Starling-May18
Projects/Katarina Stuart/KStuart.Starling-Aug18/Nc3_HihiSV/Analysis/2024-01-24.SVprofiling

PDF Version generated by

Katarina Stuart (z5188231@ad.unsw.edu.au)

on

Aug 15, 2024 @03:06 PM NZST

Table of Contents

2024-01-24.SVprofiling

Katarina Stuart (z5188231@ad.unsw.edu.au) - Jun 05, 2024, 3:26 PM GMT+12

SV Profiling

Setting up the repeat library

```
cd /nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/data/repeats

REPEATMASKER=/nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/analysis/repeats/repeatmasker_aves.fasta

cat notiomystisCincta_combined_library.fasta $REPEATMASKER > All_repeats_hihi_custom.fasta
```

Exclude TEs Repeat Analysis

Run repeat masker

```
#!/bin/bash -e

#SBATCH --job-name=2024_03_27.SVCalling_Repeatmasking.sl
#SBATCH --account=uoa00338
#SBATCH --time=00-12:00:00
#SBATCH --mem=8GB
#SBATCH --output=%x_%j.errout
#SBATCH --mail-user=katarina.stuart@auckland.ac.nz
#SBATCH --mail-type=ALL
#SBATCH --nodes=1
#SBATCH --ntasks=1
#SBATCH --cpus-per-task=8
#SBATCH --profile task

#load modules
module load RepeatMasker/4.1.0-gimkl-2020a
module load BEDTools/2.30.0-GCC-11.3.0

cd /nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering

GENOME=/nesi/nobackup/uoa00338/kstuart_projects/Nc2_HihiWGS/data/resources/Ncf_H98617_scaffolded_genome.fa
SVCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/data/variant_calling/SV_curation/hihi_filtered_curated.recode.vcf

#grab the intervals that represent the upstream (+/- 30 bp) and downstream (+/- 30 bp) sites of DUP, DEL, INV
grep -v "^#" ${SVCF} | sed 's/;/\t/g' | cut -f 1,2,10 | sed 's/SVLEN=\|-//g' | awk '{ print $1"\t"$2-30"\t"$2+30}' > repeat_interval_upstream.bed
grep -v "^#" ${SVCF} | sed 's/;/\t/g' | cut -f 1,2,10 | sed 's/SVLEN=\|-//g' | awk '{ print $1"\t"$2+$3-30"\t"$2+$3+30}' > repeat_interval_downstream.bed

#cat all of these intervals into one file
cat repeat_interval_upstream.bed repeat_interval_downstream.bed > repeat_interval.bed

#get fasta of these sequences
bedtools getfasta -fi $GENOME -bed repeat_interval.bed -fo repeat_interval_ins.fasta

LIB=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/data/repeats/All_repeats_hihi_custom.fasta
```

```
#repeat mask
RepeatMasker -pa 8 -lib ${LIB} -dir ./ repeat_interval_ins.fasta
```

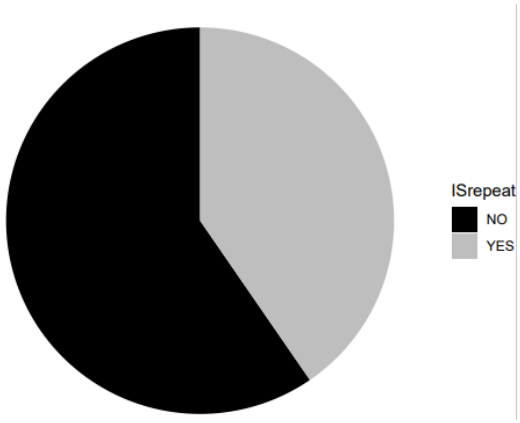
plot the results: preformatting

```
SVCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/data/variant_calling/SV_curation/hihi_filtered_curated.recode.vcf

cd /nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering

#make a file to match the CHROM|START|END signal in the repeat masked output to the variant ID from the VCF (mannually grepped for "ins_" and there was none)
grep -v "^#" ${SVCF} | sed 's/;/\t/g' | cut -f 1,2,3,10,11 | sed 's/SVLEN=\|-//g' | sed 's/SVTYPE=//g' | awk '{ print $1":"$2-30"-"$2+30,$3,$4,$5}' > repeat_interval_upstream.key
grep -v "^#" ${SVCF} |sed 's/;/\t/g' | cut -f 1,2,3,10,11 | sed 's/SVLEN=\|-//g' | sed 's/SVTYPE=//g' | awk '{ print $1":"$2+$4-30"-"$2+$4+30,$3,$4,$5}' > repeat_interval_downstream.key

cat repeat_interval_upstream.key repeat_interval_downstream.key> repeat_interval.key

#this key file can be used to link up vairant ID to the chrom+pos of the input bed file for repeatmasker
#checking the key works and contains all the variant IDs: cut -d' ' -f 2 repeat_interval.key | sort | uniq | wc -l

#format repeatmasker output
tail -n +4 repeat_interval_ins.fasta.out | sed 's/[[:blank:]]\+/\t/g' > repeat_interval_ins.fasta.out.format
cut -f6,12 repeat_interval_ins.fasta.out.format > repeat_interval_ins.fasta.out.format.cut

awk -f vlookup.awk repeat_interval_ins.fasta.out.format.cut repeat_interval.key | column | sed 's/Unspecified/Unknown/g' | sed 's/LTR\/ERV1\|LTR\/ERVK\|LTR\/ERVL\|LTR\/ERVL?/LTR/g' > merged_rep_repanalysis.samples.reps.txt
awk -f vlookup.awk repeat_interval_ins.fasta.out.format.cut repeat_interval.key | column > merged_rep_repanalysis_extradetails.samples.reps.txt

#make a base key, with just var ID, length and type
cut -d' ' -f 2-4 repeat_interval.key | sort | uniq > repeat_variant.key

#list of
grep -v "None" merged_rep_repanalysis.samples.reps.txt

#IN r, assigning repeat identities in a hierarchical design

module load R/4.1.0-gimkl-2020a
R
library(ggplot2)
library(dplyr)
library(data.table)

setwd("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering")

repeatDB <- read.table("merged_rep_repanalysis.samples.reps.txt", sep=" ", header=FALSE)

#create sublists that will be applied hierarchically
LTR <- repeatDB %>% filter(grepl("LTR",V5)) %>% select(V2)
LINE <- repeatDB %>% filter(grepl("LINE",V5)) %>% select(V2)
SINE <- repeatDB %>% filter(grepl("SINE",V5)) %>% select(V2)
DNA <- repeatDB %>% filter(grepl("DNA",V5)) %>% select(V2)
SAT <- repeatDB %>% filter(grepl("Satellite",V5)) %>% select(V2)
SIMP <- repeatDB %>% filter(grepl("Simple_repeat",V5)) %>% select(V2)
COMP <- repeatDB %>% filter(grepl("Low_complexity",V5)) %>% select(V2)
tRNA <- repeatDB %>% filter(grepl("tRNA",V5)) %>% select(V2)
RC <- repeatDB %>% filter(grepl("RC",V5)) %>% select(V2)
Unknown <- repeatDB %>% filter(grepl("Unknown",V5)) %>% select(V2)

#bring in key file and label in the above preferential order
key <- read.table("repeat_variant.key", sep=" ", header=FALSE)
colnames(key) <- c("Name","Length","Type")
```

```
key2 <- key %>% mutate(Repeat = ifelse(Name %in% LINE$V2, "LINE",
ifelse(Name %in% LTR$V2, "LTR",
ifelse(Name %in% SINE$V2, "SINE",
ifelse(Name %in% DNA$V2, "DNA",
ifelse(Name %in% SAT$V2, "Satellite",
ifelse(Name %in% SIMP$V2, "Simple Repeat",
ifelse(Name %in% COMP$V2, "Low Complexity",
ifelse(Name %in% tRNA$V2, "tRNA",
ifelse(Name %in% RC$V2, "Rolling Circle",
ifelse(Name %in% Unknown$V2, "Unknown", "None" ) ) ) ) ) ) ) ) ) ) )

write.table(key2,"repeat_annorated_SVs.txt",row.names=FALSE,sep="\t", quote = FALSE,col.names=TRUE)
```

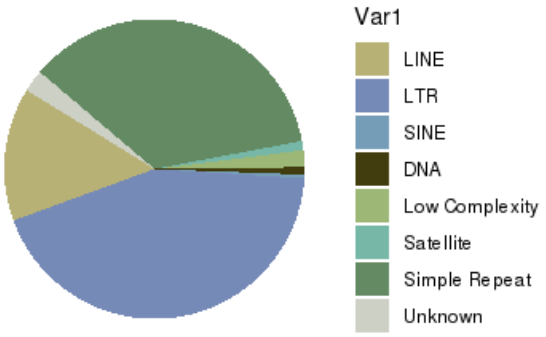
Plotting

```
module load R/4.1.0-gimkl-2020a
R
library(ggplot2)
library(dplyr)

setwd("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering")

repeats <- fread("repeat_annorated_SVs.txt")

repeats<- repeats %>% mutate(ISrepeat= ifelse(Repeat == "None", "NO", "YES"))

freq_table <-  as.data.frame( table(repeats$ISrepeat) )

#is repeat
pdf("Nc3_filtering_repeatpie.pdf", width=5, height=5)
ggplot(freq_table , aes(x = "", y =Freq, fill = Var1)) + geom_bar(stat = "identity", width = 1) +
        coord_polar("y", start = 0) + scale_fill_manual(values = c("black", "grey")) + theme_void()
dev.off()
```




```
#repeat type
repeats2 <- repeats %>% filter(!Repeat == "None")
freq_table <-  as.data.frame( table(repeats2$Repeat) )

order <- c('LINE','LTR','SINE','DNA', 'Low Complexity','Satellite','Simple Repeat','Unknown')
freq_table$Var1<- factor(freq_table$Var1, levels = order)

pdf("Nc3_filtering_repeattypepie.pdf", width=5, height=5)
ggplot(freq_table , aes(x = "", y =Freq, fill = Var1)) + geom_bar(stat = "identity", width = 1) +
        coord_polar("y", start = -45) + scale_fill_manual(values = c("#B7B176", "#768AB7",
"#769DB7","#423D0E","#9DB776","#76B7A7","#648A64","#CDD1C5")) + theme_void()
```

```
dev.off()
```



Filtering for TE overlap, and MAF/missingness

working out which SV's map to TE's

```
cd /nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering
grep "LINE\|LTR\|SINE\|DNA\|tRNA\|Rolling Circle" repeat_annorated_SVs.txt | cut -f1 > snpid_TEmappedSvs.txt
module load VCFtools/0.1.15-GCC-9.2.0-Perl-5.30.1
SVCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/data/variant_calling/SV_curation/hihi_filtered_curated.recode.vcf
vcftools --vcf $SVCF --exclude snpid_TEmappedSvs.txt --recode-INFO-all --recode --out merged_rep_noTE
```

After filtering, kept 30 out of 30 Individuals

Outputting VCF file...

After filtering, kept 936 out of a possible 1229 Sites

Check to see one final time for missingness and MAC - no impact as this was already filtered earlier but want to be sure.

```
module load VCFtools/0.1.15-GCC-9.2.0-Perl-5.30.1
vcftools --vcf merged_rep_noTE.recode.vcf --max-missing 0.5 --mac 1 --recode --recode-INFO-all --out merged_rep_missfiltered
```

After filtering, kept 936 out of a possible 936 Sites

Filtering to keep just the TES:

```
cd /nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering
module load VCFtools/0.1.15-GCC-9.2.0-Perl-5.30.1
SVCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/data/variant_calling/SV_curation/hihi_filtered_curated.recode.vcf
vcftools --vcf $SVCF --snps snpid_TEmappedSvs.txt --max-missing 0.5 --mac 1 --recode-INFO-all --recode --out merged_rep_TEonly
```

After filtering, kept 293 out of a possible 1229 Sites

fix the reference allele

https://www.biostars.org/p/347588/

```
cd /nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering

module load BCFtools/1.13-GCC-9.2.0

GENOME=/nesi/nobackup/uoa00338/kstuart_projects/Nc2_HihiWGS/data/resources/Ncf_H98617_scaffolded_genome.fa
VCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering/merged_rep_missfiltered.recode.vcf
OUTFILE=merged_rep_missfiltered_reffix.vcf

perl fixRefVCF.pl $GENOME $VCF $OUTFILE
```

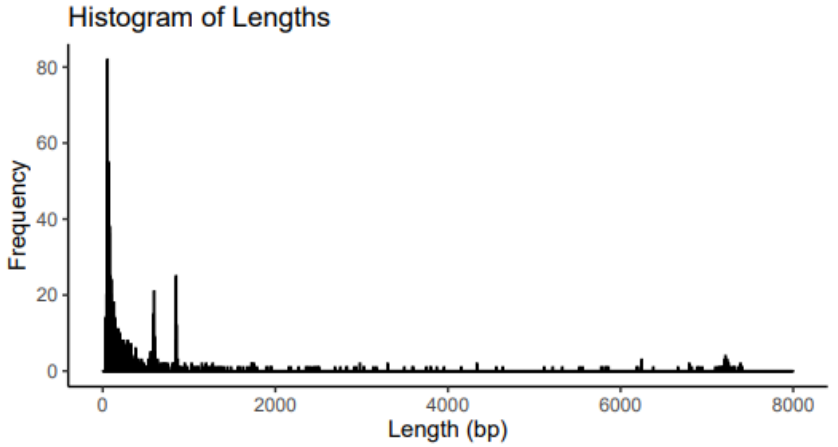
Change the reference allele to the major allele so it is not reliant on reference genome as determinant

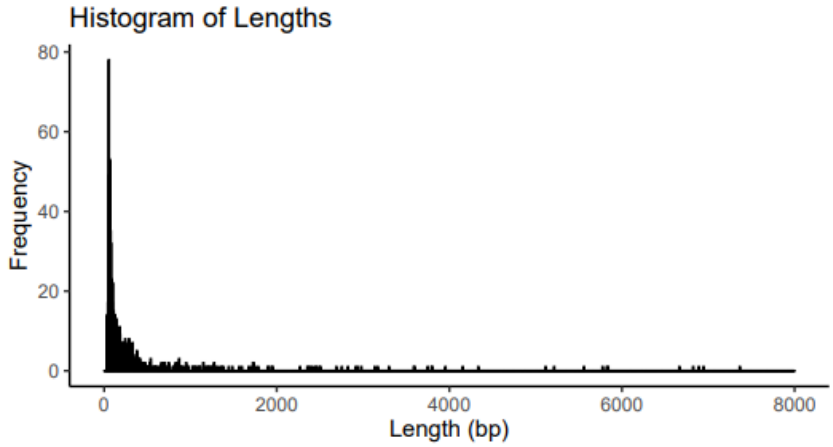
```
cd /nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering/recodeREF

VCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering/merged_rep_missfiltered_reffix.vcf
module purge
module load PLINK/2.00a2.3

plink2 --vcf $VCF --allow-extra-chr --chr-set 28 --make-bed --maj-ref  --out merged_rep_missfiltered_reffix2.plink
plink2 --bfile merged_rep_missfiltered_reffix2.plink --allow-extra-chr --chr-set 28 --recode vcf --out merged_rep_missfiltered_reffix2
```

Length Profiling

```
cd /nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/legnths

SVCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/data/variant_calling/SV_curation/hihi_filtered_curated.recode.vcf

#grab the intervals that represent the upstream (+/- 30 bp) and downstream (+/- 30 bp) sites of DUP, DEL, INV
grep -v "^#" ${SVCF} |sed 's/;/\t/g' | cut -f 1,2,10,11 | sed -e 's/SVLEN=\|-//g' -e 's/SVTYPE=//g' > hihi_filtered_curated_key.txt

VCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering/merged_rep_missfiltered.recode.vcf

#grab the intervals that represent the upstream (+/- 30 bp) and downstream (+/- 30 bp) sites of DUP, DEL, INV
grep -v "^#" ${VCF} |sed 's/;/\t/g' | cut -f 1,2,10,11 | sed -e 's/SVLEN=\|-//g' -e 's/SVTYPE=//g' > merged_rep_missfiltered_key.txt

module load R/4.1.0-gimkl-2020a
R
library(ggplot2)
library(dplyr)
```

```
library(data.table)

setwd("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/legnths")

#histogram of pre-repeat filtering
lengths <- read.table("hihi_filtered_curated_key.txt", sep="\t", header=FALSE)

pdf("Nc3_prefilter_lengths.pdf", width=4.5, height=2.5)
ggplot(lengths, aes(x = V3)) +
  geom_histogram(binwidth = 5, fill = "cornflowerblue", color = "black") +
 labs(title = "Histogram of Lengths", x = "Length (bp)", y = "Frequency") + xlim(0,8000) + theme_classic()
dev.off()



#histogram of post-repeat filtering
lengths <- read.table("merged_rep_missfiltered_key.txt", sep="\t", header=FALSE)

pdf("Nc3_postfilter_lengths.pdf", width=4.5, height=2.5)
ggplot(lengths, aes(x = V3)) +
  geom_histogram(binwidth = 5, fill = "cornflowerblue", color = "black") +
 labs(title = "Histogram of Lengths", x = "Length (bp)", y = "Frequency") + xlim(0,8000) + theme_classic()
dev.off()


#confirm identity of the peaks on the pre-filtering
repeats <-
read.table("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering/merged_rep_repanalysis_extradetails.samples.reps.txt",
sep=" ", header=FALSE)

hist_data <- hist(repeats$V3, breaks = 1200, plot = FALSE)
# Identify peaks
peak_indices <- which(diff(sign(diff(hist_data$counts))) == -2) + 1
# Print peak indices and their corresponding counts
peaks <- data.frame(Bin = hist_data$mids[peak_indices], Count = hist_data$counts[peak_indices])
print(peaks)

#check to see if plot is the same as pre-repeat filtering
ggplot(repeats , aes(x = V3)) +
  geom_histogram(binwidth = 5, fill = "cornflowerblue", color = "black") +
 labs(title = "Histogram of Lengths", x = "Length (bp)", y = "Frequency") + xlim(0,8000) + theme_classic()

#peaks
repeats %>% filter(V3 > 475 & V3 < 675) %>% count(V5) #LTR/ERVL
repeats %>% filter(V3 > 725 & V3 < 1125 ) %>% count(V5) #LTR/ERVL
repeats %>% filter(V3 > 5325& V3 < 6375) %>% count(V5) #LTR/ERVL
repeats %>% filter(V3 > 6925 & V3 < 8025) %>% count(V5) # LTR/ERVL
```

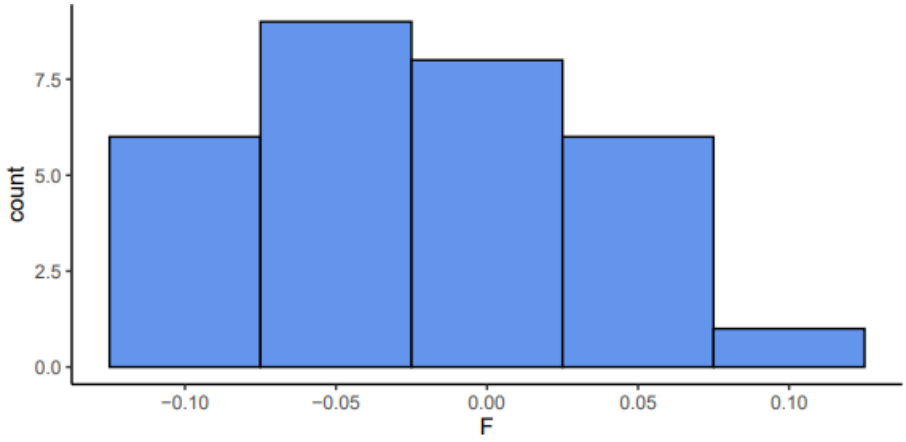


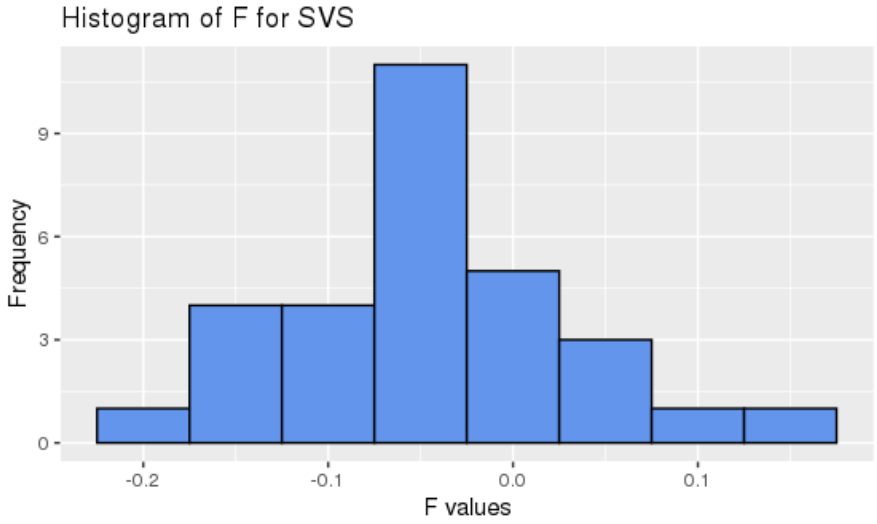


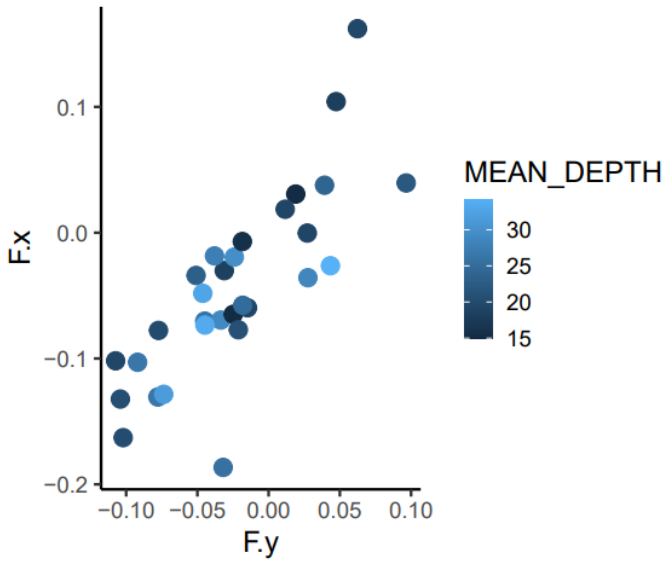
SNPs x SVs: Heterozygosity

```
cd /nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/heterozygosity

module load VCFtools/0.1.15-GCC-9.2.0-Perl-5.30.1

SNP=/nesi/nobackup/uoa00338/kstuart_projects/Nc2_HihiWGS/data/snp_variants_updated/hihi_wgs_filter_highcov_no83318_autosomes.recode.vcf
vcftools --vcf $SNP --het --out hihi_wgs_filter_highcov_no83318_autosomes

SVCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering/merged_rep_missfiltered.recode.vcf
vcftools --vcf $SVCF --het --out merged_rep_missfiltered


#and also depth
SNP=/nesi/nobackup/uoa00338/kstuart_projects/Nc2_HihiWGS/data/snp_variants_updated/hihi_wgs_filter_highcov_no83318_autosomes.recode.vcf
vcftools --vcf $SNP --depth --out hihi_wgs_filter_highcov_no83318_autosomes



module load R/4.1.0-gimkl-2020a
R
```

```
library(ggplot2)
library(data.table)
library(tidyr)
library(dplyr)

setwd("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/heterozygosity")

data <- fread("hihi_wgs_filter_highcov_no83318_autosomes.het")

pdf("Nc3_hethisto_SNPs.pdf", width=6, height=3)
ggplot(data, aes(x = F)) +
  geom_histogram(binwidth = 0.05, fill = "cornflowerblue", color = "black") + theme_classic() + xlim(-0.2,0.2)
dev.off()

data2 <- fread("merged_rep_missfiltered.het")

pdf("Nc3_hethisto_SVs.pdf", width=6, height=3)
ggplot(data2, aes(x = F)) +
  geom_histogram(binwidth = 0.05, fill = "cornflowerblue", color = "black") + theme_classic()+ xlim(-0.2,0.2)
dev.off()

read_depth <- fread("hihi_wgs_filter_highcov_no83318_autosomes.idepth")

data3 <- merge(data2, data, by.x = "INDV", by.y = "INDV")
data4 <- merge(data3, read_depth, by.x = "INDV", by.y = "INDV")

pdf("Nc3_FxF_depth.pdf", width=6, height=5)
ggplot(data4 ,aes(x=F.y ,y=F.x, col = MEAN_DEPTH))+
geom_point(size=5,alpha=1)+
theme_classic(base_size = 18)
dev.off()

model<- lm(F.x ~ F.y +MEAN_DEPTH , data=data4 )
summary(model)
anova(model)

model<- lm(F.x ~ F.y, data=data4 )
summary(model)
anova(model)
```







SVs: PCAs

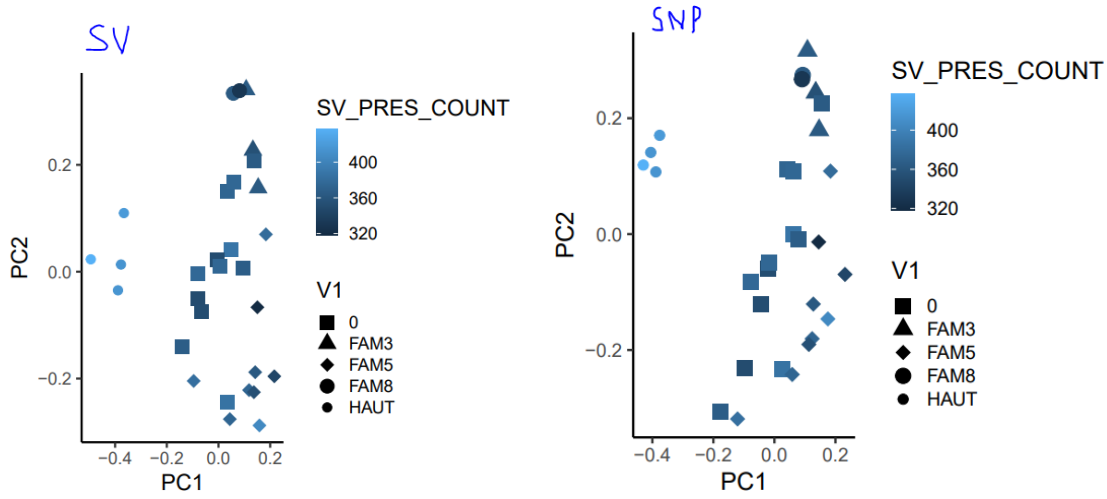
```
cd /nesi/nobackup/uoa02613/kstuart_projects/At3_TEpopgen/analysis/SV_profiling/pca

module load PLINK/1.09b6.16
module load VCFtools/0.1.15-GCC-9.2.0-Perl-5.30.1

#SV PCA
SVCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering/merged_rep_missfiltered.recode.vcf
vcftools --vcf $SVCF --plink --out merged_rep_missfiltered.plink
plink --file merged_rep_missfiltered.plink --pca --out merged_rep_missfiltered --make-rel   --allow-extra-chr --chr-set 27

#SNP PCA
SNP=/nesi/nobackup/uoa00338/kstuart_projects/Nc2_HihiWGS/data/snp_variants_updated/hihi_wgs_filter_highcov_no83318_autosomes.recode.vcf
vcftools --vcf $SNP --plink --out hihi_wgs_filter_highcov_no83318_autosomes.plink
plink --file hihi_wgs_filter_highcov_no83318_autosomes.plink --pca --out hihi_wgs_filter_highcov_no83318_autosomes --make-rel   --allow-extra-chr --chr-set 27
```

```
#also check relatedness
SNP=/nesi/nobackup/uoa00338/kstuart_projects/Nc2_HihiWGS/data/snp_variants_updated/hihi_wgs_filter_highcov_no83318_autosomes.recode.vcf
vcftools --vcf $SNP --relatedness --out hihi_wgs_filter_highcov_no83318_autosomes
```

Plotting

```
module load R/4.1.0-gimkl-2020a
R
library(ggplot2)
library(ggplot2)
library(data.table)
library(tidyr)
library(dplyr)

setwd("/nesi/nobackup/uoa02613/kstuart_projects/At3_TEpopgen/analysis/SV_profiling/pca")

# SV
pca.eigenvec <- read.table("merged_rep_missfiltered.eigenvec", sep=" ")
pca_g1 <- data.frame(PC1 = pca.eigenvec$V3,    # the first eigenvector
                     PC2 = pca.eigenvec$V4,    # the second eigenvector
                     PC3 = pca.eigenvec$V5,    # the second eigenvector
                     stringsAsFactors = FALSE)

population2 <- fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/data/high_coverage_indiv_updated_metadata_roh_depth.txt", header = T,
sep = "\t")
pca_plot <- cbind(population2 ,pca_g1)

Fam <-(fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/data/variant_calling/SV_curation/mendel/hihi_genofiltered.plink.fam", header = F,
sep = "\t"))[,c(1,2)]
Fam$V2  <- as.integer(Fam$V2)
pca_plot1 <- merge(pca_plot ,Fam, by.x = "IID", by.y = "V2")

read_depth <-
fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/heterozygosity/hihi_wgs_filter_highcov_no83318_autosomes.idepth")

pca_plot2 <- merge(pca_plot1, read_depth, by.x = "IID", by.y = "INDV")

pdf("Nc3_PCA_SV.pdf", width=6, height=5)
ggplot(pca_plot2 ,aes(x=PC1,y=PC2, col = MEAN_DEPTH, pch = V1 ))+
geom_point(size=5,alpha=1)+ scale_shape_manual(values = c(15, 17,18, 19, 20)) +
theme_classic(base_size = 18)
dev.off()

# SNP
pca.eigenvec <- read.table("hihi_wgs_filter_highcov_no83318_autosomes.eigenvec", sep=" ")
pca_g1 <- data.frame(PC1 = pca.eigenvec$V3,    # the first eigenvector
                     PC2 = pca.eigenvec$V4,    # the second eigenvector
                     PC3 = pca.eigenvec$V5,    # the second eigenvector
                     stringsAsFactors = FALSE)

population2 <- fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/data/high_coverage_indiv_updated_metadata_roh_depth.txt", header = T,
sep = "\t")
pca_plot <- cbind(population2 ,pca_g1)

Fam <-(fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/data/variant_calling/SV_curation/mendel/hihi_genofiltered.plink.fam", header = F,
sep = "\t"))[,c(1,2)]
Fam$V2  <- as.integer(Fam$V2)
pca_plot1 <- merge(pca_plot ,Fam, by.x = "IID", by.y = "V2")

read_depth <-
fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/heterozygosity/hihi_wgs_filter_highcov_no83318_autosomes.idepth")
```

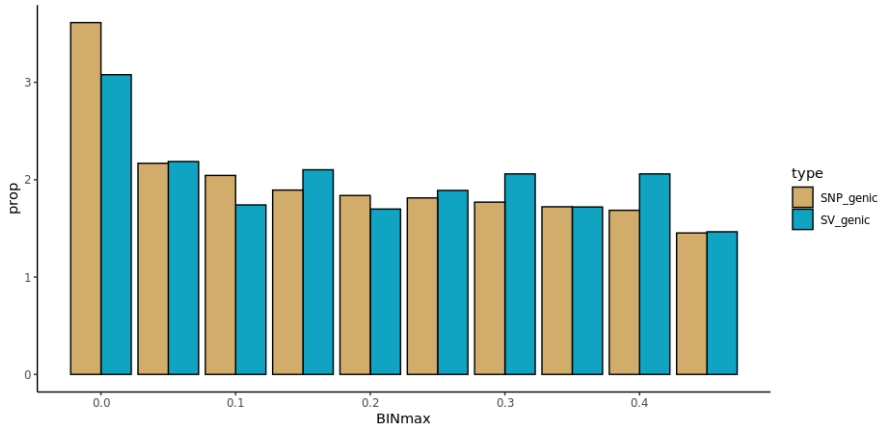
```
pca_plot2 <- merge(pca_plot1, read_depth, by.x = "IID", by.y = "INDV")

pdf("Nc3_PCA_SNP.pdf", width=6, height=5)
ggplot(pca_plot2 ,aes(x=PC1,y=PC2, col = MEAN_DEPTH, pch = V1 ))+
geom_point(size=5,alpha=1)+ scale_shape_manual(values = c(15, 17,18, 19, 20)) +
theme_classic(base_size = 18)
dev.off()
```



SNPs x SVs: MAF

```
module load PLINK/1.09b6.16
module load VCFtools/0.1.15-GCC-9.2.0-Perl-5.30.1

cd /nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/SFS

#No Genic Breakdown
SNP=/nesi/nobackup/uoa00338/kstuart_projects/Nc2_HihiWGS/data/snp_variants_updated/hihi_wgs_filter_highcov_no83318_autosomes.recode.vcf
SVCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering/merged_rep_missfiltered.recode.vcf


plink --vcf $SNP --allow-extra-chr --freq counts --chr-set 27  --out hihi_wgs_filter_highcov_no83318_autosomes
plink --vcf $SVCF --allow-extra-chr --freq counts --chr-set 27  --out merged_rep_missfiltered


#plot in R

module load R/4.1.0-gimkl-2020a
R
library(ggplot2)
library(data.table)
library(tidyr)
library(dplyr)

setwd("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/SFS")

breakpoint <- seq(0, 0.5, length.out = 11)

SV_genic <- fread("merged_rep_missfiltered.frq.counts")%>% mutate(allele_freq = pmin(C1, C2) / (C1 + C2) )
hist_SV_genic  <- hist(SV_genic $allele_freq, plot = FALSE, breaks = breakpoint )
```

```
SNP_genic <- fread("hihi_wgs_filter_highcov_no83318_autosomes.frq.counts")%>% mutate(allele_freq = pmin(C1, C2) / (C1 + C2) )
hist_SNP_genic  <- hist(SNP_genic$allele_freq, plot = FALSE, breaks = breakpoint )



table <- cbind(hist_SV_genic$density, hist_SNP_genic$density,  hist_SNP_genic$breaks[1:10])
colnames(table) <- c("SV_genic", "SNP_genic","TE_genic", "BINmax")

table3 <- gather(as.data.frame(table), key="type", value="prop", 1:2)

order <- c('SNP_genic','SNP_intergenic','SV_genic','SV_intergenic')
table3$type<- factor(table3$type, levels = order)

pdf("Nc3_SNP_SV_SFS.pdf", width=6, height=3)
ggplot(data=table3, aes(x=BINmax, y=prop, fill=type)) + geom_bar(stat="identity", position=position_dodge(),colour="black") + theme_classic() +
scale_fill_manual(values=c("#d1ac6b","#10a4c2"))
dev.off()
```



SNPs x SVs: het and hom

```
module load PLINK/1.09b6.16
module load VCFtools/0.1.15-GCC-9.2.0-Perl-5.30.1

cd /nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/het_hom

SVCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering/merged_rep_missfiltered_reffix2.vcf
SNP=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/counts/hihi_wgs_filter_highcov_no83318_autosomes_reffix2.vcf #from
next sheet
xxx



#!/bin/bash
# Read the VCF file line by line
while IFS= read -r line; do
    # Skip empty lines and lines starting with #
    if [[ -z "$line" || "$line" == \#* ]]; then
        continue
    fi
    # Extract the genotype columns
```

```
    genotypes=$(echo "$line" | awk '{for(i=10;i<=NF;i++) print $i}')

    # Count occurrences of each genotype pattern
    count_00=$(grep -o "0/0" <<< "$genotypes" | wc -l)
    count_01=$(grep -o "0/1" <<< "$genotypes" | wc -l)
    count_11=$(grep -o "1/1" <<< "$genotypes" | wc -l)

    # Print counts for the current row
    echo "$count_00 $count_01 $count_11"
done < $SVCF > SV_genotype_counts.txt
done < $SNP > SNP_genotype_counts_reffix2.txt
```

#plot in R

```
module load R/4.1.0-gimkl-2020a
R
library(ggplot2)
library(data.table)
library(tidyr)
library(dplyr)

setwd("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/het_hom")

order <- c('HomRfreq','Hetfreq','HomAfreq')

SV <- fread("SV_genotype_counts.txt")%>% mutate(HomRfreq = (V1) / (V1 + V2 + V3), Hetfreq = (V2) / (V1 + V2 + V3), HomAfreq = (V3) / (V1 + V2 + V3) )
SV_gather <- gather(SV[,4:6], key = "Frequency", value = "Value")
SV_gather$Frequency<- factor(SV_gather$Frequency, levels = order)
SV_gather <- SV_gather %>% mutate(type="SV")
```

```
#SNP <- fread("SNP_genotype_counts.txt")%>% mutate(HomRfreq = pmax(V1, V3) / (V1 + V2 + V3), Hetfreq = (V2) / (V1 + V2 + V3), HomAfreq = pmin(V1, V3) / (V1 + V2 + V3) )
SNP <- fread("SNP_genotype_counts_reffix2.txt")%>% mutate(HomRfreq = pmax(V1, V3) / (V1 + V2 + V3), Hetfreq = (V2) / (V1 + V2 + V3), HomAfreq = pmin(V1, V3) / (V1 + V2 + V3) )

#subsample?
# Set the seed for reproducibility
set.seed(123)

# Generate 1000 random indices
sample_indices <- sample(1:nrow(SNP), 1000, replace = FALSE)

# Subset the data using the sampled indices
subsampled_SNP <- SNP[sample_indices, ]

SNP_gather <- gather(subsampled_SNP[,4:6], key = "Frequency", value = "Value")
SNP_gather$Frequency<- factor(SNP_gather$Frequency, levels = order)
SNP_gather <- SNP_gather %>% mutate(type="SNP")

data <- rbind(SNP_gather , SV_gather , TE_gather  )

pdf("Nc3_SNP_SV_boxplot.pdf", width=6, height=3)
ggplot(data , aes(x = Frequency, y = Value, fill=type)) +
  geom_boxplot() +
  labs(x = "Frequency", y = "Value", title = "Boxplot of Frequencies") +
theme_classic() + scale_fill_manual(values=c("#d1ac6b","#10a4c2","#155d60"))
dev.off()
```

```
summary(aov(Value~Frequency*type, data=data))
TukeyHSD(aov(Value~Frequency*type, data=data))
```