Run Prodigal using iMicrobe

Version 2

Alise Ponsero[1]

[1]University of Arizona

dx.doi.org/10.17504/protocols.io.veje3cn

iMicrobe Metafunc course 2018

Alise Ponsero
University of Arizona

ABSTRACT

How to run Prodigal version 2.6.3 (Hyatt et al. 2010) through the iMicrobe plaform.

Prodigal is a protein-coding gene prediction software tool for bacterial and archaeal genomes. Prodigal runs smoothly on finished genomes, draft genomes, and metagenomes. Please note that Prodigal has not been tested on viruses, and that Prodigal contains no special rules or routines to handle viral genomes.

More informations about Prodigal can be found here : https://github.com/hyattpd/Prodigal/wiki

TAGS

metagenomics

imicrobe

Show tags

PROTOCOL STATUS

Working

We use this protocol in our group and it is working

Run prodigal on a single genome

1 *Note : This protocol uses as an example the bacterial isolate sample available in iMicrobe.*

After search and selection of the sample of interest, add the sample in the cart. In the 'tools' dropdown menu, select 'Apps'. You are presented the list of apps currently available on iMicrobe. Click on prodigal-2.6.3u2.

In the page app, provide the input files using the Cyverse datastore. Choose the following app parameters :

- **Select procedure :** single

Other parameters are available to the discretion of the user

- **write protein** : give the user an output fasta file with the protein translation
- **Closed ends :** Do not allow partial genes at the edges of sequence. Use this parameter if you have genomes where you are sure the first and last bases of the sequence(s) do not fall inside a gene.
- **Write nucleotides** : give the user an output fasta file with the nucleotide sequences
- **Output format** : Specify output format. The output formats are : genbank (Genbank-like format, the default output format); gff: GFF format; Simple coordinate output (this output is suitable only if the user only desires gene coordinates and nothing else.)
- **Treat runs of N as masked sequence :** By default, Prodigal's parameters are tuned for scaffolds and/or draft genomes with multiple contigs : Genes are allowed to run into gaps of N's. This parameter prevent prodigal to define genes running accross a gap sequence.
- **Write all potential genes (with scores) :** This option create an output file containing all the potential genes found by the tool with prediction score associated.

NOTE: This app uses the default translation table for prodigal. The tool uses the translation table 11 (bacteria and Archea), then table 4 (Mycoplasma/Spiroplasma)

After the job is effectively ran, you can access your results using the drop-down menu 'Tools' and selecting 'Jobs'. Select the job corresponding to your centrifuge run, and go to the section 'Outputs'.

The prodigal output files are now in your cyverse datastore. Click on 'Browse and view output files in the CyVerse Datastore'.

In the job folder created in the CyVerse datastore, the input fasta/fastq files are copied, along with the logs of the job (*.err and *.out). In order to retrieve your results go to the prodigal-out folder. It contains a folder by submitted genomes. This folder contains several output files :

prodigal.gbk/sco/gff
This file is the main Prodigal output file, which consists of gene coordinates and some metadata associated with each gene. By default, Prodigal produces a Genbank-like feature table; however, the user can specify some other output types (Simple Coordinate Output or Generic Feature Format Version 3).
Please note that the sco output will provide you with the predicted gene coordinates and nothing else.

For each individual sequence in the input file, Prodigal produces a semicolon-delimited string with information. In Genbank format, this is on a "DEFINITION" line. In GFF, this information is into the comment lines.
The fields in this header are as follows:

- **seqnum**: An ID for this sequence.
- **seqlen**: Number of bases in the sequence.
- **seqhdr**: The FASTA header line.
- **version**: Version of Prodigal used to analyze this sequence.
- **run_type**: "Ab initio" for single mode, "Anonymous" for meta mode.
- **model** *(meta mode only)*: Information about the preset training file used to analyze the sequence.
- **gc_cont**: % GC of the sequence.
- **transl_table**: The genetic code used to analyze the sequence.
- **uses_sd**: Set to 1 if Prodigal used its default RBS finder, 0 if it scanned for other motifs.

In addition, Prodigal produces a semicolon-delimited string with scoring and statistical information about each gene. In Genbank format, this is placed on a "note" line, in GFF the string is placed in the last field of the line.
The fields in this string are the following:

- **ID**: A unique identifier for each gene, consisting of the ID of the sequence and an ID of that gene within the sequence (separated by an underscore).
- **partial**: An indicator of if a gene runs off the edge of a sequence or into a gap. A "0" indicates the gene has a true boundary (a start or a stop), whereas a "1" indicates the gene is "unfinished" at that edge (i.e. a partial gene).
- **start_type**: The sequence of the start codon. Labeled "Edge" if the gene has no start codon.
- **stop_type**: The sequence of the stop codon. Labeled "Edge" if the gene has no stop codon.
- **rbs_motif**: The RBS motif found.
- **rbs_spacer**: The number of bases between the start codon and the observed motif.
- **gc_cont**: The GC content of the gene sequence.
- **gc_skew**: The GC skew of the gene sequence.
- **conf**: A confidence score for this gene, representing the probability that this gene is real.
- **score**: The total score for this gene.
- **cscore**: The hexamer coding portion of the score (how much this gene looks like a true protein).
- **sscore**: A score for the translation initiation site for this gene; it is the sum of the following three fields.
- **rscore**: A score for the RBS motif of this gene.
- **uscore**: A score for the sequence surrounding the start codon.
- **tscore**: A score for the start codon type.
- **mscore**: A score for the remaining signals.

nucl.fa
The nucleotide output file consists of all the predicted genes in multiple FASTA format. The FASTA header contains several fields separated by # sign. The first field is the gene ID. Then the leftmost coordinate in the genome, the rightmost coordinate, and the strand (1 for forward strand genes, -1 for reverse strand genes). Following the coordinate information is a semicolon-delimited string using the following fields: **ID**, **partial**, **start_type**, **stop_type**, **rbs_motif**, **rbs_spacer**, **gc_cont**, and **gc_skew**, and **conf**. (see description for these fields above).

protein.fa
The protein output file consists of all the predicted genes in multiple FASTA format. The FASTA header contains several fields separated by # sign. The first field is the gene ID. Then the leftmost coordinate in the genome, the rightmost coordinate, and the strand (1 for forward strand genes, -1 for reverse strand genes). Following the coordinate information is a semicolon-delimited string using the following fields: **ID**, **partial**, **start_type**, **stop_type**, **rbs_motif**, **rbs_spacer**, **gc_cont**, and **gc_skew**, and **conf**. (see description for these fields above).

genes.txt

This output files recapitulates all the predicted genes and their scoren before any score filtering. Thus this file contains numerous predicted genes removed from the other prodigal outputs because of their low prediction score.

The header recapitulates the informations concerning the run :

- **seqnum**: An ID for this sequence.
- **seqlen**: Number of bases in the sequence.
- **seqhdr**: The FASTA header line.
- **version**: Version of Prodigal used to analyze this sequence.
- **run_type**: "Ab initio" for single mode, "Anonymous" for meta mode.
- **model** *(meta mode only)*: Information about the preset training file used to analyze the sequence.
- **gc_cont**: % GC of the sequence.
- **transl_table**: The genetic code used to analyze the sequence.
- **uses_sd**: Set to 1 if Prodigal used its default RBS finder, 0 if it scanned for other motifs.

The rest of the file is a tab delimited table with the following fields :

- **Beg** : start position in the sequence.
- **End** : End position in the sequence
- **Std** : strand + or -
- **Total** : The total score for this gene.
- **CodPot** : The hexamer coding portion of the score
- **StrtSc** : A score for the translation initiation site for this gene.
- **Codon** : The sequence of the start codon. Labeled "Edge" if the gene has no start codon.
- **RBSMot** : The RBS motif found.
- **Spacer** : The number of bases between the start codon and the observed motif.
- **RBSScr** : A score for the RBS motif of this gene.
- **UpsScr** : A score for the sequence surrounding the start codon.
- **TypeScr** : A score for the start codon type.
- **GCCont** : The GC content of the gene sequence.

EXPECTED RESULT

Results obtained on the exemple set (), run with the following parameters :
output format : gbk
procedure : single
write nucl : true
write proteins: true
write genes: true

genes.txt
nucl.fa
prodigal.gbk
proteins.fa

Run prodigal on an assembly

2 *Note : This protocol uses as an example the sample SRS142975 from the HMP project. This WGS of the throat of a female subject was assembled by the HMP project. The assembly is available here.*

After search and selection of the sample of interest, add the sample in the cart. In the 'tools' dropdown menu, select 'Apps'. You are presented the list of apps currently available on iMicrobe. Click on prodigal-2.6.3u2.

In the page app, provide the input files using the Cyverse datastore. Choose the following app parameters :

- **Select procedure :** meta

Other parameters are available to the discretion of the user

- **write protein** : give the user an output fasta file with the protein translation
- **Closed ends :** Do not allow partial genes at the edges of sequence. Use this parameter if you have genomes where you are sure the first and last bases of the sequence(s) do not fall inside a gene.
- **Write nucleotides** : give the user an output fasta file with the nucleotide sequences
- **Output format** : Specify output format. The output formats are : genbank (Genbank-like format, the default output format); gff: GFF format; Simple coordinate output (this output is suitable only if the user only desires gene coordinates and nothing else.)
- **Treat runs of N as masked sequence :** By default, Prodigal's parameters are tuned for scaffolds and/or draft genomes with multiple contigs : Genes are allowed to run into gaps of N's. This parameter prevent prodigal to define genes running accross a gap sequence.
- **Write all potential genes (with scores) :** This option create an output file containing all the potential genes found by the tool with prediction score associated.

After the job is effectively ran, you can access your results using the drop-down menu 'Tools' and selecting 'Jobs'. Select the job corresponding to your centrifuge run, and go to the section 'Outputs'.

The prodigal output files are now in your cyverse datastore. Click on 'Browse and view output files in the CyVerse Datastore'.

In the job folder created in the CyVerse datastore, the input fasta/fastq files are copied, along with the logs of the job (*.err and *.out). In order to retrieve your results go to the prodigal-out folder. It contains a folder by submitted genomes. This folder contains several output files described in step 1.

EXPECTED RESULT

Results obtained on the exemple set (SRS142975) run with the following parameters :
output format : gbk
procedure : meta
write genes : true
write proteins : true
write nucl: true

genes.txt
nucl.fa
prodigal.gbk
proteins.fa

This is an open access protocol distributed under the terms of the Creative Commons Attribution License (https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited