Package: AntClassify (via r-universe)

May 23, 2026

Type Package

Title Functional Guilds, Invasion Status, Endemism, and Rarity of Ants

Version 0.2.0

Date 2025-04-23

Description Provides functions for the analysis of ant communities, aiming to standardize workflows in myrmecology. The package automates the assignment of species to functional guilds based on trophic strategies, feeding habits, and foraging behavior, using established classification frameworks (Silva et al., 2015 <doi:10.7476/9788574554419>; Silvestre et al., 2003 <isbn:9588151236>; Delabie et al., 2000 <https://www.researchgate.net/publication/44961742_Sampling_Ground-Dwelling_Ants_Case_Studies_from_the_Worlds_Rain_Forests>), and also includes a novel classification system implemented within the package, developed from ant species occurring in urban environments. It also includes routines to flag exotic species of Brazil (Vieira, 2025, unpublished master's thesis), identify endemic species (Silva et al., 2025 <doi:10.37885/250920259>), and classify species rarity and rarity forms of the Atlantic Forest (Silva et al., 2024 <doi:10.1016/j.biocon.2024.110640>). The package reduces manual effort and improves reproducibility, supporting research and biodiversity management of Neotropical ant communities.

License MIT + file LICENSE

URL `https://github.com/cogdebora/AntClassify`

BugReports `https://github.com/cogdebora/AntClassify/issues`

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Depends R (>= 3.5)

Imports dplyr, ggplot2, jsonlite, magrittr, rlang, scales, stats, stringr, curl

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

Config/pak/sysreqs libicu-dev libssl-dev

Repository https://cogdebora.r-universe.dev

Date/Publication 2026-04-23 20:31:35 UTC

RemoteUrl https://github.com/cogdebora/antclassify

RemoteRef HEAD

RemoteSha 9012283a4a7bab486cf7c0c87bd755602f8f728c

Contents

`ant_community` *Ant community example*

Description

A sample community matrix of ant species abundances with 3 sites and 6 species. Rows represent sites, columns represent species.

Usage

```
data(ant_community)
```

Format

A data frame with 3 rows and 6 columns:

Atta sexdens abundance of *Atta sexdens*

Camponotus atriceps abundance of *Camponotus atriceps*

Pheidole megacephala abundance of *Pheidole megacephala*

Wasmannia auropunctata abundance of *Wasmannia auropunctata*

Solenopsis saevissima abundance of *Solenopsis saevissima*

Nylanderia fulva abundance of *Nylanderia fulva*

Source

Simulated data for package examples.

Examples

```
data(ant_community)
head(ant_community)
```

`antclassify` *AntClassify Full Ecological Pipeline*

Description

Runs all ecological classification functions of the package.

Usage

```
antclassify(comm, verbose = TRUE, plot = TRUE, validate = TRUE, delay = 0.5)
```

Arguments

`comm`	Community matrix (samples x species). Species names as columns, samples as rows.
`verbose`	Logical; if TRUE, displays progress messages.
`plot`	Logical; if TRUE, displays guild proportion plots (passed to assign_guild_ants).
`validate`	Logical; if TRUE, validates species names using GBIF.
`delay`	Numeric; seconds between API calls (if validate = TRUE).

Value

A list containing results from all analyses (guilds, exotic, endemic, rarity).

`antclassify_community` *Classify ant communities across multiple sites*

Description

Applies the AntClassify pipeline to each site (row) of a community matrix and returns aggregated guild abundance and richness matrices.

Usage

```
antclassify_community(comm, guild_col = "antclassify_guild", ...)
```

Arguments

`comm`	A community matrix with sites as rows and species as columns. Species names must match the nomenclature used in the internal databases.
`guild_col`	Character string giving the name of the guild column to use for aggregation. Must be one of `"antclassify_guild"`, `"silva_guild"`, `"delabie_guild"`, or `"silvestre_guild"`. Default is `"antclassify_guild"`.
`...`	Additional arguments passed to `assign_guild_ants` (for example, `validate`, `delay`). Note that `plot` is always set to `FALSE` inside this function to avoid displaying one plot per site.

Value

A list with three components:

`by_site`	A list of length equal to `nrow(comm)`. Each element contains the full output of `assign_guild_ants` for the corresponding site.
`guild_abundance`	A matrix of guild abundances (sites × guilds).
`guild_richness`	A numeric vector of length `nrow(comm)` giving the number of distinct guilds found at each site.

Examples

```
# Small community matrix with two sites and six species
comm <- matrix(c(5, 0, 2, 0, 1, 3,
                 2, 4, 0, 2, 0, 0),
               nrow = 2, byrow = TRUE,
               dimnames = list(c("Site1", "Site2"),
                               c("Atta sexdens", "Camponotus atriceps",
                                 "Pheidole megacephala", "Wasmannia auropunctata",
                                 "Solenopsis saevissima", "Nylanderia fulva")))

# Run the classification (validate = FALSE to avoid GBIF calls in examples)
res <- antclassify_community(comm, guild_col = "antclassify_guild",
```

```
                             validate = FALSE)
res$guild_abundance
res$guild_richness
```

`assign_guild_ants` *Assign Functional Guilds for Ants*

Description

Classifies ants into functional guilds using four different ecological databases.

Usage

```
assign_guild_ants(
  comm,
  verbose = TRUE,
  plot = TRUE,
  validate = TRUE,
  delay = 0.5
)
```

Arguments

`comm`	A community matrix where species are columns and samples are rows.
`verbose`	Logical; if `TRUE`, displays progress messages.
`plot`	Logical; if `TRUE`, displays guild proportion plots.
`validate`	Logical; if `TRUE`, validates species names using GBIF before analysis.
`delay`	Numeric; seconds to wait between GBIF API calls when `validate = TRUE`.

Value

Invisibly returns a list with two elements:

`table`	A data frame containing species, abundance, percentage, and guild assignments from each system.
`plots`	A list of four ggplot2 objects, one for each guild classification system.

`check_endemic_atlantic_ants`

Identify Endemic Ant Species (Atlantic Forest)

Description

Checks a community matrix for ant species endemic to the Brazilian Atlantic Forest based on Silva et al. (2025).

Usage

```
check_endemic_atlantic_ants(
  comm,
  verbose = TRUE,
  plot = TRUE,
  plot_type = c("status", "species"),
  validate = TRUE,
  delay = 0.5
)
```

Arguments

`comm`	A community matrix where species are columns and samples are rows.
`verbose`	Logical; if `TRUE`, displays progress messages.
`plot`	Logical; if `TRUE`, displays a plot (type controlled by `plot_type`).
`plot_type`	Character; type of plot to display. `"status"` (default) shows a bar plot comparing endemic vs. other species abundance proportions. `"species"` shows a bar plot of individual endemic species abundances as a proportion of the total community.
`validate`	Logical; if `TRUE`, validates species names using GBIF before analysis.
`delay`	Numeric; seconds to wait between GBIF API calls when `validate = TRUE`.

Value

Invisibly returns a list with two elements:

`table`	A data frame containing endemic species detected, with columns species, abundance, and percentage.
`plot`	A ggplot2 object.

`check_exotic_ants` *Identify Exotic Ant Species (Brazil)*

Description

Checks a community matrix for known exotic ant species in Brazil sourced from Vieira (2025).

Usage

```
check_exotic_ants(
  comm,
  verbose = TRUE,
  plot = TRUE,
  plot_type = c("status", "species"),
  validate = TRUE,
  delay = 0.5
)
```

Arguments

`comm`	A community matrix where species are columns and samples are rows.
`verbose`	Logical; if `TRUE`, displays progress messages.
`plot`	Logical; if `TRUE`, displays a plot (type controlled by `plot_type`).
`plot_type`	Character; type of plot to display. `"status"` (default) shows a bar plot comparing exotic vs. native abundance proportions. `"species"` shows a bar plot of individual exotic species abundances as a proportion of the total community.
`validate`	Logical; if `TRUE`, validates species names using GBIF before analysis.
`delay`	Numeric; seconds to wait between GBIF API calls when `validate = TRUE`.

Value

Invisibly returns a list with two elements:

`table`	A data frame containing exotic species detected, with columns species, abundance, and percentage.
`plot`	A ggplot2 object.

`check_rarity_atlantic_ants`

Identify Ant Rarity Forms (Atlantic Forest)

Description

Checks a community matrix for ant rarity forms in the Brazilian Atlantic Forest based on Silva et al. (2024).

Usage

```
check_rarity_atlantic_ants(
  comm,
  verbose = TRUE,
  plot = TRUE,
  plot_type = c("status", "species"),
  validate = TRUE,
  delay = 0.5
)
```

Arguments

`comm`	A community matrix where species are columns and samples are rows.
`verbose`	Logical; if `TRUE`, displays progress messages.
`plot`	Logical; if `TRUE`, displays a plot (type controlled by `plot_type`).
`plot_type`	Character; type of plot to display. `"status"` (default) shows a bar plot of rarity form abundance proportions. `"species"` shows a bar plot of individual rare species abundances as a proportion of the total community, colored by rarity form.
`validate`	Logical; if `TRUE`, validates species names using GBIF before analysis.
`delay`	Numeric; seconds to wait between GBIF API calls when `validate = TRUE`.

Value

Invisibly returns a list with two elements:

`table`	A data frame containing rare species detected, with columns species, rarity_form, abundance, and percentage.
`plot`	A ggplot2 object.

`validate_species_names` *Validate species names using GBIF API*

Description

Checks species names against the GBIF taxonomic backbone. If an exact match is found, the accepted name (or canonical name) is used. If no exact match is found, the original name is kept and a warning is issued, indicating that the name may be outdated or misspelled.

Usage

```
validate_species_names(comm, verbose = TRUE, delay = 0.5)
```

Arguments

`comm`	A community matrix (rows = sites, columns = species).
`verbose`	Logical; if TRUE, displays progress messages.
`delay`	Numeric; seconds to wait between API calls (default 0.5).

Value

A community matrix with updated column names (accepted names when available).

Examples

```
## Not run:
data(ant_community)
comm_validated <- validate_species_names(ant_community, verbose = TRUE)

## End(Not run)
```

Index