

Package: modelstats (via r-universe)

October 7, 2026

Type Package

Title Run Analysis Tools

Version 0.31.0

Date 2026-09-25

Description A collection of tools to analyze model runs.

Imports dplyr, quitte, crayon, gdx2, gtools, lubridate, lucode2,
magpie4 (>= 2.33.4), piamutils, utils, gms, withr, rlang,
optparse, cli, stringi

Suggests covr, knitr, rmarkdown

URL <https://github.com/pik-piam/modelstats>

BugReports <https://github.com/pik-piam/modelstats/issues>

Encoding UTF-8

License LGPL-3

VignetteBuilder knitr

Config/roxygen2/version 8.1.0

Config/pak/sysreqs cmake libfontconfig1-dev libfreetype6-dev libfribidi-dev libgdal-dev gdal-
bin libgeos-dev git libglpk-dev make libharfbuzz-dev libgit2-dev libicu-dev libjpeg-dev libpng-
dev libtiff-dev libuv1-dev libwebp-dev libxml2-dev libssl-dev libproj-dev libsqlite3-dev libx11-
dev zlib1g-dev

Repository <https://pik-piam.r-universe.dev>

Date/Publication 2026-09-25 10:32:11 UTC

RemoteUrl <https://github.com/pik-piam/modelstats>

RemoteRef HEAD

RemoteSha eaf41517c5ff1d7df989523b771065c58591a935

Contents

modelstats-package	2
colRunType	2

commandLineInterface	3
foundInSlurm	3
getRunStatus	4
getSanityChecks	5
loopRuns	5
modeltests	6
printOutput	7
promptAndRun	8
Index	9

modelstats-package	<i>modelstats: Run Analysis Tools</i>
--------------------	---------------------------------------

Description

A collection of tools to analyze model runs.

Author(s)

Maintainer: Anastasis Giannousakis <giannou@pik-potsdam.de>

Authors:

- Anastasis Giannousakis <giannou@pik-potsdam.de>
- Oliver Richters
- Simon Krogmann

See Also

Useful links:

- <https://github.com/pik-piam/modelstats>
- Report bugs at <https://github.com/pik-piam/modelstats/issues>

colRunType	<i>colRunType</i>
------------	-------------------

Description

What is the type of this run?

Usage

colRunType(mydir = ".")

Arguments

mydir	Path to the folder where the run is performed
-------	---

Author(s)

Anastasis Giannousakis

commandLineInterface	<i>Provide a command line interface to loopRuns and getSanityChecks</i>
----------------------	---

Description

This function is made to be called by the rs tool. It takes the command line arguments supplied by the user when calling rs, evaluates them, and according to them prepares the arguments finally passed to loopRuns or getSanityChecks.

Usage

```
commandLineInterface(argv)
```

Arguments

<code>argv</code>	A character vector containing the command line arguments.
-------------------	---

Value

A string containing the status or sanity table.

Author(s)

David Klein

foundInSlurm	<i>foundInSlurm</i>
--------------	---------------------

Description

Is the run found in SLURM?

Usage

```
foundInSlurm(mydir = ".", user = NULL)
```

Arguments

mydir	Path to the folder(s) where the run(s) is(are) performed
user	the user whose runs will be sought for

Author(s)

Anastasis Giannousakis, Oliver Richters

getRunStatus

getRunStatus

Description

Returns the current status of a run or a vector of runs

Usage

```
getRunStatus(mydir = dir(), sort = "nf", user = NULL, detailed = TRUE)
```

Arguments

mydir	Path to the folder(s) where the run(s) is(are) performed
sort	how to sort (nf=newest first)
user	the user whose runs will be shown
detailed	(boolean, default: TRUE). If FALSE, avoids costly information gathering and returns less info

Author(s)

Anastasis Giannousakis

Examples

```
## Not run:  
  
a <- getRunStatus(dir())  
  
## End(Not run)
```

getSanityChecks	<i>getSanityChecks</i>
-----------------	------------------------

Description

Get overview of sanity check results for a list of REMIND runs

Usage

```
getSanityChecks(dirs = NULL)
```

Arguments

dirs	a vector of paths to REMIND runs. When NULL, the latest AMTs are used (only works on PIK cluster)
------	---

Details

Currently the following columns are printed:

- SumErr Summation Errors found when running remind2::convGDx2MIF
- RangeErr Range Errors found when running remind2::convGDx2MIF
- FixingErr Fixing Errors found when running piamInterfaces::fixOnRef
- MissingVar Missing Variables found when running piamInterfaces::checkMissingVars for ScenarioMIP
- ProjSumErr Summation Errors found when running piamInterfaces::checkSummations for ScenarioMIP
- ProjSumErrReg Regional Summation Errors found when running piamInterfaces::checkSummationsRegional for ScenarioMIP

Author(s)

Falk Benke

loopRuns	<i>loopRuns</i>
----------	-----------------

Description

Returns the output for getRunStatus nicely

Usage

```
loopRuns(mydir, user = NULL, colors = TRUE, sortbytime = TRUE)
```

Arguments

mydir	a dir or vector of dirs
user	the user whose runs will be shown
colors	boolean whether the output is colored dependent on runstatus
sortbytime	boolean whether the output is sorted by timestamp

Author(s)

Anastasis Giannousakis

Examples

```
## Not run:

loopRuns(dir())

## End(Not run)
```

modeltests

modeltests

Description

Runs a group of tests for a specific model. A "config/scenario_config.csv" file (relative) to the main folder has to exist in the model, describing the scenarios to be tested. Also, a gitlab repository is needed to push the generated README.md for documentation and automated reporting of test results.

Usage

```
modeltests(
  mydir = ".",
  gitdir = NULL,
  model = NULL,
  user = NULL,
  email = TRUE,
  compScen = TRUE,
  mattermostToken = NULL
)
```

Arguments

mydir	Path to the folder where the model is found
gitdir	path to the git clone that sends the report via email
model	Model name

<code>user</code>	the user that starts the job and commits the changes
<code>email</code>	whether an email notification will be send or not
<code>compScen</code>	whether compScen has to run or not
<code>mattermostToken</code>	token used for mattermost notifications

Author(s)

Anastasis Giannousakis, David Klein

See Also

[package2readme](#)

<code>printOutput</code>	<i>printOutput</i>
--------------------------	--------------------

Description

Returns the output for `getRunStatus` nicely

Usage

```
printOutput(  
  string,  
  len1stcol = 67,  
  lenCols = NULL,  
  colSep = "  ",  
  cols = NULL  
)
```

Arguments

<code>string</code>	list with information to print
<code>len1stcol</code>	length of first column
<code>lenCols</code>	vector with length information for all columns
<code>colSep</code>	string separating the columns
<code>cols</code>	optional vector of elements to retrieve from the passed list. Default NULL corresponds to a predefined set of elements.

Author(s)

Anastasis Giannousakis, Falk Benke

`promptAndRun`*promptAndRun*

Description

prompts runs, will be called by rs2

Usage

```
promptAndRun(mydir = ".", user = NULL, daysback = 3)
```

Arguments

<code>mydir</code>	a dir or vector of dirs
<code>user</code>	the user whose runs will be shown. For -t, the pattern used for matching
<code>daysback</code>	integer defining the number of days -c will look back in time to find runs

Author(s)

Anastasis Giannousakis, Oliver Richters

Examples

```
## Not run:  
  promptAndRun()  
  
## End(Not run)
```

Index

colRunType, [2](#)
commandLineInterface, [3](#)

foundInSlurm, [3](#)

getRunStatus, [4](#)
getSanityChecks, [5](#)

loopRuns, [5](#)

modelstats (modelstats-package), [2](#)
modelstats-package, [2](#)
modeltests, [6](#)

package2readme, [7](#)
printOutput, [7](#)
promptAndRun, [8](#)