

# Package ‘clinmon’

September 29, 2020

**Type** Package

**Title** Calculation of Hemodynamic Indexes from Clinical Monitoring

**Version** 0.2.0

**Description** Every research team have their own script for calculation of hemodynamic indexes.

This package makes it possible to insert a long-format dataframe, and add both periods of interest (trigger-periods), and delete artifacts with deleter-files.

The calculation methods are based on Czosnyka et al. (1996) <doi:10.1161/01.str.27.10.1829>.

**License** MIT + file LICENSE

**URL** <https://github.com/lilleoel/clinmon>

**BugReports** <https://github.com/cosimameyer/overviewR/issues>

**Depends** R (>= 3.5.0)

**Encoding** UTF-8

**Language** en-US

**LazyData** true

**RoxygenNote** 7.1.1

**NeedsCompilation** no

**Author** Markus Harboe Olsen [cre, aut],  
Christian Riberholt [aut]

**Maintainer** Markus Harboe Olsen <oel@oelfam.com>

**Repository** CRAN

**Date/Publication** 2020-09-29 09:20:08 UTC

## R topics documented:

CO	2
CVRi	3
Dx	4
Mx	5
PI	6
PRx	7

PWA . . . . .	8
RI . . . . .	9
Sx . . . . .	10

<b>Index</b>	<b>12</b>
--------------	-----------

---

CO	<i>Cardiovascular resistance</i>
----	----------------------------------

---

## Description

Calculates the co, and further described in following publication ... [clinmon: An R package for calculation of clinical monitoring indices (2019)] Co - <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5317099/> CO (EST) = PP / (SBP+DBP) \* HR

## Usage

```
CO(df, del_1, del_2, trigger, blocksize, freq, blockmin, output)
```

## Arguments

df	Data imported with 2 columns: time (in seconds), physiological measurement (raw measurement).
del_1	Deleter periods for pressure with two columns, first column is start of deletion period and last column is end of deletion period, every row is a deletion period. (dataframe)
del_2	Deleter periods for pressure with two columns, first column is start of deletion period and last column is end of deletion period, every row is a deletion period. (dataframe)
trigger	Trigger periods with two columns, first column is start of analysed period and last column is end of analysed period, every row is a period for analysis. (dataframe)
blocksize	Size of blocks created in seconds, default is 3 s. (numeric)
freq	Frequency of recorded data, default is 1000 Hz. (numeric)
blockmin	Minimum measurements ratio in every block, default is 0.5 (numeric)
output	select the resolution of output, either 'period' or 'block' (boolean)

## Value

Returns a dataframe with the results, with either every period or block as a rows, depending on the chosen output

## Examples

```
data <- data.frame(time=seq(1, 901, 0.01),
                   pres=rnorm(90001),
                   hr=rnorm(90001))
CO(df=data, freq=1000)
```

CVRi

*Cardiovascular resistance index***Description**

Calculates the cvri, and further described in following publication ... [clinmon: An R package for calculation of clinical monitoring indices (2019)] Cvr formel - <https://europepmc.org/article/pmc/pmc6054990>  
 $CVR = \text{mean BP} / \text{mean MCAv}$

**Usage**

```
CVRi(df, del_1, del_2, trigger, blocksize, freq, blockmin, output)
```

**Arguments**

df	Data imported with 2 columns: time (in seconds), physiological measurement (raw measurement).
del_1	Deleter periods for pressure with two columns, first column is start of deletion period and last column is end of deletion period, every row is a deletion period. (dataframe)
del_2	Deleter periods for pressure with two columns, first column is start of deletion period and last column is end of deletion period, every row is a deletion period. (dataframe)
trigger	Trigger periods with two columns, first column is start of analysed period and last column is end of analysed period, every row is a period for analysis. (dataframe)
blocksize	Size of blocks created in seconds, default is 3 s. (numeric)
freq	Frequency of recorded data, default is 1000 Hz. (numeric)
blockmin	Minimum measurements ratio in every block, default is 0.5 (numeric)
output	select the resolution of output, either 'period' or 'block' (boolean)

**Value**

Returns a dataframe with the results, with either every period or block as a rows, depending on the chosen output

**Examples**

```
data <- data.frame(time=seq(1, 901, 0.01),
                   pres=rnorm(90001),
                   mcav=rnorm(90001))
CVRi(df=data, freq=100)
```

---

Dx	<i>Diastolic flow velocity index (Dx)</i>
----	---

---

## Description

Calculates Dx, and further described in following publication ... [clinmon: An R package for calculation of clinical monitoring indices (2020)]

## Usage

```
Dx(df, del_1 = NULL, del_2 = NULL,
    trigger = NULL, blocksize = 3,
    epochsize = 20, freq, blockmin = 0.5,
    epochmin = 0.5, overlapping = FALSE,
    output = 'period')
```

## Arguments

df	Raw recording with data in 3 columns: time (in seconds), pressure curve measurement (e.g. CPP, raw measurement), and velocity curve measurement (e.g. MCAv, raw measurement). <i>[dataframe]</i>
del_1	Deleter with two columns, reference to pressure curve measurement: (1) start of deletion period and (2) end of deletion period. Every row is a deletion period. <i>[dataframe]</i>
del_2	Deleter with two columns, reference to velocity curve measurement: (1) start of deletion period and (2) end of deletion period. Every row is a deletion period. <i>[dataframe]</i>
trigger	Trigger with two columns: (1) start of analysed period and (2) end of analysed period. Every row is a period for analysis. <i>[dataframe]</i>
blocksize	Size of blocks, in seconds. <i>[numeric]</i>
epochsize	Size of epochs, in number of blocks. <i>[numeric]</i>
freq	Frequency of recorded data, in Hz. <i>[numeric]</i>
blockmin	Minimum measurements required to create a block <i>[numeric]</i>
epochmin	Minimum blocks required to create an epoch <i>[numeric]</i>
overlapping	The number of block which should overlap, and remain blank if overlapping should not be utilized <i>numeric</i>
output	Select the output which has to be either one row per 'period' or 'epoch'. <i>[boolean]</i>

## Value

Returns a dataframe with the results, with either every period or epoch as a rows, depending on the chosen output

## Examples

```
data <- data.frame(time=seq(1, 901, 0.01),
                  pres=rnorm(90001),
                  mcav=rnorm(90001))
Dx(df=data, freq=1000)
```

---

Mx	<i>Mean flow velocity index (Mx)</i>
----	--------------------------------------

---

## Description

Using raw recording, this functions calculates mean flow velocity index (Mx), initially described by Czosnyka et al. in 1996. [clinmon: An R package for calculation of clinical monitoring indices (2020)]

## Usage

```
Mx(df, del_1 = NULL, del_2 = NULL,
    trigger = NULL, blocksize = 3,
    epochsize = 20, freq, blockmin = 0.5,
    epochmin = 0.5, overlapping = FALSE,
    output = 'period')
```

## Arguments

df	Raw recording with data in 3 columns: time (in seconds), pressure curve measurement (e.g. CPP, raw measurement), and velocity curve measurement (e.g. MCAv, raw measurement). <i>[dataframe]</i>
del_1	Deleter with two columns, reference to pressure curve measurement: (1) start of deletion period and (2) end of deletion period. Every row is a deletion period. <i>[dataframe]</i>
del_2	Deleter with two columns, reference to velocity curve measurement: (1) start of deletion period and (2) end of deletion period. Every row is a deletion period. <i>[dataframe]</i>
trigger	Trigger with two columns: (1) start of analysed period and (2) end of analysed period. Every row is a period for analysis. <i>[dataframe]</i>
blocksize	Size of blocks, in seconds. <i>[numeric]</i>
epochsize	Size of epochs, in number of blocks. <i>[numeric]</i>
freq	Frequency of recorded data, in Hz. <i>[numeric]</i>
blockmin	Minimum measurements required to create a block <i>[numeric]</i>
epochmin	Minimum blocks required to create an epoch <i>[numeric]</i>
overlapping	The number of block which should overlap, and remain blank if overlapping should not be utilized <i>numeric</i>
output	Select the output which has to be either one row per 'period' or 'epoch'. <i>[boolean]</i>

**Value**

Returns a dataframe with the results, with either every period or epoch as a rows, depending on the chosen output

**Examples**

```
data <- data.frame(time=seq(1, 901, 0.01),
                   pres=rnorm(90001),
                   mcav=rnorm(90001))
Mx(df=data, freq=1000)
```

---

PI	<i>Pulsatility index (Gosling's index, PI)</i>
----	--

---

**Description**

Calculates the pulsatility index, and further described in following publication ... [clinmon: An R package for calculation of clinical monitoring indices (2020)]

**Usage**

```
PI(df, del = NULL, trigger = NULL,
   blocksize = 3, freq, blockmin = 0.5,
   output = 'period')
```

**Arguments**

df	Raw recording with data in 2 columns: time (in seconds) and pressure/velocity curve measurement (e.g. ABP, raw measurement). <i>[dataframe]</i>
del	Deleter with two columns: (1) start of deletion period and (2) end of deletion period. Every row is a deletion period. <i>[dataframe]</i>
trigger	Trigger with two columns: (1) start of analysed period and (2) end of analysed period. Every row is a period for analysis. <i>[dataframe]</i>
blocksize	Size of blocks, in seconds. <i>[numeric]</i>
freq	Frequency of recorded data, in Hz. <i>[numeric]</i>
blockmin	Minimum measurements required to create a block <i>[numeric]</i>
output	Select the output which has to be either one row per 'period' or 'block'. <i>[boolean]</i>

**Value**

Returns a dataframe with the results, with either every period or block as a rows, depending on the chosen output

## Examples

```
data <- data.frame(time=seq(1, 901, 0.01),
                  pres=rnorm(90001))
PI(df=data, freq=100)
```

PRx

*Pressure reactivity index (PRx)*

## Description

Calculates PRx, and further described in following publication ... [clinmon: An R package for calculation of clinical monitoring indices (2020)]

## Usage

```
PRx(df, del_1 = NULL, del_2 = NULL,
    trigger = NULL, blocksize = 3,
    epochsize = 20, freq, blockmin = 0.5,
    epochmin = 0.5, overlapping = FALSE,
    output = 'period')
```

## Arguments

df	Raw recording with data in 3 columns: time (in seconds), arterial blood pressure curve measurement (ABP, raw measurement), and intracranial pressure curve measurement (ICP, raw measurement). <i>[dataframe]</i>
del_1	Deleter with two columns, reference to arterial blood pressure curve measurement: (1) start of deletion period and (2) end of deletion period. Every row is a deletion period. <i>[dataframe]</i>
del_2	Deleter with two columns, reference to intracranial pressure curve measurement: (1) start of deletion period and (2) end of deletion period. Every row is a deletion period. <i>[dataframe]</i>
trigger	Trigger with two columns: (1) start of analysed period and (2) end of analysed period. Every row is a period for analysis. <i>[dataframe]</i>
blocksize	Size of blocks, in seconds. <i>[numeric]</i>
epochsize	Size of epochs, in number of blocks. <i>[numeric]</i>
freq	Frequency of recorded data, in Hz. <i>[numeric]</i>
blockmin	Minimum measurements required to create a block <i>[numeric]</i>
epochmin	Minimum blocks required to create an epoch <i>[numeric]</i>
overlapping	The number of block which should overlap, and remain blank if overlapping should not be utilized <i>numeric</i>
output	Select the output which has to be either one row per 'period' or 'epoch'. <i>[boolean]</i>

**Value**

Returns a dataframe with the results, with either every period or epoch as a rows, depending on the chosen output

**Examples**

```
data <- data.frame(time=seq(1, 901, 0.01),
                  pres=rnorm(90001),
                  mcav=rnorm(90001))
PRx(df=data, freq=1000)
```

PWA

*Pulse wave amplitude (PWA/AMP)***Description**

Calculates the pulsw ave amplitude, and further described in following publication ... [clinmon: An R package for calculation of clinical monitoring indices (2020)]

**Usage**

```
PWA(df, del = NULL, trigger = NULL,
    blocksize = 3, freq, blockmin = 0.5,
    output = 'period')
```

**Arguments**

df	Raw recording with data in 2 columns: time (in seconds) and pressure/velocity curve measurement (e.g. ABP, raw measurement). <i>[dataframe]</i>
del	Deleter with two columns: (1) start of deletion period and (2) end of deletion period. Every row is a deletion period. <i>[dataframe]</i>
trigger	Trigger with two columns: (1) start of analysed period and (2) end of analysed period. Every row is a period for analysis. <i>[dataframe]</i>
blocksize	Size of blocks, in seconds. <i>[numeric]</i>
freq	Frequency of recorded data, in Hz. <i>[numeric]</i>
blockmin	Minimum measurements required to create a block <i>[numeric]</i>
output	Select the output which has to be either one row per 'period' or 'block'. <i>[boolean]</i>

**Value**

Returns a dataframe with the results, with either every period or block as a rows, depending on the chosen output



### Examples

```
data <- data.frame(time=seq(1, 901, 0.01),
                  pres=rnorm(90001))
PWA(df=data, freq=100)
```

---

RI	<i>Resistance index (RI)</i>
----	------------------------------

---

### Description

Calculates the resistance index, and further described in following publication ... [clinmon: An R package for calculation of clinical monitoring indices (2020)]

### Usage

```
RI(df, del = NULL, trigger = NULL,
   blocksize = 3, freq, blockmin = 0.5,
   output = "period")
```

### Arguments

df	Raw recording with data in 2 columns: time (in seconds) and CBFv measurement (e.g. MCAv, raw measurement). <i>[dataframe]</i>
del	Deleter with two columns: (1) start of deletion period and (2) end of deletion period. Every row is a deletion period. <i>[dataframe]</i>
trigger	Trigger with two columns: (1) start of analysed period and (2) end of analysed period. Every row is a period for analysis. <i>[dataframe]</i>
blocksize	Size of blocks, in seconds. <i>[numeric]</i>
freq	Frequency of recorded data, in Hz. <i>[numeric]</i>
blockmin	Minimum measurements required to create a block <i>[numeric]</i>
output	Select the output which has to be either one row per 'period' or 'block'. <i>[boolean]</i>

### Value

Returns a dataframe with the results, with either every period or block as a rows, depending on the chosen output

### Examples

```
data <- data.frame(time=seq(1, 901, 0.01),
                  pres=rnorm(90001))
RI(df=data, freq=100)
```

---

Sx	<i>Systolic flow velocity index (Sx)</i>
----	--

---

## Description

Calculates Sx, and further described in following publication ... [clinmon: An R package for calculation of clinical monitoring indices (2020)]

## Usage

```
Sx(df, del_1 = NULL, del_2 = NULL,
    trigger = NULL, blocksize = 3,
    epochsize = 20, freq, blockmin = 0.5,
    epochmin = 0.5, overlapping = FALSE,
    output = 'period')
```

## Arguments

df	Raw recording with data in 3 columns: time (in seconds), pressure curve measurement (e.g. CPP, raw measurement), and velocity curve measurement (e.g. MCAv, raw measurement). <i>[dataframe]</i>
del_1	Deleter with two columns, reference to pressure curve measurement: (1) start of deletion period and (2) end of deletion period. Every row is a deletion period. <i>[dataframe]</i>
del_2	Deleter with two columns, reference to velocity curve measurement: (1) start of deletion period and (2) end of deletion period. Every row is a deletion period. <i>[dataframe]</i>
trigger	Trigger with two columns: (1) start of analysed period and (2) end of analysed period. Every row is a period for analysis. <i>[dataframe]</i>
blocksize	Size of blocks, in seconds. <i>[numeric]</i>
epochsize	Size of epochs, in number of blocks. <i>[numeric]</i>
freq	Frequency of recorded data, in Hz. <i>[numeric]</i>
blockmin	Minimum measurements required to create a block <i>[numeric]</i>
epochmin	Minimum blocks required to create an epoch <i>[numeric]</i>
overlapping	The number of block which should overlap, and remain blank if overlapping should not be utilized <i>numeric</i>
output	Select the output which has to be either one row per 'period' or 'epoch'. <i>[boolean]</i>

## Value

Returns a dataframe with the results, with either every period or epoch as a rows, depending on the chosen output

**Examples**

```
data <- data.frame(time=seq(1, 901, 0.01),  
                  pres=rnorm(90001),  
                  mcav=rnorm(90001))  
Sx(df=data, freq=1000)
```

# Index

CO, [2](#)  
CVRi, [3](#)  
  
Dx, [4](#)  
  
Mx, [5](#)  
  
PI, [6](#)  
PRx, [7](#)  
PWA, [8](#)  
  
RI, [9](#)  
  
Sx, [10](#)