Package: sunscanimport (via r-universe)

October 27, 2026

Type Package

Title Imports data from sunscan device

Version 0.3.5

Date 2026-07-29

Author Gunther Krauss <guntherkrauss@uni-bonn.de>

Maintainer Gunther Krauss <guntherkrauss@uni-bonn.de>

Description Provides functions to import, convert and visualize LAI measurements from Sunscan device. An interactive shiny app is included.

License GPL-3

Encoding UTF-8

Depends R (>= 4.1.0)

Imports stats, utils, tidyr, readr, dplyr, ggplot2, digest, shiny, DT, zip, rlang

LazyLoad yes

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

Config/roxygen2/version 8.0.0

Config/pak/sysreqs cmake make libicu-dev libuv1-dev libx11-dev zlib1g-dev

Repository https://gk-crop.r-universe.dev

Date/Publication 2026-07-29 16:20:53 UTC

RemoteUrl https://github.com/gk-crop/sunscanimport

RemoteRef HEAD

RemoteSha 859de92ce3ace27faefa0c40105d156e60a07e8b

Contents

addPlotID 2
addPlotIDFromData 3

convertSunScanDirectory . . . 4
convertSunscanFile . . . 4
countMeasurements . . . 5
createBoxplot . . . 5
createFolders . . . 6
createGridPlot . . . 6
createGridPlotLAI . . . 7
createSeriesInfo . . . 7
createSummary . . . 8
createTimePlot . . . 8
createTimePlotLAI . . . 9
generateInitialGridData . . . 9
generateReport . . . 10
generateSampleMeasurementIdData . . . 10
generateSamplePlotIdData . . . 11
getData . . . 11
getFileList . . . 12
getHeader . . . 13
getParData . . . 13
getSmallHeader . . . 14
headerFileName . . . 14
isDataLine . . . 15
isMovedFile . . . 15
isSunscanFile . . . 16
isTitleLine . . . 16
movedFileName . . . 17
moveFilesToSubfolders . . . 17
moveOriginalFile . . . 18
readConvertedHeader . . . 18
readIdData . . . 19
reorderDataColumns . . . 19
runSunscanApp . . . 20
saveHeader . . . 20
splitLines . . . 21
stripFileExtension . . . 21
sunscanimport . . . 22
transformToWideFormat . . . 23

Index **24**

`addPlotID` *Adds PlotID column to LAI data*

Description

Information about PlotID is given via additional tab delimited files.

Usage

```
addPlotID(data, path)
```

Arguments

data	data.frame with LAI data
path	of the original data

Value

data.frame with PlotID column

`addPlotIDFromData` *Adds PlotID column to LAI data*

Description

Information about PlotID is given via additional tab delimited files.

Usage

```
addPlotIDFromData(data, ids, mids)
```

Arguments

data	data.frame with LAI data
ids	dataframe with plotids
mids	dataframe with measurementids

Value

data.frame with PlotID column

convertSunScanDirectory *Converts all SunScan data files in a directory (and it's subdirectories)*

Description

Converts all SunScan data files in a directory (and it's subdirectories)

Usage

```
convertSunScanDirectory(
  directory,
  prefix = "",
  extension = ".txt",
  reportscript = ""
)
```

Arguments

directory	to search for data files
prefix	of files to take into accoung
extension	of files to take into account
reportscript	R script that is used to generate a report for each converted file

convertSunscanFile *Converts a SunScan data file*

Description

Converts a SunScan data file

Usage

```
convertSunscanFile(file, inputfolder, outputfolder)
```

Arguments

file	name of file
inputfolder	input folder
outputfolder	output folder

Value

vector with filenames of converted data

countMeasurements *Counts measurements*

Description

Counts measurements

Usage

```
countMeasurements(data)
```

Arguments

data	imported sunscan data

Value

dataframe with number of measurements

createBoxplot *Creates a boxplot for LAI data*

Description

Creates a boxplot for LAI data

Usage

```
createBoxplot(data, deleted = FALSE)
```

Arguments

data	data.frame with LAI data
deleted	include measurements marked as deleted

Value

ggplot graph with boxplot

createFolders *Creates (sub)folders for converted data and reports*

Description

Creates (sub)folders for converted data and reports

Usage

```
createFolders(file, inputfolder, outputfolder)
```

Arguments

file	filename
inputfolder	input folder
outputfolder	output folder

createGridPlot *Plots the PlotNr arranged in a grid*

Description

Plots the PlotNr arranged in a grid

Usage

```
createGridPlot(griddata)
```

Arguments

griddata	grid data

createGridPlotLAI *Plots LAI on a grid*

Description

Plots LAI on a grid

Usage

```
createGridPlotLAI(data, griddata, deleted = FALSE)
```

Arguments

data	imported sunscan data
griddata	grid data
deleted	include measurements marked as deleted

createSeriesInfo *Summarise information about measurement series*

Description

Summarise information about measurement series

Usage

```
createSeriesInfo(data)
```

Arguments

data	imported sunscan data

Value

dataframe

createSummary *Summarises data for each PlotID*

Description

Summarises data for each PlotID

Usage

```
createSummary(data, deleted = FALSE)
```

Arguments

data	data.frame with LAI data
deleted	include measurements marked as deleted

Value

data.frame with summary information

createTimePlot *Plots measurements over time*

Description

Plots measurements over time

Usage

```
createTimePlot(data, stripes = 11)
```

Arguments

data	imported sunscan data
stripes	number of color stripes

createTimePlotLAI *Plots measurements over time*

Description

Plots measurements over time

Usage

```
createTimePlotLAI(data, stripes = 11)
```

Arguments

data	imported sunscan data
stripes	number of color stripes

generateInitialGridData *Creates initial grid data*

Description

Creates initial grid data

Usage

```
generateInitialGridData(data, rows = 1, rowwise = FALSE)
```

Arguments

data	converted sunscan data
rows	number of rows of the field
rowwise	TRUE if the numbering is rowwise

Value

data frame with grids (col/row)

generateReport *Generates report concerning converted data*

Description

If one does not provide a report script, then the default report script is used.

Usage

```
generateReport(
  file,
  inputfolder,
  outputfolder,
 reportscript = system.file("reports", "default_report.R", package = "sunscanimport")
)
```

Arguments

file	filename
inputfolder	input folder
outputfolder	output folder
reportscript	script to generate the report

Details

A report script is an R script that may contain RMarkdown. Inside the script one can use the variable param with its elements param$file, param$inputfolder and param$outputfolder. The report will be placed in the outputfolder in the subdirectory report.

Notice that the package rmarkdown and knitr have to be installed to generate a report.

generateSampleMeasurementIdData *Creates Sample MeasurementID data*

Description

Creates Sample MeasurementID data

Usage

```
generateSampleMeasurementIdData(data, interval = 0)
```

Arguments

data	converted sunscan data
interval	time interval after which a new PlotNR is assumed

Value

data frame with ids

---

generateSamplePlotIdData *Generates sample ID data*

Description

Generates sample data to relate PlotNr to grids, PlotIDs or MeasurementIDs

Usage

```
generateSamplePlotIdData(data)

generateInitialPlotIdData(data)

generateInitialMeasurementIdData(data)
```

Arguments

data	LAI-Data from SunscanFile

Value

dataframe with IDs

---

getData *Extracts LAI data from the file lines*

Description

Extracts LAI data from the file lines

Usage

```
getData(lines, date)
```

Arguments

lines	vector of lines read from the data file
date	start date of measurements from meta data

Value

data.frame with LAI measurements

---

getFileList *Fetches recursively all possible data files from a directory*

Description

Fetches recursively all possible data files from a directory

Usage

```
getFileList(
  directory,
  prefix = "",
  extension = ".TXT",
  excludedir = "converted"
)
```

Arguments

directory	to search for data files
prefix	of files to take into account
extension	of files to take into account
excludedir	subdirectory name to exclude

Value

vector of potential data files

getHeader *Gets meta data from the file header*

Description

Gets meta data from the file header

Usage

```
getHeader(lines, path)
```

Arguments

lines	vector of lines read from the data file
path	path of the file

Value

named vector with meta data

getParData *Extracts PAR data from the file lines*

Description

Extracts PAR data from the file lines

Usage

```
getParData(lines, date)
```

Arguments

lines	vector of lines read from the data file
date	start date of measurements from meta data

Value

data.frame with PAR measurements

getSmallHeader *Gets meta data from the file header*

Description

Gets meta data from the file header

Usage

```
getSmallHeader(lines)
```

Arguments

lines	vector of lines read from the data file

Value

named vector with meta data

headerFileName *File name for the metadata file*

Description

File name for the metadata file

File name for the LAI data file

File name for the PAR data file

File name for the summary file

File name for the boxplot image

File name for the report

Usage

```
headerFileName(file, outputfolder)

dataFileName(file, outputfolder)

pardataFileName(file, outputfolder)

summaryFileName(file, outputfolder)

boxplotFileName(file, outputfolder)

reportFileName(file, outputfolder)
```

Arguments

file	filename of data file
outputfolder	output folder

Value

filename

---

isDataLine *Checks if the line is a data line*

Description

Data lines begin with a date

Usage

```
isDataLine(line)
```

Arguments

line	a line of text

Value

TRUE if data line

---

isMovedFile *Checks if the data file has been moved to subfolder 'original'*

Description

Checks if the data file has been moved to subfolder 'original'

Usage

```
isMovedFile(file, inputfolder, target = "original")
```

Arguments

file	filename
inputfolder	input folder
target	target folder

Value

TRUE if file has been moved to original folder

isSunscanFile *Checks if given file is a valid SunScan file*

Description

Checks if file starts with 'Created by SunData' and if the files has at least 14 lines

Usage

```
isSunscanFile(path)
```

Arguments

path	path to the file

Value

TRUE if valid SunScan file

isTitleLine *Checks if the line is a title line*

Description

A title line starts a new metatdata section in the sunscan data file.

Usage

```
isTitleLine(line)
```

Arguments

line	a line of text

Value

TRUE if title line

movedFileName *Returns path for the data file moved to 'original' subfolder*

Description

Returns path for the data file moved to 'original' subfolder

Usage

```
movedFileName(file, inputfolder, target = "original")
```

Arguments

file	filename
inputfolder	input folder
target	target folder

Value

path to data file

moveFilesToSubfolders *Moves data file to subfolder 'original'*

Description

Moves data file to subfolder 'original'

Usage

```
moveFilesToSubfolders(file, inputfolder, target = "original")
```

Arguments

file	filename
inputfolder	input folder
target	target folder

moveOriginalFile *Moves data file to subfolder 'original'*

Description

Moves data file to subfolder 'original'

Usage

```
moveOriginalFile(file, inputfolder, target = "original")
```

Arguments

file	filename
inputfolder	input folder
target	target folder

readConvertedHeader *Reads converted data*

Description

Reads converted data

Usage

```
readConvertedHeader(file, outputfolder)

readConvertedData(file, outputfolder)

readConvertedParData(file, outputfolder)

readConvertedSummary(file, outputfolder)

readConvertedFiles(file, outputfolder)
```

Arguments

file	filename
outputfolder	output folder

Value

dataframe with converted data

readIdData *Read dataframe with PlotID column*

Description

Read dataframe with PlotID column

Usage

```
readIdData(file)
```

Arguments

file	filename

Value

data frame with the ids

reorderDataColumns *Reorders columns in converted data*

Description

Reorders columns in converted data

Usage

```
reorderDataColumns(data)
```

Arguments

data	imported sunscan data

Value

dataframe with reordered columns

runSunscanApp *Runs Shiny app*

Description

Runs Shiny app

Usage

```
runSunscanApp(...)
```

Arguments

...	parameters passed to shiny::runApp

saveHeader *Saves metadata to file*

Description

Saves metadata to file

Saves LAI data

Saves PAR data

Saves summary

Saves boxplot

Usage

```
saveHeader(data, file, outputfolder)

saveData(data, file, outputfolder)

saveParData(data, file, outputfolder)

saveSummary(data, file, outputfolder)

saveBoxplot(data, file, outputfolder)
```

Arguments

data	data frame
file	filename of original file
outputfolder	output folder

splitLines *Splits the data when file contains multiple headers*

Description

Splits the data when file contains multiple headers

Usage

```
splitLines(lines)
```

Arguments

lines	vector of lines from the data file

Value

list of line vectors

stripFileExtension *Strips the file extension*

Description

Strips the file extension

Usage

```
stripFileExtension(file)
```

Arguments

file	filename

Value

file name without path and extension

sunscanimport *Functions to import, convert and visualise data from LAI Sunscan device*

Description

The package provides functions to:

- convert a file or a directory
- summarize or transform converted data
- create reports for converted data
- run a shiny app for interactive conversion

Author(s)

Gunther Krauss

Examples

```
## Not run:
  runSunscanApp()
## End(Not run)

## Not run:
  file <- "paulinenaue.TXT"
  inputfolder <-  "data/210908/original/"
  outputfolder <- "data/210908/"
  convfile <- convertSunscanFile(file,inputfolder, outputfolder)
  generateReport(convfile, inputfolder, outputfolder)
## End(Not run)

## Not run:
  convertSunScanDirectory("data/")
## End(Not run)

## Not run:
  data <- readr::read_delim("data/210908/converted/data_paulinenaue.txt", delim="\t")
  data_summary <- createSummary(data)
  data_wide <- transformToWideFormat(data)
## End(Not run)
```

`transformToWideFormat` *Transform LAI data to wide format (1 row per PlotID)*

Description

Each PlotID has only one row where each individual LAI measurements is in an own column. Number of columns is determined by the maxiumum number of measurements per PlotID. Measurements are populated from left to right and filled up by NAs.

Usage

```
transformToWideFormat(data, deleted = FALSE)
```

Arguments

`data`	data.frame with LAI data
`deleted`	include measurements marked as deleted

Value

data.frame in wide format

Index

addPlotID, 2
addPlotIDFromData, 3

boxplotFileName *(*headerFileName*)*, 14

convertSunScanDirectory, 4
convertSunscanFile, 4
countMeasurements, 5
createBoxplot, 5
createFolders, 6
createGridPlot, 6
createGridPlotLAI, 7
createSeriesInfo, 7
createSummary, 8
createTimePlot, 8
createTimePlotLAI, 9

dataFileName *(*headerFileName*)*, 14

generateInitialGridData, 9
generateInitialMeasurementIdData
*(*generateSamplePlotIdData*)*, 11
generateInitialPlotIdData
*(*generateSamplePlotIdData*)*, 11
generateReport, 10
generateSampleMeasurementIdData, 10
generateSamplePlotIdData, 11
getData, 11
getFileList, 12
getHeader, 13
getParData, 13
getSmallHeader, 14

headerFileName, 14

isDataLine, 15
isMovedFile, 15
isSunscanFile, 16
isTitleLine, 16

movedFileName, 17
moveFilesToSubfolders, 17
moveOriginalFile, 18

pardataFileName *(*headerFileName*)*, 14

readConvertedData
*(*readConvertedHeader*)*, 18
readConvertedFiles
*(*readConvertedHeader*)*, 18
readConvertedHeader, 18
readConvertedParData
*(*readConvertedHeader*)*, 18
readConvertedSummary
*(*readConvertedHeader*)*, 18
readIdData, 19
reorderDataColumns, 19
reportFileName *(*headerFileName*)*, 14
runSunscanApp, 20

saveBoxplot *(*saveHeader*)*, 20
saveData *(*saveHeader*)*, 20
saveHeader, 20
saveParData *(*saveHeader*)*, 20
saveSummary *(*saveHeader*)*, 20
splitLines, 21
stripFileExtension, 21
summaryFileName *(*headerFileName*)*, 14
sunscanimport, 22
sunscanimport-package *(*sunscanimport*)*,
22

transformToWideFormat, 23