c(0.0014708114, 0.0006940036, 0.0006720841, 0.0008276391,
                      0.0006940036, 0.0009643581, 0.0006239411, 0.0011266429,
                      0.0006720841, 0.0006239411, 0.0009387707, 0.0008728736,
                      0.0008276391, 0.0011266429, 0.0008728736, 0.0021489512),
                      nrow=4, byrow=TRUE)
assets <- dim(covariance)[1]

H <- covariance
f <- rep(0, assets)
Aeq <- rep(1, assets)
beq <- 1
lb <- rep(0, assets)
ub <- rep(1, assets)
```

```
solution <- quadprog(H, f, NULL, NULL, Aeq, beq, lb, ub)
portfolio <- solution$x
print(portfolio)
```

Index