Rsubread package: high-performance read alignment, quantification and mutation discovery

Wei Shi

21 November 2022

1 Introduction

This vignette provides a brief description to the Rsubread package. For more details, please refer to the Users Guide which can brought up in your R session via the following commands:

```
> library(Rsubread)
> RsubreadUsersGuide()
```

The Rsubread package provides facilities for processing short and long reads generated from the sequencing technologies. These facilities include quality assessment, read alignment, read summarization, exon-exon junction detection, absolute expression calling and SNP discovery.

The Subread aligner (`align` function) is a highly efficient and accurate aligner for mapping genomic DNA and RNA sequencing reads. It adopts a novel mapping paradigm called "seed-and-vote". Under this paradigm, a number of 16mers (called seeds or subreads) are extracted from each read and they were mapped to the reference genome to vote for the mapping location of the read. Read mapping performed under this paradigm has been found to be more efficient and accurate than that carried out under the conventional "seed-and-extend" paradigm (Liao et al. 2013).

Another aligner included in Rsubread is the Subjunc aligner (`subjunc` function). It was also developed under the "seed-and-vote" paradigm, but different from the Subread aligner it performs full alignment for exon spanning reads and reports exon-exon junctions in addition to the read mapping results. The Subread aligner is recommended for gene-level expression analysis. For other types of RNA-seq analyses such as alternative splicing analysis, the Subjunc aligner should be used.

An important step in processing next-gen sequencing data is to assign mapped reads to genomic features such as genes, exons and genomic windows. This package includes a general-purpose read summarization function `featureCounts` that takes mapped reads

as input and assigns them to genomic features. In-built annotations are provided for users convenience.

Different from microarray technologies, the next-gen sequencing technologies do not provide Present/Absent calls for genomic features such as genes. We have developed an algorithm to use the background noise measured from the RNA-seq data to call absolutely expressed genes. The function `detectionCall` returns a detection p value for each gene from the read mapping results.

We have also developed a new SNP calling algorithm which is being implemented in function `exactSNPs`. Our results showed that it compared favorably to competing methods, but was an order of magnitude faster.

This package also includes some other useful functions such as quality assessment (`qualityScores`, `atgcContent`), duplicate read removal (`removeDupReads`) and mapping percentage calculation (`propmapped`).

2 Read alignment

An index needs to be built first and then alignments can be carried out. Building the index is an one-off operation. The generated index can be re-used in subsequent read alignments.

Step 1: Index building

The Rsubread package includes a dummy reference sequence that was generated by concatenating 900 100bp reads that were taken from a pilot dataset generated from the SEquencing Quality Control (SEQC) project. We further extracted 100 reads from the same dataset and combine them with the 900 reads to make a read dataset for mapping. Below is the command for building an index for the reference sequence:

```
> library(Rsubread)
> ref <- system.file("extdata","reference.fa",package="Rsubread")
> buildindex(basename="reference_index",reference=ref)
```

```
WARNING: your system seems to be 32-bit. Rsubread supports 32-bit systems to a very limited level.
It is highly recommended to run Rsubread on a 64-bit system to avoid errors.


        ==========     _____ _    _ ____  _____  ______          _____
        =====         / ____| |  | |  _ \|  __ \|  ____|   /\   |  __ \
          =====      | (___ | |  | | |_) | |__) | |__     /  \  | |  | |
            ====      \___ \| |  | |  _ <|  _  /|  __|   / /\ \ | |  | |
              ====    ____) | |__| | |_) | | \ \| |____ / ____ \| |__| |
        ==========   |_____/ \____/|____/|_|  \_\______/_/    \_\_____/
       Rsubread 2.12.3

//================================= setting ==================================\\
||                                                                            ||
||                Index name : reference_index                                ||
||               Index space : base space                                     ||
||               Index split : no-split                                       ||
```

```
||          Repeat threshold : 100 repeats                                ||
||              Gapped index : no                                         ||
||                                                                        ||
||       Free / total memory : 4.2GB / 8.0GB                              ||
||                                                                        ||
||               Input files : 1 file in total                            ||
||                             o reference.fa                             ||
||                                                                        ||
\\========================================================================//

//================================ Running ================================\\
||                                                                        ||
|| Check the integrity of provided reference sequences ...                ||
|| No format issues were found                                            ||
|| Scan uninformative subreads in reference sequences ...                 ||
|| 1 uninformative subreads were found.                                   ||
|| These subreads were excluded from index building.                      ||
|| Estimate the index size...                                             ||
||    8%,   0 mins elapsed, rate=128.5k bps/s                             ||
||   16%,   0 mins elapsed, rate=236.9k bps/s                             ||
||   24%,   0 mins elapsed, rate=335.0k bps/s                             ||
||   33%,   0 mins elapsed, rate=427.4k bps/s                             ||
||   41%,   0 mins elapsed, rate=513.0k bps/s                             ||
||   49%,   0 mins elapsed, rate=593.7k bps/s                             ||
||   58%,   0 mins elapsed, rate=670.0k bps/s                             ||
||   66%,   0 mins elapsed, rate=742.6k bps/s                             ||
||   74%,   0 mins elapsed, rate=812.7k bps/s                             ||
||   83%,   0 mins elapsed, rate=880.2k bps/s                             ||
||   91%,   0 mins elapsed, rate=941.7k bps/s                             ||
|| 3.0 GB of memory is needed for index building.                         ||
|| Build the index...                                                     ||
||    8%,   0 mins elapsed, rate=5.4k bps/s                               ||
||   16%,   0 mins elapsed, rate=10.8k bps/s                              ||
||   24%,   0 mins elapsed, rate=16.1k bps/s                              ||
||   33%,   0 mins elapsed, rate=21.4k bps/s                              ||
||   41%,   0 mins elapsed, rate=26.7k bps/s                              ||
||   49%,   0 mins elapsed, rate=31.9k bps/s                              ||
||   58%,   0 mins elapsed, rate=37.1k bps/s                              ||
||   66%,   0 mins elapsed, rate=42.3k bps/s                              ||
||   74%,   0 mins elapsed, rate=47.4k bps/s                              ||
||   83%,   0 mins elapsed, rate=52.5k bps/s                              ||
||   91%,   0 mins elapsed, rate=57.6k bps/s                              ||
|| Save current index block...                                            ||
||  [ 0.0% finished ]                                                     ||
||  [ 10.0% finished ]                                                    ||
||  [ 20.0% finished ]                                                    ||
||  [ 30.0% finished ]                                                    ||
||  [ 40.0% finished ]                                                    ||
||  [ 50.0% finished ]                                                    ||
||  [ 60.0% finished ]                                                    ||
||  [ 70.0% finished ]                                                    ||
||  [ 80.0% finished ]                                                    ||
||  [ 90.0% finished ]                                                    ||
||  [ 100.0% finished ]                                                   ||
||                                                                        ||
||                      Total running time: 0.3 minutes.                  ||
||               Index reference_index was successfully built.            ||
||                                                                        ||
\\========================================================================//
```

The generated index files were saved to the current working directory. Rsubread creates a hash table for indexing the reference genome. Keys in the hash table are the 16bp sequences and hash values are their corresponding chromosomal locations. Color

space index can be built by setting the `colorsapce` argument to `TRUE`.

A unique feature of Rsubread is that it allows users to control the computer memory usage in read mapping process. Users can do this by tuning the amount of memory (in MB) to be used in read mapping.

Step 2: read mapping

After the index was successfully built, we map the read dataset (including 1,000 reads) to the reference sequence:

```
> reads <- system.file("extdata","reads.txt.gz",package="Rsubread")
> align.stat <- align(index="reference_index",readfile1=reads,output_file="alignResults.BAM",phredOffset=64)

WARNING: your system seems to be 32-bit. Rsubread supports 32-bit systems to a very limited level.
It is highly recommended to run Rsubread on a 64-bit system to avoid errors.


        ==========     _____ _    _ ____  _____  ______          _____
        =====         / ____| |  | |  _ \|  __ \|  ____|   /\   |  __ \
          =====      | (___ | |  | | |_) | |__) | |__     /  \  | |  | |
            ====      \___ \| |  | |  _ <|  _  /|  __|   / /\ \ | |  | |
              ====    ____) | |__| | |_) | | \ \| |____ / ____ \| |__| |
        ==========   |_____/ \____/|____/|_|  \_\______/_/    \_\_____/
       Rsubread 2.12.3

//================================= setting ==================================\\
||                                                                            ||
|| Function      : Read alignment (RNA-Seq)                                   ||
|| Input file    : reads.txt.gz                                               ||
|| Output file   : alignResults.BAM (BAM)                                     ||
|| Index name    : reference_index                                            ||
||                                                                            ||
||                    ------------------------------------                    ||
||                                                                            ||
||                              Threads : 1                                   ||
||                          Phred offset : 64                                 ||
||                             Min votes : 3 / 10                             ||
||                         Max mismatches : 3                                 ||
||                       Max indel length : 5                                 ||
||            Report multi-mapping reads : yes                                ||
|| Max alignments per multi-mapping read : 1                                  ||
||                                                                            ||
\\============================================================================//

//================ Running (06-Mar-2023 23:44:26, pid=10156) =================\\
||                                                                            ||
|| Check the input reads.                                                     ||
|| The input file contains base space reads.                                  ||
|| Initialise the memory objects.                                             ||
|| Estimate the mean read length.                                             ||
|| The range of Phred scores observed in the data is [2,34]                   ||
|| Create the output BAM file.                                                ||
|| Check the index.                                                           ||
|| Init the voting space.                                                     ||
|| Global environment is initialised.                                         ||
|| Load the 1-th index block...                                               ||
|| The index block has been loaded.                                           ||
|| Start read mapping in chunk.                                               ||
```

```
||                                                                      ||
||                          Completed successfully.                     ||
||                                                                      ||
\\======================================    ======================================//

//================================   Summary ================================\\
||                                                                      ||
||                 Total reads : 1,000                                  ||
||                      Mapped : 904 (90.4%)                            ||
||             Uniquely mapped : 904                                    ||
||               Multi-mapping : 0                                      ||
||                                                                      ||
||                    Unmapped : 96                                     ||
||                                                                      ||
||                      Indels : 0                                      ||
||                                                                      ||
||                Running time : 5.1 minutes                            ||
||                                                                      ||
\\======================================================================//
```

Map paired-end reads:

```
> reads1 <- system.file("extdata","reads1.txt.gz",package="Rsubread")
> reads2 <- system.file("extdata","reads2.txt.gz",package="Rsubread")
> align.stat2 <- align(index="reference_index",readfile1=reads1,readfile2=reads2,
+ output_file="alignResultsPE.BAM",phredOffset=64)

WARNING: your system seems to be 32-bit. Rsubread supports 32-bit systems to a very limited level.
It is highly recommended to run Rsubread on a 64-bit system to avoid errors.


        ==========     _____ _    _ ____  _____  ______          _____
        =====         / ____| |  | |  _ \|  __ \|  ____|   /\   |  __ \
          =====      | (___ | |  | | |_) | |__) | |__     /  \  | |  | |
            ====      \___ \| |  | |  _ <|  _  /|  __|   / /\ \ | |  | |
              ====    ____) | |__| | |_) | | \ \| |____ / ____ \| |__| |
        ==========   |_____/ \____/|____/|_|  \_\______/_/    \_\_____/
       Rsubread 2.12.3

//================================ setting ================================\\
||                                                                      ||
|| Function      : Read alignment (RNA-Seq)                             ||
|| Input file 1  : reads1.txt.gz                                        ||
|| Input file 2  : reads2.txt.gz                                        ||
|| Output file   : alignResultsPE.BAM (BAM)                             ||
|| Index name    : reference_index                                      ||
||                                                                      ||
||                    ------------------------------------              ||
||                                                                      ||
||                             Threads : 1                              ||
||                         Phred offset : 64                            ||
||               # of extracted subreads : 10                           ||
||                         Min read1 vote : 3                           ||
||                         Min read2 vote : 1                           ||
||                      Max fragment size : 600                         ||
||                      Min fragment size : 50                          ||
||                         Max mismatches : 3                           ||
||                       Max indel length : 5                           ||
||            Report multi-mapping reads : yes                          ||
|| Max alignments per multi-mapping read : 1                            ||
||                                                                      ||
\\======================================================================//
```

```
//================ Running (06-Mar-2023 23:49:30, pid=10156) ================\\
||                                                                            ||
|| Check the input reads.                                                     ||
|| The input file contains base space reads.                                  ||
|| Initialise the memory objects.                                             ||
|| Estimate the mean read length.                                             ||
|| The range of Phred scores observed in the data is [2,34]                   ||
|| Create the output BAM file.                                                ||
|| Check the index.                                                           ||
|| Init the voting space.                                                     ||
|| Global environment is initialised.                                         ||
|| Load the 1-th index block...                                               ||
|| The index block has been loaded.                                           ||
|| Start read mapping in chunk.                                               ||
||                                                                            ||
||                           Completed successfully.                          ||
||                                                                            ||
\\==================================    ==================================//

//================================   Summary ================================\\
||                                                                            ||
||             Total fragments : 1,000                                        ||
||                      Mapped : 909 (90.9%)                                  ||
||             Uniquely mapped : 909                                          ||
||               Multi-mapping : 0                                            ||
||                                                                            ||
||                    Unmapped : 91                                           ||
||                                                                            ||
||             Properly paired : 897                                          ||
||         Not properly paired : 12                                           ||
||                   Singleton : 10                                           ||
||                    Chimeric : 0                                            ||
||       Unexpected strandness : 0                                            ||
||  Unexpected fragment length : 2                                            ||
||       Unexpected read order : 0                                            ||
||                                                                            ||
||                       Indels : 0                                           ||
||                                                                            ||
||                Running time : 5.1 minutes                                  ||
||                                                                            ||
||   NOTE : No enough read-pairs to derive expected fragment length.          ||
||                                                                            ||
\\============================================================================//
```

3 Counting mapped reads for genomic features

The `featureCounts` function is a general-purpose read summarization function that assigns mapped reads (RNA-seq or gDNA-seq reads) to genomic features such as genes, exons, promoters, gene bodies and genomic windows.

This function takes as input a set of files that contain read mapping results and an annotation file that includes genomic features. It automatically detects the format of input read files (supported formats include SAM and BAM). Input reads can be name-sorted or location-sorted. Users do not need to resort the reads before feeding them to `featureCounts`.

In-built NCBI RefSeq gene annotations for genomes mm9, mm10, hg19 and hg38 are provided for convenience. These annotations include chromosomal coordinates of exons

of each gene. When these annotations are used for summarization, only reads overlapping with exons will be counted by `featureCounts`. Users can use `getInBuiltAnnotation` function to retrieve these annotations.

Below gives the example code of assigning reads and fragments, generated in the last section, to two artificial genes. Assign single end reads to genes:

```
> ann <- data.frame(
+ GeneID=c("gene1","gene1","gene2","gene2"),
+ Chr="chr_dummy",
+ Start=c(100,1000,3000,5000),
+ End=c(500,1800,4000,5500),
+ Strand=c("+","+","-","-"),
+ stringsAsFactors=FALSE)
> ann

  GeneID       Chr Start  End Strand
1  gene1 chr_dummy   100  500      +
2  gene1 chr_dummy  1000 1800      +
3  gene2 chr_dummy  3000 4000      -
4  gene2 chr_dummy  5000 5500      -

> fc_SE <- featureCounts("alignResults.BAM",annot.ext=ann)

        ==========     _____ _    _ ____  _____  ______          _____
        =====         / ____| |  | |  _ \|  __ \|  ____|   /\   |  __ \
          =====      | (___ | |  | | |_) | |__) | |__     /  \  | |  | |
            ====      \___ \| |  | |  _ <|  _  /|  __|   / /\ \ | |  | |
              ====    ____) | |__| | |_) | | \ \| |____ / ____ \| |__| |
        ==========   |_____/ \____/|____/|_|  \_\______/_/    \_\_____/
       Rsubread 2.12.3

//========================== featureCounts setting ===========================\\
||                                                                            ||
||             Input files : 1 BAM file                                       ||
||                                                                            ||
||                           alignResults.BAM                                 ||
||                                                                            ||
||              Paired-end : no                                               ||
||        Count read pairs : no                                               ||
||              Annotation : R data.frame                                     ||
||      Dir for temp files : .                                                ||
||                 Threads : 1                                                ||
||                   Level : meta-feature level                               ||
||      Multimapping reads : counted                                          ||
|| Multi-overlapping reads : not counted                                      ||
||   Min overlapping bases : 1                                                ||
||                                                                            ||
\\============================================================================//

//================================= Running ==================================\\
||                                                                            ||
|| Load annotation file .Rsubread_UserProvidedAnnotation_pid10156 ...         ||
||    Features : 4                                                            ||
||    Meta-features : 2                                                       ||
||    Chromosomes/contigs : 1                                                 ||
||                                                                            ||
|| Process BAM file alignResults.BAM...                                       ||
||    Single-end reads are included.                                          ||
||    Total alignments : 1000                                                 ||
||    Successfully assigned alignments : 31 (3.1%)                            ||
||    Running time : 0.00 minutes                                             ||
```

```
||                                                                            ||
|| Write the final count table.                                               ||
|| Write the read assignment summary.                                         ||
||                                                                            ||
\\============================================================================//

> fc_SE

$counts
      alignResults.BAM
gene1               14
gene2               17

$annotation
  GeneID                 Chr     Start       End Strand Length
1  gene1 chr_dummy;chr_dummy  100;1000  500;1800    +;+   1202
2  gene2 chr_dummy;chr_dummy 3000;5000 4000;5500    -;-   1502

$targets
[1] "alignResults.BAM"

$stat
                         Status alignResults.BAM
1                      Assigned               31
2           Unassigned_Unmapped               96
3         Unassigned_Read_Type                0
4         Unassigned_Singleton                0
5     Unassigned_MappingQuality               0
6            Unassigned_Chimera               0
7     Unassigned_FragmentLength               0
8         Unassigned_Duplicate                0
9       Unassigned_MultiMapping               0
10        Unassigned_Secondary                0
11         Unassigned_NonSplit                0
12       Unassigned_NoFeatures              873
13 Unassigned_Overlapping_Length              0
14         Unassigned_Ambiguity               0
```

Assign fragments (read pairs) to the two genes:

```
> fc_PE <- featureCounts("alignResultsPE.BAM",annot.ext=ann,isPairedEnd=TRUE)

        ==========     _____ _    _ ____  _____  ______          _____
        =====         / ____| |  | |  _ \|  __ \|  ____|   /\   |  __ \
          =====      | (___ | |  | | |_) | |__) | |__     /  \  | |  | |
            ====      \___ \| |  | |  _ <|  _  /|  __|   / /\ \ | |  | |
              ====    ____) | |__| | |_) | | \ \| |____ / ____ \| |__| |
        ==========   |_____/ \____/|____/|_|  \_\______/_/    \_\_____/
       Rsubread 2.12.3

//========================== featureCounts setting ===========================\\
||                                                                            ||
||             Input files : 1 BAM file                                       ||
||                                                                            ||
||                           alignResultsPE.BAM                               ||
||                                                                            ||
||              Paired-end : yes                                              ||
||        Count read pairs : yes                                              ||
||              Annotation : R data.frame                                     ||
||      Dir for temp files : .                                                ||
||                 Threads : 1                                                ||
||                   Level : meta-feature level                               ||
```

```
||      Multimapping reads : counted                                        ||
|| Multi-overlapping reads : not counted                                     ||
||   Min overlapping bases : 1                                              ||
||                                                                          ||
\\==========================================================================//

//================================= Running ================================\\
||                                                                          ||
|| Load annotation file .Rsubread_UserProvidedAnnotation_pid10156 ...       ||
||    Features : 4                                                          ||
||    Meta-features : 2                                                     ||
||    Chromosomes/contigs : 1                                               ||
||                                                                          ||
|| Process BAM file alignResultsPE.BAM...                                   ||
||    Paired-end reads are included.                                        ||
||    Total alignments : 1000                                               ||
||    Successfully assigned alignments : 35 (3.5%)                          ||
||    Running time : 0.00 minutes                                           ||
||                                                                          ||
|| Write the final count table.                                             ||
|| Write the read assignment summary.                                       ||
||                                                                          ||
\\==========================================================================//
```

```
> fc_PE
```

```
$counts
      alignResultsPE.BAM
gene1                 16
gene2                 19

$annotation
  GeneID                 Chr     Start       End Strand Length
1  gene1 chr_dummy;chr_dummy  100;1000  500;1800    +;+   1202
2  gene2 chr_dummy;chr_dummy 3000;5000 4000;5500    -;-   1502

$targets
[1] "alignResultsPE.BAM"

$stat
                         Status alignResultsPE.BAM
1                      Assigned                35
2           Unassigned_Unmapped                91
3          Unassigned_Read_Type                 0
4          Unassigned_Singleton                 0
5     Unassigned_MappingQuality                 0
6            Unassigned_Chimera                 0
7     Unassigned_FragmentLength                 0
8          Unassigned_Duplicate                 0
9       Unassigned_MultiMapping                 0
10         Unassigned_Secondary                 0
11          Unassigned_NonSplit                 0
12        Unassigned_NoFeatures               874
13 Unassigned_Overlapping_Length                 0
14         Unassigned_Ambiguity                 0
```

4 Quantifying 10x scRNA-seq data

The `cellCounts` function can be used to quantify the scRNA-seq data generated by the 10x Genomics Chromium platform. It employs the seed-and-vote strategy to align reads

to a reference genome, collapses reads to UMIs (Unique Molecular Identifiers) and then assigns UMIs to genes based on the `featureCounts` program. `cellCounts` starts with processing raw reads and finishes with outputting a UMI count matrix and also other information. It is able to take both BCL and FASTQ format reads as input. When input format is BCL, `cellCounts` directly processes reads from the raw data files instead of converting them into FASTQ reads before processing. `cellCounts` supports barcode correction and whitelisting. It utilizes a barcode whitelist, a list of barcode sequences included in the Chromium assay kit that can be freely downloaded from 10x Genomics website, to detect valid barcodes from experimental data. When matching cell barcodes observed in a 10x dataset against the barcode whitelist, `cellCounts` allows for one base mismatch to account for sequencing errors so as to detect more valid cell barcodes. After obtaining UMI counts for each gene in each cell, `cellCounts` uses the EmptyDrops algorithm to call valid cell barcodes. `cellCounts` reports both high-confidence and rescued cells. The rescued cells have low total UMI count but are found to have a gene expression profile that is distinct from that of ambient RNAs. Below is an example of using `cellCounts` to quantify a very small 10x scRNA-seq dataset.

```
> if(.Platform$OS.type != "windows") {
+   md5.zip <- "ffd5036b36e25e9b61efc412e71820dd"
+   URL <- "https://shilab-bioinformatics.github.io/cellCounts-Example/cellCounts-Example.zip"
+   temp.file <- tempfile()
+   temp.dir <- tempdir()
+   downloaded <- tryCatch({
+       download.file(URL, destfile = temp.file)
+       tools::md5sum(temp.file) %in% md5.zip
+     },
+     error = function(cond){
+       return(FALSE)
+     }
+   )
+   if(!downloaded) cat("Unable to download the file.\n")
+ } else downloaded <- FALSE
> if(downloaded){
+   unzip(temp.file, exdir=paste0(temp.dir,"/cellCounts-Example"))
+   library(Rsubread)
+   buildindex(paste0(temp.dir,"/chr1"),
+     paste0(temp.dir,"/cellCounts-Example/hg38_chr1.fa.gz"))
+ }

WARNING: your system seems to be 32-bit. Rsubread supports 32-bit systems to a very limited level.
It is highly recommended to run Rsubread on a 64-bit system to avoid errors.


        ==========     _____ _    _ ____  _____  ______          _____
        =====         / ____| |  | |  _ \|  __ \|  ____|   /\   |  __ \
          =====      | (___ | |  | | |_) | |__) | |__     /  \  | |  | |
            ====      \___ \| |  | |  _ <|  _  /|  __|   / /\ \ | |  | |
              ====    ____) | |__| | |_) | | \ \| |____ / ____ \| |__| |
        ==========   |_____/ \____/|____/|_|  \_\______/_/    \_\_____/
       Rsubread 2.12.3

//================================= setting ==================================\\
||                                                                            ||
||                Index name : chr1                                           ||
||               Index space : base space                                     ||
||               Index split : no-split                                       ||
```

```
||          Repeat threshold : 100 repeats                                ||
||              Gapped index : no                                         ||
||                                                                        ||
||       Free / total memory : 4.3GB / 8.0GB                              ||
||                                                                        ||
||               Input files : 1 file in total                            ||
||                             o hg38_chr1.fa.gz                          ||
||                                                                        ||
\\========================================================================//

//================================ Running =================================\\
||                                                                        ||
|| Check the integrity of provided reference sequences ...                ||
|| There were 2 notes for reference sequences.                            ||
|| The notes can be found in the log file, '/tmp/RtmpGUsLkG/chr1.log'.    ||
|| Scan uninformative subreads in reference sequences ...                 ||
|| 51717 uninformative subreads were found.                               ||
|| These subreads were excluded from index building.                      ||
|| Estimate the index size...                                             ||
||    8%,   0 mins elapsed, rate=3009.9k bps/s                            ||
||   16%,   0 mins elapsed, rate=2928.1k bps/s                            ||
||   24%,   0 mins elapsed, rate=2870.4k bps/s                            ||
||   33%,   0 mins elapsed, rate=2852.9k bps/s                            ||
||   41%,   0 mins elapsed, rate=2882.7k bps/s                            ||
||   49%,   0 mins elapsed, rate=2891.7k bps/s                            ||
||   58%,   0 mins elapsed, rate=3193.1k bps/s                            ||
||   66%,   0 mins elapsed, rate=3163.9k bps/s                            ||
||   74%,   0 mins elapsed, rate=3134.6k bps/s                            ||
||   83%,   1 mins elapsed, rate=3120.8k bps/s                            ||
||   91%,   1 mins elapsed, rate=3114.9k bps/s                            ||
|| 4.2 GB of memory is needed for index building.                         ||
|| Build the index...                                                     ||
||    8%,   0 mins elapsed, rate=1091.0k bps/s                            ||
||   16%,   0 mins elapsed, rate=1147.7k bps/s                            ||
||   24%,   0 mins elapsed, rate=1172.0k bps/s                            ||
||   33%,   1 mins elapsed, rate=1163.7k bps/s                            ||
||   41%,   1 mins elapsed, rate=1175.3k bps/s                            ||
||   49%,   1 mins elapsed, rate=1200.5k bps/s                            ||
||   58%,   1 mins elapsed, rate=1334.8k bps/s                            ||
||   66%,   2 mins elapsed, rate=1260.0k bps/s                            ||
||   74%,   2 mins elapsed, rate=1244.3k bps/s                            ||
||   83%,   2 mins elapsed, rate=1224.3k bps/s                            ||
||   91%,   3 mins elapsed, rate=1221.5k bps/s                            ||
|| Save current index block...                                            ||
||  [ 0.0% finished ]                                                     ||
||  [ 10.0% finished ]                                                    ||
||  [ 20.0% finished ]                                                    ||
||  [ 30.0% finished ]                                                    ||
||  [ 40.0% finished ]                                                    ||
||  [ 50.0% finished ]                                                    ||
||  [ 60.0% finished ]                                                    ||
||  [ 70.0% finished ]                                                    ||
||  [ 80.0% finished ]                                                    ||
||  [ 90.0% finished ]                                                    ||
||  [ 100.0% finished ]                                                   ||
||                                                                        ||
||                      Total running time: 7.2 minutes.                  ||
||             Index /tmp/RtmpGUsLkG/chr1 was successfully built.         ||
||                                                                        ||
\\========================================================================//
```

```
> if(downloaded){
+   sample.sheet <- data.frame(
+     BarcodeUMIFile = paste0(temp.dir,"/cellCounts-Example/reads_R1.fastq.gz"),
```

```
+     ReadFile = paste0(temp.dir,"/cellCounts-Example/reads_R2.fastq.gz"),
+     SampleName="Example", stringsAsFactors=FALSE
+   )
+   counts <- cellCounts(paste0(temp.dir,"/chr1"), sample.sheet,  nthreads=1,
+     input.mode="FASTQ", annot.inbuilt="hg38")
+ }

NCBI RefSeq annotation for hg38 (build 38.2) is used.
Found 3 known cell barcode sets.
Testing the cell barcodes in 3M-february-2018.txt.gz.
Loaded 6794880 cell barcodes from /Users/biocbuild/.Rsubread/cellCounts/3M-february-2018.txt.gz
Cell barcode supporting rate : 100.0%.
Found cell-barcode list '3M-february-2018.txt.gz' for the input data: supported by 100.0% reads.
Number of chromosomes/contigs matched between reference sequences and gene annotation is 1.

        ==========     _____ _    _ ____  _____  ______          _____
        =====         / ____| |  | |  _ \|  __ \|  ____|   /\   |  __ \
          =====      | (___ | |  | | |_) | |__) | |__     /  \  | |  | |
            ====      \___ \| |  | |  _ <|  _  /|  __|   / /\ \ | |  | |
              ====    ____) | |__| | |_) | | \ \| |____ / ____ \| |__| |
        ==========   |_____/ \____/|____/|_|  \_\______/_/    \_\_____/
       Rsubread 2.12.3

//========================== cellCounts settings ============================\\
||                                                                            ||
||          Index : /tmp/RtmpGUsLkG/chr1                                      ||
||     Input mode : FASTQ files                                               ||
||                                                                            ||
\\============================================================================//

//================ Running (07-Mar-2023 00:02:23, pid=10156) =================\\
||                                                                            ||
|| Sort the 28395 genes...                                                    ||
|| Load the 1-st index block...                                               ||
|| The index block has been loaded. Now map the reads...                      ||
||                                                                            ||
||                                                                            ||
|| Generate UMI count tables...                                               ||
||                                                                            ||
\\============================================================================//

Perform cell rescuing for sample 1 ...
Note: not enough cells in the data for performing cell rescuing.

The cellCounts program has finished successfully.

> if(downloaded) print(counts$sample.info)

  SampleName TotalCells HighConfidenceCells RescuedCells TotalUMI MinUMI
1    Example        100                 100            0   112846    268
  MedianUMI MaxUMI MeanUMI TotalReads MappedReads AssignedReads
1    1216.5   1807 1128.46     403390      403215        394860

> if(downloaded) print(dim(counts$counts$Example))

[1] 28395   100
```

5 Finding exon junctions

The RNA-seq technology provides a unique opportunity to identify the alternative splicing events that occur during the gene transcription process. The `subjunc` function can

be used to detect exon-exon junctions. It first extracts a number of subreads (16mers) from each read, maps them to the reference genome and identifies the two best mapping locations for each read (representing potential locations of exons spanned by the read). Then, it builds a junction table including all putative junctions. Finally, it carries out a verification step to remove false positives in junction detection by realigning all the reads. The donor ('GT') and receptor sites('AG'), are required to be present when calling exon-exon junctions. Output of this function includes the discovered exon-exon junctions and also read mapping results.

6 Base quality scores

Quality scores give the probabilities of read bases being incorrectly called, which is useful for examining the quality of sequencing data. The `qualityScores` function can be used to quickly retrieve and display the quality score data extracted from a read file.

```
> x <- qualityScores(filename=reads,offset=64,nreads=1000)

qualityScores Rsubread 2.12.3

Scan the input file...
Totally 1000 reads were scanned; the sampling interval is 1.
Now extract read quality information...

Completed successfully. Quality scores for 1000 reads (equally spaced in the file) are returned.

> x[1:10,1:10]

      1  2  3  4  5  6  7  8  9 10
 [1,] 33 33 33 20 20 24 31 15 21 16
 [2,] 33 33 30 33 33 30 34 30 32 28
 [3,] 32 33 33 32 33 33 33 20 32 24
 [4,] 33 33 33 33 33 30 29 34 31 25
 [5,] 33 33 33 33 33 34 34 34 33 30
 [6,] 33 30 31 24 24 28 33 33 30 32
 [7,] 33 33 33 33 30 28 17 25 31 33
 [8,] 33 32  2  2  2  2  2  2  2  2
 [9,] 33 33 33 34 33 33 31 33 33 33
[10,] 33 33 33 33 28 24 33 33 33 28
```

7 GC content

The `atgcContent` function returns fractions of A, T, G and C bases at each base location of reads or in the entire dataset.

8 Mapping percentage

Function `propmapped` returns the proportion of mapped reads included in a SAM/BAM file. For paired end reads, it can return the proportion of mapped fragments (ie. read pairs).

```
> propmapped("alignResults.BAM")

                NumTotal NumMapped PropMapped
alignResults.BAM     1000       904      0.904
```

9 Citation

Yang Liao, Gordon K Smyth and Wei Shi (2013). The Subread aligner: fast, accurate and scalable read mapping by seed-and-vote. Nucleic Acids Research, 41(10):e108.

Yang Liao, Gordon K Smyth and Wei Shi (2014). featureCounts: an efficient general purpose program for assigning sequence reads to genomic features. Bioinformatics, 30(7):923-30

10 Authors

Wei Shi and Yang Liao
Bioinformatics Division
The Walter and Eliza Hall Institute of Medical Research
1G Royal Parade, Parkville, Victoria 3052
Australia

11 Contact

Please post to the Bioconductor Support site (`https://support.bioconductor.org/`) if you have any questions or suggestions.