

# Package: MAAPER (via r-universe)

October 11, 2024

**Title** Analysis of Alternative Polyadenylation Using 3' End-Linked Reads

**Version** 1.1.1

**Description** A computational method developed for model-based analysis of alternative polyadenylation (APA) using 3' end-linked reads. It accurately assigns 3' RNA-seq reads to polyA sites through statistical modeling, and generates multiple statistics for APA analysis. Please also see Li WV, Zheng D, Wang R, Tian B (2021) <[doi:10.1186/s13059-021-02429-5](https://doi.org/10.1186/s13059-021-02429-5)>.

**License** GPL-3

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.1.1

**Imports** parallel, GenomicRanges, GenomicAlignments, GenomicFeatures, GenomeInfoDb, stats, utils, Rsamtools, IRanges, MASS

**URL** <https://github.com/Vivianstats/MAAPER>,  
<https://genomebiology.biomedcentral.com/articles/10.1186/s13059-021-02429-5>

**Suggests** knitr, rmarkdown

**VignetteBuilder** knitr

**Repository** <https://vivianstats.r-universe.dev>

**RemoteUrl** <https://github.com/vivianstats/maaper>

**RemoteRef** HEAD

**RemoteSha** 907b16cb8c3c27c2fab506b187fb61a01ca81ff9

## Contents

maaper . . . . .	2
<b>Index</b>	<b>4</b>

---

maaper	<i>Model-based analysis of alternative polyadenylation using 3' end-linked reads</i>
--------	--

---

## Description

Model-based analysis of alternative polyadenylation using 3' end-linked reads

## Usage

```
maaper(
  gtf,
  pas_annotation,
  output_dir,
  bam_c1,
  bam_c2,
  read_len,
  ncores = 1,
  num_pas_thre = 25,
  frac_pas_thre = 0.05,
  dist_thre = 600,
  num_thre = 50,
  run = "all",
  subset = NULL,
  region = "all",
  gtf_rds = NULL,
  verbose = FALSE,
  paired = FALSE,
  bed = FALSE
)
```

## Arguments

gtf	A character specifying the full path of the GTF file (reference genome);
pas_annotation	A list containing the pas annotation. MAAPER provides processed annotation information from PolyA_DB v3 on its Github page.
output_dir	A character specifying the full path of the output directory, which is used to store all intermediate and final outputs.
bam_c1	A character vector specifying the full paths to the bam files for condition 1 (control). The length of the vector equals the number of samples.
bam_c2	A character vector specifying the full paths to the bam files for condition 2 (experiment). The length of the vector equals the number of samples.
read_len	An integer specifying the read length.
ncores	An integer specifying the number of cores used in parallel computation.
num_pas_thre	An integer specifying the threshold on PAS's read number. Defaults to 25.

frac_pas_thre	A numeric specifying the threshold on PAS's fraction. Defaults to 0.05.
dist_thre	An integer specifying the threshold on fragment length. Defaults to 600.
num_thre	An integer specifying the threshold on gene's read number. Defaults to 50.
run	"all" (default) or "skip-train". For test and debug only.
subset	A character vector specifying genes' Ensembl IDs if only a subset of genes need to be analyzed. Check the pas_annotation files for ID formats.
region	"all" (default). For test and debug only.
gtf_rds	NULL (default). For test and debug only.
verbose	FALSE (default). For test and debug only.
paired	A boolean indicating whether to perform paired test instead of unpaired test (defaults to FALSE).
bed	A boolean indicating whether bedGraph files should be output for visualization in genome browser.

### Value

maaper saves two text files, gene.txt and pas.txt, to out\_dir. pas.txt contains the gene names, predicted PASs, and their corresponding fractions in the two conditions. gene.txt contains the genes' PAS number, p values, RED, RLDu, and RLDi scores.

### Author(s)

Wei Vivian Li, <vivian.li@rutgers.edu>

### Examples

```
## Not run:
# data used in this example can be found on the package's Github page
pas_annotation = readRDS("./mouse.PAS.mm9.rds")
gtf = "./gencode.mm9.chr19.gtf"
bam_c1 = "./NT_chr19_example.bam"
bam_c2 = "./AS_4h_chr19_example.bam"
maaper(gtf, pas_annotation, output_dir = "./",
       bam_c1, bam_c2, read_len = 76, ncores = 1)

## End(Not run)
```

# Index

maaper, [2](#)