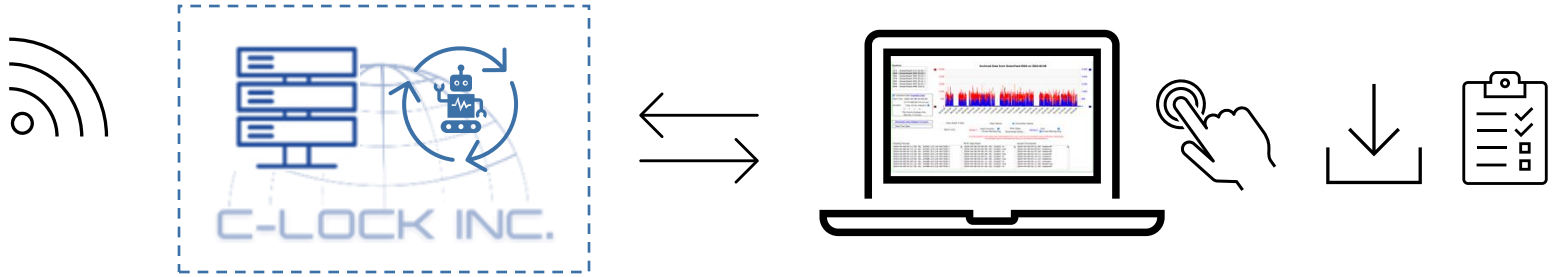
Revolutionize GreenFeed Data Management with the greenfeedr R-package

Guillermo Martinez Boggio

WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

GreenFeed

System communication and data management

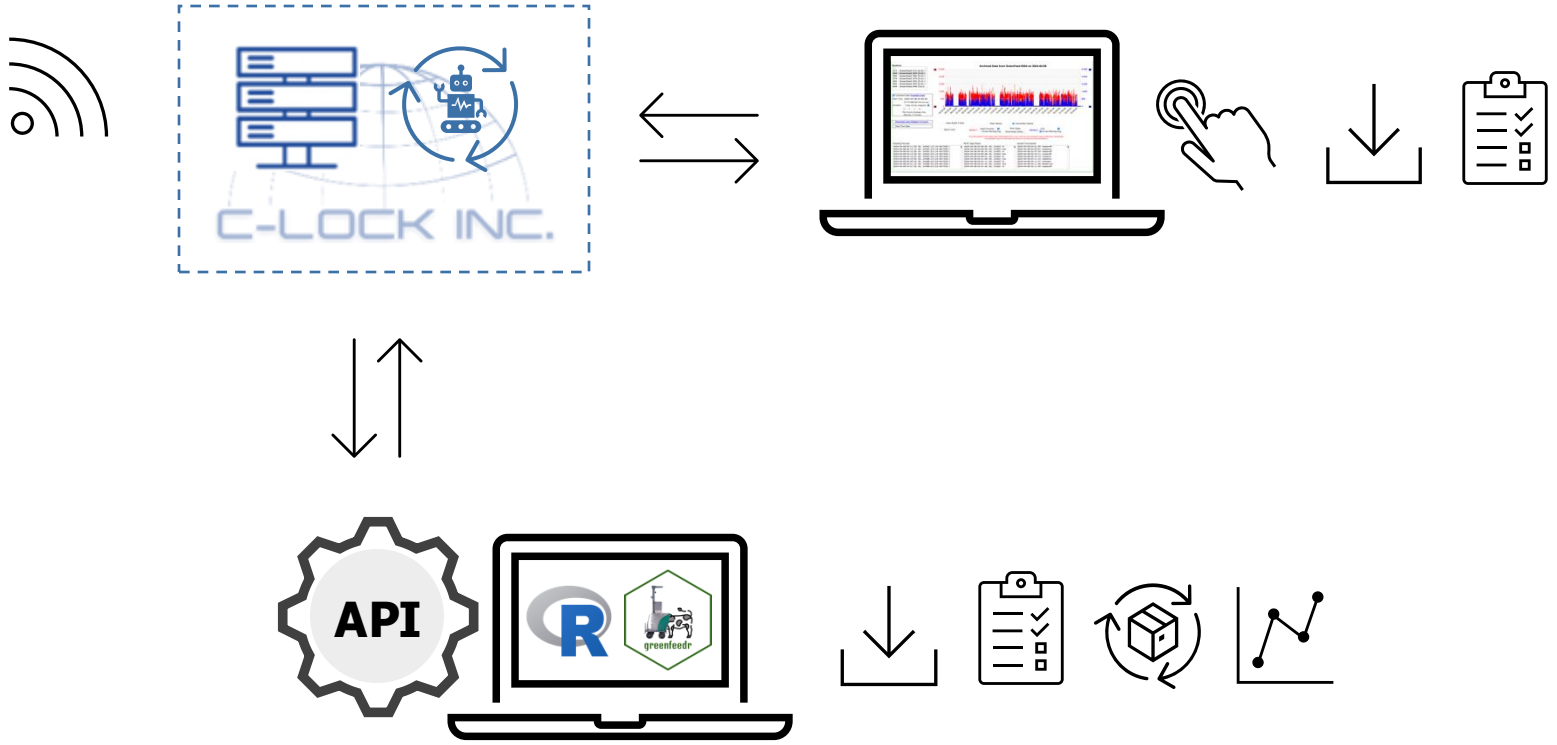


"do as little as possible by hand and as much as possible with functions"

Wickham and Bryan, 2023

GreenFeed

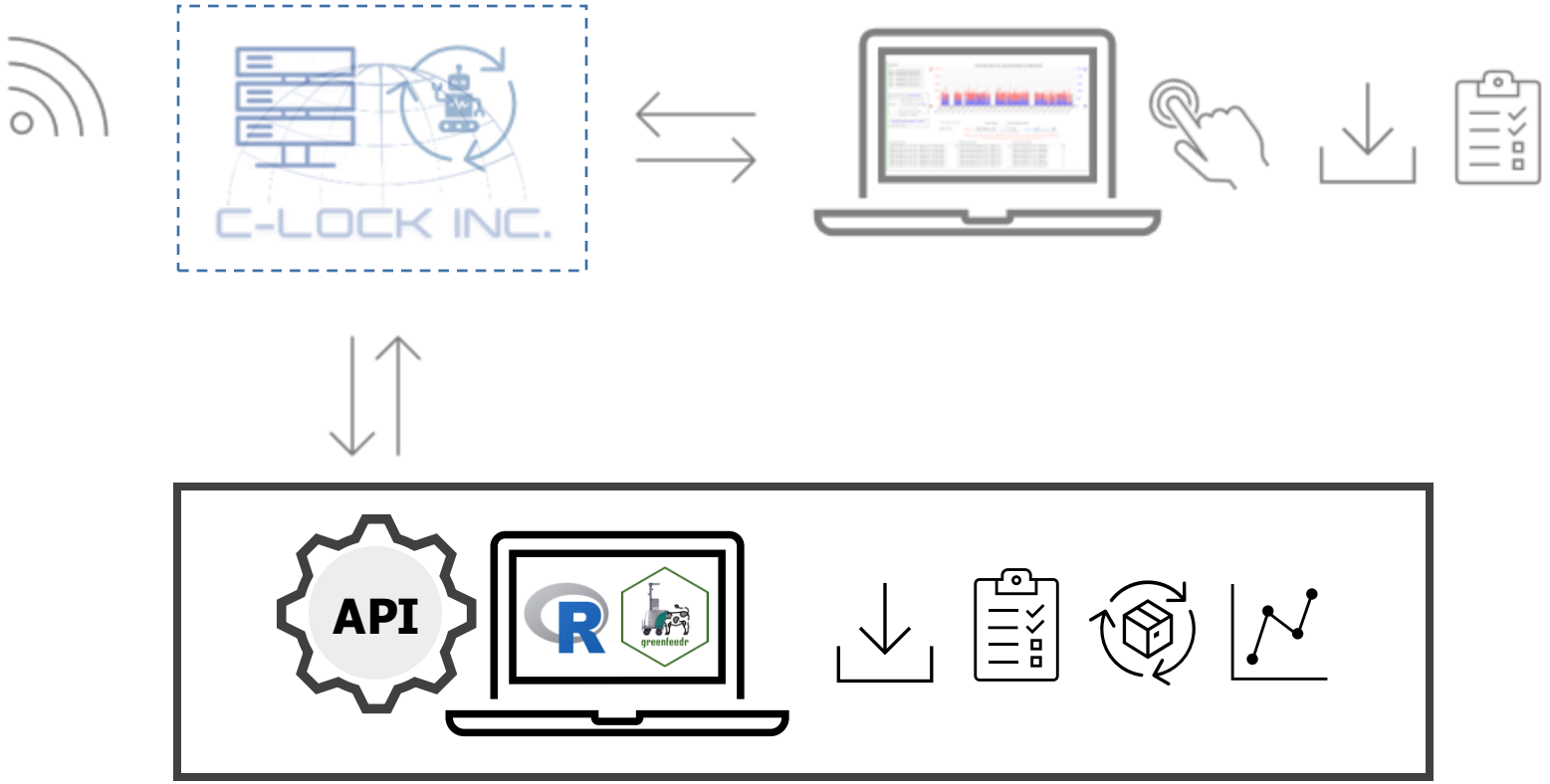
System communication and data management



An **API** (Application Programming Interface) is a set of rules that allows software applications to communicate with each other

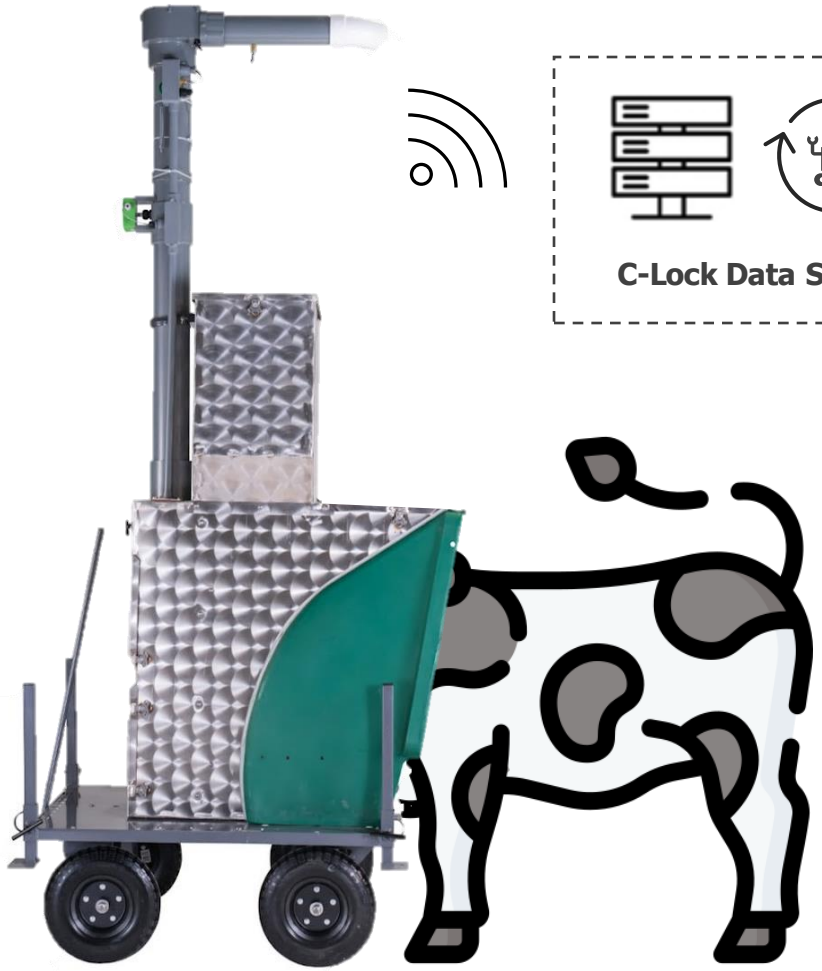
GreenFeed

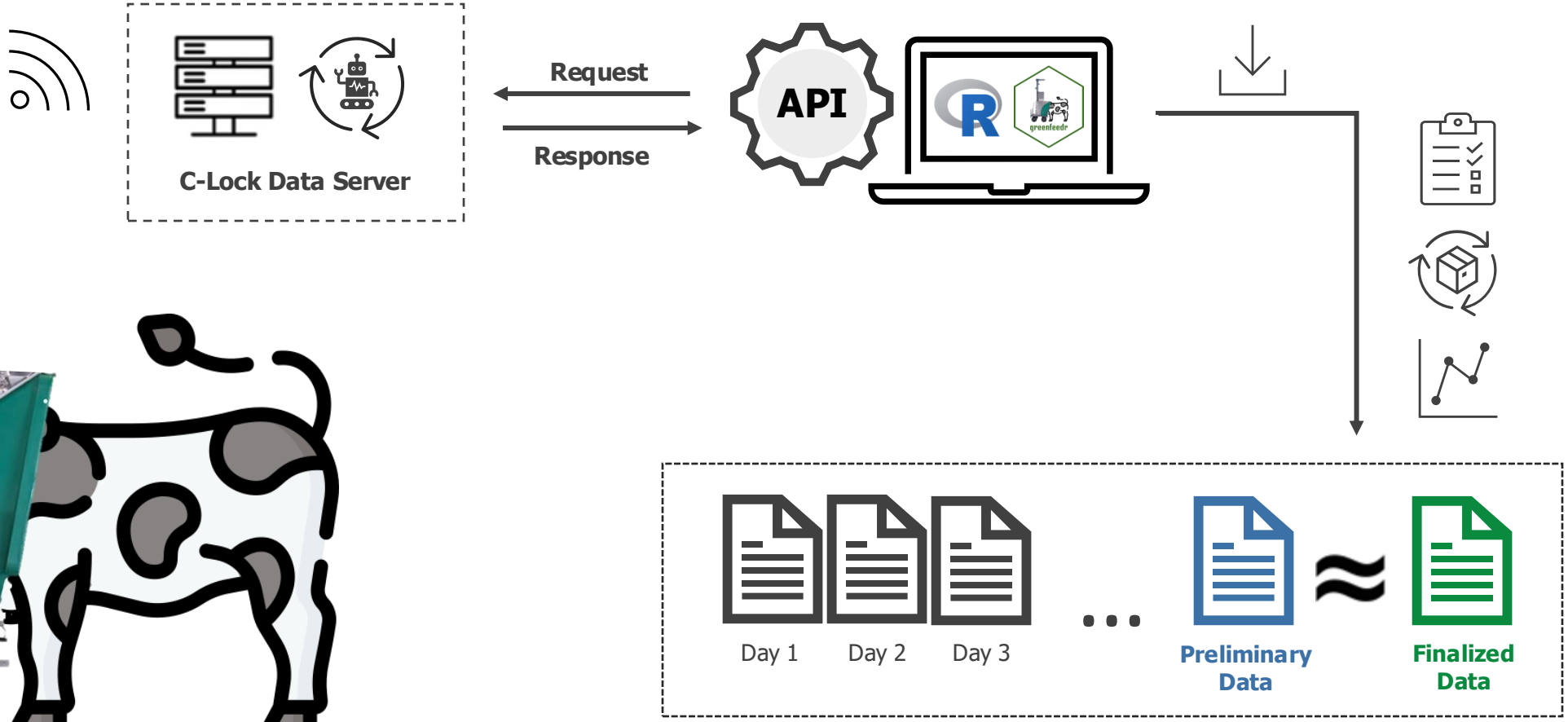
System communication and data management



greenfeedr

An R Package for Processing & Reporting GreenFeed Data





greenfeedr
Version 1.1.0

get_gfdata

Parameters:
- Login user
- Login password
- Data type
- Experiment name
- Unit(s)
- Start date
- End date
- Save directory

Return:
- Excel file with requested data

report_gfdata

Parameters:
- Login user
- Login password
- Experiment name
- Unit(s)
- Start date
- End date
- Input type
- Save directory
- Plot option
- Rfid file
- Final data file path

Return:
- A PDF report with data description

process_gfdata

Parameters:
- Data
- Start date
- End date
- Param 1 (records/day)
- Param 2 (days/week)
- Minimum time
- Outlier cutoff

Return:
- List with 3 data frames that contains filtered data, daily and weekly averages

compare_gfdata

Parameters:
- Preliminary data
- Finalized data
- Start date
- End date

Return:
- Data frame with removed records

pellin

Parameters:
- Login user
- Login password
- Unit(s)
- Grams per cup(s)
- Start date
- End date
- Save directory
- Rfid file
- Feedtimes file path(s)

Return:
- Excel file with pellet intakes for all animals

viseat

Parameters:
- Login user
- Login password
- Unit(s)
- Start date
- End date
- Rfid file
- Feedtimes file path(s)

Return:
- List with 2 data frames that contains visits per unit and per animal

Martinez-Boggio et al. (2025) Technical Note JDS Communications

Question 1

Have you used greenfeedr before?

Installation

greenfeedr R Package

Released version: greenfeedr 1.1.0

install.packages("greenfeedr")

https://cran.r-project.org/web/packages/greenfeedr/index.html

Development version: greenfeedr 1.1.0.999

install.packages("remotes")
remotes::install_github("GMBog/greenfeedr")

https://github.com/GMBog/greenfeedr

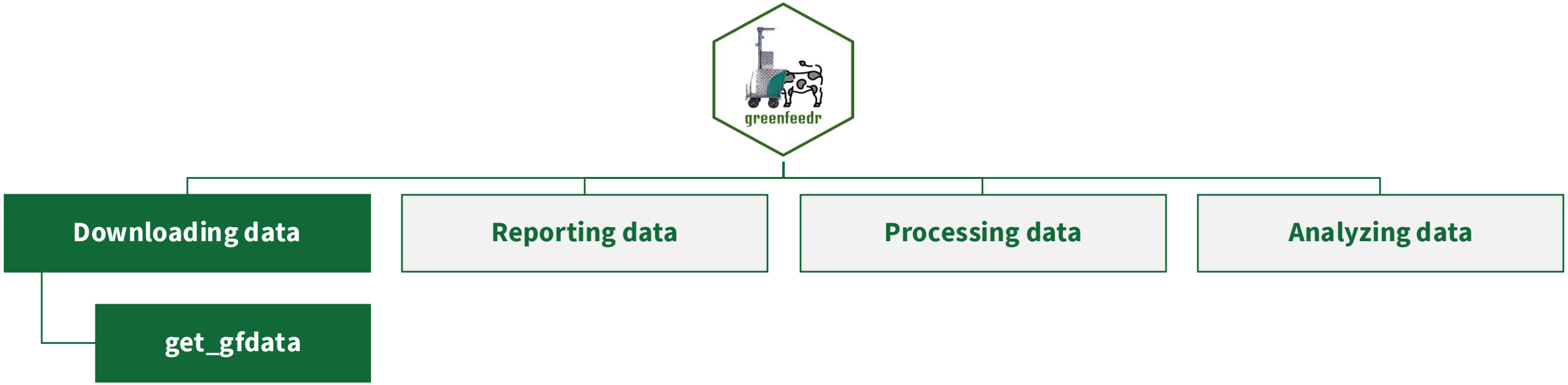
GreenFeed data management

What can greenfeedr be used for?

- Downloading data
- Reporting data
- Processing data
- Analyzing data

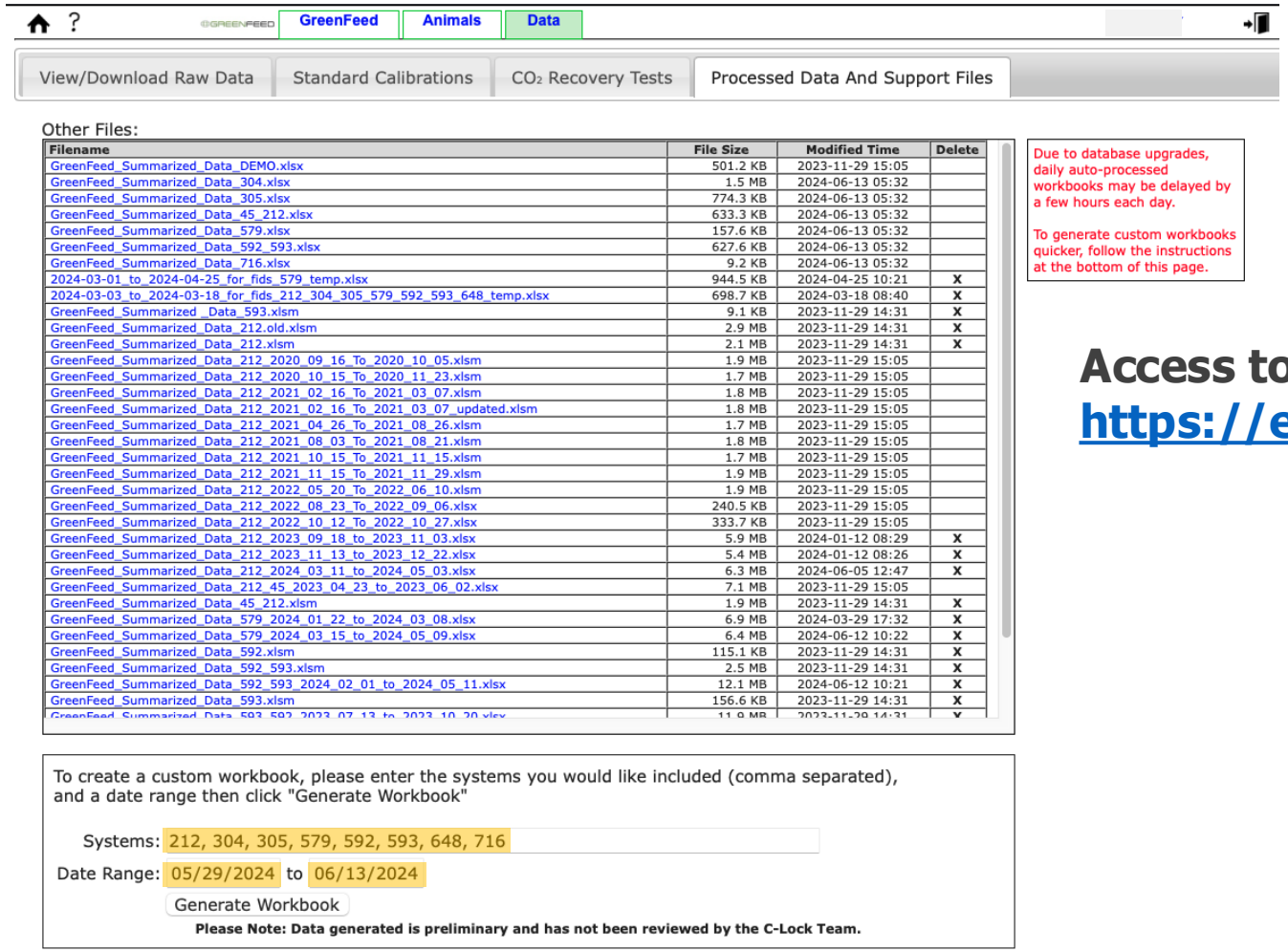
GreenFeed data management

What can greenfeedr be used for?



Downloading data

Preliminary and finalized data

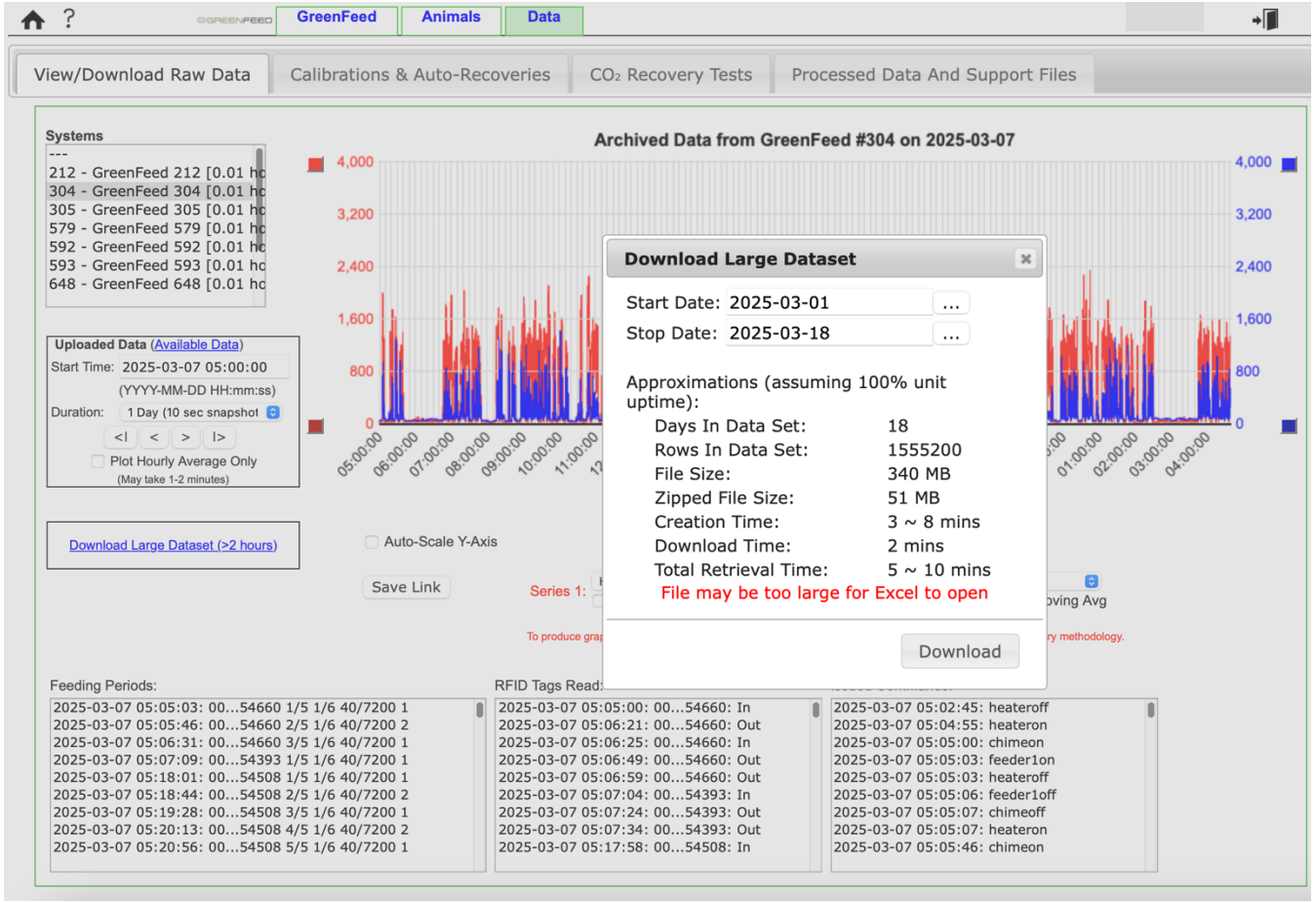


Access to:
https://ext.c-lockinc.com/greenfeed/data.php

Downloading data

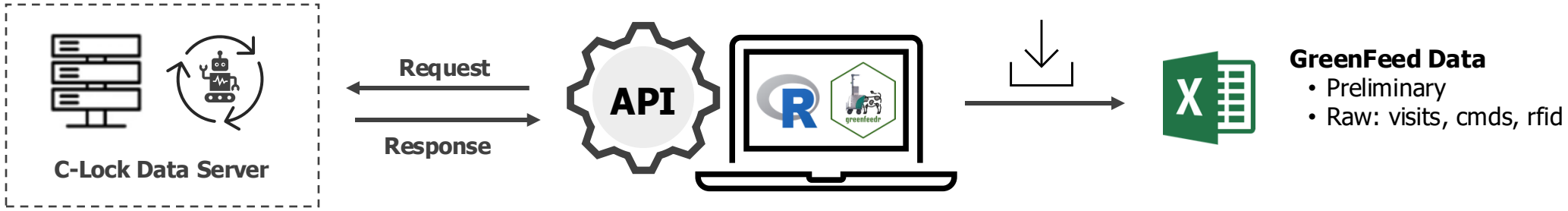
Visits, commands, and rfids

Access to:
https://ext.c-lockinc.com/greenfeed/data.php



Downloading data

How can you automate the process?

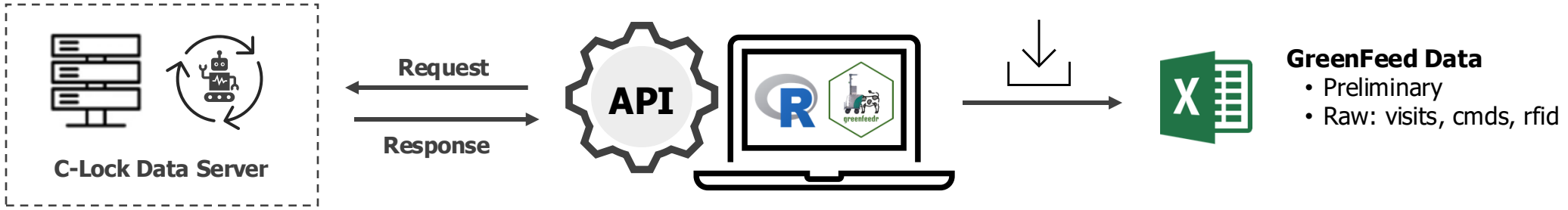


GreenFeed Data

- Preliminary
- Raw: visits, cmds, rfid

Downloading data

How can you automate the process?



get_gfdata(user = **USERNAME**, pass = **PASSWORD**, d = **DATA_TYPE**, exp = **NAME**, unit = **FID**, start_date = **MM/DD/YY**, end_date = **MM/DD/YY**, save_dir = **/DIR/)**

report_gfdata(input_type = **PRELIM/FINAL**, exp = **NAME**, unit = **FID**, start_date = **MM/DD/YY**, end_date = **MM/DD/YY**, save_dir = **/DIR/,** plot_opt = **ALL|CH4|CO2|O2|H2**, rfid_file = **/FILE_PATH/DATA**, user = **USERNAME**, pass = **PASSWORD)**

Downloading data
Function: get_gfdata

@description Downloads preliminary and raw GreenFeed data from C-Lock server via API

get_gfdata(

user = **“GUILLERMO”**,

pass = **“GF1992”**,

d = **“visits”**, # “feed”, “rfid”, “cmds”

exp = **“Exp01”**,

unit = **305**, # “304,305”, c(304,305), list(304,305)

start_date = **“02/20/2025”**, # “2025-02-20”, “2/20/25”

end_date = **“03/17/2025”**,

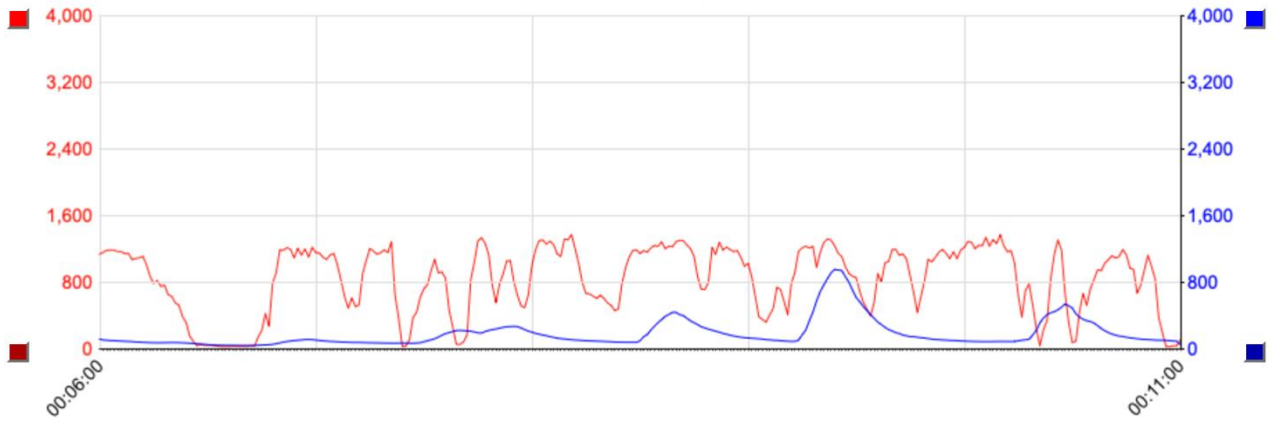
save_dir = **“/Downloads/”**

)

@return A CSV file with the specified data (visits, feed, rfids, or commands) saved in the provided directory

GreenFeed data

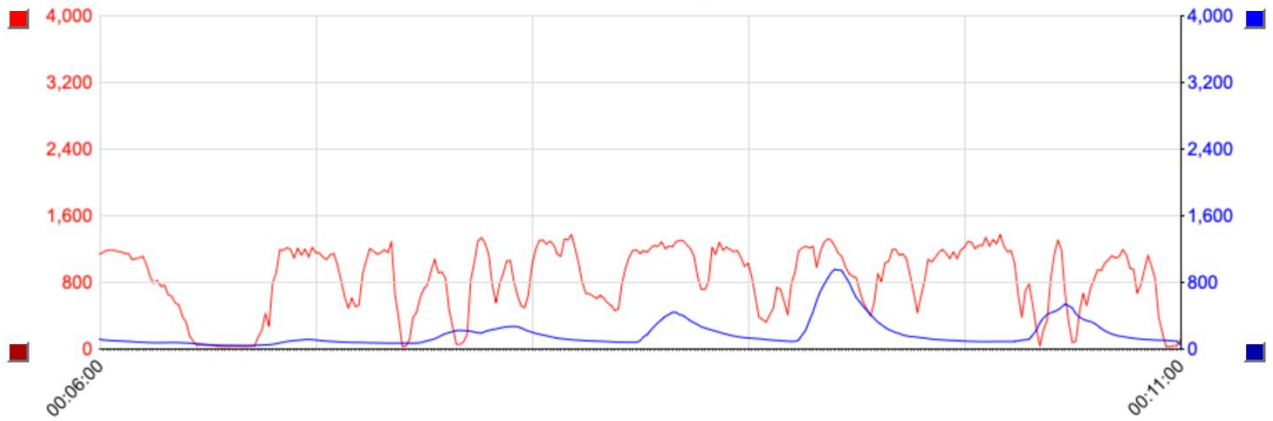
Visits and records



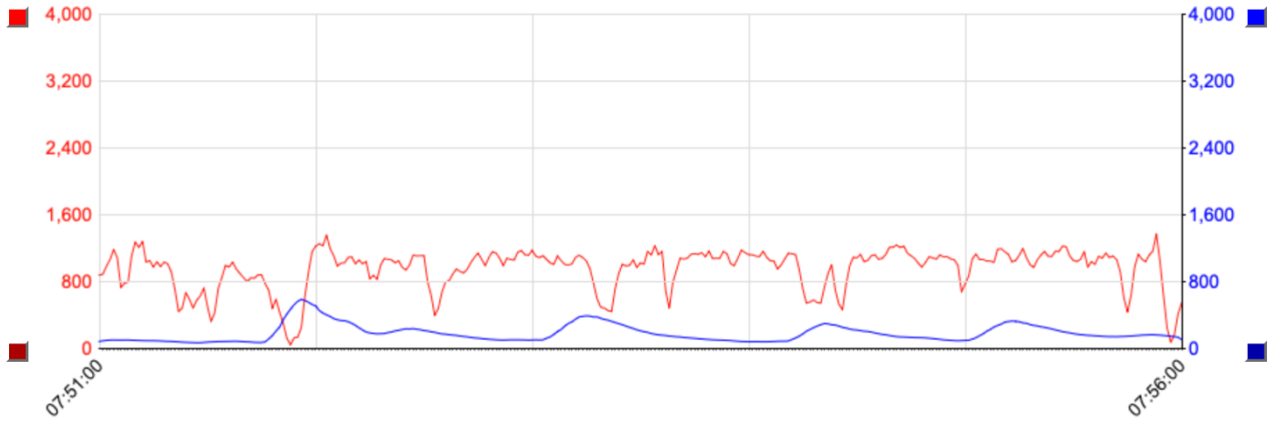
A **visit** refers to when an animal enters its head into the unit, receives food drops, and exits the unit.

GreenFeed data

Visits and records



A **visit** refers to when an animal enters its head into the unit, receives food drops, and exits the unit.



A **visit** that translated in a **record** refers to when the visit is translated on a record in the data due to some criteria (e.g., airflow, duration, etc.).

GreenFeed data

How it looks the preliminary data?

- A **record** refers to when the visit is translated on a record in the data due to some criteria (e.g., >2 minutes).

FID Animal ID Date/Time Gas records

Day 1

FeederID	AnimalName	RFID	StartTime	EndTime	GoodDataDuration	CO2GramsPerDay	CH4GramsPerDay	O2GramsPerDay	H2GramsPerDay	H2SGram	AirflowLiters	AirflowCf	WindSpeedM	WindDirDeg	WindCf	WasInterrup	InterruptingT	TempPipeDe
716	"00000000084000325068185	00000000084000325068185	1/17/25 15:57	1/17/25 16:02	0:04:03	11937.27945	413.1706076	8323.428782	1.458118937	0	32.036336	1.0098183	0	178.79661	1		""	7.7187098
716	"00000000084000323451388	00000000084000323451388	1/17/25 16:06	1/17/25 16:10	0:03:37	13949.42834	513.6369597	9635.877246	0.822802364	0	32.015106	1.0099245	0	178.81678	1		""	8.3500651
716	"00000000084000325068188	00000000084000325068188	1/17/25 16:24	1/17/25 16:27	0:02:08	15411.63206	321.3278208	10442.09501	0.53868698	0	32.036834	1.0098158	0	178.78703	1		""	7.7054621
716	"00000000084000325068180	00000000084000325068180	1/17/25 17:47	1/17/25 17:51	0:02:58	13688.66245	488.2964363	9175.229933	0.54728294	0	32.056347	1.0097183	0	178.79032	1		""	4.6788099
716	"00000000084000325068185	00000000084000325068185	1/17/25 18:36	1/17/25 18:39	0:03:21	15957.21263	646.2453375	10223.10354	3.785191748	0	32.099123	1.0095044	0	178.76738	1		""	4.2169024
716	"00000000084000323451396	00000000084000323451396	1/17/25 18:40	1/17/25 18:45	0:02:26	15863.67165	506.590961	9935.3091	1.002334145	0	31.988212	1.0100589	0	178.79591	1		""	4.8572011
716	"00000000084000323451397	00000000084000323451397	1/17/25 19:14	1/17/25 19:16	0:02:13	11793.95293	377.9065725	9229.520851	0.187687309	0	31.994322	1.0100284	0	178.68575	1		""	4.8502965
716	"00000000084000323451388	00000000084000323451388	1/17/25 19:16	1/17/25 19:18	0:02:11	16313.43316	623.1853147	12126.85668	2.017734312	0	31.981435	1.0100928	0	178.77159	1		""	5.0517467
716	"00000000084000323451396	00000000084000323451396	1/17/25 19:56	1/17/25 20:00	0:02:01	15196.57865	338.296016	10102.43615	0.607711324	0	31.954838	1.0102258	0	178.74038	1		""	4.3014249

GreenFeed data

How it looks the preliminary data?

- A **record** refers to when the visit is translated on a record in the data due to some criteria (e.g., >2 minutes).

FID Animal ID Date/Time Gas records

	FeederID	AnimalName	RFID	StartTime	EndTime	GoodDataDuration	CO2GramsPerDay	CH4GramsPerDay	O2GramsPerDay	H2GramsPerDay	H2SGram	AirflowLiters	AirflowCf	WindSpeedM	WindDirDeg	WindCf	WasInterrup	InterruptingT	TempPipeDe
Day 1	716	"000000000840003250681853"	000000000840003250681853	1/17/25 15:57	1/17/25 16:02	0:04:03	11937.27945	413.1706076	8323.428782	1.458118937	0	32.036336	1.0098183	0	178.79661	1		""	7.7187098
	716	"000000000840003234513880"	000000000840003234513880	1/17/25 16:06	1/17/25 16:10	0:03:37	13949.42834	513.6369597	9635.877246	0.822802364	0	32.015106	1.0099245	0	178.81678	1		""	8.3500651
	716	"000000000840003250681886"	000000000840003250681886	1/17/25 16:24	1/17/25 16:27	0:02:08	15411.63206	321.3278208	10442.09501	0.53868698	0	32.036834	1.0098158	0	178.78703	1		""	7.7054621
	716	"000000000840003250681802"	000000000840003250681802	1/17/25 17:47	1/17/25 17:51	0:02:58	13688.66245	488.2964363	9175.229933	0.54728294	0	32.056347	1.0097183	0	178.79032	1		""	4.6788099
	716	"000000000840003250681853"	000000000840003250681853	1/17/25 18:36	1/17/25 18:39	0:03:21	15957.21263	646.2453375	10223.10354	3.785191748	0	32.099123	1.0095044	0	178.76738	1		""	4.2169024
	716	"000000000840003234513962"	000000000840003234513962	1/17/25 18:40	1/17/25 18:45	0:02:26	15863.67165	506.590961	9935.3091	1.002334145	0	31.988212	1.0100589	0	178.79591	1		""	4.8572011
	716	"000000000840003234513970"	000000000840003234513970	1/17/25 19:14	1/17/25 19:16	0:02:13	11793.95293	377.9065725	9229.520851	0.187687309	0	31.994322	1.0100284	0	178.68575	1		""	4.8502965
	716	"000000000840003234513881"	000000000840003234513881	1/17/25 19:16	1/17/25 19:18	0:02:11	16313.43316	623.1853147	12126.85668	2.017734312	0	31.981435	1.0100928	0	178.77159	1		""	5.0517467
	716	"000000000840003234513965"	000000000840003234513965	1/17/25 19:56	1/17/25 20:00	0:02:01	15196.57865	338.296016	10102.43615	0.607711324	0	31.954838	1.0102258	0	178.74038	1		""	4.3014249
Day 2	716	"000000000840003250681886"	000000000840003250681886	1/19/25 0:09	1/19/25 0:15	0:05:00	13183.24443	391.7282611	10060.82002	0.353498333	0	33.155902	1.0042205	0	178.70879	1		""	-19.31494
	716	"000000000840003287325016"	000000000840003287325016	1/19/25 0:17	1/19/25 0:25	0:06:21	13381.90598	433.665752	8806.080682	1.981819752	0	33.076789	1.0046161	0	178.62073	1		""	-18.19648
	716	"000000000840003234513965"	000000000840003234513965	1/19/25 0:34	1/19/25 0:38	0:03:35	14381.98045	428.6615936	10732.47823	2.169819032	0	33.145633	1.0042718	0	178.59255	1		""	-19.08508
	716	"000000000840003250681901"	000000000840003250681901	1/19/25 0:39	1/19/25 0:48	0:06:43	14543.24361	478.4556924	12427.65396	0.852624657	0	32.929528	1.0053524	0	178.65563	1		""	-17.92528
	716	"000000000840003234513970"	000000000840003234513970	1/19/25 0:50	1/19/25 0:55	0:04:53	13497.88193	424.2608148	10721.96175	0.343329325	0	33.031469	1.0048427	0	178.64464	1		""	-17.73593
	716	"000000000840003250681910"	000000000840003250681910	1/19/25 1:25	1/19/25 1:32	0:05:58	11362.72786	277.3173832	8253.945337	0.669969657	0	33.08887	1.0045556	0	178.61297	1		""	-20.25465
	716	"000000000840003250681882"	000000000840003250681882	1/19/25 1:42	1/19/25 1:46	0:03:27	14536.62706	597.7691179	11076.88992	1.383252169	0	32.97472	1.0051264	0	178.69908	1		""	-19.36938
	716	"000000000840003250681888"	000000000840003250681888	1/20/25 9:21	1/20/25 9:25	0:03:27	11709.76294	413.6870601	9778.643322	1.359603209	0	32.222525	1.0088874	0	178.54247	1		""	-25.64439

GreenFeed data

How it looks the preliminary data?

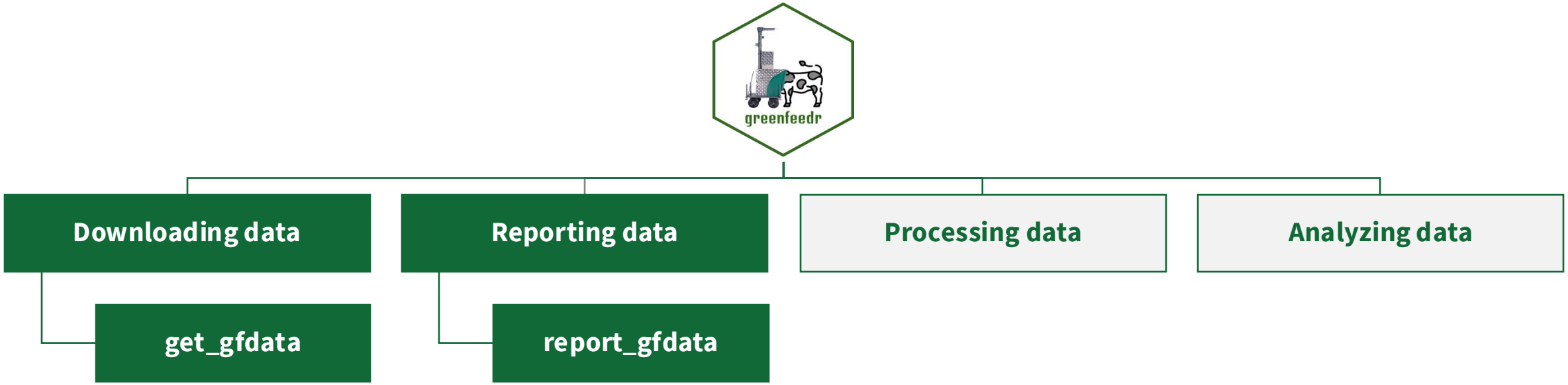
- A **record** refers to when the visit is translated on a record in the data due to some criteria (e.g., >2 minutes).

FID Animal ID Date/Time Gas records

	FeederID	AnimalName	RFID	StartTime	EndTime	GoodDataDuration	CO2GramsPerDay	CH4GramsPerDay	O2GramsPerDay	H2GramsPerDay	H2SGram	AirflowLiters	AirflowCf	WindSpeedN	WindDirDeg	WindCf	WasInterrup	InterruptingT	TempPipeDe
Day 1	716	"000000000840003250681853"	000000000840003250681853	1/17/25 15:57	1/17/25 16:02	0:04:03	11937.27945	413.1706076	8323.428782	1.458118937	0	32.036336	1.0098183	0	178.79661	1		""	7.7187098
	716	"000000000840003234513880"	000000000840003234513880	1/17/25 16:06	1/17/25 16:10	0:03:37	13949.42834	513.6369597	9635.877246	0.822802364	0	32.015106	1.0099245	0	178.81678	1		""	8.3500651
	716	"000000000840003250681886"	000000000840003250681886	1/17/25 16:24	1/17/25 16:27	0:02:08	15411.63206	321.3278208	10442.09501	0.53868698	0	32.036834	1.0098158	0	178.78703	1		""	7.7054621
	716	"000000000840003250681802"	000000000840003250681802	1/17/25 17:47	1/17/25 17:51	0:02:58	13688.66245	488.2964363	9175.229933	0.54728294	0	32.056347	1.0097183	0	178.79032	1		""	4.6788099
	716	"000000000840003250681853"	000000000840003250681853	1/17/25 18:36	1/17/25 18:39	0:03:21	15957.21263	646.2453375	10223.10354	3.785191748	0	32.099123	1.0095044	0	178.76738	1		""	4.2169024
	716	"000000000840003234513962"	000000000840003234513962	1/17/25 18:40	1/17/25 18:45	0:02:26	15863.67165	506.590961	9935.3091	1.002334145	0	31.988212	1.0100589	0	178.79591	1		""	4.8572011
	716	"000000000840003234513970"	000000000840003234513970	1/17/25 19:14	1/17/25 19:16	0:02:13	11793.95293	377.9065725	9229.520851	0.187687309	0	31.994322	1.0100284	0	178.68575	1		""	4.8502965
	716	"000000000840003234513881"	000000000840003234513881	1/17/25 19:16	1/17/25 19:18	0:02:11	16313.43316	623.1853147	12126.85668	2.017734312	0	31.981435	1.0100928	0	178.77159	1		""	5.0517467
	716	"000000000840003234513965"	000000000840003234513965	1/17/25 19:56	1/17/25 20:00	0:02:01	15196.57865	338.296016	10102.43615	0.607711324	0	31.954838	1.0102258	0	178.74038	1		""	4.3014249
Day 2	716	"000000000840003250681886"	000000000840003250681886	1/19/25 0:09	1/19/25 0:15	0:05:00	13183.24443	391.7282611	10060.82002	0.353498333	0	33.155902	1.0042205	0	178.70879	1		""	-19.31494
	716	"000000000840003287325016"	000000000840003287325016	1/19/25 0:17	1/19/25 0:25	0:06:21	13381.90598	433.665752	8806.080682	1.981819752	0	33.076789	1.0046161	0	178.62073	1		""	-18.19648
	716	"000000000840003234513965"	000000000840003234513965	1/19/25 0:34	1/19/25 0:38	0:03:35	14381.98045	428.6615936	10732.47823	2.169819032	0	33.145633	1.0042718	0	178.59255	1		""	-19.08508
	716	"000000000840003250681901"	000000000840003250681901	1/19/25 0:39	1/19/25 0:48	0:06:43	14543.24361	478.4556924	12427.65396	0.852624657	0	32.929528	1.0053524	0	178.65563	1		""	-17.92528
	716	"000000000840003234513970"	000000000840003234513970	1/19/25 0:50	1/19/25 0:55	0:04:53	13497.88193	424.2608148	10721.96175	0.343329325	0	33.031469	1.0048427	0	178.64464	1		""	-17.73593
	716	"000000000840003250681910"	000000000840003250681910	1/19/25 1:25	1/19/25 1:32	0:05:58	11362.72786	277.3173832	8253.945337	0.669969657	0	33.08887	1.0045556	0	178.61297	1		""	-20.25465
	716	"000000000840003250681882"	000000000840003250681882	1/19/25 1:42	1/19/25 1:46	0:03:27	14536.62706	597.7691179	11076.88992	1.383252169	0	32.97472	1.0051264	0	178.69908	1		""	-19.36938
	716	"000000000840003250681888"	000000000840003250681888	1/20/25 9:21	1/20/25 9:25	0:03:27	11709.76294	413.6870601	9778.643322	1.359603209	0	32.222525	1.0088874	0	178.54247	1		""	-25.64439
Day 3	716	"000000000840003250681908"	000000000840003250681908	1/20/25 10:57	1/20/25 11:00	0:02:34	15228.46137	591.45482	10505.10477	1.662371119	0	32.463959	1.0076802	0	178.55164	1		""	-25.39634
	716	"000000000840003287325016"	000000000840003287325016	1/20/25 11:01	1/20/25 11:04	0:03:23	11893.01474	201.029619	10086.42097	0.895961995	0	32.324396	1.008378	0	178.57052	1		""	-24.37435

GreenFeed data management

What can greenfeedr be used for?



Question 2

How many GreenFeed units are you using simultaneously?

Reporting data

Fast checking, tracking, and reduces manual processing

Reporting data

Fast checking, tracking, and reduces manual processing

report_gfdata(input_type = **PRELIM | DAILY**, exp = **NAME**, unit = **FID**, start_date = **MM/DD/YY**, save_dir = **/DIR/,** plot_opt = **ALL|CH4|CO2|O2|H2>**, rfid_file = **/FILE_PATH|DATAFRAME**, user = **USERNAME**, pass = **PASSWORD)**

report_gfdata(input_type = **FINAL**, exp = **NAME**, unit = **FID**, start_date = **MM/DD/YY**, end_date = **MM/DD/YY**, save_dir = **/DIR/,** plot_opt = **ALL|CH4|CO2|O2|H2**, rfid_file = **/FILE_PATH|DATAFRAME**, user = **USERNAME**, pass = **PASSWORD**, file_path = **FINAL_REPORT)**

Reporting data
Function: report_gfdata

@description Generates a PDF report of preliminary or finalized GreenFeed data

report_gfdata(

input_type = **“PRELIM”**,

exp = **“Exp01”**,

unit = **“305”**, # “304,305”, c(304,305)

start_date = **“02/20/2025”**, # “2025-02-20”, “2/20/25”

end_date = **“03/17/2025”**,

save_dir = **“/Downloads/”,**

plot_opt = **“ALL”**, # “CH4”, “CO2”, “O2”

rfid_file = **“/Exp01/rfid_file.xlsx”**, #Col1=ID, Col2=RFID

user = **“GUILLERMO”**,

pass = **“GF1992”**

)

@return A CSV file with preliminary GreenFeed data and a PDF report

Reporting data
Function: report_gfdata

@description Generates a PDF report of preliminary or finalized GreenFeed data

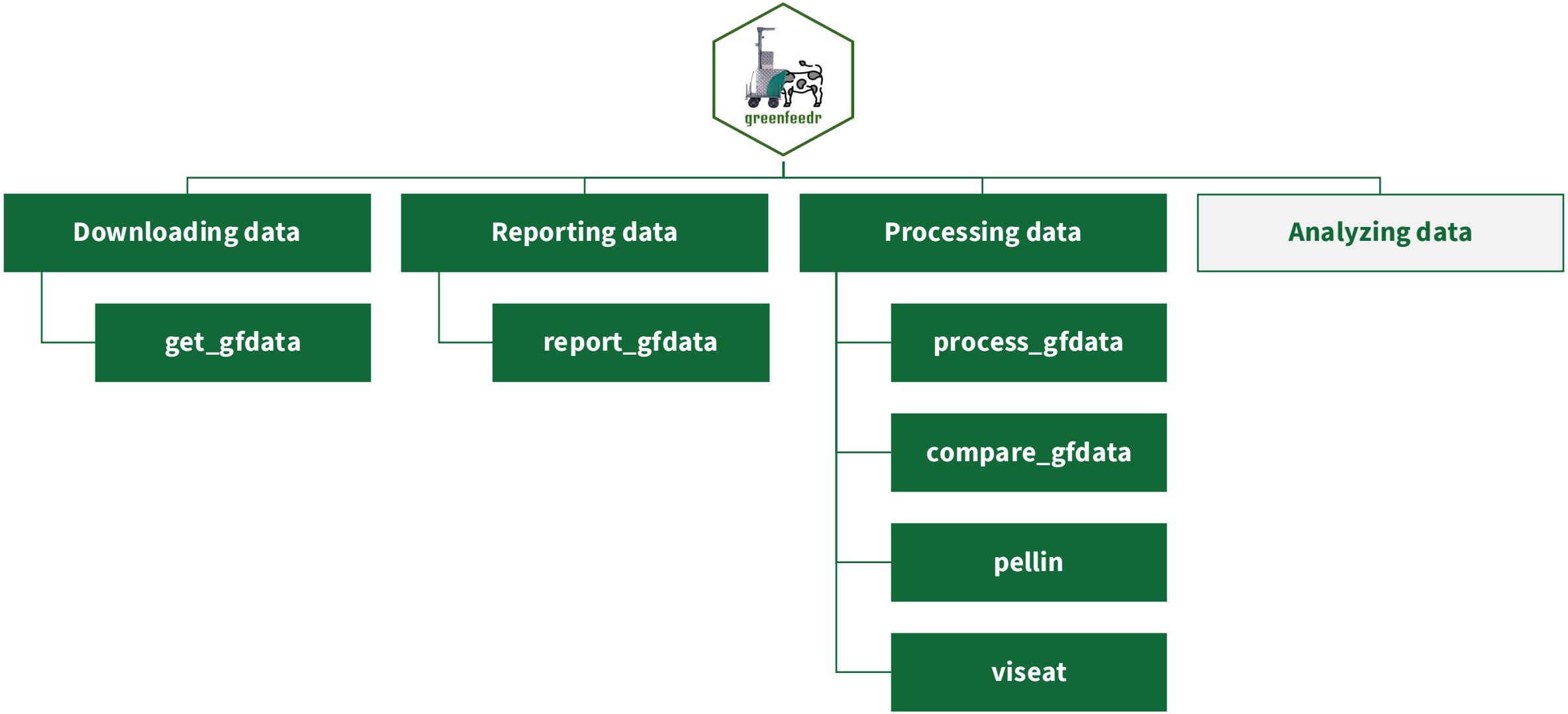
```
report_gfdata(
        input_type = "PRELIM",
        exp = "Exp01",
        unit = "305", # "304,305", c(304,305)
        start_date = "02/20/2025", # "2025-02-20", "2/20/25"
        end_date = "03/17/2025",
        save_dir = "/Downloads/",
        plot_opt = "ALL", # "CH4", "CO2", "O2"
        rfid_file = "/Exp01/rfid_file.xlsx", #Col1=ID, Col2=RFID
        user = "GUILLERMO",
        pass = "GF1992"
      )
```

```
report_gfdata(
        input_type = "FINAL", #Final
        exp = "Exp01",
        unit = "305", # "304,305", c(304,305)
        start_date = "02/20/2025", # "2025-02-20", "2/20/25"
        end_date = "03/17/2025",
        save_dir = "/Downloads/",
        plot_opt = "ALL", # "CH4", "CO2", "O2"
        rfid_file = "/Exp01/rfid_file.xlsx", #Col1=ID, Col2=RFID
        file_path = "/Exp01/SummarizedData_Date.xlsx"
      )
```

@return A CSV file with preliminary GreenFeed data and a PDF report

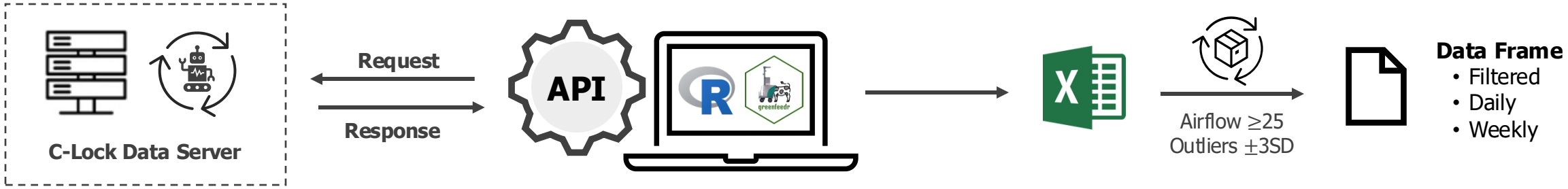
GreenFeed data management

What can greenfeedr be used for?



Processing data

Filtering and averages



Processing data

Filtering and averages

process_gfdata(data = **DATAFRAME**, start_date = **MM/DD/YY**, end_date = **MM/DD/YY**, param1 = **records**, param2 = **days**, min_time = **2**, cutoff = **3)**

Question 3

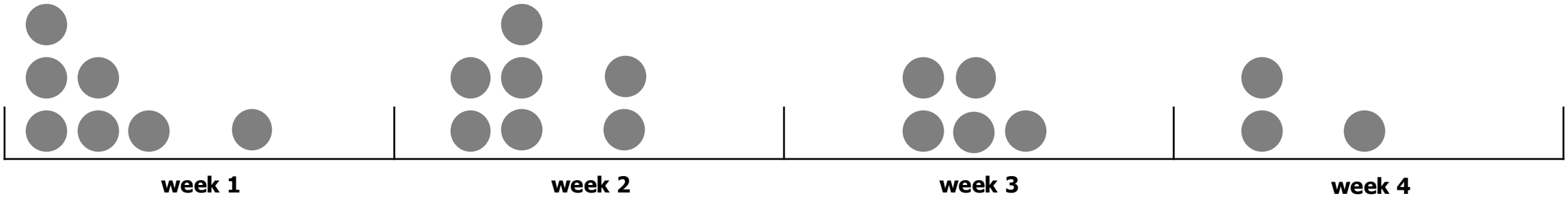
What is the minimum number of records per animal needed to ensure reliable gas emissions?

Question 4

How do you know that the gas records are well-distributed across day and weeks?

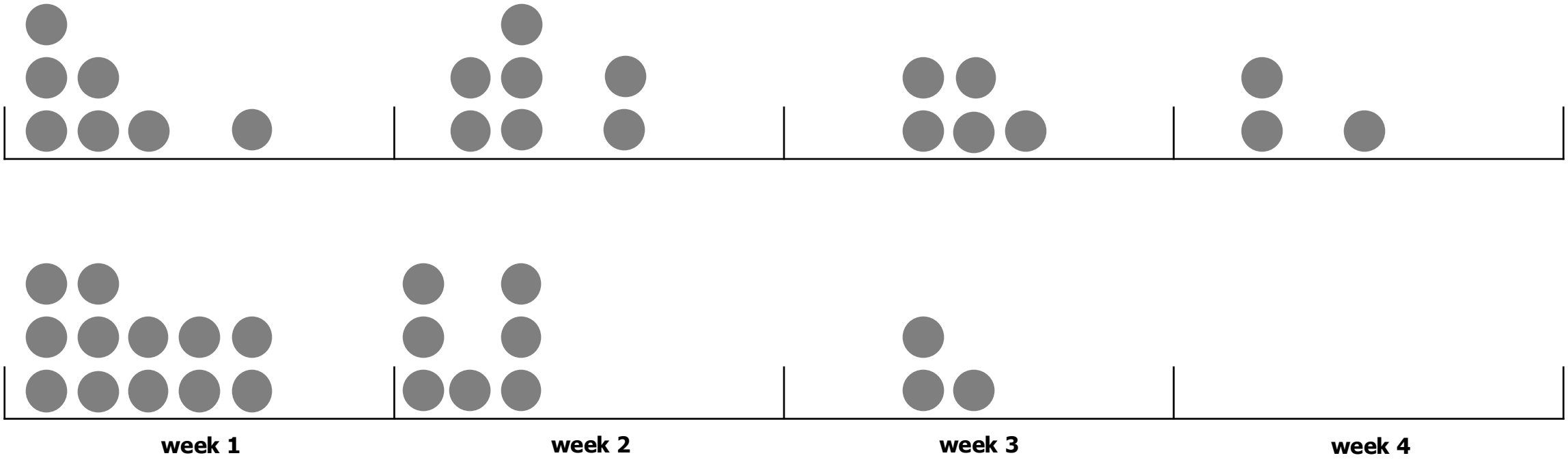
Processing data

Cow 1992: threshold > 20 records



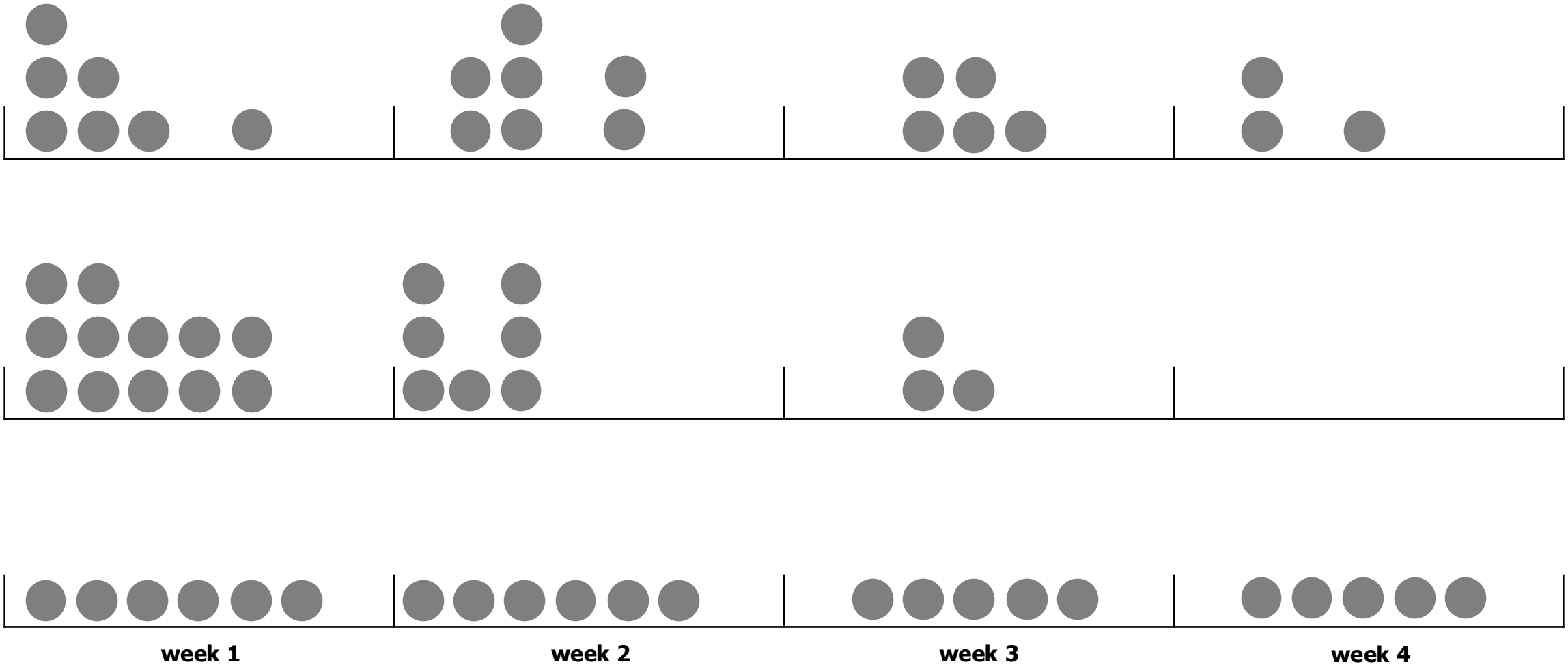
Processing data

Cow 1992: threshold > 20 records



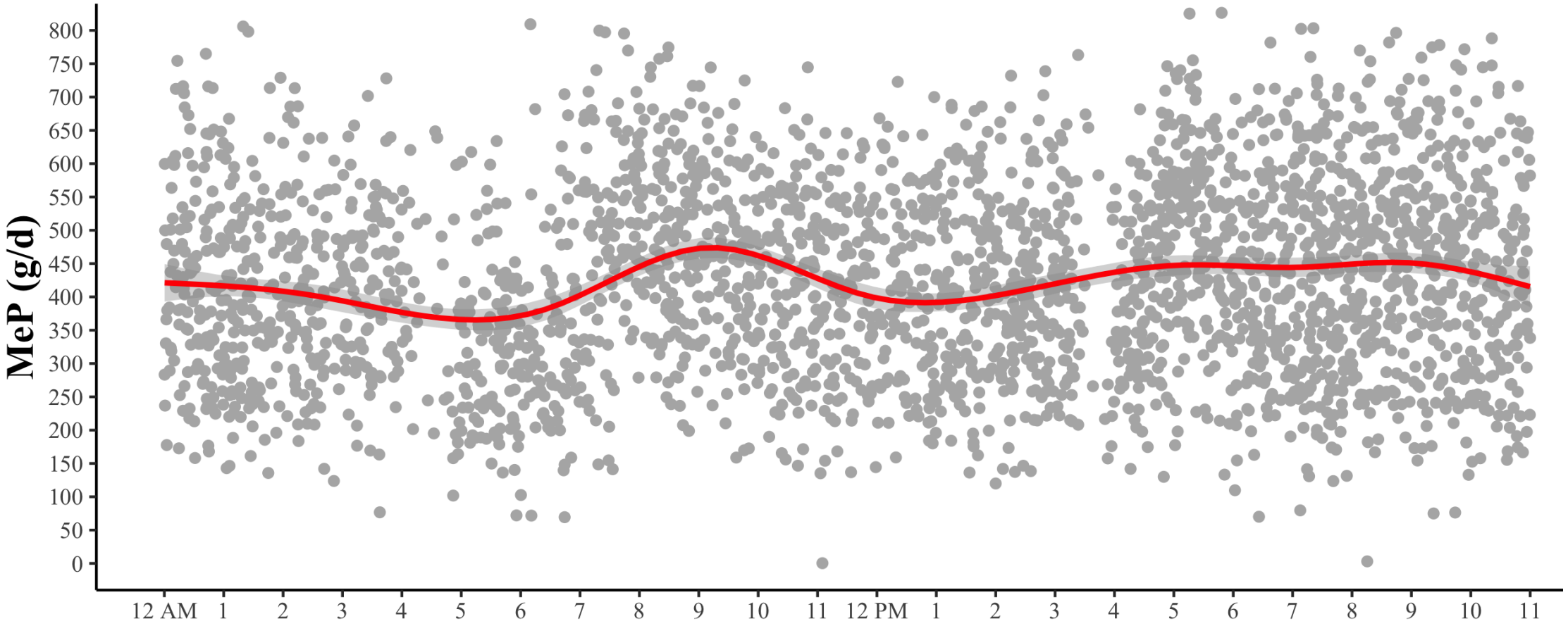
Processing data

Cow 1992: threshold > 20 records



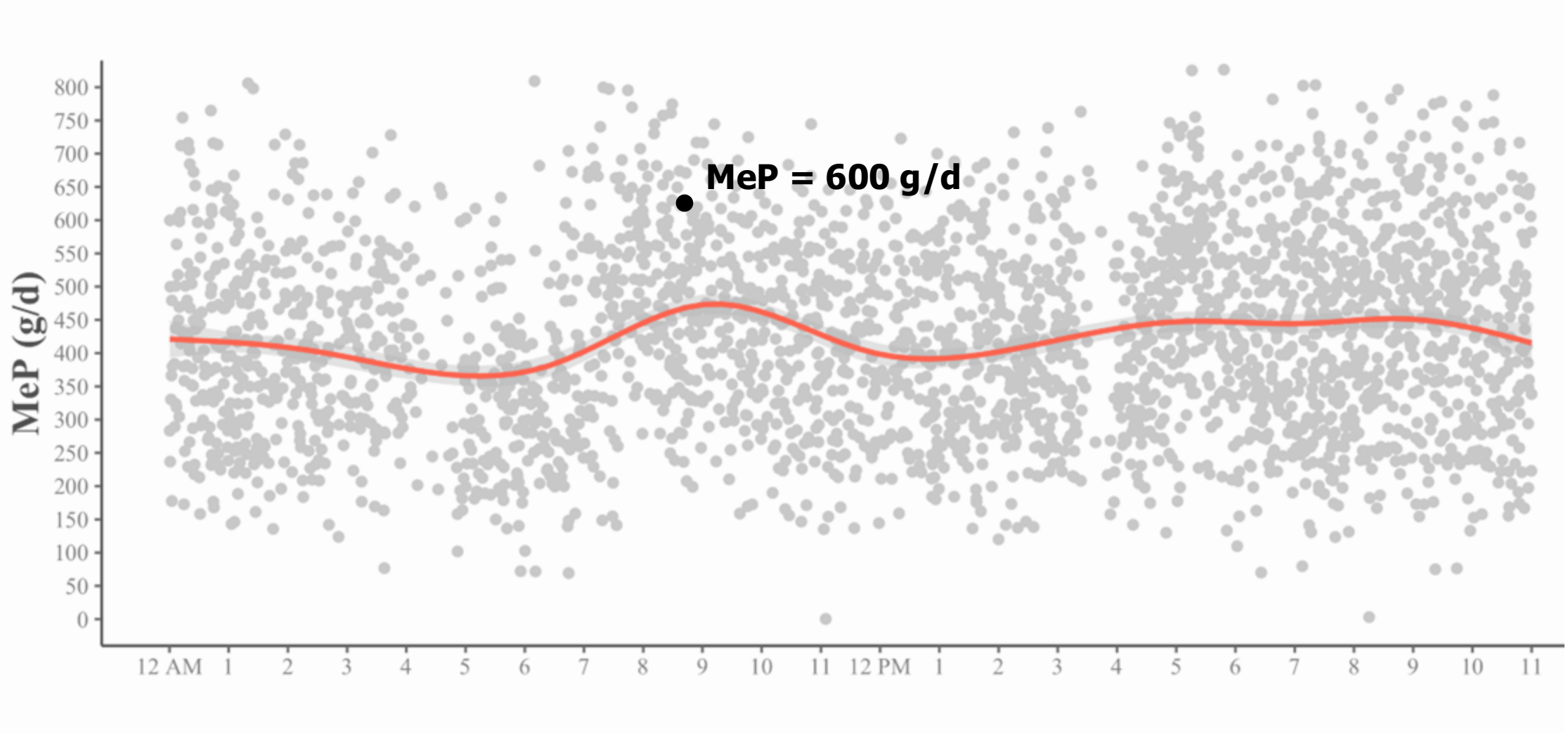
Processing data

GreenFeed system: numerous short-term records across the day for multiple days



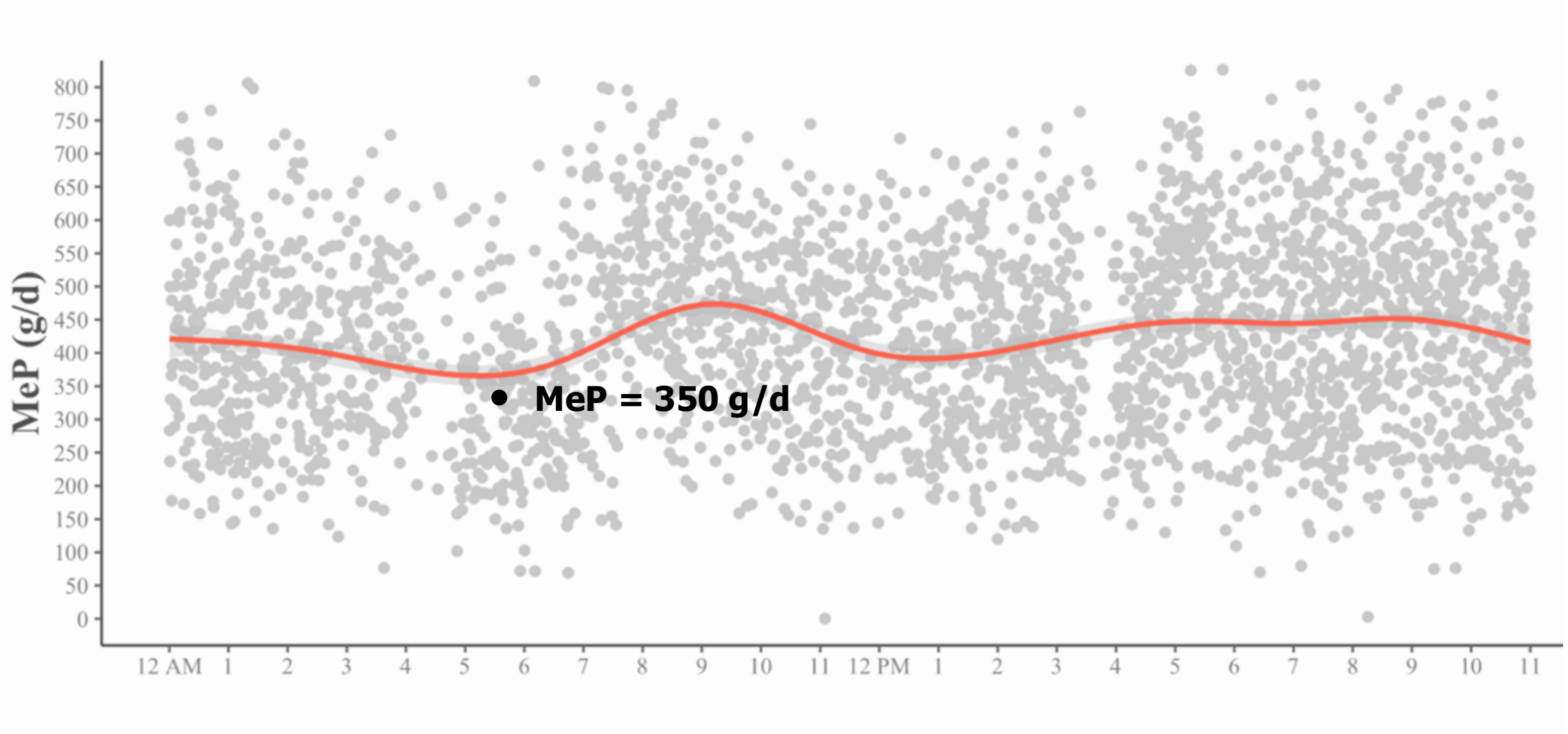
Processing data

GreenFeed: numerous short-term records across the day for multiple days



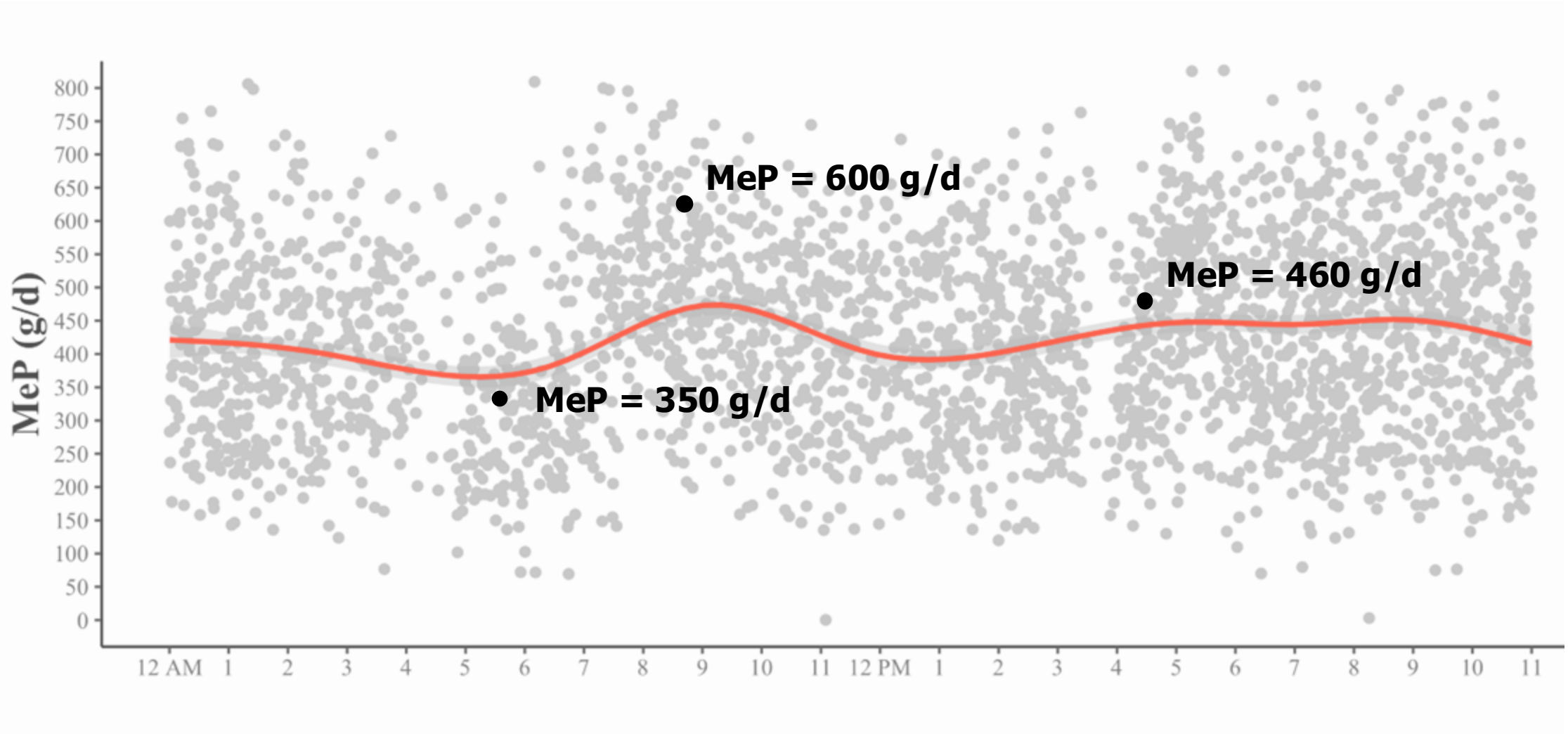
Processing data

GreenFeed: numerous short-term records across the day for multiple days



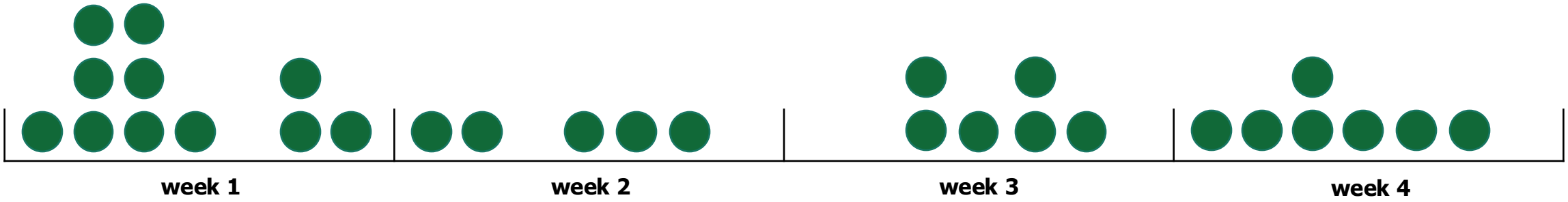
Processing data

GreenFeed: numerous short-term records across the day for multiple days



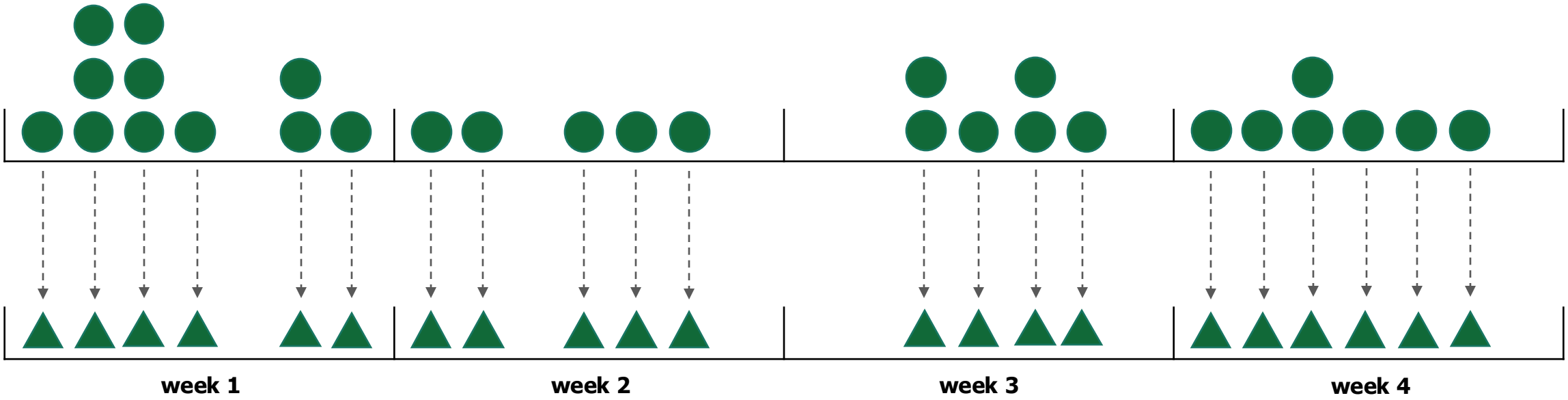
Processing data

Cow 1218: at least 1 record per day and 1 day with records



Processing data

Cow 1218: at least 1 record per day and 1 day with records



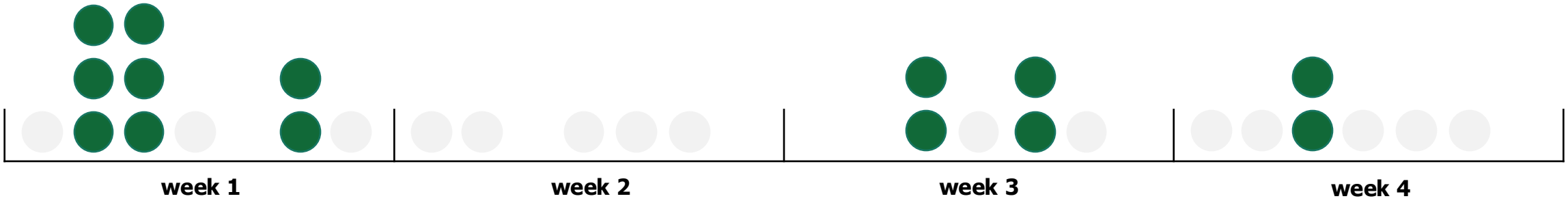
Cow 1218

29 records, 21 days, 4 weeks

Average CH4 = 447 ± 79 g/d

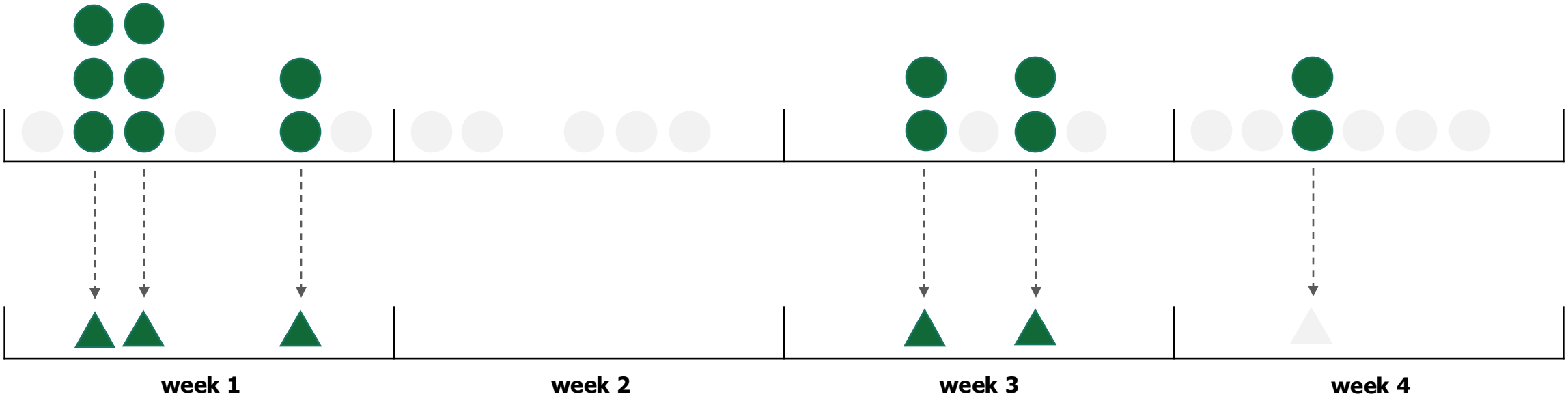
Processing data

Cow 1218: at least 2 records per day and 2 days with records



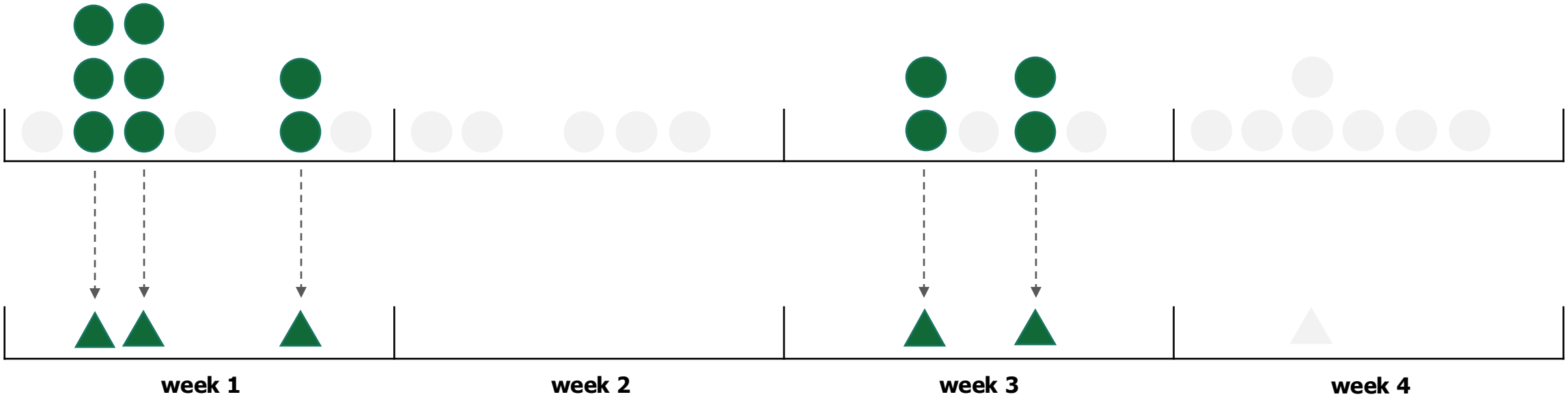
Processing data

Cow 1218: at least 2 records per day and 2 days with records



Processing data

Cow 1218: at least 2 records per day and 2 days with records



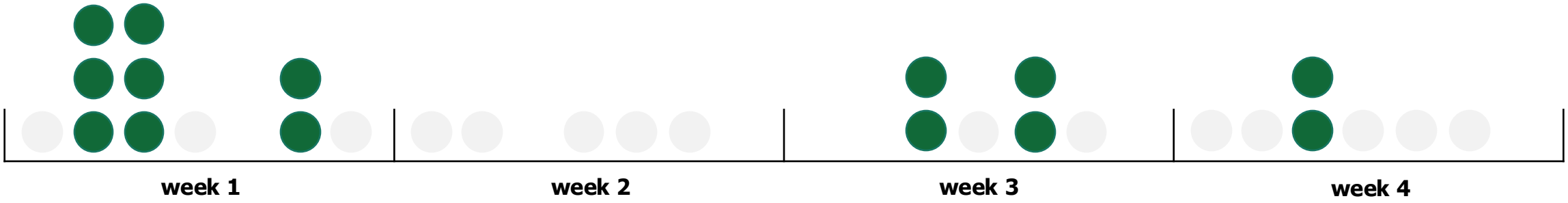
Cow 1218

12 records, 5 days, 2 weeks

Average CH4 = 382 ± 41 g/d

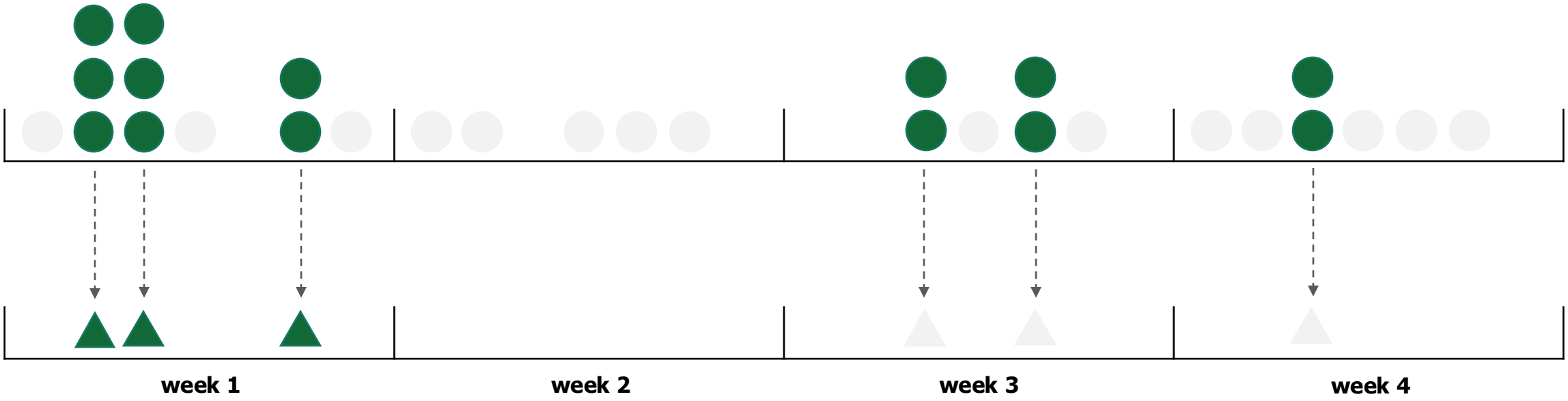
Processing data

Cow 1218: at least 2 records per day and 3 days with records



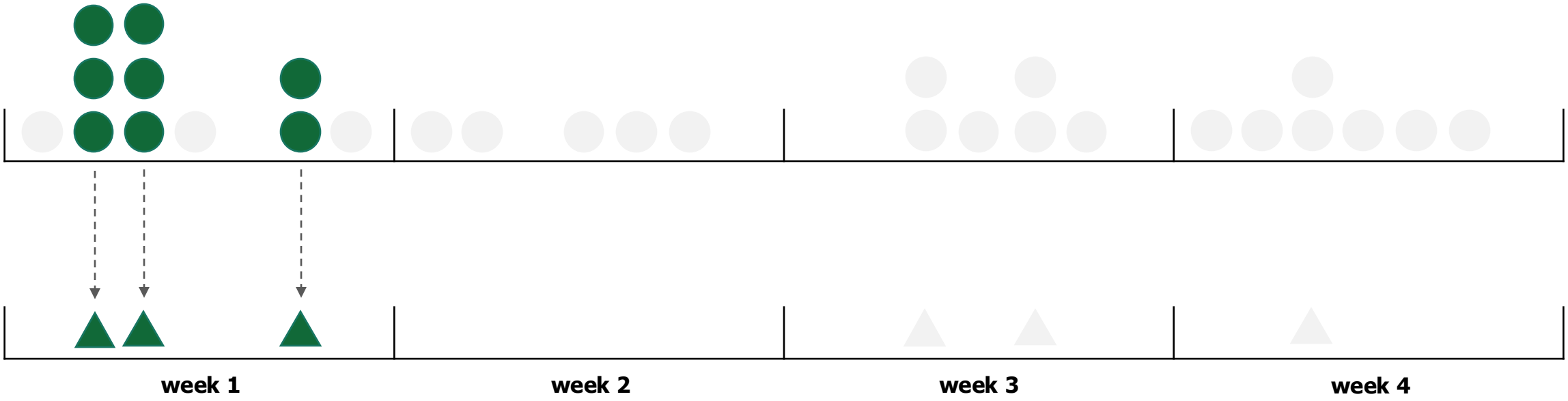
Processing data

Cow 1218: at least 2 records per day and 3 days with records



Processing data

Cow 1218: at least 2 records per day and 3 days with records



Cow 1218

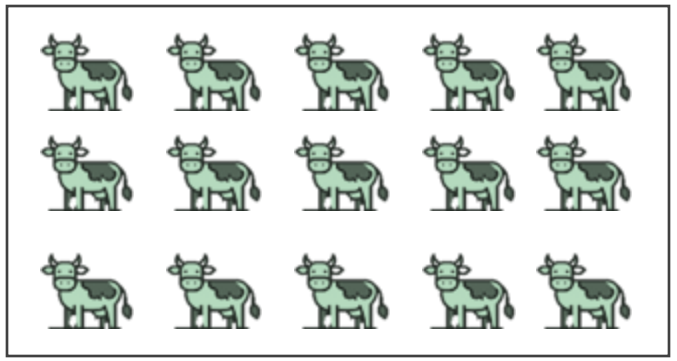
8 records, 3 days, 1 week

Average CH4 = 411 ± NA g/d

Processing data

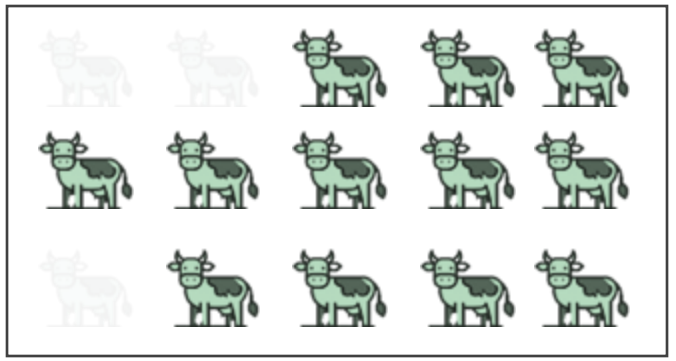
How to find the “best” combination of parameters?

Parameters 2:1:2



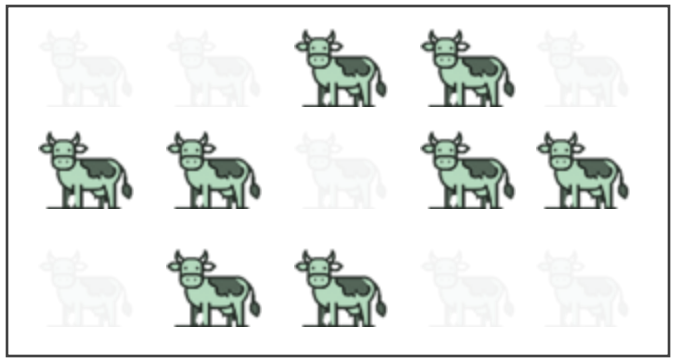
236 records, 32 cows
Average CH4 = 435 ± 87 g/d
CV CH4 = 20%

Parameters 3:3:2



82 records, 25 cows
Average CH4 = 426 ± 84 g/d
CV CH4 = 19.6%

Parameters 3:5:2



31 records, 16 cows
Average CH4 = 410 ± 74 g/d
CV CH4 = 18.5%

Processing data

Function: eval_gfparam and process_gfdata

@description Evaluate parameters that best fit for 'GreenFeed' data

eval_gfparam(

data = DATAFRAME,

start_date = MM/DD/YY,

end_date = MM/DD/YY

)

@return A data frame with number of records, cows, and mean, SD, and CV for weekly data

Processing data

Function: eval_gfparam and process_gfdata

@description Evaluate parameters that best fit for 'GreenFeed' data

eval_gfparam(data = **DATAFRAME**, start_date = **MM/DD/YY**, end_date = **MM/DD/YY**)**

@return A data frame with number of records, cows, and mean, SD, and CV for weekly data

@description Process and calculates daily and weekly averages of GreenFeed data

process_gfdata(

data = "GF_Data", #SummarizedData_Date.xlsx

start_date = "02/20/2025",

end_date = "03/06/2025",

param1 = 2,

param2 = 2,

min_time = 2

)

@return A list of 3 data frames: filtered data, daily data, and weekly data

Processing data

Calculating pellet intakes and checking visits

Processing data

Calculating pellet intakes and checking visits

pellin(user = **USERNAME**, pass = **PASSWORD**, unit = **FID**, gcup = **34**, start_date = **MM/DD/YY**, end_date = **MM/DD/YY**, save_dir = **/DIR/**,
rfid_file = **/FILE_PATH**, file_path = **/FEEDTIMES)**

viseat(user = **USERNAME**, pass = **PASSWORD**, unit = **FID**, start_date = **MM/DD/YY**, end_date = **MM/DD/YY**,
rfid_file = **/FILE_PATH**, file_path = **/FEEDTIMES)**

Processing data
Function: pellin

@description Processes feedtimes file to calculate pellet intakes

```
pellin(
        user = "GUILLERMO",
        pass = "GF1992",
        unit = "305", # c(304, 305)
        gcup = 34, # c(34, 45)
        start_date = "02/20/2025",
        end_date = "03/07/2025",
        save_dir = "/Downloads/",
        rfid_file = "/Exp01/rfid_file.xlsx", #Col1=ID, Col2=RFID
       )
```

@return An Excel file with pellet intakes for all animals and days within the specified period

Processing data
Function: viseat

@description Processes feedtimes file to check visits and drops

pellin(

user = "GUILLERMO",

pass = "GF1992",

unit = "305", # c(304, 305)

start_date = "02/20/2025",

end_date = "03/07/2025",

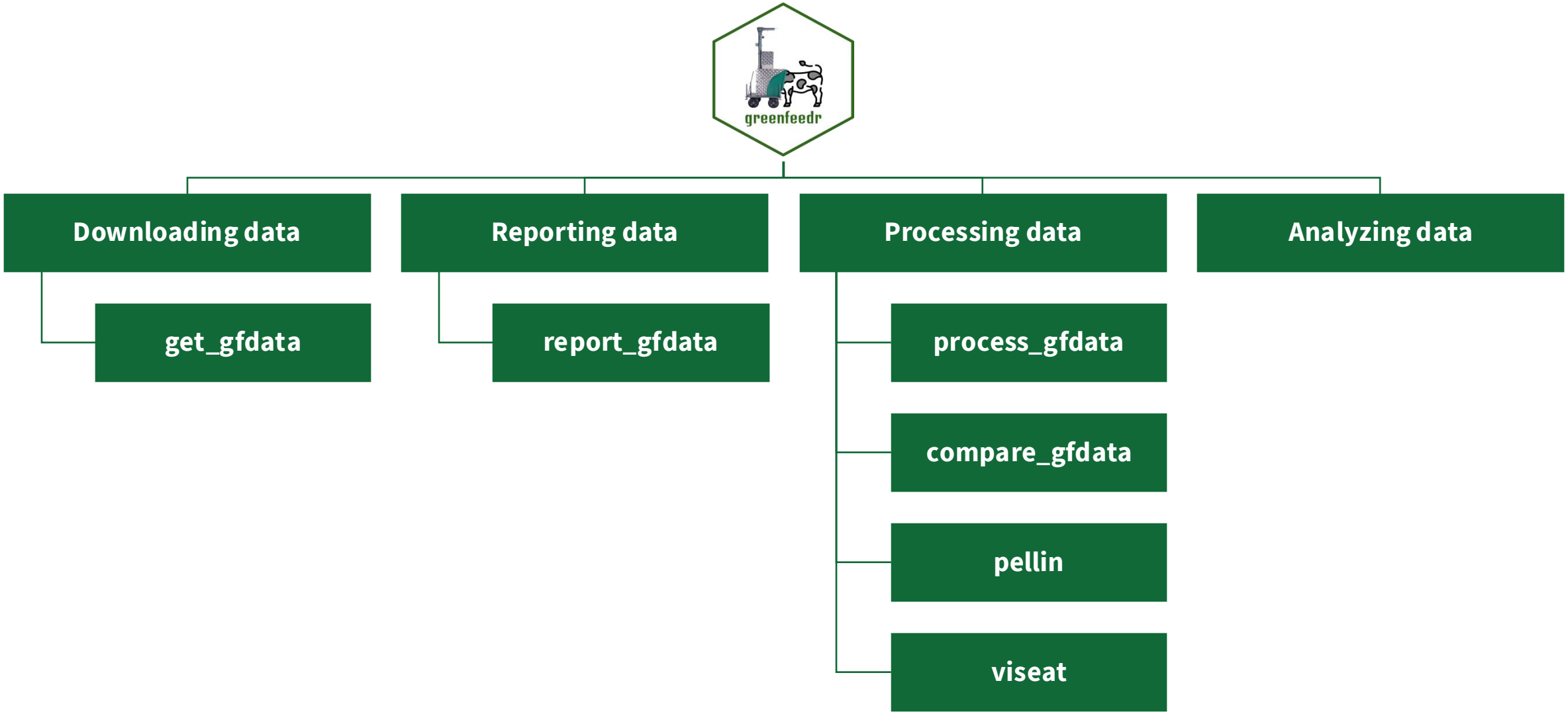
rfid_file = "/Exp01/rfid_file.xlsx", #Col1=ID, Col2=RFID

)

@return A list of two data frames: visits per unit and visits per animal

GreenFeed data management

What can greenfeedr be used for?



Thanks,
and see you on Friday!

Guillermo Martinez Boggio

guillermo.martinezboggio@wisc.edu

github.com/GMBog

Thank you for using the greenfeedr package!

Martinez-Boggio et al. (2025). Greenfeedr: an R-package for processing and reporting GreenFeed data