Analysis of RNA-seq Data

Introduction:

In this tutorial we are going to show you how to perform the analysis of RNA-seq data. We will show you how to perform the quality control of the raw data, align against the reference genome, count the alignment reads, identify differentially expressed genes, and conduct the function enrichment.

The data:

For this tutorial we use an example data, a part of RNA-seq double-end sequencing data of mouse cells at different development time points.

This experiment was performed in two time points, time0 and time1 were included in duplicates.

The raw data for this study is available at: /lustre/home/acct-lctest/stu411/RNA-seq/raw_data.

Prepare data:

下载软件

FastQC v0.11.9

BBMap version 38.86

hisat2-2.0.0-beta

samtools-1.10

subread-2.0.1-Linux-x86_64

并安装到路径：

/lustre/home/acct-lctest/stu411/RNA-seq/softwares

将软件的执行命令加入到环境变量：

vim .bashrc

PATH=$PATH:/lustre/home/acct-lctest/stu411/RNA-seq/softwares/FastQC

PATH=$PATH:/lustre/home/acct-lctest/stu411/RNA-seq/softwares/bbmap

PATH=$PATH:/lustre/home/acct-lctest/stu411/RNA-seq/softwares/hisat2-2.0.0-beta

PATH=$PATH:/lustre/home/acct-lctest/stu411/RNA-seq/softwares/samtools-1.10

PATH=$PATH:/lustre/home/acct-lctest/stu411/RNA-seq/softwares/subread-2.0.1-Linux-x86_64/bin

重新执行刚修改的初始化文件,使之生效

source .bashrc

准备小鼠基因组的 index 文件：

下载地址：ftp://ftp.ccb.jhu.edu/pub/infphilo/hisat2/data/mm10.tar.gz，下载后解压

最终路径：/lustre/home/acct-lctest/stu411/RNA-seq/index/mm10/genome

准备小鼠基因组的注释文件：

下载地址：
ftp://ftp.ebi.ac.uk/pub/databases/gencode/Gencode_mouse/release_M25/gencode.vM25.annotation.gtf.gz,下载后解压

最终路径：/lustre/home/acct-lctest/stu411/RNA-seq/annotation/gencode.vM25.annotation.gtf

Analysis data:

解压成 fastq 格式

gunzip fastqc time0_rep1_R1.fastq.gz

gunzip fastqc time0_rep1_R2.fastq.gz

第一步：质量控制

在进行数据分析之前，需要用相关的软件对测序的原始数据进行质量控制，如果质量不合格，我们需要处理后，才能进行后续分析。如果质量合格，我们可以直接继续分析。常用质量控制的软件为 FastQC.

FastQC 介绍:

FastQC 是一款基于 Java 的软件，它可以快速地对测序数据进行质量评估，其官网为：http://www.bioinformatics.babraham.ac.uk/projects/fastqc/

Command:

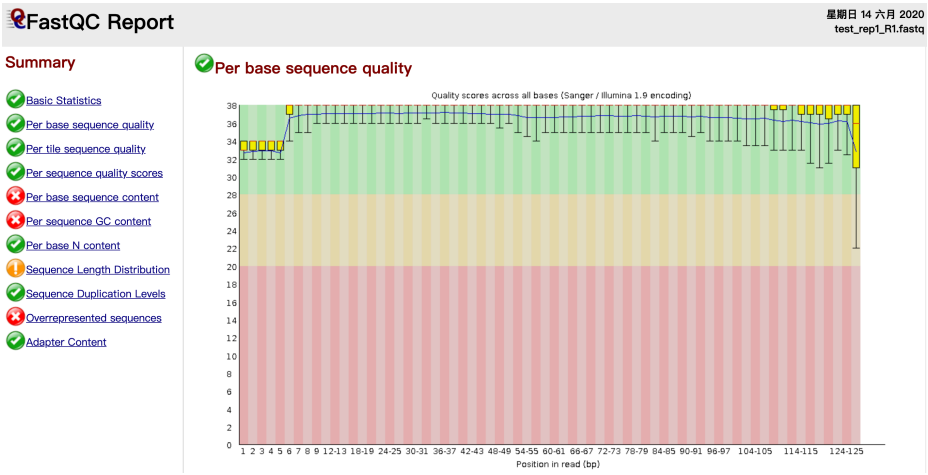
fastqc time0_rep1_R1.fastq

fastqc time0_rep1_R2.fastq

运行结束后生成两个文件一个.html 网页文件，一个是.zip 压缩文件。

```
time0_rep1_R1_fastqc.zip
time0_rep1_R1_fastqc.html
time0_rep1_R2_fastqc.zip
time0_rep1_R2_fastqc.html
```

我们使用浏览器打开其中一个 html 文件，显示如下：



上图中 Summary 的部分就是整个报告的目录，整个报告分成若干个部分。合格会有个绿色的对勾，警告是个黄色的叹号，不合格是个红色的叉子。

通常我们主要关注下面几项信息：

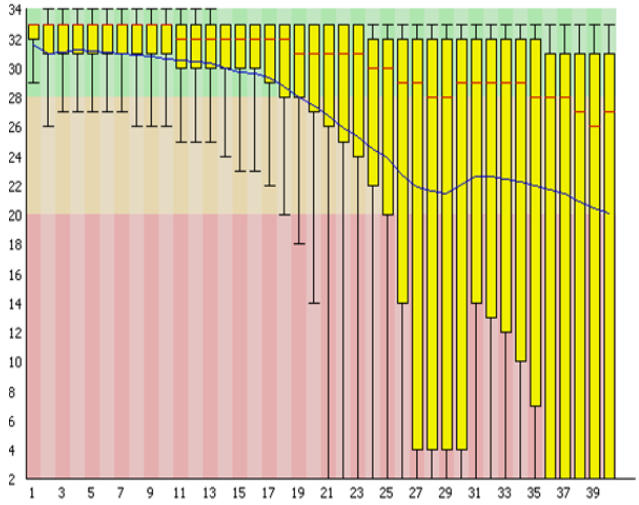
Basic Statistics 展示基本信息

Per base sequence quality 如果显示不合格，需要处理

Adapter Content 如果显示不合格，需要处理

在这个数据中显示合格，我们继续往下进行。

假如 FastQC 结果的 Per base sequence quality 和 Adapter Content 不合格：



数据需要进行下面处理：

需要用到 BBMAP 中的 BBDuk 工具。

BBMap: Short read aligner for DNA and RNA-seq data. Capable of handling arbitrarily large genomes with millions of scaffolds. Handles Illumina, PacBio, 454, and other reads; very high sensitivity and tolerant of errors and numerous large indels.

BBDuk: Filters, trims, or masks reads with kmer matches to an artifact/contaminant file.

For Adapters:

bbduk.sh –Xmx1g in=read1.fq in=read2.fq out1=clean1.fq out2=clean2.fq ref=adapters.fq ktrim=r k=23 mink=11 hdisk=1 tpe tbo

For low quality reads:

bbduk.sh –Xmx1g in=read1.fq in=read2.fq out1=clean1.fq out2=clean2.fq qtrim=r trimq=20

BBDuk 处理后的 clean data 需要进一步用 fastqc 进行质控，看是否符合要求。如果符合要求，则进行下一步操作。

第二步：reads 比对

如果只需要找差异表达基因，我们可以用 bowtie2, bwa 这类比对工具，但如果需要找到新的 isoform，或者 RNA 的可变剪切，就需要 TopHat, HISAT2 或者是 STAR 等工具。有研究比较 TopHat,HISAT2 和 STAR 等工具,结论是 HISAT2 找到 junction 正确率最高，但在总数上却比 TopHat 和 STAR 少。就速度比较，HISAT2 比 STAR 和 TopHat2 平均快上 2.5~100 倍。这里我们用 HISAT2 进行演示。

HISAT2 介绍:

高通量测序中我们需要成千上万甚至上几亿条 reads 在合理的时间内比对到参考基因组上，并且保证错误率在接受范围内。为了提高比对速度，HISAT2 根据参考基因组序列，经过 BWT 算法转换成 index，所以我们比对的序列其实是 index 的一个子集。因此不是直接把 read 比对到基因组上，而是把 read 和 index 进行比较。对于人类和小鼠的 index 有现成的，HISAT2 官网可以直接下载进行序列比对。http://ccb.jhu.edu/software/hisat2/manual.shtml

Command:

$hisat2 –x <hisat2-idx> {-1 <m1> -2 <m2> } or –U <r> -S <hit>

Parameters:

–x <hisat2-idx> : The basename of the index for the reference genome

{-1 <m1> -2 <m2> } : for pair-end reads

-U <r>: for single reads

-S: sam format output

在这个教程中下载的 index 位于 /lustre/home/acct-lctest/stu411/RNA-seq/index/mm10/genome

运行命令

hisat2 -x /lustre/home/acct-lctest/stu411/RNA-seq/index/mm10/genome -1 time0_rep1_R1.fastq -2 time0_rep1_R2.fastq -S time0_rep1_align.sam

结果文件：time0_rep1_align.sam

SAM 文件说明：

SAM 是一种序列比对格式标准，由 sanger 制定，是以 TAB 为分割符的文本格式。主要应用于测序序列 mapping 到基因组上的结果表示，当然也可以表示任意的多重比对结果。

SAM 分为两部分，注释信息（header section）和比对结果部分（alignment section）。行：除注释外，每一行是一个 read。注释信息可有可无，都是以@开头，用不同的 tag 表示不同的信息，主要有：

1.@HD，说明符合标准的版本、对比序列的排列顺序；

2.@SQ，参考序列说明；

3.@RG，比对上的序列（read）说明；

4.@PG，使用的程序说明；

5.@CO，任意的说明信息。

比对结果部分（alignment section），每一行表示一个片段（segment）的比对信息，包括 11 个必须的字段（mandatory fields）和一个可选的字段，字段之间用 tag 分割。必须的字段有 11 个，顺序固定，不可用时，根据字段定义，可以为'0'或者'*'，这是 11 个字段包括：

1. QNAME 比对片段的（template）的编号；read name，read 的名字通常包括测序平台等信息

eg.ILLUMINA-379DBF:1:1:3445:946#0/1

2. FLAG 位标识，template mapping 情况的数字表示，每一个数字代表一种比对情况，这里的值是符合情况的数字相加总和；flag 取值见备注(3)

eg.16

3. RNAME 参考序列的编号，如果注释中对 SQ-SN 进行了定义，这里必须和其保持一致，另外对于没有 mapping 上的序列，这里是'*'；

eg.chr1

4. POS 比对上的位置，注意是从 1 开始计数，没有比对上，此处为 0;

eg.36576599

5. MAPQ mappint 的质量,，比对的质量分数，越高说明该 read 比对到参考基因组上的位置越唯一;

eg.42

6. CIGAR　　简要比对信息表达式（Compact Idiosyncratic Gapped Alignment Report)，其以参考序列为基础，使用数字加字母表示比对结果，match/mismatch、insertion、deletion 对应字母 M、I、D。比如 3S6M1P1I4M，前三个碱基被剪切去除了，然后 6 个比对上了，然后打开了一个缺口，有一个碱基插入，最后是 4 个比对上了，是按照顺序的；

eg.36M 表示 36 个碱基在比对时完全匹配

###注：第七列到第九列是 mate(备注 1)的信息，若是单末端测序这几列均无意义 ###

7. RNEXT　　下一个片段（即 mate）比对上的参考序列的编号，没有另外的片段，这里是'*'，同一个片段，用'='；

eg.*

8. PNEXT　　下一个片段（即 mate）比对到参考序列上的第一个碱基位置，若无 mate,则为 0；

eg.0

9. TLEN　　Template 的长度，最左边得为正，最右边的为负，中间的不用定义正负，不分区段（single-segment)的比对上，或者不可用时，此处为 0(ISIZE, Inferred fragment size.详见 Illumina 中 paired end sequencing 和 mate pair sequencing，是负数，推测应该是两条 read 之间的间隔(待查证)，若无 mate 则为 0);

eg.0

10. SEQ　　序列片段的序列信息，如果不存储此类信息，此处为'*'，注意 CIGAR 中 M/I/S/=/X 对应数字的和要等于序列长度；

eg.CGTTTCTGTGGGTGATGGGCCTGAGGGGCGTTCTCN

11. QUAL　　序列的质量信息,read 质量的 ASCII 编码。，格式同 FASTQ 一样。

eg.PY[[YY_____QQQQbILKIGEFGKB

12.第十二列之后：Optional fields，以 tab 建分割。详见备注(2)

eg.AS:i:-1 XN:i:0 XM:i:1 XO:i:0 XG:i:0 NM:i:1 MD:Z:35T0 YT:Z:UU

可选字段（optional fields)，格式如：TAG:TYPE:VALUE，其中 TAG 有两个大写字母组成，每个 TAG 代表一类信息，每一行一个 TAG 只能出现一次，TYPE 表示 TAG 对应值的类型，可以是字符串、整数、字节、数组等。

```
[stu411@login1 process_data]$ head -n 20 time0_rep1_align.sam
@HD	VN:1.0	SO:unsorted
@SQ	SN:chr1	LN:195471971
@SQ	SN:chr10	LN:130694993
@SQ	SN:chr11	LN:122082543
@SQ	SN:chr12	LN:120129022
@SQ	SN:chr13	LN:120421639
@SQ	SN:chr14	LN:124902244
@SQ	SN:chr15	LN:104043685
@SQ	SN:chr16	LN:98207768
@SQ	SN:chr17	LN:94987271
@SQ	SN:chr18	LN:90702639
@SQ	SN:chr19	LN:61431566
```

```
[stu411@login1 process_data]$ tail -n 20 time0_rep1_align.sam
SRR1840514.17816	83	chr13	95336779	255	123M	=	95336669	-233	AAACCTAACCATAGGCTGCAGGCGCGGCAGCGCCCCCAAG
TCCAGTAGCCTGGATTAGGTACACAGACAGGATAGTCCTCACAGCTACCTCTTTGTAGTCCTGAAGGCATTTTAAGCCAGAAG	BBGEGGB@BB=GGGEGGGGGGGGGGGGGGGGGC0CGG>DDGGGGGGG>CGGGGGG
GGGGGGGCGGGGGGEGGGGGGGGGGGGGGGGGGGCCEGGGGGGGEGGGGGGGGGFGGEGFGBBABB	AS:i:0	XN:i:0	XM:i:0	XO:i:0	XG:i:0	NM:i:0	MD:Z:123	YS:i:0 Y
T:Z:CP	NH:i:1
SRR1840514.17816	163	chr13	95336669	255	126M	=	95336779	233	CTCTCCCCTGCCTCTGCTCCCCAAGGAGGACCCTGACATC
CATGACAACAACAGTGACAATGATATGGTCACCAAGGTCCTGCTGTAGCATGACTGACCGCTCCATGTGTAAACCTAACCATAGGC	=@BBBEGGGG0;EDDGGG1EFGGGGGGFDGGEDFEGGGGFGGGECFGGGGGGGGG
GGG@FGEGCGCGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGEGGGGGGGGGGGGGFEFGGC=	AS:i:0	XN:i:0	XM:i:0	XO:i:0	XG:i:0	NM:i:0	MD:Z:126	YS:i:0 Y
T:Z:CP	NH:i:1
SRR1840514.17817	83	chr1	58417363	255	21M2283N67M387N38M	=	58417326	-163	ACTGCTACACGTCTACCTCCCACT
TTATGATCGATGCATTCCACCACTTTTCCGAAAGCACCTTCACCTAAAGTATCAACAATTTCATATCTTGCACTTAGTACGTCTCCACTCTGACAGATCAGG	:GGGGGGGGGFF>F:GGGGGGGGGGGGGGGGGGGGGGGGG
GGGGGGGGGGGCGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGCBCCC	AS:i:0	XN:i:0	XM:i:0	XO:i:0	XG:i:0	NM:i:0	MD:Z:12Y
S:i:0	YT:Z:CP XS:A:-	NH:i:1
SRR1840514.17817	163	chr1	58417326	255	58M2283N67M	=	58417363	163	AGCTTCACAGTATCTATCCACATTTTTAACTA
TTTTTACTGCTACACGTCTACCTCCCACTTTATGATCGATGCATTCCACCACTTTTCCGAAAGCACCTTCACCTAAAGTATCAACAATTTCAT	BCBBBGGEGGGGGGGGGGGGGGGGFGGGGGGGGEGGGGGGGGGGGGG
GGGGGGGGGGGGG>FGGGGGGFGGGGGEGGGGGGGGGGGGGGFGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG	AS:i:0	XN:i:0	XM:i:0	XO:i:0	XG:i:0	NM:i:0	MD:Z:125	Y
S:i:0	YT:Z:CP XS:A:-	NH:i:1
```

BAM 格式介绍：

BAM 和 SAM 文件内容其实是一样的,是 SAM 的二进制版，在 SAM 的基础上运用二进制编码，极大的压缩了 SAM 文件的体积,在后续分析中经常用到 BAM 文件，因此，需要专门工具进行格式转换，samtools 可以实现 SAM/BAM 的相互转换，并且可以查看 BAM 文件。

格式转换（SAM 转 BAM）

samtools view -bS time0_rep1_align.sam > time0_rep1_unsorted.bam

samtools sort time0_rep1_unsorted.bam -o time0_rep1_sorted.bam

查看 BAM 文件

samtools view time0_rep1_sorted.bam | head

```
[stu411@login1 process_data]$ samtools view time0_rep1_sorted.bam | head
SRR1840514.636	99	chr1	3613685 255	126M	=	3613864 305	CTATATCTGGAACACAGCCACTGTGCTCTCAGAAAACACTTCCCAGAAGATGTCACCTCAATGA
TGCTGGTCTCTTCTTAATCACCGCTAATTTCTTAGCTCCAGCTAACCAGCATCAATACTCCC	BCCCCGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGEGEGGGGGGGGGGEEGGGFGGGGGGGGGGGGGGGFGGCGGGGGG
GGGGGFGGGGDGGGGGGEGGGGGGGFGGGGGGGGGGGGGGGGGEFG	AS:i:0	XN:i:0	XM:i:0	XO:i:0	XG:i:0	NM:i:0	MD:Z:126	YS:i:0	YT:Z:CP NH:i:1
SRR1840514.636	147	chr1	3613864 255	126M	=	3613685 -305	ATTCTTCAGCCCCAGCTAACCAGAACTACAGAATCTTCACAATCAAAACAGCAATGGCCCTGAA
AAGAGTCTTTAATTTTCCCTCTGAAATTTCACAAGCCAGGCCTCCATCTTCTGCACTGTTCT	AF=GF=,AF:,GCF>FGGEGCGGEGGGGGGGCGGCGGEGGGG;G@F0@GGGGGGGGGGGGGGG@0EGGGGGGGGFGGEGB
GGGGGGGGGGCD>GGGGGFGGGGGGGFEGGGFFFFEGGGGGABBBB	AS:i:0	XN:i:0	XM:i:0	XO:i:0	XG:i:0	NM:i:0	MD:Z:126	YS:i:0	YT:Z:CP NH:i:1
SRR1840514.25905	177	chr1	3820991 255	17S19M	chr17	12913799	0	GAGCTGGTAACGATGGATAGTAGAGAAAAATAGTTT	F>01>1/E
0>/11@111>;11111>1@11?100:33	AS:i:-17	XN:i:0	XM:i:0	XO:i:0	XG:i:0	NM:i:0	MD:Z:19 YS:i:0	YT:Z:DP NH:i:1
SRR1840514.5148 99	chr1	3984175 255	110M875N8M	=	3985172 1122	CAAGGTTTTCTGCGTGGAGACGTCTAATTTAGCAGGTCTTTTTTTTTCTTGAGTAG
GCATTCTGGAGCTGCTTTTTCGCTCGCTGTCTCACTCACTGGAGCACCTTGCGCCTTTAACC	BBCA?1EFGD@1C>C/EGGFGBBBGGGGCFG1FDGGGGGGGGGGGG<CG@1:<B1FDDGGGGG><C1G0FC0FGEFGC/8
C/;0CFDEG00;000:8<:0006BF=0..89CC@CGGB	AS:i:-3 XN:i:0	XM:i:1	XO:i:0	XG:i:0	NM:i:1	MD:Z:107A10	YS:i:0	YT:Z:CP XS:A:-	NH:i:1
SRR1840514.5148 147	chr1	3985172 255	125M	=	3984175 -1122	GGCTACTGCCATCAGTGTACCAGCTCTTCTGCCAGAATGTCAGCGAAGTGATGACTATGTGCAG
TCTCACACAGTGGGACAGCAGGTGGAATGAAGCAAAGGTCACTCGTTCAGTCAACGAAAAG	F@@=F,>EGGGFGGCGGGGGF?G?68:;GGCGFF?CF0B/BGF;GCFCEDE0B:0GF=CGEFDF:00EG@C@F=01C0FC
DDF=@G>BCGGGGDD@BCC<11:0=E/FDF=;;B@GBF;FB?3BB	AS:i:0	XN:i:0	XM:i:0	XO:i:0	XG:i:0	NM:i:0	MD:Z:125	YS:i:-3 YT:Z:CP NH:i:1
SRR1840514.7374 99	chr1	4351924 1	126M	=	4352244 413	ATCTTGGCTTTTCCCTTCTGGTGCACACGATGAGAGATTCATGGTAACTTGGCAGGAAGCAAGT
ACTTTTGTATGTCGTAATTTCCTGGTTTAAATGGCTCCCTTCCAGCTGCCACCACAGCTCCA	BBBBBBF11?1E@F@DGG>E@G11CDD0CG@GGC<F0=1E1=:1:FG>DFCE1EFC/BDE1EBCFG1FD000=EG>@FFG
FB0=B0:BD@G0==C>>00000BBF@@@@>F>6FGGE-504A=FC@	AS:i:-3 XN:i:0	XM:i:1	XO:i:0	XG:i:0	NM:i:1	MD:Z:40C85	YS:i:-7 YT:Z:CP NH:i:1
```

第三步：reads 计数

featureCounts 介绍：

1、在高通量测序分析中用于下游分析的关键信息是比对到每个 genomic feature（外显子、基因等）中的 read 数目，而 read counts 是下游很多分析进程的输入文件

2、featureCounts 是一款使用于 RNA-seq 和 DNA-seq 的 read summarization 工具，应用了高效率的染色体哈希算法和 feature 区块技术，它比目前存在的工具速度都快，而且需要的内存空间少，同时可以用于单端和双端的数据

Command:

$featureCounts -p –a <gtf/gff_files> -o <alignment_files>

Parameters:

–p: If specified, fragments (or templates) will be counted instead of reads. This option is only applicable for paired-end reads; single-end reads are always counted as reads.

-a: Name of an annotation file. GTF/GFF format by default.

-o: Name of output file including read counts. A separate file including summary statistics of counting results is also included in the output ('<string>.summary'). Both files are in tab delimited format.

单个样本：

featureCounts -p -t exon -g gene_id -a ./annotation/gencode.vM25.annotation.gtf -o test.featurecount_new.txt ./time0_rep1_align.sam

多个样本：

featureCounts -p -t exon -g gene_id -a ./annotation/gencode.vM25.annotation.gtf -o sample_featurecount_txt ./time0_rep1_sorted.bam ./time0_rep2_sorted.bam ./time1_rep1_sorted.bam ./time1_rep2_sorted.bam

结果文件：

ENSMUSG00000064357.1	chrM	7927	8607	+	681	0	1	102	198
ENSMUSG00000064358.1	chrM	8607	9390	+	784	0	0	64	114
ENSMUSG00000064359.1	chrM	9391	9458	+	68	0	0	1	0
ENSMUSG00000064360.1	chrM	9459	9806	+	348	0	2	11	63
ENSMUSG00000064361.1	chrM	9808	9875	+	68	0	0	0	0
ENSMUSG00000065947.3	chrM	9877	10173	+	297	0	0	9	24
ENSMUSG00000064363.1	chrM	10167	11544	+	1378	12	5	110	131

文件包含了 featureCount 结果信息，如果想了解每个基因上的 count 数，则只需要提取出第 7 列到 10 列的信息。

准备下一步输入格式

删除第一行

sed -i '1d' sample_featurecount_txt

提取 count

cut -f 1,7,8,9,10 sample_featurecount_txt > sample_featurecount_input.txt

第四步：差异表达分析

差异基因表达分析通常都是在 R 软件中，用各种 R 包进行分析的，常用的有 DESeq2，edgeR，limma 等几种。这次主要介绍用 DESeq2 来进行差异表达分析

DESeq2 对于输入数据的要求

一, DEseq2 要求输入数据是由整数组成的矩阵。

二, DESeq2 要求矩阵是没有标准化的。

DESeq2 进行差异表达分析

DESeq2 包分析差异表达基因简单来说只有三步：构建 dds 矩阵，标准化，以及进行差异分析。

构建 dds 矩阵：

BiocManager::install("DESeq2")

setwd('/lustre/home/acct-lctest/stu411/RNA-seq/process_data')

library("DESeq2")

input_data <- read.table("sample_featurecount_input.txt",header = TRUE,row.names =1)

input_data <- as.matrix(input_data)

condition <- factor(c(rep("time0",2),rep("time1",2)))

coldata <- data.frame(row.names=colnames(input_data),condition)

dds <- DESeqDataSetFromMatrix(countData =input_data,colData=coldata,design= ~condition)

```
> head(dds)
class: DESeqDataSet
dim: 6 4
metadata(1): version
assays(1): counts
rownames(6): ENSMUSG00000102693.1 ENSMUSG00000064842.1 ... ENSMUSG00000103377.1
  ENSMUSG00000104017.1
rowData names(0):
colnames(4): ..time0_rep1_sorted.bam ..time0_rep2_sorted.bam ..time1_rep1_sorted.bam
  ..time1_rep2_sorted.bam
colData names(1): condition
```

构建 dds 矩阵需要：

表达矩阵，即上述代码中的 input_data，就是我们前面通过 read count 计算后并融合生成的矩阵，行为各个基因，列为各个样品，中间为计算 reads 或者 fragment 得到的整数。我们后面要用的是这个文件（sample_featurecount_input.txt）

样品信息矩阵，即上述代码中的 coldata，它的类型是一个 dataframe（数据框），

第一列是样品名称，第二列是样品的处理情况（对照还是处理等），即 condition，condition 的类型是一个 factor。这些信息可以从 sample_featurecount_input.txt 中导出，也可以自己单独建立。

差异比较矩阵，即上述代码中的 design。 差异比较矩阵就是告诉差异分析函数是要从要分析哪些变量间的差异，简单说就是说明哪些是对照哪些是处理。

初步筛选

去掉丰度小的基因。丰度小的基因，一是可能不重要，二是误差太大.

删除 count 数小于 1 或等于 1 的

dds1 <- dds[rowSums(counts(dds)) > 1,]

对原始 dds 进行 normalize

dds2 <- DESeq(dds1)

说明：DESeq 包含三步，estimation of size factors（estimateSizeFactors），estimation of dispersion （estimateDispersons），Negative Binomial GLM fitting and Wald statistics（nbinomWaldTest），可以分布运行，也可用一步到位，最后返回 results 可用的 DESeqDataSet 对象。

```
> dds2 <- DESeq(dds1)
estimating size factors
estimating dispersions
gene-wise dispersion estimates
mean-dispersion relationship
-- note: fitType='parametric', but the dispersion trend was not well captured by the
   function: y = a/x + b, and a local regression fit was automatically substituted.
   specify fitType='local' or 'mean' to avoid this message next time.
final dispersion estimates
fitting model and testing
```

提取差异分析结果

res_0.05 <- results(dds2,alpha=0.05)

res_0.05 <- res_0.05[order(res_0.05$padj),]

resdata_0.05=merge(as.data.frame(res_0.05),as.data.frame(counts(dds2,normalized=TRUE)),by="row.names",sort=FALSE)

write.csv(resdata_0.05,file="diff_gene_whole.csv",quote=F,row.names=F)

```
> head(res_0.05)
log2 fold change (MLE): condition time1 vs time0
Wald test p-value: condition time1 vs time0
DataFrame with 6 rows and 6 columns
                            baseMean      log2FoldChange             lfcSE                stat
                           <numeric>           <numeric>         <numeric>           <numeric>
ENSMUSG00000025903.14 2.55339932081856  -0.898452263708777 1.52195273474297 -0.590328624009806
ENSMUSG00000033813.15 4.52031462251212    -1.4859630921935 1.18289789121681  -1.25620571583312
ENSMUSG00000062588.4  2.37841423000544    4.95670559995155 2.91545314518002    1.7001492917649
ENSMUSG00000033793.12 2.43025289194238 -0.0643400509877125 1.58015265310895 -0.040717617289142
ENSMUSG00000025907.14 7.55569089764318    4.09661187854398 1.25117037418615   3.27422384917697
ENSMUSG00000084353.1  1.43144020356689  -0.491717497132436  2.4970430608237 -0.196919910932667
                                pvalue              padj
                             <numeric>         <numeric>
ENSMUSG00000025903.14    0.55497035265655 0.735376840874505
ENSMUSG00000033813.15   0.209041394944393 0.401252152489529
ENSMUSG00000062588.4   0.0891028475748789 0.245517896584046
ENSMUSG00000033793.12   0.967521016676454 0.980032064305891
ENSMUSG00000025907.14 0.00105952642376644 0.015288780413121
ENSMUSG00000084353.1    0.843890212077117                NA
```

按照自定义的阈值提取差异基因并导出

通常认为 padj（p 值经过多重校验校正后的值）小于 0.05，表达倍数取以 2 为对数后大于 1 或者小于-1 的是差异表达基因。

diff_gene_deseq2 <-subset(resdata_0.05, padj < 0.05 & abs(log2FoldChange) > 1)

write.csv(diff_gene_deseq2,file= "DEG.csv",quote=F,row.names=F)

DEG.csv：

Row.names	baseMean	log2FoldCha	lfcSE	stat	pvalue	padj	..time0_rep1_sorted.bam	..time0_rep2_sorted.bam	..time1_rep1_sorted.bam	..time1_rep2_sorted.bam
ENSMUSG00000064370.1	188.470506	2.89921096	0.28652594	10.118494	4.57E-24	7.52E-21	45.40840642	43.70683894	350.8160989	313.9506784
ENSMUSG00000064351.1	112.169937	3.32181945	0.34559934	9.61176434	7.13E-22	5.87E-19	19.34061755	21.4410908	222.3817305	185.5163099
ENSMUSG00000064363.1	75.2032396	4.33795754	0.47083857	9.21325871	3.16E-20	1.73E-17	10.09075698	4.123286693	130.8127827	155.7861321
ENSMUSG00000064339.1	163.896476	2.79820467	0.3093346	9.04588319	1.48E-19	6.11E-17	38.6812351	43.70683894	334.1671993	239.0306301
ENSMUSG00000064341.1	95.842698	3.06800896	0.36368482	8.43590055	3.29E-17	1.08E-14	24.38599604	16.49314677	195.0299669	147.4616823
ENSMUSG00000023279.7	50.9402369	-4.173104	0.58735904	-7.1048605	1.20E-12	3.30E-10	89.97591643	103.0821673	3.567621345	7.13524269
ENSMUSG00000064345.1	51.4907656	3.090181	0.4388538	7.04148174	1.90E-12	4.47E-10	10.09075698	11.54520274	92.75815497	91.56894786
ENSMUSG00000041995.8	75.2024199	-2.7872673	0.40749994	-6.8399206	7.92E-12	1.63E-09	154.7249404	108.0301113	22.59493519	15.4596925
ENSMUSG00000021395.11	125.878268	-2.079699	0.30658501	-6.7834337	1.17E-11	2.14E-09	203.4969325	203.6903626	48.75749172	47.5682846
ENSMUSG00000032191.6	46.2124046	-4.6380615	0.69229035	-6.6995901	2.09E-11	3.44E-09	106.7938447	70.92053111	3.567621345	3.567621345
ENSMUSG00000106106.3	1900.74022	1.99352921	0.3145002	6.3387216	2.32E-10	3.46E-08	922.4633675	603.6491718	3250.103045	2826.745312
ENSMUSG00000032999.10	39.0201276	-4.3838385	0.69441526	-6.3129927	2.74E-10	3.75E-08	68.95350605	79.99176183	2.37841423	4.75682846
ENSMUSG00000032056.10	41.4287283	-3.5441316	0.57026109	-6.214928	5.13E-10	6.50E-08	89.13502002	63.49861506	7.13524269	5.946035575
ENSMUSG00000064358.1	52.9197166	9.43246586	1.56645697	6.02152887	1.73E-09	2.03E-07	0	0	76.10925536	135.5696111
ENSMUSG00000036036.15	34.9247902	-4.826117	0.81609038	-5.9137041	3.34E-09	3.64E-07	69.79440247	65.14792974	1.189207115	3.567621345
ENSMUSG00000064357.1	89.396698	8.74308135	1.48076774	5.90442452	3.54E-09	3.64E-07	0	0.824657339	121.2991257	235.4630088
ENSMUSG00000035191.15	45.209063	-6.2291357	1.0812992	-5.7607882	8.37E-09	8.10E-07	102.5893627	75.86847514	1.189207115	1.189207115
ENSMUSG00000004642.13	30.2642603	-3.551546	0.64880164	-5.474009	4.40E-08	4.02E-06	53.81737058	57.7260137	3.567621345	5.946035575
ENSMUSG00000064354.1	50.1569229	7.9061986	1.48399668	5.32763901	9.95E-08	8.61E-06	0.840896415	0	79.67687671	120.1099186

保存好的 **DEG.csv** 文件可以用于后续功能富集分析。

第五步：功能富集分析

富集分析分成两类

一：差异基因富集分析（不需要表达值，只需要 gene name）

二：基因集(gene set)富集分析（不管有无差异，需要全部 genes 表达值）

这里我们主要介绍第一种，将上步得到的差异基因进行 GO 和 KEGG 注释。

富集分析方法：

一，在线版：最主流的就是 DAVID,各种级别杂志总见其身影，使用非常简单。

二，客户端版：IPA（IPA 不是用的 GO 和 KEGG 数据库）和 FUNRICH，前者收费。后者是一个独立的软件工具，主要用于基因和蛋白质的功能富集和相互作用

网络分析。分析结果可以绘制成 Venn，Bar，Column，Pie 等图

三，R 包：比较常用的是 Y 叔的 clusterProfiler，以用来做各种富集分析，如 GO、KEGG、DO（Disease Ontology analysis）、Reactome pathway analysis 以及 GSEA 富集分析等。而除了富集分析，他还可以非常方便的对富集分析结果进行可视化。这里使用 clusterProfiler 进行 GO、KEGG 分析。

clusterProfiler 介绍

clusterProfiler 是一个 R 包，可以用来做各种富集分析，如 GO、KEGG、DO（Disease Ontology analysis）、Reactome pathway analysis 以及 GSEA 富集分析等。

加载包，导入数据

```
BiocManager::install("clusterProfiler")
library("clusterProfiler")
library("org.Mm.eg.db")
gene <- read.csv("DEG.csv",header = TRUE,sep=',')
```

将原始的 ENSEMBL id 的版本号去掉

```
ENSEMBL <- gsub("\\.\\d*", "", gene$Row.names)
gene$Row.names <- ENSEMBL
```

由于 clusterProfiler 富集分析推荐的输入文件是 Entrez ID，我们需要将 ENSEMBL id，转成 Entrez ID，clusterProfiler 提供了 bitr 方法进行转换

id 转换

```
id <- gene[,1]
li<-
bitr(id,fromType="ENSEMBL",toType=c("GENENAME","SYMBOL","ENTREZID
"),OrgDb =org.Mm.eg.db)
```

GO 富集

```
ego <- enrichGO(gene = li[,4],keyType = "ENTREZID",OrgDb =
org.Mm.eg.db,pvalueCutoff = 0.05, pAdjustMethod = "BH",qvalueCutoff = 0.05,ont =
"BP",readable = TRUE)
write.csv(as.data.frame(ego),"go_bp.csv",row.names =F)
```

KEGG 富集

```
ekegg <- enrichKEGG(li[,4], keyType = "kegg",organism = "mmu", pvalueCutoff =
0.05, pAdjustMethod = "BH", qvalueCutoff = 0.05)
ekegg1 <- setReadable(ekegg, OrgDb = org.Mm.eg.db, keyType="ENTREZID")
write.csv(as.data.frame(ekegg1),"kegg.csv",row.names =F)
```

结果文件：

go.bp.csv

ID	Description	GeneRatio	BgRatio	pvalue	p.adjust	qvalue	geneID	Count
GO:0009167	purine ribonucleoside monophosphate metabolic process	15/207	267/23174	1.96E-08	1.81E-05	1.70E-05	CYTB/COX1/ND4/ND2/COX	15
GO:0009126	purine nucleoside monophosphate metabolic process	15/207	268/23174	2.07E-08	1.81E-05	1.70E-05	CYTB/COX1/ND4/ND2/COX	15
GO:0009161	ribonucleoside monophosphate metabolic process	15/207	271/23174	2.39E-08	1.81E-05	1.70E-05	CYTB/COX1/ND4/ND2/COX	15
GO:0009123	nucleoside monophosphate metabolic process	15/207	280/23174	3.69E-08	2.09E-05	1.96E-05	CYTB/COX1/ND4/ND2/COX	15
GO:0042773	ATP synthesis coupled electron transport	8/207	61/23174	6.95E-08	3.15E-05	2.96E-05	CYTB/COX1/ND4/ND2/COX	8
GO:0046034	ATP metabolic process	13/207	242/23174	2.98E-07	0.00011239	0.00010552	CYTB/COX1/ND4/ND2/COX	13
GO:0015980	energy derivation by oxidation of organic compounds	13/207	251/23174	4.51E-07	0.0001461	0.00013717	CYTB/COX1/ND4/ND1/ND2	13
GO:0019693	ribose phosphate metabolic process	17/207	444/23174	5.61E-07	0.00015884	0.00014912	CYTB/COX1/ND4/ND2/COX	17
GO:0022904	respiratory electron transport chain	8/207	81/23174	6.51E-07	0.00016396	0.00015393	CYTB/COX1/ND4/ND2/COX	8
GO:0009205	purine ribonucleoside triphosphate metabolic process	13/207	267/23174	9.07E-07	0.00019468	0.00018277	CYTB/COX1/ND4/ND2/COX	13
GO:0022900	electron transport chain	8/207	85/23174	9.45E-07	0.00019468	0.00018277	CYTB/COX1/ND4/ND2/COX	8

KEGG.csv

ID	Description	GeneRatio	BgRatio	pvalue	p.adjust	qvalue	geneID	Count
mmu00190	Oxidative phosphorylation	14/96	133/8766	1.43E-10	2.38E-08	2.19E-08	CYTB/COX1/ND4/ND1	14
mmu05012	Parkinson disease	14/96	247/8766	4.20E-07	3.51E-05	3.23E-05	CYTB/COX1/ND4/ND1	14
mmu04714	Thermogenesis	13/96	230/8766	1.17E-06	6.50E-05	5.98E-05	CYTB/COX1/ND4/ND1	13
mmu05016	Huntington disease	14/96	303/8766	4.78E-06	0.00019962	0.0001837	CYTB/COX1/ND4/ND1	14
mmu05010	Alzheimer disease	15/96	368/8766	9.76E-06	0.00032609	0.00030009	CYTB/COX1/ND4/ND1	15
mmu03010	Ribosome	9/96	175/8766	0.00011972	0.00333231	0.00306661	Rpl41/Rplp0/Rpl4/Rplp	9
mmu04110	Cell cycle	7/96	123/8766	0.0003828	0.00913252	0.00840434	Skp1a/Wee2/Ywhaz/C	7
mmu04140	Autophagy - animal	7/96	138/8766	0.0007645	0.01595897	0.01468648	Ctsl/Rb1cc1/Gabarapl2	7
mmu00270	Cysteine and methionine metabolism	4/96	52/8766	0.00244626	0.04312403	0.03968552	Ldhb/Cdo1/Mdh2/Mat	4
mmu04260	Cardiac muscle contraction	5/96	87/8766	0.00258228	0.04312403	0.03968552	CYTB/COX1/COX3/CO	5
mmu04142	Lysosome	6/96	131/8766	0.00306971	0.04660376	0.0428878	Ctsl/Ctsc/Cd63/Ctsb/C	6