Genomic Exploration of Streptococcus Phylogeny

Jordyn Brooks

Intro to Applied Genomics

30 November 2023

Objective

The purpose of this project is to conduct a comprehensive analysis of the evolutionary relationship among a set of 15 Streptococcus species, utilizing various phylogenetic methods. This involves the construction of concatenated phylogenies, gene tree consensus phylogenies, and gene content trees, employing sophisticated phylogenetic analysis techniques. The inclusion of Streptococcus suis as an out-group for rooting the phylogenies is crucial in understanding the broader evolutionary context within the Streptococcus genus.

Methods

Obtaining Strain Data

FASTA files for the 15 Streptococcus species were acquired from the National Center for Biotechnology Information (NCBI) website. This process involved querying each strain using its respective accession number and retrieving available genome sequences in FASTA format, then uploading them onto Clearance (/home/jordyn/project/data/) utilizing the MobaXTerm user interface. However, the designated outgroup, S. suis, posed an exception as its RefSeq record had been removed from the NCBI website, therefore, it's outdated version's FASTA file, accessible via accession number NC_009442.1, was utilized for analysis. The decision to forgo assembly and rather start with annotation of the species samples was driven by a time-saving strategy, considering the generally favorable quality of the existing online FASTA files.

Annotation

For annotation of the Streptococcus sequences, a combination of perl scripting and the Prokka annotation pipeline was employed. Initially, headers within the FASTA files were formatted to adhere to Prokka's character requirements using a Perl one-liner that retained only the necessary sequence identifiers: perl -pe 's/^(\S+).*/$1/' -i *.fna

Following this adjustment, the Prokka annotation pipeline was implemented using a modified version of the Bash script (prokka.sh) provided for us.

prokka.sh: /home/jordyn/project/data/

```
#!/bin/bash
for file1 in *.fna
do

#replace .fna with nothing
prefix=${file1/.fna/}
#replace .fna with nothing
tag=${file1/.fna/}

/home/bio4030/opt/prokka/bin/prokka --prefix $prefix --increment 5 --locustag $tag --cpus 4
$file1

done
```

Gene Clustering

After genome annotation, gene clustering was performed in order to identify putative orthologous groups using a combination of file manipulation and clustering algorithms. GenBank (gbk) files containing the annotated genomes were relocated to a new folder ('/home/jordyn/project/data/vsearch') using the following commands:

```
cp ~/project/data/NC*/*.gbk ~/project/data/vsearch/
cp ~/project/data/A*/*.gbk ~/project/data/vsearch/
```

Additionally, the gb2cds.sh bash script provided in class was utilized to extract coding sequences from the GenBank files and prepare them for subsequent clustering analysis:

gb2cds.sh: /home/jordyn/project/data/vsearch/

```
#!/bin/bash
for file1 in *.gbk
do
#output prefix is *.gbk file name with .gbk removed
out=${file1/.gbk/}
gb2cds.pl -dna -orthomcl2 $file1 $out
done
```

To prepare the data for gene clustering using VSEARCH the extracted coding sequences were concatenated into a single file ('all.fnn'). This file was then used to employ the VSEARCH tool based on a specific sequence identity threshold (0.05) and parallel processing using multiple threads. The output of this program includes information on clustered sequences, OUT assignments, and logs capturing the clustering process. Thus, providing information into the genomic organization and potential functional relationships among the annotated sequences.

Running Vsearch: /home/jordyn/project/data/vsearch/

```
#concatenate all sequences into single file for Vsearch
cat *.fasta > all.fnn

#run Vsearch
vsearch --cluster_fast all.fnn --otutabout otu.tab --uc uc --id 0.5 --threads 4 --clusters c --log
vsearchlog
```

Concatened Tree Generation

The OTU table generated from VSEARCH clustering is utilized to help identify specific single-copy core clusters based on the presence of designated genes across the 15 samples. Before this could be done the space included in the out.tab file header needed to replace with an underscore using the following perl command: perl -p -i -e 's/#OTU ID/OTU_ID/g' otu.tab

Following this alteration, the R script ('single-copy-core.R') was revised in order to include the species specific accession numbers. The script helps to identify single-copy core clusters by reading in the OUT table and checking to see if each column contains a value of '1'

by using logical vector y, samples which are true for this condition are added to a new data frame called ‘core’ which gets exported to a file names ‘single-copy-core’.

single-copy-core.R: /home/jordyn/project/data/vsearch/

```
#!/usr/bin/env Rscript

tab=read.table("otu.tab",h=T)

y=tab[,"ADME01"]==1 & tab[,"AEQR01"]==1 & tab[,"AEVC01"]==1 &
tab[,"AEVD01"]==1 & tab[,"AEVF01"]==1 & tab[,"AFUP01"]==1 & tab[,"AFXO01"]==1 &
tab[,"NC_009009"]==1 & tab[,"NC_009442"]==1 & tab[,"NC_009785"]==1 &
tab[,"NC_013853"]==1 & tab[,"NC_014498"]==1 & tab[,"NC_015291"]==1 &
tab[,"NC_015678"]==1 & tab[,"NC_015875"]==1

core=tab[y,]

write.table(core, "single-copy-core", quote=F, sep="\t", col.names=F, row.names=F)
```

The identified single-copy core clusters were extracted for further analysis. Specific cluster IDs were isolated and transferred to a designated alignment folder (‘/home/jordyn/ project/data/align/’) for sequence alignment using MUSCLE (‘muscle.sh’).

Cut out tags: /home/jordyn/project/data/vsearch/

```
cut -f 1 single-copy-core > single-copy-core-tags

grep -w -f single-copy-core-tags c* > single-copy-core-cluster-ids

perl -p -i -e 's/:>/\t/g' single-copy-core-cluster-ids

#copy clusters IDs over to new folder “align”
cp $(cat single-copy-core-cluster-ids2) ~/project/data/align
```

Muscle.sh: /home/jordyn/project/data/align/

```
#!/bin/bash
#loop over clusters using muscle
for file1 in c*
do
#add .fnn to the value of file1
out=${file1}.fnn
muscle -in $file1 -out $out
done
```

Post-alignment, header standardization procedures were applied to ensure consistency across sequences. Specifically tailored Perl commands were executed to refine headers within the aligned sequence, followed by the concatenation for subsequent phylogenetic tree generation and the use of PhyML software to construct a phylogenetic tree from the concatenated alignment data set.

Fixing headers: /home/jordyn/project/data/align/

```
#For files starting with A
perl -p -i -e 's/\|A[A-Z]{3}[0-9]+_[0-9]+//g' *.fnn

#For files starting with NC_
perl -p -i -e 's/\|NC_[0-9]+_[0-9]+//g' *.fnn
```

Concatenation Tree Generation: /home/jordyn/project/data/align/

```
#make list of alignments
ls *.fnn > fnn.list

#remove .fnn
perl -p -i -e 's/\.fnn//g' fnn.list

#run script
cat_fas.pl fnn.list all.fnn .fnn

#Get stats on the concatenated alignment
#run script.
fasta-stats.py all.fnn > stats.txt

#Create new .fnn list
ls *.fnn > fnn.list2

#Convert format using fas2phy.pl and list
fas2phy.pl -l fnn.list2

#Phylogenetic tree construction method: (phyml)
#Chose to keep same option choices as used in class
phyml -i all.fnn.phylip -m GTR -f e -a e

#Downloaded output tree file, all.fnn.phylip_phyml_tree.txt, into figtree to download
```

Consensus Tree Generation

In this step, individual trees were constructed by specifying a bootstrap value of 100 for PhyML using the '-b' option. These trees were generated for each gene, and subsequently processed to generate a consensus tree that integrated information from these separate trees. The process involved running PhyML for each gene to create individual gene trees, each with a bootstrap value of 100, and then merging these trees to produce a consensus representation. This comprehensive overview of evolutionary relationships was based on multiple genes, providing a more robust understanding of the evolutionary landscape.

Consensus Tree Generation: /home/jordyn/project/data/align/

```
#consensus (separate gene trees)
#exclude concatenation alignment - change extension
mv all.fnn.phylip all.fnn.phylip2

#run phyml on each alignment using a loop
for i in *.phylip; do phyml -i $i -m GTR -b 100 -f e -a e; done

#concatenate all gene tree files (Newick files)
#but first, rename concatenated tree
mv all.fnn.phylip_phyml_tree.txt all.fnn.phylip_phyml_tree.txt2
cat *tree.txt > all.fnn.phylip_phyml_gene_trees.txt

#rename gene tree file to "intree"
mv all.fnn.phylip_phyml_gene_trees.txt intree

#Call phylip package and the consense package to generate the gene consensus tree
phylip consense < /home/bio4030/bin/infile > out

#Download "outtree" file and open in FigTree to visualize
```

Building the Gene Content Tree

Construction of a gene-content tree was completed by assessing similarities in gene presence/absence patterns across samples (‘otu2tre.sh’), clustering these based on the Jaccard distance metric, and finally generating a tree representation to visualize the relationship between

samples. Before this occurred, the OTU table was copied into a new directory (‘/home/jordyn/ Project/data/gene_content_tree/’) and transposed.

Gene Content Tree Generation: /home/jordyn/project/data/gene_content_tree/

```
#Made new directory gene_content_tree and copied out.tab into it
cp ~/project/data/vsearch/otu.tab ~/project/data/gene_content_tree

#Transpose table
Rscript /home/bio4030/bin/transpose.R otu.tab otu.tab.txp

#Run Rscript to build distance matrix and perform clustering
Rscript otu2tre.sh
```

otu2tre.sh: /home/jordyn/project/data/gene_content_tree/

```
#create similarity (distance) matrix called "d"
#use the R analysis package vegan
library(vegan)
d=vegdist(dat,method="jaccard")

#cluster the taxa (average = UPGMA)
h=hclust(d,method="average")

#convert the clustering results into a Newick tree
#use the R phylogenetic package APE (Analyses of Phylogenetics and Evolution)
library(ape)
p=as.phylo(h)

#write the Newick tree to a file
write.tree(p,"otu.tab.txp.tre")
```

Constructing Phylogeny Tree in FigTree

The program FigTree was used to generate the Phylogeny Trees following formation from their respective program. Each tree was re-rooted to feature S. suis (NC_009442) \as the out-group, and branches were adjusted to maintain a consistent species order across all trees, simplifying group identification and analysis.

Results and Interpretation

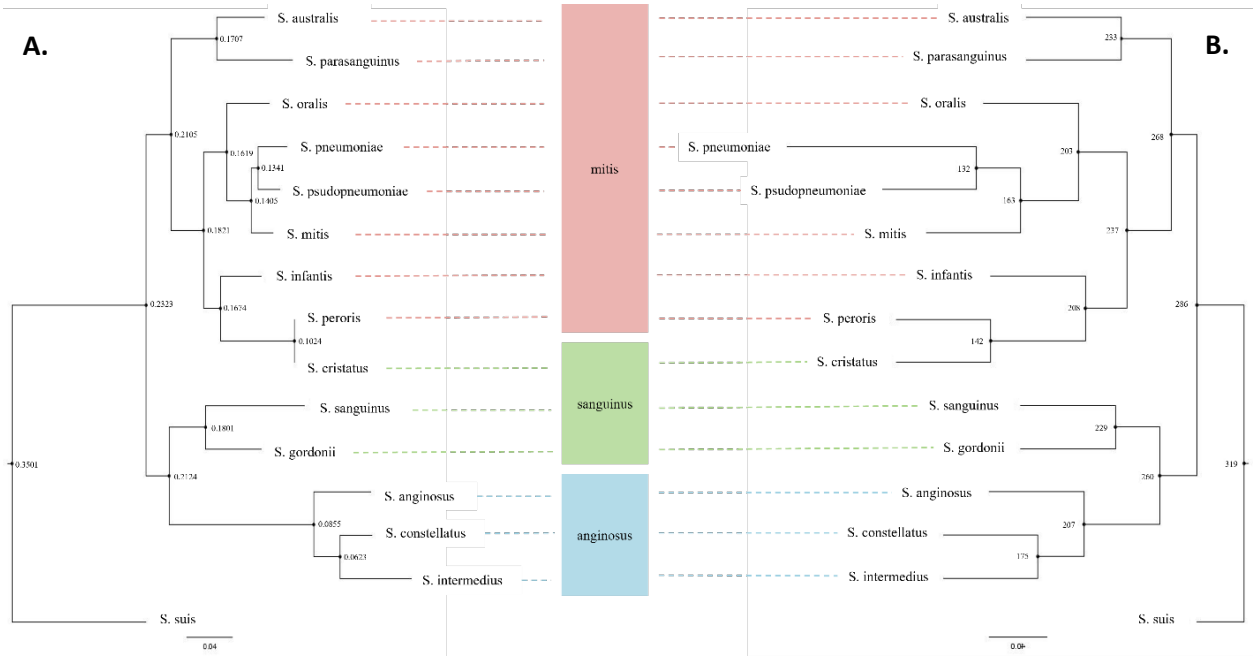

Figure 1 Concatenated and Consensus Phylogeny

Inferred relationships from the concatenated tree in Figure 1 A reveal intricate patterns of evolutionary relatedness among the 15 species of Streptococcus involved in this genomic analysis. Within this tree there are two distinct clades, and further subclades, showing a close relationship between the samples within them which suggests a shared evolutionary history or recent common ancestor. The clades can be broken down as such:

Clade 1:

Subclade A: S. australis and S. parasanguinus

Subclade B: S. oralis, S pneumoniae, S. psudopneumoniae, S. mitis, S. infantis, S. peroris, and S. cristatus

Clade 2:

Subclade A: S. sanguinis and S. gordonii

Subclade B: S. anginous. S. constellatus, and S. intermedius

In this analysis, the consensus tree (Figure 1B) showcases the broader groupings and longer branches, highlighting consensus patterns across multiple gene trees. Notably, some branches in this tree exhibit bootstrap values exceeding 100. It's important to note that bootstrap values are traditionally capped at 100, representing maximum support. Values above 100, while occasionally observed in computational outputs, are typically truncated, or presented as 100 as they don't carry additional significance beyond indicating maximum support. Despite their numerical representation, these values underscore the robustness and strong statistical support for the inferred branches, affirming the reliability of the depicted evolutionary relationships in the consensus tree.

An interesting finding from these phylogenetic trees was the unexpected placement of S. cristatus adjacent to S. peroris from the mitis group, rather than within its expected anginosus group. Streptococcus species are known for their susceptibility to horizontal gene transfer (HGT), a phenomenon prevalent in bacterial evolution. It's plausible that HGT facilitated the exchange of specific genes or segments between S. cristatus and S. peroris, potentially creating shared genetic material that led to their closer phylogenetic relationship. This genetic interchange challenges anticipated taxonomic associations. Moreover, the concept of convergent evolution, where disparate species evolve similar traits due to analogous environmental pressures, provides an alternative explanation. Shared environmental niches or functional constraints may have independently driven S. cristatus and S. peroris to develop similar genetic features, thereby appearing closely related in the phylogenetic tree despite their taxonomic divergence. These mechanisms, HGT and convergent evolution, offer intriguing avenues for understanding the unexpected relationship observed between S. cristatus and S. peroris in our analysis.

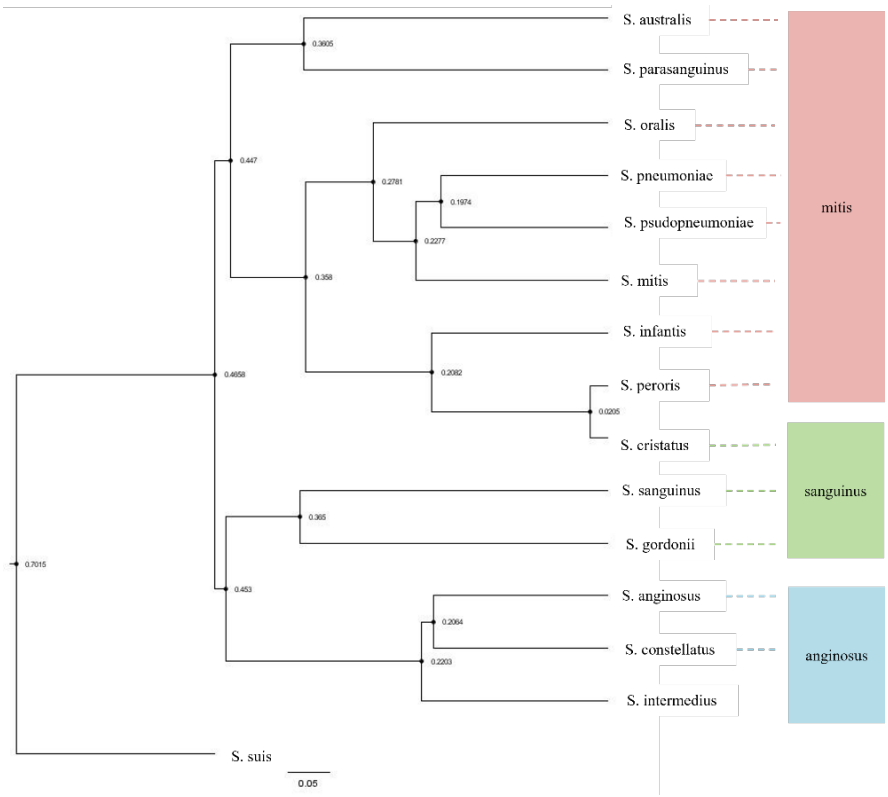

Figure 2 Gene Content Tree

The gene content approach, seen above, presents parallels with the concatenated and consensus trees. Understanding the varying node values, especially the lower values of 0.2 or below observed on outer branches, is crucial for interpreting the tree's reliability and evolutionary implications. These lower node values on outer branches might signify several factors influencing gene content dynamics. They could indicate uncertainty or variability in the presence or absence of specific genes used for constructing the tree, suggesting inconsistent gene patterns across analyzed organisms. Additionally, such values might hint at higher genomic diversity or rapid evolution within certain lineages at the outer edges of the tree. It's important to consider the possibility of incomplete or uneven data from the gathered FASTA files, potentially affecting the reliability of gene presence/absence information for these branches. Furthermore, lower node values could also hint at complex evolutionary scenarios, such as recent speciation events, horizontal gene transfers, or instances of convergent evolution among these organisms.