02-test-pca-w-unknowns

14 December 2022

Testing individual-based analyses (pca) with unknown samples from AK.

The haplotype file for the baseline samples comes from `10-complete-downsamp-self...`

I'll read in the baseline data and the unknown data, then filter both appropriately (missing data), and try reformating the dataframe and then converting it to a genid object for adegenet.

```
knitr::opts_chunk$set(warning = FALSE, message = FALSE)
knitr::opts_knit$set(root.dir = '~/Desktop/RE_BS/stock review 2025 work/')
```

```
# baseline data - curated, 997 indivs
data <- read.csv('observed_unfiltered_haplotype.csv')
```

```
# slim that down for the pca
baseline_for_combo <- data %>%
  select(indiv.ID, locus, rank, haplo)

colnames(baseline_for_combo) <- c('id','locus','rank','haplo')

head(baseline_for_combo)
```

```
##                 id       locus rank         haplo
## 1 T-15-LS-0028-BS tag_id_1049    1 AACCCGAGCCGCG
## 2       UW-201554 tag_id_1049    1 AACCCGAGCCGCG
## 3       UW-201557 tag_id_1049    1 AACCCGAGCCGCG
## 4 T-16-LS-0028-BS tag_id_1049    1 AACCCGAGCCGCG
## 5       UW-201554  tag_id_221    1    CCCGGCACGG
## 6     WTS22NAr148 tag_id_1049    1 AACCCGAGCCGCG
```

Toss out indivs with that were also removed from rubias

```
tossers <- read.csv("rockfish-species-id/removed_samples_rubias_01_16_25.csv")

baseline_for_combo <- baseline_for_combo %>%
  subset(!id %in% tossers$id)
```

In the meantime, let's move forward with the analysis.

```
# first make integers of the alleles
alle_idxs <- baseline_for_combo %>%
  dplyr::select(id, locus, rank, haplo) %>%
  group_by(locus) %>%
  mutate(alleidx = as.integer(factor(haplo, levels = unique(haplo)))) %>%
  ungroup() %>%
  arrange(id, locus, alleidx) # rubias can handle NA's, so no need to change them to 0's

# select just the columns to retain
#alle_idx2 <- alle_idxs[,-7]

# and spread the alleles
two_col <- alle_idxs %>%
  #group_by(indiv, locus) %>%
  unite(loc, locus, rank, sep = ".") %>%
  #ungroup() %>%
  select(-haplo) %>%
  pivot_wider(names_from = loc, values_from = alleidx)
```

Add the species info back on

```
spp_indiv <- read.csv('species_ID.csv')
colnames(spp_indiv) <- c('id','species', 'state', 'voucher')
```

PCA

```
# create vectors of indivs and species
spp_labels <- spp_indiv$species
indivs <- spp_indiv$id
```

```
# make factor?
spp_indiv$species <- factor(spp_indiv$species)
```

Make the df match the requirements for tidy_genomic_data

```
long_df <- alle_idxs %>%
  select(-haplo, -rank) %>%
  left_join(., spp_indiv) %>%
  select(species, everything()) %>%
  rename(INDIVIDUALS = id, STRATA = species, MARKERS = locus, GT = alleidx)
```

Genotypes should be coded with 3 integers for each alleles. 6 integers in total for the genotypes. e.g. 001002 or 111333 (for heterozygote individual). 6 integers WITH separator: e.g. 001/002 or 111/333 (for heterozygote individual). The separator can be any of these: “/”, “:”, “_“,”-“,”.”, and will be removed.

```
library("DescTools")

# create 3 digit integers from the genotypes
long_df$GT3 <- Format(long_df$GT, ldigits = 3, digits = 0)

head(long_df)
```

```
## # A tibble: 6 x 7
##   STRATA INDIVIDUALS MARKERS                              GT state voucher GT3
##   <fct>  <chr>       <chr>                             <int> <chr> <chr>   <For>
## 1 N/A    ORCH187     Plate_1_A01_Sat_GW603857_consens~     1 OR    N       001
## 2 N/A    ORCH187     Plate_1_A01_Sat_GW603857_consens~     6 OR    N       006
## 3 N/A    ORCH187     Plate_1_A01_Sat_GW603857_consens~    10 OR    N       010
## 4 N/A    ORCH187     Plate_1_A11_Sat_GE820299_consens~     1 OR    N       001
## 5 N/A    ORCH187     Plate_2_A09_Sat_EW986980_consens~     1 OR    N       001
## 6 N/A    ORCH187     Plate_2_A09_Sat_EW986980_consens~     7 OR    N       007
```

```
# NAs hold
# long_df %>%
#   filter(is.na(GT3))

# fix NAs
long_df0s <- long_df %>%
  mutate(GT3 = ifelse(is.na(GT3), "000", GT3))
```

Now combine the GT3 column per indiv/marker:

```
# make the genos characters and then try pasting them as strings
long_df0s$GT3 <- as.character(long_df0s$GT3)

long_df3digit <- long_df0s %>%
  group_by(INDIVIDUALS, MARKERS) %>%
  arrange(GT3, .by_group = TRUE) %>%
  summarise(GENOTYPE = toString(GT3))

# paste strings together
long_df3digit$GENOTYPE <- gsub(", ","",long_df3digit$GENOTYPE)


# add back on species identity as strata
df_for_conversion <- long_df0s %>%
  select(-GT, -GT3) %>%
  left_join(., long_df3digit) %>%
  unique() %>%
  rename(GT = GENOTYPE) %>%
  mutate(GT = ifelse(GT == "000000", NA, GT))

df_for_conversion$STRATA <- as.factor(df_for_conversion$STRATA)

# check on NAs here
head(df_for_conversion)
```

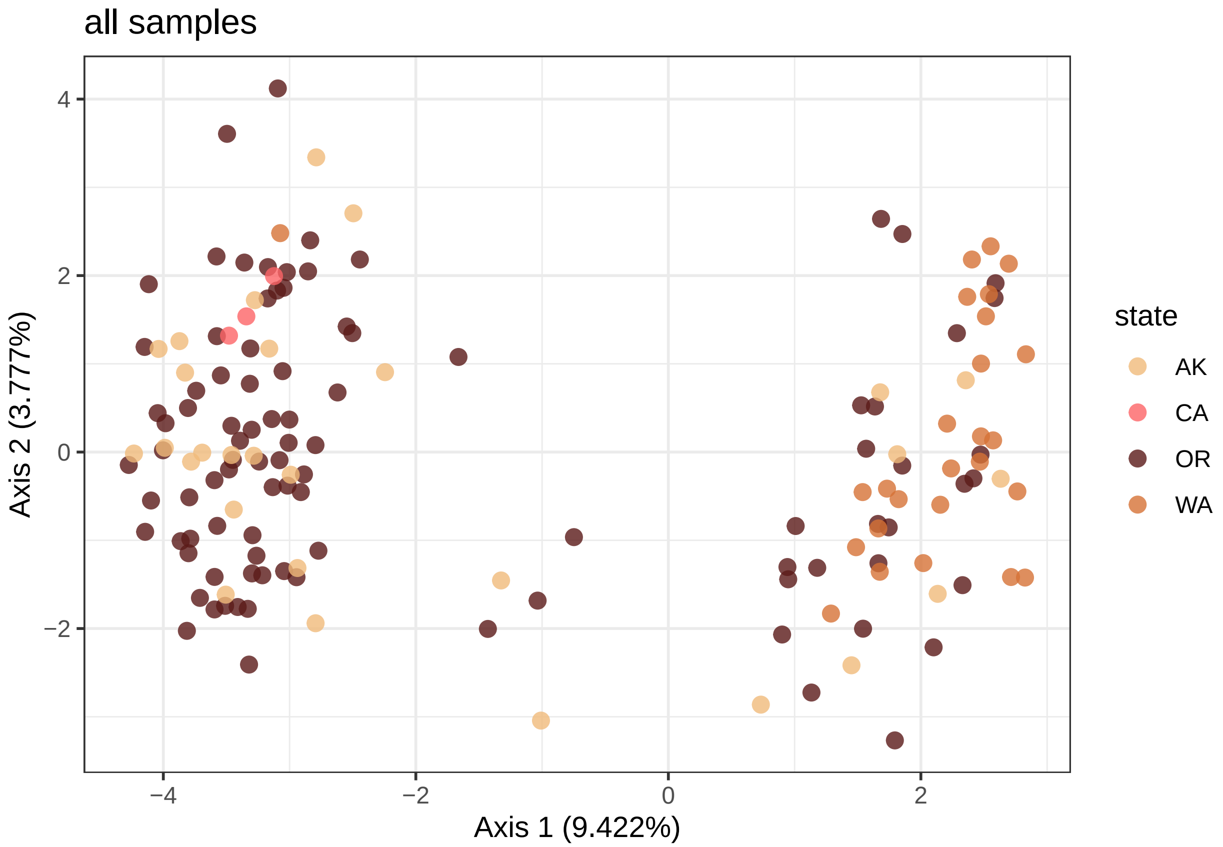
```
## # A tibble: 6 x 6
##   STRATA INDIVIDUALS MARKERS                            state voucher GT
##   <fct>  <chr>       <chr>                              <chr> <chr>   <chr>
## 1 N/A    ORCH187     Plate_1_A01_Sat_GW603857_consensus OR    N       001006010
## 2 N/A    ORCH187     Plate_1_A11_Sat_GE820299_consensus OR    N       001
## 3 N/A    ORCH187     Plate_2_A09_Sat_EW986980_consensus OR    N       001007
## 4 N/A    ORCH187     Plate_2_C08_Sat_EW987116_consensus OR    N       001
## 5 N/A    ORCH187     Plate_3_C03_Sat_GE798118_consensus OR    N       001002040
## 6 N/A    ORCH187     Plate_4_E10_Sat_EW976030_consensus OR    N       003069
```

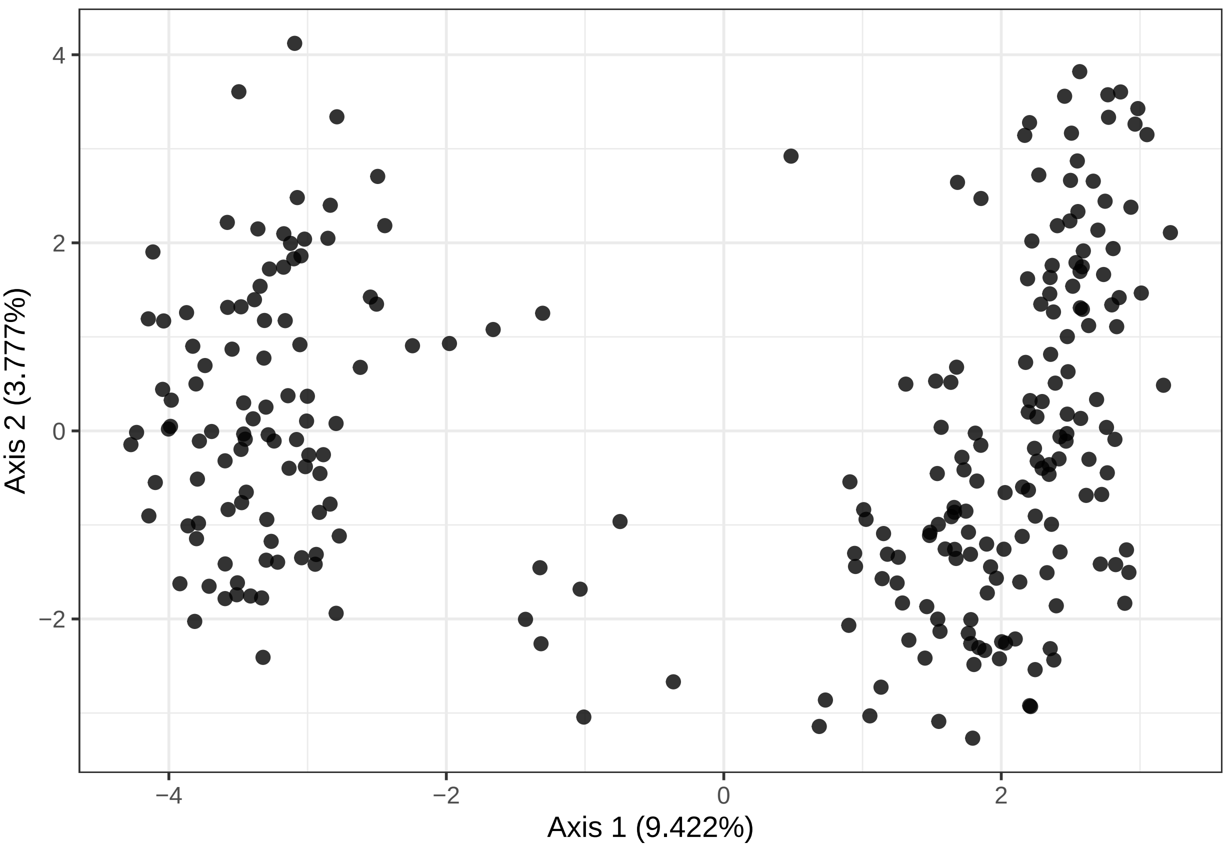
```
# use the radiator package for this conversion
genind_df <- write_genind(df_for_conversion)
```

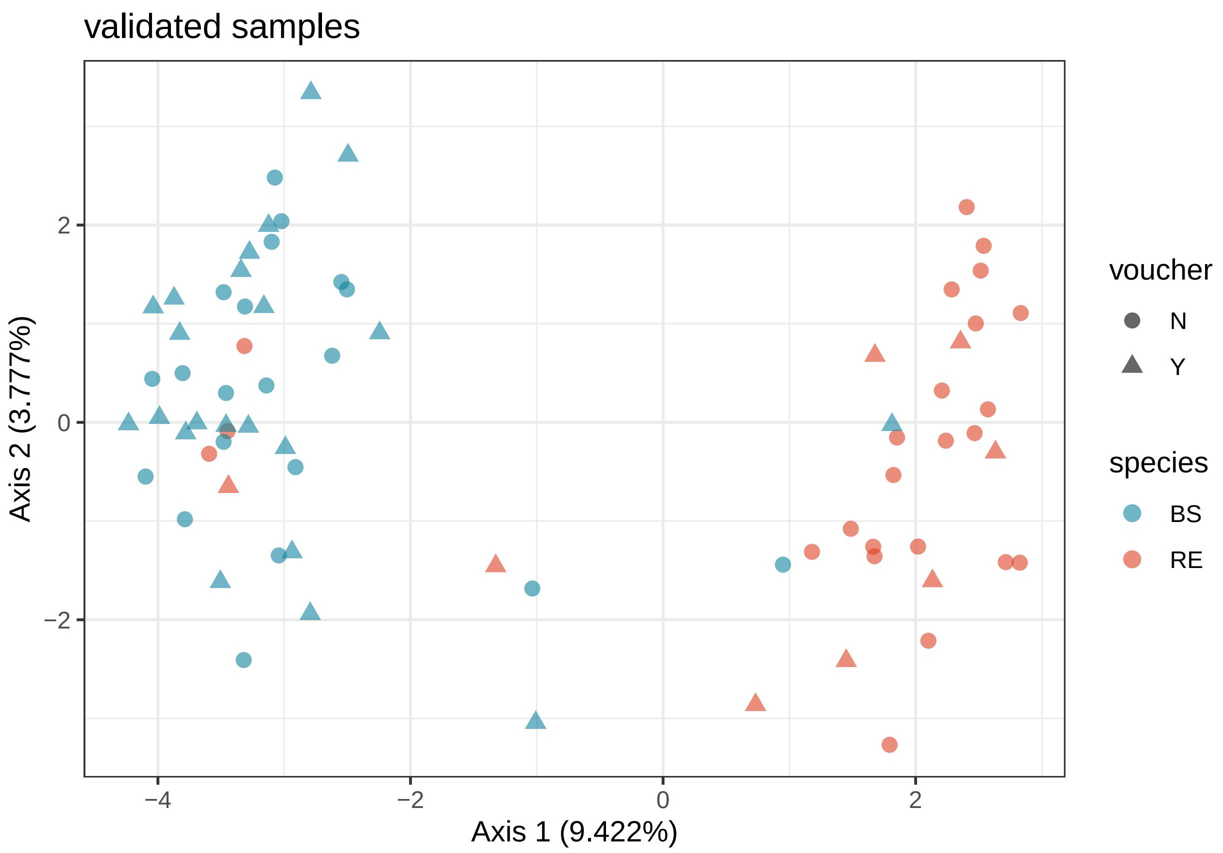
Now that the data is a genind object, go ahead and run the PCA.

Make PCA

```
## pdf
##   2
```




Axis 2 (3.777%)
4
2
0
−2
−4
−2
0
2
Axis 1 (9.422%)



```
## pdf
##   2
```

```
library(hierfstat)
pal_species <- c('melanostictus male' = "#A6CEE3",
                 'melanostictus female' = "#1F78B4",
                 'aleutianus male' = "#B2DF8A",
                 'aleutianus female' = "#33A02C",
                 'melanostictus' = "#1F78B4",
                 'aleutianus' = "#33A02C")

rubias_calls <-read.csv(file = '~/Desktop/RE_BS/stock review 2025 work/rockfish-species-id/rubias_outpu
  select(c(indiv, repunit))

colnames(rubias_calls) <- c('INDIVIDUALS','pop')
rubias_calls$INDIVIDUALS <- gsub('gtseq3_', '', rubias_calls$INDIVIDUALS)

df_for_conversion2 <- merge(df_for_conversion, rubias_calls) %>% select(-c('STRATA'))
names(df_for_conversion2)[names(df_for_conversion2) == 'pop'] <- 'STRATA'

genind_df <- write_genind(df_for_conversion2)


genet.dist(genind_df, diploid = TRUE, method = "WC84") %>% round(digits = 3)
```

```
##                 aleutianus
## melanostictus      0.048
```

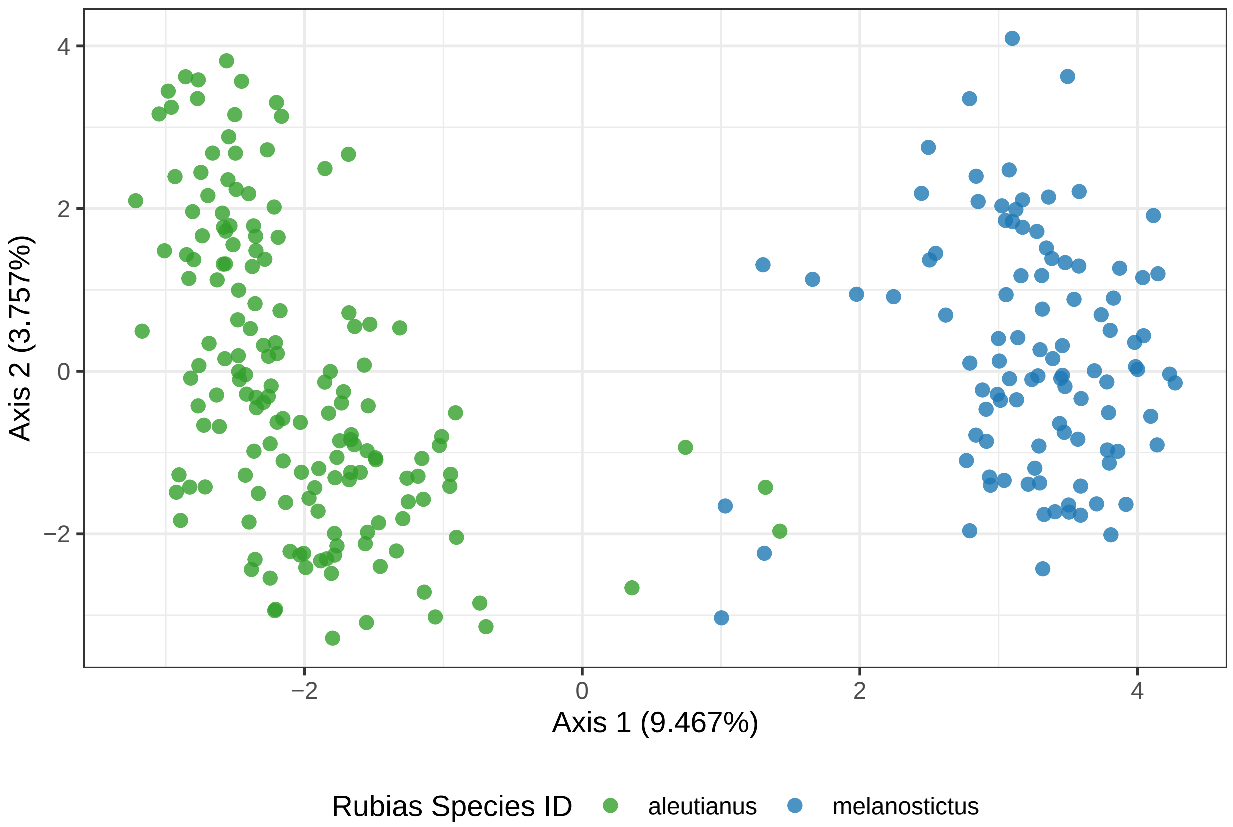
```r
# Allele presence absence data are extracted and NAs replaced using tab:
datasetX <- tab(genind_df, NA.method="mean") # double check that is this the appropriate method.

# make PCA
dataset_pca1 <- dudi.pca(datasetX, center = TRUE, scannf = FALSE, scale=FALSE, nf = 10)
PCA_df <- dataset_pca1$li

df <- tibble::rownames_to_column(PCA_df, "id")
PCA_df_w_labels <- merge(df, rubias_calls, by.x = 'id', by.y = 'INDIVIDUALS')
pca_info <- get_eigenvalue(dataset_pca1)


PCA <- ggplot(data = (PCA_df_w_labels), aes(x = Axis1, y = Axis2, color = pop)) +
  geom_point(size = 2, alpha = 0.8) +
  xlab(paste("Axis 1 (", round(pca_info$variance.percent[[1]], 3), "%)", sep = "")) +
  ylab(paste("Axis 2 (", round(pca_info$variance.percent[[2]], 3), "%)", sep = "")) +
  theme_bw() +
  scale_color_manual(values = pal_species, name = "Rubias Species ID") +
  theme(legend.position = 'bottom')


ggsave(file =  "~/Desktop/RE_BS/stock review 2025 work/figures/PCA_w_rubias_call.jpeg",
       width = 130,
       height = 90,
       units = c("mm"),
       dpi = 300)
PCA
```



```
dev.off()
```

```
## null device
##           1
```