

# Lab 3

Norah Saarman

2024-09-17

## Section 6.3 Worked Example

[https://bookdown.org/hhwagner1/LandGenCourse\\_book/WE\\_3.html](https://bookdown.org/hhwagner1/LandGenCourse_book/WE_3.html)

Load libraries

```
#require(adeigenet)
require(LandGenCourse)

## Loading required package: LandGenCourse

#require(pegas)
require(PopGenReport)

## Loading required package: PopGenReport
## Loading required package: knitr
## Loading required package: adegenet
## Loading required package: ade4

##
##    /// adegenet 2.1.10 is loaded //////////////////////////////////
##
##    > overview: '?adegenet'
##    > tutorials/doc/questions: 'adegenetWeb()'
##    > bug reports/feature requests: adegenetIssues()

## Registered S3 method overwritten by 'pegas':
##   method      from
##   print.amova ade4

## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg    ggplot2

## Registered S3 method overwritten by 'genetics':
##   method      from
##   [.haplotype pegas
require(dplyr)

## Loading required package: dplyr
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##   filter, lag
## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
require(poppr)

## Loading required package: poppr
## This is poppr version 2.9.6. To get started, type package?poppr
## OMP parallel support: available
```

## 1. Overview

The data set we will use is ralu.loci

## 2. Import straight from the package after library is loaded

```
data(ralu.loci, package="LandGenCourse")
Frogs <- data.frame(FrogID = paste(substr(ralu.loci$Pop, 1, 3),
                                     row.names(ralu.loci), sep="."), ralu.loci)
Frogs.genind <- adegenet::df2genind(X=Frogs[,c(4:11)], sep=":", ncode=NULL,
                                   ind.names= Frogs$FrogID, loc.names=NULL,
                                   pop=Frogs$Pop, NA.char="NA", ploidy=2,
                                   type="codom", strata=NULL, hierarchy=NULL)

Frogs.genind

## /// GENIND OBJECT ///////////
##
## // 181 individuals; 8 loci; 39 alleles; size: 55.5 Kb
##
## // Basic content
##   @tab: 181 x 39 matrix of allele counts
##   @loc.n.all: number of alleles per locus (range: 3-9)
##   @loc.fac: locus factor for the 39 columns of @tab
##   @all.names: list of allele names for each locus
##   @ploidy: ploidy of each individual (range: 2-2)
##   @type: codom
##   @call: adegenet::df2genind(X = Frogs[, c(4:11)], sep = ":", ncode = NULL,
##     ind.names = Frogs$FrogID, loc.names = NULL, pop = Frogs$Pop,
##     NA.char = "NA", ploidy = 2, type = "codom", strata = NULL,
##     hierarchy = NULL)
##
## // Optional content
##   @pop: population of each individual (group size range: 7-23)

Get info on genind object, check that they are polymorphic
Frogs.genind

## /// GENIND OBJECT ///////////
##
## // 181 individuals; 8 loci; 39 alleles; size: 55.5 Kb
##
```

```
## // Basic content
##   @tab: 181 x 39 matrix of allele counts
##   @loc.n.all: number of alleles per locus (range: 3-9)
##   @loc.fac: locus factor for the 39 columns of @tab
##   @all.names: list of allele names for each locus
##   @ploidy: ploidy of each individual (range: 2-2)
##   @type: codon
##   @call: adegenet::df2genind(X = Frogs[, c(4:11)], sep = ":", ncode = NULL,
##     ind.names = Frogs$FrogID, loc.names = NULL, pop = Frogs$Pop,
##     NA.char = "NA", ploidy = 2, type = "codon", strata = NULL,
##     hierarchy = NULL)
##
## // Optional content
##   @pop: population of each individual (group size range: 7-23)
summary(Frogs.genind)
```

```
##
## // Number of individuals: 181
## // Group sizes: 21 8 14 13 7 17 9 20 19 13 17 23
## // Number of alleles per locus: 3 4 4 4 9 3 4 8
## // Number of alleles per group: 21 21 20 22 20 19 19 25 18 14 18 26
## // Percentage of missing data: 10.64 %
## // Observed heterozygosity: 0.1 0.4 0.09 0.36 0.68 0.02 0.38 0.68
## // Expected heterozygosity: 0.17 0.47 0.14 0.59 0.78 0.02 0.48 0.74
```

Test for HWE with pegas:

```
round(pegas::hw.test(Frogs.genind, B = 1000), digits = 3)
```

```
##      chi^2 df Pr(chi^2 >) Pr.exact
## A  40.462  3      0.000    0.000
## B  17.135  6      0.009    0.030
## C 136.522  6      0.000    0.000
## D  83.338  6      0.000    0.000
## E 226.803 36      0.000    0.000
## F   0.024  3      0.999    1.000
## G  12.349  6      0.055    0.005
## H  76.813 28      0.000    0.000
```

```
# Chi-squared test: p-value
HWE.test <- data.frame(sapply(seppop(Frogs.genind),
                             function(ls) pegas::hw.test(ls, B=0)[,3]))
HWE.test.chisq <- t(data.matrix(HWE.test))
{cat("Chi-squared test (p-values):", "\n")
round(HWE.test.chisq,3)}
```

## Chi-squared test (p-values):

```
##           A      B      C      D      E      F      G      H
## Airplane  0.092 0.359 1.000 0.427 0.680 1.000 0.178 0.051
## Bachelor  1.000 0.557 0.576 0.686 0.716 1.000 0.414 0.609
## BarkingFox 0.890 0.136 0.005 0.533 0.739 0.890 0.708 0.157
## Bob        0.764 0.864 0.362 0.764 0.033 1.000 0.860 0.287
## Cache      1.000 0.325 0.046 0.659 0.753 1.000 0.709 0.402
## Egg        1.000 0.812 1.000 1.000 0.156 1.000 0.477 0.470
## Frog       1.000 0.719 0.070 0.722 0.587 1.000 0.564 0.172
```

```
## GentianL    0.809 0.059 1.000 0.028 0.560 0.717 0.474 0.108
## ParagonL    1.000 0.054 0.885 0.709 0.868 1.000 0.291 0.000
## Pothole     1.000 1.000 1.000 0.488 0.248 1.000 0.296 0.850
## ShipIsland  0.807 0.497 1.000 0.521 0.006 1.000 0.498 0.403
## Skyhigh     0.915 0.493 0.063 0.001 0.155 1.000 0.126 0.078
```

```
# Monte Carlo: p-value
HWE.test <- data.frame(sapply(seppop(Frogs.genind),
                             function(ls) pegas::hw.test(ls, B=1000)[,4]))
HWE.test.MC <- t(data.matrix(HWE.test))
{cat("MC permutation test (p-values):", "\n")
round(HWE.test.MC,3)}
```

```
## MC permutation test (p-values):
```

```
##           A      B      C      D      E F      G      H
## Airplane   0.012 1.000 1.000 0.405 0.639 1 0.265 0.013
## Bachelor   1.000 0.460 1.000 1.000 0.872 1 0.475 0.600
## BarkingFox 1.000 0.210 0.062 1.000 0.720 1 1.000 0.174
## Bob        1.000 1.000 1.000 1.000 0.014 1 1.000 0.270
## Cache      1.000 0.411 0.140 1.000 1.000 1 1.000 0.616
## Egg        1.000 1.000 1.000 1.000 0.091 1 0.534 0.454
## Frog       1.000 1.000 0.062 1.000 0.411 1 1.000 0.188
## GentianL   1.000 0.059 1.000 0.068 0.693 1 0.655 0.128
## ParagonL   1.000 0.164 1.000 1.000 1.000 1 0.321 0.063
## Pothole    1.000 1.000 1.000 1.000 0.510 1 0.555 1.000
## ShipIsland 1.000 0.623 1.000 0.687 0.147 1 0.559 0.471
## Skyhigh    1.000 0.329 0.150 0.092 0.121 1 0.073 0.047
```

```
alpha=0.05 # /96
Prop.loci.out.of.HWE <- data.frame(Chisq=apply(HWE.test.chisq<alpha, 2, mean),
                                   MC=apply(HWE.test.MC<alpha, 2, mean))
Prop.loci.out.of.HWE           # Type this line again to see results table
```

```
##           Chisq      MC
## A 0.00000000 0.08333333
## B 0.00000000 0.00000000
## C 0.16666667 0.00000000
## D 0.16666667 0.00000000
## E 0.16666667 0.08333333
## F 0.00000000 0.00000000
## G 0.00000000 0.00000000
## H 0.08333333 0.16666667
```

```
Prop.pops.out.of.HWE <- data.frame(Chisq=apply(HWE.test.chisq<alpha, 1, mean),
                                   MC=apply(HWE.test.MC<alpha, 1, mean))
Prop.pops.out.of.HWE
```

```
##           Chisq      MC
## Airplane   0.000 0.250
## Bachelor   0.000 0.000
## BarkingFox 0.125 0.000
## Bob        0.125 0.125
## Cache      0.125 0.000
## Egg        0.000 0.000
## Frog       0.000 0.000
## GentianL   0.125 0.000
```

```
## ParagonL    0.125 0.000
## Pothole     0.000 0.000
## ShipIsland  0.125 0.000
## Skyhigh     0.125 0.125

Chisq.fdr <- matrix(p.adjust(HWE.test.chisq,method="fdr"),
                    nrow=nrow(HWE.test.chisq))
MC.fdr <- matrix(p.adjust(HWE.test.MC, method="fdr"),
                 nrow=nrow(HWE.test.MC))

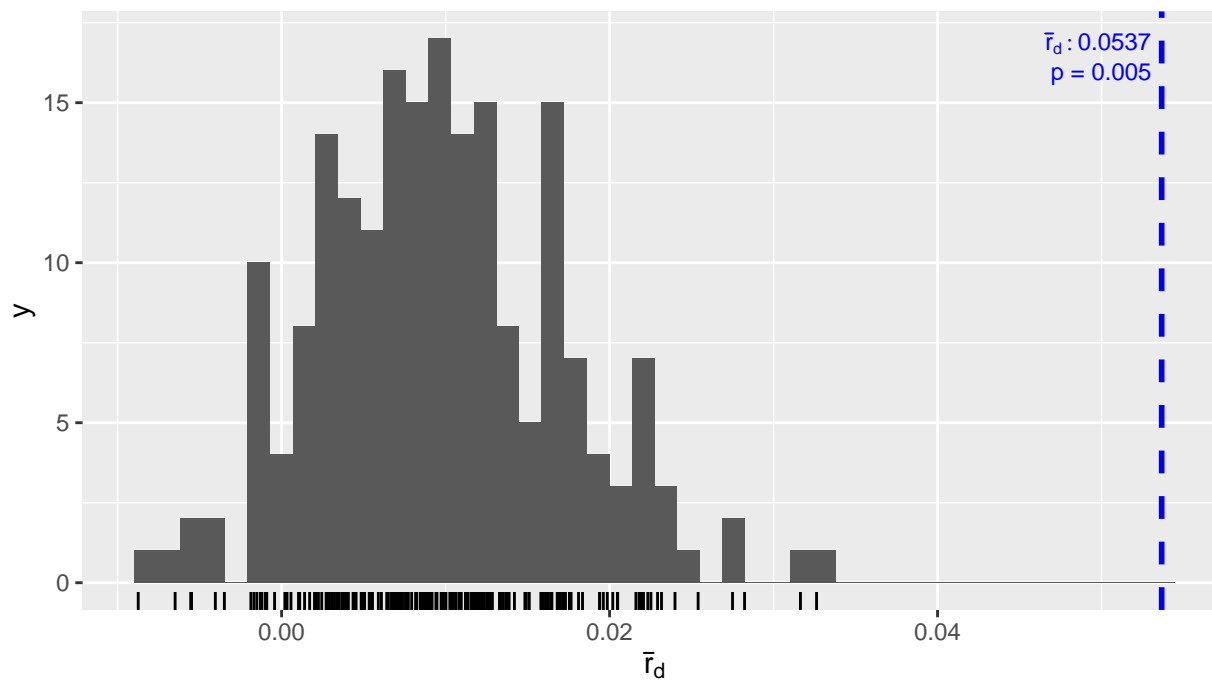
Prop.pops.out.of.HWE <- data.frame(Chisq=apply(HWE.test.chisq<alpha, 1, mean),
                                   MC=apply(HWE.test.MC<alpha, 1, mean),
                                   Chisq.fdr=apply(Chisq.fdr<alpha, 1, mean),
                                   MC.fdr=apply(MC.fdr<alpha, 1, mean))
Prop.pops.out.of.HWE
```

##		Chisq	MC	Chisq.fdr	MC.fdr
##	Airplane	0.000	0.250	0.000	0
##	Bachelor	0.000	0.000	0.000	0
##	BarkingFox	0.125	0.000	0.000	0
##	Bob	0.125	0.125	0.000	0
##	Cache	0.125	0.000	0.000	0
##	Egg	0.000	0.000	0.000	0
##	Frog	0.000	0.000	0.000	0
##	GentianL	0.125	0.000	0.000	0
##	ParagonL	0.125	0.000	0.125	0
##	Pothole	0.000	0.000	0.000	0
##	ShipIsland	0.125	0.000	0.000	0
##	Skyhigh	0.125	0.125	0.125	0

## Linkage Disequilibrium

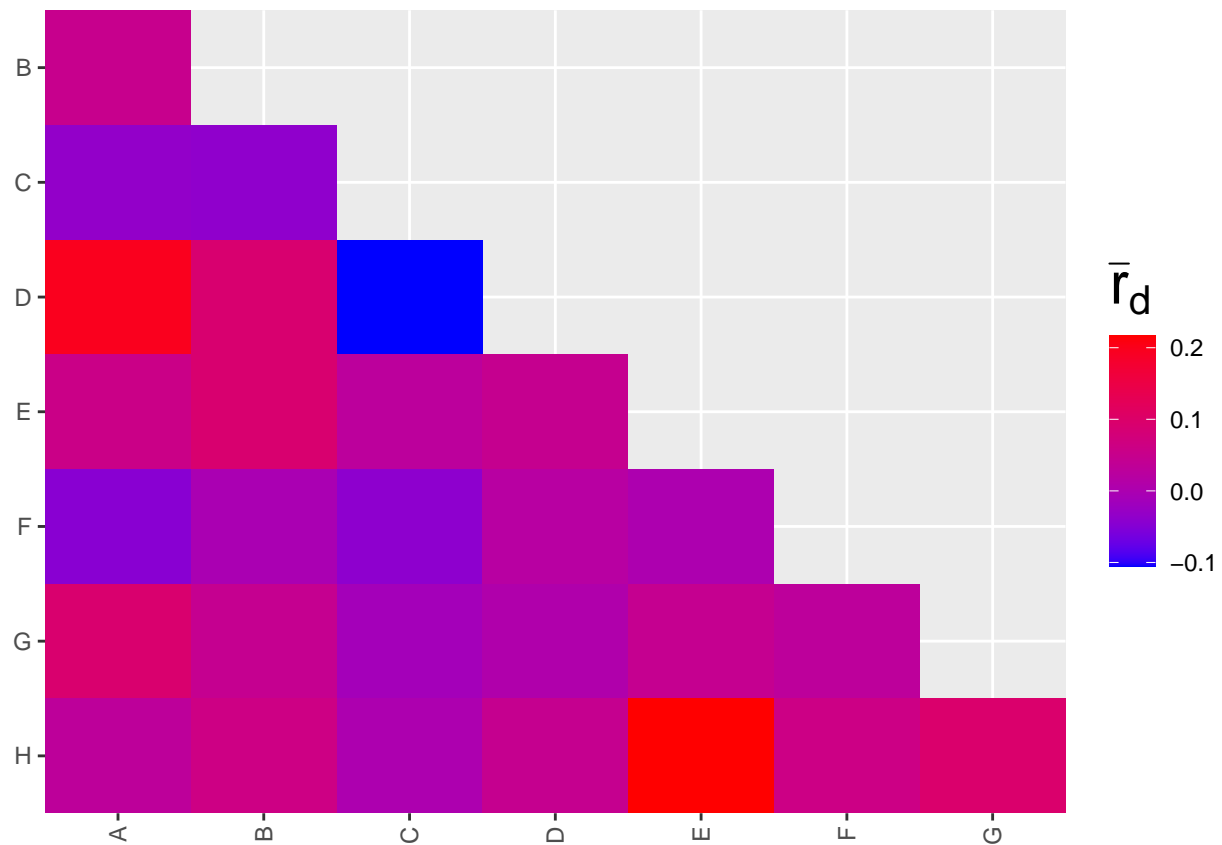
```
poppr::ia(Frogs.genind, sample=199)
```

Population: Total  
 N: 181  
 Data: Frogs.genind  
 Permutations: 199



```
##      Ia      p.Ia      rbarD      p.rD
## 0.33744318 0.00500000 0.05366542 0.00500000
```

```
LD.pair <- poppr::pair.ia(Frogs.genind)
```



LD.pair

```
##      Ia  rbarD
## A:B  0.0485  0.0492
## A:C -0.0314 -0.0335
## A:D  0.1886  0.1966
## A:E  0.0560  0.0569
## A:F -0.0272 -0.0452
## A:G  0.0931  0.0935
## A:H  0.0294  0.0304
## B:C -0.0329 -0.0375
## B:D  0.0903  0.0911
## B:E  0.0910  0.0910
## B:F -0.0013 -0.0025
## B:G  0.0451  0.0452
## B:H  0.0621  0.0623
## C:D -0.0859 -0.1049
## C:E  0.0247  0.0284
## C:F -0.0311 -0.0397
## C:G -0.0107 -0.0118
## C:H  0.0012  0.0015
## D:E  0.0455  0.0458
## D:F  0.0094  0.0199
## D:G  0.0069  0.0070
## D:H  0.0461  0.0462
## E:F  0.0013  0.0025
## E:G  0.0453  0.0454
```

```
## E:H  0.2153  0.2159
## F:G  0.0167  0.0299
## F:H  0.0296  0.0606
## G:H  0.0942  0.0953
```

## Null Alleles

```
# Null alleles: depends on method! See help file.
Null.alleles <- PopGenReport::null.all(Frogs.genind)
```

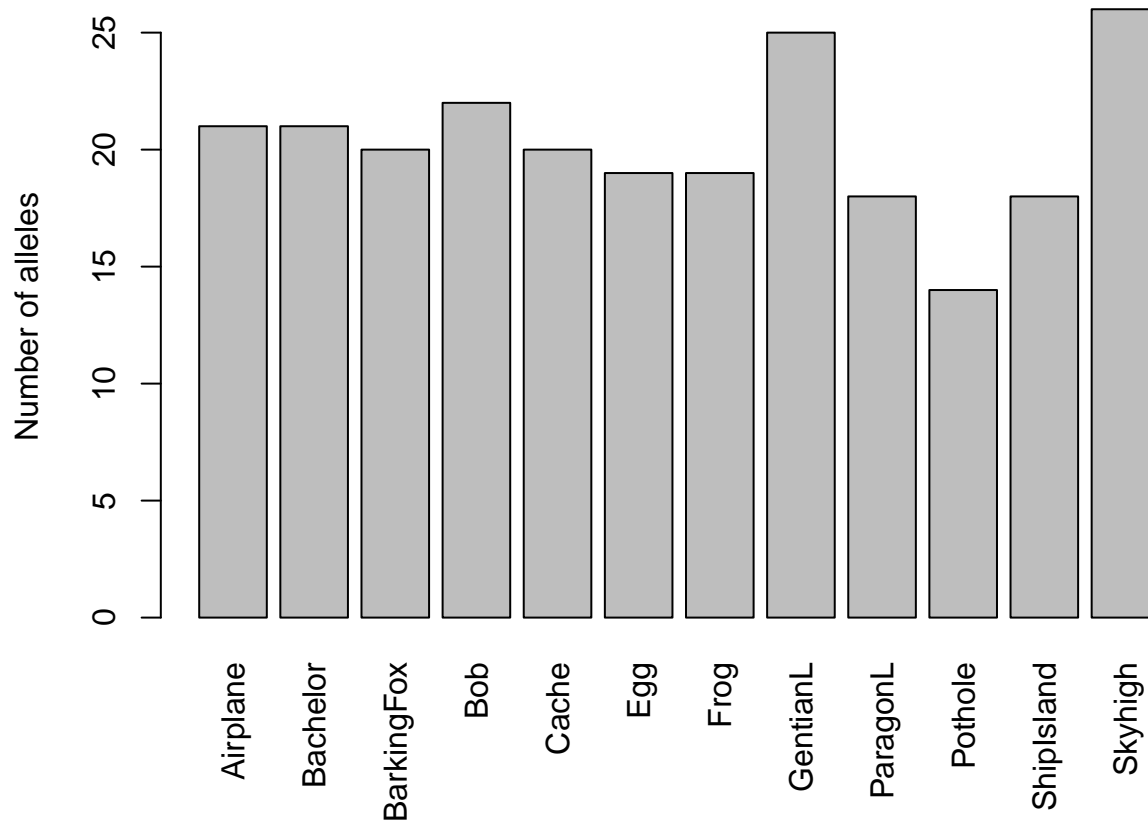
## Genetic Diversity

### allelic richness

```
Sum <- adegenet::summary(Frogs.genind)
names(Sum)

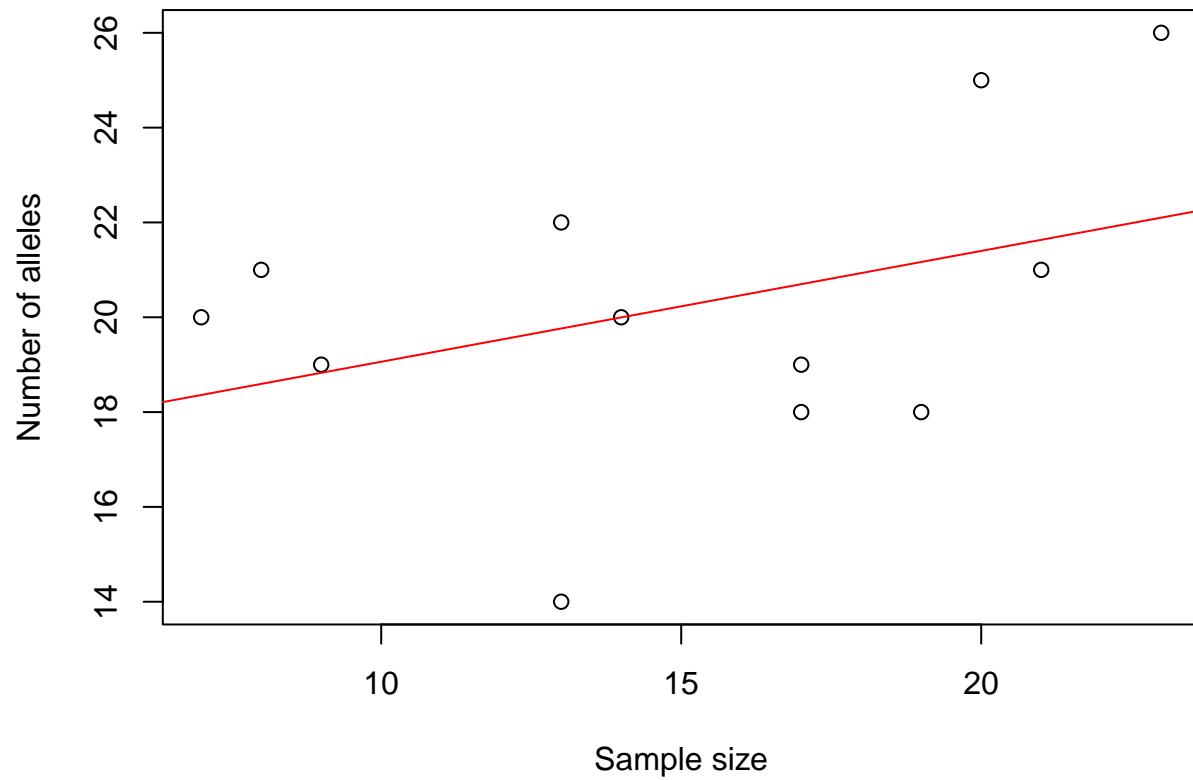
## [1] "n"          "n.by.pop"  "loc.n.all" "pop.n.all" "NA.perc"   "Hobs"
## [7] "Hexp"

par(mar=c(5.5, 4.5,1,1))
barplot(Sum$pop.n.all, las=3,
        xlab = "", ylab = "Number of alleles")
```



```
plot(Sum$n.by.pop, Sum$pop.n.all,
     xlab = "Sample size", ylab = "Number of alleles")
abline(lm(Sum$pop.n.all ~ Sum$n.by.pop), col = "red")
```

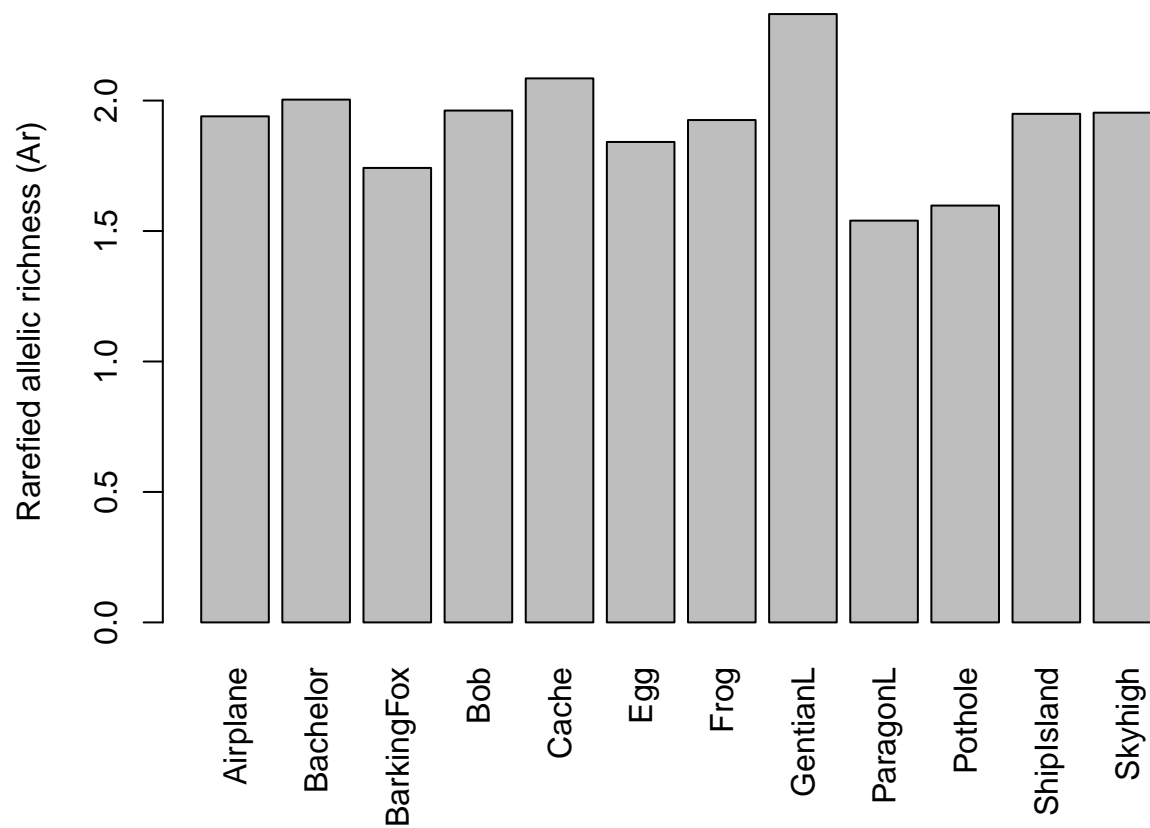




```
Richness <- PopGenReport::allel.rich(Frogs.genind, min.alleles = NULL)
Richness$alleles.sampled
```

```
## [1] 6
```

```
par(mar=c(5.5, 4.5,1,1))
barplot(Richness$mean.richness, las=3, ylab="Rarefied allelic richness (Ar)")
```



```
plot(colMeans(Richness$pop.sizes), Richness$mean.richness,
     xlab="Valid sample size",
     ylab="Rarefied allelic richness (Ar)")
abline(lm(Richness$mean.richness ~ colMeans(Richness$pop.sizes)), col="red")
```

