

Package: rGYPSY (via r-universe)

October 27, 2026

Type Package

Title R Version of GYPSY Functions

Version 0.1.0

Description An implementation of Growth and Yield Projection System functions in R.

License GPL-3

Encoding UTF-8

Imports Rcpp, data.table

LinkingTo Rcpp

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Repository <https://benpanes.r-universe.dev>

Date/Publication 2026-10-27 18:10:06 UTC

RemoteUrl <https://github.com/benpanes/rGYPSY>

RemoteRef HEAD

RemoteSha 71a6293d3a23fd5d3b337984918eb3323d469c91

Contents

GYPSY_Age	2
GYPSY_BAinc	3
GYPSY_Ht	4
GYPSY_N	5
GYPSY_PSI	6
GYPSY_SDF	7
GYPSY_SI	8
GYPSY_SI_convert	9
GYPSY_Sim	10
GYPSY_Vol	13

Index**15**

GYPSY_Age

*Age from site index and top height***Description**

Solves the GYPSY top height equations for the age at which a stand with a given site index reaches a given top height.

Usage

```
GYPSY_Age(
  spp,
  si,
  ht,
  sitype = "B",
  ndx = 1,
  iter = 1000,
  tol = 1e-08,
  bh = 1.3,
  sh = 0.3
)
```

Arguments

spp	Species group: "AW", "SB", "SW" or "PL".
si	Site index (m), of the type given by sitype.
ht	Top height (m).
sitype	Type of si: "B" SI at breast height age 50 (SI_bh), or "T" SI at total age 50 (SI_t). SI_bh is first converted to SI_t by iteration.
ndx	Output: 1 total age, 2 stump height age, 3 breast height age.
iter	Maximum iterations.
tol	Convergence tolerance.
bh	Breast height (m).
sh	Stump height (m).

Value

Numeric vector of ages, one per row. If spp, si or ht is given as a single value, it is repeated to match the longest input, with a warning.

See Also

[GYPSY_SI\(\)](#), [GYPSY_SI_convert\(\)](#)

Examples

```
spp <- c("AW", "SB", "SW", "PL")
si_bh <- GYPSY_SI(spp, ht = rep(12, 4), age = rep(40, 4))
GYPSY_Age(spp, si_bh, ht = rep(12, 4))      # total age, 40
GYPSY_Age(spp, si_bh, ht = rep(12, 4), ndx = 3) # breast height age
```

GYPSY_BAinc	<i>Annual basal area increment</i>
-------------	------------------------------------

Description

Basal area increment (m²/ha/yr) from the GYPSY basal area increment models. [GYPSY_Sim\(\)](#) adds these each year to grow BA forward.

Usage

```
GYPSY_BAinc(
  spp,
  bhage,
  sc,
  n0,
  si_b,
  psi = 0,
  ba = 0,
  sdf_aw = 0,
  sdf_sb = 0,
  sdf_sw = 0,
  sdf_pl = 0
)
```

Arguments

spp	Species group: "AW", "SB", "SW" or "PL".
bhage	Breast height age.
sc	Species composition (0-1), based on density of species at or above breast height.
n0	Initial density (*_den0 from GYPSY_SDF()).
si_b	Site index at breast height age 50.
psi	Percent stocking index (0 when unknown).
ba	Current basal area of the species (m ² /ha).
sdf_aw, sdf_sb, sdf_sw, sdf_pl	Competitor SDFs used by SW and PL; 0 for species that have not reached breast height.

Value

BA increment (m²/ha/yr). Arguments given as a single value are repeated to match the longest input, with a warning for those supplied rather than left at their defaults.

See Also

[GYPSY_Sim\(\)](#)

Examples

```
GYPSY_BAinc("PL", bhage = 20, sc = 1, n0 = 2500, si_b = 17, ba = 15)
```

GYPSY_Ht	<i>Top height from site index and age</i>
----------	---

Description

Top height from site index and age

Usage

```
GYPSY_Ht(spp, si_t, age_t)
```

Arguments

- spp Character vector of species: "AW", "SB", "SW" or "PL".
- si_t Site index at total age 50 (m).
- age_t Total age.

Value

Top height (m). Arguments given as a single value are repeated to match the longest input, with a warning.

See Also

[GYPSY_SI\(\)](#) for the inverse.

Examples

```
GYPSY_Ht(c("AW", "SB", "SW", "PL"), si_t = rep(15, 4), age_t = rep(40, 4))
```

GYPSY_N	<i>Density at age</i>
---------	-----------------------

Description

Density (stems/ha) at a given age from a species' stand density factor.

Usage

```
GYPSY_N(spp, age, sdf, si_b, psi = 0, sdf_aw = 0, sdf_sw = 0, sdf_sb = 0)
```

Arguments

- spp Species group: "AW", "SB", "SW" or "PL".
- age Breast height age for AW, total age for other species.
- sdf Stand density factor from [GYPSY_SDF\(\)](#).
- si_b Site index at breast height age 50.
- psi Percent stocking index (0 when unknown).
- sdf_aw, sdf_sw, sdf_sb Other species' SDFs in the stand (SW uses AW; PL uses AW, SW and SB); 0 when absent.

Value

Density (stems/ha); NA when sdf is not positive. Arguments given as a single value are repeated to match the longest input, with a warning for those supplied rather than left at their defaults.

See Also

```
GYPSY\_SDF\(\)
```

Examples

```
GYPSY_N("PL", age = c(10, 30, 60), sdf = 1800, si_b = 17)
```

GYPSY_PSI	<i>Percent stocking index</i>
-----------	-------------------------------

Description

Percent stocking at total age 50 for each species in a stand, from the GYPSY percent stocking models. Requires one row per stand with species in separate columns.

Usage

```
GYPSY_PSI(  
  stand,  
  aw_ps,  
  aw_age_t,  
  aw_si_b,  
  sb_ps,  
  sb_age_t,  
  sb_si_b,  
  sw_ps,  
  sw_age_t,  
  sw_si_b,  
  pl_ps,  
  pl_age_t,  
  pl_si_b  
)
```

Arguments

- stand Stand identifiers (used for length only).
- aw_ps, sb_ps, sw_ps, pl_ps Observed percent stocking (NA if unknown).
- aw_age_t, sb_age_t, sw_age_t, pl_age_t Total age.
- aw_si_b, sb_si_b, sw_si_b, pl_si_b Site index at breast height age 50.

Value

A list with aw_psi, sb_psi, sw_psi and pl_psi, capped at 100. As in the SAS, AW, SB and SW are 0 when they can't be computed, PL is NA. Arguments given as a single value are repeated for every stand, with a warning.

See Also

```
GYPSY\_SDF\(\), GYPSY\_Sim\(\)
```

Examples

```
GYPSY_PSI(1, aw_ps = 60, aw_age_t = 30, aw_si_b = 16,
           sb_ps = NA, sb_age_t = NA, sb_si_b = NA,
           sw_ps = 40, sw_age_t = 25, sw_si_b = 15,
           pl_ps = NA, pl_age_t = NA, pl_si_b = NA)
```

GYPSY_SDF	<i>Stand density factor</i>
-----------	-----------------------------

Description

Solves the GYPSY spatial density functions for each species' stand density factor (SDF, density at age 50) from its observed density, and returns the initial density implied by that SDF. Requires one row per stand with species in separate columns.

Usage

```
GYPSY_SDF(  
  stand,  
  aw_age_b,  
  aw_n,  
  aw_si_b,  
  aw_psi,  
  sb_age_t,  
  sb_n,  
  sb_si_b,  
  sb_psi,  
  sw_age_t,  
  sw_n,  
  sw_si_b,  
  sw_psi,  
  pl_age_t,  
  pl_n,  
  pl_si_b,  
  pl_psi,  
  iter = 1000,  
  tol = 1e-08  
)
```

Arguments

- stand Stand identifiers (used for length only).
- aw_age_b AW breast height age (AW density runs on breast height age).
- aw_n, sb_n, sw_n, pl_n Observed density (stems/ha).
- aw_si_b, sb_si_b, sw_si_b, pl_si_b Site index at breast height age 50.

aw_psi, sb_psi, sw_psi, pl_psi
Percent stocking index from [GYPSY_PSI\(\)](#) (0 or NA when unknown).
sb_age_t, sw_age_t, pl_age_t
Total age.
iter
Maximum iterations.
tol
Convergence tolerance.

Details

AW is computed only when its density is over 20 stems/ha, other species when density is over 0.

Value

A list with aw_sdf, sb_sdf, sw_sdf, pl_sdf (AW, SB and SW are 0 when not computed, PL is NA) and aw_den0, sb_den0, sw_den0, pl_den0: density at age 0 (AW at breast height age 0, others at total age 0). Arguments given as a single value are repeated for every stand, with a warning.

See Also

[GYPSY_N\(\)](#), [GYPSY_PSI\(\)](#)

Examples

```
GYPSY_SDF(1, aw_age_b = 30, aw_n = 1500, aw_si_b = 16, aw_psi = 0,
           sb_age_t = NA, sb_n = NA, sb_si_b = NA, sb_psi = 0,
           sw_age_t = 30, sw_n = 400, sw_si_b = 15, sw_psi = 0,
           pl_age_t = NA, pl_n = NA, pl_si_b = NA, pl_psi = NA)
```

GYPSY_SI	<i>Site index from top height and age</i>
----------	---

Description

Solves the GYPSY top height equations for site index (top height at age 50) using fixed-point iteration given an observed top height and age.

Usage

```
GYPSY_SI(  
  spp,  
  ht,  
  age,  
  agetype = "T",  
  ndx = 2,  
  iter = 1000,  
  tol = 1e-08,  
  bh = 1.3,  
  sh = 0.3  
)
```

Arguments

spp	Species group: "AW", "SB", "SW" or "PL".
ht	Top height (m).
age	Age in years, of the type given by agetype. Total age must be at least 7.5 for AW and PL and 9.5 for SB and SW.
agetype	Type of age: "T" total, "B" breast height or "S" stump height age.
ndx	Output: 1 SI at total age 50 (SI_t), 2 SI at breast height age 50 (SI_bh), 3 years to breast height, 4 years to stump height.
iter	Maximum iterations.
tol	Convergence tolerance.
bh	Breast height (m).
sh	Stump height (m).

Value

Numeric vector, one value per row. A warning is given for rows that don't converge within `iter` iterations. If `spp`, `ht` or `age` is given as a single value, it is repeated to match the longest input, with a warning.

See Also

[GYPSY_SI_convert\(\)](#) to convert between SI_t and SI_bh, [GYPSY_Age\(\)](#) for age from SI and height.

Examples

```
spp <- c("AW", "SB", "SW", "PL")
GYPSY_SI(spp, ht = rep(12, 4), age = rep(40, 4), ndx = 1) # SI_t
GYPSY_SI(spp, ht = rep(12, 4), age = rep(40, 4))          # SI_bh
```

GYPSY_SI_convert	<i>Convert between site index types</i>
------------------	---

Description

Converts SI between total age at 50 (SI_t) or breast height age at 50 (SI_bh), or returns years to breast or stump height from a known site index.

Usage

```
GYPSY_SI_convert(
  spp,
  si,
  sitype = "B",
  ndx = 4,
  iter = 1000,
```

```
tol = 1e-08,  
bh = 1.3,  
sh = 0.3  
)
```

Arguments

spp	Species group: "AW", "SB", "SW" or "PL".
si	Site index (m), of the type given by sitype.
sitype	Type of si: "B" SI_bh or "T" SI_t.
ndx	Output: 4 the other SI type (SI_t from SI_bh, or SI_bh from SI_t), 5 years to breast height, 6 years to stump height.
iter	Maximum iterations.
tol	Convergence tolerance.
bh	Breast height (m).
sh	Stump height (m).

Value

Numeric vector, one value per row. If spp or si is given as a single value, it is repeated to match the longest input, with a warning.

See Also

```
GYPSY_SI(), GYPSY_Age()
```

Examples

```
spp <- c("AW", "SB", "SW", "PL")  
GYPSY_SI_convert(spp, si = rep(16, 4)) # SI_t from SI_bh  
GYPSY_SI_convert(spp, si = rep(16, 4), sitype = "T") # SI_bh from SI_t  
GYPSY_SI_convert(spp, si = rep(16, 4), sitype = "T", ndx = 5) # years to BH
```

Description

Runs the full GYPSY growth and yield projection. Input is one row per stand and species; output is one row per stand and year with species in columns.

Usage

```
GYPSY_Sim(
  stand,
  stand_age,
  spp,
  n,
  si = NA_real_,
  age = NA_real_,
  ht = NA_real_,
  ps = NA_real_,
  ba = NA_real_,
  sitype = "B",
  agetype = "T",
  simlen = 200,
  psflag = 1,
  baflag = 1,
  bh = 1.3,
  sh = 0.3,
  stump_ht = 0.3,
  stump_dob = 15,
  top_dib = 10,
  iter = 1000,
  tol = 1e-08,
  detail = FALSE,
  data = NULL
)
```

Arguments

stand	Stand identifier.
stand_age	Stand age (total), used for the standage output column.
spp	Species: "AW", "SB", "SW" or "PL".
n	Density (stems/ha).
si	Site index (m), of the type given by sitype.
age	Age, of the type given by agetype.
ht	Top height (m).
ps	Percent stocking (NA if unknown; non-numeric values such as "." are treated as missing).
ba	Basal area (m ² /ha, NA if unknown).
sitype	"B" SI at breast height age 50, or "T" at total age 50.
agetype	"T" total, "B" breast height or "S" stump height age.
simlen	Number of years to project.
psflag	Set to 0 to ignore ps (turns off the percent stocking models).
baflag	Set to 0 to ignore ba (turns off BA localization).

bh, sh	Breast and stump height (m) for the SI and age equations.
stump_ht, stump_dob, top_dib	Merchantability standard: stump height (m, at most 0.3), minimum stump diameter outside bark (cm) and minimum top diameter inside bark (cm). Either one value for all species or a vector named AW, SB, SW, PL.
iter	Maximum iterations for the iterative solvers.
tol	Convergence tolerance for the iterative solvers.
detail	If TRUE, also return sc_b_* (breast height based species composition), ba1_* (BA grown from the start) and ba2_* (BA grown from the observed BA).
data	Optional input table (data.frame or data.table) with one row per stand and species. Columns named stand, stand_age, spp, n, si, age, ht, ps and ba are used for those arguments; other columns are ignored. Arguments given directly take precedence over columns. A missing stand, stand_age, spp or n column is an error; other missing columns are treated as missing values, with a warning.

Details

Each row needs two of si, age and ht:

- SI is computed from ht and age (total age at least 7.5 for AW and PL, 9.5 for SB and SW);
- a supplied si overrides that SI;
- when si and ht are both given, total age is recomputed from them and the supplied age is not used.

Species without an SI are not grown, but their age still counts toward the stand's leading (oldest) age. Stands with no usable age are kept and project as NA, with a warning.

The projection starts at the fractional part of the leading species' total age and runs simlen years. Observed basal area, when given, is used to localize the predicted basal area.

Observed percent stocking (ps), when given, is converted to a percent stocking index (PSI, percent stocking at total age 50) with the percent stocking models. The PSI then adjusts the density and basal area increment models; without it, those models use their no-stocking form. Set psflag = 0 to ignore ps.

stand_age, spp, n, si, age, ht, ps and ba must be either a single value or the same length as stand. A single value is repeated for every row, with a warning for those supplied rather than left at their defaults.

Value

A data.table with one row per stand and year:

- stand, year (1 to simlen), totage (leading species total age), standage;
- for each species (_aw, _sb, _sw, _pl): totage_, bhage_, topht_, n_ (density), sc_ (species composition), ba_ (basal area), qmd_, tvol_ (total volume), mvol_ (merchantable volume), mn_ (merchantable stems), mai_t_ and mai_m_ (total and merchantable MAI by species age);
- totals: tvol_con, mvol_con (conifers), tvol_all, mvol_all, and MAI by leading age mai_t_all, mai_m_all, mai_t_con, mai_m_con, mai_t_dec, mai_m_dec.

Attribute "stand_init" holds one row per stand with the widened inputs, SI, ages, PSI, SDF, initial densities (*_den0), species composition at breast height age 0 (*_sc0), starting BA (*_ba0) and the observed / predicted BA ratio (*_rate).

See Also

The components: [GYPSY_SI\(\)](#), [GYPSY_Age\(\)](#), [GYPSY_PSI\(\)](#), [GYPSY_SDF\(\)](#), [GYPSY_N\(\)](#), [GYPSY_BAinc\(\)](#), [GYPSY_Vol\(\)](#).

Examples

```
out <- GYPSY_Sim(stand = c(1, 1), stand_age = c(20, 20), spp = c("AW", "PL"),
               n = c(375, 1825), age = c(6.2, 17.5), ht = c(3.5, 5.05),
               ba = c(0.13, 3.87))
out[year %in% c(10, 50, 100), .(year, totage, ba_pl, tvol_pl, mvol_all)]

# From an input table
inp <- data.table::fread(system.file("extdata", "example_input.csv", package = "rGYPSY"))
out <- GYPSY_Sim(data = inp, simlen = 150)
```

GYPSY_Vol	<i>Total and merchantable volume</i>
-----------	--------------------------------------

Description

Total volume, merchantable volume and merchantable density for one utilization standard.

Usage

```
GYPSY_Vol(
  spp,
  ba,
  topht,
  n,
  bhage,
  stump_ht = 0.3,
  stump_dob = 15,
  top_dib = 10
)
```

Arguments

- spp Species group: "AW", "SB", "SW" or "PL".
- ba Basal area (m2/ha).
- topht Top height (m). Volume needs top height over 1.3 m.
- n Density (stems/ha).
- bhage Breast height age.

stump_ht	Stump height (m), at most 0.3.
stump_dob	Minimum stump diameter outside bark (cm).
top_dib	Minimum top diameter inside bark (cm).

Value

A list with `tvol` (total volume, m³/ha), `mvol` (merchantable volume, m³/ha, rounded to 1e-5 as in the SAS), `mn` (merchantable stems/ha) and `qmd` (quadratic mean diameter, cm). Arguments given as a single value are repeated to match the longest input, with a warning for the stand inputs (`spp`, `ba`, `topht`, `n`, `bhage`) but not the merchantability standard.

See Also

[GYPSY_Sim\(\)](#)

Examples

```
GYPSY_Vol(c("SW", "PL"), ba = c(20, 30), topht = c(18, 20), n = c(800, 1500),  
          bhage = c(55, 60))
```

Index

GYPSY_Age, [2](#)
GYPSY_Age(), [9](#), [10](#), [13](#)
GYPSY_BAinc, [3](#)
GYPSY_BAinc(), [13](#)
GYPSY_Ht, [4](#)
GYPSY_N, [5](#)
GYPSY_N(), [8](#), [13](#)
GYPSY_PSI, [6](#)
GYPSY_PSI(), [8](#), [13](#)
GYPSY_SDF, [7](#)
GYPSY_SDF(), [3](#), [5](#), [6](#), [13](#)
GYPSY_SI, [8](#)
GYPSY_SI(), [2](#), [4](#), [10](#), [13](#)
GYPSY_SI_convert, [9](#)
GYPSY_SI_convert(), [2](#), [9](#)
GYPSY_Sim, [10](#)
GYPSY_Sim(), [3](#), [4](#), [6](#), [14](#)
GYPSY_Vol, [13](#)
GYPSY_Vol(), [13](#)