Coevolution with a seed bank

03 February, 2023

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 1.3.2 --

## v ggplot2 3.4.0      v purrr   0.3.5
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.3      v forcats 0.5.2

## Warning: package 'ggplot2' was built under R version 4.2.2

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

```
library(vegan)
```

```
## Loading required package: permute
## Loading required package: lattice
## This is vegan 2.6-2
```

```
library(BiodiversityR)
```

```
## Loading required package: tcltk
## BiodiversityR 2.14-4: Use command BiodiversityRGUI() to launch the Graphical User Interface;
## to see changes use BiodiversityRGUI(changeLog=TRUE, backward.compatibility.messages=TRUE)
```

Load data

Matrix of multiplicity data organized as population X gene.

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

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

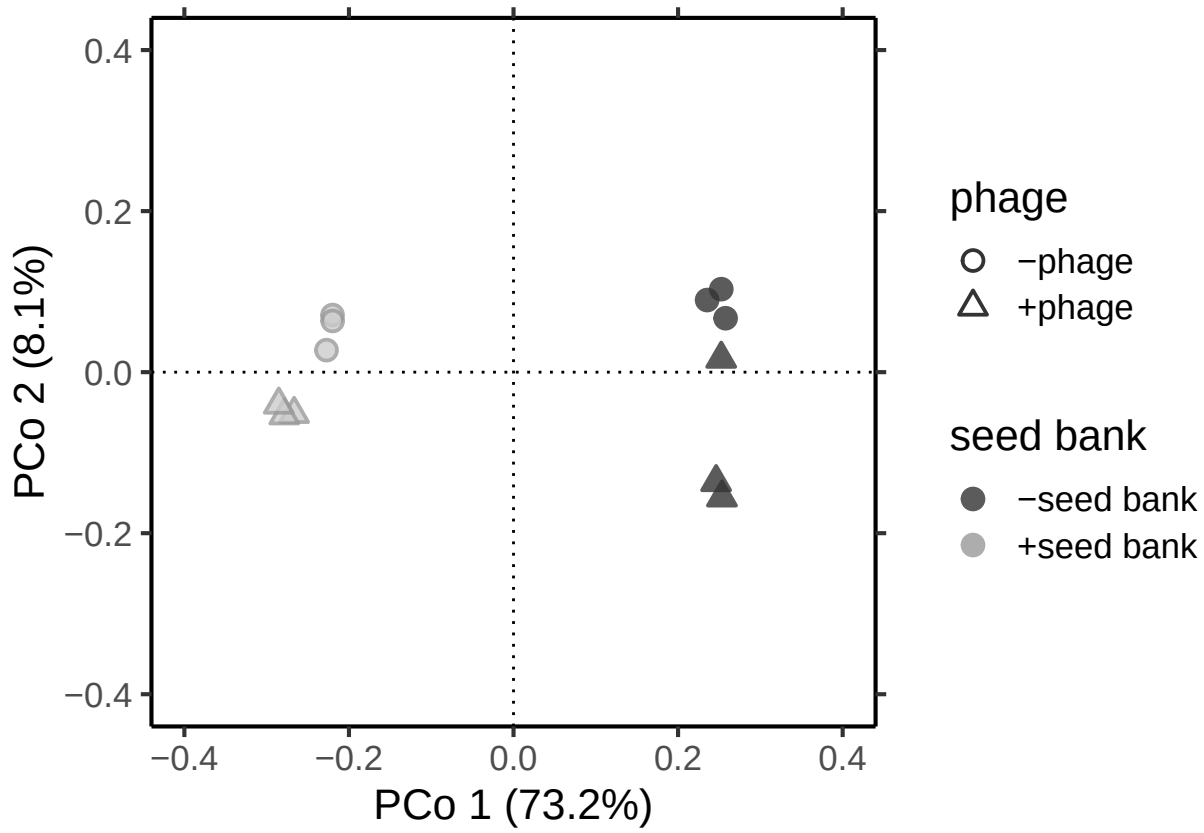
PCoA procedures

```
# Define treatments and data
seed <- str_detect(mutdat$trt, "long")
phage <- str_detect(mutdat$trt, "SPO1")

# multiplicity data only
mut <- mutdat %>% select(-trt)


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


# Principal Coordinates Analysis (PCoA)
pc <- cmdscale(mut.dist, eig = TRUE, k = nrow(mut)-1)
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)



p <-
  as_tibble(pc$points, .name_repair = "universal" ) %>%
  rename_with(~gsub("...", "PCoA", .x, fixed = TRUE)) %>%
  mutate(seed = seed, phage = phage) %>%
  relocate(seed,phage,.before = 1) %>%
    mutate(seed = if_else(seed==1, "+seed bank", "-seed bank"),
         phage= if_else(phage==1, "+phage", "-phage") ) %>%
  ggplot(aes(x=PCoA1,y=PCoA2)) +
  geom_point(aes(color = seed, fill = seed, shape = phage), size=3, stroke=1,alpha=0.8)+
  # 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("PCo 1 (",round(explainvar1,1),"%)"),
       y=paste0("PCo 2 (",round(explainvar2,1),"%)")) +
  geom_hline(yintercept = 0, linetype = 3)+
  geom_vline(xintercept = 0, linetype = 3)+
  scale_shape_manual(values = c(21,24))+
  scale_fill_grey(end = 0.8, name = "seed bank")+
  scale_color_grey(end = 0.6, name = "seed bank")+
  scale_x_continuous(sec.axis = dup_axis(name = NULL, labels = NULL),
                     limits = c(-0.4,0.4)) +
  scale_y_continuous(sec.axis = dup_axis(name = NULL, labels = NULL),
                     limits = c(-0.4,0.4))+
  theme_classic(base_size = 16)
```

```
## New names:
## * `` -> `...1`
## * `` -> `...2`
## * `` -> `...3`
## * `` -> `...4`
## * `` -> `...5`
## * `` -> `...6`
## * `` -> `...7`
## * `` -> `...8`
## * `` -> `...9`
## * `` -> `...10`
## * `` -> `...11`
```

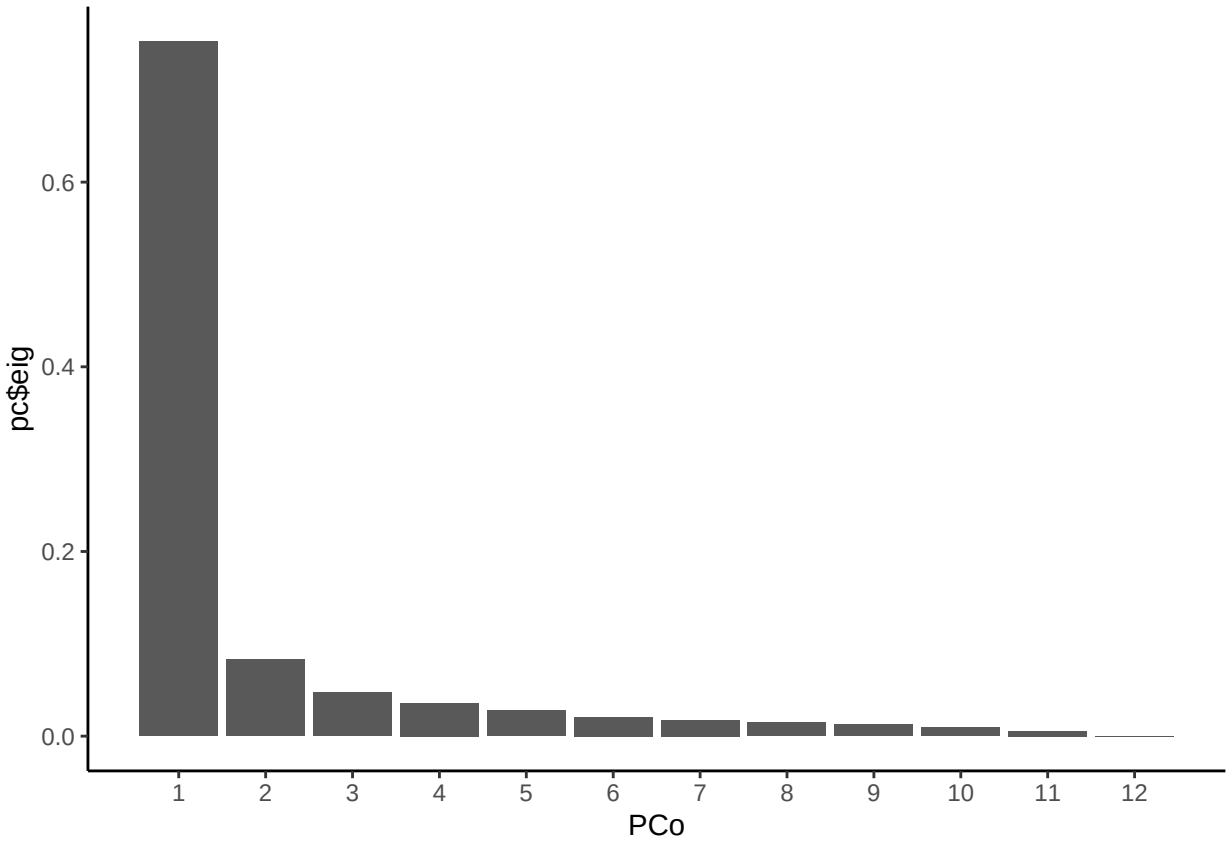
```
ggsave(here("analysis","PCoA_mult_host.png"),p, width = 5, height = 3)

p
```



No negative eigenvalues

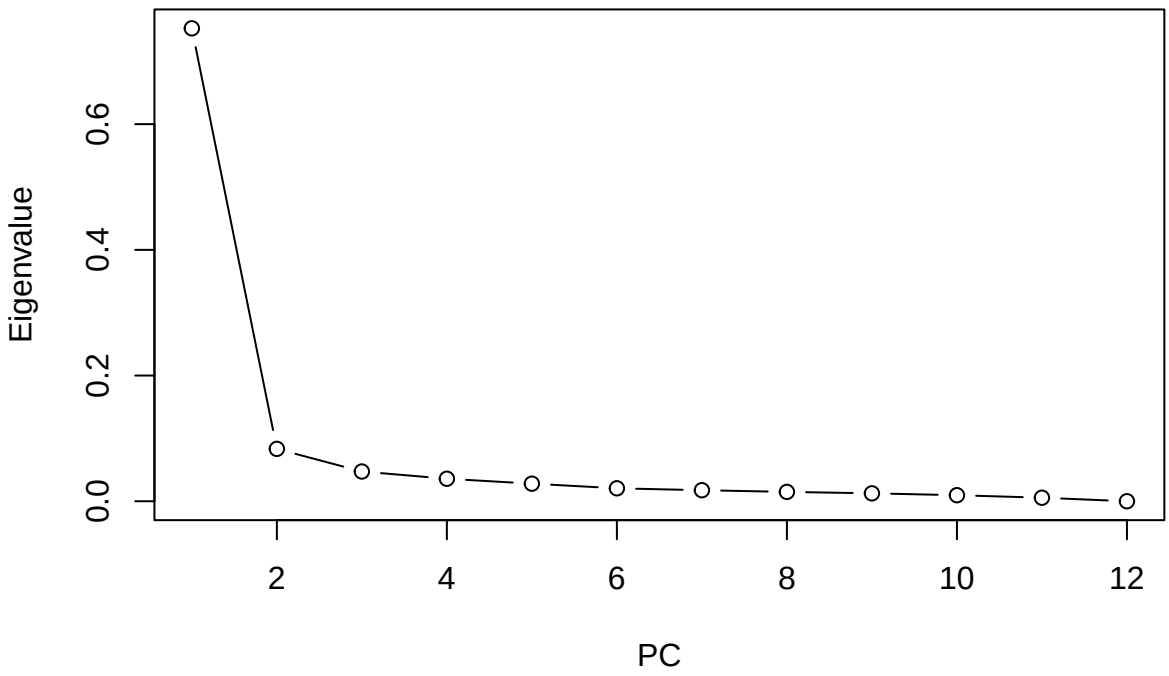
```
qplot(1:12,pc$eig, geom = "col")+
  theme_classic()+
  scale_x_continuous(breaks = 1:12)+
  xlab("PCo")
```

```
## Warning: `qplot()` was deprecated in ggplot2 3.4.0.
```

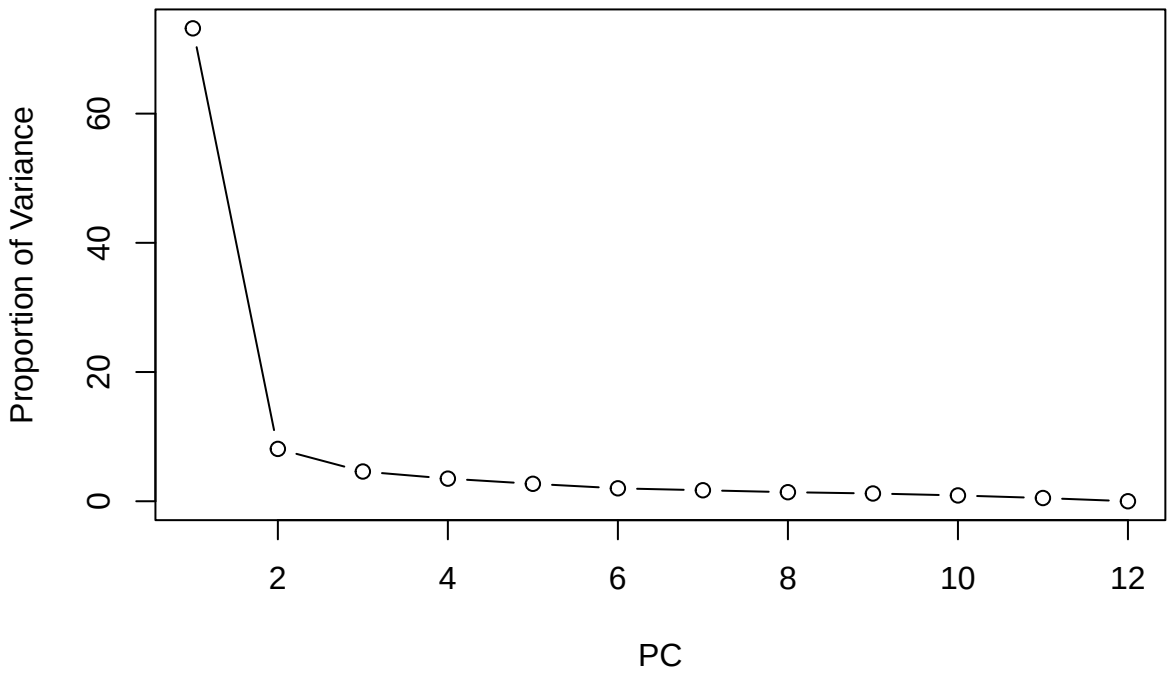


How many PCs to include in stats?

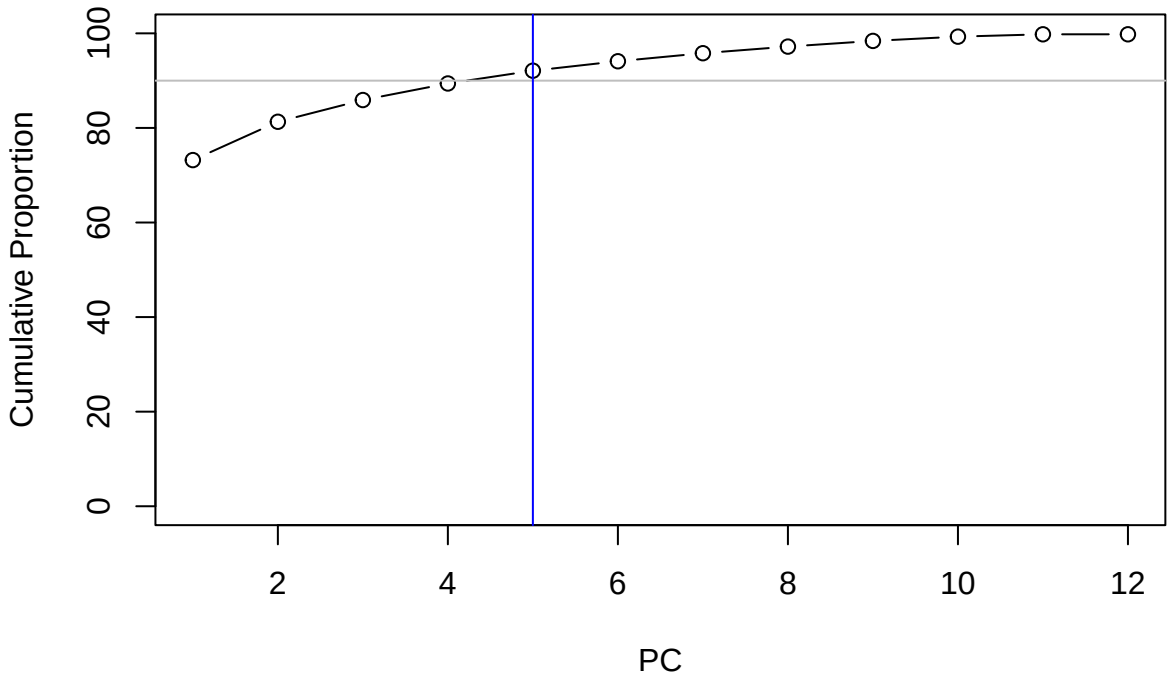
```
plot(1:length(pc$eig),pc$eig,type = "b",
     xlab = "PC", ylab = "Eigenvalue")
```




```
prop_var <- round(pc$eig / sum(pc$eig), 3) * 100
plot(1:length(pc$eig),prop_var, type = "b",
     xlab = "PC", ylab = "Proportion of Variance")
```




```
# PCs explaining 90% variation
pc_var90 <- min(which(cumsum(prop_var)>90))
plot(1:length(pc$eig),cumsum(prop_var), type = "b",
     xlab = "PC", ylab = "Cumulative Proportion", ylim = c(0,100))
abline(v=pc_var90,h=90, col = c("grey", "blue"))
```



First 5 PCs explain >90% of the variartion

PERMANOVA

```
# # on multiplicity data
# perm <-
#   adonis2(mut.dist ~ seed * phage,
#           binary = FALSE, permutations = 9999)

# on PCs explaining >90% var
perm <-
  adonis2(pc$points[,1:pc_var90] ~ seed * phage,method = "euclidean",
          binary = FALSE, permutations = 9999)

perm
```

```
## Permutation test for adonis under reduced model
## Terms added sequentially (first to last)
## Permutation: free
## Number of permutations: 9999
##
## adonis2(formula = pc$points[, 1:pc_var90] ~ seed * phage, permutations = 9999, method = "euclidean",
##            Df SumOfSqs      R2       F Pr(>F)
```

```
## seed        1  0.74770 0.78972 65.2634 0.0002 ***
## phage       1  0.07029 0.07424  6.1349 0.0213 *
## seed:phage  1  0.03715 0.03923  3.2424 0.0793 .
## Residual    8  0.09165 0.09680
## Total      11  0.94678 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Gene correlations

```
# genes in delta6 ---------------------------------
delta6_168 <- read_csv(here("data/teichoic_acid","delta6_168_cds_matched.csv"),trim_ws = T, name_repair
```

```
## Rows: 3913 Columns: 6
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (4): seqname, locus_tag.d6, strand, locus_tag.168
## dbl (2): start, end
##
## 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.
```

```
# categories_168 <- read_csv(here("data/teichoic_acid","geneCategories-2022-06-27.csv"),trim_ws = T, na
SW.export_168 <- read_csv(here("data/teichoic_acid","subtiwiki.gene.export.2022-06-27.csv"),trim_ws = T
```

```
## Rows: 6756 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (4): locus, title, description, function
##
## 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.
```

```
d.genes <- left_join(delta6_168,SW.export_168 , by = c("locus_tag.168"="locus"))


# Test correlation to PCoA axes ----------------------------------------
gene.corr <- add.spec.scores(pc, mut, method = "cor.scores")$cproj
gene.corr <-
  tibble(gene = rownames(gene.corr)) %>%
  bind_cols(as_tibble(gene.corr ))

# Genes correlated with PCo1 -------------------------------

fit <- envfit(pc, mut, choices=1, perm = 999)

d.fit1 <- tibble(gene = names(fit$vectors$r),
       r = fit$vectors$r,
       pvals = fit$vectors$pvals)
```

```r
# combine with Correlation for significant genes
sig_genes1 <- d.fit1 %>%
  filter(pvals<0.05) %>%
  left_join(., gene.corr %>% select(gene, cor = Dim1))
```

```
## Joining, by = "gene"
```

```r
# add annotations
sig_genes1 <- sig_genes1 %>%
  left_join(., d.genes, by = c("gene" = "locus_tag.d6"))

# export positively correlated
sig_genes1 %>%
  filter(cor > 0) %>%
  arrange(desc(abs(cor))) %>%
  select(locus_tag.d6=gene, locus_tag.168, title, description, `function`, strand, cor,   P_value = pval
  write_csv(here("data", "significant_genes_pc1_positive.csv"))

# export negatively correlated
sig_genes1 %>%
  filter(cor < 0) %>%
  arrange(desc(abs(cor))) %>%
  select(locus_tag.d6=gene, locus_tag.168, title, description, `function`, strand, cor,   P_value = pval
  write_csv(here("data", "significant_genes_pc1_negative.csv"))


# Genes correlated wit PCo2 ------------------------------------------------
fit <- envfit(pc, mut, choices=2, perm = 999)

d.fit2 <- tibble(gene = names(fit$vectors$r),
       r = fit$vectors$r,
       pvals = fit$vectors$pvals)

# Correlation of significant genes
sig_genes2 <- d.fit2 %>%
  filter(pvals<0.05) %>%
  left_join(., gene.corr %>% select(gene, cor = Dim2))
```

```
## Joining, by = "gene"
```

```r
# add annotations
sig_genes2 <- sig_genes2 %>%
  left_join(., d.genes, by = c("gene" = "locus_tag.d6"))

# export positively correlated
sig_genes2 %>%
  filter(cor > 0) %>%
  arrange(desc(abs(cor))) %>%
  select(locus_tag.d6=gene, locus_tag.168, title, description, `function`, strand, cor,   P_value = pval
  write_csv(here("data", "significant_genes_pc2_positive.csv"))

# export negatively correlated
```

```
sig_genes2 %>%
  filter(cor < 0) %>%
  arrange(desc(abs(cor))) %>%
  select(locus_tag.d6=gene, locus_tag.168, title, description, `function`, strand, cor,  P_value = pval
  write_csv(here("data", "significant_genes_pc2_negative.csv"))
```

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)
```

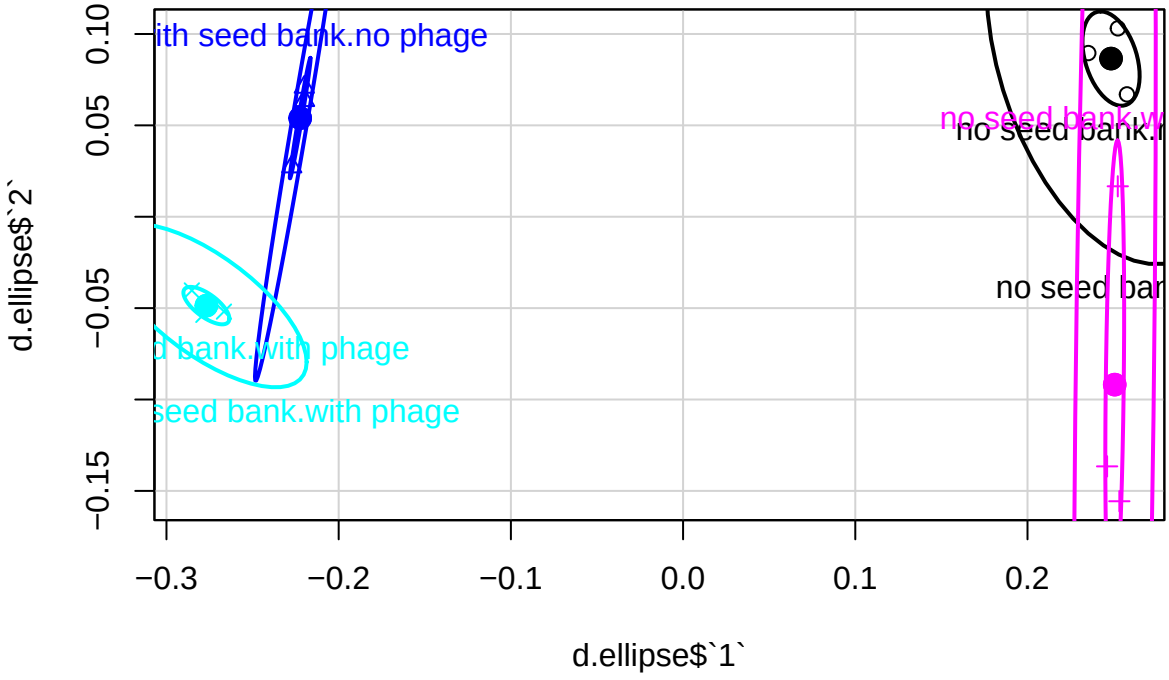
```
## Loading required package: carData
```

```
##
## 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],pc$points) %>%
  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$`1`, d.ellipse$`2`, 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 omitted as of tibble 2.0.0.
## i Using compatibility `.name_repair`.
```

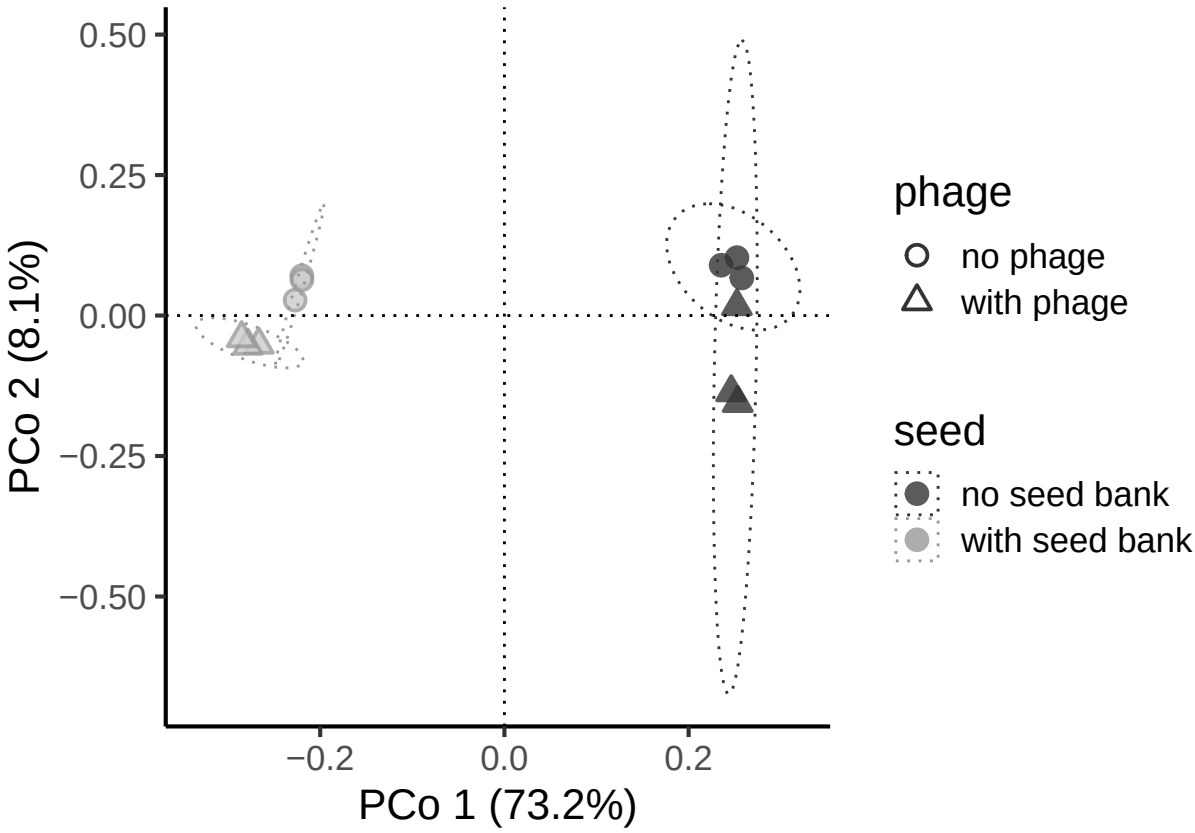
PCA with ellipses

```
p <-
  as_tibble(pc$points, .name_repair = "universal" ) %>%
  rename_with(~gsub("...", "PCoA", .x, fixed = TRUE)) %>%
```

```
  mutate(seed = seed, phage = phage) %>%
  relocate(seed,phage,.before = 1) %>%
    mutate(seed = if_else(seed==1, "with seed bank", "no seed bank"),
          phage= if_else(phage==1, "with phage", "no phage") ) %>%
  ggplot(aes(x=PCoA1,y=PCoA2)) +
  geom_point(aes(color = seed, fill = seed, shape = phage), size=3, stroke=1,alpha=0.8)+
  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("PCo 1 (",round(explainvar1,1),"%)"),
       y=paste0("PCo 2 (",round(explainvar2,1),"%)")) +
  geom_hline(yintercept = 0, linetype = 3)+
  geom_vline(xintercept = 0, linetype = 3)+
  scale_shape_manual(values = c(21,24))+
  scale_fill_grey(end = 0.8)+
  scale_color_grey(end = 0.6)+
  # scale_x_continuous(sec.axis = dup_axis(name = NULL, labels = NULL),
  #                    limits = c(-0.4,0.4)) +
  # scale_y_continuous(sec.axis = dup_axis(name = NULL, labels = NULL),
  #                    limits = c(-0.4,0.4))+
  theme_classic(base_size = 16)
```

```
## New names:
## * `` -> `...1`
## * `` -> `...2`
## * `` -> `...3`
## * `` -> `...4`
## * `` -> `...5`
## * `` -> `...6`
## * `` -> `...7`
## * `` -> `...8`
## * `` -> `...9`
## * `` -> `...10`
## * `` -> `...11`
```

```
# ggsave(here("analysis","PCA_hellinger2.png"),p, width = 5, height = 3)

p
```


0.50
0.25
0.00
−0.25
−0.50
PCo 2 (8.1%)
−0.2
0.0
0.2
PCo 1 (73.2%)
phage
no phage
with phage
seed
no seed bank
with seed bank