

Package ‘cgmquantify’

July 22, 2025

Type Package

Title Analyzing Glucose and Glucose Variability

Version 0.1.0

Imports dplyr, tidyverse, ggplot2, hms, stats, magrittr

Description Continuous glucose monitoring (CGM) systems provide real-time, dynamic glucose information by tracking interstitial glucose values throughout the day. Glycemic variability, also known as glucose variability, is an established risk factor for hypoglycemia (Kovatchev) and has been shown to be a risk factor in diabetes complications. Over 20 metrics of glycemic variability have been identified. Here, we provide functions to calculate glucose summary metrics, glucose variability metrics (as defined in clinical publications), and visualizations to visualize trends in CGM data.
Cho P, Bent B, Wittmann A, et al. (2020) <https://diabetes.diabetesjournals.org/content/69/Supplement_1/73-LB.abstract>
American Diabetes Association (2020) <https://professional.diabetes.org/diapro/glucose_calc>
Kovatchev B (2019) <[doi:10.1177/1932296819826111](https://doi.org/10.1177/1932296819826111)>
Kovdeatchev BP (2017) <[doi:10.1038/nrendo.2017.3](https://doi.org/10.1038/nrendo.2017.3)>
Tamborlane W V., Beck RW, Bode BW, et al. (2008) <[doi:10.1056/NEJMoa0805017](https://doi.org/10.1056/NEJMoa0805017)>
Umpierrez GE, P. Kovatchev B (2018) <[doi:10.1016/j.amjms.2018.09.010](https://doi.org/10.1016/j.amjms.2018.09.010)>.

License MIT License + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Suggests testthat (>= 2.0.0), knitr, rmarkdown

Config/testthat/edition 2, devtools

VignetteBuilder knitr

Depends R (>= 2.10)

NeedsCompilation no

Author Maria Henriquez [aut, com, cph, cre, trl],
Brinnae Bent [aut, cph, dtc]

Maintainer Maria Henriquez <marhenriq@gmail.com>

Repository CRAN

Date/Publication 2021-02-05 08:50:02 UTC

Contents

eA1c	2
GMI	3
HBGI	3
interdaycv	4
interdaysd	4
intradaycv	5
intradaysd	6
J_index	6
LBGI	7
LBGI_HBGI	7
MGE	8
MGN	9
plot_glucose	9
POR	10
readfile	10
summary_glucose	11
TIR	12
TOR	12

Index	14
--------------	-----------

eA1c	<i>Compute Estimated A1c</i>
------	------------------------------

Description

This function computes the estimated A1c, according to the American Diabetes Association calculator

Usage

```
eA1c(df)
```

Arguments

df Data frame read through readfile

Value

A numeric value representing eA1c

Examples

```
mydatafile <- system.file("extdata", "my_data_file.csv", package = "cgmquantify")
mydata <- readfile(mydatafile)
eA1c(mydata)
```

GMI*Compute Glycemic Management Indicator*

Description

This function computes the estimated GMI

Usage

```
GMI(df)
```

Arguments

df Data frame read through readfile

Value

A numeric value representing GMI

Examples

```
mydatafile <- system.file("extdata", "my_data_file.csv", package = "cgmquantify")
mydata <- readfile(mydatafile)
GMI(mydata)
```

HBGI*Compute High Blood Glucose Index*

Description

This function computes the high blood glucose index

Usage

```
HBGI(df)
```

Arguments

df Data frame read through readfile

Value

A numeric value representing HBGI

Examples

```
mydatafile <- system.file("extdata", "my_data_file.csv", package = "cgmquantify")
mydata <- readfile(mydatafile)
HBGI(mydata)
```

interdaycv	<i>Compute Interday Coefficient of Variation</i>
------------	--

Description

This function computes the interday coefficient of variation

Usage

```
interdaycv(df)
```

Arguments

df	Data frame read through readfile
----	----------------------------------

Value

A numeric value representing interday cv

Examples

```
mydatafile <- system.file("extdata", "my_data_file.csv", package = "cgmquantify")
mydata <- readfile(mydatafile)
interdaycv(mydata)
```

interdaysd	<i>Compute Interday Standard Deviation</i>
------------	--

Description

This function computes the interday standard deviation

Usage

```
interdaysd(df)
```

Arguments

df Data frame read through readfile

Value

A numeric value representing interday sd

Examples

```
mydatafile <- system.file("extdata", "my_data_file.csv", package = "cgmquantify")
mydata <- readfile(mydatafile)
interdaysd(mydata)
```

intradaycv	<i>Compute Intraday Coefficient of Variation</i>
------------	--

Description

This function computes the intraday coefficient of variation summary statistics: mean, median, standard deviation of all days in data

Usage

```
intradaycv(df)
```

Arguments

df Data frame read through readfile

Value

A data frame containing the mean, median, and standard deviation of the intraday coefficients of variation.

Examples

```
mydatafile <- system.file("extdata", "my_data_file.csv", package = "cgmquantify")
mydata <- readfile(mydatafile)
intradaycv(mydata)
```

`intradaysd`*Compute Intraday Standard Deviation*

Description

This function computes the intraday standard deviation summary statistics: mean, median, standard deviation of all days in data

Usage

```
intradaysd(df)
```

Arguments

`df` Data frame read through readfile

Value

A data frame containing the mean, median, and standard deviation of the intraday standard deviations.

Examples

```
mydatafile <- system.file("extdata", "my_data_file.csv", package = "cgmquantify")
mydata <- readfile(mydatafile)
intradaysd(mydata)
```

`J_index`*Compute J-index*

Description

This function computes J-index, a glycemic variability metrix

Usage

```
J_index(df)
```

Arguments

`df` Data frame read through readfile

Value

A numeric value representing J-index

Examples

```
mydatafile <- system.file("extdata", "my_data_file.csv", package = "cgmquantify")
mydata <- readfile(mydatafile)
J_index(mydata)
```

LBGI*Compute Low Blood Glucose Index*

Description

This function computes the low blood glucose index

Usage

```
LBGI(df)
```

Arguments

df Data frame read through readfile

Value

A numeric value representing LBGI

Examples

```
mydatafile <- system.file("extdata", "my_data_file.csv", package = "cgmquantify")
mydata <- readfile(mydatafile)
LBGI(mydata)
```

LBGI_HBGI*Compute Low Blood Glucose Index*

Description

This function computes the low blood glucose index

Usage

```
LBGI_HBGI(df)
```

Arguments

df Data frame read through readfile

Value

A data frame containing both the LBG1 and HBGI values

Examples

```
mydatafile <- system.file("extdata", "my_data_file.csv", package = "cgmquantify")
mydata <- readfile(mydatafile)
LBGI_HBGI(mydata)
```

MGE

Compute Mean of Glycemic Excursions

Description

This function computes the mean of glycemic excursions, glycemic excursions indicated by standard deviation, default = 1

Usage

```
MGE(df, sd = 1)
```

Arguments

df	Data frame read through readfile
sd	Standard deviation indicating glycemic excursion, default = 1

Value

A numeric value representing MAGE

Examples

```
mydatafile <- system.file("extdata", "my_data_file.csv", package = "cgmquantify")
mydata <- readfile(mydatafile)
MGE(mydata)
```

MGN*Compute Mean of Normal Glucose*

Description

This function computes the mean of normal glucose, glycemic excursions indicated by standard deviation, default = 1

Usage

```
MGN(df)
```

Arguments

df Data frame read through readfile

Value

A numeric value representing MGN

Examples

```
mydatafile <- system.file("extdata", "my_data_file.csv", package = "cgmquantify")
mydata <- readfile(mydatafile)
MGN(mydata)
```

plot_glucose*Plot Glucose Data*

Description

This function plots glycemic excursions over the time period in which data was collected.

Usage

```
plot_glucose(df)
```

Arguments

df Data frame read through readfile

Value

None

Examples

```
mydatafile <- system.file("extdata", "my_data_file.csv", package = "cgmquantify")
mydata <- readfile(mydatafile)
plot_glucose(mydata)
```

POR

Compute Percent of Time Outside Range

Description

This function computes the percent of time outside range (range in standard deviations from mean, default = 1).

Usage

```
POR(df, sd = 1, sr = 5)
```

Arguments

df	Data frame read through readfile
sd	Standard deviation indicating glycemic excursion, default = 1
sr	Sampling rate inverse in minutes of the CGM (default is Dexcom -> 5 minutes)

Value

A numeric value representing POR

Examples

```
mydatafile <- system.file("extdata", "my_data_file.csv", package = "cgmquantify")
mydata <- readfile(mydatafile)
POR(mydata, sd = 1, sr = 5)
```

readfile

Read in Data Frame

Description

This function reads in a .csv with variable names Timestamp..YYYY.MM.DDThh.mm.ss and Glucose.Value..mg.dL

Usage

```
readfile(filename)
```

Arguments

filename .csv file of data frame to be read

Value

transformed data frame for further analysis

Examples

```
mydatafile <- system.file("extdata", "my_data_file.csv", package = "cgmquantify")
readfile(mydatafile)
```

summary_glucose	<i>Compute Glucose Summary Statistics</i>
-----------------	---

Description

This function computes the mean, median, minimum, maximum, first quartile, and the third quartile of an individual's overall glucose levels

Usage

```
summary_glucose(df)
```

Arguments

df Data frame read through readfile

Value

A dataframe containing the mean, median, minimum, maximum, quartile1, and quartile3 of glucose levels

Examples

```
mydatafile <- system.file("extdata", "my_data_file.csv", package = "cgmquantify")
mydata <- readfile(mydatafile)
summary_glucose(mydata)
```

TIR	<i>Compute Time Inside Range</i>
-----	----------------------------------

Description

This function computes the time inside range (range in standard deviations from mean, default = 1).

Usage

```
TIR(df, sd = 1, sr = 5)
```

Arguments

df	Data frame read through readfile
sd	Standard deviation indicating glycemic excursions, default = 1
sr	Sampling rate inverse in minutes of the CGM (default is the Dexcom -> 5 minutes)

Value

A numeric value representing TIR

Examples

```
mydatafile <- system.file("extdata", "my_data_file.csv", package = "cgmquantify")
mydata <- readfile(mydatafile)
TIR(mydata, sd = 1, sr = 5)
```

TOR	<i>Compute Time Outside Range</i>
-----	-----------------------------------

Description

This function computes the time outside range (range in standard deviations from mean, default = 1).

Usage

```
TOR(df, sd = 1, sr = 5)
```

Arguments

df	Data frame read through readfile
sd	Standard deviation indicating glycemic excursions, default = 1
sr	Sampling rate inverse in minutes of the CGM (default is the Dexcom -> 5 minutes)

Value

A numeric value representing TOR

Examples

```
mydatafile <- system.file("extdata", "my_data_file.csv", package = "cgmquantify")  
mydata <- readfile(mydatafile)  
TOR(mydata, sd = 1, sr = 5)
```

Index

eA1c, [2](#)

GMI, [3](#)

HBGI, [3](#)

interdaycv, [4](#)

interdaysd, [4](#)

intradaycv, [5](#)

intradaysd, [6](#)

J_index, [6](#)

LBGI, [7](#)

LBGI_HBGI, [7](#)

MGE, [8](#)

MGN, [9](#)

plot_glucose, [9](#)

POR, [10](#)

readfile, [10](#)

summary_glucose, [11](#)

TIR, [12](#)

TOR, [12](#)