

Package: PhysioReport (via r-universe)

October 3, 2026

Type Package

Title Clinical Report Generation for the x-biosignal Ecosystem

Version 0.1.2

Description Parameterized, bilingual (Japanese/English), colorblind-safe clinical reports (gait, HRV, EMG) with normative overlays and MDC/MCID-annotated change for the x-biosignal physiological signal ecosystem.

Depends R (>= 4.2)

Imports PhysioExperiment (>= 2.0.0), ggplot2

Suggests PhysioCore, testthat (>= 3.2.0), patchwork, PhysioMoCap, PhysioGaitNorm, vdiff, quarto, officer, flextable, rvg, knitr, rmarkdown, PhysioMSKNet, PhysioClinStats

VignetteBuilder knitr

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

URL <https://github.com/x-biosignal/PhysioReport>,
<https://x-biosignal.github.io/PhysioReport/>

BugReports <https://github.com/x-biosignal/PhysioExperiment/issues>

Config/testthat/edition 3

Config/pak/sysreqs zlib1g-dev

Repository <https://x-biosignal.r-universe.dev>

Date/Publication 2026-10-03 13:11:47 UTC

RemoteUrl <https://github.com/x-biosignal/PhysioReport>

RemoteRef HEAD

RemoteSha 52c5b4bd12755189447b536590ff863a67577c1c

Contents

annotateChange	2
as.timelineData	3
buildDocxReport	4
clinicalGaitReport	5
gaitIndexDashboard	6
longitudinalTimeline	7
NormativeModel	8
normativeZScore	10
physioLabel	10
plotBlandAltman	11
plotChangeAnnotated	12
plotNormativeBand	12
plotOutcomeRadar	14
plotSCED	15
renderClinicalReport	16
reportTheme	17
scedStats	17
Index	19

annotateChange	<i>Classify pre-to-post change against MDC and MCID thresholds</i>
----------------	--

Description

Annotates a set of paired (pre, post) outcome scores against their Minimal Detectable Change (MDC) and Minimal Clinically Important Difference (MCID) thresholds, classifying each change as "no change" (within measurement error), "detectable (>MDC)" (a real change that clears measurement error) or "clinically meaningful (>MCID)". Thresholds are compared to the absolute change and are boundary-inclusive (a change exactly equal to the MDC is "detectable"). The direction of benefit is tracked separately so decrease-is-good metrics (pain, timed tests) are handled correctly.

Usage

```
annotateChange(pre, post, mdc, mcid, direction = "increase")
```

Arguments

pre, post	Numeric vectors of pre- and post-intervention scores (one element per metric); may be named to label the metrics.
mdc, mcid	Numeric MDC / MCID thresholds (positive), a scalar recycled across metrics or a vector matching pre. Obtain the MDC from <code>PhysioExperiment::mdc()</code> (or <code>PhysioMoCap::mdc()</code>); the MCID is the instrument's anchor-based value.
direction	Benefit direction, "increase" (higher is better, default) or "decrease" (lower is better); a scalar applied to all metrics or a vector matching pre.

Value

A data.frame of class "change_annotation" with columns metric, pre, post, change (signed post - pre), improvement (signed so positive = beneficial), mdc, mcid, exceeds_mdc, exceeds_mcid, improved, direction and classification. A metric with a missing pre or post yields NA change and classification: missingness is propagated rather than silently reported as "no change".

References

de Vet HCW et al. (2006). Minimally important change determined by a visual method. *Qual Life Res*; Shrout & Fleiss (1979) for the SEM underlying MDC.

See Also

[plotChangeAnnotated\(\)](#), `PhysioExperiment::mdc()`

Examples

```
annotateChange(pre = c(fma = 20, pain = 7),
               post = c(fma = 31, pain = 4),
               mdc = c(5.2, 1.0), mcid = c(9, 2),
               direction = c("increase", "decrease"))
```

as.timelineData

Adapt a longitudinal tracker to timeline data

Description

Converts a longitudinal-tracking result into the sessions / outcomes data frames consumed by [longitudinalTimeline](#).

Usage

```
as.timelineData(x, ...)

## Default S3 method:
as.timelineData(x, ...)

## S3 method for class 'MSKLongitudinalTracker'
as.timelineData(x, ...)
```

Arguments

x A tracker object (e.g. a MSKLongitudinalTracker from `PhysioMSKNet::mskLongitudinalTracker()`)
... Unused.

Value

A list with sessions and outcomes data frames.

See Also[longitudinalTimeline\(\)](#)**Examples**

```
# A tracker is any object carrying $metrics_table and $timepoint_labels;
# build a minimal one directly so the adapter runs without PhysioMSKNet.
tracker <- structure(
  list(metrics_table = data.frame(
    timepoint = rep(c("wk0", "wk6"), each = 2),
    muscle = rep(c("biceps", "triceps"), 2),
    metric_name = "rms",
    value = c(0.10, 0.08, 0.14, 0.12)),
    timepoint_labels = c("wk0", "wk6")),
  class = "MSKLongitudinalTracker")
td <- as.timelineData(tracker)
td$outcomes
```

buildDocxReport

*Build an editable DOCX clinical report (officer)***Description**

Renders a one-page clinical report to an editable Word document using **officer** (with **flextable** when available) — no LaTeX toolchain required. Called by [renderClinicalReport](#) for format = "docx"; usable directly.

Usage

```
buildDocxReport(
  data,
  template = c("gait", "hrv", "emg"),
  lang = c("en", "ja"),
  out
)
```

Arguments

data	A report payload list (see renderClinicalReport): subject, figures (a named list of ggplots), change (an optional change_annotation) and provenance (an optional named list).
template	One of "gait", "hrv", "emg" (drives the localized title).
lang	"en" or "ja".
out	Output .docx path.

Value

out, invisibly.

See Also[renderClinicalReport\(\)](#)**Examples**

```
# Writes an editable .docx via officer (a Suggests dependency) into tempdir().
if (requireNamespace("officer", quietly = TRUE)) {
  fig <- plotNormativeBand(NULL, NormativeModel(
    mean = sin(seq(0, pi, length.out = 101)), sd = rep(0.1, 101), time = 0:100))
  out <- tempfile(fileext = ".docx")
  buildDocxReport(list(subject = "P01", figures = list(gait_cycle = fig)),
    template = "gait", out = out)
}
```

clinicalGaitReport	<i>Clinical gait report dashboard</i>
--------------------	---------------------------------------

Description

Composes a multi-panel clinical gait report: one waveform panel per gait variable (a colorblind-safe normative $\pm$ SD corridor from [plotNormativeBand](#) overlaid with the subject's Left and Right mean traces and optional event markers), followed by a gait-index dashboard (GDI / GPS / Movement Analysis Profile). Panels are laid out with **patchwork**.

Usage

```
clinicalGaitReport(
  pe,
  norm,
  events = NULL,
  sides = c("L", "R"),
  indices = TRUE
)
```

Arguments

pe	A named list of per-variable side-split cycle waveforms: <code>pe[[variable]]</code> is a list with <code>\$left</code> and <code>\$right</code> , each a numeric cycle waveform or a points-by-cycles matrix (averaged across cycles). Variables are matched to <code>norm\$variables</code> by name; kinetic variables (moments/power) can be included when <code>norm</code> carries their corridors.
norm	A <code>gait_norm</code> -like list (e.g. from <code>PhysioGaitNorm::loadGaitNorm()</code>) with <code>\$variables</code> , row-named <code>\$mean</code> and <code>\$sd</code> matrices (variables x cycle points), and optionally <code>\$percent</code> and <code>\$features</code> (for the GDI).
events	Optional numeric vector of gait-cycle percentages at which to draw event markers (e.g. toe-off) on every waveform panel.
sides	Which sides to overlay (currently informational; both Left and Right are drawn).

indices Logical; if TRUE (default) append the gait-index dashboard panel (requires **PhysioMoCap** and all norm variables present in `pe`); omitted with a warning if it cannot be computed.

Value

A **patchwork** composite (a ggplot-compatible object).

See Also

[gaitIndexDashboard\(\)](#), [plotNormativeBand\(\)](#), [PhysioMoCap::plotSymmetry\(\)](#)

Examples

```
# clinicalGaitReport() composes panels with patchwork (a Suggests dependency);
# indices = FALSE keeps the gait-index panel (and PhysioMoCap) out of the way.
if (requireNamespace("patchwork", quietly = TRUE)) {
  np <- 101
  vars <- c("knee", "hip")
  ref <- sin(seq(0, pi, length.out = np))
  norm <- list(
    variables = vars,
    mean = matrix(rep(ref, 2), nrow = 2, byrow = TRUE,
                  dimnames = list(vars, NULL)),
    sd = matrix(0.1, nrow = 2, ncol = np, dimnames = list(vars, NULL)),
    percent = seq(0, 100, length.out = np))
  pe <- list(
    knee = list(left = ref + 0.05, right = ref - 0.05),
    hip = list(left = ref, right = ref))
  clinicalGaitReport(pe, norm, indices = FALSE)
}
```

<code>gaitIndexDashboard</code>	<i>Gait index dashboard panel (GDI / GPS / MAP)</i>
---------------------------------	---

Description

A single bar panel summarising the gait indices for a clinical gait report: one bar per Gait Variable Score (the Movement Analysis Profile) plus the overall Gait Profile Score, coloured by deviation magnitude, with the Gait Deviation Index reported in the subtitle (GDI 100 = normative mean, each 10 points is one standard deviation, lower = more deviation).

Usage

```
gaitIndexDashboard(gdi, gps, map)
```

Arguments

<code>gdi</code>	A <code>gait_deviation_index</code> (from <code>PhysioMoCap::gaitDeviationIndex()</code>) or a numeric GDI value. A named numeric vector (e.g. <code>c(L = 82, R = 61)</code>) reports a per-side GDI.
<code>gps</code>	A numeric Gait Profile Score, or a <code>movement_analysis_profile</code> (its <code>\$gps</code> is used).
<code>map</code>	A <code>movement_analysis_profile</code> (from <code>PhysioMoCap::movementAnalysisProfile()</code>) or a named numeric vector of Gait Variable Scores.

Value

A ggplot object.

See Also

[clinicalGaitReport\(\)](#)

Examples

```
gaitIndexDashboard(gdi = 82,  
  gps = 8.4,  
  map = c(pelvis = 3, hip = 9, knee = 12, ankle = 6))
```

longitudinalTimeline *Longitudinal patient timeline (swimlane)*

Description

Draws a longitudinal rehabilitation timeline: each outcome measure is a stacked lane showing its trajectory across sessions, intervention periods are shaded spans across all lanes, and points where the session-to-session change crosses the Minimal Detectable Change / Minimal Clinically Important Difference are marked (via [annotateChange](#)).

Usage

```
longitudinalTimeline(  
  sessions,  
  interventions = NULL,  
  outcomes,  
  mdc = NULL,  
  mcid = NULL  
)
```

Arguments

sessions	A data.frame of session timepoints with a time column (numeric or Date) and an optional label; drawn as light session reference lines.
interventions	Optional data.frame of intervention periods with start and end columns (and an optional label); drawn as shaded spans behind every lane.
outcomes	A long data.frame of outcome measurements with columns time, metric and value (one lane per metric).
mdc, mcid	Optional MDC / MCID thresholds used to flag crossings: a scalar applied to all metrics, or a value named by metric. Both must be supplied to draw crossing markers.

Value

A ggplot object (faceted, one lane per metric).

See Also

[annotateChange\(\)](#), [as.timelineData\(\)](#)

Examples

```
sessions <- data.frame(time = 1:5)
outcomes <- data.frame(time = rep(1:5, 2),
  metric = rep(c("gait_speed", "fma"), each = 5),
  value = c(0.6, 0.7, 0.9, 1.0, 1.1, 20, 22, 28, 30, 33))
longitudinalTimeline(sessions, outcomes = outcomes,
  mdc = c(gait_speed = 0.1, fma = 5), mcid = c(gait_speed = 0.15, fma = 9))
```

NormativeModel	<i>Normative reference model (mean / SD corridor)</i>
----------------	---

Description

Builds a reusable description of a normative reference — the expected mean and sd of a measurement, optionally stratified by covariates such as age, sex or task. A model can be *scalar* (one mean/sd), a *waveform* (per-timepoint mean/sd vectors, e.g. a 101-point gait-cycle corridor), a stratified *table* (a data.frame with the strata columns plus mean/sd and an optional time column), or a *lookup function* function(age, sex, task) returning a list(mean, sd, time). The table/function forms let a normative database drive the model; when no database is available an inline scalar/waveform/table spec degrades gracefully to the same interface.

Usage

```
NormativeModel(
  mean,
  sd = NULL,
  by = c("age", "sex", "task"),
  source = NULL,
  time = NULL
)
```

Arguments

mean	Numeric scalar or vector (the reference mean / waveform), OR a data.frame of stratified norms (must contain mean and sd columns, plus the by strata columns and an optional time column), OR a lookup function(age, sex, task) returning list(mean=, sd=, time=).
sd	Numeric scalar or vector of the same length as mean (recycled from length 1). Required for the numeric form; ignored for the table / function forms.
by	Character vector naming the stratification columns/arguments (default c("age", "sex", "task")); for a data.frame the defaults are narrowed to the columns actually present.
source	Optional character string describing the normative source (citation), used as the default plot title.
time	Optional numeric vector (same length as mean) giving the x-axis (e.g. % gait cycle) for a waveform model.

Value

An object of class "NormativeModel".

See Also

[normativeZScore\(\)](#), [plotNormativeBand\(\)](#)

Examples

```
# scalar norm (e.g. comfortable gait speed for a stratum)
m <- NormativeModel(mean = 1.30, sd = 0.15, source = "Bohannon 1997")
normativeZScore(1.00, m)

# waveform norm (101-point knee-flexion corridor)
wf <- NormativeModel(mean = sin(seq(0, pi, length.out = 101)),
  sd = rep(0.1, 101), time = 0:100)
```

normativeZScore	<i>Normative z-score of an observation</i>
-----------------	--

Description

Standardizes an observed value (or waveform) against a [NormativeModel](#): $z = (value - mean) / sd$, element-wise. A scalar model recycles across a vector of observations; a waveform model returns a per-timepoint z of the same length as the observation.

Usage

```
normativeZScore(value, model, age = NULL, sex = NULL, task = NULL)
```

Arguments

value	Numeric observation. A scalar, a vector of scalars (scored against a scalar model), or a waveform matching a waveform model's length.
model	A NormativeModel .
age, sex, task	Optional strata forwarded to the model to select the applicable norm.

Value

Numeric z-score(s), matching the length of value (or the waveform length).

See Also

[NormativeModel\(\)](#), [plotNormativeBand\(\)](#)

Examples

```
m <- NormativeModel(mean = 1.30, sd = 0.15)
normativeZScore(c(1.00, 1.30, 1.60), m)
```

physioLabel	<i>Bilingual (Japanese / English) label lookup for reports</i>
-------------	--

Description

Resolves a report label key to its localized string. Unknown keys fall back to the key itself, so partial dictionaries never break a report. Labels live in `inst/extdata/labels.csv` (UTF-8), keeping the R source ASCII.

Usage

```
physioLabel(key, lang = c("en", "ja"))
```

Arguments

key	Character label key (e.g. "bias").
lang	Language: "en" (default) or "ja".

Value

The localized label string.

Examples

```
physioLabel("bias")
physioLabel("bias", "ja")
```

plotBlandAltman	<i>Bland-Altman agreement plot</i>
-----------------	------------------------------------

Description

Draws a Bland-Altman plot (mean vs. difference of paired measurements) with bias and limit-of-agreement reference lines. The statistics come from `PhysioExperiment::blandAltman()`, so the plotted lines exactly match the computed bias and limits of agreement.

Usage

```
plotBlandAltman(x, y, confidence = 0.95)
```

Arguments

x, y	Numeric vectors of paired measurements (e.g. two methods, or baseline vs follow-up).
confidence	Confidence level for the limits of agreement (default 0.95).

Value

A ggplot object.

Examples

```
plotBlandAltman(c(1, 2, 3, 4, 5), c(1.1, 1.9, 3.2, 3.8, 5.1))
```

plotChangeAnnotated *Forest-style plot of annotated pre-to-post changes*

Description

Draws one row per metric showing the signed change as a point, with the $\pm$ MDC and $\pm$ MCID thresholds as symmetric reference bands around the no-change line, coloured by the change classification.

Usage

```
plotChangeAnnotated(changes)
```

Arguments

changes A "change_annotation" object from [annotateChange](#).

Value

A ggplot object.

See Also

[annotateChange\(\)](#)

Examples

```
ch <- annotateChange(c(fma = 20, pain = 7), c(fma = 31, pain = 4),
                    mdc = c(5.2, 1.0), mcid = c(9, 2),
                    direction = c("increase", "decrease"))
plotChangeAnnotated(ch)
```

plotNormativeBand *Plot an observation against a normative band*

Description

Draws a normative corridor as graded ± 1 SD and ± 2 SD ribbons around the reference median, overlays the observed trajectory, and annotates the z-score. Uses the ecosystem colorblind-safe palette and report theme.

Usage

```
plotNormativeBand(
  observed = NULL,
  model,
  bands = c(1, 2),
  time_axis = NULL,
  annotate_z = TRUE,
  age = NULL,
  sex = NULL,
  task = NULL,
  ...
)
```

Arguments

observed	Numeric observation (a waveform, or a scalar broadcast across the corridor). NULL draws just the corridor (no observed line and no z annotation), so callers can overlay their own traces (e.g. left/right); for a scalar model this requires time_axis to set the x extent.
model	A NormativeModel .
bands	Numeric vector of SD multiples to shade (default c(1, 2)); one geom_ribbon layer per band, widest drawn first.
time_axis	Optional numeric x-axis; defaults to the model's time or the sample index.
annotate_z	Logical; if TRUE (default) annotate the (max absolute) z-score.
age, sex, task	Optional strata forwarded to the model.
...	Currently unused (reserved for future styling arguments).

Value

A ggplot object.

See Also

[NormativeModel\(\)](#), [normativeZScore\(\)](#)

Examples

```
set.seed(1)
wf <- NormativeModel(mean = sin(seq(0, pi, length.out = 101)),
                     sd = rep(0.1, 101), time = 0:100)
obs <- sin(seq(0, pi, length.out = 101)) + rnorm(101, 0, 0.05)
plotNormativeBand(obs, wf)
```

plotOutcomeRadar	<i>Multi-domain outcome radar (spider) plot</i>
------------------	---

Description

Plots a rehabilitation outcome profile across domains as a radar chart, with an optional normative reference polygon and multi-timepoint overlay. Uses the ecosystem colorblind-safe palette and report theme.

Usage

```
plotOutcomeRadar(
  profile,
  reference = NULL,
  normalize = c("percent", "z"),
  domains = NULL
)
```

Arguments

profile	Domain scores: a named numeric vector (one timepoint), a matrix or data.frame with domains as columns and timepoints as rows, or a MSKFunctionalOutcome (from <code>PhysioMSKNet::mskPredictFunctionalOutcome()</code>), whose aggregate ROM / strength / function become the domains.
reference	Optional normative reference: a named numeric of per-domain means (for "percent"), or a data.frame with domain, mean and sd columns (required for "z"). Drawn as a reference polygon.
normalize	"percent" (each domain as a percentage of its reference mean; reference polygon at 100) or "z" (z-score versus the reference; reference polygon centered at 0).
domains	Optional character vector selecting and ordering the domains.

Value

A ggplot object (a coord_polar radar).

See Also

[plotNormativeBand\(\)](#)

Examples

```
plotOutcomeRadar(c(pain = 3, balance = 7, gait = 6, strength = 5))
```

plotSCED

Single-case (SCED) plot with phase overlays

Description

Plots a single-subject time series with SCED-standard overlays: shaded phase panels, phase-change reference lines, per-phase mean lines, the baseline $\pm 2SD$ band, and per-phase split-middle celeration (trend) lines.

Usage

```
plotSCED(data, phase, overlays = c("mean", "2SD", "celeration"))
```

Arguments

data	Numeric vector of observations, in session order.
phase	Vector of phase labels (e.g. "A"/"B") the same length as data; phases are drawn in order of first appearance.
overlays	Which overlays to draw: any of "mean" (per-phase mean lines), "2SD" (baseline mean $\pm 2SD$ band) and "celeration" (split-middle trend lines). Default: all three.

Value

A ggplot object.

References

Kratochwill TR et al. (2010). Single-case designs technical documentation (What Works Clearinghouse).

See Also

[scedStats\(\)](#)

Examples

```
plotSCED(c(2, 3, 2, 3, 6, 7, 8, 7), rep(c("A", "B"), each = 4))
```

renderClinicalReport *Render a parameterized clinical report (PDF / DOCX / HTML)*

Description

Renders a one-page, bilingual, colorblind-safe clinical report for a gait, HRV or EMG assessment. PDF and HTML are produced from a parameterized Quarto template (**quarto** + the Quarto CLI); DOCX is produced by an editable **officer** path that needs no LaTeX toolchain ([buildDocxReport](#)).

Usage

```
renderClinicalReport(
  data,
  template = c("gait", "hrv", "emg"),
  format = c("pdf", "docx", "html"),
  lang = c("en", "ja"),
  out = NULL
)
```

Arguments

data	A report payload list. Recognized elements: subject (a label), figures (a named list of ggplot panels — build them with the colorblind-safe primitives, e.g. plotNormativeBand / clinicalGaitReport), change (an optional annotateChange result rendered as an MDC/MCID table) and provenance (an optional named list for the footer, e.g. from <code>metadata(pe)</code>).
template	Report template: "gait" (default), "hrv" or "emg".
format	Output format: "pdf" (default), "docx" or "html".
lang	"en" (default) or "ja"; drives the localized headings via physioLabel .
out	Output file path; defaults to a temp file named for the template and format.

Value

The output path, invisibly.

See Also

[buildDocxReport\(\)](#), [clinicalGaitReport\(\)](#), [annotateChange\(\)](#)

Examples

```
# The LaTeX-free docx path needs only officer (a Suggests dependency); the
# pdf / html paths additionally need the quarto package and the Quarto CLI.
if (requireNamespace("officer", quietly = TRUE)) {
  fig <- plotNormativeBand(NULL,
    NormativeModel(sin(seq(0, pi, length.out = 101)), rep(0.1, 101), time = 0:100))
  renderClinicalReport(list(subject = "P01", figures = list(gait_cycle = fig)),
```

```
template = "gait", format = "docx", lang = "ja")
}
```

reportTheme	<i>The x-biosignal report theme</i>
-------------	-------------------------------------

Description

Thin wrapper returning the shared colorblind-safe ggplot2 theme from **PhysioExperiment**, so every report uses consistent, accessible styling.

Usage

```
reportTheme()
```

Value

A ggplot2 theme (see `PhysioExperiment::theme_physio`).

Examples

```
reportTheme()
```

scedStats	<i>Single-case (SCED) effect-size summary</i>
-----------	---

Description

Summarises the nonoverlap / trend effect sizes for a two-phase (A/B) single- case series, delegating to the peer-reviewed estimators in **PhysioClinStats**: the Percentage of Non-overlapping Data (as a percentage), the Nonoverlap of All Pairs, and the baseline-trend-corrected Tau-U in its bounded tau-b ("scan") form.

Usage

```
scedStats(data, phase, improvement = c("increase", "decrease"))
```

Arguments

data	Numeric vector of observations, in session order.
phase	Vector of phase labels the same length as data; the first two phases (in order of appearance) are taken as A (baseline) and B (intervention).
improvement	"increase" (default) if higher scores are better, or "decrease" if lower scores are the goal.

Value

A `data.frame` of class "sced_stats" with columns `metric` and `estimate` (PND in $[0, 100]$, NAP in $[0, 1]$, Tau-U in $[-1, 1]$).

References

Parker RI, Vannest KJ (2011). Tau-U. Behavior Therapy; Scruggs et al. (1987) PND; Parker & Vannest (2009) NAP.

See Also

`plotSCED()`, `PhysioClinStats::scedTauU()`

Examples

```
# scedStats() delegates to PhysioClinStats (a Suggests dependency)
if (requireNamespace("PhysioClinStats", quietly = TRUE)) {
  scedStats(c(2, 3, 2, 3, 6, 7, 8, 7), rep(c("A", "B"), each = 4))
}
```

Index

annotateChange, [2](#), [7](#), [12](#), [16](#)
annotateChange(), [8](#), [12](#), [16](#)
as.timelineData, [3](#)
as.timelineData(), [8](#)

buildDocxReport, [4](#), [16](#)
buildDocxReport(), [16](#)

clinicalGaitReport, [5](#), [16](#)
clinicalGaitReport(), [7](#), [16](#)

gaitIndexDashboard, [6](#)
gaitIndexDashboard(), [6](#)

longitudinalTimeline, [3](#), [7](#)
longitudinalTimeline(), [4](#)

NormativeModel, [8](#), [10](#), [13](#)
NormativeModel(), [10](#), [13](#)
normativeZScore, [10](#)
normativeZScore(), [9](#), [13](#)

physioLabel, [10](#), [16](#)
plotBlandAltman, [11](#)
plotChangeAnnotated, [12](#)
plotChangeAnnotated(), [3](#)
plotNormativeBand, [5](#), [12](#), [16](#)
plotNormativeBand(), [6](#), [9](#), [10](#), [14](#)
plotOutcomeRadar, [14](#)
plotsCED, [15](#)
plotsCED(), [18](#)

renderClinicalReport, [4](#), [16](#)
renderClinicalReport(), [5](#)
reportTheme, [17](#)

scedStats, [17](#)
scedStats(), [15](#)