

# Package ‘ie2misc’

May 8, 2026

**Title** Irucka Embry's Miscellaneous USGS Functions

**Version** 0.9.2

**Maintainer** Irucka Embry <iembry@ecoccs.com>

**Depends** R (>= 3.3.0)

**Imports** readxl (>= 1.3.1), openxlsx (>= 4.1.4), gWidgets2, stringi, mgsub, reader, lubridate, tcltk, data.table (>= 1.10.2), utils, tools, assertthat, checkmate

**Suggests** rando, spelling

**Description** A collection of Irucka Embry's miscellaneous USGS functions (processing .exp and .psf files, statistical error functions, ``+'' dyadic operator for use with NA, creating ADAPS and QW spreadsheet files, calculating saturated enthalpy). Irucka created these functions while a Cherokee Nation Technology Solutions (CNTS) United States Geological Survey (USGS) Contractor and/or USGS employee.

**URL** <https://gitlab.com/iembry/ie2misc>

**BugReports** <https://gitlab.com/iembry/ie2misc/-/issues>

**License** CC0

**Copyright** This software is in the public domain because it contains materials that originally came from the United States Geological Survey, an agency of the United States Department of Interior. For more information, see the official USGS copyright policy at <<https://www.usgs.gov/information-policies-and-instructions/copyrights-and-credits#copyright>>

**Language** en-US

**Encoding** UTF-8

**RoxygenNote** 7.3.2

**NeedsCompilation** no

**Author** Irucka Embry [aut, cre],  
Anne Hoos [ctb],  
Timothy H. Diehl [ctb]

**Repository** CRAN

**Date/Publication** 2025-09-02 20:10:02 UTC

## Contents

adaps	2
dr	5
expFileOutput	7
madstat	10
mae	12
mape	14
psfFileChange	16
qw	19
rmse	22
sat_enthalpy	24
vnse	27
%+na%	29
<b>Index</b>	<b>32</b>

---

adaps	<i>adaps, adaps2, and adapsBATCH</i>
-------	--------------------------------------

---

## Description

adaps, adaps2, and adapsBATCH process raw Automated Data Processing System (ADAPS) .rdb files from the U.S. Geological Survey (USGS) National Water Information System (NWIS). For these functions, it is only for continuous ADAPS data of the following parameters: discharge (00060), FNU turbidity (63680), and NTRU turbidity (63676 from 63680).

## Usage

```
adaps(
  file = tk_choose.files(default = "", caption =
    "Select file(s) to open & hold down Ctrl to choose more than 1 file", multi = TRUE,
    filters = matrix(c("ADAPS file", ".rdb", "ADAPS file", ".RDB"), 4, 2, byrow = TRUE)),
  interactive = TRUE,
  overwrite = TRUE
)

adapsBATCH(
  path = tk_choose.dir(caption = "Select directory with the ADAPS .rdb files"),
  overwrite = TRUE
)

adaps2(file, overwrite = TRUE)
```

**Arguments**

file	Input ADAPS .rdb file(s) to be selected through a file dialog.
interactive	If interactive is TRUE, then the user will select the filenames(s) to use for saving with the file dialog. In order to select more than one file, the user must hold down the Ctrl (Control) button while mouse clicking the chosen files. If interactive is FALSE, then the user will select the directory, via the directory dialog, to use for saving and the original filenames will be used.
overwrite	If TRUE, overwrite any existing spreadsheet.
path	Directory path of ADAPS .rdb files to be selected through a directory dialog. The user will be asked where to find the ADAPS .rdb files & then the user will be asked where to save the ADAPS .xlsx files.

**Details**

adaps function opens single or multiple raw ADAPS .rdb file(s) to modify the format and then exports the file(s) in .xlsx format. This is done for a single file or multiple files that the user selects with a file dialog.

adaps2 function opens a single raw ADAPS .rdb file to modify the format and then exports the file in .xlsx format. This is done for a single file that the user selects without a file dialog.

adapsBATCH function opens raw ADAPS .rdb files, from a directory, to modify the format and then exports the files in .xlsx format. This is done in a BATCH mode (whole directory of ADAPS .rdb files) using a directory dialog.

adaps, adaps2, and adapsBATCH functions perform the same processes on the raw ADAPS .rdb files: 1) Read in the file and remove the 1st 4 or 5 lines depending on whether NTRU data are present or not, 2) create 4 or 5 columns (depending on whether NTRU data are present or not) based on the 1st 4 or 5 lines, and 3) export the modified file in .xlsx format.

The following lines are representative of the .rdb format used in the files that these functions can operate on. Note: ntru may not be present. If so, then there will only be 3 cases of 16N in the last row. The last row will be removed in the final spreadsheet.

```

DATETIME  ght
          cfs
          fnu
          ntru
          19D    16N  16N  16N  16N

```

**Value**

ADAPS .xlsx file(s)

**Source**

1. r - How can I check if a file is empty? - Stack Overflow answered by Konrad Rudolph and edited by Geekuna Matata on Apr 23 2014. See <https://stackoverflow.com/questions/23254002/how-can-i-check-if-a-file-is-empty>.

2. r - Better error message for stopifnot? - Stack Overflow answered by Andrie on Dec 1 2011. See <https://stackoverflow.com/questions/8343509/better-error-message-for-stopifnot>.
3. RDocumentation: TclInterface tcltk. See <https://www.rdocumentation.org/packages/tcltk/versions/3.3.1>.
4. James Wettenhall & Philippe Grosjean, File Open/Save dialogs in R tcltk, December 01, 2015. See <https://web.archive.org/web/20160521051207/http://www.sciviews.org/recipes/tcltk/TclTk-file-open-save-dialogs/>. Retrieved thanks to the Internet Archive: Way-back Machine
5. r - read csv files and perform function, then bind together - Stack Overflow answered by bjoseph on Jan 8 2015. See <https://stackoverflow.com/questions/27846715/read-csv-files-and-perform-f>
6. multiple output filenames in R - Stack Overflow asked and edited by Gabelins on Feb 1 2013. See <https://stackoverflow.com/questions/14651594/multiple-output-filenames-in-r>.
7. r - Regex return file name, remove path and file extension - Stack Overflow answered and edited by Ananda Mahto on Feb 25 2013. See <https://stackoverflow.com/questions/15073753/regex-return-file-name-remove-path-and-file-extension/15073919>.
8. R help - How to change the default Date format for write.csv function? answered by William Dunlap on Dec 28, 2009. See <https://hypatia.math.ethz.ch/pipermail/r-help/2009-December/416010.html>.
9. RDocumentation: strptime base. See <https://www.rdocumentation.org/packages/base/versions/3.3.1/topics/strptime>.
10. convert date and time string to POSIX in R - Stack Overflow commented by cryo111 on Sep 18 2013. See <https://stackoverflow.com/questions/18874400/convert-date-and-time-string-to-posix-in-18874863>.

## Examples

```
## Not run:

library(ie2misc)
# Example to check the input file format
# Copy and paste the following code into the R console if you
# wish to see the ADAPS .rdb input file format.
# Note the number of lines and the row headings.
file.show(system.file("extdata", "spring_creek_partial.rdb",
  package = "ie2misc"), title = paste("spring_creek_partial.rdb"))
# opens the .rdb file using the default text editor

# Examples to change (an) ADAPS .rdb file(s) interactively and
# non-interactively
adaps2(system.file("extdata", "spring_creek_partial.rdb",
  package = "ie2misc"))

adaps() # default where interactive = TRUE
# Follow the file dialog instructions
```

```

adaps(interactive = FALSE)
# Follow the file dialog instructions

# Example to change a directory of ADAPS .rdb files
adapsBATCH()
# Follow the file dialog instructions

## End(Not run)

```

---

dr *Index of agreement (dr)*

---

### Description

This function computes the "index of agreement (dr)".

### Usage

```
dr(predicted, observed, na.rm = FALSE)
```

### Arguments

predicted	numeric vector that contains the predicted data points (1st parameter)
observed	numeric vector that contains the observed data points (2nd parameter)
na.rm	logical vector that determines whether the missing values should be removed or not.

### Details

dr is expressed as

$$d_r = 1 - \frac{\sum_{i=1}^n |P_i - O_i|}{c \sum_{i=1}^n |O_i - \bar{O}|}, \text{ when } \sum_{i=1}^n |P_i - O_i| \leq c \sum_{i=1}^n |O_i - \bar{O}|$$

$$d_r = \frac{c \sum_{i=1}^n |O_i - \bar{O}|}{\sum_{i=1}^n |P_i - O_i|} - 1, \text{ when } \sum_{i=1}^n |P_i - O_i| > c \sum_{i=1}^n |O_i - \bar{O}|$$

$d_r$  the "index of agreement (dr)"

$n$  the number of observations

$P$  the "model estimates or predictions"

$O$  the "pairwise-matched observations that are judged to be reliable"

$\bar{O}$  the "true" mean of the observations

Note: Both  $P$  and  $O$  should have the same units.

The "index of agreement (dr)" is fully discussed in the Willmott reference.

### Value

"index of agreement (dr)" as a numeric vector. The default choice is that any NA values will be kept (na.rm = FALSE). This can be changed by specifying na.rm = TRUE, such as dr(pre, obs, na.rm = TRUE).

### References

Cort J. Willmott, Scott M. Robeson, and Kenji Matsuura, "A refined index of model performance", *International Journal of Climatology*, Volume 32, Issue 13, pages 2088-2094, 15 November 2012, article from ResearchGate: [https://www.researchgate.net/publication/235961403\\_A\\_refined\\_index\\_of\\_model\\_performance](https://www.researchgate.net/publication/235961403_A_refined_index_of_model_performance).

### See Also

[mape](#) for mean absolute percent error (MAPE), [mae](#) for mean-absolute error (MAE), [madstat](#) for mean-absolute deviation (MAD), [vnse](#) for Nash-Sutcliffe model efficiency (NSE), and [rmse](#) for root mean square error (RMSE).

### Examples

```
library(ie2misc)

obs <- 1:10 # observed
pre <- 2:11 # predicted
dr(pre, obs)

library(rando)

set_n(100) # makes the example reproducible
obs1 <- r_norm(.seed = 400) # observed
pre1 <- r_norm(.seed = 500) # predicted

# using the vectors pre1 and obs1
dr(pre1, obs1)

# using a matrix of the numeric vectors pre1 and obs1
mat1 <- matrix(data = c(obs1, pre1), nrow = length(pre1), ncol = 2,
  byrow = FALSE, dimnames = list(c(rep("", length(pre1))),
```

```

c("Predicted", "Observed"))
dr(mat1[, 2], mat1[, 1])

# mat1[, 1] # observed values from column 1 of mat1
# mat1[, 2] # predicted values from column 2 of mat1

# using a data.frame of the numeric vectors pre1 and obs1
df1 <- data.frame(obs1, pre1)
dr(df1[, 2], df1[, 1])

# df1[, 1] # observed values from column 1 of df1
# df1[, 2] # predicted values from column 2 of df1

library(data.table)

# using a data.table of the numeric vectors pre1 and obs1
df2 <- data.table(obs1, pre1)
dr(df2[, 2, with = FALSE][[1]], df2[, 1, with = FALSE][[1]])

# df2[, 1, with = FALSE][[1]] # observed values from column 1 of df2
# df2[, 2, with = FALSE][[1]] # predicted values from column 2 of df2

```

---

expFileOutput

*expFileOutput and expFileOutputBATCH*


---

## Description

expFileOutput and expFileOutputBATCH process raw .exp files generated by the USGS PeakFQ program (<https://www.usgs.gov/tools/peakfq>). The .exp output file from running PeakFQ is a shortened version of the longer narrative .prt file, more suitable for tabulating the model results than the .prt file; however, the information must be changed to a fixed field format.

## Usage

```

expFileOutput(
  file = tk_choose.files(default = "", caption =
    "Select file(s) to open & hold down Ctrl to choose more than 1 file", multi = TRUE,
    filters = matrix(c("Text file", ".exp", "Text file", ".EXP"), 4, 2, byrow = TRUE)),
  output = c("csv", "xlsx", "both"),
  overwrite = TRUE
)

expFileOutputBATCH(
  path = tk_choose.dir(caption = "Select the directory with the .exp files"),
  output = c("csv", "xlsx", "both"),

```

```

    overwrite = TRUE
  )

```

### Arguments

file	Input .exp file(s), using a file dialog, to obtain 1) the goodness-of-fit and trend results & 2) the exceedance probability values.
output	The exported format for each set of results (options are .csv, .xlsx, and both file types).
overwrite	logical vector that determines whether the existing should be overwritten or not.
path	Directory path of .exp files, to be selected through a directory dialog, to obtain 1) the goodness-of-fit and trend results & 2) the exceedance probability values. The user will be asked where to find the .exp files & then the user will be asked where to save the results files.

### Details

expFileOutput converts the user-selected .exp file to 1) a tab-delimited .csv file for a single station with the goodness-of-fit and trend results (all station information exists in a single row) and 2) a tab-delimited .csv file for a single station with the exceedance probability values on each row. For the .xlsx file, tab 1 contains the goodness-of-fit and trend results and tab 2 contains the exceedance probability values.

expFileOutputBATCH converts the user-selected directory of .exp files into 1) a tab-delimited .csv file as a single table, with one record for each station, with the goodness-of-fit and trend results and 2) a tab-delimited .csv file as a single table, with a set of records for each station, with the exceedance probability values on each row. For the .xlsx file, tab 1 contains the goodness-of-fit and trend results and tab 2 contains the exceedance probability values.

In the Examples section, there is R code for the user to view the example .exp file.

### Value

.csv, .xlsx, or both file types with 1) the goodness-of-fit and trend results & 2) the exceedance probability values for individual sites in separate files. Currently, in the BATCH mode, all stations in the given directory are combined in a single data.frame with both 1) and 2).

### Author(s)

Irucka Embry, Anne Hoos

### Source

1. r - How can I check if a file is empty? - Stack Overflow answered by Konrad Rudolph and edited by Geekuna Matata on Apr 23 2014. See <https://stackoverflow.com/questions/23254002/how-can-i-check-if-a-file-is-empty>.
2. r - Better error message for stopifnot? - Stack Overflow answered by Andrie on Dec 1 2011. See <https://stackoverflow.com/questions/8343509/better-error-message-for-stopifnot>.
3. RDocumentation: TclInterface tcltk. See <https://www.rdocumentation.org/packages/tcltk/versions/3.3.1>.

4. James Wettenhall & Philippe Grosjean, File Open/Save dialogs in R tcltk, December 01, 2015. See <https://web.archive.org/web/20160521051207/http://www.sciviews.org/recipes/tcltk/TclTk-file-open-save-dialogs/>. Retrieved thanks to the Internet Archive: Wayback Machine
5. r - read csv files and perform function, then bind together - Stack Overflow answered by bjoeseh on Jan 8 2015. See <https://stackoverflow.com/questions/27846715/read-csv-files-and-perform-f>
6. r - Convert column classes in data.table - Stack Overflow answered by Matt Dowle on Dec 27 2013. See <https://stackoverflow.com/questions/7813578/convert-column-classes-in-data-table>.
7. r - Transpose rows to columns and remove only NAs in resulting columns - Stack Overflow answered by David Arenburg on April 22, 2014. See <https://stackoverflow.com/questions/23225977/transpose-rows-to-columns-and-remove-only-nas-in-resulting-columns>.
8. r - Add a Column to a Dataframe From a List of Values - Stack Overflow answered by Matthew Plourde on Jun 21 2012. See <https://stackoverflow.com/questions/11130037/add-a-column-to-a-dataframe> 11130178.
9. r - Why does is.vector() return TRUE for list? - Stack Overflow answered by Andrie on May 17 2011. See [https://stackoverflow.com/questions/6032772/why-does-is-vector-return-true-for-list/6032909](https://stackoverflow.com/questions/6032772/why-does-is-vector-return-true-for-list/).
10. warnings - How do I get rid of the NOTE's generated by R CMD check when using for example dply in my package? - Stack Overflow answered by shadow on Mar 4 2015 and edited by shadow on Mar 5 2015. See <https://stackoverflow.com/questions/28851812/how-do-i-get-rid-of-the-not>
11. multiple output filenames in R - Stack Overflow asked and edited by Gabelins on Feb 1 2013. See <https://stackoverflow.com/questions/14651594/multiple-output-filenames-in-r>.
12. r - Regex return file name, remove path and file extension - Stack Overflow answered and edited by Ananda Mahto on Feb 25 2013. See <https://stackoverflow.com/questions/15073753/regex-return-file-name-remove-path-and-file-extension/15073919>.

## Examples

```
## Not run:
# Example to check the input file format
library(ie2misc)

# Copy and paste the following code into the R console if you
# wish to see the .exp input file format.
# Note the number of lines and the row headings.
file.show(system.file("extdata", "01110000_PEAK_WATSTORE.EXP",
  package = "ie2misc"), title = paste("01110000_PEAK_WATSTORE.EXP"))
# opens the .exp file using the default text editor or within RStudio

# Examples to show you different output cases
expFileOutput(output = "csv") # returns .csv files
# Follow the file dialog instructions

expFileOutput(output = "xlsx") # returns .xlsx files
# Follow the file dialog instructions

expFileOutput(output = "both") # returns .csv and .xlsx files
```

```
# Follow the file dialog instructions

# Examples to show you different output cases (BATCH)
expFileOutputBATCH(output = "csv") # returns .csv files
# Follow the file dialog instructions

expFileOutputBATCH(output = "xlsx") # returns .xlsx files
# Follow the file dialog instructions

expFileOutputBATCH(output = "both") # returns .csv and .xlsx files
# Follow the file dialog instructions

## End(Not run)
```

---

madstat

*Mean-absolute deviation (MAD)*


---

## Description

This function computes the mean-absolute deviation (MAD) – "the average of the magnitudes of the errors or deviations."

## Usage

```
madstat(observed, na.rm = FALSE)
```

## Arguments

observed	numeric vector, matrix, data.frame, or data.table that contains the observed data points.
na.rm	logical vector that determines whether the missing values should be removed or not.

## Details

MAD is expressed as

$$n^{-1} \sum_{i=1}^n |O_i - \bar{O}|$$

$n$  the number of observations

$O$  the "pairwise-matched observations that are judged to be reliable"

$\bar{O}$  the "true" mean of the observations

Reference 1 fully discusses MAD, while Reference 2 provides the formula used to calculate the MAD.

## Value

mean-absolute deviation (MAD) as a numeric [vector](#) or a named numeric vector if using a named object ([matrix](#), [data.frame](#), or [data.table](#)). MAD has the same units as the observed values. The default choice is that any NA values will be kept (`na.rm = FALSE`). This can be changed by specifying `na.rm = TRUE`, such as `madstat(obs, na.rm = TRUE)`.

## References

1. Cort J. Willmott, Kenji Matsuura, and Scott M. Robeson, "Ambiguities inherent in sums-of-squares-based error statistics", *Atmospheric Environment*, vol. 43, no. 3, pp. 749-752, 2009, <https://www.sciencedirect.com/science/article/pii/S1352231008009564>.
2. Cort J. Willmott, Scott M. Robeson, and Kenji Matsuura, "Short Communication: A refined index of model performance", *International Journal of Climatology*, Volume 32, Issue 13, pages 2088-2094, 15 November 2012, article from ResearchGate: [https://www.researchgate.net/publication/235961403\\_A\\_refined\\_index\\_of\\_model\\_performance](https://www.researchgate.net/publication/235961403_A_refined_index_of_model_performance).
3. Nathabandu T. Kottegoda and Renzo Rosso, *Statistics, Probability, and Reliability for Civil and Environmental Engineers*, New York City, New York: The McGraw-Hill Companies, Inc., 1997, page 15.

## See Also

[mad](#) for median absolute deviation (MAD)

[mape](#) for mean absolute percent error (MAPE), [mae](#) for mean-absolute error (MAE), [dr](#) for "index of agreement (dr)", [vnse](#) for Nash-Sutcliffe model efficiency (NSE), and [rmse](#) for root mean square error (RMSE).

## Examples

```
library(ie2misc)

# Example 1.18 from Kottegoda (page 15)
obs <- c(50, 56, 42, 53, 49) # annual rainfall in cm
madstat(obs)

library(rando)

set_n(100) # makes the example reproducible
obs1 <- r_norm(.seed = 300) # observed

# using the numeric vector obs1
madstat(obs1)

# using a matrix of the numeric vector obs1
mat1 <- matrix(data = obs1, nrow = length(obs1), ncol = 1, byrow = FALSE,
               dimnames = list(c(rep("", length(obs1))), "Observed"))
madstat(mat1)
```

```
# using a data.frame of the numeric vector obs1
df1 <- data.frame(obs1)
madstat(df1)

library(data.table)

# using a data.table of the numeric vector obs1
df2 <- data.table(obs1)
madstat(df2)
```

---

mae	<i>Mean-absolute error (MAE)</i>
-----	----------------------------------

---

### Description

This function computes the mean-absolute error (MAE).

### Usage

```
mae(predicted, observed, na.rm = FALSE)
```

### Arguments

predicted	numeric vector that contains the model predicted data points (1st parameter)
observed	numeric vector that contains the observed data points (2nd parameter)
na.rm	logical vector that determines whether the missing values should be removed or not.

### Details

(MAE) is expressed as

$$n^{-1} \sum_{i=1}^n |P_i - O_i|$$

$n$  the number of observations

$P$  the "model estimates or predictions"

$O$  the "thought-to-be reliable and pairwise matched observations"

MAE is fully discussed in the Willmott reference, including a comparison to root mean square error (RMSE).

**Value**

mean-absolute error (MAE) as a numeric vector using the same units as the given variables. The default choice is that any NA values will be kept (`na.rm = FALSE`). This can be changed by specifying `na.rm = TRUE`, such as `mae(pre, obs, na.rm = TRUE)`.

**References**

Cort J. Willmott and Kenji Matsuura, "Advantages of the mean-absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance", *Climate Research*, Vol. 30: 79-82, 2005, [https://web.archive.org/web/20230119121852/climate.geog.udel.edu/~climate/publication\\_html/Pdf/WM\\_CR\\_05.pdf](https://web.archive.org/web/20230119121852/climate.geog.udel.edu/~climate/publication_html/Pdf/WM_CR_05.pdf). Retrieved thanks to the Internet Archive: Way-back Machine

**See Also**

`mape` for mean absolute percent error (MAPE), `madstat` for mean-absolute deviation (MAD), `dr` for "index of agreement (dr)", `vnse` for Nash-Sutcliffe model efficiency (NSE), and `rmse` for root mean square error (RMSE).

**Examples**

```
library(ie2misc)

obs <- 1:10 # observed
pre <- 2:11 # predicted
mae(pre, obs)

library(rando)

set_n(100) # makes the example reproducible
obs1 <- r_norm(.seed = 103) # observed
pre1 <- r_norm(.seed = 102) # predicted

# using the vectors pre1 and obs1
mae(pre1, obs1)

# using a matrix of the numeric vectors pre1 and obs1
mat1 <- matrix(data = c(obs1, pre1), nrow = length(pre1), ncol = 2,
  byrow = FALSE, dimnames = list(c(rep("", length(pre1))),
  c("Predicted", "Observed")))
mae(mat1[, 2], mat1[, 1])

# mat1[, 1] # observed values from column 1 of mat1
# mat1[, 2] # predicted values from column 2 of mat1

# using a data.frame of the numeric vectors pre1 and obs1
df1 <- data.frame(obs1, pre1)
```

```

mae(df1[, 2], df1[, 1])

# df1[, 1] # observed values from column 1 of df1
# df1[, 2] # predicted values from column 2 of df1

library(data.table)

# using a data.table of the numeric vectors pre1 and obs1
df2 <- data.table(obs1, pre1)
mae(df2[, 2, with = FALSE][[1]], df2[, 1, with = FALSE][[1]])

# df2[, 1, with = FALSE][[1]] # observed values from column 1 of df2
# df2[, 2, with = FALSE][[1]] # predicted values from column 2 of df2

```

---

mape

*Mean absolute percent error (MAPE)*


---

### Description

This function computes the mean absolute percent error (MAPE).

### Usage

```
mape(predicted, observed, na.rm = FALSE)
```

### Arguments

predicted	numeric vector that contains the predicted data points (1st parameter)
observed	numeric vector that contains the observed data points (2nd parameter)
na.rm	logical vector that determines whether the missing values should be removed or not.

### Details

MAPE is expressed as

$$\frac{1}{n} \sum_{i=1}^n 100 \frac{|X_i - Y_i|}{X_i}$$

$n$  the number of observations

$X$  the observations

$Y$  the predictions

Below are some points to remember about MAPE from the Ji reference:

1. MAPE is "a measure to validate forecast models",
2. MAPE is "a standardized value and is independent of the unit of the measurement",
3. MAPE is "meaningful only if all  $X_i$  values are positive",
4. MAPE is "unstable when  $X_i$  values are near zero", and
5. "If X and Y are interchanged, the MAPE will result in a different value."

### Value

mean absolute percent error (MAPE) as a numeric vector. The default choice is that any NA values will be kept (`na.rm = FALSE`). This can be changed by specifying `na.rm = TRUE`, such as `mape(pre, obs, na.rm = TRUE)`.

### References

Lei Ji and Kevin Gallo, "An Agreement Coefficient for Image Comparison", *Photogrammetric Engineering & Remote Sensing*, Vol. 72, No. 7, July 2006, p. 823-8335, <https://www.ingentaconnect.com/content/asprs/pers/2006/00000072/00000007/art00006>.

### See Also

`mae` for mean-absolute error (MAE), `madstat` for mean-absolute deviation (MAD), `dr` for "index of agreement (dr)", `vnse` for Nash-Sutcliffe model efficiency (NSE), and `rmse` for root mean square error (RMSE).

### Examples

```
library(ie2misc)

obs <- 1:10 # observed
pre <- 2:11 # predicted
mape(pre, obs)

library(rando)

set_n(100) # makes the example reproducible
obs1 <- r_norm(.seed = 109) # observed
pre1 <- r_norm(.seed = 124) # predicted

# using the vectors pre1 and obs1
mape(pre1, obs1)

# using a matrix of the numeric vectors pre1 and obs1
mat1 <- matrix(data = c(obs1, pre1), nrow = length(pre1), ncol = 2,
               byrow = FALSE, dimnames = list(c(rep("", length(pre1))),
               c("Predicted", "Observed")))
mape(mat1[, 2], mat1[, 1])
```

```

# mat1[, 1] # observed values from column 1 of mat1
# mat1[, 2] # predicted values from column 2 of mat1

# using a data.frame of the numeric vectors pre1 and obs1
df1 <- data.frame(obs1, pre1)
mape(df1[, 2], df1[, 1])

# df1[, 1] # observed values from column 1 of df1
# df1[, 2] # predicted values from column 2 of df1

library(data.table)

# using a data.table of the numeric vectors pre1 and obs1
df2 <- data.table(obs1, pre1)
mape(df2[, 2, with = FALSE][[1]], df2[, 1, with = FALSE][[1]])

# df2[, 1, with = FALSE][[1]] # observed values from column 1 of df2
# df2[, 2, with = FALSE][[1]] # predicted values from column 2 of df2

```

---

psfFileChange

*psfFileChange and psfFileChangeBATCH*


---

## Description

The .psf file is a script file that records the specifications used to run the USGS PeakFQ program (<https://www.usgs.gov/tools/peakfq>) for a station. psfFileChange and psfFileChangeBATCH modify the original .psf settings.

## Usage

```

psfFileChange(
  file = tk_choose.files(default = "", caption =
    "Select file(s) to open & hold down Ctrl to choose more than 1 file", multi = TRUE,
    filters = matrix(c("Text file", ".psf", "Text file", ".PSF"), 4, 2, byrow = TRUE)),
  interactive = TRUE
)

psfFileChangeBATCH(
  path = tk_choose.dir(caption = "Select the directory with the .psf files")
)

```

## Arguments

file	Input .psf file(s) to change specific SkewSE, GenSkew, and SkewOpt information to be selected through a file dialog.
interactive	If interactive is TRUE, then the user will select the filenames(s) to use for saving with the file dialog. In order to select more than one file, the user must hold down the Ctrl (Control) button while mouse clicking the chosen files. If interactive is FALSE, then the user will select the directory, via the directory dialog, to use for saving and the original filenames will be used.
path	Directory path of .psf files, to be selected through a directory dialog, to change specific SkewSE, GenSkew, and SkewOpt information. The user will be asked where to find the .psf files & then the user will be asked where to save the revised .psf files.

## Details

psfFileChange searches for a character vector of patterns (SkewSE, GenSkew, and SkewOpt) in single or multiple .psf file(s) to replace. If the patterns are missing, then the patterns are added to the file(s). This is done for a single file or multiple files that the user selects. Although these changes are currently pre-determined, future versions may allow the user to change particular settings (for example, how outliers are handled).

psfFileChangeBATCH searches for a character vector of patterns (SkewSE, GenSkew, and SkewOpt) in a directory of .psf files to replace. If the patterns are missing, then the patterns are added to the files. This is done in a BATCH mode (whole directory of .psf files). Although these changes are currently pre-determined, future versions may allow the user to change particular settings (for example, how outliers are handled).

## Value

Revised .psf text file(s)

## Author(s)

Irucka Embry, Anne Hoos

## Source

1. r - How can I check if a file is empty? - Stack Overflow answered by Konrad Rudolph and edited by Geekuna Matata on Apr 23 2014. See <https://stackoverflow.com/questions/23254002/how-can-i-check-if-a-file-is-empty>.
2. r - Better error message for stopifnot? - Stack Overflow answered by Andrie on Dec 1 2011. See <https://stackoverflow.com/questions/8343509/better-error-message-for-stopifnot>.
3. RDocumentation: TclInterface tcltk. See <https://www.rdocumentation.org/packages/tcltk/versions/3.3.1>.
4. James Wettenhall & Philippe Grosjean, File Open/Save dialogs in R tcltk, December 01, 2015. See <https://web.archive.org/web/20160521051207/http://www.sciviews.org/recipes/tcltk/TclTk-file-open-save-dialogs/>. Retrieved thanks to the Internet Archive: Way-back Machine

5. Replacing nth line in a text file in R - Stack Overflow answered by Spacedman on Aug 1 2012. See <https://stackoverflow.com/questions/11756353/replacing-nth-line-in-a-text-file-in-r>.
6. r - read csv files and perform function, then bind together - Stack Overflow answered by bjoeseh on Jan 8 2015. See <https://stackoverflow.com/questions/27846715/read-csv-files-and-perform-t>
7. multiple output filenames in R - Stack Overflow asked and edited by Gabelins on Feb 1 2013. See <https://stackoverflow.com/questions/14651594/multiple-output-filenames-in-r>.
8. r - Regex return file name, remove path and file extension - Stack Overflow answered and edited by Ananda Mahto on Feb 25 2013. See <https://stackoverflow.com/questions/15073753/regex-return-file-name-remove-path-and-file-extension/15073919>.

## Examples

```
## Not run:
# Examples to change (a) .psf file(s) interactively and non-interactively

library(ie2misc)

psfFileChange() # default where interactive = TRUE
# Follow the file dialog instructions

# These are the rows that have been added or changed as a result of this function:

# SkewSE 0.361804179633127
# GenSkew 0.0104293904
# SkewOpt Weighted

psfFileChange(interactive = FALSE)
# Follow the file dialog instructions

# These are the rows that have been added or changed as a result of this function:

# SkewSE 0.361804179633127
# GenSkew 0.0104293904
# SkewOpt Weighted

psfFileChangeBATCH() # Follow the file dialog instructions

# These are the rows that have been added or changed as a result of this function:

# SkewSE 0.361804179633127
# GenSkew 0.0104293904
# SkewOpt Weighted

## End(Not run)
```

---

 qw

*qw, qw2 and qwBATCH*


---

### Description

qw, qw2 and qwBATCH process raw QW files. The QW files can contain "selected water-quality data for stations in the U.S. Geological Survey (USGS) National Water Information System (NWIS) water-quality database. The data you have secured from the USGS NWISWeb database may include data that have not received Director's approval and as such are provisional and subject to revision."

### Usage

```
qw(
  file = tk_choose.files(default = "", caption =
    "Select file(s) to open & hold down Ctrl to choose more than 1 file", multi = TRUE,
    filters = matrix(c("QW file", "*.*", "QW file", ".rdb", "QW file", ".RDB"), 6, 2, byrow
      = TRUE)),
  interactive = TRUE,
  overwrite = TRUE
)

qwBATCH(
  path = tk_choose.dir(caption = "Select directory with the QW files"),
  pattern = "*.rdb",
  overwrite = TRUE
)

qw2(file, overwrite = TRUE)
```

### Arguments

file	Input QW file(s) to be selected through a file dialog.
interactive	If interactive is TRUE, then the user will select the filenames(s) to use for saving with the file dialog. In order to select more than one file, the user must hold down the Ctrl (Control) button while mouse clicking the chosen files. If interactive is FALSE, then the user will select the directory, via the directory dialog, to use for saving and the original filenames will be used.
overwrite	If TRUE, overwrite any existing spreadsheet.
path	Directory path of QW files to be selected through a directory dialog. The user will be asked where to find the QW files & then the user will be asked where to save the QW .xlsx files.
pattern	The default pattern is .rdb (the filename has the .rdb extension). This pattern can be changed by qwBATCH(pattern = "pattern").

## Details

qw function opens a single or multiple raw, QW file(s) to modify the format and then exports the file(s) in .xlsx format. This is done for a single file or multiple files that the user selects with a file dialog.

qw2 function opens a single QW file to modify the format and then exports the file in .xlsx format. This is done for a single file that the user selects without a file dialog.

qwBATCH function opens raw QW files, from a directory, to modify the format and then exports the files in .xlsx format. This is done in a BATCH mode (whole directory of QW files) using a directory dialog.

qw, qw2 and qwBATCH functions perform the same processes on the raw QW files: 1) remove the first row, 2) create a single column with the Date and Time rather than 2 separate Date and Time columns (if needed), and 3) exports the QW file as a spreadsheet with three sheets (sheet1 corrects the timezones for Daylight Savings Times, sheet2 includes everything except for the meta data, and sheet3 provides the meta data from the top of the file). The supported US timezones are the following: Eastern, Central, Mountain, Pacific, Alaska, Hawai'i, and Atlantic (Puerto Rico and US Virgin Islands).

## Value

QW .xlsx file(s).

## Source

1. r - How can I check if a file is empty? - Stack Overflow answered by Konrad Rudolph and edited by Geekuna Matata on Apr 23 2014. See <https://stackoverflow.com/questions/23254002/how-can-i-check-if-a-file-is-empty>.
2. r - Better error message for stopifnot? - Stack Overflow answered by Andrie on Dec 1 2011. See <https://stackoverflow.com/questions/8343509/better-error-message-for-stopifnot>.
3. RDocumentation: TclInterface tcltk. See <https://www.rdocumentation.org/packages/tcltk/versions/3.3.1>.
4. James Wettenhall & Philippe Grosjean, File Open/Save dialogs in R tcltk, December 01, 2015. See <https://web.archive.org/web/20160521051207/http://www.sciviews.org/recipes/tcltk/TclTk-file-open-save-dialogs/>. Retrieved thanks to the Internet Archive: Way-back Machine
5. r - read csv files and perform function, then bind together - Stack Overflow answered by bjoseph on Jan 8 2015. See <https://stackoverflow.com/questions/27846715/read-csv-files-and-perform-f>
6. r - Convert column classes in data.table - Stack Overflow answered by Matt Dowle on Dec 27 2013. See <https://stackoverflow.com/questions/7813578/convert-column-classes-in-data-table>.
7. Does column exist and how to rearrange columns in R data frame - Stack Overflow answered and edited by Peter McMahan on Aug 2 2009. See <https://stackoverflow.com/questions/1177919/does-column-exist-and-how-to-rearrange-columns-in-r-data-frame>.
8. time - Dealing with timestamps in R - Stack Overflow answered by Dirk Eddelbuettel on Dec 26 2009. See <https://stackoverflow.com/questions/1962278/dealing-with-timestamps-in-r/1962336>.

9. R help - How to change the default Date format for write.csv function? answered by William Dunlap on Dec 28, 2009. See <https://hypatia.math.ethz.ch/pipermail/r-help/2009-December/416010.html>.
10. RDocumentation: strptime base. See <https://www.rdocumentation.org/packages/base/versions/3.3.1/topics/strptime>.
11. National Water Information System: Help System Time Zone Codes. See [https://help.waterdata.usgs.gov/code/tz\\_query?fmt=html](https://help.waterdata.usgs.gov/code/tz_query?fmt=html).
12. multiple output filenames in R - Stack Overflow asked and edited by Gabelins on Feb 1 2013. See <https://stackoverflow.com/questions/14651594/multiple-output-filenames-in-r>.
13. r - Regex return file name, remove path and file extension - Stack Overflow answered and edited by Ananda Mahto on Feb 25 2013. See <https://stackoverflow.com/questions/15073753/regex-return-file-name-remove-path-and-file-extension/15073919>.
14. warnings - How do I get rid of the NOTE's generated by R CMD check when using for example dply in my package? - Stack Overflow answered by shadow on Mar 4 2015 and edited by shadow on Mar 5 2015. See <https://stackoverflow.com/questions/28851812/how-do-i-get-rid-of-the-not>
15. trinker/qdapRegex - dictionary\_maintenance.R. See [https://github.com/trinker/qdapRegex/blob/master/inst/dictionary\\_maintenance.R](https://github.com/trinker/qdapRegex/blob/master/inst/dictionary_maintenance.R).
16. excel - Interconverting POSIXct and numeric in R - Stack Overflow by LauraS on May 6 2016. See <https://stackoverflow.com/questions/37078772/interconverting-posixct-and-numeric-in-r>.
17. convert date and time string to POSIX in R - Stack Overflow commented by cryo111 on Sep 18 2013. See <https://stackoverflow.com/questions/18874400/convert-date-and-time-string-to-posix-in-18874863>.

## Examples

```
## Not run:
library(ie2misc)
# Examples to change (an) QW file(s) interactively and non-interactively
file1 <- "https://waterdata.usgs.gov/nwis/dv?cb_00060=on&format=rdb&site_no=03584500"
file2 <- "&period=&begin_date=1904-07-01&end_date=2016-06-22"
file3 <- paste0(file1, file2) # used to truncate the file name
qw2(file3)
# USGS 03584500 ELK RIVER NEAR PROSPECT, TN
# Discharge, cubic feet per second (Mean)

qw() # default where interactive = TRUE
# Follow the file dialog instructions

qw(interactive = FALSE)
# Follow the file dialog instructions

# Example to change QW files in batch mode
qwBATCH()
# Follow the file dialog instructions
```

```
## End(Not run)
```

---

rmse	<i>Root mean square error (RMSE)</i>
------	--------------------------------------

---

### Description

This function computes the root mean square error (RMSE).

### Usage

```
rmse(predicted, observed, na.rm = FALSE)
```

### Arguments

predicted	numeric vector that contains the predicted data points (1st parameter)
observed	numeric vector that contains the observed data points (2nd parameter)
na.rm	logical vector that determines whether the missing values should be removed or not.

### Details

RMSE is expressed as

$$\sqrt{n^{-1} \sum_{i=1}^n |P_i - O_i|^2}$$

$n$  the number of observations

$P$  the "model estimates or predictions"

$O$  the "thought-to-be reliable and pairwise matched observations"

RMSE is fully discussed in the Willmott reference, including a comparison to mean-absolute error (MAE).

### Value

mean absolute percent error (RMSE) as a numeric vector. The default choice is that any NA values will be kept (na.rm = FALSE). This can be changed by specifying na.rm = TRUE, such as rmse(pre, obs, na.rm = TRUE).

## References

Cort J. Willmott and Kenji Matsuura, "Advantages of the mean-absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance", *Climate Research*, Vol. 30: 79-82, 2005, [https://web.archive.org/web/20230119121852/climate.geog.udel.edu/~climate/publication\\_html/Pdf/WM\\_CR\\_05.pdf](https://web.archive.org/web/20230119121852/climate.geog.udel.edu/~climate/publication_html/Pdf/WM_CR_05.pdf). Retrieved thanks to the Internet Archive: Way-back Machine

## See Also

`mape` for mean absolute percent error (MAPE), `mae` for mean-absolute error (MAE), `madstat` for mean-absolute deviation (MAD), `dr` for "index of agreement (dr)", and `vNSE` for Nash-Sutcliffe model efficiency (NSE).

## Examples

```
library(ie2misc)

obs <- 1:10 # observed
pre <- 2:11 # predicted
rmse(pre, obs)

library(rando)

set_n(100) # makes the example reproducible
obs1 <- r_norm(.seed = 209) # observed
pre1 <- r_norm(.seed = 224) # predicted

# using the vectors pre1 and obs1
rmse(pre1, obs1)

# using a matrix of the numeric vectors pre1 and obs1
mat1 <- matrix(data = c(obs1, pre1), nrow = length(pre1), ncol = 2,
  byrow = FALSE, dimnames = list(c(rep("", length(pre1))),
  c("Predicted", "Observed")))
rmse(mat1[, 2], mat1[, 1])

# mat1[, 1] # observed values from column 1 of mat1
# mat1[, 2] # predicted values from column 2 of mat1

# using a data.frame of the numeric vectors pre1 and obs1
df1 <- data.frame(obs1, pre1)
rmse(df1[, 2], df1[, 1])

# df1[, 1] # observed values from column 1 of df1
# df1[, 2] # predicted values from column 2 of df1
```

```

library(data.table)

# using a data.table of the numeric vectors pre1 and obs1
df2 <- data.table(obs1, pre1)
rmse(df2[, 2, with = FALSE][[1]], df2[, 1, with = FALSE][[1]])

# df2[, 1, with = FALSE][[1]] # observed values from column 1 of df2
# df2[, 2, with = FALSE][[1]] # predicted values from column 2 of df2

```

---

sat_enthalpy	<i>Saturated Enthalpy (H)</i>
--------------	-------------------------------

---

### Description

sat\_enthalpy and sat\_enthalpy2 solve for the saturated enthalpy (H) given an elevation in feet and a range of temperature values (degrees F).

### Usage

```

sat_enthalpy(
  file = tk_choose.files(default = "", caption = "Select file to open", multi = FALSE,
    filters = matrix(c("Comma-separated value file", ".csv", "MS Excel spreadsheet",
      ".xlsx", "MS Excel 97-2003 spreadsheet", ".xls"), 6, 2, byrow = TRUE)),
  sheet = 1,
  overwrite = TRUE,
  output = c("console", "csv", "xlsx")
)

sat_enthalpy2(
  file = NULL,
  sheet = 1,
  elevation = NULL,
  tbegin = NULL,
  tend = NULL,
  tincrement = NULL,
  overwrite = TRUE,
  output = c("console", "csv", "xlsx")
)

```

### Arguments

file	Input file (.xls, .xlsx, or .csv)
sheet	Sheet number or sheet name for the input file (default is sheet 1)
overwrite	If TRUE, overwrite any existing spreadsheet.

output	The output format of the resulting data.table (console, .csv, or .xlsx)
elevation	numeric vector that contains the location elevation in feet
tbegin	numeric vector that contains the beginning temperature in degrees F
tend	numeric vector that contains the ending temperature in degrees F
tincrement	numeric vector that contains the temperature increment in degrees F

### Details

The sat\_enthalpy function provides a file dialog for the user to choose the selected file.

The sat\_enthalpy2 function either allows the user to provide the elevation and temperature values without a file or allows the user to provide a file without using a file dialog.

H is expressed as

$$H = (0.24 \times T) + [W \times (1061 + 0.444 \times T)]$$

**H** enthalpy (Btu/lb)

**T** dry-bulb temperature (degrees F)

**W** specific humidity (lbwater / lbdry air)

where W or SH (Specific Humidity) is expressed as

$$W = (0.622 \times E) / (Mb - (0.378 \times E))$$

**W** W or SH – Specific Humidity (kg/kg)

**E** Vapor pressure in saturated air at this temperature (millibars)

**Mb** Pressure (millibars)

### Value

data.table with the output parameters displayed on the console or as a spreadsheet

### Author(s)

Irucka Embry [R version], Timothy H. Diehl [Microsoft Excel(R) spreadsheet]

### Source

1. r - Better error message for stopifnot? - Stack Overflow answered by Andrie on Dec 1 2011. See <https://stackoverflow.com/questions/8343509/better-error-message-for-stopifnot>.
2. How to check file extensions in R - Stack Overflow answered by lebatnok on May 17 2014. See <https://stackoverflow.com/questions/23713284/how-to-check-file-extensions-in-r>.
3. RDocumentation: TclInterface tcltk. See <https://www.rdocumentation.org/packages/tcltk/versions/3.3.1>.

4. James Wettenhall & Philippe Grosjean, File Open/Save dialogs in R tcltk, December 01, 2015. See <https://web.archive.org/web/20160521051207/http://www.sciviews.org/recipes/tcltk/TclTk-file-open-save-dialogs/>. Retrieved thanks to the Internet Archive: Wayback Machine
5. multiple output filenames in R - Stack Overflow asked and edited by Gabelins on Feb 1 2013. See <https://stackoverflow.com/questions/14651594/multiple-output-filenames-in-r>.
6. r - Regex return file name, remove path and file extension - Stack Overflow answered and edited by Ananda Mahto on Feb 25 20134. See <https://stackoverflow.com/questions/15073753/regex-return-file-name-remove-path-and-file-extension/15073919>.

## References

1. "Andrew Revering's List of Meteorological Formulas", <https://web.archive.org/web/20131222174626/https://aprweather.com/pages/calc.htm>. Retrieved thanks to the Internet Archive: Wayback Machine
2. Functional Testing and Design Guides, *Functional Testing Guide: from the Fundamentals to the Field*, "Sidebar 3: How to Calculate Enthalpy", <https://web.archive.org/web/20150216015428/https://www.ftguide.org/ftg/IntegratedOperation/IOC-Sidebars-1-3/IOC-SB3-How-To-Calculate-Enthalpy.htm>. Retrieved thanks to the Internet Archive: Wayback Machine

## Examples

```
## Not run:
library(ie2misc)
# Example to check the input file format

# Copy and paste the following code into the R console if you
# wish to see the input file format.
library(openxlsx)
openXL(system.file("extdata", "Saturated_Enthalpy_Example_Input.xlsx",
package = "ie2misc"))
# opens the workbook using the default spreadsheet application

# Examples to show different use cases

sat_enthalpy(output = "csv")
# The sheet number is 1 (default) and the output is csv

sat_enthalpy2(system.file("extdata", "Saturated_Enthalpy_Example_Input.xlsx"
, package = "ie2misc"), output = "xlsx")
# The sheet number is 1 (default) and the output is xlsx

sat_enthalpy2(system.file("extdata", "Saturated_Enthalpy_Example_Input.csv",
package = "ie2misc"), output = "console")
# The sheet number is 1 (default) and the output is the console
```

```

sat_enthalpy2(elevation = 1200, tbegin = 32, tend = 180,
tincrement = 0.01, output = "csv")
# elevation = 1200 feet, tbegin = 32 degrees F, tend = 180 degrees F,
# tincrement = 0.01 degrees F

## End(Not run)

```

---

vnse

*Nash-Sutcliffe model efficiency (NSE)*


---

### Description

This function computes the Nash-Sutcliffe model efficiency (NSE) or "Nash and Sutcliffe's coefficient of efficiency (E)".

### Usage

```
vnse(predicted, observed, na.rm = FALSE)
```

### Arguments

predicted	numeric vector that contains the model predicted data points (1st parameter)
observed	numeric vector that contains the observed data points (2nd parameter)
na.rm	logical vector that determines whether the missing values should be removed or not.

### Details

NSE or E is expressed as

$$E = 1 - \frac{\sum_{i=1}^n (P_i - O_i)^2}{\sum_{i=1}^n (O_i - \bar{O})^2}$$

**E** "Nash and Sutcliffe's coefficient of efficiency (E)"

**n** the number of observations

**P** the "model estimates or predictions"

**O** the "pairwise-matched observations that are judged to be reliable"

$\bar{O}$  the "true" mean of the observations

Note: Both P and O should have the same units.

"Nash and Sutcliffe's coefficient of efficiency (E)" and other "dimensionless measures of average error" are fully discussed in the Willmott reference.

## Value

Nash-Sutcliffe model efficiency (NSE) as a numeric vector. The default choice is that any NA values will be kept (`na.rm = FALSE`). This can be changed by specifying `na.rm = TRUE`, such as `vnse(pre, obs, na.rm = TRUE)`.

## References

Cort J. Willmott, Scott M. Robeson, and Kenji Matsuura, "A refined index of model performance", *International Journal of Climatology*, Volume 32, Issue 13, pages 2088-2094, 15 November 2012, article from ResearchGate: [https://www.researchgate.net/publication/235961403\\_A\\_refined\\_index\\_of\\_model\\_performance](https://www.researchgate.net/publication/235961403_A_refined_index_of_model_performance).

## See Also

[mape](#) for mean absolute percent error (MAPE), [mae](#) for mean-absolute error (MAE), [madstat](#) for mean-absolute deviation (MAD), [dr](#) for "index of agreement (dr)", and [rmse](#) for root mean square error (RMSE).

## Examples

```
library(ie2misc)

obs <- 1:10 # observed
pre <- 2:11 # predicted
vnse(pre, obs)

library(rando)

set_n(100) # makes the example reproducible
obs1 <- r_norm(.seed = 609) # observed
pre1 <- r_norm(.seed = 624) # predicted

# using the vectors pre1 and obs1
vnse(pre1, obs1)

# using a matrix of the numeric vectors pre1 and obs1
mat1 <- matrix(data = c(obs1, pre1), nrow = length(pre1), ncol = 2,
  byrow = FALSE, dimnames = list(c(rep("", length(pre1))),
  c("Predicted", "Observed")))
vnse(mat1[, 2], mat1[, 1])

# mat1[, 1] # observed values from column 1 of mat1
# mat1[, 2] # predicted values from column 2 of mat1

# using a data.frame of the numeric vectors pre1 and obs1
df1 <- data.frame(obs1, pre1)
vnse(df1[, 2], df1[, 1])
```

```

# df1[, 1] # observed values from column 1 of df1
# df1[, 2] # predicted values from column 2 of df1

library(data.table)

# using a data.table of the numeric vectors pre1 and obs1
df2 <- data.table(obs1, pre1)
vnse(df2[, 2, with = FALSE][[1]], df2[, 1, with = FALSE][[1]])

# df2[, 1, with = FALSE][[1]] # observed values from column 1 of df2
# df2[, 2, with = FALSE][[1]] # predicted values from column 2 of df2

```

---

%+na%	%+na%
-------	-------

---

**Description**

This function "defines a dyadic operator that will behave differently than the "+" operator with regards to missing values". (Quote by 42- in Source 1 below). In this case, add x and y so that NA is ignored when x is a number and x = NA when x = 0.

**Usage**

```
x %+na% y
```

**Arguments**

- x                    numeric vector that does not contain any NA values
- y                    numeric vector which contains at least one NA value

**Value**

numeric vector resulting from the sum of x and y

**Source**

1. Using ‘:=’ in data.table to sum the values of two columns in R, ignoring NAs - Stack Overflow answered by 42- on Oct 28 2012. See <https://stackoverflow.com/questions/13106645/using-in-data-table-to-sum-the-values-of-two-columns-in-r-ignoring-nas>.
2. R - merge unequal regular (15 min intervals) and irregular time series within 10 minutes - Stack Overflow asked by iembry on Jan 12 2015 and edited by iembry on Feb 2 2015. See <https://stackoverflow.com/questions/27911643/r-merge-unequal-regular-15-min-intervals-and-irre>

3. r - How to not run an example using roxygen2? - Stack Overflow answered and edited by samkart on Jul 9 2017. (Also see the additional comments in response to the answer.) See <https://stackoverflow.com/questions/12038160/how-to-not-run-an-example-using-roxygen2>.
4. devtools - Issues in R package after CRAN asked to replace dontrun by dontrun - Stack Overflow answered by Hong Ooi on Sep 1 2020. (Also see the additional comments in response to the answer.) See <https://stackoverflow.com/questions/63693563/issues-in-r-package-after-cran-asked>.

## Examples

```
library(ie2misc)

a <- 1:10
b <- c(98:106, NA)

a %+na% b

## Not run:
# See Source 3 and Source 4
# Note that if the vector with the NA value(s) is first, then an error
# message will be displayed as now there are checks for both x and y.

a <- 1:10
b <- c(98:106, NA)

try(b %+na% a)

## End(Not run)

a1 <- 1:12
b1 <- c(98:106, rep(NA, 3))

a1 %+na% b1

# using a matrix of the numeric vectors a and b
mat1 <- matrix(data = c(a, b), nrow = length(b), ncol = 2, byrow = FALSE,
               dimnames = list(c(rep("", length(b))), c("a", "b")))
mat1[, 1] %+na% mat1[, 2]

# using a data.frame of the numeric vectors a and b
df1 <- data.frame(a, b)
df1[, 1] %+na% df1[, 2]

library(data.table)

# using a data.table of the numeric vectors a and b

a <- 1:10
```

*%+na%*

31

```
b <- c(98:106, NA)
```

```
df2 <- data.table(a, b)
```

```
df2[, 1, with = FALSE][[1]] %+na% df2[, 2, with = FALSE][[1]]
```

# Index

`%+na%`, [29](#)

`adaps`, [2](#)

`adaps2 (adaps)`, [2](#)

`adapsBATCH (adaps)`, [2](#)

`data.frame`, [11](#)

`data.table`, [11](#)

`dr`, [5](#), [11](#), [13](#), [15](#), [23](#), [28](#)

`expFileOutput`, [7](#)

`expFileOutputBATCH (expFileOutput)`, [7](#)

`mad`, [11](#)

`madstat`, [6](#), [10](#), [13](#), [15](#), [23](#), [28](#)

`mae`, [6](#), [11](#), [12](#), [15](#), [23](#), [28](#)

`mape`, [6](#), [11](#), [13](#), [14](#), [23](#), [28](#)

`matrix`, [11](#)

`psfFileChange`, [16](#)

`psfFileChangeBATCH (psfFileChange)`, [16](#)

`qw`, [19](#)

`qw2 (qw)`, [19](#)

`qwBATCH (qw)`, [19](#)

`rmse`, [6](#), [11](#), [13](#), [15](#), [22](#), [28](#)

`sat_enthalpy`, [24](#)

`sat_enthalpy2 (sat_enthalpy)`, [24](#)

`vector`, [11](#)

`vnse`, [6](#), [11](#), [13](#), [15](#), [23](#), [27](#)