

Package ‘sveval’

February 3, 2019

Title SV evaluation

Version 1.1.0

Description Evaluate SV in a call set against a truth set using overlap-based approaches and sequence comparison for insertions.

Depends R (\geq 3.4.4)

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 6.1.0

Imports VariantAnnotation,
GenomicRanges,
magrittr,
dplyr,
rlang,
DelayedArray,
Biostrings,
parallel,
testthat,
ggplot2

R topics documented:

sveval-package	2
filterSVs	2
plot_pcurve	3
readSVvcf	4
svevalOl	5

Index	7
--------------	----------

sveval-package

SV evaluation

Description

Evaluate SV in a call set against a truth set using overlap-based approaches and sequence comparison for insertions.

Details

Package: sveval
Type: Package
Version: 1.1.0
Date: 2019-02-03
License: MIT

Author(s)

Jean Monlong <jmonlong@ucsc.edu>

See Also

<http://www.github.com/jmonlong/sveval>

Examples

```
## Not run:  
eval = sveval01('calls.vcf', 'truth.vcf')  
plot_prcurve(eval$curve)  
  
# Comparing multiple methods  
eval.1 = sveval01('calls1.vcf', 'truth.vcf')  
eval.2 = sveval01('calls2.vcf', 'truth.vcf')  
plot_prcurve(list(eval.1$curve, eval.2$curve), labels=c('method1', 'method2'))  
  
## End(Not run)
```

filterSVs

Filter SVs for size and regions of interest

Description

Filter SVs for size and regions of interest

Usage

```
filterSVs(sv.gr, regions.gr = NULL, ol.prop = 0.5, min.size = 0,
          max.size = Inf)
```

Arguments

sv.gr	the input SVs (e.g. read from readSVvcf)
regions.gr	the regions of interest. Ignored if NULL (default).
ol.prop	minimum proportion of sv.gr that must overlap regions.gr. Default is 0.5
min.size	the minimum SV size to be considered. Default 0.
max.size	the maximum SV size to be considered. Default is Inf.

Value

a subset of sv.gr that overlaps regions.gr or in the specified size range.

Author(s)

Jean Monlong

plot_prcurve	<i>Create precision-recall graphs</i>
--------------	---------------------------------------

Description

Create a precision/recall curve using metrics computed by the sveval01 function. The sveval01 function returns a list containing a "curve" data.frame with the evaluation metrics for different quality thresholds.

Usage

```
plot_prcurve(eval, labels = NULL)
```

Arguments

eval	a data.frame, a list of data.frames, or a vector with one or several paths to files with "curve" information.
labels	the labels to use for each input (when multiple inputs are used). Ignored is NULL (default).

Details

If the input is a data.frame (or list of data.frames) it should be the "curve" element of the list returned by the sveval01 function. If the input is a character (or a vector of characters), they are considered to be file names and the data will be read from these files.

If multiple inputs are given, either using a list of data.frames or a vectors with several filenames, one curve per input will be created. This is to be used to quickly compare several methods. The "labels" parameters can be used to specify a label for each input to use for the graphs.

Value

list of ggplot graph objects

Author(s)

Jean Monlong

Examples

```
## Not run:
eval = sveval01('calls.vcf', 'truth.vcf')
plot_prcurve(eval$curve)

# Comparing multiple methods
eval.1 = sveval01('calls1.vcf', 'truth.vcf')
eval.2 = sveval01('calls2.vcf', 'truth.vcf')
plot_prcurve(list(eval.1$curve, eval.2$curve), labels=c('method1', 'method2'))

# Or if the results were previously written in files
plot_prcurve(c('methods1-prcurve.tsv', 'methods2-prcurve.tsv'), labels=c('method1', 'method2'))

## End(Not run)
```

readSVvcf

Read SVs from a VCF file

Description

Read SVs from a VCF file

Usage

```
readSVvcf(vcf.file, keep.ins.seq = FALSE, sample.name = NULL)
```

Arguments

vcf.file the path to the VCF file

keep.ins.seq should it keep the inserted sequence? Default is FALSE.

sample.name the name of the sample to use. If NULL (default), use first sample.

Value

a GRanges object with relevant information.

Author(s)

Jean Monlong

Examples

```
## Not run:
calls.gr = readSVvcf('calls.vcf')

## End(Not run)
```

sveval0l

SV evaluation based on overlap and variant size

Description

SV evaluation based on overlap and variant size

Usage

```
sveval0l(calls.gr, truth.gr, max.ins.dist = 20, min.cov = 0.5,
  min.del.rol = 0.1, ins.seq.comp = FALSE, nb.cores = 1,
  min.size = 0, max.size = Inf, bed.regions = NULL,
  bed.regions.ol = 0.5, sample.name = NULL, outfile = NULL,
  out.bed.prefix = NULL, qual.quantiles = seq(0, 1, 0.1))
```

Arguments

<code>calls.gr</code>	call set. A GRanges or the path to a VCF file.
<code>truth.gr</code>	truth set. A GRanges or the path to a VCF file.
<code>max.ins.dist</code>	maximum distance for insertions to be clustered. Default is 20.
<code>min.cov</code>	the minimum coverage to be considered a match. Default is 0.5
<code>min.del.rol</code>	minimum reciprocal overlap for deletions. Default is 0.1
<code>ins.seq.comp</code>	compare sequence instead of insertion sizes. Default is FALSE.
<code>nb.cores</code>	number of processors to use. Default is 1.
<code>min.size</code>	the minimum SV size to be considered. Default 0.
<code>max.size</code>	the maximum SV size to be considered. Default is Inf.
<code>bed.regions</code>	If non-NULL, a GRanges object or path to a BED file (no headers) with regions of interest.
<code>bed.regions.ol</code>	minimum proportion of sv.gr that must overlap regions.gr. Default is 0.5
<code>sample.name</code>	the name of the sample to use if VCF files given as input. If NULL (default), use first sample.
<code>outfile</code>	the TSV file to output the results. If NULL (default), returns a data.frame.
<code>out.bed.prefix</code>	prefix for the output BED files. If NULL (default), no BED output.
<code>qual.quantiles</code>	the QUAL quantiles for the PR curve. Default is (0, .1, ..., .9, 1).

Value

a list with

<code>eval</code>	a data.frame with TP, FP and FN for each SV type when including all variants
<code>curve</code>	a data.frame with TP, FP and FN for each SV type when using different quality thresholds

Author(s)

Jean Monlong

Examples

```
## Not run:
## From VCF files
eval = sveval01('calls.vcf', 'truth.vcf')

## From GRanges
calls.gr = readSVvcf('calls.vcf')
truth.gr = readSVvcf('truth.vcf')
eval = sveval01(calls.gr, truth.gr)

## End(Not run)
```

Index

`filterSVs`, [2](#)

`plot_prcurve`, [3](#)

`readSVvcf`, [4](#)

`sveval-package`, [2](#)

`sveval01`, [5](#)