Fitting Random Forests for WhoseEgg Shiny App

Katherine Goode

Last Updated: July 08, 2021

This document contains code that fits the three random forest models that will be used in the app: models with invasive carp species grouped into one class and all other species classified into species, genus, and family. The data used to train the model is that used in Goode et al. (2021) to train the augmented models, and the same seed (808) is used, so the models should agree.

Load packages:

```
library(dplyr)
library(randomForest)
library(purrr)
```

Make a list of the response variables:

```
vars_resp = c(
  "Family_IC",
  "Genus_IC",
  "Common_Name_IC"
)
```

Make a vector of the predictor variables:

```
vars_pred = c(
  "Month",
  "Julian_Day",
  "Temperature",
  "Conductivity",
  "Larval_Length",
  "Membrane_Ave",
  "Membrane_SD",
  "Membrane_CV",
  "Embryo_to_Membrane_Ratio",
  "Embryo_Ave",
  "Embryo_SD",
  "Embryo_CV",
  "Egg_Stage",
  "Compact_Diffuse",
  "Pigment",
  "Sticky_Debris",
  "Deflated"
)
```

Load the prepared egg data and convert necessary variables to factors:

```
eggdata_for_app <-
  read.csv("../data/eggdata_for_app.csv") %>%
  mutate_at(
    .vars = c(
      "Egg_Stage",
      "Compact_Diffuse",
      "Pigment",
      "Sticky_Debris",
      "Deflated",
      all_of(vars_resp)
    ),
    .funs = factor
  )
str(eggdata_for_app)
```

```
## 'data.frame':    1972 obs. of  25 variables:
##  $ Site                     : chr  "UPI" "DNI" "MTH" "MTH" ...
##  $ River                    : chr  "UMR" "UMR" "IAR" "IAR" ...
##  $ Year                     : int  2014 2014 2014 2014 2014 2015 2015 2015 2015 2015 ...
##  $ Month                    : int  7 8 6 6 8 8 6 6 5 5 ...
##  $ Julian_Day               : int  209 227 172 172 236 222 161 161 151 151 ...
##  $ Temperature              : num  24.7 25.3 26.3 26.3 25.5 26.3 23.1 23.1 19.2 19.2 ...
##  $ Conductivity             : num  522 440 473 473 498 514 654 654 674 674 ...
##  $ Larval_Length            : num  0 0 0 0 0 ...
##  $ Membrane_Ave             : num  1.43 1.24 3.77 2.84 1.42 ...
##  $ Membrane_SD              : num  0.0436 0.0291 0.1044 0.0685 0.0263 ...
##  $ Membrane_CV              : num  0.0305 0.0234 0.0277 0.0241 0.0185 ...
##  $ Embryo_to_Membrane_Ratio: num  1 0.821 0.336 0.568 0.78 ...
##  $ Embryo_Ave               : num  1.43 1.02 1.27 1.61 1.11 ...
##  $ Embryo_SD                : num  0.0436 0.092 0.0187 0.2585 0.1403 ...
##  $ Embryo_CV                : num  0.0305 0.0901 0.0148 0.1602 0.1268 ...
##  $ Egg_Stage                : Factor w/ 10 levels "1","2","3","4",..: 10 5 4 6 6 8 8 4 10 10 ...
##  $ Compact_Diffuse          : Factor w/ 2 levels "C","D": 2 1 1 1 1 1 1 1 2 2 ...
##  $ Pigment                  : Factor w/ 2 levels "N","Y": 2 2 1 1 2 1 1 1 1 1 ...
##  $ Sticky_Debris            : Factor w/ 2 levels "N","Y": 1 1 1 1 1 2 1 1 2 2 ...
##  $ Deflated                 : Factor w/ 2 levels "N","Y": 1 1 2 2 1 2 2 2 2 2 ...
##  $ Genus                    : chr  "Aplodinotus" "Aplodinotus" "Ctenopharyngodon" "Ctenopharyngodon" ...
##  $ Common_Name              : chr  "Freshwater Drum" "Freshwater Drum" "Grass Carp" "Grass Carp" ...
##  $ Family_IC                : Factor w/ 8 levels "Catostomidae",..: 8 8 5 5 8 3 5 5 5 5 ...
##  $ Genus_IC                 : Factor w/ 16 levels "Alosa","Aplodinotus",..: 2 2 9 9 2 13 9 9 9 9 ...
##  $ Common_Name_IC           : Factor w/ 27 levels "Banded Darter",..: 11 11 14 14 11 17 14 14 14 14 ...
```

Function for fitting a random forest model given a response variable, predictor variables, and a dataset (uses the same seed to fit the random forests as Camacho et al. (2019) and Goode et al. (2021)):

```
fit_rf <- function(resp, preds, data) {

  # Fit the random forest
  set.seed(808)
  rf <- randomForest(
    data %>% pull(resp) ~ .,
    data = data %>% select(all_of(preds)),
    importance = T,
```

```
    ntree = 1000
  )

  # Put model in a named list
  rf_list = list(rf)
  names(rf_list) = resp

  # Return the named list
  return(rf_list)

}
```

Fit the random forest models:

```
rfs_for_app <-
  map(
    .x = vars_resp,
    .f = fit_rf,
    preds = vars_pred,
    data = eggdata_for_app
  ) %>%
  flatten()
```

Save the random forests:

```
saveRDS(
  object = rfs_for_app,
  file = "../data/rfs_for_app.rds"
)
```

Session Info

```
sessionInfo()
```

```
## R version 4.0.4 (2021-02-15)
## Platform: x86_64-apple-darwin17.0 (64-bit)
## Running under: macOS Big Sur 10.16
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/4.0/Resources/lib/libRblas.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/4.0/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] purrr_0.3.4         randomForest_4.6-14 dplyr_1.0.6
```

```
##
## loaded via a namespace (and not attached):
##  [1] knitr_1.33        magrittr_2.0.1    tidyselect_1.1.1  R6_2.5.0
##  [5] rlang_0.4.11      fansi_0.5.0       stringr_1.4.0     tools_4.0.4
##  [9] xfun_0.23         utf8_1.2.1        DBI_1.1.1         htmltools_0.5.1.1
## [13] ellipsis_0.3.2    assertthat_0.2.1  yaml_2.2.1        digest_0.6.27
## [17] tibble_3.1.2      lifecycle_1.0.0   crayon_1.4.1      vctrs_0.3.8
## [21] glue_1.4.2        evaluate_0.14     rmarkdown_2.9     stringi_1.6.2
## [25] compiler_4.0.4    pillar_1.6.1      generics_0.1.0    pkgconfig_2.0.3
```