

# Package ‘GenomicInteractionNodes’

February 20, 2025

**Type** Package

**Version** 1.11.3

**Title** A R/Bioconductor package to detect the interaction nodes from HiC/HiChIP/HiCAR data

**Description** The GenomicInteractionNodes package can import interactions from bedpe file and define the interaction nodes, the genomic interaction sites with multiple interaction loops. The interaction nodes is a binding platform regulates one or multiple genes. The detected interaction nodes will be annotated for downstream validation.

**License** file LICENSE

**Depends** R (>= 4.2.0), stats

**Imports** AnnotationDbi, graph, GO.db, GenomicRanges, GenomicFeatures, GenomeInfoDb, methods, IRanges, RBGL, S4Vectors

**Suggests** RUnit, BiocStyle, knitr, rmarkdown, rtracklayer, testthat, TxDb.Hsapiens.UCSC.hg19.knownGene, org.Hs.eg.db

**URL** <https://github.com/jianhong/GenomicInteractionNodes>

**BugReports** <https://github.com/jianhong/GenomicInteractionNodes/issues>

**biocViews** HiC, Sequencing, Software

**VignetteBuilder** knitr

**RoxygenNote** 7.3.2

**Encoding** UTF-8

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

**git\_branch** devel

**git\_last\_commit** 628aad6

**git\_last\_commit\_date** 2025-02-12

**Repository** Bioconductor 3.21

**Date/Publication** 2025-02-20

**Author** Jianhong Ou [aut, cre] (ORCID: <<https://orcid.org/0000-0002-8652-2488>>), Yarui Diao [fnd]

**Maintainer** Jianhong Ou <jou@morgridge.org>

## Contents

annotateNodes . . . . .	2
createRandomNodes . . . . .	3
detectNodes . . . . .	4
enrichmentAnalysis . . . . .	4

<b>Index</b>	<b>6</b>
--------------	----------

---

annotateNodes	<i>Annotate node regions</i>
---------------	------------------------------

---

## Description

Assign gene id and gene symbols to node regions by interacted.

## Usage

```
annotateNodes(
  node_regions,
  txdb,
  orgDb,
  upstream = 2000,
  downstream = 500,
  ...
)
```

## Arguments

node_regions	GRanges object represent regions interacted with nodes.
txdb	An object of <a href="#">TxDb</a> to extract gene information
orgDb	An object of <a href="#">OrgDb</a> to extract gene symbols
upstream, downstream	An integer(1) value indicating the number of bases upstream or downstream from the transcription start site. For additional details see <a href="#">promoters</a> .
...	parameter can be passed to <a href="#">genes</a>

## Value

GRanges object with gene\_id and symbols metadata.

## Examples

```
library(TxDb.Hsapiens.UCSC.hg19.knownGene) ## for human hg19
library(org.Hs.eg.db) ## used to convert gene_id to gene_symbol
set.seed(123)
node_regions <- createRandomNodes(TxDb.Hsapiens.UCSC.hg19.knownGene)
annotateNodes(node_regions, TxDb.Hsapiens.UCSC.hg19.knownGene, org.Hs.eg.db)
```

---

createRandomNodes	<i>Create a list of random nodes</i>
-------------------	--------------------------------------

---

## Description

Generate a list of random nodes used for example or test.

## Usage

```
createRandomNodes(  
  txdb,  
  seq = "chr22",  
  size = 500,  
  upstream = 500,  
  downstream = 500,  
  maxDist = 1e+06,  
  wid = 5000  
)
```

## Arguments

txdb	An TxDb object.
seq	seqlevels to be kept.
size	the length of regions involved in nodes
upstream, downstream	upstream or downstream for promoters
maxDist	maximal distance from promoters
wid	regions width.

## Value

An GRanges object with comp\_id metadata.

## Examples

```
library(TxDb.Hsapiens.UCSC.hg19.knownGene)  
set.seed(123)  
node_regions <- createRandomNodes(TxDb.Hsapiens.UCSC.hg19.knownGene)
```

detectNodes            *Detect the interaction node*

---

### Description

Define the interaction node from input Pairs.

### Usage

```
detectNodes(interaction, pval_cutoff = 0.05, ...)
```

### Arguments

interaction	An object of <a href="#">Pairs</a> to represent interactions.
pval_cutoff	Cutoff P value for interaction node by Poisson distribution
...	Not used.

### Value

A list of interaction nodes with elements: node\_connection, Pairs object represent interactions interacted with nodes; nodes, GRanges object represent regions with maximal interactions involved in nodes; node\_regions, GRanges object represent regions interacted with nodes.

### Examples

```
library(rtracklayer)
p <- system.file("extdata", "WT.2.bedpe",
                 package = "GenomicInteractionNodes")
interactions <- import(con=p, format="bedpe")
nodes <- detectNodes(interactions)
```

---

enrichmentAnalysis    *Gene ontology enrichment analysis*

---

### Description

GO enrichment analysis for nodes

**Usage**

```
enrichmentAnalysis(
  node_regions,
  orgDb,
  onto = c("BP", "CC", "MF"),
  minGeneNum = 3,
  evidence = list(Experimental_evidence_codes = c("EXP", "IDA", "IPI", "IMP", "IGI",
    "IEP", "HTP", "HDA", "HMP", "HGI", "HEP"), `Phylogenetically-inferred_annotations` =
    c("IBA", "IBD", "IKR", "IRD"), Computational_analysis_evidence_codes = c("ISS",
    "ISO", "ISA", "ISM", "IGC", "RCA"), Author_statement_evidence_codes = c("TAS",
    "NAS"), Curator_statement_evidence_codes = c("IC", "ND"),
    Electronic_annotation_evidence_code = c("IEA")),
  ...
)
```

**Arguments**

node_regions	GRanges object represent regions interacted with nodes. The object must be annotated by <a href="#">annotateNodes</a> with comp_id and gene_id in the metadata.
orgDb	An object of <a href="#">OrgDb</a> to extract gene symbols.
onto	Ontology category.
minGeneNum	An integer(1) value indicating the minimal number of gene to start the enrichment analysis. If total gene counts is smaller than the 'minGeneNum', the NULL will be returned.
evidence	The acceptable evidence code.
...	Not used.

**Value**

A list with element enriched and enriched\_in\_compound. Or NULL if total counts of gene is smaller than 'minGeneNum'.

**Examples**

```
library(TxDb.Hsapiens.UCSC.hg19.knownGene) ## for human hg19
library(org.Hs.eg.db) ## used to convert gene_id to gene_symbol
library(GO.db)
set.seed(123)
node_regions <- createRandomNodes(TxDb.Hsapiens.UCSC.hg19.knownGene)
node_regions <-
  annotateNodes(node_regions,
               TxDb.Hsapiens.UCSC.hg19.knownGene,
               org.Hs.eg.db)
enr <- enrichmentAnalysis(node_regions, org.Hs.eg.db, onto="BP")
```

# Index

annotateNodes, [2](#), [5](#)

createRandomNodes, [3](#)

detectNodes, [4](#)

enrichmentAnalysis, [4](#)

genes, [2](#)

OrgDb, [2](#), [5](#)

Pairs, [4](#)

promoters, [2](#)

TxDb, [2](#)