Bye Bye Birdie: The Determination of Variables Which Influence Flight Initiation Distance in the Dark-Eyed Junco (*Junco hymenalis*)

Omer Lavian

February 29, 2020

Contents

1 Introduction **3**

2 Methods **3**

3 Results **5**

1 Introduction

Anthropogenic change has undoubtedly had an effect on animal behavior [4]. One element of behavior that may be affected by human presence is flight initiation distance (FID). FID is the distance an animal initially moves away from an apparent threat. In birds, previous research has suggested that flight initiation distance is significantly associated with the animal's initial distance from an approaching threat [3]. Data on initial distance, as well as a variety of other factors, have been taken by the Yeh Lab at UCLA along with FID measurements in a sparrow known as the dark-eyed junco (*Junco hymenalis*). While these data have been analyzed, to my knowledge, they have not been combined in such a way as to allow one to predict the flight initiation distance of a junco given a certain set of factors. My goal, therefore, is to utilize the Yeh Lab's FID data to perform a multiple linear regression which will make possible predictions of FID for this species [2]. The overall goal of this project is to gain a better understanding of the ways in which different aspects of urbanization interact to affect flight initiation distance in dark-eyed juncos.

2 Methods

This section is incomplete, but overall the project involves quantifying the relationships between different variables and FID and then combining that into a multiple linear regression. To quantify the relationships, I started by making a function that prints the Pearson's correlation coefficient for interval variables that are input into the function and FID. I have also made a function that performs dummy coding on categorical variables with two categories [1].

Listing 1: Function comparing FID to other variables.

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import csv

def multiple_corrs_excel(filename, colx, coly=list):
    '''
    Computes correlation coefficients of a list of
        columns in an Excel file to one particular column.
         Also creates a plot for each correlation.

    ___________
    Example:
    >>>multiple_corrs_excel("Spam_Data.xlsx", "Spam
        Consumption Frequency", ["Spam Quality", "Love for
         Eggs", "Frequency of Farts in Your General
        Direction"])
```

```
    The correlation coefficient of Spam Conspumption
        Frequency and Spam Quality is 0.75.

    PLOT APPEARS HERE

    The correlation coefficient of Spam Consumption
        Frequency and Love for Eggs is 0.84.

    PLOT APPEARS HERE

    The correlation coefficient of Spam Consumption
        Frequency and Frequency of Farts in Your General
        Direction is 1.00.

    PLOT APPEARS HERE

    '''



    DataFrame = pd.read_excel(filename)

    for item in coly:
        y = float(DataFrame[colx].corr(DataFrame[item]))
        print("The correlation coefficient of {} and {} 
            is {number:.{digits}f}. \n".format(colx, item,
             number = y, digits=2))
        corrplot = plt.plot(DataFrame[colx], DataFrame[
            item], "bo")
        plt.xlabel(colx)
        plt.ylabel(item)
        plt.title(colx + " vs. " + item)
        plt.show()
```

Listing 2: Function that performs dummy coding on categorical variable with two categories

```
import pandas as pd
import csv
import re


def dummy_coder_excel():
    #(inputfile, variable, cat1, cat2, outputfile)
    '''
    Converts categorical variables with two categories
        into dichotomous variables and stores the output
        in a separate text file. This allows categorical
```

```
    variables to be used in a multiple linear
    regression.
"category1" is stored as a 0, while "category2" is
    stored as a 1.

This is a draft function. It will be modified to be
    more generalizable.

'''

#use regex to make function robust against typos; if
    wrote hop or fly in past of present tense or in
    upper or lowecase, then fxn should still work
#no typos appear in this case, but future variability
     in data entry will have less of an effect on the
    function due to this use of regex
Dataframe = pd.read_excel("FID_Data.xlsx")
dummied_list = []
#iterate through items in category and assign values
    of 0 or 1 to each entry
for item in Dataframe["Flew.hop"]:
    matchtest1 = re.match("[Ff]*", str(item))
    matchtest2 = re.match("[Hh]*", str(item))
    if bool(matchtest1) == True:
        dummied_list.append(0)
    if bool(matchtest2) == True:
        dummied_list.append(1)
    else:
        continue
#move codes to separate file for future use
with open("FlewHop_Dummied.txt", "w+") as output:
    for item in dummied_list:
        output.write(str(item) + "\n")
```

3 Results

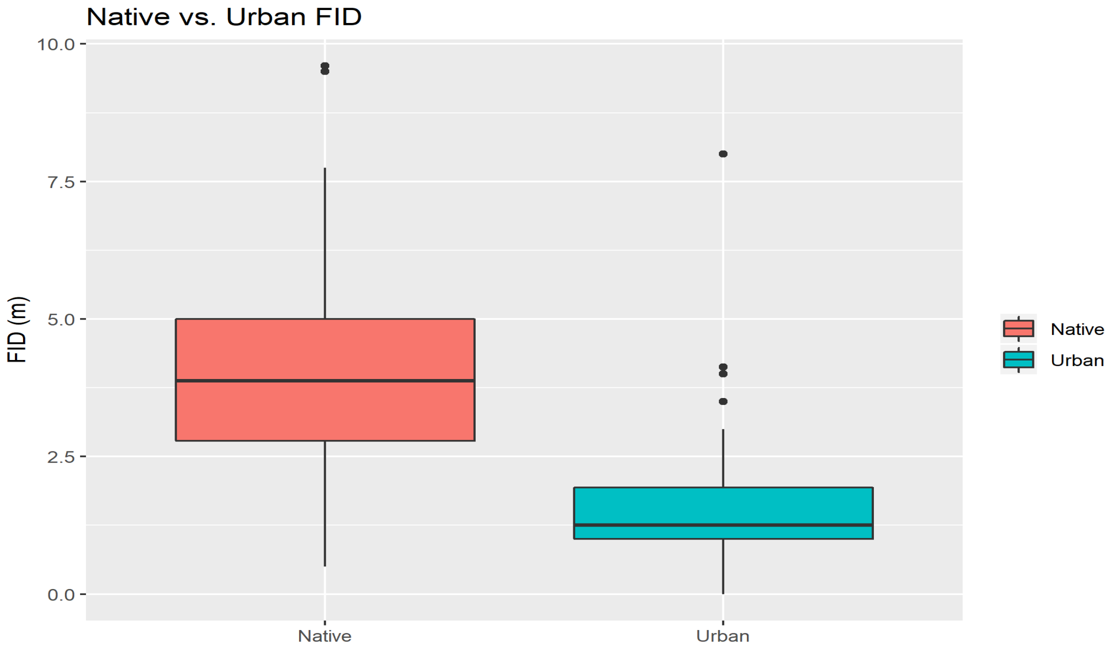
The end result of this project will (hopefully) be a multiple linear regression that combines all of the relationships with FID. Ideally, I would like to create a function that takes user input for a variety of factors and prints out a predicted FID.

Listing 3: Code for plots.

```
setwd("C:\\Users\\omela\\Documents\\EEB_C177_Notes")
FID_2 <- read.csv(file = "FID_Data.csv")
library(ggplot2)
library(dplyr)
library(tidyverse)
library(tidyr)
library(stringr)
names(FID_2)[7] <- "UrbanNative"
names(FID_2)[11] <- "FIDmeters"

#input: dataset of interest, xcoord is the independent
    variable, ycoord is the dependent variable, and ytitle
     and plottitle are pretty self-explanatory
#also saves plot as image to working directory and names
    it with imagename
BoxPlotMaker <- function(dataset, xcoord, ycoord, ytitle,
     plottitle, imagename) {
  #create a plot with the dataset of interest and the
      variables of interest
  plot <- ggplot(data=dataset, aes(x = xcoord, y = ycoord
      ))
  #make the plot a boxplot with boxes of different colors
       for each independent variable
  plot + geom_boxplot(aes(fill = xcoord)) +
  #insert the label for the y-axis and leave the x-axis
      and legend labels blank
  ylab(ytitle) + xlab("") + labs(fill="") + ggtitle(
      plottitle)
  #save the plot to the working directory as a png file
  ggsave(imagename)
  }
BoxPlotMaker(FID_2, FID_2$UrbanNative, FID_2$FIDmeters, "
    FID (m)", "Native vs. Urban FID", "UrbanNativeFID.png"
    )

#make a scatterplot with smoothers and saves image of it
    to working directiry
#input: dataset of interest, independent variable,
    dependent variable, axis labels, plot title, and name
    desired for image of plot
GGHist <- function(dataset, xcoord, ycoord, xlabel,
    ylabel, plottitle, imagename) {
  #make a plot with desired independent and dependent
      variables
```

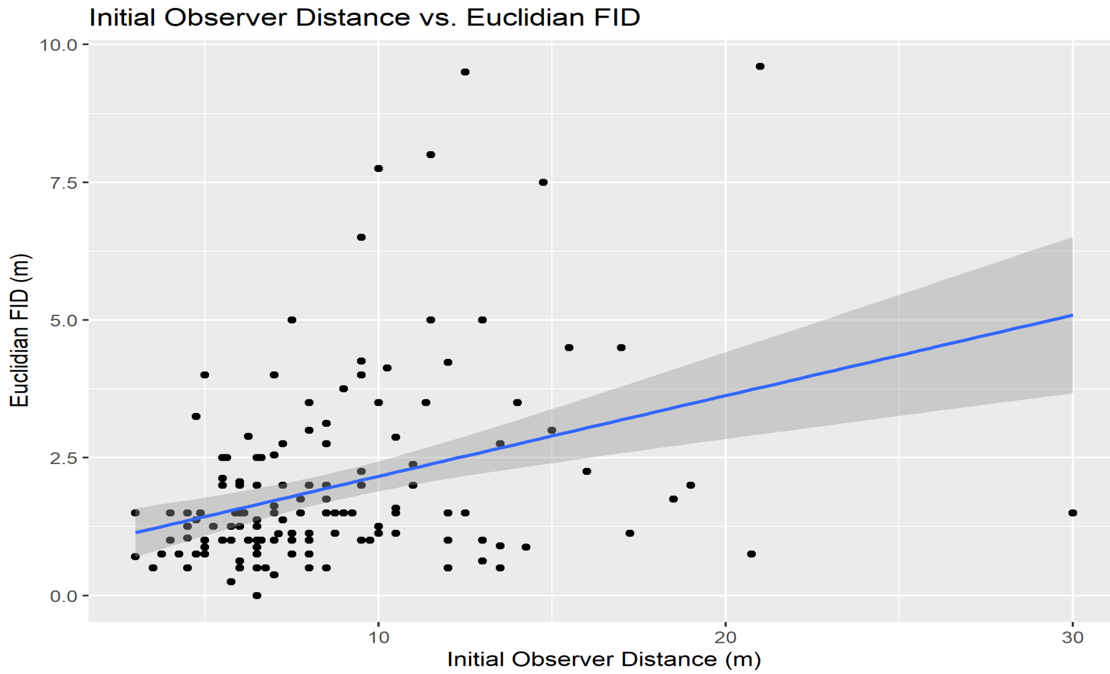
```
  plot <- ggplot(data = dataset, aes(x = xcoord, y =
      ycoord)) +
  #add axis labels, plot title, and data points
    xlab(xlabel) + ylab(ylabel) + ggtitle(plottitle) +
        geom_point()
  #add smoothers in a fashion that follows a linear model
  plot + geom_smooth(method="lm")
  #save image of plot to working directory with desired
      filename
  ggsave(imagename)

  }
GGHist(FID_2, FID_2$Initial.Distance..m., FID_2$Euc.FID,
    "Initial Observer Distance (m)", "Euclidian FID (m)",
    "Initial Observer Distance vs. Euclidian FID", "
    InitDistFID.png")
```

Figure 1: Comparison of FID measurements for native vs. urban juncos.



[h]

Figure 2: Linear model of relationship between initial observer distance and Euclidian FID.



References

[1] Dummy variable (statistics) - Wikiversity.

[2] Multiple Regression with Categorical Variables.

[3] Daniel T. Blumstein. Flight-Initiation Distance in Birds Is Dependent on Intruder Starting Distance. *The Journal of Wildlife Management*, 67(4):852, October 2003.

[4] B. B. M. Wong and U. Candolin. Behavioral responses to changing environments. *Behavioral Ecology*, 26(3):665–673, May 2015.