Journal: Tree Genetics and Genomes

Genetic diversity and population structure analysis reveals the unique genetic composition of South African selected macadamia accessions

Mary Ranketse[1,2], Charles A. Hefer[2,3], Rian Pierneef[2], Gerda Fourie[1], Alexander A. Myburg[1]

[1] Department of Biochemistry, Genetics and Microbiology, Forestry and Agricultural Biotechnology Institute (FABI), University of Pretoria, Pretoria, South Africa

[2] Biotechnology Platform, Agricultural Research Council – Onderstepoort Veterinary Institute, Pretoria, South Africa

[3] AgResearch Ltd, Lincoln, New Zealand

Email: zander.myburg@fabi.up.ac.za

Online Resource ESM_2

Script S2

R-script for generating 3D PCA plot using Plotly in RStudio

```
########## Load plotly

library(plotly)
setwd("")

########## Upload and check input file

PCA.result <- read.csv("InputFile.csv",sep=";")
head(PCA.result)

########## Add a size column to the data frame with 1 in all rows of the column

PCA.result$size <-1

########## Calls a list of colors for drawing your plot

my_colors <- c("#2e7d32","#CC9900","#424949","#ad1457","#1565c0")

########## Gives a list of names to be associated to the colors called in the previous line of code (association by order of entries)
```

```
my_colors <- setNames(my_colors, c("HAES","AUS","OTH","SA","FARM"))


########## Declares the font variable to be used in the image

f <- list(
  family = "Arial, sans-serif",
  size = 14,
  color = "#000000"
)

t <- list(
  family = "sans serif",
  size = 8)

######### Generate the 3D PCA plot

p_v2 <- plot_ly(PCA.result,
              x = ~PC1,
              y = ~PC2,
              z = ~PC3,
              text = PCA.result$SampleName,
              color = ~ OriginName,
              opacity = 1,
              size= ~ size,
              colors = my_colors) %>%
  add_markers() %>%
  add_text(textfont = t, textposition = "right", colors = my_colors) %>%
  layout(font = f,
         scene = list(xaxis = list(title ='<b>PC1: EV=90.91 (22%)</b>'),
                      yaxis = list(title = '<b>PC2: EV=39.93 (9%)</b>'),
                      zaxis = list(title = '<b>PC3: EV=33.92 (8%)</b>')))

p_v2

############ Generate the plot to be viewed online in Plotly

Sys.setenv("plotly_username"="username")
Sys.setenv("plotly_api_key"="yourAPIkey") ### The API key can be obtained from https://plot.ly/r/getting-started


chart_link = api_create(p_v2, filename="OutputFile") ### Create a link to the Plotly website where anyone with the link can interact with the plot.

chart_link   ####opens link in web browser
```

```
# In a browser:
# https://chart-studio.plotly.com/~mranketse1/17/#/

########### Exporting good quality images

png <- plotly_IMAGE(p_v2, width = 800, height = 600, format = "png", scale = 3,
                    out_file = "OutputFile.png")

###########References

sessionInfo()
```

Script S3

R-script for generating STRUCTURE plot using LEA in RStudio

```
########## Load LEA, plyr and related packages
library(LEA)
library(plyr)
setwd("")

source("http://membres-timc.imag.fr/Olivier.Francois/Conversion.R")
source("http://membres-timc.imag.fr/Olivier.Francois/POPSutilities.R")

########## Upload input file and run analysis

input.file = "InputFile.str"
struct2geno(file = input.file, TESS = TRUE, diploid = TRUE, FORMAT = 1,
        extra.row = 0, extra.col = 0, output = "./genotype.geno")

obj.snmf = snmf("InputFile.geno", K = 1:4, entropy = TRUE,
            repetitions = 20, project = "new")

project.snmf = load.snmfProject("InputFile.snmfProject")

structure_plot_names <- read.delim("StructureNames.txt")   ### Upload sample name file
head(structure_plot_names)


########## Draw cross-entropy graph

CE.graph.label <- c(expression(paste("Macadamia species cross-entropy K1-4, n = 110,
neutral genetic space (m = 13)")))
plot(project.snmf, col = "#4d4d4d",
    pch = 19, cex = 1.2,
    main = CE.graph.label)


########## K = 2

ce <- cross.entropy(project.snmf, K = 2)
best <- which.min(ce)
my.colors <- c("#ad1457","#2e7d32")

### Ordering according to Q and population groups
### (Taken from: https://www.royfrancis.com/structure-sort-by-q-explained/)

qmatrix_K2 = Q(project.snmf, K = 2, run = best)
maxval <- apply(qmatrix_K2,1,max)   ###calc max
```

```
matchval <- vector(length=nrow(qmatrix_K2))  ###check match
for(j in 1:nrow(qmatrix_K2)) matchval[j] <- match(maxval[j],qmatrix_K2[j,])  ###calc match
q2 <- cbind(matchval,maxval)  ### combine match and max into own matrix
qmatrix_K2_q2 <- cbind2(qmatrix_K2, q2)  ### combine match/max matrix to qmatrix matrix
colnames(qmatrix_K2_q2) <- c("K1", "K2", "matchval", "maxval")  ### change column names
qmatrix_K2_q2names <- cbind2(qmatrix_K2_q2, structure_plot_names) ### combine match/max/qmatrix to name matrix
qmatrix_K2_qsorted <- qmatrix_K2_q2names[order(matchval,-maxval), ]  ### sort by match and max
qmatrix_K2_qsortedPop <- arrange(qmatrix_K2_qsorted,Pop)  ###arrange according to pop
delete <- c("maxval","matchval","order","Sample","Pop","Code")
qmatrix_K2_qsortedPop2 <- qmatrix_K2_qsortedPop[,!(colnames(qmatrix_K2_qsortedPop) %in% delete), drop=FALSE]

### Draw STRUCTURE plot

graph.label <- c(expression(paste("Macadamia species, K = 2, 110 individuals, 13MSs")))

barplot(t(qmatrix_K2_qsortedPop2), col = my.colors, border = NA, space = 0,
      xlab = "Individuals", ylab = "Ancestory proportions",
      main = "Ancestry matrix") -> bp
axis(1, at = 1:length(order(bp)),
    labels = qmatrix_K2_qsortedPop$Code, las = 3,
    cex.axis = .4)


########## K = 3
ce <- cross.entropy(project.snmf, K = 3)
best <- which.min(ce)
my.colors <- c("#2e7d32","#ad1457","#1565c0")

qmatrix_K3 = Q(project.snmf, K = 3, run = best)


### Ordering according to Q and population groups
### (Taken from: https://www.royfrancis.com/structure-sort-by-q-explained/)

maxval <- apply(qmatrix_K3,1,max)  ###calc max
matchval <- vector(length=nrow(qmatrix_K3))  ###check match
for(j in 1:nrow(qmatrix_K3)) matchval[j] <- match(maxval[j],qmatrix_K3[j,])  ###calc match
q3 <- cbind(matchval,maxval)  ### combine match and max into own matrix
qmatrix_K3_q3 <- cbind2(qmatrix_K3, q3)  ### combine match/max matrix to qmatrix matrix
colnames(qmatrix_K3_q3) <- c("K1", "K2","K3", "matchval", "maxval")  ### change column names
```

```
qmatrix_K3_q3names <- cbind2(qmatrix_K3_q3, structure_plot_names) ### combine match/max/qmatrix to name matrix
qmatrix_K3_qsorted <- qmatrix_K3_q3names[order(matchval,-maxval), ]  ### sort by match and max
qmatrix_K3_qsortedPop <- arrange(qmatrix_K3_qsorted,Pop)  ###arrange according to pop
delete <- c("maxval","matchval","order","Sample","Pop","Code")
qmatrix_K3_qsortedPop_3 <-qmatrix_K3_qsortedPop[,!(colnames(qmatrix_K3_qsortedPop) %in% delete), drop=FALSE]

### Draw STRUCTURE plot

graph.label <- c(expression(paste("Macadamia species, K = 3, 110 individuals, 13MSs")))

barplot(t(qmatrix_K3_qsortedPop_3), col = my.colors, border = NA, space = 0,
      xlab = "Individuals", ylab = "Ancestory proportions",
      main = "Ancestry matrix") -> bp
axis(1, at = 1:length(order(bp)),
   labels = qmatrix_K3_qsortedPop$Code, las = 3,
   cex.axis = .4)


########## K = 4
ce <- cross.entropy(project.snmf, K = 4)
best <- which.min(ce)
my.colors <- c("#2e7d32","#1565c0","#ad1457","#CC9900")

qmatrix_K4 = Q(project.snmf, K = 4, run = best)


### Ordering according to Q and population groups
### (Taken from: https://www.royfrancis.com/structure-sort-by-q-explained/)

maxval <- apply(qmatrix_K4,1,max)  ###calc max
matchval <- vector(length=nrow(qmatrix_K4))  ###check match
for(j in 1:nrow(qmatrix_K4)) matchval[j] <- match(maxval[j],qmatrix_K4[j,])  ###calc match
q4 <- cbind(matchval,maxval)  ### combine match and max into own matrix
qmatrix_K4_q4 <- cbind2(qmatrix_K4, q4)  ### combine match/max matrix to qmatrix matrix
colnames(qmatrix_K4_q4) <- c("K1", "K2","K3","K4", "matchval", "maxval")  ### change column names
qmatrix_K4_q4names <- cbind2(qmatrix_K4_q4, structure_plot_names) ### combine match/max/qmatrix to name matrix
qmatrix_K4_qsorted <- qmatrix_K4_q4names[order(matchval,-maxval), ]  ### sort by match and max
qmatrix_K4_qsortedPop <- arrange(qmatrix_K4_qsorted,Pop)  ###arrange according to pop
delete <- c("maxval","matchval","order","Sample","Pop","Code")
```

```
qmatrix_K4_qsortedPop_4 <-qmatrix_K4_qsortedPop[,!(colnames(qmatrix_K4_qsortedPop)
%in% delete), drop=FALSE]

### Draw STRUCTURE plot

graph.label <- c(expression(paste("Macadamia species, K = 4, 110 individuals, 13MSs")))

barplot(t(qmatrix_K4_qsortedPop_4), col = my.colors, border = NA, space = 0,
      xlab = "Individuals", ylab = "Ancestory proportions",
      main = "Ancestry matrix") -> bp
axis(1, at = 1:length(order(bp)),
    labels = qmatrix_K4_qsortedPop$Code, las = 3,
    cex.axis = .4)


######### References

sessionInfo()
```