

# Package ‘Isosceles’

October 31, 2023

**Title** Isoform Single-Cell and Long-read Expression Suite

**Version** 0.0.3

**Description** Transcript detection and quantification from long reads.

**Depends** R (>= 4.2.0),  
SingleCellExperiment (>= 1.18.0)

**Imports** utils (>= 4.2.0),  
methods (>= 4.2.0),  
stats (>= 4.2.0),  
rlang (>= 1.0.4),  
assertthat (>= 0.2.1),  
magrittr (>= 2.0.3),  
tibble (>= 3.1.7),  
tidyselect (>= 1.1.2),  
dplyr (>= 1.0.9),  
tidyr (>= 1.2.0),  
glue (>= 1.6.2),  
digest (>= 0.6.29),  
Rcpp (>= 1.0.9),  
Matrix (>= 1.4-1),  
BiocParallel (>= 1.30.3),  
BiocNeighbors (>= 1.14.0),  
S4Vectors (>= 0.34.0),  
BiocGenerics (>= 0.42.0),  
Biostrings (>= 2.64.0),  
BSgenome (>= 1.64.0),  
GenomeInfoDb (>= 1.32.2),  
IRanges (>= 2.30.0),  
GenomicRanges (>= 1.48.0),  
Rsamtools (>= 2.12.0),  
GenomicAlignments (>= 1.32.1),  
rtracklayer (>= 1.56.1),  
GenomicFeatures (>= 1.48.3),  
SummarizedExperiment (>= 1.26.1),  
DEXSeq (>= 1.42.0),  
igraph (>= 1.3.4),

scuttle ( $\geq 1.6.2$ ),  
 scran ( $\geq 1.24.0$ ),  
 fastmatch ( $\geq 1.1-3$ ),  
 pheatmap ( $\geq 1.0.12$ ),  
 ggbio ( $\geq 1.44.1$ ),  
 ggplot2 ( $\geq 3.3.6$ ),  
 biovizBase ( $\geq 1.44.0$ )

**License** GPL ( $\geq 3$ )

**URL** <https://github.com/timbitz/Isosceles>

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.2.1

**Suggests** testthat ( $\geq 3.0.0$ ),  
 tools ( $\geq 4.2.0$ ),  
 knitr ( $\geq 1.39$ ),  
 rmarkdown ( $\geq 2.14$ ),  
 BiocStyle ( $\geq 2.24.0$ ),  
 viridis ( $\geq 0.6.2$ ),  
 RColorBrewer ( $\geq 1.1-3$ ),  
 dittoSeq ( $\geq 1.8.1$ ),  
 Nebulosa ( $\geq 1.6.0$ ),  
 scater ( $\geq 1.24.0$ ),  
 bluster ( $\geq 1.6.0$ )

**Config/testthat/edition** 3

**LinkingTo** Rcpp ( $\geq 1.0.9$ ),  
 RcppArmadillo ( $\geq 0.11.2.0.0$ )

**VignetteBuilder** knitr

## R topics documented:

Isosceles-package . . . . .	3
bam_to_read_structures . . . . .	3
bam_to_tcc . . . . .	4
calculate_psi_ratio_matrix . . . . .	5
export_gtf . . . . .	6
find_isoswitch . . . . .	6
neighborhood_tcc . . . . .	7
plot_psi_heatmap . . . . .	8
plot_psi_regions . . . . .	8
prepare_transcripts . . . . .	9
pseudobulk_tcc . . . . .	11
pseudotime_tcc . . . . .	11
tcc_to_dexseq . . . . .	12
tcc_to_gene . . . . .	13
tcc_to_transcript . . . . .	13

<i>Isosceles-package</i>	3
transcript_to_psi . . . . .	14
<b>Index</b>	<b>15</b>

---

Isosceles-package	<i>Isosceles: Isoform Single-Cell and Long-read Expression Suite</i>
-------------------	--

---

**Description**

Transcript detection and quantification from long reads

**Author(s)**

Tim Sterne-Weiler [sternewt@gene.com](mailto:sternewt@gene.com)  
Michal Kabza [michal.kabza@contractors.roche.com](mailto:michal.kabza@contractors.roche.com)

---

bam_to_read_structures	<i>Extract read structures from BAM files</i>
------------------------	---

---

**Description**

Extracts non-redundant read structures from one or multiple BAM files.

**Usage**

```
bam_to_read_structures(bam_files, chunk_size = 1e+06, ncpu = 1)
```

**Arguments**

- |            |  |
|------------|--|
| bam_files  | A character vector containing BAM file paths.  |
| chunk_size | An integer scalar specifying the chunk size for reading the BAM files.                 |
| ncpu       | An integer scalar specifying the number of cores to use for multicore parallelization. |

**Value**

A data frame containing non-redundant read structure data obtained from the BAM files.

bam\_to\_tcc

*Prepare a TCC SummarizedExperiment object***Description**

Prepares a TCC (Transcript Compatibility Counts) SummarizedExperiment object for the given BAM files and transcript set.

**Usage**

```
bam_to_tcc(
  bam_files,
  transcript_data,
  run_mode = "strict",
  min_read_count = 1,
  min_relative_expression = 0.1,
  extend_spliced_transcripts = 100,
  is_single_cell = FALSE,
  barcode_tag = "BC",
  chunk_size = 1e+06,
  ncpu = 1
)
```

**Arguments**

bam_files	A named character vector containing BAM file paths.
transcript_data	A named list containing transcript data returned by the <a href="#">prepare_transcripts</a> function.
run_mode	A string specifying the mode for choosing the transcript set ('strict', 'de_novo_strict', 'de_novo_loose' or 'de_novo_full').
min_read_count	An integer scalar specifying the read count threshold for transcripts extracted from the BAM files.
min_relative_expression	A numeric scalar specifying the relative expression threshold for transcripts extracted from the BAM files.
extend_spliced_transcripts	An integer scalar specifying the number of base pairs by which transcript starts and ends are extended for spliced read compatibility search.
is_single_cell	A logical scalar specifying if the BAM files contain single cell data.
barcode_tag	A string specifying the name of the BAM file tag containing cell barcodes.
chunk_size	An integer scalar specifying the chunk size for reading the BAM files.
ncpu	An integer scalar specifying the number of cores to use for multicore parallelization.

**Value**

A SummarizedExperiment object containing TCC annotation and quantification data.

---

```
calculate_psi_ratio_matrix
```

*Calculate PSI count to mean permuted PSI count ratio matrix*

---

**Description**

Calculates PSI count to mean permuted PSI count ratio matrix for pseudotime window data. This function is designed for preparing data to be visualized as a heatmap, and might take a long time to run - see the vignettes for an example.

**Usage**

```
calculate_psi_ratio_matrix(  
  se_tcc,  
  pseudotime_matrix,  
  psi_events,  
  window_sizes,  
  window_steps,  
  trim = 0,  
  n_perm = 100,  
  ncpu = 1  
)
```

**Arguments**

se_tcc	A TCC SummarizedExperiment object returned by the <a href="#">bam_to_tcc</a> function.
pseudotime_matrix	A numeric matrix containing the pseudotime values for each cell (rows) in different trajectories (columns). Cells not belonging to given trajectory should be denoted using NA values.
psi_events	A character vector specifying the PSI events to calculate the ratios for.
window_sizes	A named integer vector specifying the window size for each trajectory.
window_steps	A named integer vector specifying the window step for each trajectory.
trim	A numeric scalar specifying the fraction (0 to 0.5) of cells to be trimmed from each end of the pseudotime spectrum for each trajectory.
n_perm	An integer scalar specifying the number of PSI count permutations to calculate.
ncpu	An integer scalar specifying the number of cores to use for multicore parallelization.

**Value**

A numeric matrix containing the PSI count to mean permuted PSI count ratio values.

---

export_gtf	<i>Export data to a GTF file</i>
------------	----------------------------------

---

### Description

Exports transcripts from a SummarizedExperiment object to a GTF file.

### Usage

```
export_gtf(se, file)
```

### Arguments

se	A transcript-level SummarizedExperiment object returned by the <a href="#">tcc_to_transcript</a> function.
file	A string specifying the output file path.

### Value

Nothing is returned.

---

find_isoswitch	<i>Find isoform switching events</i>
----------------	--------------------------------------

---

### Description

Identifies isoform switching events by comparing every pair of cell groups using the [findMarkers](#) function from the [scran](#) package and searching for transcripts of the same gene showing statistically significant differences in opposite directions.

### Usage

```
find_isoswitch(se, cell_labels, min_fdr = 0.05, ncpu = 1)
```

### Arguments

se	A transcript-level SummarizedExperiment object returned by the <a href="#">tcc_to_transcript</a> function. The object must contain normalized data stored in the 'logcounts' assay, which can be prepared using functions from the <a href="#">scuttle</a> package.
cell_labels	A vector or a factor containing cell labels acting as a grouping variable.
min_fdr	A numeric scalar specifying the FDR threshold for filtering the results.
ncpu	An integer scalar specifying the number of cores to use for multicore parallelization.

**Value**

A data frame containing the following columns:

**transcript\_id** Isosceles transcript ID

**compatible\_tx** comma-separated list of annotated transcript IDs compatible with the Isosceles transcript

**gene\_id** gene ID

**gene\_name** gene symbol

**pvalue** p-value from the Wilcoxon test performed by the findMarkers function

**fdr** false discovery rate (FDR) value from the Wilcoxon test performed by the findMarkers function

**auc** area under the curve (AUC) value from the Wilcoxon test performed by the findMarkers function

**group\_1** label of the cell group in which the transcript is upregulated

**group\_2** label of the cell group compared to which the transcript is upregulated

**contrast** label of the compared cell group pair

---

neighborhood\_tcc

---

*Merge the neighboring cell TCC values in scRNA-Seq data*


---

**Description**

Prepares a TCC SummarizedExperiment object where count values from the nearest k neighbors are added to the count values of each cell.

**Usage**

```
neighborhood_tcc(se_tcc, pca_mat, k = 10, use_annoy = FALSE, ncpu = 1)
```

**Arguments**

se_tcc	A TCC SummarizedExperiment object returned by the <a href="#">bam_to_tcc</a> function.
pca_mat	A matrix containing PCA coordinates of each cell.
k	An integer scalar specifying the number of nearest neighbors to use.
use_annoy	A logical scalar indicating whether to use the Annoy algorithm for approximate nearest neighbor identification (recommended for big datasets).
ncpu	An integer scalar specifying the number of cores to use for multicore parallelization.

**Value**

A SummarizedExperiment object containing merged TCC data.

---

plot_psi_heatmap	<i>Plot a PSI heatmap</i>
------------------	---------------------------

---

### Description

Creates a heatmap of PSI (Percent Spliced In) values for the regions of a given gene across samples or cells.

### Usage

```
plot_psi_heatmap(
  se_psi,
  gene_id,
  heatmap_colors = viridis::cividis(100),
  region_colors = NULL,
  ...
)
```

### Arguments

se_psi	A PSI SummarizedExperiment object returned by the <a href="#">transcript_to_psi</a> function.
gene_id	A string containing the identifier of the gene to plot.
heatmap_colors	A character vector containing the color palette used in the heatmap.
region_colors	A named character vector of colors for the region type annotations.
...	Additional parameters for the plot, passed to the <a href="#">pheatmap</a> function.

### Value

A plot object.

---

plot_psi_regions	<i>Plot PSI regions</i>
------------------	-------------------------

---

### Description

Creates a plot showing PSI regions and transcript structures for the given gene. Individual transcript structures are colored by their relative expression, calculated from the overall TPM values and expressed in percentages. For better visualization, introns can be shrunk using the `max_intron_length` argument.



**Usage**

```
plot_psi_regions(
  se_psi,
  se_transcript,
  gene_id,
  max_transcripts = Inf,
  max_intron_length = NULL,
  region_colors = NULL
)
```

**Arguments**

<code>se_psi</code>	A PSI SummarizedExperiment object returned by the <a href="#">transcript_to_psi</a> function.
<code>se_transcript</code>	A transcript-level SummarizedExperiment object returned by the <a href="#">tcc_to_transcript</a> function.
<code>gene_id</code>	A string containing the identifier of the gene to plot.
<code>max_transcripts</code>	An integer scalar specifying the maximum number of transcripts with the highest relative expression to plot.
<code>max_intron_length</code>	An integer scalar specifying the maximum intron length after shrinking. If set to NULL, no shrinking is performed.
<code>region_colors</code>	A named character vector of colors for the PSI region types.

**Value**

A plot object.

---

prepare_transcripts	<i>Prepare transcript data for the analysis</i>
---------------------	---

---

**Description**

Prepares transcript data (reference and extracted from the BAM files) for further analysis.

**Usage**

```
prepare_transcripts(
  gtf_file,
  genome_fasta_file,
  bam_parsed,
  min_intron_length = 30,
  known_intron_motifs = c("GT-AG"),
  rescue_annotated_introns = FALSE,
```

```

    known_intron_granges = NULL,
    min_bam_splice_read_count = 2,
    min_bam_splice_fraction = 0.1,
    bin_size = 50
  )

```

## Arguments

<code>gtf_file</code>	A string containing a GTF file path.
<code>genome_fasta_file</code>	A string containing a genome FASTA file path.
<code>bam_parsed</code>	A data frame containing non-redundant read structure data returned by the <a href="#">bam_to_read_structures</a> function. If NULL, only reference transcripts are used.
<code>min_intron_length</code>	An integer scalar specifying the minimal length of introns to assign strand to.
<code>known_intron_motifs</code>	A character vector specifying the known intron motifs.
<code>rescue_annotated_introns</code>	A logical scalar specifying if introns found in genome annotations should be kept even if they don't have known intron motifs.
<code>known_intron_granges</code>	A GRanges object storing known intron positions (e.g. from short read data) used for transcript classification. If set to NULL, only introns from reference annotations are used.
<code>min_bam_splice_read_count</code>	An integer scalar specifying the read count threshold for splice sites confirmed by aligned reads.
<code>min_bam_splice_fraction</code>	A numeric scalar specifying the minimum connectivity fraction to a known splice site for splice sites confirmed by aligned reads.
<code>bin_size</code>	An integer scalar specifying the bin size for transcript start and end position binning.

## Value

A named list containing following elements:

**tx\_df** a data frame storing extracted transcript data

**tx\_granges** a GRanges object storing genomic positions of extracted transcript

**tx\_exon\_granges\_list** a GRangesList object storing exon genomic positions of extracted transcript

**tx\_intron\_granges\_list** a GRangesList object storing intron genomic positions of extracted transcript

---

pseudobulk_tcc	<i>Prepare a pseudobulk TCC SummarizedExperiment object</i>
----------------	---

---

**Description**

Prepares a pseudobulk TCC SummarizedExperiment from TCC data and given cell labels.

**Usage**

```
pseudobulk_tcc(se_tcc, cell_labels)
```

**Arguments**

se_tcc	A TCC SummarizedExperiment object returned by the <a href="#">bam_to_tcc</a> function.
cell_labels	A vector or a factor containing cell labels acting as a grouping variable.

**Value**

A pseudobulk SummarizedExperiment object containing TCC annotation and quantification data.

---

pseudotime_tcc	<i>Merge TCC values using moving window over pseudotime</i>
----------------	---

---

**Description**

Prepares a pseudotime window TCC SummarizedExperiment from TCC data and pseudotime values.

**Usage**

```
pseudotime_tcc(
  se_tcc,
  pseudotime,
  trim = 0,
  window_size = 30,
  window_step = 15
)
```

**Arguments**

se_tcc	A TCC SummarizedExperiment object returned by the <a href="#">bam_to_tcc</a> function.
pseudotime	A numeric vector containing the pseudotime values for each cell. Cells not belonging to the analyzed trajectory should be denoted using NA values.
trim	A numeric scalar specifying the fraction (0 to 0.5) of cells to be trimmed from each end of the pseudotime spectrum.
window_size	An integer scalar specifying the window size.
window_step	An integer scalar specifying the window step.

**Value**

A SummarizedExperiment object containing TCC data for pseudotime windows.

---

tcc_to_dexseq	<i>Prepare a PSI count DEXSeqDataSet object</i>
---------------	---

---

**Description**

Aggregates TCC values using pseudotime windows and creates a DEXSeqDataSet object suitable for the analysis of PSI count changes along given pseudotime trajectory.

**Usage**

```
tcc_to_dexseq(
  se_tcc,
  pseudotime,
  psi_events = NULL,
  trim = 0,
  window_size = 30,
  window_step = 15,
  remove_redundant_psi = TRUE,
  scale_pseudotime = TRUE,
  ncpu = 1
)
```

**Arguments**

se_tcc	A TCC SummarizedExperiment object returned by the <a href="#">bam_to_tcc</a> function.
pseudotime	A numeric vector containing the pseudotime values for each cell. Cells not belonging to the analyzed trajectory should be denoted using NA values.
psi_events	A character vector specifying the PSI events to restrict the analysis to (ignored if set to NULL).
trim	A numeric scalar specifying the fraction (0 to 0.5) of cells to be trimmed from each end of the pseudotime spectrum.
window_size	An integer scalar specifying the window size.
window_step	An integer scalar specifying the window step.
remove_redundant_psi	A logical scalar specifying if PSI events with redundant count profiles should be removed from the analysis.
scale_pseudotime	A logical scalar specifying if pseudotime values for the windows should be scaled.
ncpu	An integer scalar specifying the number of cores to use for multicore parallelization.

**Value**

A DEXSeqDataSet object containing PSI count data for pseudotime windows, suitable for further analysis using the DEXSeq package.

---

tcc_to_gene	<i>Prepare a gene-level SummarizedExperiment object</i>
-------------	---

---

**Description**

Prepares a gene-level SummarizedExperiment from TCC data.

**Usage**

```
tcc_to_gene(se_tcc)
```

**Arguments**

se\_tcc            A TCC SummarizedExperiment object returned by a function from the [Isosceles-package](#).

**Value**

A SummarizedExperiment object containing gene annotation and quantification data.

---

tcc_to_transcript	<i>Prepare a transcript-level SummarizedExperiment object</i>
-------------------	---

---

**Description**

Prepares a transcript-level SummarizedExperiment from TCC data using the EM algorithm.

**Usage**

```
tcc_to_transcript(
  se_tcc,
  em.maxiter = 250,
  em.conv = 0.01,
  use_length_normalization = TRUE,
  ncpu = 1
)
```

**Arguments**

se_tcc	A TCC SummarizedExperiment object returned by a function from the <a href="#">Isosceles-package</a> .
em.maxiter	An integer scalar specifying the maximum number of EM iterations.
em.conv	A numeric scalar specifying the EM convergence threshold.
use_length_normalization	A logical scalar specifying if normalization using effective transcript lengths should be used during EM.
ncpu	An integer scalar specifying the number of cores to use for multicore parallelization.

**Value**

A SummarizedExperiment object containing transcript annotation and quantification data.

---

transcript_to_psi	<i>Prepare a PSI SummarizedExperiment object</i>
-------------------	--

---

**Description**

Prepares a PSI (Percent Spliced In) SummarizedExperiment object for the given transcript-level SummarizedExperiment object. PSI values are calculated for the following types of regions:

- **TSS** - transcription start sites
- **TES** - transcription end sites
- **CE** - core exonic regions
- **RI** - retained intronic regions
- **A5** - 5' alternative exonic regions
- **A3** - 3' alternative exonic regions

TSS and TES positions are calculated based on transcripts' binned start and end coordinates extracted from their identifiers.

**Usage**

```
transcript_to_psi(se, ncpu = 1)
```

**Arguments**

se	A transcript-level SummarizedExperiment object returned by the <a href="#">tcc_to_transcript</a> function.
ncpu	An integer scalar specifying the number of cores to use for multicore parallelization.

**Value**

A SummarizedExperiment object containing PSI annotation and quantification data.

# Index

bam\_to\_read\_structures, [3](#), [10](#)

bam\_to\_tcc, [4](#), [5](#), [7](#), [11](#), [12](#)

calculate\_psi\_ratio\_matrix, [5](#)

export\_gtf, [6](#)

find\_isoswitch, [6](#)

findMarkers, [6](#)

Isosceles-package, [3](#)

neighborhood\_tcc, [7](#)

pheatmap, [8](#)

plot\_psi\_heatmap, [8](#)

plot\_psi\_regions, [8](#)

prepare\_transcripts, [4](#), [9](#)

pseudobulk\_tcc, [11](#)

pseudotime\_tcc, [11](#)

tcc\_to\_dexseq, [12](#)

tcc\_to\_gene, [13](#)

tcc\_to\_transcript, [6](#), [9](#), [13](#), [14](#)

transcript\_to\_psi, [8](#), [9](#), [14](#)