Package ‘realTimeloads’

October 18, 2023

Title Analyte Flux and Load from Estimates of Concentration and Discharge

Version 1.0.0

Description Flux (mass per unit time) and Load (mass) are computed from timeseries estimates of analyte concentration and discharge. Concentration timeseries are computed from regression between surrogate and user-provided analyte. Uncertainty in calculations is estimated using bootstrap resampling. Code for the processing of acoustic backscatter from horizontally profiling acoustic Doppler current profilers is provided. All methods detailed in Livsey et al (2020) <doi:10.1007/s12237-020-00734-z>, Livsey et al (2023) <doi:10.1029/2022WR033982>, and references therein.

License GPL (>= 3)

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

Depends R (>= 2.10)

Imports data.table, graphics, imputeTS, mice, signal, stats, TideHarmonics, utils

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Daniel Livsey [aut, cre, cph] (<https://orcid.org/0000-0002-2028-6128>)

Maintainer Daniel Livsey <livsey.daniel@gmail.com>

Repository CRAN

Date/Publication 2023-10-18 14:00:02 UTC

R topics documented:

`acoustic_backscatter_processing`

Process acoustic backscatter from hADCP

Description

Processes acoustic backscatter from horizontally profiling ADCP (hADCP). Returns attenuation of sound due to water and suspended-sediment. Applies all corrections to acoustic backscatter detailed in the guideline.

Usage

```
acoustic_backscatter_processing(
  Site,
  ADCP,
  Height,
  Sonde,
  Echo_Intensity_Beam_1,
  Echo_Intensity_Beam_2,
  Instrument_Noise_Level = NULL,
  Include_Rayleigh = FALSE,
  Include_near_field_correction = TRUE
)
```

Arguments

`Site` Data frame with site, local vertical datum, and ADCP elevation information

- **Site_name** Site name (string)
- **Site_number** Unique site code (string)
- **ADCP_elevation_above_bed_m** Elevation of the ADCP above the bed (m)

- **ADCP_elevation_above_gauge_datum_m** Elevation of the ADCP above local gauge datum (m)
- **Distance_of_gauge_datum_below_thalweg_m** Distance from local gauge datum to lower point in cross-section (m)
- **Start_date_and_time** Installation date of ADCP (time, POSIXct)
- **End_date_and_time** Date if/when ADCP is moved vertically (time, POSIXct)
- **Comment** User comment (string)

`ADCP` Data frame with various readings from ADCP

- **Site_number** Unique site code (string)
- **time** Date and time (time, POSIXct)
- **Ensemble** Measurment ensemble number (integer)
- **Accoustic_Frequency_kHz** Acoustic frequency of ADCP (kHz)
- **Transducer_radius_m** Radius of ADCP transducer (m)
- **Beam_angle_degrees** Angle of beam relative to normal (degrees)
- **Beam_aspect_ratio** Ratio of beam radius to beam length (-)
- **Range_to_bed_of_acoustic_beams_m** Normal range to bed, optional (m)
- **Range_to_water_surface_of_acoustic_beams_m** Normal range to water surface, optional (m)
- **Number_of_Cells** Number of measurement cells along beam (integer)
- **Bin_Size_m** Cell width measured normal to ADCP (m)
- **Blanking_distance_m** Blanking distance measured normal to ADCP (m)
- **Instrument_serial_number** Serial number of ADCP instrument (string)
- **CPU_serial_number** Serial number of ADCP CPU (string)
- **Ambient_Noise_Level_Beam_1_Counts** Ambient noise level for beam 1, optional (counts)
- **Ambient_Noise_Level_Beam_2_Counts** Ambient noise level for beam 2, optional (counts)
- **Distance_to_Bin_1_mid_point_m** Reported distance normal to ADCP to midpoint of bin/cell (m)
- **Distance_to_surface_m** Reported depth of ADCP from vertical beam, optional (m)
- **Speed_of_sound_m_per_s** Speed of sound used by ADCP in the field (m/s)
- **Temperature_degC** Temperature recorded by ADCP (degrees C)
- **Pressure_dbar** Pressure recorded by ADCP (dBar)
- **Salinity_PSU** Salinity in PSU recorded or assumed in ADCP data file, optional (PSU)
- **Distance_to_surface_m** Distance to water surface reported by vertical beam of ADCP (m)
- **Power_supply_voltage** Power to ADCP (V)

`Height` Data frame with timeseries of river height

- **time** Date and time (time, POSIXct)
- **Height_m** Water surface elevation above gauge datum (m)
- **Site_number** Unique site code (string)

`Sonde` Data frame with timeseries of conductivity, temperature, and depth from sonde

- **time** Date and time (time, POSIXct)
- **Water_Temperature_degC** Temperature (degrees C)
- **Conductivity_uS_per_cm** Conductivity (microS/cm)
- **Pressure_dbar** Pressure (dbar)
- **Site_number** Unique site code (string)

`Echo_Intensity_Beam_1`
Data frame of acoustic backscatter measurements from beam 2

- **Site_number** Unique site code (string)
- **time** Date and time (time, POSIXct)
- **Echo_Intensity_Counts_cell_n** Acoustic backscatter in nth cell (counts)

`Echo_Intensity_Beam_2`
Data frame of acoustic backscatter measurements from beam 2

- **Site_number** Unique site code (string)
- **time** Date and time (time, POSIXct)
- **Echo_Intensity_Counts_cell_n** Acoustic backscatter in nth cell (counts)

`Instrument_Noise_Level`
Estimate of noise level, recommended if ambient noise level is not recorded (counts)

`Include_Rayleigh`
Logical to include data within Rayleigh Distance for processing of acoustic backsactter

`Include_near_field_correction`
Logical to include near-field correction of Downing et al (1995)

Value

List with processed data, all variable names and units are written-out in list items, see Livsey (in review) for details of each variable

Author(s)

Daniel Livsey (2023) ORCID: 0000-0002-2028-6128

References

Livsey, D.N. (in review). National Industry Guidelines for hydrometric monitoring–Part 12: Application of acoustic Doppler velocity meters to measure suspended-sediment load. Bureau of Meteorology. Melbourne, Australia.

Examples

```
InputData <- realTimeloads::ExampleData
Site <- InputData$Site
ADCP <- InputData$ADCP
Height <- InputData$Height
Sonde <- InputData$Sonde
```

```
EIa <- InputData$Echo_Intensity
# example code assumes backscatter is equal across beams
EIb <- InputData$Echo_Intensity
Output <- acoustic_backscatter_processing(Site,ADCP,Height,Sonde,EIa,EIb)
```

---

attenuation_of_sound_by_water

Compute attenuation of sound in water given frequency, temperature, and salinity

---

Description

Computes attenuation of sound in water per Ainslie and McColm (1998)

Usage

```
attenuation_of_sound_by_water(freq, temp, sal)
```

Arguments

`freq`	frequency of sound (Hz)
`temp`	Water temperature (degrees C)
`sal`	Salinity (PSU)

Value

attenuation of sound in water (dB/m), divide by 20*log10(exp(1)) to convert to Nepers/m

Author(s)

Daniel Livsey (2023) ORCID: 0000-0002-2028-6128

References

Ainslie, M. A., & McColm, J. G. (1998). A simplified formula for viscous and chemical absorption in sea water. The Journal of the Acoustical Society of America, 103(3), 1671-1672.

Author modified Matlab code from David Schoellhamer

Examples

```
InputData <- realTimeloads::ExampleData
freq <- InputData$ADCP$Accoustic_Frequency_kHz*1000
cond <-InputData$Sonde$Conductivity_uS_per_cm
temp <- InputData$Sonde$Water_Temperature_degC
dbar <- InputData$Sonde$Pressure_dbar
sal <- ctd2sal(cond,temp,dbar)
aw <- attenuation_of_sound_by_water(freq,temp,sal) # dB/m
awNp <- attenuation_of_sound_by_water(freq,temp,sal)/(20*log10(exp(1))) # Np/m
```

`bootstrap_regression` *Regression parameters estimated using bootstrap resampling*

Description

Computes uncertainty in regression parameters of y(x) after Rustomji and Wilkinson (2008)

Usage

```
bootstrap_regression(Calibration, fit_eq, fit_glm = FALSE)
```

Arguments

`Calibration`	data frame with surrogate(s) followed by analyte in last column
`fit_eq`	equation used to fit y(x), string (e.g, "y ~ x + x2", "y ~ x", "log10(y)~x')
`fit_glm`	logical to use Generalized Linear Models for models with factor (i.e., categorical) predictors

Value

list with bootstrap regression parameters and list output from stats::lm()

Warning

User should inspect regression residuals and relevant statistics to ensure model form is reasonable, suggested reading: regression diagnostics in Statistical Methods in Water Resources (https://doi.org/10.3133/tm4a3).

One can call plot(fit) to view various regression diagnostic plots

Note

Bias Correction Factor (BCF) is only relevant when analyte is transformed to log units, see https://doi.org/10.3133/tm4a3 to convert a model that used log(analyte) back to linear units use: analyte = 10^(f(surrogates)) x BCF

Author(s)

Daniel Livsey (2023) ORCID: 0000-0002-2028-6128

References

Rustomji, P., & Wilkinson, S. N. (2008). Applying bootstrap resampling to quantify uncertainty in fluvial suspended sediment loads estimated using rating curves. Water resources research, 44(9).https://doi.org/10.1029/2007

Helsel, D.R., Hirsch, R.M., Ryberg, K.R., Archfield, S.A., and Gilroy, E.J., 2020, #' Statistical methods in water resources: U.S. Geological Survey Techniques and Methods, book 4, chap. A3, 458 p. https://doi.org/10.3133/tm4a3

Examples

```
# linear model
x <- 1:10
y <- 0.5*x + 10
boot <- bootstrap_regression(data.frame(x,y),"y~x")
# polynomial model, call to I() needed for squaring x in equation string
x <- 1:10
y <- x + x^2
boot <- bootstrap_regression(data.frame(x,y),"y ~ x+I(x^2)")
# power law model
# BCF returned since y is transformed to log units
x <- 1:10
y <- x^0.3
boot <- bootstrap_regression(data.frame(x,y),"log10(y)~log10(x)")
# multivariate model
a <- 1:10
b <- a*2
c <- a^2*b^3
boot <- bootstrap_regression(data.frame(a,b,c),"log10(c)~log10(a)+log10(b)")
```

---

butterworth_tidal_filter

Return non-tidal signal in data after Rulh and Simpson (2005)

Description

Applies a Butterworth filter with a 30-hour stop period and a 40-hour pass period

Usage

```
butterworth_tidal_filter(time, x)
```

Arguments

`time`	time for x (time, POSIXct)
`x`	any quantity, for example discharge (double)

Value

non-tidal signal in x with data affected by filter ringing removed

Author(s)

Daniel Livsey (2023) ORCID: 0000-0002-2028-6128

References

Ruhl, C. A., & Simpson, M. R. (2005). Computation of discharge using the index-velocity method in tidally affected areas (Vol. 2005). Denver: US Department of the Interior, US Geological Survey. https://pubs.usgs.gov/sir/2005/5004/sir20055004.pdf

Examples

```
time <- realTimeloads::ExampleData$Height$time
x <- realTimeloads::ExampleData$Height$Height_m
xf <- butterworth_tidal_filter(time,x)
```

---

compute_load   *Compute load with uncertainty on concentration estimates*

Description

Compute load with uncertainty on concentration estimates from bootstrap regression after Rustomji and Wilkinson (2008)

Usage

```
compute_load(Surrogate, Discharge, Regression, period = NULL)
```

Arguments

`Surrogate`	data frame with time (PosixCt) and surrogate(s) (x,...)
`Discharge`	data frame with time (PosixCt) and discharge in cubic meters per second
`Regression`	data frame from bootstrap_regression() that determines analyte(surrogate)
`period`	two element vector time (PosixCt) indicating period over which load is computed

Value

list with data frames of estimated concentration and flux used to compute load (i.e., the sum of flux)

Note

Surrogate and Discharge time series can be on different time steps

If period is NULL, computes load over time in Surrogate

Warning

Discharge should be in cubic meters per second

Analyte concentration estimated from surrogate should be in milligrams per second

Author(s)

Daniel Livsey (2023) ORCID: 0000-0002-2028-6128

References

Rustomji, P., & Wilkinson, S. N. (2008). Applying bootstrap resampling to quantify uncertainty in fluvial suspended sediment loads estimated using rating curves. Water resources research, 44(9).https://doi.org/10.1029/2007

Helsel, D.R., Hirsch, R.M., Ryberg, K.R., Archfield, S.A., and Gilroy, E.J., 2020, #' Statistical methods in water resources: U.S. Geological Survey Techniques and Methods, book 4, chap. A3, 458 p. https://doi.org/10.3133/tm4a3

Examples

```
Turbidity_FNU <- realTimeloads::ExampleData$Sonde$Turbidity
TSS_mg_per_l <- realTimeloads::ExampleData$Sediment_Samples$SSCpt_mg_per_liter
Discharge <- realTimeloads::ExampleData$Discharge
Calibration <- data.frame(Turbidity_FNU,TSS_mg_per_l)
time <- realTimeloads::ExampleData$Sonde$time
Surrogate <- data.frame(time,Turbidity_FNU)
Regression = bootstrap_regression(Calibration,'TSS_mg_per_l~Turbidity_FNU')
period <- c(as.POSIXct("2000-02-16 AEST"),as.POSIXct("2000-03-16 AEST"))
Output <- compute_load(Surrogate,Discharge,Regression,period)
```

---

ctd2sal — *Compute salinity (PSU) from conductivity, water temperature, and depth*

Description

Computes salinity from conductivity, water temperature, and depth.

Usage

```
ctd2sal(cond, temp, dbar)
```

Arguments

cond	Conductance (microS/cm)
temp	Water temperature (degrees C)
dbar	Pressure (dBar) or water depth (m)

Value

Salinity in PSU

Warning

If specific conductivity is returned from the sonde, the temperature at which specific conductivity is computed should be utilized

Author(s)

Daniel Livsey (2023) ORCID: 0000-0002-2028-6128

References

Fofonoff, N. P., & Millard Jr, R. C. (1983). Algorithms for the computation of fundamental properties of seawater.

Chen, C. T. A., & Millero, F. J. (1986). Thermodynamic properties for natural waters covering only the limnological range 1. Limnology and Oceanography, 31(3), 657-662.

Hill, K., Dauphinee, T., & Woods, D. (1986). The extension of the Practical Salinity Scale 1978 to low salinities. IEEE Journal of Oceanic Engineering, 11(1), 109-112.

Author modified Matlab code from David Schoellhamer

Examples

```
Sonde <- realTimeloads::ExampleData$Sonde
sal <- ctd2sal(Sonde$Conductivity_uS_per_cm,Sonde$Water_Temperature_degC,Sonde$Pressure_dbar)
```

`estimate_timeseries` *Compute timeseries with uncertainty from bootstrap regression*

Description

Compute uncertainty on timeseries from bootstrap regression after Rustomji and Wilkinson (2008)

Usage

```
estimate_timeseries(Surrogate, Regression)
```

Arguments

`Surrogate`	data frame with time (PosixCt) and surrogate(s) (x,...)
`Regression`	data frame from bootstrap_regression() that determines analyte(surrogate)

Value

list with inputs and uncertainty on timeseries estimated from Regression

Author(s)

Daniel Livsey (2023) ORCID: 0000-0002-2028-6128

References

Rustomji, P., & Wilkinson, S. N. (2008). Applying bootstrap resampling to quantify uncertainty in fluvial suspended sediment loads estimated using rating curves. Water resources research, 44(9).https://doi.org/10.1029/2007

Helsel, D.R., Hirsch, R.M., Ryberg, K.R., Archfield, S.A., and Gilroy, E.J., 2020, #' Statistical methods in water resources: U.S. Geological Survey Techniques and Methods, book 4, chap. A3, 458 p. https://doi.org/10.3133/tm4a3

Examples

```
Turbidity_FNU <- realTimeloads::ExampleData$Sonde$Turbidity
TSS_mg_per_l <- realTimeloads::ExampleData$Sediment_Samples$SSCpt_mg_per_liter
Calibration <- data.frame(Turbidity_FNU,TSS_mg_per_l)
time <- realTimeloads::ExampleData$Sonde$time
Surrogate <- data.frame(time,Turbidity_FNU)
Regression = bootstrap_regression(Calibration,'TSS_mg_per_l~Turbidity_FNU')
Output <- estimate_timeseries(Surrogate,Regression)
```

ExampleCode *Computes sediment load per guideline from ExampleData*

Description

Computes sediment load per guideline from ExampleData

Usage

```
ExampleCode()
```

Value

list with data frames of estimated concentration and flux along with data used in regression and surrogate timeseries

Author(s)

Daniel Livsey (2023) ORCID: 0000-0002-2028-6128

References

Livsey, D.N. (in review). National Industry Guidelines for hydrometric monitoring–Part 12: Application of acoustic Doppler velocity meters to measure suspended-sediment load. Bureau of Meteorology. Melbourne, Australia.

See Also

realTimeloads Package help file

Examples

```
Output <- ExampleCode()
```

`ExampleCodeSCI` *Computes sediment load from optical and acoustic backscatter measurements*

Description

Computes sediment load per guideline from optical and acoustic backscatter measurements combined to the "Sediment Composition Index" (SCI) per Livsey et al (2023)

Usage

```
ExampleCodeSCI()
```

Value

total load with uncertainty computed from estimates of concentration from SCI

Author(s)

Daniel Livsey (2023) ORCID: 0000-0002-2028-6128

References

Livsey, D.N. (in review). National Industry Guidelines for hydrometric monitoring–Part 12: Application of acoustic Doppler velocity meters to measure suspended-sediment load. Bureau of Meteorology. Melbourne, Australia.

See Also

`realTimeloads` Package help file

Examples

```
Output <- ExampleCodeSCI()
```

`ExampleData` *Example data used to demonstrate computation of real-time sediment loads from horizontal acoustic Doppler current profiler (hADCP)*

Description

Synthetic dataset from modeled sediment transport and acoustic scattering detailed in the Appendices of Livsey (in review) Following dataframes are provided in list

Usage

```
ExampleData
```

Format

`Site` **, Site, site datum, and ADCP elevation information:**

Site_name Site name (string)

Site_number Unique site code (string)

ADCP_elevation_above_bed_m Elevation of the ADCP above the bed (m)

ADCP_elevation_above_gauge_datum_m Elevation of the ADCP above local gauge datum (m)

Distance_of_gauge_datum_below_thalweg_m Distance from local gauge datum to lower point in cross-section (m)

Start_date_and_time Installation date of ADCP (time, POSIXct)

End_date_and_time Date if/when ADCP is moved vertically (time, POSIXct)

Comment User comment (string)

`ADCP` **, ADCP readings except acoustic backscatter:**

Site_number Unique site code (string)

time Date and time (time, POSIXct)

Ensemble Measurment ensemble number (integer)

Accoustic_Frequency_kHz Acoustic frequency of ADCP (kHz)

Transducer_radius_m Radius of ADCP transducer (m)

Beam_angle_degrees Angle of beam relative to normal (degrees)

Beam_aspect_ratio Ratio of beam radius to beam length (-)

Number_of_Cells Number of measurement cells along beam (integer)

Bin_Size_m Cell width measured normal to ADCP (m)

Blanking_distance_m Blanking distance measured normal to ADCP (m)

Instrument_serial_number Serial number of ADCP instrument (string)

CPU_serial_number Serial number of ADCP CPU (string)

Ambient_Noise_Level_Beam_1_Counts Ambient noise level for beam 1 (counts)

Ambient_Noise_Level_Beam_2_Counts Ambient noise level for beam 2 (counts)

Distance_to_Bin_1_mid_point_m Reported distance normal to ADCP to midpoint of bin/cell (m)

Speed_of_sound_m_per_s Speed of sound used by ADCP in the field (m/s)

Temperature_degC Temperature recorded by ADCP (degrees C)

Pressure_dbar Pressure recorded by ADCP (dBar)

Distance_to_surface_m Distance to water surface reported by vertical beam of ADCP (m)

Power_supply_voltage Power to ADCP (V)

`Echo_Intensity` **, Acoustic backscatter measurements from ADCP:**

Site_number Unique site code (string)

time Date and time (time, POSIXct)

Echo_Intensity_Counts_cell_n Acoustic backscatter in nth cell (counts)

`Sonde` **, Conductivity, temperature, and depth from sonde:**

time Date and time (time, POSIXct)

Water_Temperature_degC Temperature (degrees C)

Conductivity_uS_per_cm Conductivity (microS/cm)

Pressure_dbar Pressure (dbar)

Turbidity_FNU Turbidity (FNU)

Site_number Unique site code (string)

`Height` **, River height in meters referenced to gauge datum:**

time Date and time (time, POSIXct)

Height_m Water surface elevation above gauge datum (m)

Site_number Unique site code (string)

`Discharge`**, Discharge timeseries in cubic meters per second:**

time Date and time (time, POSIXct)

Discharge_m_cubed_per_s Dischage (cubic meters per second)

Site_number Unique site code (string)

`Sediment_Samples` **, Measured sediment concentration in milligrams per liter (SSC or TSS):**

time Date and time (time, POSIXct)

SSCxs_mg_per_liter Concentration of suspended-sediment in milligrams per liter, depth-averaged and velocity weighted average for cross-section

SSCpt_mg_per_liter Concentration of suspended-sediment in milligrams per liter, measured at-a-point at elevation of hADCP

Site_number Unique site code (string)

Examples

```
data(ExampleData) # lazy-load ony, unable to inspect contents in Rstudio
names(ExampleData) # load data for inspection in Rstudio and view names of items in list
```

Author(s)

Daniel Livsey (2023) ORCID: 0000-0002-2028-6128

Source

Livsey, D.N. (in review). National Industry Guidelines for hydrometric monitoring–Part 12: Application of acoustic Doppler velocity meters to measure suspended-sediment load. Bureau of Meteorology. Melbourne, Australia.

---

hADCPLoads *Compute sediment load per guideline using acoustic backscatter from processed hADCP data*

Description

Computes sediment load per guideline from user data in list "InputData" generated by function import_data()

Usage

```
hADCPLoads(InputData)
```

Arguments

InputData	List generated by import_data.R

Value

list with data frames of estimated concentration and flux along with data used in regression and surrogate timeseries

Author(s)

Daniel Livsey (2023) ORCID: 0000-0002-2028-6128

References

Livsey, D.N. (in review). National Industry Guidelines for hydrometric monitoring–Part 12: Application of acoustic Doppler velocity meters to measure suspended-sediment load. Bureau of Meteorology. Melbourne, Australia.

See Also

`import_data` Import data from files in user-specified folder

Examples

```
# loads example data in package folder extdata
InputData <- import_data()
# import_data(path) can be used to import user data
Output <- hADCPLoads(InputData)
```

`import_data` *Load data from comma-delimited .txt files to list to be used in function hADCPLoads()*

Description

Imports csv files to R, file names, variable names (and units) in csv text files must match variable names used in ExampleData.rda

Usage

```
import_data(data_folder)
```

Arguments

`data_folder`	file path to folder containing .txt csv files with format that matches files in extdata package folder

Value

list with data frames used in package code, see ?ExampleData for list format

Warning

Synthetic data used in ExampleData only has backscatter for one beam ("ADCP_Echo_Intensity.txt"), for user data, one should have backscatter for two beams with following names: "ADCP_Echo_Intensity_Beam_1.txt" and "ADCP_Echo_Intensity_Beam_2.txt"

Package arguments require variable names and units to match the names and variable units provided (see ?ExampleData, or .txt files in extdata folder)

Suggest saving all csv files in .txt format to ensure time format is not changed when editing/saving csv in Excel

Author(s)

Daniel Livsey (2023) ORCID: 0000-0002-2028-6128

References

Livsey, D.N. (in review). National Industry Guidelines for hydrometric monitoring–Part 12: Application of acoustic Doppler velocity meters to measure suspended-sediment load. Bureau of Meteorology. Melbourne, Australia.

See Also

hADCPLoads Process acoustic backscatter from hADCP and compute load using InputData from import_Data()

Examples

```
InputData <- import_data() # loads text files provided in package folder "extdata"
```

---

impute_data — *Returns x with gaps imputed using ARIMA and Decision Trees, optional uncertainty estimation using Monte Carlo resampling*

Description

Returns x with gaps imputed using ARIMA and Decision Trees with option to use harmonic model as predictors for x in decision tree algorithm. Uncertainty on imputed data is estimated using using Monte Carlo (MC) resampling adapting methods of Rustomji and Wilkinson (2008)

Usage

```
impute_data(
  time,
  x,
  Xreg = NULL,
  ti = NULL,
  hfit = NULL,
  harmonic = FALSE,
  only_use_Xreg = FALSE,
  MC = 1,
  ptrain = 1
)
```

Arguments

time	time for x (time, POSIXct)
x	any quantity (double)
Xreg	additional predictors for decision tree, required if harmonic is FALSE (rows = time, or if given, ti)
ti	time vector for interpolation (time, POSIXct)
hfit	model object from TideHarmonics::ftide
harmonic	logical if x exhibits tidal or diurnal variability
only_use_Xreg	logical for using Xreg only in decision tree
MC	number of Monte Carlo simulations for uncertainty estimation
ptrain	proportion of data used for training and testing model

Value

list with x imputed at time or ti, if given. Uncertainty estimated from Monte Carlo simulations

Note

If MC == 1, uncertainty is not evaluated. If ptrain == 1, uncertainty and validation accuracy are not computed

Author(s)

Daniel Livsey (2023) ORCID: 0000-0002-2028-6128

References

Rustomji, P., & Wilkinson, S. N. (2008). Applying bootstrap resampling to quantify uncertainty in fluvial suspended sediment loads estimated using rating curves. Water resources research, 44(9).

van Buuren S, Groothuis-Oudshoorn K (2011). “mice: Multivariate Imputation by Chained Equations in R.” Journal of Statistical Software, 45(3), 1-67. doi:10.18637/jss.v045.i03.

Stephenson AG (2016). Harmonic Analysis of Tides Using TideHarmonics. https://CRAN.R-project.org/package=TideHarmonics.

Moritz S, Bartz-Beielstein T (2017). “imputeTS: Time Series Missing Value Imputation in R.” The R Journal, 9(1), 207–218. doi:10.32614/RJ-2017-009.

Examples

```
# Impute non-tidal data
time <- realTimeloads::ExampleData$Sediment_Samples$time
xo <- realTimeloads::ExampleData$Sediment_Samples$SSCxs_mg_per_liter
Q <- realTimeloads::ExampleData$Discharge$Discharge_m_cubed_per_s
idata <- sample(1:length(xo),round(length(xo)*0.5),replace=FALSE)
x <- rep(NA,length(xo))
x[idata] <- xo[idata] # simulated samples
flow_concentrtion_ratio <- imputeTS::na_interpolation(Q/x)
Xreg <- cbind(Q,flow_concentrtion_ratio)
Output <- impute_data(time,x,Xreg,MC = 10,ptrain = 0.8)

# Impute tidal data
time <-TideHarmonics::Portland$DateTime[1:(24*90)]
xo <-TideHarmonics::Portland$SeaLevel[1:(24*90)]
idata <- sample(1:length(xo),round(length(xo)*0.5),replace=FALSE)
x <- rep(NA,length(xo))
x[idata] <- xo[idata] # simulated samples
Output <- impute_data(time,x,harmonic = TRUE,MC = 10,ptrain = 0.8)
```

`linear_interpolation_with_time_limit`
Linearly interpolate timeseries time(x) onto new timesetep ti

Description

Linear interpolation limited by time since previous or following reading

Usage

```
linear_interpolation_with_time_limit(time, x, ti, threshold)
```

Arguments

`time`	time for x (time, POSIXct)
`x`	any quantity, for example discharge (double)
`ti`	time where time(x) will be interpolated to (time, POSIXct)
`threshold`	maximum duration where interpolation is allowed (hours)

Value

a data frame with time (ti), x interpolated from time(x) onto ti, and logical (ibad) if interpolation exceeded threshold

Author(s)

Daniel Livsey (2023) ORCID: 0000-0002-2028-6128

References

Dowle M, and others (2023). data.table: Extension of 'data.frame'. https://cran.r-project.org/web/packages/data.table

Examples

```
InputData <- realTimeloads::ExampleData
ADCP <- InputData$ADCP
Height <- InputData$Height
# Interpolate river height to ADCP time
time <- realTimeloads::ExampleData$Height$time
x <- realTimeloads::ExampleData$Height$Height_m
ti <-realTimeloads::ExampleData$ADCP$time
threshold <- 1
Output<- linear_interpolation_with_time_limit(time,x,ti,threshold)
```

near_field_correction *Near-field correction of Downing et al (1995)*

Description

Computes dimensionless near-field correction

Usage

```
near_field_correction(freq, c, r, at)
```

Arguments

freq	Frequency of sound (Hz)
c	Speed of sound in water (m/s)
r	range to cell center measured along-beam (m)
at	Radius of ADCP transducer (m)

Value

Near-field correction (dimensionless)

Warning

See various references cautioning use of near-field correction (e.g., https://doi.org/10.1002/2016WR019695)

Author(s)

Daniel Livsey (2023) ORCID: 0000-0002-2028-6128

References

Downing, A., Thorne, P. D., & Vincent, C. E. (1995). Backscattering from a suspension in the near field of a piston transducer. The Journal of the Acoustical Society of America, 97(3), 1614-1620.

Examples

```
InputData <- realTimeloads::ExampleData
Sonde<- InputData$Sonde
freq <- InputData$ADCP$Accoustic_Frequency_kHz[1]*1000
S <- ctd2sal(Sonde$Conductivity_uS_per_cm,Sonde$Water_Temperature_degC,Sonde$Pressure_dbar)
c <- speed_of_sound(S,Sonde$Water_Temperature_degC,Sonde$Pressure_dbar)
at <- InputData$ADCP$Transducer_radius_m
r <- seq(0.1,10,0.1)
psi <- near_field_correction(freq,c[1],r,at[1])
```

`speed_of_sound` *Compute speed of sound in water given salinity, temperature, and depth*

Description

Computes speed of sound in water per Del grosso (1974)

Usage

```
speed_of_sound(sal, temp, depth)
```

Arguments

`sal`	Salinity (PSU)
`temp`	Water temperature (degrees C)
`depth`	Water depth (m) or pressure (dBar)

Value

Speed of sound in water (m/s)

Author(s)

Daniel Livsey (2023) ORCID: 0000-0002-2028-6128

References

Del Grosso, V. A. (1974). New equation for the speed of sound in natural waters (with comparisons to other equations). The Journal of the Acoustical Society of America, 56(4), 1084-1091. Author modified matlab code from David Schoellhamer

Examples

```
InputData <- realTimeloads::ExampleData
Sonde<- InputData$Sonde
sal <- ctd2sal(Sonde$Conductivity_uS_per_cm,Sonde$Water_Temperature_degC,Sonde$Pressure_dbar)
c <- speed_of_sound(sal,Sonde$Water_Temperature_degC,Sonde$Pressure_dbar)
```

surrogate_to_analyte_interpolation

Interpolate timeseries x(tx) onto y(ty)

Description

Interpolate timeseries x(tx) onto y(ty) with temporal threshold on interpolation

Usage

```
surrogate_to_analyte_interpolation(tx, x, ty, y, threshold)
```

Arguments

tx	time for x "surrogate" (time, POSIXct)
x	quantity used to estimate y, for example, accoustic backscatter
ty	time for y "analyte" (time, POSIXct)
y	measured quantity, for example, an analyte such as suspended-sediment concentration
threshold	maximum duration where interpolation is allowed (minutes)

Value

a data frame with surrogate (x) interpolated onto timestep of analyte (y), interpolated values exceeding threshold are excluded from the output

Author(s)

Daniel Livsey (2023) ORCID: 0000-0002-2028-6128

Examples

```
tx <- as.POSIXct(seq(0,24*60^2,60*1), origin = "2000-01-01",tz = "Australia/Brisbane")
x <- sin(1:length(tx))
ty <- as.POSIXct(seq(0,24*60^2,60*15), origin = "2000-01-01",tz = "Australia/Brisbane")
y <- seq(0,24*60^2,60*15)
threshold <- 10
calibration <- surrogate_to_analyte_interpolation(tx,x,ty,y,threshold)
```

Index