Package 'greenfeedr'

May 8, 2026

Type Package

Title Process and Report 'GreenFeed' Data

Version 1.3.1

Maintainer Guillermo Martinez-Boggio <gmartinezboggio@ucdavis.edu>

Description
Provides tools for downloading, processing, and reporting daily and finalized 'GreenFeed' data.

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.3.2

Imports dplyr, ggplot2, httr, lubridate, purrr, readr, readxl, rmarkdown, shiny, stats, stringr, tidyr

Depends R (>= 2.10)

Suggests knitr, testthat (>= 3.0.0)

Config/testthat/edition 3

BugReports https://github.com/GMBog/greenfeedr/issues

URL https://github.com/GMBog/greenfeedr

VignetteBuilder knitr

NeedsCompilation no

Author Guillermo Martinez-Boggio [cre, aut, cph] (ORCID: <https://orcid.org/0000-0003-3597-9426>), Meredith Harrison [ctb], Patrick Lutz [ctb]

Repository CRAN

Date/Publication 2025-09-05 19:50:02 UTC

Contents

`compare_gfdata` *Compare Preliminary and Finalized 'GreenFeed' Data*

Description

Compare preliminary and finalized 'GreenFeed' data.

Arguments

`prelimrep`	a data frame with preliminary 'GreenFeed' data
`finalrep`	a data frame with finalized 'GreenFeed' data
`start_date`	a character string representing the start date of the study (format: "DD-MM-YY" or "YYYY-MM-DD")
`end_date`	a character string representing the end date of the study (format: "DD-MM-YY" or "YYYY-MM-DD")

Value

Data frame including records removed from preliminary and final reports

Examples

```
# Datasets with preliminary and finalized GreenFeed data
prelimrep <- system.file("extdata", "StudyName_GFdata.csv", package = "greenfeedr")
finalrep <- system.file("extdata", "StudyName_FinalReport.xlsx", package = "greenfeedr")

data <- compare_gfdata(
  prelimrep,
  finalrep,
  start_date = "2024-05-13",
  end_date = "2024-05-20"
)
```

get_gfdata *Download Preliminary and Raw 'GreenFeed' Data via 'API'*

Description

Downloads preliminary and raw 'GreenFeed' data from the 'C-Lock Inc.' server via an 'API'. Retrieves data based on specified parameters (login, date range, and units), and provides a CSV file with the 'GreenFeed' preliminary data.

Usage

```
get_gfdata(
  user,
  pass,
  d = "visits",
  type = 2,
  exp = NA,
  unit,
  start_date,
  end_date = Sys.Date(),
  save_dir = tempdir()
)
```

Arguments

user	a character string representing the user name to logging into 'GreenFeed' system
pass	a character string representing password to logging into 'GreenFeed' system
d	a character string representing data to download (opts: "visits", "feed", "rfid", "cmds")
type	a numeric representing the type of data to retrieve (1=finalized and 2=preliminary)
exp	a character string representing study name or other study identifier. It is used as file name to save the data
unit	numeric or character vector, or a list representing one or more 'GreenFeed' unit numbers
start_date	a character string representing the start date of the study (format: "DD-MM-YY" or "YYYY-MM-DD")
end_date	a character string representing the end date of the study (format: "DD-MM-YY" or "YYYY-MM-DD")
save_dir	a character string representing the directory to save the output file

Value

A CSV file with the specified data (visits or raw) saved in the provided directory.

Examples

```
# Please replace "your_username" and "your_password" with your actual 'GreenFeed' credentials.
# By default, the function downloads the preliminary 'GreenFeed' data,
# if raw data is needed use options: "feed", "rfid", or "cmds"

get_gfdata(
  user = "your_username",
  pass = "your_password",
  d = "visits",
  type = 2,
  exp = "StudyName",
  unit = c(304, 305),
  start_date = "2024-01-01",
  end_date = Sys.Date(),
  save_dir = tempdir()
)
```

`pellin` *Process 'GreenFeed' Pellet Intakes*

Description

Processes 'feedtimes' file from 'GreenFeed' system. Food drops are used to calculate pellet intakes per animal. Aggregates data to provide insights into the feeding behavior and pellet consumption of the animals during a study.

Arguments

`user`	a character string representing the user name to logging into 'GreenFeed' system
`pass`	a character string representing password to logging into 'GreenFeed' system
`unit`	numeric or character vector or list representing one or more 'GreenFeed' unit numbers. The order should match with "feedtimes" files
`gcup`	a numeric value representing the grams of pellets per cup. If dual-hopper you can define multiple grams per unit
`start_date`	a character string representing the start date of the study (format: "DD-MM-YY" or "YYYY-MM-DD")
`end_date`	a character string representing the end date of the study (format: "DD-MM-YY" or "YYYY-MM-DD")
`save_dir`	a character string representing the directory to save the output file
`rfid_file`	a character string representing the file with individual IDs. The order should be Visual ID (col1) and RFID (col2)
`file_path`	a character string or list representing files(s) with "feedtimes" from 'C-Lock Inc.'

Value

An Excel file with pellet intakes for all animals and days within the specified period is saved to save_dir. The file is named "Pellet_Intakes_YYYY-MM-DD_YYYY-MM-DD.csv".

Examples

```
# You should provide the 'feedtimes' file provided by C-Lock.
# it could be a list of files if you have data from multiple units to combine
path <- system.file("extdata", "feedtimes.csv", package = "greenfeedr")

# You must include the grams of pellets per cup based on the result obtained from the 10-drops test

# If the user include an rfid file, the structure should be in col1 AnimalName or Visual ID, and
# col2 the RFID or TAG_ID. The file could be save in different formats (.xlsx, .csv, or .txt).
RFIDs <- system.file("extdata", "RFID_file.csv", package = "greenfeedr")

pellin(
  unit = 1,
  gcup = 34,
  start_date = "2024-05-13",
  end_date = "2024-05-25",
  save_dir = tempdir(),
  rfid_file = RFIDs,
  file_path = path
)
```

---

process_gfdata *Process Preliminary and Finalized 'GreenFeed' Data*

Description

Processes and calculates daily and weekly averages of 'GreenFeed' data. Handles data filtering, aggregation, and summarization to facilitate further analysis.

Arguments

data	a data frame with preliminary or finalized 'GreenFeed' data
start_date	a character string representing the start date of the study (format: "DD-MM-YY" or "YYYY-MM-DD")
end_date	a character string representing the end date of the study (format: "DD-MM-YY" or "YYYY-MM-DD")
param1	an integer representing the number of records per day to be consider for analysis
param2	an integer representing the number of days with records per week to be consider for analysis
min_time	an integer representing the minimum number of minutes for a records to be consider for analysis (default: 2 minutes)

transform	A logical representing whether to transform gas production to L/d. If TRUE, gas will be converted from grams/day to liters/day.
cutoff	an integer specifying the range for identifying outliers (default: 3 SD)

Value

A list of three data frames:

filtered_data	data frame with filtered 'GreenFeed' data
daily_data	data frame with daily processed 'GreenFeed' data
weekly_data	data frame with weekly processed 'GreenFeed' data

Examples

```
file <- system.file("extdata", "StudyName_GFdata.csv", package = "greenfeedr")
datafile <- readr::read_csv(file)

gf_data <- process_gfdata(
  data = datafile,
  start_date = "2024-05-13",
  end_date = "2024-05-25",
  param1 = 2,
  param2 = 3,
  min_time = 2
)
head(gf_data)
```

---

report_gfdata *Download and Report 'GreenFeed' Data*

Description

Generates a PDF report of preliminary and finalized 'GreenFeed' data. The report includes: number of animals using 'GreenFeed' and plots with distribution of records and gas production. If the preliminary option is used, the data is retrieved from the 'C-Lock Inc.' server through an 'API', otherwise the data processed by 'C-Lock Inc.' must be provided to generate the report.

Arguments

input_type	a character string representing type of data (options: "preliminary" and "finalized")
exp	a character string representing study name or other study identifier. It is used as file name to save the data
unit	numeric or character vector, or a list representing one or more 'GreenFeed' unit numbers

start_date	a character string representing the start date of the study (format: "DD-MM-YY" or "YYYY-MM-DD")
end_date	a character string representing the end date of the study (format: "DD-MM-YY" or "YYYY-MM-DD")
save_dir	a character string representing the directory to save the output file
plot_opt	a character string representing the gas(es) to plot (options: "All", "CH4", "CO2", "O2", "H2")
rfid_file	a character string representing the file with individual IDs. The order should be Visual ID (col1) and RFID (col2)
user	a character string representing the user name to logging into 'GreenFeed' system. If input_type is "final", this parameter is ignored
pass	a character string representing password to logging into 'GreenFeed' system. If input_type is "final", this parameter is ignored
file_path	A list of file paths containing the final report(s) from the 'GreenFeed' system. If input_type is "prelim", this parameter is ignored

Value

A CSV file with preliminary 'GreenFeed' data and a PDF report with a description of the preliminary or finalized data

Examples

```
# Please replace "your_username" and "your_password" with your actual 'GreenFeed' credentials.
# The data range must be fewer than 180 days
# Example without rfid_file (by default NA)

report_gfdata(
  user = "your_username",
  pass = "your_password",
  input_type = "preliminary",
  exp = "StudyName",
  unit = 1,
  start_date = "2023-01-01",
  end_date = Sys.Date(),
  save_dir = tempdir(),
  plot_opt = "All"
)
```

---

run_app *Launch the Greenfeedr Shiny App locally*

Description

This function launches the Greenfeedr Shiny app on your computer.

Usage

```
run_app()
```

---

run_gfapp *Run the 'greenfeedr' Shiny App locally*

Description

Launches the 'greenfeedr' Shiny application on your computer. The app provides an interactive interface for 'GreenFeed' data analysis, visualization, and reporting.

Usage

```
run_gfapp()
```

Value

This function launches the Shiny app in your default web browser; it does not return a value.

Examples

```
## Not run:
  greenfeedr::run_gfapp()

## End(Not run)
```

---

viseat *Process 'GreenFeed' Visits*

Description

Processes 'GreenFeed' visits and food drops for a requested period. Generates a list of animals not visiting the 'GreenFeed' to manage them, and a description of animals visiting the 'GreenFeed'.

Arguments

user	a character string representing the user name to logging into 'GreenFeed' system
pass	a character string representing password to logging into 'GreenFeed' system
unit	numeric or character vector or list representing one or more GreenFeed unit numbers.
start_date	a character string representing the start date of the study (format: "DD-MM-YY" or "YYYY-MM-DD")
end_date	a character string representing the end date of the study (format: "DD-MM-YY" or "YYYY-MM-DD")
rfid_file	a character string representing the file with individual RFIDs. The order should be Visual ID (col1) and RFID (col2)
file_path	a character string or list representing files(s) with feedtimes from 'C-Lock Inc.'

Value

A list of two data frames:

`visits_per_day` Data frame with daily processed 'GreenFeed' data, including columns for VisualID, Date, Time, number of drops, and visits.

`visits_per_animal` Data frame with weekly processed 'GreenFeed' data, including columns for VisualID, total drops, total visits, mean drops, and mean visits.

Examples

```
# You should provide the feedtimes files.
# it could be a list of files if you have data from multiple units to combine
path <- system.file("extdata", "feedtimes.csv", package = "greenfeedr")

# If the user include an rfid file, the structure should be in col1 AnimalName or VisualID, and
# col2 the RFID or TAG_ID. The file could be save in different formats (.xlsx, .csv, or .txt).
RFIDs <- system.file("extdata", "RFID_file.csv", package = "greenfeedr")

data <- viseat(
  file_path = path,
  unit = 1,
  start_date = "2024-05-13",
  end_date = "2024-05-25",
  rfid_file = RFIDs
)
```

Index