Fastq\_clean使用说明

*Fei lab*

*Boyce Thompson Institute for Plant Research,*

*Cornell University,*

*Tower Road, Ithaca, NY 14853,*

*United States of America,*

Jan 29, 2014

[1. 简介 3](#_Toc427598441)

[2. 下载与安装 4](#_Toc427598442)

[2.1 运行环境 4](#_Toc427598443)

[2.2 与硬件相关的参数 4](#_Toc427598444)

[2.3 从一个实例出发 4](#_Toc427598445)

[3. 处理DNA-seq和RNA-seq数据 5](#_Toc427598446)

[3.1 基本原理 5](#_Toc427598447)

[3.2 整体流程 6](#_Toc427598448)

[3.3 参数详解 8](#_Toc427598449)

[3.3.1 必须参数 8](#_Toc427598450)

[3.3.2 可选参数 9](#_Toc427598451)

[3.3.3 R脚本相关参数 10](#_Toc427598452)

[3.3.4 BWA相关参数 11](#_Toc427598453)

[3.3.5 根据测序长度调整参数 11](#_Toc427598454)

[3.3.6 根据质量控制报告调整参数 12](#_Toc427598455)

[4. 质量控制信息详解 13](#_Toc427598456)

[4.1 获取质量控制信息 13](#_Toc427598457)

[4.2 使用质量控制信息 15](#_Toc427598458)

[5. 处理small RNA-seq数据 16](#_Toc427598459)

[5.1 sRNA-seq测序接头的确定 16](#_Toc427598460)

[6. 案例分析 17](#_Toc427598461)

[6.1 rRNA污染过多 17](#_Toc427598462)

[6.1.1软件版本和数据 17](#_Toc427598463)

[6.1.2问题解决流程 17](#_Toc427598464)

[6.1.3看看无法配对的读段 19](#_Toc427598465)

[6.1.4总结 20](#_Toc427598466)

[6.2 拼接的转录组是否还有污染 20](#_Toc427598467)

[6.3 读段比对到参考基因组比例低 20](#_Toc427598468)

[6.3.1软件版本和数据 20](#_Toc427598469)

[6.3.2问题分析 21](#_Toc427598470)

[6.3.3问题解决 23](#_Toc427598471)

[6.3.4 总结 23](#_Toc427598472)

[6.4 关于pattern的检查 23](#_Toc427598473)

[6.4.1 pattern序列介绍 23](#_Toc427598474)

[7. 常见运行错误 24](#_Toc427598475)

[8. 参考文献 25](#_Toc427598476)

**用户须知：**

本程序下载地址为<https://app.box.com/s/v7sjjmjo56r71k8bvqkv>或<http://pan.baidu.com/s/1sjllrcD>（中国用户）。

版权归文章作者所有。用户可下载并在学术研究中使用本流程，但必须引用我们已发表的文章。

Zhang M, Sun H, Fei Z, Zhan F, Gong X, Gao S: Fastq\_clean: An optimized pipeline to clean the Illumina sequencing data with quality control. In: Bioinformatics and Biomedicine (BIBM), 2014 IEEE International Conference on: 2014. IEEE: 44-48.

不允许任何商业性应用或二次开发。如有任何问题或程序出现错误，请联系：

C.J.Chen（陈程杰）

Email :chengjiechen@stu.scau.edu.cn

## 简介

Fastq\_clean是一个依据Illumina平台测序数据特点优化而成的处理流程。其功能是清理原始测序数据（raw data）中的低质量碱基、测序接头、rRNA和病毒等污染。

Fastq\_clean v1.0仅处理DNA-seq和RNA-seq数据，从v2.0开始处理small RNA-seq数据。Fastq\_clean处理双端测序数据的策略是，分别按照处理正向和反向单端测序数据，然后再提取配对的数据。

## 下载与安装

### 2.1 运行环境

1）需运行于Linux操作系统

2）预装Perl，一般情况下Linux自带

3）程序BWA（0.5.7-r1310）及Samtools（0.1.18），已包含于本软件包内

4）一般情况下，仅需要安装R/Bioconductor，如下：

安装R:

|  |
| --- |
| # RPM软件管理机制（如Fedora）  sudo yum install R-devel  # DPKG软件管理机制（如Ubuntu）  sudo apt-get install r-base-dev |

安装Bioconductor:

|  |
| --- |
| R  source("http://bioconductor.org/biocLite.R");  biocLite();  biocLite("ShortRead"); |

### 2.2 与硬件相关的参数

Fastq\_clean使用最小的硬件资源，它提供两个参数thread\_num和read\_PerYield分别控制所需的线程数量和内存大小，用户可以根据实际硬件配置来设定这2个参数。内存大小的计算方式是：读段数量\*4\*100=所需内存大小。举例，5M读段需要至少（5M\*4\*100）2G内存。fastq\_clean默认使用8个线程和2G内存。**注意，如果设定内存过小，会大大延长数据处理时间。**

### 2.3 从一个实例出发

下载 fastq\_clean-X.Y（X.Y是fastq\_clean的版本号，当前为1.0）。通过运行以下命令，处理已发表的菊花转录组数据集中的六个数据,这六个数据使用了illumina单端101 bp链特异性测序。

1）下载程序后，更改目录名为 “fastq\_clean”，此目录为工作目录，记作$work\_dir:

|  |
| --- |
| mv fastq\_clean-X.Y fastq\_clean |

2）进入工作目录并赋予可执行权限:

|  |
| --- |
| cd fastq\_clean  chmod +x -R . |

3）确认两个数据库文件都在

|  |
| --- |
| ls ./databases/\*.fasta  ./databases/rRNA\_silva111.fasta ./databases/vrl\_genbank.fasta |

4）从NCBI SRA数据库下载数据集并转换为fastq格式

|  |
| --- |
| wget ftp-trace.ncbi.nlm.nih.gov/sra/sra-instant/reads/ByRun/sra/SRR/SRR921/SRR921340/SRR921340.sra  wget ftp-trace.ncbi.nlm.nih.gov/sra/sra-instant/reads/ByRun/sra/SRR/SRR921/SRR921341/SRR921341.sra;  wget ftp-trace.ncbi.nlm.nih.gov/sra/sra-instant/reads/ByRun/sra/SRR/SRR921/SRR921342/SRR921342.sra;  wget ftp-trace.ncbi.nlm.nih.gov/sra/sra-instant/reads/ByRun/sra/SRR/SRR921/SRR921321/SRR921321.sra;  wget ftp-trace.ncbi.nlm.nih.gov/sra/sra-instant/reads/ByRun/sra/SRR/SRR921/SRR921322/SRR921322.sra;  wget ftp-trace.ncbi.nlm.nih.gov/sra/sra-instant/reads/ByRun/sra/SRR/SRR921/SRR921324/SRR921324.sra;  nohup ./tools/fastq-dump SRR921340.sra  nohup ./tools/fastq-dump SRR921341.sra  nohup ./tools/fastq-dump SRR921342.sra  nohup ./tools/fastq-dump SRR921321.sra  nohup ./tools/fastq-dump SRR921322.sra  nohup ./tools/fastq-dump SRR921324.sra |

5）运行本流程，对6个数据文件进行读段清理

|  |
| --- |
| nohup ./illumina\_clean.pl forward.config |

## DNA-seq/RNA-seq数据清理

### 3.1 基本原理

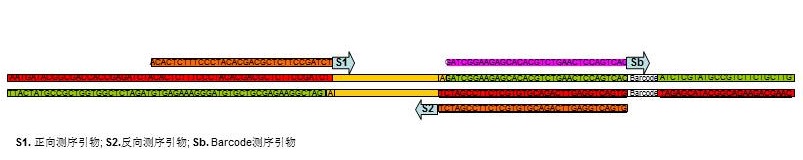


图1. Illumina接头等序列的结构

第二代测序仪（Illumina）所测的序列，无论是来自DNA-seq文库还是RNA-seq文库，自左向右依次分为3个区：5’接头（Adapter）区、目标序列区（见图1黄色区域）和3’接头（Adapter）区。对于多个样品在一个泳道（Lane）中同时测序的情况，可以使用多样品（Multiplex）技术，具体来说就是每个样品分配给一个不重复的条形码（Barcode），实际上是一个6到8位的DNA序列，测序后，用这个条形码可将不同样品分开。图5中，“S1”序列是正向测序的引物，下一个残基对应正向测序得到的序列的第一个位置；“S2”序列是反向测序的引物，下一个残基对应反向测序得到的序列的第一个位置。单端（Single end）测序指的是仅从“S1”开始测；双端（Paired end）测序指的是先从“S1”开始测，再从“S2”开始测；Barcode是根据引物“Sb”独立测序得到的。理论上，由于制备的RNA-seq文库插入长度（Insert length）的峰值常常是200bp或300bp，测序应该只得到文库中目标序列从5’端开始的部分片段。但是文库中会有少量目标序列不到测序长度（如100 bp），对于这些序列，测序可能会测到3’端接头序列，这就是所谓的接头污染。数据预处理时，如果发现接头序列过多，一般是RNA-seq文库插入长度没有控制好；如果出现大量全长的3’接头，一般是接头过量，导致了大量接头自连（Self ligation）。因此，如果使用TruSeq系列试剂盒，可以看到下面信息：

|  |
| --- |
| PCR primer 1  AATGATACGGCGACCACCGAGATCTACACTCTTTCCCTACACGACGCTCTTCCGATCT  PCR primer 2 reverse complement  AGATCGGAAGAGCACACGTCTGAACTCCAGTCACNNNNNNATCTCGTATGCCGTCTTCTGCTTG  PCR primer 2  CAAGCAGAAGACGGCATACGAGATNNNNNNGTGACTGGAGTTCAGACGTGTGCTCTTCCGATCT  PCR primer 1 reverse complement  AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGTAGATCTCGGTGGTCGCCGTATCATT  在正向测序fastq文件里面看到的是  目标序列AGATCGGAAGAGCACACGTCTGAACTCCAGTCACNNNNNNATCTCGTATGCCGTCTTCTGCTTG  在反向测序fastq文件里面看到的是  目标序列AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGTAGATCTCGGTGGTCGCCGTATCATT |

### 3.2 处理DNA-seq和RNA-seq数据

Fastq\_clean软件包在tools子目录中提供了一组程序，自动根据TruSeq的接头序列结构生成目录文件，然后**默认处理的是针对100 bp长度，从头转录组组装的RNA-seq数据；**对于其他长度的情况，要修改参数“region\_3Len”；质量较差的数据要修改参数“n\_Cutoff”；基因组组装，要修改参数“rRNA\_removal”和“virus\_removal”。

1. 将待处理数据复制或转移至工作目录$work\_dir。**必须保证$work\_dir下所有测序数据以“.fastq”为文件后缀名。正向测序数据命名格式为“sample1\_1.fastq”，反向为“sample1\_2.fastq”。**
2. 获取所有数据文件列表

|  |
| --- |
| # 进入工作目录  cd fastq\_clean  chmod +x -R .  # 执行下面脚本去处文本文件中可能存在的回车符  ./tools/remove\_cr.pl . R  # 得到目录下所有样本数据的名字（必须以fastq或fq作为后缀名）  ./tools/getFileNames.pl > samples.list  #建议用户每步处理完后查看结果，确保运行无误  cat samples.list |

1. 提取barcode序列（barcodes.list文件的第一列为sample名称，第二列为barcode）,**这步要和公司提供的barcode人工核对。**因为，这个程序提取的是数据文件中第一个read中识别的barcode，难免有测序错误，理论上应该搜索所有read识别出来的barcode，选最多的那种。

|  |
| --- |
| ./tools/getBarcodes.pl --file\_list samples.list > barcodes.list  cat barcodes.list |

如果没有提取到barcode，需要手工添加，格式如下：

|  |
| --- |
| sample1\_1 barcode1  sample2\_1 barcode2  sample1\_2 barcode1  sample2\_2 barcode2 |

1. 确认adapter\primer设计，这里以TruSeq为例

|  |
| --- |
| # 所有文件测试正向3’接头，正向测序数据都应该不为0，反向数据都为0  ./tools/checkAdapters.pl --file\_list barcodes.list --adapter AGATCGGAAGAGCACACGTCTGAACTCCAGTCAC  # 所有文件测试反向3’接头，反向测序数据都应该不为0，正向数据都为0  ./tools/checkAdapters.pl --file\_list barcodes.list --adapter AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGTA |

最好，选用正反各一个样本，人工查看一下（不是必须）

|  |
| --- |
| # 查看一个正向测序数据sample1\_1.fastq的情况  grep AGATCGGAAGAGCACACGTCTGAACTCCAGTCAC sample1\_1.fastq > 1  wc -l 1  # 查看一个反向测序数据sample1\_2.fastq的情况  grep AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGTA sample1\_2.fastq > 2  wc -l 2 |

1. 将filelist根据正反测序，分别得到2个文件

|  |
| --- |
| ./tools/split\_filelist.pl --in barcodes.list --out1 barcodes1.list --out2 barcodes2.list |

1. 生成fastqfiles1.list文件

|  |
| --- |
| ./tools/concatAdapters.pl --file\_list barcodes1.list --prefix AGATCGGAAGAGCACACGTCTGAACTCCAGTCAC --suffix ATCTCGTATGCCGTCTTCTGCTTG --with\_barcode > fastqfiles1.list  cat fastqfiles1.list |

生成fastqfiles2.list文件

|  |
| --- |
| ./tools/concatAdapters.pl --file\_list barcodes2.list --prefix AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGTAGATCTCGGTGGTCGCCGTATCATT > fastqfiles2.list  cat fastqfiles2.list |

1. 处理正向末端测序数据

|  |
| --- |
| nohup ./illumina\_clean.pl forward.config |

1. 处理反向末端测序数据

|  |
| --- |
| nohup ./illumina\_clean.pl reverse.config |

注意：7），8）两步容易报错，出现这种情况最好将程序单步运行。

1. 提取配对读段（不是必须）

由于正反向数据去除不等，会有大量读段失配，可以使用下列程序提取配对数据，失配数据输出到一个独立文件。程序运行后，得到配对的文件sample1\_1.fq sample1\_2.fq和失配的文件sample1\_S.fq

|  |
| --- |
| nohup ./tools/match\_paired.pl sample1\_1 sample1\_2 a |

注意：此处有两个可选参数值a和n，a表示sample1\_1.clean和sample1\_2.clean文件中不包含‘/1’，‘/2’模式；n则表示包含‘/1’，‘/2’模式。大部分情况下用a。

### 3.3 整体流程

整个流程分为4个部分，去除低质量（<Q20）和N、去除接头污染、去除rRNA和去除病毒（见发表文章）。其中，前两部分由R语言的ShortRead软件包完成，后两步借助BWA软件完成。

第一部分，软件完成自动识别Phred+33或Phred+64分数系统，并统一转换为Phred+33分数系统。第一部分同时将读段名字改为“sampleName-number/[12]”的格式。

由于第三四部分BWA处理，读段名字中的“[12]”会被忽略，最后一步match\_paired.pl会再次给双端测序数据分别添加“/1”和“/2”。

### 3.3 参数详解

本流程设置四类参数：必须参数、可选参数、R脚本相关选项和BWA相关选项。

#### 3.3.1 必须参数

--file\_list

Fastq\_clean被设计用于批量处理输入文件，所以用户需提供一个包含输入数据文件名的文本文件（列表文件）。该列表文件第一列为输入数据文件的文件列表，第二列为对应数据的3’接头序列。第一列应仅包括输入数据文件的文件名基名（见图1）而不带后缀。



图1.fastqfile1.list的内容

本软件包自带了两个列表文件“fastqfiles1.list”和“fastqfiles2.list”分别用于批处正向反向测序数据。“fastqfiles1.list”包括了6个文件名，来自于菊花转录组数据集。图1中3’接头序列取自IlluminaTruSeq系列试剂盒, 当前的大部分测序接头都取自此试剂盒。**实际工作中，用户可简单替换列表文件中的第一列为自己的数据文件基名，第二列为自己使用的接头序列，来运行此流程处理自己的数据。**

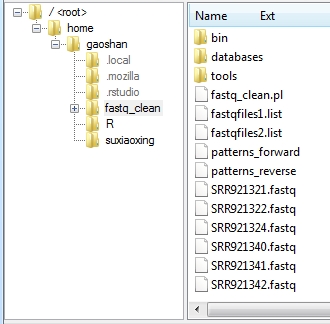


图2. 对应fastqfile1.list的实际数据文件

“fastqfiles1.list”中包括的每个文件名，都必须有一个对应的文件，并且以“.fastq”作为后缀名（见图2）。而且,所有输入文件所带数据应为Illumina原始下机数据（raw data）。**此原始数据中读段等长，不应经任何读段清理程序处理。**

#### 3.3.2 可选参数

-- run\_R

运行R脚本以去除输入数据中低质量、不确定以及接头序列对应的碱基。默认运行R脚本。**这个其实也可以算作必须参数，原始数据第一步肯定是要做这个。**

-- rRNA\_removal

运行Perl脚本以清除比对到rRNA参考序列的读段，**可以只做去除rRNA这步，但是必须保证所有数据文件后缀名是“.clean”。**

-- rRNA\_reference

若设定了“rRNA\_removal”参数，则应将命名为rRNA\_silva111.fasta的FASTA格式的rRNA参考数据库放在$work\_dir/databases文件夹下。

-- virus\_removal

运行Perl脚本以清除比对到病毒参考序列的读段,**可以只做去除病毒这步，但是必须保证所有数据文件后缀名是“.clean”。**

-- virus \_reference

若设定了“virus\_removal”参数，则应将命名为vrl\_genbank.fasta的FASTA格式的病毒参考数据库放在$work\_dir/databases文件下。

#### 3.3.3 R脚本相关参数

--pattern\_file

包含接头模式字符串列表的文件名。工作目录下有两个包含模式列表的文件“pattern\_forward”和“pattern\_reverse”。他们被设计用于去除已去除低质量、不确定和接头序列对应的碱基后的正向和反向测序数据的残余污染。这些模式也取自TruSeq RNA Sample Preparation Kits。

--quality\_Cutoff

从读段的两边末端切除质量值低于此参数的碱基。默认值为20，为质量控制最常用的阈值。

--region\_5Len

读段5’末端的低质量区长度。默认值为10。“region\_5Len”，“region\_3Len”和“n\_Cutoff”均是针对illumina 100bp 测序技术而优化设置的。

--region\_3Len

读段3’末端的低质量区长度，默认值为20。

--n\_Cutoff

去除低质量碱基后，刨除含“N”数量超过设定值的读段，默认值为2。

--adapter\_mismatch

接头与读段3’末端的错配比例，默认值为0.1。

--read\_Length

刨除长度短于设定值的读段，默认值为25。

--read\_PerYield

限定每次迭代处理的读段数目，来限定内存的占用，默认5e5。

#### 3.3.4 BWA相关参数

--max\_dist

允许最大编辑距离，默认为2。

--max\_open

允许最大空位数目，默认为1。

--max\_extension

允许最大空位长度，默认为1。

--len\_seed

取前整数个碱基为种子序列，默认为50。

--dist\_seed

种子区允许的最大编辑距离，默认为1。

--thread\_num

使用线程数目（多线程模式），默认为8。

此流程调用BWA[[2](#_ENREF_2)]将读段比对到病毒及rRNA参考序列上。命令行调用BWA程序需设定六个参数（“n”,“o”,“e”,“l”,“k”,“t”），均被相应地重命名为(“max\_dist”, “max\_open”, “max\_extension”, “len\_seed”, “dist\_seed”, “thread\_num”)。用户可通过BWA的在线手册获得更详细的参数说明（<http://bio-bwa.sourceforge.net/bwa.shtml>）。

#### 3.3.5 根据测序长度调整参数

前面介绍的很多参数，如“region\_5Len”、“region\_3Len”和“n\_Cutoff”等，都是针对Illumina 100bp 测序技术优化设置的。对于其他测序长度的数据，用户必须自行设定参数。表格1中提供了一些推荐参数，以方便用户使用。

表1. R脚本相关参数

|  |  |  |  |
| --- | --- | --- | --- |
| Length(bp) | region\_5Len | region\_3Len | n\_Cutoff |
| <=50 | 10 | 10 | 1 |
| 50-100 | 10 | 20 | 2 |
| 150-200 | 10 | 50 | 3 |
| >=250 | 10 | 70 | 4 |

表2. BWA相关参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Length(bp) | max\_dist | max\_open | max\_extension | len\_seed | dist\_seed |
| <=50 | \*2/1 | 1 | 1 | 35 | \*1/1 |
| 50-100 | \*4/2 | 1 | 1 | 50 | \*2/1 |
| 150-200 | \*6/3 | 2 | 1 | 70 | \*3/2 |
| >=250 | \*8/4 | 2 | 1 | 100 | \*4/2 |

\*左侧参数用于对数据质量要求较高的组装（无参）的应用，右侧用于对数据质量要求较低的比对（有参）的应用。

#### 3.3.6 根据质量控制报告调整参数

前面3.3.5所介绍的对于不同测序长度的数据常用的参数设置，主要是针对当前测序仪所测数据，对以往测序数据（质量不如当前）不一定适用。下面以某甘薯转录组数据为例说明如何根据质量控制报告设置参数。该转录组包括一对Illumina双端101bp测序数据，原始文件ganshu1\_1.fastq和ganshu1\_2.fastq。使用FastQC软件，得到正向测序原始数据ganshu1\_1.fastq的读段逐点质量分布图（图3A），从图中可以看到从3’端数据质量逐渐变差。通过设定参数region\_3Len为20、30和50得到不同的清理后的数据质量分布图，**可以看出参数50（图3D）得到的效果明显好于30（图3C）和20（图3B）。**

**注意：转录组从头（denovo）组装相对基于比对的有参组装需要较高的数据质量，**参数设置较严格（如50）可以提高组装质量，但是会较多损失reads，进而影响转录组定量。因此，需要根据fastq\_clean产生的质量控制报告选取一个比较适中的参数。

|  |  |
| --- | --- |
|  |  |
|  |  |

图3. 质量分布图

## 质量控制信息详解

### 4.1 获取质量控制信息

Fastq\_clean的最大特点是，它在输出清理过的数据的同时，输出大量统计信息（见质量控制信息表），帮助用户进行质量控制，通过这些信息，读者不仅清楚地了解每一步的处理情况，还可以根据这些信息定位出现问题的地方（见5案例分析）。

质量控制信息表中的前7列由文件“trimming.report”提供，第8和第9列由“cleaning.report”提供，第12和第13列见后面，第14列需要执行程序match\_paired.pl后得到。软件包的doc文件夹提供了一个excel格式的文档，用户可以把这些信息copy进该文档，得到图4和图5中的结果。

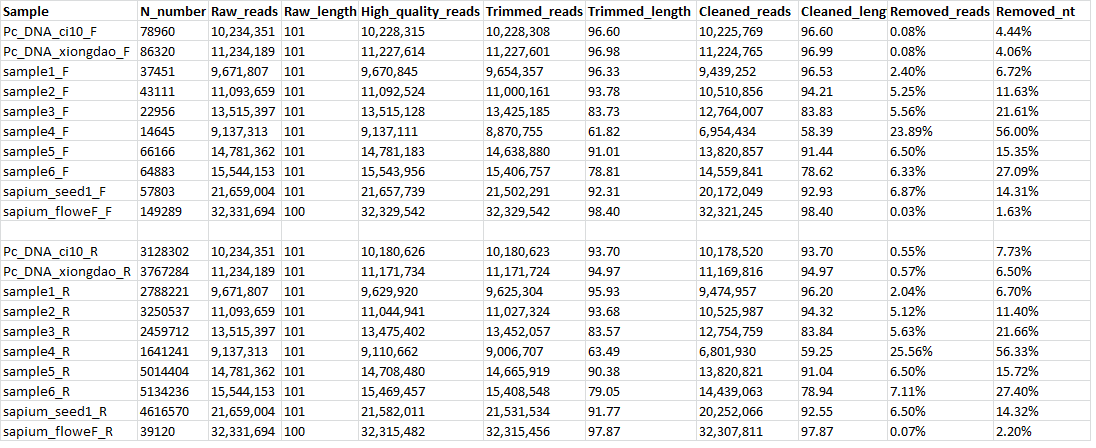


图4. 质量控制信息表一（前11列）

第1列(Sample)：样品编号，也是数据无后缀的文件名。

第2列(N\_num)：原始数据中不确定碱基（即‘N’）的总数。

第3列(Raw\_reads)：原始数据读段总数。

第4列(Raw\_length)：原始数据读段读长。

第5列(High\_quality\_reads)：高质量读段（先去除所有读段5’和3’两端低质量碱基，然后去除含nCutoff个或以上的‘N’ 的读段后剩下的读段）的数量，nCutoff默认为2(针对100bp)测序长度。

第6列(Trimmed\_reads)：高质量读段中，去除3’端测序接头后，长度大于等于readLength的Trimmed读段数量，readLength默认为25；

第7列(Trimmed\_length)：Trimmed读段的平均长度；

第8列(Cleaned\_reads)：Trimmed读段中，去除含有一定模式的读段后，剩下的Cleaned读段的数量，模式序列来自指定文件“patterns\_forward”和“patterns\_reverse”；

第9列(Cleaned\_ length)：Cleaned读段的平均长度；

第10列(Removed\_reads)：前几步清理去除的读段数所占百分比，计算公式=（Raw\_reads -Cleaned\_reads）/Raw\_reads；

第11列(Removed\_nt)：前几步清理去除的核苷酸残基所占百分比，计算公式=（Raw\_reads\*Raw\_length- Cleaned\_reads\*Clean\_length）/(Raw\_reads\*Raw\_length)。

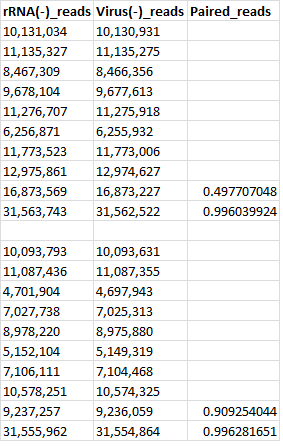


图5. 质量控制信息表二（后3列）

第12列(rRNA(-)\_reads)：去除rRNA读段后剩下的读段数量。

第13列(Virus(-)\_reads)：去除病毒读段后剩下的读段数量。

第14列(Paired\_reads)：双端测序数据中，正向或反向的读段分别经过数据清理后，能够配对的读段数量。

### 4.2 使用质量控制信息

质量控制信息表中从某一列到下一列的变化提供了丰富的信息用于质量控制: 1. 从Raw\_reads到Cleaned\_reads这几步，如果数据质量较好，Removed\_reads不超过5%，Removed\_nt不超过10%；一般情况Removed\_reads不超过10%，Removed\_nt不超过20%。**直接看这两个指标，就对数据情况大致有个印象。**

1. 从Raw\_reads到High\_quality\_reads如果减少太多，则说明测序质量有问题,特别是读段中间的高质量区有问题。**这种情况很少发生，IlluminaTM测序仪还是比较稳定的。**

2. 从High\_quality\_reads到Trimmed\_reads如果减少太多，则说明测序文库的insert size太短，有可能是测序文库insert size过短（size selection那步没控制好），也有可能是RNA-seq降解很严重（不常发生）。**这种情况很少发生，但是一旦发生后果很严重，太短的序列无法进行有效拼接，而且比对到基因组唯一性很差。**

3. 从Trimmed\_reads到Cleaned\_reads如果减少太多,则说明测序文库的接头自连情况较多。**这种情况比较常见，而且很可能来自样品RNA不足，导致接头相对过多产生接头自连。**

4. 从Cleaned\_reads到rRNA(-) reads如果减少太多,则说明测序文库中rRNA污染较多。**这种情况也比较常见，如果rRNA过多会影响RNA-seq定量。**

5. 从rRNA(-)\_reads到Virus(-)\_reads如果减少太多,则说明测序文库的病毒污染较多。**这种情况也不常见，但如果病毒过多，会影响分析结果。比如差异表达分析中，很难说明是病毒感染还是实验条件引起的基因变化，此数据不可用。**

## small RNA-seq数据清理

### 5.1 sRNA-seq测序接头的确定

sRNA-seq测序的接头确定是sRNA-seq数据分析中一个非常重要的步骤，接头如果不准确，直接导致后面无法找到正确的microRNA或者siRNA序列，这个危害是致命的，远远比DNA-seq和RNA-seq接头不准确带来的危害要大。

1）获取所有数据文件列表

|  |
| --- |
| # 进入工作目录  cd fastq\_clean  chmod +x -R .  # 执行下面脚本去处文本文件中可能存在的回车符  ./tools/remove\_cr.pl . R  # 得到目录下所有样本数据的名字（必须以fastq或fq作为后缀名）  ./tools/getFileNames.pl > samples.list  #建议用户每步处理完后查看结果，确保运行无误  cat samples.list |

2）初步查找接头

|  |
| --- |
| ./tools/smallRNA\_adapter\_finder.pl samples.list fastq > sample\_adapter.list |

人工查看结果，对比接头库，人工修正

## 案例分析

### 6.1 rRNA污染过多

#### 6.1.1软件版本和数据

软件版本：fastq\_clean v1.0。

操作系统：Fedora release 18 (Spherical Cow)。

数据来源：乌桕种子的转录组链特异性双端测序数据

数据地址：不公开

存在问题：乌桕种子反向测序数据sapium\_seed1\_R.fastq在rRNA去除那步去掉了45.61%的数据，远远大于正向测序数据sapium\_seed1\_F.fastq中rRNA的去除比例。

#### 6.1.2问题解决流程

**第一步**，得到反向测序数据sapium\_seed1\_R.fastq能对齐到rRNA数据库的读段数据sapium\_seed1\_R.mapped，得到正向测序数据sapium\_seed1\_F.fastq去除对齐到rRNA数据库的读段数据sapium\_seed1\_F.clean，这两类读段中对应的对（pair）就是反向可以对齐到rRNA，正向不能对齐到rRNA的数据。

修改fastqfiles1.list文件内容

|  |
| --- |
| sapium\_seed1\_F AGATCGGAAGAGCACACGTCTGAACTCCAGTCACAGTCAAATCTCGTATGCCGTCTTCTGCTTG |

修改fastqfiles2.list文件内容

|  |
| --- |
| sapium\_seed1\_R AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGTAGATCTCGGTGGTCGCCGTATCATT |

修改程序/bin/bwa\_remove.pl，以保留中间结果

|  |
| --- |
| #system("rm $sample.sai");  #system("rm $sample.pre.sam");  #system("rm $sample.sam");  #system("rm $sample.mapped"); |

执行fastq\_clean流程，到去除rRNA这步

|  |
| --- |
| nohup ./fastq\_clean.pl --file\_list fastqfiles1.list --run\_R --pattern\_filepatterns\_forward --rRNA\_removal --rRNA\_reference rRNA\_silva111.fasta  nohup ./fastq\_clean.pl --file\_list fastqfiles2.list --run\_R --pattern\_filepatterns\_reverse --rRNA\_removal --rRNA\_reference rRNA\_silva111.fasta |

得到正向测序去除rRNA后的读段sapium\_seed1\_F.clean，共16,873,569个，与表格里相同。得到反向测序被去除的读段sapium\_seed1\_R.mapped，共11,014,809个，与表格里相同。

得到反向可以对齐到rRNA，正向不能对齐到rRNA的数据。

|  |
| --- |
| mv sapium\_seed1\_R.mapped sapium\_seed1\_R.clean  ./tools/match\_paired.pl sapium\_seed1\_F sapium\_seed1\_R |

**第二步**，挑出一对数据对齐到NCBI核酸数据库，查看原因，找到原因后，测试所有第一步挑出的数据，以验证假说。**原因是，bowtie或bwa类软件是设计用来完成全局比对的，这里读段全长上仅仅允少量的错配（例如1到2），如果错配过多，比对失败。但是，blast是基于局部比对的，即只要局部满足要求，不看全局有多少错配，因此它可以找到更多的比对结果。**

人工查看sapium\_seed1\_F.fq和sapium\_seed1\_R.fq中数据，找到下列配对数据

|  |
| --- |
| @sapium\_seed1\_F-2/1  TGAAAGGATGTTTTCATTAATCAAGAACGAAAGTTGGGGGCTCGAAGACGATCAGATACCGTCCTAGTCTCAACCATAAACGATGCCGACCAGGGATCGGC  +  BB@DFEFFHHFHHIGHGJJJJJJJJHIIJIEIIHGIJJJJFIHIGHJEEEFFFFFCCEEEB@@BB:ACCDDEDDD@CDDDDDBBDBDB@DDDDDD?CDDD5  @sapium\_seed1\_R-2/2  CGGCGGATGTTGCTTTTAGGACTCCGCCGGCACCTTATGAGAAATCAAAGTCTTTGGGTTCCGGGGGGAGTATGGTCGCAAGGCTGAAACTTAAAGGAAT  +  DDDDDDDDBDDDDDC@>CCC?DBDDDBDC@C>CDDDEEDDCEEEDEECCCCFFFHHGJIJJJJJJIGGBFAIIGIIHDIGJJIGGIIHHHHHFFDDDC@@ |

将2个数据比对到NCBI核酸序列库，sapium\_seed1\_R-2/2可以100%比对上；而sapium\_seed1\_F-2/1最高只能达到97%，因为sapium\_seed1\_F-2/1的前3个碱基“TGA”都是错配，超过了流程规定的错配数量1，所以不能被BWA对齐到rRNA数据库。

**第三步**，下面要处理所有数据，看是否都是同一个问题

fastq格式转为fasta格式，满足blast程序输入的要求

|  |
| --- |
| ./fastq2fasta.pl sapium\_seed1\_R.fq > sapium\_seed1\_R.fa  ./fastq2fasta.pl sapium\_seed1\_F.fq > sapium\_seed1\_F.fa |

为rRNA数据库建立索引

|  |
| --- |
| nohup ./formatdb -i rRNA\_silva111.fasta -p F -o T |

将数据比对到rRNA数据库，特别注意为什么是-v 1 -b 1

|  |
| --- |
| nohup ./blastall -p blastn -i sapium\_seed1\_F.fa -d rRNA\_silva111.fasta -o sapium\_seed1\_F.paired -a 6 -F F-r 1 –q -1 -G 2 -E 1 -v 1 -b 1 -e 1e-10  nohup ./blastall -p blastn -i sapium\_seed1\_R.fa -d rRNA\_silva111.fasta -o sapium\_seed1\_R.paired -a 6 -F F-r 1 -q -1 -G 2 -E 1 -v 1 -b 1 -e 1e-6 |

将blast输出结果转换为table格式

|  |
| --- |
| nohup ./blastn\_parse\_table2.pl sapium\_seed1\_F.paired sapium\_seed1\_F.table  nohup ./blastn\_parse\_table2.pl sapium\_seed1\_R.paired sapium\_seed1\_R.table |

结果文件sapium\_seed1\_F.table包含每个查询序列的query\_start、query\_end和query\_length三列信息。用（query\_end-query\_start+1）/query\_length计算query\_coverage都不到100%，很多query\_start>= 3，与sapium\_seed1\_F-2/1都是一类问题。**因此，全部数据验证完毕。**

#### 6.1.3看看无法配对的读段

从失配数据中提取失配的反向序列，共838135条。

|  |
| --- |
| ./tools/getSingle.pl --reverse sapium\_seed1\_S.fq> reverse.fq  ./fastq2fasta.plreverse.fq>reverse.fa |

将数据比对到rRNA数据库，仅返回最好的一个比对结果

|  |
| --- |
| nohup ./blastall -p blastn -ireverse.fa -d rRNA\_silva111.fasta -o reverse.paired -a 6 -F F-r 1 -q -1-G 2 -E 1 -v 1 -b 1 -e 1e-6  ./blastn\_parse\_table2.plreverse.pairedreverse.table |

619091条序列，73.87%（619091/838135）可以比对到rRNA数据库，其中97.43%（603174/619091）的序列的query\_coverage在60%以上。仅仅统计HSP内部的错配碱基，4个及4个以下错配的HSP占94.81%（586955/619091）。根据这个统计结果，可以重新优化Fastq\_clean的参数（见表2）。

提取剩余219044条序列到文件2.fa

|  |
| --- |
| cut -f1 reverse.table>reverse.list  ./extractFromFasta.pl --inputfilereverse.fa --type list --query reverse.list --output1 1 --output2 2.fa |

将2.fa比对到NT数据库

|  |
| --- |
| nohup ./blastall -p blastn -i2.fa -d nt–o2.paired -a 6 -F F-r 1 -q -1 -G 2 -E 1 -v 5 -b 5-e 1e-10 |

#### 6.1.4总结

此流程处理的数据，反向测序比正向测序去除了较多的rRNA，主要原因是正向测序读段的5’端前3个碱基错误率较高，使很多来自rRNA的读段不能去除。

### 6.2 拼接的转录组是否还有污染

将乌桕的花和种子两个样品，4个测序文件中的数据合并后，经过trinity软件拼接得到转录组数据Trinity.fasta，共得到79738个转录本，其中，花中有254个转录本不表达，种子中有37271个转录本不表达。

然后用bowtie分别将乌桕的花正向测序文件（sapium\_flower1\_F）和种子的正向测序文件（sapium\_seed1\_F）比对到Trinity.fasta，反向数据（sapium\_flower1\_R）和sapium\_seed1\_R的对齐率只有74.55%和35.41%；反向对齐率只有74.58%和64.35%，因此推测序列中有大量污染。

|  |
| --- |
| nohup ./formatdb -i rRNA\_silva111.fasta -p F -o T  nohup ./blastall -p blastn -iTrinity.fasta -d rRNA\_silva111.fasta-oTrinity.paired -a 6 -F F -r 1 -q -1 -G 2 -E 1 -v 1 -b 1-e 1e-10  ./blastn\_parse\_table2.plTrinity.pairedTrinity.table |

所有转录本中，能够对齐到rRNA的转录本只有418条，大部分还不是rRNA。

|  |
| --- |
| bowtie-build --quiet Trinity.fasta Trinity  nohup bowtie Trinity -v 2 -p 6 -a --best --strata -q sapium\_seed1\_F.fq -S --sam-nohead sapium\_seed1\_F.pre.sam --un seed1\_F.fq  ./fastq2fasta.plseed1\_F.fq >seed1\_F.fa  nohup ./formatdb -i rRNA\_silva111.fasta -p F -o T  nohup ./blastall -p blastn -iseed1\_F.fa -d rRNA\_silva111.fasta -o seed1\_F.paired -a 6 -F F -r 1 -q -1 -G 2 -E 1 -v 1 -b 1 -e 1e-10 |

### 6.3 读段比对到参考基因组比例低

#### 6.3.1软件版本和数据

软件版本：fastq\_clean v1.0。

操作系统：Fedora release 18 (Spherical Cow)。

数据来源：人神经细胞（正向测序）6对双端测序数据

数据地址：不公开

存在问题：正向6个样本mapped ratio才40%多

|  |
| --- |
| #pan1数据的tophat运行结果  Left reads:  Input : 23821804  Mapped : 11390674 (47.8% of input)  of these: 1020992 ( 9.0%) have multiple alignments (1248 have >20)  Right reads:  Input : 23821804  Mapped : 19755317 (82.9% of input)  of these: 1400547 ( 7.1%) have multiple alignments (1285 have >20)  Unpaired reads:  Input : 1161215  Mapped : 557680 (48.0% of input)  of these: 29812 ( 5.3%) have multiple alignments (29 have >20)  65.0% overall read mapping rate. |

#### 6.3.2问题分析

检查tophat结果目录下/tmp子目录中的log，初步确认正向对齐比率低下的原因到转录组比对阶段。

直接使用bowtie实现tophat转录组比对步骤

|  |
| --- |
| bowtie -v 2 -k 60 -m 60 -p 16 ../trans\_index/transcriptome\_index ../pan1\_F.clean pan1\_F.out  #标准输出结果如下：  # reads processed: 24983019  # reads with at least one reported alignment: 10998223 (44.02%)  # reads that failed to align: 13982697 (55.97%)  # reads with alignments suppressed due to -m: 2099 (0.01%)  Reported 52889799 alignments to 1 output stream(s) |

粗略查看错配碱基主要位置，确认其主要错配碱基集中在前4位

|  |
| --- |
| cat pan1\_F.out | cut -f8|perl -F/,/ -ane 'foreach (@F){chomp;s/:.\*//;print $\_,qq/\n/}'|sort|uniq -c> mismatch.log  cat mismatch.log  4192845 #对应比对结果中无错配  27582468 0 #对应比对结果中错配碱基位置为0，即输入读段的第1位碱基  26845278 1 #对应比对结果中错配碱基位置为1，即输入读段的第2位碱基  34722 10  33735 11  …. |

对反向读段进行类似操作，对比结果，初步估计正向对齐比率低下的主要原因在于比对用的读段第一位置碱基（注意clean后的位置与clean前不同）与参考序列错配

|  |
| --- |
| head pan1\_R.out | cut -f8|perl -F/,/ -ane 'foreach (@F){chomp;s/:.\*//;print $\_,qq/\n/}'|sort|uniq -c  47963295  12674846 0  6042233 1  225365 10  236702 11  …. |

为验证以上设想，分别切除正向、反向读段文件的第1、2、3、4位碱基，并重新比对到参考序列上，查看结果

|  |
| --- |
| # 切除前1个碱基  fastx\_trimmer -Q 33 -f 2 -i pan1\_F.clean0 -o pan1\_1.clean  # 切除前2个碱基  fastx\_trimmer -Q 33 -f 3 -i pan1\_F.clean0 -o pan1\_1.clean  # 切除前3个碱基  fastx\_trimmer -Q 33 -f 4 -i pan1\_F.clean0 -o pan1\_1.clean  # 切除前4个碱基  fastx\_trimmer -Q 33 -f 5 -i pan1\_F.clean0 -o pan1\_1.clean |

对齐比率随5’端碱基切除长度变化如下表

|  |  |  |
| --- | --- | --- |
| 比对程序 | Bowtie1 | |
| 比对参数 | -v 2 -k 60 -m 60 -p 30 | |
| 参考序列 | 人类基因组注释文件提取出的转录组 | |
| 读段方向 | 5端切除碱基数 | 对齐率 |
| 正向 | 0 | 44.02 |
| 正向 | 1 | 74.23 |
| 正向 | 2 | 78.06 |
| 正向 | 3 | 79.26 |
| 正向 | 4 | 79.42 |
| 反向 | 0 | 77.17 |
| 反向 | 1 | 78.54 |
| 反向 | 2 | 79.27 |
| 反向 | 3 | 79.66 |
| 反向 | 4 | 79.89 |

至此，确认用于比对的正向读段文件第一位碱基为导致回贴率低下的最主要原因。由此查看该读段文件第一位碱基的质量值

|  |
| --- |
| perl -ne 'next if ($.%4);$con=(split(//,$\_))[0];print $con,"\n"' pan1\_F.clean | sort| uniq -c >first\_base  cat first\_base  #结果如下  902975 <  2162 ' #对应Phread+33质量值格式的质量值为6  21480 0 #对应Phread+33质量值格式的质量值为13  82037 7 #对应Phread+33质量值格式的质量值为22  23916890 B  57469 F  6 I |

#### 6.3.3问题解决

用fastx\_toolkit去除数据前4位碱基

|  |
| --- |
| fastx\_trimmer -v -Q 33 -f 5 -i pan1\_F.clean0 -o pan1\_1.clean  …… |

去除数据前4位碱基后，再次调用tophat比对到基因组，数据1-4的对齐率明显提升到80%以上，但是数据5-6只达到70%以上，再次调用bowtie将数据5-6比对到参考基因组，根据上面流程统计比对错误的位点，得到下面结果。

|  |
| --- |
| mismatch.log #来自数据pan5\_1.clean  cat mismatch.log  11701301  381703 0  288061 1  169275 10  159791 11  172325 12  174113 13  …… |

结果说明这些数据不再有错误集中的位点，即使再切除一些碱基也不可能提高对齐率。

#### 6.3.4 总结

这组数据的特点是正向测序数据的5’端前几个bp数据质量较差，而反向数据都较好。**后来通过查询实验手册得知，5’端前3个bp是认为加入用于专利保护的，以此证明分析绝对正确。**

### 6.4 关于pattern的检查

#### 6.4.1 pattern序列介绍

正向测序需要除掉的pattern

|  |
| --- |
| # P7 reverse complement(21bp)  TCGTATGCCGTCTTCTGCTTG  # PCR common(13bp)  GCTCTTCCGATCT  # PCR common reverse complement(13bp)  AGATCGGAAGAGC |

反向测序需要除掉的pattern

|  |
| --- |
| # P5 reverse complement(20bp)  TCGGTGGTCGCCGTATCATT  # PCR common(13bp)  GCTCTTCCGATCT  # PCR common reverse complement(13bp)  AGATCGGAAGAGC |

## 常见运行错误

**问题1：输入数据读段不等长，报错信息如下：**

|  |
| --- |
| Error in replaceLetterAt(seqs, at, letter) :  'x' must be rectangular (i.e. have a constant width) |

**原因：本流程只处理illumina原始数据（每行字符数必须相等）。**

**问题2：R流程运行完以后出现乱码，可以尝试运行以下代码：**

|  |
| --- |
| library(ShortRead) ;  a<-readFastq("sample1\_1.fastq");  writeFastq(a, file="sample1\_1.fq", mode="w", full=FALSE);  sample1\_1.fastq内容  @FCC3215ACXX:7:1101:1378:2068#/1  TGCGGCGAAGTCGCGGATCTTTCGACCTCCTCGAGCGCCTCCTCCCGCCCAAGACTTCGGCGAGAGAGGGACAGTGGAGTACGAGCTCGA  +  aabacceeggceghhiiiiiiiiiiiihggggedeccccZ\_abbccaaccWacccccbccac[a]aacccZ\_^aRY^bcJX`aaac\_baa |

**原因：新版本的ShortRead自1.21.0开始，将writeFastq函数中 compress参数的默认值由FALSE改为TRUE，导致默认存的是压缩格式，由于BWA能自动识别gz压缩格式，因此可以打开**

**解决：在writeFastq函数中增加compress = FALSE。**

**问题3：任何perl程序出现下面问题**

|  |
| --- |
| : No such file or directory |

**原因：程序由于在windows平台编辑，文件中多了回车符号，可以执行下面命令来去除所有文档中的回车符号。**

|  |
| --- |
| tr -d '\r' < file\_in > file\_out 或者用doc2unix filename |

**问题4：**

|  |
| --- |
| Error in writeFastq(reads, file = outfile, mode = "a", full = FALSE) :  'max\_width' must be 'integer(1)', >=0  Calls: removePatterns -> writeFastq -> writeFastq -> .Call  In addition: Warning message:  In max(c(unique(width(id(object))), unique(width(sread(object))), :  no non-missing arguments to max; returning -Inf  Execution halted |

**问题5：**

|  |
| --- |
| the quality score system (SFastqQuality=Phred+64,FastqQuality=Phred+33) is SFastqQuality  Error in .Call(.NAME, ..., PACKAGE = PACKAGE) :  negative length vectors are not allowed  Calls: trimRead -> charToRaw -> unlist -> unlist -> .Call2 -> .Call  Execution halted |

## 参考文献

1. Xu, Y., et al., *Transcriptome sequencing and whole genome expression profiling of chrysanthemum under dehydration stress.*BMC Genomics, 2013.**14**(1): p. 662.

2. Li, H. and R. Durbin, *Fast and accurate short read alignment with Burrows-Wheeler transform.*Bioinformatics, 2009.**25**(14): p. 1754-1760.