Starling-May18
Projects/Katarina Stuart/KStuart.Starling-Aug18/Nc3_HihiSV/Analysis/2024-02-29.ROHanalysis

PDF Version generated by

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

on

Aug 15, 2024 @03:07 PM NZST

Table of Contents

2024-02-29.ROHanalysis

Katarina Stuart (z5188231@ad.unsw.edu.au) - Jul 08, 2024, 2:50 PM GMT+12

ROH analysis & Variant Effects

Variant effects - VEP

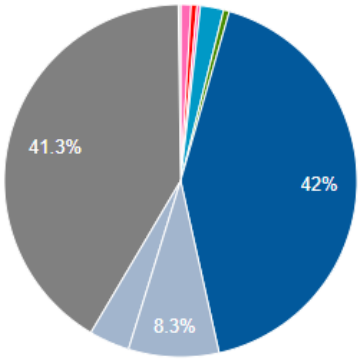
```
#!/bin/bash -e

#SBATCH --job-name=2024_03_22.VEP_snp_annotation.sl
#SBATCH --account=uoa00338
#SBATCH --time=00-12:00:00
#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

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

GFF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/data/annotation/Ncf_H98617_scaffolded_liftoff.gff
GENOME=/nesi/nobackup/uoa00338/kstuart_projects/Nc2_HihiWGS/data/resources/Ncf_H98617_scaffolded_genome.fa

module purge
module load AGAT/1.0.0-gimkl-2022a-Perl-5.34.1-R-4.2.1
module load gffread/0.12.7-GCC-11.3.0

agat_sp_keep_longest_isoform.pl --gff $GFF -o Ncf_H98617_scaffolded_liftoff_longestIsoform.gff

grep -v "#" Ncf_H98617_scaffolded_liftoff_longestIsoform.gff | awk 'BEGIN{OFS=FS="\t"} $3=="transcript" {$9=$9";biotype=protein_coding"} {print}' | sort -k1,1 -k4,4n -
k5,5n -t$'\t' | bgzip -c > data.gff.gz
tabix -p gff data.gff.gz

module purge
module load VEP/107.0-GCC-11.3.0-Perl-5.34.1

#vep
SNP=/nesi/nobackup/uoa00338/kstuart_projects/Nc2_HihiWGS/data/snp_variants_updated/hihi_wgs_filter_highcov_no83318_autosomes.recode.vcf
vep -i $SNP --gff data.gff.gz --fasta $GENOME -o vep_SNP

SVCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering/merged_rep_missfiltered.recode.vcf
vep -i $SVCF --gff data.gff.gz --fasta $GENOME -o vep_SV
```

Collate results to identify as genes as intergenic or not

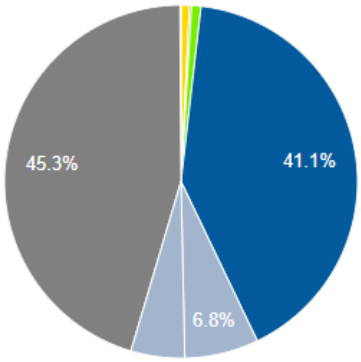
```
cd /nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/vep

#some variants might have a few different types of hits. Anything not an intergenic variant can be seen as a high impact variant then?
grep -v "^##" vep_SV | grep -v intergenic_variant | cut -f1 | sort | uniq > vep_SV_impact_variant.txt
grep -v "^##" vep_SNP | grep -v intergenic_variant | cut -f1 | sort | uniq > vep_SNP_impact_variant.txt

grep -v "^##" vep_SV | grep -v intergenic_variant | grep -v "coding_sequence_variant\|frameshift_variant\|inframe_deletion\|splice_donor_variant\|transcript_ablation" | grep
"stream" | cut -f1 | sort | uniq > vep_SV_impact_lowvariant.txt
grep -v "^##" vep_SNP | grep -v intergenic_variant | grep -v
"stop_\|start_\|missense_variant\|splice_acceptor\|splice_donor\|synonymous_variant\|splice_polypyrimidine_tract_variant\|splice_region_variant\|coding_sequence_variant"
| grep "stream" | cut -f1 | sort | uniq > vep_SNP_impact_lowvariant.txt
```

```
grep -v "^##" vep_SV | grep -v intergenic_variant | grep -v "coding_sequence_variant\|frameshift_variant\|inframe_deletion\|splice_donor_variant\|transcript_ablation" | grep -v "stream" | cut -f1 | sort | uniq > vep_SV_impact_midvariant.txt
grep -v "^##" vep_SNP | grep -v intergenic_variant | grep -v "stop_\|start_\|missense_variant\|splice_acceptor\|splice_donor\|synonymous_variant\|splice_polypyrimidine_tract_variant\|splice_region_variant\|coding_sequence_variant" | grep -v "stream" | cut -f1 | sort | uniq > vep_SNP_impact_midvariant.txt



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

SVCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering/merged_rep_missfiltered.recode.vcf
vcftools --vcf ${SVCF} --snps vep_SV_impact_variant.txt --recode-INFO-all --recode --out merged_rep_missfiltered_genic
vcftools --vcf ${SVCF} --exclude vep_SV_impact_variant.txt --recode-INFO-all --recode --out merged_rep_missfiltered_intergenic


SNP=/nesi/nobackup/uoa00338/kstuart_projects/Nc2_HihiWGS/data/snp_variants_updated/hihi_wgs_filter_highcov_no83318_autosomes.recode.vcf
#doesn't have SNP IDs - wtf? My bad.
awk -F'\t' '!/^#/ { OFS="\t"; $3 = $1"_"$2"_"$4"/"$5 } 1' $SNP > hihi_wgs_filter_highcov_no83318_autosomes_vepID.recode.vcf

vcftools --vcf hihi_wgs_filter_highcov_no83318_autosomes_vepID.recode.vcf --snps vep_SNP_impact_variant.txt --recode-INFO-all --recode --
out hihi_wgs_filter_highcov_no83318_autosomes_genic
vcftools --vcf hihi_wgs_filter_highcov_no83318_autosomes_vepID.recode.vcf --exclude vep_SNP_impact_variant.txt --recode-INFO-all --recode --
out hihi_wgs_filter_highcov_no83318_autosomes_intergenic
```

SVs



SNPs



Calculating Load

Per-individual SV counts: presence absence

```
cd /nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/counts

#Need to use the SV file where the ref is encoded as the major allele
SVCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering/merged_rep_missfiltered_reffix2.vcf

#change 1/1 genotypes to 0/1/ - therefore a minor allele count (MAC) in the method above will reflect presence/of SV allele in the data
#also need to trick the file format for vcftools
sed 's/1\/1/0\/1/g' $SVCF | grep -v autosomePairCt | sed 's/VCFv4.3/VCFv4.2/g' > merged_rep_missfiltered_reffix2_allhet.recode.vcf

vcftools --vcf merged_rep_missfiltered_reffix2_allhet.recode.vcf --het

awk '{print $0,"\t",$4-$2}' out.het > het_count_SVs.txt
```

Per-individual SV counts: load

```
cd /nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/counts

#SV
#Need to use the SV file where the ref is encoded as the major allele
SVCF=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/filtering/merged_rep_missfiltered_reffix2.vcf

#change 1/1 to 2 and  0/1 to 1 - can then sum per individual to quantify SV load
#also need to trick the file format for vcftools
sed -e 's/0\/0/0/g' -e 's/0\/1/1/g' -e 's/1\/1/2/g' $SVCF | grep -v "^##" > merged_rep_missfiltered_reffix2_load.txt


#SNP
#Need to use the SV file where the ref is encoded as the major allele - for SNP too?
SNP=/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/vep/hihi_wgs_filter_highcov_no83318_autosomes_vepID.recode.vcf
module purge
module load PLINK/2.00a2.3
plink2 --vcf $SNP --allow-extra-chr --chr-set 28 --make-bed --maj-ref --out hihi_wgs_filter_highcov_no83318_autosomes_reffix2.plink
plink2 --bfile hihi_wgs_filter_highcov_no83318_autosomes_reffix2.plink --allow-extra-chr --chr-set 28 --recode vcf --
out hihi_wgs_filter_highcov_no83318_autosomes_reffix2

#change 1/1 to 2 and  0/1 to 1 - can then sum per individual to quantify SV load
#also need to trick the file format for vcftools
sed -e 's/0\/0/0/g' -e 's/0\/1/1/g' -e 's/1\/1/2/g' hihi_wgs_filter_highcov_no83318_autosomes_reffix2.vcf | grep -v "^##"
> hihi_wgs_filter_highcov_no83318_autosomes_load.txt


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/counts")

#SV
data <- fread("merged_rep_missfiltered_reffix2_load.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 = "MID")

#assigning impact to each SV
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 == 'MID'] <- 'MID'
fun <- merge(data, merged_df[,1:2], by.x = "ID", by.y = "V1", all.x =  TRUE)
fun$IMPACT[is.na(fun$IMPACT)] <- 'NONE'

#alternate allele count (count of total alt alleles under each SV category). This is not presence absence.
SV_no_impact_count <- fun %>% filter(IMPACT=="NONE") %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%
 group_by(variable) %>%   summarize(SV_no_impact_count = sum(value, na.rm = TRUE))
SV_low_impact_count <- fun %>% filter(IMPACT=="LOW") %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%
 group_by(variable) %>%   summarize(SV_low_impact_count = sum(value, na.rm = TRUE))
SV_mid_impact_count <- fun %>% filter(IMPACT=="MID") %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%
 group_by(variable) %>%   summarize(SV_mid_impact_count = sum(value, na.rm = TRUE))
SV_high_impact_count <- fun %>% filter(IMPACT=="GENE") %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%
 group_by(variable) %>%   summarize(SV_high_impact_count = sum(value, na.rm = TRUE))
SV_count <- fun %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%  group_by(variable) %>%   summarize(SV_count =
sum(value, na.rm = TRUE))

#count of genotypes (het and homo) in gene impacting regions
SV_none_load <- fun %>% filter(IMPACT=="LOW" | IMPACT=="MID" |  IMPACT=="GENE"  ) %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value =
as.numeric(value))  %>%  group_by(variable) %>% summarize(SV_none_load = sum(value == 0, na.rm = TRUE))

SV_masked_load <- fun %>% filter(IMPACT=="LOW" | IMPACT=="MID" |  IMPACT=="GENE"  ) %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value =
as.numeric(value))  %>%  group_by(variable) %>% summarize(SV_masked_load = sum(value == 1, na.rm = TRUE))

SV_realised_load<- fun %>% filter(IMPACT=="LOW" | IMPACT=="MID" | IMPACT=="GENE"  ) %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value =
```

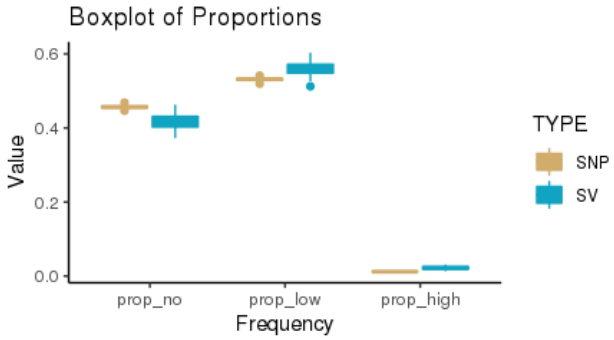
```
as.numeric(value))  %>%  group_by(variable) %>% summarize(SV_realised_load = sum(value == 2, na.rm = TRUE))

SV_putative_load<- fun %>% filter(IMPACT=="LOW" | IMPACT=="MID" | IMPACT=="GENE"  ) %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%  group_by(variable) %>% summarize(SV_putative_load= sum(value %in% c(0, 1, 2), na.rm = TRUE))

SV_none_all <- fun %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%  group_by(variable) %>% summarize(SV_none_all = sum(value == 0, na.rm = TRUE))

SV_masked_all <- fun %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%  group_by(variable) %>% summarize(SV_masked_all = sum(value == 1, na.rm = TRUE))

SV_realised_all<- fun %>%  gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%  group_by(variable) %>% summarize(SV_realised_all = sum(value == 2, na.rm = TRUE))

SV_putative_all<- fun %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%  group_by(variable) %>% summarize(SV_putative_all = sum(value %in% c(0, 1, 2), na.rm = TRUE))



#total genotyped sites per ind
SV_data2 <- fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/heterozygosity/merged_rep_missfiltered.het") %>% mutate(TOTAL = 942) %>% mutate(missing = TOTAL-N_SITES)

final <- cbind(SV_count , SV_no_impact_count[,2], SV_low_impact_count[,2], SV_mid_impact_count[,2], SV_high_impact_count[,2], SV_none_load[,2], SV_masked_load[,2], SV_realised_load[,2],  SV_putative_load[,2], SV_none_all[,2], SV_masked_all[,2], SV_realised_all[,2],  SV_putative_all[,2],  SV_data2[,c(4,7)] )

final2 <- final %>% mutate(SV_prop_none = SV_none_load/SV_putative_load, SV_prop_mask = SV_masked_load/SV_putative_load, SV_prop_real = (SV_realised_load)/SV_putative_load,
SV_prop_noneall = SV_none_all/SV_putative_all, SV_prop_maskall = SV_masked_all/SV_putative_all, SV_prop_realall = (SV_realised_all)/SV_putative_all)

#final <- cbind(SV_count , SV_no_impact_count[,2], SV_low_impact_count[,2], SV_mid_impact_count[,2], SV_high_impact_count[,2], SV_masked_load[,2], SV_realised_load[,2], SV_data2[,c(4,7)] )
#final2 <- final %>% mutate(SV_prop_mask = SV_masked_load/N_SITES, SV_prop_real = (SV_realised_load)/N_SITES)

write.table(final2,"SV_load_counts.txt",row.names=FALSE,sep="\t", quote = FALSE,col.names=TRUE)


#SNP
data <- fread("hihi_wgs_filter_highcov_no83318_autosomes_load.txt")
impact <- fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/vep/vep_SNP_impact_variant.txt", header=F) %>% mutate(IMPACT = "GENE")
vars <- fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/vep/vep_SNP_impact_lowvariant.txt", header=F) %>% mutate(IMPACT2 = "LOW")
vars2 <- fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/vep/vep_SNP_impact_midvariant.txt", header=F) %>% mutate(IMPACT3 = "MID")

#assigning impact to each SNP
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 == 'MID'] <- 'MID'
fun <- merge(data, merged_df[,1:2], by.x = "ID", by.y = "V1", all.x =  TRUE)
fun$IMPACT[is.na(fun$IMPACT)] <- 'NONE'

#alternate allele count (count of total alt alleles under each SV category). This is not presence absence.
SNP_no_impact_count <- fun %>% filter(IMPACT=="NONE") %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%
 group_by(variable) %>%   summarize(SNP_no_impact_count = sum(value, na.rm = TRUE))
SNP_low_impact_count <- fun %>% filter(IMPACT=="LOW") %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%
 group_by(variable) %>%   summarize(SNP_low_impact_count = sum(value, na.rm = TRUE))
SNP_mid_impact_count <- fun %>% filter(IMPACT=="MID") %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%
 group_by(variable) %>%   summarize(SNP_mid_impact_count = sum(value, na.rm = TRUE))
SNP_high_impact_count <- fun %>% filter(IMPACT=="GENE") %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%
 group_by(variable) %>%   summarize(SNP_high_impact_count = sum(value, na.rm = TRUE))
SNP_count <- fun %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%  group_by(variable) %>%   summarize(SNP_count = sum(value, na.rm = TRUE))

#count of genotypes (het and homo) in gene impacting regions
SNP_none_load <- fun %>% filter(IMPACT=="LOW" | IMPACT=="MID" |  IMPACT=="GENE"  ) %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%  group_by(variable) %>% summarize(SNP_none_load = sum(value == 0, na.rm = TRUE))

SNP_masked_load <- fun %>% filter(IMPACT=="LOW" | IMPACT=="MID" |  IMPACT=="GENE"  ) %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%  group_by(variable) %>% summarize(SNP_masked_load = sum(value == 1, na.rm = TRUE))
```

```
SNP_realised_load<- fun %>% filter(IMPACT=="LOW" | IMPACT=="MID" | IMPACT=="GENE"  ) %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%  group_by(variable) %>% summarize(SNP_realised_load = sum(value == 2, na.rm = TRUE))

SNP_putative_load<- fun %>% filter(IMPACT=="LOW" | IMPACT=="MID" | IMPACT=="GENE"  ) %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%  group_by(variable) %>% summarize(SNP_putative_load= sum(value %in% c(0, 1, 2), na.rm = TRUE))

SNP_none_all <- fun %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%  group_by(variable) %>% summarize(SNP_none_all = sum(value == 0, na.rm = TRUE))

SNP_masked_all <- fun %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%  group_by(variable) %>% summarize(SNP_masked_all = sum(value == 1, na.rm = TRUE))

SNP_realised_all <- fun %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%  group_by(variable) %>% summarize(SNP_realised_all = sum(value == 2, na.rm = TRUE))

SNP_putative_all<- fun %>% gather(key = "variable", value = "value", 10:39) %>% mutate(value = as.numeric(value))  %>%  group_by(variable) %>% summarize(SNP_putative_all= sum(value %in% c(0, 1, 2), na.rm = TRUE))

#total genotyped sites per ind
SNP_data2 <- fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/SV_profiling/heterozygosity/hihi_wgs_filter_highcov_no83318_autosomes.het") %>% mutate(SNPTOTAL = 3111629 ) %>% mutate(SNPmissing = SNPTOTAL -N_SITES)
names(SNP_data2)[names(SNP_data2) == "N_SITES"] <- "N_SITES_SNP"

final <- cbind(SNP_count , SNP_no_impact_count[,2], SNP_low_impact_count[,2], SNP_mid_impact_count[,2], SNP_high_impact_count[,2], SNP_none_load[,2], SNP_masked_load[,2], SNP_realised_load[,2], SNP_putative_load[,2], SNP_none_all[,2], SNP_masked_all[,2], SNP_realised_all[,2], SNP_putative_all[,2],
 SNP_data2[,c(4,7)] )

final2 <- final %>% mutate(SNP_prop_none = SNP_none_load/SNP_putative_load, SNP_prop_mask = SNP_masked_load/SNP_putative_load, SNP_prop_real = (SNP_realised_load)/SNP_putative_load , SNP_prop_noneall = SNP_none_all/SNP_putative_all, SNP_prop_maskall = SNP_masked_all/SNP_putative_all, SNP_prop_realall = (SNP_realised_all)/SNP_putative_all )

write.table(final2,"SNP_load_counts.txt",row.names=FALSE,sep="\t", quote = FALSE,col.names=TRUE)

#combine
SVload <- fread("SV_load_counts.txt")
SNPload <- fread("SNP_load_counts.txt")

allload0 <- merge(SVload,SNPload, by.x="variable", by.y="variable")

write.table(allload0 ,"ALL_load_counts.txt",row.names=FALSE,sep="\t", quote = FALSE,col.names=TRUE)
```

Assessment of load

```
module load R/4.1.0-gimkl-2020a
R
library(ggplot2)
library(data.table)
library(tidyr)
library(dplyr)
library(stringr)
library(lme4)

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

allload0<- fread("/nesi/nobackup/uoa00338/kstuart_projects/Nc3_HihiSV/analysis/impacts/counts/ALL_load_counts.txt")

allload <- allload0 %>% mutate(SV_None = SV_no_impact_count/SV_count, SV_UpDownstream = (SV_low_impact_count)/SV_count, SV_Intron = (SV_mid_impact_count)/SV_count, SV_Gene = SV_high_impact_count/SV_count, SNP_None = SNP_no_impact_count/SNP_count, SNP_UpDownstream = (SNP_low_impact_count)/SNP_count, SNP_Intron = (SNP_mid_impact_count)/SNP_count, SNP_Gene = SNP_high_impact_count/SNP_count)

allload_gather <- gather(allload, key = "Group", value = "Prop", 44:51)

allload_gather <- allload_gather %>% mutate(TYPE = str_extract(Group, "^[^_]+"), GROUP = str_extract(Group, "(?<=_).*$"))

order <- c('None','UpDownstream','Intron','Gene')
allload_gather$GROUP<- factor(allload_gather$GROUP, levels = order)
```

```
#alternate allele count proportions
pdf("Nc3_SV_SNP_impacts.pdf", width=6, height=4)
ggplot(allload_gather , aes(x = GROUP, y =  Prop , fill=TYPE, color=TYPE)) +
  geom_boxplot() +
  labs(x = "Frequency", y = "Minor allele count proportions") +
theme_classic(base_size = 18) + scale_fill_manual(values=c("#d1ac6b","#10a4c2")) + scale_color_manual(values=c("#d1ac6b","#10a4c2"))
dev.off()

summary(aov(Prop~Group*TYPE, data=allload_gather ))
TukeyHSD(aov(Prop~Group*TYPE, data=allload_gather ))
```



Katarina Stuart (z5188231@ad.unsw.edu.au) - Apr 23, 2024, 10:34 AM GMT+12

Per-individual SV counts: load

```
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/counts")

#SV
data <- fread("merged_rep_missfiltered_reffix2_load.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")

#two method
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 = "ID", by.y = "V1", all.x =  TRUE)
fun$IMPACT[is.na(fun$IMPACT)] <- 'NONE'

funSV <- fun %>%
  mutate(SVmask_count= rowSums(select(., `68158`:`98617`) == 1, na.rm = TRUE) /
                   rowSums(select(., `68158`:`98617`)  > -1, na.rm = TRUE)) %>%
  mutate(SVreal_count= rowSums(select(., `68158`:`98617`) == 2, na.rm = TRUE) /
                   rowSums(select(., `68158`:`98617`)  > -1, na.rm = TRUE)) %>% select(IMPACT, SVmask_count, SVreal_count)

funSV <- funSV %>% mutate(type = "SV")

#SNP
data <- fread("hihi_wgs_filter_highcov_no83318_autosomes_load.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")

#two method
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 = "ID", by.y = "V1", all.x =  TRUE)
fun$IMPACT[is.na(fun$IMPACT)] <- 'NONE'

funSNP <- fun %>%
  mutate(SNPmask_count= rowSums(select(., `68158`:`98617`) == 1, na.rm = TRUE) /
                   rowSums(select(., `68158`:`98617`)  > -1, na.rm = TRUE)) %>%
  mutate(SNPreal_count= rowSums(select(., `68158`:`98617`) == 2, na.rm = TRUE) /
                   rowSums(select(., `68158`:`98617`)  > -1, na.rm = TRUE)) %>% select(IMPACT, SNPmask_count, SNPreal_count)

funSNP <- funSNP %>% mutate(type = "SNP")

combine <- rbind(funSNP , funSV, use.names=FALSE)

combine_gather <- gather(combine , key = "Variant", value = "Freq", 2:3)
```

```
combine_gather <- combine_gather %>% mutate(GROUP = paste0(type,"_",Variant))

combine_gather %>% group_by(type, Variant, IMPACT) %>% summarise(average = mean(Freq),count = n(), std_dev = sd(Freq, na.rm =
TRUE))

summary(aov(Freq~IMPACT*type, data=combine_gather  ))
TukeyHSD(aov(Freq~IMPACT*type, data=combine_gather  ))

# Calculate means and standard errors
summary_data <- combine_gather %>%
  group_by(type, Variant, IMPACT) %>%
  summarise(mean_value = mean(Freq),
            std_error = sd(Freq) / sqrt(n()))  # Calculate standard error instead of standard deviation

# Plot means and standard errors
pdf("Nc3_SV_SNP_loads.pdf", width=6, height=4)
ggplot(summary_data, aes(x = type, y = mean_value, fill = 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
  labs(x = "Type", y = "Frequency", title = "Means and Standard Errors") +
  theme_classic(base_size = 18) +
  scale_fill_manual(values = c("#d1ac6b", "#10a4c2")) +
  scale_color_manual(values = c("#d1ac6b", "#10a4c2")) +
  facet_grid(. ~ Variant)
dev.off()

#backup box plot
ggplot(summary_data, aes(x = type, y = mean_value, fill = IMPACT)) +
  geom_point(position = position_dodge(width = 0.75), size = 3, shape = 21) +
  geom_errorbar(aes(ymin = mean_value - std_dev, ymax = mean_value + std_dev),
                position = position_dodge(width = 0.75), width = 0.2) +
  labs(x = "Type", y = "Frequency", title = "Means and Standard Deviations") +
  theme_classic(base_size = 18) +
  scale_fill_manual(values = c("#d1ac6b", "#10a4c2")) +
  scale_color_manual(values = c("#d1ac6b", "#10a4c2")) +
facet_grid(. ~ Variant)
```