

# Package ‘DominoEffect’

November 21, 2024

**Type** Package

**Title** Identification and Annotation of Protein Hotspot Residues

**Version** 1.27.0

**Author** Marija Buljan and Peter Blattmann

**Maintainer** Marija Buljan <marija.buljan.2@gmail.com>,  
Peter Blattmann <peter\_blattnann@bluewin.ch>

**Description** The functions support identification and annotation of hotspot residues in proteins. These are individual amino acids that accumulate mutations at a much higher rate than their surrounding regions.

**License** GPL (>= 3)

**Encoding** UTF-8

**LazyData** true

**Depends** R(>= 3.5)

**Imports** biomaRt, data.table, utils, stats, Biostrings, palign,  
SummarizedExperiment, VariantAnnotation, AnnotationDbi,  
GenomeInfoDb, IRanges, GenomicRanges, methods

**Suggests** knitr, testthat, rmarkdown

**RoxygenNote** 6.0.1

**VignetteBuilder** knitr

**biocViews** Software, SomaticMutation, Proteomics, SequenceMatching,  
Alignment

**NeedsCompilation** no

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

**git\_branch** devel

**git\_last\_commit** deeeec

**git\_last\_commit\_date** 2024-10-29

**Repository** Bioconductor 3.21

**Date/Publication** 2024-11-21

## Contents

align_to_unip . . . . .	2
calculate_boundary . . . . .	3
DominoData . . . . .	4
DominoEffect . . . . .	4
GPo_of_hotspots . . . . .	5
identify_hotspots . . . . .	6
import_txdb . . . . .	7
map_to_func_elem . . . . .	8

<b>Index</b>	<b>10</b>
--------------	-----------

---

align_to_unip	<i>Align protein segment around the hotspot to the UniProt/Swiss-Prot KB sequence.</i>
---------------	--

---

## Description

This function aligns the Ensembl protein region with a hotspot to the UniProt sequence. The Ensembl region encompasses 15 amino acids where the hotspot is in the middle. If the hotspot was at the protein start or end the region is still 15 amino acids long, but the hotspot position is shifted.

## Usage

```
align_to_unip(ens.seq, uni.seq, ensembl_mut_position)
```

## Arguments

ens.seq	AAString object with the Ensembl protein sequence corresponding to the representative transcript.
uni.seq	AAString with the UniProt sequence for the identifier matching the Ensembl gene name.
ensembl_mut_position	Numeric vector defining the hotspot position in the Ensembl sequence, i.e. in the ens.seq

## Value

Returns a list where the first element is a character vector defining whether there was a significant alignment and the second element provides the hotspot position in the UniProt sequence.

## Author(s)

Marija Buljan <buljan@imsb.biol.ethz.ch> Peter Blattmann <blattmann@imsb.biol.ethz.ch>

**Examples**

```
library(Biostrings)

ens.seq <- AAString("MDLSALREEVQNVINAMQKILECPICLELIKEPVSTKCDHIFCKFCMLK")
uni.seq <- AAString("MDLSALRVEEVQNVINAMQKILECPICLELIKEPVSTKCDHIFCKFCMLA")
ensembl_mut_position <- 25

align_to_unip(ens.seq, uni.seq, ensembl_mut_position)
```

---

calculate\_boundary      *calculate\_boundary*

---

**Description**

The function calculates boundaries of sequence windows around the mutation. It is possible to define up to two window lengths. If the mutation is close to the start or end of the protein, the region is extended into the other direction to keep the indicated size

**Usage**

```
calculate_boundary(mut_pos_numeric, length_aa, flanking_region)
```

**Arguments**

mut_pos_numeric	Amino acid position of mutation
length_aa	Length of transcript in amino acids
flanking_region	Vector containing two flanking regions

**Value**

returns a list with the boundaries for the two regions

**Author(s)**

Marija Buljan <buljan@imsb.biol.ethz.ch> Peter Blattmann <blattmann@imsb.biol.ethz.ch>

**Examples**

```
calculate_boundary(250, 500, c(200, 300))
calculate_boundary(250, 500, 300)
```

---

DominoData                      *Sample data*

---

### Description

Sample Data

### Author(s)

Marija Buljan <buljan@imsb.biol.ethz.ch> Peter Blattmann <blattmann@imsb.biol.ethz.ch>

---

DominoEffect                      *Identification of significant mutation hotspot residues.*

---

### Description

The function identifies individual amino acid residues, which accumulate a high fraction of the overall mutation load within a protein. After detecting mutation hotspots, the function obtains UniProt/Swiss-Prot KB functional and structural annotations for the affected protein regions and checks for the sequence agreement.

### Usage

```
DominoEffect(mutation_dataset, gene_data, snp_data,
min_n_muts = 5, MAF_thresh = 0.01,
flanking_region = c(200, 300),
poisson.thr = 0.01, percentage.thr = 0.15,
ratio.thr = 45, approach = "percentage", write_to_file = "NO",
ens_release = "https://feb2023.archive.ensembl.org")
```

### Arguments

mutation_dataset	Object containing a table with the mutation data (e.g. TCGA point mutations mapped to protein level).
gene_data	DominoData object containing information about Ensembl gene annotations: gene identifiers and representative transcript cDNA length.
snp_data	Object containing a table with information on population SNPs.
min_n_muts	Numeric vector defining a minimum number of mutations that need to occur at the same residue. Default: 5
MAF_thresh	Numeric vector that defines Minor allele frequency threshold for considering reported mutations as population SNPs.
flanking_region	Numeric vector that defines size of a window around the mutation that will be considered. Up to two window sizes are allowed.

poisson.thr	Numeric vector that defines a treshold for the adjusted p-value. Residues with an associated p-value that is lower than the defined value are reported. Default: 0.01
percentage.thr	Number defining the fraction of mutations within the window that need to fall on a single residue in order for it to be classified as a hotspot. Default: 0.15
ratio.thr	Number defining a requirement that a number of mutations on a single residue should exceed what would be expected by chance given a background mutation rate in the window (i.e. the surrounding region). Default: 45
approach	Option to define selection criteria to use percentage.thr or ratio.thr as criterion for finding single residue mutation clusters. Options: "both", "percentage" or "ratio". Default = "percentage"
write_to_file	Option if the identified and annotated hotspots should be written to a file (YES or NO). Default: NO
ens_release	Release of ensembl to be used. Default: <a href="https://feb2023.archive.ensembl.org">https://feb2023.archive.ensembl.org</a>

**Value**

Results table

**Author(s)**

Marija Buljan <buljan@imsb.biol.ethz.ch> Peter Blattmann <blattmann@imsb.biol.ethz.ch>

**Examples**

```
data("SnpData", package = "DominoEffect")
data("TestData", package = "DominoEffect")
data("DominoData", package = "DominoEffect")

hotspot_mutations <- DominoEffect(mutation_dataset = TestData,
gene_data = DominoData, snp_data = SnpData)
```

---

GPo\_of\_hotspots      *Converts hotspot mutation table into a GPo object*

---

**Description**

This function converts the genomic information on hotspot mutations into a GPo object.

**Usage**

```
GPo_of_hotspots(hotspot_mutations)
```

**Arguments**

hotspot\_mutations

Data frame with information on hotspot mutations generated by the DominoEffect package.

**Value**

GPo object that contains the genomic information on hotspot mutations.

**Author(s)**

Marija Buljan <buljan@imsb.biol.ethz.ch> Peter Blattmann <blattmann@imsb.biol.ethz.ch>

**Examples**

```
data("SnpData", package = "DominoEffect")
data("TestData", package = "DominoEffect")
data("DominoData", package = "DominoEffect")

hotspot_mutations <- DominoEffect(mutation_dataset = TestData,
gene_data = DominoData, snp_data = SnpData)
GPo_of_hotspots(hotspot_mutations)
```

---

identify\_hotspots      *Identify hotspots*

---

**Description**

The function identify protein hotspot mutation residues

**Usage**

```
identify_hotspots(mutation_dataset, gene_data,
snp_data, min_n_muts = 5, MAF_thresh = 0.01, flanking_region = c(200, 300),
poisson.thr = 0.01, percentage.thr = 0.15, ratio.thr = 45, approach = "percentage")
```

**Arguments**

mutation\_dataset

Object containing a table with the mutation data (e.g. TCGA point mutations mapped to protein level).

gene\_data

Data frame or Txdb object containing information about Ensembl gene annotations: gene identifiers and representative transcript cDNA length.

snp\_data

Object containing a table or vcf object with information on population SNPs.

min\_n\_muts

Numeric vector defining a minimum number of mutations that need to occur at the same residue. Default: 5

MAF_thresh	Numeric vector that defines Minor allele frequency threshold for considering reported mutations as population SNPs.
flanking_region	Numeric vector that defines size of a window around the mutation that will be considered. Up to two window sizes are allowed.
poisson.thr	Numeric vector that defines a threshold for the adjusted p-value. Residues with an associated p-value that is lower than the defined value are reported. Default: 0.01
percentage.thr	Number defining the fraction of mutations within the window that need to fall on a single residue in order for it to be classified as a hotspot. Default: 0.15
ratio.thr	Number defining a requirement that a number of mutations on a single residue should exceed what would be expected by chance given a background mutation rate in the window (i.e. the surrounding region). Default: 45
approach	Option to define selection criteria to use percentage.thr or ratio.thr as criterion for finding single residue mutation clusters. Options: "both", "percentage" or "ratio". Default = "percentage"

**Value**

An object containing information on the significant hotspots, associated Gene and protein identifiers, number of mutations, percentage of mutations within defined windows that map to the same residue and associated p-values.

**Author(s)**

Marija Buljan <buljan@imsb.biol.ethz.ch> Peter Blattmann <blattmann@imsb.biol.ethz.ch>

**Examples**

```
data("SnpData", package = "DominoEffect")
data("TestData", package = "DominoEffect")
data("DominoData", package = "DominoEffect")
hotspot_mutations <- identify_hotspots(mutation_dataset = TestData,
  gene_data = DominoData, snp_data = SnpData)
```

---

import_txdb	<i>Imports txdb data and converts it into format required for DominoEffect package</i>
-------------	--

---

**Description**

This function imports txdb data and converts into a data frame used in the DominoEffect package only extracting the required information from the txdb object.

**Usage**

```
import_txdb(txdb_object)
```

**Arguments**

txdb\_object TxDB Object, e.g. from makeTxDbFromEnsembl

**Value**

Data frame that can be used in DominoEffect package.

**Author(s)**

Marija Buljan <buljan@imsb.biol.ethz.ch> Peter Blattmann <blattmann@imsb.biol.ethz.ch>

**Examples**

```
#EnsTxDB <- makeTxDbFromEnsembl(organism="Homo sapiens", release=73,
#                               server="ensemldb.ensembl.org")
#DominoData <- import_txdb(EnsTxDB)
#head(DominoData)
```

---

map\_to\_func\_elem

*Functional annotation of significant hotspot residues.*

---

**Description**

This function retrieves Uniprot annotations for the functional elements in the proteins with significant hotspots and overlaps and maps the hotspot residues to these.

**Usage**

```
map_to_func_elem(hotspot_results, write_to_file = "NO", ens_release = "109")
```

**Arguments**

hotspot\_results Object containing information about the hotspot residues identified using the function identify\_hotspots().

write\_to\_file A character vector defining whether the resulting annotated hotspots should be saved in a file (YES or NO).

ens\_release A character vector defining whether the default gene annotations are used, i.e. Ensembl release 109, or if the gene\_data correspond to a different Ensembl release. For the current Ensembl version this should be set to: ens\_release="www.ensembl.org". For the archive versions to the corresponding archive website.

**Value**

Updated results file containing an additional column with the information on the annotated functional and structural region within which the mutation is mapped.



**Author(s)**

Marija Buljan <buljan@imsb.biol.ethz.ch> Peter Blattmann <blattmann@imsb.biol.ethz.ch>

**Examples**

```
data("TestData", package = "DominoEffect")
data("DominoData", package = "DominoEffect")
data("SnpData", package = "DominoEffect")

hotspot_mutations <- identify_hotspots(TestData, DominoData, SnpData)
hotspot_mutations <- map_to_func_elem(hotspot_mutations,
write_to_file = "NO", ens_release = "109")

head(hotspot_mutations)
```

# Index

`align_to_unip`, 2

`calculate_boundary`, 3

`DominoData`, 4

`DominoEffect`, 4

`GPo_of_hotspots`, 5

`identify_hotspots`, 6

`import_txdb`, 7

`map_to_func_elem`, 8

`SnpData (DominoData)`, 4

`TestData (DominoData)`, 4