

Package ‘RFIF’

April 10, 2026

Type Package

Title Fast Iterative Filtering (FIF) with Portable FFT Backend

Version 1.0.1

Description Provides an R interface to a C implementation of Fast Iterative Filtering (FIF) for decomposing a univariate signal into intrinsic mode functions (IMFs) and a residual. The package uses Fast Fourier Transform library FFTW, if found. If not, it provides instructions to install it for your OS. This is recommended, as R's internal `fft()`, while avoiding external FFT dependencies, is two orders of magnitude slower. See vignette 'Installing FFTW for RFIF' for RFIF installation instructions.

Suggests testthat (>= 3.0.0), knitr, rmarkdown

Config/testthat/edition 3

License MIT + file LICENSE

Encoding UTF-8

VignetteBuilder knitr

URL <https://github.com/ChuckColeman/RFIF>

BugReports <https://github.com/ChuckColeman/RFIF/issues>

NeedsCompilation yes

Author Chuck Coleman [aut, cre] (ORCID:
<<https://orcid.org/0000-0001-6940-8117>>)

Maintainer Chuck Coleman <cdcoleman113@gmail.com>

Repository CRAN

Date/Publication 2026-04-10 08:20:25 UTC

Contents

rff	2
Index	3

`rfif`*Fast Iterative Filtering (C backend)*

Description

Decompose a univariate signal into Intrinsic Mode Functions (IMFs) using the vendored C implementation of Fast Iterative Filtering (FIF).

Usage`rfif(x)`**Arguments**

`x` Numeric vector (signal).

Value

A list with components:

- `imfs`: numeric matrix with one IMF per row (rows = IMFs; columns = time).
- `residual`: numeric vector of length `length(x)`.
- `nimf`: integer number of IMFs.

Examples

```
t <- seq(0, 1, length.out = 1024)
x <- sin(2*pi*5*t) + 0.5*sin(2*pi*20*t)
res <- rfif(x)
recon <- if (res$nimf > 0) colSums(res$imfs) + res$residual else res$residual
max(abs(x - recon))
```

Index

rfif, 2