[模拟测序数据生成软件]

[V1.0]

使用手册

（如文档为软件设计说明，请把“使用手册”更改为“设计说明”）

目录

[1. 引言 2](#_Toc146553287)

[1.1 编写目的 2](#_Toc146553288)

[1.2 背景 2](#_Toc146553289)

[1.3 参考文献 4](#_Toc146553290)

[2. 总体设计 5](#_Toc146553291)

[2.1 目标 5](#_Toc146553292)

[2.2 功能 5](#_Toc146553293)

[2.3 使用方法 5](#_Toc146553294)

[3. 运行环境 8](#_Toc146553295)

[3.1 硬件 8](#_Toc146553296)

[3.2 支持软件 9](#_Toc146553297)

[4. 使用说明 9](#_Toc146553298)

[4.1 软件安装和初始化 9](#_Toc146553299)

[4.2 输入文件 11](#_Toc146553300)

[4.3 配置程序 12](#_Toc146553301)

[4.4 输出结果 13](#_Toc146553302)

[4.5 求助查询 17](#_Toc146553303)

[5. 运行说明 18](#_Toc146553304)

[5.1 程序存放位置 18](#_Toc146553305)

[5.2 运行步骤 19](#_Toc146553306)

[5.3 运行时间 20](#_Toc146553307)

[5.4 运行错误说明 20](#_Toc146553308)

[5.5 运行成功说明 21](#_Toc146553309)

# 1. 引言

## 1.1 编写目的

转录组测序技术为研究生物体如何通过基因的转录表达发挥生物学功能奠定了重要基础。为解决转录组高通量测序数据处理和分析的需要，许多生物信息学分析软件应运而生。然而，不同测序技术所产生的原始测序数据具有不同的特性（如测序质量、测序深度和reads长度等），如何能够在较短时间内根据所用测序数据的一些已知特性找到最适合数据分析需要的分析软件以及相应的最佳参数组合是转录组学分析中普遍关注的问题。

对此，我们开发了模拟测序数据生成软件（Mock Reads Generator）。该软件能够根据用户需求选择碱基随机或序列随机的两种程序运行模式，产生一套个性化的模拟转录组测序数据集。通过使用该软件产生的模拟数据集，用户能在进行真实数据的分析之前，预先快速评估出最适配真实测序数据特性的数据分析软件以及相应的最优参数组合，从而能够保证分析研究过程中所用软件和参数的适配性和准确性，同时一定程度上节省了研究者耗费在分析软件选择和参数调试上的时间，从而提高分析效率。

总之，该软件能够于用户预先给定的信息产生模拟测序数据集，帮助研究者快速评估相关转录组学数据分析工具的效果，并调试出最优的参数组合，为下游分析结果的准确性奠定重要基础。

## 1.2 背景

在分子水平上，DNA通过碱基互补配对转录为RNA，再进一步翻译成蛋白质控制着细胞中的所有过程。自从发现RNA作为基因组和蛋白质组之间的关键媒介以来，转录物鉴定和基因表达的量化一直是分子生物学中独特的核心活动。转录组是指存在于细胞中的一组RNA分子，如信使RNA （mRNA）、转运RNA （tRNA）、核糖体RNA （rRNA）和其他非编码RNA分子。通过测量mRNA水平，我们可以评估细胞如何重塑其转录组以适应现有环境（例如健康和疾病）。因此，mRNA分子是最常被研究的RNA，因为它们通过遗传密码编码蛋白质。除了编码蛋白质的mRNA外，还有多种功能性的非编码RNA （ncRNA）分子。

第二代测序或下一代测序技术（Next generation sequencing, NGS）是在第一个人类基因组序列完成以后，对更便宜和更快的测序方法需求大大增加的背景下发展起来的。NGS开创性的引入了可逆终止末端，通过边合成边测序可以对来自单个样本的数百万个DNA片段进行同时测序。这种大规模并行测序技术极大地提高了测序通量，同时降低了测序成本，可以在不到一天的时间内对整个基因组进行测序。高通量二代测序技术的引入彻底改变了转录组学。这项技术的发展消除了以前用于测量基因表达的基于杂交的微阵列和基于Sanger测序的方法所带来的许多挑战。

RNA测序（RNA sequencing, RNA-Seq）利用NGS方法提供的高通量测序能力深入了解细胞的转录组。与以前基于Sanger测序和微阵列的方法相比，RNA-Seq提供了更高的覆盖范围和更高的分辨率。RNA-seq是通过NGS平台捕获样本动态转录组的快照，从而揭示生物样本中基因表达情况的一种测序技术。除了量化基因表达外，RNA-seq产生的数据还有助于发现新的转录本，鉴定可变剪接基因，检测等位基因特异性表达。RNA-seq的工作流程从测序文库制备开始，文库制备是RNA-seq的关键步骤，它决定了cDNA序列数据对原始RNA群体的反映程度。文库制备流程根据所使用的测序平台的不同而有所差异，一般包括RNA提取、mRNA富集或是核糖体RNA去除、cDNA合成以及测序接头连接等步骤。测序文库构建完成后，通过上机测序和对下机数据进行处理，最终得到FASTQ格式的测序数据。之后，便是对RNA-seq数据展开生物信息学分析。RNA-seq数据分析的所有主要步骤，包括实验设计，质量控制，读取比对，基因和转录水平的量化，可视化，差异基因表达，选择性剪接，功能分析，基因融合检测和eQTL定位等。

FASTQ格式是一种测序数据的存储格式，用于存储生物序列及其相应的质量分数。为了简洁起见，序列字母和质量分数都分别用一个ASCII字符编码。它最初是由Sanger研究所开发，用于表示FASTA格式的序列及其质量数据，但现已成为存储高通量测序仪器输出的标准。

RNA-seq是指利用高通量测序技术对组织或细胞中所有RNA的cDNA文库进行测序，可通过序列reads来定量、分析和发现RNA转录物。这样，转录本就可以被映射到参考基因组上，以获得全面的遗传信息，如转录定位和可变剪接状态。RNA-seq在基因可变剪接、转录后修饰、基因融合、突变/ SNP、基因表达随时间变化或不同处理中基因表达的差异等研究领域都有重要的应用。另外，RNA-seq还可以用于观测其他不同的RNA群体，包括total RNA、小RNA（small RNA，如miRNA）、转运RNA（tRNA）和核糖体RNA（rRNA）。RNA-seq还可用于确定基因外显子/内含子边界，验证或修改先前注释的基因边界。RNA-Seq技术已广泛应用于生物学、医学、临床和制药研究。

对此，我们开发了模拟测序数据生成软件（Mock Reads Generator）。该软件能够根据用户需求选择碱基随机或序列随机的两种程序运行模式，产生一套个性化的模拟转录组测序数据集。通过使用该软件产生的模拟数据集，用户能在进行真实数据的分析之前，预先快速评估出最适配真实测序数据特性的数据分析软件以及相应的最优参数组合，从而能够保证分析研究过程中所用软件和参数的适配性和准确性，同时一定程度上节省了研究者耗费在分析软件选择和参数调试上的时间，从而提高分析效率。

总之，该软件能够于用户预先给定的信息产生模拟测序数据集，帮助研究者快速评估相关转录组学数据分析工具的效果，并调试出最优的参数组合，为下游分析结果的准确性奠定重要基础。

## 1.3 参考文献

1. Chu Y, Corey DR. RNA sequencing: platform selection, experimental design, and data interpretation. Nucleic Acid Therapeutics. 22 (4): 271–4(2012).
2. Wang Z, Gerstein M, Snyder M. RNA-Seq: a revolutionary tool for transcriptomics. Nature Reviews. Genetics. 10 (1): 57–63(2009).
3. Maher CA, Kumar-Sinha C, Cao X, Kalyana-Sundaram S, Han B, Jing X, et al. Transcriptome sequencing to detect gene fusions in cancer. Nature. 458 (7234): 97–101(2009).
4. Ingolia NT, Brar GA, Rouskin S, McGeachy AM, Weissman JS. The ribosome profiling strategy for monitoring translation in vivo by deep sequencing of ribosome-protected mRNA fragments. Nature Protocols. 7 (8): 1534–50(2012).
5. Stark, R., Grzelak, M. & Hadfield, J. RNA sequencing: the teenage years. Nat Rev Genet 20, 631–656 (2019).
6. Grada, Ayman, and Kate Weinbrecht. “Next-generation sequencing: methodology and application.” The Journal of investigative dermatology vol. 133,8: e11(2013).
7. Qi YX, Liu YB, Rong WH. RNA-Seq and its applications: a new technology for transcriptomics. Yi Chuan = Hereditas. 33(11):1191-1202(2011).
8. Kukurba KR, Montgomery SB. RNA Sequencing and Analysis. Cold Spring Harb Protoc. Apr 13;2015(11):951-69(2015).
9. Adams MD, Kelley JM, Gocayne JD, Dubnick M, Polymeropoulos MH, Xiao H, Merril CR, Wu A, Olde B, Moreno RF, et al. Complementary DNA sequencing: Expressed sequence tags and human genome project. Science. 252:1651–1656(1991).
10. Altshuler DM, Gibbs RA, Peltonen L, Dermitzakis E, Schaffner SF, Yu FL, Bonnen PE, de Bakker PIW, Deloukas P, Gabriel SB, et al. Integrating common and rare genetic variation in diverse human populations. Nature. 467:52–58(2010).
11. An J, Lai J, Lehman ML, Nelson CC. miRDeep\*: An integrated application tool for miRNA identification from RNA sequencing data. Nucleic Acids Res. 41:727–737(2013).
12. Anders S, McCarthy DJ, Chen Y, Okoniewski M, Smyth GK, Huber W, Robinson MD. Count-based differential expression analysis of RNA sequencing data using R and bioconductor. Nature protocols. 8:1765–1786(2013).
13. Auer PL, Doerge RW. Statistical design and analysis of RNA sequencing data. Genetics. 185:405–416(2010).
14. Mortazavi A, Williams BA, McCue K, Schaeffer L, Wold B. Mapping and quantifying mammalian transcriptomes by RNA-Seq. Nat Methods. 5:1–8(2008).
15. Levin JZ, Yassour M, Adiconis X, Nusbaum C, Thompson DA, Friedman N, et al. Comprehensive comparative analysis of strand-specific RNA sequencing methods. Nat Methods. 7:709–15(2010).
16. Katz Y, Wang ET, Airoldi EM, Burge CB. Analysis and design of RNA sequencing experiments for identifying isoform regulation. Nat Methods. 7:1009–15(2010).

# 2. 总体设计

## 2.1 目标

以用户输入的参考序列为模板，选择碱基随机或参考序列随机的模式，生成一套覆盖指定序列区间范围的模拟转录组测序数据集，用于序列比对软件的效果评估或转录组测序数据相关的生物信息学分析研究。

## 2.2 功能

1. **生成碱基随机的模拟测序数据集**

对于用户给定的一条参考序列，生成一组覆盖用户指定的序列区间的模拟reads。该序列区间的碱基类型由随机生成，可额外添加不包含随机碱基序列的阴性对照模拟reads。

1. **生成****序列随机的模拟测序数据集**

对于用户给定的一组参考序列集，生成一组覆盖用户指定的序列区间的模拟reads。这些reads随机来源于参考序列集中的其中一条。每个相同的序列位点可以设置任意数目的模拟reads。

## 2.3 使用方法

使用该程序的时候，用户可根据需求选择碱基随机或者序列随机的程序运行模式，并准备好相应的输入参考序列文件，配置相关参数写入到程序运行配置文件中。

1. **碱基随机模式的输入文件**

当选择碱基随机模式时，唯一的输入文件是一条参考序列，作为模板用于下游程序生成模拟reads。该文件只包含一行，为参考序列的碱基字符串，碱基字符的大小写不受限制。此外，需要碱基随机生成的指定序列区域需以N字符进行标识。例如图1展示的是示例程序中的参考序列，该参考序列总共包含326个碱基，其中红色横线标识的需要碱基随机生成的序列区间，共包含8个碱基，通过“NNNNNNNN”进行了标识。

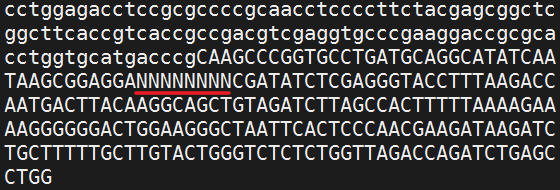


图 1 碱基随机模式中的参考序列文件

1. **序列随机模式的输入文件**

当选择碱基随机模式时，唯一的输入文件是一组参考序列集，作为模板用于下游生成模拟reads。该输入文件的格式是xlsx格式。文件由两列组成，列之间需要用制表符进行分隔。其中第一列为参考序列的名称，需以“>”开头，后接任意字符。第二列为参考序列的序列碱基字符串。首行为列名，必须分别命名为“>Seqname”和“Seq”。示例程序中的参考序列集文件共包含39条序列，图2展示了其中前5条的参考序列，例如第一条序列的名称为“>TATCCAGT”，该参考序列共包含298个碱基，序列以“AGA”开头，以“gtg”结尾。



图 2 序列随机模式中的参考序列文件

1. **程序运行配置文件**

上述输入文件需配置到程序运行配置文件中。配置文件包括各程序步骤用到的软件、输入参数以及输入文件。具体示例见4.2节。

该程序的总体**运行流程**如下图3所示：

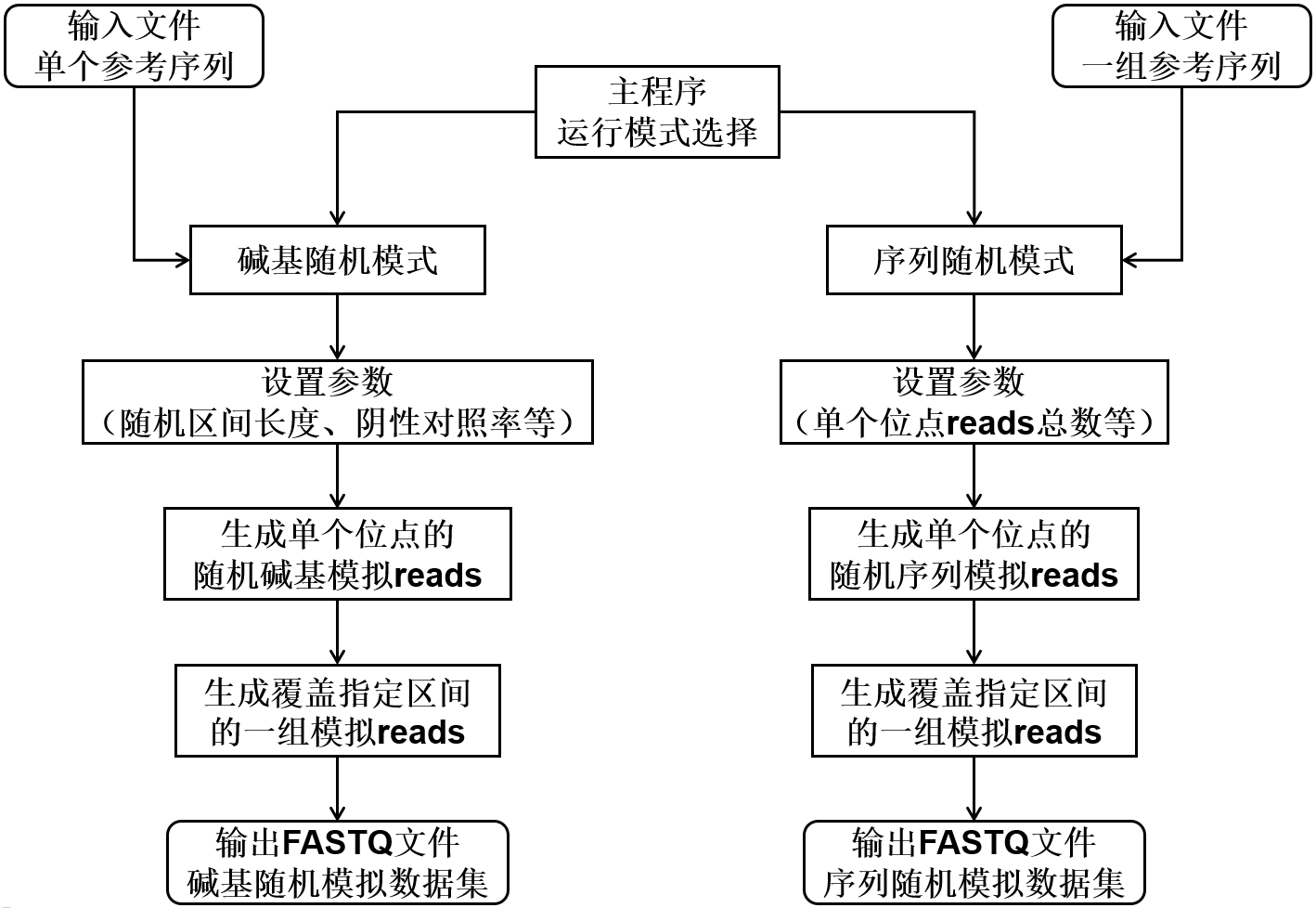


图 3 模拟测序数据生成软件的程序流程图

# 3. 运行环境

## 3.1 硬件

开发环境：

CPU：Intel（R）Core（TM）i5-7300HQ CPU@2.50GHz 2.50 GHz

RAM：8.00GB

内存：128G固态硬盘

## 3.2 支持软件

Miniconda3 Linux 64-bit

R v4.2.2

optparse v1.7.1

openxlsx v4.2.5.2

stringr v1.5.0

dplyr v1.1.2

# 4. 使用说明

## 4.1 软件安装和初始化

**4.1.1 安装Miniconda3**



图 4 Miniconda3安装步骤

**4.1.2 通过Miniconda3安装相关软件**

1. **激活conda**。激活后命令行左侧会出现base字样，代表已激活conda环境



图 5 conda环境激活

1. **安装分析软件**。conda后接：软件名=版本号，通过这种方式能便捷安装3.2节的所有软件。下图示例的是conda安装R v4.2.2。



图 6 conda安装软件

1. **检查软件安装。**安装完毕后，可通过which加软件名找到和调取软件安装的绝对路径

## 4.2 输入文件

当使用程序的碱基随机模式时，唯一的输入文件是一条参考序列；当使用程序的序列随机模式时，唯一的输入文件是一组参考序列集合。输入文件的格式要求在2.3节已做说明，在此只列出要点：

1. **碱基随机模式的输入文件：**只有一行的普通文本文件，包含参考序列的碱基字符串，不包含序列名称，碱基随机生成的序列区间用N字符标识。
2. **序列随机模式的输入文件：**文件格式为xlsx格式。包含两列信息，第一列为“>”开头的参考序列名称，第二列为参考序列碱基字符串，列之间以制表符分隔。首行为列名，必须命名分别命名为“>Seqname”和“Seq”。

Demo示例程序的输入文件存放在软件主目录下的Demo/01-input目录下，其中“BaseRandom”子目录存放的是碱基随机模式的示例输入文件；“TemplateRandom”子目录存放的是序列随机模式的示例输入文件。文件目录结构如下图7所示。

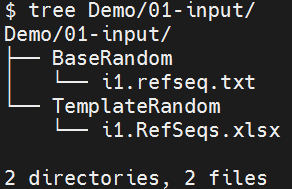


图 7 Demo示例的输入文件目录结构

## 4.3 配置程序

软件安装完成并准备好输入文件后，需要配置程序运行的相关参数。参数配置写入demo\_run.sh文件中，通过执行demo\_run.sh运行主程序。

1. **demo\_run.sh：**

文件内容示例如图8。其中以“#”开头的行是注释行，以“--”开头的行是参数行。每一个参数行对应一个参数变量，按照空格符进行分隔，“--”后指定的是主程序识别的变量名称，根据需求设置；空格符后面为变量所对应的值（参数、文件路径等）。

1. **碱基随机模式**：使用“--Mode BaseRandom”指定程序运行模式为碱基随机模式。在此模式下，“--Templates”指定生成模拟reads的单个参考序列文件路径，“--SiteNums”指定单个位点的总reads数，“--ReadLength”指定生成模拟reads的总长度，“--NegativeRate”指定单个位点不含随机碱基的阴性对照reads比例，“--BarcodeLens”指定随机碱基序列区间的长度，需与参考序列中的N数目保持一致，“--Outdir”指定输出文件目录。如不需要使用该模式，可将图8中“碱基随机模式”红框区域的代码进行删除。
2. **序列随机模式**：使用“--Mode TemplateRandom”指定程序运行模式为序列随机模式。在此模式下，“--Templates”指定生成模拟reads的一组参考序列文件路径，“--SiteNums”指定单个位点的总reads数，“--ReadLength”指定生成模拟reads的总长度，“--Outdir”指定输出文件目录。如不需要使用该模式，可将图8中“序列随机模式”红框区域内的代码进行删除。
3. **FastQ结果文件生成**：在配置好步骤1（碱基随机模式或序列随机模式）的参数后，需要设置步骤2的参数将结果转换为FASTQ文件。参数配置如图8中“生成fq文件”红框区域的代码所示，其中输入参数1指定为步骤1生成的结果文件路径，输入参数2指定最终生成的FASTQ文件路径。

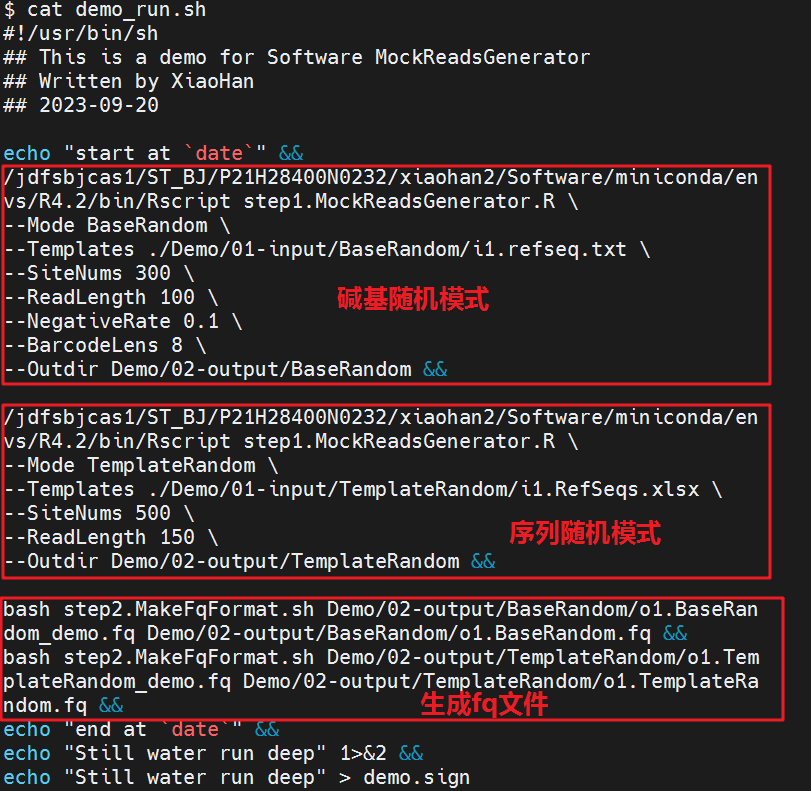


图 8 demo\_run.sh文件配置参数

**注意：在整个过程中最好均使用绝对路径！！！**

## 4.4 输出结果

运行demo\_run.sh后，在指定输出主目录下会产生一个结果目录“02-output”，该结果目录包含两个子目录，其中“BaseRandom”子目录存放的是碱基随机模式的结果文件，“TemplateRandom”子目录存放的是序列随机模式的结果文件，示例如下图9所示。**下面以demo示例脚本为例，解释结果文件。**其程序运行脚本存放在软件主目录下的demo\_run.sh脚本中，该示例脚本的输出结果文件存放在软件主目录下的Demo/02-output子目录中。

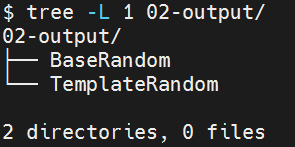


图 9 输出结果目录

**4.4.1 碱基随机模式结果说明**

结果文件存放在BaseRandom子目录下，文件结构如图10所示。箭头所指的两个文件是程序的最终输出文件。其中o1.BaseRandom.fq文件是生成的模拟测序reads数据的FASTQ文件，o2.BaseRandom\_metadata.xlsx文件是模拟测序reads数据的原始信息。o1.BaseRandom\_demo.fq是程序第一步运行产生的结果文件，用于生成最终的FASTQ文件o1.BaseRandom.fq。

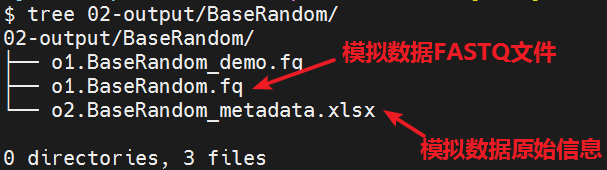


图 10 碱基随机模式结果目录结构

如图11所示，o1.BaseRandom.fq是模拟测序reads的FASTQ文件，其中每4行是一条模拟reads的信息，示例展示了其中前4条模拟reads。以图中红框框出的第一条模拟测序read为例：

1. 以“@”开头的首行记录了该模拟read的序列信息，其中RI（ReadID）记录read编号为“0000001”，BC（Barcode）记录read的随机碱基序列为“CAACCAGT”，SS（StartSite）记录随机碱基序列出现在read上的碱基起始位置为“92”，ES（EndSite）记录随机碱基序列出现在read上的碱基起始位置为“99”；
2. 第二行记录了该read的碱基序列，以不区分大小写的“ATCG”表示碱基类型。
3. 第三行以“+”开始，可以储存一些附加信息，一般为空。
4. 第4行储存的是质量信息，与第2行的碱基序列是一一对应的，其中的每一个符号代表对应碱基的测序质量值。由于本程序生成的是模拟read，因此该行中的碱基质量值统一用“F”标识，“F”表示碱基质量最佳。可根据需要对该行内容进行手动修改。

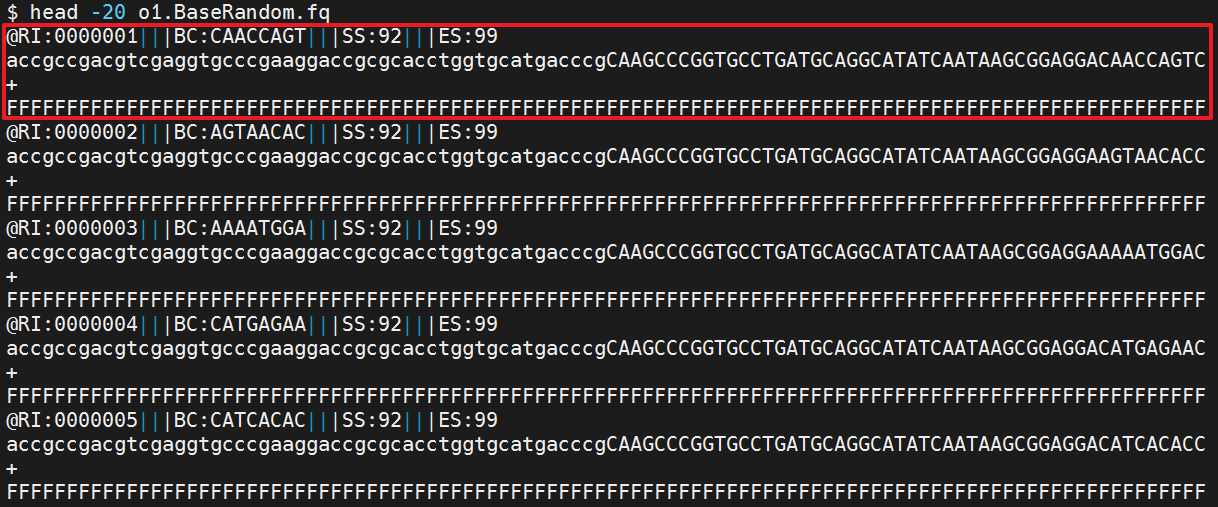


图 11 o1.BaseRandom.fq文件格式

如图12所示，o2.BaseRandom\_metadata.xlsx是xlsx格式的模拟数据原始信息。其中第一列“seqname”记录序列名，第二列“seq”记录碱基序列，第三、四、五列分别记录随机碱基序列Barcode的碱基类型，在read上的起始位点和终止位点，第六列记录该模拟read是否带有随机碱基序列，当该列内容为“WithoutBarcode”时代表该模拟read时不带有Barcode的阴性对照模拟read。

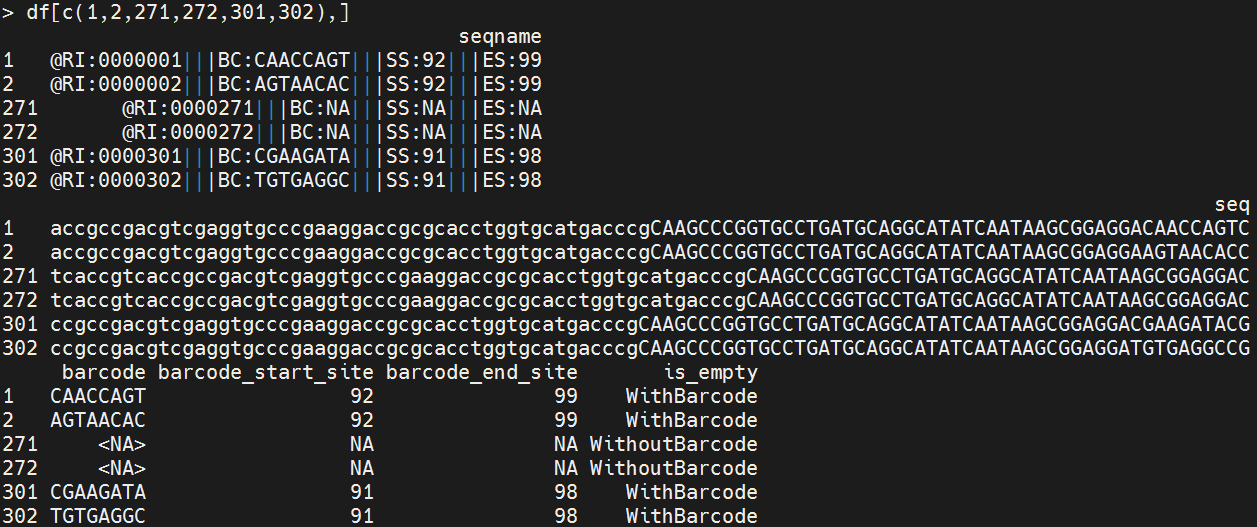


图 12 o2.BaseRandom\_metadata.xlsx文件格式

**4.4.2 序列随机模式结果说明**

结果文件存放在TemplateRandom子目录下，文件结构如图13所示。箭头所指的两个文件是程序的最终输出文件。其中o1.TemplateRandom.fq文件是生成的模拟测序reads数据的FASTQ文件，o2.TemplateRandom\_metadata.xlsx文件是模拟测序reads数据的原始信息。o1.TemplateRandom\_demo.fq是程序第一步运行产生的结果文件，用于生成最终的FASTQ文件o1.TemplateRandom.fq。

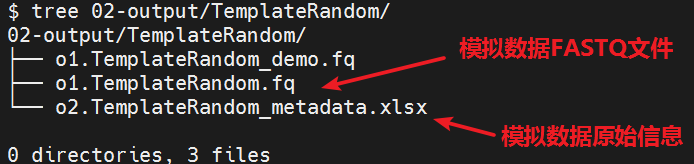


图 13 序列随机模式结果目录结构

如图14所示，o1.TemplateRandom.fq是模拟测序reads的FASTQ文件，格式同4.4.1中的o1.BaseRandom.fq文件格式。稍有所不同是，图中展示的模拟reads长度为150 bp，阴性对照的模拟read在BC标识上标记为“EmptyVector”。

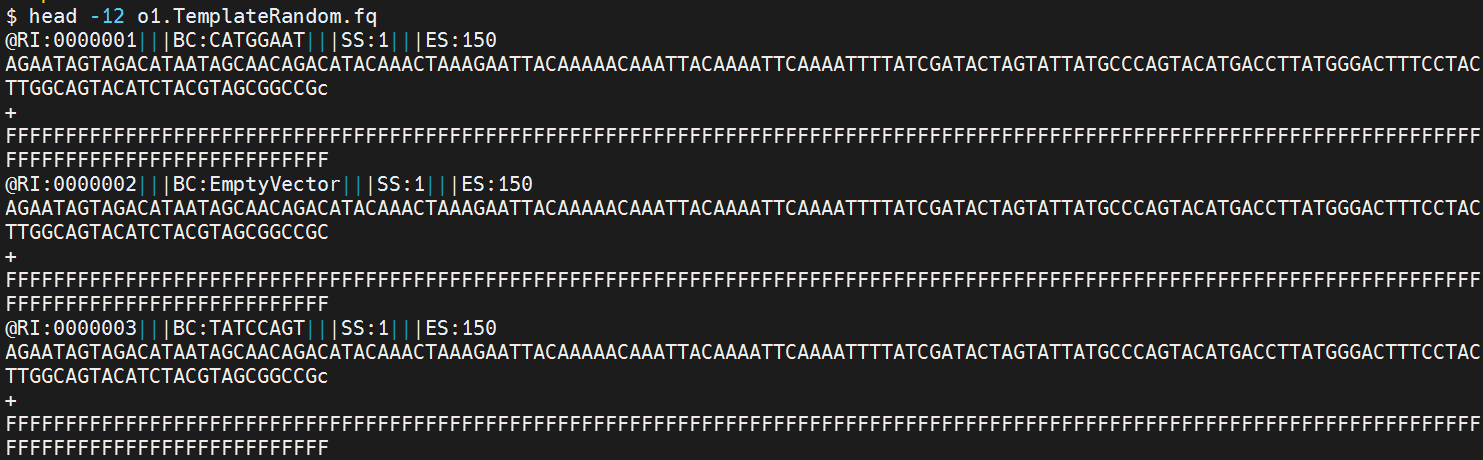


图 14 o1.TemplateRandom.fq文件格式

如图15所示，o2.BaseRandom\_metadata.xlsx是xlsx格式的模拟数据原始信息。其中第一列“seqname”记录序列名，第二列“seq”记录碱基序列，第三列“start\_site和第四列“end\_site”分别记录模拟reads位于参考序列上的碱基起始位点和碱基终止位点，第六列“is\_empty”记录该模拟reads是否为阴性对照模拟reads，当该列内容为“empty”时，代表该条模拟reads是来源于参考序列集中的空载参考序列。



图 15 o2.BaseRandom\_metadata.xlsx文件格式

## 4.5 求助查询

1. 当不清楚主程序输入参数的时候，可以通过输入以下命令查看帮助文档（在脚本后面添加--help参数）：

Rscript step1.MockReadsGenerator.R --help

具体参数说明如下图16所示：

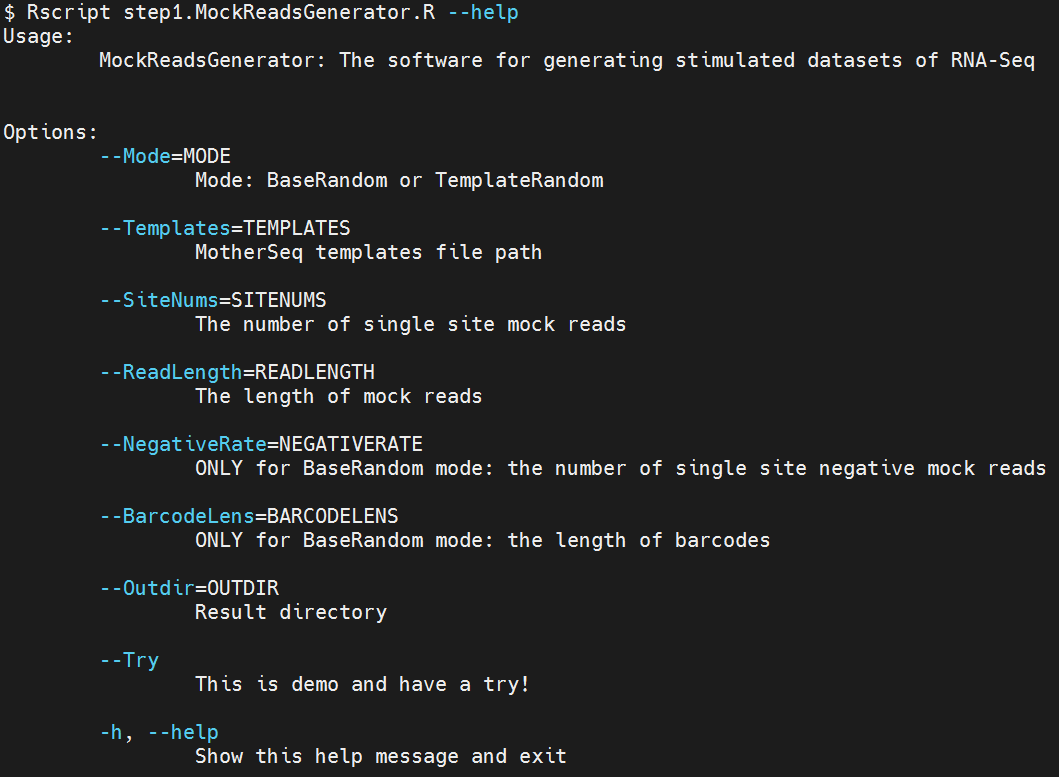


图 16 主程序参数说明

1. 当不清楚程序使用到的软件参数含义的时候，可以通过输入软件的绝对路径加—help参数查看软件说明文档：

例如，查看Miniconda3软件的参数说明：

/jdfsbjcas1/ST\_BJ/P21H28400N0232/xiaohan2/Software/miniconda/bin/conda --help

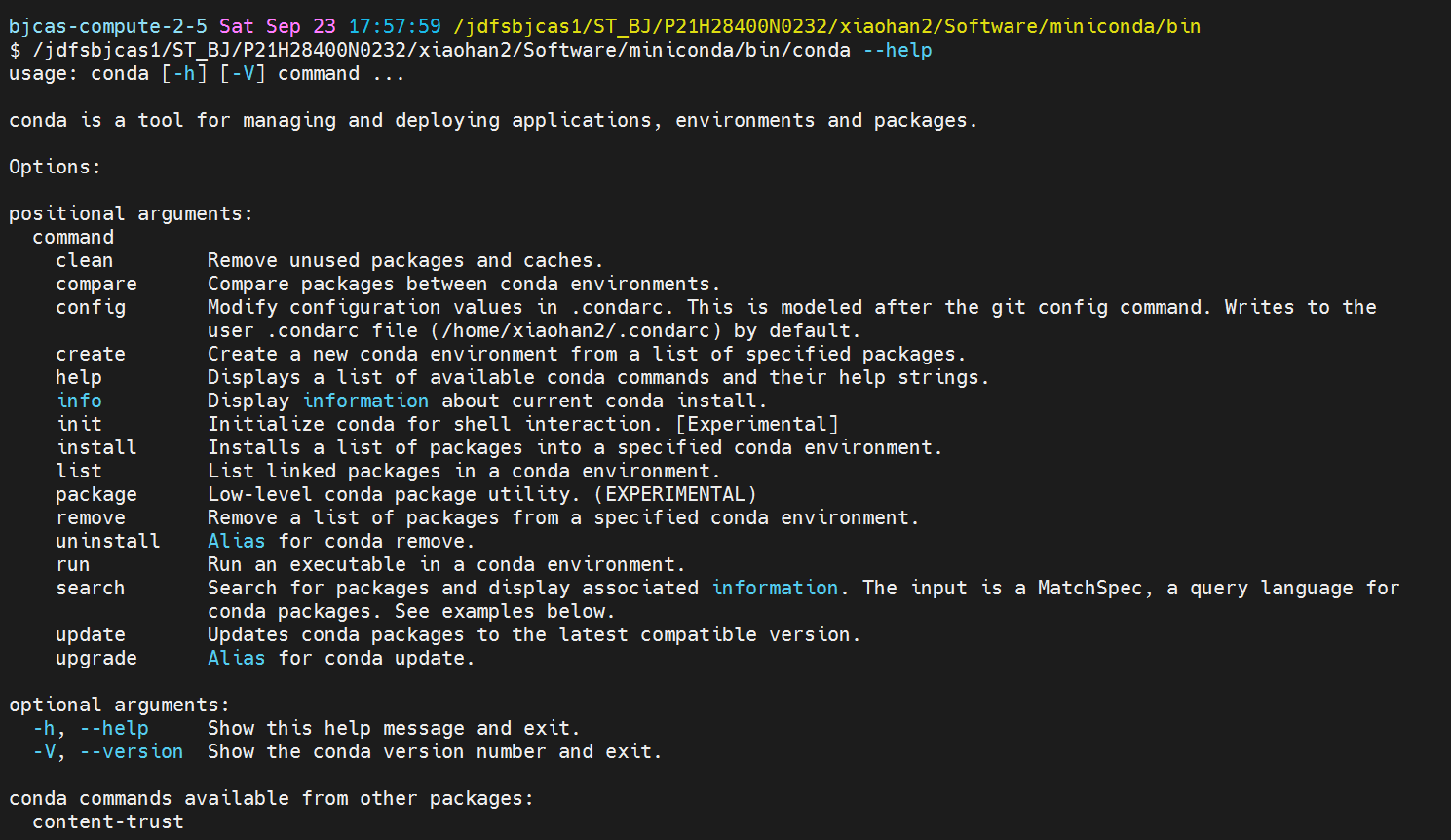


图 17 调取Miniconda3参数说明

# 5. 运行说明

## 5.1 程序存放位置

**脚本存放路径：**

/jdfsbjcas1/ST\_BJ/P21H28400N0232/xiaohan2/Software/MockReadsGenerator目录下的可执行脚本文件。

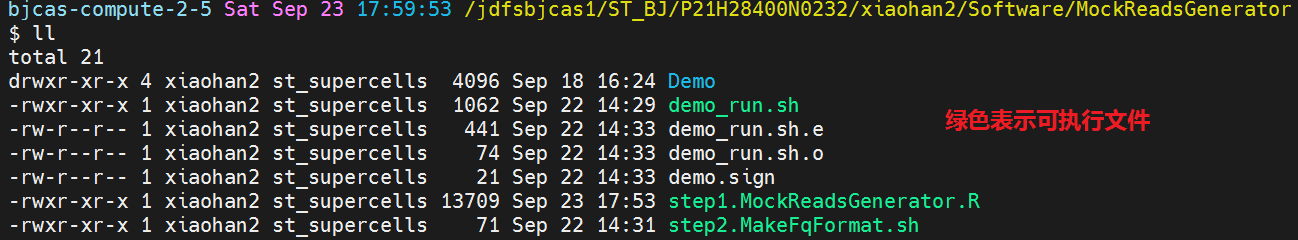


图 18 程序脚本

## 5.2 运行步骤

在此以一个**Demo数据作为示例**，展示使用程序分别运行碱基随机模式和序列随机模式的方法。

1. **准备输入文件，放在Demo/01-input目录下**

**示例输入文件主目录路径：**

/jdfsbjcas1/ST\_BJ/P21H28400N0232/xiaohan2/Software/MockReadsGenerator/Demo/01-input

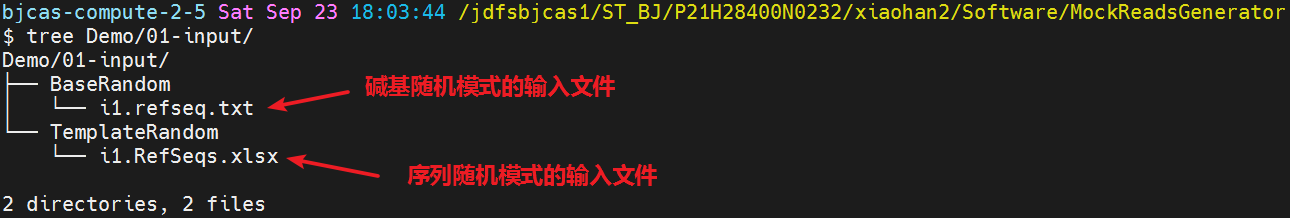


图 19 输入文件存放目录

1. **修改配置demo\_run.sh文件内容，放在程序主目录下**

**示例配置文件路径：**

/jdfsbjcas1/ST\_BJ/P21H28400N0232/xiaohan2/Software/MockReadsGenerator/demo\_run.sh

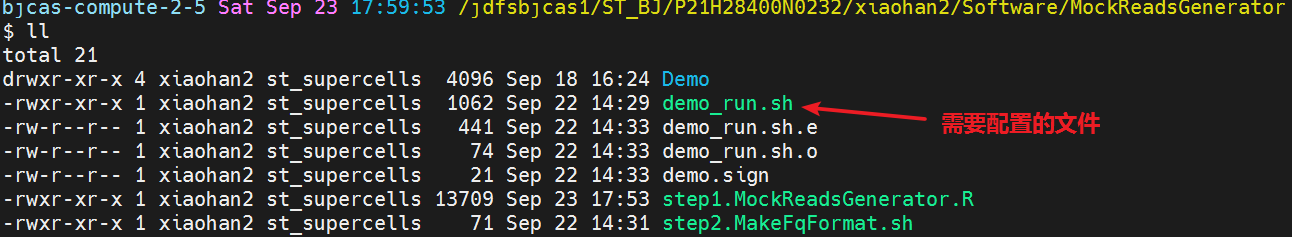


图 20 需要修改的配置文件

1. **运行demo\_run.sh生成对应的输出文件主目录**

**示例输出文件主目录路径**：/jdfsbjcas1/ST\_BJ/P21H28400N0232/xiaohan2/Software/MockReadsGenerator/Demo/02-output

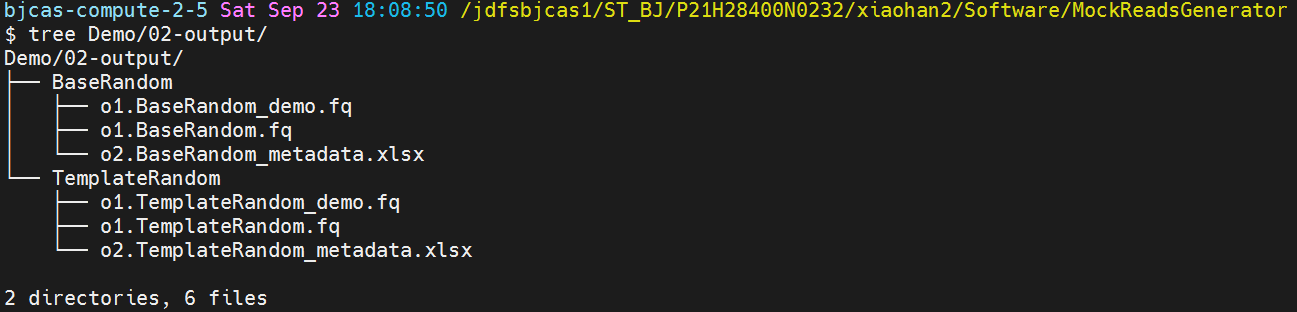


图 21 输出文件存放目录

随后可通过查看程序主目录中的如下三个文件查看运行结果，其中：

demo\_run.sh.o 该文件中保存了一些输出到屏幕的信息，例如运行的起始和终止时间；

demo\_run.sh.e 该文件中保存报错信息，如果该程序报错，则会将报错信息输入到该文件中，最后可以通过查询该文件进行问题溯源，从而解决问题。

demo.sign 当程序正确且成功运行后，会生成demo.sign文件，以提示用户程序已经成功运行完成，用户不需要通过其他方式对程序进行监控。

## 5.3 运行时间

可以通过cat demo\_run.sh.o查看文件，其中第一行是起始时间，最后一行是程序运行完后的时间。结果如下图所示：

以Demo示例为例，该示例在碱基随机模式中生成了27300条长度为100 bp的模拟reads和在序列随机模式中生成了74500条长度为150 bp的模拟reads，程序用时为12秒。程序运行耗时主要在第一步运行。若用户需要生成的模拟reads的数目更多，长度更长，则总时长取决于服务器运算速度和需求数据总量的大小。

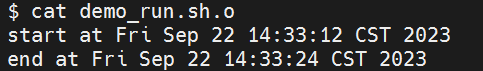


图 22 demo\_run.sh.o示例图

## 5.4 运行错误说明

当运行程序时，会生成两个文件demo\_run.sh.o和demo\_run.sh.e，其中错误输出会保存在文件demo\_run.sh.e中。也就是说，当程序出错的时候会将报错的信息保存到demo\_run.sh.e中，我们可以通过查看demo\_run.sh.e文件来检查程序是否出错，如果出错也可以从该文件中得到报错信息，进行问题溯源，从而修改输入文件或者程序来实现成功运行。

可以通过键入cat demo\_run.sh.e查看文件，结果如下图所示：

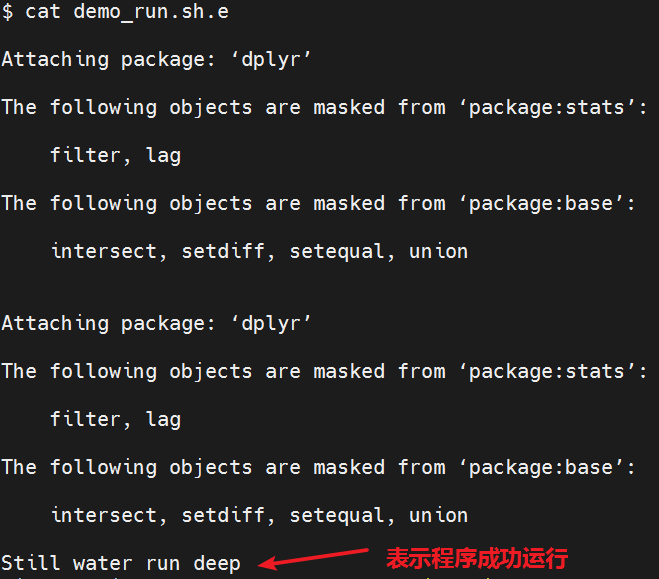


图 23 demo\_run.sh.e示例图

其中如果出现“Attaching package”是将R脚本调用包产生的，可忽略。

当该文件中出现“Still\_waters\_run\_deep”则表示该程序运行成功且运行完成，反之如果出现了其他字样可能是报错信息，需要仔细阅读，找到错误原因并修正。修正后可以重新运行。

## 5.5 运行成功说明

当程序成功完成后会生成另一个新的文件demo.sign。该文件的生成表明程序的顺利执行，并且已经完成所有运行。用户可直接查看结果内容。无需通过其他操作对程序进行监控。

可以通过键入cat demo.sign查看文件，结果如下图所示：



图 24 demo.sign示例图

该文件中只有一行内容“Still\_waters\_run\_deep”，表示程序运行完成并且成功运行。