

# Package ‘PING’

October 12, 2016

**Type** Package

**Title** Probabilistic inference for Nucleosome Positioning with  
MNase-based or Sonicated Short-read Data

**Version** 2.16.0

**Author** Xuekui Zhang <xuezhang@jhsph.edu>, Raphael Gottardo  
<raph@stat.ubc.ca>, Sangsoon Woo, <swoo@fhcrc.org>

**Maintainer** Renan Sauteraud <rsautera@fhcrc.org>

**Depends** R(>= 2.15.0), chipseq, IRanges, GenomicRanges

**Imports** methods, PICS, graphics, grDevices, stats, Gviz, fda,  
BSgenome, stats4, BiocGenerics, IRanges, S4Vectors

**Suggests** parallel, ShortRead, rtracklayer

**Description** Probabilistic inference of ChIP-Seq using an empirical  
Bayes mixture model approach.

**Collate** setClasses.R setMethods.R PING.R postPING.R segmentPING.R

**License** Artistic-2.0

**biocViews** Clustering, StatisticalMethod, Visualization, Sequencing

**NeedsCompilation** yes

## R topics documented:

CoverageTrack . . . . .	2
NucleosomeTrack . . . . .	3
PICS_IMPORT . . . . .	4
ping . . . . .	4
ping-class . . . . .	6
pingError-class . . . . .	8
pingList-class . . . . .	9
plotSummary . . . . .	12
postPING . . . . .	13
RawReadsTrack . . . . .	15
segmentPING . . . . .	16
show . . . . .	17
summary . . . . .	18

---

CoverageTrack	<i>Reads coverage track</i>
---------------	-----------------------------

---

### Description

This track displays the coverage of the genomic region by the reads used as input for the segmentation step.

### Usage

```
CoverageTrack(ping, reads, chr, gen="gen", FragmentLength=200, name="XSET")
```

### Arguments

ping	An object of class <a href="#">pingList</a> . The output of PING.
reads	An object of class <a href="#">GRanges</a> . The reads used in the segmentation process.
chr	A character string for the chromosome name.
gen	An optional character string for the genome name.
FragmentLength	An integer. The fixed length of the reads. If PE is set to TRUE, this argument can be ignored.
name	A character string. The title of the track.

### Value

This function returns an object of class `AnnotationTrack`.

### Author(s)

Renan Sauteraud

### See Also

[postPING](#) [plotSummary](#) [AnnotationTrack](#) [plotTracks](#)

---

NucleosomeTrack	<i>Track for the Nucleosome position</i>
-----------------	------------------------------------------

---

### Description

This track actually creates two tracks: it displays the position of the predicted nucleosomes as well as the standard error associated with each prediction and an histogram of the score.

### Usage

```
NucleosomeTrack(PS, chr, gen="gen", scoreTrack=TRUE, scoreThreshold=0.05, name="PING", ...)
```

### Arguments

PS	A data.frame or <a href="#">pingList</a> . The output of postPING or PING.
chr	A character string for the chromosome name.
gen	An optional character string for the genome name.
scoreTrack	A logical. If set to FALSE, the track showing the histogram of the score is removed.
scoreThreshold	A numeric. Nucleosomes predicted with a score inferior to this threshold are removed.
name	A character string for the title of the track.
...	A list of supplemental argument that can be provided to the <a href="#">AnnotationTrack</a> track constructor.

### Value

This function returns a list containing an object of class `AnnotationTrack` and a `DataTrack` if `scoreTrack` is set to `TRUE`.

### Author(s)

Renan Sauteraud

### See Also

[postPING](#) [plotSummary](#) [DataTrack](#) [AnnotationTrack](#) [plotTracks](#)

---

PICS\_IMPORT

*PICS functions.*


---

### Description

Functions imported from PICS and exposed when loading PING. Refer to PICS documentation for further information.

?PICS::segReads

?PICS::segReadsList

?PICS::segReadsPE

?PICS::segReadsListPE

?PICS::setParaEM

?PICS::setParaPrior

?PICS::bam2gr

### See Also

[makeRangedDataOutput](#) [segReads](#) [segReadsList](#) [setParaEM](#) [setParaPrior](#) [segReadsPE](#) [segReadsListPE](#)

---

ping

*Estimation of binding site positions*


---

### Description

This object contains Estimation of binding site positions and has the following slots: `segReadsList`.

### Usage

```
PING(segReadsList, paraEM=NULL, paraPrior=NULL, dataType="MNase", detail=0, rescale=1, nCores=1)
```

### Arguments

`segReadsList` This object contains segmentation of Genome

`paraEM` A list of parameters for the EM algorithm. The default parameters should be good enough for most usages.

`minK`: an integer, default=0. The minimum number of binding events per region. If the value is 0, the minimum number is automatically calculated.

`maxK`: an integer, default=0. The maximum number of binding events per region. If the value is 0, the maximum number is automatically calculated.

`tol`: a numeric, default=1e-4. The tolerance for the EM algorithm.

`B`: an integer, default=100. The maximum number of iterations to be used.

	mSelect: a character string specifying the information criteria to be used when selecting the number of binding events. Default="AIC3"
	mergePeaks: a logical stating whether overlapping binding events should be picked. Default=TRUE
	mapCorrect: a logical stating whether mappability profiles should be incorporated in the estimation, i.e: missing reads estimated. Default=TRUE
paraPrior	A list of parameters for the prior distribution. The default parameters should be good enough for most usages. xi: an integer, default=150. The average DNA fragment size. rho: an integer, default=1.2. A variance parameter for the average DNA fragment size distribution. alpha: an integer, default=10. First hyperparameter of the inverse Gamma distribution for $\sigma^2$ in the PICS model beta: an integer, default=20000. Second hyperparameter of the inverse Gamma distribution for $\sigma^2$ in the PING model lambda: an integer, default=0.000064. The precision of the prior for $\mu$ used for histone data. dMu: an integer, default=200. Our best guess for the distance between two neighboring nucleosomes.
dataType	A character string that can be set to use selected default parameters for the algorithm.
detail	An integer. Additional information are printed if set to a value > 0.
rescale	An integer.
nCores	An integer. The number of cores that should be used in parallel by the function.

## Methods

- code** signature(x = "ping"): return the error code for each list element (i.e. candidate region) of a PING object. If the string is empty, there were no errors.
- plot** signature(x = "ping"): Plot all regions in the PING object. This might be long, and should only be used to plot a few regions, so subset the object before plotting.
- sigmaSqR** signature(x = "ping"): return the variance parameter of the reverse (R) distribution for each binding event.
- sigmaSqF** signature(x = "ping"): return the variance parameter of the forward (F) distribution for each binding event.
- score** signature(x = "ping"): return the score for each binding event.
- scoreF** signature(x = "ping"): return the score of the forward (F) for each binding event.
- scoreR** signature(x = "ping"): return the score of the forward (R) for each binding event.
- maxRange** signature(x = "ping"): return the range maximum.
- minRange** signature(x = "ping"): return the range minimal.
- K** signature(x = "ping"): subset PING object.
- density** signature(x = "ping"): return the density for each binding event.

**Author(s)**

Xuekui Zhang, <ubcxzhang@gmail.com> and Raphael Gottardo, <raphael.gottardo@ircm.qc.ca>  
Sangsoon Woo, <swoo@fhcrc.org>

**See Also**

[ping](#)

---

ping-class

*The ping class*

---

**Description**

This object is used to gather all parameters from fitting PING to a single candidate region. The object contains the following slots: 'estimates', 'infMat', 'Nmerged', 'converge', 'chr'. 'estimates' is a list containing all parameters estimates as well as standard errors. 'infMat' is the Cholesky decomposition of the information matrix, 'converge' is a logical value indicating whether the EM algorithm has converged, while 'chr' is a character string corresponding to a candidate region's chromosome. 'Nmerged' gives the number of binding events that were merged; binding events that overlap are merged (see the cited paper below for details).

**Accessors**

The PING package provide accessors to directly access to most of the parameters/standard errors and chromosome. In the code snippets below, 'x' is a 'ping' object.

**'chromosome(x)'** Gets the chromosome name of the candidate region.

**'mu(x)'** Gets the position estimates of all binding sites identified in the region.

**'delta(x)'** Gets the average fragment lengths of all binding sites identified in the region.

**'sigmaSqF(x)'** Gets the F peak variances of all binding sites identified in the region.

**'sigmaSqR(x)'** Gets the R peak variances of all binding sites identified in the region.

**'se(x)'** Gets the standard errors of all binding site position estimates identified in the region.

**'seF(x)'** Gets the standard errors of all F peak modes identified in the region.

**'seR(x)'** Gets the standard errors of all R peak modes identified in the region.

**score** signature(x = "ping"): return the score for each binding event.

**scoreF** signature(x = "ping"): return the score of the forward (F) for each binding event.

**scoreR** signature(x = "ping"): return the score of the forward (R) for each binding event.

**Constructor**

`newPing(w,mu,delta,sigmaSqF,sigmaSqR,seMu,seMuF,seMuR,score,Nmerged,converge,infMat,chr)`  
construct a new 'ping' object with the following arguments:

**w** The mixture weights (a vector)

**mu** The binding site positions (a vector)

**delta** The DNA fragment lengths (a vector)

**sigmaSqF** The variance parameters for the forward distribution (vector)

**sigmaSqR** The variance parameters for the reverse distribution (vector)

**seMu** The standard errors for mu (vector)

**seMuF** The standard errors for muF (vector)

**seMuR** The standard errors for muR (vector)

**score** The scores for each binding event (vector)

**Nmerged** The number of peaks that got merged (integer)

**converge** A logical value, TRUE, if the EM algorithm converged

**infMat** The information matrix

**chr** The chromosome for the region

**Author(s)**

Xuekui Zhang <<xzhang@stat.ubc.ca>>, Sangsoon Woo, <swoo@fhcrc.org> and Raphael Gottardo <<raphael.gottardo@ircm.qc.ca>>

**References**

Xuekui Zhang, Gordon Robertson, Sangsoon Woo, Brad G. Hoffman, and Raphael Gottardo, "Probabilistic Inference for Nucleosome Positioning with MNase-based or Sonicated Short-read Data" *Genome Biology*, under review.

**See Also**

[ping pingError](#)

**Examples**

```
# Here is an example of how to construct such a region.  
# Typically, you would not do this manually, you would use the ping function to return a 'pingList'  
# that contains a list of 'ping' or a 'pingError' object.  
w<-1  
mu<-10000  
delta<-150  
sigmaSqF<-5000  
sigmaSqR<-5000  
seMu<-10  
seMuF<-10  
seMuR<-10
```

```

score<-5
Nmerged<-0
converge<-TRUE
chr<-"chr1"
range<-c(1000,2000)
# Constructor
#myPING<-newPing(w,mu,delta,sigmaSqF,sigmaSqR,seMu,seMuF,seMuR,score,Nmerged,as.integer(range),chr)

```

---

pingError-class

*The ping class*


---

### Description

This object is used to return an error code when the PING function failed to return a valid set of estimates for a candidate regions. This could be due to non-convergence of the EM algorithm, a singular information matrix, or a number of reads below the limit specified by the user. All of these are typically due to too few reads in the region and do not affect the rest of the analysis, as such regions would most likely be labelled as false positives.

### Accessors

All of the accessors defined for a 'ping' object still work for a 'pingError' object but will simply return a NULL pointer.

### Constructor

`newPingError(string)` where 'string' is the error code.

### Constructor

`newPingError<-function(string)`

**string** The mixture weights (a vector)

### Author(s)

Xuekui Zhang <<xzhang@stat.ubc.ca>>, Sangsoon Woo, <swoo@fhcrc.org> and Raphael Gottardo <<raphael.gottardo@ircm.qc.ca>>

### References

Xuekui Zhang, Gordon Robertson, Sangsoon Woo, Brad G. Hoffman, and Raphael Gottardo, "Probabilistic Inference for Nucleosome Positioning with MNase-based or Sonicated Short-read Data" *GenomeBiology*, under review.

### See Also

[ping](#)



**Examples**

```
# Here is an example on how to construct such a pingError object
# Typically, you would not do this manually, you would use the ping function to return a 'pingList'
# that contains a list of 'ping' or 'pingError' object.
# Constructor
myPingError<-newPingError("Singular information matrix")
# Accessors
# Get the standard error of Mu
se(myPingError)
# Get the standard error of MuF
seF(myPingError)
# Get the scores
score(myPingError)
```

pingList-class

*The ping class***Description**

This object is used to gather all parameters from fitting PING to multiple candidate regions (as returned by the 'segmentReads' function). The object contains the following slots: 'List', 'paraPrior', 'paraEM', 'minReads', 'N', 'Nc'. 'List' is a list of 'ping' or 'pingError' objects. 'paraPrior' is a list containing the hyperparameters used for the prior, 'paraEM' is a list of convergence parameters for the EM, 'minReads' is a list containing the minimum number of reads used to fit a region with 'PING', 'N' is the total number of reads in the ChIP samples while 'Nc' is the total number of reads in the control sample.

**Arguments**

object            An object of class ping.

**Accessors**

The PING package provide accessors to directly access to most of the parameters/standard errors and chromosomes. In the code snippets below, 'x' is a 'pingList' object. For all accessors, the 'pingError' objects are omitted, so that the accessors only return values for the 'ping' objects (i.e. all valid binding events).

'**chromosome(x)**' Gets the chromosome names of all candidate regions.

'**mu(x)**' Gets the position estimates of all binding sites identified in all candidate regions.

'**delta(x)**' Gets the average fragment lengths of all binding sites identified in all candidate regions.

'**sigmaSqF(x)**' Gets the F peak variances of all binding sites identified in all candidate regions.

'**sigmaSqR(x)**' Gets the R peak variances of all binding sites identified in all candidate regions.

'**seF(x)**' Gets the standard errors of all binding site position estimates identified in all candidate regions.

'**seF(x)**' Gets the standard errors of all F peak modes identified in all candidate regions.

'**seR(x)**' Gets the standard errors of all R peak modes identified in all candidate regions.

'**score(x)**' Gets the scores of all binding events identified in all candidate regions.

**Constructor**

`newPingList(List, paraEM, paraPrior, minReads, N, Nc)`

**List** The mixture weights (a vector)

**paraEM** The binding site positions (a vector)

**paraPrior** The DNA fragment lengths (a vector)

**N** The variance parameters for the forward distribution (vector)

**Nc** The variance parameters for the forward distribution (vector)

**Methods**

[ `signature(x = "ping")`]: subset PING object.

**Methods**

**length** `signature(x = "ping")`: subset PING object.

**as.data.frame** `signature(x = "pingList")`: Coerce a `pingList` to a `data.frame`.

**Constructor**

`newPingList<-function(List, paraEM, paraPrior, minReads, N, Nc)` constructs a new 'pingList' object with the following arguments.

**newPingList**

**w** The mixture weights (a vector)

**mu** The binding site positions (a vector)

**delta** The DNA fragment lengths (a vector)

**sigmaSqF** The variance parameters for the forward distribution (vector)

**sigmaSqR** The variance parameters for the reverse distribution (vector)

**seMu** The standard errors for mu (vector)

**seMuF** The standard errors for muF (vector)

**seMuR** The standard errors for muR (vector)

**seMuR** The standard errors for muR (vector)

**score** The scores for each binding event (vector)

**Nmerged** The number of peaks that were merged (integer)

**converge** A logical value, TRUE, if the EM as converged

**infMat** The information matrix

**chr** The chromosome for the region

**Author(s)**

Xuekui Zhang <<xzhang@stat.ubc.ca>>, Sangsoon Woo, <swoo@fhcrc.org> and Raphael Gottardo <<raphael.gottardo@ircm.qc.ca>>

## References

Xuekui Zhang, Gordon Robertson, Sangsoon Woo, Brad G. Hoffman, and Raphael Gottardo, "Probabilistic Inference for Nucleosome Positioning with MNase-based or Sonicated Short-read Data" PlosONE, under review.

## See Also

[ping](#)

## Examples

```
# Here is an example of how to construct such a region
# Typically, you would not do this manually, you would use the ping function to return a 'pingList'
# that contains a list of 'ping' or 'pingError' object.
w<-1
mu<-10000
delta<-150
sigmaSqF<-5000
sigmaSqR<-5000
seMu<-10
seMuF<-10
seMuR<-10
score<-5
Nmerged<-0
converge<-TRUE
infMat<-matrix(0)
chr<-"chr1"
range<-c(1000,2000)
# Constructor
#myPING1<-newPing(w,mu,delta,sigmaSqF,sigmaSqR,seMu,seMuF,seMuR,score,Nmerged,converge,infMat,as.integer(range))
#myPING2<-newPing(w,mu+1000,delta,sigmaSqF,sigmaSqR,seMu,seMuF,seMuR,score,Nmerged,converge,infMat,as.integer(range))

#minReads<-list(perPeak=2,perRegion=5)
#paraPrior<-list(xi=200,rho=1,alpha=20,beta=40000)
#paraEM<-list(minK=1,maxK=15,tol=10e-6,B=100)
#N<-100
#Nc<-200

#mynewPingList<-newPingList(list(myPING1,myPING2), paraEM, paraPrior, minReads, as.integer(100), as.integer(200))
# Accessors
# Get the standard error of Mu
#se(mynewPingList)
# Get the standard error of MuF
#seF(mynewPingList)
# Get the scores
#score(mynewPingList)
```

---

plotSummary

*Plot a summary of the prediction for given ranges.*


---

### Description

This function use Gviz package to summarize the nucleosome position prediction from postPING.

### Usage

```
plotSummary(PS, ping, reads, chr, gen="gen", from=NULL, to=NULL, FragmentLength=200, title="", scoreT
```

### Arguments

PS	A data.frame. The output postPING. If a list of data.frame is passed, multiple NucleosomeTrack will be plotted.
ping	A <a href="#">pingList</a> . The output of PING.
reads	A GRanges object. The reads used in the segmentation process.
chr	A character string for the chromosome name.
gen	An optional character string for the genome name.
from	An integer, the first base of the plot
to	An integer, the last base of the plot
FragmentLength	An integer, the length of XSET profile extension in the CoverageTrack. This argument will be ignored if the reads are paired-end sequencing data.
title	An optional character string that can be prepend to the automatically generated title.
scoreThreshold	A numeric. Removes nucleosome with a score inferior to the threshold from the plot.

### Value

This function returns an invisible list of tracks. The tracks can be used in other Gviz plots.

### Author(s)

Renan Sauteraud

### See Also

[CoverageTrack](#) [RawReadsTrack](#) [NucleosomeTrack](#) [plotTracks](#)

---

postPING *Post process Estimation of binding site positions obtained from PING*

---

### Description

Post process Estimation of binding site positions obtained from PING. Refit mixture models with stronger prior in candidate regions contain potential problems, and then convert final result into dataframe.

### Usage

```
postPING(ping, seg, rho2=NULL, sigmaB2=NULL, alpha2=NULL, beta2=NULL, min.dist= 100, paraEM=NULL, par
```

### Arguments

ping	A 'pingList' object containing estimation of nucleosome positions, result of 'PING' function.
seg	An object of class 'segmentReadsList' containing the results for all regions pre-processed, 'segmentReads' function.
paraEM	A list of parameters for the EM algorithm. The default parameters should be good enough for most usages. minK: An integer, default=0. The minimum number of binding events per region. If the value is 0, the minimum number is automatically calculated. maxK: An integer, default=0. The maximum number of binding events per region. If the value is 0, the maximum number is automatically calculated. tol: A numeric, default=1e-4. The tolerance for the EM algorithm. B: An integer, default=100. The maximum number of iterations to be used. mSelect: A character string specifying the information criteria to be used when selecting the number of binding events. Default="AIC3" mergePeaks: A logical stating whether overlapping binding events should be picked. Default=TRUE mapCorrect: A logical stating whether mappability profiles should be incorporated in the estimation, i.e: missing reads estimated. Default=TRUE
paraPrior	A list of parameters for the prior distribution. The default parameters should be good enough for most usages. xi: An integer. The average DNA fragment size. rho: An integer. A variance parameter for the average DNA fragment size distribution. alpha: An integer. First hyperparameter of the inverse Gamma distribution for $\sigma^2$ in the PICS model beta: An integer. Second hyperparameter of the inverse Gamma distribution for $\sigma^2$ in the PING model lambda: An integer. The lambda control Gaussian Markov Random Field prior on the distance of adjacent nucleosomes, we do not recommend user change the default value.

	dMu: An integer. Our best guess for the distance between two neighboring nucleosomes.
rho2, sigmaB2, alpha2, beta2	Integer values, the parameters in the prior of mixture models to be re-fitted.
min.dist	The minimum distance of two adjacent nucleosomes predicted from different candidate regions, smaller than that will be treated as duplicated predictions for the same nucleosomes.
score	A numeric. The score threshold used when calling FilterPING.
dataType	A character string that can be set to use selected default parameters for the algorithm.
nCores	An integer. The number of cores that should be used in parallel by the function.
makePlot	A logical. Plot a summary of the output.
FragmentLength, mart, seg.boundary, DupBound, IP, datname	Plotting parameters and options.
	IP: A GRanges object. The reads used in segmentation process.
	FragmentLength: An integer. The length of XSET profile extension

**Value**

A data.frame containing the estimation of binding site positions.

**Note**

Based on our experient on a few real data sets, we suggestion to use following values of parameters. For sonication data we use rho1=1.2; sigmaB2=6400;rho=15;alpha1=10; alpha2=98; beta2=200000. For MNase data we use rho1=3; sigmaB2=4900; rho=8; alpha1=20; alpha2=100; beta2=100000. The value of xi depends on specy of sample, since that affect the length of linker-DNA. For example, we use xi=160 for yeast and xi=200 for mouse.

**Author(s)**

Xuekui Zhang <<xzhang@stat.ubc.ca>>, Sangsoon Woo, <swoo@fhcrc.org> and Raphael Gottardo <<raphael.gottardo@ircm.qc.ca>>

**References**

Xuekui Zhang, Gordon Robertson, Sangsoon Woo, Brad G. Hoffman, and Raphael Gottardo, "Probabilistic Inference for Nucleosome Positioning with MNase-based or Sonicated Short-read Data" PlosONE, under review.

**See Also**

[PING plotSummary](#)

---

RawReadsTrack	<i>Raw reads density track</i>
---------------	--------------------------------

---

### Description

This track displays the start position of the forward and reverse reads used as input for the segmentation step

### Usage

```
RawReadsTrack(ping, reads, chr, gen="gen", from=NULL, to=NULL, ...)
```

### Arguments

ping	A <a href="#">pingList</a> . The output of PING.
reads	An object of class <a href="#">GRanges</a> . The reads used in the segmentation process.
chr	A character string for the chromosome name.
gen	An optional character string for the genome name.
from, to	Two numeric. A subset of the reads can be selected in order to speed up the computation.
...	Optional arguments ( <a href="#">DisplayParameters</a> ) that can be passed on to the <a href="#">DataTrack</a> . Refer to <a href="#">Gviz help</a> and <a href="#">vignette</a> for a list of arguments.

### Value

This function returns an object of class [DataTrack](#).

### Author(s)

Renan Sauteraud

### See Also

[segmentPING](#) [plotSummary](#) [DataTrack](#) [plotTracks](#)

segmentPING

*Segment the genome into candidate regions***Description**

Pre-process bidirectional aligned reads data to detect candidate regions with a minimum number of forward and reverse reads. These candidate regions will then be processed by PING.

**Usage**

```
segmentPING(data, dataC=NULL, map=NULL, minReads=2, minReadsInRegion=3,
            jitter=FALSE, maxLregion=1200, minLregion=80, step=NULL, width=NULL,
            islandDepth=3, min_cut=50, max_cut=1000, maxReadsWidth=500, PE=FALSE)
```

**Arguments**

data	A <a href="#">GRanges</a> object containing the IP reads. See details for more information on how to set up the data.
dataC	A <a href="#">GRanges</a> object containing the control reads. Set to NULL by default, i.e. no control.
map	A ‘RangedData’ object containing the mappability profiles. Set to NULL by default, i.e. no profiles.
minReads	The minimum number of F/R reads to be present in the sliding window.
minReadsInRegion	The minimum number of F/R reads to be present in the region.
jitter	A logical value stating whether some noise should be added to the read locations. This is recommended if the read positions have lots of duplicates.
step	An integer. The step size for the sliding window.
width	An integer. The size of the sliding window.
maxLregion	The maximum length.
minLregion	The minimum length.
PE	A logical. Set to TRUE for Paired-End sequencing data.
islandDepth, min_cut, max_cut, maxReadsWidth	Parameters used for paired-end sequencing data segmentation.
	<b>islandDepth</b> An integer. The minimum number of reads to cover a candidate region.
	<b>min_cut</b> An integer. The minimum length of a candidate region.
	<b>max_cut</b> An integer. The maximum length of a candidate region.
	<b>maxReadsWidth</b> An integer. Reads with width superior to this limit will be removed from the data.

**Value**

An object of class [segReadsList](#) containing the results for all regions pre-processed.



**Author(s)**

Xuekui Zhang, Arnaud Droit <<arnaud.droit@crchuq.ualaval.ca>> and Raphael Gottardo <<rgottard@fhcrc.org>>

**References**

Xuekui Zhang, Gordon Robertson, Sangsoon Woo, Brad G. Hoffman, and Raphael Gottardo, "Probabilistic Inference for Nucleosome Positioning with MNase-based or Sonicated Short-read Data" PlosONE.

**See Also**

[segReadsList](#)

**Examples**

```
# Read data
path<-system.file("extdata",package="PING")
dataIP<-read.table(file.path(path,"GSM351492_R4_chr1.bed"),header=TRUE)
dataIP<-as(dataIP,"GRanges")
seg<-segmentPING(dataIP, minReads=NULL, maxLregion=1200,minLregion=80, jitter=TRUE)
```

---

show

*show PING*

---

**Description**

This methods show the objects of PING

**Usage**

```
## S4 method for signature 'ping'
show(object)
## S4 method for signature 'pingError'
show(object)
## S4 method for signature 'pingList'
show(object)
## S4 method for signature 'segReads'
show(object)
## S4 method for signature 'segReadsList'
show(object)
```

**Arguments**

object            Object returned from [ping](#) .

**Details**

List of the slots include in the object

**Author(s)**

Xuekui Zhang <<xzhang@stat.ubc.ca> Sangsoon Woo, <swoo@fhcrc.org> Raphael Gottardo <<raphael.gottardo@ircm.qc.ca>

**See Also**

[summary](#)

---

summary

*summary PING*

---

**Description**

This methods summary the objects of PING.

**Usage**

```
## S4 method for signature 'ping'  
summary(object)  
## S4 method for signature 'pingList'  
summary(object)  
## S4 method for signature 'segReads'  
summary(object)  
## S4 method for signature 'segReadsList'  
summary(object)
```

**Arguments**

object            Object returned from [ping](#) .

**Author(s)**

Xuekui Zhang <<xzhang@stat.ubc.ca> Sangsoon Woo, <swoo@fhcrc.org> Raphael Gottardo <<raphael.gottardo@ircm.qc.ca>

**See Also**

[show](#)

# Index

- \*Topic **data**
  - postPING, 13
- \*Topic **functions**
  - CoverageTrack, 2
  - NucleosomeTrack, 3
  - plotSummary, 12
  - RawReadsTrack, 15
  - segmentPING, 16
- \*Topic **models**
  - ping, 4
  - ping-class, 6
  - pingError-class, 8
  - pingList-class, 9
  - postPING, 13
- \*Topic **print**
  - show, 17
  - summary, 18
- [, pingList, ANY, ANY-method (pingList-class), 9
- [[, pingList, ANY, ANY-method (pingList-class), 9
  
- AnnotationTrack, 2, 3
- as.data.frame, pingList-method (pingList-class), 9
  
- bam2gr (PICS\_IMPORT), 4
  
- chromosome (ping-class), 6
- chromosome, ping-method (ping-class), 6
- chromosome, pingError-method (pingError-class), 8
- chromosome, pingList-method (pingList-class), 9
- code (ping), 4
- code, ping-method (ping), 4
- code, pingError-method (ping), 4
- code, pingList-method (ping), 4
- coerce (ping), 4
- coerce, data.frame, pingList-method (ping), 4
- coerce, pingList, data.frame-method (ping), 4
- coerce, pingList, RangedData-method (ping), 4
- CoverageTrack, 2, 12
  
- DataTrack, 3, 15
- delta (ping-class), 6
- delta, ping-method (ping-class), 6
- delta, pingError-method (pingError-class), 8
- delta, pingList-method (pingList-class), 9
- density (ping), 4
- density, ping-method (ping), 4
- density, pingError-method (ping), 4
- density, pingList-method (pingList-class), 9
  
- GRanges, 2, 15, 16
  
- K (ping), 4
- K, ping-method (ping), 4
- K, pingError-method (ping), 4
- K, pingList-method (ping), 4
  
- length (pingList-class), 9
- length, pingList-method (pingList-class), 9
  
- makeRangedDataOutput, 4
- makeRangedDataOutput (PICS\_IMPORT), 4
- maxRange (ping), 4
- maxRange, ping-method (ping), 4
- maxRange, pingError-method (ping), 4
- maxRange, pingList-method (ping), 4
- minRange (ping), 4
- minRange, ping-method (ping), 4
- minRange, pingError-method (ping), 4

- minRange, pingList-method (ping), 4
- mu (ping-class), 6
- mu, ping-method (ping-class), 6
- mu, pingError-method (pingError-class), 8
- mu, pingList-method (pingList-class), 9
  
- newPing (ping-class), 6
- newPing, ping-method (ping-class), 6
- newPingError (pingError-class), 8
- newPingError, pingError-method (pingError-class), 8
- newPingList (pingList-class), 9
- newPingList, pingList-method (pingList-class), 9
- NucleosomeTrack, 3, 12
  
- PICS\_IMPORT, 4
- PING, 14
- PING (ping), 4
- ping, 4, 6–8, 11, 17, 18
- ping-class, 6
- ping-class (ping), 4
- pingError, 7
- pingError (pingError-class), 8
- pingError-class, 8
- pingList, 2, 3, 12, 15
- pingList (pingList-class), 9
- pingList-class, 9
- plot (ping), 4
- plot, data.frame, data.frame-method (ping), 4
- plot, ping, segReads-method (ping), 4
- plot, pingError, segReads-method (ping), 4
- plot, pingList, pingList-method (ping), 4
- plot, pingList, segReadsList-method (ping), 4
- plotSummary, 2, 3, 12, 14, 15
- plotTracks, 2, 3, 12, 15
- postPING, 2, 3, 13
  
- RawReadsTrack, 12, 15
  
- score (ping), 4
- score, ping-method (ping-class), 6
- score, pingError-method (pingError-class), 8
- score, pingList-method (pingList-class), 9
- scoreForward (ping), 4
- scoreForward, ping-method (ping), 4
- scoreForward, pingError-method (ping), 4
- scoreForward, pingList-method (ping), 4
- scoreReverse (ping), 4
- scoreReverse, ping-method (ping), 4
- scoreReverse, pingError-method (ping), 4
- scoreReverse, pingList-method (ping), 4
- se (ping-class), 6
- se, ping-method (ping-class), 6
- se, pingError-method (pingError-class), 8
- se, pingList-method (pingList-class), 9
- seF (ping-class), 6
- seF, ping-method (ping-class), 6
- seF, pingError-method (pingError-class), 8
- seF, pingList-method (pingList-class), 9
- segmentPING, 15, 16
- segReads, 4
- segReads (PICS\_IMPORT), 4
- segReadsList, 4, 16, 17
- segReadsList (PICS\_IMPORT), 4
- segReadsListPE, 4
- segReadsListPE (PICS\_IMPORT), 4
- segReadsPE, 4
- segReadsPE (PICS\_IMPORT), 4
- seR (ping-class), 6
- seR, ping-method (ping-class), 6
- seR, pingError-method (pingError-class), 8
- seR, pingList-method (pingList-class), 9
- setParaEM, 4
- setParaEM (PICS\_IMPORT), 4
- setParaPrior, 4
- setParaPrior (PICS\_IMPORT), 4
- show, 17, 18
- show, ping-method (show), 17
- show, pingError-method (show), 17
- show, pingList-method (show), 17
- show, segReads-method (show), 17
- show, segReadsList-method (show), 17
- show-methods (show), 17
- sigmaSqF (ping), 4
- sigmaSqF, ping-method (ping), 4
- sigmaSqF, pingError-method (ping), 4
- sigmaSqF, pingList-method (ping), 4
- sigmaSqR (ping), 4
- sigmaSqR, ping-method (ping), 4
- sigmaSqR, pingError-method (ping), 4

sigmaSqR, pingList-method (ping), 4  
summary, [18](#), [18](#)  
summary, ping-method (summary), [18](#)  
summary, pingList-method (summary), [18](#)  
summary, segReads-method (summary), [18](#)  
summary, segReadsList-method (summary),  
[18](#)  
summary-methods (summary), [18](#)

w (ping-class), [6](#)  
w, ping-method (ping-class), [6](#)  
w, pingError-method (pingError-class), [8](#)  
w, pingList-method (pingList-class), [9](#)