Package 'compSPOT'

November 11, 2026

Type Package

Title compSPOT: Tool for identifying and comparing significantly mutated genomic hotspots

Version 1.10.0

Description Clonal cell groups share common mutations within cancer, precancer, and even clinically normal appearing tissues. The frequency and location of these mutations may predict prognosis and cancer risk. It has also been well established that certain genomic regions have increased sensitivity to acquiring mutations. Mutation-sensitive genomic regions may therefore serve as markers for predicting cancer risk. This package contains multiple functions to establish significantly mutated hotspots, compare hotspot mutation burden between samples, and perform exploratory data analysis of the correlation between hotspot mutation burden and personal risk factors for cancer, such as age, gender, and history of carcinogen exposure. This package allows users to identify robust genomic markers to help establish cancer risk.

License Artistic-2.0

Encoding UTF-8

LazyData FALSE

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

biocViews Software, Technology, Sequencing, DNASeq, WholeGenome, Classification, SingleCell, Survival, MultipleComparison

Imports stats, base, ggplot2, plotly, magrittr, ggpubr, gridExtra, utils, data.table

Suggests BiocStyle, knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

URL https://github.com/sydney-grant/compSPOT

BugReports https://github.com/sydney-grant/compSPOT/issues

Config/testthat/edition 3

Depends R (>= 4.3.0)

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

git_branch RELEASE_3_23

git_last_commit 1d761db

git_last_commit_date 2026-04-28

Repository Bioconductor 3.23

Date/Publication 2026-11-11

Author Sydney Grant [aut, cre] (ORCID:
<https://orcid.org/0000-0003-1849-5921>),
Ella Sampson [aut],
Rhea Rodrigues [aut] (ORCID: <https://orcid.org/0000-0002-8573-8658>),
Gyorgy Paragh [aut] (ORCID: <https://orcid.org/0000-0002-6612-9267>)

Maintainer Sydney Grant <Sydney.Grant@roswellpark.org>

Contents

compare_features 2
compare_groups 3
compSPOT 4
compSPOT_example_mutations 5
compSPOT_example_regions 6
find_hotspots 7

Index **9**

`compare_features` *hotspot comparison by additional features*

Description

This function performs an exploratory data analysis comparing the relationship between user-input features to hotspot mutation burden.

Usage

```
compare_features(data, regions, feature)
```

Arguments

`data`	A dataframe containing the clinical features and the mutation count. Dataframe must contain columns with the following names: "Chromosome" <– Chromosome number where the mutation is located "Position" <– Genomic position number where the mutation is located "Sample" <– Unique ID for each sample in dataset "Gene" <– Name of the gene which mutation is located in (optional)

regions	a dataframe containing the chromosome, start and end base pair position of each region of interest
feature	A list containing all the features.

Details

This function is used to classify the features into sequential features if values are numerical or classifies them into #' categorical features. Sequential features are compared to the mutation count using Pearson correlation. Similarly, in categorical features either Wilcox or Kruskal-Wallis test is used to compare between the groups in the features based on the mutational count. Scatter plot is used to represent the sequential features along with the R and p-value from the pearson correlation. Violin plots are used to plot the groups in the categorical data and Wilcox or Kruskal-Wallis values are shown on the graph.

Value

A grid of all the violin plots for the categorical data and scatter plot for the sequential data.

Examples

```
data("compSPOT_example_mutations")
data("compSPOT_example_regions")
features <- c("AGE", "SEX", "ADJUVANT_TX", "SMOKING_HISTORY",
"TUMOR_VOLUME", "KI_67")
compare_features(data = compSPOT_example_mutations,
regions = compSPOT_example_regions, feature = features)
```

---

compare_groups *hotspot comparison by group*

Description

This function compares the mutation frequency of a panel of genomic regions between two subgroups.

Usage

```
compare_groups(data, regions, pvalue, threshold, name1, name2, include_genes)
```

Arguments

data	a dataframe containing the chromosome, base pair position, and optionally gene name of each mutation. Dataframe must contain columns with the following names: "Chromosome" <– Chromosome number where the mutation is located "Position" <– Genomic position number where the mutation is located "Sample" <– Unique ID for each sample in dataset "Gene" <– Name of the gene which mutation is located in (optional) "Group" <– Group classification ID (group.spot only)

regions	a dataframe containing the chromosome, start and end base pair position of each region of interest
pvalue	a threshold p-value for Kolmogorov-Smirnov test
threshold	the cutoff empirical distribution for Kolmogorov-Smirnov test
name1	a string containing the name of one group for the comparison
name2	a string containing the name of the second group for the comparison
include_genes	true or false whether gene names are included in regions dataframe

Details

This function creates a list of mutation frequency per unique sample for each genomic regions separated based on specified sub-groups. The regions with significant differences in mutation distribution are calculated using a Kolmogorov-Smirnov test. The difference in mutation frequency is output in a violin plot.

Value

a list containing the following:

1. A dataframe with the hotspot, group, and mutation count from input sample name
2. A plotly object violin plot comparing the mutation frequency per sample in groups as given by "name1" and "name2" variables
3. An array of ECDF plots comparing the mutation frequency per sample in groups as given by "name1" and "name2" variables

Examples

```
data("compSPOT_example_mutations")
data("compSPOT_example_regions")
compare_groups(data = compSPOT_example_mutations,
regions = compSPOT_example_regions, pvalue = 0.05, threshold = 0.4,
name1 = "High-Risk", name2 = "Low-Risk", include_genes = TRUE)
```

---

compSPOT *Mutation hotspot finder and comparison of hotspot burden.*

Description

It is well known that numerous clones of cells sharing common mutations exist within cancer, pre-cancer, and even clinically normal appearing tissues. The frequency and location of these mutations may aid in the prediction of cancer risk of certain individuals. It has also been well established that certain genomic regions have increased sensitivity to acquiring mutations. Mutation-sensitive genomic regions may therefore be used as markers for prediction of cancer risk. This package contains multiple functions for the establishment of significantly mutated hotspots, comparison of hotspot mutation burden between sub-groups, and exploratory data analysis of the correlation between hotspot mutation burden, and personal risk factors for cancer such as age, gender, and history of carcinogen exposure. This package aims to allow users to identify robust genomic markers which may serve as markers of cancer risk.

Value

A package containing functions which find statistically significant mutation hotspots and compare mutation hotspot burden between groups and correlation between clinical features.

`compSPOT_example_mutations`

Single Nucleotide Variants and Patient Features in Lung Cancer Patients

---

Description

A dataframe containing the chromosome number, base pair location, sample ID, gene name, patient features including: age, sex, adjuvant therapy treatment, smoking history, tumor volume, ki67 quantification, and risk-classification for cancer progression. Data curated from cBioPortal dataset: Non-Small Cell Lung Cancer (TRACERx, NEJM & Nature 2017)

Usage

```
compSPOT_example_mutations
```

Format

`example_mutations`**:**

a dataframe with 11 columns and 22947 rows:

Sample ID assigned to indicate each unique sample

Gene Name of gene affected by mutation

Chromosome Chromosome which the mutation is located on

Position Base pair position of mutation

AGE Age of the patient

SEX Sex of the patient

ADJUVANT_TX Statement of whether or not patient recieved adjuvant therapy

SMOKING_HISTORY Patient's history of smoking

TUMOR_VOLUME Measured volumne of patient's lung tumor

KI_67 Quantification of ki67 markers observed in each patient

Group Risk classification of patients based on observed survival

Details

compSPOT example mutation data

Value

A dataframe containing the chromosome number, base pair location, sample ID, gene name, patient features including: age, sex, adjuvant therapy treatment, smoking history, tumor volume, ki67 quantification, and risk-classification for cancer progression.

References

Abbosh C et al.; TRACERx consortium; PEACE consortium; Swanton C. Phylogenetic ctDNA analysis depicts early-stage lung cancer evolution. Nature. 2017 Apr 26;545(7655):446-451. doi: 10.1038/nature22364. Erratum in: Nature. 2017 Dec 20;: PMID: 28445469; PMCID: PMC5812436.

Jamal-Hanjani M et al.; TRACERx Consortium. Tracking the Evolution of Non-Small-Cell Lung Cancer. N Engl J Med. 2017 Jun 1;376(22):2109-2121. doi: 10.1056/NEJMoa1616288. Epub 2017 Apr 26. PMID: 28445112.

`compSPOT_example_regions`

Genomic Coordinates of Regions of Interest

Description

A dataframe containing the chromosome number, lowerbound and upperbound base pair locations of each region of interest along with the name of the gene where the region is located. Each row indicates a unique region. Regions were identified using the seq.hotSPOT package based on Lung Squamous Cell Carcinoma highly mutated regions.

Usage

```
compSPOT_example_regions
```

Format

`example_regions`**:**

a dataframe with 2 columns and 200 rows:

Lowerbound Base pair position of the start of the region

Upperbound Base pair position of the end of the region

Chromosome Chromosome which the mutation is located on

Gene Name of gene affected by mutation

Details

compSPOT example genomic regions

Value

A dataframe containing the chromosome number, base pair location, and gene names of 200 genomic regions highly mutated in Lung Squamous Cell Carcinoma identified using seq.hotSPOT

References

Grant SR et al; HotSPOT: A Computational Tool to Design Targeted Sequencing Panels to Assess Early Photocarcinogenesis. Cancers (Basel). 2023 Mar 5;15(5):1612. doi: 10.3390/cancers15051612. PMID: 36900402; PMCID: PMC10001346.

Grant S, Wei L, Paragh G (2023). seq.hotSPOT: Targeted sequencing panel design based on mutation hotspots. R package version 1.0.0, https://github.com/sydney-grant/seq.hotSPOT.

`find_hotspots` *significant hotspot calculator*

Description

Based on a panel of genomic regions, this function calculates the regions which are found to have significantly higher mutation frequency compared to less mutated regions.

Usage

```
find_hotspots(data, regions, pvalue, threshold, include_genes, rank)
```

Arguments

`data`	a dataframe containing the chromosome, base pair position, and optionally gene name of each mutation. Dataframe must contain columns with the following names: "Chromosome" <– Chromosome number where the mutation is located "Position" <– Genomic position number where the mutation is located "Sample" <– Unique ID for each sample in dataset "Gene" <– Name of the gene which mutation is located in (optional)
`regions`	a dataframe containing the chromosome, start and end base pair position of each region of interest
`pvalue`	the p-value cutoff for included hotspots
`threshold`	the cutoff empirical distribution for Kolmogorov-Smirnov test
`include_genes`	true or false whether gene names are included in regions dataframe
`rank`	true or false whether regions dataframe is already ranked and includes mutation count of total dataset

Details

This function begins by measuring the mutation frequency for each unique sample for each provided genomic region. Beginning with the top ranked hotspot, a Kolmogorov-Smirnov test is preformed on the mutation frequency of the top genomic region compared to the normalized mutation frequency of all the lower-ranked regions. This continues, then running the Kolmogorov-Smirnov test for the normalized mutation frequency of the top 2 genomic regions compared to the normalized mutation frequency of all lower-ranked regions.This process repeats itself, continuously adding an additional genomic regions each time until either the set p-value or empirical distribution threshold is not met. Once this cutoff has been reached, an established list of mutation hotspots is provided.

Value

A list containing the following:

1. dataframe containing the genomic regions with significant mutation frequency
2. plotly object Dotplot showing the percentage of samples with mutations in each ranked genomic region, highlighting significantly mutated hotspots
3. plotly object ECDF plot showing the difference in mutation frequency between hotspots and non-hotspots

Examples

```
data("compSPOT_example_mutations")
data("compSPOT_example_regions")
significant_spots <- find_hotspots(data = compSPOT_example_mutations,
regions = compSPOT_example_regions,
pvalue = 0.05, threshold = 0.2, include_genes = TRUE, rank = TRUE)
```

Index

∗ **datasets**
compSPOT_example_mutations, 5
compSPOT_example_regions, 6

compare_features, 2
compare_groups, 3
compSPOT, 4
compSPOT_example_mutations, 5
compSPOT_example_regions, 6

find_hotspots, 7