Package ‘leafR’

October 13, 2022

Type Package

Title Calculates the Leaf Area Index (LAD) and Other Related Functions

Version 0.3.5

Description A set of functions for analyzing the structure of forests based on the leaf area density (LAD) and leaf area index (LAI) measures calculated from Airborne Laser Scanning (ALS), i.e., scanning lidar (Light Detection and Ranging) data. The methodology is discussed and described in Almeida et al. (2019) <doi:10.3390/rs11010092> and Stark et al. (2012) <doi:10.1111/j.1461-0248.2012.01864.x>.

Imports lidR, sp, data.table, raster, stats

License GPL-3

Encoding UTF-8

RoxygenNote 7.1.1

URL https://github.com/DRAAlmeida/leafR, https://leafr.r-forge.r-project.org

BugReports https://github.com/DRAAlmeida/leafR/issues

Author Danilo Roberti Alves de Almeida [aut, cre] (<https://orcid.org/0000-0002-8747-0085>), Scott Christopher Stark [aut] (<https://orcid.org/0000-0002-1579-1648>), Carlos Alberto Silva [aut] (<https://orcid.org/0000-0002-7844-3560>), Caio Hamamura [aut] (<https://orcid.org/0000-0001-6149-5885>), Ruben Valbuena [aut] (<https://orcid.org/0000-0003-0493-7581>)

Maintainer Danilo Roberti Alves de Almeida <daniloflorestas@gmail.com>

Repository CRAN

Repository/R-Forge/Project leafr

Repository/R-Forge/Revision 9

Repository/R-Forge/DateTimeStamp 2021-07-02 01:39:40

Date/Publication 2021-07-04 18:30:02 UTC

NeedsCompilation no

R topics documented:

`FHD` *Foliage Height Diversity*

Description

Calculates the foliage height diversity (FHD) metric from abundances considered as per-voxel relative LAD values, as described in MacArthur and MacArthur (1961).

Usage

```
FHD(lad_profile, evenness = FALSE, LAD.threshold = -1)
```

Arguments

`lad_profile`	a data.frame including values of relative LAD at height intervals, output of the lad.profile function (use relative = TRUE)
`evenness`	boolean, defines whether FHD should be based on Shannon's diversity or evenness (Hill 1973). The default FALSE calculates Shannon diversity as the original FHD by MacArthur and MacArthur (1961); the alternative TRUE was recommended by Valbuena et al. (2012), and it calculates Shannon evenness dividing it by the natural logarithm of the number of number of voxels with LAD values above the threshold.
`LAD.threshold`	numerical (0,1), defines the minimum value of LAD for considering the relative leaf abundance of a voxel in FHD calculation. Defaults to the inverse of the total number of voxels.

Value

A `numeric` containing the Foliage Height Diversity calculated from the Leaf Area Density profile

References

Hill M. O. (1973) Diversity and evenness: a unifying notation and its consequences. Ecology. 54: 427–432. doi: 10.2307/1934352

MacArthur R.H., MacArthur J.W. (1961). On bird species diversity. Ecology 42: 594–598. doi: 10.2307/1932254

Valbuena R., Packalen P., Martín-Fernández S., Maltamo M. (2012) Diversity and equitability ordering profiles applied to the study of forest structure. Forest Ecology and Management 276: 185–195. doi: 10.1016/j.foreco.2012.03.036

Examples

```
# Get the example laz file
normlas.file = system.file("extdata", "lidar_example.laz", package="leafR")

# Calculate LAD from voxelization
VOXELS_LAD = lad.voxels(normlas.file,
                        grain.size = 2)

# Calculate the LAD profile
lad_profile = lad.profile(VOXELS_LAD, relative = TRUE)

FHD(lad_profile, evenness = FALSE)
FHD(lad_profile, evenness = TRUE)
```

GC — *Gini coefficient of foliage structural diversity*

Description

Calculates the Gini coefficient (GC) from individual LIDAR returns (i.e. without voxelization), as described for the L-coefficient of variation (equivalent to Gini) in Valbuena et al. (2017).

Usage

```
GC(normlas.file, threshold = 1)
```

Arguments

`normlas.file`	normalized las file
`threshold`	numerical, defines the minimum height considered to represent an echo from leaves.

Value

A `numeric` containing the Gini coefficient (GC) calculated from the normalized LAS file

Note

Valbuena et al. (2012) argues on why Gini is better suited to describe structural complexity the Foliage Height Diversity or the Gini-Simpon index.

References

Valbuena R., Packalen P., Martín-Fernández S., Maltamo M. (2012) Diversity and equitability ordering profiles applied to the study of forest structure. Forest Ecology and Management 276: 185–195. doi: 10.1016/j.foreco.2012.03.036 Valbuena R., Maltamo M., Mehtätalo L., Packalen P. (2017) Key structural features of Boreal forests may be detected directly using L-moments from airborne lidar data. Remote Sensing of Environment. 194: 437–446. doi: 10.1016/j.rse.2016.10.024

Examples

```
# Get the example laz file
normlas.file = system.file("extdata", "lidar_example.laz", package="leafR")

GC(normlas.file, threshold =1)
```

GS — *Gini-Simpson index of foliage structural diversity*

Description

Calculates the Gini-Simpson (GS) index metric (i.e. complement of Simpson diversity $(1 - \gamma)$ from abundances considered as per-voxel relative LAD values.

Usage

```
GS(lad_profile, evenness = FALSE, LAD.threshold = -1)
```

Arguments

`lad_profile`	a data.frame including values of relative LAD at height intervals, output of the lad.profile function (use relative = TRUE)
`evenness`	boolean, defines whether GS should be based on Simpson's diversity or evenness (Hill 1973). The default FALSE calculates Simpson's diversity (γ); the alternative TRUE was recommended by Valbuena et al. (2012), and it divides by the number of voxels with LAD values above the threshold, following Smith and Wilson (1996).
`LAD.threshold`	numerical (0,1), defines the minimum value of LAD for considering the relative leaf abundance of a voxel in GS calculation. Defaults to the inverse of the total number of voxels.

Value

A `numeric` containing the Fini-Simpson index calculated from the Leaf Area Density profile

References

Hill M. O. (1973) Diversity and evenness: a unifying notation and its consequences. Ecology. 54: 427–432. doi: 10.2307/1934352

Smith B., and Wilson J.B. (1996). A consumer's guide to evenness indices. Oikos 76: 70–82. doi: 10.2307/3545749

Valbuena R., Packalen P., Martín-Fernández S., Maltamo M. (2012) Diversity and equitability ordering profiles applied to the study of forest structure. Forest Ecology and Management 276: 185–195. doi: 10.1016/j.foreco.2012.03.036

Examples

```
# Get the example laz file
normlas.file = system.file("extdata", "lidar_example.laz", package="leafR")

# Calculate LAD from voxelization
VOXELS_LAD = lad.voxels(normlas.file,
                        grain.size = 2)

# Calculate the LAD profile
lad_profile = lad.profile(VOXELS_LAD, relative = TRUE)

GS(lad_profile, evenness = FALSE)
GS(lad_profile, evenness = TRUE)
```

---

k.coefficient — *Calculate k coefficient provided a known real LAI and the calculated LAI*

Description

Calculate k coefficient provided a known real LAI and the calculated LAI

Usage

```
k.coefficient(lidar.lai, real.lai = 6)
```

Arguments

`lidar.lai`	the output from lai() function
`real.lai`	numeric, known real LAI

Value

A `numeric` with the calculate value for k coefficient for calibrating the real LAI from calculated LAI.

Examples

```
normlas.file = system.file("extdata", "lidar_example.laz", package="leafR")

# Calculate LAD from voxelization
VOXELS_LAD = lad.voxels(normlas.file,
                        grain.size = 2)

# Calculate the LAD profile
lad_profile = lad.profile(VOXELS_LAD)

# Calculate LAI derived from LAD profile
lidar.lai = lai(lad_profile); lidar.lai

# The real LAI was measured in the field work for validation
k.coefficient(lidar.lai,  real.lai = 6)
```

---

lad.profile — *This function calculate the lad profile from the input lad.voxels*

Description

This function calculate the lad profile from the input lad.voxels

Usage

```
lad.profile(VOXELS_LAD, relative = FALSE)
```

Arguments

`VOXELS_LAD`	3D grid of LAD values (output of lad.voxels() function)
`relative`	produce lad profile by relative total LAI values. Indicate when usinh effective LAI

Value

A `data.frame` with the calculated Leaf Area Density

Examples

```
# Get the example laz file
normlas.file = system.file("extdata", "lidar_example.laz", package="leafR")

# Calculate LAD from voxelization
VOXELS_LAD = lad.voxels(normlas.file,
                        grain.size = 2)

lad_profile = lad.profile(VOXELS_LAD)
plot(lad_profile$height ~ lad_profile$lad, type = "l", ylim = c(0, 40),
```

```
     ylab = "Canopy height (m)", xlab = "LAD (m2/m3)")

# relative LAD PROFILE
relative.lad_profile = lad.profile(VOXELS_LAD, relative = TRUE)

plot(relative.lad_profile$height ~ relative.lad_profile$lad, type = "l", ylim = c(0, 40),
     ylab = "Canopy height (m)", xlab = "LAD (% of LAI)")
```

lad.voxels — *Creates a data frame of the 3D voxels information (xyz) with Leaf Area Density values from las file*

Description

Creates a data frame of the 3D voxels information (xyz) with Leaf Area Density values from las file

Usage

```
lad.voxels(normlas.file, grain.size = 1, k = 1)
```

Arguments

normlas.file	normalized las file
grain.size	horizontal resolution (suggested 1 meter for lad profiles and 10 meters for LAI maps)
k	coefficient to transform effective LAI to real LAI (k = 1; for effective LAI)

Value

A data.frame of the 3D voxels information (xyz) with Leaf Area Density values

Note

The values of LAD are not estimated below 1 meter. For the following reasons: ground points influence realtive low sampling

Examples

```
# Get the example laz file
normlas.file = system.file("extdata", "lidar_example.laz", package="leafR")

VOXELS_LAD = lad.voxels(normlas.file,
                        grain.size = 2, k=1)
```

LAHV *Leaf Area Height Volume metric*

Description

Calculates the leaf area height volume (LAHV) metric as described in Almeida et al. (2019).

Usage

```
LAHV(lad_profile, LAI.weighting = FALSE, height.weighting = FALSE)
```

Arguments

`lad_profile`	output of the lad.profile function
`LAI.weighting`	boolean, define if LAVH should be weighted by total LAI. default FALSE
`height.weighting`	boolean, define if LAVH should be weighted by the max height. default FALSE

Value

A `numeric` containing the Leaf Area Heght Volume calculated from the Leaf Area Density profile.

References

Almeida, D. R. A., Stark, S. C., Chazdon, R., Nelson, B. W., Cesar, R. G., Meli, P., . . . Brancalion, P. H. S. (2019). The effectiveness of lidar remote sensing for monitoring forest cover attributes and landscape restoration. Forest Ecology and Management, 438, 34–43. doi: 10.1016/J.FORECO.2019.02.002

Examples

```
# Get the example laz file
normlas.file = system.file("extdata", "lidar_example.laz", package="leafR")

# Calculate LAD from voxelization
VOXELS_LAD = lad.voxels(normlas.file,
                        grain.size = 2)

# Calculate the LAD profile
lad_profile = lad.profile(VOXELS_LAD)

LAHV(lad_profile, LAI.weighting = FALSE, height.weighting = FALSE)
LAHV(lad_profile, LAI.weighting = TRUE, height.weighting = FALSE)
LAHV(lad_profile, LAI.weighting = FALSE, height.weighting = TRUE)
LAHV(lad_profile, LAI.weighting = TRUE, height.weighting = TRUE)
```

lai — *calculates the lead area index (LAI)*

Description

calculates the lead area index (LAI)

Usage

```
lai(lad_profile, min = 1, max = 100)
```

Arguments

`lad_profile`	output of the lad.profile function
`min`	mix canopy height
`max`	max canopy height

Value

A `numeric` containing the LAI calculated from the Leaf Area Density

Note

The use of min and max arguments allowed the estimation of the LAI for different vertical strata

Examples

```
# Get the example laz file
normlas.file = system.file("extdata", "lidar_example.laz", package="leafR")

# Calculate LAD from voxelization
VOXELS_LAD = lad.voxels(normlas.file,
                        grain.size = 2)

# Calculate the LAD profile
lad_profile = lad.profile(VOXELS_LAD)

lidar.lai = lai(lad_profile); lidar.lai
understory.lai = lai(lad_profile, min = 1, max = 5); understory.lai

# relative LAD PROFILE
relative.lad_profile = lad.profile(VOXELS_LAD, relative = TRUE)

#understory relative LAI (% of total LAI)
relative.understory.lai = lai(relative.lad_profile, min = 1, max = 5); relative.understory.lai
```

lai.raster

Produce a raster map of LAI. The resolution of the raster depends of grain.size choosed on lad.voxel() funtion.

Description

Produce a raster map of LAI. The resolution of the raster depends of grain.size choosed on lad.voxel() funtion.

Usage

```
lai.raster(VOXELS_LAD, min = 1, relative.value = NULL)
```

Arguments

`VOXELS_LAD`	3D grid of LAD values (output of lad.voxels() function)
`min`	mix canopy height
`relative.value`	LAI map can be made in percentage of a relative lai value (indicate for effective LAI)
	A Leaf Area Index (LAI) `RasterLayer` produced from the LAD voxels output from `lad.voxels()` function.

Examples

```
library(raster)
# Get the example laz file
normlas.file = system.file("extdata", "lidar_example.laz", package="leafR")

# Calculate LAD from voxelization
# use thicker grain size to avoid voxels
# without returns
VOXELS_LAD.5 = lad.voxels(normlas.file,
                          grain.size = 5, k=1)

#Map using absolute values
lai_raster = lai.raster(VOXELS_LAD.5)
plot(lai_raster)

#############################
## RELATIVE LAI Raster
######################
# Calculate voxels LAD with finer grain size for
# better estimation of LAI
VOXELS_LAD = lad.voxels(normlas.file,
                          grain.size = 2)

# Calculate the LAD profile
lad_profile = lad.profile(VOXELS_LAD)
```

```
#Calculate LAI derived from LAD profile
lidar.lai = lai(lad_profile)

#Map using relative values (%)
relative.lai_raster = lai.raster(VOXELS_LAD.5, relative.value = lidar.lai)
plot(relative.lai_raster)
```

`lidar_example.laz` *lidar_example.laz included raw data*

Description

Tiny LiDAR data example for providing functional examples

`pointsByZSlice` *Count number of points in each Z slice*

Description

Count number of points in each Z slice

Usage

```
pointsByZSlice(Z, maxZ)
```

Arguments

`Z`	numeric vector. The heights vector.
`maxZ`	numeric. The maximum height expected in the whole dataset.

Value

A `list` of point counts in each Z slice of 1 meter

Index