Analysis of Data for Round sardines

Packages utilised

vcfR

readr

ggplot2

reshape2

pegas

ape

ade4

adegenet

poppr

readxl

readr

devtools

gstudio

hierfstat

poppr

StAMPP

lattice

dplyr

Import data into R

```
AU1 <- read.vcfR("~/Desktop/aurita/SNP.AUdp5p05FHWE2Apca.recode.vcf")
```

```
## Scanning file to determine attributes.
## File attributes:
##   meta lines: 63
##   header_line: 64
##   variant count: 1787
##   column count: 63
##
```

```
Meta line 63 read in.
## All meta lines processed.
## gt matrix initialized.
## Character matrix gt created.
##    Character matrix gt rows: 1787
##    Character matrix gt cols: 63
##    skip: 0
##    nrows: 1787
##    row_num: 0
##
Processed variant 1000
Processed variant: 1787
## All variants processed
```

convert dataset into genind objects

```
genindB <- vcfR2genind(AU1)
strata<- read.table("~/Desktop/S_A.strata.txt", header=TRUE)
strata_df <- data.frame(strata)
strata(genindB) <- strata_df
setPop(genindB) <- ~Population
```

Test for Population Structure

```
fstat(genindB)
```

```
##                pop        Ind
## Total 0.001050677 -0.4490054
## pop   0.000000000 -0.4505294
```

```
wc(genindB)
```

```
## $FST
## [1] 0.001050677
##
## $FIS
## [1] -0.4505294
```

Calculating pairwise FST values

```
Fst <- pairwise.fst(genindB)
Fst
```

```
##             1          2          3          4
## 2 0.01458640
## 3 0.01332540 0.01127876
## 4 0.01572604 0.01404788 0.01276877
## 5 0.01781229 0.01436599 0.01299351 0.01532280
```

calculate genetic differentiation between pairs of population using genetic distance

```
GD <- genet.dist(genindB, method = "WC84")
GD
```

```
##                    Benin         Ghana    Mauritania       Senegal
## Ghana       -8.851079e-04
## Mauritania   1.296542e-03 -5.242286e-06
## Senegal      6.002021e-05  3.690144e-04  2.167299e-03
## Togo         4.278014e-03 -2.334086e-04  2.239599e-03  1.389453e-03
```

Calculating WC pairwise Fst betwen all pairs of regions using heatmap

```
gindF.fst.mat <- pairwise.fst(genindB, pop = strata(genindB)$Population,res.type = "matrix")
gindF.fst.mat
```

```
##                 Benin      Ghana Mauritania    Senegal       Togo
## Benin      0.00000000 0.01458640 0.01332540 0.01572604 0.01781229
## Ghana      0.01458640 0.00000000 0.01127876 0.01404788 0.01436599
## Mauritania 0.01332540 0.01127876 0.00000000 0.01276877 0.01299351
## Senegal    0.01572604 0.01404788 0.01276877 0.00000000 0.01532280
## Togo       0.01781229 0.01436599 0.01299351 0.01532280 0.00000000
```

```
reg_names = c("Benin","Ghana","Mauritania","Senegal","Togo")
colnames(gindF.fst.mat)<- reg_names
rownames(gindF.fst.mat)<- reg_names

gindF.fst.mat.tri <- gindF.fst.mat
gindF.fst.mat.tri[lower.tri(gindF.fst.mat, diag=TRUE)] <- NA
melted <- melt(gindF.fst.mat, na.rm =TRUE)
par(mfrow=c(2,1))
ggplot(data = melted, aes(Var2, Var1, fill = value))+ geom_tile(color = "white")+
  scale_fill_gradient(low = "white", high = "red", name="FST")   +
  ggtitle(expression(atop("Pairwise FST, WC (1984)", atop(italic("N = 137, L = 9,170"), ""))))+
  labs( x = "Sampling Site", y = "Sampling Site") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, size = 6, hjust = 1),axis.text.y = element_te
  coord_fixed()
```
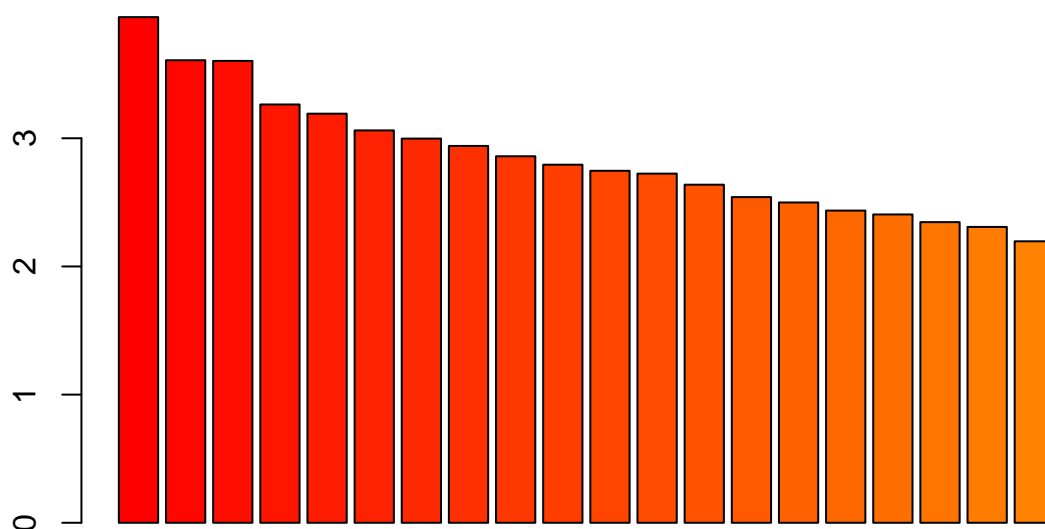
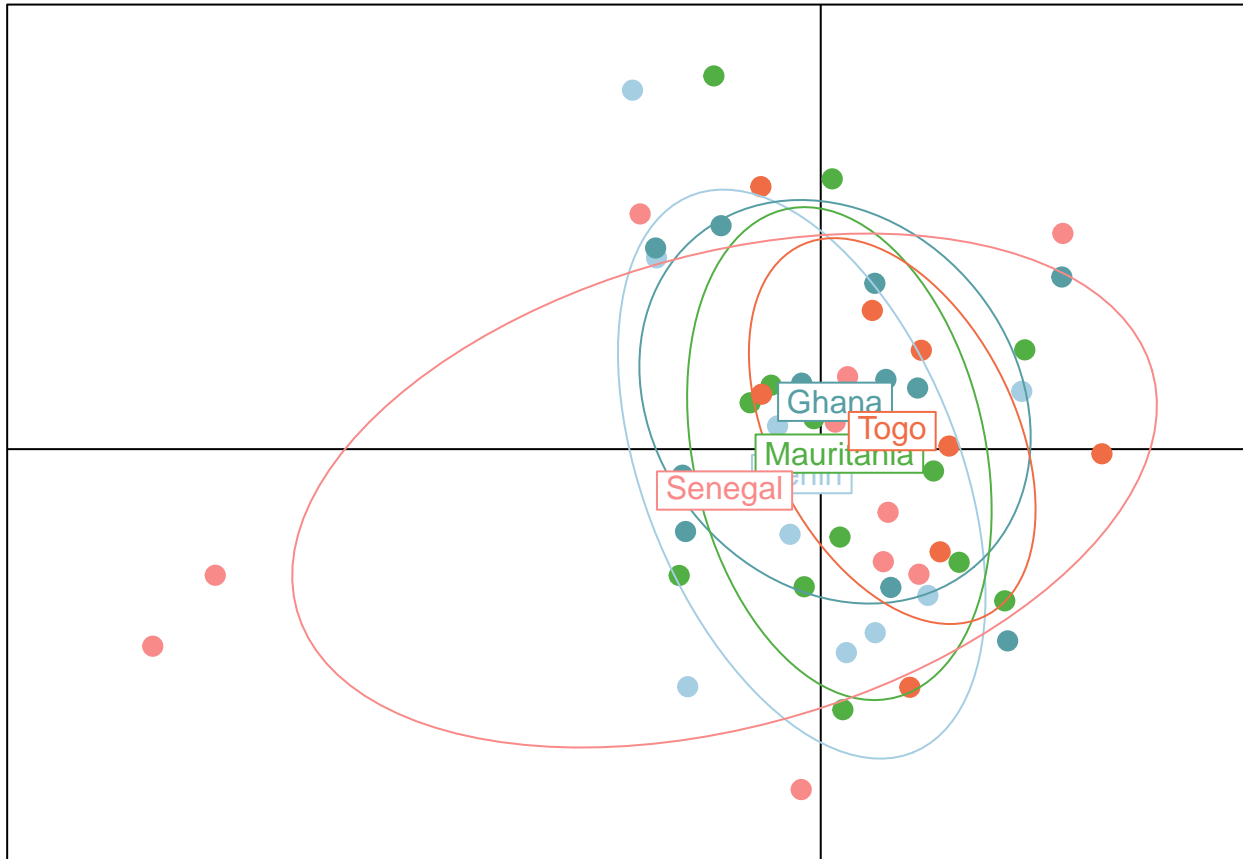## Pairwise FST, WC (1984)

N = 137, L = 9,170
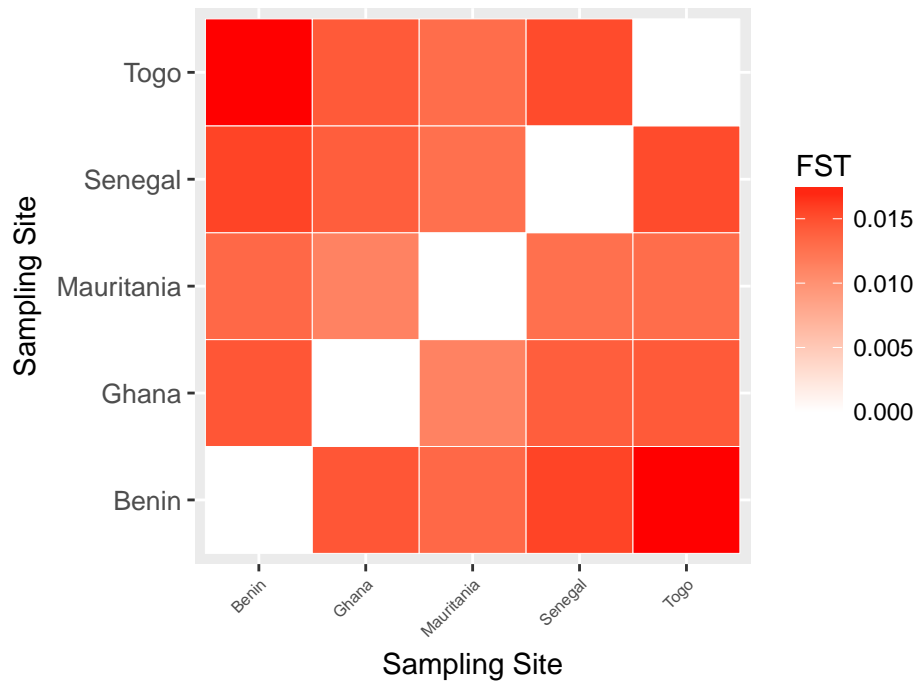
Principal Component Analysis (PCA)

```r
X <- tab(genindB, freq = TRUE, NA.method = "mean")
pca1 <- dudi.pca(X, scale = FALSE, scannf = FALSE, nf = 3)
barplot(pca1$eig[1:20], main = "PCA eigenvalues", col = heat.colors(50))
```

PCA eigenvalues

```
col <- funky(9)
s.class(pca1$li, pop(genindB),xax=1,yax=2, col=col, axesell=FALSE, cstar=0, cpoint=2, grid=FALSE)
```


Discriminant Analysis of Principal Components(DAPC) plot

```
aa.genlightB <- vcfR2genlight(AU1, n.cores=1)
locNames(aa.genlightB) <- paste(AU1@fix[,1],AU1@fix[,2],sep="_")
pop(aa.genlightB)<-substr(indNames(aa.genlightB),1,3)

dapcB <- dapc(aa.genlightB, n.pca=13, n.da=2)
scatter(dapcB,scree.da=TRUE, bg="white", posi.pca="topright", legend=FALSE,
txt.leg=paste("group", 1:5), col=c("red","blue", "green", "black", "violet"))
```
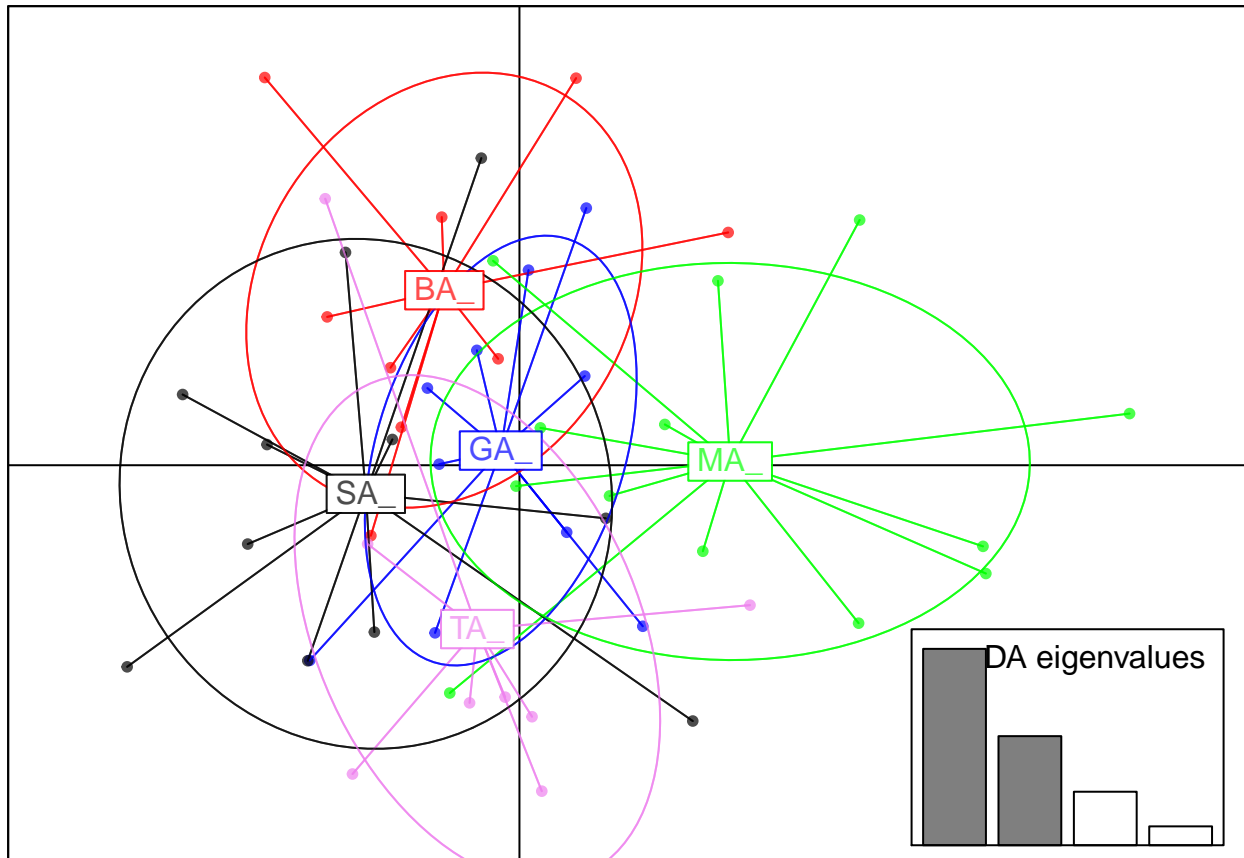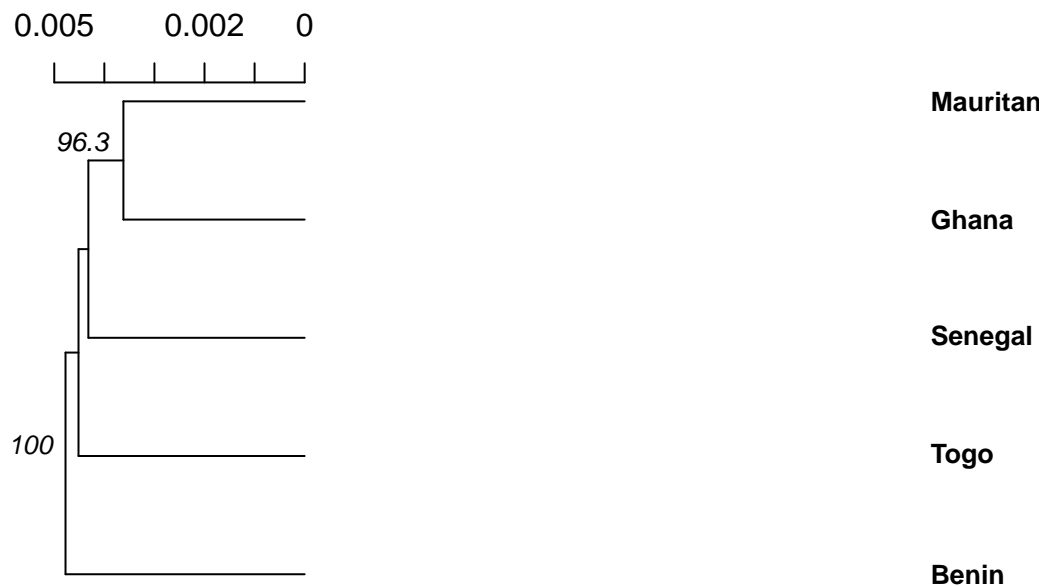
Neighbour Joining Tree plotted with boostrap values

```
set.seed(999)
genindB %>%
  genind2genpop(pop = ~Population) %>%
  aboot(cutoff = 50, quiet = TRUE, sample = 54, distance = nei.dist)

##
##  Converting data from a genind to a genpop object...
##
## ...done.
```

```
0.005      0.002     0
```

Mauritan

Ghana

Senegal

Togo

Benin

96.3

100

```
##
## Phylogenetic tree with 5 tips and 4 internal nodes.
##
## Tip labels:
## [1] "Benin"      "Ghana"      "Mauritania" "Senegal"    "Togo"
## Node labels:
## [1] 100.0000       NA       NA  96.2963
##
## Rooted; includes branch lengths.
```

structure-like plot indicating membership probability of each sample

```r
dapcB.results <- as.data.frame(dapcB$posterior)
dapcB.results$pop <- pop(genindB)
dapcB.results$indNames <- rownames(dapcB.results)
library(reshape2)
dapcB.results <- melt(dapcB.results)
```

```
## Using pop, indNames as id variables
```

```r
colnames(dapcB.results) <- c("Original_Pop","Sample","Assigned_Pop","Posterior_membership_probability")

p <- ggplot(dapcB.results, aes(x=Sample,y=Posterior_membership_probability, fill=Assigned_Pop))
p <- p + geom_bar(stat='identity')
p <- p + facet_grid(~Original_Pop, scales = "free")
p <- p + theme(axis.text.x = element_text(angle = 90, hjust = 1, size = 5))
p
```
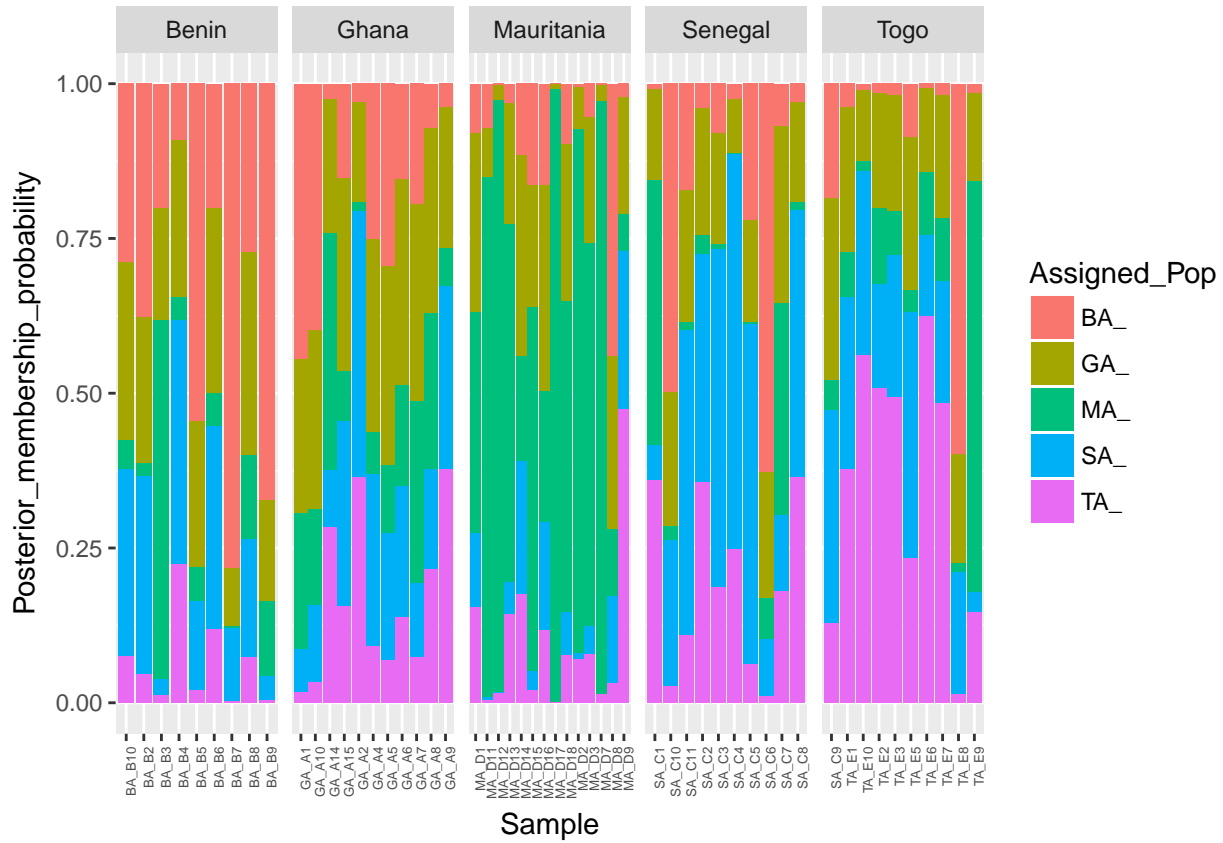
```r
# calculate mean per population
grouped_data <- group_by(dapcB.results, Original_Pop, Assigned_Pop)
data_means <- summarise(grouped_data, mean=mean(Posterior_membership_probability))
# plot means for each original pop colored by assigned pop
pieB <- ggplot(data_means, aes(x=Original_Pop,y=mean, fill=Assigned_Pop))
pieB <- pieB + geom_bar(stat='identity') + coord_polar("y",start=0)
pieB <- pieB + facet_grid(~Original_Pop, scales = "free") + theme(axis.text=element_blank(), axis.ticks=
pieB
```