Package ‘sssc’

October 14, 2022

Title  Same Species Sample Contamination Detection

Version  1.0.0

Description  Imports Variant Calling Format file into R. It can detect whether a sample contains contaminant from the same species. In the first stage of the approach, a change-point detection method is used to identify copy number variations for filtering. Next, features are extracted from the data for a support vector machine model. For log-likelihood calculation, the deviation parameter is estimated by maximum likelihood method. Using a radial basis function kernel support vector machine, the contamination of a sample can be detected.

Depends  R (>= 3.4.0)

Imports  changepoint, e1071, ggplot2, stats, VGAM

License  GPL-2

Encoding  UTF-8

LazyData  true

RoxygenNote  6.0.1

NeedsCompilation  no

Author  Tao Jiang [aut, cre]

Maintainer  Tao Jiang <tjiang8@ncsu.edu>

Repository  CRAN

Date/Publication  2018-06-15 11:22:54 UTC

R topics documented:

  config_df ......................................................... 2
  generate_feature ................................................. 3
  getAlt2 .......................................................... 4
  getAnnoRate ...................................................... 4
  getAvgLL .......................................................... 5
  getLowDepth ...................................................... 5
**Description**

A dataframe containing default parameters.

**Usage**

`config_df`

**Format**

A data frame with 12 variables:

- `threshold` Threshold for allele frequency
- `skew` Skewness for allele frequency
- `lower` Lower bound for allele frequency region
- `upper` Upper bound for allele frequency region
- `ldpthred` Threshold to determine low depth
- `hom_mle` Hom MLE of p in Beta-Binomial model
- `het_mle` Het MLE of p in Beta-Binomial model
**generate_feature**  

*Hom_thred*  Threshold between hom and high  
*High_thred*  Threshold between high and het  
*Het_thred*  Threshold between het and low  
*hom_rho*  Hom MLE of rho in Beta-Binomial model  
*het_rho*  Het MLE of rho in Beta-Binomial model  

**Source**  

Created by Tao Jiang  

---  

**generate_feature**  

*Feature Generation for Contamination Detection Model*  

**Description**  

Generates features from each pair of input VCF objects for training contamination detection model.  

**Usage**  

\[
generate\_feature(file, \text{hom}_p = 0.999, \text{het}_p = 0.5, \text{hom}_rho = 0.005, \\
\text{het}_rho = 0.1, \text{mixture}, \text{homcut} = 0.99, \text{highcut} = 0.7, \text{hetcut} = 0.3)\]  

**Arguments**  

- **file**  VCF input object  
- **hom_p**  The initial value for p in Homozygous Beta-Binomial model, default is 0.999  
- **het_p**  The initial value for p in Heterozygous Beta-Binomial model, default is 0.5  
- **hom_rho**  The initial value for rho in Homozygous Beta-Binomial model, default is 0.005  
- **het_rho**  The initial value for rho in Heterozygous Beta-Binomial model, default is 0.1  
- **mixture**  A vector of whether the sample is contaminated: 0 for pure; 1 for contaminated  
- **homcut**  Cutoff allele frequency value between hom and high, default is 0.99  
- **highcut**  Cutoff allele frequency value between high and het, default is 0.7  
- **hetcut**  Cutoff allele frequency value between het and low, default is 0.3  

**Value**  

A data frame with all features for training model of contamination detection
**getAlt2**  
*Second alternative allele percentage*

**Description**
Second alternative allele percentage

**Usage**
getAlt2(f)

**Arguments**
- **f**  
  Input raw file

**Value**
Percent of the second alternative allele

---

**getAnnoRate**  
*Annotation rate*

**Description**
Annotation rate

**Usage**
getAnnoRate(f)

**Arguments**
- **f**  
  Input raw file

**Value**
Percentage of annotation locus
getAvgLL

**Description**

Calculate average log-likelihood

**Usage**

getAvgLL(df, hom_mle, het_mle, hom_rho, het_rho)

**Arguments**

- `df`: Input modified file
- `hom_mle`: Hom MLE of p in Beta-Binomial model, default is 0.9981416 from NA12878_1_L5
- `het_mle`: Het MLE of p in Beta-Binomial model, default is 0.4737897 from NA12878_1_L5
- `hom_rho`: Hom MLE of rho in Beta-Binomial model, default is 0.04570275 from NA12878_1_L5
- `het_rho`: Het MLE of rho in Beta-Binomial model, default is 0.02224098 from NA12878_1_L5

**Value**

meanLL

getLowDepth

**Description**

Low depth percentage

**Usage**

getLowDepth(f, ldpthred)

**Arguments**

- `f`: Input raw file
- `ldpthred`: Threshold to determine low depth, default is 20

**Value**

Percentage of low depth
getRatio

*Get the ratio of allele frequencies with a region*

**Description**

Get the ratio of allele frequencies with a region

**Usage**

```r
getRatio(subdf, lower, upper)
```

**Arguments**

- `subdf`: Dataframe with calculated statistics
- `lower`: Lower bound for allele frequency region
- `upper`: Upper bound for allele frequency region

**Value**

Ratio of allele frequencies with a region

getSkewness

*Get absolute value of skewness*

**Description**

Get absolute value of skewness

**Usage**

```r
geskewness(subdf)
```

**Arguments**

- `subdf`: Input dataframe

**Value**

Absolute value of skewness
**getSNVRate**

<table>
<thead>
<tr>
<th>getSNVRate</th>
<th>SNV percentage</th>
</tr>
</thead>
</table>

**Description**

SNV percentage

**Usage**

getSNVRate(df)

**Arguments**

- **df**: Input raw file

**Value**

Percentage of SNV

---

**getVar**

<table>
<thead>
<tr>
<th>getVar</th>
<th>Calculate zygosity variable</th>
</tr>
</thead>
</table>

**Description**

Calculate zygosity variable

**Usage**

getVar(df, state, hom_mle, het_mle)

**Arguments**

- **df**: Input modified file
- **state**: Zygosity state
- **hom_mle**: MLE in hom model
- **het_mle**: MLE in het model

**Value**

Zygosity variable
locateFile

*Check input filename*

**Description**

Check input filename

**Usage**

`locateFile(fn, extension)`

**Arguments**

- `fn` - Exact full file name of input file, including directory
- `extension` - Expected input file extension: vcf & txt

**Value**

Valid directory

---

negll

*Negative Log Likelihood*

**Description**

Calculates negative log likelihood for beta binomial distribution.

**Usage**

`negll(x, size, prob, rho)`

**Arguments**

- `x` - Depth of alternative allele
- `size` - Total depth
- `prob` - Theoretical probability for heterozygous is 0.5, for homozygous is 0.999
- `rho` - Rho parameter of Beta-Binomial distribution of alternative allele
**readGATK**

Read in input vcf data in GATK format for Contamination detection

**Description**

Read in input vcf data in GATK format for Contamination detection

**Usage**

`readGATK(dr, dbOnly, depCut, thred, content, extnum, keepall)`

**Arguments**

- **dr**: A valid input object
- **dbOnly**: Use dbSNP as filter, default is FALSE, passed from read_vcf
- **depCut**: Use a threshold for min depth, default is False
- **thred**: Threshold for min depth, default is 20
- **content**: Column names in VCF files
- **extnum**: The column number or numbers to be extracted from vcf, default is 10; 0 for not extracting any columns
- **keepall**: Keep unextracted column in output, default is TRUE, passed from read_vcf

**Value**

Dataframe from VCF file

---

**readStrelka**

Read in input vcf data in strelka2 format for Contamination detection

**Description**

Read in input vcf data in strelka2 format for Contamination detection

**Usage**

`readStrelka(dr, dbOnly, depCut, thred, content, extnum, keepall)`

**Arguments**

- **dr**: A valid input object
- **dbOnly**: Use dbSNP as filter, default is FALSE, passed from read_vcf
- **depCut**: Use a threshold for min depth, default is False
- **thred**: Threshold for min depth, default is 20
- **content**: Column names in VCF files
- **extnum**: The column number or numbers to be extracted from vcf, default is 10; 0 for not extracting any columns
- **keepall**: Keep unextracted column in output, default is TRUE, passed from read_vcf

**Value**

Dataframe from VCF file
Arguments

`dr` A valid input object
`dbOnly` Use dbSNP as filter, default is FALSE, passed from read_vcf
`depCut` Use a threshold for min depth, default is False
`thred` Threshold for min depth, default is 20
`content` Column names in VCF files
`extnum` The column number or numbers to be extracted from vcf, default is 10; 0 for not extracting any columns
`keepall` Keep unextracted column in output, default is TRUE, passed from read_vcf

Value

Dataframe from VCF file

### Description

Read in input vcf data in VarDict format for Contamination detection

### Usage

`readVarDict(dr, dbOnly, depCut, thred, content, extnum, keepall)`

Arguments

`dr` A valid input object
`dbOnly` Use dbSNP as filter, default is FALSE, passed from read_vcf
`depCut` Use a threshold for min depth, default is False
`thred` Threshold for min depth, default is 20
`content` Column names in VCF files
`extnum` The column number to be extracted from vcf, default is 10; 0 for not extracting any column
`keepall` Keep unextracted column in output, default is TRUE, passed from read_vcf

Value

Dataframe from VCF file
readVarPROWL

Read in input vcf data in VarPROWL format

Description
Read in input vcf data in VarPROWL format

Usage
readVarPROWL(dr, dbOnly, depCut, thred, content, extnum, keepall)

Arguments

\begin{itemize}
  \item \texttt{dr} \hspace{1cm} A valid input object
  \item \texttt{dbOnly} \hspace{1cm} Use dbSNP as filter, default is FALSE, passed from \texttt{read_vcf}
  \item \texttt{depCut} \hspace{1cm} Use a threshold for min depth, default is False
  \item \texttt{thred} \hspace{1cm} Threshold for min depth, default is 20
  \item \texttt{content} \hspace{1cm} Column names in VCF files
  \item \texttt{extnum} \hspace{1cm} The column number or numbers to be extracted from vcf, default is 10; 0 for not extracting any columns
  \item \texttt{keepall} \hspace{1cm} Keep unextracted column in output, default is TRUE, passed from \texttt{read_vcf}
\end{itemize}

Value
vcf Dataframe from VCF file

read_vcf

VCF Data Input

Description
Reads a file in vcf or vcf.gz file and creates a list containing Content, Meta, VCF and file_sample_name

Usage
read_vcf(fn, vcffor, dbOnly = FALSE, depCut = FALSE, thred = 20, metaline = 200, extnum = 10, keepall = T)
Arguments

fn
Input vcf file name

vcffor
Input vcf data format: 1) GATK; 2) VarPROWL; 3) VarDict; 4) strelka2

dbOnly
Use dbSNP as filter, default is FALSE

depCut
Use a threshold for min depth, default is False

thred
Threshold for min depth, default is 20

metaline
Number of head lines to read in (better to be large enough), the lines will be checked if they contain meta information, default is 200

extnum
The column number to be extracted from vcf, default is 10; 0 for not extracting any column; extnum should be between 10 and total column number

keepall
Keep unextracted column in output, default is TRUE

Value

A list containing (1) Content: a vector showing what is contained; (2) Meta: a data frame containing meta-information of the file; (3) VCF: a data frame, the main part of VCF file; (4) file_sample_name: the file name and sample name, in case when multiple samples exist in one file, file and sample names might be different

Examples

file.name <- system.file("extdata", "example.vcf.gz", package = "sssc")
example <- read_vcf(fn=file.name, vcffor="VarPROWL")

rho_est
Estimate Rho for Alternative Allele Frequency

Description

Estimates Rho parameter in beta binomial distribution for alternative allele frequency

Usage

rho_est(vl)

Arguments

vl
A list of vcf objects from read_vcf function.

Value

A list containing (1) het_rho: Rho parameter of heterozygous location; (2) hom_rho: Rho parameter homozygous location;
Examples

data("vcf_example")
vcf_list <- list()
vcf_list[[1]] <- vcf_example$VCF
res <- rho_est(vl = vcf_list)
res$hetero[[1]]$par
res$hom_rhoo[[1]]$par

---

rmChangePoint

Remove CNV regions within VCF files by changepoint method

Description

Remove CNV regions within VCF files by changepoint method

Usage

rmChangePoint(vcf, threshold, skew, lower, upper)

Arguments

vcf Input VCF files
threshold Threshold for allele frequency
skew Skewness for allele frequency
lower Lower bound for allele frequency region
upper Upper bound for allele frequency region

Value

VCF object without changepoint region

---

rmCNVinVCF

Remove CNV regions within VCF files given cnv file

Description

Remove CNV regions within VCF files given cnv file

Usage

rmCNVinVCF(vcf, cnvobj)

Arguments

vcf Input VCF files
cnvobj cnv object
**sssc**

*Same Species Sample Contamination*

**Value**

VCF object without changepoint region

**Description**

Detects whether a sample is contaminated another sample of its same species. The input file should be in vcf format.

**Usage**

```r
sssc(file, rmCNV = FALSE, cnvobj = NULL, config = NULL,
class_model = NULL, regression_model = NULL)
```

**Arguments**

- `file` VCF input object
- `rmCNV` Remove CNV regions, default is FALSE
- `cnvobj` cnv object, default is NULL
- `config` config information of parameters. A default set is generated as part of the model and is included in a model object, which contains
- `class_model` An SVM classification model
- `regression_model` An SVM regression model

**Value**

A list containing (1) stat: a data frame with all statistics for contamination estimation; (2) result: contamination estimation (Class = 0, pure; Class = 1, contaminated)

**Examples**

```r
data(vcf_example)
result <- sssc(file = vcf_example)
```
**summary_vcf**

**VCF Data Summary**

**Description**
Summarizes allele frequency information in scatter and density plots

**Usage**

```r
summary_vcf(vcf, ZG = NULL, CHR = NULL)
```

**Arguments**

- `vcf` VCF object from read_vcf function
- `ZG` zygosity: (1) null, for both het and hom, default; (2) het; (3) hom
- `CHR` chromosome number: (1) null, all chromosome, default; (2) any specific number

**Value**
A list containing (1) scatter: allele frequency scatter plot; (2) density: allele frequency density plot

**Examples**

```r
data("vcf_example")
 tmp <- summary_vcf(vcf = vcf_example, ZG = 'het', CHR = c(1,2))
plot(tmp$scatter)
plot(tmp$density)
```

---

**svm_class_model**

**Default svm classification model.**

**Description**
An svm object containing default svm classification model.

**Usage**

```r
svm_class_model
```

**Format**
An svm object:

**Source**
Created by Tao Jiang
**svm_regression_model**  
Default svm regression model.

**Description**  
An svm object containing default svm regression model.

**Usage**  

```r
svm_regression_model
```

**Format**  
An svm object:

**Source**  
Created by Tao Jiang

---

**train_ct**  
Train Contamination Detection Model

**Description**  
Trains two SVM models (classification and regression) to detects whether a sample is contaminated another sample of its same species.

**Usage**  

```r
train_ct(feature)
```

**Arguments**  

- `feature`  
Feature list objects from generate_feature()

**Value**  
A list contains two trained svm models: regression & classification
**update_vcf**

*Remove CNV regions within VCF files*

### Description

Remove CNV regions within VCF files

### Usage

```r
update_vcf(rmCNV = FALSE, vcf, cnvobj = NULL, threshold = 0.1, skew = 0.5, lower = 0.45, upper = 0.55)
```

### Arguments

- **rmCNV**: Remove CNV regions, default is `FALSE`
- **vcf**: Input VCF files
- **cnvobj**: cnv object, default is `NULL`
- **threshold**: Threshold for allele frequency, default is `0.1`
- **skew**: Skewness for allele frequency, default is `0.5`
- **lower**: Lower bound for allele frequency region, default is `0.45`
- **upper**: Upper bound for allele frequency region, default is `0.55`

### Value

VCF file without CNV region

---

**vcf_example**  
*VCF example file.*

### Description

An example containing a list of 4 data frames.

### Usage

```r
vcf_example
```

### Format

A list of 4 data frames:

### Source

Created by Tao Jiang
Index

* datasets
  - config_df, 2
  - svm_class_model, 15
  - svm_regression_model, 16
  - vcf_example, 17

config_df, 2

generate_feature, 3
getAlt2, 4
getAnnoRate, 4
getAvgLL, 5
getLowDepth, 5
getRatio, 6
getSkewness, 6
getSNVRate, 7
getVar, 7

locateFile, 8

negll, 8

read_vcf, 11
readGATK, 9
readStrelka, 9
readVarDict, 10
readVarPROWL, 11
rho_est, 12
rmChangePoint, 13
rmCNVinVCF, 13

sssc, 14
summary_vcf, 15
svm_class_model, 15
svm_regression_model, 16

train_ct, 16

update_vcf, 17

vcf_example, 17