Identification of Protein Binding Sites at High Resolution with 'dpeak' Package

Dongjun Chung[1] and Sündüz Keleş[1,2]
[1]Department of Statistics, University of Wisconsin
Madison, WI 53706.
[2]Department of Biostatistics and Medical Informatics, University of Wisconsin
Madison, WI 53706.

April 28, 2020

Contents

1 Installation

```
> if(!requireNamespace("BiocManager", quietly = TRUE))
+     install.packages("BiocManager")
> BiocManager::install("dpeak")
```

2 Overview

This vignette provides an introduction to the 'dpeak' package. The 'dpeak' package implements dPeak, a parametric mixture modeling approach to identify protein binding sites in each of given peak regions at high spatial resolution.

The package can be loaded with the command:

```
R> library("dpeak")
```

3 Workflow

3.1 Reading Peak List and Aligned Reads into the R Environment

We assume that you already have a peak list and the corresponding aligned read file for your ChIP sample. The peak list and aligned reads can be imported to the R environment with the command:

```
R> exampleData <- dpeakRead( peakfile="examplePeak.txt", readfile="exampleSETRead.txt",
+     fileFormat="eland_result", PET=FALSE, fragLen=150 )
```

You can specify the names of the peak list and the aligned read file in arguments '`peakfile`' and '`readfile`', respectively. '`dpeakRead`' method assumes that first three columns of the peak list file are chromosome ID, start and end positions of each peak region. The peak list file should not include header. Standard BED file format satisfies these requirements.

The '`PET`' argument indicates whether the aligned read data is single-end tag (SET) or paired-end tag (PET) data. File format of the aligned read file can be specified in '`fileFormat`'. For SET data ('`PET=FALSE`'), '`dpeakRead`' method allows the following aligned read file formats: Eland result ('`"eland_result"`'), Eland extended ('`"eland_extended"`'), Eland export ('`"eland_export"`'), default Bowtie ('`"bowtie"`'), SAM ('`"sam"`'), and BED ('`"bed"`'). For PET data ('`PET=TRUE`'), only eland result file format ('`"eland_result"`') is supported. For SET data, you might also want to specify average fragment length in the '`fragLen`' argument (default is 200 bp). The '`fragLen`' argument is not relevant when '`PET=FALSE`' (PET data).

R package '`dpeak`' provides functions for generating simple summaries of the data. The following command prints out basic information, such as number of peaks, number of chromosomes in the data, tag type (SET or PET), number of utilized reads, and median number of reads in each peak. In addition, for PET data, median fragment length is also provided (for SET data, average fragment length provided by user will be printed out).

```
R> exampleData

------------------------------------------------------------
Summary: Dpeak data (class: DpeakData)
------------------------------------------------------------
Number of peaks: 1
Number of chromosomes: 1
Tag type: SET
Fragment length (provided by user): 150
Number of utilized reads: 3316
Median number of reads in each peak: 3316
------------------------------------------------------------
```

'`exportPlot`' method exports exploratory plots for the ChIP data to a PDF file. Its file name needs to be specified in the '`filename`' argument. These plots show number of reads (or fragments) aligned to each position within each peak region. For SET data, if '`strand=TRUE`', reads are plotted in a strand-specific manner, where each read is extended to the value specified in the '`extension`' argument from its 5' end. Moreover, if '`smoothing=TRUE`', a smoothed plot (using the smoothing spline) is provided. Unsmoothed plot is provided by default.

```
R> exportPlot( exampleData, filename="examplePlot_combined.pdf",
+         strand=FALSE )
R> exportPlot( exampleData, filename="examplePlot_strand_1.pdf",
+         strand=TRUE, extension=1, smoothing=TRUE )
```

Figures 1 and 2 display examples of the data plot without strand information and the strand-specific data plot, respectively.

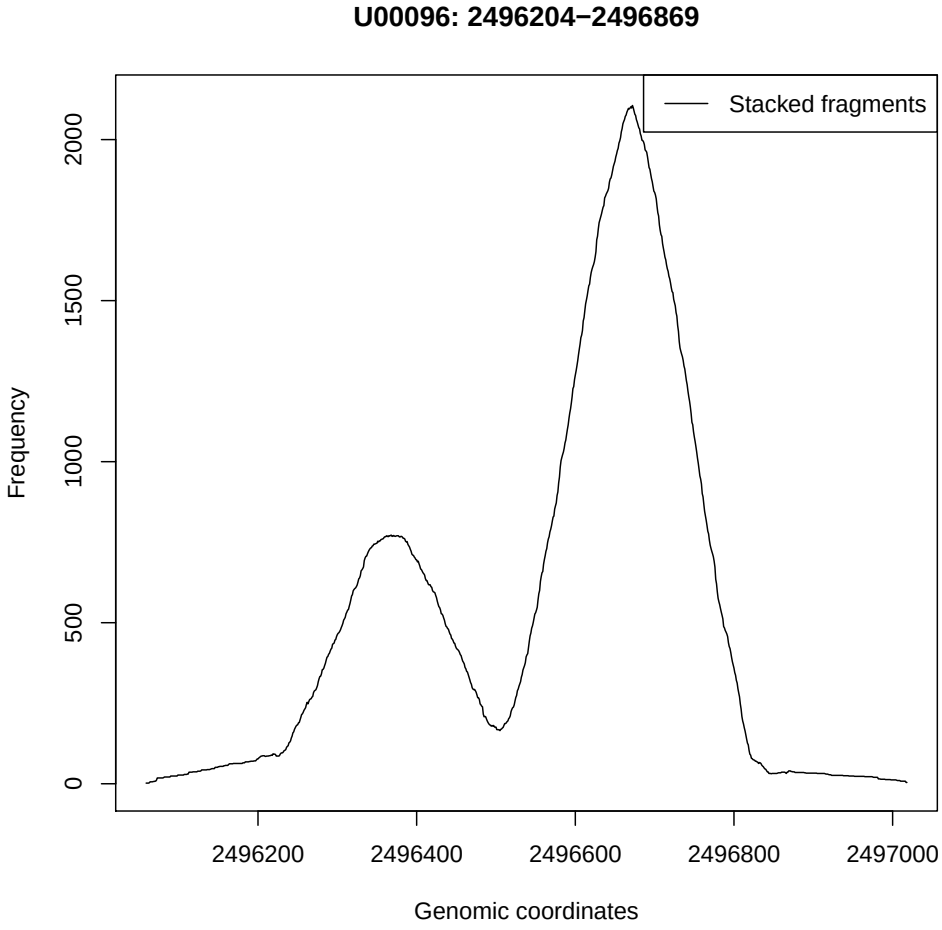

Figure 1: `Data plot.` Height at each position indicates the number of fragments aligned to the position. In SET data, fragments are defined as reads extended to average fragment length.

3.2 Identifying Binding Sites

We can now fit a dPeak model using the imported data (`exampleData`) with the command:

```
R> exampleFit <- dpeakFit( exampleData, maxComp=5)
```

'`dpeakFit`' fits models with at most '`maxComp`' binding events for each peak and chooses the best model among them for each peak, based on Bayesian Information Criterion (BIC) values. If '`multicore`' package is installed, parallel computing can be utilized and multiple number of peaks are analyzed simultaneously. You can specify number of utilized CPUs in '`nCore`' (default: 8). The following command prints out a basic summary of the fitted model, such as median number of binding events in each peak region.

```
R> exampleFit
```

```
------------------------------------------------------------
Summary: Dpeak model fitting (class: DpeakFit)
```

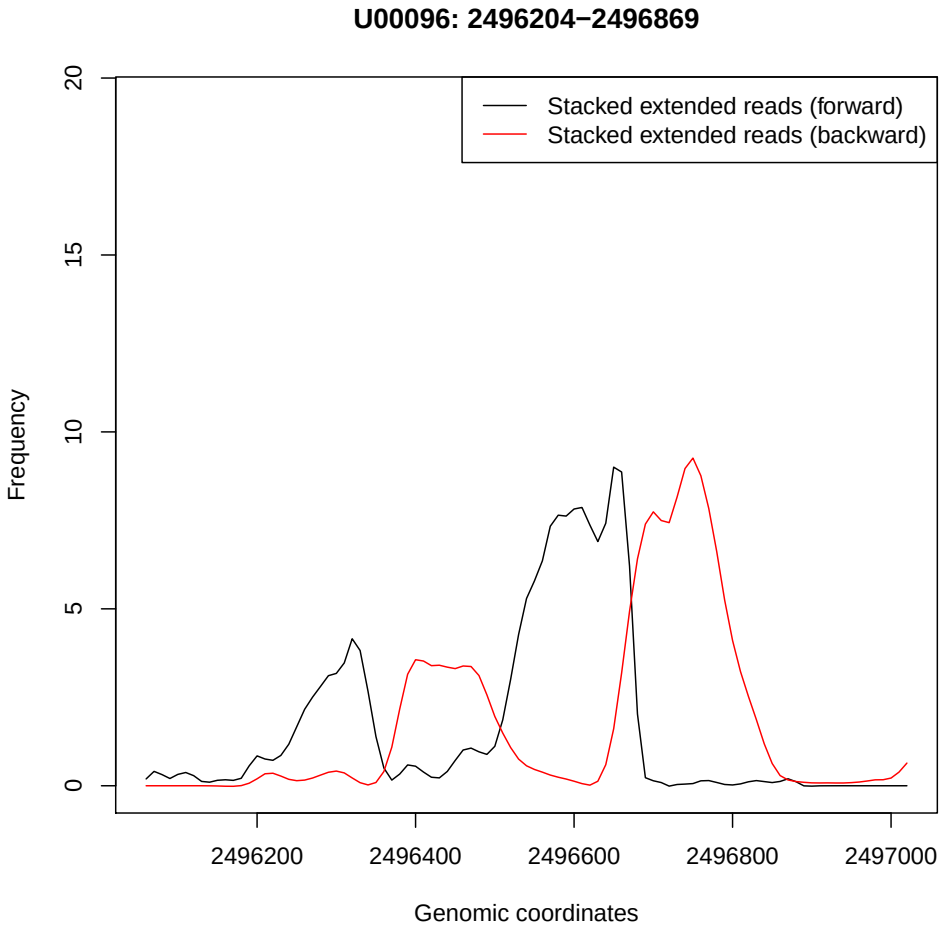

Figure 2: `Strand-specific data plot.` Height at each position indicates the number of 5' end of reads aligned to the position.

```
-------------------------------------------------------------
- Maximum possible number of binding events in each peak: 5
- Median number of binding events in each peak: 2
- Median explanation ratio: 1 %
-------------------------------------------------------------
```

'`exportPlot`' method exports the plots of estimated binding sites ('`plotType="fit"`') or the goodness of fit (GOF) plots ('`plotType="GOF"`') to a PDF file. Its file name needs to be specified in the '`filename`' argument. In both of these plots, estimated binding sites or simulated fragments are superimposed on the plots of reads (or fragments) aligned to each position (such as figures 1 and 2). For SET data, if '`plotType="fit"`' and '`strand=TRUE`', reads will be plotted in a strand-specific manner, where each read is extended to '`extension`' from its 5' end. If '`smoothing=TRUE`', a smoothed plot (using the smoothing spline) is provided. Unsmoothed plot is provided by default.

```
R> exportPlot( exampleFit, filename="exampleResult_combined.pdf" )
R> exportPlot( exampleFit, filename="exampleResult_strand_1.pdf",
```

```
+         strand=TRUE, extension=1, smoothing=TRUE )
R> exportPlot( exampleFit, filename="exGOF.pdf", plotType="GOF" )
```

Figures 3 and 4 display examples of plots of estimated binding sites. Estimated binding sites (blue vertical dashed lines) are superimposed on the data plot. Figure 5 shows an example of the GOF plot and it indicates that the model fits the data quite well.

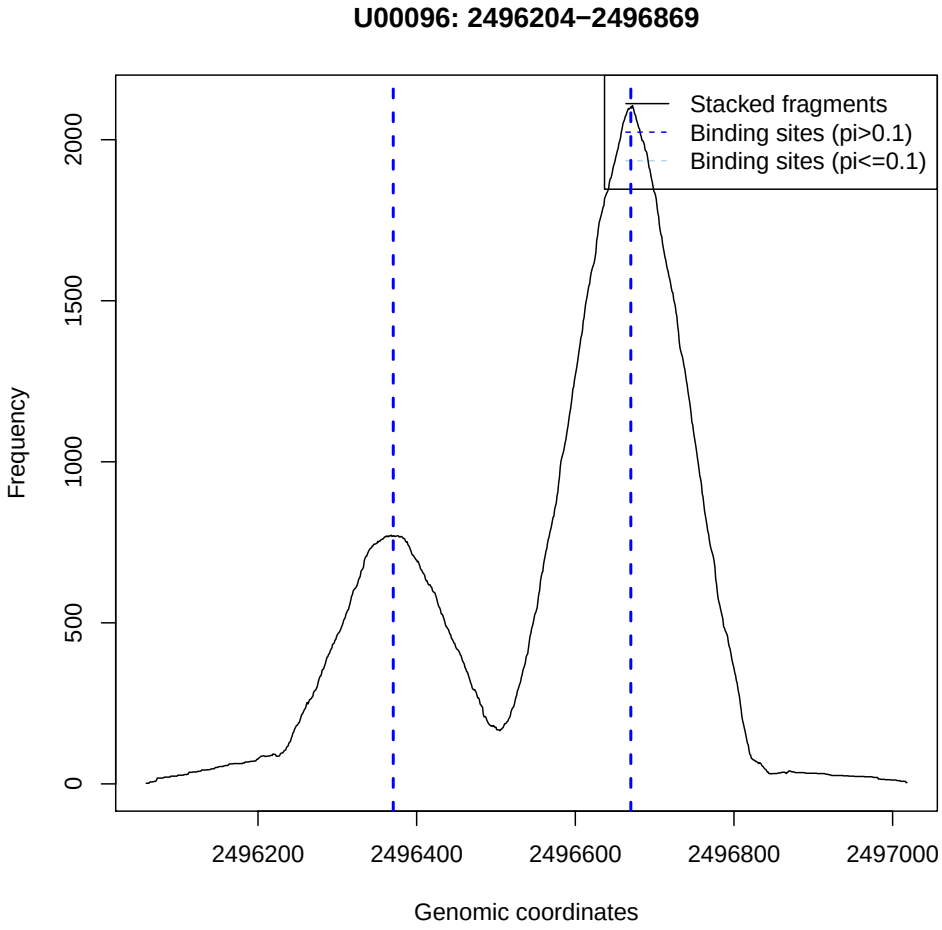

Figure 3: `Plot of estimated binding sites.` Height at each position indicates the number of fragments aligned to the position. In SET data, fragments are defined as reads extended to average fragment length. Blue vertical dashed lines indicate estimated binding sites.

You can export the binding site identification results to text files. Name of the file to be exported needs to be specified in the '`filename`' argument. '`dpeak`' package supports TXT, BED, and GFF file formats. In the exported file, TXT file format ('`type="txt"`') includes chromosome ID, binding site, relative binding strength in each peak region, and the peak region that each binding event belongs to. '`type="bed"`' and '`type="gff"`' export binding site identification results in standard BED and GFF file formats, respectively, where score is the estimated binding strength multiplied by 1000. The feature of GFF file and the name of BED file indicate the peak region that each binding event belongs to. Peak calling results can be exported in TXT, BED, and GFF file

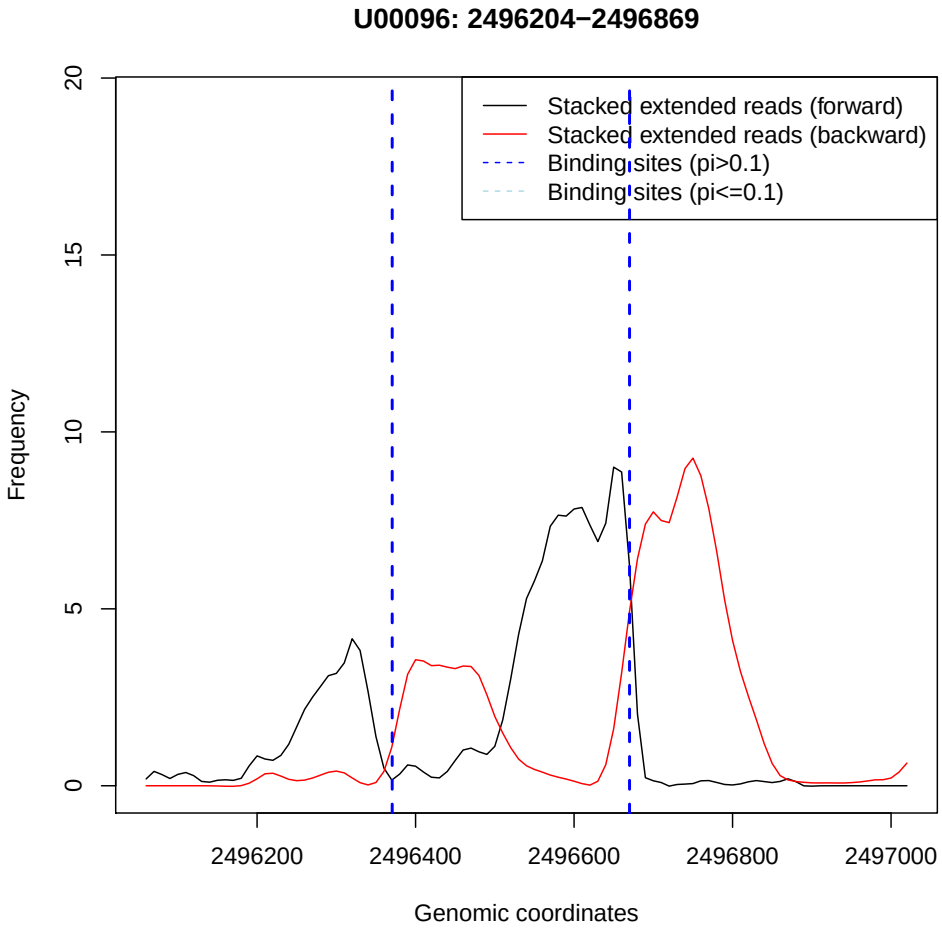

Figure 4: `Strand-specific plot of estimated binding sites.` Height at each position indicates the number of 5' end of reads aligned to the position. Blue vertical dashed lines indicate estimated binding sites.

formats, respectively, by the commands:

```
R> exportPeakList( exampleFit, type="txt", filename="result.txt" )
R> exportPeakList( exampleFit, type="bed", filename="result.bed" )
R> exportPeakList( exampleFit, type="gff", filename="result.gff" )
```

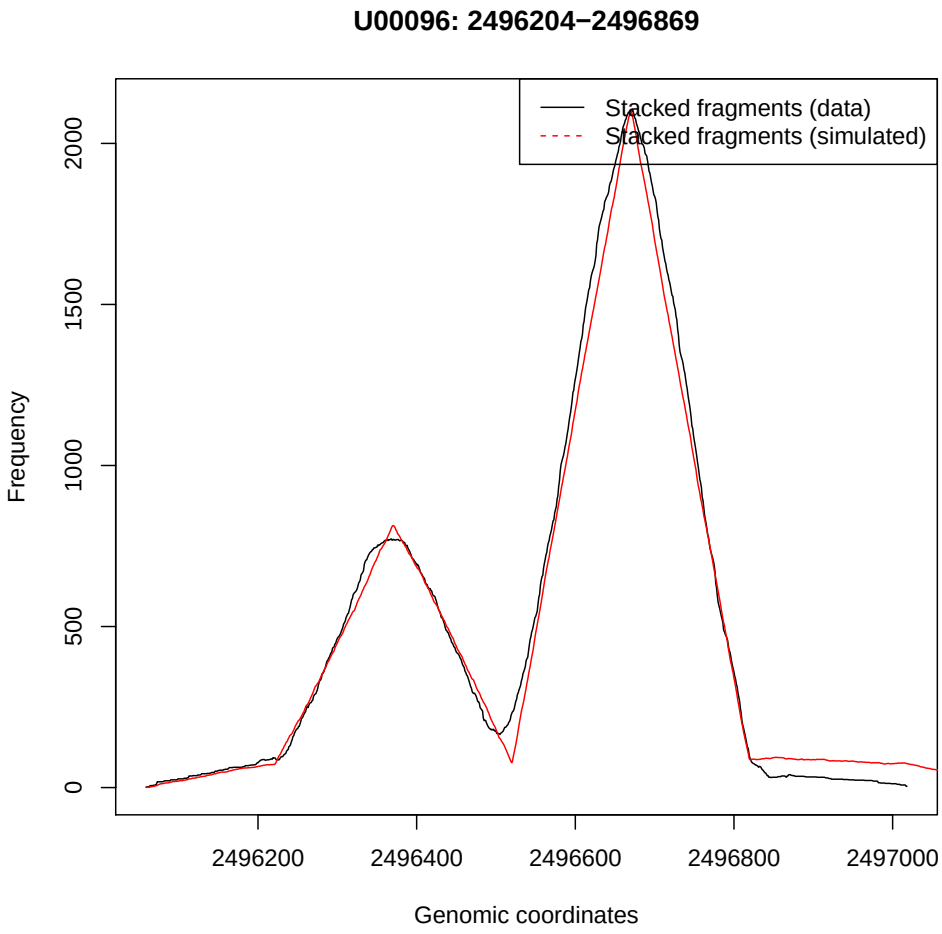

Figure 5: Goodness of Fit (GOF) plot.