Pangenome construction

[Interactive session on Barcoo]

Constructed simulated reads for the two reference genomes (GenBank Accession No. CP006621 and CP003351), including their respective plasmids (CP006621, CP006622, CP006623, CP006624, CP006625 and CP006626; CP003351, CP003351 and CP003351).

module load samtools-intel/1.1

cat CP006620.fasta ./reference\_plasmids/CP006621.fasta ./reference\_plasmids/CP006622.fasta ./reference\_plasmids/CP006623.fasta ./reference\_plasmids/CP006624.fasta ./reference\_plasmids/CP006625.fasta ./reference\_plasmids/CP006626.fasta > CP006620\_full.fasta

cat CP003351.fasta ./reference\_plasmids/CP003352.fasta ./reference\_plasmids/CP003353.fasta ./reference\_plasmids/CP003354.fasta > CP003351\_full.fasta

wgsim -e 0 -1 200 -2 200 -r 0 -R 0 -X 0 -N 1100000 CP006620\_full.fasta CP006620\_full\_1.fastq CP006620\_full\_2.fastq

wgsim -e 0 -1 200 -2 200 -r 0 -R 0 -X 0 -N 1100000 CP003351\_full.fasta CP003351\_full\_1.fastq CP003351\_full\_2.fastq

gzip \*.fastq

module load python-gcc/2.7.5

Note: all assembled contigs were renamed so each had a unique name (requirement for iterative processing in MUMMER).

Example command:

python ../**rename\_contigs.py** -i aus0087\_AC026VACXX\_ATCACG\_L008\_OrderedRenamed.fasta -o aus0087\_AC026VACXX\_ATCACG\_L008\_Renamed.fasta -p EFCH

Pangenome constructed in FASTA format.

module load mummer-intel/3.23

python /vlsci/VR0082/kholt/mummerIterativePanGenome.py -o VREf\_pangenome\_20150213.fasta -r ../CP006620.fasta ../CP003351.fasta ../reference\_plasmids/\*.fasta ./assemblies\_for\_pangenome/\*.fasta -l 200 -i 95 > pangenome\_iterative\_log.txt

\*Filtered out contig <200 bp by text editor (Sublime Text 2) - saved <200 bp contigs to pangenome\_small\_contigs.fasta on local drive. Also noted start and finish of plasmid material:

start of first plasmid

ATGAATGATTTTAATTATTACAAATCAAAAGAAATTTATCGTGAAAAATATTATCAAATG

start of last plasmid

TTGAGTAAACGTGGCACGATCATAAATGAGTGGGAAAATTTAACAGCCCTATCTAACAGG

Contigs in pangenome renamed (for mapping pipeline)

python rename\_contigs.py -i VREf\_pangenome\_20150213.fasta -o VREf\_pangenome\_20150213\_renamed.fasta -p EFcg

\*In text editor, contigs split into three parts and relabeled (where needed), numbers from rename\_contigs retained:

Origin of contig; name code

Reference genome; EFcg,

Reference plasmid; EFcp,

Assembled contig, EFcc

Three parts concatenated together in editor back into original file before saving, then submission for annotation (Prokka v1.10)

prokka\_pan.txt

#!/bin/bash

#SBATCH -p main

#SBATCH --ntasks=1

#SBATCH --mem-per-cpu=16384

#SBATCH --time=03:00:00

module load prokka/1.10

prokka --outdir VREf\_pangenome\_20150213 --cpus 1 --locustag EFGG --genus Enterococcus --species faecium --usegenus --proteins /vlsci/VR0082/shared/resistanceDB\_srst2/resistance.fasta /vlsci/VR0082/shared/data/enterococcus/faecium/refgenomes/pangenome/VREf\_pangenome\_20150213\_renamed.fasta

Prokka changes contig order in the annotated GenBank file. To restore the original order in the FASTA pangenome:

python /vlsci/VR0082/shared/code/holtlab/reorderContigsFullNames.py --ordered\_fasta VREf\_pangenome\_20150213\_renamed.fasta --input\_genbank ./VREf\_pangenome\_20150213/EFGG\_02132015.gbk --ordered\_genbank VREf\_pangenome\_20150213\_renamed.gbk

Final pangenomes:

/vlsci/VR0082/shared/data/enterococcus/faecium/refgenomes/pangenome/VREf\_pangenome\_20150213\_renamed.fasta

/vlsci/VR0082/shared/data/enterococcus/faecium/refgenomes/pangenome/ VREf\_pangenome\_20150213\_renamed.gbk

Genbank version used as reference genome for RedDog v0.5.1 pangenome run using all 88 samples and the simulated reads of the two reference genomes with plasmids, using default setting except % of reads mapped not checked.

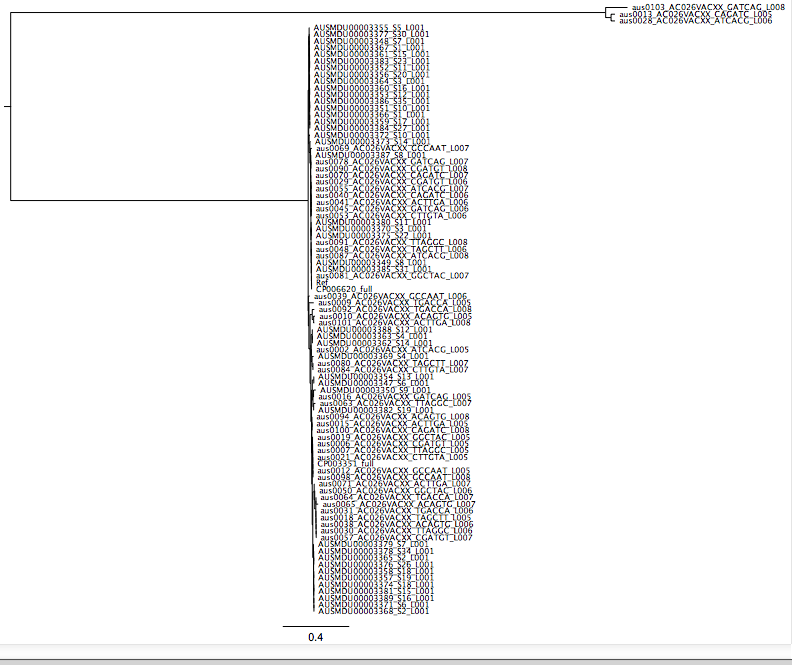
config file:

vre\_config.py

Pangenome RedDog output:

/vlsci/VR0082/shared/data/enterococcus/faecium/RedDogv51\_How2013Alf2015\_pangenome

Initial tree:



Repeats (in core genome EFcg00001 i.e. CP006620):

nucmer --maxmatch --nosimplify --prefix=EFcg00001 EFcg00001.fasta EFcg00001.fasta

show-coords -r EFcg00001.delta > EFcg00001.coords

python /scratch/VR0082/workspace/RedDog\_v51/filterCoords.py -i EFcg00001.coords -o EFcg00001.coords -I 95

Ran EFcg0001.fasta on PHAST (<http://phast.wishartlab.com/>) to identify phage, and added coordinates of hits to EFcg00001.coords (text editor). These were then used to filter out SNPs in repeat regions and phage from the allele table.

python /scratch/VR0082/workspace/RedDog\_v51/parseSNPtable.py -s VREf\_pangenome\_20150213\_renamed\_EFcg00001\_alleles\_var.csv -x /vlsci/VR0082/shared/data/enterococcus/faecium/refgenomes/pangenome/EFcg00001.coords -m filter,cons,aln -o aus0013\_AC026VACXX\_CAGATC\_L005,aus0028\_AC026VACXX\_ATCACG\_L006,aus0103\_AC026VACXX\_GATCAG\_L008 -c 0.95 -v EFcg00001 -l sequence\_list\_no\_outgroups.txt

Screen Output:

Reading SNP table from VREf\_pangenome\_20150213\_renamed\_EFcg00001\_alleles\_var.csv

outgroup(s): aus0013\_AC026VACXX\_CAGATC\_L005,aus0028\_AC026VACXX\_ATCACG\_L006,aus0103\_AC026VACXX\_GATCAG\_L008

including 87 ingroup strains listed in file sequence\_list\_no\_outgroups.txt

... finished reading 116034 SNPs in total

... keeping 21093 variable SNPs in 87 ingroup strains

... ignoring 94941 SNPs that are non-variable among these ingroup strains

Filtering SNPs that are located in excluded regions totalling 424669 bases

specified in file /vlsci/VR0082/shared/data/enterococcus/faecium/refgenomes/pangenome/EFcg00001.coords

... 18604 SNPs passed filter; printed to VREf\_pangenome\_20150213\_renamed\_EFcg00001\_alleles\_var\_3outgroups\_87strains\_var\_regionFiltered.csv

Filtering SNPs with fewer than 95.0% known alleles amongst ingroups

... 14721 SNPs passed filter; printed to VREf\_pangenome\_20150213\_renamed\_EFcg00001\_alleles\_var\_3outgroups\_87strains\_var\_regionFiltered\_cons0.95.csv

Printing alignment to file VREf\_pangenome\_20150213\_renamed\_EFcg00001\_alleles\_var\_3outgroups\_87strains\_var\_regionFiltered\_cons0.95.mfasta

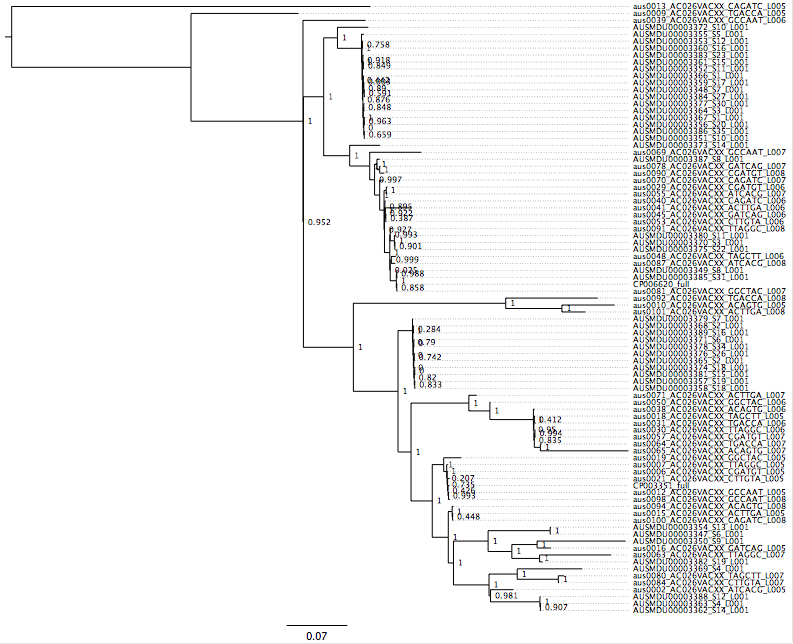
... done

rename the allele table:

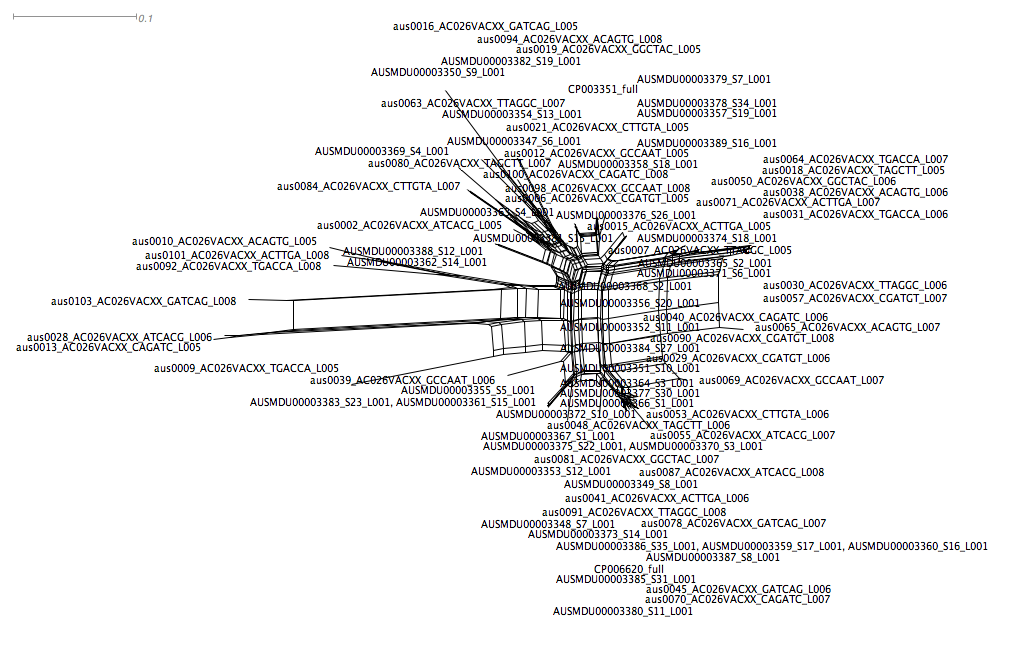
mv VREf\_pangenome\_20150213\_renamed\_EFcg00001\_alleles\_var\_3outgroups\_87strains\_var\_regionFiltered\_cons0.95.csv pangenome\_alleles\_3out\_87str\_var\_regFilt\_cons095.csv

make temp tree (for interest):

FastTree -gtr -gamma -nt VREf\_pangenome\_20150213\_renamed\_EFcg00001\_alleles\_var\_3outgroups\_87strains\_var\_regionFiltered\_cons0.95.mfasta > VREf\_pangenome\_20150213\_temp.tree



SplitsTree:

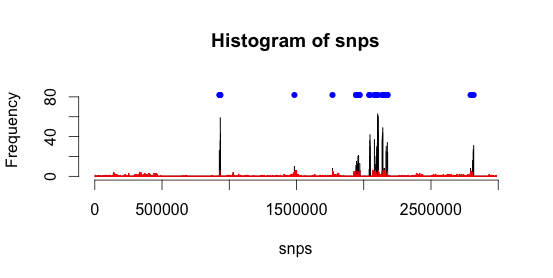


Downloaded csv file to local machine for R analysis:

source('getRecomb.R')

alleles <- read.csv("./VRE\_analysis/pangenome\_alleles\_3out\_87str\_var\_regFilt\_cons095.csv", header=T)

x <- getRecombBetweenStrains(alleles, 'CP006620\_full', 'CP003351\_full', w=1000, multiplier=1, plotResult=T)



c <- data.frame(x$block[,1],x$block[,2])

write.table(c, file = "./VRE\_analysis/recomb.csv", row.names=FALSE, na="",col.names=FALSE, sep=",")

Using txt editor coordinates in recomb.csv added to EFcg00001.coords, then rerun parseSNPtable.py

python /scratch/VR0082/workspace/RedDog\_v51/parseSNPtable.py -s VREf\_pangenome\_20150213\_renamed\_EFcg00001\_alleles\_var.csv -x /vlsci/VR0082/shared/data/enterococcus/faecium/refgenomes/pangenome/EFcg00001.coords -m filter,cons,aln -o aus0013\_AC026VACXX\_CAGATC\_L005,aus0028\_AC026VACXX\_ATCACG\_L006,aus0103\_AC026VACXX\_GATCAG\_L008 -c 0.95 -v EFcg00001 -l sequence\_list\_no\_outgroups.txt

Output:

Reading SNP table from VREf\_pangenome\_20150213\_renamed\_EFcg00001\_alleles\_var.csv

outgroup(s): aus0013\_AC026VACXX\_CAGATC\_L005,aus0028\_AC026VACXX\_ATCACG\_L006,aus0103\_AC026VACXX\_GATCAG\_L008

including 87 ingroup strains listed in file sequence\_list\_no\_outgroups.txt

... finished reading 116034 SNPs in total

... keeping 21093 variable SNPs in 87 ingroup strains

... ignoring 94941 SNPs that are non-variable among these ingroup strains

Filtering SNPs that are located in excluded regions totalling 483096 bases

specified in file /vlsci/VR0082/shared/data/enterococcus/faecium/refgenomes/pangenome/EFcg00001.coords

... 16368 SNPs passed filter; printed to VREf\_pangenome\_20150213\_renamed\_EFcg00001\_alleles\_var\_3outgroups\_87strains\_var\_regionFiltered.csv

Filtering SNPs with fewer than 95.0% known alleles amongst ingroups

... 12604 SNPs passed filter; printed to VREf\_pangenome\_20150213\_renamed\_EFcg00001\_alleles\_var\_3outgroups\_87strains\_var\_regionFiltered\_cons0.95.csv

Printing alignment to file VREf\_pangenome\_20150213\_renamed\_EFcg00001\_alleles\_var\_3outgroups\_87strains\_var\_regionFiltered\_cons0.95.mfasta

... done

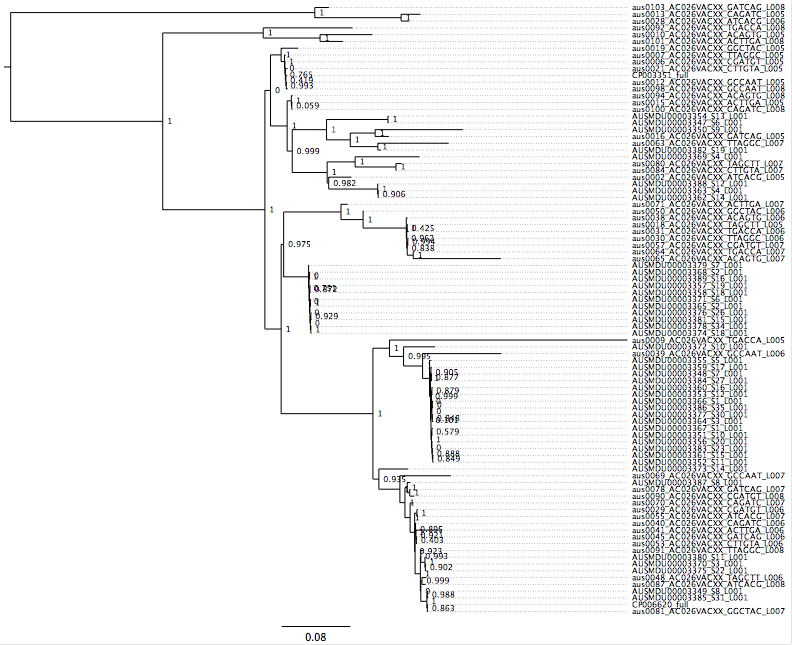
rename the alignment and allele tables:

mv VREf\_pangenome\_20150213\_renamed\_EFcg00001\_alleles\_var\_3outgroups\_87strains\_var\_regionFiltered\_cons0.95.csv VREf\_pangenome\_20150213\_final.csv

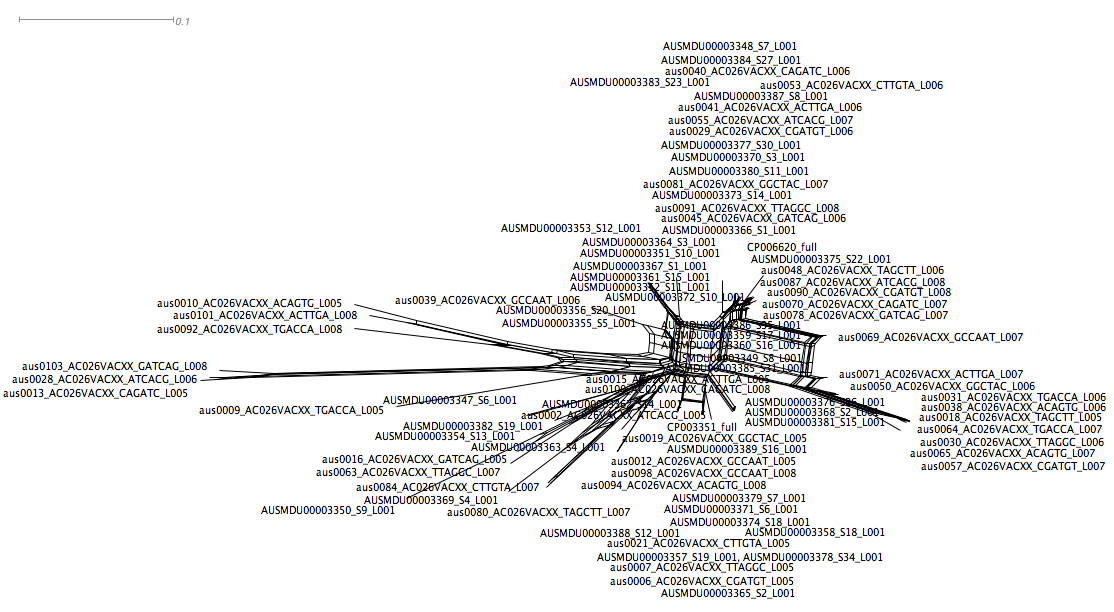
mv VREf\_pangenome\_20150213\_renamed\_EFcg00001\_alleles\_var\_3outgroups\_87strains\_var\_regionFiltered\_cons0.95.mfasta VREf\_pangenome\_20150213\_final.mfasta

create the fasttree:

FastTree -gtr -gamma -nt VREf\_pangenome\_20150213\_final.mfasta > VREf\_pangenome\_20150213\_final.tree



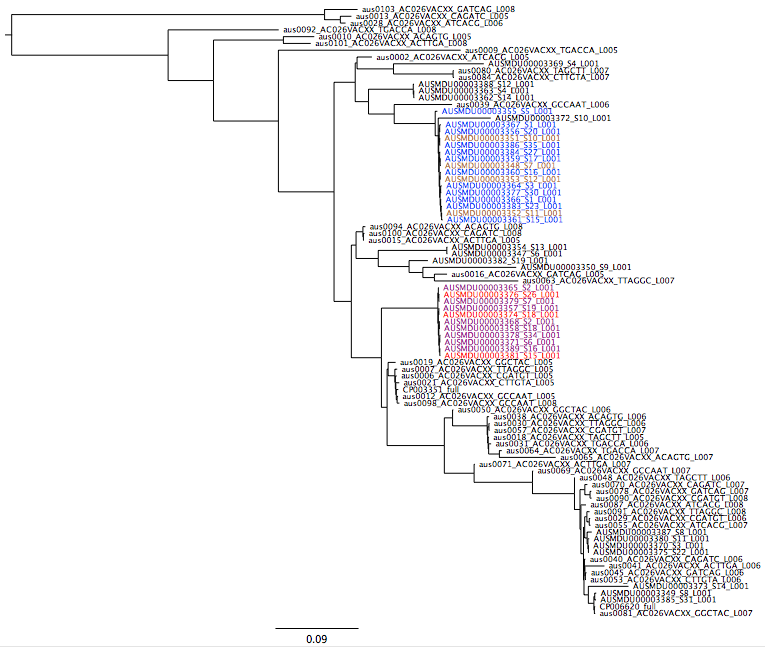
SplitsTree:

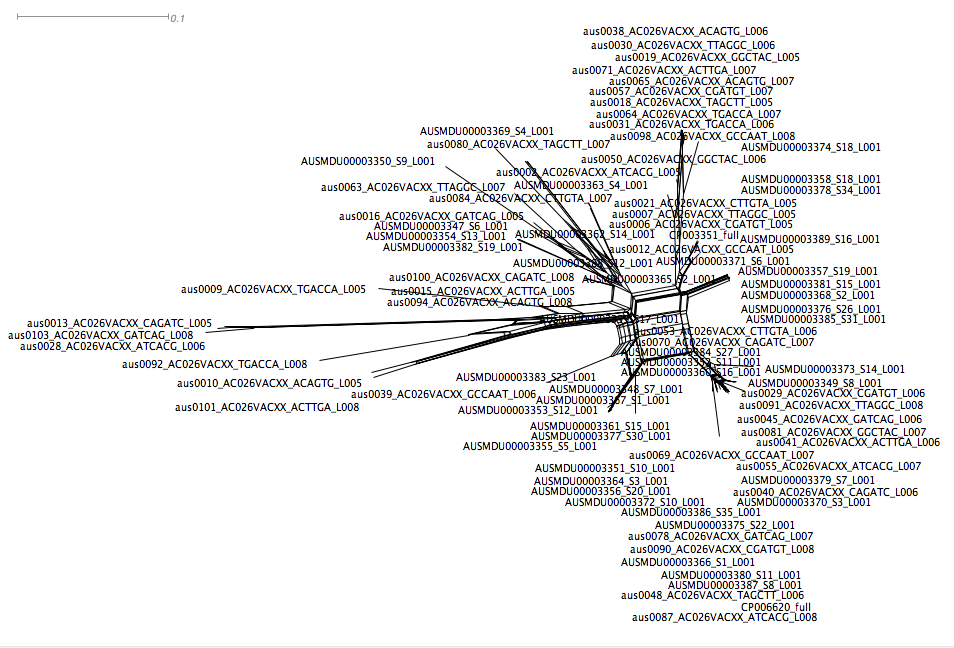


Used phylogeny to decide which pairs to use for pairwise recombination detection (as above): Aus0009, Aus0010, Aus0065 and MDU 3350

Detected recombination added to coords file and SNP table parsed again:

FastTree:



SplitsTree:

Used phylogeny to decide which pairs to use for pairwise recombination detection (as above, except multiplier=2): Aus0103 (outgroup), MDU3372, MDU 3373 and Aus0063 (six pairs); also, the last three with MDU3369, Aus0065 and MDU 3350 respectively (three pairs, nine total)

Detected recombination added to coords file and SNP table parsed again:

Output:

Reading SNP table from VREf\_pangenome\_20150213\_renamed\_EFcg00001\_alleles\_var.csv

outgroup(s): aus0013\_AC026VACXX\_CAGATC\_L005,aus0028\_AC026VACXX\_ATCACG\_L006,aus0103\_AC026VACXX\_GATCAG\_L008

including 87 ingroup strains listed in file sequence\_list\_no\_outgroups.txt

... finished reading 116034 SNPs in total

... keeping 21093 variable SNPs in 87 ingroup strains

... ignoring 94941 SNPs that are non-variable among these ingroup strains

Filtering SNPs that are located in excluded regions totalling 837633 bases

specified in file /vlsci/VR0082/shared/data/enterococcus/faecium/refgenomes/pangenome/EFcg00001.coords

... 9738 SNPs passed filter; printed to VREf\_pangenome\_20150213\_renamed\_EFcg00001\_alleles\_var\_3outgroups\_87strains\_var\_regionFiltered.csv

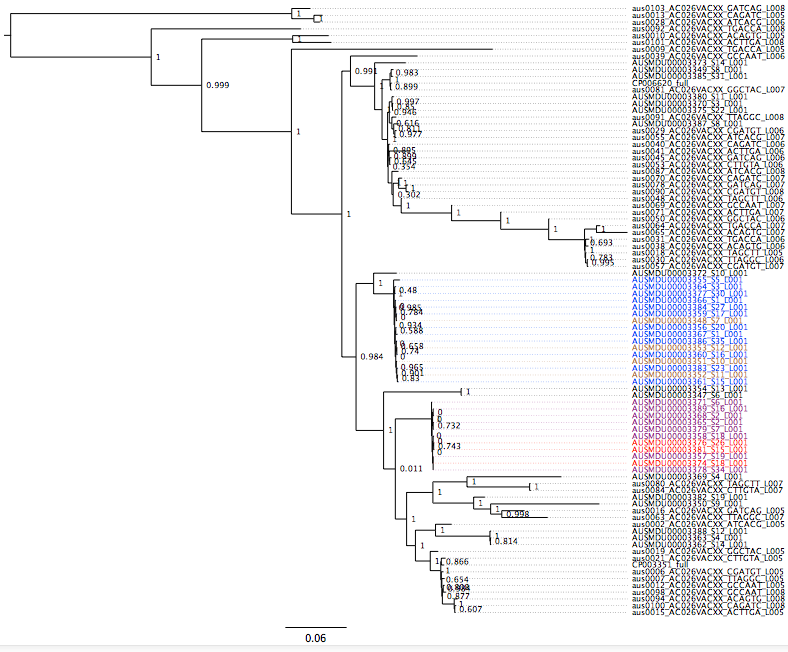
Filtering SNPs with fewer than 95.0% known alleles amongst ingroups

... 6133 SNPs passed filter; printed to VREf\_pangenome\_20150213\_renamed\_EFcg00001\_alleles\_var\_3outgroups\_87strains\_var\_regionFiltered\_cons0.95.csv

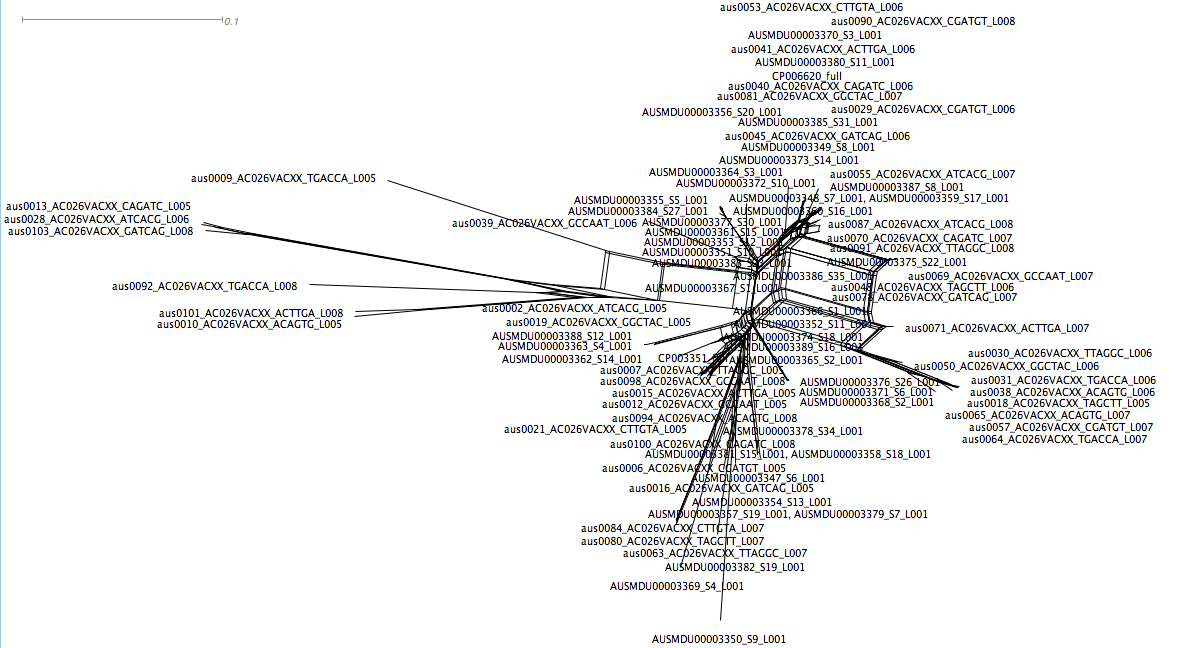
Printing alignment to file VREf\_pangenome\_20150213\_renamed\_EFcg00001\_alleles\_var\_3outgroups\_87strains\_var\_regionFiltered\_cons0.95.mfasta

... done

FastTree:



SplitsTree:

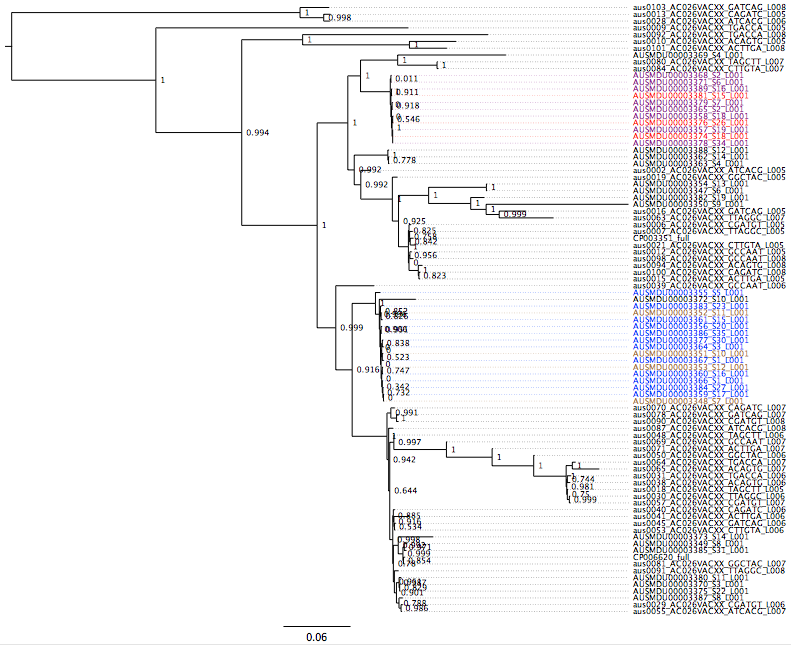


At this point - decided to go back and do all possible pairwise comparisons between strains in the set. A wrapper R script was written to collect the recombination data (note: this script is very slow, mainly because it is R, but also because of a triple nested for loop)

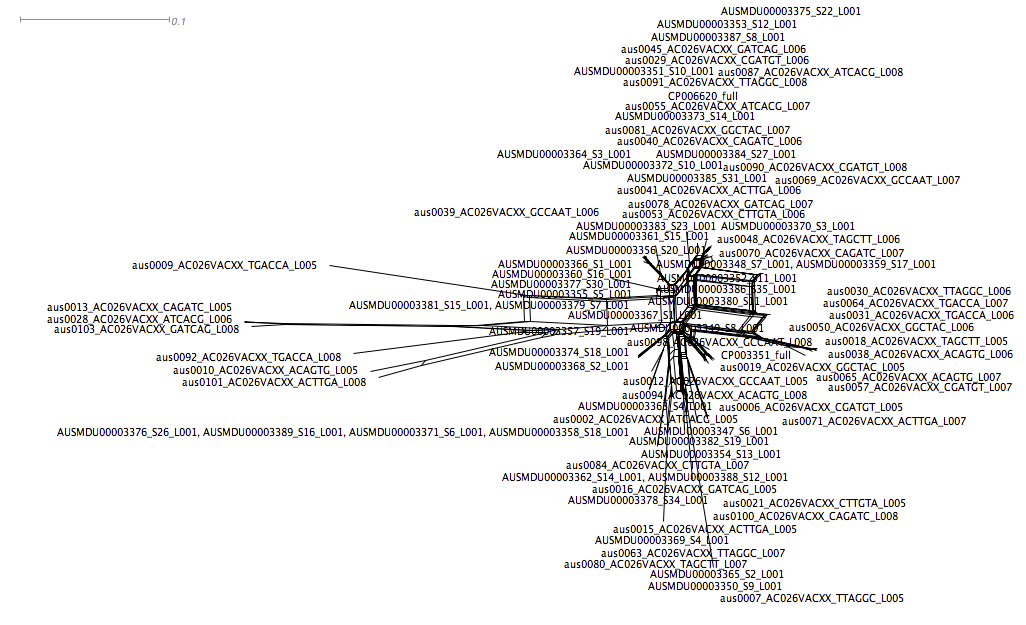
R script: findRecomb .R (which uses getRecomb.R)

settings (pairwise - must have at least 100 SNPs, multiplier=2)

FastTree:

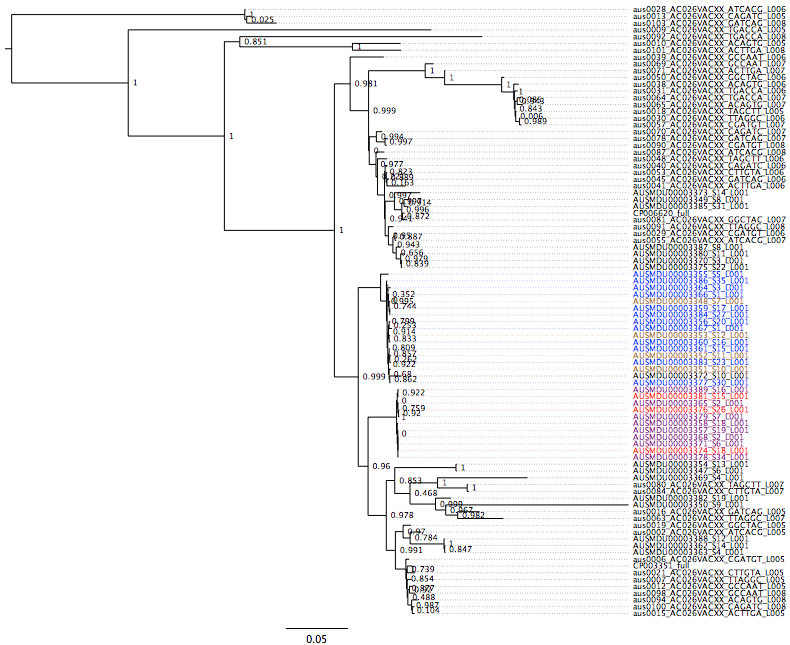


Splitstree:

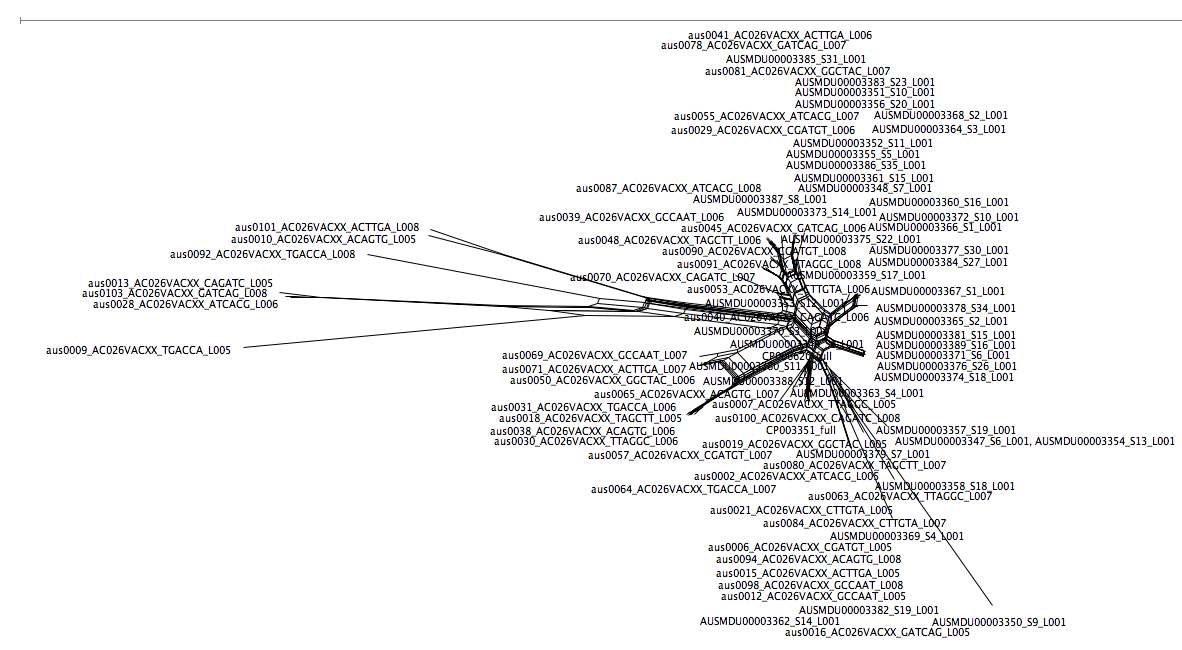


2nd Round (settings: pairwise - must have at least 100 SNPs, multiplier=2)

FastTree:



SplitsTree



Try2:

rnd1; min SNPs between pairs 300, g=3 Mb (~max window of 10 kbp), multiplier = 1 (more aggressive than first run)

rnd 2; min SNPs between pairs 100, multiplier =2

rnd3 ????

Note: After rnd1, 348 SNPs were left - the resulting splitstree STILL had networking

At this point decision made to use only the core tree from phylogenetics to map P/A of pan genome…

Gene Frequencies and Pangenome Curve:

python /vlsci/VR0082/kholt/parseGeneContent.py -g VREf\_pangenome\_20150213\_renamed\_CoverMatrix.csv -d VREf\_pangenome\_20150213\_renamed\_DepthMatrix.csv -o VREf\_pangenome\_20150213\_PresenceAbsence.csv -s VREf\_pangenome\_20150213\_GeneSummary.csv -i vre\_info.csv -c 95 -D 5

Gene Summary output:

/vlsci/VR0082/shared/data/enterococcus/faecium/RedDogv51\_How2013Alf2015\_pangenome/VREf\_pangenome\_20150213\_GeneSummary.csv

Opened in Excel to remove the outgroup frequencies and reorder the other three groups (g1, g2 and all others, g3). Saved as:

/vlsci/VR0082/shared/data/enterococcus/faecium/RedDogv51\_How2013Alf2015\_pangenome/VREf\_pangenome\_20150213\_gene\_freq.csv

Used in:

python /vlsci/VR0082/shared/davide/gene\_freq/**get\_significant\_genes.py**

Group stats output:

/vlsci/VR0082/shared/data/enterococcus/faecium/RedDogv51\_How2013Alf2015\_pangenome/VREf\_pangenome\_20150213\_genes\_by\_group.csv

Output file totals:

Significant genes between groups (as assessed by Fischer’s exact test, critical p-value = 0.01)

Each group v rest:

present absent

group1 v group 2+3: 442 542

group2 v group 1+3: 528 305

In R:

ct2 <- matrix(c(442,528,542,305), ncol=2)

chisq.test(ct2, correct=FALSE)

Pearson's Chi-squared test

data: ct2

X-squared = 61.8182, df = 1, p-value = 3.767e-15

group1 v group 2:

present

group1 370

group2 428

ct <- matrix(c(370,428), ncol=2)

chisq.test(ct)

Chi-squared test for given probabilities

data: ct

X-squared = 4.2155, df = 1, p-value = 0.04006

At critical p-value of 0.05, sign. diff., but not at critical p-value 0.01

Strict (fixed) differences:

Each group v rest:

present absent

group1 v group 2+3: 45 1

group2 v group 1+3: 0 0

group1 v group 2:

present

group1 132

group2 316

Chi-squared test for given probabilities

X-squared = 75.5714, df = 1, p-value < 2.2e-16

Edited the Pangenome P/A matrix to remove the 3 outgroups and 6 contaminated isolates not used in the pangenome constructions for the following.

Curve construction: (In R on local machine: should be run from parseGeneContent, but rpy not working - now fixed, highlighted)

content<-read.csv("./VRE\_analysis/VREf\_pangenome\_20150213\_PresenceAbsence.csv ",row.names=1);

content\_bygene<-table(apply(content,1,sum)); content0<-0;

if (0 %in% names(content\_bygene)) {content0<-content\_bygene[names(content\_bygene)==0]}; content0N<-0; if (ncol(content) %in% names(content\_bygene)) {content0N<-content\_bygene[names(content\_bygene)==ncol(content)]}; counts\_core<-matrix(ncol=2); counts\_core[1,]<-c(ncol(content),content0N); counts\_pan<-matrix(ncol=2); counts\_pan[1,]<-c(ncol(content),nrow(content)-content0);

for (i in 2:**(ncol(content))**) {for (j in 1:1000) {content\_sub <-content[, round(runif(i,1,ncol(content)))] ; content\_sum <-apply(content\_sub,1,sum);content\_core <- length(content\_sum[content\_sum==i]);content\_pan <-length(content\_sum[content\_sum>0]);counts\_core <-rbind(counts\_core,c(i,content\_core));counts\_pan <- rbind(counts\_pan,c(i,content\_pan))}};

write.table(counts\_core,file="./VRE\_analysis/VREf\_pangenome\_20150213\_1000\_CoreGenes.csv",sep=","); write.table(counts\_pan,file="./VRE\_analysis/VREf\_pangenome\_20150213\_1000\_PanGenome.csv",sep=","); csh<-boxplot(counts\_core[,2]~counts\_core[,1]); psh<-boxplot(counts\_pan[,2]~counts\_pan[,1]); pdf(file="./VRE\_analysis/VREf\_pangenome\_20150213\_1000\_PanGenome.pdf"); plot(psh$names,psh$stats[3,],ylim=c(0,6000),pch="",main="Pan genome",xlab="N isolates",ylab="N genes");

lines(psh$names,psh$stats[3,],lwd=2); lines(psh$names,psh$stats[2,],lty=2); lines(psh$names,psh$stats[4,],lty=2); lines(csh$names,csh$stats[3,],lwd=2); lines(csh$names,csh$stats[2,],lty=2); lines(csh$names,csh$stats[4,],lty=2);

dev.off();

Output:

