

# Package ‘pridit’

May 31, 2026

**Type** Package

**Title** Composite Scoring via Principal Component Analysis of Ridit Scores

**Version** 1.1.1

**Description** Implements 'PRIDIT' (Principal Component Analysis applied to 'RIDITs'), an unsupervised, nonparametric method for aggregating ordinal, categorical, and continuous indicators into a single interpretable composite score.

Originally proposed by Brockett et al. (2002)

<[doi:10.1111/1539-6975.00027](https://doi.org/10.1111/1539-6975.00027)> for insurance fraud detection and extended to hospital quality measurement by Lieberthal (2008)

<[doi:10.1111/j.1475-6773.2007.00821.x](https://doi.org/10.1111/j.1475-6773.2007.00821.x)> and Lieberthal and Comer (2013)

<[doi:10.1111/rmir.12009](https://doi.org/10.1111/rmir.12009)>.

The package provides:

- (1) low-level functions `ridit()`, `PRIDITweight()`, and `PRIDITscore()`;
- (2) a unified `pridit()` entry point returning a classed object with `print`, `summary`, `'autoplot'`, and `'coef'` methods;
- (3) `pridit_boot()` for bootstrap confidence intervals on scores and weights;
- (4) a `step_pridit()` recipe step for out-of-sample scoring within the 'tidymodels' framework; and
- (5) `pridit_longitudinal()` for panel data, computing cross-period stability of scores and weights.

**License** Apache License (>= 2)

**URL** <https://github.com/rlieberthal/PRIDIT>

**BugReports** <https://github.com/rlieberthal/PRIDIT/issues>

**Encoding** UTF-8

**LazyData** true

**Depends** R (>= 4.0.0)

**Imports** ggplot2 (>= 3.4.0), rlang, stats, utils

**Suggests** generics, patchwork, recipes (>= 1.0.0), testthat (>= 3.0.0), knitr, rmarkdown

**Config/testthat/edition** 3

**VignetteBuilder** knitr

**Config/roxygen2/version** 8.0.0

**NeedsCompilation** no

**Author** Robert D. Lieberthal [aut, cre]

**Maintainer** Robert D. Lieberthal <rlietherthal@gmail.com>

**Repository** CRAN

**Date/Publication** 2026-05-31 16:20:02 UTC

## Contents

autoplot.pridit . . . . .	2
autoplot.pridit_boot . . . . .	3
autoplot.pridit_longitudinal . . . . .	3
coef.pridit . . . . .	4
pridit . . . . .	4
PRIDITscore . . . . .	6
PRIDITweight . . . . .	7
pridit_boot . . . . .	8
pridit_longitudinal . . . . .	9
ridit . . . . .	11
step_pridit . . . . .	12
test . . . . .	13

<b>Index</b>	<b>15</b>
--------------	-----------

---

autoplot.pridit	<i>Plot a PRIDIT model</i>
-----------------	----------------------------

---

## Description

Produces a two-panel ggplot2 figure: a bar chart of the top indicator weights by magnitude (left) and a histogram of the PRIDIT score distribution (right).

## Usage

```
## S3 method for class 'pridit'
autoplot(object, top_n = 20L, ...)
```

## Arguments

object	A "pridit" object.
top_n	Integer. Number of top-weighted indicators to display. Default 20.
...	Ignored.

## Value

A ggplot object (invisibly).

---

autoplot.pridit\_boot *Plot bootstrap confidence intervals for a PRIDIT model*

---

### Description

Produces a point-and-range plot for indicator weight CIs and, if available, a ranked-score plot with error ribbons.

### Usage

```
## S3 method for class 'pridit_boot'
autoplot(object, top_n = 20L, ...)
```

### Arguments

object	A "pridit_boot" object.
top_n	Integer. Number of weights to display (by absolute estimate). Default 20.
...	Ignored.

### Value

A ggplot object (invisibly).

---

autoplot.pridit\_longitudinal  
*Plot a longitudinal PRIDIT analysis*

---

### Description

Produces two panels: (left) a heatmap of cross-period Spearman score correlations and (right) a line plot of per-indicator weight trajectories across periods.

### Usage

```
## S3 method for class 'pridit_longitudinal'
autoplot(object, top_n = 10L, ...)
```

### Arguments

object	A "pridit_longitudinal" object.
top_n	Integer. Number of indicators to show in the weight trajectory panel (by mean absolute weight across periods). Default 10.
...	Ignored.

### Value

A ggplot object (invisibly).

---

<code>coef.pridit</code>	<i>Extract PRIDIT weights</i>
--------------------------	-------------------------------

---

**Description**

Extract PRIDIT weights

**Usage**

```
## S3 method for class 'pridit'
coef(object, ...)
```

**Arguments**

<code>object</code>	A "pridit" object.
<code>...</code>	Ignored.

**Value**

Named numeric vector of PRIDIT weights.

---

<code>pridit</code>	<i>pridit: Principal Component Analysis Applied to Ridit Scoring</i>
---------------------	--

---

**Description**

The pridit package provides functions for implementing the PRIDIT (Principal Component Analysis applied to RIDITs) scoring system.

A single entry-point that runs the full PRIDIT pipeline—ridit scoring, weight estimation, and composite scoring—and returns a classed object with `print`, `summary`, `autoplot`, and `coef` methods.

**Usage**

```
pridit(data, sign_correction = TRUE)
```

**Arguments**

<code>data</code>	A data frame. The first column is treated as the observation identifier; all remaining columns must be numeric indicators.
<code>sign_correction</code>	Logical (default TRUE). If the mean weight is negative—indicating PCA chose the opposite sign convention—all weights and scores are negated so that larger positive scores correspond to the dominant high-value direction.

## Details

PRIDIT (Principal Component Analysis applied to RIDITs) was introduced by Brockett et al. (2002) for insurance fraud detection and applied to hospital quality measurement by Lieberthal (2008). Its key properties are:

1. No parametric assumptions about the data-generating process.
2. No prior knowledge of indicator direction is required; weight signs are determined entirely by the data.
3. Each indicator weight is interpretable as its contribution to the dominant latent factor.

## Value

An object of class "pridit", a list with components:

scores Data frame with columns id and PRIDITscore, sorted descending.

weights Named numeric vector of PRIDIT weights.

eigenvalue Largest eigenvalue of the ridit cross-product matrix (used for score normalisation).

eigenvalue\_ratio Ratio of the first to the second eigenvalue; large values support the single-factor interpretation.

n Number of observations.

p Number of indicators.

call Matched call.

## Author(s)

**Maintainer:** Robert D. Lieberthal <rlietherthal@gmail.com>

Authors:

- Robert D. Lieberthal <rlietherthal@gmail.com>

## References

Brockett, P. L., Derrig, R. A., Golden, L. L., Levine, A., & Alpert, M. (2002). Fraud classification using principal component analysis of RIDITs. *Journal of Risk and Insurance*, **69**(3), 341–371.

Lieberthal, R. D. (2008). Hospital quality: A PRIDIT approach. *Health Services Research*, **43**(3), 988–1005.

Lieberthal, R. D., & Comer, D. M. (2013). What are the characteristics that explain hospital quality? A longitudinal PRIDIT approach. *Risk Management and Insurance Review*, **17**(1), 17–35.

## See Also

Useful links:

- <https://github.com/rlietherthal/PRIDIT>
- Report bugs at <https://github.com/rlietherthal/PRIDIT/issues>

[pridit\\_boot](#), [pridit\\_longitudinal](#), [step\\_pridit](#)

**Examples**

```

dat <- data.frame(
  id = letters[1:10],
  x1 = runif(10),
  x2 = runif(10),
  x3 = runif(10)
)
fit <- pridit(dat)
fit
summary(fit)

```

---

PRIDITscore

*Compute PRIDIT scores*


---

**Description**

Applies a vector of PRIDIT weights to a ridit-scored data frame and returns a composite score in  $(-1, 1)$  for each observation. The score is normalised by the largest eigenvalue so that the mean score is zero by construction.

**Usage**

```
PRIDITscore(ridit_data, id_vector, weight_vec)
```

**Arguments**

<code>ridit_data</code>	A data frame returned by <a href="#">ridit</a> : first column is the ID, remaining columns are ridit scores.
<code>id_vector</code>	A vector of observation identifiers (same length and order as the rows of <code>ridit_data</code> ).
<code>weight_vec</code>	A named numeric vector of PRIDIT weights returned by <a href="#">PRIDITweight</a> .

**Value**

A data frame with columns `id` and `PRIDITscore`.

**See Also**

[ridit](#), [PRIDITweight](#), [pridit](#)

**Examples**

```

dat <- data.frame(
  id = c("A", "B", "C", "D", "E"),
  x1 = c(0.90, 0.85, 0.89, 1.00, 0.89),
  x2 = c(0.99, 0.92, 0.90, 1.00, 0.93)
)
rs <- ridit(dat)

```

```
wts <- PRIDITweight(rs)
PRIDITscore(rs, dat$id, wts)
```

---

PRIDITweight

*Compute PRIDIT weights*

---

### Description

Computes the PRIDIT weight vector from a ridit-scored data frame. Weights are the loadings of the first principal component of the ridit matrix, scaled by the column norms of that matrix. The sign of the weight vector is arbitrary (a property of PCA); pass the result to [pridit](#) rather than using this function directly if automatic sign correction is desired.

### Usage

```
PRIDITweight(ridit_data)
```

### Arguments

`ridit_data` A data frame returned by [ridit](#): first column is the ID, remaining columns are ridit scores.

### Value

A named numeric vector of PRIDIT weights, one per indicator column.

### See Also

[ridit](#), [PRIDITscore](#), [pridit](#)

### Examples

```
dat <- data.frame(
  id = c("A", "B", "C", "D", "E"),
  x1 = c(0.90, 0.85, 0.89, 1.00, 0.89),
  x2 = c(0.99, 0.92, 0.90, 1.00, 0.93)
)
rs <- ridit(dat)
PRIDITweight(rs)
```

---

pridit\_boot

*Bootstrap confidence intervals for PRIDIT scores and weights*


---

### Description

Resamples observations with replacement  $B$  times, refitting the full PRIDIT pipeline on each resample. Returns percentile confidence intervals for every indicator weight and, optionally, for every observation's score.

### Usage

```
pridit_boot(fit, data, B = 500L, conf_level = 0.95, scores = TRUE, seed = NULL)
```

### Arguments

fit	A "pridit" object from <a href="#">pridit</a> .
data	The same data frame that was passed to <code>pridit()</code> .
B	Integer. Number of bootstrap replicates. Default 500.
conf_level	Numeric in (0, 1). Coverage probability. Default 0.95.
scores	Logical. If TRUE (default), also compute CIs for per-observation scores (slower for large $n$ ).
seed	Optional integer random seed for reproducibility.

### Details

Because PCA sign is arbitrary, each bootstrap replicate's weight vector is aligned to the original fit before aggregation: if the Pearson correlation between the replicate weights and the original weights is negative, the replicate is sign-flipped.

### Value

An object of class "pridit\_boot", a list with components:

weights_ci	Data frame with columns indicator, estimate, lower, upper.
scores_ci	Data frame with columns id, estimate, lower, upper (or NULL if scores = FALSE).
B	Number of replicates used.
conf_level	Coverage probability.
call	Matched call.

### See Also

[pridit](#), [autoplot.pridit\\_boot](#)

## Examples

```
dat <- data.frame(
  id = letters[1:30],
  x1 = runif(30), x2 = runif(30), x3 = runif(30)
)
fit <- pridit(dat)
boot <- pridit_boot(fit, dat, B = 100, seed = 42)
boot
```

---

pridit\_longitudinal    *Longitudinal PRIDIT analysis*

---

## Description

Fits a separate PRIDIT model for each time period in a panel data set and summarises the stability of scores and weights across periods. The analysis follows Lieberthal & Comer (2013), who demonstrated that PRIDIT weights computed on one year's Hospital Compare data predict out-of-period outcomes in the following year, with cross-year weight correlations exceeding 0.99.

## Usage

```
pridit_longitudinal(
  data,
  id_col,
  time_col,
  indicator_cols = NULL,
  sign_correction = TRUE
)
```

## Arguments

data	A data frame in long format containing columns identified by <code>id_col</code> , <code>time_col</code> , and at least two numeric indicator columns.
id_col	Character. Name of the observation identifier column.
time_col	Character. Name of the time-period column. Periods are processed in the order returned by <code>sort(unique(data[[time_col]]))</code> .
indicator_cols	Character vector of indicator column names to include. If NULL (default), all numeric columns other than the ID and time columns are used.
sign_correction	Logical. Passed to <code>pridit</code> . Default TRUE.

## Details

Because the PCA sign is arbitrary, each period's weight vector is aligned to the *first* period before computing cross-period correlations: if the Pearson correlation between a replicate's weights and the first period's weights is negative, the replicate is sign-flipped.

Cross-period score correlations are computed only for the *balanced* panel (observations present in all periods).

## Value

An object of class "pridit\_longitudinal", a list with:

`fits` Named list of "pridit" objects, one per period.

`weight_cors` Symmetric matrix of Pearson correlations between period weight vectors.

`score_cors` Symmetric matrix of Spearman rank correlations between period scores on the balanced panel.

`scores_wide` Data frame of scores in wide format (one column per period) for the balanced panel.

`weights_long` Data frame of weights in long format with columns `period`, `indicator`, `weight`.

`periods` Sorted vector of period labels.

`n_balanced` Number of observations in the balanced panel.

`call` Matched call.

## References

Lieberthal, R. D., & Comer, D. M. (2013). What are the characteristics that explain hospital quality? A longitudinal PRIDIT approach. *Risk Management and Insurance Review*, **17**(1), 17–35.

## See Also

[pridit](#), [autoplot.pridit\\_longitudinal](#)

## Examples

```
set.seed(1)
dat <- data.frame(
  id   = rep(letters[1:20], times = 3),
  year = rep(2020:2022, each = 20),
  x1   = runif(60),
  x2   = runif(60),
  x3   = runif(60)
)
fit_long <- pridit_longitudinal(dat, id_col = "id", time_col = "year")
fit_long
```

---

ridit	<i>Compute ridit scores</i>
-------	-----------------------------

---

**Description**

Transforms a data frame of numeric indicators into ridit scores on the interval  $(-1, 1)$  using the empirical cumulative distribution of each column across the reference population. A score of zero indicates a value exactly at the median; positive scores indicate above-median values.

**Usage**

```
ridit(data)
```

**Arguments**

data	A data frame whose first column is an ID and whose remaining columns are numeric indicators.
------	--

**Details**

The ridit score for observation  $i$  on indicator  $j$  is

$$B_{ij} = F_j(x_{ij} - \varepsilon) - [1 - F_j(x_{ij})]$$

where  $F_j$  is the empirical CDF of column  $j$  and  $\varepsilon$  is a small constant that makes the lower CDF strictly left-continuous. This formulation is robust to ties and requires no parametric assumptions.

Categorical indicators should be expanded into binary dummy columns before calling `ridit()`; each dummy then receives its own ridit transformation and PRIDIT weight, with sign determined by the data rather than by the analyst.

**Value**

A data frame of the same shape as `data` with numeric columns replaced by their ridit scores. The ID column is preserved as-is.

**References**

- Bross, I. D. J. (1958). How to use ridit analysis. *Biometrics*, **14**(1), 18–38.
- Brockett, P. L., Derrig, R. A., Golden, L. L., Levine, A., & Alpert, M. (2002). Fraud classification using principal component analysis of RIDITs. *Journal of Risk and Insurance*, **69**(3), 341–371.

**Examples**

```
dat <- data.frame(
  id = c("A", "B", "C", "D", "E"),
  x1 = c(0.90, 0.85, 0.89, 1.00, 0.89),
  x2 = c(0.99, 0.92, 0.90, 1.00, 0.93)
)
ridit(dat)
```

---

step_pridit	<i>recipes step: PRIDIT composite score</i>
-------------	---

---

## Description

Creates a **recipes** preprocessing step that fits a PRIDIT model on the training data and appends a single composite score column to any data set passed to `bake()`. This enables genuine out-of-sample scoring: the empirical CDFs used for ridity transformation and the PCA weights are estimated on the training fold only and then applied to the test fold without re-fitting.

## Usage

```
step_pridit(
  recipe,
  ...,
  role = "predictor",
  trained = FALSE,
  score_name = "PRIDIT_score",
  sign_correction = TRUE,
  ecdfs = NULL,
  weights = NULL,
  max_eigval = NULL,
  col_norms = NULL,
  skip = FALSE,
  id = recipes::rand_id("pridit")
)
```

## Arguments

<code>recipe</code>	A recipe object.
<code>...</code>	One or more selector expressions passed to <code>recipes::selections()</code> that identify the numeric indicator columns to include in the PRIDIT model.
<code>role</code>	For the new score column: passed to <code>add_role()</code> . Default "predictor".
<code>trained</code>	Logical. Set automatically by <code>prep()</code> ; do not change.
<code>score_name</code>	Character. Name of the new score column. Default "PRIDIT_score".
<code>sign_correction</code>	Logical. Passed to <code>pridit</code> . Default TRUE.
<code>ecdfs</code>	Internal. Stored empirical CDFs from training.
<code>weights</code>	Internal. Stored PRIDIT weight vector from training.
<code>max_eigval</code>	Internal. Stored largest eigenvalue from training.
<code>col_norms</code>	Internal. Stored column norms from training.
<code>skip</code>	Logical. If TRUE, skip this step during <code>bake(new_data = NULL)</code> . Default FALSE.
<code>id</code>	Character. Unique step identifier.

## Details

All selected columns must be numeric. The step does not remove the original columns; use `step_rm()` afterwards if a clean feature set is required.

## Value

An updated recipe.

## Examples

```
## Not run:
library(recipes)

dat <- data.frame(
  id = letters[1:50],
  x1 = runif(50), x2 = runif(50), x3 = runif(50)
)

rec <- recipe(~ ., data = dat) |>
  update_role(id, new_role = "id") |>
  step_pridit(x1, x2, x3)

prepped <- prep(rec, training = dat)
bake(prepped, new_data = dat)

## End(Not run)
```

---

test

*Test dataset for PRIDIT analysis*

---

## Description

A sample dataset containing health quality metrics for 5 healthcare providers, used to demonstrate the PRIDIT scoring methodology.

## Usage

```
test
```

## Format

A data frame with 5 rows and 4 variables:

**ID** Character. Unique identifier for each healthcare provider (A through E)

**Smoking\_cessation** Numeric. Smoking cessation counseling rate (0.85-1.0)

**ACE\_Inhibitor** Numeric. ACE inhibitor prescription rate (0.90-1.0)

**Proper\_Antibiotic** Numeric. Proper antibiotic usage rate (0.98-1.0)

**Source**

Synthetic data created for package examples

**Examples**

```
data(test)
head(test)

# Calculate PRIDIT scores
ridit_scores <- ridit(test)
weights <- PRIDITweight(ridit_scores)
final_scores <- PRIDITscore(ridit_scores, test$ID, weights)
```

# Index

## \* datasets

test, [13](#)

autoplot.pridit, [2](#)

autoplot.pridit\_boot, [3](#), [8](#)

autoplot.pridit\_longitudinal, [3](#), [10](#)

coef.pridit, [4](#)

pridit, [4](#), [6–10](#), [12](#)

pridit-package (pridit), [4](#)

pridit\_boot, [5](#), [8](#)

pridit\_longitudinal, [5](#), [9](#)

PRIDITscore, [6](#), [7](#)

PRIDITweight, [6](#), [7](#)

ridit, [6](#), [7](#), [11](#)

step\_pridit, [5](#), [12](#)

test, [13](#)