Package 'ssw'

September 18, 2024

Title Striped Smith-Waterman Algorithm for Sequence Alignment using
SIMD

Version 0.2.1

Description Provides an R interface for 'SSW' (Striped Smith-Waterman)
via its 'Python' binding 'ssw-py'. 'SSW' is a fast 'C' and 'C++'
implementation of the Smith-Waterman algorithm for pairwise sequence
alignment using Single-Instruction-Multiple-Data (SIMD) instructions.
'SSW' enhances the standard algorithm by efficiently returning alignment
information and suboptimal alignment scores.
The core 'SSW' library offers performance improvements for various
bioinformatics tasks, including protein database searches,
short-read alignments, primary and split-read mapping,
structural variant detection, and read-overlap graph generation.
These features make 'SSW' particularly useful for genomic applications.
Zhao et al. (2013) <doi:10.1371/journal.pone.0082138> developed the
original 'C' and 'C++' implementation.

License MIT + file LICENSE

URL https://nanx.me/ssw-r/, https://github.com/nanxstats/ssw-r

BugReports https://github.com/nanxstats/ssw-r/issues

Encoding UTF-8

SystemRequirements Python (>= 3.6.0), ssw-py (>= 1.0.0). Detailed
installation instructions can be found in the README file.

VignetteBuilder knitr

Depends R (>= 4.1.0)

Imports reticulate

Suggests knitr, rmarkdown

RoxygenNote 7.3.2

NeedsCompilation no

Author Nan Xiao [aut, cre, cph] (<https://orcid.org/0000-0002-0250-5673>)

Maintainer Nan Xiao <me@nanx.me>

Repository CRAN

Date/Publication 2024-09-18 02:10:02 UTC

Contents

---

align	*Perform Smith-Waterman alignment of a read against a reference sequence*

Description

Perform Smith-Waterman alignment of a read against a reference sequence

Usage

```
align(
  read,
  reference,
  gap_open = 3L,
  gap_extension = 1L,
  start_idx = 0L,
  end_idx = 0L,
  match_score = 2L,
  mismatch_penalty = 2L
)
```

Arguments

read	A character string of the read.
reference	A character string of the reference.
gap_open	Penalty for opening a gap. Default is 3.
gap_extension	Penalty for extending a gap. Default is 1.
start_idx	Index to start the alignment search. Default is 0.
end_idx	Index to end the alignment search. The default value 0 means using the entire reference length.
match_score	An integer for scoring matches, ranging from 0 to 255. Default is 2.
mismatch_penalty	
	An integer for mismatch penalties, ranging from 0 to 255. Default is 2.

Value

A list of class `ssw` containing the ssw aligner object and the alignment results.

Examples

```
a <- align("ACGT", "TTTTACGTCCCCC")
a
a$alignment$optimal_score
a$alignment$sub_optimal_score
```

---

force_align   *Perform forced alignment with increased gap open penalty*

Description

Perform forced alignment with increased gap open penalty

Usage

```
force_align(
  read,
  reference,
  force_overhang = FALSE,
  match_score = 2L,
  mismatch_penalty = 2L
)
```

Arguments

read	A character string of the read.
reference	A character string of the reference.
force_overhang	Logical. If TRUE, ensures only one end of the alignment overhangs. Default is FALSE.
match_score	An integer for scoring matches, ranging from 0 to 255. Default is 2.
mismatch_penalty	An integer for mismatch penalties, ranging from 0 to 255. Default is 2.

Value

A list of class `ssw` containing the input sequences, the ssw aligner object, and the alignment results.

Examples

```
# Results are truncated
a <- force_align("ACTG", "TTTTCTGCCCCCACG")
a

# Format the results
b <- a |> formatter()
b

# Print the formatted results directly
a |> formatter(print = TRUE)
```

---

formatter	*Format and pretty-print SSW forced alignment results without trunca-tion*

Description

Format and pretty-print SSW forced alignment results without truncation

Usage

```
formatter(x, print = FALSE)
```

Arguments

x	An object of class ssw containing the forced alignment results.
print	Pretty-print the results? Default is FALSE.

Value

The formatted forced alignment results.

Examples

```
a <- force_align("ACTG", "TTTTCTGCCCCCACG")
b <- a |> formatter()
b
a |> formatter(print = TRUE)
```

---

install_ssw_py *Install ssw-py and its dependencies*

Description

Install ssw-py and its dependencies

Usage

```
install_ssw_py(
  ...,
  envname = "r-ssw-py",
  new_env = identical(envname, "r-ssw-py")
)
```

Arguments

...	Other arguments passed to [reticulate::py_install()]().
envname	The name or full path of the environment in which ssw-py is installed. Default is `r-ssw-py`.
new_env	Logical. If `TRUE`, the specified Python environment will be deleted and recreated if it already exists. Defaults to `TRUE` only when using the default environment name.

Value

Invisibly returns `NULL`. Primarily used for its side effect of installing the Python package in the specified environment.

Examples

```
install_ssw_py()
```

---

is_installed_ssw_py *Is ssw-py installed?*

Description

Is ssw-py installed?

Usage

```
is_installed_ssw_py()
```

Value

TRUE if installed, FALSE if not.

Examples

```
is_installed_ssw_py()
```

---

print.ssw   *Print SSW alignment results*

Description

Print SSW alignment results

Usage

```
## S3 method for class 'ssw'
print(x, start_idx = 0L, ...)
```

Arguments

x	An object of class ssw.
start_idx	Index to start printing from.
...	Additional parameters for print() (not used).

Value

Invisibly returns the input object.

Examples

```
a <- align("ACGT", "TTTTACGTCCCCC")
a
```

ssw_py

Global reference to ssw-py

Description

Global reference to ssw-py which will be initialized in `.onLoad`.

Usage

```
ssw_py
```

Format

An object of class `python.builtin.module` (inherits from `python.builtin.object`) of length 0.

Value

ssw-py reference object

Index