Package 'scimo'

July 24, 2025

Title Extra Recipes Steps for Dealing with Omics Data

Version 0.0.3

Description Omics data (e.g. transcriptomics, proteomics, metagenomics...) offer a detailed and multi-dimensional perspective on the molecular components and interactions within complex biological (eco)systems. Analyzing these data requires adapted procedures, which are implemented as steps according to the 'recipes' package.

License GPL (>= 3)

URL https://github.com/abichat/scimo

BugReports https://github.com/abichat/scimo/issues

Depends R (>= 2.10), recipes (>= 1.1)

Imports dplyr, generics, magrittr, rlang, stats, tibble, tidyr

Suggests ggplot2, knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

Encoding UTF-8

LazyData false

RoxygenNote 7.3.2

NeedsCompilation no

Author Antoine BICHAT [aut, cre] (ORCID: <https://orcid.org/0000-0001-6599-7081>), Julie AUBERT [ctb] (ORCID: <https://orcid.org/0000-0001-5203-5748>)

Maintainer Antoine BICHAT <antoine.bichat@proton.me>

Repository CRAN

Date/Publication 2025-07-24 19:20:02 UTC

Contents

`cheese_abundance` *Abundance of Fungal Communities in Cheese*

Description

Fungal community abundance of 74 ASVs sampled from the surface of three different French cheeses.

Usage

```
data("cheese_abundance", package = "scimo")

data("cheese_taxonomy", package = "scimo")
```

Format

For `cheese_abundance`, a tibble with columns:

sample Sample ID.

cheese Appellation of the cheese. One of `Saint-Nectaire`, `Livarot` or `Epoisses`.

rind_type One of `Natural` or `Washed`.

other columns Count of the ASV.

For `cheese_taxonomy`, a tibble with columns:

asv Amplicon Sequence Variant (ASV) ID.

lineage Character corresponding to a standard concatenation of taxonomic clades.

other columns Clade to which the ASV belongs.

Source

This dataset came from doi:10.24072/pcjournal.321.

Examples

```
data("cheese_abundance", package = "scimo")
cheese_abundance
data("cheese_taxonomy", package = "scimo")
cheese_taxonomy
```

---

pedcan_expression *Gene Expression of Pediatric Cancer*

Description

Gene expression of 108 CCLE cell lines from 5 different pediatric cancers.

Usage

```
data("pedcan_expression", package = "scimo")
```

Format

A tibble with columns:

cell_line Cell line name.

sex One of `Male`, `Female` or `Unknown`.

event One of `Primary`, `Metastasis` or `Unknown`.

disease One of `Neuroblastoma`, `Ewing Sarcoma`, `Rhabdomyosarcoma`, `Embryonal Tumor` or `Osteosarcoma`.

other columns Expression of the gene, given in log2(TPM + 1).

Source

This dataset is generated from DepMap Public 23Q4 primary files. https://depmap.org/portal/download/all/.

Examples

```
data("pedcan_expression", package = "scimo")
pedcan_expression
```

step_aggregate_hclust *Feature aggregation step based on a hierarchical clustering*

Description

Aggregate variables according to hierarchical clustering.

Usage

```
step_aggregate_hclust(
  recipe,
  ...,
  role = "predictor",
  trained = FALSE,
  n_clusters,
  fun_agg,
  dist_metric = "euclidean",
  linkage_method = "complete",
  res = NULL,
  prefix = "cl_",
  keep_original_cols = FALSE,
  skip = FALSE,
  id = rand_id("aggregate_hclust")
)

## S3 method for class 'step_aggregate_hclust'
tidy(x, ...)
```

Arguments

recipe	A recipe object. The step will be added to the sequence of operations for this recipe.
...	One or more selector functions to choose variables for this step. See recipes::selections() for more details.
role	For model terms created by this step, what analysis role should they be assigned? By default, the new columns created by this step from the original variables will be used as predictors in a model.
trained	A logical to indicate if the quantities for preprocessing have been estimated.
n_clusters	Number of cluster to create.
fun_agg	Aggregation function like sum or mean.
dist_metric	Default to euclidean. See stats::dist() for more details.
linkage_method	Default to complete. See stats::hclust() for more details.
res	This parameter is only produced after the recipe has been trained.
prefix	A character string for the prefix of the resulting new variables.

keep_original_cols
: A logical to keep the original variables in the output. Defaults to FALSE.

skip
: A logical. Should the step be skipped when the recipe is baked by recipes::bake()? While all operations are baked when recipes::prep() is run, some operations may not be able to be conducted on new data (e.g. processing the outcome variable(s)). Care should be taken when using skip = TRUE as it may affect the computations for subsequent operations.

id
: A character string that is unique to this step to identify it.

x
: A step_aggregate_hclust object.

Value

An updated version of recipe with the new step added to the sequence of any existing operations.

Author(s)

Antoine Bichat

Examples

```
rec <-
  iris %>%
  recipe(formula = Species ~ .) %>%
  step_aggregate_hclust(all_numeric_predictors(),
                        n_clusters = 2, fun_agg = sum) %>%
  prep()
rec
tidy(rec, 1)
bake(rec, new_data = NULL)
```

---

step_aggregate_list *Feature aggregation step based on a defined list*

Description

Aggregate variables according to prior knowledge.

Usage

```
step_aggregate_list(
  recipe,
  ...,
  role = "predictor",
  trained = FALSE,
  list_agg = NULL,
  fun_agg = NULL,
  others = "discard",
  name_others = "others",
```

```
  res = NULL,
  prefix = "agg_",
  keep_original_cols = FALSE,
  skip = FALSE,
  id = rand_id("aggregate_list")
)

## S3 method for class 'step_aggregate_list'
tidy(x, ...)
```

Arguments

recipe	A recipe object. The step will be added to the sequence of operations for this recipe.
...	One or more selector functions to choose variables for this step. See recipes::selections() for more details.
role	For model terms created by this step, what analysis role should they be assigned? By default, the new columns created by this step from the original variables will be used as predictors in a model.
trained	A logical to indicate if the quantities for preprocessing have been estimated.
list_agg	Named list of aggregated variables.
fun_agg	Aggregation function like sum or mean.
others	Behavior for the selected variables in ... that are not present in list_agg. If discard (the default), they are not kept. If asis, they are kept without modification. If aggregate, they are aggregated in a new variable.
name_others	If others is set to aggregate, name of the aggregated variable. Not used otherwise.
res	This parameter is only produced after the recipe has been trained.
prefix	A character string for the prefix of the resulting new variables that are not named in list_agg.
keep_original_cols	A logical to keep the original variables in the output. Defaults to FALSE.
skip	A logical. Should the step be skipped when the recipe is baked by recipes::bake()? While all operations are baked when recipes::prep() is run, some operations may not be able to be conducted on new data (e.g. processing the outcome variable(s)). Care should be taken when using skip = TRUE as it may affect the computations for subsequent operations.
id	A character string that is unique to this step to identify it.
x	A step_aggregate_list object.

Value

An updated version of recipe with the new step added to the sequence of any existing operations.

Author(s)

Antoine Bichat

Examples

```
list_iris <- list(sepal.size = c("Sepal.Length", "Sepal.Width"),
                  petal.size = c("Petal.Length", "Petal.Width"))
rec <-
  iris %>%
  recipe(formula = Species ~ .) %>%
  step_aggregate_list(all_numeric_predictors(),
                      list_agg = list_iris, fun_agg = prod) %>%
  prep()
rec
tidy(rec, 1)
bake(rec, new_data = NULL)
```

---

step_rownormalize_tss *Feature normalization step using total sum scaling*

Description

Normalize a set of variables by converting them to proportion, making them sum to 1. Also known as simplex projection.

Usage

```
step_rownormalize_tss(
  recipe,
  ...,
  role = NA,
  trained = FALSE,
  res = NULL,
  skip = FALSE,
  id = rand_id("rownormalize_tss")
)

## S3 method for class 'step_rownormalize_tss'
tidy(x, ...)
```

Arguments

recipe	A recipe object. The step will be added to the sequence of operations for this recipe.
...	One or more selector functions to choose variables for this step. See recipes::selections() for more details.
role	Not used by this step since no new variables are created.

trained	A logical to indicate if the quantities for preprocessing have been estimated.
res	This parameter is only produced after the recipe has been trained.
skip	A logical. Should the step be skipped when the recipe is baked by recipes::bake()? While all operations are baked when recipes::prep() is run, some operations may not be able to be conducted on new data (e.g. processing the outcome variable(s)). Care should be taken when using skip = TRUE as it may affect the computations for subsequent operations.
id	A character string that is unique to this step to identify it.
x	A step_rownormalize_tss object.

Value

An updated version of recipe with the new step added to the sequence of any existing operations.

Author(s)

Antoine Bichat

Examples

```
rec <-
  recipe(Species ~ ., data = iris) %>%
  step_rownormalize_tss(all_numeric_predictors()) %>%
  prep()
rec
tidy(rec, 1)
bake(rec, new_data = NULL)
```

---

step_select_background

Feature selection step using background level

Description

Select features that exceed a background level in at least a defined number of samples.

Usage

```
step_select_background(
  recipe,
  ...,
  role = NA,
  trained = FALSE,
  background_level = NULL,
  n_samples = NULL,
  prop_samples = NULL,
  res = NULL,
```

```
  skip = FALSE,
  id = rand_id("select_background")
)

## S3 method for class 'step_select_background'
tidy(x, ...)
```

Arguments

recipe	A recipe object. The step will be added to the sequence of operations for this recipe.
...	One or more selector functions to choose variables for this step. See recipes::selections() for more details.
role	Not used by this step since no new variables are created.
trained	A logical to indicate if the quantities for preprocessing have been estimated.
background_level	Background level to exceed.
n_samples, prop_samples	Count or proportion of samples in which a feature exceeds background_level to be retained.
res	This parameter is only produced after the recipe has been trained.
skip	A logical. Should the step be skipped when the recipe is baked by recipes::bake()? While all operations are baked when recipes::prep() is run, some operations may not be able to be conducted on new data (e.g. processing the outcome variable(s)). Care should be taken when using skip = TRUE as it may affect the computations for subsequent operations.
id	A character string that is unique to this step to identify it.
x	A step_select_background object.

Value

An updated version of recipe with the new step added to the sequence of any existing operations.

Author(s)

Antoine Bichat

Examples

```
rec <-
  iris %>%
  recipe(formula = Species ~ .) %>%
  step_select_background(all_numeric_predictors(),
                         background_level = 4, prop_samples = 0.5) %>%
  prep()
rec
tidy(rec, 1)
bake(rec, new_data = NULL)
```

step_select_cv *Feature selection step using the coefficient of variation*

Description

Select variables with highest coefficient of variation.

Usage

```
step_select_cv(
  recipe,
  ...,
  role = NA,
  trained = FALSE,
  n_kept = NULL,
  prop_kept = NULL,
  cutoff = NULL,
  res = NULL,
  skip = FALSE,
  id = rand_id("select_cv")
)

## S3 method for class 'step_select_cv'
tidy(x, ...)
```

Arguments

recipe	A recipe object. The step will be added to the sequence of operations for this recipe.
...	One or more selector functions to choose variables for this step. See recipes::selections() for more details.
role	Not used by this step since no new variables are created.
trained	A logical to indicate if the quantities for preprocessing have been estimated.
n_kept	Number of variables to keep.
prop_kept	A numeric value between 0 and 1 representing the proportion of variables to keep. n_kept and prop_kept are mutually exclusive.
cutoff	Threshold beyond which (below or above) the variables are discarded.
res	This parameter is only produced after the recipe has been trained.
skip	A logical. Should the step be skipped when the recipe is baked by recipes::bake()? While all operations are baked when recipes::prep() is run, some operations may not be able to be conducted on new data (e.g. processing the outcome variable(s)). Care should be taken when using skip = TRUE as it may affect the computations for subsequent operations.
id	A character string that is unique to this step to identify it.
x	A step_select_cv object.

Value

An updated version of recipe with the new step added to the sequence of any existing operations.

Author(s)

Antoine Bichat

Examples

```
rec <-
  recipe(Species ~ ., data = iris) %>%
  step_select_cv(all_numeric_predictors(), n_kept = 2) %>%
  prep()
rec
tidy(rec, 1)
bake(rec, new_data = NULL)
```

---

step_select_kruskal *Feature selection step using Kruskal test*

Description

Select variables with the lowest (adjusted) p-value of a Kruskal-Wallis test against an outcome.

Usage

```
step_select_kruskal(
  recipe,
  ...,
  role = NA,
  trained = FALSE,
  outcome = NULL,
  n_kept = NULL,
  prop_kept = NULL,
  cutoff = NULL,
  correction = "none",
  res = NULL,
  skip = FALSE,
  id = rand_id("select_kruskal")
)

## S3 method for class 'step_select_kruskal'
tidy(x, ...)
```

Arguments

recipe	A recipe object. The step will be added to the sequence of operations for this recipe.
...	One or more selector functions to choose variables for this step. See recipes::selections() for more details.
role	Not used by this step since no new variables are created.
trained	A logical to indicate if the quantities for preprocessing have been estimated.
outcome	Name of the variable to perform the test against.
n_kept	Number of variables to keep.
prop_kept	A numeric value between 0 and 1 representing the proportion of variables to keep. n_kept and prop_kept are mutually exclusive.
cutoff	Threshold beyond which (below or above) the variables are discarded.
correction	Multiple testing correction method. One of p.adjust.methods. Default to "none".
res	This parameter is only produced after the recipe has been trained.
skip	A logical. Should the step be skipped when the recipe is baked by recipes::bake()? While all operations are baked when recipes::prep() is run, some operations may not be able to be conducted on new data (e.g. processing the outcome variable(s)). Care should be taken when using skip = TRUE as it may affect the computations for subsequent operations.
id	A character string that is unique to this step to identify it.
x	A step_select_kruskal object.

Value

An updated version of recipe with the new step added to the sequence of any existing operations.

Author(s)

Antoine Bichat

Examples

```
rec <-
  iris %>%
  recipe(formula = Species ~ .) %>%
  step_select_kruskal(all_numeric_predictors(), outcome = "Species",
                      correction = "fdr", prop_kept = 0.5) %>%
  prep()
rec
tidy(rec, 1)
bake(rec, new_data = NULL)
```

`step_select_wilcoxon` *Feature selection step using Wilcoxon test*

Description

Select variables with the lowest (adjusted) p-value of a Wilcoxon-Mann-Whitney test against an outcome.

Usage

```
step_select_wilcoxon(
  recipe,
  ...,
  role = NA,
  trained = FALSE,
  outcome = NULL,
  n_kept = NULL,
  prop_kept = NULL,
  cutoff = NULL,
  correction = "none",
  res = NULL,
  skip = FALSE,
  id = rand_id("select_wilcoxon")
)

## S3 method for class 'step_select_wilcoxon'
tidy(x, ...)
```

Arguments

`recipe`	A recipe object. The step will be added to the sequence of operations for this recipe.
`...`	One or more selector functions to choose variables for this step. See `recipes::selections()` for more details.
`role`	Not used by this step since no new variables are created.
`trained`	A logical to indicate if the quantities for preprocessing have been estimated.
`outcome`	Name of the variable to perform the test against.
`n_kept`	Number of variables to keep.
`prop_kept`	A numeric value between 0 and 1 representing the proportion of variables to keep. `n_kept` and `prop_kept` are mutually exclusive.
`cutoff`	Threshold beyond which (below or above) the variables are discarded.
`correction`	Multiple testing correction method. One of `p.adjust.methods`. Default to `"none"`.
`res`	This parameter is only produced after the recipe has been trained.

skip	A logical. Should the step be skipped when the recipe is baked by recipes::bake()? While all operations are baked when recipes::prep() is run, some operations may not be able to be conducted on new data (e.g. processing the outcome variable(s)). Care should be taken when using skip = TRUE as it may affect the computations for subsequent operations.
id	A character string that is unique to this step to identify it.
x	A step_select_wilcoxon object.

Value

An updated version of recipe with the new step added to the sequence of any existing operations.

Author(s)

Antoine Bichat

Examples

```
rec <-
  iris %>%
  dplyr::filter(Species != "virginica") %>%
  recipe(formula = Species ~ .) %>%
  step_select_wilcoxon(all_numeric_predictors(), outcome = "Species",
                       correction = "fdr", prop_kept = 0.5) %>%
  prep()
rec
tidy(rec, 1)
bake(rec, new_data = NULL)
```

---

step_taxonomy *Taxonomic clades feature generator*

Description

Extract clades from a lineage, as defined in the {yatah} package.

Usage

```
step_taxonomy(
  recipe,
  ...,
  role = "predictor",
  trained = FALSE,
  rank = NULL,
  res = NULL,
  keep_original_cols = FALSE,
  skip = FALSE,
  id = rand_id("taxonomy")
```

```
)

## S3 method for class 'step_taxonomy'
tidy(x, ...)
```

Arguments

recipe	A recipe object. The step will be added to the sequence of operations for this recipe.
...	One or more selector functions to choose variables for this step. See recipes::selections() for more details.
role	For model terms created by this step, what analysis role should they be assigned? By default, the new columns created by this step from the original variables will be used as predictors in a model.
trained	A logical to indicate if the quantities for preprocessing have been estimated.
rank	The desired ranks, a combinaison of "kingdom", "phylum", "class", "order", "family", "genus", "species", or "strain". See yatah::get_clade() for more details.
res	This parameter is only produced after the recipe has been trained.
keep_original_cols	A logical to keep the original variables in the output. Defaults to FALSE.
skip	A logical. Should the step be skipped when the recipe is baked by recipes::bake()? While all operations are baked when recipes::prep() is run, some operations may not be able to be conducted on new data (e.g. processing the outcome variable(s)). Care should be taken when using skip = TRUE as it may affect the computations for subsequent operations.
id	A character string that is unique to this step to identify it.
x	A step_taxonomy object.

Value

An updated version of recipe with the new step added to the sequence of any existing operations.

Author(s)

Antoine Bichat

Examples

```
data("cheese_taxonomy")
rec <-
  cheese_taxonomy %>%
  select(asv, lineage) %>%
  recipe(~ .) %>%
  step_taxonomy(lineage, rank = c("order", "genus")) %>%
  prep()
rec
```

```
tidy(rec, 1)
bake(rec, new_data = NULL)
```

Index