Starling-May18
Projects/Katarina
Stuart/KStuart.Starling-Aug18/Sv5_AustraliaWGS/Analysis/2021-09-24.BalancingSelection

PDF Version generated by

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

on

Feb 20, 2023 @10:28 AM NZDT

Table of Contents

2021-09-24.BalancingSelection

Katarina Stuart (z5188231@ad.unsw.edu.au) - Feb 20, 2023, 10:13 AM GMT+13

Balancing Selection

Evolutionary origins of genomic adaptations in an invasive copepod | Nature Ecology & Evolution

thinning data?: Genomic signatures in the coral holobiont reveal host adaptations driven by Holocene climate change and reef specific symbionts (science.org)

"To ensure that only independent loci were provided to Bayescan 2, we first thinned data to ensure a physical distance of at least 10 kb. This resulted in a total of 27,109 sites available for analysis across all populations. "

Do balancing selection calcualtion on non-linkage filtered SNPs for UK individuals only. The Bayescan analysis can work on the thinned data set (independent SNPs).

```
cd /srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection/
```

```
module load vcftools/0.1.16
```

```
module load bayescan/2.1
```

```
module load R/3.5.3
module load samtools/1.10
```

Creating data subsets

```
cd /srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/

#For directional selection analysis: use linkage filtered and further thin the data
VCF=/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/data/snp_variants_processed/wgs_variantsgenotyped_filtered_miss50_r2_noIndel_WithIds.vcf

#Create VCF subset with only AU and UK individuals. SNPS: 6635273
vcftools --vcf ${VCF} --keep au_uk_inds.txt --recode --recode-INFO-all --out wgs_variantsgenotyped_filtered_miss50_r2_au_uk

#create data set with UK and AUeast/AUsouth, thinned, for selection analysis (convert using PGD spider). Total SNPS for all: 182829
vcftools --vcf wgs_variantsgenotyped_filtered_miss50_r2_au_uk.recode.vcf --thin 5000 --recode --recode-INFO-all --
out wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin
vcftools --vcf wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin.recode.vcf --keep inds_uk_AUeast.txt --recode --recode-INFO-all --
out wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUeast
vcftools --vcf wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin.recode.vcf --keep inds_uk_AUsouth.txt --recode --recode-INFO-all --
out wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUsouth

#For balancing selection analysis: use whole genome data
#Dataset with only UK SNPs (for balancing selection analysis). Map outliers onto output of this file.
VCF=/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/data/snp_variants_processed/wgs_variantsgenotyped_filtered_miss50.recode.vcf
vcftools --vcf ${VCF}  --keep vcf_pop_uk.txt --recode --recode-INFO-all --out wgs_variantsgenotyped_filtered_miss50_UK
bcftools view --types snps wgs_variantsgenotyped_filtered_miss50_UK.recode.vcf | bcftools annotate --set-id +'%CHROM\_%POS'
> wgs_variantsgenotyped_filtered_miss50_UK_noIndel_WithIds.vcf
vcftools --vcf wgs_variantsgenotyped_filtered_miss50_UK_noIndel_WithIds.vcf --mac 3 --max-mac 14 --recode --recode-INFO-all --
out wgs_variantsgenotyped_filtered_miss50_UK_noIndel_WithIds_maf015
```

For final bal sel file: After filtering, kept 7348570 out of a possible 17191328 Sites

BAYESCAN (SNP)

Run PGDSpider to convert file for bayescan:

to PGD format:

```
java -Xmx120g -Xms512m -jar /srv/scratch/z5188231/KStuart.Starling-Aug18/programs/PGDSpider_2.1.1.5/PGDSpider2-cli.jar -inputfile
wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUeast.recode.vcf -inputformat VCF -outputfile
wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUeast.txt -outputformat PGD -spid vcf_VCF_PGD_AUeast.spid
```

```
java -Xmx120g -Xms512m -jar /srv/scratch/z5188231/KStuart.Starling-Aug18/programs/PGDSpider_2.1.1.5/PGDSpider2-cli.jar -inputfile
wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUsouth.recode.vcf -inputformat VCF -outputfile
wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUsouth.txt -outputformat PGD -spid vcf_VCF_PGD_AUsouth.spid
```

then convert to bayescan

```
java -Xmx120g -Xms512m -jar /srv/scratch/z5188231/KStuart.Starling-Aug18/programs/PGDSpider_2.1.1.5/PGDSpider2-cli.jar -inputfile
wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUeast.txt -inputformat PGD -outputfile wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUeast.bs -
outputformat GESTE_BAYE_SCAN

java -Xmx120g -Xms512m -jar /srv/scratch/z5188231/KStuart.Starling-Aug18/programs/PGDSpider_2.1.1.5/PGDSpider2-cli.jar -inputfile
wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUsouth.txt -inputformat PGD -outputfile wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUsouth.bs -
outputformat GESTE_BAYE_SCAN
```

BAYESCAN RUNS

```
#!/bin/bash
#PBS -N 2021-11-21.snp_bayescan.pbs
#PBS -V
#PBS -l nodes=1:ppn=16
#PBS -l mem=124gb
#PBS -l walltime=48:00:00
#PBS -j oe
#PBS -M katarina.stuart@unsw.edu.au
#PBS -m ae

module load bayescan/2.1

cd /srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022

bayescan_2.1 ./wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUeast.bs -od ./snp_bayescan -threads 16 -n 5000 -thin 10 -nbp 20 -pilot 5000 -burn 50000 -
pr_odds 10
bayescan_2.1 ./wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUsouth.bs -od ./snp_bayescan -threads 16 -n 5000 -thin 10 -nbp 20 -pilot 5000 -burn 50000 -
pr_odds 10
```

Identify outliers:

```
module load R/3.5.3
R
library(ggplot2)
setwd("/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/snp_bayescan")
```

```
source("/srv/scratch/z5188231/KStuart.Starling-Aug18/programs/BayeScan2.1/Rfunctions/plot_R.r")
outliers.AUeast=plot_bayescan("/srv/scratch/z5188231/KStuart.Starling-
Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/snp_bayescan/wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUeas_fst.txt",FDR=0.05)
outliers.AUsouth=plot_bayescan("/srv/scratch/z5188231/KStuart.Starling-
Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/snp_bayescan/wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUsout_fst.txt",FDR=0.05)
outliers.AUeast
outliers.AUsouth
all.outliers<-union(outliers.AUeast$outliers,outliers.AUsouth$outliers)
write.table(all.outliers, file="bayscan_outliers_vcf.txt")
```

```
write.table(outliers.AUeast, file="bayscan_outliers_AUeast_vcf.txt")
write.table(outliers.AUsouth, file="bayscan_outliers_AUsouth_vcf.txt")
```

> outliers.AUeast
$outliers
[1] 127803 150485 160604

$nb_outliers
[1] 3

> outliers.AUsouth
$outliers
 [1] 9349 13890 38578 40117 41871 47087 65212 79416 82051 82797
[11] 83446 88290 88400 99078 103515 108123 115740 117962 125331 129718
[21] 133583 148322 148725 151176 152841 153489 161970 166883 176494 177236
[31] 182116 183676 183876

$nb_outliers
[1] 33

Mapping Outliers

```
cd /srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/

#create list of SNPs in VCF, assign line numbers that can be used to find matching line numbers in outliers (SNP ID is lost in bayescan, line numbers used as signifiers).
grep -v "^#" wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin.recode.vcf  | cut -f1-3 | awk '{print $0"\t"NR}'
> wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_SNPs.txt
cd snp_bayescan
awk '{print $2}' bayscan_outliers_AUeast_vcf.txt > bayscan_outliers_AUeast_vcf_numbers.txt
awk '{print $2}' bayscan_outliers_AUsouth_vcf.txt > bayscan_outliers_AUsouth_vcf_numbers.txt
cut -d ' ' -f2 bayscan_outliers_vcf.txt > bayscan_outliers_AUboth_vcf_numbers.txt

#list of outlier SNPS
awk 'FNR==NR{a[$1];next} (($4) in a)' bayscan_outliers_AUboth_vcf_numbers.txt ../wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_SNPs.txt   | cut -f3
> bayscan_outliers_vcf_SNPs.txt

awk 'FNR==NR{a[$1];next} (($4) in a)' bayscan_outliers_AUboth_vcf_numbers.txt ../wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_SNPs.txt   | cut -f3
> bayscan_outliers_vcf_SNPs.txt
awk 'FNR==NR{a[$1];next} (($4) in a)' bayscan_outliers_AUboth_vcf_numbers.txt ../wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_SNPs.txt   | cut -f3
> bayscan_outliers_vcf_SNPs.txt
```

Bayescane Log Plot

```
module load R/3.5.3
R
```

```
library(RColorBrewer)
library(SNPRelate)
library(gdsfmt)
library(scales)
library(adegenet)
library(pegas)
library(ggplot2)
library(ape)
library(poppr)
library(rgl)
library(dplyr)
```

```
setwd("/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/snp_bayescan")

bayescan.out.east<- read.table("wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUeas_fst.txt", header=TRUE)
bayescan.out.east <- bayescan.out.east %>% mutate(ID = row_number())
bayescan.outiers.east<- read.table("bayscan_outliers_AUeast_vcf_numbers.txt", header=FALSE)
outliers.plot.east <- filter(bayescan.out.east, ID %in% bayescan.outiers.east[["V1"]])
```

```
bayescan.out.south<- read.table("wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUsout_fst.txt", header=TRUE)
bayescan.out.south <- bayescan.out.south %>% mutate(ID = row_number())
bayescan.outiers.south<- read.table("bayscan_outliers_AUsouth_vcf_numbers.txt", header=FALSE)
outliers.plot.south <- filter(bayescan.out.south, ID %in% bayescan.outiers.south[["V1"]])

png("Sv5_outlierSNP_AUeast_bayescane_fst.png", width=600, height=350)
ggplot(bayescan.out.east, aes(x=log10.PO., y=alpha))+
geom_point(size=5,alpha=1)+xlim(-1.3,3.5)+ theme_classic(base_size = 18) + geom_vline(xintercept = 0, linetype="dashed", color = "black", size=3)+
geom_point(aes(x=log10.PO., y=alpha), data=outliers.plot.east, col="red", fill="red",size=5,alpha=1) + theme(axis.text=element_text(size=18),
axis.title=element_text(size=22,face="bold"))
dev.off()

png("Sv5_outlierSNP_AUsouth_bayescane_fst.png", width=600, height=350)
ggplot(bayescan.out.south, aes(x=log10.PO., y=alpha))+
geom_point(size=5,alpha=1)+xlim(-1.3,3.5)+ theme_classic(base_size = 18) + geom_vline(xintercept = 0, linetype="dashed", color = "black", size=3)+
geom_point(aes(x=log10.PO., y=alpha), data=outliers.plot.south, col="red", fill="red",size=5,alpha=1) + theme(axis.text=element_text(size=18),
axis.title=element_text(size=22,face="bold"))
dev.off()
```

Baypass (for SNPs)

```
module add vcftools/0.1.16
module load plink/1.90b6.7
cd /srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/snp_baypass
```

```
#SNP AUeast & AUsouth
for POP in AUeast AUsouth
do
#make plink file type
vcftools --vcf ../wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_${POP}.recode.vcf --plink --out wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_${POP}.plink
#remove popind labels from .PED and replace with new ones that have pop groupings
cut -f 3- wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_${POP}.plink.ped >x.delete
paste pop_uk_${POP}_plink.txt x.delete > wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_${POP}.plink.ped2
rm x.delete
mv
wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_${POP}.plink.ped2 wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_${POP}.plink.ped
#run the pop based allele frequency calculations
plink --file wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_${POP}.plink --allow-extra-chr --freq counts --family --
out wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_${POP}
#manipulate file so it has baypass format, numbers set for plink output file and pop number for column count
tail -n +2 wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_${POP}.frq.strat | awk '{ $9 = $8 - $7 } 1' | awk '{print $7,$9}' | tr "\n" " " | sed 's/ /\n/4;
P; D'> wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_${POP}_baypass.txt
done
```

Baypass

Baypass runs:

```
#!/bin/bash
#PBS -N 2022-12-05.baypass_SNP_AUsouth.pbs
#PBS -l nodes=1:ppn=16
#PBS -l mem=120gb
#PBS -l walltime=12:00:00
#PBS -j oe
#PBS -M katarina.stuart@unsw.edu.au
#PBS -m ae

module load baypass

cd /srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/snp_baypass
```

```
g_baypass -npop 2 -gfile ./wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUeast_baypass.txt -outprefix
wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUeast_baypass -nthreads 16
g_baypass -npop 2 -gfile G.btapods.SNP.AUeast  -outprefix G.btapods.SNP.AUeast -nthreads 16

g_baypass -npop 2 -gfile ./wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUsouth_baypass.txt -outprefix
wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUsouth_baypass -nthreads 16
g_baypass -npop 2 -gfile G.btapods_AUsouth  -outprefix G.btapods_AUsouth -nthreads 16
```

Running in R to make the anapod data:

```
module load R/3.6.3
R
```

```
#SNP
setwd("/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/snp_baypass")
source("/apps/baypass/2.1/utils/baypass_utils.R")
library("ape")
library("corrplot")
```

```
#AUeast
omega=as.matrix(read.table("wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUeast_baypass_mat_omega.out"))
anacore.snp.res=read.table("wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUeast_baypass_summary_pi_xtx.out",h=T)
pi.beta.coef=read.table("wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUeast_baypass_summary_beta_params.out",h=T)$Mean
bta14.data<-geno2YN("wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUeast_baypass.txt")
simu.bta<-simulate.baypass(omega.mat=omega, nsnp=5000, sample.size=bta14.data$NN,
beta.pi=pi.beta.coef,pi.maf=0,suffix="btapods_AUeast")
```

```
#AUsouth
omega=as.matrix(read.table("wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUsouth_baypass_mat_omega.out"))
anacore.snp.res=read.table("wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUsouth_baypass_summary_pi_xtx.out",h=T)
pi.beta.coef=read.table("wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUsouth_baypass_summary_beta_params.out",h=T)$Mean
bta14.data<-geno2YN("wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUsouth_baypass.txt")
simu.bta<-simulate.baypass(omega.mat=omega, nsnp=5000, sample.size=bta14.data$NN,
beta.pi=pi.beta.coef,pi.maf=0,suffix="btapods_AUsouth")
```

XtX calibration; get the pod XtX

```
pod.xtx=read.table("G.btapods_AUeast_summary_pi_xtx.out",h=T)$M_XtX
```

```
pod.xtx=read.table("G.btapods_AUsouth_summary_pi_xtx.out",h=T)$M_XtX
```

Compute the 1% threshold

```
pod.thresh=quantile(pod.xtx,probs=0.99) #AUeast: 4.227718
pod.thresh=quantile(pod.xtx,probs=0.99) #AUsouth: 4.310763
```

Filter the data for the outlier snps:

```
module load bedtools/2.27.1
```

```
#Find outliers that are above theanapod threshold
cat wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUeast_baypass_summary_pi_xtx.out | awk '$6>4.227718 ' > outliers_SNP_AUeast.txt
cat wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_AUsouth_baypass_summary_pi_xtx.out | awk '$6>4.310763' > outliers_SNP_AUsouth.txt
#create list of SNPs in VCF, assign line numbers that can be used to find matching line numbers in outliers (SNP ID is lost in bayescan, line numbers used as signifiers).
grep -v "^#" ../wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin.recode.vcf | cut -f1-3 | awk '{print $0"\t"NR}'  > wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_SNPlist.txt

#list of outlier SNPS
```

```
awk 'FNR==NR{a[$1];next} ($4) in a)' outliers_SNP_AUeast.txt ./wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_SNPlist.txt | cut -f3
> outliers_SNP_AUeast_SNPlist.txt
awk 'FNR==NR{a[$1];next} ($4) in a)' outliers_SNP_AUsouth.txt ./wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_SNPlist.txt | cut -f3
> outliers_SNP_AUsouth_SNPlist.txt
comm <(sort outliers_SNP_AUeast_SNPlist.txt) <(sort outliers_SNP_AUsouth_SNPlist.txt) > outliers_SNP_AUboth_SNPlist.txt
```

Combine outliers

```
module load plink/1.90b6.7
module load vcftools/0.1.16

cd ../snp_outliers

cat ../snp_bayescan/bayscan_outliers_vcf_SNPs.txt ../snp_baypass/outliers_SNP_AUboth_SNPlist.txt > snp_outliers_bayescan_baypass.txt

#Create list on non outlier SNPs
grep -v "^#" ../wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_SNPs.txt   | cut -f3 > all_SNPs.txt
comm -23 <(sort all_SNPs.txt) <(sort snp_outliers_bayescan_pcadapt.txt)  > snp_nonoutliers_vcf_SNPs.txt
```

PCA of outliers

```
module load plink/1.90b6.7
module load vcftools/0.1.16

#VCF of just the outliers in AU and UK individuals for PCA
vcftools --vcf ../wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin.recode.vcf --snps snp_outliers_bayescan_baypass.txt --recode --recode-INFO-all --
out  wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_outlier

vcftools --vcf wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_outlier.recode.vcf --plink --out wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_outlier.plink
plink --file wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_outlier.plink --pca --out wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_outlier --make-rel
```

PCA Plot

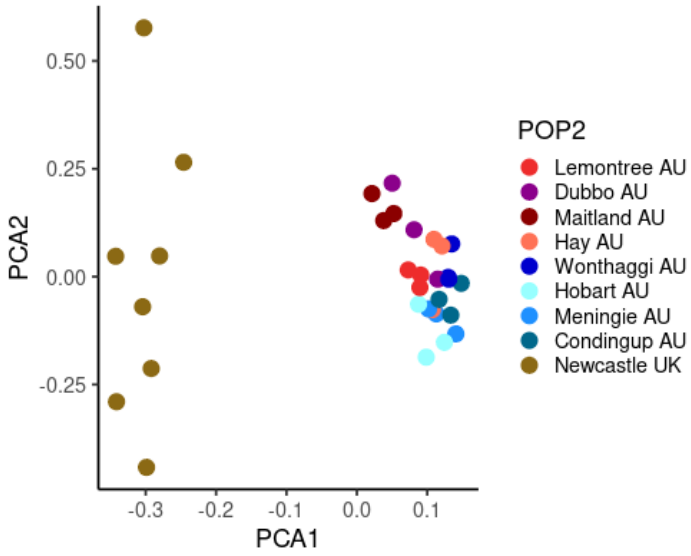
```
module load R/3.5.3
R
```

```
library(RColorBrewer)
library(SNPRelate)
library(gdsfmt)
library(scales)
library(adegenet)
library(pegas)
library(ggplot2)
library(ape)
library(poppr)
library(rgl)
library(dplyr)
```

```
setwd("/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/snp_outliers")
```

```
#Au samples
pca.eigenvec <- read.table("wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_outlier.eigenvec", sep=" ")

head(round(pca.eigenvec, 2))
pca_g1 <- data.frame(sample.id = pca.eigenvec$V1,
                     EV1 = pca.eigenvec$V3,    # the first eigenvector
                     EV2 = pca.eigenvec$V4,    # the second eigenvector
                     EV3 = pca.eigenvec$V5,    # the second eigenvector
                     stringsAsFactors = FALSE)
head(pca_g1)
pca_g1

# add labels by population in correct order at vcf file

population <- read.table("/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/snp_popgen/PCA/populations.txt", header = TRUE, sep = "\t")
population2 <- population %>% filter(POP != "New York", POP != "Munglinup")
```

```
pca_g1 <- cbind(pca_g1,population2)
pca_g1


pca_g1$POP2 <- factor(pca_g1$POP, levels = c("Lemontree", "Dubbo", "Maitland","Hay","Wonthaggi","Hobart","Meningie","Condingup","Munglinup","Newcastle"))

levels(pca_g1$POP2) <- c("Lemontree AU", "Dubbo AU", "Maitland AU","Hay
AU","Wonthaggi AU","Hobart AU","Meningie AU","Condingup AU","Munglinup  AU","Newcastle UK")
pca_g1$POP2


png("Sv5_SV_PCA12.png", width=500, height=400)
ggplot(pca_g1,aes(x=EV1,y=EV2,col=POP2))+
geom_point(size=5,alpha=1)+
scale_color_manual(values = c("firebrick2","darkmagenta","darkred","coral1","blue3","darkslategray1","dodgerblue","deepskyblue4","goldenrod4"))+
theme_classic(base_size = 18) + xlab("PCA1") + ylab("PCA2")
dev.off()
```



FST genome plot of outliers

```
module load plink/1.90b6.7
module load vcftools/0.1.16

cd /srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/snp_outliers

#VCF of just the outliers in AU and UK individuals for PCA
vcftools --vcf ../wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin.recode.vcf --weir-fst-pop ../inds_AUeast.txt --weir-fst-pop ../inds_uk.txt --out AUeast_uk
vcftools --vcf ../wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin.recode.vcf --weir-fst-pop ../inds_AUsouth.txt --weir-fst-pop ../inds_uk.txt --out AUsouth_uk

cat ../snp_bayescan/bayscan_outliers_vcf_SNPs.txt ../snp_baypass/outliers_SNP_AUboth_SNPlist.txt > snp_outliers_bayescan_baypass.txt


vcftools --vcf wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_outlier.recode.vcf --plink --out wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_outlier.plink
plink --file wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_outlier.plink --pca --out wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_outlier --make-rel
```

PCA Plot

```
module load R/3.5.3
R
```

```
library(RColorBrewer)
library(SNPRelate)
library(gdsfmt)
library(scales)
library(adegenet)
```

```
library(pegas)
library(ggplot2)
library(ape)
library(poppr)
library(rgl)
library(dplyr)
```

```
setwd("/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/snp_outliers")
```

```
#Au samples
pca.eigenvec <- read.table("wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_outlier.eigenvec", sep=" ")

head(round(pca.eigenvec, 2))
pca_g1 <- data.frame(sample.id = pca.eigenvec$V1,
                     EV1 = pca.eigenvec$V3,    # the first eigenvector
                     EV2 = pca.eigenvec$V4,    # the second eigenvector
                     EV3 = pca.eigenvec$V5,    # the second eigenvector
                     stringsAsFactors = FALSE)
head(pca_g1)
pca_g1

# add labels by population in correct order at vcf file

population <- read.table("/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/snp_popgen/PCA/populations.txt", header = TRUE, sep = "\t")
population2 <- population %>% filter(POP != "New York", POP != "Munglinup")
pca_g1 <- cbind(pca_g1,population2)
pca_g1


pca_g1$POP2 <- factor(pca_g1$POP, levels = c("Lemontree", "Dubbo", "Maitland","Hay","Wonthaggi","Hobart","Meningie","Condingup","Munglinup","Newcastle"))

levels(pca_g1$POP2) <- c("Lemontree AU", "Dubbo AU", "Maitland AU","Hay
AU","Wonthaggi AU","Hobart AU","Meningie AU","Condingup AU","Munglinup  AU","Newcastle UK")
pca_g1$POP2


png("Sv5_SV_PCA12.png", width=500, height=400)
ggplot(pca_g1,aes(x=EV1,y=EV2,col=POP2))+
geom_point(size=5,alpha=1)+
scale_color_manual(values = c("firebrick2","darkmagenta","darkred","coral1","blue3","darkslategray1","dodgerblue","deepskyblue4","goldenrod4"))+
theme_classic(base_size = 18) + xlab("PCA1") + ylab("PCA2")
dev.off()
```

SV Outliers:

```
cd /srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/
```

```
SVCFd=/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/data/survivor_2022/split_vcfs/merged_rep_filtered_dummy_edit.recode.vcf
```

```
module load vcftools/0.1.16
```

```
module load bayescan/2.1
```

```
module load R/3.5.3
```

Creating data subsets

```
#Create VCF subset with only AU and UK individuals.
vcftools --vcf ${SVCFd} --keep au_uk_inds.txt --recode --recode-INFO-all --out merged_rep_filtered_dummy_edit_au_uk

#create data set with UK and AUeast/AUsouth, for selection analysis (convert using PGD spider).
vcftools --vcf merged_rep_filtered_dummy_edit_au_uk.recode.vcf --keep inds_uk_AUeast.txt --recode --recode-INFO-all --out
merged_rep_filtered_dummy_edit_au_uk_AUeast
vcftools --vcf merged_rep_filtered_dummy_edit_au_uk.recode.vcf --keep inds_uk_AUsouth.txt --recode --recode-INFO-all --out
merged_rep_filtered_dummy_edit_au_uk_AUsouth
```

BAYESCAN (SV)

Run PGDSpider to convert file for bayescan:

to PGD format:

```
java -Xmx120g -Xms512m -jar /srv/scratch/z5188231/KStuart.Starling-Aug18/programs/PGDSpider_2.1.1.5/PGDSpider2-cli.jar -inputfile
merged_rep_filtered_dummy_edit_au_uk_AUeast.recode.vcf -inputformat VCF -outputfile merged_rep_filtered_dummy_edit_au_uk_AUeast.txt -outputformat PGD -spid
vcf_VCF_PGD_AUeast.spid
```

```
java -Xmx120g -Xms512m -jar /srv/scratch/z5188231/KStuart.Starling-Aug18/programs/PGDSpider_2.1.1.5/PGDSpider2-cli.jar -inputfile
merged_rep_filtered_dummy_edit_au_uk_AUsouth.recode.vcf -inputformat VCF -outputfile merged_rep_filtered_dummy_edit_au_uk_AUsouth.txt -outputformat PGD -spid
vcf_VCF_PGD_AUsouth.spid
```

then convert to bayescan

```
java -Xmx120g -Xms512m -jar /srv/scratch/z5188231/KStuart.Starling-Aug18/programs/PGDSpider_2.1.1.5/PGDSpider2-cli.jar -inputfile
merged_rep_filtered_dummy_edit_au_uk_AUeast.txt -inputformat PGD -outputfile merged_rep_filtered_dummy_edit_au_uk_AUeast.bs -outputformat
GESTE_BAYE_SCAN

java -Xmx120g -Xms512m -jar /srv/scratch/z5188231/KStuart.Starling-Aug18/programs/PGDSpider_2.1.1.5/PGDSpider2-cli.jar -inputfile
merged_rep_filtered_dummy_edit_au_uk_AUsouth.txt -inputformat PGD -outputfile merged_rep_filtered_dummy_edit_au_uk_AUsouth.bs -outputformat
GESTE_BAYE_SCAN
```

BAYESCAN RUNS

```
#!/bin/bash
#PBS -N  2021-11-21.sv_bayescan.pbs
#PBS -V
#PBS -l nodes=1:ppn=16
#PBS -l mem=124gb
#PBS -l walltime=12:00:00
#PBS -j oe
#PBS -M katarina.stuart@unsw.edu.au
#PBS -m ae

module load bayescan/2.1

cd /srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022

bayescan_2.1 ./merged_rep_filtered_dummy_edit_au_uk_AUeast.bs -od ./sv_bayescan -threads 16 -n 5000 -thin 10 -nbp 20 -pilot 5000 -burn 50000 -pr_odds 5
bayescan_2.1 ./merged_rep_filtered_dummy_edit_au_uk_AUsouth.bs -od ./sv_bayescan -threads 16 -n 5000 -thin 10 -nbp 20 -pilot 5000 -burn 50000 -pr_odds 5
```

Identify outliers:

```
module load R/3.5.3
R
library(ggplot2)
setwd("/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/sv_bayescan")
```

```
source("/srv/scratch/z5188231/KStuart.Starling-Aug18/programs/BayeScan2.1/Rfunctions/plot_R.r")
outliers.AUeast=plot_bayescan("/srv/scratch/z5188231/KStuart.Starling-
Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/sv_bayescan/merged_rep_filtered_dummy_edit_au_uk_AUeas_fst.txt",FDR=0.05 )
outliers.AUsouth=plot_bayescan("/srv/scratch/z5188231/KStuart.Starling-
Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/sv_bayescan/merged_rep_filtered_dummy_edit_au_uk_AUsout_fst.txt",FDR=0.05 )
```

```
outliers.AUeast
outliers.AUsouth
all.outliers<-union(outliers.AUsouth$outliers,outliers.AUeast$outliers)
write.table(all.outliers, file="bayscan_outliers_svcf.txt")
write.table(outliers.AUeast, file="bayscan_outliers_AUeast_svcf.txt")
write.table(outliers.AUsouth, file="bayscan_outliers_AUsouth_svcf.txt")
```

> outliers.AUeast
$outliers
integer(0)

$nb_outliers
[1] 0

> outliers.AUsouth
$outliers
[1] 3692

$nb_outliers
[1] 1

Mapping Outliers

```
cd /srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/

#create list of SNPs in VCF, assign line numbers that can be used to find matching line numbers in outliers (SNP ID is lost in bayescan, line numbers used as signifiers).
grep -v "^#" merged_rep_filtered_dummy_edit_au_uk.recode.vcf  | cut -f1-3 | awk '{print $0"\t"NR}' > merged_norep_mq_filtered_dummy_edit_au_uk.txt
cd sv_bayescan
awk '{print $2}' bayscan_outliers_svcf.txt > bayscan_outliers_svcf_numbers.txt
awk '{print $2}' bayscan_outliers_AUeast_svcf.txt > bayscan_outliers_AUeast_svcf_numbers.txt #use these just for plotting
awk '{print $2}' bayscan_outliers_AUsouth_svcf.txt > bayscan_outliers_AUsouth_svcf_numbers.txt #use these just for plotting

#list of outlier SNPS
awk 'FNR==NR{a[$1];next} ($4) in a)' bayscan_outliers_svcf_numbers.txt ../merged_norep_mq_filtered_dummy_edit_au_uk.txt | cut -f3 > bayscan_outliers_svcf_SNPs.txt
```

Bayescane Log Plot

```
module load R/3.5.3
R
```

```
library(RColorBrewer)
library(SNPRelate)
library(gdsfmt)
library(scales)
library(adegenet)
library(pegas)
library(ggplot2)
library(ape)
library(poppr)
library(rgl)
library(dplyr)
```

```
setwd("/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/sv_bayescan")

bayescan.out.east<- read.table("merged_rep_filtered_dummy_edit_au_uk_AUeas_fst.txt", header=TRUE)
bayescan.out.east <- bayescan.out.east %>% mutate(ID = row_number())
bayescan.outiers.east<- read.table("bayscan_outliers_AUeast_svcf_numbers.txt", header=FALSE)
outliers.plot.east <- filter(bayescan.out.east, ID %in% bayescan.outiers.east[["V1"]])

bayescan.out.south<- read.table("merged_rep_filtered_dummy_edit_au_uk_AUsout_fst.txt", header=TRUE)
bayescan.out.south <- bayescan.out.south %>% mutate(ID = row_number())
bayescan.outiers.south<- read.table("bayscan_outliers_AUsouth_svcf_numbers.txt", header=FALSE)
outliers.plot.south <- filter(bayescan.out.south, ID %in% bayescan.outiers.south[["V1"]])

png("Sv5_outlierSV_AUeast_bayescane_fst.png", width=600, height=350)
ggplot(bayescan.out.east, aes(x=log10.PO., y=alpha))+
geom_point(size=5,alpha=1)+xlim(-1.3,3.5)+ theme_classic(base_size = 18) + geom_vline(xintercept = 0, linetype="dashed", color = "black", size=3)+
```

```
theme(axis.text=element_text(size=18), axis.title=element_text(size=22,face="bold")) #+ geom_point(aes(x=log10.PO., y=alpha), data=outliers.plot.east, col="red", fill="red",size=5,alpha=1)
dev.off()

png("Sv5_outlierSV_AUsouth_bayescane_fst.png", width=600, height=350)
ggplot(bayescan.out.south, aes(x=log10.PO., y=alpha))+
geom_point(size=5,alpha=1)+xlim(-1.3,3.5)+ theme_classic(base_size = 18) + geom_vline(xintercept = 0, linetype="dashed", color = "black", size=3)+
geom_point(aes(x=log10.PO., y=alpha), data=outliers.plot.south, col="red", fill="red",size=5,alpha=1) + theme(axis.text=element_text(size=18), axis.title=element_text(size=22,face="bold"))
dev.off()
```

Baypass (for SVs)

```
module add vcftools/0.1.16
module load plink/1.90b6.7
cd /srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/sv_baypass

#SV AUsouth & AUeast
for POP in AUeast AUsouth
do
#make plink file type
vcftools --vcf ../merged_rep_filtered_dummy_edit_au_uk_${POP}.recode.vcf --plink --out merged_rep_filtered_dummy_edit_au_uk_${POP}.plink
#remove popind labels from .PED and replace with new ones that have pop groupings
cut -f 3- merged_rep_filtered_dummy_edit_au_uk_${POP}.plink.ped >x.delete
paste pop_uk_${POP}_plink.txt x.delete > merged_rep_filtered_dummy_edit_au_uk_${POP}.plink.ped2
rm x.delete
mv merged_rep_filtered_dummy_edit_au_uk_${POP}.plink.ped2 merged_rep_filtered_dummy_edit_au_uk_${POP}.plink.ped
#run the pop based allele frequency calculations
plink --file merged_rep_filtered_dummy_edit_au_uk_${POP}.plink --allow-extra-chr --freq counts --family --
out merged_rep_filtered_dummy_edit_au_uk_${POP}
#manipulate file so it has baypass format, numbers set for plink output file and pop number for column count
tail -n +2 merged_rep_filtered_dummy_edit_au_uk_${POP}.frq.strat | awk '{ $9 = $8 - $7 } 1' | awk '{print $7,$9}' | tr "\n" " " | sed 's/ /\n/4; P; D'>
merged_rep_filtered_dummy_edit_au_uk_${POP}_baypass.txt
done
```

Baypass

Baypass runs:

```
#!/bin/bash
#PBS -N 2022-12-05.baypass_SV_AUsouth.pbs
#PBS -l nodes=1:ppn=16
#PBS -l mem=120gb
#PBS -l walltime=12:00:00
#PBS -j oe
#PBS -M katarina.stuart@unsw.edu.au
#PBS -m ae

module load baypass

cd /srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/sv_baypass

g_baypass -npop 2 -gfile ./merged_rep_filtered_dummy_edit_au_uk_AUeast_baypass.txt -outprefix merged_rep_filtered_dummy_edit_au_uk_AUeast_baypass -nthreads 16
g_baypass -npop 2 -gfile G.btapods_AUeast  -outprefix G.btapods_AUeast -nthreads 16

g_baypass -npop 2 -gfile ./merged_rep_filtered_dummy_edit_au_uk_AUsouth_baypass.txt -outprefix merged_rep_filtered_dummy_edit_au_uk_AUsouth_baypass -nthreads 16
g_baypass -npop 2 -gfile G.btapods_AUsouth  -outprefix G.btapods_AUsouth -nthreads 16
```

Running in R to make the anapod data:

```
module load R/3.6.3
R
```

```
#SV
setwd("/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/sv_baypass")
source("/apps/baypass/2.1/utils/baypass_utils.R")
library("ape")
library("corrplot")
```

```
#AUeast
omega=as.matrix(read.table("merged_rep_filtered_dummy_edit_au_uk_AUeast_baypass_mat_omega.out"))
anacore.snp.res=read.table("merged_rep_filtered_dummy_edit_au_uk_AUeast_baypass_summary_pi_xtx.out",h=T)
pi.beta.coef=read.table("merged_rep_filtered_dummy_edit_au_uk_AUeast_baypass_summary_beta_params.out",h=T)$Mean
bta14.data<-geno2YN("merged_rep_filtered_dummy_edit_au_uk_AUeast_baypass.txt")
simu.bta<-simulate.baypass(omega.mat=omega, nsnp=5000, sample.size=bta14.data$NN,
beta.pi=pi.beta.coef,pi.maf=0,suffix="btapods_AUeast")
```

```
#AUsouth
omega=as.matrix(read.table("merged_rep_filtered_dummy_edit_au_uk_AUsouth_baypass_mat_omega.out"))
anacore.snp.res=read.table("merged_rep_filtered_dummy_edit_au_uk_AUsouth_baypass_summary_pi_xtx.out",h=T)
pi.beta.coef=read.table("merged_rep_filtered_dummy_edit_au_uk_AUsouth_baypass_summary_beta_params.out",h=T)$Mean
bta14.data<-geno2YN("merged_rep_filtered_dummy_edit_au_uk_AUsouth_baypass.txt")
simu.bta<-simulate.baypass(omega.mat=omega, nsnp=5000, sample.size=bta14.data$NN,
beta.pi=pi.beta.coef,pi.maf=0,suffix="btapods_AUsouth")
```

XtX calibration; get the pod XtX

```
pod.xtx=read.table("G.btapods_AUeast_summary_pi_xtx.out",h=T)$M_XtX
```

```
pod.xtx=read.table("G.btapods_AUsouth_summary_pi_xtx.out",h=T)$M_XtX
```

Compute the 1% threshold

```
pod.thresh=quantile(pod.xtx,probs=0.99) #AUeast: 4.127359
pod.thresh=quantile(pod.xtx,probs=0.99) #AUsouth: 4.34993
```

Filter the data for the outlier snps:

```
module load bedtools/2.27.1
```

```
#Find outliers that are above theanapod threshold
cat merged_rep_filtered_dummy_edit_au_uk_AUeast_baypass_summary_pi_xtx.out | awk '$6>4.127359' > outliers_SV_AUeast.txt
cat merged_rep_filtered_dummy_edit_au_uk_AUsouth_baypass_summary_pi_xtx.out | awk '$6>4.34993' > outliers_SV_AUsouth.txt
#create list of SNPs in VCF, assign line numbers that can be used to find matching line numbers in outliers (SNP ID is lost in bayescan, line numbers used as signifiers).
grep -v "^#" ../merged_rep_filtered_dummy_edit_au_uk.recode.vcf | cut -f1-3 | awk '{print $0"\t"NR}'  > merged_rep_filtered_dummy_edit_au_uk_SNPlist.txt

#list of outlier SNPS
awk 'FNR==NR{a[$1];next} ($4) in a)' outliers_SV_AUeast.txt ./merged_rep_filtered_dummy_edit_au_uk_SNPlist.txt | awk '{print $3}'  > outliers_SV_AUeast_SNPlist.txt
awk 'FNR==NR{a[$1];next} ($4) in a)' outliers_SV_AUsouth.txt ./merged_rep_filtered_dummy_edit_au_uk_SNPlist.txt | awk '{print $3}' > outliers_SV_AUsouth_SNPlist.txt
comm <(sort outliers_SV_AUeast_SNPlist.txt) <(sort outliers_SV_AUsouth_SNPlist.txt) > outliers_SV_AUboth_SNPlist.txt
```

Combine outliers

```
module load plink/1.90b6.7
module load vcftools/0.1.16

cd /srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/sv_outliers

cat ../sv_bayescan/bayscan_outliers_svcf_SNPs.txt  ../sv_baypass/outliers_SV_AUboth_SNPlist.txt > sv_outliers_bayescan_baypass.txt

#Create list on non outlier SVs
grep -v "^#" ../merged_norep_mq_filtered_dummy_edit_au_uk.txt   | cut -f3 > all_SNPs.txt
comm -23 <(sort all_SNPs.txt) <(sort sv_outliers_bayescan_baypass.txt)  > sv_nonoutliers_vcf_SNPs.txt
```

PCA of outliers (SVs)

```
module load plink/1.90b6.7
module load vcftools/0.1.16

#VCF of just the outliers in AU and UK individuals for PCA
vcftools --vcf ../merged_rep_filtered_dummy_edit_au_uk.recode.vcf --snps sv_outliers_bayescan_baypass.txt --recode --recode-INFO-all --
out  merged_rep_filtered_dummy_edit_au_uk_outlier

#VCF of just the non-outliers in AU and UK individuals, needed for betascan below
vcftools --vcf ../merged_rep_filtered_dummy_edit_au_uk.recode.vcf --exclude sv_outliers_bayescan_baypass.txt --recode --recode-INFO-all --
out  merged_rep_filtered_dummy_edit_au_uk_nonoutlier

vcftools --vcf merged_rep_filtered_dummy_edit_au_uk_outlier.recode.vcf --plink --out merged_rep_filtered_dummy_edit_au_uk_outlier.plink
plink --file merged_rep_filtered_dummy_edit_au_uk_outlier.plink --pca --out merged_rep_filtered_dummy_edit_au_uk_outlier --make-rel
```

PCA Plot

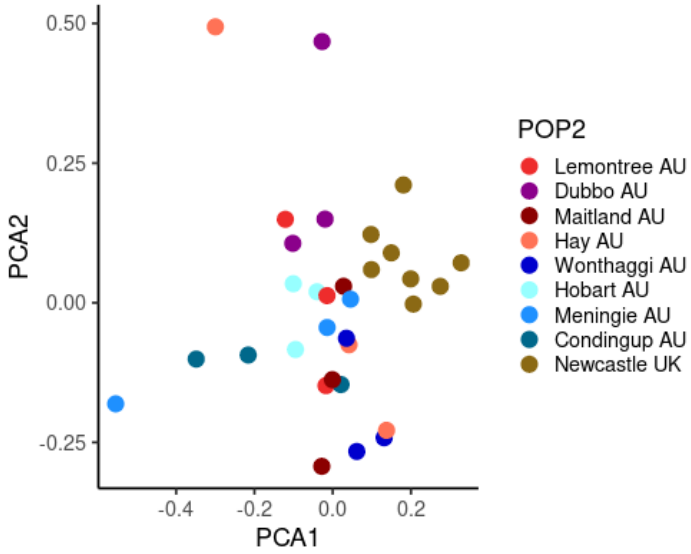
```
module load R/3.5.3
R
```

```
library(RColorBrewer)
library(SNPRelate)
library(gdsfmt)
library(scales)
library(adegenet)
library(pegas)
library(ggplot2)
library(ape)
library(poppr)
library(rgl)
library(dplyr)
```

```
setwd("/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/sv_outliers")
```

```
#Au samples
pca.eigenvec <- read.table("merged_rep_filtered_dummy_edit_au_uk_outlier.eigenvec", sep=" ")

head(round(pca.eigenvec, 2))
pca_g1 <- data.frame(sample.id = pca.eigenvec$V1,
                     EV1 = pca.eigenvec$V3,    # the first eigenvector
                     EV2 = pca.eigenvec$V4,    # the second eigenvector
                     EV3 = pca.eigenvec$V5,    # the second eigenvector
                     stringsAsFactors = FALSE)
head(pca_g1)
pca_g1

# add labels by population in correct order at vcf file

population <- read.table("/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/snp_popgen/PCA/populations.txt", header = TRUE, sep = "\t")
population2 <- population %>% filter(POP != "New York", POP != "Munglinup")
pca_g1 <- cbind(pca_g1,population2)
pca_g1

pca_g1$POP2 <- factor(pca_g1$POP, levels = c("Lemontree", "Dubbo", "Maitland","Hay","Wonthaggi","Hobart","Meningie","Condingup","Munglinup","Newcastle"))

levels(pca_g1$POP2) <- c("Lemontree AU", "Dubbo AU", "Maitland AU","Hay
AU","Wonthaggi AU","Hobart AU","Meningie AU","Condingup AU","Munglinup  AU","Newcastle UK")
```

```
pca_g1$POP2


png("Sv5_SV_PCA12.png", width=500, height=400)
ggplot(pca_g1,aes(x=EV1,y=EV2,col=POP2))+
geom_point(size=5,alpha=1)+
scale_color_manual(values = c("firebrick2","darkmagenta","darkred","coral1","blue3","darkslategray1","dodgerblue","deepskyblue4","goldenrod4"))+
theme_classic(base_size = 18) + xlab("PCA1") + ylab("PCA2")
dev.off()
```



Balancing selection (SNPS)

```
cd /srv/scratch/z5188231/KStuart.Starling-Aug18/programs
git clone https://github.com/ksiewert/BetaScan.git
module load python/2.7.15
python BetaScan.py -help
```

usage: BetaScan.py [-h] -i I [-o O] [-w W] [-onewin] [-p P] [-fold] [-B2]

[-m M] [-std] [-theta THETA] [-thetaMap THETAMAP]

[-thetaPerSNP THETAPERSNP] [-DivTime DIVTIME]

BetaScan.py: error: argument -h/--help: ignored explicit argument 'elp'

Running BetaScan on each chrom separately, then cat results

Tutorial · ksiewert/BetaScan Wiki · GitHub

```
cd /srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/snp_betascan
module load python/2.7.15
module load gcc/7.3.0
module load glactools/1.0.8
BS=/srv/scratch/z5188231/KStuart.Starling-Aug18/programs/BetaScan
FAI=/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/genome/Sturnus_vulgaris_2.3.1.simp.fasta.fai
VCF=/srv/scratch/z5188231/KStuart.Starling-
Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/wgs_variantsgenotyped_filtered_miss50_UK_noIndel_WithIds_maf015.recode.vcf

mkdir vcf_chrom_split
```

```
mkdir betascores
echo 'Position Beta1*' > betascores.allchroms.txt
echo 'SNPID' > betascore.SNPID.allchroms.txt
#have to do MAF filtering before hand so that it is easier to link up a beta score to a snp using the below code. Only using autosome chroms.
for NUM in 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
do
vcftools --vcf $VCF --chr starling${NUM} --recode --recode-INFO-all --out vcf_chrom_split/chrom${NUM}
glactools vcfm2acf --onlyGT --fai $FAI vcf_chrom_split/chrom${NUM}.recode.vcf > vcf_chrom_split/chrom${NUM}.acf.gz
glactools acf2betascan --fold vcf_chrom_split/chrom${NUM}.acf.gz > vcf_chrom_split/chrom${NUM}.beta.txt
python ${BS}/BetaScan.py -i vcf_chrom_split/chrom${NUM}.beta.txt -fold -o betascores/chrom${NUM}.betascores.txt
awk -v my_var=$NUM '{print $0,"\t", "starling" my_var}' betascores/chrom${NUM}.betascores.txt | awk '{print  $3 "_"
$1,"\t", $0}'  > betascores/chrom${NUM}.betascores.format.txt
tail -n +2 betascores/chrom${NUM}.betascores.txt | cat >> betascores.allchroms.txt
grep -v "^#" vcf_chrom_split/chrom${NUM}.recode.vcf | cut -f3 | cat >> betascore.SNPID.allchroms.txt
done

paste -d'\t' betascore.SNPID.allchroms.txt  betascores.allchroms.txt > betascores.allchroms.ID.txt
cut -f3  betascores.allchroms.ID.txt  > betascores_only.txt
```

Grab Betascan scores for outlier and nonoutlier snps:

```
#SNPs
#match first column of files to extract betascores for outlier and non outlier SNPs
awk -F'\t' 'NR==FNR{c[$1]++;next};c[$1]' ../snp_outliers/snp_outliers_bayescan_baypass.txt betascores.allchroms.ID.txt  | cut -f3 > betascores_only_outliers.txt
awk -F'\t' 'NR==FNR{c[$1]++;next};c[$1]' ../snp_outliers/snp_nonoutliers_vcf_SNPs.txt betascores.allchroms.ID.txt | cut -f3 > betascores_only_nonoutliers.txt
#to quickly look at what the data looks like
# datamash --header-out mean 1 median 1 perc:95 1 perc:99 1 <  betascores_only_outliers.txt
```

Grab Betascan scores for outlier and nonoutlier SVs:

```
#SVs
#turn 3 column betascore file into 4 column bed file
module load bedtools/2.27.1
awk '{print $1"\t"($2+1)"\t"$3}' betascores.allchroms.ID.txt  | sed 's/_/\t/g' | tail -n +2 > betascores.allchroms.ID.4col.bed
```

From email exchange

(b) For inversions use all SNPs under the inversion

(c) for insertions/duplications/deletions/translocations use the SNPs within 1000bp upstream of the break end. I think I should restrict it to just upstream because this grabs SNPs from just before the SV, and means I don't have to work out the genotypes of the individuals underlying the SNP data (i.e. if some have the insertion and others don't presumably the downstream break end will only be applicable to some of them, and therefore grabbing SNPs from this position would not make sense).

KAT NOTES: 1) keep it at 1000, 2) change ins to same method, 3) check additional divergence sel method, if not leave as is (without pcadapt)

```
module load bedtools/2.27.1
```

```
#SV: TYPE SPECIFIC APPROACH
#Find overlap between the SNP and SV outliers bed files. SV files have buffer of 1000pb up and downstream of break ends (make bed file using first line).
#all outliers are del's so apply DEL approach
grep -v "^#" ../sv_outliers/merged_rep_filtered_dummy_edit_au_uk_outlier.recode.vcf | sed 's/;/\t/g' | cut -f1,2,3,10,11 | sed 's/SVLEN=-\|SVLEN=//g' | sed
's/SVTYPE=//g'  | awk -v FS='\t' -v OFS='\t' '{print $1, $2-1000, $2}' > SV_outliers_1000bpbreakend.bed
bedtools intersect -wb -a betascores.allchroms.ID.4col.bed -b SV_outliers_1000bpbreakend.bed | awk -v FS='\t' -v OFS='\t' '{print $4, $5 "_"
$6}' > betascores_only_outliers_SV1kbbreakend.txt
#grabbing the trimmed mean for each SNP, the primary way I report it in manuscript
datamash --sort --group 2 trimmean:0.1 1 <betascores_only_outliers_SV1kbbreakend.txt | cut -f 2 > betascores_only_outliers_SV1kbbreakend_aggregate.txt
#grabbing all betascores, the secondary way I report it in supp mat
cut -f1 betascores_only_outliers_SV1kbbreakend.txt > betascores_only_outliers_SV1kbbreakend_average.txt

#likewise do the same for the non-outlier SVs, but include SV type
#need to make sure the bed start point is >=0.
grep -v "^#" ../sv_outliers/merged_rep_filtered_dummy_edit_au_uk_nonoutlier.recode.vcf | sed 's/;/\t/g' | cut -f1,2,3,10,11 | sed 's/SVLEN=-\|SVLEN=//g' |  sed
's/SVTYPE=//g' | awk -v FS='\t' -v OFS='\t' '{print $1, $2-1000, $2,$5}' | awk  -v FS='\t' -v OFS='\t' '{print $1, ($2<0?0:$2), $3, $4}' > SV_nonoutliers_1000bp.bed
bedtools intersect -wb -a betascores.allchroms.ID.4col.bed -b SV_nonoutliers_1000bp.bed  | awk -v FS='\t' -v OFS='\t' '{print $4, $8, $5 "_" $6}' >
betascores_only_nonoutliers_SV1kbbreakend.txt
#grabbing the trimmed mean for each SNP, the primary way I report it in manuscript
datamash --sort --group 3 trimmean:0.1 1 first 2 < betascores_only_nonoutliers_SV1kbbreakend.txt > betascores_only_nonoutliers_SV1kbbreakend_aggregate.txt
awk '$3 == "DEL"' betascores_only_nonoutliers_SV1kbbreakend_aggregate.txt | cut -f 2 > betascores_only_nonoutliers_SV1kbbreakend_aggregateDEL.txt
awk '$3 == "DUP"' betascores_only_nonoutliers_SV1kbbreakend_aggregate.txt | cut -f 2 > betascores_only_nonoutliers_SV1kbbreakend_aggregateDUP.txt
```

```
awk '$3 == "INV"' betascores_only_nonoutliers_SV1kbbreakend_aggregate.txt | cut -f 2 > betascores_only_nonoutliers_SV1kbbreakend_aggregateINV.txt
awk '$3 == "INS"' betascores_only_nonoutliers_SV1kbbreakend_aggregate.txt | cut -f 2 > betascores_only_nonoutliers_SV1kbbreakend_aggregateINS.txt
awk '$3 == "TRA"' betascores_only_nonoutliers_SV1kbbreakend_aggregate.txt | cut -f 2 > betascores_only_nonoutliers_SV1kbbreakend_aggregateTRA.txt
#grabbing all betascores, the secondary way I report it in supp mat
awk '$2 == "DEL"' betascores_only_nonoutliers_SV1kbbreakend.txt | cut -f 1 > betascores_only_nonoutliers_SV1kbbreakend_averageDEL.txt
awk '$2 == "DUP"' betascores_only_nonoutliers_SV1kbbreakend.txt | cut -f 1 > betascores_only_nonoutliers_SV1kbbreakend_averageDUP.txt
awk '$2 == "INV"' betascores_only_nonoutliers_SV1kbbreakend.txt | cut -f 1 > betascores_only_nonoutliers_SV1kbbreakend_averageINV.txt
awk '$2 == "INS"' betascores_only_nonoutliers_SV1kbbreakend.txt | cut -f 1 > betascores_only_nonoutliers_SV1kbbreakend_averageINS.txt
awk '$2 == "TRA"' betascores_only_nonoutliers_SV1kbbreakend.txt | cut -f 1 > betascores_only_nonoutliers_SV1kbbreakend_averageTRA.txt
```

Plotting

```
module load bioconductor/3.10
R

library(ggpubr)
library(FSA)
```

```
setwd("/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/snp_betascan")

SNP_outliers <- read.table("betascores_only_outliers.txt")
SNP_outliers <- SNP_outliers %>% mutate(type = "1-SNPoutlier")
SNP_nonoutliers <- read.table("betascores_only_nonoutliers.txt")
SNP_nonoutliers  <- SNP_nonoutliers %>% mutate(type = "2-SNPnonoutlier")

#AGGREGATE (primary method)
SV_outliers <- read.table("betascores_only_outliers_SV1kbbreakend_aggregate.txt")
SV_outliers  <- SV_outliers %>% mutate(type = "3-SVoutlier")
SV_nonoutliersDEL <- read.table("betascores_only_nonoutliers_SV1kbbreakend_aggregateDEL.txt")
SV_nonoutliersDEL <- SV_nonoutliersDEL %>% mutate(type = "4-SVnonoutliersDEL")
SV_nonoutliersDUP <- read.table("betascores_only_nonoutliers_SV1kbbreakend_aggregateDUP.txt")
SV_nonoutliersDUP <- SV_nonoutliersDUP %>% mutate(type = "5-SVnonoutliersDUP")
SV_nonoutliersINS <- read.table("betascores_only_nonoutliers_SV1kbbreakend_aggregateINS.txt")
SV_nonoutliersINS <- SV_nonoutliersINS %>% mutate(type = "6-SVnonoutliersINS")
SV_nonoutliersTRA <- read.table("betascores_only_nonoutliers_SV1kbbreakend_aggregateTRA.txt")
SV_nonoutliersTRA <- SV_nonoutliersTRA %>% mutate(type = "7-SVnonoutliersTRA")
SV_nonoutliersINV <- read.table("betascores_only_nonoutliers_SV1kbbreakend_aggregateINV.txt")
SV_nonoutliersINV <- SV_nonoutliersINV %>% mutate(type = "8-SVnonoutliersINV")

beta_aggregate <- rbind(SNP_outliers, SNP_nonoutliers, SV_outliers, SV_nonoutliersDEL, SV_nonoutliersDUP, SV_nonoutliersTRA, SV_nonoutliersINV)

bartlett.test(V1 ~ type , data = beta_aggregate )
```

Bartlett test of homogeneity of variances

data: V1 by type

Bartlett's K-squared = 174.89, df = 6, p-value < 2.2e-16

```
#non-parametric, as barlett's was significant
kruskal.test(V1 ~ type, data = beta_aggregate)
dunnTest(V1 ~ type, data = beta_aggregate)
```

Kruskal-Wallis rank sum test

data: V1 by type

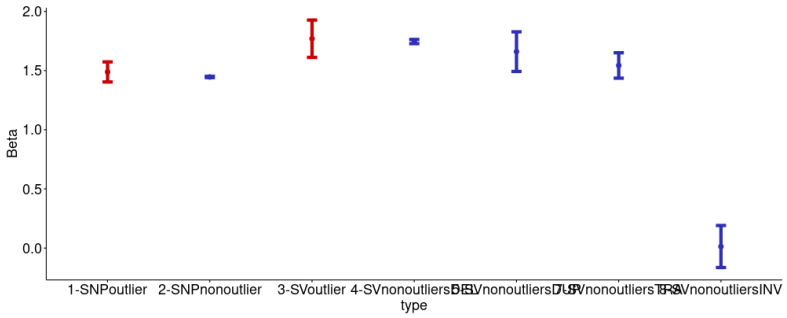
Kruskal-Wallis chi-squared = 647.88, df = 6, p-value < 2.2e-16

```
png("Sv5_beta_aggregate.png", width=1000, height=400)
ggerrorplot(beta_aggregate, x = "type", y = "V1", desc_stat = "mean_se", error.plot = "errorbar", add =
```

```
"mean", color =  c("#cc0000", "#2e2eb8","#cc0000", "#2e2eb8","#2e2eb8","#2e2eb8","#2e2eb8"), ylab="Beta",
size=2, font.x = c(18, "plain", "black"), font.y = c(18, "plain", "black"),font.tickslab = c(18,"plain", "black") )
dev.off()
```

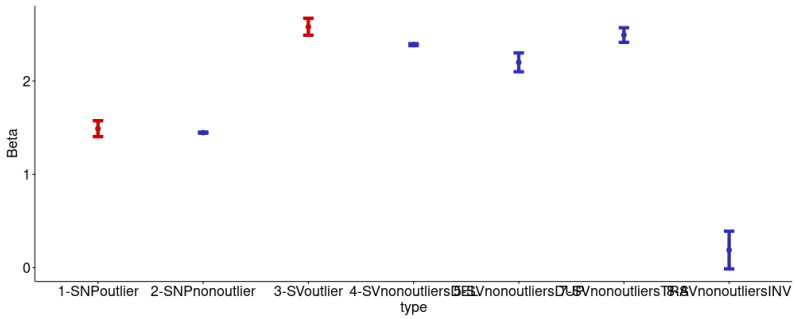
AGGREGATE:

https://stackoverflow.com/questions/12910841/custom-spacing-between-x-axis-labels-in-ggplot




```
#AVERAGE(secondary method)
SV_outliers <- read.table("betascores_only_outliers_SV1kbbreakend_average.txt")
SV_outliers  <- SV_outliers  %>%  mutate(type = "3-SVoutlier")
SV_nonoutliersDEL <- read.table("betascores_only_nonoutliers_SV1kbbreakend_averageDEL.txt")
SV_nonoutliersDEL <- SV_nonoutliersDEL %>%  mutate(type = "4-SVnonoutliersDEL")
SV_nonoutliersDUP <- read.table("betascores_only_nonoutliers_SV1kbbreakend_averageDUP.txt")
SV_nonoutliersDUP <- SV_nonoutliersDUP %>%  mutate(type = "5-SVnonoutliersDUP")
SV_nonoutliersINS <- read.table("betascores_only_nonoutliers_SV1kbbreakend_averageINS.txt")
SV_nonoutliersINS <- SV_nonoutliersINS %>%  mutate(type = "6-SVnonoutliersINS")
SV_nonoutliersTRA <- read.table("betascores_only_nonoutliers_SV1kbbreakend_averageTRA.txt")
SV_nonoutliersTRA <- SV_nonoutliersTRA %>%  mutate(type = "7-SVnonoutliersTRA")
SV_nonoutliersINV <- read.table("betascores_only_nonoutliers_SV1kbbreakend_averageINV.txt")
SV_nonoutliersINV <- SV_nonoutliersINV %>%  mutate(type = "8-SVnonoutliersINV")

beta_average <- rbind(SNP_outliers, SNP_nonoutliers, SV_outliers, SV_nonoutliersDEL, SV_nonoutliersDUP, SV_nonoutliersTRA, SV_nonoutliersINV)
kruskal.test(V1 ~ type, data = beta_average)
dunnTest(V1 ~ type, data = beta_average)

png("Sv5_beta_average.png", width=1000, height=400)
ggerrorplot(beta_average, x = "type", y = "V1", desc_stat = "mean_se", error.plot = "errorbar", add = "mean",
color =  c("#cc0000", "#2e2eb8","#cc0000", "#2e2eb8","#2e2eb8","#2e2eb8","#2e2eb8"), ylab="Beta", size=2,
font.x = c(18, "plain", "black"), font.y = c(18, "plain", "black"),font.tickslab = c(18,"plain", "black") )
dev.off()
```

AVERAGE:



Plotting the circos:

Summarise the data

```
cd /srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/analysis/balancing_selection_2022/plotting

awk 'FNR==NR{a[$1];next} (($4) in a)' ../snp_bayescan/bayscan_outliers_AUeast_vcf_numbers.txt ../wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_SNPs.txt | cut -f1-3 > SNP_Bayescan_AUeast.txt
awk 'FNR==NR{a[$1];next} (($4) in a)' ../snp_bayescan/bayscan_outliers_AUsouth_vcf_numbers.txt ../wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_SNPs.txt | cut -f1-3 > SNP_Bayescan_AUsouth.txt
awk 'FNR==NR{a[$1];next} (($4) in a)' ../snp_baypass/outliers_SNP_AUeast.txt ../wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_SNPs.txt | cut -f1-3 > SNP_Baypass_AUeast.txt
awk 'FNR==NR{a[$1];next} (($4) in a)' ../snp_baypass/outliers_SNP_AUsouth.txt ../wgs_variantsgenotyped_filtered_miss50_r2_au_uk_thin_SNPs.txt | cut -f1-3 > SNP_Baypass_AUsouth.txt

awk 'FNR==NR{a[$1];next} (($4) in a)'
../sv_bayescan/bayscan_outliers_AUsouth_svcf_numbers.txt ../sv_baypass/merged_rep_filtered_dummy_edit_au_uk_SNPlist.txt | cut -f1-3 > SV_Bayescan_AUsouth.txt
awk 'FNR==NR{a[$1];next} (($4) in a)' ../sv_baypass/outliers_SV_AUeast.txt ../sv_baypass/merged_rep_filtered_dummy_edit_au_uk_SNPlist.txt | cut -f1-3 > SV_Baypass_AUeast.txt
awk 'FNR==NR{a[$1];next} (($4) in a)' ../sv_baypass/outliers_SV_AUsouth.txt ../sv_baypass/merged_rep_filtered_dummy_edit_au_uk_SNPlist.txt | cut -f1-3 > SV_Baypass_AUsouth.txt

cat * | cut -f3 > total_snps.txt
cat *AU*.txt | sort | uniq >total_sites.txt
cat SV*.txt | sort | uniq >total_sites_SV.txt
```

Creating the tracks:

```
module load bedtools/2.27.1
GENOME=/srv/scratch/z5188231/KStuart.Starling-Aug18/Sv5_AustraliaWGS/genome/Sturnus_vulgaris_2.3.1.simp.fasta.fai
module load samtools/1.10
cut -f1,2 ${GENOME} > sizes.genome
WIDTH=1000000
bedtools makewindows -g sizes.genome -w ${WIDTH} > svulgaris_${WIDTH}bps.bed
bedtools intersect -wb -a ../snp_betascan/betascores.allchroms.ID.4col.bed -b svulgaris_1000000bps.bed | awk -v FS='\t' -v OFS='\t' '{print $4, $8, $5 "_" $6}' > betascores_heatmap_v2.txt
datamash -R2 -W -s -g 2 mean 1 <betascores_heatmap_v2.txt > heatmap_balsel_v2.txt
```