

# Package: greenfeedr (via r-universe)

October 22, 2024

**Type** Package

**Title** Process and Report 'GreenFeed' Data

**Version** 1.0.0.9000

**Maintainer** Guillermo Martinez-Boggio

<guillermo.martinezboggio@wisc.edu>

**Description** Downloading, Processing, and Reporting 'GreenFeed' Data.

The 'GreenFeed' system generates substantial amounts of daily data, which can be overwhelming for users. This package facilitates data retrieval and reporting from the 'GreenFeed' system using a streamlined process. Users can retrieve data from the 'C-Lock Inc.' system via an 'API' using the `get_gfdata()` function. Additionally, the `report_gfdata()` function allows users to generate both daily and final reports from 'GreenFeed' data. The `process_gfdata()` function further processes this data for analysis. For pellet intake processing and visit checks, the package also provides the `pellin()` and `viseat()` functions, ensuring a comprehensive toolset for managing and analyzing 'GreenFeed' data.

**License** GPL (>= 3)

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.3.2

**Imports** dplyr, ggplot2, httr, lubridate, purrr, readr, readxl, rmarkdown, stats, stringr, tidy

**Depends** R (>= 2.10)

**Suggests** knitr, testthat (>= 3.0.0)

**Config/testthat/edition** 3

**BugReports** <https://github.com/GMBog/greenfeedr/issues>

**URL** <https://gmbog.github.io/greenfeedr/>,  
<https://github.com/GMBog/greenfeedr>

**VignetteBuilder** knitr

**Repository** <https://gmbog.r-universe.dev>

**RemoteUrl** <https://github.com/gmbog/greenfeedr>

**RemoteRef** HEAD

**RemoteSha** 79aa4ae119ed1bbe9e44d75533c6d51e4c400155

## Contents

get_gfdata . . . . .	2
pellin . . . . .	3
process_gfdata . . . . .	4
report_gfdata . . . . .	5
viseat . . . . .	6

<b>Index</b>	<b>8</b>
--------------	----------

---

get_gfdata	<i>Download Daily 'GreenFeed' Data via 'API'</i>
------------	--

---

## Description

Downloads daily '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' daily data.

## Usage

```
get_gfdata(
  user,
  pass,
  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
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: "mm/dd/yyyy")  
 end\_date a character string representing the end date of the study (format: "mm/dd/yyyy")  
 save\_dir a character string representing the directory to save the output file

**Value**

A CSV file with daily 'GreenFeed' data in the specified directory

**Examples**

```
# Please replace "your_username" and "your_password" with your actual 'GreenFeed' credentials.
user <- Sys.getenv("API_USER")
pass <- Sys.getenv("API_PASS")
exp <- "StudyName"
start_date <- "2024-01-01"
end_date <- Sys.Date()
save_dir <- tempdir()

# Example with units as a vector
unit <- c(304, 305)
get_gfdata(user, pass, exp, unit, start_date, end_date, save_dir)
```

---

pellin

*Process 'GreenFeed' Pellet Intakes*

---

**Description**

Processes the "feedtimes" file from 'GreenFeed' system, including visits and food drops across a specific period, and it is used to calculate pellet intakes per animal from all units. Aggregates data to provide insights into the feeding behavior and pellet consumption of the animals during a study.

**Arguments**

file\_path a character string or list representing files(s) with "feedtimes" from 'C-Lock Inc.'

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.

start\_date a character string representing the start date of the study (format: "mm/dd/yyyy")

end\_date a character string representing the end date of the study (format: "mm/dd/yyyy")

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)

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

---

process\_gfdata

*Process Daily and Final '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 daily or finalized 'GreenFeed' data
start_date	a character string representing the start date of the study (format: "mm/dd/yyyy")
end_date	a character string representing the end date of the study (format: "mm/dd/yyyy")
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. By default min_time is 2

**Value**

A list of two data frames:

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 PDF reports of daily and final 'GreenFeed' data. If the option daily is used, data is retrieved from 'C-Lock Inc.' server via an 'API' and generates a PDF report to with number of animals, records, and gases production. However, if the option final is used, the finalized data should be provided to generates a PDF report to evaluate all 'GreenFeed' data obtained from the finalized study.

**Arguments**

input_type	a character string representing type of data (options: "daily" and "final")
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: "mm/dd/yyyy")
end_date	a character string representing the end date of the study (format: "mm/dd/yyyy")
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 "final", this parameter is ignored

**Value**

A CSV file with daily 'GreenFeed' data and a PDF report with a description of the daily or final records

**Examples**

```
# Please replace "your_username" and "your_password" with your actual 'GreenFeed' credentials.
user <- Sys.getenv("API_USER")
pass <- Sys.getenv("API_PASS")

# The data range must be fewer than 180 days
# Example without rfid_file (by default NA)

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

---

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**

file_path	a character string or list representing files(s) with feedtimes from 'C-Lock Inc.'.
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: "mm/dd/yyyy")  
`end_date` a character string representing the end date of the study (format: "mm/dd/yyyy")  
`rfid_file` a character string representing the file with individual RFIDs. The order should be Visual ID (col1) and RFID (col2)

### Value

A list of two data frames:

`visits_per_unit`

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 <- list(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

`get_gfdata`, 2

`pellin`, 3

`process_gfdata`, 4

`report_gfdata`, 5

`viseat`, 6