

# Package: intkrige (via r-universe)

September 6, 2024

**Type** Package

**Title** A Numerical Implementation of Interval-Valued Kriging

**Description** An interval-valued extension of ordinary and simple kriging. Optimization of the function is based on a generalized interval distance. This creates a non-differentiable cost function that requires a differentiable approximation to the absolute value function. This differentiable approximation is optimized using a Newton-Raphson algorithm with a penalty function to impose the constraints. Analyses in the package are driven by the 'intsp' and 'intgrd' classes, which are interval-valued extensions of 'SpatialPointsDataFrame' and 'SpatialPixelsDataFrame' respectively. The package includes several wrappers to functions in the 'gstat' and 'sp' packages.

**Version** 1.0.1

**Date** 2020-02-07

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**Imports** Rcpp (>= 1.0.0), Rdpack, methods, stats, utils,

**Suggests** doParallel, foreach, lattice, latticeExtra, grid, gridExtra, testthat, knitr, rmarkdown

**Depends** R (>= 3.5.0), sp (>= 1.3-1), gstat, raster

**RdMacros** Rdpack

**LinkingTo** Rcpp, RcppArmadillo

**RoxygenNote** 7.1.1

**Collate** 'RcppExports.R' 'data.R' 'generics.R' 'gstat-extensions.R' 'intKrigeDriver.R' 'intgrd-class.R' 'intsp-class.R' 'pkgname.R' 'supportFunctions.R' 'unloads.R'

**VignetteBuilder** knitr

**Repository** <https://beanb2.r-universe.dev>

**RemoteUrl** <https://github.com/beanb2/intkrige>

**RemoteRef** HEAD

**RemoteSha** 8d95683fe57efb68ae59959f9647eb13a16df6a3

## Contents

intkrige-package	2
as.data.frame	3
dist_cpp	4
extract	5
fit.intvariogram	6
head.intgrd	6
head.intsp	7
interval	7
interval<-	8
intgrd-class	8
intkrige	9
intsp-class	11
intvariogram	11
intvCheck	12
ohMap	13
ohTemp	13
plot,intgrd,missing-method	14
plot,intsp,missing-method	14
print.intgrd	15
print.intsp	16
print.summary.intgrd	16
print.summary.intsp	17
show,intgrd-method	17
show,intsp-method	18
summary.intgrd	18
summary.intsp	19
tail.intgrd	19
tail.intsp	20
utsnow	20
utsnow_dtl	21
utsnow_dtl2	22
<b>Index</b>	<b>24</b>

## Description

An interval-valued extension of ordinary and simple kriging. Optimization of the function is based on a generalized interval distance. This creates a non-differentiable cost function that requires a differentiable approximation to the absolute value function. This differentiable approximation is optimized using a Newton-Raphson algorithm with a penalty function to impose the constraints. Analyses in the package are driven by the 'intsp' and 'intgrd' classes, which are interval-valued extensions of 'SpatialPointsDataFrame' and 'SpatialPixelsDataFrame' respectively. The package includes several wrappers to functions in the 'gstat' and 'sp' packages.

## Functions

The package contains several generic functions such as [plot](#), [print](#), and [summary](#) to facilitate interval-valued analysis. In addition to these functions, the package also contains the following functions:

- [intkrige](#) Make predictions on interval-valued spatial data using interval-valued kriging.
- [interval<-](#) Create an interval-valued spatial object.
- [intvariogram](#) Simultaneously calculate the empirical variograms for an interval-valued spatial object.
- [fit.intvariogram](#) Automatically fit theoretical variograms to empirical variograms obtained from interval-valued spatial objects.
- [intvCheck](#) Visualize the variogram fits for interval-valued spatial objects.
- [dist\\_cpp](#) A c++ function to calculate great circle or Euclidean distances.

## Data

- [utsnow](#) An interval-valued design ground snow load dataset for Utah.
- [ohtemp](#) An interval-valued 30 year mean temperature dataset for the Ohio River Basin.
- [ohMap](#) A [SpatialPolygons](#) shapefile for the Ohio River Basin.

## Author(s)

Brennan Bean <brennan.bean.20@gmail.com>

---

as.data.frame

*Convert intgrd or intsp object back to a data frame*

---

## Description

Convert intgrd or intsp object back to a data frame

**Usage**

```
## S4 method for signature 'intgrd'
as.data.frame(x)

## S4 method for signature 'intsp'
as.data.frame(x)
```

**Arguments**

x                    An object of class intsp or class intgrd.

**Value**

An object of class data.frame.

---

dist_cpp	<i>Distance matrix calculation.</i>
----------	-------------------------------------

---

**Description**

Calculation of Euclidean or geographic distance. Based on an adaptation of `rdist.earth` in the `fields` package. Uses earth radius = 6378.388 km for great circle distance calculations.

**Usage**

```
dist_cpp(loc1, loc2, geographic)
```

**Arguments**

loc1                A two column matrix representing the longitude/latitude (i.e. x/y) coordinates of the first set of distances.

loc2                A two column matrix representing the longitude/latitude (i.e. x/y) coordinates of the second set of distances (for pairwise distance matrix set `loc1 = loc2`).

geographic        If TRUE, calculate great circle distance. If FALSE, calculate Euclidean distance.

**Details**

Note that while this function is still available to users. It is not actively used in the `intkrige` function. Instead, for user convenience, distances are calculated using distance functions in the `sp` package.

**Value**

A matrix of pairwise distances between each combination of rows from `loc1` and `loc2` respectively.

**Examples**

```
data(utsnow)
locs1 <- as.matrix(utsnow[, c("LONGITUDE", "LATITUDE")])
distMat <- dist_cpp(locs1, locs1, geographic = TRUE)
```

---

extract	<i>Extract subset of an intsp or intgrd object</i>
---------	--

---

**Description**

Extract subset of an intsp or intgrd object

**Usage**

```
## S4 method for signature 'intgrd'
x$name

## S4 replacement method for signature 'intgrd'
x$name <- value

## S4 method for signature 'intgrd,ANY,missing,missing'
x[i, j, ..., drop = TRUE]

## S4 method for signature 'intsp'
x$name

## S4 replacement method for signature 'intsp'
x$name <- value

## S4 method for signature 'intsp,ANY,missing,missing'
x[i, j, ..., drop = TRUE]

## S4 replacement method for signature 'intsp'
x$name <- value
```

**Arguments**

x	An object of class intsp or class intgrd from which to replace elements.
name	Character vector corresponding to the name of the column that will be extracted or replaced.
value	The new data used to replaced drop data in the desired slot.
i, j, ...	indices specifying elements to extract or replace. See generic function documentation for details.
drop	The requested column that may be reassigned.

**Value**

An object of class `intsp` or `intgrd`.

---

<code>fit.intvariogram</code>	<i>Function to create a variogram object for interval-valued data.</i>
-------------------------------	--

---

**Description**

Function to create a variogram object for interval-valued data.

**Usage**

```
fit.intvariogram(x, models = gstat::vgm(rep("Sph", 3)), ...)
```

**Arguments**

<code>x</code>	An object of class <code>intvariogram</code> .
<code>models</code>	an object of class <code>variogramModelList</code> . The user must specify at least two variogram models to fit (for center and radius). If less than three models are specified then the method does not fit a variogram for the center/radius interaction.
<code>...</code>	Additional arguments for <code>fit.variogram</code> .

**Value**

A list of variograms objects from the `gstat` package.

---

<code>head.intgrd</code>	<i>Print the head of an <code>intgrd</code> object.</i>
--------------------------	---

---

**Description**

This function extends `print.sp` by including a display and summary of the interval slot for the object.

**Usage**

```
## S3 method for class 'intgrd'
head(x, n = 6, ..., digits = getOption("digits"))
```

**Arguments**

<code>x</code>	An object of class <code>intgrd</code> .
<code>n</code>	Number of rows to print to the screen.
<code>...</code>	Additional arguments to print.
<code>digits</code>	Determines how numeric values are printed to the screen (default from <code>sp</code> package).

**Value**

Prints a subset of the object observations to the screen, identical to [SpatialPoints-class](#), as well as summary statistics for the interval slot.

---

head.intsp	<i>Print the head of an intsp object.</i>
------------	---

---

**Description**

This function extends printing methods in the sp package by including a display and summary of the head of the interval slot for the object.

**Usage**

```
## S3 method for class 'intsp'
head(x, n = 6, ..., digits = getOption("digits"))
```

**Arguments**

x	An object of class intsp.
n	The number of rows to print to the screen.
...	Additional arguments to <a href="#">print</a> .
digits	Determines how values are printed to the screen (default taken from sp package).

**Value**

Prints a subset of the object to the screen, identical to [SpatialPoints-class](#), as well as summary statistics for the interval slot.

---

interval	<i>Function to extract the interval of an intsp or intgrd object</i>
----------	--

---

**Description**

Function to extract the interval of an intsp or intgrd object

**Usage**

```
interval(x)

## S4 method for signature 'intgrd'
interval(x)

## S4 method for signature 'intsp'
interval(x)
```

**Arguments**

x                    An object of class intsp or intgrd.

**Value**

A matrix containing the interval data.

---

interval<-                    *Function to reassign the contents of the interval slot*

---

**Description**

Function to reassign the contents of the interval slot

**Usage**

```
interval(x) <- value

## S4 replacement method for signature 'intgrd'
interval(x) <- value

## S4 replacement method for signature 'SpatialPixelsDataFrame'
interval(x) <- value

## S4 replacement method for signature 'intsp'
interval(x) <- value

## S4 replacement method for signature 'SpatialPointsDataFrame'
interval(x) <- value
```

**Arguments**

x                    An object of class intsp or intgrd.  
value                Either a character vector of length two specifying the column names which will occupy the interval slot. Or, a matrix of two columns to fill the slot.

---

intgrd-class                    *An interval extension of a SpatialPixelsDataFrame*

---

**Description**

An interval extension of a SpatialPixelsDataFrame

**Slots**

interval A matrix of two columns representing the lower and upper endpoints of an interval.



intridge

*Algorithmic implementation of interval valued kriging.***Description**

Function to implement the interval valued extension of ordinary and simple kriging. Includes all necessary input checks and error handling. Essentially acts as a switch function between the R and c++ versions of the algorithm.

**Usage**

```
intridge(
  locations,
  newdata,
  models,
  centerFormula = center ~ 1,
  eta = 0.75,
  A = c(1, 1, 0),
  trend = NULL,
  thresh = 100,
  tolq = 0.001,
  maxq = 100,
  tolp = 0.001,
  maxp = 100,
  r = 1,
  useR = TRUE,
  fast = FALSE,
  weights = FALSE,
  cores = 1,
  progress = FALSE
)
```

**Arguments**

locations	An object of class <code>intsp</code> , specifying the prediction locations with an interval-valued response.
newdata	An object of class <code>SpatialPointsDataFrame</code> or <code>SpatialPixelsDataFrame</code> specifying the locations at which to predict intervals.
models	A list of variogram models of class <code>vgm</code> (see <a href="#">vgm</a> ) When specified, the third model represents the center/radius interaction.
centerFormula	The formula describing the relationship between the radius center and any number of explanatory variables. Must have "center" as the lone dependent variable. Note that there is no option to fit an external trend to the radius.
eta	A growth/shrink parameter for penalty term. For simple kriging: $\eta > 1$ . For ordinary kriging: $\eta < 1$ .

A	vector of length three representing the weights of the generalized L2 distance: the vector of three contains the weights for the center, radius, and center/radius respectively. $A = c(1, 1, 0)$ assumes the regular L2 distance calculation for intervals.
trend	If null, use ordinary kriging. When specified, represents the known mean of the stationary process and is an indication to use simple kriging.
thresh	Let $n = \text{length}(\text{locations})$ . When $\text{abs}(\text{lam}_i) < 1/(n*\text{thresh})$ , this lambda value is set to 0.
tolq	For a set penalty term, convergence is satisfied if $\text{max}(\text{abs}(\text{lamUp}-\text{lam})) < \text{tolq}$ .
maxq	For a set penalty term, the max number of iterations allowed for convergence.
tolp	When $\text{abs}(\text{sum}(\text{abs}(\text{lam})) - 1) < \text{tolp}$ , consider the constraints satisfied.
maxp	Maximum number of allowed iterations to satisfy equation constraints.
r	The starting value of the penalty term. Must be relatively large to ensure that the initial solution stays within the feasible region.
useR	If TRUE, use the R version of the algorithm. If FALSE, use the rcppArmadillo version.
fast	(Simple kriging only). If TRUE, allows lambdas to converge to 0 and subsets matrices accordingly. When FALSE, runs simple kriging using a barrier penalty at 0. Fast = TRUE is orders of magnitude faster than the full implementation. However, it is not recommended when input measurements are sparse as it is known to have convergence issues in these cases.
weights	If TRUE, return the vector kriging weights for each prediction. If false, simply return the predicted output.
cores	An integer (for parallel computing): specify the number of cores that will be devoted to the computation. Note that the argument 'all' will use all available cores minus one. Parallel processing is only relevant if you are predicting for more than one location. Note there is no parallel option when useR = FALSE.
progress	(Not valid when useR=FALSE or in parallel): Print a progress bar showing the progress of the predictions at the new locations.

## Details

The centerFormula argument allows for an external trend to be fit to the interval centers. No option is provided to scale the interval radii. This means that any transformations should be applied to the entire interval prior to input into interval-valued kriging. This ensures that input into the interval-valued kriging algorithm are well-defined intervals with properly ordered upper and lower endpoints.

## Value

A matrix with 4 columns where rows correspond to the prediction locations and columns correspond to:

- center prediction
- radius predictions
- kriging prediction variance
- warning column for non-convergent optimization problem (0 - no warning, 1 - warning)

**Examples**

```

# First, define the location and elevation of interest.
# (In this case we pick coordinates of Utah State University)
tproj <- sp::CRS(SRS_string = "EPSG:4326")
templocs <- data.frame(lat = 41.745, long = -111.810, ELEVATION = 1456)
sp::coordinates(templocs) <- c("long", "lat")
sp::proj4string(templocs) <- tproj

# Load the Utah Snow Load Data
data(utsnow)
utsnow.sp <- utsnow

# Convert to an 'intsp' object that inherits a SpatialPointsDataFrame
sp::coordinates(utsnow.sp) <- c("LONGITUDE", "LATITUDE")
sp::proj4string(utsnow.sp) <- tproj
interval(utsnow.sp) <- c("minDL", "maxDL")
# Define the formulas we will use to define the intervals.
temp_formula <- center ~ ELEVATION

# Define, fit and check the variogram fits.
varios <- intvariogram(utsnow.sp,
                      centerFormula = temp_formula)
varioFit <- fit.intvariogram(varios, models = gstat::vgm(c("Sph", "Sph", "Gau")))
preds <- intkrige::intkrige(locations = utsnow.sp,
                            newdata = templocs,
                            models = varioFit,
                            centerFormula = temp_formula)

```

---

intsp-class

*An interval extension of a SpatialPointsDataFrame*


---

**Description**

An interval extension of a SpatialPointsDataFrame

**Slots**

`interval` A matrix of two columns representing the lower and upper endpoints of an interval.

---

intvariogram

*Function to fit empirical variograms for an interval-valued spatial object*


---

**Description**

Function to fit empirical variograms for an interval-valued spatial object

**Usage**

```
intvariogram(x, centerFormula = center ~ 1, ...)

## S4 method for signature 'intgrd'
intvariogram(x, centerFormula = center ~ 1, ...)

## S4 method for signature 'intsp'
intvariogram(x, centerFormula = center ~ 1, ...)
```

**Arguments**

`x` An object of class `intsp` or `intgrd`.

`centerFormula` A formula specifying any external trends for the interval centers.

`...` Additional arguments for `sp::variogram()`.

**Value**

An object of class `'intvariogram'` containing empirical variograms for the center, radius, and center/radius interaction.

---

<code>intvCheck</code>	<i>Function to visualize the three variograms from an interval valued spatial data frame.</i>
------------------------	---

---

**Description**

Function to visualize the three variograms from an interval valued spatial data frame.

**Usage**

```
intvCheck(x, models, ...)
```

**Arguments**

`x` An object of class `intvariogram`.

`models` A list of fitted variogram models, typically an output of `fit.intvariogram`.

`...` Additional arguments to `plot.gstatVariogram`.

---

ohMap	<i>Ohio River Basin map</i>
-------	-----------------------------

---

**Description**

Ohio River Basin map

**Usage**

ohMap

**Format**

A SpatialPolygons objects with the boundaries of the Ohio river basin. Shapefile obtained from the U.S. Geological Survey National Watershed Boundary Dataset.

---

ohtemp	<i>30 year trimmed mean daily maximum and minimum temperatures for the Ohio river basin.</i>
--------	--

---

**Description**

Intervals are defined by the mean daily maximum and minimum temperatures for the Ohio river basin from January 1, 1988 - December 31, 2018. The 116 observations in this dataset all had at least 300 daily observations of temperature in at least 30 of the 31 considered years. The mean was calculated after trimming 10 influence of potential outliers.

**Usage**

ohtemp

**Format**

A data frame with 161 rows and 8 variables:

**ID** The global historical climatological network (GHCN) station identifier

**NAME** The GHCN station name

**STATE** The two-digit designation for the state in which each station resides

**LATITUDE** Latitude coordinate position

**LONGITUDE** Longitude coordinate position

**ELEVATION** Elevation of the measurement location (meters)

**minm** The 30 year mean daily minimum temperature (tenths of degrees Celsius)

**maxm** The 30 year mean daily maximum temperature (tenths of degrees Celsius)

---

```
plot,intgrd,missing-method
```

*Create an interval plot for spatial grid.*

---

### Description

Calls `spplot` to plot the locations, centers, and radii of an `intgrd` object in a single figure.

### Usage

```
## S4 method for signature 'intgrd,missing'
plot(x, beside = TRUE, circleCol = "black", minRad = 0.25, ...)
```

### Arguments

<code>x</code>	An object of class <code>intgrd</code> .
<code>beside</code>	If true, center and radius plotted side by side if false, center and radius are plotted in a single figure with the center plotted using color and the radius plotted using circles circumscribed within each grid cell.
<code>circleCol</code>	If <code>beside=TRUE</code> , the color of the circles that will be circumscribed within each grid cell.
<code>minRad</code>	The minimum value of the radius in the circles drawn to represent the interval radii. Must be a number between 0 and 1 where approaching 0 results in a point being drawn in the center of the grid, while approaching 1 results in every circle being circumscribed in their respective grid cell (which is not very interesting).
<code>...</code>	Additional arguments to <code>spplot</code> .

---

```
plot,intsp,missing-method
```

*Create an interval plot for spatial points.*

---

### Description

Calls `spplot` to plot the locations, centers, and radii of an interval-valued spatial data frame in a single figure.

### Usage

```
## S4 method for signature 'intsp,missing'
plot(
  x,
  locationsOnly = FALSE,
  legend.positions = c("left", "right"),
  cuts = c(5, 5),
```

```

    radSize = c(0.1, 3),
    pch = 16,
    alpha = 0.5,
    ...
)

```

### Arguments

x	An object of class <code>intsp</code> .
locationsOnly	If TRUE, simply plots geographic locations.
legend.positions	The positions of the center and radius legend relative to the plotting window.
cuts	The number of ranges of values to print in the center and radius legend respectively.
radSize	A vector of length 2 indicating the range of point sizes to plot to visualize radii magnitudes.
pch	The shape of the points (see <a href="#">plot</a> ).
alpha	The transparency of the points.
...	Additional arguments to <a href="#">spplot</a> .

---

print.intgrd	<i>Print the contents of an intgrd object</i>
--------------	---

---

### Description

This function extends `print.sp` by including a display and summary of the interval slot for the object.

### Usage

```

## S3 method for class 'intgrd'
print(x, ..., digits = getOption("digits"))

```

### Arguments

x	An object of class <code>intgrd</code> .
...	Additional arguments to <a href="#">print</a> .
digits	Determines how numeric values are printed to the screen (default from <code>sp</code> package).

### Value

Prints object to the screen, identical to [SpatialPoints-class](#), as well as summary statistics for the interval slot.

---

```
print.intsp          Print the contents of an intsp object
```

---

### Description

This function extends printing methods in the sp package by including a display and summary of the interval slot for the object.

### Usage

```
## S3 method for class 'intsp'
print(x, digits = getOption("digits"), ...)
```

### Arguments

x	An object of class intsp.
digits	Determines how numbers are displayed on the screen. Default option taken from sp package.
...	Additional arguments to <a href="#">print</a> .

### Value

Prints object to the screen, identical to [SpatialPoints-class](#), as well as summary statistics for the interval slot.

---

```
print.summary.intgrd  Print the object summary to the screen.
```

---

### Description

Print the object summary to the screen.

### Usage

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

### Arguments

x	An object of class intgrd.
...	Additional arguments to <a href="#">print</a> .

### Value

Prints a subset of the object observations to the screen, identical to [SpatialPoints-class](#), as well as summary statistics for the interval slot.



---

print.summary.intsp    *Print the object summary to the screen.*

---

**Description**

Print the object summary to the screen.

**Usage**

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

**Arguments**

x                    An object an object of class intsp.  
...                  Additional arguments to [print](#).

**Value**

Prints a series of summaries to the screen, identical to [SpatialPoints-class](#), as well as summary statistics for the interval slot.

---

show, intgrd-method    *Extension of the show function for intgrd objects*

---

**Description**

Extension of the show function for intgrd objects

**Usage**

```
## S4 method for signature 'intgrd'  
show(object)
```

**Arguments**

object              and object of class intgrd

---

show, intsp-method	<i>Extension of the show function for intgrd objects</i>
--------------------	--

---

**Description**

Extension of the show function for intgrd objects

**Usage**

```
## S4 method for signature 'intsp'
show(object)
```

**Arguments**

object	An object of class intsp.
--------	---------------------------

---

summary.intgrd	<i>Summarize the contents of an intgrd object, including special summaries for the interval slot.</i>
----------------	---

---

**Description**

Summarize the contents of an intgrd object, including special summaries for the interval slot.

**Usage**

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

**Arguments**

object	An object of class intgrd.
...	Additional arguments to <a href="#">summary</a> .

**Value**

Prints a summary of the object observations to the screen, identical to [SpatialPoints-class](#), as well as summary statistics for the interval slot.

---

summary.intsp	<i>Summarize the contents of an intsp object, including special summaries for the interval slot.</i>
---------------	--

---

**Description**

Summarize the contents of an intsp object, including special summaries for the interval slot.

**Usage**

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

**Arguments**

object	An object of class intsp.
...	Additional arguments to <a href="#">summary</a> .

**Value**

Prints a series of summaries to the screen, identical to [SpatialPoints-class](#), as well as summary statistics for the interval slot.

---

tail.intgrd	<i>Print the tail of an intgrd object.</i>
-------------	--

---

**Description**

This function extends print.sp by including a display and summary of the interval slot for the object.

**Usage**

```
## S3 method for class 'intgrd'
tail(x, n = 6, ..., digits = getOption("digits"))
```

**Arguments**

x	An object of class intgrd.
n	The number of rows to print to the screen.
...	Additional arguments to tail.
digits	Determines how numbers are displayed to the screen (default taken from package sp).

**Value**

Prints a subset of the object observations to the screen, identical to [SpatialPoints-class](#), as well as summary statistics for the interval slot.

---

tail.intsp	<i>Print the tail of an intsp object.</i>
------------	---

---

### Description

This function extends print.sp by including a display and summary of the interval slot for the object.

### Usage

```
## S3 method for class 'intsp'
tail(x, n = 6, ..., digits = getOption("digits"))
```

### Arguments

x	An object of class intsp.
n	The number of rows to print to the screen.
...	Additional arguments to <a href="#">tail</a> .
digits	Determines how numbers are displayed (default taken from sp package).

### Value

Prints a subset of the object to the screen, identical to [SpatialPoints-class](#), as well as summary statistics for the interval slot.

---

utsnow	<i>An interval-valued design ground snow load dataset for Utah.</i>
--------	---

---

### Description

A dataset containing the interval-valued data used in the analysis of Bean et. al (2019). The 415 measurement locations included in the dataset are taken from Bean et. al. (2018).

### Usage

```
utsnow
```

### Format

A data frame with 415 rows and 8 variables:

**STATION** The global historical climatological network (GHCN) station identifier

**STATION\_NAME** The GHCN station name

**LATITUDE** Latitude coordinate position

**LONGITUDE** Longitude coordinate position

- ELEVATION** Elevation of the measurement location (meters)
- minDL** The lower endpoint of the interval-valued design snow load as measured in kilopascals (kpa)
- maxDL** The upper endpoint of the design snow load interval (kpa)
- pointDL** The original point-valued design snow load from the 2018 Utah Snow Load Study (kpa)

### Details

The interval valued kriging analysis described in Bean et. al. (2019) analyzes this dataset on a log scale after removing the effect of elevation. An example of predictions using this workflow are found in the README associated with this package.

### References

- Bean B, Sun Y, Maguire M (2019). “Interval-Valued Kriging Models for Geostatistical Mapping with Uncertain Inputs.” (publication pending).
- Bean B, Maguire M, Sun Y (2018). “The Utah Snow Load Study.” [https://digitalcommons.usu.edu/cgi/viewcontent.cgi?article=4591&context=cee\\_facpub](https://digitalcommons.usu.edu/cgi/viewcontent.cgi?article=4591&context=cee_facpub).

---

utsnow_dtl	<i>An interval-valued design ground snow load dataset for Utah that only considers depth to load conversions.</i>
------------	---

---

### Description

A dataset containing the interval-valued data used in the analysis of Bean et. al. (2019). The 415 measurement locations included in the dataset are taken from Bean et. al. (2018).

### Usage

utsnow\_dtl

### Format

A data frame with 415 rows and 8 variables:

- STATION** The global historical climatological network (GHCN) station identifier
- STATION\_NAME** The GHCN station name
- LATITUDE** Latitude coordinate position
- LONGITUDE** Longitude coordinate position
- ELEVATION** Elevation of the measurement location (meters)
- minDL** The lower endpoint of the interval-valued design snow load as measured in kilopascals (kpa)
- maxDL** The upper endpoint of the design snow load interval (kpa)
- pointDL** The original point-valued design snow load from the 2018 Utah Snow Load Study (kpa)

## Details

The interval valued kriging analysis described in Bean et. al. (2019) analyzes this dataset on a log scale after removing the effect of elevation. An example of predictions using this workflow are found in the README associated with this package. Note that this dataset differs from utsnow in that intervals only consider differences in depth to load conversions.

## References

Bean B, Sun Y, Maguire M (2019). “Interval-Valued Kriging Models for Geostatistical Mapping with Uncertain Inputs.” (publication pending).

Bean B, Maguire M, Sun Y (2018). “The Utah Snow Load Study.” [https://digitalcommons.usu.edu/cgi/viewcontent.cgi?article=4591&context=cee\\_facpub](https://digitalcommons.usu.edu/cgi/viewcontent.cgi?article=4591&context=cee_facpub).

---

utsnow_dt12	<i>An interval-valued design ground snow load dataset for Utah that only considers depth to load conversions.</i>
-------------	---

---

## Description

A dataset containing the interval-valued data used in the analysis of Bean et. al (2019). The 415 measurement locations included in the dataset are taken from Bean et. al. (2018).

## Usage

```
utsnow_dt12
```

## Format

A data frame with 415 rows and 8 variables:

**STATION** The global historical climatological network (GHCN) station identifier

**STATION\_NAME** The GHCN station name

**LATITUDE** Latitude coordinate position

**LONGITUDE** Longitude coordinate position

**ELEVATION** Elevation of the measurement location (meters)

**minDL** The lower endpoint of the interval-valued design snow load as measured in kilopascals (kpa)

**maxDL** The upper endpoint of the design snow load interval (kpa)

**pointDL** The original point-valued design snow load from the 2018 Utah Snow Load Study (kpa)

**Details**

The interval valued kriging analysis described in Bean et. al. (2019) analyzes this dataset on a log scale after removing the effect of elevation. An example of predictions using this workflow are found in the README associated with this package. Note that this dataset differs from utsnow in that intervals only consider differences in depth to load conversions. This dataset differs from utsnow\_dtl in that intervals are only calculated at the final step of the analysis: after finding 50 year events using all 8 depth to load conversion techniques. utsnow\_dtl rather created annual intervals, only fitting distributions to two sets of maximums (the annual lower and upper bounds), rather fitting 8 sets of maximums on all the depth-to-load conversion types.

**References**

Bean B, Sun Y, Maguire M (2019). “Interval-Valued Kriging Models for Geostatistical Mapping with Uncertain Inputs.” (publication pending).

Bean B, Maguire M, Sun Y (2018). “The Utah Snow Load Study.” [https://digitalcommons.usu.edu/cgi/viewcontent.cgi?article=4591&context=cee\\_facpub](https://digitalcommons.usu.edu/cgi/viewcontent.cgi?article=4591&context=cee_facpub).

# Index

- \* **datasets**
  - ohMap, 13
  - ohtemp, 13
  - utsnow, 20
  - utsnow\_dtl, 21
  - utsnow\_dtl2, 22
- [, intgrd, ANY, missing, missing-method (extract), 5
- [, intgrd-method (extract), 5
- [, intsp, ANY, missing, missing-method (extract), 5
- [, intsp-method (extract), 5
- \$(extract), 5
- \$, intgrd-method (extract), 5
- \$, intsp-method (extract), 5
- \$<- (extract), 5
- \$<-, intgrd-method (extract), 5
- \$<-, intsp-method (extract), 5
- \_PACKAGE (intkrige-package), 2
  
- as.data.frame, 3
- as.data.frame, intgrd-method (as.data.frame), 3
- as.data.frame, intsp-method (as.data.frame), 3
  
- dist\_cpp, 3, 4
  
- extract, 5
  
- fit.intvariogram, 3, 6, 12
- fit.variogram, 6
  
- head.intgrd, 6
- head.intsp, 7
  
- interval, 7
- interval, intgrd-method (interval), 7
- interval, intsp-method (interval), 7
- interval<-, 8
- interval<-, intgrd-method (interval<-), 8
  
- interval<-, intsp-method (interval<-), 8
- interval<-, SpatialPixelsDataFrame-method (interval<-), 8
- interval<-, SpatialPointsDataFrame-method (interval<-), 8
- intgrd (intgrd-class), 8
- intgrd-class, 8
- intkrige, 3, 9
- intkrige-package, 2
- intsp (intsp-class), 11
- intsp-class, 11
- intvariogram, 3, 11
- intvariogram, intgrd-method (intvariogram), 11
- intvariogram, intsp-method (intvariogram), 11
- intvCheck, 3, 12
  
- ohMap, 3, 13
- ohtemp, 3, 13
  
- plot, 3, 15
- plot, intgrd, missing-method, 14
- plot, intsp, missing-method, 14
- plot.gstatVariogram, 12
- print, 3, 7, 15–17
- print.intgrd, 15
- print.intsp, 16
- print.summary.intgrd, 16
- print.summary.intsp, 17
  
- show, intgrd-method, 17
- show, intsp-method, 18
- SpatialPolygons, 3
- spplot, 14, 15
- summary, 3, 18, 19
- summary.intgrd, 18
- summary.intsp, 19
  
- tail, 20



tail.intgrd, [19](#)

tail.intsp, [20](#)

utsnow, [3](#), [20](#)

utsnow\_dtl, [21](#)

utsnow\_dtl2, [22](#)

vgm, [9](#)