Package 'lisat'

May 8, 2026

Title Longitudinal Integration Site Analysis Toolkit

Version 0.1.2

Description A comprehensive toolkit for the analysis of longitudinal integration site data, including data cleaning, quality control, statistical modeling, and visualization. It streamlines the entire workflow of integration site analysis, supports simple input formats, and provides user-friendly functions for researchers in virus integration site analysis. Ni et al. (2025) <doi:10.64898/2025.12.20.695672>.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.2.3

VignetteBuilder knitr

Depends R (>= 3.5)

Imports GenomicRanges (>= 1.50.0), IRanges (>= 2.32.0), tidyr (>= 1.3.0), dplyr (>= 1.1.4), AnnotationDbi (>= 1.50.0), S4Vectors (>= 0.32.0), GenomicFeatures (>= 1.50.0), magrittr, ggplot2, purrr, broom

Suggests TxDb.Hsapiens.UCSC.hg38.knownGene, org.Hs.eg.db, knitr, rmarkdown, grid, gt (>= 0.9.0), gtable (>= 0.3.6), this.path (>= 2.0.0), plotrix (>= 3.8-2), scales (>= 1.2.0), writexl (>= 1.4.0), ggrepel (>= 0.9.4), ggpubr (>= 0.6.0), viridisLite (>= 0.4.2), RIdeogram (>= 0.2.2), patchwork (>= 1.1.3), RColorBrewer (>= 1.1-3), colorspace, treemapify, igraph, visNetwork, Cairo (>= 1.6-1), testthat (>= 3.0.0)

NeedsCompilation no

Author Shuai Ni [aut, cre]

Maintainer Shuai Ni <Nishuai@wakerbio.com>

Repository CRAN

Date/Publication 2026-03-27 10:40:03 UTC

Contents

`chr_distribution` *Plot chromosome distribution of integration sites (IS)*

Description

Plot chromosome distribution of integration sites (IS)

Usage

```
chr_distribution(IS_raw, ref_version = "random")
```

Arguments

`IS_raw`	Data frame containing raw integration site data (must have Chr column)
`ref_version`	Reference version for simulation (options: 'random' or 'LV', default = 'random')

Value

ggplot object of chromosome distribution (percentage of IS per chromosome)

CIS *Visualize and analyze network of common integration sites (CIS)*

Description

Visualize and analyze network of common integration sites (CIS)

Usage

```
CIS(IS_raw, connect_distance = 50000)
```

Arguments

IS_raw	Data frame containing integration site data (must have Locus, Chr, nearest_gene_name columns)
connect_distance	Numeric threshold for connecting IS (default = 50000 bp)

Value

Data frame with top 10 CIS network metrics (Chr, Locus, Gene, Total_dots, etc.)

CIS_overlap *Generate colored GT table for CIS overlap across samples/timepoints*

Description

Generate colored GT table for CIS overlap across samples/timepoints

Usage

```
CIS_overlap(CIS_data, IS_raw, Timelevels = NULL)
```

Arguments

CIS_data	Data frame of CIS metrics (must have Chr and Locus columns)
IS_raw	Data frame of raw integration site data (must have Sample, Chr, Locus columns)
Timelevels	Optional vector of sample/timepoint levels for ordered display (default = NULL)

Value

gt table object with colored CIS overlap status (TRUE/FALSE)

Count_regions *Calculate regional distribution percentages of integration sites (IS)*

Description

Calculate regional distribution percentages of integration sites (IS)

Usage

```
Count_regions(IS_raw, Patient_timepoint)
```

Arguments

IS_raw	Data frame of raw integration site data (must have Sample column + regional annotation columns)
Patient_timepoint	Data frame mapping Sample_ID to Time_Point (columns: Sample_ID, Time_Point)

Value

List of data frames (per sample) with regional IS percentages (Exonic/Intronic/Enhancer etc.)

Cumulative_curve *Plot cumulative curve and perform statistical analysis*

Description

Plot cumulative curve and perform statistical analysis

Usage

```
Cumulative_curve(IS_ratio)
```

Arguments

IS_ratio	A numeric vector of integration site ratios (output of fit_cum_simple)

Value

A list containing the ggplot object, t-test results, and Wilcoxon test result.

Enhancer_check *Check if integration sites (IS) are located in enhancer regions*

Description

Check if integration sites (IS) are located in enhancer regions

Usage

```
Enhancer_check(IS_raw)
```

Arguments

IS_raw	Data frame containing raw integration site data (must have Chr and Locus columns)

Value

Data frame with an added Enhancer column (TRUE = located in enhancer, FALSE = not located in enhancer)

fit_cum_simple *Calculate normalized cumulative sum for top N elements of a numeric vector*

Description

Calculate normalized cumulative sum for top N elements of a numeric vector

Usage

```
fit_cum_simple(x)
```

Arguments

x	Non-empty numeric vector (integration site ratio data)

Value

Named vector of cumulative sums for predefined target indices + total sum (all = 1)

get_feature *Annotate integration site (IS) data with genomic features*

Description

This function adds genomic feature annotations (gene/exon/intron overlap, nearest gene info) to raw integration site data, standardizes chromosome naming, and calculates clone contribution.

Usage

```
get_feature(IS_raw)
```

Arguments

IS_raw	Data frame containing raw IS data with columns: Sample, Chr, Locus, SCount, Strand

Value

Data frame with annotated genomic features and clone contribution

ideogram_plot *Plot chromosome ideogram with integration site annotations*

Description

This function generates a chromosome ideogram plot showing the density and position of integration sites (IS) using the RIdeogram package.

Usage

```
ideogram_plot(IS_raw, output_dir)
```

Arguments

IS_raw	Data frame containing integration site data (columns: Chr, Locus required)
output_dir	Character, path to output directory for the PDF plot

Value

None (generates a PDF file in output_dir)

is_in_AE_gene *Plot AE-associated gene clone contribution*

Description

This function filters integration site data for AE-associated genes (within specified distance/threshold) and generates a dot plot of clone contribution percentages for these genes.

Usage

```
is_in_AE_gene(IS_raw, Distance = 1e+05, threashold = 0.001)
```

Arguments

IS_raw	Data frame with annotated integration site data (columns: nearest_gene_name, nearest_distance, Clone_contribution, Sample required)
Distance	Numeric, maximum distance to AE gene (default: 100000 bp)
threashold	Numeric, minimum clone contribution threshold (default: 0.001)

Value

ggplot object (dot plot of clone contribution for AE-associated genes)

is_in_CG_gene *Plot Cancer-associated gene clone contribution*

Description

This function filters integration site data for cancer-associated genes (within specified distance/threshold) and generates a dot plot of clone contribution percentages for these genes.

Usage

```
is_in_CG_gene(IS_raw, Distance = 1e+05, threashold = 0.001)
```

Arguments

IS_raw	Data frame with annotated integration site data (columns: nearest_gene_name, nearest_distance, Clone_contribution, Sample required)
Distance	Numeric, maximum distance to cancer gene (default: 100000 bp)
threashold	Numeric, minimum clone contribution threshold (default: 0.001)

Value

ggplot object (dot plot of clone contribution for cancer-associated genes)

is_in_immune_gene *Plot Immune-associated gene clone contribution*

Description

This function filters integration site data for immune-associated genes (within specified distance/threshold) and generates a dot plot of clone contribution percentages for these genes.

Usage

```
is_in_immune_gene(IS_raw, Distance = 1e+05, threashold = 0.001)
```

Arguments

IS_raw	Data frame with annotated integration site data (columns: nearest_gene_name, nearest_distance, Clone_contribution, Sample required)
Distance	Numeric, maximum distance to immune gene (default: 100000 bp)
threashold	Numeric, minimum clone contribution threshold (default: 0.001)

Value

ggplot object (dot plot of clone contribution for immune-associated genes)

IS_treemap *Generate treemap of integration site clone contribution*

Description

This function creates a treemap visualization of the top 1000 integration site (IS) clone contributions, grouped by patient time points with custom color perturbation.

Usage

```
IS_treemap(
  IS_raw = IS_raw,
  Patient_timepoint = Patient_timepoint,
  Timelevels = NULL
)
```

Arguments

IS_raw	Data frame containing IS data (columns: Sample, Locus, Clone_contribution required)
Patient_timepoint	Data frame mapping Sample_ID to Time_Point (columns: Sample_ID, Time_Point required)
Timelevels	Character vector, optional custom order of time points (default: NULL, natural sort)

Value

ggplot object (treemap of IS clone contributions)

Linked_timepoints *Generate Linked Timepoint Sankey + Stacked Bar Chart*

Description

Creates a highly customizable combined Sankey-flow + stacked bar chart to visualize clonal proportion changes across timepoints, with manual control over flow polygon shapes and precise formatting of top integration sites (top 10 + "Others" category). All core logic and data processing steps remain identical to the original code - only namespace prefixes (::) added and lag() fixed.

Usage

```
Linked_timepoints(IS_raw, Patient_timepoint, Timelevels = NULL)
```

Arguments

IS_raw	Data frame containing integration site data (required columns: Clone_contribution, Sample, nearest_gene_name, Chr, Locus)
Patient_timepoint	Data frame mapping Sample_ID to Time_Point (columns: Sample_ID, Time_Point required)
Timelevels	Character vector (optional). Custom ordered levels for time points (overrides natural sort). Default = NULL.

Value

ggplot object. Combined Sankey-flow + stacked bar chart of top 10 integration site proportions across timepoints.

plot_regions *Plot Region-wise Donut Charts*

Description

Plot Region-wise Donut Charts

Usage

```
plot_regions(Region_data, Timelevels = NULL)
```

Arguments

Region_data	Named list of data frames with Product/Share/Percentage/Time columns
Timelevels	Character vector to subset time levels (optional)

Value

Arranged ggplot object of donut charts

plot_richness_evenness *Plot Richness & Evenness Dual Y-Axis Line Chart*

Description

Creates a polished dual Y-axis line chart to visualize clonal richness and evenness over time, with automatic scaling between axes, customizable styling, and optional data labels. All core functionality and parameters remain identical to the original code - only namespace prefixes (::) added.

Usage

```
plot_richness_evenness(
  PMD_data,
  time_col = "Time",
  richness_col = "Richness",
  evenness_col = "Eveness",
  plot_title = "Clonal eveness over time",
  subtitle = NULL,
  richness_color = "#3366CC",
  evenness_color = "#CC6677",
  show_labels = TRUE,
  Timelevels = NULL
)
```

Arguments

PMD_data	Data frame containing time, richness, and evenness data (required columns specified by time_col/richness_col/evenness_col)
time_col	Character (default = "Time"). Name of column containing time points.
richness_col	Character (default = "Richness"). Name of column containing richness values.
evenness_col	Character (default = "Eveness"). Name of column containing evenness values (note: intentional spelling match to original code).
plot_title	Character (default = "Clonal eveness over time"). Main plot title (spelling preserved as original).
subtitle	Character (optional). Plot subtitle (default = NULL).
richness_color	Character (default = "#3366CC"). Hex color code for richness line/points/labels.
evenness_color	Character (default = "#CC6677"). Hex color code for evenness line/points/labels.
show_labels	Logical (default = TRUE). Whether to display numeric labels on data points.
Timelevels	Character vector (optional). Custom ordered levels for time factor (overrides default ordering).

Value

ggplot object. Dual Y-axis line chart of richness (primary) and evenness (secondary) over time.

---

pmd_analysis *Calculate PMD (Proportional Modular Diversity) for integration site data*

Description

This function computes UIS count, top clone contribution percentage, and PMD metrics (Richness/Eveness/PMD) for integration site data, and maps samples to patient time points.

Usage

```
pmd_analysis(IS_raw, Patient_timepoint)
```

Arguments

IS_raw	Data frame containing integration site data (columns: Sample, Clone_contribution required)
Patient_timepoint	Data frame mapping Sample_ID to Time_Point (columns: Sample_ID, Time_Point required)

Value

Data frame with PMD metrics (UIS, TOP_P, Richness, Eveness, PMD, Sample, Time)

pmd_plot — *Generate PMD (Proportional Modular Diversity) Scatter Plot with Inset Legend*

Description

Creates a scatter plot of Richness vs. Eveness for PMD (Proportional Modular Diversity) analysis results, including reference lines, time point labels, and an inset directional legend for polyclonal/monoclonal classification. All core logic and parameters remain identical to the original code - only namespace prefixes (::) are added.

Usage

```
pmd_plot(PMD_data, Timelevels = NULL)
```

Arguments

PMD_data	Data frame output from pmd_analysis() function (required columns: Richness, Eveness, Time)
Timelevels	Character vector (optional). Custom ordered levels for the Time factor. Default = NULL (uses natural sort)

Value

ggplot object. Combined plot (main Richness-Eveness plot + inset legend)

Promotor_check — *Check if integration sites (IS) are located in promoter regions*

Description

Check if integration sites (IS) are located in promoter regions

Usage

```
Promotor_check(IS_raw)
```

Arguments

IS_raw	Data frame containing raw integration site data (must have Chr and Locus columns)

Value

Data frame with an added Promotor column (TRUE = located in promoter, FALSE = not located in promoter)

Safeharbor_check *Check if integration sites (IS) are located in safe harbor regions*

Description

Check if integration sites (IS) are located in safe harbor regions

Usage

```
Safeharbor_check(IS_raw)
```

Arguments

IS_raw	Data frame containing raw integration site data (must have Chr and Locus columns)

Value

Data frame with an added Safeharbor column (TRUE = located in safe harbor, FALSE = not located in safe harbor)

validate_IS_raw *Validate and standardize integration site (IS) raw data frame*

Description

Validate and standardize integration site (IS) raw data frame

Usage

```
validate_IS_raw(IS_raw)
```

Arguments

IS_raw	Data frame containing IS data (expected columns: Sample, SCount, Chr, Locus)

Value

List with validation results:

- valid (logical): TRUE if data passes validation, FALSE otherwise
- errors (character): Validation messages/errors
- converted_data (data.frame): Original/cleaned data with numeric conversions (if applicable)

Index