Script P3: Open Reading Frame Prediction

HANNIGAN GC, GRICE EA, ET AL

Abstract

This protocol provides a method for predicting the locations of the open reading frames (ORFs) using the Glimmer3 toolkit. Methods based on the publication:

Hannigan, Geoffrey D., et al. "The Human Skin Double-Stranded DNA Virome: Topographical and Temporal Diversity, Genetic Enrichment, and Dynamic Associations with the Host Microbiome." *mBio* 6.5 (2015): e01578-15.

Citation: HANNIGAN GC, GRICE EA, ET AL Script P3: Open Reading Frame Prediction. **protocols.io** dx.doi.org/10.17504/protocols.io.efpbbmn

Published: 10 Mar 2016

Guidelines

Required Software:

- Glimmer3 v3.02

Relevant Files

- Output: Virome_ORFs/Contigs_no_block_with_names_glimmer_output_final.fa

After the contigs were generated from the concatenated fasta files, we predicted the locations of the open reading frames (ORFs) and extracted them from the contigs using the Glimmer3 toolkit. First we generated the needed directories and copied the contig files over to a new directory.

We then built an Interpolated Context Model (ICM), used it to predict the open reading frames, and extracted the open reading frames from the contig sequences. Finally we removed the block fasta format of the ORFs because this can interfere with some of our downstream analyses.

Before start

Supplementary information available at:

https://figshare.com/articles/The_Human_Skin_dsDNA_Virome_Topographical_and_Temporal_Diversity_Genetic_Enrichment_and_Dynamic_Associations_with_the_Host_Microbiome/1281248

Protocol

Analysis

Step 1.

Generate the needed directories.

cmd COMMAND

```
mkdir ./glimmer3
mkdir ./glimmer3/contigs
mkdir ./glimmer3/output
```

Analysis

Step 2.

Copy the contigs file (non-block format) to a new directory.

cmd COMMAND

```
cp ./ray_contigs_from_total_cat_pairs/Contigs_no_block_with_names.fasta ./glimmer3/contigs
```

Analysis

Step 3.

Perform glimmer extraction.

SOFTWARE PACKAGE (Unix)

Glimmer3 Toolkit, 3.02

The Institute for Genomic Research (TIGR)

cmd COMMAND

```
build-icm ./glimmer3/output/Contigs_no_block_with_names.icm < ./glimmer3/contigs/Contigs_no_block_with_names.fasta

echo 'Running glimmer...'
glimmer3 -g 100 ./glimmer3/contigs/Contigs_no_block_with_names.fasta ./glimmer3/output/Contigs_no_block_with_names.icm ./glimmer3/output/Contigs_no_block_with_names_glimmer_output

cat ./glimmer3/output/Contigs_no_block_with_names_glimmer_output.predict | while read line;
    do if [ "${line:0:1}" == ">" ]
        then seqname=${line#'>'}
        else
        orf="$seqname.${line%%' '*}"
        coords="${line#*' '}"
        echo -e "$orf\t$seqname\t$coords"
        fi
    done > ./glimmer3/output/Contigs_no_block_with_names_glimmer_output.predict.formatted

echo 'Performing multi-extract...'
multi-extract -l 100 --nostop ./glimmer3/contigs/Contigs_no_block_with_names.fasta ./glimmer3/output/Contigs_no_block_with_names_glimmer_output.predict.formatted > ./glimmer3/output/Contigs_no_block_with_names_glimmer_output.genes
```

Analysis

Step 4.

Remove block fasta format.

cmd COMMAND

```
sed -r 's/\s/_/g' ./glimmer3/output/Contigs_no_block_with_names_glimmer_output.genes  | sed 's/^\([A,T,G,C,n]*\)$/\1\@/g' | sed ':a;N;$!ba;s/\@\n\([A,C,G,T,n]\)/\1/g' | sed 's/\@//g' > ./glimmer3/output/Contigs_no_block_with_names_glimmer_output_final.fa
```