# Package: gigs (via r-universe)

December 19, 2024

**Type** Package

**Title** Assess Fetal, Newborn, and Child Growth with International Standards

**Version** 0.5.1.9000

**Description** Convert between anthropometric measures and z-scores/centiles in multiple growth standards, and classify fetal, newborn, and child growth accordingly. With a simple interface to growth standards from the World Health Organisation and International Fetal and Newborn Growth Consortium for the 21st Century, gigs makes growth assessment easy and reproducible for clinicians, researchers and policy-makers.

**License** GPL (>= 3)

**URL** https://github.com/ropensci/gigs/, https://docs.ropensci.org/gigs

**BugReports** https://github.com/ropensci/gigs/issues

**Depends** R (>= 4.1.0)

**Imports** checkmate, gamlss.dist, stats, rlang, vctrs (>= 0.4.0), cli

**Suggests** knitr, rmarkdown, testthat (>= 3.0.0), dplyr, units, withr, ggsci, bench, cowplot, viridis, mfp, tidyr (>= 1.0.0)

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**Config/testthat/parallel** true

**Config/testthat/start-first** who_gs, ig_nbs, ig_fet, ig_png

**Encoding** UTF-8

**Language** en-GB

**LazyData** true

**Roxygen** list(markdown = TRUE, roclets = c(``namespace'', ``rd'', ``srr::srr_stats_roclet''))

**RoxygenNote** 7.3.2

**Repository** https://ropensci.r-universe.dev

**RemoteUrl** https://github.com/ropensci/gigs

**RemoteRef** master

**RemoteSha** 484cb3e2fdd465a8f68ef963f7084e65809efcf4

# Contents

| .gigs_options | *Package-level gigs options* |
|---|---|

### Description

An environment containing seven named character vectors. The first six define how gigs handles input data with missing, undefined, or invalid elements. The last ("handle_unused_levels")

- "handle_missing_data" - How should gigs handle missing (NA) elements?
- "handle_undefined_data" - How should gigs handle undefined (NaN, Inf, -Inf) elements?
- "handle_oob_xvar" - How should gigs handle elements of x variables which are out of bounds for the standard in use?
- "handle_invalid_sex" - How should gigs handle elements of sex which are not one of "M" or "F"?
- "handle_oob_centiles" - In centile2value functions, how should gigs handle elements of p that are not between 0 and 1?
- "handle_unused_levels" - When generating growth categories as factors, should gigs drop or keep unused factor levels, and should it issue warnings when unused factor levels occur?

Each of the input-checking options can take one of three values:

- "quiet" - Invalid elements are set to NA, silently.
- "warn" - Invalid elements are set to NA, with warnings issued when this is done.
- "error" - Invalid elements will cause informative errors.

For "handle_unused_levels", there are four valid choices:

- "keep_quiet" - Keep unused factor levels, and don't issue a warning.
- "keep_warn" - Keep unused factor levels, whilst issuing a warning.
- "drop_quiet" - Drop unused factor levels, and don't issue a warning.
- "drop_warn" - Drop unused factor levels, whilst issuing a warning.

You can use gigs_option_set() or gigs_option_set() to change these options on a one-by-one basis. To change all the input-checking options to a specific value, use gigs_input_options_set().

### Value

A named environment, where each name maps onto a specific **gigs** option.

### See Also

The gigs_option_get(), gigs_option_set() and gigs_input_options_set() functions, which can be used to get and set values in .gigs_options.

### Examples

```
# Get the names of all available options
names(.gigs_options)
```

| categorise_headsize | *Categorise head circumference-for-age z-scores into head circumference-for-age strata* |
|---|---|

## Description

Categorise head circumference-for-age z-scores into head circumference-for-age strata

## Usage

```
categorise_headsize(hcaz)
```

## Arguments

hcaz            A numeric vector of length one or more containing head circumference-for-age z-scores (HCAZs).

## Details

Cut-offs for head size categories are:

| Category | Factor level | Z-score bounds |
|---|---|---|
| Severe microcephaly | "microcephaly_severe" | hcaz =< -3 |
| Microcephaly | "microcephaly" | -3 < hcaz =< -2 |
| Normal head circumference | "normal_headcirc" | abs(hcaz) < 2 |
| Macrocephaly | "macrocephaly" | hcaz >= 2 |
| Severe macrocephaly | "macrocephaly_severe" | hcaz >= 3 |

## Value

An object of class factor with the same length as hcaz, containing head circumference-for-age classifications. Its levels are c("microcephaly_severe", "microcephaly", "normal_headcirc", "macrocephaly", "macrocephaly_severe"). By default, gigs will inform you this object contains unused factor levels. You can change this behaviour using the GIGS package-level option .gigs_options$handle_unused_levels.

## References

Victora CG, Schuler-Faccini L, Matijasevich A, Ribeiro E, Pessoa A, Barros FC. **Microcephaly in Brazil: how to interpret reported numbers?** *The Lancet* 2016, *387(10019):621-624* doi:10.1016/S01406736(16)002737

Accogli A, Geraldo AF, Piccolo G, Riva A, Scala M, Balagura G, et al. **Diagnostic Approach to Macrocephaly in Children**. *Frontiers in Paediatrics* 2022, *9:794069* doi:10.3389/fped.2021.794069

## Examples

```
hcaz <- c(-6.5, -3.5, -2.5, 0, 2.5, 3.5)
categorise_headsize(hcaz)
```

---

| categorise_sfga | *Categorise birthweight centiles into size-for-gestational age strata* |

---

### Description

Categorise birthweight centiles into size-for-gestational age strata

### Usage

```
categorise_sfga(p, severe = FALSE)
```

### Arguments

p
: A numeric vector with birth weight centiles from the INTERGROWTH-21st Newborn Size standard. If elements of p are not between 0 and 1, gigs will warn you. You can customise this behaviour with the GIGS package-level options.

severe
: A single logical value specifying whether to categorise SGA values below the third centile as "SGA(<3)". Default = FALSE.

### Details

Cut-offs for size-for-gestational age categorisations are:

| Category | Factor level | Centile bounds |
|----------|--------------|----------------|
| Severely small-for-GA | "SGA(<3)" | p < 0.03 |
| Small-for-GA | "SGA" | p < 0.1 |
| Appropriate-for-GA | "AGA" | 0.1 =< p =< 0.9 |
| Large-for-GA | "LGA" | p > 0.9 |

### Value

An object of class factor with the same length as p, containing size-for-GA classifications. If severe = FALSE, levels are c("SGA", "AGA", "LGA"). If severe = TRUE, levels are c("SGA(<3)", "SGA", "AGA", "LGA"). By default, gigs will inform you this object contains unused factor levels. You can change this behaviour using the GIGS package-level option .gigs_options$handle_unused_levels.

### Note

Input vectors are recycled by vctrs::vec_recycle_common(), and must adhere to the vctrs recycling rules.

### References

WHO. **Physical status: the use and interpretation of anthropometry. Report of a WHO Expert Committee.** *World Health Organisation Technical Report Series 1995,* **854: 1–452**

Royal College of Obstetricians and Gynaecologists. **The Investigation and Management of the Small-for-Gestational-Age Fetus: Green-top Guideline No. 31.** *Technical report, Royal College of Obstetricians and Gynaecologists, London, 2013.*

## See Also

[classify_sfga()](), which wraps this function for easy use in data.frame-based analytic pipelines.

## Examples

```
p <- c(0.01, 0.07, 0.25, 0.75, 0.93, 0.99)
categorise_sfga(p = p, severe = FALSE)
categorise_sfga(p = p, severe = TRUE)
```

---

categorise_stunting    *Categorise length/height-for-age z-scores into stunting strata*

---

## Description

Categorise length/height-for-age z-scores into stunting strata

## Usage

```
categorise_stunting(lhaz, outliers = FALSE)
```

## Arguments

lhaz        A numeric vector of length one or more containing length/height-for-age z-scores (LHAZs).

outliers    A single TRUE or FALSE value specifying whether implausible z-score thresholds should be applied. Default = FALSE.

## Details

Cut-offs for stunting categories are:

| Category | Factor level | Z-score bounds |
|---|---|---|
| Severe stunting | "stunting_severe" | lhaz =< -3 |
| Stunting | "stunting" | -3 < lhaz =< -2 |
| No stunting | "not_stunting" | lhaz > -2 |
| Outlier | "outlier" | abs(lhaz) > 6 |

## Value

An object of class factor with the same length as the longest input vector, containing stunting classifications. Its levels are c("stunting_severe", "stunting", "not_stunting") if outliers = FALSE (the default), else c("stunting_severe", "stunting", "not_stunting", "outlier"). By default, gigs will inform you this object contains unused factor levels. You can change this behaviour using the GIGS package-level option .gigs_options$handle_unused_levels.

## Note

This function assumes that your measurements were taken according to WHO guidelines, which stipulate that recumbent length should not be measured after 730 days. Instead, standing height should be used. Implausible z-score bounds are sourced from the referenced WHO report, and classification cut-offs from the DHS manual.

## References

**'Implausible z-score values'** *in* World Health Organization (ed.) *Recommendations for data collection, analysis and reporting on anthropometric indicators in children under 5 years old*. Geneva: World Health Organization and the United Nations Children's Fund UNICEF, (2019). pp. 64-65.

**'Percentage of children stunted, wasted, and underweight, and mean z-scores for stunting, wasting and underweight'** *in Guide to DHS Statistics DHS-7* Rockville, Maryland, USA: ICF (2020). pp. 431-435. `https://dhsprogram.com/data/Guide-to-DHS-Statistics/Nutritional_Status.htm`

## Examples

```
lhaz <- c(-6.5, -5, -3, 0, 3, 5, 6.5)
categorise_stunting(lhaz, outliers = FALSE)
categorise_stunting(lhaz, outliers = TRUE)
```

---

| categorise_svn | *Categorise birthweight centiles and gestational ages into small vulnerable newborn strata* |
|---|---|

---

## Description

Categorise birthweight centiles and gestational ages into small vulnerable newborn strata

## Usage

```
categorise_svn(p, gest_days)
```

## Arguments

| | |
|---|---|
| p | A numeric vector with birth weight centiles from the INTERGROWTH-21st Newborn Size standard. If elements of p are not between 0 and 1, gigs will warn you. You can customise this behaviour with the GIGS package-level options. |
| gest_days | A numeric vector of length one or more with gestational age(s) in days. Used to determine whether infants are term or preterm (a gestational age > 259 days means an infant is term). |

**Details**

Cut-offs for small, vulnerable newborn categorisations are:

| SVN category | Factor level | Centile bounds | Gestational age range |
|---|---|---|---|
| Preterm-SGA | "Preterm SGA" | p < 0.1 | gest_days < 259 |
| Preterm-AGA | "Preterm AGA" | 0.1 =< p =< 0.9 | gest_days < 259 |
| Preterm-LGA | "Preterm LGA" | p > 0.9 | gest_days < 259 |
| Term-SGA | "Term SGA" | p < 0.1 | gest_days >= 259 |
| Term-AGA | "Term AGA" | 0.1 =< p =< 0.9 | gest_days >= 259 |
| Term-LGA | "Term LGA" | p > 0.9 | gest_days >= 259 |

*Abbreviations:* SGA, small-for-gestational age; AGA, appropriate-for-gestational age; LGA, large-for-gestational age.

**Value**

An object of class factor with the same length as the longest input vector, containing small vulnerable newborn classifications. Its levels are c("Preterm SGA", "Preterm AGA", "Preterm LGA", "Term SGA", "Term AGA", "Term LGA"). By default, gigs will inform you this object contains unused factor levels. You can change this behaviour using the GIGS package-level option .gigs_options$handle_unused_lev

**Note**

Input vectors are recycled by vctrs::vec_recycle_common(), and must adhere to the vctrs recycling rules.

**References**

Lawn JE, Ohuma EO, Bradley E, Idueta LS, Hazel E, Okwaraji YB et al. **Small babies, big risks: global estimates of prevalence and mortality for vulnerable newborns to accelerate change and improve counting.** *The Lancet* 2023, *401(10389):1707-1719.* doi:10.1016/S0140-6736(23)005226

**See Also**

classify_svn(), which wraps this function for easy use in analytic pipelines. Also check out categorise_sfga(), which this function uses to stratify p into size-for-gestational age categories.

**Examples**

```
p <- c(0.01, 0.07, 0.25, 0.75, 0.93, 0.99)
gest_days <- c(250, 250, 250, 280, 280, 280) # 3 preterm, 3 term
categorise_svn(p = p, gest_days = gest_days)
```

categorise_wasting *Categorise weight-for-length/height z-scores into wasting strata*

## Description

Categorise weight-for-length/height z-scores into wasting strata

## Usage

```
categorise_wasting(wlz, outliers = FALSE)
```

## Arguments

wlz A numeric vector of length one or more containing weight-for-length/height z-scores (WLZs).

outliers A single TRUE or FALSE value specifying whether implausible z-score thresholds should be applied. Default = FALSE.

## Details

Cut-offs for wasting categories are:

| Category | Factor level | Z-score bounds |
|---|---|---|
| Severe wasting | "wasting_severe" | wlz =< -3 |
| Wasting | "wasting" | -3 < wlz =< -2 |
| No wasting | "not_wasting" | abs(wlz) < 2 |
| Overweight | "overweight" | wlz >= 2 |
| Outlier | "outlier" | abs(wlz) > 5 |

## Value

An object of class factor with the same length as wlz, containing wasting classifications. Its levels are c("wasting_severe", "wasting", "not_wasting", "overweight") if outliers = FALSE (the default), else c("wasting_severe", "wasting", "not_wasting", "overweight", "outlier"). By default, gigs will inform you this object contains unused factor levels. You can change this behaviour using the GIGS package-level option .gigs_options$handle_unused_levels.

## Note

This function assumes that your measurements were taken according to WHO guidelines, which stipulate that recumbent length should not be measured after 730 days. Instead, standing height should be used. Implausible z-score bounds are sourced from the referenced WHO report, and classification cut-offs from the DHS manual.

## References

**'Implausible z-score values'** *in* World Health Organization (ed.) *Recommendations for data collection, analysis and reporting on anthropometric indicators in children under 5 years old.* Geneva: World Health Organization and the United Nations Children's Fund UNICEF, (2019). pp. 64-65.

**'Percentage of children stunted, wasted, and underweight, and mean z-scores for stunting, wasting and underweight'** *in Guide to DHS Statistics DHS-7* Rockville, Maryland, USA: ICF (2020). pp. 431-435. https://dhsprogram.com/data/Guide-to-DHS-Statistics/Nutritional_Status.htm

## Examples

```
wlz <- c(-5.5, -3, 0, 3, 5.5)
categorise_wasting(wlz, outliers = FALSE)
categorise_wasting(wlz, outliers = TRUE)
```

---

| categorise_wfa | *Categorise weight-for-age z-scores into weight-for-age strata* |
|---|---|

---

## Description

Categorise weight-for-age z-scores into weight-for-age strata

## Usage

```
categorise_wfa(waz, outliers = FALSE)
```

## Arguments

waz
: A numeric vector of length one or more containing weight-for-age z-scores (WAZs).

outliers
: A single TRUE or FALSE value specifying whether implausible z-score thresholds should be applied. Default = FALSE.

## Details

Cut-offs for weight-for-age categories are:

| Category | Factor level | Z-score bounds |
|---|---|---|
| Severely underweight | "underweight_severe" | waz =< -3 |
| Underweight | "underweight" | -3 < waz =< -2 |
| Normal weight | "normal_weight" | abs(waz) < 2 |
| Overweight | "overweight" | waz >= 2 |
| Outlier | "outlier" | waz < -6 or waz > 5 |

**Value**

An object of class factor with the same length as waz, containing weight-for-age classifications. Its levels are c("underweight_severe", "underweight", "normal", "overweight") if outliers = FALSE (the default), else c("underweight_severe", "underweight", "normal", "overweight", "outlier"). By default, gigs will inform you this object contains unused factor levels. You can change this behaviour using the GIGS package-level option .gigs_options$handle_unused_levels.

**References**

'**Implausible z-score values**' *in* World Health Organization (ed.) *Recommendations for data collection, analysis and reporting on anthropometric indicators in children under 5 years old.* Geneva: World Health Organization and the United Nations Children's Fund UNICEF, (2019). pp. 64-65.

'**Percentage of children stunted, wasted, and underweight, and mean z-scores for stunting, wasting and underweight**' *in Guide to DHS Statistics DHS-7* Rockville, Maryland, USA: ICF (2020). pp. 431-435. `https://dhsprogram.com/data/Guide-to-DHS-Statistics/Nutritional_Status.htm`

**Examples**

```
waz <- c(-6.5, -3.5, -2.5, 0, 2.5, 3.5)
categorise_wfa(waz, outliers = FALSE)
categorise_wfa(waz, outliers = TRUE)
```

---

| classify_growth | *Classify multiple growth indicators at the same time using GIGS-recommended growth standards* |
|---|---|

---

**Description**

This function permits classification of multiple growth indicators (stunting, wasting, weight-for-age, and more) at once in data.frame-like objects.

**Usage**

```
classify_growth(
  .data,
  gest_days,
  age_days,
  sex,
  weight_kg = NULL,
  lenht_cm = NULL,
  headcirc_cm = NULL,
  id = NULL,
  .outcomes = c("sfga", "svn", "stunting", "wasting", "wfa", "headsize"),
  .new = list(sfga = c("birthweight_centile", "sfga", "sfga_severe"), svn =
    c("birthweight_centile", "svn"), stunting = c("lhaz", "stunting",
    "stunting_outliers"), wasting = c("wlz", "wasting", "wasting_outliers"), wfa =
```

```
        c("waz", "wfa", "wfa_outliers"), headsize = c("hcaz", "headsize")),
    .verbose = TRUE
)
```

## Arguments

| | |
|---|---|
| .data | A data.frame-like tabular object with one or more rows. Must be permissible by [checkmate::assert_data_frame()](), so you can also supply a tibble, data.table, or similar. |
| gest_days | [<data-masking>]() The name of a column in .data which is a numeric vector of gestational age(s) at birth in days. This column, in conjunction with the column referred to by age_days, is used to select which growth standard to use for each observation. |
| age_days | [<data-masking>]() The name of a column in .data which is a numeric vector of age values in days. |
| sex | [<data-masking>]() The name of a column in .data which is a case-sensitive character vector of sexes, either "M" (male) or "F" (female). By default, gigs will warn you if any elements of sex are not "M" or "F", or are missing (NA). You can customise this behaviour using the [GIGS package-level options](). |
| weight_kg | [<data-masking>]() The name of a column in .data which is a numeric vector of weight values in kg. When performing size-for-GA and small vulnerable newborn classifications, centiles and classifications will only be provided where measurements were taken <12hrs after birth, i.e. age_days < 0.5. |
| lenht_cm | [<data-masking>]() The name of a column in .data which is a numeric vector of length/height values in cm. |
| headcirc_cm | [<data-masking>]() The name of a column in .data which is a numeric vector of head circumference values in cm. |
| id | [<data-masking>]() The name of a column in .data which is a factor variable with IDs for each observation. When not NULL, this variable is used to ensure that only the first measurement taken from each infant is used as a birth measure. If all your data is from one individual, leave this parameter as NULL. Default = NULL. |
| .outcomes | A character vector of up to six elements in length describing which growth outcomes you want to assess. Use this when you supply data that can be used to generate multiple growth indicators, but only want to run a specific set. This argument is case-sensitive, and an error will be thrown if you supply a string in .analyses which is either NA or is not a member of this argument's default. Default = c("sfga", "svn", "stunting", "wasting", "wfa", "headsize"). |
| .new | A list with names corresponding to .analyses, which describes the names of new columns to be added to .data. If any elements of the vectors in .new are equal to existing elements of colnames(.data), the function will throw an error. Excluding the birthweight centiles produced when getting size-for-gestational age ("sfga") and small vulnerable newborn ("svn") classifications, all strings in .new must be unique. Each character vector in .new is repaired with vctrs::as_names(), which will issue messages if any elements in .new are changed. |

```
                         "sfga": c("birthweight_centile", "sfga", "sfga_severe")
                         "svn": c("birthweight_centile", "svn")
                         "stunting": c("lhaz", "stunting", "stunting_outliers")
                         "wasting": c("wlz", "wasting", "wasting_outliers")
                         "wfa": c("waz", "wfa", "wfa_outliers")
                         "headsize": c("hcaz", "headsize")
```

.verbose          A single logical value. When TRUE (the default), messages about which analyses
                  were requested versus which were performed will be printed to the console.
                  Warnings from classify_growth() will still be printed even if .verbose is
                  FALSE.

### Value

A tabular object of the same class that was provided as .data, with new columns named according
to .new. The new columns will depend on the values supplied for .outcomes.

### Note

For size-for-GA and small vulnerable newborn analyses, centiles and categorisations will only be
applied for birth measurements. These are considered to be the observation per level of id where
age_days is smallest, provided that age_days is <3.

Categorical (factor) columns produced here may contain unused factor levels. By default, gigs will
inform you if these columns have unused factor levels. You can change this behaviour using the
GIGS package-level option .gigs_options$handle_unused_levels.

### References

WHO. **Physical status: the use and interpretation of anthropometry. Report of a WHO Expert
Committee.** *World Health Organisation Technical Report Series 1995,* **854: 1–452**

Royal College of Obstetricians and Gynaecologists. **The Investigation and Management of the
Small-for-Gestational-Age Fetus: Green-top Guideline No. 31.** *Technical report, Royal College
of Obstetricians and Gynaecologists, London, 2013.*

Lawn JE, Ohuma EO, Bradley E, Idueta LS, Hazel E, Okwaraji YB et al. **Small babies, big
risks: global estimates of prevalence and mortality for vulnerable newborns to accelerate
change and improve counting.** *The Lancet* 2023, *401(10389):1707-1719.* doi:10.1016/S0140-
6736(23)005226

**'Implausible z-score values'** *in* World Health Organization (ed.) *Recommendations for data col-
lection, analysis and reporting on anthropometric indicators in children under 5 years old.* Geneva:
World Health Organization and the United Nations Children's Fund UNICEF, (2019). pp. 64-65.

**'Percentage of children stunted, wasted, and underweight, and mean z-scores for stunting,
wasting and underweight'** *in Guide to DHS Statistics DHS-7* Rockville, Maryland, USA: ICF
(2020). pp. 431-435. https://dhsprogram.com/data/Guide-to-DHS-Statistics/Nutritional_
Status.htm

Victora CG, Schuler-Faccini L, Matijasevich A, Ribeiro E, Pessoa A, Barros FC. **Microcephaly in
Brazil: how to interpret reported numbers?** *The Lancet* 2016, *387(10019):621-624* doi:10.1016/
S01406736(16)002737

Accogli A, Geraldo AF, Piccolo G, Riva A, Scala M, Balagura G, et al. **Diagnostic Approach to Macrocephaly in Children**. *Frontiers in Paediatrics* 2022, *9:794069* doi:10.3389/fped.2021.794069

**See Also**

See `gigs_waz()` to learn which growth standard will be used for different combinations of gestational/chronological age.

**Examples**

```
# This dummy dataset contains data from two people, from birth (<3 days) to
# 500 days of age.
data <- data.frame(
  child = factor(rep.int(c("A", "B"), c(3, 3))),
  agedays = c(0, 100, 500, 2, 100, 500),
  gestage  = c(rep(35 * 7, 3), rep(35 * 7, 3)),
  sex = rep.int(c("M", "F"), c(3, 3)),
  wt_kg = c(3, 6, 9, 3, 6, 9),
  len_cm = rep.int(c(52.2, 60.4, 75), 2),
  head_cm = rep.int(c(30, 40, 49), 2)
)

data_classified <- classify_growth(data,
                                   age_days = agedays,
                                   gest_days = gestage,
                                   weight_kg = wt_kg,
                                   lenht_cm = len_cm,
                                   headcirc_cm = head_cm,
                                   sex = sex,
                                   id = child)

data_classified

# Use `.outcomes` to set which growth indicators will be computed
data_svn <- classify_growth(data,
                            age_days = agedays,
                            gest_days = gestage,
                            sex = sex,
                            weight_kg = wt_kg,
                            id = child,
                            .outcomes = "svn")

data_svn

# Use `.new` to specify new column names
data_svn <- classify_growth(data,
                            age_days = agedays,
                            gest_days = gestage,
                            sex = sex,
                            weight_kg = wt_kg,
                            id = child,
                            .outcomes = "svn",
```

```
                                .new = list("svn" = c("ig_nbs_centile",
                                                      "SVN_Category")))
    data_svn

    # The function will warn you if you don't provide data for your outcomes
    data_svn_stunting <- classify_growth(data,
                                      age_days = agedays,
                                      gest_days = gestage,
                                      sex = sex,
                                      weight_kg = wt_kg,
                                      id = child,
                                      .outcomes = c("svn", "stunting"))
    data_svn_stunting
```

---

| classify_headsize | *Classify head size in* data.frame-*like objects with GIGS-recommended growth standards* |
|---|---|

---

### Description

Classify head size in data.frame-like objects with GIGS-recommended growth standards

### Usage

```
classify_headsize(
  .data,
  headcirc_cm,
  age_days,
  gest_days,
  sex,
  id = NULL,
  .new = c("hcaz", "headsize")
)
```

### Arguments

| | |
|---|---|
| .data | A data.frame-like tabular object with one or more rows. Must be permissible by [checkmate::assert_data_frame()](), so you can also supply a tibble, data.table, or similar. |
| headcirc_cm | <[data-masking]()> The name of a column in .data which is a numeric vector of head circumference values in cm. |
| age_days | <[data-masking]()> The name of a column in .data which is a numeric vector of age values in days. |
| gest_days | <[data-masking]()> The name of a column in .data which is a numeric vector of gestational age(s) at birth in days. This column, in conjunction with the column referred to by age_days, is used to select which growth standard to use for each observation. |

sex             <data-masking> The name of a column in .data which is a case-sensitive character vector of sexes, either "M" (male) or "F" (female). By default, gigs will warn you if any elements of sex are not "M" or "F", or are missing (NA). You can customise this behaviour using the GIGS package-level options.

id                <data-masking> The name of a column in .data which is a factor variable with IDs for each observation. When not NULL, this variable is used to ensure that only the first measurement taken from each infant is used as a birth measure. If all your data is from one individual, leave this parameter as NULL. Default = NULL.

.new           A three-length character vector with names for the output columns. These inputs will be repaired if necessary using vctrs::vec_as_names(), which will print any changes to the console. If any elements in .new are the same as elements in colnames(.data), the function will throw an error. Default = c("hcaz", "headsize").

## Details

Cut-offs for head size categories are:

| Category | Factor level | Z-score bounds |
|---|---|---|
| Severe microcephaly | "microcephaly_severe" | hcaz =< -3 |
| Microcephaly | "microcephaly" | -3 < hcaz =< -2 |
| Normal head circumference | "normal_headcirc" | abs(hcaz) < 2 |
| Macrocephaly | "macrocephaly" | hcaz >= 2 |
| Severe macrocephaly | "macrocephaly_severe" | hcaz >= 3 |

## Value

A tabular object of the same class that was provided as .data, with three new columns named according to .new. These columns will be (from left to right):

- hcaz - Numeric vector of weight-for-length/height zscores
- headsize - Factor of head size categories

## Note

Categorical (factor) columns produced here may contain unused factor levels. By default, gigs will inform you if these columns have unused factor levels. You can change this behaviour using the GIGS package-level option .gigs_options$handle_unused_levels.

## References

Victora CG, Schuler-Faccini L, Matijasevich A, Ribeiro E, Pessoa A, Barros FC. **Microcephaly in Brazil: how to interpret reported numbers?** *The Lancet* 2016, *387(10019):621-624* doi:10.1016/S01406736(16)002737

Accogli A, Geraldo AF, Piccolo G, Riva A, Scala M, Balagura G, et al. **Diagnostic Approach to Macrocephaly in Children**. *Frontiers in Paediatrics* 2022, *9:794069* doi:10.3389/fped.2021.794069

**See Also**

See `classify_growth()` to run this analysis and others at the same time. See `gigs_waz()` to learn which growth standard will be used for different combinations of gestational/chronological age.

**Examples**

```
# This dummy dataset contains data from two people, from birth (<3 days) to
# 500 days of age.
data <- data.frame(
  child = factor(rep.int(c("A", "B"), c(3, 3))),
  agedays = c(0, 100, 500, 2, 100, 500),
  gestage  = c(rep(35 * 7, 3), rep(35 * 7, 3)),
  sex = rep.int(c("M", "F"), c(3, 3)),
  headcirc_cm = rep.int(c(40, 50, 60), 2)
)

# Use the `id` argument to ensure that `classify_headsize()` uses the correct
# standard for each observation
data |>
  classify_headsize(headcirc_cm = headcirc_cm,
                    age_days = agedays,
                    gest_days = gestage,
                    sex = sex,
                    id = child)

# If you don't specify `id`, `classify_headsize()` will assume data is from
# one child only
data |>
  classify_headsize(headcirc_cm = headcirc_cm,
                    age_days = agedays,
                    gest_days = gestage,
                    sex = sex)
```

---

classify_sfga    *Classify size-for-gestational age in* data.frame-*like objects with the INTERGROWTH-21st weight-for-gestational age standard*

---

**Description**

Classify size-for-gestational age in data.frame-like objects with the INTERGROWTH-21st weight-for-gestational age standard

**Usage**

```
classify_sfga(
  .data,
  weight_kg,
  gest_days,
  sex,
```

```
    .new = c("birthweight_centile", "sfga", "sfga_severe")
)
```

**Arguments**

| | |
|---|---|
| .data | A data.frame-like tabular object with one or more rows. Must be permissible by [checkmate::assert_data_frame()](#), so you can also supply a tibble, data.table, or similar. |
| weight_kg | <[data-masking](#)> The name of a column in .data which is a numeric vector of birth weight values in kg. It is assumed that weight measurements provided to this function are birth weights recorded <12 hours after an infant's birth. |
| gest_days | <[data-masking](#)> The name of a column in .data which is a numeric vector of gestational age(s) at birth in days between 168 and 300. By default, gigs will warn you about elements of this vector which are outside these bounds, are NA, or Inf. You can customise this behaviour using the [GIGS package-level options](#). |
| sex | <[data-masking](#)> The name of a column in .data which is a case-sensitive character vector of sexes, either "M" (male) or "F" (female). By default, gigs will warn you if any elements of sex are not "M" or "F", or are missing (NA). You can customise this behaviour using the [GIGS package-level options](#). |
| .new | A three-length character vector with names for the output columns. These inputs will be repaired if necessary using [vctrs::vec_as_names()](#), which will print any changes to the console. If any elements in .new are the same as elements in colnames(.data), the function will throw an error. Default = c("birthweight_centile", "sfga", "sfga_severe"). |

**Details**

Cut-offs for size-for-gestational age categorisations are:

| Category | Factor level | Centile bounds |
|---|---|---|
| Severely small-for-GA | "SGA(<3)" | p < 0.03 |
| Small-for-GA | "SGA" | p < 0.1 |
| Appropriate-for-GA | "AGA" | 0.1 =< p =< 0.9 |
| Large-for-GA | "LGA" | p > 0.9 |

**Value**

A tabular object of the same class that was provided as .data, with three new columns named according to .new. These columns will be (from left to right):

- birthweight_centile - Numeric vector of birthweight centiles from the INTERGROWTH-21st Newborn Size standard for weight-for-GA
- sfga - Factor of size-for-GA categories without severe small-for-GA classification
- sfga_severe - Factor of size-for-GA categories with severe small-for-GA classification

**Note**

Categorical (factor) columns produced here may contain unused factor levels. By default, gigs will inform you if these columns have unused factor levels. You can change this behaviour using the [GIGS package-level option](#) .gigs_options$handle_unused_levels.

**References**

WHO. **Physical status: the use and interpretation of anthropometry. Report of a WHO Expert Committee.** *World Health Organisation Technical Report Series 1995,* **854: 1–452**

Royal College of Obstetricians and Gynaecologists. **The Investigation and Management of the Small-for-Gestational-Age Fetus: Green-top Guideline No. 31.** *Technical report, Royal College of Obstetricians and Gynaecologists, London, 2013.*

**See Also**

See `classify_svn()` for size-for-GA classifications which are stratified by whether a newborn is term. See `classify_growth()` to run this analysis and others at the same time.

**Examples**

```
data <- data.frame(
 wt_kg = c(2.2, 2.5, 3.3, 4.0),
 gestage = 266:269,
 sex = c("F", "M", "F", "M")
)

data |>
  classify_sfga(weight_kg = wt_kg,
                gest_days = gestage,
                sex = sex)
```

---

| classify_stunting | *Classify stunting in* `data.frame`*-like objects with GIGS-recommended growth standards* |
|---|---|

---

**Description**

Classify stunting in `data.frame`-like objects with GIGS-recommended growth standards

**Usage**

```
classify_stunting(
  .data,
  lenht_cm,
  age_days,
  gest_days,
  sex,
  id = NULL,
  .new = c("lhaz", "stunting", "stunting_outliers")
)
```

## Arguments

.data
: A data.frame-like tabular object with one or more rows. Must be permissible by checkmate::assert_data_frame(), so you can also supply a tibble, data.table, or similar.

lenht_cm
: <data-masking> The name of a column in .data which is a numeric vector of length/height values in cm.

age_days
: <data-masking> The name of a column in .data which is a numeric vector of age values in days.

gest_days
: <data-masking> The name of a column in .data which is a numeric vector of gestational age(s) at birth in days. This column, in conjunction with the column referred to by age_days, is used to select which growth standard to use for each observation.

sex
: <data-masking> The name of a column in .data which is a case-sensitive character vector of sexes, either "M" (male) or "F" (female). By default, gigs will warn you if any elements of sex are not "M" or "F", or are missing (NA). You can customise this behaviour using the GIGS package-level options.

id
: <data-masking> The name of a column in .data which is a factor variable with IDs for each observation. When not NULL, this variable is used to ensure that only the first measurement taken from each infant is used as a birth measure. If all your data is from one individual, leave this parameter as NULL. Default = NULL.

.new
: A three-length character vector with names for the output columns. These inputs will be repaired if necessary using vctrs::vec_as_names(), which will print any changes to the console. If any elements in .new are the same as elements in colnames(.data), the function will throw an error. Default = c("lhaz", "stunting", "stunting_outliers").

## Details

Cut-offs for stunting categories are:

| Category | Factor level | Z-score bounds |
|---|---|---|
| Severe stunting | "stunting_severe" | lhaz =< -3 |
| Stunting | "stunting" | -3 < lhaz =< -2 |
| No stunting | "not_stunting" | lhaz > -2 |
| Outlier | "outlier" | abs(lhaz) > 6 |

## Value

A tabular object of the same class that was provided as .data, with three new columns named according to .new. These columns will be (from left to right):

- lhaz - Numeric vector of length/height-for-age zscores

- stunting - Factor of stunting categories without outlier flagging

- stunting_outliers - Factor of stunting categories with outlier flagging

**Note**

Categorical (factor) columns produced here may contain unused factor levels. By default, gigs will inform you if these columns have unused factor levels. You can change this behaviour using the GIGS package-level option .gigs_options$handle_unused_levels.

**References**

'**Implausible z-score values**' *in* World Health Organization (ed.) *Recommendations for data collection, analysis and reporting on anthropometric indicators in children under 5 years old*. Geneva: World Health Organization and the United Nations Children's Fund UNICEF, (2019). pp. 64-65.

'**Percentage of children stunted, wasted, and underweight, and mean z-scores for stunting, wasting and underweight**' *in Guide to DHS Statistics DHS-7* Rockville, Maryland, USA: ICF (2020). pp. 431-435. https://dhsprogram.com/data/Guide-to-DHS-Statistics/Nutritional_Status.htm

**See Also**

See classify_growth() to run this analysis and others at the same time. See gigs_waz() to learn which growth standard will be used for different combinations of gestational/chronological age.

**Examples**

```
# This dummy dataset contains data from two people, from birth (<3 days) to
# 500 days of age.
data <- data.frame(
  child = factor(rep.int(c("A", "B"), c(3, 3))),
  agedays = c(0, 100, 500, 2, 100, 500),
  gestage = c(rep(35 * 7, 3), rep(40 * 7, 3)),
  sex = rep.int(c("M", "F"), c(3, 3)),
  length_cm = rep.int(c(52.2, 60.4, 75), 2)
)

# Use the `id` argument to ensure that `classify_stunting` uses the correct
# standard for each observation
data |>
  classify_stunting(lenht_cm = length_cm,
                    age_days = agedays,
                    gest_days = gestage,
                    sex = sex,
                    id = child)

# If you don't specify `id`, `classify_stunting` will assume data is from one
# child only
data |>
  classify_stunting(lenht_cm = length_cm,
                    age_days = agedays,
                    gest_days = gestage,
                    sex = sex)
```

---

classify_svn                    *Classify small vulnerable newborns in* data.frame-*like objects with*
                                *the INTERGROWTH-21st weight-for-gestational age standard*

---

### Description

Classify small vulnerable newborns in data.frame-like objects with the INTERGROWTH-21st
weight-for-gestational age standard

### Usage

```
classify_svn(
  .data,
  weight_kg,
  gest_days,
  sex,
  .new = c("birthweight_centile", "svn")
)
```

### Arguments

| | |
|---|---|
| .data | A data.frame-like tabular object with one or more rows. Must be permissible by [checkmate::assert_data_frame()](), so you can also supply a tibble, data.table, or similar. |
| weight_kg | <[data-masking]()> The name of a column in .data which is a numeric vector of birth weight values in kg. It is assumed that weight measurements provided to this function are birth weights recorded <12 hours after an infant's birth. |
| gest_days | <[data-masking]()> The name of a column in .data which is a numeric vector of gestational age(s) at birth in days between 168 and 300. By default, gigs will warn you about elements of this vector which are outside these bounds, are NA, or Inf. You can customise this behaviour using the [GIGS package-level options](). |
| sex | <[data-masking]()> The name of a column in .data which is a case-sensitive character vector of sexes, either "M" (male) or "F" (female). By default, gigs will warn you if any elements of sex are not "M" or "F", or are missing (NA). You can customise this behaviour using the [GIGS package-level options](). |
| .new | A two-length character vector with names for the output columns. These inputs will be repaired if necessary using [vctrs::vec_as_names()](), which will print any changes to the console. If any elements in .new are the same as elements in colnames(.data), the function will throw an error. Default = c("birthweight_centile", "svn"). |

### Details

Cut-offs for small, vulnerable newborn categorisations are:

| SVN category | Factor level | Centile bounds | Gestational age range |
|---|---|---|---|
| Preterm-SGA | "Preterm SGA" | p < 0.1 | gest_days < 259 |
| Preterm-AGA | "Preterm AGA" | 0.1 =< p =< 0.9 | gest_days < 259 |
| Preterm-LGA | "Preterm LGA" | p > 0.9 | gest_days < 259 |
| Term-SGA | "Term SGA" | p < 0.1 | gest_days >= 259 |
| Term-AGA | "Term AGA" | 0.1 =< p =< 0.9 | gest_days >= 259 |
| Term-LGA | "Term LGA" | p > 0.9 | gest_days >= 259 |

*Abbreviations:* SGA, small-for-gestational age; AGA, appropriate-for-gestational age; LGA, large-for-gestational age.

## Value

A tabular object of the same class that was provided as `.data`, with two new columns named according to `.new`. These columns will be (from left to right):

- `birthweight_centile` - Numeric vector of birthweight centiles from the INTERGROWTH-21st Newborn Size standard for weight-for-GA
- `svn` - Factor of small vulnerable newborn (SVN) categories

## Note

Categorical (factor) columns produced here may contain unused factor levels. By default, gigs will inform you if these columns have unused factor levels. You can change this behaviour using the GIGS package-level option `.gigs_options$handle_unused_levels`.

## References

WHO. **Physical status: the use and interpretation of anthropometry. Report of a WHO Expert Committee.** *World Health Organisation Technical Report Series 1995,* **854: 1–452**

Royal College of Obstetricians and Gynaecologists. **The Investigation and Management of the Small-for-Gestational-Age Fetus: Green-top Guideline No. 31.** *Technical report, Royal College of Obstetricians and Gynaecologists, London, 2013.*

## See Also

See `classify_sfga()` for size-for-GA classifications which are not stratified by whether a newborn is term. See `classify_growth()` to run this analysis and others at the same time.

## Examples

```
data <- data.frame(
  wt_kg = c(1.5, 2.6, 2.6, 3.5),
  gestage = c(235, 257, 275, 295),
  sex = c("F", "M", "F", "M")
)

data |>
  classify_svn(weight_kg = wt_kg,
               gest_days = gestage,
```

```
        sex = sex)
```

---

classify_wasting *Classify wasting in* data.frame-*like objects with GIGS-recommended growth standards*

---

## Description

Classify wasting in data.frame-like objects with GIGS-recommended growth standards

## Usage

```
classify_wasting(
  .data,
  weight_kg,
  lenht_cm,
  age_days,
  gest_days,
  sex,
  id = NULL,
  .new = c("wlz", "wasting", "wasting_outliers")
)
```

## Arguments

| | |
|---|---|
| .data | A data.frame-like tabular object with one or more rows. Must be permissible by [checkmate::assert_data_frame()](#), so you can also supply a tibble, data.table, or similar. |
| weight_kg | <[data-masking](#)> The name of a column in .data which is a numeric vector of birth weight values in kg. It is assumed that weight measurements provided to this function are birth weights recorded <12 hours after an infant's birth. |
| lenht_cm | <[data-masking](#)> The name of a column in .data which is a numeric vector of length/height values in cm. |
| age_days | <[data-masking](#)> The name of a column in .data which is a numeric vector of age values in days. |
| gest_days | <[data-masking](#)> The name of a column in .data which is a numeric vector of gestational age(s) at birth in days. This column, in conjunction with the column referred to by age_days, is used to select which growth standard to use for each observation. |
| sex | <[data-masking](#)> The name of a column in .data which is a case-sensitive character vector of sexes, either "M" (male) or "F" (female). By default, gigs will warn you if any elements of sex are not "M" or "F", or are missing (NA). You can customise this behaviour using the [GIGS package-level options](#). |

| id | <data-masking> The name of a column in .data which is a factor variable with IDs for each observation. When not NULL, this variable is used to ensure that only the first measurement taken from each infant is used as a birth measure. If all your data is from one individual, leave this parameter as NULL. Default = NULL. |
|---|---|
| .new | A three-length character vector with names for the output columns. These inputs will be repaired if necessary using vctrs::vec_as_names(), which will print any changes to the console. If any elements in .new are the same as elements in colnames(.data), the function will throw an error. Default = c("wlz", "wasting", "wasting_outliers"). |

### Details

Cut-offs for wasting categories are:

| Category | Factor level | Z-score bounds |
|---|---|---|
| Severe wasting | "wasting_severe" | wlz =< -3 |
| Wasting | "wasting" | -3 < wlz =< -2 |
| No wasting | "not_wasting" | abs(wlz) < 2 |
| Overweight | "overweight" | wlz >= 2 |
| Outlier | "outlier" | abs(wlz) > 5 |

### Value

A tabular object of the same class that was provided as .data, with three new columns named according to .new. These columns will be (from left to right):

- wlz - Numeric vector of weight-for-length/height zscores

- wasting - Factor of wasting categories without outlier flagging

- wasting_outliers - Factor of wasting categories with outlier flagging

### Note

Categorical (factor) columns produced here may contain unused factor levels. By default, gigs will inform you if these columns have unused factor levels. You can change this behaviour using the GIGS package-level option .gigs_options$handle_unused_levels.

### References

'**Implausible z-score values**' *in* World Health Organization (ed.) *Recommendations for data collection, analysis and reporting on anthropometric indicators in children under 5 years old*. Geneva: World Health Organization and the United Nations Children's Fund UNICEF, (2019). pp. 64-65.

'**Percentage of children stunted, wasted, and underweight, and mean z-scores for stunting, wasting and underweight**' *in Guide to DHS Statistics DHS-7* Rockville, Maryland, USA: ICF (2020). pp. 431-435. https://dhsprogram.com/data/Guide-to-DHS-Statistics/Nutritional_Status.htm

**See Also**

See `classify_growth()` to run this analysis and others at the same time. See `gigs_waz()` to learn which growth standard will be used for different combinations of gestational/chronological age.

**Examples**

```
# This dummy dataset contains data from two people, from birth (<3 days) to
# 500 days of age.
data <- data.frame(
  child = factor(rep.int(c("A", "B"), c(3, 3))),
  agedays = c(0, 100, 500, 2, 100, 500),
  gestage  = c(rep(35 * 7, 3), rep(35 * 7, 3)),
  sex = rep.int(c("M", "F"), c(3, 3)),
  weight_kg = c(3, 6, 9, 3, 6, 9),
  length_cm = rep.int(c(52.2, 60.4, 75), 2)
)

# Use the `id` argument to ensure that `classify_wasting` uses the correct
# standard for each observation
data |>
  classify_wasting(weight_kg = weight_kg,
                   lenht_cm = length_cm,
                   age_days = agedays,
                   gest_days = gestage,
                   sex = sex,
                   id = child)

# If you don't specify `id`, `classify_wasting` will assume data is from one
# child only
data |>
  classify_wasting(weight_kg = weight_kg,
                   lenht_cm = length_cm,
                   age_days = agedays,
                   gest_days = gestage,
                   sex = sex)
```

---

classify_wfa  *Classify weight-for-age in* data.frame*-like objects with GIGS-recommended growth standards*

---

**Description**

Classify weight-for-age in data.frame-like objects with GIGS-recommended growth standards

**Usage**

```
classify_wfa(
  .data,
  weight_kg,
```

```
    age_days,
    gest_days,
    sex,
    id = NULL,
    .new = c("waz", "wfa", "wfa_outliers")
)
```

## Arguments

.data
: A data.frame-like tabular object with one or more rows. Must be permissible by [checkmate::assert_data_frame()](), so you can also supply a tibble, data.table, or similar.

weight_kg
: <[data-masking]()> The name of a column in .data which is a numeric vector of birth weight values in kg. It is assumed that weight measurements provided to this function are birth weights recorded <12 hours after an infant's birth.

age_days
: <[data-masking]()> The name of a column in .data which is a numeric vector of age values in days.

gest_days
: <[data-masking]()> The name of a column in .data which is a numeric vector of gestational age(s) at birth in days. This column, in conjunction with the column referred to by age_days, is used to select which growth standard to use for each observation.

sex
: <[data-masking]()> The name of a column in .data which is a case-sensitive character vector of sexes, either "M" (male) or "F" (female). By default, gigs will warn you if any elements of sex are not "M" or "F", or are missing (NA). You can customise this behaviour using the [GIGS package-level options]().

id
: <[data-masking]()> The name of a column in .data which is a factor variable with IDs for each observation. When not NULL, this variable is used to ensure that only the first measurement taken from each infant is used as a birth measure. If all your data is from one individual, leave this parameter as NULL. Default = NULL.

.new
: A three-length character vector with names for the output columns. These inputs will be repaired if necessary using [vctrs::vec_as_names()](), which will print any changes to the console. If any elements in .new are the same as elements in colnames(.data), the function will throw an error. Default = c("waz", "wfa", "wfa_outliers").

## Details

Cut-offs for weight-for-age categories are:

| Category | Factor level | Z-score bounds |
|---|---|---|
| Severely underweight | "underweight_severe" | waz =< -3 |
| Underweight | "underweight" | -3 < waz =< -2 |
| Normal weight | "normal_weight" | abs(waz) < 2 |
| Overweight | "overweight" | waz >= 2 |
| Outlier | "outlier" | waz < -6 or waz > 5 |

**Value**

A tabular object of the same class that was provided as .data, with three new columns named according to .new. These columns will be (from left to right):

- wlz - Numeric vector of weight-for-length/height zscores
- wfa - Factor of weight-for-age categories without outlier flagging
- wfa_outliers - Factor of weight-for-age categories with outlier flagging

**Note**

Categorical (factor) columns produced here may contain unused factor levels. By default, gigs will inform you if these columns have unused factor levels. You can change this behaviour using the GIGS package-level option .gigs_options$handle_unused_levels.

**References**

'Implausible z-score values' *in* World Health Organization (ed.) *Recommendations for data collection, analysis and reporting on anthropometric indicators in children under 5 years old*. Geneva: World Health Organization and the United Nations Children's Fund UNICEF, (2019). pp. 64-65.

'Percentage of children stunted, wasted, and underweight, and mean z-scores for stunting, wasting and underweight' *in Guide to DHS Statistics DHS-7* Rockville, Maryland, USA: ICF (2020). pp. 431-435. https://dhsprogram.com/data/Guide-to-DHS-Statistics/Nutritional_Status.htm

**See Also**

See classify_growth() to run this analysis and others at the same time. See gigs_waz() to learn which growth standard will be used for different combinations of gestational/chronological age.

**Examples**

```
# This dummy dataset contains data from two people, from birth (<3 days) to
# 500 days of age.
data <- data.frame(
  child = factor(rep.int(c("A", "B"), c(3, 3))),
  agedays = c(0, 100, 500, 2, 100, 500),
  gestage  = c(rep(35 * 7, 3), rep(35 * 7, 3)),
  sex = rep.int(c("M", "F"), c(3, 3)),
  weight_kg = c(3, 6, 9, 3, 6, 9)
)

# Use the `id` argument to ensure that `classify_wfa()` uses the correct
# standard for each observation
data |>
  classify_wfa(weight_kg = weight_kg,
               age_days = agedays,
               gest_days = gestage,
               sex = sex,
               id = child)
```

```
# If you don't specify `id`, `classify_wfa()` will assume data is from one
# child only
data |>
  classify_wfa(weight_kg = weight_kg,
               age_days = agedays,
               gest_days = gestage,
               sex = sex)
```

---

| compute_headsize | *Get head size categories using multiple vectors and GIGS-recommended growth standards* |
|---|---|

---

### Description

Get head size categories using multiple vectors and GIGS-recommended growth standards

### Usage

```
compute_headsize(headcirc_cm, age_days, gest_days, sex, id = NULL)
```

### Arguments

headcirc_cm    Numeric vector of length one or more with head circumference measurement(s) in cm.

age_days    Numeric vector of length one or more with age(s) in days for each child. Should be between 0 to 1856 days. By default, gigs will replace out-of-bounds elements in age_days with NA and warn you. This behaviour can be customised using the functions in gigs_options.

gest_days    Numeric vector with gestational age(s) at birth in days.

sex    Character vector of length one or more with sex(es), either "M" (male) or "F" (female). This argument is case-sensitive. By default, gigs will replace elements of sex which are not "M" or "F" with NA and warn you. You can customise this behaviour using the GIGS package-level options.

id    A factor of length one or more with an ID for each observation, either ordered or unordered, containing no missing (NA) values. When supplied, id is used to ensure that only the earliest measurement for each individual is used as a birth measure. Leave this argument as NULL if all your data comes from the same individual. Default = NULL.

### Details

Cut-offs for head size categories are:

| Category | Factor level | Z-score bounds |
|---|---|---|
| Severe microcephaly | "microcephaly_severe" | hcaz =< -3 |
| Microcephaly | "microcephaly" | -3 < hcaz =< -2 |

| Normal head circumference | "normal_headcirc" | abs(hcaz) < 2 |
| Macrocephaly | "macrocephaly" | hcaz >= 2 |
| Severe macrocephaly | "macrocephaly_severe" | hcaz >= 3 |

## Value

An object of class factor with the same length as wlz, containing wasting classifications. Its levels are c("wasting_severe", "wasting", "not_wasting", "overweight") if outliers = FALSE (the default), else c("wasting_severe", "wasting", "not_wasting", "overweight", "outlier"). By default, gigs will inform you this object contains unused factor levels. You can change this behaviour using the GIGS package-level option .gigs_options$handle_unused_levels.

## References

'**Implausible z-score values**' *in* World Health Organization (ed.) *Recommendations for data collection, analysis and reporting on anthropometric indicators in children under 5 years old*. Geneva: World Health Organization and the United Nations Children's Fund UNICEF, (2019). pp. 64-65.

'**Percentage of children stunted, wasted, and underweight, and mean z-scores for stunting, wasting and underweight**' *in Guide to DHS Statistics DHS-7* Rockville, Maryland, USA: ICF (2020). pp. 431-435. https://dhsprogram.com/data/Guide-to-DHS-Statistics/Nutritional_Status.htm

## Examples

```
# The first observation for each infant in `id` uses the INTERGROWTH-21st
#  Newborn Size standards; the next two use either the INTERGROWTH-21st
# Postnatal Growth standards or WHO Child Growth Standards.
compute_headsize(
  headcirc_cm = c(31.6, 40, 50, 31.6, 40, 50),
  age_days = c(0, 100, 500, 2, 100, 500),
  gest_days = c(245, 245, 245, 280, 280, 280),
  sex = c("M", "M", "M", "F", "F", "F"),
  id = factor(c("A", "A", "A", "B", "B", "B"))
)

# If you don't specify `id`, the function will not identify that the fourth
# data point is a birth measurement, and will categorise it as microcephalic
# instead of normal
compute_headsize(
  headcirc_cm = c(31.6, 40, 50, 31.6, 40, 50),
  age_days = c(0, 100, 500, 2, 100, 500),
  gest_days = c(245, 245, 245, 280, 280, 280),
  sex = c("M", "M", "M", "F", "F", "F")
)
```

---

compute_sfga                    *Get size-for-gestational age categories using multiple vectors and the*
                                *INTERGROWTH-21st weight-for-gestational age standard*

---

### Description

Get size-for-gestational age categories using multiple vectors and the INTERGROWTH-21st weight-for-gestational age standard

### Usage

```
compute_sfga(weight_kg, gest_days, sex, severe = FALSE)
```

### Arguments

weight_kg       Numeric vector of length one or more with weight value(s) in kg. It is assumed
                that weight measurements provided to this function are birth weights recorded
                <12 hours after an infant's birth.

gest_days       Numeric vector of length one or more with gestational ages in days. Classifica-
                tions can only be computed when gest_days is between 168 and 300, as these
                are the limits of the INTERGROWTH-21st Newborn Size standards (including
                the Very Preterm standards). By default, gigs will warn you about elements of
                gest_days which are either outside these bounds, are NA, or are Inf. You can
                customise this behaviour using the GIGS package-level options.

sex             Character vector of length one or more with sex(es), either "M" (male) or "F"
                (female). This argument is case-sensitive. By default, gigs will replace elements
                of sex which are not "M" or "F" with NA and warn you. You can customise this
                behaviour using the GIGS package-level options.

severe          A single logical value specifying whether to categorise SGA values below the
                third centile (0.03) as "SGA(<3)". Default = FALSE.

### Details

Cut-offs for size-for-gestational age categorisations are:

| Category | Factor level | Centile bounds |
|---|---|---|
| Severely small-for-GA | "SGA(<3)" | p < 0.03 |
| Small-for-GA | "SGA" | p < 0.1 |
| Appropriate-for-GA | "AGA" | 0.1 =< p =< 0.9 |
| Large-for-GA | "LGA" | p > 0.9 |

### Value

An object of class factor with the same length as the longest input vector, containing size-for-GA classifications. If severe = FALSE, levels are c("SGA", "AGA", "LGA"). If severe = TRUE, levels are c("SGA(<3)", "SGA", "AGA", "LGA"). By default, gigs will inform you this object contains unused factor levels. You can change this behaviour using the GIGS package-level option .gigs_options$handle_unused_levels.

## Note

Input vectors are recycled by `vctrs::vec_recycle_common()`, and must adhere to the vctrs recycling rules.

## References

WHO. **Physical status: the use and interpretation of anthropometry. Report of a WHO Expert Committee.** *World Health Organisation Technical Report Series 1995,* **854: 1–452**

Royal College of Obstetricians and Gynaecologists. **The Investigation and Management of the Small-for-Gestational-Age Fetus: Green-top Guideline No. 31.** *Technical report, Royal College of Obstetricians and Gynaecologists, London, 2013.*

## Examples

```
# By default, does not differentiate between p < 0.03 and p < 0.10
compute_sfga(
  weight_kg = c(2.2, 2.5, 3.3, 4.0),
  gest_days = 266:269,
  sex = c("F", "M", "F", "M")
)

# With severe = TRUE, highlights p < 0.03
compute_sfga(
  weight_kg = c(2.2, 2.5, 3.3, 4.0),
  gest_days = 266:269,
  sex = c("F", "M", "F", "M"),
  severe = TRUE
)
```

---

compute_stunting                      *Get stunting categories using multiple vectors and GIGS-recommended growth standards*

---

## Description

Get stunting categories using multiple vectors and GIGS-recommended growth standards

## Usage

```
compute_stunting(
  lenht_cm,
  age_days,
  gest_days,
  sex,
  id = NULL,
  outliers = FALSE
)
```

**Arguments**

| | |
|---|---|
| lenht_cm | Numeric vector of length one or more with length/height measurement(s) in cm. |
| age_days | Numeric vector of length one or more with age(s) in days for each child. Should be between 0 to 1856 days. By default, gigs will replace out-of-bounds elements in age_days with NA and warn you. This behaviour can be customised using the functions in gigs_options. |
| gest_days | Numeric vector with gestational age(s) at birth in days. |
| sex | Character vector of length one or more with sex(es), either "M" (male) or "F" (female). This argument is case-sensitive. By default, gigs will replace elements of sex which are not "M" or "F" with NA and warn you. You can customise this behaviour using the GIGS package-level options. |
| id | A factor of length one or more with an ID for each observation, either ordered or unordered, containing no missing (NA) values. When supplied, id is used to ensure that only the earliest measurement for each individual is used as a birth measure. Leave this argument as NULL if all your data comes from the same individual. Default = NULL. |
| outliers | A single TRUE or FALSE value specifying whether implausible z-score thresholds should be applied. Default = FALSE. |

**Details**

Cut-offs for stunting categories are:

| Category | Factor level | Z-score bounds |
|---|---|---|
| Severe stunting | "stunting_severe" | lhaz =< -3 |
| Stunting | "stunting" | -3 < lhaz =< -2 |
| No stunting | "not_stunting" | lhaz > -2 |
| Outlier | "outlier" | abs(lhaz) > 6 |

**Value**

An object of class factor with the same length as the longest input vector, containing stunting classifications. Its levels are c("stunting_severe", "stunting", "not_stunting") if outliers = FALSE (the default), else c("stunting_severe", "stunting", "not_stunting", "outlier"). By default, gigs will inform you this object contains unused factor levels. You can change this behaviour using the GIGS package-level option .gigs_options$handle_unused_levels.

**Note**

This function assumes that your measurements were taken according to WHO guidelines, which stipulate that recumbent length should not be measured after 730 days. Instead, standing height should be used. Implausible z-score bounds are sourced from the referenced WHO report, and classification cut-offs from the DHS manual.

## References

**'Implausible z-score values'** *in* World Health Organization (ed.) *Recommendations for data collection, analysis and reporting on anthropometric indicators in children under 5 years old*. Geneva: World Health Organization and the United Nations Children's Fund UNICEF, (2019). pp. 64-65.

**'Percentage of children stunted, wasted, and underweight, and mean z-scores for stunting, wasting and underweight'** *in Guide to DHS Statistics DHS-7* Rockville, Maryland, USA: ICF (2020). pp. 431-435. `https://dhsprogram.com/data/Guide-to-DHS-Statistics/Nutritional_Status.htm`

## Examples

```
# The first observation for each infant in `id` uses the INTERGROWTH-21st
#  Newborn Size standards; the next two use either the INTERGROWTH-21st
# Postnatal Growth standards or WHO Child Growth Standards.
compute_stunting(
  lenht_cm = c(52.2, 60.4, 75, 52.2, 60.4, 75),
  age_days = c(0, 100, 500, 2, 100, 500),
  gest_days = c(245, 245, 245, 280, 280, 280),
  sex = c("M", "M", "M", "F", "F", "F"),
  id = factor(c("A", "A", "A", "B", "B", "B"))
)

# With outlier flagging:
compute_stunting(
  lenht_cm = c(52.2, 75.4, 63.1),
  age_days = c(357, 375, 250),
  gest_days = c(196, 287, 266),
  sex = c("M", "M", "F"),
  id = factor(c("A", "B", "C")),
  outliers = TRUE
)

# If you don't specify `id`, the function will not identify that the fourth
# data point is a birth measurement, and will categorise it as severe
# stunting instead of not stunting
compute_stunting(
  lenht_cm = c(40.0, 60.4, 75, 40.0, 60.4, 75),
  age_days = c(0.45, 100, 500, 0.5, 100, 500),
  gest_days = c(245, 245, 245, 280, 280, 280),
  sex = c("M", "M", "M", "F", "F", "F")
)
```

---

| compute_svn | *Get small vulnerable newborn categories using multiple vectors and the INTERGROWTH-21st weight-for-gestational age standard* |
|---|---|

---

## Description

Get small vulnerable newborn categories using multiple vectors and the INTERGROWTH-21st weight-for-gestational age standard

**Usage**

```
compute_svn(weight_kg, gest_days, sex)
```

**Arguments**

| | |
|---|---|
| weight_kg | Numeric vector of length one or more with weight value(s) in kg. It is assumed that weight measurements provided to this function are birth weights recorded <12 hours after an infant's birth. |
| gest_days | Numeric vector of length one or more with gestational ages in days. Classifications can only be computed when gest_days is between 168 and 300, as these are the limits of the INTERGROWTH-21st Newborn Size standards (including the Very Preterm standards). By default, gigs will warn you about elements of gest_days which are either outside these bounds, are NA, or are Inf. You can customise this behaviour using the GIGS package-level options. |
| sex | Character vector of length one or more with sex(es), either "M" (male) or "F" (female). This argument is case-sensitive. By default, gigs will replace elements of sex which are not "M" or "F" with NA and warn you. You can customise this behaviour using the GIGS package-level options. |

**Details**

Cut-offs for small, vulnerable newborn categorisations are:

| SVN category | Factor level | Centile bounds | Gestational age range |
|---|---|---|---|
| Preterm-SGA | "Preterm SGA" | p < 0.1 | gest_days < 259 |
| Preterm-AGA | "Preterm AGA" | 0.1 =< p =< 0.9 | gest_days < 259 |
| Preterm-LGA | "Preterm LGA" | p > 0.9 | gest_days < 259 |
| Term-SGA | "Term SGA" | p < 0.1 | gest_days >= 259 |
| Term-AGA | "Term AGA" | 0.1 =< p =< 0.9 | gest_days >= 259 |
| Term-LGA | "Term LGA" | p > 0.9 | gest_days >= 259 |

*Abbreviations:* SGA, small-for-gestational age; AGA, appropriate-for-gestational age; LGA, large-for-gestational age.

**Value**

An object of class factor with the same length as the longest input vector, containing small vulnerable newborn classifications. Its levels are c("Preterm SGA", "Preterm AGA", "Preterm LGA", "Term SGA", "Term AGA", "Term LGA"). By default, gigs will inform you this object contains unused factor levels. You can change this behaviour using the GIGS package-level option .gigs_options$handle_unused_lev

**Note**

Input vectors are recycled by vctrs::vec_recycle_common(), and must adhere to the vctrs recycling rules.

**References**

Lawn JE, Ohuma EO, Bradley E, Idueta LS, Hazel E, Okwaraji YB et al. **Small babies, big risks: global estimates of prevalence and mortality for vulnerable newborns to accelerate change and improve counting.** *The Lancet* 2023, *401(10389):1707-1719.* [doi:10.1016/S0140-6736(23)005226](doi:10.1016/S0140-6736(23)005226)

**Examples**

```
compute_svn(
  weight_kg = c(1.5, 2.6, 2.6, 3.5),
  gest_days = c(235, 257, 275, 295),
  sex = c("F", "M", "F", "M")
)
```

---

compute_wasting                *Get wasting categories using multiple vectors and GIGS-recommended growth standards*

---

**Description**

Get wasting categories using multiple vectors and GIGS-recommended growth standards

**Usage**

```
compute_wasting(
  weight_kg,
  lenht_cm,
  age_days,
  gest_days,
  sex,
  id = NULL,
  outliers = FALSE
)
```

**Arguments**

| | |
|---|---|
| weight_kg | Numeric vector of length one or more with weight measurement(s) in kg. |
| lenht_cm | Numeric vector of length one or more with length/height measurement(s) in cm. |
| age_days | Numeric vector of length one or more with age(s) in days for each child. Should be between 0 to 1856 days. By default, gigs will replace out-of-bounds elements in age_days with NA and warn you. This behaviour can be customised using the functions in [gigs_options](gigs_options). |
| gest_days | Numeric vector with gestational age(s) at birth in days. |
| sex | Character vector of length one or more with sex(es), either "M" (male) or "F" (female). This argument is case-sensitive. By default, gigs will replace elements of sex which are not "M" or "F" with NA and warn you. You can customise this behaviour using the [GIGS package-level options](GIGS package-level options). |

id A factor of length one or more with an ID for each observation, either ordered or unordered, containing no missing (NA) values. When supplied, id is used to ensure that only the earliest measurement for each individual is used as a birth measure. Leave this argument as NULL if all your data comes from the same individual. Default = NULL.

outliers A single TRUE or FALSE value specifying whether implausible z-score thresholds should be applied. Default = FALSE.

## Details

Cut-offs for wasting categories are:

| Category | Factor level | Z-score bounds |
|----------|--------------|----------------|
| Severe wasting | "wasting_severe" | wlz =< -3 |
| Wasting | "wasting" | -3 < wlz =< -2 |
| No wasting | "not_wasting" | abs(wlz) < 2 |
| Overweight | "overweight" | wlz >= 2 |
| Outlier | "outlier" | abs(wlz) > 5 |

## Value

An object of class factor with the same length as wlz, containing wasting classifications. Its levels are c("wasting_severe", "wasting", "not_wasting", "overweight") if outliers = FALSE (the default), else c("wasting_severe", "wasting", "not_wasting", "overweight", "outlier"). By default, gigs will inform you this object contains unused factor levels. You can change this behaviour using the GIGS package-level option .gigs_options$handle_unused_levels.

## Note

This function assumes that your measurements were taken according to WHO guidelines, which stipulate that recumbent length should not be measured after 730 days. Instead, standing height should be used. Implausible z-score bounds are sourced from the referenced WHO report, and classification cut-offs from the DHS manual.

## References

'**Implausible z-score values'** *in* World Health Organization (ed.) *Recommendations for data collection, analysis and reporting on anthropometric indicators in children under 5 years old*. Geneva: World Health Organization and the United Nations Children's Fund UNICEF, (2019). pp. 64-65.

'**Percentage of children stunted, wasted, and underweight, and mean z-scores for stunting, wasting and underweight'** *in Guide to DHS Statistics DHS-7* Rockville, Maryland, USA: ICF (2020). pp. 431-435. https://dhsprogram.com/data/Guide-to-DHS-Statistics/Nutritional_Status.htm

## Examples

```
# The first observation for each infant in `id` uses the INTERGROWTH-21st
#  Newborn Size standards; the next two use either the INTERGROWTH-21st
# Postnatal Growth standards or WHO Child Growth Standards.
```

```
compute_wasting(
  weight_kg = c(3, 6, 11, 3, 6, 11),
  lenht_cm = c(52.2, 60.4, 61, 52.2, 60.4, 61),
  age_days = c(0, 100, 500, 2, 100, 500),
  gest_days = c(245, 245, 245, 280, 280, 280),
  sex = c("M", "M", "M", "F", "F", "F"),
  id = factor(c("A", "A", "A", "B", "B", "B"))
)

# With outlier flagging:
compute_wasting(
  weight_kg = c(3, 6, 11, 3, 6, 11),
  lenht_cm = c(52.2, 60.4, 61, 52.2, 60.4, 61),
  age_days = c(0, 100, 500, 2, 100, 500),
  gest_days = c(245, 245, 245, 280, 280, 280),
  sex = c("M", "M", "M", "F", "F", "F"),
  id = factor(c("A", "A", "A", "B", "B", "B")),
  outliers = TRUE
)

# If you don't specify `id`, the function will not identify that the fourth
# data point is a birth measurement. This means a weight-for-length z-score
# (and resulting wasting classification) will be calculated for the fourth
# data point, which is wrong here
compute_wasting(
  weight_kg = c(3, 6, 11, 3, 6, 11),
  lenht_cm = c(52.2, 60.4, 61, 52.2, 60.4, 61),
  age_days = c(0, 100, 500, 2, 100, 500),
  gest_days = c(245, 245, 245, 280, 280, 280),
  sex = c("M", "M", "M", "F", "F", "F")
)
```

---

compute_wfa                    *Get weight-for-age categories using multiple vectors and GIGS-*
                               *recommended growth standards*

---

### Description

Get weight-for-age categories using multiple vectors and GIGS-recommended growth standards

### Usage

```
compute_wfa(weight_kg, age_days, gest_days, sex, id = NULL, outliers = FALSE)
```

### Arguments

weight_kg        Numeric vector of length one or more with weight measurement(s) in kg.

| | |
|---|---|
| age_days | Numeric vector of length one or more with age(s) in days for each child. Should be between 0 to 1856 days. By default, gigs will replace out-of-bounds elements in age_days with NA and warn you. This behaviour can be customised using the functions in [gigs_options](#). |
| gest_days | Numeric vector with gestational age(s) at birth in days. |
| sex | Character vector of length one or more with sex(es), either "M" (male) or "F" (female). This argument is case-sensitive. By default, gigs will replace elements of sex which are not "M" or "F" with NA and warn you. You can customise this behaviour using the [GIGS package-level options](#). |
| id | A factor of length one or more with an ID for each observation, either ordered or unordered, containing no missing (NA) values. When supplied, id is used to ensure that only the earliest measurement for each individual is used as a birth measure. Leave this argument as NULL if all your data comes from the same individual. Default = NULL. |
| outliers | A single TRUE or FALSE value specifying whether implausible z-score thresholds should be applied. Default = FALSE. |

## Details

Cut-offs for weight-for-age categories are:

| Category | Factor level | Z-score bounds |
|---|---|---|
| Severely underweight | "underweight_severe" | waz =< -3 |
| Underweight | "underweight" | -3 < waz =< -2 |
| Normal weight | "normal_weight" | abs(waz) < 2 |
| Overweight | "overweight" | waz >= 2 |
| Outlier | "outlier" | waz < -6 or waz > 5 |

## Value

An object of class factor with the same length as wlz, containing wasting classifications. Its levels are c("wasting_severe", "wasting", "not_wasting", "overweight") if outliers = FALSE (the default), else c("wasting_severe", "wasting", "not_wasting", "overweight", "outlier"). By default, gigs will inform you this object contains unused factor levels. You can change this behaviour using the [GIGS package-level option](#) .gigs_options$handle_unused_levels.

## Note

This function assumes that your measurements were taken according to WHO guidelines, which stipulate that recumbent length should not be measured after 730 days. Instead, standing height should be used. Implausible z-score bounds are sourced from the referenced WHO report, and classification cut-offs from the DHS manual.

## References

**'Implausible z-score values'** *in* World Health Organization (ed.) *Recommendations for data collection, analysis and reporting on anthropometric indicators in children under 5 years old*. Geneva: World Health Organization and the United Nations Children's Fund UNICEF, (2019). pp. 64-65.

**'Percentage of children stunted, wasted, and underweight, and mean z-scores for stunting, wasting and underweight'** *in Guide to DHS Statistics DHS-7* Rockville, Maryland, USA: ICF (2020). pp. 431-435. [https://dhsprogram.com/data/Guide-to-DHS-Statistics/Nutritional_Status.htm](https://dhsprogram.com/data/Guide-to-DHS-Statistics/Nutritional_Status.htm)

**Examples**

```
# The first observation for each infant in `id` uses the INTERGROWTH-21st
#  Newborn Size standards; the next two use either the INTERGROWTH-21st
# Postnatal Growth standards or WHO Child Growth Standards.
compute_wfa(
  weight_kg = c(2.05, 8, 18, 2.05, 8, 18),
  age_days = c(0, 100, 500, 2, 100, 500),
  gest_days = c(245, 245, 245, 280, 280, 280),
  sex = c("M", "M", "M", "F", "F", "F"),
  id = factor(c("A", "A", "A", "B", "B", "B"))
)

# With outlier flagging:
compute_wfa(
  weight_kg = c(2.05, 8, 18, 2.05, 8, 18),
  age_days = c(0, 100, 500, 2, 100, 500),
  gest_days = c(245, 245, 245, 280, 280, 280),
  sex = c("M", "M", "M", "F", "F", "F"),
  id = factor(c("A", "A", "A", "B", "B", "B")),
  outliers = TRUE
)

# If you don't specify `id`, the function will not identify that the fourth
# data point is a birth measurement, and will categorise it as underweight
# instead of severely stunting
compute_wfa(
  weight_kg = c(2.05, 8, 18, 2.05, 8, 18),
  age_days = c(0, 100, 500, 2, 100, 500),
  gest_days = c(245, 245, 245, 280, 280, 280),
  sex = c("M", "M", "M", "F", "F", "F")
)
```

---

gigs_options　　　　　　　*Get and set gigs package-level options*

---

**Description**

By default, gigs will handle invalid input data, e.g. missing (NA) or undefined (NaN, Inf and -Inf) values, by warning you of their presence and replacing the invalid values with NA. It will also warn you about unused factor levels when running growth outcome analyses.

You can change this input-checking behaviour with gigs_option_set() so that gigs does this silently, or make gigs throw errors when confronted with bad data. You can also make gigs drop unused factor levels in its outputs, or silence its warnings about unused factor levels. You can also change all options for gigs input-checking at once using gigs_input_options_set().

**Usage**

```
gigs_option_get(option, silent = FALSE)

gigs_option_set(option, new_value, silent = FALSE)

gigs_input_options_set(new_value, silent = FALSE)
```

**Arguments**

option          A single case-sensitive character variable, one of:

- "handle_missing_data" - How should gigs handle missing (NA) elements?
- "handle_undefined_data" - How should gigs handle undefined (NaN, Inf, -Inf) elements?
- "handle_oob_xvar" - How should gigs handle elements of x variables which are out of bounds for the standard in use?
- "handle_invalid_sex" - How should gigs handle elements of sex which are not one of "M" or "F"?
- "handle_oob_centiles" - In centile2value functions, how should gigs handle elements of p that are not between 0 and 1?
- "handle_unused_levels" - When generating growth categories as factors, should gigs drop or keep unused factor levels, and should it issue warnings when unused factor levels occur?

silent          A single logical value controlling whether a message will be printed to the console describing either:

- The current value of option in gigs_option_get().
- The new_value of option in gigs_option_set().

This argument will not accept an integer in place of a logical value.

new_value       A single case-sensitive character variable. For all options **except** "handle_unused_levels", this should be one of:

- "quiet" - gigs will replace invalid vector elements with NA quietly.
- "warn" - gigs will replace invalid vector elements with NA loudly.
- "error" - gigs will throw informative errors if any invalid inputs are encountered.

For "handle_unused_levels", new_value should be one of:

- "keep_quiet" - Keep unused factor levels, and don't issue a warning.
- "keep_warn" - Keep unused factor levels, whilst issuing a warning.
- "drop_quiet" - Drop unused factor levels, and don't issue a warning.
- "drop_warn" - Drop unused factor levels, whilst issuing a warning.

By default, GIGS options are "warn" for input handling and "keep_warn" for "handle_unused_levels".

**Value**

A single-length character vector. For `gigs_option_get()`, the current value of `option`; for `gigs_option_get()`, the new value of `option`; for `gigs_input_options_set()`, `new_value`. If `silent = FALSE` (default), then informative messages will be printed to the console regarding the current/new values of `option`.

**Examples**

```
# Show the names of all available options
names(.gigs_options)

# Retrieve the value of a gigs option (if `silent = FALSE`, would print a
# message)
option_value <- gigs_option_get("handle_missing_data", silent = TRUE)
print(option_value)

# Set the value of an option
gigs_option_set("handle_undefined_data", "error", silent = TRUE)

# Check that change has occurred
option_value <- gigs_option_get("handle_undefined_data", silent = TRUE)
print(option_value)

# Suppress printed output with `silent = TRUE`
gigs_option_set("handle_undefined_data", "quiet", silent = TRUE)

# Set all GIGS options for input checking
gigs_input_options_set("warn", silent = TRUE)
```

---

gigs_waz                        *Calculate z-scores for anthropometric measures according to GIGS*
                                *guidance*

---

**Description**

These functions calculate z-scores for weight-for-age (WAZs), length/height-for-age (LHAZs), weight-for-length/height (WLZs), or head circumference-for-age (HCAZs).

The growth standard used for each observation differs based on the gestational and post-menstrual age of the infant being analysed. The different procedures are based on the advised use of each growth standard, where a measurement is considered a 'birth measurement' if it is the first measurement taken from an infant and was taken <3 days after birth:

- `gest_days` < 24 weeks (168 days):
    - Birth: No standard available
    - Postnatal: IG-21st Postnatal Growth standards from 27 to 64 weeks' PMA
    - Postnatal: Uncorrected WHO standard after 64 weeks' PMA
- `gest_days` >= 24 weeks (168 days):

- **–** Birth: IG-21st Newborn Size standards at birth (including Very Preterm Newborn Size standards if born before 33 weeks).
- **–** Postnatal: IG-21st Postnatal Growth standards from 27 to 64 weeks' PMA
- **–** Postnatal: Uncorrected WHO Child Growth standards after 64 weeks' PMA

- gest_days >= 37 weeks (259 days):
  - **–** Birth: IG-21st Newborn Size standards
  - **–** Postnatal: Uncorrected WHO Child Growth standards

- gest_days >= 43 weeks (301 days):
  - **–** Birth: No standard available
  - **–** Postnatal: Uncorrected WHO Child Growth standards

For gigs_wlz(), two WHO standards are available:

- The weight-for-length standard, which is applied when age_days < 731.
- The weight-for-height standard, which is applied when age_days >= 731.

## Usage

```
gigs_waz(weight_kg, age_days, gest_days, sex, id = NULL)

gigs_lhaz(lenht_cm, age_days, gest_days, sex, id = NULL)

gigs_wlz(weight_kg, lenht_cm, age_days, gest_days, sex, id = NULL)

gigs_hcaz(headcirc_cm, age_days, gest_days, sex, id = NULL)
```

## Arguments

| | |
|---|---|
| weight_kg | A numeric vector of length one or more with weight in kg. |
| age_days | A numeric vector of length one or more with ages in days. |
| gest_days | A numeric vector of length one or more with gestational ages in days. |
| sex | Character vector of length one or more with sex(es), either "M" (male) or "F" (female). This argument is case-sensitive. By default, gigs will replace elements of sex which are not "M" or "F" with NA and warn you. You can customise this behaviour using the GIGS package-level options. |
| id | A factor of length one or more with an ID for each observation, either ordered or unordered, containing no missing (NA) values. When supplied, id is used to ensure that only the earliest measurement for each individual is used as a birth measure. Leave this argument as NULL if all your data comes from the same individual. Default = NULL. |
| lenht_cm | A numeric vector of length one or more with length or height in cm. Should be recumbent length when age_days < 731, and standing height when age_days >= 731. |
| headcirc_cm | Numeric vector of length one or more with head circumference in cm. |

## Value

A numeric vector of z-scores, derived using the appropriate growth standard for each element-wise combination of `gest_days` and `age_days`. For gigs_wlz(), all birth WLZs will be missing (NA), as there is no INTERGROWTH-21st Newborn standard for weight-for-length.

## Note

These functions expect vectors which are recyclable with `vctrs::vec_recycle_common()`.

## Examples

```
gest_days <- c(rep(35 * 7, 3), rep(35 * 7, 3))
age_days <- c(0, 100, 500, 2, 100, 500)
sex <- rep.int(c("M", "F"), c(3, 3))
wt_kg <- c(3, 6, 9, 3, 6, 9)
len_cm <- rep.int(c(52.2, 60.4, 75), 2)
head_cm <- rep.int(c(30, 40, 49), 2)
ids <- factor(rep.int(c("A", "B"), c(3, 3)))

# Weight-for-age z-score (WAZ)
waz <- gigs_waz(weight_kg = wt_kg,
                age_days = age_days,
                gest_days = gest_days,
                sex = sex,
                id = ids)
print(waz)

# Note - if you don't specify 'id' you'll get different results!
waz_no_id <- gigs_waz(weight_kg = wt_kg,
                      age_days = age_days,
                      gest_days = gest_days,
                      sex = sex)
print(waz == waz_no_id)

# Length/height-for-age z-score (LHAZ)
lhaz <- gigs_lhaz(lenht_cm = len_cm,
                  age_days = age_days,
                  gest_days = gest_days,
                  sex = sex,
                  id = ids)
print(lhaz)

# Weight-for-length/height z-score (WLZ)
#   Note: There's no at-birth standards for weight-for-length, so elements 1
#   and 4 have `NA` values
wlz <- gigs_wlz(weight_kg = wt_kg,
                lenht_cm = len_cm,
                age_days = age_days,
                gest_days = gest_days,
                sex = sex,
                id = ids)
print(wlz)
```

```
# Head circumference-for-age z-score (HCAZ)
hcaz <- gigs_hcaz(headcirc_cm = head_cm,
                  age_days = age_days,
                  gest_days = gest_days,
                  sex = sex,
                  id = ids)
print(hcaz)
```

---

ig_fet                           *INTERGROWTH-21st Fetal Standards growth curve data*

---

## Description

A set of nested lists containing tables with reference values at different z-scores/centiles for valid gestational ages in days. The list is ordered by acronym first, then by z-score/centile (as available - some Fetal standards only have centile tables).

## Note

Where possible, tables were taken from the online sources here. If not available on the INTERGROWTH-21st website, they were taken from the publications listed here.

## Source

[INTERGROWTH-21st Pregnancy Dating (CRL)](#)

[INTERGROWTH-21st Symphysis-Fundal Height standard](#)

[INTERGROWTH-21st Fetal Growth standards](#)

[INTERGROWTH-21st Fetal Doppler standards](#)

[INTERGROWTH-21st Gestational Weight Gain standard](#)

## References

Papageorghiou AT, Ohuma EO, Altman DG, Todros T, Cheikh Ismail L, Lambert A et al. **International standards for fetal growth based on serial ultrasound measurements: the Fetal Growth Longitudinal Study of the INTERGROWTH-21st Project.** *Lancet* 2014, **384(9946):869-79.** [doi:10.1016/S01406736(14)614902](#)

Stirnemann J, Villar J, Salomon LJ, Ohuma EO, Lamber A, Victoria CG et al. **International Estimated Fetal Weight Standards of the INTERGROWTH-21st Project.** *Ultrasound Obstet Gynecol* 2016, **49:478-486** [doi:10.1002/uog.17347](#)

Stirnemann J, Salomon LJ, Papageorghiou AT. **INTERGROWTH-21st standards for Hadlock's estimation of fetal weight.** *Ultrasound Obstet Gynecol* 2020, **56(6):946-948** [doi:10.1002/uog.22000](#)

Papageorghiou AT, Ohuma EO, Gravett MG, Lamber A, Noble JA, Pang R et al. **International standards for symphysis-fundal height based on serial measurements from the Fetal Growth Longitudinal Study of the INTERGROWTH-21st Project: prospective cohort study in eight countries.** *BMJ* 2016, **355:i5662** [doi:10.1136/bmj.i5662](#)

Papageorghiou AT, Kennedy SH, Salomon LJ, Ohuma EO, Cheikh Ismail L, Barros FC et al. **International standards for early fetal size and pregnancy dating based on ultrasound measurement of crown-rump length in the first trimester of pregnancy.** *Ultrasound Obstet Gynecol* 2014, **44(6):641-48** doi:10.1002/uog.13448

Cheikh Ismail L, Bishop DC, Pang R, Ohuma EO, Kac G, Abrams B et al. **Gestational weight gain standards based on women enrolled in the Fetal Growth Longitudinal Study of the INTERGROWTH-21st Project: a prospective longitudinal cohort study.** *BMJ* 2016, **352:i555** doi:10.1136/bmj.i555

Drukker L, Staines-Urias E, Villar J, Barros FC, Carvalho M, Munim S et al. **International gestational age-specific centiles for umbilical artery Doppler indices: a longitudinal prospective cohort study of the INTERGROWTH-21st Project.** *Am J Obstet Gynecol* 2021, **222(6):602.e1-602.e15** doi:10.1016/j.ajog.2020.01.012

Rodriguez-Sibaja MJ, Villar J, Ohuma EO, Napolitano R, Heyl S, Carvalho M et al. **Fetal cerebellar growth and Sylvian fissure maturation: international standards from Fetal Growth Longitudinal Study of INTERGROWTH-21st Project** *Ultrasound Obstet Gynecol* 2021, **57(4):614-623** doi:10.1002/uog.22017

Napolitano R, Molloholli M, Donadono V, Ohuma EO, Wanyonyi SZ, Kemp B et al. **International standards for fetal brain structures based on serial ultrasound measurements from Fetal Growth Longitudinal Study of INTERGROWTH-21st Project** *Ultrasound Obstet Gynecol* 2020, **56(3):359-370** doi:10.1002/uog.21990

## Examples

```
names(gigs::ig_fet)
head(gigs::ig_fet$hcfga$zscores)
```

---

ig_fet_estimate_fetal_weight

*Estimate fetal weight in grams using the INTERGROWTH-21st predictive equation*

---

## Description

Estimate fetal weight in grams using the INTERGROWTH-21st predictive equation

## Usage

```
ig_fet_estimate_fetal_weight(abdocirc_mm, headcirc_mm)
```

## Arguments

abdocirc_mm    Numeric vector with abdominal circumference value(s) in mm. Should have length one or same length as headcirc_mm.

headcirc_mm    Numeric vector with head circumference value(s) in mm. Should have length one or same length as abdocirc_mm.

## Value

Numeric vector with estimated fetal weight(s) in g, with the same length as the longest input vector.

## Note

Inputs are recycled using `vctrs::vec_recycle_common()`.

## References

Stirnemann J, Villar J, Salomon LJ, Ohuma EO, Lamber A, Victoria CG et al. **International Estimated Fetal Weight Standards of the INTERGROWTH-21st Project.** *Ultrasound Obstet Gynecol* 2016, **49:478-486** doi:10.1002/uog.17347

## Examples

```
# Estimate fetal weight in grams
ig_fet_estimate_fetal_weight(abdocirc_mm = 31:33,
                             headcirc_mm = 25:27)

# Input vectors are recycled using vctrs::vec_recycle_common
ig_fet_estimate_fetal_weight(abdocirc_mm = 25.0,
                             headcirc_mm = 24:26)
```

---

| ig_fet_estimate_ga | *Estimate gestational age using INTERGROWTH-21st predictive equations* |
|---|---|

---

## Description

Estimate gestational age using INTERGROWTH-21st predictive equations

## Usage

```
ig_fet_estimate_ga(crl_mm = NULL, headcirc_mm = NULL, femurlen_mm = NULL)
```

## Arguments

crl_mm
: Numeric vector with crown-rump length value(s) in mm taken from an ultrasound scan obtained 56 to 97 days since the last menstrual period. If not NULL (the default), should be of length one or the same length as other inputs. The function will return NA for all elements of `crl_mm` which are not between 15 and 95 mm and have no `headcirc_mm` or `abdocirc_mm` values.

headcirc_mm
: Numeric vector with head circumference value(s) in cm taken from an ultrasound scan obtained 98 to 181 days since the last menstrual period. If not NULL (the default), should have length one or same length as other inputs.

femurlen_mm
: Numeric vector with femur length value(s) in cm. If not NULL (the default), should have length one or same length as other arguments inputs.

## Value

A numeric vector with estimated gestational ages in days. This vector will have a length equal to the outputs from `vctrs::vec_recycle_common()`.

## Note

Vector inputs to this function are recycled using `vctrs::vec_recycle_common()`. The function will attempt to estimate the gestational age (GA) in three steps, from highest to lowest accuracy. First, using crown-rump length measurements obtained in early pregnancy, then with both head circumference and femur length measurements, and finally with head circumference alone.

## References

Papageorghiou AT, Kennedy SH, Salomon LJ, Ohuma EO, Cheikh Ismail L, Barros FC et al. **International standards for early fetal size and pregnancy dating based on ultrasound measurement of crown-rump length in the first trimester of pregnancy.** *Ultrasound Obstet Gynecol* 2014, **44(6):641-48** doi:10.1002/uog.13448

Papageorghiou AT, Kemp B, Stones W, Ohuma EO, Kennedy SH, Purwar M et al. **Ultrasound-based gestational age estimation in late pregnancy.** *Ultrasound Obstet Gynecol* 2016. **48(6):719-26** doi:10.1002/uog.15894

## Examples

```
# Estimate gestational age in days using crown-rump length (most accurate)
ig_fet_estimate_ga(crl_mm = 40:45)

# Estimate gestational age in days using head circumference and femur length
# (second-most accurate)
ig_fet_estimate_ga(headcirc_mm = 250:255, femurlen_mm = 55:60)

# Estimate gestational age in days using head circumference only (least
# accurate)
ig_fet_estimate_ga(headcirc_mm = 250:255)

# The function defaults to CRL if available
ig_fet_estimate_ga(crl_mm = 40:45,
                   headcirc_mm = 250:255,
                   femurlen_mm = 55:60)

# Inputs are recycled using [vctrs::vec_recycle_common]
ig_fet_estimate_ga(headcirc_mm = 252,
                   femurlen_mm = 55:60)
```

---

ig_nbs                          *INTERGROWTH-21st Newborn Size Standards (including very preterm) growth curve data*

---

## Description

A set of nested lists containing tables with reference values at different z-scores/centiles for valid gestational ages in days. The list is ordered by acronym first, then by sex and finally by z-score/centile.

## Note

The tables in this package are combined versions of the tables published by Villar *et al.* (2014) and Villar *et al.* (2016), so they cover 168 to 300 days' gestational age. The body composition tables (`ffmfga`, `bfpfga`, and `fmfga`) cover a smaller gestational age span, ranging from only 266 to 294 days' (38 to 42 weeks') gestational age.

## Source

INTERGROWTH-21st Newborn Size in Very Preterm Infants
INTERGROWTH-21st Newborn Size Standards
INTERGROWTH-21st Newborn Size Standards - Body Composition

## References

Villar J, Cheikh Ismail L, Victora CG, Ohuma EO, Bertino E, Altman DG, et al. **International standards for newborn weight, length, and head circumference by gestational age and sex: the Newborn Cross-Sectional Study of the INTERGROWTH-21st Project.** *Lancet* 2014, **384(9946):857-68.** doi:10.1016/S01406736(14)609326

Villar J, Giuliani F, Fenton TR, Ohuma EO, Ismail LC, Kennedy SH et al. **INTERGROWTH-21st very preterm size at birth reference charts.** *Lancet* 2016, **387(10021):844-45.** doi:10.1016/S01406736(16)003846

Villar J, Puglia FA, Fenton TR, Ismal LC, Staines-Urias E, Giuliani F, et al. **Body composition at birth and its relationship with neonatal anthropometric ratios: the newborn body composition study of the INTERGROWTH-21st project.** *Pediatric Research* 2017, **82:305-316.** doi:10.1038/pr.2017.52

## Examples

```
names(gigs::ig_nbs)
head(gigs::ig_nbs$wfga$male$zscores)
```

---

ig_nbs_coeffs *INTERGROWTH-21st Newborn Size Standards GAMLSS coefficients*

---

## Description

A set of nested lists containing mu, sigma, nu and tau values across gestational ages for either sex, for the INTERGROWTH-21st weight/length/head circumference-for-gestational age standards. The lists are ordered by acronym, then sex.

**Source**

Mu/sigma/nu/tau values were provided by Dr Eric Ohuma.

**References**

Villar J, Cheikh Ismail L, Victora CG, Ohuma EO, Bertino E, Altman DG, et al. **International standards for newborn weight, length, and head circumference by gestational age and sex: the Newborn Cross-Sectional Study of the INTERGROWTH-21st Project.** *Lancet* 2014, **384(9946):857-68.** doi:10.1016/S01406736(14)609326

Villar J, Giuliani F, Fenton TR, Ohuma EO, Ismail LC, Kennedy SH et al. **INTERGROWTH-21st very preterm size at birth reference charts.** *Lancet* 2016, **387(10021):844-45.** doi:10.1016/S01406736(16)003846

Villar J, Puglia FA, Fenton TR, Ismal LC, Staines-Urias E, Giuliani F, et al. **Body composition at birth and its relationship with neonatal anthropometric ratios: the newborn body composition study of the INTERGROWTH-21st project.** *Pediatric Research* 2017, **82:305-316.** doi:10.1038/pr.2017.52

**Examples**

```
names(gigs::ig_nbs_coeffs)
head(gigs::ig_nbs_coeffs$wfga$male)
```

---

| ig_nbs_ext | *Extended INTERGROWTH-21st Newborn Size Standards (including very preterm) growth curve data* |
|---|---|

---

**Description**

A set of nested lists containing tables with reference values at different z-scores/centiles for valid gestational ages in days. The list is ordered by acronym first, then by sex and finally by z-score/centile.

**Note**

The tables in this package are combined versions of the tables published by Villar *et al.* (2014) and Villar *et al.* (2016), so they cover 168 to 300 days' gestational age. They have been extrapolated by Simon Parker (with guidance from Eric Ohuma) to include a wider range of gestational ages, and should be applied carefully by researchers. More detail on the extrapolation process is available in an article on the package website.

**Source**

INTERGROWTH-21st Newborn Size in Very Preterm Infants
INTERGROWTH-21st Newborn Size Standards
INTERGROWTH-21st Newborn Size Standards - Body Composition

### References

Villar J, Cheikh Ismail L, Victora CG, Ohuma EO, Bertino E, Altman DG, et al. **International standards for newborn weight, length, and head circumference by gestational age and sex: the Newborn Cross-Sectional Study of the INTERGROWTH-21st Project.** *Lancet* 2014, **384(9946):857-68.** doi:10.1016/S01406736(14)609326

Villar J, Giuliani F, Fenton TR, Ohuma EO, Ismail LC, Kennedy SH et al. **INTERGROWTH-21st very preterm size at birth reference charts.** *Lancet* 2016, **387(10021):844-45.** doi:10.1016/S01406736(16)003846

Villar J, Puglia FA, Fenton TR, Ismal LC, Staines-Urias E, Giuliani F, et al. **Body composition at birth and its relationship with neonatal anthropometric ratios: the newborn body composition study of the INTERGROWTH-21st project.** *Pediatric Research* 2017, **82:305-316.** doi:10.1038/pr.2017.52

### Examples

```
names(gigs::ig_nbs_ext)
head(gigs::ig_nbs_ext$wfga$male$zscores)
```

---

ig_nbs_ext_coeffs      *Extended INTERGROWTH-21st Newborn Size Standards GAMLSS coefficients*

---

### Description

A set of nested lists containing mu, sigma, nu and tau values across gestational ages for either sex, for the extended INTERGROWTH-21st weight/length/head circumference-for-gestational age standards in newborns. The lists are ordered by acronym, then sex.

### Note

The mu and sigma coefficients have been extrapolated by Simon Parker (with guidance from Eric Ohuma) to include a wider range of gestational ages, and should be applied carefully by researchers. More detail on the extrapolation process is available in an article on the package website.

### Source

Mu/sigma/nu/tau values were provided by Dr Eric Ohuma.

### References

Villar J, Cheikh Ismail L, Victora CG, Ohuma EO, Bertino E, Altman DG, et al. **International standards for newborn weight, length, and head circumference by gestational age and sex: the Newborn Cross-Sectional Study of the INTERGROWTH-21st Project.** *Lancet* 2014, **384(9946):857-68.** doi:10.1016/S01406736(14)609326

Villar J, Giuliani F, Fenton TR, Ohuma EO, Ismail LC, Kennedy SH et al. **INTERGROWTH-21st very preterm size at birth reference charts.** *Lancet* 2016, **387(10021):844-45.** doi:10.1016/S01406736(16)003846

Villar J, Puglia FA, Fenton TR, Ismal LC, Staines-Urias E, Giuliani F, et al. **Body composition at birth and its relationship with neonatal anthropometric ratios: the newborn body composition study of the INTERGROWTH-21st project.** *Pediatric Research* 2017, **82:305-316.** doi:10.1038/pr.2017.52

## Examples

```
names(gigs::ig_nbs_ext_coeffs)
head(gigs::ig_nbs_ext_coeffs$wfga$male)
```

---

ig_png                          *INTERGROWTH-21st Postnatal Growth Standards growth curve data*

---

## Description

A set of nested lists containing tables with reference values at different z-scores/centiles for valid post-menstrual ages. The list is ordered by acronym first, then by sex, and finally by z-score/centile.

## Source

INTERGROWTH-21st Postnatal Growth of Preterm Infants

## References

Villar J, Giuliani F, Bhutta ZA, Bertino E, Ohuma EO, Ismail LC et al. **Postnatal growth standards for preterm infants: the Preterm Postnatal Follow-up Study of the INTERGROWTH-21st Project.** *Lancet Glob Health* 2015, *3(11):e681-e691.* doi:10.1016/S2214109X(15)001631

## Examples

```
names(gigs::ig_png)
head(gigs::ig_png$wfa$male$zscores)
```

---

life6mo                         *Data extract from the Low birthweight Infant Feeding Exploration (LIFE) study*

---

## Description

A subset of anthropometric data for 300 singleton infants enrolled in the Low birthweight Infant Feeding Exploration (LIFE) study. The variables are as follows:

## Format

A data frame with 2,191 rows and 10 variables:

id: Unique ID for each infant in the dataset (1–300).

visitweek: Chronological age in weeks (±1) when study visit occurred (0–26).

sex: Sex of the infant as a factor (″M″ = Male; ″F″ = Female).

gestage: Best obstetric estimate of gestational age in days (181–291).

age_days: Chronological age in days at each visit; equal to pma - gestage (0–242).

pma: Post-menstrual age in days (182–528).

wt_kg: Mean weight in kg (1.24–9.40667).

len_cm: Mean length in cm (37.37–72.93).

headcirc_cm: Mean head circumference in cm (23.20–44.87).

muac_cm: Mean mid-upper arm circumference in cm (6.30–16.83).

## Note

We subsetted the full LIFE 6 month dataset for gigs. As such, this extract only includes data from 300 singleton pregnancies where the best estimate of gestational age was >168 days. We also removed rows corresponding to visit weeks where no measurement data was taken due to non-attendance of the visit.

## References

Vesel L, Bellad RM, Manji K, Saidi F, Velasquez E, Sudfeld C, et al. **Feeding practices and growth patterns of moderately low birthweight infants in resource-limited settings: results from a multisite, longitudinal observational study.** *BMJ Open* 2023, **13(2):e067316.** doi:10.1136/BMJOPEN2022067316

## Examples

```
head(gigs::life6mo)
```

---

| report_units | *Check required units for a conversion using gigs* |
|---|---|

---

## Description

Check required units for a conversion using gigs

## Usage

```
report_units(family, acronym)
```

**Arguments**

family               A single string denoting the family of growth standards in use. Must be one
of `"ig_fet"`, `"ig_nbs"`, `"ig_png"`, or `"who_gs"`, or the function will throw an
error. This argument is case-sensitive.

acronym           A single string denoting the specific growth standard in use. The allowed values
of acronym depend on the value of family, and are listed in the documentation
for the x argument. Incompatible family and acronym values will cause errors
to be thrown. This argument is case-sensitive.

**Value**

Returns a message describing which units are required for conversion in the desired growth standard.

**Examples**

```
# Get units for working with the IG-21st Newborn Size Standard for
# weight-for-GA
report_units("ig_nbs", "wfga")

# Get units for working with the IG-21st Fetal Standard for
# gestational weight gain
report_units("ig_fet", "gwgfga")
```

---

value2zscore             *Convert anthropometric measures to z-scores/centiles using interna-*
*tional growth standards*

---

**Description**

Convert anthropometric measures to z-scores/centiles using international growth standards

**Usage**

```
value2zscore(y, x, sex = NULL, family, acronym)

value2centile(y, x, sex = NULL, family, acronym)
```

**Arguments**

y                  Numeric vector of length one or more with anthropometric measurement(s) to
convert to centiles/z-scores. Units depend on which combination of family and
acronym are in use, which you can check with `report_units()`. By default,
GIGS will warn you if elements in y are missing (NA) or undefined (NaN/Inf),
and replace the latter with NA. You can customise this behaviour using the GIGS
package-level options.

x   Numeric vector of x values, which take on different units depending on the growth standard in use. To know which units to use, run `report_units(family, acronym)`. Limits for each combination are as follows:

- WHO Growth Standards (family = "who_gs")
  - Between 0 and 1856 days for "wfa", "bfa", "lhfa", and "hcfa".
  - Between 45 and 110 cm for "wfl".
  - Between 65 and 120 cm for "wfh".
  - Between 91 and 1856 days for "acfa", "ssfa", "tsfa".
- INTERGROWTH-21st Fetal Standards (family = "ig_fet")
  - Between 98 and 280 days for "hcfga", "bpdfga", "acfga", "flfga", "ofdfga", and "tcdfga".
  - Between 154 and 280 days for "efwfga".
  - Between 112 and 294 days for "sfhfga".
  - Between 58 and 105 days for "crlfga".
  - Between 19 and 95 mm for "gafcrl".
  - Between 105 and 280 days for "gwgfga".
  - Between 168 and 280 days for "pifga", "rifga", and "sdrfga" (the INTERGROWTH-21st Fetal Doppler standards).
  - Between 12 and 55 mm for "tcdfga".
  - Between 105 and 252 days for "poffga", "sffga", "avfga", "pvfga", and "cmfga" (the INTERGROWTH-21st Fetal Brain Development standards).
  - Between 126 and 287 days for "hefwfga" (the INTERGROWTH-21st standard for Hadlock-based estimated fetal weight).
- INTERGROWTH-21st Newborn Size Standards (family = "ig_nbs")
  - Between 168 and 300 days for "wfga", "lfga", "hcfga", and "wlrfga".
  - Between 266 and 280 days for "fmfga", "bfpfga", and "ffmfga".
- Extended INTERGROWTH-21st Newborn Size Standards (family = "ig_nbs_ext")
  - Between 154 and 314 days for "wfga", "lfga", and "hcfga".
- INTERGROWTH-21st Postnatal Growth Standards (family = "ig_png")
  - Between 27 and 64 weeks for "wfa", "lfa", and "hcfa".
  - Between 35 and 65 cm for "wfl".

By default, gigs will replace out-of-bounds elements in x with NA and warn you. It will also warn you about elements in x which are missing (NA) or undefined (NaN/Inf). You can customise this behaviour using the [GIGS package-level options](#).

sex   Character vector of length one or more with sex(es), either "M" (male) or "F" (female). This argument should be left as NULL (the default) if family == "ig_fet". This argument is case-sensitive. By default, gigs will replace elements of sex which are not "M" or "F" with NA and warn you. You can customise this behaviour using the [GIGS package-level options](#).

family   A single string denoting the family of growth standards in use. Must be one of "ig_fet", "ig_nbs", "ig_png", or "who_gs", or the function will throw an error. This argument is case-sensitive.

acronym            A single string denoting the specific growth standard in use. The allowed values
                   of `acronym` depend on the value of `family`, and are listed in the documentation
                   for the x argument. Incompatible `family` and `acronym` values will cause errors
                   to be thrown. This argument is case-sensitive.

## Value

Numeric vector of z-scores/centiles with length equal to the longest input vector.

## Note

Inputs other than `acronym` and `family` will be recycled by [vctrs::vec_recycle_common()](), and
must adhere to the [vctrs]() recycling rules.

## References

- For the INTERGROWTH-21st Fetal standards see the [gigs::ig_fet]() documentation
- For the INTERGROWTH-21st Newborn Size standards see the [gigs::ig_nbs]() documentation
- For the INTERGROWTH-21st Postnatal Growth standards see the [gigs::ig_png]() documentation
- For the WHO Growth standards see the [gigs::who_gs]() documentation

## See Also

[report_units()]() will tell you the units needed for y and x, for any combination of `family` and
`acronym`.

## Examples

```
# Convert in the weight-for-age standard from the WHO Child Growth Standards
value2zscore(3, 0, "M", family = "who_gs", acronym = "wfa")

# Or obtain centiles...
value2centile(3, 0, "M", family = "who_gs", acronym = "wfa")

# Inputs will be recycled if necessary
value2centile(2.5, 280, c("M", "F"), family = "ig_nbs", acronym = "wfga")
```

---

who_gs                      *WHO Child Growth Standards growth curve data*

---

## Description

A set of nested lists containing tables with reference values at different z-scores/centiles for valid x
values (usually age in days, also length or height in cm for weight-for-length (`wfl`) and weight-for-
height (`wfh`) standards, respectively). The list is ordered by acronym first, then by sex and finally
by z-score/centile.

## Source

## References

de Onis M, Garza C, Victora CG, Onyango AW, Frongillo EA, Martines J. **The WHO Multicentre Growth Reference Study: planning, study design, and methodology** *Food Nutr Bull.* 2004, **25(1 Suppl):S15-26.** [doi:10.1177/15648265040251s104](#)

World Health Organisation. **WHO child growth standards: length/height-for-age, weight-for-age, weight-for-length, weight-for-height and body mass index-for-age: methods and development.** *Technical report, WHO, Geneva*, 2006.

World Health Organisation. **WHO child growth standards: head circumference-for-age, arm circumference-for-age, triceps skinfold-for-age and subscapular skinfold-for-age: methods and development.** *Technical report, WHO, Geneva*, 2007.

## Examples

```
names(gigs::who_gs)
head(gigs::who_gs$wfa$male$zscores)
```

---

| who_gs_coeffs | *WHO Child Growth Standards LMS coefficients* |
| --- | --- |

---

## Description

A set of nested lists containing tables with LMS values for each sex in different combinations of age/length-height/BMI. The list is ordered by acronym first, then by sex.

## Source

## References

World Health Organisation. **WHO child growth standards: length/height-for-age, weight-for-age, weight-for-length, weight-for-height and body mass index-for-age: methods and development.** *Technical report, WHO, Geneva*, 2006.

World Health Organisation. **WHO child growth standards: head circumference-for-age, arm circumference-for-age, triceps skinfold-for-age and subscapular skinfold-for-age: methods and development.** *Technical report, WHO, Geneva*, 2007.

## Examples

```
names(gigs::who_gs_coeffs)
head(gigs::who_gs_coeffs$lhfa$male)
```

---

| zscore2value | *Convert z-scores/centiles to anthropometric measures using international growth standards* |
|---|---|

---

### Description

Convert z-scores/centiles to anthropometric measures using international growth standards

### Usage

```
zscore2value(z, x, sex = NULL, family, acronym)

centile2value(p, x, sex = NULL, family, acronym)
```

### Arguments

z, p          Numeric vector of length one or more with centiles/z-scores to convert to values. For p, gigs will warn you if elements of p are not between 0 and 1. You can customise this behaviour using the GIGS package-level options.

x             Numeric vector of x values, which take on different units depending on the growth standard in use. To know which units to use, run `report_units(family, acronym)`. Limits for each combination are as follows:

- WHO Growth Standards (family = "who_gs")
  - Between 0 and 1856 days for "wfa", "bfa", "lhfa", and "hcfa".
  - Between 45 and 110 cm for "wfl".
  - Between 65 and 120 cm for "wfh".
  - Between 91 and 1856 days for "acfa", "ssfa", "tsfa".
- INTERGROWTH-21st Fetal Standards (family = "ig_fet")
  - Between 98 and 280 days for "hcfga", "bpdfga", "acfga", "flfga", "ofdfga", and "tcdfga".
  - Between 154 and 280 days for "efwfga".
  - Between 112 and 294 days for "sfhfga".
  - Between 58 and 105 days for "crlfga".
  - Between 19 and 95 mm for "gafcrl".
  - Between 105 and 280 days for "gwgfga".
  - Between 168 and 280 days for "pifga", "rifga", and "sdrfga" (the INTERGROWTH-21st Fetal Doppler standards).
  - Between 12 and 55 mm for "tcdfga".
  - Between 105 and 252 days for "poffga", "sffga", "avfga", "pvfga", and "cmfga" (the INTERGROWTH-21st Fetal Brain Development standards).
  - Between 126 and 287 days for "hefwfga" (the INTERGROWTH-21st standard for Hadlock-based estimated fetal weight).
- INTERGROWTH-21st Newborn Size Standards (family = "ig_nbs")

> – Between 168 and 300 days for ″wfga″, ″lfga″, ″hcfga″, and ″wlrfga″.
>
> – Between 266 and 280 days for ″fmfga″, ″bpfpfga″, and ″ffmfga″.
>
> • Extended INTERGROWTH-21st Newborn Size Standards (family = ″ig_nbs_ext″)
>
> – Between 154 and 314 days for ″wfga″, ″lfga″, and ″hcfga″.
>
> • INTERGROWTH-21st Postnatal Growth Standards (family = ″ig_png″)
>
> – Between 27 and 64 weeks for ″wfa″, ″lfa″, and ″hcfa″.
>
> – Between 35 and 65 cm for ″wfl″.

By default, gigs will replace out-of-bounds elements in x with NA and warn you. It will also warn you about elements in x which are missing (NA) or undefined (NaN/Inf). You can customise this behaviour using the GIGS package-level options.

sex | Character vector of length one or more with sex(es), either ″M″ (male) or ″F″ (female). This argument should be left as NULL (the default) if family == ″ig_fet″. This argument is case-sensitive. By default, gigs will replace elements of sex which are not ″M″ or ″F″ with NA and warn you. You can customise this behaviour using the GIGS package-level options.

family | A single string denoting the family of growth standards in use. Must be one of ″ig_fet″, ″ig_nbs″, ″ig_png″, or ″who_gs″, or the function will throw an error. This argument is case-sensitive.

acronym | A single string denoting the specific growth standard in use. The allowed values of acronym depend on the value of family, and are listed in the documentation for the x argument. Incompatible family and acronym values will cause errors to be thrown. This argument is case-sensitive.

## Value

Numeric vector of expected measurements with length equal to the longest input vector.

## References

- For the INTERGROWTH-21st Fetal standards see the `gigs::ig_fet` documentation
- For the INTERGROWTH-21st Newborn Size standards see the `gigs::ig_nbs` documentation
- For the INTERGROWTH-21st Postnatal Growth standards see the `gigs::ig_png` documentation
- For the WHO Growth standards see the `gigs::who_gs` documentation

## Examples

```
# Convert in the weight-for-age standard from the WHO Child Growth Standards
zscore2value(-2:2, 0, ″M″, family = ″who_gs″, acronym = ″wfa″)

# Or obtain centiles...
centile2value(pnorm(-2:2), 0, ″M″, family = ″who_gs″, acronym = ″wfa″)

# Inputs will be recycled if necessary
centile2value(3, 280, c(″M″, ″F″), family = ″ig_nbs″, acronym = ″wfga″)
```

# Index