

Package ‘CaTCH’

December 6, 2018

Type Package

Title Call a hierarchy of domains based on Hi-C data

Version 1.0

Date 2016-07-05

Author Yinxiu Zhan

Maintainer Yinxiu Zhan <yinxiu.zhan@fmi.ch>

Imports parallel

Description This package allows building the hierarchy of domains starting from Hi-C data. Each hierarchical level is identified by a minimum value of physical insulation between neighboring domains.

License GPL-2 or later

NeedsCompilation yes

R topics documented:

CaTCH-package	1
domain.call	2

Index	4
--------------	----------

CaTCH-package	<i>Call a hierarchy of domains based on Hi-C data</i>
---------------	---

Description

This package allows building the hierarchy of domains starting from Hi-C data. Each hierarchical level is identified by a minimum value of physical insulation between neighboring domains.

Details

The DESCRIPTION file:

Package:	CaTCH
Type:	Package
Title:	Call a hierarchy of domains based on Hi-C data

Version: 1.0
 Date: 2016-07-05
 Author: Yinxu Zhan
 Maintainer: Yinxu Zhan <yinxu.zhan@fmi.ch>
 Imports: parallel
 Description: This package allows building the hierarchy of domains starting from Hi-C data. Each hierarchical level is identified by a minimum value of physical insulation between neighboring domains.
 License: GPL-2 or later
 NeedsCompilation: yes
 Packaged: 2018-07-10 15:18:41 UTC; yinxu

Index of help topics:

CaTCH-package	Call a hierarchy of domains based on Hi-C data
domain.call	Call a hierarchy of domains based on Hi-C data

Author(s)

Yinxu Zhan
 Maintainer: Yinxu Zhan <yinxu.zhan@fmi.ch>

References

Zhan et al, Reciprocal insulation analysis of Hi-C data shows that TADs represent a functionally but not structurally privileged scale in the hierarchical folding of chromosomes, Genome Research 2017

domain.call	<i>Call a hierarchy of domains based on Hi-C data</i>
-------------	---

Description

This package allows building the hierarchy of domains starting from Hi-C data. Each hierarchical level is identified by a minimum value of physical insulation between neighboring domains.

Usage

```

domain.call(input)
domain.call.parallel(inputs,ncpu=parallel::detectCores()-1L)

```

Arguments

input	File containing Hi-C data for the SINGLE CHROMOSOME with 6 columns: <ul style="list-style-type: none"> • col1 = chromosome • col2 = genomic coordinate of the start region • col3 = genomic coordinate of the end region • col4 = bin of the start region (genomic coordinate divided by binsize) • col5 = bin of the end region • col6 = Hi-C counts
-------	---

*****OR*****

File containing Hi-C data for the SINGLE CHROMOSOME with 4 columns:

- col1 = chromosome
- col2 = bin of the start region (genomic coordinate divided by binsize)
- col3 = bin of the end region (genomic coordinate divided by binsize)
- col4 = Hi-C counts

inputs A character vector of files as input
ncpu Number of cpu that you want to use (Default ncpu=4)

Value

Output of

domain.call A list with two elements

- ncluster: A data.frame with 3 columns
 - Chromosome (chromosome)
 - Reciprocal insulation (RI)
 - Number of domains (ndomains)
- clusters: A data.frame with 5 columns
 - Chromosome (chromosome)
 - Reciprocal insulation (RI)
 - Start of domain (start)
 - End of domain (end)
 - Actual insulation (insulation)

domain.call.parallel A list of outputs as in domain.call for each file

Author(s)

Yinxu Zhan

References

Zhan et al....

Examples

```
#R code to be here
fileinput=system.file("Test.dat.gz",package="CaTCH")
library(CaTCH)
domain.call(fileinput)
```

Index

*Topic **TADs, domain calling, Hi-C**

CaTCH-package, [1](#)

*Topic **TADs**

domain.call, [2](#)

*Topic **hierarchy**

domain.call, [2](#)

CaTCH (CaTCH-package), [1](#)

CaTCH-package, [1](#)

domain.call, [2](#)