Package: insulin.secretion (via r-universe)

May 26, 2026

Title Insulin Secretion Rate Deconvolution

Version 0.0.2

Description Calculates insulin secretion rates from C-peptide values based on the methods described in Van Cauter et al. (1992) <doi:10.2337/diab.41.3.368>. Includes functions to calculate estimated insulin secretion rates using linear or cubic spline interpolation of c-peptide values (see Eaton et al., 1980 <doi:10.1210/jcem-51-3-520> and Polonsky et al., 1986 <doi:10.1172/JCI112308>) and to calculate estimates of input coefficients (volume of distribution, short half life, long half life, and fraction attributed to short half life) as described by Van Cauter. Although the generated coefficients are specific to insulin secretion, the two-compartment secretion model used here is useful for certain applications beyond insulin.

License GPL (>= 3)

Imports glue, lifecycle, npreg, rlang

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

URL `https://github.com/kstier/isr.deconv`

BugReports `https://github.com/kstier/isr.deconv/issues`

Repository https://kstier.r-universe.dev

Date/Publication 2026-01-26 18:07:55 UTC

RemoteUrl https://github.com/kstier/isr.deconv

RemoteRef HEAD

RemoteSha 320497e33cbdff2cc7f7d9f0c370080ab5a00229

Contents

`isr.deconv` *Insulin Secretion Rate Deconvolution*

Description

Estimates insulin secretion rate based on a time series of c-peptide values based on the Van Cauter method. C-peptide values are interpolated using either linear approximation or a cubic spline; the linear method (default) will output a time series of insulin secretory rates at time points between each input time, whereas the spline method will output a function which can be called to return insulin secretory rates for specific time points.

Typically, c-peptide values are provided in pmol/mL and time is in minutes, resulting in insulin secretion rate outputs in pmol/min.

Although this function was designed with insulin secretion in mind, it uses a two-compartment secretion model that may be applied beyond insulin.

Usage

```
isr.deconv(
  timeseries,
  cpepseries,
  vol,
  shl,
  lhl,
  frc,
  method = c("linear", "spline"),
  plotspline = FALSE,
  plotisr = FALSE,
  isr.validated.vals,
  isr.validated.time
)
```

Arguments

`timeseries`	Vector of numeric time values corresponding to the c-peptide values in `cpepseries`
`cpepseries`	Vector of numeric c-peptide values at each time point in `timeseries`
`vol`	Volume of distribution in the main compartment (i.e., serum volume), which can be calculated with the `isr.volume()` function

`shl`	Short half-life, which can be calculated with the `isr.shortHL()` function
`lhl`	Long half-life, which can be calculated with the `isr.longHL()` function
`frc`	Fraction attributable to the short half-life, which can be calculated with the `isr.fraction()` function
`method`	Determines whether C-peptide values are interpolated with linear connections between the points or a cubic spline
`plotspline`	Dictates whether to print a plot of the C-peptide interpolation
`plotisr`	Dictates whether to print a plot of the ISR output values
`isr.validated.vals`	Used for debugging; validated ISR values to compare to past analyses
`isr.validated.time`	Used for debugging; time values corresponding to validated ISR values to compare to past analyses

Value

If `method = "linear"` is selected (default), returns a list of time points and insulin secretory rates. If `method = "spline"` is selected, returns a function which can be called to return insulin secretory rates at specified time points.

See Also

`isr.volume()`, `isr.shortHL()`, `isr.longHL()`, `isr.fraction()`

Examples

```
isr.deconv(
  timeseries = c(-30, 0, 30, 60, 90, 120),
  cpepseries = c(1.72, 1.72, 5.40, 5.23, 2.71, 1.79),
  vol = 6104,
  shl = 4.55,
  lhl = 31.05,
  frc = 0.78,
  method = "linear",
  plotspline = TRUE,
  plotisr = TRUE,
)
```

`isr.fraction` *C-Peptide Fraction Attributable to Short Half Life Lookup*

Description

Outputs the relevant fraction attributable to short half life for use in `isr.deconv()` based on patient type (normal, obese, or non-insulin dependent diabetes mellitus).

Usage

```
isr.fraction(subject.type = c("normal", "obese", "niddm"))
```

Arguments

`subject.type`	String for patient type.

Value

Numeric value.

See Also

`isr.deconv()`

Examples

```
isr.fraction("normal")

isr.fraction("obese")

isr.fraction("niddm")
```

`isr.longHL` *C-Peptide Long Half Life Calculator*

Description

Outputs the long half life based on patient age for use in `isr.deconv()` per the Van Cauter method of estimating insulin secretion rate.

Usage

```
isr.longHL(subject.age)
```

Arguments

`subject.age`	Numeric for patient age in years.

Value

Numeric value.

See Also

`isr.deconv()`

Examples

```
isr.longHL(18.08)
```

`isr.shortHL` *C-Peptide Short Half Life Calculator*

Description

Outputs the relevant long half life of c-peptide for use in `isr.deconv()` based on patient type (normal, obese, or non-insulin dependent diabetes mellitus).

Usage

```
isr.shortHL(subject.type = c("normal", "obese", "niddm"))
```

Arguments

`subject.type`	String for patient type.

Value

Numeric value.

See Also

`isr.deconv()`

Examples

```
isr.shortHL("normal")

isr.shortHL("obese")

isr.shortHL("niddm")
```

`isr.volume` *Serum Volume Calculator*

Description

Calculates the estimated serum volume (mL) based on patient sex, weight, and height for use in `isr.deconv()`.

Usage

```
isr.volume(subject.sex = c("m", "f"), subject.weight, subject.height)
```

Arguments

`subject.sex`	String for patient sex, `"m"` or `"f"`.
`subject.weight`	Numeric for subject weight in kilograms.
`subject.height`	Numeric for subject height in centimeters.

Value

Numeric value.

See Also

`isr.deconv()`

Examples

```
isr.volume("m", 86.2, 181.5)
```

Index