

Package: SDI (via r-universe)

May 19, 2026

Type Package

Title Slow Digestibility Index

Version 0.1.0

Description The Slow Digestibility Index (SDI) is a tool that helps users evaluate the slow-digestion properties of crops or food matrices by combining multiple factors into a single score. It considers parameters related to starch composition [total starch (TS), amylose/amylopectin ratio (Aratio), total amylose content (TAC), and total amylopectin content (TAPC)], starch digestibility [rapidly digestible starch (RDS), slowly digestible starch (SDS) and resistant starch (RS)], structural properties [relative crystallinity (RC)], non-starch components [total protein, total oil content (TOC), and total phenolic content (TPC)], and pasting behaviour [peak viscosity (PV), pasting temperature (PT), holding strength (HS), and final viscosity (FV)]. The SDI is flexible and allows users to calculate the index using all parameters or only selected ones, depending on the data available. Users can also compute a starch-based SDI (using only starch-related parameters) or a principal component analysis (PCA)-based SDI, where weights are determined automatically from the data. Thus, the SDI provides a simple way to compare and rank crops or food samples based on their slow digestion potential. The package implements SDI(), starchSDI(), genSDI(), scoreSDI(), and pcaSDI() for estimating slow digestibility index using predefined weighted TOPSIS, starch-specific TOPSIS, user-defined weighted TOPSIS, score-based normalization, and PCA based approaches, respectively. The package has been developed using the algorithm of Pandey et al. (2026) [doi:10.1016/j.jff.2026.107208](https://doi.org/10.1016/j.jff.2026.107208).

Encoding UTF-8

Imports topsis, stats, liver

License GPL-3

Config/roxygen2/version 8.0.0

RoxygenNote 7.3.3**NeedsCompilation** no**Author** Ms. Deepanyeta Goswami [aut, cre], Dr. Veda Krishnan [aut], Dr. Md Yeasin [aut], Mr. C K Mohammed Salman [aut], Dr. Archana Singh [aut], Dr. Anil Dahuja [aut], Dr. Suresh Kumar [aut]**Maintainer** Ms. Deepanyeta Goswami <deepanyetag@gmail.com>**Repository** <https://cran.r-universe.dev>**Date/Publication** 2026-05-19 09:30:19 UTC**RemoteUrl** <https://github.com/cran/SDI>**RemoteRef** HEAD**RemoteSha** d9526d69d4786b983004bcb79c91221a7c7c8363

Contents

genSDI	2
pcaSDI	4
scoreSDI	5
SDI	7
starchSDI	9

Index	11
--------------	-----------

genSDI	<i>Generalised Slow Digestibility Index</i>
--------	---------------------------------------------

Description

Computes the generalised slow digestibility index using TOPSIS with user-defined weights and criterion type.

Usage

```
genSDI(Data, weights, type)
```

Arguments

Data	A data frame or object coercible to a data frame. The first column must contain candidate names. The remaining numeric columns of parameters.
weights	A numeric vector of criterion weights. Its length must match the number of criteria columns.
type	A character vector containing + and - signs to indicate benefit and cost criteria, respectively. Its length must match the number of criteria columns.

Details

This function computes the generalised slow digestibility index using the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS). The first column of the input data is treated as the candidate identifier, while the remaining columns are treated as numeric evaluation criteria. User-supplied weights and criterion type are used to calculate the final index score and ranking.

Value

A data frame with two columns:

Score TOPSIS score

Rank Rank

References

Pandey, S., Goswami, D., Vinutha, T., Goswami, S., Kumar, R. R., Singh, S. P., Rudra, S. G., Singh, M. K., Satyavathi, T., Girija, A., Thandapilly, S. J., & Krishnan, V. (2026). Molecular and microstructural cues for starch digestive heterogeneity in Indian pearl millet (*Pennisetum glaucum*). *Journal of Functional Foods*, 138, 107208. <https://doi.org/10.1016/j.jff.2026.107208>.

Examples

```
simulate_new_crop_data <- function(n_rep = 20, seed = 123) {
  crop_names <- c("Sample 1", "Sample 2", "Sample 3", "Sample 4", "Sample 5")
  sim_one_crop <- function(crop, n) {
    data.frame(
      Crop = rep(crop, n),
      `TS %` = runif(n, 40, 80),
      Aratio = runif(n, 0.30, 2.50),
      `TAC%` = runif(n, 20, 35),
      `TAPC%` = runif(n, 10, 60),
      `RDS %` = runif(n, 35, 75),
      `SDS%` = runif(n, 5, 25),
      `RS%` = runif(n, 10, 45),
      `IGP%` = runif(n, 65, 100),
      `RC%` = runif(n, 15, 35),
      `Total Protein` = runif(n, 5, 25),
      `Total Oil` = runif(n, 1, 10),
      `Total Phenolic` = runif(n, 0.5, 7.5),
      PV = runif(n, 800, 6000),
      PT = runif(n, 60, 75),
      HS = runif(n, 700, 5000),
      FV = runif(n, 1500, 9000),
      check.names = FALSE
    )
  }
  simulated_data <- do.call(
    rbind,
    lapply(crop_names, sim_one_crop, n = n_rep)
  )
  numeric_cols <- setdiff(names(simulated_data), "Crop")
}
```

```

simulated_data[numeric_cols] <- round(simulated_data[numeric_cols], 3)
rownames(simulated_data) <- NULL
simulated_data
}
Data <- simulate_new_crop_data(n_rep = 1, seed = 123)
wt <- c(0.105, 0.105, 0.0875, 0.0525, 0.075, 0.075, 0.075, 0.075,
        0.15, 0.04, 0.04, 0.02, 0.025, 0.025, 0.025, 0.025)
tp <- c("+", "+", "+", "-", "-", "+", "+", "-", "+", "+", "+", "+", "-", "+", "+", "+")
gensDI(Data, weights = wt, type = tp)

```

pcaSDI

*PCA-Based Slow Digestibility Index***Description**

Computes the PCA-based slow digestibility index using only principal components with eigen values greater than 1.

Usage

```
pcaSDI(Data, type)
```

Arguments

Data	A data frame or object coercible to a data frame. The first column must contain candidate names. The remaining numeric columns must follow this sequence: TS %, Aratio, TAC%, TAPC%, RDS %, SDS%, RS%, IGP%, RC%, Total Protein, Total Oil, Total Phenolic, PV, PT, HS, and FV.
type	A character vector containing + and - signs to indicate benefit and cost criteria, respectively. Its length must match the number of criteria columns.

Details

The function first transforms cost criteria into benefit form, then standardizes the data and performs principal component analysis. Only those principal components with eigen values greater than 1 are retained. The criterion weights are derived from the absolute loadings weighted by the proportion of explained variance of the selected components. The final candidate score is min-max normalized and ranked in descending order.

Value

A list with three components:

SelectedPC A data frame containing the selected principal components, their eigen values, and variance proportions.

PCAWeight A data frame containing PCA-derived weights for each criterion.

Index A data frame containing normalized score and rank for each candidate.

References

Pandey, S., Goswami, D., Vinutha, T., Goswami, S., Kumar, R. R., Singh, S. P., ... & Krishnan, V. (2026). Molecular and microstructural cues for starch digestive heterogeneity in Indian pearl millet (*Pennisetum Glaucum*). *Journal of Functional Foods*, 138, 107208.

Examples

```
simulate_new_crop_data <- function(n_rep = 20, seed = 123) {
  crop_names <- c("Sample 1", "Sample 2", "Sample 3", "Sample 4", "Sample 5")
  sim_one_crop <- function(crop, n) {
    data.frame(
      Crop = rep(crop, n),
      `TS %` = runif(n, 40, 80),
      Aratio = runif(n, 0.30, 2.50),
      `TAC%` = runif(n, 20, 35),
      `TAPC%` = runif(n, 10, 60),
      `RDS %` = runif(n, 35, 75),
      `SDS%` = runif(n, 5, 25),
      `RS%` = runif(n, 10, 45),
      `IGP%` = runif(n, 65, 100),
      `RC%` = runif(n, 15, 35),
      `Total Protein` = runif(n, 5, 25),
      `Total Oil` = runif(n, 1, 10),
      `Total Phenolic` = runif(n, 0.5, 7.5),
      PV = runif(n, 800, 6000),
      PT = runif(n, 60, 75),
      HS = runif(n, 700, 5000),
      FV = runif(n, 1500, 9000),
      check.names = FALSE
    )
  }
  simulated_data <- do.call(
    rbind,
    lapply(crop_names, sim_one_crop, n = n_rep)
  )
  numeric_cols <- setdiff(names(simulated_data), "Crop")
  simulated_data[numeric_cols] <- round(simulated_data[numeric_cols], 3)
  rownames(simulated_data) <- NULL
  simulated_data
}
Data <- simulate_new_crop_data(n_rep = 1, seed = 123)
tp <- c("+", "+", "+", "-", "-", "+", "+", "-", "+", "+", "+", "+", "-", "+", "+", "+")
pcaSDI(Data, type = tp)
```

Description

Computes the score-based slow digestibility index using criterion-wise scoring followed by min-max normalization.

Usage

```
scoreSDI(Data, type)
```

Arguments

Data	A data frame or object coercible to a data frame. The first column must contain candidate names.
type	A character vector containing + and - signs to indicate benefit and cost criteria, respectively. Its length must match the number of criteria columns.

Details

For each criterion, the score is computed relative to the observed maximum for benefit criteria and the observed minimum for cost criteria. The final score is obtained by averaging the criterion-wise scores for each candidate and applying min-max normalization.

Value

A data frame with two columns:

Score TOPSIS score

Rank Rank

References

Pandey, S., Goswami, D., Vinutha, T., Goswami, S., Kumar, R. R., Singh, S. P., Rudra, S. G., Singh, M. K., Satyavathi, T., Girija, A., Thandapilly, S. J., & Krishnan, V. (2026). Molecular and microstructural cues for starch digestive heterogeneity in Indian pearl millet (*Pennisetum glaucum*). *Journal of Functional Foods*, 138, 107208. <https://doi.org/10.1016/j.jff.2026.107208>.

Examples

```
simulate_new_crop_data <- function(n_rep = 20, seed = 123) {
  crop_names <- c("Sample 1", "Sample 2", "Sample 3", "Sample 4", "Sample 5")
  sim_one_crop <- function(crop, n) {
    data.frame(
      Crop = rep(crop, n),
      `TS %` = runif(n, 40, 80),
      Aratio = runif(n, 0.30, 2.50),
      `TAC%` = runif(n, 20, 35),
      `TAPC%` = runif(n, 10, 60),
      `RDS %` = runif(n, 35, 75),
      `SDS%` = runif(n, 5, 25),
      `RS%` = runif(n, 10, 45),
      `RC%` = runif(n, 15, 35),
    )
  }
}
```

```

    `Total Protein` = runif(n, 5, 25),
    `Total Oil` = runif(n, 1, 10),
    `Total Phenolic` = runif(n, 0.5, 7.5),
    PV = runif(n, 800, 6000),
    PT = runif(n, 60, 75),
    HS = runif(n, 700, 5000),
    FV = runif(n, 1500, 9000),
    check.names = FALSE
  )
}
simulated_data <- do.call(
  rbind,
  lapply(crop_names, sim_one_crop, n = n_rep)
)
numeric_cols <- setdiff(names(simulated_data), "Crop")
simulated_data[numeric_cols] <- round(simulated_data[numeric_cols], 3)
rownames(simulated_data) <- NULL
simulated_data
}
Data <- simulate_new_crop_data(n_rep = 1, seed = 123)
tp <- c("+", "+", "+", "-", "-", "+", "-", "+", "+", "+", "+", "-", "+", "+", "+")
scoreSDI(Data, type = tp)

```

SDI

Slow Digestibility Index

Description

Computes the slow digestibility index using TOPSIS with predefined weights and criterion type.

Usage

```
SDI(Data)
```

Arguments

Data	A data frame or object coercible to a data frame. The first column must contain candidate names. The remaining numeric columns must follow this sequence: TS%, Aratio, TAC%, TAPC%, RDS%, SDS%, RS%, RC%, Total Protein, Total Oil, Total Phenolic, PV, PT, HS and FV.
------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Details

The first column of the input data is treated as the candidate identifier. All remaining columns must be numeric criteria variables.

Value

A data frame with:

Score TOPSIS score

Rank Rank

References

Pandey, S., Goswami, D., Vinutha, T., Goswami, S., Kumar, R. R., Singh, S. P., Rudra, S. G., Singh, M. K., Satyavathi, T., Girija, A., Thandapilly, S. J., & Krishnan, V. (2026). Molecular and microstructural cues for starch digestive heterogeneity in Indian pearl millet (*Pennisetum glaucum*). *Journal of Functional Foods*, 138, 107208. <https://doi.org/10.1016/j.jff.2026.107208>.

Examples

```
simulate_new_crop_data <- function(n_rep = 20, seed = 123) {
  crop_names <- c("Sample 1", "Sample 2", "Sample 3", "Sample 4", "Sample 5")
  sim_one_crop <- function(crop, n) {
    data.frame(
      Crop = rep(crop, n),
      `TS %` = runif(n, 40, 80),
      Aratio = runif(n, 0.30, 2.50),
      `TAC%` = runif(n, 20, 35),
      `TAPC%` = runif(n, 10, 60),
      `RDS %` = runif(n, 35, 75),
      `SDS%` = runif(n, 5, 25),
      `RS%` = runif(n, 10, 45),
      `RC%` = runif(n, 15, 35),
      `Total Protein` = runif(n, 5, 25),
      `Total Oil` = runif(n, 1, 10),
      `Total Phenolic` = runif(n, 0.5, 7.5),
      PV = runif(n, 800, 6000),
      PT = runif(n, 60, 75),
      HS = runif(n, 700, 5000),
      FV = runif(n, 1500, 9000),
      check.names = FALSE
    )
  }
  simulated_data <- do.call(
    rbind,
    lapply(crop_names, sim_one_crop, n = n_rep)
  )
  numeric_cols <- setdiff(names(simulated_data), "Crop")
  simulated_data[numeric_cols] <- round(simulated_data[numeric_cols], 3)
  rownames(simulated_data) <- NULL
  simulated_data
}
Data <- simulate_new_crop_data(n_rep = 1, seed = 123)
SDI(Data)
```

starchSDI

*Starch-Based Slow Digestibility Index***Description**

Starch-Based Slow Digestibility Index

Usage

starchSDI(starchData)

Arguments

starchData A data frame or object coercible to a data frame. The first column must contain candidate names. The remaining numeric columns must follow this sequence: TS %, Aratio, TAC%, TAPC%, RDS %, SDS%, RS%.

Details

This function uses a fixed weight vector and criterion type for starch-related variables. Benefit criteria are marked with + and cost criteria are marked with -.

Value

A data frame with:

Score TOPSIS score**Rank** Rank**References**

Pandey, S., Goswami, D., Vinutha, T., Goswami, S., Kumar, R. R., Singh, S. P., Rudra, S. G., Singh, M. K., Satyavathi, T., Girija, A., Thandapilly, S. J., & Krishnan, V. (2026). Molecular and microstructural cues for starch digestive heterogeneity in Indian pearl millet (*Pennisetum glaucum*). *Journal of Functional Foods*, 138, 107208. <https://doi.org/10.1016/j.jff.2026.107208>.

Examples

```
simulate_new_crop_data <- function(n_rep = 20, seed = 123) {
  crop_names <- c("Sample 1", "Sample 2", "Sample 3", "Sample 4", "Sample 5")
  sim_one_crop <- function(crop, n) {
    data.frame(
      Crop = rep(crop, n),
      `TS %` = runif(n, 40, 80),
      Aratio = runif(n, 0.30, 2.50),
      `TAC%` = runif(n, 20, 35),
      `TAPC%` = runif(n, 10, 60),
      `RDS %` = runif(n, 35, 75),
      `SDS%` = runif(n, 5, 25),
    )
  }
}
```

```
      `RS%` = runif(n, 10, 45),
      check.names = FALSE
    )
  }
  simulated_data <- do.call(
    rbind,
    lapply(crop_names, sim_one_crop, n = n_rep)
  )
  numeric_cols <- setdiff(names(simulated_data), "Crop")
  simulated_data[numeric_cols] <- round(simulated_data[numeric_cols], 3)
  rownames(simulated_data) <- NULL
  simulated_data
}
Data <- simulate_new_crop_data(n_rep = 1, seed = 123)
starchSDI(Data)
```

Index

genSDI, [2](#)

pcaSDI, [4](#)

scoreSDI, [5](#)

SDI, [7](#)

starchSDI, [9](#)