Package: TCRconvertR (via r-universe)

October 19, 2026

Type Package

Title Convert TCR Gene Names

Description Convert T Cell Receptor (TCR) gene names between the 10X Genomics, Adaptive Biotechnologies, and ImMunoGeneTics (IMGT) nomenclatures.

Version 1.0

License MIT + file LICENSE

URL https://github.com/seshadrilab/tcrconvertr, https://seshadrilab.github.io/tcrconvertr/

BugReports https://github.com/seshadrilab/tcrconvertr/issues

Roxygen list(markdown = TRUE, roclets = c(``collate", ``namespace", ``rd", ``roxyglobals::global_roclet"))

Encoding UTF-8

Imports stats, utils, rappdirs

RoxygenNote 7.3.3

Suggests knitr, rmarkdown, roxyglobals, testthat (>= 3.0.0), mockery

Config/testthat/edition 3

Config/roxyglobals/filename globals.R

Config/roxyglobals/unique FALSE

VignetteBuilder knitr

Repository https://seshadrilab.r-universe.dev

Date/Publication 2026-05-15 04:21:52 UTC

RemoteUrl https://github.com/seshadrilab/tcrconvertr

RemoteRef HEAD

RemoteSha 3465ddbff2fab7bd648b895d39a597f6a31f7fca

Contents

build_lookup_from_fastas 2
convert_gene 3
get_example_path 5

Index **6**

`build_lookup_from_fastas` *Create lookup tables*

Description

`build_lookup_from_fastas()` processes IMGT reference FASTA files in a given folder to generate lookup tables used for making gene name conversions. It extracts all gene names and transforms them into 10X and Adaptive formats following predefined conversion rules. The resulting files are created:

- `lookup.csv`: IMGT gene names and their 10X and Adaptive equivalents.
- `lookup_from_tenx.csv`: Gene names aggregated by their 10X identifiers, with one representative allele (`*01`) for each.
- `lookup_from_adaptive.csv`: Adaptive gene names, with or without alleles and gene designations, and their IMGT and 10X equivalents.

The files are stored in a given subfolder (`species`) within the appropriate application folder via `rappdirs`. For example:

- MacOS: `~/Library/Application Support/<AppName>`
- Windows: `C:\Documents and Settings\<User>\Application Data\Local Settings\<AppAuthor>\<AppName>`
- Linux: `~/.local/share/<AppName>`

If a folder named `species` already exists in that location, it will be replaced.

Usage

```
build_lookup_from_fastas(data_dir, species)
```

Arguments

`data_dir`	A string, the directory containing FASTA files.
`species`	A string, the name of species that will be used when running TCRconvert with these lookup tables.

Details

Key transformations from IMGT:

- **10X:**
  - Remove allele information (e.g., `*01`) and modify `/DV` occurrences.
- **Adaptive:**
  - Apply renaming rules, such as adding gene-level designations and zero-padding single-digit numbers.
  - Convert constant genes to `"NoData"` (Adaptive only captures VDJ) which become `NA` after the merge in `convert_gene()`.

Value

A string, path to new lookup directory

Examples

```
# For the example, create and use a temporary folder
fastadir <- file.path(tempdir(), "TCRconvertR_tmp")
dir.create(fastadir, showWarnings = FALSE, recursive = TRUE)
trav <- get_example_path("fasta_dir/test_trav.fa")
trbv <- get_example_path("fasta_dir/test_trbv.fa")
file.copy(c(trav, trbv), fastadir)

# Build lookup tables
build_lookup_from_fastas(fastadir, "rabbit")

# Clean up temporary folder
unlink(fastadir, recursive = TRUE)
```

`convert_gene` *Convert gene names*

Description

`convert_gene()` converts T-cell receptor (TCR) gene names between the IMGT, 10X, and Adaptive formats. It determines the columns to convert based on the input format (`frm`) unless specified by the user (`frm_cols`). It returns a modified version of the input data frame with converted gene names while preserving row order.

Usage

```
convert_gene(
  df,
  frm,
  to,
  species = "human",
```

```
  frm_cols = NULL,
  verbose = TRUE,
  bad_genes_col = FALSE
)
```

Arguments

df	A dataframe containing TCR gene names.
frm	A string, the input format of TCR data. Must be one of "imgt", "tenx", "adaptive", or "adaptivev2".
to	A string, the output format of TCR data. Must be one of "imgt", "tenx", "adaptive", or "adaptivev2".
species	A string,the species. Optional; defaults to "human".
frm_cols	A character vector of custom gene column names. Optional; defaults to NULL.
verbose	A boolean, whether to display messages. Optional; defaults to TRUE.
bad_genes_col	A boolean, whether to add a column of the unconvertable genes. Defaults to FALSE.

Details

Gene names are converted by performing a merge between the relevant input columns and a species-specific lookup table containing IMGT reference genes in all three formats.

Behavioral Notes

- If a gene name cannot be mapped, it is replaced with NA and a warning is raised.
- If bad_genes_col = TRUE, appends a 'bad_genes' column containing comma-separated gene names that could not be converted for each row.
- If frm is 'imgt' and frm_cols is not provided, 10X column names are assumed.
- Constant (C) genes are set to NA when converting to Adaptive formats, as Adaptive does not capture constant regions.
- The input does not need to include all gene types; partial inputs (e.g., only V genes) are supported.
- If no values in a custom column can be mapped (e.g. a CDR3 column) it is skipped and a warning is raised.

Standard Column Names

If frm_cols is not provided, these column names will be used if present:

- **IMGT**: "v_gene", "d_gene", "j_gene", "c_gene"
- **10X**: "v_gene", "d_gene", "j_gene", "c_gene"
- **Adaptive**: "v_resolved", "d_resolved", "j_resolved"
- **Adaptive v2**: "vMaxResolved", "dMaxResolved", "jMaxResolved"

Value

A dataframe with converted TCR gene names.

Examples

```
tcr_file <- get_example_path("tenx.csv")
df <- read.csv(tcr_file)[c("barcode", "v_gene", "j_gene", "cdr3")]
df
convert_gene(df, "tenx", "adaptive", verbose = FALSE)
```

`get_example_path` *Get full path to an example file or directory*

Description

`get_example_path()` takes a file or folder name that is expected to be located under the `TCRconvertR` `examples` directory and gets the full path to that item.

Usage

```
get_example_path(file_name)
```

Arguments

`file_name`	A string, the name of the example file or directory.

Value

A string, the path to example file or directory.

Examples

```
# Will probably be in a temp folder for the function example
get_example_path("tenx.csv")
```

Index

build_lookup_from_fastas, 2

convert_gene, 3

get_example_path, 5