

# Package ‘PICS’

April 22, 2019

**Type** Package

**Title** Probabilistic inference of ChIP-seq

**Version** 2.27.0

**Author** Xuekui Zhang <xzhang@stat.ubc.ca>, Raphael Gottardo  
<rgottard@fhcrc.org>

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

**Depends** R (>= 2.14.0), BiocGenerics (>= 0.1.3)

**Imports** methods, stats4, IRanges, GenomicRanges, graphics, grDevices,  
stats, Rsamtools, GenomicAlignments, S4Vectors

**Suggests** ShortRead, rtracklayer, parallel

**SystemRequirements** GSL (GNU Scientific Library)

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

**License** Artistic-2.0

**biocViews** Clustering, Visualization, Sequencing, ChIPSeq

**git\_url** <https://git.bioconductor.org/packages/PICS>

**git\_branch** master

**git\_last\_commit** c3b77e4

**git\_last\_commit\_date** 2018-10-30

**Date/Publication** 2019-04-21

## R topics documented:

bam2gr	2
makeRangedDataOutput	3
pics	4
pics-class	6
picsError-class	7
picsFDR	8
picsList-class	9
plot-FDR	11
segChrRead	12
segmentPICS	12
segReads	14

segReadsList . . . . .	14
segReadsListPE . . . . .	15
segReadsPE . . . . .	15
setParaEM . . . . .	16
setParaPrior . . . . .	17
show . . . . .	18
summary . . . . .	18
summarySeg . . . . .	19

<b>Index</b>	<b>20</b>
--------------	-----------

---

bam2gr	<i>pre-process bam files</i>
--------	------------------------------

---

## Description

Reads a bam file using Rsamtools and extract the reads for each chromosome.

## Usage

```
bam2gr(bamFile, chr=NULL, PE=FALSE, verbose=FALSE)
```

## Arguments

bamFile	A character string, the name of the .bam file to read.
chr	An optional character string. If specified, only the selected chromosome will be returned. Speed up the computation.
PE	A logical. This should be set to TRUE for paired-end sequencing data.
verbose	A logical. Print additional information about the data.

## Value

Returns a GRanges of all the reads for each chromosome.

## Note

The user might encounter a memory allocation error when using bam files of bigger sizes. Splitting the file by chromosome before calling bam2gr will solve this issue.

For Paired-End data, non matched reads are discarded.

## Author(s)

Renan Sauteraud

## See Also

[segmentPICS](#)

---

makeRangedDataOutput *Create a RangedData object from a PICS output*

---

### Description

Create a list of 'RangedData' objects from a 'pics' object. The resulting RangedData object can then be analyzed with the 'IRanges' packages and/or exported to bed/wig files with the 'rtracklayer' package.

### Usage

```
makeRangedDataOutput(obj, type="fixed", filter=list(delta=c(0,Inf),se=c(0,Inf),sigmaSqF=c(0,Inf))
```

### Arguments

obj	An object of class 'picsList' as returned by 'PICS' when running it on the IP/Control data.
type	The type of intervals to be created. The different types are 'bed', 'wig', 'ci' and 'fixed'. See details for more info.
filter	A list of filters to be used before computing the FDR. By default all regions are included, see details for more info on how to specify the filters.
length	The length to be used for the fixed type 'RangedData', see details.

### Details

'bed' will generate intervals from the forward peak max to the reverse peak max. 'wig' will generate a density profile for the forward and reverse reads. 'bed' and 'wig' types should be used to be exported to wig/bed files to be used with the UCSC genome browser. 'ci' corresponds to the binding site estimates  $\pm 3 \cdot se$ , while 'fixed' corresponds to the binding site estimates  $\pm 3 \cdot length$ . 'bed' and 'wig' files can be exported using the 'export' function fo the 'rtracklayer' package.

### Value

An object of type 'RangedData'.

### Author(s)

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

### References

X. Zhang, G. Robertson, M. Krzywinski, K. Ning, A. Droit, S. Jones, and R. Gottardo, "PICS: Probabilistic Inference for ChIP-seq" arXiv, 0903.3206, 2009. To appear in Biometrics.

### See Also

export

## Examples

```
## Not run:
rdBed<-makeRangedDataOutput(pics,type="bed",filter=list(delta=c(50,Inf),se=c(0,50),sigmaSqF=c(0,22500),si
export(rbBed,"myfile.bed")
rdBed<-makeRangedDataOutput(pics,type="wig",filter=list(delta=c(50,Inf),se=c(0,50),sigmaSqF=c(0,22500),si
export(rbBed,"myfile.wig")
## End(Not run)
```

---

pics

*Estimation of binding site positions*

---

## Description

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

## Usage

```
PICS(segReadsList,dataType=NULL, paraEM=NULL, paraPrior=NULL, nCores=1)
```

## Arguments

<code>segReadsList</code>	This object contains segmentation of Genome
<code>dataType</code>	The type of data you are processing: specified 'TF' for transcription factor.
<code>paraEM</code>	A list of parameters for the EM algorithm as returned by the <code>setParaEm</code> function. The default parameters should be good enough for most usages. <ul style="list-style-type: none"> <li><code>minK</code>: an integer, default=1. The minimum number of binding events per region. If the value is 0, the minimum number is automatically calculated.</li> <li><code>maxK</code>: an integer, default=15. The maximum number of binding events per region. If the value is 0, the maximum number is automatically calculated.</li> <li><code>tol</code>: a numeric, default=1e-4. The tolerance for the EM algorithm.</li> <li><code>B</code>: an integer, default=100. The maximum number of iterations to be used.</li> <li><code>mSelect</code>: a character string specifying the information criteria to be used when selecting the number of binding events. Default="BIC"</li> <li><code>mergePeaks</code>: a logical stating whether overlapping binding events should be picked. Default=TRUE</li> <li><code>mapCorrect</code>: a logical stating whether mappability profiles should be incorporated in the estimation, i.e: missing reads estimated. Default=TRUE</li> </ul>
<code>paraPrior</code>	A list of parameters for the prior distribution as returned by the <code>setParaPrior</code> function. The default parameters should be good enough for most usages. <ul style="list-style-type: none"> <li><code>xi</code>: an integer, default=200. The average DNA fragment size.</li> <li><code>rho</code>: an integer, default=1. A variance parameter for the average DNA fragment size distribution.</li> <li><code>alpha</code>: an integer, default=20. First hyperparameter of the inverse Gamma distribution for <math>\sigma^2</math> in the PICS model</li> </ul>

	beta: an integer, default=40000. Second hyperparameter of the inverse Gamma distribution for $\sigma^2$ in the PING model
	lambda: an integer, default=0. The precision of the prior for $\mu$ used for histone data.
	dMu: an integer, default=0. Our best guess for the distance between two neighboring nucleosomes.
nCores	An integer. The number of cores that should be used in parallel by the function.

## Methods

**code** signature( $x = \text{\`pics}'$ ): return the error code for each list element (i.e. candidate region) of a PICS object. If the string is empty, there were no errors.

**plot** signature( $x = \text{\`pics}'$ ): Plot all regions in the PICS object. This might be long, and should only be used to plot a few regions, so subset the object before plotting.

**sigmaSqR** signature( $x = \text{\`pics}'$ ): return the variance parameter of the reverse (R) distribution for each binding event.

**sigmaSqF** signature( $x = \text{\`pics}'$ ): return the variance parameter of the forward (F) distribution for each binding event.

**score** signature( $x = \text{\`pics}'$ ): return the score for each binding event.

**scoreF** signature( $x = \text{\`pics}'$ ): return the score of the forward (F) for each binding event.

**scoreR** signature( $x = \text{\`pics}'$ ): return the score of the forward (R) for each binding event.

**maxRange** signature( $x = \text{\`pics}'$ ): return the range maximum.

**minRange** signature( $x = \text{\`pics}'$ ): return the range minimal.

**K** signature( $x = \text{\`pics}'$ ): subset PICS object.

**wigDensity** signature( $x = \text{\`pics}'$ ): return the density for each binding event.

## Author(s)

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

## References

X. Zhang, G. Robertson, M. Krzywinski, K. Ning, A. Droit, S. Jones, and R. Gottardo, "PICS: Probabilistic Inference for ChIP-seq" arXiv, 0903.3206, 2009. To appear in Biometrics.

## See Also

[pics](#)

**Description**

This object is used to gather all parameters from fitting PICS 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 PICS package provide accessors to directly access to most of the parameters/standard errors and chromosome. In the code snippets below, 'x' is a 'pics' 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.

**'seF(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 = "pics"): return the score for each binding event.

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

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

**Constructor**

`newPics(w,mu,delta,sigmaSqF,sigmaSqR,seMu,seMuF,seMuR,score,Nmerged,converge,infMat,chr)`  
construct a new 'pics' 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 forward 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 got 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, Arnaud Droit <<arnaud.droit@crchuq.ualaval.ca>> and Raphael Gottardo <<rgottard@fhcrc.org>>

**References**

X. Zhang, G. Robertson, M. Krzywinski, K. Ning, A. Droit, S. Jones, and R. Gottardo, "PICS: Probabilistic Inference for ChIP-seq" arXiv, 0903.3206, 2009. To appear in Biometrics.

**See Also**

[pics](#) [picsError](#)

**Examples**

```
# Here is an example of how to construct such a region.
# Typically, you would not do this manually, you would use the pics function to return a 'picsList' that contains
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
#myPICS<-newPics(w,mu,delta,sigmaSqF,sigmaSqR,seMu,seMuF,seMuR,score,Nmerged,as.integer(range),chr)
```

---

picsError-class

*The pics class*

---

**Description**

This object is used to return an error code when the PICS 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 'pics' object still work for a 'picsError' object but will simply return a NULL pointer.

**Constructor**

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

**Constructor**

```
newPicsError<-function(string)
```

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

**Author(s)**

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

**References**

X. Zhang, G. Robertson, M. Krzywinski, K. Ning, A. Droit, S. Jones, and R. Gottardo, “PICS: Probabilistic Inference for ChIP-seq” arXiv, 0903.3206, 2009. To appear in Biometrics.

**See Also**

[pics](#)

**Examples**

```
# Here is an example on how to construct such a picsError object
# Typically, you would not do this manually, you would use the pics function to return a 'picsList' that contains
# Constructor
myPicsError<-newPicsError("Singular information matrix")
# Accessors
# Get the standard error of Mu
se(myPicsError)
# Get the standard error of MuF
seF(myPicsError)
# Get the scores
score(myPicsError)
```

---

picsFDR

*Estimate the FDR.*

---

**Description**

Estimate the false detection rate for an object of class `pics` or `picsList`.

**Usage**

```
picsFDR(picsIP,picsCont,filter=list(delta=c(0,Inf),se=c(0,Inf),sigmaSqF=c(0,Inf),sigmaSqR=c(0,Inf))
```

**Arguments**

<code>picsIP</code>	An object of class <code>pics</code> or <code>picsList</code> containing the informations for the IP reads.
<code>picsCont</code>	An object of class <code>pics</code> or <code>picsList</code> containing the informations for the control reads.
<code>filter</code>	filterA list of ranges for filtering regions based on PICS parameters. By default filter is set to 'NULL' and all regions are used. <b>delta</b> Length of the binding sites.

**se** Standard error.  
**sigmaSqF** Forward peak variance  
**sigmaSqR** Reverse peak variance

### Value

A 3 columns data.frame with the following columns: FDR, score, N.

### Author(s)

Xuekui Zhang

### See Also

[picsList](#) [pics](#)

---

`picsList-class`      *The pics class*

---

### Description

This object is used to gather all parameters from fitting PICS 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 'pics' or 'picsError' 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 'PICS', '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 `pics`.

### Accessors

The PICS package provide accessors to directly access to most of the parameters/standard errors and chromosomes. In the code snippets below, 'x' is a 'picsList' object. For all accessors, the 'picsError' objects are omitted, so that the accessors only return values for the 'pics' 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**

`newPicsList(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 = ``pics``)`: subset PICS object.

**Methods**

**length** `signature(x = ``pics``)`: subset PICS object.

**Constructor**

`newPicsList<-function(List, paraEM, paraPrior, minReads, N, Nc)` constructs a new ‘picsList’ object with the following arguments.

**newPicsList**

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

**References**

X. Zhang, G. Robertson, M. Krzywinski, K. Ning, A. Droit, S. Jones, and R. Gottardo, “PICS: Probabilistic Inference for ChIP-seq” arXiv, 0903.3206, 2009. To appear in Biometrics.

**See Also**[pics](#)**Examples**

```

# Here is an example of how to construct such a region
# Typically, you would not do this manually, you would use the pics function to return a 'picsList' that contains
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
#myPICS1<-newPics(w,mu,delta,sigmaSqF,sigmaSqR,seMu,seMuF,seMuR,score,Nmerged,converge,infMat,as.integer(r
#myPICS2<-newPics(w,mu+1000,delta,sigmaSqF,sigmaSqR,seMu,seMuF,seMuR,score,Nmerged,converge,infMat,as.inte

#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

#mynewPicsList<-newPicsList(list(myPICS1,myPICS2), paraEM, paraPrior, minReads, as.integer(100), as.integer(
# Accessors
# Get the standard error of Mu
#se(mynewPicsList)
# Get the standard error of MuF
#seF(mynewPicsList)
# Get the scores
#score(mynewPicsList)

```

plot-FDR

*FDR plot for PICS***Description**

This method plots an FDR curve showing the FDR as a function of the PICS scores.

**Usage**

```

## S4 method for signature 'picsList,picsList'
plot(x, y, filter=NULL, h=.1, ...)

```

**Arguments**

x	A <code>picList</code> object as returned by the function <code>PICS</code> run on the treatment data.
y	A <code>picList</code> object as returned by the function <code>PICS</code> run on the control data.
filter	A list of ranges for filtering regions based on PICS parameters. By default filter is set to 'NULL' and all regions are used.
h	A value between 0 and 1, representing the desired FDR. This simply draws a horizontal line at the given value.
...	Further graphical parameters passed to the generic function <code>plot</code> .

**Author(s)**

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

**References**

X. Zhang, G. Robertson, M. Krzywinski, K. Ning, A. Droit, S. Jones, and R. Gottardo, "PICS: Probabilistic Inference for ChIP-seq," *Biometrics*, iss. In press, 2010.

**See Also**

[PICS](#)

---

<code>segChrRead</code>	<i>Segmentation of paired-end sequencing data</i>
-------------------------	---

---

**Description**

These two functions are part of the segmentation step for paired-end sequencing data and are exported to be used in `PING` package.

---

<code>segmentPICS</code>	<i>Segment the genome into candidate regions</i>
--------------------------	--

---

**Description**

Pre-process bidirectional aligned reads data from a single ChIP-Seq experiment to detect candidate regions with a minimum number of forward and reverse reads. These candidate regions will then be processed by `PICS`.

**Usage**

```
segmentPICS(data, dataC=NULL, map=NULL, minReads=2, minReadsInRegion=3,
            jitter=FALSE, dataType="TF", maxLregion=0, minLregion=100)
```

**Arguments**

data	A linkS4class{GRanges} object containing the IP reads. See details for more information on how to set up the data.
dataC	A linkS4class{GRanges} 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.
dataType	Type of experiment. "TF" or "H".
maxLregion	The maximum length.
minLregion	The minimum length.

**Value**

An object of class `segReadsList` containing the results for all regions pre-processed.

**Author(s)**

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

**References**

X. Zhang, G. Robertson, M. Krzywinski, K. Ning, A. Droit, S. Jones, and R. Gottardo, "PICS: Probabilistic Inference for ChIP-seq" arXiv, 0903.3206, 2009.

**See Also**

[segReadsList](#)

**Examples**

```
# Read data
path<-system.file("extdata",package="PICS")
## Note that the col name for the chromosome needs to be space and not chr
dataIP<-read.table(file.path(path,"Treatment_tags_chr21_sort.bed"),header=TRUE,colClasses=c("factor","integer"))
dataIP<-as(dataIP,"GRanges")

dataCont<-read.table(file.path(path,"Input_tags_chr21_sort.bed"),header=TRUE,colClasses=c("factor","integer"))
dataCont<-as(dataCont,"GRanges")

map<-read.table(file.path(path,"mapProfileShort"),header=TRUE,colClasses=c("factor","integer","integer","NULL"))
map<-as(map,"GRanges")
seg<-segmentPICS(dataIP, dataC=dataCont, map=map, minReads=1)
```

segReads

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

Pre-process bidirectional aligned reads data from a single ChIP-Seq experiment to detect candidate regions with a minimum number of forward and reverse reads. These candidate regions will then be processed by PICS.

**Methods**

**map** signature(x = ``pics``): subset PICS object.

**Author(s)**

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

**References**

X. Zhang, G. Robertson, M. Krzywinski, K. Ning, A. Droit, S. Jones, and R. Gottardo, “PICS: Probabilistic Inference for ChIP-seq” arXiv, 0903.3206, 2009. To appear in Biometrics.

**See Also**

[pics](#)

segReadsList

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

Pre-process bidirectional aligned reads data from a single ChIP-Seq experiment to detect candidate regions with a minimum number of forward and reverse reads. These candidate regions will then be processed by PICS.

**Methods**

[ signature(x = ``pics``): subset gadem object.

[[ signature(x = ``pics``): subset gadem object.

**Methods**

**length** signature(x = ``pics``): subset PICS object.

**Author(s)**

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

**References**

X. Zhang, G. Robertson, M. Krzywinski, K. Ning, A. Droit, S. Jones, and R. Gottardo, “PICS: Probabilistic Inference for ChIP-seq” arXiv, 0903.3206, 2009. To appear in Biometrics.

**See Also**

[pics](#)

---

segReadsListPE	<i>List of segReadsPE objects</i>
----------------	-----------------------------------

---

**Description**

A list of segReadsPE. The class also store information related to the segmentation process, keeping a trace of the parameters used and the proportion of forward and reverse reads for the input and the control.

**Methods**

[ signature(x = ``pics``): subset gadem object.  
 [[ signature(x = ``pics``): subset gadem object.

**Extends**

Class [segReadsList](#), directly.

**Author(s)**

Xuekui Zhang

**See Also**

[segReadsPE](#) [segReadsList](#)

---

segReadsPE	<i>Class to store post-segmentation result</i>
------------	--

---

**Description**

This class stores the information of the segmentation performed by segmentPING. It is used as the input of the PING function.

**Extends**

Class [segReadsList](#), directly.

**Author(s)**

Xuekui Zhang

**See Also**

[segReads](#) [segReadsListPE](#)

---

setParaEM	<i>Function that returns a list of parameters for the EM algorithm that can be used as an argument of PICS.</i>
-----------	---

---

**Description**

This function takes from 0 to 7 EM algorithm parameters as argument, check if they are valid and returns a list to be used in a call to PICS.

**Usage**

```
setParaEM(minK=1,maxK=15,tol=1e-4,B=100,mSelect="BIC",mergePeaks=TRUE,mapCorrect=TRUE,dataType=
```

**Arguments**

minK	An integer. The minimum number of binding events per region. If the value is 0, the minimum number is automatically calculated.
maxK	An integer. The maximum number of binding events per region. If the value is 0, the maximum number is automatically calculated.
tol	A numeric. The tolerance for the EM algorithm.
B	An integer. 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.
mergePeaks	A logical stating whether overlapping binding events should be picked.
mapCorrect	A logical stating whether mappability profiles should be incorporated in the estimation, i.e: missing reads estimated.
dataType	A character. If a dataType is set, the algorithm will use the default parameters for this type of data (all the previous arguments will be ignored).

**Value**

Returns a list of parameters to be used in PICS.

**Author(s)**

Renan Sauteraud

**See Also**

PICS

---

setParaPrior	<i>Function that returns a list of parameters that can be used as an argument of PICS.</i>
--------------	--

---

### Description

This function takes from 0 to 6 parameters as argument, check if they are valid and returns a list to be used in a call to PICS.

### Usage

```
setParaPrior(xi=200, rho=1, alpha=20, beta=40000, lambda=0, dMu=0, dataType=NULL, PExi=0)
```

### Arguments

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 PICS model
lambda	An integer. The precision of the prior for mu used for histone data.
dMu	An integer. Our best guess for the distance between two neighboring nucleosomes.
dataType	A character string. If a valid dataType is specified, use our suggested parameters. "MNase" or "sonicated"
PExi	A numeric. With paired end data, 'xi' can be calculated directly from the reads. If PExi is set, it will overwrite the xi determined by the dataType.

### Value

Returns a list of 6 parameters to be used in PICS.

### Author(s)

Renan Sauteraud

### See Also

PICS

### Examples

```
# set prior for PICS data
paraPrior<-setParaPrior()
# set prior for sonicated data using our selected default parameters
paraPrior<-setParaPrior(dataType="sonicated")
```

---

show	<i>show PICS</i>
------	------------------

---

### Description

This methods show the objects of PICS

### Usage

```
## S4 method for signature 'pics'  
show(object)  
## S4 method for signature 'picsError'  
show(object)  
## S4 method for signature 'picsList'  
show(object)  
## S4 method for signature 'segReads'  
show(object)  
## S4 method for signature 'segReadsList'  
show(object)
```

### Arguments

object            Object returned from [pics](#) .

### Details

List of the slots include in the object

### Author(s)

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

### See Also

[summary](#)

---

summary	<i>summary PICS</i>
---------	---------------------

---

### Description

This methods summarized 'pics', 'picsList', 'seg' or 'segList' objects.

**Usage**

```
## S4 method for signature 'pics'  
summary(object)  
## S4 method for signature 'picsList'  
summary(object)  
## S4 method for signature 'segReads'  
summary(object)  
## S4 method for signature 'segReadsList'  
summary(object)
```

**Arguments**

object            Object returned from `pics` .

**Author(s)**

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

**See Also**

[show](#)

---

summarySeg

*Summarize a segReadsList object.*

---

**Description**

Returns info about a `segReadsList` object in a `data.frame` containing the following informations:  
chr : chromosome id NF : number of forward reads NR : number of reverse reads L : length of segment  
min: start location of segments max: end location of segments

**Usage**

```
summarySeg(seg)
```

**Arguments**

seg            An object of class `segReadsList`

**Value**

A six columns `data.frame`.

**Author(s)**

Xuekui Zhang

**See Also**

[segReadsList](#)

# Index

- \*Topic **classes**
  - segReadsListPE, 15
  - segReadsPE, 15
- \*Topic **data**
  - makeRangedDataOutput, 3
- \*Topic **functions**
  - bam2gr, 2
  - picsFDR, 8
  - segmentPICS, 12
  - setParaEM, 16
  - setParaPrior, 17
  - summarySeg, 19
- \*Topic **graphs**
  - plot-FDR, 11
- \*Topic **models**
  - pics, 4
  - pics-class, 6
  - picsError-class, 7
  - picsList-class, 9
  - segReads, 14
  - segReadsList, 14
- \*Topic **print**
  - show, 18
  - summary, 18
- \*Topic **process**
  - makeRangedDataOutput, 3
- [, picsList, ANY, ANY-method (picsList-class), 9
- [, picsList-method (picsList-class), 9
- [, segReadsList, ANY, ANY-method (segReadsList), 14
- [, segReadsList-method (segReadsList), 14
- [, segReadsListPE, ANY, ANY-method (segReadsListPE), 15
- [, segReadsListPE-method (segReadsListPE), 15
- [[, picsList, ANY, ANY-method (picsList-class), 9
- [[, picsList-method (picsList-class), 9
- [[, segReadsList, ANY, ANY-method (segReadsList), 14
- [[, segReadsList-method (segReadsList), 14
- [[, segReadsListPE, ANY, ANY-method (segReadsListPE), 15
- %in%, ANY, segReads-method (segReads), 14
- as.list, AlignedRead-method (pics), 4
- bam2gr, 2
- candidate.region (segChrRead), 12
- chromosome (pics-class), 6
- chromosome, data.frame-method (pics-class), 6
- chromosome, pics-method (pics-class), 6
- chromosome, picsError-method (picsError-class), 7
- chromosome, picsList-method (picsList-class), 9
- code (pics), 4
- code, pics-method (pics), 4
- code, picsError-method (pics), 4
- code, picsList-method (pics), 4
- coerce (pics), 4
- coerce, AlignedRead, GenomeData-method (pics), 4
- coerce, data.frame, GenomeData-method (pics), 4
- coerce, data.frame, GRanges-method (pics), 4
- coerce, data.frame, picsList-method (pics), 4
- coerce, picsList, data.frame-method (pics), 4
- coerce, picsList, RangedData-method (pics), 4
- coerce, RangedData, GenomeData-method (pics), 4
- delta (pics-class), 6
- delta, data.frame-method (pics-class), 6
- delta, pics-method (pics-class), 6
- delta, picsError-method (picsError-class), 7
- delta, picsList-method (picsList-class), 9

- density, picsList-method  
(picsList-class), 9
- FDR (plot-FDR), 11
- K (pics), 4
- K, pics-method (pics), 4
- K, picsError-method (pics), 4
- K, picsList-method (pics), 4
- length (picsList-class), 9
- length, picsList-method  
(picsList-class), 9
- length, segReadsList-method  
(segReadsList), 14
- makeRangedDataOutput, 3
- map (segReads), 14
- map, segReads-method (segReads), 14
- map, segReadsList-method (segReads), 14
- maxRange (pics), 4
- maxRange, pics-method (pics), 4
- maxRange, picsError-method (pics), 4
- maxRange, picsList-method (pics), 4
- minRange (pics), 4
- minRange, pics-method (pics), 4
- minRange, picsError-method (pics), 4
- minRange, picsList-method (pics), 4
- mu (pics-class), 6
- mu, data.frame-method (pics-class), 6
- mu, pics-method (pics-class), 6
- mu, picsError-method (picsError-class), 7
- mu, picsList-method (picsList-class), 9
- newPics (pics-class), 6
- newPics, pics-method (pics-class), 6
- newPicsError (picsError-class), 7
- newPicsError, picsError-method  
(picsError-class), 7
- newPicsList (picsList-class), 9
- newPicsList, picsList-method  
(picsList-class), 9
- PICS, 12
- PICS (pics), 4
- pics, 4, 5, 7–9, 11, 14, 15, 18, 19
- pics-class, 6
- pics-class (pics), 4
- picsError, 7
- picsError (picsError-class), 7
- picsError-class, 7
- picsFDR, 8
- picsList, 9
- picsList (picsList-class), 9
- picsList-class, 9
- plot, pics, segReads-method (pics), 4
- plot, picsError, segReads-method (pics), 4
- plot, picsList, picsList-method  
(plot-FDR), 11
- plot, picsList, segReadsList-method  
(pics), 4
- plot-FDR, 11
- score (pics), 4
- score, data.frame-method (pics-class), 6
- score, pics-method (pics-class), 6
- score, picsError-method  
(picsError-class), 7
- score, picsList-method (picsList-class),  
9
- scoreForward (pics), 4
- scoreForward, data.frame-method (pics), 4
- scoreForward, pics-method (pics), 4
- scoreForward, picsError-method (pics), 4
- scoreForward, picsList-method (pics), 4
- scoreReverse (pics), 4
- scoreReverse, data.frame-method (pics), 4
- scoreReverse, pics-method (pics), 4
- scoreReverse, picsError-method (pics), 4
- scoreReverse, picsList-method (pics), 4
- se (pics-class), 6
- se, data.frame-method (pics-class), 6
- se, pics-method (pics-class), 6
- se, picsError-method (picsError-class), 7
- se, picsList-method (picsList-class), 9
- seF (pics-class), 6
- seF, data.frame-method (pics-class), 6
- seF, pics-method (pics-class), 6
- seF, picsError-method (picsError-class),  
7
- seF, picsList-method (picsList-class), 9
- segChrRead, 12
- segmentPICS, 2, 12
- segReads, 14, 16
- segReads-class (segReads), 14
- segReadsGeneric (segmentPICS), 12
- segReadsList, 13, 14, 15, 19
- segReadsList-class (segReadsList), 14
- segReadsListPE, 15, 16
- segReadsListPE-class (segReadsListPE),  
15
- segReadsPE, 15, 15
- segReadsPE-class (segReadsPE), 15
- seR (pics-class), 6
- seR, data.frame-method (pics-class), 6
- seR, pics-method (pics-class), 6

- seR,picsError-method (picsError-class),  
7
- seR,picsList-method (picsList-class), 9
- setParaEM, 16
- setParaPrior, 17
- show, 18, 19
- show,pics-method (show), 18
- show,picsError-method (show), 18
- show,picsList-method (show), 18
- show,segReads-method (show), 18
- show,segReadsList-method (show), 18
- show-methods (show), 18
- sigmaSqF (pics), 4
- sigmaSqF,data.frame-method (pics), 4
- sigmaSqF,pics-method (pics), 4
- sigmaSqF,picsError-method (pics), 4
- sigmaSqF,picsList-method (pics), 4
- sigmaSqR (pics), 4
- sigmaSqR,data.frame-method (pics), 4
- sigmaSqR,pics-method (pics), 4
- sigmaSqR,picsError-method (pics), 4
- sigmaSqR,picsList-method (pics), 4
- summary, 18, 18
- summary,pics-method (summary), 18
- summary,picsList-method (summary), 18
- summary,segReads-method (summary), 18
- summary,segReadsList-method (summary),  
18
- summary-methods (summary), 18
- summarySeg, 19
  
- w (pics-class), 6
- w,data.frame-method (pics-class), 6
- w,pics-method (pics-class), 6
- w,picsError-method (picsError-class), 7
- w,picsList-method (picsList-class), 9
- wigDensity (pics), 4
- wigDensity,pics-method (pics), 4
- wigDensity,picsError-method (pics), 4
- wigDensity,picsList-method (pics), 4