

Package ‘GeneticMediation’

April 29, 2020

Type Package

Title An R Package for Causal Mediation Analysis of ROSMAP Data

Version 1.1.0

License GPL-3

URL <https://github.com/tydarnell/GeneticMediation>,
<https://tydarnell.github.io/GeneticMediation>

Author Ty Darnell [aut, cre]

Maintainer Ty Darnell <tydarnell@gmail.com>

Description Provides methods for cleaning and processing data from the ROSMAP study
and preparing it for causal mediation analysis

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

Depends R (>= 3.6.0)

Imports BiocManager,
data.table,
IRanges,
mediation,
readr,
stats

R topics documented:

check_data	2
chr_numeric	2
clean_data	3
combine_data_spc	3
combine_data_spg	3
last_to_first	4
lu	4
make_folder	4
match_all	5
match_snp_peak	5
med.res_spc	5
med.res_spg	6

med_all_spc	6
med_chr_spc	7
med_chr_spg	7
med_summary_spg	8
med_table_spc	8
med_table_spg	9
merge_csv	9
setup_folders	9
snp_peak_bychr	10
sum_zero	10
transpose_readcount	10

Index	11
--------------	-----------

check_data	<i>Check Data</i>
------------	-------------------

Description

Checks if the data is in the data folder

Usage

check_data()

chr_numeric	<i>Chromosome Numeric</i>
-------------	---------------------------

Description

Convert chromosome names from character to numeric. Useful when sorting a dataframe by chromosome number.

Usage

chr_numeric(Chr)

Arguments

Chr a character column or vector of chromosome names

clean_data	<i>Clean Data</i>
------------	-------------------

Description

Clean snp.info, peak.info data and save as data/chipseq.RData

Usage

```
clean_data(snp.path, peak.path)
```

Arguments

snp.path	path to SNP information dataframe
peak.path	path to Peak annotation dataframe

combine_data_spc	<i>Combine Data SNP-Peak-Clinical</i>
------------------	---------------------------------------

Description

SNP-Peak-Clinical data: combine projid, SNPs, PCs, membership, age, gender, peaks, and outcome data for a chromosome

Usage

```
combine_data_spc(chr)
```

Arguments

chr	chromosome name
-----	-----------------

combine_data_spg	<i>Combine Data SNP-Peak-Gene</i>
------------------	-----------------------------------

Description

SNP-Peak-Gene Data: combine projid, SNPs, PCs, membership, age, gender, peaks, gene and outcome data for a chromosome.

Usage

```
combine_data_spg(chr)
```

Arguments

chr	chromosome name
-----	-----------------

last_to_first	<i>Last to First</i>
---------------	----------------------

Description

Make the last column the first column in a dataframe

Usage

```
last_to_first(df)
```

Arguments

df	dataframe
----	-----------

lu	<i>Length Unique</i>
----	----------------------

Description

Get the length of unique values in a vector

Usage

```
lu(x)
```

Arguments

x	vector
---	--------

make_folder	<i>Make Folder</i>
-------------	--------------------

Description

Make a folder only if the folder does not already exist

Usage

```
make_folder(path)
```

Arguments

path	folder path
------	-------------

match_all	<i>Match SNPs Peaks All Chromosomes</i>
-----------	---

Description

Match SNPs and Peaks in all chromosomes and return dataframe of matches

Usage

```
match_all(snp.info, peak.info, chrs)
```

Arguments

snp.info	SNP information
peak.info	Peak information
chrs	character vector of chromosome names

match_snp_peak	<i>Match SNPs and Peaks</i>
----------------	-----------------------------

Description

Match SNPs and Peaks in a chromosome

Usage

```
match_snp_peak(snp.info, peak.info)
```

Arguments

snp.info	SNP information for chr
peak.info	Peak information chr

med.res_spc	<i>SNP-Peak-Clinical Mediation Data Prep</i>
-------------	--

Description

Prepare a chromosome of SNP-Peak-Clinical data for mediation analysis and save as "data/spc.res/res_chr.RData"

Usage

```
med.res_spc(chr, all_matches)
```

Arguments

chr	chromosome name
all_matches	dataframe of SNP-Peak matches for all chromosomes

med.res_spg	<i>SNP-Peak-Gene Mediation Data Prep</i>
-------------	--

Description

Prepare a chromosome SNP-Peak-Gene data for mediation analysis data and save as "data/spg.res/res_chr.RData"

Usage

```
med.res_spg(chr, med.data, matches, mediator.path)
```

Arguments

chr	chromosome name
med.data	gene mediation data
matches	SNP-Peak-Gene matches dataframe

med_all_spc	<i>SNP-Peak-Clinical Mediation Table</i>
-------------	--

Description

Create a table of SNP-Peak-Clinical mediation results for all chromosomes

Usage

```
med_all_spc(chrs, covar, simulations)
```

Arguments

chrs	character vector of chromosome names
covar	covariates string, each covariate separated by +
simulations	number of simulations to run

med_chr_spc	<i>SNP-Peak-Clinical Mediation Table Chromosome</i>
-------------	---

Description

Create SNP-Peak-Clinical mediation table for SNP-Peak matches in a chromosome

Usage

```
med_chr_spc(matches, med.res.data, covar, simulations)
```

Arguments

matches	SNP-Peak matches dataframe for a chromosome
med.res.data	mediation data
covar	covariates string, separate covariates with +
simulations	number of simulations to run

med_chr_spg	<i>SNP-Peak-Gene Mediation Table Chromosome</i>
-------------	---

Description

Create a SNP-Peak-Gene mediation table for a chromosome

Usage

```
med_chr_spg(chr, gene_matches, covar, simulations)
```

Arguments

chr	chromosome name
gene_matches	list of peak-SNP matches for each gene
covar	covariates string, separate covariates with +
simulations	number of simulations to run

med_summary_spg	<i>Mediation Summary SNP-Peak-Gene</i>
-----------------	--

Description

Mediation summary for one SNP-Peak-Gene match

Usage

```
med_summary_spg(med.res.data, match_row, simulations = 1000)
```

Arguments

med.res.data	gene mediation data
match_row	row in match dataframe: col1 SNP, col2 Peak, col3 Gene
simulations	number of simulations, 1000 by default

med_table_spc	<i>SNP-Peak-Clinical Mediation Table</i>
---------------	--

Description

Create SNP-Peak-Clinical mediation table for a SNP-Peak match

Usage

```
med_table_spc(med.res.data, match_row, covar, simulations)
```

Arguments

med.res.data	mediation data
match_row	row in match dataframe, should have 2 columns
covar	covariates character variable, separate covariates with +
simulations	number of simulations to run

med_table_spg	<i>SNP-Peak-Gene Mediation Table</i>
---------------	--------------------------------------

Description

Create a mediation table for one SNP-Peak-Gene match

Usage

```
med_table_spg(med.res.data, match_row, covar, simulations)
```

Arguments

med.res.data	gene mediation data
match_row	row in match dataframe: col1 SNP, col2 Peak, col3 Gene
covar	covariates character variable, separate covariates with +
simulations	number of simulations to run

merge_csv	<i>Merge CSV</i>
-----------	------------------

Description

merge multiple csv files into a single dataframe

Usage

```
merge_csv(path)
```

Arguments

path	path to folder containing csv files
------	-------------------------------------

setup_folders	<i>Setup Folders</i>
---------------	----------------------

Description

Creates data, data/mediator, results folders

Usage

```
setup_folders()
```

snp_peak_bychr	<i>SNPs and Peaks by Chromosome</i>
----------------	-------------------------------------

Description

save snp.info, peak.info for each chromosome as "data/chipseq/chipseq_chr.RData"

Usage

```
snp_peak_bychr(chrs, snps, peaks)
```

Arguments

chrs	Character vector of chromosomes names
snps	SNP information dataframe
peaks	Peak information dataframe

sum_zero	<i>Sum Zero</i>
----------	-----------------

Description

Get names of columns that have a sum of zero

Usage

```
sum_zero(df)
```

Arguments

df	dataframe
----	-----------

transpose_readcount	<i>Transpose Readcount</i>
---------------------	----------------------------

Description

Read in and transpose ChIP-seq Readcount dataframe and match project id

Usage

```
transpose_readcount(readcount.path)
```

Arguments

readcount.path	ChIP-seq readcount dataframe file path
----------------	--

Index

- *Topic **SNP**
 - match_all, [5](#)
 - match_snp_peak, [5](#)
 - *Topic **check**
 - check_data, [2](#)
 - *Topic **chromosome**
 - chr_numeric, [2](#)
 - *Topic **clinical**
 - combine_data_spc, [3](#)
 - *Topic **column**
 - last_to_first, [4](#)
 - *Topic **combine**
 - combine_data_spc, [3](#)
 - combine_data_spg, [3](#)
 - *Topic **csv**
 - merge_csv, [9](#)
 - *Topic **data**
 - check_data, [2](#)
 - clean_data, [3](#)
 - combine_data_spc, [3](#)
 - combine_data_spg, [3](#)
 - med.res_spc, [5](#)
 - med.res_spg, [6](#)
 - snp_peak_bychr, [10](#)
 - *Topic **first**
 - last_to_first, [4](#)
 - *Topic **folder**
 - make_folder, [4](#)
 - setup_folders, [9](#)
 - *Topic **gene**
 - combine_data_spg, [3](#)
 - *Topic **last**
 - last_to_first, [4](#)
 - *Topic **length**
 - lu, [4](#)
 - *Topic **match**
 - match_all, [5](#)
 - match_snp_peak, [5](#)
 - *Topic **mediation**
 - med_all_spc, [6](#)
 - med_chr_spc, [7](#)
 - med_chr_spg, [7](#)
 - med_summary_spg, [8](#)
 - med_table_spc, [8](#)
 - med_table_spg, [9](#)
 - *Topic **merge**
 - merge_csv, [9](#)
 - *Topic **numeric**
 - chr_numeric, [2](#)
 - *Topic **peak**
 - clean_data, [3](#)
 - match_all, [5](#)
 - match_snp_peak, [5](#)
 - *Topic **prep**
 - med.res_spc, [5](#)
 - *Topic **readcount**
 - transpose_readcount, [10](#)
 - *Topic **setup**
 - setup_folders, [9](#)
 - *Topic **snp**
 - clean_data, [3](#)
 - *Topic **summary**
 - med_summary_spg, [8](#)
 - *Topic **table**
 - med_all_spc, [6](#)
 - med_chr_spc, [7](#)
 - med_chr_spg, [7](#)
 - med_table_spc, [8](#)
 - med_table_spg, [9](#)
 - *Topic **transpose**
 - transpose_readcount, [10](#)
 - *Topic **unique**
 - lu, [4](#)
 - *Topic **zero**
 - sum_zero, [10](#)
- check_data, [2](#)
chr_numeric, [2](#)
clean_data, [3](#)
combine_data_spc, [3](#)
combine_data_spg, [3](#)

last_to_first, [4](#)
lu, [4](#)

make_folder, [4](#)
match_all, [5](#)

match_snp_peak, [5](#)
med.res_spc, [5](#)
med.res_spg, [6](#)
med_all_spc, [6](#)
med_chr_spc, [7](#)
med_chr_spg, [7](#)
med_summary_spg, [8](#)
med_table_spc, [8](#)
med_table_spg, [9](#)
merge_csv, [9](#)

setup_folders, [9](#)
snp_peak_bychr, [10](#)
sum_zero, [10](#)

transpose_readcount, [10](#)