**2b微生物流程SOP**

20200301 张荣超

目录

[一、酶切/拆分数据脚本 3](#_Toc33990763)

[二、构建2b unique标签数据库脚本 4](#_Toc33990764)

[三、定性/定量脚本 5](#_Toc33990765)

[四、多酶定性/定量结果合并脚本 8](#_Toc33990766)

[五、整体定性定量流程脚本 9](#_Toc33990767)

### 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

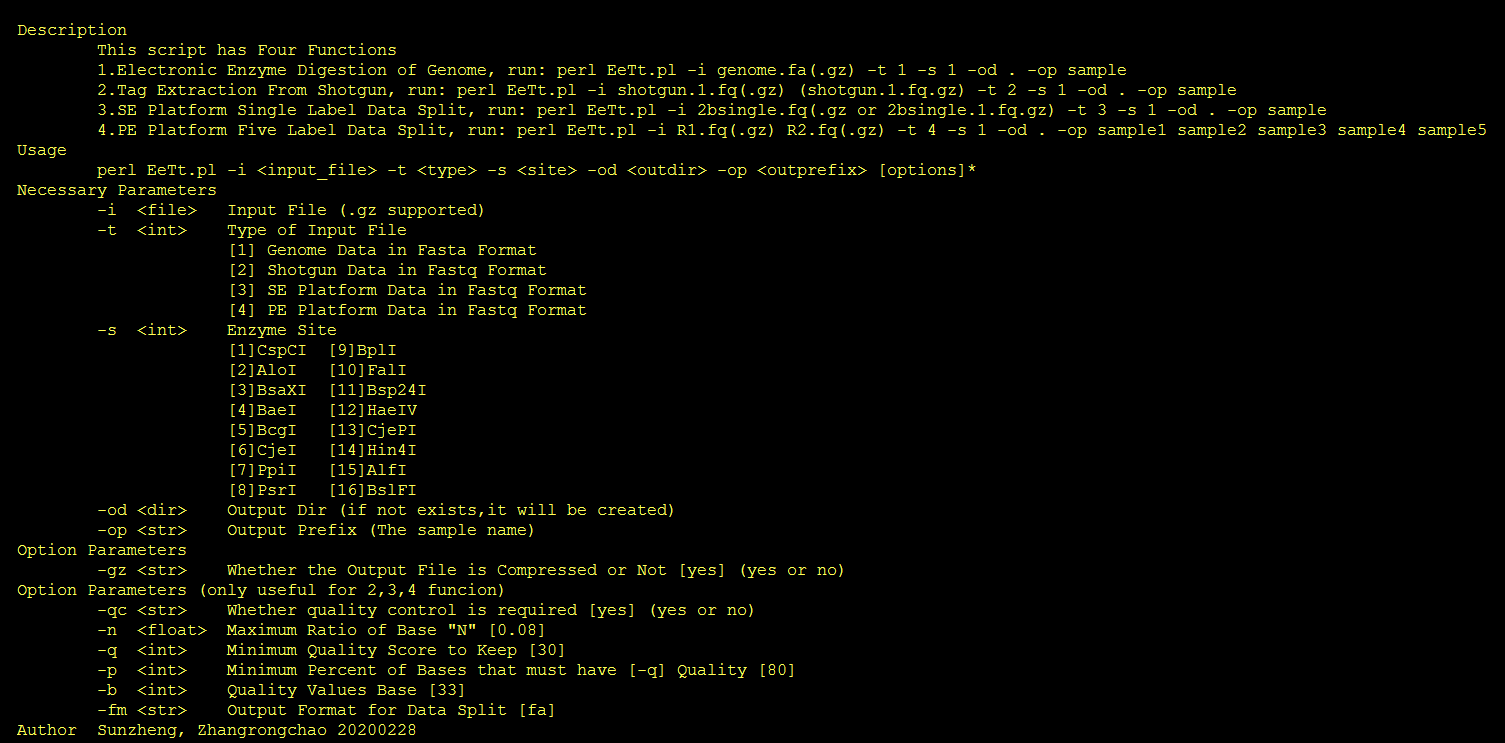
1. 如果reads长度大于50bp，则截取前50bp数据；
2. 截取后的数据是否含有酶切位点；
3. 存在酶切位点，则提取酶切标签，并进行数据质控。

##### **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



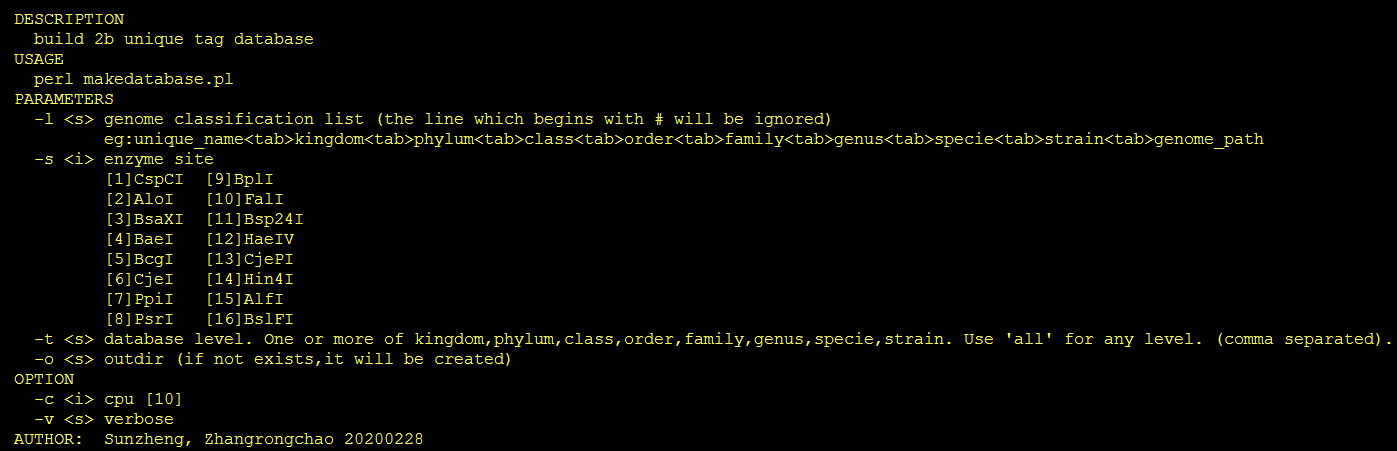
### 二、构建2b unique标签数据库脚本

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



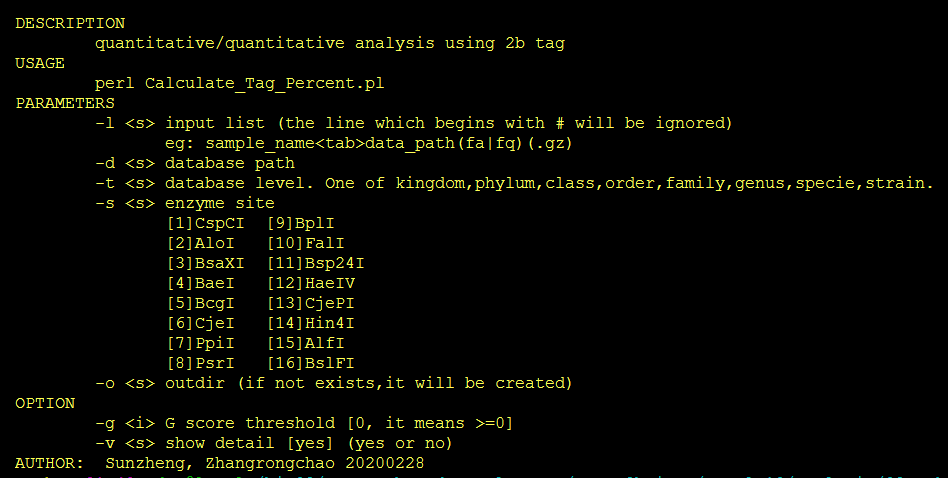
### 三、定性/定量脚本

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：
2. 根据GCF找到classify：$hs\_GCF2class{GCFid}=classify；
3. 根据标签找到GCF（可能对应多个GCF）：push @{$hs\_tag2GCF{标签}},GCFid；
4. 记录每个classify的每个GCF的每个标签次数：$hs\_tag\_theory\_num{classify}{GCFid}{标签}++。（基因组某个标签会重复的原因，见“构建2b unique标签数据库”中算法详细介绍部分）
5. 样品数据读取，首先根据数据库中标签找到对应的GCFid，然后根据GCFid找到对应的classify。得到以上信息后，将信息存入哈希。哈希结构如下：
6. 记录每个分类标签深度信息：$hs\_tag\_num{classify}{标签}++；
7. 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}；
8. 输出各分类下各GCFid检测到的标签种类数=keys %$hs\_detected\_GCF\_tag{classify}{GCFid}；（该结果可以用来过滤同一个分类下基因组过多）
9. 通过哈希$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）；
10. 通过哈希$hs\_tag\_num{classify}{标签}计算出每个分类下鉴定出的标签种类数（Sequenced\_Tag\_Num）、所有鉴定出的标签深度和（Sequenced\_Reads\_Num）、所有鉴定出的标签的平均深度（Sequenced \_Reads\_Num/Sequenced\_Tag\_Num）、所有鉴定出的标签深度>1的标签数（Sequenced\_Tag\_Num(depth>1)）；
11. 通过哈希$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）。
12. 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}；
13. 每个分类下所有GCFid平均标签数=$species\_all\_theory\_num/（keys $hs\_tag\_theory\_num{classify}）；
14. 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



#### 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。

1. Theoretical\_Tag\_Num：某分类下所有GCFid平均理论标签数
2. Sequenced\_Tag\_Num：测到的标签种类数
3. Sequenced\_Reads\_Num：测到的标签深度之和
4. Sequenced\_Tag\_Num(depth>1) ：测到的标签深度>1的种类数
5. G\_Score：gscore

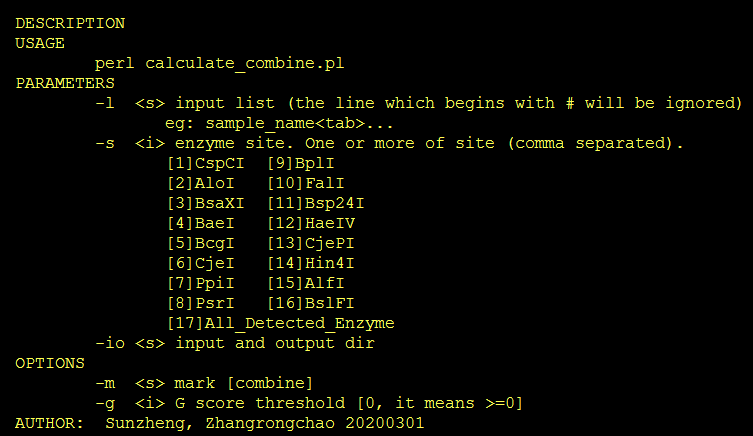
### 四、多酶定性/定量结果合并脚本

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



### 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

