Package ‘TrIdent’

July 26, 2025

Type Package

BiocType Software

Title TrIdent - Transduction Identification

Version 1.1.4

Description The `TrIdent` R package automates the analysis of transductomics data by detecting, classifying, and characterizing read coverage patterns associated with potential transduction events. Transductomics is a DNA sequencing-based method for the detection and characterization of transduction events in pure cultures and complex communities. Transductomics relies on mapping sequencing reads from a viral-like particle (VLP)-fraction of a sample to contigs assembled from the metagenome (whole-community) of the same sample. Reads from bacterial DNA carried by VLPs will map back to the bacterial contigs of origin creating read coverage patterns indicative of ongoing transduction.

License GPL-2

Encoding UTF-8

LazyData false

Imports graphics, utils, stats, dplyr, ggplot2, patchwork, stringr, tidyr, roll

biocViews Coverage, Metagenomics, PatternLogic, Classification, Sequencing

RoxygenNote 7.3.2

URL `https://github.com/jlmaier12/TrIdent`, `https://jlmaier12.github.io/TrIdent/`

BugReports `https://github.com/jlmaier12/TrIdent/issues`

Suggests BiocStyle, knitr, rmarkdown, kableExtra, testthat (>= 3.0.0)

VignetteBuilder knitr

Depends R (>= 4.2.0)

Config/testthat/edition 3

git_url https://git.bioconductor.org/packages/TrIdent

git_branch devel

git_last_commit 55904e9

git_last_commit_date 2025-06-11

Repository Bioconductor 3.22

Date/Publication 2025-07-25

Author Jessie Maier [aut, cre] (ORCID:
<`https://orcid.org/0009-0001-8575-5386`>),
Jorden Rabasco [aut, ctb] (ORCID:
<`https://orcid.org/0000-0002-6971-6678`>),
Craig Gin [aut] (ORCID: <`https://orcid.org/0000-0002-7447-663X`>),
Benjamin Callahan [aut] (ORCID:
<`https://orcid.org/0000-0002-8752-117X`>),
Manuel Kleiner [aut, ths] (ORCID:
<`https://orcid.org/0000-0001-6904-0287`>)

Maintainer Jessie Maier <`jlmaier@ncsu.edu`>

Contents

`TrIdent-package` ***TrIdent*** *-* ***Tr****ansduction* ***Ident****ification*

Description

Automatic detection, classification and characterization of transduction events in transductomics datasets using read coverage pattern-matching.

Please see [Transductomics: sequencing-based detection and analysis of transduced DNA in pure cultures and microbial communities] (https://microbiomejournal.biomedcentral.com/articles/10.1186/s40168-020-00935-5) for more information on the transductomics method, data and analysis workflow.

Details

The three main functions in TrIdent are:

1. `TrIdentClassifier` performs the pattern-matching, classification and characterization of read coverage patterns on contigs.
2. `plotTrIdentResults` plots the results from `TrIdentClassifier()`
3. `specializedTransductionID` searches contigs classified as Prophage-like by `TrIdentClassifier()` for potential specialized transduction

Author(s)

Jessie Maier <`jlmaier@ncsu.edu`> & Jorden Rabasco <`jrabasc@ncsu.edu`>

See Also

Useful links:

- `https://github.com/jlmaier12/TrIdent`
- `https://jlmaier12.github.io/TrIdent/`
- Report bugs at `https://github.com/jlmaier12/TrIdent/issues`

`allPatternMatches` *Collects pattern-match information for all classifications*

Description

Collects pattern information associated with all contigs classified as Prophage-like, Sloping and HighCovNoPattern.

Usage

```
allPatternMatches(bestMatchList, classifSummTable)
```

Arguments

`bestMatchList`	Classifications made with patternMatcher function.
`classifSummTable`	Classification summary table with whole-community:VLP-fraction read coverage ratios calculated.

Value

List

---

allProphageLikeClassifs

Collects Prophage-like classification pattern-match information

Description

Collects pattern information associated with all contigs classified as Prophage-like.

Usage

```
allProphageLikeClassifs(bestMatchList)
```

Arguments

`bestMatchList`	Classifications made with patternMatcher function.

Value

List

---

allSlopingClassifs

Collects Sloping classification pattern-match information

Description

Collects pattern information associated with all contigs classified as Sloping in the patternMatcher function.

Usage

```
allSlopingClassifs(bestMatchList)
```

Arguments

`bestMatchList`	Classifications made with patternMatcher function.

Value

List

`blockBuilder` *Builds prophage-like block patterns*

Description

Build and translate a block pattern going off the left side, right side and full length of the contig.

Usage

```
blockBuilder(viralSubset, windowSize, minBlockSize, maxBlockSize)
```

Arguments

`viralSubset`	A subset of the read coverage pileup that pertains only to the contig currently being assessed
`windowSize`	The window size used to re-average read coverage pileups
`minBlockSize`	The minimum size of the prophage-like block pattern. Default is 10000 bp.
`maxBlockSize`	The maximum size of the prophage-like block pattern. Default is NA.

Value

List containing three objects

`blockTranslator` *Full block pattern-translator*

Description

Translates full block-pattern across a contig. Translate the pattern 1000 bp at a time. Stop translating when the pattern is 5000 bp from the end of the contig.

Usage

```
blockTranslator(viralSubset, bestMatchInfo, windowSize, pattern)
```

Arguments

`viralSubset`	A subset of the read coverage pileup that pertains only to the contig currently being assessed
`bestMatchInfo`	The information associated with the current best pattern-match.
`windowSize`	The window size used to re-average read coverage pileups
`pattern`	A vector containing the values associated with the block pattern

Value

List

`changeSlope` *Change slope of sloping pattern*

Description

Change the value of the slope used for the sloping pattern-match

Usage

```
changeSlope(
  leftOrRight,
  slopeBottom,
  halfToMaxReadCov,
  cov,
  viralSubset,
  windowSize
)
```

Arguments

`leftOrRight`	Generate pattern for negative slope (left to right, i.e. 'Left') or positive slope (right to left, i.e. 'Right')
`slopeBottom`	The value for the bottom of the sloping value
`halfToMaxReadCov`	Half of the max VLP-fraction read coverage divided by 10
`cov`	The value for the top of the slope
`viralSubset`	A subset of the read coverage pileup that pertains only to the contig currently being assessed
`windowSize`	The window size used to re-average read coverage pileup

Value

List

`changeSlopeWStart` *Change slope of sloping pattern with initial start*

Description

Change the value of the slope used for the sloping with start pattern-match

Usage

```
changeSlopeWStart(
  leftOrRight,
  slopeBottom,
  slopeBottomChange,
  cov,
  viralSubset,
  windowSize
)
```

Arguments

`leftOrRight`	Generate pattern for negative slope (left to right, i.e. 'Left') or positive slope (right to left, i.e. 'Right')
`slopeBottom`	The value for the bottom of the sloping value
`slopeBottomChange`	The value used to increase the bottom of the slope
`cov`	The value for the top of the slope
`viralSubset`	A subset of the read coverage pileup that pertains only to the contig currently being assessed
`windowSize`	The window size used to re-average read coverage pileup

Value

List

`changeWindowSize` *Change the read coverage rolling mean window size*

Description

Re-averages window sizes of read coverage averages. Start with 100bp windows always. Cannot make window size less than 100bp.

Usage

```
changeWindowSize(cleanPileup, windowSize)
```

Arguments

`cleanPileup`	A read coverage dataset that has been cleaned and reformatted.
`windowSize`	The number of base pairs to average coverage values over. Options are 100, 500, 1000, or 2000 only!

Value

Dataframe

`contigClassSumm` *Summarizes pattern-match information*

Description

Summarizes the classifications made in the patternMatcher() function into a dataframe.

Usage

```
contigClassSumm(bestMatchList)
```

Arguments

`bestMatchList`	Classifications made with patternMatcher function.

Value

dataframe

`fullSlope` *Sloping pattern builder*

Description

Build a sloping pattern that consists of a sloping line spanning the contig being assessed. The line slopes from left to right. The slope of the line is changed, but the pattern is not translated across the contig.

Usage

```
fullSlope(viralSubset, windowSize, minSlope)
```

Arguments

`viralSubset`	A subset of the read coverage pileup that pertains only to the contig currently being assessed
`windowSize`	The window size used to re-average read coverage pileup
`minSlope`	The minimum slope value to test for sloping patterns

Value

List containing two objects

`leftRightBlockTranslater` *Translate left and right block patterns across contig*

Description

Translates left and right block patterns across contigs 1000 bp at a time

Usage

```
leftRightBlockTranslater(
  viralSubset,
  pattern,
  leftOrRight,
  windowSize,
  minReadCov,
  cov,
  bestMatchInfo,
  minBlockSize
)
```

Arguments

`viralSubset`	A subset of the read coverage pileup that pertains only to the contig currently being assessed
`pattern`	The pattern vector being translated
`leftOrRight`	Is the left or right block pattern being translated
`windowSize`	The window size used to re-average read coverage pileups
`minReadCov`	The baseline value used for the region outside of the block pattern (either 0 or the minimum VLP-fraction read coverage for the contig)
`cov`	The height value currently being used for the block pattern
`bestMatchInfo`	The information associated with the current best pattern-match.
`minBlockSize`	The minimum size of the Prophage-like block pattern. Default is 10,000 bp.

Value

List

`makeBlockPattern` *Make block patterns for pattern-matching*

Description

Make full, left and right block patterns for Prophage-like classifications

Usage

```
makeBlockPattern(
  viralSubset,
  windowSize,
  fullLeftRight,
  blockLength,
  nonBlock,
  minReadCov,
  cov
)
```

Arguments

`viralSubset`	A subset of the read coverage pileup that pertains only to the contig currently being assessed
`windowSize`	The window size used to re-average read coverage pileups
`fullLeftRight`	The block pattern variation being built
`blockLength`	Maximum block pattern length
`nonBlock`	Maximum non-block pattern length
`minReadCov`	Either 0 or the minimum VLP-fraction read coverage value
`cov`	The height value of the block pattern

Value

List containing two objects

`makeFullSlopes` *Make full slope patterns*

Description

Makes slope patterns sloping either left to right (Left) or right to left (right) across the contig being assessed.

Usage

```
makeFullSlopes(leftOrRight, viralSubset, newMax, minReadCov, windowSize)
```

Arguments

`leftOrRight`	Generate pattern for negative slope (left to right, i.e. ’Left’) or positive slope (right to left, i.e. ’Right’)
`viralSubset`	A subset of the read coverage pileup that pertains only to the contig currently being assessed
`newMax`	A value for the top of the sloping pattern that is slightly higher than the maximum coverage value on the viralSubset
`minReadCov`	Minimum read coverage value of the viralSubset
`windowSize`	The window size used to re-average read coverage pileups

Value

List

`makeSlopesWStarts` *Make slope patterns with starts*

Description

Makes slope patterns sloping either left to right (Left) or right to left (right) across the contig being assessed. Slope patterns contain an initiation point.

Usage

```
makeSlopesWStarts(leftOrRight, viralSubset, newMax, minReadCov, windowSize)
```

Arguments

`leftOrRight`	Generate pattern for negative slope (left to right, i.e. ’Left’) or positive slope (right to left, i.e. ’Right’)
`viralSubset`	A subset of the read coverage pileup that pertains only to the contig currently being assessed
`newMax`	A value for the top of the sloping pattern that is slightly higher than the maximum coverage value on the viralSubset
`minReadCov`	Minimum read coverage value of the viralSubset
`windowSize`	The window size used to re-average read coverage pileup

Value

List

`NARemover` *NA remover*

Description

Removes NAs from dataframe.

Usage

```
NARemover(x)
```

Arguments

`x`	dataset with potential NAs

Value

Dataframe

See Also

https://stackoverflow.com/questions/18142117/how-to-replace-nan-value-with-zero-in-a-huge-data-
18143097#18143097

`noPattern` *No pattern pattern-match*

Description

A horizontal line at the mean or median coverage should be an optimal pattern-match if the contig read coverage displays no sloping or block patterns

Usage

```
noPattern(viralSubset)
```

Arguments

`viralSubset`	A subset of the read coverage pileup that pertains only to the contig currently being assessed

Value

List

`patternBuilder` *Pattern-builder*

Description

Builds the pattern (vector) associated with the best pattern-match' for each contig classified as Prophage-like, Sloping, or HighCovNoPattern.

Usage

```
patternBuilder(viralSubset, classifList, classification, rowIndex)
```

Arguments

`viralSubset`	A subset of the read coverage pileup that pertains only to the contig currently being assessed
`classifList`	A list containing pattern match information associated with all classified contigs.
`classification`	The contig's classification assigned by the TrIdentClassifier function
`rowIndex`	The list index associated with each contig's pattern-match information

Value

Vector

`patternMatcher` *Main pattern-matching function*

Description

Creates the viralSubset, representative of one contig, that is used as input for each individual pattern-matching function. After the information associated with the best match for each pattern is obtained, the pattern with the smallest match score is used to classify the contig being assessed. Prior to the pattern-matching, contigs smaller than the minContigLength and contigs without 5,000 bp of 10x read coverage are removed.

Usage

```
patternMatcher(
  VLPpileup,
  WCpileup,
  windowSize,
  minBlockSize,
  maxBlockSize,
  minContigLength,
  minSlope,
  verbose
)
```

Arguments

`VLPpileup`	A table containing contig names, coverages averaged over 100 bp windows, and contig positions associated with mapping VLP-fraction reads to whole-community contigs
`WCpileup`	A table containing contig names, coverages averaged over 100 bp windows, and contig positions associated with mapping whole-community reads to whole-community contigs
`windowSize`	The window size used to re-average read coverage datasets
`minBlockSize`	The minimum size of the prophage-like block pattern. Default is 10,000 bp.
`maxBlockSize`	The maximum size of the prophage-like block pattern. Default is NA
`minContigLength`	The minimum contig size (in bp) to perform pattern-matching on. Must be at least 20,000 bp. Default is 30,000 bp.
`minSlope`	The minimum slope value to test for sloping patterns
`verbose`	TRUE or FALSE. Print progress messages to console. Default is TRUE.

Value

List containing three objects.

`patternMatchSize` *Pattern-match size calculator*

Description

Calculate the size (bp) of the matching region for Prophage-like and Sloping patterns

Usage

```
patternMatchSize(classifSumm, classifList, windowSize, verbose)
```

Arguments

`classifSumm`	Classification summary table
`classifList`	A list containing pattern match information associated with all contig classifications
`windowSize`	The window size used to re-average read coverage pileups
`verbose`	TRUE or FALSE. Print progress messages to console. Default is TRUE.

Value

dataframe

`pileupFormatter` *Correctly formats pileup files.*

Description

Places columns in correct order and renames columns. Cleans the contig labels to remove excess information after whitespace.

Usage

```
pileupFormatter(pileup)
```

Arguments

`pileup`	A table containing contig names, read coverages averaged over 100 bp windows,and contig positions

Value

dataframe

`plotTrIdentResults` *Plot read coverage graphs of contigs classified as Prophage-like, Sloping, or HighCovNoPattern*

Description

Plot the read coverages of a contig and its associated pattern-match for Prophage-like, Sloping and HighCovNoPattern classifications. Returns a list of ggplot objects.

Usage

```
plotTrIdentResults(
  VLPpileup,
  WCpileup,
  TrIdentResults,
  onlyPlot,
  logScale = FALSE,
  saveFilesTo
)
```

Arguments

`VLPpileup`	VLP-fraction pileup file generated by mapping sequencing reads from a sample's ultra-purified VLP-fraction mapped to the sample's whole-community metagenome assembly. The pileup file MUST have the following format: * V1: Contig accession * V2: Mapped read coverage values averaged over 100 bp windows * V3: Starting position (bp) of each 100 bp window. Restarts from 0 at the start of each new contig. * V4: Starting position (bp) of each 100 bp window. Does NOT restart at the start of each new contig.
`WCpileup`	A whole-community pileup file generated by mapping sequencing reads from a sample's whole-community mapped to the sample's whole-community metagenome assembly. The pileup file MUST have the following format: * V1: Contig accession * V2: Mapped read coverage values averaged over 100 bp windows * V3: Starting position (bp) of each 100 bp window. Restarts from 0 at the start of each new contig. * V4: Starting position (bp) of each 100 bp window. Does NOT restart at the start of each new contig.
`TrIdentResults`	Output from 'TrIdentClassifier()'.
`onlyPlot`	Optional, use to 'only plot' the contigs classified as either "Prophage-like", "Sloping", or "HighCovNoPattern".
`logScale`	TRUE or FALSE, display VLP-fraction read coverage in log10 scale. Default is FALSE.
`saveFilesTo`	Optional, Provide a path to the directory you wish to save output to. A folder will be made within the provided directory to store results.

Value

Large list containing ggplot objects

Examples

```
data("VLPFractionSamplePileup")
data("WholeCommunitySamplePileup")
data("TrIdentSampleOutput")

patternMatches <- plotTrIdentResults(
  VLPpileup = VLPFractionSamplePileup,
  WCpileup = WholeCommunitySamplePileup,
  TrIdentResults = TrIdentSampleOutput
)
```

`prophageLikeBorders` *Prophage-like border finder*

Description

Find borders of Prophage-like patterns with more specificity than pattern-matching using 100 bp window pileups and sliding standard deviation technique.

Usage

```
prophageLikeBorders(viralSubset, classificationPatterns, i, windowSize)
```

Arguments

`viralSubset`	A subset of the read coverage pileup that pertains only to the contig currently being assessed
`classificationPatterns`	The pattern match information associated with each contig classified as Prophage-like, Sloping, or HighCovNoPattern
`i`	The index for the contig currently being assessed
`windowSize`	The window size used to re-average read coverage pileups

Value

List

`prophageLikeElevation` *Determine Prophage-like read coverage elevation in whole-community*

Description

Determines whether a detected Prophage-like genetic element has read coverage in the whole-community that is either elevated or depressed compared to the average read coverage of the non-prophage region.

Usage

```
prophageLikeElevation(
  classifSummTable,
  prophageLikeClassifList,
  VLPpileup,
  WCpileup,
  windowSize,
  verbose
)
```

Arguments

`classifSummTable`	Classification summary table
`prophageLikeClassifList`	A list containing pattern match information associated with all contigs classified as Prophage-like.
`VLPpileup`	A table containing contig names, coverages averaged over 100 bp windows, and contig positions associated with mapping VLP-fraction reads to whole-community contigs
`WCpileup`	A table containing contig names, coverages averaged over 100 bp windows, and contig positions associated with mapping whole-community reads to whole-community contigs
`windowSize`	The window size used to re-average read coverage pileups
`verbose`	TRUE or FALSE. Print progress messages to console. Default is TRUE.

Value

dataframe

`prophageLikeZoom` *Prophage-like pattern zoom*

Description

'Zoom-in' on (aka subset) desired region surrounding block pattern.

Usage

```
prophageLikeZoom(viralSubset, classificationPatterns, i, zoom, windowSize)
```

Arguments

`viralSubset`	A subset of the read coverage pileup that pertains only to the contig currently being assessed
`classificationPatterns`	The pattern match information associated with each contig classified as Prophage-like, sloping, or HighCovNoPattern
`i`	The index for the contig currently being assessed

`zoom`	The number of rows outside the start and stop positions of the block pattern to zoom-in on
`windowSize`	The window size used to re-average read coverage pileups

Value

Dataframe

`resultsHisto` *Create histogram of normalized pattern-match scores*

Description

Plots a histogram of normalized match scores for all Prophage-like, Sloping and HighCovNoPattern classifications and colors the plot based on the classifications.

Usage

```
resultsHisto(summaryList)
```

Arguments

`summaryList`	Classification summary table filtered to only include contigs with Prophage-like, Sloping and HighCovNoPattern classifications

Value

ggplot object

`slopeSumm` *Summarize slopes for sloping classifications*

Description

Add slope information for sloping classifications to summary table

Usage

```
slopeSumm(classifSumm, slopingClassifList, windowSize)
```

Arguments

`classifSumm`	Classification summary table
`slopingClassifList`	A list containing pattern match information associated with all contigs classified as sloping.
`windowSize`	The window size used to re-average read coverage pileups

Value

dataframe

`slopeTranslator` *Sloping pattern translator*

Description

Translates a sloping pattern containing the initial jump-up in read coverage across a contig. Translate the pattern 1000 bp at a time. Stop translating when the pattern left on the contig reaches 20,000 bp.

Usage

```
slopeTranslator(
  viralSubset,
  bestMatchInfo,
  windowSize,
  slopeChange,
  leftOrRight
)
```

Arguments

`viralSubset`	A subset of the read coverage pileup that pertains only to the contig currently being assessed
`bestMatchInfo`	The pattern-match information associated with the current best pattern match.
`windowSize`	The window size used to re-average read coverage pileups
`slopeChange`	A list containing pattern vector, slope value, and value of slope bottom
`leftOrRight`	The direction of the sloping pattern. Either "Left" for left to right (neg) slopes or "Right" for right to left (pos) slopes.

Value

List

`slopeWithStart` *Sloping pattern with an initial jump-up in read coverage*

Description

Build, translate, and change slope of sloping pattern with slope start

Usage

```
slopeWithStart(viralSubset, windowSize, minSlope)
```

Arguments

`viralSubset`	A subset of the read coverage pileup that pertains only to the contig currently being assessed
`windowSize`	The window size used to re-average read coverage pileups
`minSlope`	The minimum slope value to test for sloping patterns

Value

List containing two objects

---

specializedTransductionID

Identify potential specialized transduction events on contigs classified as Prophage-like

Description

Search contigs classified as Prophage-like for dense read coverage outside of the pattern-match borders that may indicate specialized transduction. Returns a list with the first object containing a summary table and the second object containing a list of plots of with associated specialzied transduction search results. If the plot is green, it has been identified as having potential specialized transduction.

Usage

```
specializedTransductionID(
  VLPpileup,
  TrIdentResults,
  specificContig,
  noReadCov = 500,
  specTransLength = 2000,
  logScale = FALSE,
  verbose = TRUE,
  SaveFilesTo
)
```

Arguments

`VLPpileup`	VLP-fraction pileup file generated by mapping sequencing reads from a sample's ultra-purified VLP-fraction mapped to the sample's whole-community metagenome assembly. The pileup file MUST have the following format: * V1: Contig accession * V2: Mapped read coverage values averaged over 100 bp windows * V3: Starting position (bp) of each 100 bp window. Restarts from 0 at the start of each new contig. * V4: Starting position (bp) of each 100 bp window. Does NOT restart at the start of each new contig.
`TrIdentResults`	Output from ‘TrIdentClassifier()‘
`specificContig`	Optional, Search a specific contig classified as Prophage-like ("NODE_1").
`noReadCov`	Number of basepairs of zero read coverage encountered before specialized transduction searching stops. Default is 500. Must be at least 100.
`specTransLength`	Number of basepairs of non-zero read coverage needed for specialized transduction to be considered. Default is 2000. Must be at least 100.
`logScale`	TRUE or FALSE, display VLP-fraction read coverage in log10 scale. Default is FALSE.
`verbose`	TRUE or FALSE. Print progress messages to console. Default is TRUE.
`SaveFilesTo`	Provide a path to the directory you wish to save output to. ‘specializedTransductionID()‘ will make a folder within the provided directory to store results.

Value

Large list containing two objects

Examples

```
data("VLPFractionSamplePileup")
data("TrIdentSampleOutput")

specTransduction <- specializedTransductionID(
  VLPpileup = VLPFractionSamplePileup,
  TrIdentResults = TrIdentSampleOutput
)

specTransductionNODE62 <- specializedTransductionID(
  VLPpileup = VLPFractionSamplePileup,
  TrIdentResults = TrIdentSampleOutput,
  specificContig = "NODE_62"
)
```

`specTransductionPlot` *Specialized transduction plot*

Description

Plot search results of ‘specializedTransductionID()‘

Usage

```
specTransductionPlot(
  viralSubsetZoom,
  startPosBp,
  endPosBp,
  SpecTransLeft,
  specTransRight,
  contigName,
  classifPatternMatches,
  i,
  specTransSumm,
  logScale,
  classifSumm
)
```

Arguments

`viralSubsetZoom`	contig subset surrounding Prophage-like pattern-match
`startPosBp`	Left border position
`endPosBp`	Right border position
`SpecTransLeft`	End position of spec transduction on left border
`specTransRight`	End position of spec transduction on right border

`contigName`	The reference name of the contig currently being assessed (i.e "NODE_1")
`classifPatternMatches`	The pattern match information associated with each contig classified as prophage-like, sloping, or HighCovNoPattern
`i`	The index for the contig currently being assessed
`specTransSumm`	Results for spec transduction search
`logScale`	If TRUE, coverage is plotted in log10. If FALSE, raw coverage values are plotted. Default is FALSE.
`classifSumm`	The summary information associated with each contig classified as Prophage-like, Sloping, or HighCovNoPattern

Value

ggplot object

`specTransductionSearch` *Specialized transduction search and plot*

Description

Search contigs classified as prophage-like for potential specialized transduction and return the plot visualizing the search results.

Usage

```
specTransductionSearch(
  contigName,
  VLPpileup,
  classifPatternMatches,
  classifSumm,
  windowSize,
  i,
  noReadCov,
  specTransLength,
  logScale
)
```

Arguments

`contigName`	The reference name of the contig currently being assessed (i.e "NODE_1")
`VLPpileup`	A table containing contig names, coverages averaged over 100 bp windows, and contig positions associated with mapping VLP-fraction reads to whole-community contigs
`classifPatternMatches`	The pattern match information associated with each contig classified as prophage-like, sloping, or HighCovNoPattern
`classifSumm`	The summary information associated with each contig classified as Prophage-like, Sloping, or HighCovNoPattern

`windowSize`	The window size used to re-average read coverage pileups
`i`	The index for the contig currently being assessed
`noReadCov`	How many bp of no read coverage are encountered before searching stops? Default is 500.
`specTransLength`	How many bp of read coverage to look for outside of prophage borders? Default is 2000.
`logScale`	If TRUE, coverage is plotted in log10. If FALSE, raw coverage values are plotted. Default is FALSE.

Value

List containing two objects

`TrIdentClassifier` *Classify contigs as Prophage-like, Sloping, HighCovNoPattern, and NoPattern*

Description

Performs all the pattern-matching and summarizes the results into a list. The first item in the list is a table consisting of the summary information of all the contigs that passed through pattern-matching (i.e were not filtered out). The second item in the list is a table consisting of the summary information of all contigs that were classified via pattern-matching. The third item in the list contains the pattern-match information associated with each contig in the previous table. The fourth object in the list is a table containing the contigs that were filtered out prior to pattern-matching. The fifth item is the windowSize used for the search.

Usage

```
TrIdentClassifier(
  VLPpileup,
  WCpileup,
  windowSize = 1000,
  minBlockSize = 10000,
  maxBlockSize = Inf,
  minContigLength = 30000,
  minSlope = 0.001,
  VLPReads,
  WCReads,
  verbose = TRUE,
  SaveFilesTo
)
```

Arguments

`VLPpileup`	VLP-fraction pileup file generated by mapping sequencing reads from a sample's ultra-purified VLP-fraction mapped to the sample's whole-community metagenome assembly. The pileup file MUST have the following format: * V1: Contig accession * V2: Mapped read coverage values averaged over 100 bp windows * V3: Starting position (bp) of each 100 bp window. Restarts from 0 at the start

	of each new contig. * V4: Starting position (bp) of each 100 bp window. Does NOT restart at the start of each new contig.
`WCpileup`	A whole-community pileup file generated by mapping sequencing reads from a sample's whole-community mapped to the sample's whole-community metagenome assembly. The pileup file MUST have the following format: * V1: Contig accession * V2: Mapped read coverage values averaged over 100 bp windows * V3: Starting position (bp) of each 100 bp window. Restarts from 0 at the start of each new contig. * V4: Starting position (bp) of each 100 bp window. Does NOT restart at the start of each new contig.
`windowSize`	The number of basepairs to average read coverage values over. Options are 100, 200, 500, 1000 ONLY. Default is 1000.
`minBlockSize`	The minimum size (in bp) of the Prophage-like block pattern. Default is 10000. Must be at least 1000.
`maxBlockSize`	The maximum size (in bp) of the Prophage-like block pattern. Default is NA (no maximum).
`minContigLength`	The minimum contig size (in bp) to perform pattern-matching on. Must be at least 25000. Default is 30000.
`minSlope`	The minimum slope value to test for sloping patterns. Default is 0.001 (i.e minimum change of 10x read coverage over 100,000 bp).
`VLPReads`	Optional, the number of VLP-fraction reads used for mapping and creation of pileup.
`WCReads`	Optional, the number of WC reads used for mapping and creation of pileup.
`verbose`	TRUE or FALSE. Print progress messages to console. Default is TRUE.
`SaveFilesTo`	Optional, Provide a path to the directory you wish to save output to. A folder will be made within the provided directory to store results.

Value

Large list containing 5 objects

Examples

```
data("VLPFractionSamplePileup")
data("WholeCommunitySamplePileup")

TrIdent_results <- TrIdentClassifier(
  VLPpileup = VLPFractionSamplePileup,
  WCpileup = WholeCommunitySamplePileup
)
```

`TrIdentSampleOutput` *TrIdentSampleOutput*

Description

The TrIdentClassifier output from the VLPFractionSamplePileup and WholeCommunitySamplePileup files run with default parameters Report...

Usage

```
data('TrIdentSampleOutput')
```

Format

'TrIdentSampleOutput' A list with 6 objects:

SummaryTable A dataframe containing classifications for all contigs that were processed with pattern-matching

CleanedSummaryTable SummaryTable dataframe filtered to remove contigs that recieved a 'None' classification

PatternMatchInfo A list of lists containing pattern-match information for each classified contig

FilteredOutContigTable A dataframe containing names of contigs that were filtered out prior to pattern-matching

windowSize windowSize used in TrIdentClassifier function (1000)

ResultHistogram a histogram displaying the overall abundance and quality of pattern-matches in addition to the composition of classifications. The displayed pattern-match scores are normalized by dividing each score by its associated contig length. The scores are normalized to visualize the overall quality of pattern-matching for the entire dataset.

Details

A list object produced by the TrIdentClassifier function run on the VLPFractionSamplePileup and WholeCommunitySamplePileup files run with default parameters

---

`VLPFractionSamplePileup`

VLP-Fraction of Sample Dataset

Description

A subset of contigs from the raw VLP-fraction read coverage pileup file generated from BBMap's pileup.sh. Report...

Usage

```
data('VLPFractionSamplePileup')
```

Format

'VLPFractionSamplePileup' A data frame with 10,805 rows and 4 columns:

V1 Contig accession

V2 Mapped read coverage averaged over a 100 bp window size

V3 Starting position (bp) of each 100 bp window. Restarts from 0 at the start of each new contig.

V4 Starting position (bp) of each 100 bp window. Does NOT restart at the start of each new contig.

Details

This dataset represents one half of a complete transductomics dataset which is comprised of two parts-a whole-community fraction and a viral-like particle (VLP)-fraction. This dataset represents the VLP fraction and was generated by purifying VLPs from a conventional mouse fecal homogenate using CsCl density gradient ultracentrifugation. The VLP-fraction extracted DNA was sequenced with Illumina (paired-end mode, 150 bp reads) The sequencing reads were mapped to the associated whole-community assembly using BBMap. The bbmap.sh bincov parameter with covbinsize=100 was used to create a pileup file with 100 bp windows. A subset of 10 contigs from the pileup file were selected for this sample dataset. The contigs were chosen because their associated read coverage patterns in the VLP-fraction exemplify TrIdent's pattern-matching functionality across classifications: NODE_617:Prophage-like, active/abundant, with spec transduction NODE_135:Prophage-like, off one side of contig, no spec transduction NODE_352:Sloping, left to right slope NODE_1088: Sloping, right to left slope NODE_2060: Sloping, right to left slope with start NODE_1401: None, no pattern match NODE_62: Prophage-like, with spec transduction NODE_368: Prophage-like, not homogeneously integrated/present, no spec transduction NODE_560: HighCovNoPattern NODE_1165: None, filtered out To access the sequencing data used to generate this pileup file and for additional details on the assembly and mapping parameters, refer to the reference below: Reference: Kleiner, M., Bushnell, B., Sanderson, K.E. et al. Transductomics: sequencing-based detection and analysis of transduced DNA in pure cultures and microbial communities. Microbiome 8, 158 (2020). https://doi.org/10.1186/s40168-020-00935-5

Source

<https://microbiomejournal.biomedcentral.com/articles/10.1186/s40168-020-00935-5>

`VLPtoWCRatioCalc` *VLP-fraction:whole-community read coverage ratio calculator*

Description

Calculate the VLP-fraction:whole-community read coverage ratio for every contig using the median read coverage values. If the ratio is greater than 2 (i.e VLP-fraction read coverage is, on average, at least double the whole-community read coverage), then the contig is classified as HighCovNoPattern. If the number of VLP-fraction and whole-community reads used for mapping are provided, then the VLP/WC ratio value will be normalized to the sizes of the VLP and WC read sets.

Usage

```
VLPtoWCRatioCalc(classifSumm, WCpileup, VLPpileup, VLPReads, WCReads)
```

Arguments

`classifSumm`	Classification summary table
`WCpileup`	A table containing contig names, coverages averaged over 100 bp windows, and contig positions associated with mapping whole-community reads to whole-community contigs
`VLPpileup`	A table containing contig names, coverages averaged over 100 bp windows, and contig positions associated with mapping VLP-fraction reads to whole-community contigs
`VLPReads`	The number of VLP-fraction reads used for mapping and creation of pileup.
`WCReads`	The number of WC reads used for mapping and creation of pileup.

Value

dataframe

`WholeCommunitySamplePileup`

Whole-Community Fraction of Sample Dataset

Description

A subset of contigs from the raw whole-community fraction read coverage pileup file generated during read mapping. Report...

Usage

```
data('WholeCommunitySamplePileup')
```

Format

'WholeCommunitySamplePileup' A data frame with 10,805 rows and 4 columns:

V1 Contig accession

V2 Mapped read coverage averaged over a 100 bp window size

V3 Starting position (bp) of each 100 bp window. Restarts from 0 at the start of each new contig.

V4 Starting position (bp) of each 100 bp window. Does NOT restart at the start of each new contig.

Details

This dataset represents one half of a complete transductomics dataset which is comprised of two parts-a whole-community fraction and a viral-like particle (VLP)-fraction. This dataset represents the whole-community fraction and was generated from a conventional mouse fecal homogenate. The whole-community extracted DNA was sequenced with Illumina (paired-end mode, 150 bp reads) after which the metagenome was assembled. The sequencing reads were mapped to the assembled contigs using BBMap. The bbmap.sh bincov parameter with covbinsize=100 was used to create a pileup file with 100 bp windows. A subset of 10 contigs from the pileup file were selected for this sample dataset. The contigs were chosen because their associated read coverage patterns in the VLP-fraction exemplify TrIdent's pattern-matching and characterization functionality across classifications: NODE_617:Prophage-like, active/abundant, with spec transduction NODE_135:Prophage-like, off one side of contig, no spec transduction NODE_352:Sloping, left to right slope NODE_1088: Sloping, right to left slope NODE_2060: Sloping, right to left slope with start NODE_1401: None, no pattern match NODE_62: Prophage-like, with spec transduction NODE_368: Prophage-like, not homogeneously integrated/present, no spec transduction NODE_560: HighCovNoPattern NODE_1165: None, filtered out To access the sequencing data used to generate this pileup file and for additional details on the assembly and mapping parameters, refer to the reference below: Reference: Kleiner, M., Bushnell, B., Sanderson, K.E. et al. Transductomics: sequencing-based detection and analysis of transduced DNA in pure cultures and microbial communities. Microbiome 8, 158 (2020). https://doi.org/10.1186/s40168-020-00935-5

Source

<https://microbiomejournal.biomedcentral.com/articles/10.1186/s40168-020-00935-5>

`zeroCountSearch` *Counts zero values to the left and right of prophage-like borders*

Description

Checks to see at which point the number of consecutive zero values to the left and right of the prophage-like pattern match borders equals the noReadCov parameter

Usage

```
zeroCountSearch(startOrEnd, viralSubsetZoom, startOrEndPosRow, noReadCov)
```

Arguments

`startOrEnd`	searching the start (left side) or end (right side) of the prophage-like pattern-match
`viralSubsetZoom`	viralSubset dataframe subsetted to 50,000 bp outside the pattern match borders
`startOrEndPosRow`	The row index of the start or end position of the prophage-like pattern match
`noReadCov`	How many bp of no read coverage are encountered before specialized transduction searching stops? Default is 500.

Value

List

Index