

Lab 2

Norah Saarman

2024-09-12

Section 4.3 Worked Example

https://bookdown.org/hhwagner1/LandGenCourse_book/WE_1.html

Make sure we have the packages:

```
if(!require("adegenet")) install.packages("adegenet") if(!requireNamespace("popgraph", quietly = TRUE))
{ install.packages(c("RgoogleMaps", "geosphere", "proto", "sampling", "seqinr", "spacetime", "spdep"), de-
dependencies=TRUE) remotes::install_github("dyerlab/popgraph") } if(!requireNamespace("gstudio", quietly
= TRUE)) remotes::install_github("dyerlab/gstudio")
```

```
install.packages("here")
```

Load libraries

```
library(adegenet)
```

```
## Loading required package: ade4
##
##    /// adegenet 2.1.10 is loaded //////////////////////////////////
##
##    > overview: '?adegenet'
##    > tutorials/doc/questions: 'adegenetWeb()'
##    > bug reports/feature requests: adegenetIssues()
```

```
library(gstudio)
```

```
## Warning: replacing previous import 'dplyr::union' by 'raster::union' when
## loading 'gstudio'
##
## Warning: replacing previous import 'dplyr::intersect' by 'raster::intersect'
## when loading 'gstudio'
##
## Warning: replacing previous import 'dplyr::select' by 'raster::select' when
## loading 'gstudio'
##
## Attaching package: 'gstudio'
##
## The following objects are masked from 'package:adegenet':
##
##    alleles, ploidy
```

```
library(LandGenCourse)
library(tibble)
library(here)
```

```
## here() starts at /uufs/chpc.utah.edu/common/home/u6036559/git/usu-biol4750
```

```
library(vcfR)

##
##      *****      ***   vcfR   ***      *****
##      This is vcfR 1.15.0
##      browseVignettes('vcfR') # Documentation
##      citation('vcfR') # Citation
##      *****      *****      *****      *****

library(pinfsc50)
library(utils)
```

1. Overview

The data set we will use is ralu.loci

Import straight from the package after library is loaded

```
data("ralu.loci")
```

As a .csv file:

To download the csv file...

```
if(!dir.exists(paste0(here(), "/downloads"))) dir.create(paste0(here(), "/downloads"))
file.copy(system.file("extdata", "ralu.loci.csv",
                      package = "LandGenCourse"),
          paste0(here(), "/downloads/ralu.loci.csv"),
          overwrite=FALSE)
```

```
## [1] FALSE
```

2. Import from csv file:

```
Frogs <- read.csv(file="./downloads/ralu.loci.csv", header=TRUE)
as_tibble(Frogs)
```

```
## # A tibble: 181 x 10
##   SiteName      Pop      A      B      C      D      E      F      G      H
##   <chr>         <chr>   <chr> <chr> <chr> <chr> <chr> <chr> <chr> <chr>
## 1 AirplaneLake Airplane 1:1   1:1   1:1   1:1   1:2   1:1   1:1   4:5
## 2 AirplaneLake Airplane 2:2   1:1   NA:NA 1:1   1:1   NA:NA 2:2   NA:NA
## 3 AirplaneLake Airplane 1:1   1:1   1:1   1:1   3:3   1:1   1:1   3:3
## 4 AirplaneLake Airplane 1:1   1:1   NA:NA 2:2   1:2   NA:NA NA:NA NA:NA
## 5 AirplaneLake Airplane 1:2   1:3   1:1   1:1   1:2   1:1   1:2   4:5
## 6 AirplaneLake Airplane 1:2   1:1   1:1   3:1   1:1   1:1   1:2   4:5
## 7 AirplaneLake Airplane 2:2   1:3   1:1   1:1   3:3   1:1   1:1   2:3
## 8 AirplaneLake Airplane 2:2   1:3   1:1   1:1   3:3   1:1   1:1   2:3
## 9 AirplaneLake Airplane 3:1   1:1   1:1   1:1   1:7   1:1   1:1   3:5
## 10 AirplaneLake Airplane 2:2   1:3   1:1   1:1   3:7   1:1   1:1   3:3
## # i 171 more rows
```

Adding a column that gives us Frogs\$FrogID

```
Frogs <- data.frame(FrogID = paste(substr(Frogs$Pop, 1, 3), row.names(Frogs), sep="."), Frogs)
as_tibble(Frogs)
```

```
## # A tibble: 181 x 11
```

```
##      FrogID SiteName      Pop      A      B      C      D      E      F      G      H
##      <chr>  <chr>      <chr>  <chr> <chr> <chr> <chr> <chr> <chr> <chr> <chr>
##  1 Air.1   AirplaneLake Airplane 1:1   1:1   1:1   1:1   1:2   1:1   1:1   4:5
##  2 Air.2   AirplaneLake Airplane 2:2   1:1   NA:NA 1:1   1:1   NA:NA 2:2   NA:NA
##  3 Air.3   AirplaneLake Airplane 1:1   1:1   1:1   1:1   3:3   1:1   1:1   3:3
##  4 Air.4   AirplaneLake Airplane 1:1   1:1   NA:NA 2:2   1:2   NA:NA NA:NA NA:NA
##  5 Air.5   AirplaneLake Airplane 1:2   1:3   1:1   1:1   1:2   1:1   1:2   4:5
##  6 Air.6   AirplaneLake Airplane 1:2   1:1   1:1   3:1   1:1   1:1   1:2   4:5
##  7 Air.7   AirplaneLake Airplane 2:2   1:3   1:1   1:1   3:3   1:1   1:1   2:3
##  8 Air.8   AirplaneLake Airplane 2:2   1:3   1:1   1:1   3:3   1:1   1:1   2:3
##  9 Air.9   AirplaneLake Airplane 3:1   1:1   1:1   1:1   1:7   1:1   1:1   3:5
## 10 Air.10 AirplaneLake Airplane 2:2   1:3   1:1   1:1   3:7   1:1   1:1   3:3
## # i 171 more rows
```

```
dim(Frogs)
```

```
## [1] 181  11
```

Some useful file directory functions

```
here()
```

```
## [1] "/uufs/chpc.utah.edu/common/home/u6036559/git/usu-biol4750"
```

```
#file.choose() but don't leave within code chunk without commenting it out
```

```
paste0(here(), "/output", "/gobblygook")
```

```
## [1] "/uufs/chpc.utah.edu/common/home/u6036559/git/usu-biol4750/output/gobblygook"
```

Save output file

```
if(!dir.exists(paste0(here(), "/output"))) dir.create(paste0(here(), "/output"))
```

```
write.csv(ralu.loci, paste0(here(), "/output/ralu.loci.csv"),
          quote=FALSE, row.names=FALSE)
```

3. Create a 'genind' object using adegenet

```
Frogs[,c(4:11)]
```

```
##      A      B      C      D      E      F      G      H
##  1  1:1  1:1  1:1  1:1  1:2  1:1  1:1  4:5
##  2  2:2  1:1 NA:NA 1:1  1:1 NA:NA 2:2 NA:NA
##  3  1:1  1:1  1:1  1:1  3:3  1:1  1:1  3:3
##  4  1:1  1:1 NA:NA 2:2  1:2 NA:NA NA:NA NA:NA
##  5  1:2  1:3  1:1  1:1  1:2  1:1  1:2  4:5
##  6  1:2  1:1  1:1  3:1  1:1  1:1  1:2  4:5
##  7  2:2  1:3  1:1  1:1  3:3  1:1  1:1  2:3
##  8  2:2  1:3  1:1  1:1  3:3  1:1  1:1  2:3
##  9  3:1  1:1  1:1  1:1  1:7  1:1  1:1  3:5
## 10  2:2  1:3  1:1  1:1  3:7  1:1  1:1  3:3
## 11  1:1  1:1 NA:NA 1:1  1:3  1:1  1:1  4:5
## 12  1:1  1:1  1:1  1:2  1:1  1:1  1:1  3:3
## 13  1:1  1:1  1:1  1:1  1:3  1:1  1:1  4:5
```

## 14	1:1	1:1	NA:NA	1:1	2:7	1:1	1:1	3:5
## 15	1:1	1:1	1:1	1:2	1:3	1:1	1:2	4:5
## 16	1:1	1:3	1:1	3:1	1:1	1:1	NA:NA	4:4
## 17	1:2	1:3	NA:NA	3:1	1:3	1:1	1:1	4:5
## 18	1:1	1:1	NA:NA	1:1	1:2	1:1	1:1	3:5
## 19	1:2	1:1	NA:NA	1:1	2:3	1:1	1:1	5:5
## 20	NA:NA	1:3	NA:NA	3:1	1:1	1:1	1:1	1:4
## 21	2:2	1:1	1:1	1:2	3:7	1:1	1:1	2:3
## 22	1:1	1:1	1:2	2:2	6:8	1:1	2:2	6:4
## 23	1:1	2:2	1:1	2:2	2:2	1:1	1:2	6:4
## 24	1:1	1:2	1:1	3:2	1:2	1:1	NA:NA	1:4
## 25	NA:NA	1:3	NA:NA	2:2	2:5	1:1	NA:NA	4:5
## 26	1:1	1:2	1:1	3:2	2:5	1:1	1:2	4:5
## 27	1:1	2:2	1:2	2:2	2:3	1:1	1:1	6:4
## 28	1:1	1:1	NA:NA	2:2	2:3	1:1	1:1	4:4
## 29	1:1	1:1	NA:NA	2:2	2:8	1:1	2:2	5:5
## 30	1:1	1:1	NA:NA	4:4	1:3	1:1	1:1	5:5
## 31	1:1	1:1	1:1	4:2	1:3	1:1	NA:NA	5:5
## 32	1:1	1:1	2:2	2:2	1:2	1:1	NA:NA	4:4
## 33	1:1	1:1	1:1	2:2	3:3	1:1	1:1	NA:NA
## 34	3:1	1:1	1:1	4:4	1:1	1:1	NA:NA	5:5
## 35	1:1	1:1	1:1	4:2	1:3	1:1	NA:NA	5:5
## 36	1:1	1:3	NA:NA	4:2	1:5	1:1	1:2	4:5
## 37	1:1	1:1	NA:NA	4:2	1:3	1:1	1:2	5:5
## 38	1:1	1:1	NA:NA	2:2	1:1	1:1	1:1	5:5
## 39	1:1	2:2	1:1	4:2	3:5	1:1	1:1	5:5
## 40	1:1	1:1	NA:NA	4:2	1:2	1:1	NA:NA	3:5
## 41	1:1	1:1	1:1	4:2	1:3	1:1	1:1	3:5
## 42	1:1	1:2	1:1	2:2	1:5	2:1	1:1	5:5
## 43	1:1	1:1	NA:NA	4:2	1:3	1:1	1:1	5:5
## 44	1:1	1:2	1:2	2:2	3:5	1:1	1:1	3:5
## 45	1:1	1:1	1:2	1:2	1:3	NA:NA	NA:NA	NA:NA
## 46	1:1	1:1	1:1	2:2	1:4	1:1	1:1	6:4
## 47	1:1	4:1	1:1	2:2	NA:NA	1:1	1:1	6:5
## 48	1:1	1:1	1:1	2:2	3:5	1:1	1:1	6:4
## 49	1:1	1:1	1:2	2:2	1:2	1:1	1:1	3:4
## 50	1:1	1:1	NA:NA	2:2	3:5	1:1	1:1	6:4
## 51	1:1	1:2	1:1	2:2	1:8	1:1	NA:NA	5:5
## 52	1:1	1:2	1:1	2:2	1:8	1:1	NA:NA	5:5
## 53	3:1	1:1	1:2	2:2	1:3	1:1	1:1	4:5
## 54	1:1	1:1	1:1	2:2	3:5	1:1	1:1	6:4
## 55	1:1	4:1	1:2	1:2	2:2	1:1	NA:NA	4:4
## 56	3:1	1:1	1:1	2:2	1:8	1:1	1:2	5:5
## 57	1:1	NA:NA	NA:NA	1:2	1:2	1:1	1:1	6:4
## 58	NA:NA	1:2	1:1	2:2	2:5	1:1	NA:NA	6:4
## 59	1:1	4:1	NA:NA	2:2	2:8	1:1	NA:NA	6:4
## 60	1:1	1:2	1:1	2:2	1:4	1:1	1:2	4:5
## 61	1:1	2:2	NA:NA	1:2	1:4	1:1	1:1	6:5
## 62	1:1	1:2	2:2	2:2	2:4	1:1	2:2	5:5
## 63	1:1	1:2	1:1	2:2	4:5	1:1	1:2	1:5
## 64	1:1	1:2	1:1	2:2	NA:NA	1:1	2:2	3:4
## 65	1:1	1:2	1:1	2:2	4:5	1:1	1:2	6:5
## 66	1:1	1:1	1:1	2:2	2:8	1:1	1:2	6:4
## 67	1:1	1:1	1:1	2:2	2:5	1:1	NA:NA	6:4

## 68	1:1	1:1	1:1	2:2	4:5	1:1	1:2	4:5
## 69	1:1	1:1	NA:NA	2:2	1:6	1:1	2:2	5:5
## 70	1:1	1:2	1:1	2:2	4:5	1:1	1:2	6:5
## 71	1:1	1:2	1:1	2:2	4:5	NA:NA	NA:NA	NA:NA
## 72	1:1	1:1	1:1	2:2	1:1	1:1	NA:NA	NA:NA
## 73	1:1	2:2	1:1	2:2	2:5	1:1	1:1	4:4
## 74	1:1	1:2	NA:NA	2:2	1:4	1:1	NA:NA	1:5
## 75	1:1	1:1	1:1	2:2	2:5	1:1	NA:NA	6:4
## 76	1:1	1:1	1:1	2:2	2:8	1:1	NA:NA	6:4
## 77	1:1	1:1	1:1	2:2	2:4	1:1	1:1	6:5
## 78	1:1	1:2	NA:NA	2:2	2:2	1:1	1:1	4:5
## 79	1:1	1:1	1:1	2:2	NA:NA	1:1	1:1	6:5
## 80	1:1	1:2	1:1	2:2	1:4	1:1	1:1	1:5
## 81	1:1	1:2	1:1	3:2	2:4	1:1	NA:NA	6:4
## 82	1:1	1:1	NA:NA	1:2	NA:NA	NA:NA	NA:NA	NA:NA
## 83	1:1	2:2	1:1	3:2	2:4	1:1	1:2	4:5
## 84	1:1	1:1	1:1	1:2	2:2	NA:NA	NA:NA	NA:NA
## 85	1:1	1:1	4:4	2:2	3:4	1:1	1:1	4:5
## 86	1:1	1:1	1:2	2:2	1:4	NA:NA	NA:NA	NA:NA
## 87	1:1	1:2	NA:NA	2:2	NA:NA	NA:NA	NA:NA	NA:NA
## 88	1:1	1:2	1:1	3:2	2:4	NA:NA	NA:NA	4:5
## 89	1:1	NA:NA	1:1	2:2	2:4	1:1	2:2	4:5
## 90	1:1	1:3	1:1	3:3	1:1	1:1	2:2	5:5
## 91	1:1	1:2	1:1	3:2	3:5	1:1	1:2	3:5
## 92	1:1	1:3	NA:NA	3:1	2:6	1:3	1:2	3:4
## 93	1:1	1:3	NA:NA	3:1	1:7	1:1	1:2	3:5
## 94	NA:NA	1:3	1:1	1:1	2:5	1:1	1:1	4:5
## 95	1:2	1:2	1:1	3:1	1:2	1:1	NA:NA	4:5
## 96	1:1	NA:NA	1:1	4:2	2:2	1:3	1:2	4:4
## 97	1:1	2:3	NA:NA	3:1	1:5	1:1	2:2	5:5
## 98	1:1	2:3	1:1	4:2	1:2	1:1	1:2	4:5
## 99	1:1	1:3	NA:NA	1:2	5:7	1:1	1:2	3:5
## 100	1:1	2:3	1:1	3:2	1:6	1:1	2:2	3:5
## 101	1:2	2:3	NA:NA	3:4	1:6	1:1	2:2	3:5
## 102	1:1	2:3	NA:NA	4:2	1:2	1:3	1:2	1:4
## 103	1:1	2:3	NA:NA	3:2	5:6	1:1	1:2	1:8
## 104	1:1	1:2	NA:NA	1:1	5:7	1:1	1:2	NA:NA
## 105	1:1	2:3	NA:NA	4:2	1:7	1:1	1:2	1:4
## 106	1:1	2:2	1:1	1:1	5:7	1:1	1:1	1:3
## 107	1:1	1:2	NA:NA	1:1	5:6	1:1	NA:NA	3:5
## 108	1:1	1:2	1:1	NA:NA	1:2	1:1	2:2	4:5
## 109	1:1	2:3	1:1	3:1	2:3	1:1	2:2	1:4
## 110	1:1	1:1	1:1	2:2	1:1	1:1	1:2	3:5
## 111	1:1	1:1	1:1	2:2	1:4	1:1	1:2	3:5
## 112	1:1	4:1	1:1	2:2	1:1	1:1	1:1	5:5
## 113	1:1	1:1	NA:NA	2:2	1:2	1:1	NA:NA	2:5
## 114	1:1	4:4	1:1	2:2	1:4	1:1	1:1	5:5
## 115	1:1	1:1	NA:NA	2:2	1:4	1:1	2:2	3:5
## 116	1:1	1:1	1:5	2:2	1:1	1:1	2:2	3:3
## 117	1:1	1:1	NA:NA	2:2	1:1	1:1	2:2	3:5
## 118	1:1	1:1	1:1	2:2	1:1	1:1	1:2	5:5
## 119	1:1	1:1	1:1	2:2	1:1	1:1	1:2	3:5
## 120	NA:NA	1:1	NA:NA	2:2	1:2	1:1	NA:NA	4:8
## 121	1:1	1:1	1:1	2:2	1:1	1:1	1:1	5:5

## 122	1:1	1:1	1:1	2:2	1:1	1:1	NA:NA	3:3
## 123	1:1	1:1	1:1	4:2	1:1	1:1	NA:NA	5:5
## 124	1:1	1:1	NA:NA	4:2	1:4	1:1	1:2	3:5
## 125	1:1	1:1	1:1	4:2	1:1	1:1	1:1	3:3
## 126	1:1	4:1	1:1	2:2	1:1	1:1	2:2	5:5
## 127	1:1	1:1	NA:NA	2:2	1:1	1:1	1:1	3:3
## 128	1:1	1:1	1:1	2:2	NA:NA	1:1	NA:NA	3:5
## 129	1:1	1:1	1:1	3:1	1:1	NA:NA	NA:NA	NA:NA
## 130	1:1	1:1	1:1	1:1	1:7	1:1	NA:NA	3:5
## 131	1:1	1:1	NA:NA	NA:NA	1:1	1:1	3:2	1:5
## 132	1:1	1:1	1:1	1:1	1:1	1:1	NA:NA	5:5
## 133	1:1	1:1	1:1	1:1	1:7	1:1	3:2	3:3
## 134	1:1	1:1	NA:NA	1:1	NA:NA	1:1	1:1	NA:NA
## 135	1:1	1:1	1:1	3:1	1:7	1:1	NA:NA	3:5
## 136	1:1	1:1	1:1	1:1	1:1	1:1	2:2	5:5
## 137	1:1	1:1	1:1	1:1	1:1	1:1	3:2	1:3
## 138	1:1	1:1	1:1	1:1	1:7	1:1	2:2	1:3
## 139	1:1	1:1	NA:NA	1:1	1:7	1:1	1:2	3:5
## 140	1:1	1:1	NA:NA	3:1	1:1	1:1	NA:NA	1:5
## 141	1:1	1:1	NA:NA	3:1	1:7	1:1	NA:NA	3:5
## 142	1:2	3:3	NA:NA	3:3	2:3	1:1	NA:NA	5:5
## 143	1:1	1:1	NA:NA	1:1	10:10	NA:NA	NA:NA	3:3
## 144	2:2	1:1	1:1	1:1	2:2	1:1	1:1	4:4
## 145	1:1	1:1	NA:NA	3:1	1:3	1:1	2:2	4:5
## 146	1:2	3:3	1:1	1:1	2:2	1:1	1:1	4:5
## 147	1:1	3:3	1:1	1:1	2:3	1:1	1:1	3:5
## 148	1:1	1:3	1:1	1:1	2:2	1:1	1:1	3:3
## 149	1:2	1:1	1:1	3:1	1:2	1:1	1:2	3:4
## 150	1:2	1:1	1:1	1:2	2:2	1:1	1:1	3:4
## 151	1:1	1:3	1:1	1:2	1:2	1:1	1:2	3:4
## 152	1:2	3:3	1:1	3:1	1:2	1:1	2:2	4:4
## 153	1:1	1:3	1:1	1:2	2:2	1:1	1:1	4:5
## 154	2:2	1:3	1:1	3:1	2:2	1:1	1:1	4:5
## 155	1:1	1:1	NA:NA	1:1	1:1	1:1	1:2	4:5
## 156	1:2	1:3	NA:NA	3:1	1:2	1:1	1:2	4:5
## 157	1:2	1:3	NA:NA	1:2	1:2	1:1	1:2	4:5
## 158	1:1	1:3	1:1	3:1	1:3	1:1	NA:NA	3:5
## 159	1:1	1:1	NA:NA	2:2	2:2	1:1	1:2	6:4
## 160	1:1	1:1	1:1	2:2	2:2	1:1	1:2	6:4
## 161	1:1	1:2	1:1	3:2	2:8	1:1	1:1	6:4
## 162	1:1	2:2	1:1	2:2	2:3	1:1	1:2	3:4
## 163	1:1	1:2	1:1	2:2	1:5	1:1	NA:NA	7:6
## 164	1:1	1:2	1:1	1:2	2:2	1:1	1:1	6:4
## 165	3:1	1:1	1:1	2:2	2:2	1:1	2:5	6:6
## 166	1:1	1:2	1:1	1:2	2:2	1:1	NA:NA	6:4
## 167	1:1	1:2	1:2	1:2	8:8	1:1	NA:NA	6:4
## 168	1:1	1:1	1:2	2:2	1:8	1:1	1:2	4:4
## 169	1:1	1:2	1:1	1:2	2:5	1:1	NA:NA	NA:NA
## 170	1:1	1:1	NA:NA	4:4	1:4	NA:NA	NA:NA	5:5
## 171	1:1	1:2	NA:NA	2:2	2:8	1:1	1:2	6:6
## 172	1:1	1:2	1:1	2:2	2:2	1:1	1:2	4:4
## 173	1:1	2:2	NA:NA	2:2	2:5	1:1	NA:NA	4:5
## 174	1:1	1:1	1:1	2:2	2:5	1:1	NA:NA	4:4
## 175	1:1	1:1	1:1	2:2	2:8	1:1	NA:NA	6:4

```
## 176 1:1 1:2 1:1 2:2 2:2 1:1 1:2 6:4
## 177 1:1 1:2 1:1 1:2 3:8 1:1 1:1 6:4
## 178 1:1 1:2 NA:NA 3:2 2:2 1:1 NA:NA 5:5
## 179 1:1 1:2 1:1 2:2 2:2 1:1 1:2 6:4
## 180 1:1 4:2 1:1 2:2 3:3 1:1 1:2 4:4
## 181 1:1 1:1 2:2 2:2 1:2 1:1 NA:NA 4:4
```

```
Frogs.genind <- 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)
```

Check the genind object

```
Frogs.genind
```

```
## /// GENIND OBJECT ///////////
##
## // 181 individuals; 8 loci; 39 alleles; size: 55.2 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: 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)
```

```
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
```

4. View info stored in ‘genind’ object

Pull out subsets of info:

```
as_tibble(Frogs.genind@tab)
```

```
## # A tibble: 181 x 39
##   A.1 A.2 A.3 B.1 B.3 B.2 B.4 C.1 C.2 C.4 C.5 D.1 D.2
##   <int> <int> <int> <int> <int> <int> <int> <int> <int> <int> <int> <int> <int>
```

```
## 1 2 0 0 2 0 0 0 2 0 0 0 2 0
## 2 0 2 0 2 0 0 0 NA NA NA NA 2 0
## 3 2 0 0 2 0 0 0 2 0 0 0 2 0
## 4 2 0 0 2 0 0 0 NA NA NA NA 0 2
## 5 1 1 0 1 1 0 0 2 0 0 0 2 0
## 6 1 1 0 2 0 0 0 2 0 0 0 1 0
## 7 0 2 0 1 1 0 0 2 0 0 0 2 0
## 8 0 2 0 1 1 0 0 2 0 0 0 2 0
## 9 1 0 1 2 0 0 0 2 0 0 0 2 0
## 10 0 2 0 1 1 0 0 2 0 0 0 2 0
## # i 171 more rows
## # i 26 more variables: D.3 <int>, D.4 <int>, E.1 <int>, E.2 <int>, E.3 <int>,
## # E.7 <int>, E.6 <int>, E.8 <int>, E.5 <int>, E.4 <int>, E.10 <int>,
## # F.1 <int>, F.2 <int>, F.3 <int>, G.1 <int>, G.2 <int>, G.3 <int>,
## # G.5 <int>, H.4 <int>, H.5 <int>, H.3 <int>, H.2 <int>, H.1 <int>,
## # H.6 <int>, H.8 <int>, H.7 <int>
```

5. From here on, uses gstudio, and is optional for now.

Part 2: Section 4.4 R exercise Week 1

Do this section on your own!

Do this section on your own!

Download file:

```
file.copy(system.file("extdata", "pulsatilla_genotypes.csv",
  package = "LandGenCourse"),
  paste0(here(), "/downloads/pulsatilla_genotypes.csv"),
  overwrite=FALSE)
```

```
## [1] FALSE
```

Import from csv as data frame:

```
Frogs <- read.csv(paste0(here(), "/downloads/pulsatilla_genotypes.csv"),
  header=TRUE)
as_tibble(Frogs)
```

```
## # A tibble: 536 x 19
##   ID OffID Population      X      Y loc1_a loc1_b loc2_a loc2_b loc3_a
##   <int> <int> <chr>      <dbl> <dbl> <int> <int> <int> <int> <int>
## 1 62 0 A21 4426941. 5427173. 340 340 422 422 413
## 2 64 0 A21 4426933. 5427178. 334 334 424 424 417
## 3 65 0 A21 4426936. 5427173. 338 340 417 422 417
## 4 66 0 A21 4426937. 5427174. 340 344 422 422 411
## 5 68 0 A21 4426934. 5427171. 336 342 417 422 423
## 6 69 0 A21 4426933. 5427166. 336 346 422 422 417
## 7 75 0 A21 4426925. 5427175. 340 340 422 422 415
## 8 76 0 A21 4426925. 5427173. 338 340 417 422 413
## 9 77 0 A21 4426922. 5427174. 344 352 422 422 415
## 10 78 0 A21 4426922. 5427174. 342 352 417 424 425
## # i 526 more rows
## # i 9 more variables: loc3_b <int>, loc4_a <int>, loc4_b <int>, loc5_a <int>,
## # loc5_b <int>, loc6_a <int>, loc6_b <int>, loc7_a <int>, loc7_b <int>
```


Create 'genind' object:

```
Frogs.genind <- df2genind(X=Frogs[,c(6:19)], sep="\t",
                        ncode=NULL, ind.names= Frogs$ID,
                        loc.names=NULL, pop=Frogs$Population,
                        NA.char="NA", ploidy=2, type="codom",
                        strata=NULL, hierarchy=NULL)
```

```
## Warning in df2genind(X = Frogs[, c(6:19)], sep = "\t", ncode = NULL, ind.names
## = Frogs$ID, : duplicate labels detected for some individuals; using generic
## labels
```

Get info on genind object:

```
Frogs.genind
```

```
## /// GENIND OBJECT ///////////
##
## // 536 individuals; 14 loci; 184 alleles; size: 459.3 Kb
##
## // Basic content
##   @tab: 536 x 184 matrix of allele counts
##   @loc.n.all: number of alleles per locus (range: 5-24)
##   @loc.fac: locus factor for the 184 columns of @tab
##   @all.names: list of allele names for each locus
##   @ploidy: ploidy of each individual (range: 2-2)
##   @type: codom
##   @call: df2genind(X = Frogs[, c(6:19)], sep = "\t", ncode = NULL, ind.names = Frogs$ID,
##     loc.names = NULL, pop = Frogs$Population, NA.char = "NA",
##     ploidy = 2, type = "codom", strata = NULL, hierarchy = NULL)
##
## // Optional content
##   @pop: population of each individual (group size range: 55-128)
```

Here, there is a problem because the genind object has 14 loci, but really this should be only 7 loci.

Going back to the Frogs dataframe and combining loci before creating the genind object

```
as_tibble(Frogs)
```

```
## # A tibble: 536 x 19
##       ID OffID Population      X      Y loc1_a loc1_b loc2_a loc2_b loc3_a
##   <int> <int> <chr>      <dbl>    <dbl> <int> <int> <int> <int> <int>
## 1    62     0 A21      4426941. 5427173.   340   340   422   422   413
## 2    64     0 A21      4426933. 5427178.   334   334   424   424   417
## 3    65     0 A21      4426936. 5427173.   338   340   417   422   417
## 4    66     0 A21      4426937. 5427174.   340   344   422   422   411
## 5    68     0 A21      4426934. 5427171.   336   342   417   422   423
## 6    69     0 A21      4426933. 5427166.   336   346   422   422   417
## 7    75     0 A21      4426925. 5427175.   340   340   422   422   415
## 8    76     0 A21      4426925. 5427173.   338   340   417   422   413
## 9    77     0 A21      4426922. 5427174.   344   352   422   422   415
## 10   78     0 A21      4426922. 5427174.   342   352   417   424   425
## # i 526 more rows
## # i 9 more variables: loc3_b <int>, loc4_a <int>, loc4_b <int>, loc5_a <int>,
## #   loc5_b <int>, loc6_a <int>, loc6_b <int>, loc7_a <int>, loc7_b <int>
```

I'll make a new data frame with the first 5 columns, then paste loc1_a:loc1_b, etc.

```
Frogs <- data.frame(Frogs[,1:5],loc1 = paste(Frogs$loc1_a, Frogs$loc1_b, sep=":"),
  loc2 = paste(Frogs$loc2_a, Frogs$loc2_b, sep=":"),
  loc3 = paste(Frogs$loc3_a, Frogs$loc3_b, sep=":"),
  loc4 = paste(Frogs$loc4_a, Frogs$loc4_b, sep=":"),
  loc5 = paste(Frogs$loc5_a, Frogs$loc5_b, sep=":"),
  loc6 = paste(Frogs$loc6_a, Frogs$loc6_b, sep=":"),
  loc7 = paste(Frogs$loc7_a, Frogs$loc7_b, sep=":"))
as_tibble(Frogs)
```

```
## # A tibble: 536 x 12
##       ID OffID Population      X      Y loc1  loc2  loc3  loc4  loc5  loc6
##   <int> <int> <chr>      <dbl>    <dbl> <chr>  <chr> <chr> <chr> <chr> <chr>
## 1    62     0 A21      4426941. 5427173. 340:3~ 422:~ 413:~ 446:~ 121:~ 155:~
## 2    64     0 A21      4426933. 5427178. 334:3~ 424:~ 417:~ 444:~ 122:~ 155:~
## 3    65     0 A21      4426936. 5427173. 338:3~ 417:~ 417:~ 446:~ 135:~ 153:~
## 4    66     0 A21      4426937. 5427174. 340:3~ 422:~ 411:~ 446:~ 122:~ 157:~
## 5    68     0 A21      4426934. 5427171. 336:3~ 417:~ 423:~ 448:~ 119:~ 155:~
## 6    69     0 A21      4426933. 5427166. 336:3~ 422:~ 417:~ 444:~ 122:~ 155:~
## 7    75     0 A21      4426925. 5427175. 340:3~ 422:~ 415:~ 442:~ 121:~ 152:~
## 8    76     0 A21      4426925. 5427173. 338:3~ 417:~ 413:~ 446:~ 126:~ 155:~
## 9    77     0 A21      4426922. 5427174. 344:3~ 422:~ 415:~ 446:~ 121:~ 155:~
## 10   78     0 A21      4426922. 5427174. 342:3~ 417:~ 425:~ 446:~ 121:~ 157:~
## # i 526 more rows
## # i 1 more variable: loc7 <chr>
```

Create 'genind' object:

```
Frogs.genind <- df2genind(X=Frogs[,c(6:12)], sep=":", ncode=NULL,
  ind.names= Frogs$ID, loc.names=names(Frogs[,c(6:12)]),
  pop=Frogs$Population, NA.char="NA", ploidy=2, type="codom",
  strata=NULL, hierarchy=NULL)
```

```
## Warning in df2genind(X = Frogs[, c(6:12)], sep = ":", ncode = NULL, ind.names =
## Frogs$ID, : duplicate labels detected for some individuals; using generic
## labels
```

Get info on genind object:

```
Frogs.genind
```

```
## /// GENIND OBJECT ///////////
##
## // 536 individuals; 7 loci; 109 alleles; size: 291.1 Kb
##
## // Basic content
##   @tab: 536 x 109 matrix of allele counts
##   @loc.n.all: number of alleles per locus (range: 8-26)
##   @loc.fac: locus factor for the 109 columns of @tab
##   @all.names: list of allele names for each locus
##   @ploidy: ploidy of each individual (range: 2-2)
##   @type: codom
##   @call: df2genind(X = Frogs[, c(6:12)], sep = ":", ncode = NULL, ind.names = Frogs$ID,
##     loc.names = names(Frogs[, c(6:12)]), pop = Frogs$Population,
##     NA.char = "NA", ploidy = 2, type = "codom", strata = NULL,
##     hierarchy = NULL)
##
```

```
## // Optional content
## @pop: population of each individual (group size range: 55-128)
summary(Frogs.genind)
```

```
##
## // Number of individuals: 536
## // Group sizes: 69 128 78 75 71 55 60
## // Number of alleles per locus: 19 8 26 9 20 14 13
## // Number of alleles per group: 63 68 55 53 55 73 54
## // Percentage of missing data: 1.79 %
## // Observed heterozygosity: 0.6 0.48 0.72 0.57 0.6 0.6 0.61
## // Expected heterozygosity: 0.83 0.59 0.88 0.75 0.77 0.79 0.85
```

Number of individuals per group/pop is 55-128.