Package 'nuCpos'

March 30, 2021

Title An R package for prediction of nucleosome positions

Version 1.8.0

Description nuCpos, a derivative of NuPoP, is an R package for prediction of nucleosome positions. In nuCpos, a duration hidden Markov model is trained with a chemical map of nucleosomes either from budding yeast, fission yeast, or mouse embryonic stem cells. nuCpos outputs the Viterbi (most probable) path of nucleosome-linker states, predicted nucleosome occupancy scores and histone binding affinity (HBA) scores as NuPoP does. nuCpos can also calculate local and whole nucleosomal HBA scores for a given 147-bp sequence. Furthermore, effect of genetic alterations on nucleosome occupancy can be predicted with this package. The parental package NuPoP, which is based on an MNase-seq-based map of budding yeast nucleosomes, was developed by Ji-Ping Wang and Liqun Xi, licensed under GPL-2.

Author Hiroaki Kato, Takeshi Urano

Maintainer Hiroaki Kato <hkato@med.shimane-u.ac.jp>

License file LICENSE

Depends R (>= 3.6)

Imports graphics, methods

Suggests NuPoP, Biostrings, testthat

biocViews Genetics, Epigenetics, NucleosomePositioning, HiddenMarkovModel, ImmunoOncology

Date 2019-11-08

NeedsCompilation yes

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

git_branch RELEASE_3_12

git_last_commit b77ccdf

git_last_commit_date 2020-10-27

Date/Publication 2021-03-29

R topics documented:

nuCpos-package *An R package for nucleosome positioning prediction*

Description

nuCpos, a derivative of **NuPoP**, is an R package for prediction of nucleosome positions. In **nuCpos**, a duration hidden Markov model is trained with a chemical map of nucleosomes either from budding yeast (Brogaard et al. (2012)), fission yeast (Moyle-Heyrman et al. (2012)), or mouse embryonic stem cells (Voong et al. (2016)). **nuCpos** outputs the Viterbi (most probable) path of nucleosome-linker states, predicted nucleosome occupancy scores and histone binding affinity (HBA) scores as **NuPoP** does. **nuCpos** can also calculate local and whole nucleosomal HBA scores for a given 147-bp sequence. Furthermore, effect of genetic alterations on nucleosome occupancy can be predicted with this package. The parental package **NuPoP**, which is based on an MNase-seq-based map of budding yeast nucleosomes, was developed by Ji-Ping Wang and Liqun Xi, licensed under GPL-2. Please refer to Xi et al. (2010) and Wang et al. (2008) for technical details of **NuPoP**.

Details

Package:	nuCpos
Type:	Package
Version:	1.5.1
Date:	2019-11-08
License:	GPL-2

`predNuCpos`: R function for prediction of nucleosome positioning, nucleosome occupancy and HBA scores.

`HBA`: R function for calculation of the histone binding affinity score of a whole nucleosome.

`localHBA`: R function for calculation of the local histone binding affinity.

`mutNuCpos`: R function for predicting the effect of a genetic alteration on nucleosome positioning.

Author(s)

Hiroaki Kato and Takeshi Urano

Maintainer: Hiroaki Kato<hkato@med.shimane-u.ac.jp>

References

1. Wang JP, Fondufe-Mittendorf Y, Xi L, Tsai GF, Segal E and Widom J (2008). Preferentially quantized linker DNA lengths in *Saccharomyces cerevisiae*. *PLoS Computational Biology*, 4(9):e1000175.
2. Xi L, Fondufe-Mittendorf Y, Xia L, Flatow J, Widom J and Wang JP (2010). Predicting nucleosome positioning using a duration hidden markov model. *BMC Bioinformatics*, 11:346
3. Brogaard K, Xi L, and Widom J (2012). A map of nucleosome positions in yeast at base-pair resolution. *Nature*, 486(7404):496-501.
4. Moyle-Heyrman G, Zaichuk T, Xi L, Zhang Q, Uhlenbeck OC, Holmgren R, Widom J and Wang JP (2013). Chemical map of *Schizosaccharomyces pombe* reveals species-specific features in nucleosome positioning. *Proc. Natl. Acad. Sci. U. S. A.*, 110(50):20158-63.

5. Ichikawa Y, Morohoshi K, Nishimura Y, Kurumizaka H and Shimizu M (2014). Telomeric repeats act as nucleosome-disfavouring sequences in vivo. *Nucleic Acids Res.*, 42(3):1541-1552.

6. Voong LN, Xi L, Sebeson AC, Xiong B, Wang JP and Wang X (2016). Insights into Nucleosome Organization in Mouse Embryonic Stem Cells through Chemical Mapping. *Cell*, 167(6):1555-1570.

7. Fuse T, Katsumata K, Morohoshi K, Mukai Y, Ichikawa Y, Kurumizaka H, Yanagida A, Urano T, Kato H, and Shimizu M (2017). Parallel mapping with site-directed hydroxyl radicals and micrococcal nuclease reveals structural features of positioned nucleosomes in vivo. *Plos One*, 12(10):e0186974.

Examples

```
predNuCpos(file = system.file("extdata", "TRP1ARS1x1.fasta",
    package = "nuCpos"), species = "sc",
    ActLikePredNuPoP = TRUE)

## The prediction results are stored in the working directory.
```

HBA *R function for calculating the histone binding affinity score of a given 147-bp sequence.*

Description

This function invokes a Fortran subroutine to calculate histone binding score. Nucleosomal and linker models built upon the chemical maps are used for the calculation.

Usage

```
HBA(inseq, species = "mm", silent = FALSE)
```

Arguments

inseq	a character or DNAString object. The length of the character string must be 147 bp.
species	a character = mm, sc or sp; "mm" for mouse, "sc" for *S. cerevisiae* and "sp" for *S. pombe*.
silent	a logical value indicating whether messages are printed in the console.

Value

HBA outputs one numeric value: histone binding affinity for a whole nucleosome.

Examples

```
load(system.file("extdata","inseq.RData",package="nuCpos"))
HBA(inseq, species = "sc")
```

`localHBA` *R function for calculating the local histone binding score of a given 147-bp sequence.*

Description

This function invokes a Fortran subroutine to calculate local histone binding score. Nucleosomal and linker models built upon the chemical maps are used for the calculation.

Usage

```
localHBA(inseq, species = "mm", silent = FALSE)
```

Arguments

`inseq`	a character or DNAString object. The length of the character string must be 147 bp.
`species`	a character = mm, sc or sp; "mm" for mouse, "sc" for *S. cerevisiae* and "sp" for *S. pombe*.
`silent`	a logical value indicating whether messages are printed in the console.

Value

`localHBA` outputs a numeric vector of length 13: local histone binding affinity scores for specific regions in a nucleosome.

Examples

```
load(system.file("extdata","inseq.RData",package="nuCpos"))
localHBA(inseq, species = "sc")
```

`mutNuCpos` *R function for prediction of nucleosome positioning on a mutant sequence*

Description

This function plots the results of nucleosome positioning prediction for wild type and mutant sequences in a specified window. Nucleosomal and linker models built upon the chemical maps are used for the calculation. No file is generated in the current directry.

Usage

```
mutNuCpos(wtseq, site = 1, ins = "", del = 0,
    species = "mm", smoothHBA = FALSE, std = FALSE,
    plot.window = 501, prob.dyad = FALSE,
    show.viterbi = FALSE, occup.window = 200,
    show.occup.window = FALSE, ymax.prob = 1.1,
    ymax.occup = 1.1, ylim.HBA = c(-15, 5),
    annotation = data.frame(name = "", color = "", left = 0,
    right = 0)[0, ], full = FALSE)
```

Arguments

wtseq	a character or DNAString object. The wild-type sequence to be mutated. The string must not contain letters other than "A", "C", "G" or "T."
site	an integer. The site of mutagenesis.
ins	a character or DNAString object. The sequence to be inserted at the "site." The string must not contain letters other than "A", "C", "G" or "T." ins="" indicates no sequence will be inserted.
del	an integer. The length of the deleted region that starts at the "site." del=0 indicates no sequence will be deleted.
species	a character = mm, sc or sp; "mm" for mouse, "sc" for *S. cerevisiae* and "sp" for *S. pombe*.
smoothHBA	a logical value indicating whether smoothing of histone binding affinity should be applied as in the predNuPoP function of the parental package **NuPoP**.
std	a logical value indicating whether standardization should be applied to the histone binding affinity score.
plot.window	an integer. The window to be plotted. This must be an odd number.
prob.dyad	a logical value indicating whether the probability for the predicted dyads is plotted.
show.viterbi	a logical value indicating whether the viterbi path is plotted.
occup.window	an integer. The size of the window for the calculation of occupancy difference. occup.window=200 means that the sum of the absolute occupancy difference for the left-side and right-side 100-bp regions flanking the "site" is caluculated.
show.occup.window	a logical value indicating whether the window for the occupancy difference calculation is shown in the occupancy plots.
ymax.prob	an integer. Specify the upper limit of the y axis of the probability plots.
ymax.occup	an integer. Specify the upper limit of the y axis of the occupancy plots.
ylim.HBA	an integer vector of two values. Specify the lower and upper limits of the y axis of the histone binding affinity plots.
annotation	a data frame. Colored bars can be put under the plots.
full	a logical value indicating whether the calculation results will be returned as a data frame object.

Value

When the full argument is set as TRUE, the prediction results for the mutant sequence will be returned as a data frame object. The data frame has five columns as that produced by predNuCpos when its argument ActLikePredNuPoP was set as FALSE:

pos	position in the input DNA sequence
pstart	probability that a nucleosome starts at
nucoccup	nucleosome occupancy score
viterbi	Viterbi path (1 and 0 for the nucleosome and linker states, repsectively)
affinity	histone binding affinity score

When the full argument was set as FALSE, this function returns a named numeric vector, in which the occupancy difference and HBA scores around the target site are stored.

When ins="" and del=0 are applied, two wild-type sequences are used for the calculation and plotting; this yields no difference in the occupancy or HBA.

Examples

```
# Loading the sequence of TALS, a budding yeast
# minichromosome.
TALS <- paste(scan(file = system.file("extdata", "TALS.fasta",
    package="nuCpos"), what = character(), skip = 1), sep = "",
    collapse = "")

# Loading the telomere repeat sequence (hTELx12)
TTAGGGx12 <- paste(scan(file = system.file("extdata",
    "TTAGGGx12.fasta", package="nuCpos"), what = character(),
    skip = 1), sep = "", collapse = "")
mutNuCpos(TALS, site = 1464, ins= TTAGGGx12, species="sc",
    prob.dyad = TRUE, smoothHBA=TRUE, plot.window = 601,
    ylim.HBA = c(-11, 0),
    annotation = data.frame(name = "alpha2",
        color = "purple", left = 1534, right = 1559))

# Loading the telomere repeat isomeric sequence (SI-Ax12)
TGTAGGx12 <- paste(scan(file = system.file("extdata",
    "TGTAGGx12.fasta", package="nuCpos"), what = character(),
    skip = 1), sep = "", collapse = "")
mutNuCpos(TALS, site = 1464, ins= TGTAGGx12, species="sc",
    prob.dyad = TRUE, smoothHBA=TRUE, plot.window = 601,
    ylim.HBA = c(-11, 0),
    annotation = data.frame(name = "alpha2",
    color = "purple", left = 1534, right = 1559))

# DNA sequences used here are from Ichikawa et al. (2014).
```

`predNuCpos` *R function for prediction of nucleosome positioning*

Description

Like the `predNuPoP` function of the parental package **NuPoP** does, this function invokes Fortran codes to compute the Viterbi prediction of nucleosome positioning, nucleosome occupancy score and histone binding affinity score. Nucleosomal and linker models built upon the chemical maps are used for the calculation.

When `ActLikePredNuPoP` is `TRUE`, this function acts like the `predNuPoP` function of **NuPoP**: the function receives the path to a file containing a DNA sequence (specified by `file`) and save a text file containing the prediction results in the working directory. Nucleosome positioning throughout a long chromosome containing 'N' can be predicted.

When `ActLikePredNuPoP` is `FALSE` (dafault), this function directly receives a DNA sequence as an R object (`inseq`) and returns the prediction results as a data frame. 'N' must not be in the sequence.

Usage

```
predNuCpos(file, inseq, species="mm", smoothHBA=FALSE,
    std=FALSE, ActLikePredNuPoP = FALSE)
```

Arguments

file	The file path to the FASTA file to be tested. The FASTA must be in a single FASTA format. This will be ignored when ActLikePredNuPoP = FALSE.
inseq	a character or DNAString object. The length of the character string must be over 1 kb. This will be ignored when ActLikePredNuPoP = TRUE.
species	a character = mm, sc or sp; "mm" for mouse, "sc" for *S. cerevisiae* and "sp" for *S. pombe*.
smoothHBA	a logical value indicating whether smoothing of histone binding affinity should be applied as in the predNuPoP function of the parental package **NuPoP**.
std	a logical value indicating whether standardization should be applied to the histone binding affinity score.
ActLikePredNuPoP	a logical value indicating whether the function acts like the predNuPoP function in the parental package **NuPoP**.

Value

When the ActLikePredNuPoP argument is set as TRUE, predNuCpos outputs the prediction results into the working directory, in the same format as that generated by the predNuPoP function of **NuPoP**. Thus, it can be handled by the **NuPoP** functions readNuPoP and plotNuPoP. The output file is named after the input file with an extension "_Prediction4.txt". The output file has five columns:

Position	position in the input DNA sequence
P-start	probability that a nucleosome starts at
Occup	nucleosome occupancy score
N/L	Viterbi path (1 and 0 for the nucleosome and linker states, repsectively)
Affinity	histone binding affinity score

When the ActLikePredNuPoP argument is set as FALSE, predNuCpos outputs the prediction results as a data frame object with five columns, on which the plotNuPoP function of **NuPoP** can be applied:

pos	position in the input DNA sequence
pstart	probability that a nucleosome starts at
nucoccup	nucleosome occupancy score
viterbi	Viterbi path (1 and 0 for the nucleosome and linker states, repsectively)
affinity	histone binding affinity score

Examples

```
predNuCpos(file = system.file("extdata", "TRP1ARS1x1.fasta",
    package="nuCpos"), species="sc", smoothHBA=FALSE,
    std=FALSE, ActLikePredNuPoP = TRUE)
library(NuPoP)
results.TRP1ARS1.1 <- readNuPoP("TRP1ARS1x1.fasta_Prediction4.txt",
    startPos = 1, endPos = 1465)
results.TRP1ARS1.1[72:76,]
plotNuPoP(results.TRP1ARS1.1)
TRP1ARS1 <- paste(scan(file =
```

```
    system.file("extdata", "TRP1ARS1x1.fasta", package = "nuCpos"),
    what = character(), skip = 1), sep = "", collapse = "")
results.TRP1ARS1.2 <-
    predNuCpos(inseq = TRP1ARS1, species = "sc", smoothHBA = FALSE,
    ActLikePredNuPoP = FALSE)
results.TRP1ARS1.2[72:76,]
plotNuPoP(results.TRP1ARS1.2)
## The DNA sequence TRP1ARS1 is from Fuse et al. (2017).
```

Index