

# Package ‘DAST’

June 30, 2026

**Type** Package

**Title** Spatio-Temporal Disaggregation for Maps with Changing Areal Boundaries

**Version** 0.1.0

**Description** Tools for spatio-temporal disaggregation of areal data across multiple time points, including support for changing polygon boundaries. Implements methods for spatially aggregated log-Gaussian Cox process models with changing areal boundaries as described in Ripstein, Brown and Stafford (2026) ``Spatio-Temporal Disaggregation with Changing Areal Boundaries" <[doi:10.48550/arXiv.2606.25074](https://doi.org/10.48550/arXiv.2606.25074)>. Combines polygon-level observations, population rasters and optional covariate rasters to infer fine-scale spatial fields over time. Models can be efficiently fit using 'TMB' (Template Model Builder) and adaptive Gauss-Hermite quadrature for fast approximate inference or via 'tmbstan' for MCMC.

**License** MIT + file LICENSE

**Encoding** UTF-8

**URL** <https://github.com/nripstein/DAST>

**LinkingTo** TMB, RcppEigen

**Imports** aghq, cowplot, disaggregation, fmesher, ggplot2, Matrix, numDeriv, rSPDE, sf, sparseMVN, terra, TMB

**Suggests** knitr, rmarkdown, rstan, testthat (>= 3.0.0), tmbstan

**RoxygenNote** 7.3.3

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**NeedsCompilation** yes

**Author** Noah Ripstein [aut, cre]

**Maintainer** Noah Ripstein <[noah.ripstein@mail.utoronto.ca](mailto:noah.ripstein@mail.utoronto.ca)>

**Repository** CRAN

**Date/Publication** 2026-06-30 19:40:02 UTC

## Contents

disag_model_mmap . . . . .	2
disag_model_mmap_aghq . . . . .	4
disag_model_mmap_mcmc . . . . .	5
disag_model_mmap_tmb . . . . .	7
get_priors . . . . .	8
make_model_object_mmap . . . . .	9
plot.disag_data_mmap . . . . .	10
plot_aggregation_raster . . . . .	11
plot_covariate_raster . . . . .	12
plot_mesh . . . . .	12
plot_polygons . . . . .	13
predict.disag_model_mmap_aghq . . . . .	14
predict.disag_model_mmap_mcmc . . . . .	15
predict.disag_model_mmap_tmb . . . . .	15
prepare_data_mmap . . . . .	16
print.disag_data_mmap . . . . .	17
print.disag_model_mmap_aghq . . . . .	18
print.disag_model_mmap_mcmc . . . . .	18
print.summary.disag_model_mmap_aghq . . . . .	19
print.summary.disag_model_mmap_mcmc . . . . .	19
summary.disag_data_mmap . . . . .	20
summary.disag_model_mmap_aghq . . . . .	20
summary.disag_model_mmap_mcmc . . . . .	21
<b>Index</b>	<b>22</b>

---

disag_model_mmap	<i>Fit a multi-map disaggregation model (via AGHQ, TMB, or MCMC)</i>
------------------	--

---

## Description

Top-level fitting wrapper with engine dispatch and engine-specific argument handling. Engine-specific controls should be supplied via `engine.args`.

## Usage

```
disag_model_mmap(
  data,
  priors = NULL,
  family = "poisson",
  link = "log",
  engine = c("AGHQ", "TMB", "MCMC"),
  time_varying_betas = FALSE,
  fixed_effect_betas = TRUE,
  engine.args = NULL,
  aghq_k = 2,
```

```

    field = TRUE,
    iid = TRUE,
    silent = TRUE,
    starting_values = NULL,
    optimizer = NULL,
    verbose = FALSE,
    ...
  )

```

## Arguments

<code>data</code>	A <code>disag_data_mmap</code> object.
<code>priors</code>	Optional named list of prior overrides.
<code>family</code>	One of "gaussian", "binomial", "poisson", or "negbinomial".
<code>link</code>	One of "identity", "logit", or "log".
<code>engine</code>	Character; one of "AGHQ", "TMB", or "MCMC". The MCMC engine uses <b>tmbstan</b> .
<code>time_varying_betas</code>	Logical; if TRUE, each time point has its own fixed-effect.
<code>fixed_effect_betas</code>	Logical; if TRUE (default), beta coefficients are treated as fixed effects in the AGHQ outer parameter block (current behavior). If FALSE and <code>engine = "AGHQ"</code> , beta coefficients are moved to TMB random effects so they are integrated in the inner Laplace step.
<code>engine.args</code>	Optional named list of engine-specific options. Supported AGHQ keys are <code>aghq_k</code> , <code>optimizer</code> , and <code>outer_derivative_method</code> . Supported TMB keys are <code>iterations</code> , <code>hess_control_parscale</code> , <code>hess_control_ndeps</code> , and <code>outer_derivative_method</code> . <code>outer_derivative_method</code> may be "tmb" (default) or "finite_difference". The finite-difference option affects only the outer fixed/hyperparameter optimization and Hessian; TMB still handles the inner Laplace approximation. Supported MCMC keys are <code>chains</code> , <code>iter</code> , <code>warmup</code> , <code>thin</code> , <code>cores</code> , <code>seed</code> , <code>refresh</code> , <code>laplace</code> , <code>lower</code> , <code>upper</code> , and <code>control</code> . Additional named MCMC keys are passed through to <code>tmbstan::tmbstan()</code> and <code>rstan::sampling()</code> . <code>iter</code> is the total number of Stan iterations, including warmup.
<code>aghq_k</code>	Deprecated at wrapper level; use <code>engine.args = list(aghq_k = ...)</code> . Retained for backward compatibility.
<code>field</code>	Logical; include spatial field?
<code>iid</code>	Logical; include IID polygon effects?
<code>silent</code>	Logical; pass through to engine fit function.
<code>starting_values</code>	Optional named list of starting values.
<code>optimizer</code>	Deprecated at wrapper level; use <code>engine.args = list(optimizer = ...)</code> . Retained for backward compatibility.
<code>verbose</code>	Logical; print runtime diagnostics.
<code>...</code>	Additional arguments. Engine-specific arguments passed via <code>...</code> are deprecated in this wrapper and should be moved to <code>engine.args</code> .

**Value**

A fitted model object of class `disag_model_mmap_tmb`, `disag_model_mmap_aghq`, or `disag_model_mmap_mcmc` (all also inherit `disag_model_mmap`).

---

`disag_model_mmap_aghq` *Fit a multi-map disaggregation model via TMB + AGHQ*

---

**Description**

Builds the TMB ADFun object for a multi-map disaggregation model, then fits the model via AGHQ with desired number of quadrature points.

**Usage**

```
disag_model_mmap_aghq(
  data,
  priors = NULL,
  family = "poisson",
  link = "log",
  time_varying_betas = FALSE,
  fixed_effect_betas = TRUE,
  aghq_k = 1,
  field = TRUE,
  iid = TRUE,
  silent = TRUE,
  starting_values = NULL,
  optimizer = NULL,
  outer_derivative_method = "tmb",
  verbose = FALSE
)
```

**Arguments**

<code>data</code>	A 'disag_data_mmap' object (from 'prepare_data_mmap()').
<code>priors</code>	Optional named list of prior specifications (see internal helper).
<code>family</code>	One of "gaussian", "binomial", "poisson", or "negbinomial".
<code>link</code>	One of "identity", "logit", or "log".
<code>time_varying_betas</code>	Logical; if TRUE, each time point has its own fixed-effect
<code>fixed_effect_betas</code>	Logical; if TRUE (default), beta coefficients are in AGHQ outer parameters. If FALSE, active betas are treated as TMB random effects.
<code>aghq_k</code>	Integer $\geq 1$ : number of quadrature nodes for AGHQ ('1' = Laplace).
<code>field</code>	Logical: include the spatial random field?

iid	Logical: include polygon-specific IID effects?
silent	Logical: if TRUE, suppress TMB's console output.
starting_values	Optional named list of starting parameter values.
optimizer	Optional optimizer name passed to AGHQ control.
outer_derivative_method	Character; "tmb" (default) uses TMB's analytic outer gradient and "finite_difference" uses finite differences of obj\$fn for outer optimization and Hessian construction. The finite-difference option is only supported with optimizer = "nlminb". The inner Laplace approximation is still handled by TMB.
verbose	Logical: if TRUE, print total runtime.

**Value**

An object of class 'disag\_model\_mmap\_aghq' (a list with '\$aghq\_model', '\$data', and '\$model\_setup').

---

disag\_model\_mmap\_mcmc *Fit a multi-map disaggregation model via tmbstan MCMC*

---

**Description**

Builds the shared TMB ADFun object for a multi-map disaggregation model, then samples from it with `tmbstan::tmbstan()`. This engine supports parameter estimation only; prediction is not implemented for MCMC fits.

**Usage**

```
disag_model_mmap_mcmc(
  data,
  priors = NULL,
  family = "poisson",
  link = "log",
  time_varying_betas = FALSE,
  fixed_effect_betas = TRUE,
  chains = 4L,
  iter = 2000L,
  warmup = NULL,
  thin = 1L,
  cores = NULL,
  seed = NULL,
  refresh = NULL,
  laplace = FALSE,
  lower = numeric(0),
  upper = numeric(0),
  control = NULL,
  field = TRUE,
```

```

  iid = TRUE,
  silent = TRUE,
  starting_values = NULL,
  verbose = FALSE,
  ...
)

```

### Arguments

<code>data</code>	A 'disag_data_mmap' object (from 'prepare_data_mmap()').
<code>priors</code>	Optional named list of prior specifications.
<code>family</code>	One of 'gaussian', 'binomial', 'poisson', or 'negbinomial'.
<code>link</code>	One of 'identity', 'logit', or 'log'.
<code>time_varying_betas</code>	Logical; if TRUE, each time point has its own fixed-effect.
<code>fixed_effect_betas</code>	Logical; if TRUE (default), active beta coefficients are sampled as fixed effects. If FALSE, active beta coefficients are included in the TMB random-effect block.
<code>chains</code>	Integer $\geq 1$ ; number of MCMC chains.
<code>iter</code>	Integer $\geq 1$ ; total Stan iterations per chain, including warmup.
<code>warmup</code>	Integer $\geq 0$ and less than <code>iter</code> ; warmup iterations per chain. Defaults to <code>floor(iter / 2)</code> .
<code>thin</code>	Integer $\geq 1$ ; thinning interval.
<code>cores</code>	Integer $\geq 1$ ; number of cores passed to Stan. Defaults to <code>getOption("mc.cores", chains)</code> .
<code>seed</code>	Optional positive integer seed.
<code>refresh</code>	Optional integer $\geq 0$ ; Stan progress refresh interval.
<code>laplace</code>	Logical; passed to <code>tmbstan::tmbstan()</code> . Defaults to FALSE for full posterior sampling.
<code>lower</code>	Numeric lower bounds passed to <code>tmbstan::tmbstan()</code> .
<code>upper</code>	Numeric upper bounds passed to <code>tmbstan::tmbstan()</code> .
<code>control</code>	Optional list passed to <code>rstan::sampling()</code> .
<code>field</code>	Logical: include the spatial random field?
<code>iid</code>	Logical: include polygon-specific IID effects?
<code>silent</code>	Logical: if TRUE, suppress TMB/tmbstan console output.
<code>starting_values</code>	Optional named list of starting parameter values.
<code>verbose</code>	Logical: if TRUE, print total runtime.
<code>...</code>	Additional arguments passed through to <code>tmbstan::tmbstan()</code> and <code>rstan::sampling()</code> .

### Value

An object of class 'disag\_model\_mmap\_mcmc' with components `stanfit`, `obj`, `data`, and `model_setup`.

---

disag\_model\_mmap\_tmb *Fit a multi-map disaggregation model via TMB*

---

### Description

Builds the TMB ADFun object for a multi-map disaggregation model, then fits the model by maximizing the TMB objective and approximates uncertainty via the optimized Hessian.

### Usage

```
disag_model_mmap_tmb(
  data,
  priors = NULL,
  family = "poisson",
  link = "log",
  time_varying_betas = FALSE,
  fixed_effect_betas = TRUE,
  iterations = 1000,
  field = TRUE,
  iid = TRUE,
  hess_control_parscale = NULL,
  hess_control_ndeps = 1e-04,
  outer_derivative_method = "tmb",
  silent = TRUE,
  starting_values = NULL,
  verbose = FALSE
)
```

### Arguments

data	A 'disag_data_mmap' object (from 'prepare_data_mmap()').
priors	Optional named list of prior specifications (see internal helper).
family	One of 'gaussian', 'binomial', 'poisson', or 'negbinomial'.
link	One of 'identity', 'logit', or 'log'.
time_varying_betas	Logical; if TRUE, each time point has its own fixed-effect
fixed_effect_betas	Logical; if TRUE (default), active beta coefficients are treated as fixed effects. If FALSE, active beta coefficients are treated as random effects in the inner Laplace step.
iterations	Integer >= 1: maximum number of optimizer iterations.
field	Logical: include the spatial random field?
iid	Logical: include polygon-specific IID effects?
hess_control_parscale	Optional numeric vector for scaling the Hessian steps.

hess_control_ndeps	Numeric; relative step size for Hessian finite-difference (default 1e-4).
outer_derivative_method	Character; "tmb" (default) uses TMB's analytic outer gradient and "finite_difference" uses finite differences of obj\$fn for outer optimization and Hessian construction. The inner Laplace approximation is still handled by TMB.
silent	Logical: if TRUE, suppress TMB's console output.
starting_values	Optional named list of starting parameter values.
verbose	Logical: if TRUE, print total runtime.

### Value

An object of class 'disag\_model\_mmap\_tmb' (a list with '\$obj', '\$opt', '\$sd\_out', '\$data', and '\$model\_setup').

---

get_priors	<i>Get Default Prior Values for Disaggregation Model</i>
------------	--

---

### Description

Calculates the default Penalized Complexity (PC) prior parameters and Gaussian priors that will be used by `disag_model_mmap()` if the user does not provide overrides.

### Usage

```
get_priors(data)
```

### Arguments

`data` A `disag_data_mmap` object (output from `prepare_data_mmap`).

### Details

The default priors are dynamic and depend on the input data:

- **Range (Rho):** The lower bound `prior_rho_min` is set to 1/3 of the diagonal length of the study area's bounding box.
- **Spatial SD (Sigma):** The upper bound `prior_sigma_max` is set to the coefficient of variation of the polygon response counts.

### Value

A named list of prior specifications.

**Examples**

```

# Create minimal polygon and covariate inputs for one time point.
polygons <- sf::st_sf(
  area_id = 1:2,
  response = c(10, 12),
  geometry = sf::st_sfc(
    sf::st_polygon(list(rbind(c(0, 0), c(1, 0), c(1, 2), c(0, 2), c(0, 0)))),
    sf::st_polygon(list(rbind(c(1, 0), c(2, 0), c(2, 2), c(1, 2), c(1, 0)))),
    crs = 3857
  )
)
covariate <- terra::rast(
  ncols = 2, nrows = 2, xmin = 0, xmax = 2, ymin = 0, ymax = 2,
  crs = "EPSG:3857"
)
terra::values(covariate) <- c(1, 2, 3, 4)

data <- suppressMessages(prepare_data_mmap(
  polygon_shapefile_list = list(polygons),
  covariate_rasters_list = list(covariate),
  make_mesh = FALSE
))

# Inspect defaults and modify a prior for a later model fit.
defaults <- get_priors(data)
defaults[c("prior_rho_min", "prior_sigma_max")]
my_priors <- defaults
my_priors$prior_rho_prob <- 0.05

```

---

make\_model\_object\_mmap

*Build the TMB ADFun object for multi-map disaggregation*


---

**Description**

Internal helper. Converts data, priors, and model settings into the list of inputs required by 'TMB::MakeADFun()'.

**Usage**

```

make_model_object_mmap(
  data,
  priors = NULL,
  family = "gaussian",
  link = "identity",
  time_varying_betas = FALSE,
  fixed_effect_betas = TRUE,
  field = TRUE,
  iid = TRUE,
  silent = TRUE,

```

```

    starting_values = NULL,
    optimizer = NULL,
    verbose = FALSE
  )

```

### Arguments

data	A 'disag_data_mmap' object.
priors	NULL or named list overriding default hyperpriors.
family	One of "gaussian", "binomial", "poisson", "negbinomial".
link	One of "identity", "logit", "log".
time_varying_betas	Logical; if TRUE, each time point has its own fixed-effect
fixed_effect_betas	Logical; if FALSE, active beta coefficients are included in TMB random effects (for AGHQ inner-Laplace treatment).
field	Logical: include spatial field?
iid	Logical: include IID polygon effects?
silent	Logical: pass to 'MakeADFun()' to suppress output.
starting_values	NULL or named list of starting values.
optimizer	Optional; For changing the arguments used in AGHQ.
verbose	Logical: if TRUE, print details throughout including runtime.

### Value

A 'TMB::ADFun' object ready for 'marginal\_laplace\_tmb()'.

---

plot.disag\_data\_mmap *Visual summary plot of prepared data*

---

### Description

Combines polygons, aggregation raster, mesh, and (if present) a covariate into a 2x2 grid.

### Usage

```

## S3 method for class 'disag_data_mmap'
plot(x, y = NULL, ..., covariate = 1, time = 1, max_categories = 10)

```

**Arguments**

x	A 'disag_data_mmap' object.
y	Not used (required for S3 method compatibility).
...	Additional arguments passed to plot_prepare_summary.
covariate	Integer or name of the covariate to display (default = 1).
time	Integer time-slice (default = 1).
max_categories	Maximum number of unique values to consider categorical (default = 10).

**Value**

A ggdraw object (from cowplot) which can be printed.

---

plot\_aggregation\_raster  
*Plot the offset raster*

---

**Description**

Draws the aggregation pixel values used in the fit

**Usage**

```
plot_aggregation_raster(disag_data, time = 1)
```

**Arguments**

disag_data	A 'disag_data_mmap' object.
time	Integer time-slice (default = 1).

**Value**

A ggplot2 object.

---

plot\_covariate\_raster *Plot a single covariate raster*

---

### Description

Renders one layer of the covariate raster stack, preserving the raster's CRS, and coloring by value with a Viridis scale. Automatically detects and handles categorical covariates with appropriate discrete color scales.

### Usage

```
plot_covariate_raster(disag_data, covariate = 1, time = 1, max_categories = 10)
```

### Arguments

disag\_data      A 'disag\_data\_mmap' object.  
 covariate      Integer index or name of the covariate layer.  
 time            Integer time-slice (default = 1).  
 max\_categories   Maximum number of unique values to consider categorical (default = 10).

### Value

A ggplot2 object.

---

plot\_mesh                      *Plot the SPDE mesh with custom outer/inner boundaries*

---

### Description

Plot the SPDE mesh with custom outer/inner boundaries

### Usage

```
plot_mesh(  
  disag_data,  
  edge_col = "grey70",  
  edge_size = 0.2,  
  outer_col = "black",  
  outer_size = 1,  
  inner_col = "blue",  
  inner_size = 1,  
  node_col = "black",  
  node_size = 0.5  
)
```

**Arguments**

disag_data	A 'disag_data_mmap' object.
edge_col	Colour for internal mesh edges (default = "grey70").
edge_size	Line width for those edges (default = 0.2).
outer_col	Colour for the outer perimeter (default = "black").
outer_size	Line width for the outer perimeter (default = 1).
inner_col	Colour for any inner perimeter (default = "blue").
inner_size	Line width for inner perimeter (default = 1).
node_col	Colour for mesh nodes (default = "black").
node_size	Size for mesh nodes (default = 0.5).

**Value**

A ggplot2 object.

---

plot_polygons	<i>Plot polygon response data</i>
---------------	-----------------------------------

---

**Description**

Draws the prepared polygons colored by the response variable, with an optional title.

**Usage**

```
plot_polygons(disag_data, time = 1, show_title = TRUE)
```

**Arguments**

disag_data	A 'disag_data_mmap' object.
time	Integer index of time-slice to plot (default = 1).
show_title	Logical; if TRUE (default), add a title "Response at time X".

**Value**

A ggplot2 object.

---

```
predict.disag_model_mmap_aghq
```

*Predict mean & credible intervals for AGHQ-fitted disaggregation model*

---

### Description

Given a 'disag\_model\_mmap\_aghq' object, draws from the AGHQ marginal, builds per-cell posterior samples, and returns means and credible-interval rasters.

### Usage

```
## S3 method for class 'disag_model_mmap_aghq'
predict(
  object,
  new_data = NULL,
  predict_iid = FALSE,
  N = 1000,
  CI = 0.95,
  verbose = FALSE,
  ...
)
```

### Arguments

object	A 'disag_model_mmap_aghq' fit (from 'disag_model_mmap_aghq').
new_data	Optional covariates for prediction (see helper).
predict_iid	Currently not implemented; must be FALSE.
N	Number of marginal draws to sample (default 1000).
CI	Credible-interval level in (0,1) (default 0.95).
verbose	If TRUE, prints runtime in minutes.
...	Unused.

### Value

An object of class 'disag\_prediction\_mmap\_aghq' containing: - 'mean\_prediction': list of SpatRasters ('prediction', 'field', 'covariates'). - 'uncertainty\_prediction': list with 'predictions\_ci\$lower' & 'upper'.

---

 predict.disag\_model\_mmap\_mcmc

*Prediction guard for MCMC-fitted multi-map disaggregation models*


---

### Description

Prediction is intentionally not implemented for MCMC fits. This method provides a clear error directing users to the parameter-estimation outputs.

### Usage

```
## S3 method for class 'disag_model_mmap_mcmc'
predict(object, ...)
```

### Arguments

object	A fitted 'disag_model_mmap_mcmc' object.
...	Unused.

### Value

This function always errors.

---

 predict.disag\_model\_mmap\_tmb

*Predict for Multi-Map Disaggregation Model fit with TMB*


---

### Description

Predict for Multi-Map Disaggregation Model fit with TMB

### Usage

```
## S3 method for class 'disag_model_mmap_tmb'
predict(object, new_data = NULL, predict_iid = FALSE, N = 100, CI = 0.95, ...)
```

### Arguments

object	A fitted disag_model_mmap_tmb object.
new_data	Optionally, a new SpatRaster (or list of them) for prediction.
predict_iid	Logical. If TRUE, include the polygon iid effect in predictions.
N	Number of Monte Carlo draws for uncertainty estimation.
CI	Credible interval level (default 0.95).
...	Further arguments.

**Value**

An object of class 'disag\_prediction\_mmap' (also a list) with: - 'mean\_prediction': a list containing time-layered 'SpatRaster's named 'time\_<time point>': 'prediction' (response-scale mean prediction), 'field' (spatial-field contribution, or 'NULL' when no field was fitted), 'iid' (polygon IID contribution when requested and supported, otherwise 'NULL'), and 'covariates' (covariate-only linear predictor). - 'uncertainty\_prediction': a list containing 'realisations', a list of one 'SpatRaster' stack per time point with 'N' Monte Carlo draws, and 'predictions\_ci', a list with time-layered 'SpatRaster's 'lower' and 'upper' containing cell-wise credible bounds at level 'CI'.

---

prepare_data_mmap	<i>Prepare multi-map disaggregation data</i>
-------------------	--

---

**Description**

Given lists of polygon sf's, covariate rasters, and aggregation rasters, combines them into a single 'disag\_data\_mmap' object ready for model fitting.

**Usage**

```
prepare_data_mmap(
  polygon_shapefile_list,
  covariate_rasters_list = NULL,
  aggregation_rasters_list = NULL,
  id_var = "area_id",
  response_var = "response",
  categorical_covariate_baselines = NULL,
  sample_size_var = NULL,
  mesh_args = NULL,
  na_action = FALSE,
  make_mesh = TRUE,
  verbose = FALSE
)
```

**Arguments**

polygon_shapefile_list	List of 'sf' polygon objects, one per time point.
covariate_rasters_list	Optional list of 'SpatRaster' stacks; may be NULL.
aggregation_rasters_list	Optional list of 'SpatRaster'; if NULL, uses uniform counts.
id_var	Name of the polygon ID column in each 'sf'.
response_var	Name of the response column.
categorical_covariate_baselines	Named list; names are categorical raster layers and values are baseline levels to drop (either level labels or numeric codes).

sample_size_var	Name of the sample-size column (for binomial models); may be NULL.
mesh_args	Passed to 'build_mesh()'.
na_action	Logical; if TRUE, drop or impute NAs instead of stopping.
make_mesh	Logical; if TRUE, build the spatial mesh over all polygons.
verbose	Logical; if TRUE, print timing info.

**Value**

An object of class 'disag\_data\_mmap' with components including - 'polygon\_data', 'covariate\_data', 'aggregation\_pixels', ... - 'categorical\_covariate\_baselines' (normalized baseline labels) - 'categorical\_covariate\_schema' (internal encoding schema used for fit/predict consistency)

---

print.disag\_data\_mmap *Print method for 'disag\_data\_mmap' objects*

---

**Description**

Displays a brief overview of a multi-map disaggregation dataset: number of time points, total polygons, and total pixels.

**Usage**

```
## S3 method for class 'disag_data_mmap'
print(x, ...)
```

**Arguments**

x	A 'disag_data_mmap' object.
...	Additional arguments (unused).

**Value**

Invisibly returns the original 'disag\_data\_mmap' object.

---

```
print.disag_model_mmap_aghq
```

*Print method for 'disag\_model\_mmap\_aghq' objects*

---

### **Description**

Displays a brief overview of a multi-map disaggregation model: model family, link function, and components included.

### **Usage**

```
## S3 method for class 'disag_model_mmap_aghq'
print(x, ..., max_print = 30)
```

### **Arguments**

<code>x</code>	A 'disag_model_mmap_aghq' object.
<code>...</code>	Additional arguments (not used).
<code>max_print</code>	Maximum number of random effects details to print.

### **Value**

Invisibly returns the original 'disag\_model\_mmap\_aghq' object.

---

```
print.disag_model_mmap_mcmc
```

*Print method for 'disag\_model\_mmap\_mcmc' objects*

---

### **Description**

Displays a brief overview of a multi-map disaggregation model fit with the MCMC engine.

### **Usage**

```
## S3 method for class 'disag_model_mmap_mcmc'
print(x, ...)
```

### **Arguments**

<code>x</code>	A 'disag_model_mmap_mcmc' object.
<code>...</code>	Additional arguments (unused).

### **Value**

Invisibly returns the original object.

---

```
print.summary.disag_model_mmap_aghq
    Print method for 'summary.disag_model_mmap_aghq' objects (direct
    approach)
```

---

**Description**

Displays the summary information for a multi-map disaggregation model in a well-formatted way, directly using the AGHQ model's summary information.

**Usage**

```
## S3 method for class 'summary.disag_model_mmap_aghq'
print(x, ...)
```

**Arguments**

x                    A 'summary.disag\_model\_mmap\_aghq' object.  
 ...                  Additional arguments (not used).

**Value**

Invisibly returns the original summary object.

---

```
print.summary.disag_model_mmap_mcmc
    Print method for 'summary.disag_model_mmap_mcmc' objects
```

---

**Description**

Displays parameter estimates and MCMC diagnostics for an MCMC-fitted disaggregation model.

**Usage**

```
## S3 method for class 'summary.disag_model_mmap_mcmc'
print(x, ..., max_print = 30)
```

**Arguments**

x                    A 'summary.disag\_model\_mmap\_mcmc' object.  
 ...                  Additional arguments (unused).  
 max\_print           Maximum number of parameter rows to print.

**Value**

Invisibly returns the original summary object.

---

```
summary.disag_data_mmap
```

*Summary function for disag\_data\_mmap objects*

---

### Description

Prints counts of time points, polygons, pixels, per-time largest/smallest polygon sizes, number of covariates and their summaries and a mesh summary

### Usage

```
## S3 method for class 'disag_data_mmap'
summary(object, ...)
```

### Arguments

object            A 'disag\_data\_mmap' object (from 'prepare\_data\_mmap()').  
 ...              Additional arguments (unused).

### Value

Invisibly returns a list with components: - 'n\_times', 'n\_polygons', 'n\_pixels' - 'per\_time': data.frame with 'time', 'min\_pixels', 'max\_pixels' - 'n\_covariates', 'covariate\_summaries' (named list of summaries) - 'mesh\_nodes', 'mesh\_triangles'

---

```
summary.disag_model_mmap_aghq
```

*Summary method for 'disag\_model\_mmap\_aghq' objects (direct approach)*

---

### Description

Creates a simplified summary of a multi-map disaggregation model fit with AGHQ, directly using the AGHQ model's summary information.

### Usage

```
## S3 method for class 'disag_model_mmap_aghq'
summary(object, ...)
```

### Arguments

object            A 'disag\_model\_mmap\_aghq' object.  
 ...              Additional arguments (not used).

**Value**

An object of class 'summary.disag\_model\_mmap\_ghq' containing the summary information.

---

```
summary.disag_model_mmap_mcmc
```

*Summary method for 'disag\_model\_mmap\_mcmc' objects*

---

**Description**

Summarizes parameter estimates and MCMC diagnostics from the underlying stanfit returned by tmbstan::tmbstan().

**Usage**

```
## S3 method for class 'disag_model_mmap_mcmc'  
summary(object, pars = NULL, probs = c(0.025, 0.5, 0.975), ...)
```

**Arguments**

object	A 'disag_model_mmap_mcmc' object.
pars	Optional parameter names passed to summary.stanfit().
probs	Numeric vector of quantile probabilities.
...	Additional arguments passed to summary.stanfit().

**Value**

An object of class 'summary.disag\_model\_mmap\_mcmc'.

# Index

## \* external

- get\_priors, [8](#)
- make\_model\_object\_mmap, [9](#)
  
- disag\_model\_mmap, [2](#)
- disag\_model\_mmap\_aghq, [4](#)
- disag\_model\_mmap\_mcmc, [5](#)
- disag\_model\_mmap\_tmb, [7](#)
  
- get\_priors, [8](#)
  
- make\_model\_object\_mmap, [9](#)
  
- plot.disag\_data\_mmap, [10](#)
- plot\_aggregation\_raster, [11](#)
- plot\_covariate\_raster, [12](#)
- plot\_mesh, [12](#)
- plot\_polygons, [13](#)
- predict.disag\_model\_mmap\_aghq, [14](#)
- predict.disag\_model\_mmap\_mcmc, [15](#)
- predict.disag\_model\_mmap\_tmb, [15](#)
- prepare\_data\_mmap, [16](#)
- print.disag\_data\_mmap, [17](#)
- print.disag\_model\_mmap\_aghq, [18](#)
- print.disag\_model\_mmap\_mcmc, [18](#)
- print.summary.disag\_model\_mmap\_aghq,  
[19](#)
- print.summary.disag\_model\_mmap\_mcmc,  
[19](#)
  
- summary.disag\_data\_mmap, [20](#)
- summary.disag\_model\_mmap\_aghq, [20](#)
- summary.disag\_model\_mmap\_mcmc, [21](#)