
title: "Genetic Diversity of Invasive *Pyrus calleryana* trees"

author: "Shiwani"

date: "4/30/2020"

Pyrus calleryana (callery pear) trees were imported from China to the US to combat fireblight disease in European pear in 1917. Those callery pear trees were hugely used as rootstock for several edible pears. Now, these trees are invasive species in most of the eastern and southern states of the US. For our study, *P. calleryana* samples were collected from China, Japan, Korea and US. Initially, we had 100 samples but at the end, we were limited with only 57 samples as other samples were cutoff due to several reasons. Our study aims at studying the genetic diversity of those trees and evaluate the relatedness of US species with the Asian ones. Microsatellite loci are used for the study.

library(yaml)

library(tidyr)

library(poppr)

library(hierfstat)

library(mmod)

library(ape)

library(magrittr)

library(adeigenet)

library(MASS)

library(geiger)

library(pegas)

library(treemap)

library(car)

library(agricolae)

library(ggplot2)

library(foreign)

library(plyr)

library(caTools)

library(reshape2)

library(ecodist)

```
library(polysat)
library(radiator)
library(diveRsity)
library(doParallel)
library(xlsx)
library(shiny)
library(rJava)
library(PopGenReport)
library(devtools)
library(roxygen2)
library(pegas)
library(lattice)
library(ade4)
library(vegan)
```

```
#Loading Genalex datafile
```

```
```{r}
```

```
tk$ <- read.genalex("C:/Users/ssapkot1/Desktop/GenalexAsian.csv", ploidy = 2, geo = FALSE, region = FALSE, genclone = FALSE)
```

```
tk$
```

```
```
```

```
/// GENIND OBJECT //////////
```

```
// 57 individuals; 18 loci; 220 alleles; size: 94.5 kb
```

```
// Basic content
```

```
@tab: 57 x 220 matrix of allele counts
```

```
@loc.n.all: number of alleles per locus (range: 5-20)
```

```
@loc.fac: locus factor for the 220 columns of @tab
```

```
@all.names: list of allele names for each locus
```

```
@ploidy: ploidy of each individual (range: 2-2)
```

```
@type: codom
```

```
@call: read.genalex(genalex = "GenalexAsian.csv", ploidy = 2,  
  geo = FALSE, region = FALSE, genclone = FALSE)
```

```
// optional content
```

```
@pop: population of each individual (group size range: 5-21)
```

```
@strata: a data frame with 1 columns ( Pop )
```

Doing Clone correction

```
``{r}  
missingno(tks, type = "loci", cutoff = 0.99, quiet = FALSE,  
          freq = FALSE)  
setPop(tks) <- ~Pop  
tks<-clonecorrect(tks, strata = ~Pop, combine = FALSE, keep = 1)  
tks  
``
```

No loci with missing values above 99% found.

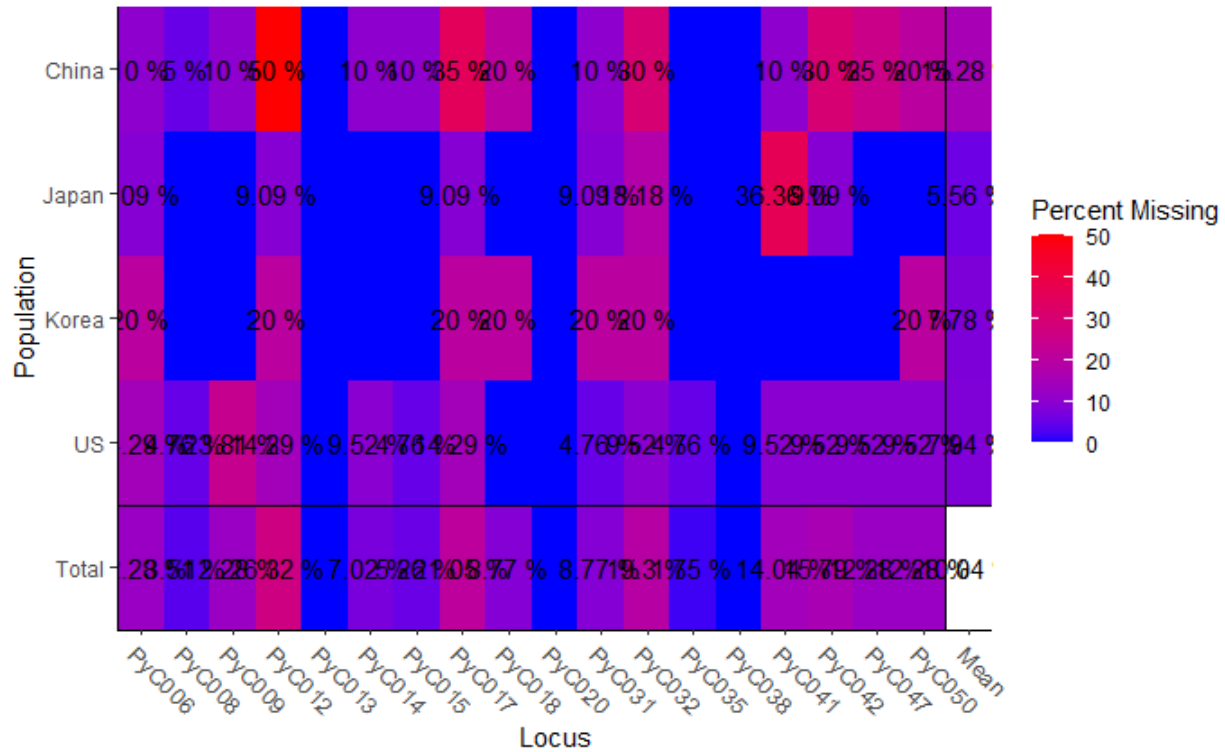
Displaying the class of datafile

```
``{r}  
class(tks)  
``
```

```
[1] "genind"  
attr(,"package")  
[1] "adegenet"
```

Displaying information about loci and populations

```
""{r}
info_table(tks, plot = TRUE)
""
```



Basic statistics

```
""{r}
summary(tks)
""
```

```
// Number of individuals: 57
// Group sizes: 20 11 5 21
// Number of alleles per locus: 10 16 15 13 14 14 18 20 16 13 20 6 13 6 8 7 6
5
// Number of alleles per group: 149 107 46 134
// Percentage of missing data: 10.04 %
// Observed heterozygosity: 0.18 0.47 0.24 0.12 0.42 0.23 0.11 0.71 0.73 0.4
0.77 0.09 0.43 0.95 0.29 0.06 0.24 0.04
// Expected heterozygosity: 0.72 0.86 0.84 0.88 0.83 0.85 0.86 0.91 0.9 0.79
0.91 0.73 0.85 0.76 0.8 0.8 0.75 0.4
```

Getting per locus information

```
```{r}
```

```
BS <- basic.stats(tks)
```

```
BS
```

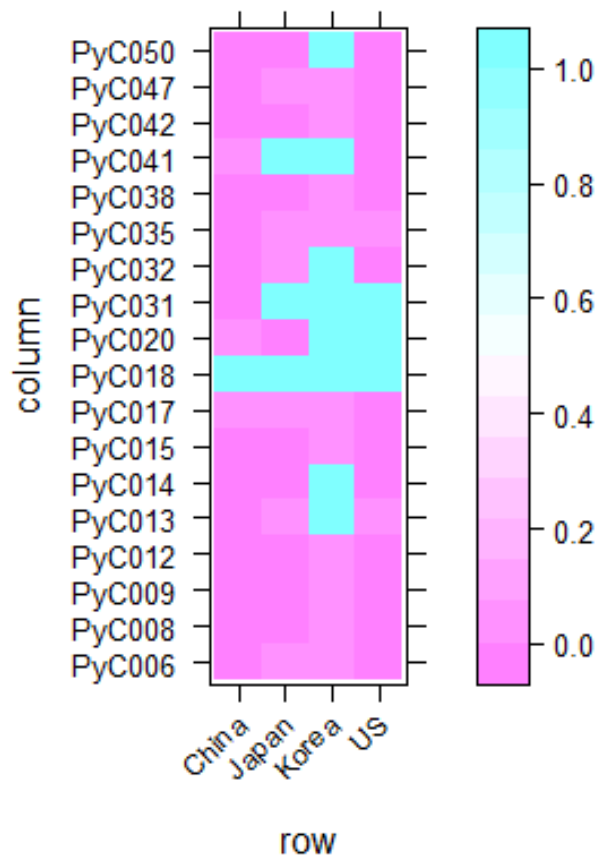
```
```
```

```
$perloc
      Ho      Hs      Ht      Dst      Htp      Dstp      Fst      Fstp      Fis
Dest
Pyc006 0.1250 0.6377 0.6919 0.0542 0.7100 0.0723 0.0784 0.1018 0.8040
0.1996
Pyc008 0.3596 0.7058 0.8867 0.1809 0.9470 0.2412 0.2040 0.2547 0.4904
0.8198
Pyc009 0.2458 0.8606 0.8460 -0.0146 0.8411 -0.0194 -0.0172 -0.0231 0.7143 -
0.1392
Pyc012 0.0806 0.7971 0.9160 0.1189 0.9556 0.1586 0.1298 0.1659 0.8989
0.7814
Pyc013 0.4348 0.7552 0.8473 0.0921 0.8779 0.1227 0.1087 0.1398 0.4242
0.5014
Pyc014 0.2072 0.7201 0.8704 0.1503 0.9205 0.2004 0.1727 0.2177 0.7122
0.7159
Pyc015 0.1187 0.6916 0.8876 0.1960 0.9529 0.2613 0.2208 0.2742 0.8284
0.8473
Pyc017 0.5983 0.8403 0.8954 0.0551 0.9138 0.0735 0.0616 0.0804 0.2880
0.4602
Pyc018 0.6518 0.8485 0.9063 0.0578 0.9256 0.0771 0.0638 0.0832 0.2319
0.5087
Pyc020 0.3872 0.7488 0.7864 0.0375 0.7989 0.0500 0.0477 0.0626 0.4829
0.1993
Pyc031 0.8389 0.8707 0.9263 0.0556 0.9448 0.0741 0.0600 0.0785 0.0365
0.5734
Pyc032 0.0913 0.5220 0.7302 0.2081 0.7996 0.2775 0.2851 0.3471 0.8252
0.5806
Pyc035 0.4091 0.8422 0.8785 0.0362 0.8905 0.0483 0.0413 0.0543 0.5143
0.3063
Pyc038 0.9318 0.7585 0.7505 -0.0080 0.7478 -0.0107 -0.0107 -0.0143 -0.2286 -
0.0441
Pyc041 0.2938 0.8132 0.8189 0.0056 0.8207 0.0075 0.0069 0.0091 0.6387
0.0402
Pyc042 0.0560 0.6428 0.8308 0.1879 0.8934 0.2506 0.2262 0.2805 0.9129
0.7016
Pyc047 0.1770 0.6775 0.7646 0.0871 0.7936 0.1161 0.1139 0.1463 0.7387
0.3599
Pyc050 0.0312 0.3485 0.3816 0.0330 0.3926 0.0441 0.0866 0.1122 0.9103
0.0676

$overall
      Ho      Hs      Ht      Dst      Htp      Dstp      Fst      Fstp      Fis      Dest
0.3355 0.7267 0.8120 0.0852 0.8404 0.1136 0.1050 0.1352 0.5384 0.4158
```

Testing Hardy-Weinberg Equilibrium

```
```{r}
hw.test(tks, B = 1000) #Rough test. Pr.exact = 0 -> Locus not in HWE.
(tks.HWE <- hw.test(tks, B = 1000)) # performs 1000 permutations
(tks.HWE.pop <- seppop(tks) %>% lapply(hw.test, B = 1000)) #by population
(tks.HWE.mat <- sapply(tks.HWE.pop, "[", i = TRUE, j = 3)) # Take the third column with all rows
alpha <- 0.05
tksHWE.mat <- tks.HWE.mat
tksHWE.mat[tksHWE.mat > alpha] <- 1
levelplot(t(tksHWE.mat), scales=list(x=list(rot=43)))
```
```



#Looking how uninformatic loci are

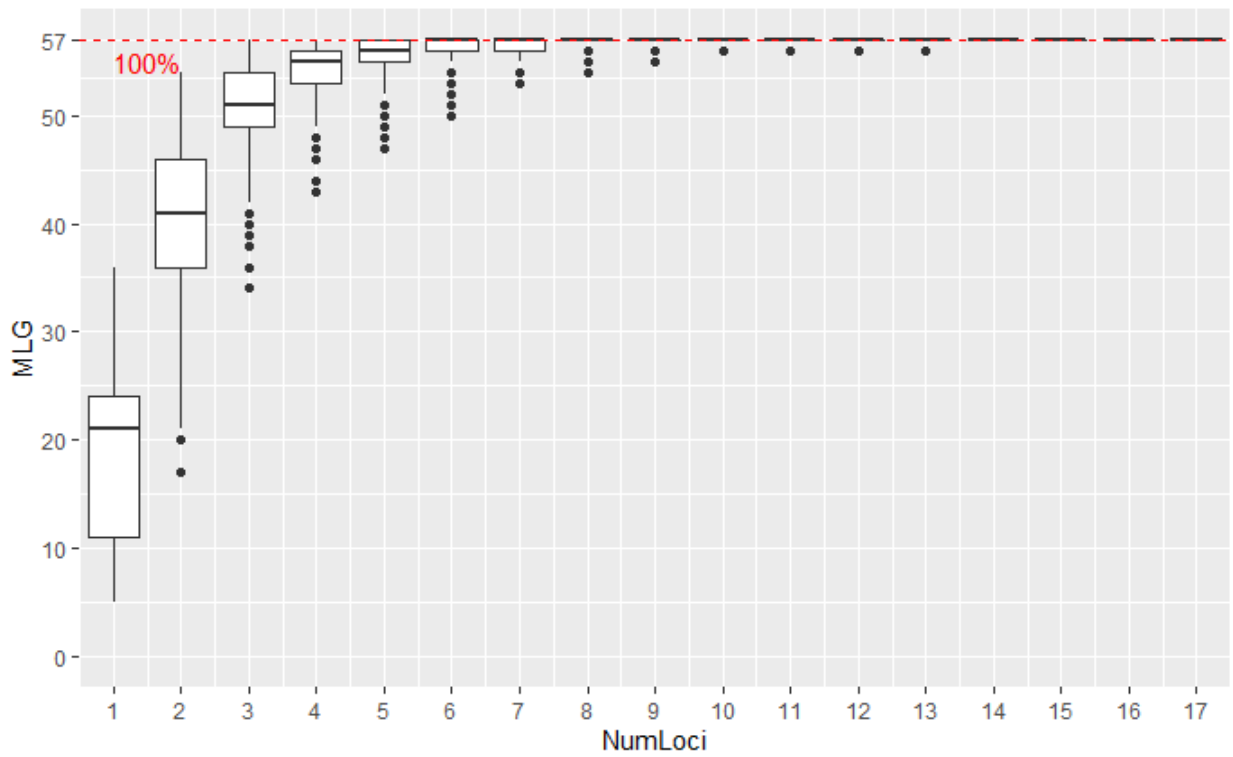
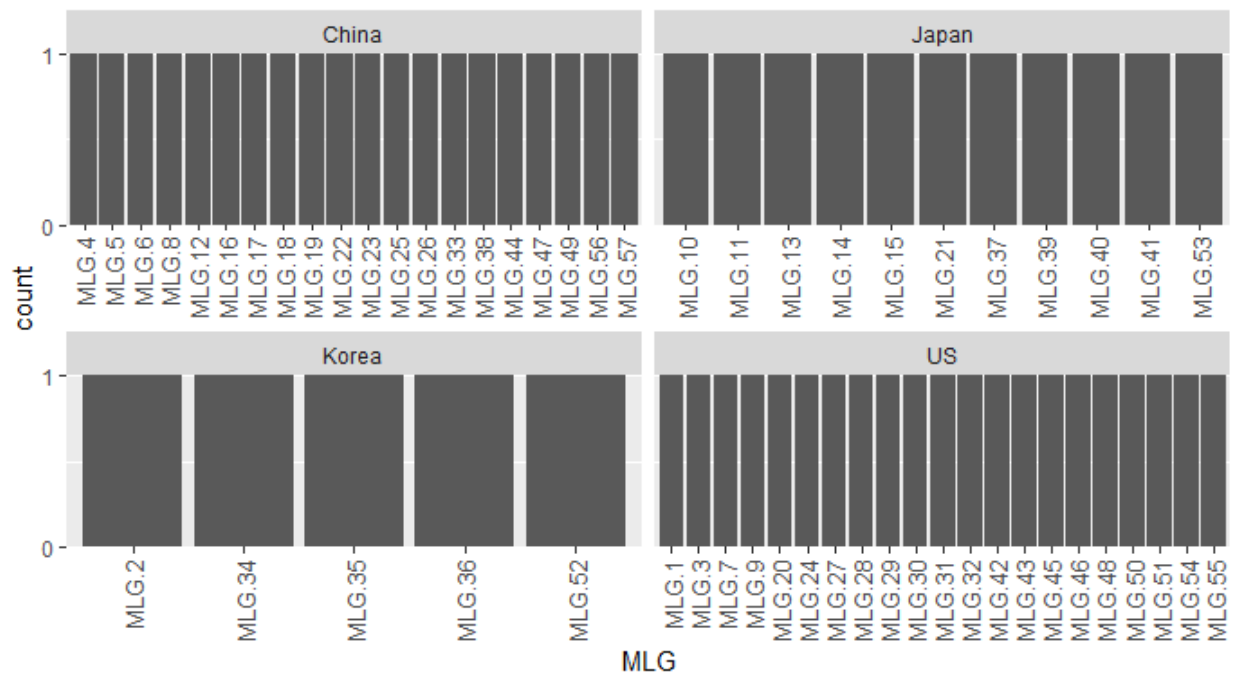
```
``{r}  
nLoc(tks)  
i_tks <- informloci(tks)  
``
```

```
[1] 18  
cutoff value: 3.50877192982456 % ( 2 samples ).  
MAF          : 0.01  
All sites polymorphic
```

#Multi Locus Genotype accumulation curve

```
``{r}  
gac <- genotype_curve(tks, sample = 1000, quiet = TRUE)  
gac  
P.tab<-mlg.table(tks)
```

Data: tks
N = 57 MLG = 57



#The index of association

```
```{r}

tk2 <- popsub(tks)

set.seed(1044)

ia(tk2, hist = TRUE, sample = 999)

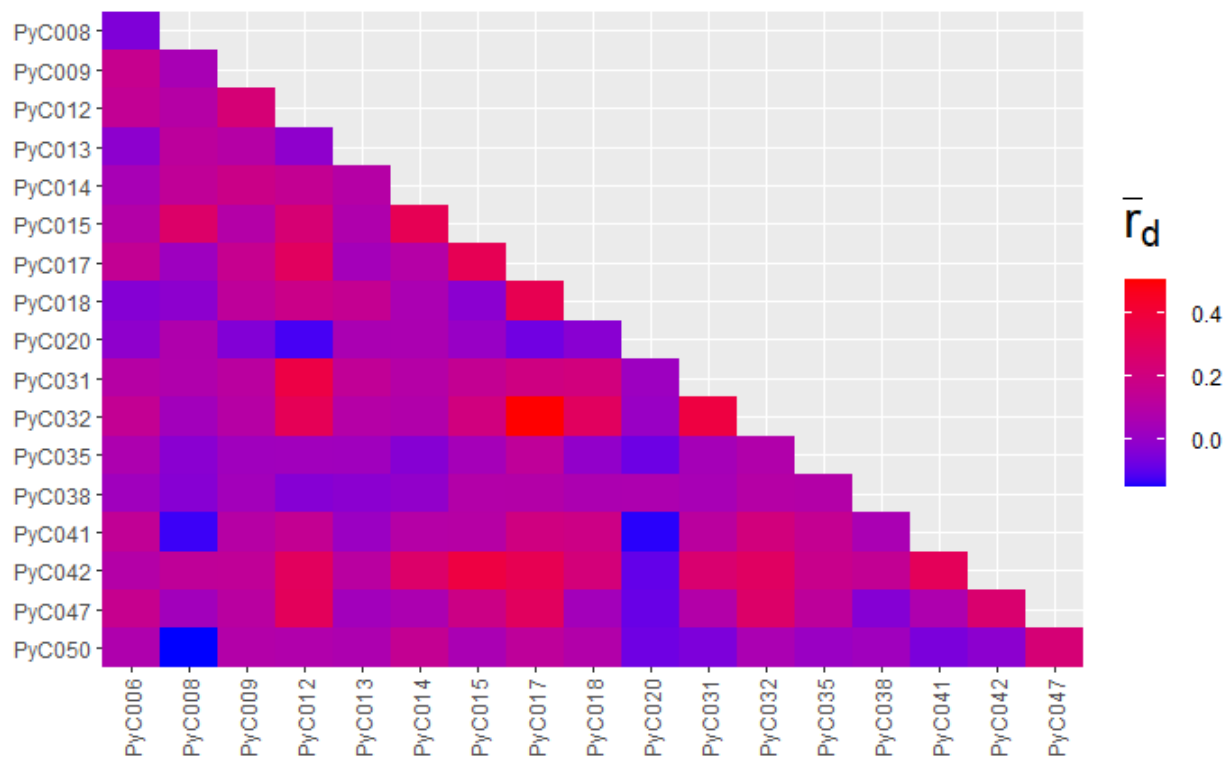
```
```

Completed after 6 m

| Ia | p.Ia | rbarD | p.rD |
|-----------|-----------|-----------|-----------|
| 1.9201013 | 0.1180000 | 0.1152366 | 0.1130000 |

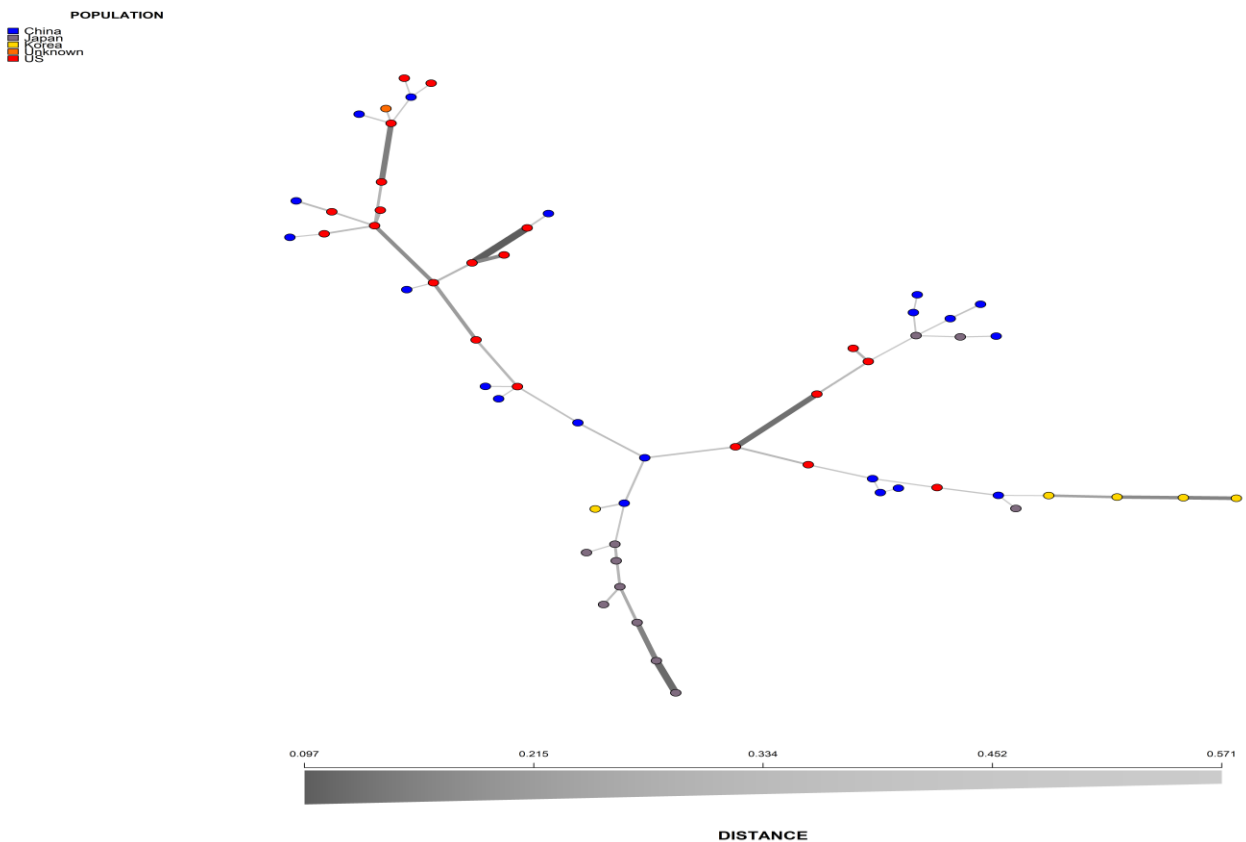
#Pairwise LD over all loci

```
``{r}
tkspair <- tks %>% clonecorrect(strata = ~Pop) %>% pair.ia
plot(tkspair, limits = plotrange, label=TRUE, low = "orange", high = "purple")
``
```



#Minimum spanning network

```
```{r}
lmsn()
tks_sub <- popsub(tks, blacklist = character(0))
tks_dist <- diss.dist(tks_sub, percent = FALSE, mat = FALSE)
min_span_net <- poppr.msn(tks_sub, tks_dist, showplot = FALSE, include.ties = TRUE)
set.seed(69)
plot_poppr_msn(tks, min_span_net, inds = "ALL", mlg = FALSE, gadj = 6, nodescale = 10,
palette = seasun, cutoff = NULL, quantiles = FALSE, beforecut = TRUE, pop.leg = TRUE,
size.leg = FALSE, scale.leg = TRUE, layfun = igraph::layout_nicely)
```
```



```
#AMOVA
```

```
```{r}

tkc <- as.genclone(tkc)

tkc

table (strata(tkc, ~Pop))

tkcamova <- poppr.amova(tkc, ~Pop)

tkcamova

```
```

This is a genclone object

Genotype information:

57 original multilocus genotypes
57 diploid individuals
18 codominant loci

Population information:

1 stratum - Pop
4 populations defined - China, Japan, Korea, US
China Japan Korea US
20 11 5 21

Found 1215 missing values.

13 loci contained missing values greater than 5%

Removing 13 loci: Pyc006, Pyc009, Pyc012, Pyc014, Pyc015, Pyc017, Pyc018,
Pyc031, Pyc032, Pyc041, Pyc042, Pyc047, Pyc050

Distance matrix is non-euclidean.

Using quasieucldid correction method. See ?quasieucldid for details.

\$call

ade4::amova(samples = xtab, distances = xdist, structures = xstruct)

\$results

| | Df | Sum Sq | Mean Sq |
|----------------------------|-----|-----------|-----------|
| Between Pop | 3 | 36.92288 | 12.307627 |
| Between samples within Pop | 53 | 280.15363 | 5.285917 |
| Within samples | 57 | 154.33570 | 2.707644 |
| Total | 113 | 471.41221 | 4.171789 |

\$componentsofcovariance

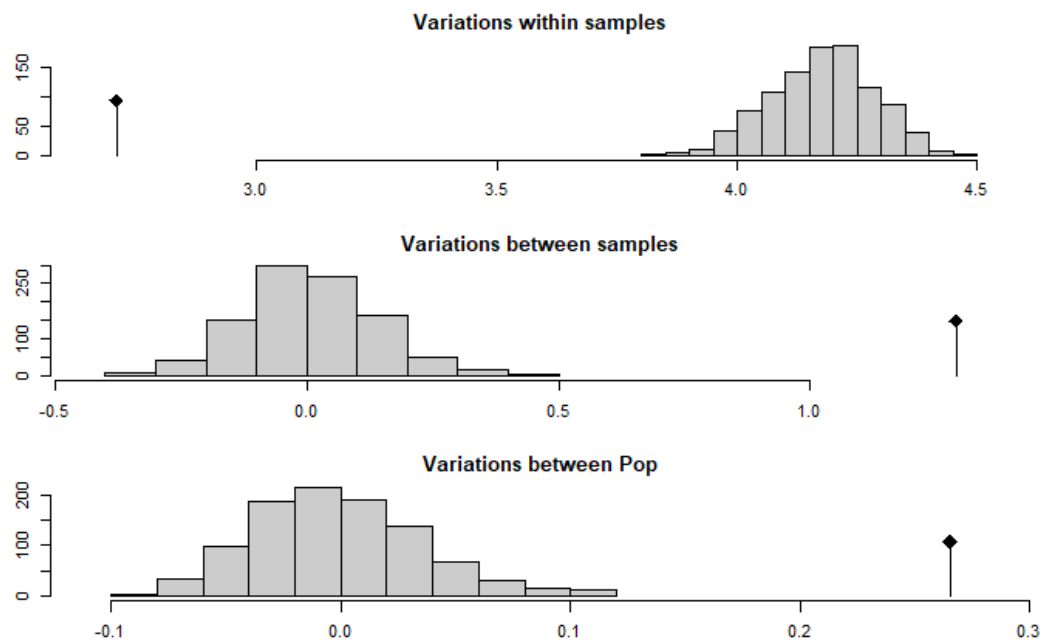
| | | Sigma | % |
|------------------|----------------------------|-----------|------------|
| Variations | Between Pop | 0.2654095 | 6.227067 |
| Variations | Between samples within Pop | 1.2891368 | 30.245877 |
| Variations | within samples | 2.7076439 | 63.527056 |
| Total variations | | 4.2621902 | 100.000000 |

\$statphi

| | Phi |
|-------------------|------------|
| Phi-samples-total | 0.36472944 |
| Phi-samples-Pop | 0.32254378 |
| Phi-Pop-total | 0.06227067 |

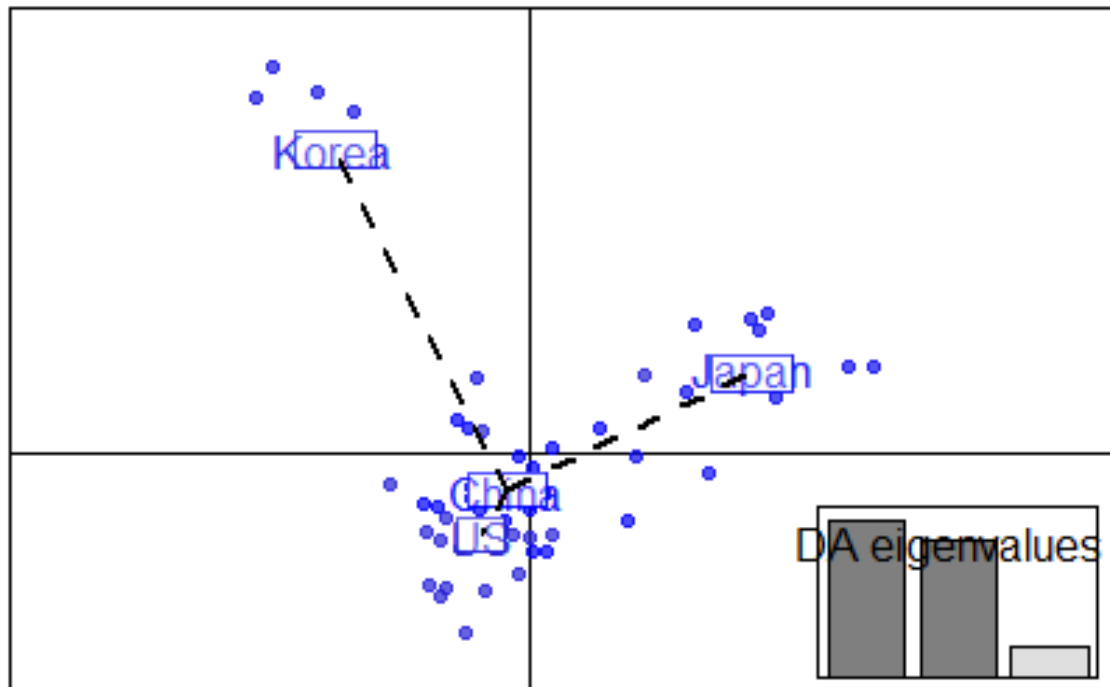
#Significance testing

```
``{r}
set.seed(1999)
tkssignif <- randtest(tksamova, nrepet = 999)
plot(tkssignif)
tkssignif
``
```



#Discriminant analysis of principal components (DAPC)

```
``{r}
l.col <- season(40)
pop(tks) <- tks$pop
dapc.tks <- dapc(tks, var.contrib = TRUE, scale = FALSE, n.pca = 19, n.da = nPop(tks) - 1)
scatter(dapc.tks, col=l.col, cell = 0, cstar = 0, mstree = TRUE, lwd = 2, lty = 2)
``
```



#DAPC cross-validation

```

```{r}
set.seed(1044)

pramx <- xvalDapc(tab(tks, NA.method = "mean"), pop(tks))

system.time(pramx <- xvalDapc(tab(tks, NA.method = "mean"), pop(tks),
 n.pca = 5:25, n.rep = 10,
 parallel = "multicore", ncpus = 4L))

names(pramx)
pramx[-1]

max(pramx$`Mean Successful Assignment by Number of PCs of PCA`)

scatter(dapc.tks, cex = 2, label.inds = FALSE, col = l.col, legend = TRUE, clabel = TRUE, posi.legend =
"bottomleft", scree.pca = FALSE, scree.da = TRUE, posi.pca = "topleft", posi.da = "bottomright", cleg =
0.75, xax = 1, yax = 2, inset.solid = 1, cstar=0)

scatter(dapc.tks, cex = 2, label.inds = NULL, col = l.col, legend = FALSE, clabel = TRUE, posi.legend = "topleft",
scree.pca = FALSE, scree.da = TRUE, posi.pca = "topright", posi.da = "bottomright", cleg = 0.75, xax = 1,
yax = 2, inset.solid = 1, cstar=0)

contrib <- loadingplot(dapc.tks$var.contr, axis = 2, thres = 0.055, lab.jitter = 1)

```

```

