

# Package ‘PoolDilutionR’

February 15, 2023

**Type** Package

**Title** Calculate Gross Biogeochemical Flux Rates from Isotope Pool Dilution Data

**Version** 1.0.0

**Description** Pool dilution is a isotope tracer technique wherein a biogeochemical pool is artificially enriched with its heavy isotopologue and the gross productive and consumptive fluxes of that pool are quantified by the change in pool size and isotopic composition over time. This package calculates gross production and consumption rates from closed-system isotopic pool dilution time series data. Pool size concentrations and heavy isotope (e.g.,  $^{15}\text{N}$ ) content are measured over time and the model optimizes production rate ( $P$ ) and the first order rate constant ( $k$ ) by minimizing error in the model-predicted total pool size, as well as the isotopic signature. The model optimizes rates by weighting information against the signal:noise ratio of concentration and heavy-isotope signatures using measurement precision as well as the magnitude of change over time. The calculations used here are based on von Fischer and Hedin (2002) <[doi:10.1029/2001GB001448](https://doi.org/10.1029/2001GB001448)> with some modifications.

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.2.3

**Suggests** testthat (>= 3.0.0), knitr, rmarkdown, ggplot2, tibble, tidyr

**Config/testthat/edition** 3

**Depends** R (>= 2.10)

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Kendalynn A. Morris [cre, aut]

(<<https://orcid.org/0000-0002-0388-6965>>),

Ben Bond-Lamberty [ctb] (<<https://orcid.org/0000-0001-9525-4633>>)

**Maintainer** Kendalynn A. Morris <kendalynn.morris@gmail.com>

**Repository** CRAN

**Date/Publication** 2023-02-15 12:20:06 UTC

## R topics documented:

frac_k_default . . . . .	2
frac_P_default . . . . .	3
Morris2023 . . . . .	3
pdr_cost . . . . .	4
pdr_estimate_k0 . . . . .	5
pdr_fractionation . . . . .	6
pdr_optimize . . . . .	6
pdr_optimize_df . . . . .	8
pdr_predict . . . . .	9

<b>Index</b>	<b>11</b>
--------------	-----------

---

frac_k_default	<i>Retrieve default k fractionation value for a pool</i>
----------------	--

---

### Description

Retrieve default k fractionation value for a pool

### Usage

```
frac_k_default(pool)
```

### Arguments

pool	Name of pool, character
------	-------------------------

### Value

The default entry for pool listed in [pdr\\_fractionation](#).

### Examples

```
frac_k_default("CH4")
```

---

frac_P_default	<i>Retrieve default P fractionation value for a pool</i>
----------------	--

---

**Description**

Retrieve default P fractionation value for a pool

**Usage**

```
frac_P_default(pool)
```

**Arguments**

pool	Name of pool, character
------	-------------------------

**Value**

The default entry for pool listed in [pdr\\_fractionation](#).

**Examples**

```
frac_P_default("CH4")
```

---

Morris2023	<i>Example time series data from a methane dilution pool experiment.</i>
------------	--

---

**Description**

Sequential measurements of methane concentration and isotopic signature were taken using a Picarro G2920 with a Small Sample Introduction module. This instrument provides gas concentrations in ppm and signatures in delta-13C, here we provide those data converted into volume of methane and atom percent.

**Usage**

```
Morris2023
```

**Format**

**id** Sample ID, a factor  
**time\_days** time in days between measurements, starting at 0  
**cal12CH4ml** ml of 12C-CH4 at each timestep  
**cal13CH4ml** ml of 13C-CH4 at each timestep  
**AP\_obs** atom percent 13C-CH4 at each timestep

pdr\_cost

*Cost function between observed and predicted pools***Description**

Cost function between observed and predicted pools

**Usage**

```
pdr_cost(
  params,
  time,
  m,
  n,
  m_prec,
  ap_prec,
  P,
  k,
  pool = "CH4",
  frac_P = frac_P_default(pool),
  frac_k = frac_k_default(pool),
  log_progress = NULL
)
```

**Arguments**

params	Named list holding optimizer-assigned values for parameters
time	Vector of numeric time values; first should be zero
m	Observed total pool size, same length as time
n	Observed pool size of heavy isotope, same length as time
m_prec	Instrument precision for pool size, expressed as a standard deviation
ap_prec	Instrument precision for atom percent, expressed as a standard deviation
P	production rate, unit pool size/unit time
k	first-order rate constant for consumption, 1/unit time
pool	Name of pool; see <a href="#">pdr_fractionation</a>
frac_P	Fractionation value for production; see <a href="#">pdr_fractionation</a>
frac_k	Fractionation value for consumption; see <a href="#">pdr_fractionation</a>
log_progress	An optional logging function

**Value**

Returns a cost metric summarizing the difference between the predicted and observed m (total pool size) and AP (atom percent).

**Note**

This implements Equations 12-14 from von Fischer and Hedin (2002).

**Author(s)**

K.A. Morris & B. Bond-Lamberty

**Examples**

```
m <- c(10, 8, 6, 5, 4, 3)
n <- c(1, 0.7, 0.6, 0.4, 0.3, 0.2)
pdr_cost(params = list(P = 0.5, k = 0.3), time = 0:5, m, n, m_prec = 0.001, ap_prec = 0.01)
```

---

pdr\_estimate\_k0

*Estimate initial k from heavy isotope concentration data*

---

**Description**

Estimate initial k from heavy isotope concentration data

**Usage**

```
pdr_estimate_k0(time, n, frac_k, quiet = FALSE)
```

**Arguments**

time	Vector of numeric time values (e.g. days); first should be zero
n	Observed heavy isotope (as a volume), same length as time
frac_k	Fractionation: $^{13}\text{C}$ consumption as a fraction of $^{12}\text{C}$ consumption
quiet	Suppress output message, logical

**Value**

Initial estimate of k0 (consumption rate constant)

**Examples**

```
pdr_estimate_k0(1:5, c(1, 0.9, 0.7, 0.65, 0.4), frac_k = 0.98)
```

---

pdr\_fractionation      *P and k fractionation values*

---

### Description

A compendium of possible production (P) and consumption (k) fractionation values, by pool.

### Usage

pdr\_fractionation

### Format

**Pool** Name of pool (gas or solid)

**frac\_P** Fractionation value of production (P)

**frac\_k** Fractionation value of consumption (k)

**Default** Default for this pool? Logical

**Source** Source paper or URL

### Note

Currently there is only one set of fractionation values available, from von Fischer and Hedin (2002, 10.1029/2001GB001448).

---

pdr\_optimize      *Optimize production and consumption parameters for pool dilution data*

---

### Description

Optimize production and consumption parameters for pool dilution data

### Usage

```
pdr_optimize(
  time,
  m,
  n,
  m_prec,
  ap_prec,
  P,
  k,
  params_to_optimize = c("P", "k"),
  pool = "CH4",
```

```

    frac_P = NULL,
    frac_k = NULL,
    other_params = list(),
    cost_fn = pdr_cost,
    prediction_fn = pdr_predict,
    include_progress = FALSE,
    quiet = FALSE
  )

```

### Arguments

time	Vector of numeric time values (e.g. days); first should be zero
m	Observed total pool size (as a volume), same length as time
n	Observed heavy isotope (as a volume), same length as time
m_prec	Instrument precision for pool size, expressed as a standard deviation
ap_prec	Instrument precision for atom percent, expressed as a standard deviation
P	production rate, unit gas/unit time
k	first-order rate constant for consumption, 1/unit time
params_to_optimize	Named vector of parameters ("P", "k", "frac_P", and/or "frac_k") to optimize against observations
pool	Name of pool to use when looking up fractionation values if they are not supplied; see <a href="#">pdr_fractionation</a>
frac_P	Fractionation value for production; see <a href="#">pdr_fractionation</a>
frac_k	Fractionation value for consumption; see <a href="#">pdr_fractionation</a>
other_params	Other parameters pass on to <a href="#">optim</a>
cost_fn	Cost function to use; the default is <a href="#">pdr_cost</a>
prediction_fn	Prediction function that the cost function will use; the default is <a href="#">pdr_predict</a>
include_progress	Include detailed optimizer progress data in output?
quiet	Suppress output messages, logical

### Value

The output of [optim](#).

### Note

Currently there is only one set of fractionation values available in [pdr\\_fractionation](#), from von Fischer and Hedin (2002, 10.1029/2001GB001448).

### See Also

[pdr\\_optimize\\_df](#)

**Examples**

```

tm <- 0:5
m <- c(10, 8, 6, 5, 4, 3)
n <- c(1, 0.7, 0.6, 0.4, 0.3, 0.2)
m_prec <- 0.001
ap_prec <- 0.01

# Optimize values for P (production) and k (consumption), provide starting values for P and k
pdr_optimize(time = tm, m, n, m_prec, ap_prec, P = 0.5, k = 0.3)
# If we don't provide a value for k, it can be estimated from the data
pdr_optimize(tm, m, n, m_prec, ap_prec, P = 0.5)
# Hold k and frac_k constant (ie., k = estimated k0, frac_k = default value), optimize P and frac_P
pdr_optimize(tm, m, n, m_prec, ap_prec, P = 0.5, params_to_optimize = c("P", "frac_P"))
# Optimize only k (provide P and exclude from params_to_optimize)
pdr_optimize(tm, m, n, m_prec, ap_prec, P = 0.5, params_to_optimize = "k")
# Optimize only k, bounding its possible values
op <- list(lower = c("k" = 0.2), upper = c("k" = 0.3))
pdr_optimize(tm, m, n, m_prec, ap_prec, 0.5, 0.27, params_to_optimize = "k", other_params = op)

```

---

pdr_optimize_df	<i>Optimize production and consumption parameters for pool dilution data</i>
-----------------	--

---

**Description**

Optimize production and consumption parameters for pool dilution data

**Usage**

```
pdr_optimize_df(...)
```

**Arguments**

... Parameters to be passed on to [pdr\\_optimize](#)

**Value**

The output of [pdr\\_optimize](#) summarized in a data frame, with one line per parameter estimates (P, k, frac\_P, and/or frac\_k).

**See Also**

[pdr\\_optimize](#)



**Examples**

```

tm <- 0:5
m <- c(10, 8, 6, 5, 4, 3)
n <- c(1, 0.7, 0.6, 0.4, 0.3, 0.2)
m_prec <- 0.001
ap_prec <- 0.01
# Optimize values for P (production) and k (consumption)
pdr_optimize_df(time = tm, m, n, m_prec, ap_prec, P = 0.5, k = 0.3)

```

---

pdr\_predict                      *Predict total pool, heavy isotope pool, and atom percent*

---

**Description**

Predict total pool, heavy isotope pool, and atom percent

**Usage**

```

pdr_predict(
  time,
  m0,
  n0,
  P,
  k,
  pool = "CH4",
  frac_P = frac_P_default(pool),
  frac_k = frac_k_default(pool)
)

```

**Arguments**

time	Vector of numeric time values (e.g. days); first should be zero
m0	total pool size at time zero, as a volume
n0	pool size of heavy isotope at time zero, as a volume
P	production rate, unit gas/unit time
k	first-order rate constant for consumption, 1/unit time
pool	Name of pool; see <a href="#">pdr_fractionation</a>
frac_P	Fractionation value for production; see <a href="#">pdr_fractionation</a>
frac_k	Fractionation value for consumption; see <a href="#">pdr_fractionation</a>

**Value**

Returns a data frame with mt, nt, and AP\_pred (atom percent) for each time step

**Note**

This is Eq. 11 from von Fischer and Hedin 2002 with a few modifications.

**Author(s)**

K.A. Morris & B. Bond-Lamberty

**Examples**

```
pdr_predict(time = 0:5, m0 = 10, n0 = 1, P = 0.5, k = 0.3)
```

# Index

## \* datasets

Morris2023, 3

pdr\_fractionation, 6

frac\_k\_default, 2

frac\_P\_default, 3

Morris2023, 3

optim, 7

pdr\_cost, 4, 7

pdr\_estimate\_k0, 5

pdr\_fractionation, 2-4, 6, 7, 9

pdr\_optimize, 6, 8

pdr\_optimize\_df, 7, 8

pdr\_predict, 7, 9