

Package ‘growthcleanr’

September 1, 2022

Type Package

Title Data Cleaner for Anthropometric Measurements

Version 2.0.1

Description Identifies implausible anthropometric (e.g., height, weight) measurements in irregularly spaced longitudinal datasets, such as those from electronic health records.

URL <https://carriedaymont.github.io/growthcleanr/index.html>,
<https://github.com/carriedaymont/growthcleanr>

BugReports <https://github.com/carriedaymont/growthcleanr/issues>

Imports R.utils (>= 2.11.0), data.table (>= 1.13.0), tidyr (>= 1.1.0),
plyr (>= 1.8.6), dplyr (>= 1.0.1), foreach (>= 1.5.0),
doParallel (>= 1.0.15), labelled (>= 2.5.0), magrittr (>= 1.5)

Depends R (>= 2.10)

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Suggests argparser (>= 0.6), bit64 (>= 4.0.2), knitr (>= 1.29),
rmarkdown (>= 2.3), testthat (>= 2.3.2)

RoxygenNote 7.2.1

NeedsCompilation no

Author Daymont Carrie [aut],
Grundmeier Robert [aut],
Miller Jeffrey [aut],
Campos Diego [aut],
Chudnov Dan [ctb],
De los Santos Hannah [ctb, cre],
Cao Lusha [ctb],
Silva Steffani [ctb],
Zhang Hanzhe [ctb],
Boyas Matt [ctb],
Freedman David [ctb],

Achilleos Andreas [ctb],
 Butts Jessica [ctb],
 Nguyen Sheila [ctb],
 Soleymani Taraneh [ctb]

Maintainer De los Santos Hannah <hdelossantos@mitre.org>

Repository CRAN

Date/Publication 2022-09-01 15:00:02 UTC

R topics documented:

acf_answers	3
adjustcarryforward	4
bmianthro	6
CDCref_d	7
cleangrowth	7
ewma	10
ext_bmiz	12
growth_cdc_ext	14
lenanthro	14
longwide	14
nhanes-reference-medians	16
read_anthro	16
recode_sex	17
sd_median	18
simple_bmi	19
splitinput	20
syngrowth	21
tanner_ht_vel	21
tanner_ht_vel_with_2sd	22
testacf	22
test_syngrowth_sas_output_compare	23
test_syngrowth_wide	24
weianthro	24
who_ht_maxvel	24
who_ht_maxvel_2sd	25
who_ht_vel_2sd	25
who_ht_vel_3sd	25
Index	26

acf_answers

*Answers for adjustcarryforward***Description**

Determines what should absolutely be reincluded or definitely excluded for a given dataset, already run through cleangrowth.

Usage

```
acf_answers(
  subjid,
  param,
  agedays,
  sex,
  measurement,
  orig.exclude,
  sd.recenter = NA,
  ewma.exp = -1.5,
  ref.data.path = "",
  quietly = TRUE
)
```

Arguments

subjid	Vector of unique identifiers for each subject in the database.
param	Vector identifying each measurement, may be 'WEIGHTKG', 'HEIGHTCM', or 'LENGTHCM' vs. 'LENGTHCM' only affects z-score calculations between ages 24 to 35 months (730 to 1095 days). All linear measurements below 731 days of life (age 0-23 months) are interpreted as supine length, and all linear measurements above 1095 days of life (age 36+ months) are interpreted as standing height. Note: at the moment, all LENGTHCM will be converted to HEIGHTCM. In the future, the algorithm will be updated to consider this difference.
agedays	Numeric vector containing the age in days at each measurement.
sex	Vector identifying the gender of the subject, may be 'M', 'm', or 0 for males, vs. 'F', 'f' or 1 for females.
measurement	Numeric vector containing the actual measurement data. Weight must be in kilograms (kg), and linear measurements (height vs. length) in centimeters (cm).
orig.exclude	Vector of exclusion assessment results from cleangrowth()
sd.recenter	Data frame or table with median SD-scores per day of life
ewma.exp	Exponent to use for weighting measurements in the exponentially weighted moving average calculations. Defaults to -1.5. This exponent should be negative in order to weight growth measurements closer to the measurement being

	evaluated more strongly. Exponents that are further from zero (e.g. -3) will increase the relative influence of measurements close in time to the measurement being evaluated compared to using the default exponent.
ref.data.path	Path to reference data. If not supplied, the year 2000 Centers for Disease Control (CDC) reference data will be used.
quietly	Determines if function messages are to be displayed and if log files (parallel only) are to be generated. Defaults to TRUE.

Value

A data frame, containing an index "n" of rows, corresponding to the original order of the input vectors, and "acf_answers", containing the answers on whether a height value should be kept or excluded (returns "Definitely Exclude", "Definitely Include", or "Unknown" for height values, NA for weight values).

adjustcarryforward	<i>adjustcarryforward</i> adjustcarryforward Uses absolute height velocity to identify values excluded as carried forward values for reinclusion.
--------------------	---

Description

adjustcarryforward adjustcarryforward Uses absolute height velocity to identify values excluded as carried forward values for reinclusion.

Usage

```
adjustcarryforward(
  subjid,
  param,
  agedays,
  sex,
  measurement,
  orig.exclude,
  exclude_opt = 0,
  sd.recenter = NA,
  ewma.exp = -1.5,
  ref.data.path = "",
  quietly = TRUE,
  minfactor = 0.5,
  maxfactor = 2,
  banddiff = 3,
  banddiff_plus = 5.5,
  min_ht.exp_under = 2,
  min_ht.exp_over = 0,
  max_ht.exp_under = 0.33,
  max_ht.exp_over = 1.5
)
```

Arguments

subjid	Vector of unique identifiers for each subject in the database.
param	Vector identifying each measurement, may be 'WEIGHTKG', 'HEIGHTCM', or 'LENGTHCM' 'HEIGHTCM' vs. 'LENGTHCM' only affects z-score calculations between ages 24 to 35 months (730 to 1095 days). All linear measurements below 731 days of life (age 0-23 months) are interpreted as supine length, and all linear measurements above 1095 days of life (age 36+ months) are interpreted as standing height. Note: at the moment, all LENGTHCM will be converted to HEIGHTCM. In the future, the algorithm will be updated to consider this difference.
agedays	Numeric vector containing the age in days at each measurement.
sex	Vector identifying the gender of the subject, may be 'M', 'm', or 0 for males, vs. 'F', 'f' or 1 for females.
measurement	Numeric vector containing the actual measurement data. Weight must be in kilograms (kg), and linear measurements (height vs. length) in centimeters (cm).
orig.exclude	Vector of exclusion assessment results from cleangrowth()
exclude_opt	Number from 0 to 3 indicating which option to use to handle strings of carried-forwards: 0. no change. <ol style="list-style-type: none"> 1. when deciding to exclude values, if we have a string of carried forwards, drop the most deviant value, and all CFs in the same string, and move on as normal. 2. when deciding to exclude values, if the most deviant in a string of carried forwards is flagged, check all the CFs in that string from 1:N. Exclude all after the first that is flagged for exclusion when comparing to the Include before and after. Do not remove things designated as include. 3. when deciding to exclude values, if the most deviant in a string of carried forwards is flagged, check all the CFs in that string from 1:N. Exclude all after the first that is flagged for exclusion when comparing to the Include before and after. Make sure remove things designated as include.
sd.recenter	Data frame or table with median SD-scores per day of life
ewma.exp	Exponent to use for weighting measurements in the exponentially weighted moving average calculations. Defaults to -1.5. This exponent should be negative in order to weight growth measurements closer to the measurement being evaluated more strongly. Exponents that are further from zero (e.g. -3) will increase the relative influence of measurements close in time to the measurement being evaluated compared to using the default exponent.
ref.data.path	Path to reference data. If not supplied, the year 2000 Centers for Disease Control (CDC) reference data will be used.
quietly	Determines if function messages are to be displayed and if log files (parallel only) are to be generated. Defaults to TRUE.
minfactor	Sweep variable for computing mindiff.next.ht in 15f, default 0.5
maxfactor	Sweep variable for computing maxdiff.next.ht in 15f, default 2
banddiff	Sweep variable for computing mindiff.next.ht in 15f, default 3

banddiff_plus Sweep variable for computing maxdiff.next.ht in 15, default 5.5
 min_ht.exp_under Sweep variable for computing ht.exp in 15f, default 2
 min_ht.exp_over Sweep variable for computing ht.exp in 15f, default 0
 max_ht.exp_under Sweep variable for computing ht.exp in 15f, default 0.33
 max_ht.exp_over Sweep variable for computing ht.exp in 15f, default 1.5

Value

Re-evaluated exclusion assessments based on height velocity.

Examples

```
# Run on a small subset of given data
df <- as.data.frame(syngrowth)
df <- df[df$subjid %in% unique(df[, "subjid"])[1:5], ]
clean_df <- cbind(df,
  "gcr_result" = cleangrowth(df$subjid,
    df$param,
    df$agedays,
    df$sex,
    df$measurement))

# Adjust carry forward values in cleaned data
adj_clean <- adjustcarryforward(subjid = clean_df$subjid,
  param = clean_df$param,
  agedays = clean_df$agedays,
  sex = clean_df$sex,
  measurement = clean_df$measurement,
  orig.exclude = clean_df$gcr_result)
```

 bmianthro

BMI Anthro

Description

Part of default CDC-derived tables

Details

Contains BMI data for calculating BMI

bmianthro.txt.gz

Used in function cleangrowth()

CDCref_d	<i>CDC BMI reference data</i>
----------	-------------------------------

Description

Used for extended BMIz computation

CDCref_d.csv.gz

Used for extended BMI computation

cleangrowth	<i>Clean growth measurements</i>
-------------	----------------------------------

Description

Clean growth measurements

Usage

```
cleangrowth(  
  subjid,  
  param,  
  agedays,  
  sex,  
  measurement,  
  recover.unit.error = FALSE,  
  sd.extreme = 25,  
  z.extreme = 25,  
  lt3.exclude.mode = "default",  
  height.tolerance.cm = 2.5,  
  error.load.mincount = 2,  
  error.load.threshold = 0.5,  
  sd.recenter = NA,  
  sdmedian.filename = "",  
  sdrecentered.filename = "",  
  include.carryforward = FALSE,  
  ewma.exp = -1.5,  
  ref.data.path = "",  
  log.path = NA,  
  parallel = FALSE,  
  num.batches = NA,  
  quietly = TRUE,  
  adult_cutpoint = 20,  
  weight_cap = Inf,  
  adult_columns_filename = ""  
)
```

Arguments

<code>subjid</code>	Vector of unique identifiers for each subject in the database.
<code>param</code>	Vector identifying each measurement, may be 'WEIGHTKG', 'WEIGHTLBS', 'HEIGHTCM', 'HEIGHTIN', or 'LENGTHCM' 'HEIGHTCM'/'HEIGHTIN' vs. 'LENGTHCM' only affects z-score calculations between ages 24 to 35 months (730 to 1095 days). All linear measurements below 731 days of life (age 0-23 months) are interpreted as supine length, and all linear measurements above 1095 days of life (age 36+ months) are interpreted as standing height. Note: at the moment, all LENGTHCM will be converted to HEIGHTCM. In the future, the algorithm will be updated to consider this difference. Additionally, imperial 'HEIGHTIN' and 'WEIGHTLBS' measurements are converted to metric during algorithm calculations.
<code>agedays</code>	Numeric vector containing the age in days at each measurement.
<code>sex</code>	Vector identifying the gender of the subject, may be 'M', 'm', or 0 for males, vs. 'F', 'f' or 1 for females.
<code>measurement</code>	Numeric vector containing the actual measurement data. Weight must be in kilograms (kg), and linear measurements (height vs. length) in centimeters (cm).
<code>recover.unit.error</code>	Indicates whether the cleaning algorithm should attempt to identify unit errors (I.e. inches vs. cm, lbs vs. kg). If unit errors are identified, the value will be corrected and retained within the cleaning algorithm as a valid measurement. Defaults to FALSE.
<code>sd.extreme</code>	Measurements more than <code>sd.extreme</code> standard deviations from the mean (either above or below) will be flagged as invalid. Defaults to 25.
<code>z.extreme</code>	Measurements with an absolute z-score greater than <code>z.extreme</code> will be flagged as invalid. Defaults to 25.
<code>lt3.exclude.mode</code>	Determines type of exclusion procedure to use for 1 or 2 measurements of one type without matching same <code>ageday</code> measurements for the other parameter. Options include "default" (standard <code>growthcleanr</code> approach), and "flag.both" (in case of two measurements of one type without matching values for the other parameter, flag both for exclusion if beyond threshold)
<code>height.tolerance.cm</code>	maximum decrease in height tolerated for sequential measurements
<code>error.load.mincount</code>	minimum count of exclusions on parameter before considering excluding all measurements. Defaults to 2.
<code>error.load.threshold</code>	threshold of percentage of excluded measurement count to included measurement count that must be exceeded before excluding all measurements of either parameter. Defaults to 0.5.
<code>sd.recenter</code>	specifies how to recenter medians. May be a data frame or table w/median SD-scores per day of life by gender and parameter, or "NHANES" or "derive" as a character vector. <ul style="list-style-type: none"> • If <code>sd.recenter</code> is specified as a data set, use the data set

- If `sd.recenter` is specified as "nhanes", use NHANES reference medians
- If `sd.recenter` is specified as "derive", derive from input
- If `sd.recenter` is not specified or NA:
 - If the input set has at least 5,000 observations, derive medians from input
 - If the input set has fewer than 5,000 observations, use NHANES

If specifying a data set, columns must include `param`, `sex`, `agedays`, and `sd.median` (referred to elsewhere as "modified Z-score"), and those medians will be used for recentering. A summary of how the NHANES reference medians were derived is available in `README.md`. Defaults to NA.

<code>sdmedian.filename</code>	Name of file to save <code>sd.median</code> data calculated on the input dataset to as CSV. Defaults to "", for which this data will not be saved. Use for extracting medians for parallel processing scenarios other than the built-in parallel option.
<code>sdrecentered.filename</code>	Name of file to save re-centered data to as CSV. Defaults to "", for which this data will not be saved. Useful for post-processing and debugging.
<code>include.carryforward</code>	Determines whether Carry-Forward values are kept in the output. Defaults to False.
<code>ewma.exp</code>	Exponent to use for weighting measurements in the exponentially weighted moving average calculations. Defaults to -1.5. This exponent should be negative in order to weight growth measurements closer to the measurement being evaluated more strongly. Exponents that are further from zero (e.g. -3) will increase the relative influence of measurements close in time to the measurement being evaluated compared to using the default exponent.
<code>ref.data.path</code>	Path to reference data. If not supplied, the year 2000 Centers for Disease Control (CDC) reference data will be used.
<code>log.path</code>	Path to log file output when running in parallel (non-quiet mode). Default is NA. A new directory will be created if necessary. Set to NA to disable log files.
<code>parallel</code>	Determines if function runs in parallel. Defaults to FALSE.
<code>num.batches</code>	Specify the number of batches to run in parallel. Only applies if <code>parallel</code> is set to TRUE. Defaults to the number of workers returned by the <code>getDoParWorkers</code> function in the <code>foreach</code> package.
<code>quietly</code>	Determines if function messages are to be displayed and if log files (parallel only) are to be generated. Defaults to TRUE
<code>adult_cutpoint</code>	Number between 18 and 20, describing ages when the pediatric algorithm should not be applied ($< \text{adult_cutpoint}$), and the adult algorithm should apply ($\geq \text{adult_cutpoint}$). Numbers outside this range will be changed to the closest number within the range. Defaults to 20.
<code>weight_cap</code>	Positive number, describing a weight cap in kg (rounded to the nearest .1, +/- .1) within the adult dataset. If there is no weight cap, set to Inf. Defaults to Inf.
<code>adult_columns_filename</code>	Name of file to save original adult data, with additional output columns to as CSV. Defaults to "", for which this data will not be saved. Useful for post-analysis. For more information on this output, please see <code>README</code> .

Value

Vector of exclusion codes for each of the input measurements.

Possible values for each code are:

- 'Include', 'Unit-Error-High', 'Unit-Error-Low', 'Swapped-Measurements', 'Missing',
- 'Exclude-Carried-Forward', 'Exclude-SD-Cutoff', 'Exclude-EWMA-Extreme', 'Exclude-EWMA-Extreme-Pair',
- 'Exclude-Extraneous-Same-Day',
- 'Exclude-EWMA-8', 'Exclude-EWMA-9', 'Exclude-EWMA-10', 'Exclude-EWMA-11', 'Exclude-EWMA-12', 'Exclude-EWMA-13', 'Exclude-EWMA-14',
- 'Exclude-Min-Height-Change', 'Exclude-Max-Height-Change',
- 'Exclude-Pair-Delta-17', 'Exclude-Pair-Delta-18', 'Exclude-Pair-Delta-19',
- 'Exclude-Single-Outlier', 'Exclude-Too-Many-Errors', 'Exclude-Too-Many-Errors-Other-Parameter'

Examples

```
# Run calculation using a small subset of given data
df_stats <- as.data.frame(syngrowth)
df_stats <- df_stats[df_stats$subjid %in% unique(df_stats[, "subjid"])[1:5], ]

clean_stats <- cleangrowth(subjid = df_stats$subjid,
                          param = df_stats$param,
                          agedays = df_stats$agedays,
                          sex = df_stats$sex,
                          measurement = df_stats$measurement)

# Once processed you can filter data based on result value
df_stats <- cbind(df_stats, "clean_result" = clean_stats)
clean_df_stats <- df_stats[df_stats$clean_result == "Include",]

# Parallel processing: run using 2 cores and batches
clean_stats <- cleangrowth(subjid = df_stats$subjid,
                          param = df_stats$param,
                          agedays = df_stats$agedays,
                          sex = df_stats$sex,
                          measurement = df_stats$measurement,
                          parallel = TRUE,
                          num.batches = 2)
```

ewma

Exponentially Weighted Moving Average (EWMA)

Description

ewma calculates the exponentially weighted moving average (EWMA) for a set of numeric observations over time.

Usage

```
ewma(agedays, z, ewma.exp, ewma.adjacent = TRUE)
```

Arguments

agedays	Vector of age in days for each z score (potentially transformed to adjust weighting).
z	Input vector of numeric z-score data.
ewma.exp	Exponent to use for weighting.
ewma.adjacent	Specify whether EWMA values excluding adjacent measurements should be calculated. Defaults to TRUE.

Value

Data frame with 3 variables:

- The first variable (ewma.all) contains the EWMA at observation time excluding only the actual observation for that time point.
- The second variable (ewma.before) contains the EWMA for each observation excluding both the actual observation and the immediate prior observation.
- The third variable (ewma.after) contains the EWMA for each observation excluding both the actual observation and the subsequent observation.

Examples

```
# Run on 1 subject, 1 type of parameter
df_stats <- as.data.frame(syngrowth)
df_stats <- df_stats[df_stats$subjid == df_stats$subjid[1] &
                    df_stats$param == "HEIGHTCM", ]

# Get the uncentered z-scores
measurement_to_z <- read_anthro(cdc.only = TRUE)
sd <- measurement_to_z(df_stats$param,
                      df_stats$agedays,
                      df_stats$sex,
                      df_stats$measurement,
                      TRUE)

# Calculate exponentially weighted moving average
e_df <- ewma(df_stats$agedays, sd, ewma.exp = -1.5)
```

 ext_bmiZ

Calculate extended BMI measures

Description

ext_bmiZ Calculates the sigma (scale parameter for the half-normal distribution), extended BMI percentile, extended BMIz, and the CDC LMS Z-scores for weight, height, and BMI for children between 2 and 19.9 years of age. Note that for BMIs \leq 95th percentile of the CDC growth charts, the extended values for BMI are equal to the LMS values. The extended values differ only for children who have a BMI $>$ 95th percentile.

Usage

```
ext_bmiZ(
  data,
  age = "agem",
  wt = "wt",
  ht = "ht",
  bmi = "bmi",
  adjust.integer.age = TRUE,
  ref.data.path = ""
)
```

Arguments

data	Input data frame or data table
age	Name of input column containing subject age in months in quotes, default "agem"
wt	Name of input column containing weight (kg) value in quotes, default "wt"
ht	Name of input column containing height (cm) value in quotes, default "ht"
bmi	Name of input column containing calculated BMI in quotes, default "bmi"
adjust.integer.age	If age inputs are all integer, add 0.5 if TRUE; default TRUE
ref.data.path	Path to directory containing reference data

Details

This function should produce output equivalent to the SAS macro provided at <https://www.cdc.gov/nccdphp/dnpao/growthcha>

The extended BMIz is the inverse cumulative distribution function (CDF) of the extended BMI percentile. If the extended percentile is very close to 100, the qnorm function in R produces an infinite value. This occurs only if the the extended BMI percentile is $>$ 99.99999999999999. This occurs infrequently, such as a 48-month-old with a BMI $>$ 39, and it is likely that these BMIs represent data entry errors. For these cases, extended BMIz is set to 8.21, a value that is slightly greater than the largest value that can be calculated.

See the README.md file for descriptions of the output columns generated by this function.

data must have columns for at least age, sex, weight, height, and bmi.

age should be coded in months, using the most precise values available. To convert to months from age in years, multiply by 12. To convert to months from age in days, divide by 30.4375 (365.25 / 12).

sex is coded as 1 (Male) or 2 (Female). Note that this is different from cleangrowth, which uses 0 (Male) and 1 (Female).

wt should be in kilograms.

ht should be in centimeters.

Specify the input data parameter names for age, wt, ht, bmi using quotation marks. See example below.

If the parameter `adjust.integer.age` is TRUE (the default), 0.5 will be added to all age if all input values are integers. Set to FALSE to disable.

By default, the reference data file `CDCref_d.csv`, made available at <https://www.cdc.gov/nccdphp/dnpao/growthcharts/resources/> is included in this package for convenience. If you are developing this package, use `ref.data.path` to adjust the path to this file from your working directory if necessary.

Value

Expanded data frame containing computed BMI values

Examples

```
# Run on a small subset of given data
df <- as.data.frame(syngrowth)
df <- df[df$subjid %in% unique(df[, "subjid"])[1:5], ]
df <- cbind(df,
            "gcr_result" = cleangrowth(df$subjid,
                                       df$param,
                                       df$agedays,
                                       df$sex,
                                       df$measurement))

df_wide <- longwide(df) # convert to wide format for ext_bmiz
df_wide_bmi <- simple_bmi(df_wide) # compute simple BMI

# Calling the function with default column names
df_bmiz <- ext_bmiz(df_wide_bmi)

# Specifying different column names; note that quotes are used
dfc <- simple_bmi(df_wide)
colnames(dfc)[colnames(dfc) %in% c("agem", "wt", "ht")] <-
  c("agemos", "weightkg", "heightcm")
df_bmiz <- ext_bmiz(dfc, age="agemos", wt="weightkg", ht="heightcm")

# Disabling conversion of all-integer age in months to (age + 0.5)
dfc <- simple_bmi(df_wide)
df_bmiz <- ext_bmiz(dfc, adjust.integer.age=FALSE)
```

growth_cdc_ext	<i>CDC Growth Percentile Table</i>
----------------	------------------------------------

Description

Part of default CDC-derived tables

Details

Contains percentiles for various ages, gender, and weights, pre-calculated by CDC

growthfile_cdc_ext.csv.gz

Used in function cleangrowth()

lenanthro	<i>Length to Age Table</i>
-----------	----------------------------

Description

Part of default CDC-derived tables

Details

Contains percentiles for various ages, gender, and weights, pre-calculated by CDC

lenanthro.txt.gz

Used in function cleangrowth()

longwide	<i>Transform data in growthcleanr format into wide structure for BMI calculation</i>
----------	--

Description

longwide transforms data from long to wide format. Ideal for transforming output from growthcleanr::cleangrowth() into a format suitable for growthcleanr::ext_bmiz().

Usage

```
longwide(
  long_df,
  id = "id",
  subjid = "subjid",
  sex = "sex",
  agedays = "agedays",
  param = "param",
  measurement = "measurement",
  gcr_result = "gcr_result",
  include_all = FALSE,
  inclusion_types = c("Include")
)
```

Arguments

long_df	A data frame to be transformed. Expects columns: id, subjid, sex, agedays, param, measurement, and gcr_result.
id	name of observation ID column
subjid	name of subject ID column
sex	name of sex descriptor column
agedays	name of age (in days) descriptor column
param	name of parameter column to identify each type of measurement
measurement	name of measurement column containing the actual measurement data
gcr_result	name of column of results from growthcleanr::cleangrowth()
include_all	Determines whether the function keeps all exclusion codes. If TRUE, all exclusion types are kept and the inclusion_types argument is ignored. Defaults to FALSE.
inclusion_types	Vector indicating which exclusion codes from the cleaning algorithm should be included in the data, given that include_all is FALSE. For all options, see growthcleanr::cleangrowth(). Defaults to c("Include").

Value

Returns a data frame transformed from long to wide. Includes only values flagged with indicated inclusion types. Note that, for each subject, heights without corresponding weights for a given age (and vice versa) will be dropped.

Examples

```
# Run on a small subset of given data
df <- as.data.frame(syngrowth)
df <- df[df$subjid %in% unique(df[, "subjid"])[1:5], ]
df <- cbind(df,
            "gcr_result" = cleangrowth(df$subjid,
                                      df$param,
```

```

df$agedays,
df$sex,
df$measurement))

# Convert to wide format
wide_df <- longwide(df)

# Include all inclusion types
wide_df <- longwide(df, include_all = TRUE)

# Specify all inclusion codes
wide_df <- longwide(df, inclusion_types = c("Include", "Exclude-Carried-Forward"))

```

nhanes-reference-medians

NHANES reference medians

Description

Contains reference median values for default recentering, derived from NHANES years 2009-2018

nhanes-reference-medians.csv.gz

Used in function cleangrowth()

read_anthro

Function to calculate z-scores and csd-scores based on anthro tables.

Description

Function to calculate z-scores and csd-scores based on anthro tables.

Usage

```
read_anthro(path = "", cdc.only = FALSE)
```

Arguments

`path` Path to supplied reference anthro data. Defaults to package anthro tables.
`cdc.only` Whether or not only CDC data should be used. Defaults to false.

Value

Function for calculating BMI based on measurement, age in days, sex, and measurement value.

Examples

```
# Return calculating function with all defaults
afunc <- read_anthro()

# Return calculating function while specifying a path and using only CDC data
afunc <- read_anthro(path = system.file("extdata", package = "growthcleanr"),
                     cdc.only = TRUE)
```

recode_sex	<i>Recode binary sex variable for compatibility</i>
------------	---

Description

recode_sex recodes a binary sex variable for a given source column in a data frame or data table. Useful in transforming output from growthcleanr::cleangrowth() into a format suitable for growthcleanr::ext_bmiz().

Usage

```
recode_sex(
  input_data,
  sourcecol = "sex",
  sourcem = "0",
  sourcef = "1",
  targetcol = "sex_recoded",
  targetm = 1L,
  targetf = 2L
)
```

Arguments

input_data	a data frame or data table to be transformed. Expects a source column containing a binary sex variable.
sourcecol	name of sex descriptor column. Defaults to "sex"
sourcem	variable indicating "male" sex in input data. Defaults to "0"
sourcef	variable indicating "female" sex in input data. Defaults to "1"
targetcol	desired name of recoded sex descriptor column. Defaults to "sex_recoded"
targetm	desired name of recoded sex variable indicating "male" sex in output data. Defaults to 1
targetf	desired name of recoded sex variable indicating "female" sex in output data. Defaults to 2

Value

Returns a data table with recoded sex variables.

Examples

```
# Run on given data
df <- as.data.frame(syngrowth)

# Run with all defaults
df_r <- recode_sex(df)

# Specify different targets
df_rt <- recode_sex(df, targetcol = "sexr", targetm = "Male", targetf = "Female")

# Specify different inputs
df_ri <- recode_sex(df_rt, sourcecol = "sexr", sourcem = "Male", sourcef = "Female")
```

sd_median	<i>Calculate median SD score by age for each parameter.</i>
-----------	---

Description

Calculate median SD score by age for each parameter.

Usage

```
sd_median(param, sex, agedays, sd.orig)
```

Arguments

param	Vector identifying each measurement, may be 'WEIGHTKG', or 'HEIGHTCM'.
sex	Vector identifying the gender of the subject, may be 'M', 'm', or 0 for males, vs. 'F', 'f' or 1 for females.
agedays	Numeric vector containing the age in days at each measurement.
sd.orig	Vector of previously calculated standard deviation (SD) scores for each measurement before re-centering.

Value

Table of data with median SD-scores per day of life by gender and parameter.

Examples

```
# Run on 1 subject
df_stats <- as.data.frame(syngrowth)
df_stats <- df_stats[df_stats$subjid == df_stats$subjid[1], ]

# Get the original standard deviations
measurement_to_z <- read_anthro(cdc.only = TRUE)
sd.orig <- measurement_to_z(df_stats$param,
                           df_stats$agedays,
                           df_stats$sex,
```

```

        df_stats$measurement,
        TRUE)

# Calculate median standard deviations
sd.m <- sd_median(df_stats$param,
                 df_stats$sex,
                 df_stats$agedays,
                 sd.orig)

```

simple_bmi

Compute BMI using standard formula

Description

simple_bmi Computes BMI using standard formula. Assumes input compatible with output from longwide().

Usage

```
simple_bmi(wide_df, wtcol = "wt", htcol = "ht")
```

Arguments

wide_df	A data frame or data table containing heights and weights in wide format, e.g., after transformation with longwide()
wtcol	name of observation height value column, default 'wt'
htcol	name of subject weight value column, default 'ht'

Value

Returns a data table with the added column "bmi"

Examples

```

# Simple usage
# Run on a small subset of given data
df <- as.data.frame(syngrowth)
df <- df[df$subjid %in% unique(df[, "subjid"])[1:5], ]
df <- cbind(df,
           "gcr_result" = cleangrowth(df$subjid,
                                     df$param,
                                     df$agedays,
                                     df$sex,
                                     df$measurement))

# Convert to wide format
wide_df <- longwide(df)
wide_df_with_bmi <- simple_bmi(wide_df)

# Specifying different column names; note that quotes are used

```

```
colnames(wide_df)[colnames(wide_df) %in% c("wt", "ht")] <-
  c("weight", "height")
wide_df_with_bmi <- simple_bmi(wide_df, wtcoll = "weight", htcol = "height")
```

splitinput

Split input data into multiple files

Description

splitinput Splits input based on keeppcol specified, yielding csv files each with at least the minimum number of rows that are written and saved separately (except for the last split file written, which may be smaller). Allows splitting input data while ensuring all records for each individual subject will stay together in one file. Pads split filenames with zeros out to five digits for consistency, assuming < 100,000 file count result.

Usage

```
splitinput(
  df,
  fname = deparse(substitute(df)),
  fdir = NA,
  min_nrow = 10000,
  keeppcol = "subjid"
)
```

Arguments

df	data frame to split
fname	new name for each of the split files to start with
fdir	directory to put each of the split files (use "." for working directory). Must be changed from default (NA), which will trigger error.
min_nrow	minimum number of rows for each split file (default 10000)
keeppcol	the column name (default "subjid") to use to keep records with the same values together in the same single split file

Value

the count number referring to the last split file written

Examples

```
# Run on given data
df <- as.data.frame(syngrowth)

# Run with all defaults (specifying directory)
splitinput(df, fdir = tempdir())
```

```
# Specifying the name, directory and minimum row size
splitinput(df, fname = "syngrowth", fdir = tempdir(), min_nrow = 5000)

# Specifying a different subject ID column
colnames(df)[colnames(df) == "subjid"] <- "sub_id"
splitinput(df, fdir = tempdir(), keepcol = "sub_id")
```

syngrowth

syngrowth

Description

A synthetic set of measurements from ~3,500 subjects generated using Synthea, with measurement errors for testing with growthcleanr. Contains both pediatric and adult data.

Usage

```
syngrowth
```

Format

A data frame with six variables: id, subjid, sex, agedays, param, and measurement

Details

Example electronic health record (heightcm, weightkg) data.

tanner_ht_vel

Tanner Growth Velocity Table

Description

Part of default CDC-derived tables

Details

Contains velocities for growth pre-calculated by CDC

tanner_ht_vel.csv.gz

Used in function cleangrowth()

 tanner_ht_vel_with_2sd

Tanner Growth Velocity Table with (2 σ)

Description

Part of default CDC-derived tables

Details

Contains velocities for growth pre-calculated by CDC, including those 2 standard deviations away.

tanner_ht_vel_with_2sd.csv.gz

Used in function acf_answers()

 testacf

Function to test adjust carried forward

Description

The goal of this script is to consider the height values that growthcleanr excludes as “carried forward” for potential re-inclusion by using a reverse absolute height velocity check based on step 15 of the Daymont et al. algorithm

Usage

```
testacf(
  infile,
  seed = 7,
  searchtype = "random",
  grid.length = 9,
  writeout = FALSE,
  outfile = paste0("test_adjustcarryforward_", format(Sys.time(),
    "%m-%d-%Y_%H-%M-%S")),
  quietly = FALSE,
  param = "none",
  debug = FALSE,
  maxrecs = 0,
  exclude_opt = 0,
  add_answers = TRUE
)
```

Arguments

infile	Input data frame/data table, cleaned by cleangrowth(), with columns as described in main README.md
seed	Numeric random seed, used only when performing random search
searchtype	Type of search to perform: random (default), line-grid, full-grid
grid.length	Number of steps in grid to search
writeout	Write output to file? Default FALSE.
outfile	"Output file name, default 'test_adjustcarrforward_DATE_TIME', where DATE is the current system date and time"
quietly	Verbose progress info
param	"none", or data frame to specify which parameters to run full search on, and values to use if not, used only when performing full-grid search
debug	Produce extra data files for debugging
maxrecs	Limit to specified # subjects, default 0 (no limit)
exclude_opt	Type of exclusion method for carried forward strings, 0 to 3. See adjustcarry-forward documentation for more information
add_answers	TRUE or FALSE, indicating whether or not to add answers (definely include/exclude) for the given dataset. Defaults to TRUE

Value

A list containing: testacf_res: data frame with adjustcarryforward results for each run, params: a data frame containing parameter values for each run. debug_filtered_data: data frame with original data, returned if debug TRUE

test_syngrowth_sas_output_compare
CDC SAS BMI Output

Description

Contains results of CDC SAS macro for calculating BMI values.

test_syngrowth_sas_output_compare.csv.gz

Used to test function ext_bmiz()

test_syngrowth_wide	<i>CDC SAS BMI Input</i>
---------------------	--------------------------

Description

Contains input data for CDC SAS macro for calculating BMI values.

test_syngrowth_wide.csv.gz

Used to test function ext_bmi3()

weianthro	<i>Weight Anthro Table</i>
-----------	----------------------------

Description

Part of default CDC-derived tables

Details

Contains median and standard deviation for weight by age and gender

weianthro.csv.gz

Used in function cleangrowth()

who_ht_maxvel	<i>WHO Maximum Height Velocity for (3σ)</i>
---------------	---

Description

Part of default WHO-derived tables

Details

Contains three standard deviations for the World Health Organization values of maximum height velocities.

who_ht_maxvel_3sd.csv.gz

Used in function cleangrowth()

who_ht_maxvel_2sd	<i>WHO Maximum Height Velocity for (2σ)</i>
-------------------	---

Description

Part of default WHO-derived tables

Details

Contains two standard deviations for the World Health Organization values of maximum height velocities.

who_ht_maxvel_2sd.csv.gz

Used in function acf_answers()

who_ht_vel_2sd	<i>WHO Height Velocity for (2σ)</i>
----------------	---

Description

Part of default WHO-derived tables

Details

Contains two standard deviations for the World Health Organization values of height velocities.

who_ht_vel_2sd.csv.gz

Used in function acf_answers()

who_ht_vel_3sd	<i>WHO Height Velocity for (3σ)</i>
----------------	---

Description

Part of default WHO-derived tables

Details

Contains three standard deviations for the World Health Organization values of height velocities.

who_ht_vel_3sd.csv.gz

Used in function cleangrowth()

Index

* datasets

- syngrowth, [21](#)

- acf_answers, [3](#)
- adjustcarryforward, [4](#)

- bmianthro, [6](#)

- CDCref_d, [7](#)
- cleangrowth, [7](#)

- ewma, [10](#)
- ext_bmiz, [12](#)

- growth_cdc_ext, [14](#)

- lenanthro, [14](#)
- longwide, [14](#)

- nhanes-reference-medians, [16](#)

- read_anthro, [16](#)
- recode_sex, [17](#)

- sd_median, [18](#)
- simple_bmi, [19](#)
- splitinput, [20](#)
- syngrowth, [21](#)

- tanner_ht_vel, [21](#)
- tanner_ht_vel_with_2sd, [22](#)
- test_syngrowth_sas_output_compare, [23](#)
- test_syngrowth_wide, [24](#)
- testacf, [22](#)

- weianthro, [24](#)
- who_ht_maxvel, [24](#)
- who_ht_maxvel_2sd, [25](#)
- who_ht_vel_2sd, [25](#)
- who_ht_vel_3sd, [25](#)