**APPENDIX S1**

# Simple trace function

tr <- function(mat){

 trace <- sum(diag(mat))

 return(trace)

}

# three way unbalanced non-parametric MANOVA

threewaynpFtest <- function(sharks, Xr, Xf, a){

 # TESTS FOR FACTOR A EFFECT

 # sharks = MATRIX WHERE EACH ROW REPRESENTS A PROFILE OF A DIFFERENT INDIVIDUAL

 # a = NUMBER OF FACTORS IN TREATMENT A BEING ANALYSED

 # Xr = THE REDUCED DESIGN MATRIX EXCLUDING FACTOR A

 # Xf = THE FULL DESIGN MATRIX

 N <- nrow(sharks)

 d <- matrix(0, nrow=N, ncol=N)

 # USING AITCHISON'S DISTANCE MEASURE aDist, CALCULATING A DISTANCE MATRIX

 for(i in 1:N){

 for(j in 1:N){

 d[i,j] <- aDist(sharks[i,],sharks[j,])

 }

 }

 dsqr <- d^2

 A <- (-1/2)\*dsqr

 Hr <- Xr%\*%solve(t(Xr)%\*%Xr)%\*%t(Xr)

 Hf <- Xf%\*%solve(t(Xf)%\*%Xf)%\*%t(Xf)

 ones <- matrix(1,N,1)

 I <- diag(N)

 g <- (I - (1/N)\*ones%\*%t(ones))

 G <- g%\*%A%\*%g

 SSA <- tr(Hf%\*%G%\*%Hf)-tr(Hr%\*%G%\*%Hr)

 SSR <- tr((I-Hf)%\*%G%\*%(I-Hf))

 SST <- tr(G)

 F <- ((tr(Hf%\*%G%\*%Hf)-tr(Hr%\*%G%\*%Hr))/(a-1))/(tr((I-Hf)%\*%G%\*%(I-Hf))/(N-ncol(Xf)))

 return(F)

}

p.value3way <- function(sharks, Xa, Xb, Xc, Xf, a, b, c, nperm=1000){

 # P-VALUE FOR THREE WAY MANOVA

 # sharks = EACH ROW IS AN INDIVIDUAL SHARK FA SIGNATURE

 # nperm = THE NUMBER OF PERMUTATIONS TO COMPUTE FOR THE P-VALUE

 # Xa = THE REDUCED DESIGN MATRIX EXCLUDING FACTOR A

 # Xb = THE REDUCED DESIGN MATRIX EXCLUDING FACTOR B

 # Xc = THE REDUCED DESIGN MATRIX EXCLUDING FACTOR C

 # a = THE NUMBER OF FACTORS IN TREATMENT A

 # b = THE NUMBER OF FACTORS IN TREATMENT B

 # c = THE NUMBER OF FACTORS IN TREATMENT C

 Foriga <- threewaynpFtest(sharks, Xa, Xf, a)

 Forigb <- threewaynpFtest(sharks, Xb, Xf, b)

 Forigc <- threewaynpFtest(sharks, Xc, Xf, c)

 # CALCULATING nperm RANDOM PERMUTATIONS AS DESCRIBED IN MARTI ANDERSON'S PAPER

 # 1000 FOR 0.05 SIGNIFICANCE, 5000 FOR 0.01 SIGNIFICANCE

 Fperma <- numeric(nperm)

 Fpermb <- numeric(nperm)

 Fpermc <- numeric(nperm)

 for(i in 1:nperm){

 ind <- c(1:nrow(sharks))

 perm <- sample(ind)

 sharks.perm <- sharks[perm,]

 Fperma[i] <- threewaynpFtest(sharks.perm, Xa, Xf, a)

 Fpermb[i] <- threewaynpFtest(sharks.perm, Xb, Xf, b)

 Fpermc[i] <- threewaynpFtest(sharks.perm, Xc, Xf, c)

 }

 # HOW MANY PERMUTATIONS WERE GREATER THAN THE ORIGINAL Forig

 p.valuea <- sum(Fperma > Foriga)/nperm

 p.valueb <- sum(Fpermb > Forigb)/nperm

 p.valuec <- sum(Fpermc > Forigc)/nperm

 return(c(Foriga, p.valuea, Forigb, p.valueb, Forigc, p.valuec))

}