Coevolution with a seed bank

23 August, 2022

Analyze composition of mutations from pooled population sequencing

Setup Work Environment

```
# Load dependencies
library(here)
```

```
## here() starts at C:/Users/danschw/GitHub/coevolution/coevo-seedbank-seq
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
require("vegan")
```

```
## Loading required package: vegan

## Loading required package: permute

## Loading required package: lattice

## This is vegan 2.6-2
```

Load data

```
mutdat <- read_csv(here("data/mult_host.csv")) %>%
  rename(trt = 1)
```

```
## New names:
## * `` -> ...1
```

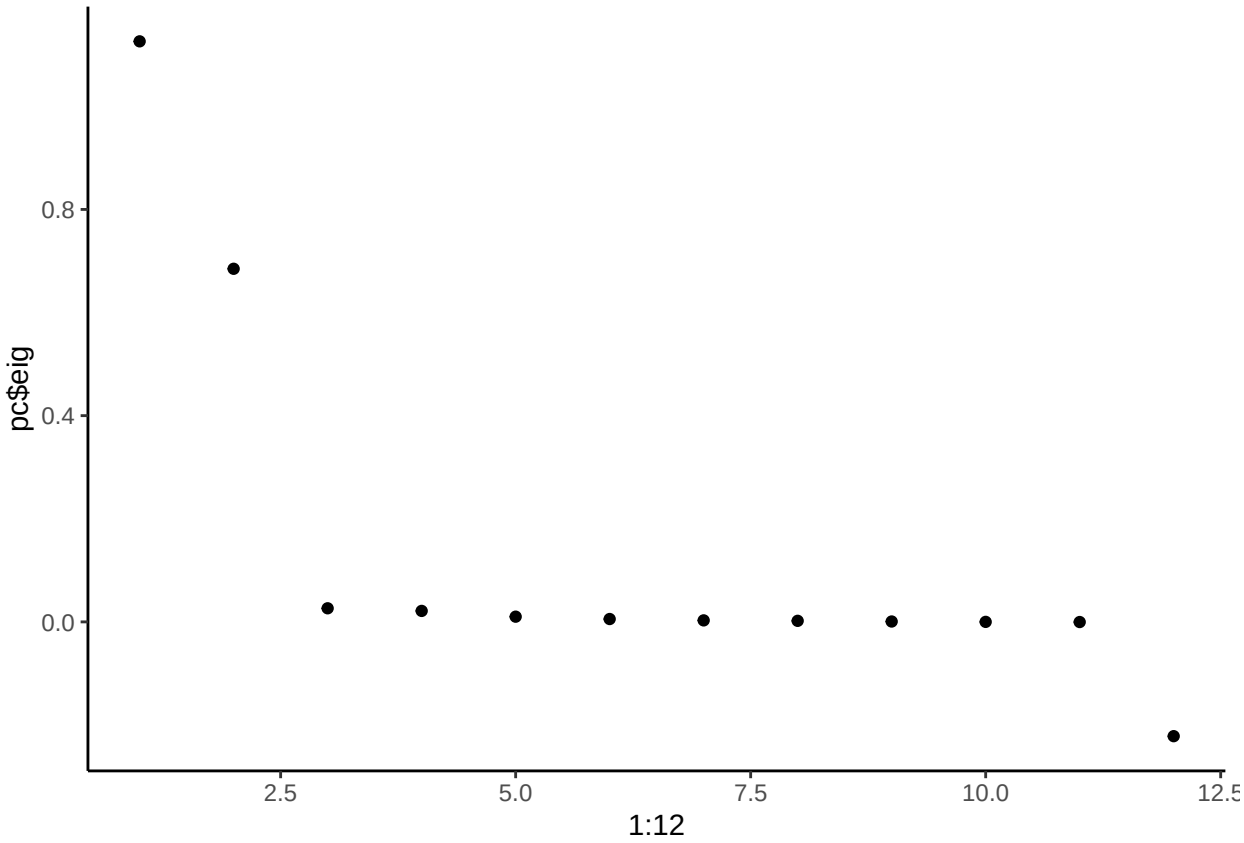
```
## Rows: 12 Columns: 375
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr   (1): ...1
## dbl (374): A8O17_RS00100, A8O17_RS00130, A8O17_RS00185, A8O17_RS00345, A8O17...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

PCoA procedures

```
# Define treatments and data
seed <- str_detect(mutdat$trt, "long")
phage <- str_detect(mutdat$trt, "SPO1")
# seed <- c(1,1,1,1,1,1,0,0,0,0,0,0)
# phage <- c(0,0,0,1,1,1,0,0,0,1,1,1)
mut <- cbind(seed, phage, mutdat[,2:ncol(mutdat)])

# "manhattan", "euclidean", "canberra", "clark", "bray", "kulczynski", "jaccard", "gower", "altGower",

# Calculate pairwise distances
mut.dist <- vegdist(mut, method = "bray", binary = "FALSE")

# Principal Coordinates Analysis (PCoA)
pc <- cmdscale(mut.dist, eig = TRUE, k = 3)
explainvar1 <- round(pc$eig[1] / sum(pc$eig), 3) * 100
explainvar2 <- round(pc$eig[2] / sum(pc$eig), 3) * 100
explainvar3 <- round(pc$eig[3] / sum(pc$eig), 3) * 100
sum.eig <- sum(explainvar1, explainvar2, explainvar3)
##  two first PCoAs explain >100% variation
```

Consulting with WRS, he noticed we have a negative eigenvalue on the order of the second largest positive eigenvalue, which would be considered a large negative eigenvalue.

```
qplot(1:12,pc$eig)+ theme_classic()
```

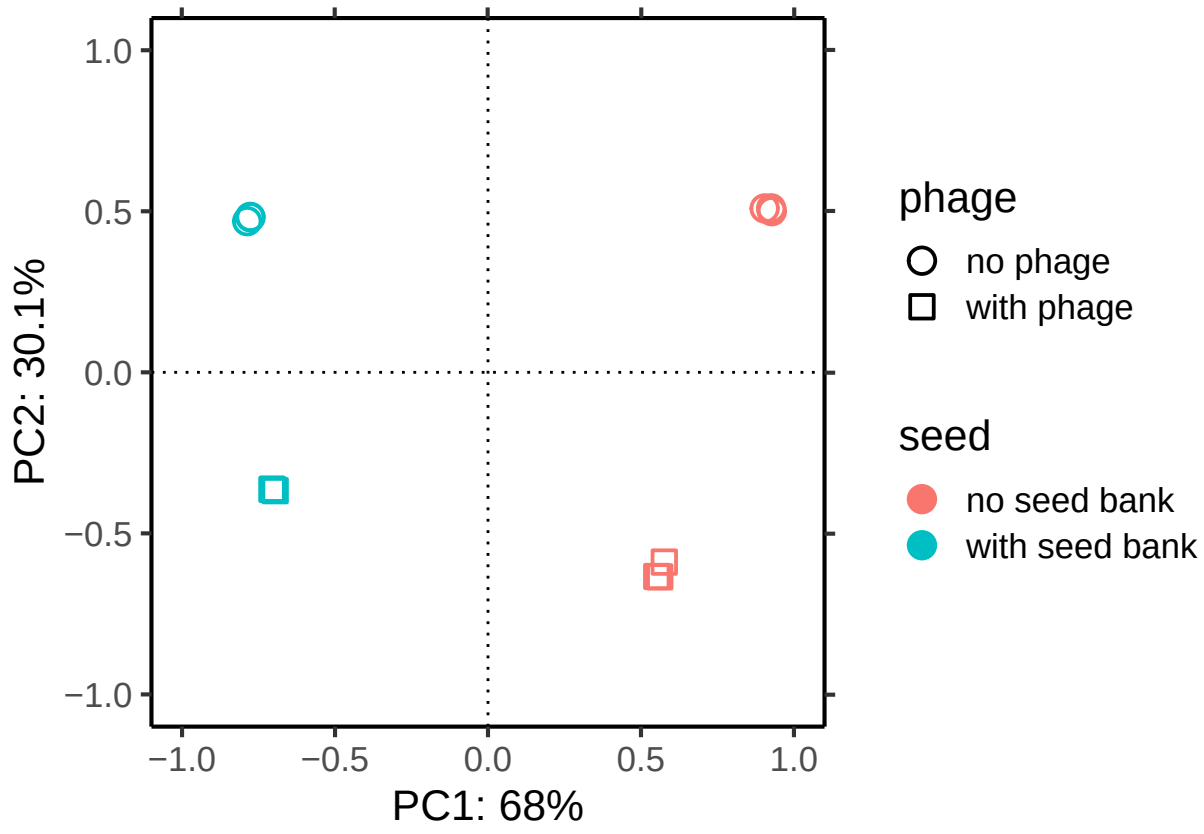


WRS suggested we try a PCA, since prcomp uses singular value decomposition instead of eigendecomposition on the covariance matrix, so it doesn't return negative.

PCA Plot

```
pca <- prcomp(mut.dist)

var_explained <- pca$sdev^2/sum(pca$sdev^2)

p <- cbind(mut[,1:2],pca$x) %>%
  as.data.frame %>%
    mutate(seed = if_else(seed==1, "with seed bank", "no seed bank"),
          phage= if_else(phage==1, "with phage", "no phage") ) %>%
  ggplot(aes(x=PC1,y=PC2)) +
  geom_point(aes(color = seed, shape = phage), size=4, stroke=1)+
  # geom_polygon(data = dl, linetype = 3 ,fill="transparent",
  #              aes(x=x, y =y, group = interaction(seed, phage),color = seed))+
  theme_bw(base_size=32) +
  labs(x=paste0("PC1: ",round(var_explained[1]*100,1),"%"),
       y=paste0("PC2: ",round(var_explained[2]*100,1),"%")) +
  geom_hline(yintercept = 0, linetype = 3)+
  geom_vline(xintercept = 0, linetype = 3)+
  scale_shape_manual(values = c(21,22))+
  scale_x_continuous(sec.axis = dup_axis(name = NULL, labels = NULL),
```

```
                    limits = c(-1,1)) +
  scale_y_continuous(sec.axis = dup_axis(name = NULL, labels = NULL),
                     limits = c(-1,1))+
  theme_classic(base_size = 16)

p
```

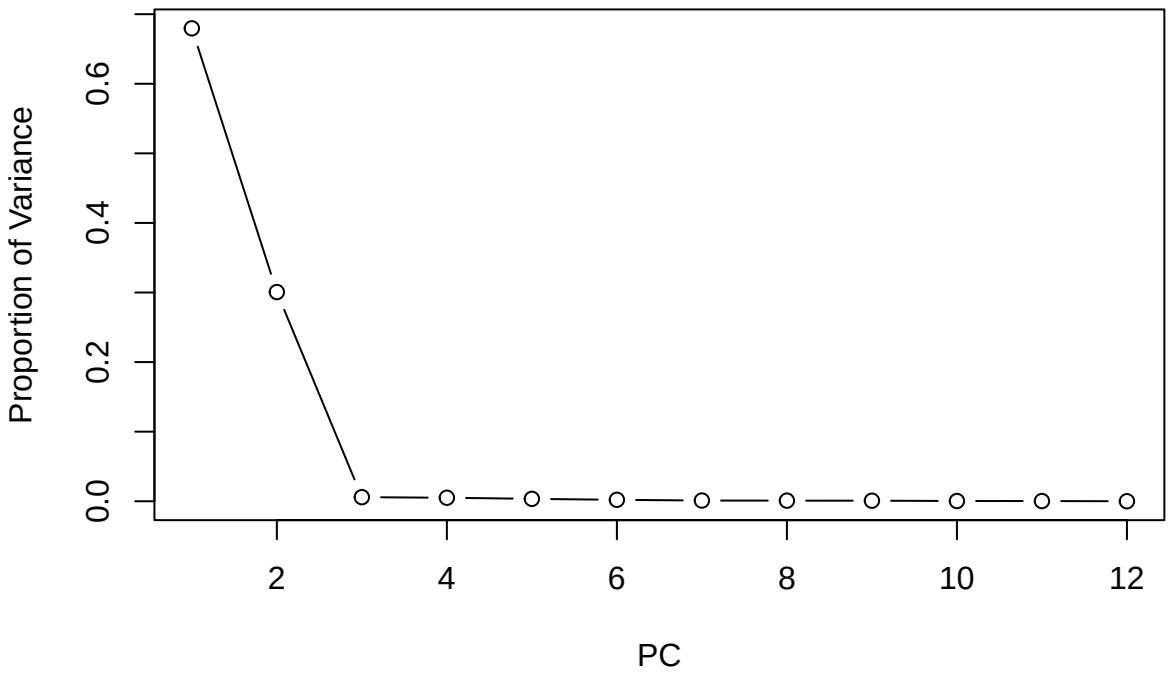



```
ggsave(here("analysis","PCA_gg.png"),p, width = 5, height = 3)
```

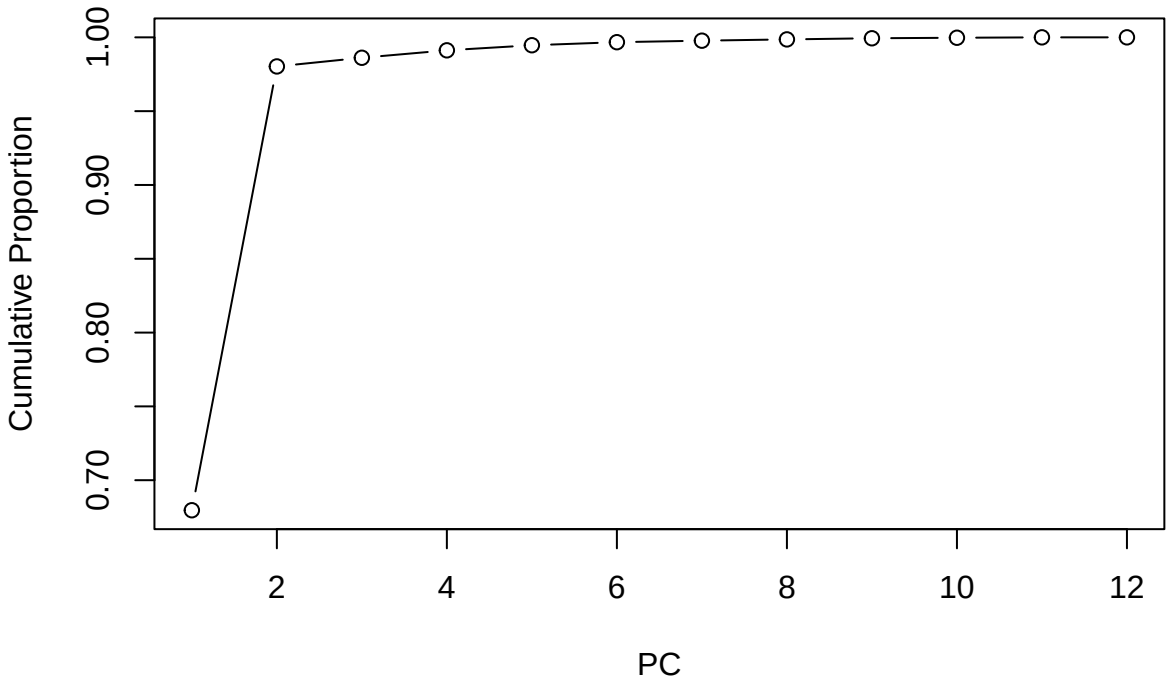
Elbow plot

How many PCs to include in stats?

```
sum.pca <- summary(pca)

prop_var <- sum.pca$importance[2,]
plot(1:12,prop_var, type = "b", xlab = "PC", ylab = "Proportion of Variance")
```




```
cum_var <- sum.pca$importance[3,]
plot(1:12,cum_var, type = "b", xlab = "PC", ylab = "Cumulative Proportion")
```



First to PCs explain 98.03% of the variartion

```
# PERMANOVA on PCA
seed <- mut[,1]
phage <- mut[,2]
perm <- adonis2(pca$x[,c(1:2)] ~ seed * phage, method = "euclidean", binary = FALSE)
perm
```

```
## Permutation test for adonis under reduced model
## Terms added sequentially (first to last)
## Permutation: free
## Number of permutations: 999
##
## adonis2(formula = pca$x[, c(1:2)] ~ seed * phage, method = "euclidean", binary = FALSE)
##            Df SumOfSqs      R2        F Pr(>F)
## seed        1   6.6299 0.67680 22880.25  0.001 ***
## phage       1   2.9624 0.30241 10223.37  0.001 ***
## seed:phage  1   0.2013 0.02055   694.85  0.001 ***
## Residual    8   0.0023 0.00024
## Total      11   9.7960 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

In summary, we analyze the composition of genes under selection by PCA analysis on multiplicity data . PERMANOVA on the first two axis (that together explain >98% of variation) shows that treatments significantly alter the composition, as does the treatment interaction. The F value shows that the seed-bank has the strongest effect, and that the interaction is much weaker, but all are significant

Ellipses

The ggplot function of stat_ellipse does not allow CI ellipses on less than 4 data points. We have three points per treatment. However three points should be alowed "because your CI depends on the variance, which takes two degrees of freedom".

According to stat_ellipses help "The method for calculating the ellipses has been modified from car::dataEllipse (Fox and Weisberg, 2011)". The limit on 3 points does not exist in the original function.

```
library(car)
```

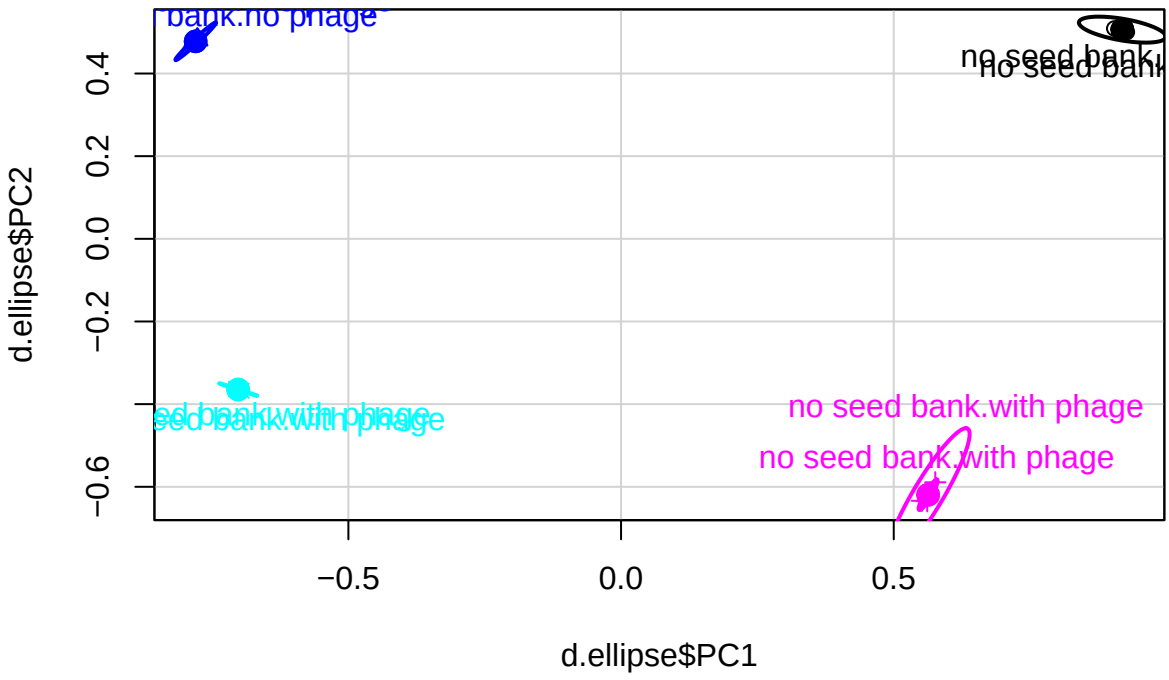
```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```
## The following object is masked from 'package:purrr':
##
##     some
```

```
d.ellipse <- cbind(mut[,1:2],pca$x) %>%
  as.data.frame %>%
  mutate(seed = if_else(seed, "with seed bank", "no seed bank"),
         phage= if_else(phage, "with phage", "no phage"),
         grp=interaction(seed,phage))

el <- dataEllipse(d.ellipse$PC1, d.ellipse$PC2, groups = d.ellipse$grp)
```




```
# unpack list
dl <- rbind(
cbind("no seed bank.no phage",el$`no seed bank.no phage`$`0.95`),
cbind("with seed bank.no phage",el$`with seed bank.no phage`$`0.95`),
cbind("no seed bank.with phage",el$`no seed bank.with phage`$`0.95`),
cbind("with seed bank.with phage",el$`with seed bank.with phage`$`0.95`)
)


dl <- dl %>%
  as_tibble() %>%
  mutate(x= as.numeric(x), y=as.numeric(y)) %>%
  separate(V1, into = c("seed", "phage"),remove = F, sep = "\\.")
```

```
## Warning: The `x` argument of `as_tibble.matrix()` must have unique column names if `.name_repair` is
## Using compatibility `.name_repair`.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
```

PCA with ellipses

```
p <- cbind(mut[,1:2],pca$x) %>%
  as.data.frame %>%
```

```
    mutate(seed = if_else(seed==1, "with seed bank", "no seed bank"),
           phage= if_else(phage==1, "with phage", "no phage") ) %>%
  ggplot(aes(x=PC1,y=PC2)) +
    geom_polygon(data = dl, linetype = 2 ,fill="transparent", size =  1,
                 aes(x=x, y =y, group = interaction(seed, phage), color = seed))+
  geom_point(aes(color = seed, shape = phage), size=4, stroke=1)+

  theme_bw(base_size=32) +
  labs(x=paste0("PC1: ",round(var_explained[1]*100,1),"%"),
       y=paste0("PC2: ",round(var_explained[2]*100,1),"%")) +
  geom_hline(yintercept = 0, linetype = 3)+
  geom_vline(xintercept = 0, linetype = 3)+
  scale_shape_manual(values = c(21,22))+
  scale_x_continuous(sec.axis = dup_axis(name = NULL, labels = NULL),
                     limits = c(-1,1)) +
  scale_y_continuous(sec.axis = dup_axis(name = NULL, labels = NULL),
                     limits = c(-1,1))+
  theme_classic(base_size = 16)

p
```