Package ‘MicroSEC’

August 25, 2024

Title Sequence Error Filter for Formalin-Fixed and Paraffin-Embedded Samples

Version 2.1.6

Maintainer Masachika Ikegami <`ikegamitky@gmail.com`>

Description Clinical sequencing of tumor is usually performed on formalin-fixed and paraffin-embedded samples and have many sequencing errors. We found that the majority of these errors are detected in chimeric read caused by single-strand DNA with micro-homology. Our filtering pipeline focuses on the uneven distribution of the artifacts in each read and removes such errors in formalin-fixed and paraffin-embedded samples without over-eliminating the true mutations detected in fresh frozen samples.

License MIT + file LICENSE

Encoding UTF-8

Language en-US

Depends R (>= 3.5.0)

RoxygenNote 7.3.1

LazyData true

Imports stringr, dplyr, Biostrings, Rsamtools, GenomeInfoDb, BiocGenerics

Suggests BSgenome.Hsapiens.UCSC.hg38, knitr, rmarkdown, BSgenome.Hsapiens.UCSC.hg19, BSgenome.Mmusculus.UCSC.mm10

VignetteBuilder knitr

URL `https://github.com/MANO-B/MicroSEC/`

BugReports `https://github.com/MANO-B/MicroSEC/issues`

NeedsCompilation no

Author Masachika Ikegami [aut, cre]

Repository CRAN

Date/Publication 2024-08-25 07:50:05 UTC

Contents

`exampleBam` *An example BAM file.*

Description

A BAM file containing the information of eight mutations.

Usage

```
exampleBam
```

Format

A list with 8 factors, each contains 46527 variables:

rname chromosome of the read

qname read ID list

seq sequence of the read, in DNAString

strand strand of the read

cigar CIGAR sequence of the read

qual Phred quality of the read

pos starting position of the read

isize insert size of the read ...

`exampleMutation` *An example mutation file.*

Description

A dataset containing the information of eight mutations.

Usage

```
exampleMutation
```

Format

A list with 8 factors, each contains 29 variables

Sample sample name

Mut_type mutation type

Chr altered chromosome

Pos altered position

Ref reference base

Alt altered base

SimpleRepeat_TRF mutation locating repeat sequence

Neighborhood_sequence neighborhood sequence ...

`fun_analysis` *Analyzing function.*

Description

This function analyzes the filtering results.

Usage

```
fun_analysis(
  msec,
  mut_depth,
  short_homology_search_length,
  min_homology_search,
  threshold_p,
  threshold_hairpin_ratio,
  threshold_short_length,
  threshold_distant_homology,
  threshold_soft_clip_ratio,
  threshold_low_quality_rate,
  homopolymer_length
)
```

Arguments

`msec`	Mutation filtering information.
`mut_depth`	Mutation coverage data.
`short_homology_search_length`	Small sequence for homology search.
`min_homology_search`	The sequence length for homology search.
`threshold_p`	The largest p value of significant errors.
`threshold_hairpin_ratio`	The smallest hairpin read ratio.
`threshold_short_length`	Reads shorter than that are analyzed.
`threshold_distant_homology`	The smallest rate of reads from other regions.
`threshold_soft_clip_ratio`	The rate of soft-clipped reads.
`threshold_low_quality_rate`	The smallest rate of low quality bases.
`homopolymer_length`	The smallest length of homopolymers.

Value

msec

Examples

```
data(msec_summarized)
data(mut_depth_checked)
fun_analysis(msec = msec_summarized,
             mut_depth = mut_depth_checked,
```

```
                short_homology_search_length = 4,
                min_homology_search = 40,
                threshold_p = 10 ^ (-6),
                threshold_hairpin_ratio = 0.50,
                threshold_short_length = 0.75,
                threshold_distant_homology = 0.15,
                threshold_soft_clip_ratio = 0.50,
                threshold_low_quality_rate = 0.1,
                homopolymer_length = 15
)
```

`fun_hairpin_check` *Hairpin-structure sequence check function*

Description

This function attempts to find hairpin structure sequences.

Usage

```
fun_hairpin_check(hairpin_seq_tmp, ref_seq, hairpin_length, hair)
```

Arguments

`hairpin_seq_tmp`	The sequence to be checked.
`ref_seq`	Reference sequence around the mutation.
`hairpin_length`	The temporal length of hairpin sequences.
`hair`	The length of sequences to be checked.

Value

list(hairpin_length, whether hairpin sequences exist or not)

`fun_homology` *Homology check function.*

Description

This function attempts to search the homologous regions.

Usage

```
fun_homology(
  msec,
  df_distant,
  min_homology_search,
  ref_genome,
  chr_no,
  progress_bar
)
```

Arguments

`msec`	Mutation filtering information.
`df_distant`	Sequences to be checked.
`min_homology_search`	Minimum length to define "homologous".
`ref_genome`	Reference genome for the data.
`chr_no`	Reference genome chromosome number (human=24, mouse=22).
`progress_bar`	"Y": You can see the progress visually.

Value

msec

Examples

```
## Not run:
data(msec_read_checked)
data(homology_searched)
fun_homology(msec = msec_read_checked,
   df_distant = homology_searched,
   min_homology_search = 40,
   ref_genome = BSgenome.Hsapiens.UCSC.hg38::BSgenome.Hsapiens.UCSC.hg38,
   chr_no = 24,
   progress_bar = "N"
)

## End(Not run)
```

`fun_load_bam` *BAM file loader*

Description

This function attempts to load the BAM file.

Usage

```
fun_load_bam(bam_file)
```

Arguments

`bam_file`	Path of the BAM file.

Value

df_bam

Examples

```
fun_load_bam(
  system.file("extdata", "sample.bam", package = "MicroSEC")
)
```

`fun_load_chr_no` *Chromosome number loading function.*

Description

This function attempts to load the chromosome number.

Usage

```
fun_load_chr_no(organism)
```

Arguments

`organism`	Human or Mouse genome.

Value

chr_no

Examples

```
fun_load_chr_no("Human")
```

`fun_load_genome` *Genome loading function.*

Description

This function attempts to load the appropriate genome.

Usage

```
fun_load_genome(organism)
```

Arguments

`organism`	Human or Mouse genome.

Value

ref_genome

Examples

```
fun_load_genome("Human")
```

`fun_load_mutation` *Mutation data file loader*

Description

This function attempts to load the mutation information file.

Usage

```
fun_load_mutation(
  mutation_file,
  sample_name,
  ref_genome,
  chr_no,
  simple_repeat_list = ""
)
```

Arguments

`mutation_file`	Path of the mutation information file.
`sample_name`	Sample name.
`ref_genome`	Reference genome for the data.
`chr_no`	Reference genome chromosome number (human=24, mouse=22).
`simple_repeat_list`	Optional, set simple repeat bed file path.

Value

df_mutation

Examples

```
fun_load_mutation(
  system.file("extdata", "mutation_list.tsv", package = "MicroSEC"),
  "sample",
  BSgenome.Hsapiens.UCSC.hg38::BSgenome.Hsapiens.UCSC.hg38,
  24
)
```

`fun_read_check` *Read check function.*

Description

This function attempts to check the mutation profile in each read.

Usage

```
fun_read_check(
  df_mutation,
  df_bam,
  ref_genome,
  sample_name,
  read_length,
  adapter_1,
  adapter_2,
  short_homology_search_length,
  min_homology_search,
  progress_bar
)
```

Arguments

`df_mutation`	Mutation information.
`df_bam`	Data from the BAM file.
`ref_genome`	Reference genome for the data.
`sample_name`	Sample name (character)
`read_length`	The read length in the sequence.
`adapter_1`	The Read 1 adapter sequence of the library.
`adapter_2`	The Read 2 adapter sequence of the library.
`short_homology_search_length`	Small sequence for homology search.

`min_homology_search`	Minimum length to define "homologous".
`progress_bar`	"Y": You can see the progress visually.

Value

list(msec, homology_search, mut_depth)

Examples

```
## Not run:
data(exampleMutation)
data(exampleBam)
fun_read_check(df_mutation = exampleMutation,
df_bam = exampleBam,
ref_genome = BSgenome.Hsapiens.UCSC.hg38::BSgenome.Hsapiens.UCSC.hg38,
sample_name = "sample",
read_length = 150,
adapter_1 = "AGATCGGAAGAGCACACGTCTGAACTCCAGTCA",
adapter_2 = "AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT",
short_homology_search_length = 4,
min_homology_search = 40,
progress_bar = "N"
)

## End(Not run)
```

`fun_repeat_check` *Repeat check function.*

Description

This function attempts to check the repetitive sequence around the mutation.

Usage

```
fun_repeat_check(rep_a, rep_b, ref_seq, ref_width, del)
```

Arguments

`rep_a`	The shorter sequence of Ref and Alt.
`rep_b`	The longer sequence of Ref and Alt.
`ref_seq`	Reference sequence around the mutation.
`ref_width`	Search length for ref_seq.
`del`	Insertion: 0, Deletion: 1

Value

list(pre_rep_status, post_rep_status, pre_rep_short, post_rep_short, homopolymer_status)

`fun_save` *Save function.*

Description

This function attempts to save the filtering results.

Usage

```
fun_save(msec, output)
```

Arguments

`msec`	Mutation filtering information.
`output`	output file name (full path).

Examples

```
## Not run:
data(msec_analyzed)
fun_save(msec = msec_analyzed,
         output = "./MicroSEC_test.tsv.gz"
)

## End(Not run)
```

`fun_setting` *Mutated position search function.*

Description

This function attempts to find the mutated bases in each read.

Usage

```
fun_setting(pre, post, neighbor_seq, neighbor_length, alt_length)
```

Arguments

`pre`	The 5' side bases of the sequence for searching.
`post`	The 3' side bases of the sequence for searching.
`neighbor_seq`	Short reference sequence around the mutation.
`neighbor_length`	The length from the mutation to the ends of the short reference sequence.
`alt_length`	The length of altered bases.

Value

list(pre_search_length, post_search_length, peri_seq_1, peri_seq_2)

`fun_summary` *Summarizing function.*

Description

This function summarizes the filtering results.

Usage

```
fun_summary(msec)
```

Arguments

`msec`	Mutation filtering information.

Value

msec

Examples

```
data(msec_homology_searched)
fun_summary(msec_homology_searched)
```

`fun_zero` *Divide function without 0/0 errors*

Description

This function attempts to divide without 0/0 errors.

Usage

```
fun_zero(a, b)
```

Arguments

`a, b`	Integers

Value

a divided by b

`homology_searched` *An example sequence information file.*

Description

A dataset containing the information of reads for homology search.

Usage

```
homology_searched
```

Format

A list with 7 factors, each contains 1508 variables:

sample_name sample name

Chr altered chromosome

Pos altered position

Ref reference base

Alt altered base

Direction 5' (pre) or 3' (post) sequence of the mutated base

Seq sequence for homology search ...

`msec_analyzed` *An example mutation file.*

Description

A dataset containing the information of eight mutations processed by the fun_homology function.

Usage

```
msec_analyzed
```

Format

A list with 37 factors, each contains 29 variables

Sample sample name

Mut_type mutation type

Chr altered chromosome

Pos altered position

Ref reference base

Alt altered base

SimpleRepeat_TRF mutation locating repeat sequence

Neighborhood_sequence neighborhood sequence

read_length read length

total_read number of mutation supporting reads

soft_clipped_read number of soft-clipped reads

flag_hairpin number of reads produced by hairpin structure

pre_support_length maximum 5’-supporting length

post_support_length maximum 3’-supporting length

short_support_length minimum supporting length

pre_farthest 5’-farthest supported base from the mutated base

post_farthest 3’-farthest supported base from the mutated base

low_quality_base_rate_under_q18 low quality base rate

low_quality_pre low quality base rate of 5’- side

low_quality_post low quality base rate of 3’- side

distant_homology_rate rate of reads derived from homologous regions

soft_clipped_rate rate of soft clipped reads

prob_filter_1 possibility of short-supporting length

prob_filter_3_pre possibility of 5’-supporting length

prob_filter_3_post possibility of 3’-supporting length

filter_1_mutation_intra_hairpin_loop filter 1

filter_2_hairpin_structure filter 2

filter_3_microhomology_induced_mutation filter 3

filter_4_highly_homologous_region filter 4

filter_5_soft_clipping filter 5

filter_6_simple_repeat filter 6

filter_7_mutation_at_homopolymer filter 7

filter_8_low_quality filter 8

msec_filter_123 any of filter 1-3

msec_filter_1234 any of filter 1-4

msec_filter_all any of filter 1-8

comment comment ...

`msec_homology_searched` *An example mutation file.*

Description

A dataset containing the information of eight mutations processed by the fun_homology function.

Usage

```
msec_homology_searched
```

Format

A list with 34 factors, each contains 29 variables

Sample	sample name
Mut_type	mutation type
Chr	altered chromosome
Pos	altered position
Ref	reference base
Alt	altered base
SimpleRepeat_TRF	mutation locating repeat sequence
Neighborhood_sequence	neighborhood sequence
read_length	read length
mut_type	mutation type
alt_length	length of the mutated bases
total_read	number of mutation supporting reads
soft_clipped_read	number of soft-clipped reads
flag_hairpin	number of reads produced by hairpin structure
hairpin_length	maximum length of palindromic sequences
pre_support_length	maximum 5'-supporting length
post_support_length	maximum 3'-supporting length
short_support_length	minimum supporting length
pre_minimum_length	minimum 5'-supporting length
post_minimum_length	minimum 3'-supporting length
pre_farthest	5'-farthest supported base from the mutated base
post_farthest	3'-farthest supported base from the mutated base
low_quality_base_rate_under_q18	low quality base rate
low_quality_pre	low quality base rate of 5'- side

low_quality_post low quality base rate of 3’- side

pre_rep_status 5’-repeat sequence length

post_rep_status 3’-repeat sequence length

homopolymer_status homopolymer sequence length

indel_status whether the mutation is indel or not

indel_length length of indel mutation

distant_homology number of reads derived from homologous regions

penalty_pre 5’-penalty score by the mapper

penalty_post 3’-penalty score by the mapper

caution comment ...

`msec_read_checked` *An example mutation file.*

Description

A dataset containing the information of eight mutations processed by the fun_read_check function.

Usage

```
msec_read_checked
```

Format

A list with 34 factors, each contains 46527 variables

Sample sample name

Mut_type mutation type

Chr altered chromosome

Pos altered position

Ref reference base

Alt altered base

SimpleRepeat_TRF mutation locating repeat sequence

Neighborhood_sequence neighborhood sequence

read_length read length

mut_type mutation type

alt_length length of the mutated bases

total_read number of mutation supporting reads

soft_clipped_read number of soft-clipped reads

flag_hairpin number of reads produced by hairpin structure

hairpin_length maximum length of palindromic sequences

pre_support_length maximum 5’-supporting length

post_support_length maximum 3’-supporting length

short_support_length minimum supporting length

pre_minimum_length minimum 5’-supporting length

post_minimum_length minimum 3’-supporting length

pre_minimum_length minimum 5’-supporting length

low_quality_base_rate_under_q18 low quality base rate

low_quality_pre low quality base rate of 5’- side

low_quality_post low quality base rate of 3’- side

pre_farthest 5’-farthest supported base from the mutated base

post_farthest 3’-farthest supported base from the mutated base

post_rep_status 3’-repeat sequence length

homopolymer_status homopolymer sequence length

indel_status whether the mutation is indel or not

indel_length length of indel mutation

distant_homology number of reads derived from homologous regions

penalty_pre 5’-penalty score by the mapper

penalty_post 3’-penalty score by the mapper

caution comment ...

`msec_summarized` *An example mutation file.*

Description

A dataset containing the information of eight mutations processed by the fun_homology function.

Usage

```
msec_summarized
```

Format

A list with 52 factors, each contains 29 variables

Sample sample name

Mut_type mutation type

Chr altered chromosome

Pos altered position

Ref reference base

Alt altered base

SimpleRepeat_TRF mutation locating repeat sequence

Neighborhood_sequence neighborhood sequence

read_length read length

mut_type mutation type

alt_length length of the mutated bases

total_read number of mutation supporting reads

soft_clipped_read number of soft-clipped reads

flag_hairpin number of reads produced by hairpin structure

hairpin_length maximum length of palindromic sequences

pre_support_length maximum 5’-supporting length

post_support_length maximum 3’-supporting length

short_support_length minimum supporting length

pre_minimum_length minimum 5’-supporting length

post_minimum_length minimum 3’-supporting length

pre_farthest 5’-farthest supported base from the mutated base

post_farthest 3’-farthest supported base from the mutated base

low_quality_base_rate_under_q18 low quality base rate

low_quality_pre low quality base rate of 5’- side

low_quality_post low quality base rate of 3’- side

pre_rep_status 5’-repeat sequence length

post_rep_status 3’-repeat sequence length

homopolymer_status homopolymer sequence length

indel_status whether the mutation is indel or not

indel_length length of indel mutation

distant_homology number of reads derived from homologous regions

penalty_pre 5’-penalty score by the mapper

penalty_post 3’-penalty score by the mapper

caution comment

distant_homology_rate rate of reads derived from homologous regions

pre_minimum_length_adj adjusted pre_minimum_length

post_minimum_length_adj adjusted pre_minimum_length

pre_support_length_adj adjusted pre_minimum_length

post_support_length_adj adjusted pre_minimum_length

shortest_support_length_adj the shortest short_support_length

minimum_length_1 theoretically minimum 5’-supporting length

minimum_length_2	theoretically minimum 3'-supporting length
minimum_length	theoretically minimum supporting length
short_support_length_adj	adjusted short_support_length
altered_length	substituted/inserted length
half_length	half of the read length
short_support_length_total	range of short_support_length
pre_support_length_total	range of pre_support_length
post_support_length_total	range of post_support_length
half_length_total	range of possible short_support_length
total_length_total	range of possible supporting length
soft_clipped_rate	rate of soft clipped reads ...

`mut_depth_checked` *An example sequence information file.*

Description

A dataset containing the information of reads for homology search.

Usage

```
mut_depth_checked
```

Format

Three lists with 201 factors, each contains 29 variables:

Index