

Package: n2kanalysis (via r-universe)

September 21, 2026

Title Generic Functions to Analyse Data from the 'Natura 2000'
Monitoring

Version 0.4.1

Description All generic functions and classes for the analysis for the 'Natura 2000' monitoring. The classes contain all required data and definitions to fit the model without the need to access other sources. Potentially they might need access to one or more parent objects. An aggregation object might for example need the result of an imputation object. The actual definition of the analysis, using these generic function and classes, is defined in dedicated analysis R packages for every monitoring scheme. For example 'abvanalysis' and 'watervogelanalysis'.

License GPL-3

URL <https://doi.org/10.5281/zenodo.3576047>,
<https://inbo.github.io/n2kanalysis/>

BugReports <https://github.com/inbo/n2kanalysis/issues>

Depends R (>= 4.2.0)

Imports assertthat (>= 0.2.1), aws.s3, digest (>= 0.6.23.2), dplyr, fs, git2rdata (>= 0.5.0), MASS, methods, multimpur (>= 0.2.14), n2khelper (>= 0.5.0), purrr, rlang, RODBC, tibble, tidyr (>= 0.4.0), tools, yaml

Suggests fmesher, INLA (>= 23.04.24), knitr, Matrix, parallel, rmarkdown, sf, sn, testthat (>= 2.0.1)

VignetteBuilder knitr

Remotes inbo/multimpur, inbo/n2khelper

Additional_repositories <https://inla.r-inla-download.org/R/stable>,
<https://inbo.r-universe.dev>

Config/checklist/communities inbo

Config/checklist/keywords analysis, reproducible research, natura 2000, monitoring

Encoding UTF-8

Language en-GB

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Collate 'n2k_parameter_class.R' 'n2k_anomaly_class.R'
 'n2k_analysis_version_class.R' 'combine.R'
 'n2k_contrast_class.R' 'n2k_analysis_metadata_class.R'
 'n2k_result_class.R' 'combine_result.R' 'connect_inbo_s3.R'
 'import_s3_classes.R' 'delete_model.R' 'display.R'
 'n2k_model_class.R' 'n2k_inla_comparison_class.R'
 'n2k_inla_class.R' 'n2k_aggregate_class.R'
 'n2k_model_imputed_class.R' 'extract.R' 'fit_every_model.R'
 'fit_model.R' 'fit_model_character.R'
 'fit_model_n2k_aggregate.R' 'n2k_composite_class.R'
 'fit_model_n2k_composite.R' 'n2k_hurdle_imputed_class.R'
 'fit_model_n2k_hurdle_imputed.R' 'fit_model_n2k_inla.R'
 'fit_model_n2k_inla_comparison.R' 'n2k_manifest_class.R'
 'fit_model_n2k_manifest.R' 'fit_model_n2k_model_imputed.R'
 'spde_class.R' 'n2k_spde_class.R' 'fit_model_n2k_spde.R'
 'fit_model_s3_object.R' 'get_analysis_date.R'
 'get_analysis_version.R' 'get_anomaly.R'
 'get_anomaly_n2k_inla.R' 'get_anomaly_n2k_model.R' 'get_data.R'
 'get_datafield_id.R' 'get_file_fingerprint.R' 'get_formula.R'
 'get_location_group_id.R' 'get_model.R' 'get_model_parameter.R'
 'get_model_parameter_n2k_aggregated.R'
 'get_model_parameter_n2k_composite.R'
 'get_model_parameter_n2k_hurdle_imputed.R'
 'get_model_parameter_n2k_inla.R'
 'get_model_parameter_n2k_inla_comparison.R'
 'get_model_parameter_n2k_model_imputed.R' 'get_model_type.R'
 'get_parents.R' 'get_result.R' 'get_result_character.R'
 'get_result_n2k_inla.R' 'get_result_n2k_model.R'
 'get_result_n2kmanifest.R' 'get_result_s3_object.R'
 'get_scheme_id.R' 'get_seed.R' 'get_species_group_id.R'
 'get_status_fingerprint.R' 'inla_inverse.R' 'make_a.R'
 'manifest_yaml_to_bash.R' 'mark_obsolete_dataset.R'
 'moving_average.R' 'moving_difference.R' 'moving_trend.R'
 'n2k_aggregated.R' 'n2k_composite.R' 'n2k_hurdle_imputed.R'
 'n2k_import_class.R' 'n2k_import.R' 'n2k_inla.R'
 'n2k_inla_comparison.R' 'n2k_manifest.R' 'n2k_model_imputed.R'
 'n2k_spde.R' 'parent_status.R' 'read_manifest.R' 'read_model.R'
 'read_result.R' 'result_estimate.R'
 'result_estimate_character.R' 'result_estimate_n2k_result.R'
 'result_metadata.R' 'result_metadata_character.R'
 'result_metadata_n2k_result.R'
 'select_factor_count_strictly_positive.R'
 'select_factor_treshold.R' 'select_observed_range.R'

'session_package.R' 'sha1.R' 'spde.R' 'spde2matern.R'
 'spde2mesh.R' 'status.R' 'store_manifest.R'
 'store_manifest_yaml.R' 'store_model.R' 'union.R'
 'valid_object.R' 'write_s3_fun.R'

Repository <https://inbo.r-universe.dev>

Date/Publication 2026-04-08 16:18:20 UTC

RemoteUrl <https://github.com/inbo/n2kanalysis>

RemoteRef HEAD

RemoteSha 1a76ccfbf57f05f40a65337df513c13491225e40

Contents

combine	5
connect_inbo_s3	5
delete_model	6
display	6
extract	7
fit_every_model	7
fit_model	8
get_analysis_date	10
get_analysis_version	11
get_anomaly	11
get_data	12
get_datafield_id	13
get_file_fingerprint	13
get_formula	14
get_location_group_id	14
get_model	15
get_model_parameter	15
get_model_type	16
get_parents	17
get_result	17
get_scheme_id	18
get_seed	19
get_species_group_id	19
get_status_fingerprint	20
inla_inverse	20
make_a	21
manifest_yaml_to_bash	21
mark_obsolete_dataset	22
moving_average	23
moving_difference	23
moving_trend	24
n2k_aggregate	25
n2k_composite	26
n2k_hurdle_imputed	27

n2k_import	28
n2k_inla	29
n2k_inla_comparison	31
n2k_manifest	32
n2k_model_imputed	32
n2k_spde	33
n2kAggregate-class	36
n2kAnalysisMetadata-class	36
n2kAnalysisVersion-class	36
n2kAnomaly-class	36
n2kComposite-class	37
n2kContrast-class	37
n2kHurdleImputed-class	37
n2kImport-class	37
n2kInla-class	38
n2kInlaComparison-class	38
n2kManifest-class	38
n2kModel-class	39
n2kModelImputed-class	39
n2kParameter-class	39
n2kResult-class	39
n2kSpde-class	40
order_manifest	40
parent_status	40
parent_status<-	41
read_manifest	41
read_model	42
read_result	42
result_estimate	43
result_metadata	43
select_factor_count_non_zero	44
select_factor_threshold	45
select_observed_range	46
session_package	46
spde	47
Spde-class	47
spde2matern	47
spde2matern,Spde-method	48
status	48
status<-	49
store_manifest	49
store_manifest_yaml	50
store_model	51
union	51
validObject,character-method	52

combine	<i>Combine multiple n2k objects</i>
---------	-------------------------------------

Description

Combine multiple n2k objects

Usage

```
combine(...)  
  
## S4 method for signature 'n2kAnalysisVersion'  
combine(...)  
  
## S4 method for signature 'n2kParameter'  
combine(...)  
  
## S4 method for signature 'n2kAnomaly'  
combine(...)  
  
## S4 method for signature 'n2kResult'  
combine(...)
```

Arguments

... n2k objects

connect_inbo_s3	<i>Set environment variables for INBO S3 bucket</i>
-----------------	---

Description

Before running this function you must have an .aws folder in your home directory with a credentials file containing the credentials for the INBO shared infrastructure. Run the `aws assume role` command to get the credentials for the INBO shared infrastructure before running this function.

Usage

```
connect_inbo_s3()
```

delete_model	<i>Delete an n2kModel object</i>
--------------	----------------------------------

Description

Delete an n2kModel object

Usage

```
delete_model(x, base, project)

## S4 method for signature 'ANY,character'
delete_model(x, base, project)

## S4 method for signature 'ANY,s3_bucket'
delete_model(x, base, project)
```

Arguments

x	the file fingerprint of the n2kModel
base	the base location
project	will be a relative path within the base location

display	<i>Display a message.</i>
---------	---------------------------

Description

This is a short cut for `if(verbose) message(x)`.

Usage

```
display(verbose, message, linefeed = TRUE)
```

Arguments

verbose	A logical. When TRUE print the message. When FALSE do nothing.
message	a vector passed to message() .
linefeed	A logical. When TRUE append a newline character at the end of the message.

extract	<i>Extract the relevant coefficients</i>
---------	--

Description

Extract the relevant coefficients

Usage

```
extract(extractor, object, base, project)

## S4 method for signature 'ANY,character'
extract(extractor, object, base, project)

## S4 method for signature 'ANY,n2kInla'
extract(extractor, object, base = NULL, project = NULL)

## S4 method for signature 'ANY,n2kModelImputed'
extract(extractor, object, base = NULL, project = NULL)
```

Arguments

extractor	the extractor function
object	the n2kModel object
base	the optional base location of the object
project	the optional subdirectory

Value

the relevant coefficients

fit_every_model	<i>Fit the model to the analysis files</i>
-----------------	--

Description

Fit the model to the analysis files

Usage

```
fit_every_model(path, status, verbose = TRUE, n_cluster = 1, ...)
```

Arguments

path	The path containing the analysis files.
status	A vector with status levels to (re-)fit the model. Defaults to c("new", "waiting") when missing.
verbose	Show the name of the current analysis file on screen. Defaults to TRUE
n_cluster	The number of clusters to use.
...	Arguments passed to <code>fit_model()</code>

fit_model	<i>Fit an n2kModel object</i>
-----------	-------------------------------

Description

Fit an n2kModel object

Usage

```
fit_model(x, ...)

## S4 method for signature 'character'
fit_model(
  x,
  base,
  project,
  status = c("new", "waiting"),
  verbose = TRUE,
  ...,
  bucket
)

## S4 method for signature 'n2kAggregate'
fit_model(x, ...)

## S4 method for signature 'n2kComposite'
fit_model(x, base, project, status = "new", ...)

## S4 method for signature 'n2kHurdleImputed'
fit_model(x, base, project, status = c("new", "waiting"), ...)

## S4 method for signature 'n2kInla'
fit_model(
  x,
  status = "new",
  ...,
  timeout = NULL,
```

```

    seed = get_seed(x),
    num_threads = NULL,
    parallel_configs = TRUE
)

## S4 method for signature 'n2kInlaComparison'
fit_model(x, base, project, status = "new", verbose = TRUE, ...)

## S4 method for signature 'n2kManifest'
fit_model(
  x,
  base,
  project,
  status = c("new", "waiting"),
  verbose = TRUE,
  ...,
  local = NULL
)

## S4 method for signature 'n2kModelImputed'
fit_model(x, ...)

## S4 method for signature 'n2kSpde'
fit_model(
  x,
  status = "new",
  ...,
  timeout = NULL,
  seed = get_seed(x),
  num_threads = NULL,
  parallel_configs = TRUE
)

## S4 method for signature 's3_object'
fit_model(x, status = c("new", "waiting"), ...)

```

Arguments

x	the n2kModel
...	other arguments. See details
base	The root of a project. Can be either a directory on a file system or an AWS S3 bucket object. Extracted from bucket or x when missing.
project	The subdirectory of the project. Is relative the base. Extracted from x when missing.
status	A vector with status levels naming the levels which should be calculated. Defaults to "new".
verbose	A logical indicating if the function should display the name of the file and the status. Defaults to TRUE.

bucket	The name of the AWS S3 bucket. Only used when base is missing.
timeout	the optional number of second until the model will time out
seed	See the same argument in <code>INLA::inla.qsample()</code> for further information. In order to produce reproducible results, you ALSO need to make sure the RNG in R is in the same state, see the example in <code>INLA::inla.posterior.sample()</code> . When seed is non-zero, num_threads is forced to "1:1" and parallel_configs is set to FALSE, since parallel sampling would not produce a reproducible sequence of pseudo-random numbers.
num_threads	The number of threads to use in the format "A:B" defining the number threads in the outer (A) and inner (B) layer for nested parallelism. A "0" will be replaced intelligently. seed != 0 requires serial computations.
parallel_configs	Logical. If TRUE and not on Windows, then try to run each configuration in parallel (not Windows) using A threads (see num_threads), where each of them is using B:0 threads.
local	A local folder into which objects from an AWS S3 bucket are downloaded.

Details

- status: A vector with status levels naming the levels which should be recalculated. Defaults to `c("new", "waiting")`.
- verbose: A logical indicating if the function should display the name of the file and the status. Defaults to TRUE.

get_analysis_date	<i>Get the analysis date of an n2kModel object</i>
-------------------	--

Description

Get the analysis date of an n2kModel object

Usage

```
get_analysis_date(x)

## S4 method for signature 'n2kAnalysisMetadata'
get_analysis_date(x)
```

Arguments

x the n2kModel object

Value

the analysis date of the object

get_analysis_version	<i>Get an analysis version</i>
----------------------	--------------------------------

Description

Get an analysis version

Usage

```
get_analysis_version(version)

## S4 method for signature 'data.frame'
get_analysis_version(version)

## S4 method for signature 'n2kAnalysisMetadata'
get_analysis_version(version)

## S4 method for signature 'sessionInfo'
get_analysis_version(version)
```

Arguments

version	the object to extract the version
---------	-----------------------------------

get_anomaly	<i>Get the anomalies from a model</i>
-------------	---------------------------------------

Description

Get the anomalies from a model

Usage

```
get_anomaly(analysis, ...)

## S4 method for signature 'n2kInla'
get_anomaly(
  analysis,
  n = 20,
  expected_ratio = 5,
  expected_absent = 5,
  random_threshold = 1.05,
  verbose = TRUE,
  ...
)
```

```
## S4 method for signature 'n2kModel'
get_anomaly(analysis, verbose = TRUE, ...)
```

Arguments

analysis	The model to add
...	Extra options. See details.
n	the number of anomalies per category.
expected_ratio	Observations that have observed / fitted > expected_ratio or fitted / observed > expected_ratio are potential anomalies. Defaults to 5, which implies that observed values that are 5 times higher or lower than the fitted values are potential anomalies.
expected_absent	Zero observations where fitted > expected_absent are potential anomalies.
random_threshold	The minimal relative effect size of a random effect. Random effect with a smaller effect size will never be an anomaly. Defaults to 1.05 (5%).
verbose	Print extra information on the screen

get_data	<i>Get the data of an n2kModel object</i>
----------	---

Description

Get the data of an n2kModel object

Usage

```
get_data(x)

## S4 method for signature 'n2kInla'
get_data(x)
```

Arguments

x	the n2kModel object
---	---------------------

Value

the data of the object

get_datafield_id	<i>Get the data field id</i>
------------------	------------------------------

Description

Get the data field id

Usage

```
get_datafield_id(table, field, datasource, root, stage = FALSE)
```

Arguments

table	The table name
field	The field name
datasource	The data source name
root	The root of a project. Can be a file path or a git-repository. Defaults to the current working directory (".").
stage	Logical value indicating whether to stage the changes after writing the data. Defaults to FALSE.

get_file_fingerprint	<i>Get the file fingerprint of an n2kModel object</i>
----------------------	---

Description

Get the file fingerprint of an n2kModel object

Usage

```
get_file_fingerprint(x)

## S4 method for signature 'n2kAnalysisMetadata'
get_file_fingerprint(x)

## S4 method for signature 'n2kManifest'
get_file_fingerprint(x)
```

Arguments

x	the n2kModel object
---	---------------------

Value

the file fingerprint of the object

get_formula	<i>Get the formula of an n2kModel object</i>
-------------	--

Description

Get the formula of an n2kModel object

Usage

```
get_formula(x)

## S4 method for signature 'n2kAnalysisMetadata'
get_formula(x)
```

Arguments

x the n2kModel object

Value

the formula of the object

get_location_group_id	<i>Get the location group id of an n2kModel object</i>
-----------------------	--

Description

Get the location group id of an n2kModel object

Usage

```
get_location_group_id(x)

## S4 method for signature 'n2kAnalysisMetadata'
get_location_group_id(x)
```

Arguments

x the n2kModel object

Value

the location group id of the object

get_model	<i>Get the model of a n2kModel object</i>
-----------	---

Description

Get the model of a n2kModel object

Usage

```
get_model(x)

## S4 method for signature 'n2kInla'
get_model(x)

## S4 method for signature 'character'
get_model(x)
```

Arguments

x the n2kModel object

Value

the model of the object

get_model_parameter	<i>Add the model parameters from a model</i>
---------------------	--

Description

Add the model parameters from a model

Usage

```
get_model_parameter(analysis, ...)

## S4 method for signature 'n2kAggregate'
get_model_parameter(analysis, ...)

## S4 method for signature 'n2kComposite'
get_model_parameter(analysis, ...)

## S4 method for signature 'n2kHurdleImputed'
get_model_parameter(analysis, ...)

## S4 method for signature 'n2kInla'
```

```

get_model_parameter(analysis, verbose = TRUE, ...)

## S4 method for signature 'n2kInlaComparison'
get_model_parameter(analysis, ...)

## S4 method for signature 'n2kModelImputed'
get_model_parameter(analysis, ...)

```

Arguments

analysis	The model to add.
...	extra options
verbose	Print extra information on the screen

get_model_type	<i>Get the model type of an n2kModel object</i>
----------------	---

Description

Get the model type of an n2kModel object

Usage

```

get_model_type(x)

## S4 method for signature 'n2kAnalysisMetadata'
get_model_type(x)

```

Arguments

x	the n2kModel object
---	---------------------

Value

the model type of the object

get_parents	<i>Get the object of the parent analyses of an n2kModel object</i>
-------------	--

Description

Get the object of the parent analyses of an n2kModel object

Usage

```
get_parents(child, base, project)
```

Arguments

child	the child object
base	the base location to read the model
project	will be a relative path within the base location

get_result	<i>Add the results from an analysis</i>
------------	---

Description

Add the results from an analysis

Usage

```
get_result(x, base, ...)
```

```
## S4 method for signature 'character,character'
get_result(x, base, ..., project, verbose = TRUE)
```

```
## S4 method for signature 'character,s3_bucket'
get_result(x, base, ..., project, verbose = TRUE)
```

```
## S4 method for signature 'n2kInla,ANY'
get_result(x, base, ..., verbose = TRUE)
```

```
## S4 method for signature 'n2kModel,ANY'
get_result(x, verbose = TRUE, ...)
```

```
## S4 method for signature 'n2kManifest,character'
get_result(x, base, ..., verbose = TRUE)
```

```
## S4 method for signature 'n2kManifest,s3_bucket'
get_result(x, base, project, ..., verbose = TRUE)
```

```
## S4 method for signature 's3_object,ANY'
get_result(x, base, ..., project, verbose = TRUE)
```

Arguments

x	object with the current results
base	the base location to read the model
...	further arguments (see Details)
project	will be a relative path within the base location
verbose	Print extra information on the screen

get_scheme_id	<i>Get the scheme_id of an n2kModel object</i>
---------------	--

Description

Get the scheme_id of an n2kModel object

Usage

```
get_scheme_id(x)

## S4 method for signature 'n2kAnalysisMetadata'
get_scheme_id(x)
```

Arguments

x	the n2kModel object
---	---------------------

Value

the scheme_id of the object

get_seed	<i>Get the seed of an n2kModel object</i>
----------	---

Description

Get the seed of an n2kModel object

Usage

```
get_seed(x)

## S4 method for signature 'n2kAnalysisMetadata'
get_seed(x)
```

Arguments

x the n2kModel object

Value

the seed of the object

get_species_group_id	<i>Get the species group id of an n2kModel object</i>
----------------------	---

Description

Get the species group id of an n2kModel object

Usage

```
get_species_group_id(x)

## S4 method for signature 'n2kAnalysisMetadata'
get_species_group_id(x)
```

Arguments

x the n2kModel object

Value

the species group id of the object

```
get_status_fingerprint
```

Get the status fingerprint of an n2kModel object

Description

Get the status fingerprint of an n2kModel object

Usage

```
get_status_fingerprint(x)
```

```
## S4 method for signature 'n2kAnalysisMetadata'
```

```
get_status_fingerprint(x)
```

Arguments

x the n2kModel object

Value

the status fingerprint of the object

```
inla_inverse
```

Calculate the mean and 95\ marginal

Description

This is useful for calculating the variance of a random effect when the marginal gives the precision of the random effect.

Usage

```
inla_inverse(marginal)
```

Arguments

marginal The INLA marginal

make_a	<i>Convert an Spde object to a A object</i>
--------	---

Description

Convert an Spde object to a A object

Usage

```
make_a(object, data)

## S4 method for signature 'Spde'
make_a(object, data)
```

Arguments

object	The Spde object
data	The data.frame to combine with the Spde object to create the A matrix.

manifest_yaml_to_bash	<i>Convert a manifest yaml file into a bash script</i>
-----------------------	--

Description

Convert a manifest yaml file into a bash script

Usage

```
manifest_yaml_to_bash(
  base,
  project,
  hash,
  shutdown = FALSE,
  split = 1,
  status = c("new", "waiting"),
  limit = FALSE,
  timeout = 4
)

## S4 method for signature 's3_bucket'
manifest_yaml_to_bash(
  base,
  project,
  hash,
  shutdown = FALSE,
```

```

    split = 1,
    status = c("new", "waiting"),
    limit = FALSE,
    timeout = 4
  )

  ## S4 method for signature 'character'
  manifest_yaml_to_bash(
    base,
    project,
    hash,
    shutdown = FALSE,
    split = 1,
    status = c("new", "waiting"),
    limit = FALSE,
    timeout = 4
  )

```

Arguments

base	the base location to store the manifest
project	will be a relative path within the base location
hash	Fingerprint of the manifest yamlfile.
shutdown	Append a shutdown command at the end of the script. Defaults to FALSE.
split	Number of scripts over which to splits the analyses. Default to 1.
status	A vector with status levels naming the levels which should be calculated. Defaults to c("new", "waiting").
limit	Limit bandwidth and CPU usage. Defaults to FALSE.
timeout	number of hours to time out the docker container. Defaults to 4.

mark_obsolete_dataset *Find and mark obsolete datasets*

Description

A dataset is obsolete when a newer version is available

Usage

```
mark_obsolete_dataset(channel)
```

Arguments

channel	An open ODBC channel
---------	----------------------

moving_average	<i>Calculate moving trend coefficients</i>
----------------	--

Description

Calculate moving trend coefficients

Usage

```
moving_average(n_year, duration, first_year = 0)
```

Arguments

n_year	Number of available years in the data.
duration	Number of years in the moving window. If the number of available years is less than this value, the trend will be calculated for the available years.
first_year	First year of the data. Only used to name the rows.

Value

A matrix with moving average coefficients One row for each window and and one column for each year in the data. The format of the row names is average_{window mid point}_{window length}. average_2001.5_4 is the average for the years 2000 to 2003.

Examples

```
moving_average(5, 3)
moving_average(5, 3, 2000)
```

moving_difference	<i>Calculate coefficients for a moving difference</i>
-------------------	---

Description

Calculate coefficients for a moving difference

Usage

```
moving_difference(n_year, duration, first_year = 1)
```

Arguments

n_year	Number of available years in the data.
duration	Number of years in the moving window. If the number of available years is less than this value, the trend will be calculated for the available years.
first_year	First year of the data. Only used to name the rows.

Value

A matrix with moving average coefficients One row for each window and and one column for each year in the data. The format of the row names is difference_{window mid point start}_{window mid point end}_{window length}. difference_2001.5_2010.5_4 is the difference of the average for period for the years 2009 to 2012 compared to the period from 2000 to 2003.

Examples

```
moving_difference(6, 2)
moving_difference(6, 2, 2000)
```

moving_trend	Calculate moving trend coefficients
--------------	-------------------------------------

Description

Calculate moving trend coefficients

Usage

```
moving_trend(n_year, duration, first_year = 0)
```

Arguments

- n_year Number of available years in the data.
- duration Number of years in the moving window. If the number of available years is less than this value, the trend will be calculated for the available years.
- first_year First year of the data. Only used to name the rows.

Value

A matrix with the moving trend coefficients. One row for each window and and one column for each year in the data. The format of the row names is trend_{window mid point}_{window length}. trend_2001.5_4 is the trend for the years 2000 to 2003.

Examples

```
moving_trend(5, 3)
moving_trend(5, 4, 2000)
```

n2k_aggregate	Create an n2kAggregate object
---------------	-------------------------------

Description

A new n2kAggregate model.

Usage

```
n2k_aggregate(...)  
  
## S4 method for signature 'ANY'  
n2k_aggregate(...)
```

Arguments

... other arguments. See below

Details

- `scheme_id`: a string holding the id of the scheme.
- `species_group_id`: a string identifying the species group.
- `location_group_id`: a string identifying the location group.
- `model_type`: a string identifying the type of model to fit to the data.
- `first_imported_year`: Oldest year considered in the data.
- `last_imported_year`: Most recent year considered in the data.
- `duration`: The width of the moving window. Defaults to the `last_imported_year - first_imported_year + 1`.
- `last_analysed_year`: Most recent year in the window. Defaults to `last_imported_year`.
- `analysis_date`: A `POSIXct` date indicating the date that the dataset was imported.
- `seed`: A single integer used as a seed for all calculations. A random seed will be inserted when missing.
- `fun`: The function to apply when aggregating.
- `filter`: An optional list for the filter argument of `multimput::aggregate_impute()`.
- `join`: An optional list for the join argument of `multimput::aggregate_impute()`.

n2k_composite

Create an n2kComposite object

Description

A new n2kComposite model is created when parent is a data.frame.

Usage

```
n2k_composite(parent_status, ...)

## S4 method for signature 'data.frame'
n2k_composite(
  parent_status,
  status = "waiting",
  result_datasource_id,
  scheme_id,
  formula,
  species_group_id,
  location_group_id,
  model_type,
  first_imported_year,
  last_imported_year,
  duration,
  last_analysed_year,
  analysis_date,
  extractor,
  ...,
  seed
)
```

Arguments

parent_status	A data.frame with columns parent_analysis (the file fingerprint of the parent), parentstatus_fingerprint (the status fingerprint of the parent), and parent_status (the status of the parent).
...	other arguments
status	A single character indicating the status of the model. Defaults to "waiting".
result_datasource_id	A string identifying the data source.
scheme_id	A single integer holding the id of the scheme.
formula	A single character identifying the comparison.
species_group_id	A string identifying the species group.

location_group_id	A string identifying the location group.
model_type	The type of the models. Must start with "inla comparison:".
first_imported_year	Oldest year considered in the data.
last_imported_year	Most recent year considered in the data.
duration	The width of the moving window. Defaults to the last_imported_year - first_imported_year + 1.
last_analysed_year	Most recent year in the window. Defaults to last_imported_year.
analysis_date	A POSIXct date indicating the date that the dataset was imported.
extractor	a function to extract the relevant parameters from the model
seed	A single integer used as a seed for all calculations. A random seed will be inserted when missing.

n2k_hurdle_imputed	<i>Create an n2kHurdleImputed object</i>
--------------------	--

Description

A new n2kHurdleImputed model.

Usage

```
n2k_hurdle_imputed(presence, count, verbose = FALSE)
```

```
## S4 method for signature 'n2kInla'
n2k_hurdle_imputed(presence, count, verbose = FALSE)
```

Arguments

presence	the n2kInla object for the presence model.
count	the n2kInla object for the count model.
verbose	display the location group ID and species group ID. Defaults to FALSE.

n2k_import	Create an n2kImport object
------------	----------------------------

Description

A new n2kImport model.

Usage

```
n2k_import(...)

## S4 method for signature 'ANY'
n2k_import(...)
```

Arguments

... other arguments. See below

Details

- `scheme_id`: a string holding the id of the scheme.
- `species_group_id`: a string identifying the species group.
- `location_group_id`: a string identifying the location group.
- `model_type`: a string identifying the type of model to fit to the data.
- `first_imported_year`: Oldest year considered in the data.
- `last_imported_year`: Most recent year considered in the data.
- `duration`: The width of the moving window. Defaults to the `last_imported_year - first_imported_year + 1`.
- `last_analysed_year`: Most recent year in the window. Defaults to `last_imported_year`.
- `analysis_date`: A POSIXct date indicating the date that the dataset was imported.
- `seed`: A single integer used as a seed for all calculations. A random seed will be inserted when missing.
- `formula`: a string holding the model formula.
- `dataset`: A data.frame with filename, fingerprint and import_date.

n2k_inla

Create an n2kInla object

Description

A new n2kInla model is created when data is a `data.frame`.

In case data is an n2kInla object, then only the model and status are updated. All other slots are unaffected.

Usage

```
n2k_inla(data, ..., model_fit)

## S4 method for signature 'data.frame,ANY'
n2k_inla(
  data,
  status = "new",
  result_datasource_id,
  scheme_id,
  family = "poisson",
  formula,
  species_group_id,
  location_group_id,
  model_type,
  first_imported_year,
  last_imported_year,
  duration,
  last_analysed_year,
  analysis_date,
  lin_comb = NULL,
  minimum = "",
  imputation_size,
  parent = character(0),
  seed,
  replicate_name = list(),
  control = list(),
  parent_status = "converged",
  parent_statusfingerprint,
  extra,
  ...,
  model_fit
)

## S4 method for signature 'n2kInla,inla'
n2k_inla(data, status, raw_imputed = NULL, ..., model_fit)
```

Arguments

<code>data</code>	a <code>data.frame</code> with the data to analyse
<code>...</code>	other arguments. See below
<code>model_fit</code>	The fitted model
<code>status</code>	A single character indicating the status of the model. Defaults to "waiting".
<code>result_datasource_id</code>	A string identifying the data source.
<code>scheme_id</code>	A single integer holding the id of the scheme.
<code>family</code>	the family to use in the INLA model.
<code>formula</code>	A single character identifying the comparison.
<code>species_group_id</code>	A string identifying the species group.
<code>location_group_id</code>	A string identifying the location group.
<code>model_type</code>	The type of the models. Must start with "inla comparison:".
<code>first_imported_year</code>	Oldest year considered in the data.
<code>last_imported_year</code>	Most recent year considered in the data.
<code>duration</code>	The width of the moving window. Defaults to the <code>last_imported_year - first_imported_year + 1</code> .
<code>last_analysed_year</code>	Most recent year in the window. Defaults to <code>last_imported_year</code> .
<code>analysis_date</code>	A <code>POSIXct</code> date indicating the date that the dataset was imported.
<code>lin_comb</code>	A model matrix to calculate linear combinations.
<code>minimum</code>	The name of the variable which holds the minimum counts. Only relevant in case of multiple imputation.
<code>imputation_size</code>	The required number of imputations defaults to 0.
<code>parent</code>	The file fingerprint of the optional parent analysis.
<code>seed</code>	A single integer used as a seed for all calculations. A random seed will be inserted when missing.
<code>replicate_name</code>	A list with the names of replicates. Defaults to an empty list. Used in case of <code>f(X, ..., replicate = Z)</code> . Should be a named list like e.g. <code>list(X = c("a", "b", "c"))</code> .
<code>control</code>	A named list passed to <code>INLA::inla()</code> when fitting the model.
<code>parent_status</code>	The status of the parent analysis.
<code>parent_statusfingerprint</code>	The status fingerprint of the parent analysis.
<code>extra</code>	a <code>data.frame</code> with extra observations not used in the model. They will be added in subsequent analyses.
<code>raw_imputed</code>	the optional <code>rawImputed</code> object

n2k_inla_comparison	Create an n2kInlaComparison object
---------------------	------------------------------------

Description

A new n2kInlaComparison model is created when parent is a character.

Usage

```
n2k_inla_comparison(parent_status, ...)

## S4 method for signature 'data.frame'
n2k_inla_comparison(
  parent_status,
  status = "waiting",
  result_datasource_id,
  scheme_id,
  formula,
  species_group_id,
  location_group_id,
  model_type,
  first_imported_year,
  last_imported_year,
  duration,
  last_analysed_year,
  analysis_date,
  ...,
  seed
)
```

Arguments

parent_status	A data.frame with columns parent_analysis (the file fingerprint of the parent), parentstatus_fingerprint (the status fingerprint of the parent), and parent_status (the status of the parent).
...	other arguments
status	A single character indicating the status of the model. Defaults to "waiting".
result_datasource_id	A string identifying the data source.
scheme_id	A single integer holding the id of the scheme.
formula	A single character identifying the comparison.
species_group_id	A string identifying the species group.
location_group_id	A string identifying the location group.

model_type	The type of the models. Must start with "inla comparison:".
first_imported_year	Oldest year considered in the data.
last_imported_year	Most recent year considered in the data.
duration	The width of the moving window. Defaults to the last_imported_year - first_imported_year + 1.
last_analysed_year	Most recent year in the window. Defaults to last_imported_year.
analysis_date	A POSIXct date indicating the date that the dataset was imported.
seed	A single integer used as a seed for all calculations. A random seed will be inserted when missing.

n2k_manifest	<i>Create an n2kManifest object</i>
--------------	-------------------------------------

Description

A new n2kManifest object is created when manifest is a data.frame.

Usage

```
n2k_manifest(manifest)

## S4 method for signature 'data.frame'
n2k_manifest(manifest)
```

Arguments

manifest	a data.frame with the manifest content
----------	--

n2k_model_imputed	<i>Create an n2kModelImputed object</i>
-------------------	---

Description

A new n2kModelImputed model.

Usage

```
n2k_model_imputed(...)

## S4 method for signature 'ANY'
n2k_model_imputed(...)
```

Arguments

... other arguments. See below

Details

- `scheme_id`: a string holding the id of the scheme.
- `species_group_id`: a string identifying the species group.
- `location_group_id`: a string identifying the location group.
- `model_type`: a string identifying the type of model to fit to the data.
- `first_imported_year`: Oldest year considered in the data.
- `last_imported_year`: Most recent year considered in the data.
- `duration`: The width of the moving window. Defaults to the `last_imported_year - first_imported_year + 1`.
- `last_analysed_year`: Most recent year in the window. Defaults to `last_imported_year`.
- `analysis_date`: A `POSIXct` date indicating the date that the dataset was imported.
- `seed`: A single integer used as a seed for all calculations. A random seed will be inserted when missing.
- `model_fun`: The `model_fun` argument of `multimput::model_impute()`.
- `package`: A character vector of package names which must be loaded for `model_fun`.
- `model_args`: An optional list for the `model_args` argument of `multimput::model_impute()`.
- `extractor`: An optional list for the `extractor` argument of `multimput::model_impute()`.
- `extractor_args`: An optional list for the `extractor_args` argument of `multimput::model_impute()`.
- `filter`: An optional list for the `filter` argument of `multimput::model_impute()`.
- `mutate`: An optional list for the ‘mutate’ argument of `multimput::model_impute()`.

n2k_spde

Create an n2kSpde object

Description

A new `n2kSpde` model is created when data is a `data.frame`.

In case data is an `n2kSpde` object, then only the model and status are updated. All other slots are unaffected.

Usage

```

n2k_spde(data, ..., model_fit)

## S4 method for signature 'data.frame,ANY'
n2k_spde(
  data,
  status = "new",
  result_datasource_id,
  scheme_id,
  family = "poisson",
  formula,
  species_group_id,
  location_group_id,
  model_type,
  spde,
  first_imported_year,
  last_imported_year,
  duration,
  last_analysed_year,
  analysis_date,
  lin_comb = NULL,
  minimum = "",
  imputation_size,
  parent = character(0),
  seed,
  replicate_name = list(),
  control = list(),
  parent_status = "converged",
  parent_statusfingerprint,
  extra,
  ...,
  model_fit
)

## S4 method for signature 'n2kSpde,inla'
n2k_spde(data, status, raw_imputed = NULL, ..., model_fit)

```

Arguments

<code>data</code>	a <code>data.frame</code> with the data to analyse
<code>...</code>	other arguments. See below
<code>model_fit</code>	The fitted model
<code>status</code>	A single character indicating the status of the model. Defaults to "waiting".
<code>result_datasource_id</code>	A string identifying the data source.
<code>scheme_id</code>	A single integer holding the id of the scheme.
<code>family</code>	the family to use in the INLA model.

formula	A single character identifying the comparison.
species_group_id	A string identifying the species group.
location_group_id	A string identifying the location group.
model_type	The type of the models. Must start with "inla comparison:".
spde	the Spde object. See [spde].
first_imported_year	Oldest year considered in the data.
last_imported_year	Most recent year considered in the data.
duration	The width of the moving window. Defaults to the last_imported_year - first_imported_year + 1.
last_analysed_year	Most recent year in the window. Defaults to last_imported_year.
analysis_date	A POSIXct date indicating the date that the dataset was imported.
lin_comb	A model matrix to calculate linear combinations.
minimum	The name of the variable which holds the minimum counts. Only relevant in case of multiple imputation.
imputation_size	The required number of imputations defaults to 0.
parent	The file fingerprint of the optional parent analysis.
seed	A single integer used as a seed for all calculations. A random seed will be inserted when missing.
replicate_name	A list with the names of replicates. Defaults to an empty list. Used in case of <code>f(X, ..., replicate = Z)</code> . Should be a named list like e.g. <code>list(X = c("a", "b", "c"))</code> .
control	A named list passed to <code>INLA::inla()</code> when fitting the model.
parent_status	A data.frame with columns parent_analysis (the file fingerprint of the parent), parentstatus_fingerprint (the status fingerprint of the parent), and parent_status (the status of the parent).
parent_statusfingerprint	The status fingerprint of the parent analysis.
extra	a data.frame with extra observations not used in the model. They will be added in subsequent analyses.
raw_imputed	the optional rawImputed object

n2kAggregate-class	<i>The n2kAggregate class</i>
--------------------	-------------------------------

Description

It holds analysis data based on an aggregated imputation

Slots

- RawImputed A rawImputed object with multiple imputations.
- Function The function to apply on each group.
- Filter The object to pass to the filter argument of `multimput::aggregate_impute()`.
- Join The object to pass to the join argument of `multimput::aggregate_impute()`.
- AggregatedImputed An aggregatedImputed object with multiple imputations.

n2kAnalysisMetadata-class	<i>The n2kAnalysisMetadata class</i>
---------------------------	--------------------------------------

Description

The n2kAnalysisMetadata class

n2kAnalysisVersion-class	<i>The n2kAnalysisVersion class</i>
--------------------------	-------------------------------------

Description

The n2kAnalysisVersion class

n2kAnomaly-class	<i>The n2kAnomaly class</i>
------------------	-----------------------------

Description

The n2kAnomaly class

n2kComposite-class	<i>The n2kComposite class</i>
--------------------	-------------------------------

Description

Calculate composite indices from multiple analysis

Slots

Extractor A function to extract the relevant parameters from the model.

Parameter A data.frame with the relevant parameter estimates of each parent analysis.

Index The composite index based on the parameters.

n2kContrast-class	<i>The n2kContrast class</i>
-------------------	------------------------------

Description

The n2kContrast class

n2kHurdleImputed-class	<i>The n2kHurdleImputed class</i>
------------------------	-----------------------------------

Description

It holds the model2 of hurdle imputed data.

Slots

Presence A rawImputed object with the presence component.

Count A rawImputed object with the count component.

Hurdle A aggregatedImputed object with the resulting hurdle object.

n2kImport-class	<i>The n2kImport class</i>
-----------------	----------------------------

Description

It hold the imported datasets

Slots

Dataset A data.frame with the datasets.

n2kInla-class	<i>The n2kInla class</i>
---------------	--------------------------

Description

It hold analysis data based on an INLA Poisson model

Slots

Data A data.frame with the data.

LinearCombination An optional matrix with the linear combinations.

ReplicateName An optional list with names of replicates.

Model Either NULL or the resulting INLA model.

Family The family of the INLA model.

Control A named list with options passed to the arguments of `INLA::inla()`.

ImputationSize The number of multiple imputations. Defaults to 0, indication no multiple imputation.

Minimum An optional string containing the name of the variable in Data holding the minimal values for imputation.

RawImputed A rawImputed object with multiple imputations.

Extra A data.frame with extra data to add to the imputations. This data is not used in the imputation model. It must contain the same variables as the original data.

n2kInlaComparison-class	<i>The n2kInlaComparison class</i>
-------------------------	------------------------------------

Description

Compare multiple models using the WAIC criterion

Slots

WAIC A data.frame with WAIC values per model.

n2kManifest-class	<i>The n2kManifest class</i>
-------------------	------------------------------

Description

The n2kManifest class

n2kModel-class	<i>The n2kModel class</i>
----------------	---------------------------

Description

A virtual superclass to contain the analysis models

n2kModelImputed-class	<i>The n2kModelImputed class</i>
-----------------------	----------------------------------

Description

It holds the model of aggregated imputed data

Slots

Function The object to pass to the `model.fun` argument of `multimput::model_impute()`.
 Package A vector of package names which must be loaded to run the function.
 ModelArgs The object to pass to the `model.args` argument of `multimput::model_impute()`.
 PrepareModelArgs An optional list containing a single function that will be applied to the object.
 The result of the function will be appended to the `ModelArgs`.
 Extractor The object to pass to the `extractor` argument of `multimput::model_impute()`.
 ExtractorArgs The object to pass to the `extractor.args` argument of `multimput::model_impute()`.
 Filter The object to pass to the `filter` argument of `multimput::model_impute()`.
 Mutate The object to pass to the `mutate` argument of `multimput::model_impute()`.
 AggregatedImputed An `aggregatedImputed` object with multiple imputations.
 Results The `data.frame` with the results of `multimput::model_impute()`.

n2kParameter-class	<i>The n2kParameter class</i>
--------------------	-------------------------------

Description

The `n2kParameter` class

n2kResult-class	<i>The n2kResult class</i>
-----------------	----------------------------

Description

The `n2kResult` class

n2kSpde-class	<i>The n2kSpde class</i>
---------------	--------------------------

Description

It hold analysis data based on an INLA model with SPDE.

Slots

Spde A list containing the information for the SPDE.

order_manifest	<i>Convert a manifest in vector of analysis fingerprints</i>
----------------	--

Description

The order of the analysis is determined by the parent-child relationship. It starts with the analyses without parents. Then it adds the analyses with parents that have already been added. This process is repeated until all analyses have been added.

Usage

```
order_manifest(manifest)
```

Arguments

manifest	the n2kManifest
----------	-----------------

parent_status	<i>Get the parent status of an n2kModel</i>
---------------	---

Description

Get the parent status of an n2kModel

Usage

```
parent_status(x)
```

```
## S4 method for signature 'n2kAnalysisMetadata'
parent_status(x)
```

Arguments

x	the n2kModel object
---	---------------------

`parent_status<-`

41

Value

the parent status of the object

<code>parent_status<-</code>	<i>Overwrite the status of an n2kAnalysisMetadata</i>
---------------------------------	---

Description

Overwrite the status of an n2kAnalysisMetadata

Usage

```
parent_status(x) <- value

## S4 replacement method for signature 'n2kComposite'
parent_status(x) <- value
```

Arguments

<code>x</code>	the n2kAnalysisMetadata object
<code>value</code>	the new values for the status

<code>read_manifest</code>	<i>Read a n2kManifest object</i>
----------------------------	----------------------------------

Description

Read a n2kManifest object

Usage

```
read_manifest(base, project, hash)

## S4 method for signature 'character'
read_manifest(base, project, hash)

## S4 method for signature 's3_bucket'
read_manifest(base, project, hash)
```

Arguments

<code>base</code>	The base location to read the manifest.
<code>project</code>	Will be a relative path within the base location.
<code>hash</code>	Optional the sha1 of the manifest. This can be abbreviated to to first unique characters. The function will return an error in case of multiple matches. If missing, then most recent manifest will be returned.

read_model	<i>Read an n2kModel object</i>
------------	--------------------------------

Description

Read an n2kModel object

Usage

```
read_model(x, base, project)

## S4 method for signature 'ANY,character'
read_model(x, base, project)

## S4 method for signature 'ANY,s3_bucket'
read_model(x, base, project)

## S4 method for signature 'ANY,ANY'
read_model(x, base, project)
```

Arguments

x	the file fingerprint of the n2kModel
base	the base location to read the model
project	will be a relative path within the base location

read_result	<i>Read an n2kResult object</i>
-------------	---------------------------------

Description

Read an n2kResult object

Usage

```
read_result(x, base, project)

## S4 method for signature 'ANY,character'
read_result(x, base, project)

## S4 method for signature 'ANY,s3_bucket'
read_result(x, base, project)

## S4 method for signature 'ANY,ANY'
read_result(x, base, project)
```

Arguments

x	the file fingerprint of the n2kResult
base	the base location to read the results
project	will be a relative path within the base location

result_estimate	<i>Aggregate all results estimates in a single dataframe</i>
-----------------	--

Description

Aggregate all results estimates in a single dataframe

Usage

```
result_estimate(x, ...)

## S4 method for signature 'character'
result_estimate(x, ..., base, project)

## S4 method for signature 'n2kResult'
result_estimate(x, ...)
```

Arguments

x	object with the current results
...	further arguments (see Details)
base	the base location to read the results
project	will be a relative path within the base location

result_metadata	<i>Aggregate all results meta data in a single dataframe</i>
-----------------	--

Description

Aggregate all results meta data in a single dataframe

Usage

```
result_metadata(x, ...)

## S4 method for signature 'character'
result_metadata(x, ..., base, project)

## S4 method for signature 'n2kResult'
result_metadata(x, ...)
```

Arguments

x	object with the current results
...	further arguments (see Details)
base	the base location to read the results
project	will be a relative path within the base location

```
select_factor_count_non_zero
```

Select data based on the number of presences per category

Description

Presences have Count > 0.

Usage

```
select_factor_count_non_zero(
  observation,
  variable,
  threshold,
  relative = FALSE,
  dimension = 1
)
```

Arguments

observation	the data.frame with observations
variable	the name of the factor
threshold	the minimal threshold
relative	When FALSE the threshold is the number of non-zero observations. When TRUE the threshold is the proportion of non-zero observations. Defaults to FALSE.
dimension	Indicates which element of variable is used for the final aggregation.

Examples

```
observation <- data.frame(
  Count = c(4, 4, 4, 4, 3, 3, 3, 0, 2, 2, 0, 0),
  LocationID = rep(1:3, each = 4),
  Year = rep(c(1, 1, 1, 1, 2, 2), 2)
)
# Select the locations with at least 3 presences
select_factor_count_non_zero(
  observation,
  variable = "LocationID",
  threshold = 3
)
```

```
)  
# Select those locations in which the species is present in at least 2 years  
select_factor_count_non_zero(  
  observation, variable = c("LocationID", "Year"), threshold = 2  
)  
# Select those years in which the species is present in at least 2 locations  
select_factor_count_non_zero(  
  observation, variable = c("LocationID", "Year"),  
  threshold = 2,  
  dimension = 2  
)
```

select_factor_threshold

Select the observations based on the average of a factor

Description

The negative binomial average of the Count variable is calculated for each level of variable. Only the levels which are equal or larger than threshold times the maximal average (in the original scale) are retained.

Usage

```
select_factor_threshold(observation, variable, threshold)
```

Arguments

observation	the data.frame with observations
variable	the name of the factor
threshold	the minimal threshold

Examples

```
observation <- data.frame(  
  Count = c(100, 101, 50, 51, 1, 0, 0, 0),  
  LocationID = factor(rep(1:4, each = 2))  
)  
select_factor_threshold(observation, "LocationID", threshold = 0.05)
```

`select_observed_range` *Limit the observation to the range in which the species is present*

Description

Limit the observation to the range in which the species is present

Usage

```
select_observed_range(observation, variable)
```

Arguments

`observation` the data.frame with observations
`variable` the name of the factor

Examples

```
observation <- data.frame(
  Count = c(0, 0, 100, 101, 0, 51, 1, 0, 0, 0),
  Year = 1:10
)
select_observed_range(observation, "Year")
```

`session_package` *Convert a sessionInfo() to a data.frame of packages*

Description

Convert a sessionInfo() to a data.frame of packages

Usage

```
session_package(session)

## S4 method for signature 'sessionInfo'
session_package(session)
```

Arguments

`session` The output of sessionInfo()

Value

a data.frame with the packages of a sessionInfo()

spde	<i>Create an Spde object</i>
------	------------------------------

Description

Create an Spde object

Usage

```
spde(coordinates, range, sigma)
```

Arguments

coordinates	a data.frame of coordinates use to define the mesh.
range	a numeric vector of length 2. Will be used as the prior.range argument of [INLA::inla.spde2.pcmatern].
sigma	a numeric vector of length 2. Will be used as the prior.sigma argument of [INLA::inla.spde2.pcmatern].

Spde-class	<i>The spde class</i>
------------	-----------------------

Description

It holds the coordinates, range and sigma parameters for the SPDE model.

Slots

Coordinates	a data.frame with the coordinates used for the mesh.
Range	a numeric vector of length 2. Will be used as the prior.range argument of [INLA::inla.spde2.pcmatern].
Sigma	a numeric vector of length 2. Will be used as the prior.sigma argument of [INLA::inla.spde2.pcmatern].

spde2matern	<i>Convert an Spde object to a matern object</i>
-------------	--

Description

Convert an Spde object to a matern object

Usage

```
spde2matern(object)
```

Arguments

object	The Spde object
--------	-----------------

spde2matern, Spde-method

Convert an Spde object to a mesh object

Description

Convert an Spde object to a mesh object

Usage

```
## S4 method for signature 'Spde'  
spde2matern(object)
```

```
spde2mesh(object)
```

```
## S4 method for signature 'Spde'  
spde2mesh(object)
```

Arguments

object The Spde object

status

Get the status of an n2kModel

Description

Get the status of an n2kModel

Usage

```
status(x)
```

```
## S4 method for signature 'n2kAnalysisMetadata'  
status(x)
```

```
## S4 method for signature 'character'  
status(x)
```

Arguments

x the n2kModel object

Value

the status of the object

status<-	<i>Overwrite the status of an n2kAnalysisMetadata</i>
----------	---

Description

Overwrite the status of an n2kAnalysisMetadata

Usage

```
status(x) <- value

## S4 replacement method for signature 'n2kInla'
status(x) <- value

## S4 replacement method for signature 'n2kComposite'
status(x) <- value

## S4 replacement method for signature 'n2kInlaComparison'
status(x) <- value

## S4 replacement method for signature 'n2kAggregate'
status(x) <- value

## S4 replacement method for signature 'n2kModelImputed'
status(x) <- value

## S4 replacement method for signature 'n2kHurdleImputed'
status(x) <- value
```

Arguments

x	the n2kAnalysisMetadata object
value	the new values for the status

store_manifest	<i>Store an n2kManifest object</i>
----------------	------------------------------------

Description

Store an n2kManifest object

Usage

```
store_manifest(x, base, project, overwrite = FALSE)

## S4 method for signature 'ANY,character'
store_manifest(x, base, project, overwrite = FALSE)

## S4 method for signature 'ANY,s3_bucket'
store_manifest(x, base, project, overwrite = FALSE)
```

Arguments

x	the n2kManifest
base	the base location to store the manifest
project	will be a relative path within the base location
overwrite	Should an existing object be overwritten? Defaults to TRUE.

store_manifest_yaml	<i>Store a Docker configuration</i>
---------------------	-------------------------------------

Description

Store a Docker configuration

Usage

```
store_manifest_yaml(x, base, project, docker, dependencies, overwrite = FALSE)

## S4 method for signature 'ANY,s3_bucket'
store_manifest_yaml(x, base, project, docker, dependencies, overwrite = FALSE)

## S4 method for signature 'ANY,character'
store_manifest_yaml(x, base, project, docker, dependencies, overwrite = FALSE)
```

Arguments

x	the n2kManifest
base	the base location to store the manifest
project	will be a relative path within the base location
docker	the docker image to use
dependencies	extra GitHub packages to install
overwrite	Should an existing object be overwritten? Defaults to TRUE.

store_model	<i>Store an n2kModel object</i>
-------------	---------------------------------

Description

Store an n2kModel object

Usage

```
store_model(x, base, project, overwrite = TRUE, validate = TRUE)

## S4 method for signature 'ANY,character'
store_model(x, base, project, overwrite = TRUE, validate = TRUE)

## S4 method for signature 'ANY,s3_bucket'
store_model(x, base, project, overwrite = TRUE, validate = TRUE)
```

Arguments

x	The n2kModel.
base	The base location to store the model.
project	Will be a relative path within the base location.
overwrite	Should an existing object be overwritten? Defaults to TRUE.
validate	Check that the object is valid before storing it. Defaults to TRUE.

union	<i>Make a union of AnalysisVersions</i>
-------	---

Description

Make a union of AnalysisVersions

Usage

```
union(...)
```

Arguments

...	the n2kAnalysisVersions to union
-----	----------------------------------

validObject,character-method

Get the validity of objects in files

Description

Get the validity of objects in files

Usage

```
## S4 method for signature 'character'  
validObject(object, test = FALSE, complete = FALSE)
```

Arguments

object	a single filename, a single path or a vector with filenames.
test	See methods::validObject()
complete	See methods::validObject()

Index

combine, 5
combine, n2kAnalysisVersion-method
 (combine), 5
combine, n2kAnalysisVersion-methods
 (combine), 5
combine, n2kAnomaly-methods (combine), 5
combine, n2kAnomaly-method (combine), 5
combine, n2kParameter-method (combine), 5
combine, n2kParameter-methods (combine),
 5
combine, n2kResult-method (combine), 5
combine, n2kResult-methods (combine), 5
connect_inbo_s3, 5

delete_model, 6
delete_model, ANY, character-method
 (delete_model), 6
delete_model, ANY, s3_bucket-method
 (delete_model), 6
display, 6

extract, 7
extract, ANY, character-method (extract),
 7
extract, ANY, n2kInla-method (extract), 7
extract, ANY, n2kModelImputed-method
 (extract), 7
extract, character-methods (extract), 7
extract, n2kInla-methods (extract), 7
extract, n2kModelImputed-methods
 (extract), 7

fit_every_model, 7
fit_model, 8
fit_model(), 8
fit_model, character-method (fit_model),
 8
fit_model, n2kAggregate-method
 (fit_model), 8

fit_model, n2kComposite-method
 (fit_model), 8
fit_model, n2kHurdleImputed-method
 (fit_model), 8
fit_model, n2kInla-method (fit_model), 8
fit_model, n2kInlaComparison-method
 (fit_model), 8
fit_model, n2kManifest-method
 (fit_model), 8
fit_model, n2kModelImputed-method
 (fit_model), 8
fit_model, n2kSpde-method (fit_model), 8
fit_model, s3_object-method (fit_model),
 8

get_analysis_date, 10
get_analysis_date, n2kAnalysisMetadata-method
 (get_analysis_date), 10
get_analysis_date, n2kModel-methods
 (get_analysis_date), 10
get_analysis_version, 11
get_analysis_version, data.frame-method
 (get_analysis_version), 11
get_analysis_version, n2kAnalysisMetadata-method
 (get_analysis_version), 11
get_analysis_version, n2kAnalysisMetadata-methods
 (get_analysis_version), 11
get_analysis_version, sessionInfo-method
 (get_analysis_version), 11
get_analysis_version, sessionInfo-methods
 (get_analysis_version), 11
get_anomaly, 11
get_anomaly, n2kInla-method
 (get_anomaly), 11
get_anomaly, n2kInla-methods
 (get_anomaly), 11
get_anomaly, n2kModel-method
 (get_anomaly), 11
get_data, 12
get_data, n2kInla-method (get_data), 12

- get_data, n2kInla-methods (get_data), 12
- get_datafield_id, 13
- get_file_fingerprint, 13
- get_file_fingerprint, n2kAnalysisMetadata-methods (get_file_fingerprint), 13
- get_file_fingerprint, n2kAnalysisMetadata-methods (get_file_fingerprint), 13
- get_file_fingerprint, n2kManifest-method (get_file_fingerprint), 13
- get_file_fingerprint, n2kManifest-methods (get_file_fingerprint), 13
- get_formula, 14
- get_formula, n2kAnalysisMetadata-method (get_formula), 14
- get_formula, n2kAnalysisMetadata-methods (get_formula), 14
- get_location_group_id, 14
- get_location_group_id, n2kAnalysisMetadata-method (get_location_group_id), 14
- get_location_group_id, n2kAnalysisMetadata-methods (get_location_group_id), 14
- get_model, 15
- get_model, character-method (get_model), 15
- get_model, character-methods (get_model), 15
- get_model, n2kInla-method (get_model), 15
- get_model, n2kInla-methods (get_model), 15
- get_model_parameter, 15
- get_model_parameter, n2kAggregate-method (get_model_parameter), 15
- get_model_parameter, n2kComposite-method (get_model_parameter), 15
- get_model_parameter, n2kHurdleImputed-method (get_model_parameter), 15
- get_model_parameter, n2kInla-method (get_model_parameter), 15
- get_model_parameter, n2kInlaComparison-method (get_model_parameter), 15
- get_model_parameter, n2kModelImputed-method (get_model_parameter), 15
- get_model_type, 16
- get_model_type, n2kAnalysisMetadata-method (get_model_type), 16
- get_model_type, n2kAnalysisMetadata-methods (get_model_type), 16
- get_parents, 17
- get_result, 17
- get_result, character, character-method (get_result), 17
- get_result, character, s3_bucket-method (get_result), 17
- get_result, n2kInla, ANY-method (get_result), 17
- get_result, n2kManifest, character-method (get_result), 17
- get_result, n2kManifest, s3_bucket-method (get_result), 17
- get_result, n2kModel, ANY-method (get_result), 17
- get_result, s3_object, ANY-method (get_result), 17
- get_scheme_id, 18
- get_scheme_id, n2kAnalysisMetadata-method (get_scheme_id), 18
- get_scheme_id, n2kAnalysisMetadata-methods (get_scheme_id), 18
- get_seed, 19
- get_seed, n2kAnalysisMetadata-method (get_seed), 19
- get_seed, n2kAnalysisMetadata-methods (get_seed), 19
- get_species_group_id, 19
- get_species_group_id, n2kAnalysisMetadata-method (get_species_group_id), 19
- get_species_group_id, n2kAnalysisMetadata-methods (get_species_group_id), 19
- get_status_fingerprint, 20
- get_status_fingerprint, n2kAnalysisMetadata-method (get_status_fingerprint), 20
- get_status_fingerprint, n2kAnalysisMetadata-methods (get_status_fingerprint), 20
- INLA::inla(), 30, 35, 38
- INLA::inla.posterior.sample(), 10
- INLA::inla.qsample(), 10
- inla_inverse, 20
- make_a, 21
- make_a, Spde-method (make_a), 21
- manifest_yaml_to_bash, 21
- manifest_yaml_to_bash, character-method (manifest_yaml_to_bash), 21
- manifest_yaml_to_bash, s3_bucket-method (manifest_yaml_to_bash), 21
- mark_obsolete_dataset, 22

message(), 6
 methods::validObject(), 52
 moving_average, 23
 moving_difference, 23
 moving_trend, 24
 multimpur::aggregate_impute(), 25, 36
 multimpur::model_impute(), 33, 39

 n2k_aggregate, 25
 n2k_aggregate, ANY-method
 (n2k_aggregate), 25
 n2k_aggregate, n2kAggregate-methods
 (n2k_aggregate), 25
 n2k_composite, 26
 n2k_composite, data.frame-method
 (n2k_composite), 26
 n2k_composite, n2kComposite-methods
 (n2k_composite), 26
 n2k_hurdle_imputed, 27
 n2k_hurdle_imputed, n2kHurdleImputed-methods
 (n2k_hurdle_imputed), 27
 n2k_hurdle_imputed, n2kInla-method
 (n2k_hurdle_imputed), 27
 n2k_import, 28
 n2k_import, ANY-method (n2k_import), 28
 n2k_inla, 29
 n2k_inla, data.frame, ANY-method
 (n2k_inla), 29
 n2k_inla, n2kInla, inla-method
 (n2k_inla), 29
 n2k_inla, n2kInla-methods (n2k_inla), 29
 n2k_inla_comparison, 31
 n2k_inla_comparison, data.frame-method
 (n2k_inla_comparison), 31
 n2k_inla_comparison, n2kInlaComparison-methods
 (n2k_inla_comparison), 31
 n2k_manifest, 32
 n2k_manifest, data.frame-method
 (n2k_manifest), 32
 n2k_manifest, n2kManifest-methods
 (n2k_manifest), 32
 n2k_model_imputed, 32
 n2k_model_imputed, ANY-method
 (n2k_model_imputed), 32
 n2k_model_imputed, n2kModelImputed-methods
 (n2k_model_imputed), 32
 n2k_spde, 33
 n2k_spde, data.frame, ANY-method
 (n2k_spde), 33
 n2k_spde, n2kSpde, inla-method
 (n2k_spde), 33
 n2k_spde, n2kSpde-methods (n2k_spde), 33
 n2kAggregate-class, 36
 n2kAnalysisMetadata-class, 36
 n2kAnalysisVersion-class, 36
 n2kAnomaly-class, 36
 n2kComposite-class, 37
 n2kContrast-class, 37
 n2kHurdleImputed-class, 37
 n2kImport-class, 37
 n2kInla-class, 38
 n2kInlaComparison-class, 38
 n2kManifest-class, 38
 n2kModel-class, 39
 n2kModelImputed-class, 39
 n2kParameter-class, 39
 n2kResult-class, 39
 n2kSpde-class, 40

 order_manifest, 40

 parent_status, 40
 parent_status, n2kAnalysisMetadata-method
 (parent_status), 40
 parent_status, n2kAnalysisMetadata-methods
 (parent_status), 40
 parent_status<-, 41
 parent_status<-, n2kComposite-method
 (parent_status<-), 41

 read_manifest, 41
 read_manifest, character-method
 (read_manifest), 41
 read_manifest, s3_bucket-method
 (read_manifest), 41
 read_model, 42
 read_model, ANY, ANY-method (read_model),
 42
 read_model, ANY, character-method
 (read_model), 42
 read_model, ANY, s3_bucket-method
 (read_model), 42
 read_result, 42
 read_result, ANY, ANY-method
 (read_result), 42
 read_result, ANY, character-method
 (read_result), 42

read_result, ANY, s3_bucket-method
 (read_result), 42
 result_estimate, 43
 result_estimate, character-method
 (result_estimate), 43
 result_estimate, n2kResult-method
 (result_estimate), 43
 result_metadata, 43
 result_metadata, character-method
 (result_metadata), 43
 result_metadata, n2kResult-method
 (result_metadata), 43

 select_factor_count_non_zero, 44
 select_factor_threshold, 45
 select_observed_range, 46
 session_package, 46
 session_package, sessionInfo-method
 (session_package), 46
 session_package, sessionInfo-methods
 (session_package), 46
 spde, 47
 Spde-class, 47
 spde2matern, 47
 spde2matern, Spde-method, 48
 spde2mesh (spde2matern, Spde-method), 48
 spde2mesh, Spde-method
 (spde2matern, Spde-method), 48
 status, 48
 status, character-method (status), 48
 status, character-methods (status), 48
 status, n2kAnalysisMetadata-method
 (status), 48
 status, n2kAnalysisMetadata-methods
 (status), 48
 status<-, 49
 status<-, n2kAggregate-method
 (status<-), 49
 status<-, n2kComposite-method
 (status<-), 49
 status<-, n2kHurdleImputed-method
 (status<-), 49
 status<-, n2kInla-method (status<-), 49
 status<-, n2kInlaComparison-method
 (status<-), 49
 status<-, n2kModelImputed-method
 (status<-), 49
 store_manifest, 49

 store_manifest, ANY, character-method
 (store_manifest), 49
 store_manifest, ANY, s3_bucket-method
 (store_manifest), 49
 store_manifest_yaml, 50
 store_manifest_yaml, ANY, character-method
 (store_manifest_yaml), 50
 store_manifest_yaml, ANY, s3_bucket-method
 (store_manifest_yaml), 50
 store_model, 51
 store_model, ANY, character-method
 (store_model), 51
 store_model, ANY, s3_bucket-method
 (store_model), 51

 union, 51

 validObject, character-method, 52
 validObject, character-methods
 (validObject, character-method),
 52