

Package: withinr (via r-universe)

October 6, 2026

Title High-Performance Fixed Effects Solver

Version 0.1.0

Author Alexander Fischer [aut, cre], Kristof Schroeder [aut]

Maintainer Alexander Fischer <alexander-fischer1801@t-online.de>

Description Fast modified LSMR solvers with Schwarz and diagonal preconditioners for absorbing high-dimensional fixed effects in panel data regressions. The computational core is written in Rust via version 0.2.0 of the 'within' crate and accessed through 'extendr'.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

SystemRequirements Rust tool chain w/ cargo, rustc

Config/rextendr/version 0.5.0

Config/pak/sysreqs libclang-dev

Repository <https://py-econometrics.r-universe.dev>

Date/Publication 2026-07-04 10:36:47 UTC

RemoteUrl <https://github.com/py-econometrics/within>

RemoteRef r-bindings

RemoteSha 1c1b0cb2c3b70e607cb31f76427883f374cf7f0e

RemoteSubdir withinr

Contents

AdditiveSchwarz	2
ApproxCholConfig	2
ApproxSchurConfig	3
LocalSolverConfig	3
LsmrOptions	4

Preconditioner	4
PreconditionerConfig	5
ReductionStrategy	5
solve	6
solve_batch	6
Solver	7
Index	8

AdditiveSchwarz	<i>Additive Schwarz preconditioner configuration</i>
-----------------	--

Description

Additive Schwarz preconditioner configuration

Usage

AdditiveSchwarz(local_solver = NULL, reduction = ReductionStrategy\$Auto)

Arguments

local_solver	NULL for the library default, or a LocalSolverConfig() object.
reduction	One of ReductionStrategy\$Auto, ReductionStrategy\$AtomicScatter, or ReductionStrategy\$ParallelReduction.

Value

A preconditioner configuration object.

ApproxCholConfig	<i>Approximate Cholesky configuration</i>
------------------	---

Description

Approximate Cholesky configuration

Usage

ApproxCholConfig(seed = 0, split_merge = NULL)

Arguments

seed	Non-negative integer random seed.
split_merge	Optional positive integer split/merge parameter, or NULL.

Value

A local-solver configuration object.

ApproxSchurConfig	<i>Approximate Schur complement configuration</i>
-------------------	---

Description

Approximate Schur complement configuration

Usage

```
ApproxSchurConfig(seed = 0, split = 1L)
```

Arguments

seed	Non-negative integer random seed.
split	Positive integer edge split factor.

Value

A local-solver configuration object.

LocalSolverConfig	<i>Local solver configuration for additive Schwarz subdomains</i>
-------------------	---

Description

Local solver configuration for additive Schwarz subdomains

Usage

```
LocalSolverConfig(
  approx_chol = NULL,
  approx_schur = .default_approx_schur(),
  dense_threshold = NULL
)
```

Arguments

approx_chol	NULL for the library default, or an ApproxCholConfig() object.
approx_schur	Omitted for the library default approximate Schur, NULL for exact Schur, or an ApproxSchurConfig() object.
dense_threshold	Optional non-negative integer dense Schur threshold.

Value

A local-solver configuration object accepted by [AdditiveSchwarz\(\)](#).

LsmrOptions	<i>Modified LSMR solver options</i>
-------------	-------------------------------------

Description

Uses Modified Golub-Kahan bidiagonalization to solve the least-squares problem directly. The preconditioner approximates $A^T A$ and is applied as one M^{-1} solve per iteration.

Usage

```
LsmrOptions(tol = 1e-08, maxiter = 1000L, local_size = NULL)
```

```
lsmr_options(tol = 1e-08, maxiter = 1000L, local_size = NULL)
```

Arguments

tol	Positive finite convergence tolerance. Default 1e-8.
maxiter	Maximum number of LSMR iterations. Default 1000L.
local_size	Optional window size for modified Gram-Schmidt reorthogonalization, or NULL to use the short recurrence.

Value

A solver options object accepted by [solve\(\)](#), [solve_batch\(\)](#), and persistent solver methods.

Preconditioner	<i>Built preconditioner handle</i>
----------------	------------------------------------

Description

Deserializes a preconditioner from raw bytes produced by `solver$preconditioner()$serialize()`. Handles returned by `solver$preconditioner()` also expose `$variant` and `$build_time_seconds` metadata. Deserialized handles keep `$variant` but report `$build_time_seconds = NA_real_`.

Usage

```
Preconditioner(data)
```

Arguments

data	Raw vector containing serialized preconditioner bytes.
------	--

Value

A `within_preconditioner` object with `$apply()` and `$serialize()` methods, `$nrows`, `$ncols`, `$variant`, and `$build_time_seconds` fields.

PreconditionerConfig	<i>Preconditioner shortcut values</i>
----------------------	---------------------------------------

Description

Use `PreconditionerConfig$Additive`, `PreconditionerConfig$Off`, or `PreconditionerConfig$Diagonal` as the preconditioner argument. `Additive` builds the default additive Schwarz preconditioner, `Off` disables preconditioning, and `Diagonal` uses diagonal/Jacobi scaling.

Usage

`PreconditionerConfig`

Format

An object of class `within_preconditioner_config` of length 3.

ReductionStrategy	<i>Reduction strategy shortcut values</i>
-------------------	---

Description

Use these values inside `AdditiveSchwarz()`.

Usage

`ReductionStrategy`

Format

An object of class `within_reduction_strategy` of length 3.

solve	<i>Solve fixed-effects normal equations</i>
-------	---

Description

Computes fixed-effect coefficients by solving the normal equations $D^T W D x = D^T W y$ where D is the dummy-variable design matrix implied by categories and W is the diagonal weight matrix.

Usage

```
solve(categories, y, options = NULL, weights = NULL, preconditioner = NULL)
```

Arguments

categories	Integer matrix of shape (n_obs, n_factors). Each column contains 1-based factor level assignments. Values must be positive integers with no NAs.
y	Numeric vector of length n_obs.
options	NULL for default LsmrOptions() or an options object.
weights	Numeric vector of length n_obs or NULL.
preconditioner	Controls preconditioning. Five input forms are accepted: NULL builds the default additive Schwarz preconditioner, PreconditionerConfig\$Off disables preconditioning, PreconditionerConfig\$Diagonal uses diagonal/Jacobi scaling, AdditiveSchwarz() overrides local-solver and reduction settings, and a built Preconditioner() object reuses an existing factorization.

Value

A named list with fields x, demeaned, converged, iterations, residual, time_total, time_setup, and time_solve.

solve_batch	<i>Solve fixed-effects normal equations for multiple response vectors</i>
-------------	---

Description

Builds the operator and preconditioner once, then solves for each column of Y in parallel.

Usage

```
solve_batch(
  categories,
  Y,
  options = NULL,
  weights = NULL,
  preconditioner = NULL
)
```

Arguments

categories	Integer matrix of shape (n_obs, n_factors). Each column contains 1-based factor level assignments. Values must be positive integers with no NAs.
Y	Numeric matrix of shape (n_obs, k).
options	NULL for default LsmrOptions() or an options object.
weights	Numeric vector of length n_obs or NULL.
preconditioner	Controls preconditioning. Five input forms are accepted: NULL builds the default additive Schwarz preconditioner, <code>PreconditionerConfig\$Off</code> disables preconditioning, <code>PreconditionerConfig\$Diagonal</code> uses diagonal/Jacobi scaling, AdditiveSchwarz() overrides local-solver and reduction settings, and a built Preconditioner() object reuses an existing factorization.

Value

A named list with matrix fields `x` and `demeaned`, plus `converged`, `iterations`, `residual`, `time_solve`, and `time_total`.

Solver	<i>Persistent fixed-effects solver</i>
--------	--

Description

Builds the preconditioner once and reuses it for repeated solves with the same design matrix.

Usage

```
Solver(categories, weights = NULL, preconditioner = NULL)
```

Arguments

categories	Integer matrix of shape (n_obs, n_factors). Each column contains 1-based factor level assignments. Values must be positive integers with no NAs.
weights	Numeric vector of length n_obs or NULL.
preconditioner	Controls preconditioning. Five input forms are accepted: NULL builds the default additive Schwarz preconditioner, <code>PreconditionerConfig\$Off</code> disables preconditioning, <code>PreconditionerConfig\$Diagonal</code> uses diagonal/Jacobi scaling, AdditiveSchwarz() overrides local-solver and reduction settings, and a built Preconditioner() object reuses an existing factorization.

Value

A `within_solver` object with `$solve()`, `$solve_batch()`, and `$preconditioner()` methods, plus `$n_dofs` and `$n_obs` fields.

Index

* datasets

- PreconditionerConfig, [5](#)
- ReductionStrategy, [5](#)

AdditiveSchwarz, [2](#)

AdditiveSchwarz(), [3](#), [5–7](#)

ApproxCholConfig, [2](#)

ApproxCholConfig(), [3](#)

ApproxSchurConfig, [3](#)

ApproxSchurConfig(), [3](#)

LocalSolverConfig, [3](#)

LocalSolverConfig(), [2](#)

lsmr_options(LsmrOptions), [4](#)

LsmrOptions, [4](#)

LsmrOptions(), [6](#), [7](#)

Preconditioner, [4](#)

Preconditioner(), [6](#), [7](#)

PreconditionerConfig, [5](#)

ReductionStrategy, [5](#)

solve, [6](#)

solve(), [4](#)

solve_batch, [6](#)

solve_batch(), [4](#)

Solver, [7](#)