Package 'rprimer'

November 12, 2026

Type Package

Title Design Degenerate Oligos from a Multiple DNA Sequence Alignment

Version 1.16.0

Description Functions, workflow, and a Shiny application for visualizing sequence conservation and designing degenerate primers, probes, and (RT)-(q/d)PCR assays from a multiple DNA sequence alignment. The results can be presented in data frame format and visualized as dashboard-like plots. For more information, please see the package vignette.

License GPL-3

URL https://github.com/sofpn/rprimer

BugReports https://github.com/sofpn/rprimer/issues

Depends R (>= 4.1)

Imports Biostrings, bslib, DT, ggplot2, IRanges, mathjaxr, methods, patchwork, reshape2, S4Vectors, shiny, shinycssloaders, shinyFeedback

Suggests BiocStyle, covr, kableExtra, knitr, rmarkdown, styler, testthat (>= 3.0.0)

VignetteBuilder knitr

Encoding UTF-8

RoxygenNote 7.1.2

RdMacros mathjaxr

biocViews Alignment, ddPCR, Coverage, MultipleSequenceAlignment, SequenceMatching, qPCR

biocType Software

Config/testthat/edition 3

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

git_branch RELEASE_3_23

git_last_commit 34a13cb

git_last_commit_date 2026-04-28

Repository Bioconductor 3.23

Date/Publication 2026-11-11

Author Sofia Persson [aut, cre] (ORCID:
<https://orcid.org/0000-0003-2611-3030>)

Maintainer Sofia Persson <sofiapersson27@gmail.com>

Contents

rprimer-package 2
checkMatch 3
consensusProfile 6
designAssays 8
designOligos 11
example-datasets 15
plotData 16
Rprimer-classes 18
runRprimerApp 20

Index **21**

`rprimer-package` *rprimer: Design Degenerate Oligos from a Multiple DNA Sequence Alignment*

Description

Functions, workflow, and a Shiny application for visualizing sequence conservation and designing degenerate primers, probes, and (RT)-(q/d)PCR assays from a multiple DNA sequence alignment. The results can be presented in data frame format and visualized as dashboard-like plots. For more information, please see the package vignette.

Details

The package contains five functions: `consensusProfile()`, `designOligos()`, `designAssays()`, `checkMatch()` and `plotData()`

Author(s)

Maintainer: Sofia Persson <sofiapersson27@gmail.com> (ORCID)

See Also

Useful links:

- https://github.com/sofpn/rprimer
- Report bugs at https://github.com/sofpn/rprimer/issues

checkMatch *Check how oligos and assays match to their target sequences*

Description

checkMatch() investigates how well oligos or assays match with their intended target sequences within a multiple DNA sequence alignment.

Usage

```
checkMatch(x, target)

## S4 method for signature 'RprimerOligo'
checkMatch(x, target)

## S4 method for signature 'RprimerAssay'
checkMatch(x, target)
```

Arguments

x	An RprimerOligo or RprimerAssay object.
target	A Biostrings::DNAMultipleAlignment alignment with intended target sequences. Note that it must be same alignment that was used for generating the oligos/assays in x.

Details

The output provides information on the proportion and names of target sequences that match perfectly as well as with one, two, three, or four or more mismatches to the oligo within the intended oligo binding region in the input alignment (on-target match). It also gives the proportion and names of target sequences that match with a maximum of two mismatches to at least one sequence variant of the oligo outside the intended oligo binding region (off-target match). The function is a wrapper to Biostrings::vcountPDict() (Pages et al., 2020).

Value

An RprimerMatchOligo or RprimerMatchAssay object, depending on whether an RprimerOligo or RprimerAssay object was used as input.

RprimerMatchOligo objects contain the following information:

iupacSequence The oligo sequence in IUPAC format.

perfectMatch Proportion of target sequences that matches perfectly to the oligo within the intended binding region.

idPerfectMatch Names of all sequences that matches perfectly.

oneMismatch Proportion of target sequences with one mismatch to the oligo within the intended binding region.

idOneMismatch Names of all sequences that matches with one mismatch.

twoMismatches Proportion of target sequences with two mismatches to the oligo within the intended binding region.

idTwoMismatches Names of all sequences that matches with two mismatches.

threeMismatches Proportion of target sequences with three mismatches to the oligo within the intended binding region.

idThreeMismatches Names of all sequences that matches with three mismatches.

fourOrMoreMismatches Proportion of target sequences with four or more mismatches to the oligo within the intended binding region.

idFourOrMoreMismatches Names of all sequences that matches with four or more mismatches.

offTargetMatch Proportion of target sequences with maximum two mismatches to at least one site outside the intended oligo binding region in the input alignment.

idOffTargetMatch Names of all off-target matching sequences.

`RprimerMatchAssay` objects contain the following information:

iupacSequenceFwd The forward primer sequence in IUPAC format.

perfectMatchFwd Proportion of target sequences that matches perfectly with the forward primer withing the intended binding region.

idPerfectMatchFwd Names of all sequences that matches perfectly.

oneMismatchFwd Proportion of target sequences with one mismatch to the forward primer within the intended binding region.

idOneMismatchFwd Names of all sequences that matches with one mismatch.

twoMismatchesFwd Proportion of target sequences with two mismatches to the forward primer within the intended binding region.

idTwoMismatchesFwd Names of all sequences that matches with two mismatches.

threeMismatchesFwd Proportion of target sequences with three mismatches to the forward primer within the intended binding region.

idThreeMismatchesFwd Names of all sequences that matches with three mismatches.

fourOrMoreMismatchesFwd Proportion of target sequences with four or more mismatches to the forward primer within the intended binding region.

idFourOrMoreMismatchesFwd Names of all sequences that matches with four or more mismatches.

offTargetMatchFwd Proportion of target sequences with maximum two mismatches to at least one site outside the intended forward primer binding region in the input alignment.

idOffTargetMatchFwd Names of all off-target matching sequences.

iupacSequenceRev The reverse primer sequence in IUPAC format.

perfectMatchRev Proportion of target sequences that matches perfectly with the reverse primer withing the intended binding region.

idPerfectMatchRev Names of all sequences that matches perfectly.

oneMismatchRev Proportion of target sequences with one mismatch to the reverse primer within the intended binding region.

idOneMismatchRev Names of all sequences that matches with one mismatch.

twoMismatchesRev Proportion of target sequences with two mismatches to the reverse primer within the intended binding region.

idTwoMismatchesRev Names of all sequences that matches with two mismatches.

threeMismatchesRev Proportion of target sequences with three mismatches to the reverse primer within the intended binding region.

idThreeMismatchesRev Names of all sequences that matches with three mismatches.

fourOrMoreMismatchesRev Proportion of target sequences with four or more mismatches to the reverse primer within the intended binding region.

idFourOrMoreMismatchesRev Names of all sequences that matches with four or more mismatches.

offTargetMatchRev Proportion of target sequences with maximum two mismatches to at least one site outside the intended reverse primer binding region in the input alignment.

idOffTargetMatchRev Names of all off-target matching sequences.

If the input assay contains probes, the following information is also added:

iupacSequencePr The probe sequence in IUPAC format.

perfectMatchPr Proportion of target sequences that matches perfectly with the probe withing the intended binding region.

idPerfectMatchPr Names of all sequences that matches perfectly.

oneMismatchPr Proportion of target sequences with one mismatch to the probe within the intended binding region.

idOneMismatchPr Names of all sequences that matches with one mismatch.

twoMismatchesPr Proportion of target sequences with two mismatches to the probe within the intended binding region.

idTwoMismatchesPr Names of all sequences that matches with two mismatches.

threeMismatchesPr Proportion of target sequences with three mismatches to the probe within the intended binding region.

idThreeMismatchesPr Names of all sequences that matches with three mismatches.

fourOrMoreMismatchesPr Proportion of target sequences with four or more mismatches to the probe within the intended binding region.

idFourOrMoreMismatchesPr Names of all sequences that matches with four or more mismatches.

offTargetMatchPr Proportion of target sequences with maximum two mismatches to at least one site outside the intended probe binding region in the input alignment.

idOffTargetMatchPr Names of all off-target matching sequences.

Methods (by class)

- `RprimerOligo`:
- `RprimerAssay`:

Limitations

There are a few limitations with this function, which is important to be aware of:

- False negatives or positives may occur due to poorly aligned sequences
- The output does not tell which strand (minus or plus) the oligo matches to. This is important to consider when assessing off-target matches to single-stranded targets
- Ambiguous bases and gaps in the target sequences are identified as mismatches
- The function checks strictly on- and off-target, and may therefore miss off-target matches that partially overlap the intended target

References

Pages, H., Aboyoun, P., Gentleman R., and DebRoy S. (2020). Biostrings: Efficient manipulation of biological strings. R package version 2.57.2.

Examples

```
#### RprimerOligo objects

data("exampleRprimerOligo")
data("exampleRprimerAlignment")

x <- exampleRprimerOligo[1:2, ]
target <- exampleRprimerAlignment

checkMatch(x, target)

#### RprimerAssay objects

data("exampleRprimerAssay")
data("exampleRprimerAlignment")

x <- exampleRprimerAssay[1:2, ]
target <- exampleRprimerAlignment

checkMatch(x, target)
```

---

consensusProfile *Get sequence information from an alignment*

Description

`consensusProfile()` takes a DNA multiple alignment as input and returns all the data needed for subsequent primer and probe design process. The function is a wrapper to `Biostrings::consensusMatrix()` (Pages et al., 2020).

Usage

```
consensusProfile(x, ambiguityThreshold = 0)
```

Arguments

x	A Biostrings::DNAMultipleAlignment object.
ambiguityThreshold	"Detection level" for ambiguous bases. All DNA bases that occur with a relative frequency higher than the specified value will be included when the IUPAC consensus character is determined. Can range from 0 to 0.2, defaults to 0.

Value

An RprimerProfile object, which contains the following information:

position Position in the alignment.

a Proportion of A.

c Proportion of C.

g Proportion of G.

t Proportion of T.

other Proportion of bases other than A, C, G, T.

gaps Proportion of gaps (recognized as "-" in the alignment).

majority Majority consensus sequence. Denotes the most frequently occurring nucleotide. If two or more bases occur with the same frequency, the consensus nucleotide will be randomly selected among these.

identity Proportion of sequences, among all sequences with a DNA base (i.e., A, C, G or T), that has the majority consensus base.

iupac The consensus sequence expressed in IUPAC format. The IUPAC consensus sequence only takes 'A', 'C', 'G', 'T' and '-' as input. Degenerate bases will be skipped. If a position only contains degenerate bases, the IUPAC consensus will be NA at that position.

coverage Proportion of sequences in the target alignment, among all sequences with a DNA base, that are covered the IUPAC consensus character. The value will be 1 if there are no "remaining" DNA bases (and/or if ambiguityThreshold = 0).

References

Pages, H., Aboyoun, P., Gentleman R., and DebRoy S. (2020). Biostrings: Efficient manipulation of biological strings. R package version 2.57.2.

Examples

```
data("exampleRprimerAlignment")
consensusProfile(exampleRprimerAlignment)

consensusProfile(exampleRprimerAlignment, ambiguityThreshold = 0.05)
```

`designAssays` *Design (RT)-PCR assays*

Description

`designAssays()` combines primers to (RT)-PCR assays from an `RprimerOligo` object. If probes are present in the input dataset, only assays with a probe present between the primer pair will be kept.

Usage

```
designAssays(x, length = c(65, 120), tmDifferencePrimers = NULL)
```

Arguments

`x`	An `RprimerOligo` object, which can be with or without probes.
`length`	Amplicon length range, a numeric vector [40, 5000], defaults to `c(65, 120)`.
`tmDifferencePrimers`	Maximum allowed difference between the mean tm of the forward and reverse primer (in Celcius degrees, as an absolute value). Defaults to `NULL`, which means that primers will be paired regardless of their tm.

Value

An `RprimerAssay` object, containing the following information:

start Position where the assay starts.

end Position where the assay ends.

length Length of the amplicon.

totalDegeneracy Total number of oligos in the assay.

score Average oligo score. The best possible score is 0 and the worst possible score is 9. See `?oligos` for more information about the scoring system.

startFwd Start position of the forward primer.

endFwd End positon of the forward primer.

lengthFwd Length of the forward primer.

iupacSequenceFwd Forward primer sequence in IUPAC format (i.e. with ambiguous bases).

identityFwd For ambiguous primers: average identity of the forward primer. For mixed primers: average identity of the 5' (consensus) part of the forward primer. The value can range from 0 to 1.

coverageFwd For ambiguous primers: average coverage of the forward primer. For mixed primers: average coverage of the 3' (degenerate) part of the forward primer. The value can range from 0 to 1.

degeneracyFwd Number of sequence variants of the forward primer.

gcContentMeanFwd Mean GC-content of all sequence variants of the forward primer.

gcContentRangeFwd Range in GC-content of all sequence variants of the forward primer.

tmMeanFwd Mean tm of all sequence variants of the forward primer (in Celcius degrees).

tmRangeFwd Range in tm of all sequence variants of the forward primer (in Celcius degrees).

deltaGMeanFwd Mean delta G of all sequence variants of the forward primer (in kcal/mol).

deltaGRangeFwd Range in delta G of all sequence variants of the forward primer (in kcal/mol).

sequenceFwd All sequence variants of the forward primer.

gcContentFwd GC-content of all sequence variants of the forward primer.

tmFwd Tm of all sequence variants of the forward primer (in Celcius degrees).

deltaGFwd Delta G of all sequence variants of the forward primer (in kcal/mol).

methodFwd Design method used to generate the forward primer: "ambiguous" or "mixedFwd".

startRev Start position of the reverse primer.

endRev End positon of the reverse primer.

lengthRev Length of the reverse primer.

iupacSequenceRev Reverse primer sequence in IUPAC format (i.e. with ambiguous bases).

identityRev For ambiguous primers: average identity of the reverse primer. For mixed primers: average identity of the 5' (consensus) part of the reverse primer. The value can range from 0 to 1.

coverageRev For ambiguous primers: average coverage of the reverse primer. For mixed primers: average coverage of the 3' (degenerate) part of the reverse primer. The value can range from 0 to 1.

degeneracyRev Number of sequence variants of the reverse primer.

gcContentMeanRev Mean GC-content of all sequence variants of the reverse primer.

gcContentRangeRev Range in GC-content of all sequence variants of the reverse primer.

tmMeanRev Mean tm of all sequence variants of the reverse primer (in Celcius degrees).

tmRangeRev Range in tm of all sequence variants of the reverse primer (in Celcius degrees).

deltaGMeanRev Mean delta G of all sequence variants of the reverse primer (in kcal/mol).

deltaGRangeRev Range in delta G of all sequence variants of the reverse primer (in kcal/mol).

sequenceRev All sequence variants of the reverse primer.

gcContentRev GC-content of all sequence variants of the reverse primer.

tmRev Tm of all sequence variants of the reverse primer (in Celcius degrees).

deltaGRev Delta G of all sequence variants of the reverse primer (in kcal/mol).

methodRev Design method used to generate the forward primer: "ambiguous" or "mixedRev.

roiStart Start position of the input consensus profile used for oligo design.

roiEnd End position of the input consensus profile used for oligo design.

If a probe is included in the input `RprimerOligo` object, the following columns are also included:

startPr Start position of the probe.

endPr End positon of the probe.

lengthPr Length of the probe.

iupacSequencePr Probe sequence in plus sense, in IUPAC format.

iupacSequenceRcPr Probe sequence in minus sense, in IUPAC format.

identityPr For ambiguous primers: average identity of the probe. For mixed primers: average identity of the 5' (consensus) part of the probe. The value can range from 0 to 1.

coveragePr For ambiguous primers: average coverage of the probe. For mixed primers: average coverage of the 3' (degenerate) part of the probe. The value can range from 0 to 1.

degeneracyPr Number of sequence variants of the probe.

gcContentMeanPr Mean GC-content of all sequence variants of the probe.

gcContentRangePr Range in GC-content of all sequence variants of the probe.

tmMeanPr Mean tm of all sequence variants of the probe (in Celcius degrees).

tmRangePr Range in tm of all sequence variants of the forward primer (in Celcius degrees).

deltaGMeanPr Mean delta G of all sequence variants of the probe (in kcal/mol).

deltaGRangePr Range in delta G of all sequence variants of the probe (in kcal/mol).

sequencePr All sequence variants of the probe, in plus sense.

sequenceRcPr All sequence variants of the probe, in minus sense.

gcContentPr GC-content of all sequence variants of the probe.

tmPr Tm of all sequence variants of the probe (in Celcius degrees).

deltaGPr Delta G of all sequence variants of the probe (in kcal/mol).

methodPr Design method used to generate the probe.

plusPr If the probe is valid in plus sense.

minusPr If the probe is valid in minus sense.

An error message will return if no assays are found.

Examples

```
data("exampleRprimerOligo")

## Design assays using default settings
designAssays(exampleRprimerOligo)

## Modify the length range
designAssays(exampleRprimerOligo, length = c(1000, 2000))
```

`designOligos` *Design primers and probes*

Description

`designOligos()` designs oligos (primers and probes) from a consensus profile.

Usage

```
designOligos(
  x,
  maxGapFrequency = 0.01,
  lengthPrimer = c(18, 22),
  maxDegeneracyPrimer = 4,
  gcClampPrimer = TRUE,
  avoidThreeEndRunsPrimer = TRUE,
  gcPrimer = c(0.4, 0.65),
  tmPrimer = c(50, 65),
  concPrimer = 500,
  designStrategyPrimer = "ambiguous",
  probe = TRUE,
  lengthProbe = c(18, 22),
  maxDegeneracyProbe = 4,
  avoidFiveEndGProbe = TRUE,
  gcProbe = c(0.4, 0.65),
  tmProbe = c(50, 70),
  concProbe = 250,
  concNa = 0.05
)
```

Arguments

`x`	An `RprimerProfile` object.
`maxGapFrequency`	Maximum allowed gap frequency at the primer and probe binding sites in the target alignment. A number [0, 1], defaults to `0.01`.
`lengthPrimer`	Primer length range. A numeric vector [15, 30], defaults to `c(18, 22)`.
`maxDegeneracyPrimer`	Maximum number of variants of each primer. A number [1, 64], defaults to `4`.
`gcClampPrimer`	If primers must have a GC clamp. A GC clamp is identified as two to three G or C:s within the last five bases (3' end) of the primer. `TRUE` or `FALSE`, defaults to `TRUE`.
`avoidThreeEndRunsPrimer`	If primers with more than two runs of the same nucleotide at the terminal 3' end should be avoided. `TRUE` or `FALSE`, defaults to `TRUE`.

`gcPrimer` GC-content range for primers. A numeric vector [0, 1], defaults to `c(0.40, 0.65)`.

`tmPrimer` Tm range for primers (in Celcius degrees). A numeric vector [30, 90], defaults to `c(55, 65)`.

`concPrimer` Primer concentration in nM, for tm calculation. A number [20, 2000], defaults to `500`.

`designStrategyPrimer` `"ambiguous"` or `"mixed"`. Defaults to `"ambiguous"` (see details below).

`probe` If probes should be designed. `TRUE` or `FALSE`, defaults to `TRUE`.

`lengthProbe` Probe length range. A numeric vector [15, 40], defaults to `c(18, 22)`.

`maxDegeneracyProbe` Maximum number of variants of each probe. A number [1, 64], defaults to `4`.

`avoidFiveEndGProbe` If probes with G at the 5' end should be avoided. `TRUE` or `FALSE`, defaults to `TRUE`.

`gcProbe` GC-content range for probes. A numeric vector [0, 1], defaults to `c(0.40, 0.65)`.

`tmProbe` Tm range for probes (in Celcius degrees). A numeric vector [30, 90], defaults to `c(55, 70)`.

`concProbe` Primer concentration in nM, for tm calculation. A numeric vector [20, 2000], defaults to `250`.

`concNa` Sodium ion (equivalent) concentration in the PCR reaction (in M). For calculation of tm and delta G. A numeric vector [0.01, 1], defaults to `0.05` (50 mM).

Details

Valid oligos

For an oligo to be considered as valid, all sequence variants must fulfill all the specified design constraints.

Furthermore, oligos with at least one sequence variant containing more than four consecutive runs of the same nucleotide (e.g. "AAAAA") and/or more than three consecutive runs of the same dinucleotide (e.g. "TATATATA") will be excluded from consideration.

Calculation of tm and delta G

Melting temperatures are calculated for perfectly matching DNA duplexes using the nearest-neighbor method (SantaLucia and Hicks, 2004), by using the following equation:

$$Tm = (\Delta H^o \cdot 1000)/(\Delta S^o + R \cdot \log[\text{oligo}]) - 273.15$$

where ΔH^o is the change in enthalpy (in cal/mol) and ΔS^o is the change in entropy (in cal/K/mol) when an oligo and a perfectly matching target sequence goes from random coil to duplex formation. K is the gas constant (1.9872 cal/mol K).

Delta G is calculated at 37 Celcius degrees, for when an oligo and a perfectly matching target sequence goes from random coil to duplex state, by using the following equation:

$$\Delta G^o_T = (\Delta H^o \cdot 1000 - T \cdot \Delta S^o)/1000$$

ASCII representation For both tm and delta G, the following salt correction method is used for ΔS^o, as described in SantaLucia and Hicks (2004):

$$\Delta S^o[\mathrm{Na}^+] = \Delta S^o[1\mathrm{MNaCl}] + 0.368 \cdot N/2 \cdot \log[\mathrm{Na}^+]$$

where N is the total number of phosphates in the duplex, and [Na+] is the total concentration of monovalent cations.

Nearest neighbor table values for ΔS^o and ΔH^o are from SantaLucia and Hicks, 2004, and can be retrieved calling `rprimer:::lookup$nn`.

Primer design strategies

Primers can be generated by using one of the two following strategies:

- The **ambiguous strategy** (default) generates primers from the IUPAC consensus sequence alone, which means that ambiguous bases can occur at any position in the primer.
- The **mixed strategy** generates primers from both the majority and the IUPAC consensus sequence. These primers consist of a shorter degenerate part at the 3' end (approx. 1/3 of the primer, targeting a conserved region) and a longer consensus part at the 5' end (approx. 2/3 of the primer), which instead of having ambiguous bases contains the most frequently occuring nucleotide at each position. This strategy resembles the widely-adopted Consensus-Degenerate Hybrid Oligonucleotide Primer (CODEHOP) principle (Rose et al., 1998), and aims to to allow amplification of highly variable targets using primers with low degeneracy. The idea is that the degenerate 3' end part will bind specifically to the target sequence in the initial PCR cycles, and promote amplification in spite of eventual mismatches at the 5' consensus part (since 5' end mismatches are generally less detrimental than 3' end mismatches). In this way, the generated products will match the 5' ends of all primers perfectly, which allows them to be efficiently amplified in later PCR cycles. To provide a sufficiently high tm in spite of mismatches, it is recommended to design relatively long primers (at least 25 bases) when using this strategy.

Probes are always designed using the ambiguous strategy.

Scoring system for oligos

All valid oligos are scored based on their identity, coverage and average GC content. The scoring system is presented below.

Identity and coverage

Value range	Score
$(0.99, 1]$	0
$(0.95, 0.99]$	1
$(0.90, 0.95]$	2
≤ 0.90	3

Average GC-content

This score is based on how much the average GC-content deviates from 0.5 (in absolute value).

Value range	Score
$[0, 0.05)$	0
$[0.05, 0.1)$	1
$[0.1, 0.2)$	2
≥ 0.2	3

These scores are summarized. The weight of each individual score is 1, and thus, the lowest and best possible score for an oligo is 0, and the worst possible score is 9.

Value

An `RprimerOligo` object, containing the following information:

type Whether the oligo is a primer or probe.

fwd `TRUE` if the oligo is valid in forward direction, `FALSE` otherwise.

rev `TRUE` if the oligo is valid in reverse direction, `FALSE` otherwise.

start Start position of the oligo.

end End positon of the oligo.

length Oligo length.

iupacSequence Oligo sequence in IUPAC format (i.e. with ambiguous bases).

iupaSequenceRc The reverse complement of the iupacSequence.

identity For ambiguous oligos: average identity of the oligo. For mixed oligos: average identity of the 5' (consensus) part of the oligo. The value can range from 0 to 1.

coverage For ambiguous oligos: average coverage of the oligo. For mixed oligos: average coverage of the 3' (degenerate) part of the oligo. The value can range from 0 to 1.

degeneracy Number of sequence variants of the oligo.

gcContentMean Mean GC-content of all sequence variants of the oligo.

gcContent Range in GC-content of all sequence variants of the oligo.

tmMean Mean tm of all sequence variants of the oligo (in Celcius degrees).

tm Range in tm of all sequence variants of the oligo (in Celcius degrees).

deltaGMean Mean delta G of all sequence variants of the oligo (in kcal/mol).

deltaG Range in delta G of all sequence variants of the oligo (in kcal/mol).

sequence All sequence variants of the oligo.

sequenceRc Reverse complements of all sequence variants.

gcContent GC-content of all sequence variants.

tm Tm of all sequence variants (in Celcius degrees).

deltaG Delta G of all sequence variants (in kcal/mol).

method Design method used to generate the oligo: "ambiguous", "mixedFwd" or "mixedRev".

score Oligo score, the lower the better.

roiStart First position of the input `RprimerProfile` object (roi = region of interest).

roiEnd Last position of the input `RprimerProfile` object.

An error message will return if no oligos are found. If so, a good idea could be to re-run the design process with relaxed constraints.

References

Rose, TM., Schultz ER., Henikoff JG., Pietrokovski S., McCallum CM., and Henikoff S. 1998. Consensus-Degenerate Hybrid Oligonucleotide Primers for Amplification of Distantly Related Sequences. Nucleic Acids Research 26 (7): 1628-35.

SantaLucia Jr, J., & Hicks, D. (2004). The thermodynamics of DNA structural motifs. Annu. Rev. Biophys. Biomol. Struct., 33, 415-440.

Examples

```
data("exampleRprimerProfile")
x <- exampleRprimerProfile

## Design primers and probes with default values
designOligos(x)
```

`example-datasets` *Example datasets*

Description

The purpose of these datasets is to illustrate the functionality of rprimer. The following datasets are provided:

- `exampleRprimerAlignment` - a `Biostrings::DNAMultipleAlignment` object (Pages et al., 2020) containing an alignment of 50 hepatitis E virus sequences collected from NCBI GenBank. See "documentation_example_alignment.txt" within the inst/script folder of this package for more details.
- `exampleRprimerProfile` - an `RprimerProfile` object, generated from the alignment above.
- `exampleRprimerOligo` - an `RprimerOligo` object, generated from the consensus profile above.
- `exampleRprimerAssay` - an `RprimerAssay` object, generated from the oligos above.
- `exampleRprimerMatchOligo` - an `RprimerMatchOligo` object, describing how well some oligos match with the sequences in exampleRprimerAlignment.
- `exampleRprimerMatchAssay` - an `RprimerMatchAssay` object, describing how well some assays match with the seuqences in exampleRprimerAlignment.

Usage

```
data("exampleRprimerAlignment")

data("exampleRprimerProfile")

data("exampleRprimerOligo")

data("exampleRprimerAssay")

data("exampleRprimerMatchOligo")

data("exampleRprimerMatchAssay")
```

Format

An object of class DNAMultipleAlignment with 50 rows and 7597 columns.

An object of class RprimerProfile with 7597 rows and 12 columns.

An object of class RprimerOligo with 322 rows and 26 columns.

An object of class RprimerAssay with 4883 rows and 65 columns.

An object of class RprimerMatchOligo with 10 rows and 13 columns.

An object of class RprimerMatchAssay with 5 rows and 39 columns.

References

H. Pages, P. Aboyoun, R. Gentleman and S. DebRoy (2020). Biostrings: Efficient manipulation of biological strings. R package version 2.57.2.

---

plotData *Plot an Rprimer object*

Description

plotData visualizes objects from all different Rprimer classes.

Usage

```
plotData(x, ...)

## S4 method for signature 'RprimerProfile'
plotData(x, type = "overview", highlight = NULL, rc = FALSE)

## S4 method for signature 'RprimerOligo'
plotData(x)

## S4 method for signature 'RprimerAssay'
plotData(x)

## S4 method for signature 'RprimerMatchOligo'
plotData(x)

## S4 method for signature 'RprimerMatchAssay'
plotData(x)
```

Arguments

x	An RprimerProfile, RprimerOligo RprimerAssay, rprimerMatchOligo or RprimerMatchAssay object.
...	Optional arguments for RprimerProfile objects.

type	For Rprimeroligo objects: Type of plot: "overview", or "nucleotide", defaults to "overview".
highlight	For Rprimeroligo objects: If a specific region within an overview plot should be highlighted. A numeric vector indicating the start and end position, e.g. c(100, 1000), defaults to NULL (i.e., no highlight).
rc	For Rprimeroligo objects, and type = "nucleotide": If the plotted sequence should be displayed as reverse complement or not. TRUE or FALSE, defaults to FALSE. See examples below.

Value

A plot.

Methods (by class)

- RprimerProfile:
- RprimerOligo:
- RprimerAssay:
- RprimerMatchOligo:
- RprimerMatchAssay:

Examples

```
#### Plot an RprimerProfile object

data("exampleRprimerProfile")
prof <- exampleRprimerProfile

## Plot an overview
plotData(prof, highlight = c(500, 1000))

## Select a region of interest
roi <- prof[prof$position >= 500 & prof$position <= 550, ]

## Plot an overview of the roi
plotData(roi)

## Plot the nucleotide distribution of the roi
plotData(roi, type = "nucleotide")

#### Plot an RprimerOligo object

data("exampleRprimerOligo")
plotData(exampleRprimerOligo)

#### Plot an RprimerAssay object

data("exampleRprimerAssay")
```

```
plotData(exampleRprimerAssay)

#### Plot an RprimerMatchOligo object

data("exampleRprimerMatchOligo")
plotData(exampleRprimerMatchOligo)

#### Plot an RprimerMatchAssay object

data("exampleRprimerMatchAssay")
plotData(exampleRprimerMatchAssay)
```

`Rprimer-classes` *S4 classes for representation of different Rprimer objects*

Description

The rprimer package contains five different S4 classes. Each class is used as input or output for the different functions within the oligo and assay design workflow:

- `RprimerProfile`: output from `consensusProfile()`, input for `oligos()`.
- `RprimerOligo`: output from `oligos()`, input for `assays()` and `checkMatch()`.
- `RprimerAssay`: output from `assays()`, input for `checkMatch()`.
- `RprimerMatchOligo`: output from `checkMatch()`.
- `RprimerMatchAssay`: output from `checkMatch()`.

These classes extends the DFrame class from S4vectors (Pages et al., 2020), without any additional slots, but with some additional checks for validity.

Usage

```
RprimerProfile(...)

RprimerOligo(...)

RprimerAssay(...)

RprimerMatchOligo(...)

RprimerMatchAssay(...)
```

Arguments

`...`	A data frame or list to be converted into an Rprimer-object.

Value

An Rprimer-object if validation succeeds, an error message otherwise.

Coercion

Each class can be converted to a traditional data frame, by using either `as()` or `as.data.frame()`.

Moreover, `as()` can also be used for converting oligo sequences within an `RprimerOligo` or `RprimerAssay` object into a `Biostrings::DNAStringSet` object (Pages et al., 2020). Note that all oligo sequences will be written in the same direction as the input alignment that was used to generate the oligos.

References

Pages, H., Lawrence, M., and Aboyoun, R. (2020). S4Vectors: Foundation of vector-like and list-like containers in Bioconductor. R package version 0.28.0.

Pages, H., Aboyoun, P., Gentleman R., and DebRoy S. (2020). Biostrings: Efficient manipulation of biological strings. R package version 2.57.2.

See Also

consensusProfile, oligos, assays, checkMatch

Examples

```
## Constructors

data("exampleRprimerProfile")
x <- as.data.frame(exampleRprimerProfile)
RprimerProfile(x)

data("exampleRprimerOligo")
x <- as.data.frame(exampleRprimerOligo)
RprimerOligo(x)

data("exampleRprimerAssay")
x <- as.data.frame(exampleRprimerAssay)
RprimerAssay(x)

data("exampleRprimerMatchOligo")
x <- as.data.frame(exampleRprimerMatchOligo)
RprimerMatchOligo(x)

data("exampleRprimerMatchAssay")
x <- as.data.frame(exampleRprimerMatchAssay)
RprimerMatchAssay(x)

## Coercion methods for RprimerOligo and RprimerAssay objects

## Convert an RprimerOligo object to a DNAStringSet
data("exampleRprimerOligo")

## Pick rows to convert
x <- exampleRprimerOligo[1:2, ]
as(x, "DNAStringSet")

## Convert an RprimerAssay object to a DNAStringSet
```

```
data("exampleRprimerAssay")

## Pick rows to convert
x <- exampleRprimerAssay[1:2, ]
as(x, "DNAStringSet")
```

---

runRprimerApp *rprimer Shiny application*

Description

runRprimerApp() starts a Shiny application where the workflow of the rprimer package can be run through a graphical user interface.

Usage

```
runRprimerApp()
```

Value

Opens the Shiny application.

Examples

```
## Only run this in interactive R sessions:
if (interactive()) {
    runRprimerApp()
}
```

Index

∗ **datasets**
example-datasets, 15
∗ **internal**
rprimer-package, 2
.RprimerAssay *(*Rprimer-classes*)*, 18
.RprimerMatchAssay *(*Rprimer-classes*)*, 18
.RprimerMatchOligo *(*Rprimer-classes*)*, 18
.RprimerOligo *(*Rprimer-classes*)*, 18
.RprimerProfile *(*Rprimer-classes*)*, 18

checkMatch, 3
checkMatch,RprimerAssay-method *(*checkMatch*)*, 3
checkMatch,RprimerOligo-method *(*checkMatch*)*, 3
coerce *(*Rprimer-classes*)*, 18
consensusProfile, 6

designAssays, 8
designOligos, 11

example-datasets, 15
exampleRprimerAlignment *(*example-datasets*)*, 15
exampleRprimerAssay *(*example-datasets*)*, 15
exampleRprimerMatchAssay *(*example-datasets*)*, 15
exampleRprimerMatchOligo *(*example-datasets*)*, 15
exampleRprimerOligo *(*example-datasets*)*, 15
exampleRprimerProfile *(*example-datasets*)*, 15

plotData, 16
plotData,RprimerAssay-method *(*plotData*)*, 16
plotData,RprimerMatchAssay-method *(*plotData*)*, 16
plotData,RprimerMatchOligo-method *(*plotData*)*, 16
plotData,RprimerOligo-method *(*plotData*)*, 16
plotData,RprimerProfile-method *(*plotData*)*, 16

rprimer *(*rprimer-package*)*, 2
Rprimer-classes, 18
rprimer-package, 2
RprimerAssay *(*Rprimer-classes*)*, 18
RprimerAssay-class *(*Rprimer-classes*)*, 18
RprimerMatchAssay *(*Rprimer-classes*)*, 18
RprimerMatchAssay-class *(*Rprimer-classes*)*, 18
RprimerMatchOligo *(*Rprimer-classes*)*, 18
RprimerMatchOligo-class *(*Rprimer-classes*)*, 18
RprimerOligo *(*Rprimer-classes*)*, 18
RprimerOligo-class *(*Rprimer-classes*)*, 18
RprimerProfile *(*Rprimer-classes*)*, 18
RprimerProfile-class *(*Rprimer-classes*)*, 18
runRprimerApp, 20