Package ‘pqsfinder’

April 26, 2017

Type Package
Title Identification of potential quadruplex forming sequences
Version 1.4.0
Date 2017-04-12
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, ggplot2, BSgenome.Hsapiens.UCSC.hg38
LinkingTo Rcpp, BH (>= 1.62.0)
SystemRequirements GNU make, C++11
VignetteBuilder knitr
RoxygenNote 6.0.1
NeedsCompilation yes

R topics documented:

as.character,PQSVViews-method .......................................................... 2
density,PQSVViews-method ............................................................... 2
maxScores ............................................................................................ 3
maxScores,PQSVViews-method .............................................................. 3
pqsfinder .............................................................................................. 4
PQSVViews ............................................................................................ 6
PQSVViews-class .................................................................................. 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("CCCCCCGGGGTTGGGTTGGGAAAA"))
density(pqs)
maxScores

Description
Get vector of maximal scores for a given object.

Usage
maxScores(x, ...)

Arguments
x
An object.
...
Additional arguments, for use in specific methods.

Value
Vector of maximal scores.

Examples
showMethods("maxScores")

maxScores,PQSViews-method
Get vector of maximal scores

Description
For each sequence position it gives the maximal score of all PQS conformations which overlap that position.

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

Arguments
x
PQSViews object.

Value
Vector of maximal scores.

Examples
pqs <- pqsfinder(DNAString("CCCCCGGGTGGGTGGGAAAA"))
maxScores(pqs)
pqsfnder

Identificate potential quadruplex forming sequences.

Description

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

Usage

pqsfnder(subject, strand = "*", overlapping = FALSE, max_len = 50L, min_score = 26L, run_min_len = 2L, run_max_len = 11L, loop_min_len = 0L, loop_max_len = 30L, max_bulges = 3L, max_mismatches = 3L, max_defects = 3L, tetrads_bonus = 40L, mismatch_penalty = 29L, edge_mismatch_penalty = 27L, bulge_penalty = 21L, bulge_len_factor = 0.7, bulge_len_exponent = 0.8, loop_mean_factor = 4.5, loop_mean_exponent = 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.
overlapping If true, than all overlapping PQS will be reported.
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. Unless the default scoring system is disabled, at most one loop can have zero length.
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).
tetrads_bonus Score bonus for one complete G tetrade.
mismatch_penalty Penalization for a mismatch in tetrad.
edge_mismatch_penalty Penalization for an mismatch in edge tetrade.
bulge_penalty Penalization for a bulge in quadruplex run.
bulge_len_factor Penalization factor for a bulge length.
bulge_len_exponent Exponent of bulge length.
loop_mean_factor
Penalization factor of loop length mean.

loop_mean_exponent
Exponent of loop length mean.

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.

Details
Use elementMetadata function to get extra PQS features like number of tetrads (nt), bulges (nb), mismatches (nm) or loop lengths (ll1, ll2, ll3).

Value
PQSViews object

Examples
pv <- pqsfinder(DNAString("CCCCCCGGGTGGGTGGGTAAAA"))
pv
elementMetadata(pv)
PQSViews  

**PQSViews class constructor**

**Description**

User friendly constructor for PQSViews class representing potential quadruplex forming sequences (PQS). PQSViews is a subclass of `XStringViews` class and adds two more slots to store PQS density and PQS score distribution.

**Usage**

```r
PQSViews(subject, start, width, strand, score, density, max_scores, nt, nb, nm, ll1, ll2, ll3)
```

**Arguments**

- `subject`  
  DNAString object.

- `start`  
  Start positions.

- `width`  
  Lengths.

- `strand`  
  Strand specifications.

- `score`  
  Scores.

- `density`  
  Numbers of PQS overlapping at each position in `subject`.

- `max_scores`  
  Score of the best PQS found at each position.

- `nt`  
  Tetrad numbers.

- `nb`  
  Bulge counts.

- `nm`  
  Mismatch counts.

- `ll1`  
  Loop 1 lengths.

- `ll2`  
  Loop 2 lengths.

- `ll3`  
  Loop 3 lengths.

**Details**

Use `elementMetadata` function to get extra PQS features like number of tetrads, bulges, mismatches or loop lengths.

**Value**

PQSViews object.

**Examples**

```r
pv <- PQSViews(DNAString("CGGGCGGGGC"), c(1,2), c(2,3), c("+", "+"),
                c(10,11), 1:10, 1:10, 0, 0, 1, 1, 1)
start(pv)
width(pv)
strand(pv)
score(pv)
density(pv)
maxScores(pv)
elementMetadata(pv)
```
Description

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

Slots

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

max_scores  Score of the best PQS found at each position.

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("CCCCCCGGGTGGGTGGGAAAA"))
score(pqs)
### show,PQSViews-method

**Show method**

**Description**
Show method

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

**Arguments**
- `object` PQSViews object.

**Value**
PQSViews object printed.

---

### strand,PQSViews-method

**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 <- pqsfinder(DNAString("CCCCCCGGGTGGGTGGGTGGGAAAA"))
strand(pqs)
```
Index

.PQSViews (PQSViews-class), 7
as.character, PQSViews-method, 2
density, PQSViews-method, 2
elementMetadata, 5, 6
maxScores, 3
maxScores, PQSViews-method, 3
pqsfinder, 4, 7
PQSViews, 5, 6
PQSViews-class, 7
score, PQSViews-method, 7
show, PQSViews-method, 8
strand, PQSViews-method, 8
XStringViews, 6