

ViWrap: 用于从宏基因组中识别、归类 and 预测病毒的宿主关系的模块化工具

Zhichao Zhou (周之超), Cody Martin, James C. Kosmopoulos,
Karthik Anantharaman*

威斯康辛大学麦迪逊分校细菌学系

小组网页: <https://github.com/AnantharamanLab/ViWrap>



Zhichao Zhou (zzhou388@wisc.edu);
Karthik Anantharaman (karthik@bact.wisc.edu)



Introduction

从宏基因组中重建的病毒组和病毒基因组



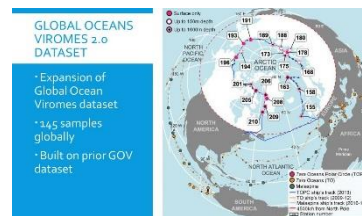
揭示地球的病毒组

David Paez-Espino & Nikos C. Kyrpides
Nature 2016 (DOE-JGI)

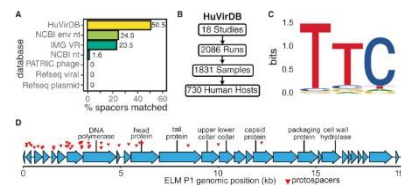
病毒组 (Virome) :
专门针对环境样本中病毒部分的宏基因组学

从宏基因组中重建的病毒基因组:
从批量宏基因组中重建的病毒基因组

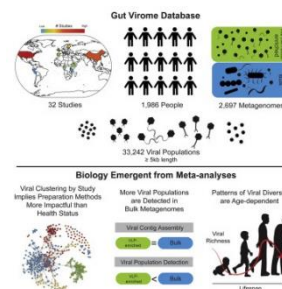
当前病毒组数据库的分布



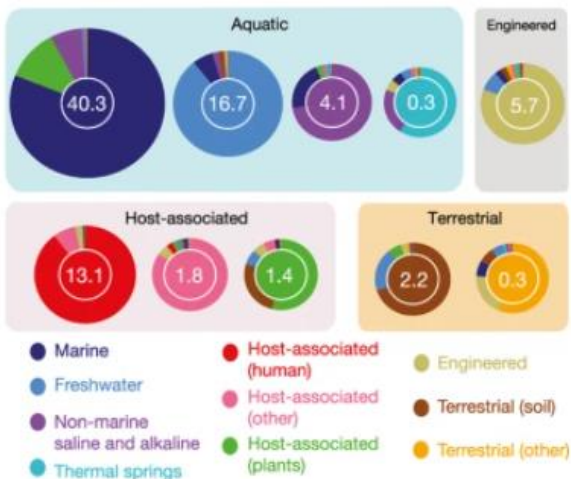
2019
Global Oceans Viromes (GOV) v2.0



2019
Human virome database (HuVirDB)

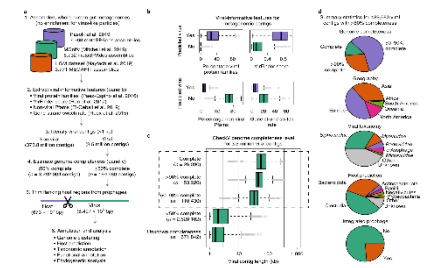
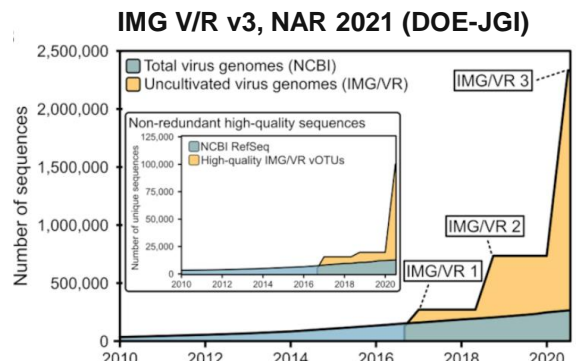


2020
Gut Virome Database (GVD)
(包含两种获得病毒组的方法)



2021 IMG V/R database (v3)

- 包含来自所有环境的病毒
- 不断进行持续更新



2021
Metagenomic Gut Virus (MGV)

背景



VIBRANT
Virus Identification By Iterative Annotation


通过混合机器学习和蛋白质相似性方法进行病毒鉴定
Anantharaman实验室, 威斯康星大学麦迪逊分校, 2020年

jiarong/VirSorter2
customizable pipeline to identify viral sequences from (meta)genomic data

一个多分类器、由专业指导的识别器
Sullivan & Roux 实验室, 俄勒冈州立大学和美国能源部基因组科学研究所, 2021年

jessieren/DeepVirFinder
Identifying viruses from metagenomic data by deep learning

基于kmer的卷积神经网络算法识别器
Sun实验室, 南加州大学, 2020年



NCBI RefSeq Virus genomes (NCBI)



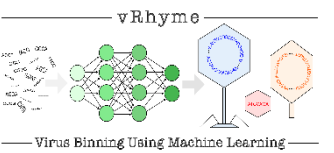
VOG HMMs (<http://vogdb.org>)



IMG/VR V3 具有分类分配的高质量虚拟操作单元 (vOTUs)
(美国能源部基因组科学研究所)

病毒分类学分类的数据库

用于病毒鉴定的工具



vRhyme
Virus Binning Using Machine Learning

基于scaffolds覆盖效果大小和核苷酸特征的病毒分箱
Anantharaman实验室, 威斯康星大学麦迪逊分校, 2022年

病毒分箱



vConTACT2

构建基于整个基因组的基因共享网络, 用于基于距离的分层聚类 and 分类预测
Sullivan实验室, 俄勒冈州立大学, 2019年

病毒分组

MrOIm/drep
Rapid comparison and dereplication of genomes

基于序列相似性对微生物/病毒基因组进行聚类和去冗余处理
Banfield实验室, 加州大学伯克利分校, 2017年



CheckV
检查病毒基因组的质量和完整性
Banfield实验室, 加州大学伯克利分校, 2017年

质量检查



iP

一种集成的机器学习框架用于宿主预测
Roux实验室, 美国能源部基因组科学研究所, 2022年

宿主预测

缺乏一个集成的流程/封装器



软件流程

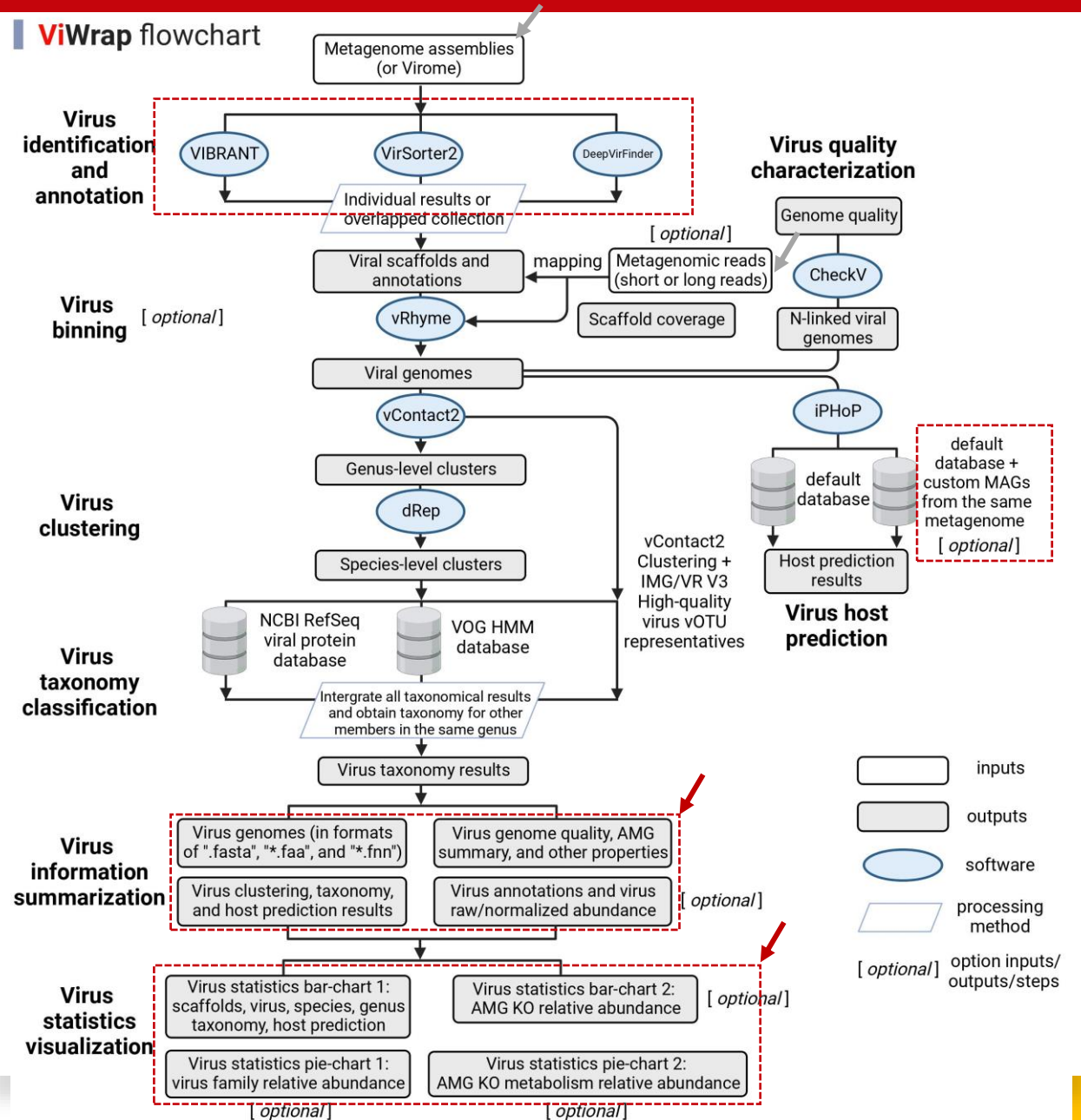
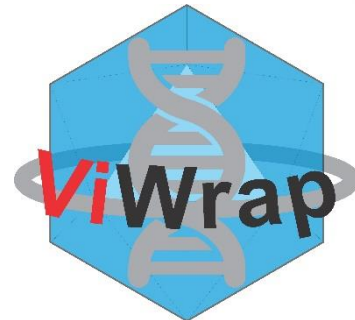
第一步： 三个病毒识别器

第二步： 宏基因组reads mapping、病毒分箱和质量检查

第三步： 将病毒聚类成属和种，并分配分类学信息

第四步： 使用iPHoP预测病毒的宿主

第五步： 总结结果并可视化病毒统计信息



结果呈现

All result folders

- `00_VIBRANT_input_metageome_stem_name`: the virus identification result (would be "00_VirSorter_input_metageome_stem_name", "00_DeepVirFinder_input_metageome_stem_name", "00_VIBRANT_VirSorter_input_metageome_stem_name", or "00_VIBRANT_VirSorter_DeepVirFinder_input_metageome_stem_name")
- `01_Mapping_result_outdir`: the reads mapping result
- `02_vRhyme_outdir`: vRhyme binning result
- `03_vConTACT2_outdir`: vConTACT2 classifying result
- `04_Nlinked_viral_gn_dir`: N-linked viral genome as CheckV inputs
- `05_CheckV_outdir`: CheckV result
- `06_dRep_outdir`: dRep clustering result
- `07_iPHoP_outdir`: iPHoP result for host prediction
- `08_ViWrap_summary_outdir`: Summarized results
- `09_Virus_statistics_visualization`: Visualized statistics of viruses
- `ViWrap_run.log`: running log file containing the issued command and time log

组织好的中间文件夹

- ```
> 08_ViWrap_summary_outdir
- Genus_cluster_info.txt # Virus genus clusters
- Species_cluster_info.txt # Virus species clusters
- Host_prediction_to_genome_m90.csv # Host prediction result at genome level
- Host_prediction_to_genus_m90.csv # Host prediction result at genus level
- Sample2read_info.txt # Reads counts and bases
- Tax_classification_result.txt # Virus taxonomy result
- Virus_annotation_results.txt # Virus annotation result
> Virus_genomes_files # Contains all fasta, faa, ffn files for virus genomes
- vRhyme*.fasta
- vRhyme*.faa
- vRhyme*.ffn
> Virus_normalized_abundance.txt # Normalized virus genome abundance (normalized by 100M reads/sample)
> Virus_raw_abundance.txt # Raw virus genome abundance
> Virus_summary_info.txt # Summarized property for all virus genomes
```

## ViWrap总结文件夹里面的结果

- ```
> 09_Virus_statistics_visualization
> Result_visualization_inputs
- virus_statistics.txt
- virus_family_relative_abundance.txt
- KO_ID_relative_abundance.txt
- KO_metabolism_relative_abundance.txt
> Result_visualization_outputs
- virus_statistics.png # the 1st bar-chart
- virus_family_relative_abundance.png # the 1st pie-chart
- KO_ID_relative_abundance.png # the 2nd bar-chart
- KO_metabolism_relative_abundance.png # the 2nd pie-chart
- virus_statistics.pdf
- virus_family_relative_abundance.pdf
- KO_ID_relative_abundance.pdf
- KO_metabolism_relative_abundance.pdf
```

可视化文件夹中的病毒统计可视化结果

总结和要点

- ViWrap整合了目前可用的工具和数据库，用于全面且严格的病毒筛查
 - 它对于鉴定方法、宏基因组reads和自定义微生物基因组的选择具有灵活性，适用于各种应用场景
 - 它具有一站式、用户友好的工作流程，并生成易于阅读和解析的结果
-
- ViWrap可以用于各种环境设置，包括自然环境、人为环境和与人类微生物组相关的环境。
 - ViWrap可以通过GitHub（<https://github.com/AnantharamanLab/ViWrap>）公开获取。软件的使用方法和结果解释的详细描述可以在该网站上找到。

软件操作

创建ViWrap的conda环境

```
zhichao@sulfur:~$ cd /storage1/data11/ViWrap
zhichao@sulfur:/storage1/data11/ViWrap$ conda create -c bioconda -p /slowdata/yml_environments/ViWrap python=3.8 biopython mamba numpy pandas pyfastx
```

进入conda环境

```
zhichao@sulfur:/storage1/data11/ViWrap$ conda activate /slowdata/yml_environments/ViWrap
(/slowdata/yml_environments/ViWrap) zhichao@sulfur:/storage1/data11/ViWrap$
```

Git克隆ViWrap软件包并将其属性设置为可执行

```
(/slowdata/yml_environments/ViWrap) zhichao@sulfur:/storage1/data11$ git clone https://github.com/AnantharamanLab/ViWrap
(/slowdata/yml_environments/ViWrap) zhichao@sulfur:/storage1/data11$ cd ViWrap
(/slowdata/yml_environments/ViWrap) zhichao@sulfur:/storage1/data11/ViWrap$ chmod +x ViWrap scripts/*.py
(/slowdata/yml_environments/ViWrap) zhichao@sulfur:/storage1/data11/ViWrap$ PATH=`pwd`:$PATH
```

软件操作

设置软件运行需要用到的conda环境

```
(/slowdata/yml_environments/ViWrap) zhichao@sulfur:/storage1/data11/ViWrap$ ViWrap set_up_env --conda_env_dir /slowdata/yml_environments/
```

```
### Set up conda env ###
```

```
[2022-10-30 19:36:16] | Looks like the input parameter is correct
[2022-10-30 19:37:38] | ViWrap-VIBRANT conda env has been installed
[2022-10-30 19:38:18] | ViWrap-vRhyme conda env has been installed
[2022-10-30 19:39:15] | ViWrap-vContact2 conda env has been installed
[2022-10-30 19:39:47] | ViWrap-CheckV conda env has been installed
[2022-10-30 19:40:38] | ViWrap-dRep conda env has been installed
[2022-10-30 19:40:55] | ViWrap-Tax conda env has been installed
[2022-10-30 19:43:50] | ViWrap-iPHoP conda env has been installed
[2022-10-30 19:44:07] | ViWrap-GTDBTK conda env has been installed
[2022-10-30 19:44:34] | ViWrap-vs2 conda env has been installed
[2022-10-30 19:44:56] | ViWrap-Mapping conda env has been installed
[2022-10-30 19:45:59] | ViWrap-DVF conda env has been installed
ViWrap-VIBRANT conda env path has been checked
ViWrap-vRhyme conda env path has been checked
ViWrap-vContact2 conda env path has been checked
ViWrap-CheckV conda env path has been checked
ViWrap-dRep conda env path has been checked
ViWrap-Tax conda env path has been checked
ViWrap-iPHoP conda env path has been checked
ViWrap-GTDBTK conda env path has been checked
ViWrap-vs2 conda env path has been checked
ViWrap-DVF conda env path has been checked
ViWrap-Mapping conda env path has been checked
The total running time is 0:09:48 (in "hr:min:sec" format)
```


软件操作

设置ViWrap数据库

```
(/slowdata/yml_environments/ViWrap) zhichao@sulfur:/storage1/data11/ViWrap$ ViWrap download --db_dir ./ViWrap_db --conda_env_dir /slowdata/yml_environments
```

```
### Welcome to ViWrap ###
```

```
[2022-10-30 19:55:08] | Looks like the input conda software is correct
```

```
...
```

```
Set VIBRANT_DATA_PATH to /storage1/data11/ViWrap/ViWrap_db2/VIBRANT_db  
Downloading VIBRANT databases to /storage1/data11/ViWrap/ViWrap_db2/VIBRANT_db...
```

```
This script will download, extract subsets and press HMM profiles for VIBRANT.  
This process will require ~20GB of temporary free storage space, but the final size requirement is ~11GB in the form of pressed HMM databases.  
Please be patient. This only needs to be run once and will take a few minutes.  
Logger started. Check log file for messages and errors.
```

```
VIBRANT v1.2.1 is good to go!  
See example_data/ for quick test files.
```

```
VIBRANT databases are downloaded successfully. Please see log file for any error messages.
```

```
[2022-10-30 20:04:07] | VIBRANT db has been set up
```

```
...
```

```
2022-10-30 22:31:04 (11.3 MB/s) - './ViWrap_db2/gtdbtk_r202_data.tar.gz' saved [50840267340/50840267340]
```

```
[2022-10-30 22:39:17] | GTDB-Tk db has been set up
```

```
[2022-10-30 22:44:05] | VirSorter2 db has been set up
```

```
Cloning into './ViWrap_db2/DVF_db_tmp'...
```

```
Updating files: 93% (28/30)^MUpdating files: 96% (29/30)^MUpdating files: 100% (30/30)^MUpdating files: 100% (30/30), done.
```

```
[2022-10-30 22:44:09] | DVF db has been set up
```

```
The total running time is 2:49:01 (in "hr:min:sec" format)
```

由于工具的数据库较大，根据服务器的互联网速度，这可能需要几个小时才能完成

软件操作

测试ViWrap

```
(/slowdata/yml_environments/ViWrap) zhichao@sulfur:/storage1/data11/ViWrap$ ViWrap -h
ViWrap v1.2.0: Analyzing wrapper for virus from metagenome

Usage: ViWrap <task> [options]

Task:
run          Run the full wrapper for identifying, classifying, and characterizing virus genomes from metagenomes
run_wo_reads Run the full wrapper for identifying, classifying, and characterizing virus genomes from metagenomes without metagenomic reads
download     Download and setup the ViWrap database
set_up_env   Set up the conda environments for all scripts
clean        Clean redundant information in each result directory

options:
-h, --help show this help message and exit
```

测试ViWrap run

```
(/slowdata/yml_environments/ViWrap) zhichao@sulfur:/storage1/data11/ViWrap$ ./ViWrap run -h
Run the full wrapper for identifying, classifying, and characterizing virus genomes from metagenomes

Usage: ViWrap run --input_metagenome <input metagenome assemblies> --input_reads <input metagenomic reads> --out_dir <output directory> [options]

Example 1: ViWrap run --input_metagenome /path/to/Lake_01_assemblies.fasta \
--input_reads /path/to/Lake_01_T1_1.fastq,/path/to/Lake_01_T1_2.fastq,/path/to/Lake_01_T2_1.fastq,/path/to/Lake_01_T2_2.fastq \
--out_dir ./ViWrap_Lake_01_outdir \
--identify_method vb-vs \
--conda_env_dir /path/to/ViWrap_conda_environments

Example 2: ViWrap run --input_metagenome /path/to/Lake_01_assemblies.fasta \
--input_reads /path/to/Lake_01_T1_1.fastq,/path/to/Lake_01_T1_2.fastq,/path/to/Lake_01_T2_1.fastq,/path/to/Lake_01_T2_2.fastq \
--out_dir ./ViWrap_Lake_01_outdir \
--db_dir /path/to/ViWrap_db \
--identify_method vb-vs \
--conda_env_dir /path/to/ViWrap_conda_environments \
--threads 30 \
--virome \
--input_length_limit 2000 \
--custom_MAGs_dir /path/to/custom_MAGs_dir
```

软件操作

跑ViWrap

```
(/slowdata/yml_environments/ViWrap) zhichao@sulfur:/storage1/data11/ViWrap$ vi run_ViWrap2.sh
```

```
python ViWrap run --input_metagenome /storage1/data11/ViWrap/Guaymas_scaffolds_min1000.subset.fasta \  
--input_reads /storage1/Reads/HydroPlume/Guaymas/Guaymas_final_reads.subset10perc_1.fastq,/storage1/Reads/HydroPlume/Guaymas/Guaymas_final_reads.subset10perc_2.fastq,/storag  
e1/Reads/HydroPlume/Guaymas/Guaymas_final_reads.subset15perc_1.fastq,/storage1/Reads/HydroPlume/Guaymas/Guaymas_final_reads.subset15perc_2.fastq \  
--out_dir ./ViWrap_outdir_vb_vs \  
--conda_env_dir /slowdata/yml_environments \  
--threads 20 \  
--input_length_limit 5000 \  
--db_dir /storage1/data11/ViWrap/ViWrap_db \  
--identify_method vb-vs \  
--custom_MAGs_dir /storage1/data11/ViWrap/Guaymas_bins
```

```
### Welcome to ViWrap ###
```

```
The issued command is:
```

```
/storage1/data11/ViWrap/ViWrap run --input_metagenome /storage1/data11/ViWrap/Guaymas_scaffolds_min1000.subset.fasta --input_reads /storage1/Reads/HydroPlume/Guaymas/Guaymas_final_reads.subset10perc_1.fastq,/storage1/Reads/HydroPlume/Guaymas/Guaymas_final_reads.subset10perc_2.fastq,/storage1/Reads/HydroPlume/Guaymas/Guaymas_final_reads.subset15perc_1.fastq,/storage1/Reads/HydroPlume/Guaymas/Guaymas_final_reads.subset15perc_2.fastq --out_dir ./ViWrap_outdir_vb_vs --db_dir /storage1/data11/ViWrap/ViWrap_db --identify_method vb-vs --conda_env_dir /slowdata/yml_environments --threads 20 --input_length_limit 5000 --custom_MAGs_dir /storage1/data11/ViWrap/Guaymas_bins
```

```
[2022-10-29 11:06:16] | Pre-check inputings. In processing...  
[2022-10-29 11:06:16] | Looks like the input metagenome and reads, database, and custom MAGs dir (if option used) are now set up well, start up to run ViWrap pipeline  
[2022-10-29 11:06:16] | Run VIBRANT-VirSorter2 method. Run VIBRANT to identify and annotate virus from input metagenome. In processing...  
[2022-10-29 11:21:16] | Run VIBRANT-VirSorter2 method. Run VIBRANT to identify and annotate viruses from input metagenome. Finished  
[2022-10-29 11:21:16] | Run VIBRANT-VirSorter2 method. Run VirSorter2 to identify viruses from input metagenome. Also plus CheckV to QC and trim, and KEGG, Pfam, and VOG HMMs to annotate viruses. In processing...  
[2022-10-29 15:04:36] | Run VIBRANT-VirSorter2 method. Run VirSorter2 the 1st time to identify viruses from input metagenome. Finished  
[2022-10-29 15:05:48] | Run VIBRANT-VirSorter2 method. Run CheckV the 1st time to QC and trim viruses identified from VirSorter2 1st run. Finished  
[2022-10-29 15:50:07] | Run VIBRANT-VirSorter2 method. Run VirSorter2 the 2nd time for CheckV-trimmed sequences. Finished  
[2022-10-29 15:51:39] | Run VIBRANT-VirSorter2 method. Run CheckV the 2nd time to get viral and host gene counts. Finished  
[2022-10-29 15:53:28] | Run VIBRANT-VirSorter2 method. Run VIBRANT to check "keep2" and "manual_check" groups and get the final VirSorter2 virus sequences. Finished  
[2022-10-29 15:53:28] | Map reads to metagenome. In processing...  
[2022-10-29 16:34:11] | Map reads to metagenome. Finished  
[2022-10-29 16:34:11] | Run vRhyne to bin viral scaffolds. In processing...  
[2022-10-29 16:35:45] | Run vRhyne to bin viral scaffolds. Finished  
[2022-10-29 16:35:45] | Run vContact2 to cluster viral genomes. In processing...  
[2022-10-29 17:32:22] | Run vContact2 to cluster viral genomes. Finished  
[2022-10-29 17:32:22] | Run CheckV to evaluate virus genome quality. In processing...  
[2022-10-29 17:36:40] | Run CheckV to evaluate virus genome quality. Finished  
[2022-10-29 17:36:40] | Run dRep to cluster virus species. In processing...  
[2022-10-29 17:36:47] | Run dRep to cluster virus species. Finished  
[2022-10-29 17:36:47] | Conduct taxonomic characterization. In processing...  
[2022-10-29 17:42:43] | Conduct taxonomic characterization. Finished  
[2022-10-29 17:42:43] | Conduct Host prediction by iPHoP. In processing...  
[2022-10-29 18:19:49] | Conduct Host prediction by iPHoP. Finished  
[2022-10-29 18:19:49] | Conduct Host prediction by iPHoP using custom MAGs. In processing...  
[2022-10-30 09:22:15] | Conduct Host prediction by iPHoP using custom MAGs. Finished  
[2022-10-30 09:22:17] | Get virus genome abundance. Finished  
[2022-10-30 09:22:17] | Get virus sequence information. Finished  
[2022-10-30 09:22:17] | Visualize the result. Finished
```

```
The total running time is 22:16:01 (in "hr:min:sec" format)
```

软件操作

结果文件夹

```
(/slowdata/yml_environments/ViWrap) zhichao@sulfur:/storage1/data11/ViWrap/ViWrap_outdir_vb_vs$ ls
00_VIBRANT_VirSorter_Guaymas_scaffolds_min1000.subset  02_vRhye_outdir      04_Nlinked_viral_gn_dir  06_dRep_outdir  08_ViWrap_summary_outdir  ViWrap_run.log
01_Mapping_result_outdir                            03_vConTACT2_outdir 05_CheckV_outdir        07_iPhoP_outdir 09_Virus_statistics_visualization
```

```
(/slowdata/yml_environments/ViWrap) zhichao@sulfur:/storage1/data11/ViWrap/ViWrap_outdir_vb_vs/08_ViWrap_summary_outdir$ ls
Genus_cluster_info.txt      Host_prediction_to_genus_m90.csv  Species_cluster_info.txt      Virus_genomes_files          Virus_raw_abundance.txt
Host_prediction_to_genome_m90.csv  Sample2read_info.txt              Tax_classification_result.txt  Virus_normalized_abundance.txt  Virus_summary_info.txt
```