浙大植物学小白的转录组笔记

原创 2017-08-24 金建峰

写在前面：本实战笔记作者为浙大植物学专业的一位同学，纯实验背景能坚持学习就实属不易，还能认真写下这么多的总结更让我们感动。就问一直收藏从不动手的你此刻惭愧不惭愧。

转录组入门（1）：计算机及软件安装

作业要求

> 最好是有mac或者linux系统，8G+的内存，500G的存储即可。需要安装的软件包括sratoolkit,fastqc,hisats,samtools,htseq-count,R,Rstudio 来源于生信技能树：http://www.biotrainee.com/forum.php?mod=viewthread&tid=1750#lastpost

计算机资源的准备

需要Linux系统：只能选择Ubuntu 16.04 LTS，这个版本是长期支持的，而且是开源系统，并且有很好的GUI，很适合菜鸟入门的系统。 8G内存：没有钱换新的电脑，只能把手头2009年至今的y450改装升级一下。原本只有2G内存，我全部卸下，在某宝购买了两根DRR3 1600 16芯的4G的内存条，我的小y已经是极限了，升级到8G的内存。 存储500G：本来的小y只有320G的西数HDD 5200转的硬盘，读取速度和开机速度均不行，我就索性将HDD换成了120G的特科芯的SSD（也是在某宝购买的），然后在光驱位买了一个1T的希捷的5200转的HDD，容量问题解决了。 内存：因为要跑比较大的数据，我就索性将CPU也升级一下吧，当然还是在某宝购买的，我原来的是T4200，实在是吃不消了，我就度娘了一下，结果很多人都推荐换成P9600，性能提升很多，而且发热比T9600少很多，所以内存也差不多了，已经是极限了。显卡我是没有办法了，因为被焊死在主板上了。 到此计算机资源算是勉强可以了，好像真的是有点惨，说到底就是科研狗比较惨，缺钱，要不然，我早就买苹果电脑了，哪来那么多的事情呢。能够做到这种程度，还是挺佩服我自己的哈。

> 以下软件安装的内容，是参考简书作者hoptop的内容进行，在此进行说明。因为自己是完全新手入门，很多东西没有办法很快入门，作者的内容对我的帮助非常大，感谢。当然在这过程中，自己也是折腾了很久，有些地方采用了自己的方法，对于植物学的实验者来说，真是不容易呢。

软件的安装

在这之前，我们需要替换Ubuntu的镜像源，方法如下（参考链接）：

```
# 备份源列表文件并将默认镜像源改为清华镜像源
$ perl -pi.bak -e  's/cn.archive.ubuntu.com/mirrors.ustc.edu.cn/g' /etc/apt/source.list
$ perl -pi -e 's/http/https/g' /etc/apt/source.list
$ perl -pi -e 's/security.ubuntu.com/mirrors.ustc.edu.cn/g' /etc/apt/source.list
# 更新升级索引
$ sudo apt-get update
$ sudo apt-get upgrade
# 创建软件下载目录src和软件目录biosoft（也是从别人那里学来的，我也就这么常规的干了）
$ cd && mkdir src && mkdir biosoft
```

1.SRA Toolkit

官网：https://trace.ncbi.nlm.nih.gov/Traces/sra/sra.cgi?view=software

```
$ cd ~/src
# 选择适合自己系统的软件，这里选择的是Ubuntu版本。
$ wget https://ftp-trace.ncbi.nlm.nih.gov/sra/sdk/2.8.2-1/sratoolkit.2.8.2-1-ubuntu64.tar.gz
# 解压并将解压后的文件剪切到biosoft目录下
$ tar -zxvf  sratoolkit.2.8.2-1-ubuntu64.tar.gz && mv  sratoolkit.2.8.2-1-ubuntu64 ~/biosoft
# vim编辑器直接编辑~/.bashrc文件，将该软件加入环境变量中，可以全局运行，不用在运行的时候切换到当前目录
$ vim ~/.bashrc
# 在文件最后增加如下内容
PATH=$PATH:~/biosoft/sratoolkit.2.8.2-1-ubuntu64/bin
# 更新
$ source ~/.bashrc
# 尝试运行软件，出现帮助信息，就说明成功安装
$ fastq-dump -h
```

功能：能够将下载的SRA格式的测序结果转换成fastq格式，便于下一步的测序数据质控。 参考中文说明：http://blog.sina.com.cn/s/blog*8034ba040101e7ru.html 官方详细文档：https://trace.ncbi.nlm.nih.gov/Traces/sra/sra.cgi?view=toolkit*doc

2.Fastqc

官网：http://www.bioinformatics.babraham.ac.uk/projects/fastqc/ 因为fastqc运行需要Java环境，所以在安装之前需要检测一下Java环境

```
# 看是否安装了Java
$ java -version
# 若不存在，则进行安装，但是Java的版本要适合。我在装了Java9之后，fastqc没法正常运行，之后降到8版本之后，就能正常运行。
$ sudo apt-get openjdk-8-jdk

$ cd ~/src
# 下载二进制包，对自己Linux有信心的同志，可以下载源码包，自己编译
$ wget http://www.bioinformatics.babraham.ac.uk/projects/fastqc/fastqc_v0.11.5.zip
$ unzip fastqc_v0.11.5.zip && mv FastQC ~/biosoft
$ vim ~/.bashrc
# 加入环境变量
PATH=$PATH:~/biosoft/FastQC
$ source ~/.bashrc
# 测试软件，出现帮助信息
$ fastqc -h
```

功能：可视化测序结果质量的软件 中文教程：https://www.plob.org/article/5987.html

3.HISAT2

官网：http://ccb.jhu.edu/software/hisat2/index.shtml

```
$ cd ~/src
# 直接下载二进制包，免去自己进行编译安装
$ wget ftp://ftp.ccb.jhu.edu/pub/infphilo/hisat2/downloads/hisat2-2.1.0-Linux_x86_64.zip
$ unzip hisat2-2.1.0-Linux_x86_64.zip && mv hisat2-2.1.0 ~/biosoft
# 添加环境变量
$ vim ~/.bashrc
PATH=$PATH:~/biosoft/hisat2-2.1.0
$ source ~/.bashrc
# 测试软件
$ hisat2 -h
```

功能：将RNA-Seq的结果比对到基因组。 使用：http://bioinformatics.xtbg.ac.cn/hello-world-2/ 官方使用手册：http://ccb.jhu.edu/software/hisat2/manual.shtml

4.HTSeq

网站：http://samtools.sourceforge.net/

```
# 首先安装pip
$ sudo apt-get install python-pip
$ pip install HTSeq
# 直接安装完成，测试
$ python
>>> import HTSeq
# 如果没有出现报错信息，说明能够正常使用
```

功能：用来计数多种mapping软件输出文件reads 使用说明：http://www.dengfeilong.com/post/htseq-count.html

5.SAMtools

网站：http://samtools.sourceforge.net/

功能：生成存放高通量测序比对结果及其他转换格式，融合文件 参考网站：http://www.cnblogs.com/freemao/p/3763498.html

6.R

R：https://www.r-project.org/

功能：统计分析 使用手册：https://www.w3cschool.cn/r/

7.Rstudio

官网：https://www.rstudio.com/

```
7. # 桌面版本Ubuntu，使用Rstudio比较方便
8. $ cd ~/src
9. $ wget https://download1.rstudio.org/rstudio-1.0.143-amd64.deb
10. # 安装，也可以直接点击deb包，直接可以安装，不用命令行也方便
11. $ dpkg -i rstudio-1.0.143-amd64.deb
```

8.感想

这是第一次用简书的markdown来写笔记，而且是带有代码的，一次崭新的开始。因为这一篇是后面补上的，刚开始的时候没有特别察觉到做笔记这件事，后来觉得还是有必要的，因此我就开始补起来，而且因为不是边做实验边进行记录，所以代码还要自己重新敲，重新确认能否使用，真是折腾死了。痛不欲生啊，尤其是需要编译软件的时候，那是真的需要耐心，不然是真的会疯掉的，总是会出错，一个接着一个的出错。不管怎么样，第一步算是成功的迈开了，接下来还是需要记录一下代码，及时整理。

转录组入门（2）：读文章获取测序数据

作业要求

> 本系列课程学习的文章是：AKAP95 regulates splicing through scaffolding RNAs and RNA processing factors. Nat Commun 2016 Nov 8;7:13347. PMID: 27824034 很容易在文章里面找到数据地址GSE81916 这样就可以下载sra文件作业，看文章里的methods部分，把它用到的软件和参数摘抄下来，然后理解GEO/SRA数据库的数据存放形式，把规律和笔记发在论坛上面！ 来源于生信技能树：http://www.biotrainee.com/forum.php?mod=viewthread&tid=1750#lastpost

实验过程

1.文献下载

一般我都会去Google镜像搜索：https://xueshu.glgoo.net/.此外还会在SCI-HUB下载，不过前段时间被起诉了，还罚款，不知道这个牛逼的网站能够撑到什么时候。在实验方法的部分GSE81916存放了测序数据。

2.数据下载

进入NCBI的GEO数据库https://www.ncbi.nlm.nih.gov/geo/，搜索GSE81916。 看到页面中的**overall design**：

实验设计

所以我们只需要下载样本9-15数据。 **数据储存的链接：**

数据存放的位置

通过ftp的方式下载数据，其中还介绍了测序平台。 **点击进入ftp,看到储存的所有测序文件：**

一共15数据，我们只需要下载9-15的数据

下面我们只要执行一个循环，可以自动下载SRR3589956-SRR3589962一共七个数据，命令如下：

```
6. # 我是将数据放在disk2/sra目录下
7. $ cd ~/disk2 && mkdir sra
8. $ for ((i=56;i<=62;i=i++));do wget ftp://ftp-trace.ncbi.nlm.nih.gov/sra/sra-instant/reads/ByStudy/sra/SRP/SRP075/SRP075747/SRR35899$i/SRR35899$i.sra ;done
9. # 只需等待，我当时就是花了一个晚上的时间，当然这里可以使用sratoolkit自带的‘prefetch accession’的形式来下载数据，并且默认下载到~/ncbi/public/sra 。
```

3.文章使用的软件工具

read 计数：HTSeq 差异基因表达分析：DESeq 差异外显子表达分析：DEXSeq 统计分析：R 基因富集分析：DAVID

转录组入门（3）：了解fastq测序数据

作业要求

> 需要用安装好的sratoolkit把sra文件转换为fastq格式的测序文件，并且用fastqc软件测试测序文件的质量！ 作业，理解测序reads，GC含量，质量值，接头，index，fastqc的全部报告，搜索中文教程，并发在论坛上面。 来源于生信技能树：http://www.biotrainee.com/forum.php?mod=viewthread&tid=1750#lastpost

---

实验过程

1.fastq-dump将sra数据转换成fastq格式

```
4. # 需要将作业2（http://www.jianshu.com/p/da377252ee96）中下载的测序数据用工具sratoolkit转换成fastq的格式。
5. $ for ((i=56;i<=62;i++));do fastq-dump --gzip --split-3 -A ~/disk2/sra/SRR35899$i.sra -O ~/disk2/data/rna-seq;done
```

fastq-dump 用法：

--gzip 使得输出的结果是*.gz 的格式 --split-3 对于PE测序，输出的结果是两个*_1.fastq.gz -A| --accession 输入你的sra文件可以是绝对路径，我的数据来源是~disk2/sra/SRR35899$i.sra （ **如果你直接写accession，那么fastq-dump会默认重新下载数据，并且会放在~/ncbi/public/sra目录下**） -O 是设置输出的目录

fastq格式：

Fastq格式是一种基于文本的存储生物序列和对应碱基（或氨基酸）质量的文件格式。最初由桑格研究所（Wellcome Trust Sanger Institute）开发出来，现已成为存储高通量测序数据的事实标准。

fastq数据格式

每条read由4行字符构成： 第一行：必须以@开头，后面跟着序列的唯一ID以及相关说明内容。 第二行：核酸序列，是有ATCGN字符组成。 第三行：“+”开头，内容和第一行@后面的一样。 第四行：每个测序碱基质量，是用ASCII码来表示的，与第二行的字符数一致。 碱基质量得分与错误率的换算关系： **Q = -10log10p（p表示测序的错误率，Q表示碱基质量分数）** ASCII值与碱基质量得分之间的关系： **Phred64 Q=ASCII转换后的数值-64 Phred33 Q=ASCII转换后的数值-33**

> 如何判断是Phred64 还是 Phred33 ？ ASCII值小于等于58（相应的质量得分小于等于25）对应的字符只有在Phred+33的编码中被使用，所有Phred+64所使用的字符的ASCII值都大于等于59。在通常情况下，ASCII值大于等于74的字符只出现在Phred+64中。如果是最近两年的测序数据，一般都是Phred33形式的。参考文章：http://blog.csdn.net/huyongfeijoe/article/details/51613827

2.Fastqc 进行测序结果的质控

用法： fastqc [-o output dir] [--(no)extract] [-f fastq|bam|sam] [-c contaminant file] seqfile1 .. seqfileN 参数： -o 输出目录，需自己创建目录 --(no)extract 是否解压输出文件，默认是自动解压缩zip文件。加上--noextract不解压文件。 -f 指定输入文件的类型，支持fastq|bam|sam三种格式的文件，默认自动识别。 -t 同时处理的文件数目。 -c 是contaminant 文件，会从中搜索overpresent 序列。

```
6. $ mkdir -p ~/disk2/data/QC
7. $ cd ~/disk2/data/rna-seq
8. # 将所有的数据进行质控，得到zip的压缩文件和html文件
9. $ fastqc -o ~/disk2/data/QC *.fastq.gz
```

质控结果文件

3.质控结果查看

质控结果有14个html文件，你可以选择用浏览器打开查看最终的QC reports。

- 首先来大概看一下QC结果报告。

QC可视化结果——双击html文件，在浏览器中直接打开

- 左边是目录概要，可以点击想要看的结果，右边会跳转到特定详细的可视化结果。绿色代表“通过”，黄色代表“警告”，红色代表“不通过，失败”。

Summary

- Basic Statistics，基本的数据统计包括文件名，文件类型，编码形式，总的序列数，质量差的序列，序列平均长度，GC含量。

基本数据统计

- Per base sequence quality，每个read各位置碱基的测序质量。横轴碱基的位置，纵轴是质量分数，Quality score=-

10log10p（p代表错误率），所以当质量分数为40的时候，p就是0.0001，质量算高了。红色线代表中位数，蓝色代表平均数，黄色是25%-75%区间，触须是10%-90%区间（黄色和触须我不是特别明白）。**若任一位置的下四分位数低于10或者中位数低于25，出现“警告”；若任一位置的下四分位数低于5或者中位数低于20，出现“失败，Fail”**。

各位置碱基质量

- Per tile sequence quality，检查reads中每一个碱基位置在不同的测序小孔之间的偏离度，蓝色代表偏离度小，质量好，越红代表偏离度越大，质量越差。

偏离度

- Per sequence quality scores，reads质量的分布，**当峰值小于27时，警告；当峰值小于20时，fail**。我的报告峰值在38。

reads质量分布

- Per base sequence content，对所有reads的每一个位置，统计ATCG四种碱基的分布，横轴为位置，纵轴为碱基含量，正常情况下每个位置每种碱基出现的概率是相近的，四条线应该平行且相近。**当部分位置碱基的比例出现bias时，即四条线在某些位置纷乱交织，往往提示我们有overrepresented sequence的污染**。本结果前10个位置，每种碱基频率有明显的差别，说明有污染。**当任一位置的A/T比例与G/C比例相差超过10%，报"WARN"；当任一位置的A/T比例与G/C比例相差超过20%，报"FAIL"**。

碱基分布

- Per Sequence GC Content，统计reads的平均GC含量的分布。红线是实际情况，蓝线是理论分布（正态分布，均值不一定在50%，而是由平均GC含量推断的）。 曲线形状的偏差往往是由于文库的污染或是部分reads构成的子集有偏差（overrepresented reads）。形状接近正态但偏离理论分布的情况提示我们可能有系统偏差。**偏离理论分布的reads超过15%时，报"WARN"；偏离理论分布的reads超过30%时，报"FAIL"**。

reads 平均GC含量分布

- Per base N content，当测序仪器不能辨别某条reads的某个位置到底是什么碱基时，就会产生“N”，统计N的比率。正常情况下，N值非常小。**当任意位置的N的比例超过5%，报"WARN"；当任意位置的N的比例超过20%，报"FAIL"**。

各位置N的reads比率

- Sequence Length Distribution，reads长度分布，**当reads长度不一致时报"WARN"；当有长度为0的read时报“FAIL”**。

reads 长度分布

- Sequence Duplication Levels，统计不同拷贝数的reads的频率。**测序深度越高，越容易产生一定程度的duplication，这是正常的现象，但如果duplication的程度很高，就提示我们可能有bias的存在**。横坐标是duplication的次数，纵坐标是duplicated reads的数目，以unique reads的总数作为100%。下图中，大于10个重复的reads占总序列的20%以上，其他依次类推。**当非unique的reads占总数的比例大于20%时，报"WARN"；当非unique的reads占总数的比例大于50%时，报"FAIL“**。

统计不同拷贝数的reads的频率

- Overrepresented sequences，一条序列的重复数，因为一个转录组中有非常多的转录本，一条序列再怎么多也不太会占整个转录组的一小部分（比如1%），如果出现这种情况，不是这种转录本巨量表达，就是样品被污染。这个模块列出来大于全部转录组1%的reads序列，但是因为用的是前200,000条，所以其实参考意义不大，完全可以忽略。

一条序列的重复数

- Adapter content，接头含量

接头含量

- Kmer content

Kmer含量

参考资料：https://www.plob.org/article/5987.html；http://blog.sina.com.cn/s/blog_1319a10ee0102vfbx.html。

4.质控结果批量查看神器——MultiQC

知乎青山屋主写的为知笔记介绍了multiQC软件--批量显示QC结果。

```
# 利用Anaconda来安装MultiQC非常方便，首先得安装Anaconda，用清华源下载，特别快，而官网实则难以接受。
# 清华源地址：https://mirrors.tuna.tsinghua.edu.cn/help/anaconda/
# 官网：https://www.continuum.io/downloads/
$ cd ~/src
$ wget  https://mirrors.tuna.tsinghua.edu.cn/anaconda/archive/
# 下载到的是shell脚本文件，直接运行，安装完成
$ bash Anaconda2-4.4.0-Linux-x86_64.sh
# 添加 Anaconda Python 免费仓库
$ conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/free/
$ conda config --set show_channel_urls yes
# 然后直接安装MultiQC
$ conda install -c bioconda multiqc
# 测试
$ multiqc --help
# 进入存放QC结果的文件夹，并执行multiqc
$ cd ~/disk2/data/QC
# 扫描结果文件，忽略html文件
$ multiqc /data/*fastqc.zip --ignore *.html
# 最后会默认生成一个名为multiqc_report.html文件，用浏览器查看，具体看青山屋主的介绍。
```

参考资料：https://mirrors.tuna.tsinghua.edu.cn/help/anaconda/ https://www.continuum.io/downloads/ http://fbb84b26.wiz03.com/share/s/3XK4IC0cm4CL22pU-r1HPcQQ1iRTvV2GwkwL2AaxYi2fXHP7

转录组入门（4）：了解参考基因组及基因注释

作业要求

在UCSC下载hg19参考基因组，群主博客有详细说明，从gencode数据库下载基因注释文件，并且用IGV去查看你感兴趣的基因的结构，例如TP53，EGFR等等。 截图几个基因的IGV可视化结构！还可以下载ENSEMBL，NCBI的GTF，也导入IGV看看，截图基因结构。了解IGV常识。 来源于生信技能树：http://www.biotrainee.com/forum.php?mod=viewthread&tid=1750#lastpost

实验过程

1.参考基因组hg19下载

Jimmy大神的博客：生信菜鸟团，详细介绍了各种基因组版本的对应关系以及下载方法。

```
# 下载USCS版本的hg19
$ mkdir -p ~/disk2/data/reference/genome
$ cd ~/disk2/data/reference/genome
$ nohup wget http://hgdownload.soe.ucsc.edu/goldenPath/hg19/bigZips/chromFa.tar.gz &
# 解压，得到所有染色体的信息
$ tar -zxvf chromFa.tar.gz
# 将所有的染色体信息整合在一起，重定向写入hg19.fa文件，得到参考基因组
$ cat *.fa > hg19.fa
# 将多余的染色体信息文件删除，节省空间
$ rm -rf chr*
```

- **查找基因组的过程介绍及USCS页面简单介绍：**1.打开USCS官网：http://genome.ucsc.edu/，导航栏：我们主要利用Downloads第五个标签，其下还有很多工具。

图1 UCSC官网首页信息

2.Downloads 标签，选择下拉菜单Genome data，进入图三页面,第一个大类是脊椎动物，人类自然是包括在内的。

图2 Genome data

图3 Genome data点击后的页面内容

3.下载人类Human的基因组，点击human，进入基因组页面，点击所有数据集（full data set），看到很多文件，我们要下载的就是chromFa.tar.gz(The assembly sequence in one file per chromosome.)。

图4 基因版本，hg19

图5 hg19版本所有数据列表

2.参考基因组注释下载

进入人和小鼠基因组注释信息官网GENCODE，选择data->human->GRCh37-mapped Releases，下载最新第26版本的hg19人类基因组注释信息。点击进入下载页面，将GTF和GFF3全部下载，解压。

图6 hg19版本基因组注释信息

图7 基因组注释信息版本GFF和GTF

GTF和GFF之间的区别：

数据结构：都是由9列构成，分别是reference sequence name; annotation source; feature type; start coordinate; end coordinate; score; strand; frame; attributes.前8列都是相同的，第9列不同。 **GFF第9列**：都是以键值对的形式，键值之间用“=”连接，不同属性之间用“；”分隔，都是以ID这个属性开始。下图中有两个ID，说明是不同的序列。

图8 gff第9列格式

GTF第9列：同样以键值对的形式，键值之间是以空格区分，值用双引号括起来；不同属性之间用“；”分隔；开头必须是gene*id, transcipt*id两个属性。

图9 gtf第9列格式

参考资料：http://www.jianshu.com/p/48b5a0972301 http://www.gencodegenes.org/faq.html

```
8. # gtf及gff3下载代码
9. $ mkdir -p ~/disk/data/reference/genome/hg19
10. $ cd ~/disk2/data/reference/genome/hg19
11. $ wget ftp://ftp.sanger.ac.uk/pub/gencode/Gencode_human/release_26/GRCh37_mapping/gencode.v26lift37.annotation.gtf.gz
12. $ wget ftp://ftp.sanger.ac.uk/pub/gencode/Gencode_human/release_26/GRCh37_mapping/gencode.v26lift37.annotation.gff3.gz
13. $ gunzip *.gz && rm -rf *.gz
```

3.IGV下载及使用

Integrative Genomics Viewer(IGV)是一种探索大型综合基因组数据的高性能交互式可视化工具。它支持各种各样的数据类型，包括基于芯片测序、二代测序数据和基因组注释数据等。

- IGV下载

```
12. # 进入IGV官网，并下载相应的软件包，有Windows，Mac，和LINUX，这里我下载Linux二进制包
13. $ cd ~/src
14. $ wget http://data.broadinstitute.org/igv/projects/downloads/IGV_2.3.97.zip
15. $ unzip IGV_2.3.97.zip && mv IGV_2.3.97 ~/biosoft
16. # 添加环境变量
17. $ vim ~/.bashrc
18. PATH=$PATH:~/biosoft/IGV_2.3.97
19. $ source ~/.bashrc
20. # 运行IGV，Linux直接运行igv.sh可以开启IGV窗口，但是会比较慢，要耐心等待。
21. $ igv.sh
```

- IGV使用

初始化窗口

1.载入基因组，选择Genome标签，load我们之前已经下载好的hg19.fa基因组。 2.载入基因组注释，但是在载入之前需要将gff3进行排序，选择Tools-Run igvtools，进入以下igvtools窗口：

igvtools窗口

3.获得sorted文件：command选择sort，再选择输入的注释文件，点击Run，就可以生成sorted.gff3文件。 4.通过file->load from file...选择sorted文件，打开。选择区域的大小，来看某些基因的信息，蓝色的粗线条就是代表基因。说到底，IGV就是一个将基因组及其注释信息可视化的工具。

载入基因组和注释信息后的窗口

参考资料： http://blog.sina.com.cn/s/blog_165caa4fd0102wh0n.html
http://www.cnblogs.com/leezx/p/5603481.html

转录组入门（5）：序列比对

作业要求

6. 比对软件很多，首先大家去收集一下，因为我们是带大家入门，请统一用hisat2，并且搞懂它的用法。
7. 直接去hisat2的主页下载index文件即可，然后把fastq格式的reads比对上去得到sam文件。
8. 接着用samtools把它转为bam文件，并且排序(注意N和P两种排序区别)索引好，载入IGV，再截图几个基因看看！
9. 顺便对bam文件进行简单QC，参考直播我的基因组系列。 来源于生信技能树：
http://www.biotrainee.com/forum.php?mod=viewthread&tid=1750#lastpost

实验过程

1. 比对软件

- HISAT2：http://ccb.jhu.edu/software/hisat2/index.shtml 参考资料：http://blog.biochen.com/archives/337
- STAR：https://codeload.github.com/alexdobin/STAR/zip/master 参考资料：http://www.bio-info-trainee.com/727.html
- TopHat：http://ccb.jhu.edu/software/tophat/index.shtml 参考资料：
http://blog.sina.com.cn/s/blog_8808cae20101amqp.html
- RapMap：https://github.com/COMBINE-lab/RapMap 参考文献：
https://academic.oup.com/bioinformatics/article/32/12/i192/2288985/RapMap-a-rapid-sensitive-and-accurate-tool-for
- CIDANE：http://ccb.jhu.edu/software/cidane/ 参考文献：
https://genomebiology.biomedcentral.com/articles/10.1186/s13059-015-0865-0
- CLASS2 ：https://sourceforge.net/projects/splicebox/files/?source=navbar 参考文献：
https://academic.oup.com/nar/article/44/10/e98/2516329/CLASS2-accurate-and-efficient-splice-variant

内容主要参考：http://www.360doc.com/content/16/1223/13/29483982_617058719.shtml

2. HISAT2的使用

人类和小鼠的索引有现成的，HISAT2官网可以直接下载进行序列比对。如下图所示：选择hg19和mm10的index，文章中RNA-Seq测序数据，可以包括人类的3个数据和小鼠的4个数据，因此需要小鼠和人类的索引。

index下载链接

- **（1）人类和小鼠index下载**

```
$ mkdir -p ~/disk2/data/reference/index
$ cd ~/disk2/data/reference/index
$ wget ftp://ftp.ccb.jhu.edu/pub/infphilo/hisat2/data/hg19.tar.gz
$ wget ftp://ftp.ccb.jhu.edu/pub/infphilo/hisat2/data/mm10.tar.gz
# 解压得到两个目录，hg19和mm10
$ tar -zxvf *.tar.gz
# 删除压缩包
$ rm -rf *.tar.gz
```

有时候没有现成的index，我们就需要自己用HISAT2重新构建索引；包括外显子、剪切位点及SNP索引的建立。
参考网站：http://blog.biochen.com/archives/337

- **（2）比对序列，得到sam文件**

Usage：hisat2 [options]* -x {-1 -2 | -U | --sra-acc } [-S]
参数： -x 指定index文件 -1 双端测序第一个文件 -2 双端测序第二个文件 -U 单端测序文件 --sra-acc SRA accession number -S 指定输出的格式，一般指定为sam

```
8. # 小鼠和人是分开各自比对自己的index
9. # 人的比对
10. $ for ((i=56;i<=58;i++));do hisat2 -t -x ~/disk2/data/reference/index/hg19/genome -1 ~/disk2/data/rna-seq/SRR35899${i}_1.fastq.gz -2 ~/disk2/data/rna-seq/SRR35899${i}_2.fastq.gz -S SRR35899${i}.sam;done
11. # 小鼠比对
12. $ for ((i=59;i<=62;i++));do hisat2 -t -x ~/disk2/data/reference/index/mm10/genome -1 ~/disk2/data/rna-seq/SRR35899${i}_1.fastq.gz -2 ~/disk2/data/rna-seq/SRR35899${i}_2.fastq.gz -S SRR35899${i}.sam;done
13. #结果一共得到7个sam文件
```

比对结束的标准输出

比对结束输出

3.SAMtools 的使用

samtools功能众多，在本次作业中，我们主要学会将sam文件转换为bam文件，并对bam文件进行sorted（其中有两种排序方式N和P），最后建立索引。

sort过程中，会出现很多tmp文件，最后会融合所有的tmp文件，形成一个sorted文件。如图所示：

tmp文件

融合所有tmp文件

samtools还有其他功能，包括flagstat命令（查看比对的大致情况）；mileup命令用于生成bcf文件，再使用bcftools进行SNP和Indel的分析；faidx命令对fasta文件建立索引,生成的索引文件以.fai后缀结尾等。

比对大致情况

参考资料：http://blog.csdn.net/sinat_38163598/article/details/72910115

4.比对结果QC

QC的软件有很多，下面3种工具都可以用来质控比对结果（参考了hoptop的简书文章），这里我们使用RSeQC来对我们的比对结果进行质控。

必须吐槽度娘，搜出来的都是什么东西，完全不能看，一脸懵逼。

- RSeQC——http://rseqc.sourceforge.net/
- Qualimap——http://qualimap.bioinfo.cipf.es/
- Picard——http://broadinstitute.github.io/picard/

```
7. # RSeQC的安装，需要先安装gcc；numpy；R；Python2.7，这里比较难装的就是numpy—可以直接利用anaconda安装（http://www.jianshu.com/p/14fd4de54402）
8. # 我的环境已经配置好了，所以直接可以用pip命令安装
9. $ pip install RSeQC
10. # 对bam文件进行质控，其余都同样的进行
11. $ bam_stat.py  -i SRR3589956_sorted.bam
```

RSeQC包括了十多个Python脚本，实现很多功能，具体每个脚本的参数用法，都可以在官网学习这里暂时不多说（之后有时间再详细介绍）。

5.IGV查看比对结果

载入参考基因组，基因组注释文件，很bam文件，看一些基因。

选了一条染色体随便一看

真正的数据分析开始，果然还是很吃力，不过依旧要加油，马上就要学会了。

转录组入门（6）：reads计数

作业要求

1.实现这个功能的软件也很多，还是烦请大家先自己搜索几个教程，入门请统一用htseq-count，对每个样本都会输出一个表达量文件。 2.需要用脚本合并所有的样本为表达矩阵。参考：生信编程直播第四题：多个同样的行列

式文件合并起来。 3.对这个表达矩阵可以自己简单在excel或者R里面摸索，求平均值，方差。 看看一些生物学意义特殊的基因表现如何，比如GAPDH,β-ACTIN等等。 来源于生信技能树：http://www.biotrainee.com/forum.php?mod=viewthread&tid=1750#lastpost

实验过程

1.数据准备

文献中测序的是PE，因此我们在对双端数据进行处理时，必须要按reads名进行排序。

```
7. # 首先将bam文件按reads名称进行排序（前期是按照默认的染色体位置进行排序的，所以要重新进行排序），这里我们主要以小鼠的数据为例子，不进行人类的测序数据。
8. $ cd ~/disk2/data/rna-seq/aligned
9. $ for ((i=59;i<=62;i++));do samtools sort -n SRR35899${i}.bam -o SRR35899${i}_nsorted.bam;done
10. # 将排序后的bam文件再次转换成sam格式的文件
11. $ for ((i=59;i<=62;i++));do samtools view -h SRR35899${i}_nsorted.bam > SRR35899${i}_count.sam;done
```

2. reads计数

数据准备已经完成，接下来要使用htseq-count对数据进行reads 计数。 Usage:htseq-count [options] 这里最好使用ensembl的基因组注释文件，小鼠的文件还是需要再次的下载。

```
10. #小鼠基因组注释文件的下载,我下载的是m10版本的，与基因组匹配
11. $ mkdir -p ~/disk2/data/reference/genome/mm10
12. $ cd ~/disk2/data/reference/genome/mm10
13. $ wget ftp://ftp.sanger.ac.uk/pub/gencode/Gencode_mouse/release_M10/gencode.vM10.chr_patch_hapl_scaff.annotation.gtf.gz
14. $ gunzip gencode.vM10.chr_patch_hapl_scaff.annotation.gtf.gz && rm -rf gencode.vM10.chr_patch_hapl_scaff.annotation.gtf.gz
15. # 之前的环境已经全部搭建完成，直接就可以使用htseq-count
16. $ mkdir -p ~/disk2/data/rna-seq/matrix && cd ~/disk2/data/rna-seq/matrix
17. $ for ((i=59;i<=62;i++));do htseq-count ~/disk2/data/rna-seq/aligned/SRR35899${i}_count.sam
~/disk2/data/reference/genome/mm10/gencode.vM10.chr_patch_hapl_scaff.annotation.gtf;done
```

最后得到4个矩阵文件

reads 计数文件

count文件的格式:基因名一列+reads计数一列

count文件结构

这一部分主要参考的是老司机Jimmy大神的博客：http://www.bio-info-trainee.com/244.html

3.合并表达矩阵

使用R将这四个文件进行合并，得到最后的融合表达矩阵，R语言代码：

```
22. >options(stringsAsFactors = FALSE)
23. # 首先将四个文件分别赋值：control1，control2，rep1，rep2
24. >control1 <- read.table("~/disk2/data/rna-seq/matrix/SRR3589959.count", sep="\t", col.names = c("gene_id","control1"))
25. >control2 <- read.table("~/disk2/data/rna-seq/matrix/SRR3589961.count", sep="\t", col.names = c("gene_id","control2"))
26. >rep1 <- read.table("~/disk2/data/rna-seq/matrix/SRR3589960.count", sep="\t", col.names = c("gene_id","akap951"))
27. >rep2 <- read.table("~/disk2/data/rna-seq/matrix/SRR3589962.count", sep="\t",col.names = c("gene_id","akap952"))
28. # 将四个矩阵按照gene_id进行合并，并赋值给raw_count
29. >raw_count <- merge(merge(control1, contol2, by="gene_id"), merge(rep1,rep2, by="gene_id"))
30. # 需要将合并的raw_count进行过滤处理，里面有5行需要删除的行，在我们的小鼠的表达矩阵里面，是1,2,48823,48824,48825这5行。并重新赋值给raw_count_filter
31. >raw_count_filt <- raw_count[-48823:-48825, ]
32. >raw_count_filter <- raw_count_filt[-1:-2, ]
33. # 因为我们无法在EBI数据库上直接搜索找到ENSMUSG00000024045.5这样的基因，只能是ENSMUSG00000024045的整数，没有小数点，所以需要进一步替换为整数的形式。
34. # 第一步将匹配到的.以及后面的数字连续匹配并替换为空，并赋值给ENSEMBL
35. >ENSEMBL <- gsub("\\.\\d*", "", raw_count_filt1$gene_id)
36. # 将ENSEMBL重新添加到raw_count_filt1矩阵
37. >row.names(raw_count_filter) <- ENSEMBL
38. # 看一些基因的表达情况，在UniProt数据库找到AKAP95的id，并从矩阵中找到访问，并赋值给AKAP95变量
39. >AKAP95 <- raw_count_filter[rownames(raw_count_filt1)=="ENSMUSG00000024045",]
40. # 查看AKAP95
41. >AKAP95
```

- raw*count*filter结构

raw_count_filter数据结构

- AKAP95基因的表达情况，可以看到表达量是提高

AKAP95reads计数情况

这一部分主要参考徐州更同学的R语言代码：http://www.jianshu.com/p/e9742bbf83b9 我的数据是用的小鼠的测序结果，所以我修改了部分的内容，更好的进行。

PS：前段时间一直在忙于实验，没有及时去完成这一部分的内容，今天正好是星期天，所以就打算抽出一点时间来完成这一部分的学习笔记。还有一个原因就是因为自己好像不会了，没法再进行下去了，说到第还是有些害怕了，实在惭愧哈，继续努力加油。

转录组入门（7）：差异基因分析

作业要求

这个步骤推荐在R里面做，载入表达矩阵，然后设置好分组信息，统一用DEseq2进行差异分析，当然也可以走走edgeR或者limma的voom流程。 基本任务是得到差异分析结果，进阶任务是比较多个差异分析结果的异同点。 来源于生信技能树：http://www.biotrainee.com/forum.php?mod=viewthread&tid=1750#lastpost

实验过程

1.读取自己表达矩阵

```
13. # 构建自己的表达矩阵并读取
14. > control1 <- read.table("~/disk2/data/rna-seq/matrix/SRR3589959.count", sep="\t", col.names = c("gene_id","control1"))
15. > control2 <- read.table("~/disk2/data/rna-seq/matrix/SRR3589961.count", sep="\t", col.names = c("gene_id","control2"))
16. > rep1 <- read.table("~/disk2/data/rna-seq/matrix/SRR3589960.count", sep="\t", col.names = c("gene_id","akap951"))
17. > rep2 <- read.table("~/disk2/data/rna-seq/matrix/SRR3589962.count", sep="\t",col.names = c("gene_id","akap952"))
18. > raw_count <- merge(merge(control1, control2,by="gene_id"),merge(rep1,rep2, by="gene_id"))
19. > raw_count_filt <- raw_count[-48823:-48825,]
20. > raw_count_filter <- raw_count_filt[-1:-2,]
21. > ENSEMBL <- gsub("\\.\\d*", "", raw_count_filter$gene_id)
22. > row.names(raw_count_filter) <- ENSEMBL
23. > raw_count_filter <- raw_count_filter[ ,-1]
```


矩阵数据结构

2.构建dds对象

```
10. # 这一步很关键，要明白condition这里是因子，不是样本名称；小鼠数据有对照组和处理组，各两个重复
11. > condition <- factor(c(rep("control",2),rep("akap95",2)), levels = c("control","akap95"))
12. # 获取count数据
13. > countData <- raw_count_filter[,1:4]
14. > colData <- data.frame(row.names=colnames(raw_count_filter), condition)
15. > dds <- DESeqDataSetFromMatrix(countData, colData, design= ~ condition)
16. # 查看一下dds的内容
17. > head(dds)
```


adds概要信息

3.DESeq标准化dds

```
10. # normalize 数据
11. > dds2 <- DESeq(dds)
12. # 查看结果的名称，本次实验中是 "Intercept", "condition_akap95_vs_control"
13. > resultsNames(dds2)
14. # 将结果用results()函数来获取，赋值给res变量
15. res <- results(dds2)
16. # summary一下，看一下结果的概要信息
17. summary(res)
```

- result结果可以看到一些基本的信息，p值默认小于0.1，上调基因有625个，下调基因有445个。

res的概要信息

4.提取差异分析结果

```
# 获取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的差异表达基因。
> table(res$padj<0.05)
> res <- res[order(res$padj),]
> diff_gene_deseq2 <-subset(res,padj < 0.05 & (log2FoldChange > 1 | log2FoldChange < -1))
> diff_gene_deseq2 <- row.names(diff_gene_deseq2)
> resdata <-  merge(as.data.frame(res),as.data.frame(counts(dds2,normalize=TRUE)),by="row.names",sort=FALSE)
# 得到csv格式的差异表达分析结果
> write.csv(resdata,file= "control_vs_akap95.cvs",row.names = F)
```

resdata数据结构

有大神们的帮助，总算是完成了这一课的内容，感谢黯蓝小伙伴的帮助，还有群里小伙伴的帮助，当然还参考了Jimmy大神的博客，此外还参考了张翼翔的贴文：http://www.biotrainee.com/thread-1984-1-2.html。继续下一课的内容，最后一课的内容，GO富集分析。

转录组入门（8）：差异基因结果注释

作业要求

我们统一选择p<0.05而且abs(log2FC)大于1的基因为显著差异表达基因集，对这个基因集用R包做KEGG/GO超几何分布检验分析。 然后把表达矩阵和分组信息分别作出cls和gct文件，导入到GSEA软件分析。 来源于生信技能树：http://www.biotrainee.com/forum.php?mod=viewthread&tid=1750#lastpost

实验过程

1.差异基因筛选

我在转录组入门（7）：差异基因分析已经完成了差异基因筛选，为了更好的衔接，我将上一步的代码也一并写入，完整流畅一些，最后我们得到的是数据diff*gene*deseq2，包含了差异表达基因。(这里就不在详细注释这些代码，请看上一篇文章)

2.GO/KEGG分析及GSEA

我们主要用到的就是Y叔的R包：clusterProfiler包，github上有详细的说明，这个包的功能很强大，我小白一个真的是整不明白，大致看了一些，不过还是有学习到很多，下面就开始贴代码。

2.1 安装clusterProfiler

安装clusterProfiler以及依赖的包，因为个人的电脑都是有差别的，所以我也不好说，这样的代码就一定适合你，因为在我参考别人的时候，就是出现了很多问题，没法安装和载入这个包。具体问题还是要具体分析，也不要那么容易放弃，稍微折腾一些，说不定就能解决。

2.2 安装构建自己的基因组注释数据

Biocouductor官网已经拥有了构建好的常用的19个注释数据库，包括了小鼠，人类和拟南芥等常用注释数据，可以用安装bioconductor包的方法来直接安装和载入注释数据，直接使用。

org.Ag.eg.db	Bioconductor Package Maintainer	Genome wide annotation for Anopheles
org.At.tair.db	Bioconductor Package Maintainer	Genome wide annotation for Arabidopsis
org.Bt.eg.db	Bioconductor Package Maintainer	Genome wide annotation for Bovine
org.Ce.eg.db	Bioconductor Package Maintainer	Genome wide annotation for Worm
org.Cf.eg.db	Bioconductor Package Maintainer	Genome wide annotation for Canine
org.Dm.eg.db	Bioconductor Package Maintainer	Genome wide annotation for Fly
org.Dr.eg.db	Bioconductor Package Maintainer	Genome wide annotation for Zebrafish
org.EcK12.eg.db	Bioconductor Package Maintainer	Genome wide annotation for E coli strain K12
org.EcSakai.eg.db	Bioconductor Package Maintainer	Genome wide annotation for E coli strain Sakai
org.Gg.eg.db	Bioconductor Package Maintainer	Genome wide annotation for Chicken
org.Hs.eg.db	Bioconductor Package Maintainer	Genome wide annotation for Human
org.Mm.eg.db	Bioconductor Package Maintainer	Genome wide annotation for Mouse
org.Mmu.eg.db	Bioconductor Package Maintainer	Genome wide annotation for Rhesus
org.Pf.plasmo.db	Bioconductor Package Maintainer	Genome wide annotation for Malaria
org.Pt.eg.db	Bioconductor Package Maintainer	Genome wide annotation for Chimp
org.Rn.eg.db	Bioconductor Package Maintainer	Genome wide annotation for Rat
org.Sc.sgd.db	Bioconductor Package Maintainer	Genome wide annotation for Yeast
org.Ss.eg.db	Bioconductor Package Maintainer	Genome wide annotation for Pig
org.Xl.eg.db	Bioconductor Package Maintainer	Genome wide annotation for Xenopus

19个注释数据库

```
9. # 我们是小鼠数据，所以直接安装载入就可以了，当然人类的也是一样。
10. # 人类的注释数据
11. biocLite("org.Hs.eg.db")
12. library(org.Hs.eg.db)
13. # 小鼠的注释数据
14. biocLite("org.Mm.eg.db")
15. library(org.Mm.eg.db)
```

- 如果没有包括在这些注释数据库里面，那么就需要使用**AnnotationHub**这个包来构建自己的**OrgDb**。代码如下：

2.3 GO（Gene Ontology）分析

这里涉及到多种类型的ID转换，我们常见的ENSEMBL，ENTREZID这两大类，这里我在分析的时候发现，ENTREZID=kegg=ncbi-geneid，这三者有事相同的ID号。jimmy博客有详细的介绍，我进行了适当的参考：http://www.bio-info-trainee.com/710.html。

- **ID转换函数介绍**

- **enrichGO()函数进行GO分析及画图**

主要函数及参数：enrichGO(gene, OrgDb, keytype = "ENTREZID", ont = "MF", pvalueCutoff = 0.05, pAdjustMethod = "BH", universe, qvalueCutoff = 0.2,minGSSize = 10, maxGSSize = 500, readable = FALSE, pool = FALSE) **gene:**差异基因ID；**ont:**主要的分为三种，三个层面来阐述基因功能，生物学过程（BP），细胞组分（CC），分子功能（MF）；**OrgDb:**指定物种注释数据；**keytype:**ID类型；**pAdjustMethod:**P值校正方法。

```
# 进行go分析
ego <- enrichGO(
  gene = row.names(diff_gene_deseq2),
  OrgDb = org.Mm.eg.db,
  keytype = "ENSEMBL",
  ont = "MF"
)
# 气泡图
dotplot(ego, font.size=5)
# 网络图
enrichMap(ego, vertex.label.cex=1.2, layout=igraph::layout.kamada.kawai)
# GO图需要安装额外的包
biocLite("topGO")
biocLite("Rgraphviz")
require(Rgraphviz)
plotGOgraph(ego)
```

气泡图

网络图

GO图

2.4 GSEA分析

基因集富集分析 (Gene Set Enrichment Analysis, GSEA) 的基本思想是使用**预定义的基因集**（通常来自功能注释或先前实验的结果），将基因按照在两类样本中的差异表达程度排序，然后检验预先设定的基因集合是否在这个排序表的顶端或者底端富集。基因集合富集分析**检测基因集合而不是单个基因**的表达变化，因此可以包含这些细微的表达变化，预期得到更为理想的结果。 参考资料：GSEA分析是什么鬼(上)和GSEA分析是什么鬼(下)。

```
# Gene Set Enrichment Analysis（GSEA）
# 获取按照log2FC大小来排序的基因列表
genelist <- diff_gene_deseq2$log2FoldChange
names(genelist) <- rownames(diff_gene_deseq2)
genelist <- sort(genelist, decreasing = TRUE)
# GSEA分析（具体参数参考：https://mp.weixin.qq.com/s/p-n5jq5Rx2TqDBStS2nzoQ）
gsemf <- gseGO(genelist,
               OrgDb = org.Mm.eg.db,
               keyType = "ENSEMBL",
               ont="MF"
)
# 查看大致信息
head(gsemf)
# 画出GSEA图
gseaplot(gsemf, geneSetID="GO:0000977")
```

GSEA结果分析图

2.5 KEGG（pathway）分析

KEGG将基因组信息和高一级的功能信息有机地结合起来，通过对细胞内已知生物学过程的计算机化处理和将现有的基因功能解释标准化，对基因的功能进行系统化的分析。 KEGG的另一个任务是一个将基因组中的一系列基因用一个细胞内的分子相互作用的网络连接起来的过程，如一个通路或是一个复合物，通过它们来展现更高一级的生物学功能。 参考文章：http://blog.sciencenet.cn/blog-364884-779116.html KEGG物种缩写：http://www.genome.jp/kegg/catalog/org_list.html GO和KEGG输出文件解读：http://www.bio-info-trainee.com/370.html

```
# 转换ID适合KEGG
```

```
x=bitr(rownames(diff_gene_deseq2),fromType = "ENSEMBL",toType = "ENTREZID", OrgDb = "org.Mm.eg.db")
# 获取keggID
kegg <- x[,2]
# KEGG分析，在KEGG官网中，物种都有对应的缩写，小鼠mmu，其他的缩写看官网：http://www.genome.jp/kegg/catalog/org_list.html
ekk <- enrichKEGG(kegg, keyType = "kegg",organism = "mmu", pvalueCutoff = 0.05, pAdjustMethod = "BH", qvalueCutoff = 0.1)
head(summary(ekk))
# 气泡图
dotplot(ekk, font.size=5)
# 将GO/KEGG结果转换成CSV格式输出
write.csv(as.data.frame(ekk),"KEGG-enrich.csv",row.names =F)
write.csv(as.data.frame(ego),"GO-enrich.csv",row.names =F)
```

KEGG分析可视化

PS：最后，终于完成了转入组入门，从小白慢慢的开始入门，确实不容易，中间有想过要放弃，真的太难，没有完全一样的流程可以让我参考，只能一遍看别人的，一遍自己摸索，慢慢的学习，我很庆幸自己坚持了来了，走了一遍流程，虽然没有那样的熟悉，但是却是个极大的进步。这里必须要感谢几个大牛的帮助：**Jimmy大神，徐洲更同学，还有Y叔**，主要参考了他们几个人的博客，边学习，边进步，着实不易。

直播基因组分析

我的基因组 | 解惑帖

一个标准的基因检测报告目录

生信技能树

编辑：思考问题的熊