

Package ‘PiC’

October 11, 2025

Type Package

Title Pointcloud Interactive Computation

Version 1.2.6

Description

Provides advanced algorithms for analyzing pointcloud data from terrestrial laser scanner in forestry applications. Key features include fast voxelization of large datasets; segmentation of point clouds into forest floor, understorey, canopy, and wood components. The package enables efficient processing of large-scale forest pointcloud data, offering insights into forest structure, connectivity, and fire risk assessment. Algorithms to analyze pointcloud data (.xyz input file).

For more details, see Ferrara & Arrizza (2025) <<https://hdl.handle.net/20.500.14243/533471>>.

For single tree segmentation details, see Ferrara et al. (2018) <[doi:10.1016/j.agrformet.2018.04.008](https://doi.org/10.1016/j.agrformet.2018.04.008)>.

License GPL (>= 3)

Depends R (>= 4.3)

Imports collapse, conicfit, data.table, dbscan, dplyr, foreach, magrittr, sf, stats, tictoc, utils

Suggests DT, fs, ggplot2, later, plotly, shiny, shinycssloaders, shinydashboard, shinydashboardPlus, shinyFeedback, shinyFiles, shinyjs, shinythemes, shinyWidgets, testthat (>= 3.0.0), tools, withr

Config/testthat/edition 3

Encoding UTF-8

RoxygenNote 7.3.2

URL <https://github.com/rupppy/PiC>

BugReports <https://github.com/rupppy/PiC/issues>

NeedsCompilation no

Author Roberto Ferrara [aut, cre] (ORCID:

<<https://orcid.org/0009-0000-3627-6867>>),

Stefano Arrizza [ctb] (ORCID: <<https://orcid.org/0009-0009-2290-3650>>)

Maintainer Roberto Ferrara <roberto.ferrara@cnr.it>

Repository CRAN

Date/Publication 2025-10-11 18:00:02 UTC

Contents

Calculate_trees_metrics	2
Floseg	3
Forest_seg	4
run_PiC	5
SegOne	6
Voxels	6
Index	8

Calculate_trees_metrics
<i>Calculate tree and canopy metrics</i>

Description

Computes metrics for individual trees and forest canopy from segmented point clouds.

Usage

```
Calculate_trees_metrics(  
  woodpoint,  
  a,  
  AGB_def,  
  Forest_floor,  
  plot,  
  filename,  
  output_path,  
  canopy_voxel_size = 0.1,  
  min_canopy_height = 1.5,  
  coverage_method = "mean_normalized"  
)
```

Arguments

woodpoint	Wood points (trunks and branches) with cluster attribute
a	Original point cloud
AGB_def	Non-wood (foliage) points
Forest_floor	Forest floor points
plot	Plot/output file prefix

filename	Original file prefix
output_path	Output directory
canopy_voxel_size	Voxel size for canopy analysis
min_canopy_height	Minimum height for canopy analysis
coverage_method	Coverage degree calculation method

Value

List containing tree metrics, canopy metrics, and file paths

Floseg	<i>Forest floor segmentation</i>
--------	----------------------------------

Description

Segments the input .xyz pointcloud file into different forestry layers: forest floor and above ground biomass.

Usage

```
Floseg(a, filename="XXX", soil_dim = 0.3, th = 20, N=500, output_path = tempdir())
```

Arguments

a	- Input file (.xyz)
filename	- Output file prefix
soil_dim	- Voxel dimension (m) for forest floor segmentation - Default = 0.30
th	- Minimum number of point to generate a voxel. Default = 20
N	- Minimum number of voxel to generate a cluster. Default = 500
output_path	Directory in cui scrivere i file di output. Default = tempdir()

Value

2 files (.txt) output. 1. Forest floor pointcloud; 2. AGB pointcloud

Forest_seg	<i>Forest component segmentation</i>
------------	--------------------------------------

Description

Segments an input .xyz point cloud file into different forestry layers (soil, wood, foliage), computes individual tree metrics, and provides summary statistics and canopy metrics.

Usage

```
Forest_seg(
  a,
  filename = "XXX",
  dimVox = 2,
  th = 2,
  eps = 2,
  mpts = 9,
  h_tree = 1,
  soil_dim = 0.1,
  N = 500,
  R = 30,
  Vox_print = FALSE,
  WoodVox_print = FALSE,
  output_path = tempdir(),
  analyze_canopy = TRUE,
  canopy_voxel_size = 0.1,
  min_canopy_height = 1.5,
  coverage_method = "mean_normalized"
)
```

Arguments

a	Input point cloud data frame (.xyz) or file path
filename	Output file prefix
dimVox	Voxel dimension (cm) for wood segmentation (default = 2)
th	Minimum number of points to generate a voxel (default = 2)
eps	Epsilon neighborhood radius for DBSCAN (default = 2)
mpts	Minimum points required in eps neighborhood for core points (default = 9)
h_tree	Minimum trunk length in meters (default = 1)
soil_dim	Voxel dimension (m) for forest floor segmentation (default = 0.1)
N	Minimum number of voxels in a wood cluster (default = 500)
R	Cluster shape parameter threshold (default = 30)
Vox_print	Logical; if TRUE, saves point cloud voxelization (default = FALSE)
WoodVox_print	Logical; if TRUE, saves wood voxelization (default = FALSE)

output_path Output directory (default = tempdir())
analyze_canopy Logical; if TRUE, performs canopy analysis (default = TRUE)
canopy_voxel_size Voxel size for canopy analysis in meters (default = 0.1)
min_canopy_height Minimum height threshold for canopy analysis (default = 1.5)
coverage_method Method for calculating coverage degree (default = "mean_normalized")

Value

List containing file paths and metrics for trees and canopy.

run_PiC	<i>Launch PiC Shiny App</i>
---------	-----------------------------

Description

Launch the Shiny app for interactive 3D point cloud processing.

Usage

```
run_PiC()
```

Details

This function launches an interactive web application for analyzing forest point cloud data. The app requires additional packages that are not installed by default. If these packages are missing, you will be prompted to install them.

Value

No return value, called for side effects (launches Shiny app)

Examples

```
## Not run:  
# Launch the interactive app  
run_PiC()  
  
## End(Not run)
```

SegOne

Single Tree wood leaf segmentation

Description

Wood - leaf segmentation of single tree

Usage

```
SegOne(a, filename = "Elab_single_tree", dimVox = 2, th = 2,
eps = 1, mpts = 4, N = 1000, R = 30, output_path = tempdir())
```

Arguments

dimVox	- voxel dimension in cm - Default = 2
th	- Minimum number of points to generate a voxel - Default = 2
filename	- Output file prefix
a	- AGB voxelized input file
eps	- size (radius) of the epsilon neighborhood - Default = 1
mpts	- number of minimum points required in the eps neighborhood for core points (including the point itself) - Default = 4
N	- Minimum number of voxel in a wood cluster - Default = 1000
R	- R = Standard deviation * Proportion of Variance - Default = 30
output_path	Directory in cui scrivere i file di output. Default = tempdir()

Value

Two file (.txt) in output - Wood points and non wood points

Voxels

Voxelize point cloud

Description

Transform pointcloud in voxel

Usage

```
Voxels(a, filename = "XXX", dimVox = 2, th = 2, output_path = tempdir())
```

Arguments

a	- input file
filename	- file output prefix
dimVox	- voxel dimension in cm - Default = 2
th	Minimum number of point to generate a voxel (Default = 1) Is a parameter that should be used with caution; it generates a lightened cloud with fewer points. To be evaluated in relation with the dimVox parameter, for high point densities it is efficace to remove noise (outliers)
output_path	Directory in cui scrivere i file di output. Default = tempdir()

Value

Voxelized pointcloud

Index

Calculate_trees_metrics, [2](#)

Floseg, [3](#)

Forest_seg, [4](#)

run_PiC, [5](#)

SegOne, [6](#)

Voxels, [6](#)