Starling-May18
Projects/Katarina Stuart/KStuart.Starling-Aug18/Nc3_HihiSV/Analysis/2024-03-28.Recombination

PDF Version generated by

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

on

Aug 15, 2024 @03:08 PM NZST

Table of Contents

2024-03-28.Recombination

Loading [MathJax]/extensions/Safe.js

Katarina Stuart (z5188231@ad.unsw.edu.au) - Jul 08, 2024, 2:35 PM GMT+12

Loading [MathJax]/extensions/Safe.js

Recombination

Recombination: prepare bed file

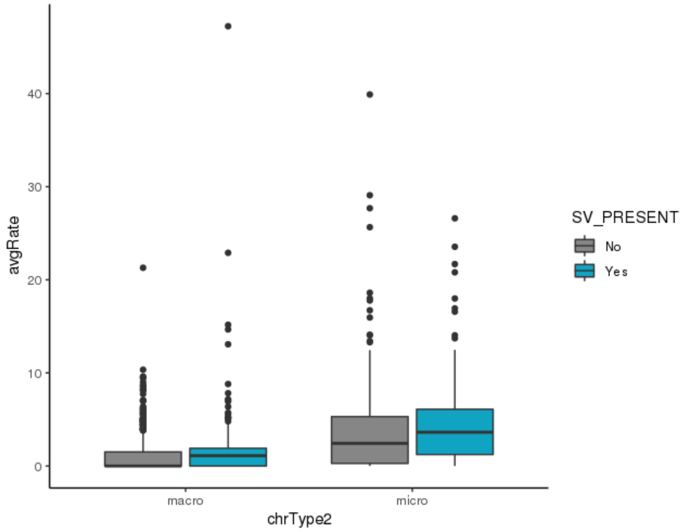
```
module load R/4.1.0-gimkl-2020a
R
library(ggplot2)
library(data.table)
library(tidyr)
library(dplyr)

setwd("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/recombination")

#recomb <- fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/data/data_20230227-physical-allIntervals-int1Mb-details-Kat20240328.txt")
recomb <- fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/data/data_20230227-physical-allIntervals-int500kb-details.txt")

recomb2 <-  recomb %>% mutate(START = phyStart*1000000, END = phyEnd*1000000, row_number = row_number()) %>% select(chr, START, END, avgRate, type, het, row_number, chrType2)

#Determine outlier sites

#what is the threshold
lower_critical_value <-  quantile(recomb2$het, 0.25) - 1.5 * IQR(recomb2$het)
higher_critical_value <-  quantile(recomb2$het, 0.75) + 1.5 * IQR(recomb2$het)

recomb2 <-  recomb2 %>% mutate(outlier = ifelse(het > higher_critical_value | het < lower_critical_value, "OUTLIER", "NO"))

#prevent scientific notation
options(scipen=999)
write.table(recomb2,"/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/recombination/recombination_heterochiasmy.bed",row.names=FALSE,sep="\t", quote = FALSE,col.names=TRUE)
```

SV recombination rates

```
cd /nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/recombination

#fix recombination file for chrom names
#Chrom 4A is 32, and 1A is 30. Replace this in file
awk -v OFS='\t'  '{if ($1 == 30) $1 = "1A"; print}' recombination_heterochiasmy.bed | awk -v OFS='\t'  '{if ($1 == 32) $1 = "4A"; print}' > recombination_heterochiasmy3.bed
tail -n +2 recombination_heterochiasmy.bed | awk -v OFS='\t'  '{if ($1 == 30) $1 = "1A"; print}' | awk -v OFS='\t'  '{if ($1 == 32) $1 = "4A"; print}' | bedtools sort > recombination_heterochiasmy2.bed


#SV
SVCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering/merged_rep_missfiltered.recode.vcf

grep -v "^#" ${SVCF} | sed 's/SVTYPE=\|SVLEN=\|SVLEN=-/\t/g' | cut -f 1,2,3,9,10 | sed 's/;/\t/g' | awk -v OFS='\t' '{ print $1"\t"$2"\t"$2+$4,$3}' | bedtools sort > SV.bed

bedtools closest -b recombination_heterochiasmy2.bed -a SV.bed -D b > SV_recomb.txt

#SNP
SNP=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/counts/hihi_wgs_filter_highcov_no83318_autosomes_reffix2.vcf
grep -v "^#" ${SNP} | awk -v OFS='\t' '{ print $1"\t"$2"\t"$2+$1,$3}' | bedtools sort > SNP.bed
bedtools closest -b recombination_heterochiasmy2.bed -a SNP.bed -D b > SNP_recomb.txt
```

Loading [MathJax]/extensions/Safe.js

```
#in R

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

setwd("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/recombination")

data <- fread("SV_recomb.txt")

impact <- fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/vep/vep_SV_impact_variant.txt") %>% mutate(IMPACT = "GENE")
vars <- fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/vep/vep_SV_impact_lowvariant.txt") %>% mutate(IMPACT2 = "LOW")
vars2 <- fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/vep/vep_SV_impact_midvariant.txt") %>% mutate(IMPACT3 = "LOW")

merged_df1 <- merge(impact , vars, by.x = "V1", by.y = "V1", all.x =  TRUE)
merged_df <- merge(merged_df1, vars2, by.x = "V1", by.y = "V1", all.x =  TRUE)
merged_df$IMPACT[merged_df$IMPACT2 == 'LOW'] <- 'LOW'
merged_df$IMPACT[merged_df$IMPACT3 == 'LOW'] <- 'LOW'
fun <- merge(data, merged_df[,1:2], by.x = "V4", by.y = "V1", all.x =  TRUE)
fun$IMPACT[is.na(fun$IMPACT)] <- 'NONE'

#LOOK AT SV AND RECOMBINATION RATES
recomb <- fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/recombination/recombination_heterochiasmy3.bed")
recomb <- recomb %>% mutate(KEY = paste0(chr,"_",START))
fun <- fun %>% mutate(KEY = paste0(V5,"_",V6))
hits <- unique(fun$KEY)
test <- recomb %>% mutate(SV_PRESENT = ifelse(KEY %in% unlist(strsplit(hits , ",")), "Yes", "No"))

model<- lm(avgRate~ SV_PRESENT*chrType2 , data=test )
summary(model)
anova(model)
TukeyHSD(aov(avgRate~ SV_PRESENT*chrType2, data=test))

aggregate(START ~ SV_PRESENT + chrType2, data = test, FUN = length)

pdf("Nc3_recombination_boxplot.pdf", width=7, height=5)
ggplot(test, aes(x=chrType2, y=avgRate, fill=SV_PRESENT)) + geom_boxplot() + theme_classic(base_size = 16) + scale_fill_manual(values=c("#868686", "#10a4c2"))
dev.off()
```



Loading [MathJax]/extensions/Safe.js

Linkage

```
cd /nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/linkage

module load BCFtools/1.13-GCC-9.2.0

VCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/counts/hihi_wgs_filter_highcov_no83318_autosomes_reffix2.vcf
SVCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering/merged_rep_missfiltered_reffix2.vcf

grep -v "^#" $SVCF | awk 'BEGIN { OFS = "\t" } { $3 = "SV_" $3; print }' > only.snps

cat $VCF only.snps | bcftools sort > SV_SNPs_joint.vcf

#List of SV variants, and subset list of SNPs x 5?
grep "SV_" SV_SNPs_joint.vcf | cut -f1-2 > SV_list.txt

for i in {1..5}
do
grep -v "SV_" SV_SNPs_joint.vcf | grep -v "^#" | shuf -n 935 | cut -f1-2 > SNP_list${i}.txt
done
```

linkage decay plot

```
#!/bin/bash -e

#SBATCH --job-name=2024_05_24.SV_SNP_pairwise_linkage.sl
#SBATCH --account=uoa02613
#SBATCH --time=00-48:00:00
#SBATCH --mem=5GB
#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=2
#SBATCH --profile task
#SBATCH --partition=milan
#SBATCH --array=1-5

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

cd /nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/linkage


#nned to not thin to get all SNPs. could thin rows after the fact?
vcftools --vcf SV_SNPs_joint.vcf --geno-r2-positions SV_list.txt --ld-window-bp 10000000  --out pairwise_SV_10000000_nothin

vcftools --vcf SV_SNPs_joint.vcf --geno-r2-positions SNP_list${SLURM_ARRAY_TASK_ID}.txt --ld-window-bp 10000000  --out
pairwise_SNP${SLURM_ARRAY_TASK_ID}_10000000_nothin
```

Plot

```
#!/bin/bash -e

#SBATCH --job-name=2023_03_30.pairwise_bin_macro.sl
=uoa02613
```

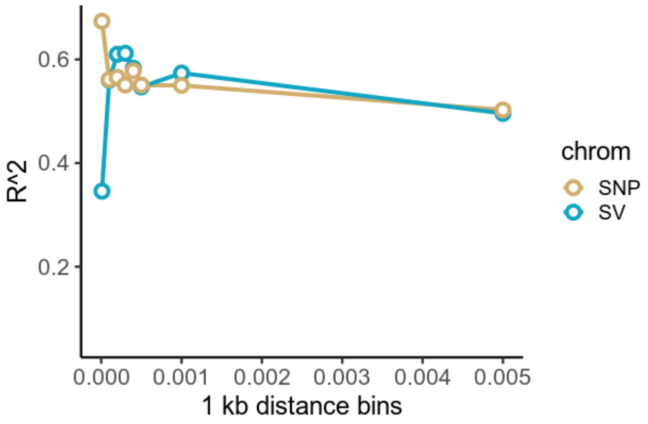
Loading [MathJax]/extensions/Safe.js

```
#SBATCH --time=00-12:00:00
#SBATCH --mem=1GB
#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=2
#SBATCH --profile task

cd /nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/linkage

#want to obtain the distance in BP between sites. Minus one SNP pos off another. And remove Nans.
#awk -v OFS="\t" '{print $1, $2-$4, $6}' pairwise_SV_10000000_nothin.list.geno.ld | grep -v "nan" > pairwise_SV_10000000_nothin.list.geno.ld_format.txt

##extra work on SVs - need to correct for the length of the SV. Change distance to take this into account, and remove overlapped SNPs (there is no equivalent of this in SNPs to compare to)
SVCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering/merged_rep_missfiltered.recode.vcf
grep -v "^#" ${SVCF} | sed 's/;/\t/g' | awk -v OFS="\t" '{print $1, $2, $3, $1"_"$2, $10}'  | sed 's/SVLEN=\|-//g' | sed 's/SVTYPE=//g' > x.test2
awk -v OFS="\t" '{print $0,$1"_"$2}' pairwise_SV_10000000_nothin.list.geno.ld > x.test
awk 'NR==FNR {a[$4]=$5; next} $7 in a {print $0, a[$7]}' x.test2 x.test > output1
awk -v OFS="\t" '{print $1, $2, $2-$4, $6, $8}' output1 | grep -v "nan" > output2
awk '{if ($3 < 0) $6 = $3 + $5; else $6 = $3; print}' output2 > output3
awk '{if (($3 < 0 && $6 < 0) || ($3 >= 0 && $6 >= 0)) $7 = "true"; else $7 = "overlap"; print}' output3 > output4
awk '$7 == "true"' output4 | awk -v OFS="\t" '{print $1, $2, $6, $4}'   > pairwise_SV_10000000_nothin.list.geno.ld_format.txt
###

rm SV_10Mb_outfile.txt

while read -r first second; do
    echo "$first" "$second"
awk -v start=${first} -v end=${second} '($3>start && $3<end)' pairwise_SV_10000000_nothin.list.geno.ld_format.txt | datamash -g 2 mean 4 | awk -v end=${second} '{print $0,end}'  >> SV_10Mb_outfile.txt
#awk -v start=${first} -v end=${second} '($3>start && $3<end)' pairwise_SV_10000000_nothin.list.geno.ld_format.txt | wc -l
echo "done"
done < interval_windows_10Mb_newversion.txt


#want to obtain the distance in BP between sites. Minus one SNP pos off another. And remove Nans.

for i in {1..5}
do

awk -v OFS="\t" '{print $1, $2, $2-$4, $6}' pairwise_SNP${i}_10000000_nothin.list.geno.ld | grep -v "nan" > pairwise_SNP${i}_10000000_nothin.list.geno.ld_format.txt

touch SNP${i}_10Mb_outfile.txt

while read -r first second; do
    echo "$first" "$second"
awk -v start=${first} -v end=${second} '($3>start && $3<end)' pairwise_SNP${i}_10000000_nothin.list.geno.ld_format.txt | datamash -g 2 mean 4 | awk -v end=${second} '{print $0,end}'   >> SNP${i}_10Mb_outfile.txt
echo "done"
done <  interval_windows_10Mb_newversion.txt
done
```

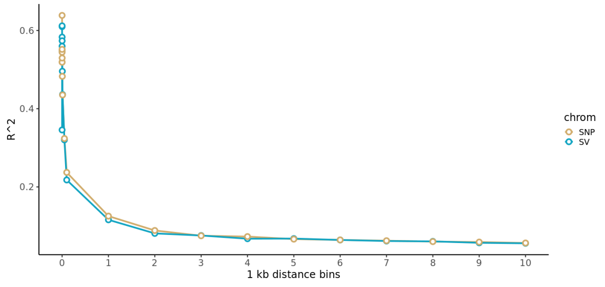
```
#Move into R
module load R/4.1.0-gimkl-2020a
R
setwd("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/linkage")
library(ggplot2)
library(dplyr)

SV_r <- read.table("SV_10Mb_outfile.txt", header=F) %>% group_by(V3) %>% summarise(avg = mean(V2))
SV_r$chrom <- c("SV")

SNP1_r <- read.table("SNP1_10Mb_outfile.txt", header=F) %>% group_by(V3) %>% summarise(avg = mean(V2))
"SNP")
```

Loading [MathJax]/extensions/Safe.js

```
SNP2_r <- read.table("SNP2_10Mb_outfile.txt", header=F) %>% group_by(V3) %>% summarise(avg2 = mean(V2))
SNP2_r$chrom <- c("SNP")

SNP3_r <- read.table("SNP3_10Mb_outfile.txt", header=F) %>% group_by(V3) %>% summarise(avg3 = mean(V2))
SNP3_r$chrom <- c("SNP")

SNP4_r <- read.table("SNP4_10Mb_outfile.txt", header=F) %>% group_by(V3) %>% summarise(avg4 = mean(V2))
SNP4_r$chrom <- c("SNP")

SNP5_r <- read.table("SNP5_10Mb_outfile.txt", header=F) %>% group_by(V3) %>% summarise(avg5 = mean(V2))
SNP5_r$chrom <- c("SNP")

SNPr2<-cbind(SNP1_r,SNP2_r[,2],SNP3_r[,2],SNP4_r[,2],SNP5_r[,2])
SNPr2$avg1<- rowMeans(SNPr2[, c("avg", "avg2", "avg3", "avg4", "avg5")])

#combine
r2<-rbind(SV_r,SNPr2[,1:3])


pdf("Nc3_linkagezoom.pdf")
ggplot(data=r2, mapping=aes(x=V3,y=avg,group=chrom)) + geom_line(size=1.4,aes(color=chrom))
+ geom_point(stroke=2,fill="white",size=3,shape=21,aes(color=chrom)) +
ylab("R^2") + xlab("1 kb distance bins") + theme_classic(base_size = 18) +
theme(axis.text=element_text(size=16))+scale_color_manual(values=c("#d1ac6b","#10a4c2"))+ xlim(0,10000)
dev.off()


pdf("Nc3_linkageall.pdf")
ggplot(data=r2, mapping=aes(x=V3,y=avg,group=chrom)) + geom_line(size=1.4,aes(color=chrom))
+ geom_point(stroke=2,fill="white",size=3,shape=21,aes(color=chrom)) +
ylab("R^2") + xlab("1 kb distance bins") + theme_classic(base_size = 18) +
theme(axis.text=element_text(size=16))+scale_color_manual(values=c("#d1ac6b","#10a4c2"))
dev.off()
```



zoom out for supp mat:



Loading [MathJax]/extensions/Safe.js

```
cd /nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/linkage

#want to obtain the distance in BP between sites. Minus one SNP pos off another. And remove Nans.
#awk -v OFS="\t" '{print $1, $2-$4, $6}' pairwise_SV_10000000_nothin.list.geno.ld | grep -v "nan" > pairwise_SV_10000000_nothin.list.geno.ld_format.txt

##extra work on SVs - need to correct for the length of the SV. Change distance to take this into account, and remove overlapped SNPs (there is no equivalent of this in SNPs to compare to)
SVCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering/merged_rep_missfiltered.recode.vcf
grep -v "^#" ${SVCF} | sed 's/;/\t/g' | awk -v OFS="\t" '{print $1, $2, $3, $1"_"$2, $10}'  | sed 's/SVLEN=\|-//g' | sed 's/SVTYPE=//g' > x.test2
awk -v OFS="\t" '{print $0,$1"_"$2}' pairwise_SV_10000000_nothin.list.geno.ld > x.test
awk 'NR==FNR {a[$4]=$5; next} $7 in a {print $0, a[$7]}' x.test2 x.test > output1
awk -v OFS="\t" '{print $1, $2-$4, $6, $8}' output1 | grep -v "nan" > output2
awk '{if ($2 < 0) $5 = $2 + $4; else $5 = $2; print}' output2 > output3
awk '{if (($2 < 0 && $5 < 0) || ($2 >= 0 && $5 >= 0)) $6 = "true"; else $6 = "overlap"; print}' output3 > output4
awk '$6 == "true"' output4 | awk -v OFS="\t" '{print $1, $5, $3}'   > pairwise_SV_10000000_nothin.list.geno.ld_format.txt
awk '$1 ~ /^(1|2|3|4|5|6|7|1A)$/' pairwise_SV_10000000_nothin.list.geno.ld_format.txt > pairwise_SV_10000000_nothin.list.geno.ld_format_macro.txt
awk '$1 !~ /^(1|2|3|4|5|6|7|1A)$/' pairwise_SV_10000000_nothin.list.geno.ld_format.txt > pairwise_SV_10000000_nothin.list.geno.ld_format_micro.txt
###

rm SV_10Mb_outfile.txt

while read -r first second; do
    echo "$first" "$second"
awk -v start=${first} -v end=${second} '($2>start && $2<end)' pairwise_SV_10000000_nothin.list.geno.ld_format_macro.txt | datamash mean 3 >> SV_10Mb_macro_outfile.txt
awk -v start=${first} -v end=${second} '($2>start && $2<end)' pairwise_SV_10000000_nothin.list.geno.ld_format_micro.txt | datamash mean 3 >> SV_10Mb_micro_outfile.txt
echo "done"
done < interval_windows_10Mb_newversion.txt


#want to obtain the distance in BP between sites. Minus one SNP pos off another. And remove Nans.

for i in {1..5}
do

awk -v OFS="\t" '{print $1, $2-$4, $6}' pairwise_SNP${i}_10000000_nothin.list.geno.ld | grep -v "nan" > pairwise_SNP${i}_10000000_nothin.list.geno.ld_format.txt
awk '$1 ~ /^(1|2|3|4|5|6|7|1A)$/' pairwise_SNP${i}_10000000_nothin.list.geno.ld_format.txt > pairwise_SNP${i}_10000000_nothin.list.geno.ld_format_macro.txt
awk '$1 !~ /^(1|2|3|4|5|6|7|1A)$/' pairwise_SNP${i}_10000000_nothin.list.geno.ld_format.txt > pairwise_SNP${i}_10000000_nothin.list.geno.ld_format_micro.txt

touch SNP${i}_10Mb_outfile.txt

while read -r first second; do
    echo "$first" "$second"
awk -v start=${first} -v end=${second} '($2>start && $2<end)' pairwise_SNP${i}_10000000_nothin.list.geno.ld_format_macro.txt | datamash mean 3 >> SNP${i}_10Mb_macro_outfile.txt
awk -v start=${first} -v end=${second} '($2>start && $2<end)' pairwise_SNP${i}_10000000_nothin.list.geno.ld_format_micro.txt | datamash mean 3 >> SNP${i}_10Mb_micro_outfile.txt
echo "done"
done <  interval_windows_10Mb_newversion.txt
done




#Move into R
module load R/4.1.0-gimkl-2020a
R
setwd("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/linkage")
library(ggplot2)

intervals <- read.table("interval_windows_10Mb_newversion.txt", sep="\t", header=F)

SVb_r <- read.table("SV_10Mb_macro_outfile.txt", sep="\t", header=F)
SVb_r$order <- (intervals[,2] / 1000000)
SVb_r$chrom <- c("SV")
SVb_r$class <- c("macro")
```

Loading [MathJax]/extensions/Safe.js

```
SVs_r <- read.table("SV_10Mb_micro_outfile.txt", sep="\t", header=F)
SVs_r$order <- (intervals[,2] / 1000000)
SVs_r$chrom <- c("SV")
SVs_r$class <- c("micro")

##

SNP1b_r <- read.table("SNP1_10Mb_macro_outfile.txt", sep="\t", header=F)
SNP1b_r$order <- (intervals[,2] / 1000000)
SNP1b_r$chrom <- c("SNP")
SNP1b_r$class <- c("macro")

SNP1s_r <- read.table("SNP1_10Mb_micro_outfile.txt", sep="\t", header=F)
SNP1s_r$order <- (intervals[,2] / 1000000)
SNP1s_r$chrom <- c("SNP")
SNP1s_r$class <- c("micro")

SNP2b_r <- read.table("SNP2_10Mb_macro_outfile.txt", sep="\t", header=F)
SNP2b_r$order <- (intervals[,2] / 1000000)
SNP2b_r$chrom <- c("SNP")
SNP2b_r$class <- c("macro")

SNP2s_r <- read.table("SNP2_10Mb_micro_outfile.txt", sep="\t", header=F)
SNP2s_r$order <- (intervals[,2] / 1000000)
SNP2s_r$chrom <- c("SNP")
SNP2s_r$class <- c("micro")

SNP3b_r <- read.table("SNP3_10Mb_macro_outfile.txt", sep="\t", header=F)
SNP3b_r$order <- (intervals[,2] / 1000000)
SNP3b_r$chrom <- c("SNP")
SNP3b_r$class <- c("macro")

SNP3s_r <- read.table("SNP3_10Mb_micro_outfile.txt", sep="\t", header=F)
SNP3s_r$order <- (intervals[,2] / 1000000)
SNP3s_r$chrom <- c("SNP")
SNP3s_r$class <- c("macro")

SNP4b_r <- read.table("SNP4_10Mb_macro_outfile.txt", sep="\t", header=F)
SNP4b_r$order <- (intervals[,2] / 1000000)
SNP4b_r$chrom <- c("SNP")
SNP4b_r$class <- c("macro")

SNP4s_r <- read.table("SNP4_10Mb_micro_outfile.txt", sep="\t", header=F)
SNP4s_r$order <- (intervals[,2] / 1000000)
SNP4s_r$chrom <- c("SNP")
SNP4s_r$class <- c("micro")

SNP5b_r <- read.table("SNP5_10Mb_macro_outfile.txt", sep="\t", header=F)
SNP5b_r$order <- (intervals[,2] / 1000000)
SNP5b_r$chrom <- c("SNP")
SNP5b_r$class <- c("macro")

SNP5s_r <- read.table("SNP5_10Mb_micro_outfile.txt", sep="\t", header=F)
SNP5s_r$order <- (intervals[,2] / 1000000)
SNP5s_r$chrom <- c("SNP")
SNP5s_r$class <- c("micro")

SNPr2b<-cbind(SNP1b_r,SNP2b_r[,1],SNP3b_r[,1],SNP4b_r[,1],SNP5b_r[,1])
SNPr2b$V1<- rowMeans(SNPr2b[, c("V1", "SNP2b_r[, 1]", "SNP3b_r[, 1]", "SNP4b_r[, 1]", "SNP5b_r[, 1]")])

SNPr2s<-cbind(SNP1s_r,SNP2s_r[,1],SNP3s_r[,1],SNP4s_r[,1],SNP5s_r[,1])
SNPr2s$V1<- rowMeans(SNPr2s[, c("V1", "SNP2s_r[, 1]", "SNP3s_r[, 1]", "SNP4s_r[, 1]", "SNP5s_r[, 1]")])

#combine
r2<-rbind(SVb_r,SVs_r,SNPr2b[,1:4],SNPr2s[,1:4])

r2b <- r2 %>% mutate(col = paste0(chrom,"_",class))

pdf("Nc3_linkagezoom_macromicro.pdf")
```

Loading [MathJax]/extensions/Safe.js

```
ggplot(data=r2b, mapping=aes(x=order,y=V1,group=col )) + geom_line(size=1.4,aes(color=col ))
+ geom_point(stroke=2,fill="white",size=3,shape=21,aes(color=col )) +
ylab("R^2") + xlab("1 kb distance bins") + theme_classic(base_size = 18) +
theme(axis.text=element_text(size=16))+scale_color_manual(values=c("#d1ac6b","#A78448","#10a4c2","#19788C"))+ scale_x_continuous(breaks = seq(0, 10, by = 1)) + xlim(0,0.005)
dev.off()
```

MAF

```
cd /nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/recombination

#MAF of SV
SVCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering/merged_rep_missfiltered_reffix2.vcf
module load VCFtools/0.1.15-GCC-9.2.0-Perl-5.30.1
vcftools --vcf $SVCF --freq --out merged_rep_missfiltered_reffix2
tail -n +2 merged_rep_missfiltered_reffix2.frq | sed 's/DUP:TANDEM/DUP/g' | sed 's/:/\t/g' > merged_rep_missfiltered_reffix2.fix.frq

#MAF of SNP
SNP=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/counts/hihi_wgs_filter_highcov_no83318_autosomes_reffix2.vcf
module load VCFtools/0.1.15-GCC-9.2.0-Perl-5.30.1
vcftools --vcf $SNP --freq --out hihi_wgs_filter_highcov_no83318_autosomes_reffix2
tail -n +2 hihi_wgs_filter_highcov_no83318_autosomes_reffix2.frq |  sed 's/:/\t/g' > hihi_wgs_filter_highcov_no83318_autosomes_reffix2.fix.frq

#in R

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

setwd("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/recombination")

#SV
data <- fread("SV_recomb.txt")

impact <- fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/vep/vep_SV_impact_variant.txt") %>% mutate(IMPACT = "LOAD")
vars <- fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/vep/vep_SV_impact_lowvariant.txt") %>% mutate(IMPACT2 = "LOAD")
vars2 <- fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/vep/vep_SV_impact_midvariant.txt") %>% mutate(IMPACT3 = "LOAD")

#assigning impact identity (does it contribute to load) to variant
merged_df1 <- merge(impact , vars, by.x = "V1", by.y = "V1", all.x =  TRUE)
merged_df <- merge(merged_df1, vars2, by.x = "V1", by.y = "V1", all.x =  TRUE)
merged_df$IMPACT[merged_df$IMPACT2 == 'LOAD'] <- 'LOAD'
merged_df$IMPACT[merged_df$IMPACT3 == 'LOAD'] <- 'LOAD'
fun <- merge(data, merged_df[,1:2], by.x = "V4", by.y = "V1", all.x =  TRUE)
fun$IMPACT[is.na(fun$IMPACT)] <- 'NONE'
```

Loading [MathJax]/extensions/Safe.js

```
##combine wth MAF
maf <- fread("merged_rep_missfiltered_reffix2.fix.frq")[,c(1,2,8)]
maf <- maf %>% mutate(KEY = paste0(V1,"_",V2))
fun <- fun %>% mutate(KEY = paste0(V1,"_",V2))
fun2 <- merge(fun , maf , by.x = "KEY", by.y = "KEY", all.x =  TRUE)
#not all chroms in the recombination map - need to remove. About 100 gone :(
fun3 <- fun2%>% filter(V5 != ".")
fun3$V8.x <- as.numeric(fun3$V8.x)
fun3SV <- fun3 %>% mutate(FREQ = ifelse(V8.y > 0.35, "COMMON", ifelse(V8.y < 0.15, "RARE", "MID")))

colnames(fun3SV) <-
c("KEY","ID","CHROM","START","END","CHROM2","BINSTART","BINEND","AVGRECOM","Heterochiasmy","measure","row_num","chrom_class","OUTLIER",
"ignore","coding_impact","CHROM3","START2","MAF","MAF_class")

fun3SV <- fun3SV %>% mutate(group_sex_out  = paste0(Heterochiasmy,"_",OUTLIER))

# Calculate means and standard errors
summary_data_SV <- fun3SV %>%
  group_by(chrom_class, coding_impact) %>%
  summarise(mean_value = mean(MAF),
            std_error = sd(MAF) / sqrt(n()))  # Calculate standard error instead of standard deviation

#SNP
data <- fread("SNP_recomb.txt")

impact <- fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/vep/vep_SNP_impact_variant.txt", header = FALSE) %>% mutate(IMPACT
= "LOAD")
vars <- fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/vep/vep_SNP_impact_lowvariant.txt", header = FALSE) %>%
mutate(IMPACT2 = "LOAD")
vars2 <- fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/vep/vep_SNP_impact_midvariant.txt", header = FALSE) %>%
mutate(IMPACT3 = "LOAD")

#assigning impact identity (does it contribute to load) to variant
merged_df1 <- merge(impact , vars, by.x = "V1", by.y = "V1", all.x =  TRUE)
merged_df <- merge(merged_df1, vars2, by.x = "V1", by.y = "V1", all.x =  TRUE)
merged_df$IMPACT[merged_df$IMPACT2 == 'LOAD'] <- 'LOAD'
merged_df$IMPACT[merged_df$IMPACT3 == 'LOAD'] <- 'LOAD'
fun <- merge(data, merged_df[,1:2], by.x = "V4", by.y = "V1", all.x =  TRUE)
fun$IMPACT[is.na(fun$IMPACT)] <- 'NONE'

##combine wth MAF
maf <- fread("hihi_wgs_filter_highcov_no83318_autosomes_reffix2.fix.frq")[,c(1,2,8)]
maf <- maf %>% mutate(KEY = paste0(V1,"_",V2))
fun <- fun %>% mutate(KEY = paste0(V1,"_",V2))
fun2 <- merge(fun , maf , by.x = "KEY", by.y = "KEY", all.x =  TRUE)
#not all chroms in the recombination map - need to remove. About 100 gone :(
fun3 <- fun2%>% filter(V5 != ".")
fun3$V8.x <- as.numeric(fun3$V8.x)
fun3 <- fun3 %>% mutate(FREQ = ifelse(V8.y > 0.35, "COMMON", ifelse(V8.y < 0.15, "RARE", "MID")))
fun3SNP <- fun3
colnames(fun3SNP) <-
c("KEY","ID","CHROM","START","END","CHROM2","BINSTART","BINEND","AVGRECOM","Heterochiasmy","measure","row_num","chrom_class","OUTLIER",
"ignore","coding_impact","CHROM3","START2","MAF","MAF_class")

summary_data_SNP <- fun3SNP %>%
  group_by(chrom_class, coding_impact) %>%
  summarise(mean_value = mean(MAF),
            std_error = sd(MAF) / sqrt(n()))  # Calculate standard error instead of standard deviation

#combine SNP and SV
summary_data_SV <- summary_data_SV %>% mutate(Variant = "SV")
summary_data_SNP <- summary_data_SNP %>% mutate(Variant = "SNP")
MAF_data <- rbind(summary_data_SV, summary_data_SNP)

pdf("Nc3_SV_SNP_MAF_loads.pdf", width=6, height=4)
ggplot(MAF_data , aes(x =chrom_class , y = mean_value, fill = coding_impact )) +
  geom_point(position = position_dodge(width = 0.75), size = 3, shape = 21) +
  geom_errorbar(aes(ymin = mean_value - std_error, ymax = mean_value + std_error),
                position = position_dodge(width = 0.75), width = 0.2) +  # Plot standard errors
```

Loading [MathJax]/extensions/Safe.js

```
  labs(x = "Type", y = "MAF") +
  theme_classic(base_size = 18) +
  scale_fill_manual(values = c("#d1ac6b", "#10a4c2")) +
  scale_color_manual(values = c("#d1ac6b", "#10a4c2")) +
  facet_grid(. ~ Variant)
dev.off()
```

Loading [MathJax]/extensions/Safe.js