# seqbias

October 25, 2011

---

`count.reads` *Counting reads across intervals*

---

### Description

Counts the number of reads starting at each position across given genomic intervals

### Usage

```
count.reads(reads_fn, I, binary=TRUE)
```

### Arguments

| | |
|---|---|
| `reads_fn` | filename of aligned reads in BAM format |
| `I` | a GRanges object giving valid genomic intervals |
| `binary` | if `TRUE`, return a 0-1 vector, otherwise return a vector counting the number of reads mapped to each position |

### Details

Given an indexed BAM file, this function counts the number of reads starting at each position of each provided interval. These counts can then be normalized by dividing by the predicted bias obtained from 'bias.predict'.

By default, a 0-1 vector is returned, where positions at which no reads are mapped are 0, and those with one or more are 1. If `binary` is `FALSE`, the number of reads mapping to each position is returned.

### Value

A list of numeric vectors is returned, one for each interval provided. Each vector gives an integer count of the number of reads beginning on that position.

### Note

The BAM file provided should be indexed with 'samtools index'.

### Author(s)

Daniel Jones <dcjones@cs.washington.edu>

## See Also

seqbias.predict

## Examples

```
reads_fn <- system.file( "extra/example.bam", package = "seqbias" )

I <- GRanges( c('seq1'), IRanges( c(1), c(5000) ), strand = c('-') )

counts <- count.reads( reads_fn, I )
```

---

kmer.freq *Measuring positional kmer frequencies*

---

## Description

Given a sample of sequences and corresponding read counts, produce a table giving the position kmer frequencies relative to read starts

## Usage

```
kmer.freq(seqs, counts, L = 50, R = 50, k = 1)
```

## Arguments

| | |
|---|---|
| seqs | a list of DNAString objects. |
| counts | a list of numeric vectors. |
| L | how many positions to the left of the read start to consider |
| R | how many positions to the right of the read start to consider |
| k | the size of each kmer |

## Details

Sequences and read counts are used to produce a table of aggregate kmer frequencies for each position relative to the read start. The position on which the read starts is numbered 0, positions to the left of the read are negative, and those to the right are positive.

The sequences and counts can be generated with the provided functions scanFa and count.reads, respectively. The reverse complement of sequences on the negative strand obtained from scanFa should be used. To properly visualize bias a relatively large random sample of intervals should be generated.

## Value

A data frame is returned with columns pos, seq, and freq. Where pos gives the position relative to te read start, seq gives the kmer, and freq gives the frequency of that kmer.

## Author(s)

Daniel Jones <dcjones@cs.washington.edu>

## See Also

count.reads

## Examples

```
library(Rsamtools)
reads_fn <- system.file( "extra/example.bam", package = "seqbias" )
ref_fn <- system.file( "extra/example.fa", package = "seqbias" )

I <- GRanges( c('seq1'), IRanges( c(1), c(5000) ), strand = c('-') )

ref_f <- FaFile( ref_fn )
open.FaFile( ref_f )

seqs <- scanFa( ref_f, I )

neg_idx <- as.logical( I@strand == '-' )
seqs[neg_idx] <- reverseComplement( seqs[neg_idx] )

counts <- count.reads( reads_fn, I )

freqs <- kmer.freq(seqs, counts, L = 30, R = 30, k = 2)
```

---

# random.intervals *Generating random genomic intervals*

---

## Description

Given a vector of sequence lengths, generate genomic intervals uniformly at random

## Usage

```
random.intervals(I, n=1, ms=10000)
```

## Arguments

| | |
|---|---|
| I | a GRanges object giving intervals from which to sample from |
| n | number of intervals to generate |
| ms | length of the intervals to generate (may be a vector) |

## Details

The function is used to place intervals of fixed sizes at random (possibly overlapping) positions across one or more sequences. The input should be a GRanges objects giving the sequence intervals in which the random intervals sholud be placed. If they are to be placed anywhere within a reference sequence, use the scanFaIndex function from Rsamtools, to obtain a set of intervals.

## Value

Returns a GRanges object giving the generated intervals.

### Author(s)

Daniel Jones <dcjones@cs.washington.edu>

### Examples

```
library(Rsamtools)
ref_fn <- system.file( "extra/example.fa", package = "seqbias" )
ref_f <- FaFile( ref_fn )
open.FaFile( ref_f )

ref_seqs <- scanFaIndex( ref_f )

I <- random.intervals( ref_seqs, n = 100, ms = 1000 )
```

---

## seqbias *The seqbias model*

---

### Description

The seqbias class maintains a model of the sequencing bias of from an experiment, which can be saved, loaded, trained, and used to make predictions of bias.

The class is manipulated with the following functions: seqbias.fit seqbias.load seqbias.predict seqbias.save

### Author(s)

Daniel Jones <dcjones@cs.washington.edu>

---

## seqbias-package *'seqbias' modeling bias in high-throughput sequencing data*

---

### Description

This package implements a model of sequencing bias in high-throughput sequencing data using a simple Bayesian network, the structure and parameters of which are trained on a set of aligned reads and a reference genome sequence.

### Author(s)

Daniel Jones <dcjones@cs.washington.edu>

`seqbias.fit` *Fitting seqbias models*

## Description

Fits a seqbias module given a reference sequence and reads in BAM format

## Usage

```
seqbias.fit(ref_fn, reads_fn, n = 1e5, L = 15, R = 15)
```

## Arguments

| | |
|---|---|
| `ref_fn` | filename of a reference sequence against which the reads are aligned, in FASTA format. |
| `reads_fn` | filename of aligned reads in BAM format. |
| `n` | train on at most this many reads. |
| `L` | consider at most L positions to the left of the read start. |
| `R` | consider at most R positions to the right of the read start. |

## Details

A Bayesian network is trained on the first `n` unique reads in the provided BAM file, predicting the posterior probability of a read beginning at a position given the surrounding sequence. This is used to discern the sequencing bias: how more or less likely a read is to fall on a particular position.

The abundance of region can be more accurately assessed by normalizing (dividing) each position by its predicted bias.

## Value

A vector of reals giving the predicted sequencing bias for each position.

## Note

Both the BAM file and the FASTA file should be indexed, with, 'samtools index' and, 'samtools faidx' respectively.

## Author(s)

Daniel Jones <dcjones@cs.washington.edu>

## See Also

`seqbias.predict`

### Examples

```
reads_fn <- system.file( "extra/example.bam", package = "seqbias" )
ref_fn <- system.file( "extra/example.fa", package = "seqbias" )

sb <- seqbias.fit( ref_fn, reads_fn )

I <- GRanges( c('seq1'), IRanges( c(1), c(5000) ), strand = c('-') )

bias <- seqbias.predict( sb, I )
```

---

## seqbias.load *Loading seqbias models*

---

### Description

Loads a seqbias model from a file written with 'seqbias.save'.

### Usage

```
seqbias.load(ref_fn, model_fn)
```

### Arguments

| | |
|---|---|
| ref_fn | filename of a reference sequence against which the reads are aligned in FASTA format. |
| model_fn | filename of a saved seqbias model |

### Details

A large seqbias model can take some time (several minutes) to fit. It is often preferable to do this just once. This function load the model from a file in YAML format, having been written with 'seqbias.save'.

### Value

A seqbias class.

### Author(s)

Daniel Jones <dcjones@cs.washington.edu>

### See Also

seqbias.save

## Examples

```
reads_fn <- system.file( "extra/example.bam", package = "seqbias" )
ref_fn <- system.file( "extra/example.fa", package = "seqbias" )

sb <- seqbias.fit( ref_fn, reads_fn )

seqbias.save( sb, "my_seqbias_model.yml" )

# load sometime later
sb <- seqbias.load( ref_fn, "my_seqbias_model.yml" )
```

---

`seqbias.predict` *Predicting sequencing bias*

---

## Description

Predicts sequencing bias given a fit seqbias model

## Usage

```
seqbias.predict(sb, I)
```

## Arguments

| | |
|---|---|
| `sb` | a seqbias object |
| `I` | a GRanges object |

## Details

Once a seqbias model is fit with 'seqbias.fit', the sequencing bias of any region in the reference sequence can be predicted using this function. Given the coordinates of a region, this function produces a vector of the same length as the sequence. Each position 'i' is given a sequence score 'v_i'.

A simple procedure is then to normalize read counts given the sequencing bias. The read count of (i.e. the number of reads beginning on) position 'i', denoted by 'x_i', can be normalized by computing 'x_i/v_i', giving an estimate of abundance that is more accurate in expectation.

## Value

A list of numeric vectors are returned, one for each genomic interval in I. The vectors are of equal length to the interval given, and the predicted sequencing bias is given for each position.

## Author(s)

Daniel Jones <dcjones@cs.washington.edu>

## See Also

`seqbias.fit`

### Examples

```
reads_fn <- system.file( "extra/example.bam", package = "seqbias" )
ref_fn <- system.file( "extra/example.fa", package = "seqbias" )

sb <- seqbias.fit( ref_fn, reads_fn )

I <- GRanges( c('seq1'), IRanges( c(1), c(5000) ), strand = c('-') )

bias <- seqbias.predict( sb, I )
```

---

## seqbias.save *Saving seqbias models*

---

### Description

Writes to a seqbias model to a file suitable to loaded with 'seqbias.load'

### Usage

```
seqbias.save(sb, fn)
```

### Arguments

| | |
|---|---|
| sb | A seqbias class created with 'seqbias.fit' or 'seqbias.load' |
| fn | A file name to write the model to |

### Details

A large seqbias model can take some time (several minutes) to fit. It is often preferable to do this just once. This function writes the model to a file in YAML format, suitable to be read with 'seqbias.load'.

### Author(s)

Daniel Jones <dcjones@cs.washington.edu>

### See Also

seqbias.load

### Examples

```
reads_fn <- system.file( "extra/example.bam", package = "seqbias" )
ref_fn <- system.file( "extra/example.fa", package = "seqbias" )

sb <- seqbias.fit( ref_fn, reads_fn )

seqbias.save( sb, "my_seqbias_model.yml" )

# load sometime later
sb <- seqbias.load( ref_fn, "my_seqbias_model.yml" )
```

# Index