GEA environment,selection and outliers - Melbourne

Bernd Gruber (based on code from Brenna Forrester)

2019-06-10

Contents

Foreword

This tutorial is based on code and comments from Brenna Forrester, Colorado State University.

The Wolf data set is from Schweizer et al. 2016. Genetic subdivision and candidate genes under selection in North American grey wolves Molecular Ecology 25, 380-402. Dryad doi:10.5061/dryad.c9b25.

Multivariate GEA: Redundancy Analysis

RDA is a multivariate ordination technique that analyzes matricies of loci and environmental predictors simultaneously. RDA determines how groups of loci covary in response to the multivariate environment.

RDA is a two-step analysis in which genetic and environmental data are analyzed using multivariate linear regression, producing a matrix of fitted values. Then PCA of the fitted values is used to produce canonical axes, which are linear combinations of the predictors.

More information on RDA & other multivariate GEAs (such as Random Forest) can be found in our paper: Forester BR, Lasky JR, Wagner HH, Urban DL (2018) Comparing methods for detecting multilocus adaptation with multivariate genotype-environment associations Molecular Ecology 27, 2215-2233.

RDA can be used on both individual and population-based sampling designs. The distinction between the two may not be straightforward in all cases. A simple guideline would be to use an individual-based framework when you have individual coordinates for most of your samples, and the resolution of your environmental data would allow for a sampling of environmental conditions across the site/study area. More on RDA with population level data in the notes below.

To run RDA on population level data, calculate allele frequencies (columns) for populations (rows). Applying a Hellinger transformation/standardization to the data will ensure it behaves well in the RDA, e.g. if our input of population allele frequencies was called “pop.data”:

```
pop.data.hellinger <- decostand(pop.data, method = "hellinger")  # square root of (allele frequency / popula
```

RDA does not require corrections for multiple tests because it analyzes the genomic and environmental data simultaneously!

I highly recommend the following book for more information on interpreting RDA output from vegan: Borcard et al. (2011) Numerical Ecology with R.

Before we can start with our redundancy analysis we need to make sure, we select “suitable” environmental predictors. Meaning the predictors should have a “good” chance to have an effect on our response (the genetic structure) and should not be correlated between each other.

Please see Dormann et al. (2013) Ecography for a discussion of these issues. Generally, the $|r| > 0.7$ “rule of thumb” is a good guideline for removing correlated predictors. We will also check for multicollinearity below using Variance Inflation Factors.

We will use the Wolf data set mentioned above to run an examplary redundancy analysis. To convert a genlight object into the correct format simply use:

```
gen <- as.matrix(gl)
```

Load the genetic data set from a csv file:

```
## [1]    94 10000
```

```
##      chr9.22100598 chr2.19649684
## [1,]             0             0
## [2,]             1             0
## [3,]             0             2
## [4,]             0             0
## [5,]             1             1
##      chr38.14463749 chr13.20658362
## [1,]              0              0
## [2,]              0              2
## [3,]              0              1
## [4,]              0              2
## [5,]              1              0
##      chr23.16849044
## [1,]              0
## [2,]              1
## [3,]              0
## [4,]              1
## [5,]              0
```

```
# Read from github

fp <- "https://raw.githubusercontent.com/green-striped-gecko/dartRworkshop/master/data"

gen <- read.csv(file.path(fp, "Wolf_Gen_sample_6pops_94indiv.csv"))
dim(gen)  # 94 individuals in rows; 10,000 SNPs in columns
gen[1:5, 1:5]  # coded as 0/1/2
```

Load the environmental data set

```
##  [1] "individual"
##  [2] "ecotype"
##  [3] "long"
##  [4] "lat"
##  [5] "ann_mean_temp"
##  [6] "mean_diurnal_range"
##  [7] "temp_seasonality"
##  [8] "max_temp_warmest_month"
##  [9] "min_temp_coldest_month"
```

```
## [10] "ann_precip"
## [11] "precip_seasonality"
## [12] "precip_coldest_quarter"
## [13] "land_cover"
## [14] "ndvi"
## [15] "elev"
## [16] "percent_tree_cover"
```

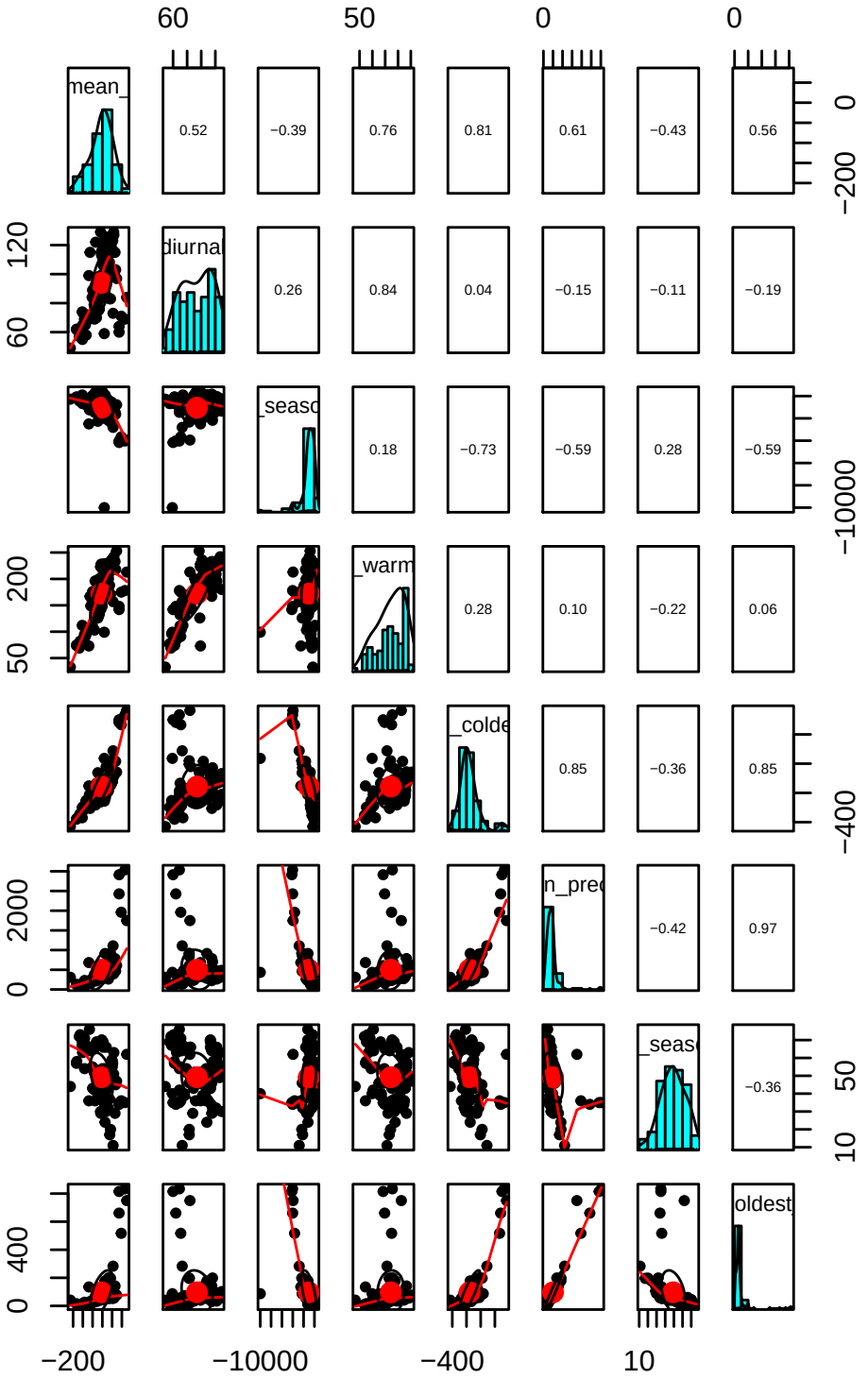
```
# Read from github

fp <- "https://raw.githubusercontent.com/green-striped-gecko/dartRworkshop/master/data"

env <- read.csv(file.path(fp, "Wolf_Env_6pops_94indiv.csv"))
names(env)
```

Or in case you are using a genlight object you should be able to use the @other$ind.metrics table from a genlight data set.

```
env <- @other$ind.metrics
```

Check for multicollinearity between predictors (|r|>0.7)

```
library(psych)
pairs.panels(env[, 5:12])  #good for contineous predictors only
```

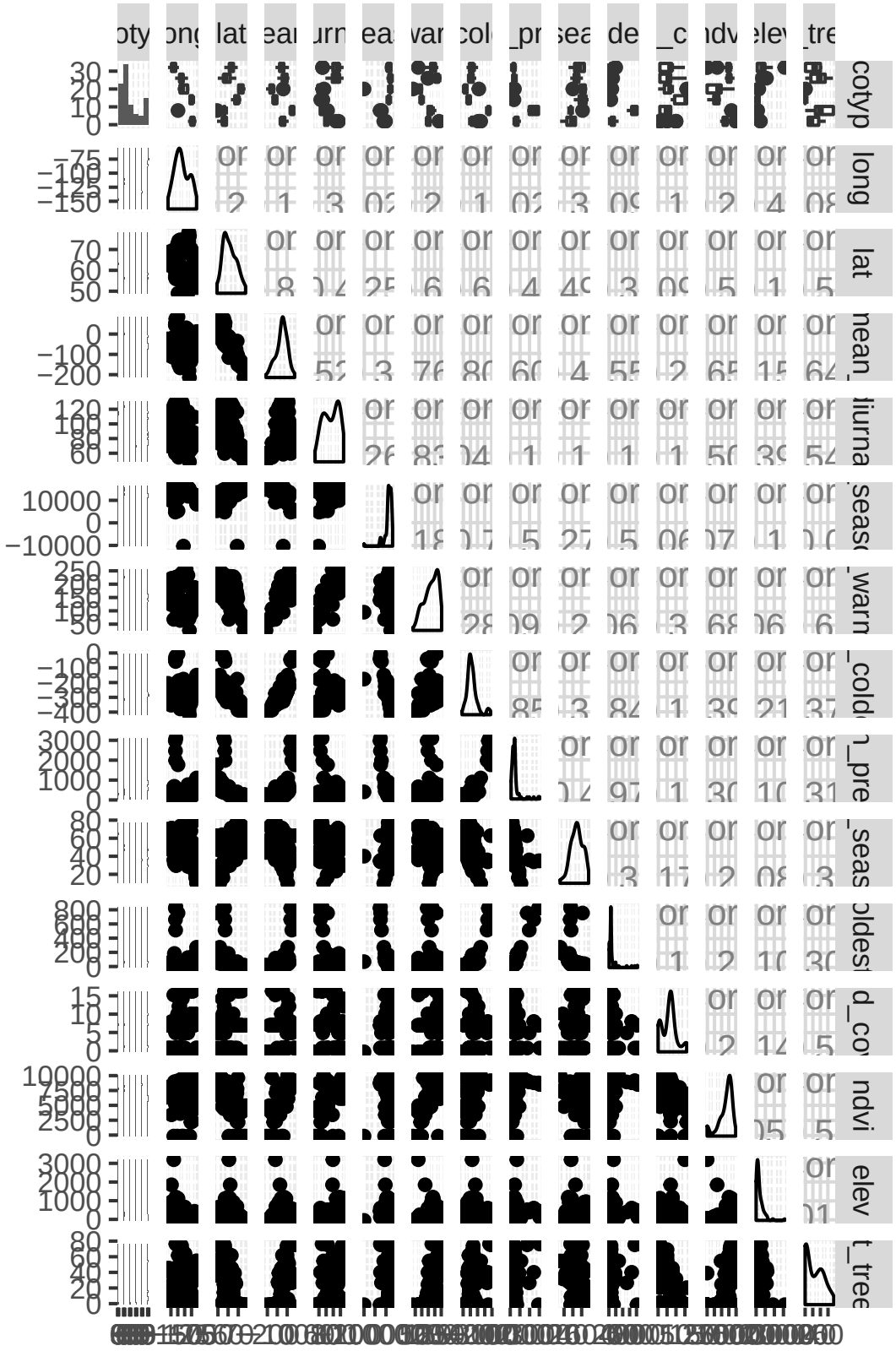

mean_
diurna
_seasc
_warm
_colde
n_prec
_seas
oldest
0.52
−0.39
0.76
0.81
0.61
−0.43
0.56
0.26
0.84
0.04
−0.15
−0.11
−0.19
0.18
−0.73
−0.59
0.28
−0.59
0.28
0.10
−0.22
0.06
0.85
−0.36
0.85
−0.42
0.97
−0.36
60
50
0
0
120
60
200
50
2000
0
400
0
−200
0
−10000
−400
50
10
−200
−10000
−400
10

```r
library(GGally)
```

```
## Loading required package: ggplot2

##
## Attaching package: 'ggplot2'

## The following objects are masked from 'package:psych':
##
##     %+%, alpha
```

```r
ggpairs(env[, -1])  #allows also factors [except individual]
```

```
## `stat_bin()` using `bins = 30`. Pick
## better value with `binwidth`.

## `stat_bin()` using `bins = 30`. Pick
## better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick
## better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick
## better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick
## better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick
## better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick
## better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick
## better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick
## better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick
## better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick
## better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick
## better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick
## better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick
## better value with `binwidth`.
```



```
library(vegan)
```

```
## Loading required package: permute
```

```
## Loading required package: lattice
```

```
## This is vegan 2.5-4
```

```
pred <- env[, 5:10]
# colnames(pred)
# <-c('AMT','MDR','sdT','AP','cvP','NDVI','Elev','Tree')

wolf.rda <- rda(gen ~ ., data = as.data.frame(pred),
    scale = T)  # vegan wants data frames, not matrices
```

Well done!!

That is the end. Well done you finished. Feel free to have another go or have a well deserved beverage!!!