

基于二代测序的真菌基因组组装和注释

Assembly and Annotation of Fungal Genome Based on Illumina Sequencing

马紫英^{1,2}, 吴琦¹, 周欣^{1,2}, 李宽¹, 蔡磊^{1*}

¹ 中国科学院微生物研究所真菌学国家重点实验室, 北京; ² 中国科学院大学生命科学学院, 北京

*通讯作者邮箱: cail@im.ac.cn

摘要: 高通量测序技术的飞速发展使得基因组测序成本不断降低, 近几年来, 越来越多的真菌物种完成了全基因组测序。对于真菌基因组, 二代测序的低成本使得大规模测序成为可能, 通过三代测序及 Hi-C 测序辅助组装等方式, 可以使基因组组装到近染色体水平, 极大的促进了人们对真菌的起源进化、群体遗传多样性、功能基因、致病机制、次级代谢产物等遗传及生物学特性的认识。基因组的组装、基因预测和注释是基因组学研究的基础, 本流程以真菌基因组 Illumina 测序数据为例, 详述了真菌基因组的组装、注释流程。

关键词: 真菌基因组, 数据质控, 基因组组装, 基因预测注释

仪器设备

1. 服务器 (型号: PowerEdge R940xa; 系统: Ubuntu18.04.3; CPU: 英特尔至强 6248, 80 核, 1T 内存)

软件和数据库

1. SRA Toolkit (version2.10.8)
2. FastQC (version0.11.8)
3. Trimmomatic (version0.39)
4. FastUniq (version1.1)
5. SPAdes (version3.12.0)
6. QUAST (version5.0.2)
7. BUSCO (version3.0.2)
8. OrthoDB (version9)
9. Funannotate (version1.4.0)

10. GeneMark-ES (version4.38)
11. BLAST (version2.2.31)
12. eggNOG-mapper (version2.0.0)
13. Swiss-Prot database (2019.10, 264M)

实验步骤

一、软件安装

注：本部分软件安装基于服务器中已安装好 *anaconda*, *python*, *java* 等基本软件。

Conda 是一个开源的软件包管理和环境管理系统，可以一键安装大多数生物学软件及其依赖关系，分为 *anaconda* 和 *miniconda*，在此以 *anaconda* 为例简述其安装过程，如果空间有限，可使用精简版 *miniconda*。

anaconda 下载安装，可根据官网或 *anaconda* 镜像自行下载需要的版本。

官网：<https://www.anaconda.com>

镜像：<https://mirrors.tuna.tsinghua.edu.cn/anaconda/archive/>

\$ `wget -c https://repo.anaconda.com/archive/Anaconda2-5.2.0-Linux-x86_64.sh`

\$ `bash Anaconda2-5.2.0-Linux-x86_64.sh`

安装时许可协议输入 *yes*，默认目录为 `~/anaconda2`，默认不运行 *conda* 直接回车。

添加生物频道

\$ `conda config --add channels defaults`

\$ `conda config --add channels conda-forge`

\$ `conda config --add channels bioconda`

详细安装说明可参考：<https://mp.weixin.qq.com/s/SzJswztVB9rHVh3Ak7jpfA>

1. 需要自行安装的软件，命令如下：

(1) SRA Toolkit

\$ `wget https://ftp-trace.ncbi.nlm.nih.gov/sra/sdk/2.10.8/sratoolkit.2.10.8-ubuntu64.tar.gz -P /home/user/software/`

\$ `tar xzf sratoolkit.2.10.8-ubuntu64.tar.gz`

(2) FastQC

```
61 $ wget http://www.bioinformatics.babraham.ac.uk/projects/fastqc/fastqc\_v0.11.8.zip
62 -P /home/user/software/
```

```
63 $ unzip /home/user/software/fastqc_v0.11.8.zip
```

```
64 $ chmod 755 /home/user/software/FastQC/fastqc
```

65 (3) OrthoDB 数据库

```
66 $ wget http://busco.ezlab.org/v2/datasets/sordariomyceta odb9.tar.gz -P /home/us
67 er/software/BUSCO/
```

```
68 $ tar xzf sordariomyceta_odb9.tar.gz
```

69 (4) Funannotate

70 由于 Funannotate 需要依赖特别多的软件，故基于 anaconda 为该软件新建一个环
71 境。

```
72 $ conda create -n funannotate
```

```
73 $ conda activate funannotate
```

```
74 $ conda install -c bioconda funannotate
```

75 #安装 funannotate_db 数据库

```
76 $ funannotate setup -d $HOME/funannotate_db
```

77 #配置 funannotate_db 环境

```
78 $ echo "export FUNANNOTATE_DB=$HOME/funannotate_db" > /conda/installatio
79 n/path/envs/funannotate/etc/conda/activate.d/funannotate.sh
```

```
80 $ echo "unset FUNANNOTATE_DB" > /conda/installation/path/envs/funannotate/et
81 c/conda/deactivate.d/funannotate.sh
```

82 (5) Genemark (运行 Funannotate 依赖的软件)

83 软件及密钥下载需要注册，下载地址为 http://topaz.gatech.edu/GeneMark/license_download.cgi

```
84 $ tar xzf /home/user/software/gm_et_linux_64.tar.gz
```

```
85 $ gzip -dc /home/user/software/gm_key_64.gz > ~/.gm_key
```

87 # 安装 perl 模块

```
88 $ cpan -i YAML Hash::Merge Logger::Simple Parallel::ForkManager
```

89 #配置 genemark 环境

```
90 $ echo "export GENEMARK_PATH=/home/user/gm_et_linux_64/gmes_petap" > /c
91 onda/installation/path/envs/funannotate/etc/conda/deactivate.d/funannotate.sh
```

92 (6) eggNOG-mapper 及数据库

93 #下载 eggNOG-mapper 安装包

94 \$ git clone <https://github.com/jhcepas/eggno-mapper.git>

95 \$ cd eggno-mapper/

96 #下载并安装数据库

97 \$./download_eggno_data.py

98

99 2. 其他软件 anaconda 安装

100 其他软件如 Trimmomatic, FastUniq, SPAdes, QUAST, BUSCO, blast 可使用 anaconda
101 安装, 以 Trimmomatic 为例, 命令如下:

102 \$ conda install trimmomatic

103

104 3. 配置环境变量

105 自行安装的软件运行时可使用软件的绝对路径或将软件路径写入环境变量, 以方便引
106 用, 以 SRA Toolkit 为例, 命令如下:

107 \$ echo 'PATH=\$PATH:/home/user/software/sratoolkit/bin/' >> ~/.bashrc

108 \$ source ~/.bashrc

109

110 二、数据获得

111 本文中的数据来自“The genome of opportunistic fungal pathogen *Fusarium oxysporum* carries a unique set of lineage-specific chromosomes” (Zhang 等, 2020) 中的二代数据 (SRX6453258) 和基因组 (GCA_009746015.1), 鉴于数据量较大, 故选取部分数据进行示例。

115 1. 下载原始测序数据

116 \$ prefetch SRR9694936

117 2. 下载的数据为 sra 格式, 需要对数据进行拆分和转换

118 \$ fastq-dump --split-files SRR9694936

119 #注: 拆分结果为 SRR9694936_1.fastq 和 SRR9694936_2.fastq

120 3. 下载基因组数据

```
$ wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCA/009/746/015/GCA_009746015.1_
ASM974601v1/GCA_009746015.1_ASM974601v1_genomic.fna.gz
$ gunzip GCA_009746015.1_ASM974601v1_genomic.fna.gz
# 4) 选取部分数据进行后续示例分析，实际分析中不用执行此步骤：
# $ head -n 20000000 SRR9694936_1.fastq > illumina.1.fastq
# $ head -n 20000000 SRR9694936_2.fastq > illumina.2.fastq
# $ seqtk sample GCA_009746015.1_ASM974601v1_genomic.fna 3 > genome.fa
```

三、使用 FastQC 对数据进行质量评估

FastQC 是一款基于 java 的软件，用于对高通量数据进行质量控制。

```
$ mkdir result_qc
```

```
$ fastqc -o result_qc -t 10 illumina.1.fastq illumina.2.fastq
```

运行结束后，使用浏览器打开 html 文件，部分结果如图 1：

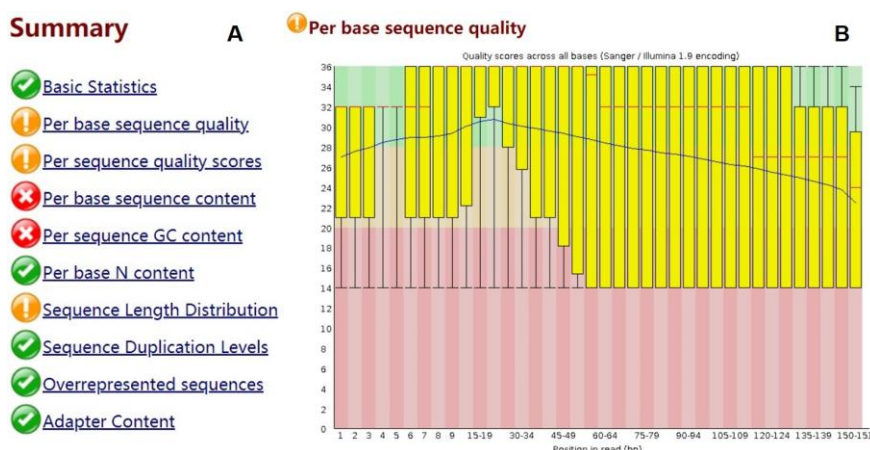


图 1. FastQC 质控结果. A 为报告目录，合格的部分用绿色对勾表示，不合格用红色叉号表示；B 表示每个位置所有碱基的质量值的范围，在绿色区域表示测序质量高。

此份数据 Adapter Content 为绿色对勾，大部分 reads 在红色区域，表示此份数据已去掉接头，但测序质量不高。

更多参数的具体结果解读可参考官网：<http://www.bioinformatics.babraham.ac.uk/projects/fastqc/Help/3%20Analysis%20Modules/>

四、使用 Trimmomatic 对数据进行质量控制

#Trimmomatic 是针对 Illumina 测序平台的数据过滤工具，用来去除接头和低质量碱基，对于未去接头的原始下机数据（Anthony 等，2014），命令如下：

```
$ trimmomatic PE -threads 4 illumina.1.fastq illumina.2.fastq illumina.1.clean.fastq
illumina.1.unpaired.fastq illumina.2.clean.fastq illumina.2.unpaired.fastq ILLUMINA
CLIP:TruSeq3-PE.fa:2:30:10 LEADING:3 TRAILING:3 SLIDINGWINDOW:4:15 MIN
LEN:50 TOPHRED33
```

主要参数：

PE/SE 设定针对 Paired-End 或者 Single-End reads 进行处理

-threads 运行线程数

ILLUMINACLIP:TruSeq3-PE.fa:2:30:10 切接头序列

LEADING:3 切除 reads 开头质量值低于 3 的碱基

TRAILING:3 切除 reads 尾部质量值低于 3 的碱基

SLIDINGWINDOW:4:15 从 reads 的 5' 端开始进行滑窗质量过滤，切掉碱基质量平均值低于阈值（15）的滑窗（4 个碱基）

MINLEN:50 保留剪切后 reads 长度的最小值

TOPHRED33 将碱基质量转换为 phred-33 格式

运行结束后，终端屏幕会出现如下过滤信息，图 2：

```
TrimmomaticPE: Started with arguments:
-threads 4 illumina.1.fastq illumina.2.fastq illumina.1.clean.fastq illumina.1.unpaired.fastq illumina.2.clean.fastq illumina.2.unpaired.fastq
LEADING:3 TRAILING:3 SLIDINGWINDOW:4:15 MINLEN:50 TOPHRED33
Quality encoding detected as phred33
Input Read Pairs: 5000000 Both Surviving: 3347732 (66.95%) Forward Only Surviving: 766582 (15.33%) Reverse Only Surviving: 384015
(7.68%) Dropped: 501671 (10.03%)
TrimmomaticPE: Completed successfully
```

图 2. Trimmomatic 过滤结果

五、使用 FastUniq 去除 PCR 重复

#illumina 文库构建中 pcr 扩增会给测序结果引入 PCR 重复，可以使用 FastUniq 来去除（Xu 等，2012）。

```
$ mkdir FastUniq
```

```
$ cd FastUniq/
```

#注：需要建立一个文本文件，写入需要处理的成对的 fastq 文件的路径。

173 \$ ls illumina.1.clean.fastq illumina.2.clean.fastq > fragment.list

174 \$ fastuniq -i fragment.list -o illumina.1.rd.fastq -p illumina.2.rd.fastq

175 主要参数:

176 -o 和 -p 均为过滤后输出的序列文件

177 #注: 对于自己测得的基因组数据, 如果测序质控比较严格, 对原始数据进行了很好的
178 过滤, 可以直接使用质控过的 **cleandata** 进行基因组拼接, 而不必执行以上步骤四和
179 五。

180

181 六、基因组 *de novo* 组装

182 二代数据的组装软件较多, 如 SPAdes, SOAPdenovo2, IDBA-UD, ALLPATHS-LG
183 等, 在此我们选用 SPAdes 软件进行 *de novo* 组装 (Anton 等, 2012)。SPAdes 适用
184 于细菌/真菌等小型基因组的组装, 不适用于大型基因组, 输入数据可以是 Ion Torrent,
185 PacBio, Oxford Nanopore, 以及 Illumina paired-end, mate-pairs 和 single reads。对
186 于小型真菌基因组的 Illumina paired-end reads, 组装命令如下:

187 \$ spades.py -k 21,33,55,77,99 --careful --cov-cutoff auto -1 illumina.1.rd.fastq -2
188 illumina.2.rd.fastq -o output

189 主要参数:

190 -k 设置 kmer 的大小

191 --careful 通过运行 MismatchCorrector 模块进行基因组上 mismatches 和 short
192 indels 的修正

193 --continue 从上一次终止处继续运行程序

194 --cov-cutoff auto 计算 coverage 值

195 #如果有多个 paired-end library 数据时, 可使用如下参数:

196 \$ spades.py -k 21,33,55,77,99 --careful --cov-cutoff auto --pe1-1 illumina1_1.fastq --
197 pe1-2 illumina1_2.fastq --pe2-1 illumina2_1.fastq --pe2-2 illumina2_2.fastq -o output

198 运行结束后, 在 output 文件夹中的 scaffolds.fasta 为拼接好的基因组文件。

199 注: 该软件目前已更新到 version 3.14.1, 更多详细参数请参考官方说明文档:

200 <http://cab.spbu.ru/files/release3.14.1/manual.html#sec1.1>

201

注：由于在上述组装示例中使用的数据量少，组装结果不足以进行下一步组装质量评估和基因预测，故选取完整基因组 `GCA_009746015.1_ASM974601v1_genomic.fna` 为例进行质量评估。

七、用 QUAST 评估组装质量

#QUAST 软件用来评估基因组的组装效果，有无参考基因组均可（Alexey 等，2013）。

```
$ quast.py GCA_009746015.1_ASM974601v1_genomic.fna -o output
```

运行结束后，在浏览器中打开 html 文件，结果如图 3：

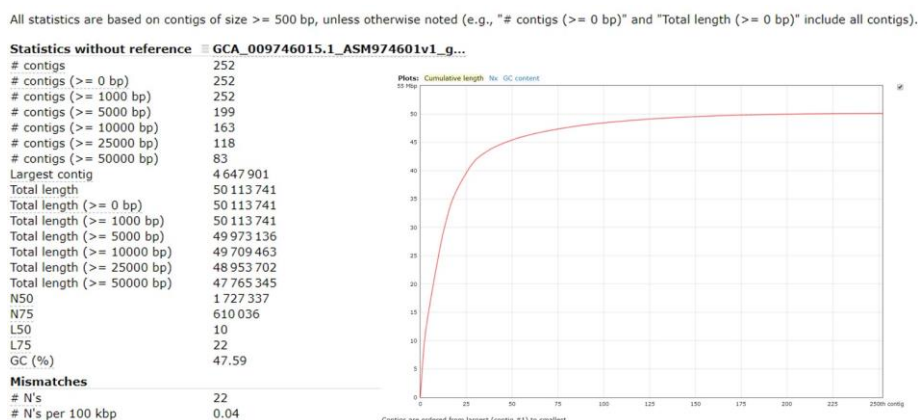


图 3. QUAST 评估结果

该基因组组装共 252 条 contig，基因组大小为约 50M，N50 值为 1727337bp，对于 2 代测序来说组装质量较好。

八、使用 BUSCO 评估组装质量

#BUSCO 软件通过同源基因数据库从基因完整度来评价基因组组装结果（Mathieu 等，2019）。

```
$ python run_BUSCO.py -i GCA_009746015.1_ASM974601v1_genomic.fna -o output -l /home/user/software//BUSCO/sordariomyceta_odb9/ -m genome -c 10
```

主要参数：

-l 指定单拷贝基因数据集

-m 评估模式，包含 3 种类型：geno or genome，基因组序列；tran or transcriptome，转录本序列；prot or proteins，蛋白氨基酸序列。

-c 程序运行线程数

运行结束后会在 run_output 中生成一系列文件，其中 short_summary_output 文档中会显示整体评估结果，如图 4:

```
C:99.0%[S:98.4%,D:0.6%],F:0.4%,M:0.6%,n:3725

3687   Complete BUSCOs (C)
3666   Complete and single-copy BUSCOs (S)
21     Complete and duplicated BUSCOs (D)
16     Fragmented BUSCOs (F)
22     Missing BUSCOs (M)
3725   Total BUSCO groups searched
```

图 4. BUSCO 评估结果

其中 C 值表示完整性，即被评估基因组与 BUSCO 基因相比的比例，S 表示单拷贝基因的比例，D 表示多拷贝基因的比例，M 值表示可能缺少的基因的比例。如图 4，C 值和 S 值均大于 98%，该基因组组装较好。

注: BUSCO 目前已更新到 BUSCO version 5 beta，使用时可下载最新版及相应的 OrthoDB (<https://busco-archive.ezlab.org/v3/>)。

更多帮助信息可参考官网: https://busco.ezlab.org/busco_userguide.html#manual-installation

九、使用 Funannotate 进行基因预测

Funannotate 是一个集基因组预测、注释和比较的综合软件 (Palmer, 2016)，最初是为注释真菌基因组 (小型真核生物，基因组大小约 30 Mb) 而写的，经过不断更新目前也可以用于较大的基因组。该软件的结果输出格式更易于 NCBI 数据库的数据提交。该预测软件整合了 AUGUSTUS, GeneMark, Snap, GlimmerHMM, BUSCO, EVIDENCE Modeler, tbl2asn, tRNAScan-SE, Exonerate, minimap2 的预测结果。

首先屏蔽基因组中的重复序列

```
$ funannotate mask -i genome.fa --cpus 12 -o genome_masked.fasta
```

主要参数:

-i 输入文件，fasta 基因组序列

-o 输出文件

-m 重复序列屏蔽方式，默认 tantna，也可选 repeatmasker 或者 repeatmodeler

-l, 如果选用 repeatmodeler 进行屏蔽，需设置本地 repeat 数据库的位置

```

252 # 使用屏蔽重复序列后的基因组进行基因预测
253 funannotate predict -i genome_masked.fasta --species "Pseudogenus specicus" -o
254 fun/ --busco_seed_species fusarium_graminearum --busco_db sordariomycetes --
255 cpus 12
256 必须参数：
257 -i 输入文件，即 mask 后的输出结果
258 -o 输出文件夹
259 -s, --species 预测基因组的物种名
260 主要可选参数：
261 --maker_gff 自行通过 MAKER2 预测的结果文件，直接用于 EVM 整合预测结果
262 -w, --weights 设置 EVM 整合的权重
263 --busco_seed_species 选择 BUSCO 中 Augustus 软件进行训练的物种名，一般选择
264 近缘物种，默认为 anidulans
265 --busco_db 选择 BUSCO 数据的模型，默认为 dikarya
266 --ploidy 基因组的倍型，默认为 1
267 --genemark_gtf 自行通过 Genemark 预测的结果文件
268 --min_intronlen 最小内含子长度，默认为 10
269 --max_intronlen 最大内含子长度，默认为 2000
270 --min_protlen 最小蛋白序列长度，默认为 50
271 --cpus CPU 默认为 2
272 运行结束后，在输出文件夹 fun 内共 3 个子文件夹，其中 logfiles 文件夹中的文档
273 记录了每一步的运行过程，可以查看结果和报错信息，predict_results 文件夹中为预测
274 结果，包括 Augustus 的训练文件，预测基因的蛋白序列和 CDS 序列文件及 gff 和
275 Genbank 格式的预测文件，可以用于后续的基因注释。

```

276

277 十、基因功能注释

278 基因功能注释主要是基于同源序列比对进行的。根据预测结果中的蛋白序列和蛋白质
 279 数据库进行比对的结果，完成相应的功能注释。常用的数据库包括：Nr 库（NCBI 官
 280 方非冗余蛋白数据库）、Swiss-Prot 数据库（蛋白序列得到实验验证）、KEGG 数据库

（代谢通路数据库）、eggNOG 数据库（直系同源蛋白分组比对数据库，是 NCBI 的 COG 数据库的扩展）、InterPro 数据库（由多种不同的数据库组成，如 CDD, Pfam, ProDom, PROSITE, SMART, SUPERFAMILY 等，两月更新一次）等，另外还有一些功能数据库如 CAZyme、EffectorP 数据库等，可以根据课题要求和分析目的进行选择。在此以 eggNOG（Jaime 等，2017&2019）和 Swiss-Prot 数据库两种不同的注释方式为例进行详述。

1. eggNOG mapper 在线版: <http://eggnog-mapper.embl.de/>

在线版只需选择蛋白序列文件、设置邮箱、提交任务即可，如图 5，操作简单方便，一次最多提交 100000 条蛋白序列，但是不适用于大规模序列注释。

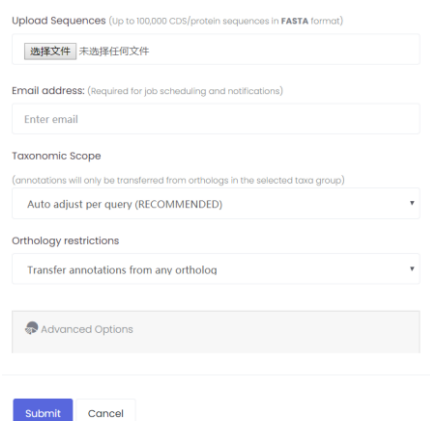


图 5. eggNOG mapper 在线版

2. eggNOG mapper 本地版

```
$ python emapper.py -i proteins.fa --output eggnog_output -m diamond -d euk
```

主要参数:

-i 输入文件

--output 输出文件前缀

-m 设置比对算法。可以选择 hmmer 或 diamond

-d 指定数据库数据，真菌选 euk

运行结束后会生产两个文件，eggnog_output.emapper.seed_orthologs 和

eggnog_output.emapper.annotations，其中 eggnog_output.emapper.seed_orthologs

为每条序列的最佳比对结果列表，重点关注 `eggnog_output.emapper.annotations`，为
对比结果，主要包括：

`query_name`: 序列名

`seed_eggNOG_ortholog`: 最佳比对的 eggNOG 编号

`seed_ortholog_evalue`: 最佳蛋白比对的 E 值

`seed_ortholog_score`: 最佳蛋白比对得分

`predicted_gene_name`: 预测基因名

GOs: GO 注释

KEGG_KO: KEGG 注释的 KO 编号

BiGG_Reactions: 代谢反应

COG functional categories: COG 功能分类

eggNOG free text desc: 功能描述

3. Swiss-Prot 注释

UniProtKB/Swiss-Prot 数据库中的蛋白质功能经过了功能验证，注释准确，数据库较
小，适合用于本地化 `blast` 进行的注释。

官网: <https://www.uniprot.org/>

下载 Swiss-Prot 的蛋白序列并构建 Blast 数据库

\$ `wget`

`ftp://ftp.uniprot.org/pub/databases/uniprot/current_release/knowledgebase/complete/`

`uniprot_sprot.fasta.gz -P /home/user/software/`

\$ `gzip -dc /home/user/software/uniprot_sprot.fasta.gz > uniprot_sprot.fasta`

\$ `makeblastdb -in uniprot_sprot.fasta -dbtype prot -title uniprot_sprot -parse_seqids -`

`out uniprot_sprot -logfile uniprot_sprot.log`

#使用 `blastp` 进行 Swiss-Prot 注释

\$ `blastp -query proteins.fa -out swiss-prot.tab -db /home/user/software/uniprot_sprot`

`-evalue 1e-5 -outfmt 7`

注：结果中每个蛋白的详细信息可查询官网: <https://www.uniprot.org/>

十一、下游分析

通过以上的基因组组装、基因预测和功能注释，我们已经获得了（1）目标物种的基因组信息，（2）蛋白序列和 CDS 序列及其在基因组上的位置，（3）大部分基因的功能预测，这些结果都为后续的个性化分析和深入研究如基因组共线性、同源蛋白分析、进化树构建、基因家族分析、功能基因验证等提供数据基础。

失败经验

1. Genemark-ES 密钥存在一定的有效期，如果密钥过期，可以通过之前的网站重新注册下载新的密钥，替换原来的密钥即可。
2. Genemark 安装 perl 模块的过程中，可能会提示某些模块安装不成功，如 hash-merge，最终使用 anaconda 安装完成。
\$ conda install -c bioconda perl-hash-merge

参考文献

1. Zhang, Y., Yang, H., David, T., Zhou, S. G., Dilay, H. A., Gregory, A. D., Guo, L., Karen, B., Nathan, W., Jeffrey, J. C. et al. (2020). [The genome of opportunistic fungal pathogen fusarium oxysporum carries a unique set of lineage-specific chromosomes.](#) *Commun Biol* 3(50): 1-12.
2. Anthony, M. B., Marc, L. and Bjoern, U. (2014). [Trimmomatic: A flexible trimmer for Illumina sequence data.](#) *Bioinformatics* 30(15): 2114-2120.
3. Xu, H. B., Luo, X., Qian, J., Pang, X. H., Song, J. Y., Qian, G. R., Chen, J. H. and Chen, S.L. (2012). [FastUniq: A fast de novo duplicates removal tool for paired short reads.](#) *PLoS ONE* 7(12): e52249.
4. Anton, B., Sergey, N., Dmitry, A., Alexey, A. G., Mikhail, D., Alexander, S. K., Valery, M. L., Sergey, I. N., Son, P., Andrey, D. P., Alexey, V. P., Alexander, V. S., Nikolay, V., Glenn, T., Max, A. A. and Pavel, A. P. (2012). [SPAdes: a new genome assembly algorithm and its applications to single-cell sequencing.](#) *J Comput Biol* 19(5): 455-477.
5. Alexey, G., Vladislav, S., Nikolay, V. and Glenn, T. (2013). [QUAST: quality assessment tool for genome assemblies.](#) *Bioinformatics* 29 (8): 1072-1075.

6. Mathieu S., Mosè, M. and Evgeny, M.Z. (2019). [BUSCO: assessing genome assembly and annotation completeness](#). *Methods Mol Biol* 1962 :227-245.
7. Palmer, J. (2016). [Funannotate: pipeline for genome annotation](https://funannotate.readthedocs.io/en/latest/index.html). <https://funannotate.readthedocs.io/en/latest/index.html>.
8. Jaime, H. C., Kristoffer, F., Luis, P. C., Damian, S., Lars, J. J., Christian, V. M. and Peer, B. (2017). [Fast genome-wide functional annotation through orthology assignment by eggNOG-mapper](#). *Mol Biol Evol* 34(8): 2115-2122.
9. Jaime, H. C., Damian, S., Davide, H., Ana, H.P., Sofia, K. F., Helen, C., Daniel, R. M., Ivica, L., Thomas, R., Lars J. J., Christian, V. M. and Peer, B. (2019). [eggNOG 5.0: a hierarchical, functionally and phylogenetically annotated orthology resource based on 5090 organisms and 2502 viruses](#). *Nucleic Acids Res* 47(Database issue): D309–D314.