File: ReadmeAMA.pdf
Author: Octavio Martinez (octavio.martinez@cinvestav.mx)
Date: Sun Apr 9 11:10:55 2023

Introduction
The "AMA" R function implements the "All Marker Alleles" algorithm as described in (Hayano-Kanashiro et al. 2017).
For details of the algorithm please see the supplementary file "ece32754-sup-0001-DataS1.pdf" of that paper that you can download from the URL:

https://onlinelibrary.wiley.com/doi/10.1002/ece3.2754

In summary, given a matrix of accessions (rows) and frequencies of different genetic markers (columns), the algorithm select the minimum collection of accessions that contains all genetic markers.

In many cases this algorithm dramatically reduces the collection of accessions to a "core" collection which represents all genetic diversity present in the larger original collection. This is useful to prioritize germplasm for conservation proposes.

Additionally, the function produces a rareness coefficient for each one of the accessions. This could be also useful to prioritize conservation of germplasm.

The algorithm is exemplified with a collection of 1338 Mexican maize accessions which are genotyped for a total of 333 combination of SSR marker/allele combinations; see Martínez et al. (2023).

Installing.
The compressed file "AMA2023.RData.zip" contains the function, data and results. After uncompressing "AMA2023.RData.zip" you get file "AMA2023.RData". Such file can be loaded into your environment with the R statement:

```
load("AMA2023.RData")
```

After this you will have in your environment the following objects:

"AMA" - R function implementing the algorithm.
"incidence" - A matrix of 1338 rows x 333 columns.
"incidence.AMA" - R object resulting from applying the AMA algorithm to the incidence matrix by the R statement:

```
incidence.AMA <- AMA(incidence)
```

Details

The AMA function.

```
## INPUT:
# mat - Matrix [class(incidence) == c("matrix" "array")]
# with accessions as rows and columns as marker-alleles.
# Data in "mat" can be allele presence / absence (1, 0),
# frequencies or any numeric scale, but zero means the ABSENCE
# of the corresponding marker/allele.
# Not missing data (NA's) are allowed!
# BOTH rows (accessions) and columns (marker/allele combinations)
# must be named with DIFFERENT names.

## OUTPUT:
# A list with the following components:
# "Set.names" - Names of the accessions: dimnames(mat)[[1]]
# "Set.nums" - Numbers of rows in mat.
# "Richness" - Accumulated richness in each step of the algorithm
# "Ties" - Cases of ties (solved by using the "reareast" accession)
# "MinMat" - Matrix containing the accessions that solve the problem.
# "rho" - Rareness coefficients for the original accessions.
# "nam.unique" - Names of marker/alleles that are "unique" (appears
only in one accession)
# "acc.unique" - Names of accessions that have unique marker/allele
combinations.
# "on.acc" - Original number of accessions in the input matrix (mat)
# "comment" - A comment that is printed as output.
# "call" - The call of the function.

##### A dummy example (easy to understand)
# We are going to simulate a very simple example to understands
# what the AMA algorithm do.
# If you which, run the lines below to perform the calculations.
# Otherwise you could simply examine the results here.

# Create an empty matrix of incidence (presence/absence marker/
alleles)
# We assume to be sampling from three "regions": r1, r2 and r3.
# From each one of the regions we sample three accessions: a1, a2, a3.

temp.acc.n <- c(paste(rep(paste(rep("r", 3), c(1:3), sep=""), each=3),
rep(paste("a", c(1:3), sep=""), 3), sep="."), "out")
temp.m.n <- paste("m", c(1:10), sep="")
temp.inc <- matrix(NA, nrow=10, ncol=10, dimnames=list(temp.acc.n,
temp.m.n))

# Fill the matrix with pseudorandom values (different for each region)
set.seed(1959)
temp.inc[1:3, 1:3] <- sample(c(0,1), size=9, prob=c(.1, .9),
replace=TRUE)
temp.inc[4:10, 1:3] <- 0
temp.inc[1:3, 4:6] <- 0
temp.inc[4:6, 4:6] <- sample(c(0,1), size=9, prob=c(.3, .7),
```

```
replace=TRUE)
temp.inc[7:10, 4:6] <- 0
temp.inc[1:6,7:9] <- 0
temp.inc[7:9, 7:9] <- sample(c(0,1), size=9, replace=TRUE)
temp.inc[10, 7:9] <- 0
temp.inc[1:10, 10] <- c(rep(0,9), 1)

temp.inc # See the matrix resulting from this process:
      m1 m2 m3 m4 m5 m6 m7 m8 m9 m10
r1.a1  1  1  1  0  0  0  0  0  0   0
r1.a2  1  1  1  0  0  0  0  0  0   0
r1.a3  0  1  1  0  0  0  0  0  0   0
r2.a1  0  0  0  1  1  1  0  0  0   0
r2.a2  0  0  0  0  1  1  0  0  0   0
r2.a3  0  0  0  1  1  1  0  0  0   0
r3.a1  0  0  0  0  0  0  1  1  0   0
r3.a2  0  0  0  0  0  0  0  0  1   0
r3.a3  0  0  0  0  0  0  0  1  0   0
out    0  0  0  0  0  0  0  0  0   1

# At simple sight we can see how the marker/alleles combinations
# are exclusive from some regions,
# m1 to m3 appear only in individuals from "r1".
# m4 to m6 appear only in individuals from "r2".
# m7 to m9 appear only in individuals from "r3" and finally,
# m10 appears only in the "out" accession ("out" for "out-group").
# Now we run the algorithm:

temp.res <- AMA(temp.inc)
# This produces the following comment:
A set of 7 accessions contains all 10 marker/alleles from the
collection of 10 accessions.
The set of selected accessions represents the 70% of the original.

# Now let's see some of the components.
# (in the lines below I include the R prompt, ">" to run those
# lines just exclude the prompt, copy and paste in the R window)

> names(temp.res) # Names of the components of the result.
 [1] "Set.names"  "Set.nums"   "Richness"   "Ties"       "MinMat"
 [6] "rho"        "nam.unique" "acc.unique" "on.acc"     "comment"
[11] "call"

> temp.res$Set.names # The names of the selected set
[1] "r3.a1" "r3.a2" "out"   "r1.a1" "r1.a2" "r2.a1" "r2.a3"
> temp.res$Set.nums # The numbers of the selected set
[1]  7  8 10  1  2  4  6
> temp.inc[temp.res$Set.nums,] # In the original matrix
      m1 m2 m3 m4 m5 m6 m7 m8 m9 m10
r3.a1  0  0  0  0  0  0  1  1  0   0
```

```
r3.a2  0  0  0  0  0  0  0  0  1   0
out    0  0  0  0  0  0  0  0  0   1
r1.a1  1  1  1  0  0  0  0  0  0   0
r1.a2  1  1  1  0  0  0  0  0  0   0
r2.a1  0  0  0  1  1  1  0  0  0   0
r2.a3  0  0  0  1  1  1  0  0  0   0
# This matrix is equivalent to the matrix with the minimum collection
> temp.res$MinMat # This contains the individuals selected:
      m1 m2 m3 m4 m5 m6 m7 m8 m9 m10
r1.a1  1  1  1  0  0  0  0  0  0   0
r1.a2  1  1  1  0  0  0  0  0  0   0
r2.a1  0  0  0  1  1  1  0  0  0   0
r2.a3  0  0  0  1  1  1  0  0  0   0
r3.a1  0  0  0  0  0  0  1  1  0   0
r3.a2  0  0  0  0  0  0  0  0  1   0
out    0  0  0  0  0  0  0  0  0   1
# (i.e., it has the same rows, even if in different order)
# Note how this collection of 7 accessions includes all the
# ten markers (there is at least one "1" in each column)

# Note that 3 accessions (rows) were excluded from the output
# because they are redundant (they do not contribute with further
# alleles)
> temp.inc[setdiff(c(1:10), temp.res$Set.nums),]
      m1 m2 m3 m4 m5 m6 m7 m8 m9 m10
r1.a3  0  1  1  0  0  0  0  0  0   0
r2.a2  0  0  0  0  1  1  0  0  0   0
r3.a3  0  0  0  0  0  0  0  1  0   0

> temp.res$rho # The values of the "rareness" coefficient.
    r1.a1     r1.a2     r1.a3     r2.a1     r2.a2     r2.a3     r3.a1
0.4370355 0.4370355 0.3619392 0.4370355 0.3619392 0.4370355 0.4370355
    r3.a2     r3.a3       out
0.3619392 0.3331666 0.3619392
# Let's order that vector from larger to smaller value of rareness:
> temp.r <- temp.res$rho[order(temp.res$rho, decreasing=TRUE)]
> round(temp.r, 4) # Rounded to 4 decimal places.
 r1.a1  r1.a2  r2.a1  r2.a3  r3.a1  r3.a2    out  r1.a3  r2.a2  r3.a3
0.4370 0.4370 0.4370 0.4370 0.4370 0.3619 0.3619 0.3619 0.3619 0.3332
# Note that here we have many ties in rareness, thus, different
# collections could be used to represent all diversity.
# Ties were broken at random and the output consist of the accessions:
> temp.res$Set.names
[1] "r3.a1" "r3.a2" "out"   "r1.a1" "r1.a2" "r2.a1" "r2.a3"


# Plot of the dendrogram in the original dummy data:
> plot(hclust(dist(temp.inc, method="euclidean"), method="average"))
# Note how the clustering is by regions (out with r3).
```

```
# Now, the dendrogram of the resulting collection:
> plot(hclust(dist(temp.res$MinMat, method="euclidean"),
method="average"))

# If you has done the calculations and want to remove the temporal
# objects created could run:
# rm(list=ls(patt="temp")) # Will remove all objects which name "temp"

##### The example with the real data included:
# Here I include the R prompt. You could examine the results
# as indicated here without performing the algorithm
# (which will take some seconds)

# Lets summarize the matrix of incidence:
> class(incidence)
[1] "matrix" "array"
> dim(incidence)
[1] 1338  333
# Thus, there are 1,338 accessions and 333 combinations of marker/
alleles
# Let's have a look to some of the elements:
> incidence[1:5, 1:5] # The first five accessions an marker alleles.
      PHI015_63 PHI015_66 PHI015_69 PHI015_72 PHI015_75
CH001         0         0         0         0         0
CH002         0         0         0         0         0
CH003         0         0         0         0         0
CH004         0         0         0         0         2
CH005         0         0         0         0         0
# Briefly, the numbers in the cells are the counts of plant batches
# that gave positive for each one of the marker allele.
# Thus the numbers in the cells are: 0, 1, 2 or 3. and there are
> prod(dim(incidence)) # Number of cells in that matrix
[1] 445554
# Now we can tabulate:
> table(as.vector(incidence))
     0      1      2      3
338451  36502  29042  41559
> table(as.vector(incidence))/445554 # In relative frequency
         0          1          2          3
0.75961836 0.08192497 0.06518177 0.09327489
# Thus, we have a very sparse matrix with more than 75% of its values
=0.

# Now, we could run the algorithm on this matrix:
# (Not needed; you already have the result)
> incidence.AMA <- AMA(incidence)

A set of 56 accessions contains all 333 marker/alleles from the
collection of 1338 accessions.
The set of selected accessions represents the 4.19% of the original.
```

```
# The sorted names of the selected accessions:
> sort(incidence.AMA$Set.names)
 [1] "CH007" "CH010" "CH015" "CH069" "CH124" "CL088" "GR050" "GR160"
 [9] "GR166" "GR170" "GR191" "GR298" "GR518" "GR521" "GT070" "MC028"
[17] "MC059" "MC070" "MC080" "MC083" "MC103" "MC112" "MC113" "MC118"
[25] "MC121" "MC140" "MC169" "MC224" "MC310" "MC378" "MC385" "PA007"
[33] "PA022" "PA027" "PA031" "PA035" "PA047" "PL015" "PL047" "PL075"
[41] "PL099" "PL102" "PL107" "PL131" "PL135" "TE004" "TE005" "TE017"
[49] "TE039" "TE063" "TL019" "TL032" "TL042" "TL044" "TS022" "TS035"

# Note that in this case a very small collection of 56 accessions is
able
# to include all marker/allele combinations.
# You can for example plot the dendrogram of this small collection:
> plot(hclust(dist(incidence.AMA$MinMat, method="euclidean"),
method="average"), cex=0.5)
```

References

Hayano-Kanashiro, C, Martínez de la Vega, O, Reyes-Valdés, MH, Pons-Hernández, J-L, Hernández-Godinez, F, Alfaro-Laguna, E, Herrera-Ayala, JL, Vega-Sánchez, Ma.C, Carrera-Valtierra, JA and Simpson, J. "An SSR-based approach incorporating a novel algorithm for identification of rare maize genotypes facilitates criteria for landrace conservation in Mexico". Ecology and Evolution. 2017; 7: 1680–1690. https://doi.org/10.1002/ece3.2754

Martínez O., Ceniceros-Ojeda A., Hayano-Kanashiro C., Reyes-Valdés H., Pons –Hernández J.L. and Simpson J. "Criteria for prioritizing selection of Mexican maize landrace accessions for conservation in situ or ex situ based on phylogenetic analysis". 2023 (In preparation).