

Package ‘BMEmapping’

July 13, 2026

Type Package

Title Spatial Interpolation using Bayesian Maximum Entropy (BME)

Version 2.0.0

Maintainer Kinspride Duah <kinspride2020@gmail.com>

Description Provides an accessible and robust implementation of core BME methodologies for spatial prediction. It enables the systematic integration of heterogeneous data sources including both hard data (precise measurements) and soft interval data (bounded or uncertain observations) while incorporating prior knowledge and supporting variogram-based spatial modeling. The BME methodology is described in Christakos (1990) <doi:10.1007/BF00890661>, Serre and Christakos (1999) <doi:10.1007/s004770050029> and Duah (2025, 2026) <doi:10.1016/j.spasta.2026.100974>.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

URL <https://github.com/KinsprideDuah/BMEmapping>

BugReports <https://github.com/KinsprideDuah/BMEmapping/issues>

RoxygenNote 7.3.3

Suggests dplyr, knitr, rmarkdown, scales, testthat (>= 3.0.0), tidyr

Config/testthat/edition 3

Imports ggplot2, gridExtra, gstat, sf, mvtnorm, stats

Depends R (>= 3.5)

VignetteBuilder knitr

Config/Needs/website rmarkdown

NeedsCompilation no

Author Kinspride Duah [aut, cre, cph] (ORCID:
<<https://orcid.org/0009-0004-3990-7985>>),
Yan Sun [aut]

Repository CRAN

Date/Publication 2026-07-13 07:10:32 UTC

Contents

bme_cv 2

bme_map 4

bme_predict 4

bme_predict_ci 6

casnowload 7

plot.BMEmapping 8

prob_zk 8

q_bme_cv 10

q_bme_predict 11

q_bme_predict_ci 12

q_prob_zk 13

summary.BMEmapping 14

utahsnowload 15

Index 16

bme_cv	<i>Leave-one-out cross validation (LOOCV) at hard data locations.</i>
--------	---

Description

bme_cv performs cross-validation to evaluate the predictive performance of the Bayesian Maximum Entropy (BME) spatial interpolation method using both hard and soft (interval) data. The function supports both leave-one-out cross-validation (LOOCV) and K-fold cross-validation, depending on the value of k. If k equals the number of hard data locations, LOOCV is performed by removing one hard observation at a time and predicting it using the remaining hard and all soft data. If k is less than the number of hard data locations, the hard data are randomly partitioned into k folds, with each fold used once as the validation set while the remaining folds are used for prediction. Depending on the type argument, predictions are returned as posterior means, posterior modes, or posterior medians.

This function is useful for validating the BME interpolation method and tuning variogram parameters.

Usage

```
bme_cv(data_object, model, nugget, sill, range,
       nsmax = 5, nhmax = 5, n = 50,
       zk_range = extended_range(data_object), type,
       k = 5)
```

Arguments

- data_object A list containing the hard and soft data.
- model A string specifying the variogram or covariance model to use (e.g., "exp", "sph", etc.).

nugget	A non-negative numeric value for the nugget effect in the variogram model.
sill	A numeric value representing the sill (total variance) in the variogram model.
range	A positive numeric value for the range (or effective range) parameter of the variogram model.
nsmax	A positive numeric value specifying the maximum number of nearby soft data points to include for estimation (default is 5).
nhmax	A positive numeric value specifying the maximum number of nearby hard data points to include for estimation (default is 5).
n	A positive numeric value indicating the number of points at which to evaluate the posterior density over zk_range (default is 50).
zk_range	A numeric vector specifying the range over which to evaluate the unobserved value at the estimation location (zk). Although zk is unknown, it is assumed to lie within a range similar to the observed data (zh, a, and b). It is advisable to explore the posterior distribution at a few locations using prob_zk() before finalizing this range.
type	A string indicating the type of BME prediction to compute: either "mean" for the posterior mean, "mode" for the posterior mode or "median" for the posterior median.
k	A positive numeric value specifying the number of folds (or partitions) into which the hard data are divided (default is 5).

Value

A data frame containing the coordinates of the hard data locations, the observed values, the corresponding BME predictions (posterior mean, mode, or median, depending on type), the posterior variance (when type = "mean"), the prediction residuals, and the cross-validation fold indices.

Examples

```
data("utsnowload")
ch <- utsnowload[2:10, c("latitude", "longitude")]
cs <- utsnowload[68:232, c("latitude", "longitude")]
zh <- utsnowload[2:10, c("hard")]
a <- utsnowload[68:232, c("lower")]
b <- utsnowload[68:232, c("upper")]
data_object <- bme_map(ch, cs, zh, a, b)
bme_cv(data_object,
  model = "exp", nugget = 0.0953, sill = 0.3639,
  range = 1.0787, type = "mean", k = 5
)
```

bme_map	<i>Create BMEmapping object</i>
---------	---------------------------------

Description

Function that creates BMEmapping objects; objects that hold all the data information necessary for BME interpolation.

Usage

```
bme_map(ch, cs, zh, a, b)
```

Arguments

ch	A two-column matrix of coordinates for hard data locations.
cs	A two-column matrix of coordinates for soft data locations.
zh	A numeric vector of observed values at the hard data locations.
a	A numeric vector of lower bounds for soft interval data.
b	A numeric vector of upper bounds for soft interval data.

Value

A list containing the input arguments ch, cs, zh, a, and b, with class "BMEmapping".

Examples

```
ch <- matrix(c(1, 2, 3, 4), ncol = 2)
cs <- matrix(c(5, 6, 7, 8), ncol = 2)
zh <- c(10, 20)
a <- c(8, 9)
b <- c(12, 13)
obj <- bme_map(ch, cs, zh, a, b)
class(obj) # "BMEmapping"
```

bme_predict	<i>BME prediction</i>
-------------	-----------------------

Description

bme_predict performs BME spatial interpolation at user-specified estimation locations. It uses both hard data (precise measurements) and soft data (interval or uncertain measurements), along with a specified variogram model, to compute either the posterior mean (and its associated variance), mode or median for each location. This function enables spatial prediction in settings where uncertainty in data must be explicitly accounted for, improving estimation accuracy when soft data is available.

Usage

```
bme_predict(x, data_object, model, nugget, sill, range,
            nsmax = 5, nhmax = 5, n = 50,
            zk_range = extended_range(data_object), type)
```

Arguments

<code>x</code>	A two-column matrix of spatial coordinates for the estimation locations.
<code>data_object</code>	A list containing the hard and soft data.
<code>model</code>	A string specifying the variogram or covariance model to use (e.g., "exp", "sph", etc.).
<code>nugget</code>	A non-negative numeric value for the nugget effect in the variogram model.
<code>sill</code>	A numeric value representing the sill (total variance) in the variogram model.
<code>range</code>	A positive numeric value for the range (or effective range) parameter of the variogram model.
<code>nsmax</code>	A positive numeric value specifying the maximum number of nearby soft data points to include for estimation (default is 5).
<code>nhmax</code>	A positive numeric value specifying the maximum number of nearby hard data points to include for estimation (default is 5).
<code>n</code>	A positive numeric value indicating the number of points at which to evaluate the posterior density over <code>zk_range</code> (default is 50).
<code>zk_range</code>	A numeric vector specifying the range over which to evaluate the unobserved value at the estimation location (<code>zk</code>). Although <code>zk</code> is unknown, it is assumed to lie within a range similar to the observed data (<code>zh</code> , <code>a</code> , and <code>b</code>). It is advisable to explore the posterior distribution at a few locations using <code>prob_zk()</code> before finalizing this range.
<code>type</code>	A string indicating the type of BME prediction to compute: either "mean" for the posterior mean, "mode" for the posterior mode or "median" for the posterior median.

Value

A data frame with either 3 or 4 columns, depending on the prediction type. The first two columns contain the geographic coordinates. If `type = "mean"`, the third and fourth columns represent the posterior mean and its associated variance, respectively. If `type = "mode"` or `"median"`, only a third column is returned for the posterior mode or median.

Examples

```
data("utsnowload")
x <- utsnowload[1:3, c("latitude", "longitude")]
ch <- utsnowload[6:67, c("latitude", "longitude")]
cs <- utsnowload[68:232, c("latitude", "longitude")]
zh <- utsnowload[6:67, c("hard")]
a <- utsnowload[68:232, c("lower")]
b <- utsnowload[68:232, c("upper")]
```

```
data_object <- bme_map(ch, cs, zh, a, b)
bme_predict(x, data_object,
  model = "exp", nugget = 0.0953,
  sill = 0.3639, range = 1.0787, type = "mean"
)
```

bme_predict_ci	<i>BME credible interval</i>
----------------	------------------------------

Description

bme_predict_ci computes posterior credible intervals for each estimation location at a user-specified credibility level. The function explicitly incorporates uncertainty in the available data, providing interval estimates that quantify the uncertainty associated with BME spatial predictions.

Usage

```
bme_predict_ci(x, data_object, model, nugget, sill, range,
  nsmax = 5, nhmax = 5, n = 50,
  zk_range = extended_range(data_object), level)
```

Arguments

x	A two-column matrix of spatial coordinates for the estimation locations.
data_object	A list containing the hard and soft data.
model	A string specifying the variogram or covariance model to use (e.g., "exp", "sph", etc.).
nugget	A non-negative numeric value for the nugget effect in the variogram model.
sill	A numeric value representing the sill (total variance) in the variogram model.
range	A positive numeric value for the range (or effective range) parameter of the variogram model.
nsmax	A positive numeric value specifying the maximum number of nearby soft data points to include for estimation (default is 5).
nhmax	A positive numeric value specifying the maximum number of nearby hard data points to include for estimation (default is 5).
n	A positive numeric value indicating the number of points at which to evaluate the posterior density over zk_range (default is 50).
zk_range	A numeric vector specifying the range over which to evaluate the unobserved value at the estimation location (zk). Although zk is unknown, it is assumed to lie within a range similar to the observed data (zh, a, and b). It is advisable to explore the posterior distribution at a few locations using prob_zk() before finalizing this range.
level	A numeric value between 0 and 1 specifying the credible interval level for uncertainty quantification. For example, level = 0.90 computes a 90% credible interval using the posterior distribution.

Value

A data frame whose first two columns contain the geographic coordinates of the estimation locations. The remaining two columns contain the lower and upper bounds of the posterior credible interval corresponding to the specified level. For example, if `level = 0.90`, the returned bounds represent the 90% posterior credible interval.

Examples

```
data("utsnowload")
x <- utsnowload[1:3, c("latitude", "longitude")]
ch <- utsnowload[5:67, c("latitude", "longitude")]
cs <- utsnowload[68:232, c("latitude", "longitude")]
zh <- utsnowload[5:67, c("hard")]
a <- utsnowload[68:232, c("lower")]
b <- utsnowload[68:232, c("upper")]
data_object <- bme_map(ch, cs, zh, a, b)
bme_predict_ci(x, data_object,
  model = "exp", nugget = 0.0953,
  sill = 0.3639, range = 1.0787, level = 0.90
)
```

casnowload

*California Snow Load Data***Description**

A subset of data from the 7964 measurement locations included in the 2020 National Snow Load Study. This data is basically on reliability-targeted snow loads (RTSL) in the state of California.

Usage

casnowload

Format

A data frame with 346 rows and 8 columns.

STATION Name of the snow measuring station

LATITUDE Latitude coordinate position

LONGITUDE Longitude coordinate position

ELEVATION Elevation of the measuring station (measured in meters)

RTSL The hard data RTSL value

LOWER The lower endpoint RTSL

UPPER The upper endpoint RTSL

TYPE Type of snow measurement, WESD is direct and SNWD is indirect measurement. Direct measurements are hard data and have the lower, upper and center values are the same. Indirect measurements have $LOWER < RTSL < UPPER$.

Source

<https://www.ncei.noaa.gov/pub/data/ghcn/daily/>

plot.BEmapping

Plot Method for BEmapping Objects

Description

Generates graphical displays for objects of class "BEmapping". The output depends on the object type:

- Objects returned by bme_map() produce a spatial plot showing hard and soft data locations.
- Objects returned by prob_zk() produce posterior density plots.
- Objects returned by bme_predict() or bme_interval() produce spatial prediction maps.
- Objects returned by bme_cv() produce diagnostic plots for evaluating prediction performance.

Usage

```
## S3 method for class 'BEmapping'
plot(x, ...)
```

Arguments

x An object of class "BEmapping".
 ... Additional arguments (currently unused).

Value

A **ggplot2** object or an arranged collection of plots, returned invisibly.

prob_zk

Posterior Density Estimation at a Single Location

Description

Computes the posterior and plots probability density function (PDF) at a single unobserved spatial location using the Bayesian Maximum Entropy (BME) framework. This function integrates both hard data (precise measurements) and soft data (interval or uncertain observations), together with a specified variogram model, to numerically estimate the posterior density across a range of possible values.

Usage

```
prob_zk(x, data_object, model, nugget, sill, range,
        nsmax = 5, nhmax = 5, n = 50,
        zk_range = extended_range(data_object))
```

Arguments

<code>x</code>	A two-column matrix of spatial coordinates for a single estimation location.
<code>data_object</code>	A list containing the hard and soft data.
<code>model</code>	A string specifying the variogram or covariance model to use (e.g., "exp", "sph", etc.).
<code>nugget</code>	A non-negative numeric value for the nugget effect in the variogram model.
<code>sill</code>	A numeric value representing the sill (total variance) in the variogram model.
<code>range</code>	A positive numeric value for the range (or effective range) parameter of the variogram model.
<code>nsmax</code>	A positive numeric value specifying the maximum number of nearby soft data points to include for estimation (default is 5).
<code>nhmax</code>	A positive numeric value specifying the maximum number of nearby hard data points to include for estimation (default is 5).
<code>n</code>	A positive numeric value indicating the number of points at which to evaluate the posterior density over <code>zk_range</code> (default is 50).
<code>zk_range</code>	A numeric vector specifying the range over which to evaluate the unobserved value at the estimation location (<code>zk</code>). Although <code>zk</code> is unknown, it is assumed to lie within a range similar to the observed data (<code>zh</code> , <code>a</code> , and <code>b</code>). It is advisable to explore the posterior distribution at a few locations using <code>prob_zk()</code> before finalizing this range.

Value

A data frame with two columns: `zk_i` (assumed `zk` values) and `prob_zk_i` (corresponding posterior densities).

Examples

```
data("utsnowload")
x <- utsnowload[1, c("latitude", "longitude")]
ch <- utsnowload[2:67, c("latitude", "longitude")]
cs <- utsnowload[68:232, c("latitude", "longitude")]
zh <- utsnowload[2:67, "hard"]
a <- utsnowload[68:232, "lower"]
b <- utsnowload[68:232, "upper"]
data_object <- bme_map(ch, cs, zh, a, b)
prob_zk(x, data_object,
  model = "exp",
  nugget = 0.0953, sill = 0.3639, range = 1.0787
)
```

q_bme_cv

Leave-one-out cross validation (LOOCV) at hard data locations using the random quantile approach.

Description

bme_cv performs cross-validation to evaluate the predictive performance of the Bayesian Maximum Entropy (BME) spatial interpolation method using both hard and soft (interval) data. The function supports both leave-one-out cross-validation (LOOCV) and K-fold cross-validation, depending on the value of k. If k equals the number of hard data locations, LOOCV is performed by removing one hard observation at a time and predicting it using the remaining hard and all soft data. If k is less than the number of hard data locations, the hard data are randomly partitioned into k folds, with each fold used once as the validation set while the remaining folds are used for prediction. Depending on the type argument, predictions are returned as posterior means, posterior modes, or posterior medians.

This function is useful for validating the BME interpolation method and tuning variogram parameters.

Usage

```
q_bme_cv(data_object, nsmax = 5, nhmax = 5, n = 50, nq = 3,
          zk_range = extended_range(data_object), type,
          k = 5)
```

Arguments

data_object	A list containing the hard and soft data.
nsmax	An integer specifying the maximum number of nearby soft data points to include for estimation (default is 5).
nhmax	An integer specifying the maximum number of nearby hard data points to include for estimation (default is 5).
n	An integer indicating the number of points at which to evaluate the posterior density over zk_range (default is 50).
nq	A positive numeric value for the number of quantile levels (default is 3).
zk_range	A numeric vector specifying the range over which to evaluate the unobserved value at the estimation location (zk). Although zk is unknown, it is assumed to lie within a range similar to the observed data (zh, a, and b). It is advisable to explore the posterior distribution at a few locations using prob_zk() before finalizing this range.
type	A string indicating the type of BME prediction to compute: either "mean" for the posterior mean or "mode" for the posterior mode.
k	A positive numeric value specifying the number of folds (or partitions) into which the hard data are divided (default is 5).

Value

A data frame containing the coordinates of the hard data locations, the observed values, the corresponding BME predictions (posterior mean, mode, or median, depending on type), the posterior variance (when type = "mean"), the prediction residuals, and the cross-validation fold indices.

Examples

```
data("utsnowload")
ch <- utsnowload[2:10, c("latitude", "longitude")]
cs <- utsnowload[68:232, c("latitude", "longitude")]
zh <- utsnowload[2:10, c("hard")]
a <- utsnowload[68:232, c("lower")]
b <- utsnowload[68:232, c("upper")]
data_object <- bme_map(ch, cs, zh, a, b)
q_bme_cv(data_object, type = "mean", k = 5)
```

q_bme_predict

*BME prediction using the random quantile approach***Description**

bme_predict performs BME spatial interpolation at user-specified estimation locations. It uses both hard data (precise measurements) and soft data (interval or uncertain measurements), along with a specified variogram model, to compute either the posterior mean (and its associated variance), mode or median for each location. This function enables spatial prediction in settings where uncertainty in data must be explicitly accounted for, improving estimation accuracy when soft data is available.

Usage

```
q_bme_predict(x, data_object, nsmax = 5, nhmax = 5, n = 50, nq = 3,
              zk_range = extended_range(data_object), type)
```

Arguments

x	A two-column matrix of spatial coordinates for the estimation locations.
data_object	A list containing the hard and soft data.
nsmax	A positive numeric value specifying the maximum number of nearby soft data points to include for estimation (default is 5).
nhmax	A positive numeric value specifying the maximum number of nearby hard data points to include for estimation (default is 5).
n	A positive numeric value indicating the number of points at which to evaluate the posterior density over zk_range (default is 50).
nq	A positive numeric value for the number of quantile levels (default is 3).

zk_range	A numeric vector specifying the range over which to evaluate the unobserved value at the estimation location (zk). Although zk is unknown, it is assumed to lie within a range similar to the observed data (zh, a, and b). It is advisable to explore the posterior distribution at a few locations using prob_zk() before finalizing this range.
type	A string indicating the type of BME prediction to compute: either "mean" for the posterior mean or "mode" for the posterior mode.

Value

A data frame with either 3 or 4 columns, depending on the prediction type. The first two columns contain the geographic coordinates. If type = "mean", the third and fourth columns represent the posterior mean and its associated variance, respectively. If type = "mode" or "median", only a third column is returned for the posterior mode or median.

Examples

```
data("utsnowload")
x <- utsnowload[1:3, c("latitude", "longitude")]
ch <- utsnowload[6:67, c("latitude", "longitude")]
cs <- utsnowload[68:232, c("latitude", "longitude")]
zh <- utsnowload[6:67, c("hard")]
a <- utsnowload[68:232, c("lower")]
b <- utsnowload[68:232, c("upper")]
data_object <- bme_map(ch, cs, zh, a, b)
q_bme_predict(x, data_object, type = "mean")
```

q_bme_predict_ci	<i>BME credible interval</i>
------------------	------------------------------

Description

bme_predict_ci computes posterior credible intervals for each estimation location at a user-specified credibility level using the random quantile approach. The function explicitly incorporates uncertainty in the available data, providing interval estimates that quantify the uncertainty associated with BME spatial predictions.

Usage

```
q_bme_predict_ci(x, data_object, nsmax = 5, nhmax = 5, n = 50, nq = 3,
                 zk_range = extended_range(data_object), level)
```

Arguments

x	A two-column matrix of spatial coordinates for the estimation locations.
data_object	A list containing the hard and soft data.
nsmax	A positive numeric value specifying the maximum number of nearby soft data points to include for estimation (default is 5).

nhmax	A positive numeric value specifying the maximum number of nearby hard data points to include for estimation (default is 5).
n	A positive numeric value indicating the number of points at which to evaluate the posterior density over zk_range (default is 50).
nq	An integer indicating the number of quantile levels (default is 3).
zk_range	A numeric vector specifying the range over which to evaluate the unobserved value at the estimation location (zk). Although zk is unknown, it is assumed to lie within a range similar to the observed data (zh, a, and b). It is advisable to explore the posterior distribution at a few locations using prob_zk() before finalizing this range.
level	A numeric value between 0 and 1 specifying the credible interval level for uncertainty quantification. For example, level = 0.90 computes a 90% credible interval using the posterior distribution.

Value

A data frame whose first two columns contain the geographic coordinates of the estimation locations. The remaining two columns contain the lower and upper bounds of the posterior credible interval corresponding to the specified level. For example, if level = 0.90, the returned bounds represent the 90% posterior credible interval.

Examples

```
data("utsnowload")
x <- utsnowload[1:3, c("latitude", "longitude")]
ch <- utsnowload[5:67, c("latitude", "longitude")]
cs <- utsnowload[68:232, c("latitude", "longitude")]
zh <- utsnowload[5:67, c("hard")]
a <- utsnowload[68:232, c("lower")]
b <- utsnowload[68:232, c("upper")]
data_object <- bme_map(ch, cs, zh, a, b)
q_bme_predict_ci(x, data_object, level = 0.90)
```

q_prob_zk

Posterior Density Estimation at a Single Location

Description

Computes the posterior and plots probability density function (PDF) at a single unobserved spatial location using the Bayesian Maximum Entropy (BME) framework. This function integrates both hard data (precise measurements) and soft data (interval or uncertain observations), together with a specified variogram model, to numerically estimate the posterior density across a range of possible values.

Usage

```
q_prob_zk(x, data_object, nsmax = 5, nhmax = 5, n = 50, nq = 3,
          zk_range = extended_range(data_object))
```

Arguments

<code>x</code>	A two-column matrix of spatial coordinates for a single estimation location.
<code>data_object</code>	A list containing the hard and soft data.
<code>nsmax</code>	A positive numeric value specifying the maximum number of nearby soft data points to include for estimation (default is 5).
<code>nhmax</code>	A positive numeric value specifying the maximum number of nearby hard data points to include for estimation (default is 5).
<code>n</code>	An integer indicating the number of points at which to evaluate the posterior density over <code>zk_range</code> (default is 50).
<code>nq</code>	An integer indicating the number of quantile levels (default is 3).
<code>zk_range</code>	A numeric vector specifying the range over which to evaluate the unobserved value at the estimation location (<code>zk</code>). Although <code>zk</code> is unknown, it is assumed to lie within a range similar to the observed data (<code>zh</code> , <code>a</code> , and <code>b</code>). It is advisable to explore the posterior distribution at a few locations using <code>prob_zk()</code> before finalizing this range

Value

A data frame with two columns: `zk_i` (assumed `zk` values) and `prob_zk_i` (corresponding posterior densities).

Examples

```
data("utsnowload")
x <- utsnowload[1, c("latitude", "longitude")]
ch <- utsnowload[2:67, c("latitude", "longitude")]
cs <- utsnowload[68:232, c("latitude", "longitude")]
zh <- utsnowload[2:67, "hard"]
a <- utsnowload[68:232, "lower"]
b <- utsnowload[68:232, "upper"]
data_object <- bme_map(ch, cs, zh, a, b)
q_prob_zk(x, data_object)
```

summary.BMEmapping	<i>Summary Method for BME Cross-Validation Results</i>
--------------------	--

Description

Provides a concise summary of LOOCV performance for BME predictions at hard data locations. Computes standard residual-based accuracy metrics, including Mean Error (ME), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE).

Usage

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

Arguments

- `object` An object of class "bme_cv", typically returned by a BME cross-validation function. Must contain a numeric vector named `residual`.
- `...` Additional arguments passed to `summary()`.

Value

- A data frame containing three columns:
- `ME` Mean Error (average residual).
 - `MAE` Mean Absolute Error.
 - `RMSE` Root Mean Squared Error.

utahsnowload	<i>A detrended reliability-targeted design ground snow loads in Utah</i>
--------------	--

Description

This dataset contains detrended reliability-targeted design ground snow load measurements from 232 locations in state of Utah. Of these, 65 sites report precise measurements, treated as hard data, while the remaining 167 sites report imprecise measurements, represented as interval (soft) data. The dataset is structured such that the first 67 rows contain hard (point) measurements, and the remaining rows represent soft data using lower and upper interval bounds. For a detailed explanation of the dataset and its use, refer to the related version described in Duah et al. (2025) [doi:10.1016/j.spasta.2025.100894](https://doi.org/10.1016/j.spasta.2025.100894)

Usage

`utsnowload`

Format

- A data frame with 232 rows and 5 variables:
- latitude** Latitude coordinate position
 - longitude** Longitude coordinate position
 - hard** The hard data value
 - lower** The lower endpoint of the soft-interval
 - upper** The upper endpoint of the soft-interval

Source

[doi:10.1016/j.spasta.2025.100894](https://doi.org/10.1016/j.spasta.2025.100894)

Index

* datasets

- casnowload, [7](#)
- utahsnowload, [15](#)

- bme_cv, [2](#)
- bme_map, [4](#)
- bme_predict, [4](#)
- bme_predict_ci, [6](#)

- casnowload, [7](#)

- plot.BMEmapping, [8](#)
- prob_zk, [8](#)

- q_bme_cv, [10](#)
- q_bme_predict, [11](#)
- q_bme_predict_ci, [12](#)
- q_prob_zk, [13](#)

- summary.BMEmapping, [14](#)

- utahsnowload, [15](#)
- utsnowload (utahsnowload), [15](#)