

Package: hibiiscus (via r-universe)

October 31, 2026

Title Modelling framework for hierachical ecological impact analysis

Version 0.1

Maintainer Richard Cornford <cornford@iiasa.ac.at>

Description This package uses principles of hierarchical modelling to assess the impacts of land use and land-use intensity on ecological communities. Data and modelling approach originate from the PREDICTS framework and are adapted for use with alternative land-use categories (legends), integration of additional ecological data, modelling with frequentist and Bayesian approaches, and extended model evaluation.

License CC BY 4.0

URL <https://github.com/iiasa/hibiiscus>,
<https://iiasa.github.io/hibiiscus/>

BugReports <https://github.com/iiasa/hibiiscus/issues>

Depends R(>= 4.1.0)

Imports assertthat (>= 0.2.0), terra (>= 1.9.0), crayon, dplyr (>= 1.0.10), purrr, R6, rlang, tibble, tidyr, utils

Suggests betapart, biscale, brms, car, caret, cowplot, FNN, geometry, geosphere, ggplot2, INLA, inlabru, insight, lme4, marginaleffects, MuMIn, performance, posterior, randomForest, rnatualearth, rnatualearthdata, reshape2, sf (>= 0.9), sp, testthat (>= 3.0.0), ncd4, stars, knitr, rmarkdown, vegan

VignetteBuilder knitr

Additional_repositories <https://inla.r-inla-download.org/R/testing>

Config/testthat/edition 3

Encoding UTF-8

Language en-GB

Roxygen list(markdown = TRUE)

LazyData true

Config/roxygen2/version 8.1.1

Config/pak/sysreqs libgdal-dev gdal-bin libgeos-dev libicu-dev libproj-dev libsqlite3-dev

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

Date/Publication 2026-10-30 21:51:15 UTC

RemoteUrl <https://github.com/iiasa/hibiiscus>

RemoteRef HEAD

RemoteSha 49f112111a4c564f7a7d353023cc518150ebab5b

Contents

hib_calc_aoa	3
hib_calc_div_between	4
hib_calc_div_within	5
hib_calc_site_contrasts	6
hib_create_db	8
hib_create_lu_legend	9
hib_cv_model	10
hib_filter_sites	11
hib_filter_species	12
hib_fit_brms	13
hib_fit_inla	14
hib_fit_lme4	16
hib_join_site_contrasts	18
hib_join_sites	19
hib_join_species	19
hib_load	20
hib_project_model	21
hib_save	23
hib_set_lu_legend	24
hib_update_db	25
HibCV	26
HibDatabase	28
HibLULegend	30
HibModelBRMS	32
HibModelINLA	34
HibModelLME4	37
HibProjection	39
prep_globiom	41

Index	43
--------------	-----------

hib_calc_aoa	<i>hib_calc_aoa: Calculate the Area-of-Applicability.</i>
--------------	---

Description

Calculate the Area-of-Applicability (AOA) given a model, cross-validation output, and the data being used to make projection/prediction.

Usage

```
hib_calc_aoa(mod_obj, cv_obj, proj_data)
```

Arguments

mod_obj	One of; HibModelLME4 , HibModelINLA or HibModelBRMS .
cv_obj	A HibCV object, based on the above mod_obj.
proj_data	A data.frame, SpatRaster or list of SpatRaster, containing covariates for projecting/predicting the model. Predicted responses should not be included.

Details

This function estimates an AOA, for each of the cross-validation approaches supplied. We follow the methods of Meyer and Pebesma 2021, to estimate a dissimilarity index (DI) using model fitting/projection variables, and cross-validation folds to estimate applicability limits. See references for further information.

The returned list includes:

- variable_weights, a data.frame of variable importance weights.
- variable_scale_params, a list of the parameters used to scale each variable.
- train_DI, a list containing DI estimates and thresholds per cross-validation type.
- proj_aoa, the AOA output, where 1 indicates a location within the AOA, 0 outside the AOA, same data type as proj_data.

Value

list

References

- Meyer, H., & Pebesma, E. (2021). Predicting into unknown space? Estimating the area of applicability of spatial prediction models. *Methods in Ecology and Evolution*, 12, 1620–1633. <https://doi.org/10.1111/2041-210X.13650>
- Meyer H, Milà C, Ludwig M, Linnenbrink J (2023). *CAST: 'caret' Applications for Spatial-Temporal Models*. R package, <https://CRAN.R-project.org/package=CAST>.

See Also[hib_cv_model HibCV](#)**Examples**

```
## Not run:
  aoa_res <- hib_calc_aoa(mod_obj, cv_obj, proj_data)

## End(Not run)
```

hib_calc_div_between *Calculate between-site (within study) diversity metrics*

Description

This function calculates a range of compositional dissimilarity/distance measures between sites (within studies) based on taxonomic diversity and for the associated HibDatabase.

Usage

```
hib_calc_div_between(
  obj,
  metrics,
  binary = FALSE,
  rel_abund = TRUE,
  rescale = TRUE,
  baseline_lui = "Primary vegetation_Minimal use"
)
```

Arguments

obj	A HibDatabase object.
metrics	Character vector of dissimilarity/distance metrics. Metrics available via <code>vegan::vegdist</code> : "manhattan", "euclidean", "canberra", "clark", "bray", "kulczynski", "jaccard", "gower", "altGower", "morisita", "horn", "mountford", "raup", "binomial", "chao", "cao", "mahalanobis", "chisq", "chord", "hellinger", "aitchison", "robust.aitchison". Other available metrics: "hypervolume", "cosine", "theta", "baselga_bray_balanced", "baselga_bray_gradient", "baselga_ruzicka_balanced", "baselga_ruzicka_gradient", "horn_dissim", "normalised_euclidean". Metrics can be provided in upper or lower case - internal processing is done using lower-case - and as unambiguous abbreviations. Note: "horn" refers to the "Morisita-Horn" index in the <code>vegan</code> package. "horn_dissim" refers to (1 - Horn overlap), calculated endogenously within this package.
binary	Logical vector - should metric be calculated using occurrence values (TRUE) or abundance (FALSE, default). Should be a single entry, or of the same length as 'metrics'. Vectors of other lengths will be recycled to the length of 'metrics'.

rel_abund	Logical vector - determining whether relative abundance should be used to calculate the dissimilarity metric, only applicable for some metrics, e.g., euclidean, hypervolume. Note: many metrics include rescaling of abundances as part of the calculation. Default values is TRUE.
rescale	Logical vector - determining if dissimilarity measure should be rescaled relative to its maximum value - applies to metrics that are not on a 0-1 scale by default - hypervolume, euclidean (if relative abundance), chord, hellinger. Default values is TRUE.
baseline_lui	Character or NULL, indicating the baseline LUI/s against which compositional dissimilarity is calculated per study. If NULL, compositional dissimilarity is calculated between all sites within each study. Default values is 'Primary vegetation_Minimal use' - the most pristine LUI within the PREDICTS database.

Details

As is standard for R6 Classes, the HibDatabase is modified in-place. If effort correction (`hib_effort_correction()`) has previously been conducted, diversity metrics will be estimated using effort corrected measurements.

Value

HibDatabase with updated '\$diversity_between' tibble

Examples

```
## Not run:
obj <- hib_calc_div_between(obj,
  c("bray", "jaccard"),
  c(F, T),
  c(T, F),
  c(F, F),
  "Primary vegetation_Minimal use")

## End(Not run)
```

hib_calc_div_within	<i>Calculate within-site diversity metrics</i>
---------------------	--

Description

This function calculates multiple, user specified measures of site-level (taxonomic) diversity for sites in the associated HibDatabase.

Usage

```
hib_calc_div_within(obj, metrics)
```

Arguments

obj A [HibDatabase](#) object.

metrics Character vector, any/all of; "Total_abundance", "Species_richness", "Chao1_diversity", "Simpson_diversity", "Rarefied_species_richness", "Mean_abundance", "Geometric_mean_abundance", "Shannon_diversity", "HillShannon_diversity", "Hill-Simpson_diversity", "PIE", "Equalness". Metrics can be provided in upper or lower case and as unambiguous abbreviations (internal processing is done using lower-case).

Details

As is standard for R6 Classes, the HibDatabase is modified in-place. If effort correction (`hib_effort_correction()`) has previously been conducted, diversity metrics will be estimated using effort corrected measurements.

Value

HibDatabase with updated '\$diversity_within' tibble

Examples

```
## Not run:
hib_db <- hib_calc_div_within(hib_db, c("Total_abundance", "Species_richness"))

## End(Not run)
```

hib_calc_site_contrasts

hib_calc_site_contrasts: Calculate differences between sites for a range of environmental variables.

Description

This function calculates measures of differences between sites within a study, which can then be used as covariates in models of compositional dis/similarity. A `data.frame` (named `site_contrasts`) containing the site-contrast pairs (SSBS1, SSBS2) and the associated 'differences' is added to the provided HibDatabase.

Usage

```
hib_calc_site_contrasts(
  obj,
  metrics,
  custom_metric_specs = NULL,
  baseline_lui = "Primary vegetation_Minimal use",
  overwrite = FALSE
)
```

Arguments

obj	A HibDatabase object.
metrics	A character vector of 'distance' metrics. Current options are; "lui_contrast", "geogr_dist_hav", "geogr_dist_geod" and "geogr_midpoint", "env_dist".
custom_metric_specs	<p>A named list (or NULL) with each named item containing the specifications for the named custom metric. At present, custom metrics must be based on vegdist distance metrics.</p> <p>Essential named values within the specifications of a custom metric are:</p> <ul style="list-style-type: none"> • dist_metr - a character indicating the metric of choice; • vars - a character vector indicating the site variables to use when calculating the distance metric. <p>For example:</p> <pre>list(Gower_dist_a = list(dist_metr = "gower", vars = c("env1", "env2", "env3")), Gower_dist_b = list(dist_metr = "gower", vars = c("env4", "env5", "env6")))</pre> <p>Metric names (as specified in the name of the sub-lists) will be converted to sentence-case in the output data.</p>
baseline_lui	A character (can be multiple entries) or NULL, indicating the baseline LUI/s against which differences are calculated per study. Value/s provided should match the value/s used when calculating compositional dis/similarity (hib_calc_div_between).
overwrite	A logical indicating whether or not any current site contrast data should be over-written. Default value is FALSE.

Details

The function currently returns site1-site2 contrasts for all within-study contrasts where site1 has the baseline_lui. Subsequent, filtering to ensure contrasts are based on the same effort level is recommended and may be incorporated in future releases.

Value

HibDatabase

References

https://adrianadepalma.github.io/BII_tutorial

Examples

```
## Not run:
obj <- hib_calc_site_contrasts(obj,
                              c("lui_contrast", "geogr_dist_hav"),
                              baseline_lui = "Primary vegetation_Minimal use")

## End(Not run)
```

hib_create_db	<i>hib_create_db: Create a HibDatabase object from input files</i>
---------------	--

Description

This function creates an [R6](#) HibDatabase object containing the loaded PREDICTS-like database.

Usage

```
hib_create_db(path_species, path_sites = "", dataset_name = "")
```

Arguments

path_species	A character pointing to a PREDICTS formatted abundance/ occurrence database.
path_sites	Optional character argument, pointing to a PREDICTS formatted site database. Default value is "". If "" is provided, site data will be generated from the species data.
dataset_name	A character indicating the dataset used to create the HibDatabase. Default is "". If "" is supplied, the parameter will be internally updated to the filename associated with path_species.

Details

This function assumes that a species database file is explicitly provided, with a site database file being optional. Provided file paths are expected to include a '.rds' extension.

All files are loaded and compiled into a single database object that can be stored for future use (see [hib_save\(\)](#)). This object is then used in other functions of this package to prepare or filter the database and enable ecological impact analyses.

Value

An object of class HibDatabase, containing site and species information which can be further modified for subsequent ecological analysis.

References

- 1. Hudson, L.N., Newbold, T., Contu, S., Hill, S.L.L., Lysenko, I., De Palma, A., Phillips, H.R.P., Senior, R.A., Bennett, D.J., Booth, H., et al. (2014). The PREDICTS database: a global database of how local terrestrial biodiversity responds to human impacts. Ecol. Evol. 4, 4701–4735. 10.1002/ece3.1303.
- 1. Hudson, L.N., Newbold, T., Contu, S., Hill, S.L.L., Lysenko, I., De Palma, A., Phillips, H.R.P., Alhusseini, T.I., Bedford, F.E., Bennett, D.J., et al. (2017). The database of the PREDICTS (Projecting Responses of Ecological Diversity In Changing Terrestrial Systems) project. Ecol. Evol. 7, 145–188. 10.1002/ece3.2579.
- 1. Lawrence Hudson; Tim Newbold; Sara Contu; Samantha L L Hill et al. (2023). The 2016 release of the PREDICTS database V1.1. Natural History Museum. <https://doi.org/10.5519/j4sh7e0w>

- 1. Sara Contu; Adriana De Palma; Rachel Bates; Jessica Borer et al. (2022). Release of data added to the PREDICTS database (November 2022). Natural History Museum. <https://doi.org/10.5519/jg7i52dg>

Examples

```
## Not run:
path_species <- "rawdata/PREDICTS/2016/database.rds"
path_sites <- "rawdata/PREDICTS/2016/sites.rds"
hibDB <- hib_create_db(path_species, path_sites)
hibDB

## End(Not run)
```

hib_create_lu_legend *hib_create_lu_legend: Create a 'new' land-use legend.*

Description

This function creates a [R6 HibLULegend](#) object containing the provided land-use legend (i.e., linking sites, SSBS, to land-uses/intensities).

Usage

```
hib_create_lu_legend(input)
```

Arguments

input	Either a character pointing to a saved rds file, or a data.frame-like object within the environment.
-------	--

Details

The provided data.frame must include the following columns: SSBS, Predominant_land_use, Use_intensity and LUI.

Value

HibLULegend

See Also

[HibLULegend](#)

Examples

```
## Not run:
lu_leg <- hib_create_lu_legend("../Data/landuse_dataframe.rds")

## End(Not run)
```

hib_cv_model	<i>hib_cv_model: Cross-validate a fitted hibiiscus mixed-effects model</i>
--------------	--

Description

This function re-fits models on different subsets of the data, and estimates a range of error metrics allowing for evaluation of model predictive performance.

Usage

```
hib_cv_model(
  mod_obj,
  db_obj,
  fold_type,
  custom_col = NULL,
  random_seed = NULL,
  re_form = list(NA),
  backtransform = NULL,
  use_original_mesh = NULL
)
```

Arguments

mod_obj	One of; HibModelLME4 , HibModelINLA or HibModelBRMS .
db_obj	A HibDatabase object, or, if the above model used multiple data sources, a list of HibDatabase objects.
fold_type	A character indicating the type of cross-validation; one of: "Random-2", "Random-3", "Random-5", "Random-10", "Source_ID", "SS", "Country", "UN_subregion", "UN_region", "Biome", "Realm", "Custom".
custom_col	A character indicating the column in the model data (sites data.frame) to use to define cross-validation folds, when fold_type == "Custom".
random_seed	A numeric indicating the random seed used to define random cross-validation folds - only used if 'Random-X' in fold_type.
re_form	A list indicating the random effects variables to be included in the cross-validation/insample predicitions - if multiple items supplied, each option will be used in a separate cross-validated prediction. Values can be either NA (no random effects), NULL (all random effects), or a rhs formula specifying particular random effects to include. Note: only test set data points with random-effect values present in the training data will have predictions if re_form is not NA.
backtransform	A function to convert the model prediction back to the original scale of interest. The inverse of any transformation used in model fitting.
use_original_mesh	A logical value indicating whether cross-validation of spatial INLA models should use the original mesh for all CV folds. Computationally this is slightly

more efficient, but will impact CV performance metrics as 'withheld' sites will still exist in the mesh.

Details

This function allows for model evaluation using multiple cross-validation strategies, and multiple random effects structures. It returns a HibCV object, which includes the original model, insample and CV performance for multiple evaluation metrics, model coefficients estimated per cross-validation fold.

Value

HibCV

See Also

[HibCV](#)

Examples

```
## Not run:
cv_out <- hib_cv_model(mod_obj, db_obj, fold_type = "Random-10")

## End(Not run)
```

hib_filter_sites	<i>hib_filter_sites: Filter the database object by site-linked variables</i>
------------------	--

Description

This function returns a database object containing only sites (and associated species) whose characteristics are retained.

Usage

```
hib_filter_sites(obj, condition)
```

Arguments

obj	A HibDatabase object.
condition	An expression identifying how to filter the sites data.frame, as would be passed to <code>dplyr::filter</code> .

Value

HibDatabase

Examples

```
## Not run:
hib_obj <- hib_filter_sites(hib_obj, UN_region == "Europe")

## End(Not run)
```

hib_filter_species	<i>hib_filter_species: Filter the database object by species-linked features.</i>
--------------------	---

Description

This function returns a database object containing only species (and associated sites) whose characteristics are retained.

Usage

```
hib_filter_species(obj, condition)
```

Arguments

obj	A HibDatabase object.
condition	An expression identifying how to filter the species data.frame, as would be passed to <code>dplyr::filter</code> .

Value

HibDatabase

Examples

```
## Not run:
hib_obj <- hib_filter_species(hib_obj, Class == "Insecta")

## End(Not run)
```

hib_fit_brms

*hib_fit_brms: Fit a brms style mixed-effects model***Description**

This function fits a brms model using the supplied database, formula and other function arguments tuning the model fitting process.

Usage

```
hib_fit_brms(
  obj,
  form,
  family = stats::gaussian,
  weight_specs = list(weights = NULL, weight_type = NULL),
  scale_specs = list(covars_to_scale = NULL, scale_func = NULL),
  relevel_specs = c(NULL),
  ...
)
```

Arguments

- | | |
|--------------|--|
| obj | A HibDatabase object or a list of them. |
| form | A formula in the style of brms (as in lme4) - including fixed and random effects. |
| family | A family function, e.g. gaussian, poisson - as allowed by brms |
| weight_specs | <p>A named list containing "weights" and "weight_type".</p> <ul style="list-style-type: none"> • weights: a numeric vector containing the relative weights to assign to each dataset/site, should sum to 1. • weight_type: a character, specifying either "dataset" or "site" - i.e. should the weights argument assign relative weights to whole datasets, or for sites in each dataset. <p>Only applicable if a list of HibDatabases are provided.
 The length of 'weights' should be equal to the number of HibDatabases.
 Note: if using this argument, the supplied formula should have a response of the form $y \mid \text{weights}(\text{data_weight}) \sim$. The data_weight variable will be generated within the function call.</p> |
| scale_specs | <p>A named list containing "covars_to_scale" and "scale_func".</p> <ul style="list-style-type: none"> • covars_to_scale is a character vector of numeric covariates that should be scaled prior to modelling. • scale_func is a function that will be used to scale the specified covariates - if NULL, base::scale() will be used. |

`relevel_specs` A vector where each item is: "`<covar_to_relevel_name>`" = "`<base_level>`".

... Any other named arguments that can be passed to `brms::brm()`, e.g., `prior`, `data2`

Details

This function takes the supplied database object (and various other arguments), fits the specified model and returns a `HibModelBRMS` object.

Value

`HibModelBRMS`

References

- Paul-Christian Bürkner (2017). `brms`: An R Package for Bayesian Multilevel Models Using Stan. *Journal of Statistical Software*, 80(1), 1-28. doi:10.18637/jss.v080.i01
- Paul-Christian Bürkner (2018). Advanced Bayesian Multilevel Modeling with the R Package `brms`. *The R Journal*, 10(1), 395-411. doi:10.32614/RJ-2018-017
- Paul-Christian Bürkner (2021). Bayesian Item Response Modeling in R with `brms` and Stan. *Journal of Statistical Software*, 100(5), 1-54. doi:10.18637/jss.v100.i05

See Also

[HibModelBRMS brm](#)

Examples

```
## Not run:
mod <- hib_fit_brms(obj, y ~ x + ( 1 | a ), ...)

## End(Not run)
```

`hib_fit_inla`

hib_fit_inla: Fit an INLA mixed-effects model

Description

This function fits an INLA model using the supplied database, formula and other function arguments tuning the model fitting process

Usage

```
hib_fit_inla(
  obj,
  response_fixed_form,
  random_covar,
  family = "gaussian",
  fit_settings,
  incl_spatial_term = FALSE,
  input_mesh = NULL,
  mesh_settings = list(mesh_dim = "2d", max.edge = c(75, 100), cutoff = 1e-12, offset =
    NULL),
  scale_specs = list(covars_to_scale = NULL, scale_func = NULL),
  relevel_specs = c(NULL),
  prior_list
)
```

Arguments

<code>obj</code>	A HibDatabase object.
<code>response_fixed_form</code>	A formula with the response and fixed effects
<code>random_covar</code>	A formula with random, non-spatial, 'iid' effects specified
<code>family</code>	A character indicating the model family, e.g., "gaussian"
<code>fit_settings</code>	In development
<code>incl_spatial_term</code>	A logical, whether or not to include a spatial random effect
<code>input_mesh</code>	Optionally provide an INLA mesh object
<code>mesh_settings</code>	Optionally specify INLA mash settings, e.g., max.edge
<code>scale_specs</code>	A named list containing "covars_to_scale" and "scale_func". <ul style="list-style-type: none"> • covars_to_scale is a character vector of numeric covariates that should be scaled prior to modelling. • scale_func is a function that will be used to scale the specified covariates - if NULL, base::scale() will be used.
<code>relevel_specs</code>	A vector where each item is: "<covar_to_relevel_name>" = "<base_level>".
<code>prior_list</code>	In development

Details

This function takes the supplied database object (and various other arguments), fits the specified model and returns a HibModelINLA object. Still in development. Weighted data integration currently not implemented.

Value

HibModelINLA

References

- Havard Rue, Sara Martino, and Nicholas Chopin (2009), Approximate Bayesian Inference for Latent Gaussian Models Using Integrated Nested Laplace Approximations (with discussion), Journal of the Royal Statistical Society B, 71, 319-39
- H. Rue, A. Riebler, S. H. Sorbye, J. B. Illian, D. P. Simpson, and F. K. Lindgren. Bayesian computing with INLA: A review. Annual Reviews of Statistics and Its Applications, 4(March):395-421, 2017. URL <http://arxiv.org/abs/1604.00860>

See Also

[HibModelINLA inla](#)

Examples

```
## Not run:
mod <- hib_fit_inla(obj, y ~ -1 + Intercept + x, ~ f(a, model = "iid"))

## End(Not run)
```

hib_fit_lme4

hib_fit_lme4: Fit an lme4 style mixed-effects model

Description

This function fits an lme4 model using the supplied database, formula and other function arguments tuning the model fitting process.

Usage

```
hib_fit_lme4(
  obj,
  form,
  family = stats::gaussian,
  REML = TRUE,
  calc_CIs = NULL,
  optimise_ranefs = FALSE,
  model_dredge = FALSE,
  model_average = FALSE,
  weight_specs = list(weights = NULL, weight_type = NULL),
  scale_specs = list(covars_to_scale = NULL, scale_func = NULL),
  releval_specs = c(NULL),
  ...
)
```

Arguments

obj	A HibDatabase object, or a list of databases.
form	Model formula in the style of lme4 - including fixed and random effects.
family	A family function, e.g. gaussian, poisson - as allowed by lme4. Default is 'gaussian'.
REML	A logical - if using a gaussian family, should the model be fitted using restricted maximum likelihood (REML, T) or maximum likelihood (F) - as in <code>lme4::lmer()</code> .
calc_CIs	An indication of whether fixed effect confidence intervals (95%) should be calculated or not, and if so how - one of NULL, "Wald", "profile", "boot".
optimise_ranefs	Not currently applicable - future dev
model_dredge	Not currently applicable - future dev
model_average	Not currently applicable - future dev
weight_specs	A named list containing "weights" and "weight_type". <ul style="list-style-type: none"> • weights: a numeric vector containing the relative weights to assign to each dataset/site, should sum to 1. • weight_type: a character, specifying either "dataset" or "site" - i.e. should the weights argument assign relative weights to whole datasets, or for sites in each dataset. <p>Only applicable if a list of HibDatabases are provided. The length of 'weights' should be equal to the number of HibDatabases.</p>
scale_specs	A named list containing "covars_to_scale" and "scale_func". <ul style="list-style-type: none"> • covars_to_scale is a character vector of numeric covariates that should be scaled prior to modelling. • scale_func is a function that will be used to scale the specified covariates - if NULL, <code>base::scale()</code> will be used.
relevel_specs	A vector where each item is: "<covar_to_relevel_name>" = "<base_level>".
...	Any other named arguments that can be passed to <code>lme4::lmer()</code> / <code>lme4::glmer()</code> , e.g., <code>na.action</code>

Details

This function takes the supplied database object (and various other arguments), fits the specified model and returns a `HibModelLME4` object.

Value

`HibModelLME4`

References

- Douglas Bates, Martin Maechler, Ben Bolker, Steve Walker (2015). Fitting Linear Mixed-Effects Models Using lme4. Journal of Statistical Software, 67(1), 1-48. doi:10.18637/jss.v067.i01.

See Also

[HibModelLME4](#) [lmer](#) [glmer](#)

Examples

```
## Not run:
mod <- hib_fit_lme4(obj, y ~ x + ( 1 | a ), ...)

## End(Not run)
```

```
hib_join_site_contrasts
```

hib_join_site_contrasts: Join external site-contrast information to a HibDatabase object

Description

This function joins external information to a [HibDatabase](#) object. Such information could be for example additional site-contrast-specific data that are not already present in the database.

Usage

```
hib_join_site_contrasts(obj, site_contrast_info)
```

Arguments

`obj` A [HibDatabase](#) object.

`site_contrast_info` A data.frame or similar of categorical or numeric information to be added to the database. Must share at least one column with `HibDatabase$site_contrasts`. Ideally, both/one of 'SSBS1' and 'SSBS2' are present.

Value

[HibDatabase](#)

Examples

```
## Not run:
hib_db <- hib_join_site_contrasts(hib_db, new_site_contrast_data_df)

## End(Not run)
```

hib_join_sites	<i>hib_join_sites: Join external site information to a HibDatabase object</i>
----------------	---

Description

This function joins external information to a [HibDatabase](#) object. Such information could be for example additional site-specific data that are not already present in the database.

Usage

```
hib_join_sites(obj, site_info)
```

Arguments

obj	A HibDatabase object.
site_info	A data.frame or similar of categorical or numeric information to be added to the site dataframe within the HibDatabase. Must share at least one column with HibDatabase\$sites, e.g., 'SSBS'.

Value

HibDatabase

Examples

```
## Not run:
hib_db <- hib_join_sites(hib_db, new_site_data_df)

## End(Not run)
```

hib_join_species	<i>hib_join_species: Join external species information to a HibDatabase object</i>
------------------	--

Description

This function joins external information to a [HibDatabase](#) object. Such information could be for example prepared data on species traits.

Usage

```
hib_join_species(obj, spp_info)
```

Arguments

obj	A HibDatabase object.
spp_info	A data.frame or similar of categorical or numeric information to be added to the species data in the HibDatabase. Must share at least one column with HibDatabase\$species, e.g. 'Best_guess_binomial'.

Value

HibDatabase

Examples

```
## Not run:
hib_db <- hib_join_species(hib_db, new_spp_data_df)

## End(Not run)
```

hib_load	<i>hib_load: Load Hib___ objects.</i>
----------	---------------------------------------

Description

Load a hibiiscus object from a saved RDS.

Usage

```
hib_load(path)
```

Arguments

path	A character pointing to a file (file names should have a ".rds" extension, if they don't, the extension will be added).
------	---

Details

The function loads the hibiiscus object from the specified file location. If the original object (HibProjection) included SpatRaster data, this will be loaded from the appropriate tif file/s.

Value

A hibiiscus object, one of: HibDatabase, HibLULegend, HibModelLME4, HibModelINLA, HibModelBRMS, HibCV, HibProjection

See Also

[hib_save\(\)](#)

Examples

```
## Not run:
hib_obj <- hib_load("an/example/file_path/hib_model.rds")

## End(Not run)
```

hib_project_model	<i>hib_project_model: Project a hibiiscus mixed-effects model</i>
-------------------	---

Description

This function projects a fitted HibModelXXX using the supplied model object and data to be used in projection.

Usage

```
hib_project_model(
  mod_obj,
  data,
  backtransform = NULL,
  rescale_specs = NULL,
  lme4_specs = NULL,
  inla_specs = NULL,
  brms_specs = NULL,
  calc_aoa = FALSE,
  cv_obj = NULL
)
```

Arguments

mod_obj	One of; HibModelLME4 , HibModelINLA or HibModelBRMS .
data	A data.frame-like object, a SpatRaster, or a list of SpatRaster. Should contain all fixed-effect variables used in the fitted model, and any required random variables. If data is a data.frame, x and y columns should also be included.
backtransform	A function that will backtransform the model response to the scale of the original variable, e.g., exp() as the inverse of log(). Default is NULL, i.e., no back-transformation.
rescale_specs	A named list (or NULL) containing details relevant to rescaling projections. Specifically, named items should consist of: <ul style="list-style-type: none"> • rescale (logical), • baseline_covar (data.frame, or NULL), • baseline_re_form (formula, NA, or NULL).

In combination, these arguments determine if predictions will be rescaled or not (rescale), the covariate values used to generate the baseline against which other predictions are rescaled (baseline_covar), and whether/how random effects should be incorporated in the generation of this baseline (baseline_re_form, see re_form in lme4/inla/brms_specs argument).

If rescale == TRUE, but no baseline_covar is provided, the intercept of the model will be used as the baseline value.

lme4_specs	<p>A named list (or NULL) containing details relevant to projecting a HibModelLME4. Named items should consist of:</p> <ul style="list-style-type: none"> • <code>bootstrap</code> (logical), indicating whether bootstrapped prediction intervals should be estimated. Can be slow for large models. • <code>n_boot</code> (numeric), how many bootstrap samples should be used to estimate prediction uncertainty. • <code>calc_se</code> (logical), specify whether or not a standard error based uncertainty estimate (via <code>marginalEffects::predictions</code>) is calculated and used to generate prediction intervals. • <code>conf_level</code> (numeric) is used to specify the confidence levels of intervals generated via bootstrapping or the SE-based approach. Values should fall between 0 and 1, typically in the range 0.5 to 0.95. Multiple values can be provided. • <code>re_form</code> (formula, NULL, or NA), as <code>re_form</code> in <code>predict.merMod()</code>, used to specify random effects to condition on when predicting.
inla_specs	<p>A named list (or NULL) containing details relevant to projecting a HibModelINLA.</p> <ul style="list-style-type: none"> • <code>re_form</code> (formula, NULL, or NA), as <code>re_form</code> in <code>predict.merMod()</code>, used to specify random effects to condition on when predicting. • <code>link_inverse</code> a function that is the inverse of the link used to fit the model. Used to backtransform the global intercept for use in rescaling predictions.
brms_specs	<p>A named list (or NULL) containing details relevant to projecting a HibModelBRMS. Named items should consist of:</p> <ul style="list-style-type: none"> • <code>incl_obs_uncert</code> (logical), to determine whether or not observation-level uncertainty is included in predictions. Setting this parameter to TRUE will produce predictions using <code>predict.brmsfit()</code> (larger uncertainty), setting to FALSE calls <code>fitted.brmsfit()</code> internally. • <code>conf_level</code> (numeric) is used to specify the confidence levels of intervals generated. Values should fall between 0 and 1, typically in the range 0.5 to 0.95. Internally converted to quantiles and passed to the <code>probs</code> argument of <code>predict.brmsfit()</code>/ <code>fitted.brmsfit()</code>. • <code>re_form</code> (formula, NULL, or NA), as <code>re_formula</code> in <code>predict.brmsfit()</code>, used to specify random effects to condition on when predicting.
calc_aoa	A logical to indicate whether or not the 'Area-of-Applicability' (AOA) should be estimated - following Meyer and Pebesma (2021).
cv_obj	A <code>HibCV</code> object, used in AOA estimation.

Details

Given a HibModelXXX, new data and specification, this function projects/predicts modelled responses for each 'unit' of the new data. Calculation of the 'Area-of-Applicability' of the model within the new data is an optional addition. Specification arguments allow for customisation of the prediction type. If specifications are not provided, defaults will be used. Projections will be returned in the same format as the provided data. If data is provided as SpatRaster, a land-use variable should be included as a single categorical layer, or as multiple, proportional layers, named to align with the levels of the categorical variable. Currently, only a single 'categorical' variable can be provided in 'proportional cover' format in a SpatRaster.

Value

HibProjection

References

- Meyer, H., & Pebesma, E. (2021). Predicting into unknown space? Estimating the area of applicability of spatial prediction models. *Methods in Ecology and Evolution*, 12, 1620–1633. <https://doi.org/10.1111/2041-210X.13650>
- Meyer H, Milà C, Ludwig M, Linnenbrink J (2023). *CAST: 'caret' Applications for Spatial-Temporal Models*. R package, <https://CRAN.R-project.org/package=CAST>.

See Also

[HibProjection](#) [hib_calc_aoa](#) [predict.merMod](#) [fitted.brmsfit](#) [predict.brmsfit](#)

Examples

```
## Not run:
proj <- hib_mod_project(hib_model, proj_data)

## End(Not run)
```

hib_save

hib_save: Save Hib___ objects.

Description

Saves a hibiiscus data object to disk.

Usage

```
hib_save(obj, path, overwrite = FALSE)
```

Arguments

obj	A hibiiscus object, one of: HibDatabase, HibLULegend, HibModelLME4, HibModelINLA, HibModelBRMS, HibCV, HibProjection.
path	A character pointing to a file (file names should have a ".rds" extension. Non-rds extensions will be amended to rds).
overwrite	A logical indicating whether a previously saved file should be overwritten. (Default: FALSE)

Details

The function saves the hibiiscus object in the specified file location. If the object (HibProjection) includes SpatRaster data, this will be separately saved in '.tif' file/s, with the same base-path.

See Also

[hib_load\(\)](#)

Examples

```
## Not run:
  hib_save(model_obj, "an/example/file_path/hib_model.rds", T)

## End(Not run)
```

hib_set_lu_legend	<i>hib_set_lu_legend: Set a HibLULegend as the LU legend in a HibDatabase</i>
-------------------	---

Description

This function adds the HibLULegend to the HibDatabase, modifying both the sites and species data.frames.

Usage

```
hib_set_lu_legend(obj, leg)
```

Arguments

obj	A HibDatabase.
leg	A HibLULegend.

Details

This function updates (and therefore overwrites) the previous Predominant_land_use, Use_intensity and LUI columns in both the sites and species data.frames.

Value

HibDatabase

See Also[HibDatabase](#) [HibLULegend](#)**Examples**

```
## Not run:
obj <- hib_set_lu_legend(obj, leg)

## End(Not run)
```

hib_update_db

*hib_update_db: Update an existing HibDatabase with further data***Description**

This function updates an existing HibDatabase, for example by adding the 2022 version of the PREDICTS data to the 2016 version.

Usage

```
hib_update_db(obj, path_species, path_sites = "", dataset_name = "")
```

Arguments

obj	A HibDatabase object.
path_species	A character pointing to a PREDICTS formatted abundance/ occurrence database.
path_sites	Optional character argument, pointing to a PREDICTS formatted site database. Default value is "". If "" is provided, site data will be generated from the species data.
dataset_name	A character indicating the dataset used to create the HibDatabase. Default is "". If "" is supplied, the parameter will be internally updated to the filename associated with path_species.

Details

Newly provided data (from files) will be appended to the previously created HibDatabase. Species data will be added to the species data.frame, and site data to the site data.frame. Meta-data/descriptors in the HibDatabase will be updated to reflect the additions. Note: Prior filtering/processing of the initial HibDatabase will not be applied to the newly provided data. Please load all data of interest into the HibDatabase prior to further processing.

Value

The updated [HibDatabase](#) object, with the supplied dataset added to its site and species data.

References

- 1. Hudson, L.N., Newbold, T., Contu, S., Hill, S.L.L., Lysenko, I., De Palma, A., Phillips, H.R.P., Senior, R.A., Bennett, D.J., Booth, H., et al. (2014). The PREDICTS database: a global database of how local terrestrial biodiversity responds to human impacts. *Ecol. Evol.* 4, 4701–4735. 10.1002/ece3.1303.
- 1. Hudson, L.N., Newbold, T., Contu, S., Hill, S.L.L., Lysenko, I., De Palma, A., Phillips, H.R.P., Alhusseini, T.I., Bedford, F.E., Bennett, D.J., et al. (2017). The database of the PREDICTS (Projecting Responses of Ecological Diversity In Changing Terrestrial Systems) project. *Ecol. Evol.* 7, 145–188. 10.1002/ece3.2579.
- 1. Lawrence Hudson; Tim Newbold; Sara Contu; Samantha L L Hill et al. (2023). The 2016 release of the PREDICTS database V1.1. Natural History Museum. <https://doi.org/10.5519/j4sh7e0w>
- 1. Sara Contu; Adriana De Palma; Rachel Bates; Jessica Borer et al. (2022). Release of data added to the PREDICTS database (November 2022). Natural History Museum. <https://doi.org/10.5519/jg7i52dg>

Examples

```
## Not run:
path_species <- "rawdata/PREDICTS/2016/database.rds"
path_sites <- "rawdata/PREDICTS/2016/sites.rds"
hibDB <- hib_create_db(path_species, path_sites, "predicts_2016")
path_species_upd <- "rawdata/PREDICTS/2022/database2ndrelease.rds"
hibDB <- update_hib(hibDB, path_species_upd, "", "predicts_2022")
hibDB

## End(Not run)
```

HibCV

HibCV

Description

This is the template object for an [R6](#) class containing the results of cross-validation. It contains functions for summarising and visualising the results.

Details

Template object for cross-validation results

Public fields

`last_update` (character)
Identifies the most recent update to the object (YYYY-MM-DD).

`model_type` (character)
Identifies the type of model; one of "HibModelLME4", "HibModelINLA", "HibModelBRMS".

`original_model_obj` (list)
Contains a list of core model information

`cv_results` (list)
 Contains the list of results from cross-validation

`insample_results` (list)
 Contains the list of results from in-sample validation

`re_form` (list)
 A list of the `re_form` model structures used in evaluation.

`use_original_mesh` (logical or NULL)
 Indicates use of original mesh for spatial INLA models.

Methods

Public methods:

- [HibCV\\$new\(\)](#)
- [HibCV\\$print\(\)](#)
- [HibCV\\$plot\(\)](#)
- [HibCV\\$clone\(\)](#)

`HibCV$new()`: Creates a new instance of this [R6](#) class. Note, this object is constructed via the [hib_cv_model](#) function found within this package.

Usage:

```
HibCV$new(mod_obj, cv_res, insamp_res, re_form, use_orig_mesh)
```

Arguments:

`mod_obj` (HibModelXXXX)
 The model object used in cross-validation.

`cv_res` (list)
 The results of cross-validation.

`insamp_res` (list)
 The results of ins-sample validation.

`re_form` (list)
 The list of random effects structures used in model evaluation.

`use_orig_mesh` (logical or NULL)
 Indicates use of original mesh for spatial INLA models

Returns: (HibCV)

`HibCV$print()`: Base printout function presents core information about the object.

Usage:

```
HibCV$print()
```

Returns: (NULL).

`HibCV$plot()`: Function to generate figures summarising the cross- validation results.

Usage:

```
HibCV$plot(type = "metrics", metrics = c("rmse", "rsq"))
```

Arguments:

`type` (character()) Options are: "coefficients", "coefficients-summary", "obs-v-pred", "obs-v-error", "pred-v-error", "metrics", "map-sites".

`metrics` (character) Options are the lower-case abbreviations of calculated error metrics. - Only needed if `type == "metrics"`.

Returns: (ggplot).

`HibCV$clone()`: The objects of this class are cloneable with this method.

Usage:

`HibCV$clone(deep = FALSE)`

Arguments:

`deep` Whether to make a deep clone.

HibDatabase

HibDatabase

Description

This is the template object for an [R6](#) database containing PREDICTS or PREDICTS-like data. Functions are provided to prepare the data for modelling and visualise the data.

Details

Template objects for databases

Public fields

`dataset_name` (character())

Identifies the dataset used.

`last_update` (character())

Identifies the most recent update to the Database (YYYY-MM-DD).

`paths` (list())

Identifies the files used to create the Database object.

`sites` (data.frame)

Stores information on the sites within the Database.

`species` (data.frame)

Stores information on the species within the Database.

`diversity_within` (tibble)

Stores information on biodiversity metrics calculated for each site (i.e. alpha diversity).

`diversity_between` (tibble)

Stores information on biodiversity metrics calculated between sites, within studies (i.e. beta diversity).

`site_contrasts` (data.frame)

Stores information on 'distances' between sites, within studies (e.g., geographic distance).

`processing_flags` (list())

Stores information on processing that has occurred (e.g., filtering data, adding data).

Methods

Public methods:

- `HibDatabase$new()`
- `HibDatabase$checkself()`
- `HibDatabase$print()`
- `HibDatabase$hib_effort_correction()`
- `HibDatabase$plot()`
- `HibDatabase$clone()`

`HibDatabase$new()`: Creates a new instance of this [R6](#) class. Note, this object is constructed via the `hib_create_db` and `hib_update_db` functions found within this package.

Usage:

```
HibDatabase$new(dataset_name, paths)
```

Arguments:

`dataset_name` (`character()`)

Text indicating the dataset contributing to the database being created.

`paths` (`list()`)

List of paths to the 'species' and 'site' data to be loaded.

Returns: (`HibDatabase`)

`HibDatabase$checkself()`: Self check function to ensure that the site and species data.frames contain the correct columns and align in content.

Usage:

```
HibDatabase$checkself(sites = self$sites, species = self$species)
```

Arguments:

`sites` (`data.frame`).

`species` (`data.frame`).

Returns: (`NULL`).

`HibDatabase$print()`: Base printout function presents core information about the Database object.

Usage:

```
HibDatabase$print()
```

Returns: (`NULL`).

`HibDatabase$hib_effort_correction()`: Function to apply effort correction to the species data. Uses a simple, linear scaling, such that sampling effort of a site is scaled relative to the maximum value per study, and multiplied by the original measure of diversity.

Usage:

```
HibDatabase$hib_effort_correction()
```

Returns: (`HibDatabase`).

`HibDatabase$plot()`: Function to generate figures summarising the site-level data present in the HibDatabase.

Usage:

```
HibDatabase$plot(type = "map-sites")
```

Arguments:

`type` (`character()`) Options are: "map-sites", "map-studies", "biome-N", "ecoregion-N", "realm-N", "class-N".

Returns: (`ggplot`).

`HibDatabase$clone()`: The objects of this class are cloneable with this method.

Usage:

```
HibDatabase$clone(deep = FALSE)
```

Arguments:

`deep` Whether to make a deep clone.

References

De Palma, A., Abrahamczyk, S., Aizen, M. et al. Predicting bee community responses to land-use changes: Effects of geographic and taxonomic biases. *Sci Rep* 6, 31153 (2016). <https://doi.org/10.1038/srep31153>

HibLULegend

HibLULegend

Description

This is the template object for an [R6](#) land-use legend that can be linked to a [HibDatabase](#) object. The legend should contain information on land-use and management/intensity types for sites with the database.

Details

Template object for land-use legend

Public fields

`last_update` (`character()`)

Identifies the most recent update to the legend object (YYYY-MM-DD)

`input_data` (`character()`)

Identifies the source of the input data, either "file ([...])", or "dataframe [...]".

`lu_df` (`data.frame`)

Contains the LU legend dataframe, i.e. A `data.frame` containing columns for "SSBS", "Predominant_land_use", "Use_intensity" and "LUI".

`land_use` (`character()`)

The unique values of `lu_df$Predominant_land_use`.

management (character())
The unique values of lu_df\$Use_intensity
land_use_management (character())
The unique values of lu_df\$LUI

Methods

Public methods:

- [HibLULegend\\$new\(\)](#)
- [HibLULegend\\$checkself\(\)](#)
- [HibLULegend\\$print\(\)](#)
- [HibLULegend\\$plot\(\)](#)
- [HibLULegend\\$clone\(\)](#)

[HibLULegend\\$new\(\)](#): Creates a new instance of this [R6](#) class. Note, this object is constructed via the [hib_create_lu_legend](#) function found within this package.

Usage:

```
HibLULegend$new(input, input_chr)
```

Arguments:

input (character() (rds filepath) or data.frame)

The source data used to generate the LU legend

input_chr (character())

Text indicating the source of the data.

Returns: (HibLULegend)

[HibLULegend\\$checkself\(\)](#): Self check function to ensure that the legend data.frame contains the correct columns.

Usage:

```
HibLULegend$checkself(df = self$lu_df)
```

Arguments:

df (data.frame).

Returns: (NULL).

[HibLULegend\\$print\(\)](#): Base printout function presents core information about the object.

Usage:

```
HibLULegend$print()
```

Returns: (NULL).

[HibLULegend\\$plot\(\)](#): Function to generate figures summarising the data present in the LU legend

Usage:

```
HibLULegend$plot(x_var = "land-use")
```

Arguments:

`x_var` (character()) Options are: "land-use", "use-intensity", "land-use:use-intensity".

Returns: (ggplot).

`HibLULegend$clone()`: The objects of this class are cloneable with this method.

Usage:

`HibLULegend$clone(deep = FALSE)`

Arguments:

`deep` Whether to make a deep clone.

HibModelBRMS

HibModelBRMS

Description

This is the template object for an [R6](#) brms model class. It contains functions for summarising and visualising the model results.

Details

Template object for brms model

Public fields

`dataset_name` (character)

Identifies the dataset used.

`last_update` (character)

Identifies the most recent update to the object (YYYY-MM-DD).

`data` (data.frame)

The data used in model fitting.

`model` (brmsfit)

The brms model object.

`extra_args` (list())

Identifies addition arguments supplied during model fitting (found in '...' when calling `hib_fit_brms`).

`fixedef_coefficients` (data.frame)

The fixed effects coefficients.

`metrics` (list)

List containing model performance metrics.

`processing_flags` (list())

Identifies the processing steps used for the HibDatabase object that was used for model fitting.

`site_locations` (data.frame)

Contains columns for "SSBS", "Longitude", "Latitude"

Methods

Public methods:

- `HibModelBRMS$new()`
- `HibModelBRMS$print()`
- `HibModelBRMS$plot()`
- `HibModelBRMS$check_model()`
- `HibModelBRMS$clone()`

`HibModelBRMS$new()`: Creates a new instance of this [R6](#) class. Note, this object is constructed via the `hib_fit_brms` function found within this package.

Usage:

```
HibModelBRMS$new(
  db_obj,
  mod_df,
  brms_obj,
  fixef_df,
  metr_ls,
  n_spp,
  n_taxa,
  n_sites,
  extra_args
)
```

Arguments:

`db_obj` (HibDatabase, or list)
The HibDatabase object/s used to fit the model.

`mod_df` (data.frame)
The data.frame of the data used in model fitting.

`brms_obj` (brmsfit)
The brms model object.

`fixef_df` (data.frame)
The fixed effects from the fitted model.

`metr_ls` (list)
The list of model performance metrics.

`n_spp` (integer or list)
Number of species contributing to the fitted model.

`n_taxa` (integer or list)
Number of original taxa names contributing to the fitted model.

`n_sites` (integer or list)
Number of sites contributing to the fitted model.

`extra_args` (list)
Additional arguments specified during model fitting.

@return (HibModelBRMS)

`HibModelBRMS$print()`: Base printout function presents core information about the object.

Usage:

HibModelBRMS\$print()

Returns: (NULL).

HibModelBRMS\$plot(): Function to generate figures summarising the model.

Usage:

HibModelBRMS\$plot(type = "fixed-effects")

Arguments:

type (character()) Options are: "fixed-effects", "random-effects", "map-sites".

Returns: (ggplot).

HibModelBRMS\$check_model(): Function to assess the model using the performance package.

Usage:

HibModelBRMS\$check_model()

Returns: (NULL).

HibModelBRMS\$clone(): The objects of this class are cloneable with this method.

Usage:

HibModelBRMS\$clone(deep = FALSE)

Arguments:

deep Whether to make a deep clone.

HibModelINLA

HibModelINLA

Description

This is the template object for an [R6](#) INLA model class. It contains functions for summarising and visualising the model results.

Details

Template object for INLA model

Public fields

dataset_name (character)

Identifies the dataset used.

last_update (character)

Identifies the most recent update to the object (YYYY-MM-DD).

data (data.frame)

The data used in model fitting.

model (inla)

The INLA::inla model object.

`processing_flags` (`list()`)
 Identifies the processing steps used for the HibDatabase object that was used for model fitting.

`fixedef_coefficients` (`data.frame`)
 The fixed effects coefficients.

`user_formula` (`formula` or `list`)
 The model formula supplied by the user (includes the fixed effects and non-spatial random effects).

`inla_fitted_formula` (`formula`)
 The fitted model formula (includes all effects).

`inla_incl_spatial_term` (`logical`)
 TRUE or FALSE, does the model include a spatial term.

`inla_stack` (`inla.data.stack`)
 The data.stack used in model fitting, an INLA object.

`inla_mesh` (`inla.mesh` or `NULL`)
 If used, the mesh used in model fitting.

`inla_mesh_settings` (`list`)
 A list of specifications for generating the mesh, if used in modelling.

`inla_mesh_source` (`character` or `NULL`)
 A description of the mesh source - if not null, can be either "user_provided" or "mesh_settings".

`site_locations` (`data.frame`)
 Contains columns for "SSBS", "Longitude", "Latitude".

Methods

Public methods:

- [HibModelINLA\\$new\(\)](#)
- [HibModelINLA\\$print\(\)](#)
- [HibModelINLA\\$plot\(\)](#)
- [HibModelINLA\\$clone\(\)](#)

`HibModelINLA$new()`: Creates a new instance of this [R6](#) class. Note, this object is constructed via the [hib_fit_inla](#) function found within this package.

Usage:

```
HibModelINLA$new(
  db_obj,
  mod_df,
  inla_obj,
  fixef_df,
  user_formula,
  inla_fitted_formula,
  inla_incl_spatial_term,
  inla_stack,
  inla_mesh,
  inla_mesh_settings,
  inla_mesh_source,
```

```

    n_spp,
    n_taxa,
    n_sites
  )

```

Arguments:

```

db_obj (HibDatabase, or list)
  The HibDatabase object/s used to fit the model.
mod_df (data.frame)
  The data.frame of the data used in model fitting.
inla_obj (inla)
  The inla model object.
fixef_df (data.frame)
  The fixed effects from the fitted model.
user_formula (formula or list)
  The model formula supplied by the user.
inla_fitted_formula (formula)
  The fitted model formula.
inla_incl_spatial_term (logical)
  TRUE or FALSE, does the model include a spatial term.
inla_stack (inla.data.stack)
  The data.stack used in model fitting, an INLA object.
inla_mesh (inla.mesh or NULL)
  If used, the mesh used in model fitting.
inla_mesh_settings (list)
  A list of specifications for generating the mesh.
inla_mesh_source (character or NULL)
  A description of the mesh source.
n_spp (integer)
  Number of species contributing to the fitted model.
n_taxa (integer)
  Number of original taxa names contributing to the fitted model.
n_sites (integer)
  Number of sites contributing to the fitted model.
@return (HibModelINLA)
metr_ls (list)
  The list of model performance metrics.

```

HibModelINLA\$print(): Base printout function presents core information about the object.

Usage:

```
HibModelINLA$print()
```

Returns: (NULL).

HibModelINLA\$plot(): Function to generate figures summarising the model.

Usage:

```
HibModelINLA$plot(type = "fixed-effects")
```

Arguments:

type (character()) Options are: "fixed-effects", "random-effects", "spatial-random-field", "map-sites".

Returns: (ggplot).

HibModelINLA\$clone(): The objects of this class are cloneable with this method.

Usage:

HibModelINLA\$clone(deep = FALSE)

Arguments:

deep Whether to make a deep clone.

HibModelLME4

HibModelLME4

Description

This is the template object for an [R6](#) lme4 model class. It contains functions for summarising and visualising the model results.

Details

Template object for lme4 model

Public fields

dataset_name (character)

Identifies the dataset used.

last_update (character)

Identifies the most recent update to the object (YYYY-MM-DD).

data (data.frame)

The data used in model fitting.

model (merMod)

The lme4 model object.

fixedef_coefficients (data.frame)

The fixed effects coefficients.

ci_type (character, or NA)

Identifies the types of confidence intervals calculated.

metrics (numeric)

Named vector containing model performance metrics.

processing_flags (list())

Identifies the processing steps used for the HibDatabase object that was used for model fitting.

site_locations (data.frame)

Contains columns for "SSBS", "Longitude", "Latitude".

Methods

Public methods:

- [HibModelLME4\\$new\(\)](#)
- [HibModelLME4\\$print\(\)](#)
- [HibModelLME4\\$plot\(\)](#)
- [HibModelLME4\\$check_model\(\)](#)
- [HibModelLME4\\$clone\(\)](#)

[HibModelLME4\\$new\(\)](#): Creates a new instance of this [R6](#) class. Note, this object is constructed via the [hib_fit_lme4](#) function found within this package.

Usage:

```
HibModelLME4$new(
  db_obj,
  mod_df,
  lme4_obj,
  fixef_df,
  ci_type,
  metr_vec,
  n_spp,
  n_taxa,
  n_sites
)
```

Arguments:

`db_obj` (HibDatabase, or list)
The HibDatabase object/s used to fit the model.

`mod_df` (data.frame)
The data.frame of the data used in model fitting.

`lme4_obj` (merMod)
The lme4 model object.

`fixef_df` (data.frame)
The fixed effects from the fitted model.

`ci_type` (character, or NA)
The type of confidence intervals calculated.

`metr_vec` (numeric)
The named vector of model performance metrics.

`n_spp` (integer or list)
Number of species contributing to the fitted model.

`n_taxa` (integer or list)
Number of original taxa names contributing to the fitted model.

`n_sites` (integer or list)
Number of sites contributing to the fitted model.

@return (HibModelLME4)

[HibModelLME4\\$print\(\)](#): Base printout function presents core information about the object.

Usage:

HibModelLME4\$print()

Returns: (NULL).

HibModelLME4\$plot(): Function to generate figures summarising the model.

Usage:

HibModelLME4\$plot(type = "fixed-effects")

Arguments:

type (character()) Options are: "fixed-effects", "random-effects", "map-sites".

Returns: (ggplot).

HibModelLME4\$check_model(): Function to assess the model using the performance package.

Usage:

HibModelLME4\$check_model()

Returns: (NULL).

HibModelLME4\$clone(): The objects of this class are cloneable with this method.

Usage:

HibModelLME4\$clone(deep = FALSE)

Arguments:

deep Whether to make a deep clone.

HibProjection

HibProjection

Description

This is the template object for an [R6](#) class containing the results of projecting a fitted HibModel based on data of interest. It contains functions for summarising and visualising the results.

Details

Template object for projection results

Public fields

last_update (character)

Identifies the most recent update to the object (YYYY-MM-DD).

model (R6)

The model used to project, one of [HibModelLME4](#), [HibModelINLA](#), [HibModelBRMS](#).

projection (data.frame, SpatRaster, or list)

The results of model projection.

projection_specs (list)

List of specifications used for modelling.

aoa (list)

Results of Area-of-Applicability calculations.

Methods

Public methods:

- [HibProjection\\$new\(\)](#)
- [HibProjection\\$print\(\)](#)
- [HibProjection\\$plot\(\)](#)
- [HibProjection\\$clone\(\)](#)

HibProjection\$new(): Creates a new instance of this [R6](#) class. Note, this object is constructed via the [hib_project_model](#) function found within this package.

Usage:

```
HibProjection$new(
  obj,
  proj,
  backtransform,
  rescale_specs,
  lme4_specs,
  inla_specs,
  brms_specs,
  aoa_res
)
```

Arguments:

obj (HibModelXXXX)
The model object used in cross-validation.

proj (data.frame, SpatRaster, or list)
The results of model projection.

backtransform (function or NULL)
Describes the backtransformation specified when making a model projection.

rescale_specs (list)
List of specifications used for rescaling predictions.

lme4_specs (list)
List of specifications used for lme4 models.

inla_specs (list)
List of specifications used for inla models.

brms_specs (list)
List of specifications used for brms models.

aoa_res (list, or NULL)
List of Area-of-Applicability results.

Returns: (HibProjection)

HibProjection\$print(): Base printout function presents core information about the object.

Usage:

```
HibProjection$print()
```

Returns: (NULL).

HibProjection\$plot(): Function to generate figures summarising the projection results.

Usage:

```
HibProjection$plot(
  type = "response-map",
  response_var = NULL,
  uncertainty_var = NULL,
  aoa_mask = FALSE,
  aoa_var = NULL
)
```

Arguments:

type (character()) Options are: "response-map", "uncertainty-map", "bivariate-map", "response-distr", "uncertainty-distr", "bivariate-distr".

response_var (character, or NULL)

uncertainty_var (character, or NULL)

aoa_mask (logical)

aoa_var (character, or NULL)

Returns: (ggplot).

HibProjection\$clone(): The objects of this class are cloneable with this method.

Usage:

```
HibProjection$clone(deep = FALSE)
```

Arguments:

deep Whether to make a deep clone.

```
prep_globiom
```

```
prep_globiom: Prepare GLOBIOM output for projection.
```

Description

Function to format a prepared GLOBIOM netCDF file for use in Hibiiscus projections.

Usage

```
prep_globiom(
  file_path,
  out_as_rast = TRUE,
  incl_pixel_area = TRUE,
  vars_to_include = NULL,
  vars_to_ignore = NULL
)
```

Arguments

- `file_path` A filepath (character) pointing to a GLOBIOM output in netCDF format.
- `out_as_rast` A logical denoting if the output type should be a (list) of `SpatRaster`, or a stars object (default: TRUE, i.e. returns `SpatRaster`).
- `incl_pixel_area` A **logical** denoting if the output should include the 'pixel_area_ha' variable (Default: TRUE).
- `vars_to_include` A character vector of variables to be extracted. If NULL, all variables will be extracted.
- `vars_to_ignore` A character vector of variables to be ignored. If NULL, all variables will be extracted - unless `vars_to_include` is not NULL.

Details

This function expects a downscaled GLOBIOM output. Variables are extracted and converted to raster format, with each raster stack representing a particular year. Likely of little use for anyone outside IIASA.

Value

A (list of) `SpatRaster` stacks with the formatted GLOBIOM predictors.

Examples

```
## Not run:  
# Expects a filename pointing to a globiom-netCDF file.  
globiom_rast <- prep_globiom("path/to/globiom_data.nc")  
  
## End(Not run)
```

Index

- * **AOA**
 - hib_calc_aoa, 3
- * **classes**
 - HibCV, 26
 - HibDatabase, 28
 - HibLULegend, 30
 - HibModelBRMS, 32
 - HibModelINLA, 34
 - HibModelLME4, 37
 - HibProjection, 39
- * **data,**
 - hib_load, 20
 - hib_save, 23
- * **database,**
 - hib_calc_div_between, 4
 - hib_calc_div_within, 5
 - hib_calc_site_contrasts, 6
 - hib_create_db, 8
 - hib_create_lu_legend, 9
 - hib_filter_sites, 11
 - hib_filter_species, 12
 - hib_join_site_contrasts, 18
 - hib_join_sites, 19
 - hib_join_species, 19
 - hib_set_lu_legend, 24
 - hib_update_db, 25
- * **evaluation**
 - hib_cv_model, 10
- * **external**
 - hib_join_site_contrasts, 18
 - hib_join_sites, 19
 - hib_join_species, 19
- * **filtering**
 - hib_filter_sites, 11
 - hib_filter_species, 12
- * **fitting**
 - hib_fit_brms, 13
 - hib_fit_inla, 14
 - hib_fit_lme4, 16
- * **land-use,**
 - hib_create_lu_legend, 9
 - hib_set_lu_legend, 24
- * **legend**
 - hib_create_lu_legend, 9
 - hib_set_lu_legend, 24
- * **load**
 - hib_load, 20
- * **model,**
 - hib_calc_aoa, 3
 - hib_cv_model, 10
 - hib_fit_brms, 13
 - hib_fit_inla, 14
 - hib_fit_lme4, 16
 - hib_project_model, 21
- * **models,**
 - hib_save, 23
- * **preparation**
 - hib_create_db, 8
 - hib_update_db, 25
- * **processing**
 - hib_calc_div_between, 4
 - hib_calc_div_within, 5
 - hib_calc_site_contrasts, 6
- * **projections,**
 - hib_calc_aoa, 3
- * **projections**
 - hib_project_model, 21
- * **save**
 - hib_save, 23
- * **utils**
 - prep_globiom, 41
- brm, 14
- fitted.brmsfit, 23
- fitted.brmsfit(), 22
- glmer, 18
- hib_calc_aoa, 3, 23

hib_calc_div_between, [4](#), [7](#)
hib_calc_div_within, [5](#)
hib_calc_site_contrasts, [6](#)
hib_create_db, [8](#), [29](#)
hib_create_lu_legend, [9](#), [31](#)
hib_cv_model, [4](#), [10](#), [27](#)
hib_filter_sites, [11](#)
hib_filter_species, [12](#)
hib_fit_brms, [13](#), [33](#)
hib_fit_inla, [14](#), [35](#)
hib_fit_lme4, [16](#), [38](#)
hib_join_site_contrasts, [18](#)
hib_join_sites, [19](#)
hib_join_species, [19](#)
hib_load, [20](#)
hib_load(), [24](#)
hib_project_model, [21](#), [40](#)
hib_save, [23](#)
hib_save(), [8](#), [20](#)
hib_set_lu_legend, [24](#)
hib_update_db, [25](#), [29](#)
HibCV, [3](#), [4](#), [11](#), [22](#), [26](#)
HibDatabase, [4](#), [6](#), [7](#), [10–13](#), [15](#), [17–19](#), [25](#),
[28](#), [30](#)
HibLULegend, [9](#), [25](#), [30](#)
HibModelBRMS, [3](#), [10](#), [14](#), [21](#), [32](#), [39](#)
HibModelINLA, [3](#), [10](#), [16](#), [21](#), [34](#), [39](#)
HibModelLME4, [3](#), [10](#), [18](#), [21](#), [37](#), [39](#)
HibProjection, [23](#), [39](#)

inla, [16](#)

lmer, [18](#)
logical, [42](#)

marginaleffects::predictions, [22](#)

predict.brmsfit, [23](#)
predict.brmsfit(), [22](#)
predict.merMod, [23](#)
predict.merMod(), [22](#)
prep_globiom, [41](#)

R6, [8](#), [9](#), [26–35](#), [37–40](#)