

# Package ‘Rbwa’

December 14, 2024

**Version** 1.11.0

**Date** 2023-06-29

**Title** R wrapper for BWA-backtrack and BWA-MEM aligners

**Depends** R (>= 4.1)

**Suggests** testthat, BiocStyle, knitr, rmarkdown

**SystemRequirements** GNU make

**Description** Provides an R wrapper for BWA alignment algorithms.

Both BWA-backtrack and BWA-MEM are available.

Convenience function to build a BWA index from a reference genome is also provided. Currently not supported for Windows machines.

**License** MIT + file LICENSE

**URL** <https://github.com/Jfortin1/Rbwa>

**BugReports** <https://github.com/crisprVerse/Rbwa/issues>

**biocViews** Sequencing, Alignment

**VignetteBuilder** knitr

**RoxygenNote** 7.1.2

**OS\_type** unix

**git\_url** <https://git.bioconductor.org/packages/Rbwa>

**git\_branch** devel

**git\_last\_commit** 441c698

**git\_last\_commit\_date** 2024-10-29

**Repository** Bioconductor 3.21

**Date/Publication** 2024-12-13

**Author** Jean-Philippe Fortin [aut, cre]

**Maintainer** Jean-Philippe Fortin <fortin946@gmail.com>

## Contents

bwa_aln . . . . .	2
bwa_build_index . . . . .	3
bwa_mem . . . . .	4
bwa_sam . . . . .	5
xa2multi . . . . .	6

<b>Index</b>	<b>8</b>
--------------	----------

---

bwa_aln	<i>R wrapper to run BWA alignment tool BWA-backtrack</i>
---------	----------------------------------------------------------

---

### Description

R wrapper to run BWA alignment tool BWA-backtrack.

### Usage

```
bwa_aln(
  type = c("single", "paired"),
  index_prefix,
  fastq_files,
  sai_files,
  ...
)
```

### Arguments

type	String specifying type of reads: "single" for single-end reads (default) or "paired" for paired-end reads.
index_prefix	String specifying prefix of the BWA index.
fastq_files	Character vector specifying paths of fastq files. If type=="single", must be of length 1. If type=="paired", must be of length 2.
sai_files	Character vector specifying filenames of the BWA alignment output files. If type=="single", must be of length 1. If type=="paired", must be of length 2.
...	Other arguments to pass to the bwa aln alignment.

### Value

No return value. Output files from bwa aln alignment are produced as side effect.

### Author(s)

Jean-Philippe Fortin

**Examples**

```
dir <- tempdir()
fasta <- system.file(package="Rbwa",
                     "fasta/chr12.fa")
fastq <- system.file(package="Rbwa",
                     "fastq/sequences.fastq")
index_prefix <- file.path(dir, "chr12")
bwa_build_index(fasta, index_prefix=index_prefix)

bwa_aln(index_prefix=index_prefix,
         fastq_files=fastq,
         sai_files=file.path(dir, "output.sai"))
```

---

bwa_build_index	<i>R wrapper to create BWA index files</i>
-----------------	--------------------------------------------

---

**Description**

R wrapper to create BWA index files from a FASTA file.

**Usage**

```
bwa_build_index(fasta, index_prefix = NULL, ...)
```

**Arguments**

fasta	String specifying path to a FASTA file.
index_prefix	String specifying prefix of the output BWA index.
...	Other arguments to pass to bwa index.

**Value**

No return value. BWA index files are produced as a side-effect.

**Author(s)**

Jean-Philippe Fortin

**Examples**

```
dir <- tempdir()
fasta <- system.file(package="Rbwa",
                     "fasta/chr12.fa")
bwa_build_index(fasta,
                index_prefix=file.path(dir, "chr12"))
```

---

`bwa_mem`*R wrapper to run BWA alignment tool BWA-MEM*

---

**Description**

R wrapper to run BWA alignment tool BWA-MEM.

**Usage**

```
bwa_mem(type = c("single", "paired"), index_prefix, fastq_files, sam_file, ...)
```

**Arguments**

<code>type</code>	String specifying type of reads: "single" for single-end reads (default) or "paired" for paired-end reads.
<code>index_prefix</code>	String specifying prefix of the BWA index.
<code>fastq_files</code>	Character vector specifying paths of fastq files. If <code>type=="single"</code> , must be of length 1. If <code>type=="paired"</code> , must be of length 2.
<code>sam_file</code>	String specifying filename of the SAM alignment output.
<code>...</code>	Other arguments to pass to the <code>bwa aln</code> alignment.

**Value**

No return value. Output SAM file is produced as side effect.

**Author(s)**

Jean-Philippe Fortin

**Examples**

```
dir <- tempdir()
fasta <- system.file(package="Rbwa",
                    "fasta/chr12.fa")
fastq <- system.file(package="Rbwa",
                    "fastq/sequences.fastq")
index_prefix <- file.path(dir, "chr12")
bwa_build_index(fasta, index_prefix=index_prefix)

bwa_mem(index_prefix=index_prefix,
        fastq_files=fastq,
        sam_file=file.path(dir, "output.sam"))
```

---

bwa_sam	<i>R wrapper to convert bwa aln output to SAM format</i>
---------	----------------------------------------------------------

---

**Description**

R wrapper to convert bwa aln output to SAM format.

**Usage**

```
bwa_sam(
  type = c("single", "paired"),
  index_prefix,
  fastq_files,
  sai_files,
  sam_file,
  ...
)
```

**Arguments**

type	String specifying type of reads: "single" for single-end reads (default) or "paired" for paired-end reads.
index_prefix	String specifying prefix of the BWA index.
fastq_files	Character vector specifying paths of fastq files. If type=="single", must be of length 1. If type=="paired", must be of length 2.
sai_files	Character vector specifying filenames of the bwa aln alignment output files. If type=="single", must be of length 1. If type=="paired", must be of length 2.
sam_file	String specifying paths of the SAM output file.
...	Other arguments to pass to bwa_sam.

**Value**

No return value. Output SAM files are produced as side effect.

**Author(s)**

Jean-Philippe Fortin

**Examples**

```
# Creating index:
dir <- tempdir()
fasta <- system.file(package="Rbwa",
                     "fasta/chr12.fa")
fastq <- system.file(package="Rbwa",
```

```
        "fastq/sequences.fastq")
index_prefix <- file.path(dir,"chr12")
bwa_build_index(fasta, index_prefix=index_prefix)

# Creating alignments:
bwa_aln(index_prefix=index_prefix,
        fastq_files=fastq,
        sai_files=file.path(dir, "output.sai"))

# Generating SAM file:
bwa_sam(index_prefix=index_prefix,
        fastq_files=fastq,
        sai_files=file.path(dir, "output.sai"),
        sam_file=file.path(dir, "output.sam"))

# Reading in alignments from SAM file:
aln <- readLines(file.path(dir, "output.sam"))
aln
```

---

xa2multi

*Unpack multiple alignments stored in BWA output*

---

## Description

Unpack multiple alignments stored in BWA output

## Usage

```
xa2multi(input_sam_file, output_sam_file)
```

## Arguments

`input_sam_file` String specifying path of the input SAM file.

`output_sam_file`

String specifying path of the output SAM file.

## Details

Each row in the SAM file produced by `bwa_aln` corresponds to the best alignment hit for a given input query sequence. Other alignments (secondary alignments, or other loci in case of multiple alignments) are stored in the XA tag.

`xa2multi` conveniently extracts the alignments from the XA tags and represent them as additional rows in the SAM format.

## Value

Returns NULL invisibly. SAM file with multiple alignments is produced as a side effect.

**Author(s)**

Jean-Philippe Fortin

**Examples**

```
# Creating index:
dir <- tempdir()
fasta <- system.file(package="Rbwa",
                     "fasta/chr12.fa")
fastq <- system.file(package="Rbwa",
                     "fastq/sequences.fastq")
index_prefix <- file.path(dir,"chr12")
bwa_build_index(fasta, index_prefix=index_prefix)

# Creating alignments:
bwa_aln(index_prefix=index_prefix,
        fastq_files=fastq,
        sai_files=file.path(dir, "output.sai"))

# Generating SAM file:
bwa_sam(index_prefix=index_prefix,
        fastq_files=fastq,
        sai_files=file.path(dir, "output.sai"),
        sam_file=file.path(dir, "output.sam"))

# Generating multiple alignments:
xa2multi(input_sam_file=file.path(dir, "output.sam"),
        output_sam_file=file.path(dir, "output.multi.sam"))

#' Reading in:
aln <- readLines(file.path(dir, "output.multi.sam"))
aln
```

# Index

bwa\_aln, [2](#), [6](#)  
bwa\_build\_index, [3](#)  
bwa\_mem, [4](#)  
bwa\_sam, [5](#)  
  
xa2multi, [6](#)