adding_vouchers_to_diana_reference

Anita Wray

2025-07-03

```
knitr::opts_knit$set(root.dir = "~/Desktop/RE_BS/stock review 2025 work/")
```

```
source("R/rockfish-funcs2.R")
setwd("~/Desktop/RE_BS/stock review 2025 work/")
# get the names of the files
fdf <- read.table("rds-file-list.txt", stringsAsFactors = FALSE, header = TRUE) %>%
  tibble::as_tibble()

dir <- "microhaplot/"



# cycle over them, read them and add the gtseq_run column on each.
# at the end, bind them together.
genos_long <- lapply(1:nrow(fdf), function(i) {
  message("Working on ", fdf$file[i])
  call_genos_from_haplotRDS(path = file.path(dir, fdf$file[i])) %>%
    mutate(gtseq_run = fdf$gtseq_run[i]) %>%
    select(gtseq_run, everything())
}) %>%
  bind_rows()
```

```
## Working on DIANA--target_fastas--diana-fasta--snps4test.rds

## Joining with `by = join_by(id, locus, rank)`
```

```
#genos_long$id <- gsub('-','',genos_long$id)



#### In the end, let us get a data frame that includes genotypes for all the individuals  ####
# and which explicitly has NAs in places where data are missing, and also
# has the NMFS_DNA_ID on there
genos_long_explicit_NAs <- genos_long %>%
  select(gtseq_run, id) %>%
  unique() %>%
  unite(col = gid, sep = "_", gtseq_run, id) %>%
  select(gid) %>%
  unlist() %>%
  unname() %>%
```

```
  expand.grid(gid = ., locus = unique(genos_long$locus), gene_copy = 1:2, stringsAsFactors = FALSE) %>%
  tibble::as_tibble() %>%
  separate(gid, into = c("gtseq_run", "id"), convert = TRUE, sep = '_') %>%
  left_join(., genos_long) %>%
  arrange(gtseq_run, id, locus, gene_copy)
```

```
## Joining with `by = join_by(gtseq_run, id, locus, gene_copy)`
```

```
genos_long_explicit_NAs_vouchers <- genos_long_explicit_NAs %>%
  subset(id %in% c("LENTIGINOSUS-UW159878","HELVOMACULATUS-UW157086", "SIMULATOR-UW159881",
                   "SIMULATOR-UW159882", "HELVOMACULATUS-UW157087",'MACDONALDI-UW202823',
                   'VARIEGATUS-UW159883','MACDONALDI-UW202928','BREVISPINIS-UW157098',
                   'LENTIGINOSUS-UW159879', 'GILLI-UW202792',
                  'MACDONALDI-UW202916', 'GILLI-UW202913',
                   'LENTIGINOSUS-UW202941',  'MACDONALDI-UW202824',
                   'GILLI-UW202915','HELVOMACULATUS-UW114035',
                   'GILLI-UW202802', 'MACDONALDI-UW202914',
                   'LENTIGINOSUS-UW202931',  'EOS-UW114068',
                   'HELVOMACULATUS-UW119876', 'HELVOMACULATUS-UW119874',
                   'SIMULATOR-UW114049','BREVISPINIS-UW119935',
                   'MACDONALDI-UW114065', 'HELVOMACULATUS-UW151755',
                   'BREVISPINIS-UW114059','LENTIGINOSUS-UW152312',
                   'ROSENBLATTI-UW152188','ROSENBLATTI-UW152338',
                   'ROSENBLATTI-UW152343', 'LENTIGINOSUS-UW152333',
                   'BREVISPINIS-UW153444','BREVISPINIS-UW153443')) %>%
  subset(id != 'HELVOMACULATUS-UW151755') %>% #this one is mislabeled
  subset(id != 'BREVISPINIS-UW114059') #and this one is too

genos_long_explicit_NAs_vouchers$species <- str_extract(genos_long_explicit_NAs_vouchers$id, "[^-]+")
```

```
fdf <- read.table("rds-file-list.txt", stringsAsFactors = FALSE, header = TRUE) %>%
  tibble::as_tibble()
```

```
## Warning in read.table("rds-file-list.txt", stringsAsFactors = FALSE, header =
## TRUE): incomplete final line found by readTableHeader on 'rds-file-list.txt'
```

```
dir <- "~/Desktop/VermilionRF/microhaplotyping/VMSURF_microhaps/microhaplot/"



# cycle over them, read them and add the gtseq_run column on each.
# at the end, bind them together.
genos_long_sunset <- lapply(1:nrow(fdf), function(i) {
  message("Working on ", fdf$file[i])
  call_genos_from_haplotRDS(path = file.path(dir, fdf$file[i])) %>%
    mutate(gtseq_run = fdf$gtseq_run[i]) %>%
    select(gtseq_run, everything())
}) %>%
  bind_rows()
```

```
## Working on DIANA--target_fastas--diana-fasta--snps4test.rds
```

```
## Warning: `tbl_df()` was deprecated in dplyr 1.0.0.
## i Please use `tibble::as_tibble()` instead.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
## Warning: `tbl_df()` was deprecated in dplyr 1.0.0.
## i Please use `tibble::as_tibble()` instead.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
## Joining with `by = join_by(id, locus, rank)`
```

```
#Pull out two known sunset individual
croc_known <- c('H-14-MI-V0272', 'H-14-MI-V0191', 'H-14-MI-V0262', 'H-14-MI-V0250', 'H-14-MI-V0264', 'H

croc <- subset(genos_long_sunset, genos_long_sunset$id %in% croc_known)

croc_NAs <- croc %>%
  select(gtseq_run, id) %>%
  unique() %>%
  unite(col = gid, sep = "_", gtseq_run, id) %>%
  select(gid) %>%
  unlist() %>%
  unname() %>%
  expand.grid(gid = ., locus = unique(croc$locus), gene_copy = 1:2, stringsAsFactors = FALSE) %>%
  tibble::as_tibble() %>%
  separate(gid, into = c("gtseq_run", "id"), convert = TRUE, sep = '_') %>%
  left_join(., croc) %>%
  arrange(gtseq_run, id, locus, gene_copy)
```

```
## Joining with `by = join_by(gtseq_run, id, locus, gene_copy)`
```

```
croc_NAs$species <- 'crocotulus'


genos_long_explicit_NAs_vouchers <- rbind(genos_long_explicit_NAs_vouchers, croc_NAs)
```

```
# slow-ish function to get the total read depth column
tdepth <- function(a, d) {
  if(any(is.na(a))) {
    return(NA)
  }
  if(a[1]==a[2]) {
    return(d[1])
  } else {
    return(d[1] + d[2])
  }

}
# this takes the highest read-depth instance of each duplicately-genotyped individual.
geno_one_each <- genos_long_explicit_NAs_vouchers %>%
  group_by(id, species, locus, gtseq_run) %>%
```

```
  mutate(total_depth = tdepth(allele, depth)) %>%
  ungroup() %>%
  arrange(id, species, locus, total_depth, gtseq_run, depth) %>%
  group_by(id, species, locus) %>%
  mutate(rank = 1:n()) %>%
  #ungroup() %>%
  filter(rank <= 2)
```

```
# read in a list of the 6 loci
to_remove <- read_csv("data/loci_to_remove.csv", show_col_types = FALSE)

# only keep the loci that are not those 6
keepers <- geno_one_each %>%
  anti_join(., to_remove, by = "locus")

# that should leave 90 loci

length(unique(geno_one_each$locus)) #looks like this isn't neccessary but maybe good to keep just incas
```

```
## [1] 90
```

Toss out indivs with missing data at more than 25 loci

Now, toss out any individual with fewer than 65 non-missing loci

```
no_hi_missers <- keepers %>%
  group_by(id, gtseq_run) %>%
  filter(sum(!is.na(allele)) >= (65*2))

unique(keepers$id)[which(!unique(keepers$id) %in% unique(no_hi_missers$id))] #which ones are dropped?
```

```
## [1] "LENTIGINOSUS-UW159878" "MACDONALDI-UW202823"   "MACDONALDI-UW202824"
```

Load baseline data

```
# baseline data - curated, 997 indivs
baseline <- readRDS("new_baseline_data/processed/sebastes_spp_id_baseline_haplotypes.rds")

# remove the 6 loci that had HWE and other issues
to_remove <- read_csv("data/loci_to_remove.csv")
```

```
## Rows: 6 Columns: 1
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): locus
##
## 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.
```

```
baseline90 <- baseline %>%
  anti_join(., to_remove)
```

```
## Joining with `by = join_by(locus)`
```

```
# add reference column to prepare data for rubias
dataset <- no_hi_missers %>%
  mutate(sample_type = "reference") %>%
  rename(collection = species) %>%
  rename(indiv = id) %>%
  mutate(repunit = collection) %>%
  ungroup() %>%
  mutate(id = indiv) %>%
  mutate(species = collection) %>%
  select(colnames(baseline90)) # reorder the columns

dataset %>%
  group_by(indiv) %>%
  tally() %>%
  arrange(desc(n))
```

```
## # A tibble: 36 x 2
##    indiv                    n
##    <chr>                <int>
##  1 BREVISPINIS-UW119935   180
##  2 BREVISPINIS-UW153443   180
##  3 BREVISPINIS-UW153444   180
##  4 BREVISPINIS-UW157098   180
##  5 EOS-UW114068           180
##  6 GILLI-UW202792         180
##  7 GILLI-UW202802         180
##  8 GILLI-UW202913         180
##  9 GILLI-UW202915         180
## 10 H-14-MI-V0191          180
## # i 26 more rows
```

```
new_baseline <- rbind(dataset, baseline90)


tossers <- new_baseline %>%
  select(indiv, gtseq_run, id) %>%
  unique() %>%
  group_by(indiv) %>%
  tally() %>%
  filter(n >1)

baseline90_one_each <- new_baseline %>%
  anti_join(., tossers) %>%
  select(-c('rank', 'total_depth'))
```

```
## Joining with `by = join_by(indiv)`
```

```
# baseline data - curated, 1028 indivs
baseline_spp_info <- baseline90_one_each %>%
  select(sample_type, repunit, collection, indiv, gtseq_run, id, species) %>%
  unique()
baseline_spp_info$gtseq_run <- as.character(baseline_spp_info$gtseq_run)
```

```
# slim that down to just the matching field with the unknowns
for_alleidx <- baseline90_one_each %>%
  select(-indiv, -c(1:3), -species)

for_alleidx$gtseq_run <- as.character(for_alleidx$gtseq_run)
```

```
# merge the two dataframes
merged_df <- for_alleidx

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


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

add back on info for reference and make two-column format for rubias

```
# baseline
reference <- two_col %>%
  left_join(., baseline_spp_info) %>%
  filter(!is.na(species)) %>%
  select(-gtseq_run, -id, -species) %>%
  select(sample_type, repunit, collection, indiv, everything())
```

```
## Joining with `by = join_by(gtseq_run, id)`
```

```
# Now that the data are in the corret format, load Rubias
library(rubias)

# perform self-assignment of reference samples
ref_self <- self_assign(reference, gen_start_col = 5)
```

```
## Summary Statistics:
##
```

```
## 1028 Individuals in Sample
##
## 90 Loci: Plate_1_A01_Sat_GW603857_consensus.1, Plate_1_A11_Sat_GE820299_consensus.1, Plate_2_A09_Sat
##
## 63 Reporting Units: melanops, caurinus, hopkinsi, mystinus, atrovirens, chrysomelas, auriculatus, en
##
## 64 Collections: melanops, caurinus, hopkinsi, mystinus, atrovirens, chrysomelas, carnatus, auriculatu
##
## 8.56% of allelic data identified as missing
```

```
# and take a quick look at those assignments
good <- ref_self %>%
  filter(inferred_repunit == repunit) %>%
  filter(scaled_likelihood > 0.95)


# look at the added vouchers
additional_vouchers <- good %>%
  subset(grepl('UW', indiv) | grepl('H-14', indiv))

table(additional_vouchers$repunit)
```

```
##
##    BREVISPINIS     crocotulus          GILLI HELVOMACULATUS   LENTIGINOSUS
##              4              6              4              1              2
##     MACDONALDI
##              4
```

```
mistakes <- ref_self %>%
  filter(inferred_repunit != repunit) %>%
  filter(scaled_likelihood > 0.80) %>%
  select(indiv, collection, inferred_collection, scaled_likelihood, z_score)


mistakes
```

```
## # A tibble: 12 x 5
##    indiv                collection inferred_collection scaled_likelihood z_score
##    <chr>                <chr>      <chr>                           <dbl>   <dbl>
##  1 R016832              rosaceus   HELVOMACULATUS                  1.00    0.295
##  2 EOS-UW114068         EOS        ruberrimus                      1       0.892
##  3 HELVOMACULATUS-UW11~ HELVOMACU~ rosaceus                        1.00   -0.472
##  4 HELVOMACULATUS-UW11~ HELVOMACU~ rosaceus                        0.999  -2.04
##  5 HELVOMACULATUS-UW11~ HELVOMACU~ rosaceus                        1.00   -0.421
##  6 ROSENBLATTI-UW152188 ROSENBLAT~ chlorostictus                   1       1.16
##  7 ROSENBLATTI-UW152338 ROSENBLAT~ chlorostictus                   1.00   -0.166
##  8 ROSENBLATTI-UW152343 ROSENBLAT~ chlorostictus                   1.00    0.185
##  9 SIMULATOR-UW114049   SIMULATOR  HELVOMACULATUS                  1.00    1.85
## 10 SIMULATOR-UW159881   SIMULATOR  HELVOMACULATUS                  0.980  -0.455
## 11 SIMULATOR-UW159882   SIMULATOR  HELVOMACULATUS                  0.987   1.03
## 12 VARIEGATUS-UW159883  VARIEGATUS zacentrus                       0.999  -5.28
```

Looks like the greenspot/greenblotched and pink/pinkrose/rosethorn aren't distinguishable with this panel

other assignments

```
# between 50-95% likelihood
ref_self %>%
  filter(inferred_repunit != repunit) %>%
  filter(scaled_likelihood > 0.5 & scaled_likelihood < 0.95) %>%
  arrange(desc(scaled_likelihood))
```

```
## # A tibble: 0 x 11
## # i 11 variables: indiv <chr>, collection <chr>, repunit <chr>,
## #   inferred_collection <chr>, inferred_repunit <chr>, scaled_likelihood <dbl>,
## #   log_likelihood <dbl>, z_score <dbl>, n_non_miss_loci <int>,
## #   n_miss_loci <int>, missing_loci <list>
```

```
saveRDS(new_baseline %>% subset(!indiv %in% mistakes$indiv), "new_baseline_data/processed/sebastes_spp_
```