Package ‘pqsfinder’

December 22, 2016

Type Package

Title Identification of potential quadruplex forming sequences

Version 1.2.3

Author Jiri Hon, Matej Lexa and Tomas Martinek

Maintainer Jiri Hon <jiri.hon@gmail.com>

Description The main functionality of this package is to detect DNA sequence patterns that are likely to fold into an intramolecular G-quadruplex (G4). Unlike many other approaches, this package is able to detect sequences responsible for G4s folded from imperfect G-runs containing bulges or mismatches and as such is more sensitive than competing algorithms.

License BSD_2_clause + file LICENSE

biocViews MotifDiscovery, SequenceMatching, GeneRegulation

LazyData TRUE

Depends Biostrings

Imports Rcpp (>= 0.12.3), GenomicRanges, IRanges, S4Vectors, methods

Suggests BiocStyle, knitr, Gviz, rtracklayer, biomaRt, BSgenome.Hsapiens.UCSC.hg38

LinkingTo Rcpp, BH (>= 1.62.0)

SystemRequirements GNU make

VignetteBuilder knitr

RoxygenNote 5.0.1

NeedsCompilation yes

R topics documented:

as.character,PQSVViews-method 2
density,PQSVViews-method 2
pqsfinder 3
PQSVViews 4
PQSVViews-class 5
score,PQSVViews-method 5
show,PQSVViews-method 6
strand,PQSVViews-method 6

Index 7
as.character,PQSViews-method
Coerce to character vector

Description
Coerce to character vector

Usage
## S4 method for signature 'PQSViews'
as.character(x)

Arguments
x
PQSViews object.

Value
Character vector representing PQS.

density,PQSViews-method
Get density vector

Description
Density vector represents numbers of PQS (potential quadruplex forming sequences) overlapping at each position in input sequence.

Usage
## S4 method for signature 'PQSViews'
density(x)

Arguments
x
PQSViews object

Value
Density vector

Examples
pqs <- pqsfinder(DNAString("CCCCCCGGGGTGGGTTGGGGAAAA"))
density(pqs)
pqsfinder

Identificate potential quadruplex forming sequences.

Description

Function for identification of all potential intramolecular quadruplex patterns (PQS) in DNA sequence.

Usage

```r
pqsfinder(subject, strand = "*", max_len = 50L, min_score = 42L,
run_min_len = 3L, run_max_len = 11L, loop_min_len = 1L,
loop_max_len = 30L, max_bulges = 3L, max_mismatches = 2L,
max_defects = 3L, tetrad_bonus = 45L, bulge_penalty = 20L,
mismatch_penalty = 31L, loop_mean_factor = 1, loop_sd_factor = 1,
run_re = "G\{1,10\}.\{0,9\}G\{1,10\}", custom_scoring_fn = NULL,
use_default_scoring = TRUE, verbose = FALSE)
```

Arguments

- **subject**  DNAString object.
- **strand**   Strand specification. Allowed values are "+", ".", or "*", where the last one represents both strands. Implicitly, the input DNAString object is assumed to encode the "+" strand.
- **max_len**  Maximal length of PQS.
- **min_score** Minimal PQS score.
- **run_min_len** Minimal length of quadruplex run.
- **run_max_len** Maximal length of quadruplex run.
- **loop_min_len** Minimal length of quadruplex loop.
- **loop_max_len** Maximal length of quadruplex loop.
- **max_bulges** Maximal number of runs with bulge.
- **max_mismatches** Maximal number of runs with mismatch.
- **max_defects** Maximum number of defects in total (max_bulges + max_mismatches).
- **tetrad_bonus** Score bonus for one complete G tetrad.
- **bulge_penalty** Penalization for a bulge in quadruplex run.
- **mismatch_penalty** Penalization for a mismatch in tetrad.
- **loop_mean_factor** Penalization factor of loop lengths mean.
- **loop_sd_factor** Penalization factor of loop lengths standard deviation.
- **run_re**  Regular expression specifying one run of quadruplex.
- **custom_scoring_fn** Custom quadruplex scoring function. It takes the following 10 arguments: subject - Input DNAString object, score - implicit PQS score, start - PQS start position, width - PQS width, loop_1 - start pos. of loop #1, run_2 - start pos. of run #2, loop_2 - start pos. of loop #2, run_3 - start pos. of run #3, loop_3 -
start pos. of loop #3, run_4 - start pos. of run #4. Return value of the function has to be new score represented as a single integer value. Please note that if use_default_scoring is enabled, the custom scoring function is evaluated AFTER the default scoring system but ONLY IF the default scoring system resulted in non-zero score (for performance reasons). On the other hand, when use_default_scoring is disabled, custom scoring function is evaluated on every PQS.

use_default_scoring

    Enables default internal scoring system. This option is particularly useful in case you intend to radically change the default behavior and specify your own scoring function. By disabling the default scoring you will get a full control above the underlying detection algorithm.

verbose

    Enables detailed output. Turn it on if you want to see all possible PQS found at each positions and not just the best one. It is highly recommended to use this option for debugging custom quadruplex scoring function. Each PQS is reported on separate row in the following format: start cnt pqs_sequence score, where start is the PQS starting position, pqs_sequence shows the PQS sequence structure with each run surrounded by square brackets and score is the score assigned to the particular PQS by all applied scoring functions.

Value

    PQSViews object

Examples

    pv <- pqsfinder(DNAString("CCCCCCGGGTGGGTGGGTGGGAAAA"))

Description

    User friendly constructor for PQSViews class representing potential quadruplex forming sequences (PQS). PQSViews is a subclass of XStringViews class and adds one more slot to store PQS density.

Usage

    PQSViews(subject, start, width, strand, score, density)

Arguments

    subject        DNAString object.
    start          Vector of PQS start positions.
    width          Vector of PQS lengths.
    strand         Vector of PQS strand specifications.
    score          Vector of PQS scores.
    density        Numbers of PQS overlapping at each position in subject.
PQSViews-class

Value

PQSViews object

Examples

pv <- PQSViews(DNAString("CGGGCGGGGC"), 1:2, 2:3, "+", 10:11, 1:10)
start(pv)
width(pv)
strand(pv)
score(pv)
density(pv)

PQSViews-class

An S4 class to represent potential quadruplex forming sequences.

Description

Represents potential quadruplex forming sequences found by pqsfinder function. This is a subclass of XStringViews-class class and adds one more slot.

Slots

density Numbers of PQS (potential quadruplex forming sequences) overlapping at each position in input sequence.

score,PQSViews-method

Get PQS score vector

Description

Get PQS score vector

Usage

## S4 method for signature 'PQSViews'
score(x)

Arguments

x PQSViews object

Value

Score vector

Examples

pqs <- pqsfinder(DNAString("CCCCCGGGTGGGTGGGTGGGAAAA"))
score(pqs)
### Show method

**Description**

Show method

**Usage**

```r
## S4 method for signature 'PQSViews'
show(object)
```

**Arguments**

- `object` : PQSViews object.

**Value**

PQSViews object printed.

---

### Get PQS strand vector

**Description**

Get PQS strand vector

**Usage**

```r
## S4 method for signature 'PQSViews'
strand(x)
```

**Arguments**

- `x` : PQSViews object

**Value**

Strand vector

**Examples**

```r
pqs <- psfinder(DNAString("CCCCCCGGGTGGGTGGGAAAA"))
strand(pqs)
```
Index

.PQSVIEWS (PQSVIEWS-class), 5
as.character, PQSVIEWS-method, 2
density, PQSVIEWS-method, 2
pqsfinder, 3, 5
PQSVIEWS, 4, 4
PQSVIEWS-class, 5
score, PQSVIEWS-method, 5
show, PQSVIEWS-method, 6
strand, PQSVIEWS-method, 6
XStringViews, 4