

Package ‘DNAcycP2’

August 10, 2025

Title DNA Cyclizability Prediction

Version 1.0.0

Description This package performs prediction of intrinsic cyclizability of every 50-bp subsequence in a DNA sequence. The input could be a file either in FASTA or text format. The output will be the C-score, the estimated intrinsic cyclizability score for each 50 bp sequences in each entry of the sequence set.

Depends R (>= 4.4.0)

Imports basilisk, reticulate

License Artistic-2.0

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

StagedInstall yes

BinaryFiles inst/extdata/irlstm/saved_model.pb,
inst/extdata/irlstm_smooth/saved_model.pb

biocViews NeuralNetwork, StructuralPrediction

VignetteBuilder knitr

Suggests knitr, rmarkdown, BiocGenerics, RUnit, tinytest, BiocStyle,
Biostrings

LazyData false

URL <https://github.com/jipingw/DNAcycP2>

BugReports <https://github.com/jipingw/DNAcycP2>

git_url <https://git.bioconductor.org/packages/DNAcycP2>

git_branch RELEASE_3_21

git_last_commit 1f037b7

git_last_commit_date 2025-04-15

Repository Bioconductor 3.21

Date/Publication 2025-08-10

cycle_fasta	<i>Predict Cyclizability</i>
-------------	------------------------------

Description

This predicts cyclizability for all subsequences of length 50bp from a .fasta input file.

Usage

```
cycle_fasta(
  file_path,
  smooth,
  n_cores = 1,
  chunk_length = 1e+05,
  save_path_prefix = ""
)
```

Arguments

file_path	.fasta input file path
smooth	Whether to predict smoothed C0 (DNAcycP2) or original C0 (DNAcycP)
n_cores	Number of cores to use for parallel processing (default=1)
chunk_length	Length of sequence that each core will predict on at a given time. (default=100000)
save_path_prefix	Base path for output files. If it is an empty string, the output files will not be saved (default="")

Details

Optionally, saves output files (use argument 'save_path_prefix')

Value

A list of predictions for each ID in the .fasta file.

Each list item has the following columns: position, c_score_norm (predictions on a normalized scale), and c_score_unnorm (predictions on an unnormalized scale).

Each list item is named "cycle_\$id\$" corresponding to the fasta id

Examples

```
# Create a temporary file
temp_file <- tempfile(fileext = ".fasta")
writeLines(">1", temp_file)
writeLines("ACTGCTAGTCACTGCTAGTCACTGCTAGTCACTGCTAGTCACTGCTAGTC", temp_file)

# Example usage of cycle_fasta
```

```
cycle_fasta(temp_file, smooth=TRUE)

# Cleanup
unlink(temp_file)
```

Index

cycle, [2](#)
cycle_fasta, [3](#)