Package 'contfrac'

October 13, 2026

Title Continued Fractions

Depends R (>= 4.2.0)

Version 1.1-13

Description Various utilities for evaluating continued fractions.

Maintainer Robin K. S. Hankin <hankin.robin@gmail.com>

License GPL-2

URL https://github.com/RobinHankin/contfrac

BugReports https://github.com/RobinHankin/contfrac/issues

Suggests testthat (>= 3.0.0)

NeedsCompilation yes

Author Robin K. S. Hankin [aut, cre] (ORCID:
<https://orcid.org/0000-0001-5982-0415>)

Repository CRAN

Date/Publication 2026-10-13 15:20:15 UTC

Contents

contfrac-package 2
as_cf 3
CF 4
convergents 5

Index **8**

contfrac-package *Continued Fractions*

Description

Various utilities for evaluating continued fractions.

Details

The DESCRIPTION file:

Package:	contfrac
Title:	Continued Fractions
Depends:	R (>= 4.2.0)
Version:	1.1-13
Authors@R:	person(given=c("Robin", "K. S."), family="Hankin", role = c("aut","cre"), email="hankin.robin@gmail.com",
Description:	Various utilities for evaluating continued fractions.
Maintainer:	Robin K. S. Hankin <hankin.robin@gmail.com>
License:	GPL-2
URL:	https://github.com/RobinHankin/contfrac
BugReports:	https://github.com/RobinHankin/contfrac/issues
Suggests:	testthat (>= 3.0.0)
Author:	Robin K. S. Hankin [aut, cre] (ORCID: <https://orcid.org/0000-0001-5982-0415>)

Index of help topics:


```
CF                      Continued fraction convergents
as_cf                   Approximates a real number in continued
                        fraction form
contfrac-package        Continued Fractions
convergents             Partial convergents of continued fractions
```


Author(s)

Robin K. S. Hankin [aut, cre] (ORCID: <https://orcid.org/0000-0001-5982-0415>)

Examples

```
## CF() takes an integer sequence and returns the value of its continued fraction:
phi <- (sqrt(5)+1)/2
phi_cf <- CF(rep(1,100))     # phi = [1;1,1,1,1,1,...]
phi - phi_cf     # should be small


## as_cf() takes a real and returns its continued fraction representation:
as_cf(phi)
```

```
as_cf(pi)
as_cf(exp(1),25)    # OK up to element 21 (which should be 14)

## GCF() is a generalized continued fraction:
GCF(a=2:100,b=2:100,b0=1,finite=FALSE)  # This due to Euler

## convergents() gives a sequence of partial convergents:
convergents(rep(1,10))
```

---

as_cf *Approximates a real number in continued fraction form*

Description

Approximates a real number in continued fraction form using a standard simple algorithm

Usage

```
as_cf(x, n = 10)
```

Arguments

x	real number to be approximated in continued fraction form
n	Number of partial denominators to evaluate; see Notes

Note

Has difficulties with rational values as expected

Author(s)

Robin K. S. Hankin

See Also

CF,convergents

Examples

```
phi <- (sqrt(5)+1)/2
as_cf(phi,50)  # loses it after about 38 iterations ... not bad ...

as_cf(pi)  # looks about right
as_cf(exp(1),20)

f <- function(x){CF(as_cf(x,30),TRUE) - x}

x <- runif(40)
plot(sapply(x,f))
```

CF *Continued fraction convergents*

Description

Returns continued fraction convergent using the modified Lentz's algorithm; function `CF()` deals with continued fractions and `GCF()` deals with generalized continued fractions.

Usage

```
CF(a, finite = FALSE, tol=0)
GCF(a,b, b0=0, finite = FALSE, tol=0)
```

Arguments

a, b	In function `CF()`, the elements of `a` are the partial denominators; in `GCF()` the elements of `a` are the partial numerators and the elements of `b` the partial denominators
finite	Boolean, with default `FALSE` meaning to iterate Lentz's algorithm until convergence (a warning is given if the sequence has not converged); and `TRUE` meaning to evaluate the finite continued fraction
b0	In function `GCF()`, floor of the continued fraction
tol	tolerance, with default `0` silently replaced with `.Machine$double.eps`

Details

Function `CF()` takes the first element of its argument to be the integer part of the convergent.

Function `CF()` is a wrapper for `GCF()`; it includes special dispensation for infinite values (in which case the value of the appropriate finite CF is returned).

The implementation is in C; the real and complex cases are treated separately in the interests of efficiency.

The algorithm terminates when the convergence criterion is achieved irrespective of the value of `finite`.

Author(s)

Robin K. S. Hankin

References

- W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling 1992. *Numerical recipes 3rd edition: the art of scientific computing*. Cambridge University Press; section 5.2 "Evaluation of continued fractions"
- W. J. Lentz 1976. Generating Bessel functions in Mie scattering calculations using continued fractions. *Applied Optics*, 15(3):668-671

See Also

convergents

Examples

```
phi <- (sqrt(5)+1)/2
phi_cf <- CF(rep(1,100))     # phi = [1;1,1,1,1,1,...]
phi - phi_cf     # should be small

# The tan function:
"tan_cf" <- function(z,n=20){
     GCF(c(z, rep(-z^2,n-1)), seq(from=1,by=2, len=n))
}

z <- 1+1i
tan(z) - tan_cf(z)   # should be small

# approximate real numbers with continued fraction:
as_cf(pi)

as_cf(exp(1),25)    # OK up to element 21 (which should be 14)

  # Some convergents of pi:
  jj <- convergents(c(3,7,15,1,292))
  jj$A / jj$B - pi


  # An identity of Euler's:
  jj <- GCF(a=seq(from=2,by=2,len=30), b=seq(from=3,by=2,len=30), b0=1)
  jj - 1/(exp(0.5)-1)   # should be small
```

---

convergents *Partial convergents of continued fractions*

Description

Partial convergents of continued fractions or generalized continued fractions

Usage

```
convergents(a)
gconvergents(a,b, b0 = 0)
nconv(a, give=FALSE)
ngconv(a, b, b0 = 0, give=FALSE)
```

Arguments

a, b	In function convergents(), the elements of a are the partial denominators (the first element of a is the integer part of the continued fraction). In gconvergents() the elements of a are the partial numerators and the elements of b the partial denominators
b0	The floor of the fraction
give	Boolean, with TRUE meaning to return all convergents, and default FALSE meaning to return the final partial convergent

Details

Function convergents() returns partial convergents of the continued fraction

$$a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{a_3 + \cfrac{1}{a_4 + \cfrac{1}{a_5 + \ddots}}}}}$$

where a = $a_0, a_1, a_2, \ldots$ (note the off-by-one issue).

Function gconvergents() returns partial convergents of the continued fraction

$$b_0 + \cfrac{a_1}{b_1 + \cfrac{a_2}{b_2 + \cfrac{a_3}{b_3 + \cfrac{a_4}{b_4 + \cfrac{a_5}{b_5 + \ddots}}}}}$$

where a = $a_1, a_2, \ldots$

Functions nconv() and ngconv() are convenience wrappers that return the numerical values of the convergents.

Value

Returns a list of two elements, A for the numerators and B for the denominators

Note

This classical algorithm generates very large partial numerators and denominators. To evaluate limits, use functions CF() or GCF().

Author(s)

Robin K. S. Hankin

References

W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling 1992. *Numerical recipes 3rd edition: the art of scientific computing*. Cambridge University Press; section 5.2 "Evaluation of continued fractions"

See Also

CF

Examples

```
# Successive approximations to pi:

jj <- convergents(c(3,7,15,1,292))
jj$A/jj$B - pi     # should get smaller


convergents(rep(1,10))

nconv(1:6,give=TRUE)
```

Index

∗ **math**
as_cf, 3
CF, 4
convergents, 5
∗ **package**
contfrac-package, 2

as_cf, 3

c_contfrac *(*convergents*)*, 5
c_contfrac_complex *(*convergents*)*, 5
c_convergents *(*convergents*)*, 5
c_convergents_complex *(*convergents*)*, 5
CF, *3*, 4, *7*
contfrac *(*contfrac-package*)*, 2
contfrac-package, 2
convergents, *3*, *5*, 5

GCF *(*CF*)*, 4
gconvergents *(*convergents*)*, 5

nconv *(*convergents*)*, 5
ngconv *(*convergents*)*, 5