splitBarcode 软件

V1.0.0

使用手册

版本号	作者	日期	新增功能
0.1.0	赵福祥	2018.10.22	支持 se 和 pe 模式，支持单 双 barcode 拆分
0.1.1	赵福祥	2018.10.25	优化速度；增加-n -m 选项分别用于控制线程个数和内存最大值；编译为静态可执行文件，无需依赖库
0.1.2	赵福祥	2018.11.05	修改特性:当用户输入内存参数时,忽略对可用内存的判断
0.1.3	赵福祥	2018.11.05	修复 BUG: 拆分结果不一致问题;SequenceStat 中 barcode 标识不一致问题
0.1.4	赵福祥	2019.03.22	修复 bug:统计二义性 reads 异常
0.1.5	赵福祥	2019.04.17	修复 bug:cycle 数过长导致整型溢出
0.1.6	赵福祥	2019.04.24	修改特性:-r 参数，更改双 barcode 反转顺序从整体反转到两个 barcode 单独反转
1.0.0	唐子淑	2020.03.16	1、修改命令格式,添加参数 split 进行 fastq 文件的 barcode 拆分； 2、单 barcode 下支持不同 barcode 长度； 3、增加日志打印文件。

目录

一、 软件简介

(一) 简介

软件名称：splitBarcode
最新版本：V1.0.0
功能：读取指定的 fastq 文件并拆分 barcode，支持 se 和 pe 模式，支持单 barcode 和双 barcode，支持 windows 和 linux 平台。
使用环境：windows 10 或 linux（centos 7.x），直接拷贝对应版本的程序文件即可使用。
使用说明：在 bin 目录下运行 splitBarcode，按下回车，可以看到程序提示如下：

```
$./bin/splitBarcode
USAGE:
  Version 1.0.0:

  cmd split <barcode> <fq> [-OPTION]

OPTION:
  -2 FILE               Fastq for PE mode.[None]
  -o DIR                Output dir for decoded fastq.[None]
  -b INT INT INT        Barcode information : startCycle, length, mismatchNum.[L
ast 10 cycle with 1 mismatch]
  -r                    Apply reverse complement of barcode sequence.[False]
  -n THREAD             Set the maximum thread numbers.[CPU number]
  -m MEMORY             Set the maximum memory(GB).[Available memory]

[ERROR] : The parameters number is not correct!
```

注意:linux 系统运行命令需要可执行权限，可以通过 chmod 755 splitBarcode 命令赋予程序可执行权限。

(二) 参数说明

本软件需要输入三个必须参数，split <barcode 文件> <fastq 文件>。split 参数代表进行 fastq 文件的 barcode 拆分， 如果只输入此三个参数，则默认 se 模式，拆分 fastq 文件的最后 10 bp，容错为 1（默认只拆分单 barcode，请确认输入正确的 barcode 文件）。
另有可选参数：

- -2 <fastq 文件> 为 2 链 fastq 文件，有此参数表示 pe 模式
- -o <输出目录> 保存拆分结果目录，请确认该目录已经存在，默认为参数 fastq 文件所在目录
- -b <起始位置 长度 容错> 三个参数以空格分隔，表示一个 barcode 拆分的信息;如果使用双 barcode，需要输入两个 -b 参数的信息；默认拆分最后 10 bp，容错为 1
- -n <最大线程数> 设置线程数，为同时处理压缩的最大线程个数，其值必须为大于 0 的整数，如 20

- -m <最大内存> 设置内存上限，注意单位为 GB，其值必须为大于 1 的整数或浮点数，如 100
- -r 此可选参数表示对 barcode 序列进行反向互补，需要用户根据实际测序情况进行选择，默认不进行反向互补

另：如果设置线程数和内存，程序会参考进行内存的分配；如果没有设置，程序会根据实际可用内存和 CPU 个数来进行内存的自动分配。

二、 使用示例

(一) 数据准备

现假设用户有数据如下:

```
-rw-r--r-- 1 bcbuild ST_BI 2.0K 2020/03/12 16:36 BarcodeV2.1.txt
-rw-r--r-- 1 bcbuild ST_BI  418 2020/02/14 20:16 BarcodeV3.0.txt
drwxr-xr-x 2 bcbuild ST_BI 4.0K 2020/03/16 17:19 bin
drwxr-xr-x 2 bcbuild ST_BI 4.0K 2020/03/16 17:21 lib
-rw-r--r-- 1 bcbuild ST_BI 123M 2020/03/17 09:44 test_read_1.fq.gz
-rw-r--r-- 1 bcbuild ST_BI 141M 2020/03/17 09:44 test_read_2.fq.gz
-rw-r--r-- 1 bcbuild ST_BI 199M 2020/03/17 09:59 test_read.fq.gz
```

其中,

- test_read_1.fq.gz 是 pe 测序中的一链数据,长度为 100bp
- test_read_2.fq.gz 是 pe 测序中的二链数据,长度为 118bp(包含 barcode 长度 10bp)
- test_read.fq.gz 是 se 测序中的一链数据,长度为 100bp(包含 barcode 长度 10bp)
- bin 和 lib 两个目录,bin 目录下的 splitBarcode 为可执行程序,lib 目录下为相关库文件。
- BarcodeV2.1.txt 是由两列数据组成的用来拆分的文本文件,第一列是 barcode 的 ID,第二列是 barcode 序列,这里长度是 10bp,两列之间用空格或 Tab 分隔,其格式如下图:

```
$cat BarcodeV2.1.txt
1       TAGGTCCGAT
2       GGACGGAATC
3       CTTACTGCCG
4       ACCTAATTGA
5       TTCGTATCCG
6       GGTAACGAGC
7       CAACGTATAA
8       ACGTCGCGTT
9       TTCTGCTAGC
10      AGGAAGATAG
11      GCTCTTGCTT
12      CAAGCACGCA
13      CGGCAATCCG
14      ATCAGGATTC
15      TCATTCCAGA
16      GATGCTGGAT
17      GTGAGTGATG
18      GAGTCAGCTG
19      TGTCTGCGAA
```

- BarcodeV3.0.txt 与 BarcodeV2.1.txt 类似,不同之处是第二列数据 barcode 长度是 20bp,其格式如下图:

```
$cat BarcodeV3.0.txt
9-9     TTCTGCTAGCTTCTGCTAGC
10-10   AGGAAGATAGAGGAAGATAG
11-11   GCTCTTGCTTGCTCTTGCTT
12-12   CAAGCACGCACAAGCACGCA
13-13   CGGCAATCCGCGGCAATCCG
14-14   ATCAGGATTCATCAGGATTC
15-15   TCATTCCAGATCATTCCAGA
16-16   GATGCTGGATGATGCTGGAT
17-17   GTGAGTGATGGTGAGTGATG
18-18   GAGTCAGCTGGAGTCAGCTG
19-19   TGTCTGCGAATGTCTGCGAA
20-20   ATTGGTACAAATTGGTACAA
21-21   CGATTGTGGTCGATTGTGGT
22-22   ACAGACTTCCACAGACTTCC
23-23   TCCACACTCTTCCACACTCT
```

(二) 示例说明

1. PE 单 barcode

一链长度 100，二链长度 108，单 barcode 在二链最后 10bp。

输入命令为:

./bin/splitBarcode split BarcodeV2.1.txt test_read_1.fq.gz -2 test_read_2.fq.gz -o result_pe_single_index -b 208 10 1 -r

命令解析:

- ./splitBarcode 程序名称
- split 第一个必需参数，代表对 fastq 文件进行拆分

- BarcodeV2.1.txt 第二个必需参数，输入的 barcode 文件
- test_read_1.fq.gz 第三个必需参数，一链的 fastq 文件
- -2 test_read_2.fq.gz 可选参数，指定二链的 fastq 文件
- -o result_pe_single_index 可选参数，指定拆分后结果的输出目录，该目录必须存在
- -b 208 10 1 可选参数，指定 barcode 拆分信息，每次三个整数的组合分别表示 barcode 起始位置,长度,和容错;此示例表示共一个 barcode，起始位置 208bp,长度 10,容错 1
- -r 表示 barcode 序列需要反向互补，一般 barcode 序列在二链末尾都需要此参数，具体是否需要反向互补由用户确定

运行过程如下图:

```
$./bin/splitBarcode  split BarcodeV2.1.txt test_read_1.fq.gz -2 test_read_2.fq.gz
-o result_pe_single_index -b 208 10 1 -r
Use PE mode
Finish write fastq, total time(s): 18.8971
```

目录 result_pe_single_index 存储此次拆分的结果，同时生成 log 目录，存放日志文件，如下：

```
$ll
total 463M
-rw-r--r-- 1 bcbuild ST_BI 2.0K 2020/03/12 16:36 BarcodeV2.1.txt
-rw-r--r-- 1 bcbuild ST_BI  418 2020/02/14 20:16 BarcodeV3.0.txt
drwxr-xr-x 2 bcbuild ST_BI 4.0K 2020/03/16 17:19 bin
drwxr-xr-x 2 bcbuild ST_BI 4.0K 2020/03/16 17:21 lib
drwxr-xr-x 2 bcbuild ST_BI 4.0K 2020/03/17 11:24 log
drwxr-xr-x 2 bcbuild ST_BI  56K 2020/03/17 11:16 result_pe_single_index
-rw-r--r-- 1 bcbuild ST_BI 123M 2020/03/17 09:44 test_read_1.fq.gz
-rw-r--r-- 1 bcbuild ST_BI 141M 2020/03/17 09:44 test_read_2.fq.gz
-rw-r--r-- 1 bcbuild ST_BI 199M 2020/03/17 09:59 test_read.fq.gz
```

2. PE 双 barcode

一链长度 100，二链长度 98，两个 barcode 在二链最后 20bp。

输入命令为：

./bin/splitBarcode split BarcodeV3.0.txt test_read_1.fq.gz -2 test_read_2.fq.gz -o result_pe_double_index -b 198 10 1 -b 208 10 1 -r

命令解析:

- ./splitBarcode 程序名称
- split 第一个必需参数，代表对 fq 文件进行拆分
- BarcodeV3.0.txt 第二个必需参数，输入的 barcode 文件
- test_read_1.fq.gz 第三个必需参数，一链的 fastq 文件
- -2 test_read_2.fq.gz 可选参数，指定二链的 fastq 文件
- -o result_pe_double_index 可选参数，指定拆分后结果的输出目录，该目录必须存在
- -b 198 10 1 -b 208 10 1 可选参数，指定 barcode 拆分信息，每次三个整数的组合分别表示 barcode 起始位置,长度,和容错;此示例表示共

两个 barcode，分别是起始位置 198bp,长度 10,容错 1 和起始位置 208bp,长度 10,容错 1

- -r 表示 barcode 序列需要反向互补，一般 barcode 序列在二链末尾都需要此参数，具体是否需要反向互补由用户确定

运行过程如下图，结果保存在目录 result_pe_double_index。

```
$./bin/splitBarcode  split BarcodeV3.0.txt test_read_1.fq.gz -2 test_read_2.fq.gz
-o result_pe_double_index -b 198 10 1 -b 208 10 1 -r
Use PE mode
Finish write fastq, total time(s): 25.8194
```

3. SE 单 barcode

一链长度 90，单 barcode 在最后 10bp。

输入命令为：

./bin/splitBarcode split BarcodeV2.1.txt test_read.fq.gz -o result_se_single_index -b 90 10 1

命令解析：

- ./splitBarcode 程序名称
- split 第一个必需参数，代表对 fq 文件进行拆分
- BarcodeV2.1.txt 第二个必需参数，输入的 barcode 文件
- test_read.fq.gz 第三个必需参数，一链的 fastq 文件(测试需要,因此使用包含 barcode 序列的二链当作一链来使用)
- -o result_se_single_index 可选参数，指定拆分后结果的输出目录，该目录必须存在
- -b 90 10 1 可选参数，指定 barcode 拆分信息，每次三个整数的组合分别表示 barcode 起始位置,长度,和容错;此示例表示共一个 barcode,起始位置 90bp,长度 10,容错 1

运行过程如下图，结果保存在目录 result_se_single_index。

```
$./bin/splitBarcode  split BarcodeV2.1.txt  test_read.fq.gz -o result_se_single_in
dex -b 90 10 1
Use SE mode
Finish write fastq, total time(s): 26.0368
```

4. SE 双 barcode

一链长度 80，双 barcode 在最后 20bp

输入命令为：

./bin/splitBarcode split BarcodeV3.0.txt test_read.fq.gz -o result_se_double_index -b 80 10 1 -b 90 10 1

命令解析：

- ./splitBarcode 程序名称
- split 第一个必需参数，代表对 fq 文件进行拆分
- BarcodeV3.0.txt 第二个必需参数，输入的 barcode 文件

- test_read.fq.gz 第三个必需参数，一链的 fastq 文件
- -o result_se_double_index 可选参数，指定拆分后结果的输出目录，该目录必须存在
- -b 80 10 1 -b 90 10 1 可选参数，指定 barcode 拆分信息，每次三个整数的组合分别表示 barcode 起始位置,长度,和容错;此示例表示共两个 barcode，分别是起始位置 80bp,长度 10,容错 1 和起始位置 90bp,长度 10,容错 1

运行过程如下图，结果保存在目录 result_se_double_index。

```
$./bin/splitBarcode  split BarcodeV3.0.txt test_read.fq.gz  -o result_se_double_in
dex -b 80 10 1 -b 90 10 1
Use SE mode
Finish write fastq, total time(s): 42.832
```

三、 查看结果

运行结束后可以在用户指定的输出目录中查看结果，结果包括拆分后的 fastq 文件，fastq 统计文件，barcode 拆分结果等文件。

以 PE 单 barcode 为例说明。查看 result_pe_single_index 目录，确认存在拆分后的 fastq 文件和 barcode 拆分结果文件:

```
$ll
total 258M
-rw-r--r-- 1 bcbuild ST_BI 1.6K 2020/03/17 11:24 BarcodeStat.txt
-rw-r--r-- 1 bcbuild ST_BI  11K 2020/03/17 11:24 G10000955E320oldDyeB_L01_100_1.fq.fqStat.txt
-rw-r--r-- 1 bcbuild ST_BI  240 2020/03/17 11:24 G10000955E320oldDyeB_L01_100_1.fq.gz
-rw-r--r-- 1 bcbuild ST_BI  12K 2020/03/17 11:24 G10000955E320oldDyeB_L01_100_2.fq.fqStat.txt
-rw-r--r-- 1 bcbuild ST_BI  247 2020/03/17 11:24 G10000955E320oldDyeB_L01_100_2.fq.gz
-rw-r--r-- 1 bcbuild ST_BI  330 2020/03/17 11:24 G10000955E320oldDyeB_L01_101_1.fq.fqStat.txt
-rw-r--r-- 1 bcbuild ST_BI   31 2020/03/17 11:24 G10000955E320oldDyeB_L01_101_1.fq.gz
-rw-r--r-- 1 bcbuild ST_BI  330 2020/03/17 11:24 G10000955E320oldDyeB_L01_101_2.fq.fqStat.txt
-rw-r--r-- 1 bcbuild ST_BI   31 2020/03/17 11:24 G10000955E320oldDyeB_L01_101_2.fq.gz
-rw-r--r-- 1 bcbuild ST_BI  24K 2020/03/17 11:24 G10000955E320oldDyeB_L01_10_1.fq.fqStat.txt
-rw-r--r-- 1 bcbuild ST_BI  33M 2020/03/17 11:24 G10000955E320oldDyeB_L01_10_1.fq.gz
```

```
-rw-r--r-- 1 bcbuild ST_BI  330 2020/03/17 11:24 G10000955E320oldDyeB_L01_999_1.fq.fqStat.txt
-rw-r--r-- 1 bcbuild ST_BI   31 2020/03/17 11:24 G10000955E320oldDyeB_L01_999_1.fq.gz
-rw-r--r-- 1 bcbuild ST_BI  330 2020/03/17 11:24 G10000955E320oldDyeB_L01_999_2.fq.fqStat.txt
-rw-r--r-- 1 bcbuild ST_BI   31 2020/03/17 11:24 G10000955E320oldDyeB_L01_999_2.fq.gz
-rw-r--r-- 1 bcbuild ST_BI  23K 2020/03/17 11:24 G10000955E320oldDyeB_L01_undecoded_1.fq.fqStat
.txt
-rw-r--r-- 1 bcbuild ST_BI 8.0M 2020/03/17 11:24 G10000955E320oldDyeB_L01_undecoded_1.fq.gz
-rw-r--r-- 1 bcbuild ST_BI  27K 2020/03/17 11:24 G10000955E320oldDyeB_L01_undecoded_2.fq.fqStat
.txt
-rw-r--r-- 1 bcbuild ST_BI 9.6M 2020/03/17 11:24 G10000955E320oldDyeB_L01_undecoded_2.fq.gz
-rw-r--r-- 1 bcbuild ST_BI 790K 2020/03/17 11:24 SequenceStat.txt
```

(一) Barcode 拆分文件

文件名：**BarcodeStat.txt**

该文件记录 barcode 拆分结果，拆分率是拆分程序的一个重要结果指标。

此文件共五列：

1、barcode 编号

2、无错拆分出来的 reads 个数

3、有容错拆分出来的 reads 个数
4、全部拆分出来的 reads 个数(即 2 列与 3 列之和)
5、全部拆分出来的 reads 个数占总 reads 的百分比

最后一列为所有拆分出来的 barcode 统计的总和，可以看出总拆分率约 93.61%。
注：二义性 reads 认为没有拆分出来，故不计入此拆分文件。

```
$cat BarcodeStat.txt
#Barcode        Correct         Corrected       Total   Percentage(%)
barcode3        0       1       1       0.000071
barcode4        1       0       1       0.000071
barcode5        9       5       14      0.000994
barcode6        22      2       24      0.001704
barcode7        13      1       14      0.000994
barcode8        33      2       35      0.002486
barcode9        176714  7690    184404  13.096159
barcode10       356343  18274   374617  26.604866
barcode11       243836  9191    253027  17.969685
barcode12       41836   1721    43557   3.093368
barcode13       63707   3750    67457   4.790718
barcode14       61122   3336    64458   4.577733
barcode15       69034   3662    72696   5.162786
barcode16       41684   1540    43224   3.069719
barcode17       58936   2043    60979   4.330658
barcode18       29773   2802    32575   2.313439
barcode19       25431   1739    27170   1.929582
barcode20       64261   3525    67786   4.814083
barcode21       8593    394     8987    0.638246
barcode22       8428    982     9410    0.668287
barcode23       6977    712     7689    0.546064
barcode28       0       1       1       0.000071
barcode34       0       2       2       0.000142
barcode35       0       1       1       0.000071
barcode36       0       1       1       0.000071
barcode46       0       1       1       0.000071
barcode49       0       1       1       0.000071
barcode52       0       1       1       0.000071
barcode55       0       1       1       0.000071
barcode58       0       1       1       0.000071
barcode65       0       1       1       0.000071
barcode67       0       1       1       0.000071
barcode69       0       1       1       0.000071
barcode70       0       1       1       0.000071
barcode71       0       1       1       0.000071
barcode80       0       1       1       0.000071
barcode87       0       1       1       0.000071
barcode88       0       1       1       0.000071
barcode90       0       2       2       0.000142
barcode96       0       1       1       0.000071
barcode100      0       1       1       0.000071
barcode105      0       1       1       0.000071
barcode107      0       2       2       0.000142
barcode113      0       1       1       0.000071
barcode117      0       8       8       0.000568
barcode124      0       3       3       0.000213
Total   1256753         61409   1318162         93.614342
```

(二) Barcode 统计文件

文件名：**SequenceStat.txt**

记录所有出现在 barcode 位置的序列统计信息；如果拆分率异常，可以通过检查此文件获取出现频率最高的序列，以分析是否给错 barcode 序列或者忘记了反转互补。

此文件共分四列：

1、barcode 序列信息

2、对应的 barcode 编号

3、barcode 序列出现次数

4、barcode 序列个数占总 reads 的百分比

```
$cat SequenceStat.txt | head  -20
#Sequence        Barcode         Count   Percentage(%)
CTATCTTCCT       barcode10       356343  25.307068
AAGCAAGAGC       barcode11       243836  17.316951
GCTAGCAGAA       barcode9        176714  12.550024
TCTGGAATGA       barcode15       69034   4.902715
TTGTACCAAT       barcode20       64261   4.563742
CGGATTGCCG       barcode13       63707   4.524397
GAATCCTGAT       barcode14       61122   4.340814
CATCACTCAC       barcode17       58936   4.185567
TGCGTGCTTG       barcode12       41836   2.971144
ATCCAGCATC       barcode16       41684   2.960349
CAGCTGACTC       barcode18       29773   2.114444
TTCGCAGACA       barcode19       25431   1.806080
ACCACAATCG       barcode21       8593    0.610265
GGAAGTCTGT       barcode22       8428    0.598547
AGAGTGTGGA       barcode23       6977    0.495498
CTATATTCCT       barcode10       6332    0.449691
ATATAATAAT       undecoded       3088    0.219306
GCTAGCAGGA       barcode9        2545    0.180743
ATATAATCAT       undecoded       2189    0.155460
```

(三) Fastq 统计文件

文件名形如：**slide_lane_barcodeID.fq.fqStat.txt**

此文件包含对应 barcode 拆分结果的统计信息，如 reads 个数，GC 含量，Q30，base 个数等指标。每一个 barcodeID 对应生成一个 fastq 统计文件。

```
#Name   result_pe_single_index/G10000955E320oldDyeB_L01_undecoded_1.fq.gz
#PhredQual      33
#ReadNum        89915
#row_readLen    100
#col    48
#BaseNum        8991500
#N_Count        2829    0.031463
#GC%    39.40
#Q10%   73.91
#Q20%   58.60
#Q30%   44.49
#EstErr%        7.095463
#Pos    A       C       G       T       N       0       1       2       3       4       5       6
1       27147   23422   14924   24421   1       1       0       0       0       3057    13389   2539
2       37531   15875   20338   16166   5       5       0       0       0       123     1798    2049
3       36439   18157   17582   17730   7       7       0       0       0       237     1994    2118
4       25640   18224   27556   18488   7       7       0       0       0       3741    10989   3497
5       37779   15017   23030   14080   9       9       0       0       0       460     1592    2160
6       11151   27311   14262   37179   12      12      0       0       0       1546    3570    8631
7       16088   23902   15234   34688   3       3       0       0       0       2732    9605    4491
8       34236   21234   18415   16023   7       7       0       0       0       313     1677    3308
9       34231   18586   15368   21724   6       6       0       0       0       190     2730    2403
10      34924   16397   17267   21281   46      46      0       0       0       447     5379    5860
11      36843   16005   17699   19291   77      77      0       0       0       188     1873    2119
12      34914   16450   17442   21095   14      14      0       0       0       2766    2090    2680
13      33579   17411   15654   23257   14      14      0       0       0       218     6252    6618
14      31384   18103   15668   24748   12      12      0       0       0       504     7130    6798
15      33676   17635   15239   23360   5       5       0       0       0       975     4919    11139
```

四、 日志文件

运行该程序，会在当前目录的 log 文件下生成日志文件，形式为 splitBarcode_年月日_时分秒-HR.txt，例如 splitBarcode_20200317_112434-HR.log，记录了程序版本，相关参数，时间等信息，以供开发人员或有需要的人员进行分析。

注：如果 log 目录不存在，则会创建 log 目录。