

## 物种鉴定 SOP:

### 分两种情况:

#### 第一种情况: 使用已有数据库:

[目前已有 nt/fish/water/mitofish/prok/euk]

202310 新增数据库:

fungi/metazoa/viridiplantae/Arthropoda/fungi\_metazoa\_viridiplantae

#### 投递物种鉴定流程:

准备文件 1: fqlist.txt: 四列: sample\_name r1.fq r2.fq db 【样品名字 测序数据 R1 测序数据 R2 以上四种数据库名字即可】 tab 分割

准备文件 2: 更改 datajson: 复制

/mnt/lustre/users/sanger-dev/sg-users/dandanzhang/09\_species\_identification/species\_identification\_module\_data.json

到自己路径下, 参数修改: 任务 id / fq\_list / confidence 【存在默认值, 若有特殊需求可自行给定】 / hitgroups 【存在默认值, 若有特殊需求可自行给定】 / ref 【去除宿主】

投递:

~/wpm/bin/run\_work -j species\_identification\_module\_data.json

结果查看路径: /mnt/lustre/sanger-dev\_workspacePractise/投递任务日期/SpeciesIdentificationModule\_投递任务 id/

## 第二种情况：老师给定物种范围需要自建库：

第一步：建立本项目数据库：

### ①：指定物种拉丁文名：

```
bash self_built_library_name.sh dbname name.list
```

注：dbname 为建立的本项目数据库名字。

name.list 为需要创建的数据库包含的物种拉丁文 list，一行一个。

建议 ssh 到计算节点进行这一步[compute-4-\*都可 compute-4-2 内存最大]!!!!!!

### ②：指定物种 tax id：获取方式：

NCBI 上查找对应级别的物种 taxid 或者根据大类 taxid 获取：

```
taxonkit list --ids 7742 --indent "" >taxid.list
```

得到 taxid 后：

```
bash self_built_library_taxid.sh dbname taxid.list
```

注：dbname 为建立的本项目数据库名字。

taxid.list 为需要创建的数据库包含的物种 tax id list，一行一个。

建议 ssh 到计算节点进行这一步[compute-4-\*都可 compute-4-2 内存最大]!!!!!!

## 第二步：投递物种鉴定流程：

准备文件 1：fqlist.txt:四列：sample\_name r1.fq r2.fq db [样品名字 测序数据 R1 测序数据 R2 自建库数据库文件夹绝对路径]tab 分割

准备文件 2：更改 datajson：复制

/mnt/lustre/users/sanger-dev/sg-users/dandanzhang/09\_species\_identification/species\_identification\_module\_data.json

到自己路径下，更改：id 和 fq\_list 参数投递即可：

~/wpm/bin/run\_work -j species\_identification\_module\_data.json

结果查看路径：/mnt/lustre/sanger-dev\_workspacePractise/投递任务日期/SpeciesIdentificationModule\_投递任务 id/

可能存在售后：KrakenTools

[GitHub - jenniferlu717/KrakenTools: KrakenTools provides individual scripts to analyze Kraken/Kraken2/Bracken/KrakenUniq output files](https://github.com/jenniferlu717/KrakenTools)

①：物种鉴定 reads 提取：

extract\_kraken\_reads.py :

- -i/--input MYFILE.BRACKEN.....Bracken output file
- -o/--output MYFILE.BRACKEN\_NEW.....Bracken-style output file with filtered taxids
- --include TID TID2.....taxonomy IDs to include in output file [space-delimited]
- --exclude TID TID2.....taxonomy IDs to exclude in output file [space-delimited] (separated by spaces)

②：结果中过滤某些分类并对剩下的结果重新分配 reads 数：

filter\_bracken\_out.py

- -i/--input MYFILE.BRACKEN.....Bracken output file

- `-o/--output MYFILE.BRACKEN_NEW.....Bracken-style output file with filtered taxids`
- `--include TID TID2.....taxonomy IDs to include in output file [space-delimited]`  
`--exclude TID TID2.....taxonomy IDs to exclude in output file [space-delimited]`