FEELnc : Fast and Effective Extraction of Long non-coding RNAs

Documentation

1. Introduction

This document is intended to give a (minimal) description of the FEELnc pipeline in order to annotate long non-coding RNAs (lncRNA).

Currently, FEELnc is composed of 3 modules (See `3- Launch FEELnc pipeline` for more details):

- `FEELnc_filter.pl` : Extract, filter candidate transcripts
- `FEELnc_codpot.pl` : Compute the coding potential of candidate transcripts
- `FEELnc_classifier.pl`: Classify lncRNAs based on their genomic localization wrt mRNAs

To get help on each of this module, type for instance:

```
FEELnc_filter.pl --help
```

Or

```
FEELnc_filter.pl --man
```

The main format to describe genes, transcripts, exon is GTF and FASTA for genome file. Basically, FEELnc users should have the following minimal input files.

- `Infile.GTF` : input GTF file (e.g cufflinks transcripts.GTF)
- ref`_annotation.GTF` : GTF annotation file**
- `ref_genome.FA` : genome FASTA file or directory with individual chrom FASTA files

** It is recommended to extract protein_coding transcripts (mRNAs) from the reference annotation file (`ref_annotation.GTF`) either manually or by using the option : `-b transcript_biotype=protein_coding` (see below)

2. Installation and requirements

a. Requirements

The following software must be installed on your machine:

- Perl5+ : tested with version `5.18.2`
    - Bioperl : tested with version `BioPerl-1.6.924`
    - Parralell::ForkManager: tested with version `1.06`
- R `(Rscript) :` tested with version `3.1.0`
    - ROCR `R library (type "install.packages('ROCR')" in a R session)`
- CPAT tool : Coding Potential Assessment Tool (tested with version 1.2.2)

b. Installation

Download and extract FEELnc archive:

```
tar xzvf FEELnc_XXX.tar.gz
```

Go to FEELnc directory

```
cd FEELnc_XXX
```

Install FEELnc

```
perl Makefile.PL PREFIX=my_dir_of_install/
```

```
make
```

```
make install
```

export PERL5LIB, FEELNC PATH and add it to your PATH

```
export PERL5LIB=$PERL5LIB:my_dir_of_install/
export FEELNCPATH=${PWD}
export PATH=$PATH:$FEELNCPATH/scripts/
```

Test if everything is ok with toy example:

```
cd test/
perl ../scripts/FEELnc_filter.pl -i transcript_chr38.gtf -a annotation_chr38.gtf -b
transcript_biotype=protein_coding > candidate_lncRNA.gtf
perl ../scripts/FEELnc_codpot.pl        -i candidate_lncRNA.gtf                -a
annotation_chr38.gtf -g genome_chr38.fa -n 190 -v 20
perl ../scripts/FEELnc_classifier.pl -i candidate_lncRNA.gtf.lncRNA.gtf -a
annotation_chr38.gtf >  candidate_lncRNA_classes.txt
```

3. Launch FEELnc pipeline

a. FEELnc_filter.pl

The first step of the pipeline (FEELnc_filter) consists in filtering out unwanted/spurious transcripts and/or transcripts overlapping (in sense) exons of the reference annotation and especially protein_coding exons as they more probably correspond to new mRNA isoforms (see -b,--biotype option).

Usage:

```
    FEELnc_filter.pl -i infile.gtf -a annotation_mRNA.gtf >
candidate_lncRNA.gtf
```

If your annotation contains transcript_biotype information (e.g protein_coding, pseudogene, miRNA...), you can subselect a specific transcript biotype to make the overlap with.

```
    FEELnc_filter.pl -i infile.gtf \
-a annotation_mRNA.gtf \
-b transcript_biotype=protein_coding \
> candidate_lncRNA.gtf
```

This option is highly recommended if you don't want to remove transcript overlapping with other transcript than mRNAs (e.g lincRNA, miRNA, pseudogene...)

b. FEELnc_codpot.pl

The second step of the pipeline (FEELnc_codpot) aims at computing the CPS i.e the coding potential score (between [0-1]) foreach of the candidate transcripts in the `candidate_lncRNA.gtf` file.
It makes use of the CPAT tool which is an alignment-free program (so fast) which relies on intrinsic properties of the fasta sequences of two training files:

- `known_mRNA.gtf` : a set of known protein_coding transcripts
- `known_lncRNA.gtf` : a set of known lncRNA transcripts

However, for most organisms, the set of known_lncRNA transcripts is not known and thus a set of genomic intergenic regions are automatically extracted as the lncRNA training set. In this case, the reference genome file is required (`ref_genome.FA`)

Usage:

```
    FEELnc_codpot.pl -i candidate_lncRNA.gtf -a known_mRNA.gtf -g
ref_genome.FA
```

To calculate the **CPS cutoff** separating coding (mRNAs) versus long non-coding RNAs (lncRNAs), `FEELnc_codpot` uses a R script that will make a 10 fold cross-validation on the input training files and finally, extracts the CPS that maximizes sensitivity (Sn) and Specificity (Sp) (thanks to the ROCR library)

Output:

Let's say your input file is called INPUT, this second module will create 4 output files

- `INPUT.cpat`: gathering all CPAT metric together with the CPS for all input tx
- `INPUT.Cutoff.png`: a .png image of the Two Graphic ROC curves to determine the optimal cutoff value.
- `INPUT.lncRNA.gtf` : a .GTF file of the transcripts below the CPS (Your final set of lncRNAs)
- `INPUT.mRNA.gtf` : a .GTF file of the transcripts above the CPS (a *a priori* new set of mRNAs)

c. `FEELnc_classifier.pl`

The last step of the pipeline consists in classifying new lncRNAs w.r.t to the annotation of mRNAs in order to annotate :

- **Intergenic lncRNAs i.e lincRNAs**
  - divergent : when the lincRNA is transcribed in an opposite direction (head to head) w.r.t to the closest mRNA
  - convergent: when the lincRNA is transcribed in a convergent direction w.r.t to the closest mRNA
  - same_strand: when the lincRNA is transcribed in a same starnd w.r.t to the closest mRNA

- **Genic lncRNAs:** lncRNAs overlapping mRNAs either
  - *Exonic:*
    - antisense : at least one lncRNA exon overlaps in antisense an mRNA exon
    - sense : there should not be since there are filtered in the first step
  - *Intronic:*
    - *antisense :* lncRNA exon overlaps in antisense mRNA introns (but none exons)
    - *sense :* lncRNA exon overlaps in sense mRNA introns (but none exons)
  - *Containing:*
    - *antisense :* lncRNA intron overlaps antisense mRNA exons
    - *sense :* lncRNA intron overlaps sense mRNA exons

```
Usage:
    FEELnc_classifier.pl –i lncRNA.gtf –a mRNA.gtf > lncRNA_classes.txt
```