Title: POPULATION GENOMICS OF THE COMMERCIAL CAPE HAKES
Authors: Sarah Forde1, Sophie von der Heyden2, Alan Le Moan3, Erica S Nielsen4, Deon Durholtz5, Paulus Kainge6, John Kathena6, Marek Lipinski7, Hilkka ON Ndjaula8, Conrad A Matthee2, and Romina Henriques1
Affiliations: 1Marine Genomics Group, Department of Biochemistry, Genetics and Microbiology, University of Pretoria, Private Bag X20, Hatfield, 0028, South Africa, 2Evolutionary Genomics Group, Department of Botany and Zoology, Stellenbosch University, Private Bag X1, Stellenbosch 7602, South Africa, 3D , 4University of California - Davis United States of America, 5Department of Forestry, Fisheries and Environment, Cape Town, South Africa, 6National Marine Information and Research Centre, Swakopmund, Namibia, 7S , 8Sam Nujoma Marine and Coastal Resources Research Centre, University of Namibia, Private Bag 13301, Windhoek, Namibia.

##
#
Supplementary Material 1
#
#
##

BIOINFORMATIVE PIPELINE

A) TRIMMING/QUALITY CONTROL

first round

#capensis data
```
trim_galore --illumina --paired -q 25 --trim-n --length 50 -o /mnt/lustre3p/users/sforde/capensis/02_trimming RH-CSWC_S2_L001_R1.fastq.gz RH-CSWC_S2_L001_R2.fastq.gz Rhenr-CCN_S1_L001_R1_002.fastq.gz Rhenr-CCN_S1_L001_R2_002.fastq.gz Rhenr-CNN_S2_L001_R1_002.fastq.gz Rhenr-CNN_S2_L001_R2_002.fastq.gz Rhenr-CWC_S3_L001_R1_002.fastq.gz Rhenr-CWC_S3_L001_R2_002.fastq.gz
```

#paradoxus data
```
trim_galore --illumina --paired -q 25 --trim-n --length 50 --fastqc -o /mnt/lustre3p/users/sforde/paradoxus/02_trimming RH-PWC_S4_L001_R2_001.fastq.gz RH-PNN_S3_L001_R2_001.fastq.gz RH-PWC_S4_L001_R1_001.fastq.gz RH-PNN_S3_L001_R1_001.fastq.gz Rhenr-PCN_S4_L001_R2_002.fastq.gz Rhenr-POR_S5_L001_R2_002.fastq.gz Rhenr-PCN_S4_L001_R1_002.fastq.gz Rhenr-POR_S5_L001_R1_002.fastq.gz
```

```
# second round

#Over-represented sequences in capensis identified with the module "Overrepresented sequences" in FASTQC:
#GATCACTTTGAAAATCCCATGCATTCCTATGGGGGGATTTTAATTTGGCC
#GATCATCCTCGAGGGAAGGTCGTAGAGAGATGGGTCCCCGATAATGGTCC
#GATCATTGATCATCCTCGAGGGAAGGTCGTAGAGAGATGGGTCCCCGATA
#GATCATCGTCAAGGGAAGGTCCTAGAGGGTTAGGGTTAGGTATAGGGGTAATAACCACGATGATC
#GATCATCGTGGTTATTACCCCTATACCTAACCCTAACCCTCTAGGACCTTCCCTTGACGATGATC

#Over-represented sequences in paradoxus identified with the module "Overrepresented sequences" in FASTQC:
#TATATATATATATATATATATATATATATATATATATATATATATATATATATA
#ATATATATATATATATATATATATATATATATATATATATATATATATATATAT
#GTTACCGAATCTGGAAGTGACTACCATTCCACAATGAAACAACAGAAATG


#capensis data
trim_galore --paired -a GATCACTTTGAAAATCCCATGCATTCCTATGGGGGGATTTTAATTTGGCC -a
GATCATCCTCGAGGGAAGGTCGTAGAGAGATGGGTCCCCGATAATGGTCC -a GATCATTGATCATCCTCGAGGGAAGGTCGTAGAGAGATGGGTCCCCGATA -a
GATCATCGTCAAGGGAAGGTCCTAGAGGGTTAGGGTTAGGTATAGGGGTAATAACCACGATGATC -a
GATCATCGTGGTTATTACCCCTATACCTAACCCTAACCCTCTAGGACCTTCCCTTGACGATGATC -a
GATCATGTTCTGTAAGTTTGGGGAAAATTTCATTTTTGGTCCAATTTTCA -a GATCCTGGCGTCCTGCCGCTGCTGATGTTGAAGATTTCAGAGCTCTAGCT -a
GATCAATGTAATGCACTTGATGTGTGTGGACTTGTTGAAAATGTGATAAA -a GATCATCTTTGCTTTGGGTTTGGAGTTTGACTTTTGACCTGAGATGTTCA -a
GATCACTGTCGCCCTCCGGTGGCCAGACTCAGGAACTGAACATGGAGATG -a GATCACCTTTGACCCTTACCTTTATTCTAACCCTGACCCCTAGGGGGGCT -a
GATCCCCAAAGCTATGGATTCTGTTTCTTATTCATTAAAATCCCTTTTTA -a TATATATATATATATATATATATATATATATATATATATATATATATATA -a
ATATATATATATATATATATATATATATATATATATATATATATATATAT -a GTTACCGAATCTGGAAGTGACTACCATTCCACAATGAAACAACAGAAATG --clip_R1 5
--clip_R2 5 --three_prime_clip_R1 5 --three_prime_clip_R2 5 --fastqc -o
/mnt/lustre3p/users/sforde/capensis/02_trimming/second_round RH-CSWC_S2_L001_R1_val_1.fq.gz RH-
CSWC_S2_L001_R2_val_2.fq.gz Rhenr-CCN_S1_L001_R1_002_val_1.fq.gz Rhenr-CCN_S1_L001_R2_002_val_2.fq.gz Rhenr-
CNN_S2_L001_R1_002_val_1.fq.gz Rhenr-CNN_S2_L001_R2_002_val_2.fq.gz Rhenr-CWC_S3_L001_R1_002_val_1.fq.gz Rhenr-
CWC_S3_L001_R2_002_val_2.fq.gz

#paradoxus data
trim_galore --paired -a GATCACTTTGAAAATCCCATGCATTCCTATGGGGGGATTTTAATTTGGCC -a
GATCATCCTCGAGGGAAGGTCGTAGAGAGATGGGTCCCCGATAATGGTCC -a GATCATTGATCATCCTCGAGGGAAGGTCGTAGAGAGATGGGTCCCCGATA -a
GATCATCGTCAAGGGAAGGTCCTAGAGGGTTAGGGTTAGGTATAGGGGTAATAACCACGATGATC -a
GATCATCGTGGTTATTACCCCTATACCTAACCCTAACCCTCTAGGACCTTCCCTTGACGATGATC -a
GATCATGTTCTGTAAGTTTGGGGAAAATTTCATTTTTGGTCCAATTTTCA -a GATCCTGGCGTCCTGCCGCTGCTGATGTTGAAGATTTCAGAGCTCTAGCT -a
GATCAATGTAATGCACTTGATGTGTGTGGACTTGTTGAAAATGTGATAAA -a GATCATCTTTGCTTTGGGTTTGGAGTTTGACTTTTGACCTGAGATGTTCA -a
GATCACTGTCGCCCTCCGGTGGCCAGACTCAGGAACTGAACATGGAGATG -a GATCACCTTTGACCCTTACCTTTATTCTAACCCTGACCCCTAGGGGGGCT -a
```

```
GATCCCCAAAGCTATGGATTCTGTTTCTTATTCATTAAAATCCCTTTTTA -a TATATATATATATATATATATATATATATATATATATATATATATATATATA -a
ATATATATATATATATATATATATATATATATATATATATATATATATATAT -a GTTACCGAATCTGGAAGTGACTACCATTCCACAATGAAACAACAGAAATG --clip_R1 5
--clip_R2 5 --three_prime_clip_R1 5 --three_prime_clip_R2 5 --fastqc -o
/mnt/lustre3p/users/sforde/paradoxus/02_trimming/second_round RH-PWC_S4_L001_R2_001_val_1.fq.gz RH-
PWC_S4_L001_R1_001_val_2.fq.gz RH-PNN_S3_L001_R2_001_val_1.fq.gz RH-PNN_S3_L001_R1_001_val_2.fq.gz Rhenr-
PCN_S4_L001_R2_002_val_1.fq.gz Rhenr-PCN_S4_L001_R1_002_val_2.fq.gz Rhenr-POR_S5_L001_R2_002_val_1.fq.gz Rhenr-
POR_S5_L001_R1_002_val_2.fq.gz


# third round

#capensis data
trim_galore --paired --clip_R1 5 --clip_R2 5 --three_prime_clip_R1 5 --three_prime_clip_R2 5 --fastqc -o
/mnt/lustre3p/users/sforde/capensis/02_trimming/third_round Rhenr-CNN_S2_L001_R1_002_val_1_val_1.fq.gz Rhenr-
CNN_S2_L001_R2_002_val_2_val_2.fq.gz Rhenr-CCN_S1_L001_R1_002_val_1_val_1.fq.gz Rhenr-
CCN_S1_L001_R2_002_val_2_val_2.fq.gz RH-CSWC_S2_L001_R1_val_1_val_1.fq.gz RH-CSWC_S2_L001_R2_val_2_val_2.fq.gz
Rhenr-CWC_S3_L001_R1_002_val_1_val_1.fq.gz Rhenr-CWC_S3_L001_R2_002_val_2_val_2.fq.gz

#paradoxus
trim_galore --paired --clip_R1 5 --clip_R2 5 --three_prime_clip_R1 5 --three_prime_clip_R2 5 --fastqc -o
/mnt/lustre3p/users/sforde/paradoxus/02_trimming/third_round Rhenr-PCN_S4_L001_R1_002_val_2_val_2.fq.gz Rhenr-
PCN_S4_L001_R2_002_val_1_val_1.fq.gz RH-PWC_S4_L001_R1_001_val_2_val_2.fq.gz RH-PWC_S4_L001_R2_001_val_1_val_1.fq.gz
Rhenr-POR_S5_L001_R1_002_val_2_val_2.fq.gz Rhenr-POR_S5_L001_R2_002_val_1_val_1.fq.gz RH-
PNN_S3_L001_R1_001_val_2_val_2.fq.gz RH-PNN_S3_L001_R2_001_val_1_val_1.fq.gz


#       B) FLASH

module add chpc/BIOMODULES
module load FLASH/1.2.11

for i in $(cat capensis_list); do
flash capensis/02_trimming/third_round/$i"_R1.fq.gz" capensis/02_trimming/third_round/$i"_R2.fq.gz" -z -o
/mnt/lustre3p/users/sforde/capensis/03_flash/$i 2>&1 | tee FLASH_default.log; done

for i in $(cat paradoxus_list); do
flash paradoxus/02_trimming/third_round/$i"_R1.fq.gz" paradoxus/02_trimming/third_round/$i"_R2.fq.gz" -z -o
/mnt/lustre3p/users/sforde/paradoxus/03_flash/$i 2>&1 | tee FLASH_default.log; done
```

```
#       C) mtDNA MAP

module add chpc/BIOMODULES
module add chpc/java/9.0.1
module load BWA/0.7.17
module load samtools/1.9
module load bbmap/37.90

### create bwa index
bwa index ref_Mm.fasta

### mapping against reference genome: European hake [Merluccius merluccius (accession number FR751402)]

# CAPENSIS

for i in $(cat 04_capensis_bwa_list.txt); do
bwa mem ref_Mm.fasta /mnt/lustre3p/users/sforde/capensis/03_flash/$i"_out.extendedFrags.fastq.gz" -a >
/mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_out.extendedFrags.sam"; done

for i in $(cat 04_capensis_bwa_list.txt); do
bwa mem ref_Mm.fasta /mnt/lustre3p/users/sforde/capensis/03_flash/$i"_out.notCombined_1.fastq.gz" -a >
/mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_out.notCombined_1.fastq.gz.sam"; done

for i in $(cat 04_capensis_bwa_list.txt); do
bwa mem ref_Mm.fasta /mnt/lustre3p/users/sforde/capensis/03_flash/$i"_out.notCombined_2.fastq.gz" -a >
/mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_out.notCombined_2.fastq.gz.sam"; done


# PARADOXUS

cd /mnt/lustre3p/users/sforde/paradoxus/03_flash/
for i in $(cat 04_paradoxus_bwa_list.txt); do
bwa mem ref_Mm.fasta /mnt/lustre3p/users/sforde/paradoxus/03_flash/$i"_out.extendedFrags.fastq.gz" -a >
/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_out.extendedFrags.sam"; done

for i in $(cat 04_paradoxus_bwa_list.txt); do
bwa mem ref_Mm.fasta /mnt/lustre3p/users/sforde/paradoxus/03_flash/$i"_out.notCombined_1.fastq.gz" -a >
/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_out.notCombined_1.fastq.gz.sam"; done
```

```
for i in $(cat 04_paradoxus_bwa_list.txt); do
bwa mem ref_Mm.fasta /mnt/lustre3p/users/sforde/paradoxus/03_flash/$i"_out.notCombined_2.fastq.gz" -a >
/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_out.notCombined_2.fastq.gz.sam"; done


### count how many reads have mapped to mtDNA

# CAPENSIS

for i in $(cat 04_capensis_bwa_list.txt); do
samtools view -S -c -F 4 /mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_combined.sam" >
/mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_combined.readcounts.txt"; done

for i in $(cat 04_capensis_bwa_list.txt); do
samtools view -S -c -F 4 /mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_1.fastq.gz.sam" >
/mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_1.readcounts.txt"; done

for i in $(cat 04_capensis_bwa_list.txt); do
samtools view -S -c -F 4 /mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_2.fastq.gz.sam" >
/mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_2.readcounts.txt"; done

# PARADOXUS

for i in $(cat 04_paradoxus_bwa_list.txt); do
samtools view -S -c -F 4 /mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_combined.sam" >
/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_combined.readcounts.txt"; done

for i in $(cat 04_paradoxus_bwa_list.txt); do
samtools view -S -c -F 4 /mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_1.fastq.gz.sam" >
/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_1.readcounts.txt"; done

for i in $(cat 04_paradoxus_bwa_list.txt); do
samtools view -S -c -F 4 /mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_2.fastq.gz.sam" >
/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_2.readcounts.txt"; done

### obtain the reads that mapped

# CAPENSIS
```

```
for i in $(cat 04_capensis_bwa_list.txt); do
samtools view -S -F 4 /mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_combined.sam" >
/mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_combined.mtdna_remove.txt"; done

for i in $(cat 04_capensis_bwa_list.txt); do
samtools view -S -F 4 /mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_1.fastq.gz.sam" >
/mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_1.mtdna_remove.txt"; done

for i in $(cat 04_capensis_bwa_list.txt); do
samtools view -S -F 4 /mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_2.fastq.gz.sam" >
/mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_2.mtdna_remove.txt"; done

# PARADOXUS

for i in $(cat 04_paradoxus_bwa_list.txt); do
samtools view -S -F 4 /mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_combined.sam" >
/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_combined.mtdna_remove.txt"; done

for i in $(cat 04_paradoxus_bwa_list.txt); do
samtools view -S -F 4 /mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_1.fastq.gz.sam" >
/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_1.mtdna_remove.txt"; done

for i in $(cat 04_paradoxus_bwa_list.txt); do
samtools view -S -F 4 /mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_2.fastq.gz.sam" >
/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_2.mtdna_remove.txt"; done


### then we use the files ".mtdna_remove.txt" to filter the original reads and take out the ones that mapped to
mtDNA, keeping only the non-mtDNA reads

# CAPENSIS
for i in $(cat 04_capensis_bwa_list.txt); do
filterbyname.sh in=/mnt/lustre3p/users/sforde/capensis/03_flash/$i"_out.extendedFrags.fastq.gz"
out=/mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_combined.clean.nomtdna.fq.gz"
names=/mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_combined.mtdna_remove.txt" include=f minlen=50 ow=t
filterbyname.sh in=/mnt/lustre3p/users/sforde/capensis/03_flash/$i"_out.notCombined_1.fastq.gz"
out=/mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_1.clean.nomtdna.fq.gz"
names=/mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_1.mtdna_remove.txt" include=f minlen=50 ow=t
```

```
filterbyname.sh in=/mnt/lustre3p/users/sforde/capensis/03_flash/$i"_out.notCombined_2.fastq.gz"
out=/mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_2.clean.nomtdna.fq.gz"
names=/mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_2.mtdna_remove.txt" include=f minlen=50 ow=t; done

# PARADOXUS
for i in $(cat 04_paradoxus_bwa_list.txt); do
filterbyname.sh in=/mnt/lustre3p/users/sforde/paradoxus/03_flash/$i"_out.extendedFrags.fastq.gz"
out=/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_combined.clean.nomtdna.fq.gz"
names=/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_combined.mtdna_remove.txt" include=f minlen=50 ow=t
filterbyname.sh in=/mnt/lustre3p/users/sforde/paradoxus/03_flash/$i"_out.notCombined_1.fastq.gz"
out=/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_1.clean.nomtdna.fq.gz"
names=/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_1.mtdna_remove.txt" include=f minlen=50 ow=t
filterbyname.sh in=/mnt/lustre3p/users/sforde/paradoxus/03_flash/$i"_out.notCombined_2.fastq.gz"
out=/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_2.clean.nomtdna.fq.gz"
names=/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_2.mtdna_remove.txt" include=f minlen=50 ow=t; done


### prepare the reads that did map to mtDNA to obtain a mtDNA dataset
# 1. sort the reads and turn them into bam files

# CAPENSIS

for i in $(cat 04_capensis_bwa_list.txt); do
samtools sort /mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_combined.sam" -o
/mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_combined.mtdna.bam"
samtools sort /mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_1.fastq.gz.sam" -o
/mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_1.mtdna.bam"
samtools sort /mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_2.fastq.gz.sam" -o
/mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_2.mtdna.bam"; done

# PARADOXUS

for i in $(cat 04_paradoxus_bwa_list.txt); do
samtools sort /mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_combined.sam" -o
/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_combined.mtdna.bam"
samtools sort /mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_1.fastq.gz.sam" -o
/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_1.mtdna.bam"
samtools sort /mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_2.fastq.gz.sam" -o
/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_2.mtdna.bam"; done
```

```
# 2. index the bam files

# CAPENSIS

for i in $(cat 04_capensis_bwa_list.txt); do
samtools index /mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_combined.mtdna.bam"
samtools index /mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_1.mtdna.bam"
samtools index /mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_2.mtdna.bam"; done

# PARADOXUS
for i in $(cat 04_paradoxus_bwa_list.txt); do
samtools index /mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_combined.mtdna.bam"
samtools index /mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_1.mtdna.bam"
samtools index /mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_2.mtdna.bam"; done

# 3. calculate the stats for a mapping quality (q) of 20 or above

# CAPENSIS

for i in $(cat 04_capensis_bwa_list.txt); do
samtools stats -q 20 -r ref_Mm.fasta /mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_combined.mtdna.bam" >
/mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_combined.mtdna.stats.txt"
samtools stats -q 20 -r ref_Mm.fasta /mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_1.mtdna.bam" >
/mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_1.mtdna.stats.txt"
samtools stats -q 20 -r ref_Mm.fasta /mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_2.mtdna.bam" >
/mnt/lustre3p/users/sforde/capensis/04_bwa_map/$i"_2.mtdna.stats.txt"; done


# PARADOXUS

for i in $(cat 04_paradoxus_bwa_list.txt); do
samtools stats -q 20 -r ref_Mm.fasta /mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_combined.mtdna.bam" >
/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_combined.mtdna.stats.txt"
samtools stats -q 20 -r ref_Mm.fasta /mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_1.mtdna.bam" >
/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_1.mtdna.stats.txt"
samtools stats -q 20 -r ref_Mm.fasta /mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_2.mtdna.bam" >
/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/$i"_2.mtdna.stats.txt"; done
```

```
#       D) nDNA MAP

module add chpc/BIOMODULES
module add chpc/java/9.0.1
module load BWA/0.7.17
module load samtools/1.9
module load bbmap/37.90
module load perl/5.24.0
#module load anaconda2/2.2.0
#module load prinseq-lite/0.20.4
#module load jre/1.8.0

# index the M. capensis genome: ref_capensis.fna [M. capensis (accession number GCA_900312945.1)]

bwa index ref_capensis.fna

# 1. combined reads from FLASH

# CAPENSIS

for i in $(cat capensis_list.txt); do
bwa mem ref_capensis.fna
/mnt/lustre3p/users/sforde/capensis/04_bwa_map/D_filtered_reads/$i"_combined.clean.nomtdna.fq.gz" -a -t 28 >
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_combined.capensis_mapped.sam"; done

# obtain the stats
for i in $(cat capensis_list.txt); do
samtools stats -r ref_capensis.fna
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_combined.capensis_mapped.sam" >
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_combined.capensis_mapped.stats.txt"; done

# convert sam to bam, keeping only the reads that mapped with high quality.
for i in $(cat capensis_list.txt); do
samtools view -b -q 20 /mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_combined.capensis_mapped.sam" -o
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_combined.capensis_q20.bam"; done

# sort the reads
for i in $(cat capensis_list.txt); do
```

```
samtools sort /mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_combined.capensis_q20.bam" -o
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_combined.capensis_q20_sorted.bam"; done

# index the reads

for i in $(cat capensis_list.txt); do
samtools index /mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_combined.capensis_q20_sorted.bam"; done

# calculate the stats

for i in $(cat capensis_list.txt); do
samtools stats -r ref_capensis.fna
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_combined.capensis_q20_sorted.bam" >
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_combined.capensis_q20.stats.txt"; done


# PARADOXUS

for i in $(cat paradoxus_list.txt); do
bwa mem ref_capensis.fna
/mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/D_filtered_reads/$i"_combined.clean.nomtdna.fq.gz" -a -t 28 >
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_combined.capensis_mapped.sam"

# obtain the stats

samtools stats -r ref_capensis.fna
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_combined.capensis_mapped.sam" >
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_combined.capensis_mapped.stats.txt"

# convert sam to bam, keeping only the reads that mapped with high quality.

samtools view -b -q20 /mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_combined.capensis_mapped.stats.txt" -o
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_combined.capensis_q20.bam"

# sort the reads

samtools sort /mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_combined.capensis_q20.bam" -o
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_combined.capensis_q20_sorted.bam"
```

```
# index the reads

samtools index /mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_combined.capensis_q20_sorted.bam"

# calculate the stats

samtools stats -r ref_capensis.fna
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_combined.capensis_q20_sorted.bam" >
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_combined.capensis_q20.stats.txt"; done


# 2. non-combined reads

# CAPENSIS

for i in $(cat capensis_list.txt); do
bwa mem ref_capensis.fna /mnt/lustre3p/users/sforde/capensis/04_bwa_map/D_filtered_reads/$i"_1.clean.nomtdna.fq.gz"
-a -t 28 > /mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_1.capensis_mapped.sam"
bwa mem ref_capensis.fna /mnt/lustre3p/users/sforde/capensis/04_bwa_map/D_filtered_reads/$i"_2.clean.nomtdna.fq.gz"
-a -t 28 > /mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_2.capensis_mapped.sam"
samtools stats -r ref_capensis.fna /mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_1.capensis_mapped.sam" >
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_1.capensis_mapped.stats.txt"
samtools stats -r ref_capensis.fna /mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_2.capensis_mapped.sam" >
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_2.capensis_mapped.stats.txt"
samtools view -b -q20 /mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_1.capensis_mapped.sam" -o
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_1.capensis_q20.bam"
samtools view -b -q20 /mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_2.capensis_mapped.sam" -o
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_2.capensis_q20.bam"
samtools sort /mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_1.capensis_q20.bam" -o
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_1.capensis_q20_sorted.bam"
samtools sort /mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_2.capensis_q20.bam" -o
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_2.capensis_q20_sorted.bam"
samtools index /mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_1.capensis_q20_sorted.bam"
samtools index /mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_2.capensis_q20_sorted.bam"
samtools stats -r ref_capensis.fna /mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_1.capensis_q20_sorted.bam"
> /mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_1.capensis_q20.stats.txt"
samtools stats -r ref_capensis.fna /mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_2.capensis_q20_sorted.bam"
> /mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_2.capensis_q20.stats.txt"; done
```

```
# PARADOXUS

for i in $(cat paradoxus_list.txt); do
bwa mem ref_capensis.fna /mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/D_filtered_reads/$i"_1.clean.nomtdna.fq.gz"
-a -t 28 > /mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_1.capensis_mapped.sam"
bwa mem ref_capensis.fna /mnt/lustre3p/users/sforde/paradoxus/04_bwa_map/D_filtered_reads/$i"_2.clean.nomtdna.fq.gz"
-a -t 28 > /mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_2.capensis_mapped.sam"
samtools stats -r ref_capensis.fna /mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_1.capensis_mapped.sam" >
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_1.capensis_mapped.stats.txt"
samtools stats -r ref_capensis.fna /mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_2.capensis_mapped.sam" >
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_2.capensis_mapped.stats.txt"
samtools view -b -q20 /mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_1.capensis_mapped.sam" -o
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_1.capensis_q20.bam"
samtools view -b -q20 /mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_2.capensis_mapped.sam" -o
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_2.capensis_q20.bam"
samtools sort /mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_1.capensis_q20.bam" -o
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_1.capensis_q20_sorted.bam"
samtools sort /mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_2.capensis_q20.bam" -o
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_2.capensis_q20_sorted.bam"
samtools index /mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_1.capensis_q20_sorted.bam"
samtools index /mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_2.capensis_q20_sorted.bam"
samtools stats -r ref_capensis.fna
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_1.capensis_q20_sorted.bam" >
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_1.capensis_q20.stats.txt"
samtools stats -r ref_capensis.fna
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_2.capensis_q20_sorted.bam" >
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_2.capensis_q20.stats.txt"; done


# 3. merge the combined and notCombined bam files, so you only have one bam file per region

# CAPENSIS

for i in $(cat capensis_list.txt); do
samtools merge -f /mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_merged.bam"
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_combined.capensis_q20_sorted.bam"
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_1.capensis_q20_sorted.bam"
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_2.capensis_q20_sorted.bam"
samtools index /mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_merged.bam"
```

```
samtools stats -r ref_capensis.fna /mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_merged.bam" >
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/$i"_merged.stats.txt"; done

# PARADOXUS

for i in $(cat paradoxus_list.txt); do
samtools merge -f /mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_merged.bam"
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_combined.capensis_q20_sorted.bam"
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_1.capensis_q20_sorted.bam"
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_2.capensis_q20_sorted.bam"
samtools index /mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_merged.bam"
samtools stats -r ref_capensis.fna /mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_merged.bam" >
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/$i"_merged.stats.txt"; done


#       E) SUBSAMPLING

module add chpc/BIOMODULES
module load samtools/1.9

# capensis
samtools view -s 1.0 -b CNN_merged.bam > CNN_merged_subsample.bam
samtools view -s 0.312133204 -b CCN_merged.bam > CCN_merged_subsample.bam
samtools view -s 0.428504925 -b CWC_merged.bam > CWC_merged_subsample.bam
samtools view -s 0.330583245 -b CWC2_merged.bam > CWC2_merged_subsample.bam

# paradoxus
samtools view -s 1.0 -b PNN_merged.bam > PNN_merged_subsample.bam
samtools view -s 0.54618301 -b PCN_merged.bam > PCN_merged_subsample.bam
samtools view -s 0.571482717 -b POR_merged.bam > POR_merged_subsample.bam
samtools view -s 0.663246677 -b PWC_merged.bam > PWC_merged_subsample.bam
samtools view -s 0.498560572 -b PSW_merged.bam > PSW_merged_subsample.bam


#       F) BAM TO MPILEUP

module add chpc/BIOMODULES
module add samtools/0.1.18
```

```
# capensis
samtools mpileup -d 1000 -Q 20 -B -f /mnt/lustre3p/users/sforde/ref_capensis.fna CNN_merged_subsample.bam
CCN_merged_subsample.bam CWC_merged_subsample.bam CWC2_merged_subsample.bam -o
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/G_subsamples/capensis_capensis.mpileup >
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/G_subsamples/capensis_capensis.mpileup

# paradoxus
samtools mpileup -d 1000 -Q 20 -B -f /mnt/lustre3p/users/sforde/ref_capensis.fna PNN_merged_subsample.bam
PCN_merged_subsample.bam POR_merged_subsample.bam PSW_merged_subsample.bam -o
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/G_subsamples/paradoxus_capensis.mpileup >
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/G_subsamples/paradoxus_capensis.mpileup


#       G) MPILEUP TO SYNC

module add chpc/BIOMODULES
module load java/9.0.1
module load popoolation/2_1201

# capensis
java -ea -Xmx40g -jar /apps/chpc/bio/popoolation/2_1201/mpileup2sync.jar --input capensis_capensis.mpileup --output
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/G_subsamples/capensis_capensis.sync --fastq-type sanger --min-
qual 20 --threads 30 2> capensis_mpileup_to_sync.stderr

# paradoxus
java -ea -Xmx40g -jar /apps/chpc/bio/popoolation/2_1201/mpileup2sync.jar --input paradoxus_capensis.mpileup --output
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/G_subsamples/paradoxus_capensis.sync --fastq-type sanger --min-
qual 20 --threads 30 2> paradoxus_mpileup_to_sync.stderr


#       H) CALL & FILTER SNPS

module add chpc/BIOMODULES
module load popoolation/2_1201
module load perl/5.34.0

# OBTAIN RC FILE
# capensis
```

```
perl /apps/chpc/bio/popoolation/2_1201/snp-frequency-diff.pl --input
/mnt/lustre3p/users/sforde/capensis/05_nuclear_map/G_subsamples/capensis_capensis.sync --output-prefix
/mnt/lustre3p/users/sforde/capensis/06_snps/capensis_capensis.diff --min-count 3 --min-coverage 20 --max-coverage
300 4>/mnt/lustre3p/users/sforde/capensis/06_snps/popoolation_snps_capensis.stderr

# paradoxus
perl /apps/chpc/bio/popoolation/2_1201/snp-frequency-diff.pl --input
/mnt/lustre3p/users/sforde/paradoxus/05_nuclear_map/G_subsamples/paradoxus_capensis.sync --output-prefix
/mnt/lustre3p/users/sforde/paradoxus/06_snps/paradoxus_capensis.diff --min-count 3 --min-coverage 20 --max-coverage
300 4>/mnt/lustre3p/users/sforde/paradoxus/06_snps/popoolation_snps_paradoxus.stderr

# FILTER FOR POP SNP TYPE ONLY

more /home/rhenriques/Sarah/capensis/capensis_capensis.diff_rc| grep -e "pop">
/home/rhenriques/Sarah/capensis/capensis_capensis.filt.rc

more /home/rhenriques/Sarah/paradoxus/paradoxus_capensis.diff_rc| grep -e "pop">
/home/rhenriques/Sarah/paradoxus/paradoxus_capensis.filt.rc

# FILTER FOR BIALLELIC SNPs

more /home/rhenriques/Sarah/capensis/capensis_capensis.filt.rc| awk '{if($4==2)print $1 "\t" $2 "\t" $10 "\t" $11
"\t" $12 "\t" $13 "\t" $14 "\t" $15 "\t" $16 "\t" $17}' > capensis_biallelic2.txt
cat capensis_biallelic.txt| tr '/' '\t' > capensis_biallelic2.tab

more /home/rhenriques/Sarah/paradoxus/paradoxus_capensis.filt.rc| awk '{if($4==2)print $1 "\t" $2 "\t" $10 "\t" $11
"\t" $12 "\t" $13 "\t" $14 "\t" $15 "\t" $16 "\t" $17 "\t" $18 "\t" $19}' > paradoxus_biallelic.txt
cat paradoxus_biallelic.txt| tr '/' '\t' > paradoxus_biallelic.tab

# OBTIAN TOTAL & PRIVATE SNPs
#       only showing for M. capensis

more capensis_biallelic2.tab|awk '{if ($3!=$4&&$3!=0) print}' > CNN.snps.txt
more capensis_biallelic2.tab|awk '{if ($5!=$6&&$5!=0) print}' > CCN.snps.txt
more capensis_biallelic2.tab|awk '{if ($7!=$8&&$7!=0) print}' > CWC.snps.txt
more capensis_biallelic2.tab|awk '{if ($9!=$10&&$9!=0) print}' > CWC2.snps.txt

more capensis_biallelic2.tab|awk '{if ($3!=$4&&$5!=0&&$7!=0&&$9!=0) print}' > CNN.privatesnps.txt
more capensis_biallelic2.tab|awk '{if ($5!=$6&&$3!=0&&$7!=0&&$9!=0) print}' > CCN.privatesnps.txt
```

```
more capensis_biallelic2.tab|awk '{if ($7!=$8&&$3!=0&&$5!=0&&$9!=0) print}' > CWC.privatesnps.txt
more capensis_biallelic2.tab|awk '{if ($9!=$10&&$3!=0&&$5!=0&&$7!=0) print}' > CWC2.privatesnps.txt

# OBTAIN MAJOR & MINOR ALLELES
#       only showing for M. capensis

more capensis_biallelic2.tab|awk '{print $1 "\t" $2 "\t" $3 "\t" $4 "\t" $5 "\t" $6 "\t" $7 "\t" $8 "\t" $9 "\t"
$10}' > capensis_biallelic2_maa.tab

more capensis_biallelic2.tab|awk '{print $1 "\t" $2 "\t" $11 "\t" $12 "\t" $13 "\t" $14 "\t" $15 "\t" $16 "\t" $17
"\t" $18}' > capensis_biallelic2_mia.tab


#       I) FILTER ORIGINAL SYNC FILES

awk 'NR==FNR{a[$1,$2];next}($1,$2)in a' capensis/01_filter_snps/capensis_biallelic.filt2.tab
capensis/capensis_capensis.sync > capensis/01_filter_snps/capensis_capensis.biallelic2.sync

awk 'NR==FNR{a[$1,$2];next}($1,$2)in a' paradoxus/01_filter_snps/paradoxus_biallelic.filt2.tab
paradoxus/paradoxus_capensis.sync > paradoxus/01_filter_snps/paradoxus_capensis.biallelic.sync


# DIVERSITY & POPULATION STRUCTURE ##############################################################################

#       A) POOL & SPECIES PILEUP

module load app/samtools/0.1.18
module load app/bwa/0.7.13
module load app/PoPoolation
module load perl/5.28.0

# index reference genome: ref_capensis.fna
bwa index ref_capensis.fna

# CAPENSIS POOLS
samtools mpileup -d 1000 -Q 20 -B -f ref_capensis.fna capensis/CNN_merged_subsample.bam >
capensis/CORRECT/02_pileup/CNN.pileup
samtools mpileup -d 1000 -Q 20 -B -f ref_capensis.fna capensis/CCN_merged_subsample.bam >
capensis/CORRECT/02_pileup/CCN.pileup
```

```
samtools mpileup -d 1000 -Q 20 -B -f ref_capensis.fna capensis/CWC_merged_subsample.bam >
capensis/CORRECT/02_pileup/CWC.pileup
samtools mpileup -d 1000 -Q 20 -B -f ref_capensis.fna capensis/CWC2_merged_subsample.bam >
capensis/CORRECT/02_pileup/CWC2.pileup

# PARADOXUS POOLS
samtools mpileup -d 1000 -Q 20 -B -f ref_capensis.fna paradoxus/PNN_merged_subsample.bam >
paradoxus/CORRECT/02_pileup/PNN.pileup
samtools mpileup -d 1000 -Q 20 -B -f ref_capensis.fna paradoxus/PCN_merged_subsample.bam >
paradoxus/CORRECT/02_pileup/PCN.pileup
samtools mpileup -d 1000 -Q 20 -B -f ref_capensis.fna paradoxus/POR_merged_subsample.bam >
paradoxus/CORRECT/02_pileup/POR.pileup
samtools mpileup -d 1000 -Q 20 -B -f ref_capensis.fna paradoxus/PSW_merged_subsample.bam >
paradoxus/CORRECT/02_pileup/PSW.pileup
samtools mpileup -d 1000 -Q 20 -B -f ref_capensis.fna paradoxus/PWC_merged_subsample.bam >
paradoxus/CORRECT/02_pileup/PWC.pileup

# CAPENSIS SPECIES
samtools mpileup -d 1000 -Q 20 -B -f ref_capensis.fna -l AFTER_LANE_EFFECT/capensis_top5_removed
capensis/CNN_merged_subsample.bam capensis/CCN_merged_subsample.bam capensis/CWC_merged_subsample.bam
capensis/CWC2_merged_subsample.bam > AFTER_LANE_EFFECT/capensis.pileup

# PARADOXUS SPECIES
samtools mpileup -d 1000 -Q 20 -B -f ref_capensis.fna -l AFTER_LANE_EFFECT/paradoxus_top10_removed
paradoxus/PNN_merged_subsample.bam paradoxus/PCN_merged_subsample.bam paradoxus/POR_merged_subsample.bam
paradoxus/PSW_merged_subsample.bam paradoxus/PWC_merged_subsample.bam > AFTER_LANE_EFFECT/paradoxus.pileup

#       C) POOL PILEUP TO SYNC
#               only showing example of one M. capensis pool

java -ea -Xmx40g -jar /home/.apps/PoPoolation/2.svn204/mpileup2sync.jar --input
/home/rhenriques/Sarah/capensis/CORRECT/02_pileup/CNN.pileup --output
/home/rhenriques/Sarah/capensis/CORRECT/02_pileup/CNN.sync --fastq-type sanger --min-qual 20 --threads 30 2>
/home/rhenriques/Sarah/capensis/CORRECT/02_pileup/CNN.stderr


#       C) POOL DIVERSITY MEASURES
#               only showing example of one M. capensis pool
```

```
# THETA

perl /apps/PoPoolation/1.2.2/Variance-sliding.pl --fastq-type sanger --measure theta --input
/home/rhenriques/Sarah/capensis/CORRECT/02_pileup/CNN.pileup --min-count 2 --min-coverage 20 --max-coverage 300 --
min-qual 20 --pool-size 70 --window-size 500 --step-size 500 --output
/home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/CNN.theta >
/home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/output.CNN.theta
more /home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/CNN.theta|awk '{if($5!="na"&&$3!="0")print}' >
/home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/CNN.theta.list.txt
more /home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/CNN.theta.list.txt|awk '{ total += $5 } END {
print total/NR }' > /home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/CNN.theta.txt

# PI

perl /apps/PoPoolation/1.2.2/Variance-sliding.pl --fastq-type sanger --measure pi --input
/home/rhenriques/Sarah/capensis/CORRECT/02_pileup/CNN.pileup --min-count 2 --min-coverage 20 --max-coverage 300 --
min-qual 20 --pool-size 70 --window-size 500 --step-size 500 --output
/home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/CNN.pi >
/home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/output.CNN.pi
more /home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/CNN.pi|awk '{if($5!="na"&&$3!="0")print}' >
/home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/CNN.pi.list.txt
more /home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/CNN.pi.list.txt|awk '{ total += $5 } END { print
total/NR }' > /home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/CNN.pi.txt


# D

perl /apps/PoPoolation/1.2.2/Variance-sliding.pl --fastq-type sanger --measure D --input
/home/rhenriques/Sarah/capensis/02_Dleup/CNN.Dleup --min-count 2 --min-coverage 20 --max-coverage 300 --min-qual 20
--pool-size 70 --window-size 500 --step-size 500 --output
/home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/CNN.D >
/home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/output.CNN.D
more /home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/CNN.D|awk '{if($5!="na"&&$3!="0")print}' >
/home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/CNN.D.list.txt
more /home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/CNN.D.list.txt|awk '{ total += $5 } END { print
total/NR }' > /home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/CNN.D.txt


#       C) SPECIES DIVERSITY MEASURES
```

```
#              only showing M. capensis

perl /apps/PoPoolation/1.2.2/Variance-sliding.pl --fastq-type sanger --measure theta --input
/home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/capensis_capensis.pileup --min-count 2 --min-coverage
20 --max-coverage 300 --min-qual 20 --pool-size 70 --window-size 500 --step-size 500 --output
/home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/capensis_capensis.theta >
/home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/output.capensis_capensis.theta

more /home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/capensis_capensis.theta|awk
'{if($5!="na"&&$3!="0")print}' >
/home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/capensis_capensis.theta.list.txt

more /home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/capensis_capensis.theta.list.txt|awk '{ total +=
$5 } END { print total/NR }' >
/home/rhenriques/Sarah/capensis/CORRECT/04_diversity_measure/capensis_capensis.theta.txt


#      D) PRINCIPAL COMPONENT ANALYSIS OF ENTIRE DATASET
#              performed in RStudio
#              only shown for M. capensis


library(dplyr)
library(ade4)
library(FactoMineR)
library(factoextra)
library(gridExtra)
library(utils)
library(ggpubr)
library(grid)

#import maa dataset
capensis.biallelic.maa <- read.delim("capensis_biallelic2_maa.tab", header=FALSE)

#create new table dividing the columns to obtain the maa frequencies
capensis.maa.freq <- mutate(capensis.biallelic.maa, CNN = V3/V4, CCN = V5/V6, CWC = V7/V8, CWC2 = V9/V10)

#create a new table where Variable 1 is Contig + Position, and adding the freq per pool as independent columns
c.maa.freq <- data.frame(Var1= c(with(capensis.maa.freq, paste(V1,V2))))
```

```
c.maa.freq$CNN=capensis.maa.freq$CNN
c.maa.freq$CCN=capensis.maa.freq$CCN
c.maa.freq$CWC=capensis.maa.freq$CWC
c.maa.freq$CWC2=capensis.maa.freq$CWC2

#make Var 1 the row names and remove Var1
rownames(c.maa.freq) <- c.maa.freq$Var1
c.maa.freq3 <- c.maa.freq[,-1]
View(c.maa.freq)

#transpose matrix, so SNPs are variables and pools are individuals
c.maa.freq2 <- t(c.maa.freq3)
nrow(c.maa.freq2)
ncol(c.maa.freq2)

#final dataset : c.maa.freq.2

########################################   PCA using the entire dataset ######################################

pca.capensis <- PCA(c.maa.freq2, scale.unit = FALSE)
pca.capensis

before_LE_cap <- fviz_pca_ind(pca.capensis, axes = c(1,2), geom = c("point", "text"), col.ind = "maroon", title =
"A") +border()

fviz_pca_ind(pca.capensis, axes = c(1,3), geom = c("point", "text"))
#fviz_pca_ind(pca.capensis, axes = c(1,4), geom = c("point", "text"))
fviz_pca_ind(pca.capensis, axes = c(2,3), geom = c("point", "text"))

fviz_pca_var(pca.capensis, axes = c(1,2), geom = c("point", "text"))
fviz_pca_var(pca.capensis, axes = c(1,2), geom = c("point"))
fviz_pca_var(pca.capensis, axes = c(1,3), geom = c("point", "text"))
fviz_pca_var(pca.capensis, axes = c(1,3), geom = c("point"))
fviz_pca_var(pca.capensis, axes = c(2,3), geom = c("point", "text"))
fviz_pca_var(pca.capensis, axes = c(2,3), geom = c("point"))

#create a new data frame that has the contribution of each SNP to the PCA
plot.data <- as.data.frame(pca.capensis$var$contrib)
rownames(plot.data) <- c.maa.freq$Var1
```

```
#add two extra columns to include contig and position ID - to extract the ones that are contributing the most
plot.data$CHR=capensis.biallelic.maa$V1
plot.data$BP=capensis.biallelic.maa$V2
View(plot.data)

top1 <- quantile(plot.data$Dim.1, probs = 0.99)
top1.3 <- quantile(plot.data$Dim.1, probs = 0.99)

top10 <- quantile(plot.data$Dim.1, probs = 0.90)
top5 <- quantile(plot.data$Dim.1, probs = 0.95)

library(openxlsx)

#Identify SNPS that are causing differentiation
top1.pc1 <- filter(plot.data, Dim.1 >= top1)
top1_pc1 <- write.csv(top1.pc1, "Top1_PC1_Capensis.csv")
View(top1.pc1)
write.xlsx(top1.pc1, "Top1_PC1_Capensis.xlsx")

top1.pc3 <- filter(plot.data, Dim.3 >= top1.3)
top1_pc3 <- write.csv(top1.pc3, "Top1_PC3_Capensis.csv")
View(top1.pc3)

top5.pc1 <- filter(plot.data, Dim.1 >= top5)
top5_pc1 <- write.csv(top5.pc1, "Top5_PC1_Capensis.csv")
write.xlsx(top5.pc1, "Top5_PC1_Capensis.xlsx")
View(top5.pc1)

top10.pc1 <- filter(plot.data, Dim.1 >= top10)
top10_pc1 <- write.csv(top2.pc1, "Top10_PC1_Capensis.csv")
write.xlsx(top10.pc1, "Top10_PC1_Capensis.xlsx")
View(top10.pc1)


#       E) ACCOUNT FOR BATCH EFFECT
#               performed in RStudio
#               only shown for M. capensis

library(dplyr)
```

```
library(openxlsx)

#import maa dataset
capensis.biallelic.maa <- read.delim("capensis_biallelic2_maa.tab", header=FALSE)
capensis.biallelic.maa <- capensis.biallelic.maa %>% rename(CTG = "V1", BP = "V2")
View(capensis.biallelic.maa)

#import % SNPs
top5_capensis <- read_excel("Top5_PC1_Capensis.xlsx", col_types = c("skip", "skip", "skip", "text", "numeric"))
top5_capensis <- top5_capensis %>% rename(CTG = "CHR")
View(top5_capensis)

top10_capensis <- read_excel("top10_PC1_Capensis.xlsx", col_types = c("skip", "skip", "skip", "text", "numeric"))
top10_capensis <- top10_capensis %>% rename(CTG = "CHR")
View(top10_capensis)

# merge columns 1 and 2 in one single column
CTG_BP <-cbind(paste(capensis.biallelic.maa$CTG,capensis.biallelic.maa$BP,sep = "_"))
capensis.biallelic.maa <- cbind(CTG_BP, capensis.biallelic.maa)
View(capensis.biallelic.maa)
write.xlsx(capensis.biallelic.maa, "capensis.biallelic.maa.xlsx")

CTG_BP <-cbind(paste(top5_capensis$CTG,top5_capensis$BP,sep = "_"))
top5_capensis <- cbind(CTG_BP, top5_capensis)
View(top5_capensis)
write.xlsx(top5_capensis, "capensis_top5.xlsx")

CTG_BP <-cbind(paste(top10_capensis$CTG,top10_capensis$BP,sep = "_"))
top10_capensis <- cbind(CTG_BP, top10_capensis)
View(top10_capensis)
write.xlsx(top10_capensis, "capensis_top10.xlsx")

# remove data from capensis.biallelic.maa that is found in top5_capensis (remove the SNPs causing differentiation)
#removed = (capensis.biallelic.maa) - (capensis_top5)

capensis_top5_removed <- capensis.biallelic.maa %>% anti_join(top5_capensis, by = "CTG_BP")
View(capensis_top5_removed)
write.xlsx(capensis_top5_removed, "capensis_top5_removed.xlsx")
```

```
capensis_top10_removed <- capensis.biallelic.maa %>% anti_join(top10_capensis, by = "CTG_BP")
View(capensis_top10_removed)
write.xlsx(capensis_top10_removed, "capensis_top10_removed.xlsx")


#       F) PRINCIPAL COMPONENT ANALYSIS OF DATASET ACCOUNTING FOR OBSERVED LANE EFFECT
#               performed in RStudio
#               only shown for M. capensis


library(dplyr)
library(ade4)
library(FactoMineR)
library(factoextra)
library(gridExtra)
library(utils)
library(openxlsx)

#import maa dataset without top % SNPs
capensis_top5_removed <- read_excel("capensis_top5_removed.xlsx", col_types = c("text", "text", "numeric",
"numeric", "numeric", "numeric", "numeric", "numeric", "numeric", "numeric"))
View(capensis_top5_removed)

capensis_top10_removed <- read_excel("capensis_top10_removed.xlsx", col_types = c("text", "text", "numeric",
"numeric", "numeric", "numeric", "numeric", "numeric", "numeric", "numeric"))
View(capensis_top10_removed)

#create new table dividing the columns to obtain the maa frequencies
capensis_top5_maafreq <- mutate(capensis_top5_removed, CNN = V3/V4, CCN = V5/V6, CWC = V7/V8, CWC2 = V9/V10)
View(capensis_top5_maafreq)
capensis_top10_maafreq <- mutate(capensis_top10_removed, CNN = V3/V4, CCN = V5/V6, CWC = V7/V8, CWC2 = V9/V10)
View(capensis_top10_maafreq)

write.xlsx(capensis_top5_maafreq, file = "capensis_top5_maafreq.xlsx")
write.xlsx(capensis_top10_maafreq, file = "capensis_top10_maafreq.xlsx")

#create a new table where Variable 1 is Contig + Position, and adding the freq per pool as independent columns
c.top5.maafreq <- data.frame(Var1= c(with(capensis_top5_maafreq, paste(CTG_BP))))
c.top5.maafreq$CNN=capensis_top5_maafreq$CNN
```

```
c.top5.maafreq$CCN=capensis_top5_maafreq$CCN
c.top5.maafreq$CWC=capensis_top5_maafreq$CWC
c.top5.maafreq$CWC2=capensis_top5_maafreq$CWC2
View(c.top5.maafreq)

c.top10.maafreq <- data.frame(Var1= c(with(capensis_top10_maafreq, paste(CTG_BP))))
c.top10.maafreq$CNN=capensis_top10_maafreq$CNN
c.top10.maafreq$CCN=capensis_top10_maafreq$CCN
c.top10.maafreq$CWC=capensis_top10_maafreq$CWC
c.top10.maafreq$CWC2=capensis_top10_maafreq$CWC2
View(c.top10.maafreq)

#make Var 1 the row names and remove Var1
rownames(c.top5.maafreq) <- c.top5.maafreq$Var1
c.top5.maafreq3 <- c.top5.maafreq[,-1]
View(c.top5.maafreq)

rownames(c.top10.maafreq) <- c.top10.maafreq$Var1
c.top10.maafreq3 <- c.top10.maafreq[,-1]
View(c.top10.maafreq)

#transpose matrix, so SNPs are variables and pools are individuals
c.top5.maafreq2 <- t(c.top5.maafreq3)
nrow(c.top5.maafreq2)
ncol(c.top5.maafreq2)

c.top10.maafreq2 <- t(c.top10.maafreq3)
nrow(c.top10.maafreq2)
ncol(c.top10.maafreq2)

#final dataset : c.maa.freq.2

########################################   PCA using the NON-LE dataset ########################################

pca.capensis5 <- PCA(c.top5.maafreq2, scale.unit = FALSE)

after_LE_cap <- fviz_pca_ind(pca.capensis5, axes = c(1,2), geom = c("point", "text"), col.ind = "maroon", title =
"C") +border()
after_LE_cap
```

```
fviz_pca_var(pca.capensis5, axes = c(1,2), geom = c("point", "text"))
fviz_pca_var(pca.capensis5, axes = c(1,2), geom = c("point"))

pca.capensis10 <- PCA(c.top10.maafreq2, scale.unit = FALSE)

fviz_pca_ind(pca.capensis10, axes = c(1,2), geom = c("point", "text"))

fviz_pca_var(pca.capensis10, axes = c(1,2), geom = c("point", "text"))
fviz_pca_var(pca.capensis10, axes = c(1,2), geom = c("point"))


#       G) SNP-SPECIFIC FST

module load app/PoPoolation
module load perl/5.28.0

# species

/apps/PoPoolation/2.svn204/fst-sliding.pl --input
/home/rhenriques/Sarah/capensis/01_filter_snps/capensis_capensis.biallelic2.sync --output
capensis_capensis.biallelic2.fst --min-count 3 --min-coverage 10 --max-coverage 300 --pool-size 70 --window-size 1 -
-step-size 1 --suppress-noninformative  2>
/home/rhenriques/Sarah/comparative/popbi_comparative/Cspecies_slidingfst.stderr

/apps/PoPoolation/2.svn204/fst-sliding.pl --input
/home/rhenriques/Sarah/paradoxus/01_filter_snps/paradoxus_capensis.biallelic.sync --output
paradoxus_capensis.biallelic.fst --min-count 3 --min-coverage 10 --max-coverage 300 --pool-size 70 --window-size 1 -
-step-size 1 --suppress-noninformative  2>
/home/rhenriques/Sarah/comparative/popbi_comparative/Pspecies_slidingfst.stderr

# comparative

/apps/PoPoolation/2.svn204/fst-sliding.pl --input capensis_paradoxus.biallelic_10_300.sync --output
capensis_paradoxus.biallelic_10_300.fst --min-count 3 --min-coverage 10 --max-coverage 300 --pool-size 70 --window-
size 1 --step-size 1 --suppress-noninformative  2>
/home/rhenriques/Sarah/comparative/popbi_comparative/2_slidingfst.stderr

# saved as: Species sliding fst (FST plots).xlsx and capensis_paradoxus.biallelic_10_300_fst.xlsx
```

```
#       H) VISUALISING FST
#               performed in RStudio

library(ggplot2)
library(tidyverse)
library(dplyr)
library(gridExtra)
library(readxl)

# 1. entire dataset

capensis_FST1 <- read_excel("C:/Users/sarah/OneDrive/Desktop/HONOURS 2021/GTK 703/Hons Research/Honours Research
Project 2021/11_comparative/FST plots/Species sliding fst (FST plots).xlsx", sheet = "capensis", col_types =
c("text", "numeric", "numeric", "numeric", "numeric", "numeric", "numeric", "numeric", "numeric", "numeric",
"numeric"))
View(capensis_FST1)

paradoxus_FST1 <- read_excel("C:/Users/sarah/OneDrive/Desktop/HONOURS 2021/GTK 703/Hons Research/Honours Research
Project 2021/11_comparative/FST plots/Species sliding fst (FST plots).xlsx", sheet = "paradoxus", col_types =
c("text", "numeric", "numeric", "numeric", "numeric", "numeric", "numeric", "numeric", "numeric", "numeric",
"numeric", "numeric", "numeric", "numeric", "numeric"))
View(paradoxus_FST1)


ggplot(data = capensis_FST1, aes(y=cnn_cwc2, x = CTG)) +
  geom_point(color = "maroon", size = 1) +
  coord_cartesian(ylim=c(0,0.7)) +
  xlab("SNP") + ylab("FST") + ggtitle("Merluccius capensis FST (including lane effect SNPs)") +
  theme_bw() +
  theme(axis.text.x = element_blank())


ggplot(data = paradoxus_FST1, aes(y=pnn_psw, x = CTG)) +
  geom_point(color = "forestgreen", size = 1) +
  coord_cartesian(ylim=c(0,0.7)) +
  xlab("SNP") + ylab("FST") + ggtitle("Merluccius paradoxus FST (including lane effect SNPs)") +
  theme_bw() +
```

```
  theme(axis.text.x = element_blank())


# 2. accounting for lane effect
#       performed in cluster


module load app/PoPoolation
module load perl/5.28.0

#capensis

cat "/home/rhenriques/Sarah/AFTER_LANE_EFFECT/Top5_PC1_Capensis.csv" | tr ',' '\t' > top5_capensis.tab
awk 'NR==FNR{a[$1,$2];next} !($1,$2)in a' top5_capensis.tab capensis_capensis.biallelic2.sync >
capensis_topSNPremoved.sync

/apps/PoPoolation/2.svn204/fst-sliding.pl --input
/home/rhenriques/Sarah/AFTER_LANE_EFFECT/capensis_topSNPremoved.sync --output capensis_topSNPremoved.fst --min-count
3 --min-coverage 10 --max-coverage 300 --pool-size 70 --window-size 1 --step-size 1 --suppress-noninformative  2>
capensis_topSNPremoved_slidingfst.stderr

#paradoxus

cat "/home/rhenriques/Sarah/AFTER_LANE_EFFECT/Top10_PC1_Paradoxus.csv" | tr ',' '\t' > top10_paradoxus.tab
awk 'NR==FNR{a[$1,$2];next} !($1,$2)in a' top10_paradoxus.tab paradoxus_capensis.biallelic.sync >
paradoxus_topSNPremoved.sync

/apps/PoPoolation/2.svn204/fst-sliding.pl --input
/home/rhenriques/Sarah/AFTER_LANE_EFFECT/paradoxus_topSNPremoved.sync --output paradoxus_topSNPremoved.fst --min-
count 3 --min-coverage 10 --max-coverage 300 --pool-size 70 --window-size 1 --step-size 1 --suppress-noninformative
2> paradoxus_topSNPremoved_slidingfst.stder


# 3. non-LE dataset

capensis_FST <- read_excel("topSNPremoved FST plots.xlsx", sheet = "capensis", col_types = c("text", "numeric",
"numeric", "numeric", "numeric", "numeric", "numeric", "numeric", "numeric", "numeric", "numeric"))
View(capensis_FST)
```

```
paradoxus_FST <- read_excel("topSNPremoved FST plots.xlsx", sheet = "paradoxus", col_types = c("text", "numeric",
"numeric", "numeric", "numeric", "numeric", "numeric", "numeric", "numeric", "numeric", "numeric", "numeric",
"numeric", "numeric", "numeric"))
View(paradoxus_FST)


ggplot(data = capensis_FST, aes(y=cnn_cwc2, x = CTG)) +
  geom_point(color = "maroon", size = 1) +
  coord_cartesian(ylim=c(0,0.7)) +
  xlab("SNP") + ylab("FST") + ggtitle("Merluccius capensis FST (excluding lane effect SNPs)") +
  theme_bw() +
  theme(axis.text.x = element_blank())

ggplot(data = paradoxus_FST, aes(y=pnn_psw, x = CTG)) +
  geom_point(color = "forestgreen", size = 1) +
  coord_cartesian(ylim=c(0,0.7)) +
  xlab("SNP") + ylab("FST") + ggtitle("Merluccius paradoxus FST (excluding lane effect SNPs)") +
  theme_bw() +




# ENVIRONMENTAL ASSOCIATION ANALYSES ##############################################################################


#       A) BAYPASS


#### LIBRARIES ####

library(dplyr)
library(poolfstat)
library(magrittr)
library(reshape)
library(tibble)
library(openxlsx)
library(readxl)
library(mvtnorm)
library(sp)
```

```
library(raster)
library(geigen)
library(ade4)


#####################################################################################################
#                                    FILE & SNP DATA PREPARATION                                    #
#####################################################################################################


#### CONVERT SYNC TO POOLDATA ####
# Input files needed = filtered/biallelic sync files after lane effect was accounted for
# get pooldata from sync file
capensis.pooldata=popsync2pooldata(sync.file="capensis_biallelic_ale.sync",poolsizes=c(74,70,80,40),
poolnames = c("CNN","CCN","CWC","CWC2"),
min.rc = 3,
min.cov.per.pool = 3,
max.cov.per.pool = 300,
min.maf = 0.05)


# get the genobaypass file and the overall allele count file
pooldata2genobaypass(capensis.pooldata, writing.dir = getwd(), subsamplesize = -1)
capensis_countdata=genobaypass2countdata(genobaypass.file="capensis_1%maf_genobaypass")


capensis_pooldata_pFST <- compute.pairwiseFST(capensis.pooldata,
                                              method = "Anova",
                                              min.cov.per.pool = 3,
                                              max.cov.per.pool = 300,
                                              min.indgeno.per.pop = -1,
                                              min.maf = 0.01,
                                              output.snp.values = TRUE,
                                              nsnp.per.bjack.block = 2,
                                              verbose = TRUE)

capensis_pooldata_FST <- computeFST(capensis.pooldata,
                                    method = "Anova",
                                    nsnp.per.bjack.block = 2,
```

```
                                        sliding.window.size = 1,
                                        verbose = TRUE)


paradoxus.pooldata=popsync2pooldata(sync.file="paradoxus_biallelic_ale.sync",poolsizes=c(80,70,72,74,76),
                                    poolnames = c("PNN","PCN","PWC","PW2C","PSW"),
                                    min.rc = 3,
                                    min.cov.per.pool = 3,
                                    max.cov.per.pool = 300,
                                    min.maf = 0.05)

# get the genobaypass file and the overall allele count file
pooldata2genobaypass(paradoxus.pooldata, writing.dir = getwd(), subsamplesize = -1)
paradoxus_countdata=genobaypass2countdata(genobaypass.file="paradoxus_genobaypass")


# calculate FST
paradoxus_pooldata_pFST <- compute.pairwiseFST(paradoxus.pooldata,
                                               method = "Anova",
                                               min.cov.per.pool = 3,
                                               max.cov.per.pool = 300,
                                               min.indgeno.per.pop = -1,
                                               min.maf = 0.05,
                                               output.snp.values = TRUE,
                                               nsnp.per.bjack.block = 2,
                                               verbose = TRUE)

paradoxus_pooldata_FST <- computeFST(paradoxus.pooldata,
                                               method = "Anova",
                                               nsnp.per.bjack.block = 2,
                                               sliding.window.size = 0,
                                               verbose = TRUE)



#########################################################################################################
#           ENVIRONMENTAL ASSOCIATION ANALYSES: POPULATION DIFFERENTIATION                              #
#########################################################################################################
```

```
#### ASSESS POPULATION DIFFERENTIATION ####

source("baypass_utils.R")
require(corrplot) ; require(ape)

# Input files needed: omega matrix [_mat_omega.out output from BayPass (standard core model) in hpc]

C.omega = as.matrix(read.table("C.BP_mat_omega.out"))
colnames(C.omega) <- c("CNN", "CCN", "CWC", "CWC2")
rownames(C.omega) <- c("CNN", "CCN", "CWC", "CWC2")

P.omega = as.matrix(read.table("P.BP_mat_omega.out"))
colnames(P.omega) <- c("PNN", "PCN", "POR", "PWC", "PSW")
rownames(P.omega) <- c("PNN", "PCN", "POR", "PWC", "PSW")

# Create a correlation matrix of the omega values (used to assess genomic differentiation between pools)
C.cor.mat=cov2cor(C.omega)
corrplot(C.cor.mat, method="color", mar=c(2,1,2,2)+0.1,
         main=expression("M. capensis: Correlation map based on"~hat(Omega)))

P.cor.mat=cov2cor(P.omega)
corrplot(P.cor.mat, method="color", mar=c(2,1,2,2)+0.1,
         main=expression("M. paradoxus: Correlation map based on"~hat(Omega)))


# Create hierarchical clustering tree (population differentiation)
C.bta14.tree=as.phylo(hclust(as.dist(1-C.cor.mat**2)))
plot(C.bta14.tree,type="p",
     main=expression("M. capensis: hier. clust. tree based on"~hat(Omega)~"("*d[ij]*"=1-"*rho[ij]*")"))

P.bta14.tree=as.phylo(hclust(as.dist(1-P.cor.mat**2)))
plot(P.bta14.tree,type="p",
     main=expression("M. paradoxus: hier. clust. tree based on"~hat(Omega)~"("*d[ij]*"=1-"*rho[ij]*")"))


#############################################################################################
#                      ENVIRONMENTAL ASSOCIATION ANALYSES: ENVIROMENTAL DATA                #
#############################################################################################
```

```
#### BIO-ORACLE DATA ####

library(sdmpredictors)
library(leaflet)

# List layers available in Bio-ORACLE v2
layers.bio2 <- list_layers( datasets="Bio-ORACLE" )
layers.bio2
View(layers.bio2)

# Download environmental data layers (at average depth) & bathymetry
# options(sdmpredictors_datadir="C:\\Users\\sarah\\AppData\\Local\\Temp\\Rtmp65G0Ua/sdmpredictors")

env.av.depth <- load_layers(layercodes = c("BO_chlomean", "BO2_curvelmean_bdmean", "BO2_dissoxmean_bdmean",
"BO2_tempmean_bdmean", "BO2_ppmean_bdmean", "BO2_salinitymean_bdmean", "BO2_tempmean_ss"),
datadir="C:\\Users\\sarah\\AppData\\Local\\Temp\\Rtmp65G0Ua/sdmpredictors", equalarea=FALSE, rasterstack=TRUE)
bathymetry <- load_layers("BO_bathymean", equalarea=FALSE, rasterstack=TRUE)

# Generate a data.frame with the sites of interest

C.sites <- data.frame(Name=c("Northern Namibia" , "Central Namibia" , "West Coast, South Africa" , "West Coast 2,
South Africa") , Lon=c(11.416666, 14.916666, 15.450833, 16.993333) , Lat=c(-18, -26, -30.1841667, -32.811666))
View(C.sites)

P.sites <- data.frame(Name=c("Northern Namibia" , "Central Namibia" , "Orange River, Namibia" , "West Coast, South
Africa", "Southwest Coast, South Africa") , Lon=c(11.383333, 13.51666, 15.08, 16.993333, 18.711666) , Lat=c(-19, -
24.6666667, -29.97, -32.811666, -35.27))
View(P.sites)


# Visualise sites of interest in google maps
c.m <- leaflet()
c.m <- addTiles(c.m)
c.m <- addMarkers(c.m, lng=C.sites$Lon, lat=C.sites$Lat, popup=C.sites$Name)
c.m

p.m <- leaflet()
p.m <- addTiles(p.m)
```

```
p.m <- addMarkers(p.m, lng=P.sites$Lon, lat=P.sites$Lat, popup=P.sites$Name)
p.m

# Extract environmental values from layers for sites of interest
# envdata in Erica's SDM tutorial

C.sites.environment <- data.frame(Name=C.sites$Name , depth=extract(bathymetry,C.sites[,2:3]) ,
extract(env.av.depth,C.sites[,2:3]) )
View(C.sites.environment)

C.env <- t(C.sites.environment)
colnames(C.env) <- as.character(unlist(C.env[1,]))
C.env = C.env[-1, ]
View(C.env)

P.sites.environment <- data.frame(Name=P.sites$Name , depth=extract(bathymetry,P.sites[,2:3]) ,
extract(env.av.depth,P.sites[,2:3]) )
View(P.sites.environment)

P.env <- t(P.sites.environment)
colnames(P.env) <- as.character(unlist(P.env[1,]))
P.env = P.env[-1, ]
View(P.env)

# Export environmental data
write.xlsx(C.env, "capensis.env.data.xlsx", col.names = TRUE, row.names = TRUE)
write.xlsx(P.env, "paradoxus.env.data.xlsx", col.names = TRUE, row.names = TRUE)


#### FILTER ENVIRONMENTAL VARIABLES TO ACCOUNT FOR COLLINEARITY ####

# Run PCA and plot correlation circle
# The `dudi.pca` function allows us to perform the PCA over the whole study area.
# We decide to keep only 2 principal component axes to summarize the whole environmental niche.
# NB: here be careful that none of your xy presence points are outside of the environmental raster layer,
# which will give NA values, which will give errors.

C.envdata <- C.sites.environment[,-1,]
View(C.envdata)
```

```
C.pca1 <- dudi.pca(C.envdata, scannf = F, nf = 2)
round(C.pca1$eig/sum(C.pca1$eig)*100, 2)

P.envdata <- P.sites.environment[,-1,]
View(P.envdata)
P.pca1 <- dudi.pca(P.envdata, scannf = F, nf = 2)
round(P.pca1$eig/sum(P.pca1$eig)*100, 2)

# Plot the PCA to look for potential outliers in the environmental data.
plot(C.pca1$li[,1:2]) # PCA scores on first two axes
summary(C.pca1$li)

plot(P.pca1$li[,1:2])
summary(P.pca1$li)

# Plot a correlation circle.
s.corcircle(C.pca1$co)
s.corcircle(P.pca1$co)

#### ELIMINATING CORRELATED VARIABLES ####


#   1. different quadrants (if same/overlapping, pick longest arrow = varies more over the environmental space)
#   2. know which environmental features are more relevant for study species though

#   Note on arrow directions:
#   - same direction = highly correlated
#   - orthogonal (this means at a 90 degree angle) = unrelated
#   - opposite directions = negatively correlated

# Replot without eliminated variables

C.select.var2 <- subset(C.envdata, select = c("BO_chlomean", "BO_bathymean", "BO2_dissoxmean_bdmean",
"BO2_tempmean_ss"))
View(C.select.var2)
P.select.var2 <- subset(P.envdata, select = c("BO_chlomean", "BO2_curvelmean_bdmean", "BO2_dissoxmean_bdmean",
"BO2_salinitymean_bdmean"))
View(P.select.var2)
```

```
C.pca3 <- dudi.pca(C.select.var2, scannf = F, nf = 2)
s.corcircle(C.pca3$co)

P.pca3 <- dudi.pca(P.select.var2, scannf = F, nf = 2)
s.corcircle(P.pca3$co)


#### ASSESS THE CORRELATION USING VIF ####

# Variance inflation factor (VIF)
# Exclude variables with a VIF>10

library(usdm)

# vifstep calculate VIF for all variables, exclude one with highest VIF (greater than threshold), repeat the
procedure until no variables with VIF greater than th remains

# Using VIF after correlation circle
C.vif2 <- vifstep(C.select.var2,th=10)
View(C.vif2@results)

P.vif2 <- vifstep(P.select.var2,th=10)
View(P.vif2@results)


# Manually create file of this filtered enviro data without headings for cluster: becomes efile for auxillary model
of BayPass

capensis.envdata2 <- subset(C.envdata, select = c("BO2_dissoxmean_bdmean", "BO2_tempmean_ss"))
paradoxus.envdata2 <- subset(P.envdata, select = c("BO_chlomean", "BO2_curvelmean_bdmean", "BO2_dissoxmean_bdmean"))

capensis.envdata2 <- t(capensis.envdata2)
paradoxus.envdata2 <- t(paradoxus.envdata2)

write.xlsx(capensis.envdata2, "capensis.envdata2.xlsx")
write.xlsx(paradoxus.envdata2, "paradoxus.envdata2.xlsx")
```

```
#############################################################################################################
#                        BAYPASS IN HPC: STANDARD COVARIATE CORE MODEL                                      #
#############################################################################################################

module load app/BayPass/2.3.1

g_baypass -npop 5 -gfile C_snp.data -efile capensis.envdata2 -scalecov -poolsizefile C_poolsize -outprefix
capensis_omega
g_baypass -npop 5 -gfile P_snp.data -efile paradoxus.envdata2 -scalecov -poolsizefile P_poolsize -outprefix
paradoxus_omega


##############################################
#                  RDA                       #
##############################################
### LIBRARIES ####

library(vegan)
library(usdm)
library(readr)

### ALLELE FREQUENCIES ####

# Use the allele counts obtained from poolfstat to calculate allele frequencies

# capensis
capensis_allele_counts <- as.data.frame(capensis_countdata@refallele.count)
capensis_total <- as.data.frame(capensis_countdata@total.count)
capensis_all <- cbind.data.frame(capensis_allele_counts,capensis_total)
capensis_all$cnn <- capensis_all$V1/capensis_all$V2
capensis_all$ccn <- capensis_all$V3/capensis_all$V4
capensis_all$cwc <- capensis_all$V5/capensis_all$V6
capensis_all$cwc2 <- capensis_all$V7/capensis_all$V8

pools <- c("cnn","ccn","cwc","cwc2")

capensis_freq <- capensis_all[pools]
write_delim(capensis_freq,"C_AF.txt")
```

```
# paradoxus
paradoxus_allele_counts <- as.data.frame(paradoxus_countdata@refallele.count)
paradoxus_total <- as.data.frame(paradoxus_countdata@total.count)
paradoxus_all <- cbind.data.frame(paradoxus_allele_counts,paradoxus_total)
paradoxus_all$pnn <- paradoxus_all$V1/paradoxus_all$V2
paradoxus_all$pcn <- paradoxus_all$V3/paradoxus_all$V4
paradoxus_all$pwc <- paradoxus_all$V5/paradoxus_all$V6
paradoxus_all$pwc2 <- paradoxus_all$V7/paradoxus_all$V8
paradoxus_all$psw <- paradoxus_all$V9/paradoxus_all$V10

pools <- c("pnn","pcn","pwc","pwc2","psw")

paradoxus_freq <- paradoxus_all[pools]
write_delim(paradoxus_freq,"P_AF.txt")

paradoxus_freq <- t(paradoxus_freq)



# Import allele frequencies

C_AF <- read.table("C_AF.txt", header = T)
View(C_AF)

P_AF <- read.table("P_AF.txt", header = T)
View(P_AF)


#### HELLINGER TRANSFORMATION OF ALLELE FREQUENCIES ####

C.snps <- decostand(C_AF, method="hellinger")
C.snps.t <- t(C.snps)

C.snps.mat <- as.matrix(C.snps.t)
popnames <- rownames(C.snps.mat)
popnames
C.snp.hel <- decostand(C.snps.mat, method = "hellinger")
```

```
C.vars <- read.table("C_env_vars.txt", header = T)
rownames(C.vars)
vif(C.vars)
C.env.scale <- scale(C.vars, scale = T, center = T)


P.snps <- decostand(P_AF, method="hellinger")
P.snps.t <- t(P.snps)

P.snps.mat <- as.matrix(P.snps.t)
popnames <- rownames(P.snps.mat)
popnames
P.snp.hel <- decostand(P.snps.mat, method = "hellinger")

P.vars <- read.table("P_env_vars.txt", header = T)
rownames(P.vars)
vif(P.vars)
P.env.scale <- scale(P.vars, scale = T, center = T)

P.vars2 <- read.table("P_env_vars2.txt", header = T)
rownames(P.vars2)
vif(P.vars2)
P.env.scale2 <- scale(P.vars2, scale = T, center = T)

#### RUN & TEST SIGNIFICANCE ####

C_rda1 <- rda(C.snp.hel ~., data = as.data.frame(C.env.scale))
summary(C_rda1)
RsquareAdj(C_rda1)
anova(C_rda1)
anova(C_rda1, by = "axis")

P_rda1 <- rda(P.snp.hel ~., data = as.data.frame(P.env.scale))
summary(P_rda1)
RsquareAdj(P_rda1)
anova(P_rda1)
anova(P_rda1, by = "axis")
```

```
P_rda2 <- rda(P.snp.hel ~., data = as.data.frame(P.env.scale2))
summary(P_rda2)
RsquareAdj(P_rda2)
anova(P_rda2)
anova(P_rda2, by = "axis")



#### ACCOUNT FOR SEQUENCING LANE IN RDA ####
write.table(P.env.scale2, "P.env.scale2.txt")
P.env.scale3 <- read.table("P.env.scale3.txt")

env.data <- as.data.frame(subset(P.env.scale3, select=c(BO_chlomean,BO2_dissoxmean_bdmean,BO2_salinitymean_bdmean)))
lane.data <- as.data.frame(subset(P.env.scale3, select=c(lane)))

P_rda3 <- rda(P.snp.hel ~ BO_chlomean + BO2_salinitymean_bdmean + Condition(lane), data = P.env.scale3)
summary(P_rda3)
RsquareAdj(P_rda3)
anova(P_rda3, step=1000)
anova(P_rda3, by = "axis")



#### PLOT RDA ####

# PLOT 1

plot(C_rda1, type="n", scaling=3)
points(C_rda1, display="species", pch=21, cex=1, col="gray32", scaling=3)
#points(C_rda1, display="sites", pch=21, cex=1.3, col="gray32", scaling=3)
text(C_rda1, display="sites", pch=21, cex=1.5, col="firebrick3", scaling=3)
text(C_rda1, scaling=3, display="bp", col="#0868ac", cex=1.5)

plot(P_rda2, type="n", scaling=3)
points(P_rda2, display="species", pch=20, cex=1, col="gray32", scaling=3)
text(P_rda2, display="sites", pch=21, cex=1.5, col="firebrick3", scaling=3)
text(P_rda2, scaling=3, display="bp", col="#0868ac", cex=1.5)

# PLOT 2
```

```
plot(C_rda1, type="n", scaling=3, xlim=c(-0.4,0.4), ylim=c(-0.4,0.4))
points(C_rda1, display="species", pch=21, cex=1, col="gray32", scaling=3)
text(C_rda1, display="sites", pch=21, cex=1, col="firebrick3", scaling=3)
text(C_rda1, scaling=3, display="bp", col="#0868ac", cex=1)

plot(P_rda2, type="n", scaling=3,xlim=c(-0.4,0.4), ylim=c(-0.4,0.45))
points(P_rda2, display="species", pch=20, cex=1, col="gray32", scaling=3)
text(P_rda2, display="sites", pch=21, cex=1, col="firebrick3", scaling=3)
text(P_rda2, scaling=3, display="bp", col="#0868ac", cex=1)


#### IDENTIFY OUTLIERS ####

# M. capensis

outliers <- function(x,z){
  lims <- mean(x) + c(-1, 1) * z * sd(x)                    # find +/- z sd from mean loading
  x[x < lims[1] | x > lims[2]]   # locus names in these tails
}

C_load.rda <- summary(C_rda1)$species[,1:3]
C_load.rda[,1]

C_cand1.3SD <- outliers(C_load.rda[,1], 3)
C_cand2.3SD <- outliers(C_load.rda[,2], 3)

View(C_cand1.3SD)
View(C_cand2.3SD)

C_ncand1.3SD <- length(C_cand1.3SD)
C_ncand2.3SD <- length(C_cand2.3SD)
C_ncand1.3SD
C_ncand2.3SD

C_ncand <- C_ncand1.3SD+C_ncand2.3SD
View(C_ncand)
```

```
C_cand1.3SD.df <- cbind.data.frame(rep(1, times = length(C_cand1.3SD)), names(C_cand1.3SD), unname(C_cand1.3SD));
colnames(C_cand1.3SD.df) <- c("axis", "snp", "loading")

C_cand2.3SD.df <- cbind.data.frame(rep(2, times = length(C_cand2.3SD)), names(C_cand2.3SD), unname(C_cand2.3SD));
colnames(C_cand2.3SD.df) <- c("axis", "snp", "loading")

C_cand <- rbind(C_cand1.3SD.df, C_cand2.3SD.df)
C_cand$snp <- as.character(C_cand$snp)

C_cand.mat <- matrix(nrow=(C_ncand), ncol=2)  # ncol = number of predictors
colnames(C_cand.mat) <- c("tempmean_ss", "dissoxmean_bdmean")

View(C_cand)

for (i in 1:length(C_cand$snp)) {
  nam <- C_cand[i,2]
  C_snp.gen <- C.snps.mat[,nam]
  C_cand.mat[i,] <- apply(C.env.scale,2,function(x) cor(x,C_snp.gen))
}

C_full.cand.df <- cbind(C_cand, C_cand.mat)
C_full.cand.df

C_cand$snp[duplicated(C_cand$snp)]  # check for duplicates
C_full.cand.df <- C_full.cand.df[!duplicated(C_full.cand.df$snp),]

View(C_full.cand.df)

for (i in 1:length(C_full.cand.df$snp)) {
  C_bar <- C_full.cand.df[i,]
  C_full.cand.df[i,6] <- names(which.max(abs(C_bar[4:5]))) # the 6 is the column to add to table, 4:5 is the columns
of env predictors
  C_full.cand.df[i,7] <- max(abs(C_bar[4:5]))              # gives the correlation
}

colnames(C_full.cand.df)[6] <- "predictor"
colnames(C_full.cand.df)[7] <- "correlation"

View(C_full.cand.df)
```

```
write.table(C_full.cand.df, file="C_full.cand.df.txt", sep="\t")

table(C_full.cand.df$predictor)
table(C_full.cand.df$axis)



# M. paradoxus

# For "chlomean", "dissoxmean", "salinitymean"

outliers <- function(x,z){
  lims <- mean(x) + c(-1, 1) * z * sd(x)                    # find +/- z sd from mean loading
  x[x < lims[1] | x > lims[2]]   # locus names in these tails
}

P_load.rda2 <- summary(P_rda2)$species[,1:3]
P_load.rda2[,1]
P_load.rda2

P_cand1.3SD2 <- outliers(P_load.rda2[,1], 3)
P_cand2.3SD2 <- outliers(P_load.rda2[,2], 3)

P_ncand1.3SD2 <- length(P_cand1.3SD2)
P_ncand2.3SD2 <- length(P_cand2.3SD2)
P_ncand1.3SD2
P_ncand2.3SD2

P_ncand2 <- P_ncand1.3SD2+P_ncand2.3SD2
P_ncand2

P_cand1.3SD.df2 <- cbind.data.frame(rep(1, times = length(P_cand1.3SD2)), names(P_cand1.3SD2),
unname(P_cand1.3SD2)); colnames(P_cand1.3SD.df2) <- c("axis", "snp", "loading")

P_cand2.3SD.df2 <- cbind.data.frame(rep(2, times = length(P_cand2.3SD2)), names(P_cand2.3SD2),
unname(P_cand2.3SD2)); colnames(P_cand2.3SD.df2) <- c("axis", "snp", "loading")

P_cand2 <- rbind(P_cand1.3SD.df2, P_cand2.3SD.df2)
```

```
P_cand2$snp <- as.character(P_cand2$snp)

P_cand.mat2 <- matrix(nrow=(P_ncand2), ncol=3)  # ncol = number of predictors
colnames(P_cand.mat2) <- c("chlomean", "dissoxmean", "salinitymean")

View(P_cand2)

for (i in 1:length(P_cand2$snp)) {
  nam2 <- P_cand2[i,1.9]
  P_snp.gen2 <- P.snps.mat[,nam2]
  P_cand.mat2[i,] <- apply(P.env.scale2,2,function(x) cor(x,P_snp.gen2))
}


P_full.cand.df2 <- cbind(P_cand2, P_cand.mat2)
P_full.cand.df2

P_cand2$snp[duplicated(P_cand2$snp)]  # check for duplicates
P_full.cand.df2 <- P_full.cand.df2[!duplicated(P_full.cand.df2$snp),]

View(P_full.cand.df2)

for (i in 1:length(P_full.cand.df2$snp)) {
  P_bar2 <- P_full.cand.df2[i,]
  P_full.cand.df2[i,7] <- names(which.max(abs(P_bar2[4:6]))) # the 8 is the column to add to table, 4:6 is the
columns of env predictors
  P_full.cand.df2[i,8] <- max(abs(P_bar2[4:6]))              # gives the correlation
}

colnames(P_full.cand.df2)[7] <- "predictor"
colnames(P_full.cand.df2)[8] <- "correlation"

View(P_full.cand.df2)

write.table(P_full.cand.df2, file="P_full.cand.df2.txt", sep="\t")

table(P_full.cand.df2$predictor)
table(P_full.cand.df2$axis)
```

```
#### PLOTTING OUTLIERS ####

# M. capensis

C_sel <- C_full.cand.df$snp
C_env <- C_full.cand.df$predictor
C_env[C_env=="tempmean_ss"] <- '#00CED1'
C_env[C_env=="dissoxmean_bdmean"] <- '#FF8C00'

C_col.pred <- rownames(C_rda1$CCA$v) # pull the SNP names
View(C_col.pred)

for (i in 1:length(C_sel)) {           # color code candidate SNPs
  C_foo <- match(C_sel[i],C_col.pred)
  C_col.pred[C_foo] <- C_env[i]
}

C_col.pred[grep("OMPL",C_col.pred)] <- '#f1eef6' # non-candidate SNPs
C_empty <- C_col.pred
C_empty[grep("#f1eef6",C_empty)] <- rgb(0,1,0, alpha=0) # transparent
C_empty.outline <- ifelse(C_empty=="#00FF0000","#00FF0000","gray32")
C_bg <- c('#00CED1','#FF8C00', "gray32")

# PLOT 1
plot(C_rda1, type="n", scaling=3, xlim=c(-1,1), ylim=c(-1,1))
points(C_rda1, display="species", pch=21, cex=1.5, col="gray32", bg=C_col.pred, scaling=3)
points(C_rda1, display="species", pch=21, cex=1.5, col=C_empty.outline, bg=C_empty, scaling=3)
text(C_rda1, scaling=3, display="sites", col="black", cex=1.5)
text(C_rda1, scaling=3, display="bp", col="#0868ac", cex=1.5)
legend("bottomleft", legend=c("tempmean_ss","dissoxmean_bdmean"), bty="n", col="gray32", pch=21, cex=1.5,
pt.bg=C_bg)

#PLOT 2
plot(C_rda1, type="n", scaling=3, xlim=c(-0.25,0.25), ylim=c(-0.3,0.25))
points(C_rda1, display="species", pch=21, cex=1.5, col="gray32", bg=C_col.pred, scaling=3)
points(C_rda1, display="species", pch=21, cex=1.5, col=C_empty.outline, bg=C_empty, scaling=3)
```

```
text(C_rda1, scaling=1, display="sites", col="black", cex=1)
text(C_rda1, scaling=3, display="bp", col="#0868ac", cex=1)
legend("bottomleft", legend=c("tempmean_ss","dissoxmean_bdmean"), bty="n", col="gray32", pch=21, cex=1, pt.bg=C_bg)


# M. paradoxus

# For "chlomean", "dissoxmean", "salinitymean"

P_sel2 <- P_full.cand.df2$snp
P_env2 <- P_full.cand.df2$predictor
P_env2[P_env2=="chlomean"] <- '#00CED1'
P_env2[P_env2=="curvelmean"] <- '#FF8C00'
P_env2[P_env2=="salinitymean"] <- '#9932CC'

P_col.pred2 <- rownames(P_rda2$CCA$v) # pull the SNP names
View(P_col.pred2)

for (i in 1:length(P_sel2)) {           # color code candidate SNPs
  P_foo2 <- match(P_sel2[i],P_col.pred2)
  P_col.pred2[P_foo2] <- P_env2[i]
}

P_col.pred2[grep("OMPL",P_col.pred2)] <- '#f1eef6' # non-candidate SNPs
P_empty2 <- P_col.pred2
P_empty2[grep("#f1eef6",P_empty2)] <- rgb(0,1,0, alpha=0) # transparent
P_empty.outline2 <- ifelse(P_empty2=="#00FF0000","#00FF0000","gray32")
P_bg2 <- c('#00CED1','#FF8C00','#9932CC', "gray32")

#PLOT 1
plot(P_rda2, type="n", scaling=3, xlim=c(-1,1), ylim=c(-1,1))
points(P_rda2, display="species", pch=21, cex=1.5, col="gray32", bg=P_col.pred2, scaling=3)
points(P_rda2, display="species", pch=21, cex=1.5, col=P_empty.outline2, bg=P_empty2, scaling=3)
text(P_rda2, scaling=3, display="sites", col="black", cex=1.5)
text(P_rda2, scaling=3, display="bp", col="#0868ac", cex=1.5)
legend("bottomleft", legend=c("chlomean", "dissoxmean", "salinitymean"), bty="n", col="gray32", pch=21, cex=1.5,
pt.bg=P_bg2)

#PLOT 2
```

```
plot(P_rda2, type="n", scaling=3, xlim=c(-0.25,0.15), ylim=c(-0.45,0.45))
points(P_rda2, display="species", pch=21, cex=1.5, col="gray32", bg=P_col.pred2, scaling=3)
points(P_rda2, display="species", pch=21, cex=1.5, col=P_empty.outline2, bg=P_empty2, scaling=3)
text(P_rda2, scaling=3, display="sites", col="black", cex=1)
text(P_rda2, scaling=3, display="bp", col="#0868ac", cex=1)
legend("bottomleft", legend=c("chlomean", "dissoxmean", "salinitymean"), bty="n", col="gray32", pch=21, cex=1,
pt.bg=P_bg2)



#### USE OUTLIER SNPS ONLY ####

C_outliers <- subset(C_cand, select = c("snp"))
View(C_outliers)

P_outliers2 <- subset(P_cand2, select = c("snp"))
View(P_outliers2)

write.xlsx(C_outliers, "C_outliers.xlsx")
write.xlsx(P_outliers2, "P_outliers2.xlsx")

C_outlier_snps <- read_excel("C_outliers.xlsx")
View(C_outlier_snps)
P_outlier_snps2 <- read_excel("P_outliers2.xlsx")
View(P_outlier_snps2)

paradoxus_biallelic_ale <- read_delim("paradoxus_biallelic_ale.sync", "\t", escape_double = FALSE, col_names =
FALSE, trim_ws = TRUE)
View(paradoxus_biallelic_ale)

CTG_BP<-cbind(paste(capensis_biallelic_ale$X1,capensis_biallelic_ale$X2,sep = "_"))
capensis_biallelic_ale <- cbind(CTG_BP, capensis_biallelic_ale)
View(capensis_biallelic_ale)

CTG_BP<-cbind(paste(paradoxus_biallelic_ale$X1,paradoxus_biallelic_ale$X2,sep = "_"))
paradoxus_biallelic_ale <- cbind(CTG_BP, paradoxus_biallelic_ale)
View(paradoxus_biallelic_ale)

C_out.sync <- capensis_biallelic_ale %>% semi_join(C_outlier_snps, by = c("CTG_BP" = "snp"))
```

```
View(C_out.sync)
P_out.sync2 <- paradoxus_biallelic_ale %>% semi_join(P_outlier_snps2, by = c("CTG_BP" = "snp"))
View(P_out.sync2)

write.xlsx(C_out.sync, "capensis_outliers.xlsx")
write.xlsx(P_out.sync2, "paradoxus_outliers2.xlsx")

# Copied _biallelic_ale.sync files, cleared contents and pasted the above .xlsx info
# (manually created sync file containing only outlier SNPs)


#### POOLFSTAT ####

library(poolfstat)

# M. capensis

C_pool_names.list <- c("CNN", "CCN", "CWC", "CWC2")
C_pool_sizes.list <- c(rep(37, 4))


capensis.pooldata <- popsync2pooldata(sync.file = "capensis_outliers.sync", C_poolsizes = C_pool_sizes.list,
                                      poolnames = C_pool_names.list,
                                      min.rc = 3,
                                      min.cov.per.pool = 3,
                                      max.cov.per.pool = 300,
                                      min.maf = 0.01,
                                      nthreads = 10)


capensis.pooldata@snp.info

capensis_pooldata_pFST <- compute.pairwiseFST(capensis.pooldata,
                                              method = "Anova",
                                              min.cov.per.pool = 3,
                                              max.cov.per.pool = 300,
                                              min.indgeno.per.pop = -1,
                                              min.maf = 0.01,
                                              output.snp.values = TRUE,
```

```
                                            nsnp.per.bjack.block = 0,
                                            verbose = TRUE
)


View(capensis_pooldata_pFST@PairwiseFSTmatrix)

capensis.fst <- computeFST(capensis.pooldata, method = "Anova")
View(capensis.fst)

plot(capensis.fst$snp.FST)


# M. paradoxus

P_pool_names.list <- c("PNN", "PCN", "POR", "PWC", "PSW")
P_pool_sizes.list <- c(rep(37, 5))

paradoxus.pooldata2 <- popsync2pooldata(sync.file = "paradoxus_outliers2.sync", poolsizes = P_pool_sizes.list,
                                        poolnames = P_pool_names.list,
                                        min.rc = 3,
                                        min.cov.per.pool = 3,
                                        max.cov.per.pool = 300,
                                        min.maf = 0.01,
                                        nthreads = 10)


paradoxus.pooldata2@snp.info

paradoxus_pooldata_pFST2 <- compute.pairwiseFST(paradoxus.pooldata2,
                                            method = "Anova",
                                            min.cov.per.pool = 3,
                                            max.cov.per.pool = 300,
                                            min.indgeno.per.pop = -1,
                                            min.maf = 0.01,
                                            output.snp.values = TRUE,
                                            nsnp.per.bjack.block = 0,
                                            verbose = TRUE
)
```

```
View(paradoxus_pooldata_pFST2@PairwiseFSTmatrix)

paradoxus.fst2 <- computeFST(paradoxus.pooldata2, method = "Anova")
View(paradoxus.fst2)

plot(paradoxus.fst2$snp.FST)


#### CALCULATE THE FST FOR EACH OUTLIER SNP USING POPOOLATION2 ON THE CLUSTER ####

# MODULES LOADED
module load app/PoPoolation
module load perl/5.28.0

# CAPENSIS
/apps/PoPoolation/2.svn204/fst-sliding.pl --input capensis_outliers.sync --output capensis_outliers_SNPfst --min-
count 3 --min-coverage 10 --max-coverage 300 --pool-size 70 --window-size 1 --step-size 1 --suppress-noninformative
2> capensis_outliers_SNPfst.stderr

# PARADOXUS
/apps/PoPoolation/2.svn204/fst-sliding.pl --input paradoxus_outliers2.sync --output paradoxus_outliers_SNPfst2 --
min-count 3 --min-coverage 10 --max-coverage 300 --pool-size 70 --window-size 1 --step-size 1 --suppress-
noninformative  2> paradoxus_outliers_SNPfst2.stderr

#### PLOT SNP-SPECIFIC FST ####

# Edit the _outliers_SNPfst output files in an .xlsx file so that:
# col1 = CTG, col2 = BP, col3 etc = pairwise FST at each SNP

capensis_outliers_SNPfst <- read_excel("capensis_outliers_SNPfst.xlsx", col_types = c("text", "numeric", "numeric",
"numeric", "numeric", "numeric", "numeric", "numeric"))
View(capensis_outliers_SNPfst)

paradoxus_outliers_SNPfst2 <- read_excel("paradoxus_outliers_SNPfst2.xlsx", col_types = c("text", "numeric",
"numeric", "numeric", "numeric", "numeric", "numeric", "numeric", "numeric", "numeric"))
View(paradoxus_outliers_SNPfst2)
```

```
library(ggplot2)

# only showing plots for one comparison of M. capensis and M. paradoxus

ggplot(data = capensis_outliers_SNPfst, aes(y=cnn_ccn, x = CTG)) +
  geom_point(color = "maroon", size = 1.5) +
  coord_cartesian(ylim=c(0,0.2)) +
  xlab("SNP") + ylab("FST") + ggtitle("M.capensis SNP-specific FST: CNN vs CCN") +
  theme_bw() +
  theme(axis.text.x = element_blank())

ggplot(data = paradoxus_outliers_SNPfst, aes(y=pnn_pcn, x = CTG)) +
  geom_point(color = "forestgreen", size = 1.5) +
  coord_cartesian(ylim=c(0,0.2)) +
  xlab("SNP") + ylab("FST") + ggtitle("M. paradoxus SNP-specific FST: PNN vs PCN") +
  theme_bw() +
  theme(axis.text.x = element_blank())


#### PCA ####

library(FactoMineR)
library(factoextra)
library(gridExtra)
library(utils)
library(ggpubr)
library(grid)

# Import allele frequencies (keep only those for the outlier SNPs)

C_AF2 <- read_excel("C_AF.xlsx", col_types = c("text", "numeric", "numeric", "numeric", "numeric"))
View(C_AF2)
P_AF2 <- read_excel("P_AF.xlsx", col_types = c("text", "numeric", "numeric", "numeric", "numeric", "numeric"))
View(P_AF2)

C_out.AF <- C_AF2 %>% semi_join(C_outlier_snps, by = c("CTG_BP" = "snp"))
View(C_out.AF)
P_out.AF2 <- P_AF2 %>% semi_join(P_outlier_snps2, by = c("CTG_BP" = "snp"))
View(P_out.AF2)
```

```
write.xlsx(C_out.AF, "capensis_outliers_AF.xlsx")
write.xlsx(P_out.AF2, "paradoxus_outliers_AF2.xlsx")

rownames(C_out.AF) <- C_out.AF$CTG_BP
C_out.AF <- C_out.AF[,-1]
View(C_out.AF)

rownames(P_out.AF2) <- P_out.AF2$CTG_BP
P_out.AF2 <- P_out.AF2[,-1]
View(P_out.AF2)

# Plots

C_PCA <- PCA(C_out.AF, scale.unit = FALSE)
capensis_ind_pca <- fviz_pca_ind(C_PCA, axes = c(1,2), geom = c("point", "text"), col.ind = "maroon") +border()
capensis_ind_pca
capensis_pca <- fviz_pca_var(C_PCA, axes = c(1,2), geom = c("point", "text"))
capensis_pca

P_PCA2 <- PCA(P_out.AF2, scale.unit = FALSE)
paradoxus_ind_pca2 <- fviz_pca_ind(P_PCA2, axes = c(1,2), geom = c("point", "text"), col.ind = "forestgreen")
+border()
paradoxus_ind_pca2
paradoxus_pca2 <- fviz_pca_var(P_PCA2, axes = c(1,2), geom = c("point", "text"))
paradoxus_pca2

capensis_pca_grid <- grid.arrange(capensis_ind_pca, capensis_pca, nrow = 1, ncol = 2)
paradoxus_pca_grid2 <- grid.arrange(paradoxus_ind_pca2, paradoxus_pca2, nrow = 1, ncol = 2)
```