Title

solve_tol() — Tolerance used by solvers and inverters

Syntax  Description  Remarks and examples  Conformability

Syntax

real scalar  solve_tol(numeric matrix Z, real scalar usertol)

Description

solve_tol(Z, usertol) returns the tolerance used by many Mata solvers to solve $AX = B$ and by many Mata inverters to obtain $A^{-1}$. usertol is the tolerance specified by the user or is missing value if the user did not specify a tolerance.

Remarks and examples

The tolerance used by many Mata solvers to solve $AX = B$ and by many Mata inverters to obtain $A^{-1}$ is

$$
eta = s * \frac{\text{trace}(\text{abs}(Z))}{n}
$$

(1)

where $s = 1e-13$ or a value specified by the user, $n$ is the $\min(\text{rows}(Z), \text{cols}(Z))$, and $Z$ is a matrix related to $A$, usually by some form of decomposition, but could be $A$ itself (for instance, if $A$ were triangular). See, for instance, [M-5] solvelower() and [M-5] cholsolve().

When $usertol > 0$ and $usertol < .$. is specified, solve_tol() returns $eta$ calculated with $s = usertol$.

When $usertol \leq 0$ is specified, solve_tol() returns $-usertol$.

When $usertol \geq .$. is specified, solve_tol() returns a default result, calculated as

1. If external real scalar _solvetolerance does not exist, as is usually the case, the value of $eta$ is returned using $s = 1e-13$.

2. If external real scalar _solvetolerance does exist,

   a. If _solvetolerance $> 0$, the value of $eta$ is returned using $s = \text{solvetolerance}$. 

   b. If _solvetolerance $\leq 0$, $-\text{solvetolerance}$ is returned.

Conformability

solve_tol(Z, usertol):

- $Z$: $r \times c$
- $usertol$: $1 \times 1$
- $result$: $1 \times 1$
Diagnostics

`solve_tol(Z, usertol)` skips over missing values in `Z` in calculating (1); `n` is defined as the number of nonmissing elements on the diagonal.

Also see

[M-4] utility — Matrix utility functions