Code: Discovery of regulatory motifs in Labyrinthulomycete genomes

Joshua Rest, Jackie Collier

Abstract

To develop broadly useful methods for the genetic manipulation of Labyrinthulomycetes, it is essential to understand the similarities and differences in regulation of gene expression among them. Toward this end we have used FIMO from the MEME suite (http://meme-suite.org/doc/fimo.html) to identify motifs similar to yeast transcription factor binding sites in each of the three available genome sequences: *Aplanochytrium kerguelense* PBS07, *Schizochytrium aggregatum* ATCC 28209, and *Aurantiochytrium limacinum* ATCC MYA-1381. We then make logograms to illustrate yeast-like GAL4 binding sites in each of the three genomes.

Citation: Joshua Rest, Jackie Collier Code: Discovery of regulatory motifs in Labyrinthulomycete genomes. **protocols.io** dx.doi.org/10.17504/protocols.io.hg8b3zw
Published: 29 Mar 2017

Protocol

Obtain genome data

Step 1.

Schag1_AssemblyScaffolds.fasta from
http://genome.jgi.doe.gov/pages/dynamicOrganismDownload.jsf?organism=Schag1

Aurli1_AssemblyScaffolds.fasta from
http://genome.jgi.doe.gov/pages/dynamicOrganismDownload.jsf?organism=Aurli1

Aplke1_AssemblyScaffolds.fasta from
http://genome.jgi.doe.gov/pages/dynamicOrganismDownload.jsf?organism=Aplke1

(in further steps, any such genome data will be called genome.fasta)

Obtain Position Weight Matrices

Step 2.

We scanned the genomes using Position Weight Matrices for putative transcription factor binding sties defined for budding yeast, *Saccharomyces cerevisae.*

We used motif file SwissRegulon_s_cer.meme obtained from http://meme-suite.org/doc/download.html

Optional step: Create a genome file in samtools

Step 3.

This creates an index file, and then a genome_File using samtools.

note: if there are blank lines in the fasta file, remove them (e.g. in vim use :g/^$/d). Otherwise, samtools will produce an error.

cmd COMMAND

```
samtools faidx genome.fasta
awk -v OFS='\t' {'print $1,$2'} genome.fasta.fai > genomeFile.txt
```

samtools 0.1.18

Optional step: Convert GFF annotations to .bed format

Step 4.

Gff annotation files were obtained from

Schag1_GeneCatalog_genes_20121220.gff from
http://genome.jgi.doe.gov/pages/dynamicOrganismDownload.jsf?organism=Schag1

Aurli1_GeneCatalog_genes_20120618.gff from
http://genome.jgi.doe.gov/pages/dynamicOrganismDownload.jsf?organism=Aurli1

Aplke1_GeneCatalog_genes_20121220.gff from
http://genome.jgi.doe.gov/pages/dynamicOrganismDownload.jsf?organism=Aplke1

In further steps, any such genome annotation data will be called GeneCatalog_genes.gff or GeneCatalog_genes.bed

cmd COMMAND (centos-release-6-8.el6.centos.12.3.x86_64)

```
gff2bed < GeneCatalog_genes.gff >GeneCatalog_genes.bed
```

gff2bed is part of BEDOPS v2.4.25

Retrieve promoter sequences 1kb upstream of coding sequences

Step 5.

Use bedtools to retrieve flanking sequence 1kb upstream of the ATG.

In these genomes, we can grep the bed file for annotations that contain Exon Number 1, which includes the ATG. This is likely to differ according to the details of your genome annotation.

Partial genes at the beginning of a contig will create annotations from 0 to 0; these should be removed.

cmd COMMAND (centos-release-6-8.el6.centos.12.3.x86_64)

```
bedtools flank -i GeneCatalog.bed -g genomeFile.txt -l 1000 -r 0 -
s > GeneCatalog_genes_1kbupstream.bed
grep "exonNumber 1$" GeneCatalog_genes_1kbupstream.bed >GeneCatalog_genes_1kbupstreamExon1.
bed

#optional: remove annotations that go from 0 to 0
grep -v -
```

```
P '0\t0\t.\t.' GeneCatalog_genes_1kbupstreamExon1.bed >GeneCatalog_genes_1kbupstreamExon1no
0.bed

bedtools getfasta -fi genome.fasta -bed GeneCatalog_genes_1kbupstreamExon1no0.bed -
fo GeneCatalog_genes_1kbupstreamExon1.bed.fa
```

Produced using bedtools 2.15.0

Scan for motifs

Step 6.

We used fimo from meme_4.11.1 (http://meme-suite.org/doc/download.html) to scan each genome for motif matches.

The output from this scan for these three genomes is available at:
http://commons.library.stonybrook.edu/inter-data/1/

cmd COMMAND (centos-release-6-8.el6.centos.12.3.x86_64)

```
fimo  --
o output_directory  SwissRegulon_s_cer.meme GeneCatalog_genes_1kbupstreamExon1.bed.fa
```

R: Make a logogram for discovered motifs for a TF of interest

Step 7.

Here, we demonstrate how to make a logogram for matches to the GAL4 PWM within R.

The resulting logograms are shown and described at
https://you.stonybrook.edu/labyrinthulomycetes/regulatory-element-discovery-in-labyrinthulomycete-genomes/

cmd COMMAND (centos-release-6-8.el6.centos.12.3.x86_64)

```
library(seqLogo)
library(Biostrings)
library(data.table)
aur1 <- fread("output_directory/fimo.txt")
aur2 <- aur1[ which(aur1$pattern=='GAL4'), ]
aurGAL4 <- DNAStringSet(aur2$match)
pfm <- consensusMatrix(aurGAL4)
pfm2 <- pfm[1:4,]
pfm3 <- prop.table(pfm2,2)
aurGAL4pwm <- makePWM(pfm3)

pdf("GAL4_logo.pdf")
seqLogo(aurGAL4pwm)
dev.off()
```

R version 3.3.2 (2016-10-31) [1] data.table_1.10.0 Biostrings_2.42.1 XVector_0.14.0 [4] IRanges_2.8.1 S4Vectors_0.12.1 BiocGenerics_0.20.0 [7] seqLogo_1.40.0 fimo.txt is the output from fimo in step 6.