

solve_tol() — Tolerance used by solvers and inverters

Description
Diagnostics

Syntax
Also see

Remarks and examples

Conformability

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.

Syntax

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

Remarks and examples

stata.com

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

$$\begin{aligned} \eta &= s * \frac{\text{trace}(\text{abs}(Z))}{n} && \text{when } s > 0 \\ \eta &= -s && \text{when } s \leq 0 \end{aligned} \tag{1}$$

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, `solvetol()` returns η calculated with $s = usertol$.

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

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

1. If the `matasolvetol` setting is set to `.` (missing), the value of η is computed using $s = 1e-13$.
2. If the `matasolvetol` setting is set to positive, the value of η is computed using $s = \text{st_numscalar}("c(\text{matasolvetol})")$.
3. If the `matasolvetol` setting is set to 0 or negative, the value of η is $-\text{st_numscalar}("c(\text{matasolvetol})")$.

Conformability

`solve_tol(Z, usertol)`:

<i>Z</i> :	$r \times c$
<i>usertol</i> :	1×1
<i>result</i> :	1×1

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