AMOVAs

This notebook contains results of AMOVA testing. All code and comment is from Eleni Petrou’s Evernote *012a: AMOVA: filtered haplotypes & SNP - BC all populations*

First, install / load packages

install.packages("poppr", repos = "http://cran.us.r-project.org")

## Installing package into 'C:/Users/Mary Fisher/Documents/R/win-library/3.4'  
## (as 'lib' is unspecified)

## package 'poppr' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Mary Fisher\AppData\Local\Temp\RtmpaEfr6E\downloaded\_packages

library(poppr, lib.loc="~/R/win-library/3.4", warn.conflicts = FALSE, quietly=TRUE)

## Warning: package 'poppr' was built under R version 3.4.4

## Warning: package 'adegenet' was built under R version 3.4.4

## Warning: package 'ade4' was built under R version 3.4.4

##   
## /// adegenet 2.1.1 is loaded ////////////  
##   
## > overview: '?adegenet'  
## > tutorials/doc/questions: 'adegenetWeb()'   
## > bug reports/feature requests: adegenetIssues()

## This is poppr version 2.7.1. To get started, type package?poppr  
## OMP parallel support: available

library(pegas, lib.loc="~/R/win-library/3.4", warn.conflicts = FALSE, quietly=TRUE)

## Warning: package 'pegas' was built under R version 3.4.2

## Warning: package 'ape' was built under R version 3.4.3

The implementation of AMOVA in poppr requires two very basic components: (1) A distance matrix derived from the data and (2) a separate table used to partition the data into different stratifications. The distance matrix can be calculated using any distance as long as it is euclidean.

library(poppr, lib.loc="~/R/win-library/3.4", warn.conflicts = FALSE, quietly=TRUE)  
library(pegas, lib.loc="~/R/win-library/3.4", warn.conflicts = FALSE, quietly=TRUE)  
  
#read in genetic data and hierarchical strata files.   
setwd("D:/Pacific cod/DataAnalysis/PCod-Compare-repo/analyses/R")  
  
data\_all\_loci <-read.genepop("../../stacks\_b8\_wgenome\_r05/batch\_8\_final\_filtered\_aligned\_genepop\_eastwest.gen")

##   
## Converting data from a Genepop .gen file to a genind object...   
##   
##   
## File description: PCod Combined data set. batch 8 final filtered genepop. Subsample of aligned loci 2018-03-25 10:29:49   
##   
## ...done.

levels(pop(data\_all\_loci))

## [1] "PO031715\_24\_92bp" "GEO020414\_9\_300\_92bp" "NA021015\_30\_92bp"   
## [4] "YS\_121316\_29\_92bp" "JB021108\_48.1\_92bp" "BOR07\_22\_92bp"   
## [7] "KOD03\_097" "AD06\_048" "WC05\_048"   
## [10] "HS04\_048" "PWS12\_134" "UP03\_048"

# Read in a dataframe containing information about the hierarchial levels  
my\_strata<- read.delim("../AMOVA/EastWest\_Strata\_for\_AMOVA.txt")  
head(my\_strata)

## Individual site population region  
## 1 PO010715\_02\_92bp Pohang KoreaSouth West  
## 2 PO010715\_04\_92bp Pohang KoreaSouth West  
## 3 PO010715\_06.1\_92bp Pohang KoreaSouth West  
## 4 PO010715\_07\_92bp Pohang KoreaSouth West  
## 5 PO010715\_08.1\_92bp Pohang KoreaSouth West  
## 6 PO010715\_10.1\_92bp Pohang KoreaSouth West

# assign those levels as strata to the genind object  
strata(data\_all\_loci) <- my\_strata  
  
# Change the format of your data from a genind to a genclone object  
my\_data <- as.genclone(data\_all\_loci)  
my\_data

##   
## This is a genclone object  
## -------------------------  
## Genotype information:  
##   
## 479 original multilocus genotypes   
## 479 diploid individuals  
## 4286 codominant loci  
##   
## Population information:  
##   
## 4 strata - Individual, site, population, region  
## 12 populations defined -   
## PO031715\_24\_92bp, GEO020414\_9\_300\_92bp, NA021015\_30\_92bp, ..., HS04\_048, PWS12\_134, UP03\_048

# View some hierarchical levels in the data set  
table(strata(my\_data, ~population))

##   
## KoreaSouth KoreaWest Kodiak Adak WashCoast   
## 173 46 43 41 41   
## HecateStrait PWSound UnimakPass   
## 46 47 42

table(strata(my\_data, ~region, combine = FALSE))

##   
## West East   
## 219 260

table(strata(my\_data, ~site))

##   
## Pohang Geoje Namhae Jinhae Bay YSBlock   
## 31 50 11 81 25   
## Boryeong Kodiak Adak WashCoast HecateStrait   
## 21 43 41 41 46   
## PWSound UnimakPass   
## 47 42

This implementation accepts any number of hierarchical levels.

The formula must be of the form d, ~ A/B/… where d is a genclone or genind object, and A, B, etc, are the hierarchical levels from the highest to the lowest one.

Any number of levels is accepted, so specifying d ~ A will simply test for population differentiation.

The poppr.amova function is a wrapper script for amova. It calculates a pairwise distance matrix by default.

### PEGAS AMOVAS

#AMOVA considering regions  
library(poppr, lib.loc="~/R/win-library/3.4", warn.conflicts = FALSE, quietly=TRUE)  
  
amova1 <- poppr.amova(my\_data, ~region, within = FALSE, method = "pegas", nperm = 1000)

##   
## Found 156584 missing values.  
##   
## 1060 loci contained missing values greater than 5%  
##   
## Removing 1060 loci: 5\_36, 75\_48, 88\_61, 91\_88, 92\_41, 121\_62,  
## 139\_71, 174\_91, 183\_55, 192\_82, 198\_50, 240\_71, 244\_78, 265\_28,  
## 331\_54, 373\_59, 381\_80, 388\_77, 410\_60, 415\_88, 440\_43, 445\_54,  
## 481\_87, 525\_48, 548\_43, 554\_63, 571\_34, 663\_90, 691\_74, 739\_62,  
## 769\_63, 792\_76, 805\_81, 825\_91, 844\_47, 885\_35, 888\_53, 901\_90,  
## 926\_29, 927\_91, 940\_90, 962\_65, 964\_90, 966\_61, 1007\_79, 1036\_89,  
## 1037\_64, 1048\_91, 1066\_48, 1082\_35, 1104\_88, 1159\_43, 1197\_34,  
## 1201\_47, 1205\_46, 1218\_29, 1228\_69, 1253\_90, 1323\_80, 1345\_80,  
## 1407\_49, 1451\_41, 1490\_77, 1499\_87, 1568\_68, 1581\_90, 1596\_64,  
## 1635\_89, 1712\_49, 1730\_80, 1770\_63, 1773\_87, 1833\_86, 1881\_78,  
## 1911\_85, 1961\_60, 1962\_65, 1963\_76, 1975\_66, 2033\_59, 2075\_67,  
## 2110\_46, 2154\_35, 2160\_85, 2206\_91, 2319\_62, 2346\_63, 2349\_51,  
## 2370\_29, 2429\_57, 2497\_89, 2498\_87, 2539\_46, 2550\_72, 2554\_89,  
## 2601\_85, 2607\_77, 2646\_90, 2672\_52, 2690\_88, 2706\_52, 2810\_38,  
## 2820\_60, 2844\_90, 2850\_85, 2950\_90, 2967\_65, 2973\_81, 3000\_77,  
## 3061\_28, 3062\_55, 3095\_80, 3111\_90, 3129\_58, 3175\_63, 3196\_76,  
## 3207\_79, 3244\_91, 3266\_37, 3280\_86, 3304\_87, 3307\_84, 3308\_52,  
## 3324\_42, 3338\_78, 3339\_82, 3343\_36, 3356\_90, 3378\_60, 3406\_49,  
## 3458\_84, 3473\_49, 3499\_78, 3505\_91, 3523\_34, 3541\_56, 3585\_53,  
## 3659\_90, 3728\_82, 3734\_80, 3754\_66, 3797\_53, 3799\_66, 3828\_44,  
## 3853\_41, 3866\_71, 3885\_54, 3960\_48, 3962\_63, 3991\_53, 4000\_51,  
## 4002\_69, 4016\_45, 4045\_60, 4062\_70, 4070\_63, 4100\_83, 4126\_88,  
## 4146\_80, 4150\_59, 4188\_69, 4214\_47, 4217\_74, 4226\_66, 4235\_65,  
## 4242\_79, 4251\_41, 4257\_74, 4285\_37, 4317\_34, 4330\_90, 4332\_84,  
## 4344\_87, 4368\_52, 4373\_68, 4402\_81, 4440\_42, 4473\_82, 4489\_82,  
## 4492\_63, 4507\_36, 4550\_33, 4556\_80, 4568\_58, 4570\_57, 4572\_41,  
## 4575\_53, 4582\_89, 4599\_66, 4615\_45, 4620\_80, 4629\_62, 4689\_53,  
## 4727\_66, 4818\_29, 4849\_91, 4878\_36, 4891\_34, 4905\_55, 4958\_56,  
## 5001\_69, 5016\_67, 5071\_32, 5079\_75, 5137\_30, 5171\_50, 5196\_90,  
## 5203\_57, 5227\_52, 5273\_82, 5413\_67, 5417\_48, 5431\_34, 5464\_49,  
## 5465\_81, 5534\_67, 5538\_91, 5562\_45, 5636\_73, 5654\_86, 5668\_75,  
## 5672\_63, 5673\_62, 5688\_87, 5692\_63, 5707\_76, 5738\_82, 5739\_51,  
## 5746\_57, 5748\_65, 5765\_84, 5766\_67, 5770\_90, 5807\_74, 5813\_49,  
## 5834\_40, 5836\_87, 5882\_73, 5891\_63, 5892\_84, 5902\_53, 5925\_45,  
## 5945\_29, 5946\_56, 5952\_85, 5966\_58, 5984\_73, 6126\_83, 6158\_79,  
## 6203\_88, 6275\_59, 6298\_55, 6312\_47, 6318\_62, 6319\_49, 6333\_65,  
## 6368\_55, 6384\_58, 6388\_28, 6426\_86, 6432\_60, 6441\_39, 6454\_85,  
## 6467\_68, 6540\_52, 6565\_44, 6567\_50, 6583\_74, 6609\_40, 6638\_64,  
## 6735\_73, 6802\_82, 6810\_52, 6886\_44, 6890\_83, 6938\_37, 6970\_91,  
## 6988\_71, 6993\_33, 7023\_32, 7058\_31, 7081\_84, 7083\_74, 7117\_67,  
## 7142\_63, 7144\_74, 7151\_56, 7192\_67, 7230\_43, 7283\_54, 7286\_64,  
## 7313\_69, 7335\_80, 7384\_89, 7412\_67, 7434\_88, 7450\_31, 7458\_83,  
## 7465\_45, 7501\_37, 7509\_89, 7514\_90, 7522\_77, 7603\_45, 7604\_75,  
## 7605\_40, 7620\_90, 7637\_38, 7647\_91, 7749\_60, 7754\_29, 7769\_76,  
## 7789\_91, 7791\_84, 7806\_50, 7807\_85, 7875\_29, 7879\_28, 7952\_34,  
## 7992\_49, 8023\_56, 8033\_75, 8048\_91, 8055\_51, 8064\_38, 8095\_30,  
## 8130\_65, 8152\_79, 8160\_81, 8190\_38, 8191\_90, 8198\_52, 8199\_45,  
## 8204\_48, 8205\_41, 8217\_47, 8218\_85, 8231\_41, 8241\_29, 8246\_32,  
## 8267\_80, 8288\_52, 8315\_86, 8322\_68, 8327\_33, 8334\_64, 8414\_44,  
## 8448\_69, 8484\_54, 8507\_76, 8547\_63, 8572\_68, 8586\_81, 8649\_72,  
## 8704\_87, 8746\_52, 8803\_51, 8804\_51, 8842\_54, 8849\_29, 8915\_80,  
## 8948\_43, 8955\_34, 9050\_81, 9091\_57, 9156\_59, 9159\_60, 9166\_67,  
## 9192\_71, 9210\_32, 9213\_55, 9215\_76, 9218\_71, 9247\_56, 9280\_40,  
## 9347\_41, 9350\_48, 9363\_69, 9369\_73, 9449\_47, 9461\_76, 9489\_77,  
## 9502\_68, 9506\_84, 9563\_42, 9565\_79, 9596\_54, 9604\_56, 9606\_48,  
## 9610\_77, 9640\_91, 9655\_40, 9673\_88, 9681\_71, 9731\_66, 9738\_31,  
## 9740\_72, 9801\_74, 9853\_60, 9872\_90, 9894\_30, 9940\_91, 9966\_58,  
## 10006\_90, 10015\_56, 10076\_78, 10086\_28, 10092\_34, 10134\_89,  
## 10139\_43, 10151\_59, 10166\_68, 10177\_88, 10199\_57, 10232\_91,  
## 10240\_37, 10260\_31, 10285\_38, 10291\_62, 10292\_74, 10295\_35,  
## 10304\_47, 10307\_46, 10316\_84, 10392\_45, 10396\_90, 10401\_33,  
## 10456\_83, 10476\_80, 10488\_53, 10501\_61, 10560\_91, 10648\_58,  
## 10662\_51, 10665\_88, 10698\_76, 10715\_59, 10718\_38, 10738\_48,  
## 10745\_28, 10758\_50, 10806\_85, 10885\_38, 10913\_90, 10928\_90,  
## 10933\_54, 10951\_60, 10960\_37, 10991\_53, 10995\_83, 11008\_50,  
## 11019\_91, 11049\_72, 11052\_86, 11069\_35, 11133\_72, 11140\_28,  
## 11170\_81, 11209\_31, 11211\_43, 11214\_87, 11225\_70, 11244\_90,  
## 11254\_36, 11260\_80, 11313\_30, 11319\_78, 11343\_35, 11348\_87,  
## 11370\_42, 11415\_86, 11437\_45, 11483\_64, 11520\_73, 11565\_35,  
## 11579\_41, 11670\_58, 11687\_81, 11736\_75, 11738\_50, 11749\_63,  
## 11783\_70, 11818\_32, 11843\_82, 11846\_91, 11849\_85, 11859\_68,  
## 11878\_34, 11943\_89, 11978\_64, 11988\_77, 11990\_79, 11992\_77,  
## 12017\_58, 12049\_68, 12182\_89, 12201\_32, 12205\_45, 12206\_91,  
## 12233\_73, 12242\_79, 12254\_48, 12288\_34, 12322\_66, 12331\_56,  
## 12334\_57, 12401\_87, 12430\_90, 12431\_39, 12479\_28, 12508\_85,  
## 12534\_32, 12544\_88, 12554\_58, 12555\_28, 12564\_38, 12567\_36,  
## 12600\_87, 12636\_76, 12641\_38, 12649\_56, 12668\_43, 12670\_30,  
## 12686\_65, 12711\_44, 12720\_88, 12747\_73, 12762\_73, 12768\_72,  
## 12770\_32, 12779\_65, 12793\_90, 12810\_70, 12823\_53, 12824\_28,  
## 12869\_41, 12889\_52, 12897\_87, 12916\_31, 12946\_56, 12972\_32,  
## 13010\_57, 13020\_73, 13041\_44, 13063\_50, 13090\_73, 13104\_71,  
## 13124\_84, 13129\_52, 13135\_70, 13167\_78, 13176\_59, 13192\_68,  
## 13237\_67, 13241\_67, 13276\_34, 13286\_89, 13297\_68, 13308\_59,  
## 13318\_44, 13322\_45, 13411\_68, 13422\_71, 13458\_69, 13461\_41,  
## 13478\_49, 13518\_85, 13544\_64, 13600\_77, 13631\_43, 13656\_80,  
## 13693\_54, 13712\_51, 13724\_29, 13725\_73, 13756\_42, 13796\_49,  
## 13892\_91, 13908\_70, 13909\_46, 13911\_47, 13935\_70, 13979\_39,  
## 13980\_49, 13982\_91, 14020\_91, 14070\_62, 14122\_74, 14127\_67,  
## 14140\_72, 14143\_82, 14199\_75, 14236\_41, 14247\_40, 14268\_77,  
## 14287\_37, 14298\_40, 14300\_53, 14371\_58, 14421\_43, 14426\_91,  
## 14473\_45, 14475\_83, 14491\_43, 14495\_31, 14502\_89, 14506\_91,  
## 14514\_84, 14522\_35, 14546\_28, 14554\_78, 14559\_64, 14572\_43,  
## 14592\_73, 14593\_87, 14594\_89, 14596\_74, 14620\_83, 14635\_44,  
## 14641\_46, 14661\_84, 14698\_30, 14710\_28, 14719\_79, 14740\_63,  
## 14754\_90, 14758\_68, 14760\_53, 14762\_75, 14764\_88, 14804\_85,  
## 14824\_50, 14912\_39, 14932\_80, 14934\_49, 14999\_55, 15024\_34,  
## 15098\_61, 15217\_73, 15267\_42, 15279\_65, 15293\_30, 15298\_89,  
## 15308\_30, 15311\_57, 15351\_37, 15354\_74, 15373\_43, 15393\_58,  
## 15398\_91, 15419\_83, 15462\_87, 15474\_35, 15488\_89, 15526\_64,  
## 15567\_66, 15568\_62, 15595\_70, 15604\_37, 15639\_33, 15646\_86,  
## 15652\_51, 15708\_90, 15735\_82, 15749\_79, 15752\_61, 15763\_64,  
## 15869\_36, 15872\_90, 15881\_80, 16005\_80, 16009\_43, 16011\_36,  
## 16036\_81, 16062\_57, 16153\_61, 16165\_77, 16175\_83, 16186\_29,  
## 16187\_71, 16190\_60, 16208\_87, 16215\_64, 16264\_67, 16360\_34,  
## 16370\_81, 16396\_68, 16456\_30, 16520\_58, 16535\_85, 16616\_66,  
## 16620\_53, 16621\_28, 16695\_39, 16697\_38, 16742\_38, 16745\_48,  
## 16789\_66, 16800\_85, 16803\_65, 16891\_82, 16897\_34, 16904\_48,  
## 16933\_53, 16944\_51, 16963\_50, 16968\_34, 16982\_91, 16989\_47,  
## 17001\_56, 17040\_59, 17046\_39, 17074\_74, 17084\_37, 17097\_62,  
## 17102\_75, 17104\_66, 17116\_75, 17118\_90, 17165\_30, 17182\_72,  
## 17190\_31, 17237\_83, 17256\_55, 17298\_54, 17312\_68, 17320\_49,  
## 17322\_88, 17343\_80, 17391\_85, 17406\_47, 17461\_62, 17475\_39,  
## 17529\_70, 17533\_47, 17588\_31, 17600\_60, 17619\_28, 17625\_52,  
## 17629\_41, 17683\_38, 17782\_84, 17826\_67, 17835\_82, 17837\_29,  
## 17860\_70, 17892\_83, 17898\_53, 17899\_49, 17901\_45, 17919\_60,  
## 17934\_60, 17959\_73, 17973\_58, 18028\_60, 18043\_70, 18047\_64,  
## 18049\_36, 18094\_89, 18108\_80, 18222\_61, 18236\_85, 18243\_78,  
## 18249\_37, 18297\_30, 18328\_65, 18330\_49, 18355\_52, 18401\_43,  
## 18438\_45, 18482\_28, 18523\_73, 18534\_48, 18558\_76, 18566\_86,  
## 18577\_79, 18583\_91, 18586\_88, 18590\_43, 18591\_38, 18633\_51,  
## 18762\_48, 18792\_76, 18796\_47, 18804\_34, 18808\_44, 18879\_56,  
## 18906\_83, 18990\_91, 19039\_89, 19070\_38, 19094\_49, 19107\_56,  
## 19146\_67, 19165\_62, 19181\_79, 19183\_90, 19193\_44, 19219\_81,  
## 19324\_32, 19329\_53, 19391\_75, 19450\_49, 19502\_50, 19509\_85,  
## 19513\_35, 19520\_38, 19543\_69, 19551\_34, 19592\_76, 19651\_48,  
## 19712\_89, 19739\_73, 19772\_39, 19784\_36, 19821\_43, 19829\_51,  
## 19854\_81, 19856\_88, 19867\_75, 19879\_89, 19881\_69, 19885\_56,  
## 19899\_55, 19937\_46, 19996\_91, 20023\_56, 20070\_33, 20081\_32,  
## 20163\_53, 20232\_89, 20242\_58, 20248\_54, 20265\_35, 20284\_87,  
## 20295\_36, 20312\_28, 20321\_44, 20354\_36, 20373\_35, 20374\_72,  
## 20473\_75, 20501\_35, 20517\_32, 20568\_56, 20578\_91, 20588\_36,  
## 20593\_54, 20604\_78, 20625\_85, 20666\_51, 20672\_73, 20689\_88,  
## 20705\_78, 20714\_70, 20715\_33, 20742\_30, 20748\_78, 20791\_58,  
## 20837\_42, 20885\_88, 20895\_28, 20923\_87, 21060\_52, 21070\_72,  
## 21119\_51, 21174\_85, 21239\_71, 21256\_66, 21277\_37, 21335\_70,  
## 21336\_39, 21360\_43, 21376\_62, 21377\_87, 21383\_69, 21403\_62,  
## 21411\_34, 21419\_81, 21470\_85, 21482\_67, 21531\_55, 21537\_40,  
## 21554\_85, 21590\_69, 21596\_52, 21611\_73, 21624\_79, 21625\_80,  
## 21647\_87, 21649\_39, 21653\_35, 21681\_83, 21686\_40, 21699\_82,  
## 21704\_83, 21721\_90, 21732\_56, 21738\_43, 21748\_28, 21785\_51,  
## 21801\_30, 21821\_80, 21833\_28, 21920\_67, 21921\_33, 21961\_33,  
## 21971\_88, 22014\_79, 22035\_63, 22068\_80, 22087\_31, 22092\_61,  
## 22242\_70, 22261\_78, 22304\_31, 22332\_59, 22349\_91, 22373\_54,  
## 22376\_41, 22410\_80, 22434\_70, 22530\_86, 22597\_90, 22615\_30,  
## 22637\_61, 22648\_45, 22694\_91, 22707\_65, 22737\_32, 22753\_87,  
## 22761\_86, 22794\_47, 22839\_63, 22862\_81, 22892\_70, 22898\_87,  
## 22913\_52, 22923\_53, 22931\_44, 22937\_74, 22941\_48, 22978\_85,  
## 22983\_49, 22984\_33, 23000\_31, 23001\_48, 23026\_49, 23066\_80,  
## 23079\_44, 23100\_30, 23111\_29, 23112\_71, 23115\_50, 23152\_80,  
## 23168\_55, 23189\_45, 23191\_74, 23214\_64, 23278\_74, 23279\_72,  
## 23313\_45, 23321\_72, 23354\_61, 23389\_81, 23413\_74, 23432\_56,  
## 23457\_29, 23487\_55, 23494\_54, 23502\_90, 23503\_79, 23528\_64,  
## 23550\_49, 23563\_35, 23643\_40, 23758\_50, 23845\_75, 23870\_81,  
## 23935\_74, 23974\_40, 23980\_84, 23990\_71, 23992\_46, 23999\_81,  
## 24005\_53, 24115\_72, 24153\_72, 24279\_79, 24334\_75, 24348\_86,  
## 24372\_49, 24389\_66, 24400\_53, 24402\_81, 24414\_56, 24444\_42,  
## 24464\_89, 24467\_66, 24591\_70, 24634\_83, 24664\_56, 24734\_83,  
## 24740\_73, 24751\_43, 24759\_47, 24796\_91, 24840\_75, 24921\_37,  
## 24940\_51, 24954\_48, 24985\_69, 25025\_77, 25052\_58, 25103\_70,  
## 25130\_80, 25135\_90, 25139\_50, 25164\_31, 25170\_36, 25208\_44,  
## 25244\_36, 25250\_29, 25271\_28, 25284\_68, 25293\_70, 25322\_53,  
## 25323\_87, 25334\_51, 25357\_31, 25365\_38, 25366\_64, 25368\_47,  
## 25372\_45, 25378\_64, 25396\_56, 25411\_39, 25416\_62, 25444\_50,  
## 25446\_74, 25450\_62, 25478\_58, 25483\_34, 25511\_84, 25516\_80,  
## 25526\_28, 25554\_66, 25555\_32, 25561\_29, 25567\_73, 25588\_29,  
## 25593\_54, 25620\_55, 25622\_82, 25624\_48, 25630\_64, 25638\_76,  
## 25641\_35, 25649\_59, 25650\_43

amova1

##   
## Analysis of Molecular Variance  
##   
## Call: pegas::amova(formula = hier, data = hierdf, nperm = nperm, is.squared = FALSE)  
##   
## SSD MSD df  
## region 65284.28 65284.2807 1  
## Error 95087.04 199.3439 477  
## Total 160371.32 335.5049 478  
##   
## Variance components:  
## sigma2 P.value  
## region 273.76 0  
## Error 199.34   
##   
## Variance coefficients:  
## a   
## 237.7453

#AMOVA considering regions and populations  
library(poppr, lib.loc="~/R/win-library/3.4", warn.conflicts = FALSE, quietly=TRUE)  
  
amova2<- poppr.amova(my\_data, ~region/population, within = FALSE, method = "pegas",   
 nperm = 1000, quiet = TRUE)  
  
amova2

##   
## Analysis of Molecular Variance  
##   
## Call: pegas::amova(formula = hier, data = hierdf, nperm = nperm, is.squared = FALSE)  
##   
## SSD MSD df  
## region 65284.281 65284.2807 1  
## population 1890.864 315.1439 6  
## Error 93196.178 197.8687 471  
## Total 160371.322 335.5049 478  
##   
## Variance components:  
## sigma2 P.value  
## region 272.749   
## population 2.433 0.999  
## Error 197.869   
##   
## Variance coefficients:  
## a b c   
## 48.20238 99.29513 237.74530

#AMOVA considering regions and populations -- switch terms for different SSD??  
library(poppr, lib.loc="~/R/win-library/3.4", warn.conflicts = FALSE, quietly=TRUE)  
  
amova2.5<- poppr.amova(my\_data, ~population/region, within = FALSE, method = "pegas",   
 nperm = 1000, quiet = TRUE)  
  
amova2.5

##   
## Analysis of Molecular Variance  
##   
## Call: pegas::amova(formula = hier, data = hierdf, nperm = nperm, is.squared = FALSE)  
##   
## SSD MSD df  
## population 67175.14 9596.4492 7  
## region 0.00 NaN 0  
## Error 93196.18 197.8687 471  
## Total 160371.32 335.5049 478  
##   
## Variance components:  
## sigma2 P.value  
## population   
## region   
## Error 197.87   
##   
## Variance coefficients:  
## a b c   
## NaN 55.50134 55.50134

### ADE4 AMOVAS

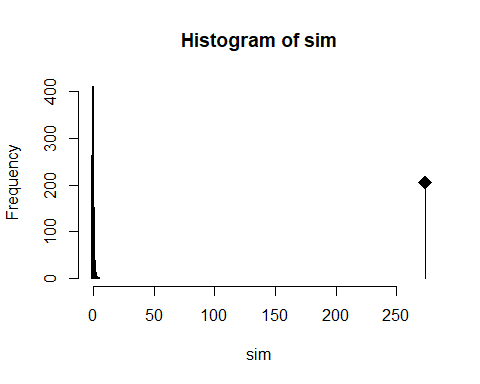
library(poppr, lib.loc="~/R/win-library/3.4", warn.conflicts = FALSE, quietly=TRUE)  
  
#AMOVA considering regions  
amova3 <- poppr.amova(my\_data, ~region, within = FALSE,   
 nperm = 1000, quiet = TRUE)  
  
amova3

## $call  
## ade4::amova(samples = xtab, distances = xdist, structures = xstruct)  
##   
## $results  
## Df Sum Sq Mean Sq  
## Between samples 1 65284.28 65284.2807  
## Within samples 477 95087.04 199.3439  
## Total 478 160371.32 335.5049  
##   
## $componentsofcovariance  
## Sigma %  
## Variations Between samples 273.7591 57.86459  
## Variations Within samples 199.3439 42.13541  
## Total variations 473.1030 100.00000  
##   
## $statphi  
## Phi  
## Phi-samples-total 0.5786459

# To test if populations are significantly different, we perform a randomization test using the function randtest() from the ade4 package.   
#This will randomly permute the sample matrices as described in (Excoffier et al., 1992).  
  
set.seed(1999)  
amova3.test <- randtest(amova3, nrepet = 999)  
amova3.test

## Monte-Carlo test  
## Call: as.randtest(sim = res, obs = sigma[1])  
##   
## Observation: 273.7591   
##   
## Based on 999 replicates  
## Simulated p-value: 0.001   
## Alternative hypothesis: greater   
##   
## Std.Obs Expectation Variance   
## 362.82738079 -0.01787485 0.56936883

plot(amova3.test)



#AMOVA considering management units  
library(poppr, lib.loc="~/R/win-library/3.4", warn.conflicts = FALSE, quietly=TRUE)  
  
amova4<- poppr.amova(my\_data, ~region/population, within = FALSE,   
 nperm = 1000, quiet = TRUE)  
  
amova4

## $call  
## ade4::amova(samples = xtab, distances = xdist, structures = xstruct)  
##   
## $results  
## Df Sum Sq Mean Sq  
## Between region 1 65284.281 65284.2807  
## Between samples Within region 6 1890.864 315.1439  
## Within samples 471 93196.178 197.8687  
## Total 478 160371.322 335.5049  
##   
## $componentsofcovariance  
## Sigma %  
## Variations Between region 272.749151 57.6574675  
## Variations Between samples Within region 2.432975 0.5143158  
## Variations Within samples 197.868742 41.8282167  
## Total variations 473.050868 100.0000000  
##   
## $statphi  
## Phi  
## Phi-samples-total 0.58171783  
## Phi-samples-region 0.01214655  
## Phi-region-total 0.57657468

set.seed(1999)  
amova4.test <- randtest(amova4, nrepet = 999)  
amova4.test

## class: krandtest lightkrandtest   
## Monte-Carlo tests  
## Call: randtest.amova(xtest = amova4, nrepet = 999)  
##   
## Number of tests: 3   
##   
## Adjustment method for multiple comparisons: none   
## Permutation number: 999   
## Test Obs Std.Obs Alter Pvalue  
## 1 Variations within samples 197.868742 -132.129426 less 0.001  
## 2 Variations between samples 2.432975 33.126884 greater 0.001  
## 3 Variations between region 272.749151 3.795503 greater 0.001

plot(amova4.test)

