Package 'easylift'

July 2, 2025

Title An R package to perform genomic liftover

Version 1.6.0

Date 2023-10-24

Description The easylift package provides a convenient tool for genomic liftover operations between different genome assemblies. It seamlessly works with Bioconductor's GRanges objects and chain files from the UCSC Genome Browser, allowing for straightforward handling of genomic ranges across various genome versions. One noteworthy feature of easylift is its integration with the BiocFileCache package. This integration automates the management and caching of chain files necessary for liftover operations. Users no longer need to manually specify chain file paths in their function calls, reducing the complexity of the liftover process.

License MIT + file LICENSE

URL https://github.com/nahid18/easylift, https://nahid18.github.io/easylift

BugReports https://github.com/nahid18/easylift/issues

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Depends GenomicRanges, BiocFileCache

Imports rtracklayer, GenomeInfoDb, R.utils, tools, methods

Suggests testthat (>= 3.0.0), IRanges, knitr, BiocStyle, rmarkdown

Config/testthat/edition 3

VignetteBuilder knitr

biocViews Software, WorkflowStep, Sequencing, Coverage, GenomeAssembly, DataImport

git_url https://git.bioconductor.org/packages/easylift

git_branch RELEASE_3_21

git_last_commit ea18a53

git_last_commit_date 2025-04-15

Repository Bioconductor 3.21

Date/Publication 2025-07-02

Author Abdullah Al Nahid [aut, cre] (ORCID:
<https://orcid.org/0000-0002-4390-0768>),
Hervé Pagès [aut, rev],
Michael Love [aut, rev] (ORCID:
<https://orcid.org/0000-0001-8401-0545>)

Maintainer Abdullah Al Nahid <abdnahid56@gmail.com>

Contents

easylift 2

Index **4**

`easylift` *Lift genomic coordinates from one genome assembly to another.*

Description

This function takes a GRanges object with genomic coordinates in one genome assembly and lifts them to target genome assembly using a chain file.

Usage

```
easylift(x, to, chain, bfc)
```

Arguments

`x`	A GRanges object with genomic coordinates in the original assembly.
`to`	The target genome assembly (e.g., "hg38").
`chain`	The path to the chain file containing the liftover mapping. Can be provided in gzipped or non-gzipped format. If omitted, the function will look in the default BiocFileCache for a properly named chain file.
`bfc`	A BiocFileCache object (optional), if not provided (most typically) the default location will be used.

Value

A GRanges object with lifted genomic coordinates.

See Also

`liftOver` function from the `rtracklayer` package, which is the basis for `easylift`.

Examples

```
# Lift over the coordinates of the first 10 genes in the hg19 assembly
# to the hg38 assembly
library(easylift)
gr <- GRanges(
  seqname = Rle(c("chr1", "chr2"), c(100000, 100000)),
  ranges = IRanges(start = 1, end = 200000)
)
# Here, "hg19" is the source genome
genome(gr) <- "hg19"

# Here, we use the `system.file()` function because the chain file is in the
# package (however if you need to point to any other file on your machine,
# just do 'chain <- "path/to/your/hg19ToHg38.over.chain.gz"'):
chain <- system.file("extdata", "hg19ToHg38.over.chain.gz", package = "easylift")

# Here, "hg38" is the target genome
easylift(gr, "hg38", chain)


# To use `BiocFileCache` for the chain file, add it to the cache as follows:
chain_file <- "/path/to/your/hg19ToHg38.over.chain.gz"
bfc <- BiocFileCache()

# Add chain file to cache if already not available
if (nrow(bfcquery(bfc, basename(chain_file))) == 0)
   bfcadd(bfc, chain_file)

# Then, use it in `easylift` like this:

easylift(gr, "hg38")
# or
gr |> easylift("hg38")
```

Index

easylift, 2

liftOver, *2*