Package: prider (via r-universe)

November 2, 2026

Title Multiplexed Primer Design by Linear Set Coverage Approximation

Version 1.0.6

Date 2026-04-21

Maintainer Manu Tamminen <mavatam@utu.fi>

Description Implementation of an oligonucleotide primer and probe design algorithm using a linearly scaling approximation of set coverage. A detailed description available at Smolander and Tamminen, 2021; <doi:10.1101/2021.09.06.459073>.

License BSD_3_clause + file LICENSE

URL https://github.com/tamminenlab/prider

Imports Rcpp (>= 1.0.5), dplyr, tidyr, purrr, stringr, magrittr, tibble, gplots

LinkingTo Rcpp

RoxygenNote 7.3.3

Encoding UTF-8

Config/pak/sysreqs libicu-dev

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

Date/Publication 2026-04-22 08:11:59 UTC

RemoteUrl https://github.com/tamminenlab/prider

RemoteRef HEAD

RemoteSha 642988b4951fbfdb1bef2ea980faec86ad543572

Contents

chunker . . . 2
new_prider . . . 2
new_primers . . . 3
new_sequences . . . 3
prepare_primer_df . . . 4
prider . . . 5
primers . . . 7
sequences . . . 8

Index **10**

`chunker` *chunker*

Description

Creates all primer candidates for a group of sequences using a sliding window.

Usage

```
chunker(seq_table, window_size = 20L)
```

Arguments

`seq_table`	A DataFrame containing a column for sequence ids (Id) and sequences (Seq).
`window_size`	An integer. Set the sliding window width.

Details

Sliding window to create chunks of DNA sequences

Value

A DataFrame containing columns for the sequence ids (Id), indexes (Ix), joined ids and indexes (Id_Ix), and the primer sequences (Seq).

Examples

```
test_csv <- system.file("extdata", "test.csv", package = "prider")

test_csv <- read.csv(test_csv)

chunks <- chunker(test_csv)
```

`new_prider` *new_prider*

Description

new_prider

Usage

```
new_prider(x = list())
```

Arguments

x	A list

Value

A prider object

`new_primers` *new_primers*

Description

Primers object constructor

Usage

```
new_primers(x)
```

Arguments

x	A tibble

Value

A primers object

`new_sequences` *new_sequences*

Description

Sequences object constructor

Usage

```
new_sequences(x)
```

Arguments

x	A tibble

Value

A sequences object

prepare_primer_df *Prepare a primer table for downstream analyses*

Description

Prepare a primer table for downstream analyses

Usage

```
prepare_primer_df(
  input_fasta,
  primer_length = 20,
  NTkeep = "basic",
  GCcheck = FALSE,
  GCmin = 0.4,
  GCmax = 0.6,
  GChalves = FALSE,
  GCsimilarity = 0.1
)
```

Arguments

input_fasta	A string. Name or filepath of the input FASTA file.
primer_length	A number. Sets the primer length. For applications involving two adjacent probes, the value should be set to two-fold the length of a single probe.
NTkeep	A string. Filters the primers based on the nucleotides. "basic" = keeps only the primers with G, C, T or A. "N" = keeps primers with G, C, T, A and N. "any" = keeps primers with the IUPAC nucleotide code characters. "all" = keeps all primers.
GCcheck	A logical. If TRUE, checks the GC contents of the primers and filters based on GCmin and GCmax.
GCmin	A decimal. If GCcheck is performed, this parameter determines the minimum proportional GC content.
GCmax	A decimal. If GCcheck is performed, this parameter determines the maximum proportional GC content.
GChalves	A logical. If TRUE, checks the GC contents separately for both halves of the primers and filters based on GCsimilarity. For example for applications involving two adjacent probes.
GCsimilarity	A number. If GChalves is performed, this parameter determines the maximum proportional GC content difference between the primer halves.

Value

A list containing sequence id conversions, primer matrix and a list of primers with their target sequences.

prider *Prider*

Description

Implementation of an oligonucleotide primer and probe design algorithm using a linearly scaling approximation of set coverage. A detailed description available at Smolander and Tamminen, 2021; doi:10.1101/2021.09.06.459073.

Prepare a nearly optimal primer coverage for an input FASTA file.

Usage

```
prider(
  fasta_file,
  primer_length = 20,
  minimum_primer_group_size = 10,
  minimum_seq_group_size = 2,
  cum_cov_decimals = 2,
  NTkeep = "basic",
  GCcheck = FALSE,
  GCmin = 0.4,
  GCmax = 0.6,
  GChalves = FALSE,
  GCsimilarity = 0.1
)

## S3 method for class 'prider'
print(x, ...)

## S3 method for class 'prider'
plot(x, ...)
```

Arguments

fasta_file	A string. Name or filepath of the input FASTA file.
primer_length	A number. Sets the primer length. For applications involving two adjacent probes, the value should be set to two-fold the length of a single probe.
minimum_primer_group_size	A number. Sets the minimum number of primers per primer cluster; smaller primer clusters will be discarded.
minimum_seq_group_size	A number. Sets the minimum number of sequences each primer cluster has to cover.
cum_cov_decimals	A number. Sets the number of decimals for cumulative coverage of primer clusters. Generally, lower value corresponds to less clusters and higher value to

	more clusters in the output. If the clusters do not cover the input sequences sufficiently, increasing this value may increase the coverage. If the clusters overlap too much, lowering the value may reduce this effect. Recommended range 1-4.
NTkeep	A string. Filters the primers based on the nucleotides. "basic" = keeps only the primers with G, C, T or A. "N" = keeps primers with G, C, T, A and N. "any" = keeps primers with the IUPAC nucleotide code characters. "all" = keeps all primers.
GCcheck	A logical. If TRUE, checks the GC contents of the primers and filters based on GCmin and GCmax.
GCmin	A decimal. If GCcheck is performed, this parameter determines the minimum proportional GC content.
GCmax	A decimal. If GCcheck is performed, this parameter determines the maximum proportional GC content.
GChalves	A logical. If TRUE, checks the GC contents separately for both halves of the primers and filters based on GCsimilarity. Used for example for applications involving two adjacent probes.
GCsimilarity	A decimal. If GChalves is performed, this parameter determines the maximum proportional GC content difference between the primer halves.
x	An object from prider function.
...	Other arguments.

Value

A list containing a sequence conversion table, primer candidates table, excluded sequences table and a primer coverage table.

Author(s)

Manu Tamminen <mavatam@utu.fi>, Niina Smolander <nijasm@utu.fi>

See Also

Useful links:

- https://github.com/tamminenlab/prider

Examples

```
test_fasta <- system.file('extdata', 'test.fasta', package = 'prider')

# Runs Prider with the default values:
primer_designs <- prider(test_fasta)

# Returns all the primers:
primers(primer_designs)
# Returns the primers of a specific primer group:
primers(primer_designs)[1]
```

```
# Returns all the sequences:
sequences(primer_designs)
# Returns the sequence of a specific Id:
sequences(primer_designs)[1]
# Plots the primers groups and the target sequences as a heatmap:
plot(primer_designs)
```

---

primers *primers*

Description

Definitions for the S3 methods for the primers classes

Usage

```
primers(prider_obj)

## Default S3 method:
primers(prider_obj)

## S3 method for class 'prider'
primers(prider_obj)

## S3 method for class 'primers'
print(x, ...)

## S3 method for class 'primers'
primer_obj[ix]
```

Arguments

prider_obj	An object from prider function.
x	An object from sequence function.
...	Other arguments.
primer_obj	An object from sequence function.
ix	A number. The number of the primer cluster.

Value

primer_obj

Examples

```
test_fasta <- system.file('extdata', 'test.fasta', package = 'prider')

primer_designs <- prider(test_fasta)

primers(primer_designs)

primers(primer_designs)[1]
```

---

sequences *sequences*

Description

Definitions for the S3 methods for the sequences classes

Usage

```
sequences(prider_obj)

## Default S3 method:
sequences(prider_obj)

## S3 method for class 'prider'
sequences(prider_obj)

## S3 method for class 'sequences'
print(x, ...)

## S3 method for class 'sequences'
sequence_obj[ix]
```

Arguments

prider_obj	An object from prider function.
x	An object from sequence function.
...	Other arguments.
sequence_obj	An object from sequence function.
ix	A number. The number of the primer cluster.

Value

sequence_obj

Examples

```
test_fasta <- system.file('extdata', 'test.fasta', package = 'prider')

primer_designs <- prider(test_fasta)

sequences(primer_designs)

sequences(primer_designs)[1]
```

Index

[.primers *(primers)*, 7
[.sequences *(sequences)*, 8

chunker, 2

new_prider, 2
new_primers, 3
new_sequences, 3

plot.prider *(prider)*, 5
prepare_primer_df, 4
prider, 5
prider-package *(prider)*, 5
primers, 7
print.prider *(prider)*, 5
print.primers *(primers)*, 7
print.sequences *(sequences)*, 8

sequences, 8