Package 'CNOGpro'

February 19, 2015

Type Package

Title Copy Numbers of Genes in prokaryotes

Version 1.1

Date 2015-01-12

Author Ola Brynildsrud, Lars-Gustav Snipen

Maintainer Ola Brynildsrud <ola.brynildsrud@nmbu.no>

Description Methods for assigning copy number states and breakpoints in resequencing experiments of prokaryotic organisms.

License GPL-2

Depends seqinr

NeedsCompilation no

Repository CRAN

Date/Publication 2015-01-12 23:06:41

R topics documented:

CNOGpro-package *CNOGpro - Copy Numbers of Genes in Prokaryotes*

Description

Detects copy number variation in resequencing experiments

Details

Package:	CNOGpro
Type:	Package
Version:	1.1
Date:	2015-01-12
License:	GPL-2

After mapping the reads from a resequencing experiment to a reference organism, CNOGpro allows the user to count coverage of the reference in sliding windows, normalize the counts based on observed GC-bias, and run two main models for inferring copy number variation and breakpoints - One approach using a Hidden Markov Model, and one using bootstrapping. The latter will also produce confidence intervals of the desired range around the observed copy number.

Author(s)

Ola Brynildsrud Maintainer: Ola Brynildsrud <ola.brynildsrud@nmbu.no>

References

Manuscript forthcoming...

carsonella *CNV experiment of C. Carsonella ruddy (artificial data)*

Description

This is an example object of class CNOGpro, as created by the CNOGpro method. The input to this file was a GenBank flat file (Acc. nr: CP003467), and a table of best-hit read locations from a resequencing experiment. The data were created by artificially duplicating one gene and removing another from the chromosome, and is meant for illustration purposes only. The reads were created by the ART sequencing simulator by Huang et al. (2011)

Usage

```
data(carsonella)
```

Format

An object of class CNOGpro

Source

Artificial data, produced by the author

References

Huang,W. et al. (2011) ART: a next-generation sequencing read simulator. *Bioinformatics*, **28**, 593–594.
Nakabachi,A. et al. (2013) Defensive bacteriome symbiont with a drastically reduced genome. *Curr. Biol.*, **23**, 1478 – 1484

Examples

```
data(carsonella)
summaryCNOGpro(carsonella)
plotCNOGpro(carsonella)
```

CNOGpro *Set up a new CNOGpro experiment*

Description

This function initializes the CNOGpro experiment.

Usage

```
CNOGpro(hitsfile, gbkfile, windowlength = 100, name =
"Default organism")
```

Arguments

hitsfile	Path to a text file containing two columns: The first column is a chromosome identifier, and the second column is the leftmost coordinate (in the reference organism) of a mapped read. Each row represents one single read.
gbkfile	The GenBank Flat Format file of the organism used as a reference in the re-sequencing study. Must contain DNA sequence, not just annotations. Multi-chromosomal files are not currently supported, and should be parsed separately.
windowlength	Integer length of the sliding window which will be used when counting read density.
name	(Optional) A name of the organism or copy number experiment.

Details

The function accepts a minimum input of the GenBank file, but will not be able to do any copy number analysis in this mode. As such, the primary use of this modes when the user just wants to print a table of genetic elements.

The hitsfile can be constructed from a SAMtools binary alignment/ map (.bam) file by entering the following at the shell command line:

```
samtools view infile.bam | perl -lane 'print \"$F[2]\t$F[3]\"' > out.hits
```

Value

An object of class `CNOGpro`, which can be used in further CNV analysis. Essentially a list containing the following elements:

`Name`	The name of the organism or project, as set by this method.
`windowlength`	The length of the sliding window when counting reads
`accession`	The accession number of the chromosome, from the provided GenBank file
`genes`	A table of genetic elements in the chromosome, including intergenic regions. The table will also hold copy numbers as inferred from the `runHMM` and `runBootstrap` methods.
`chrlength`	The length of the chromosome
`GCperwindow`	The GC-percentage, calculated in sliding windows.
`is_GC_normalized`	logical indicating whether data has been normalized or not
`ReadsprWindow`	The (raw) read counts in sliding windows
`CorrReadsprWindow`	The read counts in sliding windows, corrected for GC bias. Acquired from the `normalizeGC` method.
`HMMtable`	The data frame of breakpoints and states inferred from the `runHMM` method.

Author(s)

Ola Brynildsrud (ola.brynildsrud@nmbu.no)

References

Manuscript not yet published.

See Also

`normalizeGC`, `runBootstrap`, `runHMM`, `store`

Examples

```
carsonella <- CNOGpro(hitsfile=system.file("extdata/carsonellahits.tab", package="CNOGpro"),
gbkfile=system.file("extdata/CP003467.gbk",package="CNOGpro"), windowlength=100,
name="Artificial Carsonella ruddii data")
plotCNOGpro(carsonella)
printCNOGpro(carsonella)
```

`normalizeGC` *Normalization of read count data by GC content*

Description

Manipulates count values from windows of comparatively high or low GC content so as to bring the counts more in line with those from median GC-content windows.

Usage

```
normalizeGC(experiment)
```

Arguments

`experiment`	`experiment` An object of class `CNOGpro`.

Details

Implements the Yoon et al. (2009) normalization scheme.

Value

An object of class `CNOGpro`, containing a numeric vector `CorrReadsprWindow`.

Author(s)

Ola Brynildsrud

References

Yoon,S. et al. (2009) Sensitive and accurate detection of copy number variants using read depth of coverage. *Genome Research*, **19**, 1586–1592.

Examples

```
data(carsonella)
carsonella_normalized <- normalizeGC(carsonella)
plotCNOGpro(carsonella_normalized)
```

plotCNOGpro *Plots the results of a* CNOGpro *experiment*

Description

Plots the coverage in a number of ways: Coverage along the chromosome, density plots of coverage for each individual copy number state (as assigned by the runHMMmethod) and coverage by GC percentage of the reads, before and after normalization.

Usage

```
plotCNOGpro(experiment)
```

Arguments

experiment	An object of class CNOGpro

Author(s)

Ola Brynildsrud

See Also

CNOGpro

Examples

```
data(carsonella)
carsonella_normalized <- normalizeGC(carsonella)
carsonella_hmm <- runHMM(carsonella_normalized, nstates=5)
plotCNOGpro(carsonella_hmm)
```

printCNOGpro *Prints the results of a* CNOGpro *experiment*

Description

Results are printed to stout. The default behavior is to print the HMMtable. If this does not exist, the function will attempt to print the gene table (with copy numbers.)

Usage

```
printCNOGpro(experiment)
```

Arguments

experiment	An object of class CNOGpro

Author(s)

Ola Brynildsrud

See Also

CNOGpro

Examples

```
data(carsonella)
carsonella_normalized <- normalizeGC(carsonella)
carsonella_bootstrapped <- runBootstrap(carsonella_normalized, quantiles=c(0.01,0.99))
printCNOGpro(carsonella_bootstrapped)
```

---

runBootstrap *Gene copy numbers with confidence intervals through bootstrapping*

Description

For each genetic element in the gene table of a CNOGpro object, sample repeatedly from the read observations associated with that gene while replacing the observations.

Usage

```
runBootstrap(experiment, replicates = 1000, quantiles = c(0.025, 0.975))
```

Arguments

experiment	An object of class CNOGpro
replicates	The (integer) number of resampling experiments per genetic element. (That is, the number of bootstrap samples.)
quantiles	The quantiles of the distribution of mean copy numbers for which the mean is returned. (Given as c(lower, upper))

Details

Implements the theory that since the read counts from each genetic element come from a (overdispersed) Poisson distribution with rate parameter lambda (=mean), this lambda can be inferred by simply taking the mean of the observations from that genetic element. A confidence interval is then constructed around the mean by random resampling with replacement. For each replicate and genetic element, the same number of samples is taken as there are read count observations associated with that particular genetic element. The observations of means are then sorted and the given lower and upper quantiles are returned.

Value

An object of class CNOGpro, with proposed copy number (floating point) and the associated confidence interval for each genetic element in the table of genes.

Author(s)

Ola Brynildsrud

Examples

```
data(carsonella)
carsonella_normalized <- normalizeGC(carsonella)
carsonella_bootstrapped <- runBootstrap(carsonella, replicates=500)
printCNOGpro(carsonella_bootstrapped)
```

---

runHMM *Copy number variation inference through a Hidden Markov Model*

Description

Implements a Viterbi algorithm for assigning most likely copy number to each chromosomal position in the chromosome.

Usage

```
runHMM(experiment, nstates = 5, changeprob = 1e-04, includeZeroState = T,
errorRate = 0.001)
```

Arguments

experiment	An object of class CNOGpro
nstates	The possible number of states, not including state 0. The returned copy numbers will be in the range 0, 1, 2, ... , nstates
changeprob	The probability of transitioning from one state to another, used to set up the transition matrix.
includeZeroState	Whether or not to allow the copy number state 0 in the results
errorRate	The presumed fraction of erroneously mapped reads. Only needed when includeZeroState is set to TRUE. This numbers is used for setting the probability distribution of each observation in copy number state 0.

Details

For each read count observation the algorithm computes the probability of that observation in each possible state. The minimum path through the trellis is then calculated at the end.

Value

An object of class CNOGpro, with a HMMtable listing the breakpoints of different copy number states. The most probable states of each genetic element are also listed in the genes table of the object.

Author(s)

Ola Brynildsrud

Examples

```
data(carsonella)
carsonella_normalized <- normalizeGC(carsonella)
carsonella_hmm <- runHMM(carsonella_normalized)
plotCNOGpro(carsonella_hmm)
```

---

store *Store the results of a* CNOGpro *copy number experiment*

Description

Prints the relevant data to a text file. By default, the entire gene table including copy numbers inferred by bootstrapping and HMM is printed. However, using the flag outputEntireTable=F will make the function print only the HMM breakpoints with associated most probable states.

Usage

```
store(experiment, outputEntireTable = T, path = "./")
```

Arguments

experiment	An object of class CNOGpro.
outputEntireTable	A logical flag indicating if the entire genes attribute table is to be printed. If set to FALSE, only the HMM table of breakpoints and states will be printed.
path	The path to be written to.

Details

Outputs data to a tab-separated text-file. User must have permission to print to the output directory.

Author(s)

Ola Brynildsrud

Examples

```
data(carsonella)
## Not run: store(carsonella, path="./")
```

summaryCNOGpro *Summarize the* CNOGpro *experiment*

Description

Prints basic parameters of the experiment.

Usage

```
summaryCNOGpro(experiment)
```

Arguments

experiment	An object of type CNOGpro

Author(s)

Ola Brynildsrud

Examples

```
data(carsonella)
summaryCNOGpro(carsonella)
```

Index