1 Introduction

Manometric methods are commonly used for measuring biogas and methane production in order to determine methane potential and anaerobic biodegradability of a given substrate. In manometric methods, bottle headspace pressure is measured under constant temperature conditions from different techniques such as using pressure manometers or transducers, manometer assisted syringes, or low flow pressure [1]. The measured pressure is converted to biogas and methane (if composition is provided) volume data using manometric calculation methods. Manometric methods require accumulation of biogas within the bottle headspace. Headspace pressure is usually measured with a portable manometer, which evaluates the pressure relative to atmospheric pressure.

Determination of CH$_4$ volume from manometric measurements requires data processing. The `calcBgMan()` function was developed to process manometric data. A newly developed biogas package (available for the R environment since 2015) address issues with time-consuming calculations and lack of reproducibility among laboratories for obtaining BMP [2]. The biogas package consists of ten functions including `cumBg()`, which are used for processing volumetric, manometric, gravimetric, and gas density measurements. The resulting biogas and methane production and production rates can be further used to calculate BMP using the `summBg()` function. This document describes how to process manometric biogas measurements using the `calcBgMan()` function. We assumed that readers are familiar with biogas data collection and R.

2 Overview of the Function

`calcBgMan()` is a "high-level" function within the biogas package. The purpose of `calcBgMan()` is to convert pressure data collected in the laboratory to cumulative biogas and CH$_4$ production and production rates. Typically, these values will then be used for calculation of BMP, e.g., using the `summBg()` function. `calcBgMan()` can handle data from multiple bottles and accepts three different data structures. For simple operations (e.g., interpolation and standardization
of biogas volume) `calcBgMan()` is supported by calls to external "low-level" functions (refer to Section 5, Table 1).

Two manometric methods are commonly used to calculate methane production from pressure and composition measurements. Method 1 is based on normalized CH\(_4\) concentrations, whereas method 2 accounts for the actual CH\(_4\) in the bottle headspace. Both methods are available through `calcBgMan()` and results are expected to be virtually identical. The examples below describe cumulative biogas calculation on two different datasets. The ‘‘long’’ structured dataset is evaluated using manometric method 1, whereas the ‘‘longcombo’’ dataset is evaluated using manometric method 2, as true methane concentrations are provided.

### 2.1 Function Arguments

The arguments for the `calcBgMan()` function are:

```r
## function (dat, comp = NULL, temp = NULL,
## interval = TRUE, data.struct = "longcombo",
## id.name = "id", time.name = "time",
## pres.name = "pres", comp.name = "xCH4",
## temp.init = NULL, pres.init = NULL, pres.resid = NULL,
## rh.resid = NULL, rh.resid.init = 1,
## headspace = NULL, vol.hs.name = "vol.hs",
## absolute = TRUE, pres.amb = NULL,
## cmethod = "removed", imethod = "linear",
## extrap = FALSE, addt0 = TRUE, showt0 = TRUE,
## dry = FALSE, empty.name = NULL,
## std.message = !quiet, check = TRUE,
## temp.std =getOption("temp.std", as.numeric(NA)),
## pres.std =getOption("pres.std", as.numeric(NA)),
## unit.temp =getOption("unit.temp", "C"),
## unit.pres =getOption("unit.pres", "atm"),
## quiet = FALSE)
## NULL
```

Most of the arguments have default values, but to calculate CH\(_4\) production we must provide values for at least `dat` (data frame with pressure measurements), `comp` (data frame with gas composition), and `temp` (biogas temperature)\(^1\) along with the names of a few columns in the input data frames. If the `comp` argument (or, for the ‘‘longcombo’’ format, the `comp.name` argument) is not provided, `calcBgMan()` will return results for biogas only and not CH\(_4\).

\(^1\) By default, temperature is in °C and pressure in atm, but these can be changed in the function call with the `temp.unit` and `pres.unit` arguments, or globally with `options`. The same default values apply for temperature and pressure for presentation of biogas and methane, but these can be changed in the function call with the `temp.std` and `pres.std` arguments.
By default `interval = TRUE` and `data.struct` is set as `"longcombo"`. `"wide"` and `"long"` structured data will be restructured to `"longcombo"` internally by `cumBgdDataPrep()`, when specified by the `data.struct` argument (refer to Section 2.2). When data are cumulative, the interval argument should be set to FALSE.

Similarly, there is an `id.name` argument for the bottle identification code (ID) column (default is `"id"`). For `data.struct = "wide"`, there is no ID column. Instead data for each bottle, have individual columns and column names, which are used as ID codes. Here, the name of the column containing the response variables (`pres.name`), is set as the name of the first column with response variables. All following columns are assumed to have measurement data.

Furthermore, we need to specify the name of the time column containing time data using the `time.name` argument (default is `"time"`). If separate data frames are used for `dat` and `comp`, the name must match. Time data may be POSIXct objects, but then t0 will not be added to rows by the `calcBgdMan()` function. In addition, the `addt0` argument is used to add row with “time zero” (`time.name = 0`) for each bottle in order to calculate production rates for the first observation (default is TRUE). `showt0` determines if the “time zero” should be returned in the output (default is TRUE if `time.name` is numeric and contains 0 and otherwise FALSE). The use of t0 output is for plotting.

The `comp.name` argument is used to indicate which column within the `comp` data frame contains CH\textsubscript{4} concentrations. Default is `"xCH4"`. `comp` may also just be a single value instead of a data frame or column. When providing a single value for `comp`, this value is applied to all observations and the `comp.name` argument is not needed. The definition of xCH\textsubscript{4} depends on the `cmethod` argument. By default (`cmethod = "removed"`) the function calculates volumes following [3] as the product of standardized volume of biogas removed and normalized CH\textsubscript{4} content (as mole fraction in dry biogas, normalized so the sum of mole fractions of CH\textsubscript{4} and CO\textsubscript{2} sum to unity). If results should be based on the sum of methane removed and methane remaining in the bottle headspace, `cmethod` should be set to `"total"`. When `cmethod = "total"`, CH\textsubscript{4} concentration is expressed relative to all biogas components except water (CH\textsubscript{4}, CO\textsubscript{2}, N\textsubscript{2}, H\textsubscript{2}S, etc.) instead of CH\textsubscript{4} and CO\textsubscript{2} only.

If any CH\textsubscript{4} measurements are missing, `"xCH4"` is interpolated by the external “low-level” function `interp()`. Here, the `imethod` argument can be used to define interpolation method (default is `"linear"`), which is passed to `interp()` (default is FALSE). The `extrap` argument is used to indicate if composition data (`comp.name`) should be extrapolation (e.g. in the case of missing initial composition values). Extrapolation is constant (the value of the nearest observation is taken). In general, extrapolation and extensive interpolation should be avoided.

Initial headspace temperature and pressure are required to determine initial gas volume in the bottles and can be set using the `temp.init` and `pres.init`, respectively. Default values are NULL. Similarly, post venting headspace pressure is required for manometric calculation methods and can be set using the
pres.resid argument.

These are used to correct for apparent biogas production due simply to a change in headspace temperature or pressure between bottle sealing and the first measurements (frequently bottles are filled, flushed, and sealed at room temperature). Alternatively, this effect can be corrected by measuring pressure after the bottle temperature is increased to incubator temperature. In this case, the composition should be set to zero for this first measurement interval.

Initial and post venting headspace pressure can be absolute or gauge, depending on the absolute argument. If only a single pressure value is provided, this will be used for all observations. By default, pressure is absolute (absolute = TRUE). When absolute = FALSE, gauge pressure measurements are converted to absolute using the pres.amb argument, representing the absolute ambient pressure. pres.amb can only be set as a single value and is 101.325 kPa (1.0 atm) by default.

Additionally, a data frame containing headspace volumes is required if cmethod = "total" and should contain at least a headspace volume column (vol.hs.name) and a bottle identification column, with the same column name as in dat and comp data frames. The headspace volume column can be set using the vol.hs.name argument (default is "vol.hs"). Initial relative humidity of gas in the headspace is set to 1 by default using the rh.resid.init argument. If values are provided for relative humidity of gas in the headspace after experiment was initiated, these can be defined by using the rh.resid argument.

2.2 Data Structures

Input data may be structured in one of three ways: "long", "wide", or "longcombo". Default is "longcombo", where the composition column is in the dat data frame and no separate composition data frame is required. In the following examples all three data structures will be addressed. For more details about each of the three data structure refer to biogas_quick_start vignette.
3 Examples: Calculation of Cumulative Production of Biogas and CH\textsubscript{4} and Production Rates from Manometric Measurements

Calculation of cumulative biogas and CH\textsubscript{4} production and production rates, typically requires two data frames: Biogas volume measurements and biogas composition (CH\textsubscript{4} fraction).

3.1 "longcombo" Data Structure

In this example, we will use "longcombo" example datasets included in the biogas package: sludgeTwoBiogas for both biogas volumes and composition and sludgeTwoSetup for grouping and headspace volumes. In this experiment both manometric and gravimetric methods were used, which is where the "Two" comes from. Substrate and inoculum masses provided from sludgeTwoSetup are not interesting before calculation of BMP using summBg() (refer to Section 4).

These data are from an experiment carried out with seven different ratios of sludge + inoculum to headspace. The substrate was primary wastewater sludge. The experiment included 24 batch bottles, all with inoculum and one without substrate (blank):

- Three bottles with inoculum only (Blank50)
- Three bottles with wastewater sludge 25 (WWS25)
- Three bottles with wastewater sludge 25b (WWS25b)
- Three bottles with wastewater sludge 40 (WWS40)
- Three bottles with wastewater sludge 50 (WWS50)
- Three bottles with wastewater sludge 50b (WWS50b)
- Three bottles with wastewater sludge 60 (WWS60)
- Three bottles with wastewater sludge 75 (WWS75)

More details can be found in the helpfiles for these data.

```r
data("sludgeTwoBiogas")
dim(sludgeTwoBiogas)
## [1] 324   8
head(sludgeTwoBiogas)
```
## id time.d pres mass.init mass.final xCH4 xCO2 xCH4n
## 1 1 0.0000000 0 187.4168 187.4168 0.0000 0.0000 0.0000000
## 2 1 0.6270833 368 187.4149 187.3814 0.1010 0.1004 0.5014896
## 3 1 1.6027778 286 187.3823 187.3568 0.2253 0.1610 0.5832255
## 4 1 1.9972222 134 187.3538 187.3447 0.2485 0.1800 0.5799300
## 5 1 2.7152778 142 187.3406 187.3287 0.3246 0.2062 0.6115298
## 6 1 3.6180556 168 187.3295 187.3150 0.3674 0.2222 0.6231343

`summary(sludgeTwoBiogas)`

## id time.d pres
## Min. : 1.0 Min. : 0.000 Min. : 0.0
## 1st Qu.: 8.0 1st Qu.: 2.715 1st Qu.: 181.0
## Median :12.5 Median : 8.637 Median : 358.0
## Mean :12.0 Mean :18.113 Mean : 565.4
## 3rd Qu.:17.0 3rd Qu.:24.029 3rd Qu.: 753.0
## Max. :21.0 Max. :32.014 Max. :2919.0

## mass.init mass.final xCH4
## Min. :144.7 Min. :144.7 Min. :0.0000
## 1st Qu.:163.6 1st Qu.:163.6 1st Qu.:0.5479
## Median :183.5 Median :183.5 Median :0.6201
## Mean :184.3 Mean :184.3 Mean :0.6460
## 3rd Qu.:200.0 3rd Qu.:199.9 3rd Qu.:0.6418
## Max. :228.4 Max. :228.4 Max. :0.7377

## xCO2 xCH4n
## Min. :0.0000 Min. :0.0000
## 1st Qu.:0.2599 1st Qu.:0.4466
## Median :0.3012 Median :0.6602
## Mean :0.2791 Mean :0.6220
## 3rd Qu.:0.3354 3rd Qu.:0.6727
## Max. :0.6344 Max. :0.7637

data("sludgeTwoSetup")

dim(sludgeTwoSetup)
## [1] 18 5

head(sludgeTwoSetup)
## id descrip vol.hs m.inoc m.sub.vs
## 1 1 Blank50 79.980 80.02 0.0000000
## 2 2 Blank50 79.980 80.02 0.0000000
## 3 3 Blank50 79.950 80.05 0.0000000
## 4 7 WWS75 117.575 35.02 0.3588463
## 5 8 WWS75 117.575 35.02 0.3588463

6
The first step in processing data from a BMP trial is calculation of cumulative biogas and CH\(_4\) production and production rates. Subsequently, BMP can be calculated by the high-level function `summBg()` included in the biogas package. Cumulative biogas and CH\(_4\) production and production rates from pressure data with `sludgeTwoBiogas` and `sludgeTwoSetup` data frames as input can be calculated with `calcBgMan()`.

To calculate CH\(_4\) production from these ‘longcombo’ data, we must provide values for at least `dat` and `comp`, which is in a combined data frame (we will use `sludgeTwoBiogas`), and `temp` (biogas temperature) along with the names of a few columns in our input data frame.

We can use default values ‘longcombo’, ‘id’, and ‘pres’ for the `data.struct`, `id.name`, and `pres.name` arguments, respectively. Whereas, the `time.name` and `comp.name` arguments need to be specified as ‘time.d’ and ‘xCH4n’, respectively. Similar, default values can be used for `cmethod = ‘removed’`, evaluating CH\(_4\) concentration based on normalized CH\(_4\) and CO\(_2\) values, for `imethod = ‘linear’`, resulting in internal linear interpolation of ‘xCH4n’ by calling the `interp()` function, and for `extrap = FALSE`.

Initial headspace temperature and pressure and post venting headspace pressure are set as constants using the `temp.init`, `pres.init`, and `pres.resid` arguments, respectively. Initial and post venting headspace pressure can be absolute or gauge depending on the value of the `absolute` argument. We will set `absolute = FALSE` to account for pressure measurements provided in gauge. In order to calculate absolute pressure from gauge pressure measurements when `absolute = FALSE`, a single absolute ambient pressure value is required. Here we will set `pres.amp` to 1013 mbar. Note the unit of the pressure data in `sludgeTwoBiogas`. Default unit is atm, but can be changed using the `unit.pres` argument. In this example we set the `unit.pres = ‘mbar’` to match unit of
pres.name and pres.resid column. Absolute ambient pressure is provided in the same unit as defined in the unit.pres argument.

Finally, headspace volumes are provided from sludgeTwoSetup. The data frame containing headspace volumes is defined using the headspace argument, whereas the default value "vol.hs" can be used for the column containing headspace volume data.

```r
cum.prod.lc <- calcBgMan(sludgeTwoBiogas, temp = 30,
  time.name = "time.d", comp.name = "xCH4n",
  temp.init = 30, pres.init = 0.0,
  pres.resid = 0,
  headspace = sludgeTwoSetup,
  pres.amb = 1013, absolute = FALSE,
  unit.pres = "mbar")
```

## Pressure measurements are GAUGE. If this is incorrect, change 'absolute' argument to TRUE.
## Using a standard pressure of 1013.25 mbar and standard temperature of 0 C for standardizing volume.

Note the message about standard temperature and pressure. It is important to make sure these values are correct, therefore users are reminded by a message. Also, note warning about pressure unit. This warning is to make sure all pressure measurements are gauge, as absolute is set to FALSE, meaning all pressure data are corrected to absolute pressure by the calcBgMan() function.

The output becomes:

```r
head(cum.prod.lc)
```

## id time.d pres mass.init mass.final xCH4 xCO2 xCH4n
## 1 1 0.0000000 0 187.4168 187.4168 0.0000 0.0000 0.0000000
## 2 1 0.6270833 368 187.4149 187.3814 0.1010 0.1004 0.5014896
## 3 1 1.6027778 286 187.3823 187.3568 0.2253 0.1610 0.5832255
## 4 1 1.9972222 134 187.3538 187.3447 0.2485 0.1800 0.5799300
## 5 1 2.7152778 142 187.3406 187.3287 0.3246 0.2062 0.6115298
## 6 1 3.6180556 168 187.3295 187.3150 0.3674 0.2222 0.6231343

## vol.hs temperature pres.resid pres.abs pres.resid.abs
## 1 79.98 30 0 1013 1013
## 2 79.98 30 0 1381 1013
## 3 79.98 30 0 1299 1013
## 4 79.98 30 0 1147 1013
## 5 79.98 30 0 1155 1013
## 6 79.98 30 0 1181 1013

## rh.resid pres.resid.prev rh.resid.prev temp.prev xCH4n.prev

---

2 Remember that standard conditions can be set in the function call with `temp.std` and `pres.std`, or globally with `options()`.
The data frame that is returned has maintained the `"longcombo"` structure with all the original columns in `sludgeTwoBiogas`, plus additional columns from manometric biogas calculations.

\(v\) stands for (standardized) volume, \(cv\) (standardized) cumulative volume, \(rv\) stands for (standardized) volume production rate, and \(Bg\) and \(CH_4\) for biogas and methane. \(cvBg\) contains standardized cumulative biogas production and \(cvCH_4\) contains standardized cumulative \(CH_4\) production.

Graphical illustrations often increases interpretability. Here we will use the \texttt{ggplot} function from the \texttt{ggplot2} package to plot it.

\begin{verbatim}
library(ggplot2)

ggplot(cum.prod.lc, aes(time.d, cvCH4, colour = factor(id))) +
  geom_point() +
  geom_line(aes(group = id)) +
  labs(x = "Time [d]", y = "cvCH4 [mL]", colour = "Bottle id") +
  theme_bw()
\end{verbatim}
3.2 ‘‘long’’ Data Structure

In this example, we will use ‘‘long’’ example data set included in the biogas package: strawPressure for headspace pressure in batch bottles, strawComp for methane content of biogas, and strawSetup for grouping and headspace volumes.

These data are interval-based BMP measurement of headspace pressure in 12 bottles with straw as the substrate. The experiment included 12 batch bottles:

- Two bottles with inoculum and straw treated with treatment r3
- Two bottles with inoculum and straw treated with treatment r5
- Two bottles with inoculum and straw treated with treatment r6.5
- Two bottles with inoculum and straw treated with treatment r8
- Two bottles with inoculum and straw treated with treatment r8 no buff
- Two bottles with inoculum and straw treated with treatment r9

Bottles were ca. 600 mL glass serum bottles with butyl rubber septa and screw caps. Pressure was measured using an electronic manometer. Data in strawMass, strawSetup, and strawComp are from the same bottles. More details can be found in the helpfiles for these data.

data("strawPressure")

dim(strawPressure)
## [1] 72  5

head(strawPressure)

```
##   bottle  date.time  time   pres  pres.resid
## 1     1 2016-04-28 10:30  1.9 153.2034  101.629
## 2     2 2016-04-28 10:30  1.9 152.0888  101.629
## 3     3 2016-04-28 10:30  1.9 142.0576  101.629
## 4     4 2016-04-28 10:30  1.9 142.1590  101.629
## 5     5 2016-04-28 10:30  1.9 138.7139  101.629
## 6     6 2016-04-28 10:30  1.9 139.5245  101.629
```

summary(strawPressure)

```
##   bottle  date.time  time
##  Min. : 1.00 2016-04-28 10:30:00.00:12 Min. : 1.900
##  1st Qu.: 3.75 2016-04-29 16:00:00.00:12 1st Qu.: 3.120
##  Median : 6.50 2016-04-30 13:20:00.00:12 Median : 4.965
##  Mean : 6.50 2016-05-02 11:00:00.00:12 Mean : 5.987
`
data("strawComp")

dim(strawComp)
## [1] 63 4

head(strawComp)
## bottle date.time time xCH4
## 1 1 2016-04-28 10:30:00.00 1.9 0.4189
## 2 2 2016-04-28 10:30:00.00 1.9 0.4219
## 3 3 2016-04-28 10:30:00.00 1.9 0.3395
## 4 4 2016-04-28 10:30:00.00 1.9 0.3015
## 5 5 2016-04-28 10:30:00.00 1.9 0.2769
## 6 6 2016-04-28 10:30:00.00 1.9 0.2850

summary(strawComp)
## bottle date.time time xCH4
## Min. : 1.000 2016-04-28 10:30:00.00:12 Min. : 1.900
## 1st Qu.: 3.000 2016-05-04 14:30:00.00:11 1st Qu.: 3.120
## Median : 6.000 2016-04-30 13:20:00.00:10 Median : 4.010
## Mean : 6.175 2016-05-09 10:45:00.00: 9 Mean : 5.724
## 3rd Qu.: 9.000 2016-04-29 15:35:00.00: 1 3rd Qu.: 8.060
## Max. :12.000 2016-04-29 15:40:00.00: 1 Max. :12.910
## (Other) :19
## xCH4
## Min. :0.01187
## 1st Qu.:0.30020
## Median :0.43490
## Mean :0.39450
## 3rd Qu.:0.48925
## Max. :0.88720
data("strawSetup")

dim(strawSetup)
## [1] 12 6

head(strawSetup)
## bottle treatment start sub.mass inoc.mass
## 1 1 r3 2016-04-26 13:00:00.00 5.27 45.36
## 2 2 r3 2016-04-26 13:00:00.00 5.27 43.88
## 3 3 r5 2016-04-26 13:00:00.00 5.28 26.27
## 4 4 r5 2016-04-26 13:00:00.00 5.30 26.95
## 5 5 r6.5 2016-04-26 13:00:00.00 5.31 20.37
## 6 6 r6.5 2016-04-26 13:00:00.00 5.29 20.61

headspace
## 1 491.3
## 2 493.3
## 3 502.9
## 4 502.4
## 5 502.4
## 6 505.8

summary(strawSetup)
## bottle treatment start
## Min. : 1.00 r3 :2 2016-04-26 13:00:00.00:12
## 1st Qu.: 3.75 r5 :2
## Median : 6.50 r6.5 :2
## Mean : 6.50 r8 no buff:2
## 3rd Qu.: 9.25 r8 :2
## Max. :12.00 r9 :2
## sub.mass inoc.mass headspace
## Min. :5.240 Min. :14.57 Min. :491.3
## 1st Qu.:5.270 1st Qu.:16.48 1st Qu.:502.4
## Median :5.280 Median :18.76 Median :502.9
## Mean :5.284 Mean :23.31 Mean :503.1
## 3rd Qu.:5.300 3rd Qu.:26.44 3rd Qu.:507.5
## Max. :5.340 Max. :45.36 Max. :509.5

As with the ‘‘longcombo’’ data, cumulative production of CH\(_4\) is needed in order to calculate BMP. Again, we can calculate these with the \texttt{calcBgMan} function, using \texttt{strawPressure}, \texttt{strawComp}, and \texttt{strawSetup} data frames as input.

To calculate CH\(_4\) production from these ‘‘long’’ structured data, we must provide values for at least \texttt{dat}, \texttt{comp}, and \texttt{temp} along with the names of a few
columns in our input data frame. The \texttt{dat} and \texttt{comp} arguments are set as the data frames: \texttt{strawPressure} and \texttt{strawComp}, respectively, whereas \texttt{temp} is set as single values of 31 \degree C. For \texttt{data.struct} = \texttt{"longcombo"}, the data structure needs to be specified. Here we set \texttt{data.struct} = \texttt{"long"}. Furthermore, we need to specify the name of the id column in \texttt{strawPressure} as \texttt{bottle} using the \texttt{id.name} argument.

We can use default values \texttt{"time"}, \texttt{"pres"}, and \texttt{"xCH4"} for the \texttt{time.name}, \texttt{pres.name} and \texttt{comp.name} arguments, respectively. The id and time columns create a link between the two data frames. Observations from different times can be solved by the external \texttt{interp()} function by interpolation. Similar, default values can be used for \texttt{cmethod = "removed"}, evaluating CH$_4$ concentration based on normalized CH$_4$ and CO$_2$ values and for \texttt{imethod = "linear"}, resulting in internal linear interpolation of \texttt{"xCH4"} by calling the \texttt{interp()} function. Additionally, the response variables are interval data only and hence, we can use the default \texttt{interval = TRUE}.

In addition to interpolation for later observations, an extrapolation argument \texttt{(extrap)} can be provided if required. We do not have initial biogas composition (compare the heads of \texttt{strawPressure} and \texttt{strawComp}) so extrapolation is required to calculate initial volumes and rates. Therefore, we need to set \texttt{extrap = TRUE}.

Initial headspace temperature and pressure are set as constant values using the \texttt{temp.init} and \texttt{pres.init} arguments, respectively. Headspace pressure after venting is provided from the \texttt{strawPressure} data frame and can be defined by assigning the column name to the \texttt{pres.resid} argument. Similar to the \texttt{"longcombo"} data example, we will set \texttt{absolute = FALSE} to account for pressure measurements provided in gauge. In order to calculate absolute pressure from gauge pressure measurements when \texttt{absolute = FALSE}, a single absolute ambient pressure value is required. To correct gauge pressure data to absolute pressure, we will set \texttt{pres.amb} to 101.3 kPa. Again, note the unit of the pressure data in \texttt{strawPressure}. Pressure unit is set as atm by default, but this can be changed using the \texttt{unit.pres} argument. In this example we will set \texttt{unit.pres = "kPa"} to match unit of \texttt{pres.name} and \texttt{pres.resid} column.

Finally, headspace volumes are provided from \texttt{strawSetup}. The data frame containing these volumes is defined using the \texttt{headspace} argument, whereas the column is set as \texttt{vol.hs.name = "headspace"}.

The output becomes:

```r
cum.prod.1 <- calcBgMan(strawPressure, comp = strawComp, temp = 31,
data.struct = "long",
time.name = "time", id.name = "bottle", comp.name = "xCH4",
temp.init = 21.55, pres.resid = "pres.resid",
pres.init = 0.0,
headspace = strawSetup, vol.hs.name = "headspace",
pres.amb = 101.3, absolute = FALSE,
extrap = TRUE,
unit.pres = "kPa")
```
## Biogas composition is interpolated.
## Pressure measurements are GAUGE. If this is incorrect, change 'absolute'
## argument to TRUE.
## Using a standard pressure of 101.325 kPa and standard temperature
## of 0°C for standardizing volume.

head(cum.prod.l)

## bottle date.time time pres pres.resid
## 1 1 <NA> 0.00 NA NA
## 2 1 2016-04-28 10:30:00.00 1.90 153.2034 101.6290
## 3 1 2016-04-29 16:00:00.00 3.12 171.9485 101.8316
## 4 1 2016-04-30 13:20:00.00 4.01 140.7404 101.8316
## 5 1 2016-05-02 11:00:00.00 5.92 168.7061 102.1356
## 6 1 2016-05-04 14:30:00.00 8.06 168.4022 102.0343

head(xCH4)

## bottle date.time time pres pres.resid
## 1 NA NA NA NA
## 2 0.4189000 491.3 31 254.5034 202.9290
## 3 0.4363556 491.3 31 273.2485 203.1316
## 4 0.4413000 491.3 31 242.0404 203.1316
## 5 0.5014287 491.3 31 242.0404 203.4356
## 6 0.5152000 491.3 31 269.7022 203.3343

Note the message about standard temperature and pressure. It is important
to make sure these values are correct, therefore users are reminded by a mes-

Remember that standard conditions can be set in the function call with temp.std and
pres.std, or globally with options().
As with the ‘‘longcombo’’ data example, the \texttt{ggplot} function from the \texttt{ggplot2} package is used to increase interpretability of the results.

\begin{verbatim}
ggplot(cum.prod.l, aes(time, cvCH4, colour = factor(bottle))) + geom_point() + geom_line(aes(group = bottle)) + labs(x = "Time \[d\]", y = "cvCH4 \[mL\]", colour = "Bottle id") + theme_bw()
\end{verbatim}

Plotting results is recommended, and can be used to catch strange responses like for bottle # 2 here.

### 3.3 Additional Examples

For examples using ‘‘wide’’ structured input data frame and varying settings of other arguments (e.g. \texttt{addt0} and \texttt{showt0}), refer to vignette for volumetric \texttt{cumBg} function (\texttt{cumBgVol_function.Rnw}).

### 4 Continuing With the \texttt{calcBgMan()} Function

The \texttt{calcBgMan()} function is one of several \texttt{cumBg*()} functions within the biogas package. Results from the \texttt{cumBg*()} functions can be used directly in the \texttt{summBg()} function from the biogas package to calculate BMP for the data in question. Though, this operation often requires additional setup information (e.g. inoculum and substrate mass), which is most commonly provided in an
external data frame (**setup**). More details can be found in the associated help file.

### 5 Function Internals

In general, **cumBg** functions consist of four sections: argument checks, restructuring and sorting data, interpolation if needed, and biogas standardization and calculations. Restructuring and sorting of data and interpolation are handled by the external functions **cumBgDataPrep()** and **interp()**, respectively. From **interp()** gas composition, cumulative biogas production, and other variables can be interpolated to a specified time if required. From **cumBgDataPrep()** ‘‘wide’’ and ‘‘long’’ data structure are restructured to ‘‘longcombo’’ data, which is required for **calcBgMan()** for further data processing. Additionally, data are sorted, headspace is added if provided, and composition data is corrected if it seems to be a percentage. Subsequently, the restructured and sorted data is standardized in **cumBgVol()** by an external function called **stdVol()**.

**calcBgMan()** accepts all data structures, but the manometric calculation methods within **calcBgMan()** only process ‘‘longcombo’’ data. ‘‘wide’’ and ‘‘long’’ data are restructured internally by the “low-level” function **cumBgDataPrep()**. **cumBgDataPrep()** sorts and restructures biogas data for further calculation of cumulative biogas and methane production and production rates. The **datastruct** argument is internally passed to **cumBgDataPrep()** and restructured to ‘‘longcombo’’ structure prior to being processed by volumetric calculation methods.

Table 1: Operations done with the “low-level” functions in **calcBgMan()**. All functions are vectorized. See help files for more details.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standardize gas volume</td>
<td><strong>stdVol()</strong></td>
</tr>
<tr>
<td>Interpolate composition  etc.</td>
<td><strong>interp()</strong></td>
</tr>
<tr>
<td>Structurize and sort data</td>
<td><strong>cumBgDataPrep()</strong></td>
</tr>
</tbody>
</table>

### References

