Package ‘ascent’

October 17, 2026

Title Multi-Layer Decomposition of Functional Community Restructuring

Version 0.5.0

Description Implements the 'ASC-CFD' (Assemblage Shift Characterization - Community Functional Dynamics) framework for decomposing functional community restructuring into positional (centroid displacement), dispersive (functional dispersion), and boundary (convex hull volume) components. Tests each component against two null models: a sampling null model (individuals or replicates reassigned between communities) and a stratified trait-permutation null model, with a single global test when many communities are compared. Functional entities (species with similar traits) help detect replacement among similar species, and species-level leverage identifies the taxa driving functional shifts. Supports both temporal (paired) and spatial (pairwise) comparisons.

Depends R (>= 2.10)

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.3.3

Imports cluster, ggplot2, stats, utils, geometry, withr

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

Config/testthat/edition 3

URL https://github.com/V3ndetta96/ascent

BugReports https://github.com/V3ndetta96/ascent/issues

VignetteBuilder knitr

NeedsCompilation no

Author Rogelio R. Muñoz-Li [aut, cre],
Flavia Alvarez-Denis [aut]

Maintainer Rogelio R. Muñoz-Li <munozrogelio16@gmail.com>

Repository CRAN

Date/Publication 2026-10-17 05:30:02 UTC

Contents

asc_baseline 2
asc_entities 3
asc_null 5
asc_paired 7
asc_pairwise 9
asc_transitions 11
assess_functional_space 12
birds_sim 14
plot.ascfcd 15
plot.ascfcd_entities 15
plot.ascfcd_pw 16
plot.ascfcd_transitions 16
summary.ascfcd 17
summary.ascfcd_base 17
summary.ascfcd_entities 18
summary.ascfcd_pw 18
summary.ascfcd_transitions 19

Index **20**

`asc_baseline` *Baseline Functional Topology of Entities*

Description

Calculates the absolute functional centroid (CWM), functional dispersion (FDis), and convex hull volume (FRic) for each community or site independently.

Usage

```
asc_baseline(
  traits,
  abund,
  dist_method = "gower",
  dim_retention = c("variance", "broken_stick"),
  var_tol = 0.8,
  na.rm = TRUE,
  correction = c("none", "sqrt"),
  type = NULL,
  weights = NULL
)
```

Arguments

traits	A data frame or matrix of functional traits.
abund	A data frame or matrix of species abundances.
dist_method	Distance metric for the trait matrix. Default is "gower".
dim_retention	Character. Method for dimensionality reduction. Options are "variance" or "broken_stick".
var_tol	Numeric. Proportion of cumulative variance to retain. Default is 0.80.
na.rm	Logical. Remove missing values? Default is TRUE.
correction	Character. "none" (default) builds the PCoA on the Gower dissimilarity, as in 0.1.1. "sqrt" uses its square root, which Gower (1971) showed to be Euclidean for complete data with constant weights (no negative eigenvalues). Recommended when the space quality is below 0.80. Theoretical support only: its effect on the tests has not been evaluated by simulation, and results with and without it are not comparable.
type, weights	Optional; passed unchanged to daisy (e.g. to declare asymmetric binary traits or to down-weight traits coded in several columns). Default: not used.

Value

An S3 object of class ascfcd_base.

Examples

```
# Absolute metrics of each community (no contrasts): community-weighted
# position on each axis (CWM), dispersion (FDis) and volume (FRic)
base <- asc_baseline(birds_sim$traits, birds_sim$landscape)
summary(base)

# Compare FDis and FRic along the gradient from primary forest to farmland
base$entities_results[, c("Entity", "FDis", "FRic")]
```

---

asc_entities *Identify Functional Entities (Species Clustering)*

Description

Groups species with similar traits into functional entities by hierarchical clustering (Gower distance and Ward's method by default). If neither k nor h is given, the number of entities is chosen as the one with the highest mean silhouette width. The result can be passed to asc_null (A on counts summed by entity) and asc_transitions (displacement between vs within entities), which is the recommended way to detect replacement among functionally similar species (redundancy or compensation).

Usage

```
asc_entities(
  traits,
  dist_method = "gower",
  hclust_method = "ward.D2",
  k = NULL,
  h = NULL,
  k_max = 15,
  B = 100,
  seed = NULL,
  correction = c("none", "sqrt"),
  type = NULL,
  weights = NULL
)
```

Arguments

`traits`	A data frame or matrix of functional traits (row names = species).
`dist_method`	Character. Distance metric. Default is `"gower"`.
`hclust_method`	Character. Agglomeration method. Default is `"ward.D2"`.
`k`	Integer. Number of entities. If `NULL` (and `h` is `NULL`), chosen by the silhouette.
`h`	Numeric. Height at which to cut the dendrogram (alternative to `k`).
`k_max`	Integer. Largest `k` evaluated by the criteria. Default 15.
`B`	Integer. Number of reference data sets for the gap statistic. Default 100.
`seed`	Integer or `NULL`. Seed for the gap statistic (the user's random stream is left unchanged).
`correction`, `type`, `weights`	Passed to the trait distance; see `asc_paired`.

Details

Three criteria are computed for `k = 2, ..., k_max` on the same tree:

- **Mean silhouette width** (Rousseeuw 1987), on the trait distance. Used to choose `k` by default. Below 0.25: no clear groups; 0.25-0.5: weak structure.
- **Mojena's rule**: first fusion height above the mean plus 1.25 standard deviations of all fusion heights.
- **Gap statistic** (Tibshirani et al. 2001), on the PCoA coordinates, with the `"Tibs2001SEmax"` rule. `k = 1` means no natural groups: the entities then discretise a continuum, which is still useful, but conclusions should be checked with neighbouring values of `k`.

In the validation (continuous traits of 42 butterfly species), the silhouette choice (k = 13) separated compensation from directed change, whereas very coarse entities (k = 5, Mojena's choice there) classified part of a directed change as "within entities". With discrete traits, `h = 0` gives entities in the strict sense (species sharing exactly the same trait combination); this use has not been evaluated by simulation.

Value

An S3 object of class `ascfcd_entities` with `membership` (named vector: species -> entity, accepted by `asc_null()` and `asc_transitions()`), `k`, `entity_classification` (species, entity and traits), `indices` (silhouette and gap by `k`), `suggestions` (`k` by silhouette, Mojena and gap), `silhouette_max`, `n_species_per_entity`, `distance_matrix`, `hclust_obj`, `method` and `metric`.

References

Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics, 20, 53-65.

Tibshirani, R., Walther, G. & Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. Journal of the Royal Statistical Society B, 63, 411-423.

Mojena, R. (1977). Hierarchical grouping methods and stopping rules: an evaluation. The Computer Journal, 20, 359-363.

Examples

```
# Group the 24 species of birds_sim by their traits
ent <- asc_entities(birds_sim$traits, k_max = 10, B = 50, seed = 1)

# Suggested number of entities (silhouette, Mojena, gap) and their sizes
summary(ent)

# Is the change at "Logged" a reorganisation of the functional groups
# (shift between entities) or a replacement among similar species (within)?
res <- asc_paired(birds_sim$traits, birds_sim$counts,
                  sites = birds_sim$samples$site,
                  time = birds_sim$samples$time, ref_time = "Before")
asc_transitions(res, entities = ent)$Logged$entities
```

`asc_null` *Null Models for Multi-Layer Functional Shifts*

Description

Tests each ASC-CFD layer (position, dispersion, volume) against two complementary null models, and dispatches on the class of `x`:

- objects from `asc_paired` get one set of tests per contrast (reference vs comparison);
- objects from `asc_pairwise` get a single **global test** over all communities at once (no p-value per pair).

The letters differ from version 0.1.x: A is now the sampling null model and B the stratified trait null model. The 0.1.x models (Curveball, SAD reshuffle and trait shuffle), written A0, B0 and C0 since 0.5.0, were removed.

Usage

```
asc_null(x, n_perm = 999, seed = NULL, stratum_size = 5, entities = NULL)
```

Arguments

`x`	An object of class `ascfcd` (from `asc_paired`) or `ascfcd_pw` (from `asc_pairwise`).
`n_perm`	Integer. Number of permutations. Default is 999.
`seed`	Integer or `NULL`. If given, results are reproducible and the user's random number stream is left unchanged.
`stratum_size`	Integer. Number of species per abundance stratum in null model B. Default 5 (the validated value). Strata group species of similar abundance; larger values approach the unstratified permutation.
`entities`	Optional. An object from `asc_entities` or a named vector (names = species) giving each species' entity.

Value

The input object with `null_models` appended: a data frame with one row per contrast, test and layer, and columns `Contrast`, `Test`, `Layer`, `Observed`, `Null_mean`, `Null_sd`, `SES`, `P_upper`, `P_lower` and `P_two` (two-tailed, centred on the null mean). P-values are $(b+1)/(n+1)$ and are not rounded. For pairwise objects, `site_contribution` is also appended. `null_settings` records the options used. Use `summary()` for a reading of each test.

Null model A (sampling)

H0: the communities are samples of the same community. Individuals are pooled and reassigned at random keeping sample sizes. When both states of a contrast have replicates, whole replicates are permuted instead (A by samples), which accounts for variation among replicates (individuals that are captured in groups inflate A by individuals). A needs **counts of individuals**; with relative abundances it is skipped with a warning.

Null model B (traits, stratified)

H0: the shift does not depend on which traits the species have. Abundances are kept and species coordinates are permuted only among species of similar abundance (groups of `stratum_size` species ranked by mean abundance). Stratification removes the bias of unrestricted trait permutation in empirical trait spaces. Upper tail: the shift is directed by traits. Lower tail: the shift is smaller than expected from the traits (compensation). B has low power for compensation between similar species: use `entities` for that. In the validation, B was slightly liberal for dispersion (6.5\ captured in groups (7.2\ `stratum_size = Inf` gives the unstratified permutation among the species present, which was biased in empirical trait spaces and is not recommended.

Entities

With `entities` (from `asc_entities` or a named vector assigning each species to an entity), A is repeated on counts summed by entity. "A significant for species but not for entities", together with a large fraction of the shift occurring within entities (`asc_transitions`), indicates replacement among functionally similar species (redundancy or compensation).

Global test (pairwise objects)

Statistics: mean distance of the community centroids to their common centroid (position), and the standard deviation among communities of FDis (dispersion) and FRic (volume). A global reassigns individuals among communities; B global applies one stratified trait permutation to all communities. Upper tail of A: communities differ more than samples of one community. Upper tail of B: the variation is organised by traits. Lower tail of B: less variation than expected from the traits (compensation among communities). The distance of each community to the common centroid is returned as a descriptive contribution.

References

Gower, J. C. (1971). A general coefficient of similarity and some of its properties. Biometrics, 27, 857-871.

Examples

```
# Paired contrasts (see ?birds_sim): five surveys per state at each site
res <- asc_paired(birds_sim$traits, birds_sim$counts,
                  sites = birds_sim$samples$site,
                  time = birds_sim$samples$time, ref_time = "Before")
res <- asc_null(res, n_perm = 99, seed = 1)

# The Reading column interprets each test. At "Logged", A (sampling) and
# B (traits) detect the shift in position: the change is real and directed
# by the species' traits. At "Control" there is no evidence of change.
summary(res)

# The full table (one row per contrast, test and layer)
res$null_models[res$null_models$Layer == "Position", ]

# Many communities: one global test per layer instead of one p per pair
pw <- asc_pairwise(birds_sim$traits, birds_sim$landscape)
pw <- asc_null(pw, n_perm = 99, seed = 1)
summary(pw)
```

`asc_paired` *Paired Functional Multidimensional Restructuring*

Description

Calculates centroid displacement (Layer 1), dispersion shift (Layer 2 - FDis), and volume shift (Layer 3 - FRic) between paired states.

Usage

```
asc_paired(
  traits,
  abund,
  sites,
  time,
  ref_time,
  dist_method = "gower",
  dim_retention = c("variance", "broken_stick"),
  var_tol = 0.8,
  na.rm = TRUE,
  correction = c("none", "sqrt"),
  type = NULL,
  weights = NULL
)
```

Arguments

`traits`	A data frame or matrix of functional traits.
`abund`	A data frame or matrix of species abundances.
`sites`	A vector indicating the site identifier.
`time`	A vector indicating the temporal state.
`ref_time`	A character string specifying the reference state.
`dist_method`	Distance metric for the trait matrix. Default is "gower".
`dim_retention`	Character. Method for dimensionality reduction.
`var_tol`	Numeric. Proportion of cumulative variance to retain. Default is 0.80.
`na.rm`	Logical. Remove missing values? Default is TRUE.
`correction`	Character. `"none"` (default) builds the PCoA on the Gower dissimilarity, as in 0.1.1. `"sqrt"` uses its square root, which Gower (1971) showed to be Euclidean for complete data with constant weights (no negative eigenvalues). Recommended when the space quality is below 0.80. Theoretical support only: its effect on the tests has not been evaluated by simulation, and results with and without it are not comparable.
`type`, `weights`	Optional; passed unchanged to `daisy` (e.g. to declare asymmetric binary traits or to down-weight traits coded in several columns). Default: not used.

Value

An S3 object of class `ascfcd`.

Note

Normalization: Delta FDis and Delta FRic are absolute differences, not normalized by the regional pool. Values are not directly comparable across studies with different species pools or trait scales. Only rDelta_C (position) is normalized by Dmax_regional.

FRic = NA: When a community has fewer species than retained PCoA axes, the convex hull is geometrically undefined and FRic is set to NA, which propagates to Delta_FRic.

Temporal replicates: When multiple rows share the same site and temporal state (e.g., three reference plots), their relative abundances are averaged (column means) before computing the centroid. This is equivalent to treating replicates as a single pooled community. Intra-state variability is not propagated to downstream metrics.

Examples

```
# birds_sim: two sites surveyed 5 times before and 5 times after an event.
# At "Logged" the bird community changes; at "Control" nothing changes.
res <- asc_paired(birds_sim$traits, birds_sim$counts,
                  sites = birds_sim$samples$site,
                  time = birds_sim$samples$time, ref_time = "Before")

# Layer1_rDeltaC is the centroid shift (% of the functional space): it is
# large at "Logged" and small at "Control". The last section ("Top Functional
# Drivers") lists the species that moved the centroid most (see ?asc_transitions).
summary(res)


# Is each shift larger than sampling noise (A) and organised by traits (B)?
res <- asc_null(res, n_perm = 99, seed = 1)
summary(res)
```

---

asc_pairwise *Pairwise Functional Spatial Divergence*

Description

Calculates spatial functional divergence across three layers: position (Delta C), dispersion (FDis), and volume (FRic).

Usage

```
asc_pairwise(
  traits,
  abund,
  dist_method = "gower",
  dim_retention = c("variance", "broken_stick"),
  var_tol = 0.8,
  na.rm = TRUE,
  correction = c("none", "sqrt"),
  type = NULL,
  weights = NULL
)
```

Arguments

traits	A data frame or matrix of functional traits.
abund	A data frame or matrix of species abundances.
dist_method	Distance metric for the trait matrix. Default is "gower".
dim_retention	Character. Method for dimensionality reduction.
var_tol	Numeric. Proportion of cumulative variance to retain. Default is 0.80.
na.rm	Logical. Remove missing values? Default is TRUE.
correction	Character. "none" (default) builds the PCoA on the Gower dissimilarity, as in 0.1.1. "sqrt" uses its square root, which Gower (1971) showed to be Euclidean for complete data with constant weights (no negative eigenvalues). Recommended when the space quality is below 0.80. Theoretical support only: its effect on the tests has not been evaluated by simulation, and results with and without it are not comparable.
type, weights	Optional; passed unchanged to daisy (e.g. to declare asymmetric binary traits or to down-weight traits coded in several columns). Default: not used.

Value

An S3 object of class ascfcd_pw.

Note

Leverage direction: In pairwise mode, leverage is computed from Community_A toward Community_B. Reversing the pair order reverses the leverage signs. The direction is determined by alphabetical ordering of community names.

Normalization: Delta FDis and Delta FRic are absolute differences (see asc_paired for details).

Examples

```
# Five communities along a gradient from primary forest to farmland
pw <- asc_pairwise(birds_sim$traits, birds_sim$landscape)

# One row per pair of communities; rDelta_C_pct is the centroid shift
# between the two (% of the functional space)
summary(pw)

# The three most different pairs
pr <- pw$pairwise_results
head(pr[order(-pr$rDelta_C_pct), c("Community_A", "Community_B", "rDelta_C_pct")], 3)

# To test whether the communities differ, see the global test in ?asc_null
```

`asc_transitions` *Functional Transitions: Species Leverage, Partition and Entities*

Description

Describes *who* moved the functional centroid (no p-values; for tests see `asc_null`):

- **Species leverage**: contribution of each species to the centroid displacement (Layer 1). Leverages add up exactly to the displacement $||\Delta C||$ for any reference point.
- **Partition**: share of the displacement due to species shared by both states, gained (only in the comparison) and lost (only in the reference).
- **Entities** (optional): share of the displacement that occurs between functional entities (their total abundances change) and within entities (which species dominates inside each entity changes). The two parts add up exactly to the displacement.

Usage

```
asc_transitions(
  x,
  ref = c("midpoint", "reference", "comparison"),
  entities = NULL,
  ...
)
```

Arguments

`x`	An object of class `ascfcd` or `ascfcd_pw`.
`ref`	Character. Reference point for the projections: `"midpoint"` (default), `"reference"` or `"comparison"`.
`entities`	Optional. An object from `asc_entities` or a named vector (names = species) giving each species' entity.
`...`	Further arguments passed to or from other methods.

Details

Leverage of species i is $\Delta p_i \, (F_i - r) \cdot u$, where u is the unit vector of the displacement and r the reference point. The total does not depend on r, but how it is shared among species (and among shared, gained and lost species) does. The default `ref = "midpoint"` (half-way between the two centroids) is symmetric: it gives the same partition whichever state is called the reference, which matters for pairwise comparisons. The partition also reports its minimum and maximum across the three reference points; a wide range means the partition should be interpreted with caution.

In version 0.1.1 the reference point was the reference centroid (`ref = "reference"`); per-species values differ from 0.1.1 unless that option is used.

Value

An S3 object of class `ascfcd_transitions`: a list with one element per contrast (or pair), each containing `species_leverage` (columns `Species`, `Group`, `Delta_p`, `Projection`, `Leverage`; species absent from both states are omitted), `partition` (columns `Group`, `Fraction`, `Min_across_refs`, `Max_across_refs`), `entities` (`Shift`, `Between`, `Within`, `Within_fraction`, or `NULL`), `shift` and `ref`.

Examples

```
res <- asc_paired(birds_sim$traits, birds_sim$counts,
                  sites = birds_sim$samples$site,
                  time = birds_sim$samples$time, ref_time = "Before")
tr <- asc_transitions(res)

# Species that moved the centroid at "Logged": positive leverage pushes it in
# the direction of the change, negative leverage resists it
head(tr$Logged$species_leverage)

# Leverages add up exactly to the shift
c(sum = sum(tr$Logged$species_leverage$Leverage), shift = tr$Logged$shift)

# Share of the shift due to species shared, gained or lost
tr$Logged$partition
```

---

assess_functional_space

Assess Functional Space Quality

Description

Diagnostic tool to evaluate the quality of the PCoA-based functional space constructed from a trait matrix. Intended for use before running core analyses (`asc_paired`, `asc_pairwise`, etc.) to identify potential distortion from negative eigenvalues or excessive dimensionality reduction.

Usage

```
assess_functional_space(
  traits,
  dist_method = "gower",
  dim_retention = c("variance", "broken_stick"),
  var_tol = 0.8,
  correction = c("none", "sqrt"),
  type = NULL,
  weights = NULL
)
```

Arguments

`traits`	A data frame or matrix of functional traits.
`dist_method`	Distance metric for the trait matrix. Default is `"gower"`.
`dim_retention`	Character. Method for dimensionality reduction. Options are `"variance"` or `"broken_stick"`. Default is `"variance"`.
`var_tol`	Numeric. Proportion of cumulative variance to retain (used when `dim_retention = "variance"`). Default is 0.80.
`correction`	Character. `"none"` (default) or `"sqrt"` (square root of the Gower dissimilarity; Euclidean for complete data, Gower 1971). Use it to compare the quality of both spaces.
`type`, `weights`	Optional; passed unchanged to `daisy`.

Details

The Gower distance is not strictly Euclidean, which means PCoA may produce negative eigenvalues. This function reports the full eigenvalue spectrum and computes a quality index so users can decide whether the functional space is an adequate representation of the original distance matrix.

A quality below 0.80 indicates that negative eigenvalues represent more than 20\

- Reviewing the trait data for problematic variables.
- Using `correction = "sqrt"`: the square root of the Gower dissimilarity is Euclidean for complete data with constant weights (Gower 1971), so the PCoA has no negative eigenvalues. Its effect on the null-model tests has not been evaluated by simulation.
- Using a different distance metric (e.g., `"euclidean"` for purely continuous traits).

Value

A list of class `ascfcd_space` with:

quality Numeric. PCoA quality index (`sum(positive_eig) / sum(abs(all_eig))`). Values above 0.80 are generally acceptable.

n_species Integer. Number of species in the trait matrix.

n_traits Integer. Number of traits.

k_retained Integer. Number of PCoA axes retained.

var_retained Numeric. Cumulative variance explained by retained axes.

axis_var Numeric vector. Relative variance per retained axis.

n_neg_eigenvalues Integer. Count of negative eigenvalues.

neg_eigenvalue_pct Numeric. Percentage of total absolute eigenvalue represented by negative eigenvalues.

eigenvalues Numeric vector. All raw eigenvalues from PCoA.

Examples

```
# Check the functional space before any analysis
q <- assess_functional_space(birds_sim$traits)

# Quality >= 0.8 is adequate. FRic can only be computed in samples with more
# species than retained axes.
q

# With quality < 0.8, compare with the Gower (1971) square-root correction
assess_functional_space(birds_sim$traits, correction = "sqrt")
```

`birds_sim` *Simulated Bird Communities*

Description

Simulated data used in the vignette and examples: functional traits of 24 bird species and counts of individuals in a paired design (with replicate surveys) and along a landscape gradient. Generated by `data-raw/birds_sim.R`.

Usage

```
birds_sim
```

Format

A list with four elements:

traits Data frame, 24 species (row names) x 4 traits: `Body_Mass` (g), `Beak_Length` (mm), `Diet` (factor: insects, fruit, seeds) and `Forest_Dep` (factor: no, yes).

counts Integer matrix, 20 surveys x 24 species. Two sites ("Logged", "Control"), each surveyed 5 times before and 5 times after an event. At "Logged", forest-dependent and fruit-eating species decline and small open-habitat species increase; at "Control" nothing changes.

samples Data frame describing the rows of `counts`: `id`, `site` and `time` ("Before", "After").

landscape Integer matrix, 5 communities (primary forest to farmland) x 24 species, one survey each.

Source

Simulated; see `data-raw/birds_sim.R` in the package sources.

Examples

```
str(birds_sim, max.level = 1)
```

plot.ascfcd *Plot Multidimensional Functional Restructuring*

Description

Visualizes the functional trajectory (Centroid shift) and volume shift (Convex Hull) of a specific site/contrast in the PCoA space, alongside the species leverage.

Usage

```
## S3 method for class 'ascfcd'
plot(x, contrast, type = c("both", "pcoa", "leverage"), n_sp = 10, ...)
```

Arguments

`x`	An object of class `ascfcd`.
`contrast`	Character. The exact name of the site/contrast to plot.
`type`	Character. What to plot? `"pcoa"` (Trajectory & Hull), `"leverage"` (Drivers), or `"both"`. Default is `"both"`.
`n_sp`	Integer. Number of top species to display in the leverage plot. Default is 10.
`...`	Additional graphical arguments.

Value

A ggplot object (or patchwork object if `type = "both"`).

plot.ascfcd_entities *Plot Functional Entities Dendrogram*

Description

Plots the hierarchical clustering dendrogram of the functional entities.

Usage

```
## S3 method for class 'ascfcd_entities'
plot(x, ...)
```

Arguments

`x`	An object of class `ascfcd_entities`.
`...`	Additional graphical arguments passed to `plot.hclust`.

Value

No return value, called for side effects (plots a dendrogram).

plot.ascfcd_pw *Plot Pairwise Functional Spatial Divergence*

Description

Visualizes spatial functional restructuring (Centroid & Hull) between two communities.

Usage

```
## S3 method for class 'ascfcd_pw'
plot(x, contrast, type = c("both", "pcoa", "leverage"), n_sp = 10, ...)
```

Arguments

x	An object of class ascfcd_pw.
contrast	Character. The contrast to plot (e.g., "SiteA_vs_SiteB").
type	Character. What to plot? "pcoa", "leverage", or "both".
n_sp	Integer. Number of top species to display. Default is 10.
...	Additional graphical arguments.

Value

A ggplot object.

plot.ascfcd_transitions *Plot Functional Leverage*

Description

Visualizes the top species driving functional centroid displacement.

Usage

```
## S3 method for class 'ascfcd_transitions'
plot(x, contrast, n_sp = 10, ...)
```

Arguments

x	An object of class ascfcd_transitions.
contrast	Character. The contrast to plot.
n_sp	Integer. Number of top species to display. Default is 10.
...	Additional graphical arguments.

Value

A ggplot object.

---

summary.ascfcd *Summary for Paired Functional Centroid Displacement*

Description

Provides a concise, multi-layer overview of the paired ASC-CFD analysis, including null models and top functional drivers.

Usage

```
## S3 method for class 'ascfcd'
summary(object, ...)
```

Arguments

`object`	An object of class `ascfcd`.
`...`	Further arguments passed to or from other methods.

Value

Invisibly returns a data frame with the displacement metrics.

---

summary.ascfcd_base *Summary for Baseline Functional Entities*

Description

Provides an overview of the baseline functional topology of the evaluated entities.

Usage

```
## S3 method for class 'ascfcd_base'
summary(object, ...)
```

Arguments

`object`	An object of class `ascfcd_base`.
`...`	Further arguments passed to or from other methods.

Value

Invisibly returns the entities results data frame.

summary.ascfcd_entities *Summary for Functional Entities Classification*

Description

Provides an overview of the species clustering into functional entities.

Usage

```
## S3 method for class 'ascfcd_entities'
summary(object, ...)
```

Arguments

`object`	An object of class `ascfcd_entities`.
`...`	Further arguments passed to or from other methods.

Value

Invisibly returns the classification data frame.

summary.ascfcd_pw *Summary for Pairwise Functional Spatial Divergence*

Description

Provides a comprehensive overview of the spatial network ASC-CFD analysis.

Usage

```
## S3 method for class 'ascfcd_pw'
summary(object, ...)
```

Arguments

`object`	An object of class `ascfcd_pw`.
`...`	Further arguments passed to or from other methods.

Value

Invisibly returns the pairwise results data frame.

summary.ascfcd_transitions

Summary for Functional Transitions

Description

For each contrast, shows the species with the largest leverage, the partition of the centroid displacement among shared, gained and lost species (with its range across reference points) and, if entities were given, the split between and within entities.

Usage

```
## S3 method for class 'ascfcd_transitions'
summary(object, n_sp = 5, ...)
```

Arguments

`object`	An object of class `ascfcd_transitions`.
`n_sp`	Integer. Number of top species to display per contrast. Default is 5.
`...`	Further arguments passed to or from other methods.

Value

Invisibly returns the object.

Index

∗ **datasets**
birds_sim, 14

asc_baseline, 2
asc_entities, 3, *6*, *11*
asc_null, *3*, 5, *11*
asc_paired, *4, 5*, 7, *10*
asc_pairwise, *5*, 9
asc_transitions, *3*, *6*, 11
assess_functional_space, 12

birds_sim, 14

daisy, *3*, *8*, *10*, *13*

plot.ascfcd, 15
plot.ascfcd_entities, 15
plot.ascfcd_pw, 16
plot.ascfcd_transitions, 16

summary.ascfcd, 17
summary.ascfcd_base, 17
summary.ascfcd_entities, 18
summary.ascfcd_pw, 18
summary.ascfcd_transitions, 19