2b 微生物流程 SOP

20200301 张荣超

目录

一、酶切/拆分数据脚本

脚本功能：1）对 fasta 文件进行电子酶切；2）对单端和双端 shotgun 数据进行质控并电子酶切；3）对单标签 2brad 数据（SE50 或 PE150）进行质控并提取标签；4）对五标签 2brad 数据（PE150）进行质控并提取标签

1 算法详细介绍

1.1 对 fasta 文件进行电子酶切

这个没有太多要说的。

1.2 单端 shotgun 数据进行质控并电子酶切

1) 对数据进行质控；
2) 质控合格的 reads 进行电子酶切。

1.3 双端 shotgun 数据进行质控并电子酶切

1) 使用 flash 默认参数进行 PE reads 拼接（注意：当插入片段过小时，flash 拼接率会显著降低）；
2) 将有 overlap 的拼接数据和没有 overlap 的 R1、R2 数据合并；
3) 对数据进行质控；
4) 质控合格的 reads 进行电子酶切。

1.4 单标签 2brad 数据（SE50 或 PE150）进行质控并提取标签

1) 如果 reads 长度大于 50bp，则截取前 50bp 数据；
2) 截取后的数据是否含有酶切位点；
3) 存在酶切位点，则提取酶切标签，并进行数据质控。

1.5 对五标签 2brad 数据（PE150）进行质控并提取标签

注意：五标签文库结构需为“1 标签-3bp -2 标签-3bp-3 标签-3bp-4 标签-3bp-

5 标签”，3bp 为 adaptre

1) 使用 flash 默认参数进行 PE reads 拼接（注意：当插入片段过小时，flash 拼接率会显著降低，但是五标签 2brad 数据一般在 150bp 以上，无需考虑）；

2) 按照每个标签设定范围，提取范围序列；

3) 提取的范围序列是否有酶切位点；

4) 存在酶切位点，则提取酶切标签，并进行数据质控。

2 脚本 help

```
Description
        This script has Four Functions
        1.Electronic Enzyme Digestion of Genome, run: perl EeTt.pl -i genome.fa(.gz) -t 1 -s 1 -od . -op sample
        2.Tag Extraction From Shotgun, run: perl EeTt.pl -i shotgun.1.fq(.gz) (shotgun.1.fq.gz) -t 2 -s 1 -od . -op sample
        3.SE Platform Single Label Data Split, run: perl EeTt.pl -i 2bsingle.fq(.gz or 2bsingle.1.fq.gz) -t 3 -s 1 -od . -op sample
        4.PE Platform Five Label Data Split, run: perl EeTt.pl -i R1.fq(.gz) R2.fq(.gz) -t 4 -s 1 -od . -op sample1 sample2 sample3 sample4 sample5
Usage
        perl EeTt.pl -i <input_file> -t <type> -s <site> -od <outdir> -op <outprefix> [options]*
Necessary Parameters
        -i  <file>   Input File (.gz supported)
        -t  <int>    Type of Input File
                     [1] Genome Data in Fasta Format
                     [2] Shotgun Data in Fastq Format
                     [3] SE Platform Data in Fastq Format
                     [4] PE Platform Data in Fastq Format
        -s  <int>    Enzyme Site
                     [1]CspCI  [9]BplI
                     [2]AloI   [10]FalI
                     [3]BsaXI  [11]Bsp24I
                     [4]BaeI   [12]HaeIV
                     [5]BcgI   [13]CjePI
                     [6]CjeI   [14]Hin4I
                     [7]PpiI   [15]AlfI
                     [8]PsrI   [16]BslFI
        -od <dir>    Output Dir (if not exists,it will be created)
        -op <str>    Output Prefix (The sample name)
Option Parameters
        -gz <str>    Whether the Output File is Compressed or Not [yes] (yes or no)
Option Parameters (only useful for 2,3,4 funcion)
        -qc <str>    Whether quality control is required [yes] (yes or no)
        -n  <float>  Maximum Ratio of Base "N" [0.08]
        -q  <int>    Minimum Quality Score to Keep [30]
        -p  <int>    Minimum Percent of Bases that must have [-q] Quality [80]
        -b  <int>    Quality Values Base [33]
        -fm <str>    Output Format for Data Split [fa]
Author  Sunzheng, Zhangrongchao 20200228
```

二、构建 2b unique 标签数据库脚本

脚本功能：计算指定基因组间，各水平下（界门纲目科属种），每个基因组的 unique 标签。

1 算法详细介绍（需要安装 perl 模块 Parallel::ForkManager）

1) 调用 EeTt.pl 脚本，对所有基因组多线程进行电子酶切；

2) 循环所有基因组酶切结果，记录标签到哈希中（哈希结构：$hash{tag}{分类}++）；

3) 再次循环每个基因组的每个标签，判定该标签是否只对应一种分类。

若只对应一种分类，则为该水平下 该基因组的 unique 标签。（若某个标签在该基因组有重复且在该分类下为 unique，那么该基因组最终结果输出多个一样的标签）

2 脚本 help

```
DESCRIPTION
  build 2b unique tag database
USAGE
  perl makedatabase.pl
PARAMETERS
  -l <s> genome classification list (the line which begins with # will be ignored)
        eg:unique_name<tab>kingdom<tab>phylum<tab>class<tab>order<tab>family<tab>genus<tab>specie<tab>strain<tab>genome_path
  -s <i> enzyme site
        [1]CspCI   [9]BplI
        [2]AloI    [10]FalI
        [3]BsaXI   [11]Bsp24I
        [4]BaeI    [12]HaeIV
        [5]BcgI    [13]CjePI
        [6]CjeI    [14]Hin4I
        [7]PpiI    [15]AlfI
        [8]PsrI    [16]BslFI
  -t <s> database level. One or more of kingdom,phylum,class,order,family,genus,specie,strain. Use 'all' for any level. (comma separated).
  -o <s> outdir (if not exists,it will be created)
OPTION
  -c <i> cpu [10]
  -v <s> verbose
AUTHOR:  Sunzheng, Zhangrongchao 20200228
```

三、定性/定量脚本

脚本功能：使用测序提取的 2b 标签 reads 和数据库，鉴定某水平下菌的含量。

1 算法详细介绍

1) 数据库读取，存入哈希。哈希结构如下：

a) 根据 GCF 找到 classify：$hs_GCF2class{GCFid}=classify;

b) 根据标签找到 GCF（可能对应多个 GCF）：push @{$hs_tag2GCF{标签}},GCFid;

c) 记录每个 classify 的每个 GCF 的每个标签次数：$hs_tag_theory_num{classify}{GCFid}{标签}++。（基因组某个标签会重复的原因，见“构建 2b unique 标签数据库”中算法详细介绍部分）

2) 样品数据读取，首先根据数据库中标签找到对应的 GCFid，然后根据 GCFid 找到对应的 classify。得到以上信息后，将信息存入哈希。哈希结构如下：

a) 记录每个分类标签深度信息：$hs_tag_num{classify}{标签}++;

b) 记录检测到的标签在数据库中每个 classify 的每个 GCF 的每个

标签次数的次数：$hs_detected_GCF_tag{classify}{GCFid}{标签}=$hs_tag_theory_num{classify}{GCFid}{标签}；

3) 输出各分类下各 GCFid 检测到的标签种类数=keys %$hs_detected_GCF_tag{classify}{GCFid}；（该结果可以用来过滤同一个分类下基因组过多）

4) 通过哈希$hs_tag_num{classify}{标签}输出每个分类下检测到的每个标签的深度（每个分类形成一个文件）；

5) 通过哈希$hs_tag_num{classify}{标签}计算出每个分类下鉴定出的标签种类数（Sequenced_Tag_Num）、所有鉴定出的标签深度和（Sequenced_Reads_Num）、所有鉴定出的标签的平均深度（Sequenced _Reads_Num/Sequenced_Tag_Num）、所有鉴定出的标签深度>1 的标签数（Sequenced_Tag_Num(depth>1)）；

6) 通过哈希$hs_tag_theory_num{classify}{GCFid}{标签}得到每个分类下的每个 GCFid 的每个标签数，可以计算出每个分类下所有 GCFid 平均标签数（Theoretical_Tag_Num）。

a) 循环每个分类的每个 GCFid 的每个标签，$species_all_theory_num+=$hs_tag_theory_num{classify}{GCFid}{标签}；

b) 每个分类下所有 GCFid 平均标签数=$species_all_theory_num/（keys $hs_tag_theory_num{classify}）；

7) 计算 G_score：对 Sequenced_Tag_Num*Sequenced_Reads_Num 结果开平方。没有通过 G_score 阈值的分类在统计表中被删除。

2 脚本 help

```
DESCRIPTION
        quantitative/quantitative analysis using 2b tag
USAGE
        perl Calculate_Tag_Percent.pl
PARAMETERS
        -l <s> input list (the line which begins with # will be ignored)
               eg: sample_name<tab>data_path(fa|fq)(.gz)
        -d <s> database path
        -t <s> database level. One of kingdom,phylum,class,order,family,genus,specie,strain.
        -s <s> enzyme site
               [1]CspCI  [9]BplI
               [2]AloI   [10]FalI
               [3]BsaXI  [11]Bsp24I
               [4]BaeI   [12]HaeIV
               [5]BcgI   [13]CjePI
               [6]CjeI   [14]Hin4I
               [7]PpiI   [15]AlfI
               [8]PsrI   [16]BslFI
        -o <s> outdir (if not exists,it will be created)
OPTION
        -g <i> G score threshold [0, it means >=0]
        -v <s> show detail [yes] (yes or no)
AUTHOR:  Sunzheng, Zhangrongchao 20200228
```

3 脚本数据说明

shui-1　　某样品定性/定量目录

├── shui-1.BcgI　鉴定到的每个分类的标签深度文件夹

│　├── 1063.xls 鉴定到的每个分类的标签深度文件：第一列为标签序列，第二列为深度

│　├── 1280.xls

│　├── 1299.xls

│　├── 1309.xls

│　├── 1520.xls

│　├── 1596.xls

│　├── 1660.xls

│　├── 1680.xls

│　├── 837.xls

│　└── human.xls

├── shui-1.BcgI.GCF_detected.xls

└── shui-1.BcgI.xls

shui-1.BcgI.GCF_detected.xls 文件：鉴定到的每个分类的每个 GCF 标签种

类数 占 理论种类数的百分比。倒数第四列为 GCFid，倒数第三列为该基因组在数据库中理论标签种类数，倒数第二列为测到该基因组标签种类数，倒数第一列为百分比。

shui-1.BcgI.xls 文件：统计结果文件。

a) Theoretical_Tag_Num：某分类下所有 GCFid 平均理论标签数
b) Sequenced_Tag_Num：测到的标签种类数
c) Sequenced_Reads_Num：测到的标签深度之和
d) Sequenced_Tag_Num(depth>1) ：测到的标签深度>1 的种类数
e) G_Score：gscore

四、多酶定性/定量结果合并脚本

脚本功能：将指定组合酶的定性/定量结果进行合并，并重新计算 g_score。

1 算法详细介绍

将多种酶定性/定量结果中的 理论标签数累加、测到的标签种类数累加、测到的标签深度之和累加、测到的标签深度>1 的种类数累加，其他值根据累加后的结果重新计算。

2 脚本 help

```
DESCRIPTION
USAGE
        perl calculate_combine.pl
PARAMETERS
        -l  <s> input list (the line which begins with # will be ignored)
              eg: sample_name<tab>...
        -s  <i> enzyme site. One or more of site (comma separated).
              [1]CspCI  [9]BplI
              [2]AloI   [10]FalI
              [3]BsaXI  [11]Bsp24I
              [4]BaeI   [12]HaeIV
              [5]BcgI   [13]CjePI
              [6]CjeI   [14]Hin4I
              [7]PpiI   [15]AlfI
              [8]PsrI   [16]BslFI
              [17]All_Detected_Enzyme
        -io <s> input and output dir
OPTIONS
        -m  <s> mark [combine]
        -g  <i> G score threshold [0, it means >=0]
AUTHOR:  Sunzheng, Zhangrongchao 20200301
```

五、整体定性定量流程脚本

脚本功能：将各个脚本串联，实现一键化得到定性/定量结果。注意：定量选取的酶切位点必须包含在定量选取的酶切位点之内。

1) 对同一类型数据只进行酶切；
2) 对同一类型数据进行酶切和定性；
3) 对同一类型数据进行酶切、定性和定量；
4) 如果先进行了酶切和定性，确定了阈值，可只进行定量分析。

1 算法详细介绍

1) 对参数进行检测，对数据库进行检测；
2) 对样品进行批量酶切（已存在的酶切结果不会二次酶切，但是不会检测酶切结果的完整性）；
3) 对样品进行批量定性分析（不对结果 gscore 过滤）；
4) 整合多酶定性分析结果（不对结果 gscore 过滤）；
5) 根据合并定性分析结果，筛选大于 gscore 阈值的分类；

6) 根据每个酶检测到的 GCFid，筛选大于测到标签种类阈值的 GCF（且通过 gscore 阈值），整理成构建数据库的列表；

7) 根据列表进行某个样品指定酶和水平的数据库构建；

8) 根据数据库进行定量；

9) 整合多酶定量分析结果（不对结果 gscore 过滤）。

2 脚本 help

```
DESCRIPTION
  shotgun/2brad pipline
USAGE
  perl pipline.pl
PARAMETERS
  -t  <i> Type of Input File in sample list(para -l)
          [1] Genome Data in Fasta Format
          [2] Shotgun Data in Fastq Format(SE or PE)
          [3] SE Platform Data in Fastq Format
          [4] PE Platform Data in Fastq Format
  -l  <s> sample list (the line which begins with # will be ignored)
          [1] sample<tab>sample.fa(.gz)
          [2] sample<tab>shotgun.1.fq(.gz)(<tab>shotgun.1.fq.gz)
          [3] sample<tab>2bsingle.fq(.gz or 2bsingle.1.fq.gz)
          [4] sample1<tab>sample2<tab>sample3<tab>sample4<tab>sample5<tab>R1.fq(.gz)<tab>R2.fq(.gz)
  -d  <s> database path
  -o  <s> outdir (if not exists,it will be created)
OPTIONS of Qualitative Analysis
  -p  <s> qualitative or not [yes] (yes or no)
  -s1 <s> qualitative enzyme site. One or more of site. (comma separated) [5]
          [1]CspCI  [5]BcgI  [9]BplⅠ     [13]CjePI  [17]AllEnzyme
          [2]AloI   [6]CjeI  [10]FalⅠ    [14]Hin4I
          [3]BsaXI  [7]PpiI  [11]Bsp24I  [15]AlfI
          [4]BaeI   [8]PsrI  [12]HaeIV   [16]BslFI
  -t1 <s> qualitative database level. One of kingdom,phylum,class,order,family,genus,specie,strain. [specie]
OPTIONS of Quantitative Analysis
  -q  <s> quantitative or not [yes] (yes or no)
  -gscore <i> G score threshold of classify in qualitative analysis, it decides quantitative database. [0]
  -gcf <i> detected tag threshold of GCF in qualitative analysis, it decides quantitative database. [1]
  -s2 <s> quantitative enzyme site (refer to -s1) [5, must be included in para -s1]
  -t2 <s> quantitative database level. One of kingdom,phylum,class,order,family,genus,specie,strain. [specie]
OPTIONS of CPU
  -c1 <i> enzyme cpu [10]
  -c2 <i> calculate cpu [8] (each CPU needs about 15-30G of memory)
OPTIONS of Quality Control
  -qc   <s> quality control or not [yes] (yes or no)
  -qc_n <f> Maximum Ratio of Base "N" [0.08]
  -qc_q <i> Minimum Quality Score to Keep [30]
  -qc_p <i> Minimum Percent of Bases that must have [-q] Quality [80]
  -qc_b <i> Quality Values Base [33]
AUTHOR:  Sunzheng, Zhangrongchao 20200301
```