Labyrinthulomycete genome codon usage calculation code

Joshua Rest, Jackie Collier

Abstract

We analyzed the recently available whole genome sequences from two thraustochytrids (Aurantiochytrium limacinum ATCC MYA-1381, Schizochytrium aggregatum ATCC 28209) and one aplanochytrid (Aplanochytrium PBS07) We then calculated the genome-wide relative synonymous codon usage, codon frequencies and GC content for predicted coding sequences from each of the three species. We compared these to other stramenopiles: the diatoms Phaeodactylum tricornutum and Thalassiosira pseudonana, and the oomycete Phytophthora sojae, as well as to the ascomycete fungus Saccharomyces cerevisiae. See this page for further description.
This code was run in R version 3.3.2 (2016-10-31)

Package info: RCurl_1.95-4.8 bitops_1.0-6 ape_4.0 reshape_0.8.6 seqinr_3.3-3

Citation: Joshua Rest, Jackie Collier Labyrinthulomycete genome codon usage calculation code. **protocols.io**
dx.doi.org/10.17504/protocols.io.hgyb3xw
Published: 29 Mar 2017

Protocol

Coding sequences from Labyrinthulomycete and other genomes

Step 1.

Coding sequences were downloaded from the following files / URLs:

Schag1_GeneCatalog_CDS_20121220.fasta from
http://genome.jgi.doe.gov/pages/dynamicOrganismDownload.jsf?organism=Schag1

Aurli1_GeneCatalog_CDS_20120618.fasta from
http://genome.jgi.doe.gov/pages/dynamicOrganismDownload.jsf?organism=Aurli1

Aplke1_GeneCatalog_CDS_20121220.fasta from
http://genome.jgi.doe.gov/pages/dynamicOrganismDownload.jsf?organism=Aplke1

Physo3_GeneCatalog_CDS_20110401.fasta from
http://genome.jgi.doe.gov/pages/dynamicOrganismDownload.jsf?organism=Physo3

Thalassiosira_pseudonana.ASM14940v1.30.cds.all.fa from
ftp://ftp.ensemblgenomes.org/pub/protists/release-30/fasta/thalassiosira_pseudonana/cds/

Phaeodactylum_tricornutum.ASM15095v2.30.cds.all.fa from ftp://ftp.ensemblgenomes.org/pub/protists/release-30/fasta/phaeodactylum_tricornutum/cds/

Saccharomyces_cerevisiae.R64-1-1.30.cds.all.fa

from ftp://ftp.ensemblgenomes.org/pub/release-30/fungi/fasta/saccharomyces_cerevisiae/cds/

DATASET

Coding Sequences

R: Prepare the workspace

Step 2.

Load libraries; define file names of coding sequences to be loaded

cmd COMMAND

```
library(seqinr)
library(reshape)
library(ape)
library(RCurl)
eval( expr = parse( text = getURL("https://raw.githubusercontent.com/talgalili/R-code-snippets/master/boxplot.with.outlier.label.r")))

allcds <-
 c("Aplke1_GeneCatalog_CDS_20121220.fasta","Aurli1_GeneCatalog_CDS_20120618.fasta","Phaeodactylum_tricornutum.ASM15095v2.30.cds.all.fa","Physo3_GeneCatalog_CDS_20110401.fasta","Saccharomyces_cerevisiae.R64-1-1.30.cds.all.fa","Schag1_GeneCatalog_CDS_20121220.fasta","Thalassiosira_pseudonana.ASM14940v1.30.cds.all.fa")
```

R: Calculate GC content for each species

Step 3.

Create a vector of GC content values of the coding sequences - one for each genome.

Output example: Table 2

cmd COMMAND

```
gwGC <- lapply(allcds,function(species){
print(species)
jgi1 <- read.fasta(species)
jgi1b <- unlist(jgi1)
jgi3b <- GC(jgi1b)
return(jgi3b)
})

gwGC2 <- do.call("rbind",gwGC)
rownames(gwGC2) <-
 c("Aplanochytrium kerguelense","Aurantiochytrium limacinum","Phaeodactylum tricornutum","Phytophthora sojae","Saccharomyces cerevisiae","Schizochytrium aggregatum","Thalassiosira pseudonana")
```

R: Calculate codon usage frequency of rscu use across all coding sequences in each genome

Step 4.

Calculate frequency or rscu of the 64 codon triplets across all genes in each genome.

cmd COMMAND

```
metric <- "freq"  #freq or rscu
gwRscu <- lapply(allcds,function(species){
print(species)
jgi1 <- read.fasta(species)
jgi1b <- unlist(jgi1)
jgi3b <- uco(jgi1b, index = metric)
return(jgi3b)
})

gwRscu2 <- do.call("rbind",gwRscu)
rownames(gwRscu2) <-
 c("Aplanochytrium kerguelense","Aurantiochytrium limacinum","Phaeodactylum tricornutum","Phytophthora sojae","Saccharomyces cerevisiae","Schizochytrium aggregatum","Thalassiosira pseudonana")
gwRscu2b <- gwRscu2
rownames(gwRscu2b) <- c("Ak","Al","Pt","Ps","Sc","Sa","Tp")

save(gwRscu2,file=paste("gwRscu2",metric,"rda",sep="."))
save(gwRscu2b,file=paste("gwRscu2b",metric,"rda",sep="."))
```

R: Reformat and plot the results.

Step 5.

Make a boxplot of RSCU or codon frequency across genomes, where each column is a codon, with outliers labelled

Output example.

cmd COMMAND

```
#re-format
gwRscu3b <- melt(gwRscu2b)
save(gwRscu3b,file=paste("gwRscu3b",metric,"rda",sep="."))

png(paste("Boxgw2",metric,"png",sep="."))
par(family="mono")
boxplot.with.outlier.label(gwRscu3b$value~gwRscu3b$X2,gwRscu3b$X1,las=2, cex.axis = 0.7, xlab="codon",ylab=metric)
points(gwRscu3$X2,gwRscu3$value,cex=0.5,col=gwRscu3$X1)
savefont <- par(font=3)
legend("topright",legend=unique(gwRscu3$X1),col=unlist(subset(gwRscu3,X2=="aaa",select=X1)),pch=1,cex=0.7)
par(savefont)
dev.off()
```