Package ‘PJCcalculator’

September 21, 2025

Type Package

Title PROs-Joint Contrast (PJC) Calculator

Version 0.1.3

Description Computes the Patient-Reported Outcomes (PROs) Joint Contrast (PJC), a residual-based summary that captures information left over after accounting for the clinical Disease Activity index for Psoriatic Arthritis (cDAPSA). PROs (pain and patient global assessment) and joint counts (swollen and tender) are standardized, then each component is adjusted for standardized cDAPSA using natural spline coefficients that were derived from previously published models. The resulting residuals are standardized and combined using fixed principal component loadings, to yield a continuous PJC score and quartile groupings. This package provides a calculator for applying those published coefficients to new datasets; it does not itself estimate spline models or principal components.

License MIT + file LICENSE

Depends R (>= 3.6)

Imports dplyr, splines, rlang

Config/testthat/edition 3

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

Author Ning Meng [aut, cre],
Ji Soo Kim [aut],
Ana-Maria Orbai [aut],
Scott L. Zeger [aut]

Maintainer Ning Meng <`nmeng2@jh.edu`>

Repository CRAN

Date/Publication 2025-09-21 14:00:07 UTC

Contents

`calculate_PJC` *Calculate PJC (PROs-Joint Contrast) and quartiles*

Description

Computes PJC as a loading-weighted combination of standardized residuals for Pain, Patient Global, SJC, and TJC after adjusting each for cDAPSA via a natural spline model. Includes input "tuning": coerces character columns to numeric (warning if NAs introduced) and checks for out-of-range values (SJC 0-66, TJC 0-68, Pain/Patient Global 0-10) with configurable handling. If `cDAPSA` is not provided, it is computed as `SJC` + `TJC` + `Pain` + `Patient_Global`. If `cDAPSA` is provided, it is verified against this sum (within `cdapsa_tolerance`); any discrepancy results in an error.

Usage

```
calculate_PJC(
  data,
  cohort_id = "cohort_id",
  cDAPSA = NULL,
  Pain = "Pain",
  Patient_Global = "Patient_Global",
  SJC = "SJC",
  TJC = "TJC",
  oob_action = c("stop", "na", "drop"),
  cdapsa_tolerance = 1e-08,
  center_scale = list(Pain = c(center = 4.303191, scale = 2.798819), Patient_Global =
   c(center = 4.795213, scale = 2.791098), SJC = c(center = 3.783245, scale = 4.707089),
   TJC = c(center = 5.194149, scale = 7.371234), cDAPSA = c(center = 18.0758, scale =
    14.03964)),
  ns_knots = c(-0.7176679, -0.2190796, 0.4219626),
  ns_boundary_knots = c(-1.287483, 5.265392),
  coef_list = list(Pain = c(-1.48889, 1.93539, 2.25211, 3.35687, 2.68578), Patient_Global
   = c(-1.7289, 2.1364, 2.35881, 3.95251, 2.66605), SJC = c(-0.76905, 0.47397, 1.9502,
    4.45945, 5.98404), TJC = c(-0.74115, 0.27891, 2.50892, 4.68559, 6.15326)),
  loadings = c(0.598197, 0.5960272, -0.330572, -0.4214665),
  pjc_cutoffs = c(-Inf, -0.79954204, 0.07402262, 0.88778526, Inf),
  resid_center_scale = list(center = c(Pain = 1.155879e-15, `Patient Global` =
    9.679019e-16, `Swollen Joint Count` = -2.764596e-15, `Tender Joint Count` =
    -3.534933e-15), scale = c(Pain = 0.6478511, `Patient Global` = 0.6282206,
    `Swollen Joint Count` = 0.589554, `Tender Joint Count` = 0.3902453))
)
```

Arguments

`data`	A data.frame/tibble with the required columns.
`cohort_id`	Name of the cohort id column.

`cDAPSA`	Optional. Name of the cDAPSA column. If `NULL` (default), cDAPSA is computed as `SJC` + `TJC` + `Pain` + `Patient_Global`. If non-`NULL`, the provided column is verified to equal that sum within `cdapsa_tolerance`; otherwise an error is thrown.
`Pain`	Name of the Pain column (0-10).
`Patient_Global`	Name of the Patient Global column (0-10).
`SJC`	Name of the Swollen Joint Count column (0-66).
`TJC`	Name of the Tender Joint Count column (0-68).
`oob_action`	What to do when an input is out of its valid range (SJC 0-66, TJC 0-68, Pain/Patient Global 0-10). One of: `"stop"` (error), `"na"` (keep rows but set PJC/Quartile to NA), or `"drop"` (remove rows). Default is `"stop"`.
`cdapsa_tolerance`	Numeric tolerance for comparing provided cDAPSA to the computed sum; default `1e-8`.
`center_scale`	List of centers/scales used to standardize inputs.
`ns_knots`	Numeric vector of interior knots for the spline on standardized cDAPSA.
`ns_boundary_knots`	Numeric vector of boundary knots for the spline on standardized cDAPSA.
`coef_list`	Named list of regression coefficients (intercept + 4 spline basis) for each component.
`loadings`	Numeric loadings (length 4) for Pain, Patient Global, SJC, TJC residuals.
`pjc_cutoffs`	Numeric vector of 5 cut points to define 4 quartile bins (include.lowest=TRUE).
`resid_center_scale`	List with `center` and `scale` vectors for standardizing residuals.

Value

A tibble with `cohort_id`, `PJC`, and `PJC_quartile`.

Examples

```
# Minimal example WITHOUT a cDAPSA column (it will be computed as SJC+TJC+Pain+PG)
df1 <- data.frame(
  id = 1:3,
  pain = c(4, 6, 8),
  pg   = c(3, 7, 9),
  sjc  = c(1, 3, 5),
  tjc  = c(0, 2, 4)
)
calculate_PJC(
  df1,
  cohort_id = "id",
  cDAPSA = NULL,
  Pain = "pain",
  Patient_Global = "pg",
  SJC = "sjc",
  TJC = "tjc",
```

```
  oob_action = "na"
)

# Example WITH a consistent cDAPSA column (verified against the sum)
df2 <- transform(df1, cdapsa = pain + pg + sjc + tjc)
calculate_PJC(
  df2,
  cohort_id = "id",
  cDAPSA = "cdapsa",
  Pain = "pain",
  Patient_Global = "pg",
  SJC = "sjc",
  TJC = "tjc"
)
```

Index