数据的获取

数据集来自各类文献，存储在NCBI的SRA数据库中

- 20180228--published isogenic

paper	clones	cell types for RNA-seq/microarray	Caucasian/Asian	donor cell type	reprograming	gene editing method	GENDER	RNA-seq/microarray	Accession
Wang et al. Molecular Autism (2015)	2 WT(parental clones) vs 2 +/-	NPC and neuron	Caucasian	skin fibroblast	episomal (OSKMLshP53)	CAS9	male	RNA-seq	GSE71594
Wang et al. Molecular Autism (2017)	2 WT(parental clones) vs 4 +/-	neural organoid	Caucasian	skin fibroblast	episomal (OSKMLshP53)	CAS9	male	RNA-seq	GSE85417
Theodoris et al., 2015, Cell	3 corrected vs 2 TALEN-mut	endothelial cells	Caucasian	skin fibroblast	episomal (OSK)	TALEN	female	RNA-seq	GNomEx:351R
Xu et al.,Stem Cell Reports (2017)	1 mut vs 3 corrected	iPSC and NPC	Caucasian	fibroblast	lenti-teton (OSKMNL)	piggybac	male	microarray	GSE93767

paper	clones	cell types for RNA-seq/microarray	Caucasian/Asian	donor cell type	reprograming	gene editing method	GENDER	RNA-seq/microarray	Accession
Zhang et al.,Stem Cell Reports (2017)	1+1+1 mut(different individual) vs 1+1+1 corrected	neuron	Caucasian	skin	fibroblast	episomal(OSK)	CAS9	RNA-seq	GSE92340
Liu, G.-H. et al. Nat. Commun. (2014)	1 mut vs 1 corrected	Ipsc	Caucasian	skin	fibroblast episomal (OSKLshP53)	HDAdV	male	RNA-seq/microarray	GSE57828, GSE40865
Li Y, et al., 2013, Cell Stem Cell	1 wt WIBR1 vs 1 mut WIBR1,2 wt WIBR3 vs 2 mut WIBR3	NPC and neuron	Caucasian	TALEN	WIBR1	male, WIBR3	female	microarray	GSE50584
Ryan S. D., et al., 2013, Cell	1 mut vs 1 corrected	neuron	Caucasian	skin	fibroblast	excisable lenti	ZFN	microarray	GSE46798

paper	clones	cell types for RNA-seq/microarray	Caucasian/Asian	donor cell type	reprograming	gene editing method	GENDER	RNA-seq/microarray	Accession
Kiskinis E., et al., 2014, Cell Stem Cell	1 mut vs 1 corrected	neuron	Caucasiain	skin	fibroblast	retro OSKM ZFN	female	RNA-seq	GSE54409
Bhinge et al.,Stem Cell Reports (2017)	1 mut vs 1 corrected	neuron	Caucasiain	fibroblast	retro OSKM	CAS9	male	RNA-seq	SRX2494130, SRX2494129, SRX2494128, SRX2494127
Matsa E., et al., 2016, Cell Stem Cell	1 mut vs 1 AAVS1 overexpressoin	cardiomyocytes	Caucasiain	fibroblast	CoMiP	OSK (non-integration)	CAS9	RNA-seq	GSE83668
Reinhardt P., et al., 2013, Cell Stem Cell	1+1(same individual)+1(different individual) mut vs 1+1+1corrected	neuron	Caucasiain	fibroblast	retro OSKM	ZFN	female+male	microarray	GSE43364

paper	clones	cell types for RNA-seq/microarray	Caucasian/Asian	donor cell type	reprograming	gene editing method	GENDER	RNA-seq/microarray	Accession
Ring et al., Stem Cell Reports (2015)	1 mut vs 1 corrected, 8 replicates	ipsc and neuron	Caucasian	fibroblast retroOSKM	homologous	recombination	female	RNA-seq	GSE74201

补充数据

SRA.list

SRR2138040 SRR2138039 SRR2138038 SRR2138037 SRR2138036 SRR2138035 SRR2138034
SRR2138033 SRR4017393 SRR4017392 SRR4017391 SRR4017390 SRR4017382 SRR4017383
SRR4017388 SRR4017389 SRR4017386 SRR4017387 SRR4017384 SRR4017385 SRR5101852
SRR5101853 SRR5101854 SRR5101855 SRR5101856 SRR5101857 SRR5101858 SRR1145035
SRR1145036 SRR1145037 SRR1145038 SRR1145039 SRR5177947 SRR5177946 SRR5177945
SRR5177944 SRR3713906 SRR3713907 SRR3713908 SRR3713909 SRR3713910 SRR3713911
SRR3713912 SRR3713913 SRR3713914 SRR3713915 SRR3713916 SRR3713917 SRR3713918
SRR3713919 SRR3713920 SRR3713921 SRR3713922 SRR3713923 SRR3713924 SRR3713925
SRR3713926 SRR3713927 SRR3713928 SRR3713929 SRR3713930 SRR3713931 SRR3713932
SRR3713933 SRR3713934 SRR3713935 SRR3713936 SRR3713937 SRR3713938 SRR3713939
SRR3713940 SRR2734288 SRR2734287 SRR2734286 SRR2734285 SRR2734283 SRR2734284
SRR2734281 SRR2734280 SRR2734278 SRR2734279 SRR2734277 SRR2734276 SRR2734275
SRR2734274 SRR2734273 SRR2734272 SRR2734270 SRR2734269 SRR2734268 SRR2734267
SRR2734266 SRR2734265 SRR2734264 SRR2734263 SRR2734262 SRR2734271 SRR2734261
SRR2734260 SRR2734259 SRR2734258 SRR2734257 SRR2734256 SRR5495039 SRR5495040
SRR5495041 SRR5495042 SRR5495043 SRR5495044 SRR5495045 SRR5495046 SRR5495047
SRR5495048 SRR5495049 SRR5495050 SRR5217330 SRR5217331 SRR5217332 SRR5217333
SRR5217334 SRR5217335 SRR5217336 SRR5217337 SRR5217338 SRR5217339 SRR5217340
SRR5217341 SRR5217342 SRR5217343 SRR5217344 SRR5217345 SRR5217346 SRR5217347
SRR5217348 SRR5217349 SRR5217350 SRR5217351 SRR5217352 SRR5217353 SRR5217354
SRR5217355 SRR5217356 SRR5217357 SRR5217358 SRR5217359 SRR5217360 SRR5217361
SRR5217362 SRR5217363 SRR5217364 SRR5217365 SRR5217366 SRR5217367 SRR5217368
SRR5217369 SRR5217370 SRR5217371 SRR5217372 SRR5217373 SRR5217374 SRR5217375
SRR5217376 SRR5217377 SRR5217378 SRR5217379 SRR5217380 SRR5217381 SRR5217382
SRR5217383 SRR5256862 SRR5256863 SRR5256864 SRR5256865 SRR5256866 SRR5256867
SRR5256868 SRR5256869 SRR5256870 SRR5256871 SRR5256872 SRR5256873 SRR5256874
SRR5256875 SRR5256876 SRR5256877 SRR5256878 SRR5256879 SRR5256880 SRR5256881
SRR5256882 SRR5256883 SRR5273291 SRR5273292 SRR5273293 SRR5273294 SRR5408221
SRR5408222 SRR5408223 SRR5408224 SRR5408225 SRR5408226 SRR5408227 SRR5408228
SRR5408229 SRR5408230 SRR5408231 SRR5408232 SRR5408233 SRR5408234 SRR5408235
SRR5408236 SRR5408237 SRR5408238 SRR5408239 SRR5408240 SRR5408241 SRR5408242
SRR5408243 SRR5408244 SRR5408245 SRR5408246 SRR5408247 SRR5408248 SRR5408249
SRR5408250 SRR5408251 SRR5408252 SRR5408253 SRR5408254 SRR5408255 SRR5408256
SRR5408257 SRR5408258 SRR5408259 SRR5408260 SRR5408261 SRR5408262 SRR5408263
SRR5408264 SRR5408265 SRR5408266 SRR5408267 SRR5408268 SRR5408269 SRR5408270
SRR5408271 SRR5408272 SRR5408273 SRR5408274 SRR5408275 SRR5408276 SRR5408277
SRR5408278 SRR5408279 SRR5408280 SRR5408281 SRR5408282 SRR5408283 SRR5408284
SRR5408285 SRR5408286 SRR5408287 SRR5408288 SRR5408289 SRR5408290 SRR5408291
SRR5408292 SRR5408293 SRR5408294 SRR5408295 SRR5408296 SRR5408297 SRR5408298
SRR5408299 SRR5408300 SRR5408301 SRR5408302 SRR5408303 SRR5408304 SRR5408305
SRR5408306 SRR5408307 SRR5408308 SRR5408309 SRR5408310 SRR5408311 SRR5408312
SRR5408313 SRR5408314 SRR5408315 SRR5408316 SRR5408317 SRR5408318 SRR5408319
SRR5408320 SRR5408321 SRR5408322 SRR5408323 SRR5408324 SRR5408325 SRR5408326
SRR5408327 SRR5408328 SRR5408329 SRR5408330 SRR5408331 SRR5408332 SRR5408333
SRR5408334 SRR5408335 SRR5408336 SRR5408337 SRR5408338 SRR5408339 SRR5408340
SRR5408341 SRR5408342 SRR5408343 SRR5408344 SRR5408345 SRR5408346 SRR5408347
SRR5408348 SRR5408349 SRR5408350 SRR5408351 SRR5408352 SRR5408353 SRR5408354
SRR5408355 SRR5408356 SRR5408357 SRR5408358 SRR5408359 SRR5408360 SRR5408361
SRR5408362 SRR5408363 SRR5408364 SRR5408365 SRR5408366 SRR5408367 SRR5408368

SRR5408369 SRR5408370 SRR5408371 SRR5408372 SRR5408373 SRR5408374 SRR5408375
SRR5408376 SRR5408377 SRR5408378 SRR5408379 SRR5408380 SRR5408381 SRR5408382
SRR5408383 SRR5408384 SRR5408385 SRR5408386 SRR5408387 SRR5408388 SRR3126112
SRR3126111 SRR3126110 SRR3126109 SRR3126108 SRR3126107 SRR3126106 SRR3126105
SRR3126104 SRR3126103 SRR3126102 SRR3126101 SRR3126100 SRR3126099 SRR2453324
SRR2453325 SRR2453323 SRR2453321 SRR2453322 SRR2453320 SRR2453319 SRR2453318
SRR2453317 SRR2453316 SRR2453315 SRR2453312 SRR2453313 SRR2453314 SRR2453311
SRR2453310 SRR2453309 SRR2453308 SRR2453307 SRR2453306 SRR2453305 SRR2453303
SRR2453304 SRR2453302 SRR2453301 SRR2453299 SRR2453300 SRR2453371 SRR2453370
SRR2453369 SRR2453366 SRR2453367 SRR2453368 SRR2453365 SRR2453364 SRR2453363
SRR2453362 SRR2453361 SRR2453359 SRR2453360 SRR2453357 SRR2453356 SRR2453358
SRR2453355 SRR2453354 SRR2453353 SRR2453352 SRR2453351 SRR2453350 SRR2453349
SRR2453348 SRR2453346 SRR2453347 SRR2453345 SRR2453343 SRR2453344 SRR2453341
SRR2453340 SRR2453339 SRR2453337 SRR2453338 SRR2453336 SRR2453335 SRR2453334
SRR2453332 SRR2453333 SRR2453342 SRR2453331 SRR2453329 SRR2453330 SRR2453328
SRR2453326 SRR2453327 SRR1574511 SRR1574513 SRR1574515 SRR1574516 SRR1574517
SRR1574512 SRR1574518 SRR1574519 SRR1574520 SRR1574522 SRR1574523 SRR1574521
SRR1574524 SRR1574526 SRR1574525 SRR1574527 SRR1574529 SRR1574528 SRR1574530
SRR1574531 SRR1574532 SRR1574533 SRR1574514 SRR1799891 SRR1799889 SRR1799888
SRR1799890 SRR1799886 SRR1799885 SRR1799887 SRR1799882 SRR1799883 SRR1799884
SRR1799880 SRR1799881 SRR1799879 SRR1799878 SRR1799877 SRR1799876 SRR1800137
SRR1800136 SRR1800138 SRR1800135 SRR1182374 SRR2939131 SRR1182375 SRR2939132
SRR1182376 SRR2939133 SRR1182377 SRR2939134 SRR1182379 SRR2939136 SRR1182378
SRR2939135 SRR1182380 SRR2939137 SRR1182381 SRR2939138 SRR1182382 SRR2939139
SRR1182383 SRR2939140 SRR1182384 SRR2939141 SRR1182385 SRR2939142 SRR1182386
SRR2939143 SRR1182387 SRR2939144 SRR1182388 SRR2939145 SRR1182389 SRR2939146
SRR1182391 SRR2939148 SRR1182390 SRR2939147 SRR1182392 SRR2939149 SRR1182393
SRR2939150 SRR1182394 SRR2939151 SRR1182395 SRR2939152 SRR2910664 SRR2910663
SRR2910673 SRR2910672 SRR2910671 SRR2910670 SRR2910669 SRR2910668 SRR2910667
SRR2910666 SRR2910665 SRR1583685 SRR1583686 SRR1583687 SRR1583688 SRR1583689
SRR1583690 SRR1583691 SRR1583692 SRR1583693 SRR1583694 SRR1583695 SRR1583696
SRR1583697 SRR1583698 SRR1583699 SRR1292576 SRR1292577 SRR1292578 SRR1292579
SRR1292580 SRR1292581 SRR1292582 SRR1292583 SRR1292584

测序文件包含双端以及单端测序数据，因此我们指定命令来自行判断，对于双端文件生成2个fastq文件，对于单端则生成一个

```
for i in *
do
echo $i
/root/sratoolkit.2.9.2-centos_linux64/bin/fastq-dump --split-3 $i
done
##### 批量使用SRAtoolkit的fastq-dump 功能转换压缩格式
```

使用ftp协议进行传输，将文件从ESC挂载着的OSS传输到cluster

```
setsid scp -r  -C root@39.108.91.174:/root/ncbi/public/sra /public/home/miguel_gibh
### setsid将程序hold，使用scp进行加密压缩传输文件夹，可惜没有本地的大存储空间电脑，无法进行内网传输，浪费了大量时间
```

resm.pbs

```
#PBS -N rsem
#PBS -l walltime=20:00:00
#PBS -l nodes=1:ppn=16
#PBS -q default
#PBS -j oe
#PBS -V
cd $PBS_O_WORKDIR
cd /public/home/miguel_gibh/fastq
rsem-calculate-expression \
        -p 16 \
        --bowtie2 \
        --bowtie2-sensitivity-level very_sensitive \
        --no-bam-output \
        --estimate-rspd \
        --paired-end \
        *_1* *_2* \
        /public/software/genomics/genome/hg38/enst_v81_rsem/hg38_enst_v81 \
        ${input%_*}
```

同理我们使用循环批量处理fastq文件转化为resm表达矩阵

```
for i in *
do
qsub -v input=$i /public/home/miguel_gibh/resm.pbs
done
```

总结来讲数据采用统一处理的流程是

pipeline

SRA -> fasq-dump -> rsem

```
scp –r 要传的目录 root@目标ip:路径
setsid  scp -vrC root@39.108.91.174:/root/ncbi/public/sra/SRR4017392.sra
/public/home/miguel_gibh
```

sra to fastq

```
[shell]

for i in *sra
do
echo $i
/home/jmzeng/bio-soft/sratoolkit.2.3.5-2-ubuntu64/bin/fastq-dump --split-3 $i
done

[/shell]
```

查看下生成的fastq文件

```
head SRR1145038_1.fastq
@SRR1145038.1 ILLUMINA:382:D1K6UACXX:2:1101:1607:2222 length=100
CCGGCATCTTCAGCATGAGCCGCAGGCTCAGCCACCACTGCTCCCGTGGCATGCGGTGCTCGAAATCAGTGTCTTTCTTGAGCTCAGTGACG
GGGCTGAA
+SRR1145038.1 ILLUMINA:382:D1K6UACXX:2:1101:1607:2222 length=100
CCCFFFFFHHHHHJJJJJJJJJJJJJJJJIJIJJJJJJJIJJJJJIHFHIGHIHHFFFDEECDDBDCACDDDCCDACDDCCDDDDCDCCCCD
D<BDDBB@
@SRR1145038.2 ILLUMINA:382:D1K6UACXX:2:1101:1560:2243 length=100
TGTCCCGAGACCGCAACTGCCTGGACCCGTTCACCAGCCGCCAGAGCCTGCAGGCACTAGCCTGCTATGCTGACATCTCTGTCTCTGAGGGG
TCCGTCCC
+SRR1145038.2 ILLUMINA:382:D1K6UACXX:2:1101:1560:2243 length=100
BCCFFFD:FDHHHJJJJJHIJIJJJJIJIIIIHJJJIEGIIJHHHFFFFFEEEEDDDDDDDDDDDDDDDDEDC?
@>>CDDDDDDDDD:CBDD@BBBBBDB
@SRR1145038.3 ILLUMINA:382:D1K6UACXX:2:1101:1802:2078 length=100
ATCAGAAGCCCTTTCCTTTCTTTTACCACAAACAACCTGATCTCACCAAACAGGAAACCTCAGCATCAAGATGGCAAAGGAGCCAAGTGGGA
GATCGGAA
```

script to do RSEM on paired end fastq files

这个脚本可以用来处理所有的pair end，因此我们需要根据文件名来进行修改，（PS，当你使用单端测序的时候，你就可以删除R2）

STAR + RSEM （先比对，再定量，耗时长）

输出结果可以选择转录本定量或者基因定量 定量单位包括 feature count, FPKM, TPM 操作相对复杂

```
#PBS -N rsem
#PBS -l walltime=20:00:00
#PBS -l nodes=1:ppn=16
#PBS -q default
#PBS -j oe
#PBS -V
cd $PBS_O_WORKDIR
cd ${input}
rsem-calculate-expression \
        -p 16 \
        --bowtie2 \
        --bowtie2-sensitivity-level very_sensitive \
        --no-bam-output \
        --estimate-rspd \
        --paired-end \
        *R1* *R2* \
        /public/software/genomics/genome/hg38/enst_v81_rsem/hg38_enst_v81 \
        ${input%_*}
```

we can use this to

```
for i in */
do
qsub -v input=$i <this_script.sh>
done
##### 作业提交的绝对路径是在$PBS_O_WORKDIR ，因此返回提交目录
cd $PBS_O_WORKDIR

###### for i in */   根目录下的所有文件的名称全部赋值给刚刚声明的变量i

###### RSEM整体上来说是属于定量软件，但其支持调用其他比对软件，如Bowtie，Bowtie2和STAR，来将reads比对至转录本上

###### 用RSEM的rsem-calculate-expression命令来对reads进行bowtie2比对以及表达水平的定量
```

查看我们得到的表达量文件

```
head SRR1145038.genes.results
gene_id transcript_id(s)    length  effective_length    expected_count  TPM FPKM
ENSG00000000003
ENST00000373020,ENST00000494424,ENST00000496771,ENST00000612152,ENST00000614008 2140.85
2095.38 207.00  50.17   24.91
ENSG00000000005 ENST00000373031,ENST00000485971 940.50  895.03  0.00    0.00    0.00
ENSG00000000419
ENST00000371582,ENST00000371584,ENST00000371588,ENST00000413082,ENST00000466152,ENST00000494
752 1075.00 1029.52 134.00  66.13   32.84
ENSG00000000457
ENST00000367770,ENST00000367771,ENST00000367772,ENST00000423670,ENST00000470238 3794.89
3749.41 88.01   11.92   5.92
ENSG00000000460
ENST00000286031,ENST00000359326,ENST00000413811,ENST00000459772,ENST00000466580,ENST00000472
795,ENST00000481744,ENST00000496973,ENST00000498289 2192.24 2146.77 17.00   4.02    2.00
ENSG00000000938
ENST00000374003,ENST00000374004,ENST00000374005,ENST00000399173,ENST00000457296,ENST00000468
038,ENST00000475472 1735.29 1689.81 0.00    0.00    0.00
ENSG00000000971
ENST00000359637,ENST00000367429,ENST00000466229,ENST00000470918,ENST00000496761,ENST00000630
130 1658.00 1612.52 1.00    0.31    0.16
ENSG00000001036 ENST00000002165,ENST00000367585,ENST00000451668 2118.05 2072.57 106.77
26.16   12.99
ENSG00000001084
ENST00000229416,ENST00000504353,ENST00000505197,ENST00000509541,ENST00000510837,ENST00000513
939,ENST00000514004,ENST00000514373,ENST00000514933,ENST00000515580,ENST00000616923 2715.72
2670.25 156.00  29.66   14.73
```

```
head SRR1145038.isoforms.results
transcript_id   gene_id length  effective_length    expected_count  TPM FPKM    IsoPct
ENST00000373020 ENSG00000000003 2206    2160.52 167.55  39.38   19.55   78.50
ENST00000494424 ENSG00000000003 820 774.52  0.00    0.00    0.00    0.00
ENST00000496771 ENSG00000000003 1025    979.52  14.20   7.37    3.66    14.69
ENST00000612152 ENSG00000000003 3796    3750.52 25.24   3.42    1.70    6.81
ENST00000614008 ENSG00000000003 900 854.52  0.00    0.00    0.00    0.00
ENST00000373031 ENSG00000000005 1339    1293.52 0.00    0.00    0.00    0.00
ENST00000485971 ENSG00000000005 542 496.53  0.00    0.00    0.00    0.00
ENST00000371582 ENSG00000000419 1161    1115.52 0.00    0.00    0.00    0.00
ENST00000371584 ENSG00000000419 1073    1027.52 0.00    0.00    0.00    0.00
```

对于我们得到的resm文件，我们使用以下两个脚本进行处理

```
############################

###### 2a.gc_norm.R  #####

############################
```

```
rm(list =ls())
# EDASeq-based GC normlisation of data
# I use DESeq here to get a eset object.
library("EDASeq")
files <- Sys.glob(file.path("./RESM","*.genes.results"))
output <- vector("list", length(files))
for (i in seq_along(files)) {
  x <- read.table(files[i], stringsAsFactors = F, header = T)
  n <- stringr::str_replace(files[i], ".*/", "")
  n <- stringr::str_replace(n, "\\..*", "")
  x <- dplyr::select(x, c(gene_id, expected_count))
  rownames(x) <- x$gene_id
  x <- dplyr::select(x, -gene_id)
  colnames(x) <- n
  output[[i]] <- x
}
count_table <- rbind.data.frame(output)


geneIDs = row.names(x)
row.names(count_table) <- geneIDs ###my

# Filtering criteria:
filter_threshold = 10^1.8 # number of tags in <num_conditions> conditions to consider
expressed (This number is POST normalisation)
num_conditions = 1 # number of conditions that must be greater than <filter_threshold>
# Because of the lack of robust replicates and the relatively high filter_threshold
# It is better to keep this value low. As it might bias genes who are only expressed in a
single cell type
# Against being detected.

# Load data:
gc = read.delim("./ensg_gcpercent_v81-nodupes.txt", row.names=1) / 100.0
design = data.frame(1:dim(count_table)[2], row.names=colnames(count_table))
rownames(design) = colnames(count_table)

# Make sure gc and count_table match exactly
common = intersect(rownames(gc), rownames(count_table))  ####intersect计算交集
df = data.frame(gc=gc, spaz=gc)
feature = df[common,]
colnames(feature) = c("gc", "fill")
count_table = count_table[common,]

pdf("LibSizes.pdf")
sums = colSums(count_table)
m = mean(colSums(count_table))
s = sd(colSums(count_table))
hist(colSums(count_table), breaks=50, main="Mapped Library Sizes", xlab="Number of mapped
sequence tags")
abline(v=m, col="red")
abline(v=m+s, col="blue")
abline(v=m-s, col="blue")
sums[sums==min(sums)]
sums[sums==max(sums)]
dev.off()

pdf("GeneSizes.pdf")
sums = rowSums(count_table)
m = mean(log10(sums))
```

```
s = sd(log10(sums))
hist(log10(sums), breaks=50, main="Mapped Library Sizes", xlab="Number of mapped sequence
tags")
abline(v=m, col="red")
abline(v=m+s, col="blue")
abline(v=m-s, col="blue")
sums[sums==max(sums)]
dev.off()

pdf("biasplots.pdf", width=15, height=7)
par(mfrow=c(1,2))
plot(density(log10(count_table[,1])), main="Before Norm")
for (i in 2:dim(count_table)[2]){
  lines(density(log10(count_table[,i])))
}

data <- newSeqExpressionSet(counts=as.matrix(count_table), featureData=feature,
                            phenoData=design) # will only tolerate integers

biasPlot(data, "gc", log=T, ylim=c(-4,8))

# normalise
cdswithin = withinLaneNormalization(data, "gc", which="full")
cdsnormed = betweenLaneNormalization(cdswithin, which="full")

norm_table = normCounts(cdsnormed)
plot(density(log10(norm_table[,1])), main="Post Norm")
for (i in 2:dim(norm_table)[2]){
  lines(density(log10(norm_table[,i])))
}
abline(v=log10(filter_threshold), col="grey")
biasPlot(cdsnormed, "gc", log=T, ylim=c(-4,8))

# Filter lowly expressed
keep <- rowSums(norm_table>filter_threshold) >= num_conditions # Filtering must be done here
BEFORE DE.
norm_table = norm_table[keep,]
print(paste("Kept:", dim(norm_table)[1]))

par(mfrow=c(1,1))
boxplot(data)
boxplot(cdswithin)
boxplot(cdsnormed)
dev.off()

write.table(norm_table, "rawtags_gc_normed.tsv", sep="\t", col.names=NA)
```

基因的GC含量会影响它被二代测序的reads数量，作者开发了一个R包来矫正这个表达量差异。 关于EDAseq的详细介绍(

https://www.bioconductor.org/packages/devel/bioc/vignettes/EDASeq/inst/doc/EDASeq.html

由于测序是将将mRNA转化为cDNA片段，然后对其进行测序，以产生数百万个短读数（通常为25-100个碱基）。然后将这些读数映射回参考基因组，并且映射到特定基因的读数的数量反映了目标样品中转录物的丰度。然而，原始计数既不能在泳道内的基因之间直接比较，也不能在给定基因的复制泳道（即测定相同文库的泳道）之间直接比较，并且需要对计数进行标准化以允许准确推断转录水平的差异。实际上，通过该测定，人们期望给定基因的读数与基因长度和其转录物丰度大致成比例。由于测序深度的差异，即给定泳道中产生的读数总数，读取计数也将在复制泳道之间变化。

此外，如文献综述中所述(https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-12-480)，先前的研究报道了与基因组区域的测序效率相关的选择偏差，其中读数不仅取决于长度，还取决于序列特征，例如GC含量和可映射性（即独特性）。相比于基因组的剩余部分的特定序列,例如富含GC和贫GC的片段在RNA-Seq中往往代表性不足，因此在泳道内，读数计数在基因之间不能直接比较。此外，GC含量效应往往是泳道特异性的，因此给定基因的读数不能在泳道之间直接比较。与长度和GC含量相关的偏差混淆了差异表达（DE）结果以及下游分析，例如涉及基因本体论（GO）的那些。由于GC-含量在整个基因组中变化并且通常与功能性相关，因此可能难以从有偏差的读数计量量推断出真实的表达水平。因此，正确的读数计数正常化对于准确推断表达水平的差异至关重要。

大型文件的传输

```
setsid ssh root@39.108.91.174  "cd /root/ncbi/public/sra  && tar czvf my.tar.gz
SRR1799881_1.fastq SRR1799881_2.fastq  SRR1799882_1.fastq SRR1799882_2.fastq
SRR1799884_1.fastq SRR1799884_2.fastq SRR2138036_1.fastq SRR2138036_2.fastq  " > my.tar.gz
```

or

```
setsid ssh root@39.108.91.174  "cd /root/ncbi/public/sra  && tar cvf my.tar.gz
SRR1799881_1.fastq SRR1799881_2.fastq  SRR1799882_1.fastq SRR1799882_2.fastq
SRR1799884_1.fastq SRR1799884_2.fastq SRR2138036_1.fastq SRR2138036_2.fastq  " > my.tar
```

在那之后，我们需要用DESeq2进行后续处理， 后续处理步骤可以从上一步得到的GCnormal表达矩阵开始。

RNA-seq workflow: gene-level exploratory analysis and differential expression

RNA-seq工作流程：基因水平探索性分析和差异表达

摘要

在这里，我们使用Bioconductor包进行end-to-end端到端（输入的是原始数据，得到的是需求结果）的基因水平RNA-seq差异表达工作流程。我们将从FASTQ文件开始，显示这些文件如何与参考基因组对齐，并准备计数矩阵，该计数矩阵计算每个样品的每个基因内RNA-seq读数/片段的数量。我们将进行探索性数据分析（EDA）以进行质量评估，探索样本之间的关系，进行差异基因表达分析，并直观地探索结果。

如何构建DESeqDataSet

```
> head(count_table,6)
                 SRR1145038 SRR1182379
ENSG00000282221          0          0
ENSG00000187223          0          0
ENSG00000110514        610       2461
ENSG00000086015        804       1090
ENSG00000223410          0          0
ENSG00000255071          0          0
```

在该计数矩阵中，每行代表Ensembl基因，每列代表测序的RNA文库，并且该值给出了在每个文库中唯一分配给相应基因的片段的原始数目。我们还有每个样本的信息（计数矩阵的列）。如果已使用其他软件计算读数，则检查计数矩阵的列是否与样本信息表的行相对应非常重要。

```
coldata <- colData(se)
```

下列的分析我们将直接使用来自于上一步的输出矩阵count_data，使用 DESeq得到 eset object

Quick start

DESeq2分析的基本步骤如下：

```
dds <- DESeqDataSetFromMatrix(countData = cts, colData = coldata, design = ~batch +
condition)
dds <- DESeq(dds)
res <- results(dds, contrast=c("condition","treated","control"))
```

- cts是count matrix
- coldata是table of sample information

数据输入大体分为三种

- Salmon, Sailfish, kallito等软件输出的transcript - quantification转录本定量文件，可以用DESeqDataSetFromTximport导入。htseq-count软件输出的结果可以用DESeqDataSetFromHTSeq导入。
- RangedSummarizedExperiment用DESeqDataSet导入。
- design是指明如何对样本进行比较的信息。上述代码中是为了在控制批次效应的同时研究condition间的效应。

导入count matrix矩阵

用DESeqDataSetFromMatrix命令导入countmatrix。除countmatrix外，使用者还需要将count matrix的列名单独建立一个DataFrame，以及一个design formula。 以下我们演示DESeqDataSetFromMatrix的用法，主要过程是从pasilla包中加载count data，读取的count matrix命名为cts，样品信息表命名为coldata。下边演示的方法是从featureCounts的输出结果中提取以上信息。有了count matrix（cts）和sample information（coldata），我们就可以构建DESeqDataSet矩阵了

```
library("DESeq2")
dds <- DESeqDataSetFromMatrix(countData=cts, colData = coldata, design = ~condition)
dds
```

	individual	replicate	well	batch	sample_id
2	NA19098	r1	A01	NA19098.r1	NA19098.r1.A01
3	NA19098	r1	A02	NA19098.r1	NA19098.r1.A02
4	NA19098	r1	A03	NA19098.r1	NA19098.r1.A03
5	NA19098	r1	A04	NA19098.r1	NA19098.r1.A04
6	NA19098	r1	A05	NA19098.r1	NA19098.r1.A05
7	NA19098	r1	A06	NA19098.r1	NA19098.r1.A06
8	NA19098	r1	A07	NA19098.r1	NA19098.r1.A07
9	NA19098	r1	A08	NA19098.r1	NA19098.r1.A08
10	NA19098	r1	A09	NA19098.r1	NA19098.r1.A09
11	NA19098	r1	A10	NA19098.r1	NA19098.r1.A10
12	NA19098	r1	A11	NA19098.r1	NA19098.r1.A11
13	NA19098	r1	A12	NA19098.r1	NA19098.r1.A12
14	NA19098	r1	B01	NA19098.r1	NA19098.r1.B01
15	NA19098	r1	B02	NA19098.r1	NA19098.r1.B02
16	NA19098	r1	B03	NA19098.r1	NA19098.r1.B03
17	NA19098	r1	B04	NA19098.r1	NA19098.r1.B04

样本信息如上，由于样本来自于各大文章，因此需要进行更详尽的实验设计以及数据整理

预过滤

虽然提前去掉low count基因不是运行DESeq进行差异表达分析必须的，但是提前过滤掉low count基因有两个好处：减少dds矩阵的大小，提高DESeq运行的速度。 这里我们演示如何去掉read count在10以下的基因。 注意：更严格的过滤会在DESeq的results功能中自动实现

```
keep <- rowSums(counts(dds)) >= 10
dds <- dds[keep,]
keep <- rowSums(counts(dds)) >= 10
dds <- dds[keep,]
```

关于因子的说明

默认情况下，R会安装字母顺序选择一个level作为参考level。就是说，如果没有给DEseq2函数指定要与哪个level进行比较（即指定哪个level是对照组），默认会安装字母顺序选择排在最前面的level作为对照组。

如果要指定哪个level是对照组，可以直接在函数#results#的参数contrast中指明对照level（下文中会具体介绍），或者直接在factor level中指明。

注意，更改design公式中变量的factor level只能在运行DESeq2分析前，分析中或分析后都不能再进行更改。更改dds矩阵中的factor level可以用以下两个方法：

```
#用factor
dds$condition <- factor(dds$condition, levels = c("untreated", "treated"))
#或者用relevel，直接指定参考level
dds$condition <- relevel(dds$condition, ref = "untreated")
```

OR 如果需要从dds中提取一个子集，比如说从dds中去掉部分样品，这时有可能会同时去掉所有样品的一个或者多个level。这种情况下，可以用droplevles函数去掉没有对应sample的level：

```
dds$condition <- droplevels(dds$condition)
```

折叠技术重复 DESeq2包中的collapseReplicates函数可以用于将技术重复的定量数据折叠成一个样品的。注意不能把这个函数用到生物学重复上，更多信息请查看collapseReplicates的帮助页面。

适用于bulk RNA-seq的normalization方法

比较流行的有：

```
DESeq的size factor (SF)
relative log expression(RLE)
upperquartile (UQ)
weighted trimmed mean of M-values(TMM)
```

这些适用于 bulk RNA-seq data 的normalization方法可能并不适合 single-cell RNA-seq data ，因为它们的基本假设是有问题的。 特意为single-cell RNA-seq data 开发的normalization方法

```
LSF (Lun Sum Factors)
scran package implements a variant on CPM specialized for single-cell data
```

而scater包把这些normalization方法都包装到了normaliseExprs函数里面，可以直接调用。

并且通过plotPCA函数来可视化这些normalization的好坏。