Jun 10, 2024

Análise filogenômica

DOI
dx.doi.org/10.17504/protocols.io.q26g71mb1gwz/v1

Thiago Mafra Batista[1]

[1]Universidade Federal do Sul da Bahia

Thiago Mafra Batista
Universidade Federal do Sul da Bahia

OPEN ACCESS

DOI: dx.doi.org/10.17504/protocols.io.q26g71mb1gwz/v1

Protocol Citation: Thiago Mafra Batista 2024. Análise filogenômica. **protocols.io** **https://dx.doi.org/10.17504/protocols.io.q26g71mb1gwz/v1**

Protocol status: In development
We are still developing and optimizing this protocol

Created: June 07, 2024

Last Modified: June 10, 2024

Protocol Integer ID: 101431

Keywords: phylogenomics, IQTREE2, BUSCO

Abstract

This tutorial will guide students and researchers in constructing phylogenetic trees from genomic data. The step-by-step process includes data acquisition, assessment of genome completeness, identification of complete and single-copy orthologs present in the genomes, followed by alignment and trimming of the alignment. Phylogenetic inference will be performed using IQ-TREE2, and finally, the tree will be visualized and edited in iTOL.

Materials

Softwares utilizados neste tutorial:

1. BUSCO (https://busco.ezlab.org/)
2. BUSCO_phylogenomics (https://github.com/jamiemcg/BUSCO_phylogenomics)
3. iqtree2 (https://github.com/iqtree/iqtree2)

Scripts acessórios:
- genbank_assembly_downloader.py (https://github.com/thiagomaframg/bioinfo/blob/main/genbank_assembly_downloader.py)
- busco_run.py (https://github.com/thiagomaframg/bioinfo/blob/main/busco_run.py)

Aquisição dos genomas no formato fasta

1 A primeira etapa da construção da árvore filogenômica consiste em baixar os genomas que serão analisados. Para isso, é necessário visitar a pagina Genome do NCBI no link **https://www.ncbi.nlm.nih.gov/datasets/genome/**. Digite o nome das espécies de interesse e anote, em uma tabela ou planilha, os respectivos nomes e os identificadores do Genbank, que começam com GCA_. Essa tabela ou planilha será fornecida como material suplementar do artigo. Obs: lembre-se de incluir o genoma do outgroup da árvore.

Após anotar os identificadores de todos os genomas, copie e cole apenas os identificadores em um arquivo chamado ***accessions.txt.*** Um identificador por linha.

Feito isso, vamos rodar o script ***genbank_assembly_downloader.py*** para baixar os genomas cujo identificador está presente no arquivo accessions.txt

```
python genbank_assembly_downloader.py
```

O script irá criar um diretório chamado downloads e dentro dele serão salvos todos os arquivos *fasta* correspondentes a cada genoma. Os arquivos serão nomeados com seus respectivos códigos. Sugiro renomear estes arquivos com o nome das espécies como no exemplo:

```
mv GCA_030450195.1_ASM3045019v1_genomic.fna
Saccharomycopsis_praedatoria_UFMG-CM-Y6991_genomic.fna
```

Avaliação da Completude dos genomas

2 A construção da árvore será feita a partir do alinhamento múltiplo de todas as proteínas ortólogas de cópia simples (single-copies or 1:1) presentes em todos os genomas. Para identificá-las vamos utilizar o software **BUSCO**. Vamos rodá-lo de forma automatizada, com o script ***busco_run.py***.
Será necessário editar o script para ajustá-lo à sua realidade. Parâmetros como *busco_lineage*, *download_path*, *output_dir* e *--cpu*. Considerando que todos os arquivos fasta dos genomas estão no diretório atual de trabalho, rode:

```
python busco_run.py
```

Será criado um diretório de output contendo todos os resultados de todos os genomas. Cada diretório terá o nome das duas primeiras palavras dos arquivos fasta dos genomas. Exemplo: diretório Saccharomycops_praedatoria. Dentro de cada diretório contem um arquivo que começa com *short_summary.specific*.txt*. Precisamos destes arquivos.

```
mkdir busco_plot
```

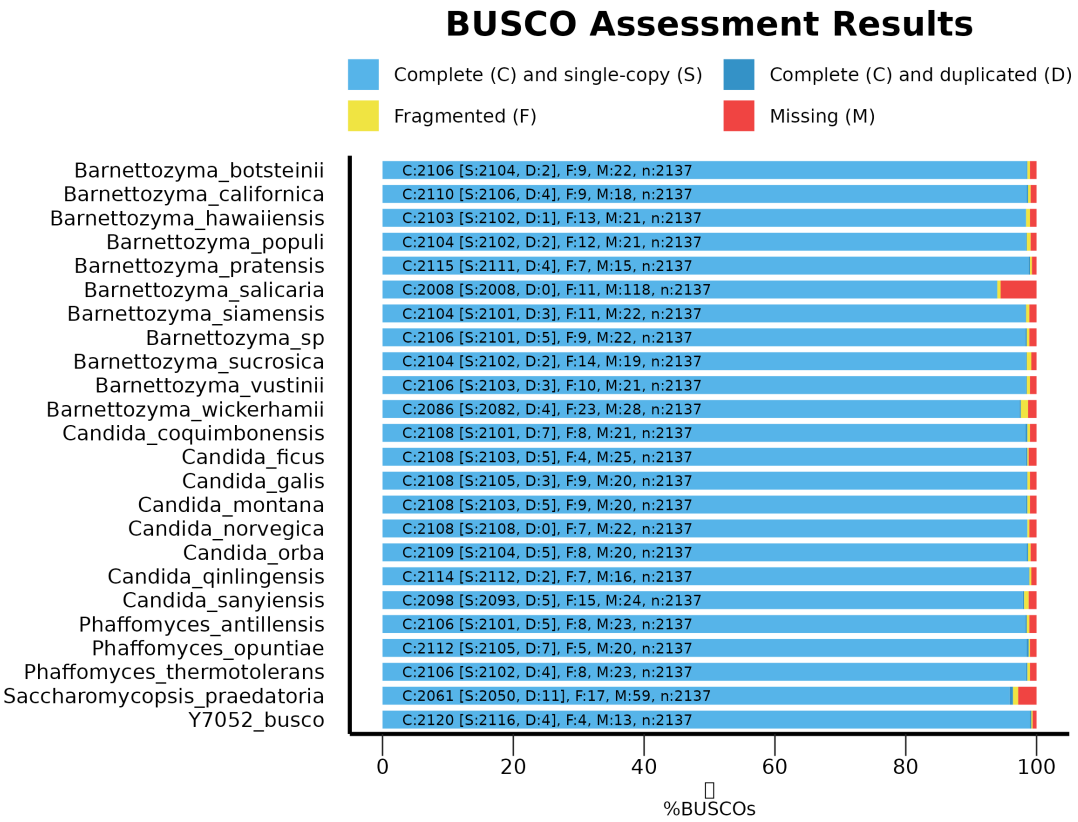
Agora copie todos os arquivos short_summary.specific*.txt para este diretório:

```
cp busco_outputs/*/short_summary.specific*.txt busco_plot
```

Vamos usar este diretório para criar uma imagem com todos os resultados com um script acessário do BUSCO chamado *generate_plot.py*. Esta imagem nos ajudará a avaliar a completude dos genomas e a decidir se é necessário excluir da análise aqueles que apresentam baixa completude.

```
$BUSCO/scripts/generate_plot.py -wd busco_plot/
```

Abaixo um exemplo deste resultado:



No exemplo acima, todos os genomas apresentam boa completude, embora Barnettozyma salicaria tenha mais ortólogos ausentes. Não é necessário excluir nenhum genoma da análise. Vamos seguir para a próxima etapa.

Identificação dos ortólogos 1:1 presentes em todos os genomas

3 Agora que já temos a avaliação da completude de todos os genomas, vamos identificar quais são os ortólogos de cópia única (single-copies ou 1:1) presentes nos genomas. Para isso vamos utilizar o script **BUSCO_phylogenomics.py**. Este script identifica proteínas BUSCO que são completas e de cópia única em todas as amostras de entrada. Alternativamente, é possível considerar os dados ausentes e optar por incluir proteínas BUSCO que sejam completas e de cópia única em uma determinada porcentagem dos genomas. Cada família BUSCO é individualmente alinhada, aparada e depois concatenada para gerar um alinhamento de supermatriz.

```
python ~/bin/BUSCO_phylogenomics/BUSCO_phylogenomics.py -i
busco_run/ -o busco_phylo -t 48 --supermatrix --gene_tree_program
iqtree
```

Dentro do diretório chamado busco_phylo serão salvos os arquivos necessários para a construção da árvore filogenômica.

Construção da árvore filogenômica

4 Agora temos os arquivos prontos para a inferência filogenética. O arquivo de input para o iqtree2 será o SUPERMATRIX.phylip

Sugiro que seja utilizado o screen ou tmux para rodar o iqtree2.

```
screen iqtree2 -s SUPERMATRIX.phylip -p SUPERMATRIX.partitions.nex
-m TESTMERGE --rclusterf 10 -B 1000 --sampling GENE -T AUTO
```

Serão gerados diversos arquivos de output. O principal arquivo será o SUPERMATRIX.partitions.nex.treefile

Este arquivo está estruturado no formato **NEXUS** e deve ser visualizado e editado no **iTOL**.