Stats Lab Bird

Joel Alder, Samir Hauser, Connor Charlton

2023-04-25

Data pre processing

We first load in the data and pre process them.

We transform our response variable to 0 and 1.

```
d.bird$sex_genetics[d.bird$sex_genetics == "M"] <- "m"
d.bird$sex_genetics[d.bird$sex_genetics == "F"] <- "f"
d.bird$sex_genetics[d.bird$sex_genetics == "m"] <- 0
d.bird$sex_genetics[d.bird$sex_genetics == "f"] <- 1
```

Next we make a new variable season, which indicates the time of the catch. We do this because some of the variables depend on the time of the year. I. e. the weight is different in summer and winter.

```
d.bird$season <- ifelse(d.bird$month >= 3 & d.bird$month <= 5, "spring",
                        ifelse(d.bird$month >= 6 & d.bird$month <= 8, "summer",
                               ifelse(d.bird$month >= 9 & d.bird$month <= 11,
                                      "autumn", "winter")))
d.bird$season <- factor(d.bird$season, levels = c("winter", "spring", "summer", "autumn"),
                        labels = c("0", "1", "2", "3"))
```

We have two variables for the feather length, P1 and P8. We are filling out P8 if there is a NA and we have a measure in the P1 column. In total there are 32 cases of it.

```
if (sum(!is.na(d.bird$P1)) > 0) {
  d.bird$P8[is.na(d.bird$P8)] <- d.bird$P1[is.na(d.bird$P8)]
} else {
  print("NO impuatation possible as P1 is NA")
}
```

There are some duplicates in the dataset which we remove. Then there is a possibility that one bird is captured more than once. If this is the case we only use the newest observation. Afterwards we select only the adult birds and keep the one for which we have the label. In the end we select the columns we want to use for training. There are all the different morphological traits.

```
# Remove duplicates
d.bird <- d.bird[!duplicated(d.bird, fromLast = TRUE), ]

# check multiple caputres
freq_table <- table(d.bird$ringnr)
freq_table <- freq_table[freq_table > 1]
freq_table <- sort(freq_table, decreasing = T)
double_caputre = as.data.frame(freq_table)
d.bird <- d.bird[!duplicated(d.bird$ringnr, fromLast = TRUE), ]

# select only the adult birds
df_adult <- d.bird[d.bird$Age>1,]

# select only rows with no NA in the response
df_adult <- df_adult[complete.cases(df_adult["sex_genetics"]),]

# select only the needed columns
cols <- c("season", "Age", "Wing", "P8", "Tarsus", "weight", "Fat", "Muscle",
          "Bill_length", "sex_genetics")
df_adult_sub = data.frame(df_adult[cols])

# chaning type of variables
df_adult_sub$Fat <- as.factor(df_adult_sub$Fat)
df_adult_sub$Muscle <- as.factor(df_adult_sub$Muscle)
df_adult_sub$sex_genetics <- as.factor(df_adult_sub$sex_genetics)
str(df_adult_sub)
```

```
## ’data.frame':    981 obs. of  10 variables:
##  $ season      : Factor w/ 4 levels "0","1","2","3": 3 3 2 3 2 2 2 2 3 3 ...
##  $ Age         : num  5 4 4 5 4 4 4 6 5 5 ...
##  $ Wing        : num  123 116 123 123 116 ...
##  $ P8          : num  98 90 97 94 90 96.5 89 89.5 89.5 95 ...
##  $ Tarsus      : num  23.4 22.7 22.9 23.4 22.5 24.2 22.6 23.2 22.8 22.7 ...
##  $ weight      : num  38 40 37.5 34 32.5 35 33 34 38 35.5 ...
##  $ Fat         : Factor w/ 7 levels "0","1","2","3",..: 3 NA 2 2 2 2 2 2 3 2 ...
##  $ Muscle      : Factor w/ 3 levels "1","2","3": 2 NA 2 2 3 1 2 3 1 1 ...
##  $ Bill_length : num  NA NA 10.8 14.1 NA NA NA NA NA NA ...
##  $ sex_genetics: Factor w/ 2 levels "0","1": 1 2 1 1 2 1 2 2 2 1 ...
```

We now train three different models for this data. A decision tree, random forest and lastly a logit model with imputed data via the mice library. We use the accuracy as our evaluation metric.

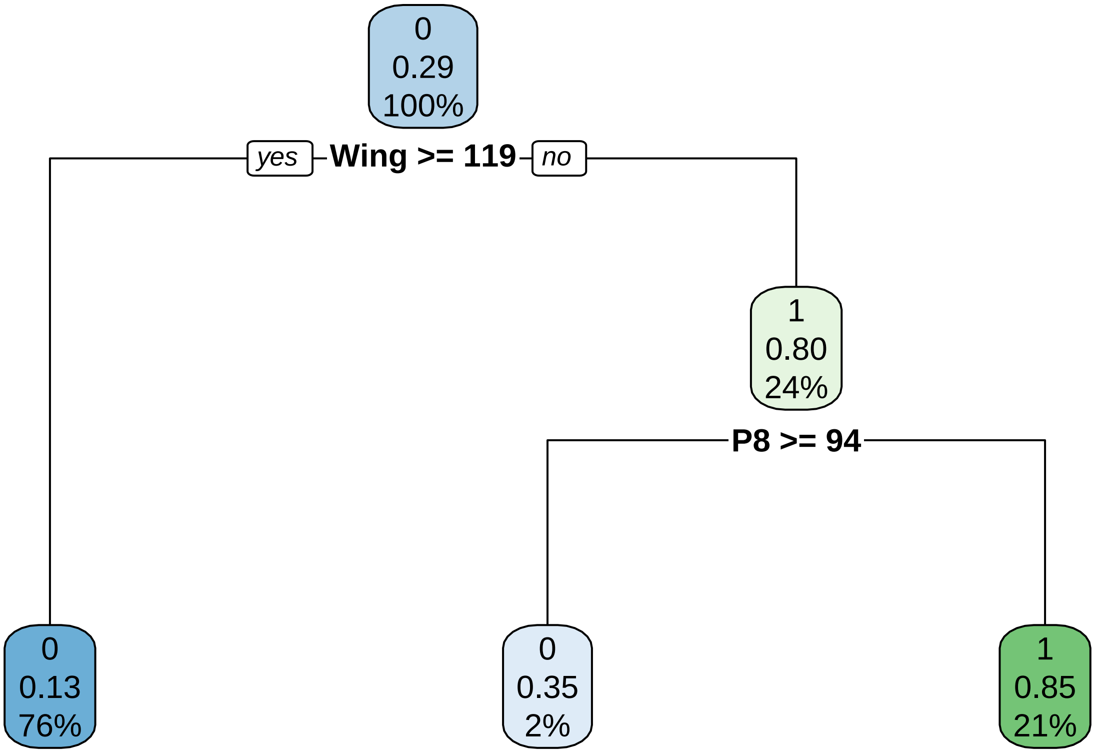
Decision Tree

We use a ten fold cross-validation to get the accuracy score of the decision tree. The advantage of this method is the it can handle the missing values.

```
set.seed(123)
n <- dim(df_adult_sub)[1]
K <- 10
folds <- sample(cut(seq(1,n),breaks=K,labels=FALSE), replace = FALSE)
Fold.error <- numeric(K)
for (i in 1:K) {
  test.ind <- which(folds == i)
  tree_model.i <- rpart(sex_genetics ~ ., data = df_adult_sub[-test.ind,], method = "class")
  test_predictions.i <- predict(tree_model.i, df_adult_sub[test.ind,], type = "class")
  Fold.error[i] <- mean(test_predictions.i == df_adult_sub[test.ind,]$sex_genetics)
}
cat("Accuracy Decision Tree:", round(mean(Fold.error), digits = 5))
```

```
## Accuracy Decision Tree: 0.84714
```

We get a accuracy around **84.7 %**. One question we have to ask is, if we should weight the two error differently. Maybe the miss classification of a female bird as a male should have a higher price than the opposite. We should keep in mind that we have a unbalanced set with two thirds male birds. Now we plot the decision tree, to see which variables are important.



We get a rather easy tree. The 0 stands for male birds and the 1 is a female bird. As we have seen in the descriptive analysis of our data, the wing variable is the most important one. We just have two variables in our tree.

Random Forest

We use a ten fold cross-validation to get the accuracy score of the random forest. The random forest can also handle missing values by specifying the na.action parameter. It will start by using median/mode for missing values, but then it grows a forest and computes proximities, then iterate and construct a forest using these newly filled values etc.

```
Fold.error <- numeric(K)
for (i in 1:K) {
  test.ind <- which(folds == i)
  rf.i <- randomForest(sex_genetics ~ ., data = df_adult_sub[-test.ind,], ntree = 200,
                       importance = TRUE, na.action = na.roughfix)
  test_data <- na.omit(df_adult_sub[test.ind,])
  test_predictions.i <- predict(rf.i, newdata = test_data, na.action = na.omit)
  Fold.error[i] <- mean(test_predictions.i == test_data$sex_genetics)
}
cat("Accuracy Random Forest:", round(mean(Fold.error), digits = 5))
```

```
## Accuracy Random Forest: 0.84051
```

We get a accuracy around **84 %**. This can be due that we have to omit the test data with NAs and therefore we have less test data compared to the decision tree.

Logit model

We use a ten fold cross-validation to get the accuracy score of the logit model. Because this model type can not handle missing values, we will impute this with the mice function. This function gives in our case five different imputed data set back. We then fit for every of this data set the logit model and then pool the estimates. With the pooled estimates we predict the sex on our test data set.

```
imp <- mice(df_adult_sub, seed = 123, print = F, m = 35)
imp <- mice::complete(imp, "all")
Fold.error <- numeric(K)
for (i in 1:K) {
  test.ind <- which(folds == i)

  train_data <- df_adult_sub[-test.ind,]
  test_data <- df_adult_sub[test.ind,]

  # mice
  imp_train = mice(train_data, seed = 123, print = F, m=35)

  # fit model
  fit_1 <- with(imp_train, glm(sex_genetics ~ season + Age
                               + Wing + P8 + Tarsus + weight +
                                 Muscle + Bill_length,
                               family = binomial))
  pooled <- pool(fit_1)

  # hack for predict
  pooled_lm = fit_1$analyses[[1]]
  pooled_lm$coefficients = summary(pooled)$estimate

  size = nrow(imp_train[[1]][test.ind,-(10)]) # remove sex

  # loop over imputed data
  dat = matrix(nrow = size, ncol = 5)
  for (k in 1:5)
  {
    predicted_values = predict(pooled_lm, newdata = imp[[k]][test.ind,-(10)],
                               type="response")
    binary_predictions <- ifelse(predicted_values > 0.5, 1, 0)
    dat[,k] = binary_predictions
  }

  #majority vote
  test_predictions.i <- apply(dat,1,Mode)

  Fold.error[i] <- mean(test_predictions.i == df_adult_sub[test.ind,]$sex_genetics)

}
cat("Accuracy Logit Model:", round(mean(Fold.error), digits = 5))
```

```
## Accuracy Logit Model: 0.87159
```

We get a slightly better accuracy of **86.5 %** for this model compared to the decision tree. Now we want to have a look at the pooled estimates of the logit model to see the significance of them.

```
summary(pooled)
```

```
##           term    estimate  std.error   statistic        df      p.value
## 1  (Intercept) 84.07714691 6.95956874 12.08079840 588.82991 3.568219e-30
## 2      season1  0.01415205 0.28083389  0.05039295 476.92571 9.598304e-01
## 3      season2 -1.66168310 0.86844791 -1.91339408  86.99317 5.898652e-02
## 4          Age -0.05263109 0.15333065 -0.34325224 479.83271 7.315590e-01
## 5         Wing -0.56426568 0.06256232 -9.01925706 756.61902 1.534345e-18
## 6           P8 -0.33150603 0.06785243 -4.88569125 132.45756 2.924452e-06
## 7       Tarsus  0.28942489 0.16708921  1.73215784  62.79215 8.815538e-02
## 8       weight -0.02307209 0.03277572 -0.70393858 438.54852 4.818446e-01
## 9      Muscle2  1.10075351 0.57367823  1.91876466 350.19382 5.582676e-02
## 10     Muscle3  2.36607051 0.71531853  3.30771595 408.58395 1.023876e-03
## 11 Bill_length  0.68287019 0.28209475  2.42071216  24.82681 2.314086e-02
```

We see a similar picture to the decision tree. The two variables wing and P8 are highly significant. But also the variable bill_length and muscle are on the 5 % level significant.

Conclusion

As a small conclusion we were able to predict the sex of the birds with around 85 % accuracy. The two most important variables to measure in the future are the Wing length and the feather length (P8). The imputing of the data via mice and then using the logit model yield to a slightly better model. For the next steps, we want to present our result to our client and discuss it with her. We also have some question left, i. e. why are there some duplicates in the data? Is it good to combine the variables P1 and P8 or are the measure different? Should we weight one miss classification more than the other one?